A Nonlinear Model Predictive Control Scheme for Cooperative Manipulation with Singularity and Collision Avoidance

Alexandros Nikou, Christos Verginis, Shahab Heshmati-alamdari and Dimos V. Dimarogonas

Abstract— **This paper addresses the problem of cooperative transportation of an object rigidly grasped by N robotic agents. In particular, we propose a Nonlinear Model Predictive Control (NMPC) scheme that guarantees the navigation of the object to a desired pose in a bounded workspace with obstacles, while complying with certain input saturations of the agents. Moreover, the proposed methodology ensures that the agents do not collide with each other or with the workspace obstacles as well as that they do not pass through singular configurations. The feasibility and convergence analysis of the NMPC are explicitly provided. Finally, simulation results illustrate the validity and efficiency of the proposed method.**

I. Introduction

Over the last years, multi-agent systems have gained a significant amount of attention, due to the advantages they offer with respect to single-agent setups. In the case of robotic manipulation and object transportation, difficult tasks involving heavy payloads as well as challenging maneuvers necessitate the employment of multiple robots.

Early works related to cooperative manipulation develop control architectures where the robotic agents communicate and share information with each other as well as completely decentralized schemes, where each agent uses only local information or observers, avoiding potential communication delays [1]–[6]. Impedance and force/motion control constitutes the most common methodology used in the related literature [1], [7]–[14]. However, most of the aforementioned works employ force/torque sensors to acquire knowledge of the manipulator-object contact forces/torques, which, however, may result to performance decline due to sensor noise or mounting difficulties. Recent technological advances allow to manipulator grippers to grasp rigidly certain objects (see e.g., [15]), which, as shown in this work, can render the use of force/torque sensors unnecessary.

Furthermore, in manipulation tasks, such as pose/force or trajectory tracking, collision with obstacles of the environment has been dealt with only by exploiting the extra degrees of freedom that appear in over-actuated robotic agents. Potential field-based algorithms may suffer from local minima and navigation functions [16] cannot be extended to multi-agent second order dynamical systems in a trivial way. Moreover, these methods usually result in high control input values near obstacles that need to be avoided, which might conflict the saturation of the actual motor inputs.

Another important property that concerns robotic manipulators is the singularities of the Jacobian matrix, which maps the joint velocities of the agent to a 6D vector of generalized velocities. Such *singular kinematic* configurations, that indicate directions towards which the agent cannot move, must be always avoided, especially when dealing with task-space control in the end-effector [17]. In the same vein, *representation* singularities can also occur in the mapping from coordinate rates to angular velocities of a rigid body.

In this work, we aim to address the problem of cooperative manipulation of an object in a bounded workspace with obstacles. In particular, given N agents that rigidly grasp an object, we design control inputs for the navigation of the object to a final pose, while avoiding inter-agent collisions as well as collisions with obstacles. Moreover, we take into account constraints that emanate from control input saturation as well kinematic and representation singularities.

For the design of a stabilizing feedback control law for each robot, such that the desired specifications are met, while satisfying constraints on the controls and the states, one would ideally look for a closed loop solution for the feedback law satisfying the constraints while optimizing the performance. However, typically the optimal feedback law cannot be found analytically, even in the unconstrained case, since it involves the solution of the corresponding Hamilton-Jacobi-Bellman partial differential equations. One approach to address this problem is the repeated solution of an open-loop optimal control problem for a given state. Control approaches using this strategy are referred to as Nonlinear Model Predictive Control (NMPC) (see e.g. [18]–[25]) which we aim to use in this work for the problem of the constraint cooperative manipulation of an object which is rigidly grasped by N agents. To the best of the authors' knowledge, NMPC approaches for cooperative manipulation have not appeared in the related literature. Due to space constraints, a more detailed version of this paper that contains omitted proofs of lemmas/theorems, can be found in [26].

II. Notation and Preliminaries

The set of positive integers is denoted as $\mathbb{N}$ and the real n-coordinate space, with $n \in \mathbb{N}$, as $\mathbb{R}^n$; $\mathbb{R}^n_{\geq 0}$ and $\mathbb{R}^n_{>0}$ are the sets of real n-vectors with all elements nonnegative and

Alexandros Nikou, Christos Verginis and Dimos V. Dimarogonas are with the ACCESS Linnaeus Center, School of Electrical Engineering, KTH Royal Institute of Technology, SE-100 44, Stockholm, Sweden and with the KTH Center for Autonomous Systems. Email: {anikou, cverginis, dimos}@kth.se. Shahab Heshmati-alamdari is with the Control Systems Lab, Department of Mechanical Engineering, National Technical University of Athens, 9 Heroon Polytechniou Street, Zografou 15780, Athens, Greece. Email: {shahab}@mail.ntua.gr. This work was supported by the H2020 ERC Starting Grant BUCOPHSYS, the Swedish Research Council (VR), the Knut och Alice Wallenberg Foundation, the European Union's Horizon 2020 Research and Innovation Programme under the Grant Agreement No. 644128 (AEROWORKS) and the EU H2020 Research and Innovation Programme under GA No. 731869 (Co4Robots).

