



HAL
open science

Cooperative multi-robot object transportation system based on hierarchical quadratic programming

Daravuth Koung, Olivier Kermorgant, Isabelle Fantoni, Lamia Belouaer

► To cite this version:

Daravuth Koung, Olivier Kermorgant, Isabelle Fantoni, Lamia Belouaer. Cooperative multi-robot object transportation system based on hierarchical quadratic programming. *IEEE Robotics and Automation Letters*, 2021, 6 (4), pp.6466-6472. 10.1109/LRA.2021.3092305 . hal-03289793

HAL Id: hal-03289793

<https://hal.science/hal-03289793v1>

Submitted on 18 Jul 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Cooperative multi-robot object transportation system based on hierarchical quadratic programming

Daravuth Koung¹, Olivier Kermorgant², Isabelle Fantoni², and Lamia Belouaer³

Abstract—Formation control gains significant attention in the multi-robot system field as it contributes to a vast range of applications, such as transportation. This paper presents a constrained optimization-based control law for cooperative logistics mission, which consists of rigid shape formation control, group navigation, individual and team obstacle avoidance tasks. These tasks are defined as equality and inequality constraints with different levels of priority. Hierarchical quadratic programming (HQP) approach is used to solve for the optimal solution with an inclusion of velocity limits as inequality constraints to ensure implementation feasibility. Experiment using actual industrial robots is demonstrated in order to validate the theory.

Index Terms—Cooperating Robots, Multi-Robot Systems, Optimization and Optimal Control.

I. INTRODUCTION

FOR decades, researchers have been pushing the boundary of multi-robot system (MRS) applications, taking advantage of its improved robustness and efficiency compared to single-robot systems [1]. Logistics is one of the emerging applications of MRS. In fact, logistics sectors can benefit from MRS's flexibility and reliability to extend the load-carrying capacity. For transportation tasks, various load handling and control strategies have been proposed. Some load handling methods include [2]: pushing-only, caging, and grasping. For the pushing-only and caging strategy, in which there is no physical joint between robots and the object, the robots need to either push [3] or entrap [4] the object to move it. In [5], an occlusion-based approach was proposed, in which robots push the object along their line-of-sight to the goal that is occluded by the object. On the other hand, the authors of [6] used the grasping strategy in the form of a gripper to hold the transporting object with leader/follower control. Similarly, a leader/helper approach is presented in [7]. The load is placed on top of two mobile robots, and the system's dynamic is analyzed based on a theoretical framework called attractor dynamics approach to behavior generation.

Manuscript received: February, 24, 2021; Revised May, 12, 2021; Accepted June, 11, 2021.

This paper was recommended for publication by Editor A. Hsieh upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the French Industrial Agreement of Training through Research between E-COBOT and CNRS (CIFRE 2018/1206).

¹Daravuth Koung is with Laboratoire des Sciences du Numérique de Nantes (LS2N), Ecole Centrale de Nantes (ECN) and E-COBOT, France. daravuth.koung@ls2n.fr

²Olivier Kermorgant is with LS2N and ECN. Isabelle Fantoni is with LS2N and CNRS. olivier.kermorgant@ec-nantes.fr, isabelle.fantoni@ls2n.fr

³Lamia Belouaer is with E-COBOT. l.belouaer@e-cobot.com
Digital Object Identifier (DOI): see top of this page.

We aim to implement a MRS in real-world environments for the cooperative transportation system, where an object is placed on top of a group of nonholonomic mobile robots. This paper proposed an optimization-based controller that solves object transportation problem by decomposing it into multiple task objectives. These tasks are represented by equality and inequality constraints with different levels of priority. HQP is used in order to solve for an optimal solution taking into account the strict hierarchy of tasks.

A. Problem statement

Fig. 1 shows a team of wheeled mobile robots, whose mission is to move an object from one point to another while avoiding collisions with the obstacle(s). The team must first form a formation before the load can be placed on top. The problem is composed of formation control, cooperative navigation, and obstacle avoidance task.

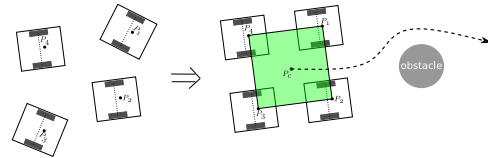


Fig. 1: Four robots scenario. HUSKY [8] are used in the experiments. They are industrial mobile robots designed for indoor applications. The green square represents a load on top of the group.

During this first phase of formation, we assume that the formation is already at least topologically correct even though the overall shape and inter-distances are not. If the robots are far from the starting position, a multi-robot trajectory generation may be used. In addition, if there are obstacles in the environment, the free space must be large enough for the team to pass through during navigation as the formation is rigid, which can not shrink or expand.

Input-output state feedback, in which the control input is applied to an offset point instead of the robot's center (more detail in section II-A), is used in order to deal with nonholonomic constraints of the robots. The payload is connected to each robot at this offset point via a pin-hole system to form a passive revolute joint on the vertical axis. By mounting as such, the load's position and heading can be directly controlled since its center is coincident with the formation frame.

Once the payload is mounted, the whole system can be seen as a single, highly overactuated mobile robot. Thus, it is of

prime importance to generate consistent control inputs for all robots such that the formation is kept. Ignoring these would lead to either set high forces on the pin-hole mechanisms or induce slipping or skidding of some of the robots. In both cases, the general motion of the formation would not be controlled correctly. The controller is thus composed of several tasks with different levels of importance. For instance, formation is always more important than navigation, and the HQP approach is a good candidate to solve such problem.

