

Інститут кібернетики імені В. М. Глушкова
Національна академія наук України

Інститут кібернетики імені В. М. Глушкова
Національна академія наук України

Кваліфікаційна наукова
праця на правах рукопису

БОГАЄНКО ВСЕВОЛОД ОЛЕКСАНДРОВИЧ

УДК 517.9:519.6

ДИСЕРТАЦІЯ

**МАТЕМАТИЧНЕ ТА КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ
ГІДРОГЕОМІГРАЦІЙНИХ ПРОЦЕСІВ З НЕКЛАСИЧНОЮ ДИНАМІКОЮ
НА ОСНОВІ ВИСОКОПРОДУКТИВНИХ ОБЧИСЛЮВАЛЬНИХ
АЛГОРИТМІВ**

01.05.02 — математичне моделювання та обчислювальні методи

11 — Математика та статистика

Подається на здобуття наукового ступеня
доктора фізико-математичних наук

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

В. О. Богаєнко

Київ - 2024

АНОТАЦІЯ

Богаєнко В. О. Математичне та комп'ютерне моделювання гідрогеоміграційних процесів з некласичною динамікою на основі високопродуктивних обчислювальних алгоритмів. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора фізико-математичних наук за спеціальністю 01.05.02 "Математичне моделювання та обчислювальні методи" (11 – Математика та статистика). – Інститут кібернетики імені В. М. Глушкова НАН України, Київ, 2024.

Дисертаційна робота присвячена розвитку засобів математичного моделювання аномальних гідрогеоміграційних процесів на основі дробово-диференціального підходу та підвищенню обчислювальної ефективності при їх комп'ютерному моделюванні. У роботі вперше побудовані дробово-диференціальні математичні моделі та здійснено моделювання серії нелокальних процесів конвективної дифузії та фільтраційної консолідації. Відповідні моделі, на відміну від існуючих, враховують фактори часової та просторової нелокальності перебігу процесів у середовищах фрактальної структури. Стосовно розроблених моделей, побудовано клас високопродуктивних алгоритмів розв'язання початково-крайових задач, зокрема, паралельних алгоритмів для систем зі спільною та розподіленою пам'яттями, які дозволяють суттєво збільшити швидкість моделювання відповідних процесів. У роботі також розглядаються питання ідентифікації параметрів розроблених моделей. Комплексно застосовуючи на практиці отримані результати вирішені задачі моделювання вологоперенесення при зрошенні дощуванням в складних гідрогеологічних умовах.

У **першому розділі** наведено деякі відомості про сучасний стан розвитку апарату дробового інтегро-диференціювання, а також відповідних математичних моделей міграційних процесів. Проаналізовані чисельні методи, що використовуються для розв'язання відповідних початково-крайових задач, та підходи до побудови ефективних обчислювальних алгоритмів. Наведено також

основні підходи до розв'язання задач ідентифікації параметрів дробово-диференціальних моделей, що є принциповим елементом при їх практичному застосуванні.

У **другому розділі** наведені результати математичного моделювання процесів конвективної дифузії у двовимірному наближенні на основі моделей, що містять похідні дробового порядку за часовою змінною. Чисельні розв'язки початково-крайових задач моделювання динаміки нелокальних процесів конвективної дифузії отримані з використанням локально-одновимірних скінченно-різницевої схем.

Сформульовані математичні моделі, поставлені початково-крайові задачі та отримано і досліджено особливості чисельних розв'язків задач моделювання динаміки процесів нелокальної у часі релаксаційної конвективної дифузії та конвективної дифузії з урахуванням явища термодифузії. Розглянуто модель на основі k -похідної Капуто, яка містить додатковий параметр у порівнянні з моделлю на основі стандартної похідної Капуто-Герасимова, що збільшує можливості її адаптації до фактичних умов перебігу процесів.

У **третьому розділі** наводяться результати математичного моделювання процесів фільтраційної консолідації ґрунтів у одновимірному наближенні на основі дробово-диференціальних моделей з використанням методу скінченних різниць для отримання чисельних розв'язків. Зокрема, сформульовано математичну модель та поставлено початково-крайову задачу моделювання нелокальних процесів фільтраційної консолідації з урахуванням явища хімічного осмосу. Відповідна модель містить похідні Капуто–Фабріціо з несингулярним ядром за часовою змінною як у рівнянні для поля фільтраційних напорів, так і для поля концентрації. Модель розглядається також у випадку врахування явища лінійної повзучості скелету ґрунтового масиву та фрактальності пористого середовища. В залежності від значення коефіцієнту фрактальної розмірності середовища, процеси, описувані моделлю, яка його враховує, можуть мати як субдифузійний, так і супердифузійний характер.

Четвертий розділ присвячено математичному моделюванню нелокальних конвективно-дифузійних процесів при врахуванні явищ масообміну між частинками

у мобільній та немобільній фазах згідно з різними законами кінетики масообміну як у одновимірному, так і у двовимірному наближенні.

Сформульовано одновимірну дробово-диференціальну модель конвективної дифузії з урахуванням масообміну, що містить дробово-диференціальне за просторовою змінною рівняння для фільтраційного напору та дробово-диференціальне за часовою змінною рівняння для поля концентрації солей у розчинній фазі. Щодо цієї моделі поставлено початково-крайову задачу та отримані чисельні розв'язки методом скінченних різниць.

Досліджується також двовимірна модель, що містить похідну Капуто–Катугампола за часовою змінною, в межах якої кінетика масообміну описується згідно з рівнянням нерівноважної оборотної сорбції при ізотермі Генрі. Результати чисельного розв'язання відповідної початково-крайової задачі показують, що варіація додаткового параметра похідної Капуто–Катугампола дозволяє отримувати як субдифузійну, так і супердифузійну картини розподілу полів концентрації. Досліджується аналогічна модель, що отримується на основі MIM (mobile-immobile media) підходу використовуючи рівняння динаміки відтоку частинок у немобільну фазу, яка містить похідну Капуто за часовою змінною, а також спрощений аналог цієї моделі, який отримується шляхом апроксимації похідної дробового порядку похідними цілого порядку. Використання спрощеної моделі дозволяє суттєво знизити час, що витрачається на моделювання, при певних обмеженнях на значення порядку дробової похідної.

З метою підтримки прийняття рішень при виборі моделі конвективної дифузії з масообміном, найбільш адекватної спостережуваному процесу, побудовано відповідний алгоритм.

У **п'ятому розділі** розглядаються оптимізовані обчислювальні схеми, зокрема паралельні алгоритми, для одно- та багатовимірних задач моделювання геоміграційних процесів, що містять похідні Капуто–Герасимова та Капуто–Фабріціо.

Запропоновано серію паралельних алгоритмів для систем з розподіленою пам'яттю для локально-одновимірних схем розщеплення у яких був використаний червоно-чорний двовимірний блоковий розподіл даних.

Розроблено паралельні алгоритми для графічних процесорів (GPU) у випадку моделей з похідними Капуто–Герасимова як за часовою, так і за просторовими змінними, зокрема, гібридний алгоритм, у якому обчислення виконуються паралельно на графічному (GPU) та центральному (CPU) процесорах. Демонструється лінійне зростання прискорення алгоритмів при зростанні розміру сітки.

Для збільшення швидкодії обчислювальних схем моделювання процесів тепло-та масоперенесення на основі моделей з похідною Капуто–Герасимова за часовою змінною запропонована процедура її наближення з заданою точністю на основі розкладення у ряди та методики розділення змінних. Процедура дозволяє знизити обчислювальну складність розв'язання задач на одному кроці за часом з лінійної до логарифмічної. При моделюванні початкових стадій процесів, експериментально доведена ефективність додаткового застосування динамічної зміни кроку за часом з підбором довжини кроку методом проб і помилок.

Процедура наближення похідної Капуто–Герасимова на основі розкладення у ряди була поширена також на випадок двовимірної дробово-диференціальної моделі геоміграційних процесів, яка містить відповідні похідні за просторовими змінними. У цьому випадку розроблені паралельні схеми розв'язання початково-крайової задачі, яка дискретизувалась за локально-одновимірною схемою, на системах з розподіленою пам'яттю, зокрема, запропонована схема розподілення даних, що є поєднанням одновимірної та червоно-чорної схем, і має більшу за схему з одновимірним розподілом швидкодію при невеликій кількості залучених обчислювальних ресурсів, та більшу масштабованість у порівнянні з червоно-чорною схемою.

У шостому розділі обчислювальну схему, що базується на розкладанні у ряди ядра інтегрального оператора, розроблено для дробово-диференціальних рівнянь, що містять ψ -похідну Капуто. Проведені обчислювальні експерименти показали, що

використання такої схеми дозволяє досягти 2,5-кратного прискорення при моделюванні нелокального за часом процесу дифузії. Для досягнення додаткового прискорення проведення обчислень, була запропонована серія GPU-алгоритмів, що використовують можливості низькорівневої оптимізації коду, зокрема, векторні арифметичні операції, типи даних різної точності, тензорні операції.

У випадку моделей, що містять ψ -похідні Капуто за просторовими змінними, запропонована методика розкладення у ряди дозволяє зменшити обчислювальну складність виконання операцій множення матриць на вектори при розв'язанні ітераційними методами систем лінійних алгебраїчних рівнянь (СЛАР), що виникають при скінченно-різницевій дискретизації відповідних початково-крайових задач. Для випадку тривимірної моделі аномальної дифузії з ψ -похідними Капуто за просторовими змінними також побудовані та досліджені паралельні алгоритми для систем з розподіленою пам'яттю, які використовують дві схеми розподілу даних – одномірну блокову та червоно-чорну.

Для тривимірного рівняння дифузії з ψ -похідними Капуто як за часом, так і за просторовими змінними досліджено точність та швидкодію неявних скінченно-різницевих схем та схем розщеплення при їх застосуванні разом з алгоритмами, направленими на підвищення швидкодії проведення обчислень. На основі отриманих оцінок точності та швидкодії представлено алгоритм автоматичного вибору оптимальної обчислювальної схеми.

Підхід розкладення у ряди з подальшим розділенням змінних був застосований також для побудови схеми обчислення значень похідної Атангана–Балеану. Отримані теоретичні та експериментальні оцінки його точності та швидкодії показують, що така процедура також дозволяє зменшити обчислювальну складність до близької до лінійної без значних втрат точності.

Сьомий розділ присвячено алгоритмам розв'язання задачі ідентифікації параметрів дробово-диференціальних рівнянь. Розглядається узагальнене рівняння вологоперенесення у термінах напорів, яке містить ψ -похідні Капуто за часовою та просторовою змінними. Для розв'язання задачі знаходження значень числових параметрів моделі за яких вона якнайточніше описує дані моніторингу процесу

вологоперенесення пропонується алгоритм методу рою частинок. Для випадку, коли функціональний параметр ψ -похідної Капуто розглядається у довільній формі, пропонується алгоритм генетичного програмування для її знаходження.

У **восьмому розділі** наводяться результати дробово-диференціального моделювання вологоперенесення при розв'язанні задач, що виникають у землеробстві при управлінні зрошенням.

За відсутності експериментально визначених параметрів або даних про гранулометричний склад ґрунту, ідентифікувати параметри невідомих коефіцієнтів моделей пропонується розв'язуючи обернені задачі методами, описаними у сьомому розділі.

Проведене для умов зрошення дощуванням на півдні України моделювання вологоперенесення та порівняння його точності при використанні класичного одновимірного рівняння Річардса та його дробово-диференціального узагальнення свідчать, що використання дробово-диференціальної моделі дозволило на 7% збільшити точність опису динаміки напорів.

Застосування дробово-диференціальної моделі вологоперенесення з ψ -похідними Капуто за часовою та просторовою змінними для опису та прогнозування динаміки вологості ґрунту в складних гідрогеологічних умовах показало її здатність додатково значним чином підвищувати точність моделювання. Зокрема, використання сигмоподібної функції у якості функціонального параметра у ψ -похідній Капуто у порівнянні з використанням базової похідної Капуто–Герасимова дозволило отримати до 2,6-кратне зниження середньої абсолютної похибки. Підбір аналітичної форми функціонального параметра дробової ψ -похідної Капуто на основі алгоритму генетичного програмування дозволив додатково отримати до 34% підвищення точності моделювання.

Ключові слова: математичне моделювання, комп'ютерне моделювання, гідрогеоміграційні процеси, некласична динаміка, дробово-диференціальні моделі, високопродуктивні обчислення, паралельні алгоритми

ABSTRACT

Bohaienko V. O. Mathematical and computer modeling of hydrogeomigration processes with non-classical dynamics based on high-performance computational algorithms. – Qualifying scientific work under the rights of a manuscript.

The dissertation for obtaining the scientific degree of Doctor of Physical and Mathematical Sciences by specialty 01.05.02 "Mathematical modeling and computational methods" (11 – Mathematics and statistics). – V.M. Glushkov Institute of Cybernetics of National Academy of Sciences of Ukraine, Kyiv, 2024.

The dissertation is devoted to the development of the tools for mathematical modeling of anomalous hydrogeomigration processes based on the fractional-differential approach and to the increase of computational efficiency in their computer modeling. In the work, fractional-differential mathematical models were built and simulations of a series of non-local processes of convective diffusion and filtration-consolidation were carried out for the first time. The corresponding models, when compared to the existing ones, take into account the factors of temporal and spatial non-locality of the processes in fractal-structured media. For the developed models a class of high-performance algorithms for solving initial-boundary problems has been built. In particular, it includes parallel algorithms for shared and distributed memory systems, which allow to significantly increase the speed of simulation of the relevant processes. The dissertation also considers parameter identification problems for the developed models. By comprehensively applying the obtained results in practice, the problems of modeling moisture transport under sprinkler irrigation in complex hydrogeological conditions are solved.

In the first chapter, information on the current state of development of the apparatus of fractional integro-differentiation and the corresponding mathematical models of migration processes is given. Numerical methods used to solve the corresponding initial-boundary value problems are analyzed along with the approaches used for building efficient computational algorithms. The main approaches to solve the problems of parameters identification for fractional-differential models, which is a fundamental element in their practical application, are also described.

In the second chapter we present the results of mathematical modeling of convective diffusion processes in a two-dimensional setting based on the models containing fractional derivatives with respect to the time variable. Numerical solutions of initial-boundary value problems of modeling the dynamics of non-local convective diffusion processes are obtained using locally one-dimensional finite difference schemes.

Mathematical models and initial-boundary value problems were stated and the peculiarities of numerical solutions of the problems of modeling the dynamics of non-local convective diffusion with time relaxation and convective diffusion taking into account the phenomenon of thermodiffusion were obtained and studied. The model based on the Caputo k -derivative, which contains an additional parameter compared to the model based on the standard Caputo–Gerasimov derivative, is considered. The usage of such a model increases possibilities of its adaptation to the actual conditions of processes.

The third chapter presents the results of mathematical modeling of soil filtration-consolidation processes in one-dimensional setting based on fractional-differential models using the finite difference method to obtain numerical solutions. In particular, a mathematical model is formulated and the initial-boundary value problem for modeling non-local processes of filtration-consolidation taking into account the phenomenon of chemical osmosis is stated. The corresponding model contains the Caputo–Fabrizio derivatives with a non-singular kernel with respect to the time variable both in the equation for the filtration pressure field and for the concentration field. The model is also considered in the case of taking into account the phenomenon of soil skeleton's linear creep and the fractality of porous medium. Depending on the value of medium's fractal dimension coefficient, the processes described by the model that considers it can have both subdiffusive and superdiffusive behavior.

The fourth chapter is dedicated to the mathematical modeling of non-local convective diffusion processes taking into account the phenomenon of mass transfer between particles in mobile and immobile phases according to different laws of mass transfer kinetics in both one-dimensional and two-dimensional approximations.

We formulate a one-dimensional fractional-differential model of convective diffusion that takes into account mass transfer and contains a fractional-differential with

respect to the spatial variable equation for the filtration pressure and a fractional-differential with respect to the time variable equation for the field of the concentration of salt in soluble phase. Regarding this model an initial-boundary value problem is stated and numerical solutions are obtained using the finite difference method.

A two-dimensional model containing the Caputo–Katugampola derivative with respect to the time variable, within which the mass transfer kinetics is described according to the equation of non-equilibrium reversible sorption with Henry isotherm, is also studied. The results of numerical solution of the corresponding initial-boundary value problem show that the variation of the additional parameter of the Caputo–Katugampola derivative allows obtaining both subdiffusive and superdiffusive patterns of the distribution of concentration fields. A similar model is obtained on the basis of the MIM (mobile-immobile media) approach to describe the dynamics of particles' outflow into the immobile phase. It contains the Caputo derivative with respect to the time variable. We also study a simplified analogue of this model obtained by approximating the derivative of fractional order by derivatives of integer order. The use of the simplified model makes it possible to significantly reduce the time spent on modeling within certain restrictions on the value of fractional derivative's order.

In order to support decision-making when choosing a model of convective diffusion with mass transfer, which is the most adequate to the observed process, an appropriate algorithm was built.

In the fifth chapter we consider optimized computational schemes, in particular parallel algorithms, for one- and multidimensional problems of modeling geomigration processes, containing the Caputo–Gerasimov and Caputo–Fabrizio derivatives.

A series of parallel algorithms for distributed memory systems is proposed for locally one-dimensional splitting schemes in which red-black two-dimensional block distribution of data is used.

Parallel algorithms for graphics processors (GPUs) have been developed in the case of the models with the Caputo–Gerasimov derivatives with respect to both the time and spatial variables, in particular, a hybrid algorithm in which calculations are performed in

parallel on GPUs and central processing units (CPU). Linear increase in the acceleration of the algorithms is observed here with the increase of grid size.

To increase the speed of computational schemes for modeling heat and mass transfer processes based on the models with the Caputo–Gerasimov derivative with respect to the time variable, a procedure for its approximation with a given accuracy based on expansion into series and variable separation techniques is proposed. The procedure makes it possible to reduce computational complexity of solving problems on one time step from linear to logarithmic. When modeling the initial stages of the processes, the effectiveness of additional application of dynamic time step change with the selection of step length by trial and error technique has been experimentally proven.

The usage of the procedure for approximating the Caputo–Gerasimov derivative based on expansion into series was also extended to the case of a two-dimensional fractional-differential model of geomigration processes, which contains the corresponding derivatives with respect to the spatial variables. In this case, parallel schemes for solving the initial-boundary value problem, which was discretized according to a locally one-dimensional scheme, were developed for distributed memory systems. In particular, we propose a data distribution scheme that is a combination of one-dimensional and red-black schemes. It has higher speed compared to the scheme with the one-dimensional distribution when a small amount of computational resources is involved and greater scalability compared to the red-black scheme.

In the sixth chapter, a computational scheme based on the expansion into series of integral operators' kernels is developed for fractional-differential equations containing the ψ -Caputo derivative. The conducted computational experiments showed that the use of such a scheme allows to achieve a 2,5-times acceleration when modeling the non-local in time diffusion process. In order to achieve additional acceleration of computations, we propose a series of GPU-algorithms that employ the possibilities of low-level code optimization, in particular, vector arithmetic operations, data types of different precision, and tensor operations.

In the case of the models containing the ψ -Caputo derivatives with respect to the spatial variables, the proposed method of the expansion into series allows to reduce

computational complexity of performing the operations of multiplying matrices by vectors when solving by iterative methods the systems of linear algebraic equations arising from finite-difference discretization of corresponding initial-boundary value problems. For the case of the three-dimensional model of anomalous diffusion with the ψ -Caputo derivatives with respect to the spatial variables, parallel algorithms for distributed memory systems that use two data distribution schemes – a one-dimensional block one and a red-black one – are also constructed and studied.

For the three-dimensional diffusion equation with the ψ -Caputo derivatives with respect to both the time and spatial variables, the accuracy and speed of implicit finite-difference schemes and splitting schemes when applied together with the algorithms aimed at increasing the speed of computations were studied. Based on the obtained accuracy and speed estimates, an algorithm for automatic selection of the optimal computational scheme is presented.

The approach of the expansion into series with subsequent separation of variables is also used to construct a scheme for calculating the values of the Atangana–Baleanu derivative. The obtained theoretical and experimental estimates of its accuracy and speed show that such a procedure also allows reducing the computational complexity to the order close to linear without significant loss of accuracy.

The seventh chapter is devoted to the algorithms for solving parameters identification problems for fractional-differential equations. We consider the generalized moisture transport equation stated in terms of pressure that contains the ψ -Caputo derivatives with respect to the time and space variables. To solve the problem of finding the values of model's numerical parameters for which it the most accurately describes monitoring data of moisture transport process, the particle swarm optimization algorithm is proposed. For the case when the functional parameter of the ψ -Caputo derivative is considered in an arbitrary form, a genetic programming algorithm is proposed for its selection.

The eighth chapter presents the results of fractional-differential modeling of moisture transport when solving problems that arise in agriculture during irrigation management.

In the absence of experimentally determined parameters or data on the granulometric composition of soil, it is proposed to identify the parameters of the unknown model's coefficients by solving inverse problems using the methods described in the seventh chapter.

The simulation of moisture transport carried out for the conditions of sprinkler irrigation in the south of Ukraine and a comparison of its accuracy when using the classic one-dimensional Richards equation and its fractional-differential generalization show that the use of the fractional-differential model made it possible to increase the accuracy of pressure dynamics description by 7%.

The application of the fractional-differential model of moisture transport with the ψ -Caputo derivatives with respect to the time and spatial variables to describe and predict the dynamics of soil moisture in complex hydrogeological conditions showed its ability to significantly increase the accuracy of modeling. In particular, the use of a sigmoid function as a functional parameter in the ψ -Caputo derivative compared to the use of the basic Caputo–Gerasimov derivative made it possible to obtain a 2,6-times decrease in average absolute error. Selection of an analytical form of the functional parameter of the ψ -Caputo derivative using the genetic programming algorithm made it possible to additionally obtain up to 34% increase in modeling accuracy.

Keywords: mathematical modeling, computer modeling, hydrogeomigration processes, non-classical dynamics, fractional-differential models, high-performance computing, parallel algorithms

Список публікацій здобувача

Наукові праці, в яких опубліковані основні наукові результати дисертації:

1) Bulavatsky VM, Bohaienko VO. Mathematical Modeling of the Fractional Differential Dynamics of the Relaxation Process of Convective Diffusion Under Conditions of Planned Filtration. *Cybernetics and Systems Analysis* 2015; 51:886–895. doi: [10.1007/s10559-015-9781-2](https://doi.org/10.1007/s10559-015-9781-2) [Scopus, Web of Science, zbMATH]

2) Bulavatsky VM, Bohaienko VO. Mathematical Modeling of the Dynamics of Nonequilibrium in Time Convection–Diffusion Processes in Domains with Free

Boundaries. Cybernetics and Systems Analysis 2016; 52:427–440. doi: [10.1007/s10559-016-9843-0](https://doi.org/10.1007/s10559-016-9843-0) [Scopus, Web of Science, zbMATH, MathSciNet]

3) Bohaienko VO, Bulavatsky VM, Kryvonos IuH. On Mathematical modeling of Fractional-Differential Dynamics of Flushing Process for Saline Soils with Parallel Algorithms. Journal of Automation and Information Sciences 2016; 10:1-12. doi: [10.1615/JautomatInfScien.v48.i10.10](https://doi.org/10.1615/JautomatInfScien.v48.i10.10) [Scopus]

4) Булавацкий ВМ, Богаенко ВА. Численное моделирование дробно-дифференциальной динамики процесса фильтрационно-конвективной диффузии на основе параллельных алгоритмов для кластерных систем. Доповіді НАНУ 2017; 1: 21-28. doi: [10.15407/dopovidi2017.01.021](https://doi.org/10.15407/dopovidi2017.01.021) [zbMATH, MathSciNet]

5) Bohaienko VO, Bulavatsky VM, Kryvonos IuH. Mathematical Modeling of Fractional-Differential Dynamics of Process of Filtration-Convective Diffusion of Soluble Substances in Nonisothermal Conditions. Journal of Automation and Information Sciences 2017; 49(4):12-25. doi: [10.1615/JautomatInfScien.v49.i4.20](https://doi.org/10.1615/JautomatInfScien.v49.i4.20) [Scopus, MathSciNet]

6) Bulavatsky VM, Bohaienko VO. Numerical Simulation of Fractional-Differential Filtration-Consolidation Dynamics Within the Framework of Models with Non-Singular Kernel. Cybernetics and Systems Analysis 2018; 54:193–204. doi: [10.1007/s10559-018-0020-5](https://doi.org/10.1007/s10559-018-0020-5) [Scopus, Web of Science, zbMATH, MathSciNet]

7) Булавацкий ВМ, Богаенко ВА. Компьютерное моделирование дробно-дифференциальной динамики некоторых фильтрационно-консолидационных процессов. Доповіді НАНУ 2018; 4:16-24. doi: [10.15407/dopovidi2018.04.016](https://doi.org/10.15407/dopovidi2018.04.016) [MathSciNet]

8) Bohaienko VO. Parallel Algorithms for Modelling Two-Dimensional Non-Equilibrium Salt Transfer Processes on the Base of Fractional Derivative Model. Fractional calculus and applied analysis 2018; 21(3):654–671. doi: [10.1515/fca-2018-0035](https://doi.org/10.1515/fca-2018-0035) [Scopus, Web of Science, zbMATH, MathSciNet]

9) Bohaienko VO, Bulavatsky VM. Mathematical Modeling of Solutes Migration Under the Conditions of Groundwater Filtration by the Model with the k-Caputo Fractional Derivative. Fractal Fract. 2018; 2(4):28. doi: [10.3390/fractalfract2040028](https://doi.org/10.3390/fractalfract2040028) [Scopus, Web of Science]

10) Bohaienko VO. Numerical schemes for modelling time-fractional dynamics of non-isothermal diffusion in soils. *Mathematics and Computers in Simulation* 2019; 157: 100–114. doi: [10.1016/j.matcom.2018.09.025](https://doi.org/10.1016/j.matcom.2018.09.025) [Scopus, Web of Science, DBLP, zbMATH, MathSciNet]

11) Булавацкий ВМ, Богаенко ВА. Компьютерное моделирование динамики процесса миграции растворимых веществ при фильтрации грунтовых вод со свободной поверхностью на основе дробно-дифференциального подхода. *Доповіді НАНУ* 2018; 12:21-29. doi: [10.15407/dopovidi2018.12.021](https://doi.org/10.15407/dopovidi2018.12.021) [zbMATH, MathSciNet]

12) Bohaienko VO. A fast finite-difference algorithm for solving space-fractional filtration equation with a generalised Caputo derivative. *Computational and Applied Mathematics* 2019; 38:105. doi: [10.1007/s40314-019-0878-5](https://doi.org/10.1007/s40314-019-0878-5) [Scopus, Web of Science, DBLP, zbMATH, MathSciNet]

13) Богаенко ВА, Булавацкий ВМ. Компьютерное моделирование на основе нелокальной модели динамики конвективной диффузии растворимых веществ в подземном фильтрационном потоке в условиях массообмена. *Международный научно-технический журнал “Проблемы управления и информатики”* 2019; 3:41-53.

14) Bohaienko VO, Bulavatsky VM. Simplified Mathematical Model for the Description of Anomalous Migration of Soluble Substances in Vertical Filtration Flow. *Fractal Fract.* 2020; 4: 20. doi: [10.3390/fractalfract4020020](https://doi.org/10.3390/fractalfract4020020) [Scopus, Web of Science]

15) Bulavatsky VM, Bohaienko VO. Some boundary-value problems of fractional-differential mobile-immobile migration dynamics in a profile filtration flow. *Cybernetics and Systems Analysis* 2020; 56(3): 410–425. doi: [10.1007/s10559-020-00257-2](https://doi.org/10.1007/s10559-020-00257-2) [Scopus, Web of Science, zbMATH, MathSciNet]

16) Bohaienko VO. Parallel finite-difference algorithms for three-dimensional space-fractional diffusion equation with ψ -Caputo derivatives. *Computational and Applied Mathematics* 2020; 39:163. doi: [10.1007/s40314-020-01191-x](https://doi.org/10.1007/s40314-020-01191-x) [Scopus, Web of Science, DBLP, zbMATH, MathSciNet]

17) Bohaienko VO, Gladky AV, Romashchenko MI, Matiash TV. A Identification of fractional water transport model with ψ -Caputo derivatives using particle swarm

optimization algorithm. Applied Mathematics and Computation 2021; 390:125665. doi: [10.1016/j.amc.2020.125665](https://doi.org/10.1016/j.amc.2020.125665) [Scopus, Web of Science, DBLP, zbMATH, MathSciNet]

18) Bohaienko VO. Accuracy and speed of splitting methods for three-dimensional space-time fractional diffusion equation with ψ -Caputo derivatives. Mathematics and Computers in Simulation 2021; 188: 226-240. doi: [10.1016/j.matcom.2021.04.004](https://doi.org/10.1016/j.matcom.2021.04.004) [Scopus, Web of Science, DBLP, zbMATH, MathSciNet]

19) Bohaienko VO. On the recurrent computation of fractional operator with Mittag-Leffler kernel. Applied Numerical Mathematics 2021; 162: 137-149. doi: [10.1016/j.apnum.2020.12.016](https://doi.org/10.1016/j.apnum.2020.12.016) [Scopus, Web of Science, zbMATH, MathSciNet]

20) Bohaienko VO. Selection of ψ -Caputo derivative functional parameter in generalized water transport equation by genetic programming technique. Results in Control and Optimization 2021; 5: 100068. doi: [10.1016/j.rico.2021.100068](https://doi.org/10.1016/j.rico.2021.100068) [Scopus, Web of Science]

21) Bohaienko VO, Bulavatsky VM. Fractional-fractal modeling of filtration-consolidation processes in saline saturated soils. Fractal and Fractional 2020; 4(4): 59. doi: [10.3390/fractalfract4040059](https://doi.org/10.3390/fractalfract4040059) [Scopus, Web of Science]

22) Bohaienko V, Gladky A. Modelling fractional-order moisture transport in irrigation using artificial neural networks. SeMA 2023. doi: [10.1007/s40324-023-00322-8](https://doi.org/10.1007/s40324-023-00322-8) [Scopus]

23) Богаєнко ВО, Булавацький ВМ, Хіміч ОМ. Математичне та комп'ютерне моделювання в задачах гідрогеоміграційної динаміки. Київ: Наукова Думка, 2022.

Праці апробаційного характеру:

24) Bohaienko VO. Numerical Integration Schemes for Finite Difference Solution of Time-Fractional Diffusion Equation with Generalized Caputo Derivative. "Інформаційні технології та комп'ютерне моделювання"; матеріали статей Міжнародної науково-практичної конференції, м. Івано-Франківськ, 14-19 травня 2018 року. Івано-Франківськ: п. Голіней О.М., 2018, с.250-253.

25) Bohaienko VO, Bulavatsky VM, Gladky AV. GPU algorithms for solving time-fractional diffusion equation with generalised Caputo derivative with respect to a function.

Fifth International Conference "High Performance Computing" HPC-UA 2018 (Ukraine, Kyiv, October 22-23, 2018), 2018, p.12-17.

26) Bohaienko VO. Efficient computation schemes for generalized two-dimensional time-fractional diffusion equation. "Інформаційні технології та комп'ютерне моделювання"; матеріали статей Міжнародної науково-практичної конференції, м. Івано-Франківськ, 20-25 травня 2019 року. Івано-Франківськ: п. Голіней О.М., 2019, с.238-241.

27) Bohaienko VO. Performance of vectorized GPU-algorithm for computing ψ -Caputo derivative values. Hu Z., Petoukhov S., Dychka I., He M. (eds) Advances in Computer Science for Engineering and Education III. ICCSEEA 2020. Advances in Intelligent Systems and Computing, vol 1247. Cham: Springer, 2020, p. 266-275 doi: [10.1007/978-3-030-55506-1_24](https://doi.org/10.1007/978-3-030-55506-1_24) [Scopus]

28) Bohaienko VO. Computing ψ -Caputo Fractional Derivative Values Using CUDA 10. Proceedings of the 9th International Conference "Information Control Systems & Technologies" Odessa, Ukraine, September 24–26, 2020. CEUR Workshop proceedings, vol. 2711, 2020, p. 49. [Scopus, DBLP]

29) Bohaienko VO, Gladky AV. On the selection of fractional-differential model of convective diffusion with mass exchange. Modeling, Control and Information Technologies: Proceedings of International Scientific and Practical Conference 2020; 4:7-10. doi: [10.31713/MCIT.2020.02](https://doi.org/10.31713/MCIT.2020.02)

ЗМІСТ

Вступ	22
Розділ 1. Аналіз сучасного стану проблеми математичного і комп'ютерного моделювання гідрогеоміграційних процесів з некласичною динамікою	32
1.1. Короткий огляд основних означень апарату інтегро-диференціювання дробового порядку	33
1.2. Дробово-диференціальні моделі процесів гідрогеоміграції.....	36
1.3. Деякі особливості чисельного розв'язання початково-крайових задач для дробово-диференціальних рівнянь.....	42
1.4. Підходи до ефективної дискретизації дробових похідних	44
1.5. Високопродуктивні алгоритми дробово-диференціального моделювання.....	46
1.6. Методи ідентифікації параметрів дробово-диференціальних моделей.....	47
1.7. Висновки до розділу 1	48
Розділ 2. Математичне моделювання дробово-диференціальної динаміки процесів конвективної дифузії розчинних речовин при фільтрації з джерел забруднень ґрунтових вод	50
2.1. Моделювання дробово-диференціальної динаміки процесу конвективної дифузії розчинних речовин при усталеній плановій фільтрації з кругового котловану.....	50
2.2. Математичне моделювання дробово-диференціальної динаміки процесу конвективної дифузії розчинних речовин при фільтрації з річок та каналів.....	58
2.3. Моделювання дробово-диференціальної динаміки процесу конвективної дифузії у неізотермічних умовах	67
2.4. Моделювання дробово-диференціальної динаміки процесу конвективної дифузії на основі моделі з k -похідною Капуто	76
2.5. Висновки до розділу 2	84

Розділ 3. Математичне моделювання нелокальних процесів фільтраційної консолідації ґрунтових середовищ на основі дробово-диференціальних моделей	86
3.1. Чисельне моделювання дробово-диференціальної фільтраційно-консолідаційної динаміки в рамках моделі з несингулярним ядром	86
3.2. Моделювання дробово-диференціальної фільтраційно-консолідаційної динаміки з урахуванням фактора повзучості	95
3.3. Математичне моделювання фільтраційно-консолідаційної динаміки в умовах солеперенесення на основі дробово-фрактальної моделі	100
3.4. Висновки до розділу 3	106
Розділ 4. Математичне моделювання нелокальних процесів конвективної-дифузійного масоперенесення в ґрунтах з урахуванням масообміну	109
4.1. Математичне моделювання дробово-диференціальної динаміки процесу промивки засолених ґрунтів у одновимірному наближенні	109
4.2. Математичне моделювання дробово-диференціальної динаміки конвективної дифузії розчинних речовин в умовах масообміну у двовимірному наближенні	116
4.3. Математичне моделювання дробово-диференціальної мобільно-немобільної міграційної динаміки у двовимірному наближенні	126
4.4. Моделювання дробово-диференціальної динаміки процесу конвективної дифузії в умовах масообміну на основі спрощеної математичної моделі	135
4.5. Алгоритм вибору дробово-диференціальної моделі конвективної дифузії з урахуванням масообміну	138
4.6. Висновки до розділу 4	147
Розділ 5. Високопродуктивні алгоритми моделювання гідрогеоміграційної динаміки на основі дробово-диференціальних моделей	150
5.1. Паралельні алгоритми схем розщеплення для систем з розподіленою пам'яттю	151
5.2. Паралельні алгоритми для графічних процесорів	155
5.2.1. Паралельний алгоритм схем розщеплення для задач з дробовою похідною за часовою змінною	155

5.2.2. Комбінований алгоритм для схем розщеплення.....	158
5.2.3. Паралельний алгоритм для одновимірних задач з дробовою похідною за просторовою змінною	161
5.2.4. Паралельні алгоритми для одновимірних задач з дробовою похідною за часовою змінною	163
5.3. Алгоритм оптимізації обчислень при моделюванні дробово-диференціальної за часовою змінною міграційної динаміки.....	172
5.3.1. Методика наближеного обчислення дробової похідної за часовою змінною	172
5.3.2. Процедура розв'язання у випадку змінної довжини кроків за часом.....	180
5.3.3. Швидкодія алгоритмів.....	182
5.4. Паралельний алгоритм та алгоритм оптимізації обчислень при моделюванні дробово-диференціальної за просторовими змінними гідрогеографічної динаміки.....	186
5.5 Висновки до розділу 5	197
Розділ 6. Високопродуктивні алгоритми моделювання гідрогеографічної динаміки на основі моделей, що містять ψ-похідну Капуто та похідну Атангана–Балеану	201
6.1. Методика наближеного обчислення ψ -похідної Капуто за часовою змінною	201
6.2. Паралельні алгоритми для графічних процесорів обчислення ψ -похідної Капуто за часовою змінною	206
6.3. Методика наближеного обчислення ψ -похідної Капуто за просторовою змінною	225
6.4. Обчислювальні схеми для рівняння дифузії з ψ -похідною Капуто за часовою та просторовими змінними.....	248
6.5. Паралельні алгоритми розв'язання тривимірного рівняння дифузії з ψ -похідної Капуто за просторовою змінною.....	268
6.6. Алгоритм оптимізації обчислень для моделей, що містять похідну Атангана–Балеану.....	282

6.7 Висновки до розділу 6	300
Розділ 7. Алгоритми розв’язання задач ідентифікації параметрів дробово-диференціальних моделей вологоперенесення	303
7.1. Початково-крайова задача та чисельний метод	303
7.2. Алгоритм рою частинок	308
7.3. Алгоритм методу генетичного програмування для пошуку аналітичної форми функціонального параметра ψ -похідної Капуто	311
7.4. Висновки до розділу 7	313
Розділ 8. Комп’ютерне моделювання та прогнозування динаміки вологоперенесення при зрошенні ґрунтових масивів дощуванням	315
8.1. Моделювання з використанням класичної моделі та моделі з похідної Капуто–Герасимова	316
8.2. Моделювання з використанням ψ -похідної Капуто з функціональним параметром фіксованої форми	338
8.3. Моделювання з використанням ψ -похідної Капуто та пошуком аналітичної форми функціонального параметра	352
8.4. Висновки до розділу 8	356
Висновки	359
Список використаних джерел	362
Додаток А. Результати тестування алгоритму оптимізації обчислень для моделей, що містять похідну Атангана–Балеану	394
Додаток Б. Довідка про використання результатів	397

Вступ

Актуальність теми.

Розробка методів математичного та комп'ютерного моделювання динаміки міграційних процесів у ґрунтових середовищах є актуальною проблемою, зокрема в аспекті охорони навколишнього середовища. При перебігу відповідних процесів у середовищах для яких характерні ефекти пам'яті та просторової нелокальності, актуальною стає також проблема підвищення ступеня адекватності існуючих кількісних моделей.

Математичне моделювання динаміки міграції вологи та солей в ґрунтах є широко поширеним і розвивається, переважно, у контексті постановок задач, характерних для класичної математичної фізики. Серед великої кількості публікацій з цього питання виділимо роботи І. І. Ляшка, С. І. Ляшка, Г. М. Положого, Г. Е. Мистецького, В. М. Булавацького, А. Я. Бомби, П. М. Матринюка, А. П. Власюка, М. М. Біляєва, М. М. Верігіна, О. В. Ликова, О. В. Голубевої, А. А. Глущенко, Г. І. Марчука, П. Я. Полубаринової-Кочиної, І. В. Сергієнка, В. С. Дейнеки, А. Ф. Чудновського, В. В. Скопецького, О. Ю. Грищенко, В. І. Лаврика, О. Я. Олійника, В. В. Оноцького, Д. А. Ключина, Я. Г. Савули, О. Б. Стелі, Є. Я. Чаплі, О. Ю. Чернухи, Я. Д. П'янило, М. М. Хлапука та ін.

У складних нерівноважних умовах, зокрема в середовищах фрактальної структури, до яких відносять і ґрунти, перебіг процесів перенесення не завжди коректно описується класичними математичними моделями. Прогрес у цьому випадку досягається, зокрема, з використанням підходу, що базується на застосуванні апарату дробово-диференціального числення. Його розвиток відбувається у напрямках розробки та дослідження нових операторів дробових інтегралів та похідних, їх застосуванню для опису фізичних процесів чи побудови феноменологічних моделей, у теорії керування, обробці зображень, штучному інтелекті тощо завдяки роботам багатьох відомих дослідників серед яких слід відзначити роботи А. А. Кілбаса, Ю. Лучко, А. А. Чикрія, А. Кочубея, С. Едельмана, С. Г. Самко, О. И. Маричева, А. Чечкіна, В. Тарасова, А. Нахушева, В. Кірьякової (V. Kiryakova), Х. Трухільо (J. Trujillo), І. Подлубного (I. Podlubny), В. Учайкіна,

Н. Вірченко, Р. Нігматтуліна, Г. Заславського, Р. Горенфло (R. Gorenflo), Р. Метцлера (R. Metzler), Ж. Тенрейру Машаду (J. Tenreiro Machado), Д. Балеану (D. Baleanu) та ін.

Відштовхуючись від узагальнень базових фізичних законів наразі побудована та досліджена значна кількість дробово-диференціальних міграційних моделей у пористих середовищах. Серед робіт присвячених цьому напрямку виокремимо роботи М. М. Меершерта (M. M. Meerschaert), Д. А. Бенсона (D. A. Benson), Д. Балеану (D. Baleanu), А. Атангани (A. Atangana), Б. Самета (B. Samet), Р. Алмейди (R. Almeida), А. Маліновської (A. Malinowska), В. М. Булавацького. Поєднання різноманітних підходів узагальнення фізичних законів переносу породжує широкі класи моделей, наприклад моделі з похідними Хільфера, Прабхакара, Капуто–Фабріціо, Атангана–Балеану. Ще одним новим напрямком розвитку моделювання аномальних міграційних процесів є використання похідних дробового порядку від функції за іншою функцією. Зауважимо, що враховуючи велику кількість можливих визначень дробових похідних та комплексність процесів у таких пористих середовищах, як ґрунти, розвиток апарату їх дробово-диференціального моделювання далекий від завершення.

Одним з найбільш розповсюджених методів чисельного розв'язання початково-крайових задач щодо дробово-диференціальних рівнянь є метод скінченних різниць якому присвячені роботи багатьох дослідників, зокрема І. Подлубного (I. Podlubny), К. Дітхельма (K. Diethelm), Р. Гарраппи (R. Garrappa), М. Стинеса (M. Stynes), Н. Форда (N. Ford), М. М. Меершерта (M. M. Meerschaert), Ч. Лі (Ch. Li), Ч. Гона (Ch. Gong), К. Оволабі (K. Owolabi), М. Х. Шканукова-Лафіщева, А. А. Аліханова та ін. Іншим підходом до моделювання є використання спектральних методів та методу Галеркіна, зокрема методу скінченних елементів. Серед авторів, що працюють у цьому напрямку виділимо А. Бхраві (A.N. Bhrawy), М. Ейнсворта (M. Ainsworth).

Одним з ефективних підходів у випадку багатовимірних задач є застосування схем розщеплення, зокрема неявних схем змінних напрямків (ADI, Alternating Direction Implicit), які отримали розвиток для дробово-диференціальних рівнянь у

роботах, зокрема, Ж. Вана (Zh. Wang), Ж. Хао (Zh. Hao), Д. Ху (D. Hu), Т. Алероєва. Застосування таких схем, окрім ефекту зниження обчислювальної складності та суттєвого зменшення зв'язаності блоків обчислень дозволяє ефективніше застосовувати методи розпаралелення.

Важливою обчислювальною проблемою, що виникає при чисельному розв'язанні початково-крайових задач для рівнянь з дробовими похідними за часовою змінною є необхідність обчислювати значення інтегральних операторів за мінімально можливий час з заданою точністю в умовах, коли інтервал інтегрування збільшується зі збільшенням номеру кроку за часом. Зменшення обчислювальної складності тут досягається в першу чергу застосуванням спеціальних дискретизацій та представлень операторів дробових похідних. Виділимо у цьому напрямку роботи Н. Форда (N. Ford), Ч. Гона (Ch. Gong), Є. Тиртишнікова, Д. Баффета (D. Baffet), Л. Юаня (L. Yuan), О. Агравала (O. Agrawal), К. Дитхельма (K. Diethelm), С. Ю. Лукашука, Р. К. Газізова, Н. С. Белевцова та інших. Зауважимо, що при використанні таких підходів, час розв'язання залишається близьким до постійного чи логарифмічно залежного від номера кроку за часом, що, проте, супроводжується зниженням точності отримуваних розв'язків.

Важливими в контексті практичного застосування дробово-диференціальних моделей аномальних процесів геоміграції, є питання ефективної програмної реалізації. Інтегро-диференціальний характер відповідних моделей дозволяє ефективно застосовувати для пришвидшення розрахунків засоби паралельних обчислень. Основним підходом, що застосовується при розв'язанні задач з похідними дробового порядку за часовою змінною, є використання розпаралелення при безпосередньому обчисленні значень дробових похідних. Виділимо у цьому напрямку роботи К. Дитхельма (K. Diethelm) та Є. Каслік (E. Kaslik). Якщо моделі містять дробові похідні за просторовими змінними, на перше місце висувається швидкодія алгоритмів розв'язання систем лінійних алгебраїчних рівнянь (СЛАР). Застосуванню паралельних алгоритмів такого класу присвячені, зокрема, роботи Ч. Гона (Ch. Gong) та В. Бао (W. Bao). Відзначимо, що наразі розробка ефективних паралельних алгоритмів чисельного розв'язання початково-крайових задач для

дробово-диференціальних рівнянь та їх систем є незавершеною і актуальною науковою проблемою.

Практичне використання дробово-диференціальних математичних моделей для прогнозування аномальної динаміки геогідроміграційних процесів вимагає проведення ідентифікації їх параметрів. Специфіка дробово-диференціальних моделей полягає у тому, що порядки та параметри дробових похідних не мають технічних засобів вимірювання. Для ідентифікації параметрів дробово-диференціальних моделей використовуються підходи аналітичного чи чисельно-аналітичного розв'язання обернених задач щодо відповідних моделей та підхід з використанням методів штучного інтелекту та метаевристичної оптимізації, зокрема методу мурашиних колоній, алгоритму зозулі, алгоритму рою частинок, штучних нейронних мереж. Варто відзначити, що більшість алгоритмів метаевристичної оптимізації є легко розпаралелюваними за даними.

Таким чином, сучасні засоби дробово-диференціального моделювання міграційних процесів охоплюють широке коло моделей, які містять інтегро-диференціальні оператори за часовою та просторовими змінними. При цьому актуальним напрямком є комплексне чисельне дослідження моделей з різнотипними дробовими похідними; визначення особливостей динаміки процесів, що ними описуються; розробка обчислювально ефективних засобів визначення моделей, найкращих для опису спостережуваного процесу, ідентифікації їх параметрів та прогнозування розвитку процесів на їх основі.

Науково-прикладною проблемою, розв'язанню якої присвячена дисертаційна робота, є розвиток методів математичного моделювання аномальних гідрогеоміграційних процесів на основі дробово-диференціального підходу та розвиток методів комп'ютерного моделювання з метою підвищення обчислювальної ефективності при чисельному моделюванні і прогнозуванні динаміки міграційних процесів та ідентифікації параметрів відповідних моделей. Зазначені процедури комп'ютерного моделювання базуються на обчислювальних методах та паралельних алгоритмах для скінченно-різницевої схем і схем розщеплення для багатовимірних

задач, з використанням, зокрема, підходу апроксимації ядер інтегро-диференціальних операторів.

Зв'язок з науковими програмами, планами, темами.

Дисертаційну роботу виконано у лабораторії методів математичного моделювання процесів екології та енергетики Інституту кібернетики імені В.М. Глушкова НАН України в рамках наукових тем “Розробити засоби математичного моделювання динаміки складних розподілених процесів стосовно задач екології на основі локальних та дробово-диференціальних моделей” (номер державної реєстрації 01194U002207693, 2019–2023 рр.), “Розробити теоретичні основи математичного моделювання геоєкологічних процесів на базі локальних та дробово-диференціальних моделей” (номер державної реєстрації 0114U002093, 2014–2018 рр.), “Розробити нові методи паралельної та розподіленої обробки надвеликих об'ємів даних для аналізу складних багатокомпонентних середовищ” (номер державної реєстрації 0114U0020930117U000471, 2017–2021 рр.), “Розробити методи математичного моделювання складних систем для суперкомп'ютерів” (номер державної реєстрації 0118U005230, 2018–2019 рр.), “Розробити методи високопродуктивних обчислень дослідження математичних моделей неоднорідних середовищ та обробки великих даних (Big Data) на основі суперкомп'ютерних технологій” (номер державної реєстрації 0120U103580, 2021 р.), “Розроблення моделей та методів високопродуктивних обчислень та їх застосування”. (номер державної реєстрації 0122U200449, 2022 р.).

Мета і завдання дослідження.

Метою дисертаційної роботи є розробка на базі дробово-диференціального підходу нових неklasичних математичних моделей геоміграційних процесів за складних умов їх перебігу; постановка та одержання розв'язків відповідних цим моделям крайових задач; розробка нових ефективних алгоритмів комп'ютерного моделювання і прогнозування динаміки процесів та ідентифікації параметрів моделей для високопродуктивних обчислювальних систем зі спільною та розподіленою пам'яттями.

Для досягнення мети вирішуються наступні задачі:

- розробка нових математичних моделей геоміграційних процесів за складних геогідрологічних умов на основі апарату диференціювання дробового порядку та постановка нових крайових задач;
- розробка класу паралельних алгоритмів моделювання геоміграційної динаміки на основі дробово-диференціальних моделей для систем зі спільною та розподіленою пам'яттю;
- розробка метаевристичних алгоритмів розв'язання задач ідентифікації параметрів дробово-диференціальних моделей геоміграційних процесів;
- застосування розробленого алгоритмічного апарату для розв'язання задач моделювання динаміки процесів масо-, тепло- та вологоперенесення у ґрунтах за складних умов їх перебігу, встановлення закономірностей зміни значень параметрів моделей при зміні характеру описуваних процесів.
- застосування розробленого алгоритмічного апарату для розв'язання прикладних задач, пов'язаних зі зрошувальним землеробством.

Об'єкт дослідження.

Об'єктом дослідження є неklasичні математичні моделі аномальної динаміки геоміграційних процесів та алгоритмічний апарат комп'ютерного моделювання геоміграційної динаміки на основі запропонованих дробово-диференціальних математичних моделей.

Предмет дослідження.

Предметом дослідження є методи та засоби математичного та комп'ютерного моделювання геоміграційних процесів.

Методи дослідження.

Для побудови і дослідження неklasичних математичних моделей в рамках тематики дисертаційної роботи використано методи математичної фізики, теорії фільтрації, масо-, тепло- та вологоперенесення, механіки ґрунтів, теорії інтегро-диференціювання дробового порядку, теорії чисельних методів. Для чисельного розв'язання початково-крайових задач використаний метод скінченних різниць, зокрема, метод розщеплення для багатовимірних задач. Для розв'язання задач

ідентифікації параметрів моделей використано метаевристичні методи розв'язання обернених задач, а саме: метод рою частинок та метод генетичного програмування. З метою побудови алгоритмів для високопродуктивних обчислювальних систем використано методи розпаралелення обчислень для систем зі спільною та розподіленою пам'яттями.

Наукова новизна отриманих результатів.

У дисертаційній роботі отримано такі нові результати:

- побудовано серію двовимірних дробово-диференціальних математичних моделей нелокальних процесів конвективної дифузії за умов планової фільтрації в ізотермічній та неізотермічній постановках, які, на відміну від існуючих, враховують фактори часової та просторової нелокальності при перебігу відповідних процесів у середовищах фрактальної структури;
- розроблено серію дробово-диференціальних моделей нелокальних процесів фільтраційної консолідаційної динаміки у разі аномального перебігу відповідних деформаційних процесів у геопористих середовищах;
- запропоновано та сформульовано серію дробово-диференціальних математичних моделей нелокальних процесів масоперенесення в ґрунтових середовищах з урахуванням явищ масообміну та факторів часової і просторової нелокальності;
- сформульовано крайові задачі щодо вказаних вище класів нових математичних моделей та одержано їх скінченнорізницеві розв'язки, на основі яких досліджено особливості динаміки процесів, що описуються відповідними моделями;
- побудовано клас паралельних алгоритмів розв'язання з використанням графічних прискорювачів одновимірних початково-крайових задач щодо дробово-диференціальних моделей геогідроміграційних процесів, які дозволяють збільшити ефективність комп'ютерного моделювання;
- створено клас ефективних паралельних алгоритмів реалізації локально-одновимірних скінченно-різницевих схем розв'язання багатовимірних дробово-диференціальних початково-крайових задач, які дозволяють збільшити швидкодію моделювання аномальних геогідроміграційних процесів;

- для розв'язання задач ідентифікації параметрів дробово-диференціальних моделей вологоперенесення, що містять ψ -похідні Капуто вперше застосовано методи рою частинок та генетичного програмування і показано їх ефективність.

Практичне значення отриманих результатів.

Практичне значення отриманих результатів полягає в тому, що

- створені засоби математичного та комп'ютерного моделювання забезпечують високу швидкодію та контрольовану точність прогнозування процесів волого- та солеперенесення у ґрунтах, необхідну для ефективного їх застосування у цивільній інженерії та сільському господарстві;

- досліджені особливості опису процесів геоміграції моделями з різними дробово-диференціальними операторами дозволяють проводити вибір моделі відповідно до конкретного застосування, а також можуть надавати допоміжну інформацію при фізичних та гідрологічних дослідженнях відповідних середовищ;

- у роботі вирішені прикладні задачі моделювання вологоперенесення при зрошенні дощуванням у складних гідрологічних умовах, а саме прогнозування строків та норм поливу за різних умов на основі дробово-диференціальних моделей вологоперенесення. Відповідні результати були апробовані у виробничих умовах.

Особистий внесок здобувача.

Усі результати, що складають основний зміст дисертаційної роботи та виносяться на захист, отримані автором самостійно. У роботах [59, 64, 67, 65], написаних у співавторстві з В. М. Булавацьким, здобувачеві належать участь у розробці методології побудови нових математичних моделей та побудові обчислювальних алгоритмів для знаходження чисельних розв'язків задач, створення відповідного математичного забезпечення, проведення числових розрахунків та аналіз отриманих результатів. У роботах [50, 51, 52], написаних у співавторстві з В. М. Булавацьким, здобувачеві належать участь у побудові математичних моделей та алгоритмів розв'язування задач, програмна реалізація алгоритмів, проведення числових розрахунків, обґрунтування комп'ютерної збіжності за даними розрахунків, проведення аналізу та систематизація отриманих результатів. У

роботі [38], написаній у співавторстві з В. М. Булавацьким, та у роботах [34, 54], написаній у співавторстві з В. М. Булавацьким та Ю. Г. Кривоносом, здобувачеві належать побудова та програмна реалізація алгоритмів розв'язування крайових задач, проведення числових розрахунків та аналізу і обґрунтування отриманих результатів. У роботах [53, 276, 277, 278] внесок усіх співавторів рівноцінний. У роботі [36], написаній у співавторстві з А. В. Гладким, М. І. Ромащенко, Т. В. Матяш та у роботах [35, 37], написаних у співавторстві з А. В. Гладким, здобувачеві належать математична постановка задач, розробка та програмна реалізація алгоритмів, проведення обчислювальних експериментів та аналіз їх результатів. У роботі [269] здобувачеві належать матеріал підрозділів 4.1–4.4 та участь у написанні підрозділів 3.6, 3.7.

Роботи [39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 55] є одноосібними публікаціями здобувача.

Апробація матеріалів дисертації.

Результати дисертації доповідались та обговорювались на конференціях "Інформаційні технології та комп'ютерне моделювання" м. Івано-Франківськ, 14–19 травня 2018 року; Fifth International Conference "High Performance Computing" HPC-UA 2018, м. Київ, 22–23 жовтня 2018; "Інформаційні технології та комп'ютерне моделювання", м. Івано-Франківськ, 20–25 травня 2019 року; The Third International Conference on Computer Science, Engineering and Education Applications (ICCSEEA2020), м. Київ, 21–22 січня 2020 року; 9th International Conference "Information Control Systems & Technologies", м. Одеса, 24–26 вересня 2020 року; Modeling, control & information technologies, м. Рівне, 5–7 листопада 2020 року; науковому семінарі "Методи обчислювальної математики", Інститут кібернетики імені В.М. Глушкова НАН України, м. Київ, 21 червня 2023 року; "Науковому семінарі з математичного моделювання та обчислювальних методів" НУВГП, м.Рівне, 26 квітня 2024 року.

Публікації.

Результати досліджень опубліковані загалом у 29 наукових роботах. Основні результати опубліковані у 1 монографії [269] та 22 статтях у фахових наукових

журналах, що з урахуванням класифікації SCImago Journal & Country Rank прирівнюється до 50 публікацій. Зокрема, 2 статті опубліковано у виданнях, що належать до квартиля Q1 [36, 46]; 9 статей у виданнях, що належать до квартиля Q2 [40, 44, 45, 47, 51, 52, 59, 64, 67]; 5 статей у виданнях, що належать до квартиля Q3 [34, 37, 39, 54, 65]; 2 статті, що індексуються у наукометричній базі Scopus, проте не класифіковані до певного квартиля [48, 50]; 4 статті у фахових виданнях України, що не індексуються у наукометричній базі Scopus [38, 276, 277, 278]. Результати також опубліковані у 6 роботах у матеріалах наукових конференцій [35, 41, 42, 43, 53, 55], з яких 2 індексуються у наукометричній базі Scopus [41, 55].

Структура та обсяг дисертації.

Дисертація складається зі вступу, восьми розділів, висновків, списку використаних джерел (321 найменування), додатків на 4 сторінках. Загальний обсяг дисертації становить 397 сторінок, основну частину викладено на 340 сторінках.

Висловлюю щирю вдячність д.т.н., проф. Володимиру Михайловичу Булавацькому за розширення кола моїх наукових інтересів та допомогу у написанні цієї роботи.

Розділ 1. Аналіз сучасного стану проблеми математичного і комп'ютерного моделювання гідрогеоміграційних процесів з неklasичною динамікою

Розробка методів математичного моделювання динаміки міграційних процесів у ґрунтових середовищах [271, 272, 273, 274, 279, 282, 283, 284, 288, 297, 299, 304, 306, 307, 308, 316] є актуальною у різних сферах, зокрема охорони навколишнього середовища, ведення ефективної сільськогосподарської діяльності та раціонального використання природних ресурсів. Відповідно, актуальною є і проблема підвищення ступеня адекватності наявних кількісних моделей таких процесів при їх перебігу у середовищах з неklasичною динамікою для яких характерні ефекти пам'яті та просторової нелокальності. У цих умовах використання класичних математичних моделей не завжди є коректним і прогрес досягається, зокрема, з використанням підходу, що базується на застосуванні апарату дробово-диференціального числення [97, 115, 139, 177, 314, 318].

Відповідні моделі розробляються, зокрема, розглядаючи процес броунівського руху за різних вихідних припущень та використовуючи апарат статистичної фізики для виведення дробово-диференціальних форм базових фізичних законів [116, 165, 227]. У випадку масо-, соле-, теплоперенесення, консолідаційних процесів у ґрунтах принциповим є перебіг великої кількості взаємопов'язаних фізичних, хімічних та біологічних процесів у них. У складних геолого-гідрологічних умовах, навіть формальне узагальнення моделей заміною операторів цілого порядку на оператори нецілого порядку дозволяє отримувати напівемпіричні моделі, застосування яких дозволяє збільшити точність прогнозування спостережуваних процесів.

Таким чином, саме дробово-диференціальні моделі є класом моделей гідрогеоміграційних процесів з неklasичною динамікою, якій дозволяє збільшити точність їх опису як у ситуації наявності необхідної кількості вихідних даних, так і у випадку їх відсутності. Практичне застосування дробово-диференціальних моделей для прогнозування міграційних процесів у ґрунтах (геоміграційних процесів) у більшості випадків вимагає чисельного підходу через нелінійність початково-крайових задач та змінність їх коефіцієнтів. При цьому, інтегро-диференціальний

характер операторів, у порівнянні з диференціальними операторами у класичних моделях, призводить до підвищення порядку обчислювальної складності. Необхідність розв'язувати обернені задачі ідентифікації параметрів, що не можуть бути безпосередньо вимірними, зокрема, порядків дробових похідних, у таких умовах актуалізує використання метаевристичних підходів додатково збільшуючи час, необхідний для отримання прогнозів розвитку процесів.

Таким чином, при побудові процедур математичного моделювання геоміграційних процесів, необхідно визначитись з типом використовуваних операторів, класом моделі, методикою чисельного розв'язання прямих задач та задач ідентифікації параметрів, необхідністю застосування та типом додаткових механізмів, які б прискорювали обчислення при комп'ютерній реалізації. Наступні пункти розділу містять огляд та аналіз цих аспектів.

1.1. Короткий огляд основних означень апарату інтегро-диференціювання дробового порядку

Історично одними з перших були запропоновані визначення інтегралу Рімана–Ліувіля [149], які щодо функції $f(x)$ на проміжку $[a, b]$ мають наступний вигляд:

$${}^{RL}I_{a+}^{\alpha} f(x) = \frac{1}{\Gamma(\alpha)} \int_a^x f(\xi)(x - \xi)^{\alpha-1} d\xi,$$

$${}^{RL}I_{a+}^0 f(x) := f(x), x > a$$

та

$${}^{RL}I_{b-}^{\alpha} f(x) = \frac{1}{\Gamma(\alpha)} \int_x^b f(\xi)(x - \xi)^{\alpha-1} d\xi,$$

$${}^{RL}I_{b-}^0 f(x) := f(x), x < b,$$

де $\Gamma(z)$ – гамма-функція Ейлера.

Дробова похідна Рімана–Ліувіля порядку α визначається наступним чином [149]:

$${}^{RL}D_{a+}^{\alpha} f(x) = \frac{1}{\Gamma(n - \alpha)} \frac{d^n}{dx^n} \int_a^x f(\xi)(x - \xi)^{n-\alpha-1} d\xi = \frac{d^n}{dx^n} {}^{RL}I_{a+}^{n-\alpha} f(x),$$

$${}^{RL}D_b^\alpha f(x) = \frac{(-1)^n}{\Gamma(n-\alpha)} \frac{d^n}{dx^n} \int_x^b f(\xi)(\xi-x)^{n-\alpha-1} d\xi, \quad n-1 < \alpha \leq n \in \mathbb{N}.$$

Відмітимо можливість визначення як ліво- та правосторонніх, так і двосторонніх на відрізку дробових похідних в сенсі Рімана–Ліувіля у формі похідної Ріца [194]:

$${}^R D_b^\alpha f(x) = -c_v \left({}^{RL}D_x^\alpha + {}^{RL}D_b^\alpha \right) f(x).$$

Надалі обмежимося розглядом лише лівосторонніх похідних при $a = 0$.

Особливістю дробової похідної Рімана–Ліувіля, що проявляється при розв'язанні задач для рівнянь, які її містять, є необхідність задання дробово-диференціальних початкових умов. Інший підхід до визначення дробової похідної, запропонований Ф. Капуто [74] та А. Герасимовим [286], є вільним від цього недоліку. Дробова похідна Капуто–Герасимова визначається наступним чином:

$$D_x^\alpha f(x) = \frac{1}{\Gamma(n-\alpha)} \int_a^x \frac{d^n}{dx^n} f(\xi)(x-\xi)^{n-\alpha-1} d\xi, \quad n = \lfloor \alpha \rfloor + 1, x > a.$$

Альтернативним визначенням є похідна Грюнвальда–Летнікова [117, 301], що при $f \in C^{\lfloor \alpha \rfloor}$ [93] збігається з похідною Рімана–Ліувіля:

$${}^{GL}D_x^\alpha f(x) = \lim_{h \rightarrow 0} \frac{1}{h^\alpha} \sum_{k=0}^{\lfloor n \rfloor} (-1)^k \frac{\Gamma(\alpha+1) f(x-kh)}{\Gamma(k+1)\Gamma(\alpha-k+1)}, \quad nh = x.$$

Завдяки визначенню через ліміт, похідна Грюнвальда–Летнікова знайшла широке застосування при чисельних розрахунках.

Наведені вище означення похідних Капуто та Рімана–Ліувіля є класичними визначеннями дробових похідних у вигляді інтегро-диференціальних операторів з сингулярним ядром. Подальший розвиток апарату дробових похідних відбувається у різних напрямках. Окреслимо два з них.

Першим є напрямок узагальнення операторів. В його межах виділимо дробові похідні Хільфера [122]

$$D_x^{\alpha,\mu} f(x) = {}^{RL}I_x^{\mu(1-\alpha)} \left(\frac{d}{dx} \right) {}^{RL}I_x^{(1-\mu)(1-\alpha)} f(x), \quad 0 \leq \mu \leq 1, 0 < \alpha < 1$$

та ψ -похідну Капуто [11] від функції f за функцією g , $g(t) \in C^1[0, +\infty)$, $g'(t) > 0 (t \geq 0)$, $g(0) = 0$, яка при $0 < \alpha \leq 1$ має вигляд

$$D_{t,g}^{(\alpha)} f(x) = \frac{1}{\Gamma(1-\alpha)} \int_0^x \frac{d}{dx} f(\xi) [g(t) - g(\xi)]^{-\alpha} d\xi.$$

При $g = t^\beta$ ψ -похідна Капуто набуває форму похідної Капуто–Катугампола [13, 135]:

$$D_{t,\beta}^{(\alpha)} f(x) = \frac{1}{\Gamma(1-\alpha)} \int_0^x \frac{d}{dx} f(\xi) [t^\beta - \xi^\beta]^{-\alpha} d\xi.$$

Іншим підходом є перехід від сингулярних до несингулярних ядер у операторах дробових похідних. Серед таких похідних виділимо похідну Капуто–Фабріціо [75] з ядром, що містить експоненційну функцію, вигляду

$${}^{CF} D_x^{\alpha,\lambda} f(x) = \frac{1}{\lambda(1-\alpha)} \int_0^x \frac{d}{dx} f(\xi) e^{\frac{\alpha}{1-\alpha} \frac{t-\xi}{\lambda}} d\xi, 0 < \alpha \leq 1$$

та похідну Атангана–Балеану [18], ядро якої містить функцію Міттаг-Лефлера [113], та яка має вигляд

$${}^{ABC} D_x^\alpha f(x) = \frac{1}{1-\alpha} \int_0^x \frac{d}{dx} f(\xi) E_\alpha \left(-\frac{\alpha}{1-\alpha} (x-\xi)^\alpha \right) d\xi, 0 < \alpha \leq 1.$$

Зауважимо, що похідні з несингулярним ядром критикуються з позиції можливості їх застосування у фізиці [190] у зв'язку з обмеженнями, які вони накладають [206], зокрема у формі необхідності задання нульових початкових умов.

Вищенаведені оператори є інтегро-диференціальними операторами з потенційно нескінченим інтервалом інтегрування. Враховуючи деякі зауваги щодо їх нефізичності [188], наразі розвивається напрямок дробових похідних з обмеженим інтервалом інтегрування, зокрема [255] у загальній формі

$$D_{l,x}^\alpha f(x) = \int_{x-l(x,\alpha)}^x \frac{d^n}{dx^n} f(\xi) K_\alpha(x-\xi) d\xi,$$

де $K_\alpha(x-\xi)$ – задане ядро інтегрального оператора.

Відзначимо, що окрім вищеописаних операторів і визначень похідних нецілого порядку, наразі існує також велика кількість інших їх визначень (див.,

наприклад, [234]), зокрема, активно розвивається апарат дробового диференціювання змінного [209] та розподіленого порядків [81].

1.2. Дробово-диференціальні моделі процесів гідрогеографії

Математичне моделювання динаміки міграції вологи і солей в ґрунтах є широко поширеним і розвивається, переважно, в рамках постановок задач характерних для класичної математичної фізики. Серед достатньо великої кількості публікацій з цього питання виділимо роботи [80, 88, 132, 221, 266, 267, 270, 271, 272, 273, 274, 279, 280, 281, 282, 283, 284, 285, 287, 288, 289, 292, 297, 298, 299, 300, 304, 306, 307, 308, 310, 311, 312, 316, 319, 320]. При перебігу процесів перенесення в складних нерівноважних умовах, зокрема в середовищах фрактальної структури до яких згідно з [320] відносяться і ґрунти, вони ефективно описуються за допомогою математичного апарату диференціювання дробового порядку [115, 139, 177, 318].

Узагальнення диференціальних моделей на дробово-диференціальний випадок проводиться наступним чином.

Перший з можливих підходів полягає у безпосередній заміні похідних цілого порядку на похідні дробового порядку, або одних похідних дробового порядку на інші. Отримувані при цьому моделі є напівемпіричними та активно використовуються у моделях складних систем, зокрема, в економіці [183], медицині [138], сільському господарстві [203]. Узагальнення шляхом заміни похідних цілого порядку на похідні дробового порядку використовується також і у теорії ігор [83, 84, 258].

Застосування такого підходу може бути проілюстровано наступною серією моделей розповсюдження коронавірусної хвороби. Так, у [138] модель розповсюдження хвороби через ринок морепродуктів у формі системи звичайних диференціальних рівнянь узагальнюється заміною похідних першого порядку похідними Атангана–Балеану. У [24] використовується похідна Капуто–Фабріціо, а у [217] – класична похідна Капуто–Герасимова. В усіх випадках, автори проводять математичне узагальнення моделей без наведення наявності підстав для такого узагальнення з позиції прикладної області.

З іншого боку, для таких процесів, як дифузія речовини, для яких існують глибоко розроблені фізичні теорії, дробово-диференціальне узагальнення моделей відбувається шляхом зміни вихідних припущень у фізичних законах, з яких вони виводяться. Зокрема, для процесів масоперенесення, використання апарату дробового диференціювання пов'язується з поняттям фрактальності середовища [174], а тип оператора – з типом явища [215]. Згідно з [215] похідні Капуто та Рімана–Ліувіля описують ефекти пам'яті та часової нелокальності, тоді як похідна Капуто–Фабріціо – явище розподіленої затримки, яке є локальним за своєю природою.

Коротко розглянемо припущення та процедури виводу дробово-диференціальних форм деяких фізичних законів.

Згідно з [227] використання традиційного оператора дивергенції для опису закону збереження маси обумовлено тим, що, зазвичай, масштаб вимірювань у середовищі суттєво більший ніж масштаб неоднорідностей у ньому. У випадку, коли розглядається потік рідини в пористому середовищі, його можна вважати континуумом лише до рівня масштабу, що в декілька разів перевищує розмір пор. Якщо для гранульованого однорідного пористого середовища інтегральний масштаб складає близько 1 см^3 , що, у багатьох випадках, менше ніж мінімальний масштаб при вимірюваннях [87], то у випадку відсутності властивості гранульованості він цілком може мати порядок $1\text{--}10 \text{ м}^3$. Для фрактальних, чи масштабно-інваріантних пористих середовищ, інтегральний масштаб є таким, що може вважатись нескінченним [162].

Фундаментальним припущенням тут є те, що контрольний об'єм прямує до нуля при прямуванні до нуля його лінійних розмірів. В такому випадку зміни у властивостях середовища малі та лінійні в межах контрольного об'єму, що дозволяє апроксимувати функцію потоку речовини лише членом першого порядку у ряді Тейлора. Оскільки контрольні об'єми для гетерогенних пористих середовищ великі, а властивості в їх межах суттєво варіюються, використання такого підходу не завжди є прийнятним.

Розкладаючи функцію потоку речовини $F(x_1)$ через грань x_1 прямокутного контрольного об'єму з довжинами сторін $\Delta x_1, \Delta x_2, \Delta x_3$ у ряд Тейлора в околі точки x_1 отримуємо

$$F(x_1 + \Delta x_1) = \Delta x_2 \Delta x_3 \left(\rho q_1 + \frac{\partial}{\partial x_1}(\rho q_1) \Delta x_1 + \frac{\partial^2}{\partial x_1^2}(\rho q_1) \frac{\Delta x_1^2}{2} + \dots \right), \quad (1.1)$$

де ρ – щільність речовини, а q_1 – питома витрата через грань x_1 .

Обмежуючи ряд у (1.1) двома першими членами, отримуємо

$$F(x_1 + \Delta x_1) = \Delta x_2 \Delta x_3 \left(\rho q_1 + \frac{\partial}{\partial x_1}(\rho q_1) \Delta x_1 \right).$$

Класична диференціальна форма закону збереження маси отримується розглядаючи потік $F(x_1) - F(x_1 + \Delta x_1)$ при $\Delta x_1 \rightarrow 0$.

Очікуючи, що потік є нелінійним та розглядаючи ряд Тейлора повністю отримуємо наступну, більш загальну, форму:

$$\frac{1}{\Delta V} \frac{\partial}{\partial t}(\Delta V \sigma \rho) = - \sum_{n=1}^{\infty} \left(\frac{(\Delta x_i)^{n-1}}{n!} \frac{\partial^n}{\partial x_i^n}(\rho q_i) \right), \quad (1.2)$$

де $\Delta V = \Delta x_1 \Delta x_2 \Delta x_3$, а σ – пористість середовища.

Недоліком рівняння (1.2) є важкість його застосування при проведенні моделювання процесів масоперенесення. Узагальнити класичну форму закону збереження маси зберігши можливість практичного використання дозволяє апарат дробово-диференціальних рядів Тейлора [166, 171], що є узагальненням поняття ряду Тейлора на випадок дробових похідних та визначається як

$$F(y) = F(x) + D_x^\alpha F(x) \frac{(y-x)^\alpha}{\Gamma(\alpha+1)} + D_x^\alpha D_x^\alpha F(x) \frac{(y-x)^{2\alpha}}{\Gamma(2\alpha+1)} + \dots,$$

де $D_x^\alpha f(t)$ – дробова похідна Капуто [74] порядку α .

Дробово-диференціальна за просторовою змінною форма закону збереження маси отримується розкладаючи функцію потоку у дробово-диференціальний ряд Тейлора обмежуючись першими двома його членами:

$$F(x_1 + \Delta x_1) = \Delta x_2 \Delta x_3 \left(\rho q_1 + D_{x_1}^\alpha (\rho q_1) \frac{\Delta x_1^\alpha}{\Gamma(\alpha + 1)} \right).$$

Аналогічним чином може бути отримана дробово-диференціальна модифікація закону Дарсі [165]. Класична форма закону Дарсі може бути виведена методами просторового усереднення з закону в'язкості Ньютона, який лінійно пов'язує в'язкість та градієнт швидкості деформації:

$$\tau = -\mu \frac{\partial v}{\partial x}, \quad (1.3)$$

де τ – дотичний компонент тензору напруги, μ – коефіцієнт внутрішнього тертя, v – швидкість.

Рівняння (1.3) обґрунтовується наступним чином [165]. Розглянемо два паралельних шари рідини, розташовані на рівнях x і $x + \Delta x$, що рухаються з швидкостями $v(x)$ та $v(x + \Delta x)$, відповідно. Сила тертя між пластинами відносно рівня x пропорційна зміні імпульсів, тобто

$$F = K_v (\rho A \Delta x) (v(x) - v(x + \Delta x)), \quad (1.4)$$

де A – площа контакту, ρ – щільність, а K_v – коефіцієнт в'язкого тертя.

Лінійно апроксимуючи

$$v(x + \Delta x) = v(x) + \frac{\partial v(x)}{\partial x} \Delta x + O(\Delta x) \quad (1.5)$$

з (1.4) отримуємо

$$F = -K_v (\rho A \Delta x) \left(\frac{\partial v(x)}{\partial x} \Delta x + O(\Delta x) \right),$$

з чого при $\Delta x \rightarrow 0$ одержується (1.3).

Проте, у гетерогенному пористому середовищі можуть виникати суттєві коливання гідравлічної провідності, що призводить до великих змін швидкостей руху рідини в масштабі пор. При цьому, в'язкі ефекти, що виникають протягом певного часу Δt , не обов'язково обмежуються просторовим масштабом Δx і для опису таких процесів локальний закон Ньютона (1.3) має бути узагальнений для врахування нелокальних ефектів.

Таке узагальнення отримується замінюючи апроксимацію (1.5), отриману з класичного ряду Тейлора, аналогічною апроксимацією, отриманою з його дробово-диференціального аналога [165, 166]. Тоді, при $\Delta x \rightarrow 0$ одержується дробово-диференціальний закон в'язкості Ньютона у формі

$$\tau = -\mu D_x^q v(x).$$

Надалі, дробово-диференціальна за просторовими змінними форма закону Дарсі отримується застосовуючи оператор усереднення до рівняння Стокса, що описує потік у мікроскопічному масштабі на основі дробово-диференціального закону Ньютона, у формі

$$\begin{aligned} -\nabla \rho + \rho g + \mu D_x^{q+1} v &= 0, \\ \nabla \cdot v &= 0, \end{aligned}$$

нехтуючи при цьому змінами щільності та в'язкості в межах області усереднення.

На основі подібного підходу з використанням апарату статистичної фізики отримується також і дробово-диференціальне узагальнення закону Фіка [100, 114, 116, 176] у формі [176]:

$$F(x,t) = P_\theta^\alpha \left[-\frac{\partial}{\partial z} C(x,t) \right],$$

де $F(x,t)$ – потік речовини, $C(x,t)$ – її концентрація, P_θ^α – псевдодиференціальний оператор з символом

$$P_\theta^\alpha := \frac{D_\theta^\alpha(\kappa)}{D^2(\kappa)} = |\kappa|^{\alpha-2} e^{i(\text{sign}\kappa)\theta\pi/2}$$

Зауважимо, що вищевикладені підходи до узагальнення закону збереження маси, законів Дарсі та Фіка не є єдино можливими (див., наприклад, оглядову роботу [120]). Серед них виділимо підхід “багатошвидкісного масоперенесення” (multi-rate mass transfer model, MRMT) [76, 105], застосуванням якого отримана, зокрема, модель виду

$$\phi \frac{\partial C}{\partial t} + \nabla \cdot (v_{\text{eff}} C - D_{\text{eff}} \nabla C) = -\frac{\partial}{\partial t} \int_0^t \phi(t-\tau) C(x,\tau) d\tau,$$

де ϕ – загальна пористість, v_{eff} – усереднена по простору швидкість перенесення, а D_{eff} відображає розповсюдження шлейфу речовини в результаті коливань швидкості. Ця модель містить інтегро-диференціальний оператор з довільною функцією пам'яті φ , форма якої залежить від структури середовища.

Відштовхуючись від узагальнень базових фізичних законів наразі побудована та досліджена велика кількість міграційних моделей (див., наприклад, [164]) у пористих середовищах різного типу, зокрема, моделі солеперенесення [4, 26, 228], вологоперенесення [121, 172, 181], теплоперенесення [144, 245], конвективної дифузії [59, 62, 63]. Більшість подібних моделей містить похідні Капуто–Герасимова за часовою та/або просторовими змінними.

Поєднання підходів узагальнення фізичних законів та безпосередньої заміни одних похідних іншими породжує ще ширший клас моделей. Наразі це в першу чергу моделі, що містять похідні Капуто–Фабріціо чи Атангана–Балеану [2, 3, 5, 202]. Іншим напрямком, який знаходиться на початковому етапі свого розвитку, є використання похідних від функції за іншою функцією. Щодо моделей міграційних процесів, виділимо тут роботу [193].

Зауважимо також, що перехід від моделей цілого порядку до дробово-диференціальних моделей змінює фізичну розмірність величин, які до них входять. Для приведення цих розмірностей у відповідність до розмірностей вимірюваних величин, до моделей вводяться параметри часової та просторової розмірності (див., наприклад, [107]). Оскільки вибір, а особливо вимірювання, значень таких параметрів є фізично складною задачею, у багатьох випадках вони приймаються рівними 1.

Враховуючи велику кількість можливих визначень дробових похідних та комплексність процесів у таких пористих середовищах, як ґрунти, розвиток апарату їх дробово-диференціального моделювання далекий від завершення. В зв'язку з великою кількістю можливих моделей, актуальним є задача експериментального визначення найбільш адекватної моделі для опису фактично вимірюваних даних (стосовно ширшого обговорення цього питання див., наприклад, [12]). Актуальності феноменологічному напрямку розвитку апарату дробово-диференціальних моделей

додають також зауваги щодо їх фізичної коректності, зокрема в контексті задання початкових умов [189].

1.3. Деякі особливості чисельного розв'язання початково-крайових задач для дробово-диференціальних рівнянь

Розв'язання задач щодо рівнянь, які містять дробові похідні, стикається з додатковими складнощами, порівнюючи з задачами щодо рівнянь з похідними цілого порядку, через їх інтегро-диференціальний характер.

Враховуючи прикладний напрямок роботи, далі будемо розглядати виключно чисельні методи розв'язання початково-крайових задач щодо дробово-диференціальних рівнянь.

Найрозповсюдженішим методом чисельного розв'язання початково-крайових задач щодо дробово-диференціальних рівнянь є метод скінченних різниць. Заміняючи похідну в інтегральному операторі її різницевиими аналогами та дискретизуючи його самого кубатурними формулами, отримується серія скінченно-різницевих апроксимацій дробових похідних (зупинимось лише на випадку класичної похідної Капуто) різного порядку точності.

Так, базова L1-апроксимація для $0 < \alpha < 1$ порядку точності $O(h^{2-\alpha})$, $t_j = t_0 + jh$ [94, 168] має вигляд

$$D_{t_0}^\alpha f(t_n) \approx \Delta_{t_0}^\alpha f(t_n) = \frac{1}{\Gamma(2-\alpha)} \sum_{j=0}^{n-1} w_{\mu,n} (f(t_{n-j}) - f(t_{n-j-1})),$$

$$w_{\mu,n} = \frac{(t_n - t_\mu)^{1-\alpha} - (t_n - t_{\mu+1})^{1-\alpha}}{t_{n-\mu} - t_{n-\mu-1}}.$$

Для $1 < \alpha < 2$ маємо L2-схему [94, 168] порядку точності $O(h^{3-\alpha})$ виду

$$D_{t_0}^\alpha f(t_n) \approx \Delta_{t_0,L2}^\alpha f(t_n) = \frac{h^{-\alpha}}{\Gamma(3-\alpha)} \sum_{k=1}^n w_{k,n} f(t_{n-k}),$$

$$w_{k,n} = \begin{cases} 1, & k = -1, \\ 2^{2-\alpha} - 3, & k = 2, \\ (k+2)^{2-\alpha} - 3(k+1)^{2-\alpha} + 3k^{2-\alpha} - (k-1)^{2-\alpha}, & 1 \leq k \leq n-2, \\ -2n^{2-\alpha} + 3(n-1)^{2-\alpha} - (n-2)^{2-\alpha}, & k = n-1, \\ n^{2-\alpha} - (n-1)^{2-\alpha}, & k = n. \end{cases}$$

У [94] доводиться, що використання схем збільшення порядку точності скінченно-різницевих апроксимацій, що базуються на збільшенні порядку поліномів, на відміну від випадку похідних цілого порядку, не збільшують порядок точності апроксимації дробових похідних з сингулярними ядрами, окрім ситуацій накладання додаткових обмежень на розв'язки рівнянь. Не зважаючи на таке обмеження, відома велика кількість скінченно-різницевих апроксимацій, розглядуваних як апроксимації підвищеного порядку точності, для дробових похідних з різними ядрами, зокрема, апроксимація дробової похідної від функції за іншою функцією, що використовує три члени ряду Тейлора для похідної функції [232], чи апроксимація четвертого порядку точності, що базується на визначенні дробової похідної Грюнвальда–Летнікова [240], та інші [146, 233]. Огляд апроксимацій високого порядку точності дається, зокрема, у роботі [90].

У випадку рівнянь, що містять дробові похідні лише за часовою змінною [118, 119, 212, 213], використання такої процедури дискретизації впливає лише на праву, незалежну від розв'язку у поточний момент часу, частину рівнянь, на які дискретизується розв'язувана задача. Вигляд обчислювальної процедури тут залежить лише від методу дискретизації задачі за просторовими змінними, проте необхідність чисельного інтегрування при обчисленні значень правих частин рівнянь у більшості випадків впливає на загальну обчислювальну складність процедури розв'язання.

При скінченно-різницевій дискретизації рівнянь, дробово-диференціальних за просторовими змінними [95, 128, 230, 251], змінюється тип матриці відповідних систем лінійних алгебраїчних рівнянь (СЛАР). Зі стрічкових вони стають повністю заповненими, що змінює тип алгоритмів їх розв'язування суттєво підвищуючи обчислювальну складність. Два найбільш розповсюджених підходи до зниження

складності тут є обчислення значень дробових похідних за просторовими змінними використовуючи значення функції на попередньому кроці за часом [154, 200] та використання того факту, що для деяких дробових похідних, зокрема класичної похідної Капуто, при рівномірній сітці матриці СЛАР можуть представлятись через матриці Теплиця [128, 157, 223]. Серед інших підходів виділимо розробку спеціальних передобумовлювачів [27, 98, 254].

У багатовимірних випадках, зниження обчислювальної складності стає ще більше важливим. Одним з найбільш використовуваних підходів, які для цього застосовуються при розв'язанні задач методом скінченних різниць, є схеми розщеплення, зокрема метод поперемінних напрямків (Alternate Directions Implicit, ADI) [148, 239, 247]. Ці методи зводять розв'язання багатовимірних задач до серій одновимірних. Окрім прямого ефекту зниження складності, суттєве зменшення зв'язаності блоків обчислень дозволяє у цьому випадку ефективніше застосовувати методи розпаралелення (див., наприклад, [229]).

Іншим підходом до розв'язання задач щодо дробово-диференціальних рівнянь є використання спектральних методів [29]. Серед них виділимо методи, що базуються на апроксимації шуканої функції поліномами, або вейвлетами на базі поліномів Лежандра [1, 20], Чебишева [145, 159], Якобі [28, 30], апроксимації В-сплайнами [182], вейвлетами Хаара [7] та ін. Один з напрямків розвитку спектральних методів розв'язання дробово-диференціальних рівнянь є використання методів Галеркина [238, 246], зокрема, методу скінченних елементів [130, 253, 257].

Зауважимо, що складність застосування спектральних методів до дробово-диференціальних рівнянь у порівнянні з методом скінченних різниць полягає у необхідності обчислення відповідних інтегралів від базисних функцій.

1.4. Підходи до ефективної дискретизації дробових похідних

Важливою проблемою, що виникає при чисельному розв'язанні початково-крайових задач для дробово-диференціальних рівнянь, особливо рівнянь з дробовими похідними за часовою змінною, є необхідність дискретизації інтегро-диференціальних операторів. Це призводить до збільшення порядку обчислювальної

складності, що суттєво ускладнює моделювання на сітках великого розміру чи/та великих часових інтервалах.

Зменшення обчислювальної складності досягається в першу чергу застосуванням спеціальних дискретизацій дробових похідних. На даний момент можна виділити два підходи до побудови таких дискретизацій.

Перший з них – це метод фіксованої пам'яті [102, 110, 178], який відкидає попередню інформацію, вклад якої вважається таким, що ним можна знехтувати. Базовий метод фіксованої пам'яті [178] апроксимує визначення дробової похідної (тут і надалі будемо розглядати похідну Капуто–Герасимова) обмежуючи інтегрування інтервалом фіксованої довжини T наступним чином:

$$D_x^\alpha f(x) \approx \frac{1}{\Gamma(n-\alpha)} \int_{x-T}^x \frac{d^n}{dx^n} f(\xi)(x-\xi)^{n-\alpha-1} d\xi. \quad (1.6)$$

Апроксимуючи (1.6) з першим порядком точності при $0 < \alpha < 1$ отримуємо

$$\Delta_{t_0}^\alpha f(t_n) \approx \frac{1}{\Gamma(2-\alpha)} \sum_{j=k}^{n-1} w_{\mu,n} \left(f(t_{n-j}) - f(t_{n-j-1}) \right), k = (x-T)/h. \quad (1.7)$$

Додаткове спрощення (принцип кускової пам'яті [110]) може бути одержано розбиваючи інтервал сумування у (1.7) на блоки з усередненням значень коефіцієнтів $w_{\mu,n}$ всередині них.

Серед застосувань таких підходів відмітимо роботи [223, 224].

Альтернативним до обмеження інтервалу інтегрування знизу є принцип логарифмічної пам'яті [102], згідно з яким чисельне інтегрування здійснюється на серії сіток з кроками, що збільшуються, які є вкладеними у початкову сітку. Таким чином, при зменшенні значення функції ядра інтегрального оператора відкидається більша кількість попередніх розв'язків. У якості прикладу застосування такого підходу до розв'язання рівняння з дробовою похідною за просторовою змінною відмітимо роботу [184].

Другий підхід до зменшення складності обчислення значень дробових похідних базується на апроксимації ядра інтегрального оператора, або альтернативному його представлені, але розгляді всіх попередньо обчислених розв'язків. Серед методів, отриманих в межах даного підходу виділимо схему

компресії ядра (kernel compression scheme) [22], згідно з яким ядро оператора дробової похідної Капуто апроксимується сумою експоненційних функцій, та метод Юаня–Агравала [241], який використовує представлення похідної Капуто–Герасимова у вигляді інтегралу на напівнескінченному проміжку, після апроксимації якого дробово-диференціальне рівняння перетворюється у систему звичайних диференціальних рівнянь. Метод Юаня–Агравала отримав розвиток, зокрема, у роботах [92, 152].

При використанні обох цих підходів, час розв'язання лишається близьким до постійного чи логарифмічно залежного від номера кроку за часом, що, проте, супроводжується зниженням точності отримуваних розв'язків.

1.5. Високопродуктивні алгоритми дробово-диференціального моделювання

Особливо важливими в контексті практичного застосування дробово-диференціальних моделей, зокрема моделей процесів геоміграції з неklasичною динамікою, є питання їх ефективної програмної реалізації. Необхідність дискретизації інтегральних операторів при чисельному розв'язанні відповідних крайових задач, з одного боку, збільшує обчислювальну складність проведення розрахунків, проте, з іншого – дозволяє ефективно застосовувати для їх пришвидшення засоби паралельних обчислень [111].

Найпростішим є застосування багатопоточного розпаралелення за даними, зокрема, реалізованого для базових операцій у відомих пакетах програм на кшталт Matlab. Як приклад такого застосування виділимо роботу [237].

Іншим підходом, що застосовується в першу чергу при розв'язанні задач з похідними дробового порядку лише за часовою змінною, є використання розпаралелення при безпосередньому обчисленні значень дробових похідних. Такий підхід використано зокрема у [21, 56, 91] при застосуванні до методу Адамса–Бешфорта–Молтона при його розпаралелюванні як на кластерних системах з використанням бібліотеки обміну повідомленнями MPI, так і на графічних процесорах (GPU) та у [31] для схеми предиктор-коректор при MPI- та багатопотоковому розпаралеленні.

Якщо моделі містять дробові похідні за просторовими змінними, на перше місце висувається швидкодія алгоритмів розв'язання СЛАР. Застосуванню паралельних алгоритмів такого класу присвячена певна кількість робіт, зокрема, у [108, 112, 150] пропонується виконувати розпаралелення операцій множення матриці на вектор та по-компонентних операцій між векторами (у [150] описується GPU-алгоритм), у [103] використовується паралельна модифікація передобумовленого ітераційного методу підпросторів Крилова, а у [225] – паралельний алгоритм розв'язання тридіагональних СЛАР.

Окремо виділимо питання розпаралелення скінченно-різницевих схем розщеплення. Не зважаючи на явний паралелізм у цих схемах, незначна кількість робіт присвячена його використанню для розв'язання задач щодо дробово-диференціальних рівнянь. Зокрема, можна виділити GPU-алгоритм, описаний у [101].

Відмітимо, що на даний час, кількість робіт, які стосуються паралельних алгоритмів для розв'язання початково-крайових задач щодо дробово-диференціальних рівнянь, в цілому незначна.

1.6. Методи ідентифікації параметрів дробово-диференціальних моделей

Практичне використання математичних моделей для прогнозування динаміки процесів вимагає проведення ідентифікації їх параметрів. Специфіка дробово-диференціальних моделей полягає у тому, що порядки та параметри дробових похідних не мають технічних засобів вимірювання, а часто лише розширюють кількість ступенів свободи відповідної моделі не маючи навіть чітко доведеного фізичного змісту. У такій ситуації, єдиним методом визначення значень параметрів дробових похідних є їх підбір таким чином, щоб модель якнайточніше описувала наявні вимірювання.

У випадку геоміграційних процесів, додатковою проблемою є розподіленість та низька точність інструментального визначення гідрофізичних параметрів ґрунтів, а також загальна складність взаємодіючих процесів у них. Підбір на основі вимірювань параметрів дробових похідних може у такій ситуації компенсувати як похибку самої моделі, так і похибки вимірювання.

Щодо гідрогеоміграційних моделей у класичній постановці, методи розв'язання обернених задач, в тому числі задач ідентифікації їх параметрів, широко досліджені, зокрема у [198, 305, 315]. Стосовно інтегральних моделей задач теплопровідності виділимо роботи [125, 293]. Загалом, виділимо два підходи, що використовуються для ідентифікації параметрів дробово-диференціальних моделей.

Перший полягає у аналітичному чи чисельно-аналітичному розв'язанні обернених задач щодо переважно одновимірних моделей. Зокрема, у [252] розв'язується задача відшукування невідомого коефіцієнту одновимірного дробово-диференціального за часовою змінною рівняння методом простої ітерації, а у [153] – задача ідентифікації функції джерела методом регуляризації Тихонова. Обернені задачі для біпараболічного рівняння в межах цього підходу розглядаються у [49, 66].

Певна кількість робіт також присвячена відновленню значення порядку дробових похідних. Зокрема у [147] вона розв'язується методом Левенберга–Марквардта [143], а у [141] методом Тихонова.

Другим підходом є, до певної міри протилежний до першого, підхід використання методів метаевристичної оптимізації. В межах такого підходу, зокрема, були отримані результати [57] щодо відновлення значення коефіцієнту теплопровідності у одновимірному дробово-диференціальному за часовою змінною рівнянні теплоперенесення методом мурашиних колоній [205]. Задачі ідентифікації систем дробового порядку розв'язується також методом композитної диференціальної еволюції [226] у [96], алгоритмом зозулі [236] у [104], алгоритмом рою частинок [249] у [242, 124], з застосуванням штучних нейронних мереж [275].

Варто відзначити, що більшість алгоритмів метаевристичної оптимізації є легко розпаралелюваними за даними. Ця властивість була використана, зокрема, у [124] для розробки відповідного алгоритму для квантових комп'ютерів.

1.7. Висновки до розділу 1

Сучасні засоби математичного моделювання аномальних міграційних процесів на базі дробово-диференціального підходу охоплюють широке коло моделей, які містять інтегро-диференціальні оператори за часовою та просторовими змінними. У випадку процесів у ґрунтах, складність використання вказаних моделей для опису

реальних випадків перенесення пов'язана зі складністю взаємодії фізичних, хімічних та біологічних чинників; розподіленістю геологічних та гідрологічних параметрів, низькою точністю їх інструментального визначення.

У такій ситуації, актуальним напрямком є комплексне чисельне дослідження моделей з різнотипними дробовими похідними; визначення особливостей динаміки процесів, що ними описуються; розробка обчислювально ефективних засобів визначення моделей, найкращих для опису спостережуваного процесу, ідентифікації їх параметрів та прогнозування розвитку гідрогеоміграційного процесу на їх основі.

За результатом аналізу розроблених на нинішній час методів розв'язання прямих та обернених задач щодо дробово-диференціальних моделей, зокрема моделей гідрогеоміграційних процесів з некласичною динамікою, можна стверджувати, що актуальним є розвиток скінченно-різницевих методів для розв'язання прямих задач та метаевристичних методів для задач ідентифікації параметрів разом з розробкою та вивченням алгоритмів для високопродуктивних обчислювальних систем, які б дозволили забезпечити значне зниження порядку обчислювальної складності до рівня, співставного з відповідним рівнем, що має місце для рівнянь з похідними цілого порядку в класичних геогідроміграційних математичних моделях.

Джерела, що використані у розділі 1

Для написання даного розділу було використано 174 джерела [1–5], [7], [11–13], [18], [20–22], [24], [26], [27–31], [49], [56], [57], [59], [62], [63], [66] [74–76], [80], [81], [83], [84], [87], [88], [90–98], [100–105], [107], [108], [110–122], [124], [125], [128], [130], [132], [135], [138], [139], [141], [143–150], [152–154], [157], [159], [162], [165], [166], [168], [171], [172], [174], [176–178], [181–184], [188–190], [193], [194], [198], [200], [202], [203], [205], [206], [209], [212], [213], [215], [217], [221], [223], [224], [226–230], [232–234], [236], [237–242], [245–247], [249], [251–255], [257], [258], [266], [267], [270–275], [279–289], [292], [293], [297–301], [304–308], [310–312], [314–316], [318–320], посилання на які зазначені в тексті розділу.

Розділ 2. Математичне моделювання дробово-диференціальної динаміки процесів конвективної дифузії розчинних речовин при фільтрації з джерел забруднень ґрунтових вод

Розробка методів математичного моделювання динаміки процесів конвективної дифузії розчинних речовин при фільтрації з різних джерел забруднень ґрунтових вод, є актуальною, зокрема, в аспекті охорони навколишнього середовища та активно розвивається [62, 63, 266, 267, 298, 299, 304]. При цьому актуальною є проблема підвищення ступеня адекватності наявних моделей дифузійних процесів при фільтрації в пористих середовищах. Особливо це стосується геопористих середовищ і систем зі складною просторово-часовою структурою, для яких характерні ефекти пам'яті, просторової нелокальності та самоорганізації. У цих умовах використання для моделювання динаміки нерівноважних конвективно-дифузійних процесів класичних математичних моделей не завжди є коректним і прогрес в моделюванні досягається з використанням підходу, що базується на використанні апарату дробового інтегро-диференціювання. При цьому, відповідні математичні моделі дифузії, які наразі інтенсивно розробляються, базуються на рівняннях дробового порядку [115, 139, 177].

2.1. Моделювання дробово-диференціальної динаміки процесу конвективної дифузії розчинних речовин при усталеній плановій фільтрації з кругового котловану

2.1.1. Математична модель та початково-крайова задача

Розглядаючи релаксаційний конвективно-дифузійний процес в середовищі зі складною просторово-часовою структурою, будемо виходити з такого узагальнення класичного закону Фіка [302]:

$$\vec{q}(x, y, t + \tilde{\tau}_q) = -d\nabla C + \vec{v}C, \quad (2.1)$$

де \vec{q} – конвективно-дифузійний потік, $кг/м^2с$, C – концентрація розчинних речовин в рідкій фазі, $кг/м^3$, \vec{v} – вектор швидкості фільтрації, $м/с$, d – коефіцієнт дифузії,

m^2/c , ∇ – оператор Гамільтона, $\tilde{\tau}_q$ – параметр релаксації потоку, x, y, t – геометричні та часова змінні, відповідно.

Вивчаючи дробово-диференціальну динаміку процесу "конвекція-дифузія з релаксацією" обмежимося в лівій частині співвідношення (2.1) першими двома членами розкладу в дробово-диференціальний ряд Тейлора [166] за часовою змінною. Тоді отримаємо дробово-диференціальне узагальнення релаксаційного закону Фіка у вигляді

$$\vec{q} + \tau_q D_t^{(\alpha)} \vec{q} = -d\nabla C + \vec{v}C, \quad (2.2)$$

де $D_t^{(\alpha)}$ – оператор дробового диференціювання Капуто–Герасимова за змінною t порядку α ($0 < \alpha < 1$) [139, 177], $\tau_q = \tilde{\tau}_q^\alpha / \Gamma(\alpha + 1)$, c^α , $\Gamma(z)$ – гамма-функція Ейлера [259].

Зауважимо, що рівняння (2.2) містить коефіцієнт τ_q розмірності нецілого порядку, який відображає властивості середовища фрактальної структури. Надалі при розгляді відповідних крайових задач вводитимемо спеціальним чином визначені безрозмірні змінні і параметри, зокрема, аналогічно [107].

У припущенні, що поле швидкостей фільтрації потенційне, з (2.2) отримуємо

$$\left(1 + \tau_q D_t^{(\alpha)}\right) \operatorname{div} \vec{q} = -d\Delta C + \vec{v} \cdot \nabla C, \quad (2.3)$$

де Δ – оператор Лапласа.

З урахуванням рівняння матеріального балансу

$$\sigma \frac{\partial C}{\partial t} + \operatorname{div} \vec{q} = 0, \quad (2.4)$$

де σ – пористість середовища, маємо з (2.3) рівняння моделі у вигляді [59]

$$\sigma \left(\frac{\partial}{\partial t} + \tau_q D_t^{(\alpha+1)} \right) C = d\Delta C - \vec{v} \cdot \nabla C \quad (0 < \alpha < 1). \quad (2.5)$$

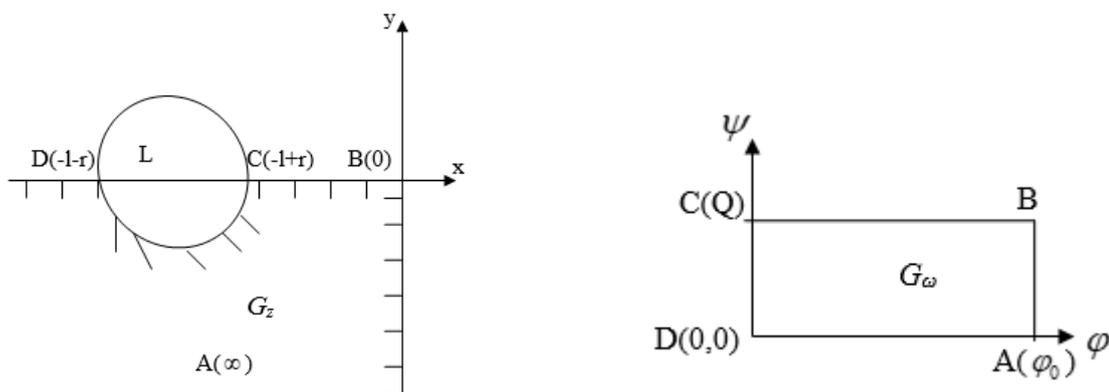
Відзначимо, що при $\alpha \rightarrow 1$ з (2.5), зокрема, маємо [279]

$$\sigma \left(\frac{\partial}{\partial t} + \tau_q \frac{\partial^2}{\partial t^2} \right) C = d\Delta C - \vec{v} \cdot \nabla C.$$

Розглянемо на основі неklasичної математичної моделі, що базується на рівнянні (2.5), задачу моделювання динаміки процесу конвективної дифузії при усталеній двовимірній плановій фільтрації з котловану кругового перетину радіусом $r > 0$ у дренажний канал відповідно до фільтраційної схеми, наведеної у [63] (рис. 2.1). Дана задача зводиться до відшукування в півплощині з круговим вирізом (рис. 2.1, а) розв'язку рівняння (2.5) при наступних крайових умовах:

$$C|_{\gamma_1} = C_1, \quad \left. \frac{\partial C}{\partial x} \right|_{x=0} = 0, \quad C(x, y, 0) = C_0(x, y), \quad C'_t(x, y, 0) = 0, \quad (2.6)$$

де C_1 – задана концентрація розчинних речовин на вході фільтраційного потоку (границі γ_1 сховища), $x=0$ – рівняння дренажного каналу (на границі каналу задається умова швидкого виносу розчинених речовин), $C_0(x, y)$ – функція початкового розподілу концентрації.



а) фізична область

б) область комплексного потенціалу течії

Рисунок 2.1 – Фільтраційна схема [63]

Оскільки область фільтрації – неканонічна область складної конфігурації, то ефективний спосіб розв'язання задач даного типу ґрунтується [63] на переході до нових змінних (φ, ψ) – точок геометрично простішої області G_ω комплексного потенціалу течії $\omega = \varphi + i\psi$ (φ – потенціал течії, ψ – функція течії), що є [298] в даній задачі прямокутником $G_\omega = \{(\varphi, \psi) : 0 < \varphi < \varphi_0, 0 < \psi < Q\}$ (рис. 2.1, б).

Крайову задачу, відповідну (2.5), (2.6), сформулюємо для області комплексного потенціалу течії у вигляді

$$\sigma \left(\frac{\partial}{\partial t} + \tau_q D_t^{(\alpha+1)} \right) C(\varphi, \psi, t) = \nu^2(\varphi, \psi) T^2 \left(d\Delta C(\varphi, \psi, t) - \frac{1}{T} \frac{\partial C}{\partial \varphi} \right), \quad (2.7)$$

$$((\varphi, \psi, t) \in G_\omega \times (0, +\infty)),$$

$$C(0, \psi, t) = C_1, \quad \frac{\partial C(\varphi_0, \psi, t)}{\partial \varphi} = 0, \quad \frac{\partial C}{\partial \psi} \Big|_{\psi=0, \psi=Q} = 0, \quad (2.8)$$

$$C(\varphi, \psi, 0) = C_0(\varphi, \psi), \quad C'_t(\varphi, \psi, 0) = 0, \quad (2.9)$$

де $\nu^2 = \nu_x^2 + \nu_y^2$, $\nu_x = \frac{1}{T} \frac{\partial \varphi}{\partial x}$, $\nu_y = \frac{1}{T} \frac{\partial \varphi}{\partial y}$, $\varphi_0 = \kappa HT$, $\Delta = \frac{\partial^2}{\partial \varphi^2} + \frac{\partial^2}{\partial \psi^2}$ – оператор Лапласа,

T – потужність напірного водоносного пласта, $Q = \pi \varphi_0 / \ln \left(\frac{l + \sqrt{l^2 - r^2}}{r} \right)$ –

фільтраційна витрата, l, r – геометричні параметри області (рис. 2.1, а), κ – коефіцієнт фільтрації, H – напір.

Введемо безрозмірні змінні та параметри співвідношеннями

$$x' = \frac{x}{r}, \quad y' = \frac{y}{r}, \quad \varphi' = \frac{\varphi}{\varphi_0}, \quad \psi' = \frac{\psi}{\varphi_0}, \quad C' = \frac{C}{C_1},$$

$$l' = \frac{l}{r}, \quad d' = \frac{dT}{\varphi_0}, \quad t' = \frac{a^2}{\varphi_0 T} t, \quad \tau'_q = \left(\frac{a^2}{\varphi_0 T} \right)^\alpha \tau_q, \quad (2.10)$$

де $a = 2Q / \pi r \sqrt{l^2 - 1}$.

Переходячи в (2.7)–(2.9) до безрозмірних змінних згідно з (2.10) та опускаючи надалі знак "штрих" над безрозмірними величинами, отримуємо в області $G_\omega \times (0, +\infty)$, $G_\omega = \{(\varphi, \psi) : 0 < \varphi < 1, 0 < \psi < \tilde{q}\}$ крайову задачу

$$\sigma \left(\frac{\partial}{\partial t} + \tau_q D_t^{(\alpha+1)} \right) C(\varphi, \psi, t) = V^2(\varphi, \psi) \left(d\Delta C - \frac{\partial C}{\partial \varphi} \right) \quad (0 < \alpha < 1), \quad (2.11)$$

$$C(0, \psi, t) = 1, \quad \frac{\partial C(1, \psi, t)}{\partial \varphi} = 0, \quad \frac{\partial C}{\partial \psi} \Big|_{\psi=0, \psi=\tilde{q}} = 0, \quad (2.12)$$

$$C(\varphi, \psi, 0) = C_0(\varphi, \psi), \quad C'_t(\varphi, \psi, 0) = 0, \quad (2.13)$$

де позначено [63]

$$V(\varphi, \psi) = \sin^2 \left(\frac{\pi\psi}{2\tilde{q}} \right) + sh^2 \left(\frac{\pi(\varphi-1)}{2\tilde{q}} \right), \quad \tilde{q} = \frac{\pi}{\ln(l + \sqrt{l^2 - 1})}.$$

2.1.2. Чисельне розв'язання крайової задачі

Наближений розв'язок задачі (2.11)–(2.13) можна отримати на основі скінченно-різницевого підходу в такий спосіб.

Введемо до розгляду сіткову область

$$\omega_{hr} = \{(\varphi_i, \psi_k, t_j) : \varphi_i = ih_1 \ (i = \overline{0, m+1}), \ \psi_k = h_2(k - 0,5) \ (k = \overline{0, n+1}), \\ t_j = j\tau \ (j = \overline{0, N})\},$$

де $h_1 = \frac{2}{2m+1}$, $h_2 = \frac{\tilde{q}}{n}$ – кроки сітки за геометричними змінними φ і ψ , відповідно,

τ – крок сітки за часовою змінною, і поставимо у відповідність крайовій задачі (2.11)–(2.13) наступний аналог [59] локально-одновимірної [313] різницевої схеми О. А. Самарського:

$$\frac{\sigma}{2} (C_t + \tau_q \Delta_t^{(\alpha+1)} \bar{C}) = V^2 \left(d\bar{C}_{\bar{\varphi}\varphi} - \bar{C}_{\varphi} \right), \quad (2.14)$$

$$\frac{\sigma}{2} (C_t + \tau_q \Delta_t^{(\alpha+1)} \hat{C}) = V^2 d\hat{C}_{\bar{\psi}\psi}. \quad (2.15)$$

Тут $\hat{C} = C^{j+1}$, $\bar{C} = C^{j+\frac{1}{2}}$, $C = C^j$, $t_{j+\frac{1}{2}} = t_j + \frac{\tau}{2}$, $\Delta_t^{(\alpha+1)}$ – різницевий аналог

оператора дробового диференціювання $D_t^{(\alpha+1)}$, який визначається як

$$\Delta_t^{(\alpha+1)} \hat{C} = \frac{4}{\tau^2 \Gamma(2-\alpha)} \sum_{s=0}^j b_s^{(j)} (C^{s+1} - 2C^{s+1/2} + C^s), \quad (2.16)$$

$$\Delta_t^{(\alpha+1)} \bar{C} = \frac{4}{\tau^2 \Gamma(2-\alpha)} \sum_{s=0}^j b_s^{(j)} (C^{s+1/2} - 2C^s + C^{s-1/2}), \quad (2.17)$$

$$b_s^{(j)} = \tau^{1-\alpha} \left[(j-s+1)^{1-\alpha} - (j-s)^{1-\alpha} \right] \ (j = \overline{0, N}). \quad (2.18)$$

Розписуючи у (2.14), (2.15) різницеві оператори з урахуванням (2.16), (2.17) та зводячи подібні члени, отримуємо на напівцілому та цілому часових шарах системи рівнянь

$$A_{ik} C_{i+1,k}^{j+\frac{1}{2}} - B_{ik} C_{ik}^{j+\frac{1}{2}} + S_{ik} C_{i-1,k}^{j+\frac{1}{2}} = \Phi_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.19)$$

$$P_{ik} C_{i,k+1}^{j+1} - Q_{ik} C_{ik}^{j+1} + P_{ik} C_{i,k-1}^{j+1} = \Omega_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.20)$$

де позначено

$$A_{ik} = \frac{V_{ik}^2}{h_1} \left(\frac{d}{h_1} - 0,5 \right), \quad S_{ik} = \frac{V_{ik}^2}{h_1} \left(\frac{d}{h_1} + 0,5 \right), \quad B_{ik} = \frac{\sigma}{\tau} + \frac{2\sigma\tau_q}{\tau^{\alpha+1}\Gamma(2-\alpha)} + A_{ik} + S_{ik},$$

$$P_{ik} = \frac{V_{ik}^2 d}{h_2^2}, \quad Q_{ik} = \frac{\sigma}{\tau} + \frac{2\sigma\tau_q}{\tau^{\alpha+1}\Gamma(2-\alpha)} + 2P_{ik},$$

$$\Phi_{ik}^j = \frac{\sigma}{\tau} \left[\frac{2\tau_q}{\tau \Gamma(2-\alpha)} \sum_{s=0}^{j-1} b_s^{(j)} (C_{ik}^{s+1/2} - 2C_{ik}^s + C_{ik}^{s-1/2}) - C_{ik}^j + \frac{2\tau_q}{\tau^\alpha \Gamma(2-\alpha)} (C_{ik}^{j-1/2} - 2C_{ik}^j) \right],$$

$$\Omega_{ik}^j = \frac{\sigma}{\tau} \left[\frac{2\tau_q}{\tau \Gamma(2-\alpha)} \sum_{s=0}^{j-1} b_s^{(j)} (C_{ik}^{s+1} - 2C_{ik}^{s+1/2} + C_{ik}^s) - C_{ik}^{j+1/2} + \frac{2\tau_q}{\tau^\alpha \Gamma(2-\alpha)} (C_{ik}^j - 2C_{ik}^{j+1/2}) \right],$$

$$\sum_{s=0}^{j-1} (\cdot) \equiv 0 \text{ при } j-1 < 0.$$

Різницеві рівняння систем (2.19), (2.20) є триточковими та ефективно розв'язуються методом прогонки [192, 313]. При цьому прогоночні співвідношення мають вигляд

$$C_{i,k}^{j+1/2} = \alpha_{i+1,k} C_{i+1,k}^{j+1/2} + \beta_{i+1,k}^j, \quad C_{ik}^{j+1} = \tilde{\alpha}_{i,k+1} C_{i,k+1}^{j+1} + \tilde{\beta}_{i,k+1}^j \quad (i = \overline{1, m}; k = \overline{1, n}),$$

а прогоночні коефіцієнти обчислюються за формулами

$$\alpha_{i+1,k} = \frac{A_{ik}}{B_{ik} - S_{ik} \alpha_{ik}}, \quad \beta_{i+1,k}^j = \frac{\alpha_{i+1,k}}{A_{ik}} (S_{ik} \beta_{ik}^j - \Phi_{ik}^j),$$

$$\tilde{\alpha}_{i,k+1} = \frac{P_{ik}}{Q_{ik} - P_{ik} \tilde{\alpha}_{ik}}, \quad \tilde{\beta}_{i,k+1}^j = \tilde{\alpha}_{i,k+1} \left(\tilde{\beta}_{ik}^j - \frac{\Omega_{ik}^j}{P_{ik}} \right) \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}).$$

Для стартових значень прогоночних коефіцієнтів отримуємо

$$\alpha_{1k} = 0, \quad \beta_{1k}^j = 1, \quad \tilde{\alpha}_{i1} = 1, \quad \tilde{\beta}_{i1}^j = 0 \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}).$$

Слід зазначити, що стійкість методу прогонки для систем (2.19), (2.20) витікає з факту діагонального переважання в матрицях коефіцієнтів цих СЛАР [313]. Подальший перехід у фізичну область G_z здійснюється відповідно до співвідношень [63]

$$x = \frac{l_*}{2} \frac{\operatorname{sh}\left(\frac{\pi}{Q}(\varphi - \varphi_0)\right)}{\operatorname{ch}^2\left(\frac{\pi}{2Q}(\varphi - \varphi_0)\right) - \cos^2\left(\frac{\pi\psi}{2Q}\right)},$$

$$y = -\frac{l_*}{2} \frac{\sin\left(\frac{\pi\psi}{Q}\right)}{\operatorname{ch}^2\left(\frac{\pi}{2Q}(\varphi - \varphi_0)\right) - \cos^2\left(\frac{\pi\psi}{2Q}\right)}, l_* = \sqrt{l^2 - r^2}.$$

2.1.3. Результати обчислювальних експериментів

Чисельне моделювання динаміки процесу міграції розчинних речовин в рамках неklasичної конвективно-дифузійної математичної моделі (2.7)–(2.9) виконано щодо безрозмірних змінних, які визначаються згідно з (2.10) для початкового розподілу поля концентрацій $C(\varphi, \varphi, 0) \equiv 0$. Вихідні коди розробленого програмного забезпечення наведено у [265]. Деякі з отриманих при цьому результатів графічно зображені на рис. 2.2–2.5 та відповідають динаміці полів концентрацій вздовж лінії течії $\psi = 0$.

Аналіз результатів чисельних експериментів дозволяє зробити наступні висновки [59] про особливості динаміки полів концентрацій розчинних речовин, досліджуваних в рамках моделі (2.7)–(2.9).

1. Для малих значень безрозмірного параметра релаксації τ_q криві концентрації в неklasичній моделі при різних значеннях порядку α дробової похідної якісно узгоджуються з відповідними кривими, отриманими в рамках класичної моделі конвективної дифузії. Наявні для зазначених τ_q на початку міграційного процесу відмінності в поведінці кривих концентрації в залежності від величини α в рамках неklasичної моделі (рис. 2.2) з розвитком міграційного процесу поступово зникають (рис. 2.3).

2. Зі збільшенням значення безрозмірного параметра релаксації τ_q при фіксованому значенні порядку дробової похідної α процес масоперенесення в рамках розглядуваної моделі набуває чітко виражений хвильовий характер зі зростаючою з часом амплітудою концентраційної хвилі (рис. 2.4).

3. Збільшення значення порядку α дробової похідної при фіксованих значеннях параметра релаксації τ_q також тягне за собою збільшення амплітуди концентраційної хвилі в кожен фіксований момент часу (рис. 2.5). Максимальне значення амплітуди хвилі, що досягається при $\alpha \rightarrow 1$, відповідає аналогу телеграфного рівняння.

Відзначимо, що такі властивості математичної моделі (2.7)–(2.9) істотно відрізняють її від моделей вивчених у [62, 63].

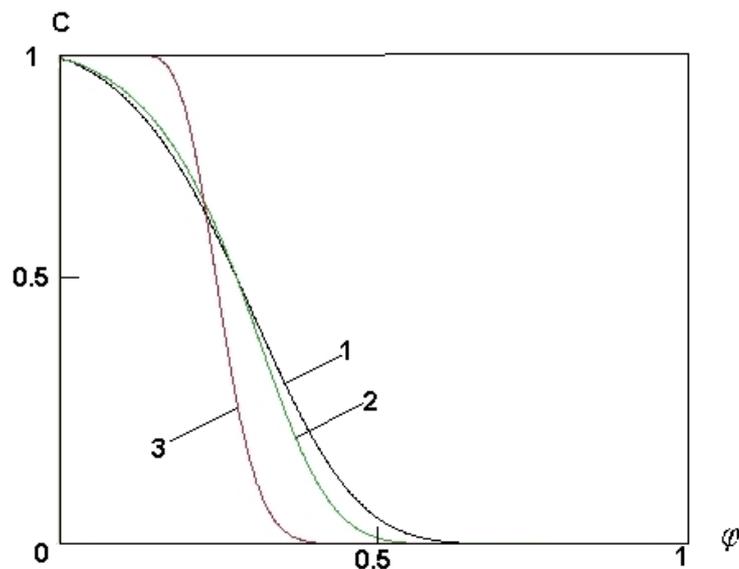


Рисунок 2.2 – Динаміка полів концентрацій вздовж лінії течії $\psi = 0$ при $t = 0,01$, $\tau_q = 0,005$ ($1 - \alpha = 0,6$; $2 - \alpha = 0,8$; $3 - \alpha = 1$).

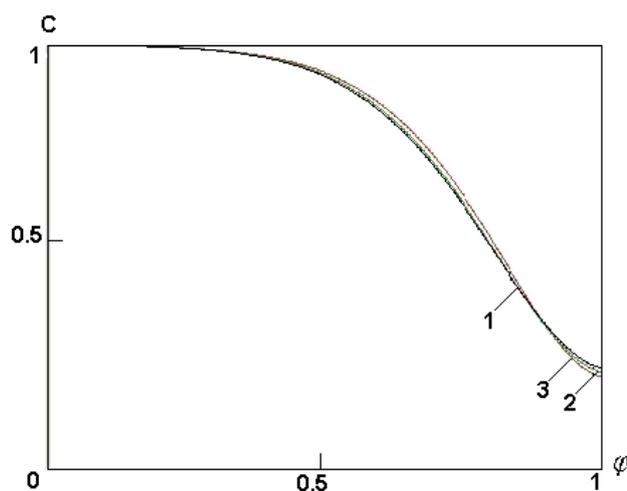


Рисунок 2.3 – Динаміка полів концентрацій вздовж лінії течії $\psi = 0$ при $t = 0,1$, $\tau_q = 0,005$ ($1 - \alpha = 0,6$; $2 - \alpha = 0,8$; $3 - \alpha = 1$)

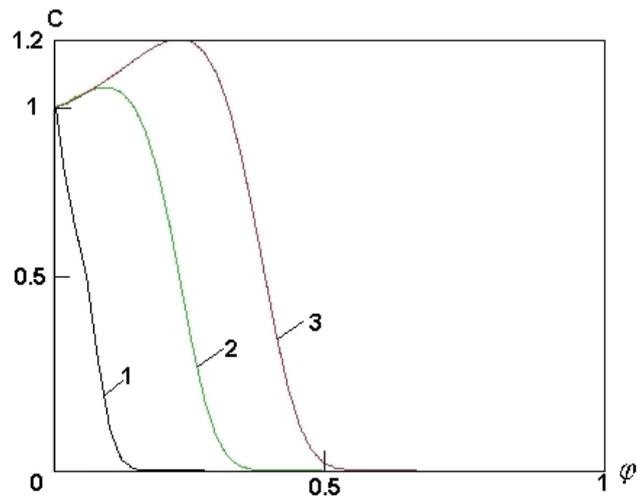


Рисунок 2.4 – Динаміка кривих концентрації для неklasичної моделі при $\alpha = 0,6$; $\tau_q = 50$ (1 – $t = 0,1$; 2 – $t = 0,5$; 3 – $t = 1$)

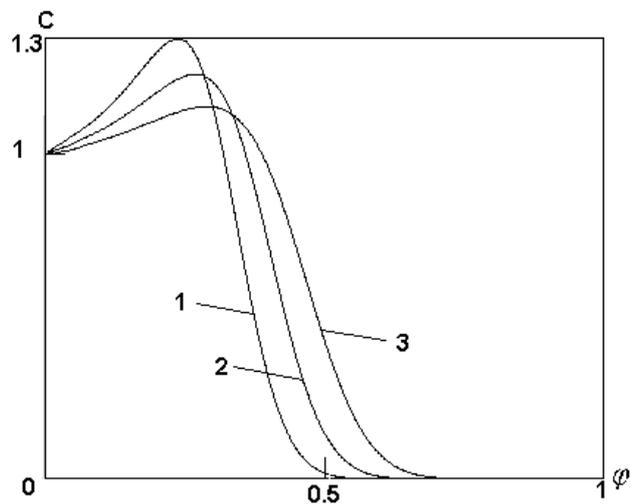


Рисунок 2.5 – Динаміка поля концентрацій у фіксований момент часу $t = 0,1$ в залежності від величини α порядку дробової похідної при $\tau_q = 0,5$ ($1 - \alpha = 0,8$; $2 - \alpha = 0,7$; $3 - \alpha = 0,6$).

2.2. Математичне моделювання дробово-диференціальної динаміки процесу конвективної дифузії розчинних речовин при фільтрації з річок та каналів

2.2.1. Математична модель та початково-крайова задача

Запишемо рівняння релаксаційного закону Фіка, що враховує час релаксації дифузійного потоку τ_q , у вигляді [302, 303]

$$\vec{q} + \tau_q \frac{\partial \vec{q}}{\partial t} = -d\nabla C + \vec{v}C. \quad (2.21)$$

За припущення виконання узагальненого рівняння матеріального балансу [85]

$$\sigma D_t^{(\alpha)} C + \operatorname{div} \vec{q} = 0, \quad (2.22)$$

де σ – пористість середовища, $D_t^{(\alpha)}$ – оператор дробового диференціювання Капуто–Герасимова за змінною t порядку α ($0 < \alpha < 1$) [115, 139, 177], зі співвідношення (2.21) з урахуванням рівняння нерозривності потоку отримуємо рівняння моделі у вигляді [64, 269]

$$\sigma \left(D_t^{(\alpha)} + \tau_q D_t^{(\alpha+1)} \right) C(x, y, t) = d\Delta C - \vec{v} \cdot \nabla C \quad (0 < \alpha < 1). \quad (2.23)$$

Звідси, при $\alpha \rightarrow 1$ маємо рівняння

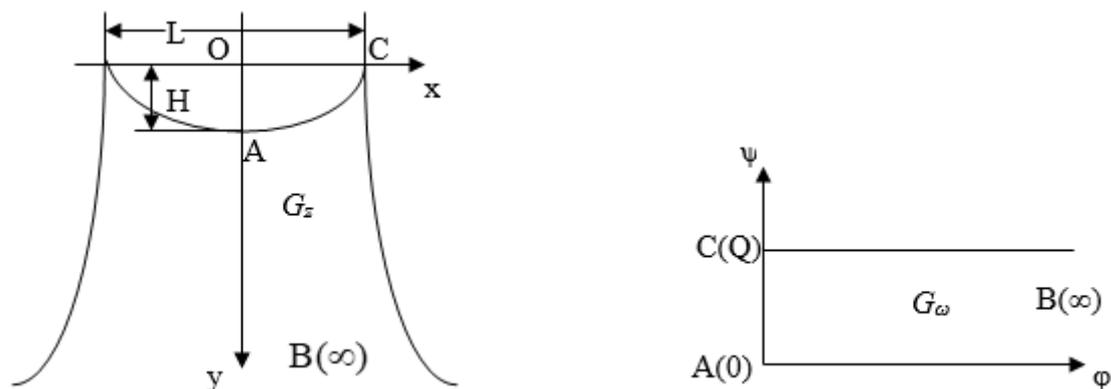
$$\sigma \left(\frac{\partial}{\partial t} + \tau_q \frac{\partial^2}{\partial t^2} \right) C = d\Delta C - \vec{v} \cdot \nabla C. \quad (2.24)$$

Зазначимо, що крайові задачі конвективної дифузії розчинних речовин при фільтрації в пористих середовищах відповідно до моделі типу Каттанео [77], яка визначається рівнянням (2.24), розглянуті, зокрема, в роботах [270, 279].

В рамках неklasичної математичної моделі конвективної дифузії, що базується на рівнянні (2.23), задача моделювання дробово-диференціальної динаміки нерівноважного в часі конвективно-дифузійного процесу у випадку плоско-вертикальної усталеної фільтрації з вільною поверхнею з річок або каналів [311] (область фільтрації в фізичній площині зображена на рис. 2.6, а) зводиться до розв'язання в області $G_z \times (0, +\infty)$ (G_z – область течії) рівняння (2.23) при крайових умовах

$$C|_{\gamma} = C_1, \quad \frac{\partial C}{\partial n} \Big|_{\gamma_i} = 0, \quad C(x, y, 0) = C_0(x, y), \quad C'_t(x, y, 0) = 0, \quad (2.25)$$

де C_1 – задана концентрація розчинних речовин на вході фільтраційного потоку з межею γ , γ_i ($i=1,2$) – межі області фільтрації, що є лініями течії, $C_0(x, y)$ – функція початкового розподілу концентрації, n – зовнішня нормаль до відповідної кривої.



а) фізична область

б) область комплексного потенціалу течії

Рисунок 2.6 – Фільтраційна схема [311]

Оскільки розв'язок фільтраційної задачі у вигляді характеристичної функції течії $z = f(\omega)$ відомий [311], переходячи у задачі (2.23), (2.25) до нових змінних (φ, ψ) – точок геометрично простішої області комплексного потенціалу течії $G_\omega = \{(\varphi, \psi) : 0 < \varphi < +\infty, 0 < \psi < Q\}$ (рис. 2.6, б) маємо

$$\sigma \left(D_t^{(\alpha)} + \tau_q D_t^{(\alpha+1)} \right) C(\varphi, \psi, t) = v^2(\varphi, \psi) \left(d\Delta C(\varphi, \psi, t) - \frac{\partial C}{\partial \varphi} \right),$$

$$((\varphi, \psi, t) \in G_\omega \times (0, +\infty))$$

$$C(0, \psi, t) = C_1, \quad \left. \frac{\partial C}{\partial \psi} \right|_{\psi=0, \psi=Q} = 0, \quad .$$

$$C(\varphi, \psi, 0) = C_0(\varphi, \psi), \quad C'_t(\varphi, \psi, 0) = 0,$$

де Q – фільтраційна витрата, v – швидкість фільтрації [311].

Для області, зображеної на рис. 2.6, а, розв'язок відповідної задачі усталеної фільтрації записується у вигляді [311]

$$x = He^{\frac{\pi\varphi}{2Q}} \sin\left(\frac{\pi\psi}{2Q}\right) + \frac{\psi}{\kappa}, \quad (2.26)$$

$$y = He^{\frac{\pi\varphi}{2Q}} \cos\left(\frac{\pi\psi}{2Q}\right) + \frac{\varphi}{\kappa}, \quad (2.27)$$

Звідси поле фільтраційних швидкостей визначається співвідношенням

$$v^2(\varphi, \psi) = \frac{4a}{\pi^2} \left(\frac{1}{e^{\pi\varphi} \sin^2\left(\frac{\pi\psi}{2}\right)} + \frac{1}{\left(e^{\frac{\pi\varphi}{2}} \cos\left(\frac{\pi\psi}{2}\right) + \kappa_1\right)^2} \right), \quad (2.28)$$

де κ – коефіцієнт фільтрації, $Q = \kappa\left(\frac{L}{2} - H\right)$ – фільтраційна витрата.

Вводячи до розгляду нові змінні та параметри співвідношеннями

$$\begin{aligned} \varphi' &= \frac{\varphi}{Q}, & \psi' &= \frac{\psi}{Q}, & t' &= \left(\frac{v_0^2}{Q}\right)^{\frac{1}{\alpha}} t, & C' &= \frac{C}{C_1}, & C_0' &= \frac{C_0}{C_1}, \\ v' &= \frac{v}{v_0}, & d' &= \frac{d}{Q}, & \tau'_q &= \left(\frac{v_0^2}{Q}\right)^{\frac{1}{\alpha}} \tau_q, \end{aligned} \quad (2.29)$$

та опускаючи надалі знак "штрих", отримуємо в області $G_\omega \times (0, +\infty)$ крайову задачу

$$\sigma\left(D_t^{(\alpha)} + \tau_q D_t^{(\alpha+1)}\right) C(\varphi, \psi, t) = v^2(\varphi, \psi) \left(d\Delta C - \frac{\partial C}{\partial \varphi} \right) \quad (0 < \alpha < 1), \quad (2.30)$$

$$C(0, \psi, t) = 1, \quad \left. \frac{\partial C}{\partial \psi} \right|_{\psi=0, \psi=1} = 0, \quad (2.31)$$

$$C(\varphi, \psi, 0) = C_0(\varphi, \psi), \quad C'_i(\varphi, \psi, 0) = 0, \quad (2.32)$$

де $G_\omega = \{(\varphi, \psi) : 0 < \varphi < +\infty, 0 < \psi < 1\}$ – горизонтальна півсмуга.

2.2.2. Чисельне розв'язання крайової задачі

Введемо до розгляду сіткову область

$$\begin{aligned} \omega_{h\tau} &= \{(\varphi_i, \psi_k, t_j) : \varphi_i = ih_1 \ (i = \overline{0, m+1}), \ \psi_k = h_2(k - 0,5) \ (k = \overline{0, n+1}), \\ & \quad t_j = j\tau \ (j = \overline{0, N})\}, \end{aligned}$$

де $h_1 = \frac{2\varphi_0}{2m+1}$, $h_2 = \frac{\tilde{q}}{n}$ – кроки сітки за геометричними змінним φ і ψ , відповідно,

τ – крок сітки за часовою змінною, $\varphi_0 = const$.

