

О.В. ДАРИНЦЕВ, А.Б. МИГРАНОВ  
**АНАЛИТИЧЕСКИЙ ОБЗОР ПОДХОДОВ К РАСПРЕДЕЛЕНИЮ  
ЗАДАЧ В ГРУППАХ МОБИЛЬНЫХ РОБОТОВ НА ОСНОВЕ  
ТЕХНОЛОГИЙ МЯГКИХ ВЫЧИСЛЕНИЙ.**

*Даринцев О.В., Мигранов А.Б. Аналитический обзор подходов к распределению задач в группах мобильных роботов на основе технологий мягких вычислений.*

**Аннотация.** Рассматривается использование различных типов эвристических алгоритмов на основе технологий мягких вычислений для распределения задач в группах мобильных роботов, выполняющих односложные операции в едином рабочем пространстве: генетические алгоритмы, муравьиные алгоритмы и искусственные нейронные сети. Показано, что данная задача является NP-сложной и ее решение прямым перебором для большого числа заданий невозможно. Исходная задача сведена к типовым NP-полным задачам: обобщенной задаче поиска оптимальной группы замкнутых маршрутов от одного депо и задаче коммивояжера. Представлены описание каждого из выбранных алгоритмов и сравнение их характеристик. Приводится пошаговый алгоритм работы с учетом выбранных генетических операторов и их параметров при заданном объеме популяции. Представлена общая структура разработанного алгоритма, позволяющего достаточно эффективно решить многокритериальную оптимизационную задачу с учетом временных затрат и интегрального критерия эффективности роботов, учитывающего энергетические затраты, функциональную насыщенность каждого агента группы и т.д. Показана возможность решения исходной задачи с использованием муравьиного алгоритма и обобщенного поиска оптимальной группы замкнутых маршрутов. Для многокритериальной оптимизации показана возможность линейной свертки полученного векторного критерия оптимальности за счет введения дополнительных параметров, характеризующих групповое управление: общее КПД функционирования всех роботов, затраты энергии на функционирование группы поддержки и энергия на размещение одного робота на рабочем поле. Для решения задачи распределения заданий с использованием нейронной сети Хопфилда произведено ее представление в виде графа, полученного в ходе перехода от обобщенной задачи поиска оптимальной группы замкнутых маршрутов от одного депо к задаче коммивояжера. Показателем качества выбран суммарный путь, пройденный каждым из роботов группы.

**Ключевые слова:** коллектив роботов, распределение задач, генетический алгоритм, муравьиный алгоритм, нейросеть Хопфилда.

**1. Введение.** Использование управляемых групп мобильных роботов является целесообразным во многих ситуациях, наиболее же явно их преимущество проявляется при работе в труднодоступной местности и опасных условиях эксплуатации. К примеру, использование мобильных роботов достаточно эффективно при контроле состояния сетей газо- и нефтепроводов [1, 2], при отслеживании лесных пожаров и стихийных бедствий [3, 4].

В связи с этим синтез алгоритмов квазиоптимального управления группами роботов при выполнении поставленной задачи

является одним из активно исследуемых направлений робототехники [5–7].

Более узкой задачей является задача построения оптимального алгоритма поведения группы роботов на рабочем поле с учетом наличия нескольких задач. Дополнительным критерием, усложняющим поиск решения данной задачи, является то, что это решение должно обеспечить управление при произвольном количестве задач и роботов.

В работе [8] проводится исследование возможности назначения ролей в группе роботов при ограниченных средствах коммуникаций. По сути это параллельная задача, в которой необходимо провести назначение заданий роботам только на один шаг. Авторы предлагают итерационный подход к распределению ролей в группе роботов, основанный на стратегии децентрализованного управления и принципах роевого взаимодействия.

В работе [9] на основе классического муравьиного алгоритма предложена новая методология для задач распределения роботов по заданиям. При этом рассматривается классическая задача MRTA (multi-robot task allocation): дано  $n$  роботов, каждый из которых способен выполнить одну задачу, и  $m$  возможных заданий, каждое из которых требует одного робота на выполнение. Обязательным в этом случае является наличие параметра эффективности для каждой из возможных пар робот–задача. И хотя муравьиный алгоритм позволил получить хорошие результаты по робастности для данной задачи (полиномиальная сложность решения –  $O(n^4)$ ), сама задача является более простой с точки зрения постановки. Рассматривается только распределение на один шаг и задается матрица эффективности решений до начала работы алгоритма.

По результатам проведенного анализа видно, что для поиска решений задач распределения, представленных в более простой формулировке, достаточно часто используются эвристические алгоритмы. При этом методики решения полной задачи для произвольного количества роботов и заданий, которые также могут меняться в процессе работы, в тематической литературе не приводятся.

Поэтому целью настоящей работы являются анализ сложности исходной задачи и выбор типа алгоритмов, которые могут быть использованы для эффективного решения данной задачи.

**2. Анализ типа решаемой задачи.** Проведем постановку исходной задачи и выделим ее особенности.

Имеется рабочее поле размером  $N \times N$ , на котором в произвольных узлах располагается  $n$  роботов с координатами  $(x_i, y_i, i \in [1, n])$  и  $m$  заданий с координатами заданий  $(x_j^*, y_j^*, j \in [1, m])$ .

Целью функционирования роботов является выполнение заданий, расположенных на рабочем поле, при этом в каждой клетке одновременно могут находиться только один робот и/или одно задание.

Роботы характеризуются четырьмя параметрами:

- энергопотребление при выполнении работы  $w_i^{FN}$ ,
- энергопотребление при перемещении  $w_i^{MV}$ ,
- скорость перемещения  $v_i$ ,
- энергия на ввод в эксплуатацию  $W_i^{ST}$ .

Показателем качества полученного решения является энергия, потребляемая группой роботов при выполнении поставленной группы заданий. Количественно показатель может быть представлен в следующем виде:

$$W_{\Sigma} = \sum_{i=1..n} (S_i w_i^{MV} + \tau_i w_i^{MV} + W_i^{ST}), \quad (1)$$

где  $S_i$  – расстояние, пройденное  $i$ -м роботом, а  $\tau_i$  – время, затраченное  $i$ -м роботом при выполнении группы заданий.

Расстояние в первом приближении может быть найдено как сумма евклидовых расстояний между соседними точками маршрута:

$$S_i = \sum_m S_m^i = \sum_m \sqrt{(x_m - x_{m-1})^2 + (y_m - y_{m-1})^2}. \quad (2)$$

Особенности задачи в текущей постановке:

- предполагаются произвольные количества роботов в группе и заданий на поле, что переводит поставленную задачу в общем случае в разряд NP-полных;
- используется показатель качества (энергетический фактор) (1), что позволяет оценить эффективность принятой тактики поведения группы;
- предъявляются жесткие требования к полноте информации о положении как роботов, так и заданий на рабочем поле.

Большое число роботов и заданий, выделяемых в рамках общей задачи, приводит к невозможности проведения поиска оптимального алгоритма распределения задач среди роботов полным перебором за приемлемое время, т.к. задача является NP-полной. Примерами близких по объему вычислений задач из класса NP-полных являются:

- 1) задача коммивояжера;
- 2) оптимальный раскрой (бумага, стальной прокат, отливка), оптимизация маршрутов в воздушном пространстве;
- 3) составление расписаний, учитывающих определённые условия.

Рассматриваемая задача может быть представлена в виде графа и сведена к одной из классических NP-полных задач. Для перехода к графу необходимо и достаточно представить всю группу роботов и заданий на поле в виде вершин графа.

Зададим смешанный граф  $G=\{V, A\}$ , где  $A$  – множество неориентированных ребер;  $V=\{R \cup W\}$  – объединение непересекающихся множеств вершин  $R$ , соответствующих роботам, и вершин  $W$ , соответствующих заданиям.

При этом в графе  $G$  подграф  $W$  является полным, а любая пара вершин из подмножества  $R$  не является соседней. Каждое ребро является взвешенным. Вес ребра  $w_{ij}^k$  определяет энергетические затраты соответствующего перехода, которые определяются направлением перехода (из вершины  $W_i$  в вершину  $W_j$ ) и классом робота ( $k$ ), выполняющего переход.

Граф, построенный для группы из двух роботов и 4 заданий, представлен на рисунке 1.

