Learn to adapt to human walking: A Model-based Reinforcement Learning Approach for a Robotic Assistant Rollator

Georgia Chalvatzaki, Xanthi S. Papageorgiou, Petros Maragos and Costas S. Tzafestas
School of Electrical and Computer Engineering, National Technical University of Athens, Greece
{gchal, xpapag}@mail.ntua.gr, {maragos,ktzaf}@cs.ntua.gr

***Abstract*— In this paper, we tackle the problem of adapting the motion of a robotic assistant rollator to patients with different mobility status. The goal is to achieve a coupled human-robot motion in a front-following setting as if the patient was pushing the rollator him/herself. To this end, we propose a novel approach using Model-based Reinforcement Learning (MBRL) for adapting the control policy of the robotic assistant. This approach encapsulates our previous work on human tracking and gait analysis from RGB-D and laser streams into a human-in-the loop decision making strategy. We use Long-Short Term Memory (LSTM) networks for designing a Human Motion Intention Model (HuMIM) and a Coupling Parameters Forecast model, leveraging on the outcome of human gait analysis. An initial LSTM-based policy network was trained via Imitation Learning (IL) from human demonstrations in a Motion Capture setup. This policy is then fine-tuned with the MBRL framework using tracking data from real patients. A thorough evaluation analysis proves the efficiency of the MBRL approach as a user-adaptive controller.**

I. Introduction

The development of robotic mobility assistants is a major research area with corresponding impact on society. The constant increase of aged population during recent years has created new challenges in the healthcare sector, causing great difficulties for the existing care and nursing staff to keep up with these evolving needs. Thus, the necessity for robotic assistants that will help with elderly mobility and rehabilitation is important. It has been now close to twenty years since the first robotic rollators emerged [1], [2]. An intelligent robotic mobility assistant should serve many purposes; postural support, gait analysis, sit-to-stand transfer, navigation and cognitive assistance. Adaptation to user needs is of great importance for seamless human-robot interaction in such applications.

In this paper, we tackle the problem of adapting the motion of a robotic rollator that moves along with an elder user while being in front of him. The applied control should comply with the user needs even if the user wants to walk supported or unsupported by the rollator, whenever feeling confident, i.e. leaving the handles and walking along with the robot in front of them (Fig. 1). However, the robot should follow and be in a close distance in front of the user, not only to provide support, whenever needed, but also to prevent possible falls.

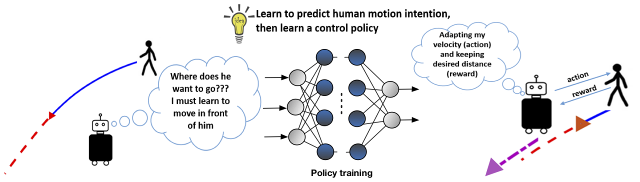

Learn to predict human motion intention,
then learn a control policy
Where does he want to go??? I must learn to move in front of him
Policy training
Adapting my velocity (action) and keeping desired distance (reward)
action
reward

Fig. 1: Robotic agent observes predicted human motion intention and learns through Model-based Reinforcement Learning to adapt its control actions, accordingly.

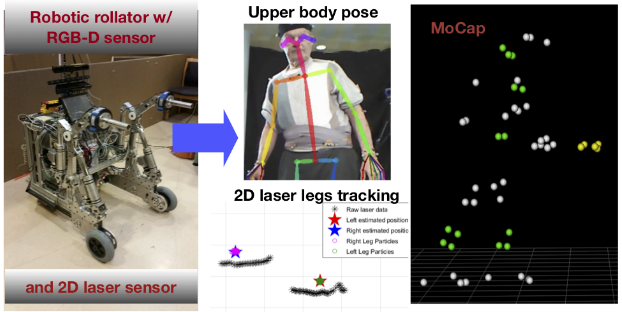


Fig. 2: **Left and middle:** Prototype robotic assistant rollator equipped with a RGB-D sensor for capturing the upper body pose and a 2D laser sensor for detecting the legs motion. **Right:** Example of the MoCap markers on an elderly user and a passive rollator, from which the data for imitation learning derived.

Motivated by this need, and taking into account the variability in human walking, and especially in pathological gait (e.g ataxic and freezing types of gait present different velocities and stepping patterns), we propose a unified method for continuous monitoring of each user and adaptation of the robotic platform's motion accordingly. We propose a MBRL method for adapting the robot's motion in front of the user. Fig. 1 encapsulates an overview of the problem that we aim to solve; the robotic assistant should infer the human's motion intention and learn a control policy using MBRL to select control actions that will comply to the human's way of walking.

We build upon our previous work, regarding human tracking and gait analysis fusing 2D laser data capturing the legs motion [3] and RGB-D streams of the upper body pose estimation using the Open Pose Library [4], from sensors mounted on a robotic rollator (Fig. 2), from which we can also infer the human gait stability [5]. In this work, we integrate the human detection and tracking approach into a human-in-the-loop control framework using MBRL for

This research has been cofinanced by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH CREATE INNOVATE (project code:T1EDK- 01248 / MIS: 5030856).

adapting the robot motion to each user.

