

МЕТАЭВРИСТИЧЕСКИЕ АЛГОРИТМЫ ДЛЯ МНОГОАГЕНТНЫХ ЗАДАЧ МАРШРУТИЗАЦИИ¹

М.С. Германчук, Д.В. Лемтюжникова, В.А. Лукьяненко

Аннотация. Рассматриваются задачи построения маршрутов в сложных сетях многими агентами-коммивояжерами. Формализация приводит к задачам псевдобоулевой дискретной оптимизации с ограничениями, учитывающими специфику построения маршрутов. Разреженность матрицы ограничений позволяет применять декомпозиционные подходы и кластеризацию сети. Разработка приближенных алгоритмов выбора маршрутов в сложных сетях связана с учетом свойств структуры сети, ее сложности, наличия ограничений, предписаний, условий достижимости, числа агентов-коммивояжеров. Показано, что решение задач маршрутизации может базироваться на применении многоагентного подхода в сочетании с кластеризацией (декомпозицией) исходной задачи и метаэвристик. Многоагентные системы с роевым интеллектом применяются для решения сложных задач дискретной оптимизации, которые нельзя эффективно решить с помощью классических алгоритмов. Агентная модель для сложной сети задачи типа многих коммивояжеров становится интеллектуализированной системой, определяющей эвристические алгоритмы поиска оптимального решения реактивными агентами (следующими заложенным в них правилам). Применяются композиции подробно описанных алгоритмов, которые хорошо себя зарекомендовали в вычислительных экспериментах: модификации генетического алгоритма, муравьиного, роевого (пчелиной колонии), имитации отжига. Предложен и реализован обобщенный алгоритм, в котором исходной сети ставится в соответствие более простая сеть (сеть облета). Численный эксперимент в этом случае проведен для задачи маршрутизации по карте ГИС для городской инфраструктуры. Реализованы алгоритмы кластеризации, в которых первоначально пройденные маршруты уточняются с помощью алгоритмов 2-орт, имитации отжига и других метаэвристик. Дается сравнение применяемых алгоритмов и иллюстрация их работы.

Ключевые слова: метаэвристические алгоритмы, многоагентные задачи оптимизации, дискретная оптимизация, псевдобоулевые задачи.

ВВЕДЕНИЕ

Моделирование процессов на сетях определяется сложностью структуры сетей. Большинство известных сетей — социальные, инфраструктурные, ресурсные, транспортные — сложные. Актуальными являются задачи построения маршрутов на таких сетях, например, задача маршрутизации типа многих коммивояжеров. С другой стороны, задачи маршрутизации на сложных сетях приводят к типичным примерам постановок в виде задач дискретной оптимизации (ДО) с разреженными матрицами [1, 2], для которых применимы декомпозиционные подходы [3]. Однако не меньший интерес

представляют метаэвристические алгоритмы на основе моделей реальных процессов генетики, движения пчелиного роя и муравьиных колоний, отжига и др. В работе применен многоагентный подход, который реализуется с помощью эволюционных алгоритмов.

Рассматриваются многоагентные задачи маршрутизации (Multiple Traveling Salesman Problems, MTSP). В классической постановке коммивояжер должен объехать все города и вернуться в начальную точку. Задача MTSP формулируется как обобщение этой задачи для многих коммивояжеров, причем в качестве стандартных целевых функций в соответствующей задаче ДО выступают минимизация общей стоимости пути m обходов всех коммивояжеров или минимизация самого длинного обхода для любого коммивояжера.

¹ Работа выполнена при финансовой поддержке РФФИ (грант № 18-31-00458, № 20-58-S52006).

Многие задачи этого типа содержат огромное число неизвестных и ограничений, поэтому они трудно решаемы. Известно, что MTSP в общем виде является NP-трудной, поэтому для больших задач весьма актуальными представляются способы уменьшения вычислений.

Разнообразие задач маршрутизации определяется структурой сети, на которой нужно выбрать наилучший маршрут. Актуальность таких задач следует из их практической полезности. Разнообразие публикаций по этой тематике подтверждает необходимость разработки приближенных алгоритмов выбора оптимальных маршрутов в сложных сетях — на графах большого размера. Задачи маршрутизации представляют собой важную составляющую прикладной (алгоритмической) теории графов, в рамках которой рассматривается построение различного рода путей, маршрутов, распределение потоков, разбиение, кластеризация и декомпозиция графов. Разработка алгоритмов в таких задачах связана с учетом свойств структуры сети, ее сложности, наличия ограничений, предписаний, условий достижимости, числа агентов-коммивояжеров.

В статье [4] показано, что для MTSP учет дополнительной информации меняет математическую постановку задачи и алгоритмы ее решения. В многоагентном подходе возникает свой класс задач, обусловленных поведением и управлением интеллектуальными агентами в условиях взаимодействия, например, в анализе алгоритмов маршрутизации, основанных на кластеризации графов. В работе предполагается, что сочетание различных постановок задач (псевдобулевой оптимизации большой размерности) с кластеризацией (декомпозицией), с применением генетических и эволюционных алгоритмов (метаэвристик [5]) позволяет эффективно применять многоагентный подход для решения задач MTSP.

В настоящей статье:

- приведен краткий обзор многоагентных систем для задачи коммивояжера;
- рассмотрен класс задач прикладной теории графов, который представляет собой продолжение исследования структуры сложных сетей;
- сделаны новые постановки задач маршрутизации типа многих коммивояжеров;
- предложен и реализован обобщенный алгоритм, в котором исходной сети ставится в соответствие более простая сеть, дугам соответствует расстояние по прямой (сеть облета);
- реализованы алгоритмы кластеризации, первоначально найденные маршруты уточняются с помощью алгоритмов 2-opt, имитации отжига и других метаэвристик;
- в результате вычислительного эксперимента выбраны такие алгоритмы, как обобщенный ге-

нетический, имитации отжига, обобщенного роя и др., которые необходимы для наполнения многоагентной системы;

— проведен численный эксперимент на примере многослойной сети ГИС Большой Ялты;

— проведен эксперимент, подтверждающий необходимость применения больших вычислений.

1. ОБЗОР МНОГОАГЕНТНЫХ ЗАДАЧ МАРШРУТИЗАЦИИ

Многоагентные системы (МАС) состоят из самостоятельных интеллектуальных агентов (в нашем случае — агентов-коммивояжеров), взаимодействующих друг с другом и с окружающей средой. Агентом может быть программа со специфическими характеристиками или абстрактный интеллектуальный агент, представляющий собой формализованное описание действующего в реальной системе лица. У каждого агента есть свое представление о среде, текущее состояние, логика и цели, формирующие его поведение. Работа агента состоит в сборе и последующей обработке информации для принятия решения. Агенты могут быть независимыми или соревноваться между собой за некоторые ресурсы. Главным отличием МАС считаются интеллектуальность ее агентов и развитость их представлений о внешней среде. Агенты делятся на когнитивных (интеллектуальных) и реактивных (следующих заложенным в них правилам). В первую группу входят МАС, построенные на небольшом числе высокоинтеллектуальных агентов (сложные программы или системы искусственного интеллекта), пользующихся обучающими нейронными сетями или экспертными системами. При этом система отвечает за распределение задач между исполнителями и обеспечивает общение агентов и их взаимодействие со средой. Вторая группа МАС строится с использованием большого числа (десятки — тысячи) агентов с простой структурой, которые не обладают сложным поведением. Их особенность заключается в проявлении сложного поведения как результате совместных действий множества простых реактивных агентов. Такие МАС называют многоагентными системами с роевым интеллектом. Как правило, возможности МАС используются для решения сложных задач ДО, которые нельзя эффективно решить классическими алгоритмами. Если в качестве среды взять сложную сеть задачи многих коммивояжеров, то агентная модель становится интеллектуализированной системой, определяющей эвристические алгоритмы для поиска оптимального решения реактивными агентами. Например, колония муравьев способна находить кратчайший путь до цели и может быть задействована агентом-коммивояжером для нахождения кратчайшего замкнутого мар-

шрута (гамильтонового цикла во взвешенном графе) [6, 7].

