

А. О. Олійник, Є. М. Федорченко, О. О. Степаненко, М. С. Рудь

Національний університет «Запорізька політехніка»
вул. Жуковського, 64, 69063 Запоріжжя, Україна

Розв'язання задачі комівояжера на основі еволюційного моделювання

Запропоновано еволюційну модель для розв'язання задачі комівояжера, виконано її апробацію у сфері аптечного бізнесу шляхом оптимізації процесу роботи засобу подачі ліків. У розробленій моделі використано модифіковані оператори ініціалізації початкової популяції. Розроблені оператори ініціалізації початкової популяції передбачають створення початкової множини рішень, виходячи з особливостей розв'язуваної задачі, що дозволяє генерувати більш пристосовані хромосоми (хромосоми з кращими значеннями функції пристосованості) на початковому етапі пошуку та наблизити початкові точки до області глобального екстремуму, зменшити час оптимізації і обсяг використаних ресурсів комп'ютера. Розроблену еволюційну модель для розв'язання задачі комівояжера було імplementовано шляхом її програмної реалізації і впровадження в аптечній мережі «Аптека низьких цін». У програмі реалізовано можливість побудови контуру обходу для пошуку ліків у роботизованому складі на 1000 чарунок, тривалість пошуку ліків є прийнятною для підприємства та складає не більше п'яти секунд.

Ключові слова: задача комівояжера, генетичний алгоритм, еволюційна модель, мінімальна відстань, граф.

Вступ

Задача комівояжера є оптимізаційною задачею, що часто виникає на практиці. Проблема побудови оптимальних маршрутів через задану множину точок на площині чи у просторі виникає у багатьох сферах людської діяльності, наприклад, у задачах планування та логістики, при виробництві друкованих плат, мінімізації рухів у робототехніці, аналізі структури ДНК тощо. Великий інтерес до цієї задачі пов'язаний із використанням її як платформи для дослідження загальних методів комбінаторної оптимізації [1].

Задача комівояжера може бути сформульована таким чином: дано множину міст, а також відстані між усіма парами міст. Необхідно знайти контур обходу, що починається з першого міста, яке повинен відвідати комівояжер, і проходить через усі міста в певному порядку по одному разу, повертаючись після цього в початкове

положення, при цьому довжина контуру має бути мінімальною. Задачу можна представити у вигляді моделі на графі. Розглянемо зв'язний орієнтований граф: $G = (V, E, h)$, в якому $V = \{v_1, v_2, \dots, v_N\}$ — множина вершин; $E = \{e_1, e_2, \dots, e_M\}$ — множина ребер (дуг); $h: E \rightarrow Z_+$ — вагова функція дуг. Для математичної постановки завдання зручно позначити значення вагової функції дуг через $c_{ij} = h(e_k)$, де дуга $e_k \in E$ відповідає впорядкованій парі вершин (v_i, v_j) , i, j — індекси вершин у множині V . Значення $c_{ij} = h(v_i, v_j)$ інтерпретується як довжина ділянки між вершинами (i, j) вихідного графа. Необхідно визначити таку підмножину дуг (і вершин), яка утворює в графі G замкнутий контур, що проходить через кожну потрібну вершину один раз (гамільтонів цикл) вигляду $X = \langle x_1, x_2, \dots, x_N \rangle$, де x_b — номер вершини $v_i \in V$ у шуканому контурі обходу з початкового положення. Контур мінімізує цільову функцію

$$F(X) = c_{x_1x_2} + c_{x_2x_3} + \dots + c_{x_Nx_1},$$

де c_{ij} — елемент матриці відстаней між вершинами графа.

Практичне використання відомих методів розв'язання задачі комівояжера ускладнюється у разі необхідності оброблення даних великих розмірностей, оскільки складність обчислень при використанні цих методів зростає експоненційно. Тому доцільною є розробка методів, що дозволяють прискорити процес пошуку оптимального рішення [2].

Аналіз літературних даних і постановка проблеми

У наш час відомі різні практичні застосування, що пов'язані з необхідністю розв'язання задачі комівояжера. Одним із таких застосувань є аптечний робот для автоматичної подачі ліків за вибором клієнта. При використанні таких роботів для вибору множини потрібних клієнтові ліків $V = \{v_1, v_2, \dots, v_N\}$ з множини усіх наявних в апараті виникає проблема розв'язання оптимізаційної задачі комівояжера. Для цього необхідно дістатися до кожного потрібного товару $v_i \in V$ таким чином, щоби пройдений роботом шлях був найменшим серед усіх можливих.