B. Related work

To carry out the logistics mission, each robot in the team generally has to navigate collectively as a team with obstacle avoidance capability. The challenge for such mission is to handle tasks priority avoiding conflicts among them. Numerous formation strategies have been introduced. Some control approaches such as consensus-based [9] and leader-follower [10] rely on control gains tuning in order to obtain a good compromise between tasks, thus priorities are not robustly enforced. In [11], the payload was connected to each robot via an arm that consisted of a revolute joint on the vertical axis and a prismatic joint on the horizontal axis. Instead of maintaining fixed inter-distances between followers and a leader, the leader-follower controller tried to keep a fixed length of the passive prismatic joint.

On the other hand, the optimization approach can be used to better solve such problem in the form of constraints. In [12], the authors proposed a convex optimization with shape analysis for formation. The goal was to minimize the displacement of each robot to the optimal pose in order to form a specified shape. The controller effectiveness was only shown in simulations, however, and nonholonomic kinematic constraints were not considered. In addition, while the shape was preserved, the scale of the formation was not. The author of [13] proposed a constrained optimization scheme consists of global path planning for collision-free navigation and local formation control. Experimental results were shown in the form of cooperative object transportation. However, the load was carried by manipulators that were on top of the mobile platforms. Thus the rigidity of the formation's shape was not maintained. In contrast, having fixed inter-distances between robots is desired in our case because the payload is connected directly to the robots.

In [14], a priority-based control was proposed using the null-space control approach. The authors demonstrated a combination of two tasks, where individual obstacle avoidance had higher priority than formation. However, the proposed control was validated on holonomic robots in simulation, and group navigation was not included in the study. Unlike the null-space approach [14][15], HQP allows inequality constraints, such as velocity limit, to be imposed at any level.

HQP is generally used to ensure the strict priority order of tasks. It is widely used in the field of humanoid robots [16], which naturally has strict constraints to be respected, such as joint limits. Applying to the MRS, authors of [17] have recently designed a method for cooperative transportation mission based on the HQP approach. Different tasks were in-

troduced, including formation, navigation, and obstacle avoidances, in different levels of priority with a finite-state machine to switch between missions. For the formation task, the authors maintained each robot to be on the circumference of a circle with uniform spatial distribution in order to cage the target object in the center to move it. Thus the inter-distances were not explicitly controlled. In addition, it can be inefficient to transport load this way due to frictional forces as the object's mass increases.

Our work leverages the HQP in order to compute online the optimal control input of the system. Different from [17], we define the formation task by using inter-distances between robots to ensure a rigid shape formation. Moreover, the formation's shape can be easily adapted to that of the object by changing the definition of the inter-distances. A novel team's orientation constraint is introduced, in addition, to the navigation task. An inequality constraint of the velocity limit is also added at every level of the optimization. The object to be transported is placed on top of each robot without needing to use grasping or caging strategy.

The rest of the paper is organized as follows. Section II describes the basic notions of the controller and methodology to define each task. HQP and overall control laws are shown in section III. Section IV demonstrates simulation and experimental results to validate the approach. Finally, the conclusion and future work are given in section V.

II. TASK DEFINITIONS

Exploiting the concept of sensor-based control, we can derive a relation between the features' rate of changes and the robots' velocities, which can be expressed as follows [18]:

$$\dot{s} = \mathbf{J}\mathbf{v} \quad (1)$$

where \dot{s} is the rate of change of the features we choose to control. \mathbf{J} and \mathbf{v} are the Jacobian matrix and robots' velocities, respectively.

Let s^* is setpoint and $e = s - s^*$ is error, a proportional controller, $\dot{e} = -\lambda e$, can be used to ensure an exponential decrease of e with $\lambda > 0$. Thus (1) can be written as follows:

$$\mathbf{J}\mathbf{v} = -\lambda e \quad (2)$$

The tasks will be generally defined for n robots, and we chose specifically $n = 4$ for implementation examples.

A. Unicycle model and input-output state feedback

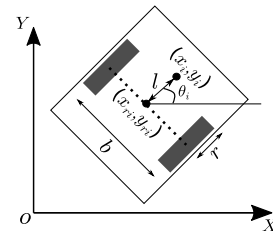


Fig. 2: A nonholonomic wheeled mobile robot.

Consider a multi-robot system with n unicycle robots, $(x_{ri}, y_{ri})^T \in \mathbb{R}^2$ is the center of the i -th robot and θ_i, v_i and

ω_i are its heading angle, linear velocity and angular velocity, respectively. The nonholonomic constraints of the wheeled mobile robot can be expressed as following:

$$\begin{pmatrix} \dot{x}_{ri} \\ \dot{y}_{ri} \\ \dot{\theta}_i \end{pmatrix} = \begin{pmatrix} \cos \theta_i & 0 \\ \sin \theta_i & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v_i \\ \omega_i \end{pmatrix} \quad (3)$$

In order to be able to apply the defined tasks on nonholonomic robots, input-output state feedback method is used by defining an offset position from the robot center, as shown in Fig. 2, and their relation is written as follows [19]:

$$\mathbf{p}_i = \begin{pmatrix} x_i \\ y_i \end{pmatrix} = \begin{pmatrix} x_{ri} + l \cos \theta_i \\ y_{ri} + l \sin \theta_i \end{pmatrix} \quad (4)$$

Differentiating (4) with respect to time:

$$\dot{\mathbf{p}}_i = \begin{pmatrix} \cos \theta_i & -l \sin \theta_i \\ \sin \theta_i & l \cos \theta_i \end{pmatrix} \begin{pmatrix} v_i \\ \omega_i \end{pmatrix} = \mathbf{B}(\theta_i) \mathbf{u}_i \quad (5)$$

where $l > 0$ is the offset length, and $\mathbf{u}_i = (v_i, \omega_i)^T$ is the control input of i^{th} robot. From (5), it is clearly shown that having length l different from zero is necessary in order to inverse matrix $\mathbf{B}(\theta_i)$.

