2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)
Banff Center, Banff, Canada, October 5-8, 2017

Learning and Robotic Imitation of Therapist's Motion and Force for Post-disability Rehabilitation

Carlos Martínez *Mahdi Tavakoli*
Department of Electrical and Computer Engineering
University of Alberta
Edmonton, Canada
martnez@ualberta.ca, mahdi.tavakoli@ualberta.ca

***Abstract*—The inclusion of robots in therapy is becoming common due to robots' power, repetitive motion ability, reprogramming capacity and adaptability to new tasks. In recent years, the demand for rehabilitation services has increased due to the rising number of patients with disability. In this paper, we propose a solution to the rising demand for therapists' services by combining Learning from Demonstration (LfD) and robotic rehabilitation. The goal of the paper is to implement LfD to model and learn the therapist's *behavior* (be it a trajectory or force) as a nonlinear dynamic system using a method called Stable Estimator of Dynamical Systems (SEDS) to later reproduce the learned behavior in the absence of the therapist using a robot. This method allows the therapists to first train a robot to learn his/her behavior such that, later when the therapist is no longer involved and the patient works alone with the robot, the robotic system determines whether and how to interact with the patient the same way the therapist would have interacted.**

***Keywords*— Learning from demonstration, Robotic rehabilitation, Stable Non-Linear Dynamical Systems.**

I. Introduction

In recent years, the number of people with disability has increased [1] due to events such as stroke, which has become one of the most common causes of disability in Canada and worldwide [2], [3]. Due to the loss of mobility in all or parts of their bodies, people with disability are usually not able to perform basic daily living activities or even simple reaching tasks.

It has been shown that in order to regain strength and mobility, the human brain can rewire neurological pathways through therapy exercises that engage patients in repetitive tasks [4]. However, due to the increasing demand for therapy services due to population ageing, therapists' time and healthcare resources are limited and cannot answer to all requests for therapeutic services. For this reason, robot-assisted therapy is becoming a popular solution. Robotic rehabilitation is divided in two categories of movement therapies: assistive therapy and resistive therapy. Assistive therapy uses haptic devices to assist the patient to complete the task, while resistive therapy the device opposes the patient's actions by applying resistive forces in order to build muscle strength. Robots are able to execute preprogramed tasks without fatigue and with a high accuracy, meaning they can be used to deliver therapy exercises to patients.

Most of the previously developed robotic systems have a common problem: they were designed for executing predefined and preprogrammed tasks, which limits the use of these systems in actual clinical settings. Due to the robots' preprogramming need, they are generally not able to adapt their behaviors based on the patient's changing needs or the task's changing skill requirements. In order to adjust the robot's therapeutic behavior in accordance with these just like a human therapist would do, robots must be reprogramed, which is difficult and inconvenient in clinical environments due to scarcity of computer programming skills in such environments.

In this paper, we propose to use LfD to help the robot reprogram itself whenever a therapist offers the robot a form of kinesthetic teaching of the required therapeutic behavior [5]. Our LfD technique learns the therapist's kinesthetically demonstrated behavior through an algorithm called SEDS in a step known as the learning or demonstration phase. Later, the robotic system reproduces and imitates the therapist's behavior in a step known as the reproduction or imitation phase – in this step, the therapist does not need to be present and the patient can repeat the therapy exercise as many times as needed.

This paper is organized as follows. In Section II, we explain the past related works. Section III gives a description about two specific therapy tasks. Section IV details the learning algorithm while Section V discussed the reproduction phase. Two experiments are presented in Section VI. Results, conclusions and future work are covered in Sections VII and VIII, respectively.

