



УДК 004.22.021.023(045)

OPERATION ORGANIZING ALGORITHM OF DISTRIBUTED DSS UNDER HEAVY LOAD CONDITIONS

АЛГОРИТМ ОРГАНІЗАЦІЇ ФУНКЦІОНУВАННЯ РОЗПОДІЛЕНОЇ СППР В УМОВАХ ВЕЛИКОГО НАВАНТАЖЕННЯ

Gulak N.K. / Гулак Н.К.

с.т.с./к.т.н.

ORCID: 0000-0001-8524-8635

Dubchak E.V. / Дубчак О.В.

ORCID: 000-0001-9739-3960

National Aviation University, Kyiv, Guzara 1, 03058

Національний авіаційний університет, м. Київ, Л. Гузара 1, 03058

Анотація. Розроблено методи та алгоритми оптимальної організації функціонування розподіленої СППР, що вимагають планування роботи СППР в умовах великого навантаження, маршрутизацію обчислень та інформаційний обмін між локальними СППР. Обґрунтовано використання нейронних мереж Кохонена для розподіленої системи підтримки прийняття рішень, яка реалізує спосіб слабкого пов'язування локальних СППР.

Ключові слова: система підтримки прийняття рішень, теорія нечітких множин, розподілена СППР, евристичні системи, «Blackboard»-системи, нейронні мережі, мережа Кохонена

Вступ

Система підтримки прийняття рішень (СППР) у більшості випадків – це інтерактивна автоматизована система, яка допомагає споживачеві використовувати дані та моделі для виявлення та вирішення проблем і прийняття рішень.

Основне завдання під час прийняття рішення – обрати найкращий варіант для досягнення певної мети або ранжувати набір можливих варіантів за ступенем їхнього впливу на досягнення поставленої мети.

Основний текст

1. Планування роботи розподіленої СППР в умовах великого навантаження. В процесі функціонування розподіленої СППР, розосереджуючи розв'язання задачі по вузлах системи, не виключена одночасна поява декількох завдань (підзавдань), що потребують обслуговування. Тому необхідно встановити оптимальну, за деяким критерієм, послідовність (розклад) їх вирішення, тобто сфокусувати увагу на вирішенні того завдання, яке має найвищий пріоритет. У загальному випадку завдання складання розкладу формулюється в такий спосіб.

Нехай $T = \{t_i\}$ – множина завдань, що вимагають обслуговування, η_i – функції штрафу за перебування завдання t_i в системі. Необхідно знайти таку перестановку завдань Π , де $\Pi(k) = i$, якщо i -е завдання має бути вирішене k -им за рахунком, то сумарна функція повинна бути мінімальною, тобто

$$\sum_{i=1}^n \eta_i \rightarrow \min. \quad (1)$$



Завдання складання розкладів у постановці (1) відноситься до класу NP -повних, які розв'язуються досить важко. Враховуючи те, що такі завдання вирішуються часто в реальному масштабі часу, необхідно використовувати методи з найменшою обчислювальною складністю. Одним з таких методів є метод експертних оцінок нечітких множин [1], обчислювальна складність якого в гіршому випадку дорівнює $O(n^2)$. У цьому методі в якості показника ефективності (критерію вартості) розкладу використовується середній зважений час завершення завдань. [2]

Для формалізації задачі P складання розкладу введемо наступні позначення: $(F_j <)$ – система завдань, де F є індексована множина з n завдань, $n \geq 0$, а $<$ – відношення часткового порядку (відношення передування), задане на F ;

$I(P)$ – набір індексів F ;

$T_j, j \in I(P)$ – елемент F ;

$\tau_j, j \in I(P)$ – потреба завдання T_j в часі обслуговування;

$\omega_j, j \in I(P)$ – вартість перебування завдання T_j в системі (важливість завдання).

Величини τ_j і ω_j визначаються або як середні значення за наявності статистичних даних, або з використанням відповідних функцій приналежності, що були отриманні експертним шляхом.

