

Optimal control goal manifolds for planar nonprehensile throwing

Alexander Pekarovski and Martin Buss

Abstract—This paper presents a throwing motion planner based on a goal manifold for two-point boundary value problem. The article outlines algorithmic and geometric issues for planar throwing of rigid objects with a nonprehensile end-effector. Special attention is paid to the challenge of controlling a desired 6-dimensional state of the object with a planar 3-DoF robot. Modeling of the contacts is discussed using a state vector of the coupled robot and object dynamics. Robustness against uncertainty due to varying model parameters such as object inertia and friction between the end-effector and the object is investigated. An approach for obtaining manifolds of terminal constraints from the goal configuration is described. Classification of these constraints is given. Finally, feasible trajectory generation conditions for successful execution of the generated optimal controls are discussed.

I. INTRODUCTION

Intensive development of space robotics, industrial applications and robotic sports require simple and cost-effective robots capable of dealing with highly dynamic and precise tasks. In areas such as part feeding [1] and transportation within logistic chains [2] one of the main challenges is to position and orient the manipulated object. An effective way is to throw the object from its initial state to the final state using a simple robot with a generic end-effector. Usually such a generic end-effector is designed as a flat plate, so the manipulation is nonprehensile.

Conventional methods for object manipulation in robotics are based on grasping the object, moving it in a workspace and finally releasing it at a desired position. To overcome shortcomings of the pick and place paradigm, dynamic manipulation was introduced. Mason and Lynch [3] defined dynamic manipulation as an operation that uses not only kinematics, static and quasi-static forces but also forces of acceleration. They discussed different manipulation tasks with control of object unactuated degrees of freedom (DoFs) using dynamic coupling. As an example, throwing of a club, including carry and release conditions, was investigated. It was shown that a larger number of DoFs of the object can be reached using simple 1-DoF robots by exploiting centrifugal and Coriolis forces and by allowing the object to slip and roll [4]. This strategy of using all possible capabilities of the simple robot mechanics was continued in a number of works.

Frank *et al.* [2] suggested to use throwing for industrial applications to speed up transportation of parts in flexible manufacturing systems. They discussed various simple throwing and shooting mechanisms and their applications in

logistic chains. A 1-DoF rotational robot capable of throwing a ball up to 8 m with high precision was discussed in [5]. A throwing model with simple kinematics including an air drag influence was presented in [6]. Miyashita *et al.* [7], [8] suggested a control strategy for a 1-DoF rotational robot with edges based on iteration optimization learning. For this purpose a nonlinear optimization problem is solved by the sequential quadratic programming (SQP) method for different initial conditions with constraints on actuator limits and final object position.

Tabata *et al.* [9] presented a passing manipulation. Their simple system consists of one throwing and one catching 1-DoF rotational robots that transfer a ball from one to another. A solution for motion planning was found using Newton's method with top height of the parabolic orbit, torque of the actuators and total passing time as performance indices. Mori *et al.* [10] showed that using a finger-link contact model makes it possible to implement pitching motion with prescribed velocity, angular velocity and direction of a circular object using only a 1-DoF robot. Global search and simulated annealing methods were used with criterion function as a weighted error between kinematic variables at the time of release and desired variables. However, the extent to which the three kinematic variables could be independently controlled is limited.

The contribution of this paper is a novel motion planning algorithm for throwing based on the manifold of possible release states leading to a desired object goal state. The goal manifold is obtained using partial knowledge of the desired state of the object. An optimal control method is designed as a two-point boundary value problem (2PBVP). Specific conditions and challenges of nonprehensile throwing tasks are discussed.

The manipulation tasks require fast and accurate motions. Therefore the main challenges lie in the field of optimality, robustness and feasibility of generated trajectories. We intend to broaden the capability of reaching the desired object goal state by increasing the number of actuators to three and improving optimal control for motion planning by introduction of corresponding constraints and objectives. The number of controlled DoFs is dictated by the minimal requirements on the number of constraints imposed on the goal data for throwing. The goal is to access the whole 6-dimensional (6-D) state of the object and increase the reachability of the system.

The throwing tasks are investigated in a two-dimensional workspace. For this purpose the experimental setup consisting of a manipulator with three rotational joints (3R) and an inclined air-table was designed. The air-table compensates the gravity component that acts perpendicular to the surface

A. Pekarovski and M. Buss are with the Institute of Automatic Control Engineering, Technische Universität München, Munich, Germany, 80290 {a.pekarovski, mb}@tum.de

with an air-flow and provides the environment for investigation of planar throwing with decreased gravity, see Fig. 1.

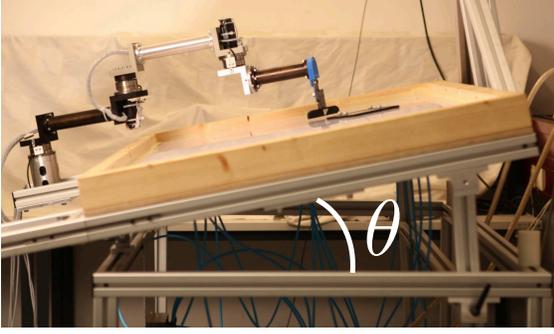


Fig. 1: Side-view of planar robotic arm mounted on the inclined air-table.

