

## АДАПТИВНИЙ КВАНТОВИЙ ГЕНЕТИЧНИЙ АЛГОРИТМ ДЛЯ 0–1 ЗАДАЧИ ПАКУВАННЯ РЮКЗАКА

В.М. ТКАЧУК

**Анотація.** Розглянуто квантовий генетичний алгоритм (QGA), який порівняно з його класичною реалізацією має ряд переваг завдяки швидкодії, невеликому розміру популяції, автоматичному балансу між глобальним та локальним пошуком розв'язку. Основу QGA становлять ідеї традиційних еволюційних алгоритмів, покладені на технологію квантових обчислень, які оперують квантовими бітами, суперпозицією станів та квантовими вимірюваннями. Запропоновано новий QGA, для реалізації якого використано новий адаптивний оператор квантового гейту та технологію відновлення квантової хромосоми під час розв'язання комбінаторних задач з обмеженнями. Для забезпечення ефективності роботи алгоритму виконано метаоптимізацію основних параметрів, покладених в основу його роботи. Можливості запропонованого підходу ілюструють модельні дослідження з використанням широкого спектру тестових 0–1 задач пакування рюкзака.

**Ключові слова:** квантові обчислення, квантовий біт, квантовий генетичний алгоритм, оператор квантового гейту, 0–1 задача пакування рюкзака.

### ВСТУП

Генетичний алгоритм є евристичним алгоритмом, що використовується для наближеного розв'язку задач оптимізації та пошуку [1]. В основу його роботи покладено принципи еволюції природних біологічних систем, зокрема відбору, схрещування та мутації [2–4]. Алгоритм працює із множиною потенційних розв'язків, використовуючи найкращий для його відтворення та послідовного наближення до точного значення в ході еволюції.

У класичній його реалізації мінімальною одиницею інформації є біт-структура із двома станами: 0 та 1. Квантовий генетичний алгоритм (QGA) є новим еволюційним алгоритмом, що ґрунтується на поєднанні ідеології квантових обчислень і технології класичних генетичних алгоритмів [5, 6].

Імовірнісний механізм квантових обчислень забезпечує глобальний пошук розв'язку за швидкої локальної збіжності та невеликого розміру популяції. Хоча теоретичного обґрунтування роботи алгоритму поки що не існує, він показує задовільну ефективність у багатьох областях. Так, цей клас алгоритмів був успішно застосований до широкого спектру задач, як то комбінаторна та функціональна оптимізація, проблеми оптимізації в машинобудуванні, оброблення зображення та багатьох інших [7–11].

Основні ідеї QGA запропонували Narayanan і Moore [12], згідно з якими мінімальною одиницею інформації в квантових обчисленнях є кубіт – квантова система, що може перебувати в двох основних станах:  $|0\rangle$  та  $|1\rangle$ . Кван-

това природа кубіта полягає в принципі суперпозиції, відповідно до якого він може перебувати у довільному стані, що є лінійною комбінацією базових:

$$|q\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle \tag{1}$$

з умовою нормування

$$\alpha_0^2 + \alpha_1^2 = 1.$$

Наслідком принципу суперпозиції є те, що простір станів кубіта незрівнянно більший від простору станів класичного біта. Інформація, що міститься в амплітудах  $\alpha_0$  та  $\alpha_1$ , є власне квантовою частиною інформації. Фактично  $\alpha_0^2$  та  $\alpha_1^2$  — імовірності перебування кубіта в стані  $|0\rangle$  та  $|1\rangle$  відповідно.

Як і класичний, квантовий біт можна виміряти. Результатом вимірювання буде кубіт в одному з основних станів. При цьому важливо, що результат вимірювання не детермінований, як у класичному обчисленні, а ймовірнісний.

