

# Матричный алгоритм роя частиц для поддержки принятия решений при составлении расписания обслуживающих бригад

А.С. Филиппова

Институт профессионального образования  
и информационных технологий  
Башкирский государственный  
педагогический университет им М.Акмуллы  
Уфа, Россия  
e-mail: annamuh@mail.ru

Е.В. Андреева

Институт профессионального образования  
и информационных технологий  
Башкирский государственный  
педагогический университет им М.Акмуллы  
Уфа, Россия  
e-mail: elena.ragulina.ufa@gmail.com

Э.И. Дямина

Институт экономики и управления  
Уфимский государственный авиационный  
технический университет  
Уфа, Россия  
e-mail: xasel@mail.ru

Е.Д. Лаптенко

Институт профессионального образования  
и информационных технологий  
Башкирский государственный  
педагогический университет им М.Акмуллы  
Уфа, Россия  
e-mail: eliseylapt@mail.ru

## Аннотация<sup>1</sup>

Рассматривается оптимизационная задача составления расписания и маршрутов движения обслуживающих бригад. Примеры задачи на практике – это обслуживающие бригады удаленных объектов, бригады текущего и капитального ремонта технических объектов и пр. Для поддержки принятия решений подобных проблем предлагается матричный алгоритм роя частиц. Алгоритм роя частиц является метаэвристическим, он позволяет получать множество допустимых решений. В статье описаны матричные способы кодирования. Приведены результаты сравнения различных вариантов алгоритма.

## 1. Обзор предметной области исследования

Увеличение объема обрабатываемой информации при решении комбинаторных задач диктует необходимость в разработке и использовании эффективных эвристических методов и алгоритмов, учитывающих прикладные ограничения. Под

эффективностью надо понимать как возможность получения оптимального решения, так и быстроту получения рационального решения.

Проблема составления расписания обслуживающих бригад предлагается рассматривать как частный случай NP-трудной задачи маршрутизации транспорта (Vehicle Routing Problems, VRP) [1]. Задача, рассматриваемая в данной статье заключается в следующем: для множества транспортных средств (агентов), расположенных в одном или нескольких депо, должен быть определен набор маршрутов до нескольких отдаленных пунктов (клиентов), так чтобы минимизировать время на посещение агентами клиентов. При решении задачи и составлении расписания и маршрутов движения агентов возможно учитывать время необходимое на обслуживание в каждом пункте. Практическими примерами являются задача составления графиков работ бригад текущего и капитального ремонта скважин [2]; составление расписания и маршрутов срочного и планового обслуживания банкоматов; составление маршрутов для доставки товаров из магазинов покупателям и т.п.

Предметная область задачи может обуславливать свои условия и ограничения к общей постановке. Так например, при создании графиков работ бригад на добывающих нефтяных предприятия решение о назначении бригады на выполнение работы зависит от ряда параметров [2] таких как длительности и сложности переезда на новое место, наличия

Труды Шестой всероссийской научной конференции "Информационные технологии интеллектуальной поддержки принятия решений", 28-31 мая, Уфа-Ставрополь, Россия, 2018

<sup>1</sup>Всероссийская научная конференция "Информационные технологии интеллектуальной поддержки принятия решений", Уфа-Ставрополь, Россия, 2018

необходимых приборов и оборудования в оснастке бригады, готовности скважины к ремонту на момент приезда бригады, планируемой длительности ремонта, показателей работы. Процесс решения состоит из двух этапов – решение подзадачи нахождения путей на графе и подзадачи распределения заданий по бригадам. При нахождении путей на графе следует учитывать возможность выбора не только минимальных путей, но и более длинных путей, так как итоговая оценка зависит от затрат на перемещение бригады и эффекта от планируемых работ скважины до и после ремонта и т.п.

При составлении расписания и маршрутов бригад обслуживающих банкоматы важен характер обслуживания. Это например, обслуживание банкомата, не затрагивающее устройств, находящихся внутри сейфа, которое проводится оператором банкомата без привлечения других служб учреждения банка в соответствии со своими функциональными обязанностями. К такому обслуживанию относятся замена бумажной ленты на принтерах банкомата, замена красящих ленточек на принтерах банкомата, изъятие контрольной ленты журнального принтера (файла журнала протокола работы банкомата), срочное изъятие задержанных карточек, контроль за внешним видом банкомата (чистотой, наличием информационного материала и т.п.). Все остальные работы по обслуживанию и ремонту узлов банкомата проводятся при полном отсутствии кассет с денежной наличностью, сбросовой кассеты в банкомате и после вывода банкомата из режима обслуживания клиентов. Кроме того, при составлении расписания обслуживания банкоматов клиентов, нужно учитывать срочность обслуживания. Что несомненно должно быть учтено при алгоритмическом обеспечении и принятии решения.