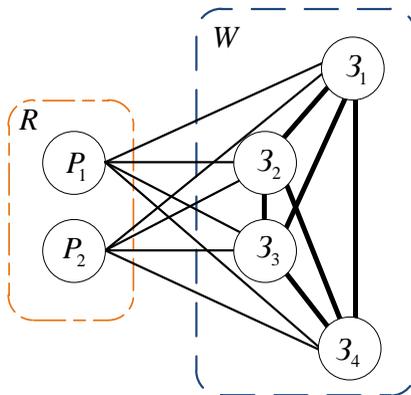


Рис. 1. Граф G для задачи с двумя роботами и четырьмя задачами

Учитывая специфику исходной постановки задачи, требуется определить порядок заданий для каждого робота таким образом, чтобы все задания были выполнены, и совокупный расход энергии был минимален. На графе это эквивалентно построению из каждой вершины подмножества  $R$  маршрутов, которые в совокупности должны охватывать все вершины графа  $G$ , не иметь смежных вершин и иметь минимальную сумму весов ребер.

После представления задачи на графе становится очевидным, что это характерно для обобщенной задачи VRP (Vehicle Routing Problem) класса – задачи комбинаторной оптимизации, в которой для парка однотипных транспортных средств (ТС) (в данном варианте - роботов) требуется определить оптимальный набор замкнутых маршрутов от единственного депо (начальная точка положения робота) до множества удаленных клиентов (заданий). На практике критерий оптимальности может выражаться любыми затратами на объезд клиентов, но чаще всего используется длина маршрута.

Обобщение задачи VRP связано с различными типами роботов Heterogeneous VRP (HVRP) и с множеством депо Multi-Depot VRP (MDVRP) [10]. MDVRP – задача маршрутизации транспорта с несколькими депо [11]. HVRP – задача маршрутизации транспорта с разнородным парком [12], в которой клиенты обслуживаются несколькими типами ТС с отличающимися характеристиками, такими как: грузоподъемность, скорость, стоимость использования и т.п.

Как NP-полная задача, задача VRP может быть сведена к любой другой задаче вышеперечисленных классов. В большинстве случаев тестирование и использование алгоритмов решения NP-полных задач проводится для решения TSP (Traveling Salesman Problem, задача коммивояжера), поэтому проведем переход от VRP к TSP задаче.

Для перехода от полученной задачи VRP класса к задаче TSP класса необходимо ввести фиктивные переходы от задач к роботам и между узлами роботов (с нулевым весом). Также конец пути  $i$ -го робота (последнее задание для данного робота) необходимо замкнуть через фиктивный путь на  $i+1$  робота, а вершину  $n$ -го маршрута - на вершину первого робота.

В результате выполненных преобразований будет получен гамильтонов цикл – одно из возможных (не обязательно оптимальное) решение TSP. Примеры гамильтоновых циклов для рассматриваемой задачи показаны на рисунке 2.

Далее будут рассмотрены алгоритмы, использующие технологии мягких вычислений, и оценена сложность решаемой задачи с их использованием.

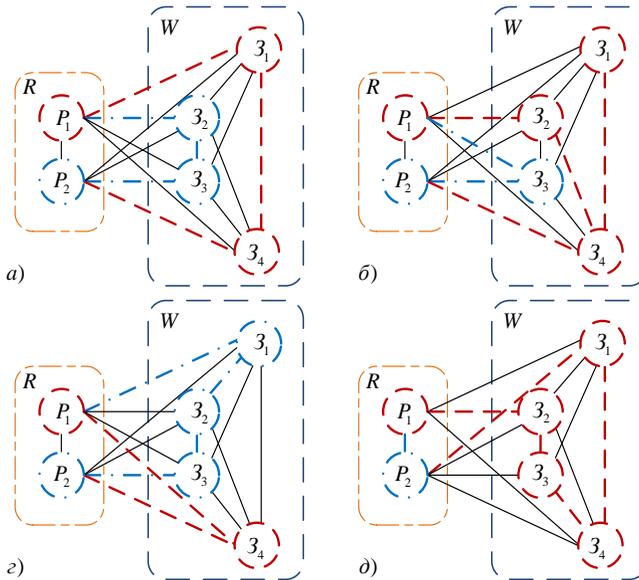


Рис. 2. Примеры гамильтоновых циклов (маршрутов-решений) на графе G для задачи с двумя роботами и четырьмя задачами

**3. Алгоритмы решения NP-задач.** Оценим сложность решения поставленной задачи при использовании алгоритма полного перебора. Для этого применим формулу оценки размера пространства поиска для задачи VRP (к которой удалось свести исходную задачу) с множественным депо и разнородным парком [13]:

$$N_{\text{VRP}} = \frac{(m+n-1)!}{(n-1)!}, \quad (3)$$

где  $n$  – число депо (в нашем случае число роботов в группе);  $m$  – число потребителей (заданий на поле).

Видно, что количество вершин-заданий определяет факториальный рост числа решений и в большей мере влияет на размер пространства поиска, нежели количество роботов. Разницу в размере пространств поиска в поставленной задаче распределения

заданий в рамках группы роботов при различных параметрах  $n$  и  $m$  легко оценить с помощью таблицы 1.

Таблица 1. Число возможных решений поставленной задачи при различном числе роботов ( $n$ ) и числе заданий ( $m$ )

Параметры	$m = 2$	$m = 3$	$m = 5$	$m = 10$
$n = 1$	2	6	120	3 628 800
$n = 2$	6	24	720	39 916 800
$n = 3$	12	60	2 520	239 500 800
$n = 4$	20	120	6 720	1 037 836 800
$n = 5$	30	210	15 120	3 632 428 800
$n = 6$	42	336	30240	10 897 286 400
$n = 7$	56	504	55440	29 059 430 400
$n = 8$	72	720	95040	70 572 902 400

Отметим, что при числе заданий  $m > 66$  объем данных превышает предел Бремерманна [14] и задача становится трансвычислительной при любом возможном количестве роботов в группе.

Поэтому при решении NP-полных задач хорошо себя зарекомендовали технологии мягких вычислений, к которым также относятся «природные вычисления» (Natural Computing). Это научное направление, объединяющее методы с природными механизмами принятия решений, среди которых:

- Genetic Algorithms – генетические алгоритмы;
- Neural Network Computing – нейро-сетевые вычисления;
- Swarm Intelligence Algorithms – роевые алгоритмы.

Из роевых алгоритмов, в свою очередь, можно выделить:

- Ant Colony Algorithms – муравьиные алгоритмы;
- Artificial Bee Colony – алгоритмы пчелиного роя;
- Particle Swarm Optimization – алгоритм роя частиц.

В большинстве известных случаев NP-полные задачи решаются с использованием эвристических алгоритмов: роевых и генетических.

Сравнительный анализ роевых алгоритмов (пчелиного роя, муравьиного и роя частиц), приведенный в работах [15-17], позволяет выбрать наиболее эффективный из них – муравьиный алгоритм.

Выбор данного алгоритма вызван следующими преимуществами:

- наличие информации о положении заданий и роботов позволяет вычислить расстояние выбранного пути для конкретного

робота между активными точками на поле и анализировать эффективность данного решения,

- возможность представления исходной задачи в виде неориентированного взвешенного графа,

- необходимость выполнения всей группы заданий на поле близко по постановке с другой NP-полной задачей – задачей коммивояжера (TSR), именно для решения которой изначально разрабатывались и тестировались муравьиные алгоритмы [18-20].

Отдельным направлением поиска методов приближенного решения оптимизационных задач данного типа является использование нейросетей [21-25]. Отметим, что для оптимизации в основном используются сеть Хопфилда или ее модификации, которые имеют ряд преимуществ при решении подобных задач:

- 1) данная сеть не требует обучения, поэтому, как результат, нет необходимости иметь обширную базу данных по решениям для анализируемой задачи;

- 2) оптимизация рассматриваемой нейросети проводится на основании минимизации энергетической функции Ляпунова.

Далее более подробно рассмотрим применение выбранных типов алгоритмов для решения задачи распределения заданий в группе роботов.

**3.1.1. Муравьиный алгоритм.** Муравьиный алгоритм независимо от модификаций может быть представлен в виде следующих шагов при решении задачи TSP:

- 1) создание «муравьев» в исходных точках;
- 2) нахождение решения (путь от исходной точки к конечной);
- 3) обновление феромона;
- 4) выполнение дополнительных действий (специфических для рассматриваемой задачи);
- 5) если глобальное условие выхода из цикла не выполнено – повтор шагов 1–4.

Феромон при этом обеспечивает не прямой обмен информацией между внутренними циклами (разными генерациями «муравьев»). По сути это разнесенное во времени взаимодействие, при котором одна особь изменяет некоторую область окружающей среды, а другие используют эту информацию позже. Для построения муравьиного алгоритма, решающего поставленную проблему, необходимо выполнить следующие этапы:

- 1) представить задачу в виде набора компонент и переходов или набора неориентированных взвешенных графов, на которых муравьи могут строить решения;

- 2) выбрать специфический ACO (Ant Colony Optimization) алгоритм и применить его для решения задачи;
- 3) настроить параметры ACO-алгоритма.

