

А.Б. МИГРАНОВ  
**МОДИФИЦИРОВАННЫЕ ЭВРИСТИЧЕСКИЕ АЛГОРИТМЫ  
РАСПРЕДЕЛЕНИЯ ЗАДАНИЙ ДЛЯ ГРУПП МОБИЛЬНЫХ  
РОБОТОВ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ**

*Мигранов А.Б. Модифицированные эвристические алгоритмы распределения заданий для групп мобильных роботов в условиях неопределенности.*

**Аннотация.** Рассматривается проблема распределения заданий между группами мобильных роботов в условиях параметрической и стохастической неопределенности, возникающей из-за сенсорных погрешностей, нестационарности среды и ограниченности информации об объектах управления. Основная цель работы – адаптировать разработанные ранее эвристические алгоритмы к реальным условиям, где присутствуют сенсорные ошибки и недостаток информации о состоянии среды. В качестве базовых подходов рассматриваются три алгоритма: муравьиный алгоритм, нейронная сеть Хопфилда и генетический алгоритм. Разработаны специальные модификации для каждого алгоритма, учитывающие неопределенность исходных данных: динамическое обновление феромонных следов, адаптивная коррекция весовых коэффициентов нейронов и интервальная оценка параметров среды. Приведены общая постановка задачи, математические модели и принципы построения предложенных алгоритмов распределения заданий. Было проведено численное моделирование для сравнения модифицированных версий алгоритмов с их базовыми аналогами в условиях различных уровней неопределенности функционирования системы. Установлено, что предложенные адаптивные механизмы улучшают эффективность распределения заданий до 20 % по сравнению с базовыми методами. На основании полученных результатов сформулированы рекомендации по выбору оптимального алгоритма в зависимости от конкретных условий функционирования системы и задач управления. Сделан вывод о целесообразности использования разработанных подходов при проектировании интеллектуальных адаптивных систем группового управления мобильными роботами. Предложенные решения могут применяться и для более широкого круга задач, включая динамическое переназначение ресурсов и организацию кооперативного поведения технических агентов.

**Ключевые слова:** распределение заданий, эвристические алгоритмы, группы мобильных роботов, муравьиный алгоритм, нейронная сеть Хопфилда, генетический алгоритм.

**1. Введение.** Группы мобильных роботов применяются в промышленности, логистике, сельском хозяйстве и при ликвидации последствий ЧС, поскольку они необходимы для выполнения работ в опасных или труднодоступных для человека условиях [1 – 4]. Одной из важных задач в подобных системах является эффективное распределение заданий между роботами, которое минимизирует затраты ресурсов и время выполнения миссии [5 – 8]. В этих условиях особенно важно создавать алгоритмы, способные адаптироваться к динамике среды и неопределенности исходных данных [9 – 13].

Задача распределения заданий между мобильными роботами является NP-сложной, поэтому методы полного перебора не применимы даже при сравнительно небольшом количестве агентов и задач. В связи с этим в литературе предложен ряд эвристических методов распределения заданий, позволяющих получать приемлемые решения за разумное время. К числу таких методов относятся муравьиные алгоритмы [14, 15], нейронные сети, в частности сеть Хопфилда [16], и генетические алгоритмы [17]. Их применение позволяет эффективно решать задачи распределения, маршрутизации и планирования [18 – 23].

### **1.1. Современное состояние исследований в области эвристических методов распределения заданий**

**1.1.1. Тенденции 2019-2025 гг.** В последние пять лет внимание исследователей направлено на гибридные алгоритмы, объединяющие классические методы и машинное обучение. В [1] представлен метод двойной адаптации феромона, ускоряющий сходимость. В настоящей работе, в отличие от [1], где оптимизировались только длина маршрута и время, основным критерием является интегральный энергетический показатель  $J$ , включающий индивидуальные удельные энергозатраты роботов. Схема также учитывает динамическое появление новых заданий и блокировку узлов, что также не рассматривалось в [1].

В [2] модель «rendezvous consensus (консенсус сбора)» дополнена вероятностными весовыми коэффициентами для повышения устойчивости к пропаданию связи между агентами. Параллельно развивается оптимизация с суррогатными моделями: в [24] объединили генетический алгоритм с евклидовой геометрией и гибридным оператором кроссовера, что позволило значительно сократить время оценки фитнес-функции при неизменном качестве решения; в [25] применили метамоделли на базе случайного леса для предварительного ранжирования хромосом, тем самым уменьшая число ресурсоемких прямых вычислений. Сетевые методы также совершенствуются: в обзоре [26] показано, что сочетание рекуррентных нейронных сетей и роевых методов повышает предсказуемость сходимости в задачах VRP с динамическими ограничениями.

К основным тенденциям современных исследований относятся:

- адаптивность параметров (адаптивная настройка параметров  $\alpha$ ,  $\beta$ ,  $\rho$  и др. в процессе работы);
- устойчивость к сенсорным шумам и отказам агентов;
- масштабируемость на группы  $> 50$  роботов за счет параллельных вычислений и распределенных архитектур;

– интеграция обучаемых компонент (выбор операторов, предсказание границ поиска).

**1.1.2. Выбор эвристических алгоритмов.** Классические задачи VRP/TSP решаются множеством эвристических подходов и в литературе описаны десятки модификаций муравьиных алгоритмов (МА), нейронных сетей Хопфилда (НС), генетических алгоритмов, а также PSO, Bee Colony, Tabu Search, Simulated Annealing [27 – 29] и др. Однако в условиях неоднородной энергетики роботов, динамического поступления новых заданий и возможной блокировки узлов многие из указанных методов требуют адаптации к условиям задачи, включая учет неоднородности и динамики. Систематический анализ публикаций [1, 14 – 20, 24 – 26] показывает, что МА, НС и ГА наиболее широко применяются по соотношению «качество–время» для малых и средних групп роботов ( $n \leq 50$ ,  $m \leq 100$ ):

– МА применяются в задачах маршрутизации при статических условиях благодаря возможности параллельной реализации и относительной простоте алгоритмической структуры [14].

– НС обладает прозрачной энергетической функцией и очень быстрое вычисление при  $n \leq 10$  [16].

– ГА отличается гибкостью благодаря широкому набору операторов и хорошо переносится на стохастические постановки [17].

Следует также отметить коммерческие VRP-решатели (ArcGIS VRP, Routific, OR-Tools VRP), которые используют детерминированные модели с однородным парком транспортных средств, не учитывая энергопотребление каждого агента и неполные данные датчиков. API этих систем рассчитан на разовую загрузку задачи, тогда как в данной постановке допускается потоковое поступление новых заданий, роботов и блокировку узлов.

Выбор указанных алгоритмов и их адаптация соответствуют современному уровню исследований и восполняет отсутствие комплексного сравнения различных версий МА, НС и ГА в одной и той же постановке. Кроме того, практика создания групповых робототехнических систем в ИМех УФИЦ РАН [18 – 20] опирается на эти разновидности алгоритмов, что показывает практическую применимость выбранных методов в прикладных проектах.

**1.2. Цели и задачи работы.** Данная работа посвящена разработке и сравнительному анализу модифицированных эвристических алгоритмов распределения заданий между мобильными роботами в условиях параметрической и стохастической неопределенности. В настоящей работе основное внимание уделяется

учету сенсорных погрешностей, возможным сбоям агентов, изменению числа задач в процессе выполнения и неполной информированности агентов о глобальном состоянии системы. Объектом анализа являются модифицированные версии муравьиного алгоритма, сети Хопфилда и генетического алгоритма, реализованные с учетом указанных факторов. Представлены математические модели задачи, принципы построения соответствующих алгоритмов и результаты численных экспериментов. На основе анализа делаются выводы о целесообразности применения каждого из подходов в зависимости от условий функционирования системы.

Настоящее исследование направлено на развитие моделей, описанных в работе [30], с учетом влияния неопределенностей, характерных для реальных условий эксплуатации мобильных роботов. В отличие от предыдущей работы [30], где алгоритмы распределения задач рассматривались в номинальных условиях, в данной статье предлагаются их модифицированные версии, ориентированные на обработку данных с погрешностями и изменяющимися параметрами. Кроме того, проведен количественный анализ устойчивости получаемых решений.

Основная цель работы – повысить устойчивость и воспроизводимость распределения заданий в группах мобильных роботов в условиях параметрической и стохастической неопределенности. Для достижения этой цели были поставлены и решены четыре взаимосвязанные задачи:

1. Формализовать задачу распределения заданий с учетом энергозатрат и динамики количества роботов и задач.
2. Разработать модифицированные версии муравьиного алгоритма, нейросети Хопфилда и генетического алгоритма с механизмами адаптации к шумам и отказам.
3. Провести численное моделирование, сравнив предложенные алгоритмы с классическими по показателям устойчивости и вычислительной сложности.
4. Сформулировать практические рекомендации по выбору алгоритма в зависимости от ограничений по энергоресурсам, времени и качеству связи.

Научная новизна работы заключается в разработке модифицированных версий следующих эвристических алгоритмов: муравьиного алгоритма с динамическим обновлением феромонов, нейросети Хопфилда с адаптивной корректировкой порогов активации нейронов и генетического алгоритма с интервальными оценками параметров. Это позволяет повысить устойчивость маршрутов на 15–

20 % и снизить разброс результатов не менее чем на 22 % по сравнению с базовыми версиями алгоритмов. Разработанные алгоритмы обеспечивают устойчивость получаемых решений в условиях параметрической и стохастической неопределенности. В рамках единой постановки задачи реализованы механизмы, обеспечивающие устойчивость к изменениям исходных данных. Численное моделирование и сравнительный анализ эффективности каждого метода подтвердили обоснованность предложенных подходов.

