2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
September 24–28, 2017, Vancouver, BC, Canada

Simultaneous Task Allocation, Data Routing, and Transmission Scheduling in Mobile Multi-Robot Teams

Eduardo Feo Flushing[1], Luca M. Gambardella[1], Gianni A. Di Caro[2]

***Abstract*— In the context of coordination of mobile multi-robot/agent networked teams, we present an integrated model that simultaneously addresses two problems arising in multi-robot missions: (i) task allocation and task scheduling; and (ii) communication provisioning in the multi-hop mobile ad hoc network built by the team. The integrated model is based on a mixed integer linear programming (MILP) formulation, which is solved in a centralized mode. For the communication part, the model solution outputs data routing policies and data transmission schedules that are aimed to maximize data delivery throughput to/from control centers. The trade-off between task and network performance optimization is strategically controlled. A refinement procedure is defined that allows to further improve communications by also minimizing network delays. We report a computational analysis of the integrated MILP model and an evaluation of the impact of a number of parameters on the trade-off between computational load and quality of the output. Results show that the model is computationally affordable for reasonably sized scenarios, and can effectively balance different performance trade-offs.**

I. Introduction

We consider the use of a *mobile multi-agent team* for performing time-extended missions over a potentially large region. The team features the presence of autonomous physical agents with heterogeneous sensory-motor capabilities, such as robots, humans, and animals. The mission consists of a set of spatially distributed tasks, each associated to a possibly different *mission utility*. In general, tasks can be *non-atomic*. Given a specified time horizon, the team aims to maximize the overall utility by means of a *task allocation and scheduling model* that defines the tasks assigned to each agent, the time when an agent has to start performing each one of its tasks, and the effort (i.e., time duration) that the agent has to spend on each task. An optimal task allocation and scheduling is one that provides a maximal team utility value under the considered mission model. Search and rescue, environment monitoring, and area patrolling are all example scenarios for the mission model that we are considering.

We assume a *centralized mode of operations*, in which management and profiling of the mission is carried out at mission *commander centers* (typically only one). Under this mode, a frequent, bidirectional data exchange (one-to-many and many-to-one) between commander centers and the robots deployed in the field, is essential to effectively monitor the mission and issue plans. Since the presence of a network infrastructure cannot be always guaranteed, a *mobile ad hoc network* needs to be set up and maintained, which makes communications even more complex to deal with. In practice, to *maximize the network throughput* one needs to define *routing paths* for multi-hop data transmissions over the data network and *transmission schedules* for data packets: deciding whom to relay data to and when.

It is apparent that in the aforementioned scenario the overall success of the mission depends both on (i) *task allocation and scheduling*, and (ii) the *capability to ensure bidirectional communications*. When this is the case, a common approach consists in dealing with the two problems in parallel, separately, hoping that the merged solutions would meet all the requirements. Yet the way the agents are deployed and move over time to perform tasks affect their ability to communicate. As such, considering task assignment without accounting for communications would treat communications as second-rank objective, which is not desirable. Conversely, any mobility aimed to support connectivity would affect task utility.

In this work we present an *integrated solution* that *simultaneously* solves the *task allocation and scheduling problem* and defines *data routing and transmission scheduling*. We build on our previous work [4] where we have proposed a mathematical model – a *mixed-integer linear program* (MILP) – for the described mission scenario using a heterogeneous multi-robot team (the problem, termed STASP-HMR, and the model, are briefly presented in Section III). In this previous work we only have addressed the task part; in a related work we considered the inclusion of proximity constraints in the model [3] to support connectivity, but it turned out to be extremely demanding for computation. Here, following a different approach, we extend the MILP with the inclusion of a network optimization component. The extended MILP model is based on the combination of a *multi-commodity flow model* and *policy-based* routing. Experimental results show that it is computationally-affordable.

The joint optimization of the mission plans and communications allows to explicitly account in the MILP for the changes in the connection topology of the network. In turn, this allows to effectively set up time-dependent paths for the data flows in terms of *routing policies* specifying which bulk of data should be forwarded where and when. The MILP's objective is a weighted combination of task

[1] Eduardo Feo Flushing and Luca M. Gambardella are with the Dalle Molle Institute for Artificial Intelligence (IDSIA), Lugano, Switzerland, {eduardo,luca}@idsia.ch

[2] Gianni Di Caro is with the Department of Computer Science, Carnegie Mellon University in Qatar (CMU-Q), gdicaro@cmu.edu.

This work was partially supported by the Swiss National Science Foundation (SNSF) through the National Centre of Competence in Research (NCCR) Robotics.

completion and data communication performance, letting the user strategically establishing their best trade-off.

The *main contribution* of this paper is a MILP formulation of the simultaneous spatial task allocation and scheduling, data routing, and transmission scheduling in mobile heterogeneous multi-robot teams. The communication model is based on an original combination of flow models, routing policies, and delay-tolerant strategies for buffering. Both open space and cluttered environments are accounted for, and both delivery ratio and end-to-end delays are optimized. A computational analysis of the model is reported, evaluating its computational efficiency and performance vs. several parameters (e.g., number of agents and tasks, mission horizon).

The rest of the paper is organized as follows. In Section II we review the relevant literature. A brief overview of the STASP-HMR is presented in Section III. In Section IV we present the communication model together with its mathematical formulation. In Section V we report a computational analysis. Finally, we draw some conclusions in Section VI.

