

Статью рекомендовал к опубликованию д.т.н., профессор Н.А. Витиска.

Гладков Леонид Анатольевич – Технологический институт федерального государственного автономного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге; e-mail: leo@tsure.ru; тел.: 88634371625; кафедра систем автоматизированного проектирования; доцент.

Gladkov Leonid Anatol'evich – Taganrog Institute of Technology – Federal State-Owned Autonomy Educational Establishment of Higher Vocational Education “Southern Federal University”; GSP 17A, 44, Nekrasovsky, Taganrog, 347928, Russia; phone: +78634371625; the department of computer aided design associate professor.

УДК 681.325

В.Б. Лебедев

МОДЕЛИ АДАПТИВНОГО ПОВЕДЕНИЯ КОЛОНИИ ПЧЕЛ ДЛЯ РЕШЕНИЯ ЗАДАЧ НА ГРАФАХ*

В работе рассматривается модифицированная парадигма пчелиной колонии для решения комбинаторных задач на графах: выделения в графе независимого подмножества вершин, нахождения максимального паросочетания в графе, раскраски графа, выделения клик в графе. На основе анализа поведенческой модели самоорганизации колонии пчёл, разработаны методы и механизмы формирования соответствующих представлений решений рассматриваемых комбинаторных задач на графах. Рассмотрены методы формирования пространства поиска. Позиция в пространстве поиска представляется в виде упорядоченного списка. Список фактически является кодом решения. Ключевой операцией пчелиного алгоритма является исследование перспективных позиций и их окрестностей в пространстве поиска. В работе предлагается метод формирования окрестностей решений с регулируемой степенью подобия и близости между ними. Предлагаются три подхода к определению числа агентов фуражиров, направляемых в окрестности каждой базовой позиции. По сравнению с существующими алгоритмами достигнуто улучшение результатов.

Роевой интеллект; пчелиная колония; адаптивное поведение; самоорганизация; граф; оптимизация.

V.B. Lebedev

MODELS OF A BEE COLONY ADAPTIVE BEHAVIOUR FOR THE DECISION OF PROBLEMS ON GRAPHS

In work the modified paradigm of a bee colony for the decision of combinatorial problems on graphs is considered. On the basis of the analysis of behavioural model of self-organizing of a colony of bees, methods and mechanisms of formation of corresponding representations of decisions of considered combinatorial problems on graphs are developed. Methods of formation of space of search are considered. The position in search space is represented in the form of the ordered list. The list actually is a decision code. Key operation of bee algorithm is research of perspective positions and their vicinities in search space. In work the method of formation of vicinities of decisions with adjustable degree of similarity and affinity between them is offered. Three approaches to definition of foraging bees number, directed to vicinities of each base position are offered. In comparison with existing algorithms improvement of results is reached.

Swarm intelligence; ant colony; adaptive behaviour; self-organizing, graph; optimization.

* Работа выполнена при частичной поддержке РФФИ (проект № 11-01-00975).

Введение. Среди набора комбинаторно-логических задач на графах важное место занимает задачи определения паросочетаний, выделения в графе независимого подмножества вершин, выделения в графе клик, раскраски графа. Алгоритмы решения данных задачи применяются при проектировании инженерных сетей, коммуникаций, построения систем поддержки принятия решений в неопределенных условиях, проектировании СБИС и т.п. Задачи такого типа относятся к переборным задачам с экспоненциальной временной сложностью. В этой связи разрабатывают различные эвристики для построения алгоритмов с полиномиальной временной сложностью. Существуют алгоритмы решения таких задач, основанные на использовании потоков в сетях [1, 2], имитационного моделирования [2], генетического поиска [3] и других эвристиках, которые обеспечивают приемлемые результаты при решении задач малой и средней сложности. Часто эти процедура используется в итерационных структурах. Это предъявляет повышенные требования к качеству и времени решения рассматриваемых задач. Побудительным мотивом исследований и разработок новых эффективных алгоритмов являются возникшие потребности в решении задач большой и очень большой размерности. Анализ литературы показывает, что наиболее успешными в этих условиях являются математические методы, в которых заложены принципы природных механизмов принятия решений [4–7]. К таким методам можно отнести, прежде всего, методы моделирования отжига [7], метод эволюционного моделирования [8, 9], генетические алгоритмы [10], эволюционной адаптации [11], алгоритмы роевого интеллекта [12–14]. Одна из последних разработок в области роевого интеллекта – алгоритм пчел довольно успешно используется для нахождения глобальных экстремумов сложных многомерных функций [15, 16].

