

MIMD-паралельний вирішувач рівнянь для мережного динамічного об'єкту з розподіленими параметрами

Гусєва Г.Б., Молдованова О.В.
Кафедра ЕОМ ДонНТУ
anna-guseva@ukr.net

Abstract

Gusyeva G.B., Moldovanova O.V. MIMD parallel equation solver for dynamic distributed parameter network objects: The article describes using of one-step two-point block numerical method for simulation of complex dynamic distributed parameters network objects (DDPNO). The approaches to parallelization of equation solver for the DNO simulation are examined in this article. Virtual parallel model of equation solver is developed and necessary steps of its devirtualization are emphasized. The target computing MIMD resource is also briefly represented. The results of experimental researches are also given in this article.

Вступ

Розвиток чисельних методів для вирішення систем нелінійних диференціальних рівнянь привів до появи широкого спектру програмних пакетів для чисельного аналізу і динамічного моделювання в багатьох галузях, таких як хімічна, біологічна, гірнична справа та інші. При збільшенні розмірності моделюючої системи, кількості її параметрів, що характеризують динамічні процеси, з'являється необхідність в аналізі підходів до розпаралелювання обчислень. Метою розпаралелювання є досягнення ефективного та швидкого вирішення на багато процесорних високопродуктивних обчислювальних системах таких складних науково-технічних задач, які неможливо вирішити на машинах фон-неймановського типу. Але на теперішній час існує не так багато чисельних методів інтегрування диференціальних рівнянь, орієнтованих на обчислювальну систему з паралельною архітектурою. Більшість з них мають складну структуру, що приводить до втрати ефективності. Тому питання дослідження нових паралельних чисельних методів, позбавлених вказаних недоліків, є безсумнівно актуальним.

Формальний опис мережних динамічних об'єктів

Мережні динамічні об'єкти (МДО) відносяться до класу технічно-складних систем, таких як газо- та водопостачання, шахтні вентиляційні мережі, електричні мережі та ін. Структура складних динамічних систем характеризується нелінійністю характеристик, великою розмірністю та

розподіленістю параметрів процесів, тому майже не піддається аналітичному аналізу. Основним методом вирішення задач автоматизації та технологічного проектування є математичне моделювання.

Формальний опис МДО складається з двох суттєвих частин [1, 2]: опису топології та опису рівнянь фізичних процесів, що мають місце в гілках та вузлах мережі.

З точки зору математичного опису динаміки процесів мережний об'єкт можна поділити на два класи: МДО з зосередженими параметрами (МДОЗП) і МДО з розподіленими параметрами (МДОРП).

Для описання МДОРП використовуються еліптичні, параболічні, гіперболічні та спеціальні диференціальні рівняння у часткових похідних. Для останньої системи рівнянь змінна стану в кожний момент часу – це функція відносно однієї або декількох координат простору.

Об'єктом дослідження є шахтна вентиляційна мережа (ШВМ) (рис. 1). Топологія ШВМ представляється у вигляді графа.

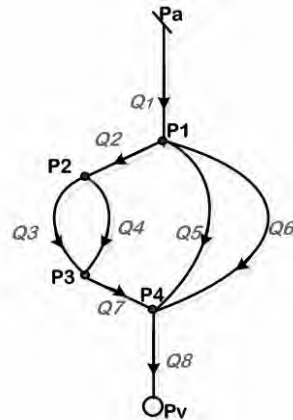


Рисунок 1 - Граф тестового мережного об'єкту

Для гілки з індексом i процеси зміни значення потоку повітря Q_i та тиску P_i представляються у вигляді наступних рівнянь [1]:

$$\begin{cases} -\frac{\partial p_i}{\partial x} = r_i Q_i^2 + \frac{\rho}{S_i} * \frac{\partial Q_i}{\partial t} \\ -\frac{\partial p_i}{\partial t} = \frac{\rho a^2}{S_i} * \frac{\partial Q_i}{\partial x} \end{cases} \quad (1)$$

де r_i – питомий аеродинамічний опір; a – швидкість звуку в повітрі (330 м/сек.); ρ – густина повітря; S_i – площа перетину гілки; x – координата вздовж гілки.

