2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

# Decentralized Nonlinear MPC for Robust Cooperative Manipulation by Heterogeneous Aerial-Ground Robots

Nicola Lissandrini, Christos K. Verginis, Pedro Roque, Angelo Cenedese, and Dimos V. Dimarogonas.

***Abstract*— Cooperative robotics is a trending topic nowadays as it makes possible a number of tasks that cannot be performed by individual robots, such as heavy payload transportation and agile manipulation. In this work, we address the problem of cooperative transportation by heterogeneous, manipulator-endowed robots. Specifically, we consider a generic number of robotic agents simultaneously grasping an object, which is to be transported to a prescribed set point while avoiding obstacles. The procedure is based on a decentralized leader-follower Model Predictive Control scheme, where a designated leader agent is responsible for generating a trajectory compatible with its dynamics, and the followers must compute a trajectory for their own manipulators that aims at minimizing the internal forces and torques that might be applied to the object by the different grippers. The Model Predictive Control approach appears to be well suited to solve such a problem, because it provides both a control law and a technique to generate trajectories, which can be shared among the agents. The proposed algorithm is implemented using a system comprised of a ground and an aerial robot, both in the robotic Gazebo simulator as well as in experiments with real robots, where the methodological approach is assessed and the controller design is shown to be effective for the cooperative transportation task.**

## I. Introduction

Multi-robot systems is a trending and pervasive topic in academic and industrial research, due to the strong potential impact that affects many application fields [1], [2]. For instance, cooperative transportation or manipulation of large or heavy objects [3], [4], inspection and servicing of infrastructures [5], monitoring and mapping of the environment [6], [3], search and rescue operations [7], are just some of the real world applications that can benefit more by these studies and the related technological developments.

Strong results have been demonstrated on the control of single and multiple robotic systems [8], [9] and, more recently, a lot of effort has been made to allow physical interaction among these systems and with the environment [10]. Robustness in the estimation and regulation actions for non-ideal actual scenarios and in presence of environment/agent constraints has been also considered [11], [12].

Heterogeneous robots with different capabilities (e.g., sensing, actuation) are an important aspect of multi-robot systems, since they offer greater flexibility and versatility in complex scenarios [13]. This paper considers the problem of cooperative object transportation via aerial-ground manipulator-endowed robots, which can be beneficial in cases where the different sensing and operating workspaces of the two robotic types might be needed.

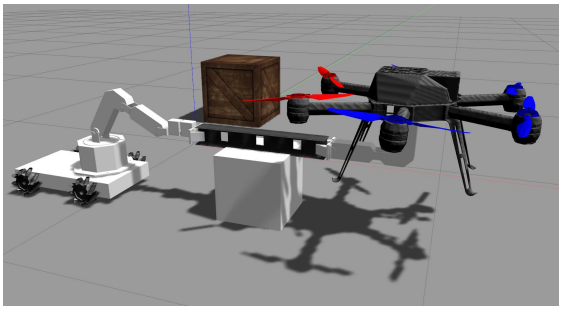


Fig. 1: Two heterogeneous robots transporting an object (in the Gazebo simulation environment).

Regarding the related literature, [14] considers the cooperative object transportation by aerial-ground mobile robots, limited to a coplanar case. Collaborative task control with heterogeneous robots has been also studied in [15], where a team of ground robots is used to stabilize the aerial vehicle, and in [16] where the interaction between a multi-rotor and an industrial manipulator is considered. Further tests for aerial ground manipulation tasks have been made in [17], where the ground vehicle is tasked to deploy the object to a position and the UAV adjusts its attitude to adapt to it.

In the context of this research, the Model Predictive Control approach in its non-linear form (NMPC) appears to be the suitable and effective framework to tackle this study, since it can be formulated as a constrained optimization problem subject to the system dynamics [18] and that accounts also for model uncertainties (e.g. [19], [20]). In [20], in particular, the problem of cooperative manipulation is solved in a non-scalable centralized way by deriving a coupled model of the agents involved in the task, with a decentralized extension being developed in [21]; These solutions, however, do not explicitly consider heterogeneous robots and rely on the strong assumption of rigid grasping robot-object points.

This paper extends the aforementioned works by proposing a multi-robot algorithm for the cooperative object transportation with collision avoidance by heterogeneous robots deriving a novel approach to address the problem, which increases robustness to non-idealities and allows the definition of a more general framework compared to the cited literature, as well as relevant experimental results. The developed scheme is decentralized, since each robot computes its own MPC control signals via a leader-follower coordination, inspired by [21]. Intuitively, a leader robot generates the

N. Lissandrini and A. Cenedese are with the Department of Information Engineering, University of Padova, Padova, Italy. E-mails: nicola.lissandrini@phd.unipd.it, angelo.cenedese@unipd.it

C.K. Verginis, P. Roque and D.V. Dimarogonas are with the Centre for Autonomous Systems at Kungliga Tekniska Hogskolan, Stockholm 10044, Sweden. E-mails: {cverginis, padr ,dimos}@kth.se

desired trajectory of the grasped object, and the rest of the robots comply via an internal force minimization problem, without assuming rigid grasping points. The proposed algorithm is implemented and tested on a system of a ground and aerial robot (see Fig. 1 for an illustrative example), both on the robotic simulator Gazebo [22] and in real laboratory experiments.