В работе излагаются методика решения родственных задач выделения в графе независимого подмножества вершин, нахождения максимального паросочетания в графе, раскраски графа, выделения клик в графе, основанная на моделировании адаптивного поведения пчелиной колонии.

Постановка комбинаторных задач на графах. Паросочетанием графа $G=(X,U)$ называется подмножество таких рёбер $U^* \subset U$, что любые два ребра $u_k, u_l \in U^*$ не имеют общих вершин, т.е. не смежны. Паросочетание максимальной мощности определяется как паросочетание, включающее максимальное число рёбер [1], $|U^*| = \max$.

Построим граф $G_d=(U,V)$ – двойственный для графа G . Вершины графа G_d – соответствуют рёбрам графа G . Пара вершин (u_i, u_j) в графе G_d связаны ребром v_k в том и только в том случае, если в графе G соответствующая пара рёбер (u_i, u_j) смежны, т.е. инциденты одной вершине.

Множество $X_0 \subset X$ вершин графа $G=(X,U)$ называется внутренне устойчивым, если любые две вершины $x_i \in X_0$ и $x_j \in X_0$ не являются смежными. Максимальное число вершин во внутренне устойчивом множестве графа G называется числом внутренней устойчивости и обозначается как $\alpha(G)$. Иногда число внутренней устойчивости называют также числом независимости графа G .

Таким образом, паросочетанию в графе G соответствует внутренне устойчивое подмножество двойственного графа G_d . Максимальному по мощности паросочетанию в графе G соответствует предельное внутренне – устойчивое подмножество (содержащее наибольшее число вершин) двойственного графа G_d .

Раскраской графа называется такое приписывание цветов его вершинам, что никакие две смежные вершины не получают одинакового цвета. Минимальное число цветов, в которое можно раскрасить граф G , называется хроматическим числом и обозначается $\chi(G)$. Если в графе G выделить s непересекающихся друг с другом внутренне – устойчивых подмножеств вершин, то граф можно раскрасить в s цветов. Другими словами, задача раскраски графа сводится к задаче формирования в графе G непересекающихся внутренне – устойчивых подмножеств вершин.

Клик графа G называется максимальное по включению множество X_0 вершин графа, любые две из которых являются смежными. Пусть $G_n=(X, U_n)$ полный граф, построенный на множестве вершин X . Граф $G_k=(X, U_k)$ является дополнением графа $G=(X, U)$, если $U_k = U_n \setminus U$, т.е. $U_n=U_k \cup U$. Нетрудно видеть, что при переходе от графа G к его дополнению G_k каждая клика в G переходит в независимое множество в G_k . Отсюда следует, что задача выделения клики в графе G сводится к задаче выделения независимого множества вершин в графе G_k , являющегося дополнением графа G .

Таким образом, в основе процедур построения максимального паросочетания, раскраски графа, выделения в графе клик лежит одна общая процедура формирования в графе $G=(X, U)$ внутренне-устойчивого множества вершин $X_1 \subset X$.

Пусть дан граф $G=(X, U)$, где X – множество вершин, $|X| = n$, U – множество ребер. Сформулируем задачу формирования в графе $G(X, U)$ внутренне-устойчивого множества вершин $X_1 \subset X$ как задачу разбиения.

Необходимо разбить множество X на два непустых и непересекающихся подмножества X_1 и X_2 , таких, что любые две вершины $x_i \in X_1$ и $x_j \in X_1$ не являются смежными, $X_1 \cup X_2 = X$, $X_1 \cap X_2 = \emptyset$. Пусть $|X_1| = n_1$, $|X_2| = n_2$, $n_1 + n_2 = n$. Критерий оптимизации – число вершин $F = n_1$ в подмножестве X_1 . Цель оптимизации максимизация критерия F .

После формирования в графе $G(X, U)$ внутренне-устойчивого множества вершин $X_1 \subset X$ для построения паросочетания или выделения в графе клики осуществляется переход от графа G к исходному графу G_n .