За годы исследований задачи коммивояжера было предложено большое число точных и эвристических алгоритмов [5]. Из точных применяются алгоритмы динамического программирования и алгоритмы, базирующиеся на методе ветвей и границ, а на практике чаще пользуются генетическими и эволюционными алгоритмами (алгоритмами роя частиц, муравьиной колонии [8]).

Приведем краткий обзор современных методов решения MTSP на основе моделирования реальных процессов. Самым распространенным для решения MTSP является генетический алгоритм (ГА). В работе [9] MTSP предлагается в качестве модели глобальной задачи оптимизации сбора нектара цветов, например, нектарными летучими мышами. Модель включает в себя несколько независимых животных и много цветов, содержание нектара в которых зависит от времени. Авторы с помощью ГА нашли оптимальное значение целевой функции с экспериментально полученными параметрами. Для определения расстояний применяется распределение Леви, что типично для естественных собирателей нектара. В отличие от многих моделей, в статье не делается предположений о природе распределения дальности полета. Представлены данные полевых экспериментов в Коста-Рике. Кроме подтверждения работоспособности модели авторы обнаружили, что летучие мыши способны запоминать положения источников пищи и частично оптимизировать свои маршруты.

Моделирование поведения капли как агента для поиска решения в задаче о рюкзаке было предложено в 2008 г. иранским ученым в работе [10].

В статье [11] рассматривается MTSP с минимизацией суммы расстояний всех маршрутов. Модель включает в себя несколько складов, закрытый путь и требование минимального числа городов, которые должен посетить каждый агент. Для решения задач предлагается два ГА. Первый применяется в сочетании с выбором рулетки и элитарным выбором, в котором предлагаются четыре новых вида мутационной операции. Второй связывает отбор и мутацию вместе. Применяются новый оператор селекции и более полный оператор мутации. Для сравнительного анализа приводятся алгоритм оптимизации роя частиц и алгоритм оптимизации инвазивных сорняков. Алгоритмы проверяются с помощью общедоступных тестов TSPLIB¹. Производительность оценивается с помощью серии сравнительных экспериментов. Показано, что второй ГА показывает лучшие результаты в сравнении с алгоритмами роя частиц и инвазивных сорняков.

¹ <https://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/index.html>.

В работе [12] предложены новые операторы для основных шагов ГА: скрещивания и инициализации популяции. Основанная на теории групп методика генерации обеспечивает уникальность членов в популяции и, следовательно, отсутствие избыточности в пространстве поиска, а также устраняет эффект случайной инициализации. В предложенном операторе скрещивания расстояние Хэмминга сохраняется и существует очень мало шансов произвести нового потомка, который совпадет с членом популяции. Для эффективного представления пространства поиска применяется метод представления нескольких хромосом для кодирования пространства поиска MTSP. Авторы оценивают и сравнивают предложенную методику с методами, включающими описанные в статье [13] операторы скрещивания для двух стандартных целевых функций. Экспериментальные результаты показывают, что предложенный ГА дает лучший результат по сравнению со всеми четырьмя методами для второй целевой функции MTSP.

В статье [14] предлагается новый эффективный ГА с локальными операторами для решения MTSP и получения высококачественного решения в разумные сроки для реальных приложений. Два новых локальных оператора предназначены для ускорения конвергенции процесса поиска и повышения качества решения. Результаты показывают, что алгоритм находит лучший набор путей с экономией 9,62 % в среднем по стоимости.

Также в литературе встречается применение гибридных алгоритмов на базе ГА. В работе [15] предлагается новый гибридный подход, который представляет собой комбинацию трех алгоритмов: модифицированной колонии муравьев, 2-орт и ГА. С помощью алгоритма муравьиной колонии генерируются решения, на которых применяется алгоритм 2-орт для их улучшения. Далее происходит улучшение качества решений посредством ГА. Причина объединения вышеупомянутых алгоритмов заключается в использовании их сильных сторон как в глобальном, так и в локальном поиске. Предлагаемый подход оценивается с участием различных экземпляров данных из стандартных контрольных показателей. По критериям TSPLIB для больших задач предложенный алгоритм показывает лучшие результаты, чем нынешний наиболее известный M-GELS (Modified Gravitational Emulation Local Search) подход [16]. Для задач меньшей размерности он демонстрирует лучшие результаты, чем другие подходы, и сопоставимые результаты с M-GELS.

Разделение задачи между роем агентов не только позволяет сократить время поиска, увеличить вероятность нахождения глобального лучшего решения, но и создает широкий простор для применения многоагентных алгоритмов: от решения оптимизации перевозки грузов до управления роём

боевых роботов. Введение нескольких роев возникает в случае преобразования задачи многих коммивояжеров в несколько обычных задач коммивояжера. Область применения такого подхода только расширяется, поскольку зачастую при решении вопроса перевозки грузов число агентов-перевозчиков больше одного и тогда необходимо разделить, например, города между агентами как можно более эффективно [17].

В докладе [18] осуществляется группирование городов в кластеры, где каждый кластер представляет собой набор смежных городов, а затем применяется один из хорошо известных оптимизационных подходов для поиска оптимального маршрута для каждого кластера. Авторы применяют алгоритм муравьиных колоний, а также ГА для последовательного и параллельного программирования. Работы [15, 17, 18] наиболее близки к подходу, применяемому в настоящей работе.

Для решения MTSP в статье [13] авторы рассмотрели две различные целевые функции. Задача первой — свести к минимуму общее расстояние, пройденное всеми коммивояжерами, а второй — свести к минимуму максимальное расстояние, пройденное любым коммивояжером. Вторая целевая функция касается баланса рабочей нагрузки между коммивояжерами. Также предложены два метаэвристических подхода к MTSP. Первый подход основан на алгоритме искусственной пчелиной колонии, а второй — на алгоритме оптимизации инвазивных сорняков. Применен локальный поиск для дальнейшего улучшения решения, полученного с помощью представленных подходов. Результаты вычислительных экспериментов показывают превосходство предложенных алгоритмов над всеми другими современными подходами к этой задаче для обеих целевых функций.

В работе [19] авторами предложена система муравьиных колоний. Описаны два алгоритма, сочетающие кластеризацию K -средних и нечетких C -средних с системами муравьиных колоний. Экспериментально исследуется эффективность предложенных алгоритмов с целевой функцией, рассчитывающей общую длину/стоимость решения и степень его балансировки, измеряемую как амплитуда его подуровней.

Таким образом, существует множество подходов к решению многоагентных задач маршрутизации. Публикации подтверждают целесообразность выбора алгоритмов, используемых в работе.

2. ПОСТАНОВКИ ЗАДАЧ МАРШРУТИЗАЦИИ ТИПА МНОГИХ КОММИВОЯЖЕРОВ

Задача с несколькими агентами-коммивояжерами в общем случае может быть формализована так. Дано $n > 1$ вершин и m агентов-коммивоя-

жеров, расположенных в одной вершине-депо. Оставшиеся вершины, которые необходимо посетить, будем называть промежуточными вершинами. Многоагентный подход к решению задачи MTSP включает в себя нахождение маршрутов для всех m коммивояжеров, которые начинают и заканчивают свой путь в определенной вершине, при этом посещая каждую промежуточную вершину не более одного раза, минимизируя общую длину маршрута для всех агентов.

