

# Learning and generalizing force control policies for sculpting

Vasiliki Koropouli, Sandra Hirche and Dongheui Lee

**Abstract**—Humans exhibit exceptional skills in using tools and manipulating objects of their environment by skillfully controlling exerted force and arm impedance. One of the basic components of this mechanism is the generation of internal models which associate kinematic variables with applied force. On the other hand, making robots capable of skillfully using tools and adapting their motor behavior to new environmental conditions is rather complex. In the present paper, we investigate learning of force control policies for robotic sculpting given multiple task demonstrations. These policies express the relationship between constrained motions and exerted force and are learned in Cartesian space where the coupling of dynamics between different directions of motion is also taken into account. In addition, a novel algorithm is proposed to generalize these policies to new motion tasks, executed in a sufficiently homogeneous environment, same with that in demonstrations, but in presence of new motion-dependent external forces. To this aim, a differential calculus approach is proposed where not only the mapping from motion to force but also from difference in motion to difference in force is learned to generalize the policies to new contexts. This is achieved by learning apart from a set of policy parameters, some newly introduced quantities, so called weight differentials, which express the rate of change of the policy parameters. The proposed approach is validated in simple real-world sculpting experiments by using a two degrees-of-freedom haptic device.

## I. INTRODUCTION

Most manipulatory tasks which involve tool use, such as screwing or sculpting, can cause instability and make the tool move unpredictably in presence of disturbances. Humans exhibit exceptional skills in using tools and manipulating objects of the environment by regulating the force and impedance of their arm [1]–[3]. Since humans show adroit adaptation to new force fields, we believe that learning from humans is a promising route to endowing robots with special force control skills in a safe manner.

In this paper, we treat the intriguing problem of learning force control policies for robotic sculpting from multiple demonstrations and generalizing these policies to new tasks. The new tasks consist of manipulating the same environment with that in demonstrations but following different motion paths which impose new disturbances on the end-effector. In our scenario, the environment consists of a deformable and sufficiently homogeneous material. Sculpting tasks induce development of varying motion-dependent external force fields whose components in different directions of motion are interconnected and this constrains tool motion. Moreover, the coupling strength is different depending on the system position and velocity states making the overall coupled

dynamics inherently hard to model explicitly. For example, sculpting a pattern in short depth from the object's surface is a quite different task than sculpting the same pattern deeper inside the object where the manipulating mass increases significantly and imposes different constraints on motion.

Learning control policies has been a popular topic of robotic research during last decades. A review on this topic is presented in [5] while learning of force control policies has been presented in previous works [6]–[8]. In [6], force patterns are learned in the form of interaction control policies from single task demonstrations. In [7], positional and force skills are separately demonstrated and learned in the form of mixtures of dynamical systems from multiple demonstrations. However, in dynamic interaction tasks, such as sculpting, position and force should not be viewed independently. In [8], force control policies are learned by reinforcement learning following a kinesthetic task demonstration. Reinforcement learning, however, requires multiple trials and is not recommended in sculpting where successful task generalization in a single trial is desired.

Existing regression methods such as Linear Regression (LR) [10], Locally Weighted Regression (LWR) [9] and probabilistic modeling [11] allow for learning a set of parameters from demonstrated data and use this set to predict the output of the system at new inputs. More advanced regression techniques such as Receptive Field Weighted Regression (RFWR) [9] and Locally Weighted Projection Regression (LWPR) [12] are proposed for incremental learning in low and high-dimensional spaces respectively. RFWR and LWPR perform receptive field-based incremental learning and allow for generalization of a learning model to new unseen inputs. During generalization, the learning structure is modified to incorporate new information. These incremental learning techniques, however, are prone to spatially localized negative interference [9]. In the present work, instead, the aim is not to incorporate all incoming knowledge into the learning model but develop a generalization-dedicated method able to predict a policy at new inputs.

In the present work, we focus on learning generalization of force control policies from multiple demonstrations to manipulate deformable environments which are sufficiently homogeneous. A directional coupling in dynamics is introduced according to which the force policy in one direction of motion depends on the motion states of other directions apart from that direction itself. More specifically, we propose a novel algorithm to generalize a force policy to new trajectories, which are realized in the same environment, but induce different motion-varying disturbance. The new trajectories may vary in depth from the object's surface or length in

The authors are with the Institute of Automatic Control Engineering (LSR), Technische Universität München, 80290 München, Germany  
vicky@lsr.ei.tum.de {hirche, dhlee}@tum.de

the direction parallel to the surface. New motions inside a material induce different disturbance since the manipulating mass and contact area of the tool with the environment change according to the tool position, see Fig. 1. We should remark that, in our setting, generalization performance is investigated in terms of position generalization accuracy without considering performance metrics which vary with the properties of the environment. Using position control in our scenario, however, is rather inappropriate since a kinematic plan incompatible with the environment or end-effector's mechanical resistance would result in high impact forces or even task failure. Instead, a compliant control policy is of interest to be learned which regulates interaction [4]. The proposed generalization algorithm is significantly less complex than computationally expensive incremental learning techniques [12] since it only allows for policy prediction given new inputs without integrating new information into the learning structure.

