

В. М. Дубовой, д-р. техн. наук, проф.; О. М. Москвін, асп.

ІНТЕЛЕКТУАЛЬНА МУЛЬТИАГЕНТНА ТЕХНОЛОГІЯ ОПТИМІЗАЦІЇ ІНТЕРНЕТ-РЕСУРСІВ

Запропоновано концептуальну модель, архітектуру, методика декомпозиції задач і засіб формалізації процесів взаємодії компонентів мультиагентної системи оптимізації Інтернет-ресурсів, основною задачею якої є аналіз слабоструктурованих даних великої розмірності та оптимізація структури зв'язків між їхніми інформаційними доменами.

Вступ

Завдяки масовій комп'ютеризації та поширенню нових інформаційних технологій Інтернет є одним з найважливіших джерел інформації. Велика кількість веб-сайтів, їх відносно низька якість, що виражена в хаотичній, неоптимізованій структурі, сповільнюють пошук необхідної інформації, яка зазвичай представлена у вигляді гіпертексту. Тому проблема оптимізації гіпертекстових систем є актуальною.

Для вирішення проблеми в роботах Т. Bernes-Lee, J. Hendler, O. Lassila, запропонована концепція семантичного павутиння (Semantic Web) [1] – частина глобальної концепції розвитку мережі Інтернет, метою якої є реалізація можливості машинної обробки інформації. Основний акцент концепції ставиться на роботі з метаданими, які однозначно характеризують властивості і зміст ресурсів Інтернет, замість використовуваного в наш час текстового аналізу документів. В основі другого підходу, розглянутого в роботах R. A. Botafogo, E. Rivlin, B. Shneiderman [2], Г. Е. Кедрова [3], лежить використання певної гіпертекстової метрики, за допомогою якої можна оцінити гіпертекстову систему певними формальними показниками.

Існуючі підходи до оптимізації гіпертекстових структур характеризуються надвеликою розмірністю даних, а отже потребують для розв'язання задачі оптимізації у зосередженому операційному середовищі надто багато часу.

В зв'язку із зазначеною складністю в роботі розглядається *задача* побудови архітектури системи розподіленого обчислення для системи оптимізації Інтернет-ресурсів.

На сьогодні існує низка стандартних підходів до розв'язання зазначених задач з обробки великих обсягів даних, що основані на паралельній обробці множиною обчислювальних вузлів, серед яких слід відмітити Map-Reduce [4]. Цей підхід характеризується чітко регламентованим алгоритмом, основаним на операціях декомпозиції вхідних даних для незалежної обробки множиною обчислювальних вузлів системи та об'єднання результату розподіленого виконання задачі. Інші засоби для здійснення розподілених обчислень та методи, що лежать в їх основі, такі як Dyuad [5], є більш спеціалізованими з точки зору типу виконуваних операцій чи мають обмеження на формат вхідних даних та середовище функціонування.

Сам по собі Map-Reduce не регламентує механізмів відновлення системи у разі виникнення збоїв, координування процесів виконання завдань та загальну топологію розподіленої обчислювальної системи. Тому нерегламентовані особливості Map-Reduce реалізуються в кожній конкретній системі розподіленої обробки даних по-різному. Але відсутність певних стандартизованих практик з розв'язання зазначених задач керування обчислювальним процесом ускладнює та обмежує впровадження систем такого роду через необхідність самостійної реалізації стека координації виконанням задач, оцінки ефективності власної реалізації та формалізації процесів обміну інформації між вузлами системи.

Метою статті є побудова інтелектуальної розподіленої обчислювальної системи оптимізації Інтернет-ресурсів, включаючи підсистему координації обчислювальних вузлів на основі мультиагентного підходу та специфікацію засобів формального опису взаємодії її підсистем.

Концептуальна модель системи

В системі розподілених обчислень основними складовими є метод розподілу завдань і підсистема координації та контролю їх виконання. Розподіл завдань за методом Map-Reduce є надійним шляхом до розв'язання цільової задачі великої розмірності, що зумовлено його загальною універсальністю. Тому, зокрема, в цій роботі для оптимізації Інтернет-ресурсів еталонном обрано саме Map-Reduce. З архітектурної точки зору цей підхід є найгнучкішим, оскільки може бути пристосований майже до будь-якої задачі та є незалежним від певного операційного середовища.

