УДК 004.042 DOI: https://doi.org/10.33099/2304-2745/2022-3-76/101-110

Фесьоха В. В., доктор філософії (0000-0001-6612-1970)
Нерознак Є. І. (0000-0001-5641-5473)
Сова О. Я., доктор технічних наук, старший науковий співробітник (0000-0002-7200-8955)
Нестеров О. М., доктор філософії (0000-0001-5092-6205)

Військовий інститут телекомунікацій та інформатизації імені Героїв Крут, Київ

Метод адаптивного балансування навантаження в кластерних системах військового призначення на основі рівноваги Неша

Резюме: У статті удосконалено метод адаптивного балансування навантаження кластерної системи військового призначення на основі рівноваги Неша, який, на відміну від існуючих, дає змогу досягти стійкої рівноваги оптимального використання серверних ресурсів із максимальною конвергенцією до соціального оптимуму.

Ключові слова: балансування навантаження; кластер; теорія ігор; рівновага Неша; імовірнісний розподіл; знаходження оптимуму.

Постановка проблеми. В умовах постійного зростання попиту на інформатизацію всіх сфер життєдіяльності суспільства залишається відкритим питання надійного та стабільного (неперервного) функціонування інформаційних систем/сервісів (ІС), які надають послуги користувачам у режимі реального часу. Так, за звітними даними [1] кількість клієнтських запитів зростає майже експоненціально, що унеможливлює вирішення цього питання за допомогою постійного горизонтального та/або вертикального масштабування інформаційних систем в умовах обмеженості серверних ресурсів кластерних систем.

У цьому контексті на особливу увагу заслуговує стабільне (неперервне) функціонування ІС критичної інфраструктури держави, зокрема ІС військового призначення (сил оборони та безпеки), адже їх функціонування як у повсякденній діяльності, так і в умовах воєнного стану передбачають щоденне прийняття важливих для національної безпеки рішень, захисту територіальної цілісності України, управління військами, провадження антитерористичної та контрдиверсійної діяльності, збереження життя, прав і свобод громадян [2, 3].

Одним з основних напрямів вирішення цього питання є підвищення продуктивності кластерних систем – апаратного фундаменту загального і спеціального програмного забезпечення, завдяки ефективному балансуванню навантаження між серверами кластерних систем для оптимального їх використання в умовах обмеження апаратних ресурсів і прийнятного часу на обробку запитів [4].

Це обумовлює актуальність подальших наукових досліджень щодо підвищення ефективності балансування навантаження кластерних систем військового призначення, спричиненого непередбачуваною зміною інтенсивності та/або важливістю (пріоритетністю) клієнтських запитів.

Аналіз останніх досліджень і публікацій. Аналіз наукових публікацій показав доцільність застосування теоретико-ігрового підходу до вирішення вказаного питання [4–14]. Дійсно, стратегії (діяльність) клієнтів (клієнтських запитів) є егоїстичними стосовно один-до-одного (оптимізація власної продуктивності без узгодження з іншими), але водночас залежать від стратегій інших учасників цього процесу і, як наслідок, впливають на результат навантаження кластерної системи загалом, оскільки пов’язані з використанням та перерозподілом спільних ресурсів [4]. Очевидно, що описана взаємодія є некооперативною грою, де гравці – клієнти (клієнтські запити), які намагаються мінімізувати очікуваний час відповіді власних завдань, а множина стратегій гри – сервери кластерної системи, до яких спрямовані клієнтські запити.

Одним з можливих вирішень завдань цього класу є концепція теорії математичних моделей прийняття оптимальних рішень в умовах конфлікту – рівновага Неша [4–14]. Це така множина стратегій або дій у грі з двома чи більше гравцями, згідно з якими кожен учасник реалізує оптимальну стратегію, передбачаючи дії суперників, за якій жоден із учасників не може збільшити виграш, змінивши вибір стратегії в односторонньому порядку, коли інші учасники не змінюють свого вибору.

У роботі [4] піднімається питання пошуку кількості кроків, необхідних для досягнення чистої рівноваги Неша в сценарії балансування навантаження розподіленої системи. У цій роботі автори досліджують час, потрібний системі для конвергенції до рівноваги Неша, а не якість отриманого розподілу. Пошук конвергенції до рівноваги Неша полягає у вирівнюванні зважених функцій навантаження серверу через надання можливості запитам (завданням) змінювати сервер для мінімізації власних витрат (поки не буде досягнуто рівноваги за Нешем, принаймні один запит (завдання) "бажає" змінити обраний сервер). Отримані результати обчислювальної складності значно різняться: деякі стратегії збігаються за поліноміальний час, для інших час збіжності може потребувати експоненціального числа кроків. До того ж запропонований підхід справедливий для профілів чистих стратегій.

У роботі [5] концепція рівноваги Неша для балансування навантаження викладена у термінах очікуваного навантаження на сервер у профілях змішаних стратегій. Доведено, що максимальне очікуване навантаження, яке відчуває будь-який користувач у рівновазі Неша, щонайбільше вдвічі перевищує мінімально можливе. Завдання балансування навантаження зводиться до ймовірнісного розрахунку очікуваного навантаження на сервер у результаті кожного запиту користувача. Проте вагою запиту є попередній час обробки, що не завжди дає змогу адекватно їх зважувати, враховуючи той факт, що реально оцінити складність запиту можливо тільки після його отримання блоком серверної логіки ІС.

У роботі [6] описано клас некооперативних ігор, у яких гравці поділяють загальний набір стратегій (серверів). Виграш, який отримує гравець, залежить лише від загальної кількості гравців, які грають за такою ж стратегією, і зменшується разом із цим числом у спосіб, який є специфічним для конкретного гравця. Показано, що кожна некооперативна гра цього класу має принаймні одну рівновагу Неша у чистих стратегіях. Розглядається пошук кращої рівноваги Неша з усіх знайдених, проте лише на профілях чистих стратегій.

