Intelligent Human Systems Integration (IHSI 2025), Vol. 160, 2025, 431–439
https://doi.org/10.54941/ahfe1005847

Optimization-Based Automated Tasking for Complex Multi-Drone Missions

Verónica Mendes Pedro and Axel Schulte

Institute of Flight Systems, Universität der Bundeswehr Munich,
Werner-Heisenberg-Weg 39, 85577 Neubiberg, Germany

ABSTRACT

In this article, we present a concept for generating collaborative behavior for a heterogeneous team of drones with different capabilities that allows executing complex tasks in future military Manned-Unmanned-Teaming (MUM-T) helicopter missions. In these missions, the pilot is required not only to control their own helicopter but also to manage a large number of drones. This scenario quickly creates excess workload, as well as possible underusage of available resources. To prevent this, we create an easy-to-use command interface for the pilot that allows to assign multiple tasks to the drones by selecting only essential parameters. These, in turn, are used to automatically distribute the resources and generate an easily adjustable mission plan. This concept will enable complex multi-drone missions and will be integrated into our mission and cockpit simulator environment for testing and verification by pilots from the German Army.

Keywords: Manned-unmanned teaming, Teaming of UAVs, Automation, Optimization

INTRODUCTION

MUM-T (Manned-Unmanned Teaming) missions consider the cooperation between manned and unmanned aerial vehicles (UAVs) to pursue a common goal. The human pilot specifies the assignments to be completed and assigns them to each UAV of the team. Flying the helicopter while simultaneously tasking and monitoring the unmanned vehicles causes considerable mental workload for the pilot.

Therefore, the way tasking is done is crucial, and it has been a long-standing focus of our research group. One of the main design patterns used is task-based guidance. This concept can be defined as the sharing of authority and pursuit of common goals by all ranks (Uhrmann & Schulte, 2012). In this case, the human agent delegates only high-level tasks to UAV agents, which then must understand, break down, and execute those tasks (Dudek & Schulte, 2022). A task can be defined by a composition of (Lindner, Schwerd, & Schulte, 2019):

- A mission target: which object the mission is performed on.
- An action: how the mission is accomplished.
- Qualifying information: mission-specific parameters or constraints.

For the assignment to succeed, it's important to balance human and automated tasks. Too much automation could cause the pilot to lose trust in the system, while too little automation may overwhelm the pilot with too many tasks. Therefore, it is crucial not only to have the right high-level parameters available, but also to provide adequate feedback. This feedback can come from assistance systems or other means to strive for transparency (Wright, Chen, & Lakhmani, 2019). Assistance systems analyze current tasks and identify if it could produce excessive workload. If so, a relevant intervention is done (Brand & Schulte, 2020). Transparency-focused feedback focuses on the amount of system information that should be provided to the pilot. Even though intuitively an over-availability of information should demand excessive amounts of pilot attention, it has been shown that there is not a significant increase on workload, and that there is higher trust in the system (Wright, Chen, & Lakhmani, 2019).

To tackle this issue in a MUM-T mission with a high number of UAVs, an automated tasking process is proposed, which can be decomposed into four parts:

1. Pilot selects a high-level task.
2. The automation system generates a proposed plan to achieve said goal, which includes:
    a. Selecting the suitable UAV team for the task.
    b. Distributing sub-tasks by the UAVs.
    c. Selecting certain parameters to perform the task optimally.
3. The pilot approves the plan or requests that it is regenerated.
4. The plan is fulfilled, and feedback is returned to the pilot.

Since the tasks that the pilot can choose from are highly generic, it is necessary to define for each one what is the expected UAV behavior and other parameters, such as mission success metrics. This is explored in detail for two specific use-cases: a coordinated attack, where multiple targets need to be engaged at the same time, and reconnaissance, where it is necessary to acquire information on a location or target.

This contribution also focuses on how to apply this method to an attack helicopter simulator with two main distinct types of UAVs: higher value drones that can be equipped with many different types of sensors and weapons; and air launched effects (ALEs) that are small disposable flying objects that can be deployed from the ownship, other UAVs, or even from the ground (launched effects, LE).