The dependency of the inclination of the table after gravity modification is

$$g_m = g \sin \theta, \quad (1)$$

where θ is the angle of inclination of the robot and the table from its horizontal position.

Sec. II formulates a problem and provides theoretical background for task definition. In Sec. III the approach for optimal throwing of the rigid object with the nonprehensile end-effector in presence of Coulomb friction is presented. The terminal constraint manifold leading to the desired end state of the object is thoroughly researched. Based on discussed assumptions and terminal constraint manifold analysis, the optimal control method is presented. Corresponding simulation and numerical results of optimization are shown. Experimental results and hardware description are to be found in Sec. V. Finally, Sec. VI contains conclusions on conducted work and future research directions.

II. PROBLEM FORMULATION

Given the 3-DoF planar robot with flat end-effector, see Fig. 2, the robot configuration is defined as

$$\mathbf{q} = [q_1 \quad q_2 \quad q_3]^T. \quad (2)$$

A simple algorithm for task execution is as follows: go to initial position and receive the object, carry it to the release point, and finally release the object. Without loss of generality, the initial configuration of the robot is taken at the initial time $t_0 = 0$, with initial position $\mathbf{q}(0) = \mathbf{q}_0$ and initial velocity $\dot{\mathbf{q}}(0) = \mathbf{0}$. The goal is to achieve a desired object velocity at a desired goal position and orientation. The object configuration

$$\mathbf{p} = [x \quad y \quad \phi]^T \quad (3)$$

is shown in Fig. 2 and Fig. 3. The robot dynamics in joint space are described as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} = \boldsymbol{\tau} - \boldsymbol{\tau}', \quad (4)$$

$$\boldsymbol{\tau}' = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) \sin \theta + \mathbf{F}(\dot{\mathbf{q}}), \quad (5)$$

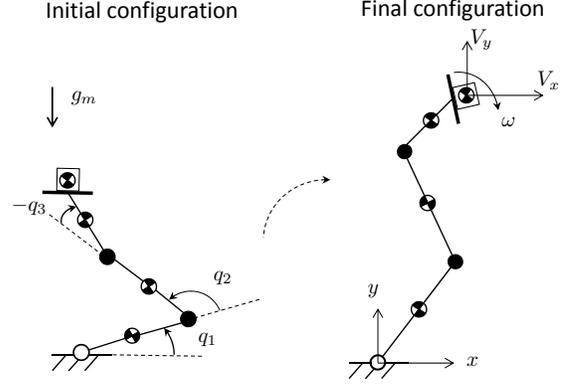


Fig. 2: Initial and final (release) postures of the robot.

where \mathbf{q} , $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ are the generalized position, velocity and acceleration vectors, $\boldsymbol{\tau}$ is the torque produced by the motors, $\mathbf{M}(\mathbf{q})$ is the inertia matrix of the manipulator, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is a matrix of Coriolis and centrifugal forces, $\mathbf{G}(\mathbf{q})$ is a matrix of gravity forces and $\mathbf{F}(\dot{\mathbf{q}})$ is a matrix of friction forces. Thus, the generalized accelerations of the joints are calculated as

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}(\mathbf{q})(\boldsymbol{\tau} - \boldsymbol{\tau}'). \quad (6)$$

State equations are defined

$$\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^{-1}(\mathbf{x}) \end{bmatrix} \mathbf{u} - \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1}(\mathbf{x})\boldsymbol{\tau}'(\mathbf{x}) \end{bmatrix} \quad (7)$$

$$\boldsymbol{\tau}'(\mathbf{x}) = \mathbf{C}(\mathbf{x}) + \mathbf{G}(\mathbf{x}) + \mathbf{F}(\mathbf{x}),$$

with the state variable $\mathbf{x} = (\mathbf{q}, \dot{\mathbf{q}})^T$ and the torque \mathbf{u} produced by the motors. The motion planner has to execute nonprehensile throwing task and generate optimal trajectories based on the choice of the goal state of the object. Thus, 3R robot is controlled in such a way that the model uncertainty influence is minimized.

III. MODELING AND CONTROL OF NONPREHENSILE THROW

A. Contact modeling

It is assumed that an object has a line contact with the end-effector. We consider a rectangular object which is displaced from the center of the end-effector, as depicted in Fig. 3. The goal is to bring the object to the release point with desired position, orientation and velocities. This task can be fulfilled with the help of dynamic grasp [3]. Due to the dynamic grasp property the object remains attached to one fixed position on the end-effector under the influence of the contact friction. This constraint is investigated for the case of a 3R robot. The contact conditions for robust carrying and release of the object are analyzed.