У роботах [7–10] досліджується питання ціни анархії (стабільності) некооперативної гри балансування навантаження кластерної системи (співвідношення найкращої (найгіршої) рівноваги Неша до соціального оптимуму) у тому числі і на сильних рівновагах, де жодна коаліція гравців не має спільного відхилення, тому кожен гравець отримує чіткий виграш. Поряд з цим, сильна рівновага за Ауманом (удосконалений варіант рівноваги Неша) не завжди існує на множині стратегій у грі такого класу, тому результати справедливі лише в умовах специфічних обмежень.

У роботах [11–13] вивчається теоретико-ігровий варіант мінімізації робочого діапазону, тобто максимального навантаження на всі сервери. Запит (завдання) намагається розміститись на сервері з найменшим показником навантаження. Вивчається ціна анархії для таких ігор із балансуванням навантаження в чотирьох різних варіантах і досліджується складність обчислювальної рівноваги. Представлено, що ціна анархії для чистої та змішаної рівноваг на ідентичних серверах мають порядок $\frac{\log m}{\log\log m}$, де m – кількість серверів. Додатково розглядається питання досягнення рівноваги Неша. У випадку з ідентичними за параметрами серверами максимальна вага найкращої відповіді досягає рівноваги лише за $O(n)$, де $O(n)$ – обчислювальна складність запитів.

У роботі [14] завдання балансування навантаження у розподіленій (кластерній) системі сформульовано як некооперативна гра між кінцевими користувачами, які намагаються мінімізувати очікуваний час відповіді своїх власних запитів. Представлено розподілений алгоритм для обчислення рівноваги Неша, який перевершує відомі схеми з погляду, так званої, справедливості навантаження. Проте за умови значної кількості гравців обчислювальна складність процесу пошуку рівноваги Неша значно зростає.

На основі проведеного аналізу можна дійти висновків:

вирішення завдання балансування навантаження у переважній більшості існуючих підходів зводиться до пошуку рівноваги Неша (конвергенції до неї) на профілях чистих стратегій, хоча гарантувати її наявність можливо лише на профілях змішаних стратегій;

у наукових працях, які розглядають у тому числі і профілі змішаних стратегій, гравцями некооперативної гри є клієнти або запити (завдання), що поступають від клієнтів до кластерної системи, проте у цьому випадку кількість гравців є необмеженою (неконтрольованою), внаслідок чого розмірність платіжної матриці може сягати надвеликих розмірів (десятки та сотні тисяч запитів), що значно збільшує обчислювальну складність процесу балансування навантаження та додатково піднімає питання визначення моменту ймовірнісного перерозподілу вибору стратегій;

використання вказаних підходів дає змогу досягти стійкої рівноваги, яка не завжди максимізує виграші гравців на платіжній матриці [4–14];

наявні підходи реалізовані у рамках специфічних обмежень, що не дає змоги методам (моделям) балансування навантаження кластерної (розподіленої) системи ефективно адаптуватися до динаміки значень усіх чинників, які одночасно впливають на цей процес (рівень завантаженості серверів, рівень складності запитів тощо);

виведення з ладу одного із серверів призводить до втрати запитів (завдань) загалом.

Представлена фактологічна основа обумовлює доцільність удосконалення існуючих підходів балансування навантаження кластерних систем на основі рівноваги Неша, зокрема систем військового призначення, що дасть змогу вирішити окреслену проблематику.

З огляду на це, виникає актуальне наукове завдання, пов’язане з розробленням адаптивного методу балансування навантаження кластерних систем військового призначення на основі рівноваги Неша.

Метою статті є удосконалення методу адаптивного балансування навантаження кластерної системи військового призначення на основі рівноваги Неша для досягнення стійкої рівноваги оптимального використання серверних ресурсів із максимальною конвергенцією до соціального оптимуму.

Виклад основного матеріалу дослідження.

Вихідні дані: гра G – процес стратегічної взаємодії множини стратегій (серверів) $S = \{s_1, \cdots, s_i, \cdots, s_n\}$, де n – кількість серверів кластерної системи із множиною клієнтських запитів $Q = \{q_1, \cdots, q_i, \cdots, q_m\}$ до них, де m – кількість клієнтських запитів, під час якої виграші $U_i: s_i \times \cdots \times s_n \to \mathbb{R}$, де $\mathbb{R}$ – множина усіх можливих комбінацій, залежать від дій один одного. Виграш U_i – значення функції виграшу у платіжній матриці на обраних (указаних) S стратегіях – мінімальне значення завантаженості сервера l_{s_i} серед всіх наявних серверів кластерної системи з урахуванням власного робочого навантаження та навантаження спричиненого як запитами, які перебувають в обробці, так і новими, які мають обробитися в один момент часу. Множина стратегій $S = \{s_1, \cdots, s_i, \cdots, s_n\}$ – опис дій клієнтських запитів в усіх можливих ситуаціях гри G. Результат гри G – збалансована комбінація обраних (вказаних) стратегій S, яка знаходиться у стійкій рівновазі Неша із прийнятною конвергенцією до соціального оптимуму (мінімальне значення завантаженості серверів кластерної системи для всіх підзадач одночасно). Запит q_i – програмне формулювання інформаційної потреби користувачем. Завдання – діяльність сервера, спричинена запитом користувача q_i. Вага w_{q_i} запиту (завдання) – рівень складності його обробки. Множина підзадач кожного отриманого завдання q_i – $SQ_{q_i} = \{sq_1^{q_i}, \cdots, sq_2^{q_i}\}$. Підзадача $sq_j^{q_i}$ – самостійний елемент батьківського завдання q_i з аналогічною множиною властивостей.

