УДК 004.056.5

[1]**Антощук, Світлана Григорівна**, д-р техніч. наук, професор, директор Інституту комп'ютерних систем, E-mail: asg@opu.ua, ORCID: 0000-0002-9346-145X**,** м. Одеса, Україна
[1]**Лобачов, Іван Михайлович,** аспірант каф. інформаційних систем, E-mail: lobachevmv@gmail.com, ORCID: 0000-0002-4859-304X, м. Одеса, Україна
[1]**Малерік, Роман Петрович**, каф. інформаційних систем, E-mail: r.maleryk@gmail.com, ORCID: orcid.org/0000-0001-7544-769X, м. Одеса, Україна
[1]Одеський національний політехнічний університет, пр. Шевченка, 1, Одеса, Україна, 65044

РОЗПОДІЛЕННЯ РЕСУРСІВ СЕНСОРНОЇ МЕРЕЖІ В УМОВАХ ГРАНИЧНИХ ОБЧИСЛЕНЬ

Анотація. *У статті представлено методику, яка дозволяє зменшити використання ресурсів бездротової сенсорної мережі шляхомзастосування спеціальної таблиці маршрутизації, яка містить дані про більш короткі маршрути від кожного вузла до суміжного вузла з урахуванням обчислювальної складності завдання, яке розв'язується. Побудова таблиці маршрутизації проводиться на етапі початкової установки параметрів і настройки мережі. З метою вибору кращого варіанту заповнення таблиці проведено пошук мінімального шляху на графі за допомогою алгоритму Дейкстри і евристичного алгоритму мурашиних колоній. Розроблено симулятор, який дозволяє досліджувати і порівнювати ефективність використання таблиці маршрутизації з випадковою маршрутизацією. Аналіз даних показав переваги застосування таблиць маршрутизації, що може застосовуватися до будь-яких мереж, які використовують периферійні обчислення, які передбачають інтелектуальну обробку даних на основі нейронних мереж.*

Ключові слова:*бездротова сенсорна мережа; граничні обчислення; розподілення ресурсів; таблиця маршрутизації; алгоритм мурашиної колонії; стимулятор*

Вступ

Широке розповсюдження розподілених сенсорних систем, а також бурхливий розвиток технології інтернету речей (Internet of Things – IoT) призвели до експоненціального зростання обсягів даних, які генерують сенсорні мережі та «розумні пристрої» різного призначення [1].

Експерти прогнозують, що до 2020 року у світі налічуватиметься близько 31 млрд «розумних» сенсорів та інших IoT-пристроїв, які будуть генерувати щонайменше 507,5 зеттабайт даних.

Ці тенденції призвели до того, що в останні кілька років традиційний спосіб обробки даних в хмарі перестав бути ефективним. Традиційна централізована обробка даних з сенсорних мереж вимагає великих фінансових та енергетичних витрат на підтримку каналу зв'язку з великою пропускною здатністю і впровадження заходів щодо захисту даних. Вона також не може забезпечити швидку реакцію системи на позаштатну ситуацію через тимчасові витрати на відправку даних та їх обробку. Особливо це відчувається при обробці відеоданих, які збираються розподіленою бездротовою сенсорною мережею (БСМ). Слід відмітити, що БСМ для збирання та аналізу відеоданих знаходять застосування в багатьох практичних застосуваннях, таких як системи відеоспостереження для забезпечення безпекиоб'єктів, організації технологічних процесів, екологічного моніторингу тощо.

© С. Г. Антощук; І. М. Лобачов;
Р. П. Малерік; 2018

Такі системи потребують негайного прийняття рішень у разі виникнення позаштатних ситуацій і є критичними до часу обробки. У зв'язку з цим, останні три роки стала стрімко набирає популярність технологія граничних або периферійних обчислень (edge computing). Ця технологія полягає в тому, що дані збираються та аналізуються якомога ближче до джерел генерації даних, в ідеалі – на тому самому пристрої або в тій же мережі. Далі система може прийняти рішення «на місці» або відправити оброблені дані в централізоване сховище для їх подальшої обробки або прийняття рішення.

