

Технічні науки

УДК 004.051

Міхерєва Валерія Яківна

студентка

Харківського національного університету радіоелектроніки

Михерева Валерия Яковлевна

студентка

Харьковского национального университета радиоэлектроники

Mikhierieva Valeriia

Student of the

Kharkiv National University of Radioelectronics

Сергієнко Олександра Сергіївна

студентка

Харківського національного університету радіоелектроніки

Сергиенко Александра Сергеевна

студентка

Харьковского национального университета радиоэлектроники

Serhiienko Oleksandra

Student of the

Kharkiv National University of Radioelectronics

Олійник Олена Володимирівна

старший викладач кафедри програмної інженерії

Харківський національний університет радіоелектроніки

Олейник Елена Владимировна

старший преподаватель кафедры программной инженерии

Харьковский национальный университет радиоэлектроники

Oliinik Olena

Assistant Professor of Software Engineering Department

Kharkiv National University of Radioelectronics

**МЕТОДИ ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ В РОЗРОБЦІ ПРОГРАМ.
ГРАНУЛЯРНІСТЬ
МЕТОДЫ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ В РАЗРАБОТКЕ
ПРОГРАММ. ГРАНУЛЯРНОСТЬ
METHODS OF PARALLEL COMPUTING IN PROGRAM DESIGNING.
GRANULARITY**

Анотація. У теперішній час сучасних швидкісних технологій все більше приділяється увага паралельному програмуванню. У статті розглянуті деякі методи паралельних обчислень, які можуть стати у нагоді під час розробки програм, підвищити ефективність або зменшити час виконання програми, а саме балансування навантаження, синхронізація, залежність даних та гранулярність. Увагу, перш за все, акцентовано саме на гранулярності та її типах, таких як крупнозернистий, дрібнозернистий та середньозернистий паралелізм.

Ключові слова: гранулярність, паралелізм, обчислення, балансування навантаження, синхронізація, залежність даних.

Аннотация. В настоящее время современных скоростных технологий все больше внимания уделяется параллельным вычислениям. В статье рассмотрены некоторые методы параллельных вычислений, которые пригодятся во время разработки программ, повысят эффективность или уменьшат время выполнения программы, а именно балансировка нагрузки, синхронизация, зависимость данных и гранулярность. Прежде всего, внимание уделяется гранулярности и ее типам, таким как крупнозернистый, мелкозернистый или среднезернистый параллелизм.

Ключевые слова: гранулярность, параллелизм, вычисления, балансировка нагрузки, синхронизация, зависимость данных.

Summary. *At this time of modern high-speed technologies attention is increasingly focused on parallel computing. In this article some methods of parallel computing, that are useful during program designing, for instance, to improve efficiency or to reduce program execution time, are considered. Among them are load balancing, synchronization, data dependencies and granularity. First and foremost attention is paid to granularity and its types, such as coarse-grained, fine-grained and medium-grained parallelism.*

Key words: *granularity, parallelism, computing, load balancing, synchronization, data dependencies.*

Весь реальний світ працює в динамічній природі, тобто багато речей відбувається в певний час, але в різних місцях одночасно. Але цих даних занадто багато для управління, саме тому виникає потреба в паралельних обчисленнях.

Паралельні обчислення — це використання одночасно декількох елементів обробки для вирішення будь-якої проблеми [3]. Проблеми розбиті на інструкції та вирішуються одночасно, оскільки кожен ресурс, який був застосований до роботи, працює одночасно. Доречно зазначити, що паралельні обчислення використовуються в наступних сферах: бази даних та видобуток даних, моделювання систем у реальному часі, наука та техніка, розширена графіка, доповнена реальність та віртуальна реальність. Отже, розглянемо найцікавіші методи паралельного обчислення.

Балансування навантаження відноситься до практики розподілу роботи між завданнями, щоб усі завдання були постійно зайняті. Це можна вважати мінімізацією незайнятого завдання. Балансування навантаження є важливим для паралельних програм з міркувань продуктивності. Наприклад, якщо всі завдання підпадають під точку бар'єрної синхронізації, найповільніше завдання визначатиме загальну продуктивність. Досягти балансування навантаження можна завдяки

рівному розділенню роботи, яку отримує кожне завдання, та використанню динамічного робочого завдання.

Під терміном синхронізації мається на увазі координація подій для управління системою погоджено [4]. Розрізняють такі типи синхронізації, як: бар'єр (зазвичай мається на увазі, що задіяні всі завдання), локер/семафор (може включати будь-яку кількість завдань) та синхронні операції зв'язку (здійює лише ті завдання, які виконують комунікаційну операцію).

Що стосується залежності даних, то це залежність між операторами програми, коли порядок виконання операторів впливає на результати програми. Залежність даних виникає в результаті багаторазового використання одних і тих самих розташувань у сховищі різними завданнями. Вирішувати проблему залежності даних можливо із застосуванням наступних технік: архітектури розподіленої пам'яті (передають необхідні дані в точках синхронізації) та архітектури спільної пам'яті (синхронізують операції читання/запису між завданнями).

У паралельних обчисленнях гранулярність завдання визначають як міру обсягу роботи (або обчислення), яку виконує це завдання. При паралельних обчисленнях гранулярність є якісним показником відношення обчислень до зв'язку. Періоди обчислень зазвичай відокремлюються від періодів комунікації подіями синхронізації. Тепер розглянемо більш детально типи гранулярності.

У дрібнозернистому паралелізмі програма розбивається на велику кількість дрібних завдань. Ці завдання покладаються на багато процесорів індивідуально. Обсяг роботи, пов'язаної з паралельним завданням, низький, і робота рівномірно розподіляється між процесорами [2]. Характерними рисами для дрібнозернистого паралелізму є порівняно невеликі обчислювальні роботи, низький коефіцієнт обчислення до зв'язку,

полегшення балансування навантаження, високі накладні витрати на зв'язок і менші можливості для підвищення продуктивності.

Що стосується крупнозернистого паралелізму, то тут відносно великі обчислювальні роботи виконуються між подіями зв'язку / синхронізації. Відрізняється високим коефіцієнтом обчислення до зв'язку та має більше можливостей для підвищення продуктивності [1].

Середньозернистий паралелізм є дещо компромісом між дрібнозернистим і крупнозернистим паралелізмом, де ми маємо розмір завдань і час спілкування більший за дрібнозернистий паралелізм і менший за крупнозернистий паралелізм. Більшість паралельних комп'ютерів загального призначення належать саме до цієї категорії.

Найефективніша гранулярність залежить від алгоритму та апаратного середовища, в якому він працює.

Література

1. Blaise B. Introduction to Parallel Computing / Barney Blaise., 2006. 187 с.
2. Hwang K. Advanced Computer Architecture: Parallelism, Scalability, Programmability / Kai Hwang., 1992. 771 с.
3. Kaeli D. Heterogeneous Computing with OpenCL 2.0 / D. Kaeli, P. Mistry., 2015. 307 с.
4. Synchronization. URL:
<https://en.wikipedia.org/wiki/Synchronization#:~:text=Synchronization%20is%20the%20coordination%20of,that%20are%20not%20are%20asynchronous...>