В системах зазначеного типу основні цілі зводяться до досягнення більшої швидкості проведення обчислювальних процесів та ефективного використання наявних ресурсів обчислювальної інфраструктури. Звичайно, ці задачі пов'язані, тому повинні розв'язуватись одночасно. Для ефективного їх розв'язання система має реалізовувати певну адаптивну поведінку, розподіляючи обчислювальне навантаження в залежності від стану всієї системи загалом та відповідно до встановлених цілей, досягнення яких має характеризувати кожен її компоненту. Визначені вимоги до підсистеми координації та контролю виконання завдань узгоджуються з концепцією мультиагентних систем, які за своєю природою розподілені, функціонують в гетерогенних середовищах, характеризуються кооперативною взаємодією для розв'язання поставлених задач, можливістю до самоорганізації відповідно до визначених цілей, надійністю та стійкістю до збоїв. Крім того, певні архітектурні рішення, побудовані на основі мультиагентних систем, стандартизовані і сертифіковані для промислового застосування, наприклад, FIPA [6], а отже загальновідомі та перевірені на практиці. Тому в цій роботі для керування координацією та контролю за виконанням завдань обрано концепцію мультиагентних систем.

Таким чином, запропонована концептуальна модель розподіленої системи обчислень використовує метод Map-Reduce для розв'язання обчислювальних задач великої розмірності в розподіленому середовищі під керуванням системи, побудованої на основі мультиагентного підходу.

Функціональна модель мультиагентної системи

Система, що розглядається, складається із підсистеми оптимізації досяжності інформації в Інтернет-ресурсах, яка за умов використання у послідовному, нерозподіленому операційному середовищі потребує багато часу, та підсистеми розподіленого обчислення, що покликана цей час зменшити. До підсистеми оптимізації надходять запити від користувачів на оптимізацію певних сегментів гіпертекстової мережі, формуючи її чергу запитів. Основними задачами підсистеми оптимізації Інтернет-ресурсів є:

- структурний та семантичний аналіз ресурсів вказаного сегменту мережі;
- аналіз діяльності користувачів у вказаному сегменті гіпертекстової мережі;
- побудова оптимальних гіпертекстових зв'язків на основі цілей користувачів;
- побудова оптимальних гіпертекстових зв'язків на основі семантичного аналізу.

Основними вимогами, що висувуються до підсистеми розподіленого обчислення, є:

- рівномірне завантаження задіяних обчислювальних систем;
- забезпечення заданої швидкості обробки запитів;
- стійкість до відмов, автоматичне відновлення у разі відмов;
- адаптивна, кооперативна поведінка компонентів, спрямована на досягнення цілей системи.

Виходячи із задач та вимог до розподіленої системи оптимізації Інтернет-ресурсів, визначимо її функціональну модель.

Сукупність агентів системи розподіленого обчислення, побудованої на основі мультиагентного підходу, що пропонується, є середовищем для виконання обчислювальних задач. Основні операції в цій системі виконує інтелектуальний агент, побудований згідно з критеріями інтелектуальності, визначеними у [7]. Кількість типів агентів визначається кількістю незалежних операцій, для виконання яких передбачається використання такої мультиагентної системи.

Інтелектуальна складова агентів у системі полягає у їх адаптивній поведінці в процесі розв'язання задачі, яка в залежності від стану зовнішнього середовища може бути частково або повністю делегована іншим агентам цього типу. Слід зазначити, що архітектура агента

підтримує можливість поетапного зниження алгоритмічної складності обчислювальної задачі і залежить від рівня функціональної декомпозиції, попередньо виконаної програмістом, що дозволяє ефективніше здійснювати її виконання.