Обмежуючи область комплексного потенціалу течії справа прямою $\varphi = \varphi_0$ ($\varphi_0 \gg 1$) і задаючи на ній додаткову граничну умову (наприклад, однорідну умову

Неймана) поставимо у відповідність крайовій задач (2.30)–(2.32) наступний аналог [64] локально-одновимірної [313] різницевої схеми О. А. Самарського:

$$\frac{\sigma}{2}(\Delta_t^{(\alpha)}\bar{C} + \tau_q \Delta_t^{(\alpha+1)}\bar{C}) = \nu^2 \left(d\bar{C}_{\bar{\varphi}\bar{\varphi}} - \bar{C}_0 \right), \quad (2.33)$$

$$\frac{\sigma}{2}(\Delta_t^{(\alpha)}\hat{C} + \tau_q \Delta_t^{(\alpha+1)}\hat{C}) = \nu^2 d\hat{C}_{\bar{\psi}\bar{\psi}}. \quad (2.34)$$

Тут $\hat{C} = C^{j+1}$, $\bar{C} = C^{j+\frac{1}{2}}$, $C = C^j$, $t_{j+\frac{1}{2}} = t_j + \frac{\tau}{2}$, $\Delta_t^{(\alpha)}, \Delta_t^{(\alpha+1)}$ – різницеві аналоги

операторів дробового диференціювання $D_t^{(\alpha)}, D_t^{(\alpha+1)}$, відповідно. Дискретна апроксимація $D_t^{(\alpha)}$ визначається аналогічно [248] як

$$\Delta_t^{(\alpha)}\hat{C} = \frac{2}{\Gamma(1-\alpha)} \sum_{s=0}^j b_s^{(j)} (C^{s+1} - C^s), \quad (2.35)$$

$$\Delta_t^{(\alpha)}\bar{C} = \frac{2}{\Gamma(1-\alpha)} \sum_{s=0}^j \gamma_s^{(j)} (C^{s+1/2} - C^{s-1/2}),$$

$$\gamma_s^{(j)} = \tau^{1-\alpha} \left[\left(j-s+\frac{1}{2} \right)^{1-\alpha} - \left(j-s-\frac{1}{2} \right)^{1-\alpha} \right],$$

а $\Delta_t^{(\alpha+1)}$ – відповідно до (2.16)–(2.18).

Розписуючи в (2.33), (2.34) різницеві оператори з урахуванням (2.16)–(2.18), (2.35) і зводячи подібні члени, отримуємо на напівцілому та цілому часових шарах системи рівнянь

$$A_{ik} C_{i+1,k}^{j+\frac{1}{2}} - B_{ik} C_{ik}^{j+\frac{1}{2}} + S_{ik} C_{i-1,k}^{j+\frac{1}{2}} = \Phi_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.36)$$

$$P_{ik} C_{i,k+1}^{j+1} - Q_{ik} C_{ik}^{j+1} + R_{ik} C_{i,k-1}^{j+1} = \Omega_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.37)$$

де позначено

$$A_{ik} = \frac{\nu_{ik}^2}{h_1} \left(\frac{d}{h_1} - 0,5 \right), \quad S_{ik} = \frac{\nu_{ik}^2}{h_1} \left(\frac{d}{h_1} + 0,5 \right),$$

$$B_{ik} = \frac{\sigma}{\tau^\alpha \Gamma(2-\alpha)} \left(\frac{1}{2^{1-\alpha}} + \frac{2\tau_q}{\tau} \right) + A_{ik} + S_{ik},$$

$$P_{ik} = \frac{v_{ik}^2 d}{h_2^2}, \quad Q_{ik} = \frac{\sigma}{\tau^\alpha \Gamma(2-\alpha)} \left(\frac{1}{2^{1-\alpha}} + \frac{2\tau_q}{\tau} \right) + 2P_{ik},$$

$$\Phi_{ik}^j = \frac{\sigma}{\tau \Gamma(2-\alpha)} \left[\frac{1}{2} \sum_{s=0}^{j-1} \gamma_s^{(j)} (C_{ik}^{s+1} - C_{ik}^s) - \left(\frac{\tau}{2} \right)^{1-\alpha} C_{ik}^j + \frac{2\tau_q}{\tau^\alpha} (C_{ik}^{j-1/2} - 2C_{ik}^j) + \frac{2\tau_q}{\tau} \sum_{s=0}^{j-1} b_s^{(j)} (C_{ik}^{s+1/2} - 2C_{ik}^s + C_{ik}^{s-1/2}) \right],$$

$$\Omega_{ik}^j = \frac{\sigma}{\tau \Gamma(2-\alpha)} \left\{ \frac{1}{2} \sum_{s=0}^{j-1} b_s^{(j)} (C_{ik}^{s+1} - C_{ik}^s) + \left(\frac{\tau}{2} \right)^{1-\alpha} \left[(2^{1-\alpha} - 1)(C_{ik}^{j+1/2} - C_{ik}^j) - C_{ik}^{j+1/2} \right] + \frac{2\tau_q}{\tau^\alpha} (C_{ik}^j - 2C_{ik}^{j+1/2}) + \frac{2\tau_q}{\tau} \sum_{s=0}^{j-1} b_s^{(j)} (C_{ik}^{s+1} - 2C_{ik}^{s+1/2} + C_{ik}^s) \right\}.$$

Тут $\sum_{s=0}^{j-1} (\cdot) \equiv 0$ при $j-1 < 0$.

Різницеві рівняння систем (2.36), (2.37) є триточковими та ефективно розв'язуються методом прогонки [192, 313] аналогічно описаному в п. 2.1.

Для стартових значень прогоночних коефіцієнтів маємо

$$\alpha_{1k} = 0, \quad \beta_{1k}^j = 1, \quad \tilde{\alpha}_{i1} = 1, \quad \tilde{\beta}_{i1}^j = 0 \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}).$$

Слід зазначити, що стійкість методу прогонки для систем (2.36), (2.37) впливає з факту діагонального переважання в матрицях коефіцієнтів цих СЛАР [313]. Подальший перехід в фізичну область G_z здійснюється відповідно до (2.26), (2.27).

2.2.3. Результати обчислювальних експериментів

Чисельне моделювання динаміки процесу міграції розчинних речовин в рамках моделі, описаної у пп. 2.2.1, виконано щодо безрозмірних змінних, які визначаються співвідношеннями (2.29) при початковому розподілі поля концентрацій $C(\varphi, \varphi, 0) \equiv 0$. Вихідні коди розробленого програмного забезпечення наведено у [265]. Деякі з отриманих при цьому результатів графічно зображені на рис. 2.7–2.11 і відповідають динаміці полів концентрацій вздовж лінії течії $\psi = 0$.

Позначимо через M_1, M_2, M_3, M_4 математичні моделі, що базуються відповідно на рівняннях:

$$\sigma \frac{\partial C}{\partial t} = d\Delta C - \vec{v} \cdot \nabla C, \quad (M_1)$$

$$\sigma \frac{\partial C}{\partial t} + \tau_q \frac{\partial^2 C}{\partial t^2} = d\Delta C - \vec{v} \cdot \nabla C, \quad (M_2)$$

$$\sigma D_t^{(\alpha)} C = d\Delta C - \vec{v} \cdot \nabla C, \quad (M_3)$$

$$\sigma(D_t^{(\alpha)} C + \tau_q D_t^{(\alpha+1)} C) = d\Delta C - \vec{v} \cdot \nabla C. \quad (M_4)$$

Відзначимо при цьому, що модель M_1 – це класична математична модель конвективної дифузії [299, 311], M_2 – релаксаційна математична модель типу Каттанео [77, 85], M_3 – дробово-диференціальна математична модель, відповідна розглядуваній фільтраційній схемі, яка вивчалась у [62], M_4 – модель, що описується рівнянням (2.23).

Аналіз результатів чисельних експериментів дозволяє зробити наступні висновки [64] про особливості динаміки полів концентрацій розчинних речовин, яка описується моделлю M_4 , у порівнянні з іншими математичними моделями перерахованими вище:

1. На початкових стадіях розвитку конвективно-дифузійного процесу, для різних математичних моделей можливі різні сценарії поведінки в часі концентраційних кривих у фіксованій точці фільтраційної області, загальною рисою яких є наявність як інтервалів часу з запізненням формування поля концентрацій в рамках некласичних моделей в порівнянні з класичною моделлю M_1 , так і часових інтервалів на яких спостерігається випередження формування поля концентрацій в рамках некласичних моделей у порівнянні з класичною (рис. 2.7).

2. На більш пізніх стадіях розвитку процесу спостерігається стійке запізнювання формування полів концентрацій при моделюванні згідно з моделлю M_4 , як в порівнянні з класичною моделлю M_1 , так і у порівнянні з моделлю типу Каттанео M_2 (рис. 2.8).

3. Порівняння результатів моделювання поля концентрацій за моделями M_3 та M_4 демонструє переважно прискорене просування фронту концентрації, при моделюванні згідно з моделлю M_4 . При малих значеннях параметра τ_q це явище характерне для початкової стадії процесу (рис. 2.9), при чому з плином часу результати розрахунку за обидвома моделями практично збігаються (рис. 2.10). При великих значеннях параметра τ_q ця властивість проявляється не тільки на початковій, а і на значно більш віддалених від початкової стадіях розвитку конвективно-дифузійного процесу (рис. 2.11).

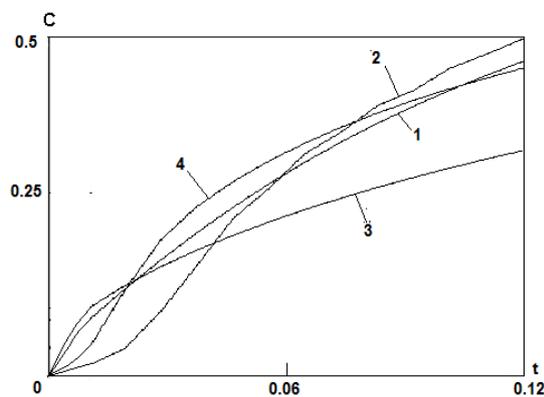


Рисунок 2.7 – Поведінка кривих концентрації в часі у фіксованій

точці $(0,12; 0)$ області комплексного потенціалу ($1 - \alpha = 1, \tau_q = 0,2$; $2 - \alpha = 1, \tau_q = 0,01$;

$3 - \alpha = 0,8, \tau_q = 0,4$; $4 - \alpha = 0,8, \tau_q = 0,01$)

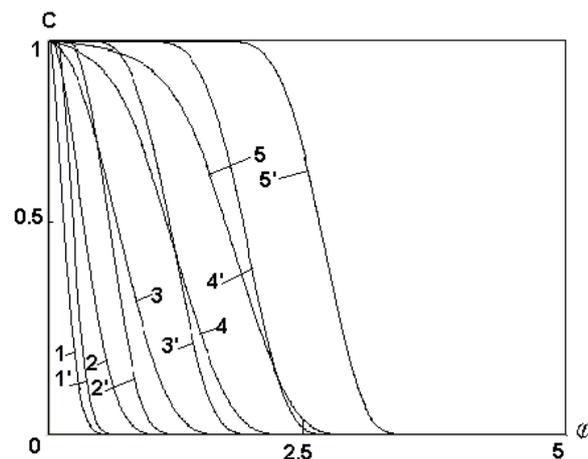


Рисунок 2.8 – Криві концентрації вздовж лінії симетрії течії $\psi = 0$ для значень $\alpha = 0,8; \tau_q = 0,01$ (криві 1–5) і значень, $\alpha = 1, \tau_q = 0,01$ (криві 1'–5')

моменти часу ($1,1' - t = 1$; $2,2' - t = 10$; $3,3' - t = 10^2$; $4,4' - t = 10^3$;

$5,5' - t = 10^4$).

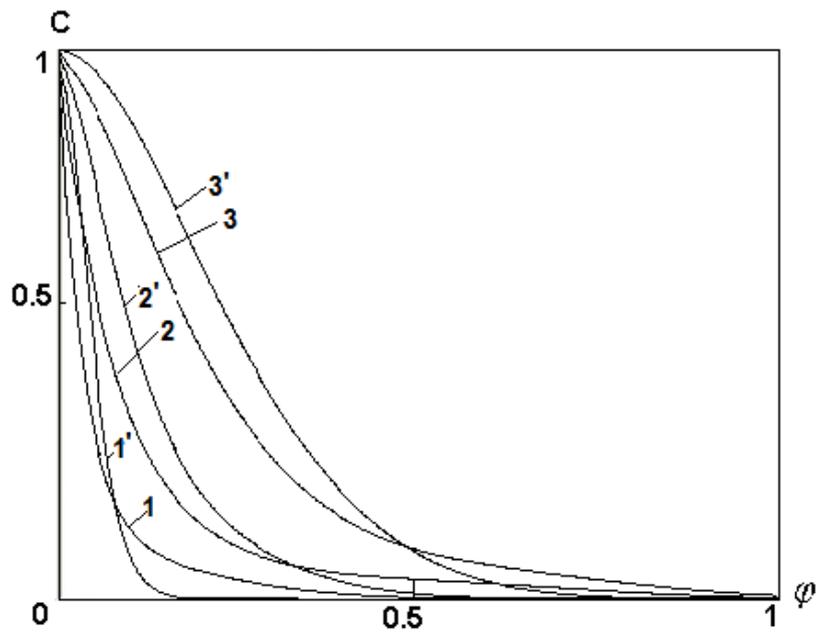


Рисунок 2.9 – Криві концентрації уздовж лінії $\psi = 0$ для значень $\alpha=0,8$; $\tau_q = 0$ (криві 1–3) та значень $\alpha = 0,8$; $\tau_q = 0,01$ (криві 1'–3') у фіксовані моменти часу (1,1' – $t = 0,01$; 2,2' – $t = 0,1$; 3,3' – $t = 1$).

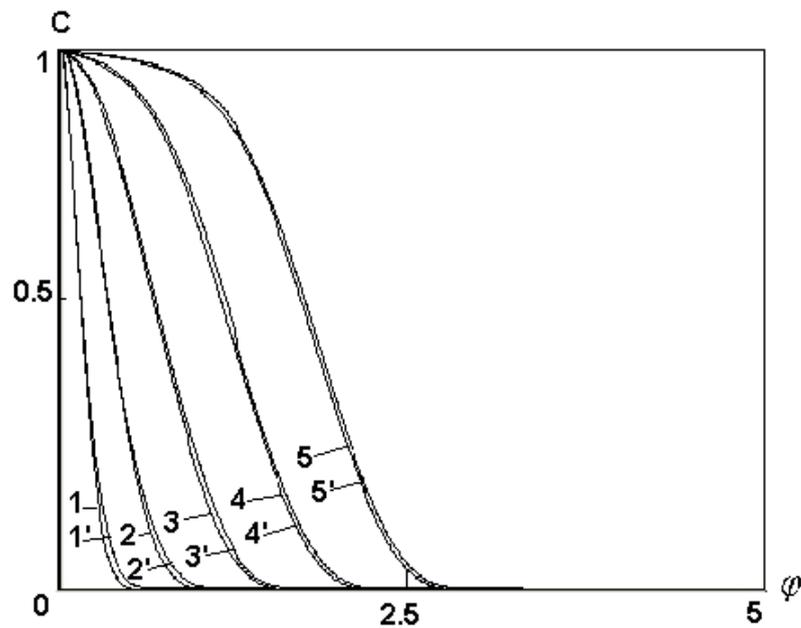


Рисунок 2.10 – Криві концентрації уздовж лінії $\psi = 0$ для значень $\alpha=0,8$; $\tau_q = 0$ (криві 1–5) та значень $\alpha = 0,8$; $\tau_q = 0,01$ (криві 1'–5') для віддалених від початкового стану стадій міграційного процесу (1,1' – $t = 1$; 2,2' – $t = 10$; 3,3' – 10^2 ; 4,4' – $t = 10^3$; 5,5' – $t = 10^4$)

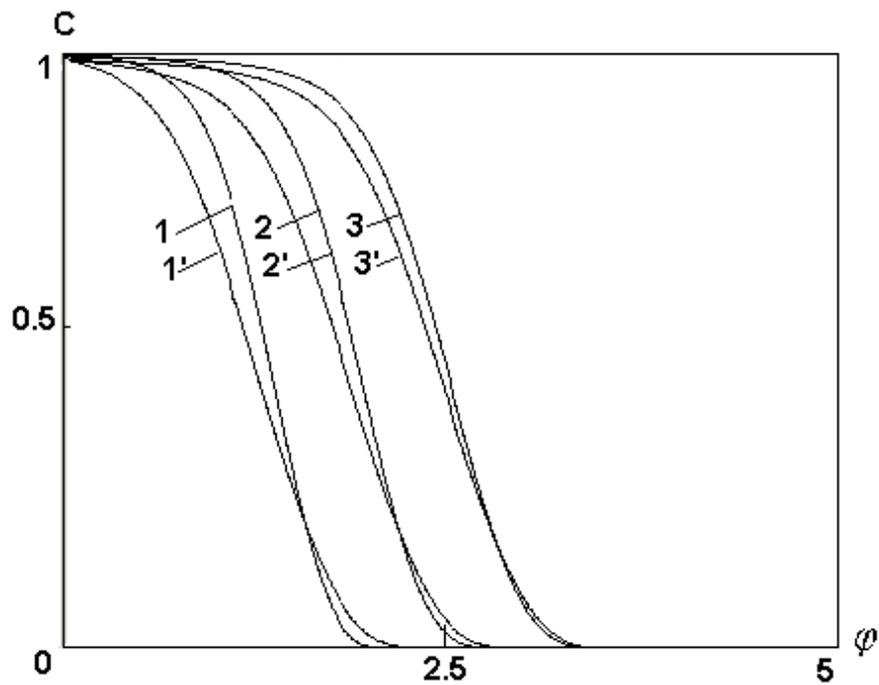


Рисунок 2.11 – Криві концентрації для значень $\alpha = 0,8$; $\tau_q = 100$ (криві 1–3) і значень $\alpha = 0,8$, $\tau_q = 0$ (криві 1'–3') у фіксовані моменти часу ($1,1' - t = 10^3$; $2,2' - t = 10^4$; $3,3' - t = 10^5$; $\psi = 0$).

2.3. Моделювання дробово-диференціальної динаміки процесу конвективної дифузії у неізотермічних умовах

2.3.1. Математична модель та початково-крайова задача

Загальновідомо, що неізотермічність процесу міграції розчинних речовин у фільтраційному потоці суттєвим чином впливає на динаміку його перебігу [77, 281, 303]. Розглядаючи в рамках дробово-диференціального підходу неізотермічний локально-нерівноважний в часі процес конвективної дифузії розчинних речовин у водонасиченому пористому середовищі зі складною просторово-часовою структурою, будемо виходити з наступного узагальнення закону Фіка:

$$\vec{q} = D_t^{1-\alpha} (-d\nabla C + \vec{v}C - d_T\nabla T), \quad (2.38)$$

де \vec{q} – конвективно-дифузійний потік, C – концентрація розчинних речовин в рідкій фазі, \vec{v} – швидкість фільтрації, d – коефіцієнт дифузії, d_T – коефіцієнт термодифузії,

$D_t^{1-\alpha}$ – оператор дробового диференціювання Рімана–Ліувілля порядку $1-\alpha$ ($0 < \alpha \leq 1$) [139, 177, 194, 318], ∇ – оператор Гамільтона.

Зі співвідношення (2.38) і рівняння матеріального балансу (2.4) отримуємо рівняння [53, 277]

$$\sigma D_t^{(\alpha)} C = d\Delta C - \operatorname{div}(C\vec{v}) + d_T \Delta T, \quad (2.39)$$

де $D_t^{(\alpha)} f(t) = \frac{1}{\Gamma(1-\alpha)} \frac{\partial}{\partial t} \left(\int_0^t (t-\tau)^{-\alpha} f(\tau) d\tau - \tau^{-\alpha} f(0) \right)$ – оператор регуляризованої дробової похідної Капуто–Герасимова [139, 177, 194, 318] порядку α , Δ – оператор Лапласа.

З урахуванням рівняння нерозривності фільтраційного потоку $\operatorname{div} \vec{v} = 0$ зі співвідношення (2.39) остаточно знаходимо

$$\sigma D_t^{(\alpha)} C = d\Delta C - \vec{v} \cdot \nabla C + d_T \Delta T. \quad (2.40)$$

Відзначимо, що при $\alpha \rightarrow 1$ з (2.40) отримуємо класичне рівняння конвективної дифузії з урахуванням термодифузії вигляду [281, 317]

$$\sigma C_t = d\Delta C - \vec{v} \cdot \nabla C + d_T \Delta T.$$

Рівняння для визначення аномального температурного поля у нерівноважних умовах запишемо у вигляді [115, 139, 177, 179, 317]

$$C_T D_t^{(\beta)} T = \lambda \Delta T - \rho C_\rho \vec{v} \cdot \nabla T, \quad (2.41)$$

де C_T – об'ємна теплоємність ґрунту, λ – коефіцієнт теплопровідності, ρ, C_ρ – щільність і питома теплоємність порового розчину, відповідно, β – порядок регуляризованої дробової похідної за часовою змінною ($0 < \beta \leq 1$).

Розглядаючи усталену плоско-вертикальну фільтрацію у потенційному полі швидкостей згідно зі схемою міграції забруднень з річок, каналів або поверхневих накопичувачів промислових стоків (пп. 2.2.1, рис. 2.6), отримуємо, що задача моделювання динаміки аномального процесу конвективної дифузії забруднень в неізотермічних умовах під впливом аномального температурного поля може бути сформульована як задача знаходження розв'язку системи рівнянь (2.40), (2.41) у області $((x, y, t) \in G_z \times (0, +\infty))$ при крайових умовах

$$C|_{AC} = C_1, \quad \frac{\partial C}{\partial n}\Big|_{AB, CB} = 0, \quad C|_{t=0} = 0, \quad (2.42)$$

$$T|_{AC} = T_1, \quad T|_{AB, CB} = T_2, \quad T|_{t=0} = 0, \quad (2.43)$$

де C_1, T_1 – задані концентрація розчинних речовин і температура на вході фільтраційного потоку, T_2 – задана температура на лініях течії АВ,СВ, n – зовнішня нормаль до відповідної кривої, G_z – права частина області фільтрації у фізичній площині (рис. 2.6, а).

У області комплексного потенціалу течії $G_\omega = \{(\varphi, \psi) : 0 < \varphi < +\infty, 0 < \psi < 1\}$ (рис. 2.6, б) крайова задача для дослідження динаміки розглядуваного міграційного процесу може бути сформульована у вигляді

$$\sigma D_t^{(\alpha)} C = \nu^2(\varphi, \psi) \left(d\Delta_{\varphi, \psi} C - \frac{\partial C}{\partial \varphi} + d_T \Delta_{\varphi, \psi} T \right), \quad (2.44)$$

$$C_T D_t^{(\beta)} T = \nu^2(\varphi, \psi) \left(\lambda \Delta_{\varphi, \psi} T - \rho C_\rho \frac{\partial T}{\partial \varphi} \right) \quad ((\varphi, \psi, t) \in G_\omega \times (0, +\infty)), \quad (2.45)$$

$$C|_{\varphi=0} = C_1, \quad \frac{\partial C}{\partial \psi}\Big|_{\psi=0, \psi=Q} = 0, \quad C|_{t=0} = 0, \quad (2.46)$$

$$T|_{\varphi=0} = T_1, \quad T|_{\psi=0, \psi=Q} = T_2, \quad T|_{t=0} = 0, \quad (2.47)$$

де $\Delta_{\varphi, \psi} = \frac{\partial^2}{\partial \varphi^2} + \frac{\partial^2}{\partial \psi^2}$ – оператор Лапласа, Q – фільтраційна витрата, $\nu^2 = \nu_x^2 + \nu_y^2$,

$\vec{v} = (\nu_x, \nu_y) = \nabla \varphi$, φ – потенціал швидкості, ψ – функція течії.

Введемо безрозмірні змінні та параметри співвідношеннями

$$x' = \frac{x}{L}, \quad y' = \frac{y}{L}, \quad \varphi' = \frac{\varphi}{Q}, \quad \psi' = \frac{\psi}{Q}, \quad H' = \frac{H}{L}, \quad C' = \frac{C}{C_1}, \quad T' = \frac{T}{T_1},$$

$$d' = \frac{d}{Q}, \quad \nu' = \frac{\nu}{\nu_0}, \quad t' = \left(\frac{\nu_0^2}{Q} \right)^{\frac{1}{\alpha}} t, \quad \kappa' = \frac{\kappa L}{Q}, \quad \kappa'_1 = \frac{2Q}{\pi \kappa H}, \quad a' = \left(\frac{Q}{\nu_0 H} \right)^2,$$

$$d'_T = \frac{T_1}{QC_1} d_T, \quad \lambda' = \frac{\lambda}{C_T} \left(\frac{Q^{\frac{\beta}{2\alpha}-1}}{\nu_0^{\frac{\beta}{\alpha}-1}} \right)^2, \quad \mu' = \frac{\rho C_\rho}{C_T} \left(\frac{Q}{\nu_0^2} \right)^{\frac{\beta}{\alpha}-1}, \quad \theta' = \frac{T_2}{T_1}, \quad (2.48)$$

де L – масштабний параметр, ν_0 – параметр швидкості.

Переходячи в (2.44)–(2.47) до безрозмірних змінних згідно з (2.48) і опускаючи надалі знак "штрих" над безрозмірними величинами, отримуємо в області $G_\omega \times (0, +\infty)$ крайову задачу

$$\sigma D_t^{(\alpha)} C = \nu^2(\varphi, \psi) \left(d\Delta_{\varphi, \psi} C - \frac{\partial C}{\partial \varphi} + d_T \Delta_{\varphi, \psi} T \right), \quad (2.49)$$

$$D_t^{(\beta)} T = \nu^2(\varphi, \psi) \left(\lambda \Delta_{\varphi, \psi} T - \mu \frac{\partial T}{\partial \varphi} \right), \quad (2.50)$$

$$C|_{\varphi=0} = 1, \quad \frac{\partial C}{\partial \psi} \Big|_{\psi=0, \psi=1} = 0, \quad C|_{t=0} = 0, \quad (2.51)$$

$$T|_{\varphi=0} = 1, \quad T|_{\psi=0, \psi=1} = \theta, \quad T|_{t=0} = 0. \quad (2.52)$$

2.3.2. Чисельне розв'язання крайової задачі

Введемо до розгляду сіткову область

$$\omega_{h\tau} = \left\{ (\varphi_i, \psi_k, t_j) : \varphi_i = ih_1 \left(i = \overline{0, m+1} \right), \psi_k = h_2(k-0,5) \left(k = \overline{0, n+1} \right), \right. \\ \left. t_j = j\tau \left(j = \overline{0, N+1} \right) \right\},$$

де $h_1 = \frac{2\varphi_0}{2m+1}$, $h_2 = \frac{1}{n}$ – кроки сітки за геометричними змінним φ і ψ відповідно,

τ – крок сітки за часовою змінною, $\varphi_0 = const$.

Обмежуючи область комплексного потенціалу справа прямою $\varphi = \varphi_0$ ($\varphi_0 \gg 1$) і задаючи на цій прямій додаткові граничні умови (наприклад, умови Неймана), поставимо у відповідність крайовій задачі (2.49)–(2.52) наступний аналог [53, 277] локально-одновимірної [313] різницевої схеми О. А. Самарського:

$$\frac{\sigma}{2} \Delta_t^{(\alpha)} \bar{C} = \nu^2 \left(d\bar{C}_{\bar{\varphi}\varphi} - \bar{C}_\varphi + d_T \bar{T}_{\bar{\varphi}\varphi} \right), \quad (2.53)$$

$$\frac{\sigma}{2} \Delta_t^{(\alpha)} \hat{C} = \nu^2 \left(d\hat{C}_{\bar{\psi}\psi} + d_T \hat{T}_{\bar{\psi}\psi} \right), \quad (2.54)$$

$$\frac{1}{2} \Delta_t^{(\beta)} \bar{T} = \nu^2 \left(\lambda \bar{T}_{\bar{\varphi}\varphi} - \mu \bar{T}_\varphi \right), \quad (2.55)$$

$$\frac{1}{2}\Delta_t^{(\beta)}\hat{T} = \nu^2 \lambda \hat{T}_{\bar{\nu}\bar{\nu}}, \quad (2.56)$$

де $\hat{C} = C^{j+1}$, $\bar{C} = C^{j+\frac{1}{2}}$, $C = C^j$, $t_{j+\frac{1}{2}} = t_j + \frac{\tau}{2}$, $\hat{T} = T^{j+1}$, $\bar{T} = T^{j+\frac{1}{2}}$, $T = T^j$,

$\Delta_t^{(\alpha)}, \Delta_t^{(\beta)}$ – різницеві аналоги операторів дробового диференціювання $D_t^{(\alpha)}, D_t^{(\beta)}$, які визначаються співвідношеннями [317]

$$\Delta_t^{(\alpha)} y = \frac{1}{\Gamma(2-\alpha)} \sum_{s=0}^j r_s^{(j)} y_{t,s}, \quad r_s^{(j)} = \tau^{1-\alpha} \left[(j-s+1)^{1-\alpha} - (j-s)^{1-\alpha} \right],$$

$$\Delta_t^{(\beta)} y = \frac{1}{\Gamma(2-\beta)} \sum_{s=0}^j b_s^{(j)} y_{t,s}, \quad b_s^{(j)} = \tau^{1-\beta} \left[(j-s+1)^{1-\beta} - (j-s)^{1-\beta} \right], \quad y_{t,s} = \frac{y^{s+1} - y^s}{\tau},$$

$\Gamma(\alpha)$ – гамма-функція [262].

Для отримання розв'язку задачі (2.49), (2.51) щодо поля концентрацій на напівцілому часовому шарі $t_{j+\frac{1}{2}}$ розпишемо в (2.53) різницеві оператори та зводячи

подібні члени отримаємо систему рівнянь

$$A_{ik} C_{i+1,k}^{j+\frac{1}{2}} - B_{ik} C_{ik}^{j+\frac{1}{2}} + S_{ik} C_{i-1,k}^{j+\frac{1}{2}} = \Phi_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.57)$$

де позначено

$$A_{ik} = \frac{\nu_{ik}^2}{h_1} \left(\frac{d}{h_1} - 0,5 \right), \quad S_{ik} = \frac{\nu_{ik}^2}{h_1} \left(\frac{d}{h_1} + 0,5 \right), \quad B_{ik} = \frac{\sigma}{2^{1-\alpha} \tau^\alpha \Gamma(2-\alpha)} + A_{ik} + S_{ik},$$

$$\Phi_{ik}^j = \frac{\sigma}{2\Gamma(2-\alpha)} \left(\sum_{s=0}^{j-1} \gamma_s^{(j)} \frac{C_{ik}^{s+1} - C_{ik}^s}{\tau} - \left(\frac{2}{\tau} \right)^\alpha C_{ik}^j \right) - \frac{d_T \nu_{ik}^2}{2h_1} \left(T_{i-1,k}^{j+\frac{1}{2}} - 2T_{ik}^{j+\frac{1}{2}} + T_{i+1,k}^{j+\frac{1}{2}} \right),$$

$$\gamma_s^{(j)} = \tau^{1-\alpha} \left[\left(j + \frac{1}{2} - s \right)^{1-\alpha} - \left(j - \frac{1}{2} - s \right)^{1-\alpha} \right].$$

Розв'язок системи (2.57) шукаємо згідно з методом прогонки у вигляді

$$C_{ik}^{j+\frac{1}{2}} = \alpha_{i+1,k} C_{i+1,k}^{j+\frac{1}{2}} + \beta_{i+1,k}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.58)$$

де прогоночні коефіцієнти обчислюються за формулами

$$\alpha_{i+1,k} = \frac{A_{ik}}{B_{ik} - S_{ik} \alpha_{ik}}, \quad \beta_{i+1,k}^j = \frac{\alpha_{i+1,k}}{A_{ik}} (S_{ik} \beta_{ik}^j - \Phi_{ik}^j) \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}).$$

Стартові значення прогонічних коефіцієнтів визначаються з урахуванням різницевого аналога граничної умови при $\varphi = 0$: $C_{0k}^j = 1$. Звідси маємо

$$\alpha_{1k} = 0, \quad \beta_{1k}^j = 1 \quad (k = \overline{1, n}; j = \overline{0, N}).$$

При цьому з різницевого аналога однорідного граничної умови другого роду в точці $\varphi = \varphi_0$: $C_{mk}^{j+\frac{1}{2}} = C_{m+1,k}^{j+\frac{1}{2}}$ ($k = \overline{1, n}; j = \overline{0, N}$) з урахуванням (2.58) знаходимо

$$C_{m+1,k}^{j+\frac{1}{2}} = \frac{\beta_{m+1,k}^j}{1 - \alpha_{m+1,k}} \quad (k = \overline{1, n}; j = \overline{0, N}).$$

На цілому часовому шарі з (2.54) отримуємо

$$P_{ik} C_{i,k+1}^{j+1} - Q_{ik} C_{ik}^{j+1} + P_{ik} C_{i,k-1}^{j+1} = \Omega_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.59)$$

де

$$P_{ik} = \frac{\nu_{ik}^2 d}{h_2^2}, \quad Q_{ik} = \frac{\sigma}{2^{1-\alpha} \tau^\alpha \Gamma(2-\alpha)} + 2P_{ik},$$

$$\Omega_{ik}^j = \frac{\sigma}{\Gamma(2-\alpha)} \left(\sum_{s=0}^j \rho_s^{(j)} \frac{C_{ik}^{s+\frac{1}{2}} - C_{ik}^s}{\tau} - \frac{1}{2^{1-\alpha} \tau^\alpha} C_{ik}^{j+\frac{1}{2}} + \sum_{s=0}^{j-1} \eta_s^{(j)} \frac{C_{ik}^{s+1} - C_{ik}^{s+\frac{1}{2}}}{\tau} \right) -$$

$$- \frac{d_T \nu_{ik}^2}{h_2^2} (T_{i,k-1}^{j+1} - 2T_{ik}^{j+1} + T_{i,k+1}^{j+1}),$$

$$\rho_s^{(j)} = \tau^{1-\alpha} \left[(j+1-s)^{1-\alpha} - \left(j + \frac{1}{2} - s \right)^{1-\alpha} \right], \quad \eta_s^{(j)} = \tau^{1-\alpha} \left[\left(j + \frac{1}{2} - s \right)^{1-\alpha} - (j-s)^{1-\alpha} \right].$$

Розв'язок системи (2.59) шукаємо у вигляді

$$C_{ik}^{j+1} = \tilde{\alpha}_{i,k+1} C_{i,k+1}^{j+1} + \tilde{\beta}_{i,k+1}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.60)$$

де

$$\tilde{\alpha}_{i,k+1} = \frac{P_{ik}}{Q_{ik} - P_{ik} \tilde{\alpha}_{ik}}, \quad \tilde{\beta}_{i,k+1}^j = \tilde{\alpha}_{i,k+1} \left(\tilde{\beta}_{ik}^j - \frac{\Omega_{ik}^j}{P_{ik}} \right) \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}).$$

З урахуванням різницевої граничної умови $C_{i0}^{j+1} = C_{i1}^{j+1}$ ($i = \overline{1, m}; j = \overline{0, N}$)

отримуємо стартові значення прогонічних коефіцієнтів у вигляді

$$\tilde{\alpha}_{i1} = 1, \quad \tilde{\beta}_{i1}^j = 0 \quad (i = \overline{1, m}; j = \overline{0, N}),$$

а з урахуванням умови $C_{in}^{j+1} = C_{i,n+1}^{j+1}$ ($i = \overline{1, m}; j = \overline{0, N}$) маємо

$$C_{i,n+1}^{j+1} = \frac{\tilde{\beta}_{i,n+1}^j}{1 - \tilde{\alpha}_{i,n+1}} \quad (i = \overline{1, m}; j = \overline{0, N}).$$

Для задачі (2.50), (2.52) щодо температурного поля, розписуючи в (2.55) різницеві оператори і зводячи подібні члени, отримуємо на напівцілому шарі $t^{j+\frac{1}{2}}$

систему рівнянь

$$R_{ik} T_{i+1,k}^{j+\frac{1}{2}} - U_{ik} T_{ik}^{j+\frac{1}{2}} + G_{ik} T_{i-1,k}^{j+\frac{1}{2}} = W_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.61)$$

де позначено

$$R_{ik} = \frac{\nu_{ik}^2}{h_1} \left(\frac{\lambda}{h_1} - 0,5\mu \right), \quad G_{ik} = \frac{\nu_{ik}^2}{h_1} \left(\frac{\lambda}{h_1} + 0,5\mu \right), \quad U_{ik} = \frac{1}{2^{1-\beta} \tau^\beta \Gamma(2-\beta)} + R_{ik} + G_{ik},$$

$$W_{ik}^j = \frac{1}{2\Gamma(2-\beta)} \left(\sum_{s=0}^{j-1} \tilde{\gamma}_s^{(j)} \frac{T_{ik}^{s+1} - T_{ik}^s}{\tau} - \left(\frac{2}{\tau} \right)^\beta T_{ik}^j \right),$$

$$\tilde{\gamma}_s^{(j)} = \tau^{1-\beta} \left[\left(j + \frac{1}{2} - s \right)^{1-\beta} - \left(j - \frac{1}{2} - s \right)^{1-\beta} \right].$$

Розв'язок системи (2.61) шукаємо у вигляді

$$T_{ik}^{j+\frac{1}{2}} = \tilde{\alpha}_{i+1,k} T_{i+1,k}^{j+\frac{1}{2}} + \tilde{\beta}_{i+1,k}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.62)$$

де прогоночні коефіцієнти обчислюються за формулами

$$\tilde{\alpha}_{i+1,k} = \frac{R_{ik}}{U_{ik} - G_{ik} \tilde{\alpha}_{ik}}, \quad \tilde{\beta}_{i+1,k}^j = \frac{\tilde{\alpha}_{i+1,k}}{R_{ik}} \left(G_{ik} \tilde{\beta}_{ik}^j - W_{ik}^j \right) \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}).$$

Стартові значення прогоночних коефіцієнтів визначаються з урахуванням різницевого аналога граничної умови при $\varphi = 0$: $T_{0k}^j = 1$. Звідси маємо

$$\tilde{\alpha}_{1k} = 0, \quad \tilde{\beta}_{1k}^j = 1 \quad (k = \overline{1, n}; j = \overline{0, N}).$$

При цьому з різницевого аналога однорідного граничної умови другого роду в точці $\varphi = \varphi_0$: $T_{mk}^{j+\frac{1}{2}} = T_{m+1,k}^{j+\frac{1}{2}}$ ($k = \overline{1, n}$; $j = \overline{0, N}$) з урахуванням (2.62) знаходимо

$$T_{m+1,k}^{j+\frac{1}{2}} = \frac{\tilde{\beta}_{m+1,k}^j}{1 - \tilde{\alpha}_{m+1,k}} \quad (k = \overline{1, n}; j = \overline{0, N}).$$

На цілому часовому шарі з (2.56) отримуємо

$$M_{ik} T_{i,k+1}^{j+1} - N_{ik} T_{ik}^{j+1} + M_{ik} T_{i,k-1}^{j+1} = F_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.63)$$

де

$$M_{ik} = \frac{\lambda v_{ik}^2}{h_2^2}, \quad N_{ik} = \frac{1}{2^{1-\beta} \tau^\beta \Gamma(2-\beta)} + 2M_{ik},$$

$$F_{ik}^j = \frac{1}{\Gamma(2-\beta)} \left(\sum_{s=0}^{j-1} \tilde{\rho}_s^{(j)} \frac{T_{ik}^{s+1} - T_{ik}^s}{\tau} - \frac{1}{2^{1-\beta} \tau^\beta} T_{ik}^{j+\frac{1}{2}} \right),$$

$$\tilde{\rho}_s^{(j)} = \tau^{1-\beta} \left[(j+1-s)^{1-\beta} - \left(j + \frac{1}{2} - s \right)^{1-\beta} \right].$$

Розв'язок системи (2.63) шукаємо у вигляді

$$T_{ik}^{j+1} = \alpha_{i,k+1}^* T_{i,k+1}^{j+1} + \beta_{i,k+1}^{*j} \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (2.64)$$

де прогоночні коефіцієнти обчислюються за формулами

$$\alpha_{i,k+1}^* = \frac{M_{ik}}{N_{ik} - M_{ik} \alpha_{ik}^*}, \quad \beta_{i,k+1}^{*j} = \alpha_{i,k+1}^* \left(\beta_{i,k}^{*j} - \frac{F_{ik}^j}{M_{ik}} \right) \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}).$$

З граничних умов $T|_{\psi=0, \psi=1} = \theta$ знаходимо

$$\alpha_{i1}^* = 0, \quad \beta_{i1}^{*j} = \theta, \quad T_{i,n+1}^{j+1} = \theta \quad (i = \overline{1, m}; j = \overline{0, N}).$$

Стійкість методу прогонки для (2.57), (2.59), (2.61), (2.63) випливає з факту діагонального переважання в матрицях коефіцієнтів цих СЛАР. Подальший перехід у фізичну область G_z здійснюється відповідно до (2.26), (2.27).

2.3.3. Результати обчислювальних експериментів

Чисельне моделювання динаміки неізотермічного локально-нерівноважного в часі процесу міграції розчинних речовин виконано щодо безрозмірних змінних, які

визначаються співвідношеннями (2.48) для вхідних даних з роботи [267]. Вихідні коди розробленого програмного забезпечення наведено у [265]. Деякі з отриманих при цьому результатів графічно зображені на рис. 2.12, 2.13.

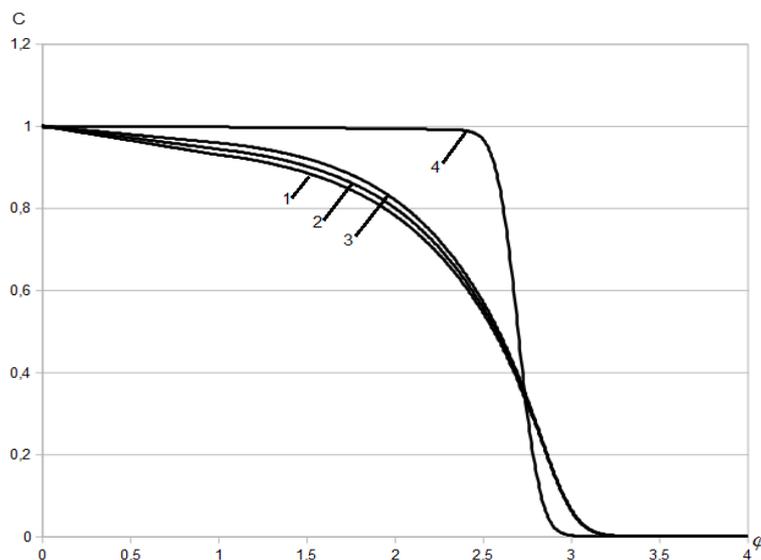


Рисунок 2.12 – Криві концентрації уздовж лінії течії $\psi = 0,5$ у фіксований момент часу $t = 0,1$ в залежності від величини порядку дробової похідної β ($1 - \beta = 1, \alpha = 0,85$; $2 - \beta = 0,9; \alpha = 0,85$; $3 - \beta = 0,85; \alpha = 0,85$; $4 - \beta = \alpha = 1$; $d_T = 0,0005$)

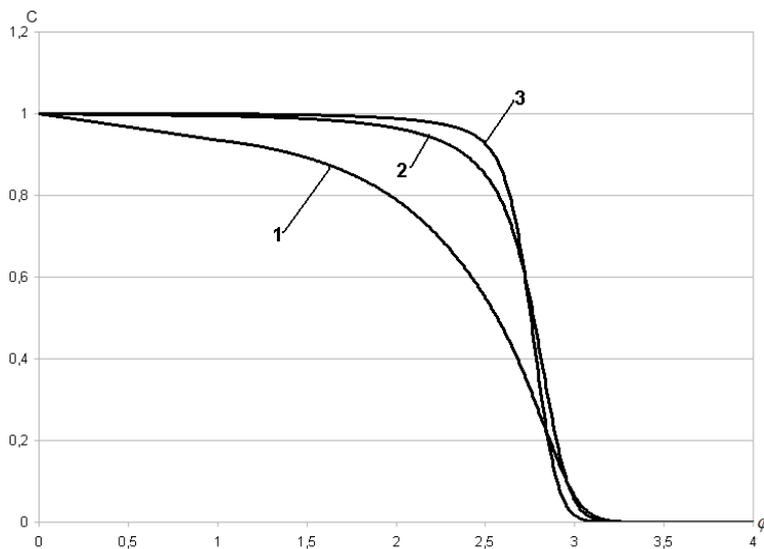


Рисунок 2.13 – Криві концентрації уздовж лінії течії $\psi = 0,5$ у фіксований момент часу $t = 0,1$ в залежності від величини порядку дробової похідної α ($1 - \alpha = 0,85$; $2 - \alpha = 0,9$; $3 - \alpha = 0,95$; $\beta = 0,95$, $d_T = 0,0005$)

Отримані результати дозволяють, зокрема, зробити висновок про те, що дробово-диференціальна динаміка неізотермічних процесів фільтраційно-конвективної дифузії переважно є не настільки інтенсивною, як відповідна динаміка зазначених неізотермічних процесів при її моделюванні в рамках класичних математичних моделей.

2.4. Моделювання дробово-диференціальної динаміки процесу конвективної дифузії на основі моделі з k -похідною Капуто

2.4.1. Математична модель та початково-крайова задача

При математичному описі динаміки нелокального в часі процесу конвективної дифузії досить поширеною є модель, що базується на наступному дробово-диференціальному аналогу класичного рівняння конвективної дифузії [15, 62, 63, 176, 196]:

$$\sigma D_t^{(\alpha)} C = d \Delta C - \operatorname{div}(C \vec{v}), \quad (2.65)$$

де $C = C(\varphi, \psi, t)$ – функція концентрації, \vec{v} – конвективна швидкість, d – коефіцієнт дифузії, σ – пористість середовища, $D_t^{(\alpha)}$ – оператор дробової похідної Капуто–Герасимова [115, 139, 177, 194] порядку α ($0 < \alpha < 1$), Δ – оператор Лапласа.

Однак, при вивченні нелокального в часі процесу конвективної дифузії розчинних речовин у водонасиченому пористому середовищі зі складною просторово-часовою структурою, математична модель, що базується на рівнянні виду (2.65) з параметром α може виявитися не достатньо адекватною. Наприклад, як зазначається в [15], у разі наявності у модельованому середовищі шарів з істотно різними властивостями, пряме використання однопараметричної моделі (2.65) без обґрунтування виконання умов її застосування, може привести до неприйнятних помилок у результатах прогнозування процесу. У цьому випадку в [15] пропонується здійснити перехід від однопараметричної моделі до відповідної моделі з більшим числом параметрів, що може в ряді випадків спростити її калібрування.

Таким чином, пропонується при математичному моделюванні динаміки нерівноважного в часі процесу конвективної дифузії виходити з такого узагальнення рівняння (2.65):

$$\sigma^k D_t^\alpha C = d\Delta C - \operatorname{div}(C\vec{v}), \quad (2.66)$$

де ${}^k D_t^\alpha$ – оператор двопараметричної дробової k – похідної Капуто порядку α ($0 < \alpha < 1$), яка визначається співвідношенням [78]

$${}^k D_t^\alpha f(t) = \frac{1}{k\Gamma_k(1-\alpha)} \int_0^t \frac{f'(\tau)}{(t-\tau)^{\frac{k+\alpha-1}{k}}} d\tau,$$

де $\Gamma_k(z) = \int_0^\infty t^{z-1} e^{-\frac{t^k}{k}} dt$ – k -гамма-функція [78].

Рівняння (2.65) при цьому є частинним випадком (2.66) при $k \rightarrow 1$.

В умовах фільтрації у потенційному полі швидкостей, з урахуванням рівняння нерозривності $\operatorname{div} \vec{v} = 0$, з (2.66) отримуємо [50]

$$\sigma^k D_t^\alpha C = d\Delta C - \vec{v} \cdot \nabla C, \quad (2.67)$$

$$\vec{v} = (v_x, v_y) = \nabla \varphi, \quad \operatorname{div} \vec{v} = 0, \quad (2.68)$$

де φ – потенціал швидкості фільтрації.

Зазначимо, що, зокрема, при $\alpha, k \rightarrow 1$ з (2.68) отримуємо класичне рівняння конвективної дифузії [311] $\sigma C_t = d\Delta C - \vec{v} \cdot \nabla C$.

Припустимо, що для фільтраційної схеми відома область комплексного потенціалу течії $\omega = \varphi + i\psi$ (ψ – функція течії), а також розв'язок відповідної задачі фільтрації, тобто відома характеристична функція течії $z = f(\omega)$. Тоді, переходячи в (2.67) до нових незалежних змінних – точок області комплексного потенціалу течії (φ, ψ) перепишемо це рівняння у вигляді

$$\sigma^k D_t^\alpha C(\varphi, \psi, t) = v^2(\varphi, \psi) \left(d \left(\frac{\partial^2 C}{\partial \varphi^2} + \frac{\partial^2 C}{\partial \psi^2} \right) - \frac{\partial C}{\partial \varphi} \right), \quad (2.69)$$

де $v^2 = v_x^2 + v_y^2$.

Будемо розглядати фільтраційну схему, відповідну задачі конвективної дифузії забруднень з річок, каналів або поверхневих накопичувачів промислових стоків, що описана у пп. 2.2.1.

Задача дослідження нелокального в часі процесу міграції забруднень в правій частині G_z симетричної області фільтрації (рис. 2.6, а) математично може бути сформульована як задача відшукування розв'язку рівняння (2.69) $((x, y, t) \in G_z \times (0, +\infty))$ при крайових умовах

$$C|_{AC} = C_1, \quad \frac{\partial C}{\partial n} \Big|_{AB, CB} = 0, \quad C|_{t=0} = 0, \quad (2.70)$$

де C_1 – задана концентрація розчинних речовин на вході фільтраційного потоку, n – зовнішня нормаль до відповідної кривої.

Крайова задача для дослідження динаміки розглядуваного міграційного процесу математично може бути сформульована для області комплексного потенціалу $G_\omega = \{(\varphi, \psi) : 0 < \varphi < +\infty, 0 < \psi < Q\}$ (рис. 2.6, б) у вигляді

$$\sigma {}^k D_t^\alpha C = \nu^2(\varphi, \psi) \left(d \left(\frac{\partial^2 C}{\partial \varphi^2} + \frac{\partial^2 C}{\partial \psi^2} \right) - \frac{\partial C}{\partial \varphi} \right) \quad ((\varphi, \psi, t) \in G_\omega \times (0, +\infty)), \quad (2.71)$$

$$C|_{AC} = C_1, \quad \frac{\partial C}{\partial \psi} \Big|_{AB, CB} = 0, \quad C|_{t=0} = 0. \quad (2.72)$$

Будемо розв'язувати задачу (2.71), (2.72) при таких обмеженнях на параметри похідної за часовою змінною: $0 < \alpha < 1$, $k > 1 - \alpha$. Легко показати, що в цьому випадку виконується

$${}^k D_t^\alpha C = \frac{1}{k^{1-\beta}} D_t^{(\beta)} C,$$

де $D_t^{(\beta)}$ – оператор стандартної похідної Капуто–Герасимова порядку β за змінною t [115, 139, 177, 194], $\beta = \frac{k + \alpha - 1}{k}$ ($0 < \beta < 1$). Тоді рівняння (2.71)

перепишеться у вигляді

$$\sigma D_t^{(\beta)} C(\varphi, \psi, t) = \tilde{\nu}^2(\varphi, \psi) \left(d \left(\frac{\partial^2 C}{\partial \varphi^2} + \frac{\partial^2 C}{\partial \psi^2} \right) - \frac{\partial C}{\partial \varphi} \right), \quad (2.73)$$

де $\tilde{\nu}^2(\varphi, \psi) = k^{1-\beta} \nu^2(\varphi, \psi)$ і моделювання динаміки аномального процесу конвективної дифузії зводиться до розв'язання в області комплексного потенціалу течії крайової задачі для рівняння (2.73) при умовах (2.72).

Переходячи в (2.73), (2.72) до безрозмірних змінних згідно з (2.48) і опускаючи надалі знак "штрих" над безрозмірними величинами, отримуємо в області $G_\omega \times (0, +\infty)$ крайову задачу

$$\sigma D_t^{(\beta)} C(\varphi, \psi, t) = \tilde{v}^2(\varphi, \psi) \left(d \left(\frac{\partial^2 C}{\partial \varphi^2} + \frac{\partial^2 C}{\partial \psi^2} \right) - \frac{\partial C}{\partial \varphi} \right), \quad (2.74)$$

$$C(0, \psi, t) = 1, \quad \frac{\partial C(\varphi, 0, t)}{\partial \psi} = 0, \quad \frac{\partial C(\varphi, 1, t)}{\partial \psi} = 0, \quad C(\varphi, \psi, 0) = 0, \quad (2.75)$$

де $\tilde{v}^2(\varphi, \psi) = k^{1-\beta} v^2(\varphi, \psi)$, $v^2(\varphi, \psi)$ визначається згідно з (2.28).

2.4.2. Чисельне розв'язання крайової задачі

Введемо до розгляду сіткову область

$$\omega_{hr} = \left\{ (\varphi_i, \psi_k, t_j) : \varphi_i = ih_1 \left(i = \overline{0, m+1} \right), \psi_k = h_2(k-0,5) \left(k = \overline{0, n+1} \right), \right. \\ \left. t_j = j\tau \left(j = \overline{0, N+1} \right) \right\},$$

де $h_1 = \frac{2\varphi_0}{2m+1}$, $h_2 = \frac{1}{n}$ – кроки сітки за геометричними змінним φ і ψ , відповідно,

τ – крок сітки за часовою змінною, $\varphi_0 = const$.

Обмежуючи область комплексного потенціалу справа прямою $\varphi = \varphi_0$ ($\varphi_0 \gg 1$) і задаючи на цій прямій додаткову граничну умову (наприклад, умову Неймана), поставимо у відповідність крайовій задачі (2.74), (2.75) наступний аналог [50] локально-одновимірної [192, 313] різницевої схеми О. А. Самарського:

$$\frac{\sigma}{2} \Delta_t^{(\beta)} \bar{C} = \tilde{v}^2 \left(d \bar{C}_{\bar{\varphi}\varphi} - \bar{C}_\varphi \right), \quad (2.76)$$

$$\frac{\sigma}{2} \Delta_t^{(\beta)} \hat{C} = \tilde{v}^2 d \hat{C}_{\bar{\psi}\psi}, \quad (2.77)$$

де $\hat{C} = C^{j+1}$, $\bar{C} = C^{j+\frac{1}{2}}$, $C = C^j$, $t_{j+\frac{1}{2}} = t_j + \frac{\tau}{2}$, $\Delta_t^{(\beta)} \bar{C}$, $\Delta_t^{(\beta)} \hat{C}$ – різницеві аналоги

оператора дробового диференціювання, що визначаються як

$$\Delta_t^{(\beta)} \bar{C} = \frac{1}{\Gamma(2-\beta)} \left[\left(\frac{2}{\tau} \right)^\beta (\bar{C} - C) + \sum_{s=0}^{j-1} b_s^{(j)} C_{t,s} \right],$$

$$\Delta_t^{(\beta)} \hat{C} = \frac{1}{\Gamma(2-\beta)} \left\{ \sum_{s=0}^{j-1} \rho_s^{(j)} C_{t,s} + \left(\frac{2}{\tau} \right)^\beta \left[\hat{C} - C + 2^{1-\beta} (\bar{C} - C) \right] \right\},$$

$$b_s^{(j)} = \tau^{1-\beta} \left[(j-s+\frac{1}{2})^{1-\beta} - (j-s-\frac{1}{2})^{1-\beta} \right], \quad \rho_s^{(j)} = \tau^{1-\beta} \left[(j-s+1)^{1-\beta} - (j-s)^{1-\beta} \right],$$

$$C_{t,s} = \frac{C^{s+1} - C^s}{\tau}, \quad \Gamma(\alpha) - \text{гамма-функція Ейлера [259]}.$$

Розписуючи в (2.76) різницеві оператори та зводячи подібні члени, отримуємо на напівцілому шарі $t_{j+\frac{1}{2}}$ систему рівнянь

$$A_{il} C_{i+1,l}^{j+\frac{1}{2}} - B_{il} C_{il}^{j+\frac{1}{2}} + S_{il} C_{i-1,l}^{j+\frac{1}{2}} = \Phi_{il}^j \quad (i = \overline{1, m}; l = \overline{1, n}; j = \overline{0, N}), \quad (2.78)$$

де

$$A_{il} = \frac{\tilde{\nu}_{il}^2}{h_1} \left(\frac{d}{h_1} - 0,5 \right), \quad S_{il} = \frac{\tilde{\nu}_{il}^2}{h_1} \left(\frac{d}{h_1} + 0,5 \right), \quad B_{il} = \frac{\sigma}{2^{1-\beta} \tau^\beta \Gamma(2-\beta)} + A_{il} + S_{il},$$

$$\Phi_{il}^j = \frac{\sigma}{2\Gamma(2-\beta)} \left(\sum_{s=0}^{j-1} b_s^{(j)} \frac{C_{il}^{s+1} - C_{il}^s}{\tau} - \left(\frac{2}{\tau} \right)^\beta C_{il}^j \right),$$

Розв'язок системи (2.78) шукаємо згідно з методом прогонки у вигляді

$$C_{il}^{j+\frac{1}{2}} = \alpha_{i+1,l} C_{i+1,l}^{j+\frac{1}{2}} + \beta_{i+1,l}^j \quad (i = \overline{1, m}; l = \overline{1, n}; j = \overline{0, N}), \quad (2.79)$$

$$\alpha_{i+1,l} = \frac{A_{il}}{B_{il} - S_{il} \alpha_{il}}, \quad \beta_{i+1,l}^j = \frac{\alpha_{i+1,l}}{A_{il}} (S_{il} \beta_{il}^j - \Phi_{il}^j) \quad (i = \overline{1, m}; l = \overline{1, n}; j = \overline{0, N}).$$

Стартові значення прогоночних коефіцієнтів визначаються з урахуванням різницевого аналога граничної умови при $\varphi = 0$: $C_{0l}^j = 1$. Звідси маємо

$$\alpha_{1l} = 0, \quad \beta_{1l}^j = 1 \quad (l = \overline{1, n}; j = \overline{0, N}).$$

З різницевого аналога однорідного граничної умови другого роду в точці $\varphi = \varphi_0$: $C_{ml}^{j+\frac{1}{2}} = C_{m+1,l}^{j+\frac{1}{2}}$ ($l = \overline{1, n}; j = \overline{0, N}$) з урахуванням (2.79) знаходимо

$$C_{m+1,l}^{j+\frac{1}{2}} = \frac{\beta_{m+1,l}^j}{1 - \alpha_{m+1,l}} \quad (l = \overline{1, n}; j = \overline{0, N}).$$

На цілому часовому шарі з (2.77) отримуємо

$$P_{il}C_{i,l+1}^{j+1} - Q_{il}C_{il}^{j+1} + P_{il}C_{i,l-1}^{j+1} = \Omega_{il}^j \quad (i = \overline{1, m}; l = \overline{1, n}; j = \overline{0, N}), \quad (2.80)$$

де

$$P_{il} = \frac{\tilde{\nu}_{il}^2 d}{h_2^2}, \quad Q_{il} = \frac{\sigma}{2^{1-\beta} \tau^\beta \Gamma(2-\beta)} + 2P_{il},$$

$$\Omega_{il}^j = \frac{\sigma}{2\Gamma(2-\beta)} \left\{ \sum_{s=0}^{j-1} \rho_s^{(j)} \frac{C_{il}^{s+1} - C_{il}^s}{\tau} + \left(\frac{2}{\tau}\right)^\beta \left[2^{1-\beta} C_{il}^{j+\frac{1}{2}} - (1+2^{1-\beta})C_{il}^j \right] \right\}.$$

Розв'язок системи (2.80) запишемо у вигляді

$$C_{il}^{j+1} = \tilde{\alpha}_{i,l+1} C_{i,l+1}^{j+1} + \tilde{\beta}_{i,l+1}^j \quad (i = \overline{1, m}; l = \overline{1, n}; j = \overline{0, N}), \quad (2.81)$$

де

$$\tilde{\alpha}_{i,l+1} = \frac{P_{il}}{Q_{il} - P_{il}\tilde{\alpha}_{il}}, \quad \tilde{\beta}_{i,l+1}^j = \tilde{\alpha}_{i,l+1} \left(\tilde{\beta}_{il}^j - \frac{\Omega_{il}^j}{P_{il}} \right) \quad (i = \overline{1, m}; l = \overline{1, n}; j = \overline{0, N}).$$

З урахуванням різницевої граничної умови $C_{i0}^{j+1} = C_{i1}^{j+1} \quad (i = \overline{1, m}; j = \overline{0, N})$

отримуємо стартові значення прогоночних коефіцієнтів

$$\tilde{\alpha}_{i1} = 1, \quad \tilde{\beta}_{i1}^j = 0 \quad (i = \overline{1, m}; j = \overline{0, N}),$$

а з урахуванням умови $C_{in}^{j+1} = C_{i,n+1}^{j+1} \quad (i = \overline{1, m}; j = \overline{0, N})$ знаходимо

$$C_{i,n+1}^{j+1} = \frac{\tilde{\beta}_{i,n+1}^j}{1 - \tilde{\alpha}_{i,n+1}} \quad (i = \overline{1, m}; j = \overline{0, N}).$$

Стійкість методу прогонки для (2.78) та (2.80) впливає з факту діагонального переважання в матрицях коефіцієнтів цих СЛАР. Подальший перехід в фізичну область G_z здійснюється відповідно до (2.26), (2.27).

2.4.3. Результати обчислювальних експериментів

Чисельне моделювання динаміки процесу міграції розчинних речовин в рамках математичної моделі з k -похідною Капуто за часовою змінною виконано щодо безрозмірних змінних (2.48) для вхідних даних з роботи [62]. Вихідні коди розробленого програмного забезпечення наведено у [265]. Деякі з отриманих при цьому результатів графічно зображені на рис. 2.14–2.16.

Аналіз результатів численних експериментів дозволяє зробити наступні висновки [50]:

1. Спостерігається запізнювання у формуванні поля концентрацій при моделюванні дифузійного процесу на основі моделі з k -похідною Капуто як в порівнянні випадком моделювання на основі класичної моделі [311], так і в порівнянні з моделлю, що базується на стандартному визначенні похідної Капуто (рис. 2.14).

2. Для фіксованих значень порядку дробової похідної α ($0 < \alpha < 1$), при зміні величини параметра k в рамках моделі з k -похідною Капуто запізнювання просування фронту концентрації є тим більшим, чим менше значення приймає параметр k (рис. 2.14, 2.15).

3. При фіксованих значеннях параметра k і змінних значеннях α , також спостерігається запізнювання просування фронту концентрації для моделі з k -похідною Капуто в порівнянні з випадком класичної математичної моделі, причому запізнювання є тим більшим, чим менше значення приймає параметр α (криві 2–4 на рис. 2.16). Динаміка запізнювання фронту концентрації, що моделюється в рамках математичної моделі з k -похідною Капуто в більшій мірі визначається зміною величини порядку похідної α , ніж зміною величини параметра k .

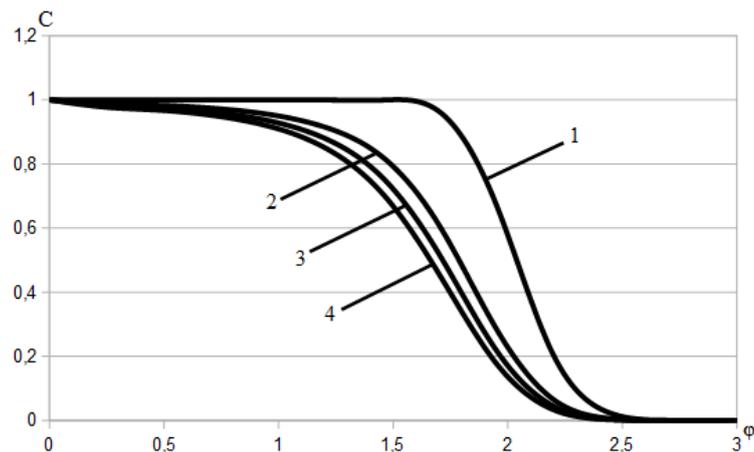


Рисунок 2.14 – Криві концентрацій вздовж лінії течії $\psi = 0,5$ при $t = 5$ для класичної моделі (крива 1), моделі зі стандартною похідною Капуто $D_t^{(\alpha)} C$

(крива 2, $\alpha = 0,7$) і моделі з k -похідною Капуто ${}^k D_t^\alpha C$

(криві 3,4; 3 – $k = 0,8$; 4 – $k = 0,7$; $\alpha = 0,7$)

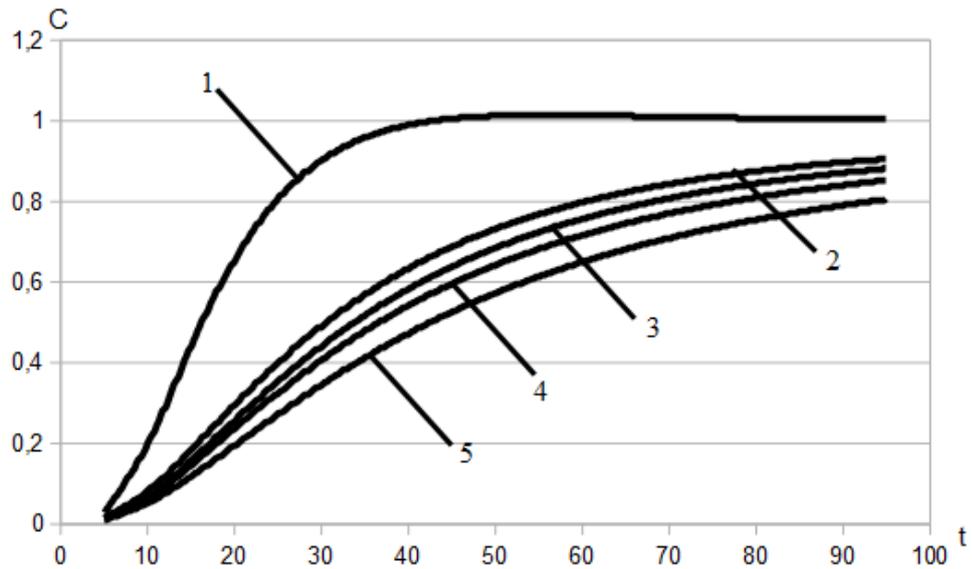


Рисунок 2.15 – Динаміка концентрації у точці ($\varphi = 2,44$; $\psi = 0,5$) для різних математичних моделей: крива 1 – класична модель конвективної дифузії; крива 2 – модель зі стандартною похідною Капуто $D_t^{(\alpha)}C$, $\alpha = 0,9$; криві 3–5 – моделі з k -похідною Капуто (3 – $k = 0,9$; 4 – $k = 0,8$; 5 – $k = 0,7$; 3,4,5 – $\alpha = 0,9$)

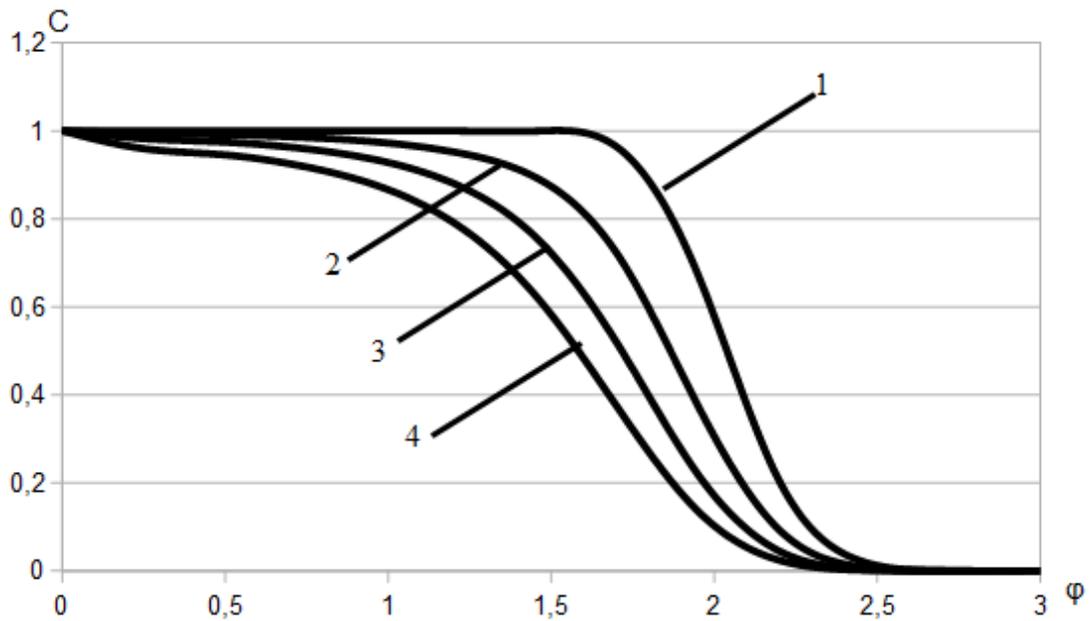


Рисунок 2.16 – Криві концентрацій вздовж лінії течії $\psi = 0,5$ при $t = 5$ для класичної моделі конвективної дифузії (крива 1) і моделі з k -похідною Капуто ${}^k D_t^\alpha C$ (криві 2–4; 2 – $\alpha = 0,9$; 3 – $\alpha = 0,8$; 4 – $\alpha = 0,7$; $k = 0,8$)

2.5. Висновки до розділу 2

У розділі наведені результати математичного моделювання процесів конвективної дифузії у двовимірному наближенні на основі розроблених моделей, що містять похідні дробового порядку за часовою змінною. Не зменшуючи загальності результатів, моделювання поля швидкостей проводилось на основі фільтраційних задач, що мають відомий аналітичний розв'язок, отримуваний методом конформних відображень в область комплексного потенціалу течії. Чисельні розв'язки початково-крайових задач моделювання динаміки нелокальних процесів конвективної дифузії отримані з використанням локально-одновимірних скінченно-різницевої схем.

У п. 2.1 сформульована математична модель та поставлена початково-крайова задача моделювання дробово-диференціальної динаміки процесу конвективної дифузії розчинних речовин при усталеній плановій фільтрації з кругового котловану. Результати чисельного моделювання показали, що процес масоперенесення в рамках такої моделі має хвильовий характер з амплітудою концентраційної хвилі, що зростає з часом, а також при збільшенні значення порядку похідної α .

У п. 2.2 виконано математичне моделювання дробово-диференціальної динаміки аномального процесу конвективної дифузії при усталеній двовимірній фільтрації з річок та каналів. Запропонована в цьому пункті математична модель міграційного процесу є більш загальною ніж модель, розглянута у п. 2.1, і враховує як час релаксації дифузійного потоку, так і часову нелокальність досліджуваних процесів. Поставлена відповідна даній моделі крайова задача, розроблено алгоритм її чисельного розв'язання у випадку вивчення динаміки міграційного процесу для схеми фільтрації з річок та каналів. Наводиться порівняння результатів чисельного моделювання згідно з цією моделлю та трьома іншими моделями конвективної дифузії — з похідними цілого та дробового порядку без врахування релаксаційних процесів та релаксаційною моделлю з похідними цілого порядку. Так, на початкових стадіях розвитку процесу спостерігається як випередження, так і запізнення формування полів концентрації згідно з неklasичними моделями у порівнянні з класичною моделлю, проте на більш пізніх стадіях спостерігається стійке запізнювання при описі процесу згідно з моделлю, досліджуваною у п. 2.2,

порівняно як з класичною моделлю, так і з релаксаційною моделлю з похідними цілого порядку. У порівнянні з дробово-диференціальною моделлю без врахування релаксаційних процесів, модель, досліджувана у п. 2.2, демонструє випередження фронту концентрацій на початкових стадіях, тоді як з плином часу результати розрахунків за обидвома моделями практично збігаються.

У п. 2.3 побудована дробово-диференціальна модель конвективної дифузії у неізотермічних умовах з урахуванням явища термодифузії. Поставлена відповідна цій моделі крайова задача моделювання конвективної дифузії забруднень згідно зі схемою їх міграції з річок, каналів, або накопичувачів промислових стоків. Розроблена методика чисельного розв'язання поставленої крайової задачі. Отримані результати показують, що дробово-диференціальна динаміка неізотермічного процесу є менш інтенсивною у порівнянні з відповідною динамікою при її моделюванні в рамках класичних математичних моделей.

У п. 2.4 сформульована нова дробово-диференціальна математична модель конвективної дифузії на основі k -похідної Капуто. Поставлена крайова задача, знайдено її чисельний розв'язок та отримано результати чисельних розрахунків. Розглядувана у п. 2.4 модель містить додатковий параметр у порівнянні з моделлю на основі стандартної похідної Капуто, що збільшує можливості її адаптації до фактичних умов перебігу процесу. Результати обчислювальних експериментів показують, що, у порівнянні з моделлю на основі стандартної похідної Капуто, запізнювання просування фронту концентрації є тим більшим, чим менше значення приймає параметр k .

Результати розділу 2 опубліковано в роботах [50, 53, 59, 64, 269, 277].

Джерела, що використані у розділі 2

Для написання даного розділу було використано 39 джерел [1], [15], [50], [53], [59], [62–64], [77], [78], [85], [107], [115], [139], [166], [177], [179], [192], [194], [248], [259], [262], [265–267], [269], [270], [277], [279], [281], [298], [299], [302–304], [311], [313], [317], [318], посилання на які зазначені в тексті розділу.

Розділ 3. Математичне моделювання нелокальних процесів фільтраційної консолідації ґрунтових середовищ на основі дробово-диференціальних моделей

Дослідження в області розробки ефективних і надійних методів моделювання динамічних процесів деформування та ущільнення водонасичених ґрунтових середовищ, зокрема основ гідротехнічних споруд, обумовлені актуальністю питань визначення умов безпечного функціонування великої кількості інженерних об'єктів, що забруднюють ґрунти та ґрунтові води. Вивчення особливостей фільтраційно-консолідаційної динаміки геопористих середовищ нерідко зводиться до розв'язання крайових задач для відповідних систем диференціальних рівнянь в частинних похідних або систем інтегро-диференціальних рівнянь [80, 132, 256, 279, 281, 289, 292, 319]. Для дослідження особливостей розвитку аномальних консолідаційних процесів у середовищах, для яких характерні ефекти пам'яті та просторових кореляцій, розроблений ряд ефективних моделей у дробово-диференціальній постановці [60, 70, 67]. Даний розділ присвячено розвитку відповідного апарату математичного моделювання.

3.1. Чисельне моделювання дробово-диференціальної фільтраційно-консолідаційної динаміки в рамках моделі з несингулярним ядром

3.1.1. Математична модель та початково-крайова задача

У ряді робіт, зокрема в [80, 132, 256, 279, 281], були побудовані та досліджені математичні моделі фільтраційної консолідації пористих ґрунтових середовищ, насичених сольовими розчинами, на основі наступного узагальнення фільтраційного закону Дарсі на випадок урахування явища хімічного осмосу:

$$u_x = -k \frac{\partial H}{\partial x} + \nu \frac{\partial C}{\partial x}, \quad (3.1)$$

де u_x – швидкість фільтрації, $H = p/\gamma$ – надлишковий напір, p – поровий тиск, γ – питома вага рідини, k – коефіцієнт фільтрації, C – концентрація солей в рідкій фазі, ν – коефіцієнт хімічного осмосу [281]. Найпростіша з таких моделей базується на системі рівнянь наведених, зокрема, в [281]

$$\frac{\partial H}{\partial t} = C_v \frac{\partial^2 H}{\partial x^2} - \mu \frac{\partial^2 C}{\partial x^2}, \quad (3.2)$$

$$\sigma \frac{\partial C}{\partial t} = d \frac{\partial^2 C}{\partial x^2} - u_x \frac{\partial C}{\partial x}, \quad (3.3)$$

де d – коефіцієнт конвективної дифузії, σ – пористість середовища, C_v – коефіцієнт консолидації [289, 292], $\mu = \nu C_v / k$, u_x – швидкість фільтрації сольового розчину, що визначається відповідно до (3.1).

У складних гірничо-геологічних умовах перебігу фільтраційно-консолидаційних процесів значний вплив на їх динаміку можуть мати локально-нерівноважні ефекти, зокрема ефекти пам'яті і просторових кореляцій, врахування яких в межах класичного походу є досить складним і неефективним. Ефективним тут є підхід пов'язаний з використанням апарату інтегро-диференціювання нецілого порядку [115, 139, 177]. Так, наприклад, в роботі [61] побудована та досліджена математична модель, що описує дробово-диференціальну консолидаційну динаміку насиченого сольовим розчином ґрунтового середовища, та базується на наступній системі рівнянь з дробовими похідними Капуто–Герасимова:

$$D_t^{(\alpha)} H = C_v \frac{\partial^2 H}{\partial x^2} - \mu \frac{\partial^2 C}{\partial x^2}, \quad (3.4)$$

$$\sigma D_t^{(\alpha)} C = d \frac{\partial^2 C}{\partial x^2} - u_x \frac{\partial C}{\partial x}, \quad (3.5)$$

де $D_t^{(\alpha)}$ – оператор дробової похідної Капуто–Герасимова [115, 139, 177] порядку α ($0 < \alpha \leq 1$). Ряд інших моделей, що описують дробово-диференціальну консолидаційну динаміку насичених сольовими розчинами пористих середовищ представлений у [60, 68]. При цьому, наявність сингулярних ядер у рівняннях моделей з дробовими похідними Капуто–Герасимова певним чином ускладнює як фізичну інтерпретацію одержуваних розв'язків, так і призводить до ускладнення алгоритмів комп'ютерного моделювання. Підхід, запропонований в [16, 17], згідно з яким моделювання фільтраційної динаміки у пористих середовищах виконується в рамках дробово-диференціальних математичних моделей з несингулярним ядром, зокрема з використанням поняття похідної Капуто–Фабріціо [75], до певної міри

позбавлений вищеописаних недоліків. Застосуємо зазначений похід при побудові дробово-диференціальних математичних моделей консолідаційної динаміки насичених сольовими розчинами ґрунтових середовищ.

Ґрунтуючись на результатах роботи [75], попередньо введемо до розгляду двопараметричну похідну Капуто–Фабріціо від функції $H(x,t)$ за змінною t наступним чином:

$${}^{CF}D_t^{\alpha,\lambda}H(x,t) = \frac{1}{\lambda(1-\alpha)} \int_0^t \frac{\partial H}{\partial \tau} e^{-\frac{\alpha}{1-\alpha} \frac{t-\tau}{\lambda}} d\tau, \quad (3.6)$$

де параметр α – порядок похідної ($0 < \alpha \leq 1$), λ – дійсний позитивний параметр. При $\lambda \rightarrow 1$ звідси отримуємо загальноприйняте визначення похідною Капуто–Фабріціо, запропоноване в [75].

Будемо виходити з наступного узагальнення рівняння нерозривності фільтраційного потоку:

$$\frac{\partial u_x}{\partial x} + \frac{k}{C_v} {}^{CF}D_t^{\alpha,\lambda}H(x,t) = 0, \quad (3.7)$$

де ${}^{CF}D_t^{\alpha,\lambda}$ – оператор дробової похідної Капуто–Фабріціо виду (3.6).

Слід зазначити, що, зокрема, при $\lambda = 1, \alpha \rightarrow 1$ зі співвідношення (3.7) отримуємо класичне рівняння нерозривності виду [292]:

$$\frac{\partial u_x}{\partial x} + \frac{k}{C_v} \frac{\partial H}{\partial t} = 0.$$

З (3.7) з урахуванням (3.1) маємо наступне рівняння для напору [67, 269, 276]:

$${}^{CF}D_t^{\alpha,\lambda}H(x,t) = C_v \frac{\partial^2 H}{\partial x^2} - \mu \frac{\partial^2 C}{\partial x^2} \quad (0 < \alpha \leq 1). \quad (3.8)$$

Аналогічно, з урахуванням відповідного дробово-диференціального узагальнення закону Фіка, можна отримати рівняння для концентрації у вигляді [67, 269, 276]

$$\sigma {}^{CF}D_t^{\alpha_*,\lambda}C(x,t) = d \frac{\partial^2 C}{\partial x^2} - u_x \frac{\partial C}{\partial x} \quad (0 < \alpha_* \leq 1). \quad (3.9)$$

Система рівнянь (3.8), (3.9) складає основу нової дробово-диференціальної моделі фільтраційної консолідації насичених сольовими розчинами пористих

середовищ, яка базується на рівняннях що не містять сингулярних ядер. Відзначимо, що при $\alpha, \alpha_* \rightarrow 1$, $\lambda = 1$ з (3.8), (3.9) отримуємо систему рівнянь (3.2), (3.3) класичної моделі фільтраційної консолідації [80, 132, 256, 279, 281].

В рамках моделі (3.8), (3.9), моделювання дробово-диференціальної динаміки нерівноважного в часі фільтраційно-консолідаційного процесу в насиченому сольовим розчином ґрунтовому середовищі у масиві скінченної потужності l з проникними межами, зводиться до розв'язання в області $\Omega = \{(x, t) : 0 < x < l, t > 0\}$ системи рівнянь (3.8), (3.9) при крайових умовах

$$H(0, t) = 0, \quad H(l, t) = 0, \quad H(x, 0) = H_0, \quad (3.10)$$

$$C(0, t) = C_0, \quad C(l, t) = 0, \quad C(x, 0) = 0, \quad (3.11)$$

де H_0 – початкове значення надлишкового напору в масиві, C_0 – задане значення концентрації солей на вході фільтраційного потоку.

Введемо до розгляду безрозмірні змінні і параметри співвідношеннями:

$$\begin{aligned} x' &= \frac{x}{l}, \quad t' = \frac{t}{\lambda}, \quad H' = \frac{H}{H_0}, \quad C' = \frac{C}{C_0}, \quad C'_v = \frac{\lambda C_v}{l^2}, \\ \mu' &= \frac{\lambda \mu C_0}{l^2 H_0}, \quad d' = \frac{d \lambda}{l^2}, \quad u' = \frac{k \lambda H_0}{l^2}, \quad v' = \frac{\lambda \nu C_0}{l^2}. \end{aligned} \quad (3.12)$$

Переходячи в (3.8)–(3.11) до нових змінних згідно з (3.12) і опускаючи надалі знак "штрих" над безрозмірними величинами, отримуємо в області $(0, 1) \times (0, \infty)$ крайову задачу

$${}^{CF}D_t^\alpha H(x, t) = C_v \frac{\partial^2 H}{\partial x^2} - \mu \frac{\partial^2 C}{\partial x^2} \quad (0 < \alpha \leq 1), \quad (3.13)$$

$$\sigma {}^{CF}D_t^{\alpha_*} C(x, t) = d \frac{\partial^2 C}{\partial x^2} + \left(u \frac{\partial H}{\partial x} - v \frac{\partial C}{\partial x} \right) \frac{\partial C}{\partial x} \quad (0 < \alpha_* \leq 1), \quad (3.14)$$

$$H(0, t) = 0, \quad H(1, t) = 0, \quad H(x, 0) = 1, \quad (3.15)$$

$$C(0, t) = 1, \quad C(1, t) = 0, \quad C(x, 0) = 0, \quad (3.16)$$

де ${}^{CF}D_t^\alpha u(t) = \frac{1}{1-\alpha} \int_0^t u'(\tau) e^{-\frac{\alpha}{1-\alpha}(t-\tau)} d\tau$ – дробова похідна Капуто–Фабріціо [75]

порядку α .

3.1.2. Чисельне розв'язання крайової задачі

Введемо до розгляду сіткову область

$$\omega_{h\tau} = \left\{ (x_i, t_j) : x_i = ih \quad (i = \overline{0, m+1}), t_j = j\tau \quad (j = \overline{0, n}) \right\},$$

де h, τ – кроки сітки за геометричною змінною і часом, відповідно. На основі монотонної різницевої схеми О. А. Самарського [313] поставимо у відповідність задачі (3.13)–(3.16) систему різницевих рівнянь, яка записується в стандартних позначеннях у вигляді

$$\Delta_t^{(\alpha)} H = C_v \hat{H}_{\bar{x}\bar{x}} - \mu \hat{C}_{\bar{x}\bar{x}} \quad (0 < \alpha \leq 1), \quad (3.17)$$

$$\sigma \Delta_t^{(\alpha_*)} C = \chi \hat{C}_{\bar{x}\bar{x}} + v^+ \hat{C}_x + v^- \hat{C}_{\bar{x}} \quad (0 < \alpha_* \leq 1), \quad (3.18)$$

де

$$\chi = \frac{d}{R}, \quad R = 1 + \frac{h|v|}{2d}, \quad v^\pm = \frac{1}{2}(v \pm |v|), \quad v = uH_x - vC_x. \quad (3.19)$$

При цьому в співвідношеннях (3.17), (3.18) оператор $\Delta_t^{(\alpha)}$ позначає дискретний аналог дробової похідної Капуто–Фабріціо порядку α , який визначається як

$$\Delta_t^{(\alpha)} u = c_\alpha \sum_{k=0}^j \omega_k^{(j)} u_{t,k}, \quad (3.20)$$

де

$$u_{t,k} = \frac{u^{k+1} - u^k}{\tau}, \quad c_\alpha = \frac{1}{\alpha} (1 - e^{-b_\alpha \tau}), \quad \omega_k^{(j)} = e^{-b_\alpha (j-k)\tau}, \quad b_\alpha = \frac{\alpha}{1-\alpha}.$$

Розписуючи в (3.17), (3.18) відповідні різницеві оператори з урахуванням (3.20) і зводячи подібні члени, редукуємо задачу до розв'язання наступних СЛАР [67, 276]:

$$AH_{i-1}^{j+1} - SH_i^{j+1} + AH_{i+1}^{j+1} = F_i^j \quad (i = \overline{1, m}; j = \overline{0, n}), \quad (3.21)$$

$$\tilde{A}_i^j C_{i-1}^{j+1} - \tilde{S}_i^j C_i^{j+1} + \tilde{B}_i^j C_{i+1}^{j+1} = \tilde{F}_i^j \quad (i = \overline{1, m}; j = \overline{0, n}), \quad (3.22)$$

де

$$A = \frac{C_v}{c_\alpha h^2}, \quad S = \frac{1}{\tau} + 2A, \quad F_i^j = \sum_{k=0}^{j-1} \omega_k^{(j)} \frac{H_i^{k+1} - H_i^k}{\tau} +$$

$$\begin{aligned}
& + \frac{\mu}{c_\alpha h^2} (C_{i-1}^{j+1} - 2C_i^{j+1} + C_{i+1}^{j+1}) - \frac{H_i^j}{\tau}, \quad \tilde{A}_i^j = \frac{1}{h} \left(\frac{\chi_i^j}{h} - (\nu^-)_i^j \right), \\
& \tilde{B}_i^j = \frac{1}{h} \left(\frac{\chi_i^j}{h} + (\nu^+)_i^j \right), \quad \tilde{S}_i^j = \frac{\sigma c_\alpha^*}{\tau} + \tilde{A}_i^j + \tilde{B}_i^j, \\
& \tilde{F}_i^j = \sigma c_\alpha^* \left(\sum_{k=0}^{j-1} \tilde{\omega}_k^{(j)} \frac{C_i^{k+1} - C_i^k}{\tau} - \frac{C_i^j}{\tau} \right), \quad c_\alpha^* = \frac{1}{\alpha_*} \left(1 - e^{-b_\alpha^* \tau} \right), \\
& b_\alpha^* = \frac{\alpha_*}{1 - \alpha_*}, \quad \tilde{\omega}_k^{(j)} = e^{-b_\alpha^* (j-k)\tau}.
\end{aligned} \tag{3.23}$$

При $j = 0$ суми у (3.23) вважаються рівними нулю.

Різницеві рівняння систем (3.21), (3.22) триточкові і ефективно розв'язуються методом прогонки [313].

При цьому прогоночні співвідношення мають вигляд

$$H_i^{j+1} = \alpha_{i+1}^j H_{i+1}^{j+1} + \beta_{i+1}^j, \quad C_i^{j+1} = \tilde{\alpha}_{i+1}^j C_{i+1}^{j+1} + \tilde{\beta}_{i+1}^j \quad (i = \overline{1, m}; \quad j = \overline{0, n}),$$

а прогоночні коефіцієнти обчислюються за формулами

$$\begin{aligned}
\alpha_{i+1}^j &= \frac{A}{S - A\alpha_i^j}, \quad \beta_{i+1}^j = \alpha_{i+1}^j \left(\beta_i^j - \frac{F_i^j}{A} \right) \quad (i = \overline{1, m}; \quad j = \overline{0, n}), \\
\tilde{\alpha}_{i+1}^j &= \frac{\tilde{B}_i^j}{\tilde{S}_i^j - \tilde{A}_i^j \tilde{\alpha}_i^j}, \quad \tilde{\beta}_{i+1}^j = \frac{\tilde{\alpha}_{i+1}^j}{\tilde{B}_i^j} \left(\tilde{A}_i^j \tilde{\beta}_i^j - \tilde{F}_i^j \right) \quad (i = \overline{1, m}; \quad j = \overline{0, n}).
\end{aligned}$$

Для визначення стартових значень прогоночних коефіцієнтів використовуємо скінченно-різницеві аналоги граничних умов отримуючи

$$\alpha_1^j, \tilde{\alpha}_1^j = 0, \quad \beta_1^j = 0, \quad \tilde{\beta}_1^j = 1, \quad (j = \overline{0, n}).$$

Стійкість методу прогонки для систем (3.21), (3.22) впливає з факту діагонального переважання в матрицях коефіцієнтів цих СЛАР.

3.1.3. Результати обчислювальних експериментів

Чисельне моделювання динаміки полів надлишкових фільтраційних напорів і полів концентрацій в рамках розглядуваної у п. 3.1 математичної моделі виконано щодо безрозмірних змінних, які визначаються співвідношеннями (3.12) для вхідних даних з роботи [281]. Вихідні коди розробленого програмного забезпечення

наведено у [265]. Деякі з отриманих при цьому результатів графічно зображені на рис. 3.1–3.6.

Аналіз результатів численних експериментів дозволяє зробити наступні висновки [67, 276] про особливості динаміки полів фільтраційних тисків і концентрацій в процесі консолідації масивів насичених сольовими розчинами:

1. В рамках розглядуваної математичної моделі з плином часу спостерігаються явища розсіювання початкового надлишкового порового тиску (рис. 3.1, 3.3, 3.5) а також зростання концентрації солей у поровому розчині (рис. 3.2, 3.4, 3.6), що адекватно відображають відому фізичну картину процесу консолідації насичених сольовими розчинами пористих середовищ [279, 281].

2. Зменшення значень порядків дробових похідних α, α_* призводить до прискорення процесу розсіювання полів надлишкових напорів в масиві (рис. 3.3, 3.5) і до прискореного зростання концентрації сольового розчину в точках фільтраційної області (рис. 3.4, 3.6) в порівнянні з описом динаміки процесу в рамках класичних математичних моделей вивчених в [80, 132, 256, 279, 281] (криві 1 на рис. 3.1–3.6).

Прискорення динаміки модельованих процесів в рамках даної неklasичної математичної моделі фільтраційної консолідації свідчить про те, що вона описує процеси у супердифузійному наближенні.

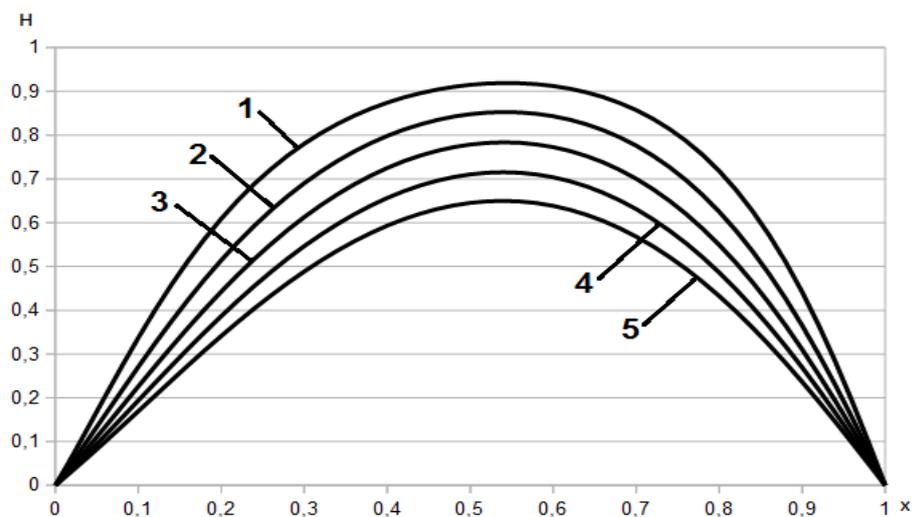


Рисунок 3.1 – Графіки функції напорів при $\alpha_* = \alpha = 0,85$ для різних значень

безрозмірною тимчасової змінної t

($1-t=0,1$; $2-t=0,3$; $3-t=0,5$; $4-t=0,6$; $5-t=0,7$).

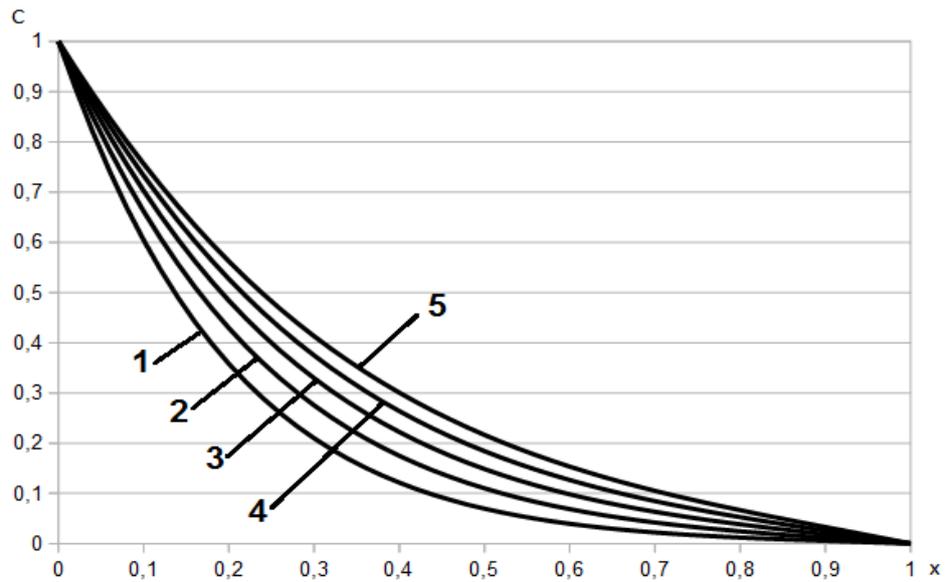


Рисунок 3.2 – Графіки функції концентрації при $\alpha_* = \alpha = 0,85$ для різних значень безрозмірною тимчасової змінної t ($1-t=0,1$; $2-t=0,3$; $3-t=0,5$; $4-t=0,6$; $5-t=0,7$).

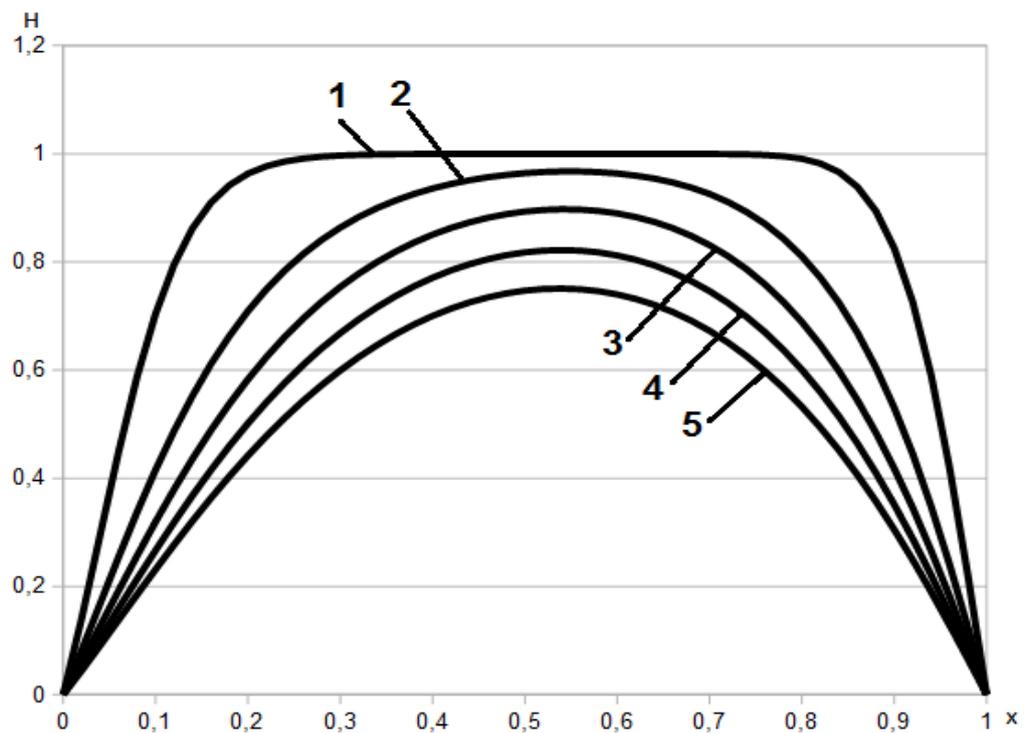


Рисунок 3.3 – Графіки функції напорів у фіксований момент часу $t = 0,05$ для різних порядків дробової похідної ($1-\alpha = 1$; $2-\alpha = 0,9$; $3-\alpha = 0,8$; $4-\alpha = 0,7$; $5-\alpha = 0,6$; $\alpha_* = \alpha$)

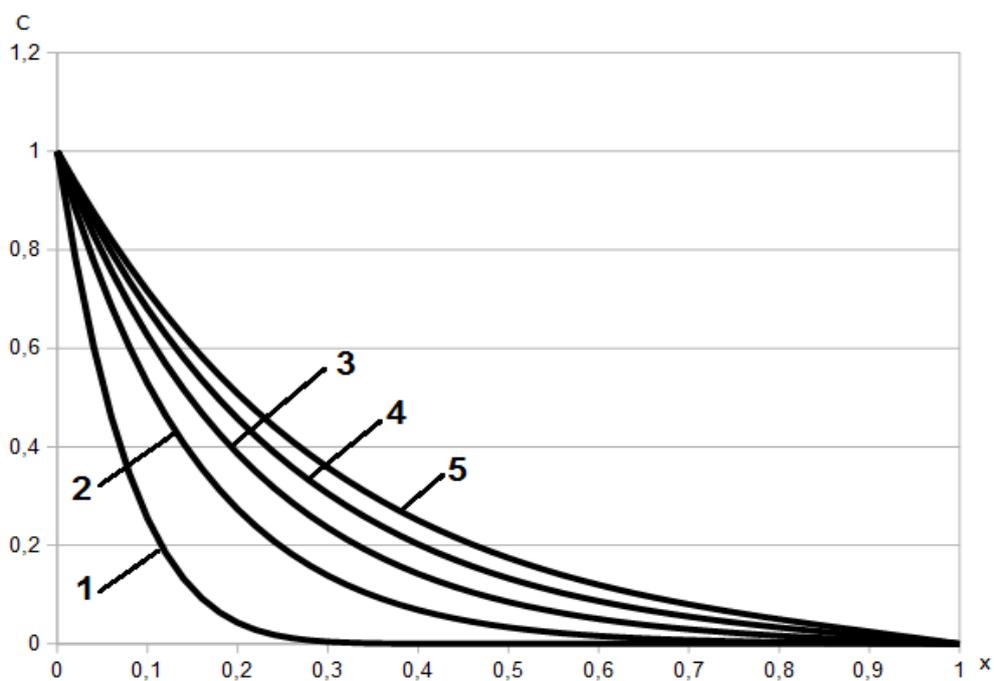


Рисунок 3.4 – Графіки функції концентрації у фіксований момент часу $t = 0,05$
для різних порядків дробової похідної

($1 - \alpha = 1$; $2 - \alpha = 0,9$; $3 - \alpha = 0,8$; $4 - \alpha = 0,7$; $5 - \alpha = 0,6$; $\alpha_* = \alpha$)

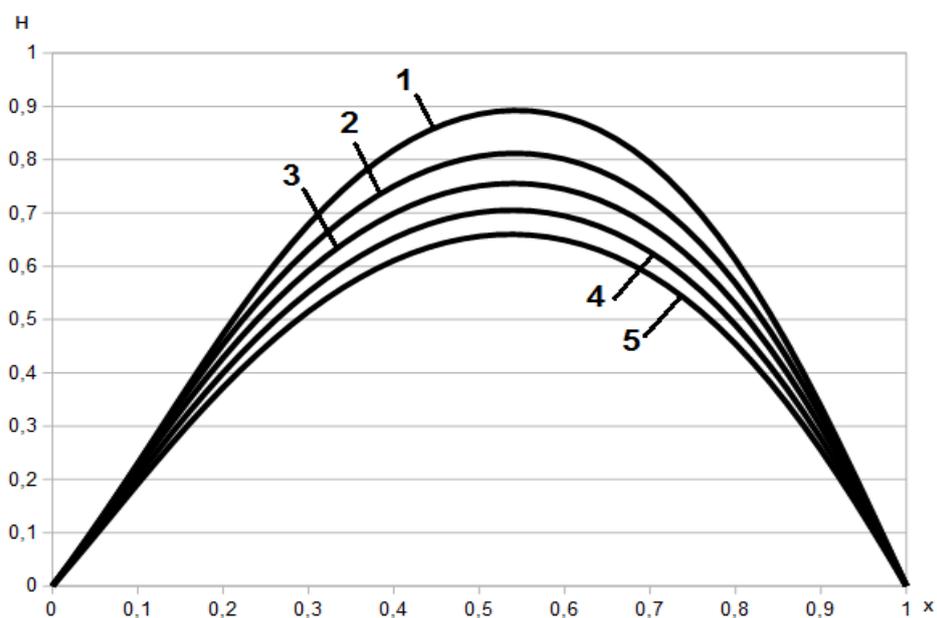


Рисунок 3.5 – Графіки функції напорів у фіксований момент часу $t = 0,5$ для
різних порядків дробової похідної

($1 - \alpha = 1$; $2 - \alpha = 0,9$; $3 - \alpha = 0,8$; $4 - \alpha = 0,7$; $5 - \alpha = 0,6$; $\alpha_* = \alpha$)

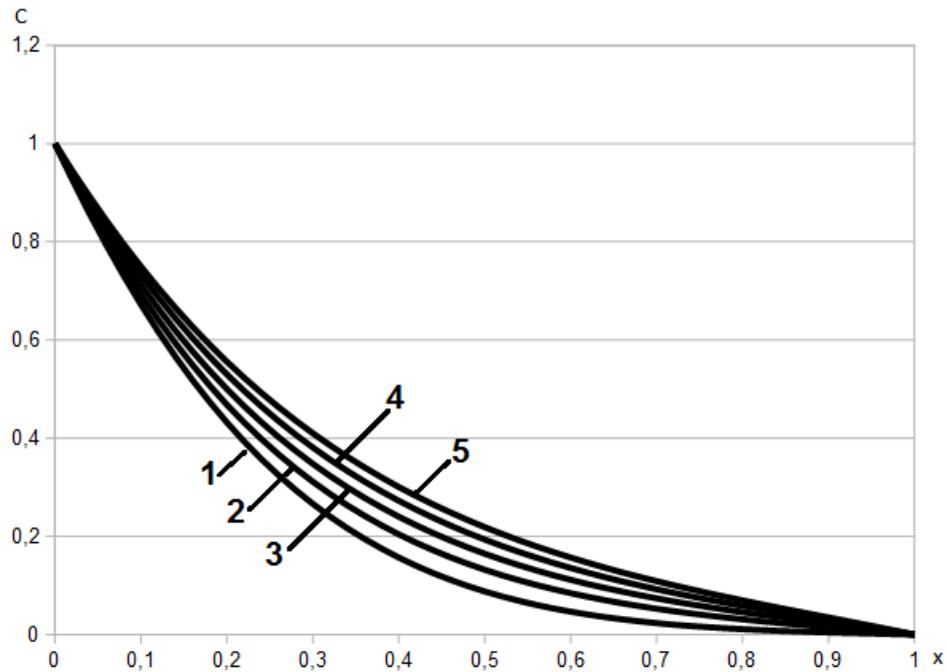


Рисунок 3.6 – Графіки функції концентрації у фіксований момент часу $t = 0,5$ для різних порядків дробової похідної
 $(1 - \alpha = 1; 2 - \alpha = 0,9; 3 - \alpha = 0,8; 4 - \alpha = 0,7; 5 - \alpha = 0,6; \alpha_* = \alpha)$

3.2. Моделювання дробово-диференціальної фільтраційно-консолідаційної динаміки з урахуванням фактора повзучості

3.2.1. Математична модель та початково-крайова задача

Поширимо на випадок врахування явища повзучості ґрунтового скелета підхід до моделювання динаміки фільтраційно-консолідаційних процесів в насичених сольовими розчинами ґрунтових середовищах описаний у п. 3.1.

Як відомо, скелет ґрунтового масиву може мати властивості повзучості, тобто може деформуватися з часом при постійному навантаженні [289, 292]. Відповідно до теорії спадкової лінійної повзучості зміну коефіцієнта пористості в часі можна представити у вигляді [263, 319]

$$e(t) = e(0) + \sigma(0)\delta(t,0) - \int_0^t \frac{\partial \sigma}{\partial \tau} \delta(t,\tau) d\tau, \quad (3.24)$$

де $e(t)$ – коефіцієнт пористості ґрунту, $\sigma(t)$ – сума головних напружень в скелеті, $\delta(t,\tau)$ – величина зміни коефіцієнта пористості до моменту часу t від одиничного навантаження, прикладеного в момент часу τ . Згідно з [263, 289, 292, 319] маємо

$$\delta(t, \tau) = a_0 + a_1 [1 - e^{-\gamma_1(t-\tau)}], \quad (3.25)$$

де a_0 – параметр миттєвої деформації, a_1, γ_1 – параметри повзучості, що визначаються експериментальним шляхом, зокрема γ_1 – характеризує швидкість наростання деформацій повзучості [292, 319].

В одновимірному випадку ущільнення шару ґрунту рівномірно розподіленим навантаженням інтенсивності $q = const$ у припущеннях основної розрахункової моделі В. А. Флоріна [319] маємо $\sigma = q - p = q - \gamma H$. Тоді, з (3.24) з урахуванням (3.25) отримуємо

$$\frac{\partial e}{\partial t} = a_0 \frac{\partial p}{\partial t} + a_1 \gamma_1 \int_0^t \frac{\partial p}{\partial \tau} e^{-\gamma_1(t-\tau)} d\tau. \quad (3.26)$$

Згідно з [292, 319] основне рівняння консолідації трифазного середовища записується у вигляді

$$\frac{\partial e}{\partial t} + \beta(1 + \bar{e}) \frac{\partial p}{\partial t} - (1 + \bar{e}) \frac{\partial u_x}{\partial x} = 0, \quad (3.27)$$

де β – коефіцієнт об'ємної стисливості газової компоненти, \bar{e} – середнє значення коефіцієнта пористості, u_x – швидкість фільтрації.

Таким чином, з (3.27) з урахуванням (3.26) маємо [292, 319]

$$\frac{\partial H}{\partial t} + \frac{a_1 \gamma_1}{a_0 + \beta(1 + \bar{e})} \int_0^t \frac{\partial H}{\partial \tau} e^{-\gamma_1(t-\tau)} d\tau + \frac{C_v}{k} \frac{\partial u_x}{\partial x} = 0, \quad (3.28)$$

де $C_v = \frac{(1 + \bar{e})k}{\gamma[a_0 + \beta(1 + \bar{e})]}$ – коефіцієнт консолідації.

Враховуючи у (3.28) співвідношення для швидкості фільтрації (3.1) остаточно отримуємо

$$\frac{\partial H}{\partial t} + \frac{a_1 \gamma_1}{a_0 + \beta(1 + \bar{e})} \int_0^t \frac{\partial H}{\partial \tau} e^{-\gamma_1(t-\tau)} d\tau = C_v \frac{\partial^2 H}{\partial x^2} - \mu \frac{\partial^2 C}{\partial x^2}, \quad (3.29)$$

де $\mu = \nu C_v / k$.

Інтегро-диференціальне рівняння (3.29) є рівнянням для напору у дробово-диференціальній моделі консолідації ґрунтових середовищ з урахуванням осмотичних явищ та фактора повзучості скелета. Аналогічно випадку насиченості

грунтового масиву лише чистою водою без урахування осмотичних явищ, рівняння (3.29) може легко бути перетворено в диференціальне, проте вже вищого порядку за часовою змінною [292, 319]. При цьому виникають проблеми отримання та фізичної інтерпретації додаткової початкової умови, необхідної для коректної постановки крайових задач, які не є тривіальними.

У зв'язку з цим для моделювання динаміки розглядуваних процесів надалі використовується безпосередньо рівняння (3.29).

Вводячи позначення

$$\gamma_1 = \frac{\alpha}{(1-\alpha)\lambda}, \quad \zeta_\alpha = \frac{a_1\alpha}{a_0 + \beta(1+\bar{e})} \quad (0 < \alpha < 1, \lambda > 0),$$

перепишемо (3.29) у вигляді

$$\frac{\partial H}{\partial t} + \zeta_\alpha {}^{CF}D_t^{\alpha,\lambda} H(x,t) = C_v \frac{\partial^2 H}{\partial x^2} - \mu \frac{\partial^2 C}{\partial x^2}, \quad (3.30)$$

де ${}^{CF}D_t^{\alpha,\lambda}$ – оператор двопараметричної похідною Капуто–Фабріціо [75], що визначається згідно з (3.6).

Зберігаючи рівняння для концентрації сольового розчину в вигляді (3.9), отримуємо систему рівнянь дробово-диференціальної математичної моделі фільтраційної консолідації насичених сольовими розчинами ґрунтових середовищ з урахуванням повзучості скелета, що складається з рівнянь (3.30), (3.9). В рамках цієї моделі, моделювання консолідаційної динаміки масиву скінченної потужності l з проникними межами зводиться до розв'язання в області Ω системи рівнянь (3.30), (3.9) з крайовими умовами (3.10), (3.11). У безрозмірних змінних (3.12), відповідна крайова задача запишеться у вигляді [67, 276] (знак "штрих" над безрозмірними величинами опущений)

$$\frac{\partial H}{\partial t} + \zeta_\alpha {}^{CF}D_t^\alpha H(x,t) = C_v \frac{\partial^2 H}{\partial x^2} - \mu \frac{\partial^2 C}{\partial x^2} \quad (0 < \alpha \leq 1), \quad (3.31)$$

$$\sigma {}^{CF}D_t^{\alpha_*} C(x,t) = d \frac{\partial^2 C}{\partial x^2} + \left(u \frac{\partial H}{\partial x} - v \frac{\partial C}{\partial x} \right) \frac{\partial C}{\partial x} \quad (0 < \alpha_* \leq 1), \quad (3.32)$$

$$H(0,t) = 0, \quad H(1,t) = 0, \quad H(x,0) = 1, \quad (3.33)$$

$$C(0,t) = 1, \quad C(1,t) = 0, \quad C(x,0) = 0. \quad (3.34)$$

3.2.2. Чисельне розв'язання крайової задачі

Вводячи до розгляду сіткову область ω_{ht} аналогічно описаному у пп. 3.1.2, на основі монотонної різницевої схеми О. А. Самарського [313], поставимо у відповідність крайовій задачі (3.31)–(3.34) систему різницевих рівнянь

$$H_t + \zeta_\alpha \Delta_t^{(\alpha)} H = C_v \hat{H}_{\bar{x}\bar{x}} - \mu \hat{C}_{\bar{x}\bar{x}} \quad (0 < \alpha \leq 1), \quad (3.35)$$

$$\sigma \Delta_t^{(\alpha_*)} C = \chi \hat{C}_{\bar{x}\bar{x}} + \nu^+ \hat{C}_x + \nu^- \hat{C}_{\bar{x}} \quad (0 < \alpha_* \leq 1), \quad (3.36)$$

де збережені позначення (3.19), (3.20).

Розписуючи у (3.35), (3.36) різницеві оператори та зводячи подібні члени, редукуємо задачу до розв'язання СЛАР виду (3.21), (3.22), де елементи матриць коефіцієнтів систем і векторів вільних членів мають вигляд (3.23) за винятком величин S, F_i^j , які в даному випадку дорівнюють

$$F_i^j = \zeta_\alpha \sum_{k=0}^{j-1} \omega_k^{(j)} \frac{H_i^{k+1} - H_i^k}{\tau} + \frac{\mu}{c_\alpha h^2} (C_{i-1}^{j+1} - 2C_i^{j+1} + C_{i+1}^{j+1}) - \frac{(1 + \zeta_\alpha) H_i^j}{\tau},$$

$$S = \frac{1 + \zeta_\alpha}{\tau} + 2A, \quad \zeta_\alpha = \alpha a_1 / [a_0 + \beta(1 + \bar{e})].$$

Подальша процедура отримання розв'язків СЛАР збігається з викладеним у пп. 3.1.2.

3.2.3. Результати обчислювальних експериментів

Чисельне моделювання динаміки полів надлишкових напорів та концентрацій в рамках моделі з урахуванням фактора повзучості виконано для вхідних даних з роботи [281] щодо безрозмірних змінних (3.12). Вихідні коди розробленого програмного забезпечення наведено у [265]. Деякі з отриманих при цьому результатів представлені на рис. 3.7–3.9.

Аналіз результатів чисельних експериментів дозволяє зробити наступні висновки [67, 276]:

1. Врахування фактора лінійної повзучості ґрунтового скелета призводить до прискореного розсіювання полів надлишкових напорів в порівнянні з випадком відсутності його врахування (рис. 3.7). Стосовно полів концентрацій сольового розчину, в розрахунках спостерігалися незначні відхилення $C_1 - C_2$ величин

концентрацій, знайдених в рамках моделей з та без урахування повзучості скелета, причому абсолютна величина різниці $C_1 - C_2$ та її знак залежать від координати відповідної точки у фізичній площині та від моменту часу t (рис. 3.8);

2. Як і у разі неврахування (криві 1–3 на рис. 3.9) так і у разі врахування (криві 1'–3' на рис. 3.9) фактора повзучості, зі зменшенням величин порядків дробових похідних спостерігається прискорення процесу розсіювання полів надлишкових напорів. Таким чином, математична модель консолідаційних процесів, що містить дробову похідну Капуто–Фабріціо та враховує фактор повзучості ґрунтового скелета також володіє чітко вираженими ознаками моделі з супердифузійними властивостями.

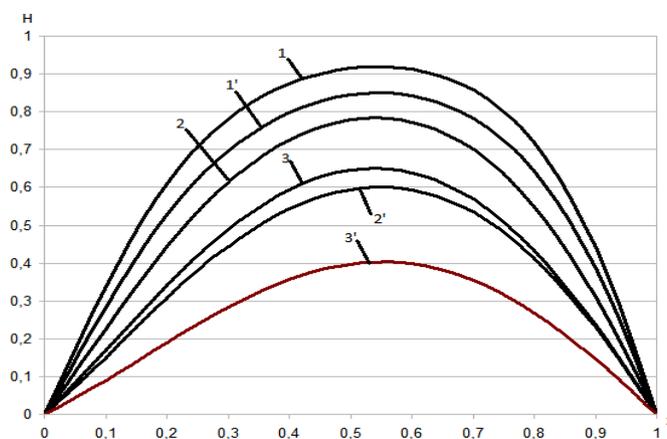


Рисунок 3.7 – Криві функції напорів для моделі без (криві 1–3) та з урахуванням фактора повзучості (криві 1'–3') при $\alpha = \alpha_* = 0,85$ та

$$1, 1' - t = 0,1; 2, 2' - t = 0,5; 3, 3' - t = 0,9$$

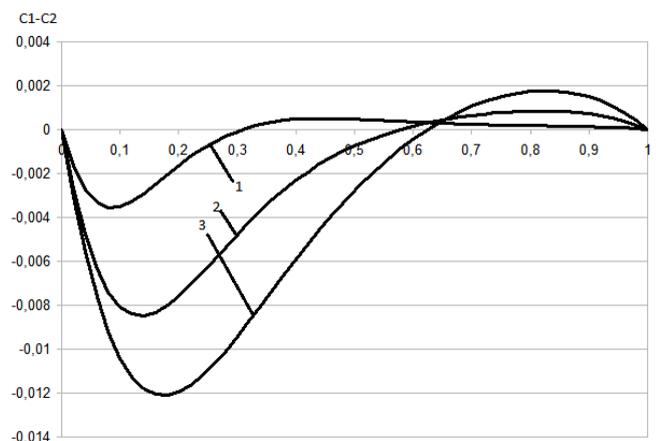


Рисунок 3.8 – Різниця $C_1 - C_2$ концентрацій в моделях без та з урахування повзучості скелета для $1, 1' - t = 0,1; 2, 2' - t = 0,5; 3, 3' - t = 0,9; \alpha = \alpha_* = 0,85$

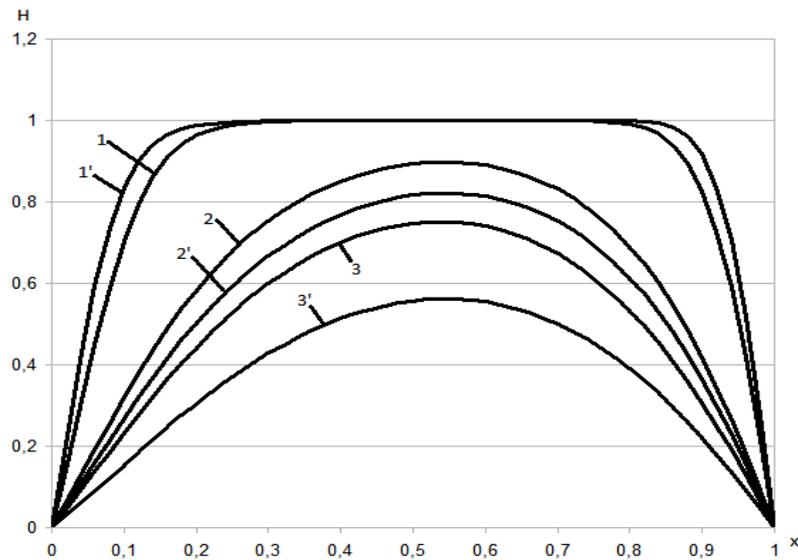


Рисунок 3.9 – Криві функції напорів в моделі без (криві 1–3) та з урахуванням повзучості скелета (криві 1'–3') у момент часу $t = 0,05$ для

$$1,1' - \alpha = 1; 2,2' - \alpha = 0,8; 3,3' - \alpha = 0,6; \alpha_* = \alpha$$

3.3. Математичне моделювання фільтраційно-консолідаційної динаміки в умовах солеперенесення на основі дробово-фрактальної моделі

3.3.1. Математична модель та початково-крайова задача

Розглядаючи ізотермічний нелокальний в часі фільтраційно-консолідаційний процес у насиченому сольовим розчином ґрунтовому середовищі фрактальної структури будемо виходити з наступних узагальнень законів Дарсі та Фіка [52]:

$$u_x = D_t^{1-\beta} \frac{\partial}{\partial x^\alpha} (-kH + \nu C), \quad (3.37)$$

$$q_c = D_t^{1-\beta} \left(-d_* \frac{\partial C}{\partial x^\alpha} + C J_t^{1-\beta} u_x \right), \quad (3.38)$$

де u_x – швидкість фільтрації, $H(x,t) = p / \gamma$ – надлишковий напір, p – поровий тиск, γ – питома вага рідини, $C(x,t)$ – концентрація солей в рідкій фазі, k – коефіцієнт фільтрації, ν – коефіцієнт хімічного осмосу [281], q_c – дифузійний потік, d_* – коефіцієнт конвективної дифузії, $J_t^{1-\beta}$ – дробовий інтеграл Рімана–Ліувілля порядку $1-\beta$, $0 < \beta \leq 1$, $D_t^{1-\beta}$ – оператор дробового диференціювання Рімана–

Ліувілля того ж порядку за змінною t [115, 139, 177, 195], $\frac{\partial}{\partial x^\alpha}$ – оператор фрактальної похідної у розімінні Хаусдорфа [10, 71, 82], $\alpha > 0$ – фрактальна розмірність.

З урахуванням (3.37) та узагальненого рівняння нерозривності фільтраційного потоку в умовах лінійного закону ущільнення

$$\frac{\partial u_x}{\partial x^\alpha} + \frac{k}{C_v} \frac{\partial H}{\partial t} = 0,$$

отримуємо рівняння [52]

$$D_t^{(\beta)} H = \frac{\partial}{\partial x^\alpha} \left(\frac{\partial}{\partial x^\alpha} (C_v H - \mu C) \right), \quad (3.39)$$

де C_v – коефіцієнт консолідації [289, 319], $\mu = \frac{\nu C_v}{k}$, $D_t^{(\beta)}$ – оператор регуляризованої дробової похідної Капуто–Герасимова порядку β за змінною t [115, 139, 177, 195].

З узагальненого співвідношення балансу солей рідкій фазі для ґрунтового середовища фрактальної структури, з урахуванням (3.38) можна отримати рівняння для визначення концентрації солей в потоці ґрунтових вод у вигляді [52]

$$\sigma D_t^{(\beta)} C = d_* \frac{\partial}{\partial x^\alpha} \left(\frac{\partial C}{\partial x^\alpha} \right) + \frac{\partial}{\partial x^\alpha} (kH - \nu C) \cdot \frac{\partial C}{\partial x^\alpha}, \quad (3.40)$$

де σ – пористість середовища.

Зі співвідношень (3.39), (3.40) при $\alpha \rightarrow 1$ отримуємо систему рівнянь для дробово-диференціальної моделі фільтраційної консолідації насиченого сольовим розчином ґрунтового середовища без урахування фрактальних властивостей [61], а при $\alpha, \beta \rightarrow 1$ – систему рівнянь моделі фільтраційної консолідації у класичній постановці [279, 281]:

$$\begin{aligned} \frac{\partial H}{\partial t} &= \frac{\partial^2}{\partial x^2} (C_v H - \mu C), \\ \sigma \frac{\partial C}{\partial t} &= d_* \frac{\partial^2 C}{\partial x^2} + \frac{\partial}{\partial x} (kH - \nu C) \cdot \frac{\partial C}{\partial x}. \end{aligned}$$

Розписуючи у (3.39), (3.40) оператори фрактальної похідної згідно з [10, 71, 82] і зводячи подібні члени, отримуємо систему рівнянь моделі у вигляді [52]

$$D_t^{(\beta)} H = C_v \left[s_\alpha(x) \frac{\partial^2 H}{\partial x^2} + r_\alpha(x) \frac{\partial H}{\partial x} \right] - \mu \left[s_\alpha(x) \frac{\partial^2 C}{\partial x^2} + r_\alpha(x) \frac{\partial C}{\partial x} \right], \quad (3.41)$$

$$\sigma D_t^{(\beta)} C = d_* \left[s_\alpha(x) \frac{\partial^2 C}{\partial x^2} + r_\alpha(x) \frac{\partial C}{\partial x} \right] + s_\alpha(x) \frac{\partial v}{\partial x} \cdot \frac{\partial C}{\partial x}, \quad (3.42)$$

$$v(x,t) = kH(x,t) - vC(x,t), \quad r_\alpha(x) = \frac{1-\alpha}{\alpha^2} x^{1-2\alpha}, \quad s_\alpha(x) = \frac{1}{\alpha^2} x^{2(1-\alpha)}.$$

В рамках моделі (3.41), (3.42), дробово-диференціальна динаміка нелокального в часі фільтраційно-консолідаційного процесу в насиченому сольовим розчином ґрунтовому середовищі фрактальної структури у випадку масиву скінченної потужності l з проникними межами описується в області $\Omega = \{(x,t): 0 < x < l, t > 0\}$ системою рівнянь (3.41), (3.42) при крайових умовах

$$H(0,t) = 0, \quad H(l,t) = 0, \quad H(x,0) = H_0, \quad (3.43)$$

$$C(0,t) = C_0, \quad C(l,t) = 0, \quad C(x,0) = 0, \quad (3.44)$$

де H_0 – початкове значення надлишкового напору в масиві, C_0 – задане значення концентрації солей на вході фільтраційного потоку.

3.3.2. Чисельне розв’язання крайової задачі

Вводячи до розгляду сіткову область ω_{hr} аналогічно описаному у пп. 3.1.2, поставимо у відповідність задачі (3.41)–(3.44) лінеаризовану різницеву схему Кранка–Ніколсон, яка записується в позначеннях роботи [313] у вигляді

$$\sigma \Delta_t^{(\beta)} C = 0.5d_* \left[s_\alpha \left(\hat{C}_{\bar{x}x} + C_{\bar{x}x} \right) + r_\alpha \left(\hat{C}_x + C_x \right) \right] + 0.5s_\alpha v_x \left(\hat{C}_x + C_x \right), \quad (3.45)$$

$$\begin{aligned} \Delta_t^{(\beta)} H = 0.5C_v \left[s_\alpha \left(\hat{H}_{\bar{x}x} + H_{\bar{x}x} \right) + r_\alpha \left(\hat{H}_x + H_x \right) \right] - \\ - 0.5\mu \left[s_\alpha \left(\hat{C}_{\bar{x}x} + C_{\bar{x}x} \right) + r_\alpha \left(\hat{C}_x + C_x \right) \right], \end{aligned} \quad (3.46)$$

де оператор $\Delta_t^{(\beta)} u$ позначає дискретний аналог дробової похідної Капуто–Герасимова $D_t^{(\beta)} u$ та визначається як

$$\Delta_{t_{j+1}}^{(\beta)} u \approx \frac{u^{j+1} - u^j}{\tau^\beta \Gamma(2 - \beta)} + \sum_{v=0}^{j-1} \omega_v^{(j)} \frac{u^{v+1} - u^v}{\tau}, \quad (3.47)$$

$$\omega_v^{(j)} = \frac{\tau^{1-\beta}}{\Gamma(2 - \beta)} \left[(j - v + 1)^{1-\beta} - (j - v)^{1-\beta} \right],$$

де $\Gamma(z)$ – гамма-функція Ейлера [259].

Розписуючи в (3.45), (3.46) різницеві оператори з урахуванням (3.47) і зводячи подібні члени, редукуємо розглядувану задачу на $(j+1)$ -ому часовому шарі до розв'язання наступних СЛАР [52]:

$$A_i^j C_{i-1}^{j+1} - S_i^j C_i^{j+1} + B_i^j C_{i+1}^{j+1} = F_i^j \quad (i = \overline{1, m}; \quad j = \overline{0, n}), \quad (3.48)$$

$$\tilde{A}_i^j H_{i-1}^{j+1} - \tilde{S}_i^j H_i^{j+1} + \tilde{B}_i^j H_{i+1}^{j+1} = \tilde{F}_i^j \quad (i = \overline{1, m}; \quad j = \overline{0, n}), \quad (3.49)$$

$$C_0^{j+1} = C_0, \quad C_{m+1}^{j+1} = 0, \quad C_i^0 = 0 \quad (i = \overline{0, m+1}; \quad j = \overline{0, n}), \quad (3.50)$$

$$H_0^{j+1} = 0, \quad H_{m+1}^{j+1} = 0, \quad H_i^0 = H_0 \quad (i = \overline{0, m+1}; \quad j = \overline{0, n}), \quad (3.51)$$

де

$$A_i^j = \frac{0.5}{h} \left[d_* \left(\frac{s_\alpha^i}{h} - \frac{r_\alpha^i}{2} \right) - \frac{s_\alpha^i}{4h} (\nu_{i+1}^j - \nu_{i-1}^j) \right],$$

$$B_i^j = \frac{0.5}{h} \left[d_* \left(\frac{s_\alpha^i}{h} + \frac{r_\alpha^i}{2} \right) + \frac{s_\alpha^i}{4h} (\nu_{i+1}^j - \nu_{i-1}^j) \right],$$

$$S_i^j = A_i^j + B_i^j + \frac{\sigma}{\tau^\beta \Gamma(2 - \beta)},$$

$$F_i^j = \sigma \left[\sum_{v=0}^{j-1} \omega_v^{(j)} \frac{C_i^{v+1} - C_i^v}{\tau} - \frac{C_i^j}{\tau^\beta \Gamma(2 - \beta)} \right] -$$

$$- \frac{0.5 d_*}{h} \left[\frac{s_\alpha^i}{h} (C_{i-1}^j - 2C_i^j + C_{i+1}^j) + \frac{r_\alpha^i}{2} (C_{i+1}^j - C_{i-1}^j) \right] - \frac{0.5 s_\alpha^i}{4h^2} (\nu_{i+1}^j - \nu_{i-1}^j) (C_{i+1}^j - C_{i-1}^j),$$

$$\tilde{A}_i^j = \frac{0.5 C_\nu}{h} \left(\frac{s_\alpha^i}{h} - \frac{r_\alpha^i}{2} \right), \quad \tilde{B}_i^j = \frac{0.5 C_\nu}{h} \left(\frac{s_\alpha^i}{h} + \frac{r_\alpha^i}{2} \right), \quad \tilde{S}_i^j = \tilde{A}_i^j + \tilde{B}_i^j + \frac{1}{\tau^\beta \Gamma(2 - \beta)},$$

$$\begin{aligned} \tilde{F}_i^j = & \sum_{v=0}^{j-1} \omega_v^{(j)} \frac{H_i^{v+1} - H_i^v}{\tau} - \frac{H_i^j}{\tau^\beta \Gamma(2-\beta)} - \\ & - \frac{0.5C_v}{h} \left[\frac{s_\alpha^i}{h} (H_{i-1}^j - 2H_i^j + H_{i+1}^j) + \frac{r_\alpha^i}{2} (H_{i+1}^j - H_{i-1}^j) \right] + \\ & + \frac{0.5\mu}{h} \left[\frac{s_\alpha^i}{h} (C_{i-1}^{j+1} - 2C_i^{j+1} + C_{i+1}^{j+1} + C_{i-1}^j - 2C_i^j + C_{i+1}^j) + \frac{r_\alpha^i}{2} (C_{i+1}^{j+1} - C_{i-1}^{j+1} + C_{i+1}^j - C_{i-1}^j) \right], \\ & s_\alpha^i = s_\alpha(x_i), \quad r_\alpha^i = r_\alpha(x_i), \quad v_i^j = kH_i^j - vC_i^j. \end{aligned}$$

При $j=0$ суми в F_i^j, \tilde{F}_i^j вважаються рівними нулю.

Різницеві рівняння (3.48), (3.49) є триточковими та ефективно розв'язуються методом прогонки [313]. Прогоночні співвідношення мають вигляд

$$C_i^{j+1} = \tilde{\xi}_{i+1}^j C_{i+1}^{j+1} + \tilde{\zeta}_{i+1}^j, \quad H_i^{j+1} = \xi_{i+1}^j H_{i+1}^{j+1} + \zeta_{i+1}^j, \quad (i = \overline{1, m}; \quad j = \overline{0, n}),$$

а прогоночні коефіцієнти обчислюються за формулами

$$\begin{aligned} \tilde{\xi}_{i+1}^j &= \frac{B_i^j}{S_i^j - A_i^j \tilde{\xi}_i^j}, \quad \tilde{\zeta}_{i+1}^j = \frac{\tilde{\xi}_{i+1}^j}{B_i^j} (A_i^j \tilde{\xi}_i^j - F_i^j) \quad (i = \overline{1, m}; \quad j = \overline{0, n}), \\ \xi_{i+1}^j &= \frac{\tilde{B}_i^j}{\tilde{S}_i^j - \tilde{A}_i^j \tilde{\xi}_i^j}, \quad \zeta_{i+1}^j = \frac{\xi_{i+1}^j}{\tilde{B}_i^j} (\tilde{A}_i^j \tilde{\xi}_i^j - \tilde{F}_i^j) \quad (i = \overline{1, m}; \quad j = \overline{0, n}). \end{aligned}$$

Для визначення стартових значень прогоночних коефіцієнтів зі скінченно-різницевого аналогу граничних умов (3.50), (3.51) отримуємо

$$\tilde{\xi}_1^j = 0, \quad \tilde{\zeta}_1^j = C_0, \quad \xi_1^j = 0, \quad \zeta_1^j = 0 \quad (j = \overline{0, n}).$$

Стійкість методу прогонки для систем (3.48), (3.49) впливає з факту діагонального переважання в матрицях коефіцієнтів цих СЛАР.

