

УДК 681.322

Є.О. Шквар

Національний авіаційний університет, Київ

## ЕФЕКТИВНІСТЬ ВИКОРИСТАННЯ ТЕХНОЛОГІЇ ВІРТУАЛЬНОЇ БАГАТОПОТОЧНОСТІ ПРИ ПАРАЛЕЛЬНОМУ РОЗВ'ЯЗАННІ РІВНЯННЯ ПУАССОНА

*Розглянуто шляхи масштабування продуктивності обчислень при чисельному розв'язанні типового елементу постановок багатьох аерогідродинамічних задач, який одночасно є одним з найвимогливіших до ресурсів обчислювальної техніки – рівняння Пуассона. Особливу увагу приділено дослідженню ефективного використання при побудові MPI алгоритмів багатоядерної архітектури вузлів сучасних кластерних систем, а також оцінці додаткового приросту швидкодії за рахунок задіявання технології Intel Hyper-threading. Показано, що ефективність використання останньої при розв'язанні рівняння Пуассона залежить від розмірності задачі і для просторового розрахункового випадку Hyper-threading може забезпечити прискорення, близьке до використання реальних ядер.*

**Ключові слова:** паралельні багатопоточні та розподілені обчислення, Hyper-Threading, рівняння Пуассона.

### Вступ

Сьогодення аерогідродинамічних досліджень характеризується стрімким зростанням долі і ефективності числових методів у порівнянні з експериментальним та наближено-аналітичним визначенням характеристик обтікання. Визначальною рисою сучасних обчислювальних технологій в галузі аерогідродинаміки є значна вимогливість до обчислювальних ресурсів, яка стимулює значний інтерес як до потужної обчислювальної техніки, так і до розробки орієнтованих на її архітектуру відповідних методів паралелізації обчислень. Зазначимо, що при існуючому різноманітті формулювань аерогідродинамічних задач, обумовленому як різними геометричними конфігураціями, так і широким діапазоном варіювання режимних параметрів, одним з базових математичних об'єктів, що присутній у значній кількості сучасних розрахункових методів для визначення полів тиску та потенціалу швидкості, є рівняння Пуассона. Це – еліптичне диференціальне рівняння у частинних похідних другого порядку, що в операторній та координатній формах представлення має наступний вигляд:

$$\Delta\Phi = \nabla^2\Phi = \frac{\partial^2\Phi}{\partial x^2} + \frac{\partial^2\Phi}{\partial y^2} + \frac{\partial^2\Phi}{\partial z^2} = S_\Phi, \quad (1)$$

де  $\Phi = \Phi(x, y, z)$  – шукана функція;  $x, y, z$  – декартові координати,  $S_\Phi$  – джерельний член, який є відомою функцією координат в розрахунковій області  $W$  і задається, як правило, у вигляді залежності від інших розрахункових змінних задачі. Зазначимо, що крім аерогідродинаміки це рівняння традиційно використовується і при розв'язанні задач теплопровідності, дифузії, магніто-електростатики і навіть при обробці зображень. Обумовлена нелінійністю вихідних рівнянь аерогідродинаміки ітераційна структура алгоритмів чисельних методів потребує багаторазового розв'язання рівняння Пуассона з метою поступового уточнення розрахункових полів, а неухильно зростаючі вимоги до роздільної здатності про-