Оскільки ми розглядаємо лише один процесор, то будемо розглядати розклади, які задаються перестановкою індексів завдань.

Перестановка $\alpha = \alpha_1, \alpha_2, \dots, \alpha_n$ елементів $I(P)$ сумісна з P , якщо $T_j < T_{j'}$ тоді $k < k'$, де $\alpha_k = j$ і $\alpha_{k'} = j'$.

Перестановка $\alpha = \alpha_1, \alpha_2, \dots, \alpha_n$ елементів $I(P)$ визначає розклад природнім чином: завдання T_{α_1} виконується першим, завдання T_{α_2} – другим і так далі. Середній зважений час завершення завдання для розкладу, який визначається перестановкою α , визначається таким чином:

$$\overline{T(\alpha)} = \sum_{j=1}^n \omega_{\alpha_j} \left(\sum_{i=1}^j \tau_{\alpha_i} \right). \quad (2)$$

Перестановка α (розклад) є оптимальною, якщо α сумісна з P і при цьому досягається мінімум $\overline{T(\alpha)}$ серед усіх перестановок, сумісних з P . Нехай $p_j = \frac{\omega_j}{\tau_j}$ для всіх $j \in I(P)$.

Теорема 1.1 [3]. Нехай P – завдання упорядкування. Перестановка $\alpha = \alpha_1, \alpha_2, \dots, \alpha_n$ оптимальна відносно до P тоді і тільки тоді, коли $p_{\alpha_1} \geq p_{\alpha_2} \geq \dots \geq p_{\alpha_n}$.

Наведена теорема по суті визначає зміст алгоритму знаходження такої перестановки.

Нехай $U \subseteq I(P)$ і $U \neq \emptyset$. Визначимо $\tau(U)$, $\omega(U)$ і $p(U)$ в такий спосіб:



$$\tau(U) = \sum_{j \in U} \tau_j, \quad \omega(U) = \sum_{j \in U} \omega_j, \quad p(U) = \frac{\omega(U)}{\tau(U)}. \quad (3)$$

Визначення 1.1. Набір $U \subseteq I(P)$ назвемо початковою множиною відносно до P , якщо:

- 1) $U \neq \emptyset$;
- 2) якщо $T_j < T_{j'}$, то $j \in U$ при цьому $j' \notin U$.

Нехай Y – клас початкових множин (відносно до P). Визначимо число $p^* = \max_{U \in Y} p(U)$.

Визначення 1.2. Множину $U \subseteq I(P)$ назвемо p -максимальною множиною відносно до P , якщо:

- 1) $U \in Y$;
- 2) $p(U) = p^*$;
- 3) якщо $V \in Y$; $p(V) = p^*$ і $V \subseteq U$, то $V = U$.

Нехай $U \subseteq I(P)$ і визначимо F/U як $F/U = \{T_j \mid T_j \in F, j \in U\}$.

Позначимо через H_j множину, що складається з індексу j та індексів i всіх завдань, для яких $T_i < T_j$. Побудова оптимальної перестановки в задачі P складання розкладу може бути здійснена наступним чином. Кожному завданню T_i ставиться у відповідність число $p(H_j)$, а потім визначається індекс l такий, що $p(H_l) \geq p(H_j)$ для $j \in I(P)$, причому $p(H_l) \geq p(H_j)$ для всіх індексів $j \neq l$. Оскільки T_l є попередником всіх завдань, індекси яких входять в H_l , то існує оптимальна щодо P перестановка α така, що $\alpha = \alpha/(H_l - \{l\}); \alpha/(I(P) - H_l)$. Таким чином, завдання P може бути "розкладене" на два менші, а саме: $P/(H_l - \{l\})$ і $P/(I(P) - H_l)$.

Алгоритм. Вхід: P – завдання складання розкладу. Вихід: оптимальний розклад для P .

```

procedure OPT(P)
begin
if P порожньо then return  $\lambda$  (порожня перестановка)
else
begin

```

Нехай l є індекс завдання з P такий, що H_l є p -максимальною множиною для P .

$$U := H_l - \{l\}; \quad V := I(P) - H_l.$$

```

return OPT(P/U), OPT(P/V)
end
end OPT.

