A Passivity-Based Decentralized Approach for the Bilateral Teleoperation of a Group of UAVs with Switching Topology

Antonio Franchi, Paolo Robuffo Giordano, Cristian Secchi, Hyoung Il Son, and Heinrich H. Bülthoff

Abstract— **In this paper, a novel distributed control strategy for teleoperating a fleet of Unmanned Aerial Vehicles (UAVs) is proposed. Using passivity based techniques, we allow the behavior of the UAVs to be as flexible as possible with arbitrary split and join decisions while guaranteeing stability of the system. Furthermore, the overall teleoperation system is also made passive and, therefore, characterized by a stable behavior both in free motion and when interacting with unknown passive obstacles. The performance of the system is validated through semi-experiments.**

I. INTRODUCTION

For several applications like surveillance, search and rescue and exploration of wide areas, the use of a group of simple robots rather than a single complex robot has proven to be very effective and the problem of coordinating a group of agents has received a lot of attention by the robotics community (see [1] for a survey). Nevertheless, when the tasks become complex (e.g., the exploration of a very cluttered, possibly unknown, environment for search and rescue applications), complete autonomy is still far to be reached and human's intervention/assistance is necessary.

In this context, teleoperation systems, where an operator drives a remote robot through a local interface, allow to exploit human's intelligence to solve tasks too complex to be solved autonomously by nowadays robots. In particular, it has been widely proven that the use of bilateral teleoperation systems, where a force information is fed back to the user, allows to obtain superior performance with respect to unilateral teleoperation where no feedback is present [2].

The goal of this paper is to study the problem of establishing a bilateral teleoperation system for remotely controlling groups of mobile robots in a distributed way. We focus our attention on flying robots (UAVs), because of their high motion flexibility and potential pervasivity in dangerous or unaccessible locations. However our results may be easily applied to ground, marine, and submarine robots as well. In our envisaged teleoperation system, the UAVs should possess some level of local autonomy and act as a *group*, e.g., by maintaining a desired formation, by avoiding obstacles, and by performing additional local tasks. At the same time, the human operator should be in control of the overall UAV motion and receive, through haptic feedback, suitable cues informative enough of the remote UAV/environment state. A lot of interest is arising in the robotics community in these topics, see, for instance, [3], [4], [5] to name a few. However, this kind of research is still far from being mature under many aspects.

Bilaterally controlling a system where a single master drives multiple slaves is more complex than controlling a traditional single-master/single-slave teleoperation system. In fact, it is not clear what is the best way to dispatch the action of the master to the slaves and what kind of force information to feed back to the master side. In [6], a multi-master/multi-slave teleoperation system with no delay is developed and a centralized strategy for controlling the cooperative behavior of the robots is proposed. On the other hand, for their higher fault tolerance and lower communication demand, distributed approaches are preferred for controlling groups of robots (see, e.g., [1], [7], [8]). In [9], an impedance controller for teleoperating a group of slaves in a leader-follower modality is proposed. In [10], a bilateral control strategy that allows to coordinate the motion between the master and the slaves under arbitrary time delay is proposed. The main limitation of these approaches are the centralization (every robot needs to communicate with the master), and the rigidity of the fleet which is not allowed, for example, to actively reshape the formation or to vary its topology online.

In this respect, this paper proposes a framework to implement a bilateral teleoperation system for controlling remotely a group of UAVs in a highly distributed way. The operator controls motion of the overall fleet and feels its actual motion state and the presence of obstacles. At the same time, the single UAVs behave in an autonomous way by ensuring inter-agent and obstacle collision avoidance, and by adapting online their formation shape and topology via local splitting and merging decisions.

The theoretical foundation on top of which the paper is built is passivity based control: passivity theory is exploited for guaranteeing a stable behavior of the group despite of autonomous maneuvers, time-varying fleet topology, and interaction with remote obstacles in a clean and powerful manner. The rest of the paper is organized as follows: after some preliminary definitions in Sect. II, Sect. III introduces one of the main contributions of the paper, i.e., a passivity-based modeling of the group of UAVs and its interaction with the environment. Then, Sect. IV briefly describes the model of the master device and Sect. V summarizes the

A. Franchi, P. Robuffo Giordano, and H. I. Son are with the Max Planck Institute for Biological Cybernetics, Spemannstraße 38, 72076 Tübingen, Germany {antonio.franchi,paolo.robuffo-giordano, hyoungil.son}@tuebingen.mpg.de.

C. Secchi is with the Department of Science and Methods of Engineering, University of Modena and Reggio Emilia, via G. Amendola 2, Morselli Building, 42122 Reggio Emilia, Italy cristian.secchi@unimore.it

H. H. Bülthoff is with the Max Planck Institute for Biological Cybernetics, Spemannstraße 38, 72076 Tübingen, Germany, and with the Department of Brain and Cognitive Engineering, Korea University, Anamdong, Seongbuk-gu, Seoul, 136-713 Korea hhb@tuebingen.mpg.de.

whole teleoperation system. Finally, results of several semi-experiments are reported in Sect. VI and Sect. VII concludes the paper.

II. PRELIMINARIES

The slave side consists of a group of agents among which a *leader* is chosen. The motion of an agent depends on the motion of the surrounding agents and obstacles. The leader is a special agent that is also controlled by the master. The remaining agents (not controlled by the master) are also referred to as *followers*.

Furthermore, in order to consider the difference between the limited workspace of a master robot and the unbounded one of a UAV, teleoperation is made in the following sense: the position of the master device becomes a velocity setpoint for the leader at the slave side, and the mismatch between the master position and the actual leader velocity is transformed into a force at the master side in order to transmit to the user a feeling of the remote side. We believe this information can provide a feeling of the behavior of the whole fleet since the other robots at the slave side influence the velocity of the leader and, consequently, the force fed back to the user.

III. THE SLAVE SIDE

The slave side consists of a group of robots coupled together. In this section we provide a control strategy for obtaining a flexible cohesive behavior of the group and, at the same time, to avoid self-collisions among the robots. We will show that with the proposed strategy the overall slave side can be modeled as a passive system. In Sect. III-B we will extend this result to also take into account split and join decisions while still preserving passivity.