Постановка проблеми

Таким чином, застосування технологій граничних обчислень дає цілий ряд переваг при побудові сенсорних мереж: зниження вимог до пропускної здатності каналу зв'язку, підвищення швидкості обробки, збільшення безпеки, але має один суттєвий недолік, пов'язаний із обмеженнями обчислювальної потужності «на місці» збору даних. Для його усунення застосовують багатоядерні прискорювачі штучного інтелекту заснований на VLIW-архітектурі (Very Long Instruction Word), з доповненими вузлами призначеними для обробки відео (наприклад, Intel Movidius Myriad 2) та забезпечують схеми раціонального розподілення ресурсів в БСМ.

This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/deed.uk)

DOI://10.15276/hait.01.2018.3
ISSN 2663-0176

Особливості маршрутизації в Ad-hoc мережах

Централізована обробка і аналіз даних дозволяє скористатися перевагою обчислювальної еластичності хмарних технологій – можливості оперативно регулювати обчислювальну потужність хмари для обробки даних. Використання технології граничних обчислень накладає суворі вимоги до проектування структури локальної мережі сенсорів і пристроїв, а також вимагає попереднього прорахунку необхідних обчислювальних ресурсів для обробки потоку даних, в залежності від завдання, що вирішується, конфігурації мережі та кількості пристроїв, які генерують дані.

Як правило, проектування структури сенсорних мереж здійснюється експертним шляхом, що не гарантує раціональну, з точки зору ресурсоємності, структуру мережі, так як вона проектується «з надлишковістю» для гарантування виконання поставлених завдань.

В рамках концепції IoT основою ефективного функціонування як кінцевих пристроїв, що володіють «розумними» функціями, так і утворених ними БСМ, стають робота в реальному часі, надійність і безпека. Цівимого не завжди можуть бути забезпечені при використанні схеми взаємодії «клієнт-сервер», характерної для «класичного» Інтернету, що базується на хмарних технологіях та обчисленнях (cloud computing). Для вирішення цих завдань була розроблена концепція граничних обчислень, яка покликана, в першу чергу, наблизити обробку і зберігання даних до пристроїв які їх генерують та використовують [1]. Потреба в таких обчисленнях виникає у багатьох галузях, включаючи виробництво, охорону здоров'я, телекомунікації та фінанси.

Виконання обчислень ближче до границі мережі дозволяє споживачам аналізувати важливі дані майже в режимі реального часу. Ці обчислення впорядковують дані локально, тому деякі з них обробляються безпосередньо на граничних пристроях, що зменшує трафік до центрального репозиторію.

Як правило, це роблять пристрої IoT, які переносять дані на локальний пристрій невеликого форм-фактору, який виконує обчислення, зберігання та підключення до мережі. Дані обробляються на границі, і весь об’єм або його частина відправляється у центральний репозиторій для обробки або зберігання в корпоративному центрі обробки даних (ЦОД), в майданчику спільного розміщення або хмарі. Граничні обчислення надають багато переваг при самих різних обставинах. По-перше, оскільки пристрої IoT мають поганий зв'язок і постійне підключення їх до центральної хмари є неефективним. По-друге, у разі, який пов'язаний з обробкою інформації, чутливої до затримок, граничні обчислення зменшують час очікування, тому що дані не повинні проходити по мережі в ЦОД або хмару для обробки. Це ідеально підходить для ситуацій, коли затримки навіть в мілісекундах можуть бути неприпустимі, наприклад, у фінансових послугах чи виробництві.

Як відомо, в БСМ комунікаційні функції вузлів здійснюються за принципами Ad-hoc мереж – децентралізованих бездротових мереж, які не мають постійної структури. Клієнтські пристрої з'єднуються «на льоту», утворюючи собою мережу [2].Кожен вузол мережі намагається переслати дані, призначені іншим вузлам. При цьому визначення того, якому вузлу пересилати дані, проводиться динамічно, на підставі зв'язності мережі. Це являється відмінністю від провідних мереж і керованих бездротових мереж, в яких завдання управління потоками даних виконують маршрутизатори (в провідних мережах) або точки доступу (в керованих бездротових мережах) [3-4].

