Ad Hoc Networks 87 (2019) 26–36

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Use of a quantum genetic algorithm for coalition formation in large-scale UAV networks

Sajad Mousavi[a,*], Fatemeh Afghah[a], Jonathan D. Ashdown[b], Kurt Turck[b]

[a] School of Informatics, Computing and Cyber Systems, Northern Arizona University, Flagstaff, AZ, United States
[b] Air Force Research Laboratory, Rome, NY, United States

ARTICLE INFO

Article history:
Received 4 June 2018
Revised 8 October 2018
Accepted 5 November 2018
Available online 5 December 2018

Keywords:
Unmanned aerial vehicles
Coalition formation
Task allocation
Evolutionary algorithms
Quantum genetic algorithm

ABSTRACT

Task allocation among a network of heterogeneous resource-constrained Unmanned Aerial Vehicles (UAVs) in an unknown and remote environment is still a challenging problem noting the limited available information about highly dynamic environment, lack of continuous and reliable communication network, and the limited energy and resources available at the UAVs. One solution for this such allocation problem is to form several efficient coalitions of the UAVs, where a complex task is assigned to a group of agents (i.e., a coalition) carrying the required resources/capabilities to perform this task. In this paper, inspired by Quantum Evolutionary Algorithms, we propose a leader-follower coalition formation algorithm in a large-scale UAV network to form the best possible coalitions of agents to accomplish the detected tasks in an unknown environment. Three main objectives have been considered in this coalition formation: (i) minimizing resource consumption in completing the assigned tasks on time; (ii) enhancing the reliability of the coalitions; and (iii) considering the most trustworthy UAVs amid the self-interested UAVs in forming the coalitions. The simulation results demonstrate the superior performances of the proposed model in different scenarios with large number of UAVs compared to existing coalition formation algorithms such as merge-and-split and a famous multi-objective genetic algorithm called NSGA-II.[1,2]

1. Introduction

A single agent system is often unable to perform complex tasks, considering the limited individual capabilities of such an agent. Therefore, cooperative multi-agent systems (MASs) can offer a practical solution to such problems by ensembling a complementary set of different capabilities/resources from several agents. However, one of the key challenges in such cooperative MASs is forming optimal sub-groups of agents (i.e., coalition formation) in order to efficiently perform the existing tasks, especially in a distributed case where no central controller is available. That being said, the coalition formation problem concerns how different coalitions can be formed considering the tasks' requirements and the agents' capabilities, so much so that the collective goals of the tasks are reached in the most effective manner. In the case of multi-UAV task allocation problem, a task can be done by a group of UAVs, resulting in a better performance.

A considerable amount of research has been recently carried out in solving coalition formation problems. This has spawned several classic methods to form stable coalitions that follow common stability concepts based on Core, Shapley value, Bargaining Set, and Kernel [1–3]. However, achieving such stability concepts often mandates high computational complexity. Coalition formation problem can also be considered as a class of games in game theory, where the agents cooperate with each other to form different teams with maximum payoffs [4–6]. In addition, many researchers have attempted to deal with the problem of coalition formation in multi-agent systems by applying various approaches including genetic algorithms [7], dynamic programming methods [8], graph theory [9,10], iterative processes [11], cooperative Multi-Agent Reinforcement Learning (MARL) [12–15], and temporal-spatial abstraction MARL [16–20].

In the majority of the previously reported works, a central unit with information on mission status and the agents' capabilities is required [21,22]. Such centralized methods are highly vulnerable to the failure of the central controller unit and also cannot easily scale up to large-scale networks. On the other hand, the decentralized task allocation approaches for heterogeneous UAV

* Corresponding author.
E-mail addresses: sajadmousavi@nau.edu (S. Mousavi), fatemeh.afghah@nau.edu (F. Afghah), jonathan.ashdown@us.af.mil (J.D. Ashdown), kurt.turck@us.af.mil (K. Turck).

[1] DISTRIBUTION A. Approved for public release: distribution unlimited. Case Number: 88ABW-2018-0096. Dated 10 Jan 2018.

[2] An earlier version of this work was presented at 2018 INFOCOM Workshop on WISARN: Wireless Sensor, Robot and UAV Networks, Honolulu, HI.

https://doi.org/10.1016/j.adhoc.2018.11.008
1570-8705/

networks involve a high level of message passing among the agents to converge to a solution [23–26]. In this paper, we study the problem of task allocation among a large-scale heterogeneous resource-constraint UAV network in an unknown and dynamic environment where no prior knowledge is available about targets or Point of Interests (PoIs) and therefore UAVs need to gain the knowledge of environment dynamically [27,28]. We propose a leader-follower coalition formation model for task allocation, in which the UAVs divide the tasks among themselves when there is no central controller in the network with the knowledge of targets' locations and required resources as well as knowledge of the UAVs' location and available resources. In this model, the UAVs are first in the search state, and then each UAV that detects a target can act as a leader to form a coalition to complete the tasks associated to this target. To the best of our knowledge, there is no previous work to provide an on-flight task allocation mechanism amongst a large number of heterogeneous UAVs when the location and type of targets are not known prior to flight. The proposed leader-follower approach is a decentralized and localized algorithm meaning that each leader devotes itself to manage its detected target by recruiting a group of UAVs in a certain distance from the target that carry a subset of required resources to complete the encountered task. Therefore, the proposed method only involves massage passing among the UAVs in a close proximity of the detected target.

Furthermore, the majority of the previously reported studies have mainly focused on a single factor in forming the optimal coalitions (i.e., reducing the costs of coalitions) and only a few of them have focused upon multiple objectives. Therefore, in this paper, we propose a multiple objective optimization to account for the following criteria in forming the optimal coalitions: (i) minimizing the cost associated with consumption of resources of the coalitions formed; (ii) maximizing the reliability of the formed coalitions; and (iii) to select the most trustworthy of UAVs among the available self-interested UAVs in the network. Reliability is a critical factor in coalition formation to perform a task successfully where all UAVs need to work perfectly during the mission. Hence, it is desired to maximize the coalition reliability by involving the most reliable UAVs in the coalition. In other words, as the leaders intend to carry out their assigned tasks with the least probability of failure in the mission, they prefer to include the UAVs with the maximum operational probabilities during the mission. Therefore, the UAVs with lower reliability have a lower chance to be selected in the mission. To the best of our knowledge, this paper is the first work that incorporates the reliability as an objective in multi-UAV coalition formation problem.

Finding the solution of such a multi-objective coalition formation problem involves an NP-hard problem. Many approaches such as mixed integer linear programming [29,30] and dynamic network flow optimization [31] have been utilized to provide an exact solution to this problem. However, since these approaches seek such a solution, applying them to a large-scale problem is computationally taxing. More recently, metaheuristic algorithms such as Particle Swarm Optimization (PSO) [32], Ant Colony Optimization [33], Genetic Algorithms (GAs) [34], and Simulated Annealing (SA) [35] have offered reasonable solutions in efficient times for a variation of multi-objective optimization problems. Inspired by the success of evolutionary approaches, this paper presents a novel leader-follower-based coalition formation algorithm using a quantum evolutionary approach whilst considering the aforementioned objectives in addressing the problem.

Potential applications of the proposed method are search and rescue, humanitarian relief, and public safety operations in unknown remote environments or military operations in remote fields. In such environments, it can be safely assumed that ground station does not have prior information about the PoIs and their positions. To evaluate the performance of the proposed method in such an unknown environment, several scenarios with different numbers of tasks ranging from 4 to 24, and a heterogeneous network of UAVs consists of a different number of UAVs ranging from 8 to 124 were simulated.

The rest of the paper is organized as follows: Section 2 presents the problem statement and the formulation of the multi-UAV coalition formation as a multi-objective optimization problem. Section 3 describes the proposed coalition formation algorithm. Section 4 reports the simulation results followed by concluding remarks in Section 5.

2. Problem definition

In this section, the decentralized task allocation problem in a network of autonomous UAVs by forming optimum coalitions is formulated. The possibility of selfish behavior of these self-interested UAVs are accounted for and reputation guidelines in selecting the most reliable of UAVs to participate in the formed coalitions are also defined. It is vital to note that the proposed algorithm considers minimizing the cost of coalition formation in terms of overspending the resources on particular tasks as well as enhancing the reliability of the formed coalitions as paramount.