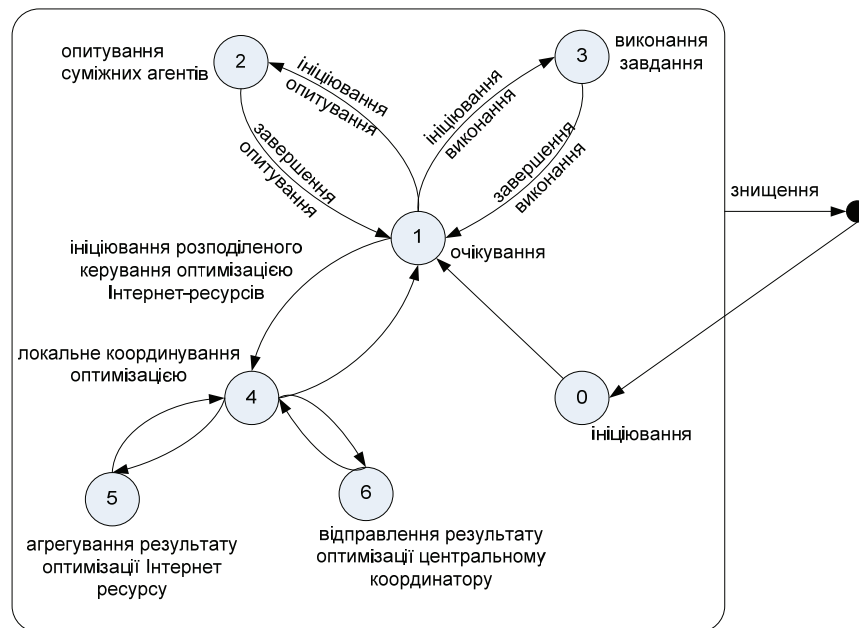


Рис. 1. Діаграма станів інтелектуального агента мультиагентної системи

Для реалізації агента описаним вище способом його поведінка повинна характеризуватись кооперативною взаємодією, попереджувальним плануванням, що спрямовані на досягнення агентом встановлених зобов'язань. Особливості поведінки, згідно з [7], відповідають загальним вимогам до концептуальної моделі інтелектуальних агентів (рис. 1). Для забезпечення зазначених особливостей архітектура інтелектуального агента мультиагентної системи оптимізації гіпертекстових ресурсів, що пропонується, має вигляд, показаний на рис. 2.

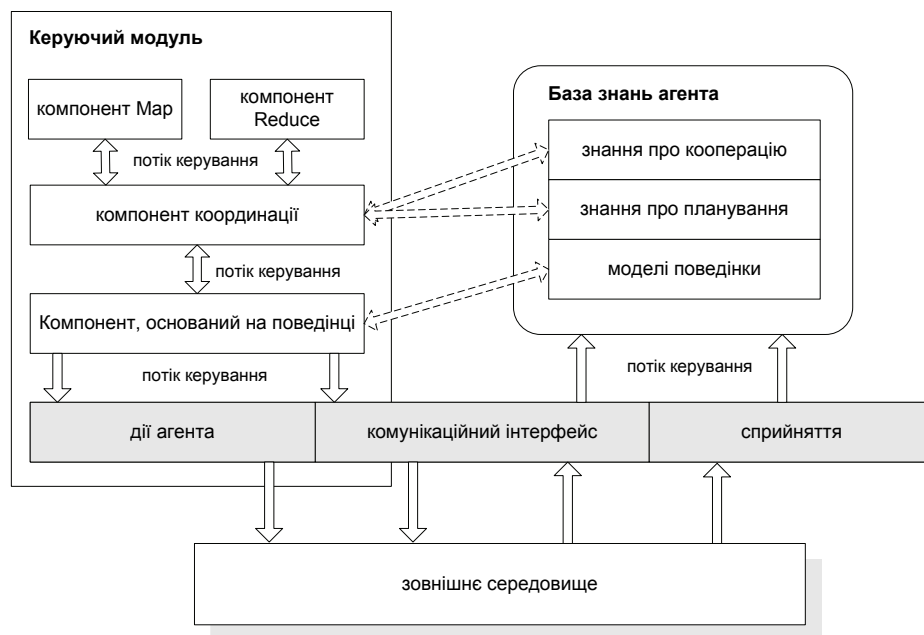


Рис. 2. Архітектура інтелектуального агента мультиагентної системи

Виходячи із структури, архітектури та вимог задачі, що розв'язується, основними функціями інтелектуального агента системи оптимізації Інтернет-ресурсів є:

— кооперація: кожен агент може бути локальним координатором задачі, що надходить із зовнішнього середовища. Задача в цьому випадку розглядається як множина незалежних підзадач $T = (t_1, t_2, \dots, t_n)$, що можуть бути виконані різними агентами цього типу;

— планування: агент здійснює попереджувальний аналіз продуктивності інших агентів цього типу, які приймають участь у обчислювальному процесі, та делегування підзадач відповідно до їх завантаженості;

— цілеспрямована поведінка: в залежності від стану навколишнього середовища, наявності вільних ресурсів, керування виконанням завдань відповідно до встановлених зобов'язань.

Зовнішнім середовищем для агента є інші агенти цього типу, центральний координатор задач мультиагентної системи та центральна база даних, що агрегує результати роботи всіх агентів системи. Функціями центрального координатора в цій системі є:

- ведення загального реєстру агентів;
- аналіз продуктивності мультиагентної системи загалом;
- контроль за загальним станом виконання задач;
- авторизація та реєстрація нових агентів системи;
- перевизначення локальних координаторів у випадку аварійної зупинки їх діяльності.