Ця задача відома через різкий контраст між простотою її формулювання та важкості рішення, вона відноситься до розряду NP -складних задач і демонструє всі аспекти комбінаторної оптимізації, слугувала й продовжує слугувати як опорна позначка для нових алгоритмічних ідей, як то імітований відпал, пошук табу, нейронні мережі та еволюційні методи [2].

У [2] запропоновано такий загальний шлях розв'язання задачі комівояжера за допомогою генетичних алгоритмів:

- 1) задати спосіб кодування контуру обходу комівояжера через міста у вигляді хромосом;
- 2) застосувати генетичні оператори для збереження працездатності та структури алгоритму;
- 3) запобігти передчасній збіжності, тобто неможливості генерації нащадків, які були би кращими за своїх батьків.

Також у цій роботі надано перелік та опис генетичних операторів, прийнятих для використання при розв'язанні цієї задачі.

Робота [3] надає опис проблеми перестановки, яку дуже важливо враховувати при розв'язанні задачі комівояжера, та перелік і опис генетичних операторів схрещування, що дозволяють розв'язати цю проблему.

Важливим елементом функціонування пристрою є алгоритм подачі потрібного клієнтові товару, тобто, розв'язання оптимізаційної задачі комівояжера, яка в даному випадку полягає в тому, щоби дістатися потрібного товару певним шляхом, за умови, що цей шлях буде найкоротшим серед усіх можливих шляхів. Тобто існує проблема знаходження оптимального контуру обходу з використанням меншого обсягу використаних ресурсів комп'ютера

Метою роботи є розробка еволюційної моделі та програмних засобів пошуку оптимального контуру обходу при виборі набору ліків, які розташовані в апараті за вимогами користувача.

Серед уже існуючих публікацій за обраною тематикою можна виділити такі статті:

— Jean-Yves Potvin у роботі «Genetic algorithms for the traveling salesman problem» (1996) проводить аналіз методів розв'язання оптимізаційної задачі комівояжера за допомогою евристичних алгоритмів і дає детальний опис способу використання генетичних алгоритмів для розв'язання комбінаторних задач, у тому числі, спосіб розв'язання проблеми перестановки. У висновку статті зазначається, що генетичні алгоритми здатні конкурувати з найкращими евристичними алгоритмами для задач середнього розміру (декількох сотень точок). Однак вони потребують багато часу та не можуть бути успішно застосовані до проблеми з розмірністю приблизно мільйон точок [4–8].

— Woan Shin Tan, Siang Li Chua, Keng Who Yong, Tuck Seng Wu у роботі «Impact of Pharmacy Automation on Patient Waiting Time: An Application of Computer Simulation» (2009) намагаються показати можливості використання комп'ютерної симуляції для розрахунку впливу автоматизованої аптечної системи розподілення на час очікування покупців і взагалі потенціал цих систем як рутинного засобу в аптечному управлінні. У висновку статті зазначається що використання автоматизованої системи розподілення призводить до підвищення швидкості обслуговування та дозволяє зменшити число робітників на ~28 % [9].

З метою усунення виявлених недоліків відомих методів розв'язання задачі комівояжера та їхнього практичного застосування в даній роботі розроблено еволюційну модель розв'язання задачі комівояжера за допомогою еволюційного пошуку.

Мета та задачі дослідження

Мета дослідження — розробка еволюційного методу розв'язання задачі комівояжера, а також програмного забезпечення пошуку оптимального контуру обходу на основі еволюційної оптимізації.

Задача дослідження:

- аналіз існуючих методів розв'язання задачі комівояжера;
- розробка еволюційної моделі розв'язання задачі комівояжера за допомогою еволюційного пошуку;
- програмна реалізація еволюційної моделі розв'язання задачі комівояжера;

— дослідження ефективності розробленої моделі.

Загалом процес роботи звичайного еволюційного алгоритму можна описати таким чином: циклічно змінювати покоління (популяція на окремій ітерації), використовуючи при цьому еволюційні оператори. Розраховувати пристосованість кожної із хромосом за допомогою обчислення фітнес-функції, доти поки не буде досягнуто вказаного ліміту ітерацій, або різниця між двома послідовними значеннями функції вартості серед поколінь не буде менше заданого числа. За правилами відбору на кожній ітерації залишаються тільки ті особини, що є найбільш пристосованими — саме вони використовуються оператором схрещування, для отримання нащадків. Інколи після схрещування відбуваються мутації — випадкові зміни у хромосомі, що дозволяють при рішенні задачі оптимізації вийти з локального мінімуму [10].

