International Journal of Intelligent Robotics and Applications (2021) 5:49–64
https://doi.org/10.1007/s41315-020-00158-9

REGULAR PAPER

A multi-robot task allocation algorithm based on universal gravity rules

Mohadese Soleimanpour-moghadam[1] · Hossein Nezamabadi-pour[2]

Received: 8 January 2020 / Accepted: 26 November 2020 / Published online: 20 February 2021

Abstract
In this paper, a new multi-robot task allocation (MRTA) algorithm inspired by the Newtonian law of gravity is proposed. In the proposed method, targets and robots are considered as fixed objects and movable objects, respectively. For each target, a constant mass is assigned, which corresponds to its quality. The fixed objects (which refer to targets) apply a gravitational force to the movable objects (which are considered as robots) and change their positions in the feasible search space and therefore, the best target allocation of robots is determined by employing the law of gravity. In the proposed scenario, task allocation consists of assigning the robots to the found targets in a 2-D feasible area. The expected distribution is obtained from the targets' qualities that are represented as scalar values. Decision-making is a distributed mechanism and robots choose their assignments, taking into account targets' qualities and distances. Moreover, a control parameter is planned to make a remarkable balance between exploration and exploitation ability of the proposed algorithm. A self-adaptive mechanism is proposed to adjust the value of the exploration parameter automatically, aiming to maintain the balance between exploration and exploitation ability of robots. Furthermore, in order to decrease the time of reaching the target and accelerate computation, a selection memory is designed. In the experiments, we examine the scalability of the proposed method in terms of the number of robots and the number of targets and speed of algorithm to deliver robots to the desired targets with comparison to other competitors. The simulation results show the scalability of the algorithm, comparing the existing methods. Moreover, some non-parametric statistical tests are utilized to compare the results obtained in experiments. The statistical comparisons confirm the superiority of the proposed method compared over the existing methods.

Keywords Multi-robot systems · Law of gravity · Scalability · Swarm intelligence · Task allocation

1 Introduction

Multi-robot systems (MRSs) are a group of robots that are designed to perform some collective behavior. By this collective behavior, some goals that are impossible for a single robot to achieve become feasible and attainable. One of the most important reasons that the topic of MRSs has become more popular is the various future advantages of MRSs compared with single robot systems. These benefits include, but are not limited to, decreased task complexity, improving the system's performance, decreasing the completion time for the defined tasks, improving reliability and simplicity in design (Nunes et al. 2017).

These benefits have attracted many researchers from academia and industry to investigate the applicability of MRSs in many applicable areas of industrial and commercial importance such as intelligent security, search and rescue, surveillance, and health care (Koes et al. 2006; Jahanshahi et al. 2017).

In order to develop and deploy robust MRSs in real-world applications, several challenging problems need to be solved. These problems include, but are not limited to, task allocation, group formation and self-organization, to name just a few (Schwarzrock 2018; Wang et al. 2016). In this paper, the task allocation problem as one of the challenging problems of MRSs is discussed in detail.

✉ Mohadese Soleimanpour-moghadam
m.soleimanpour@bam.ac.ir

Hossein Nezamabadi-pour
nezam@uk.ac.ir

1 Department of Mechanical Engineering, Higher Education Complex of Bam, Bam, Iran

2 Intelligent Data Processing Laboratory (IDPL), Department of Electrical Engineering, Shahid Bahonar University of Kerman, Kerman, Iran

Multi-robot task allocation (MRTA) problem can be seen as an optimal assignment problem where the objective is to optimally assign a set of robots to a set of tasks in such a way that optimizes the overall system performance subject to a set of constraints (Lee 2018).

To study the MRS task allocation problem as an optimization problem, at first fitness function (or cost function) and the problem constraints should be defined.

Given a robot r and a task t, if r is capable of executing t, then one can define, the values Q_{rt} and C_{rt} are defined by the robot as the quality and cost of doing the task. It should be noted that the quality of a target is an application-specific scalar value that may represent the target's priority or complexity, where a higher value requires more robots to be allocated. For example, it could represent the number of injured people in need of assistance in urban search and rescue scenarios. As another example, it could represent the richness of the mineral or water source on a planet that we want to harness. The medium by which these values are obtained is not considered in this paper. In other words, the quality values of the targets are deemed to be known and given.

Moreover, the cost of doing the task could be considered as distance or time of doing a task or the power that will be required to move and do a particular task.

Hence, we first need to express the unit definition for the fitness or utility function as follows (Gerkey and Matarić 2004):

$$U_{rt} = \begin{cases} Q_{rt} - C_{rt} \; \textit{if the robot r has the ability to perform the taskt} \\ -\infty \; \textit{otherwise} \end{cases} \tag{1}$$

where $U_{rt} = Q_{rt} - C_{rt}$ if the robot can perform the task t. Otherwise, the value of $-\infty$ is attributed to the fitness function (Gerkey and Matarić 2004).

In order to solve the optimization problem, different methods have been proposed that can be categorized in general terms into three groups of the mathematical methods (Ren 2017), the use of game theory (Jang et al. 2018; Tang and Parker 2007), and using of heuristic search algorithms (Jang et al. 2018).

Most of the mathematical methods use the set theory principle to solve MRTA problems. Because of computational complexity, the mathematical problems have been less welcomed. The second one is based on game theory, which can be referred to as a distributed market-based approach (Kanakia et al. 2016). From another perspective, the use of heuristic or meta-heuristic algorithms is utilized considering the simplicity and acceptable accuracy of solving problems, and somewhat due to the similarity to multi-robot systems.

Each MRTA method has its advantages and disadvantages in task allocation of different types of problems. Therefore, many MRTA methods have been proposed to rectify the disadvantages of these algorithms. Also, some researchers tried to suggest new algorithms inspired by nature. In this paper, a nature inspired MRTA algorithm is introduced by employing the Newtonian law of gravity. In the following, the related works are reviewed.

1.1 Related works

By reviewing the literature, it was found that different optimization approaches have been used in order to solve the general task allocation problems and MRTA problem.

Gerkey and Mataric (Gerkey and Matarić 2004) provide a taxonomy for MRTA problems based on the number of tasks, the number of robot, and the schedule for allocation. Based on Gerkey and Matarić (2004), there is some group such as Single-Task or Multiple-Task (ST-MT) robots, Single-Robot or Multiple-Robot (SR-MR) Tasks, and Instantaneous or Time extended Allocation (IA-TA).

Authors in Khamis et al. (2015) describe three of the most commonly used MRTA approaches: namely game theory-based approaches, mathematical optimization-based approaches, and heuristic-based approaches. Market-based approach as a game theory approach gained considerable attention within the robotics research community because of several desirable features, such as efficiency in satisfying the objective function, robustness, and scalability (Jang et al. 2018; Zlot and Stentz 2006). The market-based approach, which can be considered as a game theory approach, is an economically inspired approach that provides a way to coordinate the activities between robots/agents. Market-based approach is mainly based on the concept of auctions. Based on economic theory principles, an auction is defined by any mechanism of trading rules for exchange (Lagoudakis, et al. 2006; Guerrero and Oliver 2003). In other words, an auction can be considered a process of assigning a set of goods or services to a set of bidders according to their bids and the auction criteria. It can be concluded that auctions are simple and conventional ways of performing resource allocation in a multi-agent system.

It is worth mentioning that although market-based approaches have many advantages, they are not without their disadvantages. The lack of formalization in designing appropriate cost and revenue functions to capture design requirements can be considered the biggest drawback of market-based approaches (Korsah et al. 2013; Zlot et al. 2002).

Mathematical optimization-Based Approaches are the branch of applied mathematics focusing on solving a specific problem to find the optimum solution for this problem out of a set of possible solutions (Parker and Tang 2006). A set of possible solutions is restricted by a set of constraints, and the optimum solution is chosen within these constrained solutions according to specific criteria. This criterion defines the objective function of the problem that quantitatively describes the system (Lerman 2002). There is a wide variety of optimization approaches available, and the use of these approaches depends on the nature and the degree of complexity of the problem to be optimized (Lenagh 2013). Deterministic techniques include numerical and classical methods such as graphical methods, gradient, and Hessian based methods, derivative-free approaches, quadratic programming, sequential quadratic programming, penalty methods, etc. (Balas and Padberg 1976). In Atay and Bayazit (2006), a mixed-integer linear programming optimization approach was used in order to allocate heterogeneous robots for maximizing the coverage area of the regions of interest. Also, in Darrah et al. (2005), a mixed-integer linear programming approach was used for solving the task allocation problem in the context of UAV cooperation.

