Toward Safe Interaction Control with Dynamical Systems

Klas Kronander and Aude Billard

***Abstract*—Autonomous Dynamical Systems (DS) has emerged as an extremely flexible and powerful method for modeling robotic tasks. Task execution of DS models is typically done in an open-loop manner in combination with standard low level controller, e.g. position controller or impedance controller. Such an arrangement has two important drawbacks 1) it is not passive and 2) the DS model can not respond to physical perturbations on the robot body. These are severe limitations in tasks with uncertain physical contacts, e.g. object handovers. We propose a novel control architecture that closes the loop around the DS, ensures passivity and allows tuning of the impedance. We evaluate our approach in a comparative study in an uncertain manipulation task with unexpected contact.**

I. Introduction

Dynamical Systems (DS) has emerged as a general and highly flexible means of representing robot motions. It has been demonstrated that many of the proposed DS formulations lend themselves well to learning, both in a supervised setting as well as reinforcement learning. Furthermore, in special cases qualitative properties such convergence to a limit cycle or stability at an attractor point can be ensured regardless of the data provided to the learning algorithm. We believe that the capability of encoding not only a nominal motion plan but also how the robot should *respond to perturbations* makes the DS task representation very well suited to object handover tasks which are characterized by a high amount of uncertainty and specifically the need to instantly react to non-predicable behavior of a human.

In parallel to the development of DS-based learning systems, the field of robotic manipulation has in recent years seen an revitalized interest in control of mechanical interaction, a topic which largely rests upon foundations of Hogans impedance control formulation [4]. To use a DS task representation with an impedance controller, it is necessary to integrate the DS over time to yield a reference trajectory, see Fig. 1 left. In such a configuration, the reactivity of the DS is used with respect to perturbations that are captured using external sensors, e.g. the pose estimate of a moving target point or obstacles. However, the full power of the DS is not used, since the integration of the reference trajectory disallows the DS to react to physical perturbations on the robot. For physical interaction tasks such as hand-overs, such reactivity can be crucial, and it would hence be desirable to instead use a controller that feeds back the actual state of the robot to the DS, see 1, right.

An important property for controllers interacting with unknown environments is passivity. A controller that ensures a passive relation between external forces and robot velocity will yield stable behavior in free motion and in contact with any passive environment [1]. In this sense, classic impedance control is only passive in the regulation case and the passivity can no longer be ensured if the desired velocity is non-zero. The loss of passivity during tracking is an important drawback of impedance control and a problem that arises in any controller driven by time-indexed reference trajectories. A through analysis of this problem is provided in [6], which advocates to tackle it by encoding tasks using time-independent velocity fields (first order DS) and proposes a controller that ensures allows tracking of desired DS. Related work has proposed a similar approach for passivity in the curve tracking problem [2]. These works exploit a time-independent encoding of the task to ensure passivity and energy-efficient, accurate tracking of the DS. The closed-loop dynamics are however rather complicated and the specification of a mechanical impedance becomes non-intuitive.

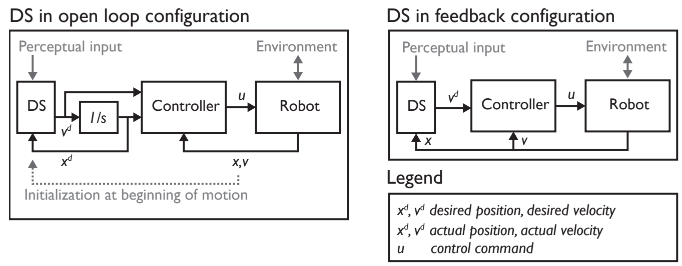


Fig. 1. Illustration of open loop and closed loop control configurations with DS. In the open loop configuration, the DS is updated with the desired position resulting from integration of the desired velocity. The actual position of the robot is only used for initializing the integration at the beginning of the motion. In contrast, the feedback configuration continuously updates the DS with the actual position and realizes a control on the velocity error.

In this work, we aim to combine the advantages of impedance control and a passive control system without dependency on time. In contrast to [6] and [2] which are based on redistribution of kinetic energy along the desired direction of motion, we propose a control structure which is based on selective dissipation of energy in directions that are irrelevant to the task. As we shall see, this allows to easily tune the mechanical impedance while ensuring passivity.