При этом, при построении паросочетания граф G рассматривается как двойственный исходному графу G_n , а при выделении клики граф G рассматривается как дополнительный исходному графу G_n .

При решении задачи раскраски графа подмножество X_1 окрашивается в один цвет и исключается из X . Далее выполняются аналогичные действия, пока не будут раскрашены все вершины.

Представления решений в алгоритмах на основе пчелиной колонии. Основу поведения пчелиного роя составляет самоорганизация, обеспечивающая достижение общих целей роя на основе низкоуровневого взаимодействия. Основная идея парадигмы пчелиной колонии заключается в использовании двухуровневой стратегии поиска. На первом уровне с помощью пчел разведчиков формируется множество перспективных областей (источников), на втором уровне с помощью пчел фуражиров осуществляется исследование окрестностей данных областей (источников). Цель пчелиной колонии найти источник, содержащий максимальное количество нектара.

В алгоритмах рассматриваемых задач каждое решение представляется в виде точки (позиции) в пространстве поиска. Найденное количество нектара представляет собой значение целевой функции в этой точке. Решение представляет комбинацию уникальных компонент (вершин и ребер графа поиска решений), выбираемых, как правило, из конечного набора конкурирующих между собой компонент. Значения целевой функции F определяется комбинациями, выбранными агентами. Целью является поиск оптимальной комбинации компонент.

В эвристических алгоритмах роевого интеллекта процесс поиска решений заключается в последовательном перемещении агентов в пространстве поиска. Процесс поиска решений итерационный. На каждой итерации агенты перемещаются в новые позиции.

Разработка поведенческой модели самоорганизации колонии пчёл, заключается в разработке методов и механизмов:

- ♦ формирования пространства поиска;

- ◆ формирования количественного состава роя агентов разведчиков и роя агентов фуражиров;
- ◆ поиска агентами разведчиками перспективных позиций;
- ◆ выбора базовых позиций среди перспективных для исследования их окрестностей,
- ◆ выбора агентами фуражирами базовых позиций;
- ◆ формирования окрестностей базовых позиций;
- ◆ выбора агентами фуражирами позиций в окрестностях базовых позиций;
- ◆ общей структуры оптимизационного процесса.

Первая задача при разработке алгоритма на основе парадигмы пчелиной колонии заключается в формировании пространства поиска. Позиция a_s пространства поиска представляется в виде упорядоченного списка $E_s = \{e_{si} \mid i=1, 2, \dots, n\}$ номеров вершин графа G , где n – количество вершин графа. Список E_s фактически является кодом решения. Формирование соответствующего списку E_s решения – внутренне-устойчивого множества X_s осуществляется пошагово путем последовательного просмотра элементов списка E_s , начиная с первого. На каждом шаге i просмотра есть список вершин, уже включенных в формируемое множество $X_s(i)$, $X_s(1) = \emptyset$. Рассматривается очередная вершина e_{si} . Если среди вершин множества $X_s(i)$ нет ни одной вершины смежной вершине e_{si} , то e_{si} включается в $X_s(i)$. В результате выполнения этой последовательной процедуры формируются внутренне-устойчивое множества X_s и список оставшихся вершин E_{os} , сформированный путем удаления из E_s вершин множества X_s . Назовем упорядоченный список E_{os} остатком.

Таким образом, позиции a_s пространства поиска, представляемой в виде упорядоченного списка E_s , соответствует внутренне-устойчивое множество X_s и остаток E_{os} . Оценкой позиции a_s является оценка множества X_s . Будем называть тройку параметров E_s, X_s, E_{os} параметрами позиции a_s .

Ключевой операцией пчелиного алгоритма является исследование перспективных позиций и их окрестностей в пространстве поиска. Остановимся на понятии окрестности. Смысл, изначально вкладываемый в понятие окрестности, заключается в том, что решения, лежащие в окрестности некоторой позиции, обладают высокой степенью подобия и, как правило, незначительно отличаются друг от друга. Однако многие исследователи, на взгляд автора, допускают ошибку, связанную с тем, что в окрестности включают решения, отличающиеся близостью их кодов. При этом решения, у которых коды близки, оказывались весьма различными при декодировании. По этой причине часто неоправданным является использование для определения степени близости между решениями расстояния Хэмминга между кодами решений

В работе предлагается метод формирования окрестностей решений с регулируемой степенью подобия и близости между ними.