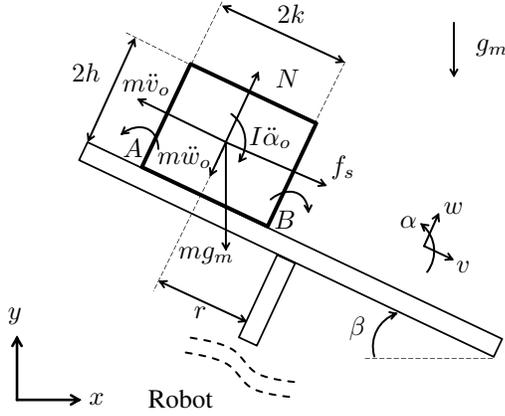


Fig. 3: Flat end-effector with rectangular object having a line contact.

1) *State space representation:* The state space equations for the object in the frame of reference related to the robot's base are

$$\dot{\mathbf{x}}_{o,gr} = \begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{p}} \end{bmatrix} = \begin{bmatrix} -\dot{v}_o \cos \beta - \dot{w}_o \sin \beta \\ -\dot{w}_o \cos \beta + \dot{v}_o \sin \beta \\ \dot{\alpha}_o \\ -\ddot{v}_o \cos \beta - \ddot{w}_o \sin \beta \\ -\ddot{w}_o \cos \beta + \ddot{v}_o \sin \beta \\ \ddot{\alpha}_o \end{bmatrix} \quad (8)$$

knowing that

$$\begin{bmatrix} \dot{v}_o & \dot{w}_o & \dot{\alpha}_o \end{bmatrix}^T = \mathbf{J}(\mathbf{q}, r) \dot{\mathbf{q}} \quad (9)$$

$$\begin{bmatrix} \ddot{v}_o & \ddot{w}_o & \ddot{\alpha}_o \end{bmatrix}^T = \mathbf{J}(\mathbf{q}, r) \ddot{\mathbf{q}} + \dot{\mathbf{J}}(\mathbf{q}, r) \dot{\mathbf{q}},$$

where \mathbf{J} is a Jacobian from the base of the robot to the center of the line contact on the end-effector, r is the distance from the center of the end-effector to the center of the contact. The force equilibrium is as follows

$$\begin{aligned} mg_m \sin \beta - m\ddot{v}_o + f_s &= 0 \\ N - m\ddot{w}_o - mg_m \cos \beta &= 0 \\ |f_s| &\leq \mu_s N. \end{aligned} \quad (10)$$

So, the constraint to prevent slipping is

$$J_{sl} = |m\ddot{v}_o - mg_m \sin \beta| - m\mu_s (g_m \cos \beta + \ddot{w}_o) \leq 0. \quad (11)$$

In order to increase margin for anti-slipping property, one has to minimize J_{sl} . In case of uncertain μ_s the cost function transforms into

$$J_{\mu_s,sl} = \frac{|\ddot{v}_o - g_m \sin \beta|}{g_m \cos \beta + \ddot{w}_o}. \quad (12)$$

If both model parameters μ_s and g_m are uncertain, separate variables $|\ddot{v}_o|$, $|\beta|$ and $-\ddot{w}_o$ are minimized with corresponding weighting coefficients. Consequently, an increase of μ_s will have a positive effect on the robustness.

The constraint for preventing rolling has the following form

$$\begin{aligned} -I_z \ddot{\alpha}_o + m(h\ddot{v}_o - hg_m \sin \beta - k\ddot{w}_o - kg_m \cos \beta) &\leq 0 \\ I_z \ddot{\alpha}_o - m(h\ddot{v}_o + hg_m \sin \beta - k\ddot{w}_o - kg_m \cos \beta) &\leq 0, \end{aligned} \quad (13)$$

where I_z is the inertia of the object around the vertex contact point. After simplification (13) becomes

$$\begin{aligned} J_{rol} &= |-I_z \ddot{\alpha}_o - hm\ddot{v}_o + hm g_m \sin \beta| \\ &\quad - km g_m \cos \beta - km\ddot{w}_o \leq 0. \end{aligned} \quad (14)$$

The minimization of J_{rol} leads to an increase of the margin of the anti-rolling property.

As for the uncertainty of multiple model parameters, e.g. I_z , k , h , m and/or g_m , the minimization of the separate variables $|\ddot{\alpha}_o|$, $|\beta|$, $-\ddot{w}_o$ and $|\ddot{v}_o|$ with corresponding weighting coefficients is being achieved. If condition (14) is not fulfilled, the object will start rolling whether around vertex A or B . In this case the positive design properties are obtained by maximizing length of the object.

