

ДАВИДЕНКО І. М.
КРАВЧЕНКО С. О.
БАБІЙ К. Ю.

ДЕЦЕНТРАЛІЗОВАНА МОДЕЛЬ КЕРУВАННЯ GRID-СИСТЕМОЮ

In this work the method of decentralized *GRID* model management by using the access method which based on *P2P* (*Peer-to-Peer*) technology for represents a promising platform for scalable demanding tasks is offered.

У роботі розглянуто спосіб організації децентралізованої моделі управління *GRID*-системою за рахунок використання методу доступу до ресурсів на основі технології *P2P*, яка представляє собою перспективну платформу для використання крупномасштабованих ресурсномістких задач.

Вступ

Сучасний розвиток обчислювальної техніки знаходиться в переломній стадії, коли окремі обчислювальні вузли через технологічні обмеження вже не в змозі обробляти поставлені завдання в необхідному обсязі із заданим рівнем якості обслуговування *QoS* (*Quality of Service*).

В таких умовах єдиною альтернативою нарощування потужності системи є об'єднання множини обчислювальних вузлів у розподілені *GRID*-системи, які включають не тільки інструменти організації обчислень, але й інструменти управління ресурсами зберігання даних, забезпечення безпеки доступу до даних, моніторингу використання та переміщення даних із необхідним рівнем *QoS* [1]. Важливо зауважити, що основною перевагою таких систем є можливість не тільки паралельної віддаленої обробки даних, але й можливість одночасного доступу до цих ресурсів віддаленим користувачам із заданим рівнем *QoS*.

Традиційно пошук ресурсів в *GRID*-системах в основному базується на централізованій моделі керування. Ця модель керування складно масштабується, що може стати потенційною проблемою продуктивності та безпеки *GRID*-систем. Таким чином, головною метою *GRID*-систем є можливість забезпечення користувачів обчислювальною потужністю, необхідною для обробки завдань за допомогою географічно розподілених ресурсів з заданим рівнем *QoS*.

1. Огляд та аналіз існуючих рішень

Під *GRID*-системою розуміється географічно розподілена інфраструктура, яка об'єднує множину ресурсів різних типів, до-

ступ до яких користувач може отримати із будь-якої точки, незалежно від місця їх розташування [2]. Основною метою *GRID*-систем є об'єднання гетерогенних, широкомасштабних та багатоцільових ресурсів для надання прозорого, надійного та скоординованого доступу до обчислюваних ресурсів, які знаходяться на великій відстані.

У зв'язку з тим, що *GRID*-система являє собою логічне налаштування над розподіленою обчислювальною системою, однією з найважливіших її частин є система управління ресурсами, що виконує завдання виявлення, розміщення, відстеження та управління мережевими ресурсами для забезпечення різних вузлових і глобальних параметрів *QoS* [3]. Крім того зауважимо, що ресурси *GRID*-системи належать різним класам, які визначаються низкою атрибутів та характеристик [4]. У кожного ресурсу є певне значення для кожного атрибуту, визначеного відповідним класом ресурсу. Вимоги до системи управління ресурсами *GRID*-системи, роблять децентралізоване управління складним і не маючим однозначного рішення.

Найбільш розповсюдженим методом децентралізованого управління ресурсами є технологія *P2P*, яка стає перспективною платформою для виконання великомасштабних ресурсномістких задач. Підхід *P2P* має ряд особливостей, які дають можливість забезпечити управління великою кількістю вузлів із заданим рівнем *QoS*, дії яких ґрунтуються на локальній інформації, що зберігається в кожному вузлі. Крім того системи, засновані на такому підході, легко масштабуються.

Для управління ресурсами та самоорганізації великомасштабованих і розподілених *GRID*-систем, найчастіше використовують

оптимізований мурашковий алгоритм *ACO* (*Ant Colony Optimization*). В роботі [5] запропоновано метод подібний до алгоритму самоорганізації мурашок, який дозволяє рівномірно розподіляти роботу серед доступних ресурсів.

Для гетерогенного обчислювального середовища запропоновано та спроектовано гібридний *ACO*, який балансує навантаження на ресурси *GRID*-системи. У зв'язку з тим, що ресурси *GRID*-системи належать різним класам (інформаційний, обчислювальний та ресурс зберігання даних), вони вимагають великої кількості параметрів, які характеризують клас саме цього ресурсу.

2. Постановка задачі

У зв'язку з тим, що більшість існуючих систем управління *GRID* архітектурою мають централізоване управління із складним процесом налаштування та обслуговування, актуальною є проблема розробки нових децентралізованих підходів для вирішення завдань надання доступу до ресурсів *GRID*-системи, для груп віддалених користувачів із заданим рівнем *QoS*.