Для реалізації QGA використовується матричне подання основних станів кубіта:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

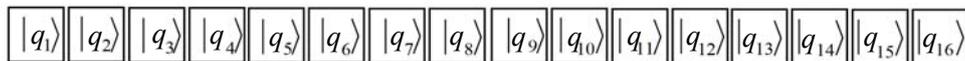
Стан суперпозиції (1) у такому поданні можна записати у вигляді

$$|q\rangle = \begin{pmatrix} \alpha_0 \\ \alpha_1 \end{pmatrix}.$$

Упорядкований набір з  $N$  кубітів формує квантову хромосому. Вектор стану такої хромосоми являє собою розклад по  $2^N$  базових станах  $|i_1, i_2, \dots, i_N\rangle$ ,  $i_1, i_2, \dots, i_N = \{0, 1\}$ :

$$|\Psi\rangle = \sum_{k=1}^N \alpha_{i_1, i_2, \dots, i_N} |i_1, i_2, \dots, i_N\rangle.$$

Оскільки заплутаність квантових станів у QGA не використовується, то хромосома може бути сформована як упорядкований набір незалежних кубітів. Так, якщо  $N = 16$  кубітів, схематично її можна подати таким чином:



Уся інформація про систему кубітів визначається вектором стану  $|\Psi\rangle$ . Єдине, що можна зробити з такою системою, — це перетворити вектор початкового стану  $|\Psi\rangle$  до деякого нового  $|\Psi'\rangle$  за допомогою оператора квантового гейту. Тобто QGA — це перехід системи з початкового в кінцевий стан відповідно до алгоритму, закладеного у роботу квантового гейту. Розв’язок задачі отримується як результат квантового вимірювання вектора кінцевого стану системи  $|\Psi'\rangle$  та переходу до його класичного подання.

## КВАНТОВИЙ ГЕНЕТИЧНИЙ АЛГОРИТМ ТА ЙОГО ВИКОРИСТАННЯ В ЗАДАЧАХ КОМБІНАТОРНОЇ ОПТИМІЗАЦІЇ

У цілому структура QGA аналогічна звичайному генетичному алгоритму, а принципова відмінність полягає лише в особливостях реалізації квантових операторів з урахуванням подання хромосоми у вигляді системи кубітів та принципу суперпозиції.

Загальна схема роботи алгоритму може бути реалізована у вигляді такого алгоритму:

### Алгоритм 1. Квантовий генетичний алгоритм

```
1  $t = 0$ 
2   ініціалізація  $Q(t)$ 
3   вимірювання  $Q(t)$  та перехід до  $P(t)$ 
4   оцінювання пристосованості  $P(t)$ 
5   відновлення  $P(t)$ 
6   відновлення  $Q(t)$ 
7   знаходження  $b$  – найкращого розв'язку в  $P(t)$ 
8   while (умова завершення еволюції)
9      $t = t + 1$ 
10    оновлення  $Q(t)$  за допомогою квантового гейту
11    вимірювання  $Q(t)$  та перехід до  $P(t)$ 
12    оцінювання пристосованості  $P(t)$ 
13    відновлення  $P(t)$ 
14    відновлення  $Q(t)$ 
15    знаходження  $b$  – найкращого розв'язку в  $P(t)$ 
16  end while
```

Тут  $Q(t)$  — квантова популяція розв'язків на момент часу  $t$ ;  $P(t)$  — класичне (бінарне) подання популяції, отримане в результаті квантового вимірювання.

Операцію квантового вимірювання (етапи 3 та 11) детально розглянуто, наприклад у працях [6, 8], й окремого опису не потребує. Необхідність операції відновлення (етапи 5, 6 і 13, 14) та алгоритм її роботи більш детально розглянуто далі.

Для розширення області пошуку та виходу з локальних мінімумів QGA у разі потреби можна доповнити традиційними операторами, притаманними чисто класичному генетичному алгоритму, як то квантової мутації, схрещування чи катастрофи [13].

Зважаючи на ймовірнісний механізм процесу квантового вимірювання у використанні оператора мутації у його класичному розумінні в розгляданому випадку потреби немає. Операція квантового гейту до певної міри виконує роль операції схрещування в її класичному розумінні.

Для реалізації QGA використовується матричне подання квантових хромосом. Наприклад, якщо хромосома складається з  $N$  кубітів, кількість яких визначається розмірністю комбінаторної задачі, то вона може бути реалізована так:

$\alpha_0^1$	$\alpha_0^2$	$\alpha_0^3$	$\alpha_0^4$	$\alpha_0^5$	...	...	...	$\alpha_0^N$
$\alpha_1^1$	$\alpha_1^2$	$\alpha_1^3$	$\alpha_1^4$	$\alpha_1^5$	...	...	...	$\alpha_1^N$