Using the introduced object state representation, slipping and rolling dynamics could be also allowed. For example, the object dynamics for slipping will take the form

$$\dot{\mathbf{x}}_{o,sl} = \begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{p}} \end{bmatrix} = \begin{bmatrix} -\dot{v}_o \cos \beta - \dot{w}_o \sin \beta \\ -\dot{w}_o \cos \beta + \dot{v}_o \sin \beta \\ \dot{\alpha}_o \\ C(-\ddot{v}_o \cos \beta - \ddot{w}_o \sin \beta) \\ C(-\ddot{w}_o \cos \beta + \ddot{v}_o \sin \beta) \\ \ddot{\alpha}_o \end{bmatrix}, \quad (15)$$

where $C = \frac{1 + \text{sgn}(|f_s| - \mu_s N)}{2}$ is a slipping condition. In case of slipping, $\mu_k N$ is used instead of f_s for calculation of \ddot{v}_o , where μ_k is a kinetic friction coefficient. However, additional constraints on the maximum distance from the center of the end-effector are imposed

$$\begin{aligned} r(t) &= r_0 + C \int_0^{t_r} \int_0^{t_r} \ddot{v}_o dt \\ r_{\min} &\leq r(t) \leq r_{\max}, \end{aligned} \quad (16)$$

where t_r is a release time.

The object state variables are appended to robot state variables. Hence, the full state vector will consist of twelve state variables

$$\mathbf{x}_{\text{ext}} = \begin{bmatrix} \mathbf{x} \\ \mathbf{x}_o \end{bmatrix}. \quad (17)$$

With the help of such a representation it is possible to completely describe the robot and the object coupled dynamics.

2) *Release conditions:* For releasing the object the following conditions are to be fulfilled

$$\begin{aligned} \ddot{w}_o(t_r) + g_m \cos \beta &\leq 0 \\ \ddot{\alpha}_o(t_r) &= 0. \end{aligned} \quad (18)$$

These equations impose additional constraints on the state at the final time of the motion.

B. Two-point boundary value problem for 6-D goal state

The throwing task is formulated as an optimal control problem with initial and final states and number of constraints including actuator limits, workspace constraints and contact conditions for nonprehensile manipulation.

1) *Constraints*: Optimization constraints are divided into equality and inequality constraints. The equality constraints are presented as a dynamical constraint (7) as well as initial robot state constraints $\mathbf{q}(0) = \mathbf{q}_0$, $\dot{\mathbf{q}}(0) = \mathbf{0}$ and object initial state $\mathbf{p}(0) = \mathbf{p}_0$, $\dot{\mathbf{p}}(0) = \mathbf{0}$. To generate feasible motions, inequality constraints on the controls $u_{min} \leq \mathbf{u}(t) \leq u_{max}$, on the joint positions $\mathbf{q}_L \leq \mathbf{q} \leq \mathbf{q}_U$ and on the angular velocities $\dot{\mathbf{q}}_L \leq \dot{\mathbf{q}} \leq \dot{\mathbf{q}}_U$ are introduced. Workspace constraints are presented as

$$\begin{aligned} l_1 c_1 + l_2 c_{12} + l_3 c_{123} &> x_{min} \\ l_1 s_1 + l_2 s_{12} + l_3 s_{123} &> y_{min}, \end{aligned} \quad (19)$$

where x_{min} and y_{min} are workspace safety boundaries of the table, $c_{ij} = \cos(q_i + q_j)$ and $s_{ij} = \sin(q_i + q_j)$ and l_i is the length of link i . The safety distances from the edges of the table are equal to half of the length of the end-effector. The robot is designed in such a way that self collisions between links are not permitted if the end-effector is within the workspace constraints. Thus, self collision constraints are omitted. Constraints in case of dynamic grasp are (8), (10), (11), (14) and (17). In case of slipping (14), (15) and (17) have to be used. Release constraints (18) are presented as explicit constraints of a second kind on a final time. Additionally, the constraints imposed by the goal definition are discussed in Sec. IV.A.

2) *Objective function*: Incomplete knowledge about the parameters, especially the contact friction, can result in undesired slipping or rolling during the carrying. The 3-DoF robot makes it possible to reduce such effects. The objective function is constructed as a sum of the components from (11) and (14). The terms are represented in a Lagrange form

$$J(\mathbf{u}, t_r) = \int_{t_0}^{t_r} l(t, x(t), u(t)) dt. \quad (20)$$

3) *2PBVP statement*: Find the control variable and free final time t_r that in case of known model parameters minimizes the functional

$$J(\mathbf{u}, t_r) = \int_0^{t_r} \begin{bmatrix} k_{sl} & k_{rol} \end{bmatrix} \begin{bmatrix} J_{sl} \\ J_{rol} \end{bmatrix} dt, \quad (21)$$

where k_{sl} and k_{rol} are the weighting coefficients for (11) and (14). In case of undefined friction coefficient we get

$$J(\mathbf{u}, t_r) = \int_0^{t_r} J_{\mu_s, sl} dt, \quad (22)$$

When several parameters are uncertain the objective function takes the following form

$$J(\mathbf{u}, t_r) = \int_0^{t_r} \left(\tilde{k}_{rol} |\ddot{\alpha}_o| + \mathbf{Q}^T \begin{bmatrix} |\beta| \\ -\ddot{w}_o \\ |\dot{v}_o| \end{bmatrix} \right) dt \quad (23)$$

subject to the constraints from Sec. III.B.

