Noninteracting Constrained Motion Planning and Control for Robot Manipulators

Manuel Bonilla[1] Lucia Pallottino[2] and Antonio Bicchi[1,2]

Abstract— In this paper we present a novel geometric approach to motion planning for constrained robot systems. This problem is notoriously hard, as classical sampling-based methods do not easily apply when motion is constrained in a zero-measure submanifold of the configuration space. Based on results on the functional controllability theory of dynamical systems, we obtain a description of the complementary spaces where rigid body motions can occur, and where interaction forces can be generated, respectively. Once this geometric setting is established, the motion planning problem can be greatly simplified. Indeed, we can relax the geometric constraint, i.e., replace the lower–dimensional constraint manifold with a full-dimensional boundary layer. This in turn allows us to plan motion using state-of-the-art methods, such as RRT*, on points within the boundary layer, which can be efficiently sampled. On the other hand, the same geometric approach enables the design of a completely decoupled control scheme for interaction forces, so that they can be regulated to zero (or any other desired value) without interacting with the motion plan execution. A distinguishing feature of our method is that it does not use projection of sampled points on the constraint manifold, thus largely saving in computational time, and guaranteeing accurate execution of the motion plan. An explanatory example is presented, along with an experimental implementation of the method on a bimanual manipulation workstation.

I. INTRODUCTION

Motion planning and control algorithms for robots interacting with the environment have been mostly studied separately. This separation of concerns typically sees first a motion planning phase for the constrained system, usually dealing with geometry and kinematics, followed by an execution phase, dealing with dynamics, where the planned trajectory is accurately executed possibly with some type of force control. This approach works well in cases where the robot-environment model is good enough, but it requires an extremely accurate (and time-consuming) planning phase. Also, there is no guarantee that errors in the constrained motion model will not generate unacceptable errors in the force interaction with the environment, nor that the force control execution will not interfere with the planned trajectory, causing violation of constraints and their consequences (e.g. bumping into obstacles, or loosing the grip on a manipulated object). This fact makes robot interactions dangerous and motivates the use of compliance in robot mechanisms.

Indeed, it should be noticed that an ever increasing number of robotic systems designed for interaction use compliant actuation. There are many examples including – but not limited to – Variable Stiffness Actuators (VSA) [1], robotic
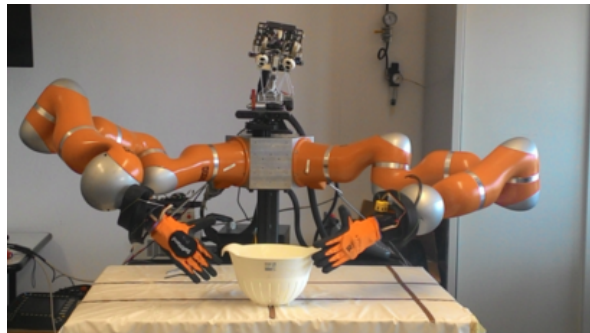
[1]Department of Advanced Robotics, Istituto Italiano di Tecnologia, Via Morego 30, 16163 Genova, Italy
[2]Research Center "E. Piaggio", Università di Pisa, Largo Lucio Lazzarino 1, 56122 Pisa, Italy

Fig. 1. VITO robot from research center "E Piaggio" at University of Pisa. This system is composed by two KUKA LWR arms controlled with compliance control. A PISA/IIT soft hand is mounted on each arm.

hands like the PISA/IIT soft hand [2], robotic arms like the DLR Hand-Arm system [3] and humanoid robots like COMAN [4] and the ones recently presented in the DARPA Robotics Challenge, see Fig. 1 for an example. These new robot structures have been recently addressed as soft robots. Their design are inspired by observing how human body use its compliance (mainly in muscles) to be robust to environment interactions.

While most of the attention has been focused in embedding compliance in the robot structure [4] or include it via the motor controllers [5], methods to plan motions for robots are still being developed for rigid structures [6], [7], [8], [9]. The main reason is that planning and control for robots in interactions is harder because the space describing robot interactions combines force and position spaces, which are objects from different physical meaning and geometric natures [10].

