A design of a scheduling system for an unmanned aerial vehicle (UAV) deployment

Sung, Inkyung; Danancier, Kevin; Ruvio, Delphine; Guillemet, Alexandre; Nielsen, Peter

Published in:
IFAC-PapersOnLine

DOI (link to publication from Publisher):
10.1016/j.ifacol.2019.11.472

Publication date:
2019

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
Sung, I., Danancier, K., Ruvio, D., Guillemet, A., & Nielsen, P. (2019). A design of a scheduling system for an unmanned aerial vehicle (UAV) deployment. *IFAC-PapersOnLine*, *52*(13), 1854-1859. https://doi.org/10.1016/j.ifacol.2019.11.472

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 52-13 (2019) 1854–1859

A Design of a Scheduling System for an Unmanned Aerial Vehicle (UAV) Deployment

Inkyung Sung [*] Kevin Danancier [**] Delphine Ruvio [**]
Alexandre Guillemet [**] Peter Nielsen [*]

[*] *Operations Research Group,*
Department of Materials and Production, Aalborg University, Denmark
(inkyung_sung@mp.aau.dk, peter@mp.aau.dk)

[**] *Department of Industrial Engineering, Polytech Engineering School*
of Aix-Marseille University, France
(kevin.danancier@etu.univ-amu.fr, delphine.ruvio@etu.univ-amu.fr,
alexandre.guillemet@etu.univ-amu.fr)

Abstract: Deriving a detailed schedule for an unmanned aerial vehicle (UAV) deployment involves a wide range of scheduling problems, where solving them as a whole is intractable. Moreover, uncertainties in operational environments of the UAV deployment bring a gap between models assumed for the scheduling problems and an actual system. Motivated by the facts, we propose a conceptual design of a scheduling system for a UAV deployment. Specifically, we first introduce fundamental scheduling problems for UAV deployments and specify dependencies between them. A design for a scheduling system, which sequentially solves the scheduling problems and evaluates the solutions using a simulation model, is then proposed. Finally, a prototype of the scheduling system implemented to demonstrate the feasibility of the design is described.

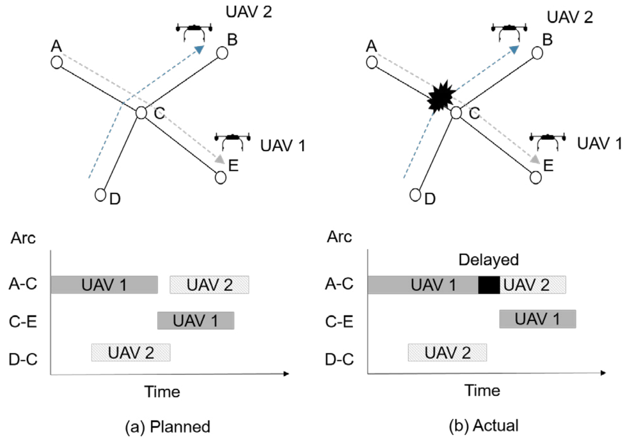
Keywords: Optimisation Methods and Simulation Tools, Scheduling, Decision Support System

1. INTRODUCTION

The technology advance in unmanned aerial vehicles (UAVs) gives rise to opportunities for various applications such as civil engineering, military and logistics, where operations by traditional means are often difficult, dangerous, and costly (Samad et al., 2007; Liu et al., 2014; Dorling et al., 2017). Providing aerial views, flexibility in movements, and a high degree of automation are the main features of a UAV, which bring great benefits to such applications.

One of the challenges in the UAV applications is to derive schedules for UAV deployments. A UAV deployment involves various scheduling problems ranging from a resource allocation to a routing problem, which are complicated to solve. Non-linear dynamics of energy consumption of UAVs, interferences by other UAVs, and other constraints associated with UAV operations make the problems much more difficult to solve.

In addition, an operation of a UAV is often deviated from its schedule in reality due to uncertainties lying on an operational environment of the UAV deployment. Weather, moving objects (e.g. other UAVs) and unknown threats are main sources of the uncertainties. Fig. 1 illustrates a scenario possibly happened by a sudden change in a wind direction causing a delay in a flight duration. In the scenario, to prevent a collision between UAVs, additional efforts to control them are necessary. To avoid such a disrupting adaptation of the schedule, a scheme to derive a schedule taking into account associated uncertainties is important.