3.3.3. Результати обчислювальних експериментів

Чисельне моделювання динаміки полів надлишкових фільтраційних напорів та полів концентрацій згідно з моделлю, описаною у пп. 3.3.1, було виконано для вхідних даних з роботи [281]. Вихідні коди розробленого програмного забезпечення наведено у [265]. Деякі з отриманих результатів розрахунків щодо змінних $x' = \frac{x}{x_0}$,

$t' = \frac{t}{t_0}$, $C' = \frac{C}{C_0}$, $H' = \frac{H}{H_0}$ представлені на рис. 3.10–3.12. Тут $C_0 = 200 \text{ г / л}$, $H_0 = 10 \text{ м}$, $x_0 = 25 \text{ м}$, $t_0 = 60 \text{ діб}$, а знак "штрих" надалі опущений.

Аналіз отриманих результатів дозволяє зробити наступні висновки [52]:

1. Загальна картина розподілу полів концентрацій та надлишкових напорів у геопористому масиві, що консолидується, отримана в рамках розглядуваної дробово-фрактальної моделі в цілому адекватна відповідній картині розподілу аналогічних полів, одержуваної в рамках як дробово-диференціальної моделі з урахуванням ефектів пам'яті, однак без урахування фрактальних властивостей [61, 70], так і класичної математичної моделі [281];

2. Зменшення значення параметра фрактальної розмірності α при $0 < \alpha < 1$ призводить до прискорення процесів як засолення так і розсіювання полів надлишкових напорів (рис. 3.11), тобто до скорочення часу фільтраційного ущільнення масиву у порівнянні з випадком опису процесу дробово-диференціальною моделлю, що враховує лише ефекти пам'яті [61];

3. Зі збільшенням значення параметра фрактальної розмірності α при $\alpha > 1$ процеси засолення і розсіювання полів надлишкових напорів (рис. 3.12) істотно сповільнюються в порівнянні з випадком моделювання цих процесів в рамках дробово-диференціальної моделі [61, 70], що свідчить про наявність субдифузійних властивостей у розглядуваній дробово-фрактальній моделі.

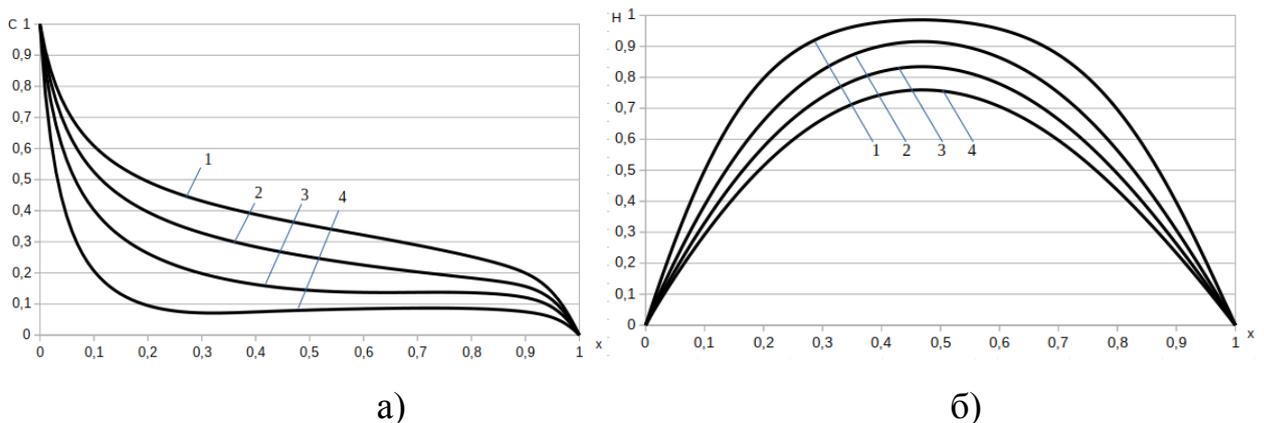


Рисунок 3.10 – Криві концентрацій C (а) та надлишкових напорів H (б) при $\alpha = 0,9$; $\beta = 0,8$ для 1 – $t = 0,33$; 2 – $t = 0,66$; 3 – $t = 1,0$; 4 – $t = 1,33$

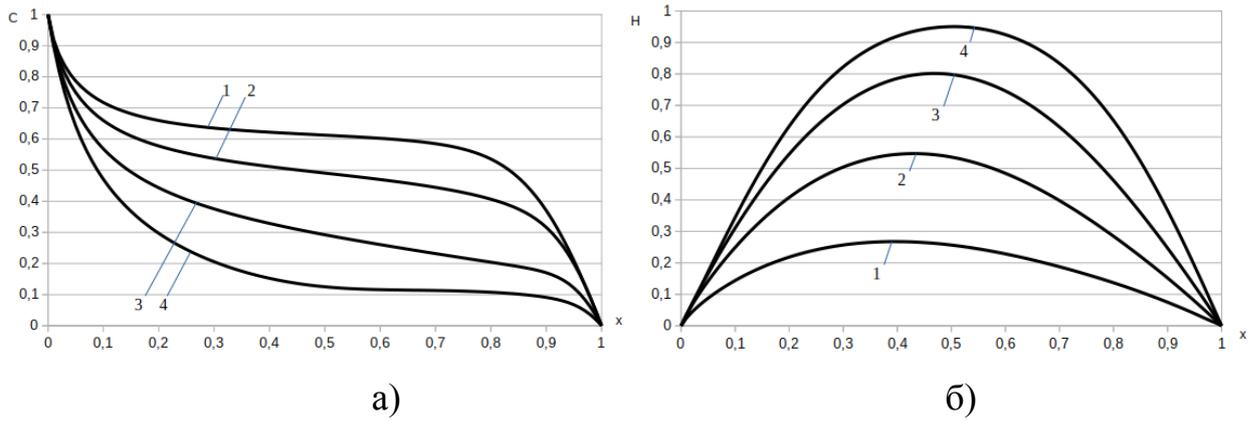


Рисунок 3.11 – Криві концентрацій C (а) та надлишкових напорів H (б) при $\beta = 0,85$; $t = 1,0$ для 1 – $\alpha = 0,7$; 2 – $\alpha = 0,8$; 3 – $\alpha = 0,9$; 4 – $\alpha = 1,0$

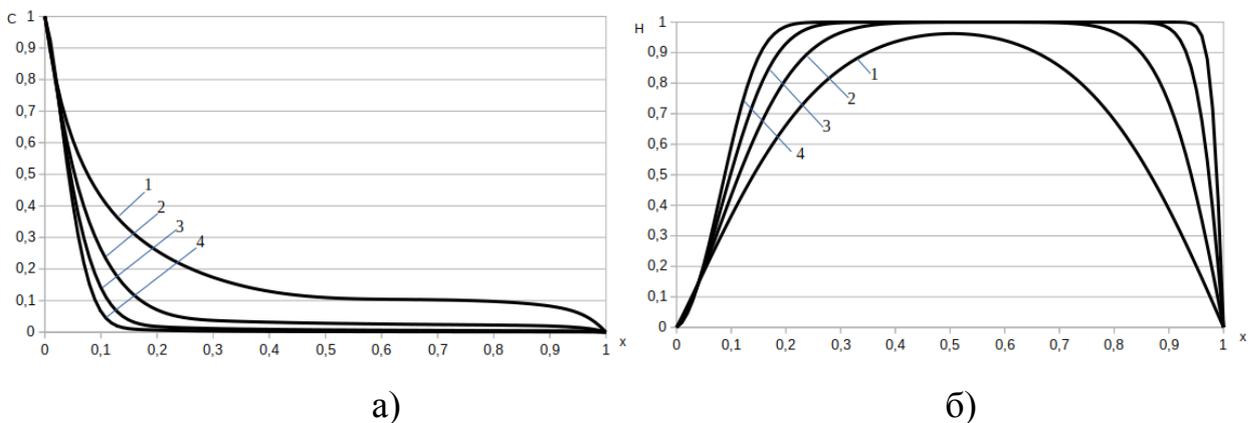


Рисунок 3.12 – Криві концентрацій C (а) та надлишкових напорів H (б) при $\beta = 0,8$; $t = 1,0$ для 1 – $\alpha = 1,0$; 2 – $\alpha = 1,2$; 3 – $\alpha = 1,4$; 4 – $\alpha = 1,6$

Отримані результати чисельного моделювання добре узгоджуються з наведеними у [10] результатами моделювання з використанням фрактального рівняння адвекції-дисперсії, які показали, що фрактальне формулювання рівняння адвекції-дисперсії описує як супердифузійні, так і субдифузійні процеси.

3.4. Висновки до розділу 3

У розділі 3 наводяться результати математичного моделювання процесів фільтраційної консолідації насичених сольовими розчинами ґрунтів у одновимірному наближенні на основі запропонованих автором разом зі співавторами [52, 67, 269, 276] нових дробово-диференціальних моделей з використанням методу скінченних різниць для отримання чисельних розв'язків поставлених нелінійних крайових задач.

У п. 3.1 сформульовано математичну модель та поставлено нелінійну початково-крайову задачу, відповідну моделюванню нелокальних процесів фільтраційної консолідації з урахуванням явища хімічного осмосу. Вказана модель містить похідні Капуто–Фабріціо з несингулярним ядром за часовою змінною як у рівнянні для поля фільтраційних напорів, так і для поля концентрації сольових розчинів. Результати проведених чисельних експериментів показали, що зменшення значень порядків дробових похідних у такій моделі призводить до прискорення процесу розсіювання полів напорів і до прискореного зростання концентрації сольового розчину у порівнянні з описом динаміки процесу в рамках класичних математичних моделей. Таким чином, можна стверджувати, що запропонована модель описує консолідаційні процеси у супердифузійному наближенні.

Розглянута у п. 3.1 математична модель консолідації насичених сольовими розчинами ґрунтових основ поширюється у п. 3.2 на випадок врахування явища лінійної повзучості скелету ґрунтового масиву. Запропонована нелінійна система дробово-диференціальних рівнянь відповідної математичної моделі, поставлена крайова задача та одержано її чисельний розв'язок. Результати чисельних експериментів засвідчують, що у порівнянні з випадком відсутності врахування явища повзучості, тут спостерігається прискорене розсіювання поля надлишкових напорів в процесі консолідації. Зазначимо, що аналогічно моделі, дослідженої у п. 3.1, модель з врахуванням явища лінійної повзучості описує супердифузійні процеси перенесення.

У п. 3.3 досліджується нова дробово-диференціальна математична модель фільтраційної консолідації насиченого сольовим розчином ґрунтового середовища з урахуванням його фрактальних властивостей, які моделюються рівняннями з фрактальними похідними у розумінні Хаусдорфа. Для запропонованої моделі поставлена крайова задача моделювання відповідного процесу у випадку масиву скінченної потужності з проникними межами та розроблена чисельна методика її розв'язання. Результати чисельних розрахунків згідно з моделлю демонструють, що, в залежності від значення коефіцієнту фрактальної розмірності середовища, описувані процеси можуть мати як субдифузійний, так і супердифузійний характер.

Результати розділу 3 опубліковано в роботах [52, 67, 269, 276].

Джерела, що використані у розділі 3

Для написання даного розділу було використано 29 джерел [10], [16], [17], [52], [60], [61], [67], [68], [70], [71], [75], [80], [82], [115], [132], [139], [177], [195], [256], [263], [265], [269], [276], [279], [281], [289], [292], [313], [319], посилання на які зазначені в тексті розділу.

Розділ 4. Математичне моделювання нелокальних процесів конвективно-дифузійного масоперенесення в ґрунтах з урахуванням масообміну

Відомо, що в ряді районів зрошуваного землеробства вельми важливою проблемою є запобігання засолення сільськогосподарських земель та розсолення земель, що вийшли з господарського обороту внаслідок засолення. Застосування дренажу в районах зрошення може привести до ряду негативних наслідків, зокрема, вимивання гумусу та поживних речовин. Ці явища погіршують стан ґрунтів і можуть мати негативний вплив, зокрема, на водопостачання. У зв'язку з цим вивчення водно-сольового режиму ґрунтів є актуальним об'єктом дослідження, зокрема, у меліорації, гідрогеології, ґрунтознавстві, прикладній математиці [123, 207, 208, 214, 221, 271, 274, 279, 280, 287, 299, 304, 310, 311, 312].

Даний розділ присвячений математичному моделюванню динаміки локально-нерівноважних процесів конвективної дифузії розчинних речовин з урахуванням впливу процесів масообміну. Такі задачі часто виникають при дослідженні вищеописаних процесів, наприклад, в питаннях розсолення і промивання ґрунтів, опріснення ґрунтових вод і їх очищення від забруднюючих речовин. Переважна більшість існуючих математичних моделей зазначених процесів побудована в рамках класичних підходів [123, 214, 221, 271, 274] без урахування нелокальності процесів перенесення за складних умов їх перебігу. У зв'язку з цим, підвищення ступеня адекватності класичних моделей процесів перенесення у системах зі складною просторово-часовою структурою є актуальною проблемою. Нижче побудовано і досліджено деякі некласичні математичні моделі процесів конвективно-дифузійного перенесення за нелокальних умов перебігу та з урахуванням кінетики масообміну.

4.1. Математичне моделювання дробово-диференціальної динаміки процесу промивки засолених ґрунтів у одновимірному наближенні

4.1.1. Математична модель та початково-крайова задача

Розглянемо процес розсолення ґрунту фрактальної структури промиванням у припущенні одновимірності фільтраційного і сольового потоків. У випадку, коли

процеси відбуваються у повністю насиченій зоні постійних розмірів, припустимо, що виконується модифікований закон фільтрації виду [318, 320]

$$v = -k_f D_x^{(\alpha)} H(x,t), \quad (4.1)$$

де k_f – фрактальний коефіцієнт фільтрації, $H(x,t)$ – напір, $D_x^{(\alpha)}$ – оператор дробової похідної Капуто–Герасимова за змінною x порядку α ($0 < \alpha \leq 1$) [115, 139, 177], v – швидкість фільтрації.

З урахуванням (4.1) рівняння неусталеної фільтрації має вигляд

$$\frac{\partial H}{\partial t} = a D_x^{(\alpha+1)} H(x,t), \quad (4.2)$$

де a – коефіцієнт п'єзопроводності [311].

Крайові умови для рівняння (4.2) запишемо у вигляді

$$H(0,t) = H_1, \quad H(l,t) = H_0, \quad H(x,0) = H_0, \quad (4.3)$$

де H_1 – задане значення напору на вході $x=0$ фільтраційного потоку, H_0 – початкове значення напору, l – розмір області фільтрації.

Введемо до розгляду безрозмірні змінні співвідношеннями

$$x' = \frac{x}{l}, \quad t' = \frac{a}{l^{\alpha+1}} t, \quad H' = \frac{H}{H_1}, \quad H'_0 = \frac{H_0}{H_1}. \quad (4.4)$$

У змінних (4.4) задача (4.2), (4.3) формулюється у вигляді (знак "штрих" над безрозмірними величинами надалі опускається) [34]

$$\frac{\partial H}{\partial t} = D_x^{(\alpha+1)} H(x,t) \quad (0 < \alpha \leq 1), \quad (4.5)$$

$$H(0,t) = 1, \quad H(1,t) = H_0, \quad H(x,0) = H_0. \quad (4.6)$$

Для отримання рівняння динаміки нерівноважного процесу солеперенесення запишемо, з урахуванням [85], рівняння релаксаційного закону Фіка у вигляді

$$q + \tau_q \frac{\partial q}{\partial t} = D_t^{1-\gamma} \left(-d D_x^{(\beta)} C + v C \right) \quad (0 < \gamma, \beta \leq 1), \quad (4.7)$$

де q – потік солеперенесення, C – концентрація солей у рідкій фазі, d – коефіцієнт конвективної дифузії, $D_t^{1-\gamma}$ – оператор дробової похідної Рімана–Ліувілля за

змінною t порядку $1-\gamma$, $D_x^{(\beta)}$ – оператор дробової похідної Капуто–Герасимова за змінною x [115, 139, 177], τ_q – параметр релаксації дифузійного потоку.

З рівняння балансу маси з урахуванням рівняння масообміну виду

$$\sigma \frac{\partial C}{\partial t} + \frac{\partial N}{\partial t} + \frac{\partial q}{\partial x} = 0$$

отримуємо наступне неklasичне рівняння солеперенесення:

$$\sigma_* (D_t^{(\gamma)} C + \tau_q D_t^{(\gamma+1)} C) = d D_x^{(1+\beta)} C - \frac{\partial}{\partial x} (\nu C) + \gamma_* (C_m - C), \quad (4.8)$$

де σ_* – пористість середовища, γ_* – константа швидкості розчинення, N – концентрація солей у твердій фазі, C_m – концентрація граничного насичення.

Відзначимо, що у рівнянні (4.8) постульовано наступне узагальнене рівняння кінетики розчинення солей:

$$D_t^{(\gamma)} N + \tau_q D_t^{(\gamma+1)} N = -\gamma_* (C_m - C).$$

Звідси при $\gamma \rightarrow 1$ і $\tau_q = 0$ отримуємо стандартне рівняння кінетики розчинення солей твердої фази

$$\frac{\partial N}{\partial t} = -\gamma_* (C_m - C).$$

Якщо параметр релаксації $\tau_q = 0$, то конвективно-дифузійне рівняння (4.8) запишеться у вигляді

$$\sigma_* D_t^{(\gamma)} C = d D_x^{(1+\beta)} C - \frac{\partial}{\partial x} (\nu C) + \gamma_* (C_m - C). \quad (4.9)$$

Звідси при $\gamma \rightarrow 1$, $\beta \rightarrow 1$ отримуємо класичне рівняння, що описує процес конвективної дифузії в умовах масообміну.

Застосуємо рівняння (4.9) для моделювання динаміки нерівноважного процесу промивання засолених ґрунтів. При цьому використовуються крайові умови

$$C(0,t) = 0, \quad C(l,t) = C_0, \quad C(x,0) = C_0, \quad (4.10)$$

де C_0 – задане початкове значення концентрації солей.

Вводячи до розгляду безрозмірні змінні та параметри рівностями (4.4) та доповнюючи їх співвідношеннями

$$C'_0 = \frac{C}{C_0}, \quad C'_m = \frac{C_m}{C_0}, \quad d' = \frac{l^{\gamma(\alpha+1)-\beta-1}}{a^\gamma} d, \quad (4.11)$$

отримуємо наступну задачу для визначення поля концентрації солей (знак "штрих" надалі опускається) [34, 269]:

$$\sigma_* D_t^{(\gamma)} C = d D_x^{(1+\beta)} C - \frac{\partial}{\partial x} (vC) + \tilde{\gamma}_* (C_m - C) \quad (0 < \gamma, \beta \leq 1), \quad (4.12)$$

$$C(0, t) = 0, \quad C(1, t) = 1, \quad C(x, 0) = 1, \quad (4.13)$$

де

$$v = -\omega D_x^{(\alpha)} H, \quad \omega = \frac{k_f H_1}{a^\gamma l^{(1-\gamma)(\alpha+1)}}, \quad \tilde{\gamma} = \gamma \frac{l^{\gamma(\alpha+1)}}{a^\gamma}.$$

4.1.2. Чисельне розв'язання крайової задачі

Розглянемо методику побудови наближеного розв'язку крайової задачі (4.5), (4.12), (4.6), (4.13).

Введемо до розгляду сіткову область

$$\omega_{ht} = \left\{ (x_i, t_j) : x_i = ih_1 \left(i = \overline{0, m} \right), t_j = j\tau \left(j = \overline{0, 1, 2, \dots} \right) \right\},$$

де h_1, τ – кроки сітки за геометричною змінною та часом, відповідно, і запишемо аналогічно, наприклад, [248], наступні апроксимації похідних за часовою та геометричною змінними:

$$D_t^{(\gamma)} u = \Delta_t^{(\gamma)} u + O(\tau), \quad \Delta_t^{(\gamma)} u = \frac{1}{\Gamma(2-\gamma)} \sum_{s=0}^j b_s^{(j)} \frac{u^{s+1} - u^s}{\tau},$$

$$b_s^{(j)} = \tau^{1-\gamma} \left[(j-s+1)^{1-\gamma} - (j-s)^{1-\gamma} \right] \quad (0 < \gamma \leq 1),$$

$$D_x^{(1+\beta)} u = \delta_\beta (u_{\bar{x}\bar{x}} + \bar{D}_x^\beta u) + O(h_1), \quad \delta_\beta = \frac{h_1^{1-\beta}}{\Gamma(2-\beta)},$$

$$D_x^{(1+\alpha)} u = \delta_\alpha (u_{\bar{x}\bar{x}} + \bar{D}_x^\alpha u) + O(h_1), \quad \delta_\alpha = \frac{h_1^{1-\alpha}}{\Gamma(2-\alpha)}, \quad (4.14)$$

$$\bar{D}_x^\beta u = \frac{1}{h_1^2} \sum_{r=1}^{i-1} \tilde{g}_r (u_{i-r+1} - 2u_{i-r} + u_{i-r-1}), \quad \tilde{g}_r = (r+1)^{1-\beta} - r^{1-\beta} \quad (0 < \beta \leq 1),$$

$$\bar{D}_x^\alpha u = \frac{1}{h_1^2} \sum_{r=1}^{i-1} g_r (u_{i-r+1} - 2u_{i-r} + u_{i-r-1}), \quad g_r = (r+1)^{1-\alpha} - r^{1-\alpha} \quad (0 < \alpha \leq 1),$$

$\Gamma(\alpha)$ – гамма-функція, $\sum_{r=1}^{i-1}(\cdot) = 0$ при $i = 1$.

З урахуванням зазначених апроксимацій поставимо у відповідність задачі (4.5), (4.12), (4.6), (4.13) наступну лінеаризовану різницеву схему, яка записується в стандартних позначеннях [192, 313] у вигляді

$$H_t = \delta_\alpha (H_{\bar{x}\bar{x}} + \bar{D}_x^\alpha H), \quad (4.15)$$

$$\sigma_* A_t^{(\gamma)} C = d\delta_\beta (\hat{C}_{\bar{x}\bar{x}} + \bar{D}_x^\beta C) - (\hat{v}^- \hat{C})_x - (\hat{v}^+ \hat{C})_{\bar{x}} + \tilde{\gamma}_* (C_m - \hat{C}), \quad (4.16)$$

де

$$v^+(x) = \frac{v(x) + |v(x)|}{2}, \quad v^-(x) = \frac{v(x) - |v(x)|}{2}.$$

Розписуючи різницеві оператори в (4.15), (4.16) і зводячи подібні члени, отримуємо наступні СЛАР [34]:

$$H_{i+1}^{j+1} - S_1 H_i^{j+1} + H_{i-1}^{j+1} = \Phi_i^j \quad (i = \overline{1, m-1}; j = 0, 1, 2, \dots), \quad (4.17)$$

$$A_i^j C_{i+1}^{j+1} - R_i^j C_i^{j+1} + B_i^j C_{i-1}^{j+1} = \Psi_i^j \quad (i = \overline{1, m-1}; j = 0, 1, 2, \dots), \quad (4.18)$$

де

$$S_1 = 2 + \frac{h_1^2}{\tau \delta_\alpha}, \quad \Phi_i^j = -h_1^2 \left(\frac{1}{\tau \delta_\alpha} H_i^j + \bar{D}_x^\alpha H_i^j \right), \quad A_i^j = \frac{1}{h_1} \left(\frac{d\delta_\beta}{h_1} - v_{i+1}^{-j+1} \right),$$

$$B_i^j = \frac{1}{h_1} \left(\frac{d\delta_\beta}{h_1} + v_{i+1}^{+j+1} \right), \quad R_i^j = \frac{\sigma_*}{\tau^\gamma \Gamma(2-\gamma)} + \tilde{\gamma}_* + \frac{2d\delta_\beta}{h_1^2} + \frac{1}{h_1} (v_i^{+j+1} - v_i^{-j+1}),$$

$$\Psi_i^j = -\frac{\sigma_*}{\tau^\gamma \Gamma(2-\gamma)} C_i^j - \tilde{\gamma}_* C_m - d\delta_\beta \bar{D}_x^\beta C_i^j + \frac{\sigma_*}{\Gamma(2-\gamma)} \sum_{s=0}^{j-1} b_s^{(j)} \frac{C_i^{s+1} - C_i^s}{\tau}.$$

Різницеві рівняння систем (4.17), (4.18) є триточковими та ефективно розв'язуються методом прогонки [192, 313]. Стійкість методу прогонки обумовлена діагональним переважанням в матрицях коефіцієнтів розглянутих СЛАР [313]. Для отримання розв'язку задачі на одному часовому шарі необхідно провести послідовне розв'язання спочатку системи (4.17) для визначення поля напорів, а потім системи (4.18) для визначення поля концентрацій.

4.1.3. Результати обчислювальних експериментів

Чисельне моделювання динаміки процесу промивання засолених ґрунтів в рамках розглядуваної неklasичної математичної моделі виконано щодо безрозмірних змінних (4.4), (4.11) для вхідних даних, наведених у роботі [312]. Вихідні коди розробленого програмного забезпечення наведено у [265]. Деякі з отриманих при цьому результатів графічно зображені на рис. 4.1–4.4.

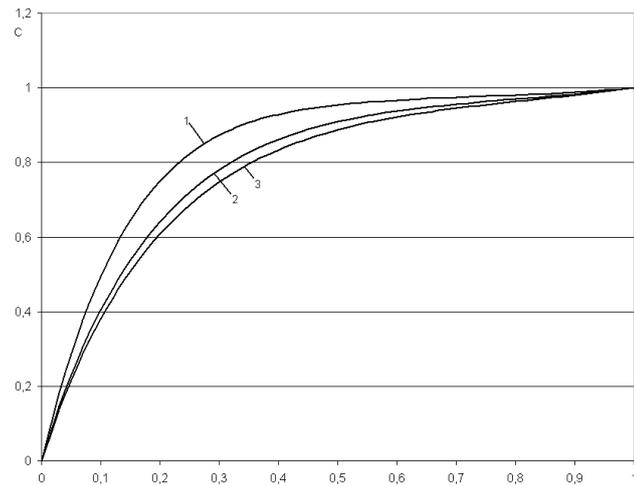


Рисунок 4.1 – Криві концентрації у рідкій фазі при відсутності масообміну для фіксованих значень порядків дробових похідних $\alpha = 0,8$; $\beta = 0,9$; $\gamma = 0,8$ та

$$1 - t = 10, 2 - t = 30, 3 - t = 50$$

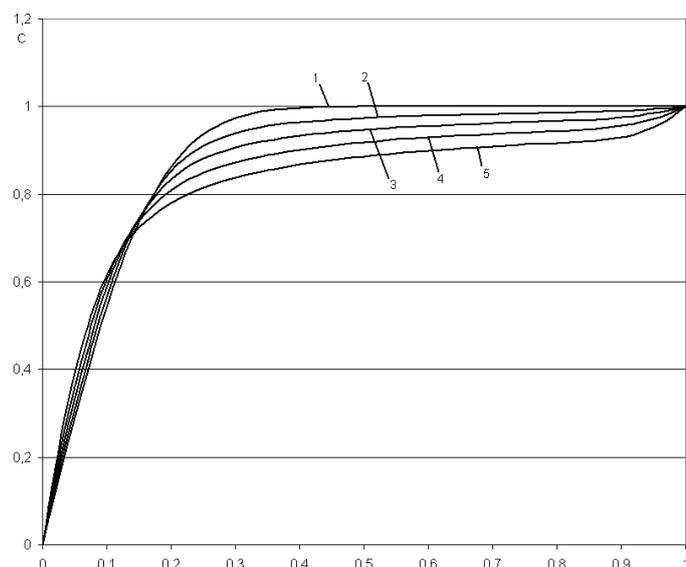


Рисунок 4.2 – Криві концентрації у момент часу $t = 3$ в залежності від зміни величини порядку β дробової похідної: $1 - \beta = 1$ ($\alpha = 1, \gamma = 1$); $2 - \beta = 0,9$;

$$3 - \beta = 0,8; 4 - \beta = 0,7; 5 - \beta = 0,6$$

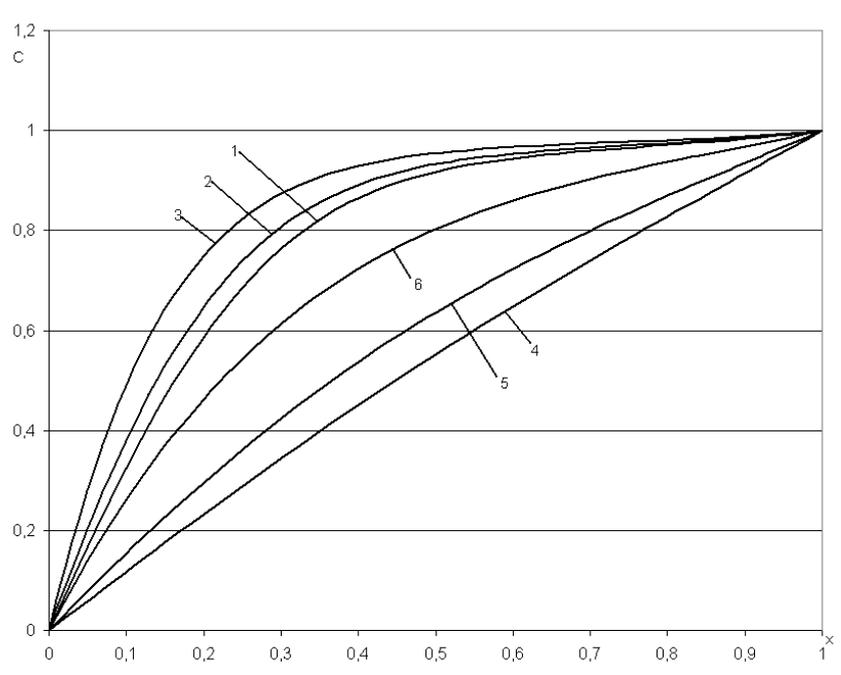


Рисунок 4.3 – Криві концентрації у рідкій фазі в моменти часу (1, 2, 3 – $t = 10$; 4, 5, 6 – $t = 80$) для $\alpha = 0,8$; $\beta = 0,9$ в залежності від зміни величини порядку γ : 1, 4 – $\gamma = 1$; 2, 5 – $\gamma = 0,9$; 3, 6 – $\gamma = 0,7$

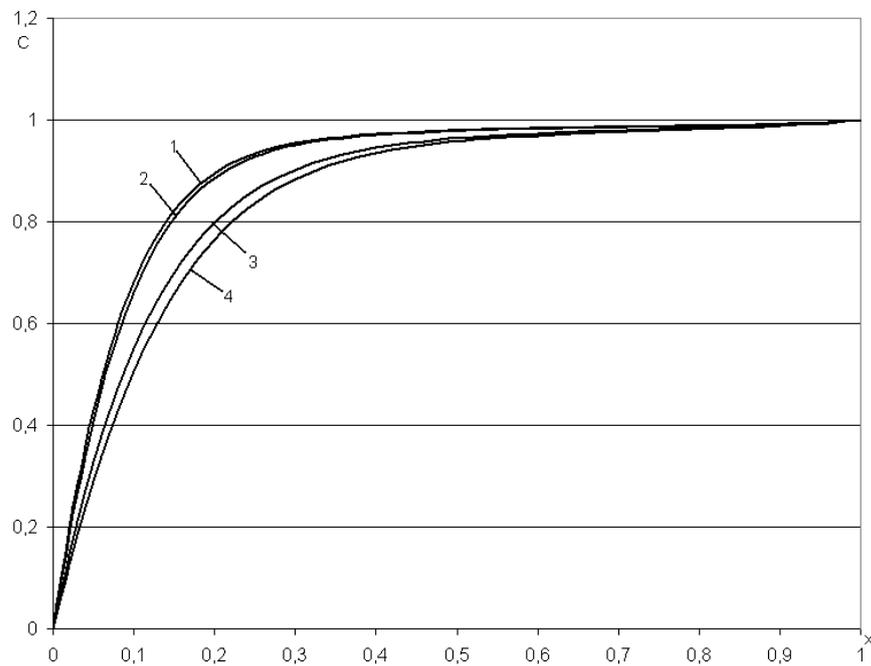


Рисунок 4.4 – Криві концентрації для $\alpha = 0,8$; $\beta = 0,9$; $\gamma = 0,7$; $t = 3$ (криві 1, 2) чи $t = 9$ (криві 3, 4) при наявності масообміну (1, 3 – $\gamma = 0,01$) або його відсутності (2, 4 – $\gamma = 0$)

Результати розрахунків дозволяють зробити наступні висновки [34]:

1. Розглядувана модель дає опис динаміки процесу розсолення ґрунтів, що якісно узгоджується з описом цього процесу в рамках класичної математичної моделі [311], проте результати моделювання в рамках некласичної моделі можуть істотно відрізнятися кількісно (рис. 4.2, 4.3);

2. На динаміку процесу розсолення істотно впливає величина порядку β дробової похідної за геометричною змінною у задачі стосовно поля концентрації солей. Так, зменшення величини β призводить до помітного прискорення процесу розсолення (рис. 4.2), зокрема і в порівнянні з описом процесу в рамках класичної моделі (крива 1 на рис. 4.2);

3. Врахування у моделі ефектів пам'яті також істотно впливає на динаміку процесу розсолення, причому зменшення величини порядку γ дробової похідної за часом у задачі стосовно поля концентрації солей призводить до запізнювання розвитку процесу розсолення (рис. 4.3);

4. Врахування масообміну з вміщуючими породами, зокрема розчинення солей у твердій фазі, сприяє уповільненню процесу розсолення (рис. 4.4).

4.2. Математичне моделювання дробово-диференціальної динаміки конвективної дифузії розчинних речовин в умовах масообміну у двовимірному наближенні

4.2.1. Математична модель та початково-крайова задача

Розглянемо фільтраційну схему, відповідну задачі моделювання конвективної дифузії забруднень з річок, каналів або поверхневих накопичувачів промислових стоків (пп. 2.2.1, рис. 2.6). Будемо вивчати процес фільтрації в потенційному полі швидкостей

$$\vec{v} = (v_x, v_y) = \nabla \varphi, \operatorname{div} \vec{v} = 0,$$

де $\varphi = -\bar{k}h$ – потенціал швидкості фільтрації \vec{v} , \bar{k} – усереднений коефіцієнт фільтрації, h – п'єзометричний тиск.

Як відомо, класична математична модель конвективної дифузії розчинних речовин з урахуванням масообміну у двовимірному фільтраційному полі швидкостей базується на рівнянні виду [280, 310, 311]

$$\sigma \frac{\partial C}{\partial t} + \frac{\partial N}{\partial t} = D \Delta_{xy} - v_x \frac{\partial C}{\partial x} - v_y \frac{\partial C}{\partial y}, \quad (4.19)$$

де C – концентрація речовин у рідкій фазі, N – концентрація речовин у твердій фазі, σ – пористість середовища, $v_x = v_x(x, y)$, $v_y = v_y(x, y)$ – складові вектора швидкості фільтрації, $D = const$ – коефіцієнт конвективної дифузії [311], Δ_{xy} – оператор Лапласа за змінними x, y .

Для коректності математичної моделі, до (4.19) необхідно додати рівняння кінетики масообміну, яке запишемо у вигляді рівняння нерівноважної оборотної сорбції при ізотермі Генрі [311]

$$\frac{\partial N}{\partial t} = \beta_* (\sigma C - \alpha_* N), \quad (4.20)$$

де $\alpha_* = 1/\Gamma$, Γ – коефіцієнт Генрі [280], β_* – константа швидкості масообміну.

При значній мінералізації порового розчину коефіцієнт фільтрації може істотно змінюватися в залежності від зміни концентрації розчинених у воді солей. Кількісні оцінки впливу ступеня мінералізації на проникність ґрунтів наведені, зокрема, в [285]. У [281] викладені результати експериментальних досліджень фільтрації сольових розчинів у піщаних ґрунтах і суглинках. Їх математична обробка у випадку суглинків призводить до наступної залежності коефіцієнта фільтрації k від концентрації C сольового розчину [281]:

$$k(C) = a_0 + a_1 C + a_2 C^2 + a_3 C^3 + a_4 C^4 + a_5 C^5, \quad (4.21)$$

де $a_0 = 1,0054 \cdot 10^{-3}$, $a_1 = 1,0563 \cdot 10^{-2}$, $a_2 = -7,4311 \cdot 10^{-2}$, $a_3 = 1,7051 \cdot 10^{-1}$, $a_4 = -1,6703 \cdot 10^{-1}$, $a_5 = 5,9404 \cdot 10^{-2}$, $C \in [0,1]$ – безрозмірна величина концентрації.

Тоді залежність коефіцієнта конвективної дифузії від концентрації C розчинних речовин матиме вигляд [281]:

$$D = D(v, C) = D_m + \lambda \frac{k(C)}{\bar{k}} |\vec{v}(x, y)|, \quad (4.22)$$

де D_m – коефіцієнт молекулярної дифузії, λ – параметр гідродинамічної дисперсії, $k(C)$ – коефіцієнт фільтрації пористого середовища, як функція концентрації C , що визначається згідно з (4.21), \bar{k} – усереднений коефіцієнт фільтрації.

Розглядатимемо дробово-диференціальну модель динаміки міграційного процесу з урахуванням масообміну з вміщуючими породами на основі нелокального оператора дробової похідної Капуто–Катугампола від функції p за функцією $g(t) = t^\rho$ ($\rho > 0$) часової змінної, який визначається [11, 134] як

$$D_{t,t^\rho}^{(\beta)} p(t) = \frac{1}{\Gamma(1-\beta)} \int_0^t \frac{p'(\tau)}{(t^\rho - \tau^\rho)^\beta} d\tau,$$

де β ($0 < \beta < 1$) – порядок похідної, $\Gamma(z)$ – гамма-функція Ейлера.

Аналогічно [38, 278] отримуємо наступну систему рівнянь для концентрацій C і N :

$$\sigma D_{t,t^\rho}^{(\beta)} C + \frac{\partial N}{\partial t} = \frac{\partial}{\partial x} \left(D(v, C) \frac{\partial C}{\partial x} \right) + \frac{\partial}{\partial y} \left(D(v, C) \frac{\partial C}{\partial y} \right) - v_x \frac{\partial C}{\partial x} - v_y \frac{\partial C}{\partial y}, \quad (4.23)$$

$$\frac{\partial N}{\partial t} = \beta_* (\sigma C - \alpha_* N). \quad (4.24)$$

Крайові умови для системи (4.23), (4.24) запишемо у вигляді

$$C|_{AC} = C_0, \quad \frac{\partial C}{\partial n} \Big|_{AB, CB} = 0, \quad C|_{t=0} = 0, \quad N|_{t=0} = N_0, \quad (4.25)$$

де C_0 – концентрація розчинних речовин на вході фільтраційного потоку, N_0 – початкова концентрація цих речовин у твердій фазі, n – зовнішня нормаль до відповідної кривої, AB – вісь симетрії потоку, CB – лінія течії (рис. 2.6, а).

Крайова задача (4.23), (4.24), (4.25) для дослідження дробово-диференціальної динаміки даного міграційного процесу, як і вище, ефективно розв'язується шляхом переходу до точок області комплексного потенціалу течії $G_\omega = \{(\varphi, \psi) : 0 < \varphi < +\infty, 0 < \psi < Q\}$ (рис. 2.6, б) і математично може бути сформульована у вигляді

$$\begin{aligned} & \sigma D_{t,t^\rho}^{(\beta)} C(\varphi, \psi, t) + \frac{\partial N(\varphi, \psi, t)}{\partial t} = \\ & = \nu^2(\varphi, \psi) \left[\frac{\partial}{\partial \varphi} \left(D(\varphi, \psi, C) \frac{\partial C}{\partial \varphi} \right) + \frac{\partial}{\partial \psi} \left(D(\varphi, \psi, C) \frac{\partial C}{\partial \psi} \right) - \frac{\partial C}{\partial \varphi} \right], \end{aligned} \quad (4.26)$$

$$\frac{\partial N(\varphi, \psi, t)}{\partial t} = \beta_*(\sigma C - \alpha_* N), \quad ((\varphi, \psi, t) \in G_\omega \times (0, +\infty)), \quad (4.27)$$

$$C|_{\varphi=0} = C_0, \quad \frac{\partial C}{\partial \psi} \Big|_{\psi=0, \psi=Q} = 0, \quad C|_{t=0} = 0, \quad N|_{t=0} = N_0, \quad (4.28)$$

де $\nu^2(\varphi, \psi)$ визначається згідно з (2.28), а коефіцієнт дифузії D визначається відповідно до (4.22).

4.2.2. Чисельне розв'язання крайової задачі

Введемо до розгляду сіткову область

$$\omega_{h\tau} = \left\{ (\varphi_i, \psi_k, t_j) : \varphi_i = ih_1 \left(i = \overline{0, m+1} \right), \psi_k = h_2(k-0,5) \left(k = \overline{0, n+1} \right), t_j = j\tau \left(j = \overline{0, J} \right) \right\},$$

де h_1, h_2, τ – кроки сітки за геометричними змінним та часом, відповідно, і обмежимо область комплексного потенціалу течії справа прямою $\varphi = \varphi_0$ ($\varphi_0 \gg 1$) зводячи її до прямокутного вигляду. Поставимо у відповідність крайової задачі (4.26), (4.27), (4.28) наступний лінеаризований аналог [38, 278] локально-одновимірної [313] різницевої схеми О. А. Самарського:

$$\frac{1}{2}(\sigma \Delta_t^{(\beta)} \bar{C} + N_t) = \nu^2 \left((\tilde{D} \bar{C}_{\bar{\varphi}})_{\varphi} - \bar{C}_0 \right), \quad (4.29)$$

$$\frac{1}{2}(\sigma \Delta_t^{(\beta)} \hat{C} + \bar{N}_t) = \nu^2 \left(\tilde{\tilde{D}} \hat{C}_{\bar{\psi}} \right)_{\psi}, \quad (4.30)$$

$$N_t = \beta_*(\sigma C - \alpha_* \bar{N}), \quad (4.31)$$

$$\bar{N}_t = \beta_*(\sigma \bar{C} - \alpha_* \hat{N}), \quad (4.32)$$

де сіткові функції $\tilde{D}, \tilde{\tilde{D}}$ обчислюються за формулами [313]

$$\tilde{D}_{ik} = 0,5(D_{i-1,k} + D_{ik}), \quad \tilde{\tilde{D}}_{ik} = 0,5(D_{i,k-1} + D_{ik}).$$

При цьому, різницеві аналоги дробової похідної Капуто–Катугампола у першому наближенні визначимо наступним чином:

$$\Delta_t^{(\beta)} \bar{C} = \frac{1}{\tau} \left(2b_j \left(C^{j+\frac{1}{2}} - C^j \right) + \sum_{s=0}^{j-1} \bar{b}_s^{(j)} \left(C^{s+1} - C^s \right) \right), \quad (4.33)$$

$$\Delta_t^{(\beta)} \hat{C} = \frac{2}{\tau} \sum_{s=0}^j \left[\bar{b}_s^{(j)} C^{s+1} + \left(\bar{q}_s^{(j)} - \bar{b}_s^{(j)} \right) C^{s+\frac{1}{2}} - \bar{q}_s^{(j)} C^s \right], \quad (4.34)$$

де

$$b_j = \frac{1}{\Gamma(1-\beta)} \int_{t_j}^{t_{j+\frac{1}{2}}} \frac{d\tau}{\left(g\left(t_{j+\frac{1}{2}}\right) - g(\tau) \right)^\beta}, \quad \bar{b}_s^{(j)} = \frac{1}{\Gamma(1-\beta)} \int_{t_s}^{t_{s+1}} \frac{d\tau}{\left(g\left(t_{j+\frac{1}{2}}\right) - g(\tau) \right)^\beta},$$

$$\bar{q}_s^{(j)} = \frac{1}{\Gamma(1-\beta)} \int_{t_s}^{t_{s+\frac{1}{2}}} \frac{d\tau}{\left(g\left(t_{j+1}\right) - g(\tau) \right)^\beta}, \quad \bar{b}_s^{(j)} = \frac{1}{\Gamma(1-\beta)} \int_{t_{s+\frac{1}{2}}}^{t_{s+1}} \frac{d\tau}{\left(g\left(t_{j+1}\right) - g(\tau) \right)^\beta}, \quad g(t) = t^\rho.$$

Розписуючи в (4.29), (4.31) різницеві оператори з урахуванням (4.33), (4.34) і зводячи подібні члени, отримуємо на напівцілому шарі $t_{j+\frac{1}{2}}$ систему лінійних алгебраїчних рівнянь

$$A_{ik}^j C_{i+1,k}^{j+\frac{1}{2}} - B_{ik}^j C_{ik}^{j+\frac{1}{2}} + S_{ik}^j C_{i-1,k}^{j+\frac{1}{2}} = \Phi_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, J}), \quad (4.35)$$

де

$$A_{ik}^j = \frac{\nu_{ik}^2}{h_1} \left(\frac{\tilde{D}_{i+1,k}^j}{h_1} - 0,5 \right), \quad S_{ik}^j = \frac{\nu_{ik}^2}{h_1} \left(\frac{\tilde{D}_{ik}^j}{h_1} + 0,5 \right), \quad B_{ik}^j = \frac{\sigma}{\tau} b_j + A_{ik}^j + S_{ik}^j,$$

$$\tilde{D}_{ik}^j = 0,5 \left(D_{i-1,k}^j + D_{ik}^j \right), \quad D_{ik}^j = D_m + \lambda \frac{k(C_{ik}^j)}{\bar{k}} |\bar{\nu}_{ik}|,$$

$$\Phi_{ik}^j = \frac{\sigma}{\tau} \left(\frac{1}{2} \sum_{s=0}^{j-1} \bar{b}_s^{(j)} \left(C_{ik}^{s+1} - C_{ik}^s \right) - b_j C_{ik}^j \right) + \frac{1}{\tau} \left(N_{ik}^{j+\frac{1}{2}} - N_{ik}^j \right),$$

$$N_{ik}^{j+\frac{1}{2}} = \frac{1}{2 + \tau \alpha_* \beta_*} \left(2N_{ik}^j + \tau \sigma \beta_* C_{ik}^j \right).$$

Аналогічно на цілому часовому шарі з (4.30), (4.32) отримуємо

$$P_{ik}^j C_{i,k+1}^{j+1} - Q_{ik}^j C_{ik}^{j+1} + R_{ik}^j C_{i,k-1}^{j+1} = \Omega_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, J}), \quad (4.36)$$

$$P_{ik}^j = \frac{\nu_{ik}^2 \tilde{D}_{i,k+1}^j}{h_2^2}, \quad R_{ik}^j = \frac{\nu_{ik}^2 \tilde{D}_{ik}^j}{h_2^2}, \quad Q_{ik}^j = \frac{\sigma}{\tau} \bar{b}_j^{(j)} + P_{ik}^j + R_{ik}^j,$$

$$\Omega_{ik}^j = \frac{\sigma}{\tau} \left(\sum_{s=0}^j \bar{q}_s^{(j)} \left(C_{ik}^{s+\frac{1}{2}} - C_{ik}^s \right) - \bar{b}_j^{(j)} C_{ik}^{j+\frac{1}{2}} + \sum_{s=0}^{j-1} \bar{b}_s^{(j)} \left(C_{ik}^{s+1} - C_{ik}^{s+\frac{1}{2}} \right) \right) + \frac{1}{\tau} \left(N_{ik}^{j+1} - N_{ik}^{j+\frac{1}{2}} \right),$$

$$N_{ik}^{j+1} = \frac{1}{2 + \tau \alpha_* \beta_*} \left(2N_{ik}^{j+\frac{1}{2}} + \tau \sigma \beta_* C_{ik}^{j+\frac{1}{2}} \right).$$

Розв'язки СЛАР (4.35), (4.36) з тридіагональними матрицями згідно з методом прогонки [313] запишуться у вигляді

$$C_{ik}^{j+\frac{1}{2}} = \alpha_{i+1,k}^j C_{i+1,k}^{j+\frac{1}{2}} + \beta_{i+1,k}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, J}), \quad (4.37)$$

$$C_{ik}^{j+1} = \tilde{\alpha}_{i,k+1}^j C_{i,k+1}^{j+1} + \tilde{\beta}_{i,k+1}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, J}), \quad (4.38)$$

де прогоночні коефіцієнти обчислюються за формулами

$$\alpha_{i+1,k}^j = \frac{A_{ik}^j}{B_{ik}^j - S_{ik}^j \alpha_{ik}^j}, \quad \beta_{i+1,k}^j = \frac{\alpha_{i+1,k}^j (S_{ik}^j \beta_{ik}^j - \Phi_{ik}^j)}{A_{ik}^j}, \quad (4.39)$$

$$\tilde{\alpha}_{i,k+1}^j = \frac{P_{ik}^j}{Q_{ik}^j - R_{ik}^j \tilde{\alpha}_{ik}^j}, \quad \tilde{\beta}_{i,k+1}^j = \frac{\tilde{\alpha}_{i,k+1}^j (R_{ik}^j \tilde{\beta}_{ik}^j - \Omega_{ik}^j)}{P_{ik}^j} \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, J}), \quad (4.40)$$

$$\alpha_{1k}^j = 0, \quad \beta_{1k}^j = C_0 \quad (k = \overline{1, n}; j = \overline{0, N}), \quad \tilde{\alpha}_{i1}^j = 1, \quad \tilde{\beta}_{i1}^j = 0 \quad (i = \overline{1, m}; j = \overline{0, J}), \quad (4.41)$$

$$C_{m+1,k}^{j+\frac{1}{2}} = \frac{\beta_{m+1,k}^j}{1 - \alpha_{m+1,k}^j} \quad (k = \overline{1, n}; j = \overline{0, N}), \quad C_{i,n+1}^{j+1} = \frac{\tilde{\beta}_{i,n+1}^j}{1 - \tilde{\alpha}_{i,n+1}^j} \quad (i = \overline{1, m}; j = \overline{0, J}). \quad (4.42)$$

Співвідношення (4.37)–(4.42) дозволяють обчислити розв'язок на цілому часовому шарі. Стійкість методу прогонки для (4.35), (4.36) впливає з факту діагонального переважання в матрицях коефіцієнтів цих СЛАР. Остаточний розв'язок задачі отримуємо після переходу в фізичну область G_z , який здійснюється відповідно до (2.26), (2.27).

4.2.3. Результати обчислювальних експериментів

Чисельне моделювання особливостей дробово-диференціальної динаміки процесу міграції розчинних речовин у рамках вищеописаної неklasичної конвективно-дифузійної моделі виконано для вхідних даних з роботи [285]. Вихідні коди розробленого програмного забезпечення наведено у [265]. Деякі з отриманих при цьому результатів в безрозмірних змінних $t' = t/t_0$, $\varphi' = \varphi/Q$, $\psi' = \psi/Q$,

$C' = C/C_0$ (t_0 – характерний часовий параметр, чисельне значення якого в розрахунках приймалося рівним $t_0 = 5 \text{ дiб}$) графічно зображені на рис. 4.5–4.8 (без урахування масообміну) та рис. 4.9–4.12 (з урахуванням масообміну). Знак "штрих" над безрозмірними змінними опущений.

Аналіз отриманих результатів дозволяє зробити наступні висновки [38, 278] щодо особливості динаміки полів концентрацій розчинних речовин при описі міграційного процесу на основі моделі з похідною Капуто–Катугампола:

1. У випадку моделювання дробово-диференціальної динаміки міграційного процесу в рамках моделі з класичною похідною Капуто–Герасимова ($\rho = 1$), фронт концентрації розчинних речовин при врахуванні нелінійної залежності $k = k(C)$ істотно випереджає фронт концентрації, розрахований при $k = \text{const}$ (рис. 4.5);

2. У припущенні наявності залежності $k = k(C)$ зі зменшенням значень порядку дробової похідної β відбувається запізнювання розвитку фронту концентрації у рідкій фазі (рис. 4.6);

3. Величина показника ступеня ρ істотним чином впливає на результати моделювання процесу конвективної дифузії в рамках досліджуваної моделі, даючи як субдифузійну (криві 2 на рис. 4.7, 4.8), так і супердифузійну (криві 3 на рис. 4.7, 4.8) картини розподілу полів концентрацій;

4. Врахування явища масообміну при моделюванні поширення забруднень з водойми у ґрунтове середовище призводить до запізнювання в розвитку фронту концентрацій у рідкій фазі (рис. 4.9–4.12);

5. Ступінь впливу масообміну на динаміку міграційного процесу залежить від величини порядку β дробової похідної наступним чином. Зменшення величини запізнювання спостерігається в разі зменшення порядку похідної і навпаки (рис. 4.9). Характер цієї закономірності не порушується при зміні величини показника ступеня ρ (рис. 4.10);

6. При фіксованому значенні показника β , вплив масообміну на динаміку полів концентрацій зростає при збільшенні показника ρ (рис. 4.11);

7. Запізнення у розвитку фронту концентрацій при наявності масообміну істотно залежить від параметрів масообміну α_* , β_* . Вплив масообміну зростає зі збільшенням значення α_* (рис. 4.12). Збільшення значення β_* при фіксованому α_* додатково уповільнює процес формування концентраційних полів, особливо на початкових його етапах (рис. 4.12).

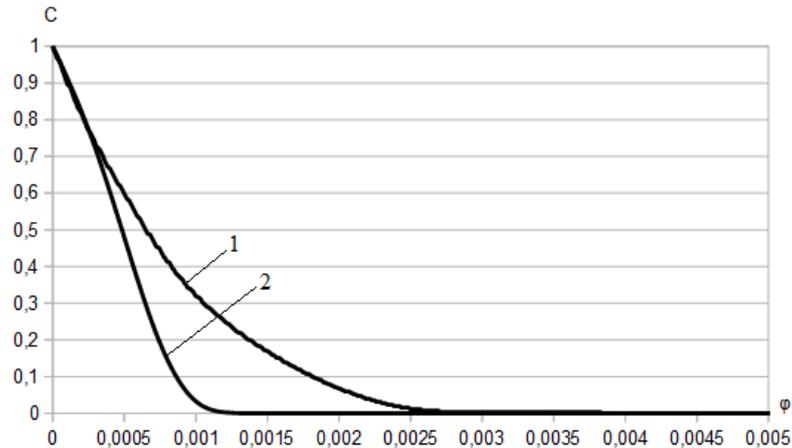


Рисунок 4.5 – Криві концентрацій вздовж лінії течії $\psi = 0,5$ при $\beta = 0,8$; $\rho = 1$; $t = 10$ та врахуванні нелінійної залежності $k = k(C)$ (крива 1) у порівнянні з випадком усередненого значення $k = 0,00084$ (крива 2)

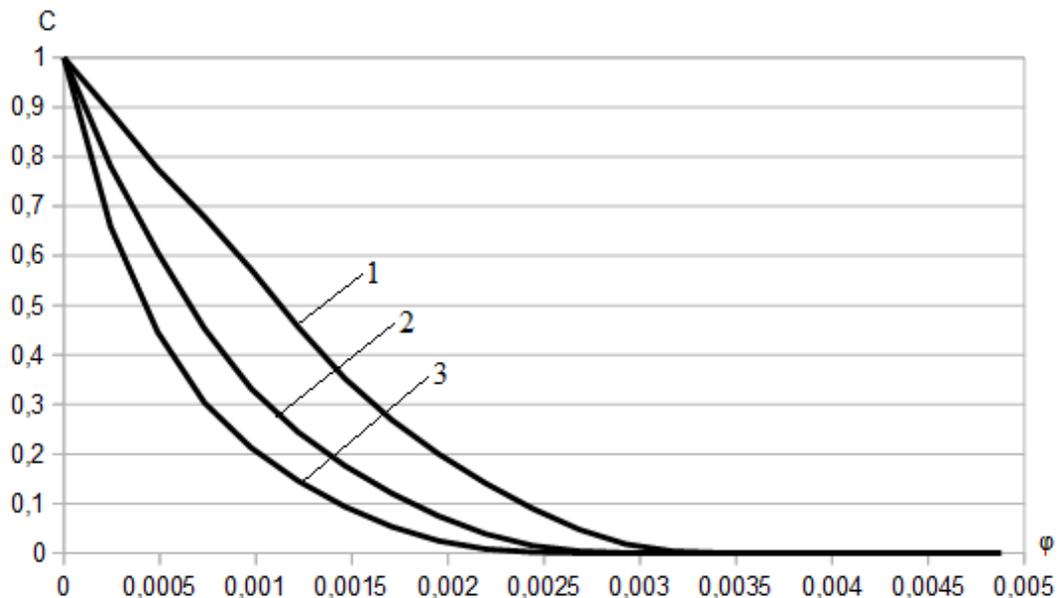


Рисунок 4.6 – Криві концентрацій вздовж лінії течії $\psi = 0,5$ при $\rho = 1$, $t = 10$ та врахуванні нелінійної залежності $k = k(C)$ (1 – $\beta = 1,0$; 2 – $\beta = 0,8$; 3 – $\beta = 0,6$; $t = 10$)

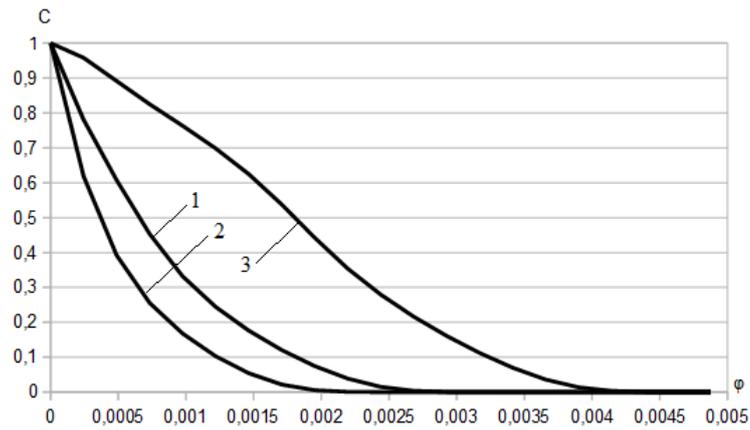


Рисунок 4.7 – Криві концентрацій вздовж лінії течії $\psi = 0,5$ при $\beta = 0,8$; $t = 10$ та врахуванні нелінійної залежності $k = k(C)$ ($1 - \rho = 1, 2 - \rho = \frac{1}{2}, 3 - \rho = 2$)

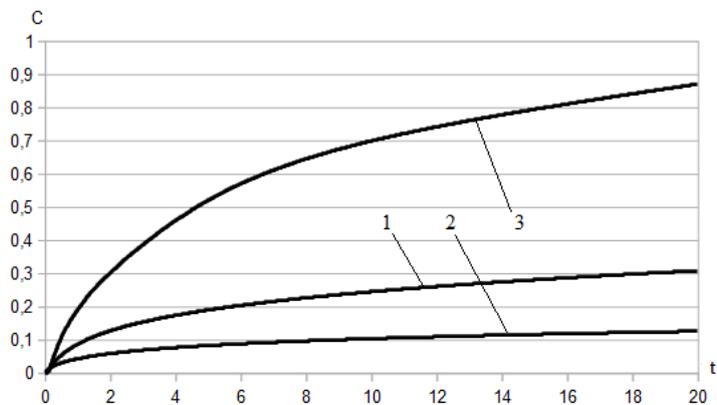


Рисунок 4.8 – Динаміка концентрації у точці $(1,25; 0,5)$ області комплексного потенціалу течії при $k = k(C)$, $\beta = 0,8$ ($1 - \rho = 1, 2 - \rho = \frac{1}{2}, 3 - \rho = 2$)

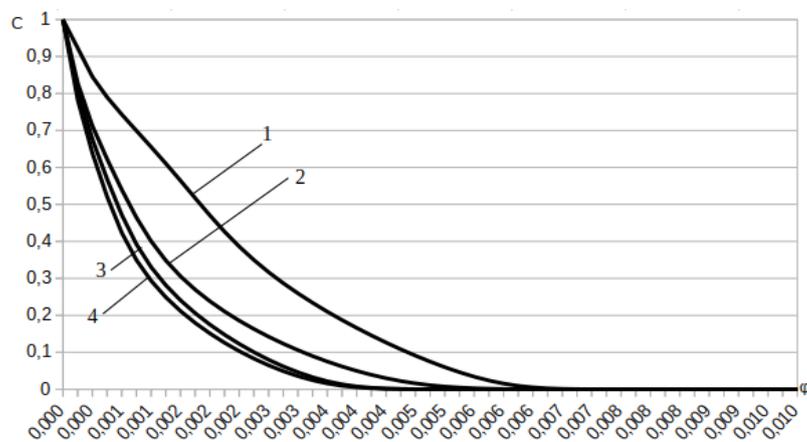


Рисунок 4.9 – Криві концентрації уздовж лінії течії $\psi = 0,5$ при $\rho = 1$, $t = 50$ та $1,3 - \beta = 1,0; 2,4 - \beta = 0,9$ без урахування масообміну (криві 1,2) і з його урахуванням (криві 3,4) при $\alpha_* = 0,1; \beta_* = 0,1$

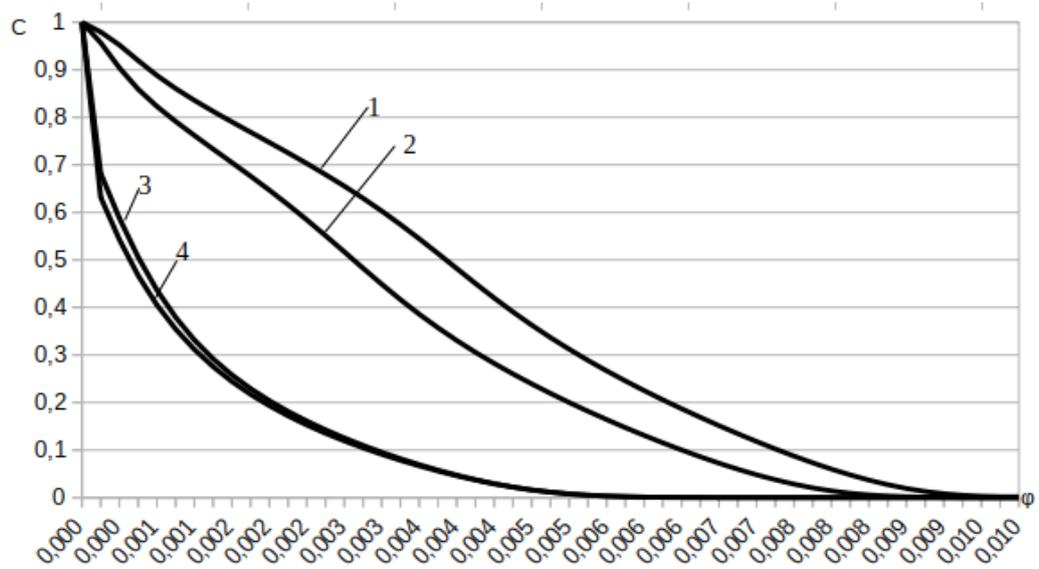


Рисунок 4.10 – Криві концентрації уздовж лінії течії $\psi = 0,5$ при $\rho = 2$, $t = 50$ та 1,3– $\beta = 0,9$; 2,4– $\beta = 0,8$ без урахування масообміну (криві 1,2) і з його урахуванням (криві 3,4) при $\alpha_* = 0,1$; $\beta_* = 0,1$

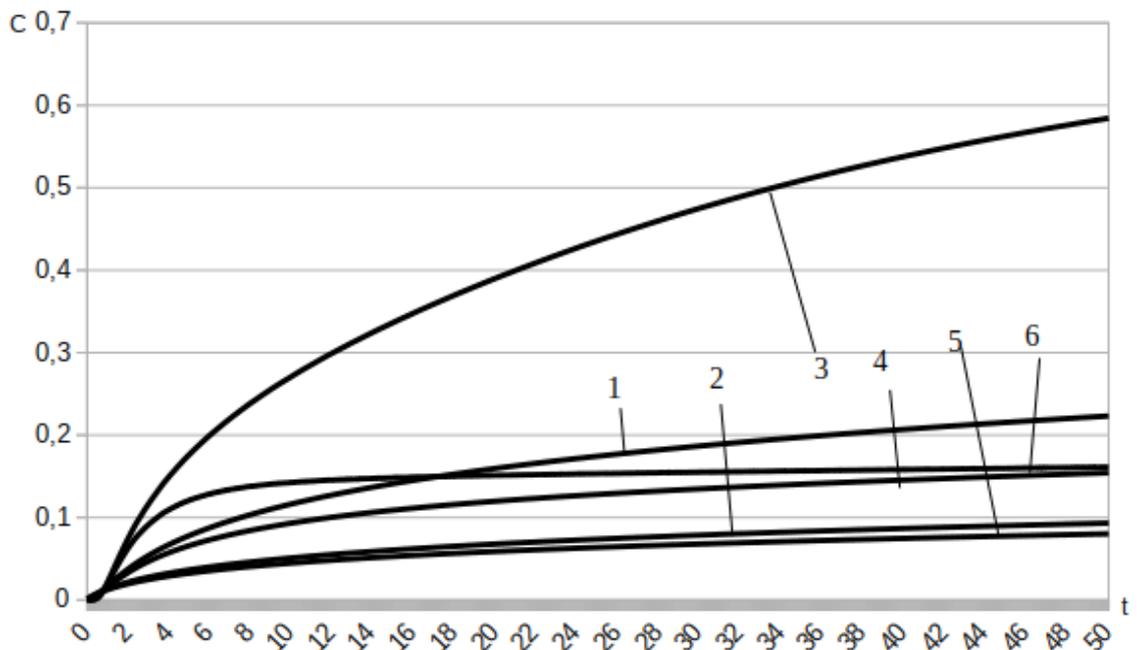


Рисунок 4.11 – Динаміка концентрації у точці $(0,0025; 0,5)$ області комплексного потенціалу при $\beta = 0,8$ і 1,4– $\rho = 1$, 2,5– $\rho = 1/2$, 3,6– $\rho = 2$ без урахування масообміну (криві 1,2,3) і з його урахуванням (криві 4,5,6) при $\alpha_* = 0,1$; $\beta_* = 0,1$

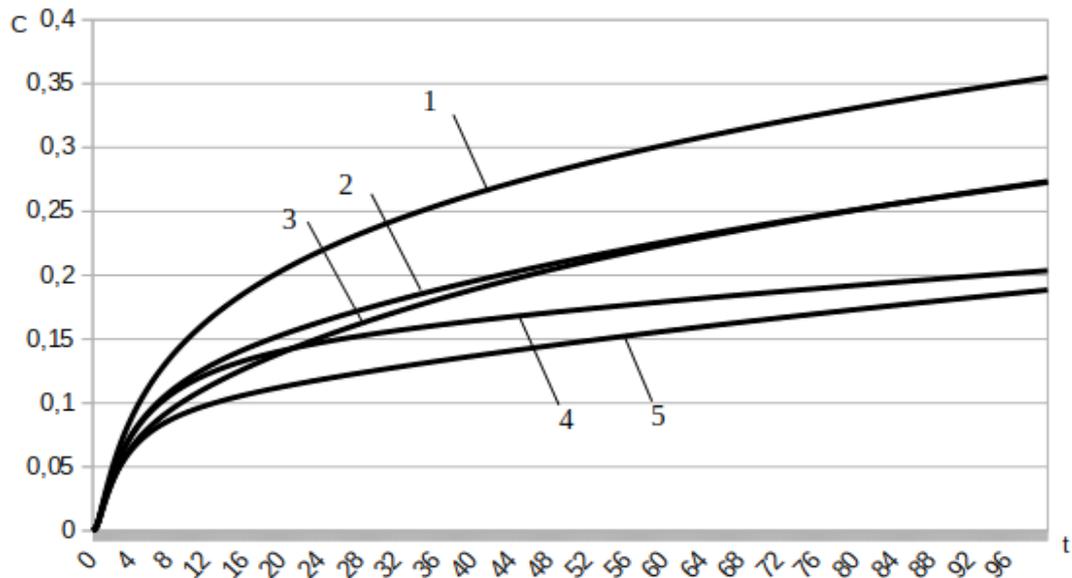


Рисунок 4.12 – Динаміка концентрації у точці $(0,0025; 0,5)$ області комплексного потенціалу при $\beta = 0,9$; $\rho = 2$ та врахуванні явища масообміну при $1 - \alpha_* = 0, \beta_* = 0$; $2 - \alpha_* = 0,1; \beta_* = 0,1$; $3 - \alpha_* = 0,1; \beta_* = 0,2$; $4 - \alpha_* = 0,4; \beta_* = 0,1$; $5 - \alpha_* = 0,4; \beta_* = 0,2$

4.3. Математичне моделювання дробово-диференціальної мобільно-немобільної міграційної динаміки у двовимірному наближенні

4.3.1. Математична модель та початково-крайова задача

Рівняння конвективної дифузії з урахуванням іммобілізації (захоплення частинки скелетом ґрунту або потрапляння її в об'єм зв'язаної рідини) має вигляд [88, 201, 221]

$$\frac{\partial C}{\partial t} + \beta \frac{\partial C_I}{\partial t} = \mathcal{L}(C), \quad (4.43)$$

де C, C_I – об'ємні концентрації частинок, що знаходяться у мобільній та немобільній фазах, відповідно, $\beta = \theta_I / \theta$, θ, θ_I – пористість у мобільній та немобільній зонах, відповідно, $\mathcal{L}(C) = d\Delta C - \vec{v}\nabla C$, \vec{v} – швидкість фільтрації, d – коефіцієнт конвективної дифузії, ∇ – оператор Гамільтона.

Згідно з [201] динаміка відтоку частинок у немобільну фазу описується співвідношенням

$$\frac{\partial C_I}{\partial t} = {}^c D_t^\gamma C, \quad (4.44)$$

де ${}^c D_t^\gamma$ – оператор дробової похідної Капуто–Герасимова порядку γ ($0 < \gamma < 1$) за змінною t [139]. Тоді, з урахуванням (4.44) рівняння (4.43) МІМ (mobile-immobile media) моделі набуває вигляду [88, 201, 221]

$$\frac{\partial C}{\partial t} + \beta {}^c D_t^\gamma C = \mathcal{L}(C), \quad (4.45)$$

а повна концентрація обчислюється згідно зі співвідношенням $C_{tot} = \theta C + \theta_I C_I$. З урахуванням ефектів пам'яті та спадкових властивостей середовища в процесі міграції у мобільній фазі, рівняння (4.45) запишеться у вигляді, аналогічному отриманому в рамках математичної моделі водообміну у мобільно-немобільних зонах ґрунтів, що набухають [207],

$${}^c D_t^\alpha C + \beta {}^c D_t^\gamma C = \mathcal{L}(C), \quad (4.46)$$

де ${}^c D_t^\alpha$ – оператор дробової похідної Капуто–Герасимова порядку α ($0 < \alpha < 1$) за часовою змінною.

Розглянемо в рамках математичної моделі, що базується на рівнянні (4.46), задачу моделювання дробово-диференціальної динаміки процесу конвективної дифузії розчинних речовин в умовах плосковертикальної усталеної фільтрації ґрунтових вод з водоюми AB до нескінченного дренажного каналу CD , заглибленого на глибину H (рис. 4.13, де наведені відповідна фізична область фільтрації (а) і область комплексного потенціалу течії (б)).

Характеристична функція течії $z = f(\omega)$ та поле швидкостей для відповідної крайової задачі фільтрації, що була отримана методом конформних відображень області фільтрації G_z (рис. 4.13, а) на область G_ω (рис. 4.13, б) у площині комплексного потенціалу течії $\omega = \varphi + i\psi$, де φ – потенціал швидкості фільтрації, ψ – функція течії, наведена у роботі [300]. Область комплексного потенціалу течії G_ω в даній задачі є [300] прямокутником зі сторонами φ_0, Q , де $\varphi_0 = \kappa H$, κ – коефіцієнт фільтрації ґрунту, Q – повна фільтраційна витрата.

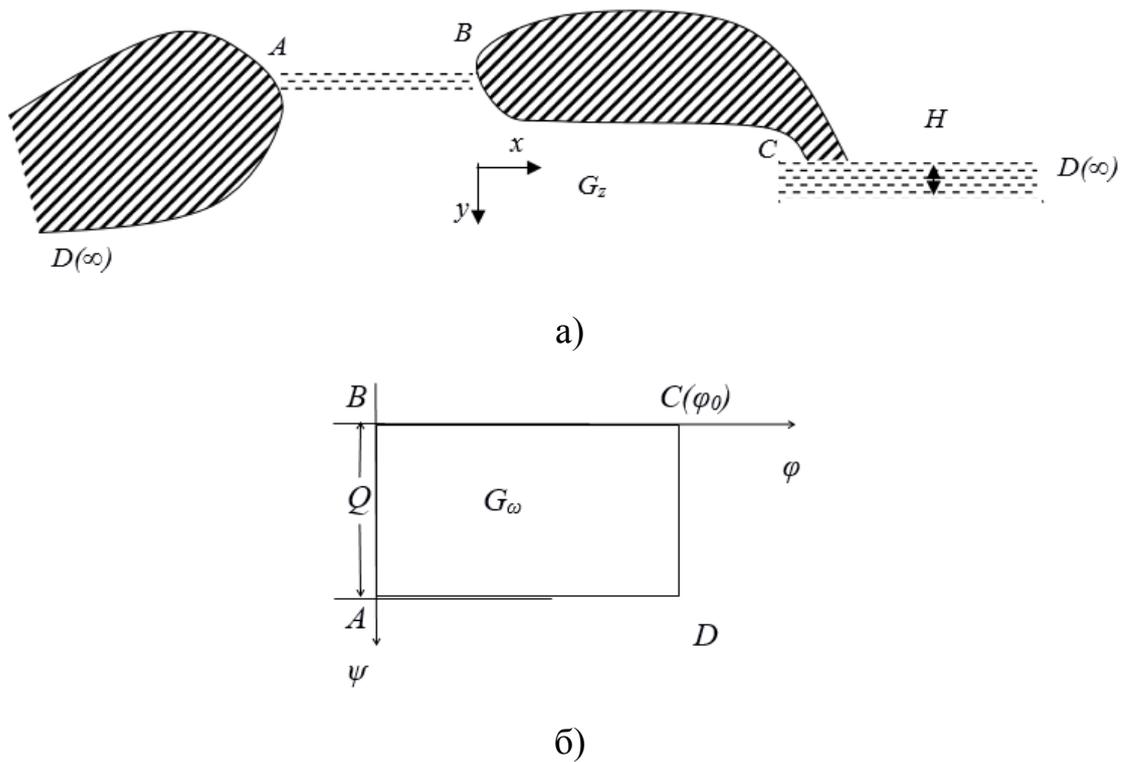


Рисунок 4.13 – Фізична область фільтрації (а) і область комплексного потенціалу течії (б)

Задачу математичного опису дробово-диференціальної динаміки процесу конвективної дифузії при усталеній профільній фільтрації ґрунтових вод відповідно до фільтраційної схеми, зображеної на рис. 4.13, можна сформулювати як задачу знаходження в області $G_z \times (0, \infty)$ розв'язку рівняння [65]

$${}^c D_t^\alpha C(x, y, t) + \beta {}^c D_t^\gamma C(x, y, t) = d \left(\frac{\partial^2 C}{\partial x^2} + \frac{\partial^2 C}{\partial y^2} \right) - v_x \frac{\partial C}{\partial x} - v_y \frac{\partial C}{\partial y} \quad (4.47)$$

$$(0 < \alpha, \gamma < 1)$$

при крайових умовах

$$C|_{AB} = C_1, \quad \frac{\partial C}{\partial n} \Big|_{AD, BC} = 0, \quad \frac{\partial C}{\partial y} \Big|_{CD} = 0, \quad (4.48)$$

$$C(x, y, 0) = C_0(x, y), \quad (4.49)$$

де $C_1 = \text{const}$ – задана концентрація розчинних речовин на вході фільтраційного потоку, $C_0(x, y)$ – задана функція початкового розподілу концентрації, n – зовнішня нормаль до відповідної кривої, d – коефіцієнт конвективної дифузії, v_x, v_y – проекції вектора швидкості фільтрації на осі Ox та Oy , відповідно.

У області комплексного потенціалу течії G_ω , крайову задачу (4.47)–(4.49) можна сформулювати у вигляді

$$\left({}^c D_t^\alpha C(\varphi, \psi, t) + \beta {}^c D_t^\gamma C(\varphi, \psi, t) \right) = \nu^2(\varphi, \psi) \left(d \Delta C(\varphi, \psi, t) - \frac{\partial C}{\partial \varphi} \right) \quad (4.50)$$

$$(0 < \alpha, \gamma < 1),$$

$$C|_{\varphi=0} = C_1, \quad \frac{\partial C}{\partial \psi} \Big|_{\psi=0, \psi=Q} = 0, \quad \frac{\partial C}{\partial \varphi} \Big|_{\varphi=\varphi_0} = 0, \quad (4.51)$$

$$C(\varphi, \psi, 0) = C_0(\varphi, \psi), \quad (4.52)$$

де $\nu^2 = \nu_x^2 + \nu_y^2 = \left(\frac{\partial \varphi}{\partial x} \right)^2 + \left(\frac{\partial \varphi}{\partial y} \right)^2$, $\Delta = \frac{\partial^2}{\partial \varphi^2} + \frac{\partial^2}{\partial \psi^2}$ – оператор Лапласа.

Вводячи до розгляду безрозмірні змінні та параметри

$$\varphi' = \frac{\varphi}{Q}, \quad \psi' = \frac{\psi}{Q}, \quad t' = \left(\frac{\nu_0^2}{Q} \right)^{\frac{1}{\alpha}} t, \quad C' = \frac{C}{C_1}, \quad \beta' = \beta \left(\frac{\nu_0^2}{Q} \right)^{\frac{\gamma}{\alpha}-1}, \quad \varphi'_0 = \frac{\varphi_0}{Q},$$

$$d' = \frac{d}{Q}, \quad \nu' = \frac{\nu}{\nu_0}, \quad (4.53)$$

де ν_0 – характерний параметр швидкості, перепишемо задачу (4.50)–(4.52) у вигляді (знак "штрих" над безрозмірними змінними надалі опускається)

$$\left({}^c D_t^\alpha + \beta {}^c D_t^\gamma \right) C(\varphi, \psi, t) = \nu^2(\varphi, \psi) \left(d \left(\frac{\partial^2 C}{\partial \varphi^2} + \frac{\partial^2 C}{\partial \psi^2} \right) - \frac{\partial C}{\partial \varphi} \right) \quad (4.54)$$

$$(0 < \alpha, \gamma < 1),$$

$$C|_{\varphi=0} = 1, \quad \frac{\partial C}{\partial \psi} \Big|_{\psi=0, \psi=1} = 0, \quad \frac{\partial C}{\partial \varphi} \Big|_{\varphi=\varphi_0} = 0, \quad (4.55)$$

$$C(\varphi, \psi, 0) = C_0(\varphi, \psi). \quad (4.56)$$

При цьому, в безрозмірних змінних область комплексного потенціалу течії визначається як $G_\omega = \{(\varphi, \psi) : 0 < \varphi < \varphi_0, 0 < \psi < 1\}$. Таким чином, моделювання динаміки розглядуваного міграційного процесу зводиться до розв'язання крайової задачі (4.54)–(4.56) з подальшим переходом з області G_ω у фізичну область G_z згідно з наведеним у роботі [300] розв'язком відповідної фільтраційної задачі.

4.3.2. Чисельне розв'язання крайової задачі

Вводячи до розгляду сіткову область

$$\omega_h = \left\{ (\varphi_i, \psi_k, t_j) : \varphi_i = ih_1 \left(i = \overline{0, m+1} \right), \psi_k = (k - 0,5)h_2 \left(k = \overline{0, n+1} \right), \right. \\ \left. t_j = j\tau \left(j = \overline{0, N+1} \right) \right\},$$

де h_1, h_2, τ – кроки сітки за геометричними змінним φ, ψ та часом t , відповідно, та позначаючи як $\Delta_t^{(\alpha)}$ дискретний аналог похідної Капуто порядку α , ставимо у відповідність задачі (4.54)–(4.56) наступний аналог [65] локально-одновимірної [313] різницевої схеми О. А. Самарського:

$$\frac{1}{2} \left(\Delta_t^{(\alpha)} \bar{C} + \beta \Delta_t^{(\gamma)} \bar{C} \right) = \nu^2 \left(d\bar{C}_{\bar{\varphi}\varphi} - \bar{C}_0 \right), \quad (4.57)$$

$$\frac{1}{2} \left(\Delta_t^{(\alpha)} \hat{C} + \beta \Delta_t^{(\gamma)} \hat{C} \right) = \nu^2 d\hat{C}_{\bar{\psi}\psi}, \quad (4.58)$$

де

$$\hat{C} = C^{j+1}, \bar{C} = C^{j+\frac{1}{2}}, C = C^j, t_{j+\frac{1}{2}} = t_j + \frac{\tau}{2},$$

$$\Delta_t^{(\alpha)} \bar{C} = \frac{1}{\Gamma(2-\alpha)} \left[\left(\frac{2}{\tau} \right)^\alpha (\bar{C} - C) + \sum_{s=0}^{j-1} b_s^{(j)} C_{t,s} \right], \quad (4.59)$$

$$\Delta_t^{(\alpha)} \hat{C} = \frac{1}{\Gamma(2-\alpha)} \left[\sum_{s=0}^{j-1} \rho_s^{(j)} C_{t,s} + \left(\frac{2}{\tau} \right)^\alpha \left(\hat{C} - \bar{C} + (2^{1-\alpha} - 1)(\bar{C} - C) \right) \right], \quad (4.60)$$

$$b_s^{(j)} = \tau^{1-\alpha} \left[\left(j + \frac{1}{2} - s \right)^{1-\alpha} - \left(j - \frac{1}{2} - s \right)^{1-\alpha} \right],$$

$$\rho_s^{(j)} = \tau^{1-\alpha} \left[(j+1-s)^{1-\alpha} - (j-s)^{1-\alpha} \right], C_{t,s} = \frac{C^{s+1} - C^s}{\tau},$$

а $C_0, C_{\bar{\varphi}\varphi}$ – центральна і друга різницеві похідні, відповідно.

Розписуючи у (4.57) різницеві оператори з урахуванням (4.59) і зводячи подібні члени, отримуємо на напівцілому часовому шарі $t_{j+\frac{1}{2}}$ систему рівнянь

$$A_{ik} C_{i+1,k}^{j+\frac{1}{2}} - B_{ik} C_{ik}^{j+\frac{1}{2}} + S_{ik} C_{i-1,k}^{j+\frac{1}{2}} = \Phi_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (4.61)$$

де

$$A_{ik} = \frac{\nu_{ik}^2}{h_1} \left(\frac{d}{h_1} - 0,5 \right), \quad S_{ik} = \frac{\nu_{ik}^2}{h_1} \left(\frac{d}{h_1} + 0,5 \right), \quad B_{ik} = w_{\alpha, \gamma} + A_{ik} + S_{ik},$$

$$\Phi_{ik}^j = \frac{1}{2} \sum_{s=0}^{j-1} \left(\frac{b_s^{(j)}}{\Gamma(2-\alpha)} + \frac{\beta \bar{b}_s^{(j)}}{\Gamma(2-\gamma)} \right) \frac{C_{ik}^{s+1} - C_{ik}^s}{\tau} - w_{\alpha, \gamma} C_{ik}^j,$$

$$w_{\alpha, \gamma} = \frac{1}{2} \left(\frac{2^\alpha}{\tau^\alpha \Gamma(2-\alpha)} + \frac{\beta 2^\gamma}{\tau^\gamma \Gamma(2-\gamma)} \right), \quad \bar{b}_s^{(j)} = \tau^{1-\gamma} \left[\left(j + \frac{1}{2} - s \right)^{1-\gamma} - \left(j - \frac{1}{2} - s \right)^{1-\gamma} \right].$$

На цілому часовому шарі з (4.58) з урахуванням (4.60) маємо

$$P_{ik} C_{i,k+1}^{j+1} - Q_{ik} C_{ik}^{j+1} + P_{ik} C_{i,k-1}^{j+1} = \Omega_{ik}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}), \quad (4.62)$$

де

$$P_{ik} = \frac{\nu_{ik}^2 d}{h_2^2}, \quad Q_{ik} = w_{\alpha, \gamma} + 2P_{ik},$$

$$\Omega_{ik}^j = \frac{1}{2} \sum_{s=0}^{j-1} \left(\frac{\rho_s^{(j)}}{\Gamma(2-\alpha)} + \frac{\beta \bar{\rho}_s^{(j)}}{\Gamma(2-\gamma)} \right) \frac{C_{ik}^{s+1} - C_{ik}^s}{\tau} + \frac{1}{2} \bar{w}_{\alpha, \gamma} \left(C_{ik}^{j+\frac{1}{2}} - C_{ik}^j \right) - w_{\alpha, \gamma} C_{ik}^{j+\frac{1}{2}},$$

$$\bar{w}_{\alpha, \gamma} = \frac{2^{1-\alpha} - 1}{\Gamma(2-\alpha)} \left(\frac{2}{\tau} \right)^\alpha + \frac{\beta(2^{1-\gamma} - 1)}{\Gamma(2-\gamma)} \left(\frac{2}{\tau} \right)^\gamma, \quad \bar{\rho}_s^{(j)} = \tau^{1-\gamma} \left[(j+1-s)^{1-\gamma} - (j-s)^{1-\gamma} \right].$$

Тут $\sum_{s=0}^{j-1} (\bullet) \equiv 0$ при $j-1 < 0$.

Різницеві рівняння систем (4.61), (4.62) є триточковими та ефективно розв'язуються методом прогонки [192, 313]. Стійкість методу прогонки для (4.61) та (4.62) впливає з факту діагонального переважання в матрицях коефіцієнтів цих СЛАР. Відповідні прогоночні співвідношення записуються у вигляді

$$C_{ik}^{j+\frac{1}{2}} = \alpha_{i+1,k} C_{i+1,k}^{j+\frac{1}{2}} + \beta_{i+1,k}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}),$$

$$C_{ik}^{j+1} = \tilde{\alpha}_{i,k+1} C_{i,k+1}^{j+1} + \tilde{\beta}_{i,k+1}^j \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}),$$

а прогоночні коефіцієнти обчислюються за формулами

$$\alpha_{i+1,k} = \frac{A_{ik}}{B_{ik} - S_{ik}\alpha_{ik}}, \quad \beta_{i+1,k}^j = \frac{\alpha_{i+1,k}}{A_{ik}} (S_{ik}\beta_{ik}^j - \Phi_{ik}^j),$$

$$\tilde{\alpha}_{i,k+1} = \frac{P_{ik}}{Q_{ik} - P_{ik}\tilde{\alpha}_{ik}}, \quad \tilde{\beta}_{i,k+1}^j = \frac{\tilde{\alpha}_{i,k+1}}{P_{ik}} (P_{ik}\tilde{\beta}_{ik}^j - \Omega_{ik}^j) \quad (i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N}).$$

Стартові значення прогоночних коефіцієнтів визначаються на основі різницевих аналогів граничних умов наступним чином:

$$\alpha_{1k} = 0, \quad \beta_{1k}^j = 1 \quad (k = \overline{1, n}; j = \overline{0, N}), \quad \tilde{\alpha}_{i1} = 1, \quad \tilde{\beta}_{i1}^j = 0 \quad (i = \overline{1, m}; j = \overline{0, N}),$$

$$C_{m+1,k}^{j+\frac{1}{2}} = \frac{\beta_{m+1,k}^j}{1 - \alpha_{m+1,k}} \quad (k = \overline{1, n}; j = \overline{0, N}), \quad C_{i,n+1}^{j+1} = \frac{\tilde{\beta}_{i,n+1}^j}{1 - \tilde{\alpha}_{i,n+1}} \quad (i = \overline{1, m}; j = \overline{0, N}).$$

4.3.3. Результати обчислювальних експериментів

Чисельне моделювання мобільно-немобільної динаміки міграційного процесу в рамках розглядуваної математичної моделі виконано щодо безрозмірних змінних, які визначаються співвідношеннями (4.53). Вихідні коди розробленого програмного забезпечення наведено у [265]. Деякі з отриманих при цьому результатів графічно зображені на рис. 4.14–4.17.

Аналіз результатів обчислювальних експериментів дозволяє зробити наступні висновки [65] про особливості динаміки полів концентрацій розчинних речовин з урахуванням впливу іммобілізації в умовах плосковертикальної усталеної фільтрації ґрунтових вод з вільною поверхнею:

1. В рамках мобільно-немобільного підходу, при заданих постійних значеннях порядків дробових похідних α та γ , зі збільшенням значення параметра β спостерігається явище запізнювання формування поля концентрацій (рис. 4.14, 4.15);
2. Запізнювання формування поля концентрацій спостерігається також і у разі зростання величини параметра γ при фіксованих значеннях α і β (рис. 4.16);
3. Явище запізнювання при формуванні полів концентрацій, що спостерігається при моделюванні згідно з розглядуваним мобільно-немобільним підходом, найбільшою мірою проявляється на початковій та прилеглих до початкової стадіях розвитку міграційного процесу (рис. 4.17).

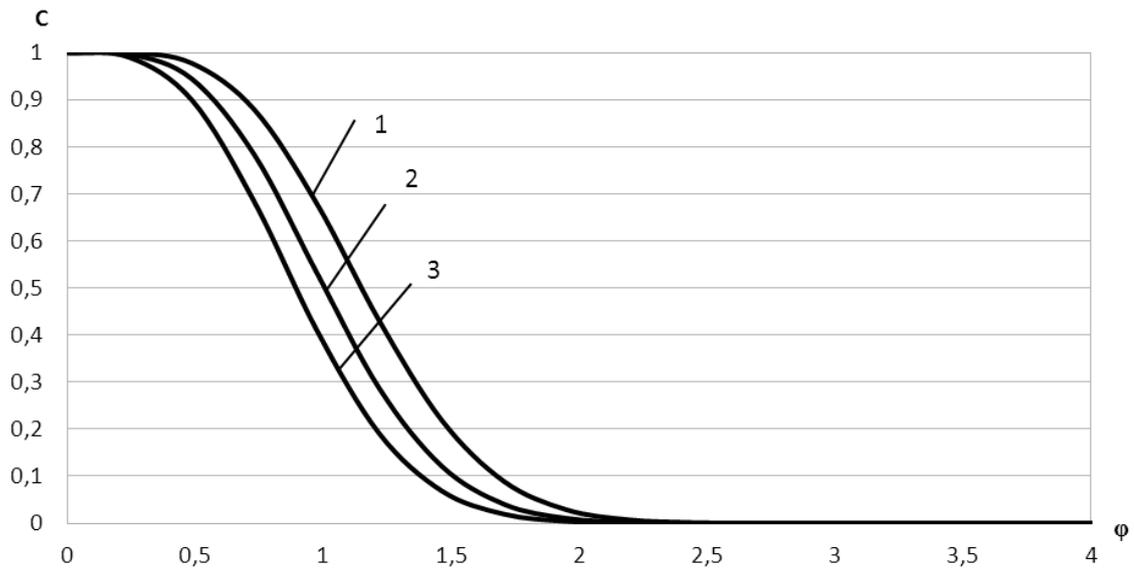


Рисунок 4.14 – Криві розподілу полів концентрацій $C(\varphi)$ в мобільній фазі уздовж лінії течії $\psi = 0,5$ в момент часу $t = 0,01$ для класичної математичної моделі при $\beta=0$ (1), $\beta=0,2$ (2), $\beta=0,4$ (3)

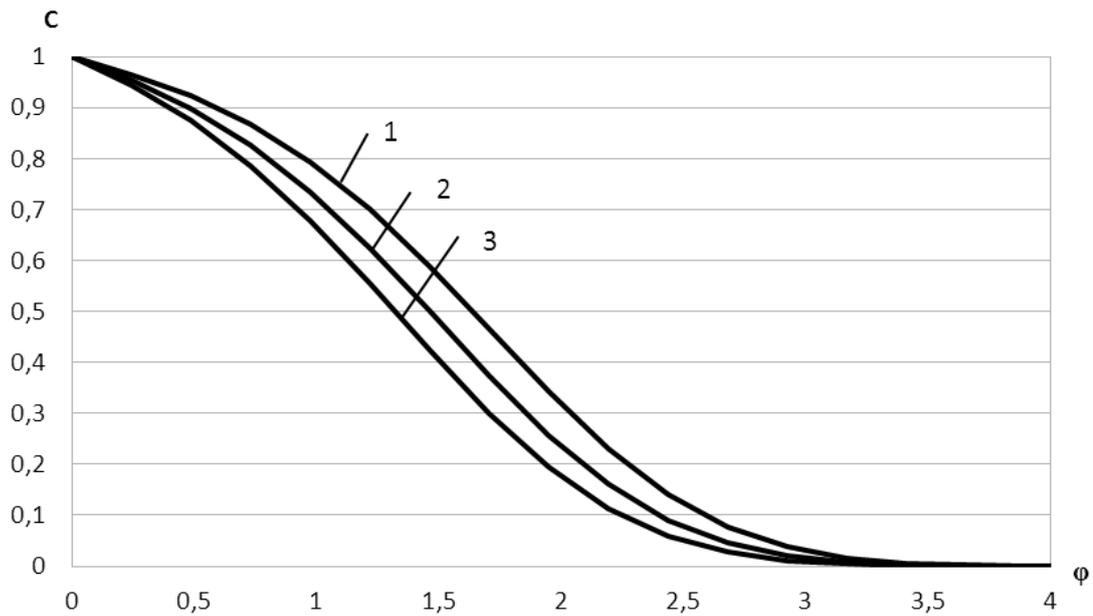


Рисунок 4.15 – Криві розподілу полів концентрацій $C(\varphi)$ в мобільній фазі уздовж лінії течії $\psi = 0,5$ в момент часу $t = 0,01$ в разі неklasичної моделі при $\alpha = 0,8, \gamma = 0,6$ і $\beta=0$ (1), $\beta=0,4$ (2), $\beta=0,8$ (3)

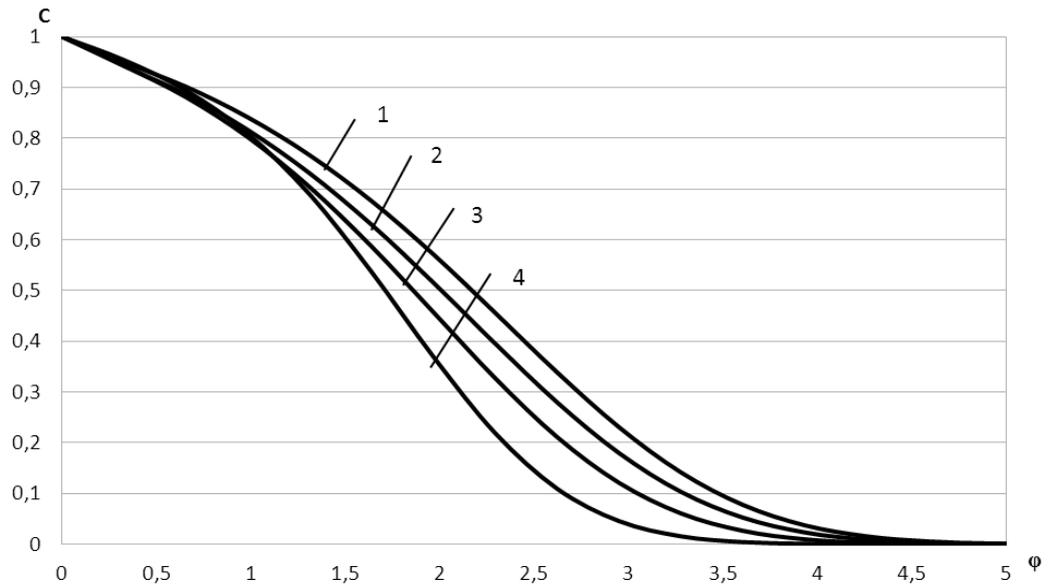


Рисунок 4.16 – Криві розподілу полів концентрацій $C(\varphi)$ в мобільній фазі уздовж лінії течії $\psi = 0,5$ в момент часу $t = 0,01$ в разі неklasичної моделі при $\alpha = 0,7$ і $\beta = 0$ (1), $\beta = 0,2$, $\gamma = 0,6$ (2), $\beta = 0,2$, $\gamma = 0,8$ (3), $\beta = 0,2$, $\gamma = 1,0$ (4)

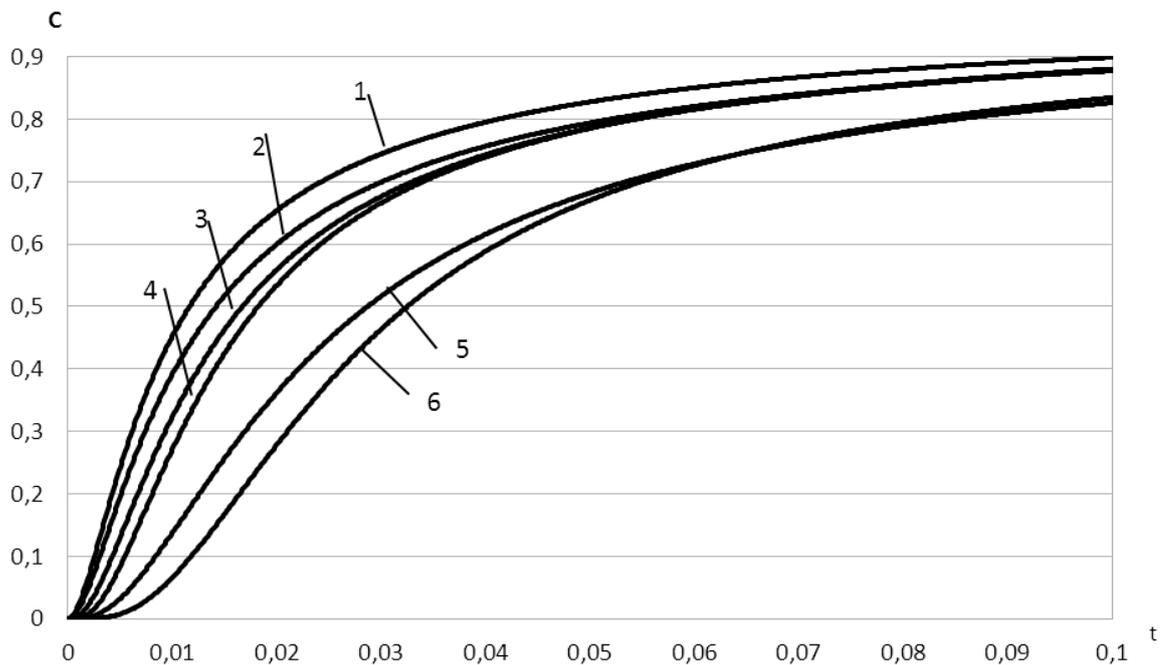


Рисунок 4.17 – Динаміка зміни концентрації $C(t)$ в мобільній фазі в точці $\varphi = 0,5$, $\psi = 0,5$ при $\alpha = 0,7$ і $\beta = 0$ (1), $\beta = 0,2$, $\gamma = 0,6$ (2), $\beta = 0,2$, $\gamma = 0,8$ (3), $\beta = 0,2$, $\gamma = 0,9$ (4), $\beta = 0,6$, $\gamma = 0,8$ (5), $\beta = 0,6$, $\gamma = 0,9$ (6)

4.4. Моделювання дробово-диференціальної динаміки процесу конвективної дифузії в умовах масообміну на основі спрощеної математичної моделі

4.4.1. Математична модель та початково-крайова задача

Будемо розглядати математичну модель процесу конвективної дифузії з урахуванням масообміну, що базується на рівнянні (4.46). Використання таких моделей в інженерній практиці з метою розробки надійних методів прогнозування динаміки міграції розчинних речовин в умовах профільної фільтрації ґрунтових вод пов'язане з певними математичними труднощами і призводить до ускладнення процесу моделювання. У ряді випадків одним з ефективних підходів для прискорення і спрощення процесу проведення оціночних інженерних розрахунків тут може стати підхід, що базується на спрощенні вихідної моделі шляхом апроксимації дробових похідних, що до неї входять. Використовуючи апроксимацію дробової похідної [6]

$${}^c D_t^\alpha u(t) \approx \alpha u'(t) + (1 - \alpha)(u(t) - u(0)),$$

отримуємо з (4.46) рівняння спрощеної математичної моделі конвективно-дифузійного процесу перенесення в умовах впливу іммобілізації у вигляді

$$a C_t'(x, y, t) = \mathcal{L}(C) - b(C(x, y, t) - C(x, y, 0)), \quad (4.63)$$

де $a = \alpha + \beta\gamma$, $b = 1 - \alpha + \beta(1 - \gamma)$.

В умовах плоско-вертикальної усталеної фільтрації ґрунтових вод з водою до заглибленого напівнескінченного дренажного каналу (пп. 4.3.1, рис. 4.13), у безрозмірних змінних (4.53) відповідна крайова задача у області комплексного потенціалу течії G_ω має вигляд (4.54)–(4.56).

У відповідній крайовій задачі, поставленій у рамках спрощеної моделі, що базується на співвідношенні (4.63), рівняння (4.54) прийме вигляд [51]

$$\frac{\partial C}{\partial t} = \tilde{\nu}^2(\varphi, \psi) \left(d \left(\frac{\partial^2 C}{\partial \varphi^2} + \frac{\partial^2 C}{\partial \psi^2} \right) - \frac{\partial C}{\partial \varphi} \right) - \nu(C - C_0), \quad (4.64)$$

$$\tilde{\nu}^2 = \frac{\nu^2(\varphi, \psi)}{\alpha + \beta\gamma}, \quad \nu = \frac{1 - \alpha + \beta(1 - \gamma)}{\alpha + \beta\gamma}, \quad C_0 = C(\varphi, \psi, 0)$$

при виконанні умов (4.55), (4.56).

4.4.2. Чисельне розв'язання крайової задачі

Аналогічно описаному у пп. 4.3.2, вводячи до розгляду сіткову область

$$\omega_h = \left\{ (\varphi_i, \psi_k, t_j) : \varphi_i = ih_1 \left(i = \overline{0, m+1} \right), \psi_k = (k - 0,5)h_2 \left(k = \overline{0, n+1} \right), \right. \\ \left. t_j = j\tau \left(j = \overline{0, N+1} \right) \right\},$$

де h_1, h_2, τ – кроки сітки за геометричними змінним φ, ψ та часом t , відповідно, поставимо у відповідність (4.64) наступний аналог [51] локально-одновимірної [313] різницевої схеми О. А. Самарського:

$$\frac{1}{2} \bar{C}_t = \tilde{\nu}^2 \left(d\bar{C}_{\varphi\varphi} - \bar{C}_{\varphi} \right) - \frac{\nu}{2} (C - C_0), \quad (4.65)$$

$$\frac{1}{2} \hat{C}_t = \tilde{\nu}^2 d\hat{C}_{\psi\psi} - \frac{\nu}{2} (C - C_0), \quad (4.66)$$

де

$$\hat{C} = C^{j+1}, \bar{C} = C^{j+\frac{1}{2}}, C = C^j, t_{j+\frac{1}{2}} = t_j + \frac{\tau}{2},$$

$$\bar{C}_t = \frac{\bar{C} - C}{\tau/2}, \quad \hat{C}_t = \frac{\hat{C} - \bar{C}}{\tau/2}, \quad C_{\varphi}, C_{\varphi\varphi}, C_{\psi\psi} - \text{центральна і другі різницеві похідні за } \varphi$$

та ψ , відповідно [313].

Розписуючи в (4.65) різницеві оператори і зводячи подібні члени, отримуємо на напівцілому часовому шарі $t_{j+\frac{1}{2}}$ систему рівнянь

$$A_{ik} C_{i+1,k}^{j+\frac{1}{2}} - B_{ik} C_{ik}^{j+\frac{1}{2}} + S_{ik} C_{i-1,k}^{j+\frac{1}{2}} = \Phi_{ik}^j \left(i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N} \right), \quad (4.67)$$

де

$$A_{ik} = \frac{\tilde{\nu}_{ik}^2}{h_1} \left(\frac{d}{h_1} - 0,5 \right), \quad S_{ik} = \frac{\tilde{\nu}_{ik}^2}{h_1} \left(\frac{d}{h_1} + 0,5 \right), \quad B_{ik} = \frac{1}{\tau} + A_{ik} + S_{ik},$$

$$\Phi_{ik}^j = \left(\frac{\nu}{2} - \frac{1}{\tau} \right) C_{ik}^j - \frac{\nu}{2} C_{0_{ik}}.$$

На цілому часовому шарі t_{j+1} з (4.66) знаходимо

$$P_{ik} C_{i,k+1}^{j+1} - Q_{ik} C_{ik}^{j+1} + P_{ik} C_{i,k-1}^{j+1} = \Omega_{ik}^j \left(i = \overline{1, m}; k = \overline{1, n}; j = \overline{0, N} \right), \quad (4.68)$$

де

$$P_{ik} = \frac{\tilde{\nu}_{ik}^2 d}{h_2^2}, \quad Q_{ik} = \frac{1}{\tau} + 2P_{ik},$$

$$\Omega_{ik}^j = \frac{\nu}{2} C_{ik}^j - \frac{1}{\tau} C_{ik}^{j+1/2} - \frac{\nu}{2} C_{0_{ik}}.$$

Різницеві рівняння систем (4.67), (4.68) є триточковими і, з використанням різницевих аналогів крайових умов (4.55), (4.56), ефективно розв'язуються методом прогонки [192, 313]. Перехід з області комплексного потенціалу течії G_ω у фізичну область G_z виконується згідно зі співвідношеннями, наведеними у [300].

4.4.3. Результати обчислювальних експериментів

Чисельне моделювання динаміки міграційного процесу в рамках вихідної МІМ-моделі (п. 4.3) та описаної у пп. 4.4.1 спрощеної математичної моделі, а також порівняльний аналіз результатів моделювання відповідно до обох зазначених моделей, виконані щодо безрозмірних змінних (4.53) за умови $C_0(\varphi, \psi) \equiv 0$. Вихідні коди розробленого програмного забезпечення наведено у [265].

Деякі з отриманих при цьому результатів для $\kappa = 0,57$ м/добу, $H = 0,45$ м, $Q = 6$ м²/добу, $d = 0,1$ м²/добу, $C_1 = \varphi_0 = 1$, $\tau = 0,0005$; $n = m = 30$, графічно зображені на рис. 4.18– 4.20.

Аналіз результатів численних експериментів дозволяє зробити наступні висновки [51]:

1. Результати моделювання полів концентрацій розчинних речовин в плоско-вертикальному фільтраційному потоці в рамках спрощеної математичної моделі якісно добре узгоджуються з відповідними результатами, отриманими в рамках вихідної дробово-диференціальної МІМ-моделі (рис. 4.18);

2. Задовільна в багатьох випадках при проведенні оціночних розрахунків в інженерній практиці точність моделювання динаміки міграційного процесу на основі спрощеної математичної моделі в порівнянні з вихідною МІМ-моделлю ілюструється наступним прикладом розрахунку: виконання нерівності

$$E_{RE}^{(\alpha, \gamma)} < 0.3,$$

де $E_{RE}^{(\alpha,\gamma)} = \frac{\|C - C_s\|}{\|C\|}$ – відносна L^2 – похибка відхилення, C – концентрація,

відповідна МІМ-моделі, а C_s – спрощеній моделі, $\|\cdot\|$ – норма в L^2 , у проведених розрахунках досягалося при $(\alpha, \gamma) \in [0.85, 1] \times [0.88, 1]$, а нерівності $E_{RE}^{(\alpha,\gamma)} < 0.2$ – при $(\alpha, \gamma) \in [0.94, 1] \times [0.97, 1]$ (рис. 4.19, 4.20). При цьому, відносні L^2 – похибки відхилення результатів розрахунків за вихідною МІМ-моделлю від результатів, отриманих в рамках спрощеної математичної моделі, швидко зменшуються при наближенні значень порядків похідних α, γ до одиниці: $\alpha, \gamma \uparrow 1 \Rightarrow E_{RE}^{(\alpha,\gamma)} \downarrow E_{RE}^{(1,1)} = 0$.

В цілому ж, залежність отриманих в проведених розрахунках значень відносної L^2 – похибки від значень порядків похідних α, γ емпірично (с максимальною абсолютною похибкою, що дорівнює 0.07) представляється у вигляді

$$E_{RE}^{(\alpha,\gamma)} = \begin{pmatrix} 1 \\ \gamma \\ \gamma^2 \end{pmatrix}^T \begin{pmatrix} -0,14 & 1,71 & -1,64 \\ 1,64 & -2,63 & 1,97 \\ -1,56 & 1,70 & -1,02 \end{pmatrix} \begin{pmatrix} 1 \\ \alpha \\ \alpha^2 \end{pmatrix};$$

3. Обчислювальна складність проведення розрахунків для вихідної МІМ-моделі лінійно залежить від номера кроку за часом, тоді як для спрощеної моделі час проведення обчислень на одному кроці постійний. Таким чином, використання спрощеної моделі лінійно прискорює процес моделювання.

Отримані дані свідчать про те, що спрощена математична модель розглядуваного геоміграційного процесу в ряді випадків може ефективно використовуватись для проведення оціночних розрахунків в інженерній практиці при розробці конструктивних рішень в області проектування екологічно небезпечних інженерних об'єктів.

4.5. Алгоритм вибору дробово-диференціальної моделі конвективної дифузії з урахуванням масообміну

Оскільки точне визначення значень параметрів дробово-диференціальних моделей є наразі відкритою проблемою, їх підбір таким чином, щоб модель якнайточніше описувала наявні вимірювання, залишається у багатьох випадках

єдиним підходом при практичному застосуванні. Для цього використовують, зокрема метаевристичні методи, які є обчислювально складними. Оскільки тип масообмінних процесів переважно також невідомий, питання вибору моделі додатково збільшує складність дослідження дифузійних процесів, в тому числі динаміки розповсюдження забруднень, яка у випадку техногенних аварій має прогнозуватись оперативно. У такій ситуації, важливої є розробка наближених алгоритмів прийняття рішень, які дозволяють проводити вибір моделі, яка є найточнішою для спостережуваного процесу.

Як показують результати порівнянь, наведені у [208], при якнайкраще підібраних значеннях параметрів, розв'язки за різними дробово-диференціальними моделями дифузійних процесів, які є близькими на початкових стадіях розвитку процесу, стають суттєво відмінними при подальшому його розвитку. У зв'язку з цим, актуальним є дослідження якісних відмінностей аномальних дифузійних процесів, описуваних різними моделями, розв'язки яких на початковому етапі відрізняються один від одного в межах похибки вимірювання, з метою визначення найбільш адекватної моделі в процесі спостереження за зміною полів концентрації.

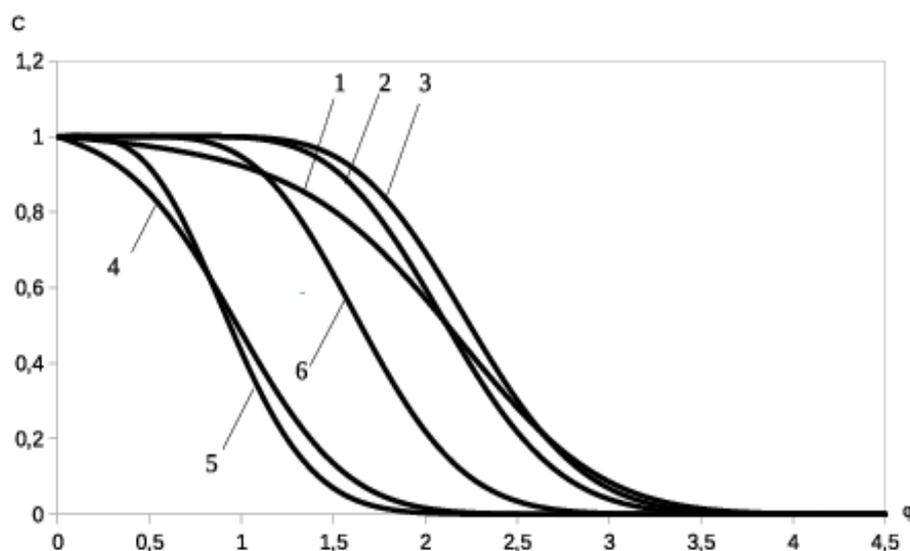


Рисунок 4.18 – Криві концентрації вздовж лінії течії $\psi = 0,5$ при $t = 4,2$ год у випадку МІМ-моделі (криві 1,4), спрощеної (криві 2,5) та класичної (криві 3,6) моделей (1,2 – $\alpha=\gamma=0,92$, $\beta=0,1$; 3 – $\alpha=\gamma=1$, $\beta=0,1$; 4,5 – $\alpha=0,84$; $\gamma=0,96$; $\beta=0,8$; 6 – $\alpha=\gamma=1$, $\beta=0,8$)

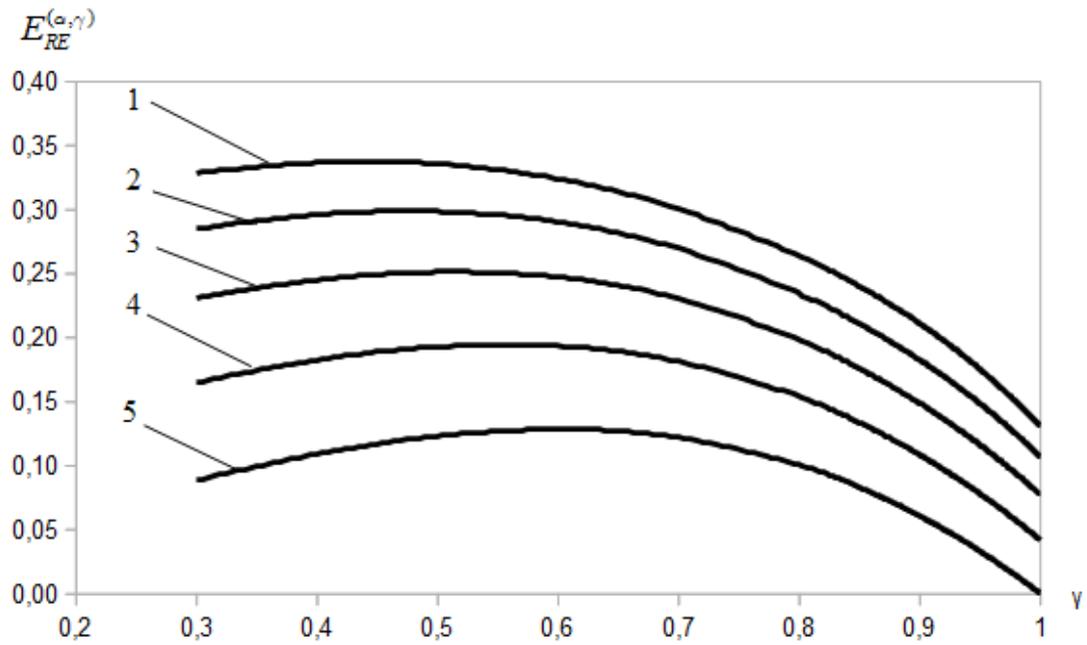


Рисунок 4.19 – Криві відносної L^2 – похибки відхилення $E_{RE}^{(\alpha, \gamma)} = \frac{\|C - C_s\|}{\|C\|}$, де

C – концентрація, відповідна МІМ-моделі, а C_s – спрощеної моделі, $\|\cdot\|$ – норма

в L^2 (1 – $\alpha=0,8$; 2 – $\alpha=0,85$; 3 – $\alpha=0,9$; 4 – $\alpha=0,95$; 5 – $\alpha=1$; 1 – 5: $\beta=0,8$; $t=0,01$)

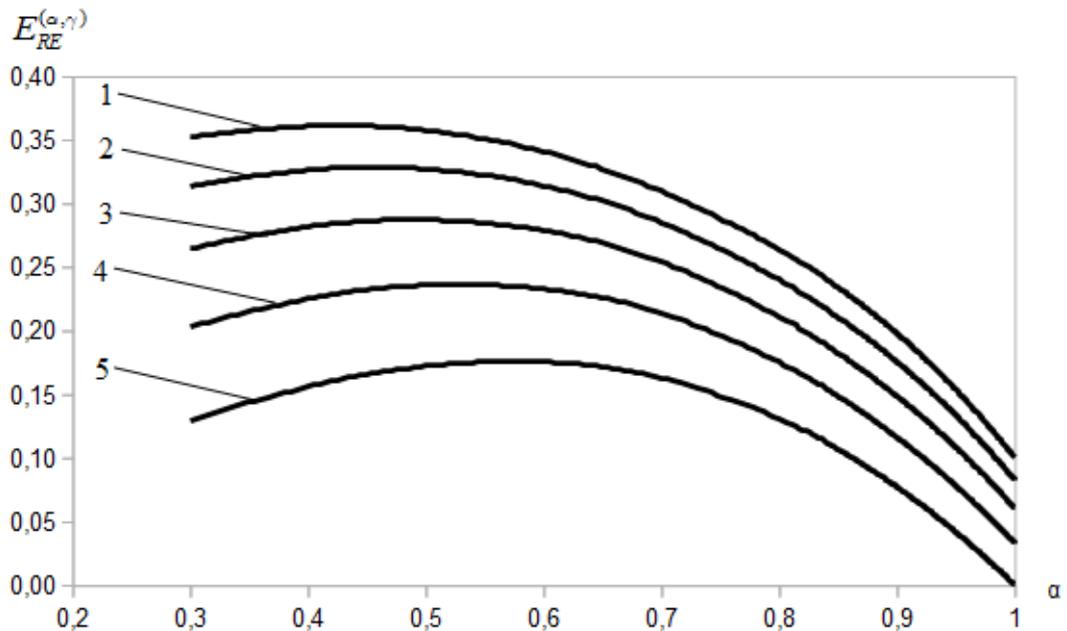


Рисунок 4.20 – Криві відносної L^2 – похибки відхилення $E_{RE}^{(\alpha, \gamma)} = \frac{\|C - C_s\|}{\|C\|}$, де

C – концентрація, відповідна МІМ-моделі, а C_s – спрощеної моделі, $\|\cdot\|$ – норма

в L^2 (1 – $\gamma=0,8$; 2 – $\gamma=0,85$; 3 – $\gamma=0,9$; 4 – $\gamma=0,95$; 5 – $\gamma=1$; 1 – 5: $\beta=0,8$; $t=0,01$)

4.5.1. Початково-крайові задачі та чисельний метод

Двовимірне дробово-диференціальне рівняння конвективної дифузії розчинних речовин в умовах масообміну представимо у вигляді [221]

$$\sigma D_t^{(\beta)} C + \frac{\partial N}{\partial t} = LC, L = D\Delta_{xy} C - v_x \frac{\partial C}{\partial x} - v_y \frac{\partial C}{\partial y} \quad (4.69)$$

де C – концентрація речовини у рідкій фазі, N – концентрація речовини у твердій (немобільній) фазі, σ – пористість, $v_x = v_x(x, y)$, $v_y = v_y(x, y)$ – складові вектору швидкості фільтрації, $D = const$ – коефіцієнт конвективної дифузії, Δ_{xy} – оператор Лапласа за змінними x, y , $D_t^{(\beta)}$ – похідна Капуто порядку β ($0 < \beta < 1$) за часовою змінною, $\Gamma(z)$ – гамма-функція Ейлера.

Будемо розглядати два варіанти моделювання динаміки відтоку частинок у немобільну фазу. Так, згідно з [201]

$$\frac{\partial N}{\partial t} = \bar{\beta} D_t^{(\gamma)} C, 0 < \gamma < 1, \quad (4.70)$$

де $\bar{\beta}$ – параметр швидкості масообміну.

Друга розглядувана модель отримується додаванням до рівняння (4.69) рівняння нерівноважної сорбції розчинної речовини при ізотермі Генрі (4.20) [311].

Аналогічно [201], розв'язуючи (4.20) відносно N за початкових умов $C|_{t=0} = N|_{t=0} = 0$ отримуємо

$$N(t) = \beta_* \sigma \int_0^t C_3(t - \tau) e^{-\alpha_* \beta_* \tau} d\tau,$$

$$\frac{\partial N}{\partial t} = \beta_* \sigma \int_0^t C_3'(\tau) e^{-\alpha_* \beta_* (t - \tau)} d\tau = \beta_* \sigma (1 - \chi)^{CFC} D_t^\chi C, \chi = \frac{\alpha_* \beta_*}{1 + \alpha_* \beta_*},$$

де ${}^{CFC} D_t^\chi C$ – дробова похідна Капуто–Фабріціо [75] порядку χ при $B(\chi) = 1$.

Таким чином, можна відзначити, що головною відмінністю між рівняннями масообміну (4.70) та (4.20) є форма ядра інтегрального оператора — степенева у рівнянні (4.70) та експоненційна у рівнянні (4.20).

Для спрощення опису будемо використовувати фільтраційну задачу, що має аналітичний розв'язок. Нехай конвективна дифузія забруднень з водою

відбувається у області G_z наведеній у [62] (рис. 4.13, а). Для такої схеми відома область комплексного потенціалу течії G_ω , $\omega = \varphi + i\psi$ (ψ – функція течії), що має вигляд горизонтальної напівсмуги [62] (рис. 4.13, б), та розв'язок відповідної фільтраційної задачі [300].

Задаючи на вході фільтраційного потоку AC концентрацію розчиненої речовини C_0 , крайові та початкові умови для моделей (4.69), (4.70) та (4.69), (4.20) запишемо у вигляді

$$C|_{AC} = C_0, \quad \left. \frac{\partial C}{\partial n} \right|_{AB, CB} = 0,$$

$$C|_{t=0} = N|_{t=0} = 0,$$

де n – зовнішня нормаль до відповідної кривої, AB – вісь симетрії потоку, CB – лінія течії.

Чисельне розв'язання початково-крайових задач для модельних систем рівнянь (4.69), (4.70) та (4.69), (4.20) будемо виконувати згідно з локально-одновимірними схемами [313] наведеними, відповідно, у [38] та [65]. При цьому використовується рівномірна скінченно-різницева сітка, що покриває обмежену справа прямою $\varphi = \varphi_0$ ($\varphi_0 \gg 1$) область комплексного потенціалу течії; скінченно-різницевий аналог оператора дробового диференціювання першого порядку точності; метод прогонки [313] для розв'язання одержуваних після дискретизації тридіагональних СЛАР.

4.5.2. Алгоритм прийняття рішень при прогнозуванні

Модельні системи рівнянь (4.69), (4.70) та (4.69), (4.20) описують дифузійні процеси з різними механізмами масообміну — нелокальним за часом, проте лінійним, у випадку рівняння (4.70) та локальним, проте нелінійним, у випадку рівняння (4.20). При моделюванні реальних процесів, необхідно проводити калібрування моделей визначаючи значення їх параметрів на основі наявних вимірювань, які не є точними. Враховуючи, що механізм масообміну у конкретному випадку часто є невідомим, постає також задача визначення моделі, яка найточніше описує вимірюваний процес. У зв'язку з цим, розглянемо питання за яких умов

обидві моделі в межах відомої точності, описують той самий набір вимірювань, отриманий у конкретний момент часу, тобто коли розв'язки, отримувані за модельними системами рівнянь (4.69), (4.70) та (4.69), (4.20), відрізняються не більше ніж на задану величину.

Побудуємо оцінки максимального відхилення розв'язків систем рівнянь (4.69), (4.70) та (4.69), (4.20) від розв'язку C_1 рівняння дифузії (4.69) без врахування масообміну.

Нехай $C_2 = C_1 + \omega_1$ задовольняє рівнянням (4.69), (4.70). Тоді, віднімаючи від рівняння (4.69) для C_2 відповідне рівняння без врахування масообміну для C_1 , отримуємо

$$\bar{\beta} D_t^{(\gamma)} C_2 = (L - \sigma D_t^{(\beta)}) \omega_1.$$

За припущення, що $|(L - \sigma D_t^{(\beta)}) \omega_2| \geq M_1 \max |\omega_1|$ та $\partial C_2 / \partial t \leq M_2$ й, відповідно, $D_t^{(\gamma)} C_2 \leq M_2 \frac{t^{1-\gamma}}{\Gamma(2-\gamma)}$, верхня оцінка максимального відхилення має вигляд [35]

$$\max |\omega_1| \leq \bar{\omega}_1 = M_3 K_1, K_1 = \frac{\bar{\beta} t^{1-\gamma}}{\Gamma(2-\gamma)}, M_3 = M_2 / M_1. \quad (4.71)$$

Для розв'язку $C_3 = C_1 + \omega_2$ системи рівнянь (4.69), (4.20) маємо

$$\frac{\partial N}{\partial t} = (L - \sigma D_t^{(\beta)}) \omega_2.$$

За припущення $|(L - \sigma D_t^{(\beta)}) \omega_2| \geq M_1 \max |\omega_2|$ та $\partial C_3 / \partial t \leq M_2$, верхня оцінка максимального відхилення матиме вигляд [35]

$$\max |\omega_2| \leq \bar{\omega}_2 \approx M_3 K_2, K_2 = \frac{\sigma}{\alpha_*} (1 - e^{-\alpha_* \beta t}). \quad (4.72)$$

Оскільки при врахуванні масообміну спостерігається запізнення розповсюдження фронту концентрації у порівнянні з ситуацією моделювання дифузії без врахування масообміну [38, 65], максимальне відхилення $d(C_2, C_3)$ між розв'язками за (4.69), (4.70) та (4.69), (4.20) можна оцінити як $d(C_2, C_3) \approx |\bar{\omega}_2 - \bar{\omega}_1|$.

При необхідності прогнозувати спостережуваний аномальний дифузійний процес в ситуації, коли невідомий фактичний механізм масообміну й, відповідно, неможливо апріорі визначити найбільш адекватну йому математичну модель, пропонується наступний алгоритм прийняття рішень [35]:

- маючи вимірювання концентрації C моменти часу $\tau_i, i = 1, \dots, n$, провести калібрування моделі, що базується на рівняннях (4.69), (4.70), тобто підібрати, наприклад метаевристичним оптимізаційним методом, значення параметрів, за яких вона якнайтоніше описує наявні вимірювання;

- Провести моделювання при визначених значеннях параметрів та при $\bar{\beta} = 0$ та отримати значення максимальних абсолютних відхилень $\max |\omega_1(\tau_i)|, i = 1, \dots, n$. Використовуючи оцінку (4.71) знайти значення M_3 ;

- Розв'язавши систему нелінійних рівнянь $M_3 K_2(\tau_i) = \max |\omega_1(\tau_i)|, i = 1, \dots, n$, наприклад градієнтним методом, отримати значення параметрів модельних рівнянь (4.69), (4.20), які згідно з оцінкою (4.72) дозволяють описувати процес, близький до спостережуваного;

- Провести прогнозне моделювання за (4.69), (4.70) та (4.69), (4.20) для $t > \tau_n$;

- При отриманні наступної серії вимірювань, вибрати для подальшого прогнозування ту з моделей, яка точніше її описує.

Використання такого алгоритму дозволяє визначити адекватнішу спостережуваному процесу математичну модель, зменшуючи часові витрати на розв'язання обернених задач для калібрування.

4.5.3. Результати обчислювальних експериментів

Знайдемо значення параметрів β^* та α^* моделі, що базується на рівняннях (4.69), (4.20), і оцінку константи M_3 згідно з вищеописаним алгоритмом для фіксованих значень параметрів $\bar{\beta}$ та γ моделі, що базується на рівняннях (4.69), (4.70), використовуючи розв'язки на певному наборі моментів часу та дослідимо яким чином в такій ситуації відмінність між розв'язками поводитиме себе в інші моменти часу.

Будемо отримувати базові розв'язки при $\beta=0,8$; $\sigma=0,8$; $L=22$; $H=1/\sqrt{5}$; $\bar{k}=2\sqrt{5}/(2,5\pi)$; $\varphi_0=5$; $m=20$; $n=30$; $\tau=0,0001$ для $t=0,001;0,002;\dots;0,01$. Визначенні значення β^* та α^* , що дозволяють отримувати розв'язки, близькі до розв'язків за моделлю (4.69), (4.70), подані у табл. 4.1.

Таблиця 4.1 – Значення коефіцієнтів моделей

$\bar{\beta}$	γ	α^*	β^*	M_3
0,2	0,6	23,03	30	1,35
0,6	0,6	7,24	103	1,22
0,2	1,0	3,75	449	0,96
0,7	0,8	2,76	606	0,95
0,6	1,0	1,3	2876	0,72

Розглядаючи залежності значень коефіцієнтів моделей від максимальних абсолютних відхилень (“запізень”) від розв'язку задачі без врахування масообміну, зауважимо, що для моделі, що базується на рівняннях (4.69), (4.20), необхідне для опису процесу значення β^* експоненційно збільшується зі збільшенням запізнення, а значення α^* при цьому має зменшуватись з порядком $x^{-1,28}$. Коефіцієнт детермінації для обох залежностей тут більше за $R^2=0,94$. Параметри моделі, що базується на рівняннях (4.69), (4.70), настільки чітко не корелюють з максимальними абсолютними відхиленнями від розв'язку без врахування масообміну, проте для їх обох залежності близькі до лінійних. Це в цілому показує більшу доцільність використання моделі, що базується на рівняннях (4.69), (4.70), з позиції збіжності алгоритмів розв'язання обернених задач ідентифікації параметрів через те, що чутливість її параметрів до зміни вихідних даних є близькою до лінійної.

Максимальні відхилення $\bar{\omega}_1$, $\bar{\omega}_2$ розв'язків за модельними системами рівнянь (4.69), (4.70) та (4.69), (4.20) від розв'язку за моделлю, у якій не враховується масообмін, та їх оцінки згідно з (4.71), (4.72) у різні моменти часу наведено для випадку $\bar{\beta}=0,7$; $\gamma=0,8$ ($\beta_*=606$, $\alpha_*=2,764$) на рис. 4.21. Зміну з часом різниці між розв'язками за модельними системами рівнянь (4.69), (4.70)

та (4.69), (4.20) за різних значень параметрів та їх оцінку згідно з (4.71), (4.72) наведено на рис. 4.22.

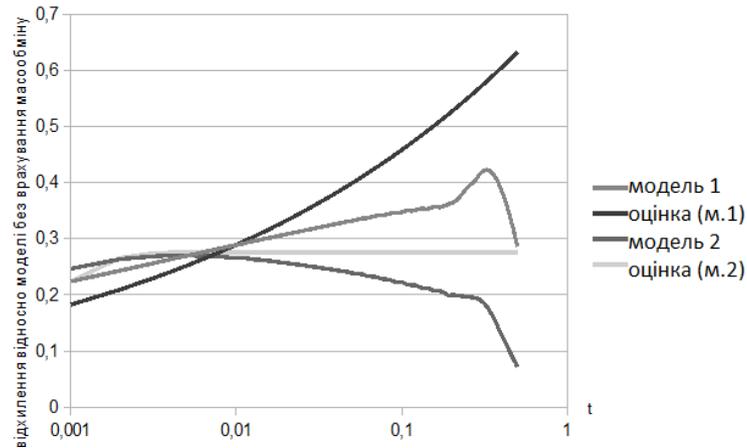


Рисунок 4.21 – Зміна з часом значень максимальних абсолютних відхилень для моделі на основі систем рівнянь (4.69), (4.70) (модель 1) та (4.69), (4.20) (модель 2) і їх оцінок при $\bar{\beta} = 0,7$; $\gamma = 0,8$ ($\beta_* = 606$, $\alpha_* = 2,764$)

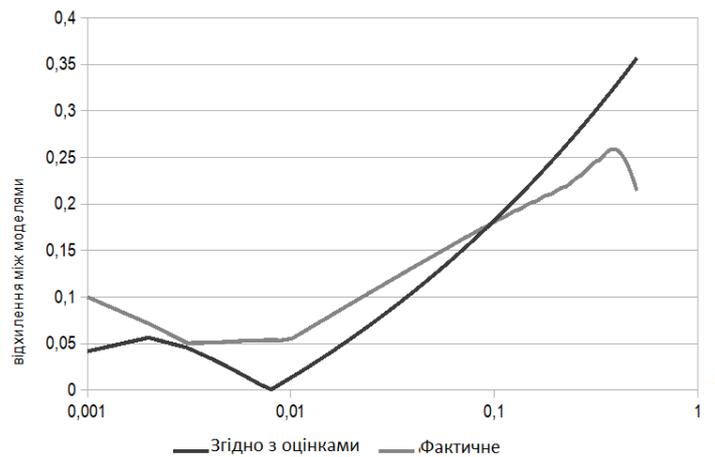


Рисунок 4.22 – Різниця між розв'язками згідно з модельними системами рівнянь (4.69), (4.70) та (4.69), (4.20) при $\bar{\beta} = 0,7$; $\gamma = 0,8$ ($\beta_* = 606$, $\alpha_* = 2,764$)

Загалом адекватно описуючи тенденції змін різниці між розв'язками, отриманих за розглядуваними моделями, оцінки (4.71), (4.72) показують, що при $\gamma \neq 1$ та $t > 0,01$ модель, заснована на системі рівнянь (4.69), (4.70), описує процес швидшого наближення поля концентрації до стаціонарного стану, ніж модель, що базується на рівняннях (4.69), (4.20). При $\gamma = 1$ відмінності між розв'язками незначні.