We evaluate our controller in a robotic reaching task with limited knowledge on 1) the robot dynamics and 2) the environment. We show that the proposed controller has advantages over classical impedance control both in respecting the shape of the desired motion as well as keeping forces low in unexpected contact.

II. Problem Statement

Let $\mathbf{f}(\boldsymbol{\xi})$ be a Dynamical System describing a nominal motion plan for a robotic task. The variable $\boldsymbol{\xi}$ represents a generalized state variable, which could be e.g. robot joint angles or Cartesian position. Any integral curve of $\mathbf{f}$ represents the desired motion of the robot in the absence of perturbations. We consider rigid-body dynamics described in the generalized state variable $\boldsymbol{\xi}$:

$$\mathbf{M}(\boldsymbol{\xi})\ddot{\boldsymbol{\xi}} + \mathbf{C}(\boldsymbol{\xi},\dot{\boldsymbol{\xi}})\dot{\boldsymbol{\xi}} + g(\boldsymbol{\xi}) = \boldsymbol{\tau}_c + \boldsymbol{\tau}_e \tag{1}$$

The goal of this work is to design a controller $\boldsymbol{\tau}_c$ so that Eq. (1) has the following properties:

1) Passivity $(\boldsymbol{\tau}_e, \dot{\boldsymbol{\xi}})$ should be preserved for the controlled system.
2) The controller should dissipate kinetic energy in directions not relevant for the task.
3) It should be possible to vary task-based impedance if the manipulator, e.g. how dynamics defining how external forces $\boldsymbol{\tau}_e$ affect the velocity $\dot{\boldsymbol{\xi}}$.

III. Selective dissipation via varying damping

Our controller is based on a state-varying damping term that dissipates selectively in directions orthogonal to the desired direction of motion given by $\mathbf{f}(\boldsymbol{\xi})$. Let $\mathbf{e}_1, \dots, \mathbf{e}_N$ be an orthonormal basis for $\mathbb{R}^N$ with $\mathbf{e}_1$ pointing in the desired direction of motion. Let the matrix $\mathbf{Q}(\boldsymbol{\xi}) \in \mathbb{R}^{N\times N}$ be a matrix whose columns are given by $\mathbf{e}_1, \dots, \mathbf{e}_N$. This matrix is a function of the state $\boldsymbol{\xi}$, since the vectors $\mathbf{e}_1$ and hence all $\mathbf{e}_1, \dots \mathbf{e}_N$ depend on $\boldsymbol{\xi}$ via $\mathbf{f}(\boldsymbol{\xi})$. We then define the state-varying damping matrix $\mathbf{D}(\boldsymbol{\xi})$ as follows:

$$\mathbf{D}(\boldsymbol{\xi}) = \mathbf{Q}(\boldsymbol{\xi})\boldsymbol{\Lambda}\mathbf{Q}(\boldsymbol{\xi})^T \tag{2}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix with non-negative values on the diagonal $\lambda_1, \dots, \lambda_N \geq 0$. By adjusting the eigenvalues, different dissipation behaviors can be achieved. For example, setting $\lambda_1 = 0$ and $\lambda_2, \dots, \lambda_N > 0$ results in a system that selectively dissipates energy in directions perpendicular to the desired motion. Hence, external work in irrelevant directions is opposed while along the integral curves of $\mathbf{f}(\boldsymbol{\xi})$ the system would be free to move.

IV. Proposed controller

Let the general DS be decomposed into a conservative part and a non-conservative part:

$$\mathbf{f}(\boldsymbol{\xi}) = \mathbf{f}_C(\boldsymbol{\xi}) + \mathbf{f}_R(\boldsymbol{\xi}) \tag{6}$$

where $\mathbf{f}_C$ is a conservative DS, i.e. there exists an associated potential function $V_C(\boldsymbol{\xi})$ such that $\mathbf{f}_C = -\nabla V_C(\boldsymbol{\xi})$ and where $\mathbf{f}_R$ denotes the non-conservative part. Note that any system can be written on this form, e.g. if no conservative part can be extracted from $\mathbf{f}$ we would simply have $\mathbf{f}_c \equiv 0$.

