2023 International Conference on Unmanned Aircraft Systems (ICUAS)
June 6-9, 2023 | Lazarski University, Warsaw, Poland

Multi-Auctioneer Market-based Task Scheduling for Persistent Drone Delivery

Marco Rinaldi[1], Stefano Primatesta[1], Giorgio Guglieri[1] and Alessandro Rizzo[2]

2023 International Conference on Unmanned Aircraft Systems (ICUAS) | 979-8-3503-1037-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICUAS57906.2023.10155855

***Abstract*— Market-based task allocation methods represent an effective strategy for scheduling heterogeneous tasks to a heterogeneous multi-agent system, *e.g.,* a fleet of different Unmanned Aerial Vehicles (UAVs). This is mainly due to their computational efficiency, ease of hybridization with optimization techniques and adaptability to different communication architectures. In this paper, a novel hybrid auction-based task allocation architecture with multi-auctioneer agents' behavior is proposed for an Urban Air Mobility application. The proposed method aims to solve the combined problem of: *(i)* scheduling parcel pick-up and delivery tasks with time deadlines while minimizing the drones' energy consumption; *(ii)* scheduling battery re-charge tasks in order to ensure the service's persistency; and *(iii)* evaluating safe aerial routes since the UAVs fly over populated areas. The validity of the approach is demonstrated through Monte Carlo simulations. Moreover, being the proposed architecture distributed among the UAVs, the impact of communication failures on well-defined solution quality parameters is also investigated.**

***Index Terms*— auctions, task scheduling, UAS, drone delivery, MRTA, communication failure, energy optimization, multi-agent system, urban air mobility, TAMP**

I. INTRODUCTION

Multi-Agent Systems (MASs) are generally used to carry out complex problems, in which a single robot can fail or, in any case, carry out with poor performance. Multi-Robot Task Allocation (MRTA) is, in general, a difficult problem to which a unique resolution approach does not exist, especially if the instance exceeds even just a few numbers of robots and tasks. This is due to the impossibility of commercial solvers to produce an optimal allocation in a reasonable amount of time, since these problems fall, most likely, into the category of NP-hard problems (Nondeterministic Polynomial time problems Hard at least as the hardest problem in the NP class of decision problems). Several techniques have been developed over the last decade to approach these categories of problems and comprehensive surveys are available at the state of the art, such as in [1], [2] and [3]. Depending on the communication architecture, MRTA approaches are mainly divided in two categories: centralized and decentralized. Centralized methods rely on a central entity that allocates tasks considering a global utility function and the information shared by the other agents, while decentralized methods exploit the partial agents' knowledge of the environment to allocate tasks in a distributed way. Apart from this main categorization, several algorithms can tackle the MRTA problem: market-based, game theory-based, optimization-based, learning-based and hybrid approaches, the latter may combine different features of these algorithms. The evaluation of such approaches is based on the following criteria: complexity, optimality, and scalability.

This work focuses on market-based task allocation applied to an Urban Air Mobility application: the drone delivery. Market-based approaches are, in general, scalable and sub-optimal. Auction algorithms have polynomial computational complexity, they can be easily combined with deterministic optimization and are adaptable to both centralized and decentralized architectures. Auction-based task allocation works according to a communication protocol between a central allocator (auctioneer) and the system's agents (bidders), repeated every round. The number of rounds of the algorithm depends on the number of tasks advertised and allocated per round. In sequential single-item auctions, which are the most suitable allocation approach for tasks with temporal and priority constraints, the auctioneer broadcasts a task from the task set to each agent of the network, which sends as a response its valuation for that task. The auctioneer awards the task to the agent with the best valuation and the winner acknowledges the award. This process is repeated for a number of rounds equal to the number of tasks. Each round requires four communication phases: broadcast, offer, award, and acknowledge.

The comprehensive study of work [4] presents the main types of auction algorithms and evaluates their performances in presence of lossy communication for two types of evaluation functions: random and TSP-based costs. A decentralized coalition auction algorithm is proposed in [5] for a UAV delivery problem. The research in [6] shows how auctions can enable cooperative multi-robot allocation of tasks with time window constraints, preserving scalability and maximizing the number of allocated tasks. A combined auction and deep learning task assignment framework is proposed in [7] in order to schedule re-charge tasks to a multi-UAV network with the unknown agents' distribution in the environment being learned online. The work in [8] optimizes a drone charge scheduling problem with a game theory-based auction model that maximizes the rechargeable drones, *i.e.,* the optimal configuration of charging drones and charging stations. An iterative auction-based strategy is proposed and validated in [9] in order to maximize the utilities of both drones and charge stations simultaneously. The bidding mechanism is such that the stations bids to the drones depending on some critical factors such as the drones' assigned task deadlines. In our previous work about

[1]Marco Rinaldi (marco_rinaldi@polito.it), Stefano Primatesta (stefano.primatesta@polito.it) and Giorgio Guglieri (giorgio.guglieri@polito.it) are with the Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Torino, Italy,

[2]Alessandro Rizzo (alessandro.rizzo@polito.it) is with the Department of Electronics and Telecommunications, Politecnico di Torino, Torino, Italy.

Unmanned Aerial System (UAS) parcel transportation in [11], we showed how hybrid market and optimization-based greedy algorithms can effectively manage the allocation of a heterogeneous task set to a heterogeneous multi-UAV fleet, handling both tasks' constraints and robots' capacities.

In this paper, we extend the analysis and implementation of such approaches by proposing a lightweight distributed task allocation architecture that can manage both delivery and charge tasks, minimizing the overall energy consumption. The proposed architecture is such that each agent runs, in a decentralized way, both a path planning algorithm and an optimization algorithm. Differently from standard auction-based task allocation approaches, each agent is intended as both the auctioneer and the bidder of the network, depending on the type of task advertised, *i.e.,* either delivery or re-charge tasks. The main contributions of this work are *(i)* the formalization of a persistent drone delivery problem that takes into account safety, battery discharge, task time deadlines, energy-aware task scheduling, *(ii)* the design and implementation of a dynamic-auctioneer market-based decision-making architecture distributed among the UAVs of the network, *(iii)* the integration of both a decentralized risk-aware path planning strategy and a task execution optimization method within the proposed task allocation framework, *(iv)* the evaluation of the proposed approach with a high number of randomly generated instances in real scenarios, considering fallible communication also.