Рассмотрим формализацию задач многих коммивояжеров при расположении всех агентов в одной вершине-депо. Пусть дан граф $G(V, U)$, где V — множество вершин $V = \{0, 1, \dots, n\}$, U — множество дуг (ребер) и $C = (c_{ij})$ — матрица весов (расстояний), связанная с каждой дугой $(i, j) \in U, i, j = \overline{1, n}$. Пусть m коммивояжеров расположены в вершине-депо $i = 0$. Многоагентная постановка при расположении всех агентов в одной вершине-депо включает в себя нахождение всех маршрутов для m коммивояжеров, таких, что они начинаются и заканчиваются в одной вершине. Все остальные вершины распределены по конкретным маршрутам. Число вершин, посещаемых агентом, находится в пределах предопределенного интервала, и общая стоимость посещения всех вершин минимизируется. Пусть

$$x_{ij} = \begin{cases} 1, & \text{если агент проходит по дуге } (i, j), \\ 0, & \text{в противном случае,} \end{cases}$$

u_i — число вершин, посещенных от источника до вершины i (т. е. номер посещения i -й вершины); L — максимальное число вершин, которые коммивояжер может посетить; K — минимальное число вершин, которые коммивояжер должен посетить, т. е. $K \leq u_i \leq L$. Формализация многоагентной задачи коммивояжера в этом случае имеет вид:

Найти

$$\min \sum_{(i,j) \in U} c_{ij} x_{ij}, \quad (1)$$

$$\sum_{j=2}^n x_{1j} = \sum_{j=2}^n x_{j1} = m, \quad (2)$$

$$\sum_{i=1}^n x_{ij} = 1, \quad j = \overline{2, n}, \quad \sum_{j=1}^n x_{ij} = 1, \quad i = \overline{2, n}, \quad (3)$$

$$u_j + (L - 2)x_{1i} - x_{i1} \leq L - 1, \quad i = \overline{2, n}, \quad (4)$$

$$u_i + x_{1i} + (2 - K)x_{i1} \geq 2, \quad i = \overline{2, n}, \quad (5)$$

$$x_{1i} + x_{i1} \leq 1, \quad i = \overline{2, n}, \quad (6)$$

$$u_i - u_j + Lx_{ij} + (L - 2)x_{ji} \leq L - 1, \quad 2 \leq i \neq j \leq n, \quad (7)$$

$$x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in U. \quad (8)$$

Целевая функция (1) минимизирует общее пройденное расстояние в маршруте. Условия (2) гарантируют, что m коммивояжеров начинают и заканчивают свой путь в одной вершине. Равенства (3) представляют собой ограничения посещения вершин. Условия (4), (5) налагают ограничения на число вершин, которые коммивояжер посетит (при $u_i = 1$, если i — первая вершина в маршруте). Ограничение (6) не позволяет коммивояжеру посещать только одну вершину. Неравенство (7) гарантирует, что $u_j = u_i + 1$ тогда и только тогда, когда $x_{ij} = 1$. Таким образом, ограничения запрещают формирование каких-либо подмаршрутов между вершинами в $V/\{1\}$, как показано в статье [20].

Следующая задача ставится при расположении всех агентов в разных вершинах и представляет собой обобщение рассматриваемой выше задачи, при котором у каждого агента разные вершины-депо. Пусть дан граф $G(V, U)$, где V — множество вершин $V = \{0, 1, \dots, n\}$ и U — множество дуг $U = \{(i, j): i, j \in V, i \neq j\}$, n — число вершин. Множество всех вершин графа есть объединение $V = D \cup \tilde{V}$, где D — множество стартовых позиций агентов — депо, а $\tilde{V}$ — остальные вершины. В узле i расположен коммивояжер m_i , общее число агентов — m . Пусть $\tilde{V} = \{d + 1, d + 2, \dots, n\}$ будет множеством клиентов, где $|D| = d$; C — матрица стоимостей переходов из одной вершины в другую графа G , которая обладает свойствами: $c_{ij} = c_{ji}$ и $c_{ij} + c_{jk} \leq c_{ik}$ для каждого $i, j, k = \overline{1, n}$.

Пусть x_{ij} , L , K определены, как и раньше. Тогда получим формализацию многоагентной задачи коммивояжера MTSP:

найти

$$\min \sum_{(i,j) \in U} c_{ij} x_{ij}, \quad (9)$$

$$\sum_{j \in \tilde{V}} x_{ij} = m_i, \quad i \in D, \quad \sum_{i \in \tilde{V}} x_{ij} = m_j, \quad j \in D, \quad (10)$$

$$\sum_{i \in \tilde{V}} x_{ij} = 1, \quad j \in \tilde{V}, \quad \sum_{j \in \tilde{V}} x_{ij} = 1, \quad i \in \tilde{V}, \quad (11)$$

$$u_j + (L - 2) \sum_{k \in D} x_{ki} - \sum_{k \in D} x_{ik} \leq L - 1, \quad i \in \tilde{V}, \quad (12)$$

$$u_i + \sum_{k \in D} x_{ki} + (2 - K) \sum_{k \in D} x_{ik} \geq 2, \quad i \in \tilde{V}, \quad (13)$$

$$x_{ki} + x_{ik} \leq 1, \quad k \in D, \quad i \in \tilde{V}, \quad (14)$$

$$u_i - u_j + Lx_{ij} + (L - 2)x_{ji} \leq L - 1, \\ i \neq j, \quad i, j \in \tilde{V}, \quad (15)$$

$$x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in \tilde{V}. \quad (16)$$

Целевая функция (9) минимизирует общее пройденное расстояние на каждом маршруте. В данной формулировке для каждого $i \in D$ равенства (10) гарантируют, что коммивояжер m_i начинает путь в вершине i . Равенства (11) представляют собой ограничения посещения вершин. Условия (12) и (13) накладывают ограничения на число вершин, которые коммивояжер посетит, при $u_i = 1$, если i — это первая вершина в маршруте. Ограничение (14) не позволяет коммивояжеру посещать только одну вершину. В статье [20] выявлено, что ограничение (15) разбивает все подмаршруты между агентами.

В постановке (1)–(16) агенты не конкурируют, а обеспечивают минимальный по стоимости (расстоянию) маршрут. В работах [4, 6] в данную модель добавляются ограничения в виде дизъюнктивных нормальных форм (ДНФ). Полученная задача псевдодвулевой условной оптимизации допускает декомпозицию по характерным блокам [1, 2], позволяет выделять задачу о назначении с дополнительными условиями. В случае конкуренции агентов вместо критерия (1) будет m критериев, отвечающих каждому агенту. Для полученной многокритериальной задачи псевдодвулевой оптимизации необходимо найти Парето-оптимальные решения.

3. АЛГОРИТМЫ ДЛЯ РЕШЕНИЯ ЗАДАЧ МАРШРУТИЗАЦИИ

Рассмотрим прикладные модели, методы и алгоритмы маршрутизации многих агентов-коммивояжеров в сложных сетях различной природы. Выбор алгоритмов, их обоснование и реализация определяются целями прикладных задач, вероятностными и метрическими характеристиками сетей, сложностью структуры соответствующих графов. Возникающие задачи ДО, как правило, NP-трудны. Разнообразие алгоритмов связано с наличием априорных знаний о решении или структуре сети, прецедентным характером знаний, а также требованиями к точности решения. Поиск приближенного решения не обходит алгоритмическую сложность таких задач. Более того, рационально применение, как точных, так и приближенных алгоритмов и их композиций. Заметим, что задачи прикладной маршрутизации возникают в сочетании с другими известными задачами, такими как распределение ресурсов, задача о ранце, кластеризации, максимального разреза, покрытия и пр. Поэтому соответствующие методы и алгоритмы должны базироваться на методах локального поиска и основанных на них метаэвристиках [21]. Численная реализация алгоритмов и их тестирование на различных прикладных задачах позволяют сформировать пакет прикладных программ маршрутизации в сложных сетях, что представляет собой актуальную задачу.