Таким чином, загальна структура системи буде виглядати відповідно до схеми на рис. 3.

Розглянемо підсистему управління агентами мультиагентної системи оптимізації Інтернет-ресурсів. Агентна платформа (АП) являє собою інфраструктуру, в якій можуть бути розміщені агенти. АП складається із обчислювальних систем, операційних систем, компонентів управління агентами і самих агентів. З точки зору АП кожний агент:

— поєднує одну або більше сервісних можливостей в інтегрованій моделі виконання;

— має супервізора (компонент системи, що делегує завдання);

— характеризується ідентифікатором, що дозволяє однозначно виділити його із загальної множини агентів.

Центральний координатор поєднує в собі три підсистеми — загальну чергу завдань, що надаються із зовнішнього середовища, систему управління агентами та службу каталогів. Центральна база даних та система її керування призначені для збереження результатів роботи груп агентів, характеризується відповідністю до принципу ACID.

Система управління агентами (СУА) — обов'язковий компонент АП, що існує в єдиному екземплярі та характеризується такими функціями:

- керує доступом до агентної платформи та її використанням;
- підтримує каталог ідентифікаторів агентів та їхніх транспортних адрес;
- реєструє агентів та здійснює видачу їм ідентифікаторів.

Служби каталогів (СК) також є невід'ємною складовою АП, за допомогою яких агенти реєструють свої сервіси і здійснюють пошук з метою знаходження агентів, що надають задані сервіси. Система моніторингу стану системи реалізує поведінку на відновлення системи у разі відмов та збоїв.

Структура АП, що відображає вищенаведені засади функціонування, показана на рис. 4.

Отже, основними складовими архітектури мультиагентної системи керування розподіленим обчислювальним процесом, що пропонується, є множина агентів, агентна платформа, централі-

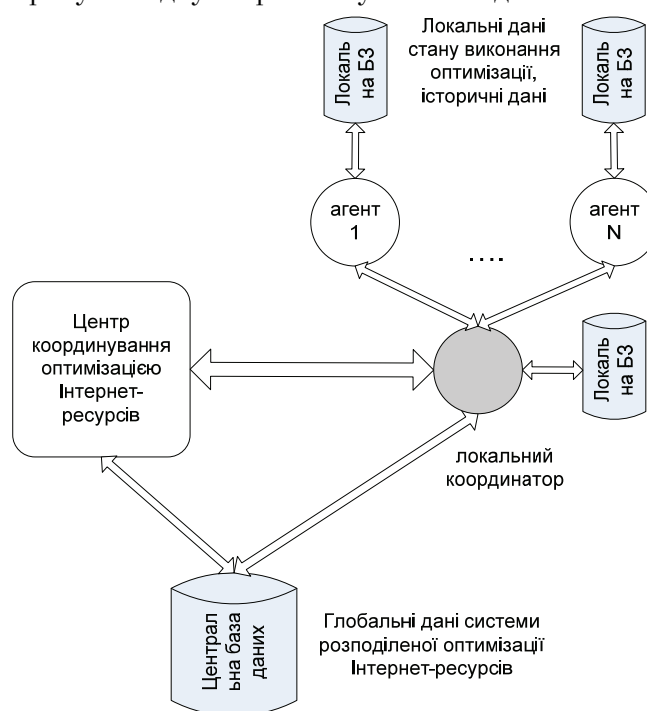


Рис. 3. Загальна структура мультиагентної систем

зована база даних для зберігання результатів функціонування агентів та центральний координатор. Основними функціями агентів є взаємодія із зовнішнім середовищем для кооперації із іншими агентами та планування і безпосереднє виконання задач системи.

Основними функціями АП є транспортна, для забезпечення середовища взаємодії агентів, контролююча, для загального моніторингу виконання завдань, та облікова, для загальної реєстрації та обліку активних агентів.

Формалізація процесів взаємодії

В зв'язку із наявністю різних за типом та призначенням агентів важливим етапом у побудові й моделюванні розподіленої системи обчислень є формалізація процесів взаємодії між її компонентами, що дозволяє чітко визначати потоки даних, ідентифікувати вхідні та вихідні дані. В цій роботі пропонується використати пі-числення (pi-calculus) для аналізу та формалізації процесів в розподіленій системі оптимізації Інтернет-ресурсів. Основна процедура полягає в передачі даних і каналів зв'язку між агентами, де одержувач може використовувати їх для подальшої взаємодії з іншими учасниками комунікаційних процесів. Саме ця особливість пі-числення робить його доцільним для моделювання мультиагентних систем, де доступні ресурси змінюються у часі.