II. Related Work

The problem of providing ad hoc communications in a multi-agent team has been addressed in several different domains such as multi-robot exploration [13], path planning and navigation [5], task allocation and planning [8], and pursuit and evasion [14]. A common way to address the problem is through the dedicated use of a group of agents as *communication providers*, whose only objective is to enable data communication. In another set of works, the agents simultaneously play the role of communication providers and task executors. To this end, the provisioning of communication and planning of the mission are usually considered as integrated issues, similarly to what we propose. However, differently from us, most of these approaches enforce the *continual satisfaction of hard communication constraints* (e.g., in terms of proximity among the agents), that can significantly restrict the ways that the mission can be accomplished. Examples include establishing permanent communication paths between a base station and a group of agents [5], [7], [9], [11], and ensuring global connectivity among the agents [10].

Enforcing continual connectivity is justified in communication-critical scenarios where lack of communication can result in the failure of the mission (e.g., tele-operated robots, real-time image streaming). Instead, in other scenarios, as ours, it can be reasonable to relax the strong requirement of permanent connectivity, allowing intermittent forms of network connectivity. Along this line, some works have adopted flexible connectivity goals such as *periodic connectivity* [8], in which the network can be disconnected during bounded time periods, regaining connectivity at fixed intervals, and *recurrent connectivity* [1], in which the system must regularly become connected and remain in that state for a minimum amount of time. In this way, mission requirements for communication can still be satisfied, while, at the same time, it is possible to enable the system to reach mission performance levels that otherwise would be impossible to achieve under strong connectivity constraints. However, in these works, it is not clear for the user what is the performance gain – if any – due to the relaxed connectivity requirements. Moreover, it is not possible to explicitly control the *trade-off* between connectivity provisioning and mission performance as we do in our MILP model.

III. Spatial task allocation & scheduling in heterogeneous multi-agent teams

In this section, we provide a brief description of the *spatial task allocation and scheduling problem in heterogeneous mobile multi-robot teams* (STASP-HMR), that we use as a reference problem scenario for the provisioning of communication support. This problem has been formally introduced in our previous work [4], [2], with a specific application to search and rescue mission scenarios [3].

The STASP-HMR scenario considers a mission that has been decomposed into a set $\mathcal{T}$ of *spatially distributed, location-dependent tasks*. Individual tasks are independent from each other. Each task τ is characterized by a *workload*, and the complete or partial execution of the workload provides a specific contribution to the system *utility* that is related to its *reward*, indicated with R_τ. The spatial layout of tasks is captured by a *traversability graph* (G) that defines how agents can move between tasks. In this respect, we define a directed traversability graph $G = (\mathcal{T}, E)$, where E contains an arc (i, j) if task j can be scheduled right after task i. In general, the presence of a direct connection between two tasks is based on their spatial distribution (e.g., physical proximity, free or obstructed path).

A team $\mathcal{A}$ of heterogeneous mobile agents – not necessarily limited to robots – is available for the mission. Different agents have different sensory-motor characteristics, that result in different efficiency when performing on the same task. The difference in performance among the agents is modeled through a *task model* $\varphi : \mathcal{A} \times \mathcal{T} \mapsto \mathbb{R}$, that relates the amount of effort (measured as *service time*) to the progress in completion of each agent-task pair. For simplicity, and without loss of generality, the whole mission time is discretized into *mission intervals* of equal length Δ_t seconds; Δ_t is the common *time unit* for the starting, ending, and duration of all tasks. It is apparent that the choice of Δ_t involves a trade-off between the size and the quality of the solutions. Large values imply a small mission time span in number of time steps, but can force some agents to spend unnecessary time on tasks for which they require small amount of time to complete. On the other hand, small values of Δ_t imply a longer mission time span but a more efficient distribution of time among the tasks.

In the STASP-HMR, tasks are allowed to be *non-atomic*. Remarkably, while atomic tasks are assumed to be completed once they are assigned, non-atomic tasks can be carried out incrementally. For instance, this property is typical of staged planning scenarios, where the plans are computed in an iterative manner. At each stage, the model needs to identify the *current completion level* of tasks that still have residual workload from the previous stages. A *completion map* $C_m : \mathcal{T} \mapsto [0, 1]$ expresses the remaining workload that each one of the tasks $\tau \in \mathcal{T}$ still requires; a value of 0 for $C_m(\tau)$ indicates that τ has been completed, and if an agent is assigned to further deal with τ, no additional utility is obtained, which amounts to a waste of time and resources.

Based on the above notions and specifications, the STASP-HMR problem can be stated as follows. Given a set of heterogeneous agents, each characterized by its task performance model, a set of assignable tasks and a traversability graph, and given a *limited time budget* T, a *solution* to STASP-HMR – a mission plan – consists of sequences of tasks, one sequence per agent, that define a task schedule: start and end times for an agent to deal with a specific task. In other words, for each agent, a task ordering and how much effort (i.e., devoted time) each of the selected tasks will receive. Due to the time budget constraint, not all tasks will necessarily be completed. An optimal solution defines plans for each one of the agents that maximize the mission utility.

Under a *linearity assumption* of the task models (φ), we have formulated the STASP-HMR problem by means of the following *mixed-integer linear program* (MILP).

$$\text{maximize} \quad \sum_{\tau \in \mathcal{T}} R_\tau \Phi_\tau \tag{1}$$

subject to

$$\sum_{(0,j)\in E} x_{0j}^k = 1 \qquad \forall k \in \mathcal{A} \tag{2}$$

$$\sum_{(i,0)\in E} x_{i0}^k = 1 \qquad \forall k \in \mathcal{A} \tag{3}$$

$$\sum_{(i,j)\in E} x_{ij}^k = \sum_{(j,i)\in E} x_{ji}^k = y_j^k \qquad \forall k \in \mathcal{A},\ j \in \mathcal{T} \tag{4}$$

$$t_i^k + w_i^k - t_j^k \leq (1 - x_{ij}^k)T \quad \forall k \in \mathcal{A}, (i,j) \in E, i,j \neq 0 \tag{5}$$

$$y_i^k \leq t_i^k, w_i^k \leq T y_i^k \qquad \forall k \in \mathcal{A},\ i \in \mathcal{T} \tag{6}$$

$$\Phi_\tau \leq \sum_{k \in \mathcal{A}} \varphi_k(\tau) w_i^k \qquad \forall \tau \in \mathcal{T} \tag{7}$$

$$0 \leq \Phi_\tau \leq C_m(\tau) \qquad \forall \tau \in \mathcal{T} \tag{8}$$

$$t_i^k,\ w_i^k \in \mathbb{N},\ x_{ij}^k,\ y_j^k \in \{0,1\} \qquad \forall k \in \mathcal{A},\ i,j \in \mathcal{T} \tag{9}$$