A. Integrating planning and control

There are many alternatives to integrate motion planning and control, recently these two problems have been combined in a unified optimization problem [11], [12], however, these approaches are local solutions and depend on an initial guess which is not easy to define. To deal with this problem randomized techniques [13] have been used, they rely on the use of reduced and linearized model of the system to connect any pair of sampled configurations. Such techniques have been mainly used for mobile robots like quadrotors.

In this paper we present new approach to generate motions for constrained mechanical systems. It is based on results coming from geometric control which suggest an organization of the output vector to be functionally controllable [14]. Such organization allows to find basis for the subspaces

describing the allowable rigid body motions and interaction forces that the system can execute [15]. Based on this result, it is possible to decouple the motion planning problem from the control of interaction forces in two separate aspects.

As the main contribution of this work we first present a geometric control scheme to achieve noninteracting control of force and position subspaces. This property allows us to regulate interaction forces that may be generated by displacements in the perpendicular direction to the tangent space of the constraint manifold. It means that errors generated in the motion planning phase are irrelevant to the control of interaction forces and vice versa. This simplify the motion planning problem so we may deal with it by relaxing the geometric constraint, i.e., replacing the lower–dimensional constrained manifold with a narrow but full-dimensional boundary layer [16].

This allows us to plan motion using state-of-the-art methods, such as RRT* to find global solutions, on points within the boundary layer, which can be efficiently sampled [17]. Such plan however will not necessarily satisfy the task constraint, as it generates displacements with respect to the constraint manifold. If used on a real compliant robot, this plan would generate motions that may drive the mechanism to generate undesired internal forces which will be regulated by the controller.

The rest of the paper is organized as follows. Section II presents the mathematical model of compliant systems under interactions. In section III, as the main contribution of this paper, we show that it is possible to decouple, and control independently, rigid motions in the constraint manifold and internal (vincular) forces in the complementary direction. Section IV recall a motion planning algorithm for closed kinematic chains presented in [16]. Section V presents the setup to integrate motion planning and control while VI provides two examples. Finally section VII presents the conclusions and future work of this research.

II. MODELING COMPLIANT MANIPULATION SYSTEMS

For the sake of simplicity, in this section we recall the theory presented in [14] to describe a manipulation system with general kinematics, specifically Theorem 1 can be rephrased as follows. Consider a mechanism with n actuated joint variables q, and c constraints $C_c(q) = 0$. Let m denote the residual mobility of the system (i.e. the number of dofs compatible with the constraints). Now consider the constraints as elastic elements, i.e., remove the constraint and allow for violations $C_c(q) \leq \epsilon$ and associate internal forces $f = K\epsilon$. K can be interpreted as a suitable stiffness matrix resulted from contacts and joints stiffness. Looking at the space of internal forces that can be actively controlled by joint torques, let p be the dimension of self-balanced (internal) forces, i.e., those which do not affect the overall position of the mechanism, a linearized model, with no disturbances, of the lumped–parameter compliant model for the multiple robot–object dynamics is,

$$\dot{x} = Ax + B_\tau \tau' + B_\omega \omega, \qquad (1)$$

defined around the equilibrium configuration

$$x = \begin{bmatrix} q_{eq}^T & u_{eq}^T & 0^T & 0^T \end{bmatrix}^T$$
$$\tau' = \tau - J^T t_{eq}$$
$$\omega = G t_{eq}.$$

Where q_{eq} and u_{eq} stand for the equilibrium joint positions and object position respectively, J is the Jacobian of the contact points and G is known as grasp matrix. Under the assumptions reported in the previously mentioned work, the dynamics matrix A, joint torque input matrix B_τ, and external disturbance matrix B_ω have the form

$$A = \left[\begin{array}{c|c} 0 & I \\ \hline -L_k & -L_b \end{array} \right], \ B_\tau = \left[\begin{array}{c} 0 \\ 0 \\ \hline M_h^{-1} \\ 0 \end{array} \right]; B_\omega = \left[\begin{array}{c} 0 \\ 0 \\ 0 \\ \hline M_o^{-1} \end{array} \right],$$

where

$$L_k = M^{-1} P_k; \ L_b = M^{-1} P_b,$$

$$M = \begin{bmatrix} M_h & 0 \\ 0 & M_o \end{bmatrix}$$

$$P_k = \begin{bmatrix} J^T \\ -G \end{bmatrix} K \begin{bmatrix} J & -G^T \end{bmatrix}$$

$$P_b = \begin{bmatrix} J^T \\ -G \end{bmatrix} B \begin{bmatrix} J & -G^T \end{bmatrix},$$

M_h and M_o are the multiple-arm and object dynamic matrices. K and B_q are the stiffness and damping matrices at the contact points.