Обмеження та допущення: гра G є некооперативною, динамічною, несиметричною, паралельною, з ненульовою сумою, з кінцевою і спільною множиною серверів (стратегій) S для кінцевої множини підзадач SQ_{q_i} (гравців) отриманого завдання q_i. Вид стратегій – змішані. Гравці SQ_{q_i} не відхиляються від обраної (вказаної) стратегії діяльності $s_i \in S$.

Необхідно: отримати таке збалансоване призначення множини зважених підзадач $SQ_{q_i}, q_i \in Q$ на S серверах $(SQ_{q_i}) \to (s_i^* \times \cdots \times s_n^*)$, що для $\forall\, sq_j^{q_i} \in SQ_{q_i}$ та $\forall\, s_i \in S$ виконано обов’язкову умову $U_i(s_i^*, s_{-i}^*) \geq U_i(s_i, s_{-i}^*)$ із врахуванням рівня завантаженості кожного сервера кластерної системи.

Рішення. Формалізуємо процес адаптивного балансування навантаження на основі рівноваги Неша як некооперативну гру G у нормальній формі з ненульовою сумою, яка задана певною множиною підзадач $SQ_{q_i} = \{sq_1^{q_i}, \cdots, sq_2^{q_i}\}$ отриманого завдання q_i від користувача з відповідною вагою складності його обробки w_{q_i}, множиною стратегій їх функціонування $S = \{s_1, \cdots, s_i, \cdots, s_n\}$ – серверами кластера, а також відповідними функціями їх виграшу $U_i: s_i \times \cdots \times s_n \to \mathbb{R}$. Узагальнену платіжну матрицю описаної гри наведено на рис. 1, де u_i – значення функції виграшу для кожного гравця $sq_j^{q_i}$ у комбінації стратегій $s_i \times \cdots \times s_n$ – показники рівня завантаженості серверів l_{s_i}; n – кількість серверів; m – кількість симетричних підзадач.

q_i		$sq_2^{q_i}$	
		s_1	s_n
$sq_1^{q_i}$	s_1	u_{11}	u_{1n}
	s_n	u_{n1}	u_{nn}

Рис. 1. Узагальнена платіжна матриця некооперативної гри адаптивного балансування навантаження на основі рівноваги Неша

Результат профілю $(s_i^* \times \cdots \times s_n^*)$ буде урівноваженим за Нешем тільки тоді, якщо для $\forall\, q_i \in Q$ та обраної (вказаної) $\forall\, s_i \in S$ виконано імплікацію, що його виграш $U_i(s_i^*, s_{-i}^*) \geq U_i(s_i, s_{-i}^*)$, де s_{-i}^* – умова (гарантія), що решта $q_j \in Q$ не змінюють обрані (вказані) $s_j \in S$. Іншими словами, профіль стратегій S^* підзадач буде урівноваженим за Нешем, якщо жодна з них не може отримати кращого значення функції виграшу, односторонньо відхилившись від власної стратегії [4–16].

Для прикладу, клієнт надсилає файл великого розміру (Big Data) на обробку до кластерної системи для виконання задачі виявлення аномалій у таких даних на основі методу еліпсоїдної апроксимації, унаслідок чого клієнт отримає аналітичні дані для прийняття подальших рішень. Після чого цю задачу (файл) q_j система має розділити на підзадачі (частини файла) $sq_j^{q_i}$ та виконати такий розподіл підзадач, при якому використання серверних ресурсів (стратегій $S = \{s_1, \cdots, s_i, \cdots, s_n\}$) кластерної системи буде оптимально збалансованим.

На рис. 2 наведено платіжну матрицю некооперативної гри (на профілі чистих стратегій) серверної взаємодії із підзадачами, що підлягають обробці. Значення виграшів – рівень завантаженості серверів за показником очікуваного часу обробки наявних підзадач разом із новими (мсек). Кількість k гравців (підзадач) – 2 (sq_1, sq_2).

q_i		$sq_2^{q_i}$	
		s_1	s_2
$sq_1^{q_i}$	s_1	300\300	100\500
	s_2	500\100	200\200

Рис. 2. Платіжна матриця некооперативної гри у рівновазі Неша на профілі чистих стратегій

З рис. 2 видно, що на приведеному профілі чистих стратегій s_1, s_2 існує одна рівновага Неша у разі вибору кожною підзадачею сервера s_1 – $(s_1^* \times s_1^*)$. Так, жодна з підзадач $sq_j^{q_i}$ не може отримати краще значення функції виграшу (сервер з найменшим показником завантаження), односторонньо відхилившись від попередньо обраної стратегії, що дає змогу отримати стійку рівновагу, до того ж оптимальну за Парето [4–16]. Особливістю цього випадку є профіль чистих стратегій $s_2^* \times s_2^*$, де значення функції виграшу мінімізується для всіх гравців – соціальний оптимум. Проте така комбінація чистих стратегій не є стійкою, оскільки кожна підзадача має можливість мінімізації власного значення функції виграшу, односторонньо відхилившись від попередньо обраної стратегії. Таким чином, профіль $(s_1^* \times s_1^*)$ хоч і не забезпечує мінімального значення функції виграшу для цих підзадач, однак є стійким у рівновазі.

На рис. 3 зображено варіанти збалансованого навантаження серверів s_1, s_2 і 4 завдання з різним ступенем складності (2 складних, 2 простих), що перебувають у рівновазі Неша (зміна сервера завданням не забезпечить максимізацію виграшу).

Рівновага Неша на рис. 3а не досягає соціального оптимуму, на рис. 3б – соціальний оптимум. На практиці досягти стійкого у рівновазі розподілу завдань, який максимізує конвергенцію до соціального оптимуму у часі практично не можливо, в силу динаміки значень впливових чинників на цей процес. Так, варіанти 3а і 3б будуть певним чином чергуватися.