A heterogeneous network of N UAVs $\mathcal{U} = \{u_1, u_2, \dots, u_N\}$, where each UAV, u_i can potentially carry a different set of resources compared to other ones is considered. $R_{u_i} = \{r^1_{u_i}, r^2_{u_i}, \dots, r^{N_r}_{u_i}\}$ denotes the set of resources available at UAV, u_i where N_r is the number of possible resources in the network. It is also assumed that there exists n tasks in the environment $T = \{T_1, T_2, \dots, T_n\}$. Each task, T_i requires a certain amount of resources to be completed. The vector of required resources for task i is defined as follows:

$$\tau^i = \{\tau^i_1, \tau^i_2, \tau^i_3, \dots, \tau^i_{N_r}\}, \tag{1}$$

where τ^i_j is the required amount of resource j for task i. It is assumed that each task is associated to one PoI (target) and the PoIs can be located in different positions with a diverse set of resource requirements. All UAVs are able to search the unknown environment for new PoIs. The tasks are carried out by the formed coalitions $\mathcal{S} = \{S_1, S_2, \dots, S_m\}$, where each coalition S_i is responsible for one task. A large search space where the PoIs are distributed far apart from each other are considered. It can therefore be assumed that the formed coalitions are sufficiently far from one another that each UAV can only be a member of a single coalition, i.e., $S_k \cap S_l = \emptyset, \forall S_k, S_l \in \mathcal{S}$. This also means that the coalitions are non-overlapping. The capability of each coalition S_i to complete its encountered task is defined as the value of coalition $v(S_i)$, $(v(S_i) \in \mathcal{R})$ as described in the next section. Moreover, $cost(S_i, T_i)$ is defined as the cost of coalition S_i in performing task T_i. The cost function for coalition S_i captures the cost imposed on all UAV members of this coalition (i.e., $\sum_{u_j \in S_i} cost(S_i, u_j)$) in which their resources have been shared to accomplish task T_i.

Another key contribution of the proposed model is considering the reliability factor in forming the optimal coalitions. While in the majority of existing techniques, it is assumed that the UAV members of formed coalitions are perfectly operational during the mission lifetime, this is obviously not a realistic assumption as the UAVs' operation can be interrupted for several reasons (e.g., exhaustion of battery or a particular resource). A practical case is considered where the UAVs are assumed to be either fully operational or one of their capabilities (e.g., resources) are bound to fail during the mission. Such failures in various capabilities are considered to be statistically independent. Furthermore, involving different types of UAVs in a coalition may result in different execution times of accomplishing the sub-tasks as the UAVs have a different set of capabilities. For instance, a given UAV may be able to fulfill its duty in a shorter time than another. The cost and reliability of a coalition indeed depend on these execution times where lower

Table 1
A simple example of failure rate and execution time.

	$UAV_{i=1}$	$UAV_{i=2}$
Failure rate(λ_{ij})	0.00005	0.0004
Execution time(k_{ij})	10	20

execution times are favored for incurring lower execution costs. In the next section, we define and formulate these factors (i.e., cost, reliability and reputation).

2.1. Definition of cost and reliability of the formed coalitions

A set of UAVs in the form of a coalition collaborate with one another to carry out the encountered task. Participation in such coalitions involves a cost of sharing and consuming the resources for the member UAVs. For a given task T_k with the required resources $\tau^k = \{\tau_1^k, \tau_2^k, \tau_3^k, \ldots, \tau_{N_r}^k\}$, where the amount of resource i for task k is denoted by $\tau_i^k \geq 0$, the execution cost of consuming resource j of UAV i is denoted by e_{ij}, where $e_{ij} = \mu_j r_{u_i}^j \quad \forall i, j$ and μ_j is a constant coefficient in order to convert the amount of resource j to a time dependent value to have the same unit as the execution time. Thus, the cost of coalition S_k, $C(S_k)$ can be calculated as follows:

$$C(S_k) = \sum_{i=1}^{N_k} \sum_{j=1}^{N_r} e_{ij} k_{ij} + a_{ij}, \tag{2}$$

where k_{ij} is the execution time if UAV i carrying resource j is involved in task k, and a_{ij} is the travel time of UAV i to task j.

Possibility of potential defects in the UAVs' resources that may result in performance failure of these UAVs during the mission is also accounted for. Considering so, the reliability of coalition S_k for a give task T_k, denoted by $R(S_k)$ is defined as follows:

$$R(S_k) = \prod_{i=1}^{N_k} e^{-\sum_{j=1}^{N_r} \lambda_{ij} k_{ij}}, \tag{3}$$

where λ_{ij} is the failure rate of resource j of UAV i. For simplicity, $\log_e$ transfer of $R(S_k)$ function as follows, $\ln(R(S_k)) = -\sum_{i=1}^{N_k} \sum_{j=1}^{N_r} \lambda_{ij} k_{ij}$ is used. For instance, suppose there are $N_k = 2$ UAVs (e.g, each owns just one resource) to accomplish task k. Table 1 presents the failure rates and execution times of involved UAVs in the task k. Considering the Eq. (3), the reliability of coalition S_k for task k is computed as follow:

$$R(S_k) = e^{-10\times 0.00005} \times e^{-20\times 0.0004}$$
$$\ln(R(S_k)) = -10 \times 0.00005 - 20 \times 0.0004 = -0.0085$$

The formulation of the reliability has been inspired by works reported in [36,37]. Interested readers are referred to these papers for more details on the probability that a system can accomplish a particular task without failure.

2.2. Definition of reputation factor

Similar to other cognitive agents, the UAVs are expected to be self-interested in the sense that they prefer to save their limited resources, and may act selfishly by not consuming enough resources during the mission. To monitor the cooperative behavior of these UAVs, an accumulative cooperative reputation related to the amount of resources that each UAV shares during the mission is defined [38–41].

In common reputation-based mechanisms, there exists a central or several audit units in the network with the role of monitoring all the agents' behavior over the course of time to measure their level of trust and then broadcast this information or share it with other agents on an demand basis [42,43]. However, such approach is not appropriate for distributed systems. On the other hand, if the agents are allowed to self-report their reputations as implemented in many cooperative multi-agent systems, the system would be at high risk of dealing with several false reputation reports by the malicious or selfish agents [44–46]. To avoid such false reputation reports, here we propose a reputation monitoring mechanism in which the leader UAVs measure the contributions of the follower UAVs in terms of the amount of resources they invested in that specific task after the completion of coalition's mission and they share that with other leaders when new coalitions are being formed.

We define two kind of reputations in this study. The first-hand reputation, which is calculated based of the leader knowledge of the follower, and the second-hand reputation, which is computed based the knowledge of other leaders in the neighborhood. The second-hand reputation is used when the reputation history of a follower is not available to a specific leader. During each mission n (i.e., accomplishing an assigned task), if the leader has a history of direct interaction with the follower (here, UAV i), the first-hand cooperative reputation of each it, ρ_i is updated as follows:

$$\rho_i^n = \begin{cases} \rho_i^{n-1} + \Delta\rho_i^n, & \exists k | u_i \in S_k \\ \rho_i^{n-1}, & \text{otherwise} \end{cases} \tag{4}$$

where n is a discrete factor, representing the index of a mission so that H_n can be the elapsed time from the beginning of the first mission till now. Also, $\Delta\rho_i^n$ is the amount of contribution of UAV i to coalition S_k in terms of sharing resources to carry out the assigned task k and can be defined as follows:

$$\Delta\rho_i^n = \frac{\Upsilon_k}{\sum_{m\in S_k} f_m^n} f_i^n, \tag{5}$$

where Υ_k is the sum of the resource requirements of task T_k denoting as $\Upsilon_k = \sum_{j=1}^{N_r} \tau_j^k$ and f_i^n is the sum of the resource contributions of UAV i after mission is completed, defined as $f_i^n = \sum_{j=1}^{N_r} \frac{r_{u_i}^{jn}}{\tau_j^k}$, where $r_{u_i}^{jn}$ is the amount of resource j in the time step n. In other words, the leaders observe the amount of resources that UAV i contributed to the mission when the mission is completed (i.e., r_{u_i}) and then update the corresponding reputation of the followers after finishing each mission. Here, the coalition reputation of all involved UAVs in the coalition S_k is computed as follows:

$$P(S_k) = \sum_{i=1}^{N_k} \rho_i^n \tag{6}$$

If there does not exist a prior record of interactions between a leader and a follower, then the leader can inquire the follower's reputation from its neighbor (i.e., adjacent leaders). In this cases, the second-hand reputation is defined as:

$$\rho_i^n = \frac{\sum_{l\in L} \rho_{il}^n}{|L|}, \tag{7}$$

where L is the set of leaders in proximity of the leader assigned to the task k.