We are interested in the combinations of states giving object positions and internal forces as outputs, which can be respectively selected by matrices $C_u = \begin{bmatrix} 0 & I & 0 & 0 \end{bmatrix}$ and $C_t = \begin{bmatrix} KJ & -KG^T & BJ & -BG^T \end{bmatrix}$. The output matrix is hence

$$C = \begin{bmatrix} \Gamma_u^+ C_u \\ E^+ C_t \end{bmatrix}.$$

Image space of matrices Γ_u consists of rigid body motions of the object being manipulated while E is a base matrix for internal forces which are asymptotically reproducible from joint torques τ.

For non-redundant mechanism, Theorem 1 in [14] states that $m + p = n$. This means that the input-output representation of the minimal A-invariant functionally controllable subspace of states is square. This implies that it is possible to devise a linear controller that uses the n inputs to decouple and to control independently rigid motions in the constraint manifold and internal forces in the complementary direction.

III. NONINTERACTING CONTROL

The control objective is to regulate interaction forces in a manipulation system generated by displacements, that may appear during the planning phase, in the perpendicular direction of the constraint. In practice it is useful to control each of the outputs independently, in this way we are able to regulate each output without affecting the others. Finding a control law where the input i affects just the corresponding

output i is known as noninteracting control. The procedure is to differentiate the output vector $y = Cx$ until the control appears. In case of the output corresponding to $E^+ C_t = E^+[KJ \ -KG^T \ BJ \ -BG^T]$, the control τ_t^* appears in the first derivate

$$
\begin{aligned}
\dot{y} &= C_t A x + C_t B_\tau \tau_t^* \\
&= C_t A x + E^+ B_q J M_h^{-1} \tau_t^*.
\end{aligned}
$$

For the output τ_u^* corresponding to the object motions $\Gamma_u^+ C_t = \Gamma_u^+[0 \ I \ 0 \ 0]$ it is necessary to compute up to the third derivate. Indeed,

$$
\begin{aligned}
\dot{y} &= C_u A x + C_u B_\tau \tau_u^* \\
&= C_u A x + 0 \tau_u^* \\
\ddot{y} &= C_u A^2 x + C_u A B_\tau \tau_u^* \\
&= C_u A^2 x + 0 \tau_u^* \\
\dddot{y} &= C_u A^3 x + C_u A^2 B_\tau \tau_u^* \\
&= C_u A^2 x + \Gamma_u^+ M_o^{-1} G B_q J M_h^{-1} \tau_u^*.
\end{aligned}
$$

Thus the corresponding output vector can be rewritten as

$$
\hat{y} = P x + Q B_\tau \tau^*
$$

where $\hat{y} = [\dddot{y}_u \ \dot{y}_t]^T$ and $\tau^* = [\tau_u^* \ \tau_t^*]^T$,

$$
P = \begin{bmatrix} C_u A^3 \\ C_t A \end{bmatrix} \text{ and } Q = \begin{bmatrix} C_u A^2 \\ C_t \end{bmatrix}.
$$

Concluding, for the system (1) , the control law

$$
\tau^* = -Q^{-1}(P x + \tau_n),
$$

provides the output vector

$$
\hat{y} = \tau_n.
$$

In other words the proposed control law is noninteracting since the rejection of position errors does not influence the force control and vice versa. The next step is to design a stable controller in τ_n designed in the planning phase as described in the following Section.

IV. MOTION PLANNING

Once we have a closed loop system that guarantees the correct robot environment interaction, and advantages of the fact that position errors have no influence in force regulation, we study the motion planning problem for constrained mechanisms where it is well known that interactions limit the valid motions of the mechanisms. Valid motions are represented by constraints $C_c(q) = 0$ which describe a non linear submanifold ($\mathcal{CS}_s$) in the configurations space ($\mathcal{CS}$) of the system. The main characteristics of $\mathcal{CS}_s$ is that it has lower dimension than $\mathcal{CS}$ and that there is not an explicit parameterization of it. This is a problem in motion planning since it is not easy to generate configurations for the mechanism that can eventually describe a path to go from one point to another while always being on $\mathcal{CS}_s$.

