An Approach Combining Consensus with Optimization for Distributed Multi-Robot Task Allocation with Limited Communications

Mohamad Ali Raad, François Guérin and Dimitri Lefebvre

***Abstract*— This paper presents a distributed approach for Multi-Robot Task Allocation (MRTA). We consider assigning sequence of tasks to a group of robots with the aim of maximizing the total received reward while respecting resource constraints. For this, a distributed approach for multi-robot task allocation was developed modeling the problem as a Vehicle Routing Problem (VRP). The approach uses a combination of the Consensus Alternating Direction Method of Multipliers (C-ADMM) with Column Generation (CG) and branch-and-bound to arrive into final assignment through communicating a limited amount of information between the robots. The simulations performed demonstrate the efficiency of the approach by achieving a small relative error with respect to the centralized solution.**

I. INTRODUCTION

The Multi-Robot Task Allocation (MRTA) attracted the interest of research several years ago. Situations like package delivery [1] search and rescue [2] monitoring [3] and transportation [4] are fields where the MRTA arises and needs solutions. Researchers addressed this problem in centralized and distributed contexts. While centralized approaches collect all data at a central agent, distributed ones do not rely on a central agent, which makes them more robust to failure and scale well with the number of tasks and robots.

The MRTA allocation can take several types of formulations. Among these, there is the Vehicle Routing Problems (VRP) which are NP-hard problems and deal with assigning sequence of tasks to robots. This work considers formulating the MRTA problem as a VRP which best suits the cases where robots should perform tasks in some order or within time windows. Specifically, in a VRP, the objective is to find routes starting from an initial position, passing through intermediate positions, and arriving to a target position while optimizing an objective function value and respecting some constraints (Fig. 1).

The problem consists of assigning a group of tasks which should be performed within some time windows to a group of robots. This makes a VRP-like formulation suitable for the problem in which robots will be assigned to routes containing ordered tasks by their service instants. In this context, the term task refers to the measurement to be done at a node, and

Mohamad Ali Raad is with Université Le Havre Normandie, GREAH Laboratory, 76600 Le Havre, France. mohamad-ali.raad@univ-lehavre.fr
François Guérin is with Université Le Havre Normandie, GREAH Laboratory, 76600 Le Havre, France. francois.guerin@univ-lehavre.fr
Dimitri Lefebvre is with Université Le Havre Normandie, GREAH Laboratory, 76600 Le Havre, France. dimitri.lefebvre@univ-lehavre.fr

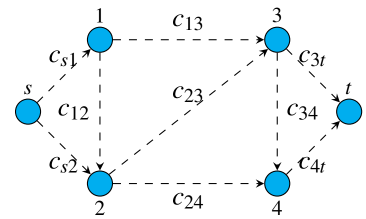

Fig. 1. An example about Vehicle Routing Problem

the term node refers to spatial position where a measurement should be done. All tasks are assumed of the same type.

The considered problem is solved in distributed context where no robot has full knowledge about the problem but only knows a part of the constraints, its own rewards and costs. The proposed approach builds on the method of Consensus Alternating Direction Method of Multipliers (C-ADMM) [5] and combines it with Column Generation (CG) [6] and branch and bound to arrive to path assignment for robots where all tasks are assigned and served. Each robot communicates with its neighbors the values of its dual variables and the new generated path after Column Generation. This combination allows for reduced data sharing between robots, which is essential especially in radio link communications.

The remainder of this paper is organized as follows. Section II presents the relevant works in the state of the art. In Section III, we define the problem with the initial mathematical model. Section IV presents the detailed theoretical approach, while Section V presents the simulation results. Finally, Section VI concludes the work with future perspectives.

II. RELATED WORKS

The researchers proposed different approaches to solve the MRTA using optimization-based approaches and market-based approaches. In [7] the authors provided a comprehensive review of the MRTA problem with its categories, taxonomies, and optimization approaches used to solve it, mentioning the pros and cons of each approach. The authors in [8] proposed a centralized approach for MRTA in which the Genetic Algorithm with a novel chromosome encoding method was used to solve a Mixed Integer Linear Programming (MILP) formulation of the allocation problem. The authors in [9] proposed an approach based on Hybrid Filtered Beam Search (HFBS) to solve task allocation for monitoring patrols using a global and a local filter to select the most promising nodes. In another work [10], the authors addressed the problem of multi-robot routing in dynamic environments to give a reconfigurable approach based on HFBS and Dijkstra's algorithm to compute an initial assignment. Then, a reconfiguration method is used to cope with dynamic events periodically where the allocation algorithm is restarted.