(a) Planned

(b) Actual

Fig. 1. Illustration of the impact of uncertainties on a UAV deployment

To tackle the challenges in scheduling for UAVs, we present a conceptual design of a scheduling system for a UAV deployment. We first define fundamental scheduling problems for a UAV deployment and the dependencies between them. The defined scheduling problems are then solved in a hierarchical way considering the dependencies. In the design, the performance of the schedules derived is evaluated

2405-8963 © 2019, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2019.11.472

using an embedded simulation model, which simulates behaviours of UAVs and their operational environment with uncertainties. The evaluation results can be used to update the schedules by adding relevant constraints to the scheduling problems or changing parameters of the problems. The updated scheduling problems will be solved again. Finally, we demonstrate the feasibility of the design by implementing a prototype of the scheduling system.

The remainder of this paper is structured as follows. We first review the underlying scheduling problems for a UAV deployment in §2 and propose a design of a simulation-based scheduling system in §3. A prototype of the scheduling system implemented based on the design is discussed in §4. This paper is concluded in §5 with remarks for future work.

2. SCHEDULING FOR UAV DEPLOYMENTS

Let us first define a task as a job that requires placing of a UAV on a certain position. The requirement of a task varies depending on its type, which can be fulfilled by a UAV. UAVs have different capabilities to fulfil the requirements, thus there is a set of UAVs for a task, which can perform the task. UAVs also have operational constraints such as a maximum payload weight and a maximum flight distance. Under the setting, a goal of a UAV deployment is set as to perform target tasks by available UAVs, while satisfying the tasks' requirements and the operational constraints of the UAVs.

The scheduling problems involved in the UAV deployment can be classified as follows:

- **UAV configuration** Given a set of tasks, to configure UAVs to fulfil requirements of the tasks or to find UAV configurations deployable for the tasks;
- **UAV allocation** Given a set of tasks and available UAVs deployable for the tasks, to allocate UAVs for the tasks, considering the matching between UAVs and tasks;
- **Task sequencing** Upon a UAV allocation for tasks, to determine a sequence of tasks to execute;
- **Path finding** To find a path for a UAV to visit a position of a task;
- **Speed optimization** Given a path, to find flight speeds of a UAV during its flight.

The UAV configuration can be conducted based on standardized descriptive schema (e.g. XML, RDFS, OWL) (Gomez et al., 2008; De Mel et al., 2009). The schema, often termed as ontology, describes tasks, resources and their relations. Based on the description, a collection of resources deployable for a particular task is recommended.

The sets of UAVs selected for each of tasks of interest are then used for the UAV allocation problem. To find optimal allocations of UAVs to the tasks, various approaches have been applied ranging from graph theory (Gomez et al., 2008) to decentralized agent-based approaches (Sensoy et al., 2011; Ponda et al., 2015).

The UAV allocation problem is then followed by the task sequencing problem. Given a set of tasks for a UAV, a sequence of the tasks to execute is determined. When we consider a return of a UAV to its original base as a task, the task sequencing problem can be formulated as a travelling salesman problem (TSP). Mixed integer programming formulations and heuristic algorithms have been actively proposed to solve the TSP for UAVs (Murray and Chu, 2015; Ha et al., 2018). Note that the UAV allocation and task sequencing problems can be formulated together as a classical vehicle routing problem (VRP). See Dorling et al. (2017) for a comprehensive description for VRPs for UAVs.

The path finding is a decision-making that finds a path from a point to another. This becomes critical when a UAV needs to avoid threats or other UAVs during its operation. Zhang et al. (2014) proposes a shortest path algorithm to minimise the exposures to threats for a UAV deployment. Alotaibi et al. (2018) proposes several waypoint generation methods, where a waypoint is a position that a UAV may use to change its flight attribute (e.g. a direction). The path finding problem can also be found in the coverage problem, where a region of interest needs to be searched/inspected by UAVs (Jiao et al., 2010; Bircher et al., 2016).

The speed optimization problem finalizes a path by specifying flight speeds on each segment of the path, where a segment is a flight between two waypoints. This problem has been studied especially to minimise fuel cost associated with the path, as fuel consumption is highly related to a speed of a vehicle (Fagerholt et al., 2010; Franceschetti et al., 2018).

