A Dynamical System Approach to Motion and Force Generation in Contact Tasks
Walid Amanhoud, Mahdi Khoramshahi and Aude Billard
LASA Laboratory, Ecole Polytechnique Federal de Lausanne (EPFL)
Email: {walid.amanhoud, mahdi.khoramshahi, aude.billard}@epfl.ch

Abstract—Many tasks require the robot to enter in contact with surfaces, be it to take support, to polish or to grasp an object. It is crucial that the robot controls forces both upon making contact and while in contact. While many solutions exist to control for contact, none offer the required robustness to adapt to real-world uncertainties, such as sudden displacement of the object prior and once in contact. To adapt to such disturbances require to re-plan on the fly both the trajectory and the force. Dynamical systems (DS) offer a framework for instant re-planning of trajectories. They are however limited to control of motions. We extend this framework here to enable generating contact forces and trajectories through DS. The framework allows also to modulate the impedance so as to show rigidity to maintain contact, and compliance to ensure safe interaction with humans.

We validate the approach in single and dual arm setting using KUKA LWR 4+ robotic arms. We show that the approach allows 1) to make smooth contact while applying large forces, 2) to maintain desired contact force when scanning non-linear surfaces, even when the surface is moved, and 3) to grasp and lift smoothly an object in the air, and to re-balance forces on the fly to maintain the grasp even when subjected to strong external disturbances.

I. INTRODUCTION

Force control is a major topic in robotics especially nowadays where numerous tasks require robots to interact with humans in dynamically changing environments (Figure 1). Object manipulation, surface operation, tele-manipulation in surgery are a few examples where force control is crucial to react to unexpected situations. The robot must not only adapt on the fly, but exhibit robustness to uncertainties on the surrounding environment. Approaches to force control can be divided into two broad categories, namely direct force control where a target force (and position) is achieved with an explicit force feedback closure and indirect force control where interaction forces are controlled through an impedance control law [1]. Direct force control is still very much influenced by the hybrid position/force control introduced in 1981 [2]. This approach decomposes the task in two orthogonal decoupled subspaces, that are task-specific [3], and where position and force are controlled separately. Transition from free motion to motion in contact is performed in stages. In free motion, a position controller drives the robot in all control directions. Once the robot is in contact, the position controller switches to the hybrid force/position controller. The strength of such approach is that it ensures accurate position and force tracking. However, as it neglects the interaction dynamics between the robot and the environment, it lacks robustness in the face of disturbances and uncertainties and may lead to instabilities at contact or contact loss at run time [4].

Impedance control was offered as a solution to some of these drawbacks [5]. Impedance control considers explicitly the interaction dynamics with the environment. It simulates the dynamical relationship between interaction forces and positional deviations through a mass-spring-damper system. The relationship is called impedance when force is generated from motion deviation and admittance in the opposite way. Impedance controllers are suitable for providing a compliant behavior in all phases of a contact task (non-contact, transition and contact) [6], but are limited in their ability to track forces, mainly due to partial knowledge of the environment (e.g., location and stiffness). To overcome this limitation, two distinct strategies are usually employed in the literature: impedance and set-point adaptation. Impedance adaptation adjusts the impedance parameters online (e.g., inertia, damping, and stiffness) to improve tracking in response to force, position, or velocity measurements [7, 8, 9, 10]. Set-point adaptation approach improves force tracking by adjusting the impedance set-point (e.g., reference position) based on force tracking error or on real-time estimation of the environment’s change in stiffness [11, 12, 13, 14, 15]. Other strategies use impedance control, but in combination with direct force control to provide accurate force tracking and a compliant robot behavior in response to external forces [16, 17].
aforementioned works are effective in terms of motion and force tracking performance, the robustness of their methods to real-time disturbances is either not mentioned or limited to small disturbances, such as in the surface location or in the environment stiffness. Large disturbances such as those introduced by human interactions are not addressed. For example, let us consider a scenario where a human operator supervises a polishing task performed by a robot on a surface as illustrated in Figure 2. At any moment, the human might stop the robot or pull it away from the surface (e.g., breaking the contact) to monitor the current execution of the task. In such situations, it is crucial for the robot to “react” and not only damp the disturbances using the impedance control (which is only effective for small disturbances). To properly react, the robot is required to re-plan the execution of the contact task from the disturbed state. This reactivity needs to be continuous and smooth as the human behavior can be highly dynamic. Representing tasks with time-indexed references for position and force profiles is the main drawback in current approaches in achieving fast reactivity toward large disturbances; see [14, 15] as examples where a time-dependent representation of the task (based on Dynamic Movement Primitive) is used. In contrast, in a state-dependent and time-invariant task representation, human disturbances can be captured by changes in the robot’s state which is used in the re-planning of the task.

