

УДК 629.544:004.77

## АВТОМАТИЗОВАНА СИСТЕМА ОБЛІКУ РОЗВАНТАЖЕННЯ СУДЕН-КОНТЕЙНЕРОВОЗІВ З ВИКОРИСТАННЯМ МОБІЛЬНИХ ПРИСТРОЇВ

**Журавська І. М.**, канд. техн. наук, докторант, доцент кафедри комп'ютерної інженерії Чорноморського національного університету ім. Петра Могили, Україна, e-mail: irina.zhuravska@chmnu.edu.ua, ORCID: 0000-0002-8102-9854;

**Мусієнко М. П.**, д-р техн. наук, професор, декан факультету комп'ютерних наук Чорноморського національного університету ім. Петра Могили, Україна, e-mail: musienko2001@ukr.net, ORCID: 0000-0001-9228-2233;

**Кулаковська І. В.**, канд. фіз.-мат. наук, в.о. завідувача кафедри інтелектуальних інформаційних систем Чорноморського національного університету ім. Петра Могили, Україна, e-mail: kulaknic@gmail.com, ORCID: 0000-0002-8432-1850

*Незважаючи на зростаючу комп'ютеризацію водних транспортних засобів, наразі процес обліку завантаження/розвантаження контейнеровозів здійснюється вручну тальманами. При такому порядку у капітана судна немає поточних даних щодо кількості розвантажених/завантажених контейнерів у кожній секції контейнеровозу. В роботі запропонований метод реєстрації контейнерів, що перенесені через борт судна портокраном, за допомогою мобільних пристроїв, які розміщуються на даху одного або декількох контейнерів у кожній секції. При відсутності у секції необхідної для обліку контейнерів кількості мобільних пристроїв, додаткові розносяться квадрокоптерами. Інформація капітану передається через розгорнуту на контейнеровозі локальну WiFi-мережу, навантаженість якої була оцінена шляхом математичного моделювання зазначеної мережі за допомогою зв'язного графу. Реєстрація події розвантаження/завантаження кожної секції визначається зміною стану не менше 20% пікселів матриці фотокамери мобільного пристрою за результатами аналізу траєкторії зображення.*

**Ключові слова:** судно-контейнеровоз, квадрокоптер, локальна WiFi-мережа, фотокамера смартфона, навантаженість мережі, зв'язний граф.

**Вступ.** Незважаючи на зростаючу комп'ютеризацію водних транспортних перевезень, наразі процес обліку завантаження/розвантаження відбувається вручну працівником вантажного комплексу – тальманом, який є робітником стивідорної компанії і надає інформацію про стан вантажу капітану судна тільки у кінці зміни у вигляді звичайного звіту, який називається тальманською розпискою [1].

Компанії, які займаються перевезенням вантажу у контейнерах, несуть відповідальність за його збереження. Тому екіпаж має забезпечити необхідну остійність судна, контролювати його крен та диферент. Зважаючи на це, капітан має потребу у отриманні детальної інформації щодо завантаження/розвантаження судна у реальному часі, під час стоянки у порту [2].

**Метою роботи** є автоматизація обліку процесу розвантаження і завантаження контейнеровозів для економії часу і ресурсів екіпажу, виключення похибок обрахунків, зумовлених людським фактором. Для досягнення мети необхідно вирішити **задачу** надання капітанові судна у режимі реального часу на ноутбук/планшет або інший пристрій інформації про кількість вантажу у кожній секції контейнерів на судні. Визначити позицію кожного контейнера на судні можливо за допомогою трьох координат: *Bay* – секція, *Row* – ряд і *Tier* – ярус (рис. 1).



Рисунок 1 – Нотація обліку розташування контейнерів на палубі

**Об'єктом дослідження** є локальна мережа Wi-Fi, через яку вищезазначений пристрій у капітана повинен приймати дані від великої кількості пристроїв-реєстраторів, розміщених на судні та поєднаних у локальну мережу (рис. 2). Навантаженість каналів передачі даних у цій мережі та режими роботи таких пристроїв-реєстраторів і є предметом дослідження.