The MILP model for the STASP-HMR shown in (1)-(9): makes use of following *decision variables*:

x_{ij}^k: binary, equals 1 if agent k traverses arc $(i,j) \in E$;
y_i^k: binary, equals 1 if agent k is assigned to task $i \in \mathcal{T}$;
Φ_τ: service provided to task $\tau \in \mathcal{T}$ by all agents;
t_i^k: starting time of execution of task $i \in \mathcal{T}$ by agent k;
w_i^k: time assigned to task $i \in \mathcal{T}$ for agent k.

The objective function (1) defines the quality of a mission plan in terms of its utility, quantifying the expected effect of agents' activities over the current state of the completion map C_m. A dummy vertex (denoted by 0) represents starting and ending point of the agents' paths. Graph G is extended with arcs from 0 to each one of the tasks that are initially accessible. Constraints (2-4) ensure path continuity. Constraints (5) eliminate sub-tours and, together with (6), they define the bounds on the variables t and w based on the time budget T. The completion levels of each task are bounded by constraints (7-8). We refer the interested reader to [4] for a detailed description.

IV. Data Generation and Communication Models

We consider two different types of situated elements: *mission robots* ($\mathcal{M}$), and *control centers* ($\mathcal{B}$). Henceforth, we will also use the word *node* to refer to an element of any of these types. All robots are engaged in a mission, which is modeled as a STASP-HMR with $\mathcal{A} = \mathcal{M} \cup \mathcal{B}$. At each time step, in addition to perform the tasks assigned to it, each robot generates a known amount of network traffic load (measured in bytes/sec); this traffic is directed to the set $\mathcal{B}$ of mission control centers. Generated data consists of time-stamped mission and status data, which is used to profile the mission in real-time, or, can be used to issue new plans, if necessary. The control centers also generate data that has to be distributed to the nodes $\mathcal{M}$. This models a typical configuration for mission control.

It is apparent that information flow in the system plays a major role, and as such needs to be supported in a reliable way in the *mobile ad hoc network* (MANET) that needs to be set up and maintained in the robot team. We assume that all the nodes are equipped with a range-limited wireless interface. To overcome range limitations, data can be delivered in a multi-hop modality: data can travel across several nodes before arriving at the control center, and vice versa for the data from the control center. The use of multi-hop routing paths can be effectively used to extend the operational range of the mission; however, setting up and maintaining routes to support real-time data flows between the control center and the mission robot can be very difficult to realize in practice when facing high mobility and/or cluttered environments. Assuming that real-time data gathering is not a strict requirement, we let the system to relax the real-time constraints and operate as a *delay-tolerant network* (DTN). To this end, we consider the possibility of *buffering data packets* at the nodes: any node can temporarily store the data it has to send in a buffer and transmit them at subsequent time instants. The use of buffers allows to implement *opportunistic DTN strategies for multi-hop data exchange* [12].

In the following we show how we include the definition of *routing paths, DTN packet buffer, and data transmission scheduling* in the STASP-HMR model. The aim is to set up and solve a model that jointly addresses and optimizes task-related and communication-related team performance. Defining data routing paths for the nodes and controlling the transmission scheduling of generated and buffered data is a way to perform a fine time-control on how data flows in a multi-hop way to/from the control center.

A. Data Routing and Transmission Scheduling

Let α_i^t be the amount of traffic that each node i generates at time step t. The value of α_i^t – as well as any term related to an amount of data henceforth – is expressed in *flow units*, f_{unit}, which we consider as a measure of bytes/sec.

Multi-hop paths are defined by means of time-dependent *routing policies*; a routing policy relating to time step t indicates, for each node k, the fraction $m_{ij}^k(t)$ of data belonging to robot i that the robot k should relay to a neighbor node j at time t. Routing policies assume that each node i has an associated buffer with a finite capacity β_i.

The definition of routing paths and scheduling of transmissions for data is based on the same time discretization employed in the definition of STASP-HMR, assuming that the *network topology* remains quasi-stationary for the duration of a time step. We assume that the location of a node corresponds to the location of the task that its executes.

At a time t, the network topology $\mathbf{T}^t$ is composed of the set of wireless links that can be reliably used for data transmission. That is, $\mathbf{T}^t \subseteq \mathcal{A} \times \mathcal{A}$, and $(k, l) \in \mathbf{T}^t$ iff k can successfully transmit data to l at time instant t. We do not explicitly model interference and medium access control. Yet we consider that shared wireless channels are necessarily *bandwidth-limited*.

B. Extended MILP formulation for STASP-HMR

In the following, we introduce a set of linear constraints that allow the inclusion of data routing policies and transmission scheduling into the MILP presented in Section III. The extended MILP defines a *multi-objective optimization problem* that considers the original objective of the STASP-HMR (i.e., the *maximization of the collected mission rewards*), as well as the *maximization of the amount of data received at the control centers* at the end of the mission.

