Journal of King Saud University – Computer and Information Sciences 35 (2023) 283–291

HOSTED BY

ELSEVIER

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Evaluation of the iterative method of task distribution in a swarm of unmanned aerial vehicles in a clustered field of targets

Vyacheslav Petrenko [a], Fariza Tebueva [a], Vladimir Antonov [a,*], Sergey Ryabtsev [a], Artur Sakolchik [b], Dina Satybaldina [c]

[a] North-Caucasus Federal University, Russian Federation
[b] Belarusian State University, Belarus
[c] L.N. Gumilyov Eurasian National University, Republic of Kazakhstan

ARTICLE INFO

Article history:
Received 19 October 2022
Revised 20 February 2023
Accepted 24 February 2023
Available online 2 March 2023

Keywords:
Multi-robotic systems
Swarm of UAVs
Task allocation
Iterative method
Collective decision-making

ABSTRACT

The purpose of this article is to evaluate the performance of the developed iterative method for distributing tasks in a swarm of UAVs in a clustered target field. This article presents an iterative method for distributing tasks among agents of a swarm of unmanned aerial vehicles (UAVs) in a clustered target field, when the number of tasks significantly exceeds the number of agents. The main tasks solved by UAVs are surveillance and reconnaissance, detection of dangerous objects or places of emergency, search for victims, etc. The efficiency of solving the problems listed above is achieved by the simultaneous use of a group of UAVs, the elements (agents) of which can carry out the tasks of inspecting and scanning various areas in parallel. At the same time, the number of tasks can significantly exceed the number of UAV swarm agents. To organize the work of a swarm of UAVs in this case, it is necessary to solve the problem of labor division, considering the conditions of the problem. The results showed the high efficiency of the proposed task distribution method according to the criterion of minimizing the travel distance of agents.

1. Introduction

Robotic technologies are being introduced into all spheres of human activity. Robots are used for performing routine tasks, exploration, liquidation of natural and man-made emergencies, in agriculture, in geology, in counterterrorist operations, etc. At the same time, the intensive development of microelectronics has led to the miniaturization of robots and the ability to use groups of numerous robots – multi-robotic systems (MRS).

This article uses the terminology of group robotics introduced in publications (Kalyaev et al., 2009; Zakiev et al., 2018). The MRS group is understood as a homogeneous or heterogeneous group of agents with identical or different sets of specializations, structure, and sensory equipment. The advantages of MRS application are high mobility, low cost of maintenance, the ability to perform multiple tasks, as well as the scalability.

MRS groups are usually stochastic and non-linear, so the construction of mathematical models for testing and optimizing the control models is difficult. The lack of methods for the transition from the specific behavior of an agent to the universal behavior of a group does not allow to build an effective system for managing groups of robots (Chung et al., 2018). In this regard, a huge class of tasks for controlling MRS groups appears. One of these tasks is commonly known as the task of the labor division. Currently, many studies are devoted to the problem of tasks distribution between agents of a swarm of UAVs or agents of a group of MRSs. Many well-known scientists propose their own methods and algorithms for solving this problem.

An analysis of Pshikhopov et al. (2015) shows a wide variety of theoretical methods for solving this problem, especially for an equal number of agents and subtasks. By popularity, we can distinguish heuristic algorithms (Kowalczyk, 2002; Mathew et al., 2015), analytical algorithms (Chopra et al., 2017; Nam and Shell, 2015; Notomista et al., 2019; Zavlanos et al., 2008), algorithms based on market economy models (Bertsekas and Castañon, 1991; Luo et al., 2015; Zavlanos et al., 2008), potential field methods (Zavlanos and Pappas, 2008, 2007), probabilistic and random algorithms (Berman et al., 2009; Liu et al., 2020), methods based on machine learning and ANNs (Mouton et al., 2011; Zhao et al., 2021), fuzzy logic methods (Mukhedkar and Naik, 2013; Wei et al., 2021), ant colony algorithms (Brutschy et al., 2012; Liao et al., 2014; Oliveira et al., 2017; Payton et al., 2005, 2001; Yuan et al., 2008), dynamic and integer programming methods

* Corresponding author.
E-mail address: v.o.antonov@mail.ru (V. Antonov).

https://doi.org/10.1016/j.jksuci.2023.02.022
1319-1578/

(Murphey, 2000; Sikanen, 2008; Yu and LaValle, 2016), genetic algorithms (Patel et al., 2020; Shima et al., 2006; Soleimanpour-Moghadam and Nezamabadi-Pour, 2020), blockchain and cloud computing (Husheng et al., 2021; Msala et al., 2019), mixed algorithms (Zhang et al., 2012), particle swarm optimization (Kong et al., 2019; Wei et al., 2020), etc.

Research (Labella et al., 2007) carried out under the supervision of the Belgian scientist Dorigo is devoted to applications based on the analysis of the behavior of insect swarms for various tasks, including the division of labor in groups of small insects. Dorigo has a task distribution method that can adapt to dynamic changes in the configuration of agents. However, the issue of changing the configuration of the MRS group in the process of performing tasks is worthy of separate consideration and will be investigated for the proposed methods in the future.

Previously, in (Petrenko et al., 2020; Ryabtsev et al., 2022), a method was proposed for tasks distribution (division of labor) in a swarm of UAVs monitoring the dynamic zone of an emergency. This paper presents an adaptation of the earlier proposed method and a wide range of experimental studies of the effectiveness of the developed method. This article considers a particular case of labor division in a swarm of UAVs, when the number of tasks significantly exceeds the number of agents, and it is proposed to distribute the agents not by tasks, but by clusters of tasks.

As an analogue of the proposed iterative method, we consider the greedy task distribution algorithm due to its universality, convergence of the solution (Buffa et al., 2012), and wide usage for task distribution in homogeneous MRS groups, including UAV swarms. The idea of the greedy algorithm is to search for local optima, each time checking the admissibility of the current partial solution (Bouamama et al., 2022).

The purpose of this work is to develop an iterative method for distributing tasks in a swarm of UAVs, which would allow to match agents to clusters, because of a collective decision-making procedure. The criterion of matching is minimal travel distance of an agent. The method is used for building a flight task to perform multiple tasks grouped into clusters.

The structure of the article includes 5 sections. In Materials and Methods, the mathematical formulation of the research problem is given, and the description of the developed method and simulation environment for experimental studies is provided. In the results section, numerical indicators of the effectiveness of the proposed method are given in comparison with the analogue. The Conclusion section summarizes the results of the study.

2. Materials and methods

2.1. Mathematical formulation of the problem

Let there be n agents a_i of the set A and m tasks q_j of the set Q. The distribution of tasks consists in the performance of each agent a_i a certain number of subtasks q_j in such a way that all the subtasks of the set Q are completed in the minimum time t if the agents have some energy potential e_i. In this paper we consider the case when the energy potential of an agent is equivalent to the path traveled by the agent. Schematically, the input data looks like this (Fig. 1).