Тут  $\{\alpha_0^i, \alpha_1^i\}$  визначає стан  $i$ -го кубіта  $|q_i\rangle$ , а  $N$  кубітів формують одну особину популяції. Оскільки початковий стан не містить інформації про розв'язок задачі, то найпростішим способом ініціалізації популяції є рівність між собою всіх амплітуд імовірностей станів [12]. Тобто на етапі 2 роботи алгоритму кожен кубіт буде переведений у стан

$$|q\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle.$$

Для пошуку розв'язку в ході еволюції відбувається зміна амплітуд імовірностей станів унаслідок послідовної дії квантових операторів (етапи роботи 9–15). Визначальним серед них є оператор квантового гейту, робота якого реалізована на етапі 10.

## МЕТА ДОСЛІДЖЕННЯ

Метою дослідження була побудова нових квантових операторів у реалізації QGA для забезпечення, порівняно із традиційними підходами, більшої ефективності роботи алгоритму як за часом роботи, так і за ефективністю пошуку оптимального значення.

Усю інформація про задачу та алгоритм її розв'язку закладено у квантовому гейті, тому його робота є визначальною для побудови будь-якого QGA. Він маніпулює амплітудами ймовірностей квантових станів, забезпечуючи виконання умови нормування.

Під час реалізації обертання квантових станів для визначення кута повороту традиційним є використання таблиці пошуку, що обмежує універсальність роботи алгоритму. Крім того, фіксоване значення кута негативно впливає на швидкість збіжності, тому QGA інколи реалізують як адаптивний процес зміни його величини в ході еволюції. Більш ефективним, як показали виконані дослідження, є закладання адаптивного характеру у сам алгоритм роботи оператора.

Роботу запропонованого оператора можна поділити на два етапи. На першому етапі збільшується амплітуда ймовірності вибраного квантового стану  $b$ :

$$(\alpha_k^b)' = \sqrt{[\alpha_k^b]^2 + \mu \cdot (1 - \alpha_k^b)}, \quad (2)$$

де  $k \in \{0, 1\}$ .

Стан  $b$  визначається  $k$ -м значенням класичного подання найкращої особини популяції, отриманої на попередній ітерації еволюції в часі. Функціональна залежність (2) забезпечує також той факт, що амплітуда ймовірності  $\alpha_k^b$  не може перевищувати 1. Значення параметра  $\mu$  лежить у межах  $[0, 1]$  і підбирається за результатами попередніх досліджень.

На другому етапі необхідно зменшити амплітуду ймовірності іншого стану кубіта для забезпечення виконання умови нормування. Загалом алго-

ритм роботи оператора до квантової хромосоми, що складається з  $N$  кубітів, можна реалізувати таким чином:

**Алгоритм 2.** Оператор квантового гейту

```

1 for  $i$  in  $1, \dots, N$  do
2    $bestamp = i$  -е значення гена найкращої особини популяції
3   if  $bestamp = 0$ 
4      $\alpha_0^i = \sqrt{[\alpha_0^i]^2 + \mu(1 - \alpha_0^i)}$ 
5      $\alpha_1^i = \sqrt{1 - [\alpha_0^i]^2}$ 
6   end if
7   if  $bestamp = 1$ 
8      $\alpha_1^i = \sqrt{[\alpha_0^i]^2 + \mu(1 - \alpha_1^i)}$ 
9      $\alpha_0^i = \sqrt{1 - [\alpha_0^i]^2}$ 
10  end if
11 end for

```

На етапі 4 (8) алгоритму відбувається адаптивний поворот вектора квантового стану, що відповідає найкращій особині популяції, а на етапі 5 (9) — перерахунок амплітуди іншої імовірності для забезпечення виконання умови нормування.

Таким чином, у кожному новому поколінні забезпечується збільшення ймовірності того, що в результаті спостереження генеруються класичні особини, більш схожі на найкращу. За такого алгоритму роботи також можна обійтися без таблиці пошуку, що є одним із принципових недоліків QGA.

Адаптивний механізм, покладений в основу роботи квантового гейту, як показали моделювання, дозволяє для розв'язання 0–1 задачі пакування рюкзака покласти  $\mu = 1$ .