II. Related Work

The implementation of LfD in robotic rehabilitation systems is a relatively new idea. Our group has previously

This research was supported by the Canada Foundation for Innovation (CFI) under grant LOF 28241; the Alberta Innovation and Advanced Education Ministry under Small Equipment Grant RCP-12-021; the Natural Science and Engineering Research Council (NSERC) of Canada under the Collaborative Health Research Projects (CHRP) Grant #316170; and the Quanser, Inc. C. Martínez, M. Tavakoli are with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada (email: martnez@ualberta.ca, mahdi.tavakoli@ualberta.ca).

developed different robotic rehabilitation systems that leverage the learning/imitation features offered by LfD to save the therapists' time. In [6], the authors developed a haptic teleoperation system, which has potentials for use in home-based telerehabilitation, and employed impedance-based learning of the therapist's behavior, which is later used in the reproduction phase to imitate it. In [7], we developed a telerobotic cooperative rehabilitation system. With help from Gaussian Mixture Model (GMM) and Gaussian Mixture Regression (GMR) to enable learning and imitation, respectively, the therapist first interacts with the patient in a cooperative therapy task (lifting a bar by holding the two ends of it). Once the system learns the therapist's position-based behavior, the system replicates the therapist's behavior in his/her absence and when the patient is alone interacting with the robotic system.

In [8], the authors developed a system that combines robotic rehabilitation, LfD, and Assist-as-Needed (AAN). The system learns the therapist's impedance using GMM. Later, GMR is used to build a model of the therapist's behavior and use in the reproduction phase in combination with ANN. The system computes the error between the patient's current performance and the learned therapist's behavior and, based on this error, determines whether to assist the patient to complete the task or not. This feature is called AAN.

In our research, we propose to use a similar system but with some improvements. We use the well-developed learning algorithm called SEDS, which still uses GMM to build a model in the form of a nonlinear dynamic system capturing the demonstrated behavior. This algorithm uses a series of conditions to ensure global asymptotical stability (GAS) of the dynamic model. The difference with the classical GMR technique is that in order to create a stable imitation of the learned behavior, the reproduction phase combines the classical GMR algorithm and the SEDS algorithm. While GMR on its own does not ensure GAS, SEDS does so.

These method proposed in the paper allows the system to be completely task-independent. Also, the improvement in terms of using SEDS rather than GMR makes for a more robust system because SEDS generalizes the learned data; in other words, given a few demonstrations, the system is capable to reproduce the task even for non-demonstrated behaviors.

III. TASK DESCRIPTION

In this paper, two different position control and force control tasks are developed to show the LfD capabilities as applied to robotic rehabilitation. The proposed system's feature in terms of task independence means that it can be used with any position or force control task. The system can learn the desired behavior in terms of performing the target task without any knowledge about the task or the task environment. Also, the generalization capability of the proposed system means not too many demonstrations will be needed for the robot to learn what therapeutic behavior it is supposed to deliver to the patient.

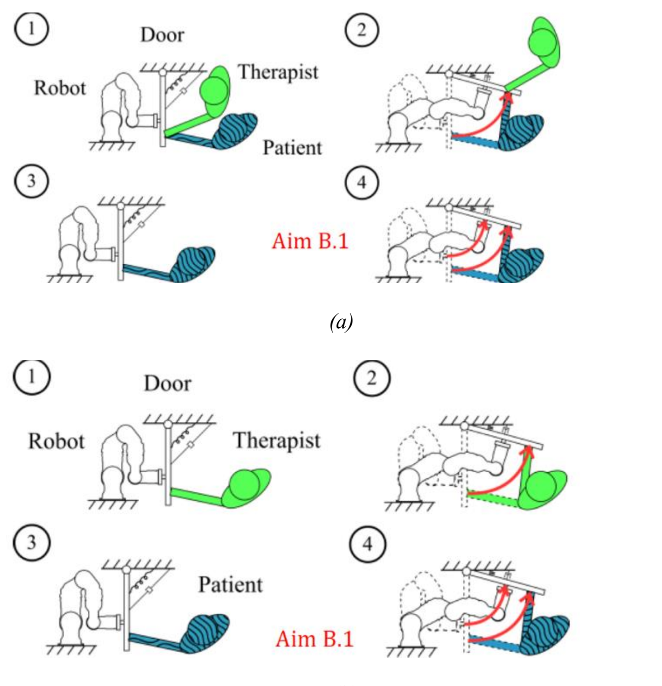
In the learning phase, the therapist executes the task a number of times, so the system builds a model of the demonstrated behavior. Figure 1 shows the two different ways to implement the task. In this paper, we implement the experiments using the second way. Once the model is built, in the reproduction phase, the robotic system monitors the patient's behavior and compares it versus the learned data (the therapist behavior previously demonstrated). Based on this error, the system compensates for the difference such that the patient feels as if interacting with the therapist even though interacting with the robot in actuality.