Auction-based methods are market-based approaches that rely on a bidding process between agents and an auctioneer. The agent with the best bid wins the targeted task. Auction algorithms were proposed by the authors in [11] to solve the MRTA problem for tasks with deadlines. The authors in [12] used consensus and bidding methods with synchronous transmission to solve the assignment problem while minimizing information exchange. Behavior trees were synthesized distributively for each agent using 2 level auction algorithms in [13] to solve the task assignment problem and second level auctioning was performed if an agent delegates a task.

The optimization-based approaches formulate the MRTA problem as Integer Linear Programming (ILP) or MILP where decision variables are integer variables that represent if a task is assigned or not. Distributed versions of the branch and price algorithms were proposed in [14] and [15]. The latter solves the problem in an asynchronous way, but requires the communication of the candidate optimal basis and the label of the problem being solved in the tree between the neighboring robots. The MRTA problem was addressed in [16] and [17] by means of a distributed cutting planes algorithm. The Alternating Direction Method of Multipliers was used by the authors in [5] and [18] with consensus to solve the MRTA. Specifically, it proposes three algorithms that differ in the communicated information between robots and give different results in terms of storage, communication, and computational complexity.

To this end, the presented work comes with the following contributions. First, it addresses the VRP formulation of the MRTA in a distributed setting and proposes a new approach to solve it. In addition, solutions very close to the centralized ones with a small error margin are guaranteed. Another positive side of the approach is the limited required data to be shared over the network while also preserving the privacy of each robot's assignment due to the integration of the C-ADMM in the algorithm. Moreover, the developed approach can be adapted to a variety of applications and problems. As shown later, the proposed algorithm can be used to allow each robot to choose its final node without being specified in advance.

III. Problem Formulation

We will adopt a Vehicle Routing Problem (VRP)-like formulation to assign a sequence of tasks to robots. The objective is to maximize the total reward gained by all robots by a Max sum objective function subjected to constraints ensuring the execution of all the tasks, within their time windows while respecting the energy limits of the robots. Servicing a task consumes an amount of energy from the robot's energy, and the robot gets a reward for servicing it. Consider V as the set containing the m nodes (defined by spatial coordinates and each having a unique ID $i \in V$) representing the positions to be visited excluding the starting position S (also called depot) and the target position T. Let $\mathscr{D}$ be the set of all robots. Consider also $W = \{(i,\ j) |\ i \in \{S\} \cup V,\ j \in V \cup \{T\}\}$ representing the set of edges that connect the nodes. The decision variable z_{ij}^d takes the following values; $z_{ij}^d = 1$ if node j is visited after node i by robot d, otherwise $z_{ij}^d = 0$. Let $z = [z_{ij}^d],\ \forall\,(i,\ j) \in W\ \forall d \in \mathscr{D}$. The travel time from node i to node j is represented by t_{ij} and t_r represents the service time (duration) required by any node. The terms e_i and l_i are the earliest and the latest instants to service node i respectively. The decision variable s_i^d indicates the instant when the robot d starts to service the node i. It has a meaningful value only if $z_{ij}^d = 1$. The term $M_{ij} = l_i — e_j$ is a scalar to ensure that the values of s_i^d are bounded between e_i and l_i. The terms c_{ij}^d and r_{ij}^d are the cost (representing the energy consumption for the transition and that of the service) and the reward to service node j coming from node i by the robot d respectively. The travel times between the nodes are directly proportional to the distance between them. Nodes far from each other require more energy (thus higher costs). In this paper, rewards are defined in correspondence with costs; edges with lower costs are associated with higher rewards. The robot's budget B represents the energy limit. The problem can be formulated as follows:

$$\max_{z} \sum_{d\in\mathscr{D}} \sum_{(i,j)\in W} r_{ij}^d\, z_{ij}^d \tag{1}$$

$$\text{subject to} \sum_{d\in\mathscr{D}} \sum_{i\in V} z_{ij}^d = 1 \quad \forall j \in V \tag{2}$$

$$\sum_{d\in\mathscr{D}} \sum_{j\in V\bigcup\{T\}} z_{ij}^d = 1 \quad \forall i \in V \tag{3}$$

$$\sum_{j:(S,j)\in W} z_{Sj}^d = 1 \quad \forall d \in \mathscr{D} \tag{4}$$

$$\sum_{j:(j,T)\in W} z_{jT}^d = 1 \quad \forall d \in \mathscr{D} \tag{5}$$

$$\sum_{j:(i,j)\in W} z_{ij}^d — \sum_{j:(j,i)\in W} z_{ji}^d = 0 \quad \forall i \in V, \forall d \in \mathscr{D} \tag{6}$$

$$\sum_{(i,j)\in W} c_{ij}^d z_{ij}^d \le B \quad \forall d \in \mathscr{D} \tag{7}$$

$$s_i^d + t_{ij} + t_r — M_{ij}(1 — z_{ij}^d) \le s_j^d, \quad \forall i, \forall j \in V, \forall d \in \mathscr{D} \tag{8}$$

$$e_i \le s_i^d \le l_i \quad \forall i \in V, \forall d \in \mathscr{D} \tag{9}$$

$$z_{ij}^d \in \{0,1\} \quad \forall d \in \mathscr{D}, \forall (i,j) \in W. \tag{10}$$

$$s_i^d \in \mathbb{Z}^+ \quad \forall d \in \mathscr{D}, \forall i \in V. \tag{11}$$