Our main contribution is a novel MBRL approach for online motion adaptation of a robotic assistant rollator that considers human motion intentions in a front-following scenario (Fig. 1). In this framework, we start by developing LSTM based prediction models for estimating human motion intention using a history of motion tracking data. We then train models which associate the human motion orientation and the estimated stride length provided by gait analysis to the desired coupling parameters for the robot's heading and position, i.e. the desired separation distance and bearing in the human-robot frame. Further on, we use this information to train a policy for suggesting robot control actions according to the human motion intentions and expected desired coupling. We developed an initial policy model trained with IL from human demonstrations using data from motion markers (VICON system), which were placed on the human and a passive rollator frame in a series of data collection experiments (Fig. 2). Although such a model behaves well for the demonstrated cases and gives insight on how the user wants the platform to be placed in front of him/her while walking, this policy does not have experience for recovering from drift cases or unexpected detection loss of the user. To cope with such situations, the proposed MBRL framework performs fine-tuning of the initial control policy (as seen in Fig. 3), while using random sampling Model Predictive Control (MPC) for planning [6]–[8]. Detailed experimental results are presented in the paper showing the efficiency of the proposed MBRL framework for the motion adaptation of a robotic assistant rollator using data from real patients.

II. Related Work

State-of-the-art research for robotic assistants mostly relies on admittance control schemes [9], [10]. A control strategy using as control inputs human velocity and orientation was proposed in [11]. A formation control for a robot-human following scenario was presented in [12], for safely navigating blind people. In our previous work [13], we have considered a front-following problem with a kinematic controller adapting to users according to their pathological mobility class. A Reinforcement Learning (RL) shared-control for a walking aid with human intention prediction from force sensors is presented in [14].

A lot of research focuses on social robot navigation [15], i.e. robot motion planning among crowds [16], using RL. Most methods for robot navigation require pedestrians motion predictions for the robot to learn how to navigate among them in a compliant way [17]. An interaction-aware motion prediction approach for pedestrians with an LSTM-based model for learning human motion behavior was presented in [18]. In [19], deep RL was used for navigating according to social norms across crowds, while in [20], RL is used for unfreezing the robot in the crowd by taking into account the coordination between robots and detected humans. In such cases the robot does not accompany humans, but it rather learns how to move through and avoid collisions with them.

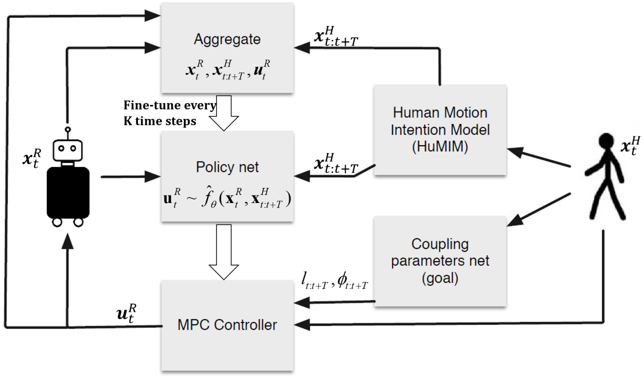

Aggregate
$x_t^R, x_{t:t+T}^H, u_t^R$
$x_{t:t+T}^H$
Fine-tune every K time steps
Human Motion Intention Model (HuMIM)
x_t^R
Policy net
$\mathbf{u}_t^R \sim \hat{f}_\theta(\mathbf{x}_t^R, \mathbf{x}_{t:t+T}^H)$
$x_{t:t+T}^H$
x_t^H
Coupling parameters net (goal)
$l_{t:t+T}, \phi_{t:t+T}$
u_t^R
MPC Controller

Fig. 3: Model-based Reinforcement Learning framework for policy adaptation using human motion intention predictions.

Regarding robotic companions, a method for human-robot navigation using the social force model and a Bayesian predictor for human motion is described in [21]. A model based on social force and human motion prediction is presented in [22], for making robots capable of approaching people with a human-like behavior, while they are walking in a side-by-side formation with a person, avoiding several pedestrians in the environment. An MPC technique that accounts for safety and comfort requirements for a robot accompanying a human in a search and rescue scenario is presented in [23].

The use of deep RL is prevalent in modern research aiming to plan robot motion [24] and control [25] for various tasks. Robot navigation systems which have integrated such RL decision-making schemes can be found in [26]–[28]. Approaches combining IL with RL for learning control policies are presented in [29], [30]. Although, model-free RL approaches have many successful applications, they require large amount of training data, which are often simulated, thus their applicability is limited. On the other hand, model-based RL firstly learns a model of the system and then trains a control policy using feedback [31]. MBRL has been used for robot control both in simulated and real world experiments [32]–[34]. MBRL relies on MPC for planning control actions, thus using learned models along with MPC as a control policy, is a matter in hand for RL and IL research [8], [35], [36]. We were inspired by recent advances in adaptive control using MBRL [7], [37]. In this work, we propose a novel MBRL framework for learning and adapting the control policy of a robotic assistant rollator to human walking. To the best of our knowledge, this is the first approach aiming to solve a front-following problem using MBRL and human motion prediction models, either for a robotic assistant or a robotic companion.

III. Preliminaries

In RL the goal is to learn a policy that will propose actions for an agent, which will maximize the sum of the expected future rewards [38]. Given the current state $x_t \in \mathbb{X}$, the agent executes an action $u_t \in \mathbb{U}$ and receives a reward $r_t = r(x_t, u_t)$, while transitioning to the next state $x_{t+1} = f(x_t, u_t) + w_t$ with initial state $x_0 \sim p(x_0)$, where f is a nonlinear function for the system's forward dynamics, w_t a Gaussian noise process and $p(x_0)$ an initial state distribution. In most cases, especially for model-free RL [39], the reward function is estimated from samples, which is a data expensive process. Model-based RL attempts to address the problem of data inefficiency by using observed data to learn the dynamics of the system. The model is used for running internal simulations of the agent's dynamics, based on which the policy is learned. The goal of MBRL is to learn an approximation of the true dynamics f. Let $\hat{f}_\theta$ be the learned discrete-time function parametrized by θ that approximates f. The objective is to find the parametrized policy $\hat{f}_\theta$ in a finite horizon that maximizes a long-term reward over a time horizon T by optimizing the parameters θ.