The paper is organized as follows. In Section II, the scenario of reference and the assumptions are defined. Section III presents the methodology behind the proposed architecture, focusing on the allocation algorithm, the communication architecture, the path planning strategy, the UAVs energy consumption model, and the optimization problems formulation. Simulation results are reported in Section IV, while conclusions and future works are drawn in Section V.

II. Problem Statement

The investigated problem is the allocation of a set of M payload transportation tasks with time deadlines to a heterogeneous fleet of N UAVs, minimizing the total energy consumption E_{TOT}. The battery discharge issue is taken into account with the formulation of another problem: allocating battery re-charge tasks to the UAVs in order to minimize the impact on the delivery process, *i.e.,* the flight time to any of the B available re-charge stations. Each delivery task requires the transportation of a payload mass m_P from a pick-up location to a delivery location, defined inside a defined populated urban area of reference, *i.e.,* the operational area, within a time window constraint $[T_{MIN}, T_{MAX}]$. The re-charge task set consists of re-charge station locations to be visited by the UAVs. Acceptance and regulation aspects, as highlighted in [12], require that the flight paths related to each UAV-task assignment be risk-aware.

The following assumptions are made:

- The payload capacity of a UAV is at most equal to its mass.
- UAVs cannot carry multiple payloads at the same time.
- UAVs cannot reach a charge station while executing a delivery task.
- The total re-charge time is defined as the sum of the battery re-charge time (linearly interpolated from the residual battery level of the UAV) and the UAV's flight time to the re-charge station.
- A delivery task requires the execution of *6* phases: take-off, cruise, and landing in both unloaded and loaded conditions.
- A re-charge task requires the execution of *3* phases since no payload is carried.
- UAV's velocity of execution of the task phases is seen as an optimization variable and is assumed to remain constant during phase execution.

III. Methodology

The proposed novel market-based task allocation architecture is conceptualized as a centralized architecture, with fallible communication and each UAV having both an auctioneer behavior and a bidder behavior depending on the task type. As shown in *Figure 1*, the fleet coordinator behaves as the network's auctioneer for delivery tasks, while it behaves as the bidder for re-charge tasks, the opposite holds for the other UAVs.

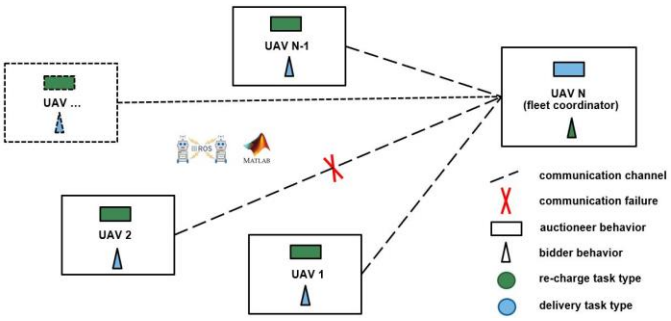

Figure 1. Architecture conceptualization.

A. Market-based Task Allocation

The approach is intended as a sequential single-item multi-robot market-based task allocation for delivery tasks. At each auction round r, the fleet coordinator broadcasts a delivery task to the other UAVs of the network (including itself); then, each UAV computes, in a decentralized manner, the safe paths related to the task and offers the optimal estimated required energy for task execution as its bid. The auctioneer awards the task to the best bidder, *i.e.,* the UAV with the minimum estimated optimal energy for the task of round r, and the winner acknowledges the assignment. *4* communication phases characterize each delivery task auction round (broadcast, offer, award, acknowledge) and each phase is associated to a probability of communication failure p. It is assumed that the auctioneer never experiences communication failures with itself. In order to guarantee the service persistency, before the auctioneer broadcasts the new delivery task, another market-inspired allocation is nested with the allocation of the delivery tasks. This time, if the UAV's energy level is below a certain threshold, a *3*-phase auction allocation begins.

Algorithm 1: Market − based Task Allocation

for $i = 1$ **to** N
  UAV^i computes E^i_{MIN} by means of $Optimization(6)$
for $r = 1$ **to** M
  for $i = 1$ **to** N
    UAV^N broadcasts(p) $delivery^r_{task}$ to UAV^i
    if $(m^i \geq m^r_p) \wedge (T^i_i \leq T^r_{MAX})$
      UAV^i computes $L^{i,r}_1, L^{i,r}_2$ by means of $PathPlanner(UAV^i, delivery^r_{task})$
      UAV^i computes $\boldsymbol{V^{i,r}_{DT}}, E_{DT}(m^i, m^i_p, \boldsymbol{V^{i,r}_{DT}}, L^{i,r}_1, L^{i,r}_2, h^r)$ by means of $Optimization(7)$
      UAV^i sets $bid^{i,r} = E_{DT}(m^i, m^i_p, \boldsymbol{V^{i,r}_{DT}}, L^{i,r}_1, L^{i,r}_2, h^r)$
      UAV^i offers(p) $bid^{i,r}$ to UAV^N
    UAV^N updates $SetOfBids$
  $flag = 0$
  while $(SetOfBids \neq \emptyset) \wedge (flag = 0)$
    for $i = 1$ **to** N
      if $(bid^{i,r} \neq null) \wedge (bid^{i,r} = \min(SetOfBids))$
        UAV^N awards(p) $delivery^r_{task}$ to UAV^i
        UAV^i acknowldges(p) the assignment to UAV^N
        if $(broadcast(p) = offer(p) = award(p) = acknowldge(p) = true)$
          $delivery^r_{task}$ is assigned to UAV^i
          UAV^i updates $E^i_{UAV}, T^i_i, UAV^i_{location}$
          $flag = 1$
        else
          UAV^N removes $bid^{i,r}$ from $SetOfBids$
  for $i = 1$ **to** N
    if $\left(E^i_{UAV} \leq E^i_{MIN} + K_T E^{MAX^i}_{UAV}\right)$
      UAV^i acknowledges'(p) $DischargedStatus = true$ to UAV^N
      for $j = 1$ **to** B
        UAV^N computes $d^{i,j}$ by means of $PathPlanner(UAV^i, charge^j_{task})$
        UAV^N sets $bid^{i,j} = d^{i,j}$
      $NearestStation = \min(bid^{i,1}, \ldots, bid^{i,B})$
      UAV^N computes $\boldsymbol{V^i_{CT}}(NearestStation)$ by means of $Optimization(8)$
      UAV^N awards'(p) $NearestStation$ to UAV^i
      UAV^i acknowldges''(p) $DischargedStatus = false$ to UAV^N
      if $(acknowldge'(p) = award'(p) = acknowldge''(p) = true)$
        $charge_{task}(NearestStation)$ is assigned to UAV^i
        UAV^i updates $E^i_{UAV}, T^i_i, UAV^i_{location}$