ведення обчислень вимагають також дискретизації розрахункової області дрібними сітками, що разом з еліптичним типом самого цього рівняння різко збільшує ресурсовитратність процедури відшукування розв'язку. До того ж, сучасні методи моделювання зсувних течій на рівні відтворення особливостей структури і динаміки турбулентних вихроутворень, такі як пряме чисельне моделювання (DNS – Direct Numerical Simulation) та моделювання великих вихорів (LES – Large Eddy Simulation), передбачають інтегрування вихідних рівнянь по часовій змінній з дуже дрібним кроком, а на кожному з цих кроків треба розв'язувати рівняння Пуассона для тиску, що потребує окрім значних витрат обчислювальних ресурсів також і суттєвих витрат часу і є найсуттєвішим обмеженням впровадженню цих перспективних технологій в дослідницьку та інженерну діяльність. Як показали дослідження [1], саме розв'язання рівняння Пуассона є найвимогливішим елементом алгоритму LES щодо утилізації обчислювального часу, оскільки складало близько 45% загального часу проведення розрахунків як при використанні послідовного алгоритму, так і при багатопоточних обчисленнях на SMP комп'ютері з двома чотириядерними процесорами. У останньому випадку прискорення обчислень на 8-ми ядрах при розв'язанні рівняння Пуассона склало лише 2,83 рази, що й обумовило величину остаточного прискорення розрахунків за багатопоточним алгоритмом застосованої реалізації метода LES у 2,86 разів. Зазначені чинники є вичерпним обґрунтуванням пошуку методів ефективного розпаралелювання процесу чисельного розв'язання рівняння Пуассона. Сучасні обчислювальні кластери мають, як правило, багатопроцесорні вузли, а сучасні процесори до того ж є переважно багатоядерними. Крім того, компанія Intel інтенсивно впроваджує в свої розробки і просуває на ринок технологію Hyper-threading (HT), яка дає змогу повніше використати ресурси кожного з процесорних ядер завдяки можливості одночасного виконання двох потоків на кожному з них.

Метою проведеного дослідження є визначення доцільності та додаткової ефективності розпаралелювання процесу розв'язання (1) при побудові розподіленого MPI алгоритму з орієнтацією на використання багатопоточних обчислень на кожному вузлі.

### Формалізація задачі: вихідні рівняння та межові умови, використані спрощення

Розглянемо задачу відшукування розв'язку рівняння (1) в просторовій кубічній (або прямокутній) області  $W$ . Зв'яжемо осі координат  $(x, y, z)$  з ребрами області і покриємо просторову розрахункову область  $W$  рівномірною сіткою. Вважатимемо (як розповсюджений типовий випадок), що на усіх межах розрахункової області задано умови Діріхле (або Неймана), тобто межові значення змінної  $\phi$  (або її похідної по нормалі до межі) передбачаються відомими і заданими відповідними розподілами. Джерельний член задаватимемо виразом  $S_\phi = (i + j - k) / (i_{\max} + j_{\max} + k_{\max})$ , де  $i, j, k$  – номери вузлів скінченно-різницевої сітки, якою покривається розрахункова область  $W$  вздовж осей  $x, y$  та  $z$  відповідно,  $1 \leq i \leq i_{\max}, 1 \leq j \leq j_{\max}, 1 \leq k \leq k_{\max}$ . Для даної модельної задачі з метою спрощення подальших перетворень використаємо рівномірну сітку з кроками вздовж осей  $\Delta x, \Delta y, \Delta z$  відповідно. Кількість вузлів в усіх напрямках візьмемо однаковою, тобто  $i_{\max} = j_{\max} = k_{\max} = M$ . Різницєва апроксимація рівняння (1) на даній сітці зводить його до вигляду

$$\begin{aligned} \Phi_{i,j,k}^{n+1} = & 0.5((\Phi_{i+1,j,k}^n + \Phi_{i-1,j,k}^n)\Delta x^{-2} + \\ & + (\Phi_{i,j+1,k}^n + \Phi_{i,j-1,k}^n)\Delta y^{-2} + \\ & + (\Phi_{i,j,k+1}^n + \Phi_{i,j,k-1}^n)\Delta z^{-2} - S_\phi^n) \times \\ & \times (\Delta x^{-2} + \Delta y^{-2} + \Delta z^{-2})^{-1}, \end{aligned} \quad (2)$$

