

Cooperative Aerial-Ground Multi-Robot System for Automated Construction Tasks

Marko Krizmancic, Barbara Arbanas, Tamara Petrovic, Frano Petric and Stjepan Bogdan

Abstract—In this paper, we study a cooperative aerial-ground robotic team and its application to the task of automated construction. We propose a solution for planning and coordinating the mission of constructing a wall with a predefined structure for a heterogeneous system consisting of one mobile robot and up to three unmanned aerial vehicles. The wall consists of bricks of various weights and sizes, some of which need to be transported using multiple robots simultaneously. To that end, we use hierarchical task representation to specify interrelationships between mission subtasks and employ effective scheduling and coordination mechanism, inspired by Generalized Partial Global Planning. We evaluate the performance of the method under different optimization criteria and validate the solution in the realistic Gazebo simulation environment.

I. INTRODUCTION

In the field of robotics, the focus of the research is shifting increasingly from the design and control of single robots towards collaborative methods for multi-robot systems. Especially interesting are heterogeneous systems since the diverse capabilities of robots in such systems foster better performance, broader space coverage, improved energy utilization, and better knowledge through data fusion. This is particularly true for aerial-ground systems due to their distinct skillsets. For instance, unmanned aerial vehicles (UAVs) may benefit from ground vehicle's extended battery life while unmanned ground vehicles (UGVs) could take advantage of far-ranging mobility of aerial robots [1], [2].

The majority of the works related to UAV-UGV teams consider cooperation in terms of: collaborative sensing, data fusion and information sharing between UAV and UGV, such as collaborative mapping of the environment [3], [4], reconnaissance and surveillance [5], [6], multi-robot localization [7], collaborative object manipulation [8], target tracking [9], as well as search and rescue [10].

Controlling a heterogeneous team of robots requires precise high-level task planning and fast and robust coordination mechanisms. Given the nature of the task and environment at hand, a decentralized approach to a multi-robot coordination problem is bound to give the best performance concerning robustness and resilience to potential robot fall-outs. Some of the most prominent solutions to the problem mentioned

above are an auction and market-based approaches [11], [12]. These mostly fit into the task allocation category, where a set of simple tasks is given, and robots use bidding mechanisms to distribute tasks amongst themselves. Application of bidding mechanisms for more complex missions, in an uncertain environment with heterogeneous agents, is given in [13], however partial ordering between tasks and tight coupling, which are the basis of our cooperative missions, are not well supported.

Probabilistic multi-robot coordination approaches based on usage of decentralized partially observable Markov decision processes have been studied as well [14], [15]. The advantage of this approach is its inherent suitability to uncertain environments; however, the scalability problem is making them unsuitable for real-world applications that include several robots and complex tasks. Although some online solutions have been reported ([16]), the problems this approach manages to cope with are still relatively simple from the reasoning perspective.

Many of recent state-of-the-art approaches rely on off-the-shelf automated reasoners based on, for example, Linear Temporal Logic (LTL) [17], [18]. Even though these approaches show significant contributions to the theoretical synthesis of correct-by-design controllers, they often suffer from intensive computational problems, as well as the inability to quantify planner objectives. In [19], the authors devised a method for the automatic synthesis of controllers for a team of robots working in a dynamic environment. The advantage of this approach is the ability to cope with the dynamic constraints of the robot and guarantees collision avoidance in 2D and 3D workspaces. The method's performance is evaluated on a simple task of garbage collection using humanoid robots. However, LTL-based approaches remain limited by the expressivity of the logic, and it is challenging to specify complex task relations.

The approach presented herein entails dynamic high-level task planning (decomposition and allocation), which appropriates TÆMS (Task Analysis, Environment Modeling, and Simulation) language [20], [21] for task representation. The advantage of TÆMS task representation is the ability to easily specify complex task interrelations (both tight and loose) and to quantify the potential solutions. It, however, requires a planner and coordinator able to exploit all the benefits of such an expressive task model. We devised a system able to do just that [2], based on general guidelines on the design of the communication and coordination protocol of the Generalized Partial Global Planning (GPGP) coordination framework [22]. In our previous work, we applied

Manuscript received: September 10, 2019; Revised: December 2, 2019; Accepted: December 29, 2019

This paper was recommended for publication by Editor Nak Y. Chong upon evaluation of the Associate Editor and Reviewers' comments.

Authors are with University of Zagreb, Faculty of Electrical Engineering and Computing, Unska 3, 10000 Zagreb, Croatia (marko.krizmancic, barbara.arbanas, tamara.petrovic, frano.petric, stjepan.bogdan) at fer.hr

the described method to different multi-robot missions, such as disaster response [1] and autonomous parcel delivery [2], [23].

Contributions of this work are following. Firstly, we extend the abilities of the framework regarding task allocation by implementing a market-based protocol, for the class of problems ST-MR-TA-XD (single-task robots, multi-robot tasks, time-extended assignment, cross-schedule dependencies) [24]. Consequently, we can cope with complex scheduling problems in real-time, without drawbacks on the optimality of obtained solutions. Further, the protocol also involves temporal and precedence constraints, which is not the usual case in the state-of-the-art literature. We apply the method to the problem of automated construction, which perfectly highlights the ability of the planner to cope with combinatorial optimization problem of heterogeneous multi-robot systems in complex missions. The main advantage of our approach is the ability to clearly and effectively quantify goals of multi-criteria optimization function, as demonstrated in the results section. Moreover, to the best of our knowledge, none of the state-of-the-art approaches provide a complete solution to the described decentralized problem, but rather focus only on one aspect of the problem, mainly (centralized) task allocation [17], [25]. Lastly, since the planner employs a mission-agnostic approach, this solution can be applied to any domain with minimal alterations, as long as task representation adheres to the required hierarchical structure.