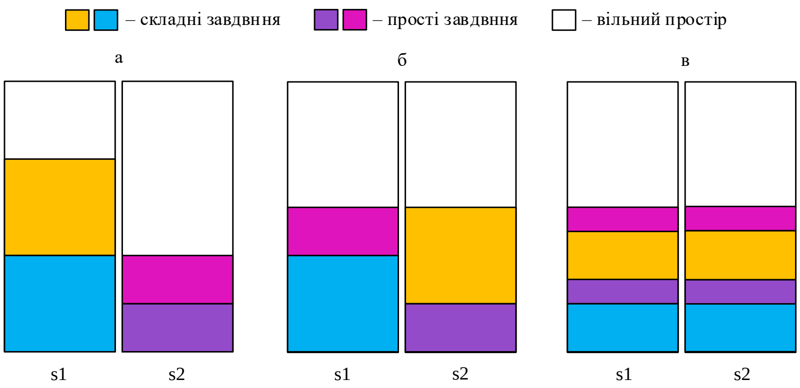

Рис. 3. Варіанти збалансованого навантаження серверів s_1, s_2 у рівновазі Неша

Суть запропонованого у статті методу, яка відрізняє його від існуючих [4–14], полягає, насамперед, у декомпозиції кожного завдання як наслідку клієнтського запиту на симетричні підзадачі програмним засобом для розподілених паралельних обчислень Map Reduce [17, 18], а також у знаходженні рівноваги Неша на множині змішаних стратегій даних підзадач шляхом визначення для них імовірнісного розподілу паралельного використання серверів з урахуванням динаміки впливових на процес балансування навантаження чинників (рівень завантаженості серверів, складність задач) в умовах прийнятної обчислювальної складності (рис. 3в). Так, поява нового завдання призведе до його декомпозиції на підзадачі з подальшим оптимальним розподілом їх паралельної обробки. Такий підхід дає змогу забезпечити ефективне використання серверних ресурсів у рівновазі за Нешем з максимальною конвергенцією до соціального оптимуму. Процес адаптивного балансування навантаження на основі рівноваги Неша представлено функціональною схемою, яка передбачає реалізацію певних етапів (рис. 4).

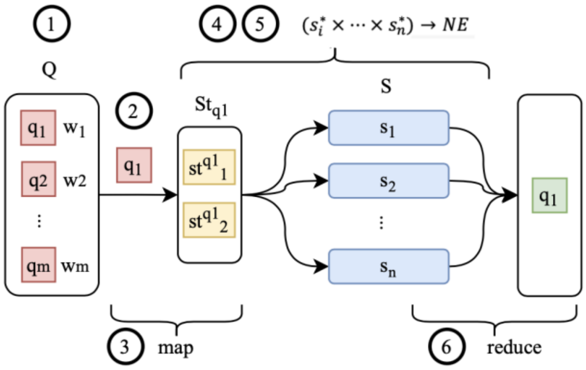

Рис. 4. Функціональна схема балансування навантаження кластерної системи на основі рівноваги Неша

1. Ініціалізація. На цьому етапі визначаються завдання, суб'єкти, об'єкти, умови, а також виграші некооперативної гри процесу балансування навантаження.

Гра (G)*:* теоретико-ігровий варіант процесу динамічного балансування навантаження кластерної системи представлено як некооперативну гру у нормальній формі з ненульовою сумою на змішаних стратегіях (комбінація чистих стратегій із певною частотою), оскільки саме такий підхід дає змогу достатньо повно описати сервісну взаємодію кластера із клієнтами у процесі задоволення інформаційних потреб останніх $Q = \{q_1, \cdots, q_i, \cdots, q_m\}$, а також гарантує наявність хоча б однієї рівноваги Неша на будь-якому профілі стратегій [4–15, 20]. Гравці SQ_{q_i} поділяють спільний набір стратегій $S = \{s_1, \cdots, s_i, \cdots, s_n\}$. Завдання – діяльність сервера, спричинена запитом користувача q_i із вагою w_{q_i}, що характеризує рівень його складності.

Гравці (SQ_{q_i})*:* у ролі гравців представлено підзадачі $SQ_{q_i} = \{sq_1^{q_i}, \cdots, sq_2^{q_i}\}$ отриманих завдань $q_i \in Q$ кластерною системою з відповідними вагами складності внаслідок запитів користувачів $Q = \{q_1, \cdots, q_i, \cdots, q_m\}$, що дає змогу уникнути неконтрольованої кількості гравців і стохастичної динаміки обчислювальної складності процесу балансування навантаження між серверами кластерної системи. Кількість підзадач дорівнює 2, оскільки на практиці доведено існування рівноваги Неша на змішаних стратегіях для біматричної гри (будь-яка кількість стратегій для 2-х гравців).

Стратегії (S)*:* існуюча множина усіх можливих стратегій (серверів кластерної системи) $S = \{s_1, \cdots, s_i, \cdots, s_n, C_{s_n}^k\}$, де $C_{s_n}^k$ – підмножина усіх можливих змішаних стратегій (комбінацій чистих стратегій s_i) без повторень, k – потужність множини підзадач SQ_{q_i}.

Виграші (U_i)*:* Виграш, який отримує підзадача $sq_j^{q_i}$ залежить від отриманого профілю стратегій $s_i^* \times \cdots \times s_n^*$ унаслідок некооперативного вибору кожною підзадачею сервера $s_i \in S$. Кожна підзадача $sq_j^{q_i}$ намагається максимізувати свій виграш U_i через мінімізацію власного часу на обробку (обрати сервер із найменшим навантаженням l_{s_i}) [4]. Виграші – значення завантаженості серверів l_{s_i} за показником очікуваного часу обробки підзадач.

