Efficient and QoS-aware Drone Coordination for Simultaneous Environment Coverage

Petra Mazdin
Karl Popper Kolleg
Alpen-Adria-Universität Klagenfurt, Austria
petra.mazdin@aau.at

Bernhard Rinner
Institute of Networked and Embedded Systems
Alpen-Adria-Universität Klagenfurt, Austria
bernhard.rinner@aau.at

Abstract

Simultaneous coverage represents an important problem for various drone applications where objects must be covered by simultaneously captured images from different viewpoints that may impose additional requirements on geometry, resolution and depth error. In order to solve simultaneous coverage the drones must plan their movements such that the required viewpoints are concurrently reached while minimizing the overall mission time and satisfying the quality requirements. In this paper, we introduce a market-based and QoS-aware coordination method for assigning drones to viewpoints during the mission. In our simulation study, we compare three algorithm variants based on the achieved mission time and the required communication effort.

Keywords—simultaneous coverage; quality of service; image quality, communication effort; task assignment

1. Introduction

Drones with onboard cameras are often used for surveillance, search-and-rescue, inspection and entertainment [8]. In these applications, the drones move in the environment, capture and pre-process multimedia data at a specified quality and transfer it to the base station. In this paper, we focus on *simultaneous coverage* where all visible surfaces of a given environment must be covered by simultaneously captured images from at least k different viewpoints [10]. Various quality of service (QoS) parameters such as target resolution, depth error and multi-drone geometry have an impact on number and positions of the required viewpoints. In order to solve simultaneous coverage (cp. Figure 1), we first need to compute all camera viewpoints (or constellations), i.e., the positions and orientations of k cameras satisfying the QoS parameters, and then to assign drones to constellations such that all constellations are visited while minimizing mission time and resource usage.

In our previous work [10] we introduced the concept of simultaneous coverage and presented a leader-follower path planner for a fixed team of two drones. In this paper we focus on how to dynamically coordinate larger teams of drones. The contribution includes (i) a method for computing the constellations based on QoS parameters, (ii) a market-based and decentralized assignment of drones to constellations and (iii) a simulation study comparing mission time and communication effort of three allocation variants. Our approach dynamically assigns the drones, plans collision-free paths along the constellations and is resource-efficient which is important for various multimedia applications.

Market-based coordination is a well studied approach for multi-robot systems [2, 12, 7, 4, 6]. Similarly, multi-camera constellations and corresponding quality parameters have been investigated in the computer vision community (e.g. [9, 3]). More recently, drone networks have been deployed in various applications including multimedia distribution and their communication requirements and design aspects have been studied (e.g. [1, 5, 11]). The key novelty of this work lies in the adoption and evaluation of market-based coordination for simultaneous coverage.

The paper is organized as follows: Section 2 briefly introduces simultaneous coverage. We present our methods for constellation computation, task assignment and path planning in Section 3 and discuss the simulation results in Section 4. Section 5 concludes the paper with a summary and outlook for future work.

2. Problem Definition

Figure 1 depicts the simultaneous coverage problem of partially unknown environments where the drones can take on different roles during the mission. Drones explore the environment for objects of interests. Once a drone has detected an object, it identifies the relevant surfaces, computes the required constellations for the coverage and initiates the

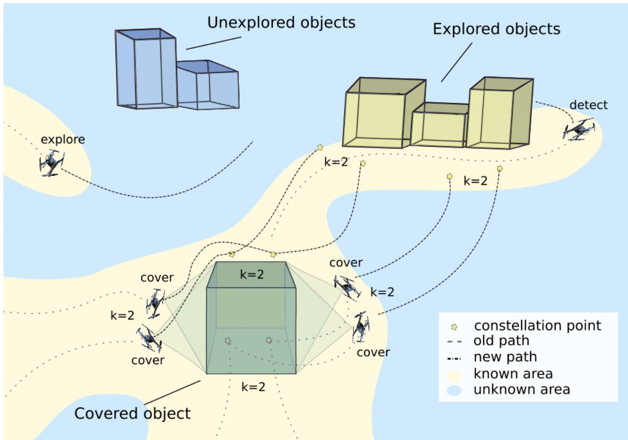

Figure 1. Simultaneous coverage of a partly unknown environment with multiple drones. Drones explore the environment for objects of interest, detect relevant surfaces and compute the required constellations as well as cover each surface simultaneously.

assignment of k drones for the simultaneous coverage.