Constraints (2) and (3) together ensure that each node will be assigned exactly once and only to one robot. Constraint (4) ensures that each robot starts from its depot and goes exactly to one node. Constraint (5) ensures that each robot arrives from exactly one node to the target node. Constraint (6) ensures that each node has exactly one entry edge and one exit edge (equivalent to the constraint to move in a circuit if the target node is the same as the source node and the robots should return to the depots). Constraint (7) ensures that the total cost for servicing the assigned nodes is at most equal to the budget of the robot. Constraint (8) is for sub-tour elimination and (9) ensures that each node is serviced within its time window.

IV. Theoretical Approach

When the number of optimization variables is large, Column Generation (CG) [6] is a suitable approach to use. We will adopt a modified formulation, where the previous formulation will be transformed into route assignment problem and route generation problem.

A. Column Generation

The CG approach is based on splitting the problem into a Restricted Master Problem (RMP) and a pricing problem. Rather than working with all variables, CG proposes to work with a subset of the variables, thus working with the RMP. After solving the RMP, the pricing problem is solved to add new columns to the problem if they will improve the value of the objective function. The RMP is given by:

$$\max_{\mathbf{x}} \sum_{d\in\mathscr{D}} \sum_{p\in\tilde{P}_d} r_{dp}x_{dp} \tag{12}$$

$$\text{subject to: } \sum_{d\in\mathscr{D}} \sum_{p\in\tilde{P}_d} a_{ip}x_{dp} = 1 \quad \forall i\in V \tag{13}$$

$$\sum_{p\in\tilde{P}_d} x_{dp} = 1 \quad \forall d\in\mathscr{D} \tag{14}$$

$$x_{dp}\in\{0,1\} \quad \forall p\in\tilde{P}_d, \forall d\in\mathscr{D} \tag{15}$$

where $\tilde{P}_d \subseteq P_d$ with P_d representing the set containing the possible paths for the robot d and $\mathbf{x} = [x_{dp}], \forall p\in\tilde{P}_d, \forall d\in\mathscr{D}$. If the path p is assigned to robot d, then $x_{dp} = 1$. The term $a_{ip} = 1$ if the path p contains node i, and the reward returned to robot d when assigned to the path p is denoted by r_{dp}. Constraint (13) ensures that each node is visited exactly by one robot and only once in a route and (14) ensures that each robot executes exactly one path. When $\tilde{P}_d = P_d$, this represents the Master problem.

The binary constraint imposed on x_{dp} is relaxed. To better suit a distributed context, consider the following formulation:

$$\max_{X} \sum_{d\in\mathscr{D}} A_d X_d \tag{16}$$

$$\text{subject to: } \sum_{d\in\mathscr{D}} A X_d = \mathbf{1_m} \tag{17}$$

$$\mathbf{1}_{|\tilde{P}_d|}^{\top} X_d = 1 \quad \forall d\in\mathscr{D}, \tag{18}$$

$$\mathbf{0}_{|\mathbf{\tilde{P}_d}|} \leq X_d \leq \mathbf{1}_{|\mathbf{\tilde{P}_d}|} \quad \forall d\in\mathscr{D}. \tag{19}$$

with $X_d = [x_{d1}, x_{d2}, \cdots, x_{d|\tilde{P}_d|}]^{\top}$, $X_d\in\mathbb{R}^{|\tilde{P}_d|\times 1}$, $A = [a_{ip}]$, $\forall i\in V, \forall p\in\tilde{P}_d$, $A\in\mathbb{R}^{m\times|\tilde{P}_d|}$, $A_d = [r_{dp}], \forall d\in\mathscr{D}, \forall p\in\tilde{P}_d$, $A_d\in\mathbb{R}^{1\times|\tilde{P}_d|}$, $X = [X_1, X_2, \cdots, X_{|\tilde{P}_d|}]$.