де  $n$  – номер ітерації. Отримане рівняння (2) дозволяє обчислювати значення змінної  $\Phi_{i,j,k}$  за значеннями  $\Phi$  в шести сусідніх з нею точках різницєвого шаблону. При розгляді двовимірної задачі різницєва схема (2) спрощується відкиданням доданків, що містять  $\Delta z$ . Для знаходження поля  $\Phi(x, y, z)$  у вигляді масиву вузлових значень  $\Phi_{i,j,k} = \Phi(x_i, y_j, z_k)$  використовуватимемо метод Лібмана. Відповідний ітераційний процес мусить тривати до тих пір, доки не буде досягнута необхідна точність, яка контролюється виконанням умови

$$\left| (\Phi_{i,j,k}^{n+1} - \Phi_{i,j,k}^n) / (\Phi_{i,j,k}^{n+1})_{\max} \right| \leq \varepsilon,$$

де  $\varepsilon$  – параметр, що задає точність розрахунків. Ураховуючи, що метою дослідження є не сам розв'язок (2) і аналіз його властивостей, а прискорення процесу його отримання, приймемо ще одне спрощення, яке полягатиме в тому, що буде розглядатися процес чисельного розв'язання (2) протягом наперед заданої кількості ітерацій  $N_{it}$ .

### Декомпозиція розрахункової області

Здійснимо геометричну декомпозицію, рівномірно розбивши побудовану сітку на блоки вздовж напрямку  $z$  (у випадку просторової області) чи у напрямку  $y$  (при розгляді двовимірної задачі) так, щоб вузли на межах блоків перекривалися, тобто межовий вузол одного блоку був внутрішнім для сусіднього з ним і навпаки. Число цих блоків мусить відповідати кількості MPI процесів, тобто числу вузлів кластера, на яких передбачається виконання обчислень. В межах кожного з цих процесів виділятимемо по декілька потоків згідно з числом ядер (у тому числі і віртуальних HT), що будуть задіяні на вузлі кластера для виконання обчислень. Для цього кожний блок розрахункової області розіб'ємо на підблоки, але вже без взаємного перекриття, враховуючи симетричність обробки даних усіма ядрами процесорів вузла (SMP, CMP) – рис. 1.

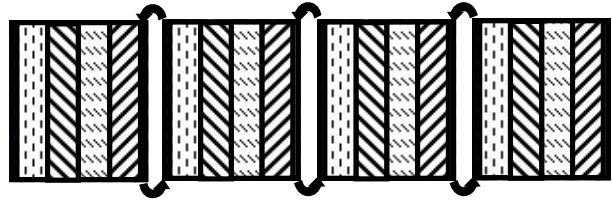


Рис. 1. Декомпозиція розрахункової області вздовж відповідної координати на блоки для обробки MPI процесами, а блоків – на підблоки (заштриховані смужки), що обробляються потоками (OpenMP)

Організуємо послідовний обхід усіх внутрішніх точок кожного з блоків обчисленнями за формулою (2), після чого значення розрахункових змінних на межах сусідніх блоків коригуються шляхом організації MPI пересилань. Отже при обрахунках використано два підходи: MPI для реалізації можливості проведення обчислень на розподілених системах та багатопоточне програмування в реалізації OpenMP. Ці технології є суттєво відмінними і, певною мірою, конкуруючими, оскільки кожна з них розрахована на принципово різні архітектури обчислювальних систем. Таким чином, в даному дослідженні реалізовано гібридний підхід, який дозволяє за рахунок реалізації багатопоточної обробки в межах кожного з вузлів суттєво зменшити об'єм інформації, якою обмінюються між собою MPI процеси завдяки багатопоточній організації обчислень всередині кожного з процесів засобами SMP.