A set of m drones $C = \{c_1, \dots, c_m\}$ explore and cover the environment. The environment is specified by a set of objects $O = \{o_1, \dots, o_p\}$ which can be known a priori or detected during the mission. Each object o_i is abstracted by a set of visible surfaces $S_i = \{s_{i1}, \dots, s_{ij}\}$ which results in the overall set of surfaces to be covered $S = \cup_{i=1}^{p} S_i$, with $|S| = q$. For each surface, a constellation of k drones is computed satisfying QoS parameters such as target resolution δ, depth error and multi-drone geometry. The QoS parameters have an effect on the number of constellations, the aggregated amount of data to be transferred and consequently on the overall mission time. In order to solve simultaneous coverage, $k \leq m$ drones must be assigned to visit each constellation and simultaneously capture images of the corresponding surface.

The key objective is to minimize the overall mission time while satisfying all QoS parameters. The key steps for solving this problem include the computation of the corresponding constellations for the surfaces that can be covered by individual drones, the assignment of drones to constellations and the planning of the paths between constellations.

3. Methods

In the following, we present details of the three key steps of our approach. For simplicity, we assume objects with constant height and a common ground plane and can therefore transform the coverage problem to 2D environments where each surface can be represented as a line segment. Furthermore, k is fixed to 2 which represents the special case of stereo coverage.

3.1. Constellation Computation

Constellation computation determines camera poses satisfying the QoS parameter target resolution δ and various camera parameters. We adopt a simple 1D camera model [10] to estimate δ as a function of the sensor resolution r, the lens aperture angle α and the distance to the surface d:

$$\delta = \frac{r}{2 \cdot d \cdot tan(\frac{\alpha}{2})}. \tag{1}$$

It is obvious that increasing δ results in a larger number of constellations since the camera (with fixed r) must cover a smaller part of the surface (either by zooming in or lowering the distance). If we assume a 50 % overlap and a baseline b between two cameras, the maximum length l_{max} of a line segment for a two-camera constellation can be estimated as $l_{max} = b = d \cdot tan(\alpha/2)$ [10]. Thus, larger surfaces must be partitioned into segments with a maximum length l_{max}.

An alternative is to cover larger surfaces with more than two drones which is depicted in Figure 2 where n drones are aligned in a constellation and the neighboring drones simultaneously cover different segments of the entire surface of length L. For this advanced constellation, the drones must be placed such that their field of views can cover three line segments of length l_n (as compared to two of length l_k for simple constellations). The relation between the number of partitions u_k for the simple constellation and the advanced constellation u_n is given as

$$\frac{u_k}{u_n} = \frac{l_n}{l_k} = \frac{2}{3}, \tag{2}$$

which means that the advanced constellation requires more partitions than the simple constellation if $l_{max} < L$. Given the total length L, δ and r, we can compute the number of drones needed to cover the whole surface simultaneously as

$$n = \frac{3 \cdot L \cdot \delta}{r} + 1. \tag{3}$$

Advanced constellations impose position and orientation constraints such as linear alignment and constant distances among neighbors for n drones. However, the surface of length L can be instantaneously covered with n constellation points as compared to $\frac{2 \cdot k \cdot (n-1)}{3}$ for simple constellations.

For the computation of the constellation points (i.e., the camera poses) we adopt our previous approach [10] which minimizes the relative angle and the depth error. The result of this optimization are constellation points with the baseline parallel to the surface and cameras perpendicularly orientated to it.