Щодо швидкодії проведення обчислень, зазначимо, що у обчислювальних експериментах, у порівнянні з моделюванням процесу конвективної дифузії без врахування масообміну, час, витрачений на розрахунки при $t=0,01$ за моделлю на основі системи рівнянь (4.69), (4.20), був більшим на $\sim 6\%$. У випадку моделі (4.69), (4.70), він був більшим на $\sim 12\%$ при $\gamma=1$ та на $\sim 31\%$ при $\gamma \neq 1$. Швидкість збіжності скінченно-різницевої схеми для моделі згідно з (4.69), (4.20) при зменшенні τ так само була вищою, ніж у випадку моделі на основі (4.69), (4.70).

Підсумовуючи вищезазначене, можна зробити наступні висновки [35]:

- моделі згідно з (4.69), (4.70) та згідно з (4.69), (4.20) дозволяють описувати аномальні дифузійні процеси, що, в межах заданої точності вимірювань, співпадають на певному інтервалі. При подальшому розвитку процесу, модель на основі (4.69), (4.70) описує його швидше наближення до стаціонарного стану при $\gamma \neq 1$;
- залежності параметрів моделі на основі (4.69), (4.70) від рівня запізнення у порівнянні з розв'язком за моделлю без врахування масообміну є більш рівномірними ніж для моделі згідно з (4.69), (4.20), роблячи модель на основі (4.69), (4.70) кращім вибором з позиції збіжності алгоритмів розв'язання обернених задач ідентифікації параметрів;
- швидкодія скінченно-різницевої схеми для моделі на основі (4.69), (4.20) є більшою за випадок моделі на основі (4.69), (4.70) внаслідок меншої обчислювальної складності та більшої швидкості збіжності.

4.6. Висновки до розділу 4

Розділ 4 присвячено математичному моделюванню нелокальних конвективно-дифузійних процесів при врахуванні явищ масообміну між частинками у мобільній та немобільній фазах згідно з різними законами кінетики масообміну як у одновимірному, так і у двовимірному наближенні.

У п. 4.1 сформульовано одновимірну дробово-диференціальну модель процесу промивки засолених ґрунтів з урахуванням масообміну, що містить дробово-диференціальне за просторовою змінною рівняння для фільтраційного напору та

дробово-диференціальне за часовою змінною рівняння для поля концентрації солей у розчинній фазі. Щодо цієї моделі поставлено початково-крайову задачу та отримані чисельні розв'язки методом скінченних різниць. Результати обчислювальних експериментів демонструють, що процес розсолення помітно прискорюється при зменшенні величини порядку дробової похідної за просторовою змінною у рівнянні для поля напорів у порівнянні з описом процесу в рамках класичної моделі. Зменшення величини порядку дробової похідної за часом у рівнянні стосовно поля концентрації солей призводить до запізнювання розвитку процесу розсолення. Врахування масообміну з вміщуючими породами сприяє уповільненню процесу розсолення.

У п. 4.2 побудована та досліджена двовимірна нелокальна дробово-диференціальна модель конвективної дифузії розчинних речовин за умов масообміну, що містить похідну Капуто–Катугампола за часовою змінною. Кінетика масообміну у розглядуваній моделі описується згідно з рівнянням нерівноважної оборотної сорбції при ізотермі Генрі. Розглядається також випадок нелінійної залежності коефіцієнта дифузії від концентрації солей у рідкій фазі. Результати чисельного розв'язання відповідної початково-крайової задачі згідно з лінеаризованою локально-одновимірною скінченно-різницевою схемою показують, що варіація додаткового параметра похідної Капуто–Катугампола дозволяє отримувати як субдифузійну, так і супердифузійну картини розподілу полів концентрації. Ступінь впливу масообміну на динаміку міграційного процесу тут залежить від величини порядку дробової похідної за часовою змінною. Так, зменшення величини запізнювання спостерігається в разі зменшення порядку похідної і навпаки. Запізнення у розвитку фронту концентрацій при наявності масообміну також суттєво залежить від параметрів рівняння нерівноважної оборотної сорбції.

У п. 4.3 будується математична модель і розв'язується двовимірна задача дробово-диференціального моделювання мобільно-немобільної динаміки процесу конвективної дифузії розчинних речовин при усталеній фільтрації ґрунтових вод до дренажу. Рівняння моделі отримується на основі МІМ (mobile-immobile media)

підходу використовуючи рівняння динаміки відтоку частинок у немобільну фазу, що містить похідну Капуто за часовою змінною. При моделюванні згідно з таким підходом, явище запізнювання формування полів концентрації спостерігається найбільшою мірою на початковій та прилеглих до початкової стадіях розвитку міграційного процесу.

У п. 4.4 розглядається спрощений аналог математичної моделі, описаної у п. 4.3, який отримується шляхом апроксимації похідної дробового порядку похідними цілого порядку. Використання моделі такого типу дозволяє суттєво знизити час, що витрачається на моделювання, а результати обчислювальних експериментів показують, що вона із задовільною точністю описує відповідний конвективно-дифузійний процес при значенні порядку дробової похідної, більшому за 0.85.

Алгоритм вибору найбільш адекватної спостережуваному процесу дробово-диференціальної моделі конвективної дифузії з масообміном описується у п. 4.5. Алгоритм побудовано на основі верхніх оцінок різниці між розв'язками задач, досліджуваних у п. 4.2 та 4.3, та задачі, що не враховує масообмінні процеси.

Результати розділу 4 опубліковано в роботах [34, 35, 38, 65, 51, 269, 278].

Джерела, що використані у розділі 4

Для написання даного розділу було використано 42 джерела [6], [11], [34], [35], [38], [51], [62], [65], [75], [85], [88], [115], [123], [134], [139], [177], [192], [201], [207], [208], [214], [221], [248], [265], [269], [271], [274], [278–281], [285], [287], [299], [300], [304], [310–313], [318], [320], посилання на які зазначені в тексті розділу.

Розділ 5. Високопродуктивні алгоритми моделювання гідрогеоміграційної динаміки на основі дробово-диференціальних моделей

Алгоритми чисельного розв'язання початково-крайових задач для інтегро-диференціальних рівнянь з дробовими похідними, розглянутих у розділах 2, 3, 4 мають вищий порядок обчислювальної складності у порівнянні з відповідними алгоритмами для моделей у класичній постановці.

Серед підходів до її зниження виділимо використання методів паралельних обчислень [34, 91, 109], застосування принципу фіксованої пам'яті [102, 110] та апроксимації ядер інтегральних операторів [14, 23].

На рівні математичних обчислювальних схем, в контексті балансування точності та складності апроксимації, принцип фіксованої пам'яті відкидає частину інформації про розв'язок на попередніх кроках за часом, тоді як принцип апроксимації ядер враховує всі розв'язки на попередніх кроках.

На алгоритмічному рівні, принцип розпаралелювання передбачає створення алгоритмів, що виконуються на специфічних обчислювальних системах, які дають можливість одночасного виконання незалежних блоків обчислень. Серед найбільш поширених паралельних обчислювальних систем, виділимо системи з розподіленою пам'яттю (кластерні системи, масово-паралельні системи), системи зі спільною пам'яттю (багатоядерні та графічні процесори), а також їх гібридні варіанти (кластерні системи з вузлами, що містять графічні процесори). Для максимально ефективного використання можливостей цих високопродуктивних систем, необхідна розробка специфічних алгоритмів, що відображають особливості математичного методу на архітектуру обчислювальної системи.

На рівні конкретних обчислювальних систем, також виникає необхідність урахування специфіки організації в них низькорівневих операцій. Алгоритми розв'язання розглянутих раніше задач через необхідність чисельного інтегрування мають високе значення відношення кількості операцій доступу до пам'яті до кількості арифметичних операцій, що робить актуальним низкорівневу їх оптимізацію в контексті підсистеми пам'яті. Так, врахування особливостей кешування може прискорити процес моделювання, як на центральному, так і на

графічних процесорах. При розпаралелюванні на останніх, виникає також необхідність врахування поділу їх пам'яті на порівняно повільну глобальну та істотно швидшу локальну, для ефективного використання якої необхідна спільна робота груп потоків у масово-багатопоточному середовищі.

У цьому розділі представлений клас високопродуктивних алгоритмів для систем з розподіленою пам'яттю та графічних процесорів призначені для розв'язання одно- та багатовимірних початково-крайових задач для моделей процесів перенесення, що містять похідну Капуто–Герасимова за часовою та просторовими змінними, а також похідну Капуто–Фабріціо.

5.1. Паралельні алгоритми схем розщеплення для систем з розподіленою пам'яттю

Розглянемо організацію паралельних обчислень на системах з розподіленою пам'яттю при розв'язанні двовимірних задач моделювання міграційних процесів на основі моделей з похідною Капуто–Герасимова за часовою змінною, зокрема, задачі, описаної у п. 2.2.

Використовувана для розв'язання таких задач локально-одновимірною схемою розщеплення може бути розпаралелена для кластерних систем [64, 269] за допомогою червоно-чорного розподілу даних по аналогії зі схемою, описаною в [267]. Кожен з N процесів паралельної програми обчислює значення прогоночних коефіцієнтів та шуканої функції для N блоків комірок сітки розміру $\left(\frac{n}{N}, \frac{m}{N}\right)$, де (n, m) – розмір сітки. Вертикальні та горизонтальні прогонки виконуються паралельно за N кроків з обміном даними після кожного з них.

Алгоритм 1 [64] знаходження розв'язку на одному кроці за часом опишемо наступним чином:

1) Обчислення прогоночних коефіцієнтів вертикальних прогонок. Процес j на i -тому кроці обробляє блок $(2N - i - j - 1) \bmod N$ та передає значення прогоночних коефіцієнтів у $\left(\frac{m}{N} - 1\right)$ -му рядку блоку процесу $(N + j - 1) \bmod N$;

2) Зворотний хід вертикальних прогонок. Процес j на i -тому кроці обробляє блок $(N+i-j)\bmod N$ та передає значення шуканої функції в першому рядку блоку процесу $(j+1)\bmod N$;

3) Обчислення прогоночних коефіцієнтів горизонтальних прогонок. Процес j на i -тому кроці обробляє блок i та передає значення прогоночних коефіцієнтів у $\left(\frac{n}{N}-1\right)$ -ому стовпці блоку процесу $(N+j-1)\bmod N$;

4) Зворотний хід горизонтальних прогонок. Процес j на i -тому кроці обробляє блок $N-i-1$ та передає значення шуканої функції у першому стовпці блоку процесу $(j+1)\bmod N$.

Час роботи алгоритму 1 можна оцінити [64] як

$$T_N(n, m) = \frac{nm}{N}t_c + 2(N-1)\left(t_{tr}\left(\frac{n}{N}\right) + t_{tr}\left(\frac{m}{N}\right)\right), \quad (5.1)$$

де t_c – сума часу, необхідного для обчислення однієї пари прогоночних коефіцієнтів, і часу обчислення розв’язку в одній точці при зворотному ході прогонки, а $t_{tr}(s)$ – час пересилання блоку, що складається з s одиниць оброблюваних даних, між процесами розподіленої програми.

Значення t_c у розглядуваній схемі лінійно залежить від номера кроку за часом:

$$t_c = t_{c0} + k_c t. \quad (5.2)$$

Твердження 5.1. За умови лінійної залежності часу t_c від номеру кроку за часом у формі (5.2) оцінка часу (5.1) роботи алгоритму 1 має вигляд

$$T_N(n, m, t) = \frac{nm}{N}(t_{c0} + k_c t) + 2(N-1)\left(t_{tr}\left(\frac{n}{N}\right) + t_{tr}\left(\frac{m}{N}\right)\right). \quad (5.3)$$

При збільшенні номера кроку за часом, основну обчислювальну складність при розв’язанні диференціальних рівнянь з дробовою похідною за часовою змінною створюють операції обчислення сум у її апроксимації. Час, затрачений на підсумовування, збільшується в процесі обчислень. На час роботи алгоритму, в цьому випадку, основний вплив має швидкодія підсистеми пам’яті та ефективність

роботи кешу. Поліпшення останньої досягається шляхом попереднього (до основного циклу визначення значень прогоночних коефіцієнтів) дворівневого часткового обчислення зазначених сум (алгоритм 1а [64]). При цьому фрагмент алгоритму, який має структуру вигляду

```
for (i = 0; i < n; i++)
  for (j = 0; j < t; j++)
    s = s + V(i, j)
```

перетворюється у наступний фрагмент:

```
for (k = 0; k < (t / G); k++)
  for (i = 0; i < n; i++)
    for (j = 0; j < G; j++)
      if ((k * G + j) < t)
        s = s + V(i, k * G + j).
```

Для алгоритмів 1 та 1а, на кластері СКІТ-4 Інституту кібернетики ім. В.М. Глушкова НАН України були проведені дві серії обчислювальних експериментів. Вихідні коди розробленого програмного забезпечення наведено у [265]. У першій серії, задачі (2.30)–(2.32) розв’язувались на одному CPU при різних розмірах сітки: 500x500, 750x750, 1000x1000 та 1500x1500. Результати тестування підтверджують лінійну залежність часу проведення обчислень на одному кроці від його номера та квадратичну залежність від розмірів сітки.

У другій серії експериментів задачі з розміром сітки 3000x3000 розв’язувались різною кількістю процесів (рис. 5.1). У цьому випадку зберігаються залежності, характерні для першої серії експериментів, а також спостерігається зниження швидкодії через неефективне використання кешу алгоритмом 1 (крива 1) та нівелювання цього негативного ефекту алгоритмом 1а (крива 6).

Прискорення паралельного алгоритму на кроках за часом разом з його оцінкою за формулою (5.3) наведено на рис. 5.2. При невеликій кількості залучених обчислювальних ресурсів ця оцінка досить точна, тоді як при їх збільшенні точність падає.

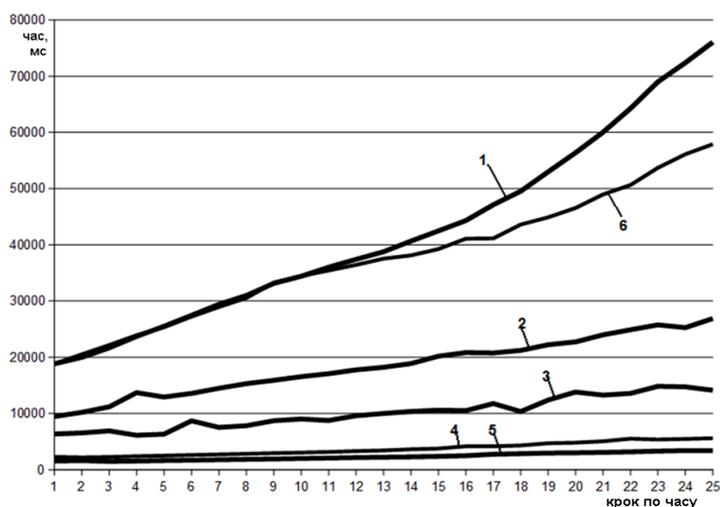


Рисунок 5.1 – Час розв’язання задачі на сітці розміром 3000x3000 з використанням алгоритму 1 (число процесів: 1 – 1, 2 – 2, 3 – 4, 4 – 8, 5 – 12) та алгоритму 1a (крива 6 – 1 процес)

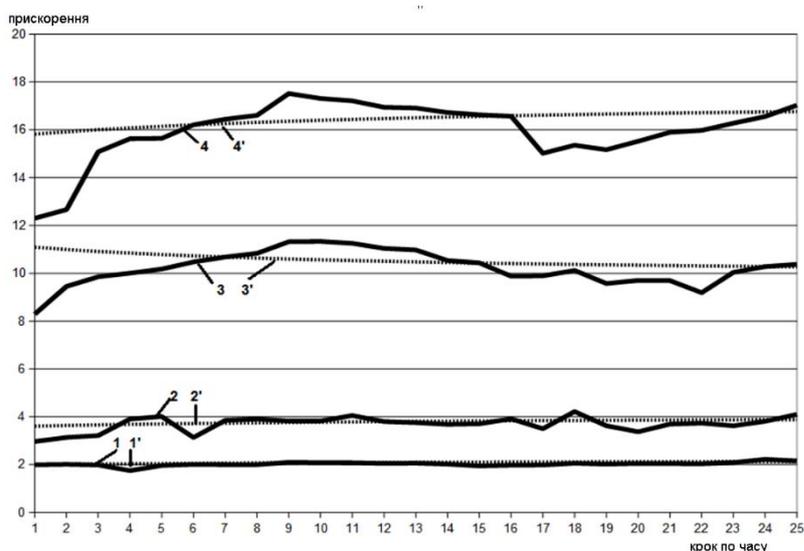


Рисунок 5.2 – Прискорення паралельного алгоритму (криві 1–4) для кластерних систем та його оцінка (криві 1'–4') (кількість процесів: 1,1' – 2; 2,2' – 4; 3,3' – 8; 4,4' – 12)

Використання багатопоточного розпаралелювання всередині процесів розподіленої програми дозволяє зменшити кількість та обсяг обмінів даними між процесами при збереженні рівня розпаралелювання обчислень. Вплив збільшення складності обчислень на прискорення паралельних алгоритмів проілюстровано на рис. 5.3 для випадку залучення 32 процесорних ядер на 8 вузлах кластера.

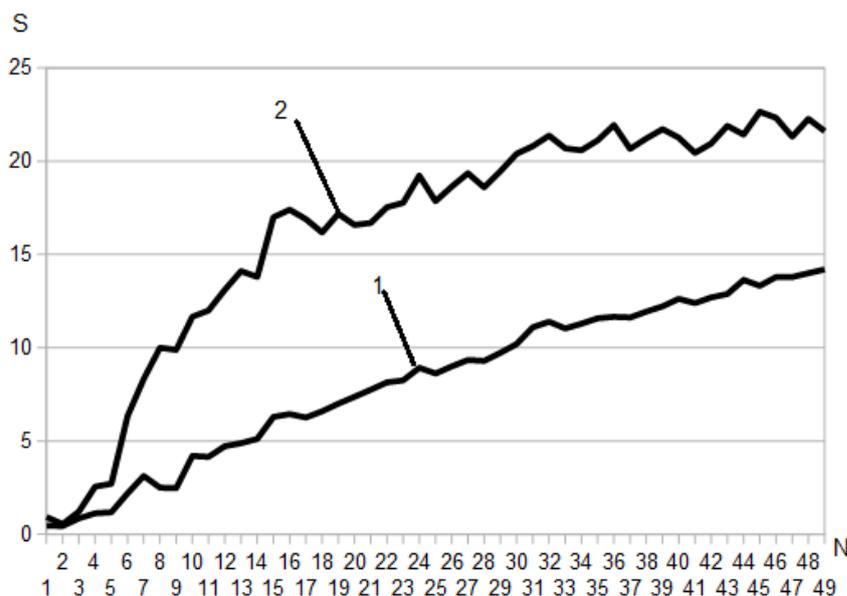


Рисунок 5.3 – Прискорення S паралельних алгоритмів в залежності від номера N кроку за часом (1 – один процес на одне ядро, 2 – один процес на один вузол кластера)

Прискорення тут очікувано збільшується зі зростанням складності на кроках за часом, а виконання обчислень на розподілених системах (таких, як кластери) за допомогою червоно-чорної блокової схеми розподілу даних у комбінації з багатопоточним розпаралелюванням є більш ефективним через зменшення обсягів обмінів даними між процесами .

5.2. Паралельні алгоритми для графічних процесорів

5.2.1. Паралельний алгоритм схем розщеплення для задач з дробовою похідною за часовою змінною

При розв'язанні двовимірних задач щодо моделей з похідною Капуто–Герасимова за часовою змінною за допомогою локально-одновимірної схеми розщеплення, алгоритми обчислення прогоночних коефіцієнтів та зворотного ходу прогонок для додаткового прискорення розрахунків також можуть бути розпаралелені для виконання графічними процесорами (GPU). Пропонується наступна схема розпаралелювання (**алгоритм 2**) [64, 269]:

- Кожен потік паралельної програми проводить незалежні обчислення в

рядку/стовпці комірок сітки;

- Для додаткового розпаралелювання (**алгоритм 2а**) при обчисленні сум в апроксимації дробових похідних (наприклад, (2.16)) один рядок/стовпець комірок може оброблятися блоком потоків, кожен з яких обчислює часткову суму, після чого потік з індексом 0 підсумовує отримані результати та проводить основні обчислення;
- Після виконання обчислень на кроці алгоритму 1 необхідно передати з пам'яті GPU в пам'ять центрального процесора (CPU) дані, які повинні бути передані іншому процесу розподіленої програми. Аналогічно, після отримання даних від іншого процесу, вони повинні бути скопійовані в пам'ять GPU;
- Після завершення обчислень, результат повинен бути скопійований з пам'яті GPU у пам'ять CPU.

Швидкодія алгоритму 2 так само, як і алгоритму 1, лінійно залежить від номера кроку за часом та від розміру сітки. Коефіцієнти цих лінійних залежностей при цьому залежать від кількості обчислювальних ресурсів графічного процесора.

Твердження 5.2. У випадку, коли GPU здатний виконати всі запуснені потоки одночасно та $m = n$, час роботи алгоритму 2 [64] можна оцінити як

$$T_N^G(n, t) = \frac{n}{N} \left(t_{gc0} + \frac{k_{gc}}{N_t} t \right) + 4(N-1) \left(t_{tr} \left(\frac{n}{N} \right) + t_s \left(\frac{n}{N} \right) \right), \quad (5.4)$$

де $t_{gc} = t_{gc0} + \frac{k_{gc}}{N_t} t$ – сума часу, необхідного для обчислення однієї пари прогоночних коефіцієнтів на GPU, N_t – кількість потоків в групі, яка проводить паралельне обчислення сум, $t_s(l)$ – час копіювання блоку, що складається з l одиниць оброблюваних даних, з пам'яті GPU у пам'ять CPU та навпаки (ці часи вважаються однаковими).

В оцінці (5.4) не враховується час, що витрачається центральним процесором на допоміжні операції, зокрема, на управління GPU.

Аналогічно тестуванню CPU-алгоритмів, були проведені дві серії обчислювальних експериментів. Вихідні коди розробленого програмного

забезпечення наведено у [265]. Дані щодо розв'язання задачі (2.30)–(2.32) на одному GPU алгоритмом 2а при різних розмірах сітки наведені на рис. 5.4.

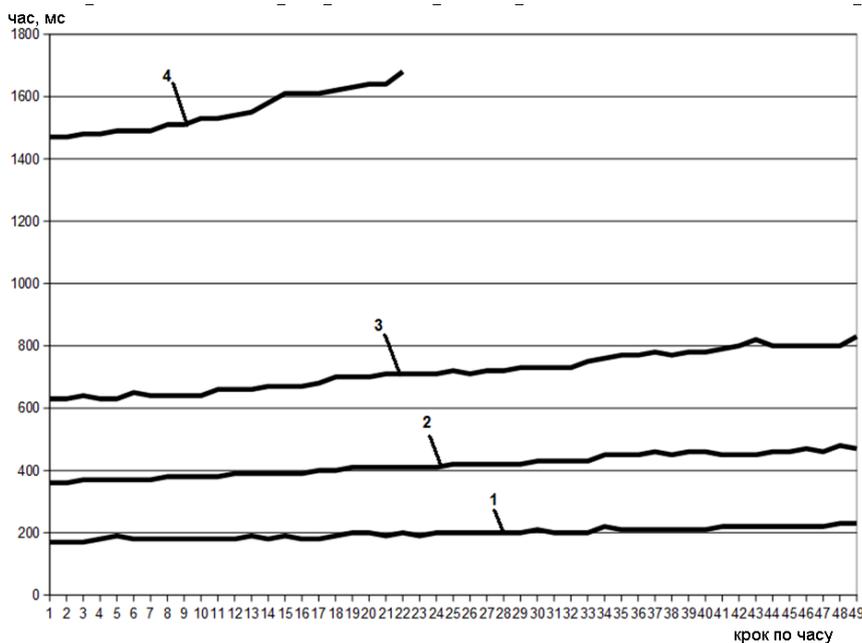


Рисунок 5.4 – Час розв'язання задачі на одному GPU

(розміри сітки: 1 – 500×500 , 2 – 750×750 , 3 – 1000×1000 , 4 – 1500×1500)

У цьому випадку також маємо лінійну залежність часу проведення обчислень від номера кроку, проте її коефіцієнти значно нижче, ніж для CPU-алгоритмів. Залежність від розмірів сітки залишається квадратичною через невиконання на реальних обчислювальних системах припущень, виходячи з яких за формулою (5.4) ця залежність повинна бути лінійною.

Друга серія експериментів полягала у розв'язанні алгоритмом 2а задачі з розміром сітки 3000×3000 при залученні різної кількості графічних процесорів (рис. 5.5). Результати експериментів показали наявність лінійних залежностей, аналогічних випадку розв'язання задачі одним GPU.

Було проведено також порівняння швидкодії базового алгоритму 2 та модифікованого алгоритму 2а, який на сітках розміру від 500×500 до 1500×1500 при отриманні розв'язку на 25 кроках за часом показав на $\sim 5\%$ більшу швидкодію.

Точність оцінки часу роботи паралельного алгоритму для кластерів з графічними процесорами, аналогічно результатам, отриманим для CPU-алгоритму, падає при збільшенні кількості використовуваних GPU [64].

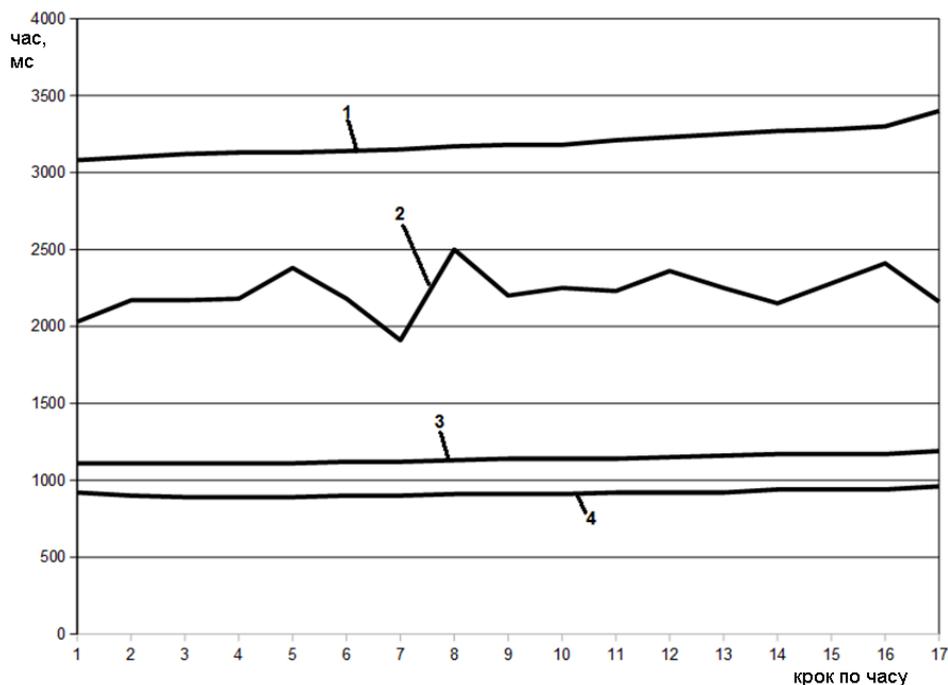


Рисунок 5.5 – Час розв'язання задачі для сітки розміру 3000×3000 (кількість залучених GPU: крива 1 – 2, крива 2 – 4, крива 3 – 8, крива 4 – 12)

Відзначимо істотно більше (до 10-кратного) прискорення GPU-алгоритмів при їх меншій масштабованості: при залученні 12 GPU в порівнянні з 12 CPU-ядрами було отримано тільки дворазове прискорення для задачі з сіткою розміру 3000×3000 .

5.2.2. Комбінований алгоритм для схем розщеплення

Додаткове прискорення можна отримати шляхом використання для проведення обчислень як графічного, так і центрального процесорів (**алгоритм 3**) [64, 269]:

- При проведенні вертикальних прогонок, обчислення в блоці стовпців $(1, \dots, n_x)$ виконуються CPU паралельно з виконанням на GPU обчислень в блоці стовпців $(n_x + 1, \dots, n)$;
- При проведенні горизонтальних прогонок, обчислення в блоці рядків $(1, \dots, n_y)$ виконуються CPU паралельно з виконанням на GPU обчислень в блоці рядків $(n_y + 1, \dots, m)$;

- Після проведення вертикальних прогонок виконується синхронізація значень шуканої функції між пам'яттю CPU та GPU: блок комірок $(n_x + 1, \dots, n) \times (1, \dots, n_y)$ копіюється з пам'яті GPU у пам'ять CPU, а блок комірок $(1, \dots, n_x) \times (n_y + 1, \dots, m)$ – з пам'яті CPU у пам'ять GPU;

- Синхронізація у зворотному напрямку виконується після проведення горизонтальних прогонок.

Твердження 5.3. Час роботи алгоритму 3 у випадку розв'язання задачі одним CPU та одним GPU при $m = n$, $n_y = n_x$ і припущенні щодо лінійності залежності часу передачі даних між пам'яттями CPU та GPU від їх обсягу ($t_s(l) = k_s l$) можна оцінити як

$$T_3(n, n_x, t) = \max \left(n_{x^2} (t_{c0} + k_c t), (n - n_x) \left(t_{gc0} + \frac{k_{gc}}{N_t} t \right) \right) + 4k_s n_x (n - n_x).$$

Пропонується два підходи до вибору розміру n_x [64]:

- Фіксований вибір (**алгоритм 3а**), який не вимагає модифікації алгоритму 3, але не враховує залежності $T_3(n, n_x, t)$ від номера кроку за часом. Тут при $t \leq t_1$, $t_1: n_{x^2} (t_{c0} + k_c t_1) = (n - n_x) \left(t_{gc0} + \frac{k_{gc}}{N_t} t_1 \right)$, швидкодія обчислювальної частини алгоритму буде більшою в порівнянні з виконанням на одному GPU, проте це прискорення може бути нівельованим додатковим часом, що витрачається на синхронізацію даних. При $t > t_1$ фіксований вибір стає неефективним оскільки основний вплив на швидкодію алгоритму матиме CPU;

- Динамічна зміна за евристичним критерієм (**алгоритм 3б**). Тут значення t_1 оцінюється, вимірюючи час виконання обчислень на кожному кроці. Вважаючи, що $t_1 > t_2$, і вибравши початкове значення n_x , будується лінійна залежність $T_3(n, n_x, t)$ від t виходячи з заміряних $T_3(n, n_x, 0)$ та $T_3(n, n_x, t_2)$. На подальших кроках $t > t_2$ відстежується точність апроксимації отриманою залежністю заміряного часу $T_3(n, n_x, t)$. Крок $t \approx t_1$ тоді, коли точність стає

незадовільною. При цьому, вважаючи, що швидкодія GPU більше ніж швидкодія CPU, проводиться зменшення $n_{x1} = k_n n_x$, де $k_n < 1$ – задана стала.

При зміні розміру n_x у процесі обчислень, необхідно виконати додаткові операції синхронізації: передати з пам'яті CPU у пам'ять GPU всі раніше обчислені значення шуканої функції у блоках $(n_{x1} + 1, \dots, n_x) \times (1, \dots, n_x)$ та $(1, \dots, n_{x1}) \times (n_{x1} + 1, \dots, n_x)$ комірок сітки.

Тестування комбінованих алгоритмів при використанні одного GPU проводилося на сітці розміру 1500x1500. Вихідні коди розробленого програмного забезпечення наведено у [265]. Фіксоване (або початкове) значення n_x було прийнято рівним kn , де k – задане число, а множник k_n для алгоритму 3б дорівнював 0,5. Отримані результати підтверджують вищеописані теоретичні очікування, зокрема, істотне погіршення швидкодії алгоритму 3а в процесі розв'язання при вищій початковій швидкодії (тестування при $k = 0,1$) та ефективність динамічної зміни параметра k в алгоритмі 3б.

Для задачі з розміром сітки 1500x1500 на 25 кроках за часом, найбільше прискорення від використання алгоритму 3а (при $k = 0,05$) в порівнянні з GPU-алгоритмом 2 склало 4,5%. При подальшому проведенні обчислень прискорення знижується, і алгоритм стає неефективним. Для алгоритму 3б найбільше прискорення було отримано при $k = 0,1$ і склало 5%. При подальшому проведенні обчислень прискорення тут також знижується і при $n_x = 0$ алгоритм 3б стає ідентичним алгоритму 2.

На задачах з розміром сітки, меншим, ніж 1500x1500, комбіновані алгоритми через істотний час, що витрачається на синхронізацію даних, показували меншу швидкодію в порівнянні з GPU-алгоритмом.

При розв'язанні задачі з сіткою розміром 3000x3000 декількома GPU та CPU, ефективність від використання комбінованих алгоритмів була отримана тільки при розпаралелюванні розв'язання на два GPU (1,5% прискорення при використанні алгоритму 3б на 25 кроках за часом). При подальшому масштабуванні, гранулярність обчислень падає і комбіновані алгоритми стають неефективними.

5.2.3. Паралельний алгоритм для одновимірних задач з дробовою похідною за просторовою змінною

Розглянемо процес чисельного моделювання фільтраційно-консолідаційних процесів в насиченому сольовими розчинами геопористому середовищі відповідно до системи (4.5), (4.12), (4.6), (4.13) в разі присутності нелокальності виключно за просторовою змінною.

Розв'язання методом прогонки тридіагональних СЛАР (4.17), (4.18), маючи порівняно невелику обчислювальну складність, передбачає, проте, ітераційне виконання на кожному кроці за часом та необхідність отримання розв'язку на досить великих часових проміжках. Це істотно знижує ефективність розпаралелювання на системах з розподіленою пам'яттю, дозволяючи при цьому ефективно використовувати ресурси систем зі спільною пам'яттю, зокрема GPU.

Виділимо три типових операції, з яких складається алгоритм проведення розрахунків [34] (**алгоритм 4**):

- 1) Обчислення елементів матриць коефіцієнтів СЛАР;
- 2) Знаходження значень елементів векторів правої частини СЛАР на даному часовому шарі, що вимагає виконання операцій підсумовування, які виникають як результат дискретизації дробової похідної за просторовою змінною;
- 3) Розв'язання СЛАР.

У межах першого блоку операцій, обчислення значень елементів незалежні при відсутності загального доступу до пам'яті. Ефективною реалізацією цих операцій на GPU є схема, в якій кожен потік обробляє один елемент вектора без об'єднання потоків у групи.

Особливістю другого блоку операцій є те, що складність виконання операцій підсумовування залежить від кількості доданків у відповідних сумах. При розпаралелюванні тут виникає проблема рівномірного завантаження потоків для максимального залучення ресурсів графічного процесора. Для розпаралелювання може бути використана наступна схема розподілу елементів векторів правих частин СЛАР по GPU-потоках [34]:

- Кожен з N/M потоків проводить обчислення M елементів вектора розміру N ;
- Вектор розбивається на блоки розміру N/M і кожен потік обчислює одне значення елемента вектора з кожного такого блоку;
- Номер елемента в блоці b , що обчислюється потоком k визначається як

$$e_{kb} = \begin{cases} k, & b \wedge 1 = 0, \\ \frac{N}{M} - k, & b \wedge 1 = 1, \end{cases} \text{ де } \wedge \text{ – по-компонентна булева операція } I.$$

При обчисленні сум з різною кількістю доданків, потоки GPU-програми звертаються до одних і тих самих елементів пам'яті, в яких зберігаються значення раніше розрахованих полів напору або концентрації. Це дозволяє використовувати для прискорення обчислень групування потоків та локальну пам'ять:

- Суми розбиваються на блоки з N/M доданків;
- Для кожного блоку, кожен з процесів завантажує з глобальної пам'яті в локальну одне значення необхідної функції;
- Подальші обчислення проводяться з використанням тільки більш швидкої локальної пам'яті.

Для розв'язання тридіагональних СЛАР може бути використана відома реалізація алгоритму паралельної циклічної редукції [250].

Відзначимо, що при проведенні обчислень, обмін даними між пам'яттю центрального та графічного процесорів проводиться тільки при ініціалізації обчислень і при отриманні їх результатів.

Алгоритми для GPU були програмно реалізовані з використанням технології OpenCL. Вихідні коди розробленого програмного забезпечення наведено у [265]. Обчислювальні експерименти щодо розв'язання задач різної розмірності використовуючи CPU та GPU проводилися на GPU серії Intel (R) HD Graphics та процесорі Intel (R) Celeron (R) CPU N2840 з тактовою частотою 2,16GHz. Дані щодо швидкодії та прискорення алгоритму 4 представлені на рис. 5.6, 5.7 [34]. Результати експериментів демонструють лінійне збільшення прискорення GPU-алгоритму при збільшенні розміру сітки.

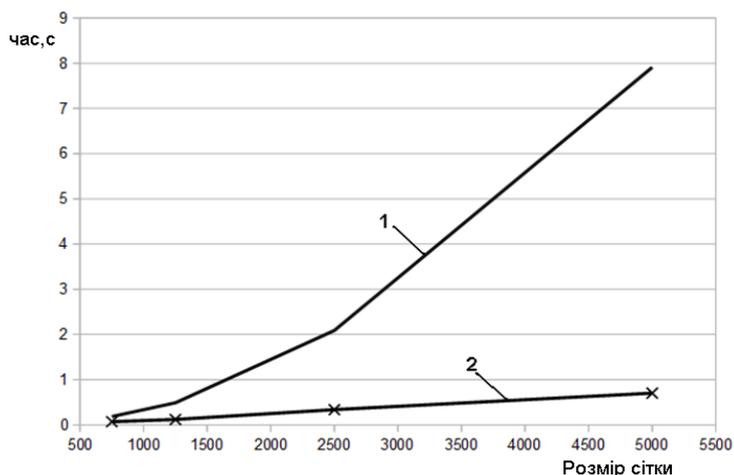


Рисунок 5.6 – Час розв’язання задачі (1 – на CPU, 2 – на GPU)

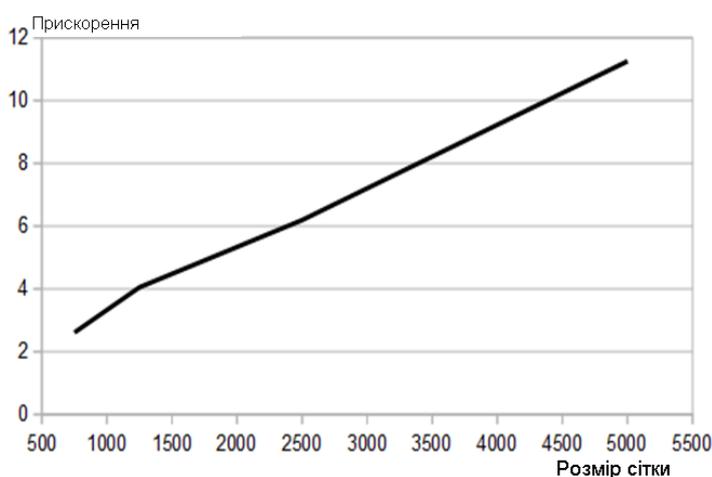


Рисунок 5.7 – Прискорення алгоритму для графічного процесора

5.2.4. Паралельні алгоритми для одновимірних задач з дробовою похідною за часовою змінною

Будемо розглядати модель дробово-диференціальної консолідаційної динаміки насиченого сольовим розчином масиву, що ґрунтується на системі рівнянь з несингулярними ядрами вигляду [67, 276] (3.13), (3.14), описаних у п. 3.1.

Обчислювальна складність визначення значень коефіцієнтів СЛАР (3.21), (3.22), до розв’язання яких зводиться розв’язання відповідної початково-крайової задачі, лінійно залежить від номера кроку за часом, що обумовлює необхідність використання паралельних обчислень при розв’язанні таких задач на великих часових проміжках [67, 276].

Використання GPU є ефективним в цьому випадку, оскільки більша частина часу при розв'язанні задачі витрачається на незалежні операції визначення значень вектора правої частини СЛАР.

Представимо алгоритм проведення розрахунків як такий, що складається з наступних операцій (**алгоритм 5**):

- Обчислення значень елементів матриці СЛАР;
- Знаходження значень елементів вектора правої частини СЛАР, що вимагає обчислення значень сум, присутніх в дискретизації (3.20) дробової похідної за часовою змінною;
- Розв'язання СЛАР.

Операції в межах першого та другого блоків незалежні одна від одної за відсутності загального доступу до пам'яті. Ефективною реалізацією цих операцій на GPU є схема, в якій один потік обробляє один елемент вектора без об'єднання потоків у групи.

Для розв'язання тридіагональних СЛАР може бути використана реалізація GPU-алгоритму паралельної циклічної редукції описана в [250]. При розв'язанні СЛАР з використанням центрального процесора (CPU) використовувався алгоритм методу прогонки.

При проведенні обчислень, обмін даними між пам'яттю центрального та графічного процесорів проводиться лише при ініціалізації обчислень та при отриманні їх результатів.

Обчислення значення дробових похідних на поточному кроці за часом вимагає зберігання значень функції, що диференціюється, отриманих на попередніх етапах. Це призводить до необхідності розширення блоку пам'яті, виділеного для зберігання таких даних на GPU. У реалізованій схемі, розмір блоку пам'яті збільшується вдвічі при його повному заповненні.

Твердження 5.4. Час проведення обчислень паралельним алгоритмом 5 з розв'язанням СЛАР методом циклічної редукції на j -тому кроці за часом без врахування обмінів даними між пам'яттю центрального та графічного процесорів можна оцінити наступним чином [67]:

$$T_1(N, j) = k_1 \left(10 + 2 \log_2 \frac{N}{2} \right) + k_2 \lfloor N / P \rfloor \left(k_3 + k_4 j + k_5 \log_2 \frac{N}{2} \right), \quad (5.5)$$

де k_1 – час, що витрачається на допоміжні операції по запуску GPU-ядер, k_2 – коефіцієнт швидкодії GPU, k_3 – коефіцієнт складності операцій обчислення значень елементів матриці СЛАР, k_4 – коефіцієнт складності обчислення значень вектора правої частини СЛАР, k_5 – коефіцієнт складності проведення обчислень на одній ітерації алгоритму паралельної циклічної редукції, P – кількість скалярних процесорів на GPU, N – кількість вузлів сітки.

Твердження 5.5. У випадку послідовного виконання алгоритму 5 з розв'язанням СЛАР методом прогонки, час роботи може бути оцінений як

$$T_2(N, j) = k_6 N (k_3 + k_4 j + k_7), \quad (5.6)$$

де k_6 – коефіцієнт швидкодії центрального процесора, k_7 – коефіцієнт складності обчислення розв'язку СЛАР методом прогонки.

Алгоритми для GPU були програмно реалізовані з використанням технології OpenCL. Вихідні коди розробленого програмного забезпечення наведено у [265]. Обчислювальні експерименти щодо розв'язання задач різної розмірності використовуючи CPU та GPU проводилися на одному вузлі кластера СКІТ-4 Інституту кібернетики НАН України [67, 276].

Час проведення обчислень в залежності від номера кроку за часом при різних розмірах сітки, а також його оцінки згідно з (5.5) та (5.6), наведені на рис. 5.8 для CPU-алгоритму та на рис. 5.9 – для GPU-алгоритму 5.

Час роботи GPU-алгоритму оцінювався для $P=1000$. Отримані результати тестування показують адекватність оцінок (5.5) та (5.6).

Прискорення, що забезпечується GPU-алгоритмом у порівнянні з CPU-алгоритмом в залежності від номера кроку за часом, представлено на рис. 5.10. Як видно з отриманих даних, воно лінійно зростає, зумовлюючи високу ефективність використання GPU при моделюванні розглядуваних процесів на великих часових інтервалах.

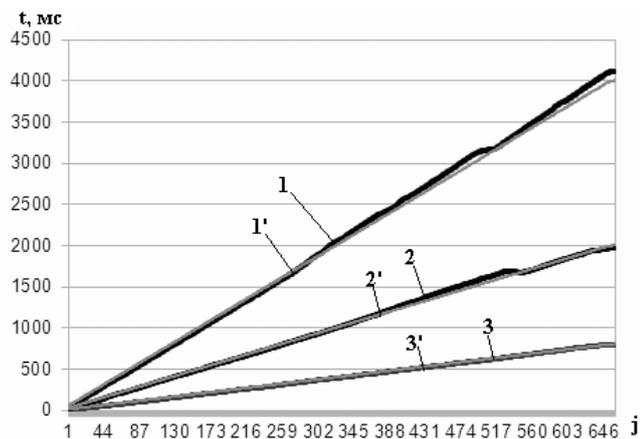


Рисунок 5.8 – Час t роботи CPU-алгоритму (1,2,3) на кроках за часом j та його оцінка ($1'$, $2'$, $3'$) ($1,1'$ – $N = 50000$; $2,2'$ – $N = 25000$; $3,3'$ – $N = 10000$)

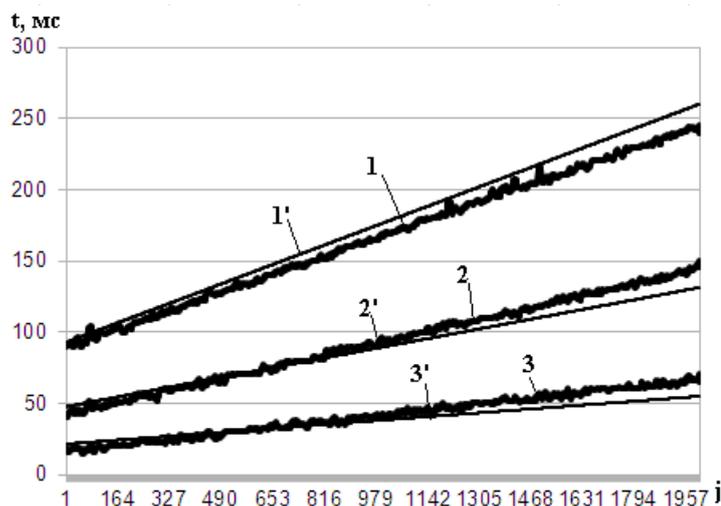


Рисунок 5.9 – Час t роботи GPU-алгоритму (1,2,3) на кроках за часом j та його оцінка ($1'$, $2'$, $3'$) ($1,1'$ – $N = 50000$; $2,2'$ – $N = 25000$; $3,3'$ – $N = 10000$)

Прискорення при проведенні обчислень на 600 кроках за часом залежно від розміру сітки показано на рис. 5.11. Час, витрачений GPU-алгоритмом 5 на допоміжні операції, істотно знижує його швидкодню на невеликих сітках. При збільшенні розміру сітки, прискорення стає постійним та залежить тільки від кількості та швидкодії скалярних процесорів GPU.

Додатково розглянемо декілька варіантів оптимізованої організації обчислень у випадку систем великого розміру при їх застосуванні до розв'язання задач, що розглядаються у п. 4.2 [53].

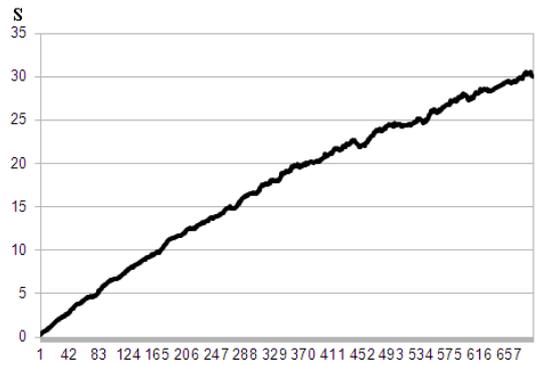


Рисунок 5.10 – Прискорення GPU-алгоритму S на кроках за часом j

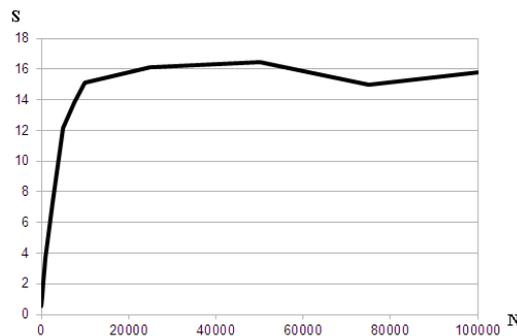


Рисунок 5.11 – Сумарне прискорення GPU-алгоритму S в залежності від розміру сітки N

Через те, що, починаючи з певного кроку за часом, домінуючою в часі роботи алгоритму стає операція обчислення значень вектора правої частини, будемо розглядати оптимізовані варіанти її організації.

Найпростішим способом обчислення значень елементів вектора правої частини на графічному процесорі є обробка одного елемента одним потоком (алгоритм 6) [53].

Твердження 5.6. Вважаючи, що GPU має K ядер, час, що витрачається на обчислення елементів векторів Φ^j правої частини СЛАР (4.35), у випадку сітки, що містить N вузлів, на кроці j за часом для алгоритму 6 може бути оцінений як

$$T_0(N, K, j) = k_0 j \left\lceil \frac{N}{K} \right\rceil,$$

де k_0 – час, що витрачається на обчислення значення одного елемента вектора Φ^j (4.35).

Оскільки значення інтегралів $b^j, \bar{b}_s^{(j)}$ залежать тільки від часової змінної, вони можуть бути скопійовані в локальну пам'ять GPU і використовуватись декількома різними потоками (**алгоритм 7**) [53]:

- N потоків об'єднуються в групи розміру S_b (не зменшуючи загальності, для спрощення викладу тут і надалі вважаємо, що $N \% S_b = 0$, де $\%$ – залишок від ділення);
- Потік t , обробляючи один елемент вектора правої частини, спочатку обчислює фіксовану складову $F_t^{(j)} = -b_j \frac{\sigma}{\tau} C_t^j$ та встановлює $S_{t,0}^{(j)} = 0$;

- Обчислення $\sum_{s=0}^{j-1} \bar{b}_s^{(j)} (C_t^{s+1} - C_t^s)$ проводиться потоками покроково, виконуючи S_b операцій додавання за один крок;

- Перед кроком i , потоки паралельно завантажують $(\bar{b}_{j-iS_b-1}^{(j)}, \dots, \bar{b}_{j-(i+1)S_b-1}^{(j)})$ у локальну пам'ять. Потім, кожен з потоків збільшує значення $S_t^{(j)}$:

$$S_{t,i}^{(j)} = S_{t,i-1}^{(j)} + \bar{S}_{t,i-1}^{(j)}, \quad \bar{S}_{t,i-1}^{(j)} = \sum_{s=j-(i+1)S_b-1}^{j-iS_b-1} \bar{b}_s^{(j)} (C_t^{(s+1)} - C_t^{(s)});$$

- Кожен з потоків обчислює Φ_t^j як $\Phi_t^{(j)} = F_t^{(j)} + \frac{\sigma}{2\tau} S_{t, \lceil j/S_b \rceil}^{(j)}$.

Твердження 5.7. Час, що витрачається на обчислення елементів вектора правої частини алгоритмом 7, може бути оцінений як $T_1(N, K, j) = k_1 j \left\lceil \frac{N}{K} \right\rceil$, $k_1 < k_0$.

Кількість потоків, які працюють паралельно в алгоритмах 6 та 7, дорівнює кількості вузлів сітки, що обмежує їх масштабованість.

Пропонується наступна схема (**алгоритм 8** [53]) для збільшення максимального числа потоків:

- Ядро (kernel) 1 виконується у $N \lceil j/S_b \rceil$ потоках, об'єднаних в групи по S_b потоків. Потік з глобальним індексом t обчислює значення $\bar{S}_{t \% N, t/N}^{(j)}$,

використовуючи локальну пам'ять для збереження значень $\bar{b}_s^{(j)}$ так само, як і в алгоритмі 7;

- Після виконання першого ядра, ядро 2, яке виконується у N потоках,

$$\text{проводить незалежні обчислення } \Phi_t^{(j)} = F_t^{(j)} + \frac{\sigma}{2\tau} \sum_{k=0}^{\lceil j/S_b \rceil} S_{t,k}^{(j)}.$$

Алгоритм 8 вимагає додаткову пам'ять для зберігання значень $\bar{S}_{t\%N, t/N}^{(j)}$.

Твердження 5.8. Час, що витрачається на обчислення елементів вектора правої частини алгоритмом 8, може бути оцінений як

$$T_2(N, K, j) = k_1 S_b \left\lceil \frac{N \lceil j / S_b \rceil}{K} \right\rceil + k_2 \left\lceil \frac{j}{S_b} \right\rceil \left\lceil \frac{N}{K} \right\rceil,$$

де k_2 – коефіцієнт швидкодії виконання ядра 2.

Для подальшого збільшення масштабованості алгоритмів в разі наявності декількох GPU на одному вузлі обчислювальної системи, пропонується розподіляти на них обчислення частин вектора $\Phi^{(j)}$, виконуючи інші операції на одному GPU. Такий підхід вимагає виконання операцій копіювання даних між пам'яттями CPU та графічних процесорів. Блоки елементів векторів $C^{(j-1)}$ повинні бути скопійовані з пам'яті основного GPU в пам'яті інших GPU перед проведенням обчислень вектора $\Phi^{(j)}$, після чого обчислені блоки елементів вектора $\Phi^{(j)}$ повинні бути передані у зворотному напрямку. Залежно від використовуваного апаратного та програмного забезпечення, ці операції можуть виконуватися з або без використання пам'яті CPU.

Твердження 5.9. Існує наступна оцінка часу виконання алгоритмів на декількох GPU [53]:

$$T_3(N, K, j, N_{GPU}) = T_i(N, KN_{GPU}, j) + 2k_r N,$$

де T_i – оцінка часу виконання для алгоритмів 6,7 або 8, N_{GPU} – кількість GPU, k_r – час, що витрачається на виконання операції копіювання одного елемента вектора з пам'яті одного GPU в пам'ять іншого (вважаємо, що копіювання N елементів лінійно залежить від k_r).

Відзначимо, що всі розглянуті алгоритми вимагають зберігання в пам'яті GPU обчислених розв'язків в усі попередні щодо поточного моменту часу. Відповідно, розмір використовуваної ними пам'яті зростає зі збільшенням номера кроку за часом. Для реалізації зберігання даних, розмір блоку глобальної пам'яті GPU, що виділяється під зберігання розв'язків, збільшується вдвічі при повному заповненні. Необхідність копіювання даних при цьому знижує прискорення при виконанні обчислень на кроці, на якому здійснюється повторне виділення пам'яті.

Вихідні коди розробленого програмного забезпечення наведено у [265]. Прискорення розглянутих алгоритмів у порівнянні з виконанням на CPU в залежності від номера блоку з 1000 кроків за часом приведено на рис. 5.12 для сітки, що містить 1000 вузлів. Прискорення тут було розраховано як час, витрачений CPU для розв'язання задачі на заданому блоці кроків за часом, поділений на такий самий час, витрачений одним з GPU-алгоритмів. Для алгоритмів 7 та 8 кількість потоків, що об'єднувались в одну групу, дорівнювало 16.

Отримані результати показують, що для всіх алгоритмів прискорення стає постійним починаючи з певного номера кроку за часом. Це відповідає моменту, коли кількість потоків стає більше, ніж кількість обчислювальних ресурсів GPU. Використання декількох графічних процесорів було ефективним для алгоритму 8, але неефективно для алгоритму 7. У разі розподілу обчислень на декілька графічних процесорів, використання менш масштабованого алгоритму 7 призводить до ситуації, коли деякі обчислювальні ресурси GPU не використовуються. Підвищення продуктивності тут обмежується необхідністю виконувати передачу даних між GPU, використовуючи пам'ять CPU, через особливості реалізації стандарту OpenCL.

У другій серії обчислювальних експериментів вивчався вплив розміру груп потоків на продуктивність алгоритмів. Прискорення алгоритму 8, виконаного на двох графічних процесорах у залежності від кількості блоків у 1000 кроків за часом для різних розмірів груп потоків наведено на рис. 5.13. Згідно з отриманими даними, оптимальний розмір групи дорівнював 128.

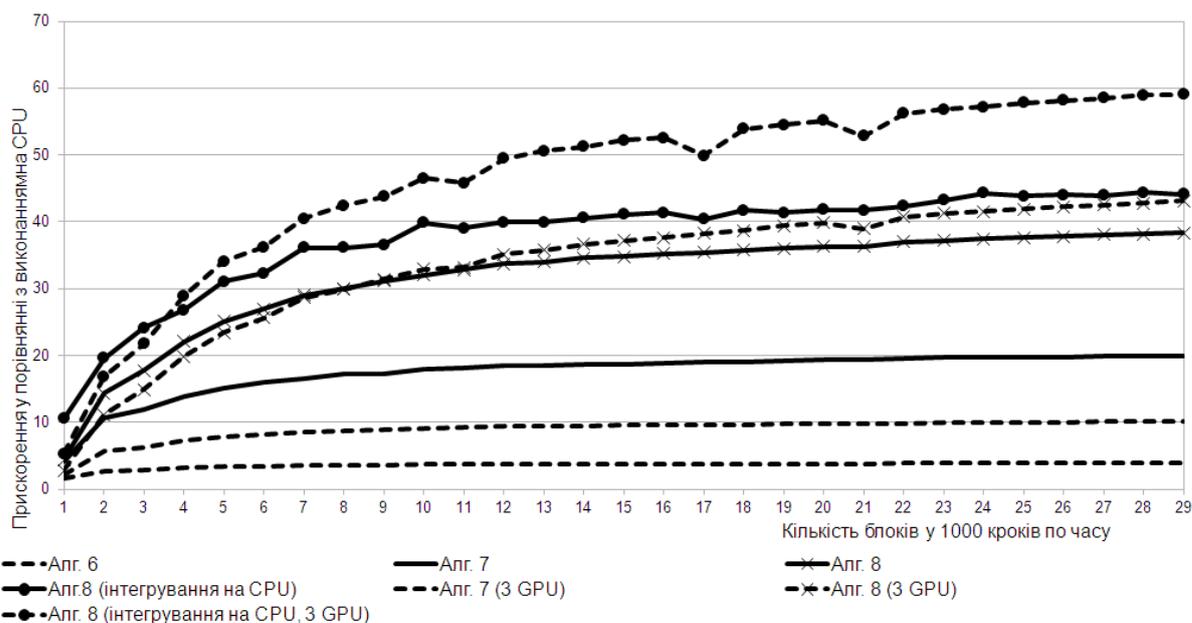


Рисунок 5.12 – Прискорення алгоритмів в залежності від номера блоку кроків за часом

Прискорення найбільш швидкого алгоритму при виконанні 2500 ітерацій у залежності від розміру сітки наведено на рис. 5.14. Результати показують, що час, який витрачається на отримання розв'язку задачі, лінійно збільшується зі збільшенням розміру сітки для всіх алгоритмів, що є теоретично очікуваним. Для сіток з кількістю вузлів більше ніж 4000, алгоритм 7 стає швидше, ніж алгоритм 8. У цьому випадку, алгоритм 7 повністю використовує обчислювальні ресурси графічного процесора, тоді як алгоритм 8 повинен виконувати додаткові операції підсумовування. Співвідношення часу, витраченого алгоритмом 8, який виконується на трьох графічних процесорах, до часу, витраченого алгоритмом 7, становить 1,2 для сіток з більш ніж 20000 вузлів. Той же коефіцієнт для алгоритму 8, який виконується на одному GPU, прямує до 0,9. Прискорення алгоритму 7 у порівнянні з виконанням на CPU прямує до 76 для сіток з більш ніж 40000 вузлів. Для досить великих сіток, час виконання алгоритму 7 залежить від розміру сітки як $T_1^{(e)}(N) = 4,09N + 3318$, тоді як відповідна залежність для випадку виконання на CPU має вигляд $T_{CPU}^{(e)}(N) = 315,967N - 716805$.

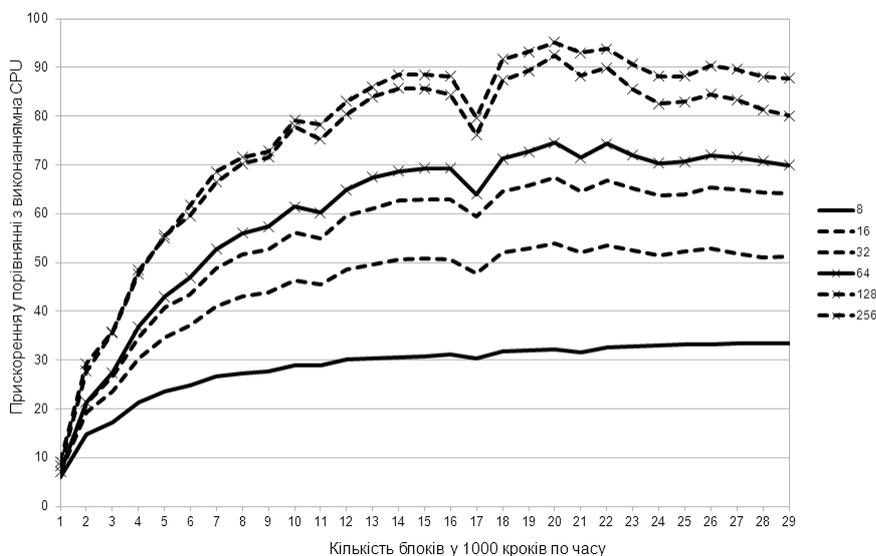


Рисунок 5.13 – Прискорення алгоритму в залежності від розміру груп потоків

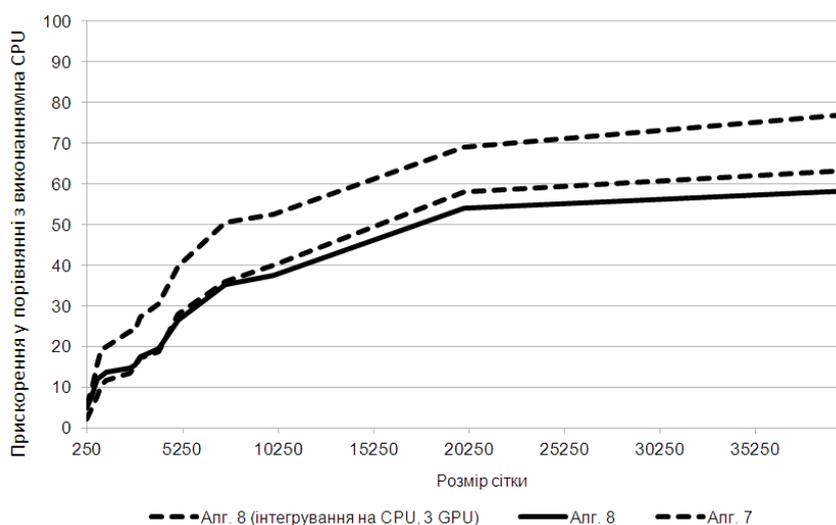


Рисунок 5.14 – Прискорення алгоритмів в залежності від розміру сітки

5.3. Алгоритм оптимізації обчислень при моделюванні дробово-диференціальної за часовою змінною міграційної динаміки

5.3.1. Методика наближеного обчислення дробової похідної за часовою змінною

При розв'язанні диференціальних рівнянь з дробовою похідною Капуто–Герасимова за часовою змінною, порядок складності при послідовному обчисленні значень $\Delta_t^{(\alpha)}$ і згідно з (2.35) може бути знижений за допомогою апроксимації сум степеневими рядами та їх модифікації в процесі обчислень [44, 269].

Нехай

$$\Delta_t^{(\alpha)} u = \frac{1}{\tau^\alpha \Gamma(2-\alpha)} \sum_{r=0}^i \left[g(r, i) \left(u^{(r+\gamma)} - u^{(r)} \right) \right], \quad (5.7)$$

$$g(r, i) = (i-r+a)^{1-\alpha} - (i-r+c)^{1-\alpha}, \quad 0 < \alpha \leq 1, a > c, \quad (5.8)$$

звідки отримуємо (2.35) при $\gamma = 1, a = 1, c = 0$.

При апроксимації функції $g(r, i)$ у вигляді відрізка ряду

$g(r, i) \approx \sum_{k=0}^{k_m} f_k^{(1)}(r) f_k^{(2)}(i)$, де k_m – обмеження на кількість його членів, отримуємо

$$\Delta_t^{(\alpha)} u \approx \frac{1}{\tau^\alpha \Gamma(2-\alpha)} \sum_{k=0}^{k_m} \left(f_k^{(2)}(i) \sum_{r=0}^i \left[f_k^{(1)}(r) \left(u^{(r+\gamma)} - u^{(r)} \right) \right] \right). \quad (5.9)$$

Розглянемо апроксимацію функції (5.8), побудовану на основі узагальненої формули бінома Ньютона [44]:

$$g(r, i) = \sum_{k=0}^{\infty} g_k^{(1)}(i, r) \approx \sum_{k=0}^{k_m} g_k^{(1)}(i, r), \quad g_k^{(1)}(i, r) = f_k^{(1,1)}(r) f_k^{(2,1)}(i), \quad (5.10)$$

$$f_k^{(2,1)}(i) = i^{1-\alpha-k}, \quad f_k^{(1,1)}(r) = \frac{(1-\alpha)(-\alpha)\dots(2-\alpha-k)}{k!} \left((a-r)^k - (c-r)^k \right).$$

При фіксованому r , $g_k^{(1)}(i, r) \leq \varepsilon$ при

$$i \geq i^{(1)}(r, k), \quad i^{(1)}(r, k) = \left(\frac{(1-\alpha)(-\alpha)\dots(2-\alpha-k) \left((a-r)^k - (c-r)^k \right)}{\varepsilon \cdot k!} \right)^{1/(\alpha+k-1)}.$$

Оцінимо похибку ε_1 апроксимації функції (5.8) відрізком ряду (5.10) при фіксованих r , k і $i \geq i^{(1)}(r, k)$. Враховуючи що $i-r+a \geq 0$ та $i-r+c \geq 0$, представимо (5.10) як різницю рядів Тейлора функції $h(l, i) = (i+l)^{1-\alpha}$ по l у точці $l=0$ при $l=a-r$ та $l=c-r$. Тоді, враховуючи, що $\frac{\partial h^{(k+1)}(i, l)}{\partial l^{(k+1)}}$ зменшується при збільшенні $i+l$, залишковий член ряду (5.10) у формі Лагранжа можна оцінити як

$$|\varepsilon_1(i, r, k)| \leq \frac{|(1-\alpha)(-\alpha)\dots(1-\alpha-k)|}{(k+1)!} \times \\ \times \left[|a-r|^{k+1} (a-r+i^{(1)}(r, k))^{-\alpha-k} + |c-r|^{k+1} (c-r+i^{(1)}(r, k))^{-\alpha-k} \right].$$

З іншого боку, оскільки $g_k^{(1)}(i, r) \leq \varepsilon$ при $i \geq i^{(1)}(r, k)$, $|\varepsilon_1(i, r, k)| > \varepsilon$.

Інша апроксимація функції (5.8) отримується як відрізок ряду Тейлора по i в точці $i = b$ [44]:

$$g(r, i) = \sum_{k=0}^{\infty} g_k^{(2)}(i, r, b) \approx \sum_{k=0}^{k_m} g_k^{(2)}(i, r, b), \\ g_k^{(2)}(i, r, b) = f_k^{(1,2)}(r, b) f_k^{(2,2)}(i, b), \quad f_k^{(2,2)}(i, b) = (i-b)^k, \quad (5.11) \\ f_k^{(1,2)}(r, b) = \frac{(1-\alpha)(-\alpha)\dots(2-\alpha-k)}{k!} \left((a+b-r)^{1-\alpha-k} - (c+b-r)^{1-\alpha-k} \right).$$

Тут, при фіксованих r та k , $|g_k^{(2)}(i, r, b)| \leq \varepsilon$ при

$$|i-b| \leq i^{(2)}(r, b, k),$$

$$i^{(2)}(r, b, k) = \left(\frac{\varepsilon k!}{|(1-\alpha)(-\alpha)\dots(2-\alpha-k) \left((a+b-r)^{1-\alpha-k} - (c+b-r)^{1-\alpha-k} \right)|} \right)^{1/k}.$$

Розписуючи залишковий член ряду (5.11) у формі Лагранжа при фіксованих r, k та $|i-b| \leq i^{(2)}(r, b, k)$ і враховуючи, що $g^{(k+1)}(r, i)$ зростає при збільшенні i , похибка ε_2 апроксимації функції (5.8) відрізком ряду (5.11) може бути оцінена як $|\varepsilon_2(i, r, b, k)| \leq |f_{k+1}^{(2,2)}(r, i + i^{(2)}(r, b, k)) f_{k+1}^{(2,2)}(i, b)|$.

З іншого боку, оскільки $|g_k^{(2)}(i, r, b)| \leq \varepsilon$ при $|i-b| \leq i^{(2)}(r, b, k)$, $|\varepsilon_2(i, r, b, k)| > \varepsilon$.

В цілому, при фіксованих r, k_m та ε , для наближеного обчислення $g(r, i)$ використаємо серію апроксимацій (5.11), які з достатньою точністю апроксимують $g(r, i)$ при $r+1 \leq i < i^{(1)}(r, k)$, та апроксимацію (5.10) на інтервалі $i \geq i^{(1)}(r, k)$. Апроксимація будується відповідно до наступного алгоритму [44]:

1) Нехай $r_0 = r$;

- 2) Знаходимо значення $b_j > r_j + 1 : b_j - i^{(2)}(r, b_j, k) = r_j$ і будуємо апроксимацію (5.11);
- 3) Якщо $b_j + i^{(2)}(r, b_j, k) < i^{(1)}(r, k)$, то будуємо наступну апроксимацію (5.11): встановлюємо $r_{j+1} = b_j + i^{(2)}(r, b_j, k)$ і переходимо до кроку 2;
- 4) Будуємо апроксимацію (5.10).

Похибка $\varepsilon_3(i)$ апроксимації функції (5.8) тут буде мати оцінки

$$\varepsilon < |\varepsilon_3(i)| \leq \max \left(\max_{j=0, \dots, n_1} |\varepsilon_2(i, r_j, b_j, k)|, |\varepsilon_1(i, r, k)| \right),$$

де n_1 – кількість апроксимацій (5.11), побудованих на кроках 2 та 3 алгоритму.

Похибка ε_4 апроксимованого обчислення значень (5.7) згідно з (5.9) буде мати наступну верхню оцінку:

$$\varepsilon_4 \leq \frac{|\varepsilon_3(i)|}{\tau^\alpha \Gamma(2 - \alpha)} \sum_{r=0}^i |u^{(r+\gamma)} - u^{(r)}|.$$

При послідовній апроксимації та обчисленні значень $\Delta_i^{(\alpha)}u$ на кроці l змінюємо та доповнюємо апроксимацію $\Delta_i^{(\alpha)}u$ для $i = l - 1$ таким чином, щоб утворити апроксимацію для $i = l$. Поточна апроксимація для $i = l - 1$ при цьому складається з множини $A^{(1)}$ апроксимацій (5.10) та множини $A^{(2)}$ апроксимацій (5.11) для кожної з яких встановлюються межі по i їх застосовності (d_l, d_h) . За замовчуванням, для апроксимацій (5.11) $d_h = b, d_l = b - i^{(2)}(r, b, k_m)$, а для апроксимацій (5.10) – $d_h = \infty, d_l = i^{(1)}(r, k_m)$. Значення $\Delta_i^{(\alpha)}u$ на кроці l обчислюються як сума тих апроксимацій, для яких l лежить в межах їх застосовності.

Процедура перебудови (**процедура 1**) наступна [44]:

- 1) Видаляємо з множини $A^{(2)}$ апроксимації, для яких $d_h(A_m^{(2)}) < l, A_m^{(2)} \in A^{(2)}$;
- 2) Для всіх $A_m^{(2)} \in A^{(2)} : d_l(A_m^{(2)}) > l$, будуємо апроксимацію (5.11) $A_l^{(2)}$ для

поточного моменту часу l та $b = b(A_m^{(2)}), r = l$. Якщо виконується умова

$$b(A_m^{(2)}) - i^{(2)}(l, b(A_m^{(2)}), k_m) \leq d_l(A_m^{(2)}) \quad (5.12)$$

додаємо значення її коефіцієнтів до значень коефіцієнтів $A_m^{(2)}$. Якщо умова (5.12) не виконується, але існує $k_m < k_l \leq k_{m_2} : b(A_m^{(2)}) - i^{(2)}(l, b(A_m^{(2)}), k_l) \leq d_l(A_m^{(2)})$, де k_{m_2} – максимально допустима кількість членів ряду (5.9), апроксимація (5.11) $A_l^{(2)}$ будується для $k = k_l$. Якщо збільшення кількості членів ряду не призводить до виконання умови (5.12), апроксимація $A_m^{(2)}$ розбивається на дві: $A_{m_1}^{(2)}, d_l(A_{m_1}^{(2)}) = d_l(A_m^{(2)}), d_h = d_l(A_l^{(2)})$ з коефіцієнтами $A_m^{(2)}$ та $A_{m_2}^{(2)}, d_l(A_{m_2}^{(2)}) = d_l(A_l^{(2)}) + 1, d_h = d_h(A_m^{(2)})$ з коефіцієнтами $A_m^{(2)} + A_l^{(2)}$. Для діапазону допустимих значень $A_{m_1}^{(2)}$ будуємо та додаємо у $A^{(2)}$ серію апроксимацій (5.11) для поточного моменту часу l . Апроксимація $A_{m_1}^{(2)}$ надалі не модифікується і використовується тільки для обчислення значень $\Delta_l^{(\alpha)} u$;

3) Якщо у множині $A^{(2)}$ існують апроксимації, для яких $d_l < l$ і $d_h > l$, знаходимо $b_m = \max_{A_m \in A^{(2)}} d_h(A_m)$, будуємо та додаємо до $A^{(2)}$ серію апроксимацій (5.11) для поточного моменту часу l та $l < i < b_m$, причому, для останньої апроксимації серії встановлюємо $d_h = b_m$. Апроксимації, для яких $d_l < l$ та $d_h > l$ надалі не модифікується і використовується тільки для обчислення значень $\Delta_l^{(\alpha)} u$;

4) Якщо $\max_{A_m^{(1)} \in A^{(1)}} d_l(A_m^{(1)}) \geq i^{(1)}(l, k_m)$, будуємо апроксимацію (5.10) для поточного моменту часу l та додаємо значення її коефіцієнтів до коефіцієнтів апроксимації $A_m^{(1)}$ з найбільшим значенням $d_l(A_m^{(1)})$;

5) Якщо $\max_{A_m^{(1)} \in A^{(1)}} d_l(A_m^{(1)}) < i^{(1)}(l, k_m)$, будуємо та додаємо до $A^{(2)}$ серію апроксимацій (5.11) для поточного моменту часу l та $\max_{A \in A^{(2)}} d_h(A) < i < i^{(1)}(l, k_m)$.

Будуємо та додаємо до $A^{(1)}$ апроксимацію (5.10) $A_l^{(1)}$ для поточного моменту часу l і встановлюємо $d_l(A_l^{(1)}) = \max_{A_m^{(2)} \in A^{(2)}} d_h(A_m^{(2)}) + 1$.

Загальна похибка ε_5 наближення апроксимації дробової похідної (5.7) при $i = n$ тут може бути оцінена як

$$\varepsilon(|A_n^{(1)}| + |A_n^{(2)}|) < \frac{|\varepsilon_5| \tau^\alpha \Gamma(2 - \alpha)}{\sum_{r=0}^i |u^{(r+\gamma)} - u^{(r)}|} \leq \left(\sum_{A^{(1)} \in A_n^{(1)} \cup A_n^{(3)}} |\varepsilon_1(n, r(A^{(1)}), k_m)| + \sum_{A^{(2)} \in A_n^{(2)} \cup A_n^{(4)}} |\varepsilon_2(n, r(A^{(2)}), b(A^{(2)}), k_{m2})| \right),$$

де $A_n^{(1,2)}$ – поточні множини апроксимацій, $A_n^{(3)}$ та $A_n^{(4)}$ – множини апроксимацій (5.10) та (5.11), коефіцієнти яких були додані до коефіцієнтів інших апроксимацій на кроках 2 і 4 зазначеної процедури при $i \leq n$, $|A_n^{(1,2,3,4)}|$ – потужності відповідних множин.

Теорема 5.1. Кількість арифметичних операцій при виконанні обчислень згідно з процедурою 1 за умов, що

- 1) потужність $|A^{(1)}|$ множини $A^{(1)}$ збільшується в процесі обчислень, а максимальне її значення дорівнює кількості виконань перетворень на кроці 5 процедури,
 - 2) підрозбиття не проводиться на кроці 2 процедури і, відповідно, $k_{m2} = k_m$,
 - 3) не проводиться побудова серій апроксимацій на кроці 3,
- має порядок $P = O(\ln n)$.

Доведення.

Обчислення значень функції (5.8) за апроксимаціями (5.10) та (5.11) вимагає виконання k_m арифметичних операцій. Відповідно, обчислення значення $\Delta_i^{(\alpha)} u$ при $i = n$ за відомою апроксимацією вимагає виконання не більше ніж [44]

$$N_1(n, k_m, k_{m2}) \leq k_m \max_i |A_i^{(1)}| + k_{m2} \max_i |A_i^{(2)}|, \quad i = 1, \dots, n \quad (5.13)$$

операцій.

Нехай $i_m^{(1)}(k) = i^{(1)}(i_{m-1}^{(1)}(k), k)$, $i_0^{(1)} = 1$ – послідовність нижніх меж застосування серії апроксимацій (5.10), що будується на кроці 5, а $m_0(n, k) : i_{m_0}^{(1)}(k) = \min_m i_m^{(1)}(k)$, $i_m^{(1)}(k) < n$ – мінімальна їх кількість, необхідна для наближення із заданою точністю функції $g(r, i)$ при $i = n$. Тоді, відштовхуючись від умови 1, $\max_i |A_i^{(1)}| \leq m_0(n, k)$ оскільки $i_j^{(1)}(k) \geq d_l(A_j^{(1)})$, де $A_j^{(1)}$ – апроксимація (5.10) додана у множину $A^{(1)}$ при j -ому виконанні перетворень на кроці 5 процедури.

Аналогічно, для побудови оцінки максимальної потужності множини $A^{(2)}$ розглянемо серію апроксимацій (5.11), яка будується на кроці 2. Нехай

$$i_m^{(2)}(k) = b_m + i^{(2)}(1, b_m, k), \quad b_m - i^{(2)}(1, b_m, k) = b_{m-1} + i^{(2)}(1, b_{m-1}, k), \quad m > 1,$$

$$b_1 - i^{(2)}(1, b_1, k) = 1,$$

де b_i – послідовність значень параметра апроксимацій (5.11).

Враховуючи що $i^{(2)}(r, b, k) = i^{(2)}(r + v, b + v, k)$, кількість апроксимацій (5.11), необхідних для наближення функції $g(r, i)$ із заданою точністю при $r \leq i < r + v$ дорівнює $m_0^{(2)}(v, k) : i_{m_0^{(2)}-1}^{(2)}(k) < v \wedge i_{m_0^{(2)}}^{(2)}(k) \geq v$.

Враховуючи умови 2 та 3, маємо $\max_i |A_i^{(2)}| < m_0(n, k) \cdot (m_0^{(2)}(1, k) + m_0^{(2)}(n, k))$, і, з огляду на те, що $m_0^{(2)}(n, k) \geq m_0^{(2)}(1, k)$, (5.13) набирає вигляду

$$N_1(n, k_m) \leq k_m \cdot m_0(n, k_m) (1 + 2m_0^{(2)}(n, k_m)). \quad (5.14)$$

Для подальшого наближення $m_0(n, k)$ будемо відштовхуватися від того, що

$$i^{(1)}(r, k) > \tilde{i}^{(1)}(r, k) = \gamma_\varepsilon(k) (r-1)^{\frac{k-1}{\alpha+k-1}},$$

$$\gamma_\varepsilon(k) = \left(\frac{(-1)^k (1-\alpha)(-\alpha)\dots(2-\alpha-k)}{\varepsilon \cdot (k-1)!} \right)^{1/(\alpha+k-1)}.$$

Тоді, враховуючи, що при $r \gg 1$ $(r-1)^\delta \approx r^\delta - 1$, $\delta = \frac{k-1}{\alpha+k-1} < 1$, отримуємо

$$i_m^{(1)}(k) \approx \tilde{i}_m^{(1)}(k) = \gamma_\varepsilon(k) \sum_{i=0}^{m-1} \delta^i = \gamma_\varepsilon(k) \frac{1-\delta^{m+1}}{1-\delta}, \quad m_0(n, k) \approx \tilde{m}_0(n, k) = \frac{1}{\ln \delta} \left(1 - \frac{(1-\delta) \ln n}{\ln \gamma_\varepsilon(k)} \right) - 1.$$

Відмітимо, що $\tilde{i}_m^{(1)}(k) \leq \tilde{i}_{\max}^{(1)}(k) = \gamma_\varepsilon(k)^{\frac{1}{1-\delta}}$, і відповідно

$$m_0(n, k) < \tilde{m}_{0, \max} = \tilde{m}_0(\tilde{i}_{\max}^{(1)}(k) - 1, k). \quad (5.15)$$

Для оцінки $m_0^{(2)}(n, k)$ будемо відштовхуватися від того, що

$$i^{(2)}(r, b, k) < \tilde{i}^{(2)}(r, b, k) = \delta_\varepsilon(k)(r - b)^{\frac{\alpha + k - 1}{k}},$$

$$\delta_\varepsilon(k) = \left(\frac{\varepsilon k!}{|(1 - \alpha)(-\alpha) \dots (2 - \alpha - k)|} \right)^{1/k}.$$

Тоді при $k \gg \alpha$ $\frac{\alpha + k - 1}{k} = 1$, $b_m > \tilde{b}_m = 1 + \frac{(1 + \delta_\varepsilon(k))^{m-1}}{(1 - \delta_\varepsilon(k))^m}$ та

$$i_m^{(2)}(k) < \tilde{i}_m^{(2)}(k) = 1 + \left(\frac{1 + \delta_\varepsilon(k)}{1 - \delta_\varepsilon(k)} \right)^m, \quad m_0^{(2)}(n, k) < \tilde{m}_0^{(2)}(n, k) = \frac{\ln(n-1)}{\ln \frac{1 + \delta_\varepsilon(k)}{1 - \delta_\varepsilon(k)}}. \quad (5.16)$$

З огляду на (5.15) та (5.16), (5.14) набирає вигляду

$$N_1(n, k_m) \leq k_m \cdot \tilde{m}_{0, \max} \cdot \left(1 + 2 \frac{\ln(n-1)}{\ln \frac{1 + \delta_\varepsilon(k_m)}{1 - \delta_\varepsilon(k_m)}} \right).$$

Оскільки складність побудови апроксимацій для одного кроку за часом дорівнює k_m , загальна складність обчислювальної процедури буде не більше, ніж [44]

$$N_2(n, k_m) \leq k_m + k_m \cdot \tilde{m}_{0, \max} \cdot \left(1 + 2 \frac{\ln(n-1)}{\ln \frac{1 + \delta_\varepsilon(k_m)}{1 - \delta_\varepsilon(k_m)}} \right) \quad (5.17)$$

з порядком складності

$$P = O(\ln n). \quad (5.18)$$

□

Варто також відзначити, що для обчислення різницевої апроксимації дробової похідної необхідно збереження всіх попередньо обчислених до поточного кроку за

часом значень шуканої функції у всіх комірках сітки. Обсяг пам'яті, використовуваної алгоритмом розв'язання, у даному випадку пропорційний обчислювальній складності й, відповідно, лінійно залежить від номера кроку. У разі застосування вищеописаної процедури, обсяг необхідної пам'яті залежить від номера кроку за часом логарифмічно, що дозволяє зменшити обмеження на застосовність алгоритму, які накладаються обчислювальним середовищем.

5.3.2. Процедура розв'язання у випадку змінної довжини кроків за часом

Динамічна в процесі обчислень зміна довжини кроку за часом – це один із часто використовуваних підходів до підвищення точності та швидкодії при моделюванні різношвидкісних сумісних процесів.

У разі змінних довжин кроків за часом, апроксимація дробової похідної (5.7) згідно з [44] має вигляд

$$\bar{D}_x^\alpha u = \sum_{r=1}^{i-1} \left[g(t_r, t_{r+1}, t_i) (u^{(r+\gamma)} - u^{(r)}) \right], \quad (5.19)$$

$$g(r_1, r_2, i) = (i - r_1)^{1-\alpha} - (i - r_2)^{1-\alpha}, \quad 0 \leq \alpha \leq 1, r \geq 0, i \geq 0, \quad (5.20)$$

де t_i – моменти часу.

Аналогічно випадку постійних довжин кроків, будемо будувати наближення функції (5.20) та апроксимації дробової похідної (5.19) у вигляді

$$g(r_1, r_2, i) \approx \sum_{k=0}^{k_m} f_k^{(1)}(r_1, r_2) f_k^{(2)}(i), \quad (5.21)$$

$$\bar{D}_x^\alpha u = \sum_{k=0}^{k_m} \left(f_k^2(t_i) \sum_{r=1}^{i-1} \left[f_k^{(1)}(t_r, t_{r+1}) (u^{(r+\gamma)} - u^{(r)}) \right] \right). \quad (5.22)$$

Розглянемо апроксимації функції (5.20), побудовані на основі узагальненої формули бінома Ньютона та у вигляді ряду Тейлора по i в точці $i = b$. В першому випадку

$$g(r_1, r_2, i) = \sum_{k=0}^{k_m} g_k^{(1)}(i, r_1, r_2), \quad g_k^{(1)}(i, r_1, r_2) = f_k^{(1,1)}(r_1, r_2) f_k^{(2,1)}(i), \quad (5.23)$$

$$f_k^{(2,1)}(i) = i^{1-\alpha-k}, \quad f_k^{(1,1)}(r_1, r_2) = \frac{(1-\alpha)(-\alpha)\dots(2-\alpha-k)}{k!} \left((-r_1)^k - (-r_2)^k \right).$$

При фіксованих r_1, r_2 , $g_k^{(1)}(i, r_1, r_2) \leq \varepsilon$ при

$$i \geq i^{(1)}(r_1, r_2, k),$$

$$i^{(1)}(r_1, r_2, k) = \left(\frac{(1-\alpha)(-\alpha)\dots(2-\alpha-k)((-r_1)^k - (-r_2)^k)}{\varepsilon \cdot k!} \right)^{1/(\alpha+k-1)}. \quad (5.24)$$

Апроксимація функції (5.20) рядом Тейлора має вигляд [44]

$$g(r_1, r_2, i) = \sum_{k=0}^{k_m} g_k^{(2)},$$

$$g_k^{(2)}(i, r_1, r_2, b) = f_k^{(1,2)}(r_1, r_2, b) f_k^{(2,2)}(i, b), \quad f_k^{(2,2)}(i, b) = (i-b)^k, \quad (5.25)$$

$$f_k^{(1,2)}(r_1, r_2, b) = \frac{(1-\alpha)(-\alpha)\dots(2-\alpha-k)}{k!} \left((b-r_1)^{1-\alpha-k} - (b-r_2)^{1-\alpha-k} \right).$$

При фіксованих r_1, r_2 , $|g_k^{(2)}(i, r_1, r_2, b)| \leq \varepsilon$ при

$$|i-b| \leq i^{(2)}(r_1, r_2, b, k),$$

$$i^{(2)}(r_1, r_2, b, k) = \left(\frac{\varepsilon k!}{|(1-\alpha)(-\alpha)\dots(2-\alpha-k)((b-r_1)^{1-\alpha-k} - (b-r_2)^{1-\alpha-k})|} \right)^{1/k}. \quad (5.26)$$

Загальний алгоритм побудови наближення функції (5.20) та апроксимації дробової похідної (5.19) з урахуванням (5.21)–(5.26) залишається незмінним.

Необхідну для проведення моделювання з заданою точністю довжину кроку за часом будемо визначати методом проб та помилок [243, 244]. Встановивши помилку ε_6 та початкову довжину кроку τ_0 , на кожній ітерації шляхом зменшення або збільшення кроку за часом τ знаходимо таку його довжину, при якій розв'язок задачі при зменшенні τ вдвічі змінюватиметься не більше ніж на ε_6 .

Позначимо як $S(\tau, i)$ розв'язок задачі, отриманий на наступних від поточного i кроках з довжиною кроку τ . Тоді алгоритм зміни довжини кроку за часом на поточній ітерації можна описати таким чином:

- $\varepsilon_\tau = \|S(\tau, 1) - S(\tau/2, 2)\|;$
- Якщо $\varepsilon_\tau \geq \varepsilon_6$, зменшуємо вдвічі довжину кроку τ доти, доки $\|S(\tau, 1) - S(\tau/2, 2)\| \geq \varepsilon_6;$

- Якщо $\varepsilon_\tau < \varepsilon_6$, збільшуємо вдвічі довжину кроку τ доти, доки $\|S(\tau,1) - S(\tau/2,2)\| < \varepsilon_6$.

При використанні такої процедури, зменшення ε_6 призводить до збільшення точності обчислень разом зі збільшенням витраченого часу. З огляду на, що на кожній ітерації, задача має бути розв'язана мінімум три рази, ефективність алгоритму в плані швидкодії повинна бути додатково експериментально досліджена.

5.3.3. Швидкодія алгоритмів

Експериментальна оцінка швидкодії процедур апроксимації проводилася на кластері СКІТ-3 Інституту кібернетики НАН України з використанням OpenMP-багатопотокового розпаралелювання при проведенні розрахунків по рядках та стовпцях скінченно-різницевої сітки [44]. Вихідні коди розробленого програмного забезпечення наведено у [265]. Розв'язувалась задача (2.49)–(2.52).

У серії обчислювальних експериментів, фіксувався час, витрачений на проведення обчислень, у залежності від номера кроку для сітки розміру 120x120 (рис. 5.15).

Тут для графіка 1 обчислення значень дробової похідної проводилися шляхом повного підсумовування, для графіка 2 – з використанням процедури апроксимації зі значенням параметру точності $\varepsilon = 10^{-7}$ ($k_m = 10$, $k_{m2} = 20$), а для графіка 3 – з використанням процедури апроксимації при $\varepsilon = 10^{-10}$ ($k_m = 20$, $k_{m2} = 40$).

Отримані результати підтверджують теоретичну оцінку (5.18) для складності запропонованої процедури апроксимації, проте практична ефективність процедури досягається тільки для кількості кроків за часом більше ніж ~ 700 . Тестування на сітках розміру 40x40 та 80x80 також підтверджує ці висновки.

Істотні відмінності у часі роботи апроксимаційного алгоритму на сусідніх ітераціях пояснюються його залежністю від вихідних даних: кількість операцій побудови нових апроксимацій у розглянутому алгоритмі залежить від попередніх множин апроксимацій.

У проведених обчислювальних експериментах по використанню процедури розв'язання зі змінними кроками за часом проводилося розв'язання задачі на сітці

розміру 250×25 при $d_T = 0,0005$ і значеннях інших параметрів таких же, як і у попередньому експерименті.\

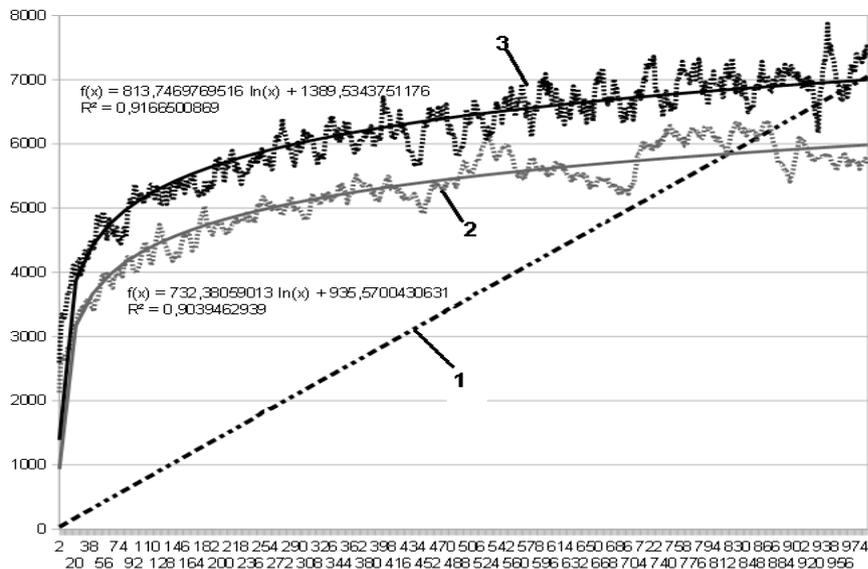


Рисунок 5.15 – Час, мс, витрачений на проведення обчислень, в залежності від номера кроку за часом

У рамках першого експерименту, задача розв'язувалась без використання процедури апроксимації в діапазоні часу $T = [10^{-5}, \dots, 10^{-4}]$ з фіксованим кроком $5 \cdot 10^{-8}$ та зі змінними кроками при $\varepsilon_6 = 0,01; 0,0075; 0,005$. Початковий крок за часом, який використовувався в алгоритмі з динамічною його зміною, вибирався рівним фіксованому кроку. Час роботи алгоритмів наведено на рис. 5.16, а різниці в розв'язках для полів температури та концентрації – на рис. 5.17.

Тоді як час роботи алгоритму з фіксованим кроком за часом лінійно збільшується, у випадку змінних кроків за часом, він зменшується на початковому етапі, коли довжина кроків збільшується, і починає збільшуватися тоді, коли довжина кроків стає постійною. Точність розв'язання, що оцінюється сумою квадратів різниць розв'язків, отримуваних алгоритмом з постійним кроком та алгоритмом зі змінними кроками, безпосередньо залежить від його довжини й, відповідно, погіршується в процесі розрахунків.

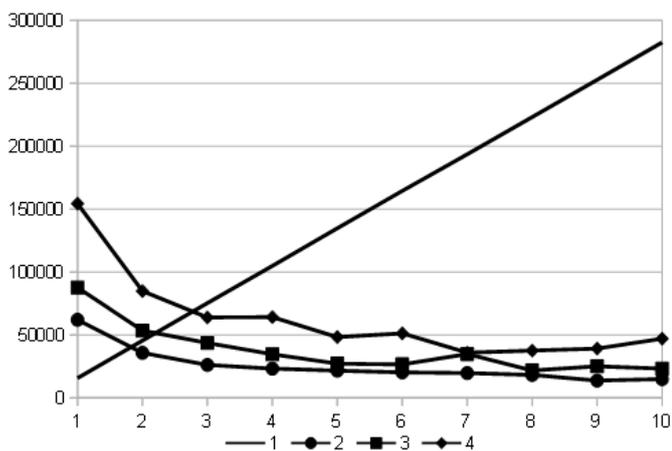


Рисунок 5.16 – Час роботи алгоритмів, мс (1 – фіксований крок за часом, 2,3,4 – змінюваний крок за часом при $\varepsilon_6 = 0,01; 0,0075; 0,005$)

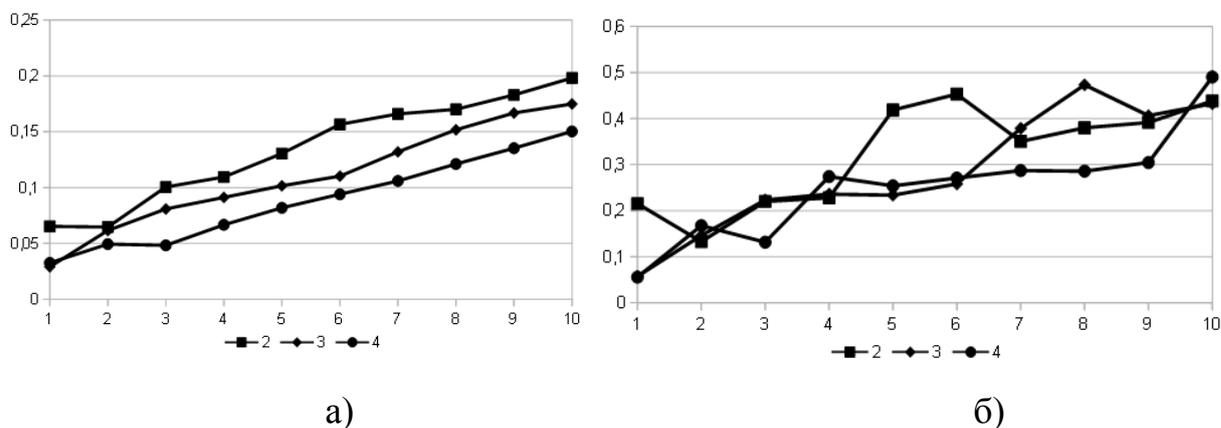


Рисунок 5.17 – Сума квадратів різниць розв'язків для полів температури (а) та концентрації (б) (2,3,4 – змінюваний крок за часом при $\varepsilon_6 = 0,01; 0,0075; 0,005$)

Таким чином, при моделюванні початкового етапу розвитку процесу, процедура зі змінним кроком за часом є ефективною в плані швидкодії при застосуванні неапроксимованого наближеного обчислення дробової похідної.

У рамках другого експерименту, задача розв'язувалась з та без використання процедури апроксимації у діапазоні часу $T = [10^{-4}, \dots, 10^{-3}]$ з фіксованими кроками $5 \cdot 10^{-6}$ ("грубий" розв'язок) та $1,25 \cdot 10^{-8}$ ("точний" розв'язок), а також зі змінним кроком при $\varepsilon_6 = 0,01$.

Крок за часом для отримання "точного" розв'язку (який був використаний як вихідний у процедурі зі зміною довжиною кроку) був збільшений в порівнянні з

першим експериментом через високу обчислювальну складність при моделюванні на момент часу $T = 10^{-3}$. Крок за часом, з яким обчислювався "грубий" розв'язок, вибирався рівним найбільшому кроку, отриманому в процесі роботи процедури з його зміною. Варто зазначити, що, при $T > 10^{-4}$, довжина цього кроку досягає свого максимуму.

Динамічна зміна кроку за часом при розв'язанні задачі на досліджуваному у другому експерименті часовому інтервалі, очікувано дозволяє отримувати наближений розв'язок швидше, ніж у випадку фіксованого кроку отримується "точний" розв'язок. Однак, це прискорення, у випадку використання процедури апроксимації, є тільки дворазовим, тоді як отримання "грубого" розв'язок у 400 разів швидше отримання "точного". При цьому, розв'язок, отриманий процедурою із змінним кроком за часом, є близьким до "точного" тільки при $T \leq 2 \cdot 10^{-4}$, а при подальшому проведенні моделювання його різниця з "точним" розв'язком зростає разом з наближенням до "грубого".

Отже, в діапазоні часу $T = [10^{-4}, \dots, 10^{-3}]$ динамічна зміна кроку за часом разом з використанням процедури апроксимації є ефективною лише на початкових етапах моделювання. Надалі, точність отриманих нею розв'язків наближається до точності "грубого" розв'язку при значно більшому витраченому часі. Така тенденція зберігається і у межах третього проведеного експерименту для діапазону часу $T = [10^{-3}, \dots, 10^{-2}]$ де використання процедури зі змінним кроком за часом є неефективним.

В цілому, проведені обчислювальні експерименти показали практичну ефективність застосування процедури, описаної у п. 5.3, при розв'язанні задач на великих часових інтервалах. Для невеликої кількості кроків за часом через високе значення коефіцієнту логарифмічної залежності, застосування запропонованих апроксимацій не дозволяє прискорити процес обчислень. Використання процедури зміни довжини кроку за часом при цьому ефективно лише при моделюванні початкових стадій процесу. При продовженні моделювання, час, що витрачається на підбір кроку, зростає і процедура стає неефективною.

5.4. Паралельний алгоритм та алгоритм оптимізації обчислень при моделюванні дробово-диференціальної за просторовими змінними гідрогеоміграційної динаміки

5.4.1. Методика наближеного обчислення дробової похідної за просторовою змінною

Кількість операцій підсумовування при обчисленні значення $\Delta_x^{(\alpha+1)}u_i$ дискретного аналога $\Delta_x^{(\alpha+1)}$ дробової похідної Капуто–Герасимова $D_x^{(\alpha+1)}$ за просторовою змінною (будемо розглядати дискретизацію вигляду (4.14)) при фіксованому i має порядок $O(i^2)$, а для всіх $i=1, \dots, n - O(n^3)$, що істотно впливає на швидкість обчислень. Для зниження порядку складності при послідовному обчисленні значень $\Delta_x^{(\alpha+1)}$ пропонується апроксимувати часткові суми степеневими рядами та модифікувати їх у процесі розрахунків [46, 269].

Аналогічно описаному в пп. 5.3.1 для дробової похідної за часовою змінною, якщо апроксимувати функцію $g(r, i)$ у вигляді ряду $g(r, i) = \sum_{k=0}^{k_m} f_k^{(1)}(r) f_k^{(2)}(i)$, маємо

$$\Delta_x^{(\alpha+1)}u_i = \frac{1}{h_1^2} \sum_{k=0}^{k_m} \left(f_k^{(2)}(i) \sum_{r=1}^{i-1} \left[f_k^{(1)}(r) (u_{r+1}^j - 2u_r^j + u_{r-1}^j) \right] \right). \quad (5.27)$$

Для отримання представлення (5.27) будемо розглядати апроксимації (5.10) та (5.11) при $a=1, c=0$.

Розглянемо наступні властивості $i^{(1)}$ та $i^{(2)}$:

- $i^{(2)}$ зменшується коли r прямує до b . Відповідно, якщо $|g_k^{(2)}(i, r, b)| \leq \varepsilon$, то при $r_1 < r$ також виконується $|g_k^{(2)}(i, r_1, b)| \leq \varepsilon$;

- $i^{(1)}$ зменшується при зменшенні r . Відповідно, якщо $|g_k^{(1)}(i, r)| \leq \varepsilon$, то при $r_1 < r$ також виконується $|g_k^{(1)}(i, r_1)| \leq \varepsilon$.

Розіб'ємо діапазон $1 \leq r \leq i-1$ на n_b блоків розміру s_b та будемо розглядати (5.27) у вигляді

$$\Delta_x^{(\alpha+1)} u = \frac{1}{h_1^2} \sum_{c=0}^{n_b-1} S_c(i), \quad (5.28)$$

$$S_c(i) = \sum_{r=r_{c0}}^{r_{c1}} \left[g(r,i) (u_{r+1}^j - 2u_r^j + u_{r-1}^j) \right], r_{c0} = 1 + c \cdot s_b, r_{c1} = r_0 + s_b.$$

З огляду на властивості $i^{(1)}$ та $i^{(2)}$, при побудові апроксимації $S_c(i)$, будемо серію апроксимацій функції $g(r,i)$ за алгоритмом, описаним у пп. 5.3.1, після чого використовуючи (5.27) отримуємо апроксимацію $S_c(i)$.

При послідовній блоковій апроксимації та обчисленні значень $\Delta_x^{(\alpha+1)} u_i, i=1, \dots, n$ на кожному кроці, при обробці блоку c , змінюємо та доповнюємо апроксимацію $\Delta_x^{(\alpha+1)} u_i$ для $r=1, \dots, r_{(c-1)1}, i > r_{(c-1)1}$ таким чином, щоб утворити апроксимацію для $r=1, \dots, r_{c1}, i > r_{c1}$. Поточна апроксимація, аналогічно описаному в пп. 5.3.1 для дробової похідної за часовою змінною, складається з множини $A^{(2)}$ апроксимацій (5.10) та множини $A^{(1)}$ апроксимацій (5.11) для кожної з яких встановлюються межі їх застосовності (d_l, d_h) . Значення $\Delta_x^{(\alpha+1)} u_i$ обчислюється як сума за тими апроксимаціями, для яких i лежить у межах їх застосовності.

Аналогічно випадку дробової похідної за часовою змінною, блокова процедура перебудови апроксимацій (**процедура 2**) наступна [46]:

1) Видаляємо з множини $A^{(2)}$ апроксимації, для яких $d_h(A_m^{(2)}) < r_{c1}, A_m^{(2)} \in A^{(2)}$;

2) Для всіх $A_m^{(2)} \in A^{(2)} : d_l(A_m^{(2)}) > r_{c1}$, будемо апроксимацію (5.11) $A_c^{(2)}$

поточного блоку c для $b = b(A_m^{(2)}), r = r_{c1}$ та додаємо її коефіцієнти до коефіцієнтів $A_m^{(2)}$. У разі, якщо існує

$k_m < k_l \leq k_{m2} : b(A_m^{(2)}) - i^{(2)}(r_{c1}, b(A_m^{(2)}), k_l) \leq d_l(A_m^{(2)})$, де k_{m2} – максимально допустима

кількість членів ряду в апроксимації, апроксимація (5.11) $A_c^{(2)}$ поточного блоку будується для $k = k_l$. Якщо виконується нерівність

$b(A_m^{(2)}) - i^{(2)}(r_{c1}, b(A_m^{(2)}), k_{m2}) > d_l(A_m^{(2)})$, апроксимація $A_m^{(2)}$ розбивається на дві:

$A_{m_1}^{(2)}, d_l(A_{m_1}^{(2)}) = d_l(A_m^{(2)}), d_h = d_l(A_c^{(2)})$ з коефіцієнтами $A_m^{(2)}$ та

$A_{m_2}^{(2)}, d_l(A_{m_2}^{(2)}) = d_l(A_c^{(2)}) + 1, d_h = d_h(A_m^{(2)})$ з коефіцієнтами $A_m^{(2)} + A_c^{(2)}$. Для діапазону допустимих значень $A_{m_1}^{(2)}$ будуюмо та додаємо до $A^{(2)}$ серію апроксимацій (5.11) поточного блоку c . Апроксимація $A_{m_1}^{(2)}$ надалі не модифікується та використовується тільки для обчислення значень $\Delta_x^{(\alpha+1)} u_i$;

3) Якщо у множині $A^{(2)}$ існують апроксимації для яких $d_l < r_{c1}$ і $d_h > r_{c1}$, знаходимо $b_m = \max_{A \in A^{(2)}} d_h(A)$, будуюмо та додаємо до $A^{(2)}$ серію апроксимацій (5.11) поточного блоку c для $r_{c1} < i < b_m$, причому, для останньої апроксимації серії встановлюємо $d_h = b_m$. Апроксимації, для яких $d_l < r_{c1}$ та $d_h > r_{c1}$, надалі не модифікуються та використовується тільки для обчислення значень $\Delta_x^{(\alpha+1)} u_i$;

4) Якщо $\max_{A_m^{(1)} \in A^{(1)}} d_l(A_m^{(1)}) \geq i^{(1)}(r_{c1}, k_m)$, будуюмо апроксимацію (5.10) для поточного блоку та додаємо її коефіцієнти до коефіцієнтів апроксимації $A_m^{(1)}$ з найбільшим значенням $d_l(A_m^{(1)})$;

5) Якщо $\max_{A_m^{(1)} \in A^{(1)}} d_l(A_m^{(1)}) < i^{(1)}(r_{c1}, k_m)$, будуюмо та додаємо до $A^{(2)}$ серію апроксимацій (5.11) поточного блоку c для $\max_i d_h(A_i^{(2)}) < i < i^{(1)}(r_{c1}, k_m)$. Будуюмо та додаємо до $A^{(1)}$ апроксимацію (5.10) $A_c^{(1)}$ поточного блоку c та встановлюємо $d_l(A_c^{(1)}) = \max_{A_m^{(2)} \in A^{(2)}} d_h(A_m^{(2)}) + 1$.

Складність проведення обчислень за процедурою 2, аналогічно описаному в пп. 5.3.1, матиме порядок, визначений згідно з (5.18).

Твердження 5.10. Враховуючи, що складність побудови апроксимацій для одного блоку дорівнює $k_m \cdot s_b$, загальна обчислювальна складність процедури 2, виходячи з (5.17), буде мати верхню оцінку [46]

$$N_2(n, k_m) \leq k_m \cdot s_b + k_m \cdot n \cdot \tilde{m}_{0, \max} \cdot \left(1 + 2 \frac{\ln(n-1)}{\ln \frac{1 + \delta_\varepsilon(k_m)}{1 - \delta_\varepsilon(k_m)}} \right). \quad (5.29)$$

5.4.2. Паралельні алгоритми та оцінки їх ефективності

У випадку розв'язання двовимірної задачі [46], відповідної одновимірній задачі, описаній у п. 4.1, локально-одновимірним методом, обчислення розглянутих сум проводяться незалежно для кожного рядка комірок сітки, після чого так само незалежно для кожного стовпця.

З огляду на високу обчислювальну складність алгоритму розв'язання, актуальним є його розпаралелювання на системах з розподіленою пам'яттю. Розглянемо дві схеми розподілу даних:

- Одновимірну, блоками рядків, з перерозподілом даних у розподіл блоками стовпців. У цій схемі, спочатку проводяться незалежні обчислення в рядках сітки, а потім перерозподіл даних та незалежні обчислення у стовпцях. Обмін даними тут є глобальним (MPI операція All-to-All), але проводиться тільки значеннями шуканої функції u ;

- Двовимірну, червоно-чорну, в якій обчислення, як по рядках, так і по стовпцях, проводяться за P (P – кількість процесів) кроків з обміном даними між сусідніми за рангом процесами. На кожному кроці, процеси обмінюються розрахованими апроксимаціями для кожного рядка або стовпця. Обміни в цьому випадку є локальними.

Прискорення від розпаралелювання виконання арифметичних операцій в обох схемах розподілу даних є лінійним, тому при порівнянні швидкодії основним фактором є час, що витрачається на обміни даними.

Твердження 5.11. При одновимірній схемі розподілу сітки розміру $n \times n$ та реалізації перерозподілу операціями парного обміну, обсяг даних, який передається одним процесом, має порядок

$$T_1 = O\left(\frac{n^2}{P}\right). \quad (5.30)$$

Твердження 5.12. При використанні двовимірної схеми розподілу даних, їх обсяг, що передається одним процесом, відштовхуючись від (5.14) та (5.18), має порядок

$$T_2 = O\left(\frac{n}{P} \sum_{i=1}^{P-1} \ln\left(i \frac{n}{P}\right)\right) = O\left(\frac{n}{P} \ln \frac{n(P-1)!}{P}\right). \quad (5.31)$$

Кількість операцій обміну в обох випадках дорівнює $N_{1/2} = P - 1$.

При невеликому P , двовимірна схема розподілу даних має більшу швидкодію, ніж одновимірна, внаслідок меншого обсягу даних, якими обмінюються процеси. При збільшенні P , обсяг даних у випадку одновимірної схеми зменшується, а у випадку двовимірної – зростає, що призводить до її неефективності.

При істотному збільшенні кількості залучених процесів, у випадку одновимірної схеми, збільшується кількість операцій обміну і, відповідно, час, що витрачається на допоміжні операції.

Для зменшення кількості операцій обміну може бути використана наступна параметрична схема розподілу даних, що поєднує в собі вищеописані одновимірну та двовимірну схеми [46]:

- При проведенні обчислень по рядках сітки, стовпці розбиваються на блоки розміром $K \frac{n^2}{P^2}$ і у межах блоку стовпців j процес i обробляє блок рядків $(i + j + K)\%P$, де $\%$ – залишок від ділення. Обчислення, аналогічно схемі з двовимірним розподілом даних, проводяться за $\frac{P}{K}$ кроків з обміном даними після кожного з них;

- Після проведення обчислень по рядках сітки, дані перерозподіляються таким чином, щоб отримати аналогічний розподіл по стовпцях. Для цього в кожному блоці стовпців, для кожного блоку з K блоків рядків проводиться операція перерозподілу, аналогічна тій, що застосовується в схемі з одновимірним розподілом даних.

Твердження 5.13. За параметричної схеми розподілу, обсяг даних, який передається одним процесом, дорівнює

$$T_3(P, n, K) = k_1(K-1) \frac{n^2}{P^2} + k_2 K \frac{n}{P} \ln \frac{Kn \left(\frac{P}{K} - 1\right)!}{P}. \quad (5.32)$$

При $K = P$, розглянута алгоритмічна схема буде збігатися зі схемою з одновимірним розподілом даних, а при $K = 1$ – зі схемою з двовимірним розподілом.

Кількість операцій обміну при такій схемі буде дорівнювати $N_3 = \frac{P}{K} + K - 2$.

Найменша кількість при цьому буде досягтися при $\bar{K} = \sqrt{P}$ і буде дорівнювати $\bar{N}_3 = 2\sqrt{P} - 2$.

Обсяг даних, що передається при використанні цієї схеми, буде більше, ніж при одновимірному розподілі, проте менше, ніж при двовимірному. При великій кількості залучених процесів, кількість операцій обміну тут істотно менше, ніж в одно- та двовимірній схемах.

Відзначимо, що незалежність обчислень по рядках та стовпцях сітки в локально-одновимірних методах дозволяє реалізовувати їх на графічних процесорах (GPU) за схемою "один потік – один рядок/стовпець" без об'єднання потоків в групи. Особливістю такого розпаралелювання є необхідність зберігати та обробляти в пам'яті графічного процесора структури змінного розміру – множини апроксимацій. Прискорення від розпаралелювання тут є лінійним та обмежується кількістю рядків/стовпців в блоці та необхідністю копіювання даних з пам'яті GPU в пам'ять центрального процесора. Останнє обмеження знімається за підтримки програмними та апаратними середовищами обміну даними прямого доступу до пам'яті GPU. Швидка локальна пам'ять GPU при цьому може використовуватися для прискорення роботи як програмно керований кеш. Оскільки при виконанні обчислень відбувається велика кількість викликів до степеневі функції, її швидкодія істотно впливає на загальну швидкодію GPU-алгоритму. Тому пропонується по можливості використовувати апаратну реалізацію степеневі функції додатного аргументу (native_powr у межах OpenCL-стандарту).

5.4.3. Швидкодія паралельних алгоритмів

Експериментальна оцінка швидкодії алгоритму, описаного в пп. 5.4.1, 5.4.2, проводилася на кластері СКІТ-3 Інституту кібернетики НАН України [46]. Вихідні коди розробленого програмного забезпечення наведено у [265]. Дані про час, що витрачається на отримання розв'язку на одному кроці за часом, наведені в табл. 5.1.

Отримані дані показують істотне, до 4,5 раз, прискорення при використанні вищеописаного методу апроксимації. Прискорення всіх трьох розглянутих паралельних алгоритмів у проведених експериментах було близьким до лінійного через велике відношення кількості арифметичних операцій до операцій доступу до пам'яті.

Таблиця 5.1 – Час, мс, отримання розв'язку на одному кроці за часом

Кількість процесів	Розмір сітки (N)	Час для алгоритму без апроксимації сум	Час для алгоритму з апроксимацією сум (розмір блоку – N/4)	Прискорення від використання апроксимації	Прискорення від розпаралелювання алгоритму без використання апроксимацій	Прискорення від розпаралелювання алгоритму з використанням апроксимацій
1	400	53560	38480	1,39		
1	1000	840370	365690	2,30		
1	2000	6733880	2388430	2,82		
4	400	13440	9650	1,39	3,99	3,99
4	1000	210730	91730	2,30	3,99	3,99
4	2000	1684940	597670	2,82	4,00	4,00
8	2000	846150	199170	4,25	7,96	11,99
16	800	27660	11690	2,37	15,55	16,93
16	2000	421550	90300	4,67	15,97	26,45
32	800	13850	6630	2,09	31,05	29,86

Дані щодо похибки оцінки часу виконання алгоритмів у послідовному режимі наведені у табл. 5.2 та показують високу точність отриманих оцінок, яка, однак, знижується при збільшенні розміру блоку для алгоритму, що використовує процедуру апроксимації.

При істотному збільшенні кількості залучених обчислювальних ресурсів, такому, як масштабування алгоритму на exascale-computing системи, вплив обмінів даними між процесами на швидкодію стає вирішальним, тому актуальним є експериментальна перевірка оцінок (5.30)–(5.32) кількості та обсягу обмінів. В процесі проведення експериментів був виміряний час, що витрачається на обміни даними алгоритмом, що використовує апроксимування, та обсяг цих обмінів. Дані щодо обсягу обмінів при використанні схеми з параметричним розподілом даних та $K = \sqrt{P}$ разом з його оцінками наведені в табл. 5.3.

Таблиця 5.2 – Точність оцінювання часу виконання алгоритмів

Розмір сітки	Розмір блоку при проведенні апроксимації	Час для алгоритму без апроксимації сум	Час для алгоритму з апроксимацією сум	Похибка оцінювання для алгоритму без апроксимації сум	Похибка оцінювання для алгоритму з апроксимацією сум
40	10	50	280	-8%	9%
100	25	840	1830	0%	-8%
200	50	6690	8270	0%	-10%
300	75	22560	19250	-1%	-15%
400	100	53560	38480	0%	-7%
600	150	183240	104190	1%	5%
800	200	430070	197960	0%	7%
1000	250	840370	365690	0%	19%
2000	500	6733880	2388430	0%	45%
2000	250	6755270	1585620	1%	18%
800	50	431430	184370	0%	0%
2000	125	6724670	1437940	0%	9%
800	25	431170	204270	0%	10%

Кількість обмінів даними при одновимірній схемі їх розподілу не залежить від вихідних даних та оцінюється точно. Що стосується двовимірної схеми, то точність оцінок падає при збільшенні розміру блоку, тоді як у разі параметричної – при зменшенні.

Оскільки залежність часу, що витрачається на локальні та глобальні обміни, від обсягу даних може відрізнятися, на основі вимірів часу, що витрачається на обміни, для розглянутої обчислювальної системи та алгоритмів пропонуються наступні залежності:

- $T(s, p) = k \cdot s \cdot \ln p$ для глобальних обмінів в схемі з одновимірним розподілом даних;

- $T(s, p) = k \cdot s$ для локальних обмінів в схемі з двовимірним розподілом даних;

- $T(s, p) = k \cdot s \cdot \left(1 + \ln \frac{p}{2}\right)$ для схеми з параметричним розподілом даних для випадку $K = \sqrt{P}$.

Дані, щодо часу, що витрачається на обміни, та його оцінок наведені в табл. 5.4.

Використовуючи отримані теоретичні та експериментальні залежності, було проведено оцінювання прискорення паралельних алгоритмів при використанні середовища обміну даними (Infiniband + OpenMPI), яке присутнє на кластері СКІТ-3, і при варіюванні кількості залучених обчислювальних ресурсів та їх швидкодії. Отримані результати стосовно ефективності розпаралелювання (прискорення, поділене на кількість процесів, що дорівнює одиниці при лінійному прискоренні) при розмірі сітки 10000x10000 та коефіцієнті швидкодії обчислювальної підсистеми, що дорівнює 10^{-3} та 10^{-4} наведені на рис. 5.18.

Таблиця 5.3 – Обсяг обмінів даними та його оцінки

Кількість процесів	Розмір сітки	Розмір блоку при проведенні апроксимування	Кількість операцій обміну	Оцінка	Точність оцінки
Схема з одновимірним розподілом даних					
8	1000	125	6125000		
8	4000	500	98000000		
16	800	50	2100000		
16	4000	250	52500000		
20	1000	50	2660000		
20	4000	200	42560000		
Схема з двовимірним розподілом даних					
8	1000	125	13089332	10849699	17%
8	4000	500	56854508	47904251	16%
16	800	50	13646380	10338671	24%
16	4000	250	72748392	54308690	25%
20	1000	50	18327844	14056870	23%
20	4000	200	76617068	58029662	24%
Схема з параметричним розподілом даних					
8	1000	125	7307352	7974390	-9%
8	4000	500	73174752	48490152	34%
16	800	50	4052000	7074213	-75%
16	4000	250	51536000	43771065	15%
20	1000	50	5309600	9473296	-78%
20	4000	200	45428800	43726372	4%

Для розглянутих оцінок, експериментально обчислений коефіцієнт швидкодії виконання арифметичних операцій на кластері СКІТ-3 має порядок 10^{-2} .

У такій ситуації, враховуючи обмеження на кількість залучених процесів, що накладається розміром сітки, прискорення, незалежно від залучених обчислювальних ресурсів, близьке до лінійного. Однак при збільшенні швидкодії обчислювальної підсистеми (зокрема, таке можливо при залученні графічних процесорів) за умови незмінності швидкодії мережевої підсистеми, її вплив на час виконання обчислень зростає.

Таблиця 5.4 – Час, що витрачають на обміни, та його оцінки

Кількість процесів	Обсяг обмінів	Час, мс	Оцінка часу, мс	Похибка оцінювання
Схема з одновимірним розподілом даних				
8	24500000	300	494	-65%
8	98000000	1230	1333	-8%
16	2100000	200	310	-55%
16	4725000	270	340	-26%
16	13125000	480	436	9%
16	52500000	990	885	11%
20	2660000	300	339	-13%
20	10640000	500	430	14%
20	42560000	1000	795	21%
Схема з двовимірним розподілом даних				
8	26276784	300	287	4%
8	56854508	800	699	13%
16	36732584	440	428	3%
16	72748392	830	914	-10%
20	39568608	390	466	-20%
20	76617068	1000	966	3%
Схема з параметричним розподілом даних				
8	20271264	160	282	-76%
8	73174752	540	692	-28%
16	15434000	210	257	-22%
16	51536000	640	537	16%
20	16401880	250	269	-8%
20	45428800	700	494	29%

З наведених на рис. 5.18 даних можна зробити висновки, що при розв'язанні задач великої розмірності на досить потужних кластерних системах [46]:

- Алгоритм з двовимірним розподілом у порівнянні з іншими розглянутими алгоритмами неефективний;

- Падіння ефективності алгоритму з параметричним розподілом даних швидше ніж для алгоритму з одновимірним розподілом при більшій ефективності першого на невеликій кількості залучених обчислювальних ресурсів;

- Ефективність алгоритму з одновимірним розподілом швидше ніж для інших алгоритмів зменшується при збільшенні швидкодії обчислювальної підсистеми.

Тестування швидкодії GPU-реалізації апроксимаційного алгоритму з використанням бібліотеки OpenCL проводилося при розв'язанні розглядуваної задачі на одному GPU GeForce GTX 670, а його результати порівнювалися з отриманими на одному вузлі кластера СКІТ-3. Дані про прискорення GPU-алгоритму наведені в табл. 5.5.

Таблиця 5.5 – Швидкодія та прискорення GPU-алгоритму

Розмір сітки	Кількість блоків	Розмір блоку	Час GPU-алгоритму, мс	Час CPU-алгоритму, мс	Прискорення
200	4	50	980	8270	8,43
400	4	100	4610	38480	8,34
600	4	150	12750	104190	8,17
800	16	50	33050	184370	5,57
400	1	400	7080	80191,6	11,33
400	4	100	4610	53560	11,62
400	8	50	4720	57256,58	12,13
400	16	25	5960	59442,16	9,97
400	40	10	8750	70019,24	8,00
400	100	4	12280	82080,36	6,68

Як видно з отриманих результатів, прискорення GPU-алгоритму при однаковій кількості блоків, незначним чином знижується при зростанні розмірів сітки. В межах одного розміру, при зростанні кількості блоків, швидкодія GPU-алгоритму падає аналогічно швидкодії CPU-алгоритму. Прискорення, що отримується завдяки використанню GPU, при цьому зростає до свого максимуму при розбитті на 8 блоків і зменшується при розбитті на більшу їх кількість. Такий ефект може бути пояснений нижчою ефективністю реалізації на GPU процедури апроксимації в порівнянні з неоптимізованими обчисленням значень дробової похідної.

Експериментально обчислений коефіцієнт швидкодії виконання арифметичних операцій при використанні GPU має порядок 10^{-3} і, при масштабуванні алгоритму,

що використовує процедуру апроксимації, на системи, обладнані GPU, одновимірний розподіл даних є максимально ефективним серед трьох розглянутих схем розподілу тільки при залученні досить великої кількості ресурсів. В інших випадках швидшою є схема параметричного розподілу.

Загалом, пропонувані обчислювальні схеми та додаткове ~ 10 -кратне прискорення розрахунків, що отримується при використанні графічних процесорів, дозволяють проводити дробово-диференціальне моделювання міграційних процесів в середовищах фрактальної структури на великих часових проміжках.

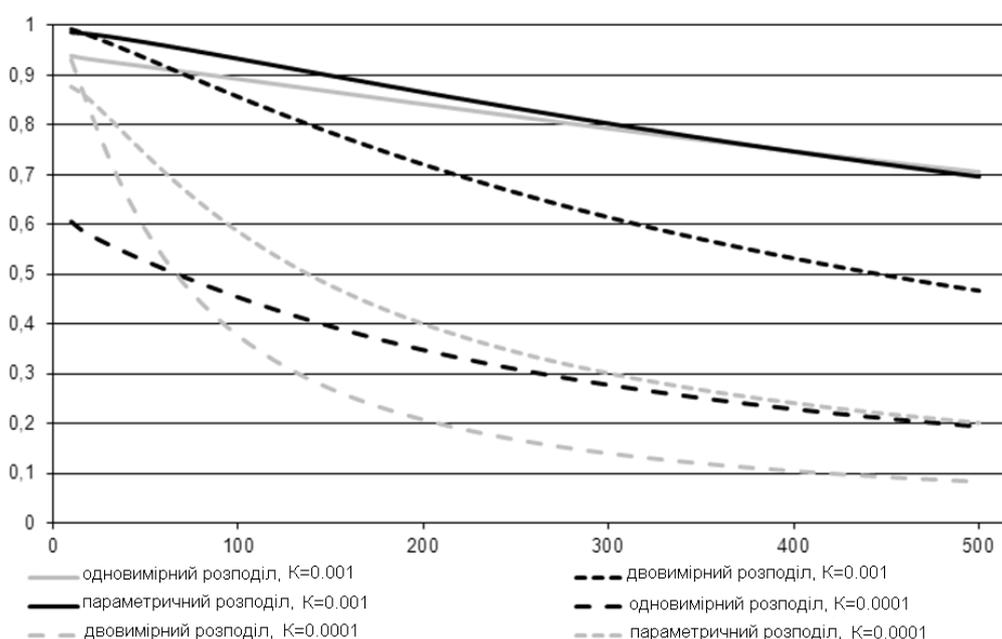


Рисунок 5.18 – Оцінка ефективності розпаралелювання при різних значеннях коефіцієнта K швидкодії обчислювальної підсистеми

5.5 Висновки до розділу 5

Той факт, що похідні дробового порядку є інтегро-диференціальними операторами збільшує обчислювальну складність розв'язання початково-крайових задач для моделей, рівняння яких їх містять, і робить актуальною розробку оптимізованих обчислювальних схем. У цьому контексті в даному розділі для одно- та багатовимірних задач моделювання геоміграційних процесів, що містять похідні Капуто–Герасимова та Капуто–Фабріціо запропонований клас паралельних алгоритмів та оптимізованих обчислювальних схем.

Так, у п. 5.1 розглядаються паралельні алгоритми для локально-одновимірних схем розщеплення застосованих до дробово-диференціальних рівнянь, які містять похідну Капуто–Герасимова за часовою змінною. Запропоновано використовувати у такому випадку червоно-чорний двовимірний блоковий розподіл даних та встановлено суттєвий вплив на швидкодію алгоритмів ефективності кешування даних. Досліджено використання багатопоточного розпаралелення при виконанні обчислень кожним процесом розподіленої програми і експериментально доведена більша швидкодія такої схеми у порівнянні з виконанням кожного процесу в одному потоці внаслідок зменшення обсягів обміну даними.

У п. 5.2 розглянуто паралельні алгоритми для графічних процесорів у випадку моделей з похідними Капуто–Герасимова як за часовою, так і за просторовими змінними.

У пп. 5.2.1 пропонується GPU-алгоритм для двовірних задач з похідною за часовою змінною, що розв'язуються за локально-одновимірною скінченно-різницевою схемою розщеплення. У пп. 5.2.2 розглядається аналогічна схема, у якій обчислення виконуються паралельно на GPU та CPU. Пропонується алгоритм динамічного перерозподілу блоків даних, оброблюваних GPU та CPU, та досліджується розпаралелення обчислень на декілька GPU в межах одного вузла. Результати тестування розроблених алгоритмів показують їх невелику ефективність (до 5% при паралельному проведенні обчислень на GPU та CPU, до 1,5% при використанні двох GPU, зі зниженням при зростанні номеру кроку за часом) через необхідність виконання великої кількості операцій копіювання даних між пам'яттями обчислювальних пристроїв.

У пп. 5.2.3 та 5.2.4 розглядаються паралельні алгоритми для GPU у випадку розв'язання задач щодо одновимірних моделей, які містять похідні Капуто–Герасимова. У пп. 5.2.3 пропонується алгоритм для випадку моделі, що містить відповідну похідну за просторовою змінною і демонструється лінійне зростання його прискорення при зростанні розміру сітки у проведених обчислювальних експериментах. У пп. 5.2.4 розглядаються питання оптимізації обчислення значень похідної Капуто–Герасимова при розв'язанні задач щодо моделей, які містять таку

похідну за часовою змінною. Пропонуються процедури організації GPU-потоків у групи задля використання швидкої локальної пам'яті, що дозволяють збільшити масштабованість базового алгоритму. Як і у випадку алгоритмів для двовимірних задач, розглянутих у пп. 5.2.2, досліджується ефективність використання декількох GPU в межах одного вузла для проведення обчислень. Показано, що підвищення швидкодії у цьому випадку суттєво обмежується необхідністю копіювання даних між пам'яттями графічних процесорів.

Для збільшення швидкодії обчислювальних схем моделювання процесів тепло-та масоперенесення на основі моделей з похідною Капуто–Герасимова за часовою змінною у п. 5.3 запропонована процедура її наближення з заданою точністю на основі розкладення у ряди та методики розділення змінних. Процедура дозволяє знизити обчислювальну складність розв'язання задач на одному кроці за часом з лінійної до логарифмічної. Проведені обчислювальні експерименти показали практичну ефективність застосування такої процедури при моделюванні на великих часових інтервалах.

Іншим дослідженням у п. 5.3 методом підвищення швидкодії обчислень є динамічна зміна довжини кроку за часом з її підбором методом проб та помилок. Ефективність цієї процедури була експериментально доведена при моделюванні початкових стадій процесів. При продовженні моделювання, час, що витрачається на підбір кроку, зростає і процедура стає неефективною.

Процедура, запропонована у п. 5.3 для рівнянь, що містять похідні Капуто–Герасимова за часовою змінною, була у п. 5.4 поширена на випадок двовимірної дробово-диференціальної моделі геоміграційних процесів, яка містить відповідні похідні за просторовими змінними. Розглядалися паралельні схеми розв'язання початково-крайової задачі, яка дискретизувалась за локально-одновимірною схемою, на системах з розподіленою пам'яттю. Були проаналізовані дві схеми розподілення даних – одновимірна блокова та червоно-чорна, що використовує розбиття сітки на двовимірні блоки. Визначено, що домінування у загальній обчислювальній складності алгоритму складової, що відповідає за обчислення дробової похідної за просторовими змінними, призводить до неефективності використання двовимірного

розподілу даних і загального зниження масштабованості алгоритму. Використання запропонованої апроксимаційної схеми у цій ситуації дозволило суттєво збільшити швидкодію алгоритмів та зменшити обсяг обмінів даними при використанні схем з двовимірним розподілом даних збільшуючи їх масштабованість.