Since MBRL aims to learn a global dynamics model, generalization is an issue, especially for robotics applications that have to affront stochastic environments and adapt to new tasks. Thus, we resort to the option of planning through the suggested policy actions to compensate for model errors. MPC is a suitable finite horizon optimal control solution which optimizes a cost function at each time step to produce a sequence of control actions. Classic MPC relies on optimizing constrained quadratic costs, requiring first or second order approximations of the dynamics for convexity, which is sometimes difficult to meet when the dynamics are approximated by neural networks. Thus, it is useful to employ a random-sampling shooting method for MPC [6], to perform rollouts through time and simulate trajectories in a short time horizon T. In MBRL framework, MPC is used for finding the trajectory with the minimum cumulative cost over time horizon T, for which only the first action u_t of the optimal sequence is applied to the system, and then re-plan at each time-step. Therefore, such an approach compensates for model inaccuracies by preventing accumulating errors and drifting from the desired trajectory. In the context of MBRL, the reward maximization can be viewed as the equivalent cost minimization problem through MPC.

IV. Problem Statement

Our problem concerns finding the optimal control policy for adapting the robotic assistant's motion to the needs of users with different mobility status. In particular, given an estimated current human state $\mathbf{x}_t^H = [\ x^H \quad y^H \quad \upsilon_x^H \quad \upsilon_y^H\ ]^T$, where $\mathbf{p}_t^H = [\ x^H \quad y^H\ ]^T$ is the position and $\mathbf{u}_t^H = [\ \upsilon_x^H \quad \upsilon_y^H\ ]^T$ the velocity along the axes, and the human-related robot coupling parameters, i.e. the desired separation distance ℓ_t and relative human robot bearing ϕ_t, we must find an optimal control action $\mathbf{u}_t^R$ that will guarantee the compliance to the human motion intention. In other words, we aim to find a policy $f_\theta(\mathbf{x}_t^R, \mathbf{x}_t^H)$ that will propose robot control actions $\mathbf{u}_t^R = [\upsilon_t \quad \omega_t]^T$, where υ_t and ω_t are the linear and angular velocities and $\mathbf{x}_t^R = [\ x^R \quad y^R\ ]^T$ the robot position along the axes, following the objective of joint human-robot navigation. Thus, the problem includes the following optimization problem aiming to find the optimal control sequence $U_t^T = \{\mathbf{u}_t, ..., \mathbf{u}_{t+T-1}\}$, over a finite time horizon T, by minimizing the following quadratic cost:

$$U_t^T = \underset{\mathbf{u}_t,\ldots,\mathbf{u}_{t+T-1}}{\arg\min} \ \frac{1}{2} \sum_{\tau=t}^{t+T-1} \left(\mathbf{x}_\tau^R\right)^T \cdot C \cdot \mathbf{x}_\tau^R + \mathbf{c}_\tau{}^T \cdot \mathbf{x}_\tau^R$$

$$\text{s.t. } \mathbf{x}_{t+1}^R = g(\mathbf{x}_t^R, \mathbf{u}_t^R) \text{ with } \mathbf{u}_t^R \sim f_\theta(\mathbf{x}_t^R, \mathbf{x}_t^H)$$

$$\text{and } \mathbf{u}_{lb} \leq \mathbf{u}_t^R \leq \mathbf{u}_{ub} \quad (1)$$

where $C \in \mathbb{R}^{2\times 2}$ is a diagonal positive definite weight matrix, $\mathbf{c}_t = -(\mathbf{p}_t^H + \mathbf{x}_t^d)$ is the goal position with $\mathbf{x}_t^d = [\ l_t \cos(\phi_t) \quad l_t \sin(\phi_t)\ ]^T$ being the desired coupling between human and robot position along the axes in the local human-robot frame. The optimization problem is subject to the robot motion model $g(\mathbf{x}_t^R, \mathbf{u}_t^R)$ w.r.t. the unknown policy $f_\theta(\mathbf{x}_t^R, \mathbf{x}_t^H)$ and constrained by some upper $\mathbf{u}_{ub}$ and lower $\mathbf{u}_{lb}$ bounds over the linear and angular velocity commands. As transition model $g(\mathbf{x}_t^R, \mathbf{u}_t^R)$ we consider the well-known unicycle model.

V. Proposed MBRL framework

The proposed method for control policy learning for a robotic assistant rollator that will adapt its motion to the user's gait, while keeping a desired relative coupling formation (distance and bearing), is depicted in Fig. 3. At each time frame, we predict the human motion over a time horizon T and forecast the evolution of the desired coupling parameters. This information is used for sampling velocities from the control policy network that approximates the dynamics of the human-robot coupled motion, where the MPC selects the optimal control sequence according to (1). The observed human and robot states, along with the selected action, are aggregated in a dataset for adapting the control policy.

Specifically, the proposed framework addresses two core sub-problems. The first sub-problem is understanding the human motion intention. This encapsulates not only the prediction of the human future trajectory in a finite time horizon given some past knowledge, but also a model for forecasting the evolution of the desired separation distance and bearing for the same time-horizon. Our second sub-problem concerns learning an optimal control policy. This policy is dependent on the human motion observation by the robot, since we have to deal with a constant interactive human-robot coupling problem. The robot should always be in front of the human, keeping a desired separation distance and orientation and adapting its control actions according to the human's current and predicted walking states. Incorporating a human motion forecast model, helps to better decide over the best long term cost of the control actions through MPC. We rely on IL for training an initial global approximator of the control policy network from human demonstrations and use this trained model in the MBRL framework for online adaptation. In the following, we will describe the human motion intention prediction models and the proposed control policy network and their implementation within the proposed MBRL framework .

