

УДК 681.3

## ПЕРСПЕКТИВНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ НА ОСНОВЕ МЕТОДОВ, ИНСПИРИРОВАННЫХ ПРИРОДНЫМИ СИСТЕМАМИ

**В.М. Курейчик**

*Южный федеральный университет, Таганрог, Россия  
kur@tgn.sfedu.ru*

### **Аннотация**

Рассматривается новая технология решения оптимизационных и комбинаторно-логических задач искусственного интеллекта (гибридных и параллельных) на графовых моделях на основе эволюционных, роевых, квантовых и генетических алгоритмов. Это позволяет получать наборы локально-оптимальных решений и строить эвристические алгоритмы с полиномиальной скоростью роста количества операций в зависимости от объема входных данных и частично решать проблему преждевременной сходимости.

***Ключевые слова:** эволюционные, квантовые, роевые, генетические алгоритмы, бионический поиск, графовые модели*

### **Введение**

При решении оптимизационных и комбинаторно-логических задач проектирования, конструирования и искусственного интеллекта эффективно используются стратегии, концепции, методы, механизмы эволюционного моделирования бионического поиска и методов, инспирированных природными системами [1-6]. Бионика – решение инженерных и технических задач на основе изучения структуры и жизнедеятельности живых организмов. Бионический поиск с точки зрения преобразования информации при решении задач на графах – это последовательное преобразование одного конечного нечеткого множества альтернативных решений в другое. Само преобразование называется алгоритмом поиска, в основе которого можно использовать генетические, квантовые, муравьиные, пчелиные и др. алгоритмы. В основе генетического алгоритма (ГА) лежит случайный, направленный или комбинированный поиск. Такие алгоритмы эффективно используют информацию, накопленную в процессе эволюции, для получения квазиоптимальных и оптимальных решений. Использование идей квантовой механики позволяет при поиске решений в задачах оптимизации использовать подходы параллельных вычислений. Согласно известному принципу суперпозиции система может одновременно находиться во всех возможных состояниях. Производя над одним состоянием произвольные действия, мы производим это одновременно над заданным множеством состояний [7-10]. При решении оптимизационных задач на графах предлагается новая технология на основе совместного бионического, квантового и роевого поиска. Описаны новые архитектуры и принципы такого поиска. Это позволяет расширить область поиска решений без увеличения времени работы и сократить преждевременную сходимость алгоритмов, повысить эффективность и качество получаемых решений.

### **1 Генетические алгоритмы**

Генетические алгоритмы отличаются от других оптимизационных и поисковых методов и алгоритмов [2-6]:

- анализируют и преобразуют закодированное множество исходных параметров;
- осуществляют поиск из части популяции или множества популяций (множества альтернативных решений), а не из одного решения;
- используют целевую функцию (функцию пригодности или приспособленности), а не ее различные приращения для оценки качества альтернативных решений;
- используют детерминированные, вероятностные и комбинированные правила анализа оптимизационных задач.

Генетические алгоритмы манипулируют популяцией хромосом на основе механизма натуральной эволюции. Приведем формальное определение ГА:

$$ГА = (P_i^o, N, P_{i,k}^T, T, L_j, A, (\text{ЦФ, ОГР, ГУ}), \text{ГО}, t),$$

где  $P_i^o$  – исходная популяция хромосом альтернативных решений,  $P_i^o = (P_{i1}^o, P_{i2}^o, \dots, P_{in}^o)$ ,  $P_{i1}^o \in P_i^o$  – хромосома (альтернативное решение), принадлежащее  $i$ -ой исходной популяции;  $N$  – мощность популяции, т.е. число входящих в нее хромосом,  $N = |P_i^T|$ ;  $P_{ik}^T \in P_i^T$  –  $k$ -я хромосома, принадлежащая  $i$ -ой популяции, находящейся в  $T$  поколении эволюции;  $T = 0, 1, 2, \dots$  – номер поколения, проходящего популяцией во время эволюции, иногда число поколений связывают с числом генераций генетического алгоритма, обозначаемых буквой  $G$ ;  $L_j$  – длина  $i$ -ой хромосомы (альтернативного решения), т.е. число генов (элементов, входящих в закодированное решение, представленное в заданном алфавите), например,  $|P_i^T| = L_j$ ;  $A$  – произвольный абстрактный алфавит, в котором, кодируются хромосомы, например,  $A_1 = \{0, 1\}$ ,  $A_2 = \{0, 1, 2, \dots, 10\}$ ,  $A_3 = \{0, 1, 2, *\}$ ,  $A_4 = \{A, B, C, D\}$ , здесь  $*$  – метка, означающая любой символ в алфавите  $A_3$  из  $0, 1, 2$ ; (ЦФ, ОГР, ГУ) – целевая функция, ограничения и граничные условия, которые определяются на основе заданной модели исходной решаемой задачи; ГО – генетические операторы,  $t$  – критерий окончания работы ГА.

## 2 Квантовые алгоритмы

В последнее время появились новые подходы решения  $NP$  – полных проблем, основанные на методах квантового поиска [7-10]. Квантовый поиск анализирует неструктурированные проблемы, которые в общем виде формулируются следующим образом [7]. Задана функция  $f(x)$ , аргументы  $x$  – целые числа,  $x = 1, 2, \dots, N$ , причем  $f(x)$  принимает значение ноль во всех случаях, кроме  $x = w$ . Необходимо найти значение  $w$ , используя наименьшее число запросов к  $f(x)$ . Задачи такого типа при небольшом  $x < 100$  решаются на основе полного перебора (исчерпывающего поиска) или методом проб и ошибок.

