IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received February 5, 2018, accepted March 15, 2018, date of publication March 23, 2018, date of current version April 23, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2818733

Multi-Type UAVs Cooperative Task Allocation Under Resource Constraints

LIWEI HUANG[1,2], HONG QU[1,2], (Member, IEEE), AND LIN ZUO[2,3]
[1]The School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China
[2]Institute for the Science and Application of Intelligent Learning, University of Electronic Science and Technology of China, Chengdu 610054, China
[3]The School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China

Corresponding author: Hong Qu (hongqu@uestc.edu.cn)

This work was supported by the National Science Foundation of China under Grant 61573081.

ABSTRACT Coordinated task allocation for multiple unmanned aerial vehicles (multi-UAVs) is an important problem. Taking considerations of the types of UAVs, and the resources are extremely significant in the coordinated control of multi-UAVs. In the interests of assigning tasks efficiently and accurately for the cooperative UAVs of different types, the advanced multi-UAVs control technology requires a universal task assignment method under resource constraints. In this paper, we introduce a novel multi-type UAVs coordinated task allocation method based on cross-entropy (CE), and take the resources required for tasks into account. The CE method takes random samples from the candidate solutions, and then uses them to update the allocation probability matrix. We address the specific processes of CE dealing with the constrained multi-type UAVs task allocation problem, and reveal that CE has the advantage of solving large scale allocation problems. Furthermore, numerical simulations of CE handling task assignment, and comparisons with the exhaust search method are conducted to validate the merits of the cross-entropy method dealing with the considered problem.

INDEX TERMS Multi-type UAVs, cross-entropy, coordinated task allocation, resource constraints.

I. INTRODUCTION

Over the last decades, unmanned aerial vehicles (UAVs) have become an increasingly feasible component of the battlefield environment as well as the civilian applications, such as modern warfare, search and rescue under disaster circumstances, traffic monitoring, planetary exploration, and many other fields [1]–[4]. The UAVs for theses applications may have limited capabilities, and may not have enough required resources to complete the task single-handedly. Therefore, the UAVs need to be deployed in cooperating teams. With increasing attention being paid to cooperating UAVs, the coordinated control technology among them seems especially important which includes UAVs coordinated path planning and their coordinated task assignment. In this paper, we focus mainly on the cooperative task allocation problem of multi-UAVs.

Task allocation has been an active research area for the past few years, and there have been numerous works that study multiple UAVs task allocation problem. Grtli and Johansen [5] dealt with task allocation using Mixed Integer Linear Programming (MILP), although it can preserve global optimality, it suffered from poor scalability. Heuristic approach had been considered in [6] and [7]which gave near-optimal results in real time, allowing it to be used for large scale problem sizes, and for dynamic scenarios. Swarm intelligence algorithms, for example, Ant Colony Optimization (ACO) [8], [9], had fast convergence speed in solving large scale task allocation problems and could obtain effective assignment scheme, however, they had risks falling into local optimum. Decentralized approaches complemented with market-based method which has low computation complexity also have been applied to the multi-agent system. Badreldin *et al.* [10] presented a comparative study between optimization-based and market-based approaches to solve the multi-robot task allocation problem that arose in the context of multi-robots system, and the results showed that the optimization-based approach outperformed the market-based approach in terms of optimal allocation and computational time. Capitan *et al.* [11] proposed a method that decentralized multi-robot partially observable Markov decision precesses (POMDPs) while maintaining cooperation between robots by using POMDP policy auctions, and the auctions provided a flexible way of coordinating individual policies modeled by POMDPs and had low communication requirements. Yaqoob *et al.* [12] presented a new intelligent distributed adaptive scheduling method for real-time tasks with energy harvesting constraints to guarantee a feasible system with a graceful quality of service. Many other algorithms are also applied to solve such problems, see [13]–[21].

The Cross-Entropy (CE) algorithm [22] is a relatively new method solving combinatorial optimization problems, and it was initially used for estimating probabilities of rare events in complex stochastic network. The convergence analysis of CE was also discussed in [23] and [24]. Besides, the authors proved that the CE method is particularly relevant for solving combinatorial optimization problems in [25]. Since then, several recent publications demonstrate the power of the CE method as a simple and efficient tool for many applications, such as vehicle routing [27], buffer allocation [26], machine learning [28]. There are also researches applying the CE method to UAV task allocation. Undurti [29] proposed an algorithm based on CE to solve three different task allocation problems. Le Thi *et al.* [30] also utilized CE method to deal with UAV task assignment problem. However, they take no considerations of specific resource constraints and the UAV types.

In task allocation schemes of UAV domain, the researchers usually assume that one UAV has to be assigned to one target only, and they do not consider the resource requirement of targets or the UAV types. Whereas, sometimes certain kinds of resources are essential in the completion of a target, as well as the types of UAVs. Besides, for large scale task allocation problems, some deterministic methods may not find the optimal solution within an acceptable time. Under this condition, we can treat the CE method as a supplementary to obtain an optimal or near-optimal result. Thus, we intend to investigate the CE method for multiple types of UAVs cooperative task allocation problem under resource constraints.

In this paper, we focus on the task allocation situation where different types of UAVs are assigned to several tasks through CE method, and the tasks require certain kinds of resources. During the allocation process, we firstly determine the number of feasible solutions according to the resource demand of the targets. The evaluation criterion of our allocation scheme is measured by the overall target score, and we aim at finding the optimal allocation scheme with the highest score. After that, we apply the Cross-Entropy method to the considered task allocation problems, and simulations are also conducted to verify the feasibility and effectiveness of CE method in solving the multi-types UAVs task allocation problem.

The rest of paper is organized as follows. Section II elaborates on an introduction of the Cross-Entropy method. In Section III, we describe the problem, giving its mathematical formulation, and the application of CE method for solving the considered problem. Section IV conducts several simulations and comparisons to show the performance and merit of the proposed method. Section V concludes this paper.

II. THE CROSS-ENTROPY METHOD

In this section, we will discuss the main ideas behind the Cross-Entropy (CE) algorithm for combinatorial optimization under our task allocation circumstances. For more detailed derivation processes, see [25] and [30].

We consider the general combinatorial optimization problem,

$$\gamma^* = S(X^*) = \max_{X \in \chi} S(X), \tag{1}$$

where χ is a finite set of states, X is a combination in χ, and S is a real-valued performance function on χ. Our goal is to find the maximum of S over χ, which is denoted as γ^*, and X^* is the corresponding states under γ^*.