Task 1:

The first task involves point-to-point reaching where from a random initial point, the user must reach a given target point following a trajectory. Notice that this task involves two degrees of freedom (DoF). During the demonstration phase, the therapist holds the robot's end effector. Demonstrations are recorded from two different initial points. Afterwards, the system learns the therapist's reaching behavior and builds a model. In the reproduction phase, the patient starts to execute a similar reaching motion but cannot complete it due to his/her disability. Based on the error between the patient's and the learned behavior, the robot compensates for the difference to keep the patient as close as possible to the demonstrated behavior.

Task 2:

The second task involves force tracking. Given a compressed virtual spring with stiffness k_e, the user is asked to slowly decompress the spring until it is fully relaxed. During



(a)

(b)

Figure 1: Top figure (a) shows the first way to implement the tasks. The therapist and the patient are working together during the demonstration phase. Later in the reproduction phase, the patient works alone with the robot. Bottom figure shows a second way to implement the task. First the therapist works alone in the demonstration phase. Later in the reproduction phase, the patient works alone with the robot.

the demonstration phase, the therapist holds the robot's end effector to start from the initial point (compressed spring); when a non-zero force is applied by the robot to the therapist's hand. The therapist moves his hand to slowly decompress the spring and bring the interaction force to zero, at which point the task is finished. Notice that this task is 1-DoF. In the reproduction phase, the therapist is no longer in the therapy and the robot takes this role. The patient is then asked to execute the task on his/her own while holding the robot end effector. Again, the robotic system compares the patient's behavior versus the learned behavior previously demonstrated by the therapist and, based on the difference, provides compensation such that the patient is kept within the demonstrated behavior. Figures 2 and 3 show the two tasks.

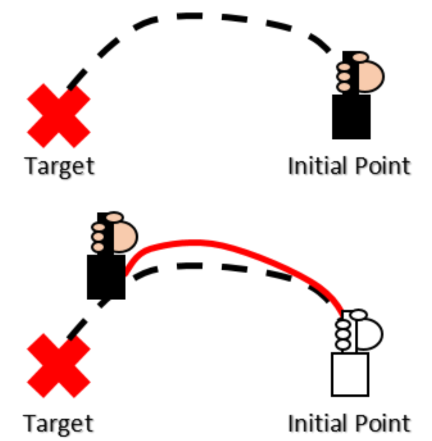


Figure 2: This figure shows the position-based task. Target and initial points are defined as a red cross and a human hand holding the robot's end effector, and the desired trajectory as demonstrated by the therapist is shown by dashed line. The top figure shows initial, target and desired trajectory at the beginning of the task, while the bottom figure shows initial, target, desired and actual trajectories, specifically the therapist's learned behavior (dashed line) and the actual reproduction phase trajectory (red solid line).

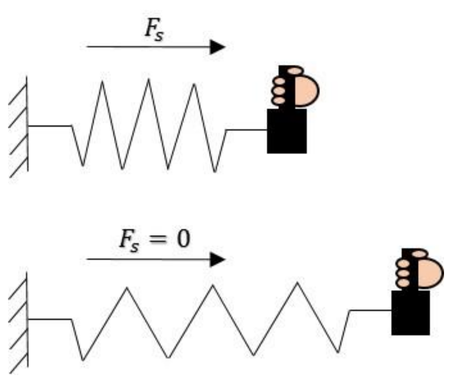


Figure 3: This figure shows the force task. The virtual spring is attached to the robot's end-effector and it applies a force Fs to the user. The top figure shows the initial position at the beginning of the task, here the spring is compressed and it is applying force to the user, while the bottom figure shows the therapist's learned behavior, the goal is to decompress the spring until Fs becomes equal to zero.

IV. LEARNING PHASE

In this section, we give introductions to the SEDS and GMM algorithms used for the learning phase. A more detailed explanation about the algorithms can be found in [7], [9].