Рассмотрим принципы формирования позиции a_s , расположенной в окрестности базовой позиции a_s^0 . Случайным образом из сформированного множества X_s удаляется λ элементов. Образуется множество $X_s(\lambda)$. Затем с помощью вышеописанной процедуры формирования множества X_{I_s} предпринимается попытка дополнить множество $X_s(\lambda)$ вершинами из остатка E_{os} . В результате этих действий будет сформировано внутренне-устойчивое множество $X_s(\lambda)$ очень близкое по своему содержанию внутренне-устойчивому множеству X_s . Степень различия регулируется управляющим параметром λ – пороговое значение размера окрестности. Удаленные из X_s вершины заносятся в конец остатка E_{os} . Отметим, что, если после удаления вершин из подмножества X_s оно не было дополнено, то подмножество $X_{I_s}(\lambda)$ исключается из рассмотрения.

Организация поисковых процедур на основе моделировании адаптивного поведения пчелиной колонии. Основными параметрами метода пчелиной колонии являются: количество агентов n_b , максимальное количество итераций L , начальное количество агентов-разведчиков n_r , ограничение максимального количества агентов-разведчиков, пороговое значение размера окрестности λ и др.

В начале процесса поиска все агенты расположены в улье, т.е. вне пространства поиска.

На первой итерации ($l=1$) агенты-разведчики в количестве n_r случайным образом размещаются в пространстве поиска. Эта операция заключается в генерации случайным образом множества отличающихся друг от друга списков $E=\{E_s|s=1,2,\dots,n_r\}$, которым соответствует множество позиций $A = \{a_s|s=1,2,\dots,n_r\}$. Для каждого списка E_s формируется внутренне-устойчивое множество X_s с остатком E_{os} , и вычисляется значение целевой функции F_s . Выбирается n_b базовых (лучших) решений $X^{\bar{o}}=\{X_s\}$, у которых значения целевой функции не меньше, чем значение целевой функции у любого не выбранного решения. Формируется множество базовых (лучших) позиций $A^{\bar{o}}=\{a^{\bar{o}}_s|s=1,2,\dots,n_b\}$, соответствующих множеству базовых (лучших) решений $X^{\bar{o}}$.

Предлагаются три подхода к определению числа агентов фуражиров, направляемых в окрестности каждой базовой позиции. При первом подходе агенты фуражиры распределяются по базовым позициям равномерно. При втором подходе агенты фуражиры распределяются по базовым позициям пропорционально значению целевой функции позиции. При третьем подходе реализуется вероятностный выбор. Вероятность $P(a^{\bar{o}}_s)$ выбора агентом фуражиром базовой позиции $a^{\bar{o}}_s \in A^{\bar{o}}$ пропорциональна значению целевой функции $F^{\bar{o}}_s$ в этой позиции и определяется как

$$P(a^{\bar{o}}_s) = F^{\bar{o}}_s / \sum_s (F^{\bar{o}}_s).$$

При первом и втором подходе число решений в окрестностях рассчитывается, при третьем подходе – определяется случайно.

После выбора агентом фуражиром b_z базовой позиции $a^{\bar{o}}_s \in A^{\bar{o}}$, реализуется вероятностный выбор позиции a_z , расположенной в окрестности базовой позиции $a^{\bar{o}}_s$. Вероятностный выбор позиции a_z и формирование соответствующего ей решения осуществляется в соответствии с процедурой, изложенной выше. При этом предварительно случайным образом определяется число удаляемых вершин λ_z , лежащее в границах $1 \leq \lambda_z \leq \lambda$.

Обозначим множество позиций, выбранных агентами фуражирами в окрестности позиции $a^{\bar{o}}_s$ как $O^{\bar{o}}_s$. Назовем множество позиций $O^{\bar{o}}_s \cup a^{\bar{o}}_s$ областью $D^{\bar{o}}_s$. В каждой области $D^{\bar{o}}_s$ выбирается лучшая позиция a^*_s с лучшей оценкой F^*_s . Назовем F^*_s оценкой области $D^{\bar{o}}_s$. Среди F^*_s выбирается лучшая оценка F^* и соответствующее ей решение, найденное на данной итерации совместно роём разведчиков и роём фуражиров. Лучшее решение с оценкой F^* сохраняется, а затем происходит переход к следующей итерации.