To solve this problem we adopt the algorithm presented in our previous work in [16]. In that work we started from the consideration that for a interacting system the constraint can be replaced to $C_c(q) \le \epsilon$. This approach consists in relaxing
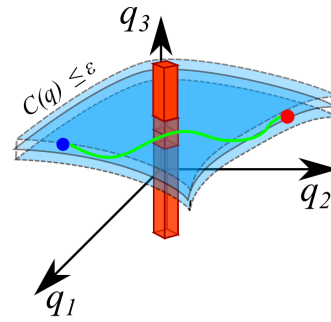

Fig. 2. Motion planning problem under relaxed constraints (boundary layer). Initial position q_{init} in blue. Final position q_{final} in red. Planned path in green. Constraint $C(q)$ in light blue.

(a) Undesired forces arising from planning on the relaxed constraint

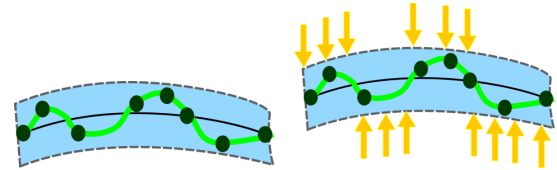(b) The task of the controller is to project the undesired forces back to the manifold

Fig. 3. Lateral view of the relaxed constraint. In green are the pushing and pulling forces against the constraint. The black dots are the nodes extracted from the tree generated by the soft-RRT*.

the constraint in the planning phase, effectively replacing the lower–dimensional constraint manifold with a narrow but full–dimensional boundary layer. Motion is planned using RRT* based on points which can be sampled in the boundary layer in an efficient manner using an adaptive k-d tree. This approach is represented in Fig. 2. The resulting plan does not satisfy the constraint, see Fig. 3, and generates displacements which, with the model of compliance in the system, can be regarded as interaction forces acting outside the tangent space to the constraint manifold.

The state-of-the-art in path planning under constraints deals with projecting a sampled point onto the constraint manifold. Essentially none of these methods try to achieve "perfect projection", actually for computational considerations they use a threshold to check if they are or not in the manifold. Even though this threshold can be interpreted as a relaxation there is not a formal analysis about how small or big it should be. In this paper we give to this threshold a different meaning which actually comes from physics. Depending on the compliance on the system and the maximum forces f_{max} to which the system can resist, we can compute the threshold using the compliance in the system $\epsilon = K^{-1} f_{max}$.

V. INTEGRATING PLANNING AND EXECUTION FOR SOFT ROBOTS

In order to implement the control and motion planning algorithms presented in previous sections we use them in the pipeline explained in Fig. 4. It consists in first finding a free obstacle path, within the boundary layer, using the
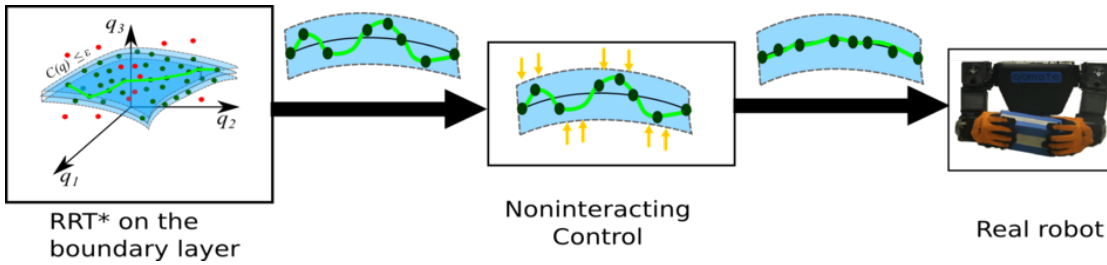
Fig. 4. Pipeline of the constrained motion planning approach proposed in this paper. First the kinematic planning is performed in fully dimensional boundary layer obtained from the ϵ relaxed constraint. Then using the noninteracting control the trajectory is projected to the constraint and the interaction forces are regulated to zero.