Разновидностью ACO-алгоритма, которую успешно используют для решения VRP, является метод мультиколонийной муравьиной системы (Ant Multi-Colony Optimization, AMCO). Алгоритм имеет полиномиальную сложность ( $O(n^3)$ ), что позволяет эффективно применять его при решении задач большей размерности [26].

В основе AMCO заложена возможность совместного построения части решения одновременно несколькими колониями муравьев. Изначально формируются группы муравьев, включающие по одному представителю от каждой колонии. Каждая такая группа является заменой одного муравья в классическом варианте алгоритма. Муравьи группы в результате случайной генерации последовательно формируют решение, включая вершины в собственные маршруты на каждом шаге в порядке, определяемом вероятностно на основе объединенного множества альтернатив.

Муравей в группе  $k$ , принадлежащий колонии  $x$ , включит в свою часть решения вершину  $i$  на итерации  $t$  с нормализованной вероятностью [27]:

$$p_{x,i}^k(t) = \frac{[\tau_{x,i}^k(t)]^\alpha \times [\eta_{x,i}^k]^\beta}{\sum_{y \in Y} \sum_{j \in J^k} [\tau_{x,j}^k(t)]^\alpha \times [\eta_{x,j}^k]^\beta}, \quad (4)$$

где  $Y$  – множество колоний;  $J^k$  – множество доступных вершин для группы  $k$ ;  $\eta_{x,i}^k$  – априорная привлекательность альтернативы, обратная энергии перехода из текущей вершины муравья группы  $k$  колонии  $x$  в вершину  $i$ ;  $\tau_{x,i}^k$  – апостериорная эффективность альтернативы, определяемая количеством феромона колонии  $x$  на переходе из текущей вершины муравья группы  $k$  колонии  $x$  в вершину  $i$ ;  $\alpha$ ,  $\beta$  – настраиваемые параметры алгоритма.

Решение для колонии считается сформированным, если множество вершин (заданий) окажется пустым, то есть все вершины будут посещены (либо муравьем данной колонии, либо любыми другими муравьями из параллельных колоний). После каждого решения происходит увеличение количества феромона на дугах, использованных соответствующими колониями, обратно пропорционально суммарному весу всех дуг решения:

$$\Delta\tau_{x,i}^k(t) = \begin{cases} \frac{Q}{L_{x,k}(t)}, & \text{если } (i,j) \in T_{x,k}(t), \\ 0, & \text{в обратном случае,} \end{cases} \quad (5)$$

где  $T_{x,k}(t)$  – маршрут, пройденный муравьем  $k$  колонии  $x$  на итерации  $t$ ;  $L_{x,k}(t)$  – длина этого маршрута;  $Q$  – регулируемый параметр, значение которого выбирается одного порядка с длиной оптимального маршрута.

Затем традиционным образом происходит частичное повсеместное испарение феромона:

$$\tau_{x,i}(t+1) = (1-\rho) \times \tau_{x,i}(t) + \Delta\tau_{x,i}(t), \quad (6)$$

где  $\Delta\tau_{x,i}(t) = \sum_{k=1}^b \Delta\tau_{x,i}^k(t)$ ,  $b$  – количество муравьев в колонии  $x$ .

Оценка эффективности переходов по дугам графа проводится с помощью расчета суммарной энергии по формуле (1). Конечной точкой маршрута муравья считается точка последней вершины. В работе [28] для многокритериальной оптимизации проведена линейная свертка полученного векторного критерия оптимальности за счет введения дополнительных параметров, характеризующих групповое управление: общее КПД функционирования всех роботов, удельная энергия на функционирование группы поддержки и энергия на размещение одного робота на рабочем поле.

В результате для запуска АМСО требуется установить значения для следующего набора параметров:

- $t$  – количество итераций;
- $g$  – количество межколониальных групп муравьев;
- $\alpha$  – вес концентрации феромона дуг;
- $\beta$  – вес эвристической привлекательности дуг;
- $\rho$  – коэффициент испарения феромона.

**3.1.2. Численные эксперименты.** Рассмотрим группу из двух роботов. Роботы должны выполнить четыре задания. Рабочее пространство представляет собой поле размером  $10 \times 10$  клеток. Исходные данные для рассматриваемого примера приведены в таблице 2. Расположение роботов и заданий на поле представлено на рисунке 3.

Таблица 2. Параметры роботов и заданий на поле

Роботы				
№	1	2		
Координаты	{4, 10}	{1, 7}		
Скорость $v_i$ , ед.раст./ед.врем	30	20		
Энергия на ввод в эксплуатацию $W_i^{ST}$ , ед.энерг.	60	40		
Расход энергии на перемещение $w_i^{MV}$ , ед.энерг./ед.раст.	30	40		
Потребление энергии при выполнении работы $w_i^{FN}$ , ед.энерг./ед.врем.	120	100		
Задания				
№	1	2	3	4
Координаты	{5, 8}	{1, 6}	{5, 4}	{2, 7}

	1	2	3	4	5	6	7	8	9	10
1				$P_1$						
9										
8					$Z_1$					
7	$P_2$	$Z_4$								
6	$Z_2$									
5										
4					$Z_3$					
3										
2										
1										

Рис. 3. Расположение роботов и заданий на рабочем поле

При решении задачи порядок заданий для каждого робота определялся таким образом, чтобы все задания были выполнены, а совокупный расход энергии – минимален. Рассматривались различные варианты маршрутов на подмножестве  $R$ , которые охватывают все вершины графа  $G$  и не имеют смежных вершин при минимальной сумме весов ребер. Четыре варианта (из возможных 120) построения маршрутов для рассматриваемого примера расположения двух роботов и четырех задач представлены на рисунке 4.

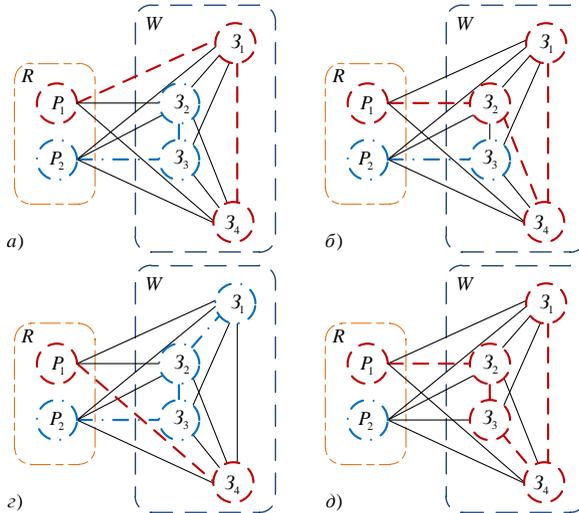


Рис. 4. Примеры маршрутов-решений на графе  $G$  для задачи с двумя роботами и четырьмя задачами

Полученное решение для рассматриваемого примера сведено в таблице 3, при этом использовались следующие параметры алгоритма:  $t = 50$ ;  $g = 40$ ;  $\alpha = 0,8$ ;  $\beta = 3$ ;  $\rho = 0,3$ .

Таблица 3. Результаты расчета

№ робота	Стратегия	Дистанция маршрута, ед. длин.	Время маршрута, ед. врем.	Число заданий	Суммарные затраты энергии, ед.эн.
1	$Z_1-Z_3$	6,236	0,208	2	420,7
2	$Z_2-Z_4$	2,414	0,121	2	

Первый робот последовательно выполняет задание 1 ( $Z_1$ ) и задание 3 ( $Z_3$ ), второй робот – задание 2 ( $Z_2$ ) и задание 4 ( $Z_4$ ). Нетрудно убедиться, что найденная стратегия является оптимальной. Рассмотренный подход может быть использован для построения оптимальной стратегии управления произвольной группой мобильных роботов и выполнения произвольного числа заданий [28].

**3.2.1. Генетический алгоритм.** Для решения поставленной задачи с помощью генетических алгоритмов (ГА) необходимо четко сформулировать, что именно является решением этой задачи,

закодировать решение в виде хромосомы и составить функцию приспособленности для таких хромосом. После этого можно решать данную задачу средствами ГА.

Введем разделение роботов на классы. Класс робота будет определять группу заданий, которую он может выполнить, энергопотребление робота в различных режимах и скорость перемещения робота. Выделим три режима (состояния) робота: сон (ожидание задания), перемещение и выполнение задания, каждое из которых характеризуется своим энергопотреблением (для  $i$ -го робота –  $W_i^{SL}$ ,  $W_i^{MV}$ ,  $W_i^{WR}$  соответственно). Каждый робот обладает начальным уровнем заряда ( $P_i^R$ ).

Энергия на функционирование определяется коэффициентом затрат в единицу времени и временем, пока робот активен – с момента получения задания (последовательности заданий) до момента выполнения всех поставленных заданий.

