УДК 004.451.24

СТІРЕНКО С.Г.

ОРГАНІЗАЦІЯ ВІДКЛАДЕНИХ ОБЧИСЛЕНЬ В КЛАСТЕРНИХ СИСТЕМАХ

Рассматриваются варианты организации отложенных вычислений в параллельных системах, применяемые в различных инструментальных средствах. Предложены ряд подходов для определения начала передачи данных. Приведен математический формализм и семантика операций, программная реализация которых позволит соответствовать математической модели.

Discusses options for lazy evaluation in parallel systems used in different tools. Proposed several approaches to determine the start of data transmission. The mathematical formalism of this model and semantics of operations on the model are described.

Відкладення передачі даних

При спробі підвищити ефективність паралельної системи розробники стикаються з проблемою ефективного завантаження її ресурсів та реорганізації обчислювальних процесів. [1] Для виконання такої оптимізації пропонується відкладати початок передачі даних $t_{comm,k}^{\mathrm{i}\to\mathrm{j}}$ на певний проміжок часу, тобто $t_{comm,k}^{\mathrm{i}\to\mathrm{j}} \geq t_{rdy,k}^{\mathrm{i}}$. Це доцільно лише за умови що дані підготовлені раніше ніж запитане перше звернення до них $t_{rdy,k}^{\mathrm{i}} < t_{req,k}^{\mathrm{j}}$. Зокрема, можна відкласти передачу даних до такого моменту часу, у який звільниться якнайменш m_k пам'яті на вузлі j, за умови якщо зберігання цього обсягу на вузлі i не матиме негативного впливу на ефективність. Пропонується не відкладати передачу даних, якщо її зберігання на вузлі і має негативний вплив на ефективність

$$M_i'(t) - M_i(t) = m_k \forall: t_{rdy,k}^{i} \leq t \leq t_{comm,k}^{i\to j} + t_{T,k}^{i\to j} \Longrightarrow \begin{cases} \text{якщо } K_E' \geq K_E, \text{ відкладати;} \\ \text{інакше не відкладати.} \end{cases} \quad (1)$$

Вибір моменту часу, у який буде розпочата передача даних, має бути виконаний з ціллю оптимізації виразів з [2] за умови виконання (1). Оскільки на час прийняття рішення $t \leq t_{comm,k}^{\mathrm{i}\to\mathrm{j}}$ в динаміці відомі лише оцінені значення $t_{req,k}^{\mathrm{j}}$ та $t_{T,k}^{\mathrm{i}\to\mathrm{j}}$ (реальні значення можуть відрізнятися через вплив багатьох факторів та будуть відомі лише після завершення передачі даних та початку обчислень над k-тим блоком даних, тому їх використання неможливе), необхідно запропонувати евристичні підходи до визначення моменту початку передачі даних.

Розглянемо декілька типових випадків, у яких відкладення передачі може мати позитивний вплив на коефіцієнт ефективності.

Розглянемо виконання обчислень паралельно на двох вузлах обчислювальної системи з локальною пам'яттю. Для спрощення вважаємо, на кожному вузлі виконується лише один процес обчислень, при цьому можуть виконуватись інші системні процеси, можливості вплинути на які в рамках користувацької програми обчислень неможливо. В певному місці обчислень у вузлі 1 генеруються блоки даних (1), (2) та (3), які в подальшому мають бути використані для обчислень, що виконуються на вузлі 2 (рис. 1), причому $t_{rdy,1}^{1} = t_{rdy,2}^{1} = t_{rdy,3}^{1}$. У момент часу $t_{comm,1}^{1\to 2} = t_{rdy,1}^{1}$ починається передача блоку даних (1) даних, яка триває $t_{T,1}^{1\to 2}$, після чого у вузлі 2 можуть бути розпочаті обчислення. Одночасно з цим може виконуватись передача блоку даних (2) $t_{comm,2}^{1\to 2} = t_{rdy,2}^{\mathrm{i}} + t_{T,1}^{1\to 2}$. У певний момент часу

$$t_x : t_{comm,1}^{1\to 2} < t_x < t_{comm,1}^{1\to 2} + t_{T,1}^{1\to 2} \quad (2)$$