A. Human Motion Intention Prediction Models

Human motion intention prediction includes two main goals, as shown in Fig. 4. The first one concerns the human

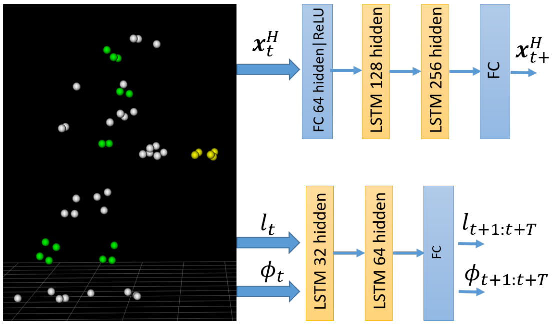


Fig. 4: Recurrent networks used for predicting future human states (HuMIM network) and the desired coupling parameters.

motion prediction in a finite-time horizon given the past human states. The second one refers to the estimation of the coupling parameters of the robot w.r.t. human. Before diving into details about the predictive models, we will briefly describe what the human state includes and how it is extracted. In our previous works, we have extensively studied human motion detection and tracking from 2D laser data along with real time gait analysis [3]. Recently, in [5] we have also used the upper body pose detection from an RGB-D sensor to perform a human Center-of-Mass (CoM) tracking by jointly using information from the pose and the legs motion (i.e. the gait velocity) to estimate the CoM motion. In this work, we use this notion of human state $\mathbf{x}_t^H$, i.e. the position and velocity of human's CoM along the axes. We exploit Motion Capture (MoCap) data to extract ground truth states and initially train our models with those smooth data and then fine-tune them with data from our tracking system.

Human Motion Intention Model (HuMIM): The HuMIM is a deep-learning network based on LSTM units [40]. LSTM constitutes a special kind of recurrent neural networks that can effectively learn long-term dependencies that exist in sequential data like in motion trajectories. This is accomplished by incorporating memory cells that allow the network to learn when to forget previous hidden states and when to update hidden states given new information. Our network architecture for HuMIM is depicted in Fig. 4. The input feature vector $\mathbf{x}_t^H$ is the current human state.[1] The network comprises a Fully Connected (FC) (Fig. 4 - blue boxes) layer, followed by a Rectified Linear Unit (ReLU) activation [41] and two LSTM (Fig. 4 - yellow boxes) layers and a FC layer that decodes the output, which is a prediction of the future human states over a time horizon T: $\mathbf{x}_{t+1:t+T}^H$.

Coupling Parameters Forecast Model: Another problem we need to solve, is to figure out the desired coupling parameters in the human-robot frame, i.e. the relative distance and bearing that will ensure the coupled human-robot motion. This is especially important for following mode cases when the robot has to follow a human from front but keeping a close distance in case assistance is needed. Those parameters are crucial for robot-control as we have already seen in Sec. IV. For computing them, we employed information from demonstrations of real patients walking with a passive rollator, while wearing motion markers (Fig. 4). We have found that the human-rollator distance, while walking, is correlated to the human stride length. Since we can apply real-time gait analysis from [3], we can compute stride lengths and use them in a prediction network that will forecast their evolution over a time window. For relative bearing we predict the human self orientation evolution in the local human frame. Therefore, given as input the current parameters ℓ_t, ϕ_t, a simple network with two LSTM layers along with a FC layer (Fig. 4), can provide the next time step predictions.

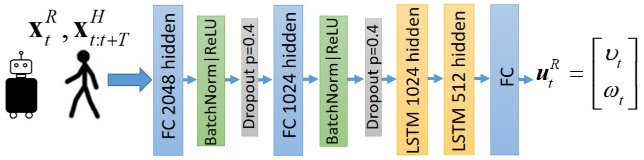


Fig. 5: Network architecture of the proposed control policy.

B. Control Policy training via Imitation Learning

We train an initial control policy for the robotic rollator following the concept of IL [42]. We benefit from the demonstrations of real patients for imitating the way they interact with the rollator while walking. The goal is to learn control actions for the robot as if the human was pushing the rollator in front of him. To this end, we have implemented the control policy network of Fig. 5. Following the narrative of Sec. IV, this network will serve as the approximator $\hat{f}_\theta$ of the true dynamics f_θ, that will propose velocity commands for the robot given the information about the human motion intention.

The proposed policy network is an LSTM-based sequence-to-sequence model using as input features the predicted human states $\mathbf{x}_{t:t+T}^H$ for a time-window T transformed w.r.t. the current robot state $\mathbf{x}_t^R$ (Fig. 5). We use two FC layers with a ReLU and a Dropout layer [43] (with probability p = 0.4) between them. The scope of the FC layers is to encode the initial features using static transformations independently of the time dependencies modelled by the LSTM units. The main encoding-decoding is implemented by the two LSTM layers. The output is decoded by the final FC layer, that gives the control action for time t, i.e. the robot velocity vector: $\mathbf{u}_t^R = [\upsilon_t \quad \omega_t]^T$.

C. Control Policy Adaptation via Model-based Reinforcement Learning

Although IL can provide good results on predicting the control actions w.r.t. the ground truth ones, its capacity is limited to the demonstrated data. Therefore, we use the learned policy via IL as as initial approximation which will be adapted to unseen human motion patterns through RL. Fig. 3 and Algorithm 1 show an overview of the proposed MBRL scheme for policy adaptation. In this setting, we resort to the HuMIM network and the coupling parameters forecast model described in Sec. V-A to predict in a time horizon T the human states $\mathbf{x}_{t:t+T}^H$ and the desired coupling

[1] We have set the initial human position to be the global reference frame.