Similar to the concept of energy tanks which are extensively used in haptics and telemanipulation [7] and recently for varying stiffness [3], we introduce a virtual state variable $s \in \mathbb{R}$ that will act as a temporary energy storage for the system. It is a virtual state to which we assign the following dynamics:

$$\dot{s} = \alpha(s)\dot{\boldsymbol{\xi}}^T\mathbf{D}\dot{\boldsymbol{\xi}} - \beta_s(z,s)\lambda_1 z \tag{7}$$

TABLE I

Specifications of the scalar functions α, β_s and β_R

For some given upper bound of the virtual storage, the scalar functions managing the flow of energy in and out of the virtual storage should be continuous functions satisfying:

$$\begin{cases} 0 \leq \alpha(s) \leq 1 & s < \overline{s} \\ \alpha(s) = 0 & s \geq \overline{s} \end{cases} \tag{3}$$

$$\begin{cases} \beta_s(z,s) = 0 & s \leq 0 \text{ and } z \geq 0 \\ \beta_s(z,s) = 0 & s \geq \overline{s} \text{ and } z \leq 0 \\ 0 \leq \beta(z,s) \leq 1 & \text{elsewhere} \end{cases} \tag{4}$$

For limiting non-passive control effort when the storage is depleted, $\beta_R(z,s)$ should be a continuous function satisfying:

$$\begin{cases} \beta_R(z,s) = \beta_s(z,s) & z \geq 0 \\ \beta_R(z,s) \geq \beta_s(z,s) & z < 0 \end{cases} \tag{5}$$

where $z = \dot{\boldsymbol{\xi}}^T\mathbf{f}_R(\boldsymbol{\xi})$ has been introduced. The scalar functions $\alpha : \mathbb{R} \mapsto \mathbb{R}$ and $\beta_s, \beta_R : \mathbb{R}\times\mathbb{R} \mapsto \mathbb{R}$ control the flow of energy between the virtual storage s and the robot. To ensure that s remains positive and bounded above, α and β_s should satisfy the constraints listed in Table I.

The proposed controller is as follows:

$$\begin{aligned} \boldsymbol{\tau}_c &= \mathbf{g}(\boldsymbol{\xi}) - \mathbf{D}(\boldsymbol{\xi})(\dot{\boldsymbol{\xi}} - \mathbf{f}_C(\boldsymbol{\xi}) - \beta_R(z,s)\mathbf{f}_R(\boldsymbol{\xi})) \\ &= \mathbf{g}(\boldsymbol{\xi}) - \mathbf{D}(\boldsymbol{\xi})\dot{\boldsymbol{\xi}} + \lambda_1\mathbf{f}_C(\boldsymbol{\xi}) + \beta_R(z,s)\lambda_1\mathbf{f}_R(\boldsymbol{\xi}) \end{aligned} \tag{8}$$

The last equality is due to the fact that $\mathbf{f}(\boldsymbol{\xi})$ is an eigenvector of $\mathbf{D}(\boldsymbol{\xi})$ as described in Section III. The scalar function $\beta_R(z,s)$ has the role of preventing non-passive control action when the virtual energy storage s is depleted, and should satisfy the conditions listed in Table I.

Consider the following storage function:

$$W(\boldsymbol{\xi},\dot{\boldsymbol{\xi}}) = \frac{1}{2}\dot{\boldsymbol{\xi}}^T\mathbf{M}(\boldsymbol{\xi})\dot{\boldsymbol{\xi}} + \lambda_1 V_C(\boldsymbol{\xi}) + s \tag{9}$$

Taking the time-derivative along the trajectories of (1) with control given by (8) yields:

$$\begin{aligned} \dot{W}(\boldsymbol{\xi},\dot{\boldsymbol{\xi}}) = \frac{1}{2}\dot{\boldsymbol{\xi}}^T(\dot{\mathbf{M}} - 2\mathbf{C})\dot{\boldsymbol{\xi}} - \dot{\boldsymbol{\xi}}^T\mathbf{D}\dot{\boldsymbol{\xi}} + \dot{\boldsymbol{\xi}}^T\boldsymbol{\tau}_e \\ +\lambda_1\dot{\boldsymbol{\xi}}^T\mathbf{f}_C(\boldsymbol{\xi}) + \lambda_1\nabla V_C^T\dot{\boldsymbol{\xi}} \\ +\beta_R(z,s)\lambda_1\mathbf{f}_R(\boldsymbol{\xi}) + \dot{s} + \boldsymbol{\tau}_e \end{aligned} \tag{10}$$