Mathematically, the problem statement can be presented in the following form. The set of the agents $\boldsymbol{a_i}$ of a swarm of UAVs is represented as $\boldsymbol{A} = [a_1; a_2; \cdots; a_i; \cdots a_n]$:

$$\boldsymbol{a_i} = [x_i; y_i; z_i; e_i], \tag{1}$$

where $x_i; y_i; z_i$ – current coordinates of the agent a_i a swarm of UAVs; e_i – energy potential of the agent a_i.

The set of subtasks $\boldsymbol{q_j}$ of global task $\boldsymbol{Q} = [q_1; q_2; \cdots; q_j; \cdots q_m]$ presented as:

$$\boldsymbol{q_j} = [x_j; y_j; z_j; e_j] \tag{2}$$

where $x_j; y_j; z_j$ – coordinates of the subtask; e_j – the energy reserve of the task.

Clusters of subtasks formed according to the geometric characteristics of the medium are represented as $\boldsymbol{W} = \{w_1; w_2; \cdots; w_k; \cdots w_c\}$, where k is the cluster number; c is the number of clusters. Clustering of the subtask field is performed by dividing the zone into cubes of the same size. Cluster of subtasks w_k after splitting includes some subtasks q_i.

Function e_{ij} characterizes the energy costs of the agent a_i to move to a subtask q_j.

The set $\boldsymbol{B} = [b_1; b_2; \cdots; b_p; \cdots b_u]$ of UAV swarm agent base stations, where p – number of the base station; u – number of stations ($u \geq 1$), characterized by coordinates, are represented as:

$$\boldsymbol{b_p} = [x_p; y_p; z_p]. \tag{3}$$

The result of the task allocation method is the mapping $\boldsymbol{R}$, matching each agent a_i of a swarm of UAVs with a unique task $q_j \in \boldsymbol{Q}$, the task of returning to the base $b_p \in \boldsymbol{B}$, or a waiting task $\varnothing$:

$$\boldsymbol{R} : \boldsymbol{A} \to \boldsymbol{Q} \cup \boldsymbol{B} \cup \varnothing. \tag{4}$$

Global task $\boldsymbol{Q}$ is considered completed (reflection F), if the current energy reserve of the subtasks $e_j = 0$ is equal to zero, provided that all agents a_i of the set $\boldsymbol{A}$ returned to the base station

$$\boldsymbol{F} : \boldsymbol{Q} = \sum_{k=1}^{K} e_k \to 0; \quad \forall a_i \in A\, R(a_i) \in \boldsymbol{B} \tag{5}$$

2.2. Basic task allocation method

The idea of the basic method of distributing tasks in a swarm of UAVs when solving multiple tasks is to carry out iterative procedures for establishing relationships of the "cluster-agent" type. In this paper, three variants of the iterative method of distributing clusters between agents are proposed: selection by agents a_i near clusters (1 V), selection by agents a_i distant clusters (2 V) and uniform distribution of clusters (3 V) between agents a_i. The difference between the variants of the method lies in the order in which clusters are determined for the selection of agents. In option 1 V, the tasks are considered q_j in clusters w_k from the nearest to the most distant from the launch site b_p agents a_i of a swarm of UAVs, in variant 2B, from the most distant to the closest clusters. In option 3B, a certain number of evenly distributed clusters are selected w_k in the goals field of the global task Q, equal to the number of agents. A schematic description of the variants of the task distribution method is presented in Fig. 2, where: (a) – the choice of near clusters; (b) – the choice of distant clusters; (c) – a uniform selection of clusters; (d) – a greedy algorithm for the distribution of tasks.

The algorithm of the basic method consists of 5 steps.

Step 1. Selecting clusters. Input data for agents a_i are the coordinates of the launch center b_p of agents, cluster centers w_p, task coordinates q_k. Additional agents a_i exchange their own coordinates $a_i = [x_i; y_i; z_i]$.

Regardless of the variant of the task distribution method, the vector lengths between the task clusters and the agent's base station are calculated:

$$L_{w_p} = \sqrt{\left(w_{p_x} - b_x\right)^2 + \left(w_{p_y} - b_y\right)^2 + \left(w_{p_z} - b_z\right)^2}.$$

Next, a ranking is made, and a list of the most remote and nearby clusters is compiled. For the variant of uniform distribu-

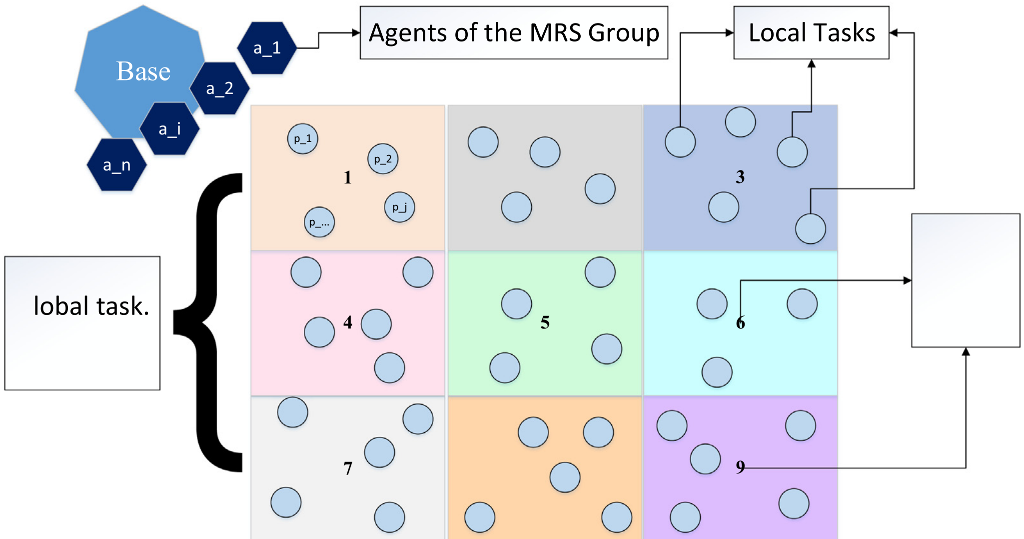

Fig. 1. The input data scheme of the proposed iterative method for distributing tasks in a swarm of UAVs.