Використовуючи нотації та оператори, визначені у [9], опишемо основні операції взаємодії агентів, де малими літерами позначено дані та канали, великими — агенти.

Локальна координація агентом обчислювального процесу

$$X = (\nu l)l(m). \bar{p}\langle \rangle | p(s_1, s_2, \dots, s_n). \bar{p}\langle m_1, m_2, \dots, m_k, r \rangle | r(r_1, r_2, \dots, r_n). \bar{z}\langle result \rangle,$$

де агент X очікує завдання t по каналу l . На канал l встановлене обмеження виключності доступу. Отримуючи повідомлення m , агент здійснює опитування суміжних агентів шляхом відправки повідомлення у канал p , де паралельний процес агента X очікує на відповіді вільних агентів, що можуть виконати завдання, після чого агент X здійснює декомпозицію завдання e на частини m_1, m_2, \dots, m_k та відправляє їх агентам s_1, \dots, s_n разом із каналом, на якому агент X буде очікувати на результат. Паралельний процес отримує відповіді по каналу r , об'єднує результат та відправляє центральному координатору по каналу зв'язку z .

Агенти, що виконують моніторинг зовнішніх об'єктів, стан яких змінюється:

$$Y = (n(x_1, x_2, \dots, x_n) | d(y_1, y_2, \dots, y_n)). ([(x_1, x_2, \dots, x_n) \neq (y_1, y_2, \dots, y_n)] \bar{q}(x_1, x_2, \dots, x_n)),$$

де n — канал, по якому агент Y отримує дані із зовнішнього середовища; d — канал, по якому агент Y отримує відповідні дані із локальної бази даних. У випадку їх незбігу агент відправляє повідомлення-запит на здійснення оновлення інформації по відповідному ресурсу по каналу q в чергу МАС.

Глобальний координатор можна описати так:

$$Z = q(t). l\langle m \rangle,$$

що фактично перенаправляє запити із черги q по внутрішньому каналу в систему транспортування повідомлень l .

Звичайного агента, що виконує завдання, можна описати таким чином:

$$E = (\nu l)l(e, x). \varepsilon_t. x\langle m \rangle,$$

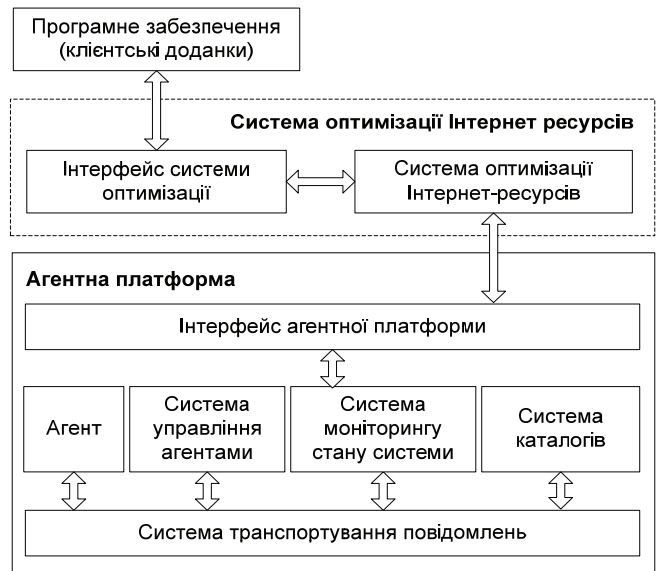


Рис. 4. Структура агентної платформи

де l — канал, по якому агент отримує завдання t , та канал x , яким необхідно буде відправити результат виконання; ε_t — визначення результату e .

Розпаралелювання задач оптимізації Інтернет-ресурсів

Виходячи із запропонованої архітектури агента та мультиагентної системи, характером взаємодії її компонентів, кожен агент характеризується можливістю розв'язання як певної підзадачі, так і керування процесом її виконання шляхом делегування. Запропонований підхід до декомпозиції і оцінки складності ресурсоемних задач передбачає використання графу залежності задач (task dependency graph) [12]. Для оцінки прискорення процесу виконання за рахунок розбиття обчислювального алгоритму на незалежні блоки і виконання їх незалежними агентами пропонується використання закону Амдала [10]. У разі відкидання витрат на комунікацію між агентами прискорення від розподіленого виконання будемо визначати як

$$S = \frac{1}{\alpha + \frac{1-\alpha}{p}}, \quad (1)$$

де α — частка обчислень, що виконуються послідовно; $1-\alpha$ — частка обчислень, що може бути розпаралелена ідеально; p — кількість задіяних вузлів.