An initial time t_0 is given and fixed. \tilde{k}_{rol} and \mathbf{Q} represent the weighting coefficients for anti-rolling and mutual anti-slipping and anti-rolling costs for completely uncertain parametric model. It increases the robustness of the whole motion. If the final state of the end-effector is defined, inverse kinematics is used to obtain the final state of the robot. Therefore, special attention is paid to two different admissible postures of the three-link planar arm as sketched in Fig. 4.

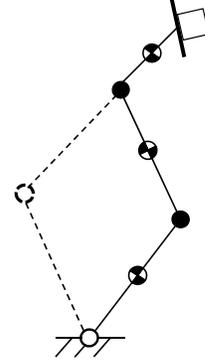


Fig. 4: Admissible release postures.

An additional constraint on the internal structure of the robot is imposed to receive well-defined final state of all 3-DoF at terminal time. For simplification, the elbow-down throwing posture is chosen. Although, in general, it has to be selected based on the optimization problem which is solved for each posture separately.

IV. SIMULATION AND NUMERICAL SOLUTION

A. Goal manifold

At the time of release t_r the state of the object in the base frame is described with position x , y and orientation ϕ , translational velocities V_x , V_y and rotational velocity ω . The object travels along the parabolic trajectory defined by these parameters. Using the inverse differential kinematics equation

$$\dot{\mathbf{q}} = \mathbf{J}^{-1}(\mathbf{q}, \mathbf{r}) \begin{bmatrix} V_x & V_y & \omega \end{bmatrix}^T, \quad (24)$$

three constraints are imposed on the robot. By integration of both sides of (24) these constraints come to a new form

$$\int_0^{t_r} \dot{\mathbf{q}} dt = \mathbf{q}(t_r) - \mathbf{q}(0) = \mathbf{f}(\mathbf{q}(t_r), \mathbf{q}(0), \mathbf{p}(t_r), \mathbf{p}(0)), \quad (25)$$

where \mathbf{f} is a nonlinear function that represents the integrals of the right side of (24). In (25) the initial position $\mathbf{q}(0) = \mathbf{q}_0$ is given, $x(0)$, $y(0)$ and $\phi(0)$ are obtained

through forward kinematics

$$\begin{bmatrix} x \\ y \\ \phi \end{bmatrix} = \begin{bmatrix} l_1 c_1 + l_2 c_{12} + l_3 c_{123} \\ l_1 s_1 + l_2 s_{12} + l_3 s_{123} \\ q_1 + q_2 + q_3 - \frac{\pi}{2} \end{bmatrix}. \quad (26)$$

Since (25) consists of five unknowns $\mathbf{q}(t_r)$, $x(t_r)$ and $y(t_r)$ and \mathbf{f} is of dimension three, the system is underconstrained. However, additional constraints on the motion of the object can be imposed. Algebraic equations for the parabolic trajectory of the object that lead to a final goal are presented

$$\begin{aligned} x(t_r) &= x_m - V_x t_{\text{hit}} \\ y(t_r) &= y_m + \frac{1}{2} g_m t_{\text{hit}}^2 - V_y t_{\text{hit}} \\ \phi(t_r) &= \phi_m - \omega t_{\text{hit}} - 2\pi n, \end{aligned} \quad (27)$$

where t_{hit} is the time when the object center of the mass hits a prescribed mark, n is the number of full rotations. The variables $x(t_r)$ and $y(t_r)$ are both dependent on the variable t_{hit} . Thus, substituting (27) into (25) reduces the number of unknowns from five to four. Therefore, different variations of t_{hit} lead to the same goal object position. Hence, the goal state of the object is reached with an infinite number of possible final states of the robot $\mathbf{x}(t_r)$. The six object state variables at release time define the terminal constraint and the final state. Depending on a number of assigned state variables there are 64 possible combinations of defined parameters without repetitions

$$\sum_{i=0}^6 \binom{6}{i} = 64, \quad (28)$$

including cases when the whole 6-D object state is defined, and when none of the parameters is given. We would like to constrain ourselves to the discussion about cases when the destination position is present. In total, there are 16 cases, as depicted in Fig. 5.

Number of fixed object DoFs	2							
	3							
	4							
	5							
	6							

Fig. 5: Classification of the possible goal states of the object with fixed center of the mass.

Example: If we fix 5-Ds of the object and specify the final orientation of the end-effector, it will be equivalent to specifying the 6-Ds of the object. The resulting manifold with varying time in the workspace of the robot is presented in Fig. 6.