We consider N agents which can be modeled as floating masses in $\mathbb{R}^3$:

$$\begin{cases} \dot{p}_i = F_i^{\mathrm{a}} + F_i^{\mathrm{e}} - B_i M_i^{-1} p_i \\ v_i = \frac{\partial \mathcal{K}_i}{\partial p_i} = M_i^{-1} p_i \end{cases} \qquad i = 1, \ldots, N \tag{1}$$

where $p_i \in \mathbb{R}^3$ and $M_i \in \mathbb{R}^{3\times 3}$ are the momentum and the inertia matrix of agent i, respectively, $\mathcal{K}_i = \frac{1}{2} p_i^T M_i^{-1} p_i$ is the kinetic energy stored by the agent during its motion, and $B_i \in \mathbb{R}^{3\times 3}$ is a positive definite matrix representing an artificial damping added for asymptotically stabilizing the behavior of the agent and also to represent typical phenomena of aerial robots such as wind/atmospehere drag. Force $F_i^{\mathrm{a}} \in \mathbb{R}^3$ represents the interaction of agent i with the other agents, while $F_i^{\mathrm{e}} \in \mathbb{R}^3$ the interaction of agent i with the environment (obstacles) and the master side through the teleoperation channel. Finally, $v_i \in \mathbb{R}^3$ is the velocity of the agent.

Remark 1: Here we assume that the UAVs are endowed with a Cartesian trajectory tracking controller able to ensure a closed-loop behavior close enough to Eq. (1) (i.e., with small/negligible tracking errors). Many UAV tracking controllers proposed in the past literature, such as [11], [12], could be used in this sense, but we will omit further details since developing/testing a UAV controller is not the scope of this paper.

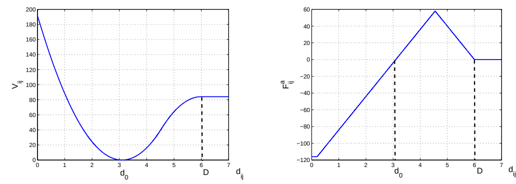


Fig. 1: The shape of the interagent potential as a function of the distance (left), and the corresponding coupling force (right)

From a sensing and communication point of view we assume that two agents are able to communicate and to measure their relative position (i.e., they are *neighbors*) if and only if their distance is less than $D \in \mathbb{R}^+$. Furthermore, an agent can measure the distance from every obstacle which is at distance less than D.

A. A Passive Cooperative Strategy

In order to achieve a collision free, flexible and cohesive behavior of the fleet, we exploit the inter-agent coupling proposed in [8] and inspired to the natural behavior of flocks of animals [13].

Let d_{ij} and $d_0 < D$ be the distance between agent i and agent j, and a desired distance between the agents respectively. For each agent j, agent i computes an interaction force F_{ij}^{a} whose magnitude and direction depends on the relative distance and bearing respectively. In particular, the force is always directed along the bearing: if $d_{ij} < d_0$ a repulsive force is generated; if $d_{ij} = d_0$ a null force is produced; if $d_0 < d_{ij} \leq D$ an attractive force is computed; if $d_{ij} > D$ (i.e., agent i cannot detect agent j) a null force is generated.

This kind of inter-agent potential can be modeled by a nonlinear elastic element (spring) that interconnects a pair of agents. A possible potential function $\bar{V}(d_{ij})$ with such a desired behavior is reported in Fig. 1. Note that the shape of the potential becomes linear as d_{ij} approaches zero for providing a bound of the maximum generated force (e.g., to comply with motor saturations[1]).

The overall force due to the interaction of agent i with the rest of the group is then given by a network of the same nonlinear springs:

$$F_i^{\mathrm{a}} = \sum_{j\neq i} F_{ij}^{\mathrm{a}} := \sum_{j\neq i} \frac{\partial \bar{V}}{\partial d_{ij}} \tag{2}$$

Remark 2: We note that the overall interaction force can be computed by each agent in a distributed way. In fact, the computation is based on the shape of the inter-agent potential (which is known from the design phase), and on the distance and bearing or agent j w.r.t. agent i. Note also that, if agent j is not detected, it is considered as being farther than D and a null force implemented.

Let $x_{ij} := x_i - x_j \in \mathbb{R}^3$ be the relative position of agent i with respect to agent j. The potential reported in Fig. 1 is

[1]Any lower bounded potential (e.g., the one proposed in [10]) can be used for generating the interagent forces. When considering the saturation of the actuators, it is necessary to limit the velocity of two neighboring agents to a value such that any collision can always be avoided by using the maximum available force. This can be passively done by acting on the damping term of each agent.

a lower bounded function of the scalar distance among the agents $d_{ij} = \|x_{ij}\|$ and, as a consequence, a lower bounded function of x_{ij}. As proposed in [14], we model the nonlinear spring representing the interaction between agent i and agent j as:

$$\begin{cases} \dot{x}_{ij} = v_{ij} \\ F^{\mathrm{a}}_{ij} = \frac{\partial V(x_{ij})}{\partial x_{ij}} \end{cases} \tag{3}$$

where $v_{ij} = v_i - v_j$ is the relative velocity of the agents.

Since the forces are symmetric, the interactions of the slave side can be modeled as an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where the vertices represent the agents and an edge (i, j) represents the presence of a spring between agent i and agent j. Defining $p = (p_1^T, \dots, p_N^T)^T \in \mathbb{R}^{3N}$, $B = diag(B_i)$, $x = (x_{12}^T, \dots, x_{1N}^T, x_{23}^T, \dots, x_{2N}^T, \dots, x_{N-1N}^T)^T \in \mathbb{R}^{3\frac{N(N-1)}{2}}$ and $F^e = (F_1^{eT}, \dots, F_N^{eT})^T \in \mathbb{R}^{3N}$, it can be easily seen that the slave side is a mechanical system described by:

$$\begin{cases} \begin{pmatrix} \dot{p} \\ \dot{x} \end{pmatrix} = \left[\begin{pmatrix} 0 & \mathcal{I} \\ -\mathcal{I}^T & 0 \end{pmatrix} - \begin{pmatrix} B & 0 \\ 0 & 0 \end{pmatrix} \right] \begin{pmatrix} \frac{\partial H}{\partial p} \\ \frac{\partial H}{\partial x} \end{pmatrix} + G F^{\mathrm{e}} \\ v = G^T \begin{pmatrix} \frac{\partial H}{\partial p} \\ \frac{\partial H}{\partial x} \end{pmatrix} \end{cases} \tag{4}$$

where

$$H = \sum_{i=1}^{N} \mathcal{K}_i + \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} V(x_{ij}) \tag{5}$$

is the total energy of the system, and $\mathcal{I} = \mathcal{I}_{\mathcal{G}} \otimes I_3$, with $\mathcal{I}_{\mathcal{G}}$ being the incidence matrix of the graph $\mathcal{G}$ whose edge numbering is induced by the entries of the vector x. Furthermore, $G = \left((I_N \otimes I_3)^T \quad 0^T \right)^T$, with I_3 and I_N being the identity matrices of order 3 and N respectively, 0 represents a null matrix of proper dimensions, and $\otimes$ denotes the Krönecker product.