2.3. Formulation of multi-objective optimization problem

To consider all aforementioned optimization criteria including reducing the coalition cost, and increasing the reliability, and reputation of the formed coalitions, a Multi-Objective Optimization Problem (MOOP) as a weighted-sum of three objectives is defined. The multi-objective optimization and its required constraints are defined as follows.

$$min \qquad O(S_k) = C(S_k) - \eta_1 \ln R(S_k) - \eta_2 P(S_k) \tag{8}$$

Table 2
Notations.

a_{ij}	Travel time of UAV i to task j.
e_{ij}	Execution cost of involving resource j of UAV i in a coalition. It is computed per unit time.
k_{ij}	Execution time of involving resource j of UAV i in a coalition. It depends on the task and capability of the UAV.
λ_{ij}	Failure rate of involving resource j of UAV i in a coalition.
ρ_i	Credit of UAV i.
N_k	Number of UAVs in coalition k.
N_r	Number of network resources.
N	Number of network UAVs.

$$subject \quad to \qquad \sum_{i=1}^{N_k} e_{ij} \geq \tau_j^k \quad \forall j = 1, 2, 3, \ldots, N_r, \tag{9}$$

where η_1 and η_2 are weighting parameters to assign the desired importance to each objective and scale them to be in comparative ranges. Constraint (9) refers to the requirement to secure enough resources in the formed coalition to complete the encountered task T_k. In Table 2, a summary of the notations used throughout this paper is presented.

3. Proposed method

The objective function described in Section 2.3 is a NP-hard problem. Standard approaches such as dynamic programming and exact algorithms involve computational complexity of $O(n^2)$ that could be intractable in large-scale networks. Hence, evolutionary algorithms such as genetic algorithm can be considered as potential options to find feasible solutions of this problem. In this paper, we propose a coalition formation algorithm based on a version of genetic algorithm called Quantum-Inspired Genetic Algorithm (QIGA) to find the solution of the multi-objective problem formulated in (8).

3.1. Review of quantum-Inspired genetic algorithm (QIGA)

The idea behind QIG algorithms is to take advantage of both GA and quantum computing mechanisms [36]. In quantum computation, the data representation is based on qubit that is the smallest information unit. A qubit is considered as a superposition of two different states $|0\rangle$ and $|1\rangle$ that can be denoted as:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{10}$$

where α and β are complex numbers such that $|\alpha|^2 + |\beta|^2 = 1$. $|\alpha|^2$ and $|\beta|^2$ are the probability of amplitudes where the qubit can be at states $|0\rangle$ and $|1\rangle$, respectively. With m qubits, the model can represent 2^m independent states. However, when the value of the qubit is measured, it leads to a single state of the quantum state (i.e., $|0\rangle$ or $|1\rangle$). Similar to a standard genetic algorithm, a chromosome's representation is defined as a string of m information units. Thus, a chromosome can be defined as a string of m qubits as:

$$\left\{ \begin{pmatrix} \alpha_0 \\ \beta_0 \end{pmatrix}, \begin{pmatrix} \alpha_1 \\ \beta_1 \end{pmatrix}, \begin{pmatrix} \alpha_2 \\ \beta_2 \end{pmatrix}, \ldots, \begin{pmatrix} \alpha_{m-1} \\ \beta_{m-1} \end{pmatrix} \right\}, \tag{11}$$

where each pair $(\alpha_i, \beta_i), i = 1, 2, 3, \ldots, m-1$ denotes a gene of the chromosome.

To evaluate a quantum chromosome, a transfer function called *measure* is applied to convert the quantum states to a classical chromosome representation. For example, each pair (α_i, β_i) is converted to a value $c_i \in \{0, 1\}$ so that the m-qubit chromosome may result in a binary string $\{c_1, c_2, c_3, \ldots, c_m\}$. More specifically, each pair becomes 0 or 1 using the corresponding qubit probabilities $|\alpha_i|^2$ and $|\beta_i|^2$. To perform this conversion, we use the measure function defined as follow:

function MEASURE(α)
  set r to a random number between 0 and 1;
  if $r > |\alpha|^2$ **then**
    return 0;
  else
    return 1;
  end if
end function

The superposition property in the quantum qubit makes a quantum chromosome encompasses exponentially more data compared to a classical chromosome of the same size (similar to what is used in original genetic algorithms). Therefore, applying an operation to the classical chromosome results in a consequence. Whereas, applying an operation to the quantum chromosome results in a superposition of all possible consequences [47]. In other words, in a classical population each chromosome (i.e, individual) can only represent one potential solution for the problem. However, in a quantum population, each chromosome is a superposition of many potential solutions. As we considered, a large number of UAVs are examined in our assumptions, therefore the proposed QIGA helps scanning a larger set of possible solutions in each round and the algorithm converges to better results. The rotation operator in QIGA is similar to the selection operation in the classical genetic algorithms. In classical GA, the selection operator values the individuals with better fitness so that the individuals with lower fitness would have less chance to be selected. However, in QIGA, the rotation operator attempts to push all individuals toward the best individual. Because of this feature, the population converges to the better states promptly [48].

Each binary string is a possible solution which is evaluated via a problem dependent fitness function. In the following section, the proposed fitness function is described.

3.2. Fitness function

The fitness function or evaluation function determines how to fit a solution with respect to the constrained optimization problem. To build the fitness function, the negative sign of the objective function $O(S_k)$, defined in (8), is used. In addition, as the solutions which meet all of the constraints get higher fitness value, the solutions which violate some of the constrains should achieve a lower objective value from the fitness function. To prevent potential violations, a penalty function technique that penalizes the solutions according to amount of constraints' mismatches is applied. Considering these facts, the fitness function is proposed as:

$$F(S_k) = -(O(S_k) + g(S_k)), \tag{12}$$

where $g(S_k)$ is the penalty function such that if there is no violation, its value will be zero, and positive otherwise. $g(S_k)$ is defined as:

$$g(S_k) = \gamma \times \sum_{i=1}^{N_r} max\left(0, \tau_j^k - \sum_{i=1}^{N_s} e_{ij}\right), \tag{13}$$

where γ is the penalty coefficient that controls the weight of the amount of constraints violated.

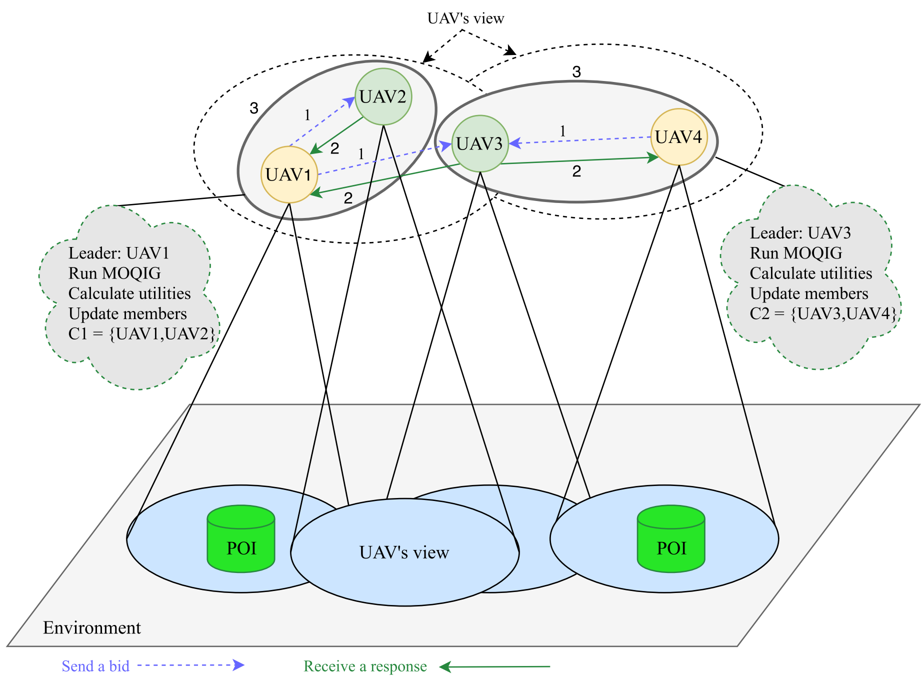


Fig. 1. An example of the proposed leader-follower coalition formation process.

Evolutionary strategies (e.g., crossover and mutation operations) are often considered in GA algorithms to improve their performance. However, as stated in [36], applying the crossover and mutation operators do not significantly improve the performance of the QIGA. Therefore, in the QIGAs, usually a qubit rotation gates operation is used. Qubit (α_i, β_i) of the m-qubit chromosome is updated according to the rotation gates in order to get more or less probabilities to states $|0\rangle$ and $|1\rangle$. Therefore, at each time step t, the update of qubit is performed based on the following rotation gate matrix $L(\theta_i)$:

$$L(\theta_i) = \begin{pmatrix} \cos(\theta_i) & -sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{pmatrix}; \begin{pmatrix} \alpha_i^t \\ \beta_i^t \end{pmatrix} = L(\theta_i)\begin{pmatrix} \alpha_i^{t-1} \\ \beta_i^{t-1} \end{pmatrix}, \tag{14}$$

where θ_i is the amount of angle rotation of qubit gate i. Algorithm 1 presents the pseudo-code for QIGA as described in [36].