Энергопотребление в результате перемещения определяется количеством энергии, необходимым на преодоление единицы длины.

Энергопотребление на выполнение задания определяется классом робота и классом задания. Задания разделяются на несколько классов. Каждый класс имеет свой уровень энергопотребления ( $W_j^{WR}$ ) – энергия, которую должен затратить робот на выполнение данного класса задания. Робот может выполнить задачу того же, либо более низкого, класса.

С учетом перечисленных характеристик объектов необходимо проработать алгоритм поиска оптимального плана распределения задач между имеющимися роботами с точки зрения минимизации затраченной энергии и времени.

Основная цель выбора показателей качества – минимизация затраченной энергии по группе роботов и времени, затраченного на выполнение плана.

Первым показателем качества является уровень затраченной энергии на выполнение предлагаемого плана распределения задач.

Согласно постановке задачи суммарная энергия, затраченная на выполнение выбранного плана, будет следующей:

$$W_{\Sigma} = \sum_{k \in N_{SL}} W_k^{SL} + \sum_{m \in N_w} (W_m^{MV} + W_m^{WR}), \quad (7)$$

где  $N_{SL}$  – множество роботов, не участвующих в выполнении заданий;  
 $N_w$  – множество роботов, задействованных в выполнении заданий.

Энергозатраты  $i$ -го робота на перемещение  $W_i^{MV}$  из начального положения в пункт выполнения  $j$ -го задания пропорциональны расстоянию от робота до задачи:

$$W_{ij}^{MV} = k_i^{MV} \times S_{ij}, \quad (8)$$

где  $k_i^{MV}$  – расход энергии робота на единицу длины при перемещении, коэффициент пропорционально зависящий от его класса;  
 $S_{ij}$  – расстояние от  $i$ -го робота до  $j$ -го задания, которое будет определяться по нейросетевому алгоритму поиска оптимальной совокупности путей всей группы роботов от исходных точек к задачам.

Вторым показателем качества является время  $t_i$ , необходимое для перемещения  $i$ -го робота из начального положения в пункт выполнения выданного ему задания. Данное время определяется не только скоростью и дистанцией до задания, но и конфигурацией пути робота с учетом объезда препятствий и поочередного прохождения «узких» мест на полигоне совместно с другими роботами, поэтому это время может также определяться в системе планирования траекторий. По совокупности этих времен определяется максимальное время, которое будет устанавливать время выполнения заданий группой роботов:

$$t_{\max} = \max_{m \in N_w} (t_m). \quad (9)$$

Третьим показателем качества является количество невыполненных заданий на данном шаге  $N_{out}$ . Так как число задач и роботов в общем случае может отличаться, необходимо отслеживать, какое количество заданий остается невыполненным.

Таким образом, для рассматриваемой системы в качестве критериев оптимальности использованы:

- 1) общее потребление:

$$j_l(\bar{z}) = W_{\Sigma}(\bar{z}) \rightarrow \min_{z \in Z};$$

- 2) максимальное время выполнения:

$$j_2(\bar{z}) = t_{\max}(\bar{z}) \rightarrow \min_{z \in Z};$$

- 3) число невыполненных заданий:

$$j_3(\bar{z}) = N_{\text{out}}(\bar{z}) \rightarrow \min_{z \in Z}.$$

Здесь  $\bar{z}$  – вектор номеров задач, каждый элемент которого  $z_i$ ,  $i = \overline{1, n}$  определяет задачу для  $i$ -го робота на следующий шаг. Область допустимых значений вектора  $\bar{z}$  образует множество  $Z$ .

Поставим трехкритериальную задачу выбора оптимального плана распределения задач между имеющимися роботами: найти значения варьируемых параметров  $\bar{z}$ , которые в пределе обеспечивают выполнение всех условий 1) – 3). Векторный критерий оптимальности  $(\phi_1(\bar{z}), \phi_2(\bar{z}), \phi_3(\bar{z}))$  определен на множестве  $Z$ , и оператор должен стремиться минимизировать значение каждого из частных критериев оптимальности  $\phi_i(\bar{z})$ ,  $i = \overline{1, n}$ :

$$J: \min_{z \in Z} (\phi_1(\bar{z}), \phi_2(\bar{z}), \phi_3(\bar{z})). \quad (10)$$

Для поиска решения предлагается использовать генетический алгоритм распределения задач в соответствии с перечисленными условиями (ограничениями) и критериями оптимальности.

Следующим шагом является создание фитнесс-функции для выбранной задачи. Укрупненный алгоритм, который был взят за основу при создании функции, следующий:

1. Декодирование хромосомы (производится дешифровка генов в номера задач для каждого робота).
2. Расчет траектории движения роботов к цели.
3. Подсчет параметра  $W_{\Sigma}$  для оцениваемой комбинации заданий согласно выражениям (7)–(9).
4. Подсчет времени, затрачиваемого роботами на выполнение выданной последовательности заданий.

5. Подсчет параметра  $N_{out}$  для оцениваемой комбинации задач как разницы между числом всех заданий  $m$  и количеством ненулевых элементов в векторе  $\bar{z}$ .

6. Вывод полученного вектора значений фитнес–функции для рассматриваемой особи  $\bar{z}$ .

С учетом специфики решаемой задачи для реализации выбранного способа был использован один из эффективных типов алгоритма ГА – алгоритм с проведением сортировки по доминированию, с элитарным отбором (NSGA-II) [29, 30] и целочисленными ограничениями [31]. В случае целочисленного ограничения операция мутации исключается из общего алгоритма (так как не существует функций мутации, которые поддерживают целочисленные ограничения).

Опишем пошагово работу выбранного ГА с учетом выбранных генетических операторов и их параметров при заданном объеме популяции  $R$ :

1. После старта алгоритма создается популяция из  $2 \cdot R$  особей. Начальная популяция генерируется случайным образом с использованием равномерного распределения. Каждая особь – алгоритм команд, состоящий из номеров заданий для каждого робота.

2. Особи оцениваются на уровень приспособляемости по критерию  $Q$ .  $R$  наиболее приспособленных особей включаются в новое поколение.

3. Проводится селекция методом парного турнирного отбора, в результате выбираются группы родителей (особей), которые дадут потомство. Проводится операция скрещивания для отобранных особей (родителей).

4. Создание потомков происходит с использованием однородного кроссовера. В нем скрещивание выполняется на основе бинарного вектора, длина которого равна числу геномов в хромосоме. Каждая позиция вектора определяет взаимную замену генома у родительских особей. Если позиция равна 1 – потомку достанется ген от родителя №1, если позиция равна 0 – от родителя №2. В результате скрещивания создается  $R$  особей-потомков;

5. Шаги повторяются 2–5, если не достигнут критерий выхода из цикла.

Критериями остановки алгоритма служат неизменность лучших особей в течение заданного числа поколений (20–50 поколений зачастую достаточно), либо достижение критического размера числа поколений. Критерий выхода позволяет сократить вероятность попадания в локальный оптимум системы. В то же время

использование NSGA-II и турнира при селекции позволяет ускорить процесс выделения глобального оптимума.

При завершении работы алгоритма среди получившихся  $R$  особей выбираются уникальные (возможно повторение одной из оптимальных особей несколько раз), которые являются оптимальным множеством решений поставленной проблемы (рисунок 5).

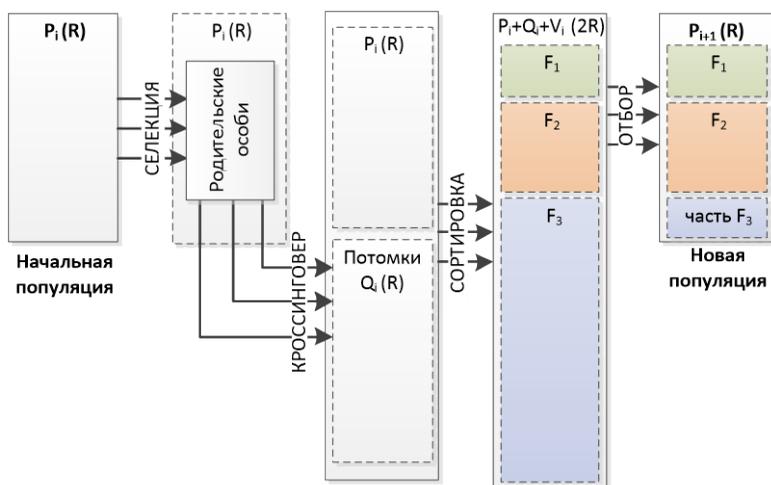


Рис. 5. Структура NSGA-II

**3.2.2. Численные эксперименты.** Рассмотрим группу из трех роботов, которым могут быть назначены шесть заданий. Рабочее пространство представляется в виде дискретного поля размером  $10 \times 10$  клеток. Исходные данные для рассматриваемого примера приведены в таблицах 4–6. Расположение роботов и заданий на поле представлено на рисунке 6.