The current paper is structured as follows. In Section II, we define the problem and, in Section III, we present a new policy generalization algorithm. In Section IV, we demonstrate our experimental setup and compare the performance of the proposed approach with existing regression techniques while in Section V we discuss the results.

## II. PROBLEM DEFINITION

In this work, we treat the problem of generalizing force control policies for motion path following inside a deformable object. The goal is to learn a force policy given multiple demonstrations of a task and generalize this policy to new motion paths, executed in the same and sufficiently homogeneous environment, but in presence of different disturbances induced by the different kinematic plan.

During motion inside a deformable material, dynamics are directionally coupled. Let us consider a sculpting task where a tool has to move inside a material such as plasticine. The tool is first moving in the direction normal to the object's surface, and then, it follows a path in the direction parallel to the surface to engrave the material, see Fig. 1. While executing this task, we observe that tool's parallel motion is easier and faster in lower than in higher depth inside the object. This is because as depth increases, the manipulating mass and, thus, parallel external force increases. Thus, parallel force is coupled to normal motion. In the present work,

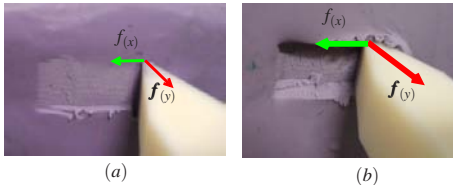


Fig. 1. Illustrating an engraving task at different depths inside a plasticine object. Different environmental disturbance  $\{f_{(x)}, f_{(y)}\}$  is experienced in each case due to changing manipulating mass. Engraving in a (a) low depth, (b) high depth.

we consider learning of force control policies in Cartesian

space described by

$$\mathbf{f} = \pi(S) \quad (1)$$

where  $\mathbf{f} \in \mathbb{R}^{M \times 1}$  is applied force at the end-effector and  $S \in \mathbb{R}^{M \times D}$  is an input matrix where  $M$  denotes the number of motion directions and  $D$  the number of motion variables per direction. The motion variables consist of position, velocity and acceleration data. The goal is to learn the policy  $\pi$  given pairs of demonstrated motion and force data  $\{S, \mathbf{f}\}$  of a task and generalize this policy to different motions  $S' \not\subseteq S$ . In other words, we desire to solve a generalization problem which consists of estimating the control policy  $\mathbf{f}' = \pi'(S')$  which is required to realize a different kinematic plan  $S'$ . Visiting new paths in homogeneous environments induces different environmental force because the manipulating mass changes and this requires proper adaptation of exerted force.

## III. LEARNING AND GENERALIZING CONTROL POLICIES

In this Section, we describe how to learn force control policies given multiple demonstrations of force and motion data of a manipulation task and propose a method to generalize these policies to new motions subjected to motion-varying disturbance. First, we analyze Linear Regression, and then, we propose a new approach which employs Linear Regression to learn a set of parameters for the generalization policy.

*Notation:* In the remainder of the paper, we denote time index by lower case letters  $(\cdot)_i$ , index of motion direction by lower case letters inside parentheses  $(\cdot)_{(i)}$  and demonstration index by upper case letters inside parentheses  $(\cdot)^{(i)}$ .

### A. Learning force control policies from multiple demonstrations

We consider a force control policy in Cartesian space

$$\mathbf{f} = \pi(\mathbf{s}_{(1)}, \dots, \mathbf{s}_{(M)}) \quad (2)$$

where  $\mathbf{f} = [f_{(1)} \dots f_{(M)}]^T$  is a vector of force profiles  $f_{(m)}$  and  $\mathbf{s}_{(m)} = \{\mathbf{s}_{(m),i}, i = 1, \dots, N\}$ ,  $m = 1, \dots, M$  is a motion plan at the  $m$ -th direction of motion where  $N$  is the number of timepoints of the plan. The force control policy is defined by

$$\pi = [\pi_{(1)}(\mathbf{s}_{(1)}) \dots \pi_{(M)}(\mathbf{s}_{(M)})]^T \quad (3)$$

where the motion plans consist of motion vectors  $\mathbf{s}_{(m),i} = [x_{(m),i} \dot{x}_{(m),i} \ddot{x}_{(m),i} \mathbf{c}_{(m),i}]$ ,  $i = 1, \dots, N$ . The  $x_{(m),i}$ ,  $\dot{x}_{(m),i}$ ,  $\ddot{x}_{(m),i}$  represent the position, velocity and acceleration respectively in the  $m$ -th direction and  $\mathbf{c}_{(m),i}$  is a vector-valued function which represents the coupling between the  $m$ -th and the other directions  $j \neq m$ . The coupling function is  $\mathbf{c}_{(m)} = \mathbf{c}_{(m)}(x_{(j)}, \dot{x}_{(j)})$ ,  $\forall j \neq m$  and establishes a dependence of the force  $f_{(m)}$  on the position and velocity states of the remaining directions  $j \neq m$ .