The paper is structured as follows. In Section II, we describe the problem in detail. Section III entails the specifics of the used method, while Section IV shows simulation results using the described system. Finally, in Section V, we give a brief conclusion and comments about future work.

II. PROBLEM DESCRIPTION

The *Mohamed Bin Zayed International Robotics Challenge* (MBZIRC)¹ aims to encourage innovation and highest quality research in emerging topics by providing a demanding set of robotics challenges that require robots working more autonomously in dynamic, unstructured environments while collaborating and interacting with each other. The technological challenges addressed in the MBZIRC 2020 Challenges include fast autonomous navigation in semi-structured, complex, dynamic environments, with reduced visibility (e.g. smoke) and minimal prior knowledge, robust perception and tracking dynamic objects in 3D, sensing and avoiding obstacles, GPS denied navigation in indoor-outdoor environments, physical interactions, complex mobile manipulations, and air-surface collaboration [26]. In this paper, we tackle the problem of automated construction where a team of aerial and ground robots needs to collaborate to autonomously locate, pick, transport and assemble various types of brick-like objects to build predefined structures. Particularly, we are interested in the decision making and coordination process which optimizes wall construction task concerning given multi-objective criteria.

The challenge specification is given as follows. Red, green, blue, and orange bricks ranging from 0.3m to 1.8m in length are separated by color and assembled in randomly located piles before the start of the challenge. Blue bricks may be collected only by the UAVs, while orange bricks need to be carried by two or more UAVs at the same time due to their size and weight. Either UAVs or the UGV may assemble red and green bricks, but those constructed by the UAVs will gain higher marks. The form of the wall structure is the only input to the system. Teams have 30 minutes to complete the challenge and score is determined by the percentage and speed of completion. Exact parameters of the challenge and brick properties are further specified in [26].

This challenge has been specifically designed to emphasize the need for collaboration between different types of agents. UAVs are agile, have a virtually unlimited reach and are superior in number, which means they can construct more in less time. However, due to their limited battery life, it is imperative to also utilize the UGV for low-scoring bricks and those closer to the ground.

The focus of this paper is the implementation of an algorithm for decentralized high-level mission planning and coordination which will supervise the task of construction. All available agents should be utilized to their full potential and carry out assigned tasks in parallel to maximize the challenge score and complete the task within the specified time.

III. METHOD

A. Model description

We decompose the wall building mission using TÆMS hierarchical task structure ([1], [21]) in a way that the immediate subtasks of root task represent transportation and assembly of individual bricks. The TÆMS tree is defined for three types of agents, "UGV", "UAV", and "UAVx2", where the last label denotes an agent, comprised of two UAVs that are capable of collaborating in the transport of the large, orange brick. Leafs of the tree represent *action nodes* that correspond to real, actionable agent behaviours. Agent behaviours used in presented applications are $GP(b_i)$ - go to the pile and locate the brick b_i , $PU(b_i)$ - pick up the brick from the pile, $GW(b_i)$ - go to the wall, and $PD(b_i)$ - place the brick in its specified location within the wall, where i signifies brick identifier. Other nodes are *task nodes* that combine action nodes into logical units. A task node that combines all actions necessary to transport a single brick b_i is denoted with $TB(b_i)$.

Four types of relations between nodes define how quality, duration, and cost of children nodes affect the parent node - q_sum_all that corresponds to logical AND, q_max that represents the XOR operator, $q_seq_sum_all$ that is AND with strictly sequential execution, and q_sum which signifies that a subset of subtasks renders parent task as finished. In TÆMS language, these relations are called *quality accumulation functions* or *QAF* for short. *Quality* of each task represents the number of points received for successfully assembling the individual brick. *Duration* of a task is estimated using the

¹<http://mbzirc.com>

length of the desired path and average moving speed of the agent. Additionally, for actions $PU(b_i)$ and $PD(b_i)$ a fixed estimated duration of grabbing and releasing behaviours is added. Finally, we determine the *cost* by multiplying duration with the per-time cost based on the complexity of the task, type of the agent performing it, and the size of the associated brick.

If a task cannot be completed due to kinematic constraints of the robot (e.g. desired placement of the brick is unreachable by the UGV's robotic arm), quality of such task is set to zero, while duration and cost are set to large positive number in order to exclude it from scheduling.

Relations between other action and task nodes are modelled using interrelationship TÆMS elements. *Enables* relations demand that one task finishes before another is started and are used to specify the order of bricks within the wall structure. The action of picking up one brick ($PU(b_i)$) enables the action of going to the pile to pick up another brick ($GP(b_j)$).

This introduces an offset in the transportation of neighboring bricks and helps to avoid situations where multiple agents are placing bricks at the same time in close proximity. The offset is introduced at the beginning of tasks because it is more efficient for the UAVs to wait on the ground than in the air while carrying a brick.

Disables relations are used to further prioritize the order of the brick placements in order to ensure the feasibility of the schedules.

B. Resolving simple redundancy

One of the most essential steps in the GPGP coordination procedure is a resolution of simple redundancy between tasks. Redundancy is a situation where one or more agents in the set denoted with A_{sr} may execute a single task. Resolution procedure starts by selecting one of the agents as a referee. The selected agent then collects task assessments from all other agents, chooses the best one and communicates the result with others. Agents that are not selected as the best discard the task from their schedules.

