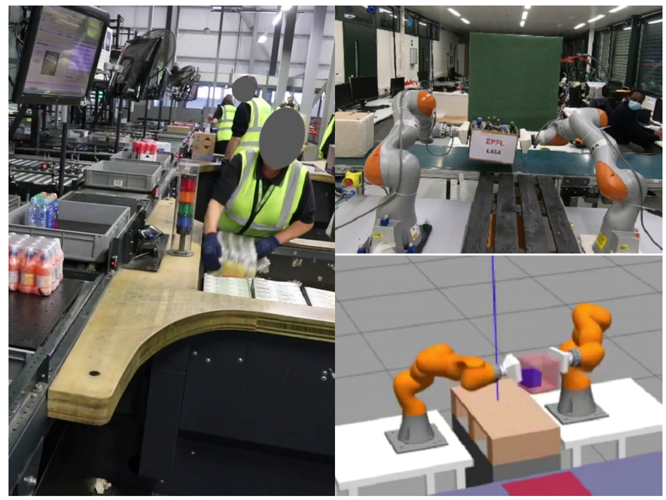
Dual-arm control for coordinated fast grabbing and tossing of an object

Michael Bombile and Aude Billard

***Abstract*—Picking up objects to toss them on a conveyor belt are activities generated on a daily basis in the industry. These tasks are still done largely by humans. This paper proposes a unified motion generator for a bimanual robotic arm system that enables two 7 degrees of freedom robotic arms to grab and toss an object in one swipe. Unlike classical approaches that grab the object with quasi-zero relative contact velocity, the proposed approach is able to grasp the object while in motion. We control the contact forces prior to and following impact so as to stabilize the robots' grip on the object. We show that such swift grasping speeds up the pick-and-place process and reduces energy expenditure for tossing. Continuous control of reach, grab and toss motion is achieved by combining a sequence of time-invariant dynamical systems in a single control framework. We introduce a state-dependent modulation function to control the generated velocity in different directions. The framework is validated in simulation and on a real dual-arm system. We show that we can precisely toss objects within a workspace of 0.2 by 0.4 square meters. Moreover, we show that the algorithm can adapt on the fly to changes in object location.**



(a) (b)

Figure 1. Illustration of a dual-arm manual and robotic pick-and-place operations. (a) human dual-arm grabbing and placing objects in a palletizing task within Vanderlande facility (photo courtesy of Vanderlande). (b) pair of two real and simulated real KUKA LBR IIWA robots grabbing an open box (top) and an object containing a small moving object inside (bottom).

I. Introduction

Swift robot manipulation of objects in unstructured and dynamic environments is crucial for the industry. In logistics, for instance, the booming of e-commerce and its related challenges have increased the need to speed up the pace of pick-and-place operations. In current applications, robots usually pick up and release products with almost zero contact velocity. One solution to speed up the process is to move from this quasi-static approach towards a dynamic one where robots are allowed to grab and release products with non-zero contact velocities. This can be achieved by designing robot controllers that are aware of induced impacts. Planning impact is challenging due to noise in perception and control. In that regard, one important aspect is to generate motion robust to imprecision as to when and how much impact is incurred. Moreover, the motion should be robust throughout the task from grabbing with impact to release be it by placing, handing over or tossing the object.

In this paper, we consider the problem of grabbing and releasing an object in one swipe with a dual-arm robotic system. The desired manipulation task is motivated by the need to perform fast pick-and-place or pick-and-toss operations in a depalletizing context, see Figure 1. Such repetitive and physically demanding work is usually performed by humans for lack of similarly fast, precise, and robust bimanual robot systems. The bimanual tasks envisioned here extend the complexity of the control problem as it requires, in addition to controlling for impact, to enforce coordination of the two arms. A poorly coordinated system, where one arm reaches the object before the other, would lead to uncontrolled impact. Bi-manual pick-and-toss requires precise coordination before and after contact with the object. Coordination at contact ensures that the object is not set off balance at pick-up. Once contact is established, the resulting interaction forces need to be controlled to ensure a stable grasp and to induce the desired velocity on the object for proper tossing. Controlling robustly coordinated motions of multi-arm systems opens the door to a larger variety of tasks. Besides depalletizing, this could include manipulations that are too complex or heavy for a single robot and require two or more robotic arms. Some applications could be fast picking up of open trays or cases, fast picking up of luggage from airport conveyor belts, etc.

Dual-arm control has been extensively studied; see for instance [1] for a review. Several methods have been proposed to coordinate multiple robots' motion [2], [3][4], to control simultaneously robots' motion and forces [5]-[8], and to optimize contact forces at run time, using two robot manipulators [9], or a humanoid robot in [10], using quadratic programming. All previous cited works assume that the object is already grasped by the robot and focus on the post-contact manipulation phase. The free motion phase and its transition towards the contact phase and after the contact phase for placing and tossing were not considered. A first approach to

All authors are with the Learning Algorithm and Systems Laboratory, EPFL, Lausanne, Switzerland, e-mail: {michael.bombile;aude.billard}@epfl.ch.

smoothly coordinate two robotic arms in free-space motion and when making simultaneous contact based on dynamical systems (DS) was proposed in our group [11]. This approach uses a virtual object to constrain and coordinate the motion of the robots. However, it did not control for the force at contact. Our group extended this further in [12] to propose a DS that could generate both motion and forces for the dual-arm system. Yet, these two approaches still assumed quasi-static grab with end-effectors velocities vanishing as they approach the grasping points on the object. Moreover, they ignored the problem of tossing or placing the object once in hand.

Recently, an impact-aware controller formulated within a quadratic-programming (QP) framework was presented in [13] and was applied to dual-arm grabbing of a box with a contact velocity of $0.15\,m/s$. This work offers a powerful approach to control impact with non-zero contact velocities. The QP scheme, however, relies on planning of the grasping motion, making it less robust to imprecise perception or dynamic changes of the object's pose. In our work, instead, we use as motion generators autonomous dynamical systems for their fast and time-independent re-planning abilities and their robustness to perturbations. The bimanual coordination problem is formulated using the Extended Cooperative Task Space (ECTS) representation [14]. We subsequently combine this flexible planning trajectory with a QP to handle the balancing of force constraints. Assuming prior but approximate knowledge of the object's mass and the friction coefficient, the QP generates online interaction wrenches that achieve a stable grasp subject to contact constraints.

The scope of this paper being motion generation, we leave aside the impact dynamics and assume that the associated states' jumps remain within the robots' limits. The interested reader is referred to [15] for the control of impact with states jump mitigation, and [13] or [16] for explicit enforcement of hardware limits during impact generation.

II. Proposed Approach

In the considered dual-arm task, the robotic system is required to reach and swiftly grasp an object and then either toss it, by releasing it at a desired position and with a specific velocity or to place it at a desired location. A schematic illustrating the considered task is given in Figure 2, where the task phases are depicted.

To induce the desired motion on the object when fast picking and when tossing requires that the two robots' arms adopt the required velocity just prior to impact (for fast picking) and prior to release (when tossing). To obtain this behavior, the robot arms must transit, at contact and release time, through desired states expressed in both position and velocity simultaneously. Unlike attractors, these transitory states are not equilibrium points and therefore the robotic system can only transit through such states. Thus, to realize the desired task in a robust way, we proposed an approach based on modulated dynamical systems, where state-dependent functions shape locally the generated motion of the robot - prior to contact or release of the object - such that the motion aligns first with the desired velocity while moving towards the desired contact

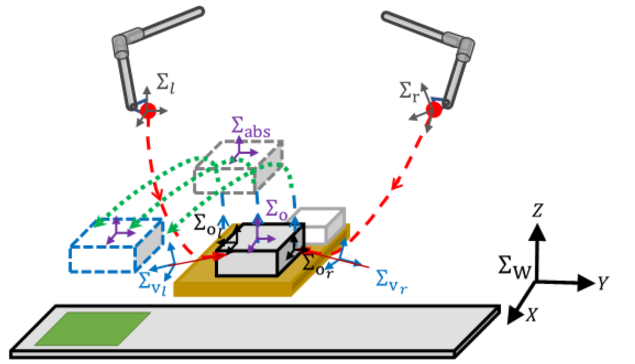


Figure 2. Schematics illustration of the considered dual-arm task, which can be seen as a succession of two main phases: free motion phase when reaching (red dashed line), and constrained motion phase when in contact and executing the placing or tossing motion (blue and green dashed lines)). Σ_{w} and Σ_{o} are the world and the object frame. Σ_l and Σ_r denote respectively the end-effector frames of the left and right robot, while Σ_{o_l} and Σ_{o_r} denote respectively their desired grasping configuration on the object side. Σ_i and Σ_{o_i} are respectively the i^{th}end-effector's and object's grasp configuration frames, while Σ_{v_i} denotes the i^{th} frame of a virtual or auxiliary attractor that shapes the trajectory for impact, with index ($i \equiv$ left, right).