More and more attention has been given to using nature-based inspired algorithms to solve MRTA problems (Mosteo 2010). Moreover, many MRTA methods have been made by hybridizing different types of evolutionary algorithms (Yi 2017).

Heuristic search algorithms always have some randomness. These techniques can be classified into trajectory-based and population-based algorithms. A trajectory-based meta-heuristic algorithm, such as Simulated Annealing (SA) uses a single agent or solution which moves through the design space or search space in a piece-wise style. On the other hand, population-based algorithms such as Genetic Algorithms (Gas), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Gravitational Search Algorithm (GSA) and Immune Optimization Algorithm (IOA) use multiple agents to search for an optimal or near-optimal solution (Benabderrahmane 2017). In Mosteo and Montano (2006), SA approach was used to solve the allocation of MRS through formulating the MRTA problem as a Travel Salesman Problem (TSP). In Juedes et al. (2004) and Kmiecik et al. (2010), SA incorporated with other heuristic approaches was used to allocate a set of tasks to several processors in computer system problems.

The GA was used in Shea et al. (2003) to provide a feasible solution for group tracking, which is capable of tracking several targets rather than individual targets. GA was also used in Jones et al. (2011) to provide a solution for the time extended task allocation of multi-robots in an application of simulated disaster scenarios. ACO algorithm as another technique of the population-based optimization approaches was used in Zlot and Stentz (2006) to solve the task allocation problem of MRSs. In Ding et al. (2003), ACO was used in the context of multi-robot cooperation to solve the task allocation problem. The task allocation problem was also solved using hybrid optimization approaches such as tabu search with random search method in Liu and Kulatunga (2007). In Liu and Kulatunga (2007), a simultaneous approach for solving the path planning and task allocation problems for an MRS is proposed, where SA and ACO approaches were investigated and applied for solving the problem. In Huang et al. (2018), a niching immune-based optimization algorithm based on Softmax regression (sNIOA) is presented to handle MRTA problem. Furthermore, a guiding mutation operator inspired by the base pair in the theory of gene mutation is introduced into sNIOA to strengthen its search ability.

In Jevtic et al. (2011), the Distributed Bee Algorithm (DBA) has been used to solve MRTA. In Jevtic et al. (2011), a scenario consisting of a community of homogeneous robots in terms of hardware and software (or some distinct groups of robots), the number of targets and their location in the environment are identified for robots. Also, each robot is unaware of the size of the community and the distribution of other robots, and each robot at any moment can perform a maximum of one of the tasks. In Tkach et al. (2018) the Modified Distributed Bee Algorithm (MDBA) has been used to solve MRTA problem for the general problem of heterogeneous robots and target quality.

In summary, the scenario can be defined as finding targets with the least energy loss due to the unnecessary rotation of robots in the environment and the absorption of robots in proportion to the target's values.

1.2 Challenges and motivation

Some of the significant challenges in MRTA algorithms are the ability to deal with the variable number of robots and targets as well as imbalanced targets qualities i.e. the scalability of the algorithm. The aim of this paper is to propose a nature-inspired MRTA algorithm, which can overcome some problems like imbalanced targets qualities, different numbers of robots and sensitivity to the initial position of robots. In the proposed MRTA algorithm, we consider the targets as fixed celestial objects with the pre-determined mass (based on quality of target) to apply a gravitational force to movable objects (robots) and change their positions in the feasible search space. The aim is to find the best allocation of robots where each robot is modeled by a movable agent with the unity mass. The robots move around the feasible search space in the influence of the gravitational force exerted by the celestial objects to find the best position. One can expect that the robots assign optimally to the targets. Moreover, in order to decrease time of delivering robots to targets and accelerate procedure of algorithm, selection memory is designed.

The rest of the paper is organized as follows. In the next Section, the basic concepts of multi-robot task allocation and the Newton's law of universal gravitation are reviewed and their properties are discussed. Section 3 is devoted to describing the proposed gravity algorithm. A comparative experimental study is given in Sect. 4. Finally, we conclude the paper in Sect. 5.

2 Basic concepts

2.1 Problem definition

Based on the represented taxonomy in Gerkey and Matarić (2004), the multi-robot system, which is used in this paper, is categorized as homogeneous and distributed, using broadcast communication. In other words, we tend to address the problem of single-task robots, multi-robot tasks and instantaneous assignment (ST-MR-IA) (Gerkey and Matarić 2004). In the utilized task allocation scenario, the setting that contains variety of tasks is considered that could be of same or completely different importance and robots that are equally capable of performing each task but each one can only be assigned to one at any given time. It should be noted that the quality of a target is a scalar value that represents the target's priority or complexity, where a higher value requires more robots to be allocated. In this paper, we do not consider how these values are obtained. The proposed scenario in this paper is found under the following assumptions:

- All the targets are made accessible to all the robots. To reach this aim, the broadcast communication range setting of the robots cover the entire feasible search space.
- Robots make decisions among all the targets in the feasible area simultaneously. The total number of targets and the positions of the targets are saved in robots' internal memory and they are changed based on the experimental setup.
- Reallocation to a different target is not allowed for each time step of the robot's movement.

Note that the above assumptions are considered for simplicity and can be modified based on problem definition.

Consider a population of R robots to be allocated among T targets. Let $Q \in \{q_1, \ldots, q_T\}$ represents the set of normalized qualities of all accessible targets, n_t^{itr}, $t \in \{1, \ldots, T\}$ as a nonnegative integer shows the number of robots allocated to target t in iteration of itr (i.e. time slot of robot movement).

Note that, the normalized qualities of all targets are evaluated as follows:

$$q_t = \frac{Q_t}{\sum_{t=1}^{T} Q_t} \tag{2}$$

where Q_t is the quality of t-th target and T is the number of targets.

It should be considered that for the proposed scenario, the combined utilities of the robots are unknown as robots have no knowledge of the decisions taken by other robots, therefore, the system optimization based on the maximum utility cannot be applied.

2.2 Newton's law of universal gravitation

The gravitation is the tendency of masses to accelerate toward each other. Newtonian law of universal gravity is one of the most important laws in all scientific subjects. Newtonian law of universal gravity states that any two objects in the universe attract each other with a force that is directly proportional to the product of their objectives and inversely proportional to the square of the distance between them (Benabderrahmane 2017; Rashedi et al. 2009). This is a general physical law derived from empirical observations by what Newton called induction. Every point mass attracts every single other point mass by a force pointing along the line intersecting both points as follows:

$$F(m_1, m_2) = G\frac{m_1 m_2}{d^2} \tag{3}$$

where F is the gravitational force between two point objects, m_1 and m_2 are the masses of two objects, d is the Euclidean distance between m_1 and m_2, and G is the gravitational constant.

3 Gravity based MRTA algorithm: GBMRTA

In order to start the basic idea of paper, consider a population of R movable robots to be allocated among T fixed targets. Let $Q \in \{q_1, \ldots, q_T\}$ denotes the set of normalized qualities of all available targets. Moreover, it is assumed that each individual robot is located at position P_r, in a 2-dimensional search space. The main idea in the proposed algorithm is to consider a movable gravity object (robot) as a movable mass and each target as a fixed gravity object. In this gravity system, the fixed objects (i.e. targets) apply the gravitational force to the movable objects (i.e. robots) and change their positions in the feasible search space. The proposed algorithm is able to deal with different numbers of robots and targets without any sensitivity to their initial distributions and it has an excellent performance to handle unbalanced targets. This is due to the nature of Newton's gravity law in which the gravitational force between two objects is inversely proportional to the distance between them. Therefore, the initial distributions of robots and targets have no significant effects on the final allocation of robots to targets. Furthermore, the comprehensive description of the problem definition is depicted in Fig. 1.