## II. Preliminary Notation and Models

We consider a generic setup where $N$ robotic agents are grasping an object. The robots are composed of a moving base and a robotic manipulator, which can have an arbitrary number of degrees of freedom. The base can be either a ground vehicle, e.g. fully actuated with holonomic wheels and 3 d.o.f., or an unmanned aerial vehicle (UAV). The aim of the agents is to transport the object along a collision-free reference trajectory in a decentralized manner avoiding internal forces exerted by the manipulators.

In the remainder of this paper, letters $i$, $j$, $h$, $k$, $\ell$, are used as indexes; scalar parameters and variables are denoted by nonbold lowercase letters, while vector and matrix quantities are denoted, respectively, by bold lowercase and bold uppercase symbols. Rotation matrices are defined in the Special Orthogonal group $\mathbb{SO}(3)$, while $[\,\cdot\,]_\times$ indicates the skew symmetric matrix associated to the argument vector. Given two frames $\{a\}$, $\{b\}$, as well as a world frame $\{W\}$, we denote by $\boldsymbol{T}_b^a$ and $\boldsymbol{T}_b$ the affine transformation from frame $\{a\}$ to frame $\{b\}$ and from $\{W\}$ to $\{b\}$, respectively. Similarly, $\boldsymbol{p}_a \in \mathbb{R}^3$ and $\boldsymbol{R}_a \in \mathbb{SO}(3)$ are the position and rotation matrix, respectively, of frame $\{a\}$ with respect to $\{W\}$. Frames $\{v,i\}$ and $\{e,i\}$ are the vehicle and end-effector frame of agent $i$.

Let $\boldsymbol{q}_i \in \mathbb{R}^{n_i}$ be the vector of joint variables describing the configuration of each manipulator, with $n_i$ being the corresponding number of degrees of freedom. The first-order kinematics of the agents can be written as follows:

$$\text{Agent } i\text{:} \quad \begin{cases} \dot{\boldsymbol{p}}_{e,i} &= \boldsymbol{A}_{p,i}\boldsymbol{u}_{v,i} + \boldsymbol{J}_{P,i}(\boldsymbol{q}_i)\boldsymbol{u}_{q,i} \\ \boldsymbol{\omega}_{e,i} &= \boldsymbol{A}_{\omega,i}\boldsymbol{u}_{\omega,i} + \boldsymbol{J}_{O,i}(\boldsymbol{q}_i)\boldsymbol{u}_{q,i} \\ \dot{\boldsymbol{R}}_{e,i} &= \left[\boldsymbol{\omega}_{e,i}\right]_\times \boldsymbol{R}_{e,i} \\ \dot{\boldsymbol{q}}_i &= \boldsymbol{u}_{q,i} \\ \dot{\boldsymbol{p}}_{v,i} &= \boldsymbol{A}_p\boldsymbol{u}_{v,i} \\ \dot{\boldsymbol{R}}_{v,i} &= \left[\boldsymbol{A}_\omega\boldsymbol{u}_{\omega,i}\right]_\times \boldsymbol{R}_{v,i} \end{cases} \tag{1}$$

where:

- $\boldsymbol{p}_{v,i}$, $\boldsymbol{p}_{e,i}(\boldsymbol{p}_{v,i}, \boldsymbol{q}_i) \in \mathbb{R}^3$ are the position of the base and the end-effector of the $i$-th agent, respectively.
- $\boldsymbol{R}_{v,i}$, $\boldsymbol{R}_{e_i}(\boldsymbol{R}_{v,i}, \boldsymbol{q}_i) \in \mathbb{SO}(3)$ refer to the corresponding rotation matrices relative to a fixed world frame $\{W\}$.
- $\boldsymbol{u}_{v,i}$, $\boldsymbol{u}_{\omega,i} \in \mathbb{R}^3$ are the linear and angular input velocities applied at the base expressed in the local frame.
- $\boldsymbol{u}_{q,i}$ are the joint velocities, which are assumed to be the manipulator's control input.
- $\boldsymbol{A}_{p,i}$, $\boldsymbol{A}_{\omega,i} \in \mathbb{R}^{3\times 3}$ allow to model constraints on the input velocity (e.g. reference frame transform or nonholonomic constraints)

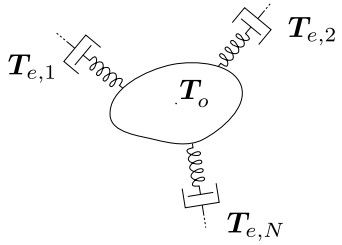



Fig. 2: Elastic joints object to model non-ideal rigidity

- $\boldsymbol{\omega}_{e,i}$ is the angular velocity of the end effector in the world frame.
- $\boldsymbol{J}_{P,i}(\boldsymbol{p}_{v,i}, \boldsymbol{q}_i)$, $\boldsymbol{J}_{O,i}(\boldsymbol{R}_{v,i}, \boldsymbol{q}_i) \in \mathbb{R}^{3\times n_i}$ are, respectively, the position and orientation Jacobian matrices, which depend on the structure of the manipulator.

The full state and input are defined as: $\boldsymbol{x}_i = [\boldsymbol{p}_{e,i}\ \boldsymbol{r}_{e,i}\ \boldsymbol{p}_{v,i}\ \boldsymbol{q}_i]^\top \in \mathbb{R}^{n_i}$, $\boldsymbol{u}_i = [\boldsymbol{u}_{q,i}\ \boldsymbol{u}_{v,i}\ \boldsymbol{u}_{\omega,i}]^\top \in \mathbb{R}^{p_i}$ where the lower case $\boldsymbol{r}_{e,i}$ refers to a vector representation of the rotation matrix $\boldsymbol{R}_{e,i}$, s.t. $[\boldsymbol{R}_{e,i}]_{(h,k)} = [\boldsymbol{r}_{e,i}]_{(3k+h)}$. We can then rewrite (1) with in compact form as $\dot{\boldsymbol{x}}_i = f_{\mathcal{A}_i}(\boldsymbol{x}_i)\boldsymbol{u}_i$, where $f_{\mathcal{A}_i}(\boldsymbol{x}_i)$ collects all the control-affine terms and can be easily inferred by (1).