Для вирішення окресленої вище задачі у цій роботі буде використовуватися еволюційний пошук.

Розробка еволюційної моделі розв'язання задачі комівояжера

Нехай маємо набір розташованих у апараті ліків $A = \{a_1, a_2, \dots, a_K\}$, що характеризується геометричними координатами їхнього розташування у двовимірній системі координат, які будемо позначати $a_k(x_k, y_k)$, де $a_k \in A$ — елемент множини A , що відповідає певному типу ліків, які розташовані в апараті за координатами (x_k, y_k) . Тоді необхідно із множини A розташованих в апараті ліків, обрати підмножину потрібних клієнтові ліків $V = \{v_1, v_2, \dots, v_N\}$ таким чином, щоб довжина шляху маніпулятора, який він пройде через всі елементи множини $V = \{v_1, v_2, \dots, v_N\}$, була мінімальною.

Відомим способом розв'язання даної задачі є метод брутфорсу [2] або грубої сили, коли ми розраховуємо кожен із можливих контурів обходу та обираємо найкращий. Проте для n точок існує $(n-1)!$ можливих контурів обходу [2], що суттєво обмежує такий підхід при застосуванні на практиці. Тому для вирішення практичних завдань, що пов'язані з необхідністю розв'язання задачі комівояжера, доцільним є застосування еволюційних, зокрема, генетичних, методів, використання яких вимагає менше часу та ресурсів комп'ютера.

Функція вартості для найпростішої форми задачі комівояжера — це відстань, яку проходить комівояжер у заданому порядку $\langle (x_n, y_n) \rangle$, $n = 1, \dots, N$, вона використовується для оцінки контурів обходу, $n = 1, \dots, N$ за формулою

$$\text{cost} = \sum_{i=1}^N \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2}, \quad (1)$$

де (x_i, y_i) — координати i -ї точки.

При використанні генетичного алгоритму для розв'язання задачі комівояжера треба звернути увагу на те, що операція схрещування може приводити до того, що у шляху будуть з'являтися однакові індекси відвідуваних точок. Така ситуація називається проблемою перестановки [2]. Вона може розв'язуватися за допомогою використання спеціальних операторів схрещування та мутації.

Для відбору хромосом для схрещування у розробленій моделі використовувався турнірний оператор відбору (Tournament selection).

Як оператор схрещування використано впорядкований оператор (Order crossover, OX). Він був запропонований Д. Девісом у 1985 році для негомологічних числових хромосом. Як оператор мутації використано класичну мутацію обміну.

Для генерації випадкових чисел використовується рівномірний розподіл. Установлено, що ефективне використання еволюційних методів суттєво залежить від формування початкової множини рішень. Еволюційні методи, як правило, використовують підхід, що передбачає випадкове формування початкової популяції, та не використовують відому апріорну інформацію про розв'язувану практичну задачу. Це в багатьох випадках призводить до того, що точки-рішення початкової популяції суттєво віддалені від області глобального екстремуму, та до необхідності суттєвих витрат часу і ресурсів комп'ютера при використанні таких методів.

У розробленій еволюційній моделі для розв'язання задачі комівояжера пропонується використовувати модифіковані оператори ініціалізації початкової популяції, які передбачають створення початкової множини рішень, виходячи з особливостей розв'язання задачі комівояжера, що дозволяє наблизити початкові точки до області глобального екстремуму, яку ще треба знайти, та зменшити час оптимізації і використання ресурсів комп'ютера.

Робота алгоритму може зупинитись у декількох випадках:

— при досягненні максимального числа ітерацій;