Нехай маємо алгоритм, який можна розпаралелити таким чином, що кожна незалежна паралельна гілка виконання оперує незалежною часткою вхідних даних, виконуючи над ними дві послідовні залежні операції. Відповідно до методики побудови графу залежності задач визначимо залежні та незалежні операції. Нехай A — перша операція, що виконується у кожній паралельній гілці, B — відповідно друга, що залежить від A і не може бути виконана із A паралельно. C — об'єднаний результат виконання паралельних гілок. Граф залежності для цього випадку показано на рис. 5.

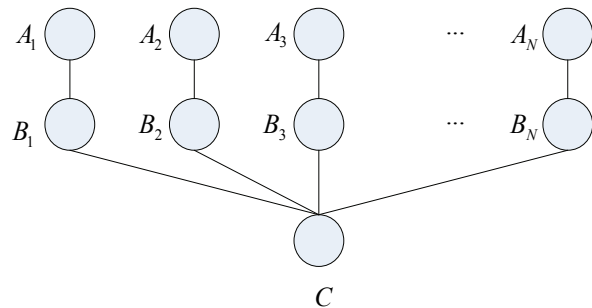


Рис. 5. Граф залежності задач для операцій побудови формального контексту

Загальна кількість компонентів в цій задачі $2N+1$. Частка $\alpha = 1/(2N+1)$ від загального об'єму обчислень може бути отримана лише послідовними розрахунками, а відповідно $1-\alpha$ може бути розпаралелена. Кількість обчислювальних систем, на яких виконується процес, $p = N+1$. Тоді прискорення від застосування мультиагентного підходу становить

$$S = \frac{1}{1/(2N+1) + \frac{1-1/(2N+1)}{N+1}}$$

Передбачається, що оскільки операції A_i та B_i залежні і повинні бути виконані у певній послідовності, то їх виконання доцільно проводити одним і тим самим агентом для мінімізації міжагентної взаємодії.

Запропонований підхід до розбиття задачі та оцінки прискорення її виконання в розподіленій системі виконання дозволяє проводити аналітичні розрахунки ефективності тієї чи іншої схеми розбиття задачі та планувати потреби у обчислювальній потужності системи в цілому.

Розподіл завдань у мультиагентній системі

Особливістю розподіленої системи обчислення для системи оптимізації Інтернет-ресурсів є те, що обчислювальне навантаження не є постійною величиною і змінюється в залежності від вхідного потоку запитів, що ініціюються користувачами або певними автоматизованими системами. Загальними вимогами, що висувуються до будь-якої мультиагентної системи, відповідно до [8], є гарантоване виконання завдання із швидкістю, що не менша за певне зазда-

легідь встановлене значення. Оскільки архітектура системи передбачає можливість динамічної зміни кількості обчислювальних ресурсів і відповідно агентів, що їх використовують для досягнення своїх цілей, постає необхідність в певний момент часу визначати кількість обчислювальних ресурсів, які необхідно залучити для обслуговування потоку запитів із регламентованою швидкістю. Визначимо параметри мультиагентної системи, від яких залежить задоволення вказаних вимог, застосовуючи апарат теорії масового обслуговування. Будемо здійснювати оцінку топології мультиагентної системи (МАС), використовуючи час реакції МАС $T_{R_{MAC}}$ – час, протягом якого МАС виконує запит та повертає результат. Час реакції $T_{R_{MAC}}$ залежить від продуктивності агентів та каналів зв'язку між ними, кількості агентів, які обробляють запит, та ступеня їх завантаженості, тобто

$$T_{R_{MAC}} = T(Q_1, Q_2, Q_3, \dots, Q_N, \lambda, \tau_{1,2}, \tau_{2,3}, \tau_{N-1,N}),$$

де $Q_1, Q_2, Q_3, \dots, Q_N$ – продуктивність агентів; N – кількість агентів, які задіяні в обробці запиту користувача; λ – інтенсивність потоку запитів; $\tau_{i,i+1}$ – час передачі повідомлення від i -го агента до $i+1$ -го. Нехай цій МАС відповідає модель системи масового обслуговування типу $G/G/1$ [11]. Час, необхідний для виконання одного запиту у випадку послідовного виконання, можна визначити за формулою

$$\tau_e = \sum_{i=1}^N t(Q_i) + \sum_{i=1}^{N-1} \tau_{i,i+1}.$$

У випадку паралельного виконання запиту множиною агентів час, необхідний для виконання запиту, можна визначити як

$$\tau_e = \frac{1}{S_e} \left(\sum_{i=1}^N t(Q_i) + \sum_{i=1}^{N-1} \tau_{i,i+1} \right),$$

де S_e , відповідно до (1), – прискорення процесу виконання певної задачі за рахунок застосування розпаралелювання.