## ЗАДАЧА ПАКУВАННЯ РЮКЗАКА

Проблема 0–1 пакування рюкзака є  $NP$  повною задачею дискретної комбінаторної оптимізації, яка традиційно використовується для тестування пошукових алгоритмів. Вона зводиться до знаходження для скінченної множини  $N$  речей такого бінарного вектора заповнення рюкзака  $X = \{x_1, x_2, x_3, \dots, x_N\}$ , який забезпечує максимальне значення функції

$$f(X) = \sum_{i=1}^N p_i x_i \quad (3)$$

з обмеженням

$$\sum_{i=1}^N \omega_i x_i \leq C, \quad (4)$$

де  $p_i$  — вартість  $i$ -го предмета;  $\omega_i$  — маса  $i$ -го предмета;  $C$  — максимально можлива маса рюкзака.

У процесі ініціалізації популяції, чи в ході її еволюції завжди є ймовірність отримати ряд «поганих» особин (3), які не задовольняють умову (4).

Для врахування цього використовується або метод штрафних функцій, або метод відновлення, який і буде використано в роботі. Процедура відновлення зводиться до видалення випадковим чином елементів з рюкзака доти, доки не буде задоволено обмеження (4).

Після відновлення функція пристосованості буде такою:

$$f(X') = \sum_{i=1}^N p_i x'_i,$$

де  $X' = \{x'_1, x'_2, x'_3, \dots, x'_N\}$  — відновлена версія бінарного вектора  $X$ .

Дослідження показали, що жадібний алгоритм відновлення, коли всі речі, що пакуються, сортуються відповідно до вагових коефіцієнтів  $p_i/\omega_i$  та видаляються з рюкзака відповідно до їх росту, веде до зменшення різноманітності популяції. Наслідком цього є зменшення ефективності глобального пошуку та погіршення кінцевих результатів.

Процедура відновлення в QGA принципово інша, ніж в генетичному алгоритмі, бо потребує коригування і квантової хромосоми. Вона може бути розбита на два етапи і потребувати значних обчислювальних ресурсів. Відсоток відновлених особин може змінюватися від 0 до 100% і є одним з параметрів роботи оператора. Надалі для простоти реалізації вважатимемо, що відновлюються всі хромосоми, що не задовольняють умову за масою (4).

На першому традиційному етапі, якщо рюкзак надто важкий, елементи видаляються з нього доти, доки не буде задоволено обмеження (4). Якщо рюкзак занадто легкий, то процедура відновлення додає елементи в рюкзак доти, доки це дозволяє обмеження маси.

Схематично перший етап можна подати у вигляді такого алгоритму.

### Алгоритм 3. Відновлення класичної хромосоми $P(t)$

```

1  переповнення_рюкзака=false
2  if  $\left(\sum_{i=1}^N \omega_i x_i > C\right)$  then переповнення_рюкзака=true
3  while (переповнення_рюкзака=true)
4       $i = \text{random}\{1..N\}$  — вибір випадковим чином пред-
мета з рюкзака
5       $x_i = 0$  — видалення предмета із рюкзака
6      if  $\left(\sum_{i=1}^N \omega_i x_i \leq C\right)$  then переповнення_рюкзака=false
7  end while
8  for  $i$  in  $\{1, 2, \dots, N\}$ 
9      if  $\left(\sum_{i=1}^N \omega_i x_i + \omega_i > C\right)$  OR  $(x_i = 0)$  then continue
10      $x_i = 1$ 
11 end for

```

Етапи 2–7 відповідають за видалення випадковим чином елементів із рюкзака, поки не виконається обмеження (4). На етапах 8–11, якщо це можливо, у рюкзак послідовно додаються додаткові елементи.

На другому етапі необхідно відкоригувати амплітуди ймовірності квантової хромосоми відповідно до відновленої хромосоми  $X'$ . Процес відновлення реалізовано у вигляді такого алгоритму:

**Алгоритм 4.** Відновлення квантової хромосоми  $Q(t)$

```

1  for  $i$  in  $\{1, 2, \dots, N\}$ 
2    if  $(x_i = 1)$ 
3       $\alpha_i^0 = \beta$ 
4       $\alpha_i^1 = \sqrt{1 - \beta^2}$ 
5    end if
6    if  $(x_i = 0)$ 
7       $\alpha_i^0 = \sqrt{1 - \beta^2}$ 
8       $\alpha_i^1 = \beta$ 
9    end if
10 end for

```

Тут  $\beta \in [0, 1]$  — параметр роботи алгоритму. Проведені дослідження показали, що він не залежить від розміру системи  $N$ , а вплив його значення на ефективність роботи QGA ілюструє рис. 1. Рівень кореляції вхідних даних не впливає на параметр  $\beta$  і оптимальним в подальших моделюваннях в даній роботі припускається  $\beta = 0,971$ .