random sample-based planning approach presented in section IV, a graphical interpretation is depicted in Fig. 3(a). It is important to note that if only the relaxed constraint is verified, the interactions with the environment can be dangerous. For example, if we implement the relaxed path in a system of multiple robots manipulating an object, it can either fall or be damaged by possible high squeezing forces. On the other hand, whenever the nominal closed kinematic constraint is verified this problem does not occur. In order to guarantee a correct execution of the plan, and at the same time guarantee a correct interaction, we use the filter presented in section III. It is applied to the path to project any position errors back to the constraint manifold and at the same time regulate the interaction forces, see Fig. 3(b). While these two approaches seem to be decoupled, they are joined by the common consideration of the compliance in the system.

VI. EXAMPLES AND EXPERIMENTAL RESULTS

This section presents a simple but explanatory example to show the validity of the proposed approach. It also includes experimental validation in a bimanual manipulation system.

A. Explanatory Example

As a first example, consider a robotic hand composed by two fingers (each one with one degree of freedom) holding an object, see Fig. 5. All valid configurations are described by the following constraint manifold

$$C(q) = F_1(q_1) - F_2(q_2) = 0, \qquad (2)$$

where $F_1(q_1)$ and $F_2(q_2)$ stand for the forward kinematics of the fingers until the contact points.

The dimension of the $\mathcal{CS}$ for this example is 2 while the dimension of the state space for the linear system used in the second stage is 10. The parameter defining the relaxation limits was $\epsilon = 0.5$. For this example just stiffness at the contact points was considered. The outcome of the first stage is depicted in Fig. 6 where the k-d tree approximation of the boundary layer is shown together with the path generated by the soft-RRT* algorithm.

For the second stage of the algorithm, we linearized the dynamic model of the hand on the configuration $q_{init} = [0\ 0]$ which is the same starting position for the first stage. The output trajectory resulting from the first and second stage are compared in Fig. 7. The maximum constraint violation
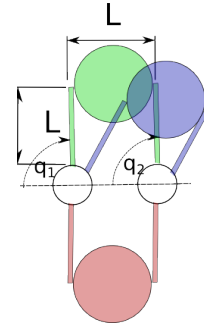

Fig. 5. Robotic hand used as an explanatory example. Initial position is represented in green, final position in red and an intermediate position in blue.
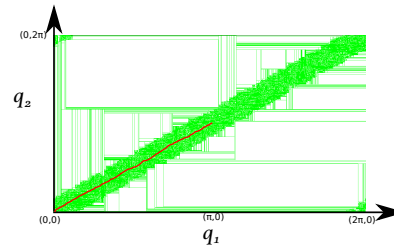

Fig. 6. This figure shows reconstructed boundary layer obtained using the adaptive k-d tree, the sampling procedure converges to a uniform distribution in the relaxed constraint.

in the first phase was 0.1486 while in the second part was reduced to 0.00864.

B. Bimanual Manipulation Example

The second example is meant to evaluate the performance of the presented approach when working in high dimensional spaces. The systems is depicted in Fig. 1, it is composed by two 7 degrees of freedom arms equipped with a PISA/IIT hand each. The theory presented in this paper, specifically the non interacting controller, was developed for non redundant manipulators so we are using just 6 degrees of freedom of each arm. The interaction forces were estimated by the joint torque sensors included in the arms.

The dimension of the relaxed $\mathcal{CS}_s$ in this example is 12, while the dimension of the state space for the filter is 36.

The experiments were performed using a laptop computer with 8 Gb of Memory and a i7-4558U CPU @ 2.80GHz × 4 processor. All codes were developed in C++ under Ubuntu
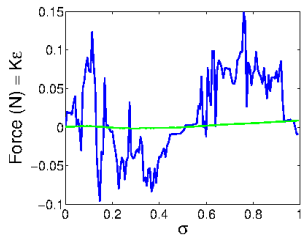
Fig. 7. Comparison of forces resulting from the planned path in the first stage (blue) and forces filtered in the control phase (green)
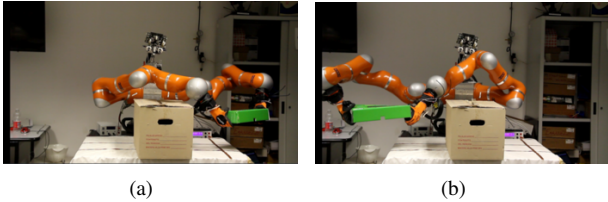

(a)   (b)

Fig. 8. Initial (a) and final (b) configurations for the example 2.