Відповідно, інтенсивність обслуговування МАС в цьому випадку визначається як

$$\mu = \frac{1}{\tau_e}.$$

При цьому час виконання запиту, що відповідає вимогам системи до гарантованого виконання завдання із певною швидкістю, для потоку з інтенсивністю λ можна визначити за формулою

$$\tau_e < \frac{1}{\lambda}.$$

Тоді час реакції МАС буде визначатися

$$T_{R_{MAC}} = \tau_e = \sum_{i=1}^N \tau(Q_i) + \sum_{i=1}^{N-1} \tau_{i,i+1}.$$

Відповідно до формули Літла [11] середня кількість запитів у системі дорівнює

$$\bar{N} = \frac{\rho}{1-\rho}, \quad (2)$$

де ρ – пропускна здатність ($\rho = \lambda / \mu$, $0 \leq \rho < 1$). Виконуючи підстановку виразу ρ у (1), отримаємо:

$$\bar{N} = \frac{\lambda \tau_e}{1 - \lambda \tau_e}.$$

Отже, в МАС буде виникати черга запитів, довжина якої визначається як

$$L = \begin{cases} \frac{\lambda\tau_e}{1-\lambda\tau_e}, & \lambda < \frac{1}{\tau_e}; \\ \theta, & \lambda \geq \frac{1}{\tau_e}, \end{cases}$$

де $\theta \rightarrow \infty$.

Враховуючи величину черги, час реакції МАС буде

$$T_{R_{MAC}} = L(\tau_e + \tau_q), \quad (3)$$

де τ_q – час очікування заявки у черзі.

За умови $\lambda < \mu$ МАС функціонує передбаченим шляхом. У протилежному випадку, $\lambda \gg \mu$, час, необхідний для обслуговування усіх запитів у черзі, не є визначеним, що суперечить вимогам до функціонування МАС, наведеним вище. Для вирішення такої невідповідності пропонується метод побудови МАС, за якого кожен її елемент має багатократне дублювання, що дозволить здійснювати розподіл задач і забезпечувати виконання вимог до часу виконання запитів. В цьому методі передбачається, що окрім агентів, що виконують запити, у систему вводяться агенти-координатори, які здійснюють їх розподіл.

Введення таких елементів призведе до збільшення часу обробки запитів.

$$\tau_e = t(Q_c) + \sum_{i=1}^N t(Q_i) + \sum_{i=1}^{N-1} \tau_{i,i+1}, \quad (4)$$

де Q_c – продуктивність координуючого агента.

Такий МАС відповідає модель системи масового обслуговування типу $G/G/M$.

Відповідно до [11] пропускна здатність такої багатоканальної системи обчислюється як

$$\rho = \frac{\lambda\tau_e}{N_{MAC}}, \quad (5)$$

отже, в цьому випадку довжина черги запитів буде дорівнювати

$$\bar{N} = \frac{\lambda\tau_e}{N_{MAC} - \lambda\tau_e}.$$

Величина черги при розподілі запитів між N_{MAC}

$$V = \begin{cases} \frac{\lambda\tau_e}{N_{MAC} - \lambda\tau_e}, & \frac{N_{MAC}}{\tau_e} < \lambda; \\ \theta, & \frac{N_{MAC}}{\tau_e} \geq \lambda, \end{cases} \quad (6)$$

де $\theta \rightarrow \infty$.

Враховуючи (3)–(6), час реакції системи буде таким:

$$T_{R_{MAC}} = \frac{\lambda\tau_e}{N_{MAC} - \lambda\tau_e} (\tau_e + \tau_q).$$

Достатньою умовою на обслуговування мультиагентною системою черги запитів є

$$\frac{\lambda\tau_e}{N_{MAC} - \lambda\tau_e} = 1.$$

Для цього кількість дублюючих агентів МАС буде визначатись як

$$N_{MAC} = 2\lambda\tau_e. \quad (7)$$

Таким чином, сутність запропонованого підходу полягає в тому, що у МАС вводиться

агент, який розподіляє потоки запитів. Після надходження запиту координуючий агент створює окремих канал передачі даних до довільно обраного спеціалізованого агента, який водночас може бути як виконавцем спеціалізованої задачі, так і координатором її виконання. Після повернення результатів від локального координатора координатор вищого рівня передає задачу на обробку іншій групі агентів згідно із встановленим порядком її обробки. Кількість агентів у кожній групі встановлюється відповідно до поточного значення пропускної здатності системи та виразу (7). Таким чином забезпечується загальна надійність системи за рахунок дублювання її компонентів та стійкість системи до перевантажень.