In particular, the UAV to be charged (behaving as an auctioneer) broadcasts its status of discharge to the fleet coordinator that, as a bidder, computes and offers both the nearest re-charge station and the optimal task execution velocity to the UAV to be charged, which then acknowledges the assignment. The pseudo-code of the proposed task allocation strategy is shown in *Algorithm 1*, with the auction communication phases being reported in blue bold. Refer to Sections III.*C* and III.*D* for the complete terminology and for the complete explanation of the energy consumption model and the optimizations adopted in *Algorithm 1*.

Considering that the addressed problem is NP-hard, it is worth noticing that the whole auction-based algorithm runs in polynomial time, with message-related complexity being *C=O(NM)*. Due to the intractability of such a heterogeneous problem, even for medium size instances, the optimality of the solution per se is not really an issue for the sake of the evaluation of the proposed approach. Also, it is important to highlight that *B* does not influence *C* as the charge-related auction allocation, nested with the main delivery tasks allocation, works equivalently to a single-task and single-agent allocation. Furthermore, since a new task is advertised and allocated at each round, the proposed allocation architecture is suitable for handling dynamic tasks as well as tasks with priorities, dynamic agents' availability and distribution in the network.

B. Path Planning

The proposed market-based task allocation architecture requires the evaluation of several paths connecting the UAV positions to the task positions. The path planning strategy adopted in this paper is taken from our previous works in [13], [14], [15], as the problem of generating safe aerial paths to enable UAVs operations over populated areas has already been tackled. The problem of minimum-risk aerial path computation is decomposed in two stages: first, a *2D* location-based risk map is generated considering the population density on the ground and the UAV parameters (UAV mass, payload mass, UAV dimensions, maximum speed, etc.) and, then, a sampling-based algorithm searches for the minimum risk path minimizing the overall risk in the risk map and the flight time. The risk map associates a risk value, expressed in expected fatalities per hour (h^{-1}), to each location. Since the sampling algorithm returns, in general,

sub-optimal solutions to be evaluated by the task allocator, the average risk of each computed path is verified to be below an Equivalent Level of Safety (ELOS) of $10^{-6}h^{-1}$, as defined in [12], before the planner solution is considered valid. For further details about the risk map generation and the risk-aware path planning algorithm refer to [13], [14] and [15].

C. Energy Consumption Model

The energy consumption model of the UAVs, taken from the model proposed in [16], has already been theoretically derived from the helicopter literature, from both the actuator disk theory and the blade element theory, and validated experimentally with a quadrotor. The power consumption model is decoupled in three main components: induced power (produces thrust by propelling air downward), profile power (overcomes rotational drag due to rotation of propellers' blades) and parasite power (resists drag during translational relative motion between UAV and wind). Equations (1), (2) and (3) show the analytical expression of the induced energy $E_i(\cdot)$, the profile energy $E_p(\cdot)$ and the parasite energy $E_{par}(\cdot)$ components, under the assumptions that *(i)* the UAV's task execution velocity V is constant over the path length L, *(ii)* the wind velocity is negligible, *(iii)* the UAVs' heading angles are small.

$$E_i(T, V_v, V, L) = T\left[\frac{V_v}{2} + \sqrt{\left(\frac{V_v}{2}\right)^2 + \frac{T}{2\rho A_r}}\right]\frac{L}{V} \tag{1}$$

$$E_p(T, V, L) = \sqrt{b^3}c_d N c \rho \frac{R^4}{8}\sqrt{T^3}\frac{L}{V} \tag{2}$$

$$E_{par}(V, L) = \frac{1}{2}c_d \rho A_d V^2 L \tag{3}$$

With the adopted notation, T is the total thrust, V_v is the vertical drone velocity, ρ is the air density, A_r is the total propeller area, b is the thrust coefficient, c_d is the drag coefficient, N is the number of blades in a propeller, c is the blade width, R denotes the blade radius, A_d denotes UAV cross section with respect to the direction of motion, m is the UAV mass. The estimated consumed energy to complete either a delivery task $E_{DT}(\cdot)$ or a re-charge task $E_{CT}(\cdot)$ is defined as the sum of the three contributions defined above, repeated for each task phase. It follows the formulations of the delivery-related energy consumption model and the charge-related one.