All the defined tasks are applied on the offsets' positions, and the relation between the velocity of the offset point and the control input of each robot in the case of four robots is expressed as follows:

$$\mathbf{v} = \mathbf{J}_x \mathbf{u} \quad (6)$$

where:

$$\mathbf{v} = \begin{pmatrix} \dot{\mathbf{p}}_1 \\ \vdots \\ \dot{\mathbf{p}}_4 \end{pmatrix}, \mathbf{J}_x = \begin{pmatrix} \mathbf{B}(\theta_1) & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{B}(\theta_4) \end{pmatrix}, \mathbf{u} = \begin{pmatrix} v_1 \\ \omega_1 \\ \vdots \\ v_4 \\ \omega_4 \end{pmatrix}$$

B. Geometric formation

The task of forming into a particular geometric shape is one of the most important aspects of MRS. The inter-distances between each robot in the group are used as the control features. With user-defined values of the inter-distances, specific geometric shape formation can be done.

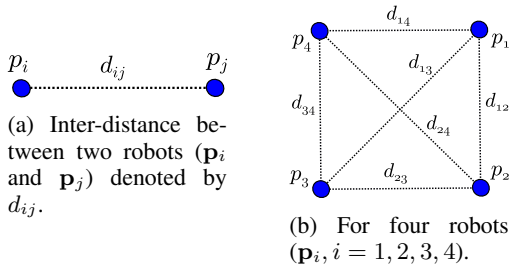


Fig. 3: Inter-distances diagram of the MRS, each blue node represents a mobile robot in 2D plane.

If we consider two points as shown in Fig. 3a with $\mathbf{p}_i, \mathbf{p}_j \in \mathbb{R}^2, i, j = 1, 2, \dots$, the distance d_{ij} is as follows:

$$d_{ij} = \|\mathbf{p}_j - \mathbf{p}_i\| \quad (7)$$

By differentiating (7) with respect to time, we can get the relation between the change of inter-distance and the change of robots' positions as follows:

$$\dot{d}_{ij} = \left(-\frac{(\mathbf{p}_j - \mathbf{p}_i)^T}{d_{ij}} \quad \frac{(\mathbf{p}_j - \mathbf{p}_i)^T}{d_{ij}} \right) \begin{pmatrix} \dot{\mathbf{p}}_i \\ \dot{\mathbf{p}}_j \end{pmatrix} \quad (8)$$

In case of four robots as shown in Fig. 3b, the formation Jacobian \mathbf{J}_f can be derived by stacking (8):

$$\dot{\mathbf{d}}_f = \mathbf{J}_f \mathbf{v} \quad (9)$$

where:

$$\dot{\mathbf{d}}_f = \begin{pmatrix} \dot{d}_{12} \\ \dot{d}_{13} \\ \dot{d}_{14} \\ \dot{d}_{23} \\ \dot{d}_{24} \\ \dot{d}_{34} \end{pmatrix}, \mathbf{J}_f = \begin{pmatrix} -\mathbf{n}_{12} & \mathbf{n}_{12} & \mathbf{0}_{1 \times 2} & \mathbf{0}_{1 \times 2} \\ -\mathbf{n}_{13} & \mathbf{0}_{1 \times 2} & \mathbf{n}_{13} & \mathbf{0}_{1 \times 2} \\ -\mathbf{n}_{14} & \mathbf{0}_{1 \times 2} & \mathbf{0}_{1 \times 2} & \mathbf{n}_{14} \\ \mathbf{0}_{1 \times 2} & -\mathbf{n}_{23} & \mathbf{n}_{23} & \mathbf{0}_{1 \times 2} \\ \mathbf{0}_{1 \times 2} & -\mathbf{n}_{24} & \mathbf{0}_{1 \times 2} & \mathbf{n}_{24} \\ \mathbf{0}_{1 \times 2} & \mathbf{0}_{1 \times 2} & -\mathbf{n}_{34} & \mathbf{n}_{34} \end{pmatrix}_{6 \times 8}$$

$$\mathbf{n}_{ij} = \frac{(\mathbf{p}_j - \mathbf{p}_i)^T}{d_{ij}}$$

From (2), the formation task is, thus, defined as follows:

$$\mathbf{J}_f \mathbf{v} = -\lambda_f (\mathbf{d}_f - \mathbf{d}_f^*) \quad (10)$$

where \mathbf{d} and \mathbf{d}_f^* are vectors of current and desired inter-distance, respectively. λ_f is the formation control gain.

C. Cooperative navigation

This task focuses on driving the formation's centroid toward a desired point $\mathbf{p}_d = (x_d, y_d)^T$ and orientation θ_d . Here the pose (position and orientation) of the centroid is used as the control features. Fig. 4 shows the task of team navigation, which the centroid $\mathbf{p}_c = (x_c, y_c)^T$ has to move toward a target point \mathbf{p}_d as well as to orient its heading to any desired angle, in that case, θ_d .

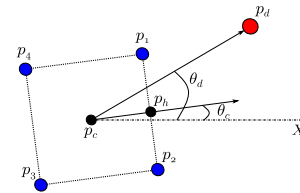


Fig. 4: Navigation scheme, the blue nodes represent robots whereas the red node indicates a target point.