In CE, the optimization problem is converted into a probability estimator problem with probability density function (pdf) $f(\cdot; u)$, which is

$$l = \mathbb{P}_u(S(X) \geq \gamma) = \mathbb{E}_u I_{\{S(X)\geq\gamma\}}, \tag{2}$$

where γ is a value close to γ^*, $\mathbb{P}_u$ is the probability measure, $\mathbb{E}_u$ denotes the corresponding expectation operator, and $I(X; \gamma)$ (also written as $I_{\{S(X)\geq\gamma\}}$) is the indicator function depicted as

$$I(\cdot; \gamma) = \begin{cases} 1, & \text{if } S(X) \geq \gamma, \\ 0, & \text{if } S(X) < \gamma. \end{cases} \tag{3}$$

Then, we can estimate l through importance sampling, which is

$$\hat{l} = \frac{1}{N} \sum_{i=1}^{N} I_{\{S(X_i)\geq\gamma\}} \frac{f(X_i; u)}{g(X_i)}, \tag{4}$$

where random samples $X_1, \ldots, X_N$ are drawn from $f(\cdot; u)$ and g is an importance sampling density on χ.

Specially, the best way to estimate l is to use the change of measure with density

$$g^*(X) = \frac{I_{\{S(X)\geq\gamma\}} f(X; u)}{l}. \tag{5}$$

Obviously, g^* depends on the unknown parameter l, and it cannot be acquired directly. Thus, we find another pdf $f(\cdot; v)$ with parameter v on χ via minimizing the cross-entropy between g^* and $f(\cdot; v)$, which is

$$\begin{aligned} &\min_v \mathbb{E}_{g^*} \ln \frac{g^*}{f(\cdot; v)} \\ &= \min_v \left(\int g^*(X) \ln g^*(X) dX - \int g^*(X) \ln f(X; v) dX \right). \end{aligned} \tag{6}$$

Minimizing the problem in (6) is equivalent to solving the maximization problem

$$\max_v \int \frac{I_{\{S(X)\geq\gamma\}} f(X; u)}{l} \ln f(X; v) dX. \tag{7}$$

Finally, we can convert (7) into the following format

$$\max_v \mathbb{E}_u I_{\{S(X)\geq\gamma\}} \ln f(X; v) = \max_v \frac{1}{N} \sum_{i=1}^{N} I_{\{S(X_i)\geq\gamma\}} \ln f(X_i; v). \tag{8}$$

Note that the probability measure l also can be the counting measure in which f is often called a probability mass

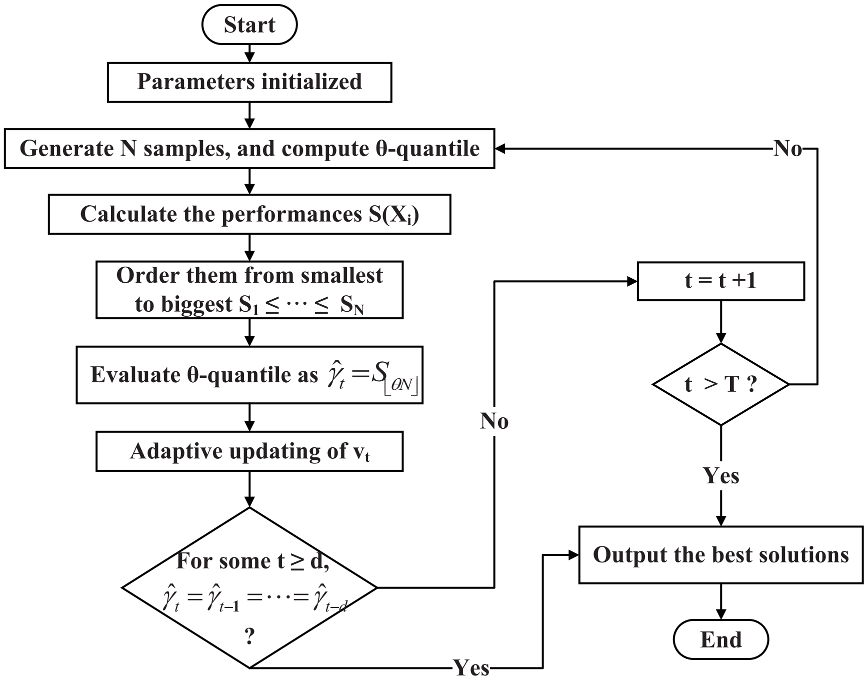

FIGURE 1. The flowchart of CE algorithm for solving combinatorial optimization problems.

function [25]. To express conveniently, we will always use the generic terms density or pdf.

It is plausible that if γ is close to γ^*, then $f(\cdot; v^*)$ assigns most of its probability mass close to X^*. To find γ and v which are close to the optimal values, two sequences of levels $\gamma_1, \ldots, \gamma_T$, and parameters $v_1, \ldots, v_T$ are constructed, and γ_T and v_T are the ones close to the optimal values. Particularly, the initialization phase is done by setting $v_0 = u$, and choosing the quantile θ, then, we can proceed the following two steps.

1. Adaptive updating of γ_t. For a fixed v_{t-1}, let γ_t be the θ-quantile of $S(X)$ under v_{t-1}, which means γ_t satisfies

$$\mathbb{P}_{v_{t-1}}(S(X) \leq \gamma_t) \geq \theta, \tag{9}$$

where $X \sim f(\cdot; v_{t-1})$.

A simple estimator $\hat{\gamma}_t$ of γ_t can be obtained by drawing a random sample $X_1, X_2, \ldots, X_N$ from $f(\cdot; v_{t-1})$, calculating the performances $S(X_i)$ for all i, and ordering them from smallest to biggest: $S_1 \leq S_2 \leq \ldots \leq S_N$. Finally, we estimate the sample θ-quantile as

$$\widehat{\gamma}_t = S_{\lfloor \theta N \rfloor}. \tag{10}$$

2. Adaptive updating of v_t. For fixed γ_t and v_{t-1}, v_t can be derived by solving the aforementioned (8).

Thus, the main procedures for combinatorial optimization problems using CE algorithm can be depicted as follows:

Step 1: Define $\widehat{v}_0 = u$, θ $(0 < \theta < 1)$, the maximum iteration T, and initialize the level counter $t = 1$.

Step 2: Generate a sample $X_1, X_2, \ldots, X_N$ from the density $f(\cdot; \widehat{v}_{t-1})$, then compute the sample θ-quantile $\widehat{\gamma}_t$ of the performance S according to (10).

Step 3: Use the same sample $X_1, X_2, \ldots, X_N$ to solve the stochastic problem in (8), and denote the solution by $\widehat{v}_t$.

Step 4: If for some $t \geq d$ (d is a constant, e.g., $d = 7$),

$$\widehat{\gamma}_t = \widehat{\gamma}_{t-1} = \cdots = \widehat{\gamma}_{t-d},$$

then stop, otherwise, set $t = t + 1$, and goto step 2.