In this work we design a control strategy to perform contact tasks with robustness to large real-time disturbances. As illustrated in Figure 1 and 2 the controller reacts not only to human interactions (e.g., stopping the robot, breaking the contact, and moving the robot arbitrarily), but also to unexpected changes in the environment (e.g., the position and orientation of the surface/object). This controller is aimed for industry 4.0 where robots need to perform dexterous contact tasks while remaining safe for their human co-workers [18]. To this end, we propose to encode contact tasks by combining the desired motions and contact force profiles in a single mathematical expression, by ways of time-invariant dynamical system (DS). DS provide very fast reactivity and enable on-the-fly re-planning of trajectories, as demonstrated for catching smoothly objects in flight [19] or reaching and moving on a surface [20]. They are also suitable for encoding tasks learned from human demonstrations [21] and generating impedance control laws providing a compliant and passive robot behavior [22]. In this paper we contribute to this literature by extending the time-invariant DS-based control framework to perform contact tasks. We propose a strategy based on local modulation of the robot’s nominal task dynamics to generate the desired motion and contact forces when the robot is close to the surface. As a result, this strategy offers:

- Stable and accurate motion and contact force generation.
- Compliant behavior in all phases of the contact task (in free motion, upon making contact and when in contact).
- Robustness to real-time disturbances.

We present our approach in section III. We evaluate it in section III on two different real-world scenarios: a polishing task on a non-flat surface and a reaching, grasping and manipulation task. We conclude with a discussion about the method and results obtained and future work in section IV.

II. PROPOSED APPROACH

An autonomous dynamical system usually takes as input a state variable (e.g., real position \(x\)) and returns the rate of change of that variable (e.g., desired velocity \(\dot{x}\)). It can be seen as a velocity vector field describing the desired behavior for any given position in space.

As illustrated in Figure 3 we assume the existence of a nominal DS \(f(x)\) that brings the robot in contact with a surface and moves it along the surface. We suppose that the contact surface is non-penetrable and that we have an explicit expression for the normal vector \(n(x)\) and distance to the surface \(\Gamma(x)\) at all points in space. The nominal DS should satisfy:

\[
\begin{align*}
    f(x)^T n(x) &= 0 & \text{in contact} \\
    f(x)^T n(x) &> 0 & \text{in free motion}
\end{align*}
\]

Such dynamics can be learned from human demonstrations and locally modulated to meet these constraints [21, 22]. Once the robot is on the surface, it must apply a state-dependent desired force profile along the normal to the surface \(F_d(x) \in [0, F_{\text{max}}]\) with \(F_{\text{max}} > 0\). Thereafter, we show how to modulate the nominal DS to generate contact forces in addition to motion.

