Model-predictive control for non-prehensile tray-based object transportation

Mario Selvaggio and Fabio Ruggiero and Bruno Siciliano

***Abstract*— This document summarizes the latest results achieved in the field of robotic non-prehensile tray-based object transportation. The problem consists in transporting along a trajectory an object placed on a tray-like end-effector of a robotic manipulator preventing it to slide and potentially fall. We developed a model-predictive control approach that computes optimal jerk-based inputs for the considered system to accomplish the task while enforcing both system and non-sliding manipulation constraints. Experiments performed on the RoDyMan humanoid robot validated our approach.**

I. Introduction

The nonprehensile transportation of an object along a desired trajectory by means of a tray-like robot end-effector is a longstanding problem in service robotics. Several authors have proposed different modeling, planning, and control strategies to accomplish the transporting task while enforcing non-sliding manipulation constraints [1]–[3].

Recently, with the increase of computing power endowed into robots, numerical optimal control techniques started to be devised. Among these, the Model-Predictive Control (MPC) aims to compute solutions for the control input that are optimal along the future predicted states of the system. In [4], we embraced this approach and developed a MPC that uses the combined manipulator/object dynamics for prediction and enforces the related non-sliding manipulation constraints. The controller features online contact forces calculation, and outputs optimal jerk-based control solutions for the robotic system to perform the considered task [5]. In this paper, we summarize the most relevant results achieved by the proposed controller evaluated on our RoDyMan humanoid robot.

II. Methodology

A. System Modeling

In absence of external interactions, the combined manipulator/object dynamics can be written in compact form as follows

$$\tilde{M}(q)\,\ddot{q} + \tilde{C}(q,\dot{q})\,\dot{q} + \tilde{n}(q,\dot{q}) = \tau, \tag{1}$$

The research leading to these results has been supported by the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017008 (Harmony).

The authors are with the Department of Electrical Engineering and Information Technology, University of Naples Federico II, name.surname@unina.it

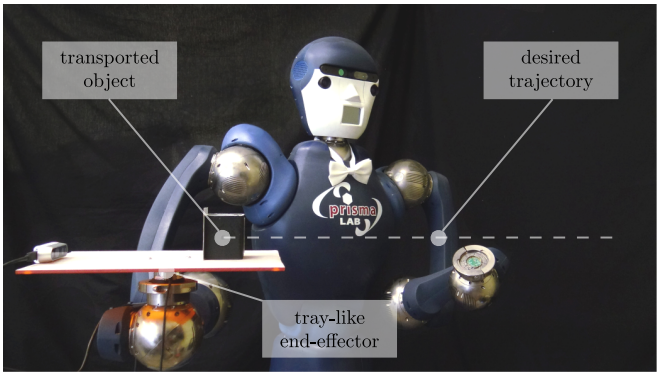


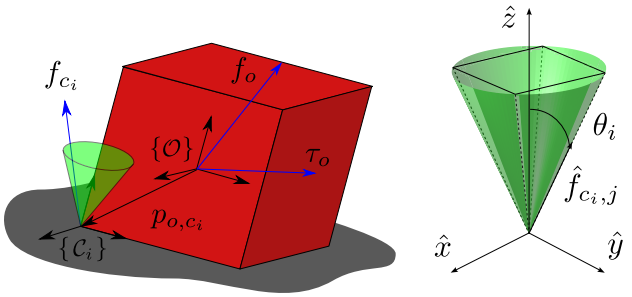
Fig. 1. Illustration of the main problem addressed in this paper: a robotic manipulator (blue) has to transport an object (black cube) along a desired trajectory (dashed grey) on a tray-like end-effector (orange/white) while guaranteeing a non-sliding behaviour.