DESIGN PATTERNS

The system introduced can be described following two distinct architectures, as proposed by (Schulte, Donath, & Lange, 2016), that can be observed in Figures 1 and 2. The human pilot assigns a task to the UAV agents. In the first example, the assignment goes to a central agent, which then breaks it down into sub-tasks and distributes them to each UAV agent. Therefore, in this case, the individual agents are solely responsible for accomplishing their own small task.

In the latter case, the goal is given to all UAV agents. These must then coordinate between themselves regarding the best way to complete the overall task successfully, and then execute their respective parts.

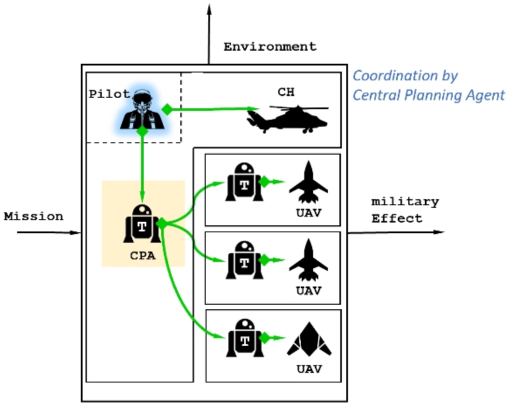

Figure 1: Work system for MUM-T missions with UAV coordination by central planning agent.

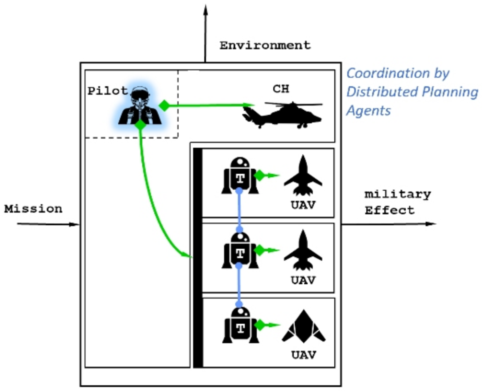

Figure 2: Work system for MUM-T missions with UAV coordination with distributed planning.

HUMAN-SYSTEM INTERACTIONS

To keep the workload at manageable levels only very simple parameters which represent high-level tasks may be available to the pilot. This section focuses on this pilot-system interaction, therefore the generation of UAV behavior is regarded as a "black box", to be explored in the next section.

As we are considering the context of military missions, the high-level tasks considered are commonly used tactics, displayed on a "Quick Tactics" menu. Our two use cases represent these tactics: the coordinated attack, where several targets need to be engaged simultaneously; and reconnaissance, where an area, point or target needs to be analyzed.

Upon selecting the desired tactic, the pilot must also choose some broad configurations, for example whether some of the assets should be kept out of the automation to be used in a different situation. Additionally, some of the parameters must be set by the human operator, such as target confirmation.

During the mission and upon its completion, it is important to provide adequate feedback in order to keep the pilot aware of the current situation while refraining from overloading the pilot's attention.

PROBLEM DEFINITION

To achieve task success, it is important to describe how to get to the desired UAV behavior. Firstly, how each task is broken down; then how success is defined within each smaller task so their individual objective can be set, and then, how the task allocation is done for each UAV and how the task itself is performed.

The problem can be formally formulated (Khamis, Hussein, & Elmogy, 2015) as finding the optimal allocation (A) of a set of tasks (T = {T1, T2, …, Tm}) to a set of UAVs (U = U1, U2, …, Un)

$$\boldsymbol{A} = \boldsymbol{T} \to \boldsymbol{U}. \tag{1}$$

Such that the probability of task (P(A)) success is maximized:

$$\max_{\boldsymbol{A}} \boldsymbol{P}(\boldsymbol{A}) \tag{2}$$

ASSET ALLOCATION

As a first approach, we are considering a centralized solution solely based on resolving the optimization problem. In other words, finding the matrix A (that makes the correspondence between UAVs and tasks) that maximizes the probability of success. However, this solution entails a hierarchy where there is an agent that commands all UAVs. Although this approach results in a very efficient usage of resources, it is also associated with a lack of robustness and, lastly, a low potential for scalability, as too complex optimization problems are associated with considerable computational times, which may not be realistic for real-time plan generation. A way to solve this last issue is to use metaheuristic algorithms, such as using a multiple travelling salesmen formulation, which, although they may not find globally optimal solutions, usually provide satisfactory results (Badreldin, Hussein, & Khamis, 2013). Once the task allocation is completed, the central agent must then compute the optimal parameters for mission success for each UAV.