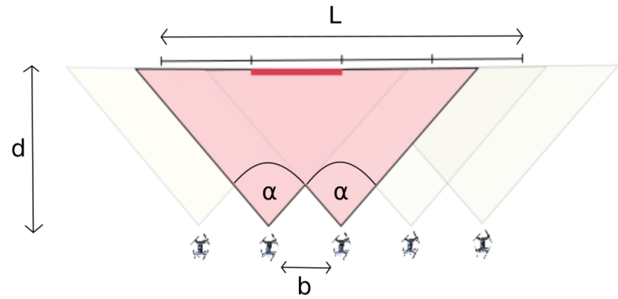

Figure 2. Advanced constellation for a surface of length $L > l_{max}$ with n drones.

3.2. Task Assignment

As constellations may be computed during the mission, they must be dynamically assigned to available drones in an efficient way. We deploy a market-based approach for the assignment of drones to constellations due to their decentralized approach and low computational effort. Figure 3 depicts the flow chart and communication of an auctioneer drone and a bidder drone. Basically, the auctioneer initiates the assignment process by broadcasting an auction message to all idle drones which choose a specific point and respond with a bid whose value is reciprocal to the flight distance from the current position. After a predefined waiting time, the auctioneer determines the winning bidders and sends out the corresponding auction messages to all bidders. The winner drones plan and move along the paths to the constellation points, potentially waiting for other drones, and capture the image. We distinguish between object-based and surface-based auction approaches.

Object-based auctioning In this variant, the auctioneer assigns k drones to object o_i and thus the same k drones cover all of the object's surfaces S_i. The auctioneer includes all constellations of o_i in the auction initiation message. The bidder drones compute their minimal aggregate flight distance to the closest constellation point and bid with that value. The flight distance is based on a collision-free path planning and serves as input for the bid valuation. The auctioneer then has to find k bids whose sum, plus the flight distance along the collision-free path going through all k closest constellation points, is minimal. The winning k drones select the starting constellation as the point which has the lowest accumulated distance from their current positions. The drones cover then all constellations of o_i along the shortest path. We label object-based auctioning as OB.

Surface-based auctioning In this variant, the auctioneer initiates an auction for each surface independently. The bidder drones respond with their flight distance from their current location to the closest constellation point. This variant also supports simple and advanced constellations composed of k and n constellation points, respectively. Thus, the auction mechanism only differs in the number of constellation points and required bids. We label the variants of surface-based auctioning as k-SB and n-SB, respectively.

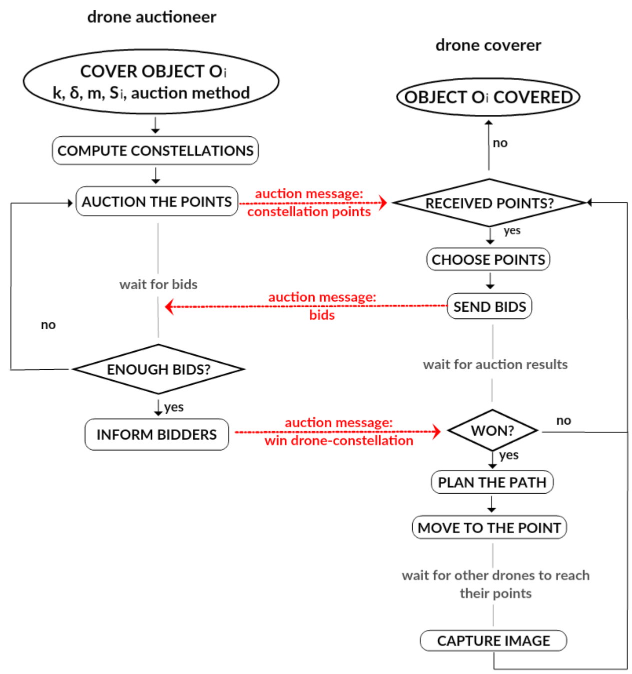

Figure 3. Flow chart for auctioning constellations among drones where the detecting drone starts auctioning (left) and idle drones (right) compete for a constellation.