$$\begin{aligned} &E_{DT}(m, m_p, \boldsymbol{V_{DT}}, L_1, L_2, h) \\ &= E_i(mg, V_1, V_1, h) + E_p(mg, V_1, h) + E_i(mg, 0, V_2, L_1) + E_p(mg, V_2, L_1) + E_{par}(V_2, L_1) \\ &+ E_i(mg, V_3, V_3, h) + E_p(mg, V_3, h) + E_i\big((m+m_p)g, V_4, V_4, h\big) + E_p\big((m+m_p)g, V_4, h\big) \\ &+ E_i\big((m+m_p)g, 0, V_5, L_2\big) + E_p\big((m+m_p)g, V_5, L_2\big) + E_{par}(V_5, L_2) \\ &+ E_i\big((m+m_p)g, V_6, V_6, h\big) + E_p\big((m+m_p)g, V_6, h\big) \end{aligned} \tag{4}$$

$$\begin{aligned} &E_{CT}(m, \boldsymbol{V_{CT}}, d, h) \\ &= E_i(mg, V_1', V_1', h) + E_p(mg, V_1', h) + E_i(mg, 0, V_2', d) + E_p(mg, V_2', d) + E_{par}(V_2', d) \\ &+ E_i(mg, V_3', V_3', h) + E_p(mg, V_3', h) \end{aligned} \tag{5}$$

$\boldsymbol{V_{DT}}=(V_1, V_2, V_3, V_4, V_5, V_6)$ and $\boldsymbol{V_{CT}}=(V_1', V_2', V_3')$ denote the delivery task velocity vector and the re-charge task velocity vector, respectively. Each task phase is associated to a velocity of execution: V_1, V_2, V_3 denote the take-off, cruise, and landing velocities during the unloaded pick-up phases, V_4, V_5, V_6 denote the take-off, cruise, and landing velocities during the loaded delivery phases, V_1', V_2', V_3' denote the take-off, cruise, and landing velocities during the unloaded re-charge task phases. For a delivery task associated to a height of flight h, the UAV flights a safe path of length L_1 to reach the pick-up location and a safe path of length L_2, carrying a payload of mass m_p, to reach the delivery location. Considering a re-charge task associated to a height of flight h, the UAV flights a safe path of length d to reach the charge station location. For the sake of clarity, we specify that the landing velocities, *i.e.,* V_3, V_6 and V_3', are considered with negative sign. Including the payload in the energy consumption model is essential since the UAVs' battery discharge ratio is known to be proportional to the carried payload mass, as shown in [10].

D. Optimization

The optimization problem formulated in (6) computes the minimum energy required by the UAV's battery E_{MIN} such that the UAV can reach a charge station, within a specified maximum flight time ΔT_{MAX}, in the worst-case scenario. In the worst-case scenario, the UAV and the charge station are located at the extremities of the operational area at distance d_{MAX}. The minimization of $E_{CT}(\cdot)$ is subject to the constraints that *(i)* the total travel time is less than ΔT_{MAX} (T_i denotes the first time instant at which the UAV is able to execute a new task, either a re-charge or a delivery task), *(ii)* $E_{CT}(\cdot)$ does not overcome the maximum energy stored in the UAV's battery E_{MAX}, *(iii)* the optimal re-charge task execution velocity $\boldsymbol{V_{CT}}$ is consistent with the upper and lower bounds of the UAV's velocity, $\boldsymbol{V_{UB}}$ and $\boldsymbol{V_{LB}}$ respectively.

$$\begin{aligned} &min\ \ E_{CT}(m, \boldsymbol{V_{CT}}, d_{MAX}, h)\ , \qquad subject\ to: \\ &T_i + h\left(\frac{1}{V_1'} - \frac{1}{V_3'}\right) + \frac{d_{MAX}}{V_2'} \le \Delta T_{MAX}\ , \\ &E_{CT}(m, \boldsymbol{V_{CT}}, d_{MAX}, h) \le E_{MAX}\ , \\ &\boldsymbol{V_{CT}} \le \boldsymbol{V^{UB}}\ , \\ &\boldsymbol{V_{CT}} \ge \boldsymbol{V^{LB}} \end{aligned} \tag{6}$$

The optimization problem formulated in (7) minimizes the estimated energy for a delivery task $E_{DT}(\cdot)$, under the constraints that *(i)* the task time deadlines are respected; *(ii)* the remaining energy of the UAV after the task completion, $E_{UAV} - E_{DT}(\cdot)$, is above the threshold E_{MIN} obtained by the optimization problem formulated in (6); *(iii)* the optimal delivery execution velocity $\boldsymbol{V_{DT}}$ is consistent with the upper and lower bounds of the UAV's velocity, $\boldsymbol{V_{UB}}$ and $\boldsymbol{V_{LB}}$ respectively. The optimization problem formulated in (8) minimizes the flight time of the UAV allocated charge station, subject to the constraints that *(i)* the energy stored in the UAV's battery E_{UAV} is enough to reach the charge station and *(ii)* the consistency with $\boldsymbol{V_{UB}}$ and $\boldsymbol{V_{LB}}$ is respected.

$$min\ \ E_{DT}(m, m_p, \boldsymbol{V_{DT}}, L_1, L_2, h)\ ,\qquad subject\ to:$$

$$T_i + h\left(\frac{1}{V_1} - \frac{1}{V_3} + \frac{1}{V_4} - \frac{1}{V_6}\right) + \frac{L_1}{V_2} + \frac{L_2}{V_5} \geq T_{MIN}\ ,$$

$$T_i + h\left(\frac{1}{V_1} - \frac{1}{V_3} + \frac{1}{V_4} - \frac{1}{V_6}\right) + \frac{L_1}{V_2} + \frac{L_2}{V_5} \leq T_{MAX}\ ,$$

$$E_{MIN} + E_{DT}(m, m_p, \boldsymbol{V_{DT}}, L_1, L_2, h) \leq E_{UAV}\ ,$$

$$\boldsymbol{V_{DT}} \leq \boldsymbol{V^{UB}}\ ,$$

$$\boldsymbol{V_{DT}} \geq \boldsymbol{V^{LB}} \tag{7}$$

$$min\ \ \left[h\left(\frac{1}{V_1'} - \frac{1}{V_3'}\right) + \frac{d}{V_2'}\right]\ ,\qquad subject\ to:$$