закінчуються обчислення над блоком даних (1), які призвели до рішення, що виконувати обчислення над блоком даних (2) не потрібно (наприклад, результат вже достатньо точний). Тому наступними на вузлі 2 будуть виконані обчислення з блоком даних (3). Однак, вони ще не передані на цей вузол і не можуть бути передані, оскільки виконується передача блоку даних (2). Лише після закінчення його передачі можна буде розпочати передачу блоку даних (3) $t_{comm,3}^{1\to 2} = t_{rdy,3}^{1} + t_{T,1}^{1\to 2} + t_{T,2}^{1\to 2}$. Таким чином, час очікування введення та виведення у вузлі 2 складатиме $t_{w,i/o}^{2} = t_{rdy,1}^{i} + \sum_{k=1}^{3} t_{T,k}^{1\to 2} - t_X$.

У випадку ж відкладення передачі даних (рис. 2) до моменту

$$t_{comm,2}^{1\to 2} \geq t_x \geq t_{T,1}^{1\to 2} \quad (3)$$

безпосередня передача даних не розпочнеться. Таким чином, час очікування введення та виведення у вузлі 2 складатиме $t_{w,i/o}^{2\prime} = t_{T,3}^{1\to 2}$.

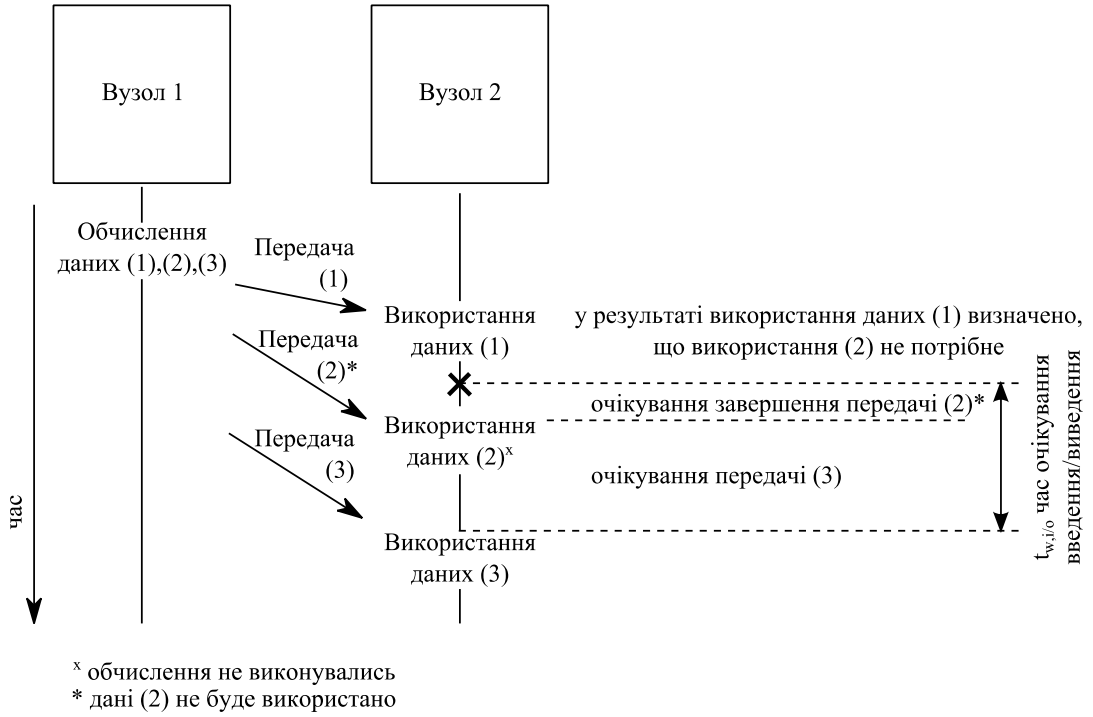

Рис. 1. Передача даних, які не будуть використані через скасування обчислень

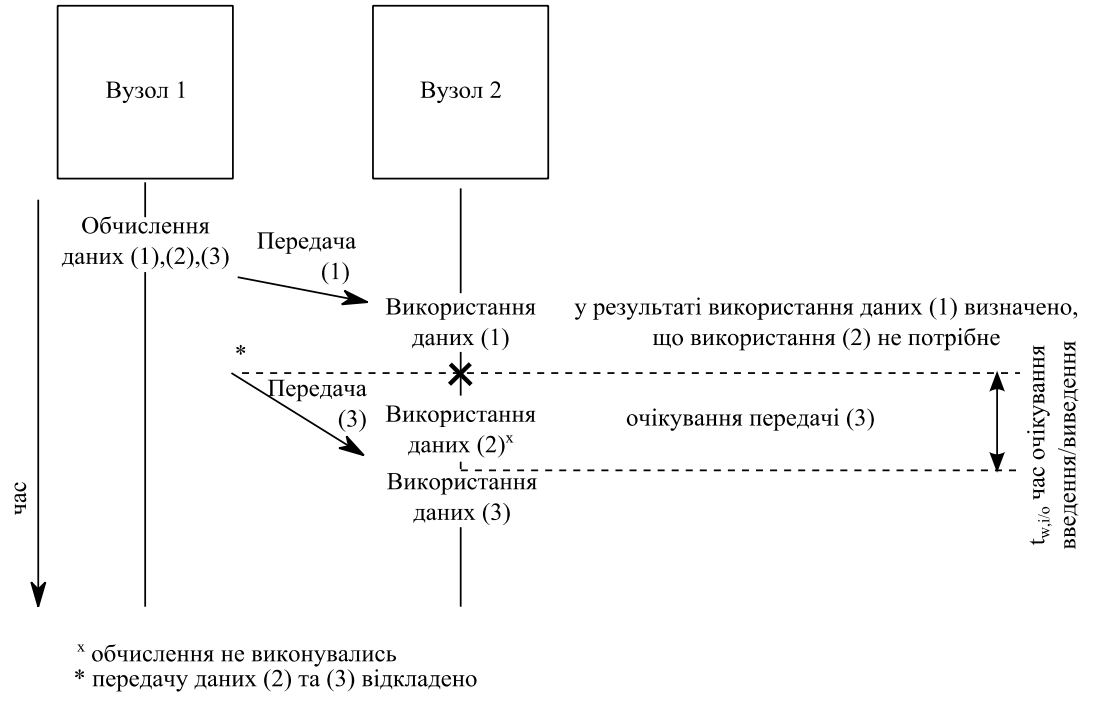

Рис. 2. Відкладення передачі даних при можливості скасування обчислень

Різниця часу очікування введення та виведення становитиме

$$t^{2\prime}_{w,i/o} - t^{2}_{w,i/o} = t^{1\to 2}_{T,3} - t^{\text{i}}_{rdy,1} - \sum_{k=1}^{3} t^{1\to 2}_{T,k} + t_x = t_x - t^{\text{i}}_{rdy,1} - t^{1\to 2}_{T,1} - t^{1\to 2}_{T,2}$$

з урахуванням (2) отримаємо

$$t^{2\prime}_{w,i/o} - t^{2}_{w,i/o} \le t^{1\to 2}_{comm,2} + t^{1\to 2}_{T,2} - t^{1\to 2}_{T,1} = t^{1\to 2}_{comm,2} - t^{1\to 2}_{T,1}$$

або

$$-t^{1\to 2}_{comm,2} + t^{1\to 2}_{T,1} \le -t^{2\prime}_{w,i/o} + t^{2}_{w,i/o}$$

звідки за визначенням (3)

$$t^{2\prime}_{w,i/o} - t^{2}_{w,i/o} \ge 0 \qquad (4)$$

Тобто час очікування введення та виведення без відкладення передачі даних *більший* ніж час очікування введення та виведення з відкладен-