## III. Cooperative Manipulation with MPC

In this section, we propose a decentralized algorithm for cooperative manipulation with obstacle collision avoidance. The objective is formalized as follows. Let $\{o\}$ be a frame attached to the object's. We consider the reference trajectory that the object is desired to follow, $\boldsymbol{T}_{o,ref}(t) \in \mathbb{SE}(3)$, $t > 0$, where $\mathbb{SE}(3) = \mathbb{R}^3 \times \mathbb{SO}(3)$. The goal is to find a control law for each agent such that the object is transported along the trajectory, while ensuring collision avoidance. To avoid the object being detached from the grasps or breaking, we also aim at minimizing the internal forces and torques applied to the object by the agents.

We assume that, at time $t = 0$, the agents are still and already grasping the object, defining an initial condition for the relative transforms from object frame to each end effector's $\boldsymbol{T}_{e,i}^o(0)$, that, for $t = 0$ only, we assume that they can measure. Also, we assume that they can communicate.

The robotic agents are heterogeneous and, especially in the case of aerial vehicles, they can be characterized by a low number of degrees of freedom. This can lead to situations where a perfect compliance of the grasps is impossible, e.g., an underactuated UAV that needs to roll-pitch to generate horizontal forces and no possibility to compensate.

To allow the algorithm to be robust to such non-idealities, we first need to theoretically allow deviations from rigidity. To this aim, we consider the object gripper joints as elastic, as depicted in Fig. 2, with the rest condition being defined by $\boldsymbol{T}_{e,i}^o(0)$, $i = 1, \ldots, N$. This can either model a case where gripper joints are actually elastic, or a case where the grasps are rigid and the object is elastic itself. We denote by $\boldsymbol{f}_i$, $\boldsymbol{\tau}_i$, the forces and torques applied from agent $i$ to the respective grasping point. In view of the elastic joint model, a part of these forces/torques will result in actual object motion while the rest will be absorbed by the imaginary elastic spring.

We denote the latter by $\boldsymbol{f}_{i,o}$, $\boldsymbol{\tau}_{i,o}$, which, since the initial condition at $t = 0$ is the rest condition, satisfy

$$\left\| \begin{bmatrix} \boldsymbol{f}_{i,o}(t) \\ \boldsymbol{\tau}_{i,o}(t) \end{bmatrix} \right\| = \kappa_i \operatorname{dist}\left( \boldsymbol{T}^o_{e,i}(t), \boldsymbol{T}^o_{e,i}(0) \right) \tag{2}$$

where $\operatorname{dist}(\boldsymbol{T}_a, \boldsymbol{T}_b) := \|\boldsymbol{p}_a - \boldsymbol{p}_b\| + \beta \|\boldsymbol{R}_a^\top \boldsymbol{R}_b - \boldsymbol{I}\|_F$, $\beta \in \mathbb{R}$, $\kappa_i > 0$ is a positive constant depending on the initial condition and physical properties of the object and the grippers, and $\|\cdot\|_F$ is the Frobenius norm. The total forces/torques $\boldsymbol{f}_o$ and $\boldsymbol{\tau}$ applied to the object are the sum of the contributions of the agents. We define as internal forces all the components that cancel out in the sum but create tension/compression stresses on the object. Given all the possible forces/torques applied at the object that produce the same acceleration for the object center of mass, we aim at outputing those with minimal internal forces that also minimize the total norm $\sum_{i=1}^N \|[\boldsymbol{f}_{i,o}(t) \quad \boldsymbol{\tau}_{i,o}(t)]\|^2$, since internal forces do not contribute in the net force but increase the norm sum. Assumption (2) then allows to translate this problem into the minimization of the corresponding displacement $\boldsymbol{T}^o_{e,i}(t)$.

In particular, we address the prescribed objectives by resorting to a nonlinear MPC formulation: at each time $t$, given the state measure $\boldsymbol{x}_i(t)$, solve the following Finite Horizon Optimal Control Problem (FHOCP) [18]:

$$\begin{aligned} \underset{\hat{\boldsymbol{u}}_1(\cdot),\ldots,\hat{\boldsymbol{u}}_N(\cdot)}{\operatorname{argmin}} & \int_t^{t+T} \Big\{ \operatorname{dist}(\hat{\boldsymbol{T}}_o(\tau), \hat{\boldsymbol{T}}_{o,ref}(\tau)) \\ & + \sum_{i=1}^N \operatorname{dist}(\hat{\boldsymbol{T}}^o_{e,i}(\tau), \hat{\boldsymbol{T}}^o_{e,i}(0)) \\ & + \sum_{i=1}^N \hat{\boldsymbol{u}}_i(\tau)^\top \boldsymbol{W}_u \hat{\boldsymbol{u}}_i(\tau) \Big\} \, \mathrm{d}\tau \\ \text{subject to:} \quad & \dot{\hat{\boldsymbol{x}}}_i = f_{\mathcal{A}_i}(\hat{\boldsymbol{x}}_i)\hat{\boldsymbol{u}}_i, \qquad i = 1, \ldots, N \\ & \hat{\boldsymbol{x}}_i(t) = \boldsymbol{x}_i(t) \\ & \hat{\boldsymbol{x}}_i(\tau) \in \mathcal{X}_i, \ \hat{\boldsymbol{u}}_i(\tau) \in \mathcal{U}_i \ \tau \in [t, t+T] \end{aligned} \tag{3}$$

where the hat terms $\hat{\cdot}$ denote the predicted variables, over a horizon $T$. The first integral term is the trajectory error, the second accounts for the minimization of internal forces, $\boldsymbol{W}_u$ is a matrix that weighs a penalty on the control effort, providing stability [18], $T$ is the finite time window (MPC horizon) and $\mathcal{X}_i$, $\mathcal{U}_i$ are the admissible sets for state and input values for each agent, that can account for singularity-avoidance and actuation limits.