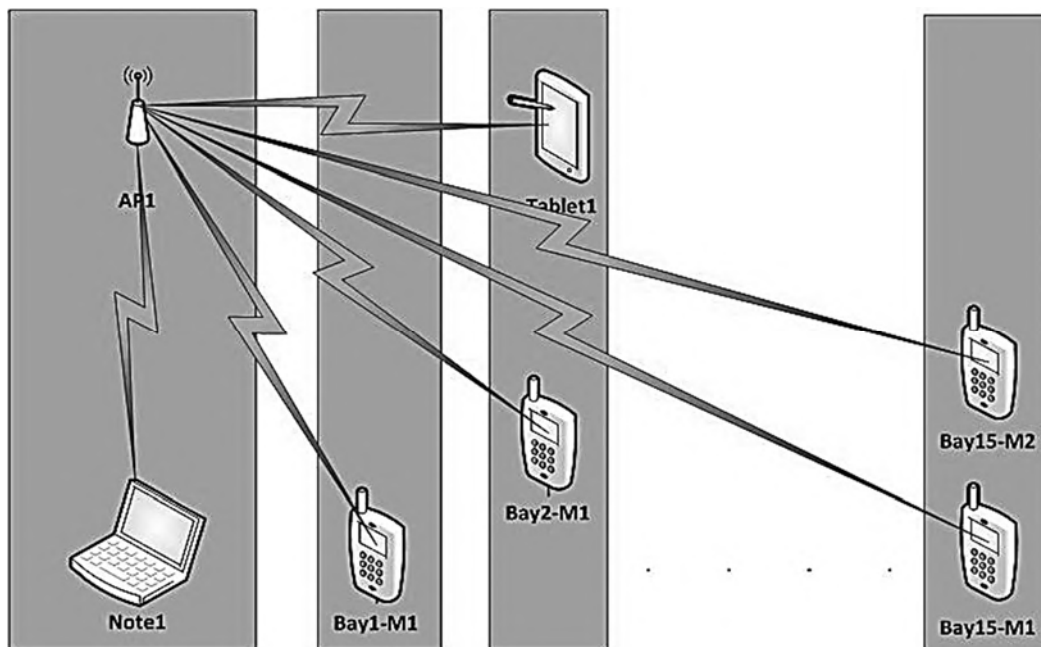


Рисунок 2 – Структурна схема локальної WiFi- мережі контейнеровозу

Нещодавно створено обчислювальний метод виявлення об'єктів, що рухаються, на серії CCD-кадрів, але він передбачає тільки малі рухи біля положення об'єкта [3]. Існують також системи, які передбачають для розгортання подібної системи обліку створення бездротового маршруту передачі інформації «судно – вантажний комплекс» [5, 6].

Але, на жаль, підключення судна у WiFi-мережу порту не завжди можливо за низкою технічних обставин (наприклад, металеві конструкції виробничих терміналів). У такому разі доцільно запропонувати рішення, за яким облікова система контейнеровозу працюватиме у власній локальній мережі та яка б не потребувала ані підключення до портових серверів, ані підключення до Інтернет.

Для впровадження висловленої ідеї на даху контейнерів у кожній секції ряду на палубі, найближчого до причалу, запропоновано розмістити мобільний пристрій на ОС Android, який реєструє контейнер, коли стріла портокрана переносить його над фотокамерою девайсу (рис. 3).

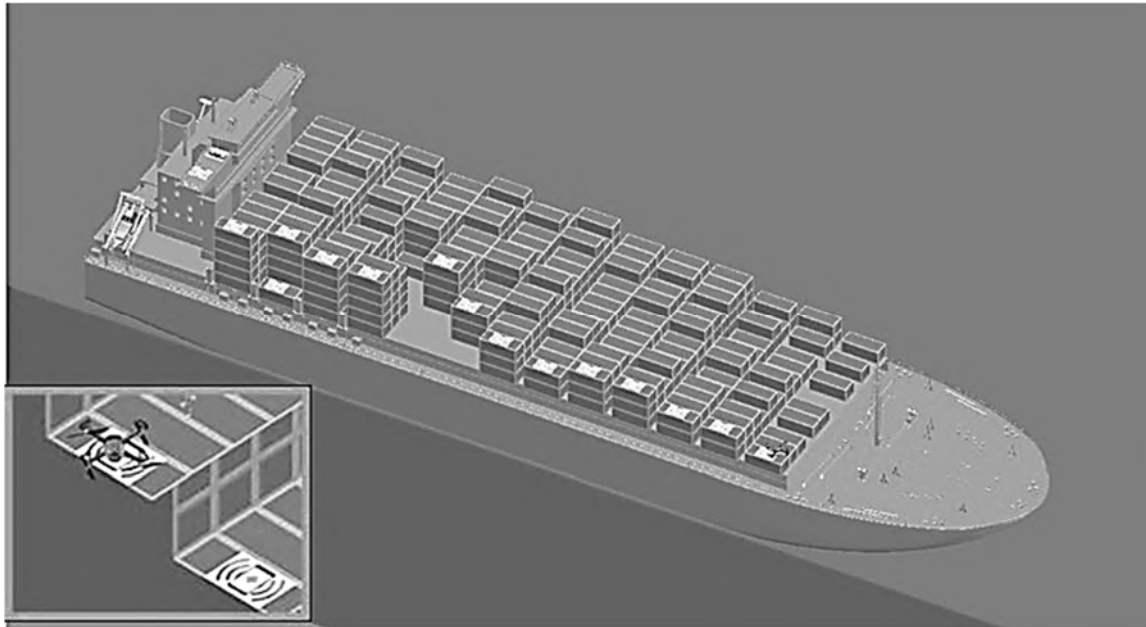


Рисунок 3 – Загальний вигляд судна-контейнеровозу з WiFi-реєстраторами

Для підвищення ефективності експлуатації контейнеровозів кількість ярусів контейнерів на палубі може бути до 8–9 шт. У такому разі, при нерівномірному розвантаженні контейнерів у рядах, в кожній секції необхідно розмістити мобільні пристрої на даху найвищого ярусу та на декількох ярусах проміжної висоти до мінімального.

**Методами дослідження** є сукупність методів встановлення параметрів фотокамер мобільних пристроїв таким чином, щоб їх лічильники спрацьовували в момент проходження стріли портокрана з контейнером, що за/вивантажується, та змінені показання лічильників передавалися на пристрій капітана через бездротову локальну мережу судна-контейнеровозу.