For sake of clarity, the extensions to the MILP are presented in three parts. Firstly, the formulation of STASP-HMR is extended with time-indexed variables. Secondly, we introduce a set of variables and linear constraints that allow to describe the varying network topologies along the time span. Finally, we formulate the data transmission scheduling as a *multi-commodity network flow model* over the time-indexed network topologies.

1) Time-indexed model: The first step is to transform the MILP presented in (1)-(9) into a *time-indexed model*. This is necessary to formulate constraints that are related to the deployment of the agents at any time step. To this end, we introduce binary time-indexed helper variables y_{ik}^t that take value 1 if agent k is assigned to task $i \in \mathcal{T}$ at time step t.

In order to define the variables y_{ik}^t, we must enforce the following set of conditions:

$$y_{ik}^t = 1 \Leftrightarrow \left(t_i^k \leq t < t_i^k + w_i^k\right) \qquad \forall k \in \mathcal{A}, i \in \mathcal{T}, t, \tag{10}$$

which is achieved with the following linear constraints:

$$(T - t)\, y_{ik}^t + t_i^k \leq T \qquad \forall k \in \mathcal{A}, i \in \mathcal{T}, t \tag{11}$$

$$(t + T + 1)\, y_{ik}^t - T \leq t_i^k + w_i^k \qquad \forall k \in \mathcal{A}, i \in \mathcal{T}, t \tag{12}$$

$$\sum_{i \in \mathcal{T}} y_{ik}^t = 1 \qquad \forall k \in \mathcal{A}, t \tag{13}$$

$$\sum_t y_{ik}^t \leq T y_i^k \qquad \forall k \in \mathcal{A}, i \in \mathcal{T} \tag{14}$$

$$y_{ik}^t \in \{0, 1\} \qquad \forall k \in \mathcal{A}, i \in \mathcal{T}, t. \tag{15}$$

2) Network Topologies: Recall that the network topology $\mathbf{T}^t$ at time t is defined as the set of wireless links that can be used for data transmission at that time. The binary variables λ_{kl}^t are used to represent the topology $\mathbf{T}^t$; each λ_{kl}^t equals 1 iff the wireless link (k, l) between nodes k and l belongs to $\mathbf{T}^t$. The existence of a link between a pair of agents at t depends on which tasks the agents are assigned to at time t.

We consider two different formulations to express the variables λ_{kl}^t which unavoidably introduce additional complexity. The first one, which we refer to as *parameter-based*, allows to use a parameter to specify for each pair of tasks (specifically their locations) (i, j) whether there could be a link or not, based on the fact that any two agents (k, l) are performing i and j at time t, respectively. The second formulation, which we refer to as *distance-based*, makes use of the Euclidean disk model [6], where the values of λ_{kl}^t depend on whether the relative distances between the two nodes k and l at time t are below a certain threshold.

Note that the distance-based formulation assumes an obstacle-free environment since the inter-node distance is the only factor affecting the possibility of data exchange. On the other hand, the parameter-based formulation allows to consider external factors (e.g., the presence of obstacles, wireless interference). Therefore the latter is more flexible than the former. Our goal is to analyze the trade-off between flexibility and the computational costs of these formulations.

In the parameter-based formulation, we consider ψ_{ij} as the binary parameter that specifies whether there can be a link between any pair of agents performing at the tasks' locations i and j at the same time. Assuming that i and j are the tasks/locations of nodes k and l at time t, respectively, $\psi_{ij} = 1$ is a necessary condition for $\lambda_{kl}^t = 1$:

$$\psi_{ij}\left(y_{ik}^t + y_{jl}^t\right) - \psi_{ij} \leq \lambda_{kl}^t \qquad \forall k, l \in \mathcal{A}, i, j \in \mathcal{T}, t \tag{16}$$

$$\lambda_{kl}^t \in \{0, 1\} \qquad \forall k, l \in \mathcal{A}, t. \tag{17}$$

Note that (16) ensures that, when y_{ik}^t and y_{jl}^t are both equal to 1, λ_{kl}^t can only be 1 if the parameter ψ_{ij} is also 1.

The distance-based formulation assumes that all locations are defined in a Cartesian coordinate system, such as each task i has associated a real-valued position vector $\mathbf{o}_i$.

The distance-based formulation makes use of an additional set of helper vector variables $\mathbf{p}_k^t$, where $\mathbf{p}_k^t = \mathbf{o}_i$ iff node k is assigned to task i at time t. Variables $\mathbf{p}_k^t$ are defined by the linear constraints:

$$\mathbf{p}_k^t = y_{ik}^t \mathbf{o}_i \qquad \forall k \in \mathcal{A}, i \in \mathcal{T}, t. \tag{18}$$

Using variables $\mathbf{p}^t$, real-valued variables d^t representing the inter-node distance are defined:

$$d_{kl}^t = \left\|\mathbf{p}_k^t - \mathbf{p}_l^t\right\|, \tag{19}$$

and the set of wireless links can be expressed using a Euclidean disk model:

$$\lambda_{kl}^t = 1 \Rightarrow d_{kl}^t \leq \psi_{\mathrm{r}} \qquad \forall t, k, l \in \mathcal{A}, \tag{20}$$

where ψ_{r} is the transmission range of the network. This condition can be expressed using linear constraints:

$$d_{kl}^t \leq \psi_{\mathrm{r}} + \bar{\psi}(1 - \lambda_{kl}^t) \qquad \forall t, \forall k, l \in \mathcal{A}, \tag{21}$$

where $\bar{\psi}$ is the maximum distance between two agents.