Please refer to Coutinho et al. (2018) for various variants of the scheduling problems described above depending on the characteristics of the problems, for example, types of UAVs and a target environment.

Note that we can observe close relationships between the fundamental scheduling problems, which makes the problems difficult to solve. First, a solution to a scheduling problem (e.g. the task sequencing) becomes an instance of a following scheduling problem (e.g. the path finding). Furthermore, the feasibility of the upstream scheduling problems is often determined by solutions to the downstream scheduling problems as the downstream problems specify the details of a final schedule, i.e. a flight plan, for a UAV deployment. For example, a flight duration between two waypoints assumed in the task sequencing problem may be different from the actual duration calculated based on solutions to the path finding and speed optimization problems.

The relations between the fundamental scheduling problems are illustrated in Fig. 2.

3. SCHEDULING SYSTEM

In order to solve the scheduling problems, we propose a conceptual design of a scheduling system. The proposed scheduling system consists of three main components, the scheduler, the simulator, and the system analyser.

The scheduler is a set of algorithms to solve the scheduling problems. As described in Fig. 2, the problems are interrelated with each other implying a necessity of applying a monolithic approach to get an optimal solution. Thus,

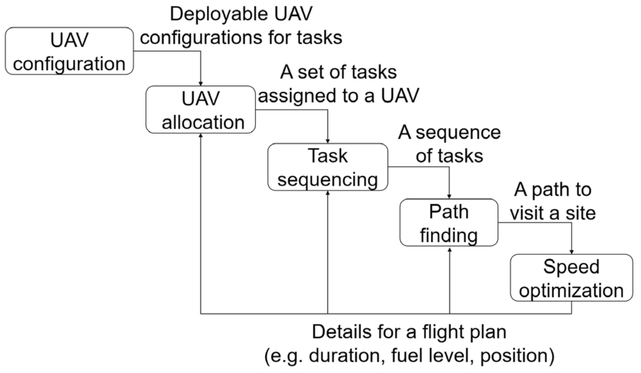

UAV configuration
Deployable UAV configurations for tasks
UAV allocation
A set of tasks assigned to a UAV
Task sequencing
A sequence of tasks
Path finding
A path to visit a site
Speed optimization
Details for a flight plan (e.g. duration, fuel level, position)

Fig. 2. The scheduling problems for a UAV deployment and their relations

the scheduling problems are often formulated as a single problem (Coutinho et al., 2018).

However, due to the complexity of the scheduling problems (Kiesel et al., 2012; Khosiawan et al., 2018), we propose a hierarchical approach, which sequentially solves the fundamental scheduling problems following the relations presented in Fig. 2. A solution to an upstream scheduling problem is derived based on approximated cost/feasibility of the solution. Given the solution, the following scheduling problem is then solved. When the downstream problem cannot find a feasible solution, the upstream problem will be re-solved with the updated constraints (Kiesel et al., 2012).

This approach can reduce the complexity of the scheduling problems (Sawik, 2009). Moreover, it makes the scheduler flexible to changes in a UAV deployment (e.g. introducing additional constraints), since the algorithms for the scheduling problems are modularized requiring a minimum amount of updates on the algorithms to incorporate the changes.

The next main component of the scheduling system is the simulator. Recall that an UAV operation may not work as scheduled in reality due to uncertainties in its operational environment. To evaluate the expected performance of schedules derived by the scheduler, a simulation model is applied, which is a common and rigid approach to evaluate a schedule under uncertainties.

The simulator consists of three parts, the scenario generator, the simulation model and the visualizer. The scenario generator samples scenarios (i.e. possible future situations) based on the dynamics in an operational environment. The simulation model mimics behaviours of UAVs for a target operation. The visualizer displays a schedule and its realization in a scenario.

The simulation model is constructed using as an agent-based model approach, which is suitable to represent the behaviours of UAVs and their interactions with an operational environment. Fig. 3 shows an abstract UML diagram for the simulation model design.

In the diagram, *UAV* is an agent that conducts tasks. The position and state of the UAV agent are updated based on the schedule and the behaviour associated with the agent. *Site* is the location where UAVs and other agents (e.g. task and waypoint) are placed. All actors of the simulation models interact with others in *Environment* agent, where uncertainties (e.g. weather) exist.