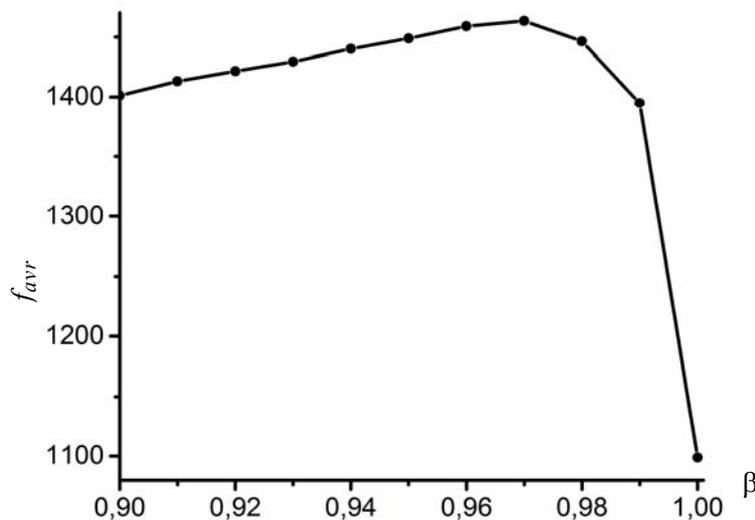


Рис. 1. Вплив  $\beta$  на середню за 100 запусками QGA пристосованість найкращої особи для  $N = 100$  слабкорельованих вхідних даних

## РЕЗУЛЬТАТИ ЧИСЛОВОГО ЕКСПЕРИМЕНТУ

QGA реалізовано мовою програмування C++, моделювання виконано на процесорі Intel Celeron CPU G1840 2,80 GHz, 4,0 Гбіт оперативної пам'яті. Для ілюстрації ефективності роботи алгоритму розглянуто ряд задач комбінаторної оптимізації різної розмірності та ступеня кореляції вхідних даних.

Потужність популяції  $s$  у QGA є важливим параметром роботи та критичним особливо за великих значень  $N$ : чим більше його значення, тим більшою має бути різноманітність потенційних розв'язків, що охоплюють всю область пошуку. Надалі для реалізації QGA взято  $s = 10$ , що забезпечує 5000 звернень до функції пристосованості у процесі еволюції в часі  $t = 500$ .

Наведені нижче параметри, де це не оговорено окремо, оцінено як результат усереднення за 1000 запусками алгоритму. Ефективність роботи оцінено за середньою пристосованістю найкращої особини популяції  $f_{avr}$  та середньоквадратичним відхиленням  $f_{sd}$ . Їх типову поведінку в часі показано на рис. 2 і 3.

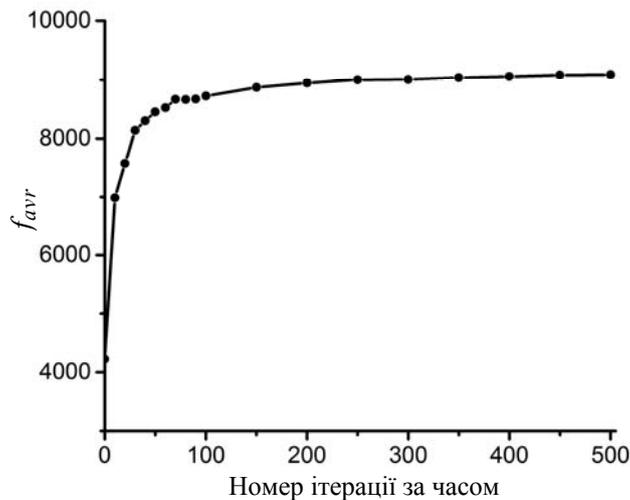


Рис. 2. Усереднена за 100 запусками алгоритму еволюція середньої пристосованості найкращої особини  $f_{avr}$  для  $N = 100$  нескорельованих вхідних даних