The terms $\mathbf{1_m}\in\mathbb{R}^{\mathbf{m}}$, $\mathbf{1}_{|\mathbf{\tilde{P}_d}|}\in\mathbb{R}^{|\mathbf{\tilde{P}_d}|}$ are vectors with their elements all set to 1, and $\mathbf{0}_{|\mathbf{\tilde{P}_d}|}\in\mathbb{R}^{|\mathbf{\tilde{P}_d}|}$ is a vector with its elements set to 0. The pricing problem for each robot is given by:

$$\max_{z^d} \sum_{(i,j)\in W} (r_{ij} - \pi_j) z_{ij} \tag{20}$$

$$\text{subject to } \sum_{j:(S,j)\in W} z_{Sj} = 1 \tag{21}$$

$$\sum_{j:(j,T)\in W} z_{jT} = 1 \tag{22}$$

$$\sum_{j:(i,j)\in W} z_{ij} - \sum_{j:(j,i)\in W} z_{ji} = 0 \quad \forall i\in V \tag{23}$$

$$\sum_{(i,j)\in W} c_{ij}z_{ij} \leq B \tag{24}$$

$$s_i + t_{ij} + t_r - M_{ij}(1 - z_{ij}) \leq s_j, \quad \forall i, \forall j \in V \tag{25}$$

$$e_i \leq s_i \leq l_i \quad \forall i \in V \tag{26}$$

$$z_{ij}\in\{0,1\} \quad \forall (i,j)\in W. \tag{27}$$

$$s_i\in\mathbb{Z}^+ \quad \forall i\in V. \tag{28}$$

where $z^d = [z_{ij}]$, $\forall (i,\ j)\in W$. Let $[\pi_j\in\mathbb{R}]\ \forall j\in V$, $[\kappa_d\in\mathbb{R}]\ \forall d\in\mathscr{D}$ be the optimal dual solution for the problem in (16)-(19) at the current iteration, with κ_d being the dual associated with (18) for a robot d and π_j is the dual associated with (17) for node j. Consider also $\pi_S = \pi_T = 0$. The reduced cost for a column is given by:

$$\tilde{r}_p = \sum_{(i,j)\in W} (r_{ij} - \pi_j) z_{ij} - \kappa_d$$

After solving the pricing problem, the reduced cost is evaluated for the new generated column. If this column has a positive reduced cost (i.e. $\sum_{(i,j)\in W} (r_{ij} - \pi_j) z_{ij} - \kappa_d > 0$), then, it will improve the value of the objective function and it will be added to the RMP. The procedure is repeated until no more columns with positive reduced costs can be generated.

B. Consensus Alternating Direction Method of Multipliers (C-ADMM)

In order to solve the problem in a distributed context, we choose to use the C-ADMM approach and we will follow the approach proposed in [5]. The binary constraint on the decision variable X_d is relaxed to be a real-valued decision variable.

$$\begin{aligned} \max_{X} \ & \sum_{d=1}^{|\mathscr{D}|} f_d(X_d) \\ \text{subject to } & \sum_{d=1}^{|\mathscr{D}|} \left(\mathbf{A}X_d - \frac{1}{|\mathscr{D}|}\mathbf{1}_m\right) = 0 \\ & \sum_{d=1}^{|\mathscr{D}|} \left(G_d X_d - \frac{1}{|\mathscr{D}|}\mathbf{1}_{|\mathscr{D}|}\right) = 0 \\ & \mathbf{0}_{|\mathbf{\tilde{P}_d}|} \leq X_d \leq \mathbf{1}_{|\mathbf{\tilde{P}_d}|} \end{aligned} \tag{29}$$

where $G_d\in\mathbb{R}^{|\mathscr{D}|\times|\tilde{P}_d|}$. We set the row with index d of G_d to $\mathbf{1}_{|\tilde{P}_d|}^{\top}$ with all other entries set to zero. In our case, the objective function of each robot is given by: $f_d(X_d) = A_d X_d$.

The Lagrangian function for (29) is assumed to have a saddle point. The Lagrangian of (29) is given by:

$$\mathscr{L}_p(X_d,\alpha,\beta)=\sum_{d=1}^{|\mathscr{D}|} f_d\left(X_d\right)—\alpha^{\top}\sum_{d=1}^{|\mathscr{D}|}\left(AX_d—\frac{1}{|\mathscr{D}|}\mathbf{1}_m\right) —\beta^{\top}\sum_{d=1}^{|\mathscr{D}|}\left(G_dX_d—\frac{1}{|\mathscr{D}|}\mathbf{1}_{|\mathscr{D}|}\right) \tag{30}$$

with dual variables $\alpha\in\mathbb{R}^m$ and $\beta\in\mathbb{R}^{|\mathscr{D}|}$. From the Lagrangian, we obtain the dual problem given by:

$$\min_{\alpha,\beta}\sum_{d=1}^{|\mathscr{D}|}\varphi_d(\alpha,\beta) \tag{31}$$

where

$$\varphi_d(\alpha,\beta)=\sup_{X_d}\left\{f_d\left(X_d\right)—\alpha^{\top}\left(AX_d—\tfrac{1}{|\mathscr{D}|}\mathbf{1}_m\right) —\beta^{\top}\left(G_dX_d—\tfrac{1}{|\mathscr{D}|}\mathbf{1}_{|\mathscr{D}|}\right)\right\} \tag{32}$$

with $\mathbf{0}_{|\tilde{\mathbf{P}}_\mathbf{d}|}\leq X_d\leq\mathbf{1}_{|\tilde{\mathbf{P}}_\mathbf{d}|}$. According to [5], to derive a distributed method, local copies of α and β are assigned to each robot. The problem is described by:

$$\begin{aligned}&\min_{\alpha,\beta}\sum_{d=1}^{|\mathscr{D}|}\varphi_d\left(\alpha_d,\beta_d\right)\\ &\text{subject to } \alpha_d=\alpha_j,\quad \beta_d=\beta_j\quad \forall j\in\mathscr{N}_d,\forall d\in\mathscr{D}\end{aligned} \tag{33}$$

where $\alpha=\left[\alpha_d^{\top},\forall d\in\mathscr{D}\right]^{\top}$, $\beta=\left[\beta_d^{\top},\forall d\in\mathscr{D}\right]^{\top}$, $\mathscr{N}_d$ represent the neighbors of robot d. From the augmented Lagrangian, and plugging in directly (32), the problem can be represented as:

$$\begin{aligned}\min_{\alpha_d,\beta_d}\max_{X_d}\Bigg\{&f_d\left(X_d\right)—\alpha^{\top}\left(AX_d—\frac{1}{|\mathscr{D}|}\mathbf{1}_m\right)\\ &—\beta^{\top}\left(G_dX_d—\frac{1}{|\mathscr{D}|}\mathbf{1}_{|\mathscr{D}|}\right)+\zeta_d^{k\top}\alpha_d\\ &+\Lambda_d^{k\top}\beta_d+\rho\sum_{j\in\mathscr{N}_d}\left\|\alpha_d—\frac{\alpha_d^k+\alpha_j^k}{2}\right\|_2^2\\ &+\rho\sum_{j\in\mathscr{N}_d}\left\|\beta_i—\frac{\beta_d^k+\beta_j^k}{2}\right\|_2^2\Bigg\}\\ &\mathbf{0}_{|\tilde{\mathbf{P}}_\mathbf{d}|}\leq X_d\leq\mathbf{1}_{|\tilde{\mathbf{P}}_\mathbf{d}|}\end{aligned} \tag{34}$$

where $\zeta_d\in\mathbb{R}^m$ and $\Lambda_d\in\mathbb{R}^{|\mathscr{D}|}$ denote Lagrange multipliers for the equality constraints in (33). The problem in (34) can be written as:

$$\begin{aligned}\min_{\alpha_d,\beta_d}\max_{X_d}\Bigg\{&f_d\left(X_d\right)+\rho\left|\mathscr{N}_d\right|\left\|\alpha_d—\psi_d^k\left(X_d\right)\right\|_2^2\\ &—\rho\left|\mathscr{N}_d\right|\left\|\psi_d^k\left(X_d\right)\right\|_2^2\\ &+\rho\left|\mathscr{N}_d\right|\left\|\beta_d—\omega_d^k\left(X_d\right)\right\|_2^2\\ &—\rho\left|\mathscr{N}_d\right|\left\|\omega_d^k\left(X_d\right)\right\|_2^2+\gamma_d^k\Bigg\}\\ &\mathbf{0}_{|\tilde{\mathbf{P}}_\mathbf{d}|}\leq X_d\leq\mathbf{1}_{|\tilde{\mathbf{P}}_\mathbf{d}|}\end{aligned} \tag{35}$$

with

$$\begin{aligned}\psi_d^k\left(X_d\right)&=\frac{1}{2\rho\left|\mathscr{N}_d\right|}\Bigg(AX_d—\frac{1}{|\mathscr{D}|}\mathbf{1}_m\\ &\qquad—\zeta_d^k+\rho\sum_{j\in\mathscr{N}_d}\left(\alpha_d^k+\alpha_j^k\right)\Bigg)\\ \omega_d^k\left(X_d\right)&=\frac{1}{2\rho\left|\mathscr{N}_d\right|}\Bigg(G_dX_d—\frac{1}{|\mathscr{D}|}\mathbf{1}_{|\mathscr{D}|}\\ &\qquad—\Lambda_d^k+\rho\sum_{j\in\mathscr{N}_d}\left(\beta_d^k+\beta_j^k\right)\Bigg)\end{aligned} \tag{36}$$

and γ_d^k contains the terms that are independent of the dual variables. The Lagrangian has a saddle point enabling us to swap the order of the optimization problems to solve (35). Thus, α_d and β_d are given by:

$$\begin{aligned}\alpha_d^{k+1}&=\psi_d^k\left(X_d^{k+1}\right)\\ \beta_d^{k+1}&=\omega_d^k\left(X_d^{k+1}\right)\end{aligned} \tag{37}$$