The flow of CE algorithm for combinatorial optimization is shown in Fig. 1.

III. MULTI-TYPE UAVS TASK ALLOCATION USING CE METHOD UNDER RESOURCE CONSTRAINTS

In this section, we will demonstrate the task assignment problem, and apply the CE method to multi-type UAVs task assignment problem under resource constraints.

A. PROBLEM STATEMENT

The problem of multi-type UAVs task allocation is to find the best combination of UAVs to cover certain targets. We suppose that some amount of resources are required by the targets, and the UAV formations must meet the resource demands of the targets to be the qualified candidates, which we call the resource constraints.

Then we formalize the problem of multi-type UAVs task allocation. Suppose that there are m targets and U_t types of UAVs. Specially, each target can be deployed maximum n UAVs, and the UAVs can belong to the same type or the different types. Let N_s be the set of all feasible plans to deploy UAVs to targets, and the size of N_s is equal to l. Note that there are maximum n UAVs in each plan in N_s.

Furthermore, let $X = (x_1, x_2, \ldots, x_m)$ be a random viable assignment vector of plans assigned to m targets. Arbitrary element x_j must belong to N_s and each target j is executed by only one UAV formation k. Thus, X must follow the limitations:

$$\begin{cases} x_j \in N_s, \\ card\{j \in \{1, \ldots, m\} : x_j = k\} = 1, \quad k \in N_s, \end{cases}$$

where card is set cardinality.

Then, let χ be the set of all feasible X satisfied the given resource constraints of targets. Thus, our object is to find the optimal solution vector $\hat{X} \in \chi$ with the largest score. And the score function $S(X)$ defining effectiveness under a certain assignment X is denoted as

$$S(X) = \sum_{j=1}^{m} [B_j - C(x_j)], \tag{11}$$

where B_j is constant, and is the rewarded benefit when completing target j, and $C(x_j)$ is the cost of applying x_j to target j, which is denoted as

$$C(x_j) = W_j e^{\beta t_j} (1 - \prod_{a \in x_j} P_{aj}), \tag{12}$$

where W_j is the threat level of target j; P_{aj} is the success probability of the UAV in plan x_j performing its desired part in target j; $e^{\beta t_j}$ determines the time cost of the UAVs arriving at target j; β is a coefficient, and t_j is the longest time among all the UAVs arriving at the target in a certain solution.

Submit (11) to (12), we have

$$S(X) = \sum_{j=1}^{m} [B_j - W_j e^{\beta t_j} (1 - \prod_{a \in x_j} P_{aj})]. \tag{13}$$

Furthermore, for a certain index γ, we denote that the optimal solution $\hat{X}$ follows $S(\hat{X}) \geq \gamma$. To simplify (13), we introduce $\hat{S}(\hat{X})$, and $\hat{\gamma}$, which are defined as

$$\hat{S}(\hat{X}) = \sum_{j=1}^{m} W_j e^{\beta t_j} (1 - \prod_{a \in x_j} P_{aj}) \leq \sum_{j=1}^{m} B_j - \gamma = \hat{\gamma}. \tag{14}$$

Consequently, to obtain the optimal solution $\hat{X}$ is equal to find a combination of m in N_s which satisfies $\hat{S}(\hat{X}) \leq \hat{\gamma}$. And this can be solved through the Cross-Entropy method.

B. THE CE METHOD FOR MULTI-TYPE UAVS TASK ALLOCATION UNDER RESOURCE CONSTRAINTS

In this subsection, we redefine the indicator function and the pdf in the classical CE method to make it better adjusted to our problem. Also, the detailed procedures for solving the considered optimization problem are revealed.

According to (14), we redefine the indicator function with respect to X for a certain thresholds $\hat{\gamma} \in \mathbb{R}$ in our problem as

$$I(\cdot; \hat{\gamma}) = \begin{cases} 1, & if\ \hat{S}(X) \leq \hat{\gamma}, \\ 0, & if\ \hat{S}(X) > \hat{\gamma}. \end{cases}$$

Let $f(\cdot; v), v \in V$ be a family of (discrete) pdfs on χ, parameterized by a real-valued parameter (vector) v. Given a certain $u \in V$, according to (8) mentioned in Section II, the optimal parameter v^* for a $f(\cdot; v)$ can be estimated by

$$\arg\max_{v} \frac{1}{N} \sum_{i=1}^{N} I(\hat{S}(X_i) \leq \hat{\gamma}) \ln f(X_i; v), \tag{15}$$

where X_i are generated from pdf $f(\cdot; u)$.

Mapped to our problem, we introduce the probability matrix $M_{m \times l}$, in which the element $p(k|j)$ represents the probability of assigning the UAV team k to target j.

$$M = \begin{pmatrix} p(1|1) & p(2|1) & \cdots & p(l|1) \\ p(1|2) & p(2|2) & \cdots & p(l|2) \\ & & \cdots & \\ p(1|m) & p(2|m) & \cdots & p(l|m) \end{pmatrix},$$

which is subjected to $\sum_{k=1}^{l} p(k|j) = 1$.

Consequently, according to M, we define the pdf in the constrained multi-type UAVs coordinated task assignment problem as:

$$f(X; M) = \prod_{j=1}^{m} \prod_{k=1}^{l} p(k|j)^{g(x_j;k)} = \prod_{j=1}^{m} p(x_j|j), \tag{16}$$

where $p(x_j|j)$ is the coefficient in the column x_j and the row j of matrix M, and $g(x; k)$ is an auxiliary function which satisfies

$$g(x; k) = \begin{cases} 1, & if\ x = k, \\ 0, & if\ x \neq k. \end{cases}$$

Then, based on the two important adaptive updating phases in Section II, we firstly suppose that $X_t^1, \ldots, X_t^N$ are the samples drawn in the t-th iteration from $f(X; u)$. Then, we calculate the performances $\hat{S}(X_t^i)$ for all i, and order them from smallest to biggest, $\hat{S}(1) \leq \cdots \leq \hat{S}(N)$. Let $\hat{\gamma}_t^* = \hat{S}(H)$ be H-th smallest sample in the t-th iteration.