14.04. Fig. 8 shows the initial and final configurations of the experiment. The first path generated by the planner was at 4.32s of execution and was stopped at 10s. The filter took 2.33s to smooth each of the paths. The results are illustrated in Fig. 9. In the same Figure the trajectories resulting from different planning times are shown, it is evident that the trajectory becomes smoother as the planning time increases. In the same way the interaction forces generated during the planning phase approaches zero. It can be observed also in the Fig. 10 that forces are filtered and regulated to a desired reference with the non interacting controller. A video of the experiment is attached to this paper.

VII. DISCUSSIONS AND CONCLUSIONS

In this paper we presented an integrated approach for motion planning and control of robots when they are interacting with the environment or with the mechanism itself. The approach was developed first by designing a noninteracting controller to regulate independently the subspaces describing position, consistent with the constraint, and forces, acting in the normal direction. This allowed us to address the motion planning an control problems separately since they does not affect to each other.

Thanks to the noninteracting property of the controller we were able to relax the motion planning problem by replacing the lower-dimensional constraint manifold with a narrow but full-dimensional boundary layer. The planner generates a path that satisfy the constraint with a ϵ violation but that are then regulated by the controller. Experimental results have been reported.

Even though theoretically there is not a limitation in the constraint relaxation, in practice it exists. Thus, there is a probability of sampling points in the boundary layer but the probability is still small which is more evident in high dimensional spaces. In practice we observed that introducing in the planner average configurations reduce dramatically the planning time. It suggests the use of motion primitives such

as synergies to reduce the search space of the planners and to help to convergence.

Future work is directed to use the approach presented in this paper for anytime control.
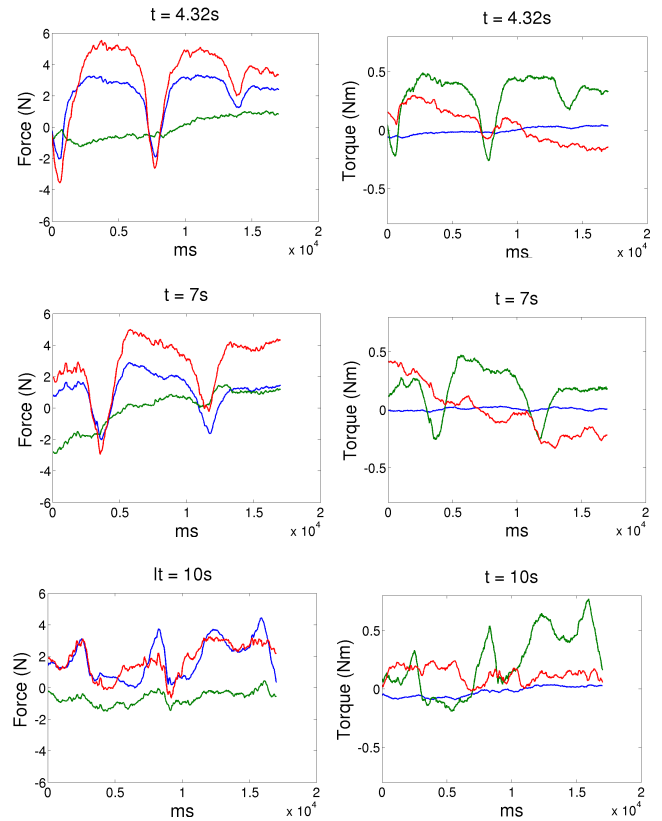

Fig. 10. Comparison of forces resulting from the execution of the planned path extracted from the planner in different times t. Fist column shows the forces in x (blue), y (green) and z (red) axis. Second column shows the torques in x (blue), y (green) and z (red) axis. The reference interaction forces for the noninteracting controller are force and torques equal to 0.