positive, respectively. The notation $\mathbb{R}^{n\times n}_{\geq 0}$ and $\mathbb{R}^{n\times n}_{>0}$, with $n \in \mathbb{N}$, stands for positive semi-definite and positive definite matrices, respectively. Moreover, $\|x\|$ is the Euclidean norm of a vector $x \in \mathbb{R}^n$. Given the sets S_1, S_2, the Minkowski addition is denoted by $\oplus$, and is defined by $S_1 \oplus S_2 = \{s_1 + s_2 : s_1 \in S_1, s_2 \in S_2\}$. The $n \times n$ identity matrix and the $n\times m$ matrix with zero entries are denoted by I_n, and $0_{n\times m}$, respectively, with $n, m \in \mathbb{N}$. The largest singular value of matrix $A \in \mathbb{R}^{n\times m}$ is denoted as $\sigma_{\max}(A)$. The vector connecting the origins of coordinate frames $\{A\}$ and $\{B\}$ expressed in frame $\{C\}$ coordinates in 3-D space is denoted as $p^C_{B/A} = [x_{B/A}, y_{B/A}, z_{B/A}]^\top \in \mathbb{R}^3$. Given $a \in \mathbb{R}^3, S(a)$ is the skew-symmetric matrix defined according to $S(a)b = a \times b$. We further denote as $\eta_{A/B} = [\phi_{A/B}, \theta_{A/B}, \psi_{A/B}]^\top \in \mathbb{T}^3 \subseteq \mathbb{R}^3$ the x-y-z Euler angles representing the orientation of frame $\{A\}$ with respect to frame $\{B\}$, with $\phi_{A/B}, \psi_{A/B} \in [-\pi, \pi]$ and $\theta_{A/B} \in [-\frac{\pi}{2}, \frac{\pi}{2}]$, where $\mathbb{T}^3$ is the 3-D torus; Moreover, $R^B_A \in SO(3)$ is the rotation matrix associated with the same orientation and $SO(3)$ is the 3-D rotation group. The angular velocity of frame $\{B\}$ with respect to $\{A\}$, expressed in frame $\{C\}$ coordinates, is denoted as $\omega^C_{B/A} \in \mathbb{R}^3$ and it holds that $\dot{R}^B_A = S(\omega^A_{B/A})R^B_A$. We further define the sets $\mathbb{M} = \mathbb{R}^3 \times \mathbb{T}^3$, $\mathcal{N} = \{1, \dots, N\}$. We define also the set $\mathcal{O}_z \triangleq \mathcal{O}(c_z, \beta_{1,z}, \beta_{2,z}, \beta_{3,z}) = \big\{p \in \mathbb{R}^3 : (p - c_z)^\top P(p - c_z) \leq 1\big\}$, as the set of an *ellipsoid* in 3D, where $c_z \in \mathbb{R}^3$ is the center of the ellipsoid, $\beta_{1,z}, \beta_{2,z}, \beta_{3,z} \in \mathbb{R}_{>0}$ the lengths of its three semi-axes and $z \geq 1$ is an index term. The eigenvectors of matrix P define the principal axes of the ellipsoid, and the eigenvalues of P are: $\beta^{-2}_{1,z}, \beta^{-2}_{2,z}$ and $\beta^{-2}_{3,z}$. For notational brevity, when a coordinate frame corresponds to an inertial frame of reference $\{I\}$, we will omit its explicit notation (e.g., $p_B = p^I_{B/I}, \omega_B = \omega^I_{B/I}, R_A = R^I_A$). Finally, all vector and matrix differentiations will be with respect to an inertial frame $\{I\}$, unless otherwise stated.

III. Problem Formulation

Consider a bounded and convex workspace $\mathcal{W} \subseteq \mathbb{R}^3$ consisting of N robotic agents rigidly grasping an object, as shown in Fig. 1, and Z obstacles described by the ellipsoids $\mathcal{O}_z, z \in \mathcal{Z} = \{1, \dots, Z\}$. The free space is denoted as $\mathcal{W}_{\text{free}} = \mathcal{W} \backslash \bigcup_{z\in\mathcal{Z}} \mathcal{O}_z$. The agents are considered to be fully actuated and they consist of a base that is able to move around the workspace (e.g., mobile or aerial vehicle) and a robotic arm. The reference frames corresponding to the i-th end-effector and the object's center of mass are denoted with $\{E_i\}$ and $\{O\}$, respectively, whereas $\{I\}$ corresponds to an inertial reference frame. The rigidity of the grasps implies that the agents can exert any forces/torques along every direction to the object. We consider that the position and velocity of the state of the agents as well as the geometric parameters of the agents and the object are known. Moreover, no interaction force/torque measurements is required.

A. System model

We denote by $q_i : \mathbb{R}_{\geq 0} \to \mathbb{R}^{n_i}$ the joint space variables of agent $i \in \mathcal{N}$, with $n_i = n_{\alpha_i} + 6$, $q_i(t) = [p^\top_{B_i}(t), \eta^\top_{B_i}(t), \alpha^\top_i(t)]^\top$, where $p_{B_i} = [x_{B_i}, y_{B_i}, z_{B_i}]^\top : \mathbb{R}_{\geq 0} \to \mathbb{R}^3, \eta_{B_i} = [\phi_{B_i}, \theta_{B_i}, \psi_{B_i}]^\top : \mathbb{R}_{\geq 0} \to \mathbb{T}^3 \subseteq \mathbb{R}^3$ is the position and Euler-angle orientation of the agent's base, and $\alpha_i : \mathbb{R}_{\geq 0} \to \mathbb{R}^{n_{\alpha_i}}, n_{\alpha_i} > 0$, are the degrees of freedom (i.e., joints) of the robotic arm. The overall joint space configuration vector is denoted as $q = [q^\top_1, \dots, q^\top_N]^\top \in \mathbb{R}^n$, with $n = \sum_{i\in\mathcal{N}} n_i$. In addition, we denote as $p_{E_i} : \mathbb{R}^{n_i} \to \mathbb{R}^3, \eta_{E_i} : \mathbb{R}^{n_i} \to \mathbb{T}^3 \subseteq \mathbb{R}^3$ the position and Euler-angle orientation of agent i's end-effector. Let also $v_i : \mathbb{R}^{n_i} \times \mathbb{R}^{n_i} \to \mathbb{R}^6$ denote the velocity of agent i's end-effector, with $v_i(q_i, \dot{q}_i) = [\dot{p}^\top_{E_i}, \omega^\top_{E_i}]^\top$, whereas $\dot{p}_{B_i}, \omega_{B_i} : \mathbb{R}^{n_i} \times \mathbb{R}^{n_i} \to \mathbb{R}^3$ are the linear and angular velocity, respectively, of the base.

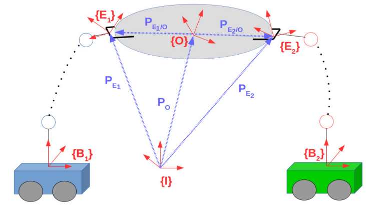

Fig. 1: Two robotic arms rigidly grasping an object with the corresponding frames.