Thus, (15) can be rewritten as follows using our pdf $f(X; M)$:

$$\arg\max_{M} \frac{1}{N} \sum_{i=1}^{N} I(\hat{S}(X_t^i) \leq \hat{\gamma}_t^*) \ln f(X_t^i; M). \tag{17}$$

In (17), the indicator function $I(X_t^i; \hat{\gamma}_t^*)$ is already known, and when $N \to \infty$, the problem in (17) is equivalent to

$$\max_{M} \sum_{h=1}^{H} \ln f(X; M). \tag{18}$$

Submit (16) to (18), we rewrite the optimization problem in (18) in the following form

$$\max_{M} \sum_{h=1}^{H} \ln f(X; M)$$
$$= \max_{p(k|j)} \sum_{h=1}^{H} \ln \left(\prod_{j=1}^{m} p(x_j^h|j) \right)$$

$$= \max_{p(k|j)} \sum_{j=1}^{m} \sum_{h=1}^{H} \ln \left(p(x_j^h|j) \right)$$
$$= \max_{p(k|j)} \sum_{j=1}^{m} \sum_{k=1}^{l} card\{h \in \{1,\dots,H\} : x_j^h = k\} \ln(p(k|j)). \tag{19}$$

Then, we redefine that $r_{kj} = p(k|j)$, $a_{kj} = card\{h \in \{1,\dots,H\} : x_j^h = k\}$, and the problem in (19) is equivalent to

$$\begin{aligned} &\min_{r_{kj}} \left(-\sum_{j=1}^{m} \sum_{k=1}^{l} a_{kj} \ln(r_{kj}) \right) \\ &s.t. \sum_{k=1}^{l} r_{kj} = 1, \quad j = 1,\dots,m, \\ &\qquad r_{kj} > 0, \quad j = 1,\dots,m, \ k = 1,\dots,l. \end{aligned} \tag{20}$$

Since it is a convex function, and let $f(r_{kj})$ be the optimization problem in (20), we can form the Lagrangian function

$$\begin{aligned} L(r_{kj}, \lambda, \mu) = f(r_{kj}) + \sum_{j=1}^{m} \lambda_j (\sum_{k=1}^{l} r_{kj} - 1) \\ + \sum_{j=1}^{m} \sum_{k=1}^{l} \mu_{kj}(-r_{kj}), \end{aligned} \tag{21}$$

where λ_j and μ_{kj} are the related restraint coefficient.

In general, the Karush-Kuhn-Tucker (KKT) condition is necessary but not sufficient for optimality. However, for convex optimization problems, the KKT condition is also sufficient [31]. Therefore, to acquire the optimal solution, we just consider the KKT conditions of the problem in (21):

$$\begin{cases} -\dfrac{a_{kj}}{r_{kj}} + \lambda_j - \mu_{kj} = 0, \\ \lambda_j \left(\displaystyle\sum_{k=1}^{l} r_{kj} - 1 \right) = 0, \\ \mu_{kj} r_{kj} = 0, \\ \lambda_j > 0, \\ \mu_{kj} \geq 0, \\ r_{kj} > 0, \end{cases} \tag{22}$$

where $j = 1,\dots,m$, and $k = 1,\dots,l$.

By solving the KKT conditions above, we get

$$\begin{cases} r_{kj} = \dfrac{a_{kj}}{\lambda_j - \mu_{kj}}, \\ \lambda_j = \displaystyle\sum_{k=1}^{l} a_{kj}, \\ \mu_{kj} = 0. \end{cases} \tag{23}$$

Submit λ_j and μ_{kj} to r_{kj}, we have the relationship between r_{kj} and a_{kj}, which is

$$r_{kj} = \frac{a_{kj}}{\sum_{k=1}^{l} a_{kj}}. \tag{24}$$

Furthermore, back to our problem, we get the updating formula of matrix M as follows:

$$p(k|j) = \frac{card\{h \in \{1,\dots,H\} : x_j^h = k\}}{H}. \tag{25}$$

Through the iterative updating of allocation matrix M, the final assignment scheme is obtained. Then, the main steps for CE algorithm solving our problem in (14) are described as follows:

Step 1: Initialize $M = M_0 = (p_0(k|j))_{m\times l}$ a uniform distribution, let n be the number of qualified solutions, then,

$$p_0(k|j) := \frac{1}{n}, \quad j = 1,\dots m, \ k = 1,\dots,l,$$

and choose $\theta \in (0, 1)$.

Step 2: Draw N valid samples $X^1,\dots,X^N$ according to M, and compute $\hat{S}(X^i)$, $i = 1,\dots,N$.

Step 3: Sort the sequence $\hat{S}(X^i)_{i=1}^N$ in the increasing orders. Set $\hat{S}(1) \leq \cdots \leq \hat{S}(N)$ and $H = \lfloor \theta N \rfloor$, then choose H best draws $\hat{S}(1),\dots,\hat{S}(H)$.

Step 4: Update M according to (25).

Step 5: Iterate step 2,3,4 until M converges, which indicates that only 0 and 1 in the matrix, and the sum of the elements in each row is 1.

IV. SIMULATION RESULTS AND ANALYSIS

To evaluate the effectiveness of the proposed algorithm, we applied it to various test problems, and compared it with the Exhaust Search method (ES). The simulations were all implemented in Matlab with version 7.12 programming environment on an Intel Core PC with 8GB memory, and no other programming solver tools were introduced in the following simulations. The system performance is measured by the total cumulative rewards that the UAV teams collects by successfully accomplishing targets during a mission horizon.

TABLE 1. Initial resource capabilities of UAVs in the simulation.

UAV	Initial resources (units)		
	a	b	c
Type A	1	2	2
Type B	1	2	1
Type C	3	3	2

In the following simulations, three types of UAVs with different initial resource capabilities are considered, as shown in Table 1. The speeds of UAVs are assumed to be identical and constant (e.g., 40m/s). The success probability of each type UAV is generated randomly from 0 to 1 for easy calculation, and UAVs with the same type have identical success probability.

For the purpose of economizing resources, we set the maximum number of cooperative UAVs to 3, which means it takes no more than 3 UAVs to achieve the targets, and the total number of each type UAV is not restricted. Therefore, in our cases, 19 solutions are feasible for each target, which are A, B, C, AA, AB, AC, BB, BC, CC, AAA, AAB, AAC, ABB, ACC,

TABLE 2. Information of resources needed, rewarded benefit, and threat level for each target.

Target	Resources (units)			Benefit B_j	Threat level W_j
	a	b	c		
Target 1	2	3	3	30	2
Target 2	5	7	4	70	6
Target 3	4	4	2	50	4
Target 4	4	5	6	100	10
Target 5	3	3	3	120	8
Target 6	3	4	2	40	5
Target 7	1	5	3	65	3
Target 8	2	3	6	78	7
Target 9	7	3	4	35	9
Target 10	1	5	2	63	5

BBB, BBC, BCC, CCC, and ABC, respectively. For other cases, we can also get the feasible solutions easily through a matching algorithm. For easy understanding, we consider 10 targets in the following cases. Resources needed to accomplish the targets are randomly generated, and meet the UAVs maximal cooperate number condition. The information of resources needed, and some parameters for each target are given in Table 2.