where dependencies of $\mathbf{M}, \mathbf{D}$ on $\boldsymbol{\xi}$ and $\mathbf{C}$ of $\dot{\boldsymbol{\xi}}$ have been omitted for cleanliness of notation. In Eq. (10) the first term is null due to the skew-symmetry of the matrix $\dot{\mathbf{M}} - 2\mathbf{C}$ and the terms on the second line cancel because $\mathbf{f}_C(\boldsymbol{\xi}) = -\nabla V_C$. Substituting $\dot{s}$ from Eq. (7) then yields:

$$\begin{aligned} \dot{W}(\boldsymbol{\xi},\dot{\boldsymbol{\xi}}) = -(1-\alpha(s))\dot{\boldsymbol{\xi}}^T\mathbf{D}\dot{\boldsymbol{\xi}} \\ +(\beta_R(z,s) - \beta_s(z,s))\lambda_1\mathbf{f}_R(\boldsymbol{\xi}) + \dot{\boldsymbol{\xi}}^T\boldsymbol{\tau}_e \end{aligned} \tag{11}$$

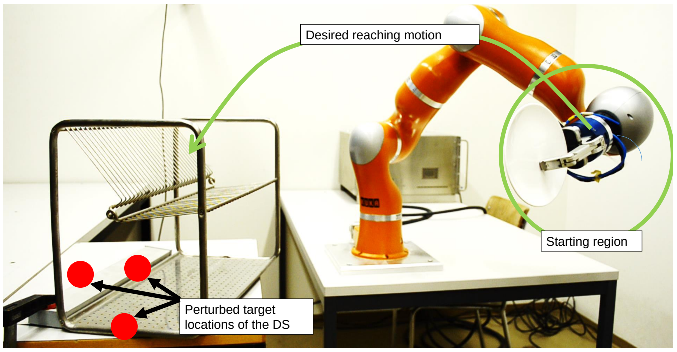
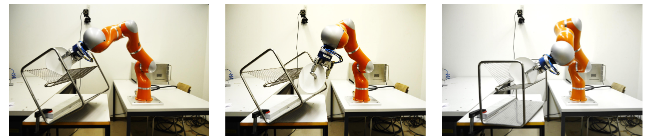


Fig. 2. The figure shows the experiment setup for the plate insertion task. The motion starts somewhere in the encircled region. The DS is describing a parapolic motion that has an atractor in one of three different perturbed target locations whose approximate position is shown with red dots.



(a) (b) (c)

Fig. 3. **Left, Middle:** Examples of final configurations for controller B. **Right:** Typical final configuration for controller A.

Note that by the conditions in Table I, we have:

1) Boundedness of s:

$$0 \leq s(0) \leq \overline{s} \Rightarrow 0 \leq s(t) \leq \overline{s} \quad \forall t > 0$$

2) Dissipation:

$$(1 - \alpha(s)) \geq 0$$

3) Limitation of the non-conservative control effort if the storage is depleted:

$$\beta_R(z, s) - \beta_s(z, s) = 0 \quad \forall z > 0$$

which proves passivity of the proposed control scheme.

The specifications of the functions α, β_s, β_R allow some freedom in the design. In our experiment, we used smooth step functions using fifth order polynomials. The open parameters of the controller are the eigenvalues of the damping matrix $\lambda_1 \ldots \lambda_N$ and the upper bound of the virtual storage $\overline{s}$. The first eigenvalue λ_1 determines the feedback gain for tracking the desired velocity and the remaining eigenvalues determine the resistance that the robot will exhibit when physically perturbed in directions orthogonal to the desired motion.

V. Robot Experiment

We consider a task of inserting a plate into a dishrack with perturbed location. A DS describing the task was learned by demonstration using Locally Modulated Dynamical Systems (LMDS). The motion is a parabolic reaching motion, see Fig. 2. Details on LMDS and the particular DS for this task can be found in [5]. The experiments were carried out on a KUKA LWR 4+ robot using the Fast Research Interface (FRI).