Идею и структуру квантового алгоритма предложил Л. Гровер [7,8]. Согласно [8] при решении неструктурированной проблемы поиска существует «оракул», определяющий является ли рассматриваемое решение искомым. Л. Гровер рассматривает  $N$  целых чисел индекса  $x = 1, 2, \dots, N$  как набор ортогональных векторов  $\vec{x} = \vec{1}, \vec{2}, \dots, \vec{k}$  в  $N$ -размерном пространстве Хильберта. Этот шаг алгоритма на языке вычисления кванта ставит в соответствие каждому возможному индексу уникальный собственный вектор. Первоначально готовится пространство,

$$\vec{S} = \frac{1}{\sqrt{N}} \sum_{x=1}^N \vec{x},$$

т.е. квантовый регистр памяти, содержащий определенное количество суперпозиций, равное количеству всех  $N$  [7,8].

Для реализации поиска это квантовое пространство развивается в общую суперпозицию, которая концентрируется в  $\vec{t}$  векторе, определяющем путь до цели поиска. Предлагается процедура квантового кругооборота  $U$ . Другими словами, если имеется отличное от нуля

совпадение между стартовым пространством  $\vec{S}$  и целевым  $\vec{t}$ , то есть  $\vec{t} |U| \vec{S} \neq 0$ , тогда можно использовать унитарную процедуру  $U$  для выполнения классического поиска цели. Л. Гровер предлагает использовать  $U$  и  $f(x)$ , чтобы построить увеличивающий амплитуду оператор  $Q$ , который изменяет амплитуду вероятности от «не-цели» векторов  $\vec{S} \neq \vec{t}$  в цель  $\vec{S} = \vec{t}$  [7]. Поведение «оракула» в алгоритме квантового поиска моделируется возвратной функцией  $f(x) = 0$ , для всех  $x$ ,  $w$  и  $f(x) = 1$ , для  $x = w$ .

Для решения  $NP$ -полных проблем на графах предлагается анализировать структуру графа чтобы «выращивать» полные решения, рекурсивно расширяя последовательные частичные решения.

Приведем модифицированный алгоритм квантового поиска [11-13].

- 1) Начало.
- 2) Ввод исходных данных.
- 3) Проверка условий существования инвариантных частей в графе.
- 4) Анализ математической модели и на его основе построение дерева частичных решений.
- 5) Суперпозиция частичных решений на основе жадной стратегии и квантового поиска.
- 6) В случае наличия тупиковых решений – последовательный поиск с пошаговым возвратом.
- 7) Если набор полных решений построен, то переход к 8, если нет, то к 5.
- 8) Лексикографический перебор полных решений и выбор из него оптимального или квази-оптимального решения.
- 9) Конец работы алгоритма.

Алгоритмы квантового поиска весьма чувствительны к изменениям и перестановкам входных параметров исходной модели. Это говорит о том, что, например, для одного вида модели объекта, представленного матрицей, можно получить решение с одним локальным оптимумом. Для этой же матрицы с переставленными строками и столбцами можно получить другое решение с лучшим локальным оптимумом. Следует отметить, что, изменяя параметры, алгоритмы и схему квантового поиска, в некоторых случаях можно выходить из локальных оптимумов. Эта проблема продолжает оставаться одной из важнейших во всех методах оптимизации.

### 3 Муравьиный алгоритм

Данный класс алгоритмов разрабатывался в рамках научного направления, которое можно назвать «природные вычисления» [13]. Исследования в этой области начались в середине 90-х годов XX века, автором идеи является Марко Дориго [13, 14]. В основе этой идеи лежит моделирование поведения колонии муравьев. Колония муравьев представляет собой систему с очень простыми правилами автономного поведения особей. Однако, несмотря на примитивность поведения каждого отдельного муравья, поведение всей колонии оказывается достаточно разумным. Основой поведения муравьиной колонии служит низкоуровневое взаимодействие, благодаря которому, в целом, колония представляет собой разумную много-агентную систему. Взаимодействие определяется через специальное химическое вещество – феромон, откладываемый муравьями на пройденном пути. При выборе направления движения муравей исходит не только из желания пройти кратчайший путь, но и из опыта других муравьев, информацию о котором получает непосредственно через уровень феромонов на каждом пути. Концентрация феромона определяет желание особи выбрать тот или иной путь. В работе [14] подробно описаны применения муравьиного алгоритма, его модификации и результаты экспериментальных исследований.

#### 4 Пчелиный алгоритм

Данный алгоритм моделирует поведение пчел в естественной среде. Идея пчелиного алгоритма заключается в том, что все пчёлы на каждом шаге будут выбирать как элитные участки для исследования, так и участки в окрестности элитных, что позволит, во-первых, разнообразить популяцию решений на последующих итерациях, во-вторых, увеличить вероятность обнаружения решений, близких к оптимальным [15]. Приведем основные понятия пчелиного алгоритма: источник нектара (цветок, участок); фуражиры (рабочие пчелы); пчелы-разведчики.

Источник нектара характеризуется значимостью, определяемой различными параметрами. Фуражиры закреплены за источниками нектара. Количество всех пчел в этих участках больше, чем на остальных. Среднее количество разведчиков в рое составляет 5-10%. Вернувшись в улей, пчелы «обмениваются информацией» посредством танцев на, так называемой, закрытой площадке для танцев [16]. Если разведчики нашли лучшие источники нектара, то за ними могут быть закреплены фуражиры.

Приведём словесное описание алгоритма пчёл. Условия остановки алгоритма определяется пользователем и зависит от требуемого времени получения результата.