A. Robot dynamics and control

The generated contact forces are not only the result of the desired motions but also of the dynamics of the robot. We express the dynamics of a \(N\) degrees of freedom robotic manipulator in the three-dimensional Cartesian space:

\[
M(x)\ddot{x} + C(x, \dot{x})\dot{x} = F_e + F_c
\]
where \( x \in \mathbb{R}^3 \) denotes the robot’s position, \( M(x) \in \mathbb{R}^{3 \times 3} \) the mass matrix, \( C(x, \dot{x}) \dot{x} \in \mathbb{R}^3 \) the centrifugal forces, while \( F_c \in \mathbb{R}^3 \) and \( F_e \in \mathbb{R}^3 \) represent the control and external forces respectively. Eq. [2] presumes that the gravity forces \( g(x) \in \mathbb{R}^3 \) are already compensated. The control force \( F_c \) allows to track a desired velocity profile \( \dot{x}_d \in \mathbb{R}^3 \) and is obtained from the DS-impedance controller in [22]:

\[
F_c = D(x)(\dot{x}_d - \dot{x}) = d_1 \dot{x}_d - D(x)\dot{x}
\]

(3)

where \( D(x) \in \mathbb{R}^{3 \times 3} \) is a state-varying damping matrix, constructed in such a way that the first eigenvector is aligned with the desired dynamics \( \dot{x}_d \) with positive eigenvalue \( d_1 \in \mathbb{R}^+ \). The first term in Eq. [3] represents the driving force along the desired dynamics where \( d_1 \) appears as an impedance gain. The last term is the damping force that can be manipulated through the last two eigenvalues of \( D(x) \) (\( d_2 \) and \( d_3 \in \mathbb{R}^+ \)) to selectively damp disturbances that are orthogonal to the desired velocity.

In this paper, the DS is applied only to the translation of the end-effector. The desired end-effector’s orientation (see Appendix [A] for details) is tracked using the axis-angle representation. The measured and desired orientation are specified as full rotation matrix by \( R = [x^E, y^E, z^E] \in \mathbb{R}^{3 \times 3} \) and \( R_d = [x^E_d, y^E_d, z^E_d] \in \mathbb{R}^{3 \times 3} \) respectively. The orientation error is computed as \( \hat{R} = R_d R^T \in \mathbb{R}^{3 \times 3} \) and the corresponding axis-angle representation \( \hat{\zeta} \) extracted to compute a control moment using a PD-like control law. The control wrench formed by the control moment and force (e.g., \( F_c \)) is then converted into joint torques using the robot’s Jacobian matrix \( J \in \mathbb{R}^{6 \times N} \). Therefore, we assume that the robot has torque sensing ability and is torque-controlled. Torque-controlled robots allow for compliant interaction control; specifically impedance control which exhibits satisfactory performance in interaction with stiff environments [20], as it is the case in our work.

B. A DS-based strategy for contact task

To achieve the desired motion and force profile with a single DS, we decompose the system as follows:

\[
\dot{x}_d = f(x) + f_n(x)
\]

(4)

with \( \dot{x}_d \), the desired velocity profile and \( f_n(x) \), a modulation term that applies only along the direction normal to the surface. Inserting Eq. [4] in Eq. [3] the control force becomes:

\[
F_c = d_1 f(x) + d_1 f_n(x) - D(x)\dot{x}
\]

(5)

The first term represents the driving force along the nominal dynamics, the second term is the damping force, while the second term denotes the modulation force along the normal direction to the surface that we design as follows:

\[
f_n(x) = \frac{F_d(x)}{d_1} n(x)
\]

(6)

As an illustration of the strategy, in Figure [2] the nominal DS presented in Figure [3] is modulated to generate a contact force once the robot reaches the surface. Before contact with the surface, the desired and nominal DS are aligned and identical. Close to contact, the normal modulation component gets generated and modulates the nominal DS to produce the desired force. To illustrate the robustness of our approach in face of disturbances, an external force disturbs the robot away from the surface while the robot is moving. The modulated DS reacts to the disturbance by realigning with the nominal one. Once the disturbance disappears, the robot reaches the surface and moves toward the target while applying the desired contact force.