Наступним кроком є приведення системи рівнянь до вигляду, придатного для чисельного вирішення. Цей крок включає апроксимацію диференціальних рівнянь у часткових похідних за методом прямих та їх розрішення відносно похідних тих змінних, що мають бути визначені [5].

Розглянемо розбивку j -ої гілки ШВМ за методом прямих (рис. 2).

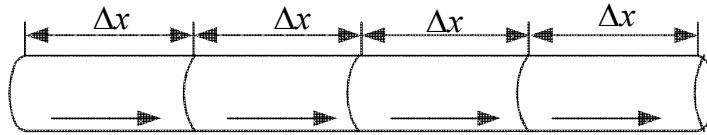


Рисунок 2 - Апроксимація j -ої гілки ШВМ за методом прямих

Тут гілку було розбито на M елементів із постійним кроком дискретизації Δx . Граничні умови: $P_n = P_{атм}$, $P_k = f(Q)$, де $P_{атм}$ – атмосферний тиск.

Система рівнянь буде виглядати наступним чином:

$$\begin{cases} \frac{P_{j,q-1} - P_{j,q}}{\Delta x} = \frac{\rho}{S} * \frac{dQ_{j,q}}{dt} + r Q_{j,q}^2 \\ \frac{dP_{j,q}}{dt} = \frac{\rho a^2}{S} * (Q_{j,q} - Q_{j,q+1}) \end{cases}, \quad (2)$$

де $q = 1, 2, \dots, M$.

Надалі необхідно обрати чисельний метод для розв'язання отриманої після апроксимації системи диференціальних рівнянь.

Блоковий чисельний метод

Блоковим будемо називати метод, при якому для блоку з k точок нові k значень функції обчислюються одночасно. Ця особливість методів, по-перше, узгоджується з архітектурою паралельних обчислювальних систем, а, по-друге, дозволяє обчислювати коефіцієнти різницевих формул не в процесі інтегрування, а на етапі розробки методу, що значно збільшує ефективність обчислень.

Паралельні блокові чисельні методи розв'язують задачу Коші. Для звичайного диференціального рівняння першого порядку задача Коші має наступний вигляд [3, 4]:

$$x' = f(t, x), \quad x(t_0) = x_0. \quad (3)$$

Розглянемо рішення задачі Коші k точковим блоковим методом (рис. 3). Множину M точок рівномірної сітки $\{t_m\}$, $m = 1, 2, \dots, M$ з кроком τ розіб'ємо на блоки, що містять по k точок, $kN \geq M$. У кожному блоці уведемо номер точки $i = 0, 1, \dots, k$ та позначимо через $t_{n,i}$ точку n -го блоку з номером i . Точку $t_{n,0}$ назвемо початком блоку n , а $t_{n,k}$ – кінцем блоку. Очевидно, що $t_{n,k} = t_n + t_{n,0}$. При цьому початкова точка до блоку не включається.

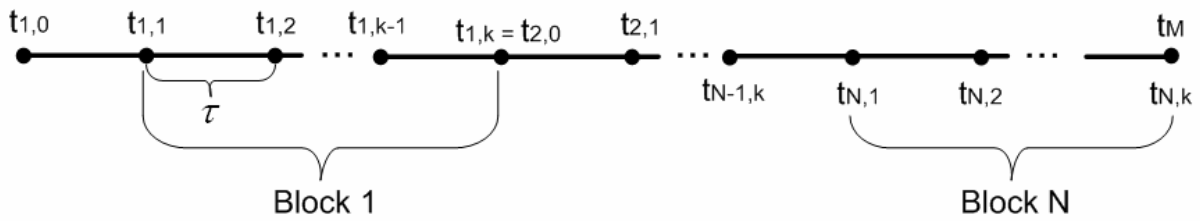


Рисунок 3 - Схема розбиття на блоки

Існує два типи блокових методів: однокрокові та багатокрокові блокові методи. Якщо для розрахунку значень у новому блоці використовується тільки остання точка попереднього блоку – мова ведеться про однокроковий блоковий метод (рис. 4).