or release position. Therefore, for a dual-arm system that requires coordination, to realize fast grabbing and afterward a tossing task, we formulate at the position level[1]the following modulated dynamical system (MDS).

$$\dot{\mathbf{x}} = M(\mathbf{x})f_n(\mathbf{x}) + f_g(\mathbf{x}) \tag{1}$$

where $\mathbf{x} = \begin{bmatrix} \mathbf{x}^L \\ \mathbf{x}^R \end{bmatrix} \in \mathbb{R}^6$ is the state vector of the DS with $\mathbf{x}^L$ and $\mathbf{x}^R$ representing respectively the position of the left and right robot of the dual-arm system. $f_n(\mathbf{x}) \in \mathbb{R}^6$ is the nominal DS that generates the coordinated motion towards transitory attractors located in the vicinity of the desired positions. $f_g(\mathbf{x})$ represents the equivalent grasping force in the motion space, whereas $M(\mathbf{x}) \in \mathbb{R}^{6\times 6}$ is the state-dependent modulation matrix that shapes locally the motion generated by $f_n(\mathbf{x})$. It is defined as

$$M(\mathbf{x}) = E(\mathbf{x})\Lambda(\mathbf{x})E^{\top}(\mathbf{x}) \in \mathbb{R}^{6\times 6} \tag{2}$$

where $E(\mathbf{x}) \in \mathbb{R}^{6\times 6}$ and $\Lambda(\mathbf{x}) \in \mathbb{R}^{6\times 6}$ are block-diagonal matrices respectively of state-dependent orthonormal basis and gains for the left and right robotic arm. They are respectively defined as $E(\mathbf{x}) = diag\{E^L(\mathbf{x}), E^R(\mathbf{x})\}$ and $\Lambda(\mathbf{x}) = diag\{\Lambda^L(\mathbf{x}), \Lambda^R(\mathbf{x})\}$. In $E(\mathbf{x})$, each basis $E^h(\mathbf{x}) = \begin{bmatrix} \mathrm{e}_1^h & \mathrm{e}_2^h & \mathrm{e}_3^h \end{bmatrix} \in \mathbb{R}^{3\times 3}$ with $h = \{L, R\}$ is designed such that its first vector e_1^h is aligned with the intended impact direction at contact. [2] That is $\mathrm{e}_1^h = \frac{\dot{\mathbf{x}}_d^h}{\|\dot{\mathbf{x}}_d^h\|} = \frac{\mathbf{x}_d^h - \mathbf{x}_t^h}{\|\mathbf{x}_d^h - \mathbf{x}_t^h\|}$, where $[\mathbf{x}_d^h \, \dot{\mathbf{x}}_d^h]$ denotes the desired impact or tossing state of the h^{th} robot, and $\mathbf{x}_t^h$ is the transitory attractor defined at a distance ρ of $\mathbf{x}_d^h$, such that $\mathbf{x}_t^h = \mathbf{x}_d^h + E^h(\mathbf{x})\begin{bmatrix} -\rho & 0 & 0 \end{bmatrix}^{\top}$. Thus, as illustrated in Figure 3 for the case of grabbing with impact, each robot is driven first towards a transitory attractor $\mathbf{x}_t^h$

[1]The control of orientation is described in appendix A

[2]The direction of impact is not limited to be normal to the contact surface, but can also have other orientations.

before being moved with the appropriate orientation along e_1^h towards $\mathbf{x}_d^h$, the real attractor.

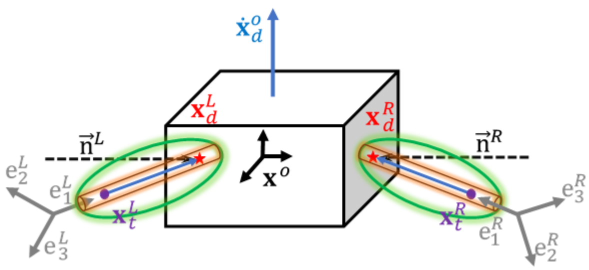


Figure 3. Geometric representation of orthonormal basis $E^L(x)$, $E^R(x)$ and illustration of modulation region (green ellipsoid) within which the dark-red cylindrical region represents the activation of the normal distance to the vector $\mathbf{x}_d^h - \mathbf{x}_t^h$

In $\Lambda(\mathbf{x})$, each robot's gain matrix $\Lambda^h(\mathbf{x}) \in \mathbb{R}^{3\times3}$ has entries $\bar{\lambda}_{ij}^h(\mathbf{x})$ defined as

$$\bar{\lambda}_{ij}^h(\mathbf{x}) = \begin{cases} \alpha(\mathbf{x})\lambda_{ij}^h(\mathbf{x}) + (1-\alpha(\mathbf{x})) & \text{if } i = j \\ \alpha(\mathbf{x})\lambda_{ij}^h(\mathbf{x}) & \text{if } i \neq j \end{cases} \tag{3}$$

where $\lambda_{ij}^h(\mathbf{x}) \in \mathbb{R}^1$ represent state-dependent scalar terms defined in section III-A, $\alpha(\mathbf{x}) \in [0,1]$ activates the modulation when the robots are in the vicinity of their desired attractors.

The modulation is active in a region defined by an ellipsoid along the vector $\mathbf{x}_d^h - \mathbf{x}_t^h$ as illustrated in Figure 3. To characterize the modulation, we define three activation parameters namely: δ_{radial}, δ_{normal} and $\delta_{tangent}$ which represent distances with their origin at $\mathbf{x}_t^h$ in the basis $E^h(\mathbf{x})$. These distances are measured respectively in $3D$, in $2D$ normal to e_1^h, and in $1D$ along e_1^h. Accordingly, we define associated activation functions $\phi_i(\mathbf{x}^h) \in [0,1]$ with *i*={*radial, normal, tangent*} as $\phi_i(\mathbf{x}^h) = \frac{1}{1+e^{-a_i(\delta_i - \Gamma_i(\mathbf{x}^h))}}$ where $\Gamma_i^h(\mathbf{x}^h)$ represents state-dependent distances of the robot given by $\Gamma_i^h(\mathbf{x}^h) = [(\mathbf{x}^h - \mathbf{x}_t^h)^\top E^h(\mathbf{x}^h)\Sigma_i(E^h(\mathbf{x}^h))^\top(\mathbf{x}^h - \mathbf{x}_t^h)]^{\frac{1}{2}}$. $\Sigma_i \in \mathbb{R}^{3\times3}$ are diagonal matrices that select the considered directions of $E^h(\mathbf{x})$. The elements of Σ_i are mainly 0, but 1 at index(es) of the desired direction(s). Hence, $\alpha(\mathbf{x})$ in (3), is designed such that $\alpha(\mathbf{x}) = \frac{1}{2}\sum_{h=1}^2 \phi_{radial}(\mathbf{x}^h)$.

The behavior of the DS as it generates the desired motion is shown in Figure 4 along with the activation of the modulation functions $\phi_i(\mathbf{x})$.

III. Modulation-based Coordinated Control

The motion coordination of the dual-arm system, in this work, exploits the cooperative task space representation [14], which relates the states of each robot to the cooperative coordinates formed by the absolute and relative states of the dual-arm system. The coordination is achieved by controlling the two robots' cooperative coordinates and mapping the resulting motion to each robot.