When controlling a robot interacting with unknown environments, one should make sure that the interaction is stable for both performance and safety purposes. A sufficient condition to achieve stability is to ensure passivity of the whole system [27, 28]. It implies that the system never generates extra energy or in other words, that the total
energy of the system is bounded by the initial stored energy plus the one injected in the system from the interaction with the environment. To achieve passivity and stability of our approach, we propose a formulation based on energy tanks [29] to monitor the energy flow in the system and prevent the generation of extra energy coming from the control actions. The details of the formulation are provided in Appendix B.

C. Application to a reaching, grasping and manipulation task

Fig. 5: Scenario for reaching, grasping and manipulating an object with two robotic arms

In this section, we investigate how the modulation strategy can be applied to the challenging task of reaching, grasping and manipulating an object using two robotic arms. The formulation presented here is experimentally evaluated in Section III-B. To start, let us consider the scenario illustrated in Figure 5. The main variables used to describe the problem are provided in Table I. For the rest of this section, superscript L and R will refer to the left and right robot respectively.

The robots’ center position $x_C$ and distance vector $x_D$ are computed from their tool tip positions $x_R$ and $x_L$:

$$x_C = \frac{x_L + x_R}{2} \quad x_D = x_R - x_L \quad (7)$$

from where we can derive the relation below:

$$\dot{x}_R = \dot{x}_C + \frac{\dot{x}_D}{2} \quad \dot{x}_L = \dot{x}_C - \frac{\dot{x}_D}{2} \quad (8)$$

To reach and manipulate the object during the task, we choose to couple the robots’ motion by controlling for a desired robots’ centre position $x_C^d$ and distance vector $x_D^d$, using simple linear dynamics:

$$\begin{align*}
\dot{x}_C^d &= A_C (x_C^d - x_C) \\
\dot{x}_D^d &= A_D (x_D^d - x_D)
\end{align*} \quad (9)$$

where $A_C$ and $A_D$ are positive gain diagonal matrices. Basically, $\dot{x}_C^d$ specifies the desired positioning behavior of the robots’ center while $\dot{x}_D^d$ defines the desired closing behavior on the object’s surface. $x_C^d$ and $x_D^d$ can be set to $x_C^o$ and $x_D^o$ respectively during the reaching phase and modified during the manipulation phase.

To do the grasping part of the task, we use the modulation strategy presented in section II-B. First, we introduce the nominal DS $f_R(x_L, x_R)$ and $f_L(x_L, x_R)$. The nominal DS should bring each robot in contact with the target surface (e.g., the object’s surface). This role is achieved by $\dot{x}_d^R$. Following
the nominal DS are defined such that:
\[
f_R(x^L, x^R) = -f_L(x^L, x^R) - \frac{\dot{x}_d^D}{2}
\]  
(10)

Once the robots reach the object’s surface, they should generate the desired contact force profile \( F_d(x^L, x^R) \geq 0 \) which is assumed to be the same for both of them. To this end, the normal modulation terms are defined as follows:
\[
f_n^i(x^L, x^R) = \frac{F_d(x^L, x^R)}{d_i} n^i \quad i = L, R
\]  
(11)

The force application directions \( n^R \) and \( n^L \) are derived from the desired distance vector. For a box (with two parallel surfaces), they are opposite for the two robots:
\[
n^L = -n^R = \frac{x_d^D}{\|x_d^D\|}
\]  
(12)

From there, the desired robots’ velocity can be finally expressed:
\[
\dot{x}_d^i = f^i(x^L, x^R) + f_n^i(x^L, x^R) + \dot{x}_d^C \quad i = L, R
\]  
(13)

which includes the desired robots’ center dynamics \( \dot{x}_d^C \), needed to properly positioned the robots’ centre. The modulated DS are then tracked with the DS impedance controller (Eq. 3). Finally, in Appendix C we show how passivity can be preserved for this bi-manual task.