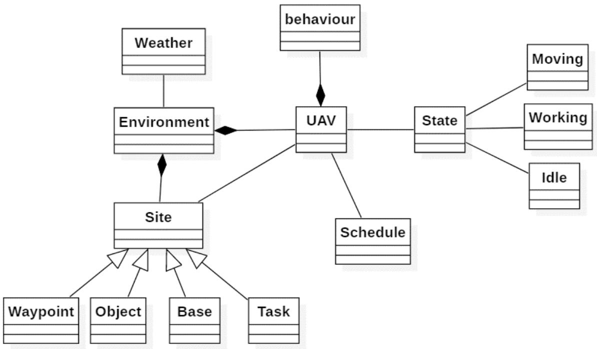


Fig. 3. UML diagram of the simulation model

The system analyser, the last component of the scheduling system, is responsible for updating constraints and parameters of the scheduling problems based on the results provided by the simulator. Specifically, if the performance of schedules derived by the scheduler is poor, the system analyser triggers the scheduler while updating settings for the scheduler to generate a new set of schedules. Otherwise, the process of the scheduling system is terminated reporting final schedules.

The structure of the proposed scheduling system is described in Fig. 4.

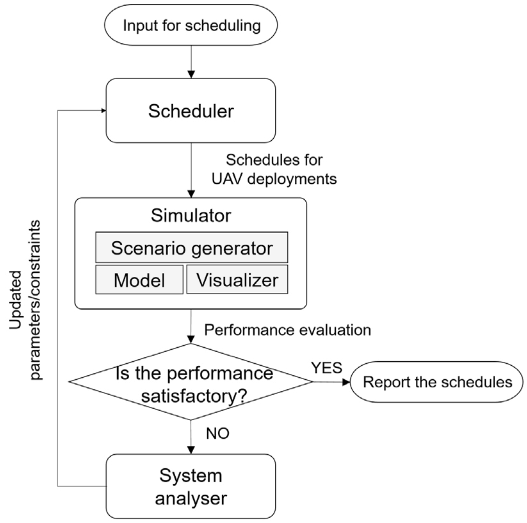


Fig. 4. Simulation-based scheduling system framework

4. PROTOTYPE OF THE SCHEDULING SYSTEM

We present a prototype of the scheduling system implemented following the design in §3. Let us first describe a mission of interest for a UAV deployment. In a region, there are multiple targets to be surveyed within their time windows by placing UAVs on their positions. Threats, which can damage UAVs, exist in the region. Locations of the threats are known, but effective ranges of the threats are uncertain. Under the settings, a mission for a UAV deployment is defined as to survey targets respecting their time windows with the minimum number of UAVs, while minimising an exposure level of the UAVs to threats.

For the target UAV deployment, the UAV allocation, the task sequencing, and the path finding problems are addressed in the prototype scheduling system. Two algorithms are implemented to solve the problems. Specifically, a genetic algorithm (GA) assigns UAVs to tasks and determines sequences of tasks to survey for each UAV. Dijkstra's algorithm then finds a path for a UAV following the sequence, which also guides the UAV to avoid threats in a region.

Main functionalities of the prototype system are summarised as follows:

- Generating schedules, which minimise the number of UAVs necessary to complete tasks within their time windows;
- Computing a flight path to perform a task considering the presence of threats;
- Evaluating schedules under uncertainties.

4.1 Genetic Algorithm

We design a GA that finds the minimum number of UAVs to complete tasks within their time windows and determines sequences of task completions for each UAV. Given N tasks, we first represent a chromosome of the GA as a sequence of integers from one to N, where each integer corresponds to each task.

Following an order specified by a chromosome, the GA then assigns tasks to a UAV until the UAV cannot complete the assigned tasks respecting their time windows. When the time window cannot be respected, the last task assigned to the UAV is re-assigned to a new UAV. These steps are repeated until all tasks are allocated to UAVs. By doing so, given a chromosome, the number of UAVs necessary to complete tasks and sequences of tasks for each UAV are computed. Fig. 5 illustrates the solution decoding steps given a sequence of 10 tasks.