**Основні результати дослідження.** Значення лічильника кадрів одразу після зміни стану пристрою, тобто фіксації руху, передається через локальну WiFi-мережу судна на пристрій капітана, де відображається зменшення кількості контейнерів у відповідній секції (рис. 4).

Коли мобільний пристрій разом з контейнером, на якому він розміщений, забирається портокраном, квадрокоптер приносить новий пристрій на найближчий до причалу контейнер у секції, де розташовувався попередній контейнер (рис. 3). Дрон переміщується виключно над судном, не заходячи на територію порту, що узгоджується зі стивідорною компанією, яка обслуговує судно-контейнеровоз у цьому порту.

Для реалізації описаного методу було розроблено програмне забезпечення (ПЗ) під керуванням ОС Android [6]. Слід підкреслити, для задач реєстрації руху та передачі даних в описаній бездротовій моніторинговій локальній мережі достатньо мінімальних технічних характеристик і ресурсу акумулятора мобільних пристроїв. Так, досліджувані мобільні пристрої дали позитивні результати у процесі реєстрації великого вантажу над

пристроєм вже на рівнях API Level 14–17, які відповідають версіям платформи Android 4.0–4.2 [7].

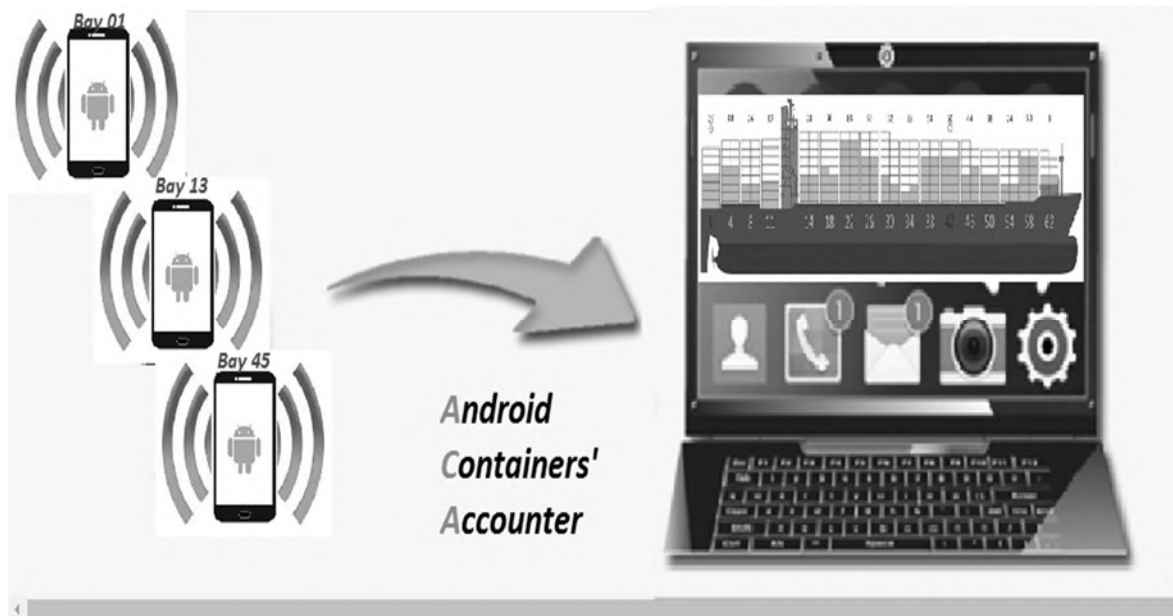


Рисунок 4 – Загальний вигляд графічного інтерфейсу на пристрої капітана

На даному етапі програма пристрою-реєстратора контейнерів працює за наступним алгоритмом.

Екран пристрою при працюючому додатку стає максимально бляклим, більш бляклим, аніж це можливо зробити за допомогою кареток, що є елементами меню пристрою, а також виставлений у режим FLAG\_KEEP\_SCREEN\_ON, що не дає екрану гаснути навіть після тривалої взаємодії з тач-скріном.

Оскільки оперативна пам'ять (RAM – Random Access Memory) у продуктивній версії системи пристроїв припускається якомога низька (до 512 Мбайт), повинна бути забезпечена висока продуктивність обробки зображень. Цього було досягнуто шляхом використання найменшого можливого розширення, що виставляється для кожного пристрою програмним шляхом при першому запуску пристрою. Це неухильно означає втрату якості зображення, проте вона є не першорядною для даної системи.

У подальших версіях через виникнення можливої потреби у вищому розширенні зображень (наприклад, для кешування), можлива модифікація не у бік погіршення якості зображення, а розпаралелювання процесу обробки даних між потоками виконання.

Щоб фіксація руху предметів перед камерою була адекватною та об'єктивною, потрібно виключити вплив руху самого пристрою. Оскільки, навіть уживані Android-пристрої мають акселерометр, бо мають можливість перевертання екрана при повороті пристрою, він є єдиним точним та чутливим сенсором, потрібним для встановлення режиму «У русі» для пристрою.