Окрім цього, у п. 5.4 запропонована схема розподілення даних, що є поєднанням одновимірної та червоно-чорної схем, і має вищу за схему з одновимірним розподілом швидкодію при невеликій кількості залучених обчислювальних ресурсів, та більшу масштабованість у порівнянні з червоно-чорною схемою. Пропоновані обчислювальні схеми та додаткове ~ 10 -кратне пришвидшення розрахунків, отримане при застосуванні графічних процесорів, дозволяють проводити моделювання міграційних процесів у середовищах фрактальної структури на великих часових проміжках.

Результати розділу 5 опубліковано в роботах [44, 46, 53, 64, 67, 269].

Джерела, що використані у розділі 5

Для написання даного розділу було використано 19 джерел [14], [23], [34], [44], [46], [53], [64], [67], [91], [102], [109], [110], [243], [244], [250], [265], [267], [269], [276], посилання на які зазначені в тексті розділу.

Розділ 6. Високопродуктивні алгоритми моделювання гідрогеоміграційної динаміки на основі моделей, що містять ψ -похідну Капуто та похідну Анатгана–Балеану

Обчислення інтегралів, які виникають при дискретизації таких похідних дробового порядку, як ψ -похідна Капуто та похідна Анатгана–Балеану, вимагає використання спеціальних часозатратних чисельних методів. Специфікою ψ -похідної Капуто, що ускладнює обчислювальні процедури розв'язання рівнянь, які її містять, є також і наявність у ній функціонального параметру. Все це робить актуальним розвиток відповідного алгоритмічного апарату високопродуктивних обчислень. У даному розділі розглядаються нові алгоритми проведення обчислень значень ψ -похідної Капуто та похідної Анатгана–Балеану, зокрема такі, що базуються на методиці розкладення у ряди та розділення змінних, застосованої у п. 5.3, 5.4 до класичної похідної Капуто.

6.1. Методика наближеного обчислення ψ -похідної Капуто за часовою змінною

Будемо розглядати наступне одновимірне рівняння дифузії з дробовою похідною за часовою змінною, побудоване аналогічно описаному в [79]:

$$\sigma D_{t,g}^{(\beta)} C(x,t) = \frac{\partial}{\partial z} \left(d(x,t) \frac{\partial C(x,t)}{\partial z} \right), \quad x \in [0, L], t \geq 0, \beta \leq 1, \quad (6.1)$$

де $C(x,t)$ – концентрація речовини, σ – пористість середовища, $d(x,t)$ – коефіцієнт дифузії, $D_{t,g}^{(\beta)}$ – дробова ψ -похідна Капуто за часовою змінною t вигляду

$$D_{t,g}^{(\beta)} H(x,t) = \frac{1}{\Gamma(1-\beta)} \int_0^t \frac{\partial H(x,\tau)}{\partial \tau} (g(t) - g(\tau))^{-\beta} d\tau.$$

До рівняння (6.1) поставимо наступні початкові та крайові умови:

$$C(0,t) = 1, \left. \frac{\partial C(x,t)}{\partial x} \right|_{x=L} = 0, C(x,0) = 0.$$

На однорідній скінченно-різницевої сітці

$$\omega = \left\{ (x_i, t_j) : x_i = ih, t_j = j\tau, i = 0, \dots, m, j = 0, 1, \dots \right\} \quad (6.2)$$

наближення $\Delta_{t,g}^{(\beta)}$ першого порядку точності оператора $D_{t,g}^{(\beta)}$ має вигляд [69]

$$\Delta_{t,g}^{(\beta)} C = \frac{1}{\Gamma(1-\beta)} \sum_{s=0}^j b_s^{(j)} \frac{C^{s+1} - C^s}{\tau}, \quad b_s^{(j)} = \int_{t_s}^{t_{s+1}} (g(t_j) - g(\tau))^{-\beta} d\tau. \quad (6.3)$$

Дискретизуючи рівняння (6.1) на сітці (6.2) з використанням (6.3), отримуємо наступну тридіагональну СЛАР (значення C_0^{j+1} і C_m^{j+1} отримуються з крайових умов):

$$A_i^j H_{i-1}^{j+1} - R_i^j H_i^{j+1} + B_i^j H_{i+1}^{j+1} = \Omega_i^j \quad (i=1, \dots, m-1, j=0, 1, \dots), \quad (6.4)$$

де

$$A_i^j = \frac{d_{i-1}}{h^2}, \quad B_i^j = \frac{d_i}{h^2}, \quad R_i^j = \frac{d_{i-1} + d_i}{h^2} + \frac{\sigma b_j^{(j)}}{\tau \Gamma(1-\beta)},$$

$$\Omega_i^j = -\frac{\sigma b_j^{(j)}}{\tau \Gamma(1-\beta)} C^j + \frac{\sigma}{\tau \Gamma(1-\beta)} \sum_{s=0}^{j-1} b_s^{(j)} (C^{s+1} - C^s).$$

Зауважимо, що СЛАР (6.4) може бути ефективно розв'язана методом прогонки [313].

Визначення значень коефіцієнтів системи (6.4) потребує проведення розрахунку інтегралів $b_s^{(j)}$, які є сингулярними при $s=j-1$ і для наближеного визначення значень яких необхідно використовувати спеціальні чисельні методи. З цією метою пропонується наступний алгоритм чисельного інтегрування на основі методики рекурсивного підрозбиття [43]:

1) Поточний інтервал інтегрування $I = [I_l, I_u]$ встановлюється рівним $I = [t_s, t_{s+1}]$;

2) Значення інтегралу на поточному інтервалі інтегрування обчислюється за формулою трапеції;

3) Інтервал ділиться на дві частини: $I_1 = \left[I_l, \frac{1}{2}(I_l + I_u) \right]$

і $I_2 = \left[\frac{1}{2}(I_l + I_u), I_u \right]$. Значення інтегралу на інтервалі $I = [I_l, I_u]$ обчислюється як су-

ма значень інтегралів на інтервалах I_1, I_2 , які, в свою чергу, обчислюються за формулою трапеції;

4) Якщо різниця між наближеннями значень інтегралу, отриманими на етапах 2 і 3, більша за задану сталу ε_1 , процедура рекурсивно повторюється для інтервалів I_1, I_2 .

У випадку, коли $s = j - 1$ і інтеграл обчислюється на інтервалі $I_3 = [t_{s+1} - \varepsilon_2, t_{s+1}]$, де ε_2 – задана стала, пропонується використовувати наступний алгоритм [43]:

1) Початковий крок встановлюється рівним $s = \varepsilon_2 / N$, де N – задане ціле число; початкове значення нижньої межі інтервалу інтегрування встановлюється рівним $t_l = t_{s+1} - \varepsilon_2$.

2) Значення інтегралу на інтервалі $[t_l, t_l + s]$ обчислюється за квадратурною формулою 4-го порядку точності;

3) Якщо $F(t_l + s/2) / F(t_l) < F(t_l + s) / F(t_l + s/2)$, де F – інтегрована функція, зменшуємо крок s ділячи його на $F(t_l + s)F(t_l) / F^2(t_l + s/2)$. Таким чином в межах інтервалу $[t_l, t_l + s]$ інтегрована функція наближається до лінійної;

4) Крок зменшується вдвічі, якщо $t_l + s > t_{s+1}$;

5) Якщо $(t_{s+1} - t_l) < \varepsilon_3$, де ε_3 – задана стала, переходимо до наступного інтервалу, встановлюючи $t_l \leftarrow t_l + s$.

Згідно з вищеописаним алгоритмом розв'язання початково-крайової задачі, кількість визначень значень інтегралів $b_s^{(j)}$ лінійно збільшується зі збільшенням номеру кроку за часом. Для зменшення часу, що витрачається на обчислення інтегралів $b_s^{(j)}$, пропонується наступна рекурентна схема [43], заснована, аналогічно обчислювальним схемам, описаним у п. 5.3, 5.4, на розкладення інтегрованої функції у ряди та подальшому розділенні змінних.

Нехай існує нескінченно диференційована обернена функція $f(\tau): f(g(\tau)) = \tau$. Тоді, інтеграл $b_s^{(j)}$ може бути представленим у вигляді ряду двома способами, викладеними нижче.

Виконавши в $b_s^{(j)}$ заміну змінних $x = g(t_j) - g(\tau) \Rightarrow \tau = f(g(t_j) - x)$ отримуємо

$$b_s^{(j)} = \int_{t_s}^{t_{s+1}} (g(t_j) - g(\tau))^{-\beta} d\tau = \int_{g(t_j)-g(t_{s+1})}^{g(t_j)-g(t_s)} f'(g(t_j) - x) x^{-\beta} dx.$$

Розклавши $f'(x)$ у ряд Тейлора в точці $x = g(t_j)$ маємо

$$\begin{aligned} b_s^{(j)} &= \sum_{n=0}^{\infty} \left((-1)^n \frac{f^{(n+1)}(g(t_j))}{n!} \int_{g(t_j)-g(t_{s+1})}^{g(t_j)-g(t_s)} x^{n-\beta} dx \right) = \\ &= \sum_{n=0}^{\infty} \left((-1)^n \frac{f^{(n+1)}(g(t_j))}{n!(n-\beta+1)} \left[\begin{aligned} &(g(t_j) - g(t_s))^{n-\beta+1} \\ &- (g(t_j) - g(t_{s+1}))^{n-\beta+1} \end{aligned} \right] \right). \end{aligned} \quad (6.5)$$

Друга схема обчислення $b_s^{(j)}$ отримується виконуючи заміну змінних $x = g(\tau) \Rightarrow \tau = f(x)$. У цьому випадку маємо

$$b_s^{(j)} = \int_{t_s}^{t_{s+1}} (g(t_j) - g(\tau))^{-\beta} d\tau = \int_{g(t_s)}^{g(t_{s+1})} f'(x) (g(t_j) - x)^{-\beta} dx.$$

Розкладаючи $(g(t_j) - x)^{-\beta}$ з використанням узагальненої формули бінома Ньютона та розкладаючи $f'(x)$ у ряд Тейлора в точці $x = g(t_{s+1})$ отримуємо

$$\begin{aligned} b_s^{(j)} &= \sum_{n=0}^{\infty} \left((-1)^n \binom{-\beta}{n} g(t_j)^{-\beta-n} S_n \right), \\ S_n(t_s, t_{s+1}) &= \sum_{m=0}^{\infty} \left[B_m \frac{f^{(m+1)}(g(t_{s+1}))}{m!} \right], \quad B_m = \int_{g(t_s)}^{g(t_{s+1})} x^m (x - g(t_{s+1}))^m dx. \end{aligned} \quad (6.6)$$

Значення інтегралів B_m можна обчислити, використовуючи наступну рекурентну формулу:

$$\begin{aligned} B_0 &= \int_{g(t_s)}^{g(t_{s+1})} x^n dx = \frac{1}{n+1} (g(t_{s+1})^{n+1} - g(t_s)^{n+1}), \\ B_{i+1} &= -\frac{n+i+2}{g(t_{s+1})(i+1)} \left(B_i - \frac{g(t_s)^{n+1} (g(t_s) - g(t_{s+1}))^{i+1}}{g(t_{s+1})(i+1)} \right). \end{aligned}$$

Особливістю схеми (6.6) є те, що значення коефіцієнтів S_n , один раз розрахованих для фіксованих t_s і t_{s+1} , можливо зберігати та використовувати надалі при зміні t_j .

Обчислювальні експерименти, проведені у випадку $g(\tau) = \tau^2$, $f(\tau) = \tau^{1/2}$, $f'(\tau) = \frac{1}{2}\tau^{-1/2}$, $f^{(n+1)}(\tau) = \frac{1/2-n}{\tau} f^{(n)}(\tau)$ показали, що збіжність ряду (6.5) погіршується при $t_s \rightarrow 0$, тоді як збіжність ряду (6.6) погіршується при $t_{s+1} \rightarrow t_j$. Для забезпечення найвищої швидкодії при фіксованій точності пропонується наступний алгоритм автоматичного вибору рядів (6.5) або (6.6) для наближення інтегралів $b_s^{(j)}$ залежно від значень їх параметрів:

1) Нехай значення інтегралів $b_s^{(j)}$ розраховуються послідовно для $t_j, j = 0, 1, \dots$ і $t_s, s = 0, \dots, j-1$;

2) $a \leftarrow 2$;

3) При $t_s \leq t_a$ для чисельного інтегрування використовується схема (6.6);

4) Якщо $t_s = t_a$ виконуємо корекцію значення параметра a . Якщо кількість ітерацій, необхідних для наближення інтегралу із заданою точністю за допомогою схеми (6.5), перевищує відповідну кількість ітерацій для схеми (6.6), значення параметра слід збільшити: $a \leftarrow a + 1$. У протилежному випадку, якщо $a \neq 1$, значення має бути зменшене: $a \leftarrow a - 1$;

5) При $t_s > t_a$ для чисельного інтегрування використовується схема (6.5).

Ефективність вищеописаних алгоритмів перевірялася, розв'язуючи початково-крайову задачу для рівняння (6.1) з $L = 3, d = 1, \sigma = 1, \beta = 0,6; \tau = 0,1$.

Вихідні коди розробленого програмного забезпечення наведено у [265]. Загальний час (мс), витрачений під час розв'язання задачі на обчислення $b_s^{(j)}$ для $g(\tau) = \tau^2$ в залежності від кількості кроків за часом наведено на рис. 6.1. Алгоритм автоматичного вибору рядів (6.5) або (6.6) тут дозволяє прискорити обчислення, тоді як використання цих рядів по-окремо через особливості їх збіжності уповільнює процес розв'язання задачі.

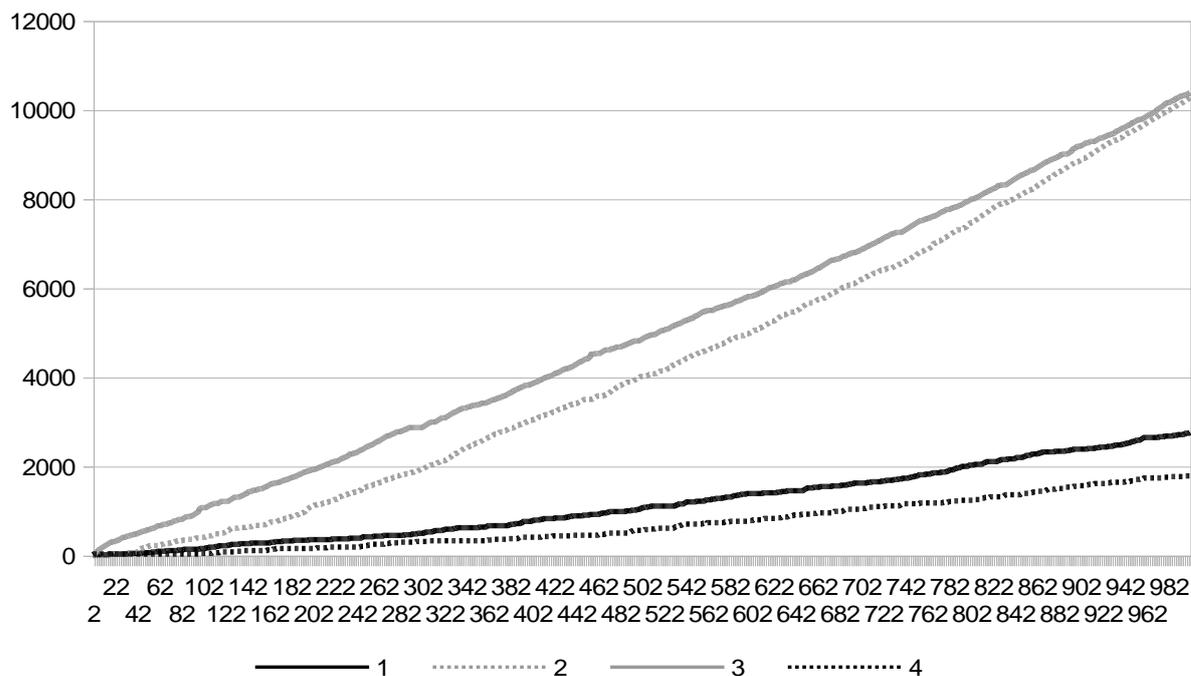


Рисунок 6.1 – Час, витрачений на обчислення значення $b_s^{(j)}$ залежно від кількості кроків за часом (1 – алгоритм рекурсивного підрозбиття, 2 – використання ряду (6.5), 3 – використання ряду (6.6), 4 – алгоритм автоматичного вибору рядів)

Проведені обчислювальні експерименти показали, що ефективність алгоритму автоматичного вибору рядів зменшується із збільшенням кількості кроків за часом. Водночас ефективність зростає зі зменшенням порядку β дробової похідної та зі збільшення довжини кроку за часом.

Відсоток часу, витраченого на обчислення $b_s^{(j)}$, у загальному часі розв'язання задачі, показано на рис. 6.2. Отримані дані свідчать про те, що вплив часу, витраченого на обчислення $b_s^{(j)}$, на загальний час зменшується зі збільшенням числа кроків за часом.

6.2. Паралельні алгоритми для графічних процесорів обчислення ψ -похідної Капуто за часовою змінною

Не зменшуючи загальність результатів, будемо розглядати наступне двовимірне рівняння аномальної конвективної дифузії [276]:

$$\sigma D_{t,g}^{(\beta)} C = D \left(\frac{\partial^2 C}{\partial x^2} + \frac{\partial^2 C}{\partial y^2} \right) - v_x \frac{\partial C}{\partial x} - v_y \frac{\partial C}{\partial y} + F, \quad (6.7)$$

де $C(x,t)$ – концентрація речовини, σ – пористість середовища, D – коефіцієнт дифузії, F – задана функція, $D_{t,g}^{(\beta)}$ – дробова ψ -похідна Капуто за часовою змінною t з функціональним параметром $g(t)$.

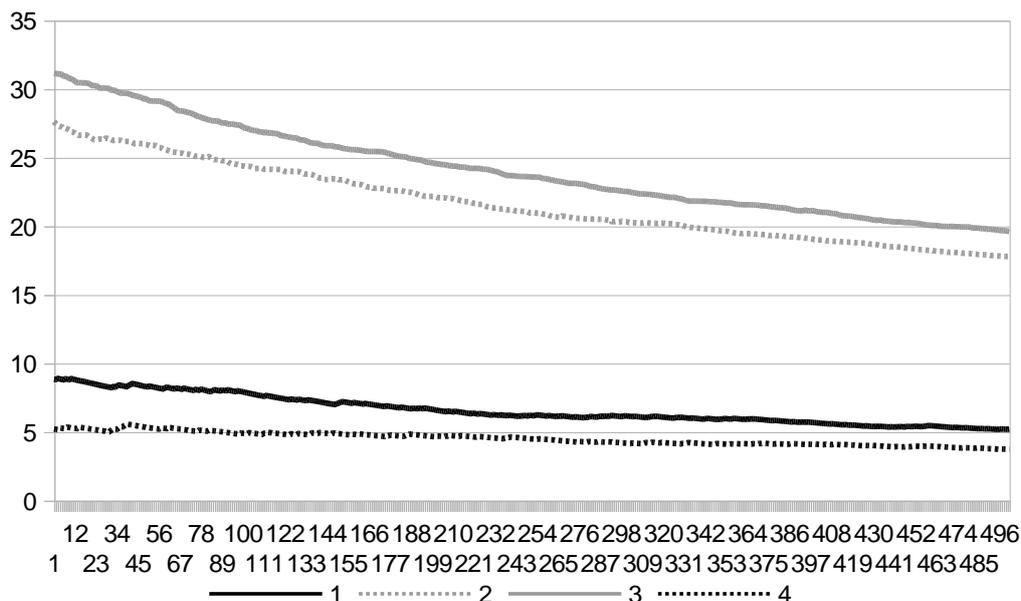


Рисунок 6.2 – Відсоток часу, витраченого на обчислення $b_s^{(j)}$, у загальному часі розв’язання задачі в залежності від кількості кроків за часом (1 – алгоритм рекурсивного підрозбиття, 2 – використання ряду (6.5), 3 – використання ряду (6.6), 4 – алгоритм автоматичного вибору ряду)

Процес фільтрації розглядається в області, описаній у [62]. Аналогічно [62, 311], компоненти вектора швидкості фільтрації v_x, v_y знаходяться на основі моделі стаціонарної фільтрації в області комплексного потенціалу течії. Виконавши у (6.7) перехід до змінних φ, ψ , які являють собою точки області комплексного потенціалу течії, отримуємо

$$\sigma D_{t,g}^{(\beta)} C(\varphi, \psi, t) = v^2(\varphi, \psi) \left[D \left(\frac{\partial^2 C}{\partial \varphi^2} + \frac{\partial^2 C}{\partial \psi^2} \right) - \frac{\partial C}{\partial \varphi} + F \right]. \quad (6.8)$$

До рівняння (6.8), слідуючи [276], ставимо початкові та граничні умови першого роду.

Вводячи до розгляду рівномірну скінченно-різницеву сітку

$$\omega_{ht} = \left\{ (\varphi_i, \psi_k, t_j) : \varphi_i = ih_1 \left(i = \overline{0, m+1} \right), \right. \\ \left. \psi_k = h_2(k - 0.5) \left(k = \overline{0, n+1} \right), t_j = jt \right\},$$

де h_1, h_2, τ – кроки за просторовими змінними та часом, відповідно, для системи (6.8) отримуємо у стандартних позначеннях наступний аналог локально-одновимірної схеми О. А. Самарського [313]:

$$\frac{\sigma}{2} \Delta_t^{(\beta)} \bar{C} = v^2 \left((D\bar{C}_{\varphi})_{\varphi} - \bar{C}_0 + 0.5F \right), \quad (6.9)$$

$$\frac{\sigma}{2} \Delta_t^{(\beta)} \hat{C} = v^2 \left((D\hat{C}_{\psi})_{\psi} - 0.5F \right), \quad (6.10)$$

$$\Delta_t^{(\beta)} \bar{C} = \frac{1}{\tau} \left(2b_j (C^{j+1/2} - C^j) + \sum_{s=0}^{j-1} \bar{b}_s^{(j)} (C^{s+1} - C^s) \right), \quad (6.11)$$

$$\Delta_t^{(\beta)} \hat{C} = \frac{2}{\tau} \sum_{s=0}^{j-1} \left[\bar{b}_s^{(j)} C^{s+1} + (\bar{q}_s^{(j)} - \bar{b}_s^{(j)}) C^{s+1/2} - \bar{q}_s^{(j)} C^s \right], \quad (6.12)$$

$$b_j = \frac{1}{\Gamma(1-\beta)} \int_{t_j}^{t_{j+1/2}} \frac{d\tau}{(g(t_{j+1/2}) - g(\tau))^\beta}, \bar{b}_s^{(j)} = \frac{1}{\Gamma(1-\beta)} \int_{t_s}^{t_{s+1}} \frac{d\tau}{(g(t_{j+1/2}) - g(\tau))^\beta}. \quad (6.13)$$

$$\bar{q}_s^{(j)} = \frac{1}{\Gamma(1-\beta)} \int_{t_s}^{t_{s+1/2}} \frac{d\tau}{(g(t_{j+1}) - g(\tau))^\beta}, \bar{b}_s^{(j)} = \frac{1}{\Gamma(1-\beta)} \int_{t_{s+1/2}}^{t_{s+1}} \frac{d\tau}{(g(t_{j+1}) - g(\tau))^\beta}.$$

Рівняння (6.9), (6.10) можуть бути легко перетворені в незалежні серії тридіагональних СЛАР, які ефективно розв'язується методом прогонки [313].

Найбільший вплив на швидкість проведення розрахунків тут має швидкодія обчислення наближених значень дробових похідних (6.11), (6.12), оскільки відповідний час лінійно збільшується зі збільшенням номеру кроку за часом. Обсяг пам'яті, необхідний для зберігання розв'язків задачі на попередніх кроках за часом, також лінійно збільшується.

Окрім проведення обчислень безпосередньо за (6.11), (6.12) (алгоритм 1), розглянемо два наступні алгоритми, що знижують обчислювальну складність.

Згідно з принципом фіксованої пам'яті, суми в (6.11), (6.12) обмежуються умовою $\bar{b}_s^{(j)} > \varepsilon_1$, де ε_1 – задана стала (алгоритм 2). Значення інтегралів (6.13) будемо обчислювати за допомогою алгоритму рекурсивного підрозбиття [43] (п. 6.1). У цьому випадку час, необхідний для виконання обчислень на одному кроці, разом із пам'яттю, необхідною для зберігання розв'язків, є близьким до постійного, починаючи з деякого кроку за часом.

Розглянемо також алгоритм 3, що базується на представленні інтегралів (6.13) у вигляді ряду (6.6) [43] (п. 6.1). Обчислення сум у (6.11) і, аналогічно у (6.12), використовуючи (6.6) будемо рекурентно проводити наступним чином:

$$\sum_{s=0}^{j-1} \bar{b}_s^{(j)} (C^{s+1} - C^s) \approx \sum_{n=0}^K \left((-1)^n \binom{-\alpha}{n} g(t_{j-1})^{-\alpha-n} S_{n,j-1} \right),$$

$$S_{n,j} = S_{n,j-1} + (C^j - C^{j-1}) S_n(t_{j-1}, t_j), S_{n,1} = 0, \quad (6.14)$$

де K – задане обмеження на кількість членів у ряді.

У зв'язку з поганою збіжністю рядів (6.6) при $t_{s+1} = t_j$, будемо обчислювати b_j та $\bar{b}_j^{(j)}$ у цьому випадку алгоритмом рекурсивного підрозбиття [43] (п. 6.1).

Обчислення правих частин СЛАР, отримуваних з (6.9), (6.10), разом із їх розв'язанням може виконуватися паралельно з застосуванням багатопотокового та GPU-розпаралелення за даними.

Швидкодія алгоритмів 1–3 перевірялася, розв'язуючи модельну задачу для рівняння (6.8) при $F(\varphi, \psi, t) = \sigma \frac{\Gamma(1+2/\gamma)}{v^2 \Gamma(1-\alpha+2/\gamma)} t^{2-\alpha\gamma} - 2(D(\varphi^2 + \psi^2) - \varphi\psi^2)$ і $g(t) = t^\gamma$. Тоді розв'язком початково-крайової задачі з відповідними умовами першого роду є функція $C(\varphi, \psi, t) = \varphi^2 \psi^2 + t^2$.

Дискретизована система (6.9), (6.10) розв'язувалась на 1000 кроках за часом довжиною $\tau = 0,001$ на сітці розміру 50x50 комірок при $\gamma = 0,5; 1; 2$.

Використовувалась багатопотокова OpenMP реалізація алгоритмів та OpenCL-реалізація для виконання на GPU. Тестування проводилось на CPU AMD E2-9000 та GPU AMD Radeon R2 240. Вихідні коди розробленого програмного забезпечення наведено у [265].

Значення параметру ε_1 алгоритму 2 приймалось рівним $\varepsilon_1 = 0,1; 0,01; 0,001$, а кількість членів K рядів у алгоритмі 3 – рівною $K = 100, 75, 50, 25$. Розподіл похибок розв'язання (середньо-квадратичного відхилення на останньому кроці за часом) та загальний час, витрачений на проведення обчислень, при виконанні 2 потоків на CPU для $g(t) = t^{1/2}$ наведено на рис. 6.3.

Хоча проведення обчислень безпосередньо за (6.11), (6.12) дозволило отримати найбільш точні розв'язки, час, що при цьому витрачається, був значно (у 3–22 рази) більшим, ніж у випадку використання інших алгоритмів. Як для алгоритму 2 методу фіксованої пам'яті, так і для алгоритму 3 розкладення у ряди, зменшення часу, витраченого на отримання розв'язку, супроводжувалось зменшенням точності. При $\varepsilon_1 \geq 0,01$ і $K \geq 50$ більш точні розв'язки були отримані алгоритмом 3. Загалом, порівнюючи з алгоритмом 1, алгоритмом 2 було отримане прискорення в 4–5 разів із втратою точності у 7–11%, тоді як для обчислювальної схеми розкладення у ряди було досягнуто 7–9-кратного прискорення з втратою точності у 8–30%. Ці висновки були справедливими для всіх перевірених значень γ .

Час обчислень лінійно збільшувався зі збільшенням номеру кроку за часом для алгоритму 1, був постійним для алгоритму 3 і ставав близьким до постійного після деякої кількості кроків за часом для алгоритму 2. Коли для обчислень використовувався GPU, пришвидшення порівняння з виконанням на CPU у 2 потоках наведено на рис. 6.4. Прискорення ставало близьким до постійного через певну кількість кроків за часом для всіх алгоритмів. Чим повільніше алгоритм, тим вищим було прискорення, і тим більше кроків за часом було необхідно для досягнення його більших значень (рис. 6.4). Загалом, GPU-реалізація алгоритмів дозволила отримувати розв'язки задач в ~ 5 разів швидше для неоптимізованого алгоритму 1 та у ~ 2 швидше для алгоритмів 2 та 3. Менша швидкодія алгоритму 3

може бути пояснена нижчою продуктивністю використаного графічного процесора в порівнянні з CPU при виконанні математичних обчислення, які є значно складнішими, ніж в алгоритмі 1.

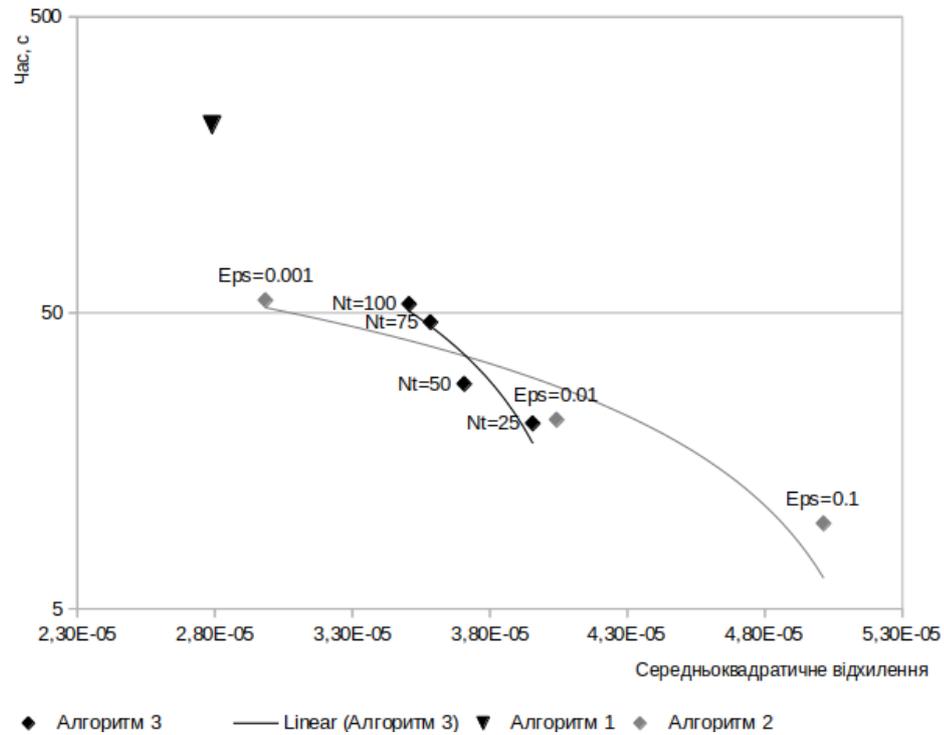


Рисунок 6.3 – Розподіл похибки розв'язання та загального часу, витраченого на проведення обчислень, для $g(t) = t^{1/2}$

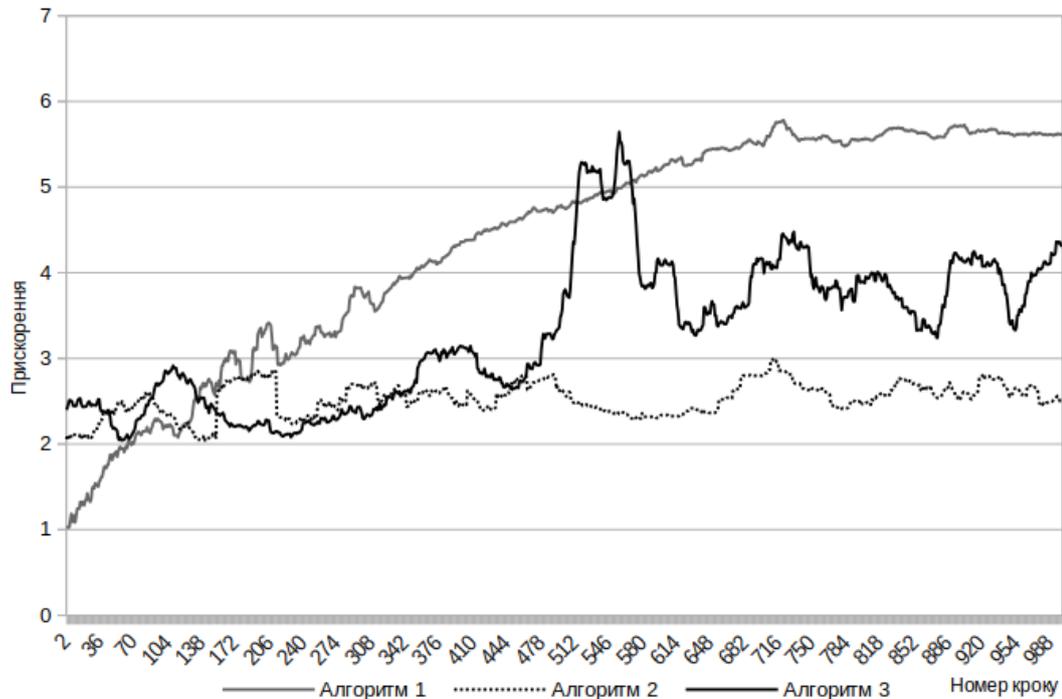


Рисунок 6.4 – Прискорення GPU алгоритмів

Таким чином, алгоритм 3 має найнижчий порядок складності серед розглянутих обчислювальних схем, проте прискорення його GPU реалізації у формі (6.14) є нижчим, ніж у випадку інших розглянутих алгоритмів. Надалі дослідимо оптимізовані схеми GPU реалізації алгоритму 3, що базується на розкладенні ядра інтегрального оператора у ряди.

Будемо використовувати наступну схему виконання алгоритму на графічному процесорі [42]. Значення $S_n(t_{l-1}, t_l)$ будемо розраховувати на CPU на кроці $l-1$ паралельно з проведенням обчислень на GPU та завантажуватимемо на кроці l у пам'ять графічного процесора разом із розв'язком $C^{(l-1)}$. Надалі, кожен потік GPU обчислює значення $F_{ijk}^{(l)}$ для конкретного вузла сітки та після завершення обчислень копіює їх у пам'ять CPU.

Для використання локальної пам'яті GPU, сформуємо групи потоків контрольованого розміру χ з обчисленнями, організованими в $\lceil l / \chi \rceil$ підкроки. На підкроці s , значення $S_n(t_{l-1}, t_l), s \lceil l / \chi \rceil \leq n < (s+1) \lceil l / \chi \rceil$ завантажуються в локальну пам'ять та відповідна частина $S_{n,l}$ оновлюється кожним потоком. Значення $(-1)^n \binom{-\alpha}{n} g(t_{l-2})^{-\alpha-n}$ заздалегідь обчислюється потоками паралельно і завантажуються в локальну пам'ять після чого обчислюються фінальні значення сум у (6.14).

З метою використання векторних типів даних, представимо суми у (6.14) у вигляді операцій з векторами, що містять 16 членів, які відповідають типам даних float16 / double16 у OpenCL, отримуючи [55]

$$\sum_{s=0}^{l-2} b_s^{(l)} \frac{C_{ijk}^{(s+1)} - C_{ijk}^{(s)}}{\tau} \approx \frac{1}{\tau} \sum_{n=0}^{\lceil K/16 \rceil} \left(\vec{v}_n^{(1)} (S_{n \cdot 16, l-1}, \dots, S_{(n+1) \cdot 16, l-1}) \right),$$

$$\vec{v}_n^{(1)} = \left((-1)^{n \cdot 16} \binom{-\beta}{n \cdot 16} g(t_l)^{-\beta-n \cdot 16}, \dots, (-1)^{(n+1) \cdot 16} \binom{-\beta}{(n+1) \cdot 16} g(t_l)^{-\beta-(n+1) \cdot 16} \right),$$

$$\begin{aligned} (S_{n \cdot 16, l}, \dots, S_{(n+1) \cdot 16, l}) &= (S_{n \cdot 16, l-1}, \dots, S_{(n+1) \cdot 16, l-1}) + \\ &+ (C^{(l)} - C^{(l-1)}) (S_{n \cdot 16}(t_{l-1}, t_l), \dots, S_{(n+1) \cdot 16}(t_{l-1}, t_l)). \end{aligned} \quad (6.15)$$

Ефективність схеми (6.15) дослідимо [55] отримуючи залежності прискорення від розміру сітки, параметра K та розміру груп потоків. У якості тестової задачі розглянемо у позначеннях, використаних для (6.7), наступне тривимірне дробово-диференціальне рівняння, що описує нелокальний за часом дифузійний процес [276]:

$$\sigma D_{t,g}^{(\beta)} C(x, y, z, t) = D \left(\frac{\partial^2 C(x, y, z, t)}{\partial x^2} + \frac{\partial^2 C(x, y, z, t)}{\partial y^2} + \frac{\partial^2 C(x, y, z, t)}{\partial z^2} \right) + F(x, y, z, t), \quad 0 \leq x \leq 1, 0 \leq y \leq 1, 0 \leq z \leq 1, t > 0, 0 < \beta \leq 1. \quad (6.16)$$

Для рівняння (6.16), слідуючи [276], поставимо початкові та граничні умови першого роду.

Будемо розв'язувати початково-крайову задачу щодо рівняння (6.16), використовуючи локально одновимірну скінченно-різницева схему [313] на рівномірній сітці

$$\omega = \left\{ (x_i, y_j, z_k, t_l) : x_i = ih_1 \left(i = \overline{0, n_1 + 1} \right), y_j = jh_2 \left(j = \overline{0, n_2 + 1} \right), z_k = kh_3 \left(k = \overline{0, n_3 + 1} \right), t_l = l\tau \right\}$$

де h_1, h_2, h_3, τ – це кроки щодо просторових та часової змінних.

Скінченно-різницева схема, подібно до [67], має вигляд [55]

$$C_{ijk}^{(l+1/3)} - \frac{\tau D}{k_1 h_1^2} (C_{i-1,j,k}^{(l+1/3)} - 2C_{ijk}^{(l+1/3)} + C_{i+1,j,k}^{(l+1/3)}) = C_{ijk}^{(l)} + \frac{\tau}{3k_1} F_{ijk}^{(l)}, \quad (6.17)$$

$$C_{ijk}^{(l+2/3)} - \frac{\tau D}{k_1 h_2^2} (C_{i,j-1,k}^{(l+2/3)} - 2C_{ijk}^{(l+2/3)} + C_{i,j+1,k}^{(l+2/3)}) = C_{ijk}^{(l+1/3)} + \frac{\tau}{3k_1} F_{ijk}^{(l)}, \quad (6.18)$$

$$C_{ijk}^{(l+1)} - \frac{\tau D}{k_1 h_3^2} (C_{i,j,k-1}^{(l+1)} - 2C_{ijk}^{(l+1)} + C_{i,j,k+1}^{(l+1)}) = C_{ijk}^{(l+2/3)} + \frac{\tau}{3k_1} F_{ijk}^{(l)}, \quad k_1 = \frac{b_l^{(l+1)}}{\Gamma(1-\beta)}, \quad (6.19)$$

$$F_{ijk}^{(l)} = F_{ijk} + \frac{1}{\Gamma(1-\beta)} \sum_{s=0}^{l-2} b_s^{(l)} \frac{C_{ijk}^{(s+1)} - C_{ijk}^{(s)}}{\tau}, \quad b_s^{(l)} = \int_{t_s}^{t_{s+1}} (g(t_s) - g(\tau))^{-\beta} d\tau. \quad (6.20)$$

Надалі, дискретизована форма граничних умов першого роду додається до систем рівнянь (6.17)–(6.19), а отримувані в результаті тридіагональні СЛАР розв'язуються методом прогонки [313]. Будемо розглядати багатопотокову OpenMP-реалізацію розв'язання систем (6.17)–(6.19) за якої кожен потік розв'язує блок СЛАР.

Розрахунок нелокальної частини $\frac{1}{\Gamma(1-\beta)} \sum_{s=0}^{l-2} b_s^{(l)} \frac{C_{ijk}^{(s+1)} - C_{ijk}^{(s)}}{\tau}$ ψ -похідної Капуто

виконується паралельно для кожного вузла сітки за допомогою GPU.

Точність розрахунків будемо визначати для випадку модельної задачі для рівняння (6.16) з $g(t) = t^\gamma$. Вихідні коди розробленого програмного забезпечення наведено у [265].

При $F(x, y, z, t) = \frac{\Gamma(1+2/\gamma)}{\Gamma(1-\beta+2/\gamma)} t^{2-\beta\gamma} - 2D(x^2y^2 + y^2z^2 + x^2z^2)$, його розв'язком є

$C(x, y, z, t) = C_0(x, y, z, t) = x^2y^2z^2 + t^2$ для випадку $\sigma = 1$ та початкових і граничних умов у вигляді $C(x, y, z, 0) = C_0(x, y, z, t)$, $C(0, y, z, t) = C_0(0, y, z, t)$, $C(1, y, z, t) = C_0(1, y, z, t)$ і так само для інших просторових змінних.

Система (6.17)–(6.19) розв'язувалась для 40 кроків за часом довжиною $\tau = 0,0025$ при $\gamma = 0,9$; $\beta = 0,8$ на сітці розміру $N \times N \times N$, $N = 20, \dots, 70$ варіюючи K у діапазоні $K = 16, \dots, 64$ на одному вузлі кластеру СКІТ-4 Інституту кібернетики імені В. М. Глушкова НАН України. Використовуваний вузол містить два процесори Intel (R) Xeon (R) Bronze 3104 із загальною кількістю ядер, що дорівнює 12, та графічний процесор Nvidia RTX 2080 Ti.

Середній час, витрачений OpenMP та OpenCL-реалізаціями для отримання розв'язку на одному кроці за часом, наведено в табл. 6.1 разом із прискоренням GPU алгоритму. Для GPU реалізації розмір групи потоків дорівнював кількості членів K у ряді (6.14). Як і очікувалося, час тут ставав близьким до постійного зі збільшенням числа кроків і залежав від параметра розміру сітки N як $O(N^3)$ як для CPU, так і для GPU-реалізації. У порівнянні з проведенням повного підсумовування згідно з (6.20), середня квадратична помилка розв'язання задачі погіршувалась не більше ніж на 6%. Зі збільшенням K похибка квадратично знижувалась.

Прискорення від використання графічного процесора зростає зі збільшенням K . При збільшенні N , воно збільшується до $N = 40$ залишаючись близьким до постійного для більших значень. Таку поведінку можна пояснити

неповним використанням ресурсів GPU при $N < 40$. У проведених експериментах використання GPU дозволило зменшити загальний час розв’язання до 4 разів.

Таблиця 6.1 – Час виконання алгоритмів на одному кроці

	K	$N=20$	30	40	50	60	70
Час для OpenMP реалізації, мс	16	5,13	18,21	43,59	85,38	163,08	233,85
	32	7,18	26,15	60,26	118,97	225,13	328,46
	64	10,77	37,18	91,03	180,77	330,51	502,82
Час для GPU реалізації, мс	16	4,87	11,79	24,87	46,41	93,33	117,18
	32	5,13	10,51	23,59	46,15	95,38	116,41
	64	4,87	11,54	23,59	48,97	90,77	122,05
Прискорення при використанні GPU реалізації, %	16	5,26%	54,35%	75,26%	83,98%	74,73%	99,56%
	32	40,00%	148,78%	155,43%	157,78%	136,02%	182,16%
	64	121,05%	222,22%	285,87%	269,11%	264,12%	211,97%

Середній час, витрачений OpenCL ядром на виконання обчислень за (6.14) та (6.15) за різних значень K та розміру груп потоків, виміряний за допомогою внутрішніх інструментів профілювання OpenCL, наведено у табл. 6.2 разом із відповідним часом, витраченим OpenMP-реалізацією на CPU.

Виконуючи обчислення на GPU згідно з (6.14) при розміру груп потоків, що дорівнює K , прискорення у порівнянні з виконанням на CPU збільшувалось логарифмічно (рис. 6.5) зі збільшенням N і досягало максимуму в 130 разів для $N = 70, K = 32$. Найвища швидкість для тестованих значень N досягалась при розмірі груп потоків, що дорівнює 32. Для більших значень прискорення знижується до 52 разів для 64 потоків у групі у порівнянні з 62-кратним прискоренням для випадку 16 потоків у групі.

Використання векторизованих операцій на GPU дає додатковий приріст продуктивності у порівнянні з GPU алгоритмом, що виконує обчислення згідно з (6.14) (рис. 6.6). Обчислення згідно з (6.15) були на від 8% до 25% швидшими незалежно від N для $K = 16, 32, 48$. Для вищих значень K , прискорення від використання векторизованої схеми (6.15) лінійно зростає зі збільшенням N досягаючи 3-разового рівня при $K = 64, N = 70$. У порівнянні з OpenMP реалізацією, прискорення у цьому випадку склало 153 разів.

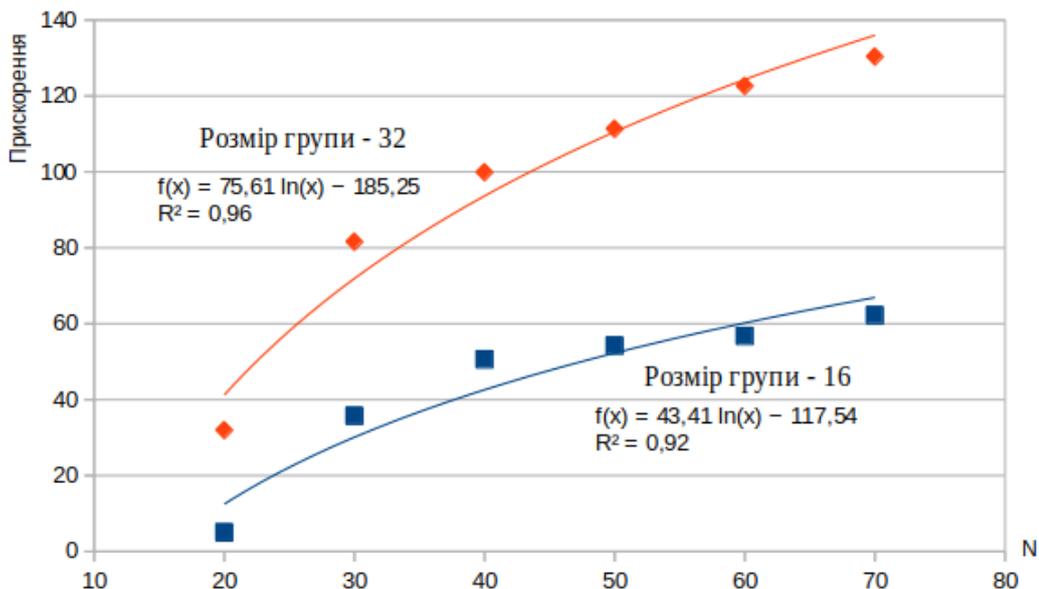


Рисунок 6.5 – Прискорення GPU-алгоритму

Таблиця 6.2 – Час, витрачений на обчислення згідно з (6.14) та (6.15), мс

K	Розмір груп потоків	N= 20	30	40	50	60	70
OpenMP							
16		0,36	6,89	20,23	40,31	70,13	119,12
32		2,41	14,84	36,90	73,90	132,18	213,73
64		6,00	25,87	67,67	135,69	237,56	388,09
GPU згідно з (6.14)							
16	16	0,07	0,19	0,40	0,74	1,24	1,91
32	32	0,08	0,18	0,37	0,66	1,08	1,64
32	16	0,11	0,31	0,66	1,24	2,06	3,20
64	64	0,14	0,34	1,33	2,64	4,66	7,42
64	16	0,19	0,56	1,19	2,23	3,70	5,77
Векторизована схема (6.15)							
16	16	0,06	0,18	0,37	0,68	1,13	1,76
32	32	0,06	0,16	0,34	0,60	0,99	1,51
32	16	0,13	0,37	0,79	1,47	2,46	3,83
64	64	0,10	0,26	0,72	1,19	1,74	2,53
64	16	0,33	1,03	2,22	4,20	6,99	10,91

Оскільки прискорення роботи GPU алгоритмів знижується при K та розмірі груп потоків більше за 32, був також протестований варіант, коли розмір груп потоків залишався рівним 16 (рис. 6.6) при зміні K . Тут продуктивність

векторизованого алгоритму значно погіршується, тоді як для базового GPU-алгоритму незначне зменшення часу виконання спостерігалось лише для великих сіток ($N > 50$).

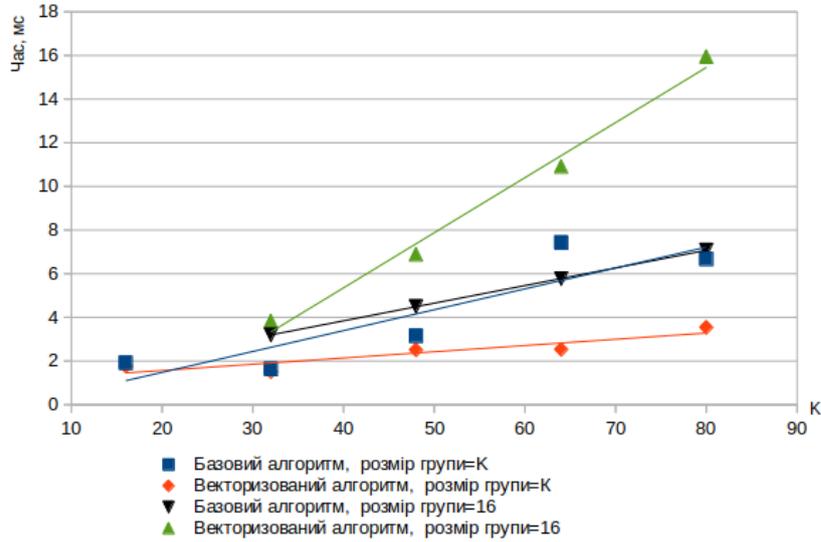


Рисунок 6.6 – Час виконання GPU-алгоритмів

Зважаючи на ефективність використання векторизації обчислень на GPU при розв'язанні розглядуваної задачі, дослідимо вплив інших низькорівневих факторів на швидкодію GPU алгоритмів [41]. Зокрема, розглянемо швидкодію при зміні розміру та точності векторних типів, а також при використанні тензорних ядер GPU, порівняємо ефективність CUDA та OpenCL-реалізацій.

Розбиваючи суми у (6.14) на операції з m -компонентними векторами, отримуємо

$$\sum_{s=0}^{l-2} b_s^{(l)} \frac{C_{ijk}^{(s+1)} - C_{ijk}^{(s)}}{\tau} \approx \frac{1}{\tau} \sum_{n=0}^{(K/m)} \left(\vec{v}_{n,m}^{(1)} \left(\bar{S}_{n \cdot m, l-1}, \dots, \bar{S}_{(n+1) \cdot m, l-1} \right) \right),$$

$$\vec{v}_{n,m}^{(1)} = \left((-1)^{n \cdot m} \binom{-\beta}{n \cdot m} g(t_l)^{-\beta - n \cdot m}, \dots, (-1)^{(n+1) \cdot m} \binom{-\beta}{(n+1) \cdot m} g(t_l)^{-\beta - (n+1) \cdot m} \right),$$

$$\begin{aligned} \left(\bar{S}_{n \cdot m, l}, \dots, \bar{S}_{(n+1) \cdot m, l} \right) &= \left(\bar{S}_{n \cdot m, l-1}, \dots, \bar{S}_{(n+1) \cdot m, l-1} \right) + \\ &+ \left(C^{(l)} - C^{(l-1)} \right) \left(S_{n \cdot m}(t_{l-1}, t_l), \dots, S_{(n+1) \cdot m}(t_{l-1}, t_l) \right). \end{aligned} \quad (6.21)$$

Будемо розглядати реалізацію схеми (6.21) з типами даних із рухомою комою різного розміру (16, 32 і 64 біти) та з векторними типами даних різного розміру (2, 4, 8, 16 елементів).

Оскільки найновіші графічні процесори NVidia апаратно реалізують операції множення матриць на матриці у так званих тензорних ядрах, перетворимо формулу (6.21) з метою можливості їх використання. Тензорна операція $T_{ma}(A, B, C)$ виконує множення двох матриць A , B розміру 16×16 , представлених 16-бітовим типом даних “половинної” точності, та додає матрицю C одинарної точності (32-бітовий тип даних) розміру 16×16 до результату множення. Результат операції представляється типом даних з одинарною точністю. Розглядаючи вектор сум $S^{(16)}$ для 16 комірок сітки $C_{(1)}, \dots, C_{(16)}$, підсумовування згідно з (6.21) для $K = 16$ можна виконати наступним чином:

$$S^{(16)} \approx \frac{1}{\tau} \left(T_{ma} \left(\tilde{S}, V, A_0 \right) \right)_1, \quad V = \left(\bar{v}_{n,16}^{(1)T}, v_0^T, \dots, v_0^T \right),$$

$$\tilde{S} = \begin{pmatrix} \hat{S}_{1,l-1,(0)} & \dots & \hat{S}_{16,l-1,(0)} \\ \dots & \dots & \dots \\ \hat{S}_{1,l-1,(16)} & \dots & \hat{S}_{16,l-1,(16)} \end{pmatrix} \quad (6.22)$$

де A_0 – нульова матриця розміру 16×16 , v_0 – 16-компонентний нульовий вектор, $\hat{S}_{n,l,(i)}$ – значення коефіцієнта $\bar{S}_{n,l}$ для комірки сітки i , $(\cdot)_1$ позначає перший стовпець матриці.

Реалізація формули (6.22) використовує лише одне множення матриці на вектор з 16 виконуваних операцією T_{ma} .

Швидкодія схем (6.21) та (6.22) тестувалась розв’язуючи вищеописану модельну задачу моделювання аномального дифузійного процесу згідно з рівнянням (6.16). Задача розв’язувалась на 200 кроках за часом довжиною $\tau = 0,0025$ при $\gamma = 0,8$; $\beta = 0,8$, $K = 16, \dots, 64$ на сітці розміру $N \times N \times N$, $N = 40, \dots, 80$ на одному вузлі кластеру СКІТ-4 Інституту кібернетики імені В. М. Глушкова НАН України. Вихідні коди розробленого програмного забезпечення наведено у [265].

Доступний набір типів даних для реалізації з використанням CUDA був *double*, *float*, *half*, *double*, *double*[2,4], *float*[2,4] і *half*2. У випадку з OpenCL розглядалися реалізовані у фреймворку CUDA10 типи даних *double*, *float*, *double*[2,4,8,16] та *float*[2,4,8,16]. Час, витрачений на обчислення згідно з (6.20), (6.21) та (6.22) на 200 ітераціях для $N = 40$, $K = 16$ та всіх розглянутих комбінацій мов та типів даних, наведений у табл. 6.3. У всіх експериментах розмір груп потоків χ дорівнював K .

Час, витрачений на одну ітерацію, є близький до сталого для схеми (6.21) та має тенденцію до лінійного збільшення при збільшенні номеру кроку за часом для підсумовування згідно з (6.20). Це лінійне збільшення є немонотонним (рис. 6.7) через неповне використання ресурсів графічного процесора у випадку $t\%K \neq 0$, де t – номер кроку за часом.

Виконуючи обчислення за (6.20) при реалізації з використанням CUDA, у порівнянні з використанням типу даних з подвійною точністю, алгоритми, реалізовані з використанням типів даних з меншою точністю, виконували обчислення в ~ 2 рази швидше для 32-бітового типу даних та у ~ 3 рази швидше для 16-бітового типу.

Таблиця 6.3 – Час, с, витрачений на обчислення згідно з (6.20), (6.21) та (6.22) для $N = 40$, $K = 16$

Тип даних	CUDA, схема (6.20)	CUDA, схема (6.21)	OpenCL, схема (6.20)	OpenCL, схема (6.21)
<i>Double</i>	344,18	78,37	286,86	81,14
<i>Float</i>	204,03	74,86	88,27	83,97
<i>Half</i>	105,77	77,32		
<i>Double2</i>	350,24	81,41	287,11	81,90
<i>float2</i>	191,38	73,02	91,08	84,97
<i>Half2</i>	98,14	74,70		
<i>Double4</i>	345,18	79,52	287,57	81,56
<i>float4</i>	182,00	70,19	90,22	84,87
<i>Double8</i>			292,92	80,81
<i>Float8</i>			90,50	85,18
<i>Double16</i>			292,41	80,99
<i>float16</i>			90,76	84,38
Використання тензорних ядер – схема (6.22)		69,49		

Використання алгоритму (6.21) до ~ 5 разів зменшувало час проведення обчислень у порівнянні з підсумовуванням згідно з (6.20). Тут використання типу даних одинарної точності скоротило час лише на $<12\%$ коли алгоритм був реалізований з використанням CUDA. Це можна пояснити великим обсягом операцій перетворення типів даних у реалізації (6.21) через збереження розв'язків C і коефіцієнтів s_n використовуючи 64-бітовий тип даних з подвійною точністю. Використання 16-бітового типу даних половинної точності у CUDA-реалізації уповільнило обчислення у порівнянні з використанням 32-бітового типу даних. Однак програмна реалізація з 16-бітовими типами даних, яка використовує тензорні ядра, була найшвидшою з усіх розглянутих варіантів. Використання типу даних одинарної точності у OpenCL-реалізації уповільнило обчислення у порівнянні з використанням типу даних подвійної точності.

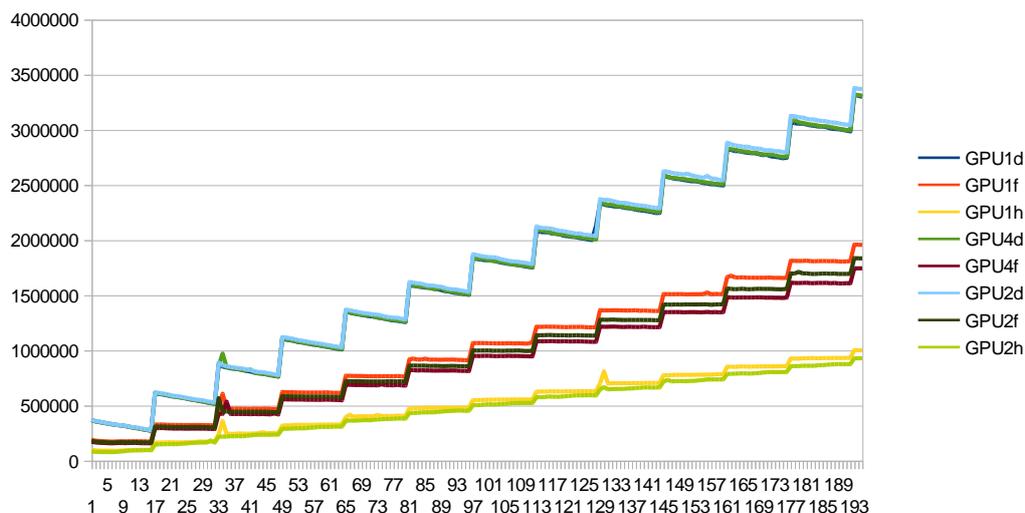


Рисунок 6.7 – Час виконання ядра CUDA, нс, для випадку підсумовування згідно з (6.20), $N = 40$, $K = 16$ (GPU[1,2][d,f,h] — використання скалярного типу даних (1) чи векторного типу з двома елементами (2) подвійної (d), одинарної (f) чи половинної (h) точності)

При $K = 16$, векторизація прискорювала обчислення лише у випадку типу даних одинарної точності у CUDA-реалізації.

Щоб пояснити спостережувану поведінку, були порівняні вихідні PTX-коди, створені компіляторами CUDA та OpenCL із включеним прапорцем FMA (використання операції множення з додаванням). З проведеного аналізу можна зробити наступні висновки:

- Обидва компілятори виконують автоматичне розгортання 16 ітерацій циклів;
- Обидва компілятори не генерують векторні арифметичні інструкції при використанні векторних типів даних;
- Обидва компілятори генерують векторизовані 128-бітові інструкції завантаження/зберігання `[ld, st].[shared, global].v2.f64` та `[ld, st].[shared, global].v4.f32` і саме їх використання прискорює обчислення при використанні векторних типів даних;
- Коли обидва компілятори виконують розгортання циклу, пам'ять адресується через фіксовані зміщення від бази як для локальної, так і для глобальної пам'яті (наприклад, `ld.global.f64 %fd77, [%rd23+40]`). Для векторизованих типів даних при кожному доступі до пам'яті для відповідного вектора обчислюються зміщення у глобальній пам'яті (наприклад, `mul.wide.s32 %rd28,%r67,128; add.s64 %rd29,%rd3,%rd28; ld.global.v2.f64 {%fd172, %fd173}, [%rd29 + 112]; ld.global.v2.f64 {%fd176,%fd177}, [%rd29 + 96]; ...`). Для обох алгоритмів (6.20) та (6.21) такі додаткові операції уповільнювали обчислення. З іншої сторони, збільшення K призводить до зменшення кількості допоміжних операцій, що дозволяє отримати до 15% прискорення для використання типу даних `double16` в OpenCL;
- Компілятор CUDA автоматично векторизує читання 32-розрядних даних з рухомою комою, що зберігаються у локальній пам'яті, використовуючи 64-розрядну інструкцію `ld.shared.v2.f32`, роблячи обчислення згідно з (6.21) швидшим, ніж у випадку використання типу даних з подвійною точністю. Відсутність такої автоматичної векторизації в OpenCL призводить до того, що OpenCL реалізація з використанням 32-бітового типу даних працює повільніше, ніж у випадку даних з подвійною точністю. Те саме стосується використання 16-бітового типу даних у

CUDA, при використанні якого генерується повільніший код, ніж у випадку типу даних з одинарною точністю;

- Виконуючи обчислення згідно з (6.20), компілятор CUDA реалізує операцію ділення у вигляді інструкції `div.rn`, тоді як компілятор OpenCL використовує наближену інструкцію `div.full`, що робить OpenCL код швидшим.

Для визначення тенденцій зміни ефективності векторизації при змінному $K \geq 16$, була проведена серія обчислювальних експериментів при $N = 40$ і $K = 16, \dots, 64$ для випадку схеми (6.21). Тут автоматичне розгортання циклу призводить до зменшення кількості допоміжних операцій зі збільшенням K . Час обчислення також зменшується до 15% при використанні типу даних `double16` у OpenCL, до 5.3% при використанні типу даних `double8` у OpenCL, до 2,6% при використанні типу даних `float16` у OpenCL та до 19% при використанні типу даних `float4` у CUDA. Використання типів даних `double[2,4]` та `float[2,8]` у OpenCL уповільнювало обчислення.

Таким чином, хоча OpenCL підтримує більші за розміром векторні типи даних, ніж CUDA (16 елементів у порівнянні з 4 елементами), неефективна компіляція OpenCL коду компілятором NVidia призводить до ситуації, коли OpenCL-реалізація схеми (6.21) дозволяє прискорювати обчислення лише для великих K і найбільших векторних типів даних.

Вплив часу обчислення значень дробових похідних на загальний час, витрачений на проведення моделювання, разом із прискореннями різних GPU-реалізацій, наведено в табл. 6.4 для підсумовування згідно з (6.20) та в табл. 6.5 для схеми (6.21) для випадків $N = 40, \dots, 80$.

У схемі (6.20), час, витрачений на обчислення значень дробових похідних (час підсумовування), складає до 36% загального часу обчислення на CPU. Це значення знижується до $\sim 4\%$ у випадку використання GPU, коли ця операція стає в 9–30 разів швидшою, а загальний час обчислення стає на $\sim 50\%$ менше. Як видно з табл. 6.4, ефективність GPU алгоритмів зростає із збільшенням N тоді як ефективність векторизації залишається на тому ж рівні.

Таблиця 6.4 – Характеристики алгоритмів у випадку підсумовування згідно з (6.20), $K = 16$, 200 ітерацій

	N	40	60	80
CPU	Відсоток часу підсумовування у загальному часі	31,93%	36,64%	36,41%
Прискорення CUDA float відносно CPU	Загальний час	40,74%	43,58%	51,15%
	Час підсумовування	939,46%	1374,70%	1568,14%
CUDA float	Відсоток часу підсумовування у загальному часі	4,32%	3,57%	3,30%
	Відсоток часу допоміжних CPU-GPU операцій у загальному часі	28,62%	22,25%	22,22%
Прискорення CUDA float4 відносно CUDA float	Час підсумовування	7,52%	5,40%	8,80%
Прискорення OpenCL float відносно CUDA float	Час підсумовування	68,19%	104,21%	95,05%

Для схеми (6.21) при $K = 16$, час обчислення значень дробових похідних становить до 40% загального часу обчислення під час виконання на CPU з незначним зменшенням цього відсотка зі збільшенням N . Ефективність GPU тут близька до випадку підсумовування згідно з (6.20): 15–27-кратне прискорення обчислень при використанні схеми (6.21), зменшення загального часу обчислення на 51–63% у порівнянні з CPU реалізацією, ~3% часу, витраченого алгоритмами GPU на обчислення значень дробових похідних. У порівнянні з повільнішою схемою (6.20), час, витрачений на допоміжні операції копіювання даних між пам'яттями CPU та GPU під час запуску ядер графічного процесора, тут більший: ~60% у порівнянні з ~25%. Ефективність векторизації не залежить від N . При використанні тензорних ядер найбільший приріст швидкодії становив 12% у порівнянні з використанням 16-бітового типу даних та 8% у порівнянні з використанням типу даних одинарної точності. Ефективність використання тензорних операцій у порівнянні з найшвидшим випадком векторизованої реалізації з типом даних float4 становить 2%.

Зміни максимальної квадратичної абсолютної помилки на ітераціях при використанні підсумовування згідно з (6.20) та схеми (6.21) для різних типів даних

наведені на рис. 6.8. Зниження точності типу даних має незначний вплив на якість розв'язків для підсумовування згідно з (6.20) внаслідок монотонного збільшення $b_s^{(i)}$ при $i \rightarrow s$, що мінімізує похибки підсумовування. Похибка для алгоритму (6.20) зменшується зі збільшенням номеру кроку за часом.

Таблиця 6.5 – Характеристики алгоритмів для схеми (6.21), $K = 16$, усереднені значення за 200 ітерацій

	N	40	60	80
CPU	Відсоток часу підсумовування у загальному часі	39,85%	38,04%	36,85%
Прискорення CUDA float відносно CPU	Загальний час	61,19%	53,05%	55,29%
	Час підсумовування	1717,88%	2536,28%	2735,08%
CUDA float	Відсоток часу підсумовування у загальному часі	3,53%	2,21%	2,02%
	Відсоток часу допоміжних CPU-GPU операцій у загальному часі	67,30%	61,63%	59,96%
Прискорення CUDA float4 відносно CUDA float	Час виконання ядра	6,43%	6,40%	6,60%
Прискорення OpenCL float відносно CUDA float	Час виконання ядра	-10,84%	-9,83%	-9,72%
Прискорення CUDA half відносно CUDA float	Час виконання ядра	-3,27%	-3,31%	-3,44%
Прискорення CUDA half2 відносно CUDA float	Час виконання ядра	0,21%	0,68%	0,69%
Прискорення CUDA half, використання тензорних ядер, відносно CUDA float	Час виконання ядра	7,53%	8,07%	8,24%
Прискорення CUDA half, використання тензорних ядер, відносно CUDA half	Час виконання ядра	11,16%	11,77%	12,09%

Близькі значення похибок порівнюючи підсумовування згідно з (6.20) та алгоритм (6.21) при $K = 16$ на початкових етапах часу показують адекватність обраного значення K . Однак для алгоритму (6.21), починаючи з деякого кроку за часом спостерігається переповнення значень типу даних у S_{nl} , що призводить до лінійного збільшення похибки. У проведених експериментах така поведінка спостерігалась для 16- та 32-бітових типів даних.

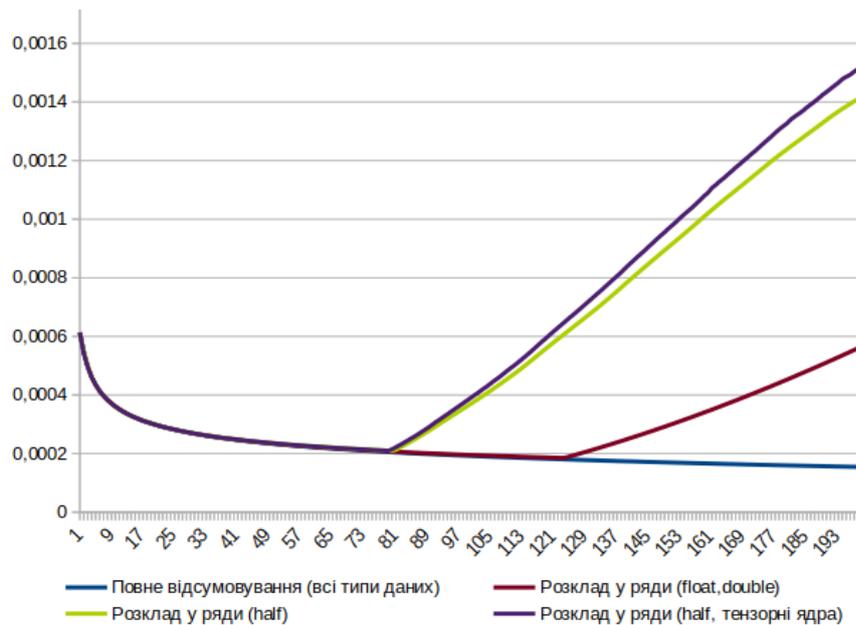


Рисунок 6.8 – Максимальна квадратична абсолютна похибка на кроках за часом при розв’язанні задачі з використанням різних типів даних

6.3. Методика наближеного обчислення ψ -похідної Капуто за просторовою змінною

Розглянемо одновимірне дробово-диференціальне за просторовою змінною рівняння фільтрації, яке, аналогічно [136], може бути представлене у вигляді [39]

$$\frac{\partial}{\partial t} H(x,t) = D_{x,g}^{(\alpha)} \left(k(H) \frac{\partial}{\partial x} H(x,t) \right) - S(x,t), \quad (6.23)$$

де $H(x,t)$ – напір, $k(H)$ – коефіцієнт фільтрації, $S(x,t)$ – функція джерел, $D_{x,g}^{(\alpha)}$ – ліва ψ -похідна Капуто за просторовою змінною виду

$$D_{x,g}^{(\alpha)} H(x,t) = {}^L D_{x,g}^{(\alpha)} H(x,t) = \frac{1}{\Gamma(1-\alpha)} \int_0^x \frac{\partial H(y,t)}{\partial y} (g(x) - g(y))^{-\alpha} dy,$$

$$0 < \alpha \leq 1, 0 < x \leq L, t \geq 0,$$

$$g(x) \in C^1[0, \infty), g'(x) > 0 (x \geq 0), g(0) = 0.$$

Також будемо розглядати рівняння (6.23) у випадку двосторонньої ψ -похідної Капуто, побудованої аналогічно класичній двосторонній похідній Капуто (наприклад, [191]) у вигляді

$${}^{LR}D_{x,g}^{(\alpha)}H(x,t) = {}^L D_{x,g}^{(\alpha)}H(x,t) + {}^R D_{x,g}^{(\alpha)}H(x,t),$$

$${}^R D_{x,g}^{(\alpha)}H(x,t) = -\frac{1}{\Gamma(1-\alpha)} \int_x^L \frac{\partial H(y,t)}{\partial y} (g(y) - g(x))^{-\alpha} dy.$$

Для рівняння (6.23) поставимо наступні початкові та крайові умови:

$$H(0,t) = H_1(t), \left. \frac{\partial H(x,t)}{\partial x} \right|_{x=L} = H_2(t), H(x,0) = H_0(x).$$

Введемо до розгляду неоднорідну скінченно-різницеву сітку

$$\omega = \left\{ (x_i, t_j) : i = \overline{0, m}, j = \overline{0, 1, 2, \dots} \right\}, h_i = x_{i+1} - x_i, \tau_j = t_{j+1} - t_j.$$

Визначимо скінченно-різницевий відповідник $\Delta_x^{(\alpha)}$ оператора диференціювання дробового порядку ${}^L D_{x,g}^{(\alpha)}$, що наближує його з першим порядком точності, наступним чином (наприклад, аналогічно [248]):

$$\Delta_x^{(\alpha)} F = \frac{1}{\Gamma(1-\alpha)} \sum_{s=0}^i \gamma_s^{(i)} \frac{F_{s+1} - F_s}{h_s}, \gamma_s^{(i)} = \int_{x_s}^{x_{s+1}} (g(x_i) - g(y))^{-\alpha} dy. \quad (6.24)$$

У випадку оператора ${}^R D_{x,g}^{(\alpha)}$ маємо

$${}^R \Delta_x^{(\alpha)} F = -\frac{1}{\Gamma(1-\alpha)} \sum_{s=i+1}^{m-1} \bar{\gamma}_s^{(i)} \frac{F_{s+1} - F_s}{h_s}, \bar{\gamma}_s^{(i)} = \int_{x_s}^{x_{s+1}} (g(y) - g(x_i))^{-\alpha} dy.$$

Обчислюючи значення коефіцієнта $k(H)$ для $H = H^{(j-1)}$, де $H^{(j-1)}$ – розв'язок задачі на $j-1$ -му кроці за часом, член $\Delta_x^{(\alpha)} \left(k \frac{\partial H}{\partial x} \right)$ на j -му кроці за часом може бути дискретизований як

$$\begin{aligned} \Delta_x^{(\alpha)} \left(k \frac{\partial H^{(j)}}{\partial x} \right) &\approx \Delta_x^{(\alpha)} \left(k_i \frac{H_i^{(j)} - H_{i-1}^{(j)}}{h_{i-1}} \right) = \\ &= \frac{1}{\Gamma(1-\alpha)} \sum_{s=0}^i \gamma_s^{(i)} \left(\frac{k_s}{h_s^2} H_{s+1}^{(j)} - \left(\frac{k_s}{h_s^2} + \frac{k_{s-1}}{h_s h_{s-1}} \right) H_s^{(j)} + \frac{k_{s-1}}{h_s h_{s-1}} H_{s-1}^{(j)} \right), \end{aligned} \quad (6.25)$$

або

$$\begin{aligned} \Delta_z^{(\alpha)} \left(k \frac{\partial H^{(j)}}{\partial x} \right) &\approx \frac{1}{\Gamma(1-\alpha)} \left(\gamma_i^{(i)} \left(\frac{k_i}{h_i^2} H_{i+1}^{(j)} - \left(\frac{k_i}{h_i^2} + \frac{k_{i-1}}{h_i h_{i-1}} \right) H_i^{(j)} + \frac{k_{i-1}}{h_i h_{i-1}} H_{i-1}^{(j)} \right) + \right. \\ &\left. + \sum_{s=0}^{i-1} \gamma_s^{(i)} \left(\frac{k_s}{h_s^2} H_{s+1}^{(j-1)} - \left(\frac{k_s}{h_s^2} + \frac{k_{s-1}}{h_s h_{s-1}} \right) H_s^{(j-1)} + \frac{k_{s-1}}{h_s h_{s-1}} H_{s-1}^{(j-1)} \right) \right). \end{aligned} \quad (6.26)$$

Дискретизуючи (6.23) за допомогою (6.24) та (6.26), отримуємо наступну тридіагональну СЛАР [39]:

$$\begin{aligned} H_0^j &= H_1(t_j), \\ A_i^{(j)} H_{i-1}^{(j)} - R_i^{(j)} H_i^{(j)} + B_i^{(j)} H_{i+1}^{(j)} &= \Omega_i^{(j)} \quad (i = \overline{1, m-1}; \quad j = 0, 1, 2, \dots), \\ -H_{m-1}^j + H_m^j &= H_2(t_j) * h_{m-1}, \end{aligned} \quad (6.27)$$

де

$$A_i^{(j)} = \frac{\gamma_i^{(i)}}{\Gamma(1-\alpha)} \frac{k_{i-1}}{h_i h_{i-1}}, \quad B_i^{(j)} = \frac{\gamma_i^{(i)}}{\Gamma(1-\alpha)} \frac{k_i}{h_i^2}, \quad R_i^{(j)} = \frac{\gamma_i^{(i)}}{\Gamma(1-\beta)} \left(\frac{k_{i-1}}{h_i h_{i-1}} + \frac{k_i}{h_i^2} \right) + \frac{1}{\tau},$$

$$\Omega_i^{(j)} = \Omega_i^{(1)(j)} - \Omega_i^{(2)(j)}, \quad \Omega_i^{(1)(j)} = -\frac{H_i^{(j-1)}}{\tau} - S_i,$$

$$\Omega_i^{(2)(j)} = -\frac{1}{\Gamma(1-\alpha)} \sum_{s=0}^{i-1} \gamma_s^{(i)} \left(\frac{k_s}{h_s^2} H_{s+1}^{(j-1)} - \left(\frac{k_s}{h_s^2} + \frac{k_{s-1}}{h_s h_{s-1}} \right) H_s^{(j-1)} + \frac{k_{s-1}}{h_s h_{s-1}} H_{s-1}^{(j-1)} \right).$$

У випадку двосторонньої похідної, дискретизація ${}^R \Delta_x^{(\alpha)} \left(k \frac{\partial H}{\partial x} \right)$, подібна

до (6.26), має вигляд

$$\begin{aligned} \Omega_i^{(2)(j)} &= -\frac{1}{\Gamma(1-\alpha)} \sum_{s=0}^{i-1} \gamma_s^{(i)} \left(\frac{k_s}{h_s^2} H_{s+1}^{(j-1)} - \left(\frac{k_s}{h_s^2} + \frac{k_{s-1}}{h_s h_{s-1}} \right) H_s^{(j-1)} + \frac{k_{s-1}}{h_s h_{s-1}} H_{s-1}^{(j-1)} \right) - \\ &- \frac{1}{\Gamma(1-\alpha)} \sum_{s=i+1}^{m-1} \bar{\gamma}_s^{(i)} \left(\frac{k_s}{h_s^2} H_{s+1}^{(j-1)} - \left(\frac{k_s}{h_s^2} + \frac{k_{s-1}}{h_s h_{s-1}} \right) H_s^{(j-1)} + \frac{k_{s-1}}{h_s h_{s-1}} H_{s-1}^{(j-1)} \right). \end{aligned}$$

Зауважимо, що СЛАР (6.27) може бути ефективно розв'язана методом прогонки [313].

При використанні дискретизації (6.25), отримується нижньотрикутна СЛАР, яка, подібно до описаного у [222], може бути представлена у вигляді [39]

$$AH = \Omega^{(1)} - \Omega^{(3)}, \quad A = M^{(1)} + TM^{(2)}, \quad (6.28)$$

$$H_m^{(j)} - H_{m-1}^{(j)} = h_m H_2(t), H_0^{(j)} = H_1(t),$$

$$M^{(1)} = \begin{pmatrix} -R_1 & B_1 & 0 & \dots & 0 \\ A_2 & -R_2 & B_2 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & A_{m-2} & -R_{m-2} & B_{m-2} \\ 0 & \dots & 0 & A_{m-1} & -R_{m-1} \end{pmatrix},$$

$$M^{(2)} = \begin{pmatrix} -\left(\frac{k_1}{h_1^2} + \frac{k_0}{h_1 h_0}\right) & \frac{k_1}{h_1^2} & 0 & \dots & 0 \\ \frac{k_2}{h_2 h_1} & -\left(\frac{k_2}{h_2^2} + \frac{k_1}{h_2 h_1}\right) & \frac{k_2}{h_2^2} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & \frac{k_{m-2}}{h_{m-1} h_{m-2}} & -\left(\frac{k_{m-1}}{h_{m-1}^2} + \frac{k_{m-2}}{h_{m-1} h_{m-2}}\right) \end{pmatrix},$$

$$T = \begin{pmatrix} 0 & \dots & \dots & \dots & 0 \\ \gamma_1^{(2)} & 0 & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \gamma_1^{(m-2)} & \dots & \gamma_{m-3}^{(m-2)} & 0 & 0 \\ \gamma_1^{(m-1)} & \dots & \dots & \gamma_{m-2}^{(m-1)} & 0 \end{pmatrix}, \Omega^{(3)} = \begin{pmatrix} -\frac{k_1}{h_2 h_1} H_0^{(j)} \\ 0 \\ \dots \\ 0 \\ -\frac{k_{m-1}}{h_{m-1}^2} H_m^{(j)} \end{pmatrix}.$$

У випадку двосторонньої похідної отримуємо

$$T = \begin{pmatrix} 0 & \bar{\gamma}_2^{(1)} & \dots & \dots & \bar{\gamma}_{m-1}^{(1)} \\ \gamma_1^{(2)} & 0 & \bar{\gamma}_3^{(2)} & \dots & \bar{\gamma}_{m-1}^{(2)} \\ \dots & \dots & \dots & \dots & \dots \\ \gamma_1^{(m-2)} & \dots & \gamma_{m-3}^{(m-2)} & 0 & \bar{\gamma}_{m-1}^{(m-2)} \\ \gamma_1^{(m-1)} & \dots & \dots & \gamma_{m-2}^{(m-1)} & 0 \end{pmatrix}.$$

Варто зазначити, що матриця T у випадку рівномірної сітки та $g(x) = x$ є матрицею Тєплиця.

При однакових значеннях кроків сітки, розв'язання задачі на основі СЛАР (6.27) є значно швидшим, ніж використання СЛАР (6.28), проте розв'язки є

менш точними. У зв'язку з цим, надалі сконцентруємося на прискоренні обчислень згідно з (6.28), розглядаючи (6.27) як базу для порівняння.

Для чисельного розв'язання початково-крайових задач щодо рівняння (6.23) необхідно обчислювати значення інтегралів $\gamma_s^{(i)}$ і $\bar{\gamma}_s^{(i)}$. При $s=i$, ці інтеграли є сингулярними і вимагають використання спеціальних методів для їх обчислення [290].

Аналогічно описаному у п. 6.1 для випадку похідної за часовою змінною, будемо припускати, що існує нескінченно диференційована функція $f(y)$, обернена до $g(y)$. Тоді, інтеграл $\gamma_s^{(i)}$ можна розкласти у ряди за двома наступними схемами.

Згідно з підходом, використаним для виводу (6.5), маємо [39]

$$\begin{aligned} \gamma_s^{(i)} &= \sum_{n=0}^{\infty} \left((-1)^n \frac{f^{(n+1)}(g(x_i))}{n!} \int_{g(x_i)-g(x_{s+1})}^{g(x_i)-g(x_s)} z^{n-\alpha} dz \right) = \\ &= \sum_{n=0}^{\infty} \left((-1)^n \frac{f^{(n+1)}(g(x_i))}{n!(n-\alpha+1)} \left[(g(x_i)-g(x_s))^{n-\alpha+1} - (g(x_i)-g(x_{s+1}))^{n-\alpha+1} \right] \right). \end{aligned} \quad (6.29)$$

У випадку $\bar{\gamma}_s^{(i)}$ отримуємо

$$\begin{aligned} \bar{\gamma}_s^{(i)} &= \int_{g(x_s)-g(x_i)}^{g(x_{s+1})-g(x_i)} f'(g(x_i)+z) z^{-\alpha} dz, \quad y = g(z) - g(x_i), \\ \bar{\gamma}_s^{(i)} &= \sum_{n=0}^{\infty} \left(\frac{f^{(n+1)}(g(x_{s+1}))}{n!} \tilde{B}_n \right), \\ \tilde{B}_n &= \int_{g(x_s)-g(x_i)}^{g(x_{s+1})-g(x_i)} z^{-\alpha} (z - (g(x_{s+1}) - g(x_i)))^n dz, \\ \tilde{B}_0 &= \frac{1}{1-\alpha} \left((g(x_{s+1}) - g(x_i))^{1-\alpha} - (g(x_s) - g(x_i))^{1-\alpha} \right), \\ \tilde{B}_{i+1} &= -\frac{-\alpha + i + 2}{(g(x_{s+1}) - g(x_i))(i+1)} \left(\tilde{B}_i - \frac{(g(x_s) - g(x_i))^{1-\alpha} (g(x_s) - g(x_{s+1}))^{i+1}}{(g(x_{s+1}) - g(x_i))(i+1)} \right). \end{aligned}$$

Згідно з підходом, використаним для виводу (6.6), отримуємо [39]

$$\gamma_s^{(i)} = \sum_{n=0}^{\infty} \left((-1)^n \binom{-\alpha}{n} g(x_i)^{-\alpha-n} S_n \right),$$

$$S_n(x_s, x_{s+1}) = \int_{g(x_s)}^{g(x_{s+1})} f'(z) z^n dz = \sum_{m=0}^{\infty} \left[B_m \frac{f^{(m+1)}(g(x_{s+1}))}{m!} \right], \quad (6.30)$$

$$B_0 = \int_{g(x_s)}^{g(x_{s+1})} z^n dz = \frac{1}{n+1} \left(g(x_{s+1})^{n+1} - g(x_s)^{n+1} \right),$$

$$B_{i+1} = -\frac{n+i+2}{g(x_{s+1})(i+1)} \left(B_i - \frac{g(x_s)^{n+1} (g(x_s) - g(x_{s+1}))^{i+1}}{g(x_{s+1})(i+1)} \right). \quad (6.31)$$

У випадку інтегралів $\bar{\gamma}_s^{(i)}$ маємо [39]

$$\bar{\gamma}_s^{(i)} = \int_{g(x_s)}^{g(x_{s+1})} f'(z) (z - g(x_s))^{-\alpha} dz = \sum_{n=0}^{\infty} \left((-1)^n \binom{-\alpha}{n} g(x_s)^n \bar{S}_n \right), \quad z = g(y),$$

$$\bar{S}_n(x_s, x_{s+1}) = \sum_{m=0}^{\infty} \left[\bar{B}_m \frac{f^{(m+1)}(g(x_{s+1}))}{m!} \right],$$

$$\bar{B}_m = \int_{g(x_s)}^{g(x_{s+1})} z^{-\alpha-n} (z - g(x_{s+1}))^m dz,$$

$$\bar{B}_0 = \int_{g(x_s)}^{g(x_{s+1})} z^{-\alpha-n} dz = \frac{1}{-\alpha-n+1} \left(g(x_{s+1})^{1-\alpha-n} - g(x_s)^{1-\alpha-n} \right),$$

$$\bar{B}_{i+1} = -\frac{-\alpha-n+i+2}{g(x_{s+1})(i+1)} \left(\bar{B}_i - \frac{g(x_s)^{-\alpha-n+1} (g(x_s) - g(x_{s+1}))^{i+1}}{g(x_{s+1})(i+1)} \right).$$

Використання представлення інтегралів $\gamma_s^{(i)}$ у вигляді рядів (6.29) та (6.30) дозволяє знизити обчислювальну складність операції множення матриці на вектори при розв'язанні СЛАР (6.28) ітераційними методами.

У випадку неоднорідних сіток та/або $g(x) \neq x$, матриця T не є матрицею Теплиця і потребує обсяг пам'яті порядку $O(m^2)$ для зберігання. Обчислювальна складність її множення на вектори також має порядок $O(m^2)$.

Використовуючи (6.30), множення матриці T на вектор v може бути виконане наступним чином:

$$Tv \approx \left(\sum_{j=1}^{i-1} \sum_{n=0}^{K_j} \left((-1)^n \binom{-\alpha}{n} g(x_i)^{-\alpha-n} v_j S_n(x_j, x_{j+1}) \right) \right)_{i=1}^m, \quad (6.32)$$

де K_j – кількість членів у ряді (6.30), необхідних для наближення значень $\gamma_j^{(i)}, i = j+1, \dots, m$ із заданою точністю.

Обчислення згідно з (6.32) організовані наступним чином [39]:

- 1) Значення K_j і $S_n(x_j, x_{j+1}), n = 1, \dots, K_j$ обчислюються для $j = 1, \dots, m-1$;
- 2) Елементи вектора Tv послідовно обчислюються як

$$(Tv)_i \approx \sum_{n=0}^{\max_{j=1, \dots, i-1} K_j} \left((-1)^n \binom{-\alpha}{n} g(x_i)^{-\alpha-n} \bar{S}_{n,i} \right),$$

$$\bar{S}_{n,i} = \bar{S}_{n,i-1} + \begin{cases} v_{i-1} S_n(x_{i-1}, x_i), n \leq K_{i-1}, \bar{S}_{n,1} = 0. \\ 0, n > K_{i-1} \end{cases} \quad (6.33)$$

Складність процедури (6.33) має порядок $O\left(m \max_{j=1, \dots, m-1} K_j\right)$. Оскільки збіжність ряду (6.30) погіршується, коли $t_{s+1} \rightarrow t_j$, пропонується наближено обчислювати значення інтегралів $\gamma_s^{(i)}, i-s \leq K_d$ використовуючи ряд (6.29). Значення інтегралів $\gamma_s^{(i)}, i-s \leq K_d$ обчислюються на етапі ініціалізації і (6.33) приймає вигляд

$$(Tv)_i \approx \sum_{m=i-K_d}^{i-1} \gamma_m^{(i)} + \sum_{n=0}^{\max_{j=1, \dots, i-1} K'_j} \left((-1)^n \binom{-\alpha}{n} g(x_i)^{-\alpha-n} S'_{n,i} \right),$$

$$S'_{n,i} = S'_{n,i-1} + \begin{cases} v_{i-1} S'_n(x_{i-1}, x_i), n \leq K'_{i-1}, S'_{n,1} = 0, \\ 0, n > K'_{i-1} \end{cases} \quad (6.34)$$

де K'_j – кількість членів у ряді (6.30), необхідних для наближення значень $\gamma_j^{(i)}, i = j+K_d, \dots, m$ із заданою точністю.

Твердження 6.1. Обчислювальна складність процедури (6.34) має порядок

$$O\left(m \left(K_d + \max_{j=1, \dots, m-1} K'_j \right)\right).$$

Пропонується знаходити оптимальне значення K_d ітеративно збільшуючи його від початкового значення $K_d = 1$ доти, доки $K_d + \max_{j=1, \dots, m-1} K'_j$ не перестане зменшуватися.

Відповідна процедура у випадку двосторонньої дробової похідної одержується подібним чином.

Теорема 6.1. Наближенне значення \bar{T} верхньої межі обчислювальної складності $T = K_d + \max_{j=1, \dots, m-1} K'_j$ процедури (6.34) розрахунку значення $(Tv)_i$ за припущень

$$1) x_i = ih, g(x) = x^\beta, f'(z) = \frac{1}{\beta} x^{\beta-1},$$

$$2) |f'(z)| \leq F, z \in [g(x_j), g(x_{j+1})], F = \max(f'(g(x_j)), f'(g(x_{j+1}))),$$

$$3) m \gg 1,$$

$$4) \text{максимальне значення } \max_{j=1, \dots, m-1} K'_j \text{ досягається при } i = m, j = m - K_d,$$

має вигляд

$$\bar{T} = \sqrt{\frac{m(2-\alpha)}{2\beta}} \ln \left(\frac{\beta}{2m(2-\alpha)} e^{2 \frac{\ln \varepsilon - K_1}{\alpha-2} + 1} \right) - 1,$$

$$\text{де } K_1 = \left(\frac{1}{2} - \alpha \right) \ln \alpha - \frac{1}{2} \ln(2\pi) + \ln F + \beta(1-\alpha) \ln(ih).$$

Ремарка 1. Припущення 2 не виконується, коли $\beta > 1$ і $0 \in [g(x_j), g(x_{j+1})]$.

Ремарка 2. Припущення 4 базується на тому факті, що кількість доданків у ряді (6.30), необхідних для обчислення значень інтегралів $\gamma_j^{(i)}$ із заданою точністю (подібний підхід також застосовний і до інтегралів $\bar{\gamma}_j^{(i)}$) збільшується зі збільшенням i і $j \rightarrow i$).

Доведення.

За вищенаведених припущень маємо

$$\begin{aligned} \gamma_j^{(i)} &= \sum_{n=0}^{\infty} I_n, I_n = (-1)^n \binom{-\alpha}{n} g(x_i)^{-\alpha-n} \int_{g(x_j)}^{g(x_{j+1})} f'(z) z^n dz \leq \\ &\leq (-1)^n \binom{-\alpha}{n} (ih)^{-\beta(\alpha+n)} \frac{F}{n+1} \left(((j+1)h)^{\beta(n+1)} - (jh)^{\beta(n+1)} \right). \end{aligned} \quad (6.35)$$

Після застосування наближення Стірлінга до множника $(-1)^n \binom{-\alpha}{n}$ отримуємо

$$(-1)^n \binom{-\alpha}{n} = \frac{(\alpha)^n}{n!} = \frac{\Gamma(\alpha+n)}{\Gamma(\alpha)n!} \approx \sqrt{\frac{\alpha}{2\pi n(\alpha+n)}} \frac{(\alpha+n)^{\alpha+n}}{\alpha^\alpha n^n}. \quad (6.36)$$

Підставивши (6.36) у (6.35), отримуємо наступну оцінку для члена ряду (6.30):

$$I_n \approx \tilde{I}_n = \sqrt{\frac{\alpha}{2\pi n(\alpha+n)}} \frac{(\alpha+n)^{\alpha+n}}{\alpha^\alpha n^n} (ih)^{-\beta(\alpha+n)} \times \frac{F}{n+1} \left(((j+1)h)^{\beta(n+1)} - (jh)^{\beta(n+1)} \right).$$

Номер $n(j, i)$ члена ряду (6.30), для якого $I_n = \varepsilon$, можна оцінити з урахуванням наступного:

$$\begin{aligned} \ln \tilde{I}_n = \ln \varepsilon \Rightarrow, \\ \left(\frac{1}{2} - \alpha \right) \ln \alpha - \frac{1}{2} \ln(2\pi) + \ln F - \\ - \left(\frac{1}{2} + n \right) \ln n - \ln(n+1) + \left(\alpha + n - \frac{1}{2} \right) \ln(\alpha+n) - \\ - \beta(\alpha+n) \ln(ih) + \ln \left(((j+1)h)^{\beta(n+1)} - (jh)^{\beta(n+1)} \right) = \ln \varepsilon. \end{aligned} \quad (6.37)$$

Для послідовного наближення використаємо нерівності $\ln(n+1) \geq \ln(n+\alpha) \geq \ln n$, $((j+1)h)^{\beta(n+1)} - (jh)^{\beta(n+1)} \leq ((j+1)h)^{\beta(n+1)}$. Тоді (6.37) приймає вигляд

$$\begin{aligned} K_1 + (\alpha - 2) \ln(n+1) + (n+1)K_2 &= \ln \varepsilon, \\ K_1 &= \left(\frac{1}{2} - \alpha \right) \ln \alpha - \frac{1}{2} \ln(2\pi) + \ln F + \beta(1-\alpha) \ln(ih), \\ K_2 &= \beta(\ln(j+1) - \ln i), \end{aligned}$$

а наближення $n_1(j, i)$ верхньої межі числа n дорівнює

$$\frac{(n+1)K_2}{\alpha-2} e^{\frac{(n+1)K_2}{\alpha-2}} = \frac{K_2}{\alpha-2} e^{\frac{\ln \varepsilon - K_1}{\alpha-2}},$$

$$n_1(j, i) = \frac{\alpha - 2}{K_2} W \left(\frac{K_2}{\alpha - 2} e^{\frac{\ln \varepsilon - \bar{K}_1}{\alpha - 2}} \right) - 1, \quad (6.38)$$

де W – функція Ламберта [86].

Наближення $n_2(j, i)$ нижньої межі числа n можна отримати з урахуванням того, що

$$((j+1)h)^{\beta(n+1)} - (jh)^{\beta(n+1)} = (jh)^{\beta(n+1)} \left(\left(\frac{j+1}{j} \right)^{\beta(n+1)} - 1 \right),$$

$$e^x \geq \left(1 + \frac{x}{n} \right)^n \Rightarrow e^{\frac{\beta(n+1)}{j}} \geq \left(\frac{j+1}{j} \right)^{\beta(n+1)},$$

$$e^{\frac{\beta(n+1)}{j}} - 1 \geq \frac{\beta(n+1)}{j} \Rightarrow ((j+1)h)^{\beta(n+1)} - (jh)^{\beta(n+1)} \approx (jh)^{\beta(n+1)} \frac{\beta(n+1)}{j}.$$

Звідси маємо

$$\bar{K}_1 + (\alpha - 1) \ln(n+1) + (n+1) \bar{K}_2 = \ln \varepsilon,$$

$$\bar{K}_1 = \left(\frac{1}{2} - \alpha \right) \ln \alpha - \frac{1}{2} \ln(2\pi) + \ln F + \beta(1 - \alpha) \ln(ih) + \ln \frac{\beta}{j},$$

$$\bar{K}_2 = \beta(\ln j - \ln i),$$

$$n_2(j, i) = \frac{\alpha - 1}{\bar{K}_2} W \left(\frac{\bar{K}_2}{\alpha - 1} e^{\frac{\ln \varepsilon - \bar{K}_1}{\alpha - 1}} \right) - 1. \quad (6.39)$$

Подібні наближення для інтегралів $\bar{\gamma}_j^{(i)}$ можна отримати, використовуючи нерівність $\ln(\alpha + n - 1) \leq \ln(n + \alpha) \leq \ln n$ та розв'язки рівнянь вигляду

$$a_1 + a_2 \ln(\alpha + n - 1) + a_3(\alpha + n - 1) = \ln \varepsilon.$$

Відштовхуючись від припущення 4 для $T = K_d + \max_{j=1, \dots, m-1} K'_j$ маємо

$$T = K_d + n(m - K_d, m). \quad (6.40)$$

Оптимальне значення K_d можна знайти розв'язуючи рівняння

$$\partial T / \partial K_d = 0.$$

Враховуючи складність оцінок числа n спростимо їх, використовуючи наступне:

$$\ln x < x - 1, x \rightarrow 1 \Rightarrow \bar{K}_2 = \beta \ln \frac{m - K_d}{m} < -\frac{\beta K_d}{m};$$

$W(x) \geq \ln x - \ln \ln x + \frac{\ln \ln x}{2 \ln x}, x > e$ й, таким чином, $W(x) \geq \ln x$, оскільки

$$-\ln \ln x + \frac{\ln \ln x}{2 \ln x} < 0, x > \sqrt{e};$$

$$\ln(m - K_d) < \ln m \Rightarrow \ln F = \ln \frac{1}{\beta} + (1 - \beta) \ln((m - K_d)h) < \ln \frac{1}{\beta} + (1 - \beta) \ln(mh).$$

За цих припущень, які виконуються при $m \gg 1$, з використанням наближення $n_1(j, i)$ отримуємо

$$T \leq \tilde{T} = K_d + \frac{(2 - \alpha)m}{\beta K_d} \ln \left(\frac{\beta K_d}{m(2 - \alpha)} e^{\frac{\ln \varepsilon - K_1}{\alpha - 2}} \right) - 1. \quad (6.41)$$

З $\partial \tilde{T} / \partial K_d = 0$ маємо

$$\tilde{\alpha} K_d^2 = \ln(\tilde{b} K_d) - 1, \tilde{\alpha} = \frac{\beta}{m(2 - \alpha)}, \tilde{b} = \tilde{\alpha} e^{\frac{\ln \varepsilon - K_1}{\alpha - 2}}. \quad (6.42)$$

Дійсним розв'язком (6.42) є

$$\bar{K}_d = \sqrt{\frac{-W_{-1} \left(-\frac{2e^2 \tilde{\alpha}}{\tilde{b}} \right)}{2\tilde{\alpha}}} \geq \bar{K}_{d0} = \sqrt{\frac{1}{2\tilde{\alpha}}}, 0 \leq \frac{2e^2 \tilde{\alpha}}{\tilde{b}} \leq \frac{1}{e},$$

а наближення \bar{T} верхньої межі обчислювальної складності процедури (6.34) має вигляд

$$\bar{T} = \tilde{T}(\bar{K}_{d0}) = \sqrt{\frac{m(2 - \alpha)}{2\beta}} \ln \left(\frac{\beta}{2m(2 - \alpha)} e^{2\frac{\ln \varepsilon - K_1}{\alpha - 2} + 1} \right) - 1. \quad (6.43)$$

□

Щодо наближення матриці T існує наступна оцінка її точності.

Теорема 6.2. Позначаючи як \tilde{T} наближення матриці T , отримане апроксимацією інтегралів $\gamma_s^{(i)}$ рядом (6.30), обмеженим згідно з умовою $I_n \geq \varepsilon$, маємо наступну оцінку $R_T(\varepsilon)$ точності наближення матриці T матрицею \tilde{T} :

$$R_T(\varepsilon) = \max_{i,j} (T_{ij} - \tilde{T}_{ij})^2 \leq \left(R_{m-\bar{K}_{d0}}^{(m)}(n_1(m, m - \bar{K}_{d0}), \varepsilon) \right)^2,$$

де $R_s^{(i)}(n, \varepsilon) = \frac{\varepsilon n^{1-\alpha} (n+1)^\alpha}{(1-\alpha)} \frac{g(x_{s+1})}{g(x_i)}$, а $n_1(j, i)$ визначається згідно з (6.38).

Доведення.

Побудуємо оцінку точності апроксимації у випадку запропонованої обчислювальної схеми, використовуючи наступний взаємозв'язок між членами ряду (6.30):

$$\frac{I_{n+1}}{I_n} = - \frac{\binom{-\alpha}{n+1} \int_{g(x_s)}^{g(x_{s+1})} z^{n+1} f'(z) dz}{\binom{-\alpha}{n} g(x_i) \int_{g(x_s)}^{g(x_{s+1})} z^n f'(z) dz}.$$

Розглядаючи функцію $f'(z)$ постійною та обмеженою на інтервалі $[g(x_j), g(x_{j+1})]$ та враховуючи, що припущення про обмеженість не виконується, коли $\beta > 1$ і $0 \in [g(x_j), g(x_{j+1})]$, маємо

$$I_{n+j} \leq (-1)^j \frac{\binom{-\alpha}{n+j} (n+1)}{\binom{-\alpha}{n} (n+j+1)} \left(\frac{g(x_{s+1})}{g(x_i)} \right)^j.$$

Похибка наближення інтегралів $\gamma_s^{(i)}$ рядом (6.30), обмеженим згідно з умовою $I_n \geq \varepsilon$ дорівнює $R_s^{(i)}(n, \varepsilon) = \sum_{j=1}^{\infty} I_{n+j}$. Застосовуючи наближення

$$\binom{-\alpha}{n} \approx \frac{1}{\Gamma(\alpha) n^{1-\alpha}} \text{ отримуємо}$$

$$R_s^{(i)}(n, \varepsilon) \leq \varepsilon n^{1-\alpha} (n+1) \sum_{j=1}^{\infty} \left(\frac{1}{(n+j)^{1-\alpha} (n+j+1)} \left(\frac{g(x_{s+1})}{g(x_i)} \right)^j \right).$$

Враховуючи, що $\frac{g(x_{s+1})}{g(x_i)} \leq 1$, $n+j+1 > n+j$ і $\sum_{j=1}^{\infty} \frac{1}{(n+j+1)^{2-\alpha}} \leq \frac{1}{(1-\alpha)(n+1)^{1-\alpha}}$.

отримуємо наступну оцінку точності обчислення інтегралів $\gamma_s^{(i)}$:

$$R_s^{(i)}(n, \varepsilon) \leq \frac{\varepsilon n^{1-\alpha} (n+1)^\alpha}{(1-\alpha)} \frac{g(x_{s+1})}{g(x_i)}.$$

Аналогічно для інтегралів $\bar{\gamma}_s^{(i)}$ отримуємо

$$\bar{R}_s^{(i)}(n, \varepsilon) \leq \frac{\varepsilon n^{1-\alpha} (n+1)^\alpha}{(1-\alpha)} \frac{g(x_i)}{g(x_{s+1})}.$$

Враховуючи, що $R_s^{(i)}(n, \varepsilon)$ збільшується при $\frac{g(x_{s+1})}{g(x_i)} \rightarrow 1$, для $R_T(\varepsilon)$ маємо

$$R_T(\varepsilon) \leq \left(R_{m-\bar{K}_{d0}}^{(m)}(n_1(m, m - \bar{K}_{d0}), \varepsilon) \right)^2. \quad (6.44)$$

□

Наслідок 1. Оскільки $\frac{g(x_{m-\bar{K}_{d0}+1})}{g(x_m)} \rightarrow 1$ та $n_1(m, m - \bar{K}_{d0}) \rightarrow \infty$ при

$m \rightarrow \infty$, $h_i \rightarrow 0$, $i = \overline{0, m}$, маємо $R_T(\varepsilon) \rightarrow \left(\frac{\varepsilon \cdot n_1(m, m - \bar{K}_{d0}, \varepsilon)}{1-\alpha} \right)^2 \rightarrow \infty$. Таким чином,

запропонована схема не є збіжною для фіксованого ε при $m \rightarrow \infty$. Використання процедури наближення не впливає на порядок точності скінченно-різницевої схеми, якщо виконується умова

$$\varepsilon = c \cdot \frac{1-\alpha}{n_1^d(m, m - \bar{K}_{d0}, \varepsilon)}, \quad d > 1, \quad (6.45)$$

де c – задана стала.

Враховуючи складність аналітичного розв'язання рівняння (6.45), будемо надалі використовувати при розв'язанні задачі (6.23) з використанням схеми (6.34) наступну спрощену оцінку при адаптації ε до розміру сітки:

$$\tilde{\varepsilon} = c \cdot \frac{1-\alpha}{n_1^d(m, m - \bar{K}_{d0}, \varepsilon_0)}, \quad (6.46)$$

де ε_0 – задана стала.

Обчислювальні схеми множення матриці A на вектори

Будемо розглядати наступні схеми виконання операції множення матриці A на вектори:

1) Значення всіх елементів матриці A обчислюються на етапі ініціалізації. Множення виконується з використанням повністю визначеної матриці A . Кількість блоків пам'яті, необхідних для зберігання матриці, тут дорівнює $N^{(1)} = \frac{1}{2}m^2 + \frac{3}{2}m - 2$, а обчислювальна складність виконання операції множення має порядок $O(m^2)$;

2) У випадку однорідної сітки та $g(x) = x$, значення елементів тридіагональних матриць $M^{(1)}$, $M^{(2)}$ та вектор $T_v = (\gamma_1^{(2)}, \dots, \gamma_1^{(m-2)})$, що представляє елементи, у цьому випадку, матриці Теплиця T , обчислюються на етапі ініціалізації. Операція множення виконується як $Av = M^{(1)}v + TM^{(2)}v$. Кількість блоків пам'яті, необхідних для зберігання матриці, дорівнює $N^{(2)} = 7m - 9$, а обчислювальна складність виконання операції множення має, як і в попередньому випадку, порядок $O(m^2)$;

3) У випадку однорідної сітки і $g(x) = x$, значення елементів тридіагональних матриць $M^{(1)}$, $M^{(2)}$ і вектор

$$T_v = (\gamma_1^{(2)}, \dots, \gamma_1^{(m-2)}, 0, 0, \dots, 0),$$

а у випадку двосторонньої похідної – вектор

$$T_v = (\gamma_1^{(2)}, \dots, \gamma_1^{(m-2)}, 0, \bar{\gamma}_1^{(m-2)}, \dots, \bar{\gamma}_1^{(2)}), \dim T_v = 2m - 5,$$

обчислюються на етапі ініціалізації. Множення виконується як $Av = M^{(1)}v + TM^{(2)}v$. Вираз $Tw, w = M^{(2)}v$ обчислюється з використанням відомого алгоритму [163] із порядком складності $O(n \ln n)$:

$$Tw = \left[iFFT \left(FFT(T_v) \otimes FFT(w_1, \dots, w_{m-2}, 0, 0, \dots, 0) \right) \right]_1^{m-2},$$

де FFT – операція швидкого перетворення Фур'є, $iFFT$ – операція зворотного швидкого перетворення Фур'є, \otimes – по-компонентний добуток векторів, $[v]_1^{m-2}$ – підвектор вектору v з індексами елементів у діапазоні від 1 до $m-2$. У цьому випадку кількість блоків пам'яті, необхідних для зберігання матриці, дорівнює $N^{(3)} = 8m - 12$, а складність операції множення має порядок $O(m \ln m)$;

4) Значення елементів тридіагональних матриць $M^{(1)}$, $M^{(2)}$ обчислюються на етапі ініціалізації. Множення виконується згідно з (6.34). Кількість блоків пам'яті, необхідних для зберігання матриці, тут дорівнює $N^{(4)} = 6m - 8 + m \left(K_d + \max_{j=1, \dots, m-1} K'_j \right)$, а складність операції множення має порядок $O \left(m \left(K_d + \max_{j=1, \dots, m-1} K'_j \right) \right)$.

Зазначимо, що у випадку рівняння (6.23) з двосторонньою дробовою похідною, складність алгоритмів 1, 2 та 4 подвоюється.

Тестування алгоритмів

У якості модельної задачі приймемо початково-крайову задачу для рівняння (6.23) з наступними значеннями параметрів:

$$k = 0,5 + x,$$

$$S = \frac{2t}{L^3 T} \left(\frac{x^{1-\alpha} \Gamma(2)}{2\Gamma(2-\alpha)} + \frac{x^{2-\alpha} \Gamma(3)}{\Gamma(3-\alpha)} \right) - \left(\frac{1}{T^3} (3t^2 - 4t) - \frac{x^2}{TL^3} \right), \quad (6.47)$$

$$\bar{H}(x, t) = \frac{tx^2}{TL^3} + \frac{1}{T^3} (t^3 - 2t^2), H_1 = \bar{H}(0, t), H_2 = \left. \frac{\partial \bar{H}}{\partial x} \right|_{x=L}, H_0 = \bar{H}(x, 0),$$

$$L = 3; T = 8,64 \cdot 10^5; \tau = 10^4.$$

Функція $\bar{H}(x, t)$ є розв'язком задачі (6.23), (6.47) при $g(x) = x$. Розв'язок задачі за різних $g(x)$ з використанням лівих та двосторонніх дробових похідних зображено на рис. 6.9. Вихідні коди розробленого програмного забезпечення наведено у [265].

СЛАР (6.28) розв'язувалась алгоритмом BiCGstab [219] із застосуванням діагонального передобумовлювача за наступних критеріїв завершення ітераційного процесу:

- Нев'язка стає менше 10^{-15} ;

- Виконано понад 5000 ітерацій.

У проведених обчислювальних експериментах вимірювався час, витрачений на розв'язання задачі на 14 кроках за часом.

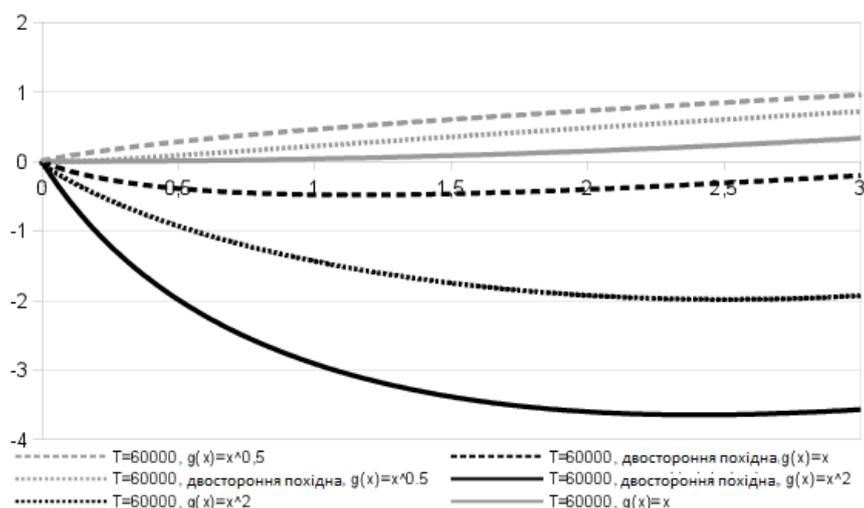


Рисунок 6.9 – Розв'язки задачі

Обчислювальна схема (6.27) була значно швидшою за схему (6.28), але діапазон її збіжності був обмежений умовою $\alpha \in [0,8;1]$. Застосовність схеми (6.28) обмежувалась ширшим діапазоном $\alpha \in [0,025;1]$, тож надалі розглядатимемо лише цю схему.

Десятковий логарифм похибки розв'язання $\varepsilon_1 = \frac{1}{m} \sum_{i=1}^m (\tilde{H}_i - \bar{H}_i)^2$, де \tilde{H} – чисельний розв'язок, для схеми (6.28) наведено на рис. 6.10 для $\alpha = 0,7$ при зміні значення m та на рис. 6.11 для $m = 256$ при зміні значення α .

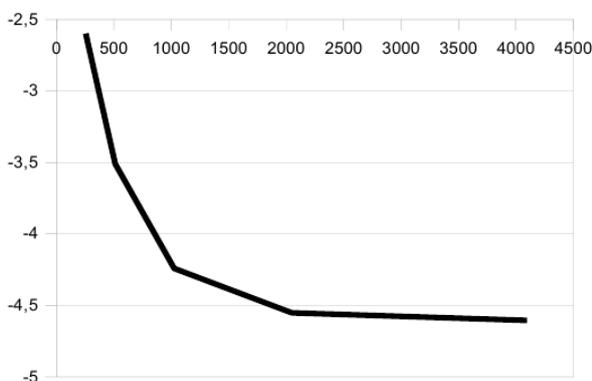


Рисунок 6.10 – Десятковий логарифм похибки розв'язання при зміні значення m

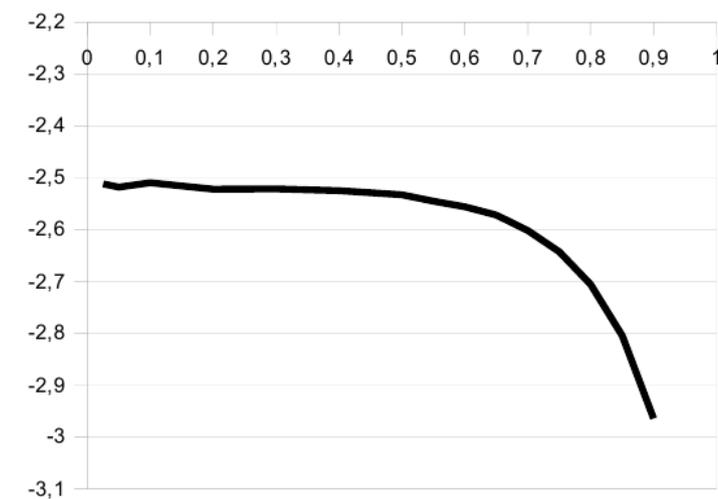


Рисунок 6.11 – Десятковий логарифм похибки розв’язання при зміні значенні α

Тут вищі значення локальних похибок були зосереджені в межах діапазону $x \in [0; 0,5]$ та зменшувалися зі збільшенням x . Використання нерівномірної сітки зі згущенням вузлів при малих значеннях x у розглядуваному випадку дозволяє поліпшити якість розв’язків без збільшення кількості вузлів сітки. Локальні похибки для $T = 6 \cdot 10^4$, $\alpha = 0.7$, $m = 256$ наведені на рис. 6.12 для двох сіток, розміри кроків яких наведені на рис. 6.13.

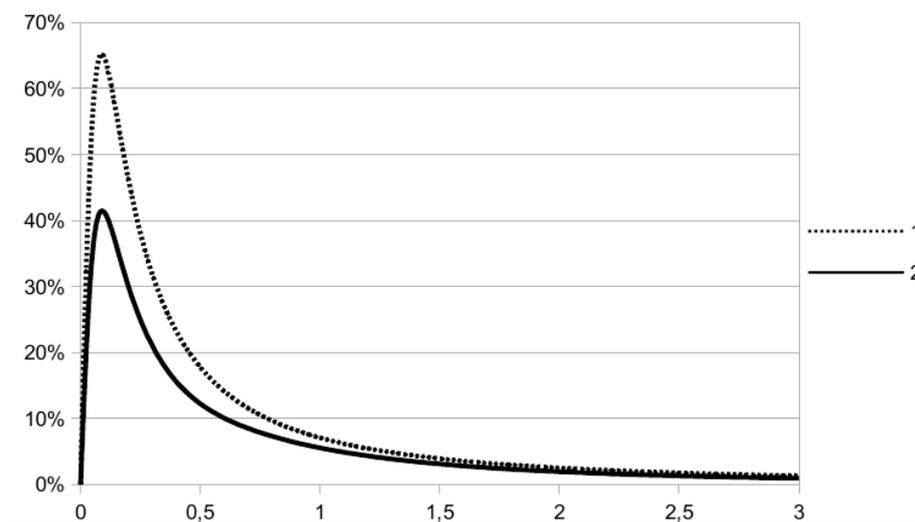


Рисунок 6.12 – Локальні похибки для однорідної (1) та нерівномірної (2) сіток

Найшвидшим алгоритмом множення матриці на вектори при розв’язанні задачі згідно зі схемою (6.28) на однорідній сітці для $g(x) = x$ був алгоритм 3, який базується на використанні швидкого множення матриць Теплиця. Залежність часу,

витраченого на обчислення на одному кроці за часом, від номера кроку m (рис. 6.14) відповідала теоретичними оцінками: була квадратичною для алгоритмів 1 і 2, логарифмічною – для алгоритму 3. Час виконання алгоритму 1 був близьким до часу виконання алгоритму 2.

Використання нерівномірної сітки не суттєво вплинуло на час роботи алгоритмів у проведених обчислювальних експериментах.

Час, витрачений на обчислення на одній ітерації для алгоритмів 1–3, не залежав від α .

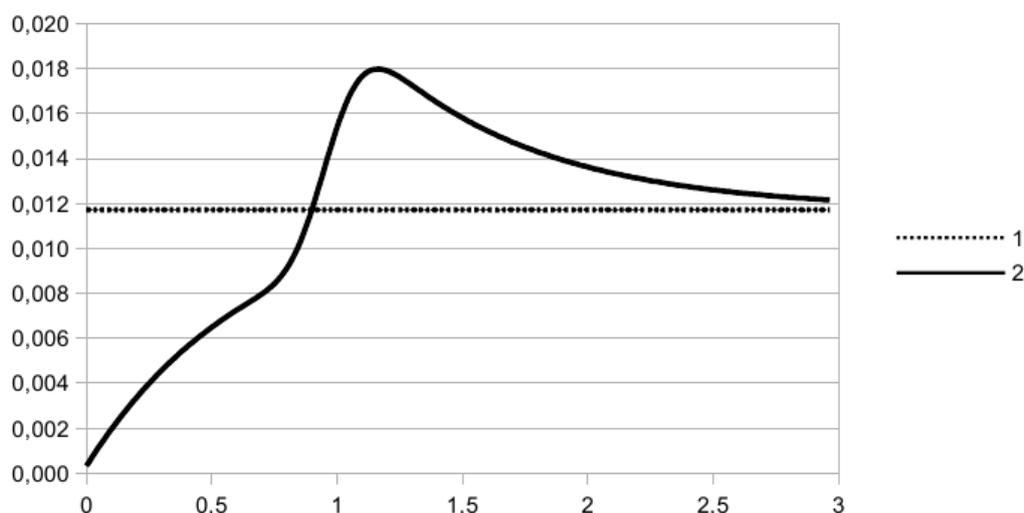


Рисунок 6.13 – Розмір кроку для однорідної (1) та нерівномірної (2) сіток

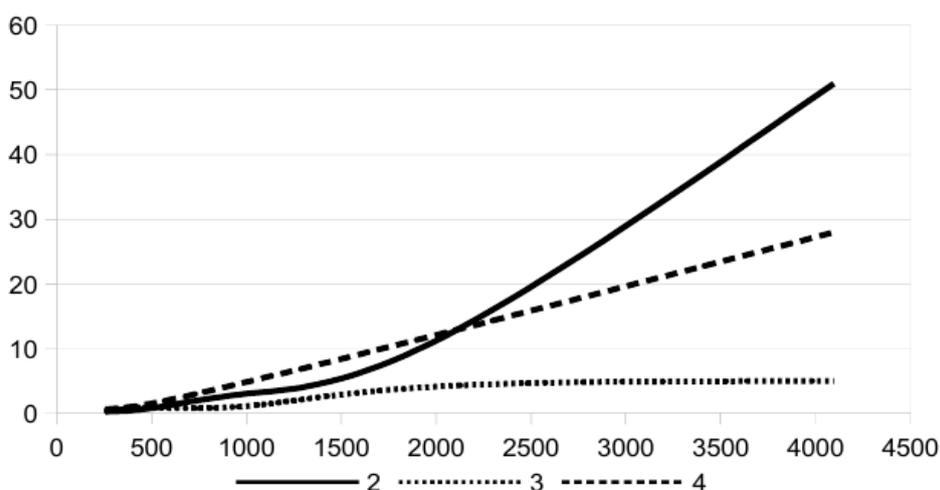


Рисунок 6.14 – Час, мс, виконання однієї ітерації BiCGstab для різних алгоритмів множення матриці на вектори в залежності від m для $\alpha = 0,7$

У випадках, коли алгоритми 2 і 3 не можуть бути застосовані, запропонований алгоритм 4 дозволяв прискорити процес розв'язання у порівнянні з базовим алгоритмом 1. Час його виконання залежав від форми функції $g(x)$ згідно з оцінкою (6.44) (рис. 6.15). На час виконання алгоритму 1 зміна форми функції $g(x)$ не впливала.

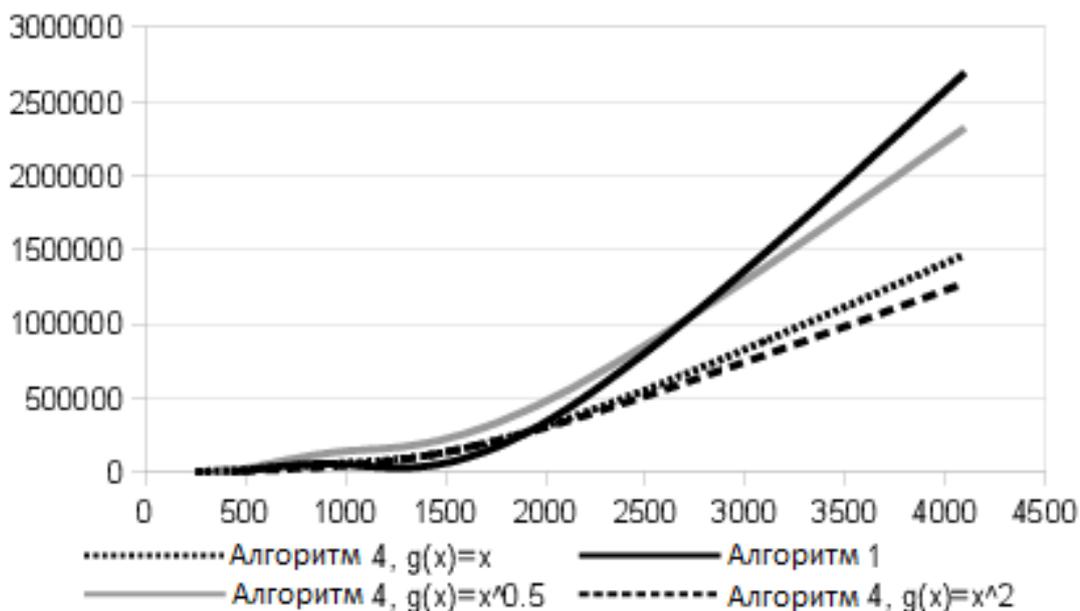


Рисунок 6.15 – Загальний час виконання, мс, для різних форм $g(x)$ і $\alpha = 0,7$

Експериментальна перевірка оцінок складності

Загалом, алгоритм 4 був швидшим, ніж алгоритм 1 при $m > 3000$. Надалі, дослідимо вплив значень параметрів задачі на обчислювальну складність її розв'язання та точність оцінок складності. На рис. 6.16 показані експериментально отримані значення числа $Q = K_d + \max_{j=1, \dots, m-1} K'_j$, яке відповідає обчислювальній складності операції множення рядків матриці T на вектори (крива 3), та його оцінки в залежності від розміру сітки m для $\alpha = 0,7; \beta = 0,5; \varepsilon = 10^{-5}$ (крива 1 відображає значення оцінок (6.38), (6.40) з наближенням функції Ламберта другого порядку точності, крива 2 – оцінки (6.41), крива 4 – оцінки (6.39), (6.38) з наближенням функції Ламберта другого порядку точності, крива 5 – оцінки (6.43)).

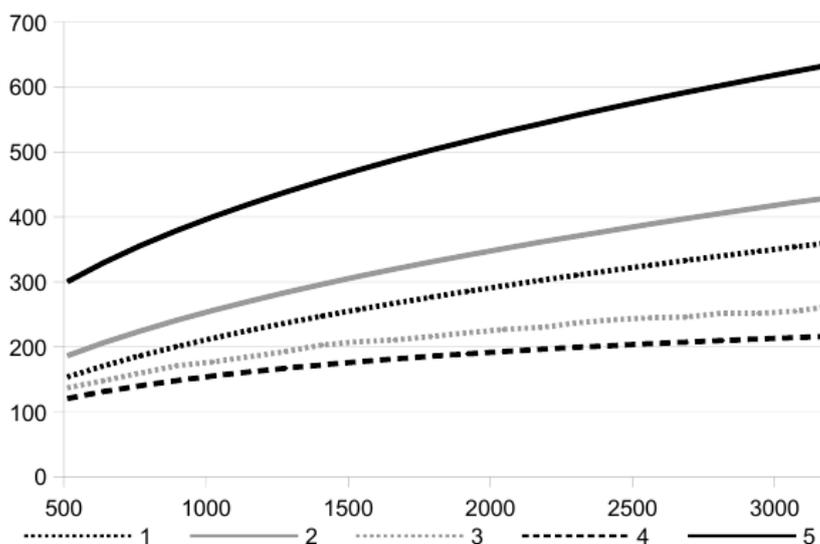


Рисунок 6.16 – Обчислювальна складність операції множення рядків матриці T на вектори в залежності від розміру сітки

Обчислювальні експерименти, проведені зі зміною значень параметрів α і β , показали, що спрощення верхніх оцінок, зокрема з використанням наближень функції Ламберта низького порядку точності, призводить до значного збільшення значень оцінок. Загальні тенденції їхньої поведінки одночасно збігаються із тенденціями змін експериментально отриманих значень.

Значення Q у випадку, коли запропонований підхід був використаний для апроксимації правосторонньої дробової похідної, суттєво не відрізнялись від значень, отриманих при апроксимації лівосторонньої похідної.

При розв'язанні тестової задачі, число n_0 ітерацій алгоритму BiCGstab, необхідних для досягнення заданої точності розв'язання системи (6.28), лінійно збільшувалось зі збільшенням розміру сітки для фіксованого α .

Кількість ітерацій також зростала для $\alpha \rightarrow 1$ і $\alpha \rightarrow 0$. Число n_0 значно зростало при $\alpha < 0,45$ у зв'язку з частим введенням алгоритму BiCGstab у стан стагнації.

У випадку двосторонньої похідної, відповідна кількість n_1 ітерацій зростала, зберігаючи загальні тенденції поведінки. Водночас значення n_1/n_0 лінійно

збільшувалось зі збільшенням розміру сітки m . Зростання значення n_1/n_0 спостерігалось також зі зменшенням α .

Для $g(x) = x^p$, вплив показника p на зміну кількості ітерацій у випадках як лівосторонньої, так і двосторонньої похідних був незначним, проте зростав зі збільшенням m . Кількість ітерацій також зростала також і зі збільшенням p .

Кількість ітерацій n_0 у проведених експериментах корелювала з числом обумовленості діагонально масштабованих матриць у нормі $\|A\|_1$ (рис. 6.17). Ступінь кореляції зменшувалася зі зменшенням α . Це можна пояснити тим, що уявна складова власних значень матриць СЛАР (6.28) зростає зі зменшенням α (рис. 6.18). Збіжність алгоритму BiCGstab у цій ситуації, відповідно до [204], погіршується зокрема через підвищену ймовірність впадання у стан стагнації.

У обчислювальних експериментах, при зміні значень параметрів в межах $\alpha \in [0.1, 0.9]$, $p \in [0.25, 2]$, $m \in [256, 768]$, число обумовленості C корелювало з коефіцієнтом детермінації, рівним 0,993, з функцією

$$\tilde{C}(\alpha, p, m) = \left(\left(1 + \frac{2p^{0,115}}{\alpha} \right)^{1/2} - 1.9 \right) \cdot m^3.$$

Ця кореляція ставала більш тісною для більших значень C .

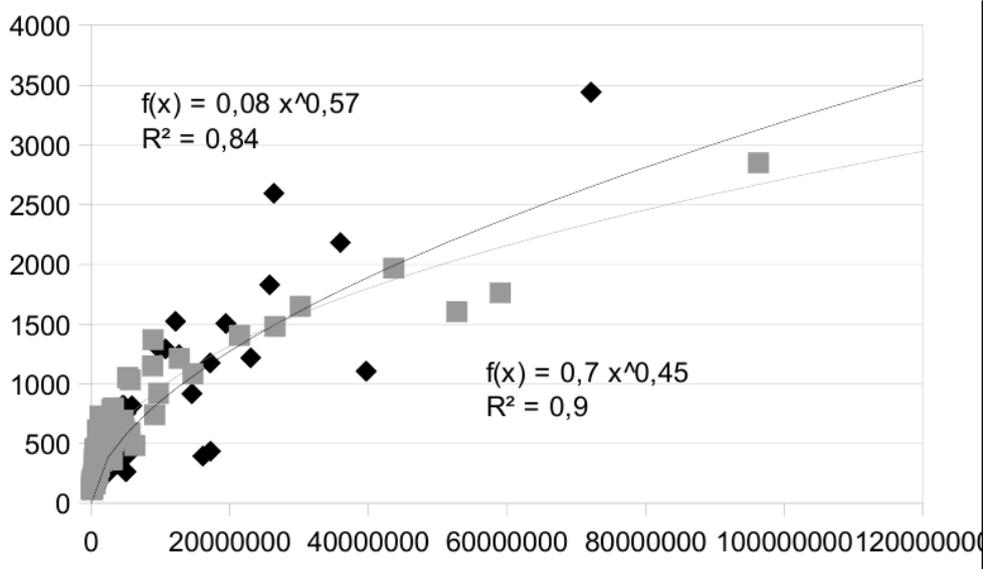


Рисунок 6.17 – Кореляція між числом обумовленості матриць та кількістю ітерацій, необхідних для розв’язання СЛАР (чорний – $\alpha \leq 0,3$; сірий – $\alpha > 0,3$)

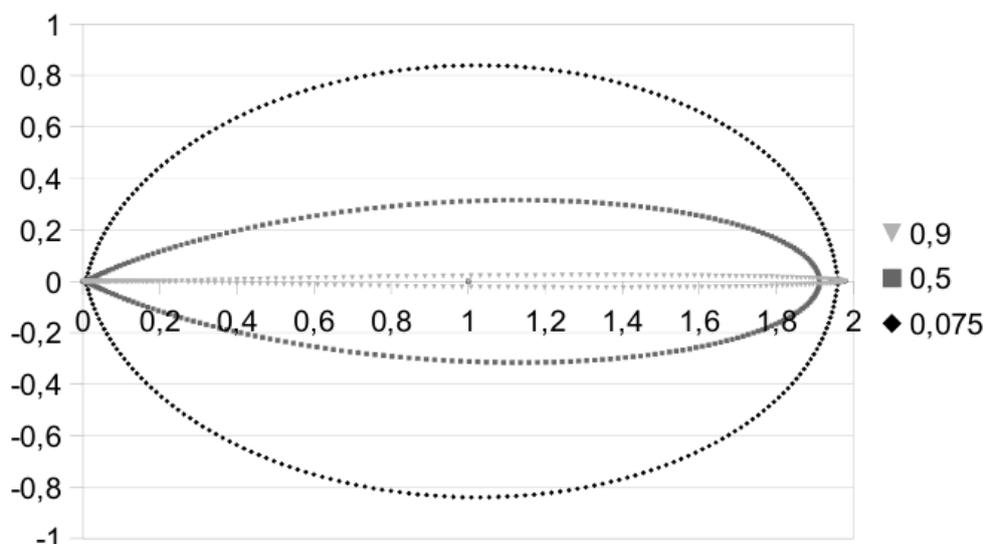


Рисунок 6.18 – Спектр власних значень матриць для різних значень порядку дробової похідної α

У наступній серії експериментів розраховувалась середньоквадратична похибка наближення матриці T .

