Modeling Compositions of Impedance-based Primitives via Dynamical Systems

Nadia Figueroa and Aude Billard

***Abstract*— In this work, we introduce a novel Dynamical System (DS)-based approach for modeling complex and compliant manipulation tasks, that are composed of a sequence of action phases with different compliance requirements; i.e. *impedance primitives*. We adopt a closed-loop (DS)-based control architecture and present the Locally Active Globally Stable (LAGS)-DS formulation. In LAGS-DS we seek to model the whole task as a *globally asymptotically stable* DS that has locally task-varying dynamics and smoothly transit between them. These locally task-varying dynamics represent the set of impedance primitives, hence, rather than modeling the task as a *discretization* of impedance primitives, we model it as a *composition* of impedance primitives in a single DS-based controller. In this paper, we present the theoretical background for this novel DS, briefly describe the learning approach and provide 2D simulations of LAGS-DS learned from toy data.**

I. Introduction

In this work, we seek to tackle the problem of modeling complex and compliant manipulation tasks from demonstrations. A complex manipulation task, is one that can be decomposed into a sequence of action phases that are executed to reach a high-level goal. On the other hand, a compliant manipulation task is one that deals with controlling for interaction between the robot and the environment, be it an obstacle, a surface or a perturbation from a human. Thus, a complex and compliant manipulation task is one that has not only more than one action phase but also requires different levels of compliance for each phase, as shown in Fig. 1. These illustrations show examples of complex compliant tasks with a point-mass (representing the end-effector of a robot), a sample trajectory and a final target (the star). The shaded areas indicate the phases in each task where the robot must precisely follow the reference trajectory. Beside these illustrations we show the targeted applications that exhibit this type of compliant behavior.

Learning this type of complex and compliant behavior is a key challenge in the robotics community as it is exhibited in many real-world manipulation tasks, such as: (1) reaching-for/grasping/lifting an object, (2) wiping a surface, (3) sliding on a surface, (4) painting a surface, (5) peeling vegetables, among others. Our goal is thus to propose a unified framework for learning such complex and compliant manipulation tasks from demonstrations while reproducing them in a passive and reactive manner. Previous works in this domain generally follow a discretization approach. Namely, the complete task is initially segmented into discrete point-to-point action phases [1], [2], [3]. Each action phase is then represented by a corresponding motion generator, together with a learned impedance/force profile if the action phase requires it, as is done in [4], [5], [6] where the motion generators are represented by time-invariant Dynamical Systems (DS). Finally, to execute the complete task one then uses a switching architecture to parametrize an impedance (or hybrid force/impedance) controller [7] with the corresponding motion generator and stiffness/force requirements of each action phase. An alternative to this approach is to learn time/phase-dependent profiles that adapt across the task for a given time-dependent reference trajectory as in [8], [9]. However, due to their ability to generate on-line motion plans robust to changes in dynamic environments, in this work we focus on motion generators represented as a first-order, autonomous DS describing a nominal motion plan, such that,

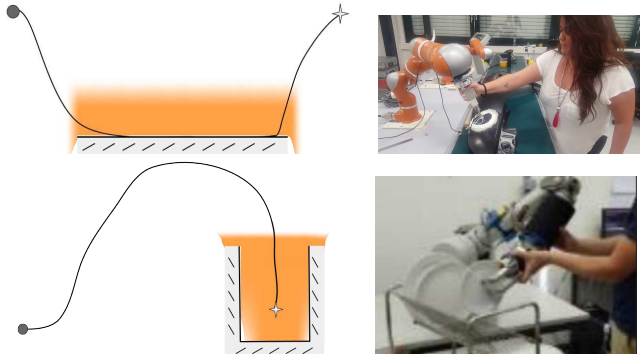

Fig. 1: Examples of *Complex Compliant Manipulation* tasks. **(left column)** 2D Illustrations of a **Point** that represents the end-effector of the robot. The **star** is the final target, while gray areas represent surfaces/obstacles. The background shading indicates different **levels of compliance** in the task, lighter being more compliant and viceversa. **(right column)** Corresponding real-world applications; i.e. wiping a surface or inserting objects in a rack.

$$\dot{\boldsymbol{\xi}} = \mathbf{f}(\boldsymbol{\xi}) \rightarrow \quad \Big\{ \lim_{t\to\infty} \|\boldsymbol{\xi} - \boldsymbol{\xi}^*\| = 0 \tag{1}$$

where $\mathbf{f}(\cdot) : \mathbb{R}^M \rightarrow \mathbb{R}^M$ is a continuous differentiable vector-valued function representing a non-linear DS that converges to a single stable equilibrium point $\boldsymbol{\xi}^*$; i.e. target or attractor. Historically, the use DS to represent motion plans for discrete action phases is referred to as *motion primitives* [10]. In this work, we extend the notion of *motion primitives* to *impedance primitives* which refer to these independent DS with corresponding/changing impedance requirements.

This current approach of switching across impedance primitives, although successful in autonomously executing the complete task, cannot ensure passivity and reactivity

*This research was funded by the European Commission via the H2020 project *COGIMON* $H2020 - ICT - 23 - 2014$

†Nadia Figueroa and Aude Billard are both with the Learning Algorithms and Systems Laboratory (LASA), École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland - 1015. {nadia.figueroafernandez,aude.billard}@epfl.ch

if the human decides to interfere or aid/modify the task. This is due to the fact that the individual DS encoded for each action phase are simply used as open-loop motion generators. To provide such interactive properties during execution we refrain from classical impedance control and instead adopt a closed-loop dynamical system (DS)-based control architecture, as the one proposed in [11], which we will herein refer to as the *passive-DS* controller. This controller combines the advantages of impedance control and a passive control system while avoiding any time-dependent reference trajectory or profile. This is achieved by encoding the nominal motion plan through an autonomous DS and allowing for varying damping through a passive velocity feedback control law. This control law ensures passivity and is formulated in such a way that one can intuitively tune the desired robot impedance.

Interestingly, if one is capable of encoding the set of impedance primitives in a single DS, through the passive-DS control architecture, we can avoid switching altogether. This paper focuses on designing such a dynamical system which we currently refer to as the Locally Active Globally Stable (LAGS)-DS. In LAGS-DS we seek to model the whole task as a *globally asymptotically stable* DS that has locally task-varying dynamics and smoothly transit between them. These locally task-varying dynamics represent the set of impedance primitives, hence, rather than modeling the task as a *discretization* of impedance primitives, we model it as a *composition* of impedance primitives in a single DS-based controller.