Let us consider  $K$  demonstrations of a task with  $N$  datapoints per demonstration. For reasons of simplicity, we omit the directional index  $m$  in the remainder of the Section. To learn the force control policy in either direction, data pairs from all demonstrations are concatenated as  $(\{\mathbf{s}_1, f_1\}, \dots, \{\mathbf{s}_K, f_K\})$ . At this point, we present

learning by LR which is also employed by the proposed generalization method. In LR, a control policy  $\pi$  is learned by minimizing the cost function

$$R = \sum_{i=1}^{K \times N} \|f_i - \pi(\mathbf{s}_i)\|^2. \quad (4)$$

The policy is represented by  $\pi = \mathbf{w}^T \phi(\mathbf{s})$  where  $\mathbf{w} \in \mathbb{R}^{D+1}$  is a parameter vector and  $\phi(\mathbf{s}) = [\mathbf{s} \ 1]^T$  is a basis function model where  $\mathbf{s} = \{\mathbf{s}_i, i = 1, \dots, N\}$  and  $\mathbf{s}_i \in \mathbb{R}^{1 \times D}$ . Based on this policy representation, the cost criterion (4) becomes

$$R = \sum_{i=1}^{K \times N} (f_i - \mathbf{w}^T \phi(\mathbf{s}_i))^2. \quad (5)$$

By minimizing (5) with respect to  $\mathbf{w}$ , we receive

$$\sum_{i=1}^{K \times N} f_i \phi^T(\mathbf{s}_i) = \mathbf{w}^T \sum_{i=1}^{K \times N} \phi(\mathbf{s}_i) \phi^T(\mathbf{s}_i) \quad (6)$$

and, thus, the estimated parameter vector  $\hat{\mathbf{w}}$  is

$$\hat{\mathbf{w}} = \mathbf{w}_1^T H^{-1} \quad (7)$$

where  $\mathbf{w}_1 = \sum_{i=1}^{K \times N} f_i \phi(\mathbf{s}_i)$  and  $H = \sum_{i=1}^{K \times N} \phi(\mathbf{s}_i) \phi^T(\mathbf{s}_i)$ .

### B. Generalization by Weight Differential Learning (GWDL)

We propose a new algorithm to generalize force policies to new inputs based on the principle of knowledge exploitation. According to this, given some demonstrated data, we learn, not only a set of parameters which show the mapping from the input to output data but also the differentials of these parameters which represent the mapping from a difference in the input to a difference in the output value. The differentials of the parameters express the higher-order rates of change of the mapping policy.

Our generalization problem consists of estimating the force which needs to be exerted by the end-effector so that we allow for a desired kinematic plan  $\mathbf{s}'$  to be executed in some motion direction. This is, in fact, a problem of approximating the true value of a function which maps a desired motion onto a required force. To solve this, we introduce an algorithm inspired by differential calculus according to which a function at a new point is approximated by an expansion consisting of a finite number of components. These components are represented by the higher-order differences between the new and a known input point and are modulated by some weighting coefficients. We introduce the  $r$ -th order differentials of a weighting vector  $\mathbf{w}$  and we symbolize them by  $\Delta \mathbf{w}_r$ . These weight differentials express the mapping from a difference in the input to a difference in the output value and play the role of the weighting coefficients in the expansion.

To learn the  $\Delta \mathbf{w}_r$ , we first generate new observation datasets  $D_r$  from demonstrated data by computing the  $r$ -th order differences between datapoints of every two demonstrations. This is realized as follows. Given the set of datapoints from all demonstrations ( $\{\mathbf{s}_1, f_1\}, \dots, \{\mathbf{s}_{N \times K}, f_{N \times K}\}$ ), we define the motion vector of demonstration  $k$  at time  $i$

as  $\mathbf{s}_i^{(k)} = \mathbf{s}_{i+N(k-1)}$  and the corresponding force element as  $f_i^{(k)} = f_{i+N(k-1)}$ . We additionally define the datasets

$$D_r^{k_1, k_2} = \{(\mathbf{s}_i^{(k_1)} - \mathbf{s}_i^{(k_2)})^r, (f_i^{(k_1)} - f_i^{(k_2)})^r, i = 1, \dots, N\}, \quad (8)$$

which consist of the differences of datapoints between two different demonstrations  $k_1$  and  $k_2$  where  $k_1, k_2 = 1, \dots, K$ . Finally, we concatenate all datasets  $D_r^{k_1, k_2}$  into a single dataset  $D_r$  as

$$D_r = \{D_r^{k_1, k_2}, k_1, k_2 = 1, \dots, K, k_1 \neq k_2\}. \quad (9)$$

By applying Linear Regression on the dataset  $D_r$ , we learn some new parameter vectors that we call weight differentials of  $r$ -th order;  $\Delta \mathbf{w}_r$ . To generalize the force control policy, we apply the following steps:

- (i) We concatenate observed motion vectors from all demonstrations as  $(\{\mathbf{s}_1^{(1)}, \dots, \mathbf{s}_i^{(k)}, \dots, \mathbf{s}_N^{(K)}\})$  and we compute the average over demonstrations motion trajectory  $\mathbf{s}^{\text{av}} = \{\mathbf{s}_i^{\text{av}}, i = 1, \dots, N\}$  where

$$\mathbf{s}_i^{\text{av}} = \sum_{k=1}^K \mathbf{s}_i^{(k)} / K. \quad (10)$$

- (ii) We estimate the points of the average motion sequence  $\mathbf{s}^{\text{av}}$  that are closest to the desired query motion plan  $\mathbf{s}'$  as:

$$\mathbf{s}^{(\min)} = \{\mathbf{s}_j^{(\min)}, j = 1, \dots, N\}, \quad (11)$$

$$\mathbf{s}_j^{(\min)} = \arg \min_{\mathbf{s}_i^{\text{av}}} \|\mathbf{s}'_j - \mathbf{s}_i^{\text{av}}\|, i = 1, \dots, N \quad (12)$$

where  $\|\cdot\|$  denotes the Euclidean distance. We compare each new motion vector  $\mathbf{s}'_j$  with the average motion plan  $\mathbf{s}^{\text{av}}$  so that we extinguish the need for memorizing demonstrated data and reduce the search space.

- (iii) Let us define the function  $h$ :

$$h_r: \mathbb{R}^{D+1} \rightarrow \mathbb{R}^{D+1} \text{ where } (h_r(\mathbf{v}))_i = (v_i)^r \quad (13)$$

where  $r \in \mathbb{R}$  and  $i = 1, \dots, D+1$ . We generalize the policy using the expansion

$$f'_j = \pi'(\mathbf{s}'_j) = \mathbf{w}^T \tilde{\mathbf{s}}_j^{(\min)} + \Delta \mathbf{w}_1^T (\tilde{\mathbf{s}}'_j - \tilde{\mathbf{s}}_j^{(\min)}) + \dots + \Delta \mathbf{w}_R^T h_R(\tilde{\mathbf{s}}'_j - \tilde{\mathbf{s}}_j^{(\min)}), j = 1, \dots, N \quad (14)$$

where  $\tilde{\mathbf{s}}_j^{(\min)} = [\mathbf{s}_j^{(\min)} \ 1]^T$ ,  $\tilde{\mathbf{s}}'_j = [\mathbf{s}'_j \ 1]^T$  and  $R$  represents the highest order of the expansion and is a predefined number.

The proposed algorithm has a strong intuitive meaning which is that, to predict future actions, we need to know the difference of the new task goal from previous goals and how this difference is mapped onto the action space.

#### IV. EXPERIMENTAL RESULTS

We test the performance of our proposed method in simple real-world sculpting experiments where the environment consists of a plasticine object which is considered sufficiently homogeneous. To ensure the object’s homogeneity, we perform many executions per task where the position variance across executions is shown to be of the order of  $10^{-6}mm^2$ . In our scenario, we are interested in task generalization in terms of position accuracy without considering other performance metrics which take into account variations in the properties of the environment.

The experiments are performed using a 2 DOF linear-actuated haptic device (*ThrustTube*), see Fig. 2 (a) where a plastic sculpting tool is attached on the end-effector of the device. The end-effector can move in two directions, one normal and one parallel to the object’s surface. In every trial, the material’s surface is initially made planar and the tool is placed such that it just touches the surface.

##### A. Data acquisition from humans

During demonstration and execution phase, the haptic device is controlled by admittance control with parameters; stiffness  $10N/m$ , damping  $30Ns/m$  and mass  $40Kg$  while the sampling rate is  $1KHz$ .