with (dropping dependencies)

$$\begin{aligned}\tilde{M} &= M_m + J_o^{\mathrm{T}} M_o J_o,\\ \tilde{C} &= C_m + J_o^{\mathrm{T}}\left(C_o J_o + M_o \dot{J}_o\right),\\ \tilde{n} &= n_m + J_o^{\mathrm{T}} n_o.\end{aligned} \tag{2}$$

where $q,\ \dot{q} \in \mathbb{R}^n$ represent the state of the robotic system in generalised coordinates, $J_o(q) \in \mathbb{R}^{6\times n}$ represents the Jacobian matrix relating $\dot{q}$ to the object linear/angular velocities, $M_m(q) \in \mathbb{R}^{n\times n}$ is the symmetric positive-definite manipulator joint-space inertia matrix, $C_m(q,\dot{q}) \in \mathbb{R}^{n\times n}$ is the manipulator matrix of centrifugal/Coriolis terms, $n_m(q) \in \mathbb{R}^n$ is the manipulator gravity vector, $\tau \in \mathbb{R}^n$ is the vector of manipulator joint torques (representing the overall control input of the robotic system), $M_o(q) \in \mathbb{R}^{6\times 6}$ is the object positive-definite mass/inertia matrix, $C_o(q,\dot{q}) \in \mathbb{R}^{6\times 6}$ is the object matrix of centrifugal/Coriolis terms, $n_o(q) \in \mathbb{R}^6$ is the object gravity vector. The model in (1) is derived assuming that the object does exhibit sliding with respect to the tray. To satisfy this assumption the contact forces between the object and the tray must be confined within the friction cone space. This constraint is enforced by control considering the contact surface of the object (assumed a cuboid) discretized with $n_c = 4$ contact points located in the object vertices in contact with the tray and approximating friction cones with polyhedrals (see Fig. 2). This allows writing the stacked vector of contact forces $F_c = [f_{c_1}^{\mathrm{T}}, \ldots, f_{c_{n_c}}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{3n_c}$ as a linear combination of the friction cone edges, i.e.

$$F_c = \hat{F}_c \Lambda, \quad \Lambda = \left(\lambda_{c_1,1}, \ldots, \lambda_{c_{n_c},k}\right) \in \mathbb{R}^{kn_c}, \tag{3}$$



(a) (b)

Fig. 2. (a) Drawing of the object body wrench $\mathcal{F}_o = (f_o, \tau_o)$, $i-$th contact force f_{c_i} and friction cone (shaded green). p_{o,c_i} is the vector defining the $i-$th contact position in the body frame $\{\mathcal{O}\}$. (b) Approximation of circular friction cones (in shaded green) with polyhedral cones (black lines) identified by unit vectors $\hat{f}_{c_i,j}$.

where

$$\hat{F}_c = \text{diag}\left(\hat{F}_{c,1}, \dots, \hat{F}_{c,n_c}\right), \quad \hat{F}_{c,i} = \left[\hat{f}_{c_i,1}, \dots, \hat{f}_{c_i,k}\right] \tag{4}$$

is the matrix whose columns denote the friction cone edges unit vectors, while Λ is the vector of contact force parameters. At this point, for F_c to belong to the derived approximated friction cone space, it is sufficient to enforce

$$\Lambda_i \geq 0, \qquad \forall\ i = 1, \dots, kn_c, \tag{5}$$

which constitutes the non-sliding manipulation constraint.

B. Model-predictive Control

MPC is realized solving the following Optimal Control Problem (OCP) at each time step

$$\begin{aligned}
\min_{u(\cdot)} \quad & \Phi(x(T)) + \int_0^T L(x(t),\, u(t),\, t)\ \mathrm{d}t && \text{(6a)}\\
\text{s.t.} \quad & x(0) = x_0 && \text{(6b)}\\
& \dot{x} = f(x(t), u(t)) && \text{(6c)}\\
& \underline{x} \leq x(t) \leq \bar{x} && \text{(6d)}\\
& \underline{u} \leq u(t) \leq \bar{u} && \text{(6e)}
\end{aligned}$$

where $\Phi(x(T)) = \|x^*(T) - x(T)\|^2_{Q_e}$ and $L(x(t),\, u(t),\, t) = \|x^*(t) - x(t)\|^2_Q + \|u(t)\|^2_R$, T is the prediction horizon, Q_e, Q and R are diagonal positive semi-definite weight matrices, and x^* is the reference state. We defined an *extended system state* which includes, besides the proper manipulator state q and $\dot{q}$, the contact force coefficients Λ (defined in (3)) and the control torques τ. Their time derivative $\dot{\tau}$, which is related to the system *jerk* and affects the variation of the contact forces, is directly specified by the control signal u computed solving (6). With this choice, the continuous time dynamic evolution of the system state writes as follows

$$\dot{x} = f\left(x, u\right) = \begin{cases} \dot{\tau} = u \\ \dot{q} = \dot{q} \\ \ddot{q} = \tilde{M}^{-1}(\tau - \tilde{C}\dot{q} - \tilde{n}) \\ \dot{\Lambda} = \left(G\hat{F}_c\right)^{\dagger}\left(A\dot{\tau} + B\tau + C\right) \end{cases}, \tag{7}$$

where

$$A = M_o J \tilde{M}^{-1}, \quad B = M_o \dot{J} \tilde{M}^{-1}, \quad C = -2M_o \dot{J} \tilde{M}^{-1} \tilde{n}. \tag{8}$$