II. Preliminaries

A. Robot Rigid Body Dynamics

We begin by assuming that the physics of an N-DOF manipulator is accurately described by the rigid-body form:

$$\mathbf{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \mathbf{B}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \mathbf{g}(\boldsymbol{q}) = \boldsymbol{\tau_c} + \boldsymbol{\tau_e} \tag{2}$$

where $\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}} \in \mathbb{R}^N$ represent joint positions, velocities and accelerations. $\mathbf{M}(\boldsymbol{q}) \in \mathbb{R}^{N\times N}$ corresponds to inertia and force matrices and $\mathbf{B}(\boldsymbol{q}, \dot{\boldsymbol{q}}), \mathbf{g}(\mathbf{q}) \in \mathbb{R}^N$ correspond to th Coriolis/centrifugal and the gravitational force vector, respectively. Finally, $\boldsymbol{\tau_c}, \boldsymbol{\tau_e} \in \mathbb{R}^N$ indicate the control and external torques in joint-space, respectively. Following [12], (2) can be re-formulated in Cartesian task-space coordinates:

$$\mathbf{M}_{\boldsymbol{\xi}}(\boldsymbol{\xi})\ddot{\boldsymbol{\xi}} + \mathbf{B}_{\boldsymbol{\xi}}(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}) + \mathbf{g}_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \boldsymbol{u_c} + \boldsymbol{u_e} \tag{3}$$

where $\boldsymbol{\xi} \in \mathbb{R}^M$ is a generalized kinematic state variable in task-space[1], consequently $\dot{\boldsymbol{\xi}}, \ddot{\boldsymbol{\xi}} \in \mathbb{R}^M$ indicate their velocities and accelerations. The inertia matrix in task-space $\mathbf{M}(\boldsymbol{\xi}) \in \mathbb{R}^{M\times M}$ is given by $\mathbf{M}_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \mathbf{J}(\boldsymbol{q})^{-T}\mathbf{M}(\boldsymbol{q})\mathbf{J}(\boldsymbol{q})$ and the vector of centrifugal/coriolis and gravity force vectors $\mathbf{B}_{\boldsymbol{\xi}}(\cdot), \mathbf{g}_{\xi}(\boldsymbol{\xi}) \in \mathbb{R}^M$ are given by $\mathbf{B}_{\boldsymbol{\xi}}(\cdot) = \mathbf{J}(\boldsymbol{q})^{-T}\mathbf{B}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \mathbf{M}(\boldsymbol{\xi})\dot{\mathbf{J}}(\boldsymbol{q})\dot{\boldsymbol{q}}$ and $\mathbf{g}_{\xi}(\boldsymbol{\xi}) = \mathbf{J}(\boldsymbol{q})^{-T}\mathbf{g}(\boldsymbol{q})$; where $\mathbf{J}(\boldsymbol{q}) \in \mathbb{R}^{M\times N}$ represents the manipulators Jacobian which projects joint-space velocities to task-space as $\dot{\boldsymbol{\xi}} = \mathbf{J}(\boldsymbol{q})\dot{\boldsymbol{q}}$. Finally, the control and external forces, $\boldsymbol{u_e}, \boldsymbol{u_c} \in \mathbb{R}^M$, are given by the desired task-space controller and interaction forces, respectively. These are transposed to joint-space with the following relationships $\boldsymbol{\tau}_c = \mathbf{J}(\boldsymbol{q})^T\boldsymbol{u_c}$ and $\boldsymbol{\tau}_e = \mathbf{J}(\boldsymbol{q})^T\boldsymbol{u_e}$.

[1] $M = 3$ for position or $M = 6$ full end-effector pose.

B. Passive-DS Controller

The *passive-DS* controller proposed in [11] generates a control signal $\boldsymbol{u_c}$ that tracks a desired velocity via the nominal motion plan $\mathbf{f}(\boldsymbol{\xi})$ while ensuring stable interaction with a passive environment. This is achieved through the following negative velocity error feedback control law:

$$\boldsymbol{u_c} = -\mathbf{D}(\boldsymbol{\xi})(\dot{\boldsymbol{\xi}} - \mathbf{f}(\boldsymbol{\xi})). \tag{4}$$

To analyze the closed loop dynamics of (3) under control (4), we begin by expanding the right-hand side of (3) as follows,

$$\mathbf{M}_{\boldsymbol{\xi}}(\boldsymbol{\xi})\ddot{\boldsymbol{\xi}} + \mathbf{B}_{\boldsymbol{\xi}}(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}) + \mathbf{g}_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \boldsymbol{u_c} + \boldsymbol{u_i} + \boldsymbol{u_e} \tag{5}$$

where $\boldsymbol{u_i} \in \mathbb{R}^M$ is a vector of inverse dynamics forces. Without loss of generality, by assuming that $\boldsymbol{u_i}$ compensates for centrifugal and gravity forces; i.e. $\boldsymbol{u_i} = \mathbf{B}_{\boldsymbol{\xi}}(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}) + \mathbf{g}_{\boldsymbol{\xi}}(\boldsymbol{\xi})$, the closed-loop dynamics of the controlled system are,

$$\mathbf{M}_{\boldsymbol{\xi}}(\boldsymbol{\xi})\ddot{\boldsymbol{\xi}} + \mathbf{D}(\boldsymbol{\xi})(\dot{\boldsymbol{\xi}} - \mathbf{f}(\boldsymbol{\xi})) = \boldsymbol{u_e} \tag{6}$$

This resulting closed-loop system is passive with respect to the input-output pair $(\boldsymbol{u_e}, \dot{\boldsymbol{\xi}})$ and can track the desired motion plan $\mathbf{f}(\boldsymbol{\xi})$ while dissipating kinetic energy in directions orthogonal to it. Following the derivations and proofs in [11], these properties hold, if the following conditions are met:

1) The state-varying damping matrix should be positive semi-definite and is defined as,

$$\mathbf{D}(\boldsymbol{\xi}) = \mathbf{Q}(\boldsymbol{\xi})\boldsymbol{\Lambda}\mathbf{Q}(\boldsymbol{\xi})^T \tag{7}$$

where $\boldsymbol{\Lambda} \in \mathbb{R}^{M\times M}$ is a diagonal matrix of non-negative values and $\mathbf{Q}(\boldsymbol{\xi}) \in \mathbb{R}^{M\times M}$ is a matrix whose columns correspond to the vectors in an orthonormal basis $\boldsymbol{E} = \{\boldsymbol{e_1}, \cdots, \boldsymbol{e_M}\}$ for $\boldsymbol{e_i} \in \mathbb{R}^M$, which is constructed such that $\boldsymbol{e_1} = \frac{\mathbf{f}(\boldsymbol{\xi})}{||\mathbf{f}(\boldsymbol{\xi})||}$ follows the direction of the desired motion and the remaining elements in the set are mutually orthogonal and normalized vectors.

