doi: 10.17586/2226-1494-2025-25-5-856-865
УДК 004.421.2

# Методы роевой оптимизации частиц и локальных эвристик для решения мультиагентной задачи коммивояжёра

**Эльдар Наилевич Мифтахов[1], Андрей Анатольевич Акимов[2], Юлия Ахнафовна Гнатенко[3]✉**

[1,2] МИРЭА — Российский технологический университет, Москва, 119454, Российская Федерация
[3] Стерлитамакский филиал Уфимского университета науки и технологий, Стерлитамак, 453103, Российская Федерация

[1] promif@mail.ru, https://orcid.org/0000-0002-0471-5949
[2] andakm@yandex.ru, https://orcid.org/0000-0003-3387-2959
[3] y.a.gnatenko@struust.ru✉, https://orcid.org/0009-0009-9264-3989

**Аннотация**
**Введение.** Представлены результаты разработки и апробации метода решения мультиагентной задачи коммивояжёра (Multiple Traveling Salesman Problem, mTSP) с целью минимизации максимальной длины маршрутов («минимаксная» оптимизация). Объектом исследования является комбинаторное пространство маршрутов, возникающее при распределении городов между несколькими агентами, что обуславливает необходимость равномерного распределения нагрузки и предотвращения перегрузки отдельных маршрутов. Новизна подхода заключается в создании дискретного аналога классического алгоритма роевой оптимизации частиц (Particle Swarm Optimization, PSO), адаптированного для работы с перестановками, а также в интеграции его с локальными эвристическими процедурами и муравьиным алгоритмом (Ant Colony Optimization, ACO). **Метод.** Предложенный метод базируется на преобразовании исходной задачи mTSP в классическую задачу коммивояжёра для одного агента (TSP) посредством введения фиктивных депо, что позволяет однозначно разделить общий маршрут на отдельные части для каждого агента. Ключевым элементом является модификация PSO с использованием новых операций для дискретного пространства, таких как вычисление минимальной последовательности обменов (транспозиций) между перестановками, масштабирование скорости и применение ее к маршруту. Данный подход позволяет эффективно исследовать комбинаторное пространство решений и предотвращать преждевременную сходимость алгоритма. **Основные результаты.** Экспериментальное исследование проведено на тестовых наборах стандартной библиотеки TSPLIB (eil51.tsp, berlin52.tsp, eil76.tsp, rat99.tsp) для задачи TSP, в ходе которого сравнивались два сценария: классический PSO со случайной инициализацией и гибридный метод PSO_ACO, где метод ACO используется для формирования начальной популяции. Результаты эксперимента продемонстрировали существенное улучшение по «минимаксному» критерию по сравнению с методами CPLEX, LKH3, OR-Tools, а также современными подходами DAN, NCE и EA, что подтверждает эффективность предложенного решения. **Обсуждение.** Разработанный алгоритм может найти применение в логистике, транспортном планировании, распределении потоков в сетях связи и иных областях, где требуется оптимальное распределение ресурсов. Представленный метод будет полезен специалистам в области оптимизации, алгоритмического моделирования и практикам, занимающимся разработкой систем управления и планирования.

**Ключевые слова**
роевый алгоритм оптимизации частиц, мультиагентная задача коммивояжёра, минимаксная оптимизация, дискретная оптимизация, муравьиный алгоритм, локальные эвристики

**Ссылка для цитирования:** Мифтахов Э.Н., Акимов А.А., Гнатенко Ю.А. Методы роевой оптимизации частиц и локальных эвристик для решения мультиагентной задачи коммивояжёра // Научно-технический вестник информационных технологий, механики и оптики. 2025. Т. 25, № 5. С. 856–865. doi: 10.17586/2226-1494-2025-25-5-856-865

# Particle swarm optimization methods and local heuristics for solving the multiple traveling salesman problem

**Eldar N. Miftakhov[1], Andrey A. Akimov[2], Yuliya A. Gnatenko[3]✉**

[1,2] MIREA — Russian Technological University, Moscow, 119454, Russian Federation
[3] Branch of the Ufa University of Science and Technology, Sterlitamak, 453103, Russian Federation

[1] promif@mail.ru, https://orcid.org/0000-0002-0471-5949
[2] andakm@yandex.ru, https://orcid.org/0000-0003-3387-2959
[3] y.a.gnatenko@struust.ru✉, https://orcid.org/0009-0009-9264-3989

**Abstract**

This paper presents the development and evaluation of a method for solving the Multiple Traveling Salesman Problem (mTSP), with the objective of minimizing the maximum route length (“minimax” optimization). The study addresses the combinatorial route-space arising from distributing cities among multiple agents, requiring balanced workload distribution to avoid overloading individual routes. The novelty of the proposed approach lies in creating a discrete analogue of the classical Particle Swarm Optimization (PSO) algorithm adapted specifically for permutation-based route representations, and integrating it with local heuristic procedures and the Ant Colony Optimization (ACO) algorithm. The proposed method transforms the original mTSP into a classical single-agent Traveling Salesman Problem (TSP) by introducing artificial (dummy) depots, thus allowing an unambiguous separation of the overall route into individual segments for each agent. A key element of the solution involves adapting the PSO algorithm through novel discrete operations, such as computing the minimal sequence of exchanges (transpositions) between permutations, scaling velocity, and applying this velocity to routes. This approach ensures efficient exploration of the combinatorial solution space and prevents premature convergence of the algorithm. The experimental study was conducted on benchmark instances from the TSPLIB library (eil51.tsp, berlin52.tsp, eil76.tsp, rat99.tsp) for the TSP, comparing two scenarios: a classical PSO with random initialization and a hybrid PSO_ACO method where the ACO algorithm is used to generate the initial population. The results demonstrate a significant improvement in the minimax criterion compared to CPLEX, LKH3, OR-Tools as well as state-of-the-art approaches DAN, NCE, and EA, confirming the effectiveness of the proposed solution. The practical importance of this research lies in potential applications of the developed algorithm in logistics, transport planning, network traffic management, and other domains where optimal resource allocation is crucial. The proposed method is valuable for specialists in optimization, algorithmic modeling, and practitioners developing planning and management systems.