In the following simulations, the notations used in the tables are displayed as:

- Ta: targets.
- Res: allocation results.
- Obj_{CE}: value of the objective function obtained by the Cross-Entropy (CE) algorithm.
- Obj_{ES}: value of the objective function obtained by the Exhaust Search (ES) method.
- Time: CPU time in seconds of each case.
- Gap(%) = $(Obj_{ES} - Obj_{CE}) \,/\, Obj_{ES}$.

A. CONSTRAINED TASK ALLOCATION USING CE: TWO CASES

In case 1, we tested multi-type UAVs task allocation under 6 targets, and the allocation results are shown in Table 3.

TABLE 3. Information of CE method allocating UAVs to 6 targets.

Ta	Res	Obj_{CE}
1	AA	273.39
2	AAC	
3	AC	
4	CCC	
5	BC	
6	AC	

From the results table, we can see that two A type UAVs are assigned to target 1, two A type UAVs and one C type UAV are assigned to target 2, one A type UAV and one C type UAV are assigned to complete target 3, three C type UAVs are sent to target 4, one B type UAV and one C type UAV are allocated to target 5, one A type UAV and one C type UAV are assigned to complete target 6, and the total score obtained is 273.39.

To clearly see the iteration processes of the allocation probability matrix, we give the corresponding changes of matrix M in diagram form of each iteration, shown in Figs. 2(a)-3(d).

TABLE 4. Information of CE method allocating UAVs to 10 targets.

Ta	Res	Obj_{CE}
1	AA	422.65
2	AAC	
3	AC	
4	CCC	
5	BC	
6	AC	
7	CC	
8	CCC	
9	BCC	
10	CC	

In the initialization process, each qualified solution has the same probability executing the tasks, as shown in Fig. 2(a). For target 1, solutions 4, 5, 6, 8, and 9 share the same probability of 1/5. The satisfied solutions 12, 14, 16, 17, 18, and 19 have the same probability 1/6 for target 2. Solutions satisfying the resource demands of target 3 are 6, 8, and 9, respectively, thus, they share the same probability 1/3 to carry out target 3. To target 4, 3 solutions satisfy the resource demands, which means solution 12, 14, and 18 share the same probability 1/3. The qualified solutions 6, 8, and 9 have the same probability 1/3 when accomplishing target 5. Solutions satisfying the resource demands of target 6 are 6, 8, and 9, respectively, thus, they share the same probability 1/3 to carry out target 6.

After the first iteration of CE, the probability matrix is updated, as shown in Fig. 2(b). From Fig. 2(b), we can obtain that, for target 1, the capable projects are solution 4 and 6, and the probabilities are 0.65 and 0.35, respectively. The reserved solutions 12, 14, and 18 are remained satisfiable for target 2, and the probabilities are 0.15, 0.5, and 0.35, respectively. To target 3, 4, 5 and 6, the iteration process has been terminated and the final suitable projects are solution 6, 18, 8, and 6, respectively.

Fig. 2(c) shows the probability matrix after CE executing twice, and target 1 has obtained solution 4 as its final project. For target 2, solution 12 and 14 are still remained satisfiable, however, solution 12 has more advantages achieving the target with probability 0.62 over project 14 with probability 0.38.

At the end of the third iteration, the algorithm converges, and the optimal allocation scheme for the 6 targets is obtained, as shown in Fig. 2(d). The final assignment scheme is { 4, 12, 6, 18, 8, 6 }. Correspondingly, the scheme including detailed information of UAVs is { AA, AAC, AC, CCC, BC, AC }, and the terminal score is 273.39 demonstrated in Table 3.

In case 2, we increase the number of targets to 10, the UAV types and resources taken remain unchanged. Then we give the results of UAVs completing 10 targets using CE method, shown in Table 4.

The corresponding changes of allocation probability matrix M in diagram form of each iteration are shown in Figs. 3(a) - 3(e).

In the initialization process, each qualified solution has the same probability executing the tasks, as shown in Fig. 3(a).

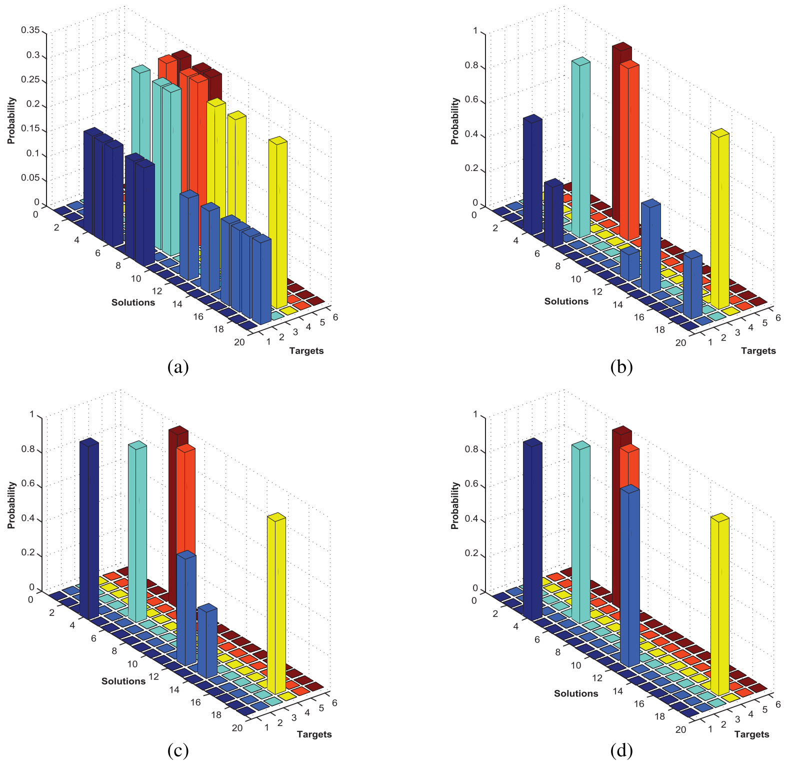

FIGURE 2. Changes of the allocation probability matrix M under 6 targets: (a) Initial allocation probability matrix M_0; (b) Probability matrix M_1 after iteration 1; (c) Probability matrix M_2 after iteration 2; (d) Probability matrix M_3 after iteration 3.

From target 1 to target 6, the allocation probabilities are 1/5, 1/6, 1/3, 1/3, 1/3, and 1/3, respectively. To target 7, solutions 6, 8, and 9 share the same probability 1/3. The qualified solutions 10, 12, 14 and 18 have the same probability 1/4 for target 8. Solutions satisfying the resource demands of target 9 are 14, 17, and 18, respectively, thus, they share the same probability 1/3. To target 10, solutions 6, 8, and 9 share the same probability 1/3.