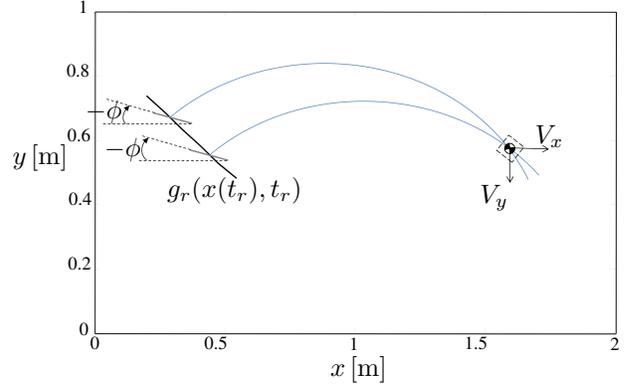


Fig. 6: Manifold of final end-effector position with constant V_x and V_y . The throw is taken from any point lying on the manifold with the angle $\phi(t_r)$ but without ω so that it results in hitting the prescribed mark with the center of gravity of the object.

Cases from Fig. 5 are discussed in detail. The analysis is split in two parts. First, we discuss the parabolic trajectory described by the first two equations of (26). When only the position is determined, the throw is made from any reachable point in the workspace. In case of one velocity being defined we observe the set of terminal manifolds dependent on the flight duration, see Fig. 7.

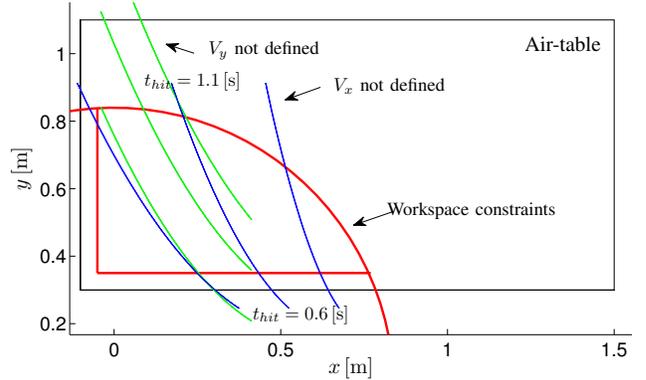


Fig. 7: Manifolds of final end-effector position with varying V_x (set of non-intersecting blue curves) and V_y (set of non-intersecting green curves). Increase of V_x results in moving the manifold left, increase of V_y results in lowering the manifold. The upper point of every curve is taken with $t_{\text{hit}} = 1.1$ [s], the lower point of every curve is taken with $t_{\text{hit}} = 0.6$ [s].

Finally, if both velocities are defined, there exists a unique manifold $g_r(x(t_r), t_r)$ in the workspace, e.g. see Fig. 6. Second, angular rotation is defined by (26). If the ϕ and ω are not assigned, then any release orientation fits. Let us describe the rotational term in the case where ω is not prescribed, thus the terminal position manifold is identical to the one from Fig. 8. In this case it is important to set the rotation direction and number of full rotations n . In case where the ϕ is not given, we have the set of curves as shown in Fig. 8. The last case is when the ϕ and ω are both defined. Thus we have a unique point for end-effector release angle. Any case from Fig. 5 can be analyzed using these manifolds in the workspace.

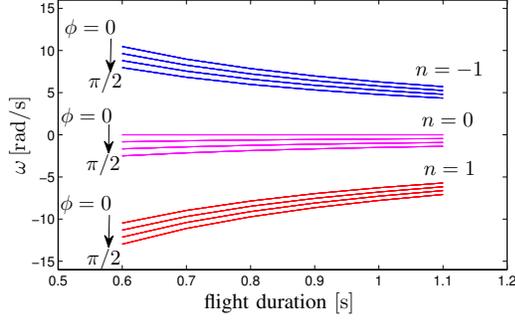


Fig. 8: Angular velocity of the end-effector dependent on the robot release orientation and number of object full rotations.

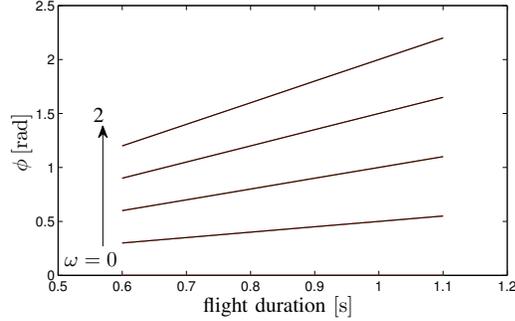


Fig. 9: Angular position of the end-effector dependent on the release angular velocity.

B. Numerical solution

For optimal control simulation we used DirCol code with the sparse SQP nonlinear method SNOPT to solve the nonlinear optimization problems (NLPs) [11]. This code is a direct collocation method that discretizes the control and the state. Thus, the infinite dimensional problem is divided in a number of optimal point-to-point trajectories with state constraints. These trajectories are used later for tracking control. Simulation results for the throwing task with the 3-DoF robot are shown in Fig. 10.