Результат: визначення стійкого у рівновазі та оптимального профілю використання серверних ресурсів під час обробки множини підзадач SQ_{q_i} із максимальною конвергенцією до соціального оптимуму.

2. Отримання завдання. На цьому етапі передбачається приймання множини зважених завдань балансувальником унаслідок отримання користувальницьких запитів $Q = \{q_1, \cdots, q_i, \cdots, q_m\}$.

3. Декомпозиція завдання на підзадачі. Отримане завдання q_i декомпозується балансувальником навантаження на множину підзадач $SQ_{q_i} = \{sq_1^{q_i}, \cdots, sq_2^{q_i}\}$, потужність якої дорівнює 2. Підхід до логічної декомпозиції задачі базується на програмній моделі MapReduce для розподіленої паралельної обробки великих масивів даних із використанням кластерів [17, 18], який показує значну ефективність у галузях Big Data, розподіленого пошуку та сортування даних, звернення графа вебпосилань, обробка статистики логів мережі, побудови інвертованих індексів, кластеризації документів, машинного навчання та статистичного машинного перекладу. Використання цього підходу передбачає реалізацію процедури map – попередньої обробки завдання q_i та його ваги w_{q_j} як переліку значень балансувальником у вигляді списку, проте на відміну від класичної моделі MapReduce [17, 18] балансувальник навантаження ділить згенерований список на дві підзадачі та передає їх на етап обчислення виграшів U_i у платіжній матриці.

4. Обчислення виграшів. Кожна підзадача $sq_j^{q_i}$ намагається максимізувати свій виграш U_i (обрати сервер із найменшим навантаженням l_{s_i} з метою максимізації його ресурсів для мінімізації власного часу на обробку $t_{sq_j^{q_i}}$).

Розрахунок серверного навантаження l_{s_i} у платіжній матриці ґрунтується на основі відношення суми ваг w_{sq_j} усіх підзадач $sq_j^{q_i}$ різних завдань q_i, що ним обслуговуються з відповідною швидкістю v_{s_i}, яку регламентовано технічними характеристиками сервера s_i:

$$l_{s_i} = \sum_1^n \frac{w_{sq_j^{q_i}}}{v_{s_i}}, \quad (1)$$

де ξ – власне навантаження сервера, не викликане клієнтськими запитами.

Значення виграшів U_i у платіжній матриці розраховуються на основі показника l_{s_i} з урахуванням очікуваного часу на обробку $t_{sq_j^{q_i}}$ для кожної нової підзадачі $sq_j^{q_i}$ на множині серверів:

$$U_i\left(sq_j^{q_i}\right) = l_{s_i} + \frac{w_{sq_j^{q_i}}}{v_{s_i}}. \quad (2)$$

5. Балансування навантаження. NE (Nash equilibrium) – процедура пошуку рівноваги Неша на множині змішаного вибору серверів через визначення такого ймовірнісного розподілу для кожної підзадачі $sq_j^{q_i} \in SQ_{q_i}$, щоб значення їх виграшів на платіжній матриці (див. рис. 1) були рівними на усіх стратегіях $s_i \in S$:

$$NE: p\left(sq_1^{q_i} \to S\right) = \begin{cases} p_{sq_1^{q_i}}(s_i) = p_{sq_2^{q_i}}(s_i) * u_{11} \pm \left(\sum\limits_{s_i \in S} p_i(s_i) - p_{sq_2^{q_i}}(s_i)\right) * u_{1n} \\ p_{sq_1^{q_i}}(s_n) = p_{sq_2^{q_i}}(s_i) * u_{n1} \pm \left(\sum\limits_{s_i \in S} p_i(s_i) - p_{sq_2^{q_i}}(s_i)\right) * u_{nn} \end{cases}, \quad (3)$$

$$NE\colon p\left(sq_2^{q_i} \to S\right) = \begin{cases} p_{sq_2^{q_i}}(s_i) = p_{sq_1^{q_i}}(s_i) * u_{ii} \overset{+}{-} \left(\sum\limits_{s_i \epsilon S} p_i(s_i) - p_{sq_1^{q_i}}(s_i)\right) * u_{ni} \\ p_{sq_2^{q_i}}(s_n) = p_{sq_1^{q_i}}(s_i) * u_{in} \overset{+}{-} \left(\sum\limits_{s_i \epsilon S} p_i(s_i) - p_{sq_1^{q_i}}(s_i)\right) * u_{nn} \end{cases} . \qquad (4)$$

Таким чином, на основі отриманого імовірнісного розподілу $p(sq_j^{q_i})$ досягається оптимальний профіль серверів для кожної підзадачі $sq_j^{q_i}$ у відповідь на змішаний вибір серверів $s_i \in S$ іншою підзадачею $sq_j^{q_i}$, оскільки забезпечується рівність значень функції виграшу на різних серверах. Іншими словами, після процедури NE окремій підзадачі $sq_j^{q_i}$ без різниці, який із серверів буде обрано іншою підзадачею (у будь-якому разі виграш буде однаковий).

Варто зауважити, що пошук рівноваги Неша на змішаних стратегіях на основі (3), (4) у більшості випадків справедливий для гри, у платіжній матриці якої немає домінуючих (строго домінуючих) змішаних стратегій.

У разі наявності в платіжній матриці домінуючих (строго домінуючих) змішаних стратегій (значення отриманого ймовірнісного розподілу може виходити за межі $0 \leq sq_j^{q_i} \leq 1$) пошук точки рівноваги необхідно вирішувати на основі підходу вибору домінуючих (строго домінуючих) змішаних стратегій (виключення стратегій, що домінуються) [19–21].