### Особливості реалізації технології розпаралелювання та основні характеристики використаної обчислювальної техніки

Запропонована гібридна технологія розпаралелювання обчислень рівняння Пуассона (2) була запрограмована мовою Fortran в реалізації Intel Visual Fortran 11 [2, 3]. При виконанні обчислень застосовано кластерну систему, побудовану з трьох вузлів, об'єднаних гігабітною мережею топології “зірка”, яку склали PCI-E мережні карти та комутатор. Кожен з

вузлів містив чотирьохядерний процесор Intel Core i7-2600k з активованою (або вимкненою) технологією HT, що працював на тактовій частоті 4.5 ГГц, та 8 ГБ оперативної пам'яті DDR3-1600, яка функціонувала у двоканальному режимі. Кластер працював під керуванням операційної системи Windows-7 64-bit, а міжвузловий інтерфейс забезпечувався пакетом MPICH2 (ver. 1.4.1p1). Крім того, з метою порівняння ефективності HT технології ряд розрахунків було здійснено на кластері НТУУ "КПІ", вузли якого містять по два чотириядерних процесори Intel Xeon E5345 з частотою 2.33 ГГц (8 реальних процесорних ядер), та 8 ГБ оперативної пам'яті. Кластер НТУУ "КПІ" функціонував під керуванням MS Windows 2008 HPC Edition.

### Результати паралельних обчислень

Розрахунки здійснювалися на кожному із зазначених вище комп'ютерів шляхом послідовного збільшення кількості виділених потоків. Перш за все, розглянемо та проаналізуємо простіший як з точки зору програмування, так і з позицій вимогливості до обчислювальних ресурсів двовимірний випадок. Розрахунки здійснювалися на кожному із зазначених вище комп'ютерів шляхом послідовного збільшення кількості виділених потоків при  $M=2000$  та  $N_{it}=500$  (рис. 2). У результаті на вузлі кластера НТУУ "КПІ" з вісьмома ядрами вдалося отримати лінійне прискорення по мірі зростання кількості задіяних ядер до 6, яке при подальшому зростанні кількості ядер до 8 продовжувало зростати, хоч і з дещо меншим градієнтом, у результаті чого було отримано максимальне прискорення в 6,86 разів.

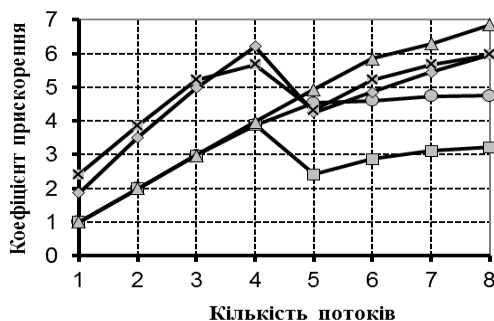


Рис. 2. Залежність коефіцієнта прискорення від кількості виділених потоків при розв'язанні рівняння Пуассона (1) в двовимірній постановці при  $M=2000$  та  $N_{it}=500$ :  $\Delta$  – Dual Intel Xeon E5345;  $\circ$  – Intel Core I7-2600k (HT on);  $\square$  – Intel Core I7-2600k (HT off);  $\diamond$  – 2 вузли;  $\times$  – 3 вузли

При застосуванні комп'ютера на процесорі Intel Core i7-2600k прискорення зростало лінійно пропорційно числу потоків лише в межах наявної кількості реальних ядер незалежно від активізації технології HT. Тим не менш, варто відмітити той факт, що задіявання останньої забезпечило хоч і незначний, але позитивний ефект від віртуальних ядер у вигляді зростання прискорення до 4,75 разів при кількості потоків, кратних восьми, тобто сумі реальних і віртуальних ядер цього процесора, тоді як при вимкненій HT максимальне прискорення не перевищило 3,22.