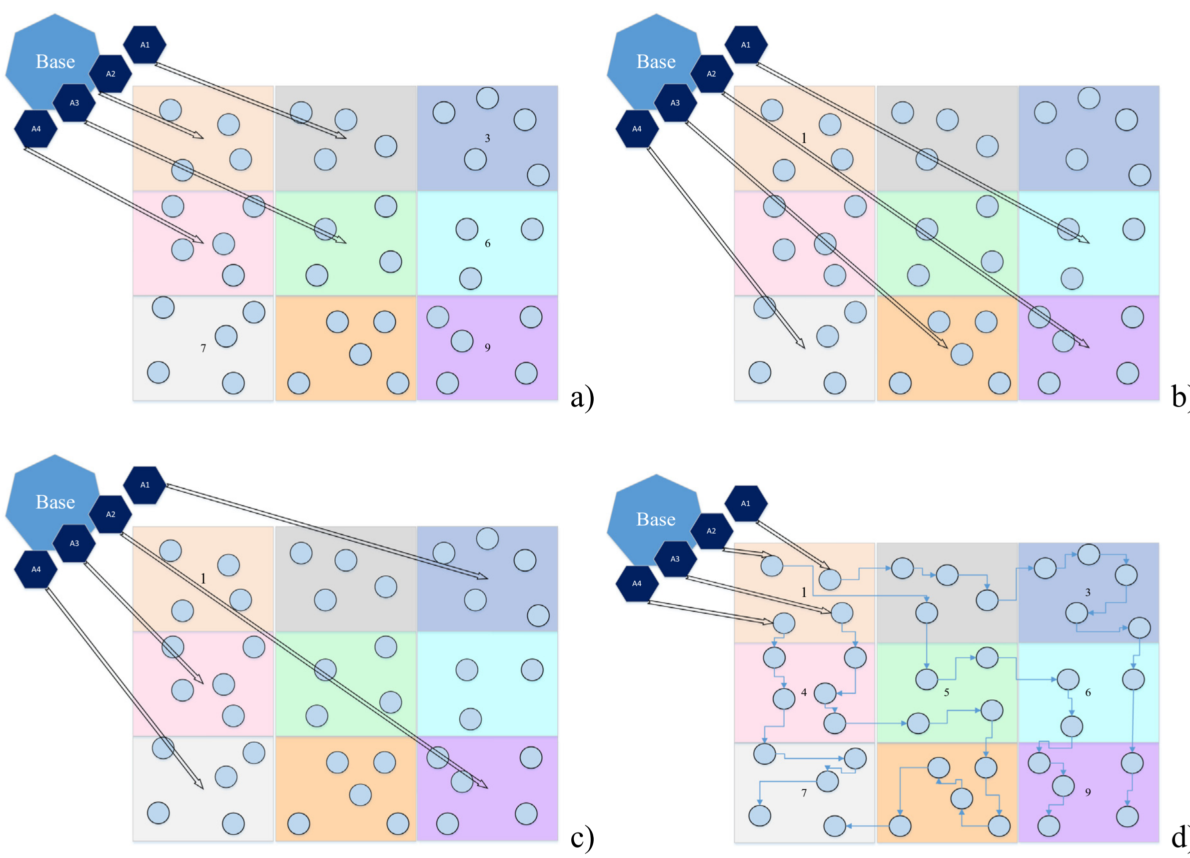

Fig. 2. Scheme of task distribution methods: (a) selection of near clusters; (b) selection of distant clusters; (c) uniform selection of clusters; (d) greedy task distribution algorithm.

tion of tasks, it is proposed to select clusters as follows. In the resulting list of cluster deletions, we calculate the length of the vector of all cluster tasks to the agent base station using an identical formula:

$$L_{q_{kw_i}} = \sqrt{(q_{ix} - b_x)^2 + \left(q_{iy} - b_y\right)^2 + (q_{iz} - b_z)^2}.$$

We select the most remote task in each cluster. Next, we compile a ranked list of the most remote tasks in each cluster. The resulting set of tasks from the list is distributed as follows. With the number of agents n, we select clusters with the first and last tasks in the list. We divide the number of remaining tasks $N_z - 2$ on the quantity $n - 2$ agents, $M_z = \frac{N_z - 2}{n - 2}$. The resulting value is rounded down to an integer and select each M_z the task from the end of the list. Thus, we get n number of tasks, each of which is tied to its own cluster. A set of these clusters will participate in the distribution.

Step 2. Cluster analysis. Depending on the selected method option (1B, 2B, and 3B), a certain cluster is received for evaluation by a group of agents W_n.

Agents calculate the metrics of efficiency and the ability to perform tasks in the cluster in the decision-making protocol, represented by the efficiency matrix F:

$$F = \begin{array}{c|cccc} & q_1^{w_k} & q_2^{w_k} & q_{\dots}^{w_k} & q_j^{w_k} \\ \hline a_1 & p_{11} & p_{12} & \cdots & p_{1j} \\ a_2 & p_{21} & p_{22} & \cdots & p_{2j} \\ a_{\dots} & \cdots & \cdots & \cdots & \cdots \\ a_i & p_{i1} & p_{i2} & \cdots & p_{ij} \end{array}$$

The decision-making protocols are identical and are found in all agents a_i. Performance metric p_{ij} is determined based on the energy potential of the tasks in the cluster and the agent's reserve as follows. The expected energy costs are calculated e_{ik} on moving the agent to the cluster and energy costs e_{ij_k} performing tasks in a cluster by solving a transport problem. If $e_{ik} + e_{ij_k} + e_b > e_i$, where e_b – the energy costs of returning the agent to the base point, then the agent refuses to perform the task

$$p_{ij} = \frac{e_i}{e_{ik} + e_{ij_k} + e_b}. \tag{6}$$

Step 3. Collective decision-making. After calculating the performance metrics, the agents a_i they begin the decision-making procedure. The procedure of collective decision-making uses the majority principle and includes 3 rounds.

In the first round the agents a_i swarms of UAVs launch decision-making protocols in which each agent sets its own metrics p_{ij} performing tasks in the cluster. The protocol is considered collected if it contains the number of metrics p_{ij}, equal to the product of the number of agents by the number of tasks in the cluster.

At the second round the agents each in their protocol supplement the performance matrix with a column containing the sum of performance metrics by rows $r_i^{w_k} = \sum_{j=1}^{n} p_{ij}$, $i = 1, \cdots, m$:

$$F' = \begin{array}{c|cccc|c} & q_1^{w_k} & q_2^{w_k} & q_{\dots}^{w_k} & q_j^{w_k} & r^{w_k} \\ \hline a_1 & p_{11} & p_{12} & \cdots & p_{1j} & r_1^{w_k} \\ a_2 & p_{21} & p_{22} & \cdots & p_{2j} & r_2^{w_k} \\ a_{\dots} & \cdots & \cdots & \cdots & \cdots & r_{\dots}^{w_k} \\ a_i & p_{i1} & p_{i2} & \cdots & p_{ij} & r_i^{w_k} \end{array}$$

The result of sum criterion for deciding on the choice of agent a_i to perform tasks in the cluster found by the formula:

$$r\prime_i^{w_k} = \max_i r_i^{w_k}. \tag{7}$$

The agent a_i, which corresponds to the value $r\prime_i^{w_k}$, is assigned the cluster w_k.

In the third round, agents check their records in the decision-making protocol. If the highest value of the product criterion does not relate to the ID of the i\agent of the UAV swarm, then the agent refuses to perform tasks in the cluster. If the agent ID matches the ID in the row with the maximum criterion, then the agent is assigned the cluster in question and the agent starts performing tasks. The cluster and the agent are excluded from the discussion for the duration of the tasks performed by the winning agent in the cluster in question. At the same time, the completion of each task is recorded by each agent and sent to the group.

Step 2 is repeated if there are free clusters and unoccupied agents.

Step 4. Performing tasks within the cluster. When assigning a cluster to an agent, the agent proceeds to the 2nd round - performing tasks within the cluster. The sequence of tasks in the cluster is determined by a simple iteration when $q_{j_{w_k}} < 10$, if $q_{j_{w_k}} > 10$ the method of simulated annealing is used to find the shortest way to perform tasks in a cluster.