In this figure, two types of objects are shown in stars, and circles instances for the robots (or movable objects/agents) and targets (fixed objects/masses), respectively. Arcs out of the star also represent gravitational forces that are applied by

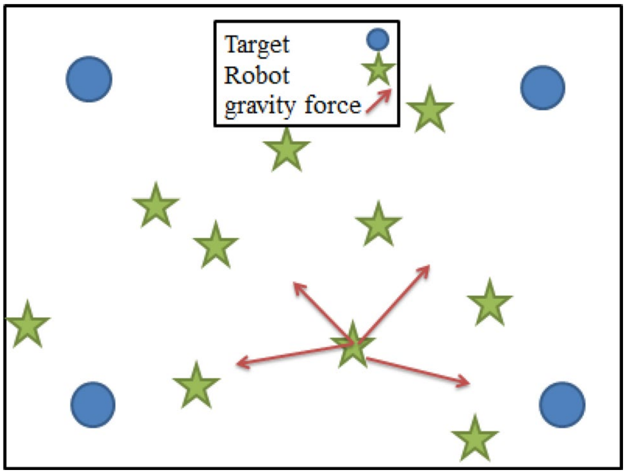

Fig. 1 Applied gravitational force to each agent, circles show targets and stars show robots

fixed objects. Decision making of each robot is proportional to the value of the total force exerted on the agent (robot) by the fixed objects (targets). It is expected that agents move towards the fixed mass of gravity and stop in an area where the robot reaches the selected target. It should be noted that targets as fixed objects are not allowed to apply force to each other.

The proposed MRTA algorithm is explained with the following steps:

Step 1: Initialize the algorithm parameters including the number of robots (R), the number of targets (T), mass of targets (m_t), maximum number of iteration (ITR) (total time), parameters used in the law of gravity (G_0) and robot position $P = \{P_1, \dots, P_R\}$.

Step 2: Generate randomly positions initial of R robots $P = \{P_1, \dots, P_R\}$ and set mass values of these objects to one. In other words, the robots positions $\left\{P_1, \dots, P_R\right\}$ are randomly initialized where P_r is a 2-dimensional parameter in search space which is shown by (x_r, y_r).

Step 3: Repeat the following until the maximum number of iterations is reached.

3-1: The gravitational force applied to the robot r from targets t is computed as follows:

$$F_r^t = G_t \frac{m_t m_r}{(d_r^t)^{\varphi} + \varepsilon} \tag{4}$$

$$d_r^t = \sqrt{\left(x_t - x_r\right)^2 + \left(y_t - y_r\right)^2} \tag{5}$$

where (x_t, y_t) and (x_r, y_r) represent target's and robot's coordinates in the arena, respectively, and m_t and m_r present mass values of the fixed object t (target) and robot r, respectively; m_t is normalized target quality and m_r is set to 1 for $r = 1, \dots, R$, d_r^t is the Euclidean distance between agent r and target t defined as Eq. (5) and φ is a parameter of the algorithm which tunes the effect of distance on the calculation of the force. Moreover G_t is a parameter of the algorithm which tunes the exploration/exploitation of algorithm. Therefore, in our proposed algorithm, the gravitational parameter G_t is considered as an adaptive controlling parameter of exploration and exploitation. To design an acceptable gravitational parameter G_t two ways are proposed which is explained later.

3-2: All calculated forces to each robot r from targets in step 3–1 (Eq. 4) is normalized as follows:

$$f_r^t = \frac{F_r^t}{\sum_{t=1}^{T} F_r^t} \tag{6}$$

3-3: The wheel-selection rule is applied as a decision-making mechanism of the proposed GBMRTA.

In wheel-selection rule, as in all selection methods, f_r^t assigns probability value to possible targets. This fitness level is used to associate a probability of selection with each individual target. In other words, If F_r^t is the effected force of target t to robot r then the probability of being selected for target t is f_r^t.

It should be considered that if selection procedure is done in each iteration, it will confuse the robot and increases the number of iterations that is needed to reach the target. To solve this issue, the selection procedure is applied every Φ iteration which named selection memory of algorithm. Note that in order to increase the performance of selection memory approach, Φ is considered as an adaptive value and evaluate as follow:

$$\Phi = round\left(\Phi i * \left(\frac{\Phi f}{\Phi i}\right)^{\frac{itr}{ITR}}\right) \tag{7}$$

where Φi and Φf illustrate the selection memory of algorithm in the first iteration and in the last iteration respectively (by the condition of $\Phi i \ll \Phi f$) and itr and ITR show the current iteration and the maximum number of iteration, respectively. Based on simulation results, the selection memory approach causes a significant decrement in iterations which is needed to reach the desired target. It can be concluded that the computational procedure can be accelerated through the way of decreasing the time through the selection memory approach.

3-4: Then, each agent (robot) r must move toward selected target ST_r based on its determined speed.

3-5: repeat the above steps until the stopping criterion is met.

Output of GBMRTA algorithm: optimal partitions $ST\{ST_1, \dots, ST_R\}$.

In this algorithm, if premature convergence happens, there will not be any recovery for the algorithm. In other words, after becoming converged, the algorithm loses its ability to explore and then becomes inactive.

Heuristic algorithms must have an acceptable balance between exploration and exploitation (also termed diversification/ intensification) to achieve both efficient global and local searches. In this way, they can efficiently solve optimization problems. As another point of view, losing the balance between exploration and exploitation may lead to premature convergence. Exploration is the ability to investigate the search space for finding new and better solutions, and exploitation is the ability to look for the optimal solution near a good one. The abilities of exploration and exploitation in every heuristic algorithm are applied with specific operators. Since each operator has its abilities of exploration and exploitation, the operators should be artfully hybridized together for a good trade-off between exploitation and exploration.

In simple words, all robots are randomly distributed in a feasible environment in the first iteration. In the following, based on the position and the quality of each target and also by applying the proposed GBMRTA algorithm, each robot selects a target. In the next iteration, all robots move one step toward the selected target and send the report to the selected targets. By collecting the information of robot in the targets, the c_t as a percent of the selected target $t \in \{1, ..., T\}$ by robots is calculated for each target. Moreover, the control parameter of G_t is evaluated.

By applying the new control parameter of G_t the applying force by targets are modified in order to decrease the difference between m_t (expected percent of the selected target) and c_t.

In can be concluded that, G_t is the control parameter that allows us to bias the decision-making mechanism toward the quality of the solution or its cost, respectively.

In this paper, to find the best definition of G_t parameter two main approaches are proposed that are discussed in Appendix A. Fractional and Exponential definition are two proposed approach to have a balance between exploration and exploitation. Based on experimental results in Appendix A, fractional formulation has better performance in comparison with exponential formulation. Consequently, $G_t = (\frac{m_t}{c_t+\varepsilon})^{1.35}$ is considered as the adaptive formulation of gravitational parameter, ε is a small positive number to avoid dividing by zero, c_t is the percent of selected target or the resulting robots' distribution and m_t is expected percent of selected target or expected robots' distribution.

In simple words, if $c_t > m_t$ which means that target t attract robots more than it's requiring. Then $G_t = (\frac{m_t}{c_t+\varepsilon})^{1.35}$ is decreased in order to decrease the discrepancy between the expected (m_t) and the resulting robots' distribution (c_t). Moreover, if $c_t < m_t$ which means that target t attract robots less than it's requirement. Then $G_t = (\frac{m_t}{c_t+\varepsilon})^{1.35}$ is increased in order to attract more robot to target t.

Algorithm1 illustrate the pseudo code of proposed GBMRTA algorithm.

Algorithm 1 The GBMRTA algorithm

1. Randomly initializes a population of R robots and initialize the algorithm parameters
2. **while** stop criterion is not satisfied do
   a. The gravitational force applied to the robot is computed by equation (4).
   b. The normalized forces are calculated by equation (6).
   c. The decision-making mechanism is applied to select target for each robot through selection memory every Φ iteration by equation (7).
   d. Update G_t
3. **return** the best solution found so far,

Some notes regarding the proposed algorithm should be mentioned as below:

- The final robots' distribution and target selection depend on their initial distribution in the feasible search space, i.e. their distances from each target prior to the target allocation.
- Due to the distributed nature of scenario, combined robots utility cannot be computed, as a result, the quality of the targets is used as the only measure for the expected robots' distribution.
- Due to the force applying to each agent (robot) from its neighborhood fixed points (targets), it can attract space around itself.
- To compute the force acting on the agent (robot) in Eq. (4), a control term G_t, has been adjusted. We use this parameter to control the balance between exploration and exploitation of the algorithm. This will helps the algorithm to escape from a local optimum, so the dependency on the initial robots' distribution is reduced.
- Control parameter of G_t adjusts the accuracy of the search. In other words, this parameter tries to help the algorithm to reach the determined quality of targets.
- In order to avoid collision between robots, a communication range is considered for each robot. The main idea of the proposed target search algorithm is that if two or more robots be in the communication range of each other, the nearest robot to the target position will get the highest priority. Moreover, when two or more robots have the same distance with their selected target position, one of them will get the highest priority randomly. It should be added that the non-prioritized robot will be idle until the prioritized robot passes away from its communication range. If the idle time is equal to an iteration, the robot rediscovers the target during this iteration; otherwise, it moves toward the allocated target in the current iteration.
- We assume that a robot is able to check for collisions between its own planned path and another robot's path. To facilitate this, we assume all their clocks are synchronized. Messaging delay can be accommodated; however, it must be negligible with respect to robot dynamics. We assume that robot detections are always happened when they come into communication range.