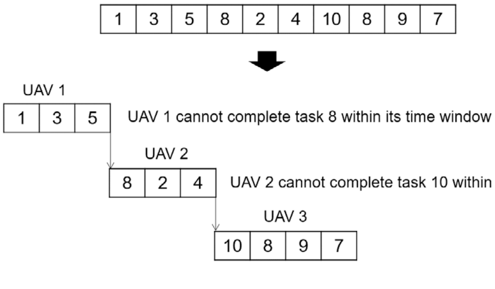

1 3 5 8 2 4 10 8 9 7
UAV 1
1 3 5
UAV 1 cannot complete task 8 within its time window
UAV 2
8 2 4
UAV 2 cannot complete task 10 within its time window
UAV 3
10 8 9 7

Fig. 5. An example of the solution generation in GA for 10 tasks

An initial population, i.e. a set of chromosomes, is created by randomly sampling sequences of integers for tasks. Given a population, chromosomes are then selected by roulette wheel selection and one-point crossover and mutation operators are applied to the selected chromosomes to generate a next population. The GA generates a certain number of populations and returns the best solution yet found.

4.2 Dijkstra's Algorithm for Path Finding

Given a sequence of tasks for a UAV determined by the GA, Dijkstra's algorithm is applied to find a path between two tasks (Soltani et al., 2002). To find the path, we first define a square grid-based network in a region, where a node in the network corresponds to a waypoint, as shown in Fig. 6.

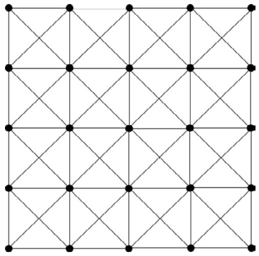


Fig. 6. Network structure for Dijkstra's algorithm

Upon the network structure, a travel cost between two nodes in a network is calculated considering the geographical distance between the nodes and the risk of being at the destination node. Following work of Zhang et al. (2014), the risk of threat t at position (x_t, y_t) to position (x, y), $f_t(x, y)$, can be calculated using a normal distribution:

$$f_t(x, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{d^2}{2\sigma}}, \quad (1)$$

where σ is the standard deviation of the risk and d is the Euclidean distance between (x, y) and (x_t, y_t). Given set of threats T, the risk of being at position (x, y) is then represented as:

$$F(x, y) = 1 - \prod_{t \in T} [1 - f_t(x, y)]. \quad (2)$$

A travel cost from node i to node j, c_{ij}, is finally computed using (2) as follows:

$$c_{ij} = d_{ij} + w \times F_j, \quad (3)$$

where d_{ij} is Euclidean distance between node i and j, w is the weight for threats, and F_j is the risk of being at node j. Finally, with the network structure and the cost function, Dijkstra's algorithm finds the minimum cost path between two task locations.

Note that the GA does not consider the presence of threats when it generates a sequence of tasks to execute for a UAV. For each pair of tasks specified in a sequence given by the GA, Dijkstra's algorithm is applied to find a path between them, which avoids threats. A final schedule for a UAV deployment is then reported by connecting all paths generated from Dijkstra's algorithm.

Fig. 7 shows an example of the schedule highlighting differences between solutions of the algorithms.

4.3 Simulation Model

The schedules derived by the algorithms are evaluated using a simulation model developed by the design presented in Fig. 3. In the simulation model, the location of threats

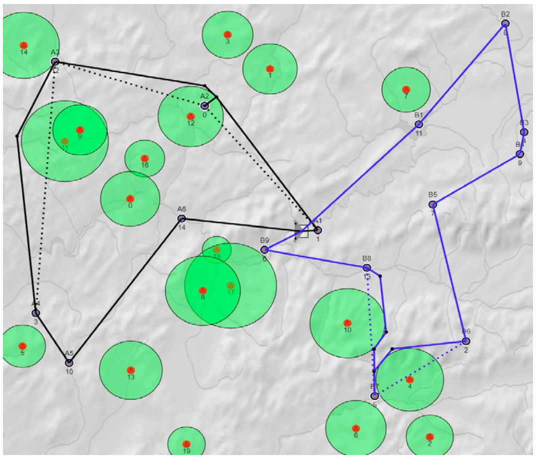


Fig. 7. Paths generated by GA (dotted line) and Dijkstra's algorithm (solid line)

are given, but their effective ranges are unknown, thus represented as random variables in the model.