The problem can be approached in a decentralized way by resorting to a leader-follower architecture, as follows.

### A. Leader and Follower Coordination

At the design stage, one agent is designed to be the leader. This choice has no theoretical limitations and it is driven by experimental evaluations. The leader computes the trajectory for its end effector such that the object tracks the prescribed trajectory, accounting for the object trajectory error as if the follower agents could not alter its behavior. In other words, the forces applied to the object along the desired trajectory are produced by the leader and the action by the followers is then obtained to minimize the internal forces.

Note that: $\boldsymbol{T}_o(t) = \boldsymbol{T}_{e,\ell}(t)\boldsymbol{T}^{e,\ell}_o(t)$, with $\ell \in \{1, \ldots, N\}$ being the leader index. $\boldsymbol{T}_{e,\ell}(t)$ is a quantity that can be controlled, and it is accounted by (1); $\boldsymbol{T}^{e,\ell}_o(t)$, on the other hand, is not controllable since it is a direct result of the forces applied at the object CoM, due to the elasticity assumption, that are not included in the model. However, if the overall dynamics is sufficiently slow and the elasticity is sufficiently low, it is reasonable to assume that, in absence of other forces and torques, the displacement is bounded over time, i.e.,

$$\operatorname{dist}(\boldsymbol{T}^o_{e,\ell}(t), \boldsymbol{T}^o_{e,\ell}(0)) < \varepsilon_\ell \tag{4}$$

for a positive constant $\varepsilon_\ell$. Without the ability to make predictions on $\boldsymbol{T}^o_{e,\ell}$, which is needed to estimate the object position given the end effector prediction, we will choose as estimate $\hat{\boldsymbol{T}}^o_{e,\ell}(t) = \boldsymbol{T}^o_{e,\ell}(0)$, $\forall t \in [0, T]$, satisfying (4). Due to this assumption, agents are not needed to measure the object frame for $t > 0$. Then, the leader aims at minimizing the cost function:

$$J_\ell(\hat{\boldsymbol{x}}_\ell(\cdot), \boldsymbol{u}_\ell(\cdot)) = \int_t^{t+T} \operatorname{dist}(\hat{\boldsymbol{T}}_{e,\ell}(\tau), \hat{\boldsymbol{T}}_{e,\ell,ref}(t)) \, \mathrm{d}\tau \tag{5}$$

through the following FHOCP:

$$\begin{aligned} \underset{\boldsymbol{u}_\ell(\cdot)}{\operatorname{argmin}} \quad & J_\ell(\boldsymbol{x}_\ell(\cdot), \boldsymbol{u}_\ell(\cdot)) + \int_t^{t+T} \boldsymbol{u}_\ell^\top \boldsymbol{W}_u \boldsymbol{u}_\ell \, \mathrm{d}\tau \\ \text{subject to:} \quad & \dot{\hat{\boldsymbol{x}}}_\ell = f_{\mathcal{A}_\ell}(\hat{\boldsymbol{x}}_\ell)\boldsymbol{u}_\ell \\ & \hat{\boldsymbol{x}}_\ell(t) = \boldsymbol{x}_\ell(t) \\ & \hat{\boldsymbol{x}}_\ell(\tau) \in \mathcal{X}_\ell, \ \boldsymbol{u}_\ell(\tau) \in \mathcal{U}_\ell, \ \tau \in [t, t+T] \end{aligned} \tag{6}$$

The solution to (6) $\boldsymbol{u}^*_\ell(\tau)$, for $\tau \in [t, t+T]$, defines a predicted state trajectory $\boldsymbol{x}^*_\ell(\tau)$ that is optimal with respect to the reference trajectory for the object. In particular, from the first 12 components of the state vector the predicted trajectory for the end effector pose can be extracted, which we will refer to as $\boldsymbol{T}^*_{e,\ell}(\cdot)$.

Conversely to the leader, the followers have to ensure that the trajectory planned by the leader is attained by adapting their system states and output forces/torques. The role of the followers is to minimize the internal forces, which is accomplished, due to (2), by minimizing the second term in (3). For $j \neq \ell$, define $\hat{\boldsymbol{T}}^o_{e,j} = \boldsymbol{T}^o_{e,j}(0)$, and then note that, by left-multiplication with $\boldsymbol{T}_o(t)$ and using (4), we obtain:

$$\begin{aligned} \operatorname{dist}(\boldsymbol{T}^o_{e,j}(t), \boldsymbol{T}^o_{e,j}(0)) &= \operatorname{dist}(\boldsymbol{T}_o(t)\boldsymbol{T}^o_{e,j}(t), \boldsymbol{T}_o(t)\hat{\boldsymbol{T}}^o_{e,j}) \\ &= \operatorname{dist}(\boldsymbol{T}_{e,j}(t), \boldsymbol{T}_{e,\ell}(t)(\hat{\boldsymbol{T}}^o_{e,\ell})^{-1}\hat{\boldsymbol{T}}^o_{e,j}) + \varepsilon_j \end{aligned} \tag{7}$$