Вибір алгоритму маршрутизації – одне з найскладніших питань, що вирішуються при проектуванні сенсорної мережі. По-перше, маршрутизація вимагає координації роботи всіх вузлів мережі. По-друге, алгоритм маршрутизації повинен справлятися з відмовами вузлів шляхом пере направлення трафіку і оновлення баз даних. По-третє, для досягнення найкращих результатів алгоритм маршрутизації повинен мати можливість змінювати маршрути при перевантаженні деяких областей мережі.

При розробці алгоритмів маршрутизації часто намагаються керуватися низкою факторів, таких як:оптимальність;простота і низькі витрати; живучість і стабільність; швидка збіжність; гнуч- кість.

Розрізняють три види маршрутизації – просту, фіксовану і адаптивну. Принципова різниця між ними полягає в ступені обліку зміни топології і навантаження мережі при вирішенні завдання вибору маршруту.

Проста маршрутизація відрізняється тим, що при виборі маршруту не враховується ні зміна топології мережі, ні зміна її стану (навантаження). Один з її видів – лавинна маршрутизація (або: заповнення пакетами всіх вільних вихідних напрямків) передбачає

передачу пакета з вузла по всім напрямкам, крім того звідки пакет надійшов в даний вузол.

Фіксована маршрутизація характеризується тим, що при виборі маршруту враховується зміна топології мережі і не враховується зміна її навантаження. Для кожного вузла призначення нап-рямку передачі вибирається по таблиці маршрутів (каталогу), яка визначає найкоротші шляхи. Фіксована маршрутизація застосовується в мережах з топологією, що мало змінюється і у випадку усталених потоків пакетів.

Адаптивною називається маршрутизація, при якій прийняття рішення про направлення передачі пакетів здійснюється з урахуванням зміни як топології, так і навантаження мережі. Існує кілька модифікацій адаптивної маршрутизації, що розрізняються тим, яка саме інформація використовується при виборі маршруту. Набули поширення такі модифікації як локальна, розподілена, централізована і гібридна маршрутизації [3-5].

Аналіз класичного підходу щодо побудови бездротових сенсорних мереж і структури систем моніторингу, які використовують таку мережі показав, що при організації обчислень, як правило, використовують системи, що працюють за принципами штучного інтелекту – нейромережеві системи. Існує два варіанти їх застосування: безпосереднє навчання нейронної мережі (НМ) на великому наборі даних, що одержані с сенсорів та використання попередньо навченої НМз її донавчанням для інтерпретації реальних даних [6-7].

Перший варіант вимагає досить великої обчислювальної потужності, тому його виконання делегується хмарним ресурсам, що є найбільш раціональним рішенням.

Другий варіант може бути реалізованим на кінцевому пристрої, а не в хмарі, що забезпечує виграш з точки зору затримок, витрат енергії та безпеки даних.

Таким чином, для організації граничних обчислень з використанням нейромережевої обробки необхідно вирішити дві актуальні задачі: розподілення ресурсів БСМ та адаптивного застосування нейронних мереж.

Метою статті є дослідження методики розподілення ресурсів при граничних обчисленнях у розподілених візуальних сенсорних мережах.

Формалізація задачі розподілення ресурсів БСМ

Для проведення дослідження методики розподілення ресурсів при граничних обчисленнях у розподілених візуальних бездротових сенсорних мережах розроблено програмний стимулятор, схема інформаційних потоків якого представлена на рис. 1.

Розглянемо представлення БСМ з організацією граничних обчислень (з обчислювачами) за допомогою теорії графів.

Рис. 1. Схема інформаційних потоків

Бездротова сенсорна мережа, що складається з *n* вузлів може бути представлена як множина вузлів:

$$\mathrm{H} = \{\mathrm{H}_1, \mathrm{H}_2, \dots, \mathrm{H}_\mathrm{n}\}, H = \{H_1, H_2, \dots, H_n\}, \quad (1)$$

де *n* – кількість вузлів.

Бездротова сенсорна мережа містить множину сенсорів:

$$S = \{S_1, S_2, \dots, S_k\}, \quad (2)$$

де *k* – кількість сенсорів.

Інформація з кожного сенсору оброблюється зі своєю періодичністю. Тобто частота появи пакетів для *k* сенсорів є множиною:

$$f = \{f_1, f_2, \dots, f_k\}, \quad (3)$$

де *k* – кількість сенсорів.