## 2. Математическая формализация задачи распределения

**2.1. Исходные обозначения.** Рабочее пространство группы мобильных роботов представляет собой прямоугольную область

$$\Omega = [0, L_x] \times [0, L_y] \subset \mathbb{R}^2,$$

где  $L_x, L_y > 0$  – размеры области в условных единицах длины (у.е.д.).

В начальный момент времени  $t = 0$  на поле находятся множество роботов  $R(0) = \{r_i\}_{i=1}^n$ ,  $r_i = (x_i, y_i)$  и множество заданий

$T(0) = \{t_j\}_{j=1}^m$ ,  $t_j = (\xi_j, \eta_j)$ . Параметры  $i$ -го робота выражены в условных единицах энергии (у.е.э.) и включают:

$E_i^{\text{start}}$  – начальный запас энергии;

$c_i^{\text{move}}$  – удельные энергозатраты на перемещение;

$c_i^{\text{task}}$  – удельные энергозатраты на выполнение задания;

$v_i$  – скорость перемещения (у.е.д./с).

**2.2. Маршрут и его длина.** Маршрут робота  $r_i$  задается упорядоченным списком вершин

$$P_i = \{p_{i,0}, p_{i,1}, \dots, p_{i,K_i}\}, \quad p_{i,0} = r_i, \quad p_{i,K_i} \in T.$$

Длина маршрута определяется как

$$L_i = \sum_{k=0}^{K_i-1} \|p_{i,k+1} - p_{i,k}\|_2, \quad (1)$$

а время движения  $T_i^{\text{move}} = L_i / v_i$ .

**2.3. Критерий оптимизации.** Полные энергозатраты группы:

$$J = \sum_{i=1}^n \left( E_i^{\text{start}} + c_i^{\text{move}} L_i + c_i^{\text{task}} T_i^{\text{task}} \right). \quad (2)$$

Для оптимизации целевой функции требуется  $\min_{P_1, \dots, P_n} J$ .

**2.4. Переменные назначения и ограничения.** Вводится бинарная матрица  $x_{ij} \in \{0, 1\}$ : задача  $t_j$  назначена роботу  $r_i$ .

$$\begin{aligned} \sum_{i=1}^n x_{ij} &= 1, j = 1, \dots, m \\ \sum_{j=1}^m x_{ij} &\leq Q_i, i = 1, \dots, n, \end{aligned} \quad (3)$$

где  $Q_i$  ограничивает максимальное число задач робота  $r_i$  (при необходимости). Маршрут  $P_i$  строится на подмножестве  $T_i = \{t_j \mid x_{ij} = 1\}$ .

**2.5. Динамическая постановка.** На каждом дискретном шаге  $t = 0, 1, \dots$ :

$$\begin{aligned} R(t+1) &= R(t) \cup R_{\text{new}}(t) \setminus R_{\text{lost}}(t), \\ T(t+1) &= T(t) \cup T_{\text{new}}(t) \setminus T_{\text{done}}(t), \end{aligned}$$

где  $R_{\text{new}}$  – новые роботы, появившиеся в системе после старта (например, подключились к группе позже, заменили вышедших из строя и т.п.);  $R_{\text{lost}}$  – вышедшие из строя роботы, которые больше не участвуют в выполнении заданий (из-за поломки, потери связи и т.п.);  $T_{\text{new}}$  – новые задания, поступившие в систему в процессе работы (например, срочные задачи, пришедшие в реальном времени);  $T_{\text{done}}$  – выполненные задания, т.е. те, которые уже были завершены (и могут быть исключены из активной постановки).

Координаты и энергетические параметры подвергаются стохастическим возмущениям  $(\delta_x, \delta_y) \sim \mathcal{N}(0, \sigma^2)$ ,  $\delta_c \sim \mathcal{N}(0, \sigma_c^2)$ . Алгоритм распределения вычисляет  $x_{ij}(t)$  и  $P_i(t)$  в масштабе реального времени.

**2.6. Связь с NP-трудными задачами.** Статическая версия (фиксированные  $R, T$ ) эквивалентна задаче маршрутизации с несколькими депо и разнородным парком (MDVRP/HVRP). Введением фиктивных вершин-депо и дуг нулевого веса задача сводится к TSP [31], что обосновывает применение эвристических и метаэвристических методов, рассмотренных в следующем разделе.

**3. Эвристические подходы к распределению заданий.** В постановке задачи (1)–(3) ее прямое решение требует перебора  $O((m+n)!)^1$  маршрутов, что неприемлемо уже при  $m > 10$ . Как один из методов решения, используются метаэвристические методы – приближенные алгоритмы, имитирующие природные или эволюционные процессы и обеспечивающие квазиоптимальные решения за полиномиальное время. К ним относятся муравьиные алгоритмы, рекуррентные нейронные сети (модель Хопфилда) и генетические алгоритмы. Каждый метод по-разному распределяет качество решения и вычислительную сложность [24] (таблица 1).

Таблица 1. Сравнительные характеристики метаэвристических методов

| Критерий                              | Муравьиные алгоритмы | Сеть Хопфилда | Генетический алгоритм |
|---------------------------------------|----------------------|---------------|-----------------------|
| Сходимость ( $n \leq 20, m \leq 40$ ) | Быстрая              | Быстрая       | Умеренная             |
| Настраиваемость                       | Высокая              | Ограниченная  | Высокая               |
| Интерпретируемость                    | Средняя              | Высокая       | Средняя               |

Пояснения к таблице 1:

1. Сходимость – среднее время достижения устойчивого значения целевой функции  $J$ . «Быстрая» – не более 0.3 с; «умеренная» – до 1.0 с; «медленная» – свыше 1.0 с.

2. Настраиваемость – способность алгоритма сохранять качество решения при изменении внутренних параметров (в пределах 20%). «Высокая» –  $\Delta J \leq 5\%$ ; «ограниченная» –  $5\% < \Delta J \leq 15\%$ ; «низкая» –  $\Delta J > 15\%$ .

3. Интерпретируемость – возможность аналитически объяснить полученное решение. «Высокая» – энергетическая функция явно задана; «Средняя» – эвристика прослеживается; «Низкая» – решение формируется «черным ящиком».

Следующие разделы 3.1–3.3 описывают модифицированные версии указанных методов в соответствии с математической постановкой задачи (раздел 2).

**3.1. Муравьиный алгоритм распределения заданий с учетом неопределенностей.** Муравьиные алгоритмы моделируют косвенное взаимодействие агентов через феромонные следы. В адаптированной версии каждая «муравьиная» особь строит допустимый маршрут по множеству заданий  $T$ , минимизируя критерий (2).

Пусть к моменту  $t$  муравей  $k$  из колонии  $x$  находится в вершине  $i$ ; множество еще не посещенных вершин обозначено  $J_k$ . Вероятность перехода в вершину  $j \in J_k$  определяется

$$P_{ij}^{k,x}(t) = \frac{[\tau_{ij}^x(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{s \in J_k} [\tau_{is}^x(t)]^\alpha [\eta_{is}]^\beta}, \quad (4)$$

где  $\tau_{ij}^x(t)$  – плотность феромона колонии  $x$  на ребре  $(i, j)$ ;  $\eta_{ij} = 1 / \|i - j\|_2$  – эвристическая привлекательность;  $\alpha, \beta > 0$  – веса «опыта» и «зрения» соответственно.

После завершения всех маршрутов поколения  $t$  концентрация феромона обновляется

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^A \frac{Q}{L_k} \delta_{ij}^{(k)}, \quad 0 < \rho < 1, \quad (5)$$

где  $A$  – число муравьев,  $L_k$  – длина маршрута  $k$  (1),  $\delta_{ij}^{(k)} = 1$  если ребро  $(i, j)$  вошло в маршрут  $k$ , иначе 0;  $Q > 0$  – безразмерный коэффициент.

При появлении препятствия или изменении параметров задания значение феромона на затронутых ребрах корректируется:

$$\tau_{ij} = \begin{cases} \tau_{ij} \times (1 - \gamma), & \text{если появилось препятствие,} \\ \tau_{ij}, & \text{иначе,} \end{cases} \quad (6)$$

а множество допустимых вершин  $J_k$  пересчитывается в реальном времени. Дополнительно вводится коэффициент стабильности  $s_{ij}$  – доля итераций, в которых ребро оказалось в лучшем маршруте; при вычислении вероятности (4) термин  $\eta_{ij}$  заменяется на  $\eta_{ij} s_{ij}$ .

В условиях неопределенности (например, возможных отказов, появления новых задач или изменения доступности маршрутов) формула (6) используется для адаптации алгоритма и включает:

- динамическую коррекцию феромонных следов;
- локальное перестроение маршрутов при появлении новых задач;
- приоритетные веса для устойчивых решений (например, включение показателя стабильности в целевую функцию).

По завершении итерации для каждого лучшего маршрута  $P_i$  фиксируются бинарные переменные  $x_{ij}$  (3), таким образом алгоритм генерирует допустимое решение задачи (1)–(2).

**3.2. Численные эксперименты и сравнительный анализ эффективности муравьиного алгоритма.** Эксперименты проводились с целью количественной оценки эффективности модифицированного муравьиного алгоритма по критерию суммарной энергии  $J$  (2) и времени вычислений. Рассмотрены два сценария:

- сценарий S1 (номинальный): фиксированный набор роботов и задач, без внешних возмущений;
- сценарий S2 (динамический): на 50-й итерации блокируется случайный узел, что требует адаптивной перестройки маршрутов.