- 1) Генерация участков для поиска нектара.
- 2) Оценка полезности участков.
- 3) Выбор участков для поиска в их окрестности.
- 4) Отправка фуражиров.
- 5) Поиск в окрестностях источников нектара.
- 6) Отправка пчёл-разведчиков.
- 7) Случайный поиск.
- 8) Оценка полезности новых участков.
- 9) Если условие останова не выполняется, то п. 2.
- 10) Конец работы алгоритма.

Таким образом, ключевой операцией алгоритма пчёл является совместное исследование перспективных областей и их окрестностей. В конце работы алгоритма популяция решений будет состоять из двух частей: пчелы с лучшими значениями целевой функции (ЦФ) элитных участков, а также группы рабочих пчёл со случайными значениями ЦФ. Зависимость временной сложности пчелиного алгоритма от числа вершин –  $O(n^2)$ .

#### 5 Метод роя частиц

Метод роя частиц (МРЧ, Particle Swarm Optimization - PSO) - метод численной оптимизации, для использования которого не требуется знать точного градиента оптимизируемой функции. МРЧ был доказан Кеннеди, Эберхартом и Ши [17] и изначально предназначался для имитации социального поведения. МРЧ оптимизирует функцию, поддерживая популяцию возможных решений, называемых частицами, и перемещая эти частицы в пространстве решений согласно простой формуле. Перемещения подчиняются принципу наилучшего найденного в этом пространстве положения, которое постоянно изменяется при нахождении частицами более выгодных положений. В основу метода положен тот факт, что частицы стремятся к некоторому центру «притяжения», постепенно замедляя свое движение. В методе оптимизации роем частиц агентами являются частицы в пространстве параметров задачи оптимизации. В каждый момент времени частицы имеют в этом пространстве некоторое положение и вектор скорости. Для каждого положения частицы вычисляется соответствующее значение целевой функции, и на этой основе по определенным правилам частица меняет свое положение и скорость в пространстве поиска.

## 6 Алгоритмы решения задач на графах

Для решения  $NP$ -полных проблем на графах предлагается анализировать структуру графа, чтобы «выращивать» полные решения, рекурсивно расширяя последовательные частичные решения. В задачах на графах важным является нахождение инвариантов. Инвариант графа  $G = (X, U)$ , где  $|X| = n$ , а  $|U| = m$ , это число, связанное с  $G$ , которое принимает одно и то же значение на любом графе, изоморфном  $G$ . Очевидно, что число вершин  $n$  и ребер  $m$  являются простейшими инвариантами графа. Полный набор инвариантов определяет граф с точностью до изоморфизма [18]. Основными инвариантами графа являются следующие числа: цикломатическое, хроматическое, внешней и внутренней устойчивости, клик, полноты, ядер, планарности и т.д.

Покажем на примере графа (рисунок 1) определение числа клик на основе квантового алгоритма [7]. Клика – это полный подграф, содержащий наибольшее число ребер. Соответственно число полноты – это наибольшее число вершин в клике.

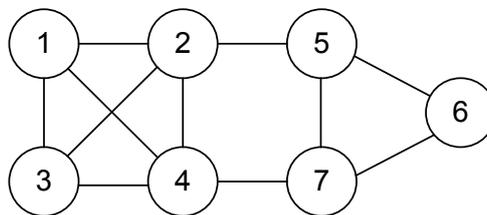


Рисунок 1 - Граф  $G$

При определении клик графа найдем частичные решения, для каждой вершины графа рекурсивно расширяя «хорошие» решения и устраняя тупиковые. На первом шаге квантового поиска для вершины 1 получим следующие частичные решения:

- $\{1, 2\}, \{1, 3\}, \{1, 4\}$  - 1 уровень,
- $\{1, 2, 3\}, \{1, 2, 4\}, \{1, 3, 4\}$  - 2 уровень,
- $\{1, 2, 3, 4\}$  - 3 уровень.

В результате после суперпозиции частичных решений получим клику  $Q_1 = \{1, 2, 3, 4\}$ ,  $|Q_1| = 4$ . На втором шаге для вершины 2 получим:  $\{2, 5\}$ ,  $Q_2 = \{2, 5\}$ ,  $|Q_2| = 2$ . Для третьей вершины частичных решений нет. Для четвертой вершины имеем  $\{4, 7\}$ ,  $Q_3 = \{4, 7\}$ ,  $|Q_3| = 2$ .

Для пятой вершины получим следующие частичные решения:

- $\{5, 6\}, \{5, 7\}$  - 1 уровень,
- $\{5, 6, 7\}$  - 2 уровень.

После суперпозиции частичных решений получим клику  $Q_4 = \{5, 6, 7\}$ ,  $|Q_4| = 3$ .

Итак, для графа  $G$  (рисунок 1) построено семейство клик  $Q = \{Q_1, Q_2, Q_3, Q_4\}$ . Причем число полноты графа равно 4.

Рассмотрим решение раскраски графа на основе квантового поиска. Раскраской графа  $G = (X, U)$  называется разбиение графа на такие непересекающиеся подмножества вершин  $X_1 \cap X_2 \cap \dots \cap X_e = \emptyset$ ,  $X_1 \cup X_2 \cup \dots \cup X_e = X$ , что вершины внутри каждого подмножества несмежны. Наименьшее число подмножеств  $X_i$  при раскраске называется хроматическим числом графа.

Алгоритм основан на нахождении частичной раскраски для подмножеств вершин. Для определения полной раскраски производится рекурсивно расширение частичных окрасок с возвратом назад в случае тупиковых решений. Элементарный способ состоит в рассмотрении дерева поиска частичных решений заданной глубины. Решение по раскраске находится на «ветвях и листьях» этого дерева.