3.1 Fault handling

Each of the local and global activities in MRS might suffer from different faults, which disrupts the entire MRS. Some faults which an MRS might encounter can be defined as individual robot malfunctions, local perspectives that are globally incoherent, inter-robot interference, software errors or incompleteness, and communications failures.

In this paper, we focus on two fundamental types of faults, which are at the core of the "multi" aspect of multi-robot systems: *the global fault* and *local fault*.

- **Global fault handling.**

  In the global scope, all robots are randomly distributed in a feasible environment in the first iteration. Based on the position and the quality of each target and by applying the proposed GBMRTA algorithm, each robot selects a target. In the next iteration, all robots move one step toward the selected target and send the report to the selected targets. By collecting the information of robot in the targets, the c_t as a percent of the selected target $t \in \{1, \ldots, T\}$ by robots is calculated for each target.

  It should be noted that c_t and m_t (expected percent of the selected target) are not the same, which causes the global fault in the MRS. To solve this issue, a control parameter of G_t is proposed in this paper.

  By applying the control parameter of G_t the applying force by targets are modified in order to decrease the difference between m_t (expected percent of the selected target) and c_t.
- **Local fault handling.**

As a local scope in MRS, for a variety of reasons, there exists the probability of collision between robots. In order to avoid collision between robots and based on practical considerations, a communication range is considered for each robot. The main idea of the proposed target search algorithm is that if two or more robots are in the communication range of each other, the nearest robot to the target position will get the highest priority. Moreover, when two or more robots have the same distance with their selected target position, each robot will randomly get the highest priority.

Nonetheless, to some extent, the general elements of fault handling depicted in the following figure (Fig. 2).

In the following section the performance of proposed algorithm is evaluated.

4 Experimental evaluation

In the following, we describe the evaluation criteria, simulation environment and experimental results.

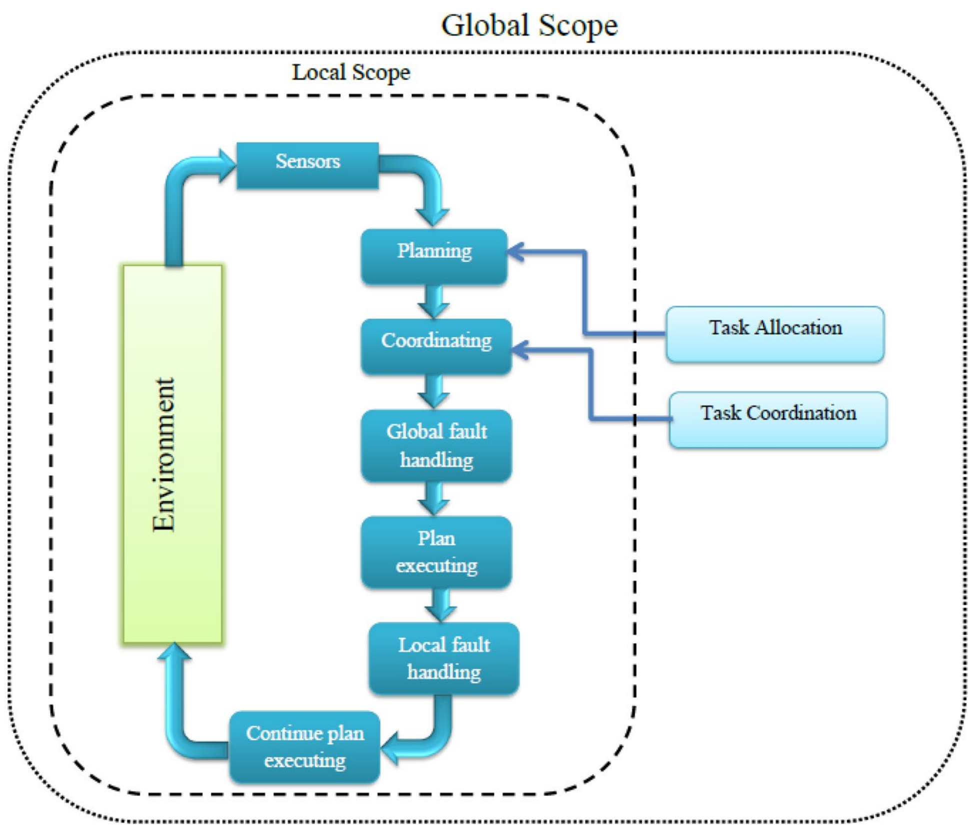

Fig. 2 The general elements of fault handling in the proposed method

Table 1 Parameters describing the feasible search space and the experimental duration in Experiments

Area dimensions (m^2)	1.5×2.125	Target 1 location $y(y, y)$ (m)	(−0.45, 0.75)
Experiment duration (time steps)	400	Target 2 location (x, y) (m)	(0.45, −0.75)
Time step duration (s)	0.1	Target 3 location (x, y) (m)	(−0.45, −0.75)
Number of iteration	4000	Target 4 location (x, y) (m)	(0.45, 0.75)

4.1 Evaluation criteria

The following performance measures were analyzed to compare the different allocation algorithms:

- Tasks completion time or average iteration to reach target (AIT).

The overall task completion time Γ is defined as an average of the individual tasks completion iterations:

$$\Gamma = \frac{1}{R}\sum_{r=1}^{R}\gamma_r \tag{8}$$

where γ_r is the number of iteration that robot r is needed to reach the selected target. In simple words, the task completion time is derived by the average amount of time elapsed from robots that arrive at the selected target. From another point of view, the average iteration to reach the target (AIT) has a direct relation with the number of fitness function evaluations.

The number of fitness function evaluations (FEs) can be considered as an evaluation criterion for heuristic search algorithms. This practice has been advocated in a number of competitions to compare the performance of population-based algorithms and has been used in many articles that contain empirical comparisons of algorithms. In this paper, average FEs or AIT is considered as a computational criterion for comparing different methods.

- Mean absolute error (MAE) of the final robots' distribution

As mentioned before due to distributed nature of scenario, it is impossible to define the combined robots utility. Therefore, we define the mean absolute error (MAE) of the final robots' distribution, which is given by:

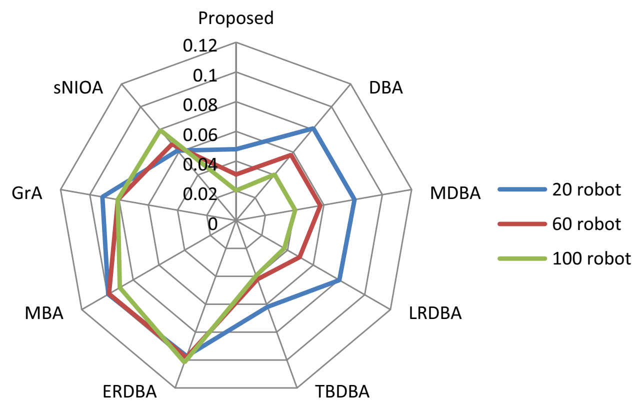

Fig. 3 Radar (spider) chart illustrates the MAE comparison for the task allocation problem of two targets with the same quality values

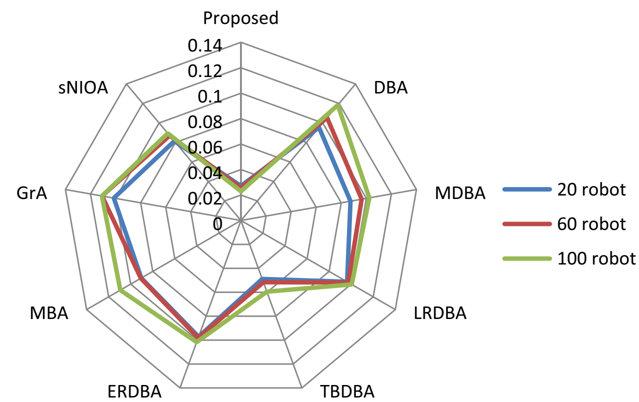

Fig. 4 Radar (spider) chart illustrates the MAE comparison between 8 methods and proposed GBMRTA method for the task allocation problem of two targets with different quality values.

$$MAE = \frac{1}{T}\sum_{t=1}^{T} |c_t - m_t| \tag{9}$$

As the name suggests, the mean absolute error is the average value of the absolute distribution error (per target) that is the result of discrepancy between the expected (m_t) and the resulting robots' distribution (c_t).