Simulation of annealing in iterative problems can be used to approximate the global minimum of functions with many free variables.

The algorithm of the simulated annealing method (Wang et al., 2022) is probabilistic and shows good results in practice when solving NP-complete problems.

Let S be the set of states of the system, which in the physical sense reflect the function of energy consumption φ of agent a_i to move through tasks q_j in claster w_k. Power consumption function φ it calculated by the agent based on the characteristics of the agent, its condition and environmental parameters, i.e. how much energy the agent will spend on moving through tasks q_j of cluster w_k in the generated sequence of visiting tasks.

Function F based on the initial state s_v (where v – iteration step, 1, 2, 3...) generates a new candidate state s_{v+1}, which the system can switch to, or maybe discard, depending on t_v – system state temperature. Here is the algorithm of the method.

1. A random state is applied to the input s_v with the initial temperature $t_v = t_{max}$, the lower temperature limit t_{min};
2. While $t_v > t_{min}$:
   a. $s_{v+1} = F(s_v)$– starting the function of generating a new system state;
   b. $\Delta\varphi = \varphi(s_{v+1}) - \varphi(s_v)$;
   c. If $\Delta\varphi \le 0$, then $t_{v+1} = t_v \frac{\varphi(s_v)}{\varphi(s_{v+1})}$;
3. If $\Delta\varphi \le 0$ then the temperature drops: $t_{max} = t_{v+1}$. A new iteration is repeated, where the state is fed to the input s_{v+1} and $t_{max} = t_{v+1}$.
4. If $\Delta\varphi > 0$ then a new iteration is carried out with probability:

$$P(\Delta\varphi) = \exp^{\frac{-v}{t_{max} * \frac{\varphi\left(s_{v+1}\right)}{\varphi(s_v)}}}. \tag{8}$$

For the effective operation of the method, restrictions on the number of iterations are additionally introduced v.

Step 5. Iterativity. The procedure is repeated from the second step until the end of free clusters. When an agent is released from performing tasks in a cluster, according to the principle of the method, the agent independently assigns itself a cluster, notifying other agents of the establishment of a new cluster affiliation.

2.3. Software simulation

The software simulation was carried out in the CoppeliaSim system. To evaluate the effectiveness of the proposed solutions for tasks distribution in a swarm of UAVs, 20,000 computational experiments were conducted. The following methods are implemented in the simulation: an iterative method of task distribution; a variant with remote clusters; a variant with nearby clusters; a variant with distributed clusters and a greedy algorithm for task distribution (Montenegro et al., 2022).

UAV swarm agents were implemented in a three-dimensional environment (Fig. 3). In the lower figure, the calculated flight paths of the agents are highlighted in color, the agents are marked, and

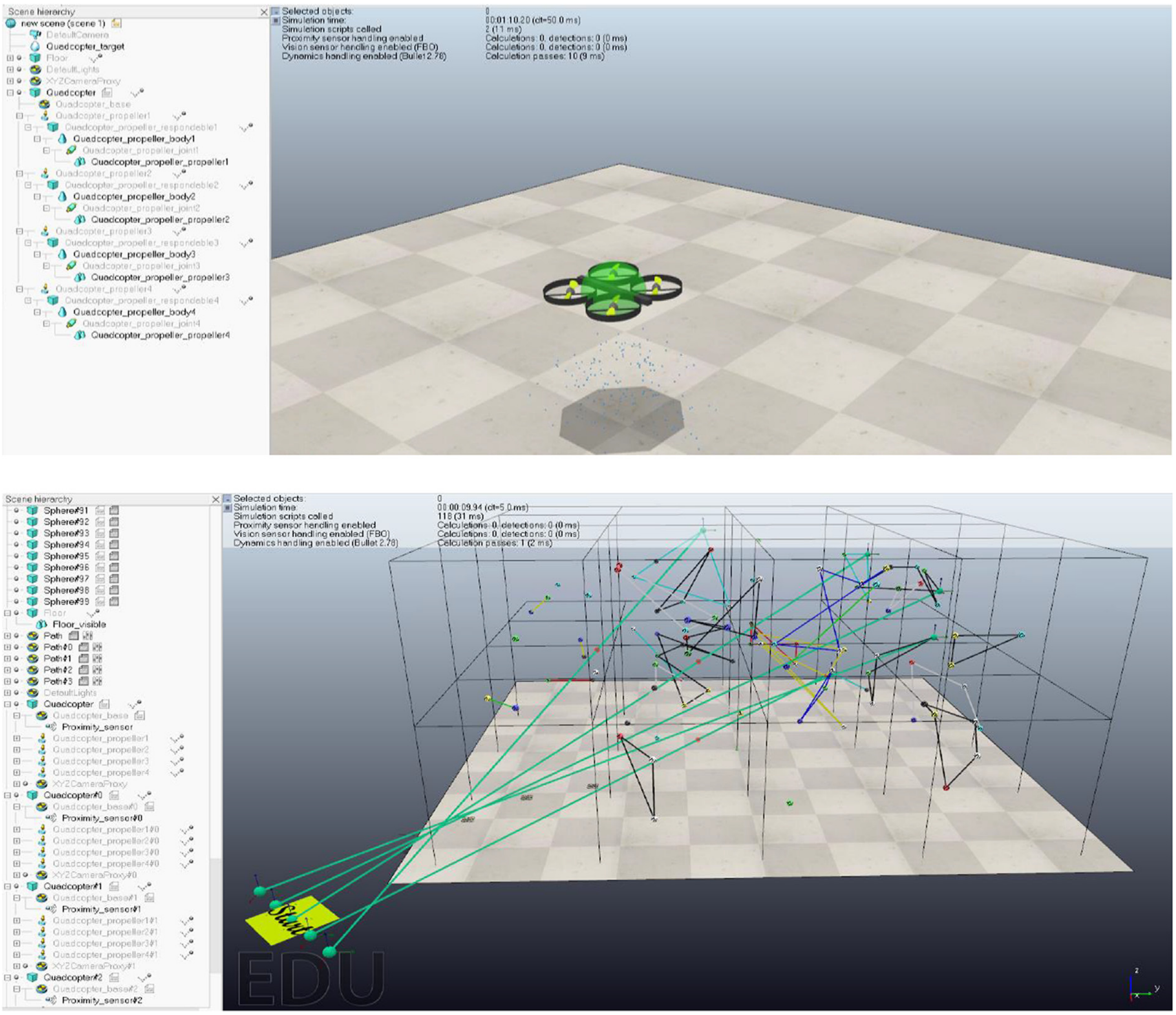

Fig. 3. An example of a software implementation of an iterative task distribution method in CoppeliaSim.

Table 1
Average time for UAV swarm agents to complete tasks in clusters.