Підписавши слухача у вигляді SensorActivity, на зміну сенсорів пристрою у методі onSensorChange(SensorEvent event) відбувається обробка даних, отриманих із event-об'єкта, що містить ці дані, зміна котрих і спричинила виклик методу.

Акселерометр повертає масив, що містить три значення у  $m/c^2$  (усі значення у системі одиниць SI):

- values[0]: Прискорення мінус  $G_x$  по осі X;
- values[1]: Прискорення мінус  $G_y$  на осі Y;
- values[2]: Прискорення мінус  $G_z$  на осі Z.

Датчик цього типу вимірює прискорення, що діє на пристрій Ad. Він робить це шляхом вимірювання сил, що діють на сам датчик  $F_s$ , використовуючи співвідношення (1):

$$A_d = - \sum F_s / \text{mass}. \quad (1)$$

Зокрема, сила тяжіння завжди впливає на вимірне прискорення:

$$A_d = - g - \sum F / \text{mass}. \quad (2)$$

Через цю причину, коли пристрій лежить нерухомо та ймовірно не прискорюється, акселерометр зчитує величину  $G = 9,81 \text{ м/с}^2$ .

Точно так, коли пристрій знаходиться у стані вільного падіння, а отже змінює прискорення у напрямку до землі на  $9,81 \text{ м/с}^2$ , його акселерометр зчитує величину  $0 \text{ м/с}^2$ .

Шляхом проведення експериментів було встановлено, що приблизне значення, при якому значення прискорення змінюється хоча б на одній з осей, є «1» (одиниця). Тобто, якщо різниця минулого зафіксованого будь-якого із осевих значень є більшою за «1», то фіксація руху на зображенні камери не відбудеться.

Аналогічні процедури проводяться для пристроїв, що мають сенсор, сприйнятливий до змін значення магнітного поля. У даному випадку, граничними значенням, що спиняє процес роботи фіксації камерою, є також «1» (одиниця), яку отримано шляхом апроксимації експериментально отриманих даних.

Для отримання достовірних даних потрібно було виключити випадок дублювання фіксації одного й того ж контейнера, що проходить приблизно за однаковий проміжок часу. Використаємо метод *callback*, котрий відповідає за отримання байтів із зображення, зробленого камерою. Тоді в *onPreviewFrame(byte[], Camera)* першим параметром є той самий масив байтів, котрий підлягає обробці, а другим – об'єкт камери, що фіксує зображення.

Частота виклику цього методу операційною системою безпосередньо залежить від *FPS (frames-per-second)*, встановленому на пристрої програмно. Це є значення частоти отримання кадрів за секунду. Можливі значення було отримано програмним шляхом із списку доступних *FPS* у пристрої – із методу *Camera.getSupportedFps()*.

UML-діаграма класів клієнтського додатку пристрою-реєстратора наведена на рис. 5.