4.2 Simulation environment and experimental results

Without loss of generality, our simulation platform is a specialized multi-robot simulator for the e-puck robots described in Jevtic et al. (2011). The e-puck is a small cylindrical wheeled mobile robot that holds eight IR[1] proximity sensors distributed around the body. The e-puck can be modeled as a cylindrical object of 3.5 cm in radius. Based on IR proximity sensors, the communication range of the e-puck Range was set to cover the whole feasible search space. Note that the feasible search space is considered as a rectangle-shaped $1.5 \times 2.125\ m^2$ simulation area, but it can be easily extended to relatively complicated geometries.

In order to have a thorough comparison on the performance of algorithm, the proposed GBMRTA is compared to the original DBA (Jevtic et al. 2011), MDBA (Tkach et al. 2018), LRDBA, ERDBA, TBDBA, a Greedy Algorithm (GrA) (Broadcast of Local Eligibility for Multi-Target Observation 2002), a Market-Based Algorithm (Zlot et al. 2002) and NIOA (Huang et al. 2018) which were implemented to fix the task allocation problem, as described in Appendix B.

In Table 1, the feasible search space and the experimental duration are defined.

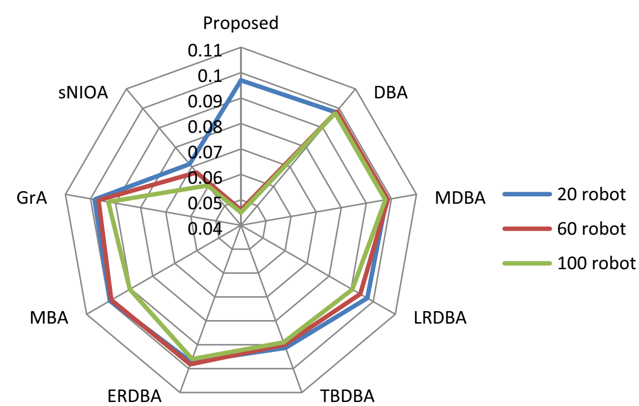

Fig. 5 Radar (spider) chart illustrates the MAE comparison between 9 methods [two targets with different quality values (0.1 and 0.9)]

It should be mentioned that in the proposed GBMRTA algorithm the initial and final selection memories are set as $\varphi i = 10$ and $\varphi f = 1000$ respectively; moreover gravitational parameter of algorithm is evaluated by power formulation as follows $G_t = (\frac{m_t}{c_t+\varepsilon})^{1.35}$.

Additional experimental setups are explained as follows:

In the DBA algorithm the control parameters α and β are same ($\alpha = \beta = 1$). As the homogeneous property of robots in this paper, $V_{ik}^{\chi} = 1$ for all robots in MDBA method and the deadline for doing a task is evaluated based on quality $\Delta_i = \frac{1}{q_i}$. In the tournament based selection strategy of TBDBA, the tournament size is defined by $K = round(T/2)$, where T is the number of targets (tasks).

[1] Infrared (IR).

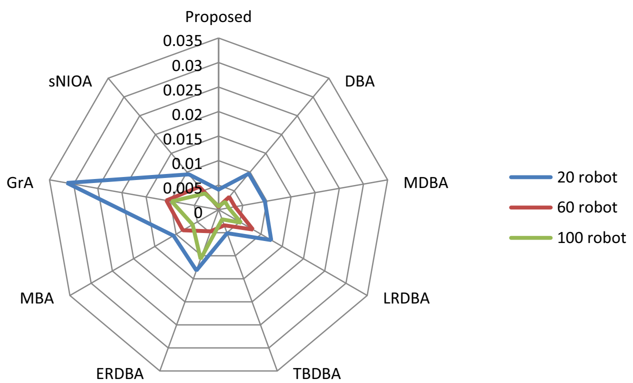

Fig. 6 Radar (spider) chart illustrates the MAE comparison between 9 methods for MRTA of four targets with same quality

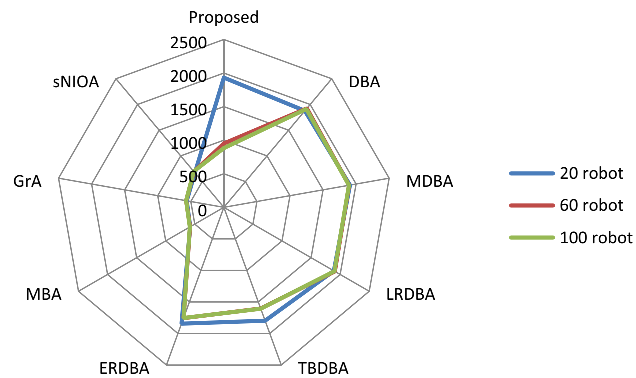

Fig. 8 Radar (spider) chart illustrates the AIT comparison the task allocation problem of two targets with different quality values

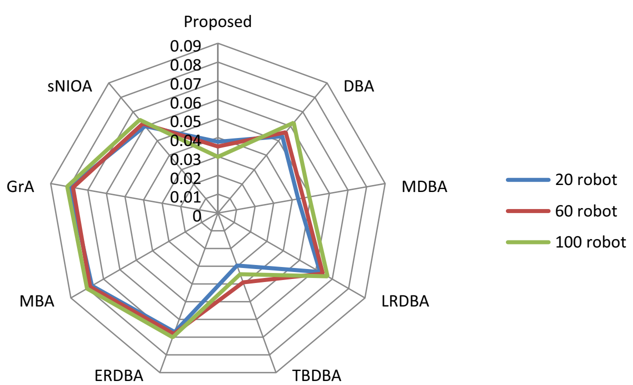

Fig. 7 Radar (spider) chart illustrates the MAE comparison between 8 method and proposed GBMRTA method for MRTA of four non-uniform quality targets

The control parameter of MBA is defined as $\delta = 0.5$ and the experimental parameters of sNIOA are set based on Huang et al. (2018).

Three different experimental setups have been chosen to compare and study the performance and scalability of the proposed algorithm. The setups were carried out in the same feasible search space where the number of robots, the number of targets and target's quality are different.

Fifty experiments were performed for each of the following swarm sizes. For all figures, the number of robots is set as 20 robots, 60 robots and 100 robots.

Figure 3 illustrates the radar chart comparison MAE of different methods on two targets with the same quality values. It should be noted that the utilized axis is designed in "reverse order" as a result of *MAE* minimization function.

We can notice that effectiveness of the algorithm increases as the size of the robot swarm increases. This was expected because of the probabilistic target allocation mechanism applied in the proposed algorithm. Moreover, in all scenarios which are plotted in Fig. 3 the proposed method has a superior performance compared to other algorithms.

In Fig. 4, the radar chart comparison MAE of different methods on two targets with different quality values (0.3 and 0.7) is illustrated.

Based on Fig. 4, the proposed method has a superior performance compared to other algorithms due to the ability of gravity rules for solving the task allocation problem and tuning parameter of G_t.

Moreover, in Fig. 5, MRTA for two unbalance targets with different quality values (0.1 and 0.9) is discussed.

Figure 5 confirms the ability of the algorithm for unbalance targets, which as a result of using the control parameter of G_t. It can be verified that by increasing the difference between the qualities of targets, the performance of the proposed method is significant in comparison with other methods.

Figure 6 illustrates the radar chart MAE comparison on four targets of same quality values. The results show that the performance of algorithm increased for larger swarms in case of four targets of the same quality.

In order to test the ability of the robot swarm to conform to a nonuniform distribution of "tasks" in the feasible search space, the experiments were performed for four targets with different quality values. To reach this aim, the qualities of 4 targets are considered 0.1, 0.2, 0.3 and 0.4. The robots' distribution changed according to a new set of targets' quality values, as shown in Fig. 7. In the same, figure we can also notice that for the DBA algorithm, the resulting robots' distribution, with respect to the expected distribution, is slightly in favor of the less valuable targets which happen as a result of non-adaptive parameters of DBA algorithm. In comparison, the proposed method has the ability to confront by non-uniform distribution of targets as a result of the control parameter of G_t.