Например, пусть задан граф  $G = (X, U)$  из [3], где  $|X| = 7$  (рисунок 2).

Построим дерево частичной раскраски (рисунок 3).

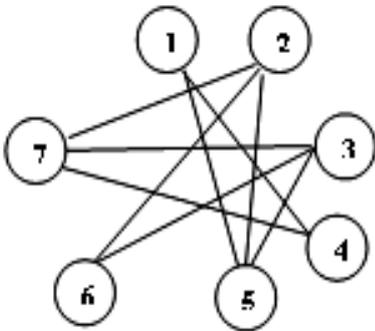


Рисунок 2 - Граф  $G$  из [3]

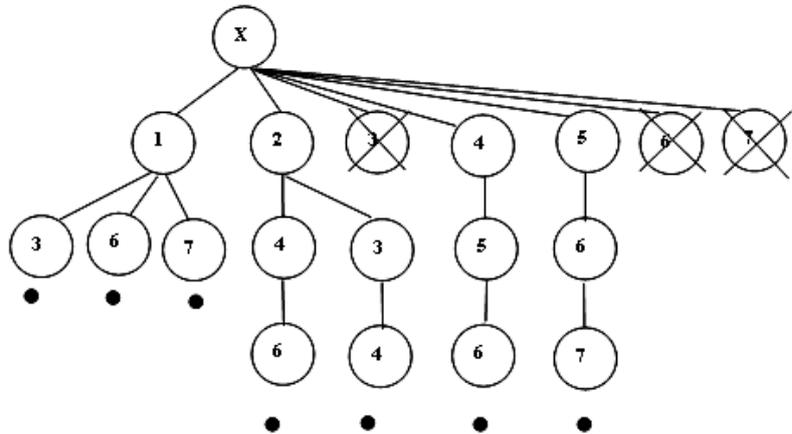


Рисунок 3 - Дерево раскраски графа  $G$  из [3]

Корневая вершина дерева содержит все вершины  $X$  графа  $G$ . На первом уровне дерева располагаются все вершины графа. Отметим, что возможно случайное и упорядоченное расположение вершин. Наиболее вероятным кажется упорядоченность вершин по уменьшению локальных степеней. Хотя, конечно, требуются экспериментальные исследования на графах различного вида. Выбираем на первом уровне вершину 1 и определяем возможные частичные решения по раскраске, включающие эту вершину. Следуя по дереву вниз, видно, что имеем три частичных решения  $\{1, 3\}$ ,  $\{1, 6\}$ ,  $\{1, 7\}$ . Дальнейшее расширение этих частичных решений невозможно. Это отмечено знаком «•». Переходим к вершине 2 и строим два новых частичных решения  $\{2, 4, 6\}$  и  $\{2, 3, 4\}$ . Вершина 3 частичных решений не дает. Вершина 4 дает частичное решение  $\{4, 5, 6\}$  и вершина 5 -  $\{5, 6, 7\}$ . Вершины 6 и 7 частичных решений не дают. Для получения альтернативных раскрасок предлагается операция суперпозиции. Суть операции в объединении  $n$  частичных решений в одно с исключением повторяющихся элементов. Например:

$$\{1, 2\} \mathbf{S} \{3, 4, 5\} \mathbf{S} \{1, 3, 7\} = [\{1, 2\}, \{3, 4, 5\}, \{7\}].$$

Здесь  $\mathbf{S}$  – знак суперпозиции. Задача - выбрать решение с наименьшим числом подмножеств, что позволит найти раскраску с наименьшим числом цветов.

Для графа  $G$  (см. рисунок 2) на основе дерева частичных решений выполним следующие операции суперпозиции.

1) Суперпозиция:

$$\{1, 3\} \mathbf{S} \{2, 4, 6\} \mathbf{S} \{4, 5, 6\} - 2 \text{ общих элемента;}$$

$$\{1, 3\} \mathbf{S} \{2, 4, 6\} \mathbf{S} \{5, 6, 7\} - 1 \text{ общий элемент.}$$

Выбираем решение с одним общим элементом. Получим первую раскраску с 3 цветами  $A, B$  и  $C$ :  $\{1, 3\} - A$ ;  $\{2, 4, 6\} - B$ ,  $\{5, 7\} - C$ .

2) Суперпозиция:

$$\{1, 6\} \mathbf{S} \{2, 3, 4\} \mathbf{S} \{4, 5, 6\} - 2 \text{ общих элемента;}$$

$$\{1, 6\} \mathbf{S} \{2, 4, 6\} \mathbf{S} \{5, 6, 7\} - 1 \text{ общий элемент.}$$

Выбираем решение с одним общим элементом. Получим вторую раскраску также с 3 цветами  $D, E$  и  $F$ :  $\{1, 6\} - D$ ,  $\{2, 3, 4\} - E$ ,  $\{5, 7\} - F$ .

3) Суперпозиция:

$$\{1, 7\} \mathbf{S} \{2, 3, 4\} \mathbf{S} \{5, 6, 7\} - 1 \text{ общий элемент;}$$

$$\{1, 7\} \mathbf{S} \{2, 3, 4\} \mathbf{S} \{4, 5, 6\} - 1 \text{ общий элемент.}$$

Выберем любое решение с 1 общим элементом. Получим третью раскраску, также с 3 цветами  $H, I, K$ :  $\{1, 7\} - H$ ,  $\{2, 3, 4\} - I$ ,  $\{5, 6\} - K$ .

На рисунке 4 показаны три возможных раскраски графа  $G$  (см. рисунок 2). Соответственно хроматическое число данного графа равно трем.