After the first iteration of CE, the probability matrix is updated, as shown in Fig. 3(b). And we can obtain that, for target 1, the capable projects are solution 4, 6, and 8, and the probabilities are 0.75, 0.1 and 0.15, respectively. The solutions 12, 14, and 18 are remained satisfiable for target 2, with corresponding probabilities 0.4, 0.25, and 0.35, respectively. To target 3 and 4, they both find the final suitable projects, which are solution 6 and 18, respectively. For target 5, the satisfied solutions are solution 6, and 8 with corresponding probabilities 0.35 and 0.65. The solutions 6, and 8 are remained satisfiable for target 6 with the probabilities 0.9 and 0.1, respectively. Target 7 and 10 both filter solution 9 to be their terminal solution. Solutions 12, 14, and 18 are remained satisfiable for target 8, and solution 18 exceed the other two solutions with the probability 0.45, while the probabilities of solution 12 and 14 are 0.4 and 0.15, respectively. To target 9, the remained solution 17 has more advantages to complete the task with probability 0.9 than solution 18.

Fig. 3(c) shows the probability matrix after the algorithm executing twice. All targets, with the exception of targets 1, 2, and 8, have obtained their final projects, which are solution 6, 18, 8, 6, 9, 17, and 9, respectively. Solutions 4 and 6 are both satisfiable for target 1 with the corresponding probabilities 0.9 and 0.1. For target 2, solution 12 excels the other two solutions with the probability 0.6. For target 8, solutions 14 and 18 remain feasible, however, solution 18 has advantage achieving the target with probability 0.9 over project 14 with probability 0.1.

Fig. 3(d) shows the probability matrix after the third iteration, target 1 and 8 have obtained their final projects, which are solution 4 and solution 18 respectively. While for target 2, solution 12, 14 and 18 are still capable with probabilities 0.5, 0.31 and 0.19, respectively.

At the end of the fourth iteration, the algorithm converges, and the optimal allocation scheme for the 10 targets is obtained, shown in Fig. 3(e). The final assignment scheme is $\{4, 12, 6, 18, 8, 6, 9, 18, 17, 9\}$. Correspondingly, the scheme including detailed information of UAVs is $\{AA, AAC, AC, CCC, BC, AC, CC, CCC, BCC, CC\}$, and the terminal score is 422.65 demonstrated in Table 4.

B. COMPARISONS BETWEEN CROSS-ENTROPY AND EXHAUST SEARCH METHOD

To investigate the performance of the CE method in solving the constrained multi-type UAVs task allocation problem, we firstly tested several cases where the number of targets are different. In CE, the objection function value, and the

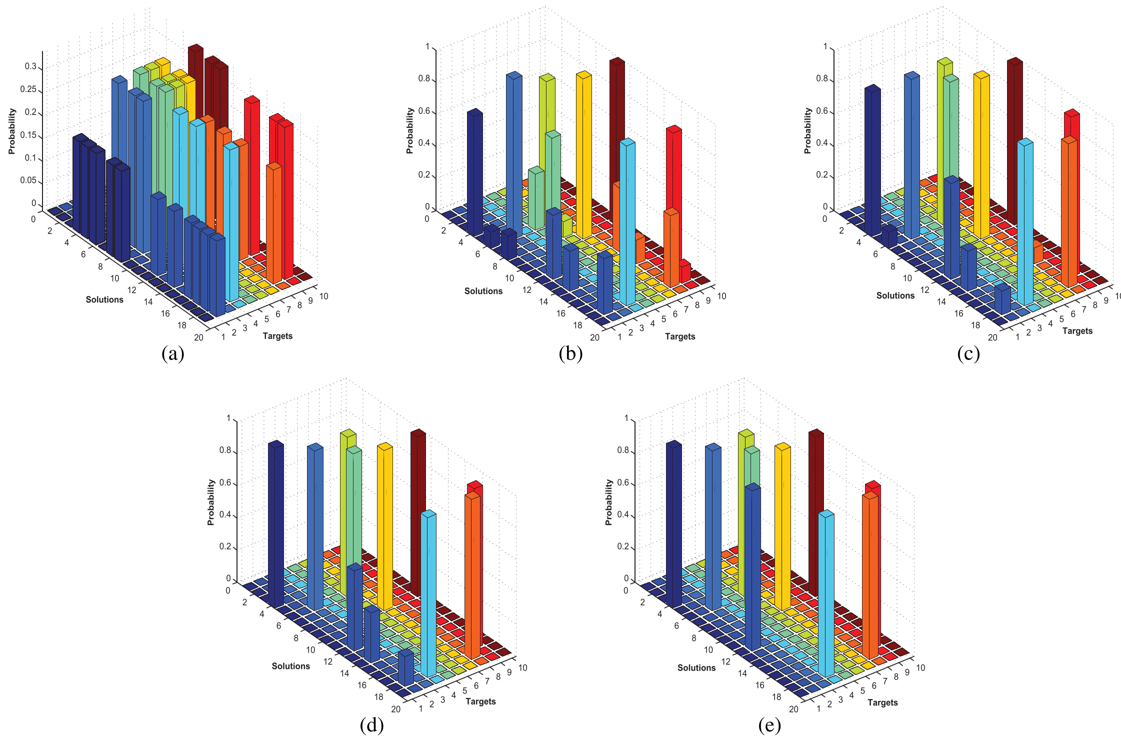

FIGURE 3. Changes of the allocation probability matrix M under 10 targets: (a) Initial allocation probability M_0; (b) Probability matrix M_1 after iteration 1; (c) Probability matrix M_2 after iteration 2; (d) Probability matrix M_3 after iteration 3; (e) Probability matrix M_4 after iteration 4.

TABLE 5. Performance of the CE method.

Target number	Maximal		Minimal		Average		Standard deviation	Samples
	Obj_{CE}	Time(s)	Obj_{CE}	Time(s)	Obj_{CE}	Time(s)		
3	81.06	0.17	81.06	0.11	81.06	0.13	0	1000
6	273.39	0.94	273.39	0.59	273.39	0.66	0	3000
10	422.65	1.88	422.42	0.72	422.64	0.93	0.03	5000
15	576.46	5.84	570.61	6.06	575.43	7.08	0.98	10000
20	851.59	39.69	846.28	41.98	850.68	51.14	1.21	50000
30	1260.0	290.7	1250.0	292.0	1257.0	306.80	2.46	200000

time consumed to obtain that value are the two main factors concerned. Due to the randomly drawn samples, we may get different allocation results each time CE is executed. Therefore, we investigate the maximal, minimal and average values of the Obj_{CE} and Time. Further more, the standard deviations of the objection function values which are obtained through certain execution times of CE are also given out for different cases to reveal the dispersion degree of the objection function values. In each case, θ is set to 0.04, and the number of samples changes. Then, we run CE for 100 times in each case, and the results are shown in Table 5.