A sculpting task is demonstrated 3 times as follows: i) first, the user grasps the perpendicular cylindrical part of the sculpting tool and inserts the tool tip into the object in the normal direction, and once the desired depth has been achieved, ii) the user moves the tool in the parallel direction, up to some length, in order to engrave a pattern, see Fig. 2 (b). The route that the tool follows in the parallel direction has always the same length across demonstrations and what varies is the normal motion; how deep and at which velocity the tool is inserted into the object. When parallel motion takes place, normal velocity is close to zero. By this task, we aim at teaching the robot the force policy to execute movements, in the same environment, but in presence of varying motion-dependent disturbance. User force is measured by a force sensor attached at the end-effector of the haptic device while position and velocity data are recorded by the device’s encoders.

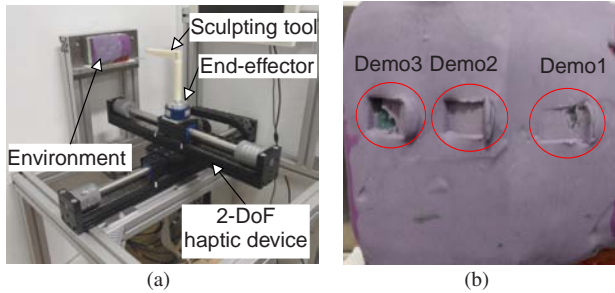


Fig. 2. (a) Experimental setup, (b) Three demonstrations of a sculpting task on a plasticine material.

##### B. Force control policies

Based on the above task description, the following laws hold according to the analysis in Section III-A:

the force policy is defined by  $\pi = [\pi_{(n)}(\mathbf{s}_{(n)}) \pi_{(p)}(\mathbf{s}_{(p)})]^T$  where the indices  $n$  and  $p$  denote the normal and parallel direction of motion respectively. The motion vectors in the two directions are  $\mathbf{s}_{(n),i} = [x_{(n),i} \dot{x}_{(n),i} \ddot{x}_{(n),i}]$  and  $\mathbf{s}_{(p),i} = [x_{(p),i} \dot{x}_{(p),i} \ddot{x}_{(p),i} x_{(n),i} \dot{x}_{(n),i}]$  where  $i$  is the time index. In this way, a coupling of the parallel force to the normal position and velocity states is established which is equal to  $\mathbf{c}_{(p)} = [x_{(n)} \dot{x}_{(n)}]$  while the coupling of normal force to parallel states is  $\mathbf{c}_{(n)} = \mathbf{0}$ . This means that the normal motion is independent of the parallel direction and we push as much deep as we desire. Parallel force, however, is responsible for extracting the mass from the material and this accumulated mass is always dependent on how deep the tool is inserted, and thus, dependent on normal motion.

##### C. Learning force control policies

In this part, we demonstrate policy learning by LR which is later employed in our proposed generalization approach. During demonstrations, human user’s speed varies with depth in both directions of motion; demonstrations require longer time when we push harder into the object and shorter time when we push less. Multidimensional Dynamic Time Warping [13] is applied to align the force and motion data from different demonstrations before learning. The two policies  $\pi_{(n)}$  and  $\pi_{(p)}$  are learned by LR and force estimation over same training data is shown in Fig. 3. The corresponding mean square force estimation error is shown in Table I.

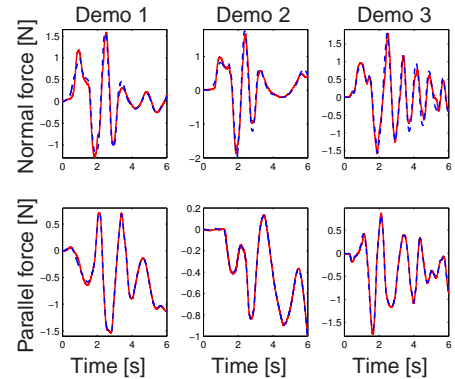


Fig. 3. Force estimation by LR given training data. Blue dashed: demonstrated force, red solid: estimated force.

TABLE I  
MEAN SQUARE FORCE ESTIMATION ERROR IN  $[N^2]$

Direction	Demo1	Demo2	Demo3
Normal	0.0131	0.0151	0.0252
Parallel	$0.666 \cdot 10^{-3}$	$0.0554 \cdot 10^{-3}$	$0.9041 \cdot 10^{-3}$

##### D. Generalization to new motion paths

In this set of experiments, we aim at testing policy generalization to new tasks by GWDL and compare it with generalization by existing regression techniques LR, LWR and LWPR. The comparison criterion consists of the tracking error between realized and desired position trajectory in the two directions of motion. We distinguish two main cases of generalization where: i) the new motions lie inside the

range of the experienced motions and we call it *policy interpolation* and ii) the new motions lie outside the range of the experienced motions and we call it *policy extrapolation*. We test generalization in two different scenarios. In the first scenario, the generalization tasks consist of engraving patterns of same parallel length as in demonstrations but at different depths inside the material. In the second scenario, generalization is tested in motions which vary in both normal depth and parallel length compared to demonstrated motions. Discrimination between these two different scenarios is significant in order to demonstrate how the coupling between the two directions affects the parallel force generalization policy.