3.3. Path Planning

For our problem, path planning is concerned with the movement of drones from their current position to constellation points avoiding collisions with objects and other drones. We use a simple heuristic based on linear movement segments because such movements naturally match the waypoint navigation of drones and can be computed in a resource-efficient way.

In order to identify a path from one constellation point to another, we check for intersections of the linear path with objects. Since objects are abstracted by line segments we can focus on the corner points of the objects, i.e., we check whether all corner points of an object lie on the same side of the linear path. If this condition is not true, the linear path causes a collision with the object and a bypass point is introduced. Thus, we insert a new waypoint at a given minimum distance from the "colliding" corner point and compute the new path via this bypass point. A direction of the bypass point is important as it affects the accumulated distance along the new path. Potentially multiple bypass points must be inserted if there are more "colliding" corner points. In order to avoid collisions among drones, a given minimum distance from the corner is increased for each additional drone in a constellation by the minimum allowed distance between drones.

4. Results

We evaluate our drone coordination methods in a simulation study where we measure the mission time and communication effort with varying number of drones and target resolution. We use two scenarios as input for our experiments: Scenario A is composed of 3 objects placed on an environment of 200 $m \times 200$ m, and scenario B is composed of 10 objects placed on a 800 $m \times 800$ m environment, respectively. The objects are randomly placed, have a convex shape and are composed of 3 or 4 line segments. Drones start from random initial positions in the environment.

For each experiment, we vary the number of drones ($m \in 4, 6, 8, 10, 12$) and the target resolution δ from 1 px/m to 251 px/m with a step of 10. We measure the overall mission time and the communication effort in terms of transferred auction messages, and aggregated amount of location messages and image data. Each experiment is performed 100 times and average values are depicted in the performance graphs.

The following parameters have been set in our simulation environment: The auctioneer waiting time for bids is set to 1 s. If an insufficient number of bids is received, a reauctioning is initiated. Drone speed is set to 14 m/s. Cover time is fixed to 3 s and accounts for deceleration and stabilization of the drone at the new constellation point. The camera parameters are set as: $f = 4.7$ mm, $\alpha = 60°$ and $r = 1920$ px (full HD sensor). Both minimum distance between drones and a distance to objects is set to 5 m. The image size is given as 6.5 MB and derived from the HD sensor resolution. Auction and location messages are estimated as 100 B. We perform experiments for each auction method, and test n-SB with two settings for the maximum size of advanced constellations: $n = 0.5 \cdot m$ and $n = m$, respectively. OB and k-SB assume a fixed $k = 2$ for all objects' surfaces.

4.1. Mission time

Figure 4 (left) depicts the overall mission time as a function of δ and m for both scenarios. In general, the mission time increases with δ due to the larger number of constellations. The three different auction methods result in different mission times, in particular for a small m. For OB auctions a fixed team of k drones is assigned for each object. If drones cannot be perfectly assigned to objects, the remaining drones need to wait until the assigned drones become available again. In such cases (e.g. in scenario A with 4 drones), the mission time of OB is significantly larger due to the waiting time of some drones.[1] For the k-SB method, a surface has to wait to be covered until the fastest k drones complete the coverage of the other surface. However, this waiting time is shorter than the OB waiting time.

Since n-SB aims to cover large surfaces with advanced constellations of size n, there is a limit on the *achievable* target resolution δ_{max} for a given length L and n. If the specified δ is larger than δ_{max}, then n-SB computes advanced constellations with δ_{max} and covers the large surfaces with that target resolution. For that reason, the graph for $m = 4$ remains almost constant.

The performance of OB improves wrt. the other approaches with increasing m. There is a threshold at $m = p \cdot k$ because this is the maximum number of drones which can simultaneously cover p objects. Increasing the number of drones further does not reduce the mission time of OB.