To linearize (19) we propose the use of linear regression based on a linear least squares fitting method. Specifically, we preprocess the locations of tasks and find a vector $\mathbf{q}$ that minimizes the Euclidean 2-norm $\left\|\|(\mathbf{o}_k - \mathbf{o}_l)\| - \mathbf{q}(\mathbf{o}_k - \mathbf{o}_l)\right\|^2$. Then, $\mathbf{q}$ is used inside the model to define the variables d^t_{kl} as follows:

$$\mathbf{q}|\mathbf{p}^t_k - \mathbf{p}^t_l| \leq d^t_{kl} \quad \forall k, l \in \mathcal{A}. \tag{22}$$

Note that absolute values in MILPs, such as the one in (22), can be easily formulated by linear constraints, which are omitted here due to space limitations.

3) Transmission Scheduling: The formulation for data routing and transmission scheduling between $\mathcal{M}$ and $\mathcal{B}$ is based on a *multi-commodity flow* network model.

In the model for the control of the $\mathcal{M} \rightarrow \mathcal{B}$ flows, we make use of the decision variables f^t_{mkl}, q^t_{mk}, and u^t_{mk} that jointly define the transmission policy related to these data flows. Variables f^t_{mkl} represent the amount of data that has been generated by $m \in \mathcal{M}$ and that is transmitted from node k to node l at time t. Variables q^t_{mk} indicate the amount of data that has been generated by $m \in \mathcal{M}$ and that is stored at node k at time t. Finally, since the storage of the nodes can be limited, we let u^t_{mk} be the amount of data generated by m that is dropped at node k at time t.

The data flows $\mathcal{M} \rightarrow \mathcal{B}$ must satisfy the constraints:

$$q^{t-1}_{mk} + \sum_{l \in \mathcal{A}, k \neq l} f^t_{mlk} + \begin{Bmatrix} 0 & \text{if } s \neq k \\ \alpha^t_m & \text{if } m = k \end{Bmatrix} = \sum_{l \in \mathcal{A}, k \neq l} f^t_{mkl} + q^t_{mk} + u^t_{mk} \tag{23}$$

$$q^0_{mk} = 0 \tag{24}$$

$$\sum_{b \in \mathcal{B}, l \in \mathcal{A}} f^t_{mbl} = 0 \tag{25}$$

$$f^t_{mkl}, q^t_{mk}, u^t_{mk} \geq 0 \tag{26}$$

$$\forall m \in \mathcal{M}, k \in \mathcal{A}, t.$$

Constraints (23) ensure flow balance at each node, while constraints (24) sets the initial state of the buffers (here assumed to be empty). Constraints (25) serve to ensure that the control centers act as a sink for the data generated by the mission nodes.

In the formulation of the data flows $\mathcal{B} \rightarrow \mathcal{M}$, an additional index to the variables f^t is needed to differentiate between data flows with respect to the destination of the data, which could be any of the mission nodes. Therefore, additional variables, f^t_{bklm}, and q^t_{bkm} $\forall b \in \mathcal{B},\ k, l \in \mathcal{A},\ m \in \mathcal{M}$, are used. Decision variables f^t_{bklm} represent the amount of data that has been generated by control center b, bound to mission node m, that is transferred from node k to node l. Decision variables q^t_{bkm} denote the amount of data that has been generated by control center b, destined to mission node m, that is stored at node k. Due to space limitations, we omit the formulation of data flows $\mathcal{B} \rightarrow \mathcal{M}$. Yet we mention that their formulation must satisfy a similar set of constraints as the ones (23)-(26) corresponding to the $\mathcal{M} \rightarrow \mathcal{B}$ flows.

Finally, assuming that both types of flows share the same buffer and wireless channel, we formulate the buffer and wireless link capacity constraints as follows:

$$\sum_{n \in \mathcal{M}} q^t_{nm} + \sum_{b \in \mathcal{B},\ n \in \mathcal{M} \setminus \{m\}} q^t_{bnm} \leq \beta_m \tag{27}$$

$$\sum_{m \in \mathcal{M}} f^t_{mkl} + \sum_{b \in \mathcal{B},\ m \in \mathcal{M}} f^t_{bklm} \leq l_{kl}\Gamma \tag{28}$$

$$\forall m \in \mathcal{M},\ k, l \in \mathcal{A},\ k \neq l,\ t$$

Constraints (27) ensure that the amount of data in the buffers do not exceed the capacity. Note that we do not set limitations to the buffer of the control centers; the buffer variables q^t_{nb}, for $b \in \mathcal{B}$, represent indeed the amount of data that has been successfully gathered at the control centers until time t. In the same way, the variables q^t_{bnn} are not considered in (27) because they represent the amount of data that has been generated by a control center s, destined to mission node n, that has successfully been received by n until time t. Finally, constraints (28) ensure that flows only traverse links that belong to the network topology, and also set the wireless link capacity (Γ) specified in terms of flow units, that is, the available bandwidth between any pair of nodes.

4) Combining Data Routing and Transmission Scheduling with STASP-HMR: The communication aspects are included in the STASP-HMR by combining the formulation presented in (1)-(9), with additional linear constraints and variables defined in this section. The resulting model becomes a multi-objective problem, in which we seek the maximization of the mission utility and, at the same time, maximizing the data that can be serviced by the communication network.

One way to deal with the multi-objective nature of the problem is to use *scalarization* to generate a single-objective, weighted optimization problem. A parameter δ is used to define the *trade-off between mission utility and the amount of data serviced*. Using scalarization, the objective function of the combined problem becomes

$$\text{maximize } \left(\Phi_{util} + \delta\Phi_{comm}\right), \tag{29}$$

where Φ_{util} results from the normalization of objective function (1) using the total utility: $\sum_{\tau \in \mathcal{T}} R_\tau C_m(\tau)$, and Φ_{comm} is a term that represents the amount of data that is serviced by the network normalized by the total amount of data generated during the mission:

$$\Phi_{comm} = \frac{\sum_{m \in \mathcal{M}, b \in \mathcal{B}} q^T_{mb} + \sum_{b \in \mathcal{B}, m \in \mathcal{M}} q^T_{bmm}}{\sum_{m \in \mathcal{M}, t} \alpha^t_m + |\mathcal{M}| \sum_{b \in \mathcal{B}, t} \alpha^t_b}. \tag{30}$$

The MILP formulation of the combined problem is composed of the objective function (29), subject to (2)-(9), (11)-(15), (23)-(28), and one of the two formulations of the variables λ^t_{kl}: (16)-(17), or (18), (21)-(22), for the parameter-based and the distance-based respectively. The solution of the new MILP including communication constraints consists – in addition to a mission plan for the team – in the definition of *time-indexed data routing policies and transmission scheduling*, specified by the flow variables f^t.