The position of formation's centroid is defined as following, with n is the number of robots in the group:

$$x_c = \frac{1}{n} \sum_{i=1}^n x_i \quad \text{and} \quad y_c = \frac{1}{n} \sum_{i=1}^n y_i \quad (11)$$

The point $\mathbf{p}_h = (x_h, y_h)^T$ can be defined anywhere, relatively to each robot in the group. This point can be used to determine the heading of formation, and it can be expressed in relation to robots' positions as follows:

$$x_h = \sum_{i=1}^n a_i x_i \quad \text{and} \quad y_h = \sum_{i=1}^n a_i y_i \quad (12)$$

where a_i is the relation coefficient to i^{th} robot, and $\sum_{i=1}^n a_i = 1$. For instance, $a_1 = a_2 = 0.5$ and $a_3 = a_4 = 0$ in the configuration shown in Fig. 4.

The group's heading can be found using \mathbf{p}_c and \mathbf{p}_h :

$$\theta_c = \arctan\left(\frac{y_h - y_c}{x_h - x_c}\right) = \arctan\frac{\sum_{i=1}^n y_i (a_i - \frac{1}{n})}{\sum_{i=1}^n x_i (a_i - \frac{1}{n})} \quad (13)$$

By differentiating (11) and (13) with respect to time, the Jacobian of navigation task \mathbf{J}_n can be determined as follows:

$$\begin{pmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\theta}_c \end{pmatrix} = \begin{pmatrix} 1/n & 0 & \cdots & 1/n & 0 \\ 0 & 1/n & \cdots & 0 & 1/n \\ -m_{y_1} & m_{x_1} & \cdots & -m_{y_n} & m_{x_n} \end{pmatrix} \begin{pmatrix} \dot{\mathbf{p}}_1 \\ \vdots \\ \dot{\mathbf{p}}_n \end{pmatrix} \quad (14)$$

where:

$$m_{x_j} = \frac{(a_j - 1/n) \sum_{i=1}^n x_i (a_i - \frac{1}{n})}{[\sum_{i=1}^n x_i (a_i - \frac{1}{n})]^2 + [\sum_{i=1}^n y_i (a_i - \frac{1}{n})]^2}$$

$$m_{y_j} = \frac{(a_j - 1/n) \sum_{i=1}^n y_i (a_i - \frac{1}{n})}{[\sum_{i=1}^n x_i (a_i - \frac{1}{n})]^2 + [\sum_{i=1}^n y_i (a_i - \frac{1}{n})]^2}$$

$$j = 1, 2, \dots, n$$

Applying to four robots configuration, we get:

$$\begin{pmatrix} \dot{x}_c & \dot{y}_c & \dot{\theta}_c \end{pmatrix}^T = \mathbf{J}_n \mathbf{v} \quad (15)$$

where $\mathbf{J}_n = \begin{pmatrix} 1/4 & 0 & \cdots & 1/4 & 0 \\ 0 & 1/4 & \cdots & 0 & 1/4 \\ -m_{y_1} & m_{x_1} & \cdots & -m_{y_4} & m_{x_4} \end{pmatrix}$

From (2), the navigation task can be derived as follows:

$$\mathbf{J}_n \mathbf{v} = - \begin{pmatrix} \lambda_p (\mathbf{p}_c - \mathbf{p}_d) \\ \lambda_\theta (\theta_c - \theta_d) \end{pmatrix} \quad (16)$$

where λ_p and λ_θ are the control gains for position and orientation, respectively.

D. Individual and team obstacle avoidance

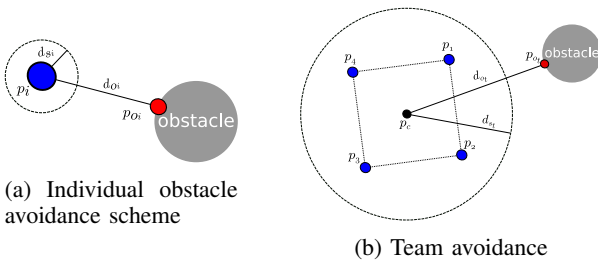


Fig. 5: The red node represents a closest detected point of obstacle (a) to robot, \mathbf{p}_{o_i} and (b) to group's centroid, \mathbf{p}_{o_t} . d_{o_i} and d_{o_t} are the distances from i^{th} robot and team's centroid to the obstacle, while d_{s_i} and d_{s_t} are the safety distances for individual and team obstacle avoidance, respectively.

Two obstacle avoidance schemes are introduced in the overall control law depending on the situation (Fig. 5). For instance, individual obstacle avoidance is useful when each robot tries to converge to a defined formation, while the team approach is more practical during navigation. They can be seen

as sets of inequality constraints whose purpose is to make sure that the distances to obstacle(s), which are the control features, can never be smaller than the safety distances [17]. They can be found as follows:

$$d_{o_i} = \|\mathbf{p}_{o_i} - \mathbf{p}_i\| \quad \text{and} \quad d_{o_t} = \|\mathbf{p}_{o_t} - \mathbf{p}_c\| \quad (17)$$

Similar to the formation task, we differentiate (17) with respect to time. For each obstacle, we get the following:

$$\dot{d}_{o_i} = \frac{(\mathbf{p}_i - \mathbf{p}_{o_i})^T}{d_{o_i}} \dot{\mathbf{p}}_i$$

$$\dot{d}_{o_t} = \frac{(\mathbf{p}_c - \mathbf{p}_{o_t})^T}{d_{o_t}} \begin{pmatrix} 1/n & 0 & \cdots & 1/n & 0 \\ 0 & 1/n & \cdots & 0 & 1/n \end{pmatrix} \begin{pmatrix} \dot{\mathbf{p}}_1 \\ \vdots \\ \dot{\mathbf{p}}_n \end{pmatrix} \quad (18)$$