Для решения подобных задач маршрутизации разработано достаточно много алгоритмов [1, 3]. Поиск решений VRP начался в 60-е годы XX века. Предложенные точные методы решения VRP, как, например, метод ветвей и границ, но время вычислений при их применении растёт слишком быстро. VRP предполагает значительно большее количество вариантов решений для просмотра, чем задача коммивояжера (Travelling Salesman Problem, TSP) при одинаковом количестве вершин. Более эффективными для практического применения считаются эвристические и метаэвристические алгоритмы [4]. Особенность метаэвристических алгоритмов в том, что они не дают точного описания порядка действий для решения задачи, и каждый из них должен быть дополнительно конкретизирован путём подбора значений управляющих параметров. Авторы метаэвристических алгоритмов приводят константы, дающие по их мнению наиболее удачные результаты, но в некоторых ситуациях оказывается возможным найти более качественные решения, если

провести дополнительные исследования влияния параметров. Как правило в метаэвристиках используется кодировка начального решения к которому применяю процедуры построения множества решений с целью нахождения локального оптимума. Таким образом, кроме управляющих параметров влияние на качество решения оказывает и способ кодирования и декодер.

Для решения комбинаторных задач известны технологии конструирования алгоритмов позволяющие формировать некоторое множество вариантов алгоритмов имеющих общую структуру. [5]. Для разработки алгоритмов воспользуемся матричной технологией.

В данной статье для решения задача составления расписания и маршрутов движения обслуживающих бригад предлагается использовать и исследуется метод роя частиц [6] с матричным декодером.

## 2. Постановка задачи

Рассматривается базовая задача составления расписания и маршрутов движения обслуживающих бригад [7].

Транспортная сеть задана полным графом  $G = (V, E)$ ,  $V = v_0, \dots, v_n$ , где вершина  $v_0$  интерпретируется как склад, а  $v_1, \dots, v_n$  – клиенты. Каждому ребру  $e_{ij} \in E$ ;  $i, j = 0..n$  сопоставлена характеристика времени движения из пункта  $v_i$  в  $v_j$ , где  $i, j = 0..n$ . Задано множество агентов  $A = (a_1, \dots, a_q)$ .

Требуется найти такое разбиение множества  $V$  на непересекающиеся подмножества  $V = \bigcup_{k=1}^q V^{a_k}$ ;  $V^{a_k} \cap V^{a_l} = \emptyset$ ;  $l \neq k$ ;  $l, k = 1..q$  и для каждого  $V^{a_k}$  определить последовательность вершин, то есть маршрут движения  $R^{a_k}$ ,  $k = 1..q$  агента по клиентам. Расписанием  $R$  назовем множество маршрутов  $(R^{a_1} \dots R^{a_q})$  для всех агентов.

Необходимо найти расписание  $R$ , для которого самый долгий по времени маршрут будет минимальным.

Кроме того, для практического решения модель задачи может быть дополнена и учтены следующие условия: каждому клиенту приписывается время обслуживания, тогда длина маршрута будет дополнена временем на обслуживание клиентов входящих в маршрут; каждому клиенту приписывается некоторый промежуток времени в течение которого возможно обслуживание клиента и пр.

Для принятия решения в конкретной области необходимо будет учитывать и другие численные и экспертные оценки и данные. Мы ограничимся в рамках данного исследования базовой постановкой задачи.

### 3. Матричный алгоритм роя частиц

#### 3.1. Описание идеи алгоритма роя частиц

Оптимизация с использованием роящихся частиц (Particle Swarm Optimization, PSO) [3, 8] является методом стохастической оптимизации и является близким к эволюционным алгоритмам. Но он моделирует не эволюционный процесс, а ройное поведение насекомых или стадное у животных. Т.е. он не создает новые популяции, а направлен на улучшение одной популяции, используя направленную мутацию [3].

Рой частиц представляет собой множество объектов, обладающих роевым интеллектом. Например, колония муравьев или стая птиц. Кружа в небе, каждая из птиц следит за своими сородичами и координирует свое движение согласно их положению. А найдя источник пищи, она оповестит их об этом. На этом факте и основывается метод роя частиц. Источники пищи обычно расположены случайным образом, поэтому в одиночестве птица вполне может погибнуть, не найдя ни один в течение долгого времени. Однако, если все птицы будут делиться с сородичами информацией о находках, шансы каждой из них на выживание резко повысятся. Частицы роя никогда не умирают, а изменяются направленной мутацией. Также определены модуль и направление скорости, в котором частица перемещается в каждый момент времени.