A. Experimental setup

The task DS has a single attractor to which all trajectories will converge. Ideally, this attractor would be placed exactly in the slot where the plate should be placed. In real scenarios, mismatch between environment state and the expected state is unavoidable. To account for this, we conduced three sets of task executions, in each of which the target location of the task DS was offset in different locations behind the real location of the dish rack, see Fig. 2. In each set of experiments, 5 task executions were carried out using two different controllers described below.

A. The controller from Section IV with $\overline{s} = s(0) = 10$, $\lambda_1 = 20$ and $\lambda_2 = \lambda_3 = 200$ is used. The value of λ_1 was chosen to the minimal value capable of overcoming static joint friction at the point of departure.

B. Openloop trajectory integrated from the initial position of the robot in combination with a simple impedance controller without inertia shaping. The stiffness was set to $\mathbf{K} = k\mathbf{I}_{3\times3}$ with $k = 100$, the minimum value capable of reaching the final point of the task in free motion.

All task executions were started somewhere in a small region shown in Fig. 2. Rotational motion of the end-effector was in both cases simply damped by a high amount (4 Ns/rad) which effectively kept the orientation constant during the task execution.

B. Results

Since very low gains were used, and no inverse dynamics control was applied, it is not expected that either controller would be able to track the nominal motion given by $\mathbf{f}$ with good accuracy. This is confirmed in Figures 4a,4b and 4c which plot the nominal and actual trajectories for each setup for each perturbed location of the dish rack. Note especially the 'shortcut' tendency of the impedance controller. The proposed controller has a clear advantage in terms of respecting shape of the desired reaching motion. In this particular task, the shortcut effect meant that the robot was approaching the rack from the wrong direction, which sometimes lead to interesting final configurations as depicted in Figures 3a and 3b. In each of the three perturbed scenarios, controller A consistently placed the plate correctly because the pattern of approach was respected, see Fig. 3c.

As is clear from Figures 4d,4e and 4f, controller A also has an advantage over controller B in terms of contact force after impact. At the time of impact, the reference point for controller B has already reached its final point, which is why there is no gradual ramp-up of the contact force as would normally be expected in contact with a timed trajectory. In the second perturbed location (Figures 4b and 4e) controller B resulted in some of the trials landing in a final configuration on the rack and some of the trials landed in a configuration under the rack. This is visible in the divergence of the trajectories near the end-point in Fig. 4b right, and also the high variance in the final contact force in Fig. 4e right. It should be emphasized that both controllers have been chosen to be as compliant as possible