The 2PBVP is sequentially simplified to get the most suitable form for the numerical optimization. First, the boundary value constraint for initial and final states is divided into separate constraints. The initial state $\mathbf{x}(0)$ consists of $\mathbf{q}(0) = \mathbf{q}_0$ and $\dot{\mathbf{q}}(0) = \mathbf{0}$, and the terminal constraint is a manifold, which is mathematically modeled as $g_r(\mathbf{x}(t_r), t_r) = 0$ with free t_r . The manifold from Sec. IV.A is transferred to the state space using inverse kinematics and inverse velocity kinematics. For the optimal control problem this manifold is presented as an implicit boundary of the second kind, i.e. it is a condition for time t_r .

V. EXPERIMENTAL VERIFICATION

A. Experimental setup

1) *Manipulator*: The robot manipulator was built using aluminum hollow links and RE-40 Maxon motors. The arm is located at the edge of the table. Every single joint is actuated by a separate motor. The end-effector was designed as a flat plastic end-effector to implement planar manipulations with the objects. The control is simulated and executed using the

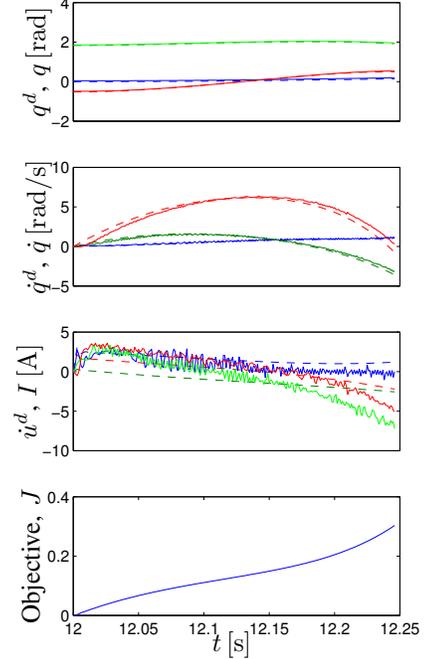


Fig. 10: Desired (dashed) and actual (solid) positions, velocities and currents of the robot arm, and optimal cost J .

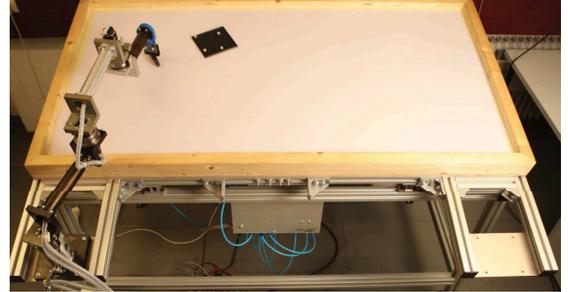


Fig. 11: Front-view of planar robotic arm mounted on the inclined air-table.

Matlab Simulink with built-in real-time Linux environment PRT. Mass-inertia parameters, as well as centers of mass of the links are found from the CAD model, where the robot was designed. The dynamical model of the 3R arm was simulated in Autolev.

2) *Air-table*: An inclined air-hockey table with dimensions of 1.6m x 0.8m serves as a testbed for simulation of planar object manipulation with decreased gravity. The air-flow is equally distributed through more than thousand of holes drilled in the surface. The inclination of the table is variable. For example, with the angles $\theta = 24^\circ$ and $\theta = 10^\circ$, the air-table produces $g_{m1} = 3.99 \text{ m/s}^2$ and $g_{m2} = 1.7 \text{ m/s}^2$. Thus the object movement on the table is comparable to the object flying in the gravitational field of Mars and Moon respectively.

3) *Objects*: The manipulated objects are convex polyhedrons made by rapid prototyping. The orientation and angular velocity during the nonprehensile manipulation plays a more significant role for these objects than for the circular objects.

B. Sources of uncertainty

Due to the absence of a plenum chamber in the constructed air-hockey table, the airflow distribution on the surface is not perfect and leads to a slight change in the object velocities. Another source of unmodeled dynamics is a highly nonlinear friction in the robot joint gears. The control scheme using optimal histories for torque in feedforward loop is used to overcome this problem. Desired position and velocities are tracked in feedback loops with PD controller. The schematic is shown in Fig. 12.

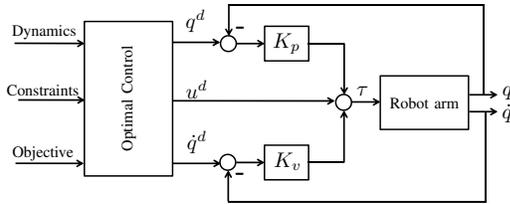


Fig. 12: The control structure consisting of presented optimal control block and feedforward compensation [12].

C. Experimental results

Each joint is controlled in a way that reduces the objective function (21). Using the obtained desired trajectory histories for optimal states and control, the throwing motion is executed successfully. Experimental results are shown in Fig. 13 and Fig. 14.

Three experiments with a robust nonprehensile throw were carried out. Several convex objects were released from different points on the manifold and resulting trajectories were obtained using the tracking system.