Отслеживаются следующие моменты:

- наиболее приспособленное положение  $x^*$ , найденное частицей  $x$ ;
- наиболее приспособленное положение  $x^+$ , найденное всеми информаторами частицы  $x$ ;
- наиболее приспособленное положение  $x^!$ , найденное всеми частицами.

#### 3.2. Способы применения алгоритма роя частиц к задаче расписания и маршрутизации обслуживающих бригад (агентов)

**Декодер с матрицами весов.** Основной проблемой в реализации алгоритма для решения задачи расписания был вопрос о способе перехода от задачи в вещественном пространстве к дискретной задаче на графе, на котором построен алгоритм роя частиц. Также определены модуль и направление  $V$  в качестве такого способа предлагается использовать декодер с матрицами весов.

Вектор координат, изначально заполненный случайными числами в диапазоне  $[-1;1]$ , хранится в одномерном массиве, из элементов которого будут составлены матрицы  $A=\{a_{ij}\}$ ;  $C=\{c_{ji}\}$ ; где  $a_{ij}$  – значимость агента  $i$  для клиента  $j$ ;  $c_{ji}$  – значимость клиента  $j$  для агента  $i$ ,  $i=1..m$ ;  $j=1..n$ . Соотносим

агентов и клиентов. В  $i$ -й строке матрицы  $C$  находим максимальный элемент, пусть этот элемент находится в столбце  $j$ . Тогда клиента  $i$  посетит агент  $j$ . Таким образом, имеем для каждого агента список посещаемых клиентов.

Сортируем список, определяющий порядок обхода. В  $i$ -й строке матрицы  $A$  сортируем элементы, соответствующие списку посещаемых клиентов по убыванию. Имеем порядок посещения.

Составляем матрицу расписаний  $R$  размерности  $m*(n+3)$ .  $R = \begin{pmatrix} 0 & \{\text{список}\} & 0 \\ \dots & \dots & \dots \\ \dots & \dots & \dots \end{pmatrix}$ , где  $\{\text{список}\}$  – это отсортированный список с номерами вершин клиентов, то есть маршрут. Во всех строках матрицы  $R$  расположены маршруты для каждого агента.

Далее вычисляем целевую функцию по времени, как длительность прохождения пути самым медленным агентом:  $T = \max_i \left( \sum_{j=1}^{n+2} D_{R_{i,j}, R_{i,j+1}} \right) \rightarrow \min$ , где  $D$  – вес ребра  $(c_j, c_{j+1})$ .

Недостаток данного метода состоит в низкой скорости вычислений, в связи с высокой сложностью вычисления времени расписания, равной  $O(m*n^2)$ .

**Декодер с вектором соответствия.** Второй вариант реализации алгоритма PSO с декодером в виде вектора соответствия.

Вектор положения частицы делится на 2 участка размерности  $n$  (количество клиентов) и  $m+1$  (количество агентов +1) соответственно, которые задают веса сортировки.

Первый участок задает порядок обхода клиентов, второй задает деление расписания по агентам. Таким образом, размерность вещественного пространства в алгоритме PSO сокращается с  $2*n*m$  до  $m+n+1$ .

### 4. Вычислительный эксперимент

Программная реализация алгоритмов роя частиц с двумя описанными видами декодеров позволила провести вычислительный эксперимент с целью определения эффективных параметров. Для проведения эксперимента были выделены следующие классы задач: класс А – размерность матрицы весов  $10 \times 10$ ; класс В – размерность матрицы весов  $30 \times 30$ ; класс С – размерность матрицы весов  $100 \times 100$ .

Для каждого класса были сгенерированы случайные исходные данные ста пятидесяти примеров и вычислялось среднее значение процента улучшения целевой функции по сравнению с жадным алгоритмом для каждого класса.

Кроме того, для каждого класса были проведены эксперименты, в которых исследовалось количество итераций, необходимых для сходимости алгоритма.

В классе А тестирование показало, что, несмотря на большое количество итераций по времени, равное

1000, алгоритм роя частиц находит решение не более чем за 30 итераций, а в половине случаев достаточно одной итерации. При этом из 50 тестов лишь в шести время расписания, полученного декодерами «Матрицы весов» и «Вектор соответствия», незначительно различается.