$$E_{UAV} - E_{CT}(m, \boldsymbol{V_{CT}}, d, h) \geq 0\ ,$$

$$\boldsymbol{V_{CT}} \leq \boldsymbol{V^{UB}}\ ,$$

$$\boldsymbol{V_{CT}} \geq \boldsymbol{V^{LB}} \tag{8}$$

The decision variable of the optimization problems formulated above is the task execution velocity ($\boldsymbol{V_{CT}}$ for re-charge tasks, $\boldsymbol{V_{DT}}$ for delivery tasks), while the other variables appear as constant parameters. Tasks and drones' related variables appear as constant parameters. All the optimization problems formulated above can be brought in the standard form of constrained optimization, shown in (9), with the optimization variables' vector being $\boldsymbol{x}=(x_1,x_2,\ldots,x_n)$, the set of m inequality constraints being $\boldsymbol{g(x)}=(g_1(\boldsymbol{x}),g_2(\boldsymbol{x}),\ldots,g_m(\boldsymbol{x}))$ and the scalar cost function being $f(\boldsymbol{x})$, with $\boldsymbol{x}\in R^n$, $f(\boldsymbol{x})$: $R^n \rightarrow R$, $\boldsymbol{g(x)}$: $R^n \rightarrow R^m$.

$$min\ \ f(\boldsymbol{x})\ ,\qquad subject\ to:$$

$$\boldsymbol{g(x)} \leq \boldsymbol{0} \tag{9}$$

Sequential Unconstrained Optimization (SUO) is an efficient methodology for iteratively solving constrained optimization problems. In general, SUO consists of the implementation of a systematic procedure that, by performing a series of unconstrained optimizations on an enhanced objective function [17], tackles constrained optimization problems. In this work, we implement a hybrid Penalty-Lagrange SUO method to find sub-optimal solutions to problems with the formulation shown in (9). The Penalty-Lagragian function is defined by the sum of the objective function of the original constrained optimization problem and the constraints-dependent penalty terms. At each execution step of the algorithm, the penalty parameters are updated depending on the satisfaction of the constraints. Also, the update criterium of one of the penalty parameters approximates the Lagrangian function's gradient-dependent computation of the Lagrange multipliers in the classical Lagrange Method for equality constraints. In general, this method has been proven to be able to sub-optimally solve constrained optimization problems in presence of nonlinearities either in the objective function or in the constraints. A finite convergence to a sub-optimal solution can be achieved for finite penalty parameters' values, the latter should not be excessively increased in order to avoid numerical problems. The implemented Penalty-Lagrange Optimization method performs unconstrained optimization on the Penalty-Lagrangian function $L(\boldsymbol{x},\boldsymbol{\lambda},\boldsymbol{\rho})$, defined in (10), at each iteration k, with the penalty parameters being $\boldsymbol{\rho}=(\rho_1, \rho_2, \ldots, \rho_m)$ and $\boldsymbol{\lambda}=(\lambda_1, \lambda_2, \ldots, \lambda_m)$, and the maximum number of iterations being N_{ITER}. *Algorithm 2* shows the method's implementation. The penalty parameters' update criteria depend on constraints' satisfaction, the variation of the penalty parameters at each iteration depends on the tunable scalars q_1, q_2, q_3. The convergence speed of the algorithm depends on q_1, q_2, q_3, μ and ξ. The stopping condition of *Algorithm 2* depends on constraints' satisfaction and the tolerance parameters μ, ξ. μ and ξ denotes the tolerances below which the difference between two consecutive optimal solutions can be neglected and the algorithm stops. There is no theoretical guarantee that *Algorithm 2* converges the global optimum. The numerical iterative approximation of λ derives from the comparison of the partial derivatives of the Penalty-Lagrangian function $L(\boldsymbol{x},\boldsymbol{\lambda},\boldsymbol{\rho})$ and the classical Lagrangian function, as in [18]. Obviously, both the performances and the convergence guarantee depend also on the unconstrained optimization solver of the Penalty-Lagrangian function.

$$L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}) = f(\boldsymbol{x}) + \sum_{i=1}^{m} max\left(\frac{1}{2}\lambda_i + \rho_i g_i(\boldsymbol{x}), 0\right)^2 \tag{10}$$

Algorithm 2: Penalty − Lagrange Optimization

$flag = 0$
for $k = 1$ **to** N_{ITER}
  $\boldsymbol{x}_*^k = \arg\min L(\boldsymbol{x}_*^{k-1}, \boldsymbol{\lambda}^{k-1}, \boldsymbol{\rho}^{k-1})$
  for $i = 1$ **to** m
    $\lambda_i^k = \max\left(\lambda_i^{k-1} + q_1\rho_i^{k-1} g_i(\boldsymbol{x}_*^k), 0\right)$
    if $(\|\lambda_i^k - \lambda_i^{k-1}\| < q_2)$
      $\rho_i^k = q_3\rho_i^{k-1}$
    else
      $\rho_i^k = \rho_i^{k-1}$
  if $(\|\boldsymbol{x}_*^k - \boldsymbol{x}_*^{k-1}\| \leq \mu) \wedge (\boldsymbol{g}(\boldsymbol{x}_*^k) \leq \boldsymbol{0}) \wedge (|f(\boldsymbol{x}_*^k) - f(\boldsymbol{x}_*^{k-1})| \leq \xi)$
    $flag = 1$
    exit
if $(flag = 1)$
  return $\boldsymbol{x}_*^k$
else
  return *null*

IV. Simulation Results

The proposed hybrid market-based allocation algorithm is implemented in MATLAB, the unconstrained optimization performed on the Lagrangian function exploits the build-in *fminsearch(·)* MATLAB function, with default settings. This optimizer implements the Nelder-Mead simplex algorithm as defined in [19]. Considering that the formulation of $L(\boldsymbol{x},\boldsymbol{\lambda},\boldsymbol{\rho})$ yields to a 6-dimensional Lagrangian function, it is straightforward that there is no theoretical guarantee that the optimization algorithm converges since the convergence is demonstrated for dimension 1 and partially for dimension 2. An example of an optimal solution provided by *Algorithm 2* is given with an instance of the problem formulated in

Optimization (7), as shown in *Figures 2* and *3*. In particular, *Figure 2* shows the evolution of the values of both the objective function $E_{DT}(\cdot)$ of the constrained optimization problem and the objective function $L(\cdot)$ of the unconstrained optimization problem at each step k. On the other hand, *Figure 3* shows the evolution of $L(\cdot)$ over the iterations of a single *fminsearch()* MATLAB function call. The medium execution time of *Algorithm 2* is *0.4s*.