Proposition 1: The system represented in Eq. (4) is passive with respect to the storage function reported in Eq. (5)

Proof:

$$\dot{H} = \begin{pmatrix} \frac{\partial^T H}{\partial p} & \frac{\partial^T H}{\partial x} \end{pmatrix} \begin{pmatrix} \dot{p} \\ \dot{x} \end{pmatrix} \tag{6}$$

and, using Eq. (4) in Eq. (6) and noting that B is positive definite, we obtain that

$$\dot{H} = -\frac{\partial^T H}{\partial p} B \frac{\partial H}{\partial p} + v^T F^{\mathrm{e}} \leq v^T F^{\mathrm{e}} \tag{7}$$

which concludes the proof. ∎

As usual in applications involving mobile agents in unknown environments, we assume that, when they are detected, obstacles are treated as repulsive potentials that produce a force which is null if the robot is far enough and grows as the robot comes closer to the obstacle. Such potentials can also be modeled as virtual springs, that is, passive systems. It is well known that the interaction between two passive systems is stable and, therefore, Proposition 1 implies that the robots at the slave side can safely interact with any passive environment.

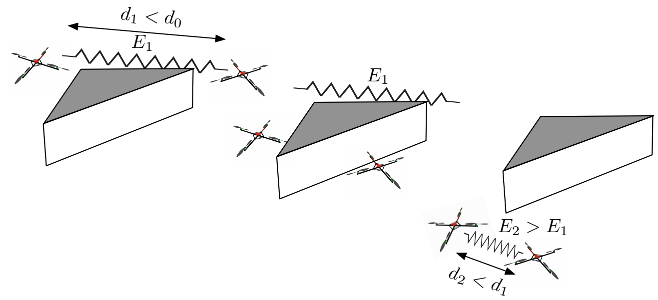


Fig. 2: When the agents split the energy E_1 is stored in the spring, while when they join the energy $E_2 > E_1$ is needed to implement the new desired coupling. In this case, without proper strategies, an amount $E_2 - E_1 > 0$ of energy would be introduced into the system, thus violating passivity.

B. Split and Join While Preserving the Passivity.

We refer to a *split* as the cancelation of the coupling force between a pair of agents i and j even though $d_{ij} \leq D$. A *join* is the (re-)establishment of the coupling, e.g., after a split. Clearly, a join can happen only if $d_{ij} \leq D$. In order to keep the results of this section more general, we will not refer to a particular strategy to determine the split/join between two agents, but we will only focus on how to safely implement them in our context. In the experiments reported in Sect. VI, inter-agent visibility will be adopted as split criterium.

Intuitively, in the framework developed in Sect. III-A, a split between two agents mimics the disconnection from the virtual elastic element that represents their coupling. Thus, the spring becomes isolated and keeps on storing the same energy that was storing before the split decision, while the agents keep on interacting with the rest of the system.

Proposition 2: When two agents split, the passivity of the slave side is preserved.

Proof: If agent i and agent j split, the behavior of the slave side can be described by a subgraph of $\mathcal{G}$, $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$, where $\mathcal{V}' = \mathcal{V}$ and $\mathcal{E}'$ is obtained by $\mathcal{E}$ by erasing the edge connecting vertex i with vertex j. The behavior of the slave side in case of split can be modeled by replacing, in Eq. (4), the incidence matrix $\mathcal{I}$ with a new incidence matrix $\mathcal{I}' = \mathcal{I}_{\mathcal{G}'} \otimes I_3$. The passivity of the system follows from the same arguments of Proposition 1. ∎

Remark 3: The fact that passivity is preserved despite of the change of matrix $\mathcal{I}$ in Eq. (4) depends on the fact that $\mathcal{I}$ enters in the definition of a skew-symmetric matrix which leads to a null term in the energy balance Eq. (7). This is a very powerful property since it allows to seamlessly consider arbitrary time varying topologies in a passive and non destabilizing way.

A join decision, on the other hand, can lead to a violation of the slave side passivity: when two agents i and j join, they instantaneously switch from a state characterized by no interaction, to the inter-agent interaction Eq. (3). This results in a new edge in $\mathcal{E}$, and in a corresponding update of the overall incidence matrix $\mathcal{I}$. While a change in $\mathcal{I}$ does not threaten passivity, some extra energy can still be produced during the join procedure. In fact, in the general case, the relative distance of two agents at the join decision can be different from their relative distance at the split decision, and this can result in a non passive behavior as shown in the illustrative example of Fig. 2 where some extra energy is produced when the agents join.

In order to implement in a passive way the join procedure, we propose to keep track of the energy dissipated by each agent. For each agent, we introduce a local variable, that we call *tank* [5], [15], for storing the energy dissipated. This energy reservoir can then be used for implementing the join without breaking the passivity of the system. Using Eq. (1), it can be seen that the energy dissipated by agent i because of the damping is

$$D_i = p_i^T {M_i^{-1}}^T B_i M_i^{-1} p_i. \tag{8}$$

Considering the tank, we propose to adopt the following extended dynamics for the agents:

$$\begin{cases} \dot{p}_i = F_i^{\mathrm{a}} + F_i^{\mathrm{e}} - B_i M_i^{-1} p_i \\ \dot{t}_i = (1-\beta_i)(\alpha_i \frac{1}{t_i} D_i + w_i) + \beta_i c_i \\ y_i = \begin{pmatrix} M_i^{-1} p_i \\ t_i \end{pmatrix} \end{cases} \tag{9}$$