GMM is a probability density function used to cluster data and has been widely used in the LfD field to encode spatial and temporal components of continuous trajectories and behaviors. GMM relies on a weighted sum of Gaussian component densities, where each component has its own mean and covariance matrix.

In order to use SEDS, the model of the learned motion must be implemented as a nonlinear autonomous dynamical system. In this paper, depending on the desired task, we take the human (and the robot) position or the human/robot interaction force as the input of the dynamical system, and the velocity or the first derivative of the force as the output. We model the system as

$$\dot{\xi} = f(\xi) \tag{1}$$

where f is a nonlinear function with a single equilibrium point and it is continuous for all time t. ξ is the position or force of the robot's end-effector in Cartesian space. Given any initial condition, the motion or force evolves according to the dynamics of $\dot{\xi}$.

The estimated model is given by

$$\dot{\xi} = \hat{f}\,(\xi) \tag{2}$$

Whereas f describes the actual dynamics of motion or force in the task demonstrated by the therapist, $\hat{f}$ is a function that tries to describe it by a set of parameters $\{\pi^K, \mu^K, \Sigma^K\} = \theta^K = \boldsymbol{\theta}$. Here, μ represents the mean of the Gaussian model, Σ is the covariance matrix, π is the prior, and K represents the index of the Gaussian model. The optimal values of these parameters are computed based on the set of demonstrations.

μ^K and Σ^K represent each Gaussian distribution K and defined by

$$\mu^K = \begin{pmatrix} \mu_\xi^K \\ \mu_{\dot{\xi}}^K \end{pmatrix} \;\&\; \Sigma^K = \begin{pmatrix} \Sigma_\xi^K & \Sigma_{\xi\dot{\xi}}^K \\ \Sigma_{\dot{\xi}\xi}^K & \Sigma_{\dot{\xi}}^K \end{pmatrix} \tag{3}$$

Every given point $\{\xi^{t,n}, \dot{\xi}^{t,n}\}_{t=0,n=1}^{T^n,N}$ where $(.)^{t,n}$ is the t-th data point of the n-th demonstration in the N demonstrations of the trajectory is linked to a probability density function given by

$$P(\xi^{t,n}, \dot{\xi}^{t,n}; \boldsymbol{\theta}) = \sum_{k=1}^{K} P(k) P(\xi^{t,n}, \dot{\xi}^{t,n} | k) \begin{cases} \forall_n \in 1..N \\ t \in 0..T^n \end{cases} \tag{4}$$

where $P(k)$ is the prior of the Gaussian distribution k , T is the total number of training data points, and $P(\xi^{t,n}, \dot{\xi}^{t,n} | k)$ is the conditional probability density function given by

$$P(\xi^{t,n}, \dot{\xi}^{t,n} | k) = \mathcal{N}(\xi^{t,n}, \dot{\xi}^{t,n}; \mu^K, \Sigma^K) = \frac{1}{\sqrt{(2\pi)^{2d} |\Sigma^k|}} e^{-\frac{1}{2}([\xi^{t,n}, \dot{\xi}^{t,n}] - \mu^K)^T (\Sigma^k)^{-1} ([\xi^{t,n}, \dot{\xi}^{t,n}] - \mu^K)} \tag{5}$$

After taking the posterior mean of $P(\dot{\xi}|\xi)$, the estimated function is given by

$$\dot{\xi} = \sum_{k=1}^{K} \frac{P(k)P(\xi|k)}{\sum_{i=1}^{K} P(i)P(\xi|i)} \left(\mu_{\dot{\xi}}^{k} + \Sigma_{\dot{\xi}\xi}^{k} \left(\Sigma_{\xi}^{k}\right)^{-1} \left(\xi - \mu_{\xi}^{k}\right) \right) \quad (6)$$

using the following simplification:

$$\begin{cases} A^k = \Sigma_{\dot{\xi}\xi}^{k} \left(\Sigma_{\xi}^{k}\right)^{-1} \\ b^k = \mu_{\dot{\xi}}^{k} - A^k \mu_{\xi}^{k} \\ h^k(\xi) = \frac{P(k)P(\xi|k)}{\sum_{i=1}^{K} P(i)P(\xi|i)} \end{cases} \quad (7)$$