Another way to achieve adequate task allocation for the different assets is to use decentralized approaches, which are usually more robust, highly scalable and require less communication. However, these methods may provide sub-optimal solutions (Khamis, Hussein, & Elmogy, 2015). Besides, extracting feedback may pose an additional challenge, as the individual solutions may be so distinct and incomparable that creating a success criterion for the whole team may not be possible.

One example are market-based algorithms, such as auctions. In this case, each UAV communicates its utility (for example, the probability that it has of completing the task successfully) for each subtask to the current agent with the role of coordinator. Then, the coordinator evaluates the "bids" based on an optimization strategy and selects the "winning" UAV. The process is then repeated until all the subtasks have been assigned. After the process has been completed, each UAV is responsible for computing the optimal mission parameters to complete the respective subtask.

USE-CASES

Coordinated Attack

Once the pilot selects this task, a set of automated processes begin. Figure 3 is a flowchart on how the interaction between pilot and system is carried out. Firstly, the interface shows the targets available to be selected by the pilot. Then, the automation is activated, and a plan is proposed. The pilot must then accept, decline, or make small alterations to the plan. As this is an engagement task, two assignments are imperatively done by the pilot: confirmation that each target is, in fact, a threat and confirmation of a kill, after engagement. Currently, the pilot needs to select each target individually to sense it with the sensor (from the ownship or from an UAV) and to perform the classification manually. To make the process faster, we propose having this procedure done during the preliminary transfer to target location, and having the sensors automatically point to each target and have a "confirm target" button. The same process is then repeated for battle damage assessment, but with a "confirm destroyed" button.

Looking now into the black box "UAV Behavior Generator", it is first necessary to define the desired goal. In this case, the task is successful once all targets are engaged and destroyed. Therefore, it is crucial to define the probability of kill, once the target is hit, $P(K|H)$, along with the probability of hitting the target, $P(H)$. To be consistent with the model used in the simulator, we consider that a precise number of hits, N, is necessary to achieve a kill (Chircop & Kachoyan, 2018), therefore the previous probability is always 1 after N hits. The probability of kill also changes with the type of weapon that is considered, so, for each UAV. For example, if we consider a standard number of hits for $P(K|H)$, and a certain UAV has a weapon where one hit is equivalent to two standard ones, this expression must be updated accordingly.

A UAV can only be considered capable of engaging a certain target if the probability of kill is above a certain defined threshold. Consequently, it may be necessary to consider more than one UAV to engage a single target.

The initial proposed architecture for these types of missions uses a centralized agent to task the UAVs. Usually, this process is relatively computationally light (as it is only necessary to optimize which UAV covers each target, with each weapon, and what the velocity of approach needs to be), and so it is possible to get optimized results with a satisfactory computational time. As might be expected, if the parameters increase in complexity (for example, to also calculate the best trajectory with an optimal angle of approach), using a centralized architecture may no longer be possible.