Наступна серія розрахунків проводилася на кластері з вузлами на процесорах Intel Core i7-2600k з метою оцінки кумулятивного ефекту від застосування MPI та OpenMP технологій розпаралелювання. У випадку використання двох та трьох вузлів прискорення обчислень досягалося лише при кількості потоків, що виділялися на кожному з вузлів, яка не перевищувала числа реальних ядер. Подальше збільшення кількості потоків приводило до немонотонної зміни прискорення, отже позитивний ефект від HT для двовимірної постановки задачі досягти не вдалося. Наступний етап досліджень стосувався тривимірної постановки задачі і здійснювався при двох наборах параметрів, а саме:  $M=500$ ,  $N_{it}=500$  та  $M=1200$ ,  $N_{it}=100$ . На вузлі кластера НТУУ "КПІ" з вісьмома ядрами при  $M=500$  і  $N_{it}=500$  вдалося отримати лінійне прискорення по мірі зростання кількості задіяних ядер до 8, у результаті чого було отримано максимальне прискорення в 7,27 разів (рис. 3). При застосуванні комп'ютера на процесорі Intel Core i7-2600k прискорення зростало лінійно аналогічно попередньому випадку, тобто віртуальні ядра поводити себе в даних обчисленнях у повній мірі подібно до реальних. Подальше збільшення кількості потоків призвело на обох обчислювальних системах до немонотонного розподілу коефіцієнта прискорення, причому кращі результати відповідали кількості потоків, що була кратною сумарній кількості задіяних ядер (як реальних, так і віртуальних). При вимкненій HT лінійне зростання прискорення спостерігалось лише до максимального значення 3,76 при кількості потоків, що не перевищували наявної кількості реальних ядер, тобто 4, після чого залежність поведи себе немонотонним чином.



Рис. 3. Залежність коефіцієнта прискорення від кількості виділених потоків при розв'язанні рівняння Пуассона (1) в просторовій області при  $M=500$  та  $N_{it}=500$ :  $\Delta$  – Dual Intel Xeon E5345;  $\circ$  – Intel Core I7-2600k (HT on);  $\square$  – Intel Core I7-2600k (HT off)

Отриманий результат демонструє значно кращу, ніж відповідні двовимірні розрахунки, адаптованість розрахункового алгоритму до використання HT технології при задіяванні лише його багатопоточних властивостей, оскільки ці результати здобуті в розрахунках лише на одному вузлі кластерної системи. Результати, отримані в результаті обчислень на кластері з вузлами на процесорах Intel Core i7-2600k при  $M=1200$  та  $N_{it}=100$ , наведені на рис. 4.

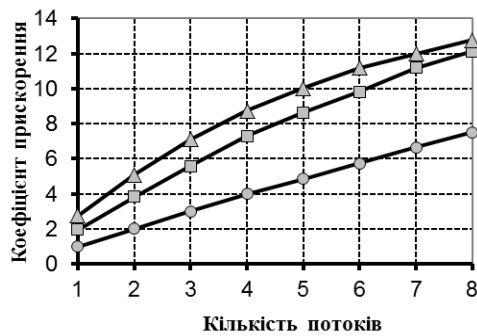


Рис. 4. Залежність коефіцієнта прискорення від кількості виділених потоків при розв'язанні рівняння Пуассона (1) в просторовій області на кластері при  $M = 1200$  та  $N_{ii} = 100$ :  
○ – 1 вузол; □ – 2 вузли; △ – 3 вузли

Як і в попередньому випадку, в однопоточному варіанті спостерігалось лінійне зростання прискорення, причому при 8 задіяних ядрах максимальне прискорення склало 7,5 разів, що на 3,2% перевищує попередній результат, отриманий при  $M = 500$ , отже збільшення як розмірності задачі, так і розміру масивів є позитивними факторами, який впливає на масштабування продуктивності обчислень. При запуску задачі на двох та трьох вузлах кластера ефективність обчислень зростає, але з меншим градієнтом, до того ж залежності втрачають лінійність, причиною чому є затримки, обумовлені досить повільним у порівнянні зі швидкістю доступу до оперативної пам'яті гігабітним мережним інтерфейсом. Тим не менш, максимальне прискорення, отримане на двох та трьох вузлах кластера склало 12,13 і 12,77 разів, що, враховуючи задіяні для цього відповідно 8 та 12 реальних ядер, виглядає досить обнадійливим результатом, який отриманий завдяки використанню НТ. Також варто зазначити, що є всі підстави очікувати суттєвого покращення отриманих результатів при застосуванні більш продуктивного міжвузлового інтерфейсу, ніж використана в даному дослідженні гігабітна мережа.