— при виродженні популяції — коли відсоток поліпшення значення кращої хромосоми F є меншим за мінімально допустимий. Нехай t_{tr} позначає кількість ітерацій алгоритму, ρ_{tr} позначає поріг коефіцієнта поліпшення значень фітнес-функції кращої хромосоми. Починаючи з циклу $(t_{tr} + 1)$, у генетичних операторах обчислюється значення цільової функції для кожної хромосоми останньої популяції і вибирається із цих значень найкраще (найменше) значення цільової функції $f_{best\ t}$. З попередніх поколінь вибирається найкраще значення цільової функції $f_{best\ t-1}$ [10–28]. Після чого обчислюється коефіцієнт поліпшення за формулою:

$$\rho = \frac{f_{best\ t} - f_{best\ t-1}}{f_{best\ t-1}}. \quad (2)$$

Для порівняння результатів ефективності роботи маніпулятора для надання ліків будемо використовувати такі показники як довжина пройденого їм контуру обходу, а також час його проходження

$$time = \frac{L}{V}, \quad (3)$$

де $time$ — час проходження контуру обходу від початку руху маніпулятора та до його закінчення, с; L — довжина пройденого маніпулятором контуру обходу в процесі вибору ліків з роботизованого складу, м; V — швидкість руху маніпулятора, м/с.

Розроблена еволюційна модель для розв'язання задачі комівояжера передбачає використання модифікованих операторів ініціалізації початкової популяції (випадкової і змішано-максимальної ініціалізації), що дозволяє генерувати більш пристосовані хромосоми (хромосоми з кращими значеннями функції пристосова-

ності) на початковому етапі пошуку та таким чином зменшити час еволюційної оптимізації.

Результати дослідження еволюційної моделі задачі комівояжера

З метою аналізу розробленої моделі було проведено експерименти з дослідження операторів ініціалізації початкової популяції: випадкова, змішано-максимальна. Враховуючи стохастичний характер еволюційного пошуку, експерименти проводилися 20 разів з такими параметрами: кількість ітерацій — 1500; розмір популяції — 100 хромосом (структур даних, кожна з яких представляє собою певний контур обходу, і які модифікуються в процесі еволюційної оптимізації); коефіцієнт елітизму (частки кращих хромосом у популяції, які автоматично переходять на наступну ітерацію) — 0,02; коефіцієнт схрещування — 0,9; коефіцієнт мутації — 0,01. Вхідні дані — 95 точок системи координат. Результати експериментів наведено у таблиці та на рис. 1.

Результати експериментів

Назва оператора ініціалізації початкової популяції	Початкове значення відстані, м	Результуюче значення відстані, м	Час виконання, с
Змішаний	1,784	0,521	8,258
Змішано-максимальний	1,671	0,506	8,042

На рис. 1, який відображає результати першого експерименту з дослідження змішаного підходу до ініціалізації, зображено графіки залежності кількості ітерацій від функції вартості та відстані між точками при руху каретки аптечного робота між ліками.

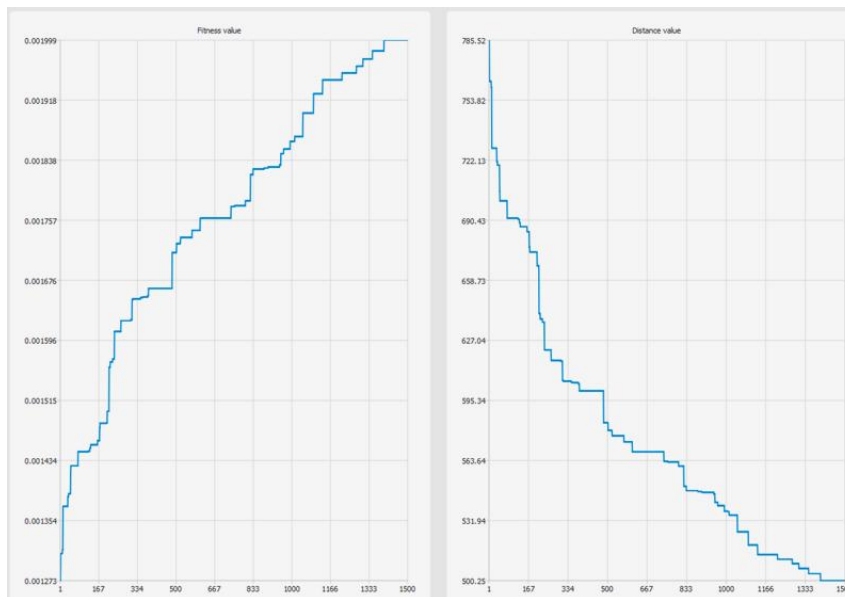


Рис. 1. Графік зміни значення цільової функції (зліва) та відстані, яку проходить аптечний робот (справа), залежно від номеру ітерації

Після проведення експериментів варто відзначити, що кращий результат забезпечено при використанні змішано-максимальної ініціалізації початкової популяції. Це пояснюється тим, що у цій задачі комівояжера використання початкової популяції дозволяє мати менше початкове значення розрахованої функції пристосованості та, відповідно, дистанції.