Now, X_d can be computed by solving:

$$\begin{aligned}\max_{X_d}\{&f_d\left(X_d\right)—\rho\left|\mathscr{N}_d\right|\left\|\psi_d^k\left(X_d\right)\right\|_2^2\\ &—\rho\left|\mathscr{N}_d\right|\left\|\omega_d^k\left(X_d\right)\right\|_2^2+\gamma_d^k\Big\}\\ &\mathbf{0}_{|\tilde{\mathbf{P}}_\mathbf{d}|}\leq X_d\leq\mathbf{1}_{|\tilde{\mathbf{P}}_\mathbf{d}|}\end{aligned} \tag{38}$$

Robot d updates ζ_d^{k+1} and Λ_d^{k+1} using (39):

$$\begin{aligned}\zeta_d^{k+1}&=\zeta_d^k+\rho\sum_{j\in\mathscr{N}_d}\left(\alpha_d^{k+1}—\alpha_j^{k+1}\right)\\ \Lambda_d^{k+1}&=\Lambda_d^k+\rho\sum_{j\in\mathscr{N}_d}\left(\beta_d^{k+1}—\beta_j^{k+1}\right)\end{aligned} \tag{39}$$

In this part, each robot computes its assignment variable X_d^k by solving the problem in (38), then updates its dual variables using the forms in (37) and finally updates its Lagrange multipliers using (39). Each robot communicates the values of its local dual variables α_d^k and β_d^k with its neighbors. The procedure is performed in parallel and is repeated until the convergence criteria are met.

C. Branch and Bound

To return to the domain of integer values for X_d, the whole process of column generation and C-ADMM will be embedded in a branch and bound procedure. Once the column generation procedure is terminated, the values of the variables and the value of the objective function are checked. The decision is to branch out on the variable that has the largest fractional part. The algorithm keeps track for an incumbent (the best solution obtained so far). When the solutions are not integers, but the value of the objective function is higher than that of the incumbent, the decision is to branch. If the solutions are integers and the objective function value is also greater than the incumbent, the decision is to update the incumbent to be the current solution. Nodes producing infeasible solutions or having objective function value less than that of the incumbent will be pruned.

D. Algorithm Overview

The algorithm starts by solving the RMP using C-ADMM to get an assignment. Then, each robot solves the pricing problem independently and adds the generated paths to a common set called P^{new}, and the corresponding service times of the nodes in the path to a set Sr. For the paths in this new set P^{new}, each robot computes its rewards. It also computes the costs of the new paths shared by the other robots. If the cost of a path generated by another robot exceeds the robot's budget, the robot adds a constraint to a set of constraints marked as C_d imposing the variable of the corresponding path to 0; i.e. $x_{dp} = 0$. Then, each robot merges the set P^{new} into its already existing set of paths. The column generation process is repeated until no more columns can be added to the RMP. Once column generation is finished, the algorithm starts the branch and bound procedure. Algorithm 1 describes how C-ADMM is used in CG.

Algorithm 1: C-ADMM and CG

```
k = 0
forall d ∈ D do
    Initialize X_d^0, α_d^0, β_d^0, ζ_d^0, Λ_d^0
end
while new columns can be generated do
    while not converged or k ≤ max iterations do
        k = k + 1
        forall d ∈ D do
            Compute X_d^{k+1} using (38)
            Compute α_d^k, β_d^k using (37)
            Compute ζ_d^k, Λ_d^k using (39)
        end
    end
    forall d ∈ D do
        Solve the pricing problem using (20)
        Add generated path to set P^new
        Share generated path with neighbors
    end
    forall d ∈ D do
        forall p ∈ P^new do
            Compute payoff of the generated path r_dp
            Compute cost of the generated path c_dp
            if c_dp > budget then
                Add constraint x_dp = 0 to C_d
            end
        end
        P_d = P_d ⋃ P^new
        Update A, A_d, G_d
    end
    P^new = ∅
end
```