Algorithm 1 Quantum-Inspired Genetic Algorithm.

```
t ← 0
Initialize Q(t) as the population
Make P(t) by measuring Q(t)
Evaluate P(t)
Store the best solution b among P(t)
repeat
    t ← t + 1
    Make P(t) by measuring Q(t − 1)
    Evaluate P(t)
    Update Q(t) using quantum gates L(t)
    Store the best solution b among P(t)
until the termination-condition
```

3.3. Multi-UAV coalition formation

Fig. 1 illustrates an example of the proposed coalition formation process where there are two targets (POIs) in the environment. To establish a multi-UAV coalition, a leader-follower coalition formation method is followed. Initially, the UAVs are uniformly distributed in a search space to look for the PoIs. When a PoI is detected by a UAV, this UAV computes the resource requirements of the detected PoI and serves as a leader to form an optimal coalition. After calculating the required resources, the leader UAV calls other UAVs within a certain distance to join it in forming a coalition. Then, the UAVs with at least one of the required resources can respond to this call by reporting the amount of resources that they are able to contribute to help accomplish the task. It is also assumed that the UAVs are self-interested, meaning that if a UAV receives multiple requests, it will consider joining the coalition which offers the highest benefit. The UAV i, u_i measures the value of each request based on travel time to reach the task and the expected cooperative reputation credit received. Thus, its utility value can be defined as:

$$U(u_i, S_k) = \rho_i - \delta a_{ij}, \tag{15}$$

where ρ_i and a_i are the cooperation credit and travel time of u_i when it attempts to join coalition k. δ is the weight indicating the relative significance of the travel time compared to the expected credit. Algorithm 2 shows the pseudo-code for the multi-UAV coalition formation, called Multi-Objective Quantum Genetic Algorithm (MOQGA) coalition formation, and Fig. 2 presents a flow chart of the proposed algorithm.

4. Experimental results

To evaluate the performance of the proposed QIGA-based coalition formation method, two main scenarios with different number of UAVs and PoIs were simulated. It is assumed that the UAVs and PoIs are uniformly distributed across the region and the closest UAV to the PoI is considered to be the one that first detects the PoI and form a coalition (as a coalition leader). It is also assumed that each UAV has five different types of resources. The values for the UAVs' resources are generated with a random uniform distribution and the UAV's resource failure rates are produced randomly in the range $(5 \times 10^{-5}, 10^{-4})$. Furthermore, the execution times of the identified tasks are computed randomly in the range between 10 and 20, depending on the task and the capability of the UAV.

Algorithm 2 Multi-UAV coalition formation using the leader-follower method and QIGA.

Search PoIs in the search space
Initialize each leader as a singleton coalition committed to a single PoI
coalition_members ← [ ]
while there exists an idle UAV around **do**
  for all unsatisfied PoIs so far **do**
    coal_mems ← execute MOQIG method regarding algorithm 1
    coalition_members.append(*coal_mems*)
  end for
  Send bids to UAVs as potential followers
  Calculate the utility values of the followers and receive bid responses
  Update coalition members of each leader with respect to the bid responses
end while

In the first scenario, the resources are being fully recharged after the completion of the mission. It is assumed that the majority of resources depend on the battery usage and during each mission the main portion of the battery is being depleted, therefore the involved UAVs have to return to the station after each mission in order to recharge their consumed resources. While, in the second scenario, the UAVs are able to join new missions (i.e., after completing one) until they have enough resources.

The performance of our proposed algorithm is compared with three well-known algorithms including: i) the distance-based coalition formation method in which the coalitions are formed with the closest UAVs to the leader (i.e., the leader only considers the UAVs in a certain distance of the PoI to be in the coalition and do not evaluate them in terms of cooperative reputation or available resources), ii) the common merge-and-split coalition formation [49], and iii) a Non-Dominated Sorting Genetic Algorithm (NSGA-II) which is a fast, elitist and heuristic-based multi-objective algorithm. Table 3 shows the corresponding parameters for these algorithms.

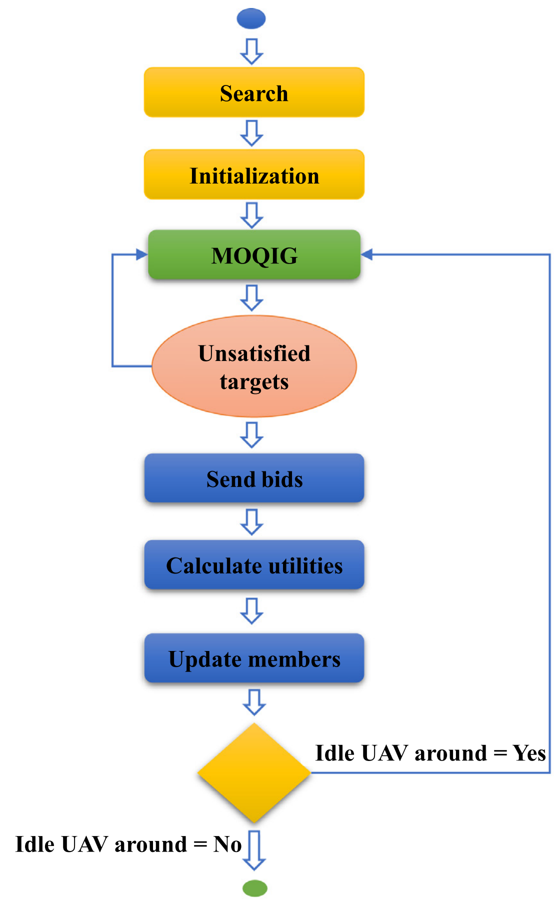


Fig. 2. Flowchart of the proposed algorithm.

Table 4 represents some statistics regarding the completed tasks and resources violations for different algorithms, where the resources violation is defined as the formed coalition exceeds the required resources to perform a given task. It demonstrates that the performance of the proposed coalition formation method is quite

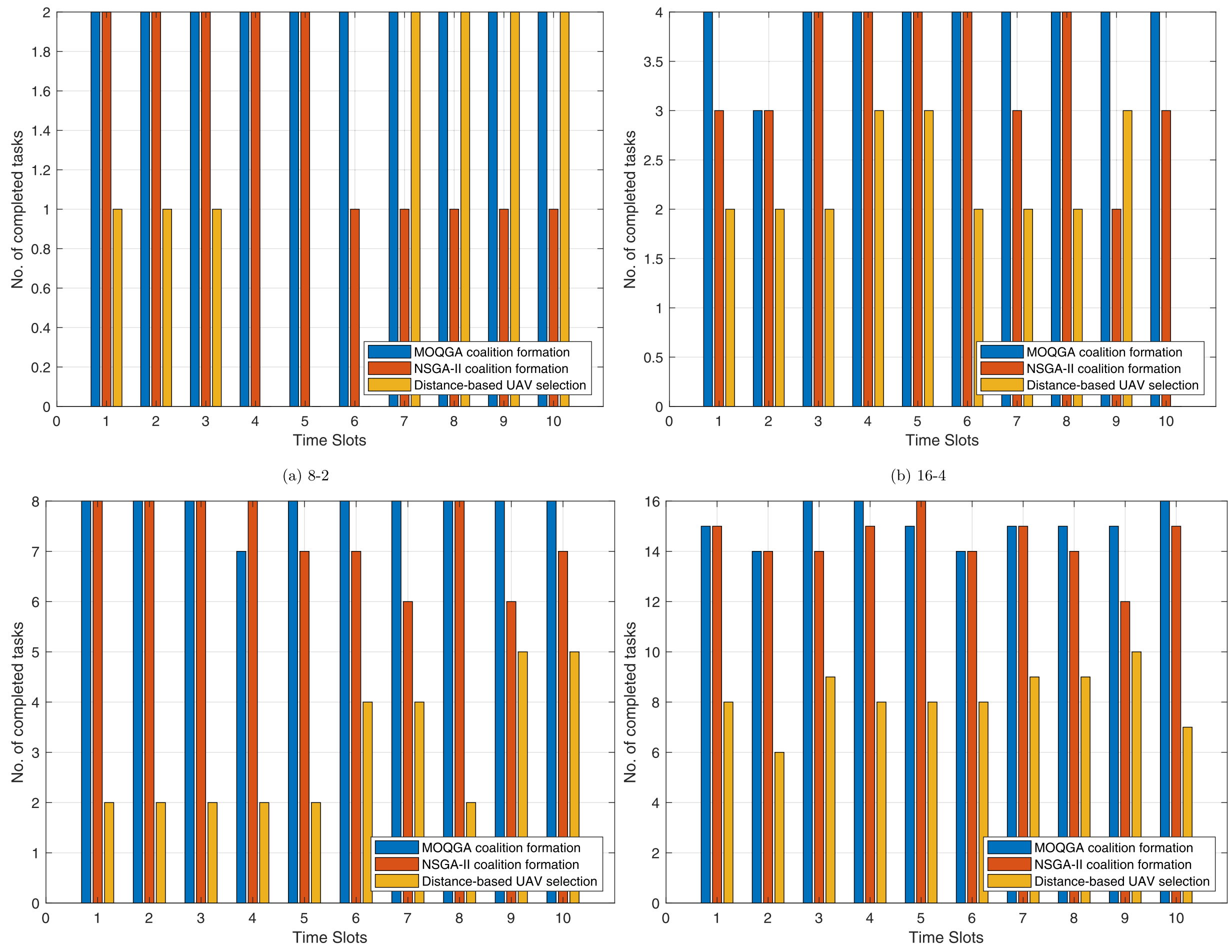
Table 3
Initial values of algorithms' parameters.