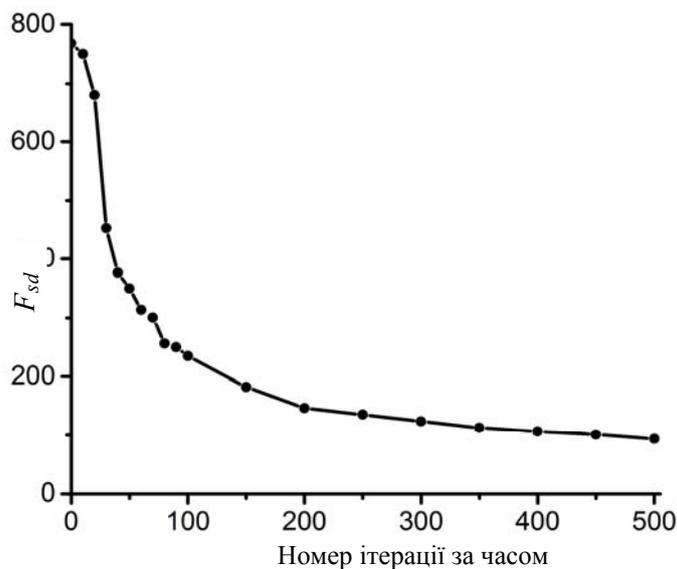


Рис. 3. Усереднена за 100 запусками алгоритму еволюція середньоквадратичного відхилення найкращої особини  $f_{sd}$  для  $N = 100$  нескорельованих вхідних даних

Ініціалізована на початку популяція ( $t = 0$ ) включає всі розв'язки з однаковою ймовірністю (рис. 4). Це означає, що робота QGA починається з випадкового пошуку, а окремі піки на розподілі зумовлені дискретним характером оптимізації та роботою оператора відновлення квантової хромосоми.

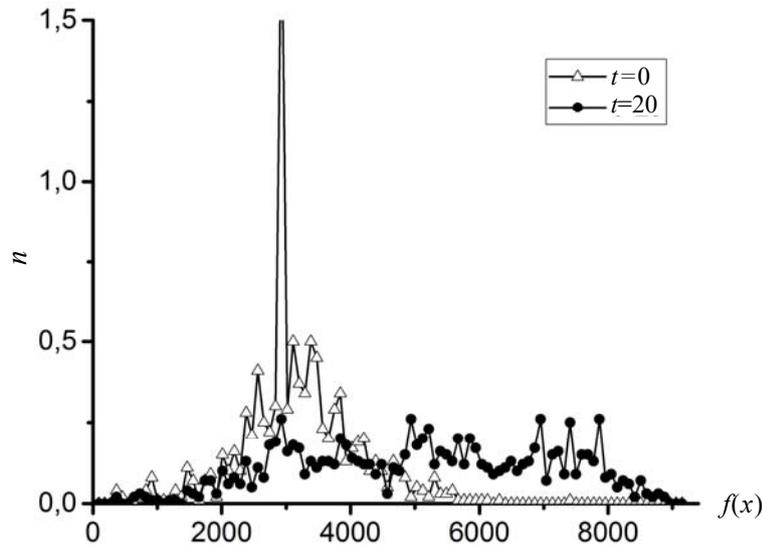


Рис. 4. Розподіл особин популяції по області пошуку на початкових етапах еволюції

У міру еволюції в часі характер розподілу зазнає принципових змін (рис. 5).