Substituting (7) into (6) yields:

$$\dot{\xi} = \hat{f}\,(\xi) = \sum_{k=1}^{K} h^k(\xi)(A^k \xi + b^k) \quad (8)$$

Note that (8) is a sum of linear dynamical systems that results in a nonlinear function. In this new equation, $A^k \xi + b^k$ corresponds to a line with slope A^k and passes through the center of the Gaussians μ^K. The nonlinearity of this equation is given by $h^k(\xi)$; this is a weighting term that gives the influence of each Gaussian in the estimated function.

Given the previous conditions, (8) is asymptotically stable. To ensure GAS, the system must meet the two following conditions:

$$\begin{cases} b^k = -A^k \xi^* \\ A^k + (A^k)^T \prec 0 \end{cases} \quad \forall k = 1, \dots, K \quad (9)$$

Here, $\prec 0$ denotes the negative definiteness of a matrix. The proof is showed in [9].

Given the conditions and equations to compute and ensure GAS of the estimated function, the next step is to find the unknown parameters of (8), i.e., $\{\pi^K, \mu^K, \Sigma^K\} = \theta^K = \boldsymbol{\theta}$. To solve this problem, SEDS is used to compute the optimal values. Mean Square Error (MSE) is an optimization objective function that provides a solution to this problem. It is combined with SEDS to measure the accuracy of the estimations based on the recorded data. This minimization gives the optimal parameters:

$$\min_{\theta} \mathcal{J}(\theta) = \frac{1}{2\mathcal{T}} \sum_{n=1}^{N} \sum_{t=0}^{T^n} \left\| \hat{\dot{\xi}}^{t,n} - \dot{\xi}^{t,n} \right\|^2 \quad (10)$$

subject to the following constraints:

$$\begin{cases} b^k = -A^k \xi^* \\ A^k + (A^k)^T \prec 0 \\ \Sigma^k \succ 0 \\ 0 < \pi^K \leq 1 \\ \sum_{k=1}^{K} \pi^K = 1 \end{cases} \quad \forall k = 1, \dots, K \quad (11)$$

Note that the first two constraints in (11) are the previously defined conditions for stability presented in (9). MSE can be taken as a non-linear programming problem [10], and can be solved using Successive Quadratic Programming (SQP) [11].

V. Reproduction Phase

Given the learned parameters that ensure GAS, the next step is to retrieve the expected values based on the computed model through (8). The robot's current position or force is used as the input, and the output helps us find the next desired value for the same variable. In the position task, we integrate the desired velocity to get the next desired position value, while in the force task, we use the integral to get the next desire force value. Figure 4 shows the block diagram of the system. Notice that this configuration does not depend on the task nor the environment.

VI. Experiments

In this section, we present the experiments. We are using a Rehabilitation Robot (Quanser, Inc. Markham, Canada) [12] and a Gamma force sensor (ATI-IA, NC, USA) [13]. Figure 5 shows the rehab robot used in this paper.

Task 1:

During the demonstration phase, the therapist executed ten different demonstrations from two different initial points; Figure 6 shows the recorded data. A model of the demonstrated trajectory was built. As shown in the block diagram, a controller and a position feedback are needed in this task. A PD position controller is used to ensure the robot follows the desired trajectory in the reproduction phase. In the reproduction phase, the patient is asked to do the task from six different initial points. Section VII shows and discusses the obtained results.