**Keywords**

particle swarm optimization, multiple traveling salesman problem, minimax optimization, discrete optimization, ant colony optimization, local heuristics

**For citation:** Miftakhov E.N., Akimov A.A., Gnatenko Yu.A. Particle swarm optimization methods and local heuristics for solving the multiple traveling salesman problem. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2025, vol. 25, no. 5, pp. 856–865 (in Russian). doi: 10.17586/2226-1494-2025-25-5-856-865

## Введение

Мультиагентная задача коммивояжёра (Multiple Traveling Salesman Problem, mTSP) широко применяется в логистике, транспортном планировании, распределении потоков связи и производственных процессах [1]. В классической постановке требуется, чтобы $m$ коммивояжёров (агентов), начинающих и заканчивающих маршрут из фиксированного депо, посетили $n$ городов ровно по одному разу с оптимизацией общей стоимости маршрутов (например, суммарного или максимального расстояния — критерий «минимакс») [1, 2]. «Минимаксная» постановка обеспечивает равномерное распределение нагрузки и предотвращает перегрузку отдельных маршрутов [3, 4].

Метод оптимизации роя частиц (Particle Swarm Optimization, PSO), изначально разработанный для непрерывных задач [5], получил множество дискретных модификаций, применимых к задачам маршрутизации [6]. Основная идея PSO состоит в том, что каждая частица (агент) перемещается в пространстве решений, опираясь на свое личное лучшее решение и коллективный опыт роя — лучшее решение, обнаруженное в глобальном или локальном окружении. Для адаптации PSO к mTSP необходимо переопределить понятия «позиция» и «скорость» в терминах перестановок, где маршрут представляется как упорядоченный список городов, а «псевдоскорость» — как набор операций, изменяющих порядок обхода или распределение городов между агентами. Дополнительная интеграция алгоритма с локальными процедурами поиска (например, методом 2-opt) повышает точность решения и предотвращает преждевременную сходимость [7].

Таким образом, основная проблема стандартного PSO при решении mTSP заключается в переходе от непрерывного к дискретному представлению решений, что требует разработки корректного дискретного аналога PSO для эффективного исследования комбинаторного пространства маршрутов в условиях «минимаксной» оптимизации.

## Обзор тематических научных работ

Классическая задача коммивояжёра является NP-полной, что обуславливает применение эвристических и метаэвристических методов для поиска приближенных решений [2]. Среди них широко используются генетические алгоритмы [8, 9] и алгоритмы оптимизации колонии муравьев (Ant Colony Optimization, ACO) [5], демонстрирующие высокую эффективность при решении комбинаторных задач.

Отдельную группу методов решения задачи mTSP представляют нейросетевые способы, в том числе метод Decentralized Attention Network (DAN) [10], который сочетает графовую сеть внимания и локальный поиск, позволяя агентам кооперативно распределять города и тем самым минимизировать максимальную длину тура. Близкий по идее метод Neuro Cross-Exchange (NCE) использует графовую нейросеть для прогнозирования выгоды операций CROSS-exchange, сокращая сложность поиска с $O(n^4)$ до $O(n^2)$ без ущерба качеству решений [10].

Метод PSO вызывает значительный интерес, однако его применение осложнено дискретным характером mTSP. Поскольку стандартный PSO разработан для непрерывных пространств решений, его нельзя напрямую применять к mTSP. Для преодоления этой проблемы предложены различные методы дискретизации PSO. Например, в работе [11] разработано правило наименьшего значения позиции (Special Purpose Vehicle), позволяющее преобразовывать непрерывное представление в дискретное, а исследования в [3] показали, что включение генетических операторов (например, кроссовера и мутации) способствует улучшению поиска оптимальных решений в дискретном пространстве. Работа [12] направлена на переопределение базовых операций PSO (обновления скорости и положения частиц) для работы с перестановками, что снижает избыточность обновлений и предотвращает преждевременную сходимость.

Современные исследования все чаще фокусируются на гибридных алгоритмах, объединяющих PSO, ACO и другие эвристические подходы для решения mTSP с учетом критериев балансировки нагрузки (задача «минимакс») [4, 13]. Эффективная дискретизация PSO, позволяющая сохранять разнообразие решений и детально исследовать комбинаторное пространство маршрутов, является ключевым условием реализации таких гибридных методов.

Целью настоящей работы является разработка модифицированного алгоритма роевой оптимизации частиц, адаптированного для эффективного решения мультиагентной задачи коммивояжёра, позволяющего улучшить качество маршрутов по «минимаксному» критерию и предотвратить преждевременную сходимость за счет интеграции с локальными эвристиками и алгоритмом ACO.

## Методология

Одним из методов решения мультиагентной задачи коммивояжёра mTSP является преобразование ее в классическую задачу коммивояжёра TSP с одним агентом, для которой можно применить современные эвристические алгоритмы [14]. Все города вместе с депо представляются в виде графа, где вершина 1 (депо) служит общей стартовой и конечной точкой для всех агентов. Исходную задачу с $m$ агентами и $n$ городами сведем к TSP с $n + m - 1$ городами путем добавления $m - 1$ фиктивных депо, номера которых задаются по схеме:

$$1, k_2^{(1)}, k_3^{(1)}, \ldots, n+1, k_i^{(2)}, k_{i+1}^{(2)}, \ldots, n+2, \ldots, n+m-1, \\ k_j^{(m)}, k_{j+1}^{(m)}, \ldots, k_{n-1}^{(m)}, k_n^{(m)}. \quad (1)$$

Задача — минимизировать самый длинный путь из маршрутов всех коммивояжёров (задача «минимакс»).

Для решения полученной одноагентной задачи коммивояжёра в работе [8] предложен алгоритм PSO, основанный на моделировании поведения стаи птиц. Принцип алгоритма PSO состоит в том, что каждая «птица» в стае рассматривается как частица, обладающая механизмом «памяти», который помогает находить оптимальное решение путем взаимодействия с другими частицами в рое.

В стандартном PSO каждая частица рассматривается как точка без массы и объема, заданная в $N$-мерном пространстве. Позиции частиц являются потенциальными решениями задачи. Положение $j$-ой частицы представляется вектором $\mathbf{X}_j = (X_{j1}, X_{j2}, \ldots, X_{jn})$, а скорость полета частицы вектором $\mathbf{v}_j = (v_{j1}, v_{j2}, \ldots, v_{jn})$.

Алгоритм PSO является итерационным методом оптимизации. На каждой итерации частица $j$ последовательно обновляет свою скорость и положение:

$$\mathbf{v}_j^{i+1} = \omega \mathbf{v}_j^i + c_1 r_1 \cdot (\mathbf{P}_j - \mathbf{X}_j^i) + c_2 r_2 \cdot (\mathbf{G} - \mathbf{X}_j^i), \quad (2)$$

$$\mathbf{X}_j^{i+1} = \mathbf{X}_j^i + \mathbf{v}_j^{i+1}, \quad (3)$$

где $\mathbf{v}_j^i$ и $\mathbf{X}_j^i$ — скорость и положение частицы $j$ на $i$-ой итерации; $\mathbf{P}_j$ — лучшая позиция $j$-й частицы (личный опыт); $\mathbf{G}$ — лучшая позиция всего роя частиц (глобальный опыт); $\omega$ — коэффициент инерции скорости; $c_1$, $c_2$ — коэффициенты, определяющие влияние личного и коллективного опыта; $r_1$, $r_2$ — случайные числа из промежутка [0, 1].