III. EXPERIMENTAL EVALUATION

In this section, we evaluate the DS-based strategy proposed in two real world tasks: a) polishing of a non-flat surface using a single robotic arm robot and b) reaching, grasping and manipulating an object with two robotic arms. We assess the ability of the approach to generate the desired force profile and to do so in the different types of disturbances, by moving unexpectedly the surface/object prior and during contact, or breaking the contact. These experimental evaluations can be watched in a video available on-line: https://youtu.be/lz0uxUEVc3g

A. Polishing task on a non-flat surface

The DS modulation strategy is first tested on a circular polishing task on a non-flat surface as illustrated in Figure 2. A 7-DOF robotic arm (KUKA LWR IV+) is used to perform the task. The robot is equipped with joint torque sensors at the actuators and can be torque-controlled. A 6-axis ATI force-torque sensor is also mounted on the end-effector on which a 3D printed finger tool is attached. The non-flat and rigid surface is fabricated by deforming a Plexiglas sheet using heat. It is attached on a wooden plate whose pose is tracked by a motion capture system. The robot’s behavior is evaluated in a simple scenario: the robot comes in contact with the target surface to perform a circular motion on the surface while applying the desired contact force and experiences disturbances from a human. The technical details of the implementation are provided in Appendix D-A.

Figure 6 shows the measured and desired force profiles recorded during the experiment. The robot firstly reaches the surface to perform the polishing task without experiencing any disturbances. The force generation is relatively accurate with a RMS force error of around 1.9 N (19% of the desired force) during this period. After a while, the human intentionally breaks the contact with the surface by pulling the robot away from the surface. No instabilities are observed during this phase. Once released by the human, the robot simply returns back to the surface following the flow of the DS to perform the task. Then, the human interacts with the system by pushing and stalling the robot while the robot is in contact with the surface. The measured force remains smooth denoting a safe and stable interaction. Finally, after breaking the contact a second time, the robot reaches the surface again and the human momentarily change the inclination of the surface. The robot smoothly complies to the disturbance without getting unstable.

B. Reaching, grasping and manipulation task

The second experimental evaluation is done with two KUKA LWR IV+ robots to reach and grasp a cardboard box as shown in Figure 1. The box has a mass of 0.65±0.05 kg and is tracked by the motion capture system to get its pose. Both robots are equipped with a 6-axis ATI force-torque sensor at the end-effector on which a flat palm is mounted for grasping. The evaluation scenario is designed such that the two arms
reach and grasp the object before a human comes and interacts with the system by moving it around, changing its orientation or even breaking the grasp. The implementation is based on section II-C while the technical details are provided in Appendix D-B.

Figure 7a illustrates the measured and desired contact forces. The RMS force error when the object is grasped and without human disturbances are around 1.7 N (11.3% of the desired force) for both robots. The non-contact/contact transition in the reaching and grasping phases is smooth and no instability is observed in the force profiles when the human intentionally breaks the grasp. Similarly, despite disturbances applied on the system after grasping (e.g., fast shocks on the box, changing's system pose), the measured forces remain smooth guaranteeing the stability and delivering a satisfactory compliant behavior.

Figure 7b illustrates the behavior of the energy tanks for both robots (see Appendix C). The tanks are initialized at the maximum allowed level which is set to 4.0 J. When the robots are initially moving toward the object, energy is mainly dissipated. However, this dissipated energy cannot be stored in the tanks since they are already full. Close to contact, a desired contact force starts to be generated while the robots are still slightly moving. These non-passive actions are implemented by extracting energy from the tanks. Once the object is grasped, the tanks levels remain constant until the human moves the robots to lift the object. This dissipated energy is stored in the tanks but in a non-symmetrical way due to the interaction. When the human applies fast shocks on the object, the tanks level is barely changing as the robots barely move. Then, moving the system to the left direction (from human’s point of view) cause the right arm to generate extra energy as it moves in the direction where it applies the force while the left robot dissipates energy. A high amount of energy is extracted from the tank of the right robot to execute this non-passive action and maintain the grasp. When pushing the system to the right, the opposite behavior happens with energy being generated by the left robot and dissipated by the right one, leading their associated tanks to be respectively drained and filled. A similar reasoning can be applied to the other disturbance phases where the human moves the arms to change the object’s orientation or breaks the grasp.