The first part of the criteria function for selecting the best agent to execute a redundant task j is defined as follows:

$$r_{Q,i}(j) = \frac{Q_i(j) - Q_{min}(j)}{Q_{max}(j) - Q_{min}(j)}, \quad (1)$$

$$r_{D,i}(j) = \frac{D_{max}(j) - D_i(j)}{D_{max}(j) - D_{min}(j)}, \quad (2)$$

$$r_{C,i}(j) = \frac{C_{max}(j) - C_i(j)}{C_{max}(j) - C_{min}(j)}, \quad (3)$$

$$R_i(j) = \alpha \cdot r_{Q,i}(j) + \beta \cdot r_{D,i}(j) + \gamma \cdot r_{C,i}(j), \quad (4)$$

where $Q_i(j)$ is the quality of task j assessed by the agent $i \in A_{sr}$, $Q_{min}(j) = \min_{i \in A_{sr}}(Q_i(j))$ and $Q_{max}(j) = \max_{i \in A_{sr}}(Q_i(j))$. Analogous definitions are given for $D_i(j)$, $D_{min}(j)$ and $D_{max}(j)$ which represent the duration assessment, and $C_i(j)$, $C_{min}(j)$ and $C_{max}(j)$ for the cost. Parameters α , β and γ are user-defined positive real constants, $\alpha + \beta + \gamma = 1$, that give different importance to various criteria.

The problem arises when redundant tasks or agents are very similar. In this case, task assessments are almost identical, but usually, one agent has a marginally better outcome and is selected to execute all redundant tasks. Other agents are therefore not utilized, and tasks cannot execute in a parallel manner. In order to better allocate the redundant tasks, we utilize additional market-based criteria function, as described in the following paragraph.

Based on initial duration assessments and interrelationships between tasks, a simple market-based task allocation function creates an allocation scheme in a way which aims to minimize the total mission duration. The result of the function is a list S of pairs in the form of (*task*, *agent assigned to the task*). Such allocation is a preferred solution for the redundancy (r_2) but task assessments of each agent (r_1) are still used to cover the cases where some agent has significantly better execution outcome. The total rating of the agent i for the given task j is:

$$r_{1,i}(j) = \frac{R_i(j) - R_{min}(j)}{R_{max}(j) - R_{min}(j)}, \quad (5)$$

$$r_{2,i}(j) = \begin{cases} 1, & \text{if } (j, i) \in S, \\ 0, & \text{otherwise,} \end{cases} \quad (6)$$

$$R_{total,i}(j) = \delta \cdot r_{1,i}(j) + (1 - \delta) \cdot r_{2,i}(j), \quad (7)$$

where $R_i(j)$ is a rating of the task j based on quality, duration, and cost assessment of the agent $i \in A_{sr}$, as defined in (4), $R_{min}(j) = \min_{i \in A_{sr}}(R_i(j))$, $R_{max}(j) = \max_{i \in A_{sr}}(R_i(j))$, and δ is a user-defined positive constant, $\delta \in [0.5, 1)$, that gives different importance to the two parts of the criteria function. Note that for $\delta < 0.5$, rating r_1 would not affect the total rating because $r_{1,i}(j) \leq r_{2,i}(j)$, $\forall i \in A_{sr}$ and $\sum_{i \in A_{sr}} r_{2,i}(j) = 1$. In other words, for each task j , there would be only one agent for which $r_{2,i} = 1$ and that agent would have better total rating than all others, regardless of the value of r_1 .

C. Resolving complex redundancy

Complex redundancy is an extension to the TÆMS model with the addition of a so-called local *QAF*. It represents the situation where a task has multiple subtasks that need to be executed simultaneously and therefore, it is necessary to have a one-on-one mapping between subtasks and agents. Such a situation occurs in the case of the task of joint transportation of orange brick. Two agents must schedule their part of the task at the same time and execute it together. Globally, for the task to be executed, both subtasks must be performed, while locally to each agent, only one of those tasks is to be scheduled. Therefore, tasks are modeled as complexly redundant by defining the *QAF* of their parent task as q_sum or q_sum_all and QAF_{local} as q_max . We use the term redundancy since the effect is similar to one of the simply redundant tasks, but its resolution needs to be managed differently. Instead of multiple agents competing for a single task, as is the case during the resolution of simple redundancy, m agents compete for n tasks, $m \geq n$, such that

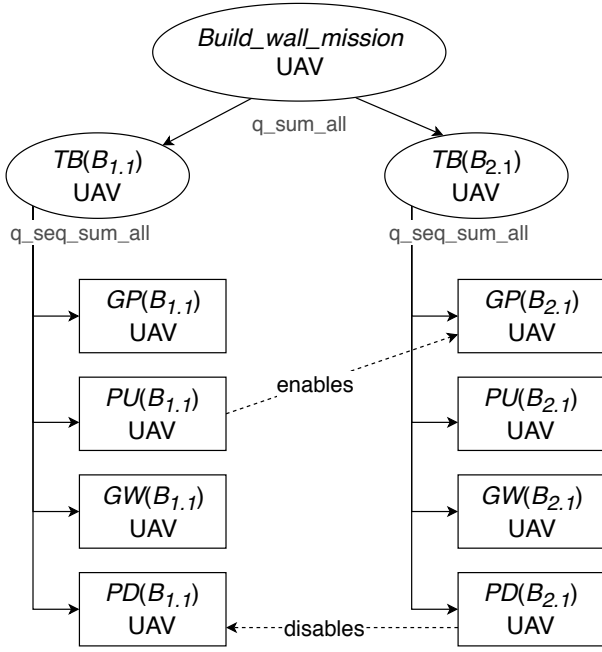


Fig. 1. Task structure of the example mission

each agent schedules only one subtask and that all subtasks are assigned.