Вибір сервера s_i^* є строго домінуючим для окремої підзадачі $sq_j^{q_i}$ у грі балансування навантаження серверів кластерної системи, якщо він строго домінує будь-який вибір іншого сервера $s_i \in S$.

Відповідно до [21] кожна чиста стратегія гравця може розглядатися як його змішана стратегія, у якій ця чиста стратегія вибирається з імовірністю 1, а решта – з ймовірністю 0. Так, визначення ситуації рівноваги ґрунтується на твердженні, що стратегія, яка строго домінується іншою не може входити у будь-яку рівновагу Неша (імовірність вибору цієї стратегії у процесі змішування дорівнює нулю) [21]. До того ж, змішаний вибір серверів зможе бути таким, який строго домінується навіть у разі якщо він використовує чисті стратегії з позитивною ймовірністю, які не є такими, що домінуються.

До того ж у профілі змішаного вибору серверів завжди існує хоча б одна рівновага Неша. Узагальнену схему оптимального профілю серверів наведено на рис. 5.

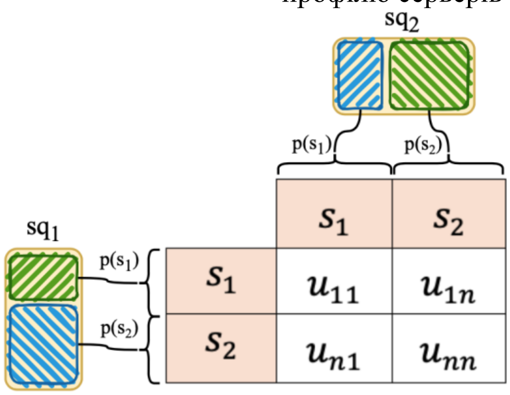

Рис. 5. Узагальнена схема профілю серверів на основі рівноваги Неша

Отже, на основі визначеної рівноваги Неша на профілі змішаного вибору серверів $s_i \in S$ підзадачами $sq_j^{q_i} \in SQ_{q_i}$ до кожного сервера s_i передається вказана частина підзадачі для подальшої обробки, розмір якої відповідає частоті його вибору у грі адаптивного балансування серверного навантаження кластерної системи, що призводить зі свого боку до отримання оптимального та водночас стійкого варіанта використання наявних серверних ресурсів.

Далі множина (пара) підзадач SQ_{q_i} розподіляється на профіль серверів $s_i \in S$ за вказівкою отриманого значення розподілу, де кожен сервер $s_i \in S$, який отримав підзадачу записує результат у форматі "ключ-значення" (ключі раніше створені процедурою *map*) у тимчасове сховище для подальшої обробки.

6. Об'єднання підзадач. На цьому етапі процесу балансування навантаження кластерної системи на основі рівноваги за Нешем передбачається реалізація процедури *reduce* – паралельна розподілена обробка кожним сервером груп даних за порядком проходження ключів та з’єднання результатів балансувальником навантаження. Отриманий результат – рішення завдання, отриманого на основі користувальницького запиту $q_i \in Q$.

Приклади:

Варіант №1 (сервери збалансовано). Розглянемо некооперативну гру серверної взаємодії з підзадачами, платіжну матрицю якої зображено на рис. 6 (етапи: 1-4 реалізовано). Необхідно визначити оптимальний і водночас стійкий у рівновазі профіль використання серверних ресурсів під час виконання множини підзадач SQ_{q_i} із максимальною конвергенцією до соціального оптимуму. Поточне навантаження серверів однакове (100 мс очікуваного часу на обробку наявних підзадач). Очікуваний час навантаження серверів новими 2-ма підзадачами становить 3 секунди для кожної підзадачі. Відповідно до (2) значення навантаження серверів представлено у платіжній матриці у вигляді очікуваного часу обробки наявних підзадач із врахуванням очікуваного часу на обробку пари нових підзадач.

q_i		$sq_2^{q_i}$	
		s_1	s_2
$sq_1^{q_i}$	s_1	106\106	103\103
	s_2	103\103	106\106

Рис. 6. Платіжна матриця некооперативної гри на профілі змішаних стратегій (варіант № 1)

Для реалізації етапу 5 необхідно виконати процедуру пошуку рівноваги Неша (NE) на профілях змішаних стратегій $s_1 \times s_2$. Оскільки у цій платіжній матриці немає домінуючих (строго домінуючих) змішаних стратегій, то ймовірність вибору підзадачею $sq_1^{q_i}$ сервера s_1 позначимо σ, а підзадачею $sq_2^{q_i} - \varphi$. Імовірності вибору ними сервера s_2 становлять $1-\sigma$ та $1-\varphi$ відповідно.

Визначимо значення розподілу для використання серверів s_1, s_2 для першої та другої підзадач на основі (3):

$$p_{sq_1^{q_i}}(s_1) = 100\varphi + 103(1-\varphi) = 106\varphi + 103 - 103\varphi = 3\varphi + 103$$
$$p_{sq_1^{q_i}}(s_2) = 103\varphi + 106(1-\varphi) = 103\varphi + 106 - 106\varphi = 106 - 3\varphi$$
$$3\varphi + 103 = 106 - 3\varphi$$
$$3\varphi + 3\varphi = 106 - 103$$
$$6\varphi = 3$$
$\varphi = 0.5$, відповідно $1 - \varphi = 0.5$.

Оскільки значення платіжної матриці симетричні, значення ймовірнісного розподілу серверів для підзадачі $sq_2^{q_i}$ також становить $\sigma = 0.5$, відповідно $1 - \sigma = 0.5$.

Таким чином, на кожен окремий сервер s_1, s_2, буде передано по одній підзадачі. Отриманий результат є єдиною рівновагою Неша на профілі змішаних стратегій описаної гри балансування навантаження та відповідає соціальному оптимуму виграшів у представленій платіжній матриці (оптимум за Парето).