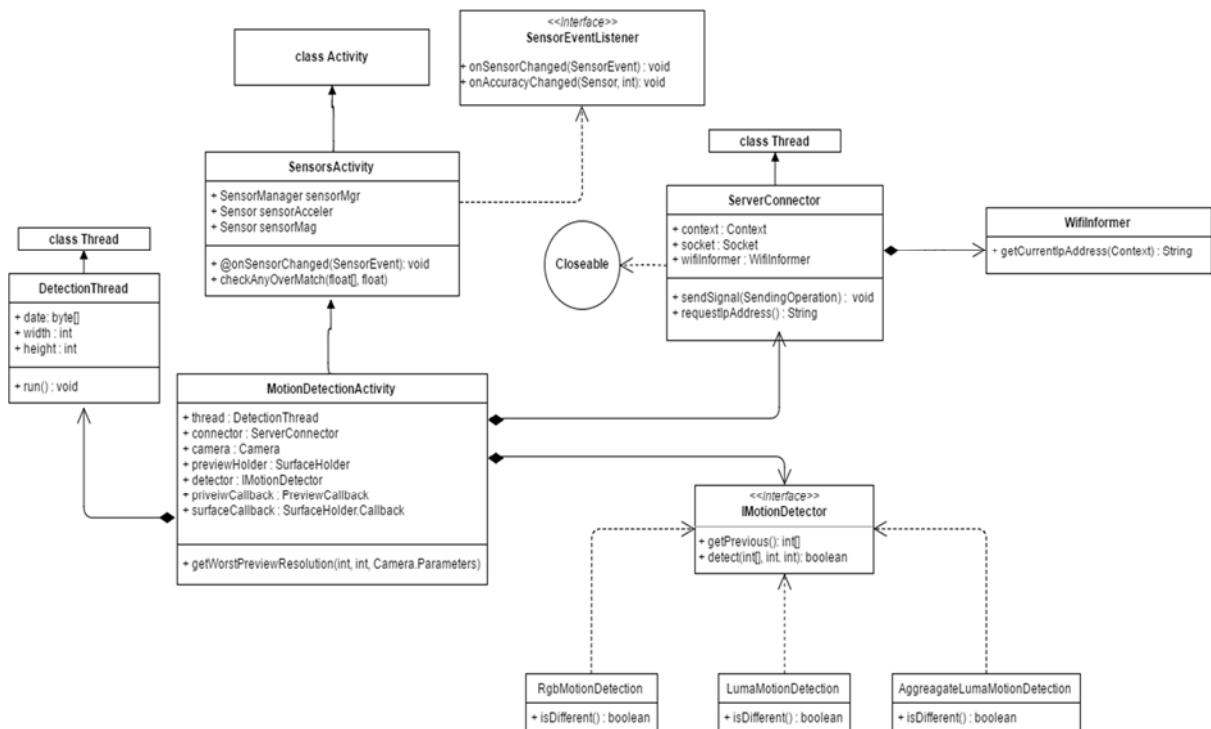


Рисунок 5 – Діаграма класів клієнтського додатку програми Motion Detector

Експериментально було встановлено, що частоти хоча б 1 кадр/с досягти неможливо програмним шляхом. Тому було вирішено налаштувати не частоту отримання кадрів, а лише проводити обробку зображення, а цей процес уміщує у собі конвертацію пікселів із формату *YUV420SP* у *ARGB*, перевірку даних акселерометра на нерухомість пристрою і перевірку схожості зображень попіксельно, щосекундно або за інші проміжки часу, які можна встановити програмно. «Старіння» даних при виконанні розрахунків для фіксації руху є неможливим, оскільки системний метод *onPreviewFrame* викликається із великою частотою *FPS*, що набагато більша за бажану частоту замірів, передаючи на обробку «свіжі» дані.

Оригінальне візуальне оформлення програми, що відображає зміни на сервері для особи, відповідальної за завантаження/розвантаження контейнерів, включає в себе анімаційні ефекти, які безпосередньо залежать від даних, які надходять від пристроїв-реєстраторів. UX побудований досить ергономічно і компактно, одночасно і функціонально, оскільки дає можливість користувачеві своєчасно отримати інформацію не тільки про завантаженість беїв, а й про доступність пристроїв-реєстраторів, якості їх зв'язку з сервером.

Працездатність і завантаженість моніторингової облікової *WiFi*-мережі контейнеровозу була оцінена шляхом математичного моделювання зазначеної мережі за допомогою зв'язного графу.

Мережі *ad-hoc*, або як їх ще називають бездротові самоорганізовані мережі, є децентралізованими бездротовими мережами, де всі вузли є рівноправними з точки зору управління мережею [8]. В таких мережах відсутні виділені сервери, де кожен вузол використовується або як клієнт, або для виконання функцій сервера, що дозволяє зберігати працездатність мережі при будь-якій кількості доступних вузлів. При функціонуванні мережі з різних причин можуть губитися зв'язки між окремими вузлами, тобто конфігурація мережі непостійна і, найчастіше, непередбачувана.

Для мережі такої змінної топології доцільно використовувати математичну модель невизначеності структури мережі як випадковий граф [9]. У такому разі оцінюються теоретико-ймовірнісні характеристики потоків у випадкових графах [10].

Представимо мережу у вигляді зв'язного графа  $G = (V, U)$ , де  $V$  – множина вершин або вузлів,  $U$  – множина ребер, та змінної множини  $W$  повідомлень (пакетів), які генеруються в ході роботи мережі. Розглянемо мережу з фіксованою кількістю вузлів  $n$ . Інші складові мережі вважатимемо випадковими.

Множина графа  $G = (V, U)$  формується наступним чином: генерується випадкове число  $|U|$  – потужність множини вершин, виходячи з дискретного розподілу ймовірностей  $PU$ . Далі з множини  $V \times V \setminus \text{diag}(V \times V)$  рівномірно обирається  $|U|$  різних вершин, які утворюють випадкову множину  $U'$ . Утворений граф  $G = (V, U')$  – зв'язний (від дерева з однією вершиною до дерева з  $n$  вершинами), тоді  $U' \subset U$ .

В мережі кожний вузол  $v_i, i = 1..n$  є джерелом потоку повідомлень  $w_i$  до деякого кореневого вузла  $v_j, j = 1..n, j \neq i$ , в якому повідомлення цього потоку повинні оброблятися. Потік  $w_i$  – це випадковий дискретний процес з кінцевою кількістю подій передачі повідомлень. Кількість подій в потоці – випадкова величина з дискретним розподіленням ймовірностей  $PS$ . Її значення не перевищує числа  $P$ . Інтервали часу  $\tau_i$  між послідовними подіями в потоці описуються функціями розподілу ймовірностей  $A_i(x)$  (скільки вузлів буде задіяно для передачі повідомлення).

Первинні процеси  $w_i$  (проходження контейнера над вузлом-реєстратором) породжують потоки дублюючих повідомлень, кожне передане повідомлення в мережі йде через цільовий вузол обробки. При надходженні повідомлення до цільового вузла, воно обробляється протягом часу  $\sigma_i$  з функцією розподілу ймовірностей  $R_i(x)$  (через перевантаження центрального вузла і створення черги FIFO).