Figure 8 illustrates the radar chart comparison of Γ which is named as Average Iteration to reach Target (AIT) (Eq. 8)

Table 2 Average rankings of the compared algorithms (Friedman)

Algorithm	Ranking
Proposed GBMRTA	1.1111
DBA	5.0556
MDBA	3.8333
LRDBA	6.1111
TBDBA	2.1111
ERDBA	7
MBA	6.8889
GrA	8.6667
sNIOA	4.2222

Table 3 The proposed method vs. other algorithms by Post Hoc comparison table for $\alpha = 0.05$ (Friedman test)

Algorithm	$z = (R_0 - R_i)/SE$	p	Li
GrA	5.852508	0	0.029549
ERDBA	4.561514	0.000005	0.029549
MBA	4.475447	0.000008	0.029549
LRDBA	3.872983	0.000108	0.02954
DBA	3.055354	0.002248	0.029549
sNIOA	2.409856	0.015959	0.029549
MDBA	2.108624	0.034977	0.029549
TBDBA	0.774597	0.438578	0.05

of different methods on two targets with different quality values (0.3 and 0.7). It should be noted that the utilized axis is designed in reverse order as a result of *AIT* minimization function.

We can notice that MBA, GrA, sNIOA, and proposed method have the ability to deliver robots to the desired targets in fewer numbers of iterations in comparison with other methods. It should be noted that based on Eqs. (15) and (21) MBA and GrA select the target based on a greedy approach which cause decreasing in the time of arrival to the target and as a drawback causes increasing in MAE value. Achieving a less *AIT* in sNIOA algorithm is a result of selection in the first iteration and not to change it during the process of algorithm which causes the increment in MAE value similar to MBA and GrA as a main drawback of the algorithm.

As mentioned before, the proposed approach in AIT comparison has a similar behavior as MBA, GrA, sNIOA which is a result of the selection memory parameter. Note that selection memory approach has a significant effect on decreasing iterations which is needed to deliver robots to the desired targets.

Experiment results that are discussed above show that the proposed method achieves a lower average MAE rate in the vast majority of experiment cases. However, this observation-based evaluation does not reflect whether or not the differences among the methods are significant. To solve this issue, the statistical test is used to make sure that the difference is significant, that is, very unlikely to be caused by chance—the so-called p-value of the test (Sheskin 2007). To evaluate the performance of the proposed method, Friedman test is applied which is a non-parametric statistical analysis based on multiple comparisons procedures. In order to perform multiple comparisons, it is necessary to check whether all results obtained by the algorithms present any inequality. Friedman test ranks the algorithms for each data set separately, the best performing algorithm obtaining the rank of 1, the second-best rank 2, and so on. In case of ties, the average ranks are assigned. Under the null-hypothesis, it is stated that all the algorithms are equivalent, so a rejection of this hypothesis implies the existence of differences among the performance of all the algorithms studied. In simple words, Friedman test can be considered as a hypothesis testing procedure.

According to statistic principles, hypothesis testing can be used to obtain inferences about one or more algorithms from the given sample. This can be achieved by defining two types of hypothesis, the null hypothesis H_0 and the alternative hypothesis H_1. The null hypothesis is a statement of no effect or no difference, whereas the alternative hypothesis represents a significant difference between algorithms. Friedman's test is a comprehensive test which can be used to carry out these types of comparison. It allows us to detect differences, considering the global set of algorithms. Table 2 contains the results of Friedman Aligned test. The average ranks obtained by each method in the Friedman test are presented in Table 2.

Note that Friedman test is applied to MAE comparison data. It should be noted that in Friedman test, achieving a lower-ranking value reflects the superiority of the algorithm. As Table 2 shows, the proposed algorithm attains the lowest ranking value among other competitors.

In addition, the comparison of the proposed method vs. other algorithms by Holm post–hoc procedure for Friedman Aligned test is described in Table 3. Table 3 shows the p-value with the Holm post-hoc test on a performance measure, which determines the validity of the corresponding ranks in Table 2.

Holm Post–Hoc procedure is a multiple comparison procedure that can work with the best algorithm (which is the best, according to Friedman rankings computation) and is compared with the remaining methods.

It should be mentioned that the proposed method is considered as the best algorithm according to Table 2.

Note that, a p-value smaller than 0.05 indicates that the null-hypothesis can be rejected. In other words, Post-Hoc comparison procedure verifies that the proposed method performs better than all other approaches, because most approaches have a p-value ≤ 0.05. Moreover, Li's procedure rejects those hypotheses that have a $p-$ value ≤ 0.029549.

This test also verifies the effectiveness of the proposed method. It is clearly shown in Table 3 that our proposed method can reject the other algorithms with most of the time significant difference.

In Friedman Aligned test, the proposed method achieved the best ranking with the obvious differences with other methods. The Friedman Aligned test emphasizes that the proposed method of this paper can outperform other compared algorithms in the defined multi-robot task allocation scenario.

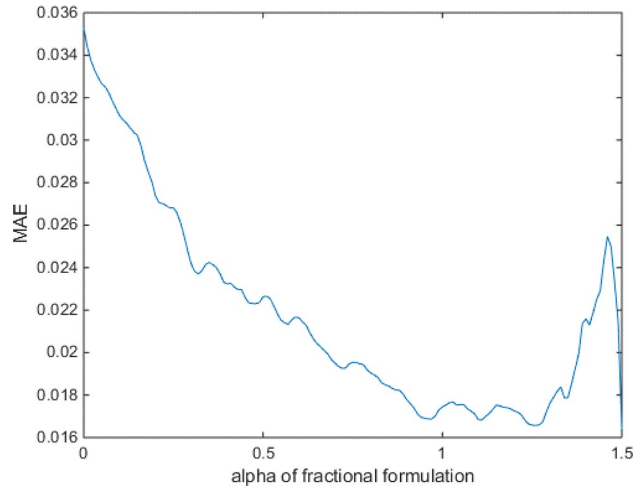

Fig. 9 MAE versus varying value of alpha for fractional formulation of gravitational parameter

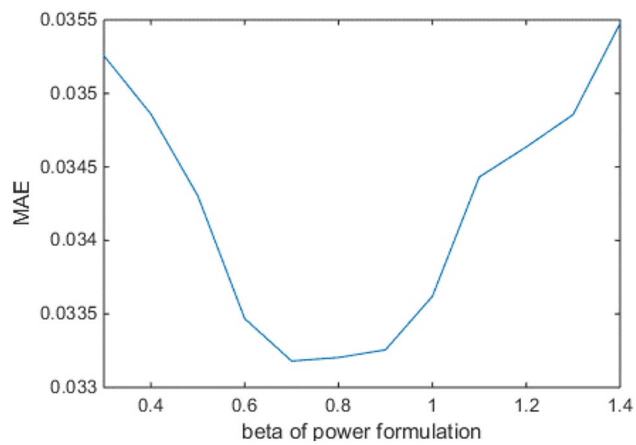

Fig. 10 MAE versus varying value of beta for power formulation of gravitational parameter

5 Conclusion

In this paper, a new nature inspired multi-robot task allocation algorithm based on Newtonian law of gravity was presented. Various applications for large MRS require efficient task allocation in terms of individual and combined robots' utilities. The quality of the solution is analyzed using a defined performance metrics, which in our case was MAE (a mean absolute error of the resulting robots' distribution with respect to the qualities of the available targets in the robot feasible search space) and AIT (Average Iteration to reach Target). In the case of large, autonomous, multi-robot systems, the scalability and the ability to adapt to different environments are the features of utmost importance. Our experiments through simulation showed that the proposed GBMRTA provides the robot swarm with scalability in terms of the number of robots and the number of targets. The proposed GBMRTA has an acceptable exploration and exploitation ability. The importance of the control parameter, G_t is that it provides a mechanism to adjust the robot swarm behavior depending on the quality of tasks. Furthermore, the proposed selection memory approach has the ability of decreasing time of reaching target and accelerating computation. It can be concluded that simulation results clearly demonstrated the high performance and efficiency of our proposed method. Moreover, this claim was confirmed by the results of Friedman test as a nonparametric comparison approach.

Appendix A

Performance evaluation of two proposed gravitational parameters

In order to find the best definition of G_t parameter two main approaches are proposed that are discussed in detail:

(A) Fractional definition

To strive for a balance between exploration and exploitation in the optimization process of the proposed method, we propose a self-adaptive mechanism which can get feedback from the current population to control the value of G_t. G_t is the control parameter that allows us to bias the decision-making mechanism toward the quality of the solution or its cost, respectively.