Для $g(x) = x^p$, похибка зростала зі збільшенням розміру сітки m та зменшувалась зі збільшенням p . Оцінки похибок (6.44) були значно вищими за фактичні значення, але точно відображали їх поведінку при зміні параметрів задачі.

Залежність похибки від α у проведених експериментах змінювалась зі зміною точності наближення ε . Для $\varepsilon = 10^{-5}$ похибка зменшувалась зі збільшенням α , тоді як із зменшенням ε ($\varepsilon = 10^{-20}$, рис. 6.19) характер залежності змінювався і похибка починала зростати зі збільшенням α . Оцінка (6.44) точно відображає другий сценарій, але була принципово неточною при $\varepsilon = 10^{-5}$. Це можна пояснити тим, що в такій ситуації припущення $\max_{j=1, \dots, m-1} K'_j \gg 1$, яке використовується при виведенні оцінки, не виконується.

Збільшення похибки наближення матриці T призвело до збільшення похибки розв'язання тестової задачі для фіксованого $\varepsilon = 10^{-5}$ (рис. 6.20). У випадку, коли значення $\varepsilon(m)$ змінюється відповідно до (6.46) для $\varepsilon_0 = \varepsilon(256) = 10^{-5}$ і $d = 2$ точність розв'язання була близькою до точності, отриманої за алгоритмом 1.



Рисунок 6.19 – Залежність від α логарифму похибки наближення матриці T та його оцінки для $g(x) = x$, $m = 256$, $\varepsilon = 10^{-20}$

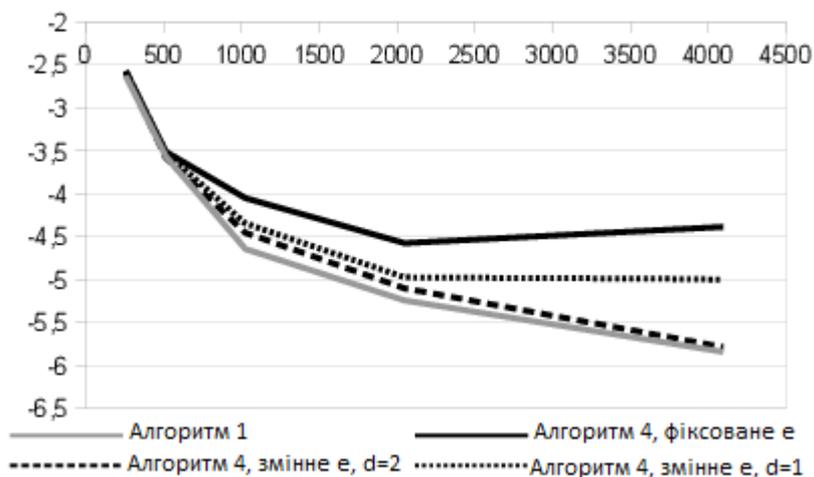


Рисунок 6.20 – Логарифм похибки розв'язання тестової задачі в залежності від m для $\alpha = 0,7$; $g(x) = x$

Середньоквадратична похибка наближення матриці T збільшувалась у випадку фіксованого $\varepsilon = 10^{-5}$, незначним чином збільшувалась для $d = 1$, і залишалась близькою до постійної при $d = 2$, що відповідає теоретичним очікуванням.

Час виконання тут, очевидно, збільшувався, але алгоритм 4 залишався швидшим, ніж алгоритм 1 для $m > 2000$.

6.4. Обчислювальні схеми для рівняння дифузії з ψ -похідною Капуто за часовою та просторовими змінними

Дослідимо поєднання обчислювальних схем для рівняння дифузії з ψ -похідними Капуто за часовою (п. 6.1, 6.2) та просторовими змінними (п. 6.3) з локально-одновимірними схемами розщеплення з позиції балансу їх точності та швидкодії при виконанні без використання додаткових підходів до розпаралелення обчислень.

Будемо розглядати тривимірне рівняння дифузії, отримане на основі дробово-диференціальних узагальнень законів збереження маси [227] та Фіка [176], що має вигляд [40]

$$D_{t,g}^{(\alpha)} C = D_{x,g}^{(\beta)} \left(d \frac{\partial C}{\partial x} \right) + D_{y,g}^{(\beta)} \left(d \frac{\partial C}{\partial y} \right) + D_{z,g}^{(\beta)} \left(d \frac{\partial C}{\partial z} \right) - S, \quad (6.48)$$

де $C(x, y, z, t)$ – концентрація речовини, $d(x, y, z)$ – коефіцієнт дифузії, $S(x, y, z, t)$ – функція джерел, $0 < x \leq L_x$, $0 < y \leq L_y$, $0 < z \leq L_z$, $t \geq 0$, $0 < \alpha \leq 1$, $0 < \beta \leq 1$, $d(x, y, z) > 0$, $D_{t,g}^{(\alpha)}$ – ψ -похідна Капуто [11] за часовою змінною, $D_{x,g}^{(\beta)}$, $D_{y,g}^{(\beta)}$, $D_{z,g}^{(\beta)}$ – аналогічні лівосторонні похідні за просторовими змінними x , y , z , $g(\bullet)$ – задана функція-параметр.

До рівняння (6.48) поставимо початкову умову у вигляді $C(x, y, z, 0) = C_0(x, y, z)$. При $x = y = z = 0$ будемо розглядати крайові умови першого роду $C(0, y, z, t) = f_1^{(x)}(y, z, t)$ (аналогічно і для інших змінних), де $f_1^{(x/y/z)}$ – задані функції. При $x = L_x, y = L_y, z = L_z$, будемо розглядати випадки крайових умов як першого, так і другого родів: $C(L_x, y, z, t) = f_2^{(x)}(y, z, t)$, або $\left. \frac{\partial C}{\partial x} \right|_{x=L_x} = f_3^{(x)}(y, z, t)$ (аналогічно і для інших змінних), де $f_{2/3}^{(x/y/z)}$ – задані функції.

Встановимо в області $\Omega = (0, L_x) \times (0, L_y) \times (0, L_z) \times (0, +\infty)$ наступні обмеження на вихідні дані та розв'язки: будемо припускати, що існує єдиний розв'язок C

рівняння (6.48), що належить до простору C^3 разом з функціями початкових та крайових умов, $d \in C^3$, $S \in C^2$. Припустимо також, що всі функції та їх похідні, що використані для подальшої оцінки точності чисельних методів, є обмеженими в області визначення.

Введемо до розгляду рівномірну скінченно-різницеву сітку

$$\omega = \left\{ (x_{i_1}, y_{i_2}, z_{i_3}, t_j) : i_1 = \overline{0, n_1}, i_2 = \overline{0, n_2}, i_3 = \overline{0, n_3}, j = 0, 1, 2, \dots \right\}, \quad (6.49)$$

$$x_{i_1} = i_1 h_1, x_{i_2} = i_2 h_2, x_{i_3} = i_3 h_3, t_j = j \tau,$$

$$h_1 = L_x / n_1, h_2 = L_y / n_2, h_3 = L_z / n_3,$$

$$C_{i_1 i_2 i_3}^{(j)} = C(x_{i_1}, y_{i_2}, z_{i_3}, t_j), d_{i_1 i_2 i_3}^{(j)} = d(x_{i_1}, y_{i_2}, z_{i_3}, t_j), S_{i_1 i_2 i_3}^{(j)} = S(x_{i_1}, y_{i_2}, z_{i_3}, t_j).$$

Скінченно-різницеві аналоги $\Delta_x^{(\beta)}$, $\Delta_y^{(\beta)}$, $\Delta_z^{(\beta)}$ операторів $D_{x,g}^{(\beta)}$, $D_{y,g}^{(\beta)}$, $D_{z,g}^{(\beta)}$, які можуть бути отримані, аналогічно, наприклад, [167], будемо розглядати у формі (6.24).

У випадку похідної $D_{t,g}^{(\alpha)}$ за часом, маємо аналогічно (6.3)

$$\Delta_t^{(\alpha)} C \Big|_{x=x_k, y=y_l, z=z_m, t=t_j} = \frac{1}{\Gamma(1-\alpha)} \sum_{s=0}^{j-1} \bar{\gamma}_s^{(j)} \frac{C_{klm}^{(s+1)} - C_{klm}^{(s)}}{\tau}, \quad (6.50)$$

$$\bar{\gamma}_s^{(i)} = \int_{t_s}^{t_{s+1}} (g(t_i) - g(\tau))^{-\alpha} d\tau.$$

Для різницевого оператора (6.50), використаємо найпростішу оцінку, яку можна отримати з того факту, що [313] $\left| \frac{C^{(s+1)} - C^{(s)}}{\tau} - \frac{\partial C}{\partial t} \right| \leq O(\tau)$. Тоді, припускаючи що інтеграл $\int_0^t (g(t) - g(\xi))^{-\alpha} d\xi$ обмежений, маємо [317] $|D_t^{(\alpha)} C - \Delta_t^{(\alpha)} C| \leq O(\tau)$. Таку саму оцінку стосовно довжин кроків за просторовими змінними аналогічним чином можна отримати для $D_{x,g}^{(\beta)}$, $D_{y,g}^{(\beta)}$, $D_{z,g}^{(\beta)}$.

Аналогічно описаному у п. 6.3, член $A_x C = D_{x,g}^{(\beta)} \left(d \frac{\partial C}{\partial x} \right)$ на j -му кроці за часом

дискретизується аналогічно (6.25) як

$$\begin{aligned}
A_x C \Big|_{x=x_k, y=y_l, z=z_m, t=t_j} &\approx \tilde{L}_x C \Big|_{x=x_k, y=y_l, z=z_m, t=t_j} = \\
&= \frac{1}{h_1^2 \Gamma(1-\beta)} \sum_{s=0}^{k-1} \gamma_s^{(k)} \left(d_{slm}^{(j)} C_{s+1,l,m}^{(j)} - \left(d_{slm}^{(j)} + d_{s-1,l,m}^{(j)} \right) C_{slm}^{(j)} + d_{s-1,l,m}^{(j)} C_{s-1,l,m}^{(j)} \right),
\end{aligned}$$

Оскільки тут застосовується центральна різницева похідна другого порядку, дотримуючись логіки, використаної для (6.50), маємо верхню оцінку $|A_x C - \tilde{L}_x C| < O(h_1^2)$.

У матричній формі, вважаючи, що комірка сітки $(x_{i_1}, y_{i_2}, z_{i_3})$ відповідає рядку

$r_{i_1 i_2 i_3} = i_1 n_2 n_3 + i_2 n_3 + i_3$, $D_{x,g}^{(\beta)} \left(d \frac{\partial C}{\partial x} \right)$ можна представити у вигляді

$$D_{x,g}^{(\beta)} \left(d \frac{\partial C}{\partial x} \right) \Big|_{t=t_j} \approx \frac{1}{h_1^2 \Gamma(1-\beta)} (T_x + D_x) M_x^{(j)} \vec{C}^{(j)},$$

$$\dim M_x^{(j)} = \dim T_x = \dim D_x = n_1 n_2 n_3 \times n_1 n_2 n_3, \dim \vec{C} = 1 \times n_1 n_2 n_3,$$

$$\vec{C}^{(j)} \left[r_{i_1 i_2 i_3} \right] = C_{i_1 i_2 i_3}^{(j)}, \quad (6.51)$$

$$M_x^{(j)} = \sum_{l=1}^{n_2-1} \sum_{m=1}^{n_3-1} \bar{M}_{xlm}^{(j)}, \dim \bar{M}_{xlm}^{(j)} = n_1 n_2 n_3 \times n_1 n_2 n_3,$$

$$\bar{M}_{xlm}^{(j)} \left[r_{i_1 lm} \right] \left[r_{i_1 lm} \right] = d_{i_1 lm}^{(j)} + d_{i_1-1, l, m}^{(j)},$$

$$\bar{M}_{xlm}^{(j)} \left[r_{i_1-1, l, m} \right] \left[r_{i_1 lm} \right] = d_{i_1-1, l, m}^{(j)}, \bar{M}_{xlm}^{(j)} \left[r_{i_1+1, l, m} \right] \left[r_{i_1 lm} \right] = d_{i_1 lm}^{(j)},$$

$\bar{M}_{xlm}^{(j)} \left[r_{i_1 i_2 i_3} \right] \left[r_{i_4 i_5 i_6} \right] = 0$ в інших випадках,

$$T_x = \sum_{l=1}^{n_2-1} \sum_{m=1}^{n_3-1} \bar{T}_{xlm}, \dim \bar{T}_{xlm} = n_1 n_2 n_3 \times n_1 n_2 n_3,$$

$\bar{T}_{xlm} \left[r_{slm} \right] \left[r_{klm} \right] = \gamma_s^{(k)}$, $s < k$, $\bar{T}_{xlm} \left[r_{i_1 i_2 i_3} \right] \left[r_{i_4 i_5 i_6} \right] = 0$ в інших випадках,

$$D_x = \sum_{l=1}^{n_2-1} \sum_{m=1}^{n_3-1} \bar{D}_{xlm}, \dim \bar{D}_{xlm} = n_1 n_2 n_3 \times n_1 n_2 n_3,$$

$\bar{D}_{xlm} \left[r_{klm} \right] \left[r_{klm} \right] = \gamma_{k-1}^{(k)}$, $\bar{D}_{xlm} \left[r_{i_1 i_2 i_3} \right] \left[r_{i_4 i_5 i_6} \right] = 0$ в інших випадках,

де $\vec{C}^{(j)}[a]$ – елемент a вектора $\vec{C}^{(j)}$; $\bar{M}_{xlm}^{(j)}[a,b]$, $\bar{T}_{xlm}[a,b]$, $\bar{D}_{xlm}[a,b]$ – елементи у стовпці a і рядку b матриць $\bar{M}_{xlm}^{(j)}$, \bar{T}_{xlm} , \bar{D}_{xlm} . Надалі будемо використовувати такі ж самі позначення для елементів інших векторів та матриць.

У випадку $g(t) = t$ (класична похідна Капуто), матриці, сформовані з непорожніх рядків і стовпців \bar{T}_{xlm} є матрицями Теплиця.

Інший варіант представлення $D_{x,g}^{(\beta)} \left(d \frac{\partial C}{\partial x} \right)$ можна, аналогічно (6.26), отримати, використовуючи розв'язки задачі на попередньому кроці за часом:

$$\begin{aligned} D_{x,g}^{(\beta)} \left(d \frac{\partial C}{\partial x} \right) \Big|_{t=t_j} &\approx \bar{L}_x C \Big|_{t=t_j} = \\ &= \frac{1}{h_1^2 \Gamma(1-\beta)} \left(T_x M_x^{(j)} \vec{C}^{(j-1)} + D_x M_x^{(j)} \vec{C}^{(j)} \right). \end{aligned} \quad (6.52)$$

Оскільки $\frac{\partial}{\partial x} \left(d \frac{\partial C}{\partial x} \right) \Big|_{t=t_{j-1}} = \frac{\partial}{\partial x} \left(d \frac{\partial C}{\partial x} \right) \Big|_{t=t_j} + O(\tau)$, похибка наближення (6.52) має

верхню оцінку порядку $O(\tau + h_1^2)$.

Наближення $\bar{L}_y C \approx \tilde{L}_y C \approx A_y C = D_{y,g}^{(\beta)} \left(d \frac{\partial C}{\partial y} \right)$ і $\bar{L}_z C \approx \tilde{L}_z C \approx A_z C = D_{z,g}^{(\beta)} \left(d \frac{\partial C}{\partial z} \right)$

будуються з використанням аналогічно визначених матриць $M_y^{(j)}$, T_y , D_y , $M_z^{(j)}$, T_z , D_z .

Неявна скінченно-різницева схема (схема 1) для рівняння (6.48) на кроці j отримується за допомогою (6.50),(6.51) у вигляді наступної СЛАР:

$$A^{(j)} \vec{C}^{(j)} = \Omega_1^{(j)} + \Omega_2^{(j)}, A^{(j)} = A_1^{(j)} - A_2^{(j)} + A_3^{(j)}, \quad (6.53)$$

$$\begin{aligned} \dim A^{(j)} &= \dim A_1^{(j)} = \dim A_2^{(j)} = \dim A_3^{(j)} = n_1 n_2 n_3 \times n_1 n_2 n_3, \\ \dim \Omega_1^{(j)} &= 1 \times n_1 n_2 n_3, \dim \Omega_2^{(j)} = 1 \times n_1 n_2 n_3. \end{aligned}$$

У (6.53) матриця $A_1^{(j)}$ і вектор $\Omega_1^{(j)}$ відповідають дискретизації похідної за часом та функції джерел:

$$A_1^{(j)} = \text{diag} \left(\frac{\bar{\gamma}_{j-1}^{(j)}}{\tau \Gamma(1-\alpha)} \right),$$

$$\Omega_1^{(j)} [r_{i_1 i_2 i_3}] = \frac{1}{\Gamma(1-\alpha)} \left(\frac{\bar{\gamma}_{j-1}^{(j)} C_{i_1 i_2 i_3}^{(j-1)}}{\tau} - \sum_{s=0}^{j-2} \bar{\gamma}_s^{(j)} \frac{C_{i_1 i_2 i_3}^{(s+1)} - C_{i_1 i_2 i_3}^{(s)}}{\tau} \right) - S_{i_1 i_2 i_3}^{(j)};$$

матриця $A_2^{(j)}$ – дискретизації правої частини рівняння (6.48) без функції джерел:

$$A_2^{(j)} = \frac{1}{\Gamma(1-\beta)} \left[\frac{1}{h_1^2} (T_x + D_x) M_x^{(j)} + \frac{1}{h_2^2} (T_y + D_y) M_y^{(j)} + \frac{1}{h_3^2} (T_z + D_z) M_z^{(j)} \right];$$

матриця $A_3^{(j)}$ і вектор $\Omega_2^{(j)}$ – дискретизації крайових умов:

$$A_3^{(j)} = \sum_{l=1}^{n_2-1} \sum_{m=1}^{n_3-1} \bar{A}_{3xlm}^{(j)} + \sum_{l=1}^{n_1-1} \sum_{m=1}^{n_3-1} \bar{A}_{3ylm}^{(j)} + \sum_{l=1}^{n_1-1} \sum_{m=1}^{n_2-1} \bar{A}_{3zlm}^{(j)},$$

$$\dim \bar{A}_{3xlm}^{(j)} = \dim \bar{A}_{3ylm}^{(j)} = \dim \bar{A}_{3zlm}^{(j)} = n_1 n_2 n_3 \times n_1 n_2 n_3,$$

$$\Omega_2^{(j)} = \sum_{l=1}^{n_2-1} \sum_{m=1}^{n_3-1} \bar{\Omega}_{2xlm}^{(j)} + \sum_{l=1}^{n_1-1} \sum_{m=1}^{n_3-1} \bar{\Omega}_{2ylm}^{(j)} + \sum_{l=1}^{n_1-1} \sum_{m=1}^{n_2-1} \bar{\Omega}_{2zlm}^{(j)},$$

$$\dim \bar{\Omega}_{2xlm}^{(j)} = \dim \bar{\Omega}_{2ylm}^{(j)} = \dim \bar{\Omega}_{2zlm}^{(j)} = 1 \times n_1 n_2 n_3.$$

Для крайової умови при $x=0$ $\bar{A}_{3xlm}^{(j)} [r_{0lm}] [r_{0lm}] = 1$, $\bar{\Omega}_{2xlm}^{(j)} [r_{0lm}] = C_x(y_l, z_m, t_{j+1})$.

Для крайової умови першого роду при $x=L_x$ $\bar{A}_{3xlm}^{(j)} [r_{n_1, l, m}] [r_{n_1, l, m}] = 1$,

$\bar{\Omega}_{2xlm}^{(j)} [r_{n_1, l, m}] = C_\xi(y_{i_2}, z_{i_3}, t_{j+1})$. Для крайової умови другого роду при $x=L_x$

$\bar{A}_{3xlm}^{(j)} [r_{n_1, l, m}] [r_{n_1, l, m}] = 1$, $\bar{A}_{3xlm}^{(j)} [r_{n_1-1, l, m}] [r_{n_1, l, m}] = -1$, $\bar{\Omega}_{2xlm}^{(j)} [r_{n_1, l, m}] = h_1 \bar{C}_\xi(y_{i_2}, z_{i_3}, t_{j+1})$. В

інших випадках елементи $\bar{A}_{3xlm}^{(j)}$ і $\bar{\Omega}_{2xlm}^{(j)}$ дорівнюють 0. Елементи $\bar{A}_{3ylm}^{(j)}$, $\bar{A}_{3zlm}^{(j)}$, $\bar{\Omega}_{2ylm}^{(j)}$,

$\bar{\Omega}_{2zlm}^{(j)}$ визначаються аналогічним чином.

Початкові умови дискретизуються у вигляді $C_{i_1 i_2 i_3}^{(0)} = C_0(x_{i_1}, y_{i_2}, z_{i_3})$.

Твердження 6.2. Похибка апроксимації рівняння (6.48) за допомогою схеми (6.53) має верхню оцінку порядку $O(\tau + h_1^2 + h_2^2 + h_3^2)$.

Схеми розщеплення

Дробово-диференціальне рівняння (6.48) на інтервалі $t \in [t_j, t_{j+1}]$ може бути апроксимоване за допомогою (6.50) у вигляді

$$k_1 \frac{\partial C}{\partial t} = (A_x + A_y + A_z)C - S_1 + O(\tau), \quad k_1 = \frac{\bar{\gamma}_j^{(j+1)}}{\Gamma(1-\alpha)}, \quad (6.54)$$

$$S_1 = S + \frac{1}{\Gamma(1-\alpha)} \int_0^{t_j} \frac{\partial C}{\partial t} (g(t_{j+1}) - g(\tau))^{-\alpha} d\tau.$$

Апроксимуючи надалі $\frac{\partial C}{\partial t}$ отримуємо

$$\left(E - \frac{\tau}{k_1} A_x - \frac{\tau}{k_1} A_y - \frac{\tau}{k_1} A_z \right) C(x, y, z, t_{j+1}) = C(x, y, z, t_j) - \frac{\tau}{k_1} S_1 + O\left(\frac{\tau^2}{k_1}\right). \quad (6.55)$$

Розглядаючи три взаємопов'язані задачі

$$\begin{aligned} \left(E - \frac{\tau}{k_1} A_x \right) C_{1/3} &= C^{(j)} - \frac{\tau}{3k_1} S_1, \quad C_{1/3}^{(j)} = C^{(j)} \\ \left(E - \frac{\tau}{k_1} A_y \right) C_{2/3} &= C_{1/3} - \frac{\tau}{3k_1} S_1, \quad C_{2/3}^{(j)} = C_{1/3}^{(j+1)} \\ \left(E - \frac{\tau}{k_1} A_z \right) C_1 &= C_{2/3} - \frac{\tau}{3k_1} S_1, \quad C_1^{(j)} = C_{2/3}^{(j+1)}, \end{aligned} \quad (6.56)$$

де $C^{1/3}$, $C^{2/3}$ – розв'язки (6.48) на проміжних кроках, $C^{(j+1)} = C_1^{(j+1)}$, і підставляючи перше та друге рівняння у третє, отримуємо схему, яка відповідає рівнянню (6.55):

$$\begin{aligned} \left(E - \frac{\tau}{k_1} A_x - \frac{\tau}{k_1} A_y - \frac{\tau}{k_1} A_z \right) C(x, y, z, t_{j+1}) &= C(x, y, z, t_j) - \frac{\tau}{k_1} S_1 + R \\ R &= \left(\frac{\tau^3}{k_1^3} A_z A_y A_x - \frac{\tau^2}{k_1^2} (A_y A_x + A_z A_x + A_z A_y) \right) C(x, y, z, t_{j+1}) + \\ &+ \frac{\tau^2}{3k_1^2} A_y S_1 + \frac{2\tau^2}{3k_1^2} A_z S_1 - \frac{\tau^3}{3k_1^3} A_z A_y S_1. \end{aligned}$$

Для першого рівняння системи (6.56) (і, аналогічно, для двох інших) ставимо граничні умови

$$C_{1/3}(0, y, z, t_{j+1}) = f_1^{(x)}(x, y, t_{j+1}), \quad C_{1/3}(L_x, y, z, t_{j+1}) = f_2^{(x)}(y, z, t_{j+1}),$$

$$\text{або } \left. \frac{\partial C_{1/3}}{\partial x} \right|_{x=L_x} = f_3^{(x)}(y, z, t_{j+1}).$$

Дискретизуючи (6.56) за просторовими змінними та дискретизуючи S_1 використовуючи (6.50), маємо наступну послідовну обчислювальну схему:

$$\begin{aligned} \left(E - \frac{\tau}{k_1} \tilde{L}_x \right) C_{1/3} &= C^{(j)} + \frac{\tau}{3k_1} S_2, \\ \left(E - \frac{\tau}{k_1} \tilde{L}_y \right) C_{2/3} &= C_{1/3} + \frac{\tau}{3k_1} S_2, \\ \left(E - \frac{\tau}{k_1} \tilde{L}_z \right) C_1 &= C_{2/3} + \frac{\tau}{3k_1} S_2, \\ S_2 &= S + \frac{1}{\Gamma(1-\alpha)} \sum_{s=0}^{j-2} \tilde{\gamma}_s^{(j)} \frac{C^{(s+1)} - C^{(s)}}{\tau}. \end{aligned} \quad (6.57)$$

Теорема 6.3. За умови, що виконуються нерівності $|A_y A_x C| < M$, $|A_z A_x C| < M$, $|A_z A_y C| < M$, $|A_z A_y A_x C| < M$, $|A_y S_1| < M$, $|A_z S_1| < M$, $|A_z A_y S_1| < M$, де M – константа, для похибки апроксимації рівняння (6.54) системою (6.57) справедлива оцінка $\varepsilon_1 \leq O(\tau^{\min(2\alpha, 2-\alpha)} + h_1^2 + h_2^2 + h_3^2)$.

Доведення.

Позначивши як \bar{C} розв'язок рівняння (6.54), а як $\hat{C}_{i_1 i_2 i_3}^{(j)} = \bar{C}_{i_1 i_2 i_3}^{(j)} + u_{i_1 i_2 i_3}^{(j)}$ – розв'язок системи (6.57), та віднімаючи (6.57) з (6.54), отримуємо наступне рівняння для похибки усічення:

$$\begin{aligned} k_1 \frac{u_{i_1 i_2 i_3}^{(j+1)} - u_{i_1 i_2 i_3}^{(j)}}{\tau} &= (\tilde{L}_x + \tilde{L}_y + \tilde{L}_z) u \Big|_{x=x_{i_1}, y=y_{i_2}, z=z_{i_3}} + \varepsilon_{1i_1 i_2 i_3}^{(j)}, \\ u_{0i_2 i_3}^{(j)} &= u_{i_1 0i_3}^{(j)} = u_{i_1 i_2 0}^{(j)} = u_{n_1-1, i_2, i_3}^{(j)} = u_{i_1, n_2-1, i_3}^{(j)} = u_{i_1, i_2, n_3-1}^{(j)} = 0, u^{(0)} = 0, \\ \varepsilon_{1i_1 i_2 i_3}^{(j)} &= -k_1 \left(\frac{\partial C}{\partial t} - \frac{C^{(j+1)} - C^{(j)}}{\tau} \right) \Big|_{x=x_{i_1}, y=y_{i_2}, z=z_{i_3}, t=t_j} + \\ &+ ((A_x - \tilde{L}_x + A_y - \tilde{L}_y + A_z - \tilde{L}_z) C + R) \Big|_{x=x_{i_1}, y=y_{i_2}, z=z_{i_3}, t=t_j}. \end{aligned}$$

Щоб оцінити похибку схеми (6.57), спочатку необхідно оцінити порядок $\bar{\gamma}_j^{(j+1)}$.

Для цього, подібно до [42], виконуємо заміну змінних $z = g(t_{j+1}) - g(\tau)$ і розкладаємо $f'(z)$ у ряд Тейлора в точці $z = g(t_{j+1})$. Оскільки ряди, отримувані після такого перетворення, є знакопереміжними, отримуємо оцінку

$$\bar{\gamma}_j^{(j+1)} \leq \frac{f'(g(t_{j+1}))}{1-\alpha} (g(t_{j+1}) - g(t_{j+1} - \tau))^{1-\alpha}.$$

Оскільки $g(t_{j+1}) - g(t_{j+1} - \tau) = \tau g'(t_{j+1} - \tau) + O(\tau^2)$, для $\tau \rightarrow 0$ маємо

$$\bar{\gamma}_j^{(j+1)} \leq \frac{f'(g(t_{j+1})) (g'(t_{j+1} - \tau))^{1-\alpha}}{\Gamma(1-\alpha)} \tau^{1-\alpha} = O(\tau^{1-\alpha}),$$

$$\frac{\tau}{k_1} \leq O(\tau^\alpha).$$

Тоді, використовуючи умови щодо обмеженості операторів, маємо $R \leq O(\tau^{2\alpha})$

та наступну верхню оцінку похибки апроксимації рівняння (6.54) системою (6.57):

$$\varepsilon_1 \leq O(\tau^{\min(2\alpha, 2-\alpha)} + h_1^2 + h_2^2 + h_3^2). \quad (6.58)$$

□

Загальна помилка апроксимації рівняння (6.48) системою (6.57) на інтервалі $t \in [t_j, t_{j+1}]$ матиме верхню оцінку $O(\tau^{\min(1, 2\alpha)} + h_1^2 + h_2^2 + h_3^2)$.

Зазначимо, що система (6.57) (далі – схема 2а) являє собою ряд незалежних нижньотрикутних СЛАР. Систему, розв'язок якої має меншу обчислювальну складність (схема 2б), можна отримати за допомогою наближення A_x у формі \bar{L}_x згідно з (6.52). Схема 2б має вигляд ряду незалежних тридіагональних СЛАР. Верхня оцінка похибки усічення в цьому випадку залишається незмінною.

Окрім цього, розглянемо схему, основу на розщеплені рівняння (6.48) на диференціальному рівні.

Представимо рівняння (6.48) у вигляді

$$\begin{aligned}
(E - I_t^{(\alpha)} A_x - I_t^{(\alpha)} A_y - I_t^{(\alpha)} A_z) C &= I_t^{(\alpha)} S, \\
(E - I_t^{(\alpha)} A_x)(E - I_t^{(\alpha)} A_y)(E - I_t^{(\alpha)} A_z) C &= I_t^{(\alpha)} S + R_1, \\
R_1 &= (I_t^{(\alpha)} A_z I_t^{(\alpha)} A_y I_t^{(\alpha)} A_z - I_t^{(\alpha)} A_x I_t^{(\alpha)} A_y + I_t^{(\alpha)} A_x I_t^{(\alpha)} A_z + I_t^{(\alpha)} A_y I_t^{(\alpha)} A_z) C,
\end{aligned} \tag{6.59}$$

де $I_t^{(\alpha)} \bar{f}(t) = \frac{1}{\Gamma(\alpha)} \int_0^t g'(\tau) (g(t) - g(\tau))^{\alpha-1} \bar{f}(\tau) d\tau$ – оператор дробового інтегрування [11], обернений до ψ -похідної Капуто за часовою змінною.

Розрахунки згідно з (6.59) на інтервалі $t \in [t_j, t_{j+1}]$ можуть бути організовані шляхом послідовного розв'язання задач

$$\begin{aligned}
(E - I_t^{(\alpha)} A_x) C_{1/3} &= I_t^{(\alpha)} S \Rightarrow D_t^{(\alpha)} C_{1/3} = A_x C_{1/3} + S, \\
(E - I_t^{(\alpha)} A_y) C_{2/3} &= C_{1/3} \Rightarrow D_t^{(\alpha)} C_{2/3} = A_y C_{2/3} + D_t^{(\alpha)} C_{2/3}, \\
(E - I_t^{(\alpha)} A_z) C_1 &= C_{2/3} \Rightarrow D_t^{(\alpha)} C_1 = A_z C_1 + D_t^{(\alpha)} C_{2/3}.
\end{aligned} \tag{6.60}$$

До системи (6.60) ставимо ті самі початкові та крайові умови, що і для системи (6.56).

Дискретизуючи (6.60) за часовою змінною та враховуючи що

$$\Delta_t^{(\alpha)} (C_{2/3} - C_{1/3}) = \frac{k_1}{\tau} (C_{2/3} - C_{1/3}), \quad \Delta_t^{(\alpha)} (C_1 - C_{2/3}) = \frac{k_1}{\tau} (C_1 - C_{2/3}) \text{ отримуємо}$$

$$\begin{aligned}
\left(E - \frac{\tau}{k_1} A_x\right) C_{1/3} &= C^{(j)} + \frac{\tau}{k_1} S_1, \\
\left(E - \frac{\tau}{k_1} A_y\right) C_{2/3} &= C_{1/3}, \\
\left(E - \frac{\tau}{k_1} A_z\right) C_1 &= C_{2/3}.
\end{aligned} \tag{6.61}$$

Підставляючи рівняння системи (6.61) одне в одне, маємо, як і для попередньої схеми,

$$\begin{aligned}
\left(E - \frac{\tau}{k_1} A_x - \frac{\tau}{k_1} A_y - \frac{\tau}{k_1} A_z\right) C(x, y, z, t_{j+1}) &= C(x, y, z, t_j) - \frac{\tau}{k_1} S_1 + R_2, \\
R_2 &= \left(\frac{\tau^3}{k_1^3} A_z A_y A_x - \frac{\tau^2}{k_1^2} (A_y A_x + A_z A_x + A_z A_y)\right) C(x, y, z, t_{j+1}).
\end{aligned}$$

Звідси обчислювальну схему 3a отримуємо з (6.61) дискретизуючи A_x як \tilde{L}_x . Подібно до (6.57), дискретизуючи A_x у формі \bar{L}_x згідно з (6.52) отримуємо "явну" схему 3b. Оскільки залишковий член R_2 тут має той самий порядок, що і R_1 і використовуються ті ж самі скінченно-різницеві аналоги операторів дробового диференціювання, схеми 3a та 3b мають ті самі верхні оцінки похибки усічення, що і схеми 2a та 2b.

Процедури оптимізації розрахунків

Для зменшення складності розрахунків при дискретизації дробових похідних за просторовими змінними, будемо використовувати підхід [39, 46], заснований на розкладенні ядра інтегрального оператора в ряди, описаний у п. 6.3. Використання відповідної процедури не впливає на порядок точності скінченно-різницевих схем у випадку виконання умови (6.45).

Стосовно похідної за часом, виконуючи обчислення згідно з (6.50), будемо розглядати безпосередній розрахунок відповідних сум (алгоритм 1) та додатково два алгоритми, що зменшують обчислювальну складність. У випадку з алгоритмом 1 час обчислення, як і необхідний обсяг пам'яті, лінійно збільшується зі збільшенням номеру кроку за часом і з певного моменту починає домінувати у загальний час, що витрачається на проведення розрахунків.

У якості алгоритму 2, аналогічно описаному у п. 6.2, будемо розглядати алгоритм принципу фіксованої пам'яті у якому підсумовування виконується доти, доки виконується умова $\bar{\gamma}_s^{(j+1)} > \varepsilon_3$, де ε_3 – задана стала. У цьому випадку час розв'язання задачі та необхідний обсяг пам'яті є близьким до постійного і незначним чином змінюється, починаючи з певного кроку за часом.

Теорема 6.4. Для похибки обчислення значення дробової ψ -похідної Капуто на кроці за часом $j+1$ при застосуванні алгоритму 2 за умов, що функція $g'(t)$ обмежена та $\forall s \in [0, L], \exists M : g(t_s + \tau) - g(t_s) = \tau M$, справедлива оцінка

$$\varepsilon_4 \leq j\tau^{1-\alpha} - O\left(\varepsilon_3^{-1/\alpha} \tau^{\frac{1-\alpha^2}{\alpha}}\right).$$

Доведення.

Похибка обчислення значення дробової похідної, спричинена застосуванням алгоритму 2, дорівнює

$$\varepsilon_4 = \frac{1}{\Gamma(1-\alpha)} \sum_{s=0}^L \int_{t_s}^{t_{s+1}} \frac{\partial C}{\partial t} \left(g(t_{j+1}) - g(\tau) \right)^{-\alpha} d\tau,$$

де L – найбільший член суми, для якої $\bar{\gamma}_L^{(j+1)} \leq \varepsilon_3$.

Проводячи у $\bar{\gamma}_s^{(j+1)}$, подібно до [42], заміну змінних $z = g(t_{j+1}) - g(\tau)$, розкладаючи $f'(z)$ у ряд Тейлора в точці $z = g(t_{j+1})$, та враховуючи, що ряди, отримані після такого перетворення, є знакопереміжними, маємо

$$\begin{aligned} \bar{\gamma}_s^{(j+1)} &\leq \left| \frac{f'(g(t_{j+1}))}{\Gamma(1-\alpha)} \left[\left(g(t_{j+1}) - g(t_s + \tau) \right)^{1-\alpha} - \left(g(t_{j+1}) - g(t_s) \right)^{1-\alpha} \right] \right| \leq \\ &\leq \left| \frac{f'(g(t_{j+1}))}{\Gamma(1-\alpha)} \left(g(t_{j+1}) - g(t_s + \tau) \right)^{1-\alpha} \left[1 - \left(1 + \frac{g(t_{j+\tau}) - g(t_s)}{g(t_{j+1}) - g(t_s + \tau)} \right)^{1-\alpha} \right] \right|. \end{aligned}$$

Оскільки $(1+x)^r < 1+rx$ при $x \geq -1, r \in (0,1)$, маємо

$$\bar{\gamma}_s^{(j+1)} \leq \left| \frac{f'(g(t_{j+1}))}{\Gamma(1-\alpha)} \left(g(t_{j+1}) - g(t_s + \tau) \right)^{-\alpha} \left(g(t_s + \tau) - g(t_s) \right) \right|.$$

Розкладаючи $g(t)$ у ряд Тейлора в точці $t = t_{j+1}$, отримуємо $g(t_s + \tau) = g(t_{j+1}) + (t_s - t_{j+1} + \tau)g'(t_{j+1}) + O(\tau^2)$. Припускаючи, що функція $g'(t)$ обмежена та існує стала M , така що $g(t_s + \tau) - g(t_s) = \tau M$, отримуємо

$$\bar{\gamma}_s^{(j+1)} \leq \left| \frac{f'(g(t_{j+1}))}{\Gamma(1-\alpha)} \tau^{1-\alpha} (j-s)^{-\alpha} \bar{M} \right| \leq O(\tau^{1-\alpha}), \bar{M} = M \cdot g(t_{j+1})^{-\alpha}. \quad (6.62)$$

Надалі, за припущення, що похідна $\frac{\partial C}{\partial t}$ є сталою на інтервалі $[t_s, t_{s+1}]$, похибка матиме верхню оцінку $\varepsilon_4 \leq O(L\tau^{1-\alpha})$, що залежить від кількості L членів ряду.

Щоб отримати оцінку, яка залежить від обмеження ε_3 на значення члену ряду, використовуючи визначення L як $\bar{\gamma}_L^{(j)} = \varepsilon_3$, з (6.62) маємо

$$L \leq j - \left(\frac{\varepsilon_3}{\tau^{1-\alpha} M} \right)^{-1/\alpha} \Rightarrow \varepsilon_4 \leq j\tau^{1-\alpha} - O\left(\varepsilon_3^{-1/\alpha} \tau^{\frac{1-\alpha^2}{\alpha}} \right). \quad (6.63)$$

□

Алгоритм 3 є аналогом схеми (6.34) для ψ -похідної за часовою змінною для випадку $K_d = 0$ і фіксованої кількості K_t членів ряду. Час, що витрачається на обчислення, та необхідний обсяг пам'яті для алгоритму 3 має порядок $O(K_t)$.

Помилка наближення $\bar{y}_j^{(i)}$ тут має [42] порядок $\varepsilon_5 \leq O\left(I_{K_t} K_t^{1-\alpha} (K_t + 1)^\alpha \right)$, де I_{K_t} – значення K_t -го члену ряду (6.30). За припущення, що $f'(t)$ обмежена на

інтервалі $[t_j, t_{j+1}]$, маємо $I_{K_t} \leq O\left(\frac{1}{K_t + 1} \binom{-\alpha}{K_t} \right)$. Наближаючи $\binom{-\alpha}{K_t}$ у вигляді

$\binom{-\alpha}{K_t} \approx \frac{1}{\Gamma(\alpha) K_t^{1-\alpha}}$, отримуємо $\varepsilon_5 \leq O(K_t^{\alpha-1})$. Загальна верхня оцінка похибки

наближення схеми, відповідно, матиме порядок

$$\varepsilon_6 \leq O(jK_t^{\alpha-1}). \quad (6.64)$$

Алгоритми розв'язання СЛАР

Розв'язання СЛАР у скінченно-різницевих схемах 1, 2а та 3а будемо проводити алгоритмом BiCGstab [219]. Тридіагональні СЛАР у схемах 2b і 3b будемо розв'язувати методом прогонки [313].

При використанні алгоритму BiCGstab будемо використовувати наступні схеми множення матриці $A_2^{(j)}$ на вектори:

1) Схема з повним обчисленням значень елементів матриць на етапі ініціалізації (аналогічно схемі 1 з п. 6.3);

2) Схема з використанням представлення інтегралів у вигляді ряду (аналогічно схемі 4 з п. 6.3);

3) Для випадку $g(x) = x$, схема, аналогічна схемі 3 з п. 6.3.

Обчислювальні експерименти

У якості тестової задачі будемо розглядати рівняння (6.48) при $g(t) = t^z$,
 $d = 1/2 + z$,

$$S = (1 - 2t^2) \left[\frac{2x^2 y^2}{L_x^2 L_y^2 L_z^2} \left(0.5 z^{1-\beta z} \frac{\Gamma(1/\chi + 1)}{\Gamma(1/\chi - \beta + 1)} + z^{2-\beta z} \frac{\Gamma(2/\chi + 1)}{\Gamma(2/\chi - \beta + 1)} \right) + \right. \\
\left. + \frac{2z^2 y^2}{L_x^2 L_y^2 L_z^2} (0.5 + z) x^{1-\beta z} \frac{\Gamma(1/\chi + 1)}{\Gamma(1/\chi - \beta + 1)} + \right. \\
\left. + \frac{2z^2 x^2}{L_x^2 L_y^2 L_z^2} (0.5 + z) y^{1-\beta z} \frac{\Gamma(1/\chi + 1)}{\Gamma(1/\chi - \beta + 1)} \right] - 2t^{2-\alpha z} \frac{x^2 y^2 z^2}{L_x^2 L_y^2 L_z^2} \frac{\Gamma(2/\chi + 1)}{\Gamma(2/\chi - \alpha + 1)}.$$

При $C_0(x, y, z) = \bar{C}(x, y, z, 0)$, $f_1^{(x)}(y, z, t) = \bar{C}(0, y, z, t)$, $f_2^{(x)}(y, z, t) = \bar{C}(L_x, y, z, t)$,

або $f_3^{(x)}(y, z, t) = \left. \frac{\partial \bar{C}}{\partial x} \right|_{x=L_x}$ і, аналогічно, для інших просторових змінних, відповідна

початково-крайова задача матиме розв'язок $\bar{C}(x, y, z, t) = (1 - 2t^2) \frac{x^2 y^2 z^2}{L_x^2 L_y^2 L_z^2}$. Надалі

розглядатимемо випадок $L_x = L_y = L_z = 3$.

У якості оцінки точності будемо використовувати середньоквадратичне відхилення $\bar{\varepsilon}$ чисельного розв'язку від аналітичного. Вихідні коди розробленого програмного забезпечення наведено у [265].

Для оцінки зміни похибки при зміні довжини кроку за часом, тестова задача розв'язувалась на сітці розміру $10 \times 10 \times 10$ комірок при $0,00075 \leq \tau \leq 0,01$ з використанням неоптимізованих схем розрахунку дробових похідних.

Для всіх обчислювальних схем похибка залежала від довжини кроку часу як $\bar{\varepsilon}(\tau) = k_2 \tau^{k_3} - k_1$. Розв'язки, отримані за схемами розщеплення 3а, 3б у випадку крайових умов першого роду, у всіх випадках були менш точними (в середньому у 6 разів для схеми 3а та у 9 разів для схеми 3б), ніж розв'язки, отримані за схемами 2а, 2б. У випадку крайових умов другого роду, розв'язки, отримані за схемою 3а, були на $\sim 30\%$ точнішими, ніж розв'язки, отримані за схемою 2а, але були менш точними (в середньому у 3,5 рази), ніж розв'язки, отримані за схемою 2б. У зв'язку з цим, надалі будемо досліджувати виключно схеми 1, 2а, та 2б.

Коефіцієнт k_3 відповідає за збільшення швидкості збіжності схем розщеплення зі збільшенням порядку α дробової похідної за часовою змінною. Відповідно до теоретичних оцінок, він був менше ніж $2\min(1, 2\alpha)$ у більшості випадків. За вищої точності, досягнутої схемою 2b (у середньому в 2 рази більшої для крайових умов першого роду; до 60 разів більшої для крайових умов другого роду з тенденцією до збільшення при зменшенні α та/або β), значення коефіцієнту k_3 для схем 2a і 2b дещо відрізнялись. У випадку неявної схеми 1 швидкість збіжності була близька до лінійної (середнє значення k_3 дорівнювало 1.1 для крайових умов першого роду та 0.95 для крайових умов другого роду), що узгоджується з її теоретичною верхньою межею. При збільшенні порядку β дробових похідних за просторовими змінними, швидкість збіжності дещо зменшується для схем 2a, 2b (до 13%) і збільшується для схеми 1 (до 10%). Вплив параметра похідної χ на швидкість збіжності був незначним. Вища точність у випадку всіх розглянутих схем була досягнута при його нижчих значеннях (26% різниці між випадками $\chi = 1,5$ та $\chi = 0,5$ для схеми 1, 6% різниці для схеми 2b та 60% різниці для схеми 2a). Вища точність та швидкість збіжності, як і очікувалось, спостерігалась при розв'язанні задачі з крайовими умовами першого роду (в середньому, в 500 разів менша похибка для схеми 1).

З метою оцінки швидкості збіжності схем за просторовими змінними була досліджена зміна похибки розв'язання у випадку $n = n_x = n_y = n_z = [10, \dots, 80]$, $\tau = 0,0025$. Враховуючи, що найточніші розв'язки при застосуванні схем розщеплення, були отримані схемою 2b, при вивченні впливу параметрів дробових похідних на швидкість збіжності порівнювалась лише ця схема та неявна схема 1.

Зміни значення параметра χ в діапазоні $\chi \in [0,5; 1,5]$ та зміни значення параметра α несуттєво впливали на швидкість збіжності. Схема 2b демонструвала порядки точності та швидкості збіжності, порівнянні зі схемою 1 для випадку крайових умов другого роду при $\alpha \geq 0,5$ та для випадку крайових умов першого роду при $\alpha \geq 0,9$. Точність схеми 2b у цих випадках була не більш ніж у 2 рази нижчою при $n = [10, \dots, 80]$.

Збільшення значення параметра β не впливало суттєво на швидкість збіжності для схеми 1. Точність, у цьому випадку, збільшувалась для крайових умов як першого, так і другого роду. Для схеми 2b швидкість збіжності зменшувалась зі збільшенням β . При достатньо малих h_1, h_2, h_3 , точність також зменшувалась при зростанні β .

Обчислювальні експерименти для оцінки додаткової похибки, спричиненої використанням алгоритмів, що знижують обчислювальну складність, були проведені для $\tau = 0,0025$ і $n_x = n_y = n_z = 10$.

Загалом експерименти підтвердили отримані теоретичні оцінки. Для крайових умов як першого, так і другого роду, для схем 2a, 2b та 3a, 3b точність при використанні як методу фіксованої пам'яті (алгоритм 2), так і розкладання у ряди (алгоритм 3) зменшується зі збільшенням α та збільшенням β . При збільшенні значення χ , точність розв'язання зменшується у випадку, коли використовується алгоритм 2. Для алгоритму 3, який був розроблений з урахуванням особливостей ψ -похідної Капуто, точність зростає зі збільшенням χ .

У випадку неявної схеми 1 використання алгоритму 2 суттєво (у 20–30 разів) знижує точність отриманих розв'язків. При використанні алгоритму 3, точність зменшується більше, ніж у випадку схем 2a, 2b та 3a, 3b, з однаковими тенденціями її зміни при зміні значень параметрів дробових похідних.

Використання алгоритму обчислення значень $\Delta_x^{(\beta)}$, $\Delta_y^{(\beta)}$, $\Delta_z^{(\beta)}$ шляхом розкладання в ряди несуттєво вплинуло на точність розрахунків для $\varepsilon_2 \leq 10^{-4}$ коли один з розмірів сітки був менше 400, як для неявної схеми 1, так і для схем розщеплення. При $\alpha = \beta = \chi = 0,5$ і $10^{-4} > \varepsilon_2 \geq 10^{-3}$, точність починала зменшуватися, коли один із розмірів сітки ставав більшим за 160. Для $\varepsilon_2 > 10^{-3}$, у більшості випадків точність зменшувалась більш ніж у 10 разів.

Залежності часу розв'язання t від розміру $N = n_x = n_y = n_z$ сітки у формі $t = k_t^{(1)} N^{k_t^{(2)}}$ наведено на рис. 6.21 для випадку крайових умов другого роду, $\alpha = \beta = \chi = 0,5$, $\tau = 0,0025$, та $T = 2$ для схем 1, 2a та 2b без використання

алгоритмів оптимізації. Для схеми 3а залежність мала той самий порядок $k_t^{(2)}$ що і для схеми 2а, проте час був на $\sim 20\%$ вище. Схема 3b була на $\sim 5\%$ повільніше, ніж схема 2b.

Зміна типу крайових умов на умови першого роду впливає на матриці СЛАР змінюючи кількість ітерацій, необхідних для їх розв'язання, і, відповідно, впливає на час роботи алгоритмів. Швидкодія схеми 2b у цьому випадку збільшується на $\sim 5\%$, швидкодія схеми 3b зменшується на $\sim 6\%$, а швидкодія схеми 1 збільшується на $\sim 40\%$.

Значення порядку $k_t^{(2)}$ відповідають теоретичним оцінкам для схем 2а та 3а. У цьому випадку відповідний порядок 3,31 трохи вище теоретично передбаченого 3, що може бути спричинено додатковими факторами, такими як кешування. Для схем 2b і 1 – порядки 4,33 та 4,43 були близькими і також перевищували теоретично передбачене значення 4.

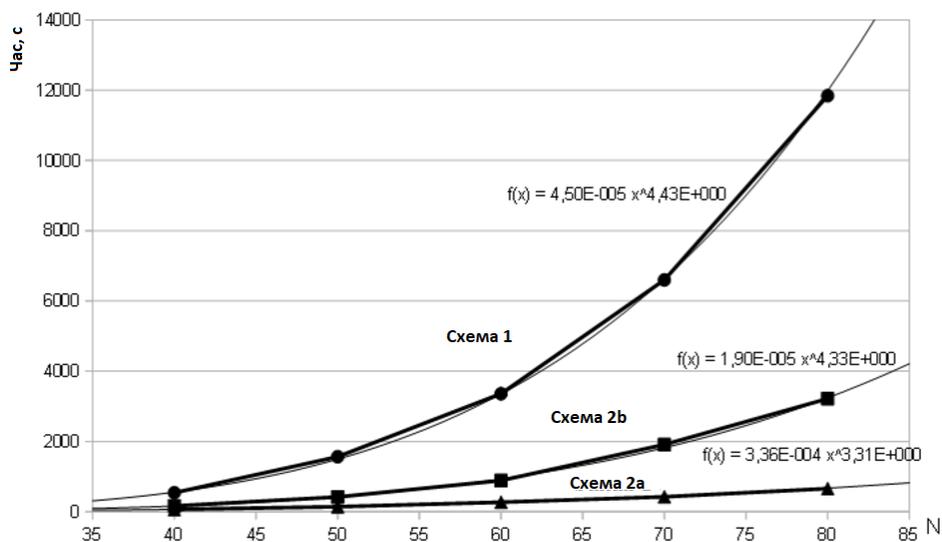


Рисунок 6.21 – Час, с, розв'язання задачі в залежності від розміру сітки N

При виконанні повного підсумовування при обчислення значень $\Delta_t^{(\alpha)}C$, час розв'язання квадратично залежить від номера кроку за часом. Використання методу фіксованої пам'яті або підходу, що базується на розкладенні у ряди, зменшує порядок складності до лінійного. Залежно від значення K_t , алгоритм 3 може дещо

збільшити час розв'язання на початкових етапах моделювання. У порівнянні з повним підсумовуванням, метод фіксованої пам'яті у всіх випадках зменшує час, витрачений на обчислення.

Прискорення від використання множення матриці та вектори згідно з (6.34) спостерігалось лише на сітках, де принаймні один із розмірів був більшим за 360. При проведенні обчислень на одному кроці за часом на сітці розміру $360 \times 40 \times 40$ комірок, 18% прискорення було досягнуто для $\varepsilon_2 = 10^{-3}$ і супроводжувалось 250% втратою точності. Для $\varepsilon_2 = 2 \cdot 10^{-3}$ прискорення у 98% супроводжувалось 650% втратою точності. При розв'язанні тестової задачі з класичними похідними Капуто, алгоритм 3, який використовує представлення матриць СЛАР через матриці Теплиця, також може бути застосований для виконання операцій множення матриці на вектори. Для сітки розміру $n_1 \times 40 \times 40$ комірок, алгоритм 3 був на 15% швидшим, ніж неоптимізований алгоритм 1 при $n_1 < 360$. У цьому випадку алгоритм 2 пришвидшував обчислення при $\varepsilon_2 > 10^{-3}$, проте прискорення супроводжувалось значною втратою точності: для $n_1 = 400$ 50% прискорення було отримано зі збільшенням похибки на 550%.

Обсяг використаної пам'яті лінійно збільшується, коли для обчислення значень $\Delta_i^{(\alpha)} C$ використовувалось не оптимізоване підсумовування. У випадку алгоритму 3 зберігаються лише коефіцієнти рядів, тому обсяг пам'яті залишається постійним. У випадку методу фіксованої пам'яті для класичної похідної Капуто обсяг пам'яті лінійно збільшується до певного фіксованого значення і надалі залишається постійним. Для похідної Капуто–Катугамполи, досягаючи максимуму, обсяг пам'яті починає дещо зменшуватися при $\chi > 1$ та дещо збільшуватись при $\chi < 1$.

Баланс точності та швидкодії

Точність та швидкодія розглянутих алгоритмів для $\alpha = \beta = \chi = 0,5$; $n_1 = n_2 = n_3 = 40$, $\tau = 0,005; 0,0025$; $T = 2$ показана на рис. 6.22. Оскільки точність схеми 1 найвища, будемо вважати схему оптимальною, якщо вона швидша за

схему 1 і дозволяє отримувати розв'язки з похибкою $\varepsilon_0 \leq K_0 \bar{\varepsilon}_1$, де $\bar{\varepsilon}_1$ – похибка розв'язку, отриманого схемою 1.

Як видно з рис. 6.22, алгоритм 2 множення матриці на вектори тут є повільнішим, ніж неоптимізований алгоритм множення, коли поєднується з усіма розглянутими скінченно-різницевиими схемами на сітках такого малого розміру.

Використання оптимізованих схем для розрахунку значення $\Delta_i^{(\alpha)} C$ не суттєво скорочує час розв'язання при виконанні 400 кроків за часом ($\tau = 0,005$). За 800 кроків за часом ($\tau = 0,0025$), вони на до $\sim 25\%$ пришвидшують обчислення, але це супроводжується більш ніж $\sim 500\%$ втратою точності для алгоритму 3 і більше $\sim 1500\%$ втратою точності для алгоритму 2.

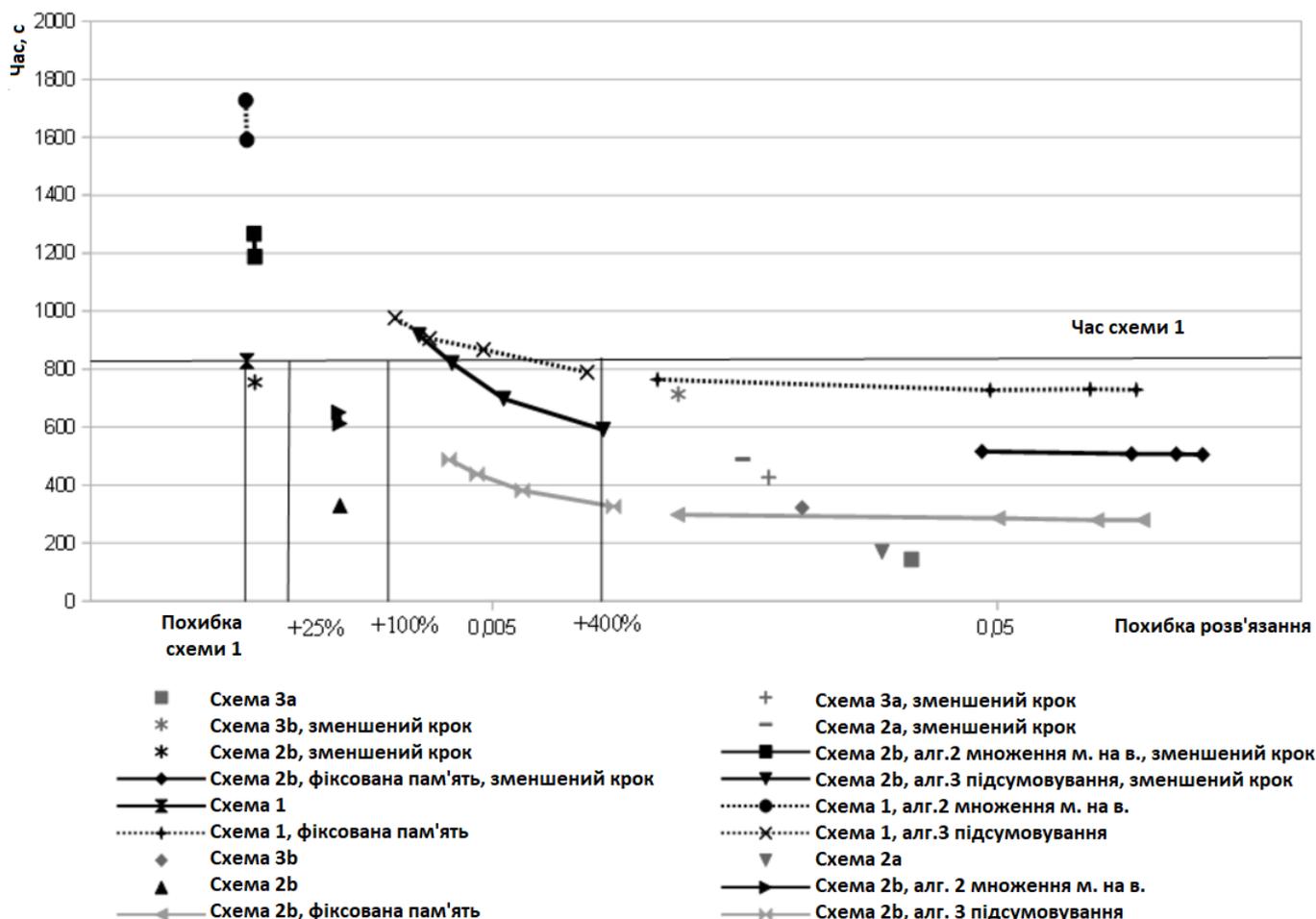


Рисунок 6.22 – Точність і швидкодія алгоритмів для $\alpha = \beta = \chi = 0,5$;

$$n_1 = n_2 = n_3 = 40; \tau = 0,005; \text{зменшений крок} - \tau = 0,0025; T = 2$$

Враховуючи допустиме погіршення точності на рівні 25%, єдиним варіантом, що прискорює обчислення, є використання схеми 2b зі зменшеною у два рази довжиною кроку за часом. При допустимому погіршенні точності в 2 рази схема 2b дає більш ніж у 2 рази вищу швидкість при рівній довжині кроку. Використання алгоритму 3 для обчислення значень $\Delta_t^{(a)}C$ у схемі 2b дозволяє на $\sim 25\%$ прискорити обчислення при 5-кратному допустимому погіршенні точності та $\tau = 0,0025$.

Схеми 2a, 3a та 3b були ефективні лише при допустимій втраті точності більш ніж у 18 разів.

Алгоритм вибору оптимальної схеми обчислень

Виходячи з вищенаведеного, пропонується наступний алгоритм автоматичного вибору оптимальної обчислювальної схеми в процесі розрахунків, який базується на припущенні, що L^2 -норма розв'язку не суттєво змінюється в процесі моделювання:

1) Позначимо як $d_{alg}^{(i)}$ L^2 -відстань між розв'язками (з L^2 -нормою $s_{alg}^{(i)}$), отриманими схемами $alg = 1, 2a, 2b, 3a, 3b$ ($alg(\varepsilon_2)$ означає застосування алгоритму 2 для множення матриці на вектори) на часовому кроці i та розв'язком, отриманим схемою 1 на тому ж кроці. Позначимо час, витрачений на отримання розв'язку на кроці i схемою alg , як $t_{alg}^{(i)}$;

2) Проводимо обчислення на першому кроці за часом за схемами 1, 2a, 2b, 3a та 3b без використання додаткових процедур оптимізації, вимірюючи час проведення обчислень $t_{alg}^{(0)}$. В якості базової вибираємо найшвидшу схему b серед схем, для яких $d_{alg}^{(0)} \leq K_1 s_1^{(0)}$, де K_1 – задана стала;

3) Якщо базовою схемою є схема 1, 2a або 3a, намагаємося знайти алгоритм множення матриці на вектори, який дозволив би прискорити обчислення. У випадку $\chi = 1$, розв'язуємо задачу на першому кроці за часом, використовуючи алгоритм 3, базований на представленні матриць \bar{T} з використанням матриць Теплиця. Для алгоритму 2, заснованому на представленні інтегралів у вигляді ряду,

знаходимо найбільше значення ε_2 за якого $\left|d_{b(\varepsilon_2)}^{(0)} - d_b^{(0)}\right| \leq K_2 d_b^{(0)}$, де K_2 – задана стала. Вибираємо алгоритм, застосування якого робить базову неоптимізовану схему швидшою;

4) Вибір найкращого алгоритму для розрахунку значення $\Delta_t^{(\alpha)}C$ виконується на кроці j для якого $t_b^{(j)} \geq K_3 t_b^{(0)}$, де K_3 – задана стала. Спочатку обчислення на одному кроці за часом виконуються за допомогою алгоритму 2 методу фіксованої пам'яті з $\varepsilon_3 = \bar{\gamma}_j^{j+1}, \bar{\gamma}_{2j/3}^{j+1}, \bar{\gamma}_{j/3}^{j+1}$. Часи обчислення t_i вимірюються разом з обчисленням L^2 -відстаней E_i до розв'язку, отриманого за неявною схемою 1. За допомогою методу найменших квадратів обчислюються коефіцієнти залежностей $t = a_i \varepsilon_3^{b_i}$ та $E(\varepsilon_3) = a_e e^{b_e/\varepsilon_3}$. На основі цих залежностей знаходимо значення ε_3 , для якого $E(\varepsilon_3) = K_1 s_1^{(0)}$. Виконуємо подібну процедуру для алгоритму 3, будуючи при $\varepsilon_2 = 25, 50, 75$ залежності від L^2 -відстаней у формі $E(\varepsilon_2) = a_{e2} \varepsilon_2^{b_{e2}}$. Обчислюємо лінійний прогноз загального часу обчислень. Для подальших розрахунків вибираємо алгоритм, для якого прогноз загального часу проведення розрахунків є найменшим.

Ефективність запропонованого алгоритму була перевірена шляхом розв'язання задачі прогнозування динаміки дифузії речовини у пласт із трьома проникними та трьома непроникними гранями. Крайові умови для такої задачі мали вигляд

$$C(0, y, z, t) = C(x, 0, z, t) = C(x, y, 0, t) = 1, \quad \left. \frac{\partial C}{\partial x} \right|_{x=L_x} = \left. \frac{\partial C}{\partial y} \right|_{y=L_y} = \left. \frac{\partial C}{\partial z} \right|_{z=L_z} = 0.$$

аналітичного розв'язку втрата точності оцінювалась як відносне відхилення від середнього значення розв'язку, отриманого за неявною схемою 1. Прискорення визначалось шляхом порівняння з часом розв'язання за схемою 1. Задача розв'язувалась при $n_x = n_y = n_z = n = 40, 80$ для різних значень порядків та параметрів дробових похідних α, β, χ при $K_1 = 1\%$, $K_2 = 10\%$, $K_3 = 1,5$.

Отримані результати показують, що втрата точності при використанні алгоритму автоматичного вибору обчислювальної схеми в ~ 10 разів перевищує

очікуване значення $K_1 = 1\%$. Втрата точності зростала зі зменшенням χ . Зниження точності супроводжувалось збільшенням прискорення обчислень.

При $n = 40$, згідно з алгоритмом автоматичного вибору обчислювальної схеми, використання представлення інтегралів у вигляді ряду для наближення похідних за просторовими змінними є неефективним. Однак для $n = 80$, в деяких випадках його використання може пришвидшити обчислення. Алгоритм 3 апроксимації $\Delta_t^{(\alpha)}C$, в більшості випадків, був обраний найбільш ефективним. При цьому необхідна кількість членів у рядах зменшувалась зі зменшенням χ .

Загалом, алгоритм автоматичного вибору обчислювальної схеми дозволяє прискорити обчислення в порівнянні як зі схемою 2b (до 4,1 разів у проведених обчислювальних експериментах), так і зі схемою 3a (до 3,6 разів). Однак прискорення супроводжується $\sim 10\%$ втратою точності.

6.5. Паралельні алгоритми розв'язання тривимірного рівняння дифузії з ψ -похідної Капуто за просторовою змінною

Будемо розглядати алгоритми розпаралелення обчислень на системах з розподіленою пам'яттю при скінченно-різницевому розв'язанні згідно зі схемами, описаними в п. 6.3 та 6.4, початково-крайової задачі щодо тривимірного рівняння аномальної дифузії вигляду (6.48) у випадку наявності дробових похідних лише за просторовими змінними ($\alpha = 1$) та крайових умов першого роду.

Використовуючи розпаралелення за даними, дослідимо два підходи [47] до розподілення скінченно-різницевої сітки в межах процесів програми, що виконується у середовищі MPI. У рамках першого підходу дані розподіляються одновимірними блоками. За необхідності, виконується перерозподіл з розподілу згідно з однією координатою на розподіл згідно з іншою координатою за допомогою глобальних операцій обміну даними. У другому підході, для організації розрахунків виключно з використанням локальних операцій обміну кожен з P процесів обробляє P^2 блоки комірок сітки, розділених за наступною червоно-чорною схемою. Припустимо, що в сітці розміру $N \times N \times N$ (тут і надалі N/P вважається цілим)

комірка (i, j, k) належить блоку $(iP/N, jP/N, kP/N)$. Блок (a, b, c) належить і обробляється процесом p якщо $a = a(p, b, c) = (P + ((b + c) \% P) - p) \% P$.

Спершу будемо розглядати випадок одновимірного блокового розподілу даних.

При використанні схеми розщеплення (6.57) блоки СЛАР, що відповідають рядкам комірок сітки зі змінною координатою x , розв'язуються на першому етапі незалежно з використанням багатопотокового розпаралелення в межах кожного процесу (початковий розподіл повинен відповідати цій координаті – процес p має містити та обробляти блоки комірок сітки (p, \bullet, \bullet)). На наступному кроці дані перерозподіляються для паралельного розв'язання СЛАР, що відповідають координаті y . Після розв'язання СЛАР, що відповідають всім трьом координатам, дані перерозподіляються для подальших обчислень по координаті x .

Твердження 6.3. За припущення, що обчислення виконуються P процесами, кожен з яких залучає C процесорних ядер, час виконання обчислень за схемою розщеплення (6.57) у випадку одновимірного блокового розподілу даних на одному кроці для сітки розміру $N \times N \times N$ може бути, враховуючи, що складність алгоритму BiCGstab в залежності від розміру матриці є лінійною, оцінений як

$$T_1(N, P, C) \approx 3k_1k_2(N)k_3 \frac{N^3}{PC} + 3(P-1)t_{tr} \left(\frac{N^3}{P} \right), \quad (6.65)$$

де k_1 – коефіцієнт продуктивності обчислювальної системи, $k_2(N)$ – середня кількість ітерацій, необхідних для отримання розв'язку СЛАР з матрицею розміру $N \times N$ алгоритмом BiCGstab, k_3 – кількість операцій множення матриці на вектори на одній ітерації алгоритму BiCGstab, $t_{tr}(S) = t_s S + t_c$ – час, що витрачається на обмін між процесами блоком даних розміру S , який вважається лінійно залежним від S .

Формула (6.65) не враховує час, витрачений на допоміжні операції для багатопотокового розпаралелювання в межах певного процесу.

Твердження 6.4. При $t_c = 0$ прискорення алгоритму, що визначається як відношення часу роботи алгоритму на одному ядрі процесора до часу роботи у паралельному середовищі, може бути оцінено як

$$S_1(N, P, C) \approx \frac{k_1 k_2(N) k_3 P C}{k_1 k_2(N) k_3 + C(P-1)t_s}. \quad (6.66)$$

У випадку неявної схеми (6.53) компонентом алгоритму, який вимагає обміну даними при виконанні розподілених обчислень, є множення векторів на матрицю $A^{(j)}$. Оскільки компоненти матриці $A^{(j)}$ можуть бути розкладені на суми координатно незалежних матриць $M_x^{(j)}$, T_x , D_x , $M_y^{(j)}$, T_y , D_y , $M_z^{(j)}$, T_z , D_z , множення виконується подібно до обчислень за схемою розщеплення: багатопотокове виконання незалежних множень на матриці $M_x^{(j)}$, T_x , D_x з подальшим перерозподілом необхідних даних та виконанням множення за матриці, що відповідають інших координатам.

Твердження 6.5. Час, необхідний для отримання розв'язку на одному кроці за неявною схемою (6.53) у випадку одновимірного блокового розподілу даних, можна оцінити як

$$T_2(N, P, C) \approx k_2(N^3) k_3 \left(k_1 \frac{N^3}{PC} + 3(P-1)t_{tr} \left(\frac{N^3}{P} \right) \right). \quad (6.67)$$

При $t_c = 0$ прискорення алгоритму можна оцінити як

$$S_2(N, P, C) \approx \frac{k_1 P C}{k_1 + 3C(P-1)t_s}. \quad (6.68)$$

У випадку червоно-чорного розподілу даних, при використанні схеми розщеплення (6.57) та виконанні обчислень за координатою x , кожен потік у процесі p розв'язує СЛАР, які відповідають рядкам сітки (\bullet, j, k) і належать до блоків (\bullet, b, c) . Множення матриці T_x на вектор v виконується згідно з алгоритмом розкладення у ряди, описаним у п. 6.3. Таким чином, кожен потік обробляє фрагмент рядка, що належить блоку $(a(p, b, c), b, c)$. Для множення на тридіагональну матрицю $M_x^{(j)}$, у випадку $p \neq 0$, потоку процесу p необхідно отримати значення $v \left[r_{a(p, b, c)(P/N)-1, j, k} \right]$ від відповідного потоку процесу $p-1$ і надіслати значення $v \left[r_{a(p, b, c)P/N, j, k} \right]$ до нього. У випадку $p \neq P-1$, потік повинен отримати

значення $v\left[r_{(a(p,b,c)+1)P/N,j,k}\right]$ від відповідного потоку процесу $p+1$ і надіслати значення $v\left[r_{(a(p,b,c)+1)P/N-1,j,k}\right]$ до нього. Для множення на матрицю T_x , потік повинен отримати від потоків інших процесів значення $v\left[r_{a(p,b,c)(P/N)-K_d,j,k}\right], \dots, v\left[r_{a(p,b,c)(P/N)-1,j,k}\right]$ і $v\left[r_{(a(p,b,c)+1)P/N,j,k}\right], \dots, v\left[r_{(a(p,b,c)+1)(P/N)+K_d,j,k}\right]$.

Щоб виконати такий обмін, потік, що виконується в межах процесу p повинен надіслати елементи $v\left[r_{a(p,b,c)(P/N)+l,j,k}\right], l < P/N$ до потоку, що виконується в межах процесу p_1 у випадку якщо

$$a(p_1,b,c)(P/N) - K_d \leq a(p,b,c)(P/N) + l < a(p_1,b,c)P/N$$

або

$$(a(p_1,b,c)+1)(P/N) < a(p,b,c)(P/N) + l \leq (a(p_1,b,c)+1)(P/N) + K_d.$$

Крім того, потік, у випадку $p \neq 0$, повинен отримати від процесу $p-1$ значення вектора $S'_{\bullet,a(p,b,c)(P/N)-1}$ і відправити до процесу $p+1$, у випадку $p \neq P-1$, його змінені значення $S'_{\bullet,(a(p,b,c)+1)(P/N)-1}$.

Для виконання вищеописаних обмінів даними, унікальний MPI тег має позначати потік повідомлень, пов'язаний з певним рядком сітки. Для запобігання ситуацій взаємного блокування, потоки різних процесів повинні одночасно обробляти однакові рядки сітки. Крім того, кожен процес повинен одночасно виконувати обчислення у всіх блоках двовимірної червоно-чорної декомпозиції сітки. Для цього пропонується встановити рівну кількість потоків у кожному процесі $C = N_c P$, де N_c – задана стала. Потік t_h при цьому обробляє рядки, які належать до блоку, для якого $a(p,b,c) = t_h / N_c$. Потоки повинні бути синхронізовані після того, як кожен з них обробив один рядок сітки. Оскільки кількість ітерацій, необхідних для розв'язання СЛАР для кожного з рядків, може змінюватися, потреба в синхронізації негативно впливає на завантаження ядер CPU.

Процедура проведення подальших обчислень за координатами y і z тут подібна до процедури проведення обчислень за координатою x .

Твердження 6.6. Припустімо, що для розв'язання СЛАР для кожного з рядків сітки потрібна однакова кількість ітерацій; операції обміну даними між потоками різних процесів не розпаралелюються; операція синхронізація при множенні векторів на фіксовану частину матриці T_x виконується між P^k процесами. Тоді час роботи паралельного алгоритму для локально-одновимірної скінченно-різницевої схеми розщеплення, який використовує червоно-чорне розбиття сітки, можна оцінити як

$$T_3(N, P, C) \approx 3k_2(N)k_3 \left(k_1 \frac{N^3}{PC} + N^2 (4t_{tr}(1) + 4P^k t_{tr}(K_d) + t_{tr}(K_m)) \right). \quad (6.69)$$

Через різну кількість ітерацій, необхідних для розв'язання СЛАР, що відповідають різним рядкам сітки, виконання операцій обміну між потоками різних процесів є асинхронним. Кожна операція обміну повинна виконуватися окремо, що призводить до їх великої кількості та, відповідно, великого обсягу часу, який витрачається на допоміжні операції. Це значно уповільнює процес розв'язання, нівелюючи позитивний ефект від розпаралелення. Для збільшення розміру повідомлень і зменшення їх кількості, пропонується наступний алгоритм поєднання запитів на надсилання та отримання даних, генерованих різними потоками:

1) Масиви запитів на надсилання та отримання даних створюються в кожному процесі для кожного іншого процесу. Кожен запит містить: буфер надсилання/отримання; кількість елементів і тип даних; тег (значення, необхідні для виклику відповідної функції MPI); ідентифікатор потоку, який ініціював обмін; прапорець, що вказує завершеність операції; conditional variable та відповідний mutex для синхронізації потоків; прапорець, який повідомляє потік-ініціатор запиту про його завершення;

2) Масиви отриманих повідомлень створюються в кожному процесі для кожного іншого процесу. Вони містять значення, необхідні для виклику відповідної функції MPI та прапорець завершення операції;

3) Операції надсилання даних виконуються у неблокуючому режимі. При виконанні операції надсилання, значення параметрів функції MPI записуються в масив запитів на надсилання для відповідного процесу;

4) Операція отримання даних виконується наступним чином:

(a) Запит на отримання додається до масиву, що відповідає процесу-відправнику;

(b) Перевіряється наявність необхідних даних у масивах отриманих повідомлень, які відповідають необробленим запитам на отримання. За наявності даних, вони копіюються у відповідні буфери і прапорці завершення операції встановлюються як для запиту на отримання, так і для отриманого повідомлення. Перевірка відбувається лише серед необроблених запитів та повідомлень;

(c) Виконання операції завершується, якщо необхідні для поточного запиту дані були знайдені;

(d) Якщо всі потоки виконали принаймні одну операцію отримання даних після останнього відправлення, виконується надсилання всіх наявних для відправлення даних відповідним процесам;

(e) Виконується наступний цикл очікування даних з виходом з нього при завершенні виконання запиту:

i. Якщо інший потік чекає отримання даних, перехід у стан очікування conditional variable запиту з подальшим виходом із циклу;

ii. Сповіднення потоків, які очікують завершення оброблених запитів, використовуючи conditional variable;

iii. За необхідності надсилання даних, накопичених для відправлення іншим процесам;

iv. Очікування отримання даних від процесу-відправника.

У найкращій ситуації, використання вищеописаного алгоритму зменшує кількість необхідних операцій обміну даними у C разів. Оцінка (6.69), припускаючи, що операції обміну виконуються з блоками даних однакового розміру, приймає вигляд

$$T_4(N, P, C) \approx 3k_2(N)k_3 \left(k_1 \frac{N^3}{PC} + \frac{9N^2}{C} t_{tr} \left(\frac{C}{9} (4 + 4P^\kappa K_d + K_m) \right) \right). \quad (6.70)$$

Прискорення алгоритму може бути оціненим як

$$S_4(N, P, C) \approx \frac{3k_1NPC}{3k_1N + PC(4 + 4P^\kappa K_d + K_m)t_s + 9Pt_c}. \quad (6.71)$$

У випадку неявної скінченно-різницевої схеми та червоно-чорного розподілу даних, обміни, описані для попереднього алгоритму, повинні виконуватися при виконанні кожної операції множення матриці на вектор. Для кожного рядка відповідні обміни включають

- 1) Обмін даними для комірок, які лежать на границях блоків, для множення на тридіагональну матрицю $M_x^{(j)}$;
- 2) Обмін K_d значеннями елементів векторів для множення на фіксовану частину матриці T_x ;
- 3) Обмін K_m значеннями членів ряду (6.32) для множення на апроксимовану частину матриці T_x .

Оскільки обміни в неявній схемі є синхронними, їх можна об'єднати для всіх оброблюваних рядків.

Твердження 6.7. Для часу, необхідного для отримання розв'язку на одному кроці за неявною скінченно-різницевою схемою при червоно-чорному розподілі даних, справедлива оцінка

$$T_5(N, P, C) \approx k_2(N^3)k_3 \left(k_1 \frac{N^3}{PC} + 4t_{tr}(N^2) + 4P^\kappa t_{tr}(K_d N^2) + t_{tr}(K_m N^2) \right). \quad (6.72)$$

При $t_c = 0$ прискорення алгоритму можна оцінити як

$$S_5(N, P, C) \approx \frac{k_1 k_2 (N^3) k_3 NPC}{k_1 k_2 (N^3) k_3 N + PC(4 + 4P^\kappa K_d + K_m)t_s}. \quad (6.73)$$

Обчислювальні експерименти

У якості тестової задачі використовувалась задача, описана у п. 6.4. Критерієм точності було середньоквадратичне відхилення $\bar{\varepsilon}$ чисельного розв'язку задачі від аналітичного.

Тестова задача розв'язувалась на одному кроці за часом довжиною $\tau = 0,025$ на сітці розміру $N \times N \times N$, $N = 20, \dots, 150$ на кластері СКІТ-4 Інституту кібернетики

імені В. М. Глушкова НАН України. Вихідні коди розробленого програмного забезпечення наведено у [265]. Вивчалась масштабованість алгоритмів при використанні до 3 вузлів. Ітераційний процес алгоритму BiCGstab зупинявся, коли середнє значення нев'язки ставало менше, ніж 10^{-15} . Значення параметру точності ε_2 алгоритму розкладення у ряди, описаного у п. 6.3, дорівнювало 10^{-5} .

У випадку неявної схеми (6.53) отримані дані про час виконання, прискорення S та ефективність розпаралелювання E , яке визначається як відношення прискорення до кількості використовуваних ядер CPU, наведено у табл. 6.6 та на рис. 6.23, 6.24, 6.25.

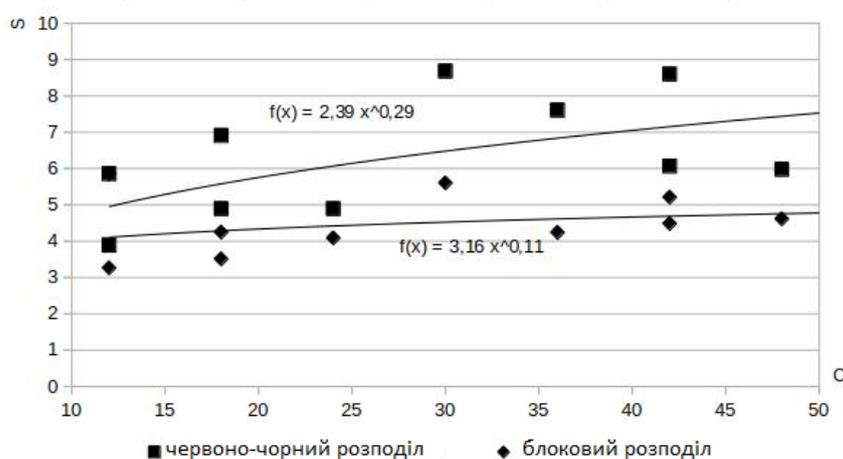


Рисунок 6.23 – Прискорення S алгоритмів залежно від кількості C використовуваних ядер для неявної схеми та $N = 150$

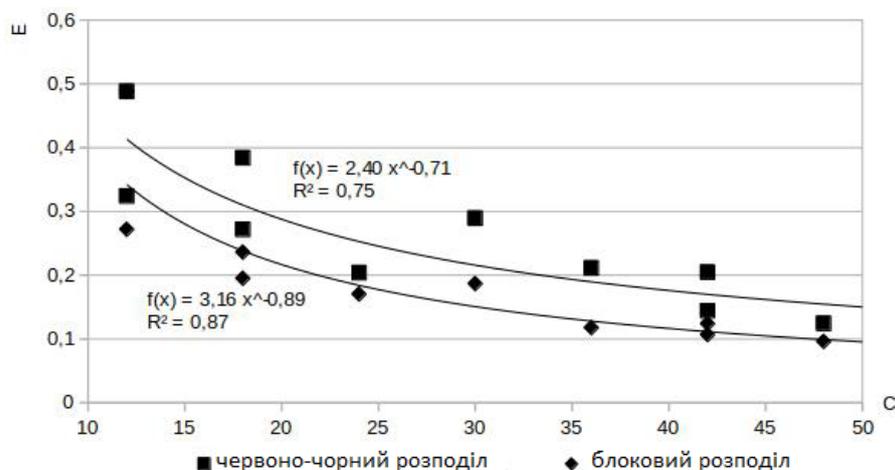


Рисунок 6.24 – Ефективність розпаралелювання E в залежності від кількості C використовуваних ядер для неявної схеми та $N = 150$

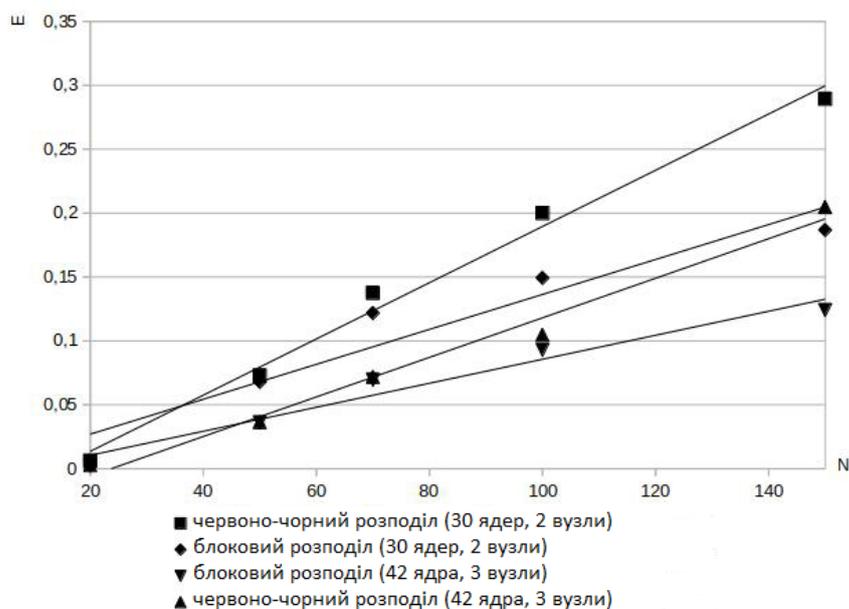


Рисунок 6.25 – Ефективність розпаралелювання E в залежності від розміру сітки N для неявної схеми

Таблиця 6.6 – Час виконання обчислень, с, для неявної схеми на одному кроці (найменші отримані часи для заданої кількості вузлів кластера)

Кількість процесів	1	4	4	6
Кількість ядер	12	24	30	42
Кількість вузлів	1	1	2	3
	OpenMP	MPI		
N	Червоно-чорний розподіл			
20	0,05	4,26	0,77	1,24
50	0,88	7,64	2,13	3,05
70	3,29	11,14	4,56	6,24
100	12,95	21,07	12	16,38
150	55,13	66	37,2	37,56
	Одновимірний блоковий розподіл			
20		4,11	0,78	1,51
50		7,63	2,29	3,07
70		11,53	5,15	6,44
100		22,73	16,08	18,44
150		79	57,66	62

Як видно з даних, наведених у табл. 6.6, при використанні неявної скінченно-різницевої схеми паралельний алгоритм з червоно-чорним розподілом є швидшим, ніж алгоритм з блоковим розподілом. За фіксованого розміру сітки, загальна масштабованість алгоритмів (рис. 6.23) є невисокою, а ефективність розпаралелювання (рис. 6.24) алгоритму з червоно-чорним розподілом зменшується зі збільшенням кількості ядер швидше, ніж для алгоритму з блоковим розподілом. У випадку одновимірного блокового розподілу, експериментальні дані щодо прискорення неявної схеми відповідають залежності (6.68), при $t_s / k_1 \approx 0,0275$. Для червоно-чорного розподілу відповідність залежності (6.73) досягається за припущення, що k_2 є фіксованим, та $k_1 k_2 k_3 \approx 3,4$.

При збільшенні розміру сітки (рис. 6.25), ефективність алгоритму з червоно-чорним розподілом збільшується з трендом, близьким до лінійного, відповідно до оцінки (6.73). При використується блокового розподілу, збільшення ефективності має тренд, близький до логарифмічного, прямуючи до фіксованого теоретичного значення (6.68) для сіток великого розміру.

Для неявної схеми кількість операцій обміну на одній ітерації алгоритму BiCGstab не залежить від розміру сітки та лінійно збільшується зі збільшенням кількості процесів для обох розглядуваних схем розподілу даних. Водночас більша кількість операцій виконується при блоковому розподілі. Загальна кількість комунікацій між процесами на одній ітерації алгоритму BiCGstab в залежності від кількості процесів збільшується для червоно-чорного розділення згідно з (6.72) та $\kappa \approx 1$, що означає наявність глобальних операцій обміну даними, які обмежують масштабованість схеми при $\varepsilon_2 = 10^{-5}$. У випадку блокового розподілу даних обсяг обмінів, згідно з (6.67), зменшується зі збільшенням кількості процесів.

Отримані дані щодо часу виконання, прискорення та ефективності розпаралелювання для схеми розщеплення наведені в табл. 6.7 та рис. 6.26, 6.27.

Як видно з даних, наведених у табл. 6.7, використання червоно-чорного розподілу для розпаралелювання скінченно-різницевої схеми розщеплення є суттєво неефективним. Використання одновимірного блокового розподілу дозволяє отримати близьке до лінійного прискорення обчислень зі зменшенням ефективності

при збільшенні кількості використовуваних ядер (рис. 6.26). Ефективність зростає зі збільшенням розміру сітки (рис. 6.27), прямуючи до фіксованої теоретичної оцінки (6.66). Близьке до лінійного збільшення прискорення зі збільшенням кількості використовуваних ядер адекватно описується оцінкою (6.66) для тих самих значень коефіцієнтів продуктивності t_s і $k_1 k_2 k_3$, що і у випадку неявної скінченно-різницевої схеми. Збільшення прискорення забезпечується обернено пропорційною залежністю обсягів обмінів даними від розміру сітки та низькою лінійно зростаючою кількістю операцій обміну.

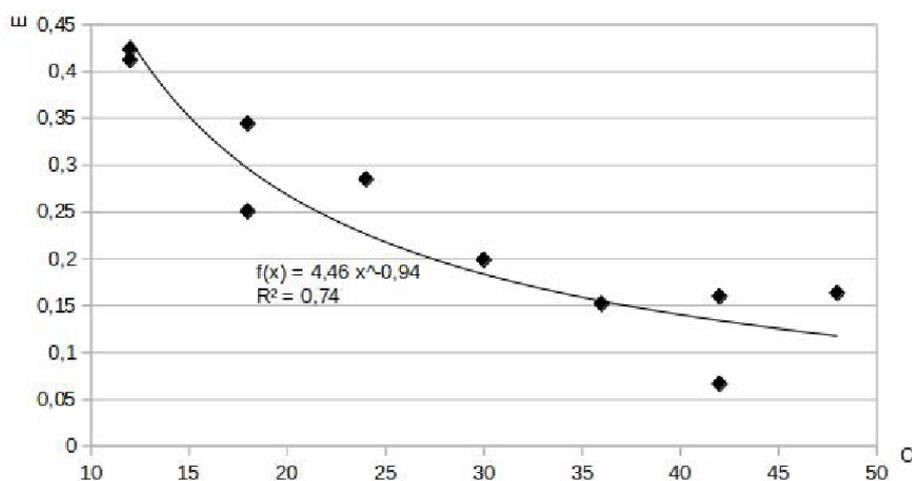


Рисунок 6.26 – Ефективність розпаралелювання E в залежності від кількості C використаних ядер для схеми розщеплення та блокового розподілення даних, $N = 150$

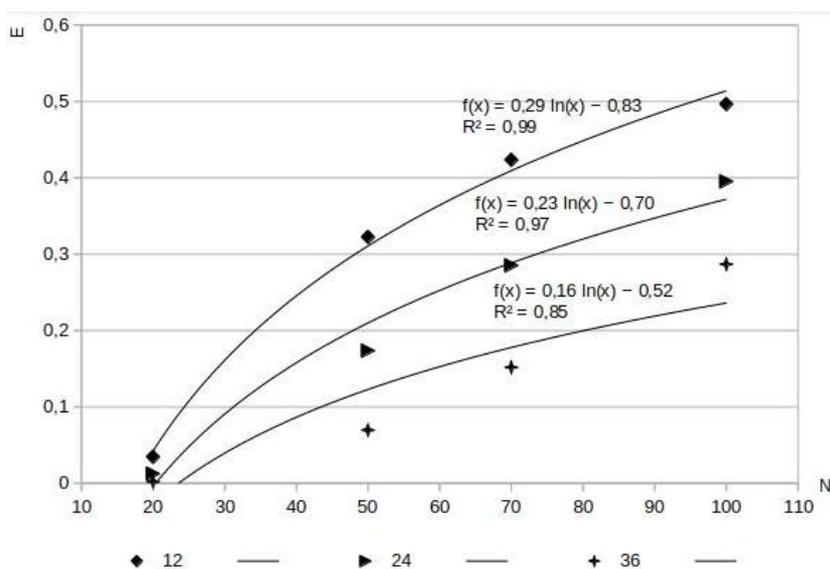


Рисунок 6.27 – Ефективність розпаралелювання E в залежності від розміру сітки N для схеми розщеплення та блокового розподілу даних

Таблиця 6.7 – Час виконання обчислень, с, на одному кроці за часом для схеми розщеплення (найменші отримані часи для заданої кількості вузлів кластера)

Кількість процесів	1	2	2	3
Кількість ядер	12	12	18	36
Кількість вузлів	1	1	2	3
N	Червоно-чорний розподіл			
	OpenMP	MPI		
20	0,04	2,28	5,22	6,88
50	0,58	22,59	43,05	57,6
70	2,14	55,14	100	143
100	8,19	147	249	390
150	42,19	438	723	1220
	Одновимірний блоковий розподіл			
	OpenMP	MPI		
Кількість процесів		4	8	12
Кількість ядер		24	48	68
Кількість вузлів		1	2	3
20		0,4	0,23	4,83
50		0,78	1,12	2,54
70		1,9	1,66	4,93
100		5,67	3,89	3,56
150		21,32	15,47	13,07

Прискорення алгоритму з червоно-чорним розподілом даних адекватно описується оцінкою (6.71) у випадку значень коефіцієнтів продуктивності, отриманих для інших алгоритмів, і значення коефіцієнта t_c , що відповідає за час, витрачений на допоміжні операції при обміні даними, рівного 5000. Це підтверджує той факт, що неефективність такого алгоритму зумовлена великою кількістю необхідних операцій обміну, що, в свою чергу, зумовлено їх асинхронним характером через різну кількість ітерацій алгоритму BiCGstab, необхідних для розв'язання СЛАР.

Для оцінки впливу значення параметра ε_2 алгоритму розкладення у ряди на точність розв'язків та швидкодію алгоритмів було проведено серію обчислювальних експериментів на сітці з $N=100$ використовуючи, з одного боку, апроксимаційну схему при $\varepsilon_2 = 10^{-1}, \dots, 10^{-7}$, а, з іншого боку, повне обчислення значень елементів матриць T_x, T_y, T_z . Розв'язання відбувалося з використанням неявної скінченно-різницевої схеми для випадків обох розглядуваних підходів до розподілу даних. За

допомогою схеми розщеплення розв'язання виконувалось лише для випадку одновимірного блокового розподілу даних. Отримані дані про час розв'язання наведені в табл. 6.8. Точність розв'язків, кількість ітерацій алгоритму BiCGStab для неявної схеми та значення параметрів K_d і K_m алгоритму розкладення у ряди наведені у табл. 6.9.

Таблиця 6.8 – Час виконання обчислень, с, при зміні значення ε_2

Кількість процесів	1	4	4	1	4	4	1	4	4
Кількість ядер	12	24	30	12	24	30	12	24	30
Кількість вузлів	1	1	2	1	1	2	1	1	2
ε_2	Схема розщеплення (блоковий розподіл)			Неявна схема (червоно-чорний розподіл)			Неявна схема (блоковий розподіл)		
10^{-1}	2,99	2,21	1,56	18,59	21,52	9,35	17,78	25,71	15,39
10^{-2}	3,14	1,98	1,75	13,73	16,2	7,46	13,35	19,4	12,1
10^{-3}	6,97	4,14	2,79	16,61	17,85	9,24	15,47	20,64	14,18
10^{-4}	8,55	5,06	3,45	17,52	20,1	11,05	17,17	22,66	15,8
10^{-5}	9,22	5,43	3,73	18,15	21,23	12,23	17,8	22,77	15,99
10^{-6}	9,48	5,4	4,12	19,26	21,24	12,74	18,61	23,96	17,8
10^{-7}	10	5,67	4,24	19,63	22,07	12,8	19,22	24,59	17,57
Повний розрахунок матриць $T_{x/y/z}$	4,06	3,06	2,31	13,81	22,36	14,31	13,62	19,4	12,57

При розмірі сітки $N = 100$, використання алгоритму розкладення у ряди разом зі схемою розщеплення дозволяє прискорити обчислення при $\varepsilon_2 \geq 10^{-2}$ у порівнянні з випадком повного розрахунку матриць $T_{x/y/z}$. Прискорення дорівнює 29% для 12 ядер і збільшується до 84% для 30 ядер при падінні точності більш ніж у 15 разів. При $\varepsilon_2 \geq 10^{-3}$, зниження точності не є значним, але більша кількість членів у відповідних рядах, необхідних для її досягнення, не дозволяє пришвидшити обчислення.

Використання неявної схеми у порівнянні зі схемою розщеплення дозволяє отримати у 25 разів точніші розв'язки, коли значення елементів матриць T_x , T_y , T_z обчислюються повністю або при $\varepsilon_2 \geq 10^{-3}$. При використанні алгоритму розкладення у ряди з $\varepsilon_2 < 10^{-3}$, точність значно падає, досягаючи рівня точності схеми розщеплення при більш ніж у 5 разів більшому часі, необхідному для отримання розв'язків.

Таблиця 6.9 – Точність розв'язання та параметри алгоритмів

ε_2	Похибка $\bar{\varepsilon}$ для схеми розщеплення	Похибка $\bar{\varepsilon}$ для неявної схеми	Кількість ітерацій (неявна схема)	K_d	$K_d + K_m$
10^{-1}	$7.39 \cdot 10^{-6}$	$7.66 \cdot 10^{-6}$	41	1	3
10^{-2}	$1.66 \cdot 10^{-6}$	$1.52 \cdot 10^{-6}$	29	2	7
10^{-3}	$1.06 \cdot 10^{-7}$	$4.06 \cdot 10^{-9}$	26	10	30
10^{-4}	$1.06 \cdot 10^{-7}$	$4.06 \cdot 10^{-9}$	26	16	43
10^{-5}	$1.06 \cdot 10^{-7}$	$4.06 \cdot 10^{-9}$	27	20	52
10^{-6}	$1.06 \cdot 10^{-7}$	$4.06 \cdot 10^{-9}$	26	23	60
10^{-7}	$1.06 \cdot 10^{-7}$	$4.07 \cdot 10^{-9}$	27	27	66
Повний розрахунок матриць $T_{x/y/z}$	$1.06 \cdot 10^{-7}$	$4.09 \cdot 10^{-9}$	28	100	0

З іншого боку, ефективність паралельної реалізації неявної схеми у випадку повного обчислення елементів матриць T_x , T_y , T_z є нижчою для червоно-чорного розподілу у порівнянні з випадком використання алгоритму розкладення у ряди. Причиною цього є необхідність виконувати глобальні операції обміну при множенні повністю заповнених матриць на вектори. У випадку одновимірного блокового розділу ефективності розпаралелення суттєво не відрізняються.

Загальний час проведення обчислень на одному кроці для схеми розщеплення та час виконання однієї ітерації алгоритму BiCGstab для неявної схеми наведені на рис. 6.28 в залежності від $K_d + K_m$ для випадку 4 процесів, що виконуються на двох вузлах із загальною кількістю 30 ядер. Для всіх трьох розглянутих алгоритмів зростання часу тут близьке до лінійного, що узгоджується з теоретичними очікуваннями.

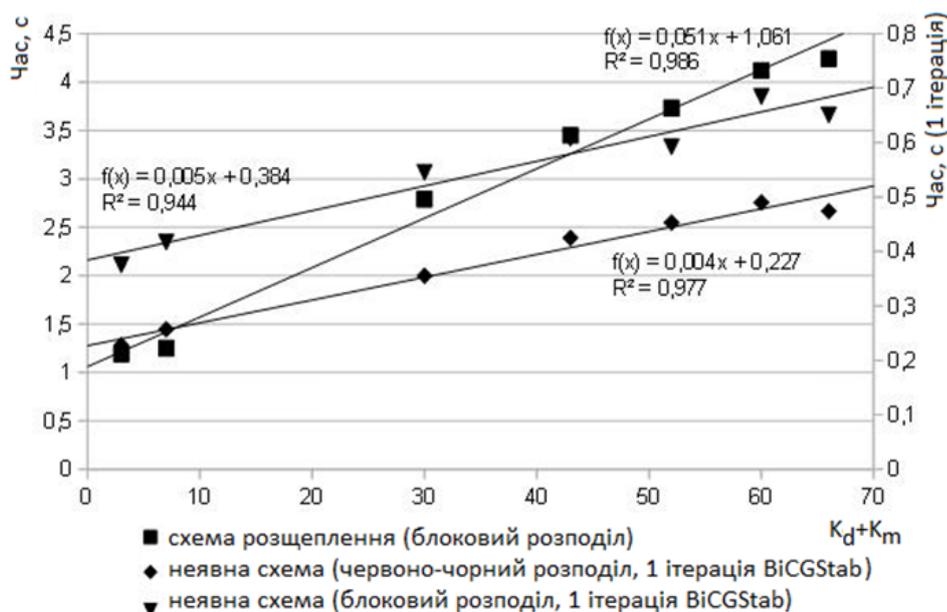


Рисунок 6.28 – Час проведення обчислень в залежності від $K_d + K_m$

6.6. Алгоритм оптимізації обчислень для моделей, що містять похідну Атангана–Балеану

Похідна Атангана–Балеану є дробовою похідною з несингулярним ядром на основі функції Міттаг–Леффлера. Чисельні методи, що використовуються для обчислення її значень, переважно базуються на комбінуванні її представлення у вигляді ряду та у вигляді інтегралу [199]. Безпосереднє використання таких методів під час розв’язування диференціальних рівнянь, що містять дробові похідні Атангана–Балеану, призводить до лінійного порядку обчислювальної складності в залежності від номера кроку за часом.

Для зниження обчислювальної складності застосуємо [45] підхід [44, 46], базований на розділенні змінних у представленнях ядер інтегральних операторів, який показав свою ефективність для дробової похідної Капуто–Герасимова (п. 5.3, 5.4) та ψ -похідної Капуто (п. 6.1–6.4).

Похідна Атангана–Балеану [18] від неперервно диференційованої функції f порядку α має вигляд

$${}^{ABC}D_t^\alpha f(t) = \frac{1}{1-\alpha} \int_0^t f'(z) E_\alpha \left(-\frac{\alpha}{1-\alpha} (t-z)^\alpha \right) dz, \quad 0 < \alpha < 1, t > 0, \quad (6.74)$$

де $E_\alpha(z)$ – функція Міттаг-Леффлера [158].

Наближаючи $f'(z)$ у (6.74) з першим порядком точності, для $t_j = j\tau$ (τ – довжина кроку за часом, $j > 0$ – номер кроку) отримуємо наступне скінченно-різницеве представлення:

$${}^{ABC}D_t^\alpha f(t)\Big|_{t=t_j} \approx {}^{ABC}\Delta_t^\alpha f(t)\Big|_{t=t_j} = \frac{1}{1-\alpha} \sum_{s=0}^{j-1} \frac{f(t_{s+1}) - f(t_s)}{\tau} I_{sj}, \quad (6.75)$$

$$I_{sj} = \int_{t_s}^{t_{s+1}} E_\alpha \left(-\frac{\alpha}{1-\alpha} (t_j - z)^\alpha \right) dz. \quad (6.76)$$

Для розрахунку I_{sj} розглянемо спочатку базове представлення функції Міттаг-Леффлера у вигляді наступного ряду, збіжного при $|z| \leq 1$,

$$E_\alpha(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + 1)}.$$

Аналогічно [44, 46], розкладаючи $(t_j - z)^\beta$ у ряд у вигляді

$$(t_j - z)^\beta = \sum_{l=0}^{\infty} \left(\binom{\beta}{l} (-1)^l z^l t_j^{\beta-l} \right), \quad \binom{\beta}{l} = \frac{\beta(\beta-1)\dots(\beta-l+1)}{l!},$$

отримуємо для інтеграла (6.76)

$$E_\alpha \left(-\frac{\alpha}{1-\alpha} (t_j - z)^\alpha \right) = \sum_{k=0}^{\infty} \left(\left(-\frac{\alpha}{1-\alpha} \right)^k \frac{1}{\Gamma(\alpha k + 1)} \sum_{l=0}^{\infty} \left(\binom{\alpha k}{l} (-1)^l z^l t_j^{\alpha k - l} \right) \right),$$

$$I_{sj} = \sum_{l=0}^{\infty} \left((t_{s+1}^{l+1} - t_s^{l+1}) \sum_{k=0}^{\infty} \left((-1)^{l+k} \left(\frac{\alpha}{1-\alpha} \right)^k \frac{1}{\Gamma(\alpha k + 1)(l+1)} \binom{\alpha k}{l} t_j^{\alpha k - l} \right) \right). \quad (6.77)$$

У випадку $|z| > 1$ використаємо представлення [169] функції Міттаг-Леффлера у вигляді інтегралу

$$E_\alpha(z) = -\frac{1}{\pi\alpha} \int_0^\infty \frac{z \sin(\pi\alpha)}{u^2 - 2uz \cos(\pi\alpha) + z^2} e^{-u^{1/\alpha}} du. \quad (6.78)$$

Виконуючи в (6.78) заміну змінних $u = v(t_j - z)^\alpha$ отримуємо

$$E_\alpha \left(-\frac{\alpha}{1-\alpha} (t_j - z)^\alpha \right) = -\frac{1}{\pi\alpha} \int_0^\infty \frac{p \sin(\pi\alpha)}{v^2 - 2vp \cos(\pi\alpha) + p^2} e^{-v^{1/\alpha}(t_j - z)} dv,$$

$$p = -\frac{\alpha}{1-\alpha}$$

і

$$I_{sj} = \frac{1}{\pi\alpha} \int_0^{\infty} \frac{p \sin(\pi\alpha)}{v^2 - 2vp \cos(\pi\alpha) + p^2} v^{-1/\alpha} e^{-v^{1/\alpha} t_j} \left(e^{v^{1/\alpha} t_s} - e^{v^{1/\alpha} t_{s+1}} \right) dv. \quad (6.79)$$

Для обчислення I_{sj} використовуючи (6.79), обмежимо верхню межу інтегрування точкою C , у якій інтегрована функція стає меншою ніж задана стала ε_1 , і застосуємо до інтеграла квадратурну формулу з вагами w_l і вузлами $v_l, l=1, \dots, N$ отримуючи

$$I_{sj} = \sum_{l=1}^N \left(\left(e^{v_l^{1/\alpha} t_s} - e^{v_l^{1/\alpha} t_{s+1}} \right) \frac{w_l}{\pi\alpha} \frac{p \sin(\pi\alpha)}{v_l^2 - 2v_l p \cos(\pi\alpha) + p^2} v_l^{-1/\alpha} e^{-v_l^{1/\alpha} t_j} \right). \quad (6.80)$$

Не знижуючи загальності формули (6.80), будемо надалі використовувати метод прямокутників для чисельного інтегрування та застосуємо стратегію розподілу на підінтервали, подібну до застосованої в [142] для похідної Атангана–Балеану та в [129] для похідної Капуто. Подібно до [142], представимо інтеграл (6.79) у вигляді суми експонент, надалі застосовуючи до такої суми техніку розділення змінних аналогічно [44, 46], отримуючи знижену обчислювальну складність при виконанні обчислень зі збільшенням значень аргументу.

Вибір ваг і вузлів чисельного інтегрування будемо проводити з урахуванням того, що функція $\frac{p \sin(\pi\alpha)}{v^2 - 2vp \cos(\pi\alpha) + p^2}$ має максимум при $v_m = p \cos(\pi\alpha)$ і має дві точки перегину навколо цього максимуму в $v_m \pm v_{inf}, v_{inf} = \sqrt{(p^2 - v_m^2)/3}$. Інша частина інтегрованої функції зменшується зі збільшенням v для досить великих значень v . Вводячи множник n , що впливає на точність, рівномірно розподілимо $1/3$ вузлів у діапазоні $[0, v_m - nv_{inf}]$ з вагами $w^{(1)} = 3(v_m - nv_{inf})/N$ і $1/3$ вузлів – в діапазоні $[v_m - nv_{inf}, v_m + nv_{inf}]$ з вагами $w^{(2)} = 6nv_{inf}/N$. Остання $1/3$ вузлів розподіляється у діапазоні $v > v_m + nv_{inf}$ з відстанню між вузлами і, відповідно,

вагами, що лінійно зростають від $w^{(2)}$ до $w^{(3)} = 6\left(C - (v_m + nv_{inf})\right) / N$ для забезпечення $v_N > C$.

Для виконання послідовних обчислень за (6.75), обмежимо ряд у представленні (6.77) виконуючи підсумовування по l для $l=1, \dots, N$. Підсумовування по k будемо виконувати до тих пір, поки значення заданої кількості послідовних членів ряду не стане менше $\frac{\varepsilon_2}{l!}$, де ε_2 – задана стала, виконуючи щонайменше l операцій. Після такого обмеження підсумовування у (6.77) обидва представлення (6.77) та (6.80) матимуть вигляд

$$I_{sj} = \sum_{l=0}^N f_{1l}(t_s, t_{s+1}) f_{2l}(t_j),$$

а (6.75) перетвориться на

$${}^{ABC} \Delta_t^\alpha f(t) \Big|_{t=t_j} = \frac{1}{1-\alpha} \sum_{l=0}^N S_{l,j-1} f_{2l}(t_j), \quad (6.81)$$

де $S_{l,j-1}$ задається рекурентним співвідношенням

$$S_{l,j-1} = \sum_{s=0}^{j-1} \frac{f(t_{s+1}) - f(t_s)}{\tau} f_{1l}(t_s, t_{s+1}) = S_{l,j-2} + \frac{f(t_j) - f(t_{j-1})}{\tau} f_{1l}(t_{j-1}, t_j), S_{l0} = 0.$$

Оцінки точності та швидкодії для представлення рядом (6.77)

Теорема 6.5. Для похибки усічення ряду (6.77) справедлива оцінка

$$\varepsilon_3(N) \approx \frac{S_N}{N+1} \left(e^{t_{s+1} t_s^{N+2}} - e^{t_s t_s^{N+2}} \right).$$

Доведення.

Позначимо $I_{sj}^{(1)} = \sum_{l=0}^N \left(f_{1l}^{(1)} \sum_{k=0}^{m_l} f_{2kl}^{(1)} \right)$, $m_l > l$, $\left| f_{2km_l}^{(1)} \right| \leq \frac{\varepsilon_2}{l!}$, $f_{1l}^{(1)} = t_{s+1}^{l+1} - t_s^{l+1}$, $f_{2kl}^{(1)} = \frac{f_{3kl}^{(1)}}{l!}$,

$$f_{3kl}^{(1)} = (-1)^{l+k} \left(\frac{\alpha}{1-\alpha} \right)^k \frac{(\alpha k)_l}{\Gamma(\alpha k + 1)(l+1)} t_j^{\alpha k - l}, \text{ де } (\cdot)_l \text{ – спадаючий факторіал.}$$

Оскільки ряд по k у (6.77) є знакопереміжним для $k > l$, маємо

$$I_{sj} > I_{sj}^{(1)} + \sum_{l=N+1}^{\infty} \left(f_{1l}^{(1)} \sum_{k=0}^{m_l} \frac{f_{3kl}^{(1)}}{l!} \right).$$

Оскільки $f_{3kl}^{(1)}$ збільшується зі збільшенням k , оцінка знизу матиме вигляд

$$I_{sj} > I_{sj}^{(1)} + s_N N! \sum_{l=N+1}^{\infty} \frac{f_{1l}^{(1)}}{l!}, s_N = \sum_{k=0}^{m_l} f_{3kl}^{(1)}.$$

Надалі, використаємо оцінку $\sum_{l=N+1}^{\infty} \frac{f_{1l}^{(1)}}{l!}$ у формі

$$\sum_{l=N+1}^{\infty} \frac{f_{1l}^{(1)}}{l!} = t_{s+1} \sum_{l=N+1}^{\infty} \frac{t_{s+1}^l}{l!} - t_s \sum_{l=N+1}^{\infty} \frac{t_s^l}{l!}.$$

Оскільки $\sum_{l=N+1}^{\infty} \frac{t^l}{l!}$ – залишок ряду Тейлора для e^t , оцінимо його як $e^t \frac{t^{N+1}}{(N+1)!}$

отримуючи

$$\sum_{l=N+1}^{\infty} \frac{f_{1l}^{(1)}}{l!} \approx \frac{1}{(N+1)!} \left(e^{t_{s+1}} t_{s+1}^{N+2} - e^{t_s} t_s^{N+2} \right)$$

та наступне наближення нижньої оцінки похибки усічення ряду (6.77):

$$\varepsilon_3(N) = I_{sj} - I_{sj}^{(1)} \approx \frac{s_N}{N+1} \left(e^{t_{s+1}} t_{s+1}^{N+2} - e^{t_s} t_s^{N+2} \right). \quad (6.82)$$

□

Маючи (6.82), оцінимо кількість N_1 членів ряду стосовно l , необхідних для отримання похибки $\varepsilon_3(N_1) = \varepsilon$ обчислення інтегралів I_{sj} згідно з усіченим рядом (6.77). Припускаючи, що $t_s^{N+2} \approx t_{s+1}^{N+2}$ для достатньо великих t_s , маємо

$$\varepsilon \approx \frac{t_{s+1}^{N_1+1}}{N_1+1} R_1 = \frac{e^{(N_1+1)\ln t_{s+1}}}{(N_1+1)\ln t_{s+1}} R_1 \ln t_{s+1}, R_1 = s_N \left(e^{t_{s+1}} t_{s+1} - e^{t_s} t_s \right).$$

Виконавши заміну змінних $M = (N_1+1)\ln t_{s+1}$ отримуємо

$$a(\varepsilon)M \approx e^M, a(\varepsilon) = \varepsilon R_1 \ln t_{s+1} \Rightarrow M = -W\left(-\frac{1}{a(\varepsilon)}\right) \Rightarrow N_1 \approx -\frac{1}{\ln t_{s+1}} W\left(-\frac{1}{a(\varepsilon)}\right) - 1,$$

де W – функція Ламберта [86].

Для $a(\varepsilon) < 0$, використовуючи наближення W у формі $W(z) \approx \ln(z) - \ln(\ln(z))$ [86], отримуємо

$$N_1(\varepsilon) \approx -\frac{1}{\ln t_{s+1}} \left(\ln \left(-\frac{1}{a(\varepsilon)} \right) - \ln \left(\ln \left(-\frac{1}{a(\varepsilon)} \right) \right) \right) - 1. \quad (6.83)$$

Теорема 6.6. Для часу виконання обчислень на основі усіченого ряду (6.77) справедлива оцінка

$$T^{(1)}(N) = O(N^2 + kN \ln t_j), \quad k = -1, \alpha > 0.5; \quad k = 1, \alpha \leq 0.5.$$

Доведення.

Час, необхідний для проведення підсумовування усіченого ряду (6.77), дорівнює

$$T^{(1)}(N) = \sum_{l=0}^N k_1^{(1)} m_l,$$

де $k_1^{(1)}$ – час, необхідний для обчислення $f_{1l}^{(1)} f_{2kl}^{(1)}$, припускаючи його сталим.

Оцінку значення m_l отримаємо відштовхуючись від його визначення. З умови

$|f_{2km_l}^{(1)}| \leq \frac{\varepsilon_2}{l!}$ для $\alpha m_l > l$ отримуємо

$$\left(\frac{\alpha}{1-\alpha} \right)^{m_l} \frac{(\alpha m_l)_l}{\Gamma(\alpha m_l + 1)(l+1)} t_j^{\alpha m_l - l} = \varepsilon_2.$$

Оскільки $\frac{\lfloor \alpha m_l \rfloor!}{(\lfloor \alpha m_l \rfloor - l)!} \leq (\alpha m_l)_l \leq \frac{(\lfloor \alpha m_l \rfloor + 1)!}{(\lfloor \alpha m_l \rfloor + 1 - l)!}$ і

$$(\lfloor \alpha m_l \rfloor + 1)! \geq \Gamma(\alpha m_l + 1) \geq \lfloor \alpha m_l \rfloor!,$$

де $\lfloor x \rfloor$ – ціла частина числа, маємо

$$\frac{1}{(\lfloor \alpha m_l \rfloor + 1)(\lfloor \alpha m_l \rfloor - l)!} \leq \frac{(\alpha m_l)_l}{\Gamma(\alpha m_l + 1)} \leq \frac{\lfloor \alpha m_l \rfloor + 1}{(\lfloor \alpha m_l \rfloor + 1 - l)!}.$$

Оскільки $e \left(\frac{n}{e} \right)^n \leq n! \leq e \left(\frac{n+1}{e} \right)^{n+1}$, виконуються нерівності

$$\left(\frac{\alpha}{1-\alpha} \right)^{m_l} \frac{\lfloor \alpha m_l \rfloor + 1}{e(l+1)} \left(\frac{e}{\lfloor \alpha m_l \rfloor + 1 - l} \right)^{\lfloor \alpha m_l \rfloor + 1 - l} t_j^{\alpha m_l - l} \geq \varepsilon_2,$$

$$\left(\frac{\alpha}{1-\alpha}\right)^{m_l} \frac{1}{([\alpha m_l] + 1)e(l+1)} \left(\frac{e}{[\alpha m_l] + 1 - l}\right)^{[\alpha m_l] + 1 - l} t_j^{\alpha m_l - l} \leq \varepsilon_2.$$

Підсилюючи умову $\alpha m_l > l$ у вигляді $[\alpha m_l] > l$ маємо

$$\frac{[\alpha m_l] + 1}{(l+1)} \leq [\alpha m_l] + 1 - l$$

i

$$\left(\frac{\alpha}{1-\alpha}\right)^{m_l} \left(\frac{e}{[\alpha m_l] + 1 - l}\right)^{[\alpha m_l] - l} t_j^{\alpha m_l - l} \geq \varepsilon_2,$$

$$\frac{\left(\frac{\alpha}{1-\alpha}\right)^{m_l}}{e^2(l+1)^2} \left(\frac{e}{[\alpha m_l] + 1 - l}\right)^{[\alpha m_l] + 2 - l} t_j^{\alpha m_l - l} \leq \varepsilon_2.$$

Розглянемо випадок $\alpha > 0,5$, $\frac{\alpha}{1-\alpha} > 1$.

Оскільки у цьому випадку $\left(\frac{\alpha}{1-\alpha}\right)^{m_l} > 1$ і $[\alpha m_l] + 1 - l < [\alpha m_l] + 2 - l$, маємо

$$\left(\frac{e}{[\alpha m_l] + 2 - l}\right)^{[\alpha m_l] + 2 - l} t_j^{\alpha m_l - l} \leq \varepsilon_2 e^2 (l+1)^2. \text{ При } \frac{e}{[\alpha m_l] + 2 - l} \leq 1 \text{ наступна нерівність}$$

$$\text{виконується оскільки } \alpha m_l + 2 - l \geq [\alpha m_l] + 2 - l: \left(\frac{t_j e}{\alpha m_l + 2 - l}\right)^{\alpha m_l + 2 - l} \leq t_j^2 \varepsilon_2 e^2 (l+1)^2.$$

Виконуючи заміну змінних $\frac{t_j e}{\alpha m_l + 2 - l} = w = e^u$ отримуємо

$$w \approx e^{-w \left(\frac{2 \ln t_j + \ln \varepsilon_2 + 2 + 2 \ln(l+1)}{t_j e} \right)}.$$

Подальша апроксимація $W(x)$ у формі $W(x) \approx \ln(x)$ призводить до

$$\alpha m_l + 2 - l \approx -2 \ln t_j - \ln \varepsilon_2 - 2 - 2 \ln(l+1) \Rightarrow m_l \approx O(l - \ln t_j).$$

Таким чином, не беручи до уваги залежність від α , оцінка часу розрахунків для $\alpha > 0.5$ має вигляд

$$T^{(11)}(N) = O(N^2 - N \ln t_j). \quad (6.84)$$

Розглянемо випадок $\alpha \leq 0,5$, $\frac{\alpha}{1-\alpha} \leq 1$.

Оскільки у цьому випадку $\left(\frac{\alpha}{1-\alpha}\right)^{m_l} \leq 1$; $\lfloor \alpha m_l \rfloor + 1 - l > \lfloor \alpha m_l \rfloor - l$; і

$$t_j^{\alpha m_l - l} \leq t_j^{\lfloor \alpha m_l \rfloor + 1 - l}, \text{ маємо } \left(\frac{et_j}{\lfloor \alpha m_l \rfloor - l}\right)^{\lfloor \alpha m_l \rfloor - l} \geq t^{-1} \varepsilon_2.$$

Аналогічно підходу, використаному для $\alpha < 0.5$, використовуючи наближення $W(x)$ у формі $W(x) \approx \ln(x)$ отримуємо

$$\lfloor \alpha m_l \rfloor - l \approx \ln t_j - \ln \varepsilon_2 \Rightarrow m_l \approx O(l + \ln t_j).$$

Відповідно, оцінка часу розрахунків у випадку $\alpha \leq 0.5$ має вигляд

$$T^{(12)}(N) = O(N^2 + N \ln t_j). \quad (6.85)$$

Поєднуючи (6.84) та (6.85), загальна оцінка часу розрахунків має порядок

$$T^{(1)}(N) = O(N^2 + kN \ln t_j), \quad k = -1, \alpha > 0.5; \quad k = 1, \alpha \leq 0.5. \quad (6.86)$$

□

Оцінки точності та швидкодії для випадку представлення у вигляді інтегралу

Теорема 6.7. Для похибки наближення інтегралу (6.79) формулою (6.80) справедлива оцінка

$$\varepsilon_5(N) \leq \frac{1}{N} \varepsilon_6 + R_C, \quad R_C \leq f_2^{(2)}(C) \left(\operatorname{atan} \left(\frac{p \cos(\pi\alpha) - C}{p \sin(\pi\alpha)} \right) + \frac{\pi}{2} \right).$$

$$\begin{aligned} \varepsilon_6 = & \frac{3}{2} (v_m - nv_{inf}) \left(f_3^{(2)}(0) + f_3^{(2)}(v_m - nv_{inf}) \right) + \\ & + 3nv_{inf} \left(2f_{max}^{(2)} - f_3^{(2)}(v_m - nv_{inf}) - f_3^{(2)}(v_m + nv_{inf}) \right) + \\ & + \left(C - (v_m + nv_{inf}) \right) \left(f_3^{(2)}(v_m + nv_{inf}) - f_3^{(2)}(C) \right). \end{aligned}$$

$$\text{де } f_1^{(2)}(v) = \frac{p \sin(\pi\alpha)}{v^2 - 2vp \cos(\pi\alpha) + p^2}, \quad f_2^{(2)}(v) = f_{2a}^{(2)}(v) f_{2b}^{(2)}(v), \quad f_{2a}^{(2)}(v) = v^{-1/\alpha} e^{-v^{1/\alpha} t_j},$$

$$f_{2b}^{(2)}(v) = e^{v^{1/\alpha} t_s} - e^{v^{1/\alpha} t_{s+1}}.$$

Доведення.

Оскільки перша похідна $f_2^{(2)}(v)$ має сингулярність при $v \rightarrow 0$, використовуватимемо оцінку точності для формули прямокутників у формі $\left| (b-a)f\left(\frac{b-a}{2}\right) \right| \leq \frac{b-a}{2} \int_a^b |f'(v)| dv$ [89]. Оскільки $f_2^{(2)}(v)$ зменшується для досить великих v , припустимо, що інтегрована функція $f_3^{(2)}(v) = f_1^{(2)}(v)f_2^{(2)}(v)$ має один мінімум $f_{min}^{(2)} \geq 0$ в діапазоні $[0, v_m - nv_{inf}]$ і один максимум $f_{max}^{(2)} \leq f_1^{(2)}(v_m)f_2^{(2)}(v_m - nv_{inf})$ в діапазоні $[v_m - nv_{inf}, v_m + nv_{inf}]$. При $v > v_m + nv_{inf}$ функція зменшується. Тоді маємо

$$\begin{aligned} \int_0^{v_m - nv_{inf}} |f_3^{(2)}(v)| dv &\leq (f_3^{(2)}(0) - 2f_{min}^{(2)} + f_3^{(2)}(v_m - nv_{inf})) \leq \\ &\leq (f_3^{(2)}(0) + f_3^{(2)}(v_m - nv_{inf})), \\ \int_{v_m - nv_{inf}}^{v_m + nv_{inf}} |f_3^{(2)}(v)| dv &\leq (2f_{max}^{(2)} - f_3^{(2)}(v_m - nv_{inf}) - f_3^{(2)}(v_m + nv_{inf})), \\ \int_{v_m + nv_{inf}}^C |f_3^{(2)}(v)| dv &\leq (f_3^{(2)}(v_m + nv_{inf}) - f_3^{(2)}(C)). \end{aligned}$$

Похибку, спричинену усіченням верхньої межі інтегрування, можна оцінити, беручи до уваги, що $f_2^{(2)}(v)$ зменшується при $v > v_m + nv_{inf}$. Звідси маємо

$$\begin{aligned} \int_0^\infty f_3^{(2)}(v) dv &= \int_0^C f_3^{(2)}(v) dv + R_C, R_C = \int_C^\infty f_3^{(2)}(v) dv \leq f_2^{(2)}(C) \int_C^\infty f_1^{(2)}(v) dv, \\ R_C &\leq f_2^{(2)}(C) \left(\operatorname{atan} \left(\frac{p \cos(\pi\alpha) - C}{p \sin(\pi\alpha)} \right) + \frac{\pi}{2} \right). \end{aligned}$$

Загальна похибка матиме вигляд

$$\begin{aligned} \varepsilon_5(N) &\leq \frac{1}{N} \varepsilon_6 + R_C, \\ \varepsilon_6 &= \frac{3}{2} (v_m - nv_{inf}) \left(f_3^{(2)}(0) + f_3^{(2)}(v_m - nv_{inf}) \right) + \end{aligned} \quad (6.87)$$

$$+3nv_{inf} \left(2f_{max}^{(2)} - f_3^{(2)}(v_m - nv_{inf}) - f_3^{(2)}(v_m + nv_{inf}) \right) + \\ + \left(C - (v_m + nv_{inf}) \right) \left(f_3^{(2)}(v_m + nv_{inf}) - f_3^{(2)}(C) \right).$$

□

Наслідок 1. Оцінка кількості підрозбиттів, необхідних для отримання похибки $\varepsilon_5(N) = \varepsilon$, має вигляд

$$N_2(\varepsilon) \leq \frac{\varepsilon_6}{\varepsilon - R_C}. \quad (6.88)$$

Твердження 6.8. Час, необхідний для виконання обчислень згідно з (6.80), має порядок

$$T^{(2)}(N) = O(N). \quad (6.89)$$

Поєднання алгоритму апроксимації похідної зі скінченно-різницевою схемою для рівняння конвективної дифузії

Будемо розглядати рівняння конвективної дифузії у вигляді [231]

$${}^{ABC}D_t^\alpha C(x,t) = \frac{\partial^2 C(x,t)}{\partial x^2} - \frac{\partial C(x,t)}{\partial x} + f(x,t), \quad 0 \leq x \leq 1, t \geq 0 \quad (6.90)$$

з початковими та граничними умовами

$$C(0,t) = C(1,t) = C(x,0) = 0.$$

Необхідність постановки нульових початкових умов для рівняння (6.90) витікає з властивостей похідної Атангана–Балеану [206]. Однак навіть з таким обмеженням рівняння (6.90) дозволяє описувати такі процеси масоперенесення, як забруднення ґрунту після аварійної ситуації, при якій речовина раптово починає дифундувати із зовнішнього джерела.

Вводячи до розгляду сіткову область

$$\omega_{h\tau} = \left\{ (x_i, t_j) : x_i = ih \left(i = \overline{0, m} \right), t_j = j\tau \left(j = \overline{0, n} \right) \right\}, \quad h = \frac{1}{m}, \tau = \frac{T}{n},$$

дискретизуємо похідні за просторовою змінною як [192, 313]

$$\left. \frac{\partial^2 C(x,t)}{\partial x^2} \right|_{x=x_i, t=t_j} = \frac{C_{i+1,j} - 2C_{i,j} + C_{i-1,j}}{h^2} + O(h^2),$$

$$\left. \frac{\partial C(x,t)}{\partial x} \right|_{x=x_i, t=t_j} = \frac{C_{i+1,j} - C_{i-1,j}}{2h} + O(h^2),$$

$$C_{i,j} = C(x_i, t_j).$$

Позначаючи значення нелокальної частини похідної (6.74) у точці (x_i, t_j) як

$$\bar{\Delta}_{i,j}^\alpha C(x,t) = \frac{1}{1-\alpha} \sum_{s=0}^{j-2} \frac{C(x_i, t_{s+1}) - C(x_i, t_s)}{\tau} I_{sj}, \quad \text{рівняння (6.90) на кроці за часом}$$

$j = \overline{1, n+1}$ може бути дискретизованим у вигляді наступної тридіагональної СЛАР

$$\begin{aligned} & \left(-\frac{1}{h^2} - \frac{1}{2h} \right) C_{i-1,j} + \left(\frac{2}{h^2} + \frac{I_{j-1,j}}{1-\alpha} \right) C_{i,j} + \left(-\frac{1}{h^2} + \frac{1}{2h} \right) C_{i+1,j} = \\ & = I_{j-1,j} C_{i,j-1} + f_{i,j} - \bar{\Delta}_{i,j}^\alpha, \quad i = \overline{1, m-1}, \\ & C_{0,j} = C_{m,j} = 0, \\ & f_{i,j} = f(x_i, t_j). \end{aligned} \quad (6.91)$$

Зауважимо, що система (6.91) може бути ефективно розв'язана методом прогонки [313]. Порядок точності вищеописаної схеми не більший за $O(\tau + h^2)$.

Алгоритм розрахунку значень нелокальної частини $\bar{\Delta}_{i,j}^\alpha$ похідної (6.74)

наступний. Будемо використовувати схему (6.81) з $f_{1l} = f_{1l}^{(1)}$, $f_{2l} = \sum_{k=0}^{m_l} f_{2kl}^{(1)}$

при $\left| -\frac{\alpha}{1-\alpha} t_j^\alpha \right| < 1$ (представлення у вигляді усіченого ряду згідно з (6.77)) та

з $f_{1l} = f_{2b}^{(2)}$, $f_{2l} = f_1^{(2)} f_{2a}^{(2)}$ (наближення представлення у вигляді інтегралу згідно з (6.80)) для більших значень t_j .

Для кожного вузла сітки будемо зберігати значення $S_{l,j-1}$, $l = 1, \dots, N$ для обох

представлень і модифікувати їх обох при $\left| -\frac{\alpha}{1-\alpha} t^\alpha \right| < 1$. Для більших значень t_j

продовжуватимемо модифікувати лише значення $S_{l,j-1}$ для наближення (6.80)

представлення у вигляді інтегралу.

Складність обчислення значень $S_{l,j-1}$, $l=1,\dots,N$ для обох представлень, як і складність обчислень значень $f_1^{(2)} f_{2a}^{(2)}$, є сталою. У випадку $\sum_{k=0}^{m_l} f_{2kl}^{(1)}$ маємо обчислювальну складність порядку $O(m_l)$. Таким чином, при $\left| -\frac{\alpha}{1-\alpha} t^\alpha \right| < 1$ обчислення за (6.81) мають той самий порядок складності, що і значення $T^{(1)}$, визначене згідно з (6.86), а для вищих значень t_j – такий же самий порядок як і значення $T^{(2)}$, визначене згідно з (6.89).

У порівнянні з традиційними схемами апроксимації (наприклад, [19]), які мають лінійну обчислювальну складність відносно номеру j кроку за часом, складність запропонованого алгоритму залежить лише від $\ln t_j$ при $\left| -\frac{\alpha}{1-\alpha} t_j^\alpha \right| < 1$ і не залежить від j для вищих значень t_j .

Особливості програмної реалізації

Під час виконання програмної реалізації запропонованого алгоритму виникає декілька проблем, пов'язаних із переповненням типів даних із рухомою комою.

Виконуючи обчислення згідно з (6.77), складові $f_{2kl}^{(1)}$ можуть переповнювати тип даних подвійної точності, тоді як загальне значення $f_{2kl}^{(1)}$ залишається в межах діапазону чисел, що може бути представлений цим типом. Для вирішення цієї проблеми, пропонується представляти $f_{2kl}^{(1)}$ у вигляді експоненти суми логарифмів його складових. Варто зазначити, що в цьому випадку надзвичайно важливо для загальної швидкодії алгоритму використовувати за можливості кешування обчислених значень логарифмів.