1) *Policy generalization in normal direction*: First, we test generalization in case of interpolation where the new normal position trajectory lies inside the range of the demonstrated ones. To this aim, we generate a new kinematic plan  $\{\mathbf{s}'_{(n)}, \mathbf{s}'_{(p)}\}$  where  $\mathbf{s}'_{(n)} = [x'_{(n)} \dot{x}'_{(n)} \ddot{x}'_{(n)}]$  and  $\mathbf{s}'_{(p)} = [x^{(3)}_{(p)} \dot{x}^{(3)}_{(p)} \ddot{x}^{(3)}_{(p)} x'_{(n)} \dot{x}'_{(n)}]$  where  $\{x^{(3)}_{(p)} \dot{x}^{(3)}_{(p)} \ddot{x}^{(3)}_{(p)}\}$  is the parallel motion plan of demonstration 3, see Fig. 4. The  $x'_{(n)} = \{x'_{(n),i} = 0.8 x^{(3)}_{(n),i}, i = 1, \dots, N\}$  where  $x^{(3)}_{(n)}$  is the normal positional profile of Demo 3 while  $\dot{x}'_{(n)}$  and  $\ddot{x}'_{(n)}$  are computed from  $x'_{(n)}$ . The only new motion variables in the parallel direction are the normal states. Generalization is realized by LR, LWR, LWPR and GWDL (3 approximation terms in (14)) and the results are shown in Fig. 4. Some force profiles may not be visible in the figure because of their small difference with the other force signals. The figure depicts demonstrated and generalized forces by all 4 methods as well as measured position in each case. By comparing the position signals, we observe that the generalization performance of LR and LWR is similar and neither of them achieves efficient approximation of  $x'_{(n)}$ . On the contrary, LWPR and GWDL which also exhibit similar performance, achieve accurate approximation of the new desired normal trajectory  $x'_{(n)}$ . The tracking error in the parallel motion is lowest in case of GWDL while LWPR exhibits similar parallel tracking error with that of LWR. Table II shows the total square tracking error between desired and generalized positional profiles  $E_{(n)} = \sum_{i=1}^N (x'_{(n),i} - \hat{x}_{(n),i})^2$  and  $E_{(p)} = \sum_{i=1}^N (x^{(3)}_{(p),i} - \hat{x}_{(p),i})^2$  for all methods. The  $\hat{x}$  represents, in general, the generalized position.

TABLE II

GENERALIZATION TO NEW NORMAL KINEMATIC PLAN-INTERPOLATION

Error	LR	LWR	LWPR	GWDL (3 terms)
$E_{(n)}$	0.0020	0.0029	0.00055	0.00058
$E_{(p)}$	0.0564	0.1294	0.1385	0.0343

To test extrapolation to new normal paths outside the experienced position range, a new normal positional profile  $x'_{(n)} = \{x'_{(n),i} = 1.2 x^{(3)}_{(n),i}, i = 1, \dots, N\}$  is generated and the new plan is  $\{\mathbf{s}'_{(n)}, \mathbf{s}'_{(p)}\}$  where  $\mathbf{s}'_{(n)} = [x'_{(n)} \dot{x}'_{(n)} \ddot{x}'_{(n)}]$  and  $\mathbf{s}'_{(p)} = [x^{(3)}_{(p)} \dot{x}^{(3)}_{(p)} \ddot{x}^{(3)}_{(p)} x'_{(n)} \dot{x}'_{(n)}]$ , see Fig. 5. The figure visualizes demonstrated and generalized force by LR, LWR, LWPR and GWDL (2 approximation terms in (14)) as well

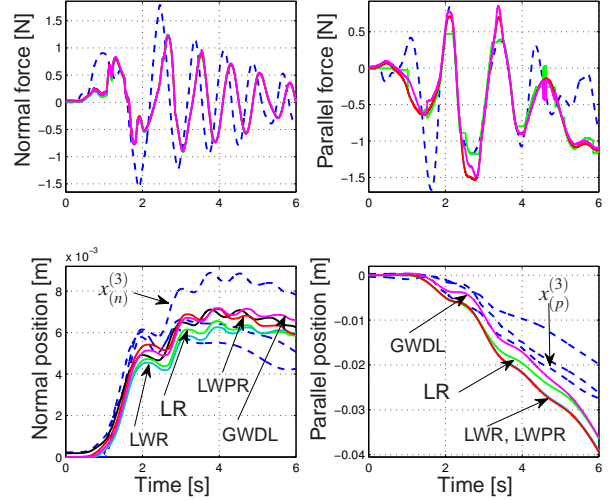


Fig. 4. Policy interpolation to new normal motion. Upper row: exerted force, lower row: measured position. Blue dashed: demonstrated signals, green: generalization by LR, turquoise: generalization by LWR, red: generalization by LWPR, magenta: generalization by GWDL-3 terms. New desired normal position  $x'_{(n)}$  is shown by black color.