Благодаря учету информации о ранее достигнутых «лучших» позициях (как индивидуальных, так и общих для всего роя), каждая частица в процессе итераций перемещается в те области пространства, где с наибольшей вероятностью находится (глобальное или локальное) оптимальное решение поставленной задачи.

## Применение метода PSO для решения задачи TSP

При решении задачи TSP с помощью алгоритма PSO положение частицы в многомерном пространстве определяет возможный маршрут — последовательность посещения городов (перестановок) вида (1). Скорость должна изменять положение частицы и представляется в виде последовательности обменов (транспозиций) городов.

Формулы (2), (3) основаны на стандартных операциях над векторами. В отличие от непрерывных векторных пространств, пространство перестановок не является линейным, и прямое сложение или вычитание таких последовательностей не дает корректного результата. Исходя из этого, вводятся новые операции.

1. $\alpha \cdot v$, $0 \leq \alpha \leq 1$ — скалярное умножение скорости (масштабирование вектора скорости), $\alpha \cdot v$ определяется как выбор первых $k = \lceil \alpha \cdot |v| \rceil$ преобразований (обменов) из последовательности $v$. Здесь $|v|$ — длина (число обменов) последовательности $v$. Округление вверх до ближайшего целого числа $\lceil \cdot \rceil$ гарантирует при любом $\alpha > 0$ выбор по крайней мере одного элемента из последовательности $v$, что обеспечивает ненулевой результат.

Если $\alpha > 1$, то $k = \lceil \alpha \cdot |v| \rceil > |v|$. В этом случае требуется взять элементов больше, чем доступно в $v$. В дискретных преобразованиях это не имеет смысла, поэтому случай $\alpha > 1$ не рассматривается.

Таким образом, $\alpha \cdot v$ — усеченная версия последовательности $v$, от которой берутся первые $k$ преобразований (обменов) пропорционально коэффициенту $\alpha$.

2. Операция $X \oplus v$ — приложение скорости к позиции. Пусть $X$ — позиция (перестановка городов), а $v = [s_1, s_2, \ldots, s_m]$ — скорость, т. е. последовательность обменов (транспозиций). Тогда $X \oplus v = s_m \circ s_{m-1} \circ \ldots \circ s_1(X)$. Иными словами, оператор $\oplus$ поочередно применяет все обмены из $v$ к перестановке $X$, начиная с $s_1$ и заканчивая $s_m$.

3. Операция $Y\Theta X$ — минимальная разность позиций $Y\Theta X = v_{\min}$, где $v_{\min}$ – минимальная по длине последовательность обменов, преобразующая $X$ в $Y$. Другими словами, $Y\Theta X$ — скорость (цепочка преобразований), применяя которую к $X$, получим $Y$.

В формуле (2) на текущей итерации учитывается влияние как личного опыта частицы $\mathbf{P}_j$, так и коллективного опыта роя $\mathbf{G}$. Однако в дискретном пространстве, где решения представлены в виде перестановок, попытки совместить два направления обновления часто приводят к тому, что результирующая последовательность обменов оказывается неинформативной или содержит избыточные операции. Может возникнуть ситуация, когда локальные минимумы частиц начнут слишком быстро сближаться с глобальным минимумом, что, в свою очередь, уменьшит разнообразие решений в рое и будет способствовать преждевременному «застреванию» алгоритма в локальном оптимуме.

Для решения этой проблемы предлагается вектор обновления скорости определить непосредственно через минимальную последовательность обменов, которая преобразует текущее решение в выбранное целевое (личное или глобальное). Если разница между текущим и личным минимумом оказывается меньше заданного порога $\varepsilon > 0$ (частица достигает стационарного состояния, «застревает»), применяется случайное возмущение, например, эвристический метод локального поиска 2-opt: решение улучшается путем инвертирования последовательности вершин между двумя случайно выбранными вершинами маршрута. В результате получается новый маршрут, позволяющий выйти из локального минимума. Это предотвращает полное совпадение локального и глобального минимумов, сохраняя разнообразие решений и обеспечивая способность алгоритма продолжать эффективное исследование поискового пространства.

С учетом введенных операций рассмотрим этапы дискретного аналога алгоритма PSO.

Этап 1. Инициализация:

1) генерация начальных решений: случайным образом сгенерировать начальные перестановки $\{X_j^0\}_{j=1}^N$ (маршруты коммивояжёров);
2) установка минимумов: для каждой частицы установить личное лучшее решение $P_j \leftarrow X_j^0$ и выбрать глобальное решение $G = \arg\min_j f(X_j^0)$, где $f(X)$ — целевая функция задачи mTSP (задача «минимакс»).

Этап 2. Основной цикл (итерации $i = 0, 1, \ldots, I - 1$). Для каждой частицы $j$ выполняется:

1) выбор целевого решения $T_j$:
$$T_j = \begin{cases} G, \text{ с вероятностью } p, \\ P_j, \text{ с вероятностью } 1 - p; \end{cases}$$
2) вычисление разности: рассчитывается минимальная последовательность обменов, переводящая $X_j^i$ в $T_j$: $v_j = T_j \Theta X_i^j$;
3) масштабирование разности (усечение): выбирается коэффициент $r$ в зависимости от цели:
— если $T_j = G$, то $r = r_1 \in (0, 1)$,
— если $T_j = P_j$, то $r = r_2 \in (0, 1]$,
тогда усеченная последовательность обменов:
$$v_j^{i+1} = r \cdot v_j = [s_1, s_2, \ldots, s_k],\ k = |r \cdot |v_j||;$$
4) обновление позиции частицы:
— если частица не находится в стационарном состоянии $|f(X_j^i) - f(P_j)| \geq \varepsilon$, то обновить положение по формуле $X_j^{i+1} = X_j^i \oplus (r \cdot (T_j \Theta X_j^i))$;
— если частица находится в стационарном состоянии $|f(X_j^i) - f(P_j)| < \varepsilon$, то применяется случайное возмущение $X_j^{i+1} = X_j^i \oplus R(X_j^i, k')$, где $R(X_j^i, k')$ — оператор случайного возмущения, генерирует случайную последовательность из $k'$ обменов;
5) обновление личного минимума: если $f(X_j^{i+1}) < f(P_j)$ и $|f(X_j^i) - f(P_j)| \geq \varepsilon$, то обновляется $P_j \leftarrow X_j^{i+1}$;
6) обновление глобального решения: после обработки всех частиц глобальное решение пересматривается $G = \arg\min_j f(X_j^{i+1})$.