VI. CONCLUSIONS AND FUTURE WORK

By using a well defined 2PBVP with researched dynamics, task constraints and objectives, a motion planning algorithm for nonprehensile throw has been presented. It was shown that the goal definition leads to different types of constraint manifolds. Therefore, the proposed method takes advantage of task redundancy. An objective function consisting of components responsible for robustness of task execution chooses a single 6-dimensional point on this manifold.

This method can be generalized to the other multiple contact tasks with a flat end-effector that deal with slipping

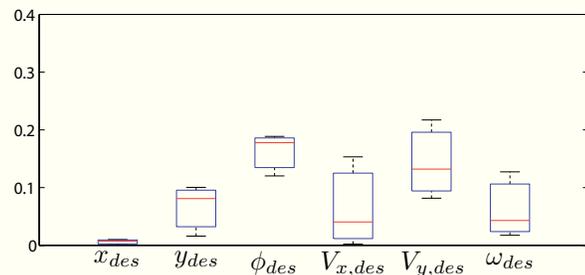


Fig. 13: The object state error from the desired 6-dimensional goal. In all the experiments states are taken in the vicinity of fixed x_{des} . The central red line is the median, the edges of the box are the 25th and 75th percentiles.

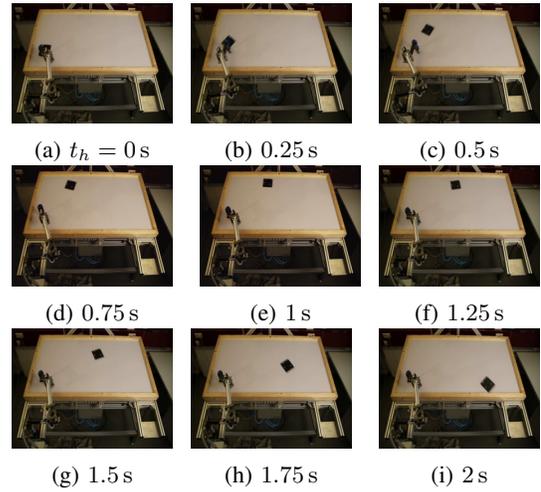


Fig. 14: Frames of the throwing experiment.

and rolling. The future work will focus on the multiple objects nonprehensile manipulation including robot-robot cooperation, task synchronization and online action planning.

ACKNOWLEDGMENT

The authors would like to thank Dirk Wollherr, Moritz Stoetter, and Philine Donner for giving valuable comments to the paper. This work is supported in part within the ERC Advanced Grant SHRINE Agreement No. 267877 (www.shrine-project.eu).

REFERENCES

- [1] S. Akella, W. Huang, K. Lynch, and M. Mason, "Sensorless parts feeding with a one joint robot," *Algorithms for Robotic Motion and Manipulation*, pp. 229–237, 1996.
- [2] H. Frank, D. Barteit, and F. Kupzog, "Throwing or shooting - a new technology for logistic chains within production systems," in *Technologies for Practical Robot Applications. IEEE Int. Conf. on*, pp. 62–67, nov. 2008.
- [3] M. T. Mason and K. M. Lynch, "Dynamic manipulation," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 152–159, 1993.
- [4] K. M. Lynch and M. T. Mason, "Dynamic underactuated nonprehensile manipulation," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 889–896, 1996.
- [5] H. Frank, A. Mittnacht, T. Moschinsky, and F. Kupzog, "1-dof-robot for fast and accurate throwing of objects," in *Emerging Technologies Factory Automation. IEEE Conf. on*, pp. 1–7, sept. 2009.
- [6] H. Frank, T. Frank, A. Mittnacht, and C. Sichau, "A bioinspired 2-dof throwing robot," in *AFRICON*, pp. 1–6, sept. 2011.
- [7] H. Miyashita, T. Yamawaki, and M. Yashima, "Control for throwing manipulation by one joint robot," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp. 1629–1634, 2009.
- [8] H. Miyashita, T. Yamawaki, and M. Yashima, "Parts assembly by throwing manipulation with a one-joint arm," in *Intelligent Robots and Systems, IEEE/RSJ Int. Conf. on*, pp. 43–48, oct. 2010.
- [9] T. Tabata and Y. Aiyama, "Passing manipulation by 1 degree-of-freedom manipulator - catching manipulation of tossed object without impact," in *Assembly and Task Planning. Proc. of the IEEE International Symposium on*, pp. 181–186, july 2003.
- [10] W. Mori, J. Ueda, and T. Ogasawara, "1-dof dynamic pitching robot that independently controls velocity, angular velocity, and direction of a ball: contact models and motion planning," in *Proceedings of the IEEE Int. Conf. on Robotics and Automation*, pp. 2179–2185, 2009.
- [11] O. Von Stryk, "User's guide for dircol (version 2.1): a direct collocation method for the numerical solution of optimal control problems," *Report, Lehrstuhl M2 Höhere Mathematik und Numerische Mathematik, Technische Universität München*, 1999.
- [12] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: modelling, planning and control*. Springer, 2011.