To derive the last equation in (7) we adopt for Λ the minimum two-norm solution of $\mathcal{F}_o = G\hat{F}_c\Lambda$, where $\mathcal{F}_o = (f_o, \tau_o)$ represents the object body wrench and $G \in \mathbb{R}^{6\times 3n_c}$, usually referred to as *grasp matrix*, maps F_c to $\mathcal{F}_o$. This is realized by the Moore–Penrose inverse operator applied to the matrix $G\hat{F}_c$. Moreover, we assume that all the matrices entering the dynamic model hold constant over the time horizon, i.e. their time derivative is null.

It is worth mentioning that our ultimate goal is to transport the object to the target pose following a desired trajectory (see Fig. 1), i.e., we aim to realize $x_o = x_o^*(t)$ and $\dot{x}_o = \dot{x}_o^*(t)$, where $x_o^*(t)$, $\dot{x}_o^*(t)$ are the desired object states (parametrized pose and its time derivative), while satisfying both non-sliding manipulation and robotic system constraints. From this, we can calculate the reference values for the extended state x^*, in particular q_o^*, $\dot{q}_o^*$ using an a standard inverse kinematics routine, under the assumption that the object is rigidly attached to the manipulator.

As for the constraints, lower/upper bounds on the system (joint positions, joint velocities, joint torques) and manipulation states (contact force coefficients), are included in (6d) via $\underline{x}, \bar{x}$, respectively, while those on input are included in (6e) via $\underline{u}, \bar{u}$, respectively. The feedback term denoted by (6b) is directly retrieved from the robotic system. Note that while the measure of q, $\dot{q}$, τ is readily available in torque-controlled manipulators, the measure of Λ (or F_c) must be retrieved indirectly from the measure of $\mathcal{F}_o$ (which can be instead conveniently measured through a F/T sensor installed at the end-effector of the manipulator).

The solution of the problem in (6) (i.e., the time derivative of the joint torques) is used to obtain the required torque at each time instant t according to the following integration rule

$$\tau(t) = \int_0^t \dot{\tau}(t)\ \mathrm{d}t. \tag{9}$$

The problem in (6), together with the dynamics (7), and the integration rule (9) are discretized with time step equal to the control loop cycle time. The controller has been realized using the `acados` which allowed to conveniently formulating the OCP in `MATLAB` and later generate the C/C++ library for the real world implementation. For details about the controller/solver settings the reader can refer to [4].

The devised controller with an NMPC horizon length $N = 10$ takes 5.4 ms on an average (1.3 ms standard deviation) to solve one step of the problem on a Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz.

A simulation environment that uses a KUKA LWR-IIWA manipulator can be downloaded from the following link: https://github.com/prisma-lab/nonprehensile-object-transp, while the video of both simulation and real-world experiments is at this link: https://youtu.be/H14NDnmpcNg.