Average time and cluster size	Option of distant clusters	A variant of near clusters	Uniform distribution option	Greedy algorithm
5 × 18 × 250	43.56	47.74	43.04	**38.80**
5 × 32 × 250	40.25	39.04	39.49	**38.80**
5 × 50 × 250	39.49	40.08	40.10	**38.80**
7 × 18 × 250	36.06	38.37	39.78	**24.42**
7 × 32 × 250	34.68	35.64	33.22	**24.42**
7 × 50 × 250	33.50	34.09	35.03	**24.42**
10 × 18 × 250	29.81	34.12	26.29	**24.02**
10 × 32 × 250	29.43	32.51	29.52	**24.02**
15 × 18 × 250	28.63	30.99	25.50	**21.03**
15 × 32 × 250	25.21	26.65	25.94	**21.03**

the launch points are green. The points in the cluster field are indicated by color spheres, and the clusters are highlighted as cubes.

The simulation was performed for 100 tasks for swarm sizes of 5, 7, 10 and 15 agents. Initial coordinates of agents and tasks were generated randomly with a uniform distribution on the map. The cluster sizes were 18, 32 and 50 tasks. The results were summed up for each generated map. In total, 250 maps were generated for one set of agents and clusters.

3. Results and discussion

To evaluate the effectiveness in the software environment, the distance traveled by all agents to complete tasks and the map execution time were measured. The results of the task completion time and distance traveled for 5, 7, 10 and 15 agents when dividing the task field into 18, 32 and 50 clusters are presented in Tables 1 and 2. The tables in bold indicate the best values of time or dis-

Table 2
Average distance travelled when UAV swarm agents perform tasks in clusters.

Average time and cluster size	Option of distant clusters	A variant of near clusters	Uniform distribution option	Greedy algorithm
5 × 18 × 250	717.68	647.11	701.54	**596.13**
5 × 32 × 250	707.25	**487.76**	582.73	596.13
5 × 50 × 250	708.37	**501.00**	585.10	596.13
7 × 18 × 250	819.80	**752.10**	800.71	927.53
7 × 32 × 250	791.44	**629.82**	699.79	927.53
7 × 50 × 250	723.58	**603.83**	650.08	927.53
10 × 18 × 250	974.50	**910.94**	1226.57	951.11
10 × 32 × 250	924.57	**803.26**	815.64	951.11
15 × 18 × 250	1196.70	**1165.47**	1226.82	1264.70
15 × 32 × 250	1187.22	**1107.00**	1153.00	1264.70

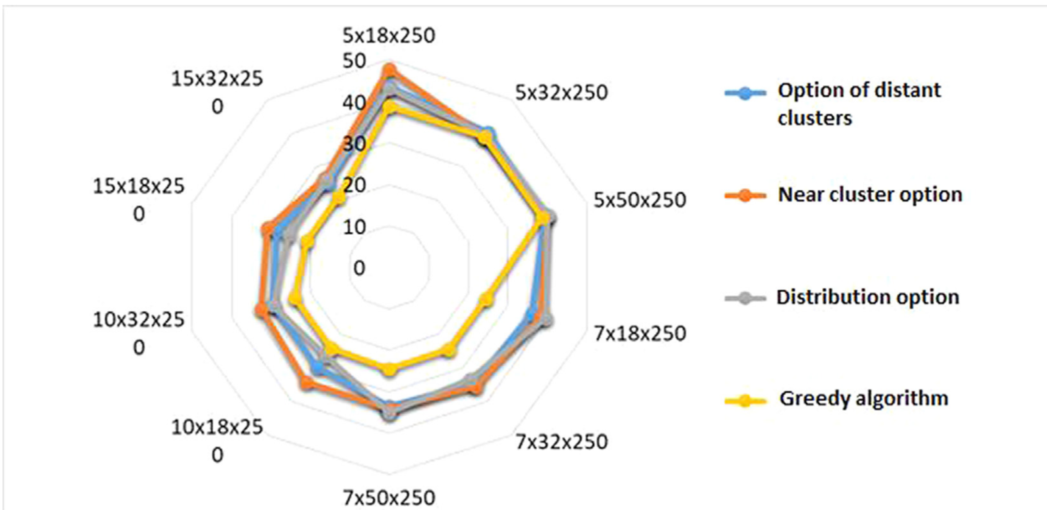

Fig. 4. A petal diagram of the total execution time by agents when performing tasks with the studied variants of the method.

tance for the method of distributing tasks in a swarm of UAVs. Figs. 4 and 5 show the results of the studies presented by the petal diagrams. The proximity of the contour of the diagram to the center of the petal reflects the best results of the experiments.

Based on the results of the study, it can be noted that according to Table 1, the greedy task allocation algorithm shows the best results in terms of task execution time.

The time efficiency of the greedy task allocation algorithm is from 2% to 34% reduction in time relative to the iterative method.

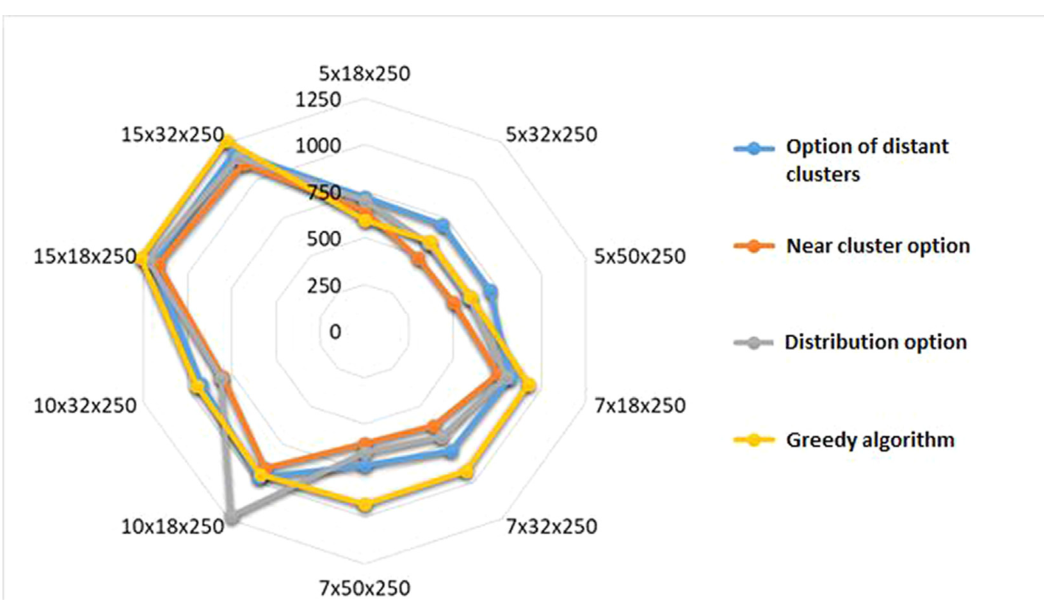

Fig. 5. A petal diagram of the total distance traveled by agents when performing tasks by the methods under study.

At the same time, it is worth noting that the optimal energy efficiency is provided by a variant of the near cluster method. The reduction of energy consumption by the method of near clusters relative to the greedy algorithm for the distribution of tasks is up to 28%, which is a scientific increment obtained because of the study.

Additionally, the standard deviations of the presented variants of the iterative method considered. The standard deviation will allow you to assess how variable the effectiveness of the distribution of tasks. Tables 3 and 4 show the standard deviations of the experimental results in terms of time and distance travelled by agents. The standard deviation diagrams are shown in Figs. 6 and 7.