2) The motion plan $\mathbf{f}(\boldsymbol{\xi})$ must have a conservative component; i.e. $\mathbf{f}_C(\boldsymbol{\xi}) = -\nabla V_f(\boldsymbol{\xi})$ where $\mathbf{f}(\boldsymbol{\xi}) = \mathbf{f}_C(\boldsymbol{\xi}) + \mathbf{f}_{\bar{C}}(\boldsymbol{\xi})$ and $\mathbf{f}_{\bar{C}}(\boldsymbol{\xi})$ is the non-conservative component.

C. Compliance Tuning with Passive-DS

Following we relate the passive-DS control-law to the classical impedance control terms.

1) On Damping: The apparent damping $\mathbf{D}_a(\boldsymbol{\xi}) \in \mathbb{R}^{M\times M}$ of the controlled system (6) can be computed as,

$$\mathbf{D}_a(\boldsymbol{\xi}) = \frac{\partial \boldsymbol{u}_e}{\partial \dot{\boldsymbol{\xi}}} = \mathbf{D}(\boldsymbol{\xi}) \tag{8}$$

As can be seen, the apparent damping $\mathbf{D}_a(\boldsymbol{\xi})$ is equivalent to the damping term used in our control law (4), as in the classical impedance control formulation [7]. Hence, the diagonal elements of $\boldsymbol{\Lambda}$ in (7), i.e. the eigenvalues $\lambda_1, \ldots, \lambda_M$ are the desired damping values in all possible directions of motion.

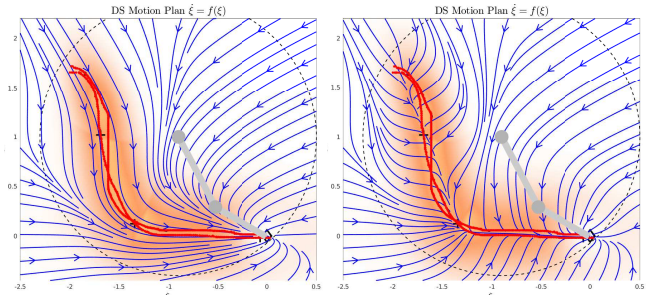

Fig. 2: Examples of a DS motion plan with **(left)** no stiffness-like behavior [simulation-link-1] and with **(right)** stiffness-like behavior around the reference trajectory. [simulation-link-2]

Via the construction of $\mathbf{D}(\boldsymbol{\xi})$ defined in the first condition of Sec. II-B, (4) becomes:

$$\boldsymbol{u_c} = -\mathbf{D}(\boldsymbol{\xi})\dot{\boldsymbol{\xi}} + \lambda_1 \mathbf{f}(\boldsymbol{\xi}), \tag{9}$$

where λ_1 corresponds to the first eigenvalue of $\mathbf{D}(\boldsymbol{\xi})$ and acts as a control gain for the desired motion, while the remaining eigenvalues represent how much deviation from the desired motion plan $\mathbf{f}(\boldsymbol{\xi})$ is allowed when the robotic system is subject to external forces coming from the environment or a human interacting with it. The equivalence of (9) to (4) results from the fact that $\mathbf{f}(\boldsymbol{\xi})$ is the first eigenvector of $\mathbf{D}(\boldsymbol{\xi})$.

2) On Stiffness: The apparent stiffness $\mathbf{K}_a(\boldsymbol{\xi}) \in \mathbb{R}^{M\times M}$ of the controlled system (6) has a less straightforward manifestation than the damping term as it is computed by,

$$\mathbf{K}_a(\boldsymbol{\xi}) = \frac{\partial \boldsymbol{u}_e}{\partial \boldsymbol{\xi}} = -\mathbf{D}(\boldsymbol{\xi})\frac{\partial \mathbf{f}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}}. \tag{10}$$

As shown, the classical notion of stiffness in our controller is dependent not only on the damping term $\mathbf{D}(\boldsymbol{\xi})$, but also on the properties of the DS, specifically its convergence rate represented by the Jacobian $\frac{\partial \mathbf{f}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}}$. As derived in [13], the stiffness in a particular direction $\mathbf{K}_{a|s}(\boldsymbol{\xi}, \boldsymbol{\xi}_s)$, i.e. at state $\boldsymbol{\xi}$ with unit norm direction $\boldsymbol{\xi}_s$, can be estimated as follows,

$$\mathbf{K}_{a|s}(\boldsymbol{\xi}, \boldsymbol{\xi}_s) = \boldsymbol{\xi}_s^T \mathbf{K}_a(\boldsymbol{\xi})\boldsymbol{\xi}_s = -\mathbf{D}(\boldsymbol{\xi})\boldsymbol{\xi}_s^T \frac{\partial \mathbf{f}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}}\boldsymbol{\xi}_s. \tag{11}$$

To summarize, the desired impedance behavior is encoded through the state-dependent continuous function $\mathbf{D}(\boldsymbol{\xi}) \in \mathbb{R}^{M\times M}$ which defines the desired damping of the robot while executing the DS $\mathbf{f}(\boldsymbol{\xi})$. This is shown in the (left) illustration of Fig. 2 and the provided simulation video, where the passive-DS control architecture is being executed with a DS $\mathbf{f}(\boldsymbol{\xi})$ shaped from a reference trajectory (red lines). Such parametrization, however, is not sufficient to encode a symmetrically converging behavior along the reference trajectory, as in the classical notion of stiffness attraction. To encapsulate such converging behavior, it should be encoded within the DS itself, as shown in the (right) illustration of Fig. 2 and the provided simulation video. This type of DS representation is seldom in literature. Hence, in this paper we introduce LAGS-DS, a flexible DS representation that can model and smoothly transition between different compliant behaviors around the reference trajectories provided from demonstrations.