Этап 3. Завершение. Алгоритм выполняется до достижения заданного количества итераций $I$ или до отсутствия улучшения решения в течение нескольких итераций. В конце возвращается глобальное решение $G$ как оптимальное найденное решение.

Таким образом, каждая частица на каждом этапе учитывает собственное лучшее найденное решение $P_j$ и глобальное лучшее решение $G$, а также вносит случайные корректировки, позволяющие «выходить» из возможных локальных минимумов в задаче минимизации максимального маршрута (метрика «минимакс»).

## Вычислительный эксперимент

Алгоритм PSO реализован на языке Python. Вычисления проводились на аппаратной платформе, оснащенной процессором 11-го поколения Intel® Core™ i7-11800H с тактовой частотой 2,30 ГГц. Использовалось 16 ядер процессора, что позволило задействовать 16 параллельных потоков для повышения производительности вычислений.

Для оценки абсолютной эффективности алгоритма PSO при решении задачи mTSP применялись тестовые наборы стандартной библиотеки TSPLIB: eil51.tsp, berlin52.tsp, eil76.tsp, rat99.tsp на 51, 52, 76 и 99 вершин соответственно [15].

Все эксперименты проводились при фиксированном наборе параметров. Размер популяции частиц

$N = 8 \cdot 10^5$; заданное количество частиц обеспечивает устойчивое покрытие пространства решений для наборов до 100 городов. При обновлении скоростей на текущей итерации использованы только «когнитивная» и «социальная» составляющие с равными коэффициентами $c_1 = c_2 = 0{,}5$. В предлагаемой модификации метода PSO инерционный вес не учитывался ($\omega = 0$), так как движение «по инерции» подразумевает сохранение прежнего «направления» (т. е. повторное применение той же цепочки преобразований к последовательности городов), что может привести к циклическому возвращению к уже посещенным перестановкам. Вместо инерционного слагаемого вводится случайное возмущение — короткая цепочка из двух независимых транспозиций (эвристический метод локального поиска 2-opt).

Рассматривалось два экспериментальных сценария. В сценарии 1 (алгоритм PSO_random) на этапе 1 случайным образом генерировались 800 тыс. начальных частиц (перестановок с добавлением фиктивных депо для разбиения маршрута между коммивояжёрами), затем в течение 600 итераций каждая частица улучшала свое решение согласно механизму PSO. В сценарии 2 (алгоритм PSO_ACO) сначала работал алгоритм ACO, на каждой итерации которого отбиралось 45 лучших решений, что в итоге привело к накоплению 800 тыс. маршрутов. Далее полученные маршруты использовались в качестве начального приближения для оптимизации методом PSO в течение 300 итераций.

Для алгоритма ACO применена классическая схема с изоляцией лучших особей. После генерации всех маршрутов феромон усиливался вдоль лучших 45 маршрутов, определяемых, так называемыми «элитными» муравьями, и одновременно испарялся с коэффициентом $\rho = 0{,}85$. Веса феромона и видимости выбраны $\alpha = 1{,}2$ и $\beta = 1{,}09$ соответственно. Эти значения параметров обеспечили наилучший баланс «поиск/эксплуатация» на предварительных прогонах.

На рис. 1 изображены графики динамики оптимизации для двух, трех, 5 и 7 агентов для набора rat99.tsp (каждый график в отдельной системе координат). Для остальных наборов и при различном количестве агентов наблюдаются аналогичные тенденции: на начальном этапе происходит резкое снижение целевой метрики, за которым следует фаза постепенного уменьшения. Варианты с бóльшим количеством агентов демонстрируют более низкое итоговое значение максимальной длины маршрута (рис. 2).

Алгоритм PSO имеет вероятностный характер, что проявляется в снижении темпов улучшения решения по мере увеличения числа итераций. Начальные итерации обеспечивают значительное улучшение, однако с течением времени прирост оптимизации становится менее выраженным. При сравнении экспериментов с двумя тремя и с 5–7 агентами видно, что лучшие показатели минимаксного критерия достигаются при большем числе агентов. Это может быть связано с более равномерным распределением нагрузки между маршрутами