У випадку апроксимації (6.80) представлення у вигляді інтегралу, подібний підхід необхідно застосовувати для обчислення як $S_{l,j-1}$ так і $f_1^{(2)} f_{2a}^{(2)}$. Оскільки $f_1^{(2)} f_{2a}^{(2)}$, як правило, прямує до 0, будемо зберігати його знак та значення його логарифму в ситуації, коли значення $f_1^{(2)} f_{2a}^{(2)}$, представлене у типі даних подвійної точності, дорівнює 0. Додатково будемо зберігати прапорець із

зазначенням, у якій формі, абсолютній чи логарифмічній, зберігається значення $f_1^{(2)} f_{2a}^{(2)}$. В процесі модифікації значень $S_{l,j-1}$ переходитимемо до їх зберігання у формі логарифму, встановлюючи відповідний прапорець, у ситуації коли вони переповнюють тип даних. Збільшуючи надалі значення $S_{l,j-1}$, збережене як його логарифм, виходячи з припущення, що $S_{l,j-1} \gg S_{l,j-2}$, нехтуємо попередніми даними. Залежно від значень прапорців, які визначають форму зберігання змінних, остаточне обчислення згідно з (6.81) виконується 4 різними способами, зберігаючи максимально можливу точність.

Тестування оцінок точності та ефективності обчислення значень I_{sj}

Для перевірки отриманих оцінок похибок (6.82), (6.87) була проведена серія обчислювальних експериментів по оцінці значень I_{sj} використовуючи усічений ряд (6.77) та наближення (6.80) представлення у формі інтегралу. Вихідні коди розробленого програмного забезпечення наведено у [265]. Таким чином отримані значення I_{sj} порівнювались зі значеннями, розрахованими шляхом чисельного інтегрування на основі алгоритму рекурсивного підрозбиття [180]. Для цього значення t_s варіювались в діапазоні $t_s = 0; 0,1; \dots; 2,9$ для $t_j = 3$ і $N = 25, 50, \dots, 225$, $\alpha = 0,8$; $\varepsilon_1 = \varepsilon_2 = 10^{-10}$. У подальшому, для $N = 100$ варіювались $\varepsilon_1 = \varepsilon_2 = 10^{-5}, 10^{-6}, \dots, 10^{-15}$. У наступній серії експериментів значення α варіювались в діапазоні $\alpha = 0,01; 0,1; 0,2, \dots, 0,9; 0,99$ для $N = 100$, $\varepsilon_1 = \varepsilon_2 = 10^{-10}$. Наостанок, значення t_j варіювалось у діапазоні від 0,1 до 0,9 з кроком 0,1 для $\alpha = 0,8$ та $t_s < t_j$.

В обчислювальних експериментах оцінка (6.82) була наближенням знизу, а точність її зміни при зміні значень параметрів зростала зі зменшенням точності розрахунку I_{sj} . Верхня оцінка (6.87) адекватно відображала зміну значень параметрів, проте її залежність від фактичних абсолютних різниць була близькою до експоненціальної.

Обидві оцінки значно занижували або завищували похибку. Для оцінки впливу такого заниження чи завищення на оцінку необхідної кількості N доданків у сумах, у всіх проведених експериментах оцінювались $N_1(\varepsilon)$, $N_2(\varepsilon)$ згідно з (6.83), (6.88) для $\varepsilon = \varepsilon_7$, де ε_7 – абсолютні різниці між значеннями I_{sj} , обчисленими за допомогою (6.77) або (6.80), та значеннями, отриманими чисельним інтегруванням з рекурсивним підрозбиттям. Надалі, отримані значення порівнювались зі значеннями N , що використовувались при обчисленнях I_{sj} . Так, оцінка (6.83) генерувала значення, що були від 10 разів більше до 100 разів менше фактичних N . Оцінка (6.88) генерувала значення, що були від 2 разів меншими до 10^8 разів більшими, ніж фактична N . Таким чином, (6.83) та (6.88) можуть використовуватись лише для оцінки загальних тенденцій у зміні N , а не для підтримки прийняття рішень під час практичного моделювання.

Час, необхідний для виконання розрахунків згідно з (6.77), квадратично збільшувався ($R^2 = 0,99$) із збільшенням N для фіксованого $t_j = 3$ і лінійно зростав зі збільшенням $\ln t_j$ при фіксованих значеннях інших параметрів, що відповідає оцінці (6.86). Він також лінійно збільшувався зі збільшенням α ($R^2 = 0,96$).

Відповідно до оцінки (6.89), вимірний час, необхідний для виконання розрахунків згідно з (6.80), лінійно збільшувався зі збільшенням N ($R^2 = 0,93$) і незначним чином змінювався при зміні значень інших параметрів.

Тестування точності обчислення значень похідної Атангана–Балеану

Щоб перевірити точність загальної процедури розрахунку значень похідної Атангана–Балеану, будемо обчислювати значення ${}^{ABC}D_t^\alpha f(t)$ для $f(t) = t^\beta$, $0 < \beta \leq 1$ згідно з нею, порівнюючи результати зі значеннями, обчисленими з використанням чисельного інтегрування з рекурсивним підрозбиттям [180]. Отримані наближення значень похідної для $\varepsilon_1 = \varepsilon_2 = 10^{-10}$, $N = 100$, $\tau = 0,01$; при змінні α , β і моменту часу t представлені в табл. А.1.

Відносні похибки порівнюючи значення, отримані за запропонованою процедурою, та з використанням чисельного інтегрування з рекурсивним

підрозбиттям були $>10\%$ лише у випадках $\beta < 0,5$; $\alpha \leq 0,9$; $t \leq 0,5$. Відносні похибки зменшувались зі збільшенням t , β , і α . Додаткове збільшення кількості N доданків у пропонованих апроксимаціях зменшувало похибку.

Точність та швидкодія при розв'язанні дробово-диференціального рівняння конвективної дифузії

З метою тестування розглянемо три випадки функції $f(x,t) = f_k(x,t), k = 1, 2, 3$ у рівнянні (6.90), які були досліджені раніше у [231] та для яких рівняння має наступні аналітичні розв'язки $C_k(x,t)$:

$$C_1(x,t) = t^4, f_1(x,t) = \frac{24t^4}{1-\alpha} E_{\alpha,5} \left(-\frac{\alpha}{1-\alpha} t^\alpha \right),$$

$$C_2(x,t) = x(x-1)t^2, f_2(x,t) = \frac{2x(x-1)t^2}{1-\alpha} E_{\alpha,3} \left(-\frac{\alpha}{1-\alpha} t^\alpha \right) + (2x-3)t^2,$$

$$C_3(x,t) = \sin(\pi x)t^5,$$

$$f_3(x,t) = \frac{120\sin(\pi x)t^5}{1-\alpha} E_{\alpha,6} \left(-\frac{\alpha}{1-\alpha} t^\alpha \right) + \pi t^5 (\pi \sin(\pi x) + \cos(\pi x)).$$

Слідуючи [231], розглядатимемо оцінку похибки чисельного розв'язку \bar{C} в залежності від значення параметра сітки M у формі

$$E_{k,n}(M) = \frac{1}{nm} \sum_{j=0}^n \sum_{i=0}^m (\bar{C}_{i,j} - C_k(x_i, t_j))^2.$$

Тоді швидкість збіжності при послідовному моделюванні зі збільшенням $M = M_1, M_2, \dots, M_l$ можна оцінити як [231]

$$R_{k,n}(l) = \frac{\log_{10}(E_{k,n}(M_{l-1}) / E_{k,n}(M_l))}{\log_{10}(M_l / M_{l-1})}.$$

У першій серії обчислювальних експериментів порівнювалась точність та швидкість збіжності при використанні пропонованої схеми та схеми першого порядку точності, наведеної у [231]. Отримані результати (деякі з них представлені в табл. А.2 при зміні $M = n = t$ та в табл. А.3 для $m = 1000$ і зміні $M = n$, $T = 1$, $N = 1000$, $\varepsilon_2 = 10^{-10}$) узгоджуються з результатами, наведеними у [231], для

випадку $\alpha \leq 0,5$ (див. табл. А.4, А.5 для випадку $\alpha = 0,5$) і демонструють збільшення $R_{k,n}$ зі збільшенням n . Для $\alpha \leq 0,5$ використовувалося лише представлення (6.77) у вигляді усіченого ряду. При $\alpha > 0,5$, схема (6.77) використовувалась на початкових етапах моделювання, а наближення (6.80) представлення у вигляді інтегралу — при збільшені t . Попри те, що значення параметра N наближених схем було однаковим для них обох, були отримані різні порядки похибок усічення та швидкостей збіжності. Результати обчислювальних експериментів показали, що похибка схеми (6.80) у цьому випадку більша, ніж для схеми (6.77), що призводить до нижчих швидкостей збіжності пропонованого алгоритму для $\alpha > 0,5$.

Загальний час проведення обчислень (табл. А.2, А.3) суттєво не залежав від форми розв'язків зменшуючись зі збільшенням порядку α похідної.

У другій серії обчислювальних експериментів вивчалась швидкодія пропонованого алгоритму, порівнюючи її з базовим випадком, коли значення похідної Атангана–Балеану обчислювались згідно з (6.75) із наближенням $\frac{\partial C}{\partial t}$ першого порядку точності при обчисленні інтегралів (6.76) з використанням чисельного інтегрування з рекурсивним підрозбиттям [180] і базовою квадратурною формулою методу прямокутників та обчисленні значень функції Міттаг-Леффлера за допомогою алгоритму, подібного до описаного в [169].

Розглядався випадок $f(x,t) = f_2(x,t)$, $m = 100$, $n = 6000$, $T = 1$, $N = 50$ і вимірювався накопичений час, витрачений на отримання розв'язків для $t = i\tau$, де i — номер кроку за часом (рис. 6.29 для $\alpha = 0,6$); оцінка похибки $E_{2,i}$ (рис. 6.30 для $\alpha = 0,6$); і максимальне число $mL(i) = \max m_l(i), l = \overline{0, N-1}$ членів внутрішньої суми в усіченому ряді (6.77) (рис. 6.31).

Виміряні значення часу проведення розрахунків (рис. 6.29) підтверджують, що пропонований алгоритм знижує загальну обчислювальну складність процедури моделювання від другого порядку для випадку базового алгоритму до порядку близького до першого. Однак, таке зниження дозволило прискорити обчислення у

розглянутому тестовому випадку лише при виконанні обчислень для більше ніж ~ 5000 кроків за часом.

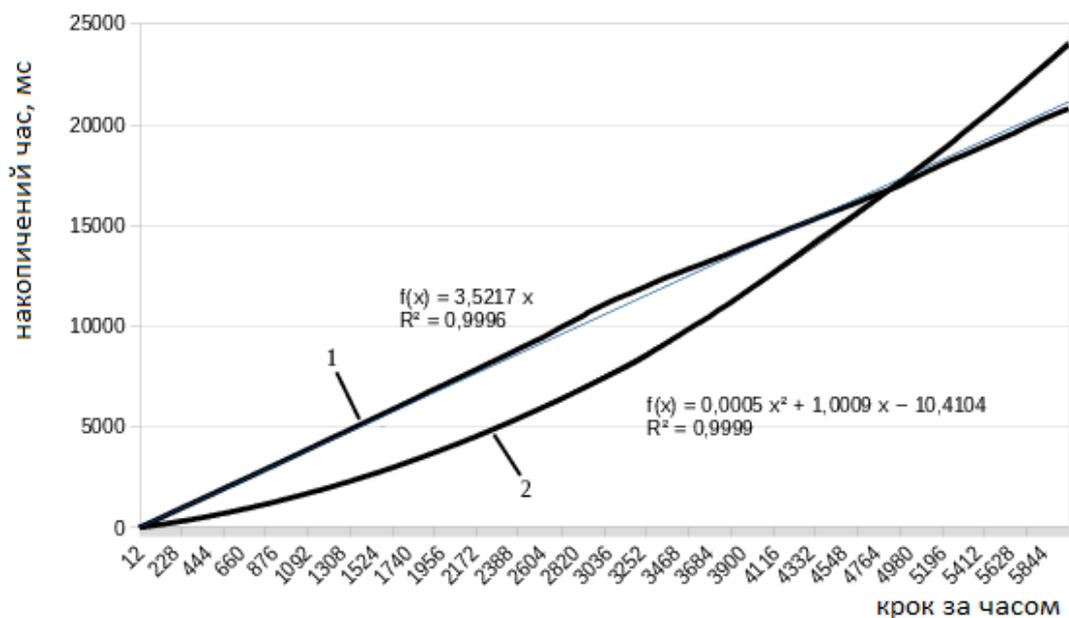


Рисунок 6.29 – Накопичений час, витрачений на отримання розв’язку для $t = i\tau$ в залежності від номера кроку за часом i при $\alpha = 0,6$ для пропонованого (крива 1) та базового алгоритмів (крива 2)

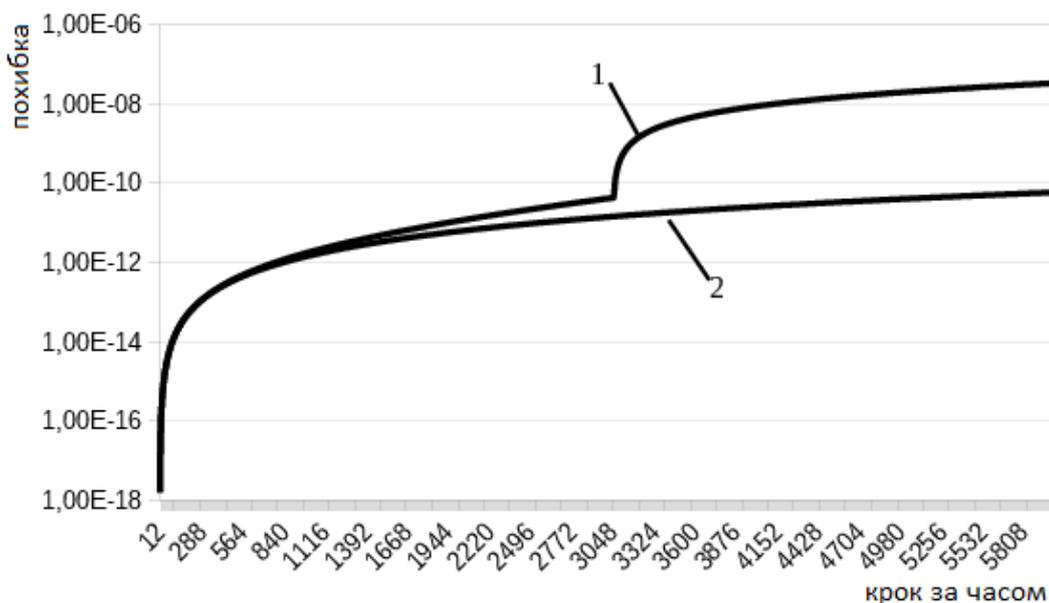


Рисунок 6.30 – Оцінка похибки $E_{2,i}$ в залежності від номера кроку за часом i при $\alpha = 0,6$ для пропонованого (крива 1) та базового алгоритмів (крива 2)

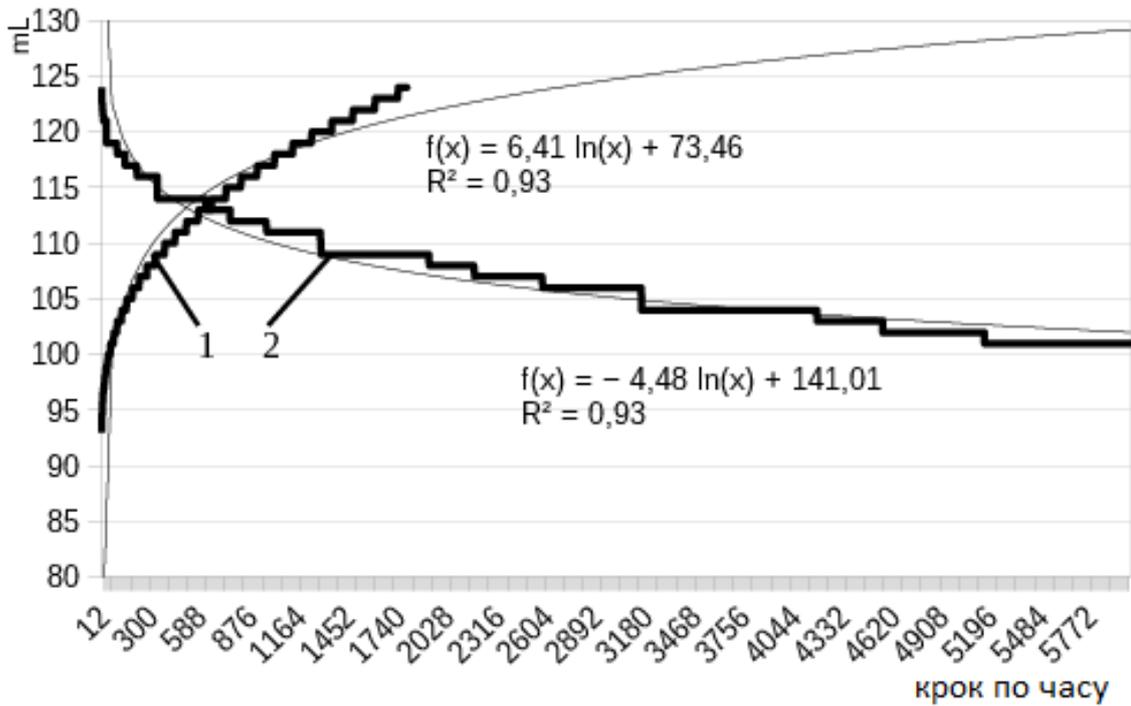


Рисунок 6.31 – Максимальне число $mL(i)$ доданків внутрішньої суми в усіченому ряді (6.77) в залежності від номера кроку за часом i при $\alpha = 0,7$ (крива 1) та $\alpha = 0,4$ (крива 2)

Дані, наведені на рис. 6.30 демонструють, що при однаковій кількості доданків, наближення (6.80) має меншу точність у порівнянні з наближенням (6.77), і їх заміна на ітераціях моделювання швидко призводить до зниження точності на 2 порядки. З іншої сторони, відносні різниці $A(i) = \frac{E_a(i) - E_b(i)}{E_b(i)}$ (E_a – похибка пропонованого алгоритму, E_b – похибка базового алгоритму) на кроці за часом i збільшуються з порядком $O(i^2)$ у випадку схеми (6.77) та з порядком $O(i^{0,43})$ для схеми (6.80), демонструючи, що остання має нижчий рівень накопичення помилок.

Отримані максимальні кількості доданків внутрішньої суми у усіченому ряді (6.77), наведені на рис. 6.31, підтверджують їх оцінки (6.84), та (6.85) для випадків, відповідно, $\alpha \leq 0,5$ та $\alpha > 0,5$. Одержані результати підтверджують, що при $\alpha > 0,5$ час роботи алгоритму може повільно збільшуватися зі збільшенням

числа кроків за часом, тоді як для $\alpha \leq 0,5$ може спостерігатись незначне його зниження.

6.7 Висновки до розділу 6

Розділ 6 присвячено обчислювальним схемам для дробово-диференціальних рівнянь, що містять ψ -похідну Капуто, яка узагальнює класичну похідну Капуто–Герасимова та містить додатковий функціональний параметр, а також похідну Анатгана–Балеану.

Сингулярні інтеграли, що виникають при дискретизації ψ -похідної Капуто, мають бути чисельно розраховані при розв'язанні початково-крайових задач для диференціальних рівнянь, що її містять. Оскільки швидкодія ефективних у цьому випадку квадратурних формул рекурсивного підрозбиття є низькою, у п. 6.1 були запропоновані схеми меншого порядку обчислювальної складності, базовані на представленні відповідних інтегралів у вигляді ряду. Проведені обчислювальні експерименти показали, що використання таких схем дозволяє досягти 2,5-кратного прискорення при моделюванні аномального процесу дифузії.

Час, що витрачається на розрахунок значень дробових похідних, складає $>60\%$ від загального часу моделювання, причому цей відсоток збільшується зі збільшенням номеру кроку за часом. У зв'язку з цим, для одержання додаткового прискорення у п. 6.2 була запропонована серія GPU алгоритмів, що використовують можливості низькорівневої оптимізації коду для графічних прискорювачів. Зокрема, використовуються векторні арифметичні операції, типи даних різної точності, тензорні операції. Показано, що прискорення векторизованих алгоритмів у порівнянні з неекторизованими зростає зі збільшенням обсягу обчислень. Використання тензорних ядер, що виконують операції множення матриць представлених 16-бітовим типом даних, дозволило виконувати обчислення на 12% швидше, ніж при використанні 16-бітового типу даних без векторизації. Однак, у порівнянні з ефективністю найшвидшої векторизованої реалізації, яка використовує тип даних float4, було отримано лише 2% прискорення.

У випадку моделей, що містять ψ -похідні Капуто за просторовими змінними, запропонована методика представлення інтегралів у вигляді ряду дозволяє зменшити

обчислювальну складність виконання операцій множення матриць на вектори при розв'язанні ітераційними методами СЛАР, що виникають при скінченно-різницевій дискретизації відповідних початково-крайових задач. У п. 6.3 наведено результати теоретичного та експериментального аналізу точності та обчислювальної складності запропонованого алгоритму. Показано, що прискорення оптимізованого алгоритму, який виконує множення матриці на вектори, збільшується зі збільшенням розміру сітки та зменшенням порядку дробової похідної.

У п. 6.4 аналізується баланс точності та швидкодії скінченно-різницевих схем для тривимірного рівняння дифузії з ψ -похідними Капуто як за часом, так і за просторовими змінними, зокрема, при застосуванні схем оптимізації обчислень, описаних у п. 6.1, 6.3. Визначено, що неявна схема є очікувано, найточнішою (до 10 разів у проведених обчислювальних експериментах), проте також і найповільнішою (до 4 разів). Швидші схеми розщеплення, з іншої сторони, мають обмеження щодо своєї ефективності в контексті точності розв'язання. Проведені обчислювальні експерименти показують, що їх слід використовувати лише при наступних обмеженнях на порядки дробових похідних: $\alpha > 0,5; \beta > 0,5$. Схеми розщеплення на рівні дискретної задачі у порівнянні з схемами розщеплення на диференціальному рівні є більш точними, проте на $\sim 20\%$ повільнішими.

Порівняння використання методу фіксованої пам'яті та алгоритму, базованого на представленні інтегралів у вигляді ряду, для пришвидшення обчислень показує, що при апроксимації похідної за часом, при однаковій швидкодії менша втрата точності досягається при використанні алгоритму представлення інтегралів у вигляді ряду. Подібний алгоритм для дробових похідних за просторовими змінними ефективний лише для сіток великого розміру.

На основі отриманих оцінок точності та швидкодії у п. 6.4 представлений алгоритм автоматичного вибору оптимальної обчислювальної схеми, який дозволив для тестової задачі отримати 6-кратне прискорення розрахунків у порівнянні з неявною скінченно-різницевою схемою при до 10% втраті точності.

У п. 6.5 для випадку тривимірної моделі аномальної дифузії з ψ -похідними Капуто за просторовими змінними побудовані та досліджені паралельні алгоритми

для систем з розподіленою пам'яттю, які використовують дві схеми розподілу даних — одномірну блокову та червоно-чорну. Результати обчислювальних експериментів показали, що найбільш ефективною з точки зору швидкодії є локально-одновимірна схема розщеплення з одновимірним блоковим розподілом даних. Така схема має високу швидкість і масштабованість, які, однак, супроводжуються ~ 20 -кратною втратою точності у порівнянні з неявною схемою. Паралельна реалізація неявної схеми в експериментах із залученням до 48 процесорних ядер була більш ефективною при використанні червоно-чорного розподілу даних. Проте, при збільшенні кількості залучених обчислювальних ресурсів, теоретичні оцінки показують, що масштабованість алгоритму з блоковим розподілом вища.

Використання підходу розкладення у ряди з подальшим розділенням змінних було застосовано у п. 6.6 для обчислення значень функції Міттаг-Леффлера, що дозволило побудувати ефективний алгоритм обчислення значень похідної Атангана–Балеану. Отримані теоретичні та експериментальні оцінки його точності та швидкодії показують, що така процедура, як і у випадках похідної Капуто–Герасимова та ψ -похідної Капуто, дозволяє зменшити обчислювальну складність до близької до лінійної без значних втрат точності при розв'язанні дробово-диференціальних рівнянь, зокрема рівняння конвективної дифузії з дробовою похідною за часовою змінною.

Результати розділу 6 опубліковано в роботах [39, 40, 41, 42, 43, 45, 47, 55].

Джерела, що використані у розділі 6

Для написання даного розділу було використано 44 джерела [11], [18], [19], [39–47], [55], [62], [67], [69], [79], [86], [89], [129], [136], [142], [158], [163], [167], [169], [176], [180], [191], [192], [199], [204], [206], [219], [222], [227], [231], [248], [265], [276], [290], [311], [313], [317], посилання на які зазначені в тексті розділу.

Розділ 7. Алгоритми розв'язання задач ідентифікації параметрів дробово-диференціальних моделей вологоперенесення

Практичне застосування дробово-диференціальних моделей потребує проведення ідентифікації їх параметрів. Оскільки безпосередній фізичний зміст порядків та інших параметрів дробових похідних у більшості випадків є невідомим, актуальним є їх визначення на основі даних спостереження за модельованими процесами. При ідентифікації параметрів моделей на основі наявних спостережень, використання дробових похідних [11, 127, 139, 170, 235] збільшує кількість ступенів свободи, роблячи їх більш адаптивними. Специфіка таких задач зумовлює ефективність застосування метаевристичних алгоритмів для їх розв'язання.

Зазначимо, що важливим узагальненням поняття дробової похідної за Капуто є так звана ψ -похідна Капуто від функції за іншою функцією [11, 139, 170]. Нижче, при розгляді рівняння вологоперенесення, що містить такі похідні, застосовуватимемо, зокрема, алгоритм рою частинок [249] для ідентифікації параметрів моделі.

У випадку вологоперенесення у ґрунтах, визначення основної гідрофізичної характеристики (ОГХ) та залежностей коефіцієнту вологоперенесення від напорів може бути досить неточним у польових умовах, зокрема, при застосуванні у сільському господарстві. Через нелінійний характер параметрів моделі, визначення яких буває суттєво неточним, ефективним є вибір такої форми функціонального параметра ψ -похідної Капуто, яка мінімізує нев'язку між результатами моделювання та вимірними значеннями. Серед специфічних обчислювальних методів, розроблених для отримання символічних розв'язків подібних оптимізаційних задач, виділимо метод генетичного програмування [140].

7.1. Початково-крайова задача та чисельний метод

Будемо розглядати одновимірне дробово-диференціальне рівняння вологоперенесення, яке є узагальненням класичного рівняння Річардса, записаного у термінах напорів [36, 37].

Таке рівняння отримується з дробово-диференціального узагальнення закону збереження маси у формі [227]

$$\frac{\partial \theta}{\partial t} = -D_{z,g}^{(\alpha)} q,$$

де q (m^α / s) – потік фрактальної розмірності порядку α , що описується законом Дарсі у формі $q = -k(H) \frac{\partial H}{\partial z}$, $k(H)$ – гідравлічна провідність у фрактальній розмірності (m^α / s), θ – об'ємний вміст вологи (%), H – напір (m), $D_{z,g}^{(\alpha)}$ – ψ -похідна Капуто [11, 139] за просторовою змінною вигляду

$$D_z^{(\alpha)} H = \frac{1}{\Gamma(1-\alpha)} \int_0^z \frac{\partial H}{\partial \xi} (g_z(z) - g_z(\xi))^{-\alpha} d\xi, \quad 0 < \alpha \leq 1, \quad (7.1)$$

де $g_z(\xi)$ – допоміжна неперервно диференційовна на $[0, +\infty)$ функція. Зокрема, у випадку класичної похідної Капуто $g_z(\xi) \equiv \xi$.

Поєднуючи закон збереження маси та закон Дарсі з урахуванням основної гідрофізичної характеристики ґрунту $\theta(H)$ і врахувавши у правій частині рівняння функцію S екстракції вологи, яка моделює взаємодію між кореневою системою рослин та ґрунтом, отримуємо

$$\frac{\partial H}{\partial t} = C^{-1}(H) \left[D_z^{(\alpha)} \left(k(H) \frac{\partial H}{\partial z} \right) - S \right], \quad 0 \leq z \leq L, \quad t \geq 0, \quad (7.2)$$

де $C^{-1}(H) = \frac{\partial H}{\partial \theta}$.

Для врахування нелокальності дифузійного процесу за часовою змінною використаємо ψ -похідну Капуто $D_t^{(\beta)} H$ за часовою змінною вигляду

$$D_t^{(\beta)} H = \frac{1}{\Gamma(1-\beta)} \int_0^t \frac{\partial H}{\partial \tau} (g_t(t) - g_t(\tau))^{-\beta} d\tau, \quad 0 < \beta \leq 1, \quad (7.3)$$

і будемо розглядати коефіцієнт вологоперенесення $k(H)$ у фрактальній розмірності як за часом, так і за простором (m^α / s^β). Таким чином отримується наступне рівняння, яке збігається з (7.2) при $\beta = 1$ та з класичною формою рівняння Річардса при $\alpha = \beta = 1$:

$$D_t^{(\beta)} H = C^{-1}(H) \left[D_z^{(\alpha)} \left(k(H) \frac{\partial H}{\partial z} \right) - S \right], \quad 0 < z < L, \quad t > 0, \quad 0 < \alpha, \beta \leq 1. \quad (7.4)$$

Значення коефіцієнта $C^{-1}(H)$ обчислюється з використанням експериментально визначеної основної гідрофізичної характеристики (ОГХ), апроксимованої згідно з моделлю ван Генухтена [220].

На верхній границі $z = 0$, у випадку, коли ґрунт перебуває у вологонасиченому стані та вода накопичується на його поверхні, задається [218] крайова умова першого роду m , де F – задана функція. В інших ситуаціях задається крайова умова другого роду

$$k \frac{\partial H}{\partial z} = Q_e - Q_p - Q_i, \quad (7.5)$$

де Q_e , Q_p , Q_i – потоки у фрактальній розмірності (m^α / s^β), викликані, відповідно, випаровуванням, опадами та зрошенням.

У фізичній розмірності (m/c) потік вологи Q_i викликаний зрошенням визначається наступним чином:

$$Q_i(t) = \begin{cases} \frac{N_{irr}}{\rho_w t_{irr}}, & t_{i0} \leq t < t_{i0} + t_{irr}, \quad t_{i0} : \frac{1}{v} \int_0^v C(H(z, t_{i0})) dz < \theta_{min}, \\ 0, & \text{у протилежному випадку.} \end{cases} \quad (7.6)$$

У співвідношенні (7.6) використані наступні позначення: ρ_w – щільність зрошувальної води, v – глибина кореневмісного шару ґрунту (M),

$N_{irr} = 100 \cdot \rho \cdot v \cdot (\theta_{max} - \theta_{min})$ – поливна норма $\left(\frac{K\mathcal{L}}{M^2} \right)$, θ_{max} – рівень найменшої

вологоємності, ρ – середня щільність ґрунту в межах кореневмісного шару $\left(\frac{K\mathcal{L}}{M^3} \right)$,

θ_{min} – мінімально допустима середня вологість ґрунту в межах кореневмісного шару,

t_{i0} – час початку поливу, t_{irr} – час, за який поливна норма виливається на поверхню ґрунту.

Початкова умова для рівняння (7.4) має вигляд $H(z, 0) = H_0(z)$.

Використовується наступна функція S екстракції вологи з ґрунту кореневою системою рослин [160]:

$$S = \frac{TL(z)}{\int_0^v L(z)dz}, \quad (7.7)$$

де $L(z)$ – функція щільності кореневої системи, T – транспірація у фрактальній розмірності (m^α / s^β).

Маючи оцінку значення фактичного випаровування ET , потік випаровування Q_e з поверхні ґрунту та транспірації рослин T (для $\alpha = \beta = 1$) обчислюються [106] як

$$\begin{aligned} Q_e &= (1 - M)ET, \\ T &= M \cdot ET, \\ M &= 1 - e^{-\mu \cdot LAI}, \end{aligned} \quad (7.8)$$

де LAI – індекс площі листової поверхні, μ – задана константа.

Значення індексу LAI може бути обчислене використовуючи дані дистанційного зондування Землі на основі його залежності від Нормалізованого Різницевого Вегетативного Індексу (NDVI), наведеної у [151].

Зазначимо, що рівняння (7.4), (7.6), (7.7) містять коефіцієнти просторово-часової фрактальної розмірності. Щоб узгодити їх із вимірюваними значеннями, слідуючи [107] введемо параметри σ_t, σ_z часу та простору і представимо коефіцієнт

вологперенесення у фрактальній розмірності як $k(H) = k_0(H) \frac{\sigma_t^{1-\beta}}{\sigma_z^{1-\alpha}}$, де $k_0(H)$ –

його вимірювана величина (m/s). Така ж процедура проводиться з компонентами евапотранспірації Q_e і T . Розглядаючи модель (7.4)–(7.8) як напівемпіричну, встановимо $\sigma_t = \sigma_z = 1$ і будемо надалі визначати лише порядки дробових похідних та форми функціональних параметрів.

Для моделювання похибки оцінювання евапотранспірації будемо домножувати її оцінку ET на постійний множник k_f . Аналогічно введемо до розгляду множник k_p для потоків Q_p, Q_i , викликаних опадами та зрошенням.

Для чисельного розв'язання прямої задачі (7.4)–(7.8), аналогічно [34, 63], введемо до розгляду рівномірну скінченно-різницеву сітку [36]

$$\omega = \left\{ (z_i = ih, t_j = j\tau) : i = 0, \dots, m, j = 0, 1, 2, \dots \right\}, \quad (7.9)$$

де h, τ – кроки за часовою та просторовою змінними.

На сітці (7.9) визначимо скінченно-різницеві аналоги $\Delta_t^{(\beta)}$ і $\Delta_z^{(\alpha)}$ дробових похідних (7.1) і (7.3) наступним чином:

$$\Delta_t^{(\beta)} H \Big|_{t=t_j} = \frac{1}{\Gamma(1-\beta)} \sum_{s=0}^{j-1} b_s^{(j)} \frac{H^{s+1} - H^s}{\tau}, \quad b_s^{(j)} = \int_{t_s}^{t_{s+1}} (g_t(t_j) - g_t(\xi))^{-\beta} d\xi, \quad (7.10)$$

$$\Delta_z^{(\alpha)} H \Big|_{z=z_i} = \frac{1}{\Gamma(1-\alpha)} \sum_{s=0}^{i-1} \gamma_s^{(i)} \frac{H_{s+1} - H_s}{h}, \quad \gamma_s^{(i)} = \int_{z_s}^{z_{s+1}} (g_z(z_i) - g_z(\xi))^{-\alpha} d\xi. \quad (7.11)$$

З урахуванням (7.11), член $D_{z,g}^{(\alpha)} \left(k \frac{\partial H}{\partial z} \right)$ у рівнянні (7.4) на часовому кроці j

дискретизується як

$$D_{z,g}^{(\alpha)} \left(k \frac{\partial H^{(j)}}{\partial z} \right) \approx \frac{1}{h^2 \Gamma(1-\alpha)} \left(\gamma_{i-1}^{(i)} (k_i H_i^{(j)} - (k_{i-1} + k_{i-2}) H_{i-1}^{(j)} + k_{i-2} H_{i-2}^{(j)}) + \right. \\ \left. + \sum_{s=0}^{i-2} \gamma_s^{(i)} (k_s H_{s+1}^{(j-1)} - (k_s + k_{s-1}) H_s^{(j-1)} + k_{s-1} H_{s-1}^{(j-1)}) \right). \quad (7.12)$$

Дискретизуючи рівняння (7.4) за допомогою (7.10) та (7.12), отримуємо для розв'язку H^{j+1} на кроці $j+1$ за часом наступну тридіагональну СЛАР (значення H_0^{j+1} і H_m^{j+1} отримуються з граничних умов):

$$A_i^j H_{i-1}^{j+1} - R_i^j H_i^{j+1} + B_i^j H_{i+1}^{j+1} = \Omega_i^j \quad (i = 1, \dots, m-1, j = 0, 1, 2, \dots), \quad (7.13)$$

де

$$A_i^j = C^{-1}(H^j) \frac{\gamma_{i-1}^{(i)}}{\Gamma(1-\alpha)} \frac{k_{i-2}}{h^2}, \quad B_i^j = C^{-1}(H^j) \frac{\gamma_{i-1}^{(i)}}{\Gamma(1-\alpha)} \frac{k_{i-1}}{h^2},$$

$$R_i^j = C^{-1}(H^j) \frac{\gamma_{i-1}^{(i)}}{h^2 \Gamma(1-\alpha)} (k_{i-2} + k_{i-1}) + \frac{1}{\tau \Gamma(1-\beta)} b_{j-1}^{(j)},$$

$$\Omega_i^j = \Omega_i^{(1,j)} - \Omega_i^{(2,j)},$$

$$\Omega_i^{(1,j)} = -\frac{1}{\tau \Gamma(1-\beta)} b_{j-1}^{(j)} H^j + \frac{1}{\Gamma(1-\beta)} \sum_{s=0}^{j-2} b_s^{(j)} \frac{H^{s+1} - H^s}{\tau} + C^{-1}(H^j) S, \quad (7.14)$$

$$\Omega_i^{(2,j)} = -\frac{C^{-1}(H^j)}{h^2 \Gamma(1-\alpha)} \sum_{s=0}^{j-2} \gamma_s^{(i)} \left(k_s H_{s+1}^{(j-1)} - (k_s + k_{s-1}) H_s^{(j-1)} + k_{s-1} H_{s-1}^{(j-1)} \right). \quad (7.15)$$

СЛАР (7.13)–(7.15) будемо розв'язувати використовуючи метод прогонки [313]. Для обчислення значень інтегралів $b_s^{(i)}$ і $\gamma_s^{(i)}$ будемо використовувати їх наближення у вигляді рядів [39] (п. 6.1). Застосування такої схеми обмежується випадками, коли відповідні ряди збігаються. У протилежному випадку, будемо використовувати рекурсивний алгоритм описаний у [42] (п. 6.1).

7.2. Алгоритм рою частинок

Вхідними даними для ідентифікації параметрів моделі (7.4)–(7.8) є значення всмоктуючого тиску, отримані від відповідних датчиків. Будемо вважати, що їх показання є достовірними, тоді як інші вихідні дані, такі як оцінки евапотранспірації та ОГХ, можуть мати високі похибки оцінювання. Зокрема, ґрунт може містити такі структури, як плужна підшва, що створюються внаслідок тривалого зрошеного землеробства і можуть бути недостатньо точно зафіксовані під час експериментального аналізу ґрунту.

Для визначення значень параметрів моделі (7.4)–(7.8), які потребують ідентифікації, на основі показань датчиків, спочатку сформулюємо обернену задачу для рівняння (7.4) наступним чином:

- припустимо, що є N відомих значень напору H_i , виміряного у моменти часу T_i на глибинах Z_i ;
- необхідно знайти такий вектор значень параметрів моделі \vec{x} , що мінімізує цільову функцію згідно з методом найменших квадратів (вважаємо, що для похибок вимірювання виконуються умови теореми Гауса–Маркова) [32]:

$$F(\vec{x}) = \sum_{i=1}^N \left(H(Z_i, T_i, \vec{x}) - H_i \right)^2, \quad (7.16)$$

де $H(z, t, \vec{x})$ – чисельний розв’язок прямої задачі (7.4)–(7.8) зі значеннями параметрів з вектора \vec{x} , отриманий згідно з (7.13)–(7.15) для глибини z та моменту часу t .

Беручи до уваги складність оберненої задачі та те, що параметри, які підлягають ідентифікації, представляються числами з рухомою комою, її ефективно розв’язувати за допомогою алгоритму рою частинок (PSO) [249], який для випадку задачі мінімізації опишемо наступним чином [36, 268]:

- Позначимо через S – кількість частинок у рої; x_i – вектор координат частинки i , значення компонентів якого відповідають значенням параметрів моделі і який має розмір M ; v_i – швидкість частинки i ; як p_i координати частинки i , що відповідають найменшому значенню цільової функції (7.16), отриманому нею; g – координати, що відповідають найменшому значенню цільової функції, отриманому роєм; як $\omega, \varphi_p, \varphi_g$ параметри алгоритму;
- На етапі ініціалізації, координати частинок генеруються випадковим чином з урахуванням обмежень на значення параметрів. Швидкості частинок встановлюються рівними нулю. Значення цільової функції обчислюються для кожної частинки разом з початковими значеннями p_i і g .
- Поки не досягнуто заданої максимальної кількості ітерацій I_{\max} , не досягнуто нульового найменшого значення цільової функції і різниця між найбільшим та найменшим значеннями цільової функції для частинок у рої більша за задану сталу, на ітерації j алгоритму для кожної частинки i
 - генеруються випадкові числа $r_p, r_g \in [0, 1]$;
 - змінюється швидкість:

$$v_i^{(j+1)} = \omega \cdot v_i^{(j)} + \varphi_p r_p (p_i^{(j)} - x_i^{(j)}) + \varphi_g r_g (g^{(j)} - x_i^{(j)});$$
 - змінюються координати частинок: $x_i^{(j+1)} = x_i^{(j)} + v_i^{(j)}$;
 - обчислюються значення цільової функції та оновлюються p_i і g .

Оскільки значення параметрів алгоритму PSO $\omega, \varphi_p, \varphi_g$ суттєво впливають на його збіжність та враховуючи складний характер прямої задачі, будемо використовувати наступну схему динамічної зміни $\omega, \varphi_p, \varphi_g$, аналогічну описаній у [211], в процесі проведення обчислень [36]:

- На етапі ініціалізації обчислюється $v_{\max,j} = \left(\max_i x_{i,j} - \min_i x_{i,j} \right) / n, i = 1, \dots, S$,

де $x_{i,j}$ – j -тий компонент вектору координат i -тої частинки (ті самі позначення надалі використовуються і для інших векторів), n – задана стала;

обчислюється $v_0 = \frac{1}{S} \sqrt{\sum_{j=1}^M \sum_{i=1}^S x_{i,j}^2}$; встановлюється початкове значення $\omega = 1$;

- Перед кожною ітерацією k , змінюються значення $\omega, \varphi_p, \varphi_g$ наступним чином (тут індекси позначають номер ітерації):

- $\omega^{(k)} = \omega^{(k-1)} / \omega_p$ якщо $v^{(k-1)} > v_e^{(k)}$, або $\omega^{(k)} = \omega^{(k-1)} \omega_p$ якщо $v^{(k-1)} < v_e^{(k)}$, де

$$v^{(k-1)} = \frac{1}{S} \sqrt{\sum_{j=1}^M \sum_{i=1}^S (v_{i,j}^{(k-1)})^2} \quad \text{і} \quad v_e^{(k)} = v_0 e^{-2(k+1)/(I_{\max} - k)} \cdot \frac{2(k+1)}{I_{\max} - k} \cdot \omega^{(k)} = \omega^{(k-1)} \quad \text{якщо}$$

$$v^{(k-1)} = v_e^{(k)};$$

- $\varphi_{p,j} = C v_{\text{avg},j}^{(k)} / v_{\max,j}$, $\varphi_{g,j} = C (1 - v_{\text{avg},j}^{(k)}) / v_{\max,j}$ де $v_{\text{avg},j}^{(k)} = \frac{1}{S} \sqrt{\sum_{i=1}^S (v_{i,j}^{(k)})^2} \cdot \varphi_{p,j}$

і $\varphi_{g,j}$ додатково обмежені в діапазоні $[0, 2]$. При зміні швидкостей частинок, множники $\varphi_{p,j}$ і $\varphi_{g,j}$ використовуються по-компонентно замість φ_p та φ_g для компонента j вектора швидкості.

Щоб додати варіативності пошуку на останніх етапах ітераційного процесу, виконується випадкова повторна ініціалізація частинок з імовірністю $(1 - \omega^{(k)}) P_r$, де P_r – задане число.

Крім того, для збільшення точності розв'язків, знайдених алгоритмом PSO, може бути додатково використаний метод градієнтного спуску. Пропонується наступна його варіація з чисельним розрахунком градієнта цільової функції [36]:

- Встановлюється початковий крок $s^{(0)} = 1$ і як $p^{(0)}$ позначається вектор початкових параметрів, отриманих згідно з алгоритмом PSO;

- На кожній ітерації $k = 1, 2, \dots$:

- 1) Встановити $s^{(k)} = s^{(k-1)}$;

- 2) Отримати чисельну оцінку вектору градієнта G у формі

$$G_i = \left(F(p^{(k-1)} + \vec{i}_i s^{(k)}) - F(p^{(k-1)}) \right) / s^{(k)}, \text{ де } \vec{i}_{i,j} = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases};$$

- 3) Змістити $p^{(k-1)}$ вздовж вектору градієнта: $p^{(k)} = p^{(k-1)} - s^{(k)} G$;

- 4) Якщо $F(p^{(k)}) > F(p^{(k-1)})$ зменшити крок: $s^{(k)} = s^{(k-1)} / 2$ та перейти до кроку 2 алгоритму якщо $s^{(k)} > \varepsilon_1$, де ε_1 – задане обмеження на розмір кроку;

- 5) вийти з циклу якщо $\left| F(p^{(k)}) - F(p^{(k-1)}) \right| \leq \varepsilon_2$, де ε_2 – задана стала;

- 6) Вийти з циклу, якщо виконана задана кількість ітерацій.

7.3. Алгоритм методу генетичного програмування для пошуку аналітичної форми функціонального параметра ψ -похідної Капуто

Будемо розглядати початково-крайову задачу та чисельний метод її розв'язання, що описані у п. 7.1, та вважатимемо конкретні форми функціональних параметрів g_t, g_z невідомими.

Аналогічно описаному у п. 7.2, вважатимемо, що маємо N достатньо точно визначених значень напору H_i , виміряних у моменти часу T_i на глибинах Z_i . Інші вхідні дані, такі як оцінки евапотранспірації та ОГХ, мають похибки, вплив яких можна знизити за допомогою правильного вибору параметрів моделі — функцій $g_t(t)$ та $g_z(z)$. Таким чином, необхідно знайти такі символічні форми функцій g_t, g_z , які відповідають обмеженням, описаним у р. 5.5 і мінімізують цільову функцію (7.16).

Будемо шукати розв'язок такої оберненої задачі за допомогою наступного алгоритму генетичного програмування [48], який базується на алгоритмі, описаному в [140].

Нехай кожен індивід є парою функцій g_t і g_z , які представлені у вигляді дерев з вузлами, що являють собою одну з операцій: $a + b$, $a * b$, a^b , $\log_{b_0+c}(1+a)$, $b_0 > 1$ – задана стала, або константу $c, 0 < c < c_{max}$, де c_{max} – задане значення, або змінну $x, x > 0$. Кожен вузол, залежно від його типу, може мати до 2 дочірніх вузлів a , b . Такий набір операцій та обмежень щодо їх аргументів гарантують, що функція, представлена відповідним деревом, відповідає обмеженням, описаним у п. 5.5.

Перед обчисленням значень g_t, g_z відповідно до таких дерев, виконуються дві процедури оптимізації для запобігання суттєвого збільшення їх розмірів при виконанні алгоритму генетичного програмування. Перша процедура полягає у обчисленні значень піддерев, які мають у якості значень вузлів лише константи, та заміні їх відповідними вузлами константного типу. Друга процедура проводить заміну ланцюжків операцій $+$ та $*$, що мають варіації форми $+(c_1, +(c_2, x))$, ланцюжками виду $+(c_1 + c_2, x)$.

Початкова популяція алгоритму генетичного програмування, що має розмір S_p , генерується випадковим чином із заданим обмеженням на максимальну глибину дерев d_m .

На кожній ітерації алгоритму виконуються наступні кроки [48]:

1) Зважений випадковий вибір двох індивідів i_1, i_2 з ваговими коефіцієнтами, що залежать від відповідних значень цільової функції. Оскільки недоречний вибір g_t, g_z може призвести до отримання розв'язків зі значною помилкою та враховуючи те, що розв'язується задача мінімізації, при випадковому виборі як ваговий коефіцієнт використовується значення $\frac{1}{\ln(2 + F(g_t, g_z))}$;

2) Кросовер вибраних індивідів. Новий індивід i_n формується як копія i_1 . Рівномірно випадкові піддерева вибираються в представленнях g_t у i_1 та i_2 і піддерево, вибране в i_2 , замінює в i_n піддерево, вибране в i_1 . Те ж саме виконується для g_z . Обчислюється значення цільової функції для i_n і якщо воно є меншим за

найбільше значення цільової функції серед усіх індивідів популяції, i_n замінює у ній "найгіршого" індивіда;

3) З заданою ймовірністю p_m виконується операція мутації. Рівномірно випадковим чином вибрані піддерева всередині рівномірно випадковим чином вибраного індивіда замінюється випадково сформованими піддеревами глибини не більше d_m . Якщо кількість вузлів у мутованого індивіда більше, ніж задане значення n_{max} , операція не проводиться.

Критеріями зупинки є зниження мінімального значення цільової функції у популяції нижче заданого порогового значення ε_{g1} ; зниження різниці між максимальним та мінімальним значеннями цільової функції в межах популяції нижче заданого порогового значення ε_{g2} ; досягнення максимального числа i_m виконаних ітерацій алгоритму.

7.4. Висновки до розділу 7

Розділ 7 присвячено побудові алгоритмів розв'язання задачі ідентифікації параметрів деяких дробово-диференціальних моделей вологоперенесення. Розглядається узагальнене рівняння вологоперенесення у термінах напорів (п. 7.1) що містить ψ -похідні Капуто за часовою та просторовою змінними. Використані дробові похідні узагальнюють похідну Капуто–Герасимова і містять функціональні параметри.

У п. 7.2 для розв'язання задачі знаходження значень числових параметрів моделі за яких вона якнайточніше описує дані моніторингу процесу вологоперенесення пропонується алгоритм методу рою частинок (PSO), зокрема, його варіація у якій виконується динамічна зміна значень параметрів PSO. Додатково пропонується після виконання алгоритму PSO застосовувати процедуру градієнтного спуску з початковим значенням, згенерованим PSO.

У п. 7.3 для випадку, коли конкретна форма функціональних параметрів ψ -похідних Капуто заздалегідь є невідомою, запропоновано алгоритм генетичного програмування для знаходження їх аналітичної форми.

Пропоновані у розділі 7 алгоритми надалі застосовано у розділі 8 до розв'язання прикладної задачі прогнозування динаміки вологості ґрунту у зрошувальному землеробстві.

Результати розділу 7 опубліковано в роботах [36, 37, 48, 268].

Джерела, що використані у розділі 7

Для написання даного розділу було використано 25 джерел [11], [32], [34], [36], [37], [39], [42], [48], [63], [106], [107], [127], [139], [140], [151], [160], [170], [211], [218], [220], [227], [235], [249], [268], [313], посилання на які зазначені в тексті розділу.

Розділ 8. Комп'ютерне моделювання та прогнозування динаміки вологоперенесення при зрошенні ґрунтових масивів дощуванням

Ефективність зрошення суттєво залежить від точності визначення строків і норм поливу. У практиці управління зрошенням для їх визначення використовують різні методи [131], але найпоширенішими з них є ті, що базуються на інструментальних вимірюваннях вологості ґрунту [72] та розрахунках водного балансу ґрунту.

Ці методи широко використовуються як самостійно, так і у складі інформаційно-аналітичних та інформаційно-дорадчих систем управління зрошенням, застосування яких набуває все більшого поширення.

Одним з підходів до побудови інформаційних систем управління водорегулюванням, зокрема зрошенням, є використання математичних моделей вологоперенесення для прогнозування динаміки вмісту вологи у ґрунті та визначення на їх основі графіків поливів, поливних та зрошувальних норм.

Найбільш вживаною моделлю наразі є диференціальне рівняння Річардса, що описує вологоперенесення у ґрунтовому середовищі, масштаб вимірювань у якому суттєво більший за масштаб неоднорідностей. Апарат дробово-диференціальних рівнянь дозволяє підвищити точність моделювання процесів масоперенесення у пористих середовищах, структура яких має ознаки фрактальної, та у яких вищезазначене припущення не виконується [174].

Дробово-диференціальні рівняння вологоперенесення базуються на узагальненнях диференціальних форм закону збереження маси [227] та закону Дарсі [165]. Зауважимо, що наразі існує лише невелика кількість робіт, присвячених виведенню, теоретичному та експериментальному математичному аналізу важливих для практичного застосування у зрошуваному землеробстві дробово-диференціальних моделей вологоперенесення в термінах напорів у насичено-ненасиченому ґрунтовому середовищі з нелінійними гідрофізичними параметрами. Роботи стосовно дробово-диференціального узагальнення рівняння Річардса стосуються в першу чергу його формулювання у термінах вологості [156, 173, 175, 210] або випадку лінійних гідрофізичних

характеристик [264, 309] та фокусуються переважно на теоретичному аналізі. Лише у декількох роботах [136, 137, 216] досліджувались формулювання дробово-диференціальних моделей вологоперенесення в термінах напорів, які можуть безпосередньо застосовуватись для умов зрошення дощуванням.

8.1. Моделювання з використанням класичної моделі та моделі з похідної

Капуто–Герасимова

Будемо розглядати задачу моделювання вологоперенесення при зрошенні дощуванням у одновимірному наближенні на основі моделі, описаної у п. 7.1, у випадку її класичного формулювання та використання у ній похідних Капуто–Герасимова.

Враховуючи суттєвий вплив на точність моделювання такого вихідного параметру моделі, як оцінка евапотранспірації, спершу розглянемо існуючі підходи для виконання такої оцінки та визначимо її вплив на точність моделювання у випадку моделі з похідними цілого порядку.

8.1.1. Моделі евапотранспірації

Існує значна кількість методів визначення евапотранспірації. Експериментально точний розрахунок евапотранспірації може проводитись з використанням вагових лізиметрів, технології турбулентної коваріації та іншими розрахунковими методами [8, 73, 161]. Такі методи мають обмеження, оскільки здатні надавати інформацію про евапотранспірацію лише в конкретній точці та не можуть забезпечити її оцінку навіть в розрізі досить великого господарства. Найбільш використовуваними в умовах України розрахунковими методами оцінки евапотранспірації є методи Пенмана–Монтейта [9], А. М. Алпатьєва та С. М. Алпатьєва [261], Д. А. Штойка [321], М. І. Будико [58], М. М. Іванова [291], Блейні і Крідла [33].

Будемо розглядати чотири найчастіше використовуваних для території України методи оцінки евапотранспірації [186].

Комбінована формула методу Пенмана–Монтейта, що був розроблений ФАО як стандарт для визначення еталонного сумарного випаровування [9], має вигляд

$$ET_0 = \frac{0.408\Delta(R_n - G) + \gamma \frac{900}{T + 273} u_2 (e_s - e_a)}{\Delta + \gamma(1 + 0.34u_2)}, \text{ мм},$$

де ET_0 – еталонна евапотранспірація (мм доба⁻¹); R_n – чиста радіація на поверхні рослин (МДж м⁻² доба⁻¹); G – густина теплового потоку ґрунту (МДж м⁻² доба⁻¹); T – середньодобова температура повітря на висоті 2 м (°C); u_2 – швидкість вітру на висоті 2 м (м с⁻¹); e_s – тиск насиченої пари (кПа); e_a – фактичний тиск (кПа); $(e_s - e_a)$ – дефіцит тиску насиченої пари (кПа); Δ – ухил кривої тиску пари (кПа °C⁻¹); γ – психрометрична стала (кПа °C⁻¹).

Обчислення за формулою Пенмана–Монтейта проводились використовуючи алгоритм, реалізований у плагіні i.evapo.pm для GRASS GIS [126].

Іншим методом, що дозволяє розраховувати уточнені значення сумарної евапотранспірації є метод Штойка [321], в якому при розрахунку сумарного водоспоживання за основу приймається температура і відносна вологість повітря. Цей біофізичний метод базується на тому, що при оптимальній вологості ґрунту процес випаровування практично не регулюється рослиною і ґрунтом, оскільки приплив вологи до транспіруючої рослини або до випаровуючої поверхні необмежений. В цих умовах сумарне випаровування визначається, головним чином, зовнішніми кліматичними факторами – вологістю повітря та його температурою. У початковий період вегетації (від початку появи сходів до значного затінення ґрунту рослинами) рекомендована формула

$$ET = \sum t^0 \left(0,1t_c^0 - \frac{r}{100} \right), \text{ мм},$$

а за повного затінення ґрунту рослинами –

$$ET = \sum t^0 [0,1t_c^0 + (1 - r / 100)], \text{ мм},$$

де $\sum t^0$ – сума середньодобових температур повітря за розрахунковий період (°C); t_c^0 – середньодобова температура повітря за той самий період (°C); r – середня відносна вологість повітря за той самий період (%).

При обчисленні за формулами Штойко, пропонується оцінки евапотранспірації для зімкненого та розімкненого рослинного покриву комбінувати оцінюючи зімкненість за індексом біомаси NDVI, який розраховувався за знімками супутника Sentinel-2. При $NDVI < 0,4$ використовувалась формула для розімкненого покриву, при $NDVI \geq 0,8$ – для зімкненого, а при $0,4 \leq NDVI < 0,8$ – їх лінійна комбінація.

Третім досліджуваним методом оцінки евапотранспірації на основі даних наземного спостереження є формула М.М. Іванова [291], яка має вигляд

$$E = 0.0006(25 + t_c^0)^2 (100 - r), \text{ мм.}$$

Значення оцінок евапотранспірації за формулами Штойко та Іванова обчислювались для розрахункового періоду в одну годину. Для цього, формули Штойко змінювались враховуючи, що $\sum t^0 = t_c^0 = T$, T – температура повітря, усереднена за годину.

Альтернативою методам оцінки евапотранспірації за наземними даними є використання даних дистанційного зондування Землі (ДЗЗ). У цьому контексті досліджувався алгоритм SEBAL [25] (Surface Energy Balance Algorithm for Land – Алгоритм балансу енергії поверхні для землі), реалізований у відкритому програмному забезпеченні Sebal70 [197], що працює у середовищі GRASS GIS. Джерелом вихідних даних для проведення обчислень згідно з цим алгоритмом були знімки супутника Landsat-8.

8.1.2. Вихідні дані

При оцінці точності методів розрахунку евапотранспірації використовувалась модель вологоперенесення, заснована на рівнянні Річардса цілого порядку, записаного у термінах напорів (п. 7.1). Для ідентифікації її параметрів, використовувались дані щодо вологості ґрунту та дані спостереження за біометричними характеристиками вирощуваної культури., зібрані у виробничих умовах на полях ДП ДГ "Брилівське" (с.Привітне, Олешківський р-н, Херсонська обл.) у 2018 році [186].

На дослідній ділянці вирощувалась соя в умовах зрошення дощуванням машиною кругової дії. Точкові вимірювання вологості проводилися за допомогою

датчиків вологості ґрунту Irrrometer 200SS-5 Watermark Soil Moisture Sensor з використанням Інтернет-метеостанції Imetos® Pessl Instruments. Датчики були закладені на глибинах 0,1м, 0,25м, 0,4м, 0,55м, 0,7м та 0,85м. Зауважимо, що для датчиків Watermark за стандартного калібрування відомо (doi: [10.1007/s13201-012-0032-7](https://doi.org/10.1007/s13201-012-0032-7)), що оцінка похибки вимірювання не перевищує 3.6% об'ємного вмісту вологи.

Для точнішого моделювання кореневої системи рослин та її взаємодії з ґрунтовим середовищем, проводились заміри глибини розповсюдження коренів сої. Так, на 12.06.2018 глибина розповсюдження коренів досягла в середньому 10–12 см, на 21.06.2018 – 13–16 см, а на 30.07.2018 – 50–57 см. Для вимірювання глибини та розподілу коріння використано метод вимивання у спеціально влаштованому шурфі [294].

Для визначення основної гідрофізичної характеристики ґрунту і коефіцієнта вологоперенесення у жовтні 2017 р був проведений відбір зразків ґрунтів непорушеної структури та їх експериментальні лабораторні дослідження. Визначено, що ґрунт на полі є темнокаштановим. Відібрано 3 зразки ґрунту: з орного горизонту 5...20 см, підорного (перехідного) горизонту 30...45 см, материнської породи 65...80 см. У лабораторних умовах на спеціально оснащених монолітах ґрунту протягом квітня–червня 2018 р проведено визначення гідрофізичних характеристик ґрунту за методикою, описаною у [295, 296].

Основана гідрофізична характеристика ґрунту визначалася шляхом підбору параметрів моделі ван Генухтена [220] (табл. 8.1) таким чином, щоб вона якнайточніше чином описувала експериментально визначені дані щодо залежності вологості ґрунту від всмоктуючого тиску ґрунтової вологи. Залежність коефіцієнту вологоперенесення від тиску визначалась за формулою Аверьянова [260], що точніше описувала експериментально отримані дані у порівнянні з моделлю ван Генухтена–Муалема [220]. Отримане значення коефіцієнту фільтрації дорівнювало *6,7 см/доба*.

Вибрана початкова дата для верифікації моделі вологоперенесення – 21.05.2018 року. В цей момент часу між першим та другим поливами, вологість у

верхніх шарах ґрунту згідно з показаннями датчиків Watermark почала видимо зменшуватись. Кінцева дата моделювання – 21 серпня – дата, коли соя була зібрана з дослідного поля.

Дані, необхідні для обчислення потенційної та фактичної евапотранспірації, вимірювались відповідними датчиками інтернет-метеостанції, розміщеної на дослідному полі.

Таблиця 8.1 – Значення параметрів моделі ван Генухтена

Шар/крива	θ_0	θ_1	a	n	Максимальна відносна похибка	Середня відносна похибка
0,05-0,2 м/сорбція	0,05	0,3	1,3	1,2	12,66%	2,39%
0,05-0,2 м/висушування	0,05	0,3	0,27	1,3	6,16%	3,89%
0,3-0,45 м/сорбція	0,05	0,375	2,0	1,22	7,26%	1,96%
0,3-0,45 м/висушування	0,05	0,375	0,3	1,35	4,17%	1,39%
0,65-0,8 м/сорбція	0,09	0,39	5,6	1,14	12,65%	2,19%
0,65-0,8 м/висушування	0,09	0,39	0,6	1,21	3,83%	0,93%

Замірні зміни температури та вологості повітря у приземному шарі у діапазоні часу з 21.05.2018 по 17.06.2018 наведені на рис. 8.1.

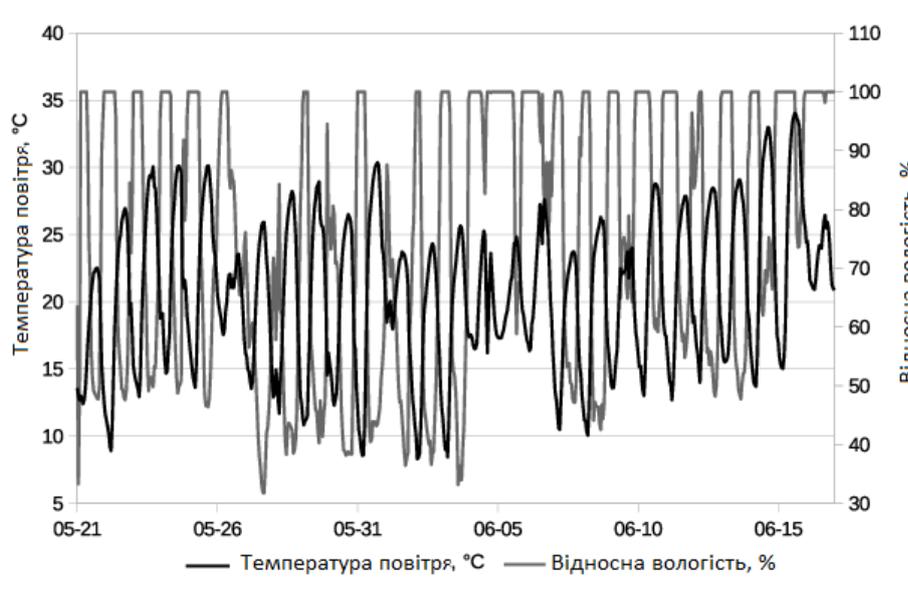


Рисунок 8.1 – Температура, °C, та відносна вологість, %, приземного шару повітря

За відсутності засобів вимірювання на використовуваній інтернет-метеостанції, необхідні для проведення обчислень за формулою Пенмана–Монтейта швидкість вітру на висоті 2 м та сонячна радіація (рис. 8.2) брались за даними NASA POWER (<https://power.larc.nasa.gov/docs/v1/>).

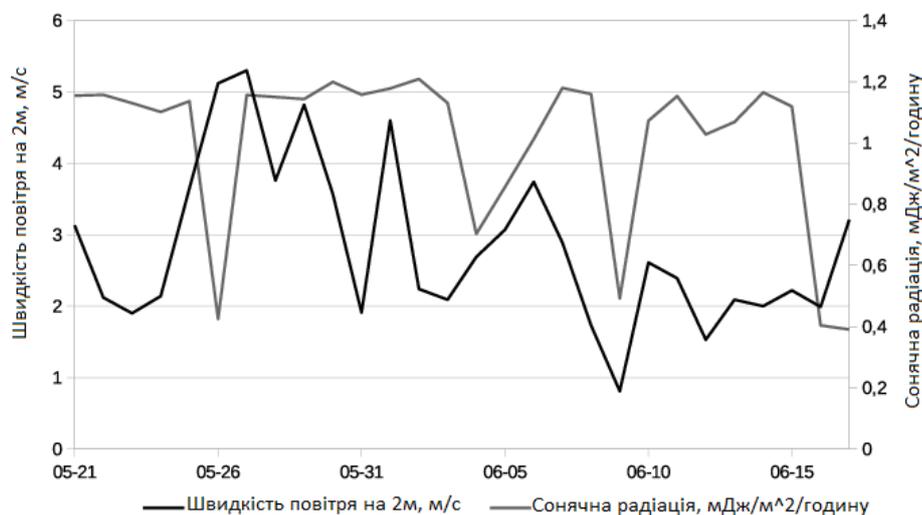


Рисунок 8.2 – Швидкість вітру на висоті 2 м, м/с, та сонячна радіація, МДж/м²/год

Можливість використання цих даних визначена шляхом їх співставлення з даними вимірювань, які проводились метеостанцією, що була встановлена на сусідньому полі. Коефіцієнт кореляції усереднених за добу значень, виміряних метеостанцією та отриманих з NASA POWER, в період спостереження становив $R=0,92$.

Висота рослин, потрібна для використовуваної варіації формули Пенмана–Монтейта, обчислювалась виходячи з проведених замірів як лінійно залежна від NDVI (0,05м при $NDVI < 0,56$; 0,9м при $NDVI \geq 0,8$; лінійно змінювана при $0,56 \leq NDVI < 0,8$).

Коефіцієнт культури слідуючи [133], приймався лінійно залежним від NDVI зі значенням 0,3 для мінімального NDVI та 1 для максимального (максимальне та мінімальне значення взяті згідно з рекомендаціями ФАО для сої [99]).

8.1.3. Оцінка точності моделей евапотранспірації

Отримані різними методами оцінки добової фактичної евапотранспірації наведені на рис. 8.3.

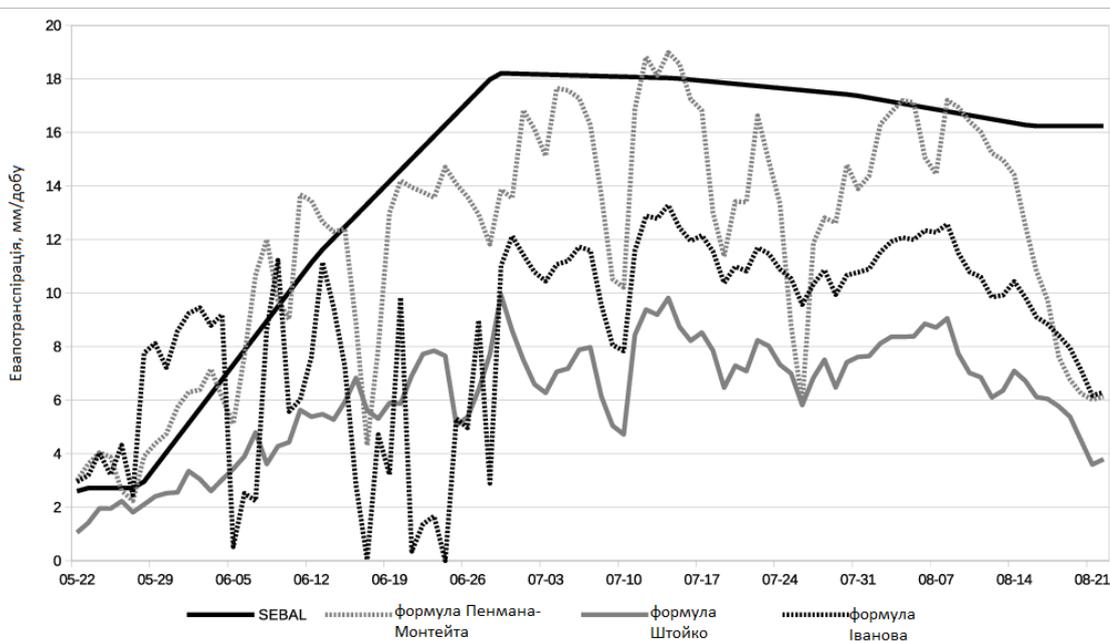


Рисунок 8.3 – Сумарна добова евапотранспірація для діапазону часу з 22.05.2018 по 22.08.2018

Точність оцінок евапотранспірації перевірялась шляхом моделювання за класичною моделлю вологоперенесення зміни вологості ґрунту та порівняння отриманих даних щодо напорів з заміряними датчиками Watermark [187]. Будемо вважати, що точність збільшується при зменшенні похибки моделювання – суми квадратів різниць замірянних та отриманих згідно з моделлю значень напорів у точках спостереження. Початкові умови отримувались розв’язуючи стаціонарну задачу з зафіксованими значеннями напорів на глибинах закладання датчиків.

Використовуючи заміри за період з 21.05.2018 по 5.06.2018, методом рою частинок [249] (п. 7.2) підбирався множник для оцінки евапотранспірації, що моделює її постійну похибку. Множник підбирався таким чином, щоб для заданих оцінок евапотранспірації мінімізувати сумарні квадратичні відхилення змодельованої динаміки напорів від вимірної. Надалі, зміни вологості моделювались для періодів з 21.05.2018 по 25.06.2018 та з 21.05.2018 по 21.08.2018. Вихідні коди розробленого програмного забезпечення наведено у [265].

Розраховані зміни напорів при використанні деяких з розглядуваних методів оцінки евапотранспірації наведено для глибини 0,25м на рис. 8.4. Підібрані значення

множника для оцінки евапотранспірації та середні абсолютні похибки моделювання наведені у табл. 8.2.

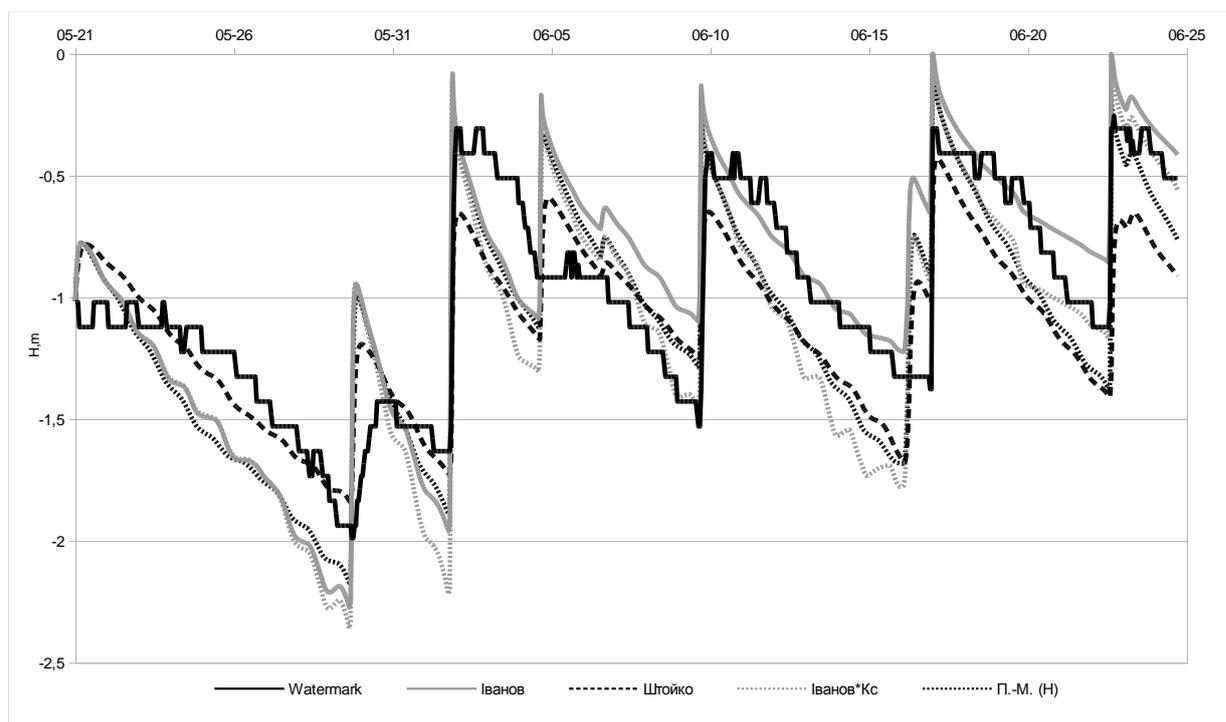


Рисунок 8.4 – Зміни напорів на глибині 0,25м при використанні різних методів оцінки евапотранспірації (“*Кс” означає використання коефіцієнту культури)

Таблиця 8.2 – Значення множника для евапотранспірації та середні абсолютні похибки моделювання

	Множник для евапотранспірації	Середні відхилення, м, для періоду з 21.05 по 5.06	Середні відхилення, м, для періоду з 21.05 по 24.06
Формула Іванова	1,06	0,493	0,424
Формула Іванов з врахуванням коефіцієнту культури	0,92	0,529	0,489
Формула Штойко	2,09	0,458	0,501
Формула Штойко з врахуванням коефіцієнту культури	1,78	0,463	0,847
Формула Пенмана–Монтейта	1,37	0,490	0,479
Формула Пенмана–Монтейта з врахуванням коефіцієнту культури	1,09	0,486	0,876
SEBAL	1,34	0,486	1,443

Чотирма алгоритмами, якими були отримані найменші похибки моделювання на інтервалі з 21.05.2018 по 24.06.2018, було проведено моделювання для максимального досліджуваного інтервалу — з 21.05.2018 до 21.08.2018 (рис. 8.5).

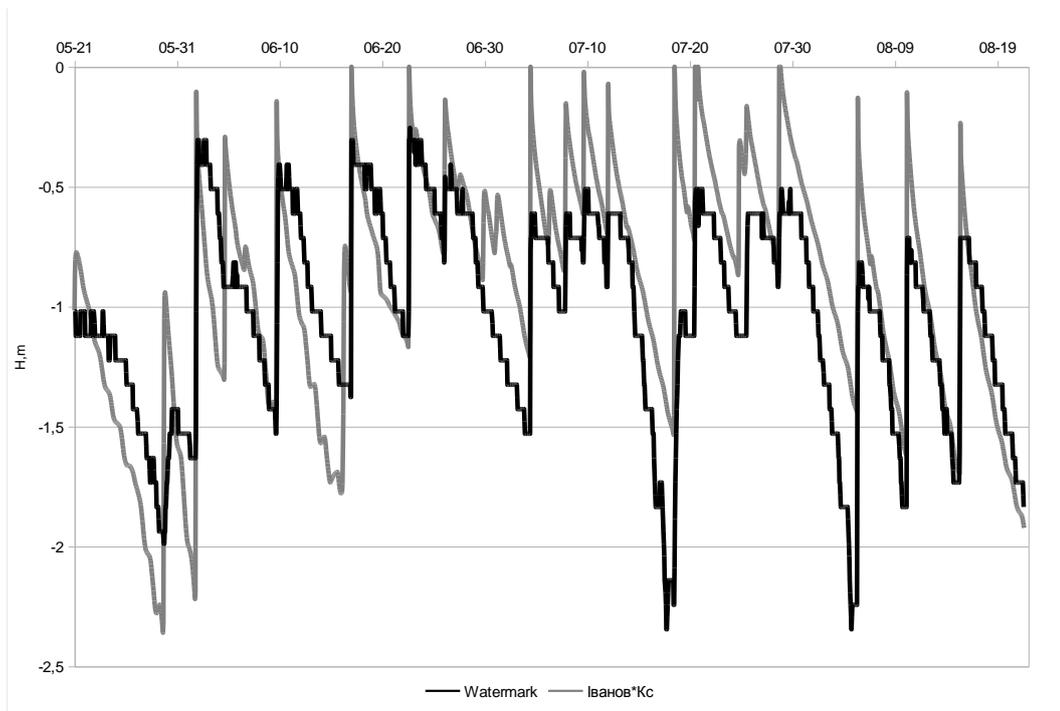


Рисунок 8.5 – Зміни напорів на глибині 0,25м для діапазону часу з 21.05.2018 по 21.08.2018 (Watermark – виміряні напори, Іванов*Кс — змодельовані при використанні формули Іванова помноженої на коефіцієнт культури)

Додатково проводилось моделювання вологості з використанням лінійної комбінації розглядуваних оцінок евапотранспірації. Отримані значення множників та похибки моделювання наведені у табл. 8.3.

Для діапазону часу з 21.05.2018 по 5.06.2018, дані в межах якого використовувались для підбору множника оцінки евапотранспірації, всіма алгоритмами вдалось отримати адекватні результати. Множник для евапортанспірації при цьому був меншим, коли оцінка евапортанспірації і так множилась на коефіцієнт культури. В цей час біомаса й, відповідно, коефіцієнт культури, лінійно збільшувались і така поведінка моделі виглядає логічною. Найкращі, згідно зі середньою абсолютною похибкою моделювання динаміки напорів, оцінки евапортанспірації у цьому випадку давав метод Штойко. Проте він домножувався на множник, що суттєво відрізняється

від 1, тобто оцінки за методом Штойко адекватніше за інші описуючи динаміку евапотранспірації, мали суттєві похибки у абсолютних її значеннях. Ближче всього до 1 були множники для методів Іванова та Пенмана–Монтейта, а трохи більший – для оцінок за алгоритмом SEBAL.

Таблиця 8.3 – Значення множників та похибка моделювання при використанні лінійної комбінації оцінок евапотранспірації

	Оцінки евапотранспірації домножені на коефіцієнт культури	Оцінки евапотранспірації без врахування коефіцієнту культури
Коефіцієнт для формули Іванова	0,230	0,255
Коефіцієнт для формули Штойко	0,599	0,691
Коефіцієнт для формули Пенмана–Монтейта	0,290	0,378
Коефіцієнт для SEBAL	0,207	0,213
Середні відхилення, м, для періоду з 21.05 по 5.06	0,476	0,478
Середні відхилення, м, для періоду з 21.05 по 24.06	0,455	0,420
Середні відхилення, м, для періоду з 21.05 по 21.08	0,555	0,699

На довшому інтервалі (з 21.05.2018 по 24.06.2018) адекватні значення вдалось отримати лише при оцінці евапотранспірації методом Штойко та методом Пенмана–Монтейта без врахування коефіцієнту культури, а також методом Іванова як з, так і без врахування коефіцієнта культури.

На цьому, більшому, інтервалі NDVI вже не настільки лінійно змінювався і чим менша була кореляція оцінки евапотранспірації з NDVI, тим точніше процес моделювався у випадку, коли евапотранспірація домножується на коефіцієнт культури (тобто, методи, що не враховують біологічну складову, зокрема метод Іванова, мають бути домножені на коефіцієнт, що її моделює). Для ситуації коли оцінка не множиться на коефіцієнт культури, ця залежність куди менш суттєва (у

випадку методу Штойко, який є біофізичним, додаткове множення на коефіцієнт культури погіршує точність моделювання).

Для моделювання більшої частини сезону вегетації, єдиним адекватним алгоритмом оцінки евапотранспірації виявився метод Іванова з домноженням на коефіцієнт культури.

Результати моделювання з використанням лінійної комбінації оцінок евапотранспірації демонструють, що у обох випадках, з та без домноження на коефіцієнт культури, найбільший вклад у таку лінійну комбінацію вносить оцінка за методом Штойко. В результаті, похибка моделювання для періоду з 21.05.2018 по 5.06.2018 була близька до похибки при оцінці евапотранспірації виключно за методом Штойко (0,458), яка є найменшою для цього інтервалу часу поміж усіх розглянутих варіантів. Для періоду з 21.05.2018 по 24.06.2018 похибка у випадку лінійної комбінації оцінок як з, так і без врахування коефіцієнту культур, співставна з найменшою похибкою, отриманою при оцінці евапотранспірації виключно за методом Іванова (0,424). У випадку моделювання усього сезону вегетації використання коефіцієнту культури тут стає критичним так само як і при використанні лише однієї з оцінок. Моделювання за лінійною комбінацією оцінок дає найменшу похибку серед усіх тестованих варіантів — 0,555 у порівнянні з 0,593 для методу Іванова. Зменшення похибки, щоправда, складає лише 6,5%.

8.1.4. Точність моделювання вологоперенесення за дробово-диференціальною моделлю

Для дробово-диференціальної моделі вологоперенесення, описаної у п. 7.1 [187], у випадку класичних похідних Капуто будемо ідентифікувати значення наступних параметрів, одержання яких експериментальним шляхом ускладнене:

- Показники дробових похідних;
- Стала μ , необхідна для розділення фактичної евапотранспірації на евапораційну та транспіраційну складову;
- k_I – множник для евапотранспірації, що моделює похибки її оцінок, вважаючи їх постійними;

- k_p – множник для даних щодо величин опадів та норм поливу, який моделює некоректності у роботі вимірювальних приладів та/або некоректності обчислювальної схеми, що виникають через різку зміну значень коефіцієнтів крайових умов на верхній границі області моделювання. Цей множник може також моделювати некоректності, що виникають внаслідок застосування одновимірного наближення для моделювання тривимірного за своєю суттю процесу зрошення дощувальними машинами;
- Коефіцієнти формули ван Генухтена та коефіцієнт фільтрації у випадку відсутності експериментальних даних для їх визначення.

Ідентифікація проводилась з використанням алгоритму рою частинок, описаного у п. 7.2.

Обчислення фактичної евапотранспірації проводились за формулою Іванова [291], що була визначена як найбільш точна [186] у розглядуваних умовах (п. 8.1.3).

Алгоритм виконання обчислень на одному кроці за часом при моделюванні вологості наступний [187]:

- Обчислюється оцінка фактичної евапотранспірації, що вважається постійною в межах часового інтервалу, для якого не змінюються вихідні дані для обчислення;
- На основі фактичної евапотранспірації обчислюються окремо значення евапораційного та транспіраційного її компонентів. Значення індексу листової поверхні при цьому обчислюється за даними ДЗЗ, або експертним шляхом;
- Отримується розв'язок дискретизованої початково-крайової задачі;
- Проводиться перевірка умови призначення поливу та обчислюється значення потоку вологи Q_i викликаного зрошенням для наступного кроку за часом. Обчислені значення є рекомендаціями щодо призначення поливів. Для подальшого моделювання з метою підтримки прийняття рішень мають використовуватись фактичні дані про проведені поливи, які можуть не збігатись з пропонованими.

Аналіз моделювання за класичною моделлю.

Аналогічно обчислювальним експериментам, описаним у пп. 8.1.3, розглядалися три діапазони часу – від 21.05.2018 до 21.08.2018, від 21.05.2018 по 5.06.2018 (діапазон, за даними, замірними у який, відбувався підбір параметрів моделі) та від 21.05.2018 по 24.06.2018 (діапазон, в межах якого біомаса наростала, а після якого трималась на відносно стабільному рівні).

У випадку моделі з похідними цілого порядку, параметри, значення яких підбиралися методом рою частинок, були наступні:

- Множник для евапотранспірації k_I ;
- Множник для даних щодо величин опадів та норм поливу k_P ;
- k_F – коефіцієнт фільтрації, м/с.

У якості показника точності моделювання використовувалась середня квадратична похибка — сума квадратів різниць між показаннями датчиків Watermark та змодельованими значеннями напорів, поділена на кількість вимірювань. Вихідні коди розробленого програмного забезпечення наведено у [265].

Параметри алгоритму рою частинок варіювались у діапазонах $N=[20,50]$, $\omega=\varphi_p=\varphi_g=[0,2;0,8]$. Найвища точність була досягнута при $N=40$, $\omega=\varphi_p=\varphi_g=0,6$.

Значення множника евапотранспірації для усіх знайдених наближень до оптимуму варіювалось незначним чином і було близьке до 1, що демонструє адекватність оцінки евапотранспірації методом Іванова у розглядуваній ситуації. Значення множника для опадів та зрошення коливалось у діапазоні від 2,6 до 3,6. При моделюванні у діапазонах часу від 21.05.2018 до 5.06.2018 та 24.06.2018, вища точність отримувалась при менших його значеннях.

Найсуттєвіший вплив на точність моделювання мало значення коефіцієнту фільтрації. Отримані наближення до оптимальних розв'язків можна розділити на дві групи — при коефіцієнті фільтрації $\sim 5 \cdot 10^{-6}$ м/с похибка для діапазону часу від 21.05.2018 до 5.06.2018 коливалась навколо значення 0,3, а при коефіцієнті фільтрації $\sim 1 \cdot 10^{-6}$ м/с — навколо 0,16, тобто, розв'язок був вдвічі точніший. Для діапазону часу від 21.05.2018 до 24.06.2018 характерне таке ж розділення, тоді як для

діапазону часу від 21.05.2018 до 21.08.2018 залежність точності моделювання від підбраного значення коефіцієнту фільтрації протилежна.

Така поведінка похибок моделювання може бути пояснена збільшенням коефіцієнту фільтрації ґрунту протягом сезону вегетації, що може виникнути при системі обробки no-till через розпушення ґрунту кореневою системою рослин [155]. Вважаючи збільшення коефіцієнту фільтрації лінійним, була проведена серія обчислювальних експериментів для визначення оптимального значення коефіцієнта такої лінійної залежності. При значеннях параметрів моделі, за якими було отримано мінімальне значення цільової функції (множник для евапотранспірації – 0,968, базовий коефіцієнт фільтрації $k_{bF}=7,73*10^{-7}$ м/с, множник для опадів/зрошення–2,68), шукалось значення коефіцієнту k_l лінійного збільшення коефіцієнту фільтрації k_F з часом t ($k_F=k_{bF}+ k_l*t$, $t=0$ на початковий момент моделювання), за якого модель з найбільшою точністю описувала б процес у інтервалі часу з 21.05.2018 по 21.08.2018. Оптимальне знайдене значення k_l дорівнювало $7,5*10^{-14}$ м/с². Похибка моделювання на інтервалі часу з 21.05.2018 по 21.08.2018 при цьому зменшувалась на 40%, на інтервалі часу з 21.05.2018 по 24.06.2018 змінювалась незначним чином, а на початковому інтервалі з 21.05.2018 по 5.06.2018 — збільшувалась на 8%. При визначенні середньої об'ємної вологості кореневмісного шару найбільша абсолютна похибка склала 9,5%, а середня – 1,9%.

Використання змінного коефіцієнта фільтрації суттєво збільшує точність моделювання, що може слугувати підтвердженням факту розпушення ґрунту рослинами протягом періоду вегетації. Погіршення точності моделювання у початковому інтервалі часу може бути пояснено нелінійним характером такої зміни.

Коефіцієнт лінійного збільшення значення коефіцієнта фільтрації разом з іншими параметрами моделі підбирався також методом рою частинок згідно з даними, зібраними у діапазоні часу 21.05.2018 по 5.06.2018. Знайдене значення базового коефіцієнта фільтрації при цьому склало $k_{bF}=5,83*10^{-7}$ м/с, а значення k_l – $2,32*10^{-14}$ м/с². Інші параметри лишилися незмінними у порівнянні з вищеописаними експериментами. Тут, для середньої об'ємної вологості кореневмісного шару ґрунту найбільша абсолютна похибка склала 8,77%, а

середня – 1,44%. Таким чином, в умовах проведеного експерименту, запропонована методика дозволяє ідентифікувати зміни коефіцієнту вологоперенесення ґрунту з часом на основі замірів у початковий період сезону вегетації.

Аналіз моделювання динаміки вологості за дробово-диференціальною моделлю. При моделюванні динаміки зміни вологості ґрунту за дробово-диференціальною моделлю, застосовувались дві схеми знаходження оптимальних значень її параметрів. У першій схемі, значення показників дробових похідних α (за часовою змінною) та β (за просторовою змінною) знаходились для фіксованих значень множників та коефіцієнту фільтрації, які, у свою чергу, знаходились як наближені до оптимальних для класичної моделі. У другій схемі, значення всіх коефіцієнтів підбирались алгоритмом рою частинок одночасно.

Знайдені оптимальні значення параметрів для класичної та неklasичної моделей та середні квадратичні похибки моделювання наведено у табл. 8.4.

Підбираючи показники похідних за фіксованих інших параметрів неklasичної моделі, була отримана похибка моделювання у початковому діапазоні часу менша на 1,2% за похибку моделювання за класичною моделлю. При чому застосування змінного коефіцієнта фільтрації нівелювало і це зменшення. Для більших діапазонів часу похибка ставала більшою.

При підборі одночасно всіх параметрів неklasичної моделі, на початковому діапазоні часу похибка моделювання була на 10% меншою ніж при застосуванні класичної моделі. У випадку змінного коефіцієнта фільтрації виграш від застосування неklasичної моделі зменшувався. Для більших діапазонів часу, як і у попередньому випадку, похибка моделювання за неklasичною моделлю була більшою за похибку моделювання за класичною моделлю.

Знайдені значення показника дробової похідної за просторовою змінною були близькі до 1, тоді як показник дробової похідної за часовою змінною був близький до 0,99. Такі значення описують незначну часову нелокальність процесу, що може виникати, з однією сторони як наслідок фрактальності ґрунтового середовища, а з іншої – як наслідок затримок у реакції датчиків на зміни вологості ґрунту. Неefективність підбору виключно показників дробових похідних при фіксованих

інших параметрах моделі може бути пояснена зміною одиниць вимірювання цих параметрів.

Таблиця 8.4 – Середні квадратичні похибки для класичної та дробово-диференціальної моделей

Модель	Множник для евапо-транспірації	Коефіцієнт фільтрації, м/с	Множник для опадів/зрошення	Показники дробових похідних		Середня квадратична похибка для діапазону часу з 21.05 по		
				α	β	5.06	24.06	21.08
Класична	0,97	$7,73 \cdot 10^{-7}$	2,68			0,143	0,144	0,440
Класична (змінний коефіцієнт фільтрації)						0,155	0,143	0,261
Некласична (підбір показників похідних)				0,99	1	0,142	0,147	0,473
Некласична (підбір показників похідних, змінний коефіцієнт фільтрації)						0,156	0,144	0,261
Відхилення похибок (у порівнянні з класичною моделлю):						1,22%	-1,53%	-7,5%
Відхилення похибок (змінний коефіцієнт фільтрації, у порівнянні з класичною моделлю):						-0,27%	-0,36%	-1%
Некласична (підбір множників, коефіцієнту фільтрації та показників похідних)	1,22	$5,65 \cdot 10^{-7}$	3,25	0,98	0,99	0,129	0,147	0,528
Некласична (підбір множників, коефіцієнту фільтрації та показників похідних, змінний коефіцієнт фільтрації)						0,145	0,145	0,275
Відхилення похибок (у порівнянні з класичною моделлю):						10,19%	-1,15%	-20%
Відхилення похибок (змінний коефіцієнт фільтрації, у порівнянні з класичною моделлю):						7,05%	-1,03%	-5,3%

некласичної) збільшення точності моделювання для діапазону часу, на даних зібраних у який проводиться підбір значень порядків похідних. Середня відносна похибка при цьому суттєво не змінюється і складає 25%. При екстраполяційному моделюванні на більші діапазони часу, ефекту від використання некласичної моделі не спостерігалось.

Аналіз моделювання за умов ідентифікації значень коефіцієнтів моделі ван Генухтена. Моделювання вологоперенесення за умов ідентифікації значень коефіцієнтів моделі ван Генухтена за замірами напорів датчиками Watermark проводилось, аналогічно випадку з відомими гідрофізичними характеристиками ґрунту

- за класичною моделлю (множники для евапотранспірації та опадів/зрошення варіювались у діапазоні $[0,5]$, коефіцієнт фільтрації – у діапазоні $[5 \cdot 10^{-8} \text{ м/с}, 5 \cdot 10^{-5} \text{ м/с}]$);

- за некласичною моделлю з одночасним підбором значень всіх параметрів (множники для евапотранспірації та опадів/зрошення варіювались у діапазоні $[0,50]$, коефіцієнт фільтрації – у діапазоні $[5 \cdot 10^{-8} \text{ м/с}, 5 \cdot 10^{-5} \text{ м/с}]$, порядки дробових похідних – у діапазоні $[0,5;1]$);

- за некласичною моделлю з підбором порядки дробових похідних після підбору інших параметрів (порядки дробових похідних варіювались у діапазоні $[0,9;1]$).

Усі випадки розглядались як з фіксованим, так і зі змінним з часом коефіцієнтом фільтрації.

Загалом, у порівнянні з найкращими розв'язками (тут і надалі під "найкращими розв'язками" будемо розуміти розв'язки з найменшим значенням цільової функції в межах проведеного обчислювального експерименту), отриманими за експериментально вимірних коефіцієнтів моделі ван Генухтена, їх підбір дозволяє збільшити точність моделювання динаміки напорів до 27% на інтервалі часу, за даними в межах якого відбувається підбір параметрів. Проте, при екстраполяційному моделюванні на більші діапазони часу, точність зменшується. Так, на інтервалі від 21.05.2018 до 24.06.2018 вища на 7% точність була отримана

лише неklasичною моделлю з підбором порядків дробових похідних після підбору оптимізованих значень інших параметрів з застосуванням лінійного збільшення коефіцієнту фільтрації з часом.

При в цілому адекватному моделюванні процесу вологоперенесення, підібрані значення коефіцієнтів моделі ван Генухтена, а також коефіцієнту фільтрації, суттєво відрізняються від експериментально виміряних. Так, коефіцієнти, що впливають на форму кривої водоутримання, підбирались в більшості обчислювальних експериментів близькими один до одного та до виміряних значень (середнє $n=1,47$ при стандартному відхиленні у 0,08 та виміряному значенні у 1,35, середнє $a=0,336$ при стандартному відхиленні у 0,162 та виміряному значенні у 0,3). Але значення мінімальної та максимальної вологоємностей, хоч і були так само близькими один до одного, суттєво відрізнялись від виміряних (середнє $\theta_0=0,295$ при стандартному відхиленні у 0,055 та виміряному значенні у 0,05, середнє $\theta_1=0,7$ при стандартному відхиленні у 0,08 та виміряному значенні у 0,375). При цьому, при переході від менш до більш точних розв'язків спостерігалось незначне зменшення коефіцієнту n , збільшення рівня повної вологоємності θ_1 та варіації без явного тренду параметру a , що було причиною більшого, у порівнянні з іншими коефіцієнтами, стандартного відхилення серед отриманих його оптимізованих значень.

При адекватному моделюванні поля напорів, при підборі значень усіх чотирьох коефіцієнтів моделі ван Генухтена, змодельовані значення вологості не описують фізичний стан ґрунту й, відповідно, не можуть застосовуватись при управлінні зрошенням для визначення норм поливів. Для забезпечення можливості прогнозувати у цьому випадку вологість ґрунту, необхідно зафіксувати принаймні одне значення на кривій водоутримання. Пропонується задля мінімізації гідрофізичного обстеження ґрунтів, проводити одне вимірювання вологості θ_t разом з фіксацією напору H_t , після чого ідентифікувати параметри a , n , θ_0 моделі ван Генухтена методом рою частинок, а параметр θ_1 визначати з умови $S(H_t) = \theta_t$. При проведенні такого моделювання при $H_t = -10 \text{ кПа}$, $\theta_t = 22,5\%$, за незначним чином зміненої точності моделювання напорів, найбільша абсолютна похибка оцінки середньої об'ємної вологості кореневмісного шару ґрунту склала 10,43% для випадку

постійного коефіцієнта фільтрації та 9,92% – для змінного. Середня абсолютна похибка склала 2,26% для постійного коефіцієнта фільтрації та 2,93% – для змінного. Отримані похибки співставні з похибками, що одержуються при проведенні моделювання за відомих гідрофізичних характеристик ґрунту.

Таким чином, запропонована методика дозволяє моделювати зміну вологості ґрунту ідентифікуючи форму кривої водотримання на основі замірів напорів у початковий період вегетації та одного вимірювання вологості ґрунту.

Моделювання призначення поливів. Будемо проводити оцінку сумарної за сезон норми зрошення використовуючи модель з параметрами, ідентифікованими за даними, зібраними на початку сезону. Ефективність такої оцінки перевірялась порівнянням змодельованих норм зрошення з фактичними в межах проведеного польового експерименту.

При моделюванні, призначення зрошення відбувалось при зменшенні середньої вологості кореневмісного шару ґрунту нижче заданого нижнього граничного значення. Зрошення відбувалось до того моменту, доки середнє значення було менше за задане верхнє граничне. Граничні значення визначались виходячи зі значень, за яких зрошення призначалось та закінчувалось в межах польового експерименту. Середнє значення вологості у кореневмісному шарі ґрунту згідно з показаннями датчиків Watermark визначалось наступним чином. Оскільки при проведенні моделювання розглядалась фіксована глибина кореневмісного шару, що дорівнювала 0,5м, показники напорів для датчиків, розміщених на глибинах 0,1м, 0,25м та 0,4м перераховувались у об'ємну вологість згідно з моделлю ван Генухтена. Середня вологість після цього обчислювалась як зважене середнє з ваговим коефіцієнтом 0,35 для вологості, обчисленої за даними датчиків, розміщених на глибинах 0,1м та 0,4м, та ваговим коефіцієнтом 0,3 для вологості, обчисленої за даними датчику, розміщеного на глибині 0,25м. Зміна таким чином обчисленої середньої вологості з часом наведена на рис. 8.7.

Як видно з рис. 8.7, середня вологість за якої призначалось зрошення варіювалась в діапазоні від 21% до 25% об'єму, тоді як фінальна вологість після проведення зрошення варіювалась в діапазоні від 29% до 36%. Така розбіжність

може бути пояснена тим, що польовий експеримент проводився у виробничих умовах, і зрошення призначалось експертним шляхом. Падіння вологості з моменту закінчення попереднього зрошення до призначення наступного у більшості випадків складало 70–75% від середньої вологості на кінець попереднього зрошення. У зв'язку з цим, для проведення моделювання, у якості верхнього граничного значення при призначенні зрошення було вибрано рівень у 96% від максимальної вологоємності (36%), а у якості нижнього — рівень у 70% від максимальної вологоємності (28,2%).

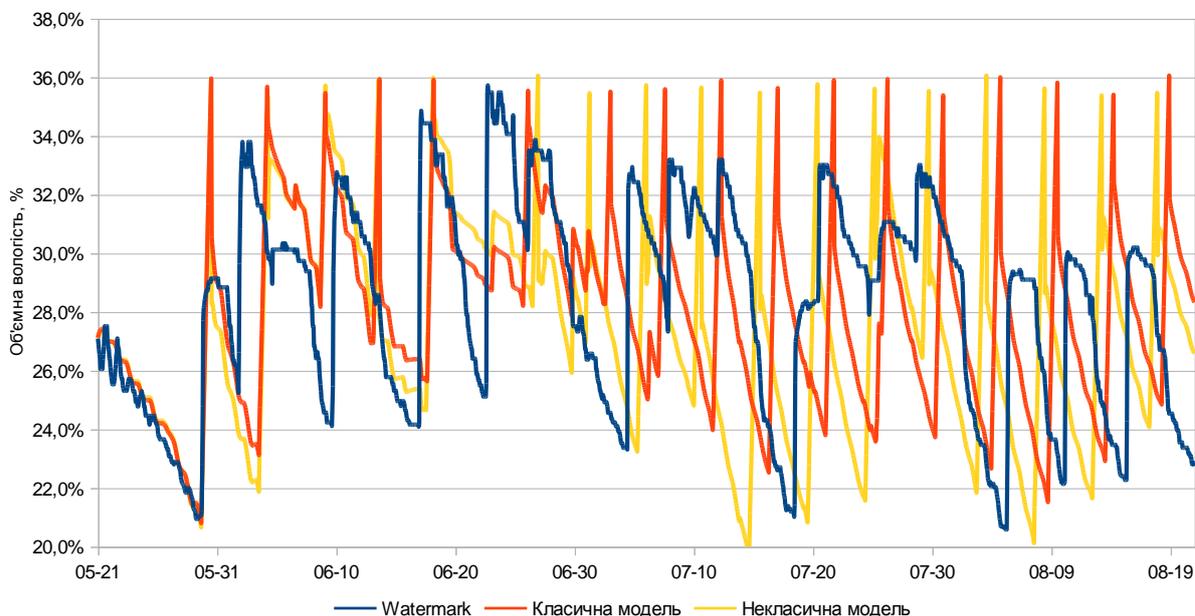
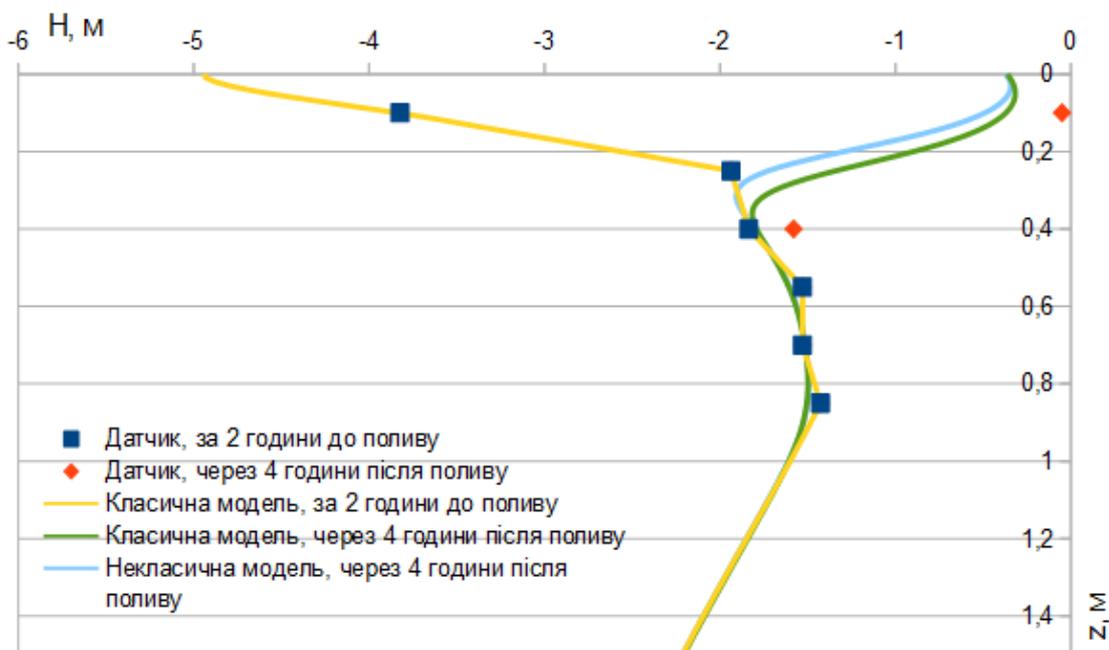


Рисунок 8.7 – Зміна середньої вологості у кореневмісному шару

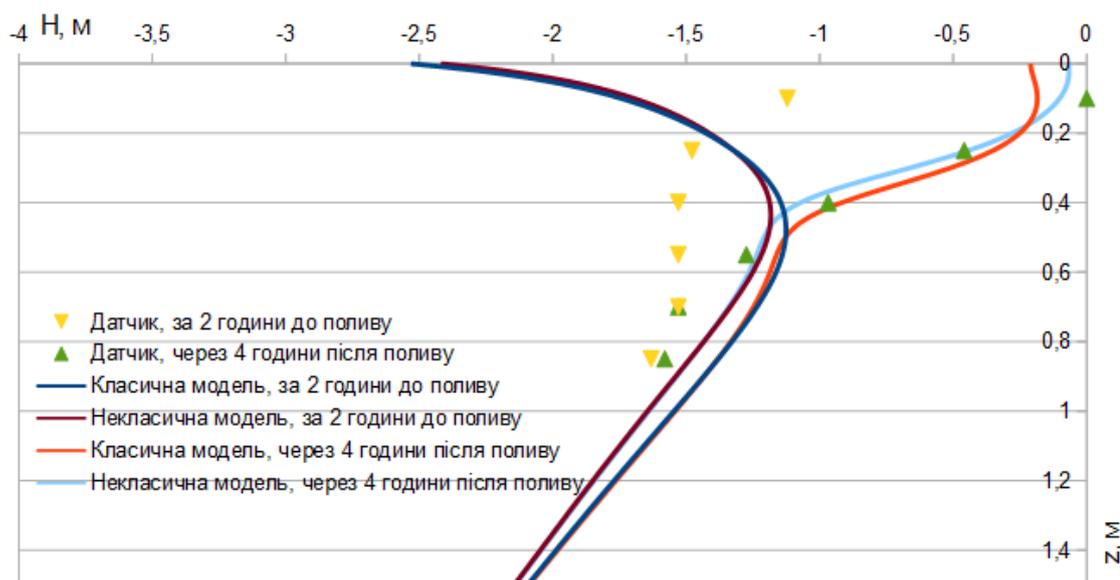
Епюри напорів у моменти часу за 2 години до поливу та за 4 години після поливу для першого та четвертого поливів наведено на рис. 8.8.

Отже, адекватні результати вдалось отримати використовуючи чотири варіанти проведення моделювання — за класичною моделлю з постійним та змінним коефіцієнтом фільтрації та за некласичної моделлю з сумісним визначенням оптимальних значень параметрів з постійним та змінним коефіцієнтом фільтрації. Сумарні норми зрошення та кількості поливів для періоду з 21.05 по 21.08 наведені у табл. 8.5.

З отриманих даних видно, що в усіх випадках змодельована сумарна норма зрошення відрізняється від фактичної не більше ніж на 12%. Модельована кількість поливів відрізнялась від фактичної не більше ніж на 1. Використання змінного коефіцієнта фільтрації та неklasичної моделі не вплинуло на точність результату. Точність моделювання сумарної норми зрошення не корелювала з точністю моделювання зміни вологості в ґрунті.



а) полив 29.05.2018 (17 мм)



б) полив 9.06.2018 (17 мм)

Рисунок 8.8 – Напори до та після поливів

Таблиця 8.5 – Сумарні норми зрошення

	Сумарна норма зрошення, мм	Середня величина норми поливу, мм	Кількість поливів	Середня квадратична похибка моделювання напорів для діапазону часу з 21.05 по 21.08	Відхилення змодельованої сумарної норми зрошення від фактичної
Фактичне зрошення	445,6	22,0	17		
Класична модель	458,8	22,8	17	0,440	2,95%
Класична модель (змінний коефіцієнт фільтрації)	498,9	23,7	18	0,261	11,96%
Некласична модель	484,8	24,3	17	0,528	8,79%
Некласична модель (змінний коефіцієнт фільтрації)	489,0	23,2	18	0,275	9,74%

8.2. Моделювання з використанням ψ -похідної Капуто з функціональним параметром фіксованої форми

Приймаючи за вихідну гіпотезу твердження, що форма ядра оператора дробової похідної відбиває нелокальні особливості структури ґрунтів, з метою збільшення точності прогнозування їх водного режиму будемо розглядати два варіанти похідної (7.1). Перший – це похідна Капуто–Катугампола [134], яка отримується з (7.1) у випадку $g_z(z) = z^\chi$. У другому варіанті як ядро оператора (7.1) використаємо наступні сигмоподібні та зворотні сигмоподібні функції [36]:

$$g_z(z) = \frac{1}{\sigma_a} \left(\frac{1}{1 + e^{-\chi(2z - \sigma_2)}} - \sigma_b \right), \chi \geq 0,$$

$$g_z(z) = -\frac{1}{2\chi} \ln \left(\frac{\sigma_a z + \sigma_b}{1 - \sigma_a z - \sigma_b} \right) + 1, \chi < 0, \quad (8.1)$$

$$\sigma_a = \frac{1}{1 + e^{-\chi(2 - \sigma_2)}} - \sigma_b, \sigma_b = \frac{1}{1 + e^{|\chi|\sigma_2}}.$$

Для похідної (7.3) за часовою змінною будемо розглядати лише випадок $g_t(t) = t^\gamma$.

Функцію щільності розподілу коріння $L(z)$ будемо розглядати у лінійному вигляді $L(z) = \begin{cases} 1 - z/v, & 0 \leq z \leq v \\ 0, & z < 0 \vee z > v \end{cases}$ для $v = 0,5 \text{ м}$. Значення коефіцієнту розподілу евапотранспірації на евапораційний та транспіраційний компоненти приймемо таким, що дорівнює $\mu = 0,5$. Для оцінки евапотранспірації будемо застосовувати метод Штойко згідно зі схемою, описаної у пп. 8.1.1.

У якості набору вихідних даних будемо розглядати вимірювання динаміки вологості ґрунту в процесі вирощування сої за зрошення дощуванням в Херсонській області [36]. Для проведення вимірювань вихідних даних у 2019 році на полі, розташованому за координатами $46^{\circ}34'33,87''\text{N}$, $33^{\circ}49'33,3192''\text{E}$, була встановлена метеостанція iMetos. Станція містила датчики температури та вологості повітря і три датчики Watermark, які вимірюють всмоктуючий тиск у ґрунті. Датчики Watermark були розташовані на глибинах 15, 35 і 55 см. Показання датчиків автоматично знімалися раз на годину.

Основна гідрофізична характеристику (ОГХ) та коефіцієнт фільтрації ґрунту експериментально визначали, використовуючи метод, описаний у [296], для трьох шарів ґрунту. Значення коефіцієнтів моделі ван Генухтена, отримані аналогічно описаному у п. 8.1, наведені в табл. 8.6 разом із значеннями коефіцієнтів фільтрації. Залежності між коефіцієнтом вологоперенесення та напором визначались за формулою Авер'янова [260].

Розглянемо три періоди часу, у які ґрунт просушувався між зрошеннями. Перший період з 3.07.2019 11:00:00 по 8.07.2019 04:00:00 був використаний для ідентифікації параметрів моделі, тоді як наступні два періоди з 9.07.2019 03:00:00 до 19.07.2019 07:00:00 та з 28.07.2019 22:00:00 до 4.08.2019 02:00:00 використовувались для оцінки точності прогнозування.

Таблиця 8.6 – Значення коефіцієнтів моделі ван Генухтена та коефіцієнтів фільтрації K

Шар	n	a	θ_0	θ_1	$K, \text{ м/с}$
0 см – 30 см	1,1	0,2	0,099	0,512	$1,38 \cdot 10^{-7}$
30 см – 60 см	1,1	0,2	0,0986	0,502	$9,49 \cdot 10^{-8}$
Глибше 60 см	1,15	0,04	0,1	0,5086	$4,16 \cdot 10^{-8}$

Розглядався шар ґрунту глибиною $L = 1 \text{ м}$. Фіксоване значення напору $H_L = -40 \text{ кПа}$ встановлювалось на нижній межі області моделювання. Кроки скінченно-різницевої сітки були обрані такими, що дорівнюють $\tau = 300 \text{ с}$ і $h = 2,5 \text{ см}$ ($m = 40$).

Для ідентифікації параметрів моделі використовувався алгоритм рою частинок (PSO) [249] з розміром популяції, що дорівнювала $S = 60$. Виконувалось не більше 100 ітерацій алгоритму PSO. Ітераційний процес зупинявся, коли різниця між найбільшим і найменшим значеннями цільової функції серед частинок у рої ставала меншою за 50. Параметри алгоритму динамічної зміни параметрів PSO дорівнювали $n = 10$; $\omega_p = 1,2$; $C = 0,5$. Вихідні коди розробленого програмного забезпечення наведено у [265].

Визначення значень параметрів моделі (7.4)–(7.8), використання яких дозволяє отримати найнижчі значення цільової функції (7.16), проводилось на основі вимірювань у період з 3.07.2019 11:00:00 по 8.07.2019 04:00:00. Окрім класичної моделі, розглядалися 5 варіантів дробово-диференціальних моделей:

- З похідними Капуто–Герасимова за як часом, так і за просторовою змінною ($g_t(t) = t$, $g_z(z) = z$);
- З похідною Капуто–Катугампола за просторовою змінною та похідною Капуто–Герасимова за часовою змінною ($g_t(t) = t$, $g_z(z) = z^\lambda$);
- З похідною Капуто–Катугампола за як часовою, так і просторою змінними ($g_t(t) = t^\gamma$, $g_z(z) = z^\lambda$);

- З ψ -похідною Капуто з сигмоподібним функціональним параметром за просторовою змінною та з похідною Капуто–Герасимова за часовою змінною ($g_t(t) = t$, $g_z(z)$ у формі (8.1));
- З ψ -похідною Капуто з сигмоподібним функціональним параметром за просторовою змінною та з похідною Капуто–Катугампола за часовою змінною ($g_t(t) = t^\gamma$, $g_z(z)$ у формі (8.1)).

Для векторів значень параметрів \vec{x} , отриманих алгоритмом PSO, які дозволяють отримувати найнижчі значення цільової функції (7.16), середні абсолютні похибки $\varepsilon(\vec{x}) = \sqrt{\frac{F(\vec{x})}{N}}$ для усіх 3 датчиків і для кожного датчика окремо наведено в табл. 8.7. Виміряна та змодельована динаміка напорів для деяких датчиків наведена на рис. 8.9, 8.10. У проведених обчислювальних експериментах виконувалось 20 запусків використовуюваного різновиду алгоритму PSO з динамічною зміною його параметрів (п. 7.2) і у якості отриманих векторів \vec{x} значень параметрів моделі вибирались вектори за яких досягалось найнижче значення цільової функції серед 20 запусків.

Таблиця 8.7 – Середня абсолютна похибка ε , м, для різних типів похідних та періоду, починаючи з 3.07.2019 11:00:00

Глибина розміщення датчика	Класична модель	$g_t(t) = t$, $g_z(z) = z$	$g_t(t) = t$, $g_z(z) = z^\chi$	$g_t(t) = t$, g_z має форму (8.1)	$g_t(t) = t^\gamma$, $g_z(z) = z^\chi$	$g_t(t) = t^\gamma$, g_z має форму (8.1)
3 датчика	14,88	6,81	6,13	2,59	5,76	2,63
15 см	18,77	3,54	2,99	2,81	2,60	2,86
35 см	16,72	3,59	4,31	1,77	4,25	1,64
55 см	5,71	10,67	9,24	3,01	8,65	3,15

У розглядуваному випадку складні умови вирощування сільськогосподарських культур призводять до неприпустимих помилок ($\varepsilon = 14,88$ м), отримуваних при використанні класичної моделі ($\alpha = \beta = 1$). Використання дробово-диференціальної моделі з похідними Капуто дозволяє отримати в ~ 2 рази меншу

похибку ($\varepsilon = 6,81 \text{ м}$). Це значення надалі використовувалось як база для порівняння при визначенні ефективності підбору параметрів ψ -похідних Капуто.

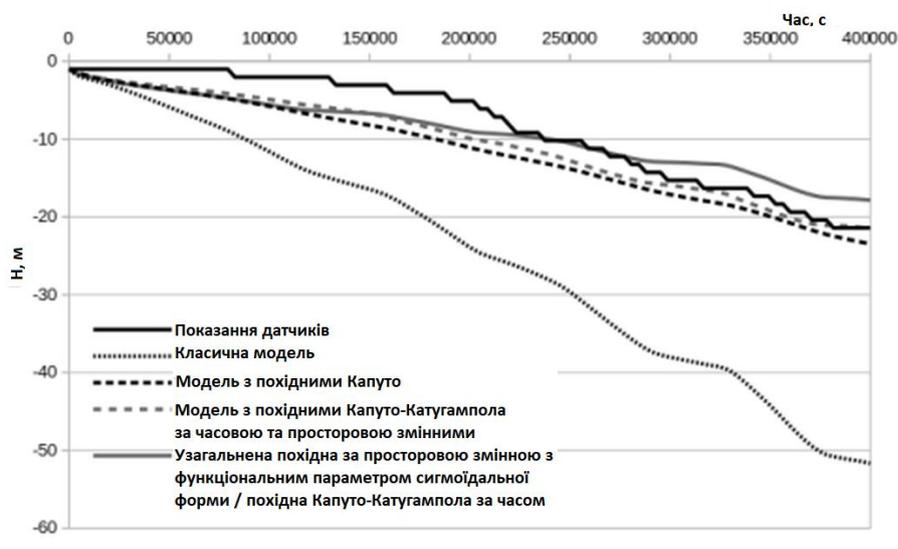


Рисунок 8.9 – Виміряні та змодельовані значення напору на глибині 15 см для періоду, починаючи з 3.07.2019 11:00:00

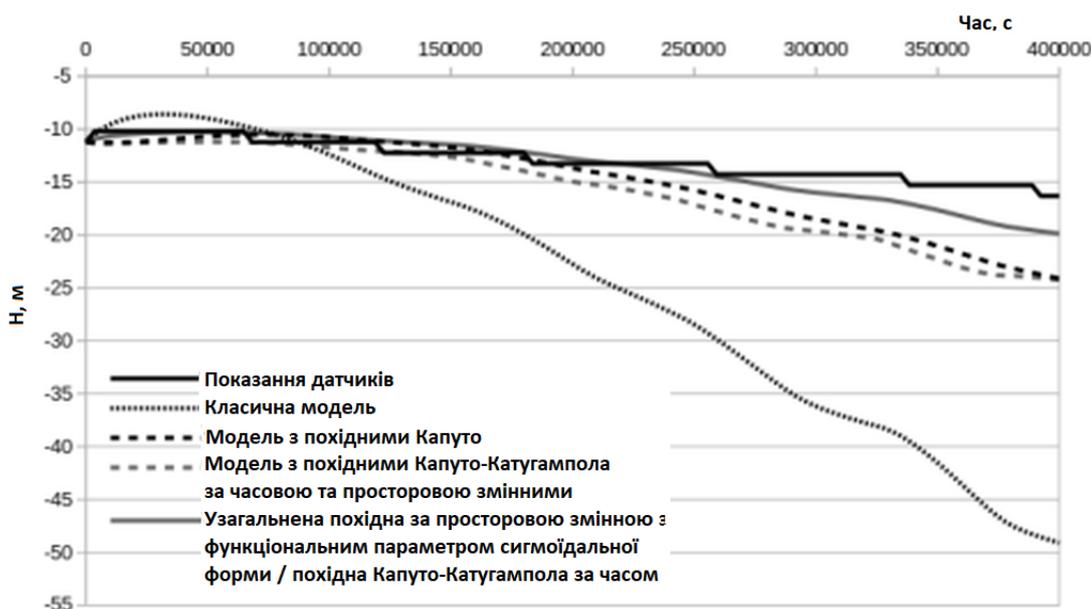


Рисунок 8.10 – Виміряні та змодельовані значення напору на глибині 35 см для періоду, починаючи з 3.07.2019 11:00:00

Так, сумарна похибка була додатково знижена на $\sim 11\%$ при використанні похідної Капуто–Катугампола за просторовою змінною. Використання цієї похідної як для часової, так і для просторової змінної, дозволяє збільшити точність моделювання на $\sim 18\%$. Як видно з рис. 8.9, 8.10, використання степеневі форми

функціональних параметрів дещо змінює форму кривих з огляду як на часову, так і на просторову динаміку.

Використання сигмоподібної форми g_z у похідній за просторовою змінною дозволило отримати розв'язки, які суттєво відрізняються від випадку похідної Капуто–Герасимова і дозволяють отримати в 2,62 раза вищу точність моделювання. Використання похідної Капуто–Катугампола за часовою змінною в цьому випадку вимагає підбору значень 5 параметрів і застосовані варіації алгоритму PSO не дозволили отримати тут вищу точність у порівнянні з випадком підбору значень 4 параметрів при використанні похідної Капуто–Герасимова за часовою змінною.

Як видно з табл. 8.7 мають місце наступні висновки. Ефективність дробово-диференціальної моделі досягається насамперед завдяки кращій відповідності вимірюванням у двох найвищих шарах ґрунту (15 см і 35 см). Оскільки ці шари містять більшу частину коренів рослин, така поведінка моделі збільшує її ефективність при використанні для прогнозування висихання ґрунту з метою планування зрошення. Класична модель для цих шарів суттєво переоцінює швидкість висихання ґрунту і використання таких прогнозів при управлінні зрошенням може сприяти надмірному зволоженню ґрунту, потенційним втратам урожаю та неефективному використанню зрошувальної води.

У порівнянні з класичною моделлю, для датчика, розміщеного на глибині 15 см (рис. 8.9), дробово-диференціальні моделі дозволили отримати в 5–7 разів нижку похибку моделювання. Неточності моделювання за класичною моделлю тут можна пояснити завищенням випаровування, яке, однак, може бути певною мірою виправлене використанням дробово-диференціальних моделей. Для датчика, розміщеного на глибині 35 см, використання сигмоподібної функції у ψ -похідної Капуто дозволяє збільшити точність більш ніж у 2 рази у порівнянні з базовою похідною Капуто–Герасимова та використанням степеневі функції у ядрі оператора. Останнє також справедливо для випадку найглибшого датчика. Однак класична модель описує динаміку напорів для цього датчика в 3 рази точніше, ніж для інших датчиків, і лише модель із сигмоподібною функцією g_z дозволяє більш точно прогнозувати вміст вологи в ґрунті для цього шару. На точність моделювання для

глибин 35 і 55 см найбільше впливають неточності у гідрофізичній моделі ґрунту. Як показують результати експериментів, цей вплив можна пом'якшити, змінивши форму функціонального параметра у похідній за просторовою змінною.

Ефективність алгоритмів PSO

Значення параметрів моделі (7.4)–(7.8), за яких при використанні динамічної варіації алгоритму PSO досягались найнижчі значення цільової функції, наведені в табл. 8.8 разом із їх середніми значеннями та стандартними відхиленнями в серії з 20 запусків алгоритму, середнім часом, витраченим на один запуск, та середнім відносним зменшенням значення цільової функції при застосуванні градієнтного методу (п. 7.2) відносно оптимальних значень, знайдених PSO.

Отримані результати показують, що в межах виконаної серії запусків алгоритму PSO, різниця між найбільшим та найменшим отриманим значенням цільової функції зростає зі збільшенням кількості параметрів, значення яких підбираються. У випадку похідних Капуто–Герасимова (2 параметри), алгоритм PSO збігався до єдиного розв'язку у всіх запусках. У випадку використання похідної Капуто–Катугампола, він збігався до двох близьких підмножин розв'язків, які відрізняються на 8,7% у випадку похідної Капуто–Катугампола лише за просторовою змінною (3 параметри) та на 19,1% у випадку $g_t(t) = t^\gamma$, $g_z(z) = z^\chi$ (4 параметри).

Коли функція g_z мала форму (8.1), різниця між найбільшою та найменшою похибками моделювання становила 23,8% у випадку $g_t(t) = t$ (4 параметри) та 30,1% у випадку $g_t(t) = t^\gamma$ (5 параметрів). У цьому випадку алгоритм PSO генерує серію розв'язків, значення цільової функції яких близько до лінійного залежить від значень параметрів α та σ_2 . Найменші значення цільової функції були отримані для нижчих значень α та вищих значень σ_2 . Спостерігалась також близька до обернено-степеневі ($F(\chi) = a\chi^{-b}$, $a > 0, b > 0$) залежність між значеннями цільової функції та параметром χ – нижчі значення F були досягнуті при вищих значеннях χ .

Додаткове використання градієнтного методу дало лише до 0,8% збільшення точності моделювання.

Таблиця 8.8 – Оптимальні значення параметрів, отримані за допомогою динамічного алгоритму PSO та середнє відносне зменшення F при додатковому використанні градієнтного методу

	F	β	α	χ	σ_2	γ
$g_t(t) = t^\gamma$, g_z має форму (8.1)						
Середні значення	2861,24	0,693	0,460	16,231	0,726	1,394
Стандартне відхилення	303,169	0,136	0,162	3,103	0,124	0,270
Значення для мінімального значення F	2331,61	0,722	0,320	20,891	0,775	1,193
Середній час на 1 запуск, с – 0,980	Середнє відносне зменшення F при застосуванні градієнтного методу – 0,83%					
$g_t(t) = t^\gamma$, $g_z(z) = z^\chi$						
Середні значення	12633	0,586	0,481	2,548		1,481
Стандартне відхилення	978,793	0,285	0,071	3,935		0,608
Значення для мінімального значення F	11156,8	0,451	0,528	0,047		2,000
Середній час на 1 запуск, с – 0,941	Середнє відносне зменшення F при застосуванні градієнтного методу – 0,24%					
$g_t(t) = t$, g_z має форму (8.1)						
Середні значення	2507,73	0,874	0,389	18,228	0,760	
Стандартне відхилення	211,547	0,014	0,101	3,846	0,045	
Значення для мінімального значення F	2252,48	0,880	0,259	24,009	0,806	
Середній час на 1 запуск, с – 0,450	Середнє відносне зменшення F при застосуванні градієнтного методу – 0,29%					
$g_t(t) = t$, $g_z(z) = z^\chi$						
Середні значення	13348,8	0,732	0,493	4,035		
Стандартне відхилення	498,696	0,244	0,071	4,184		
Значення для мінімального значення F	12634,8	0,948	0,519	0,051		
Середній час на 1 запуск, с – 0,430	Середнє відносне зменшення F при застосуванні градієнтного методу – 0,60%					
$g_t(t) = t$, $g_z(z) = z$						
Середні значення	15582,7	1,000	0,536			
Стандартне відхилення	0,000	0,000	0,000			
Значення для мінімального значення F	15582,7	1,000	0,536			
Середній час на 1 запуск, с – 0,431	Середнє відносне зменшення F при застосуванні градієнтного методу – 0,00%					

Для оцінки ефективності динамічної зміни параметрів PSO була проведена серія обчислювальних експериментів, використовуючи базовий алгоритм PSO з різними значеннями параметрів для випадку сигмоподібної форми функції g_z у похідній за просторовою змінною та похідній Капуто–Герасимова за часовою змінною (табл. 8.9). Значення ω було зафіксовано і дорівнювало 0,8, тоді як значення (φ_p, φ_g) дорівнювали (0,25;0,25), (0,1;0,4), (0,4;0,1). Це відповідало

діапазоном, в яких значення цих параметрів змінювались у динамічній варіації алгоритму PSO. Також використовувалась динамічна варіація алгоритму PSO без випадкової повторної ініціалізації частинок.

Таблиця 8.9 – Оптимальні значення параметрів, отримані алгоритмом PSO для випадку сигмоподібної функції g_z у похідній за просторовою змінною

	F	β	α	χ	σ_2
$\omega = 0,8, \varphi_p = 0,25, \varphi_g = 0,25$					
Середні значення	2395,03	0,819	0,346	19,368	0,775
Стандартне відхилення	164,920	0,197	0,087	3,745	0,036
Значення для мінімального значення F	2239,88	0,258	0,244	25,405	0,813
$\omega = 0,8; \varphi_p = 0,1, \varphi_g = 0,4$					
Середні значення	2247,86	0,881	0,254	24,498	0,810
Стандартне відхилення	17,821	0,004	0,021	1,619	0,006
Значення для мінімального значення F	2232,51	0,882	0,237	25,902	0,815
$\omega = 0,8; \varphi_p = 0,4, \varphi_g = 0,1$					
Середні значення	2520,08	0,874	0,404	17,392	0,751
Стандартне відхилення	244,521	0,007	0,085	2,455	0,040
Значення для мінімального значення F	2318,87	0,881	0,313	20,060	0,790
Динамічна варіація алгоритму PSO без випадкової повторної ініціалізації					
Середні значення	2799,27	0,814	0,396	16,822	0,870
Стандартне відхилення	796,326	0,197	0,238	10,992	0,280
Значення для мінімального значення F	2246,71	0,890	0,226	25,758	0,821

Найменші отримані значення цільової функції серед 20 запусків у всіх випадках були близькі, проте стандартне відхилення знижувалось зі збільшенням φ_g . Оскільки динамічна зміна φ_g і φ_p спрямована на концентрацію пошуку в околицях локальних мінімумів на початкових етапах процесу та на пришвидшення збіжності до глобального мінімуму на завершальних етапах, базовим алгоритмом PSO з фіксованим високим значенням φ_g у проведеному експерименті була отримана вища швидкість збіжності, ніж при використанні динамічної варіації алгоритму у випадку сигмоподібної функції g_z . Це може бути пояснено відсутністю розділених локальних мінімумів цільової функції в цьому випадку. Використання динамічної варіації алгоритму PSO без випадкової повторної ініціалізації частинок призводить до $\sim 3,5$ рази вищого стандартного відхилення

отриманих значень цільової функції у порівнянні з випадком використання цієї процедури, підтверджуючи її ефективність.

У наступній серії обчислювальних експериментів порівнювалась динамічна варіація алгоритму PSO зі значеннями основних параметрів PSO, якими у попередній серії експериментів було отримано найменше значення цільової функції ($\omega = 0,8; \varphi_p = 0,1; \varphi_g = 0,4$). Отримані результати для всіх розглянутих комбінацій типів дробових похідних наведені в табл. 8.10.

Для випадку g_z у формі (8.1) базовий алгоритм PSO був ефективнішим, ніж його динамічна варіація як для $g_t(t) = t$, так і для $g_t(t) = t^\gamma$. Однак для $g_z(z) = z^\chi$, у ситуації, коли спостерігались декілька розділених локальних мінімумів цільової функції, висока фіксована величина φ_g змушує алгоритм PSO частіше збігатись до гірших локальних мінімумів. У випадку $g_t(t) = t$, ця особливість навіть не дозволила отримати найнижче значення, отримане при використанні динамічної варіації у всіх виконаних запусках.

Таблиця 8.10 – Оптимальні значення параметрів, отримані алгоритмом PSO з

$$\omega = 0,8; \varphi_p = 0,1; \varphi_g = 0,4$$

	F	β	α	χ	σ_2	γ
$g_t(t) = t^\gamma, g_z$ має форму (8.1)						
Середні значення	2298,637	0,606	0,300	22,523	0,773	1,493
Стандартне відхилення	31,273	0,159	0,041	2,567	0,019	0,329
Значення для мінімального значення F	2262,770	0,566	0,243	26,378	0,786	1,521
$g_t(t) = t^\gamma, g_z(z) = z^\chi$						
Середні значення	13195,46	0,365	0,476	6,531		1,845
Стандартне відхилення	761,966	0,226	0,041	2,310		0,491
Значення для мінімального значення F	11147,30	0,447	0,549	0,040		2,000
$g_t(t) = t, g_z$ має форму (8.1)						
Середні значення	2247,865	0,881	0,254	24,498	0,810	
Стандартне відхилення	17,821	0,004	0,021	1,619	0,006	
Значення для мінімального значення F	2232,510	0,882	0,237	25,902	0,815	
$g_t(t) = t, g_z(z) = z^\chi$						
Середні значення	13758,17	0,504	0,477	7,516		
Стандартне відхилення	2,119	0,001	0,011	0,204		
Значення для мінімального значення F	13757,50	0,504	0,481	7,451		

Точність прогнозування динаміки вологості у наступних циклах зрошення

Отримані оптимальні значення параметрів моделі (7.4)–(7.8), наведені в табл. 8.8, були використані для прогнозування висихання ґрунту під час циклів зрошення, наступних за тим, що використовувався для ідентифікації параметрів (період 1): починаючи з 9.07.2019 03:00:00 (період 2) і з 28.07.2019 22:00:00 (період 3).