III. Locally Active Globally Stable Dynamical Systems (LAGS-DS)

Most state-of-the-art autonomous DS estimation approaches focus on solving the following problem: *"Given a set of reference trajectories* $\{\boldsymbol{\Xi}, \dot{\boldsymbol{\Xi}}\} = \{\boldsymbol{\xi}_t^{ref}, \dot{\boldsymbol{\xi}}_t^{ref}\}_{t=1\dots T_N}$ *estimate the parameters of the non-linear function* (1) *such that it captures the invariant features of the provided trajectories, and is capable of generating motions that resemble them, while ensuring that the target* $\boldsymbol{\xi}^*$ *will always be reached."* This can be summarized into two objectives: (i) mimicking the motion pattern and (ii) converging to the attractor, as shown in the *left* illustration of Fig. 2. Assume now that we have a third objective; i.e. (iii) symmetrically converging to the reference trajectory (or trajectories), as shown in the *right* illustration of Fig. 2. This DS exhibits a behavior inside the shaded regions which is qualitatively similar to a *stiffness attraction* around a local reference trajectory. Once the local attractor is reached it either converges to another reference trajectory or ultimately to the global attractor. We thus assume that the over-arching dynamical behavior of the system is to (i) converge to the global attractor $\boldsymbol{\xi}^*$, while (ii) *globally* mimicking the motion pattern and (iii) *locally* converging to the reference trajectories if desired.

In this paper, we propose a DS formulation and estimation approach that is capable of generating such *locally active* behaviors while ensuring *global asymptotic stability*. We refer to this novel DS as the Locally Active Globally Stable - DS (LAGS-DS). The main idea behind LAGS-DS is: *"Let* $\mathbf{f}_g$ *be a global/nominal DS which should strictly converge to the global attractor* $\boldsymbol{\xi}_g^*$*, as shown in Fig. 3 (top-left). We also have a set of local dynamics* $\mathbf{f}_l^i$ *for* $i = 1, \dots, K$*, as the ones shown in Fig. 3 (right column), that exhibit a specific trajectory tracking behavior around a local attractor* $\boldsymbol{\xi}_i^* \neq \boldsymbol{\xi}_g^*$*. Finally, we have a set of local activation regions (the orange shaded areas that smoothly decay as they move away from the reference trajectories), indicating where the local DS* $\mathbf{f}_l^i$ *should be active. The goal of a LAGS-DS is to evolve according to* $\mathbf{f}_g$ *where the local activation regions are inactive. In the regions where a local activation region is active, rather than following the integral curves from* $\mathbf{f}_g$ *the state evolves according to the locally active* $\mathbf{f}_l^i$*. In the absence of perturbations, if the state is in a locally active region it will reach the local attractor* $\boldsymbol{\xi}_i^*$ *and then transit back to* $\mathbf{f}_g$ *or to another local DS, ultimately reaching the global attractor* $\boldsymbol{\xi}_g^*$*."* In order to achieve this desiderata, we propose the following LAGS-DS formulation:

$$\dot{\boldsymbol{\xi}} = \underbrace{\alpha(\boldsymbol{\xi})\mathbf{f}_g(\boldsymbol{\xi})}_{\substack{\text{Global Dynamics} \\ \lim_{t\to\infty}\|\boldsymbol{\xi}-\boldsymbol{\xi}_g^*\|=0}} + \underbrace{(1-\alpha(\boldsymbol{\xi}))\sum_{k=1}^{K}\gamma_k(\boldsymbol{\xi})\mathbf{f}_l^k(h_k(\boldsymbol{\xi}),\boldsymbol{\xi})}_{\substack{\text{Set of } K \text{ Local Dynamics} \\ \lim_{t\to\infty}\|\boldsymbol{\xi}-\boldsymbol{\xi}_k^*\|=0 \text{ if } h_k(\boldsymbol{\xi}) \geq 1 \\ \lim_{t\to\infty}\|\boldsymbol{\xi}\|=\infty \text{ if } h_k(\boldsymbol{\xi}) < 1}} \tag{12}$$

where $\mathbf{f}_g(\boldsymbol{\xi})$ is a globally asymptotically stable dynamical system that converges towards the global attractor $\boldsymbol{\xi}_g^*$ and $\mathbf{f}_l^k(\boldsymbol{\xi})$ represent the k-th local DS with local attractors $\boldsymbol{\xi}_k^*$

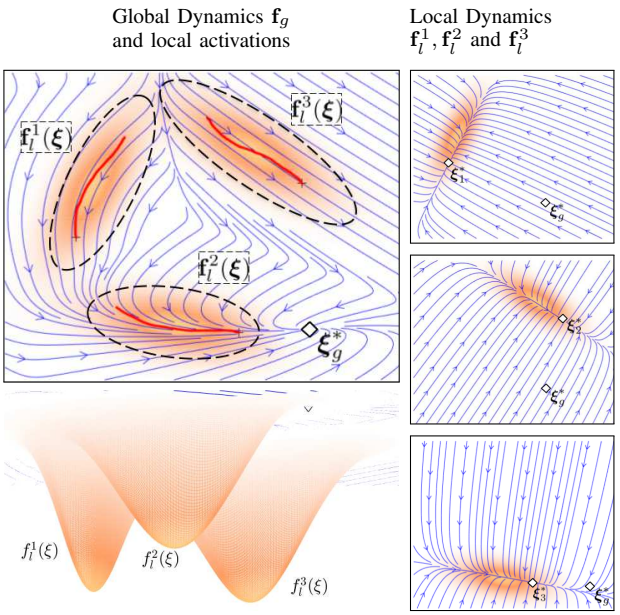

Fig. 3: Illustration for constructing the Locally Active Globally Stable (LAGS)-DS. The **top-left** plot shows a Globally Asymptotically Stable DS. The shaded regions represent the locally-active regions where the system should follow some *local dynamics*. The **right column** shows the 3 local dynamics corresponding to each region.

that symmetrically converge towards the local reference trajectory. Following we describe the properties and roles of the remaining terms in (12).