Consider for our four robots, the individual and team obstacle avoidance Jacobian, \mathbf{J}_{o_i} and \mathbf{J}_{o_t} , respectively, are written as follows:

$$\mathbf{J}_{o_i} = \frac{(\mathbf{p}_i - \mathbf{p}_{o_i})^T}{d_{o_i}} \quad (19)$$

$$\mathbf{J}_{o_t} = \frac{(\mathbf{p}_c - \mathbf{p}_{o_t})^T}{4d_{o_t}} (\mathbf{I}_2 \quad \mathbf{I}_2 \quad \mathbf{I}_2 \quad \mathbf{I}_2)_{2 \times 8}$$

The inequality constraints of the obstacle avoidance tasks can be, therefore, expressed as follows:

$$\mathbf{J}_{o_i} \mathbf{v}_i \geq -\lambda_{o_i} (d_{o_i} - d_{s_i}) \quad (20)$$

$$\mathbf{J}_{o_t} \mathbf{v} \geq -\lambda_{o_t} (d_{o_t} - d_{s_t})$$

where λ_{o_i} and λ_{o_t} are the control gains for individual and team obstacle avoidance, respectively.

The distances can be in the form of vectors or scalars depending on the number of obstacles detected in the workspace, and they can also be null (thus no inequality constraint) if there is no presence of the obstacle.

E. Velocity limits

An upper and lower velocity limit are imposed on the robot's left and right wheel as inequality constraints. Consider b and r as, respectively, the distance between the two wheels and wheel's radius (as shown in Fig. 2), the relation of linear and angular velocity (v_i, ω_i) to wheel's velocity ($\omega_{l,i}, \omega_{r,i}$) of i^{th} robot can be expressed as follows:

$$\omega_{l,i} = \frac{1}{r} (v_i - 0.5b\omega_i) \quad (21)$$

$$\omega_{r,i} = \frac{1}{r} (v_i + 0.5b\omega_i)$$

Thus, the inequality constraint of the velocity limit for four robots can be written as following:

$$\mathbf{J}_{lim} \mathbf{u} \leq \boldsymbol{\omega}_{lim} \quad (22)$$

where:

$$\mathbf{J}_{lim} \text{ is a } 16 \times 8 \text{ block diagonal matrix of } \mathbf{G},$$

$$\mathbf{G} = \frac{1}{r} \begin{pmatrix} 1 & -0.5b \\ 1 & 0.5b \\ -1 & 0.5b \\ -1 & -0.5b \end{pmatrix}, \boldsymbol{\omega}_{lim} = \begin{pmatrix} \mathbf{H} \\ \mathbf{H} \\ \mathbf{H} \\ \mathbf{H} \end{pmatrix}, \mathbf{H} = \begin{pmatrix} \omega_{l,max} \\ \omega_{r,max} \\ -\omega_{l,min} \\ -\omega_{r,min} \end{pmatrix}$$

III. OPTIMIZATION-BASED CONTROLLER

The hierarchical quadratic programming (HQP) framework [20] is used to solve for control input ensuring strict priority order. HQP is a sequence of QP that try to minimize a cost function subjected to equality and inequality constraints as well as solutions of higher priority tasks. At k -level of priority, HQP can be formulated as follows:

$$\begin{aligned} \mathbf{u}_k = & \arg \min_{\mathbf{x}} \|\mathbf{A}_k \mathbf{x} - \mathbf{b}_k\|^2 \\ \text{s.t.} & \mathbf{A}_i \mathbf{x} = \mathbf{A}_i \mathbf{u}_i, \forall i < k \\ & \mathbf{C}_k \mathbf{x} \leq \mathbf{d}_k \\ & \mathbf{C}_i \mathbf{x} \leq \mathbf{d}_i, \forall i < k \end{aligned} \quad (23)$$

where \mathbf{u}_k is the optimal solution that minimizes the cost function of equality task k , \mathbf{x} is the optimization variable, Matrix \mathbf{A}_k and vector \mathbf{b}_k describe the cost function, whereas \mathbf{A}_i and \mathbf{b}_i represent the cost function of previous hierarchies. \mathbf{C} and \mathbf{d} are inequality constraints of the current and previous levels. The equality constraint ensures that the objective task is minimized at best without disturbing previous higher priority tasks.

Note that the equality constraints inherited from previous hierarchy levels are actually $\|\mathbf{A}_i \mathbf{x} - \mathbf{b}_i\|^2 = \|\mathbf{A}_i \mathbf{u}_i - \mathbf{b}_i\|^2$. Writing them as in (23) makes the constraints stronger but keeps them linear. The final solution can be chosen as the minimal-norm one by solving the last hierarchy level with the pseudo-inverse.

The control scheme is composed of two behavior states:

- Stage 1 (S1): Initial formation ensures that the desired shape is reached before mounting the payload on top. At this stage, only one level of optimization is needed. Hence a formation task that is subjected to individual obstacle avoidance and velocity limits constraints, which leads to the initial formation.
- Stage 2 (S2): Cooperative navigation is responsible for driving the centroid to a target point, taking into account preservation of formation, team obstacle avoidance, and velocity limit. Thus two hierarchies are proposed. A higher level for solving the formation task, and a second priority for navigation task. From (23), the HQP problem

can be written as following:

$$\begin{aligned} \text{Level 1 : } & \mathbf{u}_1 = \arg \min_{\mathbf{x}} \|\mathbf{A}_1 \mathbf{x} - \mathbf{b}_1\|^2 \\ & \text{s.t.} \quad \mathbf{C}_1 \mathbf{x} \leq \mathbf{d}_1 \\ \text{Level 2 : } & \mathbf{u} = \arg \min_{\mathbf{x}} \|\mathbf{A}_2 \mathbf{x} - \mathbf{b}_2\|^2 \\ & \text{s.t.} \quad \mathbf{A}_1 \mathbf{x} = \mathbf{A}_1 \mathbf{u}_1 \\ & \quad \mathbf{C}_2 \mathbf{x} \leq \mathbf{d}_2 \end{aligned} \quad (24)$$

The program always starts from S1 behavior until inter-distance errors between robots fall below a certain threshold before switching to S2. In addition, if there is any sudden increase of the inter-distance errors above a threshold, the controller will transition back to S1 to ensure the formation before moving again.