ням передачі даних. При цьому, за умови відкладення передачі блоку даних (2) у вузлі 2 не виділяється пам'ять для його зберігання, тобто

$$\begin{cases} M_1'(t) = M_1(t)\ \forall t; \\ M_2'(t) = M_2(t)\ \forall t : t < t_{rdy,1}^1 + t_{T,1}^{1\to 2}; \\ M_2'(t) = M_2\left(t - t_{w,i/o}^2 + t_{w,i/o}^{2\prime}\right) - \\ \quad -m_2 \forall t : t \geq t_{rdy,1} + t_{T,1}^{1\to 2}, \end{cases} \quad (5)$$

що можна скоротити до

$$M_i'(t) - M_i(t) \leq 0\ \forall i\ \in \overline{(1,2)}, t \quad (6)$$

Підставляючи (4), (6) до визначення коефіцієнту ефективності [1], отримаємо

$$K_E'(t) - K_E(t) \geq 0 \quad (7)$$

тобто відкладення передачі даних збільшило коефіцієнт ефективності.

Іншою типовою ситуацією, в якій відкладання передачі даних на певний час може зменшити час очікування введення та виведення, є використання обчислювальної системи, в якій підтримується балансування навантаження. Розглянемо виконання обчислень на системі з трьома вузлами з локальною пам'яттю, які попарно зв'язані між собою. Вузол 1 виконує обчислення з генерації блоків даних (1) та (2) одночасно $t_{rdy,1}^1 = t_{rdy,2}^1$, а на вузлі 2 мають виконуватись обчислення над цими блоками даних послідовно (рис. 3). Передача блоку даних (1) починається відразу по його готовності $t_{comm,1}^{1\to 2} = t_{rdy,1}^1$, після завершення якої розпочинається передача блоку даних (2) $t_{comm,2}^{1\to 2} = t_{rdy,2}^1 + t_{T,1}^{1\to 2}$. В деякий момент часу t_B, такий що

$$0 \leq t_B - t_{comm,2}^{1\to 2} = t_{T,2}^{1\to 2} \quad (8)$$

в обчислювальній системі відбувається балансування навантаження, після якого обчислення, з використанням блоку даних (2) переносяться до вузла 3. При цьому, блок даних (2) має бути переданий повторно. Час очікування введення та виведення даних в цьому випадку складатиме

$$t_{w,i/o}^3 = t_{rdy,2}^1 + t_{T,1}^{1\to 2} + t_{T,2}^{1\to 2} + t_{T,2}^{1\to 3} - t_B \quad (9)$$

причому на вузлі 2 було використано пам'ять для тимчасового збереження блоку даних (2), який фактично не використовувався на ньому.