We consider that we have access to each agent's state q_i as well as $\dot{p}^{B_i}_{B_i}, \omega^{B_i}_{B_i}$, and $\dot{\alpha}_i$ via on-board sensors. Then, $\dot{p}_{B_i}, \omega_{B_i}$ can be obtained via $\dot{p}_{B_i} = R_{B_i}(\eta_{B_i})\dot{p}^{B_i}_{B_i}$, $\omega_{B_i} = R_{B_i}(\eta_{B_i})\omega^{B_i}_{B_i}$, where $R_{B_i} : \mathbb{T}^3 \to SO(3)$ is the rotation matrix of the ith base. Moreover, $\dot{\eta}_{B_i}$ is related to ω_{B_i} via $\omega_{B_i} = J_{B_i}(\eta_{B_i})\dot{\eta}_{B_i}$, where $J_{B_i} : \mathbb{T}^3 \to \mathbb{R}^{3\times 3}$, with

$$J_{B_i}(\eta_{B_i}) = \begin{bmatrix} 1 & 0 & \sin(\theta_{B_i}) \\ 0 & \cos(\phi_{B_i}) & -\cos(\theta_{B_i})\sin(\phi_{B_i}) \\ 0 & \sin(\phi_{B_i}) & \cos(\theta_{B_i})\cos(\phi_{B_i}) \end{bmatrix}. \quad (1)$$

The pose of the ith end-effector can be computed via $p_{E_i}(q_i) = p_{B_i} + R_{B_i}(\eta_{B_i})k_{p_i}(\alpha_i), \eta_{E_i}(q_i) = k_{\eta_i}(\eta_{B_i}, \alpha_i)$, where $k_{p_i} : \mathbb{R}^{n_{\alpha_i}} \to \mathbb{R}^3, k_{\eta_i} : \mathbb{T}^3 \times \mathbb{R}^{n_{\alpha_i}} \to \mathbb{T}^3$ are the forward kinematics of the robotic arm [17]. Then, v_i can be computed as

$$v_i(q_i, \dot{q}_i) = \begin{bmatrix} \dot{p}_{B_i} - S(R_{B_i}k_{p_i})\omega_{B_i} + R_{B_i}\frac{\partial k_{p_i}}{\partial \alpha_i}\dot{\alpha}_i \\ \omega_{B_i} + R_{B_i}J_{A_i}\dot{\alpha}_i \end{bmatrix}, \quad (2)$$

where $J_{A_i} : \mathbb{R}^{n_{\alpha_i}} \to \mathbb{R}^{3\times n_{\alpha_i}}$ is the angular Jacobian of the robotic arm with respect to the agent's base [17]. The differential kinematics (2) can be written as

$$v_i(q_i, \dot{q}_i) = \begin{bmatrix} \dot{p}_{E_i}(q_i, \dot{q}_i) \\ \omega_{E_i}(q_i, \dot{q}_i) \end{bmatrix} = J_i(q_i)\dot{q}_i, \quad (3)$$

where $J_i : \mathbb{R}^{n_i} \to \mathbb{R}^{6\times n_i}$ is the agent's geometric Jacobian matrix [17], with

$$J_i(q_i) = \begin{bmatrix} I_3 & -S(R_{B_i}(\eta_{B_i})k_{p_i}(\alpha_i))J_{B_i}(\eta_{B_i}) & R_{B_i}(\eta_{B_i})\frac{\partial k_{p_i}(\alpha_i)}{\partial \alpha_i} \\ 0_{3\times 3} & J_{B_i}(\eta_{B_i}) & R_{B_i}(\eta_{B_i})J_{A_i}(q_i) \end{bmatrix}.$$

Remark 1. Note that J_{B_i} becomes singular at representation singularities, when $\theta_{B_i} = \pm\frac{\pi}{2}$ and J_i becomes singular at kinematic singularities defined by the set $\mathcal{Q}_i = \{q_i \in \mathbb{R}^{n_i} : \det(J^\top_i J_i) = 0\}, i \in \mathcal{N}$. In the following, we will aim at guaranteeing that q_i will always be in the closed set $\widetilde{\mathcal{Q}}_i =$

$\{q_i \in \mathbb{R}^{n_i} : |\det(J_i^\top J_i)| \geq \varepsilon > 0\}, i \in \mathcal{N}$, for a small positive constant ε.

The joint-space dynamics for agent $i \in \mathcal{N}$ can be computed using the Lagrangian formulation [17]:

$$B_i(q_i)\ddot{q}_i + N_i(q_i, \dot{q}_i)\dot{q}_i + g_{q_i}(q_i) = \tau_i - J_i^\top \lambda_i, \quad (4)$$

where $B_i : \mathbb{R}^{n_i} \to \mathbb{R}^{n_i \times n_i}$ is the joint-space positive definite inertia matrix, $N_i : \mathbb{R}^{n_i} \times \mathbb{R}^{n_i} \to \mathbb{R}^{n_i \times n_i}$ represents the joint-space Coriolis matrix, $g_{q_i} : \mathbb{R}^{n_i} \to \mathbb{R}^{n_i}$ is the joint-space gravity vector, $\lambda_i \in \mathbb{R}^6$ is the generalized force vector that agent i exerts on the object and $\tau_i \in \mathbb{R}^{n_i}$ is the vector of generalized joint-space inputs, with $\tau_i = [\lambda_{B_i}^\top, \tau_{\alpha_i}^\top]^\top$, where $\lambda_{B_i} = [f_{B_i}^\top, \mu_{B_i}^\top]^\top \in \mathbb{R}^6$ is the generalized force vector on the center of mass of the agent's base and $\tau_{\alpha_i} \in \mathbb{R}^{n_{\alpha_i}}$ is the torque inputs of the robotic arms' joints. By inverting (4) and using (3) and its derivative, we can obtain the corresponding task-space agent dynamics [17]:

$$M_i(q_i)\dot{v}_i + C_i(q_i, \dot{q}_i)v_i + g_i(q_i) = u_i - \lambda_i. \quad (5)$$

The task-space input wrench u_i can be translated to the joint space inputs $\tau_i \in \mathbb{R}^{n_i}$ via $\tau_i = J_i^\top(q_i)u_i + (I_{n_i} - J_i^\top(q_i)\bar{J}_i^\top(q_i))\tau_{i_0}$, where $\bar{J}_i$ is a generalized inverse of J_i [17]. The term τ_{i_0} concerns over-actuated agents and does not contribute to end-effector forces.

We define by $\mathcal{A}_i(q_i) \triangleq \mathcal{O}_i, i \in \mathcal{N}$, the ellipsoid that bounds the ith agent's volume with the corresponding centers c_i and semi-axes $\beta_{i,1}, \beta_{i,2}, \beta_{i,3}$, i.e., the workspace of the arm of agent i [17] enlarged so that it includes the ith base. Note that $\mathcal{A}_i$ depends on q_i and can be explicitly found.