#### ЭФФЕКТИВНОСТЬ ИСПОЛЬЗОВАНИЯ ТЕХНОЛОГИИ ВИРТУАЛЬНОЙ МНОГОПОТОЧНОСТИ ПРИ ПАРАЛЛЕЛЬНОМ РЕШЕНИИ УРАВНЕНИЯ ПУАССОНА

Е.А. Шквар

*Рассмотрены пути масштабирования эффективности вычислений при численном решении типового элемента постановок многих аэродинамических задач, который одновременно является одним из наиболее требовательных к ресурсам вычислительной техники – уравнения Пуассона. Особое внимание уделено исследованию эффективного использования при построении MPI алгоритмов многоядерной архитектуры узлов современных кластерных систем, а также оценке дополнительного прироста продуктивности вычислений за счет задействования технологии Intel Hyper-threading. Показано, что эффективность применения последней при решении уравнения Пуассона зависит от размерности задачи и для пространственного расчетного случая Hyper-threading может обеспечить ускорение, близкое к использованию реальных ядер.*

**Ключевые слова:** параллельные многопоточные и распределенные вычисления, Hyper-Threading, уравнение Пуассона.

#### EFFICIENCY OF USAGE THE VIRTUAL MULTITHREADING TECHNOLOGY FOR PARALLEL SOLVING THE POISSON EQUATION

Ye.O. Shkvar

*The ways of computational performance scaling for the numerical solution of a typical element of many CFD problems, which is also one of the most demanding to the computer resources – the Poisson equation are considered. Special attention was paid to the efficient usage of MPI algorithms together with multi-core architecture of nodes of modern cluster systems, as well as evaluating additional productivity increasing due to Intel Hyper-threading technology. The obtained results demonstrate the dependence of Hyper-threading efficiency for the Poisson equation solving on the dimensionality of the problem and the Hyper-threading ability to provide the acceleration in 3D case, which is close to the use of real cores.*

**Keywords:** parallel multithreaded and distributed computations, Hyper-Threading, Poisson equation.

#### ВИСНОВКИ

Застосування гібридного паралельного алгоритму, що об'єднує MPI та OpenMP технології, розподіленого та багатопоточного програмування для розв'язання рівняння Пуассона на кластері з вузлами, процесори яких підтримують НТ технологію, не забезпечило ніяких переваг від наявності останньої при розв'язанні двовимірної задачі. Натомість, при вирішенні цієї ж задачі в просторовій постановці НТ продемонструвала таку ефективність, яка не поступається результатам використання тієї ж кількості реальних ядер. Це дає можливість розглядати НТ як потужний резерв масштабування обчислювальної продуктивності при вирішенні широкого кола вимогливих до обчислювальних ресурсів прикладних задач, тим чи іншим чином пов'язаних з розв'язанням рівняння Пуассона, а також обґрунтовує доцільність використання при побудові кластерних систем елементної бази з підтримкою НТ.

#### Список літератури

1. Шквар Є.О. Оцінювання масштабування обчислювальної продуктивності паралельної реалізації методу великих вихорів / Є.О. Шквар // Вісник НАУ. – 2012. – № 1. – С. 157-166.
2. Шквар Є.О. Інтегрована гібридна технологія паралельних обчислень / Є.О. Шквар // Інтегровані технології та енергозбереження: щоквартальний науково-практичний журнал НТУ (ХП). – Х.: НТУ (ХП). – 2010. – № 1. – С. 86-99.
3. Шквар Є.О. Гібридний метод паралельних обчислень / Є.О. Шквар // Вісник Черкаського університету: серія Прикладна математика. Інформатика. – Вип. 172. – 2010. – С. 123-136.

Надійшла до редколегії 2.03.2012

**Рецензент:** д-р техн. наук, проф. С.А. Калкаманов, Харківський університет Повітряних Сил ім. І. Кожедуба, Харків.