Отметим, что в рассматриваемой парадигме пчелиной колонии не важно, знать, каким агентом выбрана позиция в пространстве поиска. Важно знать число агентов-разведчиков и число агентов-фуражиров, а также, какие именно позиции выбраны агентами-разведчиками и какие – агентами-фуражирами.

На второй и последующих итерациях множество базовых позиций $A^{\bar{o}}(l)$ ($l=2,3,\dots,L$), формируется из двух частей $A^{\bar{o}1}(l)$ и $A^{\bar{o}2}(l)$, $A^{\bar{o}1}(l) \cup A^{\bar{o}2}(l) = A^{\bar{o}}(l)$. В первую часть $A^{\bar{o}1}(l)$ включаются $n_{\bar{o}1}$ лучших позиций a^*_s , найденных агентами в каждой из областей, сформированных на предыдущей итерации. Вторая часть $A^{\bar{o}2}(l)$

формируется пчелами-разведчиками также как и на первой итерации. Отличие заключается в числе n_{r1} агентов-разведчиков, выбирающих случайным способом новые позиции. $n_{r1} < n_r$. В множество $A^{\delta 2}(l)$ включается $n_{\delta 2}$ лучших позиций из n_{r1} новых позиций, найденных агентами-разведчиками на l -ой итерации. $n_{\delta 1} + n_{\delta 2} = n_{\delta}$. Далее выполняются действия, аналогичные действиям, рассмотренным на первой итерации. Рассчитывается число агентов-фуражиров, направляемых в окрестности каждой базовой позиции. Каждым агентом-фуражиром b_z выбирается базовая позиция $a_s(l)$ и позиция $a_z(l)$, расположенная в окрестности этой базовой позиции. В каждой области $D_s^{\delta}(l)$ выбирается лучшая позиция $a_s^*(l)$ с лучшей оценкой решения $F_s^*(l)$. Среди оценок $F_s^*(l)$ выбирается лучшая $F^*(l)$. Если $F^*(l)$ лучше $F^*(l-1)$, то сохраняется решение с этой оценкой, а затем происходит переход к следующей итерации.

Схема работы роевого алгоритма размещения включает следующие шаги:

1. Задаются основные параметры метода пчелиной колонии: L – максимальное количество итераций; n_r – начальное количество агентов-разведчиков; n_{δ} – количество базовых позиций; λ – пороговое значение размера окрестности; n_f – начальное количество агентов-фуражиров; $n_{\delta 1}$ – количество базовых позиций, формируемых из лучших позиций $a_s^*(l)$, найденных роем на l -ой итерации; n_{r1} – количество агентов-разведчиков, выбирающих случайным способом новые позиции; $n_{\delta 2}$ – количество базовых позиций, формируемых из лучших новых позиций, найденных агентами-разведчиками на l -ой итерации.

2. $l=1$ (l – номер итерации).

3. Генерация случайным образом множества отличающихся друг от друга списков $E(l) = \{E_s(l) | s=1,2,\dots,n_r\}$, которым соответствуют множество позиций $A(l) = \{a_s(l) | s=1,2,\dots,n_r\}$.

4. Для каждого списка $E_s(l)$ формируется внутренне-устойчивое множество $X_s(l)$ с остатком $E_{os}(l)$, вычисляется значение целевой функции $F_s(l)$.

5. Формируется множество базовых решений $X^{\delta}(l) \subset X(l)$ с лучшими значениями целевых функций $F_s(l)$ и соответствующее множество базовых позиций $A^{\delta}(l) \subset A(l)$.

$$|A^{\delta}(l)| = |X^{\delta}(l)| = n_{\delta}.$$

6. $z=1$ (z – порядковый номер агента-фуражира).

7. Выбор с вероятностью $P(a_s^{\delta}) = F_s^{\delta} / \sum_s (F_s^{\delta})$ базовой позиции $a_s^{\delta}(l) \in A^{\delta}(l)$.

8. Вероятностный выбор позиции $a_z(l)$, расположенной в окрестности базовой позиции $a_s^{\delta}(l)$, с соответствующим решением $X_z(l)$.

9. Если позиция $a_z(l)$ совпадает с ранее выбранными позициями, то переход к 8, иначе переход к 10.