Приведем эвристическое правило.

*При наличии нескольких возможных альтернатив суперпозицию в квантовом алгоритме выполнять для подмножеств, имеющих наименьшее число совпадающих элементов.*

Для решения оптимизационных задач в последнее время применяются жадные алгоритмы, которые являются различными модификациями алгоритмов динамического программирования, но жадные алгоритмы проще и быстрее [6]. Основой жадного алгоритма является локально-оптимальный выбор на каждом шаге, с прогнозом, что окончательное решение будет оптимальным. Очевидно, что в общем случае жадный алгоритм может привести в локальный минимум, который далек от оптимального. Хотя для многих задач эти алгоритмы дают возможность получать оптимум искомой целевой функции [4, 6].

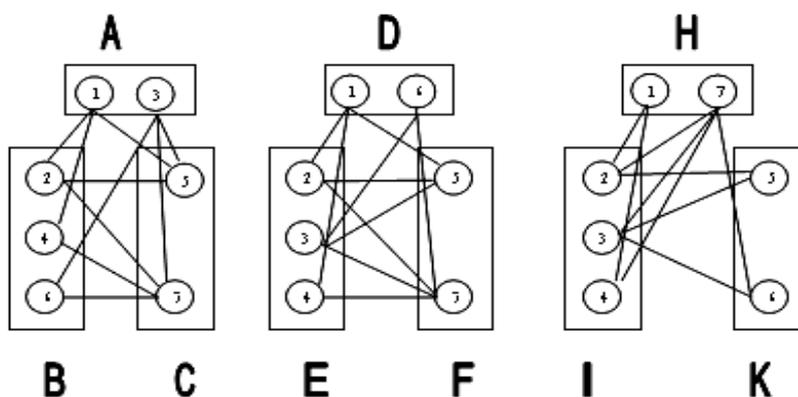


Рисунок 4 - Три возможных раскраски

Приведем ряд эвристик жадного выбора:

- Э1 – жадный выбор на первом и последующих шагах не должен закрывать путь к оптимальному решению;
- Э2 – для упрощения жадного выбора и повышения скорости подзадача, вытекающая после жадного выбора на первом шаге, должна быть аналогична исходной;
- Э3 – задачи, решенные на основе жадных алгоритмов, должны обладать свойством оптимальности для подзадач.

## 7 Архитектуры поиска

На рисунке 5. приведены схемы взаимодействия квантовых и генетических алгоритмов. Очевидно, что данные схемы можно взять как строительные блоки и наращивать иерархически. При этом возможно построить схему последовательного или параллельного совместного поиска любой сложности. Здесь МА – муравьиный, КА – квантовый, ГА – генетический, ПА – пчелиный, РА – роя частиц алгоритмы. На правой схеме (рисунок 5) возможны различные случаи взаимодействия. Например, сначала выполняется генетический алгоритм, а затем квантовый, муравьиный, пчелиный и роя частиц.

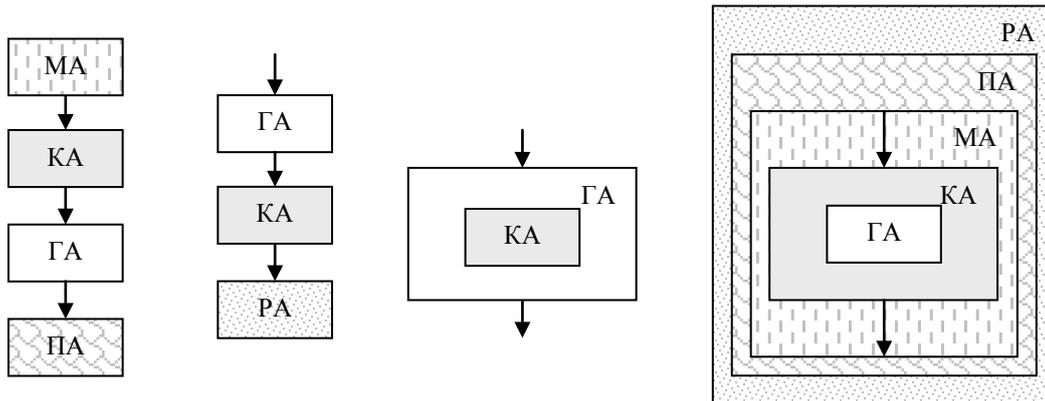


Рисунок 5 - Схемы взаимодействия генетических и квантовых алгоритмов

Отметим также, что квантовый поиск может ускорить классический случайный алгоритм, создавая суперпозиции частотных решений, увеличивая пространство поиска искомого решения.

Предлагается модифицированная схема совместного поиска, состоящая из трех основных блоков (рисунок 6). Первый блок назовем препроцессором. Здесь выполняется создание одной или некоторого множества начальных популяций. Второй блок состоит из 1, 2, ...,  $n$  уровней. Каждый уровень состоит из четырех этапов: выбор представления решения; разработка операторов случайных, направленных и комбинированных изменений; определение законов выживания решения; рекомбинация. Третий блок назовем постпроцессором. Здесь реализуются принципы эволюционной адаптации к внешней среде (лицу, принимающему решение) и самоорганизации. Отметим, что строительные блоки совместного квантового и генетического поиска (рисунок 5) могут эффективно работать в составе процессора и постпроцессора. Преимущество такой архитектуры совместного поиска состоит в том, что в ней все уровни связаны с уровнем внешней среды и могут общаться между собой.