as corresponding measured position in each case. Some force profiles seem rather overlapped by other force profiles because of their small value difference. We observe that GWDL outperforms LWPR and LWR while LWPR seems to approximate  $x^{(3)}_{(n)}$  after some time of movement. We also note that performance of GWDL and LWR improves with movement time. In the parallel direction, highest performance is exhibited by GWDL and LR which have similar performance while LWPR and LWR exhibit larger error. We should note that, when 3 approximation terms are employed in (14) for extrapolation, generalization by GWDL is unsuccessful and the task cannot be achieved. Table III shows the total square position tracking error  $E_{(n)} = \sum_{i=1}^N (x'_{(n),i} - \hat{x}_{(n),i})^2$  and  $E_{(p)} = \sum_{i=1}^N (x^{(3)}_{(p),i} - \hat{x}_{(p),i})^2$  in all generalization cases.

TABLE III

GENERALIZATION TO NEW NORMAL KINEMATIC PLAN-EXTRAPOLATION

Error	LR	LWR	LWPR	GWDL (2 terms)
$E_{(n)}$	0.0393	0.0062	0.0059	0.0042
$E_{(p)}$	0.0501	0.1114	0.1286	0.0488

2) *Policy generalization in normal and parallel direction*: In this scenario, we demonstrate generalization to new normal and parallel motions compared to the demonstrated ones. The new kinematic plan is  $\{\mathbf{s}'_{(n)}, \mathbf{s}'_{(p)}\}$  where  $\{x'_{(n)} = 0.8 x^{(3)}_{(n)}\}$  and  $\{x'_{(p)} = 1.1 x^{(3)}_{(p)}\}$  where  $\mathbf{s}'_{(n)} = [x'_{(n)} \dot{x}'_{(n)} \ddot{x}'_{(n)}]$  and  $\mathbf{s}'_{(p)} = [x^{(3)}_{(p)} \dot{x}^{(3)}_{(p)} \ddot{x}^{(3)}_{(p)} x'_{(n)} \dot{x}'_{(n)}]$ . New velocity and acceleration signals are computed from corresponding position signals. Fig. 6 shows generalized force by LR, LWR, LWPR and GWDL (3 terms) as well as corresponding measured position in each case. The new task in the normal direction is the same with that of case 1) – *interpolation*. In the parallel direction, the task changes to a longer and faster motion. In this direction, LWR and

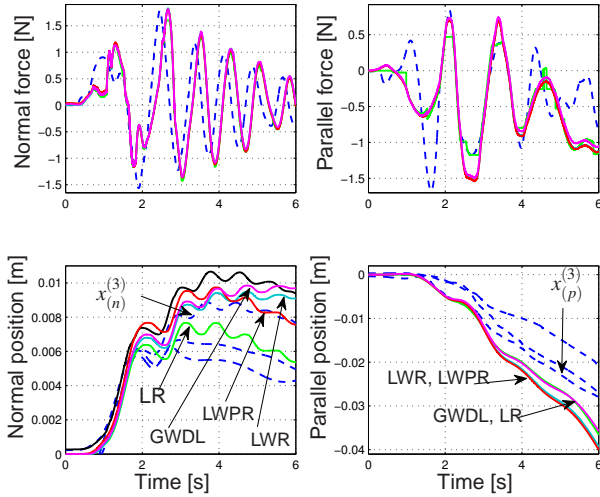


Fig. 5. Policy extrapolation to new normal motion. Upper row: exerted force, lower row: measured position. Blue dashed: demonstrated signals, green: generalization by LR, turquoise: generalization by LWR, red: generalization by LWPR, magenta: generalization by GWDL-2 terms. New desired normal position  $x'_{(n)}$  is shown by black color.

LWPR most successfully approximate  $x'_{(p)}$  with GWDL exhibiting larger tracking error.

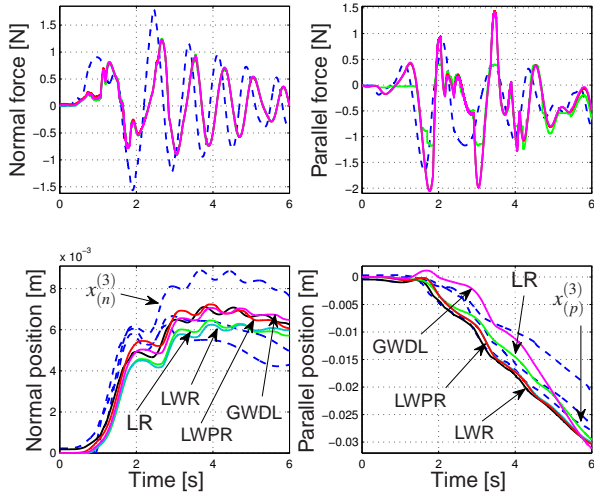


Fig. 6. Policy generalization to new normal and parallel motion. Upper row: exerted force, lower row: measured position. Blue dashed: demonstrated signals, green: generalization by LR, turquoise: generalization by LWR, red: generalization by LWPR, magenta: generalization by GWDL-3 terms. New desired normal and parallel position trajectories  $x'_{(n)}$  and  $x'_{(p)}$  are shown by black color.