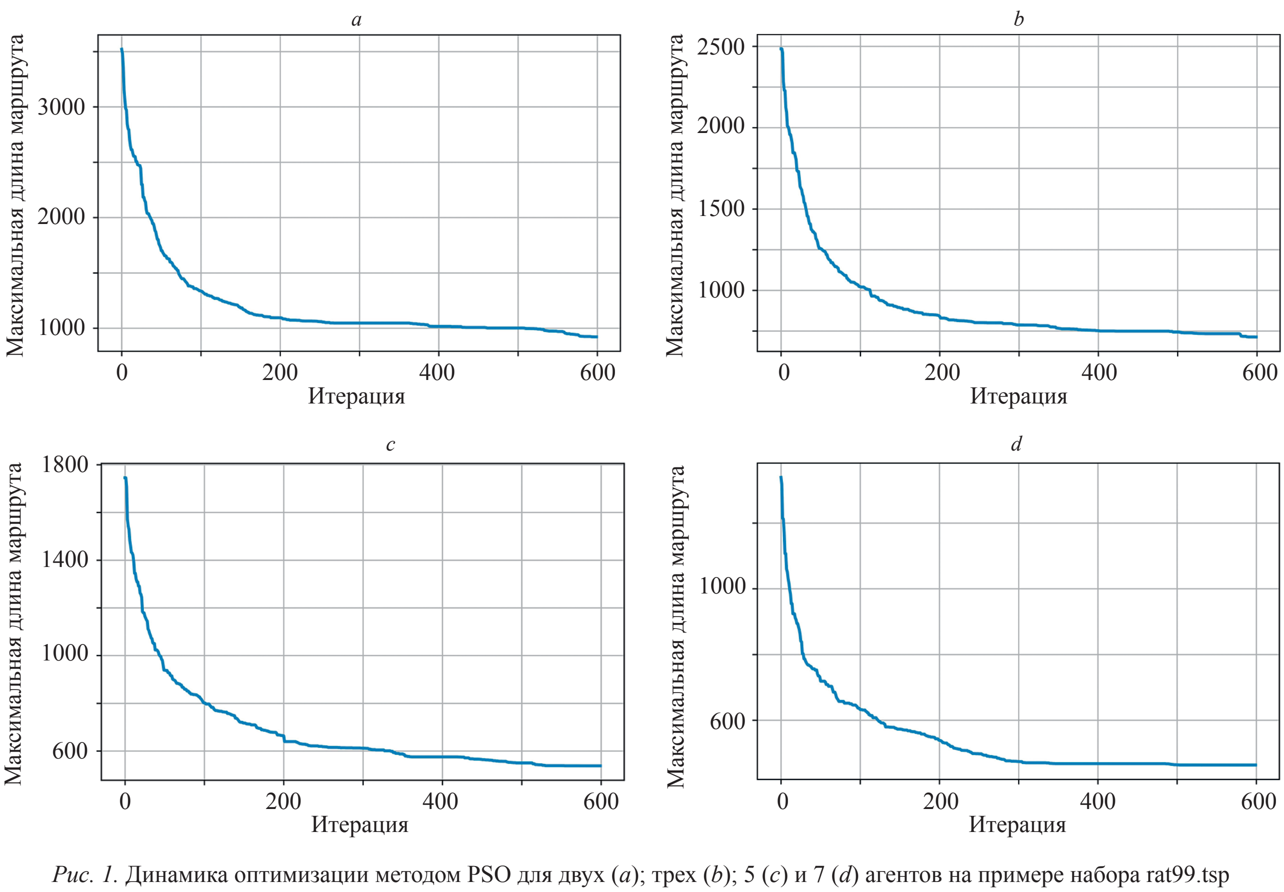


*Рис. 1.* Динамика оптимизации методом PSO для двух (*a*); трех (*b*); 5 (*c*) и 7 (*d*) агентов на примере набора rat99.tsp

*Fig. 1.* PSO optimization dynamics for the multiple traveling salesmen problem with 2 (*a*), 3 (*b*), 5 (*c*), and 7 (*d*) salesmen on the rat99.tsp benchmark instance

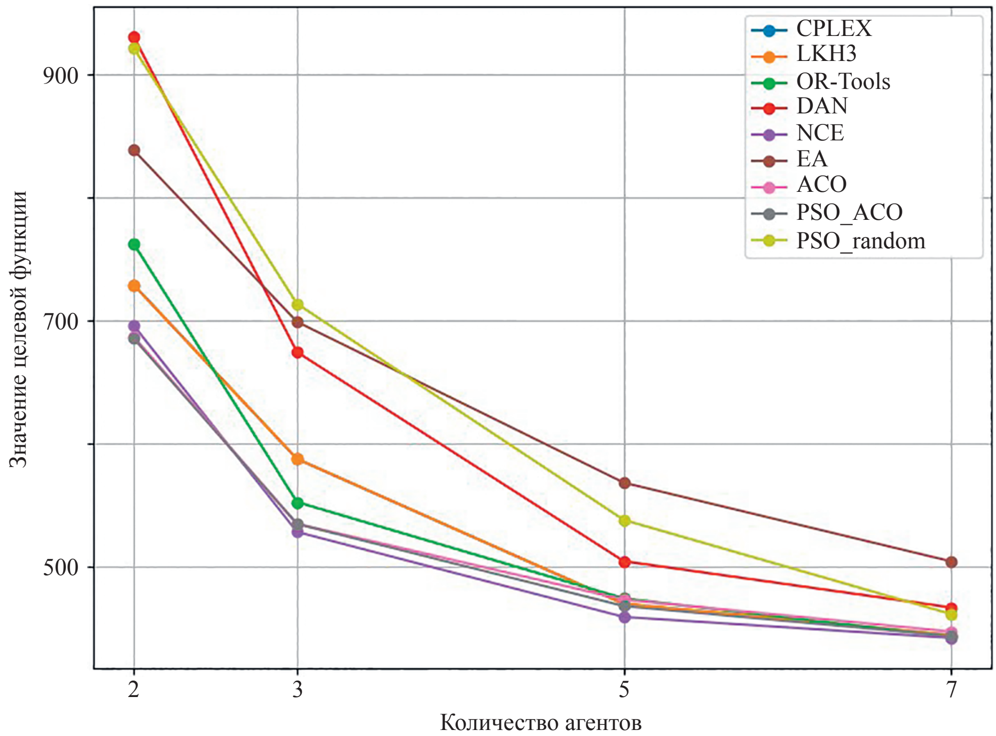


*Рис. 2*. Сравнение итоговых значений метрики «минимакс» для разных методов при различном числе агентов на примере rat99.tsp

*Fig. 2*. Comparison of final minimax metric values for different methods with varying numbers of salesmen on the rat99.tsp benchmark instance

и особыми свойствами окрестностей глобального оптимума в пространстве решений.

По ходу оптимизации PSO наблюдается поиск решений в локальной окрестности текущего глобального оптимума, которые можно определить как множество маршрутов, отличающихся от глобального оптимума одним (loc-1), двумя (loc-2) или несколькими обменами городов. По мере увеличения такого расстояния (количества обменов) плотность благоприятных решений уменьшается, и вероятность найти маршрут, более оптимальный, чем текущий глобальный, снижается. Например, для 99 вершин количество вариантов перестановок в окрестности loc-2 имеет порядок $10^8$, а в окрестности loc-3 — порядок $10^{12}$, при этом общее число вариантов, которые генерируются программой метода PSO при 600 итерациях и 800 тыс. частиц, не превышает $5 \cdot 10^8$, что еще раз подчеркивает эффективность и важность целенаправленного поиска в суженном пространстве решений.

Для объективного сравнения результаты работы алгоритма PSO сопоставлялись с характеристиками базовых методов (CPLEX [10], LKH–3 [15], ORTools [16]), нейросетевых методов (DAN, NCE [10]), генетического (эволюционного) метода (EA [9]), алгоритма ACO и комбинированного подхода к методам оптимизации PSO_ACO. Количественные показатели базовых методов были взяты из работ [9, 10, 15, 16]. Для корректного сопоставления использовалась метрика «минимакс», так как именно этот показатель представлен в большинстве базовых исследований. В таблице представлены сравнительные результаты работы перечисленных методов для двух, трех, 5 и 7 агентов и на разных тестовых наборах.

Сопоставление значений по «минимаксному» критерию, представленных в таблице, показывает, что комбинированная схема PSO_ACO на всех четырех наборах библиотеки TSPLIB (eil51, berlin52, eil76, rat99) и при любом из рассматриваемом числе агентов (два, три, 5, 7) неизменно возвращает маршруты меньшей длины, чем метод PSO_random. Наибольший выигрыш отмечается для более крупных наборов при двух-трех агентах, тогда как для eil51 при 7 агентах разница минимальна, но остается в пользу PSO_ACO.