```

Процедура OPT є рекурсивною, вона має в якості єдиного вхідного параметра задачу впорядкування; результатом її виконання являється оптимальна перестановка відносно до вхідного завдання. Виконання оператора return <вираз> полягає в обчисленні <вираз> і повернення в точку виклику



процедури з обчисленими значеннями в якості результату.

2. Маршрутизація обчислень в розподіленій СППР. Розподіл потоків інформації в мережі, утвореній локальними СППР, покладається на систему управління мережею. Задача розподілу полягає у виборі та встановленні оптимального шляху передачі інформації від джерела інформації до споживача з урахуванням ситуації у мережі. В [4] показано, що найкоротший шлях між вузлами i та j мережі є найбільш надійним.

В основу алгоритму маршрутизації покладемо модифікований алгоритм Флойда [5]. Нижче наведено структурну схему алгоритму (рисунки 1), яка включає наступні операції:

1. Вводиться розмірність даної мережі n .
2. Вводиться трикутна матриця вагових коефіцієнтів мережі $w(i, j)$.
3. Трикутна матриця вагових коефіцієнтів перетвориться в симетричну матрицю з нулями на головній діагоналі.
4. Будується квадратна матриця послідовно зростаючої розмірності $a(i, j)$, елементами якої є довжини найкоротших маршрутів, що з'єднують вузли i та j за умови, що проміжними вузлами цих маршрутів можуть бути тільки вузли, номер яких не перевищує розмірності побудованої матриці.
5. Будується квадратна матриця послідовно зростаючої розмірності $s(i, j)$, елементи якої - номери вузлів, записані в певному порядку, дозволяють відновити найкоротший маршрут, що з'єднує вузли i та j .

$$s(i, j) = \begin{cases} \text{номер першої проміжної вершини на поточному} \\ \text{маршруті від } i \text{ до } j, \text{ якщо така існує;} \\ j, \text{ якщо проміжних вершин не існує.} \end{cases}$$

6. Розглядаємо матрицю $a(i, j)$ розмірності 2.

$$a(1,2) = a(2,1) = w(1,2); \quad s(1,2) = s(2,1) = 1.$$

7. Ітерація по $k + 1$. Побудувавши матрицю $a(i, j)$ k -ї розмірності, матрицю $(k + 1)$ -ї розмірності будемо таким чином:

- а) в матриці вагових коефіцієнтів $w(i, j)$ переглядаємо $k + 1$ стовпець і запам'ятовуємо вагові коефіцієнти і номери вузлів, з яких існує маршрут у вузол $k + 1$: $s(p, j) = p$;
- б) обчислюємо вагові коефіцієнти можливих маршрутів $a(i, j) = a(i, p) + w(p, j)$ і вибираємо серед них маршрут з мінімальною вагою;
- в) заповнюємо $k + 1$ стовпець і $k + 1$ рядок матриці $a(i, j)$ отриманими ваговими коефіцієнтами.

8. $k := k + 2$.

9. Якщо $k = n + 1$, то зупиняємось, інакше - перехід на 7.

За допомогою цього алгоритму можна отримати матрицю найкоротших відстаней між вузлами мережі. Ця матриця зберігається в кожній локальній СППР і генерується кожен раз при зміні топології мережі.