IV. SUMMARY AND CONCLUSION

In this work, we proposed a Dynamical System approach to generate motion and force in contact tasks with robotic manipulators. The strategy is based on local modulation of a nominal DS that allows to reach a target surface and move on it. The use of the DS framework provides flexibility and smoothness in the motion and force generation as well as robustness to real-time disturbances. As in traditional impedance controller, the generation of the contact forces is achieved implicitly and results in a relatively accurate tracking performance as demonstrated in the experimental
Let us then consider a storage function $W$ with $f$. In this work, the desired orientation profile of the robot’s end-effector is defined based on substituting $\mathbf{M}$ corrected either using the approach described in section II-A after converting $\kappa > 0$. Therefore, we cannot guarantee passivity of the system with respect to the environment. To restore passivity, we consider an approach based on energy tanks. First proposed in [29], energy tanks have been quite exploited in the literature since they appear as a very flexible way to keep track of the energy flow in the system and prevent instabilities coming from control actions [30, 31, 32]. A tank is basically a reservoir of energy with given initial and maximum (allowed) levels. The energy dissipated by the system is used to fill the tank from where energy is extracted to temporarily execute potential non-passive actions. Energy extraction is allowed as long as the storage is not depleted such that passivity of the whole system (including the tank) is preserved.

Let us therefore introduce a virtual tank state $s$ that stores the dissipated energy in the system mainly coming from the damping term $\mathbf{P}$. We use this energy to modulate the tank’s dynamics, which is coupled with the robot’s state $(x, \dot{x})$ as follows:

$$\dot{s} = \alpha(s)p_d - \beta(s, p_d)p_r - \beta_n(s, p_r)p_n$$

(23)

The scalar functions $\alpha(s), \beta(s, p_d)$ and $\beta_n(s, p_r)$ control the energy flow between the virtual tank and the robot. Let us define $s_m$ as the maximum energy level allowed to be stored in the tank. To ensure that $s$ remains bounded between 0 and $s_m$, we define the scalar functions such that:

$$\alpha(s) = \Upsilon_{s_m - \delta_s, s_m}(s)$$

(24)

$$\beta(s, p_d) = \begin{cases} 0 & \text{if } s < 0 \text{ and } p_d > 0 \\ 0 & \text{if } s > s_m \text{ and } p_d < 0 \\ 1 & \text{otherwise} \end{cases}$$

$$\beta_n(s, p_r) = \begin{cases} 1 & \text{if } p_r < 0 \\ \beta_n(s, p_r) & \text{otherwise} \end{cases}$$

where $\delta_s \in [0, s_m]$ while the function $\Upsilon_{\text{min}, \text{max}}(s)$ provides a smooth transition from 1 to 0 as $s$ transits from a to b:

$$\Upsilon_{a,b}(x) = \begin{cases} 1 - \frac{1}{2} \left( 1 + \cos \left( \frac{\pi}{4} \left( x - a \right) \right) \right) & a \leq x \leq b \\ 0 & x > b \end{cases}$$

(25)

with $x, a, b \in \mathbb{R}$ and $a < b$. If the tank is depleted, the controller should not generate the potential non-passive actions. The control law should be corrected accordingly by taking the state of the tank and the power variables into account. To this end, we correct the modulation law in Eq. 23 as follows:

$$\dot{x}_d = f_1'(x) + f_2'(x)$$

(26)

with:

$$f_1'(x) = f_1(x) + \beta(s, p_d)f_3(x)$$

$$f_2'(x) = \beta_n(s, p_r)f_4(x)$$

(27)

where $\beta(s, p_d)$ and $\beta_n(s, p_r)$ are scalar functions satisfying:

$$\beta(s, p_d) = \begin{cases} 1 & \text{if } p_d < 0 \\ \beta(s, p_d) & \text{otherwise} \end{cases}$$

$$\beta_n(s, p_r) = \begin{cases} 1 & \text{if } p_r < 0 \\ \beta_n(s, p_r) & \text{otherwise} \end{cases}$$