where $T_i = \frac{1}{2} t_i^2$ is the function representing the amount of energy stored in the tank. The quantity $\alpha_i \in \{0, 1\}$ is a design parameter that disables/enables the storage of the energy dissipated by the system. The quantity $\beta_i \in \{0, 1\}$ is a design parameters which allows to switch the behavior of the tank element between a *storage mode* (i.e., the energy dissipated by the agent is stored) to a *consensus mode* (i.e., a consensus algorithm is run among the tanks). It is easy to see that, when the system is in storage mode ($\beta_i = 0$), we have that:

$$\dot{T}_i = \alpha_i D_i + w_i t_i \tag{10}$$

If $\alpha_i = 1$, all the energy dissipated because of the damping injection on the dynamics of agent i is stored back into the tank. This is the energy that can be "used" in the system without violating the passivity constraint. Because of the reasons reported in [16], it is wise to disable the energy storage for avoiding an excess of energy stored that would allow to implement unstable behaviors in the system. Thus, we set:

$$\alpha_i = \begin{cases} 0 & \text{if } \; T_i \geq \bar{T}_i \\ 1 & \text{otherwise} \end{cases} \tag{11}$$

where $\bar{T}_i$ is a proper bound to be selected depending on the particular application. The input w_i can be used to exchange energy with the tank. In order to avoid singularities in Eq. (9) (i.e., $t_i = 0$), we also set a threshold $\varepsilon > 0$ below which it is forbidden to extract energy from the tank.

When the system switches to consensus mode ($\beta_i = 1$), the term c_i is used for redistributing the energy among the tanks. A distributed strategy is implemented for equally leveling the energy stored in the tanks right before the join. This is done by running a consensus algorithm [1]

$$\dot{T}_i = -\sum_{j \in \mathcal{N}_i} (T_i - T_j) \tag{12}$$

Procedure PassiveJoin

Data: x_{ij}, E_{ij}, $T_j(t_j)$
1 *Compute* $V(x_{ij})$ and $\Delta E = V(x_{ij}) - E_{ij}$;
2 **if** $\Delta E \leq 0$ **then**
3   *Store* $(-\Delta E)/2$ in the tank;
else
4   **if** $T_i(t_i) + T_j(t_j) < \Delta E + 2\varepsilon$ **then**
5   Run a *consensus* on the tank variables;
6   **if** $2T_i(t_i) < \Delta E + 2\varepsilon$ **then**
7   *Dampen* until $T(t_i) + T(t_j) \geq \Delta E + 2\varepsilon$;
8   *Extract* $\Delta E \, T(t_i)/(T(t_i) + T(t_j))$ from the tank;
9 *Join*;

where $\mathcal{N}_i$ indicates the set of neighbors of agent i, namely the set of agents whose distance with agent i is lower than D. The energy redistribution can be implemented acting on the variable t_i. In fact, since $\dot{T}_i = \dot{t}_i t_i$, Eq. (12) is equivalent to setting

$$c_i = -\frac{1}{t_i} \sum_{j \in \mathcal{N}_i} (T_i(t_i) - T_j(t_j)). \tag{13}$$

When agent i and j split, they save in a local variable E_{ij} the amount of energy stored by the elastic element that represents their interconnection just before the split. If agents i and j never split before, E_{ij} is initialized at the value at infinity $\bar{V}_{ij}(\infty) = \bar{V}_{ij}(D)$.

When two agents i and j want to join, the `PassiveJoin` Procedure is run on agent i (and on agent j with proper modifications on the notation). The procedure requires x_{ij} and E_{ij}, which are locally available on agent i, and $T(t_j)$ that can be sent via local communication by agent j. The interaction force corresponding to the joining with agent j is $F_{ij}^a(x_{ij})$ in Eq. (3). Thus, agent i computes $V(x_{ij})$ and the quantity ΔE (line 1). If $\Delta E \leq 0$, the energy needed for implementing the join is lower than the energy previously stored in the spring and, therefore, the join process dissipates energy. Half of the energy dissipated, $(-\Delta E)/2$, can be stored back in the tank of agent i (line 3) and (after the tank update) the agents can safely join (line 9).

If $\Delta E > 0$, extra energy is needed for implementing the decision and, at this point, the energy stored in the tanks is exploited. First, the agents check if there is enough energy in their tanks to cover for ΔE (line 4). If this is the case, part of the energy in the tank of agent i is used for implementing the desired interagent behavior and, therefore, the state of the tank is updated (line 8). In order to keep an energetic balance, the energy extracted from the tank of agent i is:

$$\frac{T(t_i)}{T(t_i) + T(t_j)} \Delta E \tag{14}$$

As before, once the tank is updated the join decision can be safely implemented (line 9).

If the energy stored in the tanks of the two agents is not sufficient, there is still a chance to passively join the agents without intervening directly on the dynamics of the robots. In fact, it may happen that the tanks of the rest of the fleet, in average, contain enough energy. Thus (line 5), agent i asks the fleet to activate β_i in order to switch to consensus mode[2]. Then, the consensus is run until the redistribution of the energy in the tanks is completed. Eventually, all the agents switch back to normal mode ($\beta_i = 0$): all the tanks will contain the same amount of energy, but the total tank energy will remain unchanged. After this redistribution, agents i and j check again if there is enough energy in the tanks for joining (line 6). If this is the case, the tank of agent i is updated as in Eq. (14) and the join decision is implemented (lines 8, 9). If, after all, the energy in the tanks is not yet sufficient, it is necessary to act directly on the robots to refill the tanks. This is always possible by augmenting the artificial damping on the agent for increasing the energy dissipation rate. The damping is augmented to its maximum value (compatible with the saturation of the motors) until $T(t_i) + T(t_j) \geq \Delta E$ and the join decision can be passively implemented (lines 7, 8, 9).

Remark 4: We assume the convergence time of the consensus to be fast enough compared to the dynamics of the fleet for joining the agents and re-establishing the desired behavior as quickly as possible. In fact, if the algorithm is too slow, the agents may come very close to each other without feeling any repulsive force. If the consensus is not fast enough and some dangerous situation is detected, it can be switched off for dampening the system in order to refill the tanks.