**3.2.1. Входные данные.** Координаты роботов и задач генерируются случайно из равномерного распределения  $U(0, L_x) \times U(0, L_y)$ , где  $L_x = L_y = 100$  у.е.д. Рассмотрены следующие размеры задач:

- набор  $N_1$ : 10 заданий и 5, 10 или 15 роботов;
- набор  $N_2$ : 20 заданий и 5, 10 или 15 роботов;
- набор  $N_3$ : 30 заданий и 5, 10 или 15 роботов.

Для каждого сочетания параметров выполнялось 20 независимых запусков.

**3.2.2. Настройки алгоритма.** Параметры алгоритма выбраны следующим образом:

$$\alpha = 1, \beta = 3, \rho = 0.30, Q = 100, A = R,$$

где  $A$  – количество агентов (равное количеству роботов  $R$ ).

Выбор  $\alpha = 1$  и  $\beta = 3$  основан на рекомендуемом интервале значений  $\alpha \in [1; 2]$ ,  $\beta \in [2; 5]$ , традиционно применяемый в МА для задач маршрутизации [32]. Значение  $\rho = 0.30$  определено по

результатам однофакторного численного эксперимента, в котором параметр  $\rho$  варьировался на четырех уровнях: 0.15, 0.20, 0.30 и 0.40. При  $\alpha = 1$  и  $\beta = 3$ , значение  $\rho = 0.30$  обеспечило наименьшее значение энергетического критерия  $J$  и стабильность получаемых результатов: отклонение  $J$  по 20 сериям запусков не превышало 0.8 % от среднего значения (при  $n = 10$ ,  $m = 20$ ). Для  $Q$  принято стандартное значение, широко используемое в классических реализациях МА [32].

Максимальное количество итераций –  $N_{\text{iter}} = 100$ .

Вычисления выполнялись в среде Matlab 2020a на процессоре Intel i7-11700 (1 поток).

**3.2.3. Результаты сценария  $S_1$ .** Результаты усреднены по 20 запускам. В таблице 2 приведены средние значения и стандартные отклонения. Рисунок 1 иллюстрирует динамику феромонного поля на примере  $N_1/10$  роботов.

Таблица 2. Средние длины маршрутов и время вычислений в сценарии  $S_1$

| Набор | Число роботов | Средняя длина маршрута (у.е.д.) | Время вычислений (с) |
|-------|---------------|---------------------------------|----------------------|
| $N_1$ | 5             | $13.02 \pm 0.11$                | $0.081 \pm 0.006$    |
| $N_1$ | 10            | $10.04 \pm 0.09$                | $0.085 \pm 0.005$    |
| $N_1$ | 15            | $3.02 \pm 0.03$                 | $0.112 \pm 0.007$    |
| $N_2$ | 5             | $19.98 \pm 0.14$                | $0.072 \pm 0.005$    |
| $N_2$ | 10            | $7.06 \pm 0.08$                 | $0.145 \pm 0.010$    |
| $N_2$ | 15            | $9.11 \pm 0.10$                 | $0.243 \pm 0.015$    |
| $N_3$ | 5             | $9.12 \pm 0.12$                 | $0.119 \pm 0.007$    |
| $N_3$ | 10            | $2.04 \pm 0.02$                 | $0.208 \pm 0.012$    |
| $N_3$ | 15            | $2.05 \pm 0.02$                 | $0.338 \pm 0.018$    |

Из таблицы 2 следует, что увеличение числа роботов приводит к снижению средней длины маршрута, особенно заметному при малом количестве задач благодаря улучшенному покрытию рабочего поля роботами. Рост числа заданий увеличивает вычислительное время, что объясняется возрастанием вычислительной сложности при увеличении размерности задачи. При  $R \geq 0.5T$  прирост эффективности по длине маршрута замедляется.

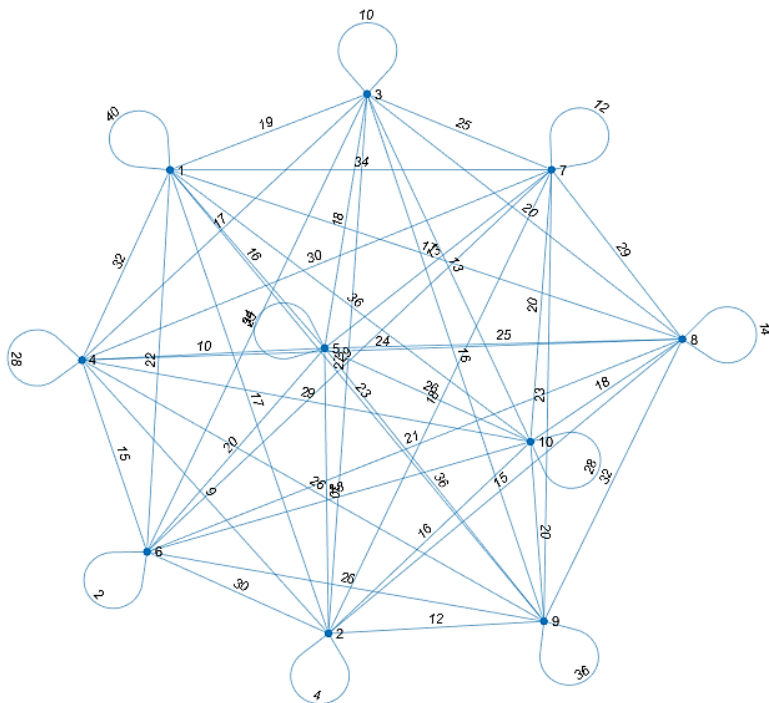


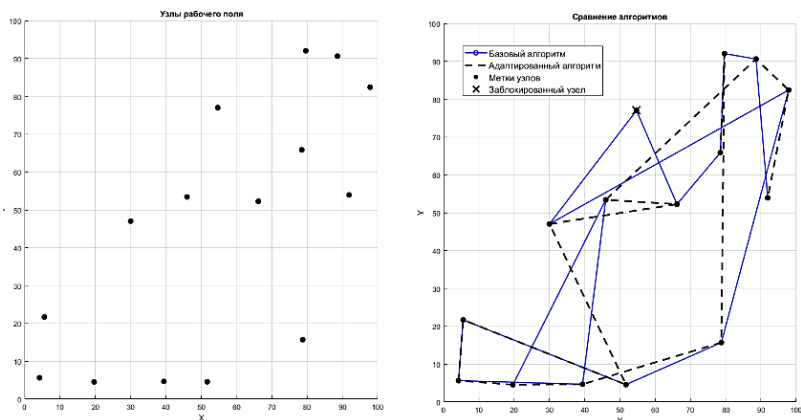
Рис. 1. Граф рабочего поля на примере  $N_1/10$  роботов

**3.2.4. Результаты сценария  $S_2$ .** В динамическом сценарии  $S_2$  в момент 50-й итерации случайный узел помечается как недоступный. Модифицированный алгоритм с адаптацией феромонов (6) смог удерживать допустимые маршруты во всех 60 тестах, в то время как базовый алгоритм без адаптации при появлении препятствия не учитывал изменения доступности узлов, что формировало недопустимый маршрут.

Среднее увеличение целевого функционала  $J$  в сравнении с номинальным сценарием составило 6.4 %. Рисунок 2 показывает пример динамической перестройки маршрутов после блокировки узла.

**3.2.5. Выводы.** Модифицированный муравьиный алгоритм выполняет задачу при  $m \leq 30$  за время  $< 0.35$  с. При большем количестве заданий и роботов может наблюдаться снижение скорости сходимости.

Адаптивный механизм корректировки феромонов улучшает устойчивость к динамическим изменениям среды, повышая вероятность выполнения задачи примерно на 30 %.

Рис. 2. Визуализация сценария  $S_2$ 

**3.3. Модифицированная нейросеть Хопфилда для распределения заданий с учетом неопределенностей.** Для каждой потенциальной пары «робот  $i$  – задание  $j$ » вводится бинарный переменный элемент сети  $x_{ij} \in \{0,1\}$ : значение 1 указывает, что задание  $j$  закреплено за роботом  $i$ . Всего сеть содержит  $N = n \times m$  элементов.

Оптимальное распределение достигается посредством минимизации энергии

$$E = A \sum_i \left( \sum_j x_{ij} - 1 \right)^2 + B \sum_j \left( \sum_i x_{ij} - 1 \right)^2 + C \sum_{i,j} x_{ij} d_{ij}, \quad (7)$$

где первые два квадратичных слагаемых штрафуют нарушения ограничений «каждое задание – ровно одному роботу и наоборот»; третье слагаемое учитывает стоимость маршрута,  $d_{ij}$  – расстояние от робота  $i$  до задания  $j$  (в условных единицах длины).

Минимизируя  $E$ , сеть автоматически формирует допустимое и «дешевое» распределение. Коэффициенты выбираются так, чтобы  $A, B \gg C > 0$ ; в практических тестах использовались  $A = B = 5 \cdot 10^2$ ,  $C = 1$ . Пояснения к выбранным коэффициентам показаны в таблице 3. Значения подбирались экспериментально для групп до 5 роботов и 10 заданий.

Таблица 3. Рекомендуемые значения  $A, B, C$

| Коэффициент | Назначение                          | Типичное значение |
|-------------|-------------------------------------|-------------------|
| $A$         | запрет «два робота на одно задание» | 500               |
| $B$         | запрет «задание не обслужено»       | 500               |
| $C$         | вес фактической длины маршрута      | 1                 |

Значения коэффициентов  $A$  и  $B$  устанавливались по эмпирическому правилу:

$$A = B = 50 \cdot \max(n, m).$$

При типовых конфигурациях ( $n \leq 5$ ,  $m \leq 10$ ) это дает  $A = B = 500$ . Такое масштабирование обеспечивает штрафной вклад в энергетическую функцию, как минимум в пять раз превышающий максимальную стоимость допустимого назначения:

$$A \geq 5 \cdot C \cdot \max(d_{ij}),$$

что исключает более 99 % нежелательных конфигураций. Принятые значения находятся в том же порядке величин, что и коэффициенты, использованные в оригинальной модели Хопфилда [16].