REFERENCES

[1] G. Grioli, S. Wolf, M. Garabini, M. G. Catalano, E. Burdet, D. G. Caldwell, R. Carloni, W. Friedl, M. Grebenstein, M. Laffranchi, D. Lefeber, S. Stramigioli, N. G. Tsagarakis, M. V. Damme, B. Vanderborght, A. Albu-Schaeffer, and A. Bicchi, "Variable stiffness actuators: the user's point of view," *Int. Journal of Robotics Research (IJRR)*, 03/2015 2015.
[2] M. G. Catalano, G. Grioli, E. Farnioli, A. Serio, C. Piazza, and A. Bicchi, "Adaptive synergies for the design and control of the pisa/iit softhand," *Int. Journal of Robotics Research (IJRR)*, vol. 33, p. 768–782, 2014.
[3] M. Grebenstein, A. Albu-Schaffer, T. Bahls, M. Chalon, O. Eiberger, W. Friedl, R. Gruber, S. Haddadin, U. Hagn, R. Haslinger, H. Hoppner, S. Jorg, M. Nickl, A. Nothhelfer, F. Petit, J. Reill, N. Seitz, T. Wimbock, S. Wolf, T. Wusthoff, and G. Hirzinger, "The dlr hand arm system," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, May 2011, pp. 3175–3182.
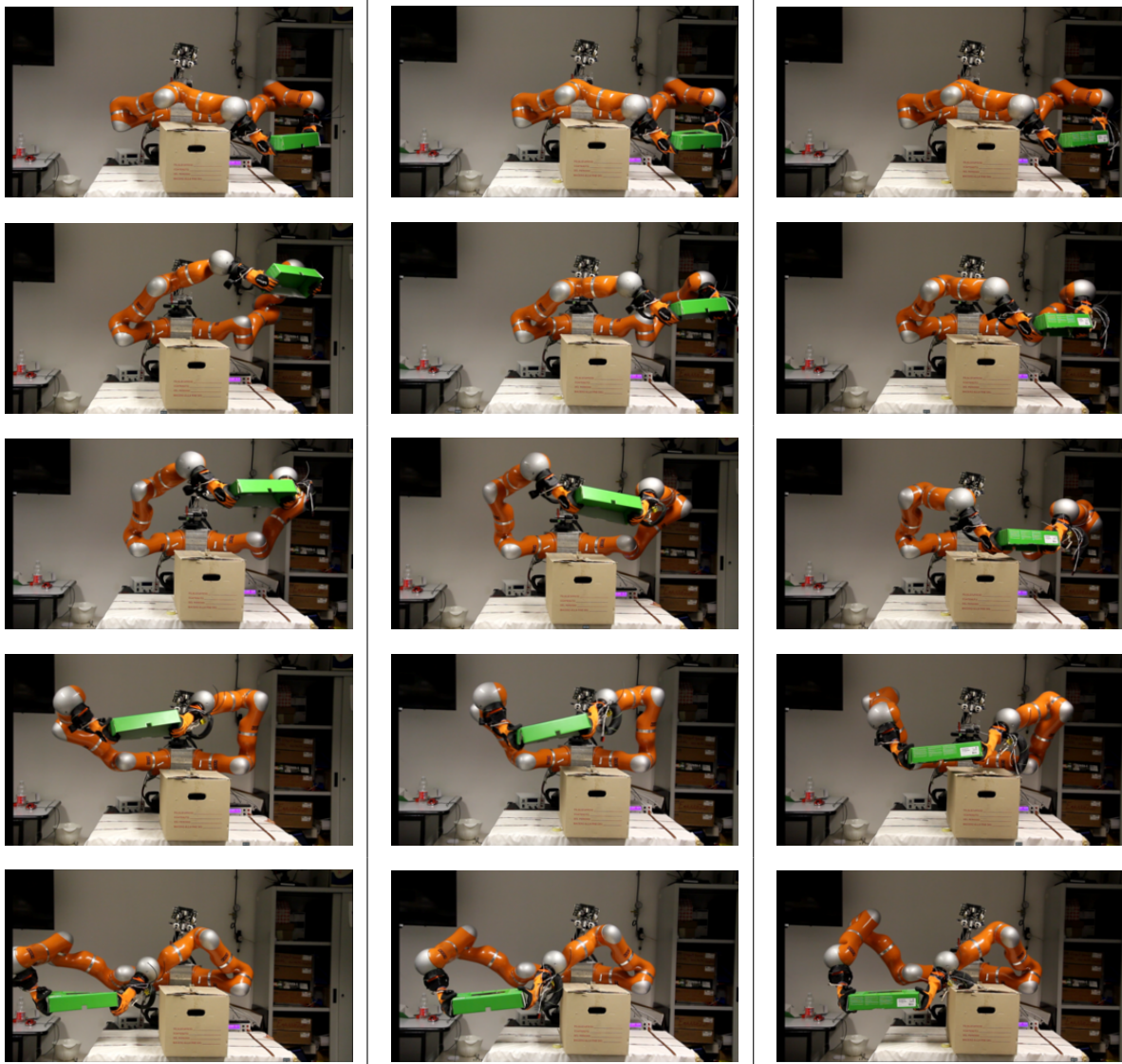
Fig. 9. Snapshots of the experiment using VITO robot. First column shows the first trajectory generated by the planner obtained at 4.32s. Second column is the execution of the trajectory at 7s and the third column is the execution of the trajectory obtained at 10s. It is clear that the trajectory improves with time.