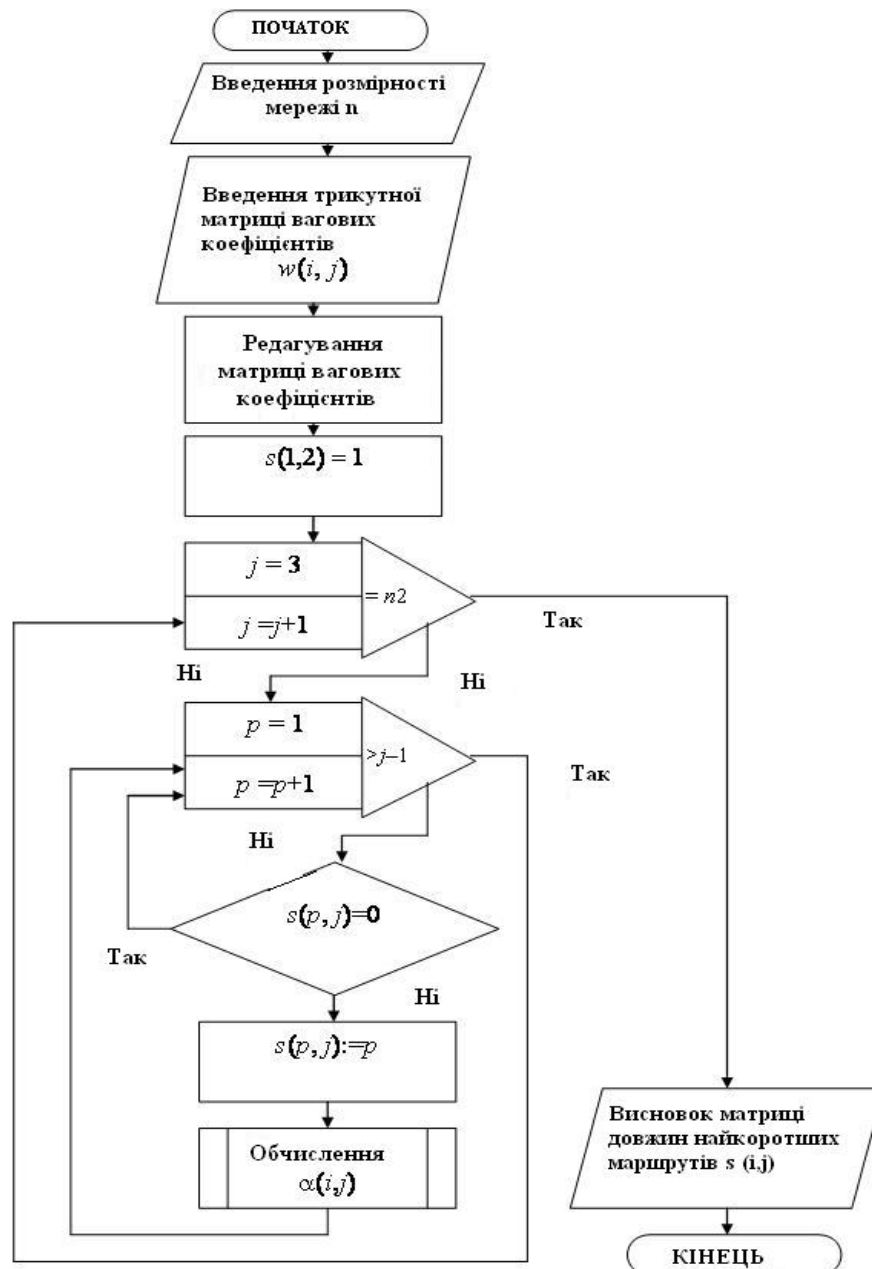


Рисунок 1 - Структурна схема алгоритму знаходження найкоротших маршрутів

3. Стратегія інформаційного обміну між локальними СППР. Для реалізації обміну даними між локальними СППР в процесі функціонування розподіленої СППР доцільно використовувати так звану «дошку оголошень» [6]. «Дошка оголошень» - це глобальна структура даних, призначена для здійснення обміну інформацією про факти, логічні висновки та цілі між різними локальними СППР, що утворюють вузли розподіленої СППР. Її можливості визначаються способом забезпечення загальної основи та інтерфейсу для спільного використання корисної інформації як всередині окремих локальних СППР, так і мережі взаємодіючих локальних систем.