Apart from the linear combination of mission-related and communication-related performance proposed here, several other ways of combining the two terms can be defined. For instance, trying to enforce a minimum communication performance by restricting the term Φ_{comm} to be greater or equal than certain value. However, this may easily result into unfeasible or over-constrained problems.

5) Improving communications via LP-based refinement for routing policies: The solutions obtained in the previous section only address the total *delivery ratio* for the transmitted data. It does not account an important metric that characterizes the quality of delivering the data: the *transmission delays*. Including this aspect in the MILP would probably lead to an overly difficult model from the computational point of view. Instead, here we propose refinements for the routing policies after they have been computed. This second step precisely and efficiently addresses this additional metric.

Note that after computing a solution to the combined problem, a time-indexed set of network topologies $\{\mathbf{T}^t\}$ can be extracted by observing the planned tasks for each node at any single time step. Let D be the normalized amount of network data estimated to be serviced by the routing policies from the combined solution.

Without modifying task assignments, or decreasing the value of D, it is still possible to improve the network performance in terms of transmission delays. We propose an LP that takes the network topologies, the value of D, the nodes' demands α^t, the buffer capacities β, and link capacities Γ, and whose solution consists of a set of refined routing policies that provide the same performance in terms of amount of received data, but are of better quality in terms of transmission delays.

The LP model is mostly composed of the same variables and linear constraints related to the transmission scheduling presented in Section IV-B.3, with the addition of

$$\lambda^t_{kl} = \begin{cases} 1 & \text{if } (k,l) \in \mathbf{T}^t \\ 0 & \text{otherwise} \end{cases} \quad \forall k,l \in \mathcal{A} \tag{31}$$

$$D \leq \Phi_{comm}. \tag{32}$$

Constraints (31) fix the topology variables λ^t to match the topologies that are taken as an input (derived from the original combined solution). Constraints (32) ensure that the new policies deliver an amount of data at least equal to D.

Now, it remains to formulate the objective function of the LP, which can be done in several ways according to the desired optimization criteria. We propose one formulation that considers the *minimization of the transmission delays*:

$$\min \text{DELAYS} = \sum_{m\in\mathcal{M},k\in\mathcal{A},t} q^t_{mk} + \sum_{b\in\mathcal{B},k\in\mathcal{A},m\in\mathcal{M},k\neq l} q^t_{bkm}. \tag{33}$$

Note that we are seeking the minimization of delays by minimizing the occupancy of all buffers along the mission.

V. Experimental Results

In the following, we first describe the instances of the extended STASP-HMR problem that we use as benchmark. Next, we use the benchmark to study the computational performance of the MILP formulation including the communication model. Finally, we evaluate the LP-based refinement for delay minimization on the computed solutions.

A. Experimental setup

We consider problem instances in which the tasks composing the mission are placed according to a grid decomposition of the area. The grid is embedded into a 2D Cartesian plane, and defines the location of each task as the center of a cell. There is a one-to-one correspondence between tasks and grid cells. Robots can only move between adjacent cells that share at least one vertex. We set the communication range ψ_r so as to let two agents communicating only when they are inside the same cell or in adjacent cells.

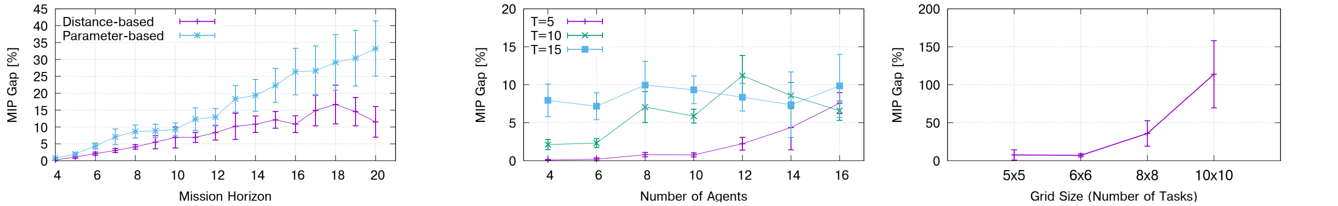
The set of mission agents includes an equal number of two types of agents with different sensory-motor characteristics, which result in different efficiency accomplishing the different tasks. As introduced in Section III, the value of φ precisely relates each robot to the efficiency in doing a specific task. We consider that any robot type exhibits one out of three different efficiency levels for each task, ranging from completely inefficient (requiring sixteen time steps for the robot to complete the task) up to highly efficient (requiring four time steps to fulfill the task).

We consider a scenario where all agents play the role of mission nodes, and one control center is situated in one corner of the grid. Each node generates 1 unit of data at each time step, that need to be relayed to the control center.

B. Efficiency of the formulation

In order to solve our model, we use CPLEX® both as MILP and as LP solver. When solving MILP instances, we also use a simple custom greedy heuristic that provides an initial feasible solution to the solver, with the aim of improving the search of solutions of better quality. The numerical results presented hereafter have been obtained with an AMD Opteron® Processor (2.0 GHz). Only one core was involved in the computation of each solution.