4.2. Communication effort

Data communication among drones and the base station can be grouped into (i) auction messages (i.e., items announced by an auctioneer, drone bids and auction results), (ii) location messages (i.e., current and future waypoints among the assigned drones), and (iii) imagery of the covered surfaces transferred to the base station.

Figure 4 (center) compares the different auction mechanisms in terms of exchanged auction messages. The graphs clearly indicate two settings with long waiting times and hence a high number of auction (re-)initiations. First, OB requires many re-initiations for a low m because too few drones are available to cover an object and they must wait for other drones. Second, n-SB requires many re-initations for a larger m because large surfaces may require a large m for advanced constellations and the remaining drones may need to wait until the large surface has been covered.

Figure 4 (right) plots the overall amount of location messages among the m drones. These messages are exchanged between k or n drones when being assigned to the same constellation. They exchange their current location with the waypoint they are heading to. The number of messages depends only on δ for OB and k-SB. For n-SB the number of messages increases with m because there is the possibility of increasing the achievable resolution δ_{max} and con-

[1] For the same reason, the mission time and the number of auctions of OB for $m = 4$ in scenario B is approximately three times longer than for $m = 6$. We have removed these graphs in the chart's bottom left and bottom center to avoid rescaling of the vertical axis and thus maintain readability.

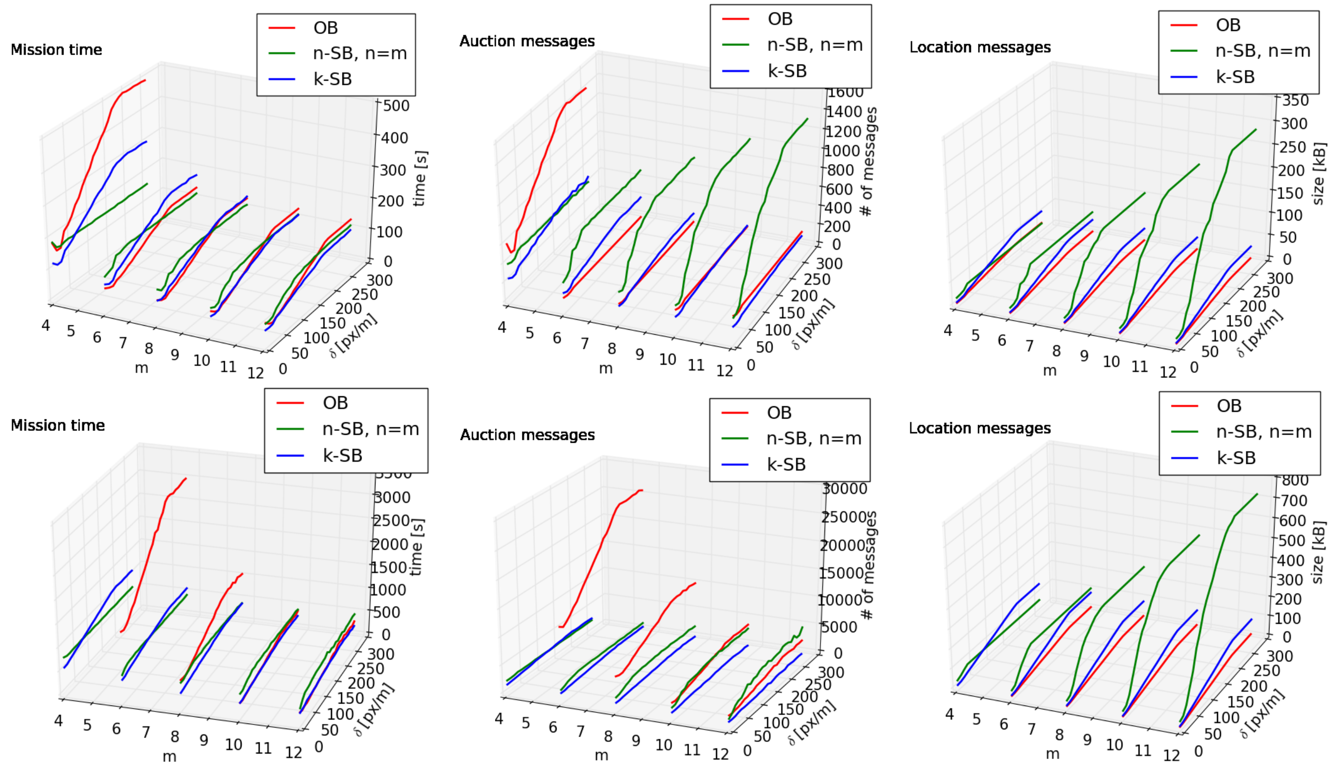

Figure 4. Results for scenario A (top) and scenario B (bottom).