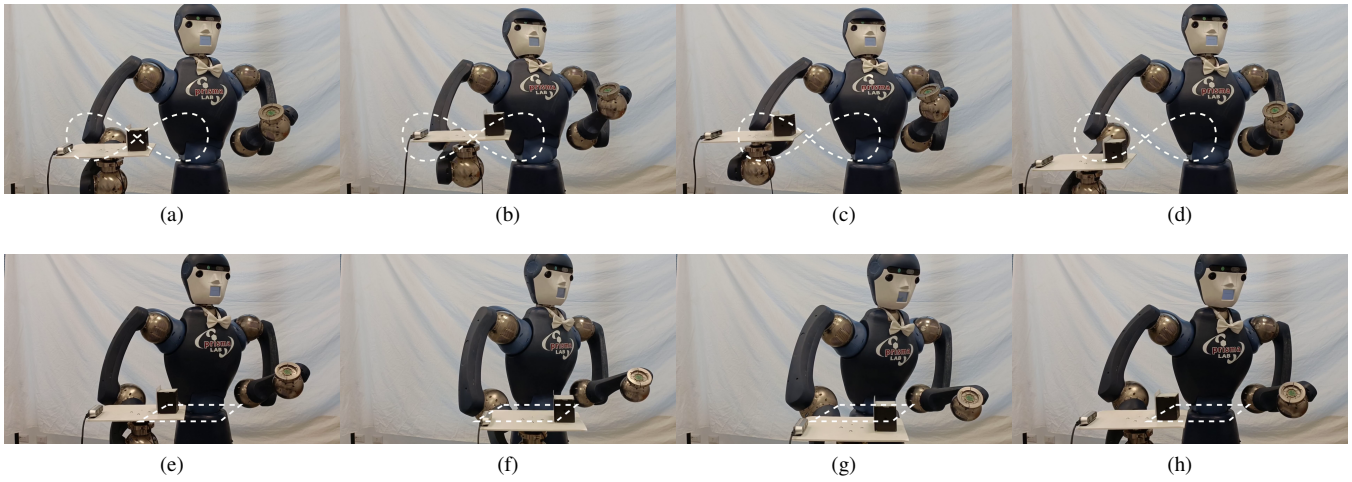


Fig. 3. Experiments using the proposed model-predictive non-sliding manipulation control approach. The desired trajectory is shown as a white dashed line. (a) – (d) Key frames of RoDyMan robot tracking a 5.5 s Lemniscate-like trajectory. (e) – (h) Key frames of RoDyMan robot tracking a 4.5 s rectangular trajectory featuring three via points.

III. Results

To demonstrate the validity of our approach, we conducted real experiments employing the RoDyMan humanoid robot. It is a 21-DoF robot made of a custom-built mobile base, a two-DoFs torso, two one-Dof shoulders, and two six-DoFs Shunk Powerball arms. Additional construction details can be found in [6]. For our experiments, we employed only the kinematic chain starting at the torso and ending at the tip of the robot's right arm (9 DoFs). A plastic tray-like end-effector was attached to it through a 3D printed support, which embedded a Shunk 6-Axis F/T sensor. A calibrated Intel RealSense Depth Camera D415 was mounted on the tray with the purpose of tracking and recording the object displacement thanks to a QR-code pattern and the VISP auto tracker module. The object is a steel hollowed cuboid of dimensions $60 \times 60 \times 70$ mm, whose inertial properties are: mass $m_o = 0.236$ kg and diagonal inertial matrix, $I_o = \text{diag}(4.5375 \times 10^{-5})$ kgm^2. The friction coefficient between the object and the tray has been experimentally identified in $\mu = 0.2$. The robot was position-controlled, and its set point was extracted from the output trajectory solution of the MPNSM controller given in (6). The robot control cycle time is set to 8 ms. Tables that contains the real system control parameters and the robot physical limits are given in [4]. A picture of the experimental setup is given in Fig. 1.

To prove the robustness of our controller we consider two trajectories: (i) a rectangular path in the horizontal plane, shown at the top of Fig. 4, featuring three via points obtained imposing trapezoidal velocity profiles with acceleration time equal to 0.2 seconds at the transitions between segments; (ii) a Lemniscate-like path in the vertical plane, shown at the top of Fig. 5, obtained employing a piecewise cubic B-spline curve enclosed by its control points. The validation of the performance using the proposed MPNSM control onto the Rodyman robot is shown in the graphs of Fig. 4 and Fig. 5, respectively. A timed sequence of key frames taken during the performed experiments is shown in Fig. 3 (a) – (d), (e) – (h), where the desired trajectory is shown in overlay. As a measure of the tracking performance we introduce the error term $\mathcal{E}(t) = (e_p(t), e_o(t))$, with $e_p(t) = ||p_o^*(t) - p_o(t)||$ and $e_o(t) = ||\phi_e(t)||$, where ϕ_e is the vector of Euler Angles extracted from the rotation matrix error, i.e. $R_o^{*\mathrm{T}} R_o$, where R_o^* and R_o are is the desired and the current object rotation matrix, respectively. In both cases, it can be noted that when the robotic system constraints become binding ($\dot{q}$ - bottom graph), the tracking performance is penalised ($\mathcal{E}$ - top graph), while the contact force coefficients (Λ - middle graph) are still kept greater than zero.