Учитывая малую размерность графа, найденные расписания в худшем случае являются локальным оптимумом, а в лучшем - глобальным.

Значит для нахождения решения на графах малой размерности, можно уменьшить время симуляции и сократить время работы программы. Те случаи, когда расписания, полученные двумя методами, различаются, свидетельствуют о том, что частицы недостаточно хорошо исследовали область поиска. Это можно исправить настройкой констант в алгоритме роя частиц, см. рис. 1. Для графов размерности 100 на 100 такой закономерности не выявлено.

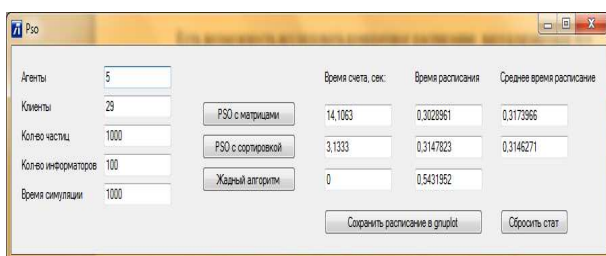


Рис. 1. Окно настройки управляющих параметров

В классе С был проведен эксперимент, сравнивающий время работы разработанных декодеров «Матрицы весов» и «Вектор соответствия».

Декодер «Матрицы весов» требует значительно больше времени для счета, но это компенсируется полученными результатами, которые по эффективности в графах больших размерностей превосходят результаты декодера «Вектор соответствия».

Анализ вычислительного эксперимента показал в среднем на исследуемых тестовых примерах улучшение значения целевой функции алгоритмом роя частиц по сравнению с жадным алгоритмом: среднее улучшение по всем примерам составляет 15,98% (наименьшее – 0,11%, наибольшее – 89,51%).

#### 4. Заключение

- Использование идеи метода роя частиц позволило разработать алгоритмы решения задачи составления расписания и маршрутов движения обслуживающих бригад с декодерами на матричной основе. Использование метода роя частиц в поставленной задаче составления расписания позволило в целом получать более эффективные расписания, чем это возможно при помощи жадного алгоритма.

- Проведенный первоначальный вычислительный эксперимент и анализ его результатов показал эффективность предлагаемых способов решения, их перспективность и необходимость в проведении дальнейшего исследования и корректировки управляющих параметров.

- Обзор примеров задач практического использования в конкретных предметных областях (нефтяной и банковской сферах) показал, что необходимо будет сформировать соответствующие ограничения и требования к поиску решения и учесть их в реализации процесса поддержки принятия решений.

#### Список используемых источников

- The VRP Web; URL: <http://neo.lcc.uma.es/vrp/> (дата обращения: 01.03.2018).
- Филиппова А.С., Фролов Р.В. О создании оперативных сетевых графиков работ бригад текущего и капитального ремонта скважин // –М.: Информационные технологии, 2011, №6. –С. 39 – 41.
- Luke S. Essentials of Metaheuristics. / Luke S. // – 1st ed. – San Francisco (Calif): Creative Commons., 2009. – 237 p.: ill; Bibliogr: p. 214-216. – ISBN: 978-0-557-14859-2.
- Р.Р. Рамазанова. Обзор основных задач маршрутизации и методов их решения / Р.Р. Рамазанова, А.С. Филиппова // Вычислительная техника и новые информационные технологии: межвузовский научный сборник / Уфимск. гос. авиац. техн. ун-т. – Уфа: Уфимск. гос. авиац. техн. ун-т. 2015. -С.16 – 21.
- А.С. Филиппова. Технологии разработки алгоритмов геометрического размещения / А.С. Филиппова, Э. И. Дямина // Proceedings of the 4th International Conference “Information Technologies for Intelligent Decision Making Support”, May 17-19, Ufa, Russia, 2016. V.1 -P.219-223.
- Карпенко А.П., Селиверстов Е.Ю. Обзор методов роя частиц для задачи глобальной оптимизации (Particle Swarm Optimization) // Наука и образование. МГТУ им. Н.Э. Баумана. Электрон. журн. 2009. № 3.
- Филиппова А.С. Решение задач составления расписаний / Филиппова А.С., Рагулина (Андреева) Е.В., Бадгетдинова К.М. // Проблемы оптимизации и экономические приложения: материалы V Всероссийской конференции. – Омск: Изд-во Ом.гос.ун-та, 2012. –С.166.
- Kennedy, J., Eberhart, R. Particle Swarm Optimization // Proceedings of IEEE International Conference on Neural Networks. 1995. Vol. IV., P. 1942–1948.