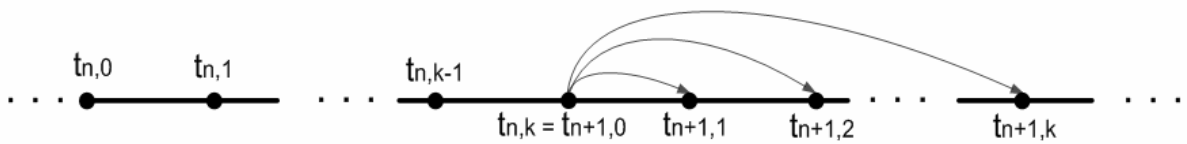


Рисунок 4 - Схема обчислення однокроковим блоковим методом

У загальному випадку рівняння m -крокового k -точкового блокового методу записується у наступному вигляді [3, 4]:

$$u_{n,i} = u_{n,0} + i\tau \left[\sum_{j=1}^m b_{i,j} F_{n,j-m} + \sum_{j=1}^k a_{i,j} F_{n,j} \right], \quad (4)$$

де $i = \overline{1, k}$ $n = \overline{1, N}$ $F_{n,j} = f(t_n + j\tau, u_{n,j})$.

Надалі будемо досліджувати однокроковий блоковий метод. На рис. 5 представлено шаблон однокрокової k -точкової різницевої схеми.

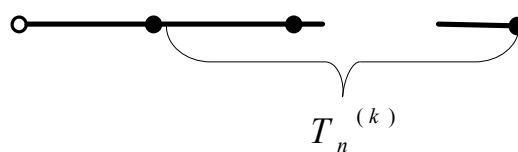


Рисунок 5 - Шаблон однокрокової k -точкової різницевої схеми

Формула для обчислення нових k значень є наступною:

$$u_{n,i} = u_{n,0} + i\tau \left[b_i F_{n,0} + \sum_{j=1}^k a_{i,j} F_{n,j} \right]. \quad (5)$$

Для визначення коефіцієнтів $a_{i,j}$ та b_i треба побудувати інтерполяційний багаточлен Лагранжа $L_k(t)$ із вузлами інтерполяції $t_n, i, i=0, 1, \dots, k$ та відповідними до них значеннями $F_{n,i} = f(t_n + i\tau, u_{n,i})$ і проінтегрувати у границях $(t_n, t_n + i\tau)$. Формули для двоточкового блокового методу при $k = 2$ будуть мати наступний вигляд:

$$U_{n,1} = \frac{\tau}{12}(5F_{n,0} + 8F_{n,1} - F_{n,2}) + U_{n,0};$$

$$U_{n,2} = \frac{\tau}{3}(F_{n,0} + 4F_{n,1} + F_{n,2}) + U_{n,0}. \quad (6)$$

Оскільки для обчислення наближених значень рішення задачі Коши необхідно вирішити нелінійну систему рівнянь, можна використовувати для цього наступний ітераційний процес:

$$u_{n,i,0} = u_{n,0} + i\tau F_{n,0}, \quad i = \overline{1, k}, \quad n = 1, 2, \dots, N$$

$$u_{n,i,s+1} = u_{n,0} + i\tau(b_i F_{n,0} + \sum_{j=1}^k a_{i,j} F_{n,i,s}), \quad s = \overline{0, k-1},$$

де n – номер блоку, i – номер точки блоку, s – номер ітерації.

Метод ітерацій дозволяє проводити обчислення паралельно для кожного вузла блоку. Після виконання k кроків обчислювань локальна помилка у вузлах блоку буде мати порядок $O(\tau^{k+1})$. Таким чином, двоточковий метод має порядок апроксимації $O(\tau^3)$ [3, 4].

Розробка послідовного вирішувача рівнянь для ШВМ та проведення експериментальних досліджень

На основі блокового однокрокового двоточкового методу було розроблено послідовний вирішувач рівнянь для тестової ШВМ [6]. Для встановлення працездатності послідовної імплементації вирішувача рівнянь було проведено чотири групи експериментів (табл. 1).