На рис. 3 показаны маршруты для 5 агентов на наборе eil51 с указанием соответствующих значений «минимаксного» критерия: PSO_random (рис. 3, *a*) формирует явно неравномерную нагрузку («минимакс» равен 151,7), ACO (рис. 3, *c*) и PSO_ACO (рис. 3, *d*) демонстрируют более сбалансированные маршруты («минимакс» равен 118,2 и 118,1 соответственно), а CPLEX уступает ACO и PSO_ACO по визуальной четкости и показателю «минимакс», равному 124,0 (рис. 3, *b*).

Таким образом, PSO_ACO не только устойчиво превосходит PSO_random количественно на всех тестах, но и визуально обеспечивает более равномерное распределение нагрузки между агентами. Во всех 16 экспериментальных конфигурациях значения, полученные PSO_ACO, не превышают соответствующие результаты PSO_random, что указывает на более высокую результативность гибридного алгоритма и его стабильность при варьировании размеров задачи и числа агентов. Данное превосходство подтверждает высокий потенци-

*Таблица*. Значения «минимаксного» критерия для разных методов решения

*Table*. Minimax criterion values for the various solution methods

| Метод решения | Тестовый набор | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | eil51 | | | | berlin52 | | | | eil76 | | | | rat99 | | | |
| | Число агентов | | | | Число агентов | | | | Число агентов | | | | Число агентов | | | |
| | 2 | 3 | 5 | 7 | 2 | 3 | 5 | 7 | 2 | 3 | 5 | 7 | 2 | 3 | 5 | 7 |
| CPLEX | **222,7** | **159,6** | 124,0 | **112,1** | **4110,2** | 3244,4 | **2441,4** | **2440,9** | **280,9** | 197,3 | 150,3 | 139,6 | 728,8 | 587,2 | 469,3 | 443,9 |
| LKH3 | **222,7** | **159,6** | 124,0 | **112,1** | **4110,2** | 3244,4 | **2441,4** | **2440,9** | **280,9** | 197,3 | 150,3 | 139,6 | 728,8 | 587,2 | 469,3 | 443,9 |
| OR-Tools | 243,3 | 170,5 | 127,5 | **112,1** | 4665,5 | 3311,3 | 2482,6 | **2440,9** | 318,0 | 212,4 | **143,4** | 128,3 | 762,2 | 552,1 | 473,7 | 442,5 |
| DAN | 274,2 | 178,9 | 158,6 | 118,1 | 5226,0 | 4278,0 | 2759,0 | 2697,0 | 361,1 | 251,5 | 170,9 | 148,5 | 930,8 | 674,1 | 504,0 | 466,4 |
| NCE | 235,0 | 170,3 | 121,6 | **112,1** | **4110,2** | 3274,0 | 2660,0 | 2441,0 | 285,5 | 211,0 | 144,6 | **127,6** | 695,8 | 527,8 | 458,6 | **441,6** |
| EA | 248,5 | 190,6 | 134,8 | 116,5 | 4472,8 | 3567,4 | 2694,6 | 2441,4 | 342,0 | 259,2 | 190,5 | 158,8 | 838,4 | 698,7 | 567,9 | 504,0 |
| ACO | **222,7** | **159,6** | 118,2 | 112,5 | 4115,9 | 3188,1 | 2466,8 | **2440,9** | 281,9 | 197,0 | 150,9 | 128,2 | 686,6 | 534,4 | 472,7 | 446,7 |
| PSO_ACO | **222,7** | **159,6** | **118,1** | **112,1** | **4110,2** | **3171,6** | 2455,6 | **2440,9** | 281,8 | **196,0** | 143,8 | 127,8 | **685,2** | **534,3** | **467,5** | 443,2 |
| PSO_random | 264,5 | 183,7 | 151,7 | 112,7 | 5200,3 | 3561,1 | 2758,2 | 2445,2 | 382,4 | 262,6 | 167,7 | 149,5 | 921,5 | 713,4 | 537,6 | 461,1 |

*Примечание*. Полужирным шрифтом выделены наименьшие значения «минимаксного» критерия для каждого теста и различного числа агентов.