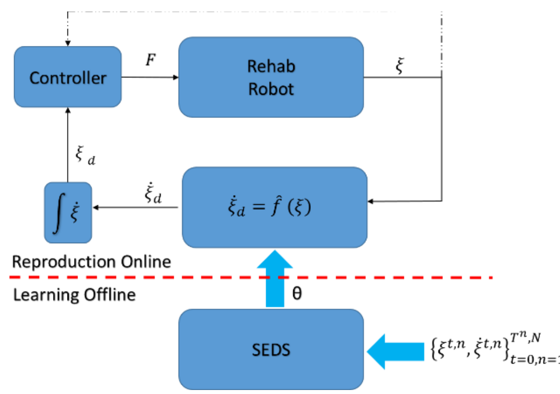


Figure 4: Block diagram showing the learning and reproduction phases

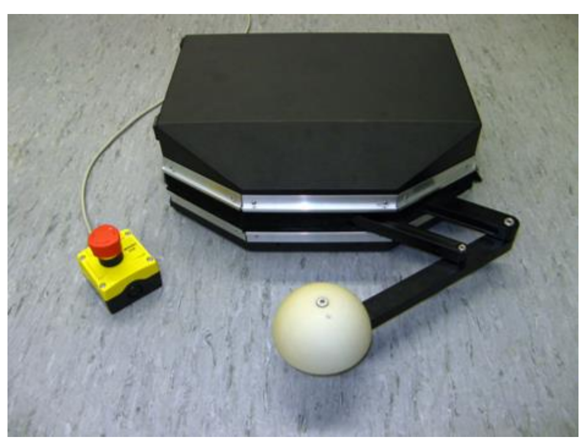


Figure 5: Rehab robot.

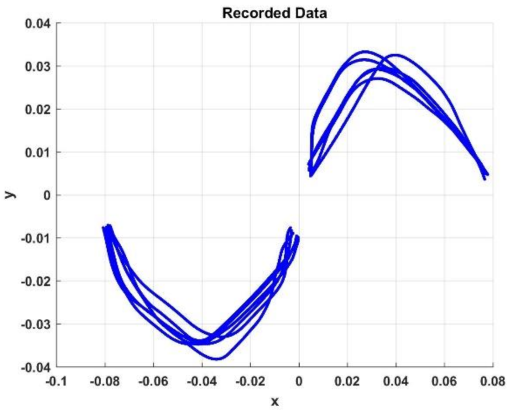


Figure 6: Recorded data during demonstration phase for position task.

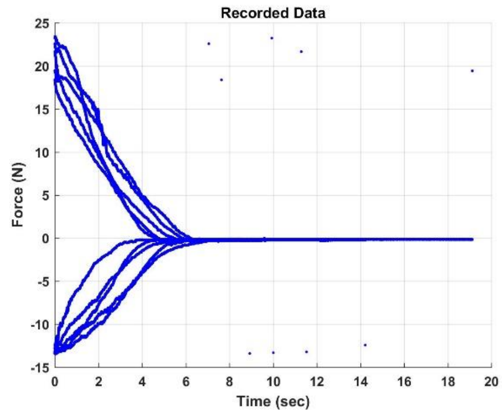


Figure 7: Recorded data during demonstration phase for force task.

Task 2:

As mentioned in Section III, the task involves a compressed virtual spring; the force applied by the robot's end effector on the human hand due to this spring is computed through the Hook's law [14]. In order to build a fair and challenging task for the patients, in this experiment, we used k_e=50. In this task, the controller is equal to one, and the feedback is not needed. The therapist executed ten different demonstrations of unloading the compressed spring to train the system, and Figure 7 shows the recorded data. The robot/human interaction force, its first derivative, and time were recorded and used by the learning algorithm. After the demonstrations, the system builds a nonlinear dynamical model capturing the data. Then, after the learning phase, the patient is asked to do the task. Section VII shows and discusses the obtained results.

VII. Results

In this section, we show and discuss the obtained results during the reproduction phase. In the position-based task, the system learned and generalized the demonstrated trajectories. As can be seen in Figure 8, the system is able to build a model, accept different initial points (different from the recorded during demonstration phase) and follow as close as possible the demonstrated trajectory. Notice that while the robot does not reach position [0,0] exactly because near the origin the velocity is too small to move the robot, the end effector gets close enough to the target from a practical perspective. Therefore, we can conclude that the system is actually helping the patient reach the target point.

In the force-based task, the system learned and generalized the demonstrated force. The results are shown in Figure 9, where the black and solid line is the current force applied by the user, this force is equal to the spring force but with an opposite sign, while the red and dashed line is the desired force. Notice that the desire force is always smaller than the current force and it always tries to converge to the origin (F=0). The gap between the desire force and the current force is as small as the therapist trained the robot. This gap demonstrates that the system learned the demonstrated force. It not only follows the force, but also makes it converge to zero. Therefore, the system learned and reproduced the demonstrated behavior properly.