(28)

Let us now define the final storage function $W(x, \dot{x})$ taking the tank’s dynamics into account:

$$W(x, \dot{x}) = \frac{1}{2} \dot{x}^T \mathbf{M}(x) \dot{x} + d_1 V_c(x) + s$$

(29)

Substituting $s$ by Eq. 23 and $\dot{x}_d$ by Eq. 26 the rate of change of $W(x, \dot{x})$ becomes:

$$W(x, \dot{x}) = \frac{1}{2} \dot{x}^T \mathbf{M}(x) \dot{x} + d_1 V_c(x) + s$$

(30)

The first two terms are now both non-positives, while the third one remains dissipative since $1 - \alpha(s) \geq 0$. As a result, the full system is passive with respect to $\dot{x}^T \mathbf{M} \dot{x}$. The control law is:

$$\dot{x}_d = -\nabla x V_c(x')$$

(31)

**Appendix C**

**Passivity analysis: bi-manual task**

Guaranteeing the passivity of the bi-manual system is difficult in particular if the desired dynamics of both robots are coupled. However, let us assume that the desired dynamics $\dot{x}_d^1$ and $\dot{x}_d^2$ are both conservatives such that:

$$\dot{x}_d^1 = -\nabla x V_c(x')$$

(31)
where \( V_C(\mathbf{x}_C) \) and \( V_D(\mathbf{x}_D) \) are potential functions. Setting the same impedance gains for both robots: \( d_1 = d_2 = d_3 \), one can consider similarly to Eq. 18 a global storage function \( W(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) \) taking into account both robots’ kinetic energy and the potential functions such that:

\[
W(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) = \frac{1}{2} \left[ \dot{\mathbf{x}}_R^T \mathbf{B}_R(\mathbf{x}_R) \dot{\mathbf{x}}_R + \dot{\mathbf{x}}_L^T \mathbf{B}_L(\mathbf{x}_L) \dot{\mathbf{x}}_L \right] + 2d_1 V_C(\mathbf{x}_C) + \frac{d_1}{2} V_D(\mathbf{x}_D)
\]

where \( \mathbf{B}_R(\mathbf{x}_R) \) and \( \mathbf{B}_L(\mathbf{x}_L) \) are the robots’ inertia matrices.

Differentiating Eq. 32 using Eq. 21 leads to:

\[
\dot{W}(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) = \frac{d}{dt} \left[ \dot{\mathbf{x}}_R^T \mathbf{B}_R(\mathbf{x}_R) \dot{\mathbf{x}}_R + \dot{\mathbf{x}}_L^T \mathbf{B}_L(\mathbf{x}_L) \dot{\mathbf{x}}_L \right] = -\dot{d}_1 \dot{\mathbf{x}}_C^T \dot{\mathbf{x}}_a - \frac{d_1}{2} \dot{\mathbf{x}}_D^T \dot{\mathbf{x}}_D
\]

From the definition of \( \dot{\mathbf{x}}_C^T \) and \( \dot{\mathbf{x}}_D \) in Eq. 7 it becomes:

\[
\dot{W}(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) = \frac{1}{2} \left[ \dot{\mathbf{x}}_R^T \mathbf{B}_R(\mathbf{x}_R) \dot{\mathbf{x}}_R + \dot{\mathbf{x}}_L^T \mathbf{B}_L(\mathbf{x}_L) \dot{\mathbf{x}}_L \right] - \dot{d}_1 \left( \dot{\mathbf{x}}_R + \dot{\mathbf{x}}_L \right)^T \dot{\mathbf{x}}_a - \frac{d_1}{2} \left( \dot{\mathbf{x}}_R - \dot{\mathbf{x}}_L \right)^T \dot{\mathbf{x}}_D
\]