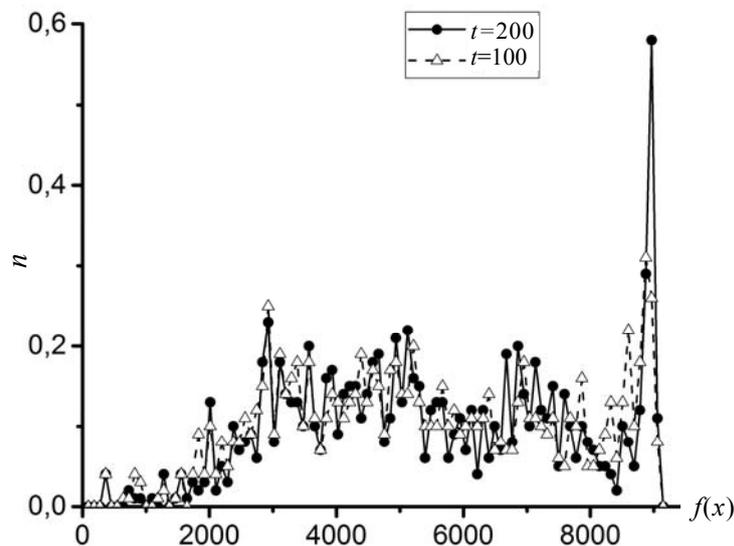


Рис. 5. Розподіл особин популяції по області пошуку на кінцевих етапах еволюції

Зростає ймовірність знаходження особини в області точного розв'язку, який в розгляданому випадку дорівнює 9147. Велика їх кількість тут веде до ефективного локального пошуку за рахунок квантового гейту. Оператор відновлення квантової хромосоми забезпечує різноманітність популяції.

Так, у разі його вилучення із QGA, як показують результати дослідження, різноманітність популяції набагато менша. У результаті зростає ймовірність потрапляння в локальні оптимуми та зменшується ефективність глобального пошуку.

Підсумовуючи, можна відзначити, що QGA починає свою роботу з глобального пошуку та автоматично переходить до локального завдяки зміні структурних характеристик розподілу особин популяції.

Статистичні результати, включаючи середнє значення найкращої особи популяції  $f_{avr}$ , його середньоквадратичне відхилення  $f_{sd}$ , найкращий та найгірший результати, середній час роботи алгоритму  $t_{avr}$ , отримані під час комбінаторної оптимізації тестових завдань різної розмірності, зведено в табл. 1 і 2.

**Таблиця 1.** Результати оптимізації тестових задач малої розмірності

Розмірність $N$	$f_{avr}$	$t_{avr}$ , с	Точне значення
4	35	0,0015	35
10	52	0,0039	52
15	481	0,0061	481
20	1024	0,0078	1024
23	9767	0,0093	9767

**Таблиця 2.** Результати оптимізації тестових задач великої розмірності

Нескорельовані дані						
$N$	$f_{avr}$	$f_{sd}$	$t_{avr}$ , с	Найкраще значення	Найгірше значення	Точне значення
100	9084,1	98,2	0,049	9147	8628	9147
200	10763,0	331,9	0,111	11227	8861	11238
200*	11178,0	68,28	0,591	11238	10814	
500	14668,0	942,2	0,284	18551	12012	28857
500*	16963,0	908,3	1,454	20127	14539	
Слабоскорельовані дані						
100	1462,5	51,7	0,050	1514	1208	1514
200	1476,7	31,28	0,114	1600	1387	1634
200*	1530,5	25,11	0,602	1624	1449	
500	3283,8	84,49	0,276	3635	3044	4566
500*	3448,2	77,80	1,434	3805	3161	
Сильноскорельовані дані						
100	2338,4	48,74	0,050	2397	2292	2397
200	2552,5	72,68	0,110	2697	2297	2697
200*	2628,4	58,10	0,599	2697	2397	
500	5065,7	156,0	0,271	5712	4715	7117
500*	5394,7	151,98	1,447	5914	4917	

\* Моделювання виконано за розміру популяції 50 особин.

Як впливає з наведених результатів, середній час роботи  $t_{avr}$  залежить лише від розміру області пошуку, яка визначається розміром квантової хромосоми  $N$ .

Роль оператора відновлення ілюструють дані табл. 3. Така реалізація QGA потребує підбору параметра  $\mu$  та показує значно гірші результати. Використання операції відновлення квантової хромосоми веде до збільшення часу роботи алгоритму майже в два рази.

**Таблиця 3.** Результати оптимізації тестових задач великої розмірності в разі реалізації QGA без оператора відновлення квантової хромосоми ( $s = 10$ )

Розмірність $N$	$f_{avr}$	$f_{sd}$	$t_{avr}, c$
<b>Нескорельовані дані</b>			
100	8253,9	486,6	0,027
200	8758,0	701,1	0,055
500	14425,0	1472,1	0,145
<b>Слабоскорельовані дані</b>			
100	1271,9	93,3	0,027
200	1297,9	82,9	0,054
500	3004,4	145,3	0,135
<b>Сильноскорельовані дані</b>			
100	2065,3	112,3	0,028
200	2147,0	123,6	0,055
500	4359,1	210,2	0,135

У цілому отримані результати порівняно з даними, доступними в літературі [6, 14], показують підвищення ефективності QGA за рахунок використання запропонованих квантових операторів.