Пакети даних сенсорів оброблюються *р*обчислювачами:

$$C = \{C_1, C_2, \dots, C_p\}, \quad (4)$$

де *p* – кількість обчислювачів.

Інформація з кожного сенсора оброблюється за допомогою підготовленої нейромережевої моделі, час виконання операцій обробки з кожного сенсора є множиною:

$$t_s = \{t_{s1}, t_{s2}, \dots, t_{sk}\}, \quad (5)$$

де k – кількість сенсорів.

Так як мережа, що моделюється, має здатність делегувати обробку пакетів іншим сумісним вузлам, час обробки (5) можна представити у вигляді:

$$t_r = \{t_{r1}, t_{r2}, \dots, t_{rk}\}, \quad (6)$$

причому:

$$t_{ri} = t_{si} + 2 \cdot h \cdot q, \quad (7)$$

де *h* – кількість «стрибків» у мережі для передачі пакету (i = 1.. *k*);

q – час відправки пакета у мережі (постійна величина для даної мережі або розрахована з урахуванням розміру пакета та швидкості передачі по певному каналу зв'язку).

Отже кожен вузол мережі характеризується набором сенсорів S_i(2), частотою надходження даних f_i(3) та набором обчислювачів C_j(4) з часом обробки t_r (6):

$$H\{[\ S\{f\},\dots\ ],[\ C\{t_r\},\dots]\}.$$

Для раціонального розподілуресурсів в мережі у разі делегування обробки іншому вузлу необхідно мінімізувати витрати на пересилку пакетів між вузлами, тобто зменшити кількість «стрибків» h.

Для вирішення даної задачі пропонується застосовуватинаперед розраховану таблицю маршрутизації, яка містить найкоротші маршрути від кожного вузла до суміжного з ним вузлом з урахуванням завдання, що вирішується. При моделюванні вважається, що різні вузли мережі мають різні моделі нейронних мереж, тобтоне кожен вузол може прийняти делегований пакет даних для обробки, тобто необхідно враховувати тип обчислювального завдання.

Складання даної таблиці здійснюється на етапі початкової установки та налаштування мережі. Для заповнення змісту таблиці необхідно вирішення завдання пошуку мінімального шляху на графі [8].

Розглянемо створення таблиці маршрутизації на прикладі БСМ, граф якої представлений матрицею суміжності (табл. 1). Відмічені кольором клітинки, що мають можливість спільної обробки інформації, тобто делегування пакетів.

Таблиця 1. Матриця суміжності БСМ

	H0	H1	H2	H3	H4	H5	H6	H7
H0	0	1	1	1	0	0	0	**0**
H1	1	0	1	**0**	1	0	0	0
H2	1	1	0	1	**1**	1	**0**	0
H3	1	**0**	1	0	0	**1**	1	0
H4	0	1	**1**	0	0	1	**0**	1
H5	0	**0**	1	**1**	1	0	1	1
H6	0	0	**0**	1	**0**	1	0	1
H7	**0**	0	0	0	1	1	1	0

Проведено порівняльний аналіз двох варіантів пошуку: за допомогою алгоритму Дейкстри та алгоритму мурашиної колонії.

1. Пошук мінімальних шляхів за алгоритмом Дейкстри, який знаходить маршрути та їхню довжину між початковим вузлом та усіма іншими вузлами графа. Цей алгоритм пошагово перебирає усі вузли графа та призначає їм мітки, які є відомою мінімальною відстанню від початкового вузла до поточного [9].

Результати розрахунку мінімальних шляхів, які одержані за алгоритмом Дейкстри наведеноу таблиці 2.