## V. DISCUSSION

In this paper, we present a new approach to generalize force control policies to new motions, executed in the same environment with that in demonstrations, but subjected to different motion-dependent disturbances. The proposed algorithm is intuitive and simple and is based on concepts from differential calculus theory.

The test cases presented here show that proposed GWDL algorithm is advantageous over non-incremental learning techniques by exhibiting, in overall, better performance.

Compared to LWPR, GWDL could outperform in specific cases such as in extrapolating to new normal motions. However, apart from their generalization performance, presented techniques also involve significant differences in computational efficiency. At this point, we briefly discuss on the newly introduced GWDL and LWPR which is the most advanced incremental learning technique to solve generalization in high-dimensional spaces. As explained earlier, LWPR is based on incremental update of a learning structure to incorporate new knowledge into a system while it also deals with the problem of irrelevant input data. On the contrary, GWDL learns initially a set of parameters and uses these parameters to generalize to unseen inputs without updating the learning model which makes it significantly less expensive than LWPR. Then, to generalize by GWDL, demonstrations are exploited and all extracted knowledge (parameters and their differentials) is incorporated into a structure consisting of a finite number of weighted higher-order differences between a new and a known input point. It is important to note that, so far, it is not possible to automatically define the optimal number of approximation terms for generalization by GWDL and we select it a priori.

## ACKNOWLEDGMENTS

This research is partly supported by the DFG excellence initiative research cluster ‘‘Cognition for Technical Systems - CoTeSys’’ and the Bavarian Elite Aid Act (BayEFG)-Elite Network of Bavaria.

## REFERENCES

- [1] E. Burdet, K. P. Tee, I. Mareels, T. E. Milner, C. M. Chew, D. W. Franklin, R. Osu, M. Kawato, Stability and motor adaptation in human arm motions, *Biological Cybernetics*, vol. 94, pp. 20-32, 2006.
- [2] E. Burdet, R. Osu, D. W. Franklin, T. E. Milner and M. Kawato, The central nervous system stabilizes unstable dynamics by learning optimal impedance, *Nature*, vol. 414, pp. 446-9, 2001.
- [3] M. Kawato, Internal models for motor control and trajectory planning, *Current Opinion in Neurobiology*, vol. 9, pp. 718-727, 1999.
- [4] N. Hogan and S. P. Buerger, Impedance and Interaction Control. Chapter 19, *Robotics and Automation Handbook*, T.R.Kurfess, (ed.) CRC Press, 2004.
- [5] S. Schaal, C. Atkeson, Learning Control in Robotics, *Robotics and Automation Magazine, IEEE*, vol.17, no.2, pp. 20-29, 2010.
- [6] V. Koropouli, D. Lee and S. Hirche. Learning interaction control policies by demonstration, *Intelligent Robots and Systems (IROS)*, pp. 344-349, 2011.
- [7] 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.
- [8] M. Kalakrishnan, L. Righetti, P. Pastor and S. Schaal. Learning force control policies for compliant manipulation, *Intelligent Robots and Systems (IROS)*, pp. 4639-4644, 2011.
- [9] S. Schaal and C. G. Atkeson, Constructive Incremental Learning From Only Local Information, *Neural Computation*, vol. 10, 1997, pp 2047-2084.
- [10] C. M. Bishop, Pattern Recognition and Machine Learning, *New York: Springer*, 2006, ch. 3.
- [11] S. Calinon, F. Guenter and A. Billard, On learning, representing, and generalizing a task in a humanoid robot, *IEEE Transactions on Systems, Man and Cybernetics. Part B. Cybernetics: A Publication of the IEEE Systems, Man, and Cybernetics Society*, vol. 37(2), pp. 286-298, 2007.
- [12] S. Vijayakumar and S. Schaal, Locally weighted projection regression: An O(n) algorithm for incremental real time learning in high dimensional spaces, *Proceedings of the Seventeenth International Conference on Machine Learning*, 1, pp. 288-293, 2000.
- [13] E. Keogh and C. A. Ratanamahatana, Exact indexing of dynamic time warping, *KAIS*, vol. 7, pp. 358-386, 2005.