

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Яценко В.В. Введение в криптографию. – М.: МЦНМО, 2003.
2. Баричев С., Криптография без секретов [Электронный ресурс] – <http://forum.armkb.com>.
3. Василенко О.Н. Теоретико-числовые алгоритмы в криптографии. – М.: МЦНМО, 2003.
4. МакКоннелл Дж. Основы современных алгоритмов. – М.: Техносфера, 2004.
5. Гладков Л.А., Курейчик В.М., Курейчик В.В. Генетические алгоритмы. – Ростов-на-Дону: ООО «Ростиздат», 2004.

УДК 681.31

П.В. Афонин, О.В. Кокшагина

ГИБРИДНЫЕ ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ ДЛЯ ЗАДАЧИ
СОСТАВЛЕНИЯ РАСПИСАНИЯ ПРОЕКТА*

Введение. Интеграция и гибридизация различных методов и компьютерных технологий позволяет решать сложные задачи, которые трудно или нельзя решить на основе каких-либо отдельных методов и технологий. В гибридной системе (ГС), объединяющей несколько методов или подходов, эффективность одного подхода может компенсировать слабость другого. Комбинируя различные подходы, можно обойти недостатки, присущие каждому из них в отдельности.

Классификация гибридных систем по уровню интеграции приведена в [1]. Согласно этой классификации выделяются автономные, трансформационные, слабосвязанные, сильно связанные и полностью интегрированные системы. Идея гибридных генетических алгоритмов (hybrid genetic algorithms) заключается в сочетании генетического алгоритма (ГА) с некоторым другим методом поиска [2].

В статье рассматривается решение задачи составления расписания проекта с помощью гибридных генетических алгоритмов. Предложены гибридные схемы с использованием улучшающих эвристических алгоритмов, гибридная схема формирования начальной популяции, модифицированные операторы скрещивания и мутации для случая кодирования строки-хромосома как последовательности работ в расписании. Основное внимание уделено построению гибридного алгоритма на основе трансформационной схемы. Программная реализация алгоритмов и проведение экспериментальных исследований осуществлялось в программной среде MatLab 7.1.

Постановка задачи. Рассмотрим следующую постановку задачи: проект представляет собой совокупность работ и ресурсов. Ресурс характеризуется нормальным или сверхурочным потреблением в день (в единицах ресурса), стоимостью единицы ресурса при нормальном потреблении и сверхурочном потреблении соответственно. Работа характеризуется длительностью, списком назначенных ресурсов, списком предшествующих и последующих работ, задержками между работами (например, в случае закладки фундамента).

Необходимо составить расписание работ проекта. Под расписанием понимается функция, которая каждому ресурсу l и моменту времени t сопоставляет работу, обслуживаемую ресурсом l в момент времени t , либо указывает, что прибор l в момент времени t ресурс простаивает.

* Работа выполнена при поддержке РФФИ (гранты № 08-07-00337, 08-07-00343).

$$S = \left(90! \cdot \frac{1}{2 \cdot 10^9 \cdot 90 \cdot 3600 \cdot 24 \cdot 365} \right)^6 = (1,574 \cdot 10^{126})^6.$$

В большинстве случаев при планировании проекта приходится искать баланс между его стоимостью и длительностью. Поэтому в качестве целевой функции используется произведение или сумма длительности и стоимости выполнения работ проекта.

Мощность множества допустимых решений для проекта, состоящего из 90 работ и 6 ресурсов, составляет:

По видам задач теории расписания в [3] рассматриваемая постановка задачи сводится к TP7 – минимум максимальных затрат. Она разрешима за полиномиальное время только в том случае, если условие частичного порядка представимо в виде параллельно-последовательного графа. Для рассматриваемой задачи составления расписания это условия не выполняется, поэтому она является NP-трудной задачей дискретной оптимизации и для её решения лучше воспользоваться приближенными методами [4].

Гибридные схемы и операторы генетического алгоритма. При решении задачи составления расписаний в качестве хромосомы выбрана последовательность работ. В этом случае необходимо выполнить требование, чтобы значения всех генов (аллели) были уникальными, т.е. каждая работа должна быть включена в расписание один раз, при этом должны быть выполнены все работы проекта, т.е. каждая работа должна попасть в расписание. При добавлении очередной работы в расписание ресурсы на ее выполнение должны быть свободны, а предшествующие работы (при их наличии) включены в расписание. Соответственно в качестве хромосом могут использоваться не любые случайные последовательности работ, а только заранее допустимые. Данные ограничения обуславливают разработку модифицированного генетического алгоритма, т.е. помимо алгоритма для вычисления функции пригодности, разработаны алгоритмы формирования начальной популяции, операторы кроссинговера, мутации.