Аналіз роботи еволюційної моделі розв'язання задачі комівояжера

При використанні базового генетичного алгоритму [12–15] та розробленої еволюційної моделі розв'язання задачі комівояжера були отримані такі результати: $L_1 = 5,2$ м, $t_1 = 3,8$ с та $L_2 = 3,7$ м, $t_2 = 2,7$ с, відповідно.

Як видно з таблиці, запропонована модель дозволяє збільшити швидкість і зменшити довжину контуру обходу, пройденого маніпулятором. Порівнюючи результати експериментів, видно, що час руху каретки аптечного робота скоротився на 1,1 секунди: $\Delta t = 3,8 - 2,7 = 1,1$ с. Також зазначимо, що відстань, яку долає маніпулятор при пошуку набору ліків, у середньому скорочується на $L_1 - L_2 = 1,5$ м.

Варто відзначити, що при збільшенні кількості вузлів економія часу Δt порівняно з базовим генетичним алгоритмом [12–15] також збільшиться.

Програмна реалізація моделі

Розроблена еволюційна модель розв'язання задачі комівояжера була імплементована шляхом її програмної реалізації і впровадження в аптечній мережі «Аптека низьких цін» (АНЦ). У програмі реалізовано можливість побудови контуру обходу для пошуку ліків у роботизованому складі на 1000 чарунок, тривалість пошуку ліків є прийнятною для аптеки та складає не більше п'яти секунд.

Програмне забезпечення розроблене із використанням принципів об'єктно-орієнтованого програмування та складається з кількох взаємопов'язаних модулів — класів з їхніми складовими — методами та даними, тому можна зобразити структуру програми за допомогою UML-діаграми класів (рис. 2).

На діаграмі класів видно що головним класом є `Widget`. З ним, асоціативним зв'язком (композиція) пов'язані класи `CGeneticAlghoritm`, `CMyGraphView` та за допомогою агрегації клас `CresultForm`. Головні класи також наслідують класи бібліотеки Qt як то `QObject`, `QRunnable`, `QWidget`, `QgraphicsView`.

Для більш детального огляду функціональної складової системи доречно навести UML-діаграми послідовностей, кооперації, станів, діяльності, що зображені відповідно на рис. 3–6.

На діаграмі послідовностей зображено головні об'єкти розробленої програми, їхні лінії життя та існуючі між ними зв'язки. Ця діаграма дозволяє побачити у хронологічному порядку послідовність створення вказаних об'єктів та обміну повідомлень між ними. З неї видно, що першим створюється головне вікно програми — об'єкт класу `Widget`, разом з яким створюються об'єкти класів `CGeneticAlghoritm`, `CMyGraphView`. Об'єкт класу `CGeneticAlghoritm` виконує основну роботу з розв'язуванням задачі оптимізації та оновлення інтерфейсу користувача. Він по завершенні роботи надає результати об'єкту класу `Widget`, який у свою чергу, для їхнього відображення створює форму — об'єкт класу `CResultForm`.