The problem of assigning only one agent to each task in a way that optimizes total quality, duration and cost is modelled as a generalized assignment problem [27]:

$$\max \sum_{i \in A_{cr}} \sum_{j \in T_{cr}} R_i(j) \cdot x_{ij}, \quad (8)$$

$$\text{subject to} \sum_{i \in A_{cr}} x_{ij} = 1, \quad \forall j \in T_{cr}, \quad (9)$$

$$\sum_{j \in T_{cr}} x_{ij} = 1, \quad \forall i \in A_{cr}, \quad (10)$$

where A_{cr} is a set of agents that are able to schedule a complexly redundant task, T_{cr} is a set of subtasks of the complexly redundant task, $R_i(j)$ is the rating of the subtask j based on assessment of the agent i defined in equation (4) and $x_{ij} \in \{0, 1\}$ is a binary decision variable for the assignment of agent i to the subtask j . Equations (9) and (10) ensure that each subtask is assigned to only one agent and that each agent performs only one subtask respectively. Described optimization problem is solved using the custom implementation of the branch and bound algorithm [28].

D. Resources

Each set of action nodes connects to its parent task with a $q_seq_sum_all$ function, meaning that they must carry out in a strictly specified order. However, this does not prevent an agent from scheduling and executing other tasks in the meantime.

Consider a system with only one UAV and two blue bricks that need to be stacked on top of each other. TÆMS tree structure for such mission is shown in Figure 1. The bottom brick is labeled as $B_{1,1}$, while the top one is labeled

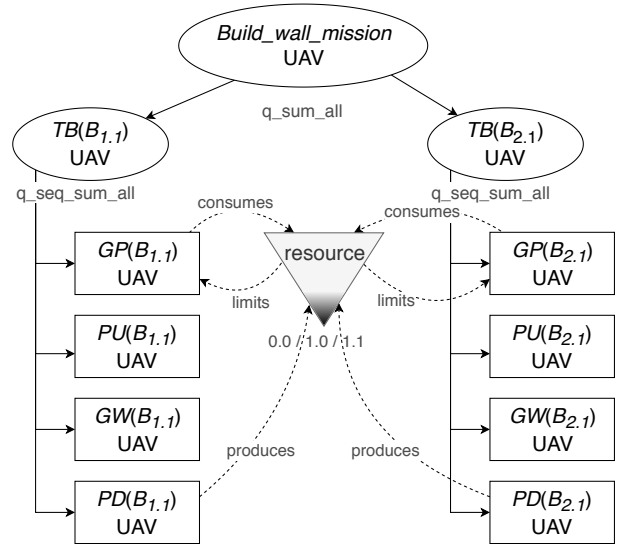


Fig. 2. Task structure of the example mission extended with resources. Enables relations are omitted for clarity.

$B_{2,1}$. Expected execution schedule for specified mission is as follows:

$$GP(B_{1,1}), PU(B_{1,1}), GW(B_{1,1}), PD(B_{1,1}), \\ GP(B_{2,1}), PU(B_{2,1}), GW(B_{2,1}), PD(B_{2,1}).$$

However, successful execution of action $PU(B_{1,1})$ enables action $GP(B_{2,1})$ and final execution schedule is in fact:

$$GP(B_{1,1}), PU(B_{1,1}), GP(B_{2,1}), PU(B_{2,1}), \\ GW(B_{1,1}), GW(B_{2,1}), PD(B_{1,1}), PD(B_{2,1}).$$

The scheduling algorithm is intentionally unaware of the agents' specific abilities, so it suits various applications. The final schedule is the best solution for the given problem because it minimizes the total mission duration, even though UAVs cannot carry multiple bricks at the same time. In order to model such limitation, TÆMS virtual resources are used in the extended model of the mission. Each agent entails a resource, which represents a slot for carrying a brick. The initial state of the resource is 1, while its lower and upper limits are 0 and 1.1 respectively. Every time an agent executes $GP(b_i)$ type of action, one unit of resource is consumed, making it insufficient. Insufficient resources automatically disable all other $GP(b_j)$, ($i \neq j$) actions for that agent until their state is restored by executing the $PD(b_i)$ action. The improved model of TÆMS tree structure is shown in Figure 2.

IV. SIMULATION RESULTS

A. Testbed description

We devised the environment for the second challenge of MBZIRC 2020 in the *Gazebo* simulator (Figure 3). Using its *Robot Operating System* (ROS) interface, we can realistically simulate planning, coordination, and execution aspects of the proposed solution for wall building mission.

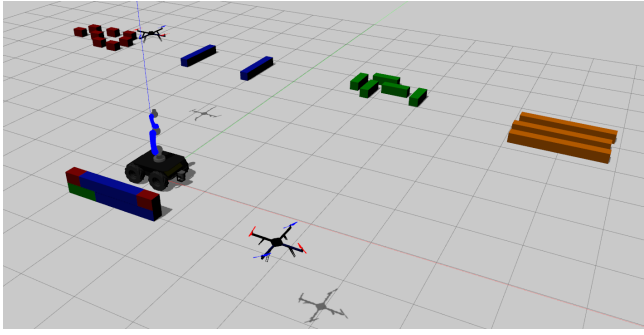


Fig. 3. Simulation environment with robots

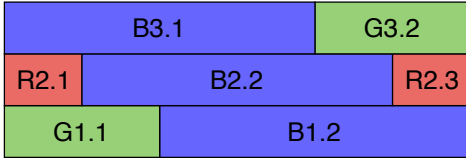


Fig. 4. Specified wall structure for the first mission

We simulate physically accurate models of multicopters for UAV agents and a Gazebo model of Husky by Clearpath Robotics with mounted Schunk Powerball LWA 4P robotic arm for the UGV agent. Both models have standard sensors like inertial measurement unit and a generic pose sensor which provide true position and orientation of the robots. UAVs use a simple position PID controller for movement, while UGV uses a navigation algorithm provided by *move_base* ROS package. Mid-air collisions are prevented by commanding a different predefined flying height to each UAV.