Сумарно первинні процеси утворюють багатовимірний (розподілений по вузлах мережі) випадковий процес  $W$  з кінцевим числом подій. Цей процес не визначається

однозначно тільки множиною первинних процесів  $\{w_i\}$ . Він залежить також від алгоритму  $R$  маршрутизації. Використовується оптимальна маршрутизація, заснована на прогнозі часів очікування та обробки повідомлення від вузла-джерела до цільового вузла. Основними завданнями аналізу є вивчення таких характеристик розподіленого випадкового процесу  $W$ :

- 1) час  $\alpha$  відповіді на повідомлення від моменту генерації вузлом-джерелом до моменту одержання його цільовим вузлом);
- 2) час  $\beta$  обробки потоку дублюючих повідомлень (від генерації першого повідомлення до отримання останнього повідомлення з одної секції);
- 3) відношення числа повідомлень у процесі  $W$  до загальної можливості надходження подій в первинних процесах  $\{w_i\}$ .

Обчислювалися функції розподілу ймовірностей за наведених умовах (1):

$$RM(x) = P\{\alpha \leq x\} \text{ та } RS(x) = P\{\beta \leq x\}, \quad (3)$$

для кількості вузлів у мережі  $n = 1 \dots 20$ .

Була поставлена задача маршрутизації повідомлень у системі складної структури, заснованої на ефективному виконанні всіх процесів, які протікають у цій системі. Стан системи визначається величезною кількістю повідомлень, які обробляються та очікують у черзі на обробку повідомлень. Потрібно визначити оптимальний час обробки повідомлень в кореновому вузлі, витративши при цьому мінімальний час (виокремлення дублюючих).

При представленні моделі в часі кожний вузол та повідомлення мають структуру, подану на рис. 6.



Рисунок 6 – Елементи моделі бездротової мережі контейнеровозу

Аналіз ефективності маршрутизації проводився на подійно-орієнтованій моделі, яка задовольняє наступним умовам:

- зміна часу роботи моделі залежить від кількості повідомлень, які змінюють стан моделі та її навантаженість;
- логіка настання подій визначає кількість згенерованих вузлів та надісланих повідомлень з одної секції, які пов'язані з настанням цих подій.

Під час виконання випадково згенерованої системою моделі в конкретний момент часу  $t$  утворюється список вузлів (вершин) в кожній секції та список надісланих від них повідомлень, кожне з яких має свою імовірність появи. Після чого обчислюється загальний час на обробку згенерованої моделі (рис. 7).

Враховуючи, що в портах зазвичай не задіюють більше чотирьох кранів для завантаження/розвантаження одного судна, необхідно ввести функцію вибору ймовірнісного розташування кранів та вибору секції, з якою працюють портокрани.

При імітації систем великих масштабів для оцінки стійкості даних необхідно коректно обчислення параметрів, при яких виконання таких систем буде надійним. Розглянемо ці параметри і залежність кінцевих даних від цих змінних:

- 1) кількість вершин графа  $|V|$ ;
- 2) кількість згенерованих розміщень вершин графа. Для отримання коректних даних необхідні розгляд і обчислення даних по всім можливим генераціям графа. Тобто, маючи

$|V|$  кількість вершин, потрібно розглянути всі можливі варіанти генерації цих графів, кожен раз генеруючи новий набір розміщень вузлів з множини  $V$ .

3) кількість надісланих повідомлень з кожного графа. Після генерації графа необхідне проходження певної кількості тестів для виявлення надійних даних (порядок повідомлень повинен йти в порядку зростання номеру вершин).

4) розмірність потоку (знаходження дублюючих). Варто враховувати, що при кожному проході відбувається повторне породження потоків повідомлень  $w_i \in W$  однієї і тієї ж розмірності.