Утверждение 1. *Методология разработки алгоритмов решения задач маршрутизации может быть основана на формировании по исходной сложной сети более простой (относительно реализации алгоритмов маршрутизации) по своей структуре сети.*

Далее предлагается такой обобщенный алгоритм, не конкретизированный для многоагентной реализации.

Для решения ряда задач ДО на графах существуют хорошие алгоритмы в случае планарных, метрических графов с весовыми коэффициентами, удовлетворяющими неравенству треугольника $l_{ij} \leq l_{ik} + l_{kj}$, $l_{ij} \geq 0$. В реальных задачах между вершинами заданы расстояния $c_{ij} \geq 0$, где $(i, j) \in U$ — множество дуг, $i \in V$ — множество вершин. Зная координаты вершин $i \in V$, $|V| = n$, можно поставить расстояниям c_{ij} не удовлетворяющим неравенству треугольника, в соответствие расстояния $l_{ij} = \rho(i, j)$, равные расстоянию по прямой между вершинами i и j . Такого типа задачи возникают для задач облета вершин (объектов) на некоторой высоте над поверхностью. Понятно, что в общем случае $c_{ij} \geq l_{ij}$, а некоторым маршрутам нельзя поставить в соответствие маршруты на поверхности, поскольку более короткий путь может лежать через море, горы и другие препятствия. Система запретов и предписаний позволяет выбирать серию модельных графов даже меньшей размерности, допускающих построение приближенных решений за приемлемое время. Для абстрактных сложных сетей построить отображение (обратимое), сводящее исходную задачу к более простой, не всегда возможно. Предлагается следующий обобщенный гибридный алгоритм.

Вход: исходный граф $G(V, U)$ и весовая матрица C .

Выход: приближенное решение задачи ДО на графе $G(V, U)$.

1. Задать граф $G(V, U)$ и весовую матрицу C .
2. Найти преобразование $\rho: C \rightarrow L$, т. е. по матрице C построить матрицу L с элементами $l_{ij} \geq 0$, $(i, j) \in U$, удовлетворяющими неравенству треугольника.
3. Учесть априорную информацию, запреты и предписания; преобразовать матрицу L в матрицу $\tilde{L}$, учитывающую данную информацию.
4. Провести анализ и упрощение матрицы $\tilde{L}$ (метрические характеристики; структурные составляющие: мосты, сочленения, висячие вершины; необходимость кластеризации), сформировать упрощенную матрицу $\bar{\tilde{L}}$.
5. Для упрощенной матрицы $\bar{\tilde{L}}$ решить задачу ДО.
6. Построить обратное соответствие $\bar{\tilde{L}} \rightarrow \tilde{L} \rightarrow C$ и получить вариант решения, проверить на соответствие.

7. Предъявить приближенное решение исходной задачи ДО.

Предложенный обобщенный алгоритм инспирирован рядом актуальных прикладных задач: задачей планирования многодневных туристических маршрутов на инфраструктурной сети достопримечательностей Крыма и задачей доставки ресурсов (воды) экипажами (агентами-коммивояжерами) по территории Большой Ялты в условиях чрезвычайных ситуаций (ЧС). В условиях ЧС МАС маршрутизации является довольно сложной для моделирования системой. Здесь сочетаются задачи выбора решения (принятия решения ЛПР); управления; распределения ресурсов; синтеза сети (вершин-источников ресурсов); устойчивости сети в зависимости от удаления вершины, дуги или некоторого маршрута; кластеризации сети в зависимости от изменяющихся условий ЧС; обмена тактической информацией между агентами; потоковые задачи; задачи прокладки кратчайших путей и замкнутых маршрутов. Для такой МАС необходим набор протестированных приближенных эвристических алгоритмов, композиция которых решает поставленную задачу.

Минимальный набор таких алгоритмов предлагается в настоящей работе для решения задач типа MTSP. Проведена реализация обобщенного гибридного алгоритма для задачи маршрутизации в условиях ЧС. Входной информацией является Яндекс карта с дорожной инфраструктурой Большой Ялты, на которой отмечены источники ресурсов и их потребители. Заданы координаты объектов и можно найти расстояние по прямой или по дорогам, соединяющим эти объекты. В модельном случае варьировалось число коммивояжеров и параметры сети. На рис. 1 (см. третью стр. обложки) приведены случаи решения для двух и пяти коммивояжеров. На этапе кластеризации выделено три кластера, отвечающих Ялте и пригородам (рис. 1, а). На рис. 1, б представлено решение только для центрального кластера (Ялта).

На рис. 1 представлены сбалансированные маршруты (с дополнительным условием равномерного распределения вершин между коммивояжерами). Депо располагается в вершине, близкой к центру масс. Начальное приближение выбрано в результате упорядочивания вершин по углу (координаты вершин приведены к полярным координатам). Такое приближение подходит для алгоритмов имитации отжига или генетических алгоритмов. Для поиска субоптимального маршрута на начальном приближении исследована вариация алгоритма 2-opt, который лежит в основе алгоритма имитации отжига. Дальнейшее улучшение достигается обменом вершинами между коммивояжерами. Полученная структура маршрутов хорошо согласуется с дорожной инфраструктурой.

Рисунки к статье Германчук М.С., Лемтюжиновой Д.В., Лукьяненко В.А.

«МЕТАЭВРИСТИЧЕСКИЕ АЛГОРИТМЫ ДЛЯ МНОГОАГЕНТНЫХ ЗАДАЧ МАРШРУТИЗАЦИИ»

(с. 3–13)

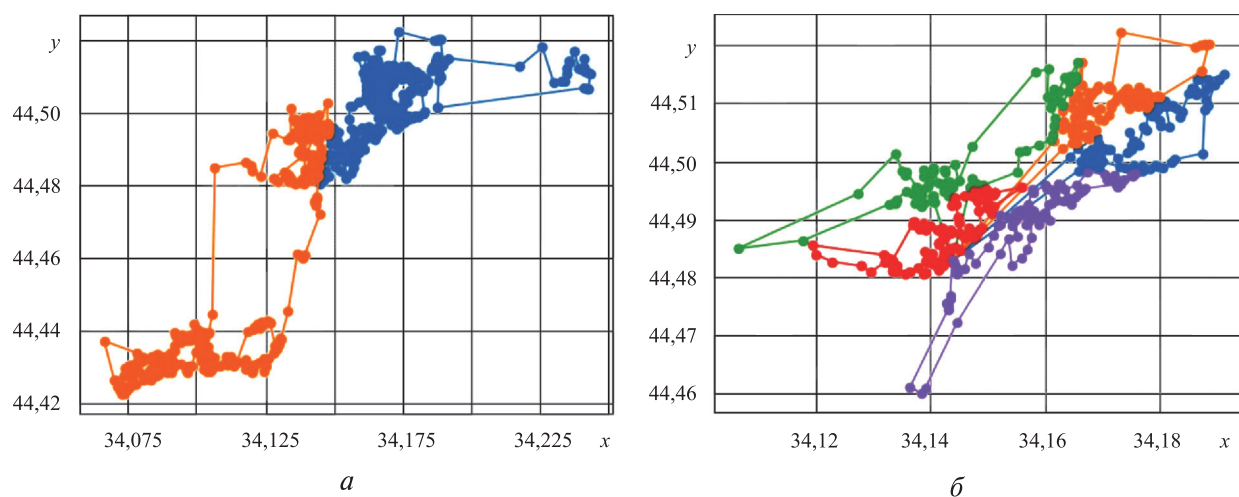


Рис. 1. Решение задачи MTSP для:
а — двух коммивояжеров (Большая Ялта); *б* — пяти коммивояжеров (Ялта)