Варіант № 2 (сервери не збалансовано). Розглянемо некооперативну гру серверної взаємодії із підзадачами, платіжну матрицю якої зображено на рис. 7 (етапи: 1–4 реалізовано). Поточне навантаження серверів не однакове (100 мс очікуваного часу на обробку наявних підзадач сервером s_1, 60 – сервером s_2). Очікуваний час навантаження серверів новими 2-ма підзадачами становить 3 мс для кожної підзадачі. Відповідно до (2) значення навантаження серверів представлено у платіжній матриці у вигляді очікуваного часу обробки наявних підзадач із врахуванням очікуваного часу на обробку пари нових підзадач. Соціальний оптимум цієї гри – $s_2 \times s_2$. У наведеному прикладі не існує рівноваги Неша у профілі чистих стратегій.

q_i		$sq_2^{q_i}$	
		s_1	s_2
$sq_1^{q_i}$	s_1	106\106	103\63
	s_2	63\103	66\66

Рис. 7. Платіжна матриця некооперативної гри на профілі змішаних стратегій (варіант № 2)

Для реалізації етапу 5 необхідно виконати процедуру пошуку рівноваги Неша (NE) на профілях змішаних стратегій $s_1 \times s_2$. Оскільки у цій платіжній матриці є строго домінуючі стратегії (для підзадачі $sq_1^{q_i}$ знизу, для підзадачі $sq_2^{q_i}$ праворуч), а значення отриманого розподілу дорівнює – 6.16 (виходить за межі $0 \le sq_j^{q_i} \le 1$), то шляхом виключення чистої стратегій, які домінуються (вибираються з нульовою імовірністю) отримуємо профіль $s_2 \times s_2$, який знаходиться як у рівновазі за Нешем, так і оптимальним за Парето.

Отже, на сервер s_2 буде передано отриману задачу повністю. Отриманий результат є єдиною рівновагою Неша на профілі змішаних стратегій $s_2 \times s_2$ описаної гри балансування навантаження та відповідає соціальному оптимуму виграшів у представленій платіжній матриці (оптимум за Парето).

Конвергенція до соціального оптимуму. Оскільки, під соціальним оптимумом гри балансування навантаження між серверами кластерної системи, розуміється мінімальне значення завантаженості серверів для всіх підзадач одночасно, то доцільно визначити міру збіжності (конвергенцію) знайденої рівноваги Неша до соціального оптимуму.

Наведені розрахунки демонструють, що у процесі балансування навантаження серверів кластерної системи на основі запропонованого методу конвергенція отриманого виграшу до соціального оптимуму є максимальною (дорівнює 100 %) (відношення соціального оптимуму до знайденої рівноваги Неша).

Для гри, платіжна матриця якої зображена на рис. 2, отриманий профіль змішаних стратегій знаходиться у рівновазі Неша, значення виграшу якого $s_1 \times s_1$ не співпадає із значенням соціального оптимуму гри $s_2 \times s_2$ (дорівнює 67 %) тому, що для прикладу було взято випадкові значення (без збереження залежності між значеннями виграшів на основі (2)).

Висновки. У статті розглянуто актуальне наукове завдання динамічного балансування навантаження кластерної системи військового призначення, спричинене інтенсивністю клієнтських запитів. Проведено аналіз та зроблено висновок про доцільність формалізації серверної взаємодії із клієнтськими запитами у межах завдання балансування навантаження на основі використання змішаних стратегій.

Удосконалено метод адаптивного балансування навантаження кластерної системи військового призначення на основі рівноваги Неша, який, на відміну від існуючих, дає змогу досягти стійкої рівноваги оптимального використання серверних ресурсів із максимальною конвергенцією до соціального оптимуму через визначення для кожної отриманої зваженої задачі ймовірнісного розподілу паралельного використання серверів з урахуванням поточного рівня їх завантаженості в умовах прийнятної обчислювальної складності. До того ж, запропонований підхід дає змогу вирішити описану у статті проблематику:

пошук рівноваги у змішаних стратегіях завжди має рішення;

кількість гравців обмежено біматричною грою, що дає змогу знаходити рівновагу Неша за досить короткий час;

метод ефективно адаптується до динаміки значень усіх розглянутих впливових на процес балансування навантаження факторів;

виведення з ладу одного із серверів не призводить до втрати завдань у цілому, які йому було делеговано, втрачається лише їх частина (можливо відновити за контрольною сумою).

Отримані результати відповідають актуальним вимогам до функціонування ІС критичної інфраструктури держави, зокрема ІС військового призначення у контексті їх стабільності (неперервності) та надійності).

Перспективним **напрямом подальших досліджень** є розроблення методу інтелектуального зважування задач балансувальником навантаження, отриманих унаслідок запитів користувачів.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Сервіс публічної транснаціональної корпорації Google. Google Trends. URL: https://trends.google.com.ua/trends/explore?date=today%205-y (дата звернення: 08.09.2022).
2. Про національну безпеку України : Закон України від 21.06.2018 р. № 2469-VIII // Відомості Верховної Ради. 2018. № 31. Ст. 241.
3. Фесьоха В., Фесьоха Н. Модель нечіткої автентифікації користувачів інформаційних систем органів військового управління на основі поведінкової біометрії // Захист інформації (НАУ). 2021. Т. 23, № 2. С. 116–123.
4. Convergence Time to Nash Equilibria. Eyal Even-Dar, Alex Kesselman, and Yishay Mansour School of Computer Science, Tel-Aviv University. 2003.
5. Comparing Nash equilibria to Optimal Solutions in the Load Balancing Game with randomized strategies. CS 684 Algorithmic Game Theory Scribe: Mateo Restrepo Instructor: Eva Tardos February 23, 2004.
6. Congestion Games with Player-Specific Payoff Functions. Igal Milchtaich, Department of Mathematics, The Hebrew University of Jerusalem, Jerusalem 91904, Israel Received October 9, 1993.
7. Strong price of anarchy. Nir Andelman, Michal Feldman, Yishay Mansour. Games and Economic Behavior 65 (2009) 289–317.
8. Strong Price of Anarchy for Machine Load Balancing. Amos Fiat, Haim Kaplan, Meital Levy, and Svetlana Olonetsky. International Colloquium on Automata, Languages, and Programming ICALP 2007: Automata, Languages and Programming.Tel Aviv University. P. 583–594.
9. Chen B., Li S., Zhang Y. Strong stability of Nash equilibria in load balancing games // Sci. China