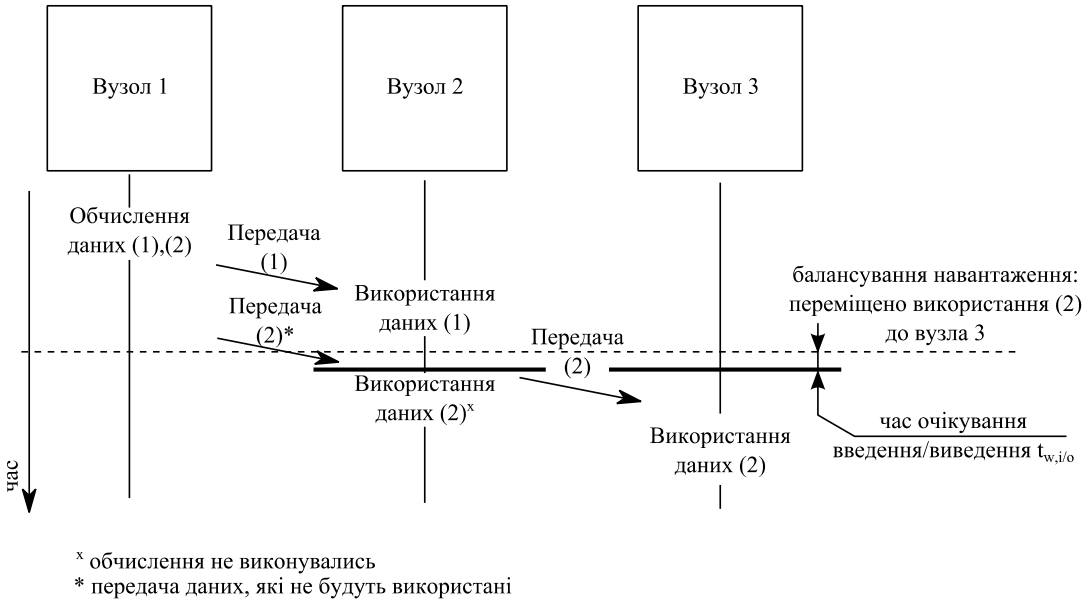

Рис. 3. Повторна передача даних через балансування навантаження

У випадку відкладання передачі даних до часу

$$t_{comm,2}^{1\to 2} \geq t_B \quad (10)$$

(рис. 4), тобто балансування навантаження відбудеться до початку передачі даних, можна встановити, що виконувати передачу даних необхідно одразу до вузла (3). В цьому випадку, відсутня необхідність передавати дані на проміжковий вузол (2), та час очікування введення та виведення даних становитиме

$$t_{w,i/o}^{3\prime} = t_{T,2}^{1\to 3} \quad (11)$$

Розрахуємо різницю часу очікування введення та виведення без відкладення обчислень та з таким

$$t_{w,i/o}^3 - t_{w,i/o}^{3\prime} = t_{rdy,2}^1 + t_{T,1}^{1\to 2} + t_{T,2}^{1\to 2} + t_{T,2}^{2\to 3} - t_B - t_{T,2}^{1\to 3}$$

що з урахуванням (8) може бути перетворене на нерівність

$$t_{w,i/o}^3 - t_{w,i/o}^{3\prime} \geq t_{rdy,2}^1 + t_{T,1}^{1\to 2} + t_{T,2}^{1\to 2} + t_{T,2}^{2\to 3} - t_{comm,2}^{1\to 2} - t_{T,2}^{1\to 3}$$