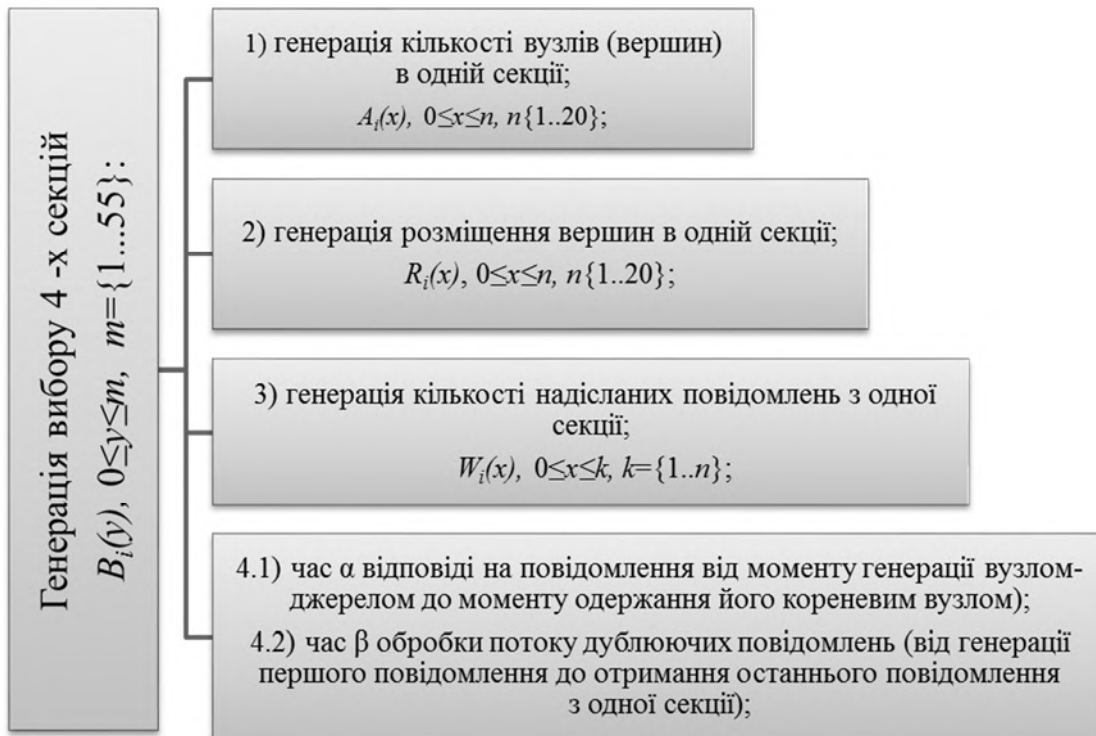


Рисунок 7 – Порядок генерації функції моделювання ймовірностей надходження повідомлень на кореневий вузол

**Висновки.** В роботі розглянуто питання, пов'язані з функціонуванням сучасних бездротових комп'ютерних мереж, які розгортаються в межах відокремленого технологічного об'єкта (судна-контейнеровозу). Побудована математична модель для проведення оцінки вартості потоків на графах сегментів бездротової комп'ютерної мережі при розвантаженні контейнеровозів та розраховано навантаження трафіку сегменту топології мережі для одного з *Bayplan* (вантажного плану на 4 секції).

Впровадження системи автоматизації обліку розвантаження суден контейнеровозів дозволить капітану дізнаватися, скільки контейнерів розвантажено з кожного ролу в режимі реального часу, тобто автоматизувати облік розвантаження контейнеровозу. Це, в свою чергу, дозволить капітану зберегти метacentричну висоту в допустимому діапазоні, тобто запобігти утворення крену і диференту судна, зменшити паперовий документообіг, підвищити продуктивність праці, скоротити час на обробку інформації.

Як наслідок, утворюються додаткові тимчасові ресурси для розробки і реалізації нових економічних і інвестиційних проектів. Таким чином вирішується проблема відшкодування і рентабельність впровадження моніторингової мережі на базі мобільних пристроїв, що розміщуються за місцем моніторингу за допомогою дронів.

**Перспективи роботи.** Перспективами подальшого розвитку роботи є вивчення основних характеристик поведінки розподілених випадкових процесів, що протікають в аналізованій системі. Необхідно створення технології для імітації процесів та проведення необхідних тестів для забезпечення коректності та стійкості вихідних даних. Бажано провести аналіз залежності кількості надісланих повідомлень, загального часу обробки повідомлень і навантаження корневих вузлів, визначити оптимально-мінімальну



кількість тестів для імітаційної моделі мережі. Провести дослідження поведінки і час обробки первинних і вторинних процесів залежно від кількості вершин, які задаються в системі.

Для організації кросплатформеності ПЗ розробленої автоматизованої системи немає необхідності повністю переписувати код додатка. Алгоритм визначення руху об'єктів, який використано, сам по собі є універсальним. Проте при розробці додатків для інших мобільних операційних систем (*Windows Mobile, iOS* тощо) необхідно буде використовувати відповідні нативні бібліотеки для маніпулювання робочими режимами камер реєстраторів.

Для подальшого розвитку методу реєстрації великих об'єктів доцільно дослідити можливості датчика освітленості мобільних пристроїв у разі тривалого перекриття такого датчика контейнером, що рухається.

Такий комплексний підхід надасть можливість швидшого масового поширення продукту.

### СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Ушаков Д. В. Организация контейнерных перевозок грузов. – М. : ТрансЛит, 2015. – 208 с.
2. Container Stowage Planning and how it works [Електронний ресурс]. – Режим доступу : <http://panbras.com.br/CONTAINER%20STOWAGE%20PLANNING.pdf>.
3. Накул Ю. А. Модель системи завантаження крупнотонажних суден / Ю. А. Накул, Стовманенко В. О. // Наукові праці [Чорномор. нац. ун-ту ім. Петра Могили]. Серія : Комп'ютерні технології : наук.-метод. журн. – 2016. – Т. 275, Вип. 287. – С. 27–33.
4. Нікольський В. В. Система підтримки прийняття рішення по навантаженню великотонажного контейнеровоза / В. В. Нікольський М. В. Нікольський, Ю. А. Накул // Наукові праці [Чорномор. нац. ун-ту ім. Петра Могили]. Серія : Комп'ютерні технології : наук.-метод. журн. – 2016. – Т. 271, Вип. 283. – С. 70–73.
5. Хламов С. В. Разработка вычислительного метода обнаружения околонулевого видимого движения объектов на серии ссд-кадров / С. В. Хламов, В. Е. Саваневич, А. Б. Брюховецкий, С. С. Орышич // Восточно-Европейский журнал передовых технологий. – 2016. – № 2/9 (80). – С. 41–48.
6. Харди Б. Android. Программирование для профессионалов ; 2-е изд / Б. Харди, Б. Филлипс, К. Стюарт, К. Марсикано. – М. : Питер, 2016. – 640 с.
7. Уровни API, версии Android, –target, файл AndroidManifest.xml [Электронный ресурс] / Бражников Р. – Оpubл. 2015/03/20. – Режим доступа : <http://romanbrazhnikov.ru/blog/2015/03/20/уровни-апи-версии-android-target-файл-androidmanifest-xml/>.
8. Миков А. И. Стохастические характеристики разрезов в графах AD НОС сетей // Информатизация и связь. – 2014. – № 1. – С. 76–79.
9. Karnon J. Modeling using discrete event simulation // Medical decision making. – 2012. – P. 705–706.
10. Hekmat R. Ad-hoc networks: Fundamental properties and network topologies. – Zoetermeer, The Netherland : Springer, 2006. – 154 p.