При реалізації принципу «дошки оголошень» в повному обсязі вдається використовувати три способи подання. Перший з них представляється фактами та висновками. Факт - це певний елемент даних, що вводиться в систему



людиною або потрапляє в неї з вхідного інформаційного потоку. Факту може бути приписаний деякий показник достовірності відповідно до надійності джерела, проте сама система фактів не породжує. На противагу цьому, логічний висновок формується в результаті застосування тих чи інших знань до фактів і раніше зроблених висновків, зафіксованих у системі. Коефіцієнт довіри, приписуваний конкретному логічному висновку, визначається ступенем достовірності системних знань, на основі яких зроблено введення, та інших фактів і висновків, покладених в основу даного висновку.

Другий спосіб подання інформації - це керуюча інформація, яка може поділятися на джерела знань і цілі. Джерела знань - це процедури, які визначають, яким чином у системі повинні будуватися висновки та виконуватися конкретні дії при нормальному режимі функціонування. Цілі ж є не що інше, як подання стандартних дій, які можуть вживатися експертом. У деяких публікаціях для реалізації концепції цілей використовується спеціальна структура даних, яка ще називається планом дій. Необхідність в ній виникає тому, що локальні СППР мережі, як правило, мають численні цілі та повинні за деяким критерієм здійснювати на цій множині вибір цілі, який найбільшою мірою задовольняє конкретний момент часу.

Третій спосіб подання інформації при реалізації принципу «дошки оголошень» визначено розподіленою структурою системи. Це інформація, яка потрібна для управління інформаційними потоками, що надходять від «дошки оголошень» однієї локальної СППР до «дошок оголошень» інших вузлів мережі. У зв'язку з тим, що «дошка оголошень» - це ресурс, який грає найважливішу роль в процесі логічного висновку, у разі централізованої «дошки оголошень» навантаження ліній передачі даних виявилось б надмірно високим. Крім того, за наявності єдиної загальної «дошки оголошень» знижується надійність системи. Отже, кожна локальна СППР мережевої архітектури повинна мати власну локальну «дошку оголошень» для інформації, безпосередньо доступної даних СППР. Таким чином, концепція «дошки оголошень» - це зручний спосіб слабкого зв'язування різних СППР для забезпечення можливості вирішення конфліктів і автономного їх функціонування.

Для вибору інформації, яка повинна передаватися або прийматися на локальну «дошку оголошень» СППР, повинна бути передбачена відповідна стратегія інформаційного обміну. В якості такої стратегії пропонується використовувати стратегію розподіленого вирішення завдань [7]. Розглянемо алгоритм реалізації зазначеної стратегії.

Нехай задача S надійшла на деяку СППР Y_0 .

Крок 1. СППР Y_0 буде декомпозицією вихідної задачі S . Нехай у результаті декомпозиції задачі утворилося S_i підзадач, де $i = \overline{0, \dots, n}$.

Крок 2. СППР Y_0 перевіряє на суперечливість множину S_0 . Якщо S_0 несуперечливо, процес завершується - вихідна задача S_0 не має рішення. Інакше СППР Y_0 визначає дочірні СППР Y_i і пересилає їм завдання S_i ($i = \overline{1, \dots, n}$) для вирішення.



Крок 3. СППР Y_i ($i = \overline{1, \dots, n}$) перевіряють на суперечливість множини $S_i \cdot \Theta_0$ ($i = \overline{1, n}$), Θ_0 де – розв’язок задачі S_0 . Результати вирішення цих завдань СППР Y_i пересилаються в СППР Y_0 .

Крок 4. Якщо СППР Y_0 визначила, що всі перевірені множини суперечливі, то переходимо до кроку 5. Інакше вихідна задача S рішення не має.

Крок 5. У СППР Y_0 формується комбінація $\Theta_{1,n}$ рішень $\Theta_1, \Theta_2, \dots, \Theta_n$ і процес завершується $\Theta_{1,n}$ - рішенням вихідного завдання. Якщо комбінації Θ не існує, скажімо, через рішення множин $S_n \cdot \Theta_0$, то формується комбінація $\Theta_{1,n-1}$ рішень $\Theta_1, \Theta_2, \dots, \Theta_{n-1}$ і вона пересилається в СППР Y_n .