The graphs of standard deviations show that the method of near clusters, when demonstrating the best results in minimizing the distance travelled by agents, has a large spread of values in the presented set with the average value of the set. The distributed cluster method shows the best results of the standard deviation for estimating the distance travelled by agents. It can be concluded that the effectiveness of the distribution of tasks according to the method of nearby clusters is the most volatile, while, even in conditions of increased variability, the method is able to provide optimal energy efficiency.

According to the results of experimental researches special attention should be paid to the method of distant clusters, which, in terms of the global task execution time and energy efficiency, shows worse results compared to the method of near and distributed clusters. Figs. 4 and 5 show that the far cluster method shows average results between task execution time and energy efficiency. The method of distant clusters will be of interest in the division of labor in a rarefied field of tasks with a limited energy supply of agents. In the future, studies will be carried out on the effectiveness of this method when performing the most remote tasks. This article shows that the application of the method of distant clusters in the conditions of uniformly distributed tasks in the field of goals is not effective.

This article describes and evaluates the effectiveness of an iterative method of tasks distribution in a swarm of MRS with a

Table 3
Standard deviations in the time when agents perform tasks in clusters.

Mean square deviations in time and cluster size	Variant of distant clusters	Variant of near clusters	Variant of uniform distribution	Greedy algorithm
5 × 18 × 250	1.43	3.23	1.51	**1.10**
5 × 32 × 250	0.99	4.17	**0.85**	1.10
5 × 50 × 250	0.87	2.37	**0.64**	1.10
7 × 18 × 250	1.18	1.41	1.65	**0.89**
7 × 32 × 250	2.23	3.89	2.30	**0.89**
7 × 50 × 250	**0.86**	1.61	1.15	0.89
10 × 18 × 250	2.44	2.22	1.42	**0.60**
10 × 32 × 250	1.33	0.88	**0.29**	0.60
15 × 18 × 250	1.98	2.98	0.84	**0.61**
15 × 32 × 250	1.06	1.05	**0.57**	0.61

Table 4
Standard deviations in distance when agents perform tasks in clusters.

Mean square deviations in time and cluster size	Variant of distant clusters	Variant of near clusters	Variant of uniform distribution	Greedy algorithm
5 × 18 × 250	**13.79**	19.04	15.12	27.19
5 × 32 × 250	19.13	32.34	**16.74**	27.19
5 × 50 × 250	**18.92**	44.54	19.05	27.19
7 × 18 × 250	23.64	16.90	**12.83**	19.77
7 × 32 × 250	37.00	33.97	33.16	**19.77**
7 × 50 × 250	20.65	22.36	24.30	19.77
10 × 18 × 250	28.29	**14.82**	15.45	30.18
10 × 32 × 250	19.60	**10.30**	25.20	30.18
15 × 18 × 250	47.05	19.38	**9.70**	31.09
15 × 32 × 250	45.09	**7.01**	51.85	31.09

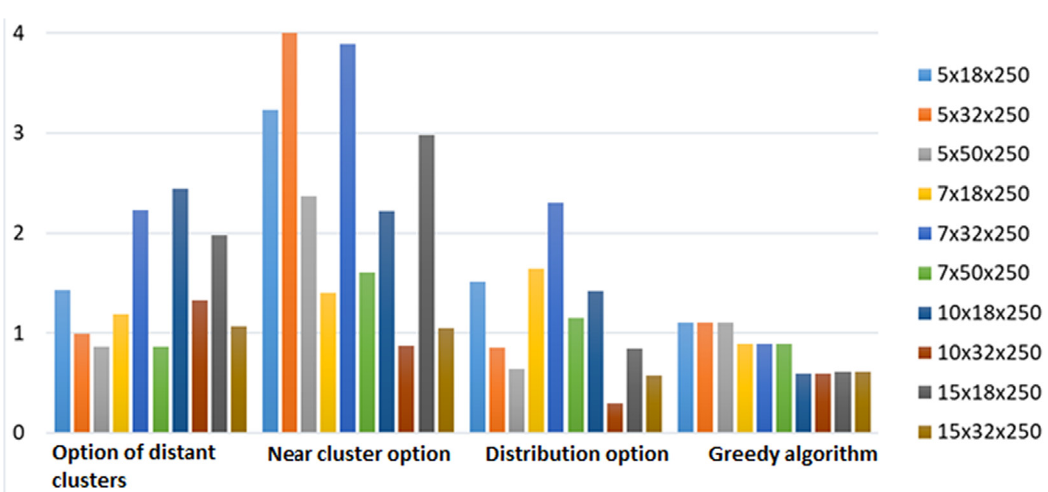

Fig. 6. A bar chart of standard deviations in the time of execution of tasks by agents using the methods under study.

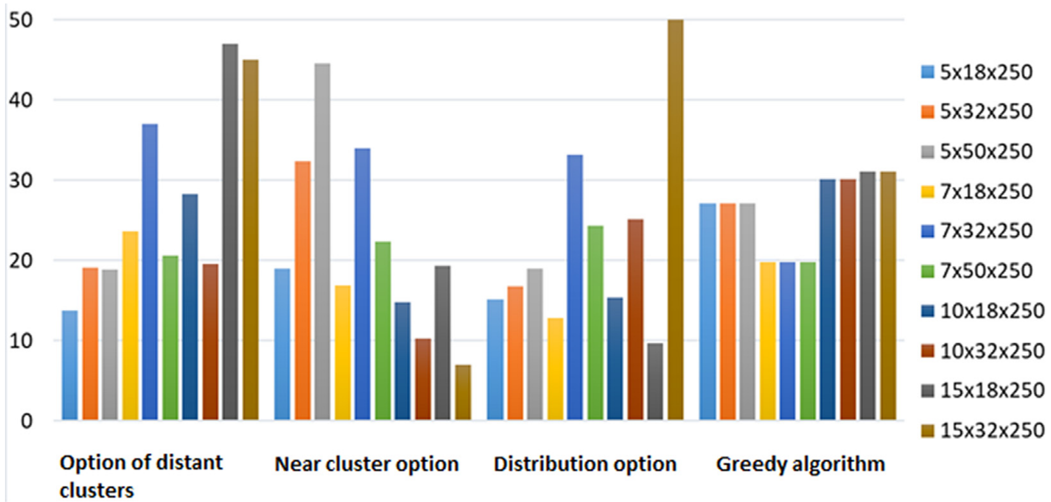

Fig. 7. A bar chart of standard deviations by the total distance traveled by agents when performing tasks by the methods under study.

significant excess of the number of tasks over the number of agents (by 5–20 times). Three variants of this method were proposed, which differ in the order of choosing clusters of tasks for establishing the "cluster-agent" relationship. To evaluate the effectiveness of the proposed methods, we compared them to a greedy task distribution algorithm. To conduct the experimental studies, we implemented a simulation model of a group of UAVs in the CoppeliaSim environment (Elmokadem and Savkin, 2021). Based on the results of 20,000 experiments, we conclude that it is promising to use the iterative method from near clusters to reduce the energy expended by agents. The travel distance-wise efficiency of the near cluster variant relative to the greedy task distribution algorithm is up to 28% higher, depending on the number of agents and tasks in the cluster, which is a significant advantage. At the same time, in comparison with the greedy algorithm, the variant of near clusters receives a loss in task execution time by an amount from 2% to 34%.