We impose a time limit of one hour and a memory limit of 2 GB. Upon reaching the limits of these computational resources, the solver returns the best solution found. It also provides a measure of the solution quality in terms of a *relative optimality gap*, also called *MIP gap*, that quantifies the estimated distance of the current solution from the optimum. The MIP gap is related to an upper bound that is derived from the linear programming (i.e., relaxation) solution of the subproblems that are automatically generated in the search tree generated by branch and bound and cutting planes procedures. We stop the solver when the gap becomes less than 1%, meaning that we have obtained a satisfactory solution. In practice, controlling the MIP gap, we obtain an *anytime* algorithm .



(a) (b) (c)

Fig. 1. Computational efficiency vs. mission horizon (a), vs. size of the team $\mathcal{A}$ (b), and vs. number of tasks (c). (a) and (b) consider 25 tasks arranged in a grid of size 5×5. (b) and (c) consider the distance-based model.

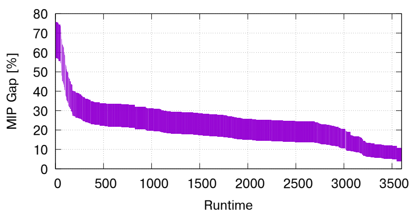


Fig. 2. Run-time distribution of the MIP gaps (95% confidence intervals) among all instances with grid sizes of $\{5 \times 5,\ 6 \times 6\}$.

In the first set of experiments, we consider a grid of size 5×5, that defines missions composed of 25 tasks. Buffer and link capacities are set to a large value so as to ignore their effect in the initial experiments.

We first compare the computational efficiency of the two different formulations, parameter-based and distance-based, that define the variables λ^t. The parameter-based formulation allows to define the wireless links using a binary parameter, while the distance-based one is a more compact formulation but is limited to a Euclidean open-space disk model. We compute solutions to the MILP using both formulations and analyze the MIP gaps returned by the solver. Fig. 1(a) shows the distribution of the MIP gaps over mission horizons ($4 \leq T \leq 20$). The error bars in this and in the other figures, indicate the 95% confidence intervals over a minimum of 50 problem instances. An increasing mission horizon corresponds to a more accurate plan, but also to an increase in problem complexity. Results show that the distance-based formulation is relatively insensitive to the increase of the mission horizon, keeping providing solutions that have a low MIP gap. Instead, the parameter-based formulation shows a quasi linear increase with a relatively steep slope. This is mainly due to the much larger number of constraints that the parameter-based model involves with respect to the distance-based model. Given the observed behavior, we use the distance-based model in the rest of the experiments.

Next, we analyze how the solution quality scales with respect to the number of agents. We consider three mission horizon lengths $T = \{5, 10, 15\}$, and a number of agents ranging from 4 up to 16. Fig. 1(b) shows the resulting distribution of MIP gaps. The results suggest that the model scales quite well with respect to the number of agents, providing solutions with a MIP gap below 10%. In addition, we observe that for longer horizons, increasing the number of agents does not increase the MIP gaps. This suggests that the impact on computational complexity of the horizon is more significant than the one of the number of agents.

So far, we have considered missions with $|\mathcal{T}| = 25$ tasks. Fig. 1(c) shows the impact that larger missions have on the complexity of the problem. Missions have $\mathcal{T}$ defined by grids of sizes $\{5 \times 5,\ 6 \times 6,\ 8 \times 8,\ 10 \times 10\}$, with 50 missions instances per each size. Results indicate that the number of tasks has by far the higher effect over the complexity of the problem with respect to the other parameters considered above. By observing the size of the corresponding MILPs (i.e., in terms of number of variables and number of constraints), we notice that the relative differences among the size of the MILPs is not significant enough so as to justify the enormous impact of the number of tasks over the complexity of the problem. We conjecture that the main reason for this is that a larger grid increases network sparsity, such that the provisioning of communication becomes more difficult.

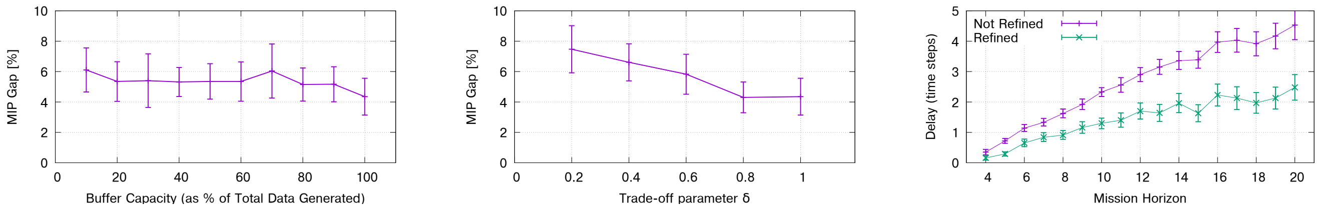
In Fig. 3(a) and Fig. 3(b) we analyze the effect of the buffer capacities and the trade-off parameter δ on the complexity of the problem. Buffer capacities introduce hard constraints on the communication network and force solutions to strike a balance between networking and tasks' utility. Instead, δ directly weights the importance of the communication term in the objective function. We expect the data routing and transmission scheduling decisions have a predominant share of the computational complexity in the combined problem as δ increases.

Fig. 3(a) shows the effect of varying buffer capacities over the complexity of the problem, where the capacity of the buffers are set as the percentage of the total amount of data generated during the mission. The results do not indicate any significant impact of the buffer capacity parameters on the distribution of the MIP gaps. Similarly, we do not observe any significant impact of the parameter δ on the MIP gaps, as shown in Fig. 3(b).