[4] N. Tsagarakis, S. Morfey, G. Cerda, L. Zhibin, and D. Caldwell, "Compliant humanoid coman: Optimal joint stiffness tuning for modal frequency control," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, May 2013, pp. 673–678.

[5] M. T. Mason, "Compliance and force control for computer controlled manipulators," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 11, no. 6, pp. 418–432, June 1981.

[6] J. Yakey, S. M. LaValle, and L. E. Kavraki, "Randomized path planning for linkages with closed kinematics chains," *IEEE Transactions on Robotics*, vol. 17, no. 6, pp. 951–959, 2001.

[7] J. Cortes and T. Simon, "Sampling-based motion planning under kinematic loop-closure constraints," in *Algorithmic Foundations of Robotics VI*. Springer, 2005, p. 7590. [Online]. Available: http://link.springer.com/chapter/10.1007/10991541_7

[8] M. Stilman, "Global manipulation planning in robot joint space with task constraints," *IEEE Transactions on Robotics*, vol. 26, no. 3, 2010.

[9] T. McMahon, S. Thomas, and N. Amato, "Sampling based motion planning with reachable volumes: Application to manipulators and closed chain systems," in *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, Sept 2014, pp. 3705–3712.

[10] G. Liu and Z. Li, "A unified geometric approach to modeling and control of constrained mechanical systems," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 4, pp. 574–587, Aug 2002.

[11] M. Posa, C. Cantu, and R. Tedrake, "A direct method for trajectory optimization of rigid bodies through contact," *Int. Journal of Robotics Research (IJRR)*, vol. 33, no. 1, pp. 69–81, Jan. 2014. [Online]. Available: http://dx.doi.org/10.1177/0278364913506757

[12] S. Alatartsev, A. Belov, M. Nykolaychuk, and F. Ortmeier, "Robot trajectory optimization for the relaxed end-effector path," in *Informatics in Control, Automation and Robotics (ICINCO), 2014 11th International Conference on*, vol. 01, Sept 2014, pp. 385–390.

[13] A. Masoud, "Kinodynamic motion planning," *IEEE Robotics and Automation Magazine*, vol. 17, no. 1, pp. 85–99, Mar. 2010.

[14] D. Prattichizzo and A. Bicchi, "Consistent specification of manipulation tasks for defective mechanical systems," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 119, pp. 767–777, December 1997.

[15] ——, "Dynamic analysis of mobility and graspability of general manipulation systems," *IEEE Transactions on Robotics and Automation*, vol. 14, pp. 241–258, 1998.

[16] M. Bonilla, E. Farnioli, L. Pallottino, and A. Bicchi, "Sample-based motion planning for robot manipulators with closed kinematic chains," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2015.

[17] J. Bialkowski, M. Otte, and E. Frazzoli, "Free-configuration biased sampling for motion planning," in *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013, pp. 1272–1279.