Результати експериментів підтвердили працездатність послідовного вирішувача рівнянь, заснованого на блоковому однокроковому двоточковому чисельному методі, і можливість його використання для моделювання складних МДО.

Експериментально було визначено, що блоковий метод відзначається кращою збіжністю та точністю обчислень, ніж метод Адамса-Башфорта другого порядку точності.

Таблиця 1. Опис проведених експериментів

Номер групи експериментів	Мета проведення експериментів
Перша група	Визначення впливу параметрів τ та M на процеси $Q_I(t)$ та $Q_M(t)$.
Друга група	Визначення впливу фізичних характеристик виробки на хід процесу.
Третя група	Дослідження залежності характеру зміни процесів $Q_I(t)$ та $Q_M(t)$ від форми вхідного значення тиску у гілці $P(t)$.
Четверта група	Порівняння результатів роботи блокового методу із методом Адамса-Башфорта другого порядку точності.

Віртуальна паралельна модель вирішувача рівнянь для ШВМ та етапи її девіртуалізації

Одним із необхідних етапів при розробці паралельного програмного забезпечення для моделювання МДОРП є створення паралельної віртуальної моделі об'єкту, що буде являти собою набір взаємопов'язаних паралельно працюючих процесів, кожний з яких займається обчисленням визначеної частини графу дискретизованого мережного об'єкту [5].

Обсяг даних, якими обмінюються віртуальні паралельні процеси, і кількість самих процесів, потрібних для рішення задачі моделювання МДОРП, залежать не тільки від розмірності та складності топології мережного об'єкту, але й від особливостей архітектури наявного паралельного обчислювального ресурсу та способу угруповання й розподілу віртуальних процесів. Процес переходу від віртуальної паралельної MIMD-EOM до цільової, особливостям якої повинна відповідати структура програмного забезпечення, що розроблюється, називається процесом девіртуалізації [5].

Процес девіртуалізації містить у собі наступні етапи:

1. Априорна оцінка паралельної віртуальної моделі, визначення підходів до її розпаралелювання: розгляд можливих рівнів розпаралелювання та варіантів, які мають якісно кращі показники по рівномірності завантаження процесів та по обсягу передачі даних;

2. Вибір та аналіз паралельної цільової архітектури; аналіз характеристик наявного обчислювального ресурсу: визначення числа доступних обчислювальних вузлів та процесорів на кожний вузол, схеми зв'язків між ними.

3. Вибір засобів паралельного програмування, орієнтованих на обрану цільову архітектуру. Аналіз характеристик функцій обраної комунікаційної бібліотеки та визначення найбільш ефективних з них для реалізації обміну даними між віртуальними процесами.

Було розглянуто чотири наступних рівня розпаралелювання вирішувачів рівнянь для ШВМ (рис. 6).

При реалізації паралельного вирішувача для всієї МДОРП на перших рівнях розпаралелювання процес девіртуалізації стає майже неможливим, особливо у тому випадку, коли кожний віртуальний процес становиться еквівалентним одному процесору цільового паралельного ресурсу. Тоді стає задача об'єднання віртуальних процесів з метою зменшення кількості процесорів, задіяних у процесі моделювання. Цей спосіб отримання ефективності від розпаралелювання вирішувача веде до наступних рівнів. На третьому рівні віртуальні процеси паралельно розраховують кожну гілку або вузол моделюючого об'єкту. Підхід до розпаралелювання на четвертому рівні називається розбиттям графа мережі. Для отримання

ефективних результатів необхідно розподілити гілки по процесах таким чином, щоб мінімізувати кількість операцій обмінів між ними.