Висновки

Запропонована інтелектуальна мультиагентна технологія є підходом до побудови розподіленої системи оптимізації Інтернет-ресурсів з використанням методу Map-Reduce.

В роботі визначена архітектура мультиагентної системи оптимізації Інтернет-ресурсів, включаючи агентів та агентну платформу, запропоновано функціональну модель та метод розподілу завдань і планування обчислювальних потужностей. Також в роботі запропоновано підхід до формалізації функціональної декомпозиції задач та процесів взаємодії в такій системі.

Запропонована технологія може бути застосована не тільки для оптимізації Інтернет-ресурсів, але і для інших систем, що оперують слабоструктурованими даними великої розмірності.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. The Semantic Web [Електронний ресурс] / Tim Berners-Lee, James Hendler, Ora Lassila // Scientific American Magazine. — May, 2001. — Режим доступу : <http://www.sciam.com/article.cfm?id=00048144-10D2-1C70-84A9809EC588EF21>.
2. Botafogo R. A. Identifying hierarchies and useful metrics / R. A. Botafogo, E. Rivlin, B. Shneiderman // ACM Transactions on Information Systems (TOIS). — 1992. — № 2. — P. 142—180.
3. Кедрова Г. Е. Методы оптимизации компьютерной обучающей среды по лингвистике для систем дистанционного обучения в Интернете [Электронный ресурс] / Г. Е. Кедрова // Эффективность использования новых информационных технологий в учебном процессе (ЭНИТ-2000) : матер. науч.-практ. конфер. — Ульяновск, 2000. — Режим доступа : <http://www.philol.msu.ru/~kedr/ke-dr-ulj.htm>.
4. Fabrizio Marozzo. P2P-MapReduce: Parallel data processing in dynamic Cloud environments / Marozzo Fabrizio, Talia Domenico, Trunfio Paolo // Journal of Computer and System Sciences. — Vol. 78, № 5. — P. 1382—1402, Elsevier Science, September, 2012.
5. Michael Isard. European Conference on Computer Systems (EuroSys) / Isard Michael, Budiu Mihai. — Lisbon, Portugal, March 21—23, 2007.
6. A FIPA Compliant Agent Platform for Federated Information Systems / [M. Panti, L. Penserini, L. Spalazzi, S. Valenti] // ACIS International Journal of Computer & Information Science — Special issue on software engineering applied to networking & parallel/distributed computing archive. — 2000. — Vol. 1. — № 3. — P. 145—156.
7. Бугайченко Д. Ю. Абстрактная архитектура интеллектуального агента и методы ее реализации [Электронный ресурс] / Д. Ю. Бугайченко, Д. Ю. Соловьев // Системное программирование. — СПб. : изд-во СПбГУ, 2005. — Вып. 1. — С. 36—67. — Режим доступа : <http://www.sysprog.info/2005/03.pdf>.
8. Jennings N. R. Commitments and conventions: The foundation of coordination in multi-agent systems / N. R. Jennings // Knowledge Engineering Review, 1993. — № 8(3). — P. 223—250.
9. Communicating and Mobile Systems: the Pi-Calculus, Robin Milner. Cambridge University Press, 1999. — 161 p.
10. Mark D. Hill. Amdahl's Law in the Multicore Era / D. Hill Mark, R. Michael Marty // IEEE Computer. — Vol. 41. — № 7. — P. 33—38, July 2008.
11. Клейнрок Л. Теория массового обслуживания / Л. Клейнрок ; пер. с англ. — М. : Машиностроение, 1979. — 432 с.
12. Дубовой В. М. Оптимізація підсистем збору даних АСУТП в умовах комбінованої невизначеності : моногр. / В. М. Дубовой, О. Д. Никитенко. — Вінниця : УНІВЕРСУМ-Вінниця, 2011. — 169 с.

Рекомендована кафедрою комп'ютерних систем управління

Стаття надійшла до редакції 18.10.12

Рекомендована до друку 24.10.12

Дубовой Володимир Михайлович — завідувач кафедри, **Москвін Олексій Михайлович** — аспірант. Кафедра комп'ютерних систем управління, Вінницький національний технічний університет, Вінниця