Таблиця 2. Матриця суміжності (алгоритмом Дейкстри)

	H0	H1	H2	H3	H4	H5	H6	H7
H0	0	1	0	1	0	0	0	0
H1	1	0	0	0	1	0	0	0
H2	0	0	0	1	1	0	0	0
H3	1	0	1	0	0	1	1	0
H4	0	1	1	0	0	0	0	1
H5	0	0	0	1	0	0	0	0
H6	0	0	0	1	0	0	0	0
H7	0	0	0	0	1	0	0	0

2. Пошук мінімальних шляхів за алгоритмом мурашиної колонії.Основна ідея цього алгоритму ґрунтуються на принципах поведінки колонії біологічних мурах – колективна поведінка біологічних мурах забезпечує знаходження найкоротшого шляху. На початку кожної ітерації у всіх пунктах мережі розміщується по одній мурасі. Для вибору подальшого напряму руху на маршруті складається список з індексів невідвіданих пунктів та визначається наступний пункт маршруту. Коли усі пункти відвідано, мураха повертається у свій початковий пункт. Обчислюються довжини маршрутів, отриманих всіма мурахами та вибирається найкоротший. Він порівнюється з поточним значенням довжини найкоротшого на момент ітерації і маршруту і якщо він є меншим, то знайдений розв'язок вважається новим рекордом (найкоротшим маршрутом, відомим на ітерації і) [10-11].

При реалізації алгоритму використовуючи наступний набір вхідних параметрів алгоритму:

- кількість мурах: 5;
- кількість ітерацій: 10;
- коефіцієнт вивітрювання феромону: 0,95;
- величина «жадібності» алгоритму: 0,3;

– величина «стадності» алгоритму: 0,2.

Результати розрахунку мінімальних шляхів, які одержані за алгоритмом мурашиної колонії наведено у табл. 3.

Проведено аналіз одержаних результатів (табл. 4). Для даної тестової конфігурації мережі, коли кількість вузлів невелика, алгоритм Дейкстри дозволяє згенерувати таблицю швидше, ніж алгоритмом мурашиної колонії. Але беручи до уваги обчислювальну складність алгоритму Дейкстри, при великій кількості вузлів, мурашиний алгоритм «справляється» швидше.

Також перевагою є те, що використання алгоритму мурашиної колонії дозволяє отримати декілька запасних маршрутів (з меншою концентрацією феромону), що будуть використовуватися у разі недосяжності вузла із основного маршруту.

Таблиця 3. Матриця суміжності нового графу

	H0	H 1	H 2	H3	H 4	H 5	H6	H 7
H0	0	1	0	1	0	0	0	0
H1	1	0	0	0	0	0	0	0
H2	0	0	0	0	1	1	0	0
H3	1	0	0	0	0	1	1	0
H4	0	0	1	0	0	0	0	0
H5	0	0	1	1	0	0	1	0
H6	0	0	0	1	0	1	0	**1**
H7	0	0	0	0	0	0	1	0

Таблиця 4. Порівняння методів генерації таблиці маршрутизації

Метод	Цикломатичнечисло	Структурна надмірність	Час генерації (мс)
Алгоритм Дейкстри	1	0,14	52
Мурашиний алгоритм	1	0,14	260

Симулятор передбачає використання таблиці маршрутизації та широкомовну розсилку для спілкування зі сусідніми вузлами. Вигляд інтерфейсу користувача наведено на рис. 2.

Під час моделювання симулятор розраховує наступні показники:

– середній час очікування пакета в рамках одного вузла та всієї мережі;

– кількість пакетів у черзі;

– кількість отриманих та відправленим пакетів для віддаленої обробки.

Проведено порівняння випадкової маршрутизації (лавинної) та за допомогою таблиці маршрутизації.

Початковими даними є ресурси мережі (табл. 5): кількість сенсорів та обчислювачів для кожного вузла. Наприклад, в мережі є три типи сенсорів: S1, S2, S3, для обробки інформації з кожного з них застосовується навчена нейромережева модель.

Рис. 2. Інтерфейс користувача

Таблиця 5. Структура та ресурси мережі

Вузол	Набір сенсорів	Кількість обчислювачів	Шлях (для кожного з сенсорів)
H0	1) S1 2) S2 3) S1	2	1) H0 – H3 – H5 – H7 2) H0 – H1 3) H0 – H3 – H5 – H7
H1	1) S2 2) S2 3) S2 4) S3	4	1) H1 – H0 – H3 2) H1 – H0 – H3 3) H1 – H0 – H3 4) H1 – H2
H2	1) S2 2) S3 3) S3	4	1) H2 – H1 2) H2 – H4 3) H2 – H4
H3	1) S2	1	1) H3 – H5
H4	1) S3 2) S2 3) S1	2	1) H4 – H2 2) H4 – H1 3) H4 – H7
H5	1) S2 2) S2 3) S2	3	1) H5 – H3 2) H5 – H3 3) H5 – H3
H6	1) S3	2	1) H6 – H5 – H2