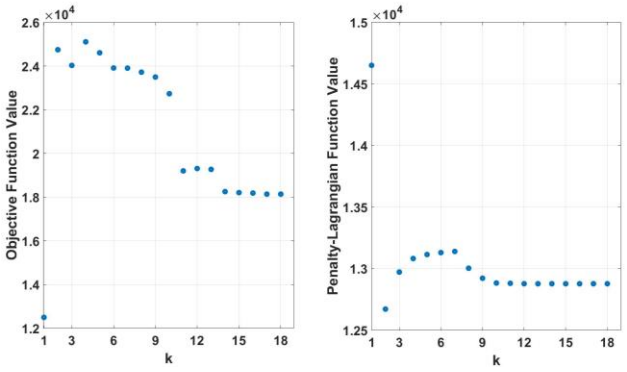

Figure 2. On the left, evolution of the constrained problem's objective function values at each execution step k. On the right, evolution of the Penalty-Lagrangian function values at each execution step k.

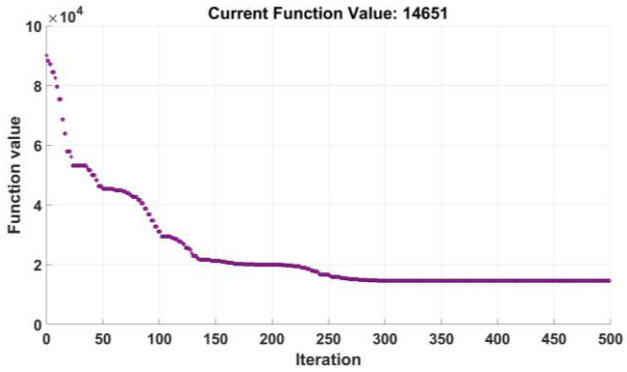

Figure 3. Evolution of the Penalty-Lagrangian function values between $k=0$ and $k=1$ (unconstrained optimization).

The risk-aware path planner is implemented in ROS/C++ and it is called in *Algorithm 1* by means of build-in MATLAB functions. The communication among the UAVs is simulated by means of ROS Publisher and Subscriber Nodes. The capability of the proposed approach of allocating a heterogenous task set, with different payloads and time deadlines, to a heterogenous multi-agent system, *i.e.,* the UAS, is demonstrated by means of Monte Carlo simulations repeated for different scenarios. Different types of UAVs are considered, the characteristics are reported in *Table I*. The operational area is defined as a squared region in a populated urban area in the city of Turin, Italy. The latitude and longitude extremities that define the UAS operational area are *[45.0379°,45.0743°]* and *[7.6146°,7.6894°]* respectively.

TABLE I. UAVs CHARACTERISTICS

UAV type	A	B	C	D
m [kg]	1.0	2.0	3.0	4.0
$\mathbf{A_r}$ [m^2]	0.20	0.28	0.36	0.44
$\mathbf{A_d}$ [m^2]	0.40	0.60	0.80	1.0
$\mathbf{c_d}$	0.1	0.2	0.3	0.4
b [kg·m]	0.001	0.002	0.003	0.004
c [m]	0.02	0.02	0.03	0.04
R [m]	0.10	0.12	0.15	0.18
$\mathbf{V_{MAX}}$ [m/s]	16.0	19.0	20.0	22.0
$\mathbf{E_{MAX}}$ [MJ]	0.68	0.90	1.17	1.43

Before presenting the simulation results that demonstrate the validity of the proposed approach, a simplified scenario is illustrated in order to show how the combined task allocation and path planning works. The following instance is selected: a task set of *3* payload masses of *1.0kg*, *2.0kg*, and 3.*0kg* to be delivered within the same "soft" time window deadline of *[0.0, 1.0]h*; a fleet of *3* UAVs of type *A*, *B* and *C* (with the following initial energy in the batteries: $E_{UAV}{}^{A}=E_{MAX}{}^{A}$, $E_{UAV}{}^{B}=0.3\cdot E_{MAX}{}^{B}$, $E_{UAV}{}^{B}=E_{MIN}{}^{C}$); 1 re-charge station, disposed in the operational area as in *Figure 4*. The tasks are sequentially broadcasted by the fleet coordinator (*UAV C*) according to ascending payload values. Each algorithm's round r produces the following result: *Task 1* is allocated to *UAV A* and the re-charge task is allocated to *UAV C* ($r=1$), *Task 2* is allocated to *UAV C* ($r=2$), *Task 3* is allocated to *UAV C* ($r=3$). It is evident that the allocation solution differs from the "intuitive" expected result of having *Task 2* allocated to *UAV B*, which is closer to the parcel pick-up position than *UAV C*. This is due the energy-aware allocation criterium.

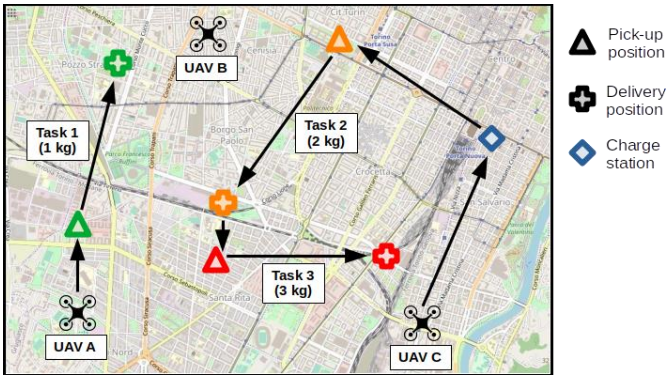

Figure 4. Representation of a problem instance's example.