Крок 6. У СППР Y_n перевіряється на суперечливість множина $S_n \cdot \Theta_{n-1}$. Якщо ця множина суперечлива, то процес припиняється. Це рішення і буде рішенням вихідної задачі S . Інакше рішення Θ_n пересилається в СППР Y_i ($i = \overline{1, \dots, n-1}$).

Крок 7. У СППР Y_i ($i = \overline{1, \dots, n-1}$) перевіряються на суперечливість множини $S_i \cdot \Theta_n$. Якщо вони суперечливі, то кожній СППР Y_i пересилається рішення задачі $S_i \cdot \Theta_n$ у СППР Y_0 .

Крок 8. У СППР Y_0 формується комбінація Θ з рішень задач $S_i \cdot \Theta_n$ ($i = \overline{1, \dots, n-1}$) і завершується робота. Комбінація Θ – це вирішення завдання S .

Таким чином, концепція «дошки оголошень» - це зручний спосіб слабого зв'язування локальних СППР для забезпечення можливості вирішення конфліктів і автономного їх функціонування.

В умовах великого завантаження задачі з маршрутизації обчислень та інформаційний обмін між локальними СППР можуть бути вирішеними за допомогою нейронних мереж.

Оскільки представлена розподілена СППР має спосіб слабого пов'язування різних СППР для забезпечення можливості вирішення конфліктів та їх функціонування, то вона може бути реалізована слабо пов'язаними нейронними мережами (мережі Кохонена), які можуть вирішувати ці завдання [8].

У мережах Кохонена нейрони розташовуються у вузлі прямокутної (гексогональної) решітки (рисунок 2).

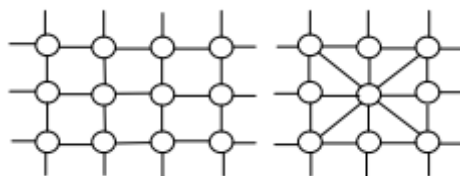


Рисунок 2 - Топологія мережі Кохонена

За класифікацією нейромережових структур такі мережі мають тип навчання - без вчителя.



Підсумки та висновки

1. В загальному вигляді сформульована задача синтезу оптимальної структури розподіленої СППР як задачі нелінійного математичного програмування. Показана доцільність застосування агрегативно-декомпозиційного підходу для вирішення сформульованої задачі.

2. Розроблено методи та алгоритми оптимальної організації функціонування розподіленої СППР, що вимагають планування роботи СППР в умовах великого завантаження, маршрутизацію обчислень та інформаційний обмін між локальними СППР.

3. На основі представленого алгоритму для способу слабкого пов'язування різних СППР запропоновано використання нейронних мереж з топологією мережі Кохонена, що дозволить обрати оптимальний варіант правильного рішення.

Література:

1. В.М. Дубовой, О.О. Ковалюк. Моделі прийняття рішень в управлінні розподіленими динамічними системами: Монографія – Вінниця: УНІВЕРСУМВінниця, 2008. – 186 с.

2. Бідюк П.І. Системи і методи підтримки прийняття рішень /П.І. Бідюк, О.Л. Тимошук, А.Є. Коваленко, Л.О. Коршевніук. – К.: КПІ ім. Ігоря Сікорського, 2022. – 610 с.

3. Глонь О.В., Дубовой В.М. Моделювання систем керування в умовах невизначеності: Монографія. – Вінниця: УНІВЕРСУМВінниця, 2004. – 170 с.

4. Виклюк Я.І. Моделювання складних систем / Я.І. Виклюк, Р.М. Камінський, В.В. Пасічник. – Львів: Новий Світ – 2000, 2020. – 400 с.

5. Глухов В.С. Дослідження і проектування комп'ютерних систем і мереж /В.С. Глухов, А.Т. Костик. – К.: Магнолія, 2006. – 2024. – 256 с.

6. Н.Р. Nii. Blackboard systems. The model of problem solving. AT Magazine, 2001, vol.7 p.p. 38-53.