V. Numerical Simulations

We study the case of the characterization and dispersion of smoke plumes with complex composition during an industrial fire. For that, a swarm of UAVs is used to measure air quality or take samples with embedded canister at different points in space [19], [20]. Consider the following modifications:

$$W = \{(i,\ j) | i,\ j\ \in\ \{S\} \bigcup V\}$$

We remove (22) and (23) is modified to be:

$$-1 \leq \sum_{j:(i,j)\in W} z_{ij} - \sum_{j:(j,i)\in W} z_{ji} \leq 0 \quad \forall i \in V \tag{40}$$

We will add also the following constraints to the pricing problem:

$$\sum_{j:(i,j)\in W} z_{ij} \leq 1 \quad \forall i \in V \tag{41}$$

$$\sum_{i:(i,j)\in W} z_{ij} \leq 1 \quad \forall j \in V \tag{42}$$

As long as the robot's budget to visit points is enough, it should not return to the depot; it will stay at the last visited point until new tasks appear. In this case, we update the temporary depot of the robot to be its last visited node.
The costs are defined as follows: $c_{ij}^k = 1$ EU (energy unit) if the nodes i and j are adjacent (positioned directly beside each other), else $c_{ij}^k = c_{ij'}^k + c_{j'j}^k$ where the node j' is intermediate between the nodes i and j with $(i,j) \in W$. The response time to make a measurement is set to $t_r = 2$ TU expressed in time unit. The travel time between every two adjacent nodes was set to $t_{ij} = 1$ TU and for non-adjacent nodes it was set proportional to the distance between the two nodes in a sense respecting the triangular inequality assuming constant speed for transitions.

A. Simulation Results

We perform all simulations on laptop with Intel(R) Core(TM) i5-13600H 2.80 GHz processor with 32.0 GB RAM. We used CVXPY as a modeling language with Python programming language. We tested the proposed approach with 2, 3 robots, with 8 and 12 tasks and using 4 robots with 12 and 16 tasks. Table I shows the relative error with respect to the centralized solution of the problem. We evaluated the performance of the proposed method in terms of the number of generated routes, the total time spent, and the number of communication rounds needed for convergence (Table II).

TABLE I
Relative error with respect to centralized solution

Case (#Robots, #Tasks)	Relative error
(2, 8)	0.085 %
(2, 12)	0.588 %
(3, 8)	0.0944 %
(3, 12)	0.3734 %
(4, 12)	0.0117 %
(4, 16)	0.00375 %

TABLE II

AVERAGE AND STANDARD DEVIATION OF NUMBER OF GENERATED PATHS, COMMUNICATION ROUNDS AND TIME LEFT FOR EACH SCENARIO

Case (#Robots, #Tasks)	Generated Paths (Avg − Std)	Communication Rounds (Avg − Std)	Elapsed Time (Avg − Std)
(2, 8)	104.9 − 117.773	553.0 − 587.549	5.893 − 2.699
(2, 12)	121.0 − 74.793	629.8 − 370.523	6.583 − 4.106
(3, 8)	278.9 − 246.766	1494.3 − 1263.965	22.293 − 10.286
(3, 12)	213.0 − 184.977	1079.727 − 926.551	19.583 − 10.562
(4, 12)	686.0 − 361.251	3125.9 − 1513.654	232.819 − 336.992
(4, 16)	553.5 − 554.088	2909.7 − 2472.709	172.38 − 136.751

B. Discussion

From Table I, the algorithm achieved a relative error $< 0.6\%$ for all cases. Getting optimal solutions or near optimal ones depends on two factors: the first one is the convergence criteria used for the C-ADMM algorithm, and the second one is the branching scheme which can lead to solutions far from optimality. Apparently from Table II, the algorithm needs a large number of communication rounds to converge. The number of communication rounds needed for convergence depends on the graph topology connecting the robots. We considered a fully connected graph with each robot broadcasting its message to other robots. The number of communication rounds needed for convergence is expected to increase as the connectivity of the graph decreases. As we considered an undirected graph connecting the nodes and sought paths having only one specified node (depot), while the target is not specified, there exist a huge number of candidate solutions. The algorithm still works with larger numbers of tasks and robots, but is directly influenced by the graph topology connecting robots and that connecting the nodes. With a greater number of tasks, there may exist a large number of feasible paths which increases the number of iterations needed. A larger number of robots directly impacts the number of communication rounds needed for convergence. Although the incurred computational cost and time increased significantly with the case of 4 robots, we argue that this is also due to the modifications introduced earlier, and we expect the algorithm to perform and scale better with the number of robots and tasks when ignoring the introduced modifications.