IV. Conclusions

This paper summarized results achieved by the model-predictive non-sliding manipulation control approach for non-prehensile object transportation developed in [4]. We reported the combined manipulator/object dynamic model and the associated non-sliding constraints that are enforced by the controller. The proposed optimization-based controller has been capable of safely accomplishing trajectory tracking tasks with an object being transported in a non-prehensile way on a tray-like manipulator end-effector. The controller imposes that the manipulation and physical constraints of the robotic system are always respected during the executed trajectory at the expense of the tracking performances. An interesting future research direction on this topic is the inclusion of the variable tray orientation technique introduced in [7].

References

[1] J. Z. Woodruff and K. M. Lynch, "Planning and control for dynamic, nonprehensile, and hybrid manipulation tasks," in *Proc. IEEE Int. Conf. Robot. Autom.*, pp. 4066–4073, May 2017.

[2] G. Martucci, J. Bimbo, D. Prattichizzo, and M. Malvezzi, "Maintaining stable grasps during highly dynamic robot trajectories," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, pp. 9198–9204, 2020.

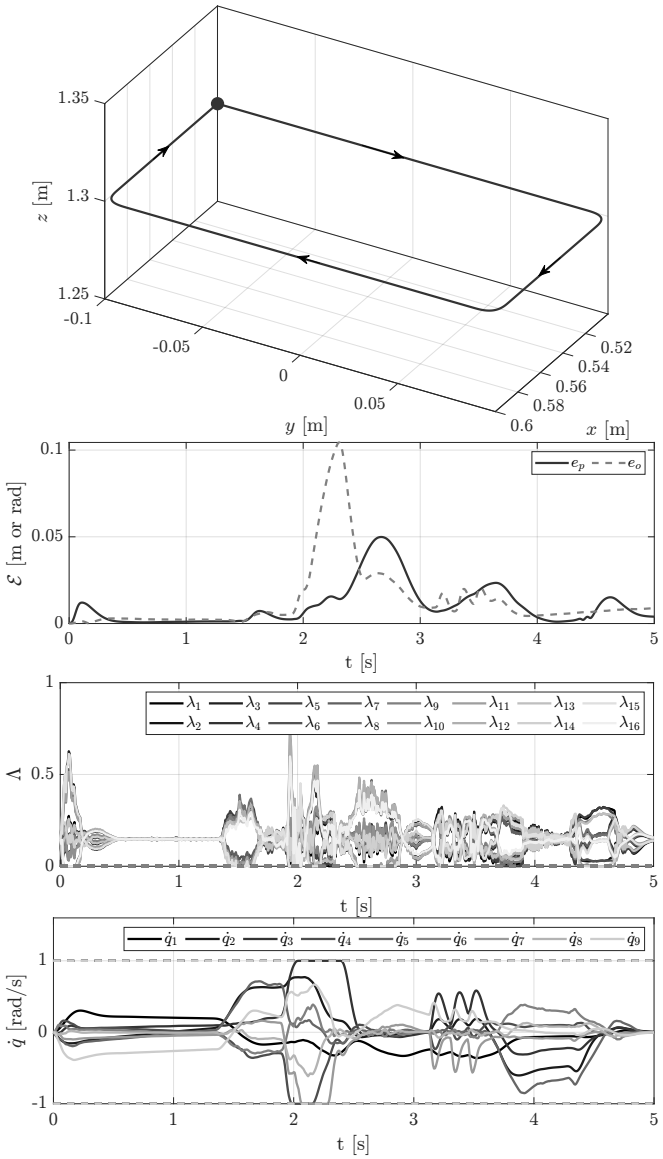


Fig. 4. Validation of the tracking performance along the rectangular, 4.5 s duration trajectory. The black dot denotes the start/end point, the arrows indicate the direction. The norm of the error terms (e_p and e_o – top graph) is higher as manipulation (Λ – middle graph) and system velocity constraints ($\dot{q}$ – bottom graph) are met.