Thus, assuming that the nominal DS $f_n(\mathbf{x})$ is linear, the coordinated motion that it generates can be written as:

$$f_n(\mathbf{x}) = \dot{\mathbf{x}} = \underbrace{T_b^{-1}AT_b}_{A'}(\mathbf{x} - \mathbf{x}_*) \tag{4}$$

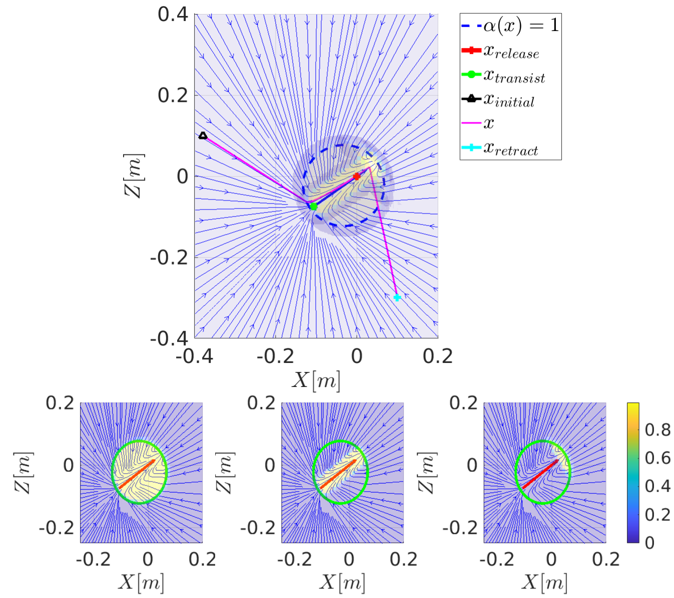


Figure 4. Example of the flow of generated motion outside and within the modulated region (thick dotted blue line), where the motion is shaped to pass through the desired release position (red dot). The activation of the modulation terms $\phi_{radial}(\mathbf{x})$, $\phi_{normal}(\mathbf{x})$, and $\phi_{tangent}(\mathbf{x})$ are shown respectively on the three bottom sub-figures

where $T_b = \begin{bmatrix} \frac{1}{2}\mathbf{I}_3 & \frac{1}{2}\mathbf{I}_3 \\ -\mathbf{I}_3 & \mathbf{I}_3 \end{bmatrix} \in \mathbb{R}^{6\times6}$ is a matrix that maps the two robot positions ($\mathbf{x}^L$ and $\mathbf{x}^R$) to the absolute position $\mathbf{x}^{abs} \in \mathbb{R}^3$ and relative position $\mathbf{x}^{rel} \in \mathbb{R}^3$ of the dual-arm system, such that $\begin{bmatrix} \mathbf{x}^{abs} \\ \mathbf{x}^{rel} \end{bmatrix} = T_b \begin{bmatrix} \mathbf{x}^L \\ \mathbf{x}^R \end{bmatrix}$ and where $\mathbf{I}_3$ is a 3×3 unit matrix. In (4), $A \in \mathbb{R}^{6\times6}$ denotes the dynamics or gain matrix, which is negative definite ($A < 0$) to ensure stability and convergence to a given attractor $\mathbf{x}_*$.

The coordination is thus achieved by controlling the dynamics of $\mathbf{x}^{abs}$ and $\mathbf{x}^{rel}$, which amounts to control respectively the two robots joint motion and their relative displacement and thereby their synchronization. The resulting motion of $\mathbf{x}^{abs}$ and $\mathbf{x}^{rel}$ is then mapped through T_b^{-1} to each robot's motion.

The modulation shapes the behavior of the nominal DS in the region where it is active. This shaping must preserve the DS stability and convergence properties of the nominal dynamical system $f_n(\mathbf{x})$ towards its equilibrium points $\mathbf{x}_*$.

A. Stability and convergence to attractors

Based on the previous work [17] on single robot, we can state the following proposition for dual-arm system:

1) Proposition 1: For any given state $\{\mathbf{x} \in \mathbb{R}^3 | \, \alpha(\mathbf{x}) = 1, f_n^h(\mathbf{x}) \neq 0\}$ setting the state dependent coefficients of the modulation matrix $\Lambda^h(\mathbf{x})$ as

$$\lambda_{ij}^h(\mathbf{x}) = (e_i^h)^\top f_{m_i}^h(\mathbf{x}) \frac{f_n^h(\mathbf{x})^\top e_j^h}{f_n^h(\mathbf{x})^\top f_n^h(\mathbf{x})}, \tag{5}$$

the motion generated by (1) will be governed by the dynamical system $f_{m_i}^h(\mathbf{x})$. Moreover, if $f_{m_i}^h(\mathbf{x})$ is a stable linear or linear parameter varying (LPV) DS, for instance, of the

form of (4), the state $\mathbf{x}$ will asymptotically reach its attractor $\mathbf{x}_*$ while maintaining the coordination between robots of the dual-arm system. That is $\lim_{t\to\infty} \|\mathbf{x}-\mathbf{x}_*\| = 0$.
Proof: See Appendix B.

B. Generation of impact velocity

To generate desired grabbing impact velocities with the dual-arm system, we introduce the following proposition:

1) Proposition 2: For any given state $\{\mathbf{x} \in \mathbb{R}^3 |, f_r^h(\mathbf{x}) \neq 0\}$ setting the attractor $\mathbf{x}_*$ as follows

$$\mathbf{x}_* = \begin{cases} (1-\gamma(\mathbf{x}))\mathbf{x}_t + \gamma(\mathbf{x})(\mathbf{x} - A'^{-1}\dot{\mathbf{x}}_d) & \text{if } \mathrm{e}^h_{i=1} \\ \mathbf{x}_t & \text{if } \mathrm{e}^h_{i\neq 1} \end{cases} \tag{6}$$

with $\gamma(\mathbf{x}) \triangleq \phi_{radial}(\mathbf{x})\phi_{normal}(\mathbf{x})$, then for any state, the modulated DS (1) with (4) will generate motion first towards $\mathbf{x}_t$ and then, when $\gamma(\mathbf{x}) = 1$, it will generate desired velocity $\dot{\mathbf{x}}_d = \begin{bmatrix} \dot{\mathbf{x}}_d^L \\ \dot{\mathbf{x}}_d^R \end{bmatrix} \in \mathbb{R}^6$ along the vectors e_1^L and e_1^R while maintaining attraction towards them in their respective orthogonal directions $((\mathrm{e}_2^L, \mathrm{e}_3^L)$ and $(\mathrm{e}_2^R, \mathrm{e}_3^R))$.
Proof: See Appendix C.

C. Manipulation task

To generate desired motion of an object carried by a dual-arm system using the proposed DS (1), we introduce the following proposition:

1) Proposition 3: For an object grasped by a dual-arm system and whose position is given by $\mathbf{x}^o$ and desired attractor is given $\mathbf{x}_d^o$, for any state $\{\mathbf{x} \in \mathbb{R}^3 | \, \kappa(\mathbf{x}) = 1, f_r^h(\mathbf{x}) \neq 0\}$ setting the attractor $\mathbf{x}_*$ of the proposed modulated DS (1) as

$$\mathbf{x}_* = \kappa(\mathbf{x}) T_b^{-1} \begin{bmatrix} \mathbf{x}_d^o + (\mathbf{x}^{abs} - \mathbf{x}^o) \\ \mathbf{x}_o^R - \mathbf{x}_o^L \end{bmatrix} \tag{7}$$

where $(\mathbf{x}^{abs} - \mathbf{x}^o)$ denotes the offset between the end-effector absolute position and the object's origin, and where $\mathbf{x}_o^L$ and $\mathbf{x}_o^R$ denote respectively the positions of the grasping points of left and right robot, the proposed modulated DS will generate coordinated robots motion that will make the grasped object's position to asymptotically converge to $\mathbf{x}_d^o$.

The scalar function $\kappa(\mathbf{x}) \in [0, 1]$ indicates the contact and it is defined as $\kappa(\mathbf{x}) \triangleq \gamma(\mathbf{x})\phi_{tangent}(\mathbf{x})$.
Proof: See Appendix D.