Parameter	Method		
	Distance – Based	*NSGA – II*	*MOQGA*
`Population size`	*NA*	200	200
`Maximum number of iterations`	*NA*	500	500
`Number of objectives`	1	3	3
`Mutation probability`	*NA*	10%	*NA*
`Crossover probability`	*NA*	90%	*NA*
`Distribution index for crossover`	*NA*	20	*NA*
`Distribution index for mutation`	*NA*	100	*NA*

NA: Not Available

Table 4
Percentage of completed tasks and average of resource violations for different algorithms in 30 missions (with different numbers of UAVs and tasks).

No. of UAVs and tasks	Method					
	Distance – Based		*NSGA – II*		MOQGA	
	Completed tasks	Resource violations	Completed tasks	Resource violations	Completed tasks	Resource violations
8-2	67%	1.60	80%	0.43	**90%**	**0.30**
16-4	43%	3.83	86%	0.60	**90%**	**0.43**
32-8	39%	6.93	91%	0.73	**97%**	**0.20**
64-16	45%	11.97	92%	2.03	**95%**	**1.23**
128-24	47%	20.33	90%	4.33	**94%**	**2.47**



(a) 8-2 (b) 16-4 (c) 32-8 (d) 64-16

Fig. 3. Comparison of performance of the number of completed tasks for 10 missions separately for the proposed method and other algorithms, when there are different numbers of UAVs and PoIs (targets) in the environment.

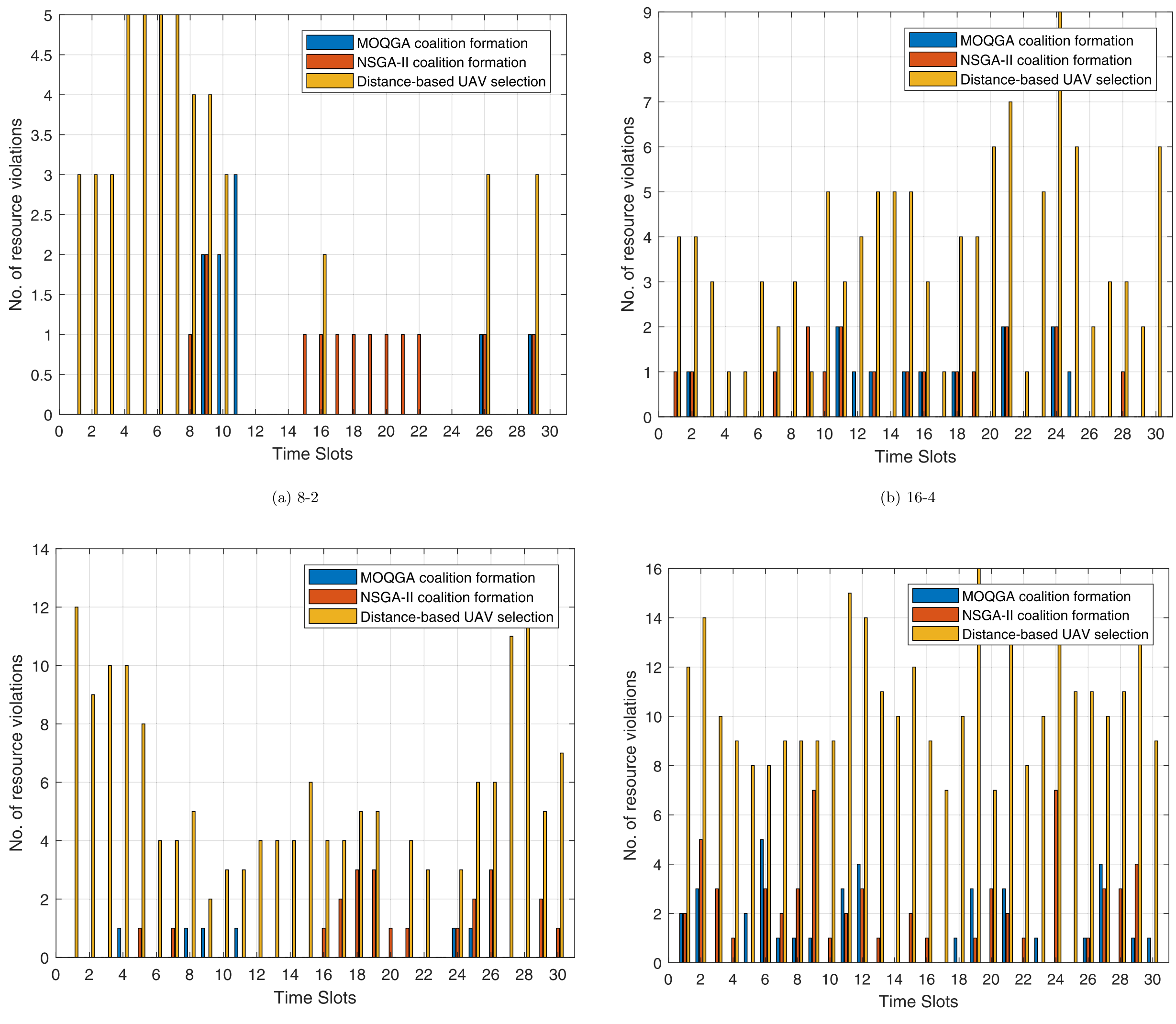
better than other algorithms addressed here in terms of percentage of completed tasks and average of resource violations. We also compared the proposed method against the merge-and-split coalition formation algorithm. The percentage of completed tasks for the merge-and-split method in 30 missions for different numbers of UAVs and tasks were between 46% and 50%, while, as shown in Table 4, our method could achieve rate of 90% in task completion. Regardless of the lower qualities of solutions that the merge-and-split method provides, it also suffers from instability and being time consuming, as it takes a lot of time to compare each two coalitions in the system in order to merge or split.

Fig. 3 shows the number of completed tasks of the proposed coalition formation method compared to the method of selecting the closest UAVs to target (regardless of resources), and NSGA-II for 10 different missions. As shown in this figure, we observe that the MOQGA coalition formation method can form the required coalition successfully to carry out the identified tasks in most of the missions while other methods often fail to do that. Fig. 4 also illustrates the number of resource violations of different coalition formation algorithms for 30 various missions. As seen in Fig. 4, the proposed method is able to complete the assigned tasks in significantly fewer number resource violations compared to other experimented methods. Fig. 5 compares the qualities of solutions (i.e., the Pareto-optimal fronts) of the MOQGA to NSGA-II algorithm. Fig. 5 demonstrates that the performance of the proposed method is better than the NSGA-II algorithm and it results in superior quality solutions in terms of lower cost and higher reliability.

In the second scenario, we experimented how the resources are consumed when the UAVs join the coalitions and showed how it impacts the future missions. Fig. 6 depicts the number of completed tasks of the proposed coalition formation method against the method of selecting the closest UAVs to target and NSGA-II for 10 different missions. As we can see in the figure, the number of completed tasks decrease overtime for all algorithms. The reason is that the resources of included UAVs in the missions are depleted every time they join a mission, and consequently it makes them to not have enough resources for the next missions. Therefore, the number of completed tasks decrease, and accordingly the resource violations for each mission increase. Nevertheless, The proposed coalition formation algorithm still performs better in comparison with other tested methods.

Fig. 7 displays the changes in UAV's cooperative reputation over the course of time. It is assumed that UAVs, u_5 and u_6 are not trustworthy in the sense that they do not consume all the resources they originally committed to when being selected for a coalition. As seen in the figure, the reputations of these UAVs decrease at each time slot, therefore it is less likely that these UAVs are being selected by the leaders over the course of time.