Далее рассмотрим алгоритмы и схемы реализации оператора создания начальной популяции, оператора скрещивания и оператора мутации.

Алгоритм оператора создания популяции:

1. Подключить базу данных.
2. Сформировать случайную последовательность работ.
3. Для текущей работы в последовательности проверить предшественников
4. Проверить, включены ли предшественники в расписание. Если включены, или предшественников нет, то включить текущую работу в расписание.
5. Если условие 2 не выполняется, то меняем текущую работу с соседней.
6. Повторяем пункты 2, 3.
7. Аналогичным образом формируется вся последовательность.

Таким образом, получаем допустимое расписание.

Алгоритм оператора мутации:

1. Сгенерировать случайное число k.
2. Выбрать все работы до k в последовательности, которые могут быть переставлены с k работой (т.е. выбрать те работы, которые не являются предшественниками k).
3. Выбрать все работы до k в последовательности, которые могут быть переставлены с k работой (т.е. выбираем те работы, предшественником которых не является k-ая работа).
4. Сгенерировать случайное число sl.

5. В получившейся последовательности возможных работ для перестановки выбрать работу на sl -ой позиции.
6. Если $sl < k$, то проверяем, чтоб работы sl не была предшественником работ, находящихся до k -ой работы.
7. Если $sl > k$, то проверяем, чтоб работы предшественники работы sl не находились между sl и k .
8. Если условия 6 или 7 не выполняются, то заново генерируем число sl . Повторяем пункты 5, 6, 7.
9. Меняем местами k -ую и sl -ую работу.

Схема оператора мутации представлена на рис. 1.

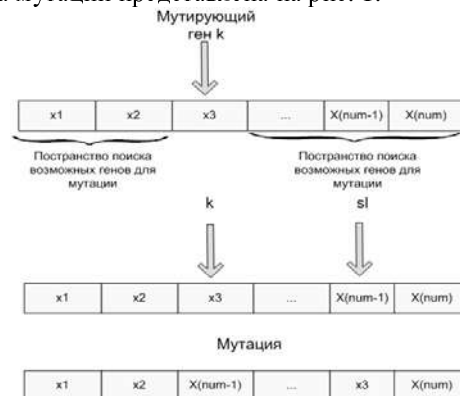


Рис. 1. Схема оператора мутации

Алгоритм оператора скрещивания:

1. Генерация случайных чисел a, b .
2. Выбор родительской пары из популяции a -ая особь и b -ая.
3. Формирование особи потомка:
 - a. Первый ген особи 1го родителя принимается по умолчанию a .
 - b. По 2-ому родителю производится поиск 1-ого гена 1 родителя.
 - c. Выбирается следующий за ним ген (c).
 - d. Выполняется проверка:
 - i. Включен ген c в расписание:
 - Да, поиск элемента в текущем родителе после элемента c .
 - Нет, переход к поиску ii.
 - ii. Включении ли его предшественники:
 - Да, переход дальше.
 - Нет, поиск элементов в текущем родителе с включенными предшественниками или их отсутствием.
4. Включаем ген в особь потомка.
5. Переходим на другого родителя и повторяем пункты 3с и d.

Схема оператора скрещивания представлена на рис. 2.

В гибридной схеме формирования начальной популяции учтены знания разработчиков (набор эвристических правил), использован алгоритм случайного задания популяции и алгоритм имитации отжига, который имитирует поведение металла в процессе его отжига (рис. 3).

Далее рассмотрим две трансформационные схемы реализации ГС, в которых для улучшения решений ГА используется эвристический алгоритм Resource. Данный алгоритм учитывает наблюдения разработчика о последовательной

загрузке ресурсов, минимизации времени их простоя, включении в расписание в первую очередь работы с максимальным числом последователей.