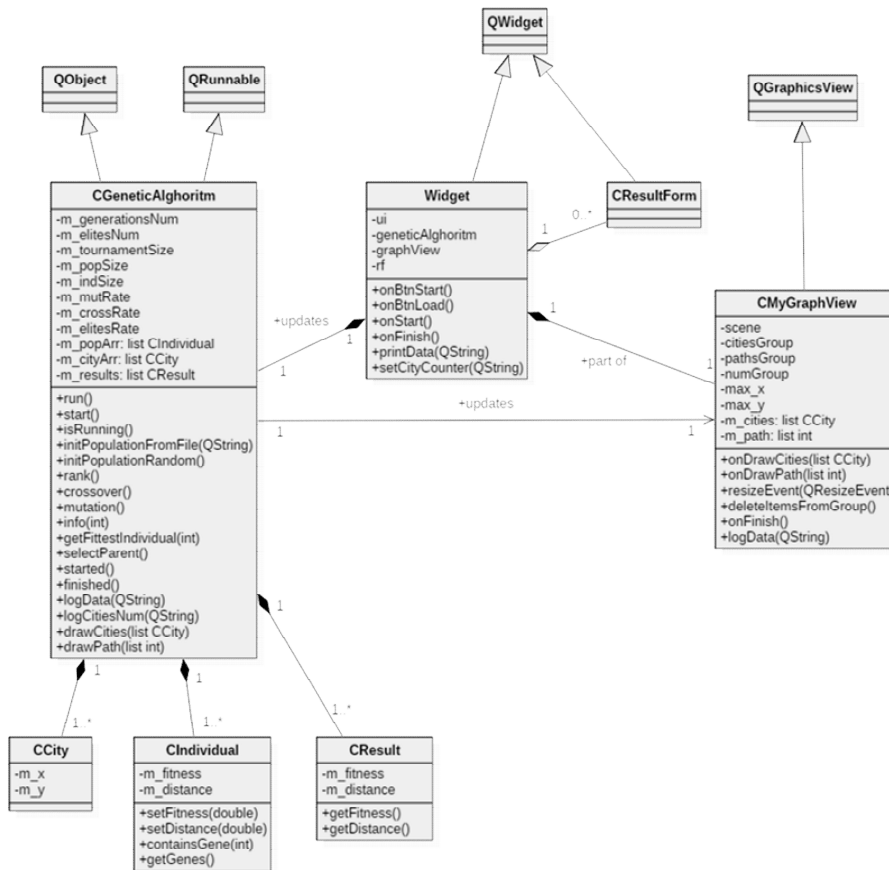


Рис. 2. Діаграма класів

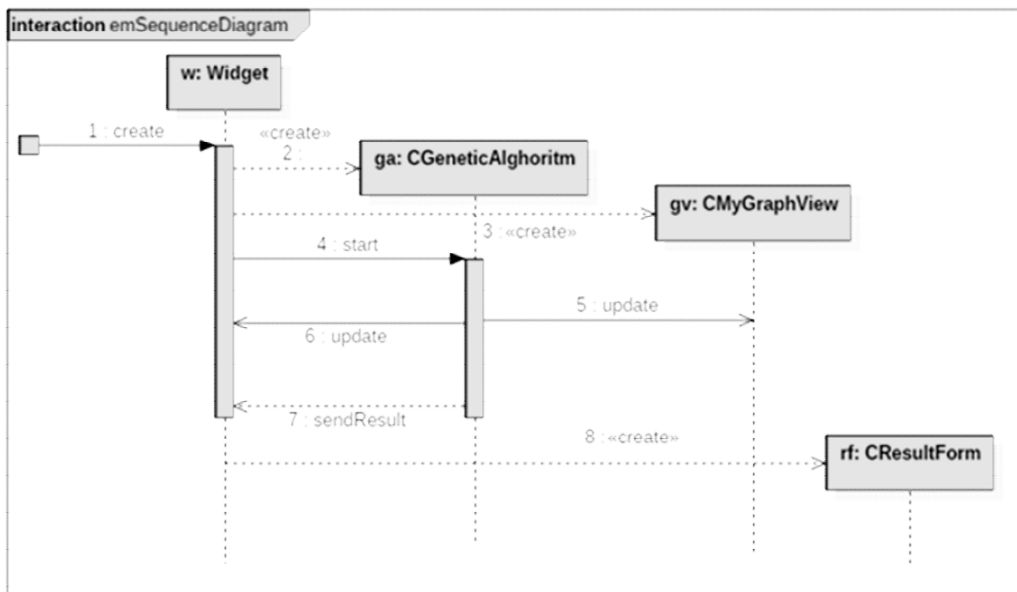


Рис. 3. Діаграма послідовностей

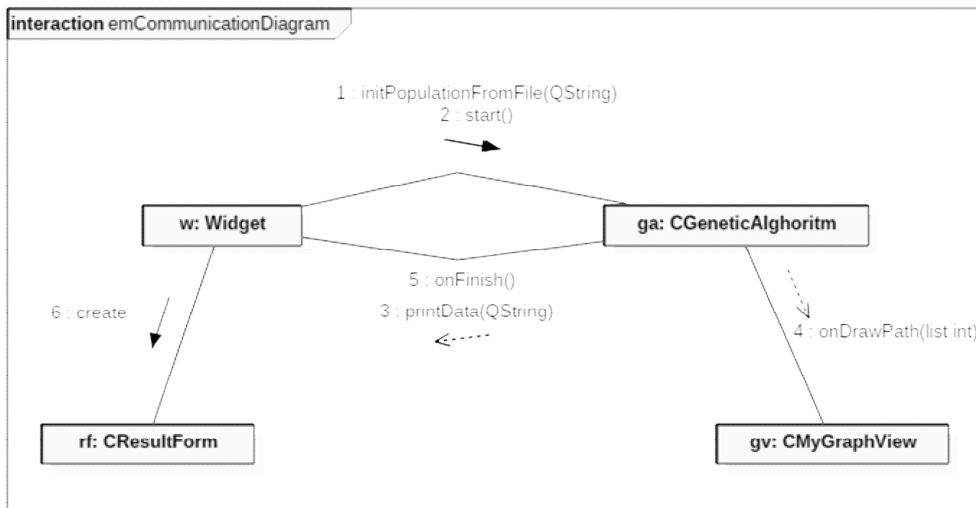


Рис. 4. Діаграма кооперації



Рис. 5. Діаграма станів



Рис. 6. Діаграма діяльності

На діаграмі кооперації зображено основні об'єкти розробленої програми та функціональний зв'язок між ними. Вона надає можливість прослідкувати за порядком і напрямком повідомлень між об'єктами. Ця діаграма не включає у себе

шкалу життя об'єкта, тому є більш компактною для аналізу взаємодії зазначених об'єктів. Діаграма є миттєвим знімком процесу роботи ГА після виконання методу start(). На ній показано такі події: оновлення інтерфейсу користувача (printData (QString)), візуалізація роботи алгоритму (onDrawPath(list int)), відображення результуючої інформації (create Cresultform).

На діаграмі станів зазначено послідовність подій, які приводять до зміни абстракцій значень і зв'язків системи, тобто її станів. Діаграма показує, що зміна до деяких станів, наприклад «Запущено алгоритм», потребує щоб система була в двох попередніх станах — «Завантажено дані», «Обрано параметри».

Таким чином, розроблене програмне забезпечення надає можливість побудови контуру обходу для пошуку ліків у роботизованому складі на основі запропонованої еволюційної моделі розв'язання задачі комівояжера.

Висновки

1. Розроблено еволюційну модель розв'язання задачі комівояжера для оптимізації контуру обходу руху маніпулятора аптечного робота. На відміну від існуючих методів, розроблена еволюційна модель розв'язання задачі комівояжера передбачає використання модифікованих операторів ініціалізації початкової популяції, що дозволяє генерувати більш пристосовані хромосоми (хромосоми з кращими значеннями функції пристосованості) на початковому етапі пошуку та таким чином зменшити час еволюційної оптимізації.