where $\varepsilon_j$ is an error due to the approximations of (4). This means that $\boldsymbol{T}^o_{e,j}$, for the followers, is controllable up to $\varepsilon_j$. In this way we explicitly express the displacement of the grasps from the rest condition in terms of controllable quantities. In view of (7) and given $\boldsymbol{T}^*_{e,\ell}(\cdot)$, which is the leader trajectory that minimizes (6), each follower agent $j$ aims at minimizing the following cost function:

$$J_j(\hat{\boldsymbol{x}}_j(\cdot), \boldsymbol{u}_j(\cdot)) = \int_t^{t+T} \operatorname{dist}(\hat{\boldsymbol{T}}_{e,j}(\tau), \hat{\boldsymbol{T}}_{e,j,ref}(\tau)) \, \mathrm{d}\tau \tag{8}$$

where $\boldsymbol{T}_{e,j,ref}(t) = \boldsymbol{T}^{*}_{e,\ell}(t)(\hat{\boldsymbol{T}}^{o}_{e,\ell})^{-1}\hat{\boldsymbol{T}}^{o}_{e,j}$ is a transformation of the trajectory predicted from the leader. This is achieved, by iteratively solving the following FHOCP problem:

$$\begin{aligned} \underset{\boldsymbol{u}_j(\cdot)}{\text{argmin}} \quad & J_j(\hat{\boldsymbol{x}}_j(\cdot), \hat{\boldsymbol{u}}_j(\cdot)) + \int_t^{t+T} \hat{\boldsymbol{u}}_j^{\top} \boldsymbol{W}_u \hat{\boldsymbol{u}}_j \, \mathrm{d}\tau \\ \text{subject to:} \quad & \dot{\hat{\boldsymbol{x}}}_j = f_{\mathcal{A}_j}(\hat{\boldsymbol{x}}_\ell)\hat{\boldsymbol{u}}_j \\ & \hat{\boldsymbol{x}}_j(t) = \boldsymbol{x}_j(t) \\ & \hat{\boldsymbol{x}}_j(\tau) \in \mathcal{X}_j, \ \hat{\boldsymbol{u}}_j(\tau) \in \mathcal{U}_j, \ \tau \in [t, t+T] \end{aligned} \tag{9}$$

### B. Obstacle avoidance

While collision avoidance could be implemented in the MPC constraints, the optimal solution is in most cases on the boundary of the available set and then if for some error the state falls outside this set the problem becomes infeasible. Although this could be solved with slack variables, we propose a different approach that is to be particularly convenient in an experimental environment: this consists in the introduction of an additional term in the cost function, which avoids increasing the number of optimization variables.

Mobile robots are extended objects, so we consider a set of $K$ points defined on the robot as a function of the state $\boldsymbol{p}_{i,k}(\boldsymbol{x}_i) \in \mathbb{R}^3$, e.g. $\boldsymbol{p}_v$, $\boldsymbol{p}_{e,i}$, directly extracted from the state, or any link origin $\boldsymbol{p}_{j,i}(\boldsymbol{x}_i)$, defined according the forward kinematics. A set of $M$ obstacles is defined by their positions $\boldsymbol{o}_m \in \mathbb{R}^3$ and a radius $d_m \in \mathbb{R}$ defining the minimum distance avoiding collision. We then define:

$$J_{i,o}(\boldsymbol{x}_i) = \sum_{k=1}^{K} \sum_{m=1}^{M} C_{i,k,m} e^{-\lambda_{i,k,m}\left(\|\boldsymbol{p}_{i,k}(\boldsymbol{x}_i) - \boldsymbol{o}_m\| + d_m\right)} \tag{10}$$

where $C_{i,k,m}$ is the desired cost value on the boundary of the sphere defined by $(\boldsymbol{o}_m, d_m)$ and $\lambda_{i,k,m}$ determines the decay rate of the cost. Note that due to the exponential, the cost is negligible outside a radius defined by $\lambda_{i,k,m}$. Finally, in (6) and (9) we replace $J_i$ with $J_{full} = J_i + J_{i,o}$, and the same algorithm apply.

## IV. Simulations and Experiments

The proposed framework is validated through a realistic simulation (in Gazebo environment) and experimental results with 2 heterogeneous robots, where the continuous time formulation is discretized with a multiple shooting method. A common implementation for both the simulation and the experiment has been realized via a ROS network, that allows a common input-output interface, so that the same algorithm runs the on the two environment, as illustrated in Fig. 3.

The used heterogeneous robots consist of one ground and one aerial vehicles, as shown in Fig. 5. The ground robot is composed of an omnidirectional base, which is fully actuated on the floor plane, whereas the aerial robot is a planar hexacopter, both equipped with manipulators, of 4 and 2 revolute joints, respectively. The proposed framework is implemented with the ground robot being the leader and the aerial one being the follower.