## ВИСНОВКИ

У роботі побудовано новий QGA та проілюстровано ефективність його роботи з використанням набору тестових даних різної розмірності та рівня кореляції. Запропонований підхід ілюструє хорошу можливість глобального пошуку завдяки використанню процедури відновлення квантової хромосоми та оператора квантового вимірювання. Швидка локальна збіжність забезпечується адаптованим алгоритмом роботи оператора квантового гейту, який для своєї реалізації не потребує використання таблиці пошуку.

Експериментальні результати 0–1 задачі пакування рюкзака ілюструють можливості застосування QGA до задач різної розмірності. Результати, отримані за значних розмірів досліджуваних систем ( $N = 500$ ), указують на необхідність збільшення розміру популяції та пошуку додаткових методів поліпшення роботи алгоритму, що буде предметом наступних досліджень.

## ЛІТЕРАТУРА

1. Holland J.H. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence / J.H. Holland. — Ann Arbor, MI: University of Michigan Press, 1975.
2. Whitley Darrell. A genetic algorithm tutorial / Darrell Whitley. — Statistics and computing. — 1994. — Vol. 4, N 2. — P. 65–85.

3. Michalewicz Z. Genetic Algorithms + Data Structures = Evolution Programs / Z. Michalewicz // New York: Springer, 3rd (extended) edition, 1996.
4. Jantos P. Evolutionary algorithms for global parametric fault diagnosis in analogue integrated circuits / P. Jantos, D. Grzechca, J. Rutkowski // Bull. Pol. Ac.: Tech. — 2012. — Vol. 60, N 1. — P. 133–142.
5. Han K.-H. Genetic quantum algorithm and its application to combinatorial optimization problem / K.-H. Han, J.-H. Kim // Proceedings of IEEE Congress on Evolutionary Computation, USA. — 2000. — Vol. 2. — P. 1354–1360.
6. Nowotniak R. Higher-Order Quantum-Inspired Genetic Algorithms / R. Nowotniak, J. Kucharski // Federated Conference on Annals of Computer Science and Information Systems. — 2014. — P. 465–470.
7. Jopek Ł. Zastosowanie kwantowych algorytmów genetycznych do selekcji cech / Ł. Jopek, R. Nowotniak, M. Postolski et al. // Automatyka, Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie. — 2009. — Vol.13, N 3. — P. 1219–1231.
8. Talbi H. A Quantum-Inspired Evolutionary Algorithm for Multiobjective Image Segmentation / H. Talbi, M. Batouche, A. Draa // International Journal of Mathematical, Physical and Engineering Sciences. — 2007. — Vol. 1, N 2. — P. 109–114.
9. Li B.B. A hybrid quantum-inspired genetic algorithm for multiobjective flow shop scheduling / B.B. Li, L. Wang // IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics. — 2007. — Vol. 37, N 3. — P. 576–591.
10. Lau T. Quantum-inspired evolutionary algorithm approach for unit commitment / T. Lau, C. Chung, K. Wong, T. Chung, S. Ho // IEEE Trans. on Power Systems. — 2009. — Vol. 24, N 3. — P. 1503–1512.
11. Su-Hua L. Application of quantum-inspired evolutionary algorithm in reactive power optimization / L. Su-Hua, W. Yao-Wu, P. Lei, X. Xin-Yin // Relay. — 2005. — Vol. 33. — P. 30–35.
12. Narayanan A. Quantum-inspired genetic algorithms / A. Narayanan, M. Moore // IEEE Evolutionary Computation. — 1996. — Vol.1. — P. 61–66.
13. Wang H. The improvement of quantum genetic algorithm and its application on function optimization / H. Wang, J. Liu, J. Zhi, C. Fu // Math. Probl. Eng. — 2013. — Vol. 2013. — P. 1–10.
14. Kuk-Hyun Han. Quantum-Inspired Evolutionary Algorithm for a Class of Combinatorial Optimization / Kuk-Hyun Han, Jong-Hwan Kim // IEEE Transactions on Evolutionary Computation. — 2002. — Vol. 6, N 6. — P. 580–593.

Надійшла 12.04.2018