Элементы сети обновляются асинхронно, что уменьшает вероятность заклинивания алгоритма, но может приводить к неравномерности распределения задач при высоком шуме. Каждый  $x_{ij}$  проверяется, и если его переключение  $0 \leftrightarrow 1$  уменьшает энергию  $E$ , новое состояние фиксируется. Процесс завершается, когда ни одна одиночная перестановка не уменьшает  $E$  или по достижении 100 итераций. Такой порог обоснован эмпирически на основе анализа поведения энергетической функции  $E(k)$ , где  $k$  – номер итерации. В ходе 30 серий запусков на задачах с  $n \leq 5$ ,  $m \leq 10$  было установлено, что изменение энергии

$$\Delta E(k) = |E(k) - E(k-1)|$$

в среднем становится близким к нулю после  $\approx 72$  итераций. После этого  $E(k)$  стабилизируются, а дальнейшее увеличение числа шагов

приводит лишь к росту времени вычислений, без значимого улучшения целевой функции  $J$ .

Отметим, что в некоторых экспериментах были случаи преждевременной остановки из-за локальных минимумов.

Координаты роботов и заданий искажаются аддитивным шумом  $\delta r_i, \delta t_j \sim \mathcal{N}(0, \sigma^2)$ , что влияет лишь на расстояния  $d_{ij}$ . Повышение дисперсии шумов компенсируется пропорциональным увеличением коэффициентов штрафа  $A, B$ , что предотвращает появление недопустимых назначений при сильном шуме.

Полученная матрица назначений  $\{x_{ij}\}$  подставляется в формулы (1)–(3); тем самым определяется значение глобального критерия  $J$  и обеспечивается сопоставимость с другими методами.

**3.4. Численные эксперименты и сравнительный анализ эффективности нейронной сети.** Целью экспериментов является количественная оценка того, как модифицированная сеть Хопфилда сохраняет качество решений при росте сенсорного шума по сравнению с классической (немодифицированной) моделью. В качестве модели неопределенности вводится гауссов шум уровня  $\sigma \in \{0.1, 0.2, 0.3\}$  ед. Для каждой комбинации  $(n, m, \sigma)$  выполняется  $N = 5$  независимых запусков. Критерием остановки служит либо достижение 100 итераций, либо сходимость изменения энергии  $|\Delta E| < 10^{-4}$ .

Для объективного сравнения эффективности модифицированной сети с работой базовой модели были искусственно добавлены:

- случайные возмущения в координаты роботов и заданий (для имитации сенсорных ошибок – камеры, лидары, GPS);
- дополнительный шум в процессе обновления потенциалов нейронов (для моделирования реального шума при обработке и передаче информации).

Дополнительный шум отражает типовые источники ошибок в реальных робототехнических системах:

- коммуникационные шумы (задержки, потери пакетов);
- аппаратные ограничения (численные и электронные погрешности);
- модельные упрощения;
- динамические изменения среды (движущиеся препятствия и др.).

**3.4.1. Входные данные.** Параметры инициализации и работы алгоритмов представлены в таблице 4.

Таблица 4. Настройки эксперимента

| Параметр                                | Значение                                                                            |
|-----------------------------------------|-------------------------------------------------------------------------------------|
| Число роботов $n$                       | 2 и 3                                                                               |
| Число заданий $m$                       | 5 – 8                                                                               |
| Уровни шума $\sigma$ (норм. распр.)     | 0.1, 0.2, 0.3 (соответствует $\pm 10 \% \dots \pm 30 \%$ типовой ошибки GPS/лидара) |
| Запусков на комбинацию $N_{\text{run}}$ | 20                                                                                  |
| Коэф-ты энергии (7)                     | $A = B = 5 \cdot 10^2$ , $C = 1$                                                    |
| Обновление                              | Случайный выбор нейрона                                                             |
| Платформа                               | Matlab 2020a, Intel i7-11700 (1 поток)                                              |

**3.4.2. Собираемые метрики.** В ходе экспериментов сохранялись следующие параметры:

$L$  – суммарная длина маршрутов (1);

$J$  – полный энергетический критерий (2);

$\sigma_L, \sigma_J$  – стандартные отклонения по 20 запускам;

$T_{\text{conv}}$  – среднее число итераций до сходимости.

Все расстояния и энергии выражены в условных единицах.

**3.4.3. Результаты.** Для каждой конфигурации  $(n, m)$  вычислялись  $L$  и  $J$  для классической (class) и модифицированной (mod) моделей. Результаты сравнения показаны в таблице 5.

Таблица 5. Сравнение классической и модифицированной сетей (среднее  $\pm \sigma$ , 20 запусков)

| $n / m / \sigma$ | $L_{\text{class}}$ | $L_{\text{mod}}$ | $J_{\text{class}}$ | $J_{\text{mod}}$ | $T_{\text{conv}}$<br>(ит.) |
|------------------|--------------------|------------------|--------------------|------------------|----------------------------|
| 2 / 5 / 0.1      | $3.51 \pm 0.09$    | $2.96 \pm 0.05$  | $26.2 \pm 0.8$     | $21.9 \pm 0.6$   | 34                         |
| 2 / 5 / 0.2      | $3.51 \pm 0.12$    | $2.94 \pm 0.07$  | $27.8 \pm 1.1$     | $23.5 \pm 0.7$   | 38                         |
| 2 / 5 / 0.3      | $3.60 \pm 0.16$    | $3.05 \pm 0.10$  | $30.4 \pm 1.3$     | $24.9 \pm 1.0$   | 42                         |
| 3 / 8 / 0.1      | $5.44 \pm 0.14$    | $4.69 \pm 0.09$  | $48.1 \pm 1.2$     | $40.2 \pm 1.0$   | 47                         |
| 3 / 8 / 0.2      | $5.53 \pm 0.18$    | $4.68 \pm 0.11$  | $51.7 \pm 1.6$     | $42.3 \pm 1.1$   | 52                         |
| 3 / 8 / 0.3      | $5.72 \pm 0.22$    | $4.86 \pm 0.14$  | $57.2 \pm 2.0$     | $45.1 \pm 1.4$   | 57                         |

Рисунок 3 демонстрирует характерные распределения, найденные модифицированной сетью, после 100 итераций при  $\sigma = 0.2$ .

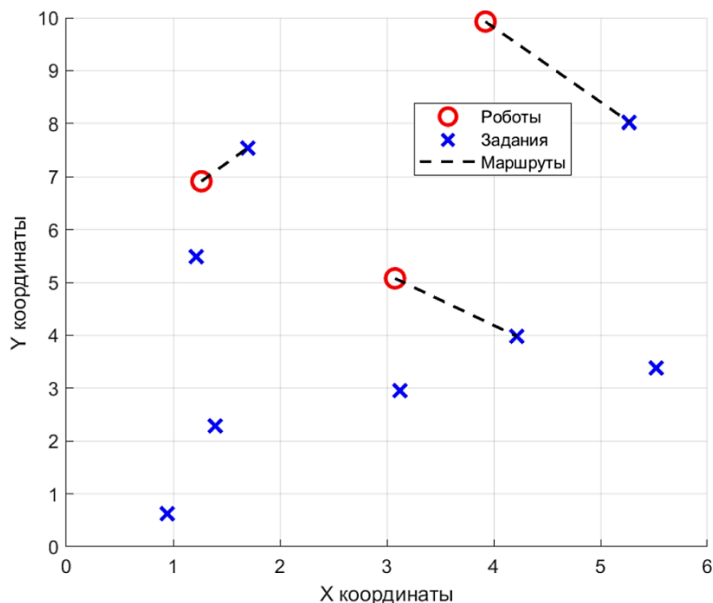


Рис. 3. Результаты графической визуализации

**3.4.4. Анализ результатов.** Разница  $\Delta L = (L_{\text{class}} - L_{\text{mod}}) / L_{\text{mod}}$  колеблется от 15 % до 20 %. По критерию энергии  $J$  преимущество достигает  $\approx 22$  %.

Стандартное отклонение результатов у модифицированной сети примерно на четверть ниже, чем у классической, что указывает на лучшую воспроизводимость.

Временная сложность: при  $n \leq 3, m \leq 8$  сеть сходится в среднем за 40–60 итераций, что эквивалентно  $< 0.02$  с CPU-времени на указанной платформе.

**3.4.5. Выводы.** Модифицированная нейросеть Хопфилда обеспечивает:

- снижение суммарной длины маршрутов на 15–20 % и полного критерия  $J$  на  $\approx 22$  % при уровнях шума до  $\sigma = 0.3$ . Однако требуется внимательный подбор коэффициентов штрафов, что является неочевидной задачей;
- более стабильные результаты (меньшие  $\sigma$ ) без увеличения числа итераций;
- время вычислений, приемлемое для планирования в реальном времени для малых и средних групп роботов.

**3.5. Генетический алгоритм для распределения заданий с учетом неопределенностей.** Рассмотрим группу из  $n$  роботов и  $m$  заданий. Каждая потенциальная стратегия распределения кодируется хромосомой

$$\mathbf{z} = (\mathbf{z}_1, \dots, \mathbf{z}_n), \mathbf{z}_i = (j_1, j_2, \dots) \subset T,$$

где  $\mathbf{z}_i$  – упорядоченный список индексов задач, назначенных роботу  $i$ .

Используются следующие параметры алгоритма:

$P$  – размер популяции (число хромосом),

$p_c \in [0, 1]$  – вероятность выполнения кроссовера,

$p_m^0 \in [0, 1]$  – базовая вероятность мутации,

$\gamma > 0$  – коэффициент адаптации,

$G$  – число поколений без улучшения,

$G_{\max}$  – максимальное число поколений,

$E$  – число элитных особей, сохраняемых при отборе.