To evaluate the performance of the algorithm, we consider $\boldsymbol{e}_\ell$ and $\boldsymbol{e}_o$, the position tracking error for the leader and the object, respectively derived from $\boldsymbol{T}_{e,\ell,ref}$ and $\boldsymbol{T}_{e,o,ref}$ Moreover, we consider the respective orientation metrics $\theta_\ell$, $\theta_o$, defined as $\theta_* = \cos^{-1}(2((\tilde{\boldsymbol{q}}_*)^{\top}\tilde{\boldsymbol{q}}_{*,ref})^2 - 1)$, with $\tilde{\boldsymbol{q}}$ being the quaternion errors between the desired and the actual attitude. Since the follower does not follow an explicit trajectory but rather solves an optimization problem, we assess the performance for the aerial vehicle by inspecting the cost value of the MPC problem, which encodes the object displacement from the initial condition and then it is proportional to the internal forces, according to assumption (2). Under ideal conditions, the MPC scheme should always be able to keep it close to zero and, in practice, this should still be bounded. We can assess the validity of the algorithm by verifying that the value does not increase over time.

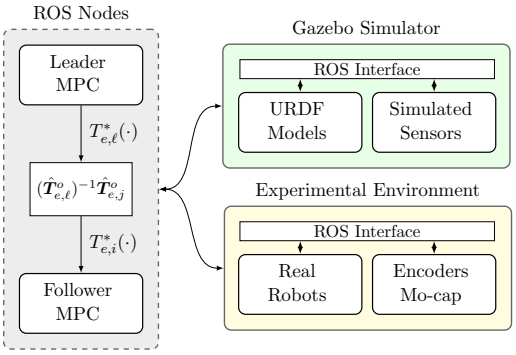



Fig. 3: The MPCs are implemented in ROS, allowing for a common interface to both the simulation and laboratory.

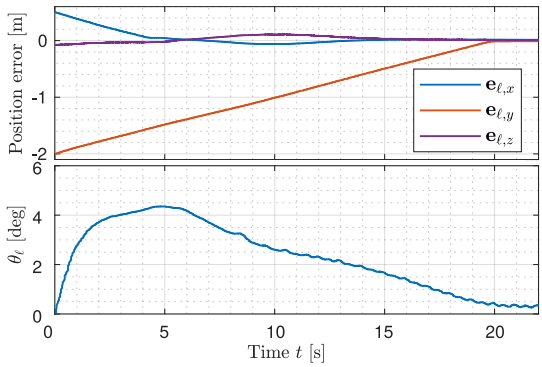
### A. Lower level controllers

In both the simulation and experiment, the MPCs run in different nodes on a off-board computer within the same ROS network as the robots, at 10 Hz and with an horizon length of $T = 1\,\mathrm{s}$, producing a velocity setpoint for the joints and the vehicle. In the case of the ground robot, these are directly supplied to the (real or simulated) motor drivers. The UAV relies on an attitude stabilization and a controller that converts the MPC command to desired roll-pitch-yaw-thrust. To increase robustness, the latter aims at tracking both the computed velocity and the first sample of pose from the trajectory predicted by the MPC.

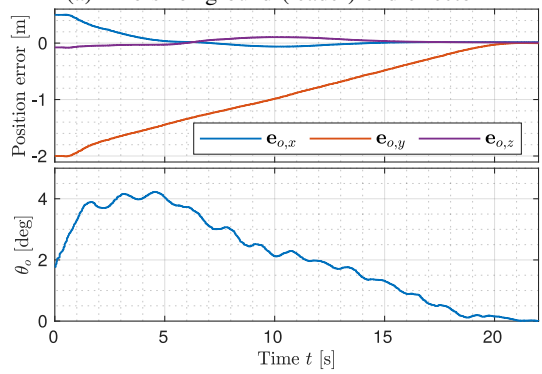
### B. Gazebo simulations

Gazebo is a multi-robot simulator based on *Open Dynamic Engine* physics-engine that allows for realistic robot simulations [22]. The two robots are simulated in Gazebo via custom URDF models, as represented in Fig. 1. The dynamic of all joints and the base of the ground robot are simulated via `ros_control`, while `RotorS` [23] is employed for $n$-rotor flight simulation.

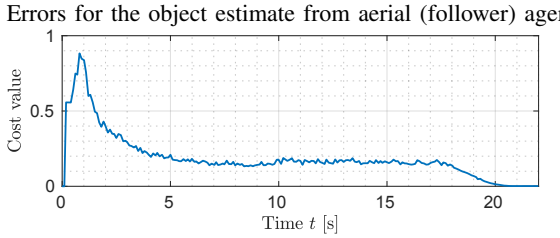
To launch the scenario, the UAV takes off and starts hovering in a predefined position near the object, in the grasp position. After this, the experiment starts ($t = 0$) when the end effectors of ground and aerial robots are at $[0.39,\ 0.02\ 0.34]^{\top}$ [m] and $[0.81,\ 0.00,\ 0.04]^{\top}$ [m]. Fig. 4 shows the results of a simulation where the algorithm is tested with a constant setpoint in $[0, 1.5, 0.3]^{\top}$ [m] and the




(a) Errors for ground (leader) end-effector




(b) Errors for the object estimate from aerial (follower) agent




(c) Aerial MPC objective cost value for predicted window

Fig. 4: Gazebo simulation results

same orientation as the initial state, with an box placed at $[0.0, 0.5, 0.0]^\top$ [m] Fig. 4 shows that both the ground and the aerial robots are able to drive the object error to zero by avoiding the obstacle. In particular, figure 4c shows that the displacement of the follower with respect to the prescribed object trajectory is bounded through the transportation and eventually converges to zero, despite an initial peak due to a delay in the reference tracking. The simulation example is clearly illustrated in the accompanying video.