During a simulation run, UAVs are deployed according to schedules generated. The total travel time of UAVs spent within a range of a threat and their arrival time at each target are reported as an output. *Anylogic*, a commercial simulation tool, is used to develop the simulation model.

4.4 System Analyser

Based on the report from the simulation model, the system analyser updates values for the parameters used in Dijkstra's algorithm, and this is repeated until the scheduling system obtains schedules with an acceptable performance level.

During the steps, the system analyser updates the density of nodes in a network, which determines the elaborateness of a path for UAVs. When the density of nodes in the network is high, a path with sophisticated and complex patterns can be derived by Dijkstra's algorithm. This would be beneficial to find a path under a situation, where threats are densely located. Fig. 8 shows the differences between two paths generated with different values for the density of nodes in a network.

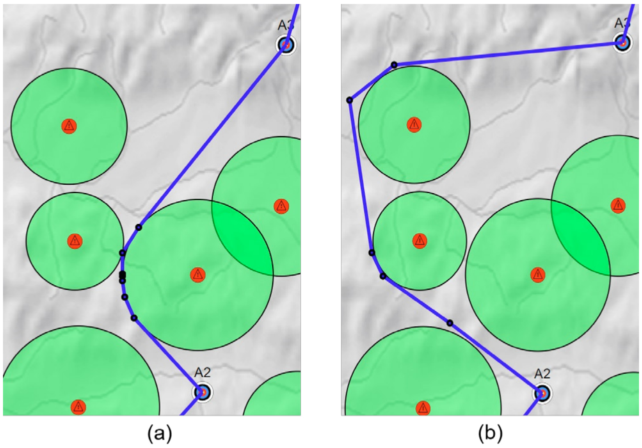


Fig. 8. Path generation in a network (a) with many waypoints (b) with fewer waypoints

As shown in Fig. 8, with many nodes in a network, Dijkstra's algorithm can find a path, which penetrates a small area between two threats resulted in a short travel distance, whereas the algorithm derives a path that bypasses all threats, when the number of nodes in the network is low.

Next, the system analyser updates the weight for threats w, which influences the proximity between a path and threats. As shown in Fig. 9, when we put high value for w, a path being averse to threats is generated (Fig. 9-(a)), whereas a path that passes through a risky area is generated, when the value for w is small (Fig. 9-(b)).

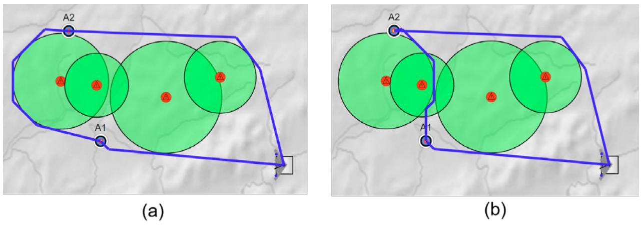


Fig. 9. Paths with different weights for threats (a) high w value (b) low w value

5. CONCLUDING REMARKS

In this study, we present a design of a simulation-based scheduling system for a UAV deployment. The advantages of having such a system can be noted in the following aspects:

- Flexible to address scheduling problems for various UAV operations
- Less effort to develop a scheduling system for different applications
- Less complex to get feasible schedules
- Easy to evaluate schedules for UAV deployments under uncertainties
- Better performance of a UAV deployment in reality

The proposed design of a UAV scheduling system can also be further improved. First, solid and standardized interfaces between the fundamental scheduling algorithms in the scheduler should be designed to fully support the hierarchical approach to the scheduling problems. Next, a way to formulate a fundamental scheduling problem while minimising dependencies and conflicts with other scheduling problems can be addressed to improve the performance of scheduling algorithms. Last, a simplified version of the simulator can be embedded into the scheduler so that the gap between the performances of a schedule estimated by the scheduler and the simulator is reduced.

REFERENCES

Alotaibi, K.A., Rosenberger, J.M., Mattingly, S.P., Punugu, R.K., and Visoldilokpun, S. (2018). Unmanned aerial vehicle routing in the presence of threats. *Computers & Industrial Engineering*, 115, 190–205.