Algorithm 1 Model-based RL for coupled HR motion

```
Require: Training data and empty dataset D for aggregation
Require: Aggregation frequency $K \in \mathbb{Z}$, MPC horizon $T \in \mathbb{Z}$
Require: Pre-trained control policy $\hat{f}_\theta$
for $i = 1, \dots$ do
    if $i$ mod $K$ = 0 then
        for $t$:1,..,$T$ do
            get future human states $\mathbf{x}^H_{t:T+T}$ via HuMIM net
            get desired coupling parameters $l_{t:t+T}, \phi_{t:t+T}$
            sample $N_s$ velocities from policy $\mathbf{u}^R_t \sim \hat{f}_\theta(\mathbf{x}^R_t, \mathbf{x}^H_{t:T+T})$ and add exploration noise
            perform MPC rollouts to find the optimal control sequence $U^T_t$ using (1)
        execute first action $\mathbf{u}^R_t$ from selected sequence $U^T_t$
        add $(\mathbf{x}^R_t, \mathbf{x}^H_{t:T+T}, \mathbf{u}^R_t)$ in dataset D
    else
        perform fine-tuning on policy of Fig. 5 using the aggregated data D
```

parameters $l_{t:t+T}, \phi_{t:t+T}$ deriving from the current estimated stride length and human self orientation. Those parameters will be used to form the desired coupling state $\mathbf{x}^d_t$ for the MPC controller in (1) as described in Sec. IV. At each time step we use the predicted human states provided by HuMIM, transformed in the respective robot frame, to sample N_s new velocities from the policy. We use the dropout layer to apply Monte Carlo Markov Chain sampling on the network's outputs to take advantage of the network's uncertainty [43]. We also apply extra random exploration noise on the sampled velocities in order to widen the sample distribution.

Moreover, we apply random white noise on the estimated robot state $\mathbf{x}^R_t$ to simulate possible errors in real case scenarios like localization errors, drifting, etc. Our aim is to learn policies for recovering the robot from false states by applying the best possible control action. The N_s velocity samples are used for the MPC rollouts by simulating trajectories over a finite time horizon T for each sample. The simulated trajectory with the minimum cost computed by (1) (i.e. highest reward in the RL narrative) is selected, while only the first action from the selected control sequence is applied to the robot. Through re-planning at each time step, we compensate for possible model errors. The robot state along with the applied control action and the current human state $\mathbf{x}^R_t, \mathbf{x}^H_{t:T+T}, \mathbf{u}^R_t$ (Fig. 3, Alg. 1) are aggregated in a new, initially empty, dataset D intending to be used for policy adaptation by fine-tuning the network every $K \in \mathbb{Z}$ time steps. The on-policy data aggregation and retraining of the model adapts the policy to new state-action tuples possibly previously unseen to the network by augmenting the respective distributions and improving the controller's performance.

VI. Experimental Results

A. Experimental setup & data

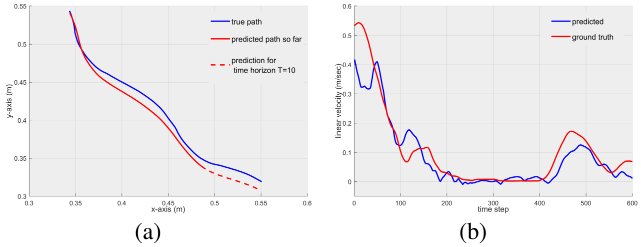


Fig. 6: (a) Example of a predicted human path with the HuMIM network. (b) Example of predicted robot's linear velocity from the IL policy network.

The data used in this work were collected in Agaplesion Bethanien Hospital - Geriatric Center in Heidelberg with the participation of fourteen patients. The participants presented moderate to mild mobility impairment, according to clinical evaluation. The subjects had to perform several everyday life scenarios using a passive robotic rollator, used for the purpose of data collection, while wearing motion markers from a MoCap setup (Fig. 2). The subjects had to perform several experimental scenarios in a special hospital room, walking supported (i.e. holding on the rollator) or unsupported (i.e. the rollator was "following" the human from a close distance without physical interaction). The data were collected by a Kinect v.1 sensor and a Hokuyo UBG-04LX-F01 laser sensor that were mounted on the rollator (Fig. 2). For the purpose of this work, we have employed 20.000 frames of MoCap data (synchronized to the laser frame rate, i.e. 0.028sec/scan), and a dataset of approximately 5.000 frames of human CoM tracking data from four patients from the supported mode scenarios used for fine-tuning/testing our models and for training MBRL. An extra dataset of 2.000 tracking data were kept for the experimental testing of the MBRL framework. In the following, we provide detailed results that demonstrate the efficiency of the proposed models and the performance of the proposed MBRL method.

B. Evaluation of Human Motion Intention Prediction Models

Implementation of HuMIM network: We have trained HuMIM using the MoCap and tracking data with a 80%-20% partition for training and testing respectively, for 500 epochs with learning rate 10^{-4} and weight decay 10^{-4}. For training we have used Stochastic Gradient Descent optimizer with the Mean Squared Error (MSE) loss (L2-loss) computed between the predicted $\mathbf{x}^H$ and the ground truth, $\hat{\mathbf{x}}^H$ of the output features.

Evaluation: To evaluate the HuMIM network we compute the MSE loss for the training and testing datasets, which gives an indication of the overall prediction performance of our models. More specifically, the MSE training loss was $4 \cdot 10^{-4}$ while the testing loss for a $T = 10$ prediction horizon was $2 \cdot 10^{-3}$, meaning that our model provides a very good fit on the data and accurate future predictions for the human motion intention. Fig. 6a depicts an example of a predicted path w.r.t. to the ground truth human path, where the dashed line shows an example of the forecast path for ten time steps.