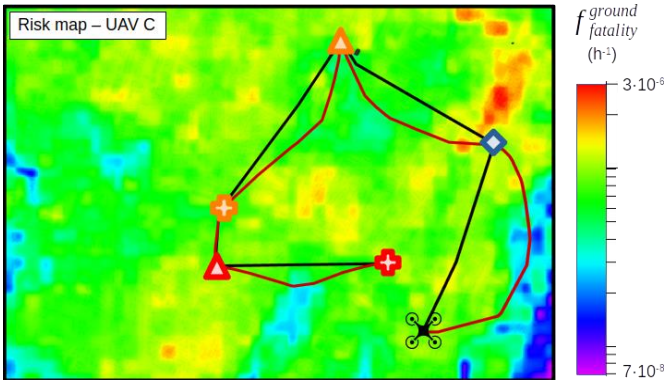

Figure 5. Risk map computed for a UAV type C , with line-of-sight paths (black) and minimum-risk paths (red).

Figure 5 shows how the task assignment-related paths of UAV *C* differs from the Line-Of-Sight (LOS) paths, since the path planner is risk-aware. The distribution of the risk values is shown in *Figure 6*, with the average risk of the path planner solution being lower than both the ELOS and the average risk of the LOS paths.

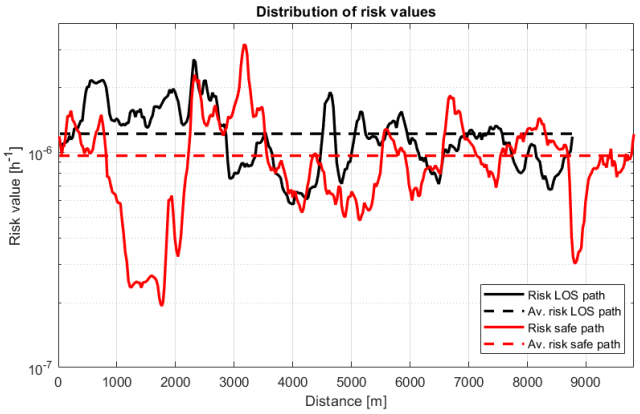

Figure 6. Distribution of risk values of the paths of *Figure 3*.

It follows the simulation settings and the considered scenario. The UAS is composed by *4* different UAVs ($N=4$), with *1* UAV per type. Since UAV *D* is the heaviest UAV of the fleet and potentially able to execute all tasks, it is set as the coordinator of the network. The task set is composed of *24* tasks ($M=24$), with *6* tasks per payload mass. Payloads of *0.5kg*, *1.0kg*, *1.5kg*, and *2.0kg* are considered. *4* re-charge stations are available inside the operational area ($B=4$), the maximum re-charge time for the linear interpolation of the battery re-charge time is set to *0.5h*, $K_T=0.1$, $d_{MAX}=7km$, $\Delta T_{MAX}=0.25h$, $\rho=1.3kg \cdot m^{-3}$. The optimization parameters' settings are the following: $\xi=5$, $\mu=0.1$, $\boldsymbol{x^0=0.1}$, $\boldsymbol{\rho^0=1}$, $\boldsymbol{\lambda^0=0}$, $N_{ITER}=100$, $q_1=2$, $q_2=0.5$, $q_3=5$. For the sake of the evaluation, two scenarios of reference are defined:

- Scenario S_A: each delivery task has a random deadline T_{MAX} from *0* to *3h* and $T_{MIN}=0$.
- Scenario S_B: each delivery task has a random deadline T_{MIN} from *0* to *1hour*, and a random deadline T_{MAX} from *1* to *3h*.

The quality of the task allocation results is quantified by means of well-defined solution quality parameters: the total energy consumption E_{TOT}, the number of unassigned delivery tasks $N_{DT}{}^{UA}$ and the number of re-charge stations visited N_{CT}. *Table II* shows the mean value θ and the standard deviation σ computed from *50* Monte Carlo simulations repeated for each scenario; with UAVs locations, tasks pick-up, height of flight *h*, delivery and re-charge station's locations being initialized randomly (within the operational area) at the beginning of each simulation. The height of flight associated to each task is limited to the range *[50,120]m*. Tasks are advertised according to ascending values of T_{MAX}, so that tasks with stricter deadlines are associated to a higher priority. This is done also to minimize the total number of unassigned tasks. Each UAV is assumed to be fully charged at the beginning of the simulations. In order to highlight the validity of the proposed allocation strategy, the results are compared to a variant of the approach that minimizes the total flight time instead of E_{TOT} ($S_A{}^*$ and $S_B{}^*$). This variant in the allocation criterium is implemented by substituting the cost function of the optimization problem formulated in (7) with the UAV's task-related flight time. Consequently, the UAVs' bids in *Algorithm 1* contain the flight time values instead of the estimated energy consumption values. *Table II* shows the results obtained assuming an ideal communication between the UAVs.

TABLE II. SIMULATION RESULTS WITH PERFECT COMMUNICATION

	E_{TOT} [MJ]		$N_{DT}{}^{UA}$		N_{CT}	
	θ	σ	θ	σ	θ	σ
S_A	3.12	1.02	3	1.44	2	1.19
$S_A{}^*$	27.57	1.12	1	0.82	23	0.96
S_B	3.03	0.81	4	1.59	1	0.97
$S_B{}^*$	25.80	2.02	1	1.21	21	1.99

The impact of communication failures on the solution quality parameters is also investigated, *Table III* shows how different levels of probability of communication failure affect the task allocation solution. Since the architecture is centralized, the lack of robustness to lossy communication is expected. The results shown in *Table III* are obtained from *50* Monte Carlo simulations repeated for each considered value of probability of failure *p*. One instance of scenario S_A and one instance of scenario S_B (always the same) initialize each simulation.