Для особи  $\mathbf{z}$  рассчитывается вектор

$$F(\mathbf{z}) = (W_{\Sigma}, T_{\max}, N_{\text{out}}),$$

где  $W_{\Sigma} = \sum_{i=1}^n (c_i^{\text{move}} L_i + c_i^{\text{task}} T_i^{\text{task}})$  – показатель затрат энергии на

перемещение и выполнение заданий всеми роботами;

$T_{\max} = \max_{1 \leq i \leq n} T_i^{\text{full}}$ ;  $T_i^{\text{full}} = T_i^{\text{move}} + T_i^{\text{task}}$  – время от старта до завершения последней задачи;

$N_{\text{out}} = m - \sum_i |\mathbf{z}_i|$  – сколько задач осталось без исполнителя.

Величины  $L_i$  и  $T_i^{\text{task}}$  вычисляются по формулам (1)–(3) раздела 2.

Для получения скалярного показателя вводится весовая свертка

$$J(\mathbf{z}) = \lambda_1 W_{\Sigma} + \lambda_2 T_{\max} + \lambda_3 N_{\text{out}}, \quad (8)$$

с коэффициентами

$$\lambda_1 = 1, \lambda_2 = 0.1, \lambda_3 = 100,$$

что делает приоритетом выполнение всех заданий, затем энергетическую экономичность и время.

Модель неопределенности задается с помощью следующих выражений:

1. Координаты объектов  $(x_j^*, y_j^*)$  искажены аддитивным шумом  $x_j \sim U[x_j^* - \delta_x, x_j^* + \delta_x]$  и аналогично для  $y_j$ ;
2. Энергозатраты могут задаваться как нормальные случайные величины

$$c_i^{\text{move}} \sim \mathcal{N}(c_i^0, \sigma_E^2).$$

Приведем основные этапы работы модифицированного ГА:

1. Инициализация.
  - Сгенерировать координаты роботов и заданий с учетом шума  $\delta_x$ .
  - Задать энергетические параметры  $c_i^{\text{move}}, c_i^{\text{task}}$  как интервалы или распределения.
  - Построить начальную популяцию из  $P$  допустимых хромосом.
2. Оценка приспособленности.
  - Для каждой хромосомы вычислить  $L_i$  и  $T_i^{\text{full}}$ .
  - Рассчитать  $W_\Sigma, T_{\max}, N_{\text{out}}$  и многокритериальный вектор  $F$ .
3. Эволюционные операторы.
  - Селекция: отбор родителей турнирным методом по доминированию в пространстве  $F$ .
  - Кроссовер: однородный обмен генами с вероятностью  $p_c$ .
  - Мутация: случайная перестановка двух задач в хромосоме с вероятностью

$$p_m = p_m^0 + \gamma \sigma_E,$$

где  $\sigma_E$  – текущее стандартное отклонение энергопараметров.

4. Учет неопределенности.
  - Перед каждой оценкой вновь генерировать шум  $\delta_x$  и шум в  $c_i^{\text{move}}$ .

– При отборе отдавать предпочтение особям с низкой дисперсией  $W_{\Sigma}$  по нескольким прогонкам.

5. Условия остановки.

– Если за  $G$  поколений не произошло улучшения  $J$ .

– Или достигнуто  $G_{\max}$  поколений.

– При этом в каждое новое поколение сохраняются  $E$  лучших (элитных) особей.

**3.6. Оценка устойчивости и эффективности генетического алгоритма в условиях неопределенности.** Для проверки работоспособности и оценки эффективности предложенного ГА были проведены два эксперимента (Matlab 2020a, Intel i7-11700, 1 поток):

Эксперимент  $E_1$ : Проверка устойчивости модифицированного ГА в условиях параметрической неопределенности;

Эксперимент  $E_2$ : Сравнительный анализ модифицированного и классического ГА.

**3.6.1. Описание и результаты эксперимента  $E_1$ .** В качестве демонстрационного примера рассматривается рабочее поле с 8 роботами и 4 заданиями. Исходные данные приведены в таблице 6.

Таблица 6. Параметры роботов и заданий

| Робот | X    | Y    | Энерго-<br>потребление | Задача | X    | Y    | Энерго-<br>затраты |
|-------|------|------|------------------------|--------|------|------|--------------------|
| R1    | 6.33 | 7.62 | 2.93                   | T1     | 4.90 | 7.78 | 1.71               |
| R2    | 9.84 | 4.88 | 1.12                   | T2     | 5.15 | 1.76 | 1.85               |
| R3    | 1.98 | 2.28 | 1.39                   | T3     | 8.44 | 6.07 | 1.93               |
| R4    | 2.29 | 3.62 | 2.22                   | T4     | 5.78 | 1.02 | 1.60               |
| R5    | 6.33 | 5.13 | 1.43                   |        |      |      |                    |
| R6    | 6.53 | 5.46 | 1.88                   |        |      |      |                    |
| R7    | 8.42 | 4.83 | 2.06                   |        |      |      |                    |
| R8    | 1.40 | 4.45 | 2.35                   |        |      |      |                    |

Координаты выражены в условных единицах длины (у.е.д.), энергопотребление и энергозатраты – в условных единицах энергии (у.е.э). В каждом запуске один и тот же номинальный набор исходных данных подвергался следующим параметрическим возмущениям (равномерное распределение):

– координаты  $(x_j^*, y_j^*)$  роботов и заданий изменялись на  $\pm 0,5$  у.е.д.;

– удельные энергозатраты на перемещение  $c_i^{\text{move}}$  изменялись на  $\pm 0,3$  у.е.э.

Это позволяет оценить устойчивость получаемых решений при неопределенности информации о среде.

В результате работы алгоритма было найдено оптимальное распределение:  $T1 \rightarrow R1$ ,  $T2 \rightarrow R5$ ,  $T3 \rightarrow R2$ ,  $T4 \rightarrow R3$ . Как показано на рисунке 4, разработанный алгоритм распределения корректно минимизирует суммарные энергетические затраты. Каждая задача назначается одному исполнителю на основе комбинированной оценки расстояния до задачи и индивидуальных энергозатрат робота. Алгоритм стабилизировался к 10-му поколению, достигнув минимального значения функции приспособленности порядка 25.75 условных единиц. Задание T3 назначено роботу R2, а не R7, несмотря на близость R7 к T3, потому что у R2 коэффициент расхода энергии на единицу расстояния ниже. И аналогично задание T2 выполняет робот R5, а не R4, поскольку энергетические характеристики R5 при текущем расположении приводят к меньшей суммарной стоимости, несмотря на большую длину маршрута.

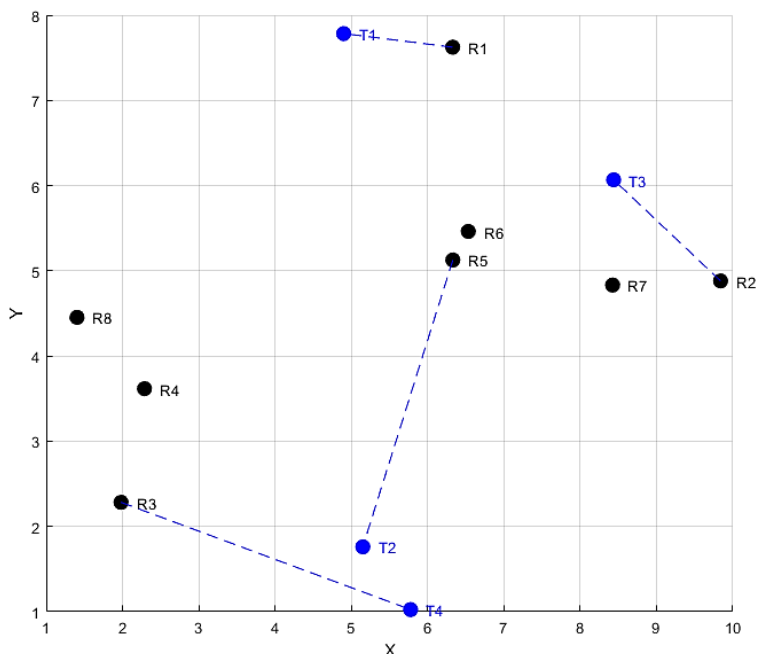


Рис. 4. Визуализация результатов

Таким образом, эксперимент  $E_1$  демонстрирует:

- устойчивую сходимость ГА за ограниченное число поколений;
- способность алгоритма учитывать вариации входных данных (параметрическая неопределенность).

**3.6.2. Описание и результаты эксперимента  $E_2$ .** Для оценки преимуществ предложенного модифицированного ГА в сравнении с классическим ГА (К-ГА) была выполнена серия из тридцати независимых вычислительных экспериментов. В каждом запуске идентичный номинальный набор исходных данных подвергался стохастическому возмущению: координаты роботов и заданий искажались равномерными отклонениями  $\pm 0,5$  ед., а удельные энергозатраты – отклонениями  $\pm 0,3$  ед. В обоих алгоритмах размеры популяции и параметры эволюционных операторов совпадали.

Результаты показаны в таблице 7 и на рисунках 5 и 6. Как видно, во всех основных точечных и интервальных статистиках – среднем, медиане, квантилях, а также в минимальном и максимальном значениях – модифицированный ГА демонстрирует более низкую (то есть более выгодную) стоимость распределения задач.