Remark 5: When the damping of the agents is augmented, it may take some time to refill the tanks to the desired value of energy. During this period, agents i and j can still move because of the interaction with the rest of the group: in this case, their relative distance d_{ij} and the amount of energy necessary for implementing the join will change. Therefore, it is necessary to continuously update ΔE when the agents are in damping mode.

When two agents decide to join, the behavior of the slave side when the `PassiveJoin` Procedure is implemented can be described by the following system:

$$\begin{cases} \begin{pmatrix} \dot{p} \\ \dot{x} \\ \dot{t} \end{pmatrix} = \left[\begin{pmatrix} 0 & \mathcal{I} & 0 \\ -\mathcal{I}^T & 0 & \mathcal{I}_\gamma^T \\ 0 & -\mathcal{I}_\gamma & 0 \end{pmatrix} - \right. \\ \left. - \begin{pmatrix} B & 0 & 0 \\ 0 & 0 & 0 \\ -(I-\beta)\alpha P B & 0 & 0 \end{pmatrix} \right] \begin{pmatrix} \frac{\partial \mathcal{H}}{\partial p} \\ \frac{\partial \mathcal{H}}{\partial x} \\ \frac{\partial \mathcal{H}}{\partial t} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \beta c \end{pmatrix} + G F^{\mathrm{e}} \\ v = G^T \begin{pmatrix} \frac{\partial \mathcal{H}}{\partial p} \\ \frac{\partial \mathcal{H}}{\partial x} \\ \frac{\partial \mathcal{H}}{\partial t} \end{pmatrix} \end{cases} \tag{15}$$

where

$$\mathcal{H} = \sum_{i=1}^{N} \mathcal{K}_i + \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} V(x_{ij}) + \sum_{i=1}^{N} T_i \tag{16}$$

is the augmented total energy of the system. The matrix $\mathcal{I}_\gamma = \Gamma \circ (\mathbf{1} \otimes \mathcal{I}_\mathcal{G})$, where $\circ$ is the element-wise product, $\mathbf{1} = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}^T$, and Γ is a matrix of proper dimensions whose elements represent an energetic interconnection between tanks and springs. In fact, updating a tank means transferring part of its energy into the elastic element representing the interaction between two agents in order to obtain the desired force. During normal behavior, no energy needs to be transferred from the tanks into the elastic elements and, therefore, all the elements of Γ are set to 0. Suppose now that the tank i needs to transfer some energy into the elastic element connecting agent i to agent j and that the state of the elastic element is in the k^{th} position in the stack of states x. This can be modeled by properly setting $\Gamma_{i,k}$ as proposed in [5] or [15]. Finally, $\alpha = diag(\alpha_i)$ and $\beta = diag(\beta_i)$ are matrices containing the mode switching parameters, $P = diag(\frac{1}{t_i} p_i^T M_i^{-T})$, $t = (t_1, \dots, t_N)^T$, and $c = (c_1, \dots, c_N)^T$.

Proposition 3: The system represented in Eq. (15) is passive.

Proof: Consider as storage function the total energy of the system $\mathcal{H}$. We have that

$$\dot{\mathcal{H}} = \begin{pmatrix} \frac{\partial^T \mathcal{H}}{\partial p} & \frac{\partial^T \mathcal{H}}{\partial x} & \frac{\partial^T \mathcal{H}}{\partial t} \end{pmatrix} \begin{pmatrix} \dot{p} \\ \dot{x} \\ \dot{t} \end{pmatrix}. \tag{17}$$

Using Eq. (15) with Eq. (17) we obtain that:

$$\dot{\mathcal{H}} = -\frac{\partial^T \mathcal{H}}{\partial p} B \frac{\partial \mathcal{H}}{\partial p} + \frac{\partial^T \mathcal{H}}{\partial t} (I-\beta)\alpha P B \frac{\partial \mathcal{H}}{\partial p} + \beta \frac{\partial \mathcal{H}}{\partial t} c + v^T F^{\mathrm{e}} = h_1 + h_2 + h_3 + v^T F^{\mathrm{e}} \tag{18}$$

The system is passive if the sum of the first three terms of Eq. (18) is lower or equal than 0. The first term h_1 is always non positive because B is positive semidefinite. The parameter β can either be equal to the null or identity matrix.

When $\beta = 0$, $h_3 = 0$ and the second term

$$h_2 = \frac{\partial^T \mathcal{H}}{\partial t} \alpha P B \frac{\partial \mathcal{H}}{\partial p} = \begin{pmatrix} t_1 \dots t_N \end{pmatrix} \alpha \begin{pmatrix} \frac{1}{t_1} D_1 \\ \vdots \\ \frac{1}{t_N} D_N \end{pmatrix} = \sum_{i=1}^{N} \alpha_i D_i \tag{19}$$

is, because of Eq. (11), at most equal to the energy dissipated by the agents (first term h_1). Therefore, $\dot{\mathcal{H}} \leq v^T F^{\mathrm{e}}$.

When $\beta = I$, $h_2 = 0$ and the consensus is running among the tanks. By recalling Eq. (13), h_3 can be written as

$$h_3 = \frac{\partial^T \mathcal{H}}{\partial t} c = \sum_{i=1}^{N} \dot{T}_i. \tag{20}$$

Because of the property of the consensus, the overall energy stored in the tanks remains the same and, therefore, $h_3 = 0$ and $\dot{\mathcal{H}} \leq v^T F^{\mathrm{e}}$. ■

IV. THE MASTER SIDE

The master can be a generic mechanical system and it can be modeled by the following Euler-Lagrange equations:

$$M_m(x_m)\ddot{x}_m + C(x_m, \dot{x}_m)\dot{x}_m + D_M \dot{x}_m = F_M \tag{21}$$

where M_m represents the inertia matrix, $C(x_m, \dot{x}_m)\dot{x}_m$ is a term representing the centrifugal and Coriolis effects, D_M

[2]This can be done by using a distributed procedure (e.g., the classic flooding algorithm [17]) so that all the agents belonging to the same connected component of the communication graph set $\beta_i = 1$.

is matrix representing the viscous friction present in the system. As often happens for master devices, we assume that gravity effects are compensated by a local controller. The variables x_m and $\dot{x}_m$ represent the position and the velocity of the end-effector. A system described by Eq. (21) is passive with respect to the force-velocity pair (F_M, v_m) [18], where $v_m := \dot{x}_m$. This kind of passivity is well suited in standard passivity based bilateral teleoperation, where the velocity of the master and the velocity of the slave need to be synchronized.