D. Tossing task

A tossing task is a form of manipulation task where the grabbed object needs to be released at a desired position simultaneously with a desired velocity. Thus, to perform a tossing task of an object with a dual-arm system using the DS (1), we propose to generate the corresponding desired coordinated motion by setting the desired absolute and relative velocities ($\dot{\mathbf{x}}_d^{abs}$ and $\dot{\mathbf{x}}_d^{rel}$) as follows

$$\dot{\mathbf{x}}_d^{abs} = M^o(\mathbf{x}^o) f_o(\mathbf{x}^o) \, \text{and} \, \dot{\mathbf{x}}_d^{rel} = -[\mathbf{x}_o^R - \mathbf{x}_o^L]_\times \boldsymbol{\omega}^o \tag{8}$$

where similarly to (1), $f_o(\mathbf{x}^o) \in \mathbb{R}^3$ denotes the nominal DS of the object and $M^o(\mathbf{x}^o) = E^o(\mathbf{x}^o)\Lambda^o(\mathbf{x}^o)(E^o(\mathbf{x}^o))^\top \in \mathbb{R}^{3\times 3}$ is the associated modulation matrix with $E^o(\mathbf{x}) \in \mathbb{R}^{3\times 3}$ an orthonormal basis. To meet the tossing motion constraints, $E^o(\mathbf{x}^o)$ is defined with its origin at the desired release position $\mathbf{x}_r^o$ and with its first vector e_1^o aligned with the desired release velocity ($\mathrm{e}_1^o = \frac{\dot{\mathbf{x}}_r^o}{\|\dot{\mathbf{x}}_r^o\|}$). $\boldsymbol{\omega}^o$ is the angular velocity of the object. In pure translation ($\boldsymbol{\omega}^o = 0$), $\dot{\mathbf{x}}_d^{rel}$ is set to 0.

Now, the convergence to the release position with the desired release velocity is a corollary of *Proposition 1* and *Proposition 2* applied to the object state, and with

$$\mathbf{x}_*^o = \begin{cases} (1-\gamma(\mathbf{x}^o))\mathbf{x}_t^o + \gamma(\mathbf{x}^o)(\mathbf{x}^o - A'^{-1}\dot{\mathbf{x}}_r^o) & \text{if } \mathrm{e}^o_{i=1} \\ \mathbf{x}_t^o & \text{if } \mathrm{e}^o_{i\neq 1} \end{cases} \tag{9}$$

where $\dot{\mathbf{x}}_r^o$ denotes the desired release velocity and $\mathbf{x}_t^o \in \mathbb{R}^3$ and $\gamma(\mathbf{x}^o)$ are defined as in (6), but this time for the object.

E. Contact force generation

When the end-effectors of the robots have reached the object, the grasp quality depends on the induced internal wrenches. The contacts being unilateral with limited friction, insufficient or excessive wrenches might result in contact slippage or damage of the object and/or end-effectors. Under the assumption that the object's mass and the friction coefficient (at least its minimum value) are known, we propose to generate the contact wrenches, $\mathbf{f}_\mathrm{e}^d$, satisfying the desired task and the contact constraints through the following optimization

$$\begin{aligned} \mathbf{f}_\mathrm{e}^d = \operatorname{argmin} \, & \|\mathbf{G}_\mathrm{o}\mathbf{f}_\mathrm{e} - \mathbf{f}_\mathrm{o}^d\|^2_{\mathrm{w_o}} + \|\mathbf{f}_\mathrm{e} - \mathbf{f}_\mathrm{e}^*\|^2_{\mathrm{w_e}} \\ s.t. \quad & \mathbf{C}_\mathrm{f}\mathbf{f}_\mathrm{e} \leq \mathbf{d}_\mathrm{f} \\ & \mathbf{Q}_\mathrm{eo}\mathbf{f}_\mathrm{e} = \mathbf{0} \end{aligned} \tag{10}$$

where $\mathbf{f}_\mathrm{o}^d \in \mathbb{R}^6$ is the object's effective wrench required to perform the desired manipulation task. $\mathrm{w_o}$ and $\mathrm{w_e}$ are weight matrices. The constraint $\mathbf{C}_\mathrm{f}\mathbf{f}_\mathrm{e} \leq \mathbf{d}_\mathrm{f}$ encapsulates the friction cone and center of pressure constraints, whereas the constraints $\mathbf{Q}_\mathrm{eo}\mathbf{f}_\mathrm{e} = \mathbf{0}$ represents the complementary condition between the force components normal to the contact and the normal distance to the contact.

To find the object's effective wrench $\mathbf{f}_\mathrm{o}^d$ required in (10), if, for instance, the desired task is encoded as $\dot{\mathbf{x}}_\mathrm{o}^d = \boldsymbol{f}(\mathbf{x}_\mathrm{o})$, one can use a passive DS approach [18] to compute as [3] $\mathbf{f}_\mathrm{o}^d = -\mathbf{D}_\mathrm{o}(\mathbf{x}_\mathrm{o})(\dot{\mathbf{x}}_\mathrm{o} - \boldsymbol{f}(\mathbf{x}_\mathrm{o})) + \mathbf{b}_\mathrm{o}$, where $\mathbf{D}_\mathrm{o}(\mathbf{x}_\mathrm{o}) \in \mathbb{R}^{6\times 6}$ denotes a state varying damping matrix and $\mathbf{b}_\mathrm{o}$ is the object's Coriolis and gravity forces.

Optimization (10) is based on an object-centered approach and performs nothing but a distribution of $\mathbf{f}_\mathrm{o}^d$ between the two robot's hands. It seeks a distribution that will yield minimum internal wrenches and satisfy the contact constraints.

[3] Alternatively, if the desired acceleration of the object $\ddot{\mathbf{x}}_\mathrm{o}^d$ is available, $\mathbf{f}_\mathrm{o}^d$ can also be obtained from the inverse dynamics of the object as $\mathbf{f}_\mathrm{o}^d = \mathbf{M}_\mathrm{o}\ddot{\mathbf{x}}_\mathrm{o}^d + \mathbf{b}_\mathrm{o} + \mathbf{f}_\mathrm{env}$

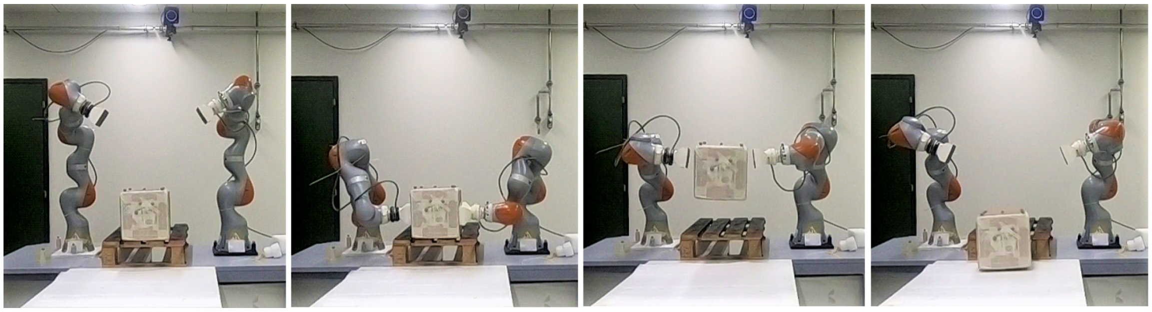


Figure 5. Snapshots of fast dual arm grabbing with impact and holding an object. From left to right, the snapshots show the initial robots' configuration, the pose of the robots' arm at initial contact with the object, the dual-arm tossing the object with the desired tossing velocity and at the desired release location, and finally, the landing of the object.

IV. Empirical Validation

The validation of the proposed algorithm is carried out on a dual-arm robotic system consisting of two KUKA LBR robots, an IIWA7 and an IIWA14 spaced by 1m in the lateral direction. Each robot end-effector is mounted with a 3D printed grabber offering a contact surface of $(0.15 \times 0.1)\, m^2$. The robots are connected to a 3.4 GHz i7 PC, that runs the dual-arm controller. The desired task space velocity generated by the proposed law (1) is executed by a passivity inspired torque controller of the form [18].