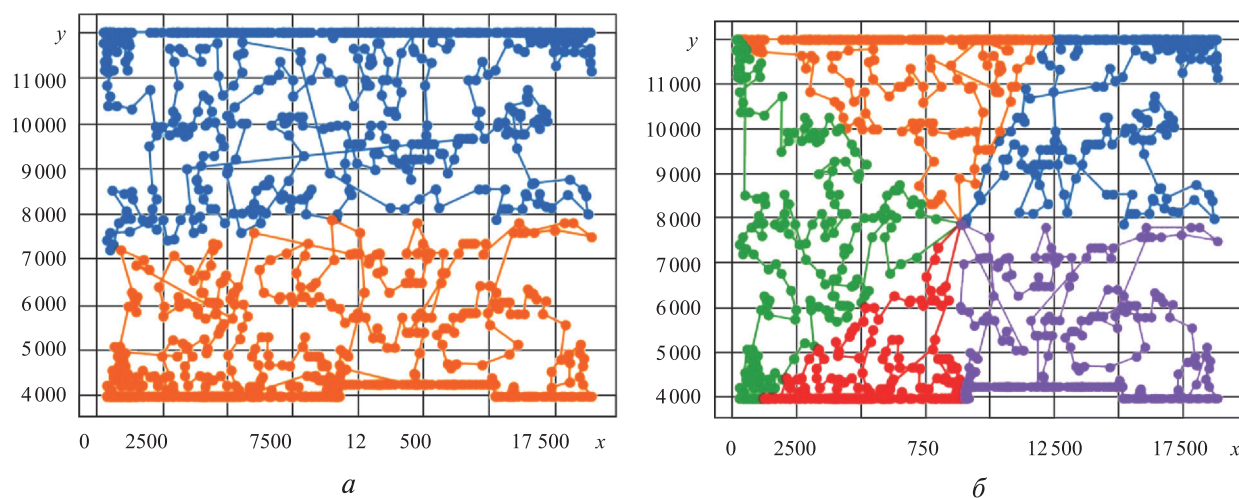


Рис. 2. Решения задачи MTSP для:
а — двух коммивояжеров на 1084 вершинах; *б* — пяти коммивояжеров на 1084 вершинах



Рассмотрим более подробно применяемые в работе (наиболее известные) приближенные алгоритмы для решения задач маршрутизации: алгоритм муравьиной колонии, алгоритм пчелиного роя, метод имитации отжига. Алгоритм муравьиной колонии [22] основан на поведении муравьев в муравьиной колонии. Здесь муравьиная колония представляет собой многоагентную систему, в которой каждая отдельная особь или агент выполняет набор простых операций и в одиночку не способен обеспечить глобальное решение поставленной задачи. Сложность данного алгоритма зависит от времени жизни колонии $t_{\max}$, числа вершин n и числа муравьев в колонии m . Рассмотрим классический алгоритм муравьиной колонии.

1. Инициализируем граф G (например, вводим матрицу расстояний D).

2. Задаем параметры α , β , Q , $t_{\max}$, m .

3. Задаем значение маркера для каждого ребра $\delta_{ij} \in G$ (феромон).

4. Помещаем муравьев в произвольно выбранные вершины, без повторов.

5. Задаем произвольный кратчайший маршрут T^* и его длину L^* .

6. Создаем цикл по времени жизни колонии от $t = 1$ до $t_{\max}$.

6.1. Создаем цикл обхода всех муравьев колонии от $k = 1$ до $k = m$. Строим путь $T_k(t)$ и рассчитываем длину $L_k(t)$. Вероятность перехода муравья k в город i из города j определяется формулой

$$P_{ij,k} = \begin{cases} \tau_{ij}^\alpha(t) \eta_{ij}^\beta(t) / \sum_{j \in J_{ik}} \tau_{ij}^\alpha(t) \eta_{ij}^\beta(t), \\ 0, \end{cases}$$

где $j \in J_{ik}$, α и β задают значимость уровня феромона и видимости города при выборе следующего города. При $\alpha = 0$ будет выбран ближайший город, что соответствует жадному алгоритму. Если $\beta = 0$, тогда работает лишь феромонное усиление, что влечет за собой быстрое вырождение маршрутов к одному субоптимальному решению.

6.2. Конец цикла обхода всех муравьев.

6.3. Проверка каждого полученного значения $L_k(t)$ на лучший маршрут по сравнению с L^* .

6.4. Если $L_k(t)$ лучше, чем L^* , обновить L^* и T^* .

6.5. Создаем цикл обхода всех ребер графа. Обновляем $\tau_{i,j}$ согласно правилам:

$$\Delta\tau_{ij,k}(t) = \begin{cases} \frac{Q}{L_k(t)}, (i,j) \in T_k(t), \\ 0; \end{cases}$$

где τ_{ij} — количество феромона на ребре (i,j) ; $\eta_{i,j} = 1/D_{ij}$, где D_{ij} — длина ребра (i,j) , J_{ik} — список городов, которые необходимо посетить муравью k , находящемуся в вершине i . Здесь $T_k(t)$ — маршрут, пройденный муравьем k на итерации t , $L_k(t)$ — длина этого маршрута, Q — некоторый регулируемый параметр. Правило обновления феромона:

$$\tau_{ij,k}(t+1) = (1-p)\tau_{ij,k}(t) + \Delta\tau_{ij,k}(t),$$

где $\Delta\tau_{ij,k}(t) = \sum_{k=1}^m \Delta\tau_{ij,k}(t)$, m — число муравьев, p —

регулируемый параметр испарения феромона, принадлежащий отрезку $[0, 1]$.

6.6. Конец цикла по ребрам.

7. Конец цикла по времени жизни колонии.

8. Передать на вывод кратчайший маршрут T^* и его длину L^* .

Для рассмотрения алгоритма пчелиного роя [23, 24] введем обозначения: s — число пчел-разведчиков; m — число выбранных точек (решений) для дальнейшего исследования, $m < s$; e — число лучших (элитных) точек, $e < m$; s_e — число пчел для более полного исследования e элитных решений; s_m — число пчел для более полного исследования оставшихся $m - e$ выбранных решений; Δ — размер окрестности, в которой пчелы выполняют более тщательный поиск. Алгоритм пчелиного роя состоит в выполнении следующих шагов.

1. Случайно формируется s решений, каждое из которых представляет собой одну пчелу-разведчика.

2. Выбрать лучшие m из этих решений и провести локальный поиск, для этого в их Δ -окрестности рассматриваются r случайных точек ($r = s_e$ или $r = s_m$ в зависимости от элитности данной точки), из которых отбирается лучшая.

3. Оставшиеся $s - m$ решений отбрасываются и заменяются на случайные точки из пространства решений.

4. Алгоритм останавливается, когда выполняется некое условие останова (достигнута необходимая точность, исчерпано число итераций и др.).