Таблица 4. Параметры роботов и задач на поле

Робот, №	Координаты	Класс	$P_i^R$ ед.эн.	Задания №	Координаты	Класс
1	{6,2}	3	40	1	{8,3}	1
2	{4,10}	1	40	2	{1,6}	1
3	{1,7}	2	40	3	{5,4}	2
				4	{4,9}	3
				5	{2,10}	3
				6	{2,7}	1

Таблица 5. Энергетические параметры

Класс робота	$W^{SL}$ , ед.эн.	$k^{MV}$ , ед.эн./ед.дл.	$M_v$ , ед.дл./ ед.вр.	Класс заданий	$W^{WR}$ , ед.эн.
1	2,1	0,7	100	1	3,2
2	1,1	1,2	200	2	2,8
3	1,7	0,9	150	3	5,8

Таблица 6. Время обработки заданий

Класс робота	Класс заданий		
	1	2	3
1	3	-	-
2	5	4	-
3	5	7	2

	1	2	3	4	5	6	7	8	9	10
1		$z_5^{(3)}$		$P_2^{(1)}$						
9				$z_4^{(3)}$						
8										
7	$P_3^{(2)}$	$z_6^{(1)}$								
6	$z_2^{(1)}$									
5										
4					$z_3^{(2)}$					
3								$z_1^{(1)}$		
2						$P_1^{(3)}$				
1										

Рис. 6. Расположение роботов и задач на рабочем поле

Расчет проводится с глубиной анализа в 4 задания. Другими словами, для каждого робота генерируется последовательная цепь из 4 заданий.

Полные результаты решения трехкритериальной оптимизационной задачи представлены в таблице 7.

Таблица 7. Оптимальные решения

№ решения	$W_{\Sigma}$ , ед.эн.	$t_{\max}$ , ед.вр.	$N_{\text{out}}$
1.1	125,18	59,94	0
1.2	117,66	60,40	0
1.3	112,83	49,01	1
1.4	105,39	51,53	1
1.5	89,09	54,14	1
1.6	89,29	33,84	2
1.7	82,35	35,51	2
1.8	81,13	36,90	2
1.9	80,77	41,64	2
1.10	49,00	22,93	3
1.11	65,19	19,93	4
1.12	55,66	15,93	5

Рассмотрим более подробно полученные решения для  $N_{\text{out}} = 0$ . Цепи заданий для каждого из роботов представлены в таблице 8.

Таблица 8. Оптимальные решения

№	Стратегия (цепи заданий)			$W_{\Sigma}$ , ед.эн.	$t_{\max}$ , ед.вр.	$N_{\text{out}}$
	1	2	3			
1.1	3 <sub>4</sub> -3 <sub>5</sub>	3 <sub>1</sub>	3 <sub>2</sub> -3 <sub>3</sub> -3 <sub>6</sub>	125,18	59,94	0
1.2	3 <sub>4</sub> -3 <sub>5</sub>	3 <sub>2</sub> -3 <sub>6</sub>	3 <sub>1</sub> -3 <sub>3</sub>	117,66	60,40	0

Таким образом, получено 12 уникальных стратегий поведения роботов, из которых два близки к оптимальным решениям - с нулевым количеством невыполненных заданий. В зависимости от требуемых условий поиска полученные решения позволяют:

1. оценить максимально возможное число выполненных операций данной группой роботов из условия ограниченного резерва по энергии или по времени;
2. выбрать наиболее быструю, либо наиболее энергосберегающую стратегию исходя из требуемого числа выполненных задач.

В работе [32] более подробно показано использование ГА для распределения задач и решения многокритериальной оптимизационной задачи с учетом временных затрат и интегрального критерия эффективности роботов, учитывающего энергетические затраты, функциональную насыщенность каждого агента группы и т.д.

**3.3.1. Нейросетевой алгоритм Хопфилда.** В 1985 г. Хопфилд и Танк [33] показали, что нейронная сеть с обратной связью может быть использована для решения задач комбинаторной оптимизации, таких как TSP. В настоящее время модель Хопфилда успешно применяется во многих областях, ключевым вопросом при ее применении является выбор весов в функции энергии, позволяющих получить быструю сходимость и подавить ложные состояния.

В процессе работы динамика таких сетей сходится к одному из положений равновесия, которые являются локальными минимумами функционала энергии сети [34].

Функционал энергии сети является функцией оптимизации, которая в общем виде содержит 4 компонента:

$$\begin{aligned}
 E(g) &= E_1 + E_2 + E_3 + E_4; \\
 \left\{ \begin{aligned}
 E_1 &= A \sum_X \sum_i \sum_{j \neq i} g_{X_i} g_{Y_j}; \\
 E_2 &= B \sum_i \sum_X \sum_{X'Y} g_{X_i} g_{Y_i}; \\
 E_3 &= C \left( \sum_X \sum_i g_{X_i} - N \right)^2; \\
 E_4 &= D \sum_X \sum_{Y \neq X} \sum_i d_{XY} g_{X_i} (g_{Y_i} + g_{Y_{i-1}});
 \end{aligned} \right. \quad (11)
 \end{aligned}$$

где  $g_{X_i}$  – логический параметр узла  $X_i$  графа (может принимать значение 0 или 1);  $d_{XY}$  – вес ненаправленного ребра между узлами  $X$  и  $Y$ ;  $A, B, C, D$  – относительные веса слагаемых.

Первые три слагаемых –  $E_1, E_2, E_3$  – отвечают за допустимость маршрута: каждый город должен быть посещен не более чем один раз (в каждой строке матрицы имеется не более одной единицы), под каждым номером должно посещаться не более одного города (в каждом столбце – не более одной единицы) и, кроме того, общее число посещений равно числу городов  $N$  (в матрице всего имеется ровно  $N$  единиц).

Четвертое слагаемое  $E_4$  – коэффициент, пропорциональный энергии (суммарному весу выбранного маршрута) нейросети.

Метка  $g_{X_i}$  определяется потенциалом узла  $u_{X_i}$  следующим выражением:

$$g_{xi} = \frac{1 + \tanh(u_{xi}/u_0)}{2}. \quad (12)$$

Начальное значение потенциала:

$$u_{in} = \frac{u_0}{2} \ln(n-1), \quad (13)$$

где  $u_0$  – пороговый уровень потенциала.

Входной потенциал имеет следующее значение:

$$u_{xi} = -\frac{\partial E}{\partial u_{xi}}. \quad (14)$$

С учетом данных выражений модель нейродинамики сети Хопфилда задается в виде:

$$\frac{du_{xi}}{dt} = -\frac{u_{xi}}{\tau} - \left[ A \sum_{ji} g_{xi} + B \sum_{xy} g_{yi} + C \left( \sum_x \sum_i g_{xi} - N \right) + D \sum_Y d_{xy} (g_{yi+1} + g_{yi-1}) \right], \quad (15)$$

где  $\tau$  – числовой параметр.

Пересчитанные в (14) и (15) значения потенциала можно использовать при перерасчете меток  $g_{xi}$  согласно выражению (12). Процесс повторяется некоторое число раз. Если рассчитанные метки обнуляют уравнения (11), выбранный вариант матрицы принимается оптимальным.

В итоговой маршрутной матрице (состоит из оценок меток  $g_{xi}$ ) номер столбца отвечает номеру шага обхода, а номер строки в столбце – номеру узла, к которому необходимо перейти.

**3.3.2. Численные эксперименты.** В качестве работы алгоритма приведем результаты моделирования для ранее рассмотренного примера для двух роботов и четырех задач (рисунок 3) с исходными данными из таблицы 1.

Для перехода к графу необходимо и достаточно представить всю группу роботов и заданий на поле в виде вершин графа. Зададим смешанный граф  $G = \{V, A\}$ , где  $A$  – множество неориентированных

ребер;  $V = \{R \cup W\}$  – объединение непересекающихся множеств вершин  $R$ , соответствующих роботам, и вершин  $W$ , соответствующих заданиям.

В графе  $G$  подграф  $W$  является полным. Любая пара вершин из подмножества  $R$  не является соседней. Каждое ребро является взвешенным. Граф, построенный для заданного положения роботов и задач, будет аналогичен графу, показанному на рисунке 1.

С помощью формулы (1) задаются веса соответствующих ребер в системе. Результатом работы нейросети является гамильтонов цикл, который показан на рисунке 7 слева, а справа приведен результат обратного преобразования результата решения задачи TSP в стратегию поведения роботов.

Использование нейросети Хопфилда позволяет корректно решить задачу оптимизации маршрута при управлении малыми группами роботов (число вершин равно  $N=6$ ). Для задач большей размерности (к примеру, для  $N=20$ ) результат решения и время расчета будет определяться выбором свободных параметров сети.