H7	1) S1 2) S3 3) S3 4) S2	2	1) H7 – H4 – H2 – H0 2) H7 – H4 3) H7 – H4 4) H7 – H5

Навчені НМ моделі однакові на всіх вузлах, які мають сенсори даного типу. Мінімальні шляхи до вузлів, яким будуть делегуватися пакети, були визначені за допомогою алгоритму мурашиної колонії.

Розроблений симулятор дозволяє завантажити модель БСМ з довільною структурою та заданими ресурсами (сенсорами та обчислювачами). Також він дозволяє використовувати таблицю маршрутизації для делегування обробки пакетів або випадковий вибір вузла. Отже, для порівняння результатів цих двох випадків буде проведений експеримент. Для побудови графіків, дані знімалися один раз в секунду. Час одного експерименту становив 120 секунд. Для спрощення проведення експерименту, всі сенсори генерують пакет з даними кожну 1 с, а обчислювач оброблює пакет за 600 мс.

В результаті моделювання одержані наступні залежності:

1. Графік зміни кількості пакетів у черзі для кожного вузла (рис. 3). Середнє значення кількості пакетів у черзі для всіх вузлів складає 2,47 пакетів. Це на 20 % менше в порівняні з випадковою маршрутизацією.

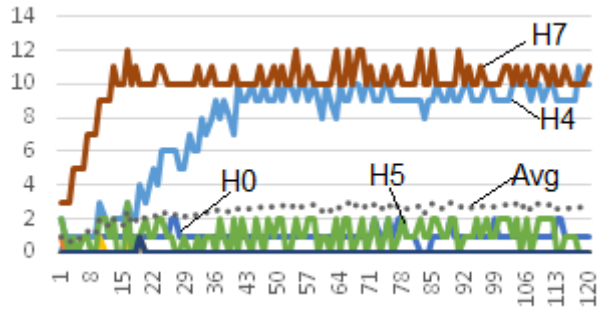

Рис. 3. Графік зміни черги для табличної маршрутизації

2. Графік зміни часу очікування пакету для кожного вузла з використанням таблиці маршрутизації (рис. 4). Середнє значення – 678 мс, що також є меншим ніж при випадковій маршрутизації (782 мс).

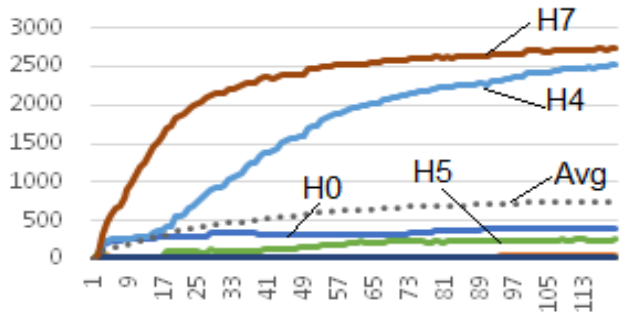

Рис. 4. Графік зміни часу очікування для табличної маршрутизації

Слід відмітити, що перевагою використання таблиць маршрутизації у порівнянні з випадковою маршрутизацією є вагоме зменшення кількості «стрибків» у мережі з 298 до 100 відповідно, що істотно дозволить зменшити витрати енергії.

Висновки й перспективи подальших досліджень

Проведено моделювання роботи сенсорної мережі із заданою структурою та ресурсами за допомогою створеного симулятора. У ході моделювання одержані дані про стан черги (кількість пакетів), середній час очікування в черзі на кожному вузлі та в цілому в мережі та кількість «стрибків» між вузлами мережі.

Одержані результати показали, що застосування таблиць маршрутизації при відправці делегованих пакетів інформації дозволила незначно підвищити оперативність (до 20 %) та значно зменшити кількість «стрибків», тобто знизити витрати енергії на відправку та прийом пакетів.

Таким чином, для раціонального розподілення ресурсів сенсорної мережі для організації граничних обчислень доцільно використовувати наперед розраховані таблиці маршрутизації.