sequentially the number of constellation points. There is a fast increase for advanced constellations, and it surpasses the simple one due to the fact that advanced constellations require more partitions of the surface to satisfy δ than the simple constellations as shown in Equ. (2). As each partition has to be covered by 2 drones in both variants, and these drones need to exchange their locations when heading to cover that partition, the number of location messages is larger for advanced constellations.

Naturally, the transfer of the captured images has the strongest impact on the communication load. Figure 5 depicts the overall amount of image data captured by all drones and transferred to the base station. The overall amount depends on the number of constellation points which is determined by δ for OB and k-SB. For n-SB the overall amount may be lower due to the lower achievable target resolution δ_{max} for a given n. If $\delta \leq \delta_{max}$ for the n-SB method, a larger amount of data is transferred, in comparison to k-SB, due to the same reason as for the location messages: the more partitions of a surface to cover, the more images to exchange.

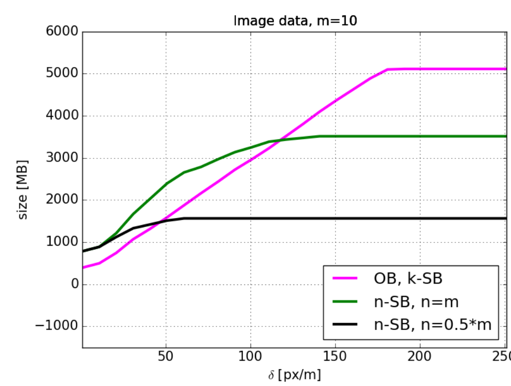

Figure 5. Overall amount of image data for scenario B.

4.3. Discussion

As our experiments have shown, the performance of our auction methods significantly depends on the number of drones. For low m, k-SB outperforms OB concerning mission time and auction messages. For $m \approx p \cdot k$, OB outperforms k-SB because waiting time is avoided and auctioning entire objects is more efficient than individual surfaces. Moreover, mission time and number of transferred messages increase with larger δ due to increasing number of constellations. Note that image data clearly dominates the data transfer requirements since the data amount is orders of magnitudes larger than for the other data types.

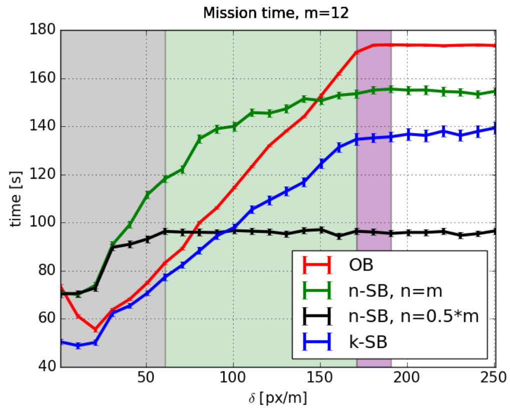

Figure 6. Average mission time and 97.5% confidence intervals for OB, k-SB and two variants of n-SB for scenario A. The colored background indicates the limits of the achievable target resolution δ_{max}.