Since the focus of this work is on planning and coordination, robots do not have end effectors for interaction with the bricks onboard. In the simulation, bricks are teleported using Gazebo's *spawn* and *delete* services.

Locations of all four brick piles and wall origin are known before the start of the mission. In the real world, challengers should acquire this information by scouting the arena with UAVs at the start of the trial.

B. Simulation results

In order to test the system described in this paper, we employ the team consisting of two UAVs and a UGV on a task specified as in Figure 4. We conducted the simulated mission with a series of different criteria setups, three of which we analyze here. The costs of execution per time unit are kept constant for UAVs and UGV in all of the scenarios.

In the first scenario, we define the total quality of the mission as the most weighty component of the planning criteria while other components play a less significant role, $\alpha = 0.5, \beta = 0.35, \gamma = 0.15$. As expected, only UAVs participate in the construction because bricks assembled in that way carry more points. Our task allocation method assigns tasks to both UAVs alternately in order to achieve parallel execution and shorten the total mission duration. Time diagram of tasks executed by each agent is shown in

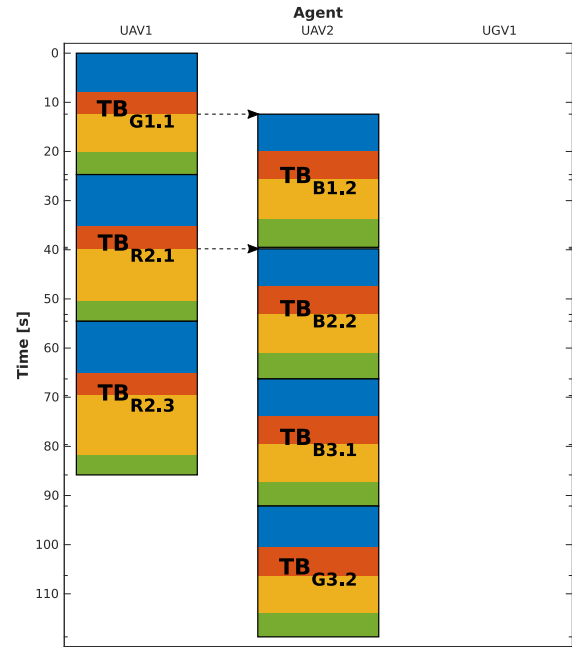


Fig. 5. Execution schedule of the first mission for the criteria defined as $\alpha = 0.5, \beta = 0.35, \gamma = 0.15$

Figure 5. Colored segments split each task $TB(b_i)$ into its actions, where blue corresponds to action GP , red to PU , yellow to GW , and green to PD . Dashed arrows represent *enables* relations which affect the final schedule.

In the second scenario, the most weight is given to the cost component of the criteria $\alpha = 0.35, \beta = 0.15, \gamma = 0.50$. Because the UGV has a lot smaller cost per time unit than the UAVs, tasks related to green bricks are assigned to the UGV. We have defined that red bricks provide 100% more points when assembled by UAVs, as opposed to 40% for the green bricks. Since the quality of the mission is still somewhat important, a better outcome is achieved if the UAVs assemble red bricks. The execution schedule for this scenario is shown in Figure 6.

Figure 7 shows the execution schedule of the third scenario which is fully focused on minimizing the total cost of the mission. The criteria is defined as $\alpha = 0, \beta = 0, \gamma = 1$. In this case, tasks related to red bricks are also assigned to the UGV since their increased score potential does not affect the mission outcome. Blue bricks are still assembled by the UAVs as is defined in the challenge description.

Lastly, we add a third UAV and employ the team of robots on a more complex mission specified as in Figure 8. In order to reduce the mission duration, but still include the UGV in the execution, we use the same criteria as in the second scenario of the first mission, $\alpha = 0.35, \beta = 0.15, \gamma = 0.5$.

Generated execution schedule for the last mission is shown in Figure 9. As in the previous scenarios, tasks are evenly distributed between agents in order to shorten the total mission duration and reduce overall cost while maximizing the score. Results confirm that the proposed method is capable of producing optimized high-level plans for different

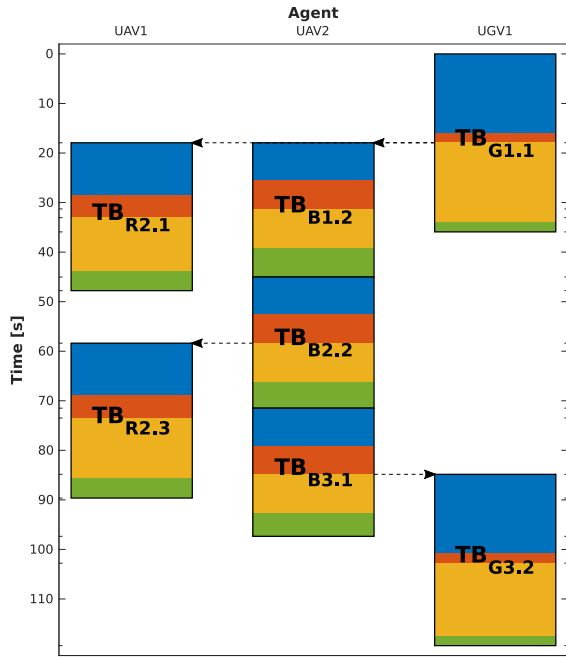


Fig. 6. Execution schedule of the first mission for the criteria defined as $\alpha = 0.35, \beta = 0.15, \gamma = 0.50$

