



Solving task allocation problem in multi Unmanned Aerial Vehicles systems using Swarm intelligence



Janaína Schwarzrock^{a,*}, Iulisloio Zacarias^a, Ana L.C. Bazzan^a,
Ricardo Queiroz de Araujo Fernandes^b, Leonardo Henrique Moreira^b,
Edison Pignaton de Freitas^a

^a Institute of Informatics Federal University of Rio Grande do Sul, Brazil

^b Software Development Center - Brazilian Army, Brazil

ARTICLE INFO

Keywords:

Unmanned Aerial Vehicles
Task allocation
Multi-agent systems
Swarm intelligence

ABSTRACT

The envisaged usage of multiple Unmanned Aerial Vehicles (UAVs) to perform cooperative tasks is a promising concept for future autonomous military systems. An important aspect to make this usage a reality is the solution of the task allocation problem in these cooperative systems. This paper addresses the problem of tasks allocation among agents representing UAVs, considering that the tasks are created by a central entity, in which the decision of which task will be performed by each agent is not decided by this central entity, but by the agents themselves. The assumption that tasks are *created* by a central entity is a reasonable one, given the way strategic planning is carried up in military operations. To enable the UAVs to have the ability to decide which tasks to perform, concepts from swarm intelligence and multi-agent system approach are used. Heuristic methods are commonly used to solve this problem, but they present drawbacks. For example, many tasks end up not begin performed even if the UAVs have enough resources to execute them. To cope with this problem, this paper proposes three algorithm variants that complement each other to form a new method aiming to increase the amount of performed tasks, so that a better task allocation is achieved. Through experiments in a simulated environment, the proposed method was evaluated, yielding enhanced results for the addressed problem compared to existing methods reported in the literature.

1. Introduction

The use of unmanned aerial vehicles (UAVs) to perform the so called dull, dirty and dangerous (3D) missions is becoming very common. There are many research focusing this theme, such as Shirzadeh et al. (2017), Sun et al. (2015) and Kladis et al. (2011). A special case is the use of UAVs for military purpose (Nonami et al., 2010). New applications with multiple UAVs have been planned (Zheng et al., 2004; Smith and Stengel, 2014; Song et al., 2014; Qu et al., 2015), in which UAVs cooperatively patrol perimeters, monitor areas of interest or escort convoys. Areas of difficult access, borders regions and critical infrastructure are examples of application scenarios that are easier monitored by groups of UAVs.

Several kinds of sensors can be used for monitoring purposes, varying according to the situation. Examples of these sensors are: image sensors like RGB or thermal cameras, chemical sensors, radar sensors, among

others. For instance, RGB cameras can be used in surveillance tasks to identify an object of interest while thermal cameras can be used in search and rescue operations, detection of fire spots or night vision. Each UAV can be equipped with one or more of these sensors, but they have limited resources such as time or energy (batteries or fuel that limit their endurance). In order to deploy an application in which a fleet of UAVs is designed to monitor a given area, these aspects have to be taken into account. If a massive usage of this type of system is considered, a centralized approach to allocate surveillance tasks to the UAVs does not scale (Alighanbari and How, 2005).

In military operations it is common to have a central command unit that coordinates and delegates missions to be performed by military teams acting on the field. However, most commonly, the teams that receive these missions have autonomy to internally decide which members will perform the different parts of the mission. Observing this organization structure, this work focuses on teams of UAVs that must

* Corresponding author.

E-mail addresses: jschwarzrock@inf.ufrgs.br (J. Schwarzrock), izacarias@inf.ufrgs.br (I. Zacarias), bazzan@inf.ufrgs.br (A.L.C. Bazzan), ricardo@cds.eb.mil.br (R.Q. de Araujo Fernandes), moreira@cds.eb.mil.br (L.H. Moreira), epfreitas@inf.ufrgs.br (E.P. de Freitas).

autonomously and cooperatively complete a mission assigned by the central command entity. A team of UAVs is seen as a group that receives a given mission, which contains a set of tasks, and internally has to take care of the division of it among its members.

This problem can be handled as a task allocation among agents, in which the UAVs are the agents and the mission is associated with a set of tasks. Many efforts have ever been made to solve the task allocation problem in several domains, and different approaches have been proposed, such as threshold-based (Ferreira Jr. et al., 2007; Scerri et al., 2005; Ferreira Jr et al., 2010; Ikemoto et al., 2010) and market-based methods (Lemaire et al., 2004; Landén et al., 2010; Ibrí et al., 2012; Tolmidis and Petrou, 2013). In Scerri et al. (2005), for instance, a threshold-based algorithm was proposed to solve the task allocation in a rescue operation scenario. In Landén et al. (2010), an auction-based method was proposed to solve the multi-agent task allocation in the context of a multi-UAV system. Unlike the present problem, in Scerri et al. (2005) and Landén et al. (2010) the tasks are not sent by a central entity, but they are perceived by the agents in the environment.

Swarm intelligence is an appropriate alternative to deal with the multi-UAV task allocation problem in a decentralized way by using a threshold-based approach. Thus, each UAV can decide which tasks it will perform considering only local information, such as its location and resources status. This problem can be modeled using the generalized assignment problem (GAP). The GAP is known to be NP-Complete (Shmoys and Tardos, 1993). In the related literature, there is a heuristic method for task allocation based on swarm intelligence, called Swarm-GAP (Ferreira Jr. et al., 2007) (see Section 2), which allows agents to perform task allocation in an autonomous and decentralized way. In this method, there is no central command unit that has knowledge of the set of tasks. Rather, these tasks are perceived by the agents and they “communicate” (pass information about perceived tasks) to other agents through a token-based communication protocol.

Swarm-GAP presents efficient results when the agents themselves perceive the tasks that need to be performed in the environment in which they act, create the tokens, and send them to other agents. Swarm-GAP works best when several tokens are created, each containing few tasks. However, when a token contains many tasks, as is the case of tokens created by a central command unit, Swarm-GAP is less suitable because it cannot make an efficient use of agents’ resources. The consequence is that many tasks are not selected by the agents, even if they have enough resources to execute them.

As mentioned before, the assumption of the existence of a central commander unit is reasonable: in the case of military operations, the central entity’s role that creates the tasks is of capital importance because this entity has a holistic view of the situation, which facilitates strategic planning. In general, when it comes to tasks that are delegated by a central entity, a token contains multiple tasks. Since the Swarm-GAP algorithm is not efficient for this situation, there is a need to provide a more suitable solution.

Therefore, the contribution of this paper is the proposal of a new method with three algorithm variants, which aim to: (i) allow the agents use their resources to perform as many tasks as possible; (ii) avoid the agents assigning their resources to tasks that are not very suitable for them; and (iii) allow an efficient workload balance among the agents, that is, to prevent some agents from being overloaded with tasks while others remain idle.

The remainder of this paper is organized as follows. In Section 2, the Swarm-GAP algorithm is briefly presented. Section 3 states the problem. The proposed solutions are described in Section 4. The description of the experimental setup is described in Section 5. The experiments and their results are presented and discussed in Section 6. Section 7 discusses related work. Section 8 then makes concluding remarks and indicates future directions.

2. Background

The authors in Ferreira Jr. et al. (2007) have proposed the Swarm-GAP, an algorithm for distributed task allocation based on theoretical models of division of labor in social insect colonies. As the work here proposed is based on the Swarm-GAP, this section briefly presents an overview of this algorithm.

Swarm-GAP allows that each agent chooses which tasks it will perform. This decision is based just on local information, thus leading to little communication the agents. The task allocation is modeled as a generalized assignment problem (GAP) and the goal is to maximize the total capability. Swarm-GAP is a probabilistic approach, by using the mathematical response threshold model formulated by Theraulaz et al. in (Theraulaz et al., 1998).

The response threshold $T_{\theta_{ij}}$ (Eq. (1)) expresses the likelihood of an agent to react to task-associated stimuli st_j and thus to execute it. The threshold θ_{ij} is based on the agent’s capability to perform a task (Eq. (2)). The higher the capacity, the lower the threshold. In case in which the task has a low-stimulus, it is performed by specialized agents. As the stimulus of a task increases (and hence the just mentioned likelihood), less specialized agents also start performing the task (Bonabeau et al., 1999).

$$T_{\theta_{ij}}(st_j) = \frac{st_j^2}{st_j^2 + \theta_{ij}^2} \quad (1)$$

$$\theta_{ij} = 1 - k_{ij} \quad (2)$$

Swarm-GAP uses a communication model based on a token passing protocol, as can be observed in its pseudo code presented in Algorithm 1. Once an agent receives a token (line 1), it decides which tasks it will perform (line 3 to 8). The agent likelihood to choose a task is determined by the tendency $T_{\theta_{ij}}$ (Eq. (1)). This tendency is calculated with the task’s stimulus st_j and threshold θ_{ij} .

The decision also depends on the available resources (line 6), i.e., the agent must have enough resource to perform the task. When the agent decides to carry out a task, this task is allocated to the agent (line 7) and the agent’s resources are reduced accordingly (line 8). After that, the agent is marked as visited (9) and if there are still unallocated tasks, the agent sends the token to an agent which has not yet received that specific token (line 10 to 11).

Algorithm 1: Pseudo code of the Swarm-GAP

```

1: Receive Token
2: Compute available resources  $r_i$ 
3: for all available tasks do
4:   Compute capability  $k_{ij}$ 
5:   Compute tendency  $T_{\theta_{ij}}(st)$ 
6:   if roulette() <  $T_{\theta_{ij}}(st)$  and  $r_i \geq c_j$  then
7:     Allocate task  $j$  to agent  $i$ 
8:     Decrease resource  $r_i$ 
9: Mark agent as visited in the token
10: if there are still available tasks then
11:   Send the token to a not yet visited agent

```

Swarm-GAP also allows that an agent creates a token when it perceives a task in the environment that needs to be done. This feature is not further discussed here because, in the context of this work, the token with the tasks is only created by a central entity. For more details about Swarm-GAP, see Ferreira Jr. et al. (2007) or Ferreira Jr et al. (2010).

3. Problem formulation

In the problem addressed in this work, the central command unit (henceforth referred simply as central) creates the missions and sends

Download English Version:

<https://daneshyari.com/en/article/6854159>

Download Persian Version:

<https://daneshyari.com/article/6854159>

[Daneshyari.com](https://daneshyari.com)