Nevertheless, in our setting, in order to consider the difference between the workspace of the master and that of the robots at the slave side, it is necessary to synchronize the position of the master with the velocity of the leader. Unfortunately, a mechanical system is not passive with respect to the position-force pair but it is possible to implement a local control loop on the master that makes it passive with respect to the pair (F_M, r_1) with

$$r_1 = v_m + \lambda x_m, \qquad \lambda \in (0, \lambda_{\max}], \tag{22}$$

where $\lambda_{\max}$ can be made arbitrarily large by a proper choice of the damping action D_M in (21), see [19]. Using r_1, at steady state, the master is passive w.r.t. its position but, during transients, the effect of v_m can be non negligible.

To remedy this side-effect, we propose to consider the variable:

$$r_m = \rho r_1(t) = \rho v_m + \rho\lambda x_m, \qquad \rho > 0. \tag{23}$$

The following result then easily follows:

Proposition 4: A mechanical system which has been made passive with respect to the pair (r_1, F_M) is also passive with respect to the pair (r_m, F_M)

Proof: Since the system is passive with respect to the pair (r_1, F_M), there exists a lower bounded function S_m such that

$$r_1^T F_M \geq \dot{S} \tag{24}$$

Using Eq. (23), we have that

$$r_m^T F_M = \rho r_1^T F_M \geq \dot{S} \tag{25}$$

which implies that

$$r_m^T F_M \geq \frac{1}{\rho}\dot{S}. \tag{26}$$

Therefore, the system is passive w.r.t the lower bounded function $\bar{S} = \frac{1}{\rho}S$. ■

Thus, by properly choosing the parameters ρ and λ it is possible to make negligible the contribution related to $\dot{x}_m$ (by choosing a small ρ), and to make the second term proportional to the position with a desired scaling factor K (by choosing $\lambda = \frac{K}{\rho}$)[3].

[3]Note that $\lambda_{\max}$ limits the maximum achievable scaling K to $K_{\max} = \rho\lambda_{\max}$.

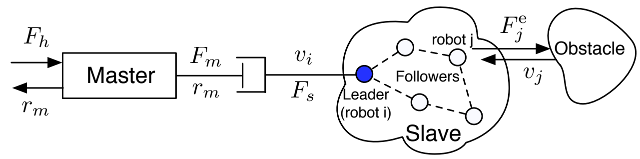


Fig. 3: The overall teleoperation system.

V. THE TELEOPERATION SYSTEM

Exploiting the results developed so far, we have that both master and slave sides are passive systems. Thus, by designing a proper passive interconnection between the local and the remote systems will yield a passive bilateral teleoperation system characterized by a stable behavior in case of interaction with passive environments (as the obstacles, modeled as potentials, with which the fleet is interacting).

Suppose that agent i is chosen as the leader. It is possible to write $F_i^{\mathrm{e}} = F_s + F_i^{\mathrm{env}}$, where F_i^{env} is the component of the force due to the interaction with the external environment (obstacles) and F_s is the component due to the interaction with the master side. Similarly, we can decompose F_M as $F_M = F_m + F_h$, where F_h is the component due to the interaction with the user and F_m is the force acting on the master because of the interaction with the slave.

For achieving the desired teleoperation behavior, we propose to join master and slave using the following interconnection:

$$\begin{cases} F_s = b(r_m - v_i) \\ F_m = -b(r_m - v_i) \end{cases} \tag{27}$$

This is equivalent to joining the master and the leader using a damper which generates a force proportional to the difference of the two velocity-like variables of the master and the leader. Since r_m is "almost" the position of the master, we have that the force fed back to the master and the control action sent to the leader are the desired ones. The overall teleoperation system is represented in Fig. 3 and it consists of the interconnection of a passive master side, a passive interconnection and a passive slave side. Recalling that the interconnection of passive systems is again passive [18], we have that the teleoperation system is passive, as desired.

Remark 6: Intuitively, the followers behave as an environment the leader is interacting with. In case of no followers, the steady-state synchronization error between r and v_i can be made arbitrarily small by suitably choosing a (large) value for b in (27). When some followers are present, the leader interacts with a passive moving environment. Thus, at steady state, the synchronization error increases as in standard bilateral teleoperation (see, e.g., [20]). On the other hand, this creates a beneficial force F_m on the master that allows the user to feel the status of the fleet. Indeed, this force is proportional to the number of followers, to their velocities and to their relative positions.

Since the teleoperation system is the interconnection of a local passive master and of a remote passive slave, it is possible to passively consider communication delays between local and remote sites using one of the techniques developed for single master single slave telemanipulation systems, like, for example, wave variables [20]. In this way, the system

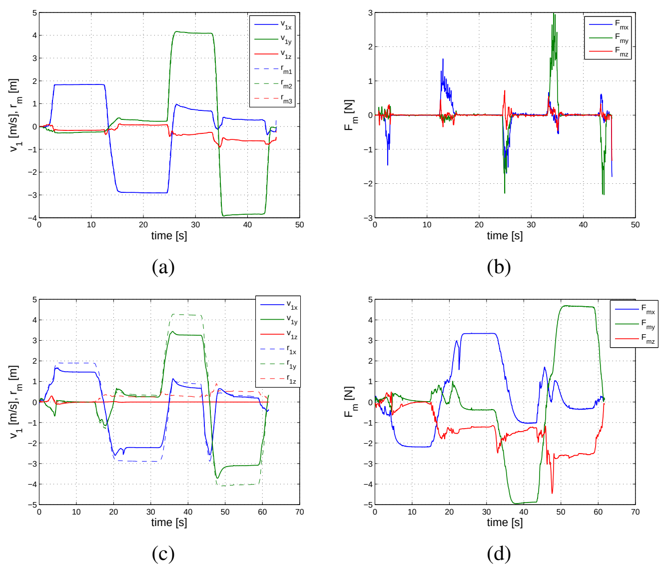


Fig. 4: Results of the first two semi-experiments. Fig. 4(a): superimposed behavior of $v_1(t)$ (the leader velocity) and $r_m(t)$ (the master command) with no followers. Note the almost *perfect match* between the two quantities: the solid and dashed lines are basically superimposed. Fig. 4(b): force $F_m(t)$ applied to the master from the slave. The peaks occur during the small deviations of $v_1(t)$ from $r_m(t)$. Figs. 4(c–d): same as Figs. 4(a–b) when 5 followers are considered. Note the steady state error between $v_1(t)$ and $r_m(t)$ because of the friction exerted by the followers on the leader.

would keep on exhibiting a stable behavior independently of any delay between local and remote site.