Метод имитации отжига [25] основан на идее поведения материального тела при отвердевании, когда применяется процедура отжига, во время которой температура последовательно понижается до нуля. В ходе «отжига» металл сначала нагревают до некоторой температуры, из-за чего атомы покидают позиции в кристаллической решетке. Затем начинается медленное и контролируемое охлаждение. Атомы стремятся попасть в состояние с меньшей энергией, однако с определенной веро-

ятностью они могут перейти и в состояние с большей. Эта вероятность уменьшается вместе с температурой. Переход в худшее состояние помогает атомам отыскать состояние с энергией, меньшей, чем начальная. Процесс завершается, когда температура падает до заранее заданного значения. В задаче ДО минимизация энергии представляется как целевая функция, где S — множество всех состояний — решений задачи, s_i — состояние на i -м шаге, t_i — температура на i -м шаге. Алгоритм имитаций отжига формулируется так.

1. Инициализируем граф G (например, вводим матрицу расстояний D).

2. Задаем параметры T_s и T_f .

3. Задаем произвольное начальное состояние S и считаем энергию для этого состояния E по формуле

$$E = \sum_{i \in S, i \neq n} w(i, i+1) + w(n, 0),$$

где $w(i, j)$ — расстояние от вершины i до вершины j .

4. Создаем лучшее состояние S^* с энергией E^* .

5. Инициализируем цикл «остывания» до тех пор, пока $T > T_f$.

5.1. Случайным образом генерируем новое состояние-кандидат CS , считаем его энергию CE .

5.2. Сравниваем CE с E .

5.2.1. Если $CE > E$, то переходим в состояние-кандидат и делаем его текущим, т. е. $E = CE$; $S = CS$.

5.2.2. Если $CE < E$, то переходим в это состояние с вероятностью P , вычисленной по формуле $P_i = \exp(-\Delta E/T_i)$ где $\Delta E = CE - E$.

5.3. Сравниваем энергию текущего состояния E с энергией лучшего состояния E^* . Если $E < E^*$, то заменяем лучшее состояние на текущее, т. е. $E^* = E$; $S^* = S$.

5.4. Изменяем T по правилу $T_{i+1} = 0,1 T_i/i$, где $T_0 = T_s$.

6. Конец цикла «остывания».

7. Выводим значение лучшей энергии E^* и соответствующего состояния S^* .

В алгоритме приняты обозначения: T_s — начальная температура; T_f — конечная температура; предполагается, что граф G полносвязный.

4. МОДИФИКАЦИЯ ГЕНЕТИЧЕСКОГО АЛГОРИТМА

Генетический алгоритм основан на идее естественного отбора. В общем виде он формулируется следующим образом.

1. Создание начальной популяции.

2. Скрещивание и/или мутация.

3. Отбор.

4. Формирование нового поколения.

4.1. Если результат не достигнут — переход к п. 2.

4.2. В противном случае текущая популяция является результирующей.

Рассмотрим модификацию генетического алгоритма для решения многоагентной задачи коммивояжера, в котором исходный граф разбивается на m кластеров и для каждого кластера решается обычная задача коммивояжера. Предложенный подход включает создание популяции из хромосом длины $n + m$, где n вершин представлены в виде перестановки чисел от 1 до n . Перестановка поделена на m подмаршрутов при помощи добавления m положительных чисел (от 1 до m) для отличия одного коммивояжера от другого. Пусть m_i отвечает числу вершин, пройденных i -м коммивояжером, тогда, согласно условию задачи, $K \leq m_i \leq L$.

Один из лучших кроссоверов в смысле скорости и качества — это двухточечный кроссовер. Предлагаемая модификация кроссовера состоит в том, что случайно выбранные точки разделяют родительские строки на левые и правые подстроки и выбираются родительские правые — правила выбора подстрок. Далее процесс такой же, как и для двухточечного кроссовера, отличие состоит в том, что вместо выбора случайных нескольких позиций в родительском маршруте выбраны все позиции справа от случайно выбранной точки пересечения. В данном алгоритме может быть использован один из двух видов мутации: либо оператор выбирает две случайные точки (разрезающие точки) в строке и меняет их местами, либо реализуется возможность реверсирования всех точек, которые находятся между разрезающими точками. Данный подход обуславливает нахождение лучшего решения по сравнению с применением генетического, «жадного» или алгоритма муравьиной колонии для решения обычной задачи коммивояжера для каждого кластера графа, как следует из статьи [26]. В работе [27] показано, что снижение размерности задачи маршрутизации в прикладной теории сетей связывается с предобработкой доступных данных о сети, а также с использованием особенностей взвешенных графов для более быстрого определения их характеристик (центр, радиус, диаметр и др.).

Утверждение 2. *Снижение сложности вычислений в построении рациональных приближенных решений задач MTSP (многих коммивояжеров) на сложных сетях большой размерности может быть достигнуто по схеме:*

— на первом этапе решается задача распределения сети между коммивояжерами с помощью кластеризации (декомпозиции);

— на втором этапе решаются задачи коммивояжера на каждом кластере с помощью метаэвристики;

— в зависимости от полученного результата уточняются границы используемых кластеров;

— в дальнейшем агенты-коммивояжеры могут обучаться и быть автономными.

Пусть найдено разбиение исходного графа сети на подграфы, и задача построения маршрута коммивояжера решается на сети меньшей размерности. В качестве исходных данных выбрана база тестовых примеров для задачи коммивояжера, взятых из Национальных агентств изображений и картографии разных стран². Каждый блок данных содержит в себе список двумерных координат. С помощью этого списка можно построить решение своими методами и сравнить с оптимальным решением, которое также входит в состав тестового примера. На первом этапе было выбрано несколько тестовых примеров графов небольшой размерности, соответствующих кластеру сложной сети — dj38, eil51 и qa194, d493, vm1084 с 38, 51 и 194, 493, 1084 вершинами соответственно. Затем проведен численный эксперимент с помощью реализованных алгоритмов.

Результаты приведены в таблице. Предварительно получено, что алгоритм муравьиной колонии нуждается в доработке, в то время как алгоритм имитации отжига дает хорошее приближение к оптимальному решению. Результаты по гибридным алгоритмам показывают возможность оптимизационной комбинации базовых алгоритмов.

Для иллюстрации на рис. 2 (см. третью стр. обложки) приведены решения задачи MTSP для двух и пяти коммивояжеров. Для сети из 1084 вершин начальное приближение, кластеризация и улучшение решения аналогичны представленным на рис. 1.

Подчеркнем, что в рамках разрабатываемой интеллектуализированной МАС, которая ориентирована на кластеризацию задачи (ее декомпозицию) и дальнейшее распараллеливание, обладание более точной дополнительной информацией о сети и задаче позволит подключать композицию алгоритмов, наиболее подходящих для конкретного случая.

В реализации алгоритмов использовались возможности объектно-ориентированного программирования, которые представляет Python. В реализации используется модуль tslib95³ для импорта данных задачи коммивояжера из файлов. Эта библиотека содержит множество функций по предобработке данных [4]. Эксперимент с параллельными вычислениями на данном этапе показал, что

Работа приближенных алгоритмов

Алгоритм	Граф					
	dj38		eil51		qa194	
	Длина пути	Время, мс	Длина пути	Время, мс	Длина пути	Время, мс
Муравьиный	7451	223	572	328	11 657	23 147
Генетический	7895	156	559	117	1164	10 938
Гибридный	7115	263	498	448	29 234	10 506
Имитация отжига	7158	15	441	20	10 506	69
Пчелиной колонии	6656	21	439	26	11 695	645

однопоточный вариант с ускорением Numba требует меньших затрат времени, чем модификация с параллельными вычислениями на потоках Python. Необходимо распараллеливание, которое предложено в работе [28] и использовалось в решении задач ДО.