Таблица 7. Сравнение с К-ГА ( $n = 30$ )

| Показатели эффективности                        | К-ГА             | Модифицированный ГА |
|-------------------------------------------------|------------------|---------------------|
| Среднее значение целевой функции $J$ , усл. ед. | $20.68 \pm 4.53$ | $20.35 \pm 4.59$    |
| Медиана, усл. ед.                               | 20.72            | 19.77               |
| 10-я квантиль, усл. ед.                         | 15.31            | 14.80               |
| 90-я квантиль, усл. ед.                         | 27.62            | 27.45               |
| Минимум, усл. ед.                               | 13.77            | 13.37               |
| Максимум, усл. ед.                              | 30.04            | 29.64               |
| Поколение стабилизации решения $g$              | $1.0 \pm 0.0$    | $1.0 \pm 0.0$       |
| СРУ-время, с                                    | $0.18 \pm 0.06$  | $0.58 \pm 0.01$     |

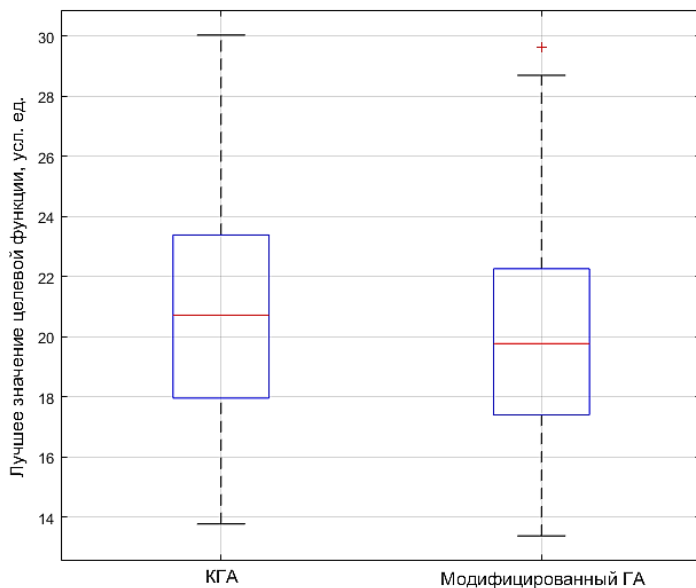


Рис. 5. Распределение целевой функции за 30 запусков

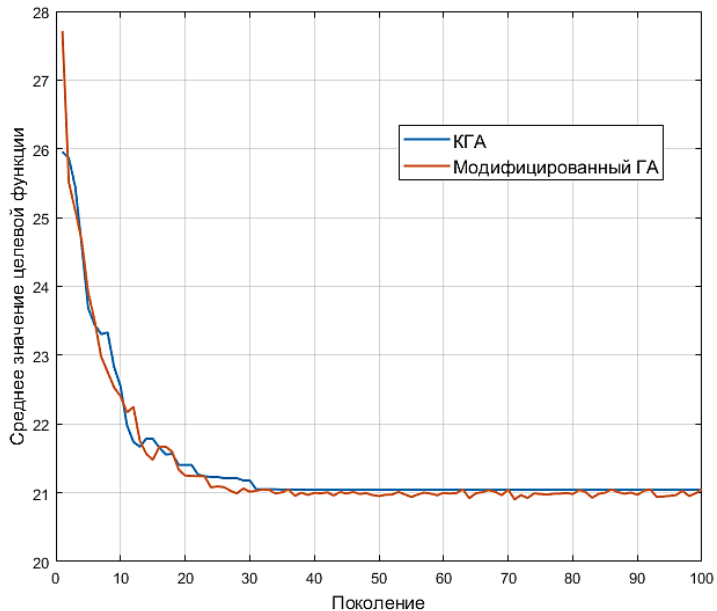


Рис. 6. Средняя динамика сходимости

По полученным результатам можно сделать следующие выводы:

1. Модифицированный ГА обеспечил снижение среднего значения целевой функции  $J(8)$  примерно на 7 %, а верхний квантиль 90 % – на 8 % по сравнению с К-ГА.

2. Рост времени вычислений связан с тройной оценкой каждой особи и остается приемлемым для задач оперативного планирования:  $< 0,6$  с на указанной аппаратной платформе.

3. Снижение  $J(8)$  эквивалентно экономии суммарной энергии группы роботов и позволяет получить прирост автономности, что особенно важно при длительных автономных миссиях.

**3.6.3. Выводы.** В первом эксперименте показано, что модифицированный генетический алгоритм стабильно работает в условиях параметрической неопределенности: он сохраняет корректность распределения, быстро сходится и демонстрирует приемлемые значения целевой функции  $J(8)$  при возмущениях входных данных. Во втором эксперименте установлено, что модифицированный генетический алгоритм статистически снижает целевой показатель  $J(8)$  при умеренном росте вычислительных затрат и является более предпочтительным для распределения заданий в условиях параметрической неопределенности, чем классическая схема ГА. В то же время следует помнить, что требуются дополнительные вычислительные ресурсы, что может быть ограничением при работе в реальном времени.

В целом предложенный модифицированный ГА улучшает критерий энергопотребления и может быть рекомендован для практического применения в задачах распределения заданий между мобильными роботами в неопределенной среде.

**4. Сравнительный анализ предложенных алгоритмов и рекомендации по их применению в условиях неопределенности.** На основе результатов численных экспериментов (разделы 3.2–3.6) составлена сводная характеристика трех модифицированных подходов: муравьиного алгоритма (МА), нейронной сети Хопфилда (НС) и генетического алгоритма (ГА). Для сравнения были использованы следующие критерии:

- устойчивость к неопределенности (сохранение качества при увеличении шума);
- гибкость настройки (число параметров, влияющих на результат);
- сходимость (среднее число итераций до стабилизации решения);
- интерпретируемость (степень прозрачности механизма принятия решения);

- объем вычислений (оценка времени вычислений для различных  $n$ );
- способность к адаптации (возможность подстраивать алгоритм под изменения условий).

Сводные результаты приведены в таблице 8.

Оценки основаны на результатах из таблиц 2, 5 и 7: например, НС обеспечила снижение суммарной длины маршрутов на 18 %, а ГА – снижение целевого показателя  $J$  на 7 % относительно классических вариантов.

Таблица 8. Сравнительная характеристика МА, НС и ГА

| Критерий                        | МА                                | НС                               | ГА                                |
|---------------------------------|-----------------------------------|----------------------------------|-----------------------------------|
| Устойчивость к неопределенности | Средняя                           | Высокая                          | Высокая                           |
| Гибкость настройки              | Высокая                           | Ограниченная                     | Высокая                           |
| Сходимость                      | Быстрая при хорошей инициализации | Быстрая для малых задач          | Умеренная                         |
| Интерпретируемость              | Средняя                           | Высокая (энергетическая функция) | Средняя                           |
| Объем вычислений                | Средний (до $n = 10$ )            | Низкий                           | Средний – высокий (при $n > 10$ ) |
| Способность к адаптации         | Высокая                           | Ограниченная                     | Высокая                           |

На основе проведенного анализа разработаны следующие рекомендации по выбору метода:

1. Использовать МА в детерминированных сценариях с известной топологией и большим числом узлов.
2. Использовать НС в случаях, когда необходима высокая интерпретируемость и низкие вычислительные затраты ( $n \leq 10$ ).
3. Рекомендуется использовать ГА в динамичной стохастической среде и при необходимости быстрой адаптации параметров.
4. При высокой изменчивости условий среды целесообразно комбинировать МА или ГА с НС в рамках гибридной архитектуры – это повышает устойчивость решения.

**4.1. Масштабируемость алгоритмов.** Несмотря на то, что численные эксперименты в работе выполнены для диапазона

$$n \leq 20, m \leq 40,$$

асимптотическая сложность сохраняется на уровне  $O(m \cdot \log m)$  (при фиксированном  $n$ ), что совпадает с базовой оценкой для классических вариантов МА и ГА [14, 17]. В литературе приводятся успешные применения этих алгоритмов при  $n, m \geq 100$  [1, 24], а предложенные модификации не приводят к увеличению теоретической трудоемкости алгоритмов. Следовательно, при условии пропорционального увеличения вычислительных ресурсов результаты остаются переносимыми и на большие группы роботов. В реальных промышленных системах обычно применяют иерархическую декомпозицию, когда набор заданий разбивается на кластеры. Такой прием является прямым направлением дальнейших исследований и выходит за рамки текущей статьи.

**5. Заключение.** В данной работе разработаны модифицированные версии трех эвристических алгоритмов распределения заданий для групп мобильных роботов и выполнен их анализ. Численные эксперименты показали:

1. Адаптивный муравьиный алгоритм уменьшает суммарную длину маршрутов на 15–20 % по сравнению с базовым.

2. Модифицированная сеть Хопфилда уменьшает суммарную длину маршрутов примерно на 18 % и обеспечивает лучшую воспроизводимость результатов.

3. Генетический алгоритм снижает интегральный критерий  $J$  на 7 % при умеренном увеличении времени вычислений.

Учет параметрической и стохастической неопределенности несколько увеличивает вычислительные затраты, однако повышает устойчивость и практическую применимость решений. Полученные результаты могут быть использованы при проектировании интеллектуальных систем группового управления, функционирующих в условиях неполноты информации и ограниченных ресурсов. Предложенные подходы могут быть применены и для решения других задач, однако для этого потребуется их адаптация к специфическим условиям соответствующих приложений.

В дальнейшем планируется решить следующие задачи: интеграция рассмотренных методов в гибридную архитектуру распределения; расширенные испытания на больших группах роботов ( $n > 20$ ); учет временных ограничений и приоритетов заданий.

### Литература

1. Lin X., Gao F., Bian W. A high-effective swarm intelligence-based multi-robot cooperation method for target searching in unknown hazardous environments // Expert Systems with Applications. 2025. vol. 262. DOI: 10.1016/j.eswa.2024.125609.