Тому необхідно запропонувати спосіб організації децентралізованої моделі управління *GRID*-системами, який гарантує надійну доставку інформації. Проте, щоб гарантувати доступ до даних з вимогами *QoS*, які пред'являються до спільних обчислень, необхідна більш проста структура, яка буде використовувати метод доступу до даних заснований на концепції *P2P*-систем. В даний час існує ряд рішень цієї задачі, але залишаються відкритими ще такі питання, як: швидке та динамічне виявлення та резервування обчислювальних ресурсів; визначення і пошук вільних ресурсів; зменшення часу аналізу і планування ресурсів (затримок); динамічне виявлення та аналіз ресурсів; вирішення задач для зменшення затримок з урахуванням балансування завантаження; розподіл мережних ресурсів між задачами.

3. Рішення поставленої задачі

У роботі запропоновано підхід організації децентралізованої моделі управління *GRID*-системою за рахунок використання методу доступу до ресурсів за технологією *P2P*. Однак, для забезпечення гарантованого доступу до ресурсів із заданим *QoS* необхідна більш проста структура. Для цього у роботі запро-

поновано підхід організації системи моніторингу та виявлення *MDS* (*Monitoring and Discovery System*) ресурсів *GRID*-системи. Розроблена *MDS* працює на основі архітектури *P2P*, на відміну від традиційної ієрархічної, за рахунок використання розподіленої системи мурашкових колоній *DACS* (*Distributed Ant Colony System*) [6] для виявлення необхідних ресурсів, зареєстрованих у *MDS*. У основу алгоритму покладено поведінку мурашкових колоній, де агент *MDS* керує множиною запитів на ресурс із маркування найбільш вдалим шляхів більшим значенням вагового коефіцієнту.

Нехай *GRID*-система представляє собою граф $G(V,E)$, який складається з n вузлів та m ліній зв'язку $e(i, j)$, кожна з яких з'єднує два вузли i, j (рис. 1).

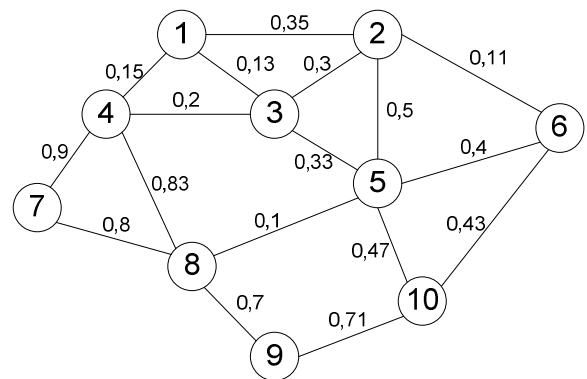


Рис.1. Приклад *GRID*-системи із заданими ваговими коефіцієнтами пошуку k -го ресурсу

На першому етапі у вершинах графа розміщуються запити на ресурс, після чого починається їх направлене розповсюдження по графу згідно алгоритму, наведеному нижче.

Мітка запиту на ресурс розташована у вершині V_i і визначає тип необхідних ресурсів k , і в разі їх виявлення відправляє вузлу-відправнику. Якщо ж ресурс не виявлено, мітка запиту на ресурс передається до наступної вершини V_n , при цьому для переходу обчислюється ймовірність переходу на кожну з сусідніх вершин з використанням вагового коефіцієнта кожної лінії зв'язку $e(i,j)$ для k -го типу ресурсу. Ця ймовірність обчислюється за формулою:

$$P_{ij}^k = \frac{\tau_{ij}^k}{\sum_q \tau_{iq}^k}, \quad q \in J_i, \quad (1)$$

де J_i – представляє сусідів вершини, τ_{iq}^k – значення вагового коефіцієнта відповідного зв'язку з сусідньою вершиною.

Коли на деякому переході мітка запиту буде виявлено потрібний ресурс, мітка повертається до вершини-відправника запиту. Наступним кроком є оновлення значення вагового коефіцієнта для кожного зв'язку на знайденому шляху, що визначається за функцією:

$$\tau_{ij}^k = (1 - \rho)\tau_{ij}^k + \rho\Delta_{ij}^{best}, \quad \rho \in (0,1],$$

$$\Delta_{ij}^{best} = \frac{1}{L_{best}}, \quad (2)$$

де ρ – коефіцієнт зміни значення вагового коефіцієнта, Δ_{ij}^{best} – ваговий коефіцієнт, що встановлюється по шляху проходження кращою міткою запиту, L_{best} – довжина найкращого шляху проходження для заданої позначки.

Нехай мітка запиту на ресурс розташована у вершині V_5 , а ресурс k -го типу, що потребується згідно запиту, знаходиться у вершині V_7 . За формулою (1) проведемо розрахунки ймовірностей переходів. Оскільки ми розраховуємо ймовірності переходів, то шлях може бути обрано випадковим чином із поступовим збільшенням ймовірності знаходження найоптимальнішого шляху. В даному прикладі прокладемо шлях переходами з найбільшими значеннями ймовірності. Отримані результати наведено у табл. 1.