The results with perfect communication show that by paying, in average, the price of having a few more unassigned tasks with respect to the minimum flight time variances, the minimum energy allocation ensures a significant reduction of E_{TOT} (up to *90%*) and N_{CT} (up to *95%*). As far as the results with lossy communication are concerned, the poor robustness of the architecture to communication failures is reflected by the number of unassigned tasks, which increases with the probability of failure.

TABLE III. SIMULATION RESULTS WITH LOSSY COMMUNICATION

	p	E_{TOT} [MJ]		$N_{DT}{}^{UA}$		N_{CT}	
		θ	σ	θ	σ	θ	σ
S_A	10%	3.60	0.70	6	1.38	1.5	0.862
	20%	3.40	0.85	8	1.84	1	0.781
	30%	2.71	0.75	11	1.99	1	0.817
	40%	2.72	0.72	13	2.15	1	0.720
	50%	2.64	0.63	15	1.76	1	0.714
S_B	10%	3.77	0.96	5	1.50	2	0.96
	20%	3.56	0.71	7	1.57	2	0.82
	30%	3.53	1.24	10	1.99	2	1.15
	40%	3.30	0.90	13	1.75	2	0.81
	50%	3.02	1.29	15	2.08	2	1.05

V. Conclusion

In this work, a task allocation problem related to a persistent drone delivery service is formulated, and a distributed multi-agent market-based solution is presented. The proposed architecture is shown to be able to tackle tasks with temporal constraints, minimizing the heterogeneous fleet of UAVs' energy consumption. The architecture's robustness to communication failures is also investigated by means of well-defined solution quality parameters. Evaluations are made with Monte Carlo simulations. Future works will be focused on dynamic task handling, resource sharing among the UAVs and methods to reduce the impact of communication failures on the solution quality parameters of the task allocation architecture.

Acknowledgment

This work was supported by a PhD fellowship from Distretto Aerospaziale Piemonte and was carried out within the MOST – Sustainable Mobility National Research Center and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.4 – D.D. 1033 17/06/2022, CN00000023). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

References

[1] G. M. Skaltsis, H. S. Shin and A. Tsourdos, "A survey of task allocation techniques in MAS," *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, Athens, Greece, 2021, pp. 488–497.

[2] A. Khamis, A. Hussein and A. Elmogy, "Multi-robot task allocation: a review of the state-of-the-art," *Studies in Computational Intelligence*, vol. 604, pp. 31–51, 2015.

[3] S. Poudel and S. Moh, "Task assignment algorithms for unmanned aerial vehicle networks: a comprehensive survey," *Vehicular Communication*, vol. 35, 2022.

[4] M. Otte, M.J. Kuhlman and D. Sofge, "Auctions for multi-robot task allocation in communication limited environments," *Autonomous Robots*, vol. 44, pp. 547–584, 2020.

[5] M. Braquet and E. Bakolas, "Greedy decentralized auction-based task allocation for multi-agent systems," *IFAC-PapersOnLine*, vol. 24, pp. 675–680, 2021.

[6] E. Nunes and M. Gini, "Multi-robot auctions for allocation of tasks with temporal constraints," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, 2015.

[7] M. Shin, J. Kim and M. Levorato, "Auction-Based Charging Scheduling With Deep Learning Framework for Multi-Drone Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, pp. 4235–4248, 2019.

[8] M. Torky, M. El-Dosuky, E. Goda, V. Snášel and A.E. Hassanien, "Scheduling and Securing Drone Charging System Using Particle Swarm Optimization and Blockchain Technology," *Drones*, vol. 6, no. 9, p. 237, 2022.

[9] V. Hassija, V. Saxena and V. Chamola, "Scheduling drone charging for multi-drone network based on consensus time-stamp and game theory," *Computer Communications,* vol. 149, pp. 51–61, 2020.

[10] M. Torabbeigi, G.J. Lim and S.J. Kim, "Drone Delivery Scheduling Optimization Considering Payload-induced Battery Consumption Rates," *Journal of Intelligent & Robotic Systems*, vol. 97, pp. 471–487, 2020.

[11] M. Rinaldi, S. Primatesta, G. Guglieri and A. Rizzo, "Auction-based task allocation for safe and energy efficient UAS parcel transportation," *Transportation Research Procedia*, vol. 65, pp. 60–69, 2022.

[12] K. Dalamagkidis, K.P. Valavanis and L.A. Piegl, "On unmanned aircraft systems issues, challenges and operational restrictions preventing integration into the National Airspace System," *Progress in Aerospace Sciences*, vol. 44, pp. 503–519, 2008.

[13] S. Primatesta, A. Rizzo and A. la Cour-Harbo, "Ground risk map for unmanned aircraft in urban environments," *Journal of Intelligent & Robotic Systems*, vol. 97, pp. 489–509, 2020.

[14] S. Primatesta, G. Guglieri and A. Rizzo, "A risk-aware path planning strategy for UAVs in urban environments," *Journal of Intelligent & Robotic Systems*, vol. 95, pp. 629–643, 2019.

[15] M. Milano, S. Primatesta and G. Guglieri, "Air Risk Maps for Unmanned Aircraft in Urban Environments," *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1073–1082, 2022.

[16] Z. Liu, R. Sengupta and A. Kurzhanskiy, "A power consumption model for multi-rotor small unmanned aircraft systems," *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, Miami, FL, USA, 2017, pp. 310–315.

[17] C. Byrne, "Sequential unconstrained minimization algorithms for constrained optimization," *Inverse Problems*, vol. 24, pp. 015013, 2008.

[18] S. Dong, "Methods for Constrained Optimization," Massachusetts Institute of Technology, Cambridge, Mass, USA, 2006.

[19] J.C. Lagarias, J.A. Reeds, M.H. Wright and P.E. Wright, "Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions,". *SIAM Journal on Optimization,* vol. 9, pp. 112–147, 1998.