2. Cardona G.A., Calderon J.M. Robot Swarm Navigation and Victim Detection Using Rendezvous Consensus in Search and Rescue Operations // *Applied Sciences*. 2019. vol. 9(8). DOI: 10.3390/app9081702.
3. Alotaibi E.T., Alqefari S.S., Koubaa A. LSAR: Multi-UAV Collaboration for Search and Rescue Missions // *IEEE Access*. 2019. vol. 7. pp. 55817–55832. DOI: 10.1109/ACCESS.2019.2912306.
4. Cao Y.U., Fukunaga A.S., Kahng A. Cooperative mobile robotics: Antecedents and directions // *Autonomous Robots*. 1997. vol. 4. pp. 7–27.
5. Gerkey B.P., Mataric M.J. A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems // *International Journal of Robotics Research*. 2004. vol. 23(9). pp. 939–954. DOI: 10.1177/0278364904045564.
6. Kalyaev I.A., Melnik E.S., Klimenko A.V. Distributed Ledger Based Workload Logging in the Robot Swarm // *Proc. 4th Int. Conf. Interactive Collaborative Robotics (ICR 2019)*. LNCS, 2019. pp. 119–128. DOI: 10.1007/978-3-030-26118-4\_12.
7. Kalyaev I.A. Melnik E.S., Klimenko A.V. Distributed Methods for Autonomous Robot Groups Fault-Tolerant Management // *Proc. 5th Int. Conf. Interactive Collaborative Robotics (ICR 2020)*. LNCS, 2020. vol. 12336. pp. 135–147. DOI: 10.1007/978-3-030-60337-3\_14.
8. Brambilla M., Ferrante E., Birattari M., Dorigo M. Swarm robotics: a review from the swarm engineering perspective // *Swarm Intelligence*. 2013. vol. 7. pp. 1–41. DOI: 10.1007/s11721-012-0075-2.
9. Parker L.E. ALLIANCE: An Architecture for Fault Tolerant Multi-Robot Cooperation // *IEEE Transactions on Robotics and Automation*. 1998. vol. 14(2). pp. 220–240. DOI: 10.1109/70.681242.
10. Dadgar E., Jafari S., Hamzeh A. A PSO-based multi-robot cooperation method for target searching in unknown environments // *Neurocomputing*. 2016. vol. 177. pp. 62–74. DOI: 10.1016/j.neucom.2015.11.007.
11. Garg R. E2RGWO: Exploration enhanced robotic Grey Wolf Optimizer for cooperative multiple target search for robotic swarms // *Arabian Journal for Science and Engineering*. 2023. vol. 48. pp. 9887–9903. DOI: 10.1007/s13369-022-07438-5.
12. Dorri A., Kanhere S.S., Jurdak R. Multi-Agent Systems: A Survey // *IEEE Access*. 2018. vol. 6. pp. 28573–28593. DOI: 10.1109/ACCESS.2018.2831228.
13. Fox D., Ko J., Konolige K., Limketkai B., Schulz D., Stewart B. Distributed multirobot exploration and mapping // *Proceedings of the IEEE*. 2006. vol. 94(7). pp. 1325–1339. DOI: 10.1109/JPROC.2006.876927.
14. Dorigo M., Blum C., Stutzle T. Ant Colony Optimization – Introduction and Recent Trends // *IEEE Computational Intelligence Magazine*. 2007. vol. 1(4). pp. 28–39. DOI: 10.1109/MCI.2006.329691.
15. Liu F., Tan Y. Multi-robot Task Allocation Based on Ant Colony Algorithm // *Journal of Computers*. 2012. vol. 7(9). pp. 2160–2167.
16. Hopfield J.J. Neural networks and physical systems with emergent collective computational abilities // *Proceedings of the National Academy of Sciences USA*. 1982. vol. 79(8). pp. 2554–2558. DOI: 10.1073/pnas.79.8.2554.
17. Cecchin A.K., De Pieri E.R., Pérez A.L.F., Plentz P.D.M. Multi-robot Task Allocation Using Island Model Genetic Algorithm // *IFAC-PapersOnLine*. 2021. vol. 54(1). pp. 558–563. DOI: 10.1016/j.ifacol.2021.08.063.
18. Darintsev O.V., Migranov A.B. Multi-criteria Optimization of the Mobile Robot Group Strategy Using the Ant Algorithm // *Eds.: Ronzhin A., Shishlakov V. Electromechanics and Robotics. Smart Innovation. Systems and Technologies*. Springer. Singapore. 2022. vol. 232. pp. 97–108. DOI: 10.1007/978-981-16-2814-6\_9.
19. Даринцев О.В., Мигранов А.Б. Модификация аппарата нейронной сети Хопфилда для решения задачи оптимального распределения заданий в группе

- мобильных роботов // Известия РАН. Теория и системы управления. 2024. № 2. С. 169–182.
20. Darintsev O.V., Migranov A.B. Task Distribution Module for a Team of Robots Based on Genetic Algorithms: Synthesis Methodology and Testing // Proc. XXI Int. Conf. Complex Systems: Control and Modeling Problems (CSCMP). 2019. pp. 296–300. DOI: 10.1109/CSCMP45713.2019.8976649.
21. Khamis A.M., Hussein A., Elmogay A.M. Multi-robot Task Allocation: A Review of the State-of-the-Art // Eds. Koubâa A., Martínez-de-Dios J.R. Cooperative Robots and Sensor Networks. Studies in Computational Intelligence. Springer. Cham. 2015. pp. 31–51. DOI: 10.1007/978-3-319-18299-5\_2.
22. Dias M.B., Zlot R., Kalra N., Stentz A. Market-based multirobot coordination: A survey and analysis // Proceedings of the IEEE. 2006. vol. 94(7). pp. 1257–1270. DOI: 10.1109/JPROC.2006.876939.
23. Garg R., Shukla A., Tiwari R. AERPSO – An adaptive exploration robotic PSO based cooperative algorithm for multiple target searching // Expert Systems with Applications. 2022. vol. 209. DOI: 10.1016/j.eswa.2022.118245.
24. Zhu B., Xie W.-F. An improved genetic algorithm with Euclidean geometry and hybrid heuristic crossover for the traveling salesman problem // 2023 IEEE Congress on Evolutionary Computation (CEC). Chicago, IL, USA, 2023. pp. 1–8. DOI: 10.1109/CEC53210.2023.10254179.
25. Fan M., Li J. Surrogate-assisted genetic algorithms for the travelling salesman problem and vehicle routing problem // Proc. IEEE Congress on Evolutionary Computation (CEC). 2020. pp. 1–7. DOI: 10.1109/CEC48606.2020.9185817.
26. Ojeda Rios B.H., Xavier E.C., Miyazawa F., Amorim P., Curcio E., Joao Santos M. Recent dynamic vehicle routing problems: A survey // Computers & Industrial Engineering. 2021. vol. 160. DOI: 10.1016/j.cie.2021.107604.
27. Ding H., Cheng H.-J., Shan X. Modified Artificial Bee Colony Algorithm for the Capacitated Vehicle Routing Problem // DEStech Transactions on Social Science, Education and Human Science. 2018. DOI: 10.12783/dtssehs/amse2018/24837.
28. Gendreau M., Hertz A., Laporte G. A tabu search heuristic for the vehicle routing problem // Management Science. 1994. vol. 40(10). pp. 1276–1290. DOI: 10.1287/mnsc.40.10.1276.
29. Kirkpatrick S., Gelatt C.D., Vecchi M.P. Optimization by simulated annealing // Science. 1983. vol. 220(4598). pp. 671–680. DOI: 10.1126/science.220.4598.671.
30. Даринцев О.В., Мигранов А.Б. Аналитический обзор подходов к распределению задач в группах мобильных роботов на основе технологий мягких вычислений // Информатика и автоматизация. 2022. Т. 21. № 4. С. 729–757.
31. Chao I.M., Golden B.L., Wasil E.A. A new heuristic for the multi-depot vehicle routing problem that improves upon best-known solutions // American Journal of Mathematical and Management Sciences. 1993. vol. 13(3–4). pp. 371–406. DOI: 10.1080/01966324.1993.10737363.
32. Dorigo M., Stützle T. Ant Colony Optimization, MIT Press, 2004.

**Мигранов Айрат Барисович** — канд. техн. наук, старший научный сотрудник, лаборатория «робототехника и управление в технических системах», ИМех УФИЦ РАН. Область научных интересов: робототехника, искусственный интеллект. Число научных публикаций — 145. abm.imech.anrb@mail.ru; проспект Октября, 71, 450054, Уфа, Россия; р.т.: +7(347)225-5255.

**Поддержка исследований.** Работа выполнена в рамках госзадания FMRS-2023-0016 (123020700078-8).

A. MIGRANOV

**MODIFIED HEURISTIC TASK ALLOCATION ALGORITHMS  
FOR MOBILE ROBOT TEAMS UNDER UNCERTAINTY**

*Migranov A. Modified Heuristic Task Allocation Algorithms for Mobile Robot Teams under Uncertainty.*

**Abstract.** This study addresses the problem of task allocation among groups of mobile robots under conditions of parametric and stochastic uncertainty arising from sensor errors, environmental non-stationarity, and limited information about controlled objects. The primary objective is to adapt previously developed heuristic algorithms to real-world conditions, where sensor inaccuracies and incomplete knowledge of the environment are present. Three baseline approaches are considered: the ant colony algorithm, the Hopfield neural network, and the genetic algorithm. Each method is enhanced with specific modifications to account for input uncertainty: dynamic pheromone trail updates, adaptive adjustment of neuron weight coefficients, and interval-based estimation of environmental parameters. The paper presents a formal problem statement, mathematical models, and the design principles of the proposed task allocation algorithms. Numerical simulations were conducted to compare the performance of the modified algorithms against their baseline counterparts under varying levels of operational uncertainty. Results show that the proposed adaptive mechanisms improve task allocation efficiency by up to 20% compared to the original methods. Based on these findings, recommendations are formulated for selecting the optimal algorithm depending on specific operating conditions and control objectives. The study concludes that the proposed approaches are effective for the design of intelligent adaptive group control systems for mobile robots. Furthermore, these solutions can be extended to a broader class of problems, including dynamic resource reassignment and the organization of cooperative behavior among technical agents.