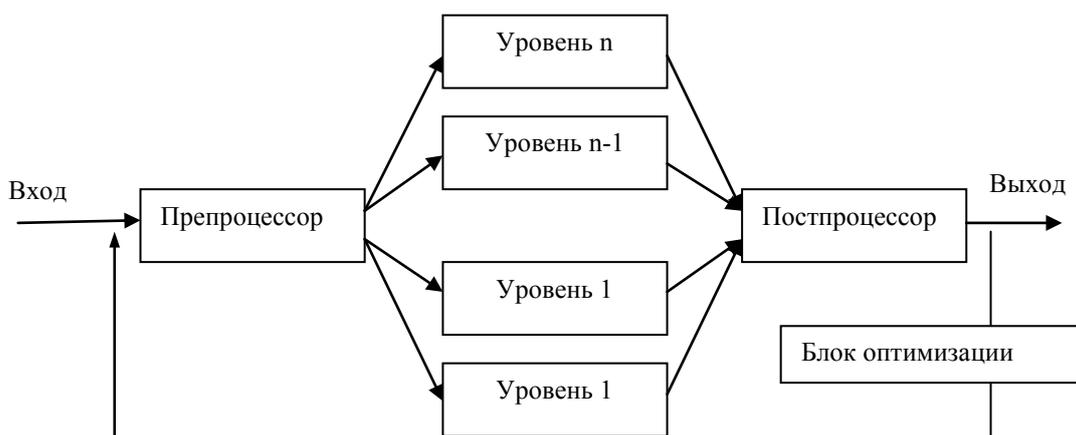


Рисунок 6 - Горизонтальная архитектура совместного поиска

Для управления и реализации процессом совместного поиска используются следующие принципы [6]:

- Принцип синергетики (целостности). В квантовых и генетических алгоритмах значение целевой функции альтернативного решения не сводится к сумме целевых функций частных решений.

- Принцип чувствительности к начальным условиям. Результат работы квантовых и генетических алгоритмов зависит от представления входных данных исследуемой модели.
- Принцип дополнительности. При решении оптимизационных задач возникает необходимость использования различных не совместимых и взаимодополняющих моделей эволюции и исходных объектов.
- Принцип неточности. При росте сложности анализируемой задачи уменьшается возможность построения точной модели.
- Принцип управления неопределенностью. Необходимо вводить различные виды неопределенности в квантовые и генетические алгоритмы.
- Принцип соответствия. Язык описания исходной задачи должен соответствовать наличию имеющейся о ней информации.
- Принцип «007». Используй только те входные данные, которые необходимы для решения задачи.
- Принцип единства и противоположности порядка и хаоса. «Хаос не только разрушителен, но и конструктивен», т.е. в хаосе области допустимых решений обязательно содержится порядок, определяющий искомое решение.
- Принцип иерархичности. Квантовые и генетические алгоритмы могут подстраиваться сверху вниз и снизу вверх.
- Принцип «Бритвы Оккама». Нежелательно увеличивать сложность архитектуры поиска без необходимости.
- Принцип гомеостаза. Квантовые и генетические алгоритмы конструируются таким образом, чтобы любое полученное альтернативное решение не выходило из области допустимых.

## Заключение

Отметим, что можно организовать различное количество связей внутри схемы совместного поиска между блоками по принципу полного графа, по принципу звезды и т.д. Такие схемы в случае наличия большого количества вычислительных ресурсов могут быть доведены до  $N$  блоков. Причем  $N-1$  блоков могут параллельно осуществлять эволюционную адаптацию и через блоки миграции обмениваться лучшими представителями решений. Последний блок, собирая лучшие решения, может окончить результат работы или продолжить оптимизацию. Такие стратегии решения задач позволяют учитывать влияние внешней среды и знания о решаемых задачах и в отличие от существующих методов позволяют во многих случаях выходить из локальных оптимумов.

В настоящее время одной из основных проблем является параллельное выполнение биоинспирированных методов. Существует пять основных методов распараллеливания [3-6]:

- производить параллельно заданное количество запусков алгоритма;
- выполнять сначала один запуск алгоритма. Далее на этапе вычисления целевой функции эта задача распределяется по вспомогательным удаленным компьютерам. В литературе этот метод известен как Хозяин-Раб (Master-Slave);
- исполнить несколько запусков различных алгоритмов (или их частей) с различными операторами селекции кроссинговера, мутации, инверсии и др. с миграцией альтернативных решений (AP), т.е. особей из одного алгоритма в другой;
- выполняется один запуск алгоритма, в котором этап вычисления ЦФ анализируется в несколько потоков на одной ЭВМ. Часто здесь могут анализироваться вопросы размножения, инициализации, селекции и др.;

- выполняется один запуск алгоритма с процедурой селекции, предполагающий, что особи (AP) записаны в параллельный массив на векторном или многоядерном компьютере. Такие модели называются мелкозернистыми или пространственно вложенными;
- выполняется раздельные запуски алгоритма или его частей, между которыми в заданное или случайно выбранное время производится обмен выбранным AP. В [3] это называется «распространением добра» или островной моделью;
- различные комбинации первых шести методов с блоками адаптации и экспертной системы.

Вышеописанные алгоритмы были разработаны и реализованы в среде разработки Code Gear C++ Builder, Microsoft Visual Studio для решения следующих задач: разбиение графа; раскраска графа; задача коммивояжера; маршрутизация автотранспорта; планирование СБИС; задача о назначениях. Экспериментальные исследования показали эффективность алгоритмов роевого интеллекта по сравнению с генетическими и эволюционными алгоритмами. Для решения задачи были реализованы следующие алгоритмы: итерационный, эволюционный, генетический, муравьиный, пчелиный.