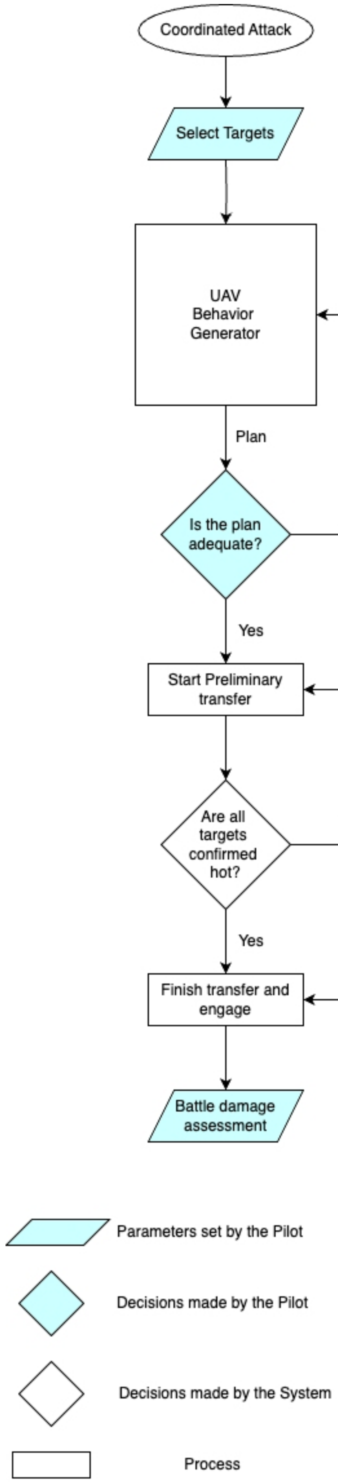

Figure 3: Diagram for coordinated attack system-pilot interaction.

Reconnaissance

The approach for the reconnaissance interface is significantly distinct, as there is no need for target confirmation or battle damage assessment, therefore the interactions (decision and input parameters) by the human operator are significantly reduced. This is evidenced by Figure 4, that presents this process. The pilot only needs to select the mission goals (selecting if it is to survey a point or an area, and if the latter, if it is time-sensitive or not) and set the respective locations. In this instance it is very important to provide appropriate feedback to the pilot. For example, in an extensive search it is only necessary to provide minimum feedback, unless the search finds any targets, or the mission is completed. On the other hand, if the goal is to survey a target, the best approach may be for the trajectory of the UAV to be automated, but for the pilot to assess via sensors the level of threat.

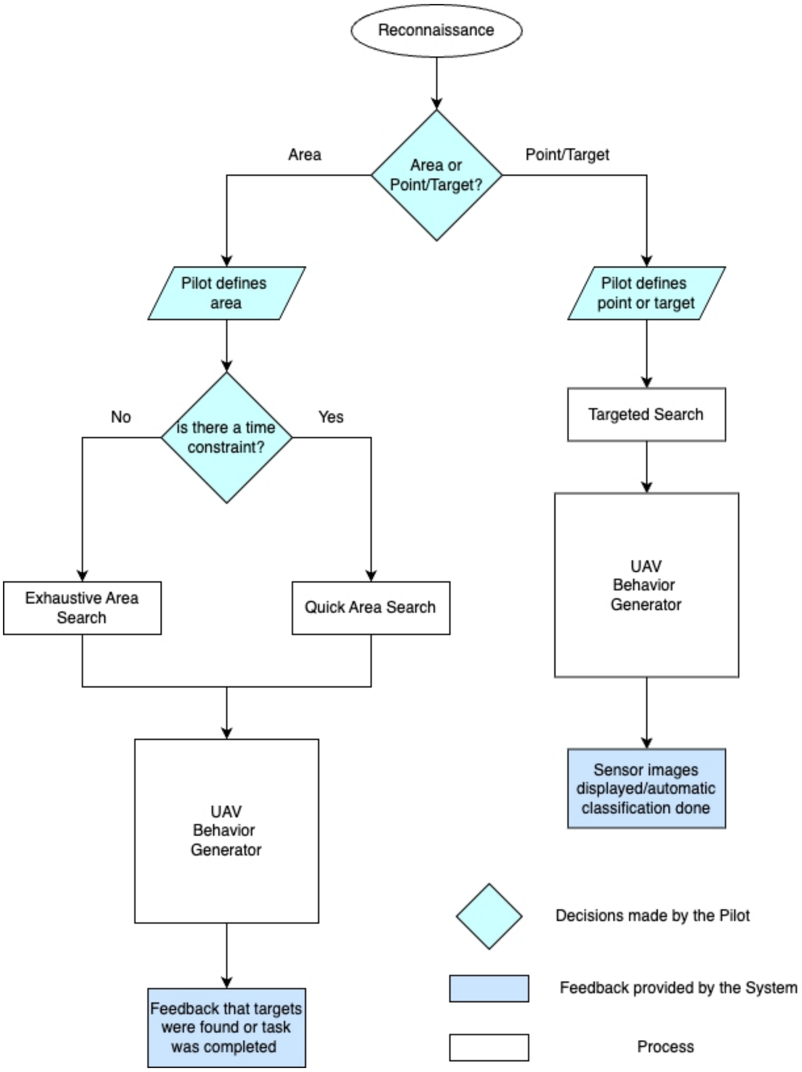

Figure 4: Diagram for reconnaissance attack system-pilot interaction.