7. Кузьміна Н.Ф. Аналіз основних характеристик розподіленої системи підтримки прийняття групових рішень / Н.Ф. Кузьміна // Сборник научных трудов SWorld. – Иваново: МАРКОВА АД. – 2013. – Выпуск 3. – Том 6. – С. 84–87.

8. Булгакова О.С. Методи та системи штучного інтелекту: теорія і практика /О.С. Булгакова, В.В. Зосімов, В.О. Поздєєв. – Миколаївський національний університет ім. В. О. Сухомлинського. – Херсон : Олді-плюс, 2020. – 356 с.

References

1. V.M. Dubovoy, & Ye.A. Kovalyuk. (2008). *Modeli prinyattya reshen' v upravlinni rozpodilenymy dynamichnymy sistemamy*. Vinnitsa: UNIVERSUM Vinnitsa [in Ukrainian].

2. Bidyuk P.I., Timoshchuk O.L. et al. (2022). *Sistemy i metody pidtrymky pryynyattya rishen'*. K.: KPI im. Igorya Sikors'koho [in Ukrainian].

3. Hlon' O.V. & Dubovoy V.M. (2004). *Modelyuvannya system keruvannya v umovakh nevyznachenosti*. Vinnitsa: UNIVERSUM Vinnitsa [in Ukrainian].

4. Vykyuk Ya.I., R.M. Kamins'kyu & V.V. Pasichnyk (2020). *Modelyuvannya skladnykh system*. L'viv: Novyy Svit-2000 [in Ukrainian].



5. Hlukhov V.S. & Kostyk A.T. (2024). *Doslidzhennya i proektuvannya komp'yuternykh system i merezh*. K.: Mahnoliya, 2006 [in Ukrainian].
6. H.P. Nii. Blackboard systems. The model of problem solving. AL Magazine, 2001, vol.7 p.p. 38-53.
7. Kuz'mina N.F. (2013). Analiz osnovnykh kharakterystyk rozpodilenoji systemy pidtrymky pryynyattya hrupovykh rishen' . *SWorld*, 3, vol .6, 84–87. [in Ukrainian].
8. Bulhakova O.S., Zosimov V.V. & Pozdyeyev V.O. (2020). *Metody ta systemy shtuchnoho intelektu: teoriya i praktyka*. Kherson: Oldi-plyus, [in Ukrainian].

Abstract. Introduction. *The decision support system (DSS) is an interactive automated system. It helps to use data and models to identify and solve problems, make decisions. The main task in decision making is to choose the best option to achieve a certain goal or to rank a set of possible options according to the degree of its impact on achieving the goals.*

Planning the operation of a distributed DSS under heavy load conditions. *Several maintenance tasks may arise during the DSS distribution process. The sequence (schedule) of its solution according to a certain criterion is provided.*

Routing of calculations in a distributed DSS. *The network management system determines the distribution of information flows in the network formed by local DSS. The modified Floyd algorithm is the basis for the information flow routing algorithm.*

Strategy for information exchange between local DSS. *Blackboard systems are used to implement data exchange between local DSS during the operation of a distributed DSS. A distributed problem solving strategy is proposed for information exchange.*

Neural networks are proposed to solve the problem of routing calculations and information exchange between local DSS under heavy load conditions.

Results and conclusions. *1. The problem of synthesizing the optimal structure of a distributed DSS is formulated in general form as a problem of mathematical nonlinear programming. The feasibility of using the aggregate-decomposition approach to solve the problem is shown. 2. Methods and algorithms for the optimal organization of the functioning of a distributed DSS, which require planning the operation of the DSS under heavy load conditions, routing calculations and exchange between local DSS, have been developed. 3. Neural networks with the topology of the Kohonen's network, based on the presented algorithm for the method of weakly linking various DSS, are proposed. The optimal option for the correct solution has been selected.*

Key words: *decision support system, fuzzy set theory, a distributed DSS, heuristic systems, Blackboard systems, neural networks, Kohonen's network*