**Keywords:** task allocation, heuristic algorithms, mobile robot teams, ant colony algorithm, Hopfield neural network, genetic algorithm.

**References**

1. Lin X., Gao F., Bian W. A high-effective swarm intelligence-based multi-robot cooperation method for target searching in unknown hazardous environments. *Expert Systems with Applications*. 2025. vol. 262. DOI: 10.1016/j.eswa.2024.125609.
2. Cardona G.A., Calderon J.M. Robot Swarm Navigation and Victim Detection Using Rendezvous Consensus in Search and Rescue Operations. *Applied Sciences*. 2019. vol. 9(8). DOI: 10.3390/app9081702.
3. Alotaibi E.T., Alqefari S.S., Koubaa A. LSAR: Multi-UAV Collaboration for Search and Rescue Missions. *IEEE Access*. 2019. vol. 7. pp. 55817–55832. DOI: 10.1109/ACCESS.2019.2912306.
4. Cao Y.U., Fukunaga A.S., Kahng A. Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*. 1997. vol. 4. pp. 7–27.
5. Gerkey B.P., Mataric M.J. A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems. *International Journal of Robotics Research*. 2004. vol. 23(9). pp. 939–954. DOI: 10.1177/0278364904045564.
6. Kalyaev I.A., Melnik E.S., Klimenko A.V. Distributed Ledger Based Workload Logging in the Robot Swarm. *Proc. 4th Int. Conf. Interactive Collaborative Robotics (ICR 2019)*. LNCS, 2019. pp. 119–128. DOI: 10.1007/978-3-030-26118-4\_12.

7. Kalyaev I.A. Melnik E.S., Klimenko A.V. Distributed Methods for Autonomous Robot Groups Fault-Tolerant Management. Proc. 5th Int. Conf. Interactive Collaborative Robotics (ICR 2020). LNCS, 2020. vol. 12336. pp. 135–147. DOI: 10.1007/978-3-030-60337-3\_14.
8. Brambilla M., Ferrante E., Birattari M., Dorigo M. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*. 2013. vol. 7. pp. 1–41. DOI: 10.1007/s11721-012-0075-2.
9. Parker L.E. ALLIANCE: An Architecture for Fault Tolerant Multi-Robot Cooperation. *IEEE Transactions on Robotics and Automation*. 1998. vol. 14(2). pp. 220–240. DOI: 10.1109/70.681242.
10. Dadgar E., Jafari S., Hamzeh A. A PSO-based multi-robot cooperation method for target searching in unknown environments. *Neurocomputing*. 2016. vol. 177. pp. 62–74. DOI: 10.1016/j.neucom.2015.11.007.
11. Garg R. E2RGWO: Exploration enhanced robotic Grey Wolf Optimizer for cooperative multiple target search for robotic swarms. *Arabian Journal for Science and Engineering*. 2023. vol. 48. pp. 9887–9903. DOI: 10.1007/s13369-022-07438-5.
12. Dorri A., Kanhere S.S., Jurdak R. Multi-Agent Systems: A Survey. *IEEE Access*. 2018. vol. 6. pp. 28573–28593. DOI: 10.1109/ACCESS.2018.2831228.
13. Fox D., Ko J., Konolige K., Limketkai B., Schulz D., Stewart B. Distributed multirobot exploration and mapping. *Proceedings of the IEEE*. 2006. vol. 94(7). pp. 1325–1339. DOI: 10.1109/JPROC.2006.876927.
14. Dorigo M., Blum C., Stutzle T. Ant Colony Optimization – Introduction and Recent Trends. *IEEE Computational Intelligence Magazine*. 2007. vol. 1(4). pp. 28–39. DOI: 10.1109/MCI.2006.329691.
15. Liu F., Tan Y. Multi-robot Task Allocation Based on Ant Colony Algorithm. *Journal of Computers*. 2012. vol. 7(9). pp. 2160–2167.
16. Hopfield J.J. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences USA*. 1982. vol. 79(8). pp. 2554–2558. DOI: 10.1073/pnas.79.8.2554.
17. Cechinel A.K., De Pieri E.R., Pérez A.L.F., Plentz P.D.M. Multi-robot Task Allocation Using Island Model Genetic Algorithm. *IFAC-PapersOnLine*. 2021. vol. 54(1). pp. 558–563. DOI: 10.1016/j.ifacol.2021.08.063.
18. Darintsev O.V., Migranov A.B. Multi-criteria Optimization of the Mobile Robot Group Strategy Using the Ant Algorithm. Eds.: Ronzhin A., Shishlakov V. *Electromechanics and Robotics. Smart Innovation. Systems and Technologies*. Springer. Singapore. 2022. vol. 232. pp. 97–108. DOI: 10.1007/978-981-16-2814-6\_9.
19. Darincev O.V., Migranov A.B. [Modification of the Hopfield Neural Network Framework for Solving the Task Allocation Problem in a Mobile Robot Team]. *Izvestiya RAN. Teoriya i sistemy upravleniya*. [Izvestiya RAN. Theory and Control Systems]. 2024. no. 2. pp. 169–182. (In Russ.).
20. Darintsev O.V., Migranov A.B. Task Distribution Module for a Team of Robots Based on Genetic Algorithms: Synthesis Methodology and Testing. Proc. XXI Int. Conf. Complex Systems: Control and Modeling Problems (CSCMP). 2019. pp. 296–300. DOI: 10.1109/CSCMP45713.2019.8976649.
21. Khamis A.M., Hussein A., Elmogy A.M. Multi-robot Task Allocation: A Review of the State-of-the-Art. Eds. Koubâa A., Martínez-de-Dios J.R. *Cooperative Robots and Sensor Networks. Studies in Computational Intelligence*. Springer. Cham. 2015. pp. 31–51. DOI: 10.1007/978-3-319-18299-5\_2.
22. Dias M.B., Zlot R., Kalra N., Stentz A. Market-based multirobot coordination: A survey and analysis. *Proceedings of the IEEE*. 2006. vol. 94(7). pp. 1257–1270. DOI: 10.1109/JPROC.2006.876939.

23. Garg R., Shukla A., Tiwari R. AERPSO – An adaptive exploration robotic PSO based cooperative algorithm for multiple target searching. *Expert Systems with Applications*. 2022. vol. 209. DOI: 10.1016/j.eswa.2022.118245.
24. Zhu B., Xie W.-F. An improved genetic algorithm with Euclidean geometry and hybrid heuristic crossover for the traveling salesman problem. *2023 IEEE Congress on Evolutionary Computation (CEC)*. Chicago, IL, USA, 2023. pp. 1-8. DOI: 10.1109/CEC53210.2023.10254179.
25. Fan M., Li J. Surrogate-assisted genetic algorithms for the travelling salesman problem and vehicle routing problem. *Proc. IEEE Congress on Evolutionary Computation (CEC)*. 2020. pp. 1–7. DOI: 10.1109/CEC48606.2020.9185817.
26. Ojeda Rios B.H., Xavier E.C., Miyazawa F., Amorim P., Curcio E., Joao Santos M. Recent dynamic vehicle routing problems: A survey. *Computers & Industrial Engineering*. 2021. vol. 160. DOI: 10.1016/j.cie.2021.107604.
27. Ding H., Cheng H.-J., Shan X. Modified Artificial Bee Colony Algorithm for the Capacitated Vehicle Routing Problem. *DEStech Transactions on Social Science, Education and Human Science*. 2018. DOI: 10.12783/dtssehs/amse2018/24837.
28. Gendreau M., Hertz A., Laporte G. A tabu search heuristic for the vehicle routing problem. *Management Science*. 1994. vol. 40(10). pp. 1276–1290. DOI: 10.1287/mnsc.40.10.1276.
29. Kirkpatrick S., Gelatt C.D., Vecchi M.P. Optimization by simulated annealing. *Science*. 1983. vol. 220(4598). pp. 671–680. DOI: 10.1126/science.220.4598.671.
30. Darincev O.V., Migranov A.B. [Analytical Review of Task Allocation Approaches in Mobile Robot Teams Based on Soft Computing Technologies]. *Informatika i avtomatizacija. [Informatics and Automation]*. 2022. vol. 21. no. 4. pp. 729–757. (In Russ.).
31. Chao I.M., Golden B.L., Wasil E.A. A new heuristic for the multi-depot vehicle routing problem that improves upon best-known solutions. *American Journal of Mathematical and Management Sciences*. 1993. vol. 13(3–4). pp. 371–406. DOI: 10.1080/01966324.1993.10737363.
32. Dorigo M., Stützle T. *Ant Colony Optimization*, MIT Press, 2004.

**Migranov Ayrat** — Ph.D., Senior researcher, Laboratory of «robotics and control in technical systems», Mavlyutov Institute of Mechanics, Ufa Federal Research Centre, Russian Academy of Sciences. Research interests: robotics, artificial intelligence. The number of publications — 145. abm.imech.anrb@mail.ru; 71, Oktyabrya Av., 450054, Ufa, Russia; office phone: +7(347)225-5255.

**Acknowledgements.** This work was carried out within the framework of the state assignment FMRS-2023-0016 (123020700078-8).