Finally, the impact of reliability in coalition formation is studied in Table 5, where a pre-defined failure rate of 90% is considered



(a) 8-2

(b) 16-4

(c) 32-8

(d) 64-16

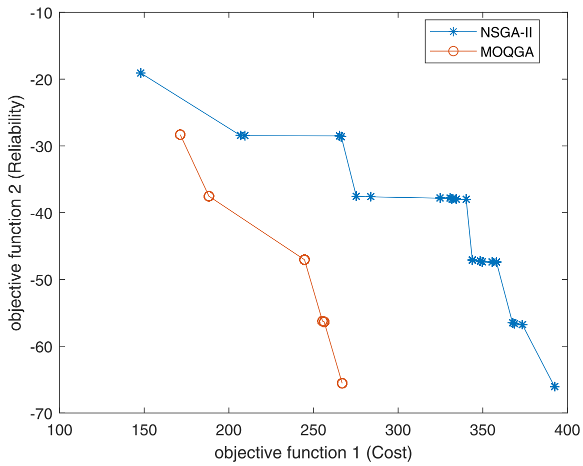
Fig. 4. The number of resource violations for the proposed coalition formation method at each mission compared to the distance-based and NSGA-II algorithms, when there are different numbers of UAVs and tasks in the environment, respectively.

Table 5
The selected UAVs by leaders in 10 missions, assuming that the UAVs u_5 and u_6 have the highest failure rate among all the UAV members (the failure rate of UAVs u_5 and u_6 is 90%).

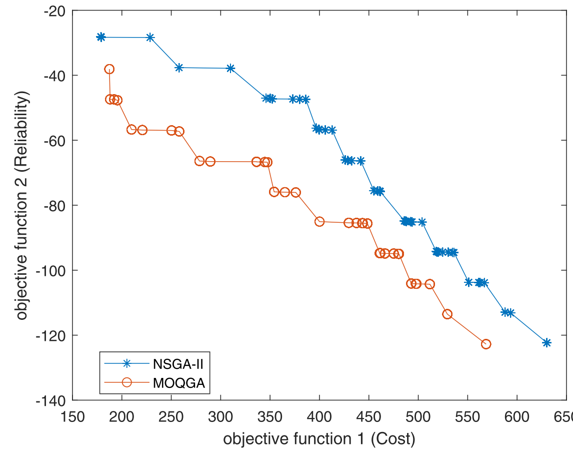
Time slot	Coalition 1	Satisfied	Coalition 2	Satisfied	Unreliable UAVs
1	$L\{u_6\}$; $F\{u_3, u_4, u_8\}$	Yes	$L\{u_1\}$; $F\{u_7, u_2\}$	Yes	{}
2	$L\{u_2\}$; $F\{u_3, u_8, u_4\}$	Yes	$L\{u_1\}$; $F\{u_3, u_7\}$	Yes	{}
3	$L\{u_2\}$; $F\{u_3, u_5, u_8\}$	Yes	$L\{u_6\}$; $F\{u_4\}$	Yes	$\{u_5\}$
4	$L\{u_2\}$; $F\{u_4, u_7, u_3\}$	Yes	$L\{u_1\}$; $F\{u_5, u_6, u_8\}$	No	$\{u_5, u_6\}$
5	$L\{u_5\}$; $F\{u_1, u_3, u_4\}$	Yes	$L\{u_2\}$; $F\{u_7, u_8\}$	Yes	{}
6	$L\{u_1\}$; $F\{u_3, u_4\}$	Yes	$L\{u_2\}$; $F\{u_5, u_6, u_8, u_7\}$	No	$\{u_5, u_6\}$
7	$L\{u_5\}$; $F\{u_2, u_3, u_8\}$	Yes	$L\{u_1\}$; $F\{u_4, u_7\}$	Yes	{}
8	$L\{u_5\}$; $F\{u_2, u_3, u_6\}$	Yes	$L\{u_1\}$; $F\{u_4, u_7\}$	Yes	$\{u_6\}$
8	$L\{u_5\}$; $F\{u_2, u_6, u_7, u_8\}$	Yes	$L\{u_1\}$; $F\{u_3, u_4\}$	Yes	$\{u_6\}$
9	$L\{u_5\}$; $F\{u_2, u_7, u_8\}$	No	$L\{u_1\}$; $F\{u_3, u_4\}$	Yes	{}
10	$L\{u_5\}$; $F\{u_1, u_3, u_4\}$	Yes	$L\{u_2\}$; $F\{u_7, u_8\}$	No	{}

L: Leader, F: Follower

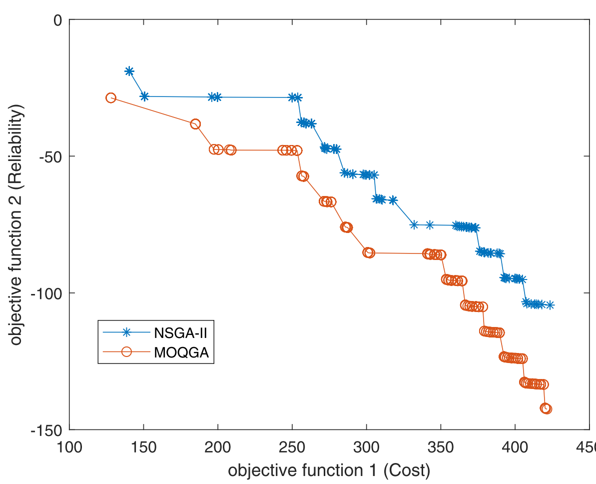
for resources of UAVs u_5 and u_6. Table 5 represents the selected UAVs (e.g., followers) by the leaders where there are two unreliable UAVs. As shown in the table, the proposed method tries to not select unreliable UAVs in most cases. The reason for the selection of unreliable UAVs in some cases is that the problem is a MOOP and the method has to consider other objectives (i.e., cost and reputation) in the coalition formation as well.



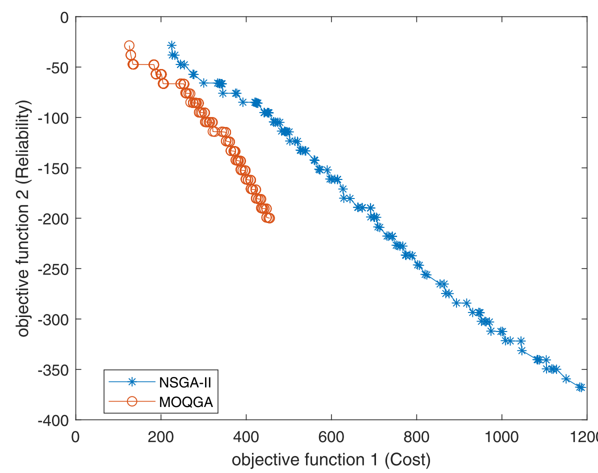
(a) 8-2



(b) 16-4

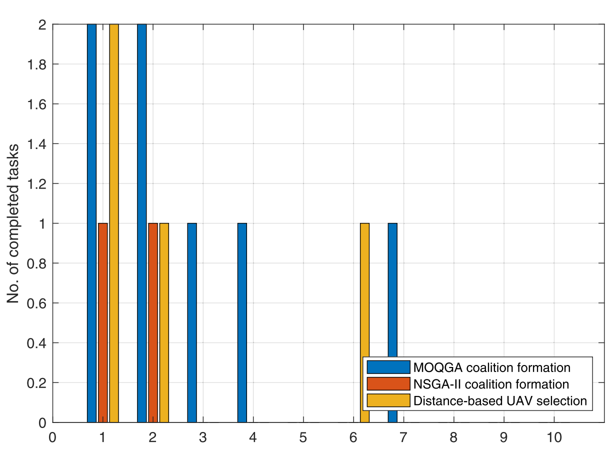


(c) 32-8

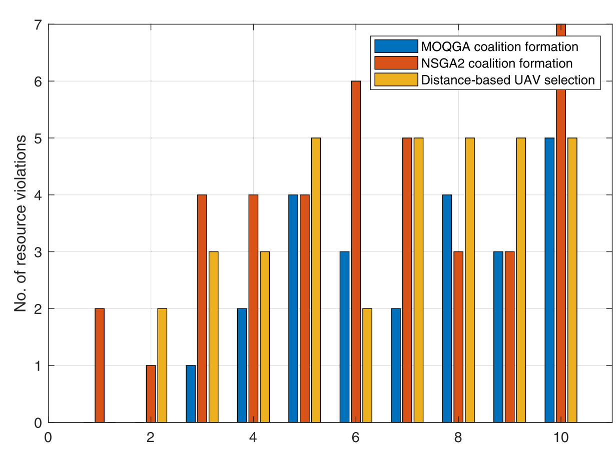


(d) 64-16

Fig. 5. The Pareto-optimal fronts for the simultaneous optimization of the coalition formation cost and reliability (multi-objective optimization) provided by the proposed method (MOQGA) and NSGA-II.