To evaluate adaptive, firstly, c_t as a percent of the selected target $t \in \{1, ..., T\}$ by robots is calculated as follows:

$$c_t = \frac{n_t}{R} \quad t \in \{1, ..., T\} \tag{10}$$

Table 4 Comparison of two proposed gravitational parameters

Num. of robots	Fractional formulation		Power formulation	
	Average	Maximum	Maximum	Maximum
Exp. setup 1				
10	**0.1000**	0.2000	0.1550	0.4000
20	0.0775	0.2000	**0.0750**	0.2000
40	0.0663	0.1250	**0.0650**	0.1750
60	**0.0400**	0.1500	0.0542	0.1333
100	**0.0305**	0.0900	0.0435	0.1200
Exp. setup 2				
10	**0.0855**	0.1750	0.1085	0.1750
20	**0.0560**	0.1000	0.0705	0.1500
40	**0.0498**	0.1125	0.0610	0.1125
60	**0.0393**	0.0833	0.0443	0.0917
100	**0.0308**	0.0650	0.0380	0.0650

Bold values indicate best results

where n_t present the number of robots which select the target t and R is the total number of robots.

It should be explained that based on MRTA problems, MAE is one of the most crucial criterion. In order to decrease the MAE value, the difference between c_t(percent of selected target or the resulting robots' distribution) and m_t (expected percent of selected target or expected robots' distribution) should be decreased.

To reach this aim in our proposed algorithm, the gravitational parameter G_t is considered as an adaptive controlling parameter which is evaluated as follows:

$$G_t = \left(\frac{m_t}{c_t + \varepsilon}\right)^{\alpha} \tag{11}$$

where ε is a small positive number to avoid dividing by zero and α is a parameter of the algorithm which tunes the effect of G_t on the calculation of the force.

(B) Exponential definition

$$G_t = \beta^{c_t - m_t} \tag{12}$$

In the following, the performance of two proposed gravitational parameters are evaluated. To reach this aim, Figs. 2 and 3 illustrate the MAE versus variable value of alpha for gravitational constant 1 and 2 when the number of robot is 100.

It can be concluded from Figs. 9 and 10 that the best values of α and β occur in 1.35 and 0.74, respectively. Moreover, the MAE value of fractional formulation is less than power formulation. To have a more precise investigation, the average and the maximum values of MAE obtained for setup 1 (number of targets is two with similar qualities) and setup 2 (four targets with same qualities) from the experiments are presented in Table. Based on Table 4, the performance of G_t based on fractional formulation is better than power formulation.

As a result power formulation $G_t = \left(\frac{m_t}{c_t+\varepsilon}\right)^{1.35}$ is considered as the gravitational parameter of algorithm.

Appendix B

Discussion of algorithms for comparison

Distributed bees algorithm (DBA)

In Jevtic et al. (2011), the task allocation problem is solved with a swarm-based meta-heuristic technique named DBA algorithm which is inspired by the intelligent behavior of the "bees" to optimize their search for "food" resources. In this algorithm, each robot is represented as a ‘bee’, and task utility, p_{ik}, is defined as a probability that the task k is allocated to the task i and depends on both target’s quality and the distance of the task from the robot:

$$p_{ik} = \frac{q_i^{\alpha}\left(\frac{1}{D_{ik}}\right)^{\beta}}{\sum_{j=1}^{M} q_j^{\alpha}\left(\frac{1}{D_{jk}}\right)^{\beta}} \tag{13}$$

where q_i and D_{ik} shows the quality or priority of target i and distance between target i and robot k, respectively. Moreover, α and β are control parameters that bias importance of the priority and distance, respectively, ($\alpha, \beta > 0; \alpha, \beta \in R$).

In DBA algorithm, roulette wheel selection method is chosen in which a target with a probabilistic procedure is selected by each robot. In other words, in DBA the selection probability of target is proportional to their fitness value.

The probabilities p_{ik} are normalized, and it is easy to show that:

$$\sum_{i=1}^{M} p_{ik} = 1 \tag{14}$$

Modified distributed bees algorithm (MDBA)

In the MDBA, the original DBA robot utility function is modified to take advantage of heterogeneous robots or targets with different performances aiming to improve system performance by correlating the robot’s utility with their performances. In order to apply MDBA to heterogeneous robots or targets, a control parameter was defined in Tkach et al. (2018) as a function of the robot’s performance on a target. In simple words, when a robot receives information about an available target, it calculates its performance for that task. The robot’s utility function is updated accordingly, and depends on the target quality, the distance from the task and the robot’s performance on that task:

$$p_{ik} = \frac{q_i^{\alpha}\left(\frac{1}{D_{ik}}\right)^{\beta} V_{ik}^{\chi}}{\sum_{j=1}^{M} q_j^{\alpha}\left(\frac{1}{D_{jk}}\right)^{\beta} V_{jk}^{\chi}} \quad \mathit{if}\ \Delta_i > 0 \tag{15}$$

where χ is a control parameter that biases the importance of the robots performance and V_{ik} is the performance of robot k on task i. Moreover, each task has a time limit, or a deadline which is evaluated as follows:

$$\Delta_i = \frac{1}{q_i} > 0, \quad q_i > 0 \tag{16}$$

The MDBA decision-making mechanism applies the same wheel-selection rule that is used in DBA to choose from a set of available tasks.

Linear ranking for distributed bees algorithm (LRDBA)

LRDBA shows a linear ranking selection for DBA. In order to do linear ranking selection, each target is defined by its fitness score which is named as "*rank of target*". In other words, the selection probability of each target ($Prob_i$) in linear ranking selection is evaluated as follows:

$$Prob_i = q - \left(q - q_0\right) \times \frac{R_i - 1}{T - 1} \tag{17}$$

where q and q_0 are probability of selection of the best target and the worst target. Moreover R_i is the rank of target i and T shows the total number of targets.

Exponential ranking for distributed bees algorithm (ERDBA)

In ERDBA an exponential ranking selection for DBA is utilized. This technique is different from linear ranking selection technique in a way that the probabilities of ERDBA are exponentially weighted as follows:

$$Prob_i = q(1 - q)^{R_i - 1} \tag{18}$$

Tournament based for distributed bees algorithm (TBDBA)

TBDBA is a tournament based selection strategy for DBA. Tournament Selection is a selection procedure used for selecting the fittest candidates from the current generation. These selected candidates are then passed on to the next generation. In a K-way tournament selection, k-individuals are selected and run a tournament among them. Only the fittest candidate among those selected candidates is chosen and is passed on to the next generation. In this way many such tournaments take place and the final selection of candidates who move on to the next generation are given.

Market-based algorithm (MBA)

A market-based algorithm used in Zlot et al. (2002) for distributed system was applied with application specific modifications. In this approach, the bid of robot k to task i is defined as (19):

$$Bid_{ik} = q_{ik} + \delta\left(\frac{V_{ik}}{D_{ik}} - q_{ik}\right) \tag{19}$$

where q_{ik} serves as the reservation price of task i, and δ is a control parameter with values between 0 and 1. A task i is selected by robot k if it maximizes its bid value:

$$select = \max(Bid_{ik}) \tag{20}$$

Greedy algorithm (GrA)

A greedy algorithm that was used previously for a multi target observation problem with broadcast messaging (Broadcast of Local Eligibility for Multi-Target Observation 2002) was modified to fit the described problem. The greedy algorithm was set to perform task allocation based on the best possible allocation of each individual robot to task that maximizes $\frac{V_{ik}}{D_{ik}}$, where V_{ik} is the k-th robot's performance on the i-th task and D_{ik} is the Euclidean distance between the robot and the task:

$$task_k = \max\left(\frac{V_{ik}}{D_{ik}}\right) \quad i \in Z \tag{21}$$

where $task_k$ is the task chosen by the k-th robot, i shows the index of task, and Z is the set of tasks within k-th robot range. Note that Z is a subset of all M available tasks.

Niching immune-based optimization algorithm based on softmax regression (sNIOA)

The sNIOA presents a niching immune-based optimization algorithm based on Softmax regression (sNIOA) to handle it (Huang et al. 2018). A pre-judgment of population is done before entering an evaluation process to reduce the evaluation time and to avoid unnecessary computation. Furthermore, a guiding mutation operator inspired by the base pair in theory of gene mutation is introduced into sNIOA to strengthen its search ability. It should be mentioned that in Huang et al. (2018), a discrete version of immune optimization algorithm is used in which each antibodies is utilized to present the allocated robot to defined tasks.