Table I summarizes the overall control law of the system. The control input is always a solution from the last hierarchy, which is solved using the pseudo-inverse approach in order to avoid having multiple sets of solutions.

IV. RESULTS

A. Simulations

Two simulations are done by using Gazebo and ROS. In the first simulation, we compare the proposed control with a widely use leader-follower approach for formation from [11]. The load-carrying strategies are similar. However, our approach doesn't require any arm to be mounted on top of the mobile platforms. The second simulation combined S1 and S2 behaviors in order to have the desired formation shape before starting to navigate in an environment with obstacles.

1) *Simulation I*: In this simulation, a group of four robots has to move a load from an initial point of (0.0m, 0.0m) to a goal point of (5.0m, 3.0m). Fig. 6 clearly shows an advantage of the proposed control. Because the inter-distances can be well-maintained (Fig. 6b), a simple pin-hole system can be used in order to connect the load. In contrast, followers of [11] response only to the leader's motion, thus there is no interaction among followers as shown in Fig. 6d, in which Robot1 is the leader.

2) *Simulation II*: Fig. 8 shows a simulation environment, in which there are three obstacles: a round, a square, and a rotated square. The desired formation shape is illustrated in Fig. 9. The behavior S2 is triggered as soon as the norm of

TABLE I: Hierarchy levels of tasks for the two behavior states

	S1: $\mathbf{u} \equiv \mathbf{u}_f$	S2: $\mathbf{u} \equiv \mathbf{u}_n$
Level 1	$\begin{aligned} \min_{\mathbf{u}_f} & \left\ \mathbf{J}_f \mathbf{J}_x \mathbf{u}_f + \lambda_f (\mathbf{d}_f - \mathbf{d}_f^*) \right\ ^2 \\ \text{s.t.} & \mathbf{J}_{o_i} \mathbf{J}_x \mathbf{u}_i \geq -\lambda_{o_i} (\mathbf{d}_{o_i} - \mathbf{d}_{s_i}) \\ & \mathbf{J}_{lim} \mathbf{u}_f \leq \boldsymbol{\omega}_{lim} \end{aligned}$	$\begin{aligned} \min_{\mathbf{u}_f} & \left\ \mathbf{J}_f \mathbf{J}_x \mathbf{u}_f + \lambda_f (\mathbf{d}_f - \mathbf{d}_f^*) \right\ ^2 \\ \text{s.t.} & \mathbf{J}_{lim} \mathbf{u}_f \leq \boldsymbol{\omega}_{lim} \end{aligned}$
Level 2	none	$\begin{aligned} \min_{\mathbf{u}_n} & \left\ \mathbf{J}_n \mathbf{J}_x \mathbf{u}_n + \begin{pmatrix} \lambda_p (\mathbf{p}_c - \mathbf{p}_d) \\ \lambda_\theta (\theta_c - \theta_d) \end{pmatrix} \right\ ^2 \\ \text{s.t.} & \mathbf{J}_f \mathbf{J}_x \mathbf{u}_n = \mathbf{J}_f \mathbf{J}_x \mathbf{u}_f \\ & \mathbf{J}_{o_t} \mathbf{J}_x \mathbf{u}_n \geq -\lambda_{o_t} (\mathbf{d}_{o_t} - \mathbf{d}_{s_t}) \\ & \mathbf{J}_{lim} \mathbf{u}_n \leq \boldsymbol{\omega}_{lim} \end{aligned}$

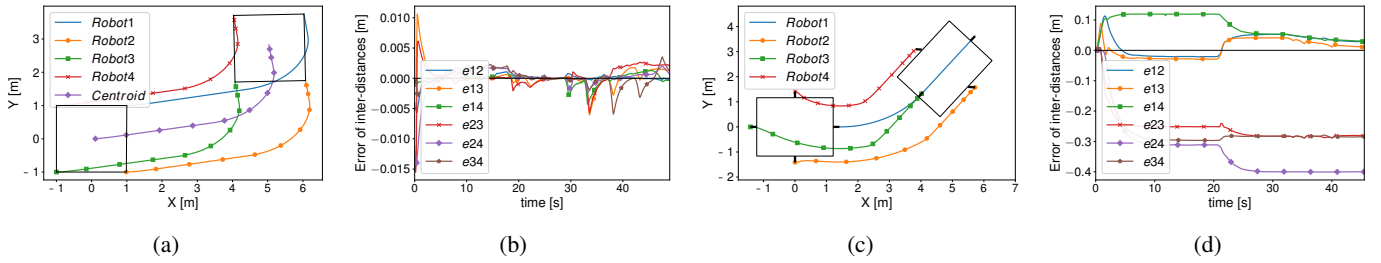


Fig. 6: Plots of the simulation I: robots' trajectories and errors of inter-distances of the proposed control ((a)-(b)), and of the leader-follower control ((c)-(d)). Solid black dash in (c) represents the arm that connects each robot to load's side.