Regarding the object, we denote as $x_O : \mathbb{R}_{\geq 0} \to \mathbb{M}$, $v_O : \mathbb{R}_{\geq 0} \to \mathbb{R}^6$ the pose and velocity of the object's center of mass, with $x_O(t) = [p_O^\top(t), \eta_O^\top(t)]^\top$, $p_O(t) = [x_O(t), y_O(t), z_O(t)]^\top$, $\eta_O(t) = [\phi_O(t), \theta_O(t), \psi_O(t)]^\top$ and $v_O(t) = [\dot{p}_O^\top(t), \omega_O^\top(t)]^\top$. The second order dynamics of the object are given by:

$$\dot{x}_O(t) = J_{O_r}^{-1}(x_O)v_O(t), \quad (6a)$$

$$\lambda_O = M_O(x_O)\dot{v}_O(t) + C_O(x_O, v_O)v_O(t) + g_O(x_O), \quad (6b)$$

where $M_O : \mathbb{M} \to \mathbb{R}^{6\times 6}$ is the positive definite inertia matrix, $C_O : \mathbb{M} \times \mathbb{R}^6 \to \mathbb{R}^{6\times 6}$ is the Coriolis matrix, $g_O : \mathbb{M} \to \mathbb{R}^6$ is the gravity vector, which are derived from the Newton-Euler formulation. In addition, $J_{O_r} : \mathbb{M} \to \mathbb{R}^{6\times 6}$ is the object representation Jacobian $J_{O_r}(x_O) = \text{diag}\{I_3, J_{O_{r,\theta}}(\eta_O)\}$, where $J_{O_{r,\theta}}$ has the same structure as (1), and is singular when $\theta_O = \pm\frac{\pi}{2}$. Finally, $\lambda_O \in \mathbb{R}^6$ is the force vector acting on the object's center of mass. Also, similarly to the robotic agents, we define by $\mathcal{C}_O(x_O) \triangleq \mathcal{O}_o$, as the bounding ellipsoid of the object.

Consider N robotic agents rigidly grasping an object. Then, in view of Fig. 1, we have that

$$p_{E_i}(q_i(t)) = p_O(t) + p_{E_i/O}(q_i) = p_O(t) + R_{E_i}(t)p_{E_i/O}^{E_i}, \quad (7a)$$

$$\eta_{E_i}(q_i(t)) = \eta_O(t) + \eta_{E_i/O}, \quad (7b)$$

$\forall i \in \mathcal{N}$, where $p_{E_i/O}^{E_i}$ represents the constant distance and $\eta_{E_i/O}$ the relative orientation offset between the ith agent's end-effector and the object's center of mass, which are considered known. The grasp rigidity implies that $\omega_{E_i} = \omega_O$, $\forall i \in \mathcal{N}$. Therefore, by differentiating (7), we obtain

$$v_i(q_i, \dot{q}_i(t)) = J_{O_i}(q_i)v_O(t), \quad (8)$$

where $J_{O_i} : \mathbb{R}^n \to \mathbb{R}^{6\times 6}$ is a smooth mapping representing the Jacobian from the object to the i-th agent: $J_{O_i}(q_i) = \begin{bmatrix} I_3 & S(p_{O/E_i}(q_i)) \\ 0_{3\times 3} & I_3 \end{bmatrix}$, and is always full rank due to the grasp rigidity.

Remark 2. Since the geometric object parameters $p_{E_i/O}^{E_i}$ and $\eta_{E_i/O}$ are known, we can compute p_O, η_O and v_O simply by inverting (7) and (8), respectively, without employing any sensory data for the object. In the same vein, we can also compute the object's bounding ellipsoid $\mathcal{C}_O$, which depends on q.

The Kineto-statics duality [17] along with the grasp rigidity suggest that the force λ_O acting on the object center of mass and the generalized forces $\lambda_i, i \in \mathcal{N}$, exerted by the agents at the contact points are related through

$$\lambda_O = G^T(q)\bar{\lambda}, \quad (9)$$

where $\bar{\lambda} = [\lambda_1^\top, \cdots, \lambda_N^\top]^\top \in \mathbb{R}^{6N}$ and $G : \mathbb{R}^n \to \mathbb{R}^{6N\times 6}$ is the grasp matrix, with $G(q) = [J_{O_1}^\top, \cdots, J_{O_N}^\top]^\top$.

Next, we substitute (8) and its derivative in (5) and by employing (6) and (9), we obtain in vector form after some straightforward calculations:

$$\widetilde{M}(q)\dot{v}_O + \widetilde{C}(q, \dot{q})v_O + \widetilde{g}(q) = G^T(q)u, \quad (10)$$

where $\widetilde{M}(q) = M_O(q) + G^\top(q)\bar{M}(q)G(q)$, $\widetilde{C}(q, \dot{q}) = C_O(q) + G^\top(q)\bar{M}(q)\dot{G}(q, \dot{q}) + G^\top(q)\bar{C}(q)G(q)$, and $\widetilde{g}(q) = g_O(q) + G^\top(q)\bar{g}(q)$ and we have used the stack forms $\bar{M} = \text{diag}\{[M_i]_{i\in\mathcal{N}}\}$, $\bar{C} = \text{diag}\{[C_i]_{i\in\mathcal{N}}\}$, $\bar{g} = [g_1^\top, \ldots, g_N^\top]^\top$, and $u = [u_1^\top, \ldots, u_N^\top]^\top$, as well as the dependence of x_O on q from (7).

Remark 3. Note that the agents dynamics under consideration hold for generic robotic agents comprising of a moving base and a robotic arm. Hence, the considered framework can be applied for mobile, aerial, or underwater manipulators.

We can now formulate the problem considered in this work:

Problem 1. Consider N robotic agents rigidly grasping an object, governed by the coupled dynamics (10). Given the desired pose $x_{O,\text{des}}$, design the control input $u : \mathbb{R}_{\geq 0} \to \mathbb{R}^{6N}$ such that $\lim_{t\to\infty} x_O(t) = x_{O,\text{des}}$, while ensuring the satisfaction of the following collision avoidance and singularity properties: 1)$\mathcal{A}_i(q_i) \cap \mathcal{O}_z = \emptyset, \forall i \in \mathcal{N}, z \in \mathcal{Z}$; 2)$\mathcal{C}_O(x_O) \cap \mathcal{O}_z = \emptyset, \forall z \in \mathcal{Z}$; 3)$\mathcal{A}_i(q_i) \cap \mathcal{A}_j(q_j) = \emptyset, \forall i, j \in \mathcal{N}, i \neq j$; 4)$-\frac{\pi}{2} < -\bar{\theta} \leq \theta_O \leq \bar{\theta} < \frac{\pi}{2}$; 5)$-\frac{\pi}{2} < -\bar{\theta} \leq \theta_{B_i} \leq \bar{\theta} < \frac{\pi}{2}$; 6)$q_i \in \widetilde{\mathcal{Q}}_i$, for a $0 < \bar{\theta} < \frac{\pi}{2}$, as well as the input and velocity magnitude and input constraints: $|\tau_{i_k}| \leq \bar{\tau}_i, |\dot{q}_{i_k}| \leq \bar{\dot{q}}_i, \forall k \in \{1, \ldots, n_i\}, i \in \mathcal{N}$, for some positive constants $\bar{\tau}_i, \bar{\dot{q}}_i, i \in \mathcal{N}$.