4. Conclusions

Thus, understanding the effectiveness of the proposed methods and measures of variability of feature values, in subsequent studies, it is possible to develop an algorithm for choosing a method for distributing tasks in a swarm of UAVs, which will allow choosing the method with the highest efficiency indicators for performing a global task in a clustered field of targets.

The analysis of the standard deviations of the results showed the acceptable stability of the variant of distributed clusters.

In the continuation of the research, we propose to develop a methodology for choosing the method of tasks distribution in the MRS group, which will allow choosing the method with the highest efficiency indicators for fulfilling global targets. We plan to refine the iterative method of distributing tasks in the MRS group to perform heterogeneous tasks with importance criteria by introducing "master–slave" subsystems of agents.

The proposed task distribution methods are able to adapt to changing the configuration of a group of agents by redistributing task clusters in the process of their execution. The effectiveness of the proposed methods, taking into account the limitation of the energy reserve of agents under conditions of a change in the configuration of a group of agents (a decrease or increase in their number), is planned to be studied in detail in the following works. An analogue of the proposed research will be algorithms inspired by natural systems.

5. Contributors

All authors contributed equally. All authors approved the final article.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work supported by a grant from the President of the Russian Federation for young scientists - candidates of science (No. MK-300.2022.4 Development of methods and algorithms for the UAV swarm control system when performing heterogeneous tasks).

References

Berman, S., Halasz, A., Hsieh, M.A., Kumar, V., 2009. Optimized Stochastic Policies for Task Allocation in Swarms of Robots. IEEE Trans. Rob. 25, 927–937. https://doi.org/10.1109/TRO.2009.2024997.

Bertsekas, D.P., Castañon, D.A., 1991. Parallel synchronous and asynchronous implementations of the auction algorithm. Parallel Comput. 17, 707–732. https://doi.org/10.1016/S0167-8191(05)80062-6.

Bouamama, S., Blum, C., Pinacho-Davidson, P., 2022. A Population-Based Iterated Greedy Algorithm for Maximizing Sensor Network Lifetime. Sensors (Basel, Switzerland) 22 (5), 1804. https://doi.org/10.3390/s22051804.

Brutschy, A., Scheidler, A., Ferrante, E., Dorigo, M., Birattari, M., 2012. "Can ants inspire robots?" Self-organized decision making in robotic swarms, in: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, pp. 4272–4273. https://doi.org/10.1109/IROS.2012.6386273.

Buffa, A., Maday, Y., Patera, A.T., Prud'Homme, C., Turinici, G., 2012. A priori convergence of the Greedy algorithm for the parametrized reduced basis method. ESAIM: Math. Modell. Numer. Anal. 46 (3), 595–603. https://doi.org/10.1051/m2an/2011056.

Chopra, S., Notarstefano, G., Rice, M., Egerstedt, M., 2017. A Distributed Version of the Hungarian Method for Multirobot Assignment. IEEE Trans. Rob. 33, 932–947. https://doi.org/10.1109/TRO.2017.2693377.

Chung, S.-J., Paranjape, A.A., Dames, P., Shen, S., Kumar, V., 2018. A Survey on Aerial Swarm Robotics. IEEE Trans. Rob. 34, 837–855. https://doi.org/10.1109/TRO.2018.2857475.

Elmokadem, T., Savkin, A.V., 2021. Towards Fully Autonomous UAVs: A Survey. Sensors (Basel, Switzerland) 21 (18), 6223. https://doi.org/10.3390/s21186223.

Husheng, W., Hao, L., Renbin, X., 2021. A blockchain bee colony double inhibition labor division algorithm for spatio-temporal coupling task with application to UAV swarm task allocation. J. Syst. Eng. Electron. 32, 1180–1199. https://doi.org/10.23919/JSEE.2021.000101.