*3) **Global Dynamics**:* The global DS $\mathbf{f}_g$ in (12) is a non-linear DS that converges to the global attractor $\boldsymbol{\xi}_g^*$, while approximating the motion pattern generalized from the reference trajectories. It is defined as a weighted combination of linear DS with the following form:

$$\mathbf{f}_g(\boldsymbol{\xi}) = \sum_{k=1}^{K} \gamma_k(\boldsymbol{\xi})(\mathbf{A}_g^k \boldsymbol{\xi} + \boldsymbol{b}_g^k) \tag{13}$$

where $\mathbf{A}_g^k \in \mathbb{R}^{M\times M}$ and $\boldsymbol{b}_g^k \in \mathbb{R}^M$ are the individual linear systems matrices and bias terms and $\gamma_k(\boldsymbol{\xi})$ is a state-dependent mixing function. To ensure that (13) is globally asymptotically stable at $\boldsymbol{\xi}_g^*$, the bias terms are defined as $\boldsymbol{b}_g^k = -\mathbf{A}_g^k \boldsymbol{\xi}_g^*$ and the system matrices should meet the following condition: $(\mathbf{A}_g^k)^T \mathbf{P}_g + \mathbf{P}_g \mathbf{A}_g^k \prec 0$ for a symmetric positive-definite $\mathbf{P}_g \in \mathbb{R}^{M\times M}$ matrix; and the mixing function must be $0 < \gamma_k(\boldsymbol{\xi}) \leq 1$ and $\sum_{k=1}^{K} \gamma_k(\boldsymbol{\xi}) = 1$. These conditions are derived from Lyapunov's stability theorem, detailed proof is provided in [14].

*4) **Activation Function**:* As depicted in Fig. 4, the activation function $\alpha(\boldsymbol{\xi})$ indicates where the global and local dynamics should be activated. The local dynamics are activated when $\alpha(\boldsymbol{\xi}) < 1$, which as shown, this only happens in the compact sets $\boldsymbol{\chi} = \chi^1 \cup \ldots \cup \chi^K$ for $\chi^k \subset \mathbb{R}^M$. When $0 < \alpha(\boldsymbol{\xi}) < 1$ a mixture of both local and global dynamics are activated, while when $0 = \alpha(\boldsymbol{\xi})$ solely the local dynamics are active. On the other hand, in the converging region $\mathcal{C} \subset \mathbb{R}^M$ and the ball $\mathcal{B}_r$ of radius r centered at $\boldsymbol{\xi}_g^*$ the global dynamics are active; i.e. $\alpha(\boldsymbol{\xi}) = 1$, which should

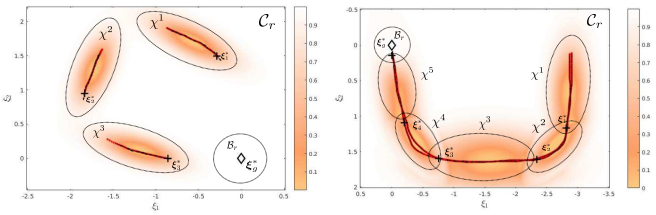

Fig. 4: State-space partitioning examples for activation function $\boldsymbol{\alpha}(\boldsymbol{\xi})$ (15)

asymptotically converge to the global attractor $\boldsymbol{\xi}_g^*$. Since we assume that each k-th locally active region is represented by a compact set $\chi^k \subset \mathbb{R}^M$. $\boldsymbol{\chi}$ can be represented by a Gaussian Mixture Models (GMM), where each k-th Gaussian distribution corresponds to each local compact set χ^k as,

$$p(\boldsymbol{\xi}|\theta_\gamma) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{\xi}|\boldsymbol{\mu}^k, \boldsymbol{\Sigma}^k) \tag{14}$$

where $\boldsymbol{\mu}^k$ and $\boldsymbol{\Sigma}^k$ are the mean and Covariance of the k-th Gaussian distribution $\mathcal{N}(\boldsymbol{\xi}|\boldsymbol{\mu}^k, \boldsymbol{\Sigma}^k)$ that represents the k-th compact set χ^k. The complete set of parameters $\theta_\gamma = \{\pi_k, \boldsymbol{\mu}^k, \boldsymbol{\Sigma}^k\}_{k=1}^K$, where π_k are the priors (or mixing weights) of each Gaussian component, satisfying the constraint $\sum_{k=1}^{K} \pi_k = 1$. Hence, we parametrize the $\alpha(\boldsymbol{\xi})$ as:

$$\alpha(\boldsymbol{\xi}) = \Big(1 - r(\boldsymbol{\xi})\Big)\left(1 - \frac{\mathcal{N}(\boldsymbol{\xi}|\boldsymbol{\mu}^{k^*}, \tilde{\boldsymbol{\Sigma}}^{k^*})}{\mathcal{N}(\boldsymbol{\mu}^{k^*}|\boldsymbol{\mu}^{k^*}, \tilde{\boldsymbol{\Sigma}}^{k^*})}\right) + r(\boldsymbol{\xi})$$

where

$$k^* = \arg\max_k \{p(k|\boldsymbol{\xi}, \theta_\gamma)\} \quad \text{for} \quad k^* \in [1, \ldots, K] \tag{15}$$

The parameters $\theta_\gamma = \{\pi_k, \boldsymbol{\mu}^k, \boldsymbol{\Sigma}^k\}_{k=1}^K$ used for (15) are also used in (13) to parametrize $\gamma_k(\boldsymbol{\xi})$ with the *a posteriori* probability for a k-th component; i.e. $\gamma(\boldsymbol{\xi}) = p(k|\boldsymbol{\xi}, \theta_\gamma)$, this defines the contribution of each linear DS in (13). $\tilde{\boldsymbol{\Sigma}}$ is a scaled Covariance matrix of the reference trajectory. Moreover, by normalizing the Gaussian distribution with its maximum value, we are flattening the probability values on the reference trajectory such that regions where $\boldsymbol{\alpha}(\boldsymbol{\xi}) \approx 0$ are wider within the compact set χ. Finally, $r(\boldsymbol{\xi})$ is a radial exponential function centered at $\boldsymbol{\xi}_g^*$ as,

$$r(\boldsymbol{\xi}) = 1 - \exp\Big(-c||\boldsymbol{\xi} - \boldsymbol{\xi}_g^*||\Big) \tag{16}$$

where c is proportional to the radius r of $\mathcal{B}_r$. Intuitively, this function enforces $\boldsymbol{\alpha}(\boldsymbol{\xi}) = 1$ in the region within $\mathcal{B}_r$.