Specifications $1) - 3)$ in the aforementioned problem stand for collision avoidance between the agents, the objects, and the workspace obstacles and specifications $4) - 6)$ stand for representation and kinematic singularities.

In order to solve the aforementioned problem, we need the following reasonable assumption regarding the workspace:

Assumption 1. (Problem Feasibility Assumption) The distance between any pair of obstacles is sufficiently large such that the coupled system object-agents can navigate among them without collisions.

Define also the sets : $S_{i,O}(q) = \{q_i \in \mathbb{R}^{n_i} : \mathcal{A}_i(q_i) \cap \mathcal{O}_z \neq \emptyset, \forall z \in \mathcal{Z}\}$, $S_{i,A}(q) = \{q_i \in \mathbb{R}^{n_i} : \mathcal{A}_i(q_i) \cap \mathcal{A}_j(q_j) \neq \emptyset, \forall j \in \mathcal{N}\backslash\{i\}\}, \forall i \in \mathcal{N}$, as well as $S_O(x_O) = \{x_O \in \mathbb{M} : \mathcal{C}_O(x_O) \cap \mathcal{O}_z \neq \emptyset\}$, associated with the desired collision-avoidance properties.

IV. Problem Solution

In this section, a systematic solution to Problem 1 is introduced. Our overall approach builds on designing a Nonlinear Model Predictive control scheme the system of the manipulators and the object.

By employing (8), (3), and (10), we can write the coupled *nonlinear dynamics* of the system object-agents in compact form as:

$$\dot{x} = f(x, u), x(0) = x_0, \tag{11}$$

where $x = [x_O^\top, v_O^\top, q^\top]^\top \in \mathbb{R}^{n+12}$, $u \in \mathbb{R}^{6N}$, $f = [f_1^\top, f_2^\top, f_3^\top]^\top : \mathbb{R}^{(n+12)\times(6N)}$, with $f_1(x, u) = J_{O_r}^{-1}(x_O)v_O$, $f_2(x, u) = \widetilde{M}^{-1}(q)\Big[G^\top(q)u - \widetilde{C}(q, \dot{q})v_O - \widetilde{g}(q)\Big]$, $f_3(x, u) = \hat{J}(q)J_O(q)\widetilde{I}v_O$, $\hat{J}(q) = \text{diag}\Big\{\big[(J_i^\top J_i)^{-1}J_i^\top\big]_{i\in\mathcal{N}}\Big\} \in \mathbb{R}^{n\times 6N}$, $J_O(q) = \text{diag}\Big\{\big[J_{O_i}\big]_{i\in\mathcal{N}}\Big\} \in \mathbb{R}^{6N\times 6N}$ and $\widetilde{I} = [I_6, \cdots, I_6]^\top \in \mathbb{R}^{6N\times 6}$. Note that f is *locally Lipschitz continuous* in its domain since it is continuously differentiable in its domain. Next, we define the respective error $e : \mathbb{R}_{\geq 0} \to \mathbb{R}^{n+12}$ as

$$e(t) = x(t) - x_{\text{des}} \tag{12}$$

where $x_{\text{des}} = [x_{O,\text{des}}^\top, 0_{6\times 1}^\top, q_{\text{des}}^\top]^\top \in \mathbb{R}^{n+12}$ is the constant desired vector, and $q_{\text{des}} = [q_{1,\text{des}}, \ldots, q_{N,\text{des}}]^\top$ is appropriately chosen such that $x_O(t) = x_{O,\text{des}}, \forall t \in \{t \in \mathbb{R}_{\geq 0} : q(t) = q_{\text{des}}\}$ (see (7)). The error dynamics are then $\dot{e}(t) = f(x(t), u(t))$, which can be appropriately transformed to be written as:

$$\dot{e}(t) = f_e(e(t), u(t)), e(0) = e_0 = x(0) - x_{\text{des}}, \tag{13}$$

where $f_e(t) \triangleq f(e(t) + x_{\text{des}}, u(t))$. By ignoring over-actuated input terms, we have that $\tau_i = J_i^\top(q_i)u_i$, which becomes

$$\|\tau_i\| \leq \bar{\tau}_i \Leftrightarrow \sigma_{\min,i}\|u_i\| \leq \bar{\tau}_i, \tag{14}$$

where we have employed the property $\sigma_{\min}(J_i^\top)\|u_i\| \leq \|J_i^\top u_i\|$, with $\sigma_{\min}(J_i^\top)$ denoting the minimum singular value of $J_i^\top$, which is strictly positive, if the constraint $q_i \in \widetilde{\mathcal{Q}}_i$ is always satisfied. Hence, the constraint $|\tau_{i_k}| \leq \bar{\tau}_i$ is equivalent to $\|u_i\| \leq \frac{\bar{\tau}_i}{\sigma_{\min}(J_i^\top)}, \forall i \in \mathcal{N}$. Let us now define the following set $U \subseteq \mathbb{R}^{6N}$:

$$U = \{u \in \mathbb{R}^{6N} : \|u_i\| \leq \frac{\bar{\tau}_i}{\sigma_{\min}(J_i^\top)}, \forall i \in \mathcal{N}\}, \tag{15}$$

as the set that captures the control input constraints of the error dynamics system (13). Define also the set $X \subseteq \mathbb{R}^{n+12}$:

$$X = \Big\{x \in \mathbb{R}^{n+12} : \theta_O(t) \in [-\bar{\theta}, \bar{\theta}], \theta_{B_i}(t) \in [-\bar{\theta}, \bar{\theta}],$$
$$|\dot{q}_{k_i}| \leq \bar{\dot{q}}_i, q_i \in \widetilde{\mathcal{Q}}_i \backslash \left(\mathcal{S}_{i,O}(q) \cup \mathcal{S}_{i,A}(q)\right),$$
$$x_O \in \mathbb{R}^3 \backslash S_O(x_O), \forall t \in \mathbb{R}_{\geq 0}\Big\}.$$

The set X captures all the state constraint of the system dynamics (11). In order to translate the constraints that are dictated for the state x into constraints regarding the error state e from (12), define the set $E = X \oplus (-x_{\text{des}})$. Then, the following implication holds: $x \in \mathbb{R}^{n+12} \Rightarrow e \in E$.