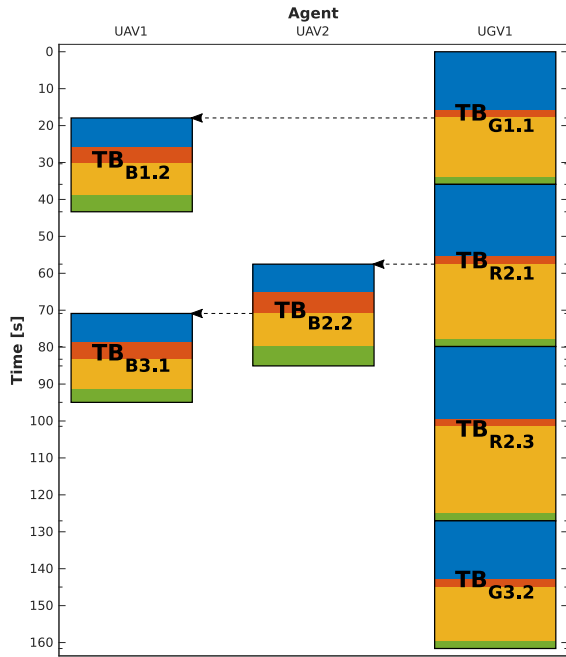


Fig. 7. Execution schedule of the first mission for the criteria defined as $\alpha = 0, \beta = 0, \gamma = 1$

agent and criteria setups, and that it meets current challenge requirements.

We conducted 20 additional simulations for problems of similar dimensionality and compared the performance with solutions obtained by the Gurobi Optimizer [29]. Gurobi Optimizer uses exact mathematical methods for solving mixed integer linear programming (MILP) problems and for the specified mission set always returns an optimal solution. We

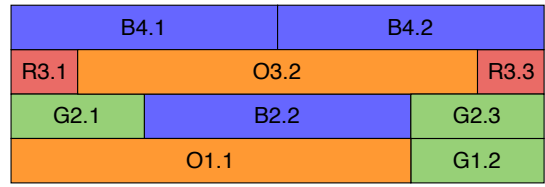


Fig. 8. Specified wall structure for the second mission

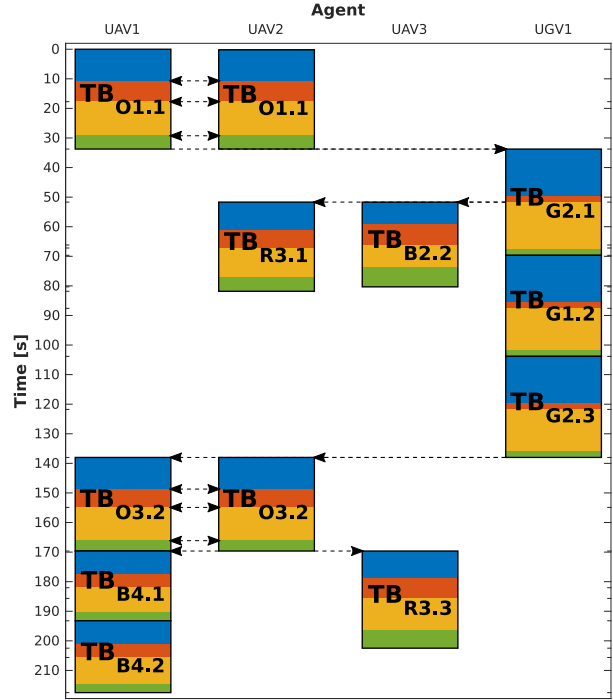


Fig. 9. Execution schedule of the second mission

also compare generated schedules with the state-of-the-art iterated auction-based approach with a single central agent acting as an auctioneer [25]. We tested each of the 20 mission scenarios on a set of five different criteria specifications and recorded the average value and the standard deviation of the objective function in respect to the optimal solution. The results of the experiments are shown in the Table I.

Results show that the proposed method on average performs within 12% of the optimum for the majority of tested criteria specifications, with the exception of the criteria fully oriented on minimizing the mission makespan. During the resolution of simple redundancy, tasks are always assigned to faster agents (e.g. UAVs), meaning that agents with significantly worse duration assessments (e.g. UGV) do not participate in the mission execution, thus prolonging the total duration.

Auction-based approach performs slightly better in terms of the achieved optimality gap, however, our approach is fully decentralized and has the ability to handle scheduling and simultaneous execution of multi-agent tasks, e.g. transportation of orange bricks. Another major advantage of the proposed method is that it is mission-agnostic and only a hierarchical task structure is needed, contrary to the Gurobi

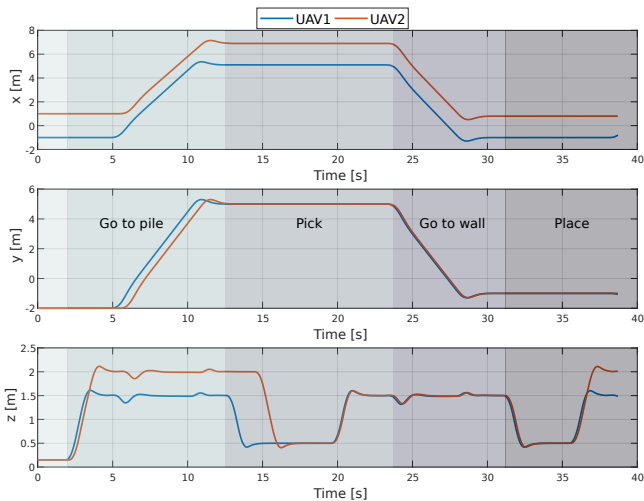


Fig. 10. Trajectories of UAVs during collaborative transportation of the orange brick

Optimizer which requires an exact mathematical formulation. Furthermore, mathematical methods used by Gurobi often fail to provide a valid solution when applied to more complex problems.