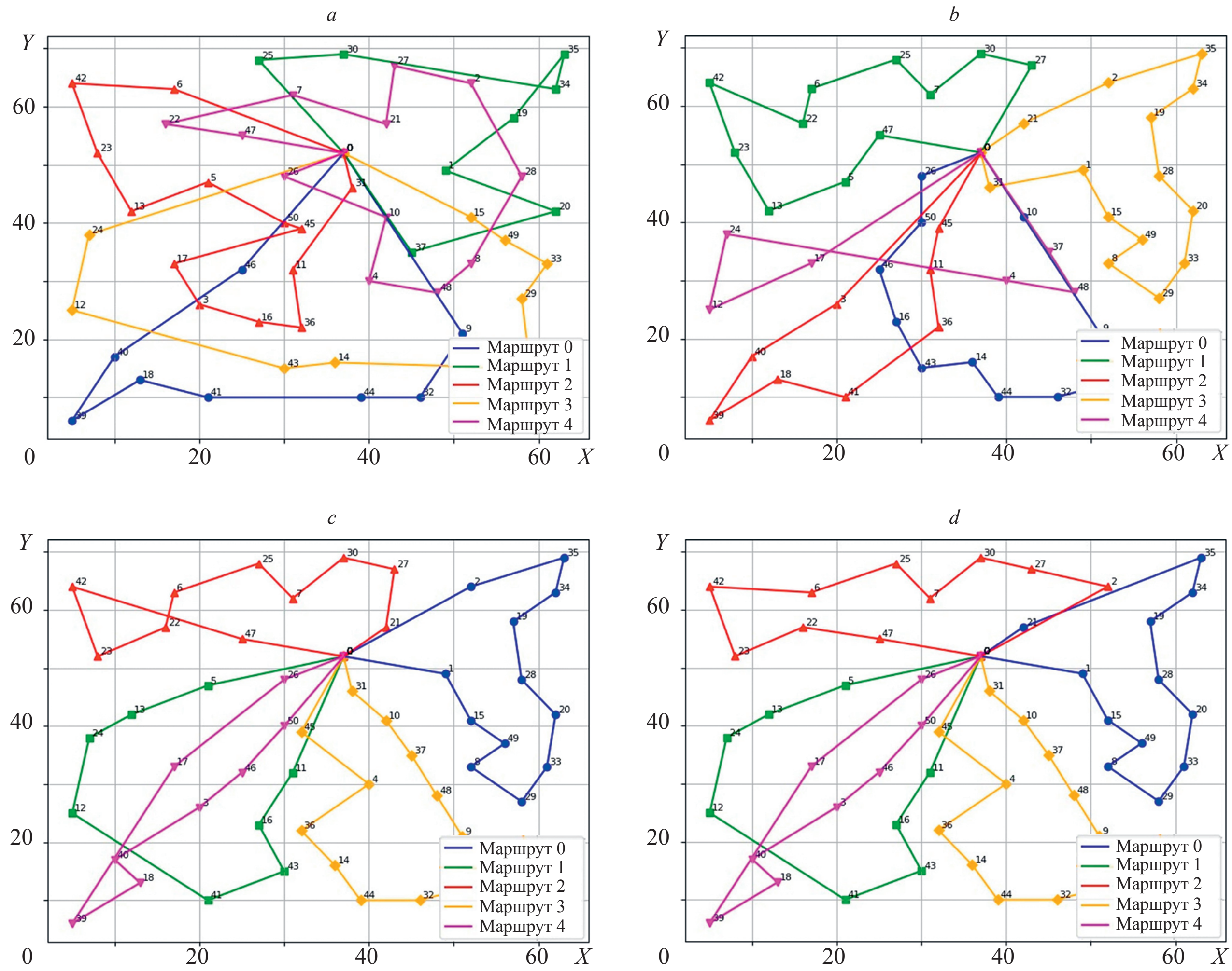


*Рис. 3*. Визуализация маршрутов для 5 агентов на наборе eil51 с общим депо 0 при разных методах решения: PSO_random (*a*); CPLEX (*b*); ACO (*c*); PSO_ACO (*d*)

*Fig. 3*. Routes of 5 agents on the eil51 benchmark (common depot 0) obtained by different methods: PSO_random (*a*), CPLEX (*b*), ACO (*c*), PSO_ACO (*d*)

ал комбинирования роевой оптимизации с алгоритмами ACO для решения задачи mTSP по «минимаксному» критерию.

## Заключение

Алгоритм роя частиц (Particle Swarm Optimization, PSO) обладает рядом преимуществ, таких как способность к глобальному поиску, простота реализации и возможность параллелизации вычислений.

Можно выделить несколько направлений для дальнейшего развития метода PSO. Одним из них является интеграция дискретного PSO с локальными эвристиками, такими как 2-opt, а также с элементами алгоритмов муравьиной оптимизации для более глубокого исследования пространства решений. Важным направлением является адаптивная настройка параметров алгоритма PSO по мере эволюции роя, что позволяет избежать преждевременного «застревания» в локальных минимумах. Кроме того, представляет интерес целенаправленное исследование разреженных областей в окрестностях локальных минимумов (loc-k) с использованием дополнительных метрик расстояния между перестановками и методов кластеризации. Наконец, дальнейший теоретический анализ вычислительной сложности алгоритма, а также получение новых результатов о верхних границах длины цепочек обменов будут способствовать более глубокому пониманию и более надежному применению данного подхода.

Проведенный вычислительный эксперимент демонстрирует потенциал разработанного подхода к методу PSO для решения мультиагентной задачи коммивояжёра (Multiple Traveling Salesman Problem, mTSP) по «минимаксному» критерию, а представленные результаты позволяют визуально и количественно оценить эффективность предложенной методики на тестовых TSP-наборах.

## Литература


1. Carter A.E., Ragsdale C.T. A new approach to solving the multiple traveling salesperson problem using genetic algorithms // European Journal of Operational Research. 2006. V. 175. N 1. P. 246–257. https://doi.org/10.1016/j.ejor.2005.04.027
2. Смирнов А.В. Исследование влияния степени овражности целевой функции на погрешность определения координат ее минимума // Российский технологический журнал. 2023. Т. 11. № 6. С. 57–67. https://doi.org/10.32362/2500-316X-2023-11-6-57-67
3. Boudjelaba K., Ros F., Chikouche D. Potential of particle swarm optimization and genetic algorithms for FIR filter design // Circuits, Systems, and Signal Processing. 2014. V. 33. N 10. P. 3195–3222. https://doi.org/10.1007/s00034-014-9800-y
4. Soylu B. A general variable neighborhood search heuristic for multiple traveling salesmen problem // Computers & Industrial Engineering. 2015. V. 90. P. 390–401. https://doi.org/10.1016/j.cie.2015.10.010
5. Elloumi W., Abeda H.E., Abraham A., Alimi A.M. A comparative study of the improvement of performance using a PSO modified by ACO applied to TSP // Applied Soft Computing. 2014. V. 25. P. 234–241. https://doi.org/10.1016/j.asoc.2014.09.031
6. Tang L., Liu J., Rong A., Yang Z. A multiple traveling salesman problem model for hot rolling scheduling in Shanghai Baoshan Iron & Steel Complex // European Journal of Operational Research. 2000. V. 124. N 2. P. 267–282. https://doi.org/10.1016/S0377-2217(99)00380-X
7. Lu L.C., Yue T.W. Mission-oriented ant-team ACO for min–max MTSP // Applied Soft Computing. 2019. V. 76. P. 436–444. https://doi.org/10.1016/j.asoc.2018.11.048
8. Eberhart R.C., Shi Y. Comparison between genetic algorithms and particle swarm optimization // Lecture Notes in Computer Science. 1998. V. 1447. P. 611–616. https://doi.org/10.1007/BFb0040812
9. Lupoaie V.-I., Chili I.-A., Breaban M.E., Raschip M. SOM-guided evolutionary search for solving MinMax Multiple-TSP // Proc. of the IEEE Congress on Evolutionary Computation (CEC). 2019. P. 73–80. https://doi.org/10.1109/cec.2019.8790276
10. Kim M., Park J., Park J. Learning to CROSS exchange to solve min-max vehicle routing problems // Proc. of the 11th International Conference on Learning Representations (ICLR). 2023. P. 1–12.
11. Tasgetiren M.F., Sevkli M., Liang Y.C., Gencyilmaz G. Particle swarm optimization algorithm for permutation flowshop sequencing problem // Lecture Notes in Computer Science. 2004. V. 3172. P. 382–389. https://doi.org/10.1007/978-3-540-28646-2_38
12. Liao C.J., Tseng C.T., Luarn P. A discrete version of particle swarm optimization for flowshop scheduling problems // Computers and Operations Research. 2007. V. 34. N 10. P. 3099–3111. https://doi.org/10.1016/j.cor.2005.11.017
13. Junjie P., Dingwei W. An ant colony optimization algorithm for Multiple Travelling Salesman Problem // Proc. of the First International Conference on Innovative Computing, Information and Control (ICICIC’06). 2006. V. 1. P. 210–213. https://doi.org/10.1109/icicic.2006.40
14. Necula R., Breaban M., Raschip M. Tackling the Bi-criteria facet of Multiple Traveling Salesman Problem with Ant Colony Systems // Proc. of the IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI). 2015. P. 873–880. https://doi.org/10.1109/ICTAI.2015.127
15. Helsgaun K. An Extension of the Lin-Kernighan-Helsgaun TSP solver for constrained traveling salesman and vehicle routing problems // Occasional Paper of the Roskilde University, International Development Studies. 2017. V. 12. P. 966–980.
16. Perron L., Furnon V. OR-Tools v9.6. 2019. URL: https://developers.google.com/optimization/