The problem in hand is the design of a control input $u \in U$ such that $\lim_{t\to\infty} \|e(t)\| = 0$ while ensuring $e \in E$, $\forall t \in \mathbb{R}_{\geq 0}$. In order to solve the aforementioned problem, we propose a Nonlinear Model Predictive scheme, that is presented hereafter.

Consider a sequence of sampling times $\{t_i\}_{i\geq 0}$ with a constant sampling period $0 < h < T_p$, where T_p is the prediction horizon, such that $t_{i+1} = t_i + h, \forall\ i \geq 0$. In the sampled-data NMPC, a finite-horizon open-loop optimal control problem (OCP) is solved at discrete sampling time instants t_i based on the current state error information $e(t_i)$. The solution is an optimal control signal $\hat{u}(t)$, for $t \in [t_i, t_i + T_p]$. For more details, the reader is referred to [19]. The open-loop input signal applied in between the sampling instants is given by the solution of the following Optimal Control Problem (OCP):

$$\min_{\hat{u}(\cdot)} J(e(t_i), \hat{u}(\cdot))$$
$$= \min_{\hat{u}(\cdot)} \left\{ V(\hat{e}(t_i + T_p)) + \int_{t_i}^{t_i+T_p} \Big[F(\hat{e}(s), \hat{u}(s))\Big] ds \right\} \tag{16a}$$

subject to:

$$\dot{\hat{e}}(s) = f_e(\hat{e}(s), \hat{u}(s)), \hat{e}(t_i) = e(t_i), \tag{16b}$$
$$\hat{e}(s) \in E, \hat{u}(s) \in U, s \in [t_i, t_i + T_p], \tag{16c}$$
$$\hat{e}(t_i + T_p) \in \mathcal{E}_f, \tag{16d}$$

where the hat $\hat{(\cdot)}$ denotes the predicted variables (internal to the controller), i.e. $\hat{e}(\cdot)$ is the solution of (16b) driven by the control input $\hat{u}(\cdot) : [t_i, t_i + T_p] \to \mathcal{U}$ with initial condition $e(t_i)$. Note that the predicted values are not necessarily the same with the actual closed-loop values [19]. The term $F : E \times U \to \mathbb{R}_{\geq 0}$, is the *running cost*, and is chosen as $F(e(t), u(t)) = e(t)^\top Q e(t) + u(t)^\top R u(t)$. The terms $V : E \to \mathbb{R}_{>0}$ and $\mathcal{E}_f$ are the *terminal penalty cost* and *terminal set*, respectively, and are used to enforce the stability of the system. The terminal cost is given by $V(e(t)) = e(t)^\top P e(t)$. The terms $Q \in \mathbb{R}_{\geq 0}^{(n+12)\times(n+12)}$, $P \in \mathbb{R}_{>0}^{(n+12)\times(n+12)}$ and $R \in \mathbb{R}_{>0}^{6N\times 6N}$ are chosen as

$Q = \mathrm{diag}\{\widetilde{q}_1, \ldots, \widetilde{q}_{n+12}\}, P = \mathrm{diag}\{\widetilde{p}_1, \ldots, \widetilde{p}_{n+12}\}, R = \mathrm{diag}\{\widetilde{r}_1, \ldots, \widetilde{r}_{6N}\}$, where $\widetilde{q}_i \in \mathbb{R}_{\geq 0}, \widetilde{p}_i \in \mathbb{R}_{>0}, \forall i \in \{1, \ldots, n+12\}$ and $\widetilde{r}_j \in \mathbb{R}_{>0}, \forall j \in \{1, \ldots, 6N\}$ are constant weights. For the running cost, it holds that $F(0,0) = 0$, as well as: $m\|e\|^2 \leq m \left\| \begin{bmatrix} e \\ u \end{bmatrix} \right\|^2 \leq F(e,u) \leq M \left\| \begin{bmatrix} e \\ u \end{bmatrix} \right\|^2 \leq M\|e\|^2$, where $m = \min\{\widetilde{q}_1, \ldots, \widetilde{q}_{n+12}, \widetilde{r}_1, \ldots, \widetilde{r}_{6N}\}$ and $M = \max\{\widetilde{q}_1, \ldots, \widetilde{q}_{n+12}, \widetilde{r}_1, \ldots, \widetilde{r}_{6N}\}$.

The solution of the OCP (16a)-(16d) at time t_i provides an optimal control input denoted by $\hat{u}^\star(s; e(t_i))$, for $s \in [t_i, t_i + T_p]$. It defines the open-loop input that is applied to the system until the next sampling instant t_{i+1}: $u(t; e(t_i)) = \hat{u}^\star(t_i; e(t_i)), t \in [t_i, t_{i+1})$. The corresponding *optimal value function* is given by $J^\star(e(t_i)) \triangleq J^\star(e(t_i), \hat{u}^\star(\cdot; e(t_i)))$, where $J(\cdot)$ is given in (16a). The control input $u(t; e(t_i))$ is of the feedback form, since it is recalculated at each sampling instant using the new state information. The solution of (13) at time s, $s \in [t_i, t_i + T_p]$, starting at time t_i from an initial condition $e(t_i)$, applying a control input $u : [t_i, s] \to \mathcal{U}$, is denoted by $e(s; u(\cdot), e(t_i)), s \in [t_i, t_i + T_p]$. The predicted state of the system (13) at time s, $s \in [t_i, t_i + T_p]$ is denoted by $\hat{e}(s; u(\cdot), e(t_i))$, $s \in [t_i, t_i + T_p]$ and it is based on the measurement of the state $e(t_i)$ at time t_i, when a control input $u(\cdot; e(t_i))$ is applied to the system (13) for the time period $[t_i, s]$. Thus, it holds that $e(t_i) = \hat{e}(t_i; u(\cdot), e(t_i))$. We define an admissible control input as:

Definition 1. A control input $u : [t_i, t_i + T_p] \to \mathbb{R}^{6N}$ for a state $e(t_i)$ is called *admissible*, if all the following hold:

1) $u(\cdot)$ is piecewise continuous;
2) $u(s) \in U, \forall\ s \in [t_i, t_i + T_p]$;
3) $e(s; u(\cdot), e(t_i) \in E, \forall\ s \in [t_i, t_i + T_p]$;
4) $e(T_p; u(\cdot), e(t_i) \in \mathcal{E}_f$;

Lemma 1. *The terminal penalty function $V(\cdot)$ is Lipschitz in $\mathcal{E}_f$, with Lipschitz constant $L_V = 2\varepsilon_0 \sigma_{\max}(P)$, for all $e \in \mathcal{E}_f$.*

Proof. The proof can be found in [26, Appendix I, page 10]. ☐

The following theorem summarizes the results of this work.