### REFERENCES

1. Ushakov D. V. (2015) *Organization of container cargo transportation*. M. : TransLit.
2. Container Stowage Planning and how it works [Electronic resource]. Available at: <http://panbras.com.br/CONTAINER%20STOWAGE%20PLANNING.pdf>.
3. Khlamov S. V., Savanevich V. E., Briukhovetskyi A. B., Oryshych S. S. (2016) Development of a computational method for detecting near zero visible motion of objects on a

series of CCD-frames. *Eastern-European Journal of enterprise technologies*. No 2/9 (80). – 41–48.

4. Nakul Y. A., Stovmanenko V. O. Model of system for transport tonnage ships downloading. *Scientific papers*. Series «Computer Technology». Vol. 275, Issue 287. 27–33.

5. Nikolskiy V. V., Nikolskiy M. V., Nakul Y. A. (2016) Decision support system for loading bulk container carriers. *Scientific papers*. Series «Computer Technology». Vol. 271, Issue. 283. 70–73.

6. Brazhnikov P. (2015) API Level, v. Android, –target, file AndroidManifest.xml [Electronic resource]. Available at : <http://romanbrazhnikov.ru/blog/2015/03/20/уровни-апи-версии-android-target-файл-androidmanifest-xml/>.

7. Khardi B., Phillips B., Stewart C., Marsicano C. (2016) *Android. Programming for professionals*; 2nd ed. M. : Питер.

8. Hekmat R. (2006) *Ad-hoc networks: Fundamental properties and network topologies*. Zoetermeer, The Netherland : Springer.

9. Mykov A. I. (2014) Stochastic characteristics of sections in AD HOC graphs. *Informatization and communication*. No 1. 76–79.

10. Karnon J. (2012) Modeling using discrete event simulation. *Medical decision making*. 705–706.

**Журавская И. Н., Мусиенко М. П., Кулаковская И. В. АВТОМАТИЗИРОВАННАЯ СИСТЕМА УЧЁТА РАЗГРУЗКИ СУДОВ КОНТЕЙНЕРОВОЗОВ С ИСПОЛЬЗОВАНИЕМ МОБИЛЬНЫХ УСТРОЙСТВ**

*Несмотря на растущую компьютеризацию водных транспортных средств, пока процесс учета загрузки/разгрузки контейнеровозов осуществляется вручную тальманами. При таком порядке у капитана судна нет текущих данных о количестве разгруженных/загруженных контейнеров в каждой секции контейнеровоза. В работе предложен метод регистрации контейнеров, перенесенных через борт судна порткраном, с помощью мобильных устройств, размещаемых на крыше одного или нескольких контейнеров в каждой секции. При отсутствии в секции необходимого для учета контейнеров количества мобильных устройств, дополнительные разносятся квадрокоптерами. Информация капитану передается через развернутую на контейнеровозе локальную WiFi-сеть, нагруженность которой была оценена путем математического моделирования указанной сети с помощью связанного графа. Регистрация события разгрузки/загрузки каждой секции определяется изменением состояния не менее 20 % пикселей матрицы фотокамеры мобильного устройства по результатам анализа траектории изображения.*

**Ключевые слова:** судно-контейнеровоз, квадрокоптер, локальная WiFi-сеть, фотокамера смартфона, нагруженность сети, связанный граф.

**Zhuravska I. M., Musyienko M. P., Kulakovska I. V. AUTOMATED SYSTEM USING MOBILE TO UNLOAD THE CONTAINER SHIPS**

*Despite the growing computerization of water vehicles, the recording process of the loading/unloading of container ships is carried out manually by talmans now. In this way, the captain of the vessel does not have the current data on the number of unloaded/loaded containers in each section of the container ship. The paper proposes a method for registering containers carried onboard by a port crane. This method uses mobile devices placed on the roof of one or more containers in each section. If there is no enough number of mobile devices in the section which are necessary for accounting the containers, additional ones are carried by quadcopters. The information is transmitted to the captain through a local Wi-Fi network deployed on the container ship. The network loading was estimated by mathematical modeling using a messenger graph. The registration of an unloading/loading event of each section is determined by a change in the state of at least 20 % of the pixels of the camera matrix of the mobile device based on the basis of the analysis of the image trajectory.*

**Keywords:** container ship, quadcopter, local Wi-Fi network, smartphone camera, network loading, connected graph.

© Журавська І. М., Мусієнко М. П., Кулаковська І. В.

Статтю прийнято  
до редакції 03.05.17