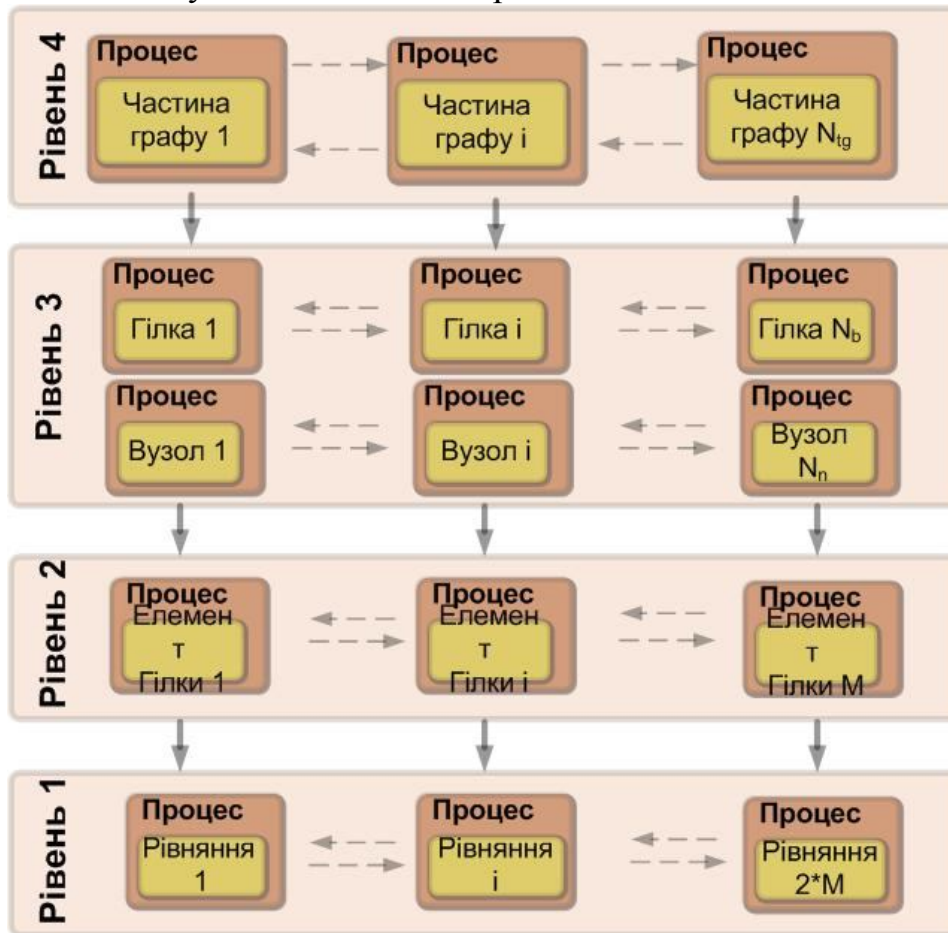


Рисунок 6 - Рівні розпаралелювання вирішувачів рівнянь

Цільова паралельна обчислювальна архітектура

У якості цільового паралельного обчислювального ресурсу виступає система NEC-SX8 Штутгартського Обчислювального Центру Високої Продуктивності (HLRS).

Концепція системи NEC-SX8 (рис. 7) побудована на основі трьох підсистем, які, апаратно доповнюючи одна одну, створюють потужний обчислювальний комплекс. Це вузли SX-8, система AsAma TX-7 та кластер Intel Nosona, які мають доступ до одних й тих самих даних через волоконно-оптичний канал. Система IA64 із загальною пам'яттю, іменована AsAma, служить для виконання преобробки даних перед початком моделювання на векторних процесорах SX-8. Кластер, побудований на основі процесорів Intel EM64T та технології Infiniband, призначений для постобробки та візуалізації даних, а також виконує роль обчислювального сервера для багатодисциплінарних задач.