TABLE I

OPTIMALITY GAPS OF THE PROPOSED SOLUTION AND AUCTION BASED APPROACH COMPARED TO GUROBI OPTIMIZER. LOWER VALUES ARE BETTER.

Criteria		Proposed solution	Auction
$\alpha = 0.5, \beta = 0.35, \gamma = 0.15$	μ	11.15 %	8.18 %
	σ	6.26 %	6.00 %
$\alpha = 0.35, \beta = 0.15, \gamma = 0.5$	μ	6.43 %	8.10 %
	σ	10.96 %	3.82 %
$\alpha = 1, \beta = 0, \gamma = 0$	μ	0 %	0 %
	σ	0 %	0 %
$\alpha = 0, \beta = 1, \gamma = 0$	μ	42.38 %	2.18 %
	σ	11.54 %	4.187 %
$\alpha = 0, \beta = 0, \gamma = 1$	μ	2.61 %	0 %
	σ	1.62 %	0 %

An example of collaboration between two UAVs during the assembly of the large, orange brick is also examined. Due to the complexity of this task, other tasks are temporarily suspended during its execution in order to avoid collisions and various disturbances. *Enables* relations are therefore placed between *PD* and *GP* actions of each orange and its neighbouring bricks.

Figure 10 shows the trajectories of both UAVs during the execution of the described task. The figure is plotted concerning time, clearly marking agents' specific actions (takeoff, going to position, picking and placing the brick). Synchronization of the UAVs is achieved by modelling their tasks as complexly redundant and interconnecting action nodes with *enables* relations. During execution, participating agents form a direct communication link to share their status and further synchronize all steps of the currently executing action.

Video of the described simulations can be found at [30].

We also applied the method to the mission of constructing a wall with a larger number of bricks to highlight the scalability of the proposed solution. The figure of the resulting schedule is too large to be conveniently included in the paper itself, but it is available at [31].

V. CONCLUSION AND FUTURE WORK

In this paper, we have proposed and tested a decentralized task planning and coordination framework for a cooperative aerial-ground robotic team on a case study of the task of automated construction. The team needs to collaborate to autonomously locate, pick, transport and assemble various types of brick-like objects to build predefined structures. The approach assumes that the location of bricks and hierarchical decomposition of system mission are known a priori. We simulate physically accurate models of multicopters for UAV agents and a Gazebo model of Husky by Clearpath Robotics with mounted Schunk Powerball LWA 4P robotic arm for the UGV agent. Results show that our method can successfully generate schedules for cooperative missions of tightly coupled tasks under various setups of a multi-criteria objective.

As future work, we aim to implement and test the developed solution on a real aerial-ground robotic system used in the MBZIRC 2020 competition. Further, we plan to improve the method in terms of robustness and resilience to failed attempts of highly risky task execution with the inclusion of on-line re-planning capability. Finally, we plan to investigate the possible integration of dynamic task assignment algorithms into the already developed framework to make it even more applicable in real life situations.

ACKNOWLEDGMENT

This work has been supported by European Commission Horizon 2020 Programme through project under G. A. number 810321, named Twinning coordination action for spreading excellence in Aerial Robotics - AeRoTwin and the Mohammed Bin Zayed International Robotics Challenge. The work of doctoral student Barbara Arbanas has been supported in part by the "Young researchers' career development project—training of doctoral students" of the Croatian Science Foundation funded by the European Union from the European Social Fund.

REFERENCES

- [1] T. Petrovic, T. Haus, B. Arbanas, M. Orsag, and S. Bogdan, "Can uav and ugv be best buddies? towards heterogeneous aerial-ground cooperative robot system for complex aerial manipulation tasks," in *2015 12th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, vol. 01, July 2015, pp. 238–245.
- [2] B. Arbanas, A. Ivanovic, M. Car, M. Orsag, T. Petrovic, and S. Bogdan, "Decentralized planning and control for uav–ugv cooperative teams," *Autonomous Robots*, February 2018.
- [3] H. Qin, Z. Meng, W. Meng, X. Chen, H. Sun, F. Lin, and M. H. Ang, "Autonomous Exploration and Mapping System Using Heterogeneous UAVs and UGVs in GPS-Denied Environments," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1339–1350, February 2019.
- [4] C. Potena, R. Khanna, J. Nieto, R. Siegwart, D. Nardi, and A. Pretto, "AgriColMap: Aerial-Ground Collaborative 3D Mapping for Precision Farming," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1085–1092, April 2019.