Bircher, A., Kamel, M., Alexis, K., Burri, M., Oettershagen, P., Omari, S., Mantel, T., and Siegwart, R. (2016). Three-dimensional coverage path planning via viewpoint resampling and tour optimization for aerial robots. *Autonomous Robots*, 40(6), 1059–1078.

Coutinho, W.P., Battarra, M., and Fliege, J. (2018). The unmanned aerial vehicle routing and trajectory optimisation problem, a taxonomic review. *Computers & Industrial Engineering*.

De Mel, G., Sensoy, M., Vasconcelos, W., and Preece, A.D. (2009). Flexible resource assignment in sensor networks: A hybrid reasoning approach.

Dorling, K., Heinrichs, J., Messier, G.G., and Magierowski, S. (2017). Vehicle routing problems for drone delivery. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(1), 70–85.

Fagerholt, K., Laporte, G., and Norstad, I. (2010). Reducing fuel emissions by optimizing speed on shipping routes. *Journal of the Operational Research Society*, 61(3), 523–529.

Franceschetti, A., Honhon, D., Laporte, G., and Van Woensel, T. (2018). A shortest-path algorithm for the departure time and speed optimization problem. *Transportation Science*.

Gomez, M., Preece, A., Johnson, M.P., De Mel, G., Vasconcelos, W., Gibson, C., Bar-Noy, A., Borowiecki, K., La Porta, T., Pizzocaro, D., et al. (2008). An ontology-centric approach to sensor-mission assignment. In *International Conference on Knowledge Engineering and Knowledge Management*, 347–363. Springer.

Ha, Q.M., Deville, Y., Pham, Q.D., and Hà, M.H. (2018). On the min-cost traveling salesman problem with drone. *Transportation Research Part C: Emerging Technologies*, 86, 597–621.

Jiao, Y.S., Wang, X.M., Chen, H., and Li, Y. (2010). Research on the coverage path planning of uavs for polygon areas. In *Industrial Electronics and Applications (ICIEA), 2010 the 5th IEEE Conference on*, 1467–1472. IEEE.

Khosiawan, Y., Park, Y., Moon, I., Nilakantan, J.M., and Nielsen, I. (2018). Task scheduling system for uav operations in indoor environment. *Neural Computing and Applications*, 1–29.

Kiesel, S., Burns, E., Wilt, C., and Ruml, W. (2012). Integrating vehicle routing and motion planning. In *Twenty-Second International Conference on Automated Planning and Scheduling*.

Liu, P., Chen, A.Y., Huang, Y.N., Han, J.Y., Lai, J.S., Kang, S.C., Wu, T., Wen, M.C., Tsai, M., et al. (2014). A review of rotorcraft unmanned aerial vehicle (uav) developments and applications in civil engineering. *Smart Struct. Syst*, 13(6), 1065–1094.

Murray, C.C. and Chu, A.G. (2015). The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transportation Research Part C: Emerging Technologies*, 54, 86–109.

Ponda, S.S., Johnson, L.B., Geramifard, A., and How, J.P. (2015). Cooperative mission planning for multi-uav teams. In *Handbook of Unmanned Aerial Vehicles*, 1447–1490. Springer.

Samad, T., Bay, J.S., and Godbole, D. (2007). Network-centric systems for military operations in urban terrain: The role of uavs. *Proceedings of the IEEE*, 95(1), 92–107.

Sawik, T. (2009). Monolithic versus hierarchical approach to integrated scheduling in a supply chain. *International Journal of Production Research*, 47(21), 5881–5910.

Sensoy, M., Le, T., Vasconcelos, W.W., Norman, T.J., and Preece, A.D. (2011). Resource determination and allocation in sensor networks: A hybrid approach. *Comput. J.*, 54(3), 356–372.

Soltani, A.R., Tawfik, H., Goulermas, J.Y., and Fernando, T. (2002). Path planning in construction sites: performance evaluation of the dijkstra, a, and ga search algorithms. *Advanced engineering informatics*, 16(4), 291–303.

Zhang, B., Liu, W., Mao, Z., Liu, J., and Shen, L. (2014). Cooperative and geometric learning algorithm (CGLA) for path planning of uavs with limited information. *Automatica*, 50(3), 809–820.