From Table 5, we can see that for 3 and 6 targets, the standard deviations are 0, which means the 100 Obj_{CE} values are identical, and only the execution time varies. When the target number increases to 10, 15, 20, and 30, the final results with distinct values come into sight, however, the results are very close for the small standard deviation value. The final results also have something todo with the sample numbers, the larger the sample number, the higher probability to achieve the best result, and the longer the time consumed. It is crucial to balance the best result and the sample number, and sometimes we have to sacrifice time or keep it in an acceptable range to get the optimal or near optimal results especially under large scale problem conditions. While for the off-line assignment issues, we can take our time, and increase the sample number to get better results.

We then compare the Cross-Entropy (CE) method with the Exhaust Search method (ES) in values of objection function, time taken (for CE, we use the average value), and the gap by varying the number of targets. We use ES here because it can get the exact optimal result, and can act as the baseline. The comparison results are shown in Table 6.

From Table 6, we can see that ES is superior to CE in time consumed under a low number of targets, such as 3 targets and 6 targets. However, both CE and ES achieve the same optimal results. When the number of targets increased to 10, CE spent

TABLE 6. Comparisons between CE and ES.

Target number	Objection function value		Time taken (s)		Gap(%)
	Obj_{CE}	Obj_{ES}	CE	ES	
3	81.06	81.06	0.13	0.02	0
6	273.39	273.39	0.66	0.13	0
10	422.64	422.65	0.93	21.84	0.002
14	521.77	522.30	6.15	2784.2	0.10
15	575.43	- -[1]	7.08	- -[1]	- -[1]
20	850.68	- -[1]	51.14	- -[1]	- -[1]
30	1257.0	- -[1]	306.80	- -[1]	- -[1]

[1] "- -" means the computer fail to figure out the allocation scheme due to insufficient memory.

TABLE 7. Maximum value of Obj_{CE} and the percentage.

Target number	$\max(Obj_{CE})$	Obj_{ES}	percentage
3	81.06	81.06	100%
4	147.49	147.49	100%
5	248.23	248.23	100%
6	273.39	273.39	100%
7	328.24	328.24	100%
8	351.61	351.61	100%
9	371.37	371.37	100%
10	422.65	422.65	99%
11	452.20	452.20	96%
12	485.95	485.95	91%
13	460.44	460.44	88%
14	522.30	522.30	85%

averagely 0.93 s to obtain the medial objection function value 422.64, whereas, it took ES 21.84 s to achieve the best value 422.65, and the Gap between them is only 0.002%. For the 14 targets situation, CE is distinctly superior to ES in time taken, since CE took 6.15 s to get the final objection function value 521.77 medially, however, ES took nearly 47 minutes to get 522.30 with Gap 0.10%. When the targets' number reached 15, ES failed to figure out the final allocation results because of insufficient memory. However, CE spent only 7.08 seconds to obtain the objection function value 575.43 averagely. When the number of targets turned to 20, it took CE 51.14 s to get the score value 850.68 averagely, which is much slower compared to 15 targets situation due to the exponentially increase of the solution space. Then, we tested a 30 targets situation, the CE method took 306.80 seconds to acquire the score value 1257.0 medially. Although the time consumed increased exponentially along with the increment of targets numbers, it still stayed capable for those off-line task allocation problems, or we can promote the hardware performance of our machine to tackle the time consumption problem.

What's more, using the ES score values as references, we count the numbers of the maximum CE score values of several cases (in each case, we run CE for 100 times), and the statistics results are illustrated in Table 7.

In the statistics results, we can see that for target numbers 3 to 9, the CE method achieves 100% the same results as ES. For 10 targets, we get 99 (out of 100) times of the best result. Because of the selected samples and sample quantity, for 11 to 14 targets, the percentages of the optimal results descend gradually, however, the optimal results account the majority of all the results.

V. CONCLUSION

In this paper, we propose a novel multi-type UAVs coordinated task assignment method based on Cross-Entropy. We consider different types of UAVs cooperatively accomplishing tasks which need corresponding resources. We apply the CE method to constrained task allocation issue, enriching its sphere of application in solving complicated combinatorial optimization problems with a simple, efficient and general way. Simulation results validate the concrete processes, the feasibility, and effectiveness of the CE method in handling resource constrained multi-type UAVs coordinated task assignment problems.

However, there still exist questions in CE processing optimization problems, such as sample selection, appropriate parameter settings, and so on. Therein, we'll focus on the promotion of CE in its efficiency and time consumption, as well as on task assignment issues in complicated dynamic and uncertain situations for future research.

REFERENCES

[1] J. Scherer *et al.*, "An autonomous multi-UAV system for search and rescue," in *Proc. 1st Workshop Micro Aerial Vehicle Netw., Syst., Appl. Civilian Use*, Florence, Italy, 2015, pp. 33–38.

[2] K. R. Ramsley and J. W. Head, "Exploring mars with micro-UAV squadrons and high data rate communications," in *Proc. 46th Lunar Planetary Sci. Conf.*, Woodlands, TX, USA, 2015, p. 1185.

[3] S. Siebert and J. Teizer, "Mobile 3D mapping for surveying earthwork projects using an Unmanned Aerial Vehicle (UAV) system," *Autom. Construction*, vol. 41, pp. 1–14, May 2014.

[4] B. A. Ritter, "Use of unmanned aerial vehicles (UAV) for urban tree inventories," M.S. thesis, Graduate School, Clemson Univ., Clemson, SC, USA, 2014.

[5] E. I. Grøtli and T. A. Johansen, "Path planning for UAVs under communication constraints using SPLAT! And MILP," *J. Intell. Robot. Syst.*, vol. 65, nos. 1–4, pp. 265–282, Jan. 2012.

[6] C. Sabo, D. Kingston, and K. Cohen, "A formulation and heuristic approach to task allocation and routing of UAVs under limited communication," *Unmanned Syst.*, vol. 2, no. 1, pp. 1–17, Jan. 2014.

[7] J. Wang *et al.*, "A heuristic mission planning algorithm for heterogeneous tasks with heterogeneous UAVs," *Unmanned Syst.*, vol. 3, no. 3, pp. 205–219, Jul. 2015.

[8] J. J. Li, R. B. Zhang, and Y. Yang, "Meta-heuristic ant colony algorithm for multi-tasking assignment on collaborative AUVs," *Int. J. Grid Distrib. Comput.*, vol. 8, no. 3, pp. 135–144, Jun. 2015.