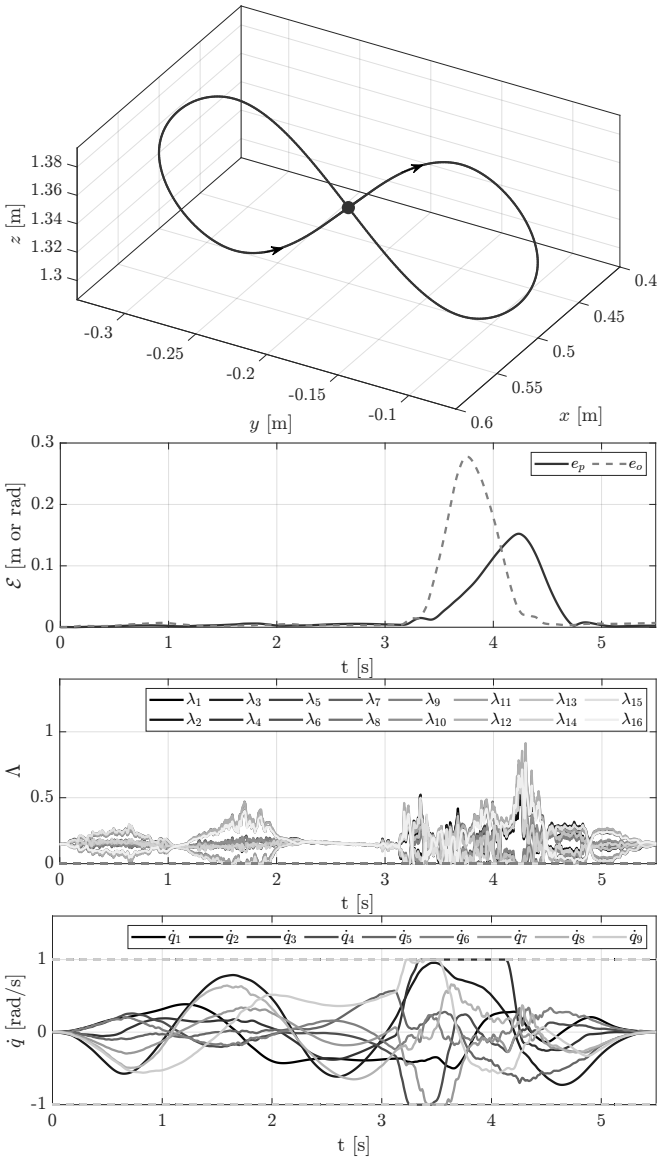


Fig. 5. Validation of the tracking performance along the Lemniscate-like, 5.5 s duration trajectory. The black dot denotes the start/end point, the arrows indicate the direction. The norm of the error terms (e_p and e_o – top graph) is higher as manipulation (Λ – middle graph) and system velocity constraints ($\dot{q}$ – bottom graph) are met.

[3] M. Selvaggio, J. Cacace, C. Pacchierotti, F. Ruggiero, and P. Robuffo Giordano, "A shared-control teleoperation architecture for nonprehensile object transportation," *IEEE Trans. Robot.*, vol. 38, no. 1, pp. 569–583, 2022.

[4] M. Selvaggio, A. Garg, F. Ruggiero, G. Oriolo, and B. Siciliano, "Non-prehensile object transportation via model predictive non-sliding manipulation control," *IEEE Trans. Control Syst. Technol.*, vol. 31, no. 5, pp. 2231–2244, 2023.

[5] A. Gazar, G. Nava, F. J. A. Chavez, and D. Pucci, "Jerk control of floating base systems with contact-stable parameterized force feedback," *IEEE Trans. Rob.*, vol. 37, no. 1, pp. 1–15, 2021.

[6] F. Ruggiero, A. Petit, D. Serra, A. C. Satici, J. Cacace, A. Donaire, F. Ficuciello, L. R. Buonocore, G. A. Fontanelli, V. Lippiello, L. Villani, and B. Siciliano, "Nonprehensile manipulation of deformable objects: Achievements and perspectives from the robotic dynamic manipulation project," *IEEE Robot. Autom. Mag.*, vol. 25, no. 3, pp. 83–92, 2018.

[7] R. Subburaman, M. Selvaggio, and F. Ruggiero, "A non-prehensile object transportation framework with adaptive tilting based on quadratic programming," *IEEE Robot. Autom. Lett.*, vol. 8, no. 6, pp. 3581–3588, 2023.