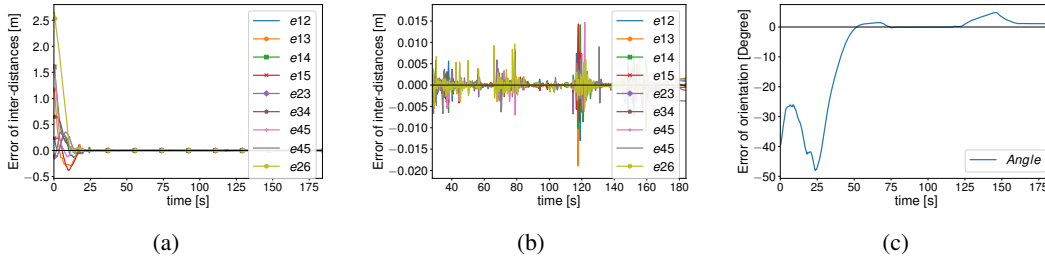


Fig. 7: Plots of the simulation II: errors of inter-distance between robots (a), magnified version (b), and heading error (c).

the inter-distance errors, $\|\mathbf{d}_f - \mathbf{d}_f^*\|$, is smaller than 0.02 m. From Fig. 7a, we can see that the inter-distance errors converge to form the formation (S1) at around $t = 30$ s. After that, the robots start to move to a goal point with a desired formation heading of -90° . The shape is preserved throughout the navigation with a maximum inter-distance error of ± 0.018 m even though they have to navigate through obstacles (Fig. 7b). The team's orientation error is also well-converged with an error of less than 5° as shown in Fig. 7c.

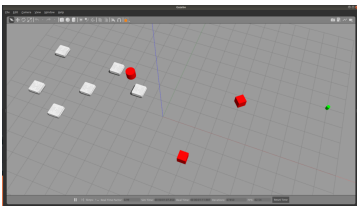


Fig. 8: Simulation environment: six robots, a goal point (green column), and three obstacles (red columns).

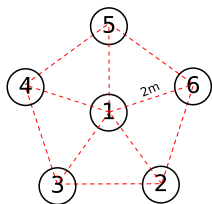


Fig. 9: A desired pentagon shape with 6 robots.

B. Experiment

Two experiments to validate the proposed control for the formation and cooperative navigation are shown using four robots. In both cases, a square shape formation of sides of 2m is desired. Here S1 and S2 are separated because the group

has to first get into a good formation shape, then the payload is manually attached before starting the navigation. ROS is used as the middleware. Each robot has its own independent *ROS_MASTER*. There is no external sensor to locate each robot accurately. The self-localization in a map is done using an onboard lidar sensor with an estimated error of 0.034m. A video of the simulations and experiments can be found following these links ¹.

Fig. 10 shows a scenario, which the goal is to have the desired shape at the end of the formation while also avoiding any un-mapped obstacle. The convergence of the inter-distances errors can be seen in Fig. 11a. The desired shape is well-reached (Fig. 11b) even though there is an obstacle in front of Robot3 thanks to the individual obstacle avoidance inequality constraint.

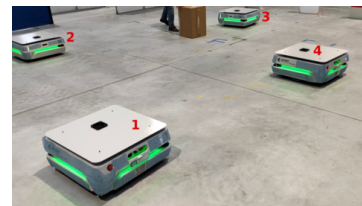


Fig. 10: Formation experiment with an obstacle.

Fig. 12 shows the navigation scenario of the experiment. A plate with a payload of 20kg is added to the system. The team has to reach a target point and come back while trying not to collide with the two obstacles. The formation shape is kept during navigation, as demonstrated by the plot of the inter-distance errors (Fig. 11c). Even though the self-localization as well as the velocity tracking low-level control of each robot

¹<https://youtu.be/LYJWcOHd1L8>
¹<https://youtu.be/E1OdhVi11DU>

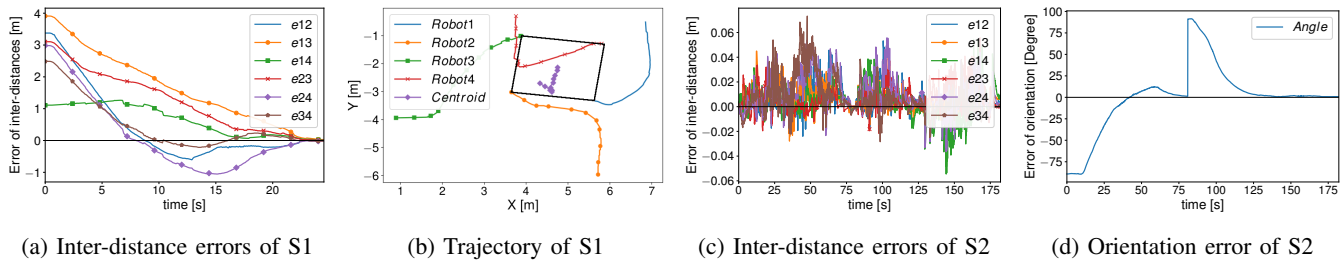


Fig. 11: Plots of the experimental results: formation (two lefts) and navigation (two rights).

are not accurately perfect, the maximum error is just about $\pm 0.06\text{m}$. The fleet starts at $(2.5\text{m}, -2.0\text{m}, 0^\circ)$, and moves to a goal pose of $(12\text{m}, -4.0\text{m}, 90^\circ)$. It reaches the desired pose at around $t = 80\text{s}$ before coming back to the initial pose. The team's desired heading angle is also reached for both journey as the error converges to zero (Fig. 11d).

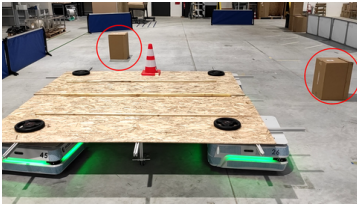


Fig. 12: Navigation scenario with two obstacles (circled in red) in the environment.