VI. SEMI-EXPERIMENTS

In this Section, we will report the result of several semi-experiments conducted to validate the theoretical framework developed so far. We used a commercial haptic device (Omega6, Force Dimension) as a master robot. The Omega6 is a 6-DOF haptic device with 3 translational actuated axes, and 3 rotational non-actuated axes, and its local control loop runs at about 2.5 kHz on a dedicated gnu/linux machine. The UAVs dynamics and control logic, on the other hand, were simulated in a custom-made simulation environment based on the Ogre3D engine (for 3D rendering and computational geometry computations), the PhysX libraries for simulating the physical interaction between the UAVs and the environment, and the MIP framework (http://www.dis.uniroma1.it-/~labrob/software/MIP) for the multi-robot communication and control aspects.

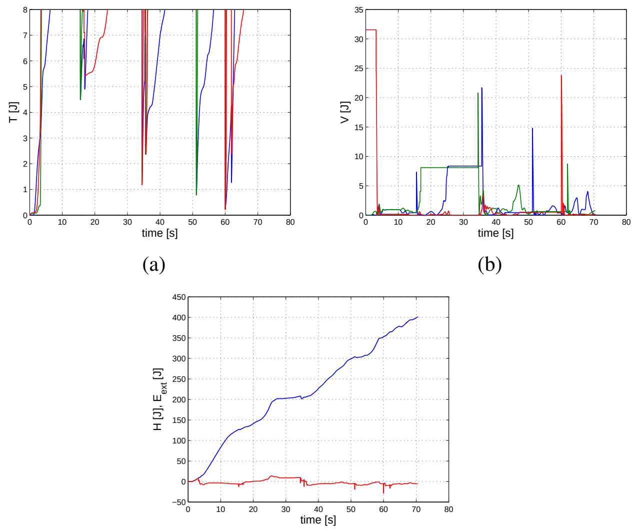
The criterium adopted to decide a split between agents (see Sect. III-B) is visibility: we assume that two agents decide to split whenever their line-of-sight is obstructed by an obstacle (or by another UAV), thus simulating the possible loss of visual/radio connectivity in this kind of situations. Of course, different criteria are possible, but they are equivalent w.r.t. the conceptual behavior of the `PassiveJoin` Procedure. We assume w.l.o.g. that the leader is agent 1.

In the first two semi-experiments, whose results are reported in Figs. 4(a–d), we tested the overall performance of the teleoperation scheme during free-motion (i.e., sufficiently away from obstacles). The goal was to show the stable behavior of our teleoperation system and to point out its force reflection characteristics in a steady-state regime.

In the first case (Figs. 4(a–b)), no followers are considered, i.e., $N = 1$, and the sole leader is teleoperated with piecewise constant velocity commands. Figure 4(a) shows the superimposition of the leader velocity v_1 (solid lines) and the master "position" r_m (dashed lines): as clear from the plot, v_1 and r_m match almost perfectly during the whole operation — the solid lines are basically superimposed on the dashed lines, indicating high coordination between the master command r_m and the slave velocity v_1. Figure 4(b) reports the behavior of F_m over time: one can note that F_m is always negligible apart from some transient peaks occurring during the sudden changes of the commanded velocity r_m. Indeed, in these phases the second-order dynamics (Eq. (1)) of agent 1 naturally lags behind the commanded velocity r_m because of the viscous coupling F_s in (Eq. (27)). These force cues, however, are useful to inform the operator about the (transient) discrepancy between v_1 and r_m.

In the second semi-experiment, we tested the same free-motion scenario but by considering 5 followers coupled with the leader. Figures 4(c–d) report the same quantities of the previous experiment. In this case, we can note in Fig. 4(c) a persistent (and larger) steady-state error between v_1 and r_m during all phases: this is the cumulative effect of the follower's elastic couplings and damping terms B_i which keep dissipating energy during the motion and acts as a major drag on the leader. The force F_m in Fig. 4(d) correctly reproduces this mismatch by displaying to the user a constant force cue opposing to the direction of the motion, thus conveying the information that the user is actually 'pulling a load with friction'. We note again that this steady-state mismatch is the expected behavior of our teleoperation system, see Remark 6.

As last semi-experiment, we report the teleoperation of 3 UAVs (1 leader and 2 followers) moving in an environment cluttered with obstacles, thus enabling the possibility of split and rejoin decisions. The parameters of the inter-agent potential $\bar{V}(d_{ij})$ were set to $d_0 = 0.2$ [m], $D = 6$ [m], and $\bar{V}(D) = 31.5$ [J]. Figures 5(a–b) show the evolution of the 3 tank energy reservoirs T_i, and of the 3 inter-agent potentials $V(x_{ij})$. At the beginning of the experiment, two agents are disconnected (one potential starts at the 'infinity' value $\bar{V}(D)$) and at time $t_1 = 3$ [s] of motion they join getting closer than D. Note that this join decision does not affect the tank behavior (they keep on storing the dissipated energy (Eq. (8))) because $\Delta E \leq 0$ by construction in this case. During the rest of the motion, several split and join decisions are taken, and in particular at times $\{t_2 = 16,\ t_3 = 34,\ t_4 = 51.\ t_5 = 60\}$ [s] the tank energies are used to re-establish the links, i.e., when positive jumps of the inter-agent potentials (with $\Delta E > 0$) are counter-balanced by negative jumps of the tank values. Finally, Fig. 5(c) shows the behavior of $E_{\text{ext}}(t) = \int_{t_0}^{t} v^T(\tau)F^e(\tau)\mathrm{d}\tau$ (blue line) and $E_{\text{in}}(t) = \mathcal{H}(t) - \mathcal{H}(t_0)$ (red line) over time. One can then



(a) (b)

(c)

Fig. 5: Results of the third semi-experiment: Figs. 5(a–b): behavior of $T(t)$ and $V(x_{ij}(t))$ over time during several split and join decisions. Note how the positive jumps in $V(x_{ij}(t))$ correspond to negative jumps in $T(t)$ (energy exchange between tanks and link potentials. Figure 5(c): behavior of $E_{\rm ext}(t)$ (blue line) $E_{\rm in}(t)$ (red line), validating the slave side passivity condition (Eq. (18)) also during the split/join decisions.

check that $E_{in}(t) \leq E_{ext}(t)$, $\forall t \geq t_0$, as required by the slave-side passivity condition (Eq. (18)).