Табл. 1. Шлях мітки запиту зі значеннями ймовірності вибору наступного переходу

V_5	P_{ij}^k	V_2	P_{ij}^k	V_1	P_{ij}^k	V_4	P_{ij}^k
V_3	0.18	V_1	0.46	V_2	0	V_1	0
V_2	0.28	V_3	0.4	V_3	0.46	V_3	0.1
V_6	0.22	V_5	0	V_4	0.54	V_7	0.47
V_8	0.06	V_6	0.14			V_8	0.43
V_{10}	0.26						

Зверніть увагу, що ймовірність переходу на вершину, з якої прийшла мітка, встановлюється рівною 0, оскільки в цій вершині мітка вже була і розшукуваний ресурс там знайдено не було.

Після того, як мітка запиту, що згенеровано, повернеться до відправника запиту V_5 , на основі довжини найліпшого шляху L_{best} та ймовірності вибору мітки запиту P_{ij}^k по формулі (2) визначаються оновлені значення вагового коефіцієнта для кожного зв'язку на знайденому шляху.

У даному випадку, це будуть зв'язки за маршрутом, який проходить через вершини: $V_5 - V_2 - V_1 - V_4 - V_7$.

У роботі запропоновано рішення задачі виявлення ресурсів у *GRID*-системі, перевагою якого є можливість децентралізованого пошуку ресурсів в *GRID*-системі і простоті реалізації.

4. Моделювання

В рамках роботи проведено моделювання запропонованого алгоритму *MY*, *DACS* і алгоритму *AS*, який є одним з існуючих рішень, що також заснований на алгоритмі мурашиних колоній (рис.2).

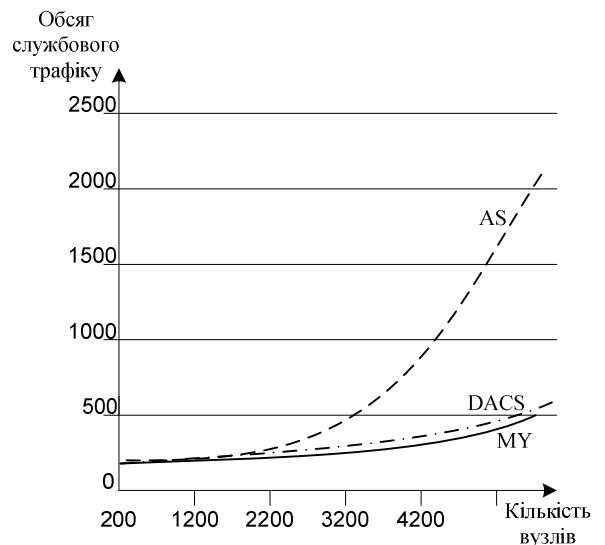


Рис. 2. Порівняння часових характеристик запропонованого алгоритму MY із DACS та AS.

За критерієм обсягу службового трафіку запропонований алгоритм *MY* показав найкращі результати, але не значним зменшення службового трафіку по відношенню до алгоритмів *DACS* та *AS*. Істотне зменшення стає при загальній кількості вершин більше 3200.

Висновки

Розроблена *MDS* для розв'язання задач пошуку ресурсів в *GRID*-системі має ряд переваг, основними серед яких є можливість децентралізованого управління системою. Розроблений спосіб організації управління *GRID*-системою у повній мірі відповідає всім необхідним вимогам динамічного середовища *GRID* та має властивості децентралізованого виконання, що збільшує швидкість його роботи та надійність, зменшує обсяг службового трафіку та надає можливість функціонування у динамічному середовищі.

Список літератури

1. J. Coomer, C. Chaubal, Introduction to the Cluster Grid. – Grid Computing, Sun BluePrints, 2002. – P. 416.
2. I. Foster, C. Kesselman, S. Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. – International Journal Supercomputer Applications, 2001. – Vol.15(3). – P.315 – 327.
3. J. M. Pullen, R. Brunton, D. Brutzman, D. Drake, M. Hieb, K. Morse, A. Tolk, Using web services to integrate heterogeneous simulations in a grid environment. – Future Generation Computer Systems, 2005. – Vol. 21. – P.97–106.
4. J. Luo, P. Ji, X. Wang, Y. Zhu, F. Li, T. Ma, X. Wang, Resource management and task scheduling in grid computing. – Computer Supported Cooperative Work in Design, 2004. – Vol. 2. –P. 431–436.
5. C. Blum, Beam-ACO – Hybridizing ant colony optimization with beam search: an application to open shop scheduling. – Comput. Oper. – 2005. – Vol. 32. – P.1565–1591.
6. S. Fattahi, N. Charkari, Distributed ACS Algorithm for Resource Discovery in Grid, 2009. – P.37. –45.