Implementation of the Coupling Parameters forecast models: From our analysis we have found that the stride length and actual human-robot distance data are correlated, with correlation coefficient $\rho = 0.972$ and a mean difference

Arch.	1LSTM	2 FC	1FC + 2LSTM	2FC+1LSTM	2FC+2LSTM
hidden states	512	[1024, 512]	[2048,1024,512]	[2048,1024,512]	[2048,1024,1024,512]
Train Loss	0.186	0.184	0.045	0.052	**0.029**
Test Loss	0.158	0.156	0.057	0.065	**0.043**

TABLE I: Evaluation results for various architecture designs for the Control Policy Network of Fig. 5 in the IL setting.

between them $\delta\ell = 0.15$ m. In following cases, $\delta\ell$ is used as a constant bias added to the predictions of the desired separation as a safety distance. For implementation we have used the same training parameters and loss function as for HuMIM.

Evaluation: For the desired coupling parameters, we acquired equally good model fittings and MSE losses. For the relative bearing parameter ϕ_t the training loss was $2.2 \cdot 10^{-3}$ rad and testing loss $2.8 \cdot 10^{-3}$ rad, while for the relative separation (trained both on demonstrated data and the extracted stride lengths from our tracking framework), the train loss was $6 \cdot 10^{-3}$ m and test loss 10^{-2} m. Those results show that our trained models can effectively predict human motion intentions in a robot-human joint walking framework.

C. Evaluation of IL Control Policy

Implementation: For training the control policy network of Fig. 5 with IL, we have used the MoCap data, from which we have computed the human states and the robot's ground truth velocities. For the IL training of the network we have not used HuMIM for human predictions, but we rather packed the training data into time-overlapping feature vectors $\mathbf{x}^H_{t:t+T}$. The network was trained using mini-batches of 512 clips, with initial learning rate 10^{-3}, momentum 0.9 and weight decay 10^{-4}. The learning rate is divided by 10 after half the epochs. We used Adam optimizer and for imitation loss we have employed the L1-loss, i.e. the mean absolute error between the predicted $\hat{\mathbf{u}}^R_t$ and the actual $\mathbf{u}^R_t$ robot velocities.

Evaluation: Table I provides the training and testing L1 losses for the predicted control velocities w.r.t. the ground truth ones for the IL setup. It is evident that the LSTM layers are a requisite for decoding the sequences of human motion. Moreover, the combination of FC and LSTM layers seems to provide the necessary encoding-decoding scheme for translating a predicted human trajectory (considering that humans move in an holonomic way) into linear and angular velocities for the robotic assistant. We choose the architecture with the 2 FC and 2 LSTM layers since it is the one having the smallest prediction error. An example of the performance of the proposed policy network is depicted in Fig. 6b, where we compare the predicted linear velocity w.r.t. ground truth from the testing dataset of the IL policy.

D. Evaluation of the MBRL approach:

Implementation: For the fine-tuning process of the control policy network of Fig. 5 according to the MBRL framework (Fig. 3), we use the Adam optimizer and the Huber loss, which is less sensitive to outliers in data:

$$L_\varepsilon = \begin{cases} \frac{1}{2}\sum \left\| u^R_t \hat{-} \mathbf{u}^R_t \right\|^2, \text{ for } \left| u^R_t \hat{-} \mathbf{u}^R_t \right| \leq \varepsilon \\ \varepsilon \left| u^R_t \hat{-} \mathbf{u}^R_t \right| - \frac{1}{2}\varepsilon^2, \text{ otherwise} \end{cases} \tag{2}$$

where $\varepsilon > 0$ is a small value. Below we evaluate different design decisions regarding the MBRL setup. For the MBRL training we have employed 5000 frames of tracking data from four patients, while for testing the controller's performance we have kept 2000 data from one patient unseen to the training set.

Evaluation of MBRL training: For the MBRL training procedure we have explored different design parameters for acquiring the best possible solution to our problem. Fig. 7 presents the learning curves, which represent the cumulative costs for the task of human-robot coupled motion, for different design parameters. Since we are considering costs, the lower the cumulative cost, the better the performance by the corresponding MBRL setting. The best outcome from this evaluation will be considered for testing with a new patient.

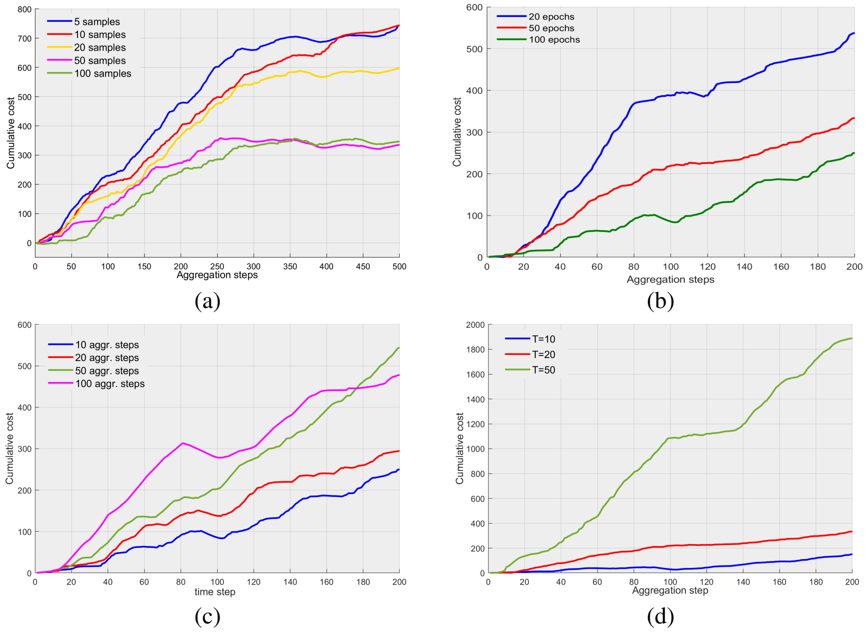
Specifically, in Fig. 7a the impact of different number of samples N_s used for the MPC rollouts is presented, for a range of 5-100 samples. The outcome seems reasonable, since for less samples (i.e. 5-20) the limited exploration by the controller leads to accumulating larger errors and thus costs. Interestingly, the learning curve of 50 samples behaves the same as the curve for 100 samples, while both parameter settings converge very quickly at a steady performance. The slopes of the curves for 50 and 100 samples show that after 250 aggregation steps we have a stabilized performance. We choose to use 50 samples as a computationally cheaper solution.