Expanding the time derivative of the robots’ kinetic energy as done in Eq. 19 and 20 results in:

\[
\dot{W}(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) = \dot{\mathbf{x}}_R^T \mathbf{B}_R(\mathbf{x}_R) \dot{\mathbf{x}}_R + \dot{\mathbf{x}}_L^T \mathbf{B}_L(\mathbf{x}_L) \dot{\mathbf{x}}_L - \dot{d}_1 \left( \dot{\mathbf{x}}_R + \dot{\mathbf{x}}_L \right)^T \dot{\mathbf{x}}_a - \frac{d_1}{2} \left( \dot{\mathbf{x}}_R - \dot{\mathbf{x}}_L \right)^T \dot{\mathbf{x}}_D
\]

Substituting \( \mathbf{F}_R^R \) and \( \mathbf{F}_L^L \) using Eq. 13 and 3 gives:

\[
\begin{align*}
\dot{W}(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) &= d_1 \dot{\mathbf{x}}_R^T \mathbf{B}_R(\mathbf{x}_R) \dot{\mathbf{x}}_R + d_1 \dot{\mathbf{x}}_L^T \mathbf{B}_L(\mathbf{x}_L) \dot{\mathbf{x}}_L - \frac{d_1}{2} \dot{\mathbf{x}}_D^T \dot{\mathbf{x}}_D \\
&= \dot{d}_1 \dot{\mathbf{x}}_C^T \dot{\mathbf{x}}_a - \frac{d_1}{2} \dot{\mathbf{x}}_D^T \dot{\mathbf{x}}_D
\end{align*}
\]

Cancelling the terms with \( \dot{\mathbf{x}}_a^T \) and \( \dot{\mathbf{x}}_D \) (using Eq. 10) finally leads to:

\[
\dot{W}(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) = \dot{W}^i(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) + \dot{W}^C(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R)
\]

where:

\[
\dot{W}^i(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) = d_1 \dot{\mathbf{x}}_R^T \mathbf{B}_R(\mathbf{x}_R) \dot{\mathbf{x}}_R - \dot{d}_1 \dot{\mathbf{x}}_C^T \dot{\mathbf{x}}_a - \frac{d_1}{2} \dot{\mathbf{x}}_D^T \dot{\mathbf{x}}_D
\]

\[
\dot{W}^C(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) = \dot{d}_1 \dot{\mathbf{x}}_C^T \dot{\mathbf{x}}_a - \frac{d_1}{2} \dot{\mathbf{x}}_D^T \dot{\mathbf{x}}_D
\]

\[
\dot{W}(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) = \dot{d}_1 \dot{\mathbf{x}}_C^T \dot{\mathbf{x}}_a - \frac{d_1}{2} \dot{\mathbf{x}}_D^T \dot{\mathbf{x}}_D
\]

\[
\dot{W}(\mathbf{x}_L, \mathbf{x}_R, \dot{\mathbf{x}}_L, \dot{\mathbf{x}}_R) = \dot{d}_1 \dot{\mathbf{x}}_C^T \dot{\mathbf{x}}_a - \frac{d_1}{2} \dot{\mathbf{x}}_D^T \dot{\mathbf{x}}_D
\]

The desired force profile in contact \( F_d(\mathbf{x}) \) is implemented such that:

\[
F_d(\mathbf{x}) = \begin{cases} 
F_T & \mu F_x \geq \epsilon_F \wedge \Gamma(\mathbf{x}) \leq \epsilon_T \\
F_{T,\text{min}} & \mu F_x < \epsilon_F \wedge \Gamma(\mathbf{x}) \leq \epsilon_T \\
0 & \text{otherwise}
\end{cases}
\]

SecondHands.

The authors would like to thank for the support from the Hasler Foundation and the European Community’s Horizon 2020, in particular the robotics program ICT-23-2014 under grant agreement 644727-CogIMon and the Research and Innovation programme ICT-2014-1 under grant agreement 643950-SecondHands.
REFERENCES


[26] Christian Ott, Ranjan Mukherjee, and Yoshihiko Nakamura. Unified impedance and admittance control. In