(a) The completed tasks



(b) The resource violations

Fig. 6. The number of completed tasks and the number of resource violations for 10 missions for three different algorithms. Here, it is assumed that the resources are depleted every time the UAVs finish the mission.

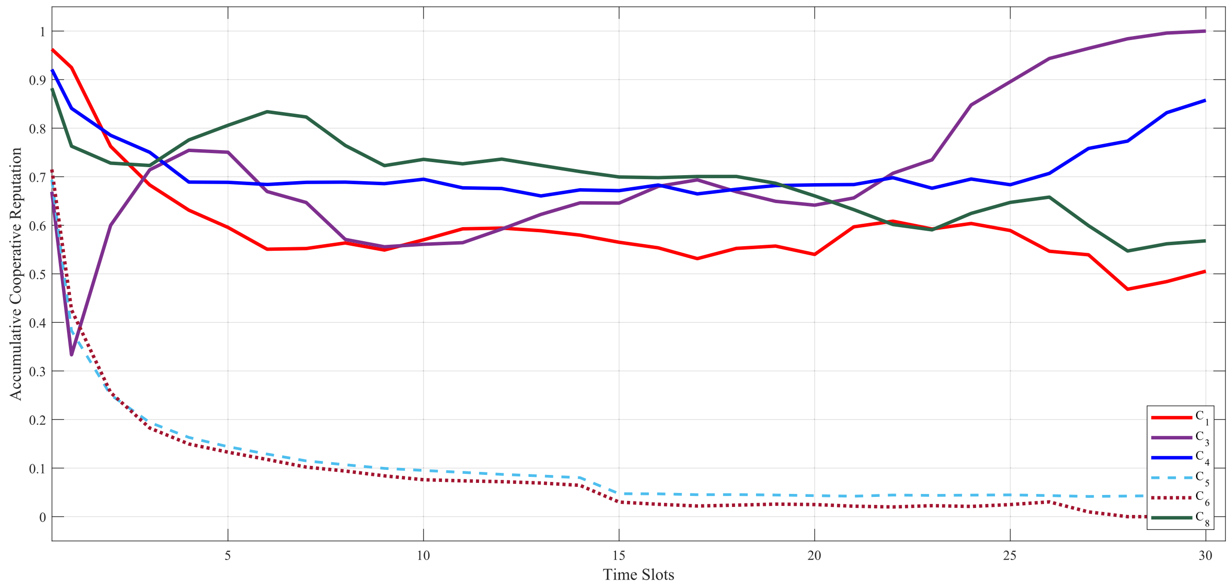


Fig. 7. Changes in UAVs' cooperative reputation over time, where UAVs u_5 and u_6 are assumed to be selfish.

5. Conclusion

In this paper, we proposed a leader-follower UAV coalition formation method to provide a practical solution for distributed task allocation in an unknown environment with a large-scale network of UAVs. Three critical aspects of cost minimization, reliability maximization, and the potential selfish behavior of the UAVs were considered in this coalition formation problem (i.e., a multi-objective optimization coalition formation problem), and a quantum-inspired genetic algorithm is proposed to find the optimal coalitions with a low level of computational complexity. To evaluate the performance of the proposed method, several scenarios with different numbers of tasks ranging from 4 to 24, and a heterogeneous network of UAVs consists of a various number of UAVs ranging from 8 to 124 were experimented. The proposed approach led to promising results compared to existing solutions with respect to completing a higher number of tasks and minimally overspending the resources.

Acknowledgment of Support and Disclaimer

The authors acknowledge the U.S. Government's support in the publication of this paper. This material is based upon work funded by AFRL. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the US government or AFRL.

References

[1] T.W. Sandholm, Distributed rational decision making, Multiagent syst. (1999) 201–258.
[2] S.P. Ketchpel, Coalition formation among autonomous agents, in: European Workshop on Modelling Autonomous Agents in a Multi-Agent World, Springer, 1993, pp. 73–88.
[3] J. Contreras, M. Klusch, J. Yen, Multi-agent coalition formation in power transmission planning: a bilateral shapley value approach., in: AIPS, 1998, pp. 19–26.
[4] W. Saad, Z. Han, M. Debbah, A. Hjorungnes, T. Basar, Coalitional game theory for communication networks, IEEE Signal Process. Mag. 26 (5) (2009) 77–97.
[5] A.R. Korenda, M. Zaeri-Amirani, F. Afghah, A hierarchical stackelberg-coalition formation game theoretic framework for cooperative spectrum leasing, in: Information Sciences and Systems (CISS), 2017 51st Annual Conference on, IEEE, 2017, pp. 1–6.
[6] M. Zaeri-Amirani, F. Afghah, S. Mousavi, A feature selection method based on shapley value to false alarm reduction in icus, a genetic-algorithm approach. (2018) arXiv:1804.11196.
[7] J. Yang, Z. Luo, Coalition formation mechanism in multi-agent systems based on genetic algorithms, Appl. Soft. Comput. 7 (2) (2007) 561–568.
[8] F. Cruz-Mencía, J. Cerquides, A. Espinosa, J.C. Moure, J.A. Rodriguez-Aguilar, Optimizing performance for coalition structure generation problems' idp algorithm, in: Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA), The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2013, p. 706.
[9] F. Bistaffa, A. Farinelli, J. Cerquides, J. Rodríguez-Aguilar, S.D. Ramchurn, Anytime coalition structure generation on synergy graphs, in: Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems, International Foundation for Autonomous Agents and Multiagent Systems, 2014, pp. 13–20.
[10] L. Sless, N. Hazon, S. Kraus, M. Wooldridge, Forming coalitions and facilitating relationships for completing tasks in social networks, in: Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems, International Foundation for Autonomous Agents and Multiagent Systems, 2014, pp. 261–268.
[11] P. Janovsky, S.A. DeLoach, Multi-agent simulation framework for large-scale coalition formation, in: Web Intelligence (WI), 2016 IEEE/WIC/ACM International Conference on, IEEE, 2016, pp. 343–350.
[12] B. Ghazanfari, N. Mozayani, Enhancing nash q-learning and team q-learning mechanisms by using bottlenecks, J. Intell. Fuzzy Syst. 26 (6) (2014) 2771–2783.
[13] B. Ghazanfari, N. Mozayani, Extracting bottlenecks for reinforcement learning agent by holonic concept clustering and attentional functions, Expert Syst. Appl. 54 (2016) 61–77.
[14] A. Shamsoshoara, M. Khaledi, F. Afghah, A. Razi, J. Ashdown, Distributed cooperative spectrum sharing in uav networks using multi-agent reinforcement learning (2018) arXiv:1811.05053.
[15] S.S. Mousavi, M. Schukat, E. Howley, Deep reinforcement learning: an overview, in: Proceedings of SAI Intelligent Systems Conference, Springer, 2016, pp. 426–440.
[16] B. Ghazanfari, M.E. Taylor, Autonomous extracting a hierarchical structure of tasks in reinforcement learning and multi-task reinforcement learning (2017) arXiv:1709.04579.
[17] S.S. Mousavi, B. Ghazanfari, N. Mozayani, M.R. Jahed-Motlagh, Automatic abstraction controller in reinforcement learning agent via automata, Appl. Soft. Comput. 25 (2014) 118–128.
[18] S.S. Mousavi, M. Schukat, E. Howley, P. Mannion, Applying q (λ)-learning in deep reinforcement learning to play atari games, AAMAS Adaptive Learning Agents (ALA) Workshop (2017).
[19] S.S. Mousavi, M. Schukat, E. Howley, Traffic light control using deep policy-gradient and value-function-based reinforcement learning, IET Intel. Transp. Syst. 11 (7) (2017) 417–423.
[20] S. Mousavi, M. Schukat, E. Howley, A. Borji, N. Mozayani, Learning to predict where to look in interactive environments using deep recurrent q-learning (2016) arXiv:1612.05753.
[21] Y. Jin, A.A. Minai, M.M. Polycarpou, Cooperative real-time search and task allocation in uav teams, in: Decision and Control, 2003. Proceedings. 42nd IEEE Conference on, 1, IEEE, 2003, pp. 7–12.
[22] J. Bellingham, M. Tillerson, A. Richards, J.P. How, Multi-task allocation and path planning for cooperating uavs, in: Cooperative Control: Models, Applications and Algorithms, Springer, 2003, pp. 23–41.