V. CONCLUSION

In this paper, we demonstrated a method for cooperative object transportation with obstacle avoidance based on an optimization approach. Different tasks were formulated to either equality or inequality constraints, which are solved using the hierarchical quadratic programming method. Validations of the proposed approach are shown in simulation and actual experiments using HUSKY commercial robots. Un-mapped obstacles in the environment can be detected and evaded during runtime, thus a reactive control. The control, however, can be at risk of getting stuck in a local minima due to the obstacle's shape, for instance, the "U" shape. To solve this, a global planner can be used in order to generate a set of waypoints that will be tracked by the navigation task.

Future works will be focused on the high-level coordination of MRS. In a real-world scenario where multiple resources (robots) and mission requirements occur, the group has to be able to autonomously coordinate, form coalitions, and allocate tasks among themselves. By combining this coordination level with our proposed execution level, we can ensure a solution to the cooperative transportation problem in real applications requiring little operator's interventions.

REFERENCES

- [1] J.-W. Kwon and D. Chwa, "Hierarchical Formation Control Based on a Vector Field Method for Wheeled Mobile Robots," *IEEE Trans. on Robotics*, vol. 28, no. 6, pp. 1335–1345, dec 2012.
- [2] E. Tuci, M. H. M. Alkilabi, and O. Akanyeti, "Cooperative Object Transport in Multi-Robot Systems: A Review of the State-of-the-Art," *Frontiers In Robotics and Ai*, vol. 5, may 2018.

- [3] M. H. Alkilabi, A. Narayan, and E. Tuci, "Cooperative object transport with a swarm of e-puck robots: robustness and scalability of evolved collective strategies," *Swarm Intelligence*, vol. 11, pp. 185–209, 2017.
- [4] W. Wan, B. Shi, Z. Wang, and R. Fukui, "Multirobot Object Transport via Robust Caging," *IEEE Trans. on Systems, Man, and Cybernetics: Systems*, vol. 50, pp. 270–280, jan 2020.
- [5] J. Chen, M. Gauci, W. Li, A. Kolling, and R. Groß, "Occlusion-Based Cooperative Transport with a Swarm of Miniature Mobile Robots," *IEEE Trans. on Robotics*, vol. 31, pp. 307–321, apr 2015.
- [6] Z. Wang and M. Schwager, "Kinematic multi-robot manipulation with no communication using force feedback," in *2016 IEEE Int. Conf. on Robotics and Automation (icra)*, Stockholm, Sweden, may 2016.
- [7] T. Machado, T. Malheiro, S. Monteiro, W. Erlhagen, and E. Bicho, "Multi-constrained joint transportation tasks by teams of autonomous mobile robots using a dynamical systems approach," *2016 IEEE Int. Conf. on Robotics and Automation (icra)*, may 2016.
- [8] E-COBOT, "HUSKY," 2021.
- [9] O. Saif, I. Fantoni, and A. Zavala-Río, "Distributed integral control of multiple UAVs: precise flocking and navigation," *IET Control Theory & Applications*, vol. 13, no. 13, pp. 2008–2017, sep 2019.
- [10] A. Loria, J. Dasdemir, and N. Alvarez Jarquin, "Leader-Follower Formation and Tracking Control of Mobile Robots Along Straight Paths," *IEEE Trans. on Control Systems Technology*, vol. 24, no. 2, pp. 727–732, 2016.
- [11] X. Yang, K. Watanabe, K. Izumi, and K. Kiguchi, "A decentralized control system for cooperative transportation by multiple non-holonomic mobile robots," *Int. Journal of Control*, vol. 77, pp. 949–963, aug 2004.
- [12] J. C. Derenick and J. R. Spletzer, "Convex optimization strategies for coordinating large-scale robot formations," *IEEE Trans. on Robotics*, vol. 23, pp. 1252–1259, dec 2007.
- [13] J. Alonso-Mora, S. Baker, and D. Rus, "Multi-robot formation control and object transport in dynamic environments via constrained optimization," *Int. Journal of Robotics Research*, pp. 1000–1021, 2017.
- [14] M. A. Trujillo, H. M. Becerra, D. Gómez-Gutiérrez, J. Ruiz-León, and A. Ramírez-Treviño, "Priority Task-Based Formation Control and Obstacle Avoidance of Holonomic Agents with Continuous Control Inputs," *Ifac-papersonline*, vol. 51, pp. 216–222, 2018.
- [15] G. Antonelli, F. Arrichiello, and S. Chiaverini, "Experiments of formation control with multirobot systems using the null-space-based behavioral control," *IEEE Trans. on Control Systems Technology*, vol. 17, no. 5, pp. 1173–1182, 2009.
- [16] A. Escande, N. Mansard, and P.-b. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *Int. Journal of Robotics Research*, vol. 33, pp. 1006–1028, 2014.
- [17] H. M. Pérez-Villeda, G. Arechavaleta, and A. Morales-Díaz, "Multi-vehicle coordination based on hierarchical quadratic programming," *Control Engineering Practice*, vol. 94, jan 2020.
- [18] O. Kermorgant and F. Chaumette, "Dealing with constraints in sensor-based robot control," *IEEE Trans. on Robotics*, pp. 244–257, feb 2014.
- [19] W. Ren and R. W. Beard, *Distributed Consensus in Multi-vehicle Cooperative Control*. London: Springer London, 2008.
- [20] O. Kanoun, F. Lamiroux, and P. B. Wieber, "Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task," *IEEE Trans. on Robotics*, vol. 27, no. 4, pp. 785–792, 2011.