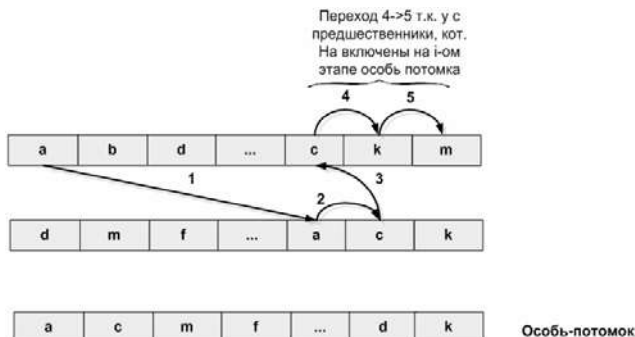


Рис. 2. Схема оператора скрещивания

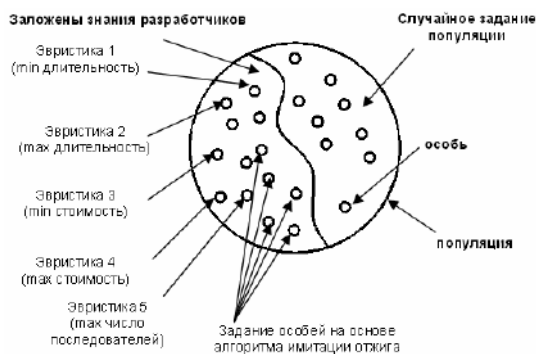


Рис.3. Гибридная схема формирования начальной популяции

В схеме 1 (рис. 4) на каждом поколении работы ГА особи оптимизируется улучшающим алгоритмом Resource, после чего производятся обычные для ГА действия.



Рис. 4. Гибридная схема 1

В схеме 2 (рис. 5) решение, полученное за все время работы генетического алгоритма, улучшается применением алгоритма Resource.



Рис. 5. Гибридная схема 2

Результаты работы гибридного генетического алгоритма. Все операторы ГА и гибридные схемы были реализованы в исследовательской среде MatLab 7.1. Для случая расчета 1000 случайных решений, применение алгоритма Resource позволяет составить расписание с целевой функцией, в среднем равной 200 (рис. 6). Применение генетического алгоритма на основе гибридной схемы формирования начальной популяции позволяет получить значение целевой функции, равной 65 (рис. 7). Реализация трансформационной гибридной схемы 2 в среднем улучшает решение на 5% после применения ГА с гибридной схемой формирования начальной популяции (рис. 8).



Рис. 6. Применение алгоритма Resource

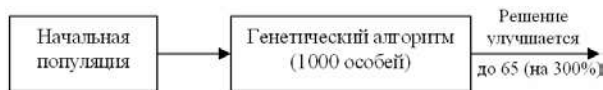


Рис. 7. Применение модифицированного генетического алгоритма

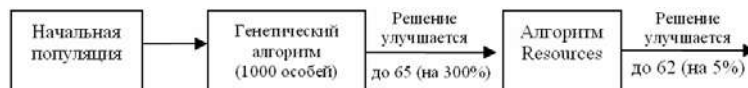


Рис. 8. Применение трансформационной ГС

При этом в случае задания начальной популяции с помощью гибридной схемы получается решение на 30% лучше, чем при случайном задании начальной популяции (рис. 9).

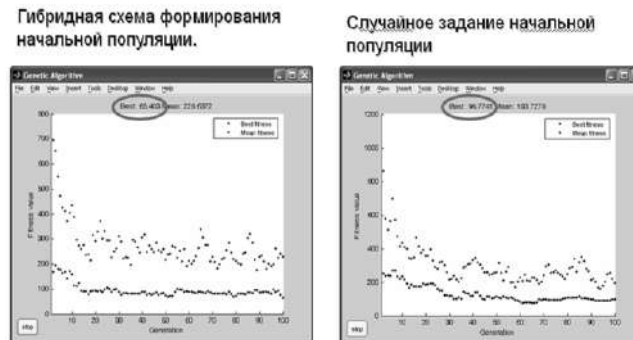


Рис. 9. Сравнение работы ГА при различных способах формирования начальной популяции

Заключение. Экспериментальные исследования показали эффективность предложенных схем реализации гибридного ГА. В результате применения гибридного ГА были получены расписания с лучшим значением целевой функции по сравнению со стандартным генетическим алгоритмом. При этом наилучшие результаты показал гибридный ГА, реализованный на основе трансформационной схемы совместно с гибридной схемой формирования начальной популяции.

Модификацией рассмотренной задачи может быть добавление новых ограничений и ввод новых понятий. Например, ресурсы могут объединяться в смены, в группы взаимозаменяемых ресурсов. Дополнительным ограничением могут стать календари ресурсов и работ. При планировании проекта каждому ресурсу можно назначить календарь, в соответствии с которым будут рассчитываться его рабочие часы и длительность работы.

Дальнейшая работа связана с рассмотрением модульной структуры при составлении расписаний, разработкой эвристических алгоритмов, новых гибридных схем и интерфейса пользователя.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Medsker L.R.* Hybrid Intelligent Systems. – Kluwer Academic Publ., 1995.
2. *Tarek A. El-Mihoub, Adrian A. Hopgood, Lars Nolle, Alan Battersby.* Hybrid Genetic Algorithms: A Review. *Engineering Letters* 13(2): 124-137 (2006).
3. *М. Герц, Д. Джонсон.* Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982.
4. *Емельянов В.В., Курейчик В.М., Курейчик В.В.* Теория и практика эволюционного моделирования. – М.: ФИЗМАТЛИТ, 2003.