Math. 2014. Vol. 57. P. 1361–1374. URL: https://doi.org/10.1007/s11425-014-4814-2 (дата звернення: 08.09.2022).
10. Baruch Awerbuch, Yossl Azar, Yossl Richter, Dekel Tsur. Tradeoffs in worst-case equilibria // Theoretical Computer Science. 2006. Vol. 361. P. 200–209. URL: https://doi.org/10.1016/ j.tcs.2006.05.010 (дата звернення: 09.09.2022).
11. Vöcking B. (2007). Selfish Load Balancing. In N. Nisan, T. Roughgarden, E. Tardos, & V. Vazirani (Eds.) // Algorithmic Game Theory P. 517–542. URL: https://doi.org/10.1017/CBO9780511800481.022 (дата звернення: 09.09.2022).
12. Christodoulou G., Koutsoupias E., Nanavati A. Coordination Mech-anisms. In Proceedings of the 31st International Colloquium on Automata, Languages and Programming (ICALP). 2004. P. 345–357.
13. Christodoulou G., Koutsoupias E., Nanavati A. (2004). Coordination Mechanisms. In: Díaz J., Karhumäki J., Lepistö A., Sannella D. (eds) Automata, Languages and Programming. ICALP 2004. Lecture Notes in Computer Science, vol 3142. Springer, Berlin, Heidelberg. URL: https://doi.org/10.1007/978-3-540-27836-8_31 (дата звернення: 09.09.2022).
14. Satish Penmatsa, Anthony T. Chronopoulos. Game-theoretic static load balancing for distributed systems // Journal of Parallel and Distributed Computing. Vol. 71, Issue 4. 2011. P. 537–555. URL: https://doi.org/10.1016/j.jpdc.2010.11.016 (дата звернення: 10.09.2022).
15. Ігнатенко О. П., Одобеску В. Я. Теоретико-ігровий аналіз планувальників у багатопроцесорних системах. Імітаційна модель // Проблеми програмування. 2018. № 2–3. С. 75–82.
16. Петросян Л. А., Зенкевич Н. А., Шевкопляс Е. В. Теория игр. Санкт-Петербург : БХВ-Петербург, 2012. 432 с.
17. Jeffrey Dean, Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters // Communications of the ACM. January 2008. Vol. 51, Issue 1. P. 107–113. URL: https://doi.org/10.1145/ 1327452.1327492 (дата звернення: 10.09.2022).
18. Laemmel R. Google's MapReduce Programming Model – Revisited. Data Programmability Team. Microsoft Corp. Redmond, WA, USA. URL: http://web.archive.org/web/20160304084223/http://userpages.unikoblenz.de/~laemmel/MapReduce/pape.pdf (дата звернення: 10.09.2022).
19. Диксит А., Скит С., Рейли-младший Д. Стратегические игры. Доступный учебник по теории игр / пер. с англ. Н. Яцюк ; науч. ред. А. Минько. Москва : Манн, Иванов и Фербер, 2017. 880 с.
20. Nash J. F. Equilibrium Points in N-Person Games // Proceedings of the National Academy of Scienses of the USA. 1950. Vol. 36. P. 48–49.
21. Кремлев А. Г. Основные понятия теории игр : учебное пособие. Екатеринбург : Урал. ун-т, 2016. 144 с.

Стаття надійшла до редакційної колегії 10.10.2022

Method of adaptive load balancing in cluster systems for military purposes based on Nash equilibrium

Annotation

In the context of the ever-growing demand for informatization of all spheres of society, the issue of reliable and stable functioning of information systems/services (IS) providing services to users in real time remains open. In this context, the stable (continuous) functioning of the critical infrastructure of the state, in particular, military IS, deserves special attention.

One of the possible solutions to this class of problems is the concept of the theory of mathematical models for making optimal decisions in a conflict – the Nash equilibrium. This is a set of strategies or actions in a game with two or more players, according to which each participant implements an optimal strategy, anticipating the actions of rivals, in which none of the participants can increase the gain by changing the choice of strategy unilaterally, when other participants do not change their choice.

The purpose of the article is to improve the method of adaptive load balancing of a military cluster system based on Nash equilibrium to achieve a stable equilibrium of optimal use of server resources with maximum convergence to the social optimum.

The author proposes an approach that allows solving the problems described in the article:

the search for equilibrium in mixed strategies always has a solution;

the number of players is limited to a bimatrix game, which allows finding a Nash equilibrium in a fairly short time;

the method effectively adapts to the dynamics of the values of all the factors considered that influence the load balancing process;

the failure of one of the servers does not lead to the loss of the tasks as a whole that were delegated to it, only a part of them is lost (it can be restored by the checksum).

The obtained results meet the current requirements for the functioning of the state's critical infrastructure, in particular, military-use IS in the context of their stability (continuity) and reliability.

Keywords: load balancing; cluster; game theory; Nash equilibrium; probability distribution; finding the optimum.