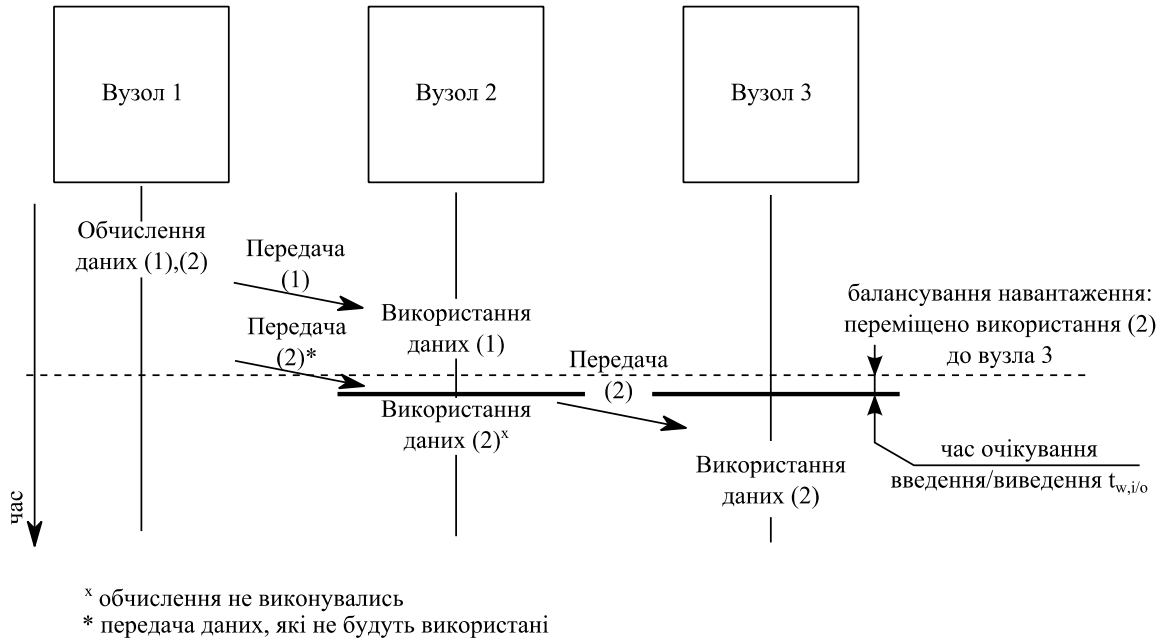

Рис. 4. Відкладення передачі даних при балансуванні навантаження

а оскільки за умови негайної передачі даних $t_{comm,2}^{1\to2} = t_{rdy,2}^{1} + t_{T,1}^{1\to2}$ 2, можемо спростити до

$$t_{w,i/o}^{3} - t_{w,i/o}^{3\prime} \geq t_{rdy,2}^{1} + t_{T,1}^{1\to2} + t_{T,2}^{1\to2} + t_{T,2}^{2\to3} - t_{rdy,2}^{1} - t_{T,1}^{1\to2} - t_{T,2}^{1\to3} = = t_{T,2}^{1\to2} + t_{T,2}^{2\to3} + t_{T,2}^{2\to3}$$

Тобто різниця в часі очікування введення та виведення даних залежить від значення виразу $(t_{T,2}^{1\to2} + t_{T,2}^{2\to3}) - t_{T,2}^{1\to3}$, а саме від різниці часу, що необхідний на передачу блоку даних (2) з вузла 1 до вузла 3 через проміжковий вузол 2 або напряму. Причому час передачі розглядається на програмному рівні, тобто не обов'язково залежить від наявності та параметрів фізичних зв'язків між вузлами. Обчислювальна система намагатиметься обрати такий шлях та спосіб передачі даних, який би мінімізував час виконання передачі [2], тому для більшості випадків за умови застосування відомих стратегій організації обчислювального процесу $(t_{T,2}^{1\to2} + t_{T,2}^{2\to3}) - t_{T,2}^{1\to3} \geq 0$, звідки випливає

$$t_{w,i/o}^{3} - t_{w,i/o}^{3\prime} \geq 0 \qquad (12)$$

Таким чином, завдяки відкладенню передачі даних існує можливість зменшити час очікування введення та виведення, який матиме вплив на коефіцієнт ефективності аналогічний розглянутому в (4), (7). Тим не менш у ряді випадків за умови застосування вкрай неоптимальної стратегії вибору шляху передачі даних між вузлами неоднорідної системи, може спостерігатися зворотній ефект. Для запобігання цьому необхідно вести статистику під час виконання та враховувати її при прийнятті рішення про відкладення передачі даних.

Окрім цих типових випадків, які були показані у спрощеному вигляді у порівнянні з реальними обчисленнями на системах з локальною пам'яттю, існує ще багато більш складних випадків. Зокрема випадки, у яких відкладення передачі даних може зменшити обсяги пам'яті, що використовується в певному вузлі, що також може впливати на коефіцієнт ефективності при виконанні обчислень.

Висновки

При застосуванні низькорівневих моделей програмування, розв'язання цієї задачі покладається на користувача обчислювальної системи, який завдяки знанню особливостей задачі може забезпечити коректну організацію передачі даних, тобто таку, за якої дані доступні на необхідному вузлі перед початком обчислень. Однак, така реалізація не завжди матиме достатню ефективність через невикористання можливостей перевпорядкування передачі, довільного порядку передачі та інших. Тому в сучасних системах застосовуються моделі програмування з більш високим рівнем абстракцій, зокрема такі, що мають переносити дані користувача за описаними ним правилами автоматично на вузол, на якому будуть виконуватися обчислення з використанням цих даних. Використання таких моделей дозволяє виконувати ряд оптимізації, в тому числі врахування особливостей гетерогенної системи, в процесі виконання обчислень з метою збільшення їх ефективності.