2. Запропоновану еволюційну модель розв'язання задачі комівояжера було імплементовано шляхом її програмної реалізації і впровадження в аптечній мережі АНЦ. У програмі реалізовано можливість побудови контуру обходу для пошуку ліків у роботизованому складі на 1000 чарунок, тривалість пошуку ліків є прийнятною для підприємства та складає не більше п'яти секунд.

3. Вирішено практичне завдання побудови контуру обходу руху маніпулятора аптечного робота при пошуку ліків.

1. The Travelling Salesman Problem and its Applications. CO@W Berlin, 2009. URL: [http://coat-work.zib.de/berlin2009/downloads/2009-09-21/2009-09-21-1600-MG-TSP-and Applications.pdf](http://coat-work.zib.de/berlin2009/downloads/2009-09-21/2009-09-21-1600-MG-TSP-and%20Applications.pdf)

2. Gen M., Cheng R. Genetic algorithms and engineering design. New Jersey: John Wiley & Sons, 1997. 352 p.

3. Haupt Randy L., Haupt Sue Ellen. Practical genetic algorithms. 2nd ed., 2004. 261 p.

4. Kleiman H. The General Traveling Salesman Problem. Createspace Independent Pub, 2014. 470 c. ISBN: 149438714X 9781494387143.

5. Мудров В.И. Задача о коммивояжере. URSS, 2019. 70 с. ISBN: 978-5-397-04106-5.

6. Tan W., Chua S., Yong K., Wu T. Impact of Pharmacy Automation on Patient Waiting Time: An Application of Computer Simulation. *Simulating Impact of Pharmacy Automation*. 2009. Vol. 38. P. 501–507.

7. Олійник А.О., Субботін С.О., Олійник О.О. Еволюційні обчислення та програмування: навч. посіб. — Запоріжжя: ЗНТУ, 2010. 324 с.

8. Субботін С.О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень: навч. посіб. — Запоріжжя: ЗНТУ, 2008. 341 с.