Theorem 1. *Consider the Assumptions 1. Suppose also that:*

1) *The OCP* (16a)-(16d) *is feasible for the initial time* $t = 0$.
2) *The terminal set $\mathcal{E}_f \subseteq E$ is closed, with $0_{n+12} \in \mathcal{E}_f$.*
3) *The terminal set $\mathcal{E}_f$ is chosen such that there exists an admissible control input $u_f : [0, h] \to \mathcal{U}$ such that for all $e(s) \in \mathcal{E}_f$ it holds that:*
   a) $e(s) \in \mathcal{E}_f, \forall\ s \in [0, h]$.
   b) $\frac{\partial V}{\partial e} f_e(e(s), u_f(s)) + F(e(s), u_f(s)) \leq 0, \forall\ s \in [0, h]$.

Then, the closed loop system (13), *under the proposed control methodology, converges to the set $\mathcal{E}_f$ for $t \to \infty$.*

Proof. The proof consists of two parts: in the first part it is established that initial feasibility implies feasibility afterwards. Based on this result it is then shown that the error $e(t)$ converges to the terminal set $\mathcal{E}_f$. The proof can be found in [26, Section IV, page 6]. ☐

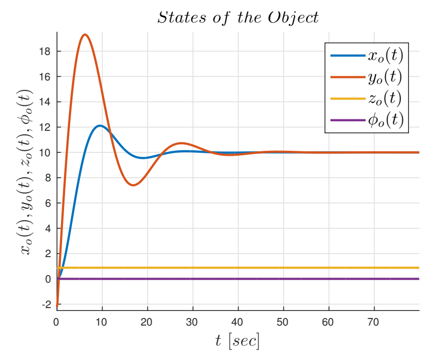

Fig. 2: The states of the object.

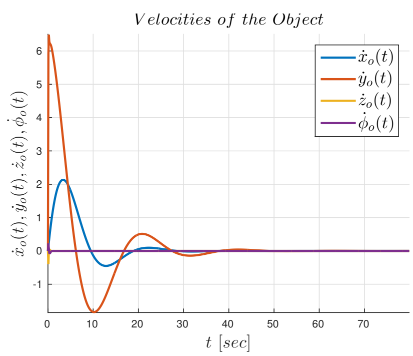

Fig. 3: The velocities of the object.

V. Simulation Results

To demonstrate the efficiency of the proposed control protocol, we consider a simulation example with $N = 2$ ground vehicles equipped with 2 DOF manipulators, rigidly grasping an object with $n_1 = n_2 = 4, n = n_1 + n_2 = 8$. From (11) we have that $x = [x_O^\top, v_O^\top, q^\top]^\top \in \mathbb{R}^{16}, u \in \mathbb{R}^8$, with $x_O = [p_O^\top, \phi_O]^\top \in \mathbb{R}^4$, $v_O = [\dot{p}_O^\top, \omega_O]^\top \in \mathbb{R}^4$, $p_O = [x_O, y_O, z_O]^\top \in \mathbb{R}^3, q = [q_1^\top, q_2^\top]^\top \in \mathbb{R}^8$, $q_i = [p_{B_i}^\top, \alpha_i^\top]^\top \in \mathbb{R}^4$, $p_{B_i} = [x_{B_i}, y_{B_i}]^\top \in \mathbb{R}^2$, $\alpha_i = [\alpha_{i_1}, \alpha_{i_2}]^\top \in \mathbb{R}^2, i \in \{1, 2\}$. The agents become singular when $\sin(\alpha_{i_1}) = 0, i \in \{1, 2\}$, thus the state constraints for the agents are set to $\epsilon \leq \alpha_{1_1} \leq \frac{\pi}{2} - \epsilon, -\frac{\pi}{2} + \epsilon \leq \alpha_{1_2} \leq \frac{\pi}{2} - \epsilon, -\frac{\pi}{2} + \epsilon \leq \alpha_{2_1} \leq -\epsilon, -\frac{\pi}{2} + \epsilon \leq \alpha_{2_2} \leq \frac{\pi}{2} - \epsilon$, with $\epsilon = 0.01$. We also consider the input constraints $-10 \leq u_i(t) \leq 10, i \in \{1, \ldots, 8\}$. The initial conditions are set to: $x_O(0) = [0, -2.2071, 0.9071, 0]^\top, v_O(0) = [0, 0, 0, 0]^\top, q_1(0) = [0, 0, \frac{\pi}{4}, \frac{\pi}{4}]^\top$ and $q_2(0) = [0, -4.4142, -\frac{\pi}{4}, -\frac{\pi}{4}]^\top$. The desired goal states are set to: $x_{O,\text{des}} = [10, 10, 0.9071, 0]^\top, v_O(0) = [0, 0, 0, 0]^\top, q_1(0) = [10, 12.2071, \frac{\pi}{4}, \frac{\pi}{4}]^\top$ and $q_2(0) = [10, 7.7929, -\frac{\pi}{4}, -\frac{\pi}{4}]^\top$. We also consider that there exists a spherical obstacle the obstacle at $[5, 5, 1]$ and radius 1m. We chose the sampling time as $h = 0.1\,\text{sec}$, the horizon as 3, and the total simulation time $80\,\text{sec}$. The simulation results are depicted

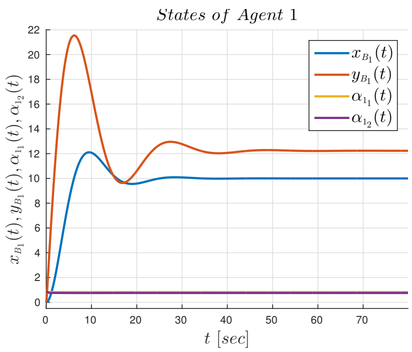

Fig. 4: The states of agent 1.