10. Позиция $a_z(l)$ включается в множество $O_s(l)$.

11. Расчет значения целевой функции $F_z(l)$ решения $X_z(l)$.

12. Если $z < n_f$, то $z = z+1$ и переход к 7, иначе переход к 13.

13. Формирование для каждой базовой позиции $a_s^{\delta}(l)$ области $D_s(l) = O_s(l) \cup a_s^{\delta}(l)$.

14. В каждой области $D_s(l)$ выбирается лучшая позиция $a_s^*(l)$ с лучшим решением $X_s^*(l)$.

15. Среди $X_s^*(l)$ выбирается лучшее решение $X^*(l)$.

16. Если $X^*(l)$ лучше $X^*(l-1)$, то оно сохраняется, иначе $X^*(l) = X^*(l-1)$.

17. Если $l < L$, то $l = l+1$ и переход к 18, иначе переход к 22.

18. В первую часть $X^{\delta 1}(l)$ включаются $n_{\delta 1}$ лучших позиций, среди позиций $x_s^*(l-1)$, найденных агентами в каждой из $D_s(l-1)$ областей, сформированных на предыдущей итерации.

19. Генерация случайным образом множества, отличающихся друг от друга списков $E(l) = \{E_s(l) | s=1,2,\dots,n_{rl}\}$, которым соответствует множество позиций $A(l) = \{a_s(l) | s=1,2,\dots,n_{rl}\}$. $|E(l)|=n_{rl}$.

20. Включение в множество $A^{\delta 2}(l)$ $n_{\delta 2}$ лучших позиций из множества $A(l)$ новых позиций, найденных агентами разведчиками на l -ой итерации $n_{\delta 1} + n_{\delta 2} = n_{\delta}$.

21. Формирование множества базовых позиций $A^{\delta}(l) = A^{\delta 1}(l) \cup A^{\delta 2}(l)$. Переход к б.

22. Конец работы алгоритма. Решение $X^*(l)$ – лучшее решение, найденное ро- ем агентов.

Временная сложность этого алгоритма зависит от времени жизни колонии l (число итераций), количества позиций c и числа агентов m , и определяется как $O(l * c^2 * m)$.

Заключение. В работе рассмотрена модифицированная парадигма пчелиной колонии для решения комбинаторных задач на графах: выделения в графе независимого подмножества вершин, нахождения максимального паросочетания в графе, раскраски графа, выделения клик в графе. На основе анализа поведенческой модели самоорганизации колонии пчёл, разработаны методы и механизмы формирования соответствующих представлений решений рассматриваемых задач. Предложен метод формирования окрестностей решений с регулируемой степенью подоби- я и близости между ними. По сравнению с существующими алгоритмами достигнуто улучшение результатов.

Для преодоления локального барьера, используются подходы, основанные на сочетании различных видов эволюции. В частности, эффективность показала композитная архитектура бионического поиска, заключающаяся в последовательной работе муравьиного и эволюционного алгоритма нахождения максимального паросочетания, рассмотренного в работе [8]. Сравнение с известными алгоритмами показало, что при меньшем времени работы новый алгоритм дает более качественные решения.

Рассматриваемая парадигма имеет универсальный характер и является эффективным способом поиска рациональных решений для задач оптимизации, допускающих интерпретацию в виде упорядоченных списков

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Андерсон Д.* Дискретная математика и комбинаторика. – М.: Вильямс, 2003.
2. *Кормен К., Лейзерсон Ч., Ривест Р.* Алгоритмы, построение и анализ. – М.: МЦМНО, 2000.
3. *Курейчик В.В., Курейчик В.М.* Генетический алгоритм определения паросочетаний графа. Труды 10-ой Международной конференции “Knowledge-dialogue-solution”. – Варна, Болгария, 2003. – С. 246-251.
4. *Engelbrecht A.P.* Fundamentals of Computational Swarm Intelligence. John Wiley & Sons, Chichester, UK, 2005.
5. *Курейчик В.В., Курейчик В.М., Гладков Л.А., Сороколетов П.В.* Бионспирированные методы в оптимизации. – М.: Физмалит, 2009.
6. Методы и алгоритмы принятия решений на основе бионического поиска. Научное издание / Под редакцией *В.М. Курейчика*. – Таганрог: Изд-во ТТИ ЮФУ, 2009.
7. Концепция поиска оптимальных решений при проектировании. Научное издание / Под ред. *Б.К. Лебедева*. – Таганрог: Изд-во ТТИ ЮФУ, 2010.
8. *Лебедев Б.К., Лебедев О.Б.* Эволюционный алгоритм нахождения максимального паросочетания // 3-й Международный НТС “Интегрированные модели и мягкие вычисления в искусственном интеллекте”. – М: Изд-во Физматлит, 2005. – С. 274-280.
9. *Лебедев Б.К., Лебедев О.Б.* Эволюционный алгоритм раскраски графов // Известия ТРТУ. – 2005. – № 3 (47). – С. 202-204.