### C. Experiments

The experiment, whose setup shown in Fig. 5, was conducted at the Smart Mobility Lab[1], at KTH Royal Institute of Technology. A motion capture system was employed to measure the quantities that are part of the state in (1), i.e. the poses of ground base and end effector, and the vehicle of the UAV. The latter's end-effector, instead, was occluded by the vehicle and was estimated via the open-loop forward kinematics. In the initial configuration the robots are assumed

[1]https://www.kth.se/dcs/research/control-of-transport/smart-mobility-lab/smart-mobility-lab-1.441539 - Accessed, March 1st, 2020.

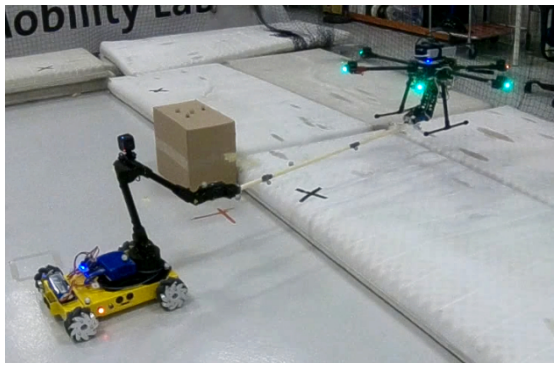


Fig. 5: Experimental setup

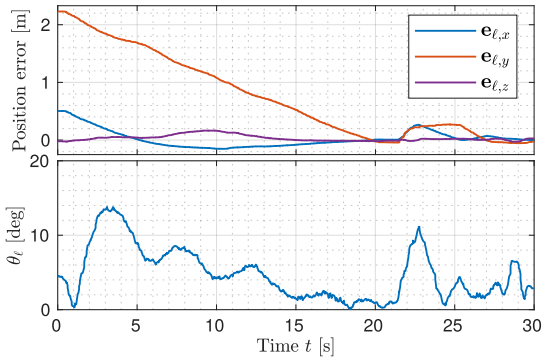
to be already grasping the object, which is is a plastic bar, 0.85 m long, that allows for some elastic deformation.

The experiment starts with the UAV hovering, grasping the object with its end-effector at $[-0.42,\ 0.60,\ 0.15]^\top$ [m], while the ground end-effector is at $[0.42,\ 0.61,\ 0.2687]^\top$ [m] and the bar at $[0.09,\ 0.60,\ 0.2067]$ [m]. Two obstacles, one traffic cone and one box, are placed at $[1.00,\ -0.61,\ 0.00]$ and $[1.00,\ -0.61,\ 0.00]^\top$ [m], forcing the vehicles to perform an avoidance maneuver. The results of a constant-setpoint tracking experiment, where the goal position is set at $[-0.0175,\ -1.5652,\ 0.3000]^\top$ [m] and the goal orientation is the same as the initial one, similar to simulation are reported in Fig. 6a, 6b and 6c.

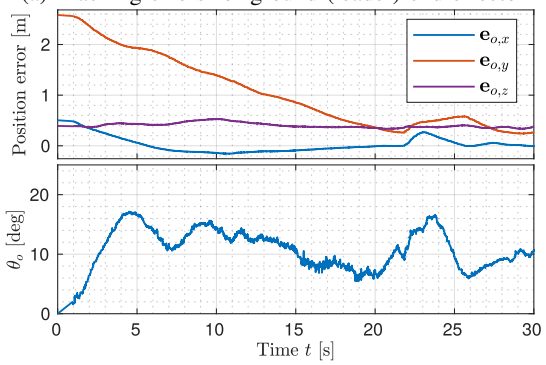
It can be noticed from Fig. 6 that, while the leader robot is able to converge to its setpoint, the object has some error. This means that the relative transforms between the robots and the object, $\boldsymbol{T}^o_{e,\ell}(t)$ and $\boldsymbol{T}^o_{e,j}(t)$, are not exactly equal to the initial condition $\boldsymbol{T}^o_{e,\ell}(0)$ and $\boldsymbol{T}^o_{e,j}(0)$. Fig. 6c shows that while the cost is bounded, and then the displacements are within the physical limits of object detachment, it is not driven to zero, as in the gazebo simulations. This cannot be attributed to the effects of the internal forces because, intuitively, they would tend to push the follower to lower the error. Instead, the degradation of performance can be caused by saturation in the low level controller of the UAV, and ground effects that arise since the latter is flying close to the ground and the obstacles. Nevertheless, both the error and the cost in Fig. 6c are bounded and the MPC scheme is able to keep the system stable, even when at time $t = 22\,\mathrm{s}$, when a fictitious external disturbance is simulated by applying a short impulse to the ground vehicle arm joints commands. In that case, the plots show that the algorithm is able to handle the disturbance and keep the error bounded. The experiment is clearly illustrated in the accompanying video, which shows that the algorithm is able to complete the transportation task without having the object detached or damaged.

## V. Conclusion

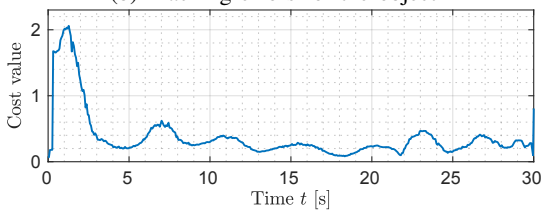
In this work we proposed a decentralized algorithm to coordinate a team of heterogeneous robotic agents that are designed to transport an object to a prescribed target pose. The procedure is designed to be robust to uncertainties and unmodelled dynamics such as underactuation and non-ideal tracking of the computed control inputs. The technique was




(a) Tracking errors for ground (leader) end-effector




(b) Tracking errors for the object




(c) Aerial MPC objective cost value for predicted window

Fig. 6: Experimental results

tested both in a realistic simulation framework and with a laboratory experiment. In the former, the task was completed with converging errors, whereas in the latter an error is present at steady-state, mostly due to imperfect low-level control tracking. Nevertheless, the system is still able to keep the error bounded and react to unexpected external disturbances. The main limitations of the algorithm are that the inertia of the object and the ability of the followers to actually cope with the dynamics of the computed trajectory cannot be accounted. These aspects will be part of future development of this work.