Середні абсолютні похибки з урахуванням усіх 3 датчиків і кожного датчика окремо для цих періодів наведені в табл. 8.11 для всіх розглянутих комбінацій дробових похідних у рівнянні (7.4). Виміряна та змодельована динаміка напорів для глибини 15 см наведена на рис. 8.11 для періоду 2 та на рис. 8.12 для періоду 3.

Таблиця 8.11 – Середня абсолютна похибка ε , m для різних типів похідних та періодів, починаючи з 9.07.2019 03:00:00 (період 2) і з 28.07.2019 22:00:00 (період 3)

Глибина розміщення датчика	Класична модель	$g_t(t) = t$, $g_z(z) = z$	$g_t(t) = t$, $g_z(z) = z^\lambda$	$g_t(t) = t$, g_z має форму (8.1)	$g_t(t) = t^\gamma$, $g_z(z) = z^\lambda$	$g_t(t) = t^\gamma$, g_z має форму (8.1)
Період 2						
3 датчики	26,97	12,50	11,893	8,93	11,00	8,40
15 см	30,35	8,32	7,51	3,32	5,32	2,62
35 см	34,71	15,28	14,60	10,42	11,21	8,71
55 см	7,51	12,88	12,44	10,92	14,47	11,36
Період 3						
3 датчики	26,39	10,89	9,78	10,20	9,44	10,23
15 см	34,02	12,34	10,05	12,80	9,88	12,94
35 см	29,46	10,71	9,82	11,07	9,74	10,91
55 см	8,04	9,43	9,46	5,06	8,64	5,27

Для періоду 2, який безпосередньо слідує за періодом 1, спостерігалися подібні тенденції:

- Більш ніж у 2 рази менша похибка моделювання при застосуванні дробово-диференціальної моделі у порівнянні з класичною моделлю;
- Зниження похибки на $\sim 10\%$ при використанні похідних Капуто–Катугампола у порівнянні з випадком похідних Капуто–Герасимова;

- Найнижча похибка, отримана у випадку, коли функція g_z мала форму (8.1). У випадку періоду 2 похибка зменшувалась у 1,48 рази у порівнянні з випадком похідних Капуто–Герасимова, тоді як для періоду 1 це зменшення склало 2,62 рази;
- Точність для конкретних датчиків зменшується з глибиною для всіх розглянутих варіантів дробово-диференціальної моделі та зростає для класичної моделі.

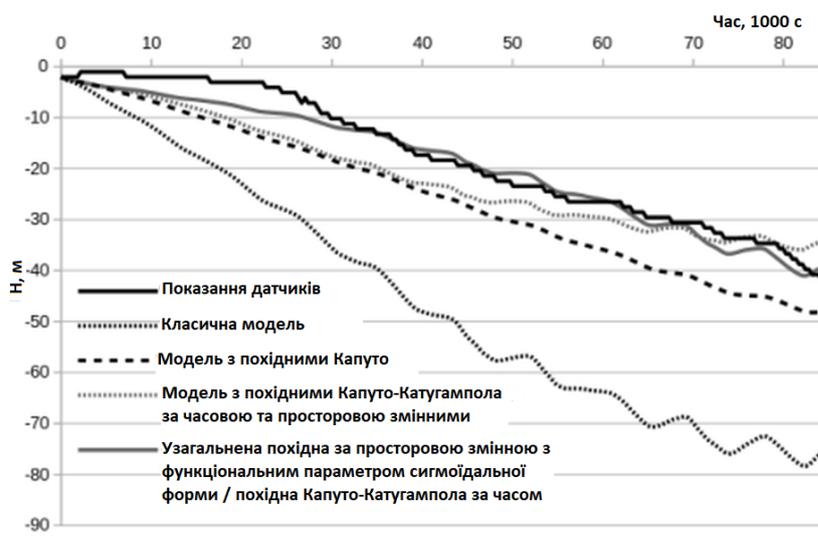


Рисунок 8.11 – Виміряні та змодельовані значення напорів на глибині 15 см для періоду, що починається з 9.07.2019 03:00:00

Початок періоду 3–20 днів після закінчення періоду 1, який використовувався для ідентифікації параметрів. У цьому випадку дробово-диференціальна модель все ще дозволяє отримати у 2,5 рази меншу похибку моделювання, ніж класична модель, проте вплив форми функцій $g_t(t)$ та $g_z(z)$ на точність моделювання стає нижчим. Найбільше зниження похибки на 13% у порівнянні з випадком $g_t(t) = t$, $g_z(z) = z$ було отримано при $g_t(t) = t^\gamma$, $g_z(z) = z^\lambda$. Точність прогнозування для конкретних датчиків тут знижується зі збільшенням глибини як для класичної, так і для дробово-диференціальної моделей. Для практично важливого верхнього шару ґрунту похибка збільшується у ~ 3 рази у порівнянні з періодом 1. Як видно з рис. 8.11, 8.12, це пов'язано із завищенням швидкості висихання ґрунту усіма моделями. Джерелом таких похибок може бути більш інтенсивна транспірація рослин через їх зростання

протягом 20-денного періоду. Ці зміни неможливо врахувати за допомогою використовуваних у обчислювальних експериментах формул Штойко оцінки евапотранспірації.

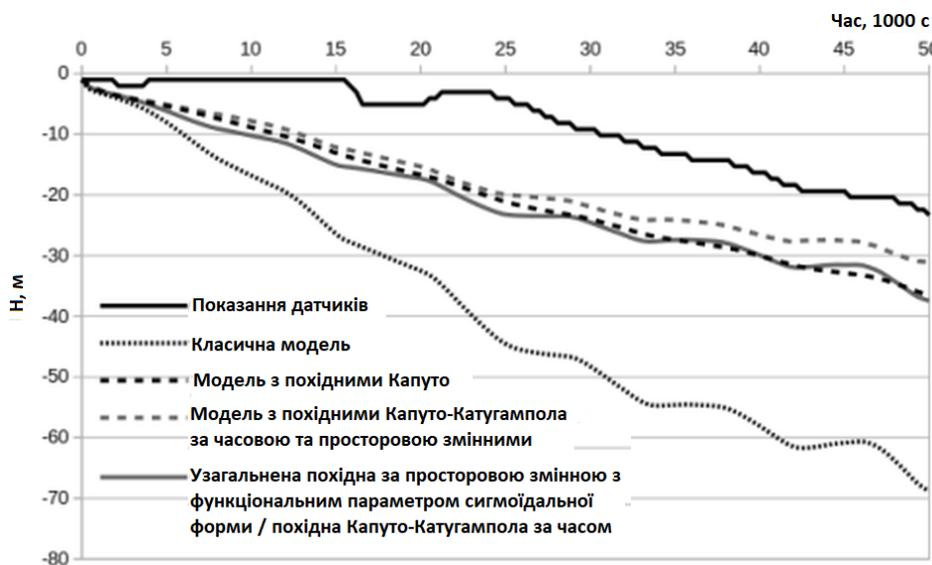


Рисунок 8.12 – Виміряні та змодельовані значення напорів на глибині 15 см для періоду, починаючи з 28.07.2019 22:00:00

Час виконання обчислень

При виконанні обчислень на ~ 1500 кроках за часом, час (табл. 8.8) отримання одного розв'язку прямої задачі є в ~ 2 рази вищим у випадку, коли розглядається похідна Капуто–Катугампола за часовою змінною, через необхідність наближеного обчислення значень інтегралів $\gamma_s^{(i)}$, які обчислюються у замкненому вигляді при $g_t(t) = t$. У порівнянні з розв'язанням початково-крайової задачі щодо класичного рівняння Річардса ($\alpha = \beta = 1$), час, що витрачається на розв'язання задачі для рівняння з нелокальною дробовою похідною Капуто–Герасимова, тобто випадку $g_t(t) = t, g_z(z) = z$, на 36% більший. Час, що витрачається на обчислення значень інтегралів $\gamma_s^{(i)}$ та $b_s^{(j)}$, складає 13–16% від загального часу розв'язання у випадку, коли вони знаходяться аналітично, та підвищується до 52–61%, коли їх потрібно обчислювати наближено.

Було перевірено, як час розрахунків змінюється в залежності від номера кроку за часом при застосуванні представлення інтегралів у вигляді ряду (п. 6.1) у

порівнянні з використанням кубатурної формули з рекурсивним підрозбиттям (п. 6.1). Загальний час, витрачений на проведення обчислень, та час, витрачений на обчислення значень $\gamma_s^{(i)}$ і $b_s^{(j)}$ у випадку $g_t(t) = t^\gamma$ та функції g_z у формі (8.1), наведено на рис. 8.13. У цьому випадку складність алгоритму рекурсивного підрозбиття вища, ніж складність схеми розкладення у ряди, при моделюванні до ~ 750 -го кроку за часом. Алгоритм рекурсивного підрозбиття значно повільніший на початкових етапах, проте час його роботи збільшується лінійно, але з меншим нахилом, ніж у випадку алгоритму розкладення у ряди, час роботи якого демонструє (очікувано згідно з [39, 46]) незначний логарифмічний характер зростання. Середній загальний час при розв'язанні розглядуваної початково-крайової задачі був на $\sim 7\%$ нижчим у випадку, коли для обчислення інтегралів за складної форми функціонального параметра у ψ -похідних Капуто використовувався алгоритм розкладення у ряди.

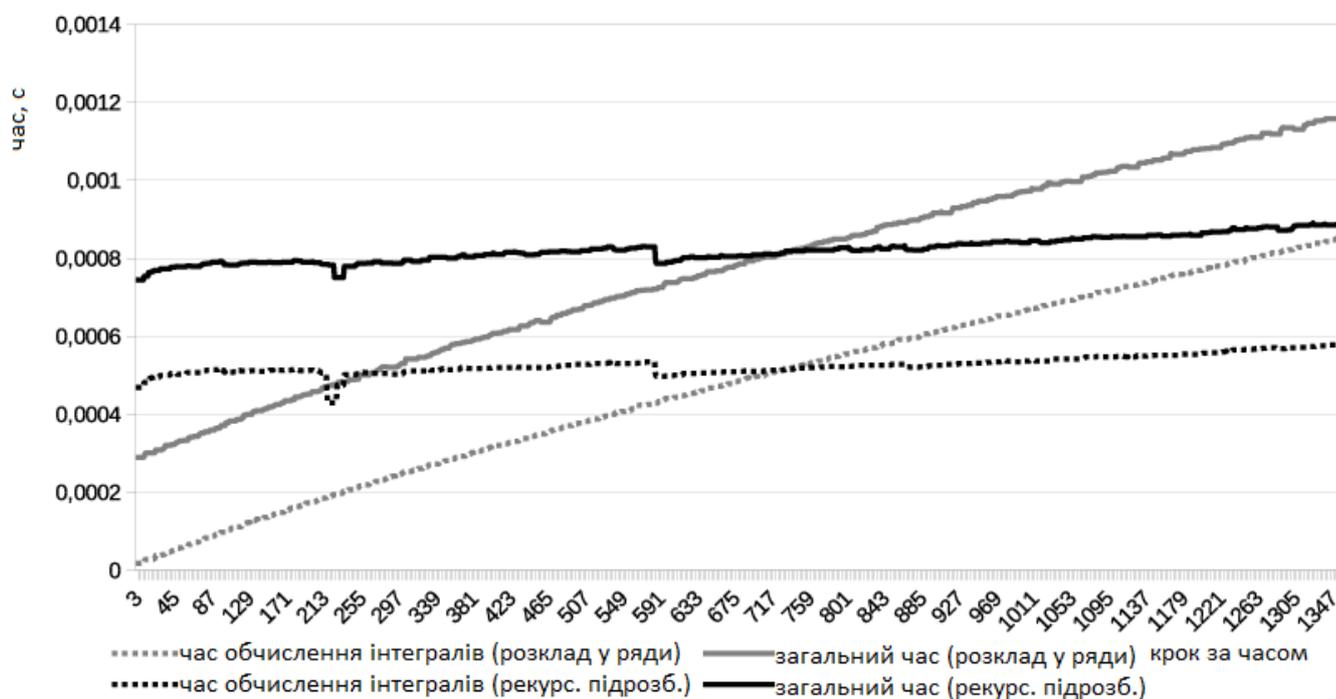


Рисунок 8.13 – Загальний час, витрачений на обчислення в залежності від номера кроку за часом, та час, витрачений на обчислення значень інтегралів $\gamma_s^{(i)}$

та $b_s^{(j)}$ для $g_t(t) = t^\gamma$, функції g_z у формі (8.1)

8.3. Моделювання з використанням ψ -похідної Капуто та пошуком аналітичної форми функціонального параметра

Гнучкість моделей, що містять ψ -похідні Капуто, забезпечується, зокрема, можливістю автоматичного вибору форми їх параметрів, зокрема, алгоритмом генетичного програмування (п. 7.3) [48]. Практична перевірка цього твердження була виконана [48] з використанням набору даних, отриманого в ході експерименту, описаного в [36], п. 8.2.

Аналогічно п. 8.2, будемо розглядати два періоди часу, коли ґрунт просушувався між зрошеннями. Перший період з 3.07.2019 11:00:00 по 8.07.2019 04:00:00 був використаний для вибору форми функціональних параметрів, тоді як наступний період з 9.07.2019 03:00:00 по 19.07.2019 07:00:00 – для оцінки точності прогнозування. Вихідні коди розробленого програмного забезпечення наведено у [265].

Значення параметрів скінченно-різницевої схеми та моделі за виключенням параметрів дробових похідних приймалися аналогічно описаному у п. 8.2. Значення порядків дробових похідних приймалися рівними $\alpha = 0,948$ і $\beta = 0,519$ [36]. У якості бази для порівняння використовувались розв'язки, отримані за функціональних параметрів [36] дробових похідних у формі $g_t(t) = t$, $g_z(z) = z^{0,051}$ (варіант з найменшим значенням цільової функції, отриманий у [36] для випадку $g_t(t) = t$, $g_z(z) = z^x$).

Значення параметрів алгоритму генетичного програмування приймалися наступними:

$$S_p = 100, d_m = 2, p_m = 0,05; \varepsilon_{g1} = 0,1; \varepsilon_{g1} = 10^{-10}, i_m = 5000, c_{max} = 10, b_0 = 1,5.$$

$$\text{Середні абсолютні похибки моделювання } \varepsilon = \left(\frac{1}{N_s} F(g_t, g_z) \right)^{1/2} (m), \text{ де } F$$

визначається згідно з (7.16) з підсумовуванням за даними конкретного датчика (N_s – кількість відповідних його показань), наведені в табл. 8.12 для випадку найменшого значення цільової функції, описаного в [36] (контрольний випадок), та чотирьох запусків алгоритму генетичного програмування. Відповідна динаміка напорів для

датчика, розміщеного у найвищому шарі ґрунту, представлена на рис. 8.14 для випадку значень параметрів моделі, за яких досягалось найменше значення цільової функції серед чотирьох запусків алгоритму.

У всіх випадках форма функціонального параметра похідної за часом, за якої досягалось найменше значення цільової функції, дорівнювала $g_t(t) = t$. Форма функціонального параметра $g_z(z)$ похідної за просторовою змінною, використання якої дозволило отримати найменше значення цільової функції (7.16), мала вигляд [48]

$$\log_{4,60916}(1 + \log_{3,37998}(1 + 310,245 * \log_{2,88761}(1 + \log_{4,60916}(1 + 310,245 * \log_{3,37998}(1 + 11,0135 * x^2 * \log_{3,37998}(1 + 11,0135 * x^3) * \log_{3,37998}(1 + \log_{4,60916}(1 + \log_{2,88761}(1 + \log_{4,48114}(1 + x)))))))))).$$

Графічно він представлений на рис. 8.15, а значення $\gamma_s^{(i)}$ для $i = 40$ наведені на рис. 8.16.

Таблиця 8.12 – Середні абсолютні похибки моделювання за період, починаючи з 3.07.2019 11:00:00

Глибина	Контрольний випадок	Запуск 1	Запуск 2	Запуск 3	Запуск 4
15 см	0,536	0,211	0,207	0,141	0,197
35 см	0,670	0,494	0,596	0,616	0,427
55 см	0,745	0,673	0,616	0,522	0,687
Середнє значення	0,650	0,460	0,473	0,426	0,437
Зниження похибки		29,32%	27,26%	34,44%	32,81%

Використання ψ -похідної Капуто з оптимізованою формою функціонального параметра дозволило отримати зменшення загальної похибки моделювання до 34% . Підвищення точності було найвищим для верхнього шару ґрунту і зменшувалось для глибших шарів (табл. 8.12). У той час як у контрольному випадку базова модель завищувала швидкість висихання верхнього шару ґрунту (рис. 8.14), використання оптимізованої форми функціонального параметра дало його більш точний опис без збільшення похибки моделювання для глибших шарів. Такий ефект був досягнутий

завдяки нижчим значенням функціонального параметра (рис. 8.15) та інтегралів $\gamma_s^{(i)}$ (рис. 8.16) поблизу θ . Загалом можна стверджувати, що базова форма g_z у вигляді $z^{0,051}$ описує інтенсивний взаємовплив усіх шарів ґрунту, що можна пояснити різними факторами, пов'язаними з будовою ґрунту або взаємодією рослин із ним. Оптимізоване значення g_z знижує вплив певного верхнього шару на процеси в шарах під ним.

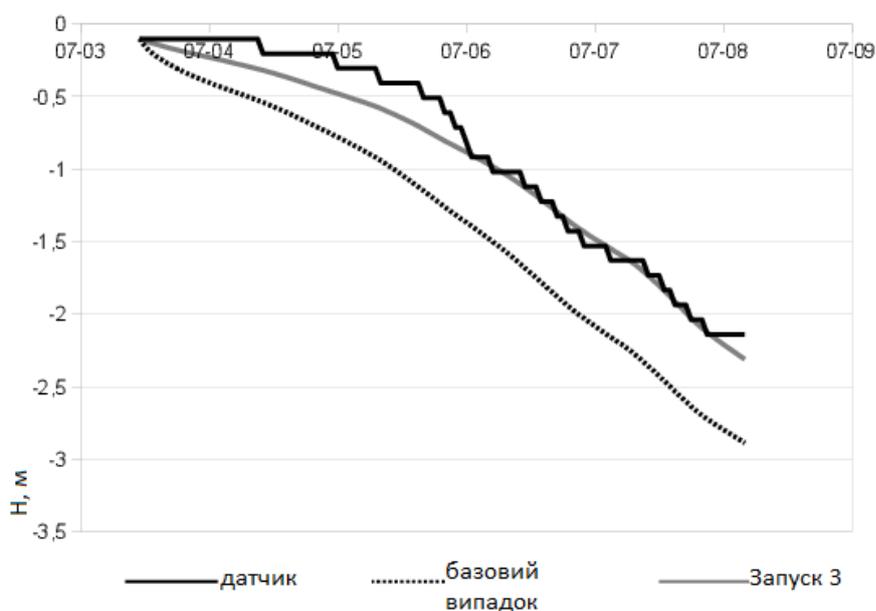


Рисунок 8.14 – Виміряні та змодельовані значення напорів на глибині 15 см за період, що починається з 3.07.2019 11:00:00

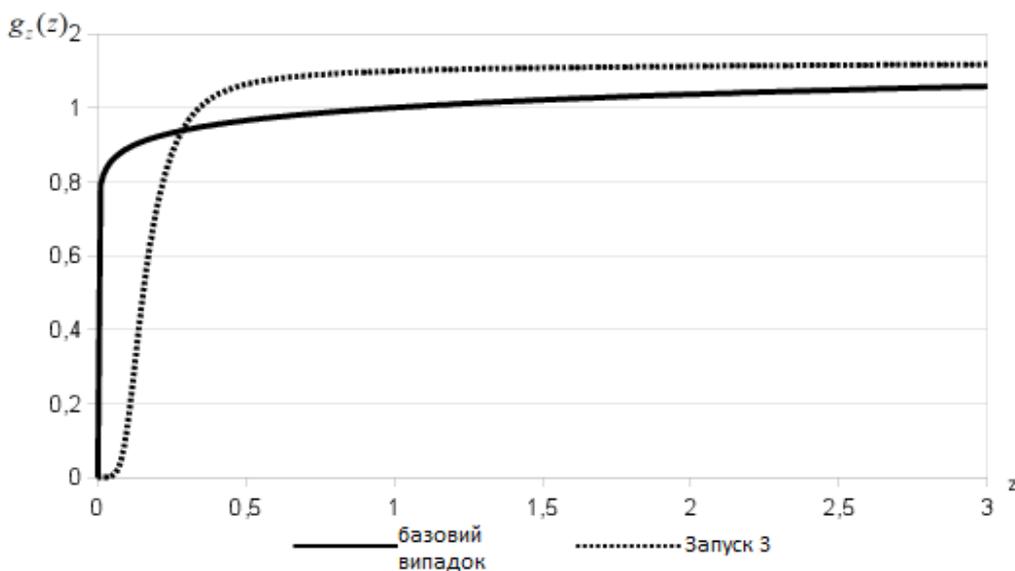


Рисунок 8.15 – Форми функціональних параметрів похідної за просторовою змінною

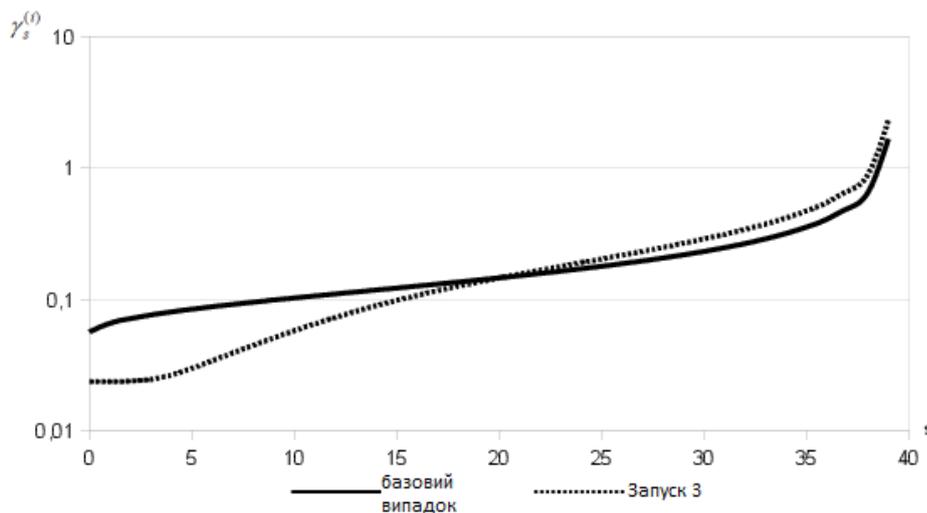


Рисунок 8.16 – Значення інтегралів $\gamma_s^{(i)}$ для $i = 40$

Як і в [36], використаємо отримані оптимізовані значення $g_z(z)$ для прогнозування висихання ґрунту під час зрошувального циклу, який починається з 9.07.2019 03:00:00 і який є наступним за тим, що використовувався для ідентифікації параметрів моделі. Середні абсолютні похибки у цьому випадку наведені у табл. 8.13, а динаміка напорів для верхнього датчика у випадку найменшого значення цільової функції, отриманого алгоритмом генетичного програмування, наведена на рис. 8.17.

Таблиця 8.13 – Середні абсолютні похибки моделювання за період, починаючи з 9.07.2019 03:00:00

Глибина	Контрольний випадок	Запуск 1	Запуск 2	Запуск 3	Запуск 4
15 см	1,051	0,725	0,807	0,792	0,611
35 см	1,690	1,525	1,603	1,611	1,429
55 см	1,067	1,114	1,061	1,025	1,182
Середнє значення	1,269	1,121	1,157	1,143	1,074
Зменшення похибки		11,65%	8,85%	9,96%	15,38%

У цьому випадку було отримано до 15% зниження похибки моделювання, що доводить можливість використання запропонованої розрахункової схеми для більш точного прогнозування динаміки вологості ґрунту. Для другого зрошувального циклу

спостерігались ті самі тенденції, що і для періоду, який використовувався для ідентифікації параметрів: підвищення точності було нижчим для глибших шарів ґрунту і модель описувала повільніше висихання верхнього шару ґрунту.

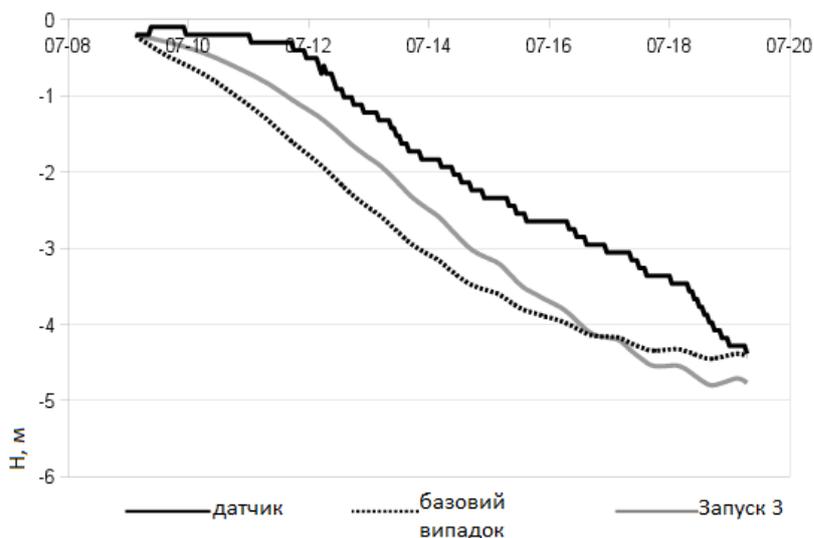


Рисунок 8.17 – Виміряні та змодельовані значення напорів на глибині 15 см за період, починаючи з 9.07.2019 03:00:00

8.4. Висновки до розділу 8

Наведені у п. 8.1 результати проведеного для умов зрошення дощуванням на півдні України моделювання вологоперенесення та аналізу точності за класичним одновимірним рівнянням Річардса та його дробово-диференціальним узагальненням свідчать, що при визначенні середньої об'ємної вологості кореневмісного шару ґрунту найбільша абсолютна похибка склала 9,5%, а середня – 1,9%. Використання дробово-диференціальної моделі (пп. 8.1.4) дозволило на 7% знизити похибку опису динаміки напорів, а змодельована сумарна норма зрошення відрізнялася від фактичної не більше ніж на 12%.

Проведені обчислювальні експерименти (пп. 8.1.3) дозволило також визначити, що найточнішим при застосуванні у пропонованій схемі моделювання методом оцінки евапотранспірації є метод Іванова з домноженням на коефіцієнт культури, лінійно залежним від спектрального індексу біомаси NDVI. Деяко вищу точність моделювання можна отримати використовуючи лінійну комбінацію оцінок за методами Іванова, Штойка та Пенмана–Монтейта. Саме такий підхід

пропонується використовувати для розрахунку евапотранспірації у випадку, коли найточніший метод невідомий.

При відсутності експериментально визначених параметрів або даних про гранулометричний склад ґрунту, ідентифікувати параметри невідомих коефіцієнтів моделей пропонується розв'язуючи обернені задачі метаевристичним методом рою частинок. У пропонованому підході для ідентифікації невідомих коефіцієнтів моделей вологоперенесення використовуються вимірювання вологості ґрунту чи всмоктуючих тисків на початковому етапі вегетації рослин. Це дозволяє виконувати коротко чи середньострокові прогнози водопотреби для використання їх при економічному плануванні.

Застосування дробово-диференціальної моделі вологоперенесення з ψ -похідними Капуто за часовою та просторовою змінними для опису та прогнозування динаміки вологості ґрунту в складних гідрогеологічних умовах показало її здатність значно підвищити точність моделювання.

Використання різних форм функціональних параметрів у ψ -похідній Капуто (п. 8.2) у порівнянні з базовою похідною Капуто–Герасимова збільшує точність моделювання завдяки додатковим параметрам, значення яких можуть бути підібраними відповідно до вимірюваних даних. Функціональний параметр у степеневій формі (похідна Капуто–Катугампола) має один додатковий числовий параметр і його використання знижувало похибку на $\sim 19\%$. Використання складнішої сигмоподібної функції з двома додатковими параметрами дозволило отримати до 2,6-кратного зниження середньої абсолютної похибки.

Тенденції у розподілі похибок залишалися подібними при прогнозуванні висихання ґрунту протягом зрошувальних циклів, наступних за цикл, дані зібрані в межах якого використовувались для ідентифікації параметрів моделі. Це доводить здатність та потенційну ефективність використання дробово-диференціальної моделі вологоперенесення з ψ -похідною Капуто з сигмоподібним функціональним параметром при автоматизованому плануванні зрошення.

Підбір аналітичної форми функціонального параметра дробової ψ -похідної Капуто на основі алгоритму генетичного програмування (п. 8.3) пропонує додаткові

можливості для адаптації моделей вологоперенесення до польових умов. Використання запропонованої процедури автоматичного вибору форми функціонального параметра дозволило отримати до 34% зниження похибки моделювання для набору даних, що використовувався для ідентифікації параметрів, і до 15% для тестового набору даних. Окрім підвищення точності, отримані оптимізовані форми функціональних параметрів можуть надавати додаткову інформацію про поточний стан непорушеного ґрунту, доповнюючи дані лабораторного дослідження відібраних зразків.

Результати розділу 8 опубліковано в роботах [36, 48, 186, 187].

Джерела, що використані у розділі 8

Для написання даного розділу було використано 44 джерела [8], [9], [25], [33], [36], [39], [46], [48], [58], [72], [73], [99], [126], [131], [133], [134], [136], [137], [155], [156], [161], [165], [173–175], [185–187], [197], [210], [216], [220], [227], [249], [260], [261], [264], [265], [291], [294–296], [309], [321], посилання на які зазначені в тексті розділу.

Висновки

У дисертаційній роботі розв'язано науково-технічну проблему створення методів та засобів математичного та комп'ютерного моделювання аномальних гідрогеоміграційних процесів на основі дробово-диференціального підходу шляхом побудови нових математичних моделей, що враховують фактори часової та просторової нелокальності при перебігу складних процесів масотеплоперенесення у геосередовищах, та побудови класу високопродуктивних алгоритмів розв'язання відповідних цим моделям початково-крайових задач, зокрема, паралельних алгоритмів для систем зі спільною та розподіленою пам'яттями, які дозволяють суттєво збільшити швидкість моделювання відповідних процесів.

Отримано наступні нові наукові результати:

1. сформульовано дробово-диференційні математичні моделі з похідними дробового порядку Капуто–Герасимова та k -похідною Капуто, поставлено початково-крайові задачі, розроблено скінченно-різницеві алгоритми їх розв'язання та чисельно досліджено особливості динаміки процесів конвективної дифузії розчинних речовин при двовимірній усталеній фільтрації на основі цих дифузійних математичних моделей.

2. побудовано дробово-диференційні математичні моделі процесів фільтраційної консолідації насичених сольовими розчинами ґрунтових середовищ, поставлено відповідні цим моделям початково-крайові задачі, розроблено скінченно-різницеві алгоритми їх розв'язання та чисельно досліджено особливості динаміки цих процесів, зокрема для моделі фільтраційної консолідації з урахуванням явища хімічного осмосу, що містить похідні Капуто–Фабріціо;

3. побудовано математичні моделі, виконано постановки початково-крайових задач, розроблено скінченно-різницеві алгоритми їх розв'язання та чисельно досліджено особливості динаміки нелокальних конвективно-дифузійних процесів при врахуванні явищ масообміну між частинками у мобільній та немобільній фазах згідно з різними законами кінетики масообміну як у одновимірному, так і у двовимірному наближенні, зокрема, на основі моделей, що містять дробову похідну Капуто–Катугампола;

4. розроблено клас паралельних алгоритмів для систем з розподіленою пам'яттю щодо локально-одновимірних схем розщеплення при їх застосовуванні до дробово-диференціальних рівнянь, які містять похідну Капуто–Герасимова за часовою змінною, з використанням одновимірного та червоно-чорного блокового розподілу даних;

5. розроблено клас паралельних алгоритмів для графічних процесорів у випадку моделей перенесення з похідними Капуто–Герасимова та ψ -похідною Капуто як за часовою, так і за просторовими змінними, зокрема таких, що використовують можливості низькорівневої оптимізації;

6. розроблено обчислювальні процедури апроксимації похідних Капуто–Герасимова, Атангана–Балеану та ψ -похідної Капуто на основі розкладення у ряди ядер інтегральних операторів та методики розділення змінних, використання яких при моделюванні геоміграційних процесів дозволяє знизити обчислювальну складність на одному кроці за часом з лінійної до логарифмічної чи близької до сталої.

7. на основі обчислювальних процедур апроксимації похідних Капуто–Герасимова та ψ -похідної Капуто розроблено паралельні алгоритми розв'язання багатовимірних початково-крайових міграційних задач з відповідними похідними за просторовими змінними, що дискретизуються згідно з локально-одновимірними скінченно-різницевиими схемами, на системах з розподіленою пам'яттю. Запропонована схема розподілення даних, що є поєднанням одновимірної та червоно-чорної схем, і дозволяє отримувати вищу за схему з одновимірним розподілом швидкодію при невеликій кількості залучених обчислювальних ресурсів та більшу масштабованість у порівнянні з червоно-чорною схемою.

8. для тривимірного рівняння дифузії з ψ -похідними Капуто як за часом, так і за просторовими змінними, досліджено точність та швидкодію неявних скінченно-різницевих схем та схем розщеплення при їх застосуванні разом з алгоритмами, направленими на підвищення швидкодії проведення обчислень та розроблено алгоритм автоматичного вибору оптимальної обчислювальної схеми;

9. на основі розроблених методів та алгоритмів, розв'язано ряд задач дробово-диференціального моделювання вологоперенесення у зрошувальному землеробстві згідно з моделями, що містять похідні Капуто–Герасимова та ψ -похідні Капуто. Запропонована методика їх прикладного застосування, у якій для ідентифікації параметрів методами рою частинок та генетичного програмування використовуються вимірювання вологості ґрунту чи всмоктуючих тисків на початковому етапі вегетації рослин, після чого проводиться прогнозне моделювання на декілька циклів зрошення з повторною ідентифікацією параметрів за необхідності. Це дозволяє виконувати коротко чи середньострокові прогнози водопотреби для використання їх при економічному плануванні, а також визначати найточнішу у конкретних умовах та на конкретній стадії розвитку рослин модель оцінювання евапотранспірації.

Список використаних джерел

1. Abdelkawy MA, Ahmed EA, Sanchez P. A method based on Legendre pseudo-spectral approximations for solving inverse problems of parabolic types equations. *Math. Sci. Lett.* 2015; 4: 81–90.
2. Abro KA, Gómez-Aguilar JF. A comparison of heat and mass transfer on a Walter's B fluid via Caputo-Fabrizio versus Atangana-Baleanu fractional derivatives using the Fox H function. *Eur. Phys. J. Plus* 2019; 134: 101. doi: [10.1140/epjp/i2019-12507-4](https://doi.org/10.1140/epjp/i2019-12507-4)
3. Abro KA, Khan I, Tassaddiq A. Application of Atangana-Baleanu fractional derivative to convection flow of MHD Maxwell fluid in a porous medium over a vertical plate. *Math. Model. Nat. Phenom.* 2018; 13: 1. doi: [10.1051/mmnp/2018007](https://doi.org/10.1051/mmnp/2018007)
4. Adams EE, Gelhar LW. Field study of dispersion in a heterogeneous aquifer: 2. Spatial moments analysis. *Water Resour. Res.* 1992; 28: 3293–3307. doi: [10.1029/92WR01757](https://doi.org/10.1029/92WR01757)
5. Agarwal R, Yadav MP, Agarwal RP. Fractional flow equation in fractured aquifer using dual permeability model with non-singular kernel. *Arab.J.Math.* 2021;10:1- doi: [10.1007/s40065-020-00293-y](https://doi.org/10.1007/s40065-020-00293-y)
6. Ahmadian A, Salahshour S, Ali-Akbari M, Ismail F, Baleanu D. A novel approach to approximate fractional derivative with uncertain conditions. *Chaos, Solitons and Fractals* 2017; 104: 68–76. doi: [10.1016/j.chaos.2017.07.026](https://doi.org/10.1016/j.chaos.2017.07.026)
7. Ali A, Minamoto T, Saeed U, Rehman MU. ψ -Haar wavelets method for numerically solving fractional differential equations. *Engineering Computations* 2020; 38(2): 1037-1056. doi: [10.1108/EC-01-2020-0050](https://doi.org/10.1108/EC-01-2020-0050)
8. Allen RG. REF-ET: Reference Evapotranspiration Calculation Software for FAO and ASCE Standardized Equations, University of Idaho. [Електронний ресурс], 2000, URL: www.kimberly.uidaho.edu/ref-et/ (дата звернення: 10.06.2018).
9. Allen RG, Pereira L, Raes D, Smith M. Crop Evapotranspiration. Rome, Italy: Food and Agriculture Organization of the United Nations, 1998.

10. Allwright A, Atangana A. Fractal advection-dispersion equation for groundwater transport in fractured aquifers with self-similarities. *The European Physical Journal Plus* 2018; 133(2): 1–14. doi: [10.1140/epjp/i2018-11885-3](https://doi.org/10.1140/epjp/i2018-11885-3)
11. Almeida R. A Caputo fractional derivative of a function with respect to another function. *Communications in Nonlinear Science and Numerical Simulation* 2017; 44:460-481. doi: [10.1016/j.cnsns.2016.09.006](https://doi.org/10.1016/j.cnsns.2016.09.006)
12. Almeida R. What is the best fractional derivative to fit data? *Applicable Analysis and Discrete Mathematics* 2017; 11(2): 358-368. doi: [10.2298/AADM170428002A](https://doi.org/10.2298/AADM170428002A)
13. Almeida R, Malinowska AB, Odziejewicz T. Fractional differential equations with dependence on the Caputo-Katugampola derivative. *J. Comput. Nonlinear Dynam* 2016; 11(6): 061017. doi: [10.1115/1.4034432](https://doi.org/10.1115/1.4034432)
14. Atanackovic TM, Stankovic B. On a numerical scheme for solving differential equations of fractional order. *Mech. Res. Comm* 2008; 35(7): 429–438. doi: [10.1016/j.mechrescom.2008.05.003](https://doi.org/10.1016/j.mechrescom.2008.05.003)
15. Atangana A. Derivative with two fractional orders: New door of investigation toward revolution in fractional calculus. *The European Physical Journal Plus* 2016; 131: 373. doi: [10.1140/epjp/i2016-16373-2](https://doi.org/10.1140/epjp/i2016-16373-2)
16. Atangana A, Alkahtani B. New model of groundwater flowing within a confine aquifer: application of Caputo-Fabrizio derivative. *Arabian journal of geosciences* 2015; 9(1): 1-6. doi: [10.1007/s12517-015-2060-8](https://doi.org/10.1007/s12517-015-2060-8)
17. Atangana A, Baleanu D. Caputo-Fabrizio derivative applied to groundwater flow within confined aquifer. *Jornal of Engineering Mechanics* 2017; 143(5): D4016005. doi: [10.1061/\(ASCE\)EM.1943-7889.0001091](https://doi.org/10.1061/(ASCE)EM.1943-7889.0001091)
18. Atangana A, Baleanu D. New fractional derivatives with nonlocal and non-singular kernel: Theory and application to heat transfer model. *Thermal Science* 2016; 20: 763-769. doi: [10.2298/TSCI160111018A](https://doi.org/10.2298/TSCI160111018A)
19. Atangana A, Gomez-Aguilar J. Numerical approximation of Riemann-Liouville definition of fractional derivative: from Riemann-Liouville to Atangana-

Baleanu. Numer. Methods Partial. Differ. Equ. 2018; 34: 1502–1523. doi: [10.1002/num.22195](https://doi.org/10.1002/num.22195)

20. Avazzadeh Z, Heydari MH, Cattani C. Legendre wavelets for fractional partial integro-differential viscoelastic equations with weakly singular kernels. Eur. Phys. J. Plus 2019; 134: 368. doi: [10.1140/epjp/i2019-12743-6](https://doi.org/10.1140/epjp/i2019-12743-6)

21. Baban A, Bonchis C, Fikl A, Rosu F. Parallel simulations for fractional-order systems. In: SYNASC 2016, 2016, p. 141–144. doi: [10.1109/SYNASC.2016.033](https://doi.org/10.1109/SYNASC.2016.033)

22. Baffet D. A Gauss–Jacobi Kernel Compression Scheme for Fractional Differential Equations. J Sci Comput 2019; 79: 227–248. doi: [10.1007/s10915-018-0848-x](https://doi.org/10.1007/s10915-018-0848-x)

23. Baffet D, Hesthaven JS. A Kernel Compression Scheme for Fractional Differential Equations. SIAM J. Numer. Anal. 2017; 55(2): 496–520. doi: [10.1137/15M1043960](https://doi.org/10.1137/15M1043960)

24. Baleanu D, Mohammadi H, Rezapour S. A fractional differential equation model for the COVID-19 transmission by using the Caputo–Fabrizio derivative. Adv. Differ. Equ. 2020; 2020: 299. doi: [10.1186/s13662-020-02762-2](https://doi.org/10.1186/s13662-020-02762-2)

25. Bastiaanssen WGM. Regionalization of surface flux densities and moisture indicators in composite terrain. Doctoral thesis. Wageningen: Wageningen Agricultural University, 1995.

26. Benson DA, Wheatcraft SW, Meerschaert MM. Application of a fractional advection-dispersion equation. Water Resour. Res. 2000; 36:1403–1412. doi: [10.1029/2000WR900031](https://doi.org/10.1029/2000WR900031)

27. Bertaccini D, Durastante F. Efficient Preconditioner Updates for Semilinear Space–Time Fractional Reaction–Diffusion Equations. In: Bini D., Di Benedetto F., Tyrtyshnikov E., Van Barel M. (eds) Structured Matrices in Numerical Linear Algebra. Springer INdAM Series, vol 30. Cham: Springer, 2019. doi: [10.1007/978-3-030-04088-8_15](https://doi.org/10.1007/978-3-030-04088-8_15)

28. Bhrawy AH, Abdelkawy MA. A fully spectral col- location approximation for multi-dimensional fractional Schrodinger equations. J. Comput. Phys. 2015; 294: 462–483. doi: [10.1016/j.jcp.2015.03.063](https://doi.org/10.1016/j.jcp.2015.03.063)

29. Bhrawy AH, Taha TM, Machado JAT. A review of operational matrices and spectral techniques for fractional calculus. *Nonlinear Dyn.* 2015; 81: 1023–1052. doi: [10.1007/s11071-015-2087-0](https://doi.org/10.1007/s11071-015-2087-0)
30. Bhrawy AH, Tharwat MM, Alghamdi MA. A new operational matrix of fractional integration for shifted Jacobi polynomials. *Bull. Malays. Math. Sci. Soc.* 2014; 37(4): 983–995.
31. Biala TA, Khaliq AQM. Parallel algorithms for nonlinear time–space fractional parabolic PDEs. *Journal of Computational Physics* 2018; 375: 135-154. doi: [10.1016/j.jcp.2018.08.034](https://doi.org/10.1016/j.jcp.2018.08.034)
32. Björck Å. Numerical methods for least squares problems. Philadelphia: SIAM, 1996. doi: [10.1137/1.9781611971484](https://doi.org/10.1137/1.9781611971484)
33. Blaney HF, Criddle W. Determining Requirements Water in Irrigated Areas from Climatological and Irrigation Data. Washington Soil Conservation Service, 1950, 48.
34. Bogaenko VA, Bulavatsky VM, Kryvonos IuG. On mathematical modeling of fractional-differential dynamics of flushing process for saline soils with parallel algorithms usage. *Journal of Automation and Information Sciences* 2016; 48(10): 1–12. doi: [10.1615/JAutomatInfScien.v48.i10.10](https://doi.org/10.1615/JAutomatInfScien.v48.i10.10)
35. Bohaienko V, Gladky A. On the selection of fractional-differential model of convective diffusion with mass exchange. *Modeling, Control and Information Technologies: Proceedings of International Scientific and Practical Conference 2020*; 4: 7–10. doi: [10.31713/MCIT.2020.02](https://doi.org/10.31713/MCIT.2020.02)
36. Bohaienko V, Gladky A, Romashchenko M, Matiash T. Identification of fractional water transport model with psi-caputo derivatives using particle swarm optimization algorithm. *Applied Mathematics and Computation* 2021; 390:125665. doi: [10.1016/j.amc.2020.125665](https://doi.org/10.1016/j.amc.2020.125665)
37. Bohaienko V, Gladky A. Modelling fractional-order moisture transport in irrigation using artificial neural networks. *SeMA 2023* doi: [10.1007/s40324-023-00322-8](https://doi.org/10.1007/s40324-023-00322-8)
38. Bohaienko VA, Bulavatskiy VM. Computer Simulation Based on Non-local Model of the Dynamics of Convective Diffusion of Soluble Substances in the

Underground Filtration Flow under Mass Exchange Conditions. Journal of Automation and Information Science 2019; 51(5): 16–29. doi: [10.1615/JAutomatInfScien.v51.i5.20](https://doi.org/10.1615/JAutomatInfScien.v51.i5.20)

39. Bohaienko VO. A fast finite-difference algorithm for solving space-fractional filtration equation with a generalised Caputo derivative. Computational and Applied Mathematics 2019; 38(3): 105. doi: [10.1007/s40314-019-0878-5](https://doi.org/10.1007/s40314-019-0878-5)

40. Bohaienko VO. Accuracy and speed of splitting methods for three-dimensional space-time fractional diffusion equation with ψ -Caputo derivatives. Mathematics and Computers in Simulation 2021; 188: 226-240. doi: [10.1016/j.matcom.2021.04.004](https://doi.org/10.1016/j.matcom.2021.04.004)

41. Bohaienko VO. Computing ψ -Caputo Fractional Derivative Values Using CUDA 10. Proceedings of the 9th International Conference "Information Control Systems & Technologies" Odessa, Ukraine, September 24–26, 2020. CEUR Workshop proceedings, vol. 2711, 2020, p. 49.

42. Bohaienko VO. Efficient computation schemes for generalized two-dimensional time-fractional diffusion equation. "Інформаційні технології та комп'ютерне моделювання"; матеріали статей Міжнародної науково-практичної конференції, м. Івано-Франківськ, 20-25 травня 2019 року. Івано-Франківськ: п. Голіней О.М., 2019, с.238-241.

43. Bohaienko VO. Numerical Integration Schemes for Finite Difference Solution of Time-Fractional Diffusion Equation with Generalized Caputo Derivative. "Інформаційні технології та комп'ютерне моделювання"; матеріали статей Міжнародної науково-практичної конференції, м. Івано-Франківськ, 14-19 травня 2018 року. Івано-Франківськ: п. Голіней О.М., 2018, с.250-253.

44. Bohaienko VO. Numerical schemes for modelling time-fractional dynamics of non-isothermal diffusion in soils. Mathematics and Computers in Simulation 2019; 157: 100–114. doi: [10.1016/j.matcom.2018.09.025](https://doi.org/10.1016/j.matcom.2018.09.025)

45. Bohaienko VO. On the recurrent computation of fractional operator with Mittag-Leffler kernel. Applied Numerical Mathematics 2021; 162: 137-149. doi: [10.1016/j.apnum.2020.12.016](https://doi.org/10.1016/j.apnum.2020.12.016)

46. Bohaienko VO. Parallel algorithms for modelling two-dimensional non-equilibrium salt transfer processes on the base of fractional derivative model. *Fractional Calculus and Applied Analysis* 2018; 21(3): 654-671. doi: [10.1515/fca-2018-0035](https://doi.org/10.1515/fca-2018-0035)
47. Bohaienko VO. Parallel finite-difference algorithms for three-dimensional space-fractional diffusion equation with ψ -Caputo derivatives. *Computational and Applied Mathematics* 2020; 39:163. doi: [10.1007/s40314-020-01191-x](https://doi.org/10.1007/s40314-020-01191-x)
48. Bohaienko VO. Selection of ψ -Caputo derivative functional parameter in generalized water transport equation by genetic programming technique. *Results in Control and Optimization* 2021; 5: 100068. doi: [10.1016/j.rico.2021.100068](https://doi.org/10.1016/j.rico.2021.100068)
49. Bohaienko VO, Bulavatsky VM. On a New Analog of the Biparabolic Evolution Equation with Conformable Fractional Derivatives. *Journal of Automation and Information Sciences* 2020; 52(6): 1-14. doi: [10.1615/JAutomatInfScien.v52.i6.10](https://doi.org/10.1615/JAutomatInfScien.v52.i6.10)
50. Bohaienko VO, Bulavatsky VM. Mathematical Modeling of Solutes Migration Under the Conditions of Groundwater Filtration by the Model with the k -Caputo Fractional Derivative. *Fractal Fract.* 2018; 2(4):28. doi: [10.3390/fractalfract2040028](https://doi.org/10.3390/fractalfract2040028)
51. Bohaienko VO, Bulavatsky VM. Simplified Mathematical Model for the Description of Anomalous Migration of Soluble Substances in Vertical Filtration Flow. *Fractal Fract.* 2020; 4: 20. doi: [10.3390/fractalfract4020020](https://doi.org/10.3390/fractalfract4020020)
52. Bohaienko VO, Bulavatsky VM. Fractional-fractal modeling of filtration-consolidation processes in saline saturated soils. *Fractal and Fractional* 2020; 4(4): 59. doi: [10.3390/fractalfract4040059](https://doi.org/10.3390/fractalfract4040059)
53. Bohaienko VO, Bulavatsky VM, Gladky AV. GPU algorithms for solving time-fractional diffusion equation with generalised Caputo derivative with respect to a function. Fifth International Conference "High Performance Computing" HPC-UA 2018 (Ukraine, Kyiv, October 22-23, 2018), 2018, p.12-17.
54. Bohaienko VO, Bulavatsky VM, Kryvonos IuH. Mathematical Modeling of Fractional-Differential Dynamics of Process of Filtration-Convective Diffusion of Soluble Substances in Nonisothermal Conditions. *Journal of Automation and Information Sciences* 2017; 49(4):12-25. doi: [10.1615/JAutomatInfScien.v49.i4.20](https://doi.org/10.1615/JAutomatInfScien.v49.i4.20)

55. Bohaienko VO. Performance of vectorized GPU-algorithm for computing ψ -Caputo derivative values. Hu Z., Petoukhov S., Dychka I., He M. (eds) *Advances in Computer Science for Engineering and Education III. ICCSEEA 2020. Advances in Intelligent Systems and Computing*, vol 1247. Cham: Springer, 2020, p. 266-275. doi: [10.1007/978-3-030-55506-1_24](https://doi.org/10.1007/978-3-030-55506-1_24)
56. Bonchis C, Kaslik E, Rosu F. HPC optimal parallel communication algorithm for the simulation of fractional-order systems. *J Supercomput* 2019; 75:1014–1025. doi: [10.1007/s11227-018-2267-z](https://doi.org/10.1007/s11227-018-2267-z)
57. Brociek R, Chmielowska A, Słota D. Comparison of the Probabilistic Ant Colony Optimization Algorithm and Some Iteration Method in Application for Solving the Inverse Problem on Model With the Caputo Type Fractional Derivative. *Entropy* 2020; 22: 555. doi: [10.3390/e22050555](https://doi.org/10.3390/e22050555)
58. Budyko MI. *Climate and Life*. New York: Academic Press, 1974.
59. Bulavatsky VM, Bogaenko VA. Mathematical modeling of the fractional differential dynamics of the relaxation process of convective diffusion under conditions of planned filtration. *Cybernetics and Systems Analysis* 2015; 51(6):886–895. doi: [10.1007/s10559-015-9781-2](https://doi.org/10.1007/s10559-015-9781-2)
60. Bulavatsky VM. Fractional differential mathematical models of the dynamics of nonequilibrium geomigration processes and problems with nonlocal boundary conditions. *Cybernetics and Systems Analysis* 2014; 50(1): 81–89. doi: [10.1007/s10559-014-9594-8](https://doi.org/10.1007/s10559-014-9594-8)
61. Bulavatsky VM. Mathematical Model of Geoinformatics for Investigation of Dynamics for Locally Nonequilibrium Geofiltration Processes. *Journal of Automation and Information Sciences* 2011; 43(12): 12–20. doi: [10.1615/JAutomatInfScien.v43.i12.20](https://doi.org/10.1615/JAutomatInfScien.v43.i12.20)
62. Bulavatsky VM. Mathematical modeling of dynamics of the process of filtration convection diffusion under the condition of time nonlocality. *Journal of Automation and Information Science* 2012; 44(2): 13–22. doi: [10.1615/JAutomatInfScien.v44.i4.20](https://doi.org/10.1615/JAutomatInfScien.v44.i4.20)

63. Bulavatsky VM. Numerical modeling of the dynamics of a convection diffusion process locally non-equilibrium in time. *Cybernetics and Systems Analysis* 2012; 48(6):861–869. doi: [10.1007/s10559-012-9465-0](https://doi.org/10.1007/s10559-012-9465-0)
64. Bulavatsky VM, Bogaenko VA. Mathematical modeling of the dynamics of the nonequilibrium in time convective-diffusion process in domain with free boundaries. *Cybernetics and Systems Analysis* 2016; 52(3):427–440. doi: [10.1007/s10559-016-9843-0](https://doi.org/10.1007/s10559-016-9843-0)
65. Bulavatsky VM, Bohaienko VO. Some Boundary-Value Problems of Fractional-Differential Mobile–Immobile Migration Dynamics in a Profile Filtration Flow. *Cybernetics and Systems Analysis* 2020; 56(3): 410-425. doi: [10.1007/s10559-020-00257-2](https://doi.org/10.1007/s10559-020-00257-2)
66. Bulavatsky VM, Bohaienko VO. Some Consolidation Dynamics Problems within the Framework of the Biparabolic Mathematical Model and its Fractional-Differential Analog. *Cybernetics and Systems Analysis* 2020; 56(5): 770-783. doi: [10.1007/s10559-020-00298-7](https://doi.org/10.1007/s10559-020-00298-7)
67. Bulavatsky VM, Bohaienko VO. Numerical Simulation of Fractional-Differential Filtration-Consolidation Dynamics Within the Framework of Models with Non-Singular Kernel. *Cybernetics and Systems Analysis* 2018; 54:193–204. doi: [10.1007/s10559-018-0020-5](https://doi.org/10.1007/s10559-018-0020-5)
68. Bulavatsky VM, Krivonos YuG. Mathematical modelling in the geoinformation problem of the dynamics of geomigration under space- time nonlocality. *Cybernetics and Systems Analysis* 2012; 48(4): 539– 546. doi: [10.1007/s10559-012-9432-9](https://doi.org/10.1007/s10559-012-9432-9)
69. Bulavatsky VM, Kryvonos YuG. Mathematical Models with a Control Function for Investigation of Fractional-differential Dynamics of Geomigration Processes. *Journal of Automation and Information Sciences* 2014; 46(6): 1-11. doi: [10.1615/JAutomatInfScien.v46.i6.10](https://doi.org/10.1615/JAutomatInfScien.v46.i6.10)
70. Bulavatsky VM, Kryvonos YuG. The numerically analytical solutions of some geomigratory problems within the framework of fractional-differential mathematical models. *Journal of Automation and Information Science*. 2014; 46(2): 1 – 11. doi: [10.1615/JAutomatInfScien.v46.i2.10](https://doi.org/10.1615/JAutomatInfScien.v46.i2.10)

71. Cai W, Chen W, Wang F. Three-dimensional Hausdorff derivative diffusion model for isotropic/anisotropic fractal porous media. *Thermal Science* 2018; 22: S1-S6. doi: [10.2298/TSCI170630265C](https://doi.org/10.2298/TSCI170630265C)
72. Campbell GS, Campbell MD. Irrigation scheduling using soil moisture measurements: theory and practice. In: Hillel D, editor. *Advances in irrigation*, vol.1. New York (NY): Academic Press, 1982, p. 25-42. doi: [10.1016/B978-0-12-024301-3.50008-3](https://doi.org/10.1016/B978-0-12-024301-3.50008-3)
73. Campbell GS, Norman JM. *An Introduction To Environmental Biophysics*, Second Edition, Springer, 1998. doi: [10.1007/978-1-4612-1626-1](https://doi.org/10.1007/978-1-4612-1626-1)
74. Caputo M. Linear model of dissipation whose Q is almost frequency independent. II. *Geophysical Journal International* 1967; 13 (5): 529–539. doi: [10.1111/j.1365-246X.1967.tb02303.x](https://doi.org/10.1111/j.1365-246X.1967.tb02303.x)
75. Caputo M, Fabrizio M. A new definition of fractional derivative without singular kernel. *Progress Fractional Differentiation and Applications* 2015; 1(2): 73-85.
76. Carrera J, Sánchez-Vila X, Benet I, Medina Germán Galarza A, Guimerà J. On matrix diffusion: formulations, solution methods and qualitative effects. *Hydrogeology Journal* 1998; 6: 178–190. doi: [10.1007/s100400050143](https://doi.org/10.1007/s100400050143)
77. Cattaneo C. A form of heat conduction equation which eliminates the paradox of instantaneous propagation. *Comp. Rend.* 1958; 247: 431 –433.
78. Cerutti RA. The k-fractional logistic equations with k-Caputo derivative. *Pure mathematical sciences* 2015; 4(1): 9–15. doi: [10.12988/pms.2015.4818](https://doi.org/10.12988/pms.2015.4818)
79. Çetinkaya A, Kıymaz O. The solution of the time-fractional diffusion equation by the generalized differential transform method. *Mathematical and Computer Modelling* 2013; 57(9-10): 2349-2354. doi: [10.1016/j.mcm.2011.12.015](https://doi.org/10.1016/j.mcm.2011.12.015)
80. Chassemi A, Diek A. Linear chemo-poroelasticity for swelling shales: theory and application. *Journal petroleum sciences engineering* 2003; 38: 199–211. doi: [10.1016/S0920-4105\(03\)00033-0](https://doi.org/10.1016/S0920-4105(03)00033-0)
81. Chechkin AV, Gorenflo R, Sokolov IM, Gonchar VY. Distributed order time fractional diffusion equation. *Fractional Calculus and Applied Analysis* 2003; 6(3):259-280.

82. Chen W. Time-space fabric underlying anomalous diffusion. *Chaos, Soliton. Fract.* 2006; 28(4): 923–929. doi: [10.1016/j.chaos.2005.08.199](https://doi.org/10.1016/j.chaos.2005.08.199)
83. Chikrii AA, Eidelman SD. Game problems for fractional quasilinear systems. *Computers & Mathematics with Applications* 2002; 44(7): 835-851. doi: [10.1016/S0898-1221\(02\)00197-9](https://doi.org/10.1016/S0898-1221(02)00197-9)
84. Chikriy AA, Chentsov AG, Matychyn II. Differential games of the fractional order with separated dynamics. *Journal of Automation and Information Sciences* 2009; 41(11):17-27. doi: [10.1615/JAutomatInfScien.v41.i11.20](https://doi.org/10.1615/JAutomatInfScien.v41.i11.20)
85. Compte A, Metzler R. The generalized Cattaneo equation for the description of anomalous transport processes. *J. Phys. A.: Math. Gen.* 1997; 30: 7277– 7289. doi: [10.1088/0305-4470/30/21/006](https://doi.org/10.1088/0305-4470/30/21/006)
86. Corless RM, Gonnet GH, Hare DEG, Jeffrey DJ, Knuth DE. On the Lambert W function. *Advances in Computational Mathematics* 1996; 5: 329–359. doi: [10.1007/BF02124750](https://doi.org/10.1007/BF02124750)
87. Cushman JH. On measurement, scale, and scaling. *Water Resour Res* 1986; 22(2):.129–134. doi: [10.1029/WR022i002p00129](https://doi.org/10.1029/WR022i002p00129)
88. Deans HA, Rice U. A mathematical model for dispersion in the direction of flow in porous media. *Soc. Petrol. Eng. Journal* 1963; 3(1): 49–52. doi: [10.2118/493-PA](https://doi.org/10.2118/493-PA)
89. Dedic L, Matic M, Pecaric J. On Euler midpoint formulae. *ANZIAM J.* 2005; 46: 417–438. doi: [10.1017/S144618110000835X](https://doi.org/10.1017/S144618110000835X)
90. D'Elia M, Du Q, Glusa CA, Gunzburger MD, Tian XCh, Zhou Zh. Numerical methods for nonlocal and fractional models. *Acta Numerica* 2020; 29: 1-124. doi: [10.1017/S096249292000001X](https://doi.org/10.1017/S096249292000001X)
91. Diethelm K. An efficient parallel algorithm for the numerical solution of fractional differential equations. *Fract Calc Appl Anal* 2011; 14(3):475–490. doi: [10.2478/s13540-011-0029-1](https://doi.org/10.2478/s13540-011-0029-1)
92. Diethelm K. An investigation of some nonclassical methods for the numerical approximation of Caputo-type fractional derivatives. *Numer Algor* 2008; 47:361–390. doi: [10.1007/s11075-008-9193-8](https://doi.org/10.1007/s11075-008-9193-8)

93. Diethelm K. The Analysis of Fractional Differential Equations. An application-oriented exposition using differential operators of Caputo type. Lecture Notes in Mathematics, vol. 2004, Berlin: Springer-Verlag, 2010. doi: [10.1007/978-3-642-14574-2](https://doi.org/10.1007/978-3-642-14574-2)
94. Diethelm K, Garrappa R, Stynes M. Good (and Not So Good) Practices in Computational Methods for Fractional Calculus. Mathematics 2020; 8: 324. doi: [10.3390/math8030324](https://doi.org/10.3390/math8030324)
95. Ding HF, Li ChP. A High-Order Algorithm for Time- Caputo-Tempered Partial Differential Equation with Riesz Derivatives in Two Spatial Dimensions. Journal of Scientific Computing 2019; 80: 81–109. doi: [10.1007/s10915-019-00930-5](https://doi.org/10.1007/s10915-019-00930-5)
96. Du W, Tong L, Tang Y. Metaheuristic optimization-based identification of fractional-order systems under stable distribution noises. Physics Letters A 2018; 382(34): 2313-2320. doi: [10.1016/j.physleta.2018.05.043](https://doi.org/10.1016/j.physleta.2018.05.043)
97. Eidelman SD, Kochubei AN. Cauchy problem for fractional diffusion equations. Journal of Differential Equations 2004; 199(2): 211-255. doi: [10.1016/j.jde.2003.12.002](https://doi.org/10.1016/j.jde.2003.12.002)
98. Fang ZhW, Sun HW, Wei HQ. On Approximate Inverse Preconditioning Techniques for Spatial Fractional Diffusion Equations with Piecewise Continuous Coefficients. International Journal of Computer Mathematics 2020; 97(3): 523-545. doi: [10.1080/00207160.2019.1579313](https://doi.org/10.1080/00207160.2019.1579313)
99. FAO. Soybean crop information [internet]. Available from: <http://www.fao.org/land-water/databases-and-software/crop-information/soybean/en/>
100. Feller W. On a generalization of Marcel Riesz' potentials and the semi-groups generated by them. Gleerup, 1962.
101. Foadaddini A, Zolfaghari SA, Darian HM, Saadatfar H. An efficient GPU-based fractional-step domain decomposition scheme for the reaction–diffusion equation. Computational and Applied Mathematics 2020; 39:305. doi: [10.1007/s40314-020-01357-7](https://doi.org/10.1007/s40314-020-01357-7)
102. Ford N, Simpson A. The numerical solution of fractional differential equations: Speed versus accuracy. Numerical Algorithms 2001; 26: 333–346. doi: [10.1023/A:1016601312158](https://doi.org/10.1023/A:1016601312158)

103. Fu HF, Wang H. A preconditioned fast parareal finite difference method for space-time fractional partial differential equation. *Journal of Scientific Computing* 2018; 78(3): 1724–1743. doi: [10.1007/s10915-018-0835-2](https://doi.org/10.1007/s10915-018-0835-2)
104. Gao F, Lee XJ, Tong HQ, Fei FX, Zhao HL. Identification of Unknown Parameters and Orders via Cuckoo Search Oriented Statistically by Differential Evolution for Noncommensurate Fractional-Order Chaotic Systems. *Abstract and Applied Analysis* 2013; 2013: 382834. doi: [10.1155/2013/382834](https://doi.org/10.1155/2013/382834)
105. Geiger S, Dentz M, Neuweiler I. A Novel Multi-Rate Dual-Porosity Model for Improved Simulation of Fractured and Multiporosity Reservoirs. *SPE J.* 2013; 18 (04): 670–684. doi: [10.2118/148130-PA](https://doi.org/10.2118/148130-PA)
106. Gigante V, Iacobellis V, Manfred S, Milell P, Portoghese I. Influences of Leaf Area Index estimations on water balance modeling in a Mediterranean semi-arid basin. *Nat. Hazards Earth Syst. Sci.* 2009; 9:979–991. doi: [10.5194/nhess-9-979-2009](https://doi.org/10.5194/nhess-9-979-2009)
107. Gómez-Aguilar JF, Miranda-Hernández M, López-López MG, Alvarado-Martínez VM, Baleanu D. Modeling and simulation of the fractional space-time diffusion equation. *Communications in Nonlinear Science and Numerical Simulation* 2016; 30:115-127. doi: [10.1016/j.cnsns.2015.06.014](https://doi.org/10.1016/j.cnsns.2015.06.014)
108. Gong CY, Bao WM, Tang GJ. A parallel algorithm for the Riesz fractional reaction-diffusion equation with explicit finite difference method. *Fractional Calculus & Applied Analysis* 2013; 16(3): 654–669. doi: [10.2478/s13540-013-0041-8](https://doi.org/10.2478/s13540-013-0041-8)
109. Tao J, Liu YH, Li Q, Ren JL, Wang DSh An accelerated novel meshless coupled algorithm for Non-local nonlinear behavior in 2D/3D space-fractional GPEs, *Computer Physics Communications* 2024; 296: 109023. doi: [10.1016/j.cpc.2023.109023](https://doi.org/10.1016/j.cpc.2023.109023)
110. Gong ChY, Bao WM, Liu J. A Piecewise memory principle for fractional derivatives. *Fractional Calculus and Applied Analysis* 2017; 20: 1010–1022. doi: [10.1515/fca-2017-0052](https://doi.org/10.1515/fca-2017-0052)
111. Gong ChY, Bao WM, Tang GJ, Jiang YW, Liu J. Computational Challenge of Fractional Differential Equations and the Potential Solutions: A Survey. *Mathematical Problems in Engineering* 2015; 2015: 258265. doi: [10.1155/2015/258265](https://doi.org/10.1155/2015/258265)

112. Gong ChY, Bao WM, Tang GJ, Yang B, Liu J. An efficient parallel solution for Caputo fractional reaction-diffusion equation. *The Journal of Supercomputing* 2014; 68(3): 1521–1537. doi: [10.1007/s11227-014-1123-z](https://doi.org/10.1007/s11227-014-1123-z)
113. Gorenflo R, Kilbas AA, Mainardi F, Rogosin SV. Mittag-Leffler functions, related topics and applications. Berlin: Springer Verlag, 2014. doi: [10.1007/978-3-662-43930-2](https://doi.org/10.1007/978-3-662-43930-2)
114. Gorenflo R, Mainardi F. Approximation of Levy–Feller diffusion by random walk. *J. Anal. Appl. (ZAA)* 1999; 18: 231–246. doi: [10.4171/zaa/879](https://doi.org/10.4171/zaa/879)
115. Gorenflo R, Mainardi F. Fractional calculus: integral and differential equations of fractional order. In: A. Carpinteri, F. Mainardi (eds) *Fractals and Fractional Calculus in Continuum Mechanics*. Wien: Springer Verlag, 1997, p.223–276. doi: [10.1007/978-3-7091-2664-6_5](https://doi.org/10.1007/978-3-7091-2664-6_5)
116. Gorenflo R, Mainardi F. Random walk models for space-fractional diffusion processes. *Fractional Calculus Appl. Anal.* 1998; 1:167–191.
117. Grünwald AK. Derivationen und deren Anwendung, *Z. Angew. Math. Phys.* 1867; 12: 441–480.
118. Guo X, Li YT, Zeng TY. A finite difference scheme for Caputo-Fabrizio fractional differential equations. *International journal of numerical analysis and modeling* 2020; 17(2): 195–211.
119. Hajipour M, Jajarmi A, Baleanu D, Sun HG. On an accurate discretization of a variable-order fractional reaction-diffusion equation. *Commun Nonlinear Sci Numer Simulat* 2019; 69: 119–133. doi: [10.1016/j.cnsns.2018.09.004](https://doi.org/10.1016/j.cnsns.2018.09.004)
120. Hashan M, Jahan LN, Uz-Zaman T, Imtiaz S, Hossain ME. Modelling of fluid flow through porous media using memory approach: A review. *Mathematics and Computers in Simulation* 2020; 177: 643–673. doi: [10.1016/j.matcom.2020.05.026](https://doi.org/10.1016/j.matcom.2020.05.026)
121. Heydari MH, Hosseininia M. A new variable-order fractional derivative with non-singular Mittag–Leffler kernel: application to variable-order fractional version of the 2D Richard equation. *Engineering with Computers* 2022; 38: 1759–1770. doi: [10.1007/s00366-020-01121-9](https://doi.org/10.1007/s00366-020-01121-9)

122. Hilfer R. Applications of Fractional Calculus in Physics. New York: World Scientific Publishing, 2000. doi: [10.1142/9789812817747](https://doi.org/10.1142/9789812817747)
123. Hillel D. Fundamentals of soil physics. Academic press, 2013.
124. Huang Y, Guo F, Li Y, Liu Y. Parameter estimation of fractional-order chaotic systems by using quantum parallel particle swarm optimization algorithm. PLoS ONE 2015; 10(1): e0114910. doi: [10.1371/journal.pone.0114910](https://doi.org/10.1371/journal.pone.0114910)
125. Ivanyuk V, Ponedilok V, Sterten J. Solving inverse problems of dynamics of non linear objects based on the Volterra series. Computational problems of electrical engineering 2016; 6(1): 9–16.
126. i.evapo.pm GRASS plugin [internet]. Available from: <https://grass.osgeo.org/grass72/manuals/i.evapo.pm.html>
127. Jarad F, Abdeljawad Th, Baleanu D. On the generalized fractional derivatives and their Caputo modification. J. Nonlinear Sci. Appl. 2017; 10:2607–2619. doi: [10.22436/jnsa.010.05.27](https://doi.org/10.22436/jnsa.010.05.27)
128. Jian HY, Huang TZ, Zhao XL, Zhao YL. A fast implicit difference scheme for a new class of time distributed-order and space fractional diffusion equations with variable coefficients. Advances in Difference Equations 2018; 2018:205. doi: [10.1186/s13662-018-1655-2](https://doi.org/10.1186/s13662-018-1655-2)
129. Jiang S, Zhang J, Zhang Q, Zhang Z. Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations. Commun. Comput. Phys. 2017; 21: 650–678. doi: [10.4208/cicp.OA-2016-0136](https://doi.org/10.4208/cicp.OA-2016-0136)
130. Jin BT, Zhou Zh. Incomplete iterative solution of subdiffusion. Numerische Mathematik 2020; 145: 693–725. doi: [10.1007/s00211-020-01128-w](https://doi.org/10.1007/s00211-020-01128-w)
131. Jones HG. Irrigation scheduling – comparison of soil, plant and atmosphere monitoring approaches. Acta Hort. 2008; 792: 391-403. doi: [10.17660/ActaHortic.2008.792.46](https://doi.org/10.17660/ActaHortic.2008.792.46)
132. Kaczmarek M, Huekel T. Chemo-mechanical consolidation of clays: analytical solution for a linearized one-dimensional problem. Transport in porous media 1998; 32: 49–74. doi: [10.1023/A:1006530405361](https://doi.org/10.1023/A:1006530405361)

133. Kamble B, Irmak A, Hubbard K. Estimating Crop Coefficients Using Remote Sensing-Based Vegetation Index. *Remote Sens.* 2013; 5: 1588-1602. doi: [10.3390/rs5041588](https://doi.org/10.3390/rs5041588)
134. Katugampola UN. New approach to generalized fractional derivatives. *Bul. Math. Anal. Appl.* 2014; 6: P.1–15.
135. Katugampola UN. New approach to a generalized fractional integral. *Appl. Math. Comput.* 2011; 218: 860–865. doi: [10.1016/j.amc.2011.03.062](https://doi.org/10.1016/j.amc.2011.03.062)
136. Kavvas M, Ercan A, Polsinelli J. Governing equations of transient soil water flow and soil water flux in multi-dimensional fractional anisotropic media and fractional time. *Hydrol Earth Syst Sci* 2017; 21:1547–1557. doi: [10.5194/hess-21-1547-2017](https://doi.org/10.5194/hess-21-1547-2017)
137. Kavvas ML, Tu T, Ercan A, Polsinelli J. Fractional governing equations of transient groundwater flow in confined aquifers with multi-fractional dimensions in fractional time. *Earth Syst. Dynam.* 2017; 8: 921-929. doi: [10.5194/esd-8-921-2017](https://doi.org/10.5194/esd-8-921-2017)
138. Khan MA, Atangana A. Modeling the dynamics of novel coronavirus (2019-nCov) with fractional derivative. *Alexandria Eng. J.* 2020; 59(4): 2379-2389. doi: [10.1016/j.aej.2020.02.033](https://doi.org/10.1016/j.aej.2020.02.033)
139. Kilbas AA, Srivastava HM, Trujillo JJ. *Theory and applications of fractional differential equations.* Amsterdam: Elsevier, 2006.
140. Koza J. *Genetic Programming: On the Programming of Computers by Means of Natural Selection.* Bradford Books, 1992.
141. Krasnoschok M, Pereverzyev S, Siryk SV, Vasylyeva N. Determination of the fractional order in quasilinear subdiffusion equations/ *Fractional Calculus and Applied Analysis* 2020; 23(3): 694-722. doi: [10.1515/fca-2020-0035](https://doi.org/10.1515/fca-2020-0035)
142. Lam PH, So HC, Chan CF. Exponential sum approximation for Mittag-Leffler function and its application to fractional Zener wave equation. *Journal of Computational Physics* 2020; 410: 109389. doi: [10.1016/j.jcp.2020.109389](https://doi.org/10.1016/j.jcp.2020.109389)
143. Levenberg K. A Method for the Solution of Certain Non-Linear Problems in Least Squares. *Quarterly of Applied Mathematics* 1944; 2 (2): 164–168. doi: [10.1090/qam/10666](https://doi.org/10.1090/qam/10666)

144. Li B, Wang J, Wang L, Zhang G. Anomalous heat conduction and anomalous diffusion in nonlinear lattices, single walled nanotubes, and billiard gas channels. *Chaos An Interdiscip J Nonlinear Sci* 2005; 15: 15121. doi: [10.1063/1.1832791](https://doi.org/10.1063/1.1832791)
145. Li C, Zeng F, Liu F. Spectral approximations to the fractional integral and derivative. *Fractional Calculus and Applied Analysis* 2012; 15(3):383–406. doi: [10.2478/s13540-012-0028-x](https://doi.org/10.2478/s13540-012-0028-x)
146. Li Z, Liang Z, Yan Y. High-order numerical methods for solving time fractional partial differential equations. *J. Sci. Comput.* 2017; 71: 785–803. doi: [10.1007/s10915-016-0319-1](https://doi.org/10.1007/s10915-016-0319-1)
147. Liao KF, Wei T. Identifying a fractional order and a space source term in a time-fractional diffusion-wave equation simultaneously. *Inverse Problems* 2019; 35: 115002. doi: [10.1088/1361-6420/ab383f](https://doi.org/10.1088/1361-6420/ab383f)
148. Lin XL, Ng MK, Sun HW. Crank-Nicolson alternative direction implicit method for space-fractional diffusion equations with nonseparable coefficients. *SIAM J. Numer. Anal.* 2019; 57(3): 997–1019. doi: [10.1137/18M1195693](https://doi.org/10.1137/18M1195693)
149. Liouville J. Mémoire: Sur le calcul des différentielles à indices quelconques, *J. Éc. Polytech.* 1832; 13: 71.
150. Liu J, Gong ChY, Bao WM, Tang GJ, Jiang YW. Solving the Caputo Fractional Reaction-Diffusion Equation on GPU. *Discrete Dynamics in Nature and Society* 2014; 2014: 820162. doi: [10.1155/2014/820162](https://doi.org/10.1155/2014/820162)
151. Liu J, Pattey E, Jago G. Assessment of vegetation indices for regional crop green LAI estimation from landsat images over multiple growing seasons. *Remote Sensing of Environment* 2012; 123:347–358. doi: [10.1016/j.rse.2012.04.002](https://doi.org/10.1016/j.rse.2012.04.002)
152. Liu QX, Chen YM, Liu JK. An improved Yuan–Agrawal method with rapid convergence rate for fractional differential equations. *Computational Mechanics* 2019; 63: 713–723. doi: [10.1007/s00466-018-1621-6](https://doi.org/10.1007/s00466-018-1621-6)
153. Long LD, Luc NH, Zhou Y, Nguyen C. Identification of Source Term for the Time-Fractional Diffusion-Wave Equation by Fractional Tikhonov Method. *Mathematics* 2019; 7: 934. doi: [10.3390/math7100934](https://doi.org/10.3390/math7100934)

154. Macias-Diaz JE. An explicit dissipation-preserving method for Riesz space-fractional nonlinear wave equations in multiple dimensions. *Communications in Nonlinear Science and Numerical Simulation* 2018; 59: 67-87. doi: [10.1016/j.cnsns.2017.10.019](https://doi.org/10.1016/j.cnsns.2017.10.019)
155. Meek BD, DeTar WR, Rolph D, Rechel ER, Carter LM. Infiltration Rate as Affected by an Alfalfa and No-till Cotton Cropping System. *Soil Science Society of America Journal* 1990; 54(2): 505-508. doi: [10.2136/sssaj1990.03615995005400020036x](https://doi.org/10.2136/sssaj1990.03615995005400020036x)
156. Meerschaert MM, Benson D, Scheffler HP, Baeumer B. Fractional time derivatives induce subordination by an inverse Levy process. *Phys. Rev. E.* 2002; 65: 041103. doi: [10.1103/PhysRevE.65.041103](https://doi.org/10.1103/PhysRevE.65.041103)
157. Meerschaert MM, Tadjeran C. Finite difference approximations for fractional advection-dispersion flow equations. *J. Comput. Appl. Math.* 2004; 172(1): 65–77. doi: [10.1016/j.cam.2004.01.033](https://doi.org/10.1016/j.cam.2004.01.033)
158. Mittag-Leffler G. Sur la nouvelle fonction $Ea(x)$. *C. R. Math. Acad. Sci. Paris.* 1903; 137: 554.
159. Mockary S, Babolian E, Vahidi AR. A fast numerical method for fractional partial differential equations. *Advances in Difference Equations* 2019; 2019:452. doi: [10.1186/s13662-019-2390-z](https://doi.org/10.1186/s13662-019-2390-z)
160. Molz FJ, Remson I. Extraction term models of soil moisture use by transpiring plants. *Water Resour. Res.* 1970; 6:1346–1356. doi: [10.1029/WR006i005p01346](https://doi.org/10.1029/WR006i005p01346)
161. Monteith JL, Unsworth MH. *Principles of Environmental Physics*, Second Edition, Butterworth Heinemann, 1990.
162. Neuman SP. Universal scaling of hydraulic conductivities and dispersivities in geologic media. *Water Resour Res* 1990; 26(8): 1749–1758. doi: [10.1029/WR026i008p01749](https://doi.org/10.1029/WR026i008p01749)
163. Ng M. (2004) *Iterative Methods for Toeplitz Systems*. Numer. Math. Sci. Comput., New York, Oxford University Press, 2004.
164. Obembe AD, Al-Yousef HY, Hossain ME, Abu-Khamsin SA. Fractional derivatives and their applications in reservoir engineering problems: A review. *Journal of*

- Petroleum Science and Engineering 2017; 157: 312–327.
doi: [10.1016/j.petrol.2017.07.035](https://doi.org/10.1016/j.petrol.2017.07.035)
165. Ochoa-Tapia JA, Valdes-Parada FJ, Alvarez-Ramirez J. A fractional-order Darcy's law. *Physica A* 2007; 374: 1–14. doi: [10.1016/j.physa.2006.07.033](https://doi.org/10.1016/j.physa.2006.07.033)
166. Odibat ZM, Shawagfeh NT. Generalized Taylor's formula. *Appl Math Comput* 2007; 186: 286–293. doi: [10.1016/j.amc.2006.07.102](https://doi.org/10.1016/j.amc.2006.07.102)
167. Oldham K, Spanier J. *The Fractional Calculus: Theory and Applications of Differentiation and Integration to Arbitrary Order*. New York: Academic Press, 1974.
168. Oldham KB, Spanier J. Theory and applications of differentiation and integration to arbitrary order. In *The Fractional Calculus*. New York, NY, USA; London, UK: Academic Press, 1974.
169. Ortigueira MD, Lopes AM, Tenreiro Machado J. On the Numerical Computation of the Mittag-Leffler Function. *International Journal of Nonlinear Sciences and Numerical Simulation* 2019; 20: 725–736. doi: [10.1515/ijnsns-2018-0358](https://doi.org/10.1515/ijnsns-2018-0358)
170. Osler TJ. Fractional derivatives of a composite function. *SIAM J. Math. Anal.* 1970; 1:288–293. doi: [10.1137/0501026](https://doi.org/10.1137/0501026)
171. Osler TJ. Taylor's series generalized for fractional derivatives and applications. *SIAM J. Math. Anal.* 1971; 2: 37–47. doi: [10.1137/0502004](https://doi.org/10.1137/0502004)
172. Ozcan O. *Fractional Diffusion in Naturally Fractured Unconventional Reservoirs*. Colorado School of Mines. Arthur Lakes Library, 2014.
173. Pachepsky Y, Benson D, Rawls W. Simulating Scale-Dependent Solute Transport in Soils with the Fractional Advective-Dispersive Equation. *Soil Sci. Soc. Am. J.* 2000; 64:1234-1243. doi: [10.2136/sssaj2000.6441234x](https://doi.org/10.2136/sssaj2000.6441234x)
174. Pachepsky Y, Timlin D. Water transport in soils as in fractal media. *Journal of Hydrology* 1998; 204(1-4): 98-107. doi: [10.1016/S0022-1694\(97\)00110-8](https://doi.org/10.1016/S0022-1694(97)00110-8)
175. Pachepsky Y, Timlin D, Rawls W. Generalized Richards' equation to simulate water transport in unsaturated soils. *Journal of Hydrology* 2003; 272: 3–13. doi: [10.1016/S0022-1694\(02\)00251-2](https://doi.org/10.1016/S0022-1694(02)00251-2)

176. Paradisi P, Cesari R, Mainardi F, Tampieri F. The fractional Fick's law for non-local transport processes. *Physica A* 2001; 293: 130 –142. doi: [10.1016/S0378-4371\(00\)00491-X](https://doi.org/10.1016/S0378-4371(00)00491-X)
177. Podlubny I. *Fractional differential equations*. New York: Academic Press, 1999.
178. Podlubny I. Numerical Solution of Ordinary Fractional Differential Equations by the Fractional difference Method. In: *Proceedings of the Second International Conference in Difference Equations*, Gordon and Breach Scientific Publishers, 1997, p. 507-515.
179. Povstenko Y. *Linear fractional diffusion-wave equation for scientists and engineers*. Switzerland: Springer Int. Publ., 2015. doi: [10.1007/978-3-319-17954-4](https://doi.org/10.1007/978-3-319-17954-4)
180. Press WH, Teukolsky SA, Vetterling WT, Flannery BP. Section 4.7. Adaptive Quadrature, *Numerical Recipes: The Art of Scientific Computing* (3rd ed.), Cambridge University Press, 2007.
181. Raghavan R. Fractional diffusion: performance of fractured wells. *J. Pet. Sci. Eng.* 2012; 92: 167–173. doi: [10.1016/j.petrol.2012.06.003](https://doi.org/10.1016/j.petrol.2012.06.003)
182. Ravi Kanth ASV, Garg N. A numerical approach for a class of time-fractional reaction–diffusion equation through exponential B-spline method. *Computational and Applied Mathematics* 2020; 39:37. doi: [10.1007/s40314-019-1009-z](https://doi.org/10.1007/s40314-019-1009-z)
183. Rehman HU, Darus M, Salah J. A Note on Caputo's Derivative Operator Interpretation in Economy. *Journal of Applied Mathematics* 2018; 1260240. doi: [10.1155/2018/1260240](https://doi.org/10.1155/2018/1260240)
184. Roberts JA, Savostyanov DV, Tyrtysnikov EE. Superfast solution of linear convolutional Volterra equations using QTT approximation. *Journal of Computational and Applied Mathematics* 2014; 260:434-448. doi: [10.1016/j.cam.2013.10.025](https://doi.org/10.1016/j.cam.2013.10.025)
185. Romashchenko M, Shatkovskiy A, Zhuravlov O, Vasiuta V, Cherevychnyi Y. Adapting the Penman-Montheith method to transplanted tomato crop in production conditions when applying drip irrigation. *Land Reclamation and Water Management* 2018; 108(2): 5-18. doi: [10.31073/mivg20180108-146](https://doi.org/10.31073/mivg20180108-146)

186. Romashchenko MI, Bohaienko VO, Matiash TV, Kovalchuk VP, Danylenko IuIu. Influence of evapotranspiration assessment on the accuracy of moisture transport modeling under the conditions of sprinkling irrigation in the south of Ukraine. Archives of Agronomy and Soil Science 2020; 66(10): 1424-1435. doi: [10.1080/03650340.2019.1674445](https://doi.org/10.1080/03650340.2019.1674445)
187. Romashchenko MI, Bohaienko VO, Matiash TV, Kovalchuk VP, Krucheniuk AV. Numerical simulation of irrigation scheduling using fractional Richards equation. Irrigation science, 2021; 39:385–396. doi: [10.1007/s00271-021-00725-3](https://doi.org/10.1007/s00271-021-00725-3)
188. Sabatier J, Farges C, Tartaglione V. Some alternative solutions to fractional models for modelling long memory behaviors. Mathematics 2020; 8: 196. doi: [10.3390/math8020196](https://doi.org/10.3390/math8020196)
189. Sabatier J, Merveillaut M, Malti R, Oustaloup A. How to Impose Physically Coherent Initial Conditions to a Fractional System? Communications in Nonlinear Science and Numerical Simulation 2010; 15(5): 1318-1326. doi: [10.1016/j.cnsns.2009.05.070](https://doi.org/10.1016/j.cnsns.2009.05.070)
190. Sabatier J. Fractional-Order Derivatives Defined by Continuous Kernels: Are They Really Too Restrictive? Fractal Fract. 2020; 4: 40. doi: [10.3390/fractalfract4030040](https://doi.org/10.3390/fractalfract4030040)
191. Salehi Y, Schiesser W. Numerical Integration of Space Fractional Partial Differential Equations: Vol 2 - Applications from Classical Integer PDEs. Morgan & Claypool Publishers, 2017. doi: [10.1007/978-3-031-02412-2](https://doi.org/10.1007/978-3-031-02412-2)
192. Samarskii AA, Vabishchevich PN. Computational Heat Transfer. Vol. 2. New York: Wiley, 1995.
193. Samet B, Zhou Y. On ψ -Caputo time fractional diffusion equations: extremum principles, uniqueness and continuity with respect to the initial data. RACSAM 2019; 113: 2877–2887. doi: [10.1007/s13398-019-00666-9](https://doi.org/10.1007/s13398-019-00666-9)
194. Samko SG., Kilbas AA., Marichev OI. Fractional Integrals and Derivatives: Theory and Applications. New York: Gordon and Breach, 1993.
195. Sandev T, Tomovsky Z. Fractional equations and models. Theory and applications. Cham, Switzerland: Springer Nature Switzerland AG, 2019.
196. Saxena RK, Mathai AM, Haubold HJ. Fractional reaction-diffusion equations, Astrophysics and Space Science 2006; 305: 289–296. . doi: [10.1007/s10509-006-9189-6](https://doi.org/10.1007/s10509-006-9189-6)

197. SEBAL GRASS plugin [internet]. Available from: https://github.com/wwolff7/SEBAL_GRASS/blob/master/Sebal70.py
198. Sergienko IV, Petryk MR, Fraissard J, Leclerc S. Highly efficient methods of the identification of competitive diffusion parameters in inhomogeneous media of nanoporous particles. *Cybernetics and Systems Analysis* 2015; 51(4): 529-546. doi: [10.1007/s10559-015-9744-7](https://doi.org/10.1007/s10559-015-9744-7)
199. Seybold H, Hilfer R. Numerical Algorithm for Calculating the Generalized Mittag-Leffler Function. *SIAM Journal on Numerical Analysis* 2008; 47(1): 69-88. doi: [10.1137/070700280](https://doi.org/10.1137/070700280)
200. Shen S, Liu F. Error analysis of an explicit finite difference approximation for the space fractional diffusion equation with insulated ends. *ANZIAM J.* 2005; 46(E):C871–C887. doi: [10.21914/anziamj.v46i0.995](https://doi.org/10.21914/anziamj.v46i0.995)
201. Shumer R, Benson DA, Meershaert MM, Baeumer B. Fractal mobile/immobile solute transport. *Water Resour. Res.* 2003; 39(10): 1296–1309. doi: [10.1029/2003WR002141](https://doi.org/10.1029/2003WR002141)
202. Sibatov RT, Sun HG. Dispersive Transport Described by the Generalized Fick Law with Different Fractional Operators. *Fractal Fract.* 2020; 4: 42. doi: [10.3390/fractalfract4030042](https://doi.org/10.3390/fractalfract4030042)
203. Singh J, Kumar D, Baleanu D. A new analysis of fractional fish farm model associated with Mittag-Leffler-type kernel. *International Journal of Biomathematics* 2020; 13(2): 2050010. doi: [10.1142/S1793524520500102](https://doi.org/10.1142/S1793524520500102)
204. Sleijpen G, Fokkema D. Bicgstab(l) for linear equations involving unsymmetric matrices with complex spectrum. *Electronic Transactions on Numerical Analysis* 1993; 1:11–32.
205. Socha K, Dorigo M. Ant colony optimization for continuous domains. *Eur. J. Oper. Res.* 2008; 185: 1155–1173. doi: [10.1016/j.ejor.2006.06.046](https://doi.org/10.1016/j.ejor.2006.06.046)
206. Stynes M. Fractional-order derivatives defined by continuous kernels are too restrictive. *Applied Mathematics Letters* 2018; 85: 22–26. doi: [10.1016/j.aml.2018.05.013](https://doi.org/10.1016/j.aml.2018.05.013)

207. Su N. Distributed-order infiltration, absorption and water exchange in mobile and immobile zones of swelling soils. *Journal of Hydrology* 2012; 468–469: 1–10. doi: [10.1016/j.jhydrol.2012.07.033](https://doi.org/10.1016/j.jhydrol.2012.07.033)
208. Sun H, Li ZhP, Zhang Y, Chen W. Fractional and fractal derivative models for transient anomalous diffusion: Model comparison. *Chaos, Solitons and Fractals* 2017; 102: 364–353. doi: [10.1016/j.chaos.2017.03.060](https://doi.org/10.1016/j.chaos.2017.03.060)
209. Sun HG, Chang A, Zhang Y, Chen W. A review on variable-order fractional differential equations: mathematical foundations, physical models, numerical methods and applications. *Fractional Calculus and Applied Analysis* 2019; 22(1):27–59. doi: [10.1515/fca-2019-0003](https://doi.org/10.1515/fca-2019-0003)
210. Sun HG, Meerschaert MM, Zhang Y, Zhu JT, Chen W. A fractal Richards' equation to capture the non-Boltzmann scaling of water transport in unsaturated media. *Advances in Water Resources* 2013; 52: 292–295. doi: [10.1016/j.advwatres.2012.11.005](https://doi.org/10.1016/j.advwatres.2012.11.005)
211. Sun HM, Yu JZh, Zhang XL, Wang BG, Jia RSh. The adaptive particle swarm optimization technique for solving microseismic source location parameters. *Nonlin. Processes Geophys.* 2019; 26:163–173. doi: [10.5194/npg-26-163-2019](https://doi.org/10.5194/npg-26-163-2019)
212. Sun ZhZh, Wu XN. A fully discrete difference scheme for a diffusion-wave system. *Applied Numerical Mathematics* 2006; 56: 193–209. doi: [10.1016/j.apnum.2005.03.003](https://doi.org/10.1016/j.apnum.2005.03.003)
213. Sweilam NH, Al-Mekhlafi SM. A novel numerical method for solving the 2-D time fractional cable equation. *Eur. Phys. J. Plus* 2019; 134: 323. doi: [10.1140/epjp/i2019-12730-y](https://doi.org/10.1140/epjp/i2019-12730-y)
214. Tan KH. Principles of soil chemistry. CRC press, 2010. doi: [10.1201/9781439894606](https://doi.org/10.1201/9781439894606)
215. Tarasov VE, Tarasova SS. Fractional Derivatives and Integrals: What Are They Needed For? *Mathematics* 2020; 8: 164. doi: [10.3390/math8020164](https://doi.org/10.3390/math8020164)
216. Tu T, Ercan A, Kavvas ML. Time–space fractional governing equations of transient groundwater flow in confined aquifers: Numerical investigation. *Hydrological Processes* 2018;32:1406–1419. doi: [10.1002/hyp.11500](https://doi.org/10.1002/hyp.11500)

217. Tuan NH, Mohammadi H, Rezapour Sh. A mathematical model for COVID-19 transmission by using the Caputo fractional derivative. *Chaos, Solitons & Fractals* 2020; 140: 110107. doi: [10.1016/j.chaos.2020.110107](https://doi.org/10.1016/j.chaos.2020.110107)
218. van Dam JC, Feddes RA. Numerical simulation of infiltration, evaporation and shallow groundwater levels with the Richards equation. *Journal of Hydrology* 2000; 233:72–85. doi: [10.1016/S0022-1694\(00\)00227-4](https://doi.org/10.1016/S0022-1694(00)00227-4)
219. van der Vorst H. Bi-cgstab: a fast and smoothly converging variant of bi-cg for the solution of nonsymmetric linear systems. *SIAM J Sci Stat Comput* 1992; 13:631-644. doi: [10.1137/0913035](https://doi.org/10.1137/0913035)
220. van Genuchten MT. A closed-form equation for predicting the hydraulic conductivity of unsaturated soils. *Soil Sci. Soc. Am. J.* 1980; 44:884–900. doi: [10.2136/sssaj1980.03615995004400050002x](https://doi.org/10.2136/sssaj1980.03615995004400050002x)
221. van Genuchten MT, Wierenga PJ. Mass transfer studies in sorbing porous media, 1: Analytical solutions. *Soil Science Society of America Journal* 1976;. 40: 473-480. doi: [10.2136/sssaj1976.03615995004000040011x](https://doi.org/10.2136/sssaj1976.03615995004000040011x)
222. Wang H, Basu T. A fast finite difference method for two-dimensional space-fractional diffusion equations. *SIAM J Sci Comput* 2012; 34:2444–2458. doi: [10.1137/12086491X](https://doi.org/10.1137/12086491X)
223. Wang H, Wang K, Sircar T. A direct $O(N \log^2 N)$ finite difference method for fractional diffusion equations. *J. Comput. Phys.* 2010; 229(21): 8095–8104. doi: [10.1016/j.jcp.2010.07.011](https://doi.org/10.1016/j.jcp.2010.07.011)
224. Wang H, Wang KX. An $O(N \log^2 N)$ alternating-direction finite difference method for two-dimensional fractional diffusion equations. *Journal of Computational Physics* 2011; 230: 7830–7839. doi: [10.1016/j.jcp.2011.07.003](https://doi.org/10.1016/j.jcp.2011.07.003)
225. Wang QL, Liu J, Gong ChY, Tang XT, Fu GT, Xing ZCh. An efficient parallel algorithm for Caputo fractional reaction-diffusion equation with implicit finite-difference method. *Advances in Difference Equations* 2016; 2016:207. doi: [10.1186/s13662-016-0929-9](https://doi.org/10.1186/s13662-016-0929-9)

226. Wang Y, Cai Z, Zhang Q. Differential evolution with composite trial vector generation strategies and control parameters. *IEEE Trans. Evol. Comput.* 2011; 15(1): 55–66. doi: [10.1109/TEVC.2010.2087271](https://doi.org/10.1109/TEVC.2010.2087271)
227. Wheatcraft S, Meerschaert MM. Fractional conservation of mass. *Advances in Water Resources* 2008; 31: 1377–1381. doi: [10.1016/j.advwatres.2008.07.004](https://doi.org/10.1016/j.advwatres.2008.07.004)
228. Wu GC, Baleanu D, Zeng SD, Deng ZG. Discrete fractional diffusion equation. *Nonlinear Dyn.* 2015; 80: 281–286. doi: [10.1007/s11071-014-1867-2](https://doi.org/10.1007/s11071-014-1867-2)
229. Wu LF, Yang XZh. An Efficient Alternating Segment Parallel Difference Method for the Time Fractional Telegraph Equation. *Advances in Mathematical Physics* 2020; 6897815. doi: [10.1155/2020/6897815](https://doi.org/10.1155/2020/6897815)
230. Xu WY, Sun H. A fast second-order difference scheme for the space–time fractional equation. *Numer Methods Partial Differential Eq.* 2019; 1–17.
231. Yadav S, Pandey RK, Shukla AK. Numerical approximations of Atangana-Baleanu Caputo derivative and its application. *Chaos, Solitons and Fractals* 2019; 118: 58-64. doi: [10.1016/j.chaos.2018.11.009](https://doi.org/10.1016/j.chaos.2018.11.009)
232. Yadav S, Pandey RK, Shukla AK, Kumar K. High-order approximation for generalized fractional derivative and its application *International Journal of Numerical Methods for Heat & Fluid Flow* 2019; 29(9): 3515-3534. doi: [10.1108/HFF-11-2018-0700](https://doi.org/10.1108/HFF-11-2018-0700)
233. Yan Y, Pal K, Ford NJ. Higher order numerical methods for solving fractional differential equations. *BIT Numer. Math.* 2014; 54: 555–584. doi: [10.1007/s10543-013-0443-3](https://doi.org/10.1007/s10543-013-0443-3)
234. Yang XJ. *General fractional derivatives : theory, methods, and applications.* CRC Press, Taylor & Francis Group, 2019. doi: [10.1201/9780429284083](https://doi.org/10.1201/9780429284083)
235. Yang XJ, Tenreiro Machado JA. A new fractional operator of variable order: Application in the description of anomalous diffusion. *Physica A: Statistical Mechanics and its Applications* 2017; 481:276–283. doi: [10.1016/j.physa.2017.04.054](https://doi.org/10.1016/j.physa.2017.04.054)
236. Yang XS, Deb S. Cuckoo search via Lévy flights. In: *World Congress on Nature & Biologically Inspired Computing (NaBIC 2009).* IEEE Publications. 2009, p. 210–214. doi: [10.1109/NABIC.2009.5393690](https://doi.org/10.1109/NABIC.2009.5393690)

237. Yang XZh, Wu LF. An Efficient Parallel Approximate Algorithm for Solving Time Fractional Reaction-Diffusion Equations. *Mathematical Problems in Engineering* 2020; 4524387. doi: [10.1155/2020/4524387](https://doi.org/10.1155/2020/4524387)
238. Yang Y. Jacobi spectral Galerkin methods for fractional integro-differential equations. *Calcolo* 2015; 52: 519–542. doi: [10.1007/s10092-014-0128-6](https://doi.org/10.1007/s10092-014-0128-6)
239. Yin XC, Fang ShM, Guo ChG. Alternating-direction implicit finite difference methods for a new two-dimensional two-sided space-fractional diffusion equation. *Advances in Difference Equations* 2018; 2018:389. doi: [10.1186/s13662-018-1836-z](https://doi.org/10.1186/s13662-018-1836-z)
240. Yu YY, Deng WH, Wu YJ. Fourth order quasi-compact difference schemes for (tempered) space fractional diffusion equations. *Commun. Math. Sci.* 2017; 15: 1183-1209. doi: [10.4310/CMS.2017.v15.n5.a1](https://doi.org/10.4310/CMS.2017.v15.n5.a1)
241. Yuan L, Agrawal OP. A numerical scheme for dynamic systems containing fractional derivatives. *J Vib Acoust* 2002; 124:2321–2324. doi: [10.1115/1.1448322](https://doi.org/10.1115/1.1448322)
242. Yuan LG, Yang QG. Parameter identification and synchronization of fractional-order chaotic systems. *Commun. Nonlinear Sci. Numer. Simul.* 2012; 17(1): 305–316. doi: [10.1016/j.cnsns.2011.04.005](https://doi.org/10.1016/j.cnsns.2011.04.005)
243. Yuste SB, Quintana-Murillo J. Corrigendum to “A finite difference method with non-uniform timesteps for fractional diffusion equations” [*Comput. Phys. Comm.* 183 (12) (2012) 2594–2600]. *Computer Physics Communications* 2014; 185(3): 1192. doi: [10.1016/j.cpc.2013.11.002](https://doi.org/10.1016/j.cpc.2013.11.002)
244. Yuste SB, Quintana-Murillo J. Fast, accurate and robust adaptive finite difference methods for fractional diffusion equations. *J. Numer Algor* 2016; 71(1): 207-228. doi: [10.1007/s11075-015-9998-1](https://doi.org/10.1007/s11075-015-9998-1)
245. Zaslavsky G. Chaos, fractional kinetics, and anomalous transport. *Phys. Rep.* 2002; 371: 461–580. doi: [10.1016/S0370-1573\(02\)00331-9](https://doi.org/10.1016/S0370-1573(02)00331-9)
246. Zayernouri M, Ainsworth M, Karniadakis GE. A unified Petrov-Galerkin spectral method for fractional PDEs. *Comput. Methods Appl. Mech. Eng.* 2015; 283: 1545-1569. doi: [10.1016/j.cma.2014.10.051](https://doi.org/10.1016/j.cma.2014.10.051)

247. Zhai ShY, Weng ZhF, Feng XL, Yuan JY. Investigations on several high-order ADI methods for time-space fractional diffusion equation. *Numerical Algorithms* 2019; 82: 69-106. doi: [10.1007/s11075-018-0594-z](https://doi.org/10.1007/s11075-018-0594-z)
248. Zhang W, Cai X, Holm S. Time-fractional heat equations and negative absolute temperatures. *Comput. Math. Appl.* 2014; 67: 164–171. doi: [10.1016/j.camwa.2013.11.007](https://doi.org/10.1016/j.camwa.2013.11.007)
249. Zhang Y. A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications. *Mathematical Problems in Engineering* 2015; 2015: 931256. doi: [10.1155/2015/931256](https://doi.org/10.1155/2015/931256)
250. Zhang Y, Cohen J, Owens JD. Fast Tridiagonal Solvers on the GPU. In: *Proceedings of the 15th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP 2010)*, January 9–14, 2010, Bangalore, India, 2010, p. 127-136. doi: [10.1145/1693453.1693472](https://doi.org/10.1145/1693453.1693472)
251. Zhang YX, Ding HF. Numerical Algorithm for the Time-Caputo and Space-Riesz Fractional Diffusion Equation. *Communications on Applied Mathematics and Computation* 2020; 2: 57–72. doi: [10.1007/s42967-019-00032-x](https://doi.org/10.1007/s42967-019-00032-x)
252. Zhang Z. An undetermined time-dependent coefficient in a fractional diffusion equation. *Inverse Problems & Imaging* 2017; 11 (5) : 875-900. doi: [10.3934/ipi.2017041](https://doi.org/10.3934/ipi.2017041)
253. Zhao X, Hu X, Cai W, Karniadakis GE. Adaptive finite element method for fractional differential equations using hierarchical matrices. *Comput. Methods Appl. Mech. Engrg.* 2017; 325: 56-76. doi: [10.1016/j.cma.2017.06.017](https://doi.org/10.1016/j.cma.2017.06.017)
254. Zhao YL, Zhu PY, Gu XM, Zhao XL, Cao JX. A limited-memory block bi-diagonal Toeplitz preconditioner for block lower triangular Toeplitz system from time-space fractional diffusion equation. *Journal of Computational and Applied Mathematics* 2019; 362: 99-115. doi: [10.1016/j.cam.2019.05.019](https://doi.org/10.1016/j.cam.2019.05.019)
255. Zheng ZhB, Zhao W, Dai HZh. A new definition of fractional derivative. *International Journal of Non-Linear Mechanics* 2019; 108: 1-6. doi: [10.1016/j.ijnonlinmec.2018.10.001](https://doi.org/10.1016/j.ijnonlinmec.2018.10.001)

256. Zhou Y, Rajapakse RK, Graham J. A coupled thermo-poroelastic model with thermo-osmosis and thermal-filtration. International journal of solids and structures 1998; 35: 4659 – 4683. doi: [10.1016/S0020-7683\(98\)00089-4](https://doi.org/10.1016/S0020-7683(98)00089-4)

257. Zhou Z, Wu H. Finite element multigrid method for the boundary value problem of fractional advection dispersion equation, J. Appl. Math. 2013; 2013: 385463. doi: [10.1155/2013/385463](https://doi.org/10.1155/2013/385463)

258. Zhukovskiy VJ, Chikriy AA. On discrete conflict-controlled processes described by Grunvald-Letnikov fractional systems. Journal of Automation and Information Sciences 2015; 47(1): 24-34. doi: [10.1615/JAutomatInfScien.v47.i1.30](https://doi.org/10.1615/JAutomatInfScien.v47.i1.30)

259. Абрамовиц М, Стиган И. Справочник по специальным функциям с формулами, графиками и таблицами. Москва: Наука. Главная редакция физико-математической литературы, 1979.

260. Аверьянов СФ. Фильтрация из каналов и ее влияние на режим грунтовых вод. М.: Колос, 1982.

261. Алпатьев АМ. О методах расчёта потребностей в воде культурных фитоценозов в связи с развитием орошения в СССР. Биологические основы орошаемого земледелия. М.: Наука, 1974, с.85-89.

262. Арсенин ВЯ. Математическая физика. Основные уравнения и специальные функции. М.: Наука, 1966.

263. Арутюнян НХ. Некоторые вопросы теории ползучести. Москва-Ленинград: Гостехиздат, 1952.

264. Афонин АА, Сухинов АИ. Математические модели геофильтрации и геомиграции в пористых средах, обладающих фрактальной структурой. Известия ЮФУ. Технические науки, Тематический выпуск 2009,с.62-70.

265. Богаенко ВО. Програмні коди [Internet]. Available from: https://github.com/sevaboh/theses_code

266. Богаенко ВА, Булавацкий ВМ, Скопецкий ВВ. Математическое моделирование динамики геохимических процессов загрязнения водоносных горизонтов. Управляющие системы и машины 2009; 4: 60–66.

267. Богаенко ВА, Булавацкий ВМ, Скопецкий ВВ. Параллельный алгоритм расчета фильтрационно-конвективной диффузии загрязнений из водоносных горизонтов. *Управляющие системы и машины* 2008; 5: 18–23.

268. Богаєнко ВО, Булавацький ВМ, Гладкий АВ. Ідентифікація параметрів однієї дробово-диференціальної моделі міграції розчинних речовин у Математичне та комп'ютерне моделювання. Серія: Технічні науки: зб. наук. праць 2019; 19:5-10. doi: [10.32626/2308-5916.2019-19.5-10](https://doi.org/10.32626/2308-5916.2019-19.5-10)

269. Богаєнко ВО, Булавацький ВМ, Хіміч ОМ. Математичне та комп'ютерне моделювання в задачах гідрогеоміграційної динаміки. Київ: Наукова Думка, 2022.

270. Бомба АЯ, Булавацький ВМ, Скопецький ВВ. Нелінійні математичні моделі процесів геогідродинаміки. Київ: Наукова думка, 2007.

271. Бомба АЯ, Гаврилюк ВІ, Сафоник АП, Фурсачик ОА. Нелінійні задачі типу фільтрація-конвекція-дифузія-масообмін за умов неповних даних: монографія. Рівне: НУВГП, 2011.

272. Бомба АЯ, Присяжнюк ІМ, Присяжнюк ОВ. Асимптотичний метод розв'язання одного класу модельних сингулярно збурених задач процесу масопереносу в різнопористих середовищах. *Доповіді Національної академії наук України* 2013; 3: 28-34.

273. Бомба АЯ, Присяжнюк ІМ, Присяжнюк ОВ. Методи теорії збурень прогнозування процесів тепломасоперенесення в пористих та мікропористих середовищах. Рівне: О. Зень, 2017.

274. Бомба АЯ, Сафоник АП. Математическое моделирование процесса аэробной очистки сточных вод в условиях диффузионного и массообменного возмущений. *Инженерно-физический журнал* 2018; 91(2): 338-344.

275. Бондаренко АН, Бугуева ТВ, Дедок ВА. Нейросетевой подход к решению обратных задач теории аномальной диффузии, *Сиб. журн. индустр. матем.* 2016; 19(3): 3–14.

276. Булавацький В.М., Богаєнко В.О. Численне моделювання дробно-диференціальної динаміки процесу фільтрационно-конвективної диффузії на

основе параллельных алгоритмов для кластерных систем. Доповіді НАНУ 2017; 1: 21-28.

277. Булавацький ВМ, Богаєнко ВО. Компьютерное моделирование дробно-дифференциальной динамики некоторых фильтрационно-консолидационных процессов. Доповіді НАНУ 2018; 4:16-24.

278. Булавацький ВМ, Богаєнко ВО. Компьютерное моделирование динамики процесса миграции растворимых веществ при фильтрации грунтовых вод со свободной поверхностью на основе дробно-дифференциального подхода. Доповіді НАНУ 2018; 12:21-29.

279. Булавацький ВМ, Кривонос ЮГ, Скопецький ВВ. Некласичні математичні моделі процесів тепло– та масопереносу. К.: Наук. думка, 2005.

280. Веригин НН, Васильев СВ, Куранов НР. Методы прогноза солевого режима грунтов и грунтовых вод. Москва: Колос, 1979.

281. Власюк АП, Мартинюк ПМ. Математичне моделювання консолідації ґрунтів в процесі фільтрації сольових розчинів. Рівне: Вид–во УДУВГП, 2004.

282. Власюк АП, Мартинюк ПМ. Математичне моделювання фільтраційної консолідації з урахуванням просідань та повзучості скелета ґрунту. Вісник Київського національного університету імені Тараса Шевченка. Серія: Фізико-математичні науки 2013; 2: 113-117.

283. Власюк АП, Мартинюк ПМ. Чисельне розв’язування задач консолідації та фільтраційного руйнування ґрунтів в умовах тепло-масопереносу методом радіальних базисних функцій. Рівне: НУВГП, 2010.

284. Власюк АП, Мартинюк ПМ. Фильтрационная консолидация трёхфазных грунтов с учётом ползучести скелета и влияния солепереноса в неизотермическом режиме. Математическое моделирование 2010; 22(4): 32-56.

285. Власюк АП, Остапчук ОП. Математичне моделювання переносу сольових розчинів при фільтрації підземних вод у ґрунтових масивах. Рівне: НУВГП, 2015.

286. Герасимов АН. Обобщение линейных законов деформации и их приложение к задачам внутреннего трения. Прикладная математика и механика 1948; 12: 251–260.
287. Гладкий АВ, Ляшко ИИ, Мистецкий ГЕ. Алгоритмизация и численный расчет фильтрационных схем. Киев: Вища школа, 1981.
288. Дейнека ВС, Сергиенко ИВ, Скопецкий ВВ. Модели и методы решения задач с условиями сопряжения. К.: Наук. Думка, 1998.
289. Зарецкий ЮК. Теория консолидации грунтов. Москва: Наука, 1967.
290. Иванов ВВ. Теория приближенных методов и её применение к численному решению сингулярных интегральных уравнений. Киев: Наук. Думка, 1968.
291. Иванов НН. Об определении величин испаряемости. М.: Изв. ГГО, 1954. с. 189 – 196.
292. Иванов ПЛ. Грунты и основания гидротехнических сооружений. Москва: Высшая школа, 1991.
293. Іванюк ВА, Федорчук ВА. Адаптивний метод ідентифікації моделей нелінійних динамічних систем інтегральними рядами Вольтерри. Електронне моделювання 2019; 41(3): 33–42. doi: [10.15407/emodel.41.03.033](https://doi.org/10.15407/emodel.41.03.033)
294. Качинский НА. Корневая система растений в почвах подзолистого типа. (Исследование в связи с водным и питательным режимом почв). Ч. 1 / С вступ. ст. А.П. Левицкого. М. : Изд-во МОСХОЗ, 1925.
295. Ковальчук ПІ, Волошин ММ, Ковальчук ВП, Коломієць СС. Ідентифікація параметрів математичних моделей оперативного планування поливів при зрошенні. Меліорація і водне господарство 2003; 89: 19-27.
296. Коломієць СС, Яцик МВ, Ковальчук ВП, Пужай ОМ. Грунтово-гідрофізичне забезпечення математичного моделювання водного режиму меліорованих земель. Меліорація і водне господарство 2005; 92: 65-75.
297. Кутя ТВ, Герус ВА, Мартинюк ПМ. Моделювання вологоперенесення у ґрунтах з урахуванням теплових і хімічних чинників. Математичні методи та фізико-механічні поля 2017; 60(1): 169-177.

298. Лаврик ВИ, Бомба АЯ. О приближенном решении одной краевой задачи конвективной диффузии при плановой напорной фильтрации. В кн.: Математические методы исследования физических полей.-Киев: Институт математики НАН Украины, 1980, с.26-36.
299. Лаврик ВИ, Никифорович НА. Математическое моделирование в гидроэкологических исследованиях. К.: Фитосоциоцентр, 1998.
300. Лаврик ВИ, Фильчакова ВП, Яшин АА. Конформные отображения физико-топологических моделей. Киев: Наук. думка, 1990.
301. Летников АВ. Об историческом развитии теории дифференцирования с произвольным указателем. Математический сборник 1868; 3: 1–66.
302. Лыков АВ, Берковский БМ. Законы переноса в неньютоновских жидкостях. В кн.: Тепло-и массообмен в неньютоновских жидкостях. М.: Энергия,1968, с. 5–14.
303. Лыков АВ, Михайлов ЮА. Теория тепло- и массопереноса. М.;Л.: Госэнергоиздат, 1963.
304. Ляшко ИИ, Демченко ЛИ, Мистецкий ГЕ. Численное решение задач тепло-и массопереноса в пористых средах. К.: Наук. думка, 1991.
305. Ляшко СИ, Ключин ДА, Стешенко ГМ. Якісне та чисельне дослідження сингулярного керування конвективною дифузиею суміші радіоізотопів. Доповіді Національної академії наук України 2012; 8: 42-46.
306. Ляшко СИ. Оптимизация и математическое моделирование массопереноса подземных вод. К.: Наук. думка, 1998.
307. Ляшко СИ, Ключин ДА, Тригуб АС. Моделирование и оптимизация подземного массопереноса. К.: Наукова думка, 1998.
308. Ляшко СИ, Грищенко ОЮ, Федорова ВС, Загородня ГО. Про один скінченно-різницевий алгоритм моделювання процесів кінетики адсорбції. Доповіді Національної академії наук України 2015; 7: 32-38. doi: [10.15407/dopovidi2015.07.032](https://doi.org/10.15407/dopovidi2015.07.032)
309. Мелихова ЕВ. Математическое моделирование солевого режима при фертигации в почвогрунтах фрактальной структуры. Известия Нижневолжского

агроуниверситетского комплекса, Наука и высшее профессиональное образование 2017; 46(2): 259-255.

310. Мистецкий ГЕ. Гидростроительство. Автоматизация расчета асопереноса в почвогрунтах. Киев: Будівельник, 1985.

311. Полубаринова-Кочина ПЯ. Теория движения грунтовых вод. Москва: Наука, 1977.

312. Рыбакова СТ. Двумерные задачи конвективной диффузии в пористой среде. Фильтрация и массоперенос в пористых средах 1989; 90: 93–105.

313. Самарский АА. Теория разностных схем. М.: Наука, 1977.

314. Самко СГ, Килбас АА, Маричев ОИ. Интегралы и производные дробного порядка и некоторые их приложения. Минск: Наука и техника, 1987.

315. Сергиенко ИВ, Дейнека ВС. Идентификация градиентными методами параметров задач диффузии двухкомпонентных веществ в нанопористых средах. Доповіді НАН України 2010; 12: 42-49.

316. Сергиенко ИВ, Скопецкий ВВ, Дейнека ВС. Математическое моделирование и исследование процессов в неоднородных средах. Київ: Наук. думка, 1991.

317. Таукенова ФИ, Шхануков–Лафишев МХ. Разностные методы решения краевых задач для дифференциальных уравнений дробного порядка. Журнал вычислительной математики и математической физики 2006; 46(10): 1871–1881.

318. Учайкин ВВ. Метод дробных производных. Ульяновск: Изд-во "Артишок", 2008.

319. Флорин ВА. Основы механики грунтов. Т.2. Москва: Госстройиздат, 1961.

320. Хасанов ММ, Булгакова ГТ. Нелинейные и неравновесные эффекты в реологически сложных средах. Москва–Ижевск: Институт компьютерных исследований, 2003.

321. Штойко ДА, Писаренко ВА, Бичко ОС, Слаженко ЛІ. Розрахункові методи визначення сумарного випаровування і строків поливу с.-г. культур. Зрошувальне землеробство 1977; 3-8.

**Додаток А. Результати тестування алгоритму оптимізації обчислень для
моделей, що містять похідну Атангана–Балеану**

Таблиця А.1 – Значення $D(t) = {}^{ABC} \Delta_t^\alpha f(t)$ для $\varepsilon_1 = \varepsilon_2 = 10^{-10}$, $N = 100$, $\tau = 0,01$

та значення $D_r(t)$ з використанням чисельного інтегрування

α	β	$D(0,1)/D_r(0,1)$	$D(0,5)/D_r(0,5)$	$D(1,0)/D_r(1,0)$	$D(2,0)/D_r(2,0)$
0,1	0,1	0,28 / 0,43	0,42 / 0,50	0,48 / 0,53	0,55 / 0,56
0,1	0,3	0,38 / 0,46	0,69 / 0,73	0,87 / 0,89	1,09 / 1,09
0,1	0,6	0,24 / 0,25	0,65 / 0,66	0,99 / 0,99	1,49 / 1,49
0,1	1	0,10 / 0,10	0,51 / 0,51	1,00 / 1,00	1,99 / 2,00
0,5	0,1	0,42 / 0,63	0,47 / 0,55	0,45 / 0,49	0,42 / 0,42
0,5	0,3	0,57 / 0,68	0,79 / 0,84	0,85 / 0,87	0,87 / 0,87
0,5	0,6	0,37 / 0,39	0,79 / 0,79	1,02 / 1,03	1,28 / 1,28
0,5	1	0,16 / 0,16	0,64 / 0,64	1,10 / 1,11	1,86 / 1,87
0,9	0,1	1,24 / 1,72	0,36 / 0,39	0,19 / 0,20	0,10 / 0,11
0,9	0,3	1,74 / 2,03	0,75 / 0,77	0,46 / 0,47	0,30 / 0,30
0,9	0,6	1,22 / 1,28	0,97 / 0,98	0,77 / 0,78	0,62 / 0,62
0,9	1	0,57 / 0,58	1,01 / 1,03	1,12 / 1,14	1,22 / 1,25
0,99	0,1	0,88 / 0,92	0,18 / 0,20	0,10 / 0,10	0,05 / 0,05
0,99	0,3	1,59 / 1,66	0,48 / 0,50	0,30 / 0,30	0,18 / 0,18
0,99	0,6	1,51 / 1,57	0,77 / 0,79	0,58 / 0,59	0,44 / 0,43
0,99	1	0,95 / 1,00	0,96 / 1,03	0,96 / 1,07	0,97 / 1,12

Таблиця А.2 – Похибки $E_{k,n}$, швидкості збіжності $R_{k,n}$ та часи виконання T_n , с

α	$M = n = m$	$E_{1,n}$	$E_{2,n}$	$E_{3,n}$	$R_{1,n}$	$R_{2,n}$	$R_{3,n}$	T_n
0,1	10	$1,23 \cdot 10^{-4}$	$2,93 \cdot 10^{-6}$	$1,37 \cdot 10^{-4}$				6,72 с
0,1	20	$3,57 \cdot 10^{-5}$	$8,45 \cdot 10^{-7}$	$3,19 \cdot 10^{-5}$	1,78	1,79	2,11	1,11 с
0,1	40	$9,62 \cdot 10^{-6}$	$2,28 \cdot 10^{-7}$	$7,80 \cdot 10^{-6}$	1,89	1,88	2,03	2,29 с
0,1	80	$2,50 \cdot 10^{-6}$	$5,95 \cdot 10^{-8}$	$1,94 \cdot 10^{-6}$	1,94	1,93	2,00	5,44 с
0,1	160	$6,40 \cdot 10^{-7}$	$1,53 \cdot 10^{-8}$	$4,87 \cdot 10^{-7}$	1,96	1,96	1,997	16,68 с
0,5	10	$3,04 \cdot 10^{-4}$	$7,23 \cdot 10^{-6}$	$2,98 \cdot 10^{-4}$				5,20 с
0,5	20	$9,75 \cdot 10^{-5}$	$2,31 \cdot 10^{-6}$	$8,15 \cdot 10^{-5}$	1,64	1,64	1,87	0,73 с
0,5	40	$2,84 \cdot 10^{-5}$	$6,73 \cdot 10^{-7}$	$2,23 \cdot 10^{-5}$	1,78	1,77	1,87	1,54 с
0,5	80	$7,83 \cdot 10^{-6}$	$1,86 \cdot 10^{-7}$	$6,00 \cdot 10^{-6}$	1,85	1,85	1,89	4,29 с
0,5	160	$2,10 \cdot 10^{-6}$	$4,98 \cdot 10^{-8}$	$1,59 \cdot 10^{-6}$	1,89	1,90	1,91	12,50 с
0,9	10	$4,11 \cdot 10^{-3}$	$1,00 \cdot 10^{-4}$	$3,36 \cdot 10^{-3}$				0,03 с
0,9	20	$1,67 \cdot 10^{-3}$	$4,32 \cdot 10^{-5}$	$1,26 \cdot 10^{-3}$	1,29	1,21	1,41	0,33 с
0,9	40	$5,61 \cdot 10^{-4}$	$1,52 \cdot 10^{-5}$	$4,07 \cdot 10^{-4}$	1,57	1,51	1,63	0,50 с
0,9	80	$1,75 \cdot 10^{-4}$	$4,90 \cdot 10^{-6}$	$1,24 \cdot 10^{-4}$	1,67	1,63	1,70	1,55 с
0,9	160	$5,63 \cdot 10^{-5}$	$1,61 \cdot 10^{-6}$	$3,94 \cdot 10^{-5}$	1,63	1,60	1,66	6,20 с

Таблиця А.3 – Похибки $E_{k,n}$, швидкості збіжності $R_{k,n}$ та часи виконання T_n , с, для $n = 1000$ при зміні $M = m$

α	m	$E_{1,n}$	$E_{2,n}$	$E_{3,n}$	$R_{1,n}$	$R_{2,n}$	$R_{3,n}$	T_n
0,1	10	$1,35 \cdot 10^{-4}$	$3,17 \cdot 10^{-6}$	$1,01 \cdot 10^{-4}$				9,86 с
0,1	20	$3,75 \cdot 10^{-5}$	$8,83 \cdot 10^{-7}$	$2,82 \cdot 10^{-5}$	1,85	1,84	1,85	8,41 с
0,1	40	$9,85 \cdot 10^{-6}$	$2,33 \cdot 10^{-7}$	$7,38 \cdot 10^{-6}$	1,92	1,91	1,93	18,86 с
0,1	80	$2,53 \cdot 10^{-6}$	$6,02 \cdot 10^{-8}$	$1,89 \cdot 10^{-6}$	1,96	1,95	1,96	40,46 с
0,1	160	$6,44 \cdot 10^{-7}$	$1,53 \cdot 10^{-8}$	$4,81 \cdot 10^{-7}$	1,97	1,97	1,97	82,86 с
0,5	10	$3,34 \cdot 10^{-4}$	$7,82 \cdot 10^{-6}$	$2,52 \cdot 10^{-4}$				9,75 с
0,5	20	$1,02 \cdot 10^{-4}$	$2,41 \cdot 10^{-6}$	$7,69 \cdot 10^{-5}$	1,71	1,69	1,71	8,32 с
0,5	40	$2,90 \cdot 10^{-5}$	$6,89 \cdot 10^{-7}$	$2,18 \cdot 10^{-5}$	1,81	1,80	1,81	16,41 с
0,5	80	$7,92 \cdot 10^{-6}$	$1,88 \cdot 10^{-7}$	$5,94 \cdot 10^{-6}$	1,87	1,87	1,87	31,10 с
0,5	160	$2,11 \cdot 10^{-6}$	$5,00 \cdot 10^{-8}$	$1,58 \cdot 10^{-6}$	1,90	1,91	1,90	64,82 с
0,9	10	$4,51 \cdot 10^{-3}$	$1,08 \cdot 10^{-4}$	$3,39 \cdot 10^{-3}$				2,69 с
0,9	20	$1,75 \cdot 10^{-3}$	$4,51 \cdot 10^{-5}$	$1,28 \cdot 10^{-3}$	1,36	1,26	1,39	6,04 с
0,9	40	$5,75 \cdot 10^{-4}$	$1,55 \cdot 10^{-5}$	$4,13 \cdot 10^{-4}$	1,61	1,54	1,64	11,22 с
0,9	80	$1,77 \cdot 10^{-4}$	$4,95 \cdot 10^{-6}$	$1,25 \cdot 10^{-4}$	1,69	1,64	1,71	23,17 с
0,9	160	$5,66 \cdot 10^{-5}$	$1,62 \cdot 10^{-6}$	$3,96 \cdot 10^{-5}$	1,64	1,61	1,66	45,34 с

Таблиця А.4 – Похибки $E_{k,n}$, отримані за запропонованою схемою; $E_{k,n}^{(1)}$, наведені у [275] для методу скінченних різниць; $E_{k,n}^{(2)}$, наведені у [275] для методу розкладу у ряди Тейлора при зміні $M = n = m$ та $\alpha = 0,5$

M	$E_{1,n} / E_{1,n}^{(1)} / E_{1,n}^{(2)}$	$E_{2,n} / E_{2,n}^{(1)}$
10	$1,23 \cdot 10^{-4} / 6,16 \cdot 10^{-3} / 7,31 \cdot 10^{-4}$	$2,93 \cdot 10^{-6} / 3,90 \cdot 10^{-5}$
20	$3,57 \cdot 10^{-5} / 1,65 \cdot 10^{-3} / 9,83 \cdot 10^{-5}$	$8,45 \cdot 10^{-7} / 1,01 \cdot 10^{-5}$
40	$9,62 \cdot 10^{-6} / 4,33 \cdot 10^{-4} / 1,28 \cdot 10^{-5}$	$2,28 \cdot 10^{-7} / 2,61 \cdot 10^{-6}$
80	$2,50 \cdot 10^{-6} / 1,11 \cdot 10^{-4} / 1,64 \cdot 10^{-6}$	$5,95 \cdot 10^{-8} / 6,66 \cdot 10^{-7}$
160	$6,40 \cdot 10^{-7} / 2,85 \cdot 10^{-5} / 2,09 \cdot 10^{-7}$	$1,53 \cdot 10^{-8} / 1,68 \cdot 10^{-7}$

Таблиця А.5 – Похибки $E_{k,n}$, отримані за пропонованою схемою; $E_{k,n}^{(1)}$, наведені у [275] для методу скінченних різниць; $E_{k,n}^{(2)}$, наведені у [275] для методу розкладу у ряди Тейлора для $n = 1000$, $\alpha = 0,5$ при зміні m

m	$E_{2,n} / E_{2,n}^{(1)} / E_{2,n}^{(2)}$	$E_{3,n} / E_{3,n}^{(1)} / E_{3,n}^{(1)}$
10	$7,82 \cdot 10^{-6} / 3,90 \cdot 10^{-5} / 7,56 \cdot 10^{-6}$	$2,52 \cdot 10^{-4} / 7,89 \cdot 10^{-4} / 1,30 \cdot 10^{-4}$
20	$2,41 \cdot 10^{-6} / 1,01 \cdot 10^{-5} / 1,45 \cdot 10^{-6}$	$7,69 \cdot 10^{-5} / 2,15 \cdot 10^{-4} / 7,87 \cdot 10^{-5}$
40	$6,89 \cdot 10^{-7} / 2,61 \cdot 10^{-6} / 2,72 \cdot 10^{-7}$	$2,18 \cdot 10^{-5} / 5,73 \cdot 10^{-5} / 5,11 \cdot 10^{-5}$
80	$1,88 \cdot 10^{-7} / 6,66 \cdot 10^{-7} / 5,04 \cdot 10^{-8}$	$5,94 \cdot 10^{-6} / 1,53 \cdot 10^{-5} / 3,51 \cdot 10^{-5}$
160	$5,00 \cdot 10^{-8} / 1,68 \cdot 10^{-7} / 9,19 \cdot 10^{-9}$	$1,58 \cdot 10^{-6} / 4,47 \cdot 10^{-6} / 2,52 \cdot 10^{-5}$

Додаток Б. Довідка про використання результатів



НАЦІОНАЛЬНА АКАДЕМІЯ АГРАРНИХ НАУК УКРАЇНИ
ІНСТИТУТ ВОДНИХ ПРОБЛЕМ І МЕЛІОРАЦІЇ
(ІВПіМ)

вул. Васильківська 37, м. Київ, 03022, тел/факс (044) 257-40-01, тел. 257-40-30

E-mail: iwpim.naan@gmail.com

ДОВІДКА

про використання результатів дисертаційного дослідження
Богаєнко Всеволода Олександровича

на тему: «Математичне та комп'ютерне моделювання гідрогеоміграційних процесів з неklasичною динамікою на основі високопродуктивних обчислювальних алгоритмів»

Дана довідка засвідчує те, що результати дисертаційних досліджень старшого наукового співробітника Інституту кібернетики ім. В.М. Глушкова НАН України Богаєнко Всеволода Олександровича на тему: «Математичне та комп'ютерне моделювання гідрогеоміграційних процесів з неklasичною динамікою на основі високопродуктивних обчислювальних алгоритмів» використовується в Інституті водних проблем і меліорації НААН України в якості інструменту наукових досліджень та в системах підтримки прийняття рішень при управлінні процесами водоподачі та водовідведення.

Наукові розробки Богаєнко В.О. з моделювання процесів волого- та масоперенесення за складних гідрогеологічних умов були апробовані у виробничих умовах. Зокрема, використовуючи зібрані в польових умовах дані щодо динаміки доступності вологи рослинам, розроблені неklasичні моделі було успішно адаптовано до фактичних умов перебігу процесів у системі «грунт-рослина-атмосфера». При цьому, високопродуктивні обчислювальні алгоритми дозволили з достатньою для сільськогосподарської практики швидкістю та точністю прогнозувати динаміку доступності вологи та пропонувати час та норми поливів.

Варто зазначити, що запропоновані нові математичні моделі дробового порядку дозволили точно прогнозувати зміни вологості ґрунту у зонах, що вміщують коріння рослин, у ситуаціях, коли складна структура ґрунту зазнає суттєвого впливу, зокрема неоднорідних процесів ущільнення та розпушення. На відміну емпіричних методів, стандартних для сільськогосподарської практики, так і методів, базованих на диференціальних моделях цілого порядку, широко використовуваних у гідрологічній та сільськогосподарській науці, застосування у системах підтримки прийняття рішень нових моделей значно підвищило точність та ефективність управління процесами водоподачі при зрошенні та водорегулюванні.

Таким чином, отримані апробаційні результати є достовірними та свідчать про високий рівень обґрунтованості розроблених методів та засобів математичного моделювання і ефективність їх практичного використання.

Заступник директора з наукової роботи,
д. с.-г. н., професор,
член-кореспондент НААН



Андрій ШАТКОВСЬКИЙ