Fig. 2 shows the run-time distribution of the MIP gaps (95% confidence intervals) for grid sizes of $\{5 \times 5,\ 6 \times 6\}$. We observe that after 15 minutes of computation the average MIP gap lower than 25%. This shows that even using a relatively short time, good quality solutions can be easily found.

C. LP improvement of the solutions for delay minimization

We evaluate the performance of the LP-based improvement procedure proposed in Sec. IV-B.5, that allows to



(a) (b) (c)

Fig. 3. Computational efficiency vs. buffer capacity (a), vs. trade-off parameter δ (b), and delay decrease after the LP improvement phase (c). All instances consider 25 tasks arranged in a grid of size 5×5 and the distance-based model. (a) and (b) consider 10 agents and $T = 10$.

separately optimize end-to-end data transmission delays in a second stage. We apply the improvement to each solution computed in the previous part, and simulate the execution of the routing policies – both unrefined and refined ones. Since the improvement is based on an LP, the computation time is no more than a few seconds. In the simulation, at each time step, nodes generate one piece of data. Then, they transfer the data stored in their buffers to the next node as specified by the routing policy for that time step. Preference is given to older packets, so as to simulate buffers following a FIFO policy. Each time a data packet arrives to the control center we measure its delay as the number of time steps passed since the packet was generated. In each scenario, we compute the median value of the delays.

Fig. 3(c) shows the distribution of the delays. From these results, the benefits of post-processing the solutions are apparent. In general, delays tend to increase as the mission horizon is increased. This is likely due to two reasons. First, the quality of the solution decreases for longer horizons as shown in Fig. 3(c), which results into sub-optimal provisioning of networking. Second, as the mission evolves, some tasks may have zero residual workload and, as a result, mission and network topologies become sparser.

VI. Conclusions

Building on previous research, we present a MILP formulation that combines the problem of the spatial task allocation and scheduling problem in heterogeneous mobile multi-robot teams, with the problem of providing communications in the robot MANET to and from a control center. The proposed formulation models and controls communications using a multi-commodity flow approach combined with time-dependent routing policies. The new MILP, in addition to optimize task completion, also allows to define data routes and transmission schedules that maximize the total delivery ratio of the generated data.

Through extensive numerical experiments, we have studied the computational properties of the MILP formulation. The formulation has proven to be computationally affordable for reasonably sized scenarios. We have also proposed an LP-based refinement procedure that allows to minimize network delay in addition to the delivery ratio.

Future work will include further analysis of the model, use of heuristic approaches to speed up computation, and validation of the system using a team of real robots.

References

[1] D. Anisi, P. Ogren, and X. Hu, "Cooperative minimum time surveillance with multiple ground vehicles," *IEEE Trans. on Autom. Control.*, vol. 55, no. 12, pp. 2679–2691, dec 2010.

[2] E. Feo-Flushing, L. Gambardella, and G. A. Di Caro, "On decentralized coordination for spatial task allocation and scheduling in heterogeneous teams," in *Proc. AAMAS*, 2016, pp. 988–996.

[3] E. Feo-Flushing, M. Kudelski, L. Gambardella, and G. A. Di Caro, "Connectivity-aware planning of search and rescue missions," in *Proc. 11th IEEE Int. Sym. on Safety, Security, Rescue Rob. (SSRR)*, 2013.

[4] E. Feo Flushing, L. M. Gambardella, and G. A. Di Caro, "A mathematical programming approach to collaborative missions with heterogeneous teams," in *IEEE/RSJ IROS*, 2014, pp. 396–403.

[5] J. Fink, A. Ribeiro, and V. Kumar, "Robust Control of Mobility and Communications in Autonomous Robot Teams," *IEEE Access*, vol. 1, pp. 290–309, 2013.

[6] S. Gil, S. Kumar, D. Katabi, and D. Rus, "Adaptive communication in multi-robot systems using directionality of signal strength," *Int. J. Rob. Res.*, vol. 34, no. 7, pp. 946–968, jun 2015.

[7] E. Grotli and T. Johansen, "Path-and data transmission planning for cooperating UAVs in delay tolerant network," *Globecom Workshops*, pp. 1568–1573, dec 2012.

[8] G. A. Hollinger and S. Singh, "Multirobot coordination with periodic connectivity: Theory and experiments," *IEEE Trans. Robot.*, vol. 28, no. 4, pp. 967–973, aug 2012.

[9] M. A. Hsieh, A. Cowley, V. Kumar, and C. J. Taylor, "Maintaining network connectivity and performance in robot teams," *J. F. Robot.*, vol. 25, no. 1-2, pp. 111–131, jan 2008.

[10] Z. Mi, Y. Yang, and G. Liu, "Coverage enhancement of mobile multi-agent networks while preserving global connectivity," in *Proc, IEEE Int. Conf. Robot. Autom. (ICRA)*, no. 1, 2011, pp. 5381–5386.

[11] Y. Pei, M. W. Mutka, and N. Xi, "Connectivity and bandwidth-aware real-time exploration in mobile robot networks," *Wirel. Commun. Mob. Comput.*, vol. 13, no. 9, pp. 847–863, jun 2013.

[12] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Efficient Routing in Intermittently Connected Mobile Networks: The Multiple-Copy Case," *IEEE/ACM Trans. Netw.*, vol. 16, no. 1, pp. 77–90, feb 2008.

[13] D. Tardioli, D. Sicignano, L. Riazuelo, A. Romeo, J. L. Villarroel, and L. Montano, "Robot Teams for Intervention in Confined and Structured Environments," *J. F. Robot.*, vol. 33, no. 6, pp. 765–801, 2016.

[14] J. Thunberg and P. Ogren, "An iterative Mixed Integer Linear Programming Approach to pursuit evasion problems in polygonal environments," in *Proc. of ICRA*, 2010, pp. 5498–5503.