Acknowledgements This paper has been financially fully supported by the deputy of research and technology of Higher Education Complex of Bam. The Grant number was 3998112133.

References

Atay, N., Bayazit, B.: Mixed-integer linear programming solution to multi-robot task allocation problem, Washington University of St. Louis, Technical Report (2006)

Balas, E., Padberg, M.W.: Set partitioning: a survey. SIAM Rev. **18**(4), 710–760 (1976)

Benabderrahmane, S.: Combining boosting machine learning and swarm intelligence for real time object detection and tracking: towards new meta-heuristics boosting classifiers. Int. J. Intell. Robot. Appl. **1**(4), 410–428 (2017)

Darrah, M., Niland, W., Stolarik, B.M.: Multiple UAV dynamic task allocation using mixed integer linear programming in a sead mission. In: American Institute of Aeronautics and Astronautics, pp. 2324–2334 (2005)

Ding, Y., He, Y., Jiang, J.: Multi-robot cooperation method based on the ant algorithm. In: IEEE Swarm Intelligence Symposium, pp. 14–18 (2003)

Gerkey, B.P., Matarić, M.J.: A formal analysis and taxonomy of task allocation in multi-robot systems. Int. J. Robot. Res. **23**(9), 939–954 (2004)

Guerrero, J., Oliver, G.: Multi-robot task allocation strategies using auction-like mechanisms. Artif Intell. Res. Dev. Front. Artif. Intell. Appl. **100**, 111–122 (2003)

Huang, L., Ding, Y., Zhou, M., Jin, Y., Hao, K.: Multiple-solution optimization strategy for multirobot task allocation. IEEE Trans. Syst. Man Cybern. Syst. **50**(11), 4283–4294 (2018)

Jahanshahi, M.R., Shen, W.M., Mondal, T.G., Abdelbarr, M., Masri, S.F., Qidwai, U.A.: Reconfigurable swarm robots for structural health monitoring: a brief review. Int. J. Intell. Robot. Appl. **1**(3), 287–305 (2017)

Jang, I., Shin, H.S., Tsourdos, A.: Anonymous hedonic game for task allocation in a large-scale multiple agent system. IEEE Trans. Robot. **34**(6), 1534–1548 (2018)

Jevtic, A., Gutierrez, A., Andina, D., Jamshidi, M.: Distributed bees algorithm for task allocation in swarm of robots. IEEE Syst. J. **6**(2), 296–304 (2011)

Jones, E.G., Dias, M., Stentz, A.: Time-extended multi-robot coordination for domains with intra-path constraints. Auton. Robots **30**(1), 41–56 (2011)

Juedes, D., Drews, F., Welch, L., Fleeman, D.: Heuristic resource allocation algorithms for maximizing allowable workload in dynamic, distributed real-time systems. In: Parallel and Distributed Processing Symposium, pp. 1631–1638 (2004)

Kanakia, A., Touri, B., Correll, N.: Modeling multi-robot task allocation with limited information as global game. Swarm Intell. **10**(2), 147–160 (2016)

Khamis, A., Hussein, A., Elmogy, A.: Multi-robot task allocation: a review of the state-of-the-art. In: Cooperative Robots and Sensor Networks, pp. 31–51 (2015)

Kmiecik, W., Wojcikowski, M., Koszalka, L., Kasprzak, A.: Task allocation in mesh connected processors with local search metaheuristic algorithms. In: Intelligent Information and Database Systems, Springer, pp. 215–224 (2010)

Koes, M., Nourbakhsh, I., Sycara, K.: Constraint optimization coordination architecture for search and rescue robotics. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 3977–3982 (2006).

Korsah, G.A., Stentz, A., Dias, M.B.: A comprehensive taxonomy for multi-robot task allocation. Int. J. Robot. Res. **32**(12), 1495–1512 (2013)

Lagoudakis, M.G., Markakis, E., et al: Auction-based multi-robot routing. In: Proceedings of Robotics: Science and Systems, Cambridge, USA (2006).

Lee, D.H.: Resource-based task allocation for multi-robot systems. Robot. Auton. Syst. **103**, 151–161 (2018)

Lenagh, W.H., Multi-robot task allocation: a spatial queuing approach, Ph.D. dissertation, University of Nebraska, Omaha (2013)

Lerman, K., Galstyan, A.: Mathematical model of foraging in a group of robots: effect of interference. Auton. Robots **13**(2), 127–141 (2002)

Liu, D.K., Kulatunga, A. K.: Simultaneous planning and scheduling for multi-autonomous vehicles. In: Evolutionary Scheduling. Springer, pp. 437–464 (2007)

Mosteo, A.R.: Multi-robot task allocation for service robotics: from unlimited to limited communication range. Ph.D. Thesis, Universidad de Zaragoza (2010)

Mosteo, A.R., Montano, L.: Simulated annealing for multi-robot hierarchical task allocation with flexible constraints and objective functions. In: Workshop on Network Robot Systems Toward Intelligent Robotic Systems Integrated with Environments (2006)

Nunes, E., Manner, M., Mitiche, H., Gini, M.: A taxonomy for task allocation problems with temporal and ordering constraints. Robot. Auton. Syst. **90**, 55–70 (2017)

Parker, L.E., Tang, F.: Building multirobot coalitions through automated task solution synthesis. Proc. IEEE **94**(7), 1289–1305 (2006)

Rashedi, E., Nezamabadi-pour, H., Saryazdi, S.: GSA: a gravitational search algorithm. Inf. Sci. **179**(13), 2232–2248 (2009)

Ren, L., et al.: An optimal task allocation approach for large-scale multiple robotic systems with hierarchical framework and resource constraints. IEEE Syst. J. **12**(4), 3877–3880 (2017)

Schwarzrock, J., et al.: Solving task allocation problem in multi unmanned aerial vehicles systems using swarm intelligence. Eng. Appl. Artif. Intell. **72**, 10–20 (2018)

Shea, P.J., Alexander, K., Peterson, J.: Group tracking using genetic algorithms. In: Proceedings of the International Society Information Fusion (2003)

Sheskin, D.: Handbook of parametric and nonparametric statistical procedures, 4th edn. Chapman and Hall/CRC, London (2007)

Tang, F., Parker, L.E.: A complete methodology for generating multi-robot task solutions using asymtred and market-based task allocation. In: IEEE International Conference on Robotics and Automation, pp. 3351–3358 (2007).

Tkach, I., Jevtić, A., Nof, S., Edan, Y.: A modified distributed bee's algorithm for multi-sensor task allocation. Sensors **18**(3), 759 (2018)

Wang, D., Wang, H., Liu, L.: Unknown environment exploration of multi-robot system with the FORDPSO. Swarm Evol. Comput. **26**, 157–174 (2016)

Werger, B., Mataric, M.J.: Broadcast of local eligibility for multi-target observation. Distrib. Auton. Robot. Syst. **4**, 347–356 (2002)

Yi, X., et al.: A bio-inspired approach to task assignment of swarm robots in 3-D dynamic environments. IEEE Trans. Cybern. **47**(4), 974–983 (2017)

Zlot, R., Stentz, A.: Market-based multi-robot coordination for complex tasks. Int. J. Robot. Res. **25**(1), 73–101 (2006)

Zlot, R., Stentz, A., Dias, M.B., Thayer, S.: Multi-robot exploration controlled by a market economy. In: Proceedings of the IEEE International Conference on Robotics and Automation, Washington, DC, USA (2002)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Mohadese Soleimanpour-moghadam was born in 1986, Iran. She received her B.S., M.S. and Ph.D. degrees in electrical engineering from Shahid Bahonar University of Kerman (SBUK), Kerman, Iran, in 2008, 2011 and 2016, respectively. In 2017, she joined the Department of Mechanical Engineering at Higher Education Complex of Bam, as an Assistant Professor. Her research interests include Optimization, wireless communications, soft computing and robotics.

Hossein Nezamabadi-pour is Professor of Elec Eng (EE) at SBUK. He received his M.Sc. and Ph.D. degrees in EE from the Tarbait Moderes University, Tehran, Iran in 2000 and 2004. His research interests include image processing, pattern recognition and evolutionary computation. Dr Nezamabadi-pour is the coauthor of more than 400 peer reviewed journal and conf. papers. He is a recipient of the outstanding researcher award by SBUK in 2009, 2012, 2013, 2018, 2020 and earned provincial distinguished research awards in 2010 and 2016.