9. Дубровін В.І., Субботін С.О. Методи оптимізації та їх застосування в задачах навчання нейронних мереж: навч. посіб. — Запоріжжя: ЗНТУ, 2003. 136 с.

10. Stroustrup Bjarne. The C++ programming language. Fourth edition. 2013. 1366 p.

11. Шлее М. Qt 5.5 Профессиональное программирование на C++. Санкт-Петербург: «БХВ-Петербург», 2015. 896 с.
12. Dopico J. Encyclopedia of artificial intelligence; Eds.: J.R. Dopico, J.D. de la Calle, A.P. Sierra. New York: *Information Science Reference*. 2009. Vol. 1–3. 1677 p.
13. Nagata Y., Kobayashi S. A Powerful Genetic Algorithm Using Edge Assembly Crossover for the Traveling Salesman Problem. 2012. P. 346–363. doi: 10.1287/ijoc.1120.0506.
14. Hoffman K., Rinaldi G. Traveling Salesman Problem. *Advertising Response, Encyclopedia of Operations Research*. 2013. P. 1573–1578. doi: 10.1007/978-1-4419-1153-7_1068.
15. Wang Y. The hybrid genetic algorithm with two local optimization strategies for traveling salesman problem. *Computers & Industrial Engineering*. 2014. Vol. 70. P. 124–133. doi: 10.1016/j.cie.2014.01.015.
16. Sanches D., Whitley D. Improving an exact solver for the traveling salesman problem using partition crossover. Proceedings of the Genetic and Evolutionary Computation Conference. 2017. P. 337–344. doi: 10.1145/3071178.3071304.
17. Hussain A., Muhammad Y. Genetic Algorithm for Traveling Salesman Problem with Modified Cycle Crossover Operator. *Computational Intelligence and Neuroscience*. 2017. Vol. 2017. P. 7. doi: 10.1155/2017/7430125.
18. Tsai C., Tseng S. A High-Performance Genetic Algorithm: Using Traveling Salesman Problem as a Case. *The Scientific World Journal*. 2014. Vol. 2014. P. 14. doi: 10.1155/2014/178621.
19. Прогресивні технології моделювання, оптимізації та інтелектуальної автоматизації етапов життєвого циклу авіадвигателів: монографія/Богуслаєв А.В. і др.; под ред. Д.В. Павленко, С.А. Субботина. Запоріжжя: ОАО «Мотор Сич», 2008. 468 с.
20. Рассел С., Норвіг П. Искусственный интеллект: современный подход. Москва: Вильямс, 2006. 1408 с.
21. Applegate David L., Robert E. Bixby, Chvatal Vasek. The Traveling Salesman Problem A Computational Study. Princeton University Press Princeton, NJ, USA, 2007. 608 с. ISBN: 0691129932 9780691129938.
22. Руденко О.Г., Бодяньський Є.В. Штучні нейронні мережі. Харків: Компанія СМІТ, 2006. 404 с.
23. Cook William J. In Pursuit of the Traveling Salesman: Mathematics at the Limits of Computation. Princeton University Press, 2012. 228 p. ISBN: 0691152705 9780691152707.
24. Zuhori S.T. Traveling Salesman Problem. Lap Lambert Academic Publishing GmbH KG, 2012. 56 p. ISBN: 3846583057 9783846583050.
25. The practical handbook of genetic algorithms / Ed. L.D. Chambers. Florida: CRC Press, 2000. Vol. I: Applications. 520 p.
26. Chambers L. The practical handbook of genetic algorithms / Ed. L.D. Chambers. Florida: CRC Press, 2000. Vol. II: New frontiers. 421 p.
27. Chambers L. The practical handbook of genetic algorithms / Ed. L.D. Chambers. Florida: CRC Press LLC, 2000. Vol. III: Complex coding systems. 659 p.
28. Cantu-Paz E. Efficient and accurate parallel genetic algorithms. Massachusetts: Kluwer Academic Publishers, 2001. 162 p.
29. Bao Lin Xiaoyan Sun and Sana Salous. Solving Travelling Salesman Problem with an Improved Hybrid Genetic Algorithm. *Journal of Computer and Communications*. 2016. 4. P. 98–06. URL: <http://www.scirp.org/journal/jcc>. ISSN Online: 2327-5227. ISSN Print: 2327-5219.

Надійшла до редакції 31.07.2019