*5) **Local Dynamics**:* As shown in (12), the local DS component $\mathbf{f}_l(\cdot)$ is a combination of *local DS* as follows:

$$\mathbf{f}_l(\boldsymbol{\xi}) = \sum_{k=1}^{K} \gamma_k(\boldsymbol{\xi}) \mathbf{f}_l^k(h_k(\boldsymbol{\xi}), \boldsymbol{\xi}) \tag{17}$$

where each $\mathbf{f}_l^k(h_k(\boldsymbol{\xi}), \boldsymbol{\xi})$ denotes the dynamics that will induce the desired behavior in the locally active regions. Since χ^k encapsulates the local attractor $\boldsymbol{\xi}_k^*$, $\mathbf{f}_l^k(\cdot)$ must be designed such that in the composed system (12) the sole equilibrium point is $\boldsymbol{\xi}_g^*$. To ensure the vanishing of $\boldsymbol{\xi}_k^*$ as an equilibrium point in the composed system and

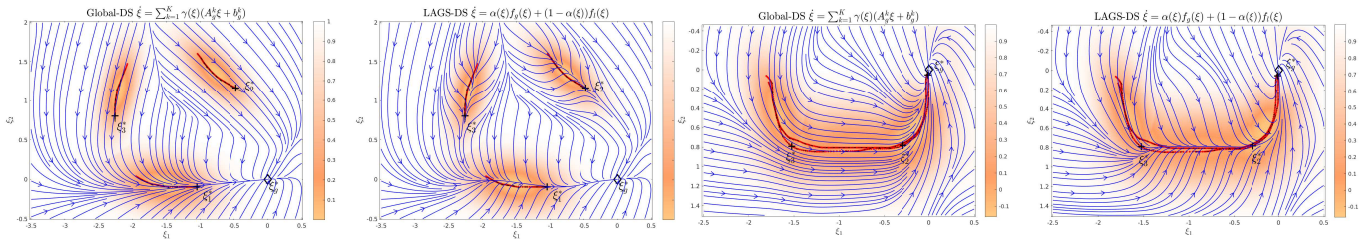

Fig. 5: Illustrative 2D Example for *non-linear* LAGS-DS Exposition. **(first)** *Non-linear* Global-DS **(second)** LAGS-DS with symmetrically converging local behaviors and *non-linear* Global-DS **(third)** *Non-linear* Global-DS **(fourth)** LAGS-DS with partial symmetrically converging local behaviors and *non-linear* Global-DS

provide a smooth transition between each χ^k and $\mathcal{C}$ and $\chi^k \rightarrow \chi^j \forall j \neq k$, we formulate $\mathbf{f}_l^k(\cdot)$ as a combination of a *locally active* and a *locally deflective* DS via a partitioning function $h_k(\boldsymbol{\xi})$. The *locally active* DS is defined as:

$$\mathbf{f}_{l,a}^k(\boldsymbol{\xi}) = \mathbf{A}_{l,a}^k \boldsymbol{\xi} + \boldsymbol{b}_{l,a}^k \tag{18}$$

where $\mathbf{A}_{l,a}^k \in \mathbb{R}^{M \times M}$ defines the desired local behavior in the compact set χ^k and $\boldsymbol{b}_{l,a}^k = -\mathbf{A}_{l,a}^k \boldsymbol{\xi}_k^*$ is the bias of the linear system. To provide the stiffness-like behavior of our desiderata; i.e. a *symmetrically converging DS* as in Fig. 3 we propose to parametrize $\mathbf{A}_{l,a}^k$ as follows,

$$\mathbf{A}_{l,a}^k = \mathbf{U}_{l,a}^k \mathbf{\Lambda}_{l,a}^k (\mathbf{U}_{l,a}^k)^T \tag{19}$$

$$\text{for} \quad \mathbf{U}_{l,a}^k = \begin{bmatrix} \overline{\boldsymbol{\xi}}_{\mathbf{0}}^{\,k} & \overline{\boldsymbol{\xi}}_{0\perp}^{\,k} \end{bmatrix}, \; \mathbf{\Lambda}_{l,a}^k = \begin{bmatrix} (\lambda_{l,a}^1)^k & 0 \\ 0 & (\lambda_{l,a}^2)^k \end{bmatrix} \tag{20}$$

where $\overline{\boldsymbol{\xi}}_{\mathbf{0}}^{\,k}, \overline{\boldsymbol{\xi}}_{0\perp}^{\,k} \in \mathbb{R}^M$ are orthonormal vectors indicating the direction of the reference trajectory; i.e. $(\overline{\boldsymbol{\xi}}_{\mathbf{0}}^{\,k})$ towards the local attractor $\boldsymbol{\xi}_k^*$ and the direction of convergence towards the reference trajectory $(\overline{\boldsymbol{\xi}}_{0\perp}^{\,k})$. To ensure such symmetric convergence to the reference trajectory, the eigenvalues must comply with the following condition $|\kappa(\lambda_{l,a}^2)^k| > |(\lambda_{l,a}^1)^k|$, where $\kappa \in \mathbb{R}_+$ is a positive value > 1 and indicates the "stiffness" of the DS around the reference trajectory. We can translate this notion of "stiffness" to the apparent stiffness of the robot, by directly plugging (18) into (10) yields,

$$\begin{aligned} \mathbf{K}_{a,\mathbf{f}_{l,a}^k}(\boldsymbol{\xi}) &= -\mathbf{D}(\boldsymbol{\xi}) \frac{\partial \mathbf{f}_{l,a}^k(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} \\ &= -\mathbf{D}(\boldsymbol{\xi}) \mathbf{U}_{l,a}^k \begin{bmatrix} (\lambda_{l,a}^1)^k & 0 \\ 0 & (\lambda_{l,a}^2)^k \end{bmatrix} (\mathbf{U}_{l,a}^k)^T \end{aligned} \tag{21}$$