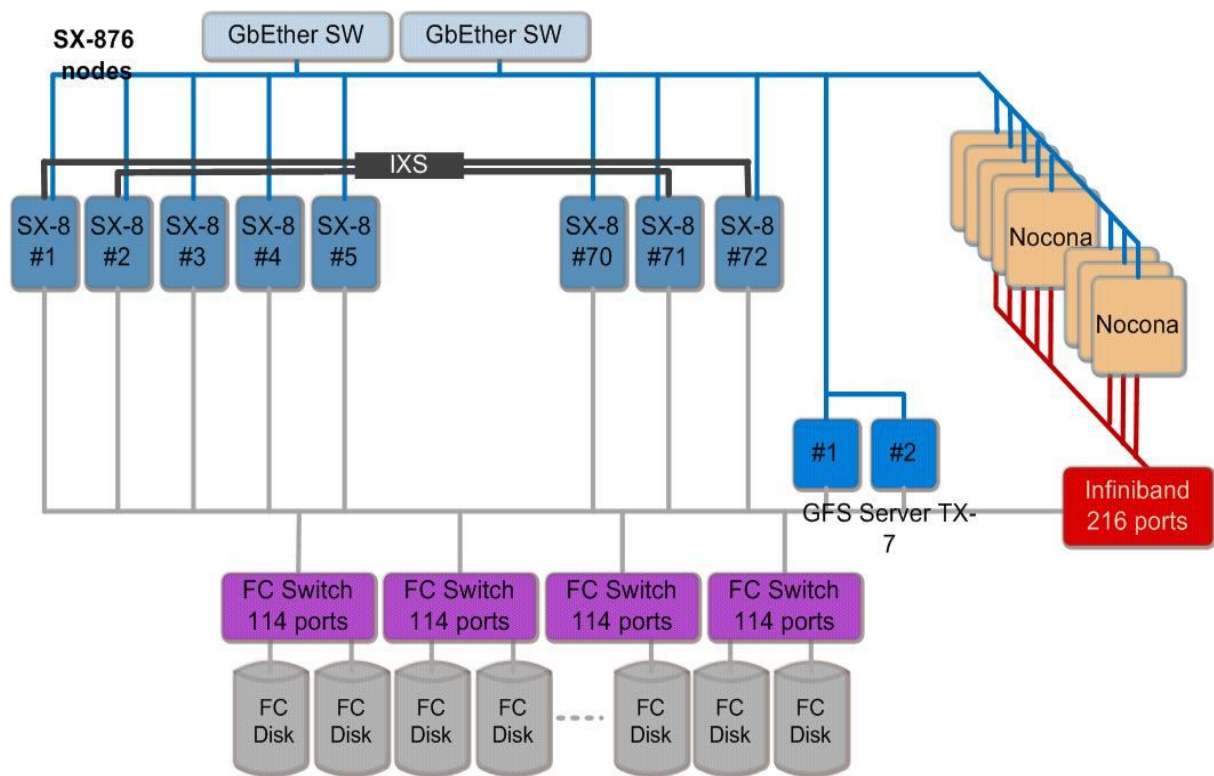


Рисунок 7 - Загальна структура системи NEC-SX8

Розробка паралельного вирішувача рівнянь для ШВМ та проведення експериментальних досліджень

На кожному з розглянутих рівнів розпаралелювання було розроблено алгоритм роботи вирішувача рівнянь та імплементовано за допомогою мови С та стандарту MPI взаємодії паралельних процесів у паралельному середовищі [6].

Запуск розробленого програмного забезпечення здійснювався на двох обчислювальних системах: на кластері AsAma TX-7 і на системі SX-8.

Найбільшу результативність показав четвертий рівень розпаралелювання. У якості показників ефективності роботи паралельної програми є коефіцієнт прискорення S і показник ефективності E . У табл. 2 зведені результати запуску паралельної програми на кластері AsAma TX-7.

На системі SX-8 також були проведені експерименти щодо часу здійснення міжвузлової комунікації між MPI-процесами. Результати цих експериментів наведені у табл. 3, де $N_{\text{вуз}}$ – кількість вузлів, необхідних для розрахунків, а $N_{\text{проц}}$ – кількість ЦП на кожний вузол.

Крім того, було протестовано декілька комунікаційних MPI-функцій з метою визначення найбільш ефективних з них для реалізації вирішувача рівнянь. Було розглянуто однонаправлені неблокувальні функції MPI_ISend , MPI_IRecv та двонаправлені блокувальні функції $MPI_SendRecv$.