Finally, we also encourage the reader to watch the video clip attached to the paper where a teleoperation of 6 UAVs in a cluttered environment with frequent split and join decisions can be appreciated.

VII. CONCLUSIONS AND FUTURE WORK

In this paper we have proposed a distributed control strategy based on passivity for teleoperating a group of UAVs. By monitoring the exchange of energy among the robots, it is possible to constrain as less as possible the behavior of the fleet which can smoothly change the shape of its formation and also perform split and join actions in a stable way. By properly passifying the master robot, a bilateral teleoperation system that couples the motion of the master to the velocity of the slave side has been proposed.

In the future, we aim at running a comparative analysis of the techniques available in the single-master single slave teleoperation (e.g., wave variables) for determining which one can be better adapted to the proposed multi-slave scenario. Furthermore, we would like to consider more leaders at the slave side in order to have a better control of the motion of the UAVs. Finally, we also plan to passively implement some extra forces at the master side in order to convey some extra information about the connectivity and for improving the telepresence feeling of the user.

ACKNOWLEDGEMENTS

The authors thank Johannes Lächele, Markus Ryll and Volker Grabe for their intensive support in implementing the simulation environment and running the semi-experiments. We also gratefully acknowledge the support of the Max Planck Society and the WCU (World Class University) program through the National Research Foundation of Korea funded by the Ministry of Education, Science and Technology (R31-2008-000-10008-0).

REFERENCES

[1] R. M. Murray, "Recent research in cooperative control of multi-vehicle systems," *ASME Journal on Dynamic Systems, Measurement, and Control*, vol. 129, no. 5, pp. 571–583, 2007.

[2] B. Hannaford, "Stability and performance tradeoffs in bi-lateral telemanipulation," in *1989 IEEE Int. Conf. on Robotics and Automation*, Scottsdale, AZ, May 1989, pp. 1764–1767.

[3] N. Guenard, T. Hamel, and L. Eck, "Control Laws For The Tele Operation Of An Unmanned Aerial Vehicle Known As An X4-flyer," in *2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Oct. 2006, pp. 3249–3254.

[4] R. Mahony, F. Schill, P. Corke, and Y. S. Oh, "A new framework for force feedback teleoperation of robotic vehicles based on optical flow," in *2009 IEEE Int. Conf. on Robotics and Automation*, Kobe, Japan, May 2009, pp. 1079–1085.

[5] S. Stramigioli, R. Mahony, and P. Corke, "A novel approach to haptic tele-operation of aerial robot vehicles," in *2010 IEEE Int. Conf. on Robotics and Automation*, May 2010, pp. 5302–5308.

[6] S. Sirouspour, "Modeling and control of cooperative teleoperation systems," *IEEE Trans. on Robotics*, vol. 21, no. 6, pp. 1220–1225, 2005.

[7] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Trans. on Automatic Control*, vol. 9, no. 3, pp. 1465–1476, 2004.

[8] N. E. Leonard and E. Fiorelli, "Virtual leaders, artificial potentials and coordinated control of groups," in *40th IEEE Conf. on Decision and Control*, Orlando, FL, Dec. 2001, pp. 2968–2973.

[9] Y. Cheung and J. S. Chung, "Cooperative control of a multi-arm system using semi-autonomous telemanipulation and adaptive impedance," in *14th Int. Conf. on Advanced Robotics*, Munich, Germany, Jun. 2009, pp. 1–7.

[10] E. J. Rodriguez-Seda, J. J. Troy, C. A. Erignac, P. Murray, D. M. Stipanovic, and M. W. Spong, "Bilateral teleoperation of multiple mobile agents: Coordinated motion and collision avoidance," *IEEE Trans. on Control Systems Technology*, vol. 18, no. 4, pp. 984–992, 2009.

[11] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro," in *2005 IEEE Int. Conf. on Robotics and Automation*, Barcelona, Spain, May 2005, pp. 2247–2252.

[12] N. Guenard, T. Hamel, and R. Mahony, "A Practical Visual Servo Control for an Unmanned Aerial Vehicle," *IEEE Trans. on Robotics*, vol. 24, no. 2, pp. 331–340, 2008.

[13] I. Couzin, "Collective minds," *Nature*, vol. 445, p. 715, 2007.

[14] C. Secchi and C. Fantuzzi, "Formation control over delayed communication networks," in *2008 IEEE Int. Conf. on Robotics and Automation*, Pasadena, CA, May 2008, pp. 563–568.

[15] C. Secchi, S. Stramigioli, and C. Fantuzzi, "Position drift compensation in port-hamiltonian based telemanipulation," in *2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Beijing, China, Oct. 2006, pp. 4211–4216.

[16] D. J. Lee and K. Huang, "Passive-set-position-modulation framework for interactive robotic systems," *IEEE Trans. on Robotics*, vol. 26, no. 2, pp. 354–369, 2010.

[17] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks*, ser. Applied Mathematics Series. Princeton University Press, 2009, electronically available at http://coordinationbook.info.

[18] C. Secchi, S. Stramigioli, and C. Fantuzzi, *Control of Interactive Robotic Interfaces: a port-Hamiltonian Approach*, ser. Springer Tracts in Advanced Robotics. Springer, 2007.

[19] D. J. Lee and D. Xu, "Feedback r-Passivity of Lagrangian Systems for Mobile Robot Teleoperation," in *2011 IEEE Int. Conf. on Robotics and Automation*, Shanghai, China, May 2011.

[20] G. Niemeyer and J.-J. Slotine, "Telemanipulation with time delays," *Intern. Journal of Robotics Research*, vol. 23, no. 9, pp. 873–890, 2004.