Hence, under a perturbation, (18) will pull the robot towards the reference trajectory, in a spring-like manner with a stiffness proportional to the eigenvalues $\mathbf{\Lambda}_{l,a}^k$. Once the robot has reached the local attractor $\boldsymbol{\xi}_k^*$ after following the symmetrically converging DS, the *locally deflective* DS takes over, which ensures a transitioning to the next compact set χ^j for $j \neq k$ or to the converging region $\mathcal{C}$, defined as:

$$\mathbf{f}_{l,d}^k(\boldsymbol{\xi}) = \mathbf{A}_{l,d}^k \boldsymbol{\xi} + \boldsymbol{b}_{l,d}^k. \tag{22}$$

(22) is a repulsive DS centered at $\boldsymbol{\xi}_k^*$; i.e. $\mathbf{A}_{l,d}^k = \lambda_d \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ for $\lambda_d > 0$ and $\boldsymbol{b}_{l,d} = \mathbf{A}_{l,d}^k \boldsymbol{\xi}_k^*$. To combine (18) and (22) we use a partition function $h_k(\boldsymbol{\xi})$ to indicate in which region of the state-space each DS belongs to. $h_k(\boldsymbol{\xi})$ is a linear hyper-plane that is parametrized by the reference trajectory $\{\boldsymbol{\xi}_t^{\text{ref}}\}_{t=1\ldots T_N}$ and attractor $\boldsymbol{\xi}_l^*$ as follows:

$$\begin{aligned} h_k(\boldsymbol{\xi}) &= \mathbf{w}_k^T \boldsymbol{\xi} + b_k, \\ \mathbf{w}_k &= \frac{\boldsymbol{\mu}^k - \boldsymbol{\xi}_k^*}{||\boldsymbol{\mu}^k - \boldsymbol{\xi}_k^*||} \quad \text{and} \quad b_k = 1 - \mathbf{w}_k^T \boldsymbol{\xi}_k^* \end{aligned} \tag{23}$$

where $\boldsymbol{\mu}^k$ is the sample mean of the reference trajectory, which is assumed to be the mean of the Gaussian component for corresponding χ^k (23) can be considered as a simple linear classifier, where $h_k(\boldsymbol{\xi}) \geq 1$ denotes the region of the '+' class, corresponding to the *locally active* DS and $h_k(\boldsymbol{\xi}) < 1$ denotes the region of the '-' class, corresponding to the *locally deflective* DS. We convert (23) to the following non-negative function (using the same values for $\mathbf{w}_k$ and b_k):

$$h_k(\boldsymbol{\xi}) = \frac{1}{2}(\mathbf{w}_k^T \boldsymbol{\xi} + b_k + ||\mathbf{w}_k^T \boldsymbol{\xi} + b_k||) \tag{24}$$

and propose the following combined DS:

$$\begin{aligned} \mathbf{f}_l^k(h_k(\boldsymbol{\xi}), \boldsymbol{\xi}) &= \tilde{h}(\boldsymbol{\xi}) \mathbf{f}_{l,a}^k(\boldsymbol{\xi}) + (1 - \tilde{h}(\boldsymbol{\xi})) \mathbf{f}_{l,d}^k(\boldsymbol{\xi}) \\ &\quad - \lambda_k(\boldsymbol{\xi}) \nabla_{\boldsymbol{\xi}} h_k(\boldsymbol{\xi}). \\ \text{for} \quad \tilde{h}_k(\boldsymbol{\xi}) &= \begin{cases} 1 & \text{if} \quad h_k(\boldsymbol{\xi}) \geq 1 \\ h_k(\boldsymbol{\xi}) & \text{if} \quad h_k(\boldsymbol{\xi}) < 1 \end{cases} \end{aligned} \tag{25}$$

(25) modulates the two DS with the continuous partition function $\tilde{h}_k(\boldsymbol{\xi}) \in [0, 1]$, this allows for a smoother transition between the *active* and *deflective* regions in the composed DS (12). Furthermore, the term $-\lambda_k(\boldsymbol{\xi}) \nabla_{\boldsymbol{\xi}} h_k(\boldsymbol{\xi})$ adds velocity components in the direction of the negative gradient of the partition function modulated with an exponential radial basis function, as (16) centered at the local attractor $\boldsymbol{\xi}_k^*$. The addition of this term is necessary when in the composed system (12) the direction of motion of the global DS $\mathbf{f}_g(\cdot)$ and the local DS $\mathbf{f}_l^k(\cdot)$ are either perpendicular or opposing each other. In Fig. 5 we illustrate different combinations of global and local DS. Furthermore, we also show illustrations of non-linear reference trajectories composed of multiple locally active models, combined with a *non-linear* global DS. As can be seen, LAGS-DS is highly flexible in the sense that it allows the modeling of a myriad of different dynamic behaviors with the same formulation.

*6) **Global Asymptotic Stability Analysis of the Composed System**:* To ensure the global asymptotic stability of (12) we propose a complex Lyapunov candidate function which is a combination of a global parametrized quadratic Lyapunov function (QLF) and a set of locally active asymmetric QLF's

as follows,

$$V(\boldsymbol{\xi}) = (\boldsymbol{\xi} - \boldsymbol{\xi}_g^*)^T \mathbf{P}_g (\boldsymbol{\xi} - \boldsymbol{\xi}_g^*) + \sum_{k=1}^{K} \beta^k(\boldsymbol{\xi}) (\boldsymbol{\xi} - \boldsymbol{\xi}_g^*)^T \mathbf{P}_l^k (\boldsymbol{\xi} - \boldsymbol{\xi}_k^*) \tag{26}$$

where,

$$\beta^k(\boldsymbol{\xi}) = \begin{cases} 1 & \forall \boldsymbol{\xi} : (\boldsymbol{\xi} - \boldsymbol{\xi}_g^*)^T \mathbf{P}_l^k (\boldsymbol{\xi} - \boldsymbol{\xi}_k^*) \geq 0 \\ 0 & \forall \boldsymbol{\xi} : (\boldsymbol{\xi} - \boldsymbol{\xi}_g^*)^T \mathbf{P}_l^k (\boldsymbol{\xi} - \boldsymbol{\xi}_k^*) < 0 \end{cases} \tag{27}$$