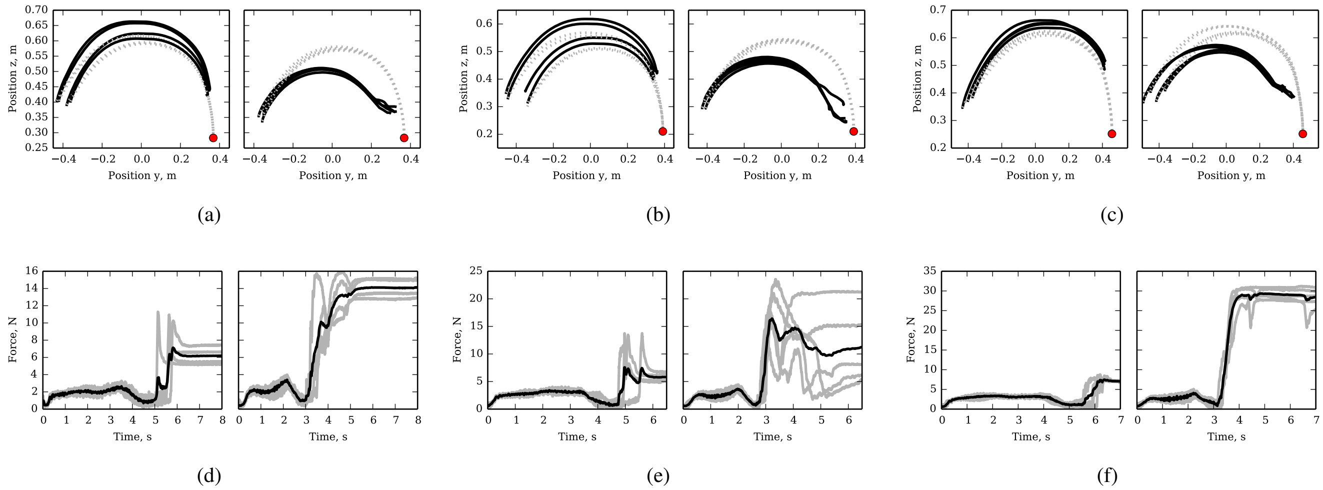


Fig. 4. **Top:** Actual (solid lines) and nominal (dotted lines) trajectories for the dish rack experiment. Figures **a,b,c** show the results of the three different perturbed scenarios, in each case with controller A on the left plot and controller B on the right plot. The motion in the YZ-plane is shown since the position in X direction remains almost constant during the motion. **Bottom:** The plots show the norm of the external force over time. The raw data are plotted in gray and a temporal average is plotted in black. Figures **d,e,f** show the data from each of the three perturbed scenarios with controller A on the left plot and controller B on the right plot. Due to imperfect estimation of the external force from the torque sensors of the robot, there is a small force even before impact with the dishrack.

for this task, but the low stiffness is not enough to ensure a low contact force for positioning errors of this magnitude.

VI. Conclusion

We have proposed a controller allowing to use the full modeling power of DS task representations at task execution time by closing the loop so that the DS is continuously updated with the actual state of the robot. We demonstrated the approach in a robotic manipulation task with unexpected collisions and showed the advantage over the classical approach of integrating a reference trajectory and following it with an impedance controller. We believe that the passivity (which ensures to some extent safe human-robot interaction), and the reactivity of closed loop DS make the proposed control scheme ideal for object hand-over scenarios and we will look specifically at this application in future research.

The results from this initial study are encouraging and show great potential for using closed loop DS in interaction control scenarios. The impedance of the robot is determined 1) by the DS and 2) by the choice of the eigenvalues of the damping matrix. An interesting perspective of the proposed controller is that the *resistance* to a perturbation can be tuned independently of the *recovery* from a perturbation. In impedance control, both of these are essentially determined by the stiffness term and are hence identical: if a force tries to push the robot off its reference trajectory it will be opposed, and when the perturbing force is released the robot will return to the reference trajectory. This behavior is just a special case of what is possible with the proposed controller. For example, the proposed controller would also oppose a force that moves the robot perpendicular to its desired velocity. But after the force is released, the robot may resume the task along a different path.

Acknowledgments

This research was funded by the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement no 288533 ROBOHOW.COG.

References

[1] Ed Colgate and Neville Hogan. Robust control of dynamically interacting systems. *International Journal on Control*, 48(1):65–88, 1988.

[2] V. Duindam, S. Stramigioli, and J.M.a. Scherpen. Passive Compensation of Nonlinear Robot Dynamics. *IEEE Transactions on Robotics and Automation*, 20(3):480–487, June 2004. ISSN 1042-296X. doi: 10.1109/TRA.2004.824693.

[3] Federica Ferraguti, Cristian Secchi, and Cesare Fantuzzi. A tank-based approach to impedance control with variable stiffness. In *2013 IEEE International Conference on Robotics and Automation*, pages 4948–4953. Ieee, May 2013. ISBN 978-1-4673-5643-5. doi: 10.1109/ICRA.2013.6631284.

[4] N. Hogan. Impedance control: An approach to manipulation. *Journal of Dynamic Systems Measurement and Control*, 107/17(March):304–313, 1985.

[5] K. Kronander, S.M. Khansari-Zadeh, and A. Billard. Incremental Motion Learning with Locally Modulated Dynamical Systems. *Robotics and Autonomous Systems*, 2015.

[6] P.Y. Li and R. Horowitz. Passive velocity field control of mechanical manipulators. *IEEE Transactions on Robotics and Automation*, 15(4): 751–763, 1999. ISSN 1042296X. doi: 10.1109/70.782030.

[7] Stefano Stramigioli, Cristian Secchi, Arjan J. van der Schaft, and Cesare Fantuzzi. Sampled data systems passivity and discrete port-Hamiltonian systems. *IEEE Transactions on Robotics*, 21:574–587, 2005. ISSN 15523098. doi: 10.1109/TRO.2004.842330.