ЗАКЛЮЧЕНИЕ

Рассмотрен класс задач маршрутизации — многоагентные задачи. Исследованы современные методы для их решения. Установлены лучшие алгоритмы для разных типов задач среди метаэвристических алгоритмов, основанных на реальных моделях. Модификация ГА, муравьиной колонии, пчелиного роя, имитации отжига для MTSP зависит от сложности сети. Постановка MTSP в виде задачи псевдобулевой оптимизации позволяет включать различные ограничения и предписания в виде ДНФ. На примере задачи маршрутизации в сложных условиях для Ялты показана реализация методологии, основанной на преобразовании сложной сети к сети, для которой уже применимы быстрые алгоритмы (например, полиномиальной сложности). Управляемое и согласованное с MTSP разбиение сети может применяться в МАС MTSP. Показана эффективность методов решения задачи многоагентной задачи коммивояжера, однако для решения разреженных задач маршрутизации представляет интерес распараллеливание, как это показано в статье [28]. В дальнейшем возможно распространение полученных результатов на нелинейные постановки по аналогии с работами [29, 30].

Авторы выражают признательность рецензенту за замечания и рекомендации, которые привели к повышению качества представления материалов исследования.

² <http://www.math.uwaterloo.ca/tsp/world/countries.html>.

³ TSPLB 95 documentation. <https://tsplib95.readthedocs.io/en/stable/index.htm>.

ЛИТЕРАТУРА

1. Шербина О.А., Лемтюжникова Д.В., Цурков В.И. Многомерные задачи с квазиблочными матрицами. — М.: Физматлит, 2018. — 256 с. [Shcherbina, O.A., Lemtyuzhnikova, D.V., Tsurkov, V.I. *Mnogomernye zadachi s kvaziblochnymi matritsami*. — М.: Fizmatlit, 2018. — 256 s. (In Russian)]
2. Kovkov, D.V., Lemtyuzhnikova, D.V. Decomposition in Multi-dimensional Boolean-Optimization Problems with Sparse Matrices / J. Computer and Systems Sciences International. — 2018. — Vol. 57, no. 1. — P. 97–108.
3. Tsurkov, V.I. Decomposition Principle for Block-separable Systems // Dokl. Akad. Nauk SSSR, 1979. — Vol. 246, no. 1. — P. 27–31.
4. Германчук М.С. Использование дополнительной информации в задачах дискретной оптимизации типа многих коммивояжеров // Таврический вестник информатики и математики. — 2016. — № 4 (33). — С. 68–82. [Germanchuk, M.S. *Ispol'zovanie dopolnitel'noi informatsii v zadachakh diskretnoi optimizatsii tipa mnogikh kommivoyazherov* // Tavrisheskii vestnik informatiki i matematiki. — 2016. — № 4 (33). — S. 68–82. (In Russian)]
5. Шербина О.А. Метаэвристические алгоритмы для задач комбинаторной оптимизации (обзор) / Таврический вестник информатики и математики. — 2014. — № 1. — С. 56–73. [Shcherbina, O.A. *Metaevristicheskie algoritmy dlya zadach kombinatornoj optimizatsii (obzor)* / Tavrisheskii vestnik informatiki i matematiki. — 2014. — № 1. — S. 56–73. (In Russian)]
6. Германчук М.С., Козлова М.Г., Лукьяненко В.А. Задачи дискретной оптимизации типа многих коммивояжеров // Математические методы распознавания образов: Тезисы докладов 18-й Всероссийской конференции с международным участием. — Таганрог, 2017. — М.: TOPYC ПРЕСС, 2017. — С. 48. [Germanchuk, M., Kozlova, M., Lukianenko, V. *The Problems of Discrete Optimization Such as Multiple Traveling Salesman Problem* // Matematicheskie metody raspoznavaniya obrazov: Tezisy dokladov 18-i Vserossiiskoi konferentsii s mezhduнародnym uchastiem. — Taganrog, 2017. — М.: TORUS PRESS, 2017. — S. 48. (In Russian)]
7. Германчук М.С., Козлова М.Г., Пивовар А.Е. Многоагентный подход к решению задачи коммивояжера // Математика, информатика, компьютерные науки, моделирование, образование: сборник научных трудов научно-практической конференции МИКМО—2017 и Таврической научной конференции студентов и молодых специалистов по математике и информатике / Под ред. В.А. Лукьяненко. — Симферополь: ИП Корниенко А.А., 2017. — С. 114–119. [Germanchuk, M.S., Kozlova, M.G., Pivovarov, A.E. *Solving the Multiple Traveling Salesman Problem* // Matematika, informatika, komp'yuternye nauki, modelirovanie, obrazovanie: sbornik nauchnykh trudov nauchno-prakticheskoi konferentsii MIKMO—2017 i Tavrisheskoi nauchnoi konferentsii studentov i molodykh spetsialistov po matematike i informatike / Pod red. V.A. Luk'yanenko. — Simferopol': IP Kornienko, A.A., 2017. — S. 114–119. (In Russian)]
8. Poli, R. Analysis of the Publications of the Applications of Particle Swarm Optimization // Hindawi Publishing Corporation Journal of Artificial Evolution and Applications. — URL: <https://www.hindawi.com/journals/jaea/2008/685175/> <https://doi.org/10.1155/2008/685175>.
9. Jungsbluth, M., Thiele, J., Winter, Y., et al. Vertebrate Pollinators: Phase Transition in a Time-Dependent Generalized Traveling-Salesperson Problem // arXiv preprint arXiv: 1803.08015. 2018.
10. Shah-Hosseini, H. Intelligent Water Drops Algorithm: A New Optimization Method for Solving the Multiple Knapsack Problem // International Journal of Intelligent Computing and Cybernetics. — 2008. — Vol. 1, no. 2. — P. 193–212. URL: <https://www.emerald.com/insight/content/doi/10.1108/1756378081087417/full/html>.
11. Zhou, H., Song, M., Pedrycz, W. A Comparative Study of Improved GA and PSO in Solving Multiple Traveling Salesmen Problem // Applied Soft Computing. — 2018. — Vol. 64. — P. 564–580.
12. Singh, D.R., Singh, M.K., Singh, T., et al. Genetic Algorithm for Solving Multiple Traveling Salesmen Problem Using a New Crossover and Population Generation // Computación y Sistemas. — 2018. — Vol. 22, no. 2. — P. 491–503.
13. Venkatesh, P., Singh, A. Two Metaheuristic Approaches for the Multiple Traveling Salesperson Problem // Applied Soft Computing. — 2015. — Vol. 26. — P. 74–89.
14. Lo, K.M., Yi, W.Y., Wong, P.K., et al. A Genetic Algorithm with New Local Operators for Multiple Traveling Salesman Problems // International J. Computational Intelligence Systems. — 2018. — Vol. 11, no. 1. — P. 692–705.
15. Harrath, Y., Salman, A.F., Alqaddoumi, A., et al. A Novel Hybrid Approach for Solving the Multiple Traveling Salesmen Problem // Arab J. Basic and Applied Sciences. — 2019. — Vol. 26, no. 1. — P. 103–112.
16. Shokouhi, R.A., Farahnaz, M., Hengameh, K., Hosseinabadi, A.R. Solving Multiple Traveling Salesman Problem using the Gravitational Emulation Local Search Algorithm // Applied Mathematics & Information Sciences. — 2015. — Vol. 9, no. 2. — P. 699–709.
17. Huizing, D. Solving the mTSP for fresh food delivery / Report on behalf of the Delft Institute of Applied Mathematics. — Delft, Netherlands, 2015. — URL: <https://repository.tudelft.nl/islandora/object/uuid%3A8af405cc-bdd1-46c0-a790-a66471leadb3f>.
18. Othman, A., Mouhssine, R., Ezziyyani, M., et al. An Effective Parallel Approach to Solve Multiple Traveling Salesmen Problem // Intern. Conf. on Advanced Intelligent Systems for Sustainable Development. — Cham: Springer, 2018. — P. 647–664.
19. Necula, R., Raschip, M., Breaban, M. Balancing the Subtours for Multiple TSP Approached with ACS: Clustering-based Approaches vs. MinMax Formulation // EVOLVE-A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation VI. — Cham: Springer, 2018. — P. 210–223.
20. Kara, I., Bektas, T. Integer Linear Programming Formulations of Multiple Salesman Problems and its Variations // European J. Operational Research. — 2006. — Vol. 174. — P. 1449–1458.
21. Макаров О.О., Германчук М.С. Разработка алгоритмов маршрутизации в сложных сетях // Математика, информатика, компьютерные науки, моделирование, образование: сборник научных трудов научно-практической конференции МИКМО—2018 и Таврической научной конференции студентов и молодых специалистов по математике и информатике / Под ред. В.А. Лукьяненко. — Симферополь: ИП Корниенко А.А., 2018. — Вып. 2. — С. 127–135. [Makarov, O.O., Germanchuk, M.S. *Razrabotka algoritmov marshrutizatsii v slozhnykh setyakh* // Matematika, informatika, komp'yuternye nauki, modelirovanie, obrazovanie: sbornik nauchnykh trudov nauchno-prakticheskoi konferentsii MIKMO—2018 i Tavrisheskoi nauchnoi konferentsii studentov i molodykh spetsialistov po matematike i informatike / Pod red. V.A. Luk'yanenko. — Simferopol': IP Kornienko A.A., 2018. — Vyp. 2. — S. 127–135. (In Russian)]
22. Dorigo, M., Stutzle, T. Ant Colony Optimization. — Cambridge: Bradford Book. — 2004.
23. Зайцев А.А., Курейчик В.В., Полупанов А.А. Обзор эволюционных методов оптимизации на основе роевого интеллекта // Известия Южного федерального университета. Технические науки. — 2010. — № 12 (113). — С. 7–12. [Zaitsev, A.A., Kureychik, V.V., Polupanov, A.A. *Review of evolutionary optimization methods based on swarm intelligence* // Proceedings of the southern Federal University. Technical Sciences. — 2010. — № 12 (113). — P. 7–12. (In Russian)]
24. Karaboga, D., Akay, B. A Survey: Algorithms Simulating Bee Swarm Intelligence // Artificial Intelligence Review. — 2009. — Vol. 31, no. 1–4. — P. 61–85.
25. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P. Optimization by Simulated Annealing // Science. — 1983. — Vol. 220, no. 4598. — P. 671–680.
26. Carter, A.E., Ragsdale, C.T. A New Approach to Solving the Multiple Traveling Salesperson Problem Using Genetic Algo-