For selecting the number of epochs used at each aggregation step for adapting our policy we have experimented with 20, 50 and 100 epochs. Fig. 7b shows the cumulative costs for 200 aggregation steps and 50 samples for the MPC. It is obvious that 100 epochs of training has the best performance, however the 50 epochs setting follows closely, thus we will select those for computational reasons.

In Fig. 7c we present the evaluation results for different aggregation frequencies K (Algorithm 1). From experimentation, we have found that aggregating and adapting the control policy every 10 time steps yields better performance to the proposed algorithm. It is important to note, that those timings have been chosen to resemble timings in human gait. Specifically, we know from previous work [3], that approximately every 10 time frames a human performs a leg swing for stepping through and about every 50 time frames a gait cycle is completed. Therefore, we notice that adaptation for each stepping yields lower cumulative costs.

In the same way, we have explored different time-horizon settings for our MPC (we have changed the settings of our prediction and policy networks accordingly for this experiment). Longer time-horizons than $T = 10$ accumulate greater errors. Longer horizons mean larger prediction errors from the human motion intention models, hence leading to greater errors for the policy estimation. Evidently, adaptation in a frequency relative to the gait's swing phase is more appropriate.

According to the above evaluation, we have decided to employ the implementation of MBRL using $N_s = 50$ samples for the MPC rollout, 50 epochs for policy fine-tuning, $K =$



(a)

(b)

(c)

(d)

Fig. 7: Learning curves for the MBRL method presenting the cumulative costs for various design parameters: (a) Number of sampled trajectories N_s. (b) Number of epochs used for fine-tuning the policy. (c) Aggregation frequency K. (d) Prediction horizon T.

10 time-steps aggregation frequency and $T = 10$ prediction horizon. This implementation had an average Root MSE (RMSE) 0.068 cm regarding the coupling error w.r.t. the actual one and average Huber loss for fine-tuning was 0.031.

Evaluation results for human-following: For the task of human front-following, we evaluate the MBRL approach with tracking data from a new patient, i.e. unseen to all the training procedures described above. We aim to investigate how the control policy can adapt to a new patient with medium mobility impairment, while following from front as if the user was pushing the robotic rollator. Fig. 8 presents graphs comparing paths, linear velocities and separation distances from the MBRL approach w.r.t. the actual data provided by the MoCap analysis.

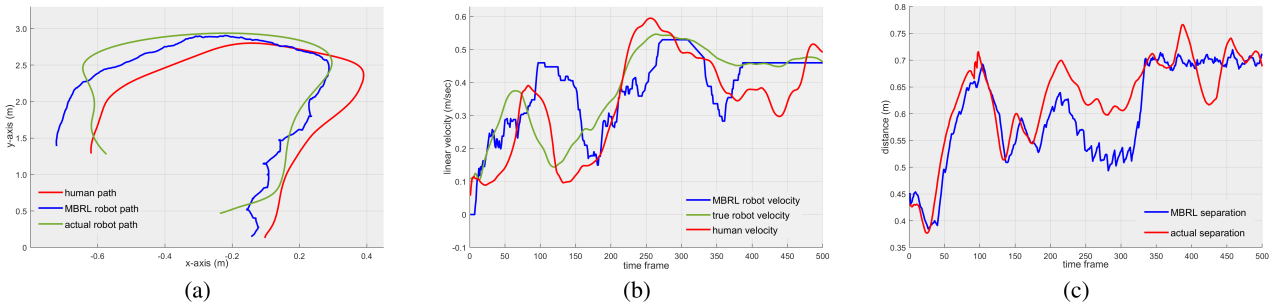
In Fig. 8a we are comparing the paths performed by the tracked human for 500 time frames, the one derived by the MBRL process for the robot and the actual one derived by the MoCap data when the patient pushed the rollator during the data collection experiments. It is evident that the MBRL method provides a trajectory very close to the actual one. This can also be demonstrated by the results in Fig. 8b, where we compare the linear velocities for the patient, the ones proposed by MBRL and the actual rollator. The MBRL velocity decisions follow closely the human velocity patterns. It is interesting to mention that in contrast to Fig. 6b presenting the IL results where the policy followed the actual rollator velocities, now we can see that the policy has adapted to the actual motion pattern of the patient. There is however a small lag, of approximately 20 time frames in detecting turning points, which might also be inherited by the HuMIM network performance. Finally, we compare the separation distance in the human-robot coupling. Yet again, the MBRL policy follows the actual pattern, meaning also that using the stride length as inference for the desired separation distance is a valid assumption. Concluding this analysis we provide results over all 2000 tracking frames for the task of patient following. The average RMSE between the MBRL proposed robot path and the actual one is **0.18m**, the RMSEs for velocities are **0.15 m/sec** for linear and **0.24 rad/sec** for angular velocity. The average RMSE for human-robot separation distance w.r.t. the actual one is **0.22m**.