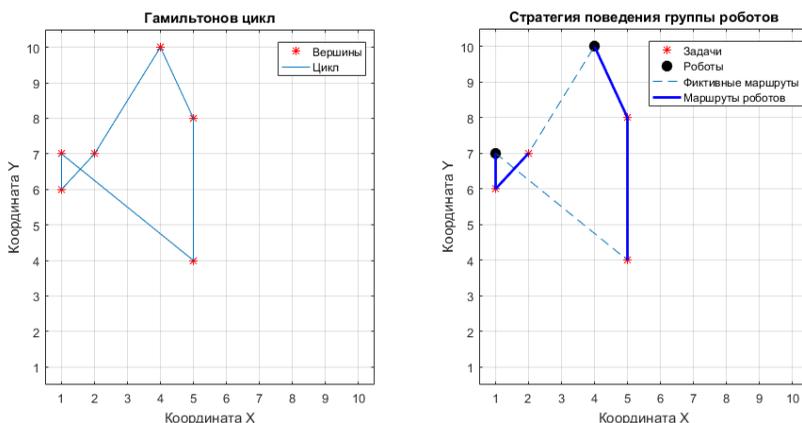


Рис. 7. Результат расчета стратегии поведения группы роботов с использованием нейросети Хопфилда

Таким образом, для возможности использования нейронной сети исходная задача была сведена к задаче коммивояжера, уже в решении которой был использован классический алгоритм нейронной сети Хопфилда.

Отличительной особенностью данного типа рекуррентной нейросети является отсутствие необходимости обучения при решении

оптимизационных задач, что позволяет использовать данную нейросеть для решения рассматриваемой задачи.

Поиск стационарного состояния нейросети позволил получить оптимальную стратегию поведения роботов для задачи низкой размерности (6 вершин) при минимизации суммарной длины пути следования. Более подробное описание метода приводится в работе [35].

**4. Оценка эффективности алгоритмов.** После анализа каждого из алгоритмов приведем сравнение их характеристик (таблица 9).

В качестве основных характеристик рассмотрим:

- точность решения (насколько эффективен алгоритм при поиске глобального оптимума);
- метод оптимизации (какой метод лежит в основе алгоритма);
- сложность (вычислительная сложность алгоритма в зависимости от размерности задачи).

Таблица 9. Сравнение характеристик рассмотренных алгоритмов

Характеристика	Алгоритм		
	Муравьиный	Генетический	Нейронная сеть Хопфилда
Точность решения	близкая к границе Парето	близкая к границе Парето, свойственна остановка на локальном оптимуме	близкая к границе Парето, свойственно появление случайных неоптимальных решений
Метод оптимизации	метаэвристическая оптимизация по примеру поведения муравьев	наследование, мутация, отбор и кроссинговер	минимизация энергетической функции Ляпунова
Сложность	$O(n^3)$	$O(n^2)$	

Следует отметить, что для каждого алгоритма решение может быть улучшено за счет использования метаоптимизации. При этом временная сложность алгоритма увеличится (зависит от используемых методов метаоптимизации), однако решение приблизится к глобальному оптимуму за счет исключения локальных дефектов.

Количественное сравнение эффективности алгоритмов при решении исходной задачи может быть проведено только после реализации каждого из алгоритмов и анализа решений группы типовых задач распределения заданий для группы автономных роботов.

**5. Заключение.** В работе проведен анализ исходной задачи распределения заданий для группы автономных роботов. Определено, что данная задача является NP-сложной и ее решение прямым перебором для большого числа заданий (более 66) невозможно. Проведено сведение исходной задачи к типовым NP-полным задачам: обобщенной задаче поиска оптимальной группы замкнутых маршрутов от одного депо (VRP) и задаче коммивояжера (TSP).

Проведен обзор эвристических методов поиска квазиоптимального решения NP-полных задач. Выделено три алгоритма, которые наиболее часто используются для решения данного типа задач: муравьиный, генетический и нейросетевой.

В приведенных в статье алгоритмах не учитывался один из наиболее используемых критериев – величина запаса энергии на борту робота, – так как рассматривался вариант построения группы с возможностью обмена энергией между агентами.

Проведено описание каждого из выбранных алгоритмов, показана специфика их применения и выполнено сравнение ключевых характеристик. В дальнейшем планируется провести более подробный сравнительный анализ применимости данных алгоритмов для решения задачи оптимизации распределения заданий для групп с большим числом роботов как для гомогенных, так и для гетерогенных составов.

### **Литература**

1. Kalyaev I. et al. A novel method for distribution of goals among UAVs for oil field monitoring // IEEE 6th ICIEVISMHT, Himeji, Japan. 2017. pp. 1–4.
2. ONDRACEK J. Intelligent Algorithms for Monitoring of the Environment Around Oil Pipe Systems Using Unmanned Aerial Systems. – Bachelor’s thesis. Czech Technical University in Prague, 2014.
3. CASBEER D.W. et al. Forest fire monitoring with multiple small UAVs // Proc. of the American Control Conference, June 2005, Portland, Oregon. 2005. pp. 3530–3535.
4. SUJIT P.B., KINGSTON D., BEARD R. Cooperative forest fire monitoring using multiple UAVs // 46th IEEE Conference on Decision and Control, 10–11 December 2007, New Orleans, Louisiana USA. 2007. pp. 4875–4880.
5. Khamis A., Hussein A., Elmogy A. Multi-robot Task Allocation: A Review of the State-of-the-Art. In: Koubâa A., Martínez-de Dios J. (eds) Cooperative Robots and Sensor Networks 2015. Studies in Computational Intelligence, vol 604. Springer, Cham.
6. Yi-Lin, L., Kuo-Lan, S. Multi-robot-based intelligent security system. Artif. Life Robot. 16, 137–141 (2011).

7. Marino, A., Parker, L.E., Antonelli, G., Caccavale, F. A decentralized architecture for multi-robot systems based on the null-space-behavioral control with application to multi-robot border patrolling. *J. Intell. Robot. Syst.* 71, pp. 423–444.
8. Иванов Д.Я. Распределение ролей в коалициях роботов при ограниченных коммуникациях на основе роевого взаимодействия // *Управление большими системами. Институт проблем управления им. В.А. Трапезникова РАН*, 2019. Vol. 78. С. 23–45.
9. Jian-Ping Wang, Yuesheng Gu and Xiao-Min Li Multi-robot Task Allocation Based on Ant Colony Algorithm // *Journal of Computers* vol. 7, no. 9, pp. 2160-2167, 2012.
10. Кубил В.Н. Обзор обобщений и расширений задачи маршрутизации транспорта // *Вестник РГУПС*, № 2, 2018. С. 97-109.
11. Chao I.M., Golden B.L., Wasil E. A new heuristic for the multi-depot vehicle routing problem that improves upon best-known solutions // *American Journal of Mathematical and Management Sciences*, Vol. 13, No. 3-4, 1993. pp. 371-406.
12. Choi E., Tcha D.W. A column generation approach to the heterogeneous fleet vehicle routing problem // *Computers & Operations Research*, Vol. 34, No. 7, 2007. pp. 2080-2095.
13. Кубил В.Н. Пространство решений задач коммивояжера и маршрутизации транспорта // *Фундаментальные исследования, методы и алгоритмы прикладной математики в технике, медицине и экономике: материалы 16-ой Междунар. молодежн. науч.-практ. конф. ЮРГПУ (НПИ). Новочеркасск : Лик. 2017. С. 33-39.*
14. Bremermann H.J. Optimization through evolution and recombination // *Yovits M.C., Jacobi G.T. and Goldstein G.D. (Eds.), Self-Organizing Systems*, 1962. pp. 93-106.
15. Зайцев А.А. Обзор эволюционных методов оптимизации на основе роевого интеллекта / А.А. Зайцев, В.В. Курейчик, А.А. Полуланов // *Известия ЮФУ. 2010. № 12 (113). С. 7–12.*
16. Engelbrecht A.P. *Fundamentals of Computational Swarm Intelligence*. John Wiley & Sons, Chichester, UK, 2005.
17. Литовка Н.В. Роевой интеллект в задачах оптимального размещения объектов пространственно распределенного предприятия/ Н.В. Литовка// *Научные труды КубГТУ*, № 11, 2018 год.
18. Dorigo M. *Swarm Intelligence, Ant Algorithms and Ant Colony Optimization // Reader for CEU Summer University Course «Complex System»*. Budapest, Central European University, 2001. pp. 1–38.
19. Caro G.D., Dorigo M. *Anet a Mobile Agents Approach to Adaptive Routing*. Technical Report IRIDA 97-12. IRIDA. Universite Libre de Bruxelles. Brussels, Belgium, 1997. 27 p.
20. Курейчик В.В., Запорожец Д.Ю. Роевой алгоритм в задачах оптимизации // *Известия ЮФУ. Технические науки. 2010. №7. С. 28-32.*
21. Liping Zhu Remarkable problem-solving ability of unicellular amoeboid organism and its mechanism// *Liping Zhu, Song-Ju Kim, Masahiko Hara and Masashi Aono*, 2018 (<https://royalsocietypublishing.org/doi/10.1098/rsos.180396>).
22. Ежов А.А. Дообучение нейронной сети Хопфилда: поиск глобального минимума функционала и модель быстрого сна// *Ежов А.А., Черепнев А.С./ Математическое моделирование. 2009. Т. 21. № 5. С. 10-20.*
23. Лоскутов А.И. Решение задачи о ранце на основе динамической нейронной сети Хопфилда// *Лоскутов А.И., Горбулин В.И., Карпушев С.И., Ряхова Е.А./ Нелинейный мир. 2019. Т. 17. № 3. С. 25-35.*
24. Хайкин С. *Нейронные сети. Полный курс. Изд. 2-е, испр. М.: Вильямс. 2016. 1104 с.*
25. Музычин В.В. Исследование возможности использования рекуррентной нейронной сети Хопфилда для решения задачи коммивояжера// *Музычин В.В.,*