- rithms // *Europ. J. Operational Research.* — 2006. — Vol. 175. — P. 246–257.
27. Ураков А.Р., Тимеряев Т.В. Использование особенностей взвешенных графов для более быстрого определения их характеристик // *Прикладная Дискретная Математика.* — 2012. — № 2 (16). — С. 95–99. [Urakov, A.R., Timeryaev, T.V. Using Weighted Graphs Features for Fast Searching Their Parameters // *Prikladnaya Diskretnaya Matematika.* — 2012. — No. 2 (16). — P. 95–99. (In Russian)]
28. Voloshinov, V.V., Lemtyuzhnikova, D.V., Tsurkov, V.I. Grid Parallelizing of Discrete Optimization Problems with Quasi-Block Structure Matrices / *J. Computer and Systems Sciences International.* — 2017. — Vol. 56, no. 6. — P. 930–936.
29. Миронов А.А., Федорчук В.В., Цурков В.И. Минимакс в моделях транспортного типа с интегральными ограничениями. II // *Изв. РАН. ТиСУ.* — 2005. — No. 5. — С. 66–86. [Mironov, A.A., Fedorchuk, V.V., Tsurkov, V.I. Minimax in Transportation Models with Integral Constraints: II // *Journal of Computer and Systems Sciences International.* — 2005. — Vol. 44, no 5. — P. 732–752.]
30. Миронов А.А., Цурков В.И. Минимакс в моделях транспортного типа с интегральными ограничениями. I // *Изв. РАН. ТиСУ.* — 2003. — No. 4. — С. 69–81. [Mironov, A.A., Tsurkov, V.I. Minimax in Transportation Models with Integral Constraints: I // *Journal of Computer and Systems Sciences International.* — 2003. — Vol. 42, no 4. — P. 562–574.]
- Статья представлена к публикации членом редколлегии В.Н. Бурковым.
- Поступила в редакцию 29.01.2020, после доработки 4.08.2020.
Принята к публикации 4.09.2020.
- Германчук Мария Сергеевна** — ассистент, Крымский Федеральный Университет им. В.И. Вернадского, г. Симферополь, ✉ m.german4uk@yandex.ru,
- Лемтюжникова Дарья Владимировна** — канд. физ.-мат. наук, Институт проблем управления им. В.А. Трапезникова РАН, г. Москва; доцент, Московский авиационный институт, ✉ darabbbt@gmail.com,
- Лукьяненко Владимир Андреевич** — канд. физ.-мат. наук, Крымский Федеральный Университет им. В.И. Вернадского, г. Симферополь, ✉ art-inf@yandex.ru.

METAHEURISTIC ALGORITHMS FOR MULTI-AGENT ROUTING PROBLEMS

M.S. Germanchuk¹, D.V. Lemtyuzhnikova^{2,3}, V.A. Lukianenko¹

¹ V.I. Vernadsky Crimean Federal University, Simferopol, Republic of Crimea

² V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, Moscow Russia

³ Moscow Aviation Institute

✉ m.german4uk@yandex.ru, ✉ darabbbt@gmail.com, ✉ art-inf@yandex.ru

Abstract. The problems of constructing routes in complex networks by many sales agents are considered. Formalization leads to problems of pseudo-Boolean discrete optimization with restrictions that take into account the specifics of route construction. The sparsity of the constraint matrix makes it possible to apply decomposition approaches and network clustering. The development of approximate algorithms for selecting routes in complex networks involves taking into account the properties of the network structure, its complexity, the presence of restrictions, regulations, reachability conditions, and the number of sales agents. It is shown that the solution of routing problems can be based on the application of a multi-agent approach in combination with clustering (decomposition) of the original problem and metaheuristics. Multi-agent systems with swarm intelligence are used to solve complex discrete optimization problems that cannot be effectively solved by classical algorithms. The agent model for a complex network of problems like many traveling salesmen becomes an intellectualized system that defines heuristic algorithms for finding the optimal solution by reactive agents (that follow the rules laid down in them). The compositions of the algorithms described in detail, which have proven themselves well in computational experiments, are used; those are modification of the genetic algorithm, ant colony optimization, artificial bee colony algorithm, simulated annealing. A generalized algorithm is proposed and implemented, in which a simpler network (a flyover network) is matched to the source network. In this case, a numerical experiment was performed for the problem of routing on a GIS map for urban infrastructure. Clustering algorithms are implemented, in which the initially traversed routes are refined using 2-opt algorithms, simulated annealing, and other metaheuristics. A comparison of the algorithms used and an illustration of their operation are given.

Keywords: metaheuristic algorithms, multi-agent optimization problems, discrete optimization, pseudo-Boolean problems.

Funding. The work was performed with financial support of the Russian Foundation of Basic Research (grant no. 18-31-00458, no. 20-58-S52006).

Acknowledgments. The authors express their appreciation to the reviewer for his comments and recommendations which led to a better quality of representing the research contents.