10. Гладков Л.А., Курейчик В.В., Курейчик В.М., Родзин С.И. Основы теории эволюционных вычислений. – Ростов-на-Дону: Изд-во ЮФУ, 2010
11. Курейчик В.М., Лебедев Б.К., Лебедев О.Б. Поисковая адаптация: теория и практика. – М.: Физматлит, 2006.
12. Курейчик В.М., Кажаров А.А. Использование роевого интеллекта в решении NP-трудных задач // Известия ЮФУ. Технические науки. – 2011. – № 7 (120). – С. 30-36.
13. Курейчик В.В., Запорожец Д.Ю. Роевой алгоритм в задачах оптимизации // Известия ЮФУ. Технические науки. – 2010. – № 7 (108). – С. 28-32.
14. Лебедев О.Б. Решения комбинаторных задач на графах на основе метода муравьиной колонии // Труды конгресса по интеллектуальным системам и информационным технологиям «AIS- IT'09». Научное издание в 4-х томах. Т.1. – М.: Физматлит, 2009. – С. 51-58.
15. Курейчик В.М., Кажаров А.А. Применение пчелиного алгоритма для раскраски графов // Известия ЮФУ. Технические науки. – 2010. – № 12 (113). – С. 30-36.
16. Лебедев Б.К., Лебедев В.Б. Размещение на основе метода пчелиной колонии // Известия ЮФУ. Технические науки. – 2010. – № 12 (113). – С. 12-19.

Статью рекомендовал к опубликованию д.т.н., профессор Ю.О. Чернышев.

Лебедев Владимир Борисович – Технологический институт федерального государственного автономного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге; e-mail: lbk@tsure.ru; 347928, г. Таганрог, пер. Некрасовский, 44, ГСП 17А; тел.: 88634371743; кафедра системного анализа и телекоммуникаций; доцент.

Lebedev Vladimir Borisovich – Taganrog Institute of Technology – Federal State-Owned Autonomy Educational Establishment of Higher Vocational Education “Southern Federal University”; e-mail: lbk@tsure.ru; GSP 17A, 44, Nekrasovsky, Taganrog, 347928, Russia; phone: +78634371743; the department of system analysis and telecommunications; associate professor.

УДК 004.8.023 + 004.81.85

С.И. Родзин

ОБОБЩЕННАЯ МОДЕЛЬ БАЗОВОГО ЦИКЛА ЭВОЛЮЦИОННЫХ ВЫЧИСЛЕНИЙ*

Вычисления, инспирированные природными системами и процессами, являются междисциплинарным направлением и включают три взаимосвязанных области: биоинформатика, вычисления на основе молекулярных, ДНК и квантовых компьютеров, а также эволюционные вычисления. В статье рассматривается базовый цикл эволюционных вычислений. Это математические преобразования, позволяющие трансформировать входной поток информации в выходной по правилам, имитирующим механизм эволюции. Приводится алгоритм организации эволюционных вычислений в базовом цикле и его модификация, а также реализация базового цикла эволюционных вычислений в алгоритме табуцированного поиска.

Эволюционные вычисления; базовый цикл; алгоритм табуцированного поиска.

S.I. Rodzin

GENERALIZED MODEL OF BASIC CYCLE EVOLUTIONARY COMPUTATION

The calculations are inspired by natural systems and processes are interdisciplinary focus and includes three interrelated areas: bioinformatics, molecular computing-based, DNA and quantum computers, as well as evolutionary computation. This article discusses the basic cycle of

* Работа выполнена при поддержке гранта РФФИ 11-07-00094-а.