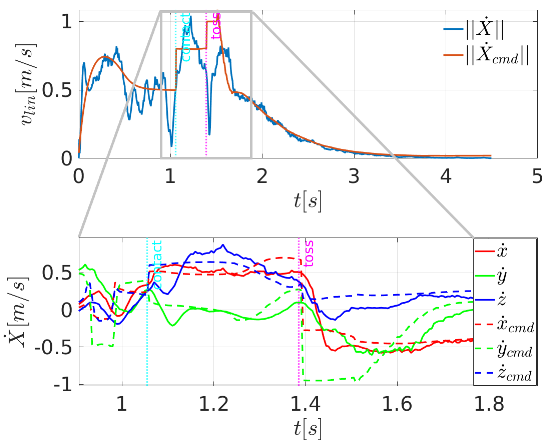
We extensively tested our algorithm both in Gazebo simulation and on the real robots, but we report primarily on results obtained on the real platforms, unless explicitly stated otherwise. To evaluate the performance of the proposed algorithm, we compare it to a coordination algorithm that contacts the object with near-zero velocity. We assess the controller's ability to properly coordinate the robots' motion to ensure that the two robot impact the object simultaneously. We further quantify the performace of the process by computing task duration and energy expenditure. Finally, we also show that such a tossing procedure expands the workspace of the robots and quantify the extension. A video[4] of the corresponding experiments is provided as supplementary material and the code[5] made available.

A. Fast object grabbing with impact

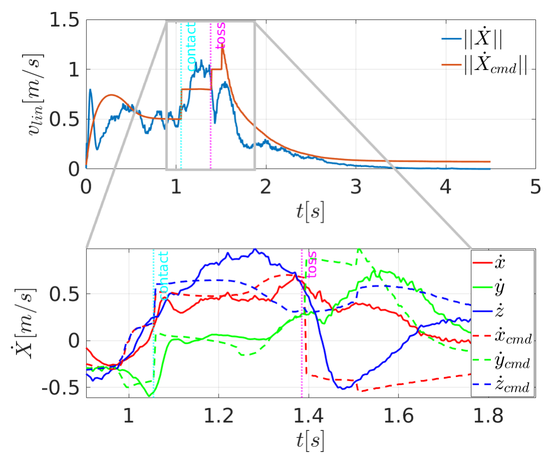
We evaluate here the motion coordination capability of our control scheme and its ability to execute fast grabbing with impact while generating contact wrenches that stabilize the grasp and perform the desired tossing motion of the object. The robotic task consists of reaching and swiftly grabbing an object of dimension $(0.2,\ 0.2\, 0.2)\, m$, mass $0.7\, kg$, initially located at $(0.41,\ 0.0\, 0.32)\, m$ and tossing it at $(0.7,\ 0.0\, 0.7)\, m$ with of velocity of $(0.8,\ 0.0\, 0.8)\, m$. Figure 5 provides some snapshots of the task execution. The velocities of the dual arm system throughout the task are shown in Figure 6, where for both the left and the right arm, the norm of the velocity appears on top and a close-up views of the velocities between the contact and the release of the object appear

[4]https://youtu.be/USXdkoH5t5g

[5]https://github.com/epfl-lasa/iam_dual_arm_control



(a)

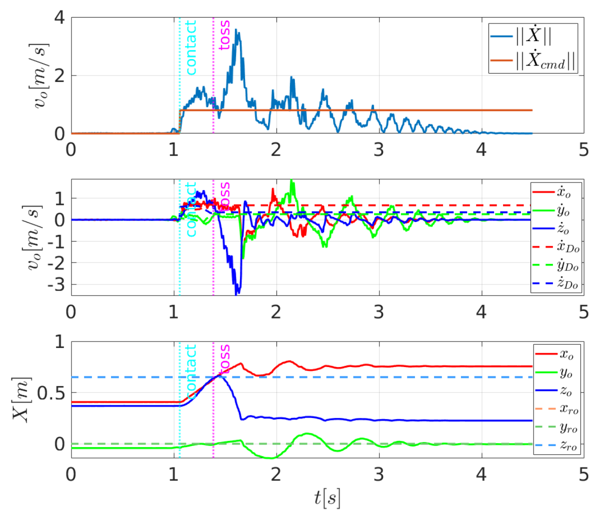


(b)

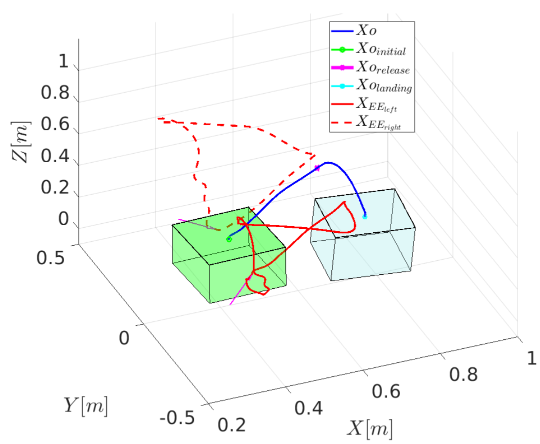
Figure 6. Velocities of dual-arm end-effectors during a fast grabbing and tossing of an object at $0.8\, m/s$. (a): left arm (KUKA IIWA7) and (b): right arm (KUKA IIWA14). For both sub-figures (a) and (b): top row shows the norm of the velocities, and the bottom row shows a close-up view of the robots' linear velocities in the time period running from before contact and after releasing of the object.

at the bottom. The DS generating these velocities has been "boosted" by converting the generated motion flow into a unitary vector field modulated in amplitude by reaching speed and tossing speed during the reaching phase and the tossing phase, respectively. In Figure 6, it can be observed that the end-effectors reach and impact the object around t=1.05s with a speed of 0.5 m/s. The impact induces a drop of the real velocities despite the increase of the desired end-effector velocities. A closer look at the bottom sub-figure indicates that it is meanly the y components (normal to the contact) that is dropping at the contact, whereas the x and z components slightly dropped while continuing their pre-impact trends in the direction of the desired motion of the object. During the tossing phase, which starts from contact to release, the DS generates for the object a velocity with the desired tossing speed which is distributed between the arms.

The object's desired and real motion over time are shown in sub-figure 7a, whereas sub-figure 7b shows the 3D trajectories of the object and those of the dual-arm system for a task cycle.



(a)



(b)

Figure 7. (a). Motion of the object during the fast grabbing and tossing task. top: norm of the linear velocity; middle: linear velocity; and bottom: position. The grabbing and lifting phases start at the contact indicated by a cyan vertical line, whereas the release instant is indicated by the magenta vertical line. (b). 3D trajectories of the overall dual arm and object system during the fast grabbing and tossing task. The red continuous and dotted lines are for the left and right end-effector, respectively, while the blue line is for the object. The object at its initial and final location is depicted by the light green and blue box, respectively. The magenta arrows indicate the direction of the impact velocity when the contact is established.

In sub-figure 7a, one can see from top to bottom, the time evolution of the velocity norm, the linear velocity components and the position of the object. It can be observed that the object carried by the two end-effectors follows its desired motion, although with tracking error, until it reaches the desired release location within a tolerance of 0.03m. Then, the object is suddenly released and its motion is now governed by the projectile dynamics. As the object is falling under the gravity field, its vertical velocity component (blue continuous lines on the middle plot) can be seen to linearly decrease while its position (on the bottom plot) decreases in a quadratic-like manner. How well, under the proposed scheme, the real position of the object denoted by (x_o, y_o, z_o) satisfies its desired position denoted by (x_{ro}, y_{ro}, z_{ro}) at the release time can be seen on the bottom plot of sub-figure 7a.

In sub-figure 7b, the 3D trajectory of the object is shown in solid blue from its initial position represented by the green box to its landing location represented by the light blue box. The end-effector trajectories represented in red described loops that start at the initial positions, go to the grasping points of the object, lift the latter towards the release position and go back to their initial positions. The magenta arrows represent the directions of the impact velocities of the end-effectors when grabbing the object; it can be noticed that they are aligned with the desired object motion.

A set of 50 trajectories of the tossed object with their landing positions along with the reachable space of the end-effectors is shown in Figure 8. Thus, with the range of achieved velocities, the "tossing reachable space" extends the dual-arm system reachable workspace by at least 20%.