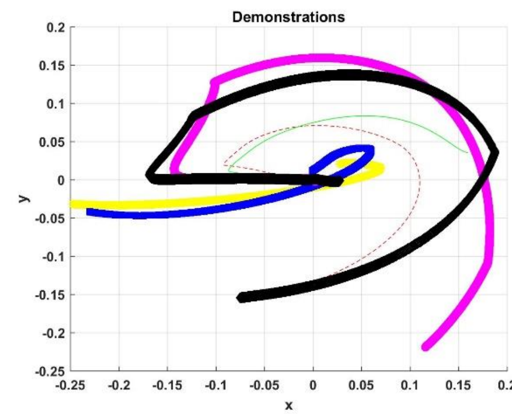
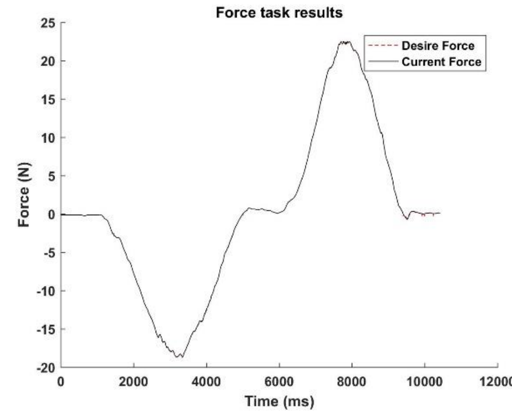


Figure 8: Position task demonstrations from different initial points.

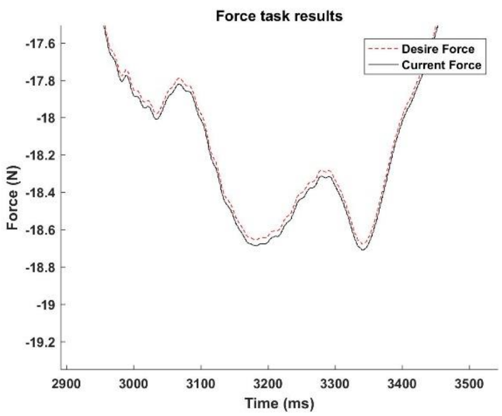
The demand for therapy services is increasing due to population ageing. This is becoming a serious problem due to the limited resources of the healthcare system, especially because of the limited number of therapists. Robots' characteristics make them an excellent device to carry out the physically demanding and repetitive tasks of therapy. In this paper, we demonstrated that the presence of therapists are not necessary throughout the entire therapy session; robots can be used to learn and continue their role and help the patients while therapists can share their time with other patients. Even though LfD is not a new tool, it has a huge potential in robotic rehabilitation in order to develop more intelligent and more reliable devices. In the future, we will implement an LfD system combined with AAN feature [8]. This improvement will allow the system to assist the patient only when it is necessary. In other words, if the patient is performing the task, then the system will not interfere. If the patient is stopping short of advancing the task, however, the system will assist the patient.

VIII. Conclusion and Future work

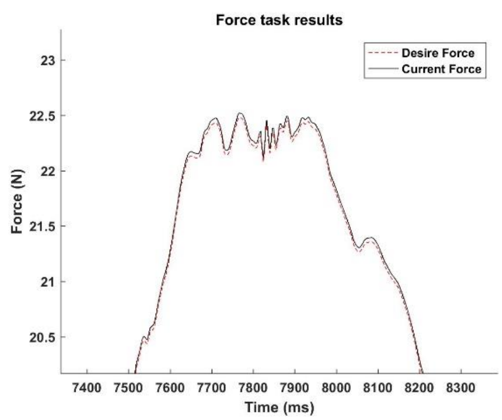
The demand for therapy services is increasing due to population ageing. This is becoming a serious problem due to the limited resources of the healthcare system, especially because of the limited number of therapists. Robots' characteristics make them an excellent device to carry out the physically demanding and repetitive tasks of therapy.