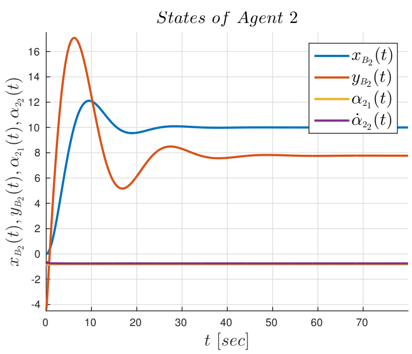

Fig. 5: The states of agent 2.

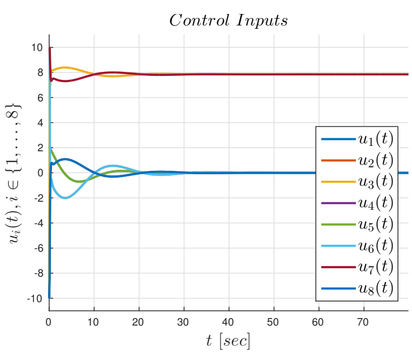

Fig. 6: The control inputs of the actuators.

in Fig. 2-6, which shows that the states of the agents and the object converge to the desired ones while guaranteeing that the obstacle is avoided and all state and input constraints are met. The simulations were carried out by using the NMPC toolbox given in [22] in MATLAB Environment.

VI. Conclusions and Future Work

In this work we proposed a NMPC scheme for the cooperative transportation of an object rigidly grasped by N robotic agents. The proposed control scheme deals with singularities of the agents, inter-agent collision avoidance as well as collision avoidance between the agents and the object with the workspace obstacles. Future efforts will be devoted towards including load sharing coefficients, internal force regulation, and complete decentralization of the proposed method. Finally, we will try to decrease the overall complexity and carry out real-time experiments.

References

[1] S. Schneider and R. Cannon, "Object impedance control for cooperative manipulation: Theory and experimental results," *TRA*, 1992.

[2] Y.-H. Liu and S. Arimoto, "Decentralized adaptive and nonadaptive position/force controllers for redundant manipulators in cooperations," *IJRR*, vol. 17, no. 3, pp. 232–247, 1998.

[3] M. Zribi and S. Ahmad, "Adaptive control for multiple cooperative robot arms," *CDC*, pp. 1392–1398, 1992.

[4] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal, "Decentralized cooperation between multiple manipulators," *IWRHC*, pp. 183–188, 1996.

[5] F. Caccavale, P. Chiacchio, and S. Chiaverini, "Task-space regulation of cooperative manipulators," *Automatica*, 2000.

[6] J. Gudiño-Lau, M. A. Arteaga, L. A. Munoz, and V. Parra-Vega, "On the control of cooperative robots without velocity measurements," *TCST*, vol. 12, no. 4, pp. 600–608, 2004.

[7] F. Caccavale, P. Chiacchio, A. Marino, and L. Villani, "Six-dof impedance control of dual-arm cooperative manipulators," *TMECH*, vol. 13, no. 5, pp. 576–586, 2008.

[8] D. Heck, D. Kostić, A. Denasi, and H. Nijmeijer, "Internal and external force-based impedance control for cooperative manipulation," *ECC*, pp. 2299–2304, 2013.

[9] S. Erhart and S. Hirche, "Adaptive force/velocity control for multi-robot cooperative manipulation under uncertain kinematic parameters," *IROS*, pp. 307–314, 2013.

[10] J. Szewczyk, F. Plumet, and P. Bidaud, "Planning and controlling cooperating robots through distributed impedance," *JRS*, 2002.

[11] A. Tsiamis, C. K. Verginis, C. P. Bechlioulis, and K. J. Kyriakopoulos, "Cooperative manipulation exploiting only implicit communication," *IROS*, pp. 864–869, 2015.

[12] F. Ficuciello, A. Romano, L. Villani, and B. Siciliano, "Cartesian impedance control of redundant manipulators for human-robot co-manipulation," *IROS*, pp. 2120–2125, 2014.

[13] A. Ponce-Hinestroza, J. Castro-Castro, H. Guerrero-Reyes, V. Parra-Vega, and E. Olguỳn-Dỳaz, "Cooperative redundant omnidirectional mobile manipulators: Model-free decentralized integral sliding modes and passive velocity fields," *ICRA*, pp. 2375–2380, 2016.

[14] W. Gueaieb, F. Karray, and S. Al-Sharhan, "A robust hybrid intelligent position/force control scheme for cooperative manipulators," *TMECH*, vol. 12, no. 2, pp. 109–125, 2007.

[15] M. Ciocarlie, F. M. Hicks, R. Holmberg, J. Hawke, M. Schlicht, J. Gee, S. Stanford, and R. Bahadur, "The velo gripper: A versatile single-actuator design for enveloping, parallel and fingertip grasps," *The International Journal of Robotics Research*, 2014.

[16] D. E. Koditschek and E. Rimon, "Robot navigation functions on manifolds with boundary," *Advances in applied mathematics*, 1990.

[17] B.Siciliano, L. Sciavicco, and L. Villani, *"Robotics: Modelling, Planning and Control"*. Springer, 2009.

[18] K. S. de Oliveira and M. Morari, "Contractive Model Predictive Control for Constrained Nonlinear Systems," *TAC*, 2000.

[19] 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.

[20] H. Chen and F. Allgöwer, "A Quasi-Infinite Horizon Nonlinear Model Predictive Control Scheme with Guaranteed Stability," *Automatica*, vol. 34, no. 10, pp. 1205–1217, 1998.

[21] R. Findeisen, L. Imsland, F. Allgöwer, and B. Foss, "Towards a Sampled-Data Theory for Nonlinear Model Predictive Control," *NDCA*, pp. 295–311, 2003.

[22] L. Grüne and J. Pannek, *Nonlinear Model Predictive Control*. Springer London, 2011.

[23] E. Camacho and C. Bordons, "Nonlinear Model Predictive Control: An Introductory Review," pp. 1–16, 2007.

[24] J. Frasch, A. Gray, M. Zanon, H. Ferreau, S. Sager, F. Borrelli, and M. Diehl, "An Auto-Generated Nonlinear MPC Algorithm for Real-Time Obstacle Cvoidance of Ground Vehicles," *ECC*, 2013.

[25] F. Fontes, "A General Framework to Design Stabilizing Nonlinear Model Predictive Controllers," *Systems and Control Letters*, 2001.

[26] A. Nikou, S. Heshmati, C. Verginis, and D. Dimarogonas, "A Nonlinear Model Predictive Control Scheme for Cooperative Manipulation with Singularity and Collision Avoidance," *https://arxiv.org/abs/1705.01426*.