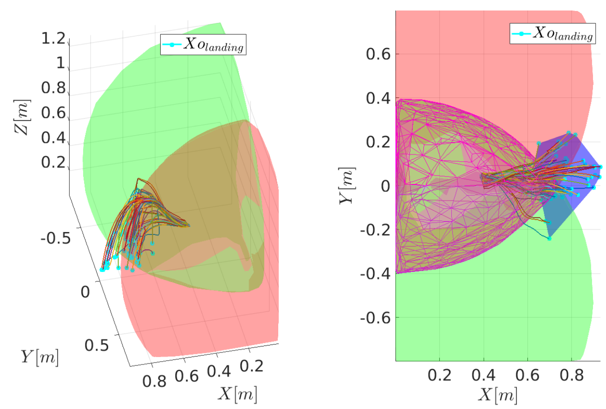
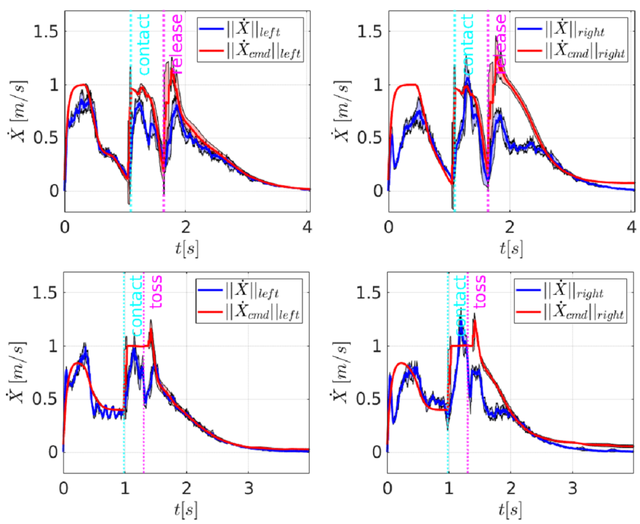


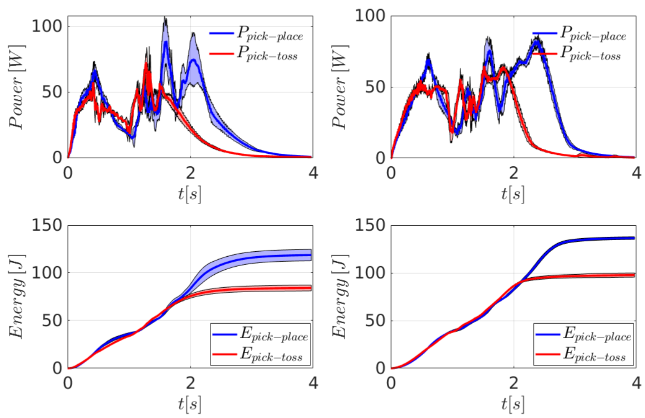
Figure 8. Extension of the dual-arm system's reachable space with the tossing reachable space. (left): set of 50 3D trajectories of the tossed object with respect the dual-arm workspace (the green and red areas depict the left and right arms space of reachable positions). (right): top view of the tossing-based extension (blue area with the landing positions in cyan) of the joint reachable space depicted with the meshed region in magenta.

B. *Pick-and-place vs. proposed pick-and-toss*

The goal of this experiment is to evaluate the benefits obtained, in terms of task duration (cycle time) and energy expenditure, in a depalletizing task when using the usual pick-and-place operation with near-zero contact and release velocities, and when using the proposed fast grabbing with impact and tossing. As previously, the task of the dual-arm system consists of grabbing an object from a pallet and moving it to a table located in front of the robots. The velocities of the robots resulting from the two approaches are shown respectively in Figure 9a (top) and (bottom), whereas the associated power and energy expenditure of both the left and right arms are shown in Figure 9b for the classical (top), and the proposed approach (bottom), respectively.



(a)



(b)

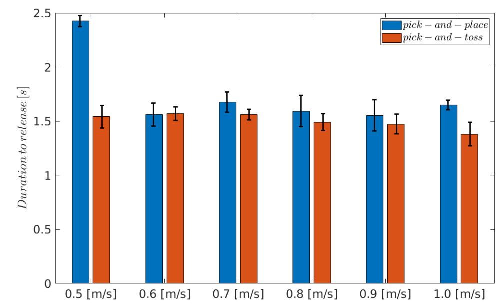
Figure 9. (a) Time evolution of mean and standard deviation of measured and commanded velocity norms of the left and right arm for five pick-and-placing (top), and five pick-and-tossing (bottom) experiments at 1.0 m/s. (b) Time evolution of estimated energy consumption of the left and right arm for five pick-and-placing (blue), and five pick-and-tossing (red) experiments at 1.0 m/s.

In the pick-and-place case, the robots contact the object with a velocity of 0.11 m/s mainly in the normal direction to the contact surfaces. The object is then moved with a velocity up to 1.0 m/s and later released with a velocity of about 0.13 m/s at $t \approx 1.64$ m/s (see Figure 9a (top)). In the pick-and-toss case, like previously, the robots impact the object with an average speed of 0.40 m/s. The object is then moved throughout the task with a desired velocity of 1.0 m/s, but tossed instead of being gently placed (see Figure 9a (bottom)).

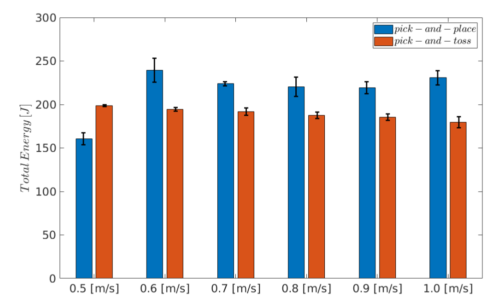
Thus, by reducing the deceleration phase at pickup and release time, the task duration in the proposed approach for tossing velocity of 1.0 m/s is around 19% shorter than in the classical pick-and-place operation. Regarding the energy expenditure for the considered velocity, it can be observed in Figure 9b that the proposed approach consumed 24% less than the classical pick-and-place.

The difference in the observed variances between the two robots is mainly due to their different dynamic characteristics. The left arm corresponds to the KUKA LBR IIWA7 robot, which has smaller inertia properties and is less damped than the right arm (KUKA LBR IIWA14 robot), and therefore it can accelerate much faster for the same applied torque.

For velocities ranging from 0.5 m/s to 1.0 m/s, the comparison of the average task duration and the overall energy expenditure of the two approaches is summarized by the bar plots shown in Figure 10a and Figure 10b, respectively.



(a)



(b)

Figure 10. (a) Comparison for velocities ranging between 0.5 and 1.0 m/s of the task duration when the pick-up and placing happened at near-zero velocity (blue) and when the dual arm system leverage impact at the pickup and toss the object (red). (b) Comparison of overall energy expenditure of the dual arm system for tossing velocities ranging between 0.5 and $1.0 m/s$ when grabbing the object with near-zero velocity (blue) and when performing fast object grabbing in a swipe (red).

V. Conclusion

This paper proposed, for a dual-arm robotic system, a modulated DS-based unified motion generation algorithm that allows swift grabbing and release of an object. The desired states at contact and release are achieved through a local shaping of the robots movement. The proposed motion generation algorithm, whose stability and convergence towards the desired states was theoretically proved, has been experimentally validated in simulation and on real robots. The results confirmed that the proposed approach, besides motion coordination, enables to generate, for the dual-arm system, desired impact and tossing motions. The obtained results also suggest that grabbing with impact and tossing, especially when the impact direction anticipates the upcoming motion of the object, lead to shorter and more energy efficient pick and place tasks.

However, the application realm of such a dynamic pick and release extends well beyond the experiments conducted in this work. Bimanual grabbing of an object from the sides is more suitable than picking from the top with suction cups mechanisms in many situations, such as picking up filled trays with open lid, lifting cases with too fragile cover for supporting the case's weight, or when placing boxes in shelves with limited vertical space. Also, it is worth noting that the proposed pick-and-toss scheme should be used for objects that may not risk breaking under the impact or when such damage is not important. This is important as the robot by itself cannot forecast the effects of impact on tossed objects. Tossing of boxes filled with open bottles, as shown the supplementary material, may be applicable when recycling or dispatching used bottles.

The system we described here can be integrated easily in current industrial settings using existing industrial manipulators. Our synchronized control law generates a reference trajectory that can be embedded in reference trajectories for an industrial controller. Additionally, nothing in our control system prevents a user to equip the robots with suction cups, instead of the flat grippers, used in our experiments. This would allow the robots to lift heavier weights. Moreover, the synchronization of the two robots could be easily disabled (by modifying Eq. 4 and setting A' to be block diagonal, and $\mathbf{x}_*$ to specify one attractor for each arm). This would allow each robot to perform individual tasks, when appropriate; and to switch back to coordinated control, when the object that needs to be grabbed requires a bimanual grasp. Such an approach would then increase the flexibility of current industrial manipulators.