- Мациевский С.В./ Современная наука: актуальные проблемы теории и практики. Серия: Естественные и технические науки. 2020. № 5. С. 93-99.
26. Кубил В.Н. Исследование и разработка методов решения многокритериальных задач маршрутизации транспорта на основе муравьиного алгоритма: дис.. канд. т. наук. Южно-Российский гос. политехнический университет имени М.И. Платова, Новочеркасск, 2019.
  27. Кубил В.Н., Мохов В.А. Многоколонийный муравьиный алгоритм с модификациями для решения многокритериальных задач маршрутизации транспорта // Известия высших учебных заведений. Электромеханика, Т. 61, № 6, 2018. С. 94-109.
  28. Migranov A.B., Darintsev O.V. Choosing a Swarm Algorithm to Synthesis an Optimal Mobile Robot Team Control Strategy // 2020 International Multi-Conference on Industrial Engineering and Modern Technologies, Vladivostok, Russia, 2020, pp. 1-5.
  29. Лотов А.В. Многокритериальные задачи принятия решений: Учебное пособие/ А.В. Лотов, И.И. Пospelова. – М.: МАКС Пресс, 2008. – 197 с.
  30. Goldberg D. Genetic Algorithms in Search, Optimization, and Machine Learning/ D. Goldberg. – Massachusetts: Addison-Wesley, 1989.
  31. Панченко Т.В. Генетические алгоритмы: учебно-методическое пособие/ Т.В. Панченко; под ред. Ю.Ю. Тарасевича. – Астрахань: Издательский дом «Астраханский университет», 2007. – 87 с.
  32. Migranov A.B., Darintsev O.V. The Use of Genetic Algorithms for Distribution of Tasks in Groups of Mobile Robots with Minimization of Energy Consumption // 2019 International Multi-Conference on Industrial Engineering and Modern Technologies, Vladivostok, Russia, 2019, pp. 1-6.
  33. Hopfield, J.J., Tank, D.W. (1985). Neural Computation of Decisions in Optimization Problems. Biological Cybernetics, Vol. 52, pp.141-152.
  34. Кононов А.А. Использование метода нейронных сетей Хопфилда для решения задачи маршрутизации в сети// Кононов А.А./ Московский экономический журнал, №9, 2019. С.74
  35. Darintsev O.V. Migranov A.B. Using the Hopfield Neural Network to Select a Behaviour Strategy for the Group of Mobile Robots // IOP Publishing, 2021, J. Phys.: Conf. Ser. 2096 012086

**Даринцев Олег Владимирович** — д-р техн. наук, доцент, главный научный сотрудник, лаборатория «робототехника и управление в технических системах», Институт механики им. Р.Р. Мавлютова — обособленное структурное подразделение Уфимского федерального исследовательского центра Российской академии наук; профессор, кафедра автоматизации технологических процессов, Уфимский государственный авиационный технический университет. Область научных интересов: робототехника, микроробототехника, МЭМС, искусственный интеллект, системы обработки информации, СТЗ. Число научных публикаций — 230. ovd@uimech.org; Проспект Октября, 71, 450054, Уфа, Россия; р.т.: +7(347)225-5255.

**Мигранов Айрат Барисович** — канд. техн. наук, старший научный сотрудник, лаборатория «робототехника и управление в технических системах», Институт механики им. Р.Р. Мавлютова — обособленное структурное подразделение Уфимского федерального исследовательского центра Российской академии наук. Область научных интересов: микроробототехника, МЭМС, искусственный интеллект. Число научных публикаций — 138. abm.imech.anrb@mail.ru; Проспект Октября, 71, 450054, Уфа, Россия; р.т.: +7(347)225-5255.

**Поддержка исследований.** Работа выполнена в рамках госзадания FWGZ-2019-0089.

O. DARINTSEV, A. MIGRANOV

## ANALYTICAL REVIEW OF APPROACHES TO THE DISTRIBUTION OF TASKS FOR MOBILE ROBOT TEAMS BASED ON SOFT COMPUTING TECHNOLOGIES

*Darintsev O., Migranov A. Analytical Review of Approaches to the Distribution of Tasks for Mobile Robot teams Based on Soft Computing Technologies.*

**Abstract.** The use of various types of heuristic algorithms based on soft computing technologies for the distribution of tasks in groups of mobile robots performing monosyllabic operations in a single workspace is considered: genetic algorithms, ant algorithms and artificial neural networks. It is shown that this problem is NP-complex and its solution by direct iteration for a large number of tasks is impossible. The initial problem is reduced to typical NP-complete problems: the generalized problem of finding the optimal group of closed routes from one depot and the traveling salesman problem. A description of each of the selected algorithms and a comparison of their characteristics are presented. A step-by-step algorithm of operation is given, taking into account the selected genetic operators and their parameters for a given population volume. The general structure of the developed algorithm is presented, which makes it possible to solve a multi-criteria optimization problem efficiently enough, taking into account time costs and the integral criterion of robot efficiency, taking into account energy costs, functional saturation of each agent of the group, etc. The possibility of solving the initial problem using an ant algorithm and a generalized search for the optimal group of closed routes is shown. For multi-criteria optimization, the possibility of linear convolution of the obtained vector optimality criterion is shown by introducing additional parameters characterizing group control: the overall efficiency of the functioning of all robots, the energy costs for the functioning of the support group and the energy for placing one robot on the work field. To solve the task distribution problem using the Hopfield neural network, its representation is made in the form of a graph obtained during the transition from the generalized task of finding the optimal group of closed routes from one depot to the traveling salesman problem. The quality indicator is the total path traveled by each of the robots in the group.

**Keywords:** team of robots, distribution of tasks, genetic algorithm, ant algorithm, Hopfield neural network.

### References

1. Kalyaev I. et al. A novel method for distribution of goals among UAVs for oil field monitoring // IEEE 6th ICIEVISCMT, Himeji, Japan. 2017. pp. 1–4.
2. ONDRACEK J. Intelligent Algorithms for Monitoring of the Environment Around Oil Pipe Systems Using Unmanned Aerial Systems. – Bachelor's thesis. Czech Technical University in Prague, 2014.
3. CASBEER D.W. et al. Forest fire monitoring with multiple small UAVs // Proc. of the American Control Conference, June 2005, Portland, Oregon. 2005. pp. 3530–3535.
4. SUJIT P.B., KINGSTON D., BEARD R. Cooperative forest fire monitoring using multiple UAVs // 46th IEEE Conference on Decision and Control, 10-11 December 2007, New Orleans, Louisiana USA. 2007. pp. 4875–4880.
5. Khamis A., Hussein A., Elmogy A. Multi-robot Task Allocation: A Review of the State-of-the-Art. In: Koubãa A., Martínez-de Dios J. (eds) Cooperative Robots and Sensor Networks 2015. Studies in Computational Intelligence, vol. 604. Springer, Cham.