(a)



(b)



(c)

Figure 9: (a) shows the entire demonstration task, (b) shows a zoom in when the spring is compressed, and (c) shows a zoom in when the spring is expanded. Notice that in (b) and (c) the desire force tries to converge to zero.

In this paper, we demonstrated that the presence of therapists are not necessary throughout the entire therapy session; robots can be used to learn and continue their role and help the patients while therapists can share their time with other patients. Even though LfD is not a new tool, it has a huge potential in robotic rehabilitation in order to develop more intelligent and more reliable devices. In the future, we will implement an LfD system combined with AAN feature [8]. This improvement will allow the system to assist the patient only when it is necessary. In other words, if the patient is performing the task, then the system will not interfere. If the patient is stopping short of advancing the task, however, the system will assist the patient.

References

[1] S. Balasubramanian, H. Zhang, S. Buchanan, H. Austin, R. Herman, J. He, Cooperative and active assistance based interactive therapy, in: Complex Medical Engineering (CME), 2010 IEEE/ICME International Conference on, 2010, pp. 311–315.

[2] W. H. Federation, “World heart federation,” 2016. [Online]. Available: http://www.world-heart-federation.org/cardiovascular-health/stroke/. Accessed: Feb. 14, 2017.

[3] Heart & Stroke Foundation, “Statistics - heart and stroke foundation of Canada,” in *Heart & Stroke Foundation*, heartandstroke.ca, 2016. [Online]. Available: http://www.heartandstroke.com/site/c.ikIQLcMWJtE/b.3483991/k.34A8/Statistics.htm. Accessed: Feb. 28, 2017.

[4] P. Lum *et al*., “Robotic devices for movement therapy after stroke: Current status and challenges to clinical acceptance,” *Topics in Stroke Rehabilitation*, vol. 8, no. 4, pp. 40–53, 2002.

[5] S. H. Lee, I. H. Suh, S. Calinon, and R. Johansson, “Learning basis skills by autonomous segmentation of humanoid motion trajectories,” 2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012), 2012.

[6] R. Tao, Haptic Teleoperation Based Rehabilitation Systems for Task-Oriented Therapy. Edmonton, AB: UNIVERSITY OF ALBERTA, 2014.

[7] C. Martínez, J. Fong, and M. Tavakoli, “Demonstration-based learning and robotic imitation of therapist’s interaction with patient in a cooperative therapy task during telerobotic rehabilitation,” unpublished.

[8] M. Maaref, A. Rezazadeh, K. Shamaei, and M. Tavakoli, “A gaussian mixture framework for co-operative rehabilitation therapy in assistive impedance-based tasks,” IEEE Journal of Selected Topics in Signal Processing, vol. 10, no. 5, pp. 904–913, 2016.

[9] S. M. Khansari-Zadeh and A. Billard, “Learning stable nonlinear dynamical systems with gaussian mixture models,” IEEE Transactions on Robotics, vol. 27, no. 5, pp. 943–957, 2011.

[10] S. P. Bradley, A. C. Hax, and T. L. Magnanti, Applied mathematical programming. Reading, Mass.: Addison-Wesley, 1992.

[11] J. Nocedal and S. Wright, “18 Sequential quadratic programming,” in Numerical Optimization, Springer, 2006, pp. 529–562.

[12] E. C. Lu, R. Wang, R. Huq, D. Gardner, P. Karam, K. Zabjek, D. Hebert, J. Boger, and A. Mihailidis, “Development of a robotic device for upper limb stroke rehabilitation: A user-centered design approach,” Paladyn, Journal of Behavioral Robotics, vol. 2, no. 4, Jan. 2011.

[13] A. T. I. I. Automation, “ATI Industrial automation: F/T sensor gamma,” ATI Industrial Automation: F/T Sensor Gamma. [Online]. Available: http://www.atiia.com/products/ft/ft_models.aspx?id=Gamma. [Accessed: 03-Apr-2017].

[14] B. Lautrup, Physics of continuous matter: exotic and everyday phenomena in the macroscopic world. Boca Raton: CRC Press, 2011.