[23] H.-L. Choi, A.K. Whitten, J.P. How, Decentralized task allocation for heterogeneous teams with cooperation constraints, in: American Control Conference (ACC), 2010, IEEE, 2010, pp. 3057–3062.
[24] D.J. Kwak, S. Moon, S. Kim, H.J. Kim, Optimization of decentralized task assignment for heterogeneous uavs, IFAC Proc. Volumes 46 (11) (2013) 251–256.
[25] K. Amelin, N. Amelina, O. Granichin, V.V. Putov, Task allocation algorithm for the cooperating group of light autonomous unmanned aerial vehicles, IFAC Proc. Volumes 46 (30) (2013) 152–155.
[26] A. Shamsoshoara, Y. Darmani, Enhanced multi-route ad hoc on-demand distance vector routing, in: Electrical Engineering (ICEE), 2015 23rd Iranian Conference on, IEEE, 2015, pp. 578–583.
[27] A. Rovira-Sugranes, A. Razi, Predictive routing for dynamic uav networks, in: 2017 IEEE International Conference on Wireless for Space and Extreme Environments (WiSEE), 2017, pp. 43–47, doi:10.1109/WiSEE.2017.8124890.
[28] A. Razi, F. Afghah, J. Chakareski, Optimal measurement policy for predicting uav network topology, in: 2017 51st Asilomar Conference on Signals, Systems, and Computers, 2017, pp. 1374–1378, doi:10.1109/ACSSC.2017.8335579.
[29] C. Schumacher, P. Chandler, M. Pachter, L. Pachter, Uav task assignment with timing constraints via mixed-integer linear programming, in: Proc. AIAA 3rd Unmanned Unlimited Systems Conference, 2004.
[30] M.A. Darrah, W.M. Niland, B.M. Stolarik, Multiple uav dynamic task allocation using mixed integer linear programming in a sead mission, Infotech@ Aerospace (2005) 26–29.
[31] C. Schumacher, P.R. Chandler, S.R. Rasmussen, Task allocation for wide area search munitions, in: American Control Conference, 2002. Proceedings of the 2002, 3, IEEE, 2002, pp. 1917–1922.
[32] J. Kennedy, Particle swarm optimization, in: Encyclopedia of Machine Learning, Springer, 2011, pp. 760–766.
[33] M. Dorigo, L.M. Gambardella, Ant colony system: a cooperative learning approach to the traveling salesman problem, IEEE Trans. Evol. Comput. 1 (1) (1997) 53–66.
[34] D.E. Goldberg, Genetic algorithms, Pearson Education India, 2006.
[35] P.J. Van Laarhoven, E.H. Aarts, Simulated Annealing, in: Simulated annealing: Theory and applications, Springer, 1987, pp. 7–15.
[36] K.-H. Han, J.-H. Kim, Genetic quantum algorithm and its application to combinatorial optimization problem, in: Evolutionary Computation, 2000. Proceedings of the 2000 Congress on, 2, IEEE, 2000, pp. 1354–1360.
[37] P.-Y. Yin, S.-S. Yu, P.-P. Wang, Y.-T. Wang, Multi-objective task allocation in distributed computing systems by hybrid particle swarm optimization, Appl Math Comput 184 (2) (2007) 407–420.
[38] F. Afghah, M. Costa, A. Razi, A. Abedi, A. Ephremides, A reputation-based stackelberg game approach for spectrum sharing with cognitive cooperation, in: Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on, IEEE, 2013, pp. 3287–3292.
[39] M. Khaledi, A. Rovira-Sugranes, F. Afghah, A. Razi, On greedy routing in dynamic uav networks (2018) arXiv:1806.04587.
[40] F. Afghah, A. Razi, Game theoretic study of cooperative spectrum leasing in cognitive radio networks, International Journal of Handheld Computing Research (IJHCR) 5 (2) (2014) 61–74.
[41] F. Afghah, A. Shamsoshoara, L. Njilla, C. Kamhoua, A reputation-based stackelberg game model to enhance secrecy rate in spectrum leasing to selfish iot devices (2018) arXiv:1802.05832.
[42] Y. Zhao, M. Song, C. Xin, A weighted cooperative spectrum sensing framework for infrastructure-based cognitive radio networks, Comput Commun 34 (12) (2011) 1510–1517.
[43] Q. He, D. Wu, P. Khosla, Sori: a secure and objective reputation-based incentive scheme for ad-hoc networks, in: Wireless communications and networking conference, 2004. WCNC. 2004 IEEE, 2, IEEE, 2004, pp. 825–830.
[44] J. Granatyr, V. Botelho, O.R. Lessing, E.E. Scalabrin, J.-P. Barthès, F. Enembreck, Trust and reputation models for multiagent systems, ACM Comput. Surv. 48 (2) (2015) 27:1–27:42, doi:10.1145/2816826.
[45] Y. Zhang, L. Lazos, W. Kozma, Amd: audit-based misbehavior detection in wireless ad hoc networks, IEEE Trans. Mob. Comput. 15 (8) (2016) 1893–1907.
[46] S. Soltanali, S. Pirahesh, S. Niksefat, M. Sabaei, An efficient scheme to motivate cooperation in mobile ad hoc networks, in: null, IEEE, 2007, p. 98.
[47] B. Rylander, T. Soule, J. Foster, J. Alves-Foss, Quantum genetic algorithms, in: Proceedings of the 2nd Annual Conference on Genetic and Evolutionary Computation, Morgan Kaufmann Publishers Inc., 2000. 373–373
[48] R. Lahoz-Beltra, Quantum genetic algorithms for computer scientists, Computers 5 (4) (2016) 24.
[49] F. Afghah, M. Zaeri-Amirani, A. Razi, J. Chakareski, E. Bentley, A coalition formation approach to coordinated task allocation in heterogeneous uav networks (2017) arXiv:1711.00214.

Sajad Mousavi received the B.Sc. degree in Computer Engineering from Zanjan University (ZNU), Zanjan, Iran, and the M.Sc. degree in Artificial Intelligence and Robotics from Iran University of Science and Technology (IUST), Tehran, Iran, in 2010 and 2012, respectively. He worked as a research assistant in machine learning and deep learning at the National University of Ireland Galway, Galway, Ireland, from 2015 to 2017. Currently, he is a Ph.D. student in the School of Informatics, Computing & Cyber Systems at Northern Arizona University. His main research interests are machine learning, deep learning, computer vision, multi-agent systems and task allocation.

Fatemeh Afghah received the B.Sc. and M.Sc. degrees (with Hons.) in Electrical Engineering from Khajeh Nassir Toosi University of Technology (K.N.T.U), Tehran, Iran, and the Ph.D. degree in Electrical & Computer Engineering from the University of Maine (UMaine), Orono, ME, USA, in 2005, 2008, and 2013, respectively. She was a Visiting Student with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD, USA, from 2011 to 2012. Currently, she is an Assistant Professor with the School of Informatics, Computing and Cyber Systems, Northern Arizona University (NAU), Flagstaff, AZ, USA, where she is the Director of Wireless Networking and Information Processing (WiNIP) Laboratory. Prior to joining NAU, she was an Assistant Professor with the Electrical and Computer Engineering Department, North Carolina A&T State University, Greensboro, NC, USA, from 2013 to 2015. Her research interests include wireless communication networks, decision making in multi-agent systems radio spectrum management, game theoretical optimization, and biomedical signal processing.

Jonathan Ashdown was born in Niskayuna, NY, USA, in 1984. He received the B.S., M.S., and Ph.D. degrees from Rensselaer Polytechnic Institute, Troy, NY, USA, in 2006, 2008, and 2012, respectively, all in electrical engineering. His Ph.D. dissertation was on a high-rate ultrasonic through-wall communication system using MIMO-OFDM in conjunction with interference mitigation techniques. In 2012, he was a recipient of the Best Unclassified Paper Award at the IEEE Military Communications Conference. From 2012 to 2015, he worked as a Civilian Research Scientist with the Department of Defense (DoD), SPAWAR Systems Center Atlantic, Charleston, SC, USA where he was involved in several basic and applied research projects for the U.S. Navy, mainly in the area of software defined radio. In 2015, he transferred within DoD. He is currently an electronics engineer with Air Force Research Laboratory, Rome, NY, USA, where he is involved in the research and development of advanced emerging technologies for the U.S. Air Force.

Kurt Truck is a communications program manager for the Air Force Research Laboratory in Rome, NY with over 20 years of experience focused on the development and integration of innovative and affordable information transmission and reception technologies to enhance warfighting utility, survivability, and mission assurance. His areas of expertise are Wireless Communications, Mobile Ad-Hoc Networking, Cross Layer Communications, Airborne Networking and Modeling & Simulation. In the past 16 years he has worked in networking and communications for the US Government. Some projects include OPNET modeling for directional tactical radios, Bio-Inspired Networking, and Spectrum Mapping.