Nevertheless, this work in its implementation has limitations. The Passive DS controller [18] used to compute the joint torque from the generated motion has shown limited tracking abilities, particularly during the tossing. Moreover, we used first-order linear DS, which works well for fixed or slowly varying attractors, but have no compensation ability in tracking and therefore defer this burden to the low-level controller (here the Passive DS). Future work could use second-order DS able to address the compensation problem in tracking as in [19]. In addition, instead of the Passive DS, one could use an inverse dynamic controller that not only compensates for gravity, but also for the inertial torques produced by strong accelerations during tossing. Finally, the impact and the object dynamics could also be included to guarantee hardware safety and improve task accuracy.

Acknowledgment

This work was supported in part by the European Union's Horizon 2020 research and innovation program under grant agreement No. 871899. I.AM. project: https://i-am-project.eu/.

The authors would like to thank Patrick Sgro for his support in the implementation on the real-robots.

References

[1] C. Smith, Y. Karayiannidis, L. Nalpantidis, X. Gratal, P. Qi, D. V. Dimarogonas, and D. Kragic, "Dual arm manipulation a survey," *Robotics and Autonomous systems*, vol. 60, no. 10, pp. 1340–1353, 2012.

[2] E. Nakano, S. Ozaki, T. Ishida, and I. Kato, "Cooperational control of the anthropomorphous manipulator'melarm'," in *Proc. of 4th International Symposium on Industrial Robots*, vol. 251, 1974, p. 260.

[3] F. Caccavale and M. Uchiyama, "Cooperative manipulators," in *Springer handbook of robotics*. Springer, 2008, pp. 701–718.

[4] M. Uchiyama and P. Dauchez, "Symmetric kinematic formulation and non-master/slave coordinated control of two-arm robots," *Advanced Robotics*, vol. 7, no. 4, pp. 361–383, 1992.

[5] S. A. Schneider and R. H. Cannon, "Object impedance control for cooperative manipulation: Theory and experimental results," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 383–394, 1992.

[6] N. Xi, T.-J. Tarn, and A. K. Bejczy, "Intelligent planning and control for multirobot coordination: An event-based approach," *IEEE transactions on robotics and automation*, vol. 12, no. 3, pp. 439–452, 1996.

[7] R. Bonitz and T. C. Hsia, "Internal force-based impedance control for cooperating manipulators," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 78–89, 1996.

[8] F. Caccavale, P. Chiacchio, A. Marino, and L. Villani, "Six-dof impedance control of dual-arm cooperative manipulators," *IEEE/ASME Transactions On Mechatronics*, vol. 13, no. 5, pp. 576–586, 2008.

[9] N. Dehio, J. Smith, D. L. Wigand, P. Mohammadi, M. Mistry, and J. J. Steil, "Enabling impedance-based physical human–multi–robot collaboration: Experiments with four torque-controlled manipulators," *The International Journal of Robotics Research*, vol. 41, no. 1, pp. 68–84, 2022. [Online]. Available: https://doi.org/10.1177/02783649211053650

[10] K. Bouyarmane, J. Vaillant, K. Chappellet, and A. Kheddar, "Multi-robot and task-space force control with quadratic programming," 2017.

[11] S. S. Mirrazavi Salehian, N. Figueroa, and A. Billard, "Dynamical system-based motion planning for multi-arm systems: Reaching for moving objects," In Proceedings of International Joint Conference on Artificial Intelligence 2017, Melbourne, Australia, 2017.

[12] W. Amanhoud, M. Khoramshahi, and A. Billard, "A dynamical system approach to motion and force generation in contact tasks," in *Proceedings of Robotics: Science and Systems*, FreiburgimBreisgau, Germany, June 2019.

[13] Y. Wang, N. Dehio, A. Tanguy, and A. Kheddar, "Impact-Aware Task-Space Quadratic-Programming Control," Nov. 2020, working paper or preprint. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02741682

[14] H. A. Park and C. G. Lee, "Extended cooperative task space for manipulation tasks of humanoid robots," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 6088–6093.

[15] M. Rijnen, A. Saccon, and H. Nijmeijer, "Reference spreading: Tracking performance for impact trajectories of a 1dof setup," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 3, pp. 1124–1131, 2019.

[16] N. Dehio and A. Kheddar, "Robot-Safe Impacts with Soft Contacts Based on Learned Deformations," in *ICRA*, Xi'an, China, May 2021. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02973947

[17] S. S. Mirrazavi Salehian and A. Billard, "Stable transitions from free-space to contact: A dynamical system based approach," In Proceedings of the Workshop on Towards Robots that Exhibit Manipulation Intelligence, 2018.

[18] K. Kronander and A. Billard, "Passive interaction control with dynamical systems," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 106–113, 2016.

[19] S. S. M. Salehian, M. Khoramshahi, and A. Billard, "A dynamical system approach for softly catching a flying object: Theory and experiment," *IEEE Transactions on Robotics*, vol. 32, no. 2, pp. 462–471, 2016.

[20] E. Malis, F. Chaumette, and S. Boudet, "2 1/2 d visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 238–250, April 1999.

Appendix

A. *Orientation control*

To control the orientation task, which consists of driving the current orientation of the h^{th}end-effector represented by the rotation matrix $\mathbf{R}_c^h \in \mathbb{R}^{3x3}$ towards its desired value $\mathbf{R}_d^h \in \mathbb{R}^{3x3}$, we define a state vector $\xi_\theta^h \in \mathbb{R}^3$ using the axis/angle representation of the relative orientation, ${}^d\mathbf{R}_c^h \triangleq (\mathbf{R}_d^h)^\top \mathbf{R}_c^h$. Hence, $\xi_\theta^h \triangleq \theta\boldsymbol{\mu}({}^d\mathbf{R}_c^h)$, where $\boldsymbol{\mu} \in \mathbb{R}^3$ and $\theta \in \mathbb{R}$ represent respectively the axis and the angle associated with the rotation matrix ${}^d\mathbf{R}_c^h$.

With ξ_θ^h defined as above, its desired value is located at the origin, that is $\xi_{\theta^d}^h = 0$. Thus, similarly to the position task, if we assume a linear or linear parameters varying (LPV) DS for the orientation, we can write

$$\dot{\xi}_\theta^h = A_\theta(\xi_\theta^h - \xi_{\theta d}^h) = A_\theta \xi_\theta^h$$

where $A_\theta \in \mathbb{R}^{3\times3}$ is the dynamic matrix chosen to be negative definite ($A_\theta < 0$) to ensure asymptotic converges of ξ_θ^h towards its attractor $\mathbf{0}$ ($\lim_{t\to\infty} \xi_\theta^h = 0$). Such convergence indicates the matching $\mathbf{R}_c^h$ with $\mathbf{R}_d^h$. The angular velocity associated with the orientation DS is obtained as follows

$$\omega^h = \mathbf{L}_{\xi^h}^{-1} \dot{\xi}_\theta^h = \mathbf{L}_{\xi^h}^{-1} A_\theta \xi_\theta^h$$

where $\mathbf{L}_{\xi^h} \triangleq \mathbf{L}_{\theta\boldsymbol{\mu}}^h(\mathbf{R}_c^h)^\top$ with $\mathbf{L}_{\theta\boldsymbol{\mu}}^h \in \mathbb{R}^{3\times3}$ a matrix mapping the angular velocity to the time derivative of orientation state vector ξ^h and given by [20]

$$\mathbf{L}_{\theta\boldsymbol{\mu}}^h = \mathbf{I}_{3\times3} - \frac{\theta}{2}[\boldsymbol{\mu}^h]_\times + \left(1 - \frac{sinc\,\theta}{sinc^2\frac{\theta}{2}}\right)[\boldsymbol{\mu}^h]_\times^2$$

where $\theta sinc\,\theta = sin\theta$ and $[\boldsymbol{\mu}^h]_\times \in \mathbb{R}^{3\times3}$ denotes a skew-symmetric matrix associated with $\boldsymbol{\mu}^h$.

To coordinate the position and the orientation task, the latter was coupled to the position task using a state-depend coefficient $\eta(\mathbf{x})$ function of the error on the absolute position: $\eta(\mathbf{x}) = 1 - \exp\left(-\frac{\sigma}{\|\mathbf{x}^{abs} - \mathbf{x}_d^{abs}\| + \varepsilon}\right)$, where $\sigma > 0$ is a scalar that tunes how fast $\eta(\mathbf{x})$ varies within $[0, 1]$.