VI. CONCLUSION AND FUTURE WORK

We proposed a distributed algorithm for solving MRTA that can be modeled as a VRP using C-ADMM to solve column generation and then branch and bound. Future work will be to propose methods to solve the pricing problem more efficiently and to develop more efficient branching schemes. It is essential to study the effect of information loss and the use of different communication topologies between robots on the performance of the algorithm.

REFERENCES

[1] X. Bai, M. Cao, W. Yan, and S. S. Ge, "Efficient routing for precedence-constrained package delivery for heterogeneous vehicles," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 1, pp. 248–260, 2019.

[2] J. Turner, Q. Meng, G. Schaefer, A. Whitbrook, and A. Soltoggio, "Distributed task rescheduling with time constraints for the optimization of total task allocations in a multirobot system," *IEEE transactions on cybernetics*, vol. 48, no. 9, pp. 2583–2597, 2017.

[3] J. Scherer and B. Rinner, "Multi-robot persistent surveillance with connectivity constraints," *IEEE Access*, vol. 8, pp. 15 093–15 109, 2020.

[4] B. Shirani, M. Najafi, and I. Izadi, "Cooperative load transportation using multiple uavs," *Aerospace Science and Technology*, vol. 84, pp. 158–169, 2019.

[5] O. Shorinwa, R. N. Haksar, P. Washington, and M. Schwager, "Distributed multirobot task assignment via consensus admm," *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 1781–1800, 2023.

[6] P. C. Gilmore and R. E. Gomory, "A linear programming approach to the cutting-stock problem," *Operations research*, vol. 9, no. 6, pp. 849–859, 1961.

[7] H. Chakraa, F. Guérin, E. Leclercq, and D. Lefebvre, "Optimization techniques for multi-robot task allocation problems: Review on the state-of-the-art," *Robotics and Autonomous Systems*, p. 104492, 2023.

[8] H. Chakraa, E. Leclercq, F. Guérin, and D. Lefebvre, "A centralized task allocation algorithm for a multi-robot inspection mission with sensing specifications," *IEEE Access*, 2023.

[9] M. Gam, A. J. Telmoudi, and D. Lefebvre, "Hybrid filtered beam search algorithm for the optimization of monitoring patrols," *Journal of Intelligent & Robotic Systems*, vol. 107, no. 2, p. 26, 2023.

[10] S. Hsaini, R. Ammour, L. Brenner, M. E. H. Charaf, I. Demongodin, and D. Lefebvre, "A reconfiguration method for muti-robot monitoring patrols," *Cybernetics and Systems*, pp. 1–26, 2023.

[11] L. Luo, N. Chakraborty, and K. Sycara, "Distributed algorithms for multirobot task assignment with task deadline constraints," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 3, pp. 876–888, 2015.

[12] P. Mahato, S. Saha, C. Sarkar, and M. Shaghil, "Consensus-based fast and energy-efficient multi-robot task allocation," *Robotics and Autonomous Systems*, vol. 159, p. 104270, 2023.

[13] T. G. Tadewos, L. Shamgah, and A. Karimoddini, "On-the-fly decentralized tasking of autonomous vehicles," in *2019 IEEE 58th Conference on Decision and Control (CDC)*, 2019, pp. 2770–2775.

[14] S. Karaman and G. Inalhan, "Large-scale task/target assignment for uav fleets using a distributed branch and price optimization scheme," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 13 310–13 317, 2008.

[15] A. Testa and G. Notarstefano, "Generalized assignment for multi-robot systems via distributed branch-and-price," *IEEE Transactions on Robotics*, vol. 38, no. 3, pp. 1990–2001, 2022.

[16] A. Testa, A. Rucco, and G. Notarstefano, "A finite-time cutting plane algorithm for distributed mixed integer linear programming," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 3847–3852.

[17] ——, "Distributed mixed-integer linear programming via cut generation and constraint exchange," *IEEE Transactions on Automatic Control*, vol. 65, no. 4, pp. 1456–1467, 2019.

[18] R. N. Haksar, O. Shorinwa, P. Washington, and M. Schwager, "Consensus-based admm for task assignment in multi-robot teams," in *The International Symposium of Robotics Research*. Springer, 2019, pp. 35–51.

[19] B. Berthelot, F. Germain, N. Kerthe, F. Guérin, F. Guinand, E. Petitprez, A. Level, S. A. Ali, S. Le Meur, J. Queron *et al.*, "Autonomous aerial swarm robotics for the management of the environmental and health impact in a post-accident situation," in *21. International metrology congress (CIM 2023)*, 2023.

[20] E. Petitprez, F. Guérin, F. Guinand, F. Germain, and N. Kerthe, "Decentralized coordination of a multi-uav system for spatial planar shape formations," *Sensors*, vol. 23, no. 23, p. 9553, 2023.