VII. Conclusion & Future Work

We proposed a novel approach using Model-based Reinforcement Learning for adapting the motion of a robotic assistant rollator to the walking patterns of elderly and patients with various mobility inabilities. The aim is to develop a control policy that will propose optimal control actions for coupling the robot's motion with each user, as if the user was pushing the rollator. In this setting, we consider the problem to be a front-following human-robot coupled motion. To this end, we have designed LSTM-based networks for predicting future human motion intentions and forecasting the desired coupling parameters in the robot-human setting. An initial control policy that suggests control velocities given the human kinematic state evolution in a short time-horizon was trained through IL. In the MBRL framework we adapt this policy employing a MPC planner and using tracking data from patients. Through extensive experimentation with real data, we provide evidence that prove MBRL to be efficient as a decision making approach for a user-adaptive controller in a robotic assistant rollator. In our future work, we plan to test different planning methods in the MBRL framework and a combination with model-free methods.

References

[1] S. Dubowsky, F. Genot, S. Godding, H. Kozono, A. Skwersky, and and, "Pamm - a robotic aid to the elderly for mobility assistance and monitoring: a "helping-hand" for the elderly," in *ICRA 2000*.



(a) (b) (c)

Fig. 8: Testing results of the MBRL policy adaptation for a new patient. (a) Human, robot and ground truth paths (RMSE **0.18m** MBRL vs. actual). (b) Comparison of human, robot and actual robot linear velocities (RMSE **0.15 m/sec** MBRL vs. actual). (c) Comparison of the MBRL separation distance w.r.t. the actual one (RMSE **0.22m**).

[2] A. Morris, R. Donamukkala, A. Kapuria, A. Steinfeld, J. T. Matthews, J. Dunbar-Jacob, and S. Thrun, "A robotic walker that provides guidance," in *ICRA '03*.

[3] G. Chalvatzaki, X. S. Papageorgiou, C. S. Tzafestas, and P. Maragos, "Augmented human state estimation using interacting multiple model particle filters with probabilistic data association," *IEEE RAL*, 2018.

[4] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *CVPR*, 2017.

[5] G. Chalvatzaki, P. Koutras, J. Hadfield, X. S. Papageorgiou, C. S. Tzafestas, and P. Maragos, "On-line human gait stability prediction using lstms for the fusion of deep-based pose estimation and lrf-based augmented gait state estimation in an intelligent robotic rollator," *ICRA '19 (to appear)*. [Online]. Available: http://arxiv.org/abs/1812.00252

[6] A. G. Richards, "Robust constrained model predictive control," *Ph.D. dissetration*, 2004.

[7] A. Nagabandi, G. Kahn, R. S. Fearing, and S. Levine, "Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning," in *ICRA '18*.

[8] G. Williams, N. Wagener, B. Goldfain, P. Drews, J. M. Rehg, B. Boots, and E. A. Theodorou, "Information theoretic mpc for model-based reinforcement learning," in *ICRA '17*.

[9] M. Geravand et. al, "An integrated decision making approach for adaptive shared control of mobility assistance robots," *Int'l J. of Social Robotics*, 2016.

[10] Y. Hirata et. al, "Motion control of intelligent passive-type walker for fall-prevention function based on estimation of user state," in *ICRA 2006*.

[11] C. A. Cifuentes and A. Frizera, *Development of a Cognitive HRI Strategy for Mobile Robot Control*, 2016.

[12] S. Scheggi et al., "Cooperative human-robot haptic navigation," in *ICRA '14*, 2014.

[13] G. Chalvatzaki, X. S. Papageorgiou, P. Maragos, and C. S. Tzafestas, "User-adaptive human-robot formation control for an intelligent robotic walker using augmented human state estimation and pathological gait characterization," in *IROS '18*.

[14] W. Xu, J. Huang, Y. Wang, C. Tao, and L. Cheng, "Reinforcement learning-based shared control for walking-aid robot and its experimental verification," *Advanced Robotics '15*.

[15] T. Kruse, A. K. Pandey, R. Alami, and A. Kirsch, "Human-aware robot navigation: A survey," *Robotics and Autonomous Systems '13*.

[16] P. Ciou, Y. Hsiao, Z. Wu, S. Tseng, and L. Fu, "Composite reinforcement learning for social robot navigation," in *IROS '18*.

[17] M. Fahad, Z. Chen, and Y. Guo, "Learning how pedestrians navigate: A deep inverse reinforcement learning approach," in *IROS '18*.

[18] M. Pfeiffer, G. Paolo, H. Sommer, J. Nieto, R. Siegwart, and C. Cadena, "A data-driven model for interaction-aware pedestrian motion prediction in object cluttered environments," in *ICRA '18*.

[19] Y. F. Chen, M. Everett, M. Liu, and J. P. How, "Socially aware motion planning with deep reinforcement learning," in *IROS '17*.

[20] T. Fan, X. Cheng, J. Pan, P. Long, W. Liu, R. Yang, and D. Manocha, "Getting robots unfrozen and unlost in dense pedestrian crowds," *RA-L '19*.

[21] G. Ferrer, A. Garrell, and A. Sanfeliu, "Robot companion: A social-force based approach with human awareness-navigation in crowded environments," in *IROS '13*.

[22] E. Repiso, A. Garrell, and A. Sanfeliu, "Robot approaching and engaging people in a human-robot companion framework," in *IROS '18*.

[23] D. Lee, C. Liu, Y. Liao, and J. K. Hedrick, "Parallel interacting multiple model-based human motion prediction for motion planning of companion robots," *IEEE Trans. on Automation Science and Engineering*, 2017.

[24] C. Finn and S. Levine, "Deep