## References


1. Carter A.E., Ragsdale C.T. A new approach to solving the multiple traveling salesperson problem using genetic algorithms. *European Journal of Operational Research*, 2006, vol. 175, no. 1, pp. 246–257. https://doi.org/10.1016/j.ejor.2005.04.027
2. Smirnov A.V. Investigation of influence of objective function valley ratio on the determination error of its minimum coordinates. *Russian Technological Journal*, vol. 11, no. 6, pp. 57–67. (in Russian). https://doi.org/10.32362/2500-316X-2023-11-6-57-67
3. Boudjelaba K., Ros F., Chikouche D. Potential of particle swarm optimization and genetic algorithms for FIR filter design. *Circuits, Systems, and Signal Processing*, 2014, vol. 33, no. 10, pp. 3195–3222. https://doi.org/10.1007/s00034-014-9800-y
4. Soylu B. A general variable neighborhood search heuristic for multiple traveling salesmen problem. *Computers & Industrial Engineering*, 2015, vol. 90, pp. 390–401. https://doi.org/10.1016/j.cie.2015.10.010
5. Elloumi W., Abeda H.E., Abraham A., Alimi A.M. A comparative study of the improvement of performance using a PSO modified by ACO applied to TSP. *Applied Soft Computing*, 2014, vol. 25, pp. 234–241. https://doi.org/10.1016/j.asoc.2014.09.031
6. Tang L., Liu J., Rong A., Yang Z. A multiple traveling salesman problem model for hot rolling scheduling in Shanghai Baoshan Iron & Steel Complex. *European Journal of Operational Research*, 2000, vol. 124, no. 2, pp. 267–282. https://doi.org/10.1016/S0377-2217(99)00380-X
7. Lu L.C., Yue T.W. Mission-oriented ant-team ACO for min–max MTSP. *Applied Soft Computing*, 2019, vol. 76, pp. 436–444. https://doi.org/10.1016/j.asoc.2018.11.048
8. Eberhart R.C., Shi Y. Comparison between genetic algorithms and particle swarm optimization. *Lecture Notes in Computer Science*, 1998, vol. 1447, pp. 611–616. https://doi.org/10.1007/BFb0040812
9. Lupoaie V.-I., Chili I.-A., Breaban M.E., Raschip M. SOM-guided evolutionary search for solving MinMax Multiple-TSP. *Proc. of the IEEE Congress on Evolutionary Computation (CEC)*, 2019, pp. 73–80. https://doi.org/10.1109/cec.2019.8790276
10. Kim M., Park J., Park J. Learning to CROSS exchange to solve min-max vehicle routing problems. *Proc. of the 11th International Conference on Learning Representations (ICLR)*, 2023, pp. 1–12.
11. Tasgetiren M.F., Sevkli M., Liang Y.C., Gencyilmaz G. Particle swarm optimization algorithm for permutation flowshop sequencing problem. *Lecture Notes in Computer Science*, 2004, vol. 3172, pp. 382–389. https://doi.org/10.1007/978-3-540-28646-2_38
12. Liao C.J., Tseng C.T., Luarn P. A discrete version of particle swarm optimization for flowshop scheduling problems. *Computers and Operations Research*, 2007, vol. 34, no. 10, pp. 3099–3111. https://doi.org/10.1016/j.cor.2005.11.017
13. Junjie P., Dingwei W. An ant colony optimization algorithm for Multiple Travelling Salesman Problem. *Proc. of the First International Conference on Innovative Computing, Information and Control (ICICIC’06)*, 2006, vol. 1, pp. 210–213. https://doi.org/10.1109/icicic.2006.40
14. Necula R., Breaban M., Raschip M. Tackling the Bi-criteria facet of Multiple Traveling Salesman Problem with Ant Colony Systems. *Proc. of the IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI)*, 2015, pp. 873–880. https://doi.org/10.1109/ICTAI.2015.127
15. Helsgaun K. An Extension of the Lin-Kernighan-Helsgaun TSP solver for constrained traveling salesman and vehicle routing problems. *Occasional Paper of the Roskilde University, International Development Studies*, 2017, vol. 12, pp. 966–980.
16. Perron L., Furnon V. *OR-Tools v9.6*, 2019. Available at : https://developers.google.com/optimization/


## Авторы


**Мифтахов Эльдар Наилевич** — доктор физико-математических наук, профессор, МИРЭА — Российский технологический университет, Москва, 119454, Российская Федерация, sc 56178153800, https://orcid.org/0000-0002-0471-5949, promif@mail.ru

**Акимов Андрей Анатольевич** — кандидат физико-математических наук, доцент, доцент, МИРЭА — Российский технологический университет, Москва, 119454, Российская Федерация, sc 56428598700, https://orcid.org/0000-0003-3387-2959, andakm@yandex.ru


## Authors


**Eldar N. Miftakhov** — D.Sc. (Physics & Mathematics), Professor, MIREA — Russian Technological University, Moscow, 119454, Russian Federation, sc 56178153800, https://orcid.org/0000-0002-0471-5949, promif@mail.ru

**Andrey A. Akimov** — PhD (Physics & Mathematics), Associate Professor, Associate Professor, MIREA — Russian Technological University, Moscow, 119454, Russian Federation, sc 56428598700, https://orcid.org/0000-0003-3387-2959, andakm@yandex.ru

**Гнатенко Юлия Ахнафовна** — кандидат физико-математических наук, доцент, доцент, Стерлитамакский филиал Уфимского университета науки и технологий, Стерлитамак, 453103, Российская Федерация, sc 9234055300, https://orcid.org/0009-0009-9264-3989, y.a.gnatenko@struust.ru

**Yuliya A. Gnatenko** — PhD (Physics & Mathematics), Associate Professor, Associate Professor, Branch of the Ufa University of Science and Technology, Sterlitamak, 453103, Russian Federation, sc 9234055300, https://orcid.org/0009-0009-9264-3989, y.a.gnatenko@struust.ru

*Статья поступила в редакцию 11.03.2025*
*Одобрена после рецензирования 20.08.2025*
*Принята к печати 24.09.2025*

*Received 11.03.2025*
*Approved after reviewing 20.08.2025*
*Accepted 24.09.2025*