## REFERENCES


[1] S. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar, "A survey on aerial swarm robotics," *IEEE Trans. on Robotics*, vol. 34, no. 4, pp. 837–855, 2018.

[2] S. Wilson, P. Glotfelter, L. Wang, S. Mayya, G. Notomista, M. Mote, and M. Egerstedt, "The robotarium: Globally impactful opportunities, challenges, and lessons learned in remote-access, distributed control of multirobot systems," *IEEE Control Systems Magazine*, vol. 40, no. 1, pp. 26–44, 2020.

[3] J. Alonso-Mora, S. Baker, and D. Rus, "Multi-robot formation control and object transport in dynamic environments via constrained optimization," *The International Journal of Robotics Research*, vol. 36, no. 9, pp. 1000–1021, 2017.

[4] H. Lee, H. Kim, and H. J. Kim, "Planning and control for collision-free cooperative aerial transportation," *IEEE Trans. on Automation Science and Engineering*, vol. 15, no. 1, pp. 189–201, 2018.

[5] A. Suarez, G. Heredia, and A. Ollero, "Lightweight compliant arm with compliant finger for aerial manipulation and inspection," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2016, pp. 4449–4454.

[6] J. Cortés, "Coverage optimization and spatial load balancing by robotic sensor networks," *IEEE Trans. on Automatic Control*, vol. 55, no. 3, pp. 749–754, 2010.

[7] J. León, G. A. Cardona, A. Botello, and J. M. Calderón, "Robot swarms theory applicable to seek and rescue operation," in *Int. Conf. on Intelligent Systems Design and Applications*. Springer, 2016, pp. 1061–1070.

[8] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[9] A. Franchi, C. Secchi, M. Ryll, H. H. Bulthoff, and P. R. Giordano, "Shared control : Balancing autonomy and human assistance with a group of quadrotor uavs," *IEEE Robotics Automation Magazine*, vol. 19, no. 3, pp. 57–68, 2012.

[10] A. Gawel, M. Kamel, T. Novkovic, J. Widauer, D. Schindler, B. P. von Altishofen, R. Siegwart, and J. Nieto, "Aerial picking and delivery of magnetic objects with mavs," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2017, pp. 5746–5752.

[11] S. Zhao, D. V. Dimarogonas, Z. Sun, and D. Bauso, "A general approach to coordination control of mobile agents with motion constraints," *IEEE Trans. on Automatic Control*, vol. 63, no. 5, pp. 1509–1516, 2017.

[12] M. Corah and N. Michael, "Active estimation of mass properties for safe cooperative lifting," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2017, pp. 4582–4587.

[13] K. M. Wurm, C. Dornhege, B. Nebel, W. Burgard, and C. Stachniss, "Coordinating heterogeneous teams of robots using temporal symbolic planning," *Autonomous Robots*, vol. 34, no. 4, pp. 277–294, 2013.

[14] T. Nguyen and E. Garone, "Control of a uav and a ugv cooperating to manipulate an object," in *American Control Conference (ACC)*, 2016, pp. 1347–1352.

[15] R. Naldi, A. Gasparri, and E. Garone, "Cooperative pose stabilization of an aerial vehicle through physical interaction with a team of ground robots," in *IEEE Int. Conf. on Control Applications*, 2012, pp. 415–420.

[16] K. Kondak, F. Huber, M. Schwarzbach, M. Laiacker, D. Sommer, M. Bejar, and A. Ollero, "Aerial manipulation robot composed of an autonomous helicopter and a 7 degrees of freedom industrial manipulator," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2014, pp. 2107–2112.

[17] N. Staub, M. Mohammadi, D. Bicego, D. Prattichizzo, and A. Franchi, "Towards robotic magmas: Multiple aerial-ground manipulator systems," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2017, pp. 1307–1312.

[18] R. Findeisen, L. Imsland, F. Allgower, and B. A. Foss, "State and output feedback nonlinear model predictive control: An overview," *European Journal of Control*, vol. 9, no. 2-3, pp. 190–206, 2003.

[19] T. P. Nascimento, A. P. Moreira, and A. G. S. Conceição, "Multi-robot nonlinear model predictive formation control: Moving target and target absence," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1502–1515, 2013.

[20] A. Nikou, C. Verginis, S. Heshmati-Alamdari, and D. V. Dimarogonas, "A nonlinear model predictive control scheme for cooperative manipulation with singularity and collision avoidance," in *25th Mediterranean Conf. on Control and Automation (MED)*, 2017, pp. 707–712.

[21] C. K. Verginis, A. Nikou, and D. V. Dimarogonas, "Communication-based decentralized cooperative object transportation using nonlinear model predictive control," *European Control Conf. (ECC)*, pp. 733–738, 2018.

[22] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, 2004, pp. 2149–2154.

[23] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, *Robot Operating System (ROS): The Complete Reference (Volume 1)*. Cham: Springer International Publishing, 2016, ch. RotorS—A Modular Gazebo MAV Simulator Framework, pp. 595–625.