Список використаної літератури

1. К вопросу о туманных вычислениях и интернете вещей – [Електронний ресурс] – Режим доступу: https://cyberleninka.ru/article/n/k-voprosu-o-tuma nnyh-vychisleniyah-i-internete-veschey.

2. Свойства и характеристики Ad Hoc сетей – [Електронний ресурс] – Режим доступу: https://moluch.ru/archive/115/31245/.

3. Основы маршрутизации – [Електронний ресурс] – Режим доступу: http://citforum.ru/nets/ito/2.shtml.

4. Концепция организации беспроводных сенсорных сетей и их применение – [Електронний ресурс] – Режим доступу: https://elibrary.ru/item.asp?id= 18486138.

5. Структурно – топологические характеристики систем. – [Електронний ресурс] – Режим доступу: https://intellect.ml/strukturno-topologicheskie-kharakteristiki-sistem-strukturnaya-izbytochnost-6241.

6. Малерик Р. П., Филягин Д. Г., Лобачев И. М. Разработка методики построения динамических систем для внесерверной обработки данных [«Сучасні Інформаційні Технології 2018»], (Одеса, 22-24 травня, 2018). – Одеса : 2018 – С. 166-168.

7. Малерик Р. П. Integration of Neural Networks Into Smart Sensor Networks // Р. П. Малерик, Д. Г. Филягин, И. М. Лобачев, С. Г. Антощук, М. В. Лобачев [“2018 IEEE 9-th International Conference on Dependable Systems, Services and Technologies”], (Київ, 24-27 травня, 2018). – Київ : – 2018. – С. 578-582.

8. Характеристические числа графов – [Електронний ресурс] – Режим доступу: https://www.intuit.ru/studies/courses/650/506/lecture/11521.

9. Лебедев С. С., Новиков Ф. А. Необходимое и достаточное условие применимости алгоритма Дейкстры / С. С. Лебедев, Ф. А. Новиков // Компьютерные инструменты в образовании. – 2017. – № 4. – С. 5-13.

10. Штовба С. Д. Муравьиные алгоритмы: теория и применение [Текст] / С. Д. Штовба // Программирование. – 2005. – № 4. – С. 1-16.

11. Штовба С. Д. Муравьиные алгоритмы // Exponenta Pro. Математика в приложениях. – 2003. – № 4. – С. 70-75.

Отримано 26.11.2018

References

1. “On the issue of fog computing and the Internet of Things”. – [Electronic resource] – Available at: https:// https://cyberleninka.ru/article/n/k-voprosu-o-tuma nnyh-vychisleniyah-i-internete-veschey.

2. “Properties and characteristics of Ad Hoc networks”. – [Electronic resource] – Available at : https://moluch.ru/archive/115/31245/.

3. “Basics of routing”. – [Electronic resource] – Available at : http://citforum.ru/nets/ito/2.shtml

4. “The concept of the organization of wireless sensor networks and their application”. – [Electronic resource]. – Availble at : https://elibrary.ru/item.asp?id= 18486138.

5. “Structural – topological characteristics of systems”. – [Electronic resource] – Availble at : https://intellect.ml/strukturno-topologicheskie-kharakteristiki-sistem-strukturnaya-izbytochnost-6241.

6. Malerik, R. P., Filyagin, D. G., & Lobachev, I. M. (2018). Razrabotka metodov postroenija dinamicheskih sistem dlja obrabotki dannyh vne server [Development of methods for constructing dynamic systems for off-server data processing], “Modern Information Technologies 2018”, (Odessa, Grass 22-24, 2018), Odessa, Ukraine, 2018, pp. 166-168 (in Russian).

7. Malerik, R. P., Filyagin, D. G., Lobachev I. M., Antoshchuk, S. G., & Lobachev, M. V. (2018). “Integration of Neural Networks Into Smart Sensor Networks”– 2018 IEEE 9-th International Conference on Dependable Systems, Services and Technologies, (Kyiv, Grass 24-27, 2018), Kyiv, Ukraine, pp. 578-582.

8. “Characteristic numbers of graphs”. – [Electronic resource] – Availble at : https://www . intuit.ru/studies/courses/650/506/lecture/11521.