- [5] M. Langerwisch, T. Wittmann, S. Thamke, T. Remmersmann, A. Tiderko, and B. Wagner, "Heterogeneous teams of unmanned ground and aerial robots for reconnaissance and surveillance - A field experiment," in *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. IEEE, October 2013, pp. 1–6.
- [6] P. Maini, K. Yu, P. B. Sujit, and P. Tokekar, "Persistent Monitoring with Refueling on a Terrain Using a Team of Aerial and Ground Robots," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, October 2018, pp. 8493–8498.
- [7] O. De Silva, G. K. I. Mann, and R. G. Gosine, "Pairwise observable relative localization in ground aerial multi-robot networks," in *2014 European Control Conference (ECC)*. IEEE, June 2014, pp. 324–329.
- [8] N. Staub, M. Mohammadi, D. Bicego, D. Prattichizzo, and A. Franchi, "Towards robotic MAGMaS: Multiple aerial-ground manipulator systems," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2017, pp. 1307–1312.
- [9] H. Yu, K. Meier, M. Argyre, and R. W. Beard, "Cooperative Path Planning for Target Tracking in Urban Environments Using Unmanned Air and Ground Vehicles," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 2, pp. 541–552, April 2015.
- [10] L. Marconi, S. Leutenegger, S. Lynen, M. Burri, R. Naldi, and C. Melchiorri, "Ground and aerial robots as an aid to alpine search and rescue: Initial SHERPA outcomes," in *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. IEEE, October 2013, pp. 1–2.
- [11] B. Gerkey and M. Mataric, "Sold!: auction methods for multirobot coordination," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 758–768, October 2002.
- [12] L. Liu and D. Shell, "Optimal Market-based Multi-Robot Task Allocation via Strategic Pricing," *Robotics: Science and Systems*, 2013.
- [13] H. ElGibreen and K. Youcef-Toumi, "Dynamic task allocation in an uncertain environment with heterogeneous multi-agents," *Autonomous Robots*, 2019.
- [14] C. Amato, G. Konidaris, G. Cruz, C. A. Maynor, J. P. How, and L. P. Kaelbling, "Planning for decentralized control of multiple robots under uncertainty," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2015, pp. 1241–1248.
- [15] B. Floriano, G. A. Borges, and H. Ferreira, "Planning for Decentralized Formation Flight of UAV fleets in Uncertain Environments with Dec-POMDP," in *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, June 2019, pp. 563–568.
- [16] F. Wu, S. Zilberstein, and X. Chen, "Online planning for multi-agent systems with bounded communication," *Artificial Intelligence*, vol. 175, no. 2, pp. 487–511, February 2011.
- [17] M. Gombolay, R. Wilcox, and J. Shah, "Fast Scheduling of Multi-Robot Teams with Temporospatial Constraints," *IEEE T-RO*, pp. 1–20, 2018.
- [18] P. Schillinger, M. Burger, and D. V. Dimarogonas, "Multi-objective search for optimal multi-robot planning with finite LTL specifications and resource constraints," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2017, pp. 768–774.
- [19] J. A. DeCastro, J. Alonso-Mora, V. Raman, D. Rus, and H. Kress-Gazit, "Collision-Free Reactive Mission and Motion Planning for Multi-robot Systems," in *Springer Proceedings in Advanced Robotics*. Springer, Cham, 2018, pp. 459–476.
- [20] B. Horling, V. Lesser, R. Vincent, T. Wagner, A. Raja, S. Zhang, K. Decker, and A. Garvey, "The taems white paper," 1999.
- [21] V. Lesser, K. Decker, T. Wagner, N. Carver, A. Garvey, B. Horling, D. Neiman, R. Podorozhny, M. N. Prasad, A. Raja, R. Vincent, P. Xuan, and X. Q. Zhang, "Evolution of the gpgp/taems domain-independent coordination framework," *Autonomous Agents and Multi-Agent Systems*, vol. 9, no. 1, pp. 87–143, July 2004.
- [22] K. S. Decker and V. R. Lesser, "Designing a family of coordination algorithms," in *Proceedings of the First International Conference on Multi-Agent System*. AAAI Press, 1995, pp. 73–80.
- [23] B. Arbanas, A. Ivanovic, M. Car, T. Haus, M. Orsag, T. Petrovic, and S. Bogdan, "Aerial-ground robotic system for autonomous delivery tasks," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2016.
- [24] G. A. Korsah, A. Stentz, and M. B. Dias, "A comprehensive taxonomy for multi-robot task allocation," *The International Journal of Robotics Research*, vol. 32, no. 12, pp. 1495–1512, 2013.
- [25] E. Nunes, M. McIntire, and M. Gini, "Decentralized multi-robot allocation of tasks with temporal and precedence constraints," *Advanced Robotics*, vol. 31, no. 22, pp. 1193–1207, 11 2017.
- [26] MBZIRC, "The challenge 2020," available: <https://www.mbzirc.com/challenge/2020>, Accessed: 23 July 2019. [Online]. Available: <https://www.mbzirc.com/challenge/2020>
- [27] O. E. Kundakcioglu and S. Alizamir, "Generalized assignment problem," in *Encyclopedia of Optimization*, C. A. Floudas and P. M. Pardalos, Eds. Boston, MA: Springer US, 2009, pp. 1153–1162.
- [28] J. D. C. Little, K. G. Murty, D. W. Sweeney, and C. Karel, "An algorithm for the traveling salesman problem," *Operations Research*, vol. 11, no. 6, pp. 972–989, December 1963.
- [29] Gurobi Optimization, LLC, "Gurobi optimizer reference manual," 2019. [Online]. Available: <http://www.gurobi.com>
- [30] Laboratory for robotics and intelligent control systems youtube channel, Accessed: 3 January 2020. [Online]. Available: <https://youtu.be/-0DU3w8jWWA>
- [31] Laboratory for robotics and intelligent control systems wiki page, Accessed: 2 December 2019. [Online]. Available: http://larics.github.io/GPGP_page