with $\mathbf{P}_g, \mathbf{P}_l^k \in \mathbb{R}^{M \times M}$ being positive definite matrices. (27) is a simplification of the weighted sum of asymmetric quadratic functions (WSAQF) proposed in [15] which was used as a Control Lyapunov function. As opposed to the original formulation, we restrict all $\mathbf{P}$ matrices to be symmetric, such that we can use them to impose stability constraints on our system matrices $\mathbf{A}_g^k, \mathbf{A}_{l,a}^k \forall k = 1, \ldots, K$. The first term in (26) is a standard QLF centered at the global attractor $\boldsymbol{\xi}_g^*$, while the second term is a set of asymmetric local QLF shaped by the local attractors $\boldsymbol{\xi}_k^*$ and activated via $\beta^k(\boldsymbol{\xi}) \ \forall k = 1, \ldots, K$. The asymmetry, in this case, comes from the relative error $(\boldsymbol{\xi} - \boldsymbol{\xi}_g^*)^T (\boldsymbol{\xi} - \boldsymbol{\xi}_k^*)$. The structure of (26) is tied to that of (12), namely the first term dictates the stability for the global DS $\mathbf{f}_g(\cdot)$, while bounding the rate of contraction of the local DS $\mathbf{f}_l^k(\cdot)$ through the interaction with the local asymmetric QLF's. Sufficient conditions for *global asymptotic stability* of (12) with the proposed Lyapunov candidate function (26) are derived and provided in [14].

IV. Learning LAGS-DS from Demonstrations

We briefly describe the learning approach for LAGS-DS. The interested reader can refer to [14]. Our learning scheme (as depicted in Fig. 6) is comprised of a three-step procedure. Initially, we must discover the locally active regions from the training data through locally linear trajectory clustering. This is done via a Gaussian Mixture Model (GMM), where each cluster is assumed to represent a locally linear trajectory. Then, the set of $K + 1$ symmetric $\mathbf{P}$ matrices is learned from the training data and the estimated local attractors via a simplification of the constrained optimization problem proposed in [15]. Finally, the set of system dynamics matrices are estimated via a constrained optimization problem with the derived Lyapunov stability conditions from (26) parametrized by the the pre-learnt $\mathbf{P}$ matrices and the task requirements defined by the user; i.e. the type of locally active dynamics.

V. Simulations and Next Steps

For a wiping task as the one illustrated in Fig. 1 we provide 2D simulations of a learned LAGS-DS from reference trajectories as the ones shown in the bottom illustrations of Fig. 5 being executed with the passive-DS controller (9). This simulation can be found in the following simulation-link-3, where we can see a stiffness attraction to the reference trajectory that corresponds to the "surface". In simulation-link-4 we show the behavior of the robot with a global-DS without any locally active regions. As can be seen, the robot simply follows the generalized motion pattern,

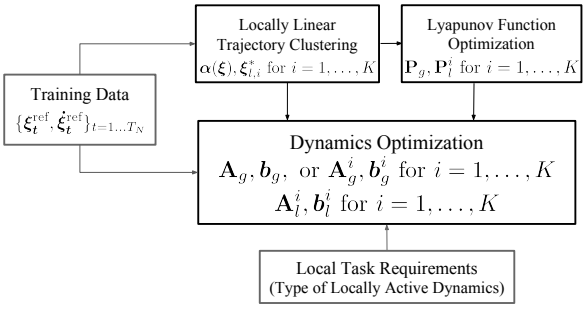

Fig. 6: Schematic of the proposed LAGS-DS Learning Approach.

without coming back to the desired reference trajectory. While LAGS-DS smoothly comes back to the reference trajectory while allowing perturbations or changes in the state-space without inducing any extra energy or instabilities in the closed-loop system. Current efforts are targeted at learning such LAGS-DS from real demonstrations focused on the tasks presented in Fig. 1.

References

[1] S. H. Lee, I. H. Suh, S. Calinon, and R. Johansson, "Autonomous framework for segmenting robot trajectories of manipulation task," *Autonomous Robots*, vol. 38, no. 2, pp. 107–141, Feb 2015.

[2] S. Niekum, S. Osentoski, G. Konidaris, and A. G. Barto, "Learning and generalization of complex tasks from unstructured demonstrations," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2012, pp. 5239–5246.

[3] N. Figueroa and A. Billard, "Transform-Invariant Non-Parametric Clustering of Covariance Matrices and its Application to Unsupervised Joint Segmentation and Action Discovery," *ArXiv e-prints*, Oct. 2017.

[4] A. L. P. Ureche, K. Umezawa, Y. Nakamura, and A. Billard, "Task parameterization using continuous constraints extracted from human demonstrations," *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1458–1471, 2015.

[5] N. Figueroa, A. L. P. Ureche, and A. Billard, "Learning complex sequential tasks from demonstration: A pizza dough rolling case study," in *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, March 2016, pp. 611–612.

[6] N. Figueroa and A. Billard, "Learning complex manipulation tasks from heterogeneous and unstructured demonstrations," In Proceedings of Workshop on Synergies between Learning and Interaction. IEEE/RSJ IROS, 2017.

[7] N. Hogan, "Impedance control: An approach to manipulation," in *1984 American Control Conference*, June 1984, pp. 304–313.

[8] J. R. Medina, D. Sieber, and S. Hirche, "Risk-sensitive interaction control in uncertain manipulation tasks," in *2013 IEEE International Conference on Robotics and Automation*, May 2013, pp. 502–507.

[9] P. Kormushev, S. Calinon, and D. G. Caldwell, "Imitation learning of positional and force skills demonstrated via kinesthetic teaching and haptic input," *Advanced Robotics*, vol. 25, no. 5, pp. 581–603, 2011.

[10] S. Schaal, *Dynamic Movement Primitives -A Framework for Motor Control in Humans and Humanoid Robotics*. Tokyo: Springer Tokyo, 2006, pp. 261–280.

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

[12] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal on Robotics and Automation*, vol. 3, no. 1, pp. 43–53, February 1987.

[13] M. Khoramshahi and A. Billard, "A dynamical system approach to task-adaptation in physical human-robot interaction," *Autonomous Robots*, 2018.

[14] N. Figueroa and A. Billard, "Locally active globally stable dynamical systems: Theory and learning," *[In preparation]*, 2018.

[15] S. M. Khansari-Zadeh and A. Billard, "Learning control lyapunov function to ensure stability of dynamical system-based robot reaching motions," *Robotics and Autonomous Systems*, vol. 62, no. 6, pp. 752 – 765, 2014.