Таблиця 2. Ефективність роботи вирішувача рівнянь на кластері AsAma на четвертому рівні розпаралелювання

Функції MPI	τ	$T_{нар}, c$	$T_{посл}, c$	S	E
<i>MPI_Isend / MPI_Irecv</i>	0,00007	48,65	141,4	2,906	1,4532
	0,00008	39,94	119,27	2,986	1,4931
<i>MPI_SendRecv</i>	0,00007	45,76	141,4	3,09	1,545
	0,00008	39,62	119,27	3,01	1,5052

Таблиця 3. Ефективність роботи вирішувача рівнянь на SX-8 на четвертому рівні розпаралелення

$N_{вуз}$	$N_{проц}$	Функції MPI	τ	$T_{нар}, c$	$T_{посл}, c$	S	E
1	2	<i>ISend / Irecv</i>	0,00007	119,85	141,4	1,1798	0,59
			0,00008	99,88	119,27	1,1941	0,597
		<i>SendRecv + ISend/Irecv</i>	0,00007	111,09	141,4	1,2728	0,636
			0,00008	93,01	119,27	1,2823	0,641
2	1	<i>ISend / Irecv</i>	0,00007	123,76	141,4	1,1425	0,571
			0,00008	102,34	119,27	1,1654	0,583
		<i>SendRecv + ISend/Irecv</i>	0,00007	115,54	141,4	1,2238	0,612
			0,00008	96,95	119,27	1,2302	0,615

Висновки

Проведені експериментальні дослідження доказали, що блоковий однокроковий двоточковий метод може використовуватися для рішення задач моделювання МДОРП. Крім того, у порівнянні з досить розповсюдженим методом Адамса-Башфорта другого порядку він показує кращі результати у тривалості рішення та має вищу точність обчислень.

Експериментально було визначено, що при розробці паралельної віртуальної моделі вирішувача рівнянь для складних МДО не має сенсу розбивати задачу на такі невеликі обчислювальні ділянки, яких потребують перші два рівні розпаралелювання. Час роботи паралельної задачі збільшується за рахунок великої кількості обмінів між процесорами, при тому що реальний обсяг обчислень на кожний процесор є незначним. Тому була поставлена задача об'єднання або «укрупнення» віртуальних процесів з метою зниження кількості комунікаційних ліній. При розбивці графа МДО, тобто на четвертому рівні, було отримано найвищу ефективність від розпаралелювання.

Крім того, слід відмітити, що при використанні блокувальних двонаправлених та неблокувальних однонаправлених MPI-функцій, що здійснюють комунікацію між MPI-процесами, досягається більша ефективність від розпаралелювання задачі моделювання МДО. Таку комбінацію функцій рекомендується використовувати тільки у тих

випадках, коли треба здійснити невелику кількість операцій обміну. Хоча двонаправлена функція має вищу швидкість передачі даних, але час виконання паралельної задачі може збільшитися за рахунок очікування завершення обмінів кожної з цих блокувальних функцій.

Література

1. Абрамов Ф.А., Фельдман Л.П., Святный В.А. Моделирование динамических процессов рудничной аэрологии.– К.:Наука.думка, 1981.– 284с.
2. Святный В.А. Моделирование аэрогазодинамических процессов и разработка систем управления проветриванием систем шахт. Докт. дисс., Донецьк, ДПИ, 1986.
3. Фельдман Л.П., Петренко А.І., Дмитрієва О.А. Чисельні методи в інформатиці. – К.: Видавнича група ВНУ, 2006. – 480с.
4. Дмитриева О.А. Алгоритмические методы повышения эффективности параллельных вычислительных систем при решении многомерных динамических задач: Дис. канд. тех. наук: 05.13.13. – Донецк, 2001.
5. Svjatnyj V., Moldovanova O., Smagin A., Resch M., Keller R.: Rabenseifner R. Virtuelle Simulationsmodelle und ein Devirtualisierungsvorgang für die Entwicklung der parallelen Simulatoren von komplexen dynamischen Systemen/ 19. Symposium ASIM 2006, Tagungsband, 2006, S. 416 – 421.
6. Гусєва Г.Б. Магістерська дисертація, ДонНТУ, Донецьк, 2007

Дата надходження до редакції 03.12.2007 р.