Figure 6 plots the average mission time of OB, k-SB and two variants of n-SB for different target resolution δ for scenario A with 12 drones. The vertical bars depict the 97.5% confidence interval of the mission time for the specified δ. The effect of n on the achievable δ_{max} is represented by the different background colors, i.e., 61 px/m for n-SB with n = 6, 171 px/m for n-SB with n = 12, and 191 px/m for OB and k-SB.

There is a tradeoff between mission time, achievable target resolution and required computation and communication effort. A hybrid approach could exploit this tradeoff by changing the auction method based on the QoS parameters and mission requirements.

5. Conclusion

We have presented a market-based and QoS-aware coordination method for simultaneous coverage where drones plan their movements such that the overall mission time is minimized and quality requirements are satisfied. Our approach computes the drone constellations for each surface and dynamically assigns drones to constellations by a simple decentralized auction mechanism. Drones move along collision-free paths from constellation to constellation to simultaneously cover all objects. In a simulation study we have compared three coordination methods based on the achieved mission time, the required communication effort and the achieved target resolution.

As future work, we will (i) investigate in hybrid methods able to adapt the auction mechanism, (ii) deploy the algorithms on our drones and (iii) demonstrate simultaneous coverage in an autonomous 3D reconstruction case study.

References

[1] T. Andre, K. A. Hummel, A. P. Schoellig, E. Yanmaz, M. Asadpour, C. Bettstetter, P. Grippa, H. Hellwagner, S. Sand, and S. Zhang. Application-Driven Design of Aerial Communication Networks. *IEEE Communications Magazine*, 52(5):129–137, 2014.

[2] M. B. Dias, R. Zlot, N. Kalra, and A. Stentz. Market-Based Multirobot Coordination: A Survey and Analysis. *Proc. IEEE*, 94(7):1257–1270, 2006.

[3] D. Gallup, J.-M. Frahm, P. Mordohai, and M. Pollefeys. Variable Baseline/Resolution Stereo. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[4] B. P. Gerkey and M. J. Mataric. Sold!: Auction Methods for Multirobot Coordination. *IEEE Transactions on Robotics and Automation*, 18(5):758–768, 2002.

[5] S. Hayat, E. Yanmaz, and R. Muzaffar. Survey on Unmanned Aerial Vehicle Networks for Civil Applications: A Communications Viewpoint. *IEEE Communications Surveys&Tutorials*, 18(4):2624–2661, 2016.

[6] J. Jackson, M. Faied, P. Kabamba, and A. Girard. Communication-Constrained Distributed Task Assignment. In *Proc. 50th IEEE Conference on Decision and Control and European Control Conference*, pages 570–577, 2011.

[7] B. Kalecı, O. Parlaktuna, M. Özkan, and G. Kirlik. Market-Based Task Allocation by using Assignment Problem. In *Proc. IEEE International Conference on Systems Man and Cybernetics*, pages 135–141, 2010.

[8] A. Khan, B. Rinner, and A. Cavallaro. Cooperative Robots to Observe Moving Targets: A Review. *IEEE Transactions on Cybernetics*, 48(1):187–198, 2018.

[9] M. Okutomi and T. Kanade. A Multiple-Baseline Stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353–363, 1993.

[10] M. Weyrer and B. Rinner. UAV Motion Planning and Control for Multi-Coverage of 3D Environments. In *Proc. International Conference on Unmanned Aircraft Systems*, pages 939–946, 2018.

[11] E. Yanmaz, S. Yahyanejad, B. Rinner, H. Hellwagner, and C. Bettstetter. Drone networks: Communications, coordination, and sensing. *Ad Hoc Networks*, 68:1–15, 2018.

[12] R. Zlot and A. Stentz. Market-based Multirobot Coordination for Complex Tasks. *The International Journal of Robotics Research*, 25(1):73–101, 2006.