Отметим, что автор специально не рассматривал место и положение нейрокомпьютерных систем в данной области, в связи с тем, что они подробно изложены в существующей литературе.

В заключение отметим, что наряду с роевыми алгоритмами, перспективными представляются методы моделирования «принципов стаи». Совместное выполнение роевых алгоритмов и алгоритмов стаи, по мнению автора, позволит повысить качество и эффективность полученных решений.

## Благодарности

Автор благодарен членам своей научно-педагогической школы в области эволюционного моделирования и искусственного интеллекта профессорам В.В. Курейчику, Б.К. Лебедеву, Е.В. Нужнову, В.И. Родзину; доцентам Л.А. Гладкову, Ю.А. Кравченко; кандидатам наук В.В. Гудилову, А.А. Кажарову за совместные научные исследования, позволившие поднять рейтинг нашей школы в России и за рубежом, а также докторантам, аспирантам, магистрантам и студентам за разработку новых программно – аппаратных средств и проведение большого количества эффективных экспериментальных исследований. С особой теплотой хочу поблагодарить свою супругу доктора педагогических наук, профессора Веронику Игоревну Писаренко за поддержку всех моих начинаний, а также за лингвистическую помощь в исследованиях.

## Список источников

- [1] *Люггер, Дж.Ф.* Искусственный интеллект: стратегии и методы решения сложных проблем / Дж.Ф. Люггер. – М.: Издательский дом «Вильямс», 2003.- 864 с.
- [2] *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: University of Michigan Press, 1975. - 211 p.
- [3] *Goldberg, D.E.* Genetic Algorithms in Search, Optimization and Machine Learning / D.E. Goldberg. - Boston: Addison-Wesley Longman Publishing Co., Inc., 1989. – 412 p.
- [4] *Гладков, Л.А.* Биоинспирированные методы в оптимизации / Л.А. Гладков, В.В. Курейчик, В.М. Курейчик, П.В. Сороколетов. - М.: ФИЗМАЛИТ, 2009. – 384 с.
- [5] *Курейчик, В.М.* Генетические алгоритмы и их применение / В.М. Курейчик. – Таганрог: Изд-во ТРТУ, 2002.- 244 с.
- [6] *Гладков, Л.А.* Генетические алгоритмы / Л.А. Гладков, В.В. Курейчик, В.М. Курейчик: под ред. В.М. Курейчика. 2-ое изд. исправл. и доп. – М.: .: ФИЗМАТЛИТ, 2010.- 366 с.

- [7] **Grover, L.K.** A Fast Quantum Mechanical Algorithm for Data-base Search / L.K. Grover // Proc. 28th Ann. ACM Press, New York, 1996. - P. 212-219.
  - [8] **Grover, L.K.** Synthesis of Quantum Superpositions by Quantum Computation / L.K. Grover // Physical Rev. Letters. 2000. Vol 85. No.6. - P. 1334-1337.
  - [9] **Williams, C.P.** Quantum Search Algorithms in Sciences and Engineering / C.P. Williams // Computing in sciences and engineering. 2001. March April. - P. 44-51.
  - [10] **Курейчик, В.М.** Новый подход к решению графовых задач на основе квантовых алгоритмов / В.М. Курейчик // Перспективные информационные технологии и интеллектуальные структуры. 2004. №2(18).
  - [11] **Курейчик, В.М.** Совместные методы квантового и бионического поиска / В.М. Курейчик // Труды конференций IEEE AIS'04, CAD-2004. - М.: Физматлит, 2004. - С. 12-19.
  - [12] **Курейчик, В.М.** Новый подход к раскраске и определению клик графа на основе квантовых алгоритмов / В.М. Курейчик // Известия ТРТУ. 2004. №3. - С. 29-34.
  - [13] *New Ideas in Optimization* / Eds.: D. Corne, M. Dorigo, F. Glover/ - N.-Y.: McGraw-Hill. 1999.
  - [14] **Дориго, М.** Web-сайт Марко Дориго по оптимизации с помощью колонии муравьев - <http://iridia.ulb.ac.be/dorigo/ACO/ACO.html> (актуально 20.10.2013)
  - [15] **Курейчик, В.В.** Эволюционная оптимизация на основе алгоритма колонии пчёл / В.В. Курейчик, Е.Е. Полупанова // Известия ЮФУ. Технические науки. 2009. №12. – С.41-46.
  - [16] **Кажаров, А.А.** Применение пчелиного алгоритма для раскраски графов / А.А. Кажаров, В.М. Курейчик // Известия ЮФУ. Технические науки. 2010. №12. – С.30-37.
  - [17] **Mendes, R.** Watch thy neighbor or how the swarm can learn from its environment / R. Mendes, J. Kennedy, J. Neves // Proc. of Swarm Intelligence Symposium 2003. – IEEE, 2003. - P. 88-94.
  - [18] **Курейчик, В.В.** Концепция эволюционных вычислений, инспирированных природными системами / В.В. Курейчик, В.М. Курейчик, С.И. Родзин // Известия ЮФУ. Технические науки. 2009. №4. – С.16-25.
- 

## PERSPECTIVE INFORMATION TECHNOLOGIES BASED ON THE METHODS INSPIRED BY NATURAL SYSTEMS

**V.M. Kureychik**

*Southern Federal University, Taganrog, Russia*

*kur@tgn.sfedu.ru*

### Abstract

The new technology of solving the optimization and combinatorial logic problems of an artificial intellect, hybrid and parallel, on graph models based on integrated evolutionary, hive, quantum and genetic algorithms is considered. It allows to receive sets of local and optimum solutions and to build heuristic algorithms with polynomial growth rate of amount of operations, depending on volume of entrance data and, partially, to solve a problem of premature convergence.