Kalyaev, I.A., Gaiduk, A.R., Kapustyan, S.G., 2009. Models and Algorithms of Collective Control in Groups of Robots. FIZMATLIT, Moscow.
Kong, X., Gao, Y., Wang, T., Liu, J., Xu, W., 2019. Multi-robot Task Allocation Strategy based on Particle Swarm Optimization and Greedy Algorithm, in: 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC). IEEE, pp. 1643–1646. https://doi.org/10.1109/ITAIC.2019.8785472.
Kowalczyk, W., 2002. Target assignment strategy for scattered robots building formation, in: Proceedings of the Third International Workshop on Robot Motion and Control, 2002. RoMoCo'02. Poznan Univ. Technol, Poznan, Poland, pp. 181–185. https://doi.org/10.1109/ROMOCO.2002.1177105.
Labella, T.H., Dorigo, M., Deneubourg, J.L., 2007. Self-Organised Task Allocation in a Group of Robots. In: Alami, R., Chatila, R., Asama, H. (Eds.), Distributed Autonomous Robotic Systems, 6. Springer, Tokyo. https://doi.org/10.1007/978-4-431-35873-2_38.
Liao, T., Socha, K., Montes de Oca, M.A., Stutzle, T., Dorigo, M., 2014. Ant Colony Optimization for Mixed-Variable Optimization Problems. IEEE Trans. Evolut. Comput. 18, 503–518. https://doi.org/10.1109/TEVC.2013.2281531.
Liu, R., Seo, M., Yan, B., Tsourdos, A., 2020. Decentralized task allocation for multiple UAVs with task execution uncertainties, in: 2020 International Conference on Unmanned Aircraft Systems (ICUAS). IEEE, pp. 271–278. https://doi.org/10.1109/ICUAS48674.2020.9213989.
Luo, L., Chakraborty, N., Sycara, K., 2015. Provably-Good Distributed Algorithm for Constrained Multi-Robot Task Assignment for Grouped Tasks. IEEE Trans. Rob. 31, 19–30. https://doi.org/10.1109/TRO.2014.2370831.
Mathew, N., Smith, S.L., Waslander, S.L., 2015. Planning Paths for Package Delivery in Heterogeneous Multirobot Teams. IEEE Trans. Autom. Sci. Eng. 12, 1298–1308. https://doi.org/10.1109/TASE.2015.2461213.
Montenegro, G., Chacón, R., Fabregas, E., Garcia, G., Schröder, K., Marroquín, A., Dormido-Canto, S., Farias, G., 2022. Modeling and Control of a Spherical Robot in the CoppeliaSim Simulator. Sensors (Basel, Switzerland) 22 (16), 6020. https://doi.org/10.3390/s22166020.
Mouton, H., Roodt, J., le Roux, H., 2011. Applying Reinforcement Learning to the Weapon Assignment Problem in Air Defense. Scientia Militaria – S. Afr. J. Military Stud. 39, 1–15. https://doi.org/10.5787/39-2-115.
Msala, Y., Hamlich, M., Mouchtachi, A., 2019. A new Robust Heterogeneous Multi-Robot Approach Based on Cloud for Task Allocation, in: 2019 5th International Conference on Optimization and Applications (ICOA). IEEE, pp. 1–4. https://doi.org/10.1109/ICOA.2019.8727618.
Mukhedkar, R., Naik, S., 2013. Weapon Target Allocation Problem Using Fuzzy Model. IJAIEM 2, 279–289.
Murphey, R.A., 2000. Target-Based Weapon Target Assignment Problems, in: Pardalos, P.M., Pitsoulis, L.S. (Eds.), Nonlinear Assignment Problems. pp. 39–53. https://doi.org/10.1007/978-1-4757-3155-2_3.
Nam, C., Shell, D.A., 2015. Assignment Algorithms for Modeling Resource Contention in Multirobot Task Allocation. IEEE Trans. Autom. Sci. Eng. 12, 889–900. https://doi.org/10.1109/TASE.2015.2415514.
Notomista, G., Mayya, S., Hutchinson, S., Egerstedt, M., 2019. An Optimal Task Allocation Strategy for Heterogeneous Multi-Robot Systems, in: 2019 18th European Control Conference (ECC). IEEE, pp. 2071–2076. https://doi.org/10.23919/ECC.2019.8795895.
Oliveira, S., Hussin, M.S., Roli, A., Dorigo, M., Stutzle, T., 2017. Analysis of the population-based ant colony optimization algorithm for the TSP and the QAP, in: 2017 IEEE Congress on Evolutionary Computation (CEC). IEEE, pp. 1734–1741. https://doi.org/10.1109/CEC.2017.7969511.
Patel, R., Rudnick-Cohen, E., Azarm, S., Otte, M., Xu, H., Herrmann, J.W., 2020. Decentralized Task Allocation in Multi-Agent Systems Using a Decentralized Genetic Algorithm, in: 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, pp. 3770–3776. https://doi.org/10.1109/ICRA40945.2020.9197314.
Payton, D., Daily, M., Estowski, R., Howard, M., Lee, C., 2001. Pheromone robotics. Auton. Robots 11, 319–324. https://doi.org/10.1023/A:1012411712038.
Payton, D., Estkowski, R., Howard, M., 2005. Pheromone Robotics and the Logic of Virtual Pheromones. In: Lecture Notes in Computer Science. Springer-Verlag, Berlin, pp. 45–57. https://doi.org/10.1007/978-3-540-30552-1_5.
Petrenko, V.I., Tebueva, F.B., Ryabtsev, S.S., Gurchinsky, M.M., Struchkov, I.V., 2020. Consensus achievement method for a robotic swarm about the most frequently feature of an environment. IOP Conf. Ser. Mater. Sci. Eng. 919,. https://doi.org/10.1088/1757-899X/919/4/042025 042025.
Pshikhopov, V.K., Soloviev, V.V., Titov, A.E., Finaev, V.I., Shapovalov, I.O., 2015. Group Control of Moving Objects in Uncertain Environments. FIZMATLIT.
Ryabtsev, S., Sakolchik, A., Antonov, V., Petrenko, V., Tebueva, F., Makarenko, S., 2022. Iterative Method of Labor Division for Multi-Robotic Systems. Proceedings of International Conference on Artificial Life and Robotics, 27, pp. 699–702. https://doi.org/10.5954/ICAROB.2022.OS17-7.
Shima, T., Rasmussen, S.J., Sparks, A.G., Passino, K.M., 2006. Multiple task assignments for cooperating uninhabited aerial vehicles using genetic algorithms. Comput. Oper. Res. 33, 3252–3269. https://doi.org/10.1016/j.cor.2005.02.039.
Sikanen, T., 2008. Solving Weapon Target Assignment Problem with Dynamic Programming, in: Independent Research Projects in Applied Mathematics.
Soleimanpour-Moghadam, M., Nezamabadi-Pour, H., 2020. Discrete Genetic Algorithm for Solving Task Allocation of Multi-robot Systems, in: 2020 4th Conference on Swarm Intelligence and Evolutionary Computation (CSIEC). IEEE, pp. 006–009. https://doi.org/10.1109/CSIEC49655.2020.9237316.
Wang, Z., Wang, Y., Xu, H., Xie, H., 2022. Application of Simulated Annealing Algorithm in Core Flow Distribution Optimization. Energies 15, 8242. https://doi.org/10.3390/en15218242.
Wei, T., Yongjiang, H., Yuefei, Z., Wenguang, L., Xiaomeng, Z., 2021. Multi-UAV Task Allocation Based on Type Mamdani Fuzzy Logic, in: 2021 7th International Symposium on Mechatronics and Industrial Informatics (ISMII). IEEE, pp. 184–187. https://doi.org/10.1109/ISMII52409.2021.00046.
Wei, C., Ji, Z., Cai, B., 2020. Particle Swarm Optimization for Cooperative Multi-Robot Task Allocation: A Multi-Objective Approach. IEEE Robot. Autom. Lett. 5, 2530–2537. https://doi.org/10.1109/LRA.2020.2972894.
Yu, J., LaValle, S.M., 2016. Optimal Multirobot Path Planning on Graphs: Complete Algorithms and Effective Heuristics. IEEE Trans. Rob. 32, 1163–1177. https://doi.org/10.1109/TRO.2016.2593448.
Yuan, M., Ling, M.-X., Zeng, Q.-S., 2008. An AntColony Algorithm Based on Pheromone Declining for Solving the WTA Problem. Int. J. Comput. Simulat. 25, 23–25.
Zakiev, A., Tsoy, T., Magid, E., 2018. Swarm Robotics: Remarks on Terminology and Classification, in: Third International Conference, ICR 2018. Leipzig, Germany, pp. 291–300. https://doi.org/10.1007/978-3-319-99582-3_30.
Zavlanos, M.M., Pappas, G.J., 2007. Sensor-Based Dynamic Assignment in Distributed Motion Planning, in: Proceedings 2007 IEEE International Conference on Robotics and Automation. IEEE, pp. 3333–3338. https://doi.org/10.1109/ROBOT.2007.363987.
Zavlanos, M.M., Spesivtsev, L., Pappas, G.J., 2008. A distributed auction algorithm for the assignment problem, in: 2008 47th IEEE Conference on Decision and Control. IEEE, pp. 1212–1217. https://doi.org/10.1109/CDC.2008.4739098.
Zavlanos, M.M., Pappas, G.J., 2008. Dynamic Assignment in Distributed Motion Planning With Local Coordination. IEEE Trans. Rob. 24, 232–242. https://doi.org/10.1109/TRO.2007.913992.
Zhang, J., Wang, X., Xu, C., 2012. ACGA Algorithm of Solving Weapon - Target Assignment Problem. Open J. Appl. Sci. 02, 74–77. https://doi.org/10.4236/ojapps.2012.24B018.
Zhao, H., Dorigo, M., Allwright, M., 2021. General Dynamic Neural Networks for the Adaptive Tuning of an Omni-Directional Drive System for Reactive Swarm Robotics, in: 2021 25th International Conference on Methods and Models in Automation and Robotics (MMAR). IEEE, pp. 79–84. https://doi.org/10.1109/MMAR49549.2021.9528468.