The orientation state vector ξ_θ^h is now computed as $\xi_\theta^h \triangleq \theta\boldsymbol{\mu}({}^*\mathbf{R}_c^h)$, with ${}^*\mathbf{R}_c^h \triangleq (\mathbf{R}_*^h(\eta))^\top \mathbf{R}_c^h$. Here $\mathbf{R}_*^h(\eta)$ denotes the rotation matrix computed from the spherical interpolation between a resting orientation $\mathbf{R}_r^h$ and the desired orientation $\mathbf{R}_d^h$ as function of $\eta(\mathbf{x})$. When $\eta(\mathbf{x}) \to 0$, $\mathbf{R}_*^h(\eta) \to \mathbf{R}_r^h$ and when $\eta(\mathbf{x}) \to 1$, $\mathbf{R}_*^h(\eta) \to \mathbf{R}_d^h$.

B. *Proof of Proposition 1*

Substituting (5) in (2) and then in (1) and multiplying by $diag\{(\mathrm{e}_i^L)^\top, (\mathrm{e}_i^R)^\top\}$ gives

$$\begin{array}{rcl} (\mathrm{e}_i^L)^\top \dot{\mathbf{x}}^L & = & (\mathrm{e}_i^L)^\top E^L(\mathbf{x})\Lambda^L(\mathbf{x})(E^L(\mathbf{x}))^\top f_r^L(\mathbf{x}) \\ (\mathrm{e}_i^R)^\top \dot{\mathbf{x}}^R & & (\mathrm{e}_i^R)^\top E^R(\mathbf{x})\Lambda^R(\mathbf{x})(E^R(\mathbf{x}))^\top f_r^R(\mathbf{x}) \end{array}$$

Hence, for each $h = \{L, R\}$ component, we have [6]

$$\begin{aligned} (\mathrm{e}_i^h)^\top \dot{\mathbf{x}}^h &= \sum_{j=1}^{3} \lambda_{ij}^h(\mathbf{x})(\mathrm{e}_j^h)^\top f_n^h(\mathbf{x}) \\ &= (\mathrm{e}_i^h)^\top f_{m_i}^h(\mathbf{x}) \underbrace{\sum_{j=1}^{3} \frac{f_n^h(\mathbf{x})^\top \mathrm{e}_j^h}{f_n^h(\mathbf{x})^\top f_n^h(\mathbf{x})}(\mathrm{e}_j^h)^\top f_n^h(\mathbf{x})}_{I} \\ &= (\mathrm{e}_i^h)^\top f_{m_i}^h(\mathbf{x}) \end{aligned} \tag{11}$$

Finally, substituting (4) for $f_{m_i}^h$ in (11) yields

$$(\mathrm{e}_i^h)^\top \dot{\mathbf{x}}^h = (\mathrm{e}_i^h)^\top A_h'(\mathbf{x} - \mathbf{x}_*) \tag{12}$$

which gives in terms of the left and right components

$$\begin{array}{rcl} (\mathrm{e}_i^L)^\top \dot{\mathbf{x}}^L & = & (\mathrm{e}_i^L)^\top [A_{LL}'(\mathbf{x}^L - \mathbf{x}_*^L) + A_{LR}'(\mathbf{x}^R - \mathbf{x}_*^R)] \\ (\mathrm{e}_i^R)^\top \dot{\mathbf{x}}^R & & (\mathrm{e}_i^R)^\top [A_{RL}'(\mathbf{x}^L - \mathbf{x}_*^L) + A_{RR}'(\mathbf{x}^R - \mathbf{x}_*^R)] \end{array} \tag{13}$$

Clearly, Eq. (13) shows the two robots interaction, which is necessary to preserve the coordination. At the same time, all robots converge towards their attractors since at the equilibrium ($\dot{\mathbf{x}}^L = 0$ and $\dot{\mathbf{x}}^R = 0$), we have

$$0 = \begin{bmatrix} A_{LL}' & A_{LR}' \\ A_{RL}' & A_{RR}' \end{bmatrix} \begin{pmatrix} \mathbf{x}^L - \mathbf{x}_*^L \\ \mathbf{x}^R - \mathbf{x}_*^R \end{pmatrix} = A'(\mathbf{x} - \mathbf{x}_*)$$

This implies that $\mathbf{x} - \mathbf{x}_* = 0$ given that A' is full rank ■

[6]where we use the following simplification $\sum_{j=1}^{3} \frac{f_n^h(\mathbf{x})^\top \mathrm{e}_j^h}{f_n^h(\mathbf{x})^\top f_n^h(\mathbf{x})}(\mathrm{e}_j^h)^\top f_n^h(\mathbf{x}) = \frac{f_n^h(\mathbf{x})^\top \sum_{j=1}^{3} \mathrm{e}_j^h(\mathrm{e}_j^h)^\top f_n^h(\mathbf{x})}{f_n^h(\mathbf{x})^\top f_n^h(\mathbf{x})} = I$

C. *Proof of proposition 2*

Proving the first motion towards $\mathbf{x}_t$ when $\gamma(\mathbf{x}) = 0$ is straightforward given (4) and similar expression for $f_{m_i}^h(\mathbf{x})$, since $\mathbf{x}_* = \mathbf{x}_t$.

However, when $\gamma(\mathbf{x}) = 1$ the attractor becomes $\mathbf{x}_* = \mathbf{x} - A'^{-1}\dot{\mathbf{x}}_d$ and when substituted in (12), the later becomes

$$(\mathrm{e}_1^h)^\top \dot{\mathbf{x}}^h = (\mathrm{e}_1^L)^\top \left[A_h'(A')^{-1}\begin{pmatrix} \dot{\mathbf{x}}_d^L \\ \dot{\mathbf{x}}_d^R \end{pmatrix}\right]$$

Given that $A'(A')^{-1} = \mathbf{I} \in \mathbb{R}^{6\times6}$ then

$$A_L'(A')^{-1} = \begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix} \text{ and } A_R'(A')^{-1} = \begin{bmatrix}\mathbf{0} & \mathbf{I}\end{bmatrix}$$

Therefore, the generated velocities along e_1^L and e_1^R for the two robots become

$$(\mathrm{e}_1^L)^\top \dot{\mathbf{x}}^L = (\mathrm{e}_1^L)^\top \dot{\mathbf{x}}_d^L \quad \text{and} \quad (\mathrm{e}_1^R)^\top \dot{\mathbf{x}}^R = (\mathrm{e}_1^R)^\top \dot{\mathbf{x}}_d^R$$

whereas for e_i^h with $i = 2$ and 3, the dynamics will remain $(\mathrm{e}_i^h)^\top \dot{\mathbf{x}}^h = (\mathrm{e}_i^h)^\top A_h'(\mathbf{x} - \mathbf{x}_t)$. ■

D. *Proof of Proposition 3*

Following the definition of T_b in section III, when the DS, as shown in *Proposition 1*, asymptotically converges to its attractor, we have $\mathbf{x} = \mathbf{x}_*$. When using $\mathbf{x}_*$ as defined in (7) we have

$$T_b^{-1}\begin{bmatrix} \mathbf{x}_d^{abs} \\ \mathbf{x}_d^{rel} \end{bmatrix} = T_b^{-1}\begin{bmatrix} \mathbf{x}^{abs} + (\mathbf{x}_d^o - \mathbf{x}^o) \\ \mathbf{x}_o^R - \mathbf{x}_o^L \end{bmatrix}$$

Rewriting and simplifying the previous equation yields

$$\begin{aligned} (\mathbf{x}_d^{abs} - \mathbf{x}_d^o) - (\mathbf{x}^{abs} - \mathbf{x}^o) &= 0 \\ \mathbf{x}_d^{rel} - (\mathbf{x}_o^R - \mathbf{x}_o^L) &= 0 \end{aligned}$$

The above expression is zero as it represents the difference between the offset($\mathbf{x}^{abs} - \mathbf{x}^o$) respectively the relative end-effectors position($\mathbf{x}_o^R - \mathbf{x}_o^L$) $= \mathbf{x}^{rel}$ and their values after convergence. For stable grasp these quantities remain constant throughout the task and therefore their difference is 0. ■