**Key words:** *evolutionary, quantum, plenty, genetic algorithms, bionical search, graph models.*

### References

- [1] **Lugger J.F.** *Iskusstvennyj intellekt: strategii i metody resheniya slozhnykh problem* [Artificial Intelligence: Strategies and methods for solving complex problems] Moscow: «Vil'yams» publ., 2003, 864 p. (In Russian)
- [2] **Holland J.H.** *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence.* Ann Arbor: University of Michigan Press, 1975, 211 p.
- [3] **Goldberg D.E.** *Genetic Algorithms in Search, Optimization and Machine Learning.* Boston: Addison-Wesley Longman Publishing Co., Inc., 1989, 412 p.
- [4] **Gladkov L.A., Kureychik V.V., Kureychik V.M., Sorokoletov P.V.** *Bioinspirirovannye metody v optimizatsii* [Bio-inspired methods in optimization]. Moscow: FIZMATLIT publ., 2009, 384 p. (In Russian)
- [5] **Kureychik V.V.** *Geneticheskie algoritmy i ikh primenenie* [Genetic algorithms and their applications]. Taganrog: TRTU publ., 2002, 244 p. (In Russian)
- [6] **Gladkov L.A., Kureychik V.V., Kureychik V.M.** *Geneticheskie algoritmy* [Genetic algorithms], second edition. Moscow: FIZMATLIT publ., 2010, 366 p. (In Russian)

- [7] **Grover L.K.** A Fast Quantum Mechanical Algorithm for Data-base Search. Proc. 28th Ann. New York: ACM Press, 1996, pp. 212-219.
- [8] **Grover L.K.** Synthesis of Quantum Superpositions by Quantum Computation. Physical Rev. Letters. 2000. Vol 85. No.6, pp. 1334-1337.
- [9] **Williams C.P.** Quantum Search Algorithms in Sciences and Engineering. Computing in sciences and engineering. 2001, March April, pp. 44-51.
- [10] **Kurejchik V.M.** Novyj podkhod k resheniyu grafovyykh zadach na osnove kvantovykh algoritmov [A new approach to solving problems based on graph of quantum algorithms]. Perspektivnye informatsionnye tekhnologii i intellektual'nye struktury [Advanced information technology and intellectual structures], 2004, no. 2(18). (In Russian)
- [11] **Kurejchik V.M.** Sovmestnye metody kvantovogo i bionicheskogo poiska [Participatory methods of quantum and bionic search]. Proc. IEEE AIS'04, CAD-2004. - Moscow: FIZMATLIT, 2004, pp. 12-19. (In Russian)
- [12] **Kurejchik V.M.** Novyj podkhod k raskraske i opredeleniyu klik grafa na osnove kvantovykh algoritmov [A new approach to color and definition of cliques of the graph based on quantum algorithms]. TRTU [TRTU journal], 2004, no. 3, pp. 29-34. (In Russian)
- [13] **Corne D., Dorigo M., Glover F.** eds. New Ideas in Optimization New York: McGraw-Hill publ., 1999.
- [14] **Dorigo M.** Marco Dorigo Web-site on optimization using ant colony <http://iridia.ulb.ac.be/dorigo/ACO/ACO.html> (accessed 20.10.2013)
- [15] **Kurejchik V.V., Polypanova E.E.** Evolyutsionnaya optimizatsiya na osnove algoritma kolonii pchyol [Evolutionary optimization algorithm based on bee colony]. Izvestiya UFY. Tekhnicheskie nauki [UFY journal, technical sciences], 2009, no. 12, pp. 41-46. (In Russian)
- [16] **Kazharov A.A., Kurejchik V.M.** Primenenie pchelino algoritma dlya raskraski grafov [Application of bee algorithm for graph coloring]. Izvestiya UFY. Tekhnicheskie nauki [UFY journal, technical sciences], 2010, no. 12, pp. 30-37. (In Russian)
- [17] **Mendes R., Kennedy J., Neves J.** Watch thy neighbor or how the swarm can learn from its environment. Proc. of Swarm Intelligence Symposium 2003. – IEEE, 2003, pp. 88-94.
- [18] **Kurejchik V.V., Kurejchik V.M., Rodzin S.I.** Kontseptsiya ehvolyutsionnykh vychislenij, inspirirovannykh prirodnyimi sistemami [The concept of evolutionary computation inspired by natural systems]. Izvestiya UFY. Tekhnicheskie nauki [UFY journal, technical sciences], 2009, no. 4, pp. 16-25. (In Russian)

### Сведения об авторе



**Курейчик Виктор Михайлович**, 1945 г. рождения. Окончил Таганрогский радиотехнический институт в 1967 г., к.т.н. (1971), д.т.н. (1978), профессор (1982). Заведующий кафедрой дискретной математики и методов оптимизации факультета автоматике и вычислительной техники Южного федерального университета. Заслуженный деятель науки РФ (1995). В списке научных трудов около 400 работ, включая 15 монографий в области интеллектуальных САПР, эволюционного моделирования и искусственного интеллекта.

**Viktor Mikhailovich Kurejchik**, (b. 1945). Has graduated from the Taganrog Radio Engineering Institute in 1967, Cand. of Eng. Sc., (1971), Dr. of Eng. Sc. (1978), Professor (1982). The Honored Worker of Science of the Russian Federation (1995). Chairman of Discrete Mathematics And Optimization Methods Department of College of Automation and Computer Science of Southern Federal University. He has written more than 400 works, including 15 monographies in the field of intellectual CAD, evolutionary modeling and an artificial intellect.