[9] K. Zhao, "Multiple-agent task allocation algorithm utilizing ant colony optimization," *J. Netw.*, vol. 8, no. 11, pp. 2599–2606, Nov. 2013.

[10] M. Badreldin, A. Hussein, and A. Khamis, "A comparative study between optimization and market-based approaches to multi-robot task allocation," *Adv. Artif. Intell.*, vol. 2013, Sep. 2013, Art. no. 256524.

[11] J. Capitan, M. T. J. Spaan, L. Merino, and A. Ollero, "Decentralized multi-robot cooperation with auctioned POMDPs," *Int. J. Robot. Res.*, vol. 32, no. 6, pp. 650–671, May 2013.

[12] I. Yaqoob, E. Ahmed, A. Gani, S. Mokhtar, and M. Imran, "Heterogeneity-aware task allocation in mobile ad hoc cloud," *IEEE Access*, vol. 5, pp. 1779–1795, Feb. 2017.

[13] M. B. Qureshi, M. A. Alqahtani, and N. Min-Allah, "Grid resource allocation for real-time data-intensive tasks," *IEEE Access*, vol. 5, pp. 22724–22734, Oct. 2017.

[14] Y. Deng, P.-P. J. Beaujean, E. An, and E. Carlson, "Task allocation and path planning for collaborative autonomous underwater vehicles operating through an underwater acoustic network," *J. Robot.*, vol. 2013, Dec. 2013, Art. no. 483095.

[15] S. Moon, E. Oh, and D. H. Shim, "An integral framework of task assignment and path planning for multiple unmanned aerial vehicles in dynamic environments," *J. Intell. Robot. Syst.*, vol. 70, nos. 1–4, pp. 303–313, Apr. 2013.

[16] I. Mezei, M. Lukic, V. Malbasa, and I. Stojmenovic, "Auctions and iMesh based task assignment in wireless sensor and actuator networks," *Comput. Commun.*, vol. 36, no. 9, pp. 979–987, May 2013.
[17] P. Sarunic and R. Evans, "Hierarchical model predictive control of UAVs performing multitarget-multisensor tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 3, pp. 2253–2268, Jul. 2014.
[18] W. Meng, Z. He, R. Teo, R. Su, and L. Xie, "Integrated multi-agent system framework: Decentralised search, tasking and tracking," *IET Control Theory Appl.*, vol. 9, no. 3, pp. 493–502, Dec. 2014.
[19] L. Paull, C. Thibault, A. Nagaty, M. Seto, and H. Li, "Sensor-driven area coverage for an autonomous fixed-wing unmanned aerial vehicle," *IEEE Trans. Cybern.*, vol. 44, no. 9, pp. 1605–1618, Sep. 2014.
[20] L. Lin and M. A. Goodrich, "Hierarchical heuristic search using a Gaussian mixture model for UAV coverage planning," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2532–2544, Dec. 2014.
[21] D. S. dos Santos and A. L. C. Bazzan, "Distributed clustering for group formation and task allocation in multiagent systems: A swarm intelligence approach," *Appl. Soft Comput.*, vol. 12, no. 8, pp. 2123–2131, Aug. 2012.
[22] R. Y. Rubinstein, "Optimization of computer simulation models with rare events," *Eur. J. Oper. Res.*, vol. 99, no. 1, pp. 89–112, 1997.
[23] R. Y. Rubinstein, "The cross-entropy method for combinatorial and continuous optimization," *Methodol. Comput. Appl. Probab.*, vol. 1, no. 2, pp. 127–190, Sep. 1999.
[24] R. Y. Rubinstein, "Combinatorial optimization, cross-entropy, ants and rare events," in *Stochastic Optimization: Algorithms and Applications*, vol. 54. Boston, MA, USA: Springer, 2001, pp. 303–363.
[25] P.-T. de Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Ann. Oper. Res.*, vol. 134, no. 1, pp. 19–67, 2005.
[26] G. Alon, D. P. Kroese, T. Raviv, and R. Y. Rubinstein, "Application of the cross-entropy method to the buffer allocation problem in a simulation-based environment," *Ann. Oper. Res.*, vol. 134, no. 1, pp. 137–151, Feb. 2005.
[27] K. Chepuri and T. Homem-de-Mello, "Solving the vehicle routing problem with stochastic demands using the cross-entropy method," *Ann. Oper. Res.*, vol. 134, no. 1, pp. 153–181, 2005.
[28] R. Y. Rubinstein and D. P. Kroese, "The cross-entropy method: A unified approach to combinatorial optimization Monte-Carlo simulation and machine learning," *Technometrics*, vol. 48, no. 1, pp. 147–148, Feb. 2006.
[29] A. Undurti and J. How, "A cross-entropy based approach for UAV task allocation with nonlinear reward," in *Proc. AIAA Guid., Navigat., Control Conf.*, Toronto, ON, Canada, 2010, pp. 1–16.
[30] H. A. Le Thi, D. M. Nguyen, and T. P. Dinh, "Globally solving a nonlinear UAV task assignment problem by stochastic and deterministic optimization approaches," *Optim. Lett.*, vol. 6, no. 2, pp. 315–329, Feb. 2012.
[31] Z.-Q. Luo and W. Yu, "An introduction to convex optimization for communications and signal processing," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1426–1438, Aug. 2006.

LIWEI HUANG received the B.S. degree in information security from the University of Electronic Science and Technology of China, Chengdu, China, in 2013.

She is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, Institute for the Science and Application of Intelligent Learning, University of Electronic Science and Technology of China. Her research interests focus on reinforcement learning, robotics and coordination among agents.

HONG QU (M'09) received the Ph.D. degree in computer science from the University of Electronic Science and Technology of China, Chengdu, in 2006. From 2007 to 2008, he was a Post-Doctoral Fellow with the Advanced Robotics and Intelligent Systems Lab, School of Engineering, University of Guelph, Guelph, ON, Canada. From 2014 to 2015, he was a Visiting Scholar with the Potsdam Institute for Climate Impact Research, Potsdam, Germany, and also with the Humboldt University of Berlin. He is currently a Professor with the Computational Intelligence Laboratory, and the Director of the Institute for the Science and Application of Intelligent Learning, University of Electronic Science and Technology of China. His research interests include neural networks, machine learning, and big data.

LIN ZUO received the Ph.D. degree in computer science from the University of Electronic Science and Technology of China, Chengdu, in 2011. From 2009 to 2010, she was a Visiting Pre-Doctoral Fellow with Northwestern University. She is currently an Associate Professor with the School of Information and Software Engineering, University of Electronic Science and Technology of China. Her research interests include artificial intelligence and neural networks.