9. Lebedev, S. C., & Novikov, F. A. (2017). Neobhodimoe i dostatochnoe uslovie primenimosti algoritma Dejkstry [Necessary and sufficient condition for the applicability of the Dijkstra algorithm], *Computer tools in education Publ.*, No. 4, pp. 5-13 (in Russian).

10. Shtovba, S. D. (2005). Algoritmy optimizacii kolonii murav'ev: teorija i primenenie [Ant colony optimization algorithms: theory and application], *Programming Publ*., No. 4. pp. 1-16 (in Russian).

11. Shtovba, S. D. (2003). Algoritmy optimizacii kolonii [Ant colony optimization algorithms], Exponenta Pro. Mathematics in Applications, No. 4, pp. 70-75 (in Russian).

Svitlana G. Antoshchuk[1]**,** Doctor of Technical Sciences, Professor, Director of the Computer Systems Institute, E-Mail: asg@opu.ua, ORCID : 0000-0002-9346-145X**,** Odessa, Ukraine
Ivan M. Lobachev[1]**,** postgraduate student Department of the Information Systems**,**
E-Mail: lobachevmv@gmail.com, ORCID: 0000-0002-4859-304X, Odessa, Ukraine
Roman P. Maleryk[1], Department of the Information Systems**,** E-Mail: r.maleryk@gmail.com,
ORCID: orcid.org/0000-0001-7544-769X, Odessa, Ukraine

[1]Odessa National Polytechnic University, Shevchenko Ave., 1, Odessa, Ukraine, 65044

METHOD OF THE SENSOR NETWORK RESOURCES ALLOCATION IN THE CONDITIONS OF EDGE COMPUTING

Abstract. *The article describes the developed methodology, which allows to reduce the resource usage of the wireless sensor network by reducing the cost of sending data using a special routing table. The simulator was developed that allows you to explore and compare the use of routing table and random routing. The analysis of the data showed the advantages of using the established methodology, which can be applied to any networks that use edge computing and have intelligent data processing.*

Keywords: *wireless touch screen; boundary calculations; resource allocation; routing table; algorithm of an ant colony; stimulator*

[1]**Антощук, Светлана Григорьевна,** д-р технич. наук, профессор, директор Института компьютерных систем, E-mail: asg@opu.ua**,**ORCID: 0000-0002-9346-145X**,** г. Одесса, Украина, 65044
[1]**Лобачев, Иван Михайлович,** аспирант каф. информационных систем, E-mail:lobachevmv@gmail.com, ORCID: 0000-0002-4859-304X, г. Одесса, Украина, 65044
[1]**Малерик, Роман Петрович**, каф. информационных систем, E-mail: r.maleryk@gmail.com, ORCID: orcid.org/0000-0001-7544-769X, г. Одесса, Украина, 65044
[1]Одесский национальный политехнический университет, пр-т Шевченко, 1, г. Одесса, Украина, 65044

РАСПРЕДЕЛЕНИЕ РЕСУРСОВ СЕНСОРНОЙ СЕТИ В УСЛОВИЯХ ГРАНИЧНЫХ ВЫЧИСЛЕНИЙ

Аннотация. *В статье представлена методика, которая позволяет уменьшить использование ресурсов беспроводной сенсорной сети путем применения специальной таблицы маршрутизации, которая содержит данные о более коротких маршруты от каждого узла до смежного узла с учетом вычислительной сложности и особенностей решаемой задачи. Построение таблицы маршрутизации проводится на этапе начальной установки параметров и настройки сети. С целью выбора лучшего варианта заполнения таблицы проведен поиск минимального пути на графе с помощью алгоритма Дейкстры и эвристического алгоритма муравьиных колоний. Разработан симулятор, который позволяет исследовать и сравнивать эффективность использования таблицы маршрутизации со случайной маршрутизацией. Анализ показал преимущества применения таблиц маршрутизации. Такая методика может применяться к любым сетям, которые используют периферийные вычисления, которые предусматривают интеллектуальную обработку данных на основе нейронных сетей.*

Ключевые слова: *беспроводная сенсорная сеть; предельные вычисления; распределения ресурсов; таблица маршрутизации; алгоритм муравьиной колонии; стимулятор*