Організація передачі даних може впливати на коефіцієнт ефективності через два фактори: час очікування та виконання введення та виведення даних за мережею, яка зазвичай використовується для зв'язку вузлів системи з локальною пам'яттю, та обсягу використаної пам'яті. Останнє впливає на коефіцієнт ефективності опосередковано: так обсяг використаної пам'яті впливає на кількість кеш-промахів, яка в свою чергу визначає час очікування підкачки даних з диску в основну пам'ять, причому останній входить до часу очікування, у який не виконуються обчислення даної задачі, а лише у найкращому випадку деякі системні задачі завдяки принципу розподілу часу.

Запропоновано застосувати технологію відкладення обчислень до передачі даних в системах з локальною пам'яттю, а саме відкладати момент початку передачі даних між вузлами від моменту готовності даних, якщо дані підготовані раніше ніж використовуються перший раз. У зворотньому випадку, відкладення даних недоцільне, оскільки лише збільшить час очікування введення та виведення, під час якого не виконуються обчислення. Відкладення обчислень особливо корисне у випадку наявності в обчислювальній системі механізмів балансування навантаження, які можуть призвести до надлишкового копіювання даних, якщо не було однозначно визначено на якому вузлі будуть виконуватись обчислення. Цей підхід найкраще застосовувати в процесі виконання обчислень, оскільки він може дозволити не передавати дані, якщо в процесі обчислень прийнято рішення про невиконання певної частини обчислень або відсутність необхідності у використанні ряду даних завдяки специфічному для задачі аналізу, наприклад якщо необхідну точність обчислень вже досягнуто.

З іншого боку, відкладення передачі даних передбачає необхідність зберігання їх у вузлі-джерелі впродовж певного часу, що збільшить на цей час обсяг використаної пам'яті в цьому вузлі. Збільшення обсягу використаної пам'яті може зменшити коефіцієнт ефективності та нівелювати ефект, досягнутий завдяки відкладенню передачі даних. Тому необхідно дослідити взаємний вплив цих параметрів та запропонувати методи, які враховують обидва, для розрахунку часу початку передачі даних, такого що мінімізує час очікування передачі даних та середній обсяг використаної пам'яті за час передачі даних.

Запропоновано ряд підходів до визначення часу початку передачі даних. Серед них слід відзначити два підходи, які є відповідно нижнім і верхнім обмеженням на доцільні значення часу початку передачі даних: передавати дані відразу по готовності, тобто фактично на виконувати відкладення, та передавати дані за запитом, тобто відкладати передачу на максимально можливий термін. Обмеження знизу є гарантією збереження коректності обчислень, оскільки дані будуть підготовані перед їх пересилкою для використання, а вихід за обмеження зверху через необхідність додаткового очікування після запиту на використання даних вносить у час виконання затримку, під час якої обчислення не виконуються, що в результаті знижує коефіцієнт ефективності. Запропоновано також два підходи до більш гнучкого обчислення часу початку передачі даних, один з яких базується на статистичній обробці попередніх значень часу очікування введення та виведення та обсягів пам'яті під час розв'язання даної задачі, а інший – на ймовірнісній оцінці часу першого запиту даних та часу, необхідного для передачі цих даних між вузлами. В рамках кожного з підходів може бути запропоновано декілька різних способів визначення невідомих величин, а саме можна змінювати алгоритми статистичної обробки або ймовірнісні моделі прогнозування. Для останніх найбільш простою є використання моделі програмування, що базується на розбитті на підзадачі, яка якісно відображає задачі з великою кількістю даних для обробки.

Список посилань:

1. Стиренко, С.Г. Модель организации вычислений в распределенной системе / С.Г. Стиренко, А.И. Зиненко, Д.В. Грибенко // Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка. – 2012. –Т. 57. – С. 101–109.
2. Стіренко С.Г. Модель технології програмування паралельних систем. / С.Г. Стиренко // Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка. – 2013. –Т. 58. – С. 158–164.
3. Симоненко, В. П. Организация вычислительных процессов в ЭВМ, комплексах, сетях и системах / В. П. Симоненко. – К. : ВЕК+, 1997. –304 с.