6. Yi-Lin, L., Kuo-Lan, S. Multi-robot-based intelligent security system. *Artif. Life Robot.* 16, 2011. pp. 137–141.
7. Marino, A., Parker, L.E., Antonelli, G., Caccavale, F. A decentralized architecture for multi-robot systems based on the null-space-behavioral control with application to multi-robot border patrolling. *J. Intell. Robot. Syst.* 71, pp. 423–444.
8. Ivanov D.Ja. Raspredelelenie rolev v koalitsijah robotov pri ogranichenennyh kommunikacijah na osnove roevogo vzaimodejstvija // *Upravlenie bol'shimi sistemami. Institut problem upravlenija im. VA Trapeznikova RAN*, 2019. Vol. 78. pp. 23–45. (In Russ.).
9. Jian-Ping Wang, Yuesheng Gu and Xiao-Min Li Multi-robot Task Allocation Based on Ant Colony Algorithm // *Journal of Computers* vol. 7, no. 9, 2012. pp. 2160-2167.
10. Kubil V.N. Obzor obobshhenij i rasshirenij zadachi marshrutizacii transporta // *Vestnik RGUPS*, № 2, 2018. pp. 97-109. (In Russ.).
11. Chao I.M., Golden B.L., Wasil E. A new heuristic for the multi-depot vehicle routing problem that improves upon best-known solutions // *American Journal of Mathematical and Management Sciences*, Vol. 13, No. 3-4, 1993. pp. 371-406.
12. Choi E., Tcha D.W. A column generation approach to the heterogeneous fleet vehicle routing problem // *Computers & Operations Research*, Vol. 34, No. 7, 2007. pp. 2080-2095.
13. Kubil V.N. Prostranstvo reshenij zadach kommivojazhera i marshrutizacii transporta // *Fundamentalnye issledovanija, metody i algoritmy prikladnoj matematiki v tehnikе, medicine i jekonomike : materialy 16-oj Mezhdunar. molodezhn. nauch.-prakt. konf. JuRGPU (NPI). Novocherkassk : Lik.* 2017. pp. 33-39. (In Russ.).
14. Bremermann H.J. Optimization through evolution and recombination // *Yovits M.C., Jacobi G.T. and Goldstein G.D. (Eds.), Self-Organizing Systems*, 1962. pp. 93-106.
15. Zajcev A.A. Obzor jevoljucionnyh metodov optimizacii na osnove roevogo intelekta / A. A. Zajcev, V. V. Kurejchik, A. A. Polupanov // *Izvestija JuFU*. 2010. № 12 (113). pp. 7–12. (In Russ.).
16. Engelbrecht A.P. *Fundamentals of Computational Swarm Intelligence*. John Wiley & Sons, Chichester, UK, 2005.
17. Litovka N.V. Roevoj intellekt v zadachah optimal'nogo razmeshhenija ob'ektov prostranstvenno raspredelenogo predprijatija/ N.V. Litovka// *Nauchnye trudy KubGTU*, № 11, 2018. (In Russ.).
18. Dorigo M. *Swarm Intelligence, Ant Algorithms and Ant Colony Optimization // Reader for CEU Summer University Course «Complex System»*. Budapest, Central European University, 2001. pp. 1–38.
19. Caro G.D., Dorigo M. *Anet a Mobile Agents Approach to Adaptive Routing*. Technical Report IRIDA 97-12. IRIDA. Universite Libre de Bruxelles. Brussels, Belgium, 1997. 27 p.
20. Kurejchik V.V., Zaporozhec D.Ju. Roevoj algoritm v zadachah optimizacii // *Izvestija JuFU. Tehnicheskie nauki*. 2010. №7. pp. 28-32. (In Russ.).
21. Liping Zhu Remarkable problem-solving ability of unicellular amoeboid organism and its mechanism// Liping Zhu, Song-Ju Kim, Masahiko Hara and Masashi Aono, 2018 (<https://royalsocietypublishing.org/doi/10.1098/rsos.180396>).
22. Ezhov A.A. Doobuchenie nejronnoj seti Hopfilda: poisk global'nogo minimuma funkcionala i model' bystrogo sna// Ezhov A.A., Cherepnev A.S./ *Matematicheskoe modelirovanie*. 2009. T. 21. № 5. pp. 10-20. (In Russ.).
23. Loskutov A.I. Reshenie zadachi o rance na osnove dinamicheskoj nejronnoj seti Hopfilda// Loskutov A.I., Gorbulin V.I., Karpushev S.I., Rjahova E.A./ *Nelinejnij mir*. 2019. T. 17. № 3. pp. 25-35. (In Russ.).
24. Hajkin S. *Nejronnye seti. Polnyj kurs*. Izd. 2-e, ispr. M.: Vil'jams. 2016. 1104 p. (In Russ.).
25. Muzychin V.V. Issledovanie vozmozhnosti ispol'zovanija rekurrentnoj nejronnoj seti Hopfilda dlja reshenija zadachi kommivojazhera// Muzychin V.V., Macievskij S.V./

- Sovremennaja nauka: aktual'nye problemy teorii i praktiki. Serija: Estestvennye i tehničeskie nauki. 2020. № 5. pp. 93-99. (In Russ.).
26. Kubil V.N. Issledovanie i razrabotka metodov reshenija mnogokriterial'nyh zadach marshrutizacii transporta na osnove murav'inogo algoritma: dis. kand. t. nauk. Juzhno-Rossijskij gos. politehničeskij universitet imeni M.I. Platova, Novočerkassk, 2019. (In Russ.).
  27. Kubil V.N., Mohov V.A. Mnogokolonial'nyj murav'inij algoritm s modifikacijami dlja reshenija mnogokriterial'nyh zadach marshrutizacii transporta // Izvestija vysshih učebynyh zavedenij. Jelektromehanika, T. 61, № 6, 2018. pp. 94-109. (In Russ.).
  28. Migranov A.B., Darintsev O.V. Choosing a Swarm Algorithm to Synthesis an Optimal Mobile Robot Team Control Strategy // 2020 International Multi-Conference on Industrial Engineering and Modern Technologies, Vladivostok, Russia, 2020, pp. 1-5. Deb, Kalyanmoy Multi-Objective Optimization Using Evolutionary Algorithms. John Wiley & Sons, 2001.
  29. Lotov A.V. Mnogokriterial'nye zadachi prinyatiya reshenij: Učebnoe posobie/ A.V. Lotov, I.I. Pospelova. MAKS Press, 2008. 197 p. (In Russ.).
  30. Goldberg D. Genetic Algorithms in Search, Optimization, and Machine Learning/ D. Goldberg. – Massachusetts: Addison-Wesley, 1989. Kononov A.A. Ispol'zovanie metoda nejronnyh setej Hopfilda dlja reshenija zadachi marshrutizacii v seti// Kononov A.A/ Moskovskij jekonomičeskij zhurnal, №9, 2019. p.74(In Russ.).
  31. Panchenko T.V. Genetičeskie algoritmy: učebno-metodičeskoe posobie/ T.V. Panchenko; pod red. YU.YU. Tarasevicha. Astrahan': Izdatel'skij dom "Astrahanskij universitet", 2007. 87 p. (In Russ.).
  32. Migranov A.B., Darintsev O.V. The Use of Genetic Algorithms for Distribution of Tasks in Groups of Mobile Robots with Minimization of Energy Consumption // 2019 International Multi-Conference on Industrial Engineering and Modern Technologies, Vladivostok, Russia, 2019, pp. 1-6.
  33. Hopfield, J.J., Tank, D.W. (1985). Neural Computation of Decisions in Optimization Problems. Biological Cybernetics, Vol. 52, pp.141-152.
  34. Kononov A.A. Ispol'zovanie metoda nejronnyh setej Hopfilda dlja reshenija zadachi marshrutizacii v seti// Kononov A.A/ Moskovskij jekonomičeskij zhurnal, №9, 2019. p.74(In Russ.).
  35. Darintsev O.V. Migranov A.B. Using the Hopfield Neural Network to Select a Behaviour Strategy for the Group of Mobile Robots // IOP Publishing, 2021, J. Phys.: Conf. Ser. 2096 012086

**Darintsev Oleg** — Ph.D., Dr.Sci., Associate Professor, Chief researcher, Laboratory "robotics and control in technical systems", Mavlyutov Institute of Mechanics - Subdivision of the Ufa Federal Research Center of the Russian Academy of Sciences; Professor, Department of automation of technological processes, Ufa State Aviation Technical University. Research interests: robotics, microrobotics, MEMS, artificial intelligence, information processing systems, technical vision system. The number of publications — 230. ovd@uimech.org; 71, Prospekt Oktyabrya, 450054, Ufa, Russia; office phone: +7(347)225-5255.

**Migranov Ayrat** — Ph.D., Senior researcher, Laboratory "robotics and control in technical systems", Mavlyutov Institute of Mechanics - Subdivision of the Ufa Federal Research Center of the Russian Academy of Sciences. Research interests: robotics, microrobotics, artificial intelligence. The number of publications — 138. abm.imech.anrb@mail.ru; 71, Prospekt Oktyabrya, 450054, Ufa, Russia; office phone: +7(347)225-5255.

**Acknowledgements.** This research is supported by the framework of the state task (FWGZ-2019-0089).