In reconnaissance missions, the goal can be to either sweep an area for potential targets or to surveil a particular point or target. In the first case, the mission is successful if all the targets in the area are detected. Although a more extensive sweep is usually a "background" task for the pilot, within a battle context it is not possible to have the UAVs occupied during a time frame of, for example, hours. Therefore, it is always relevant to set a time limit on the UAVs' task execution. This maximum time frame, along with the size of the area chosen by the pilot, determine the optimal number of UAVs to be used. Consequently, the optimization to be done is to be done is determining the UAV trajectories that maximize the probability to find all targets. to be done is determining the UAV trajectories that maximize the probability to find all targets.

In the case where the objective is to surveil a single point or target, the goal is usually to survey the area around the target in a certain pattern trajectory, which will depend on the terrain or on the geometry of the object.

Another criterion for mission success that may be relevant is the ability to complete the mission in high-risk areas without the loss of the UAV. This can be achieved, for example, if we include constraints for concealed flight in the optimization.

Reconnaissance missions usually entail computing an optimized trajectory to maximize surveillance capabilities. Due to the inherent complexity and high number of the variables to be optimized (trajectory points, velocity and asset distribution) a distributed architecture is proposed.

CONCLUSION

This work gives a holistic overview of how UAV team tasks can be implemented. Firstly, with the design of a human-systems interaction approach that aims to find the balance between minimizing the pilot's workload and maintaining confidence in the system by not over-automating it. Then, the drone behavior generation is considered, firstly by modelling use-cases and defining success metrics, and then by considering several arguments and architectures to be used. Implementing these two use-cases, and trying different approaches is going to be the first step into studying UAV team task behavior in our combat helicopter simulator. Then, the system is to be evaluated experimentally by pilots from the German Army.

REFERENCES

Badreldin, M., Hussein, A., & Khamis, A. (2013). A Comparative Study between Optimization and Market-Based Approaches to Multi-Robot Task Allocation. *Advances in Artificial Intelligence*.

Brand, Y., & Schulte, A. (2020). Workload-adaptive and task-specific support for cockpit. *Human-Intelligent Systems Integration*.

Chircop, P. A., & Kachoyan, B. J. (2018). Damage accumulation and probability of kill for gun and target engagements. Naval Research Logistics, 65(2), 160–175.

Dudek, M., & Schulte, A. (2022). Meaningful Guidance of Unmanned Aerial Vehicles in Dynamic Environments. *In Proceedings of the 1st International Conference on Cognitive Aircraft Systems - Volume 1: ICCAS*, 10–14.

Khamis, A., Hussein, A., & Elmogy, A. (2015). Multi-robot Task Allocation: A Review of the State-of-the-ArtMulti-robot Task Allocation: A Review of the State-of-the-Art. *Cooperative Robots and Sensor Networks 2015 Studies in Computational Intelligence*, 31–51.

Lindner, S., Schwerd, S., & Schulte, A. (2019). Defining Generic Tasks to Guide UAVs in a MUM-T Aerial Combat Environment. *Intelligent Human Systems Integration*, 777–782.

Schulte, A., Donath, D., & Lange, D. (2016). Design Patterns for Human-Cognitive Agent Teaming. In: Harris, D. (eds) Engineering Psychology and Cognitive Ergonomics. *Lecture Notes in Computer Science*.

Uhrmann, J., & Schulte, A. (2012). Concept, Design and Evaluation of Cognitive Task-based UAV Guidance. International journal on advances in intelligent systems, 145–15.

Wright, J., Chen, J., & Lakhmani, S. (2019). Agent transparency and reliability in human–robot interaction: The influence on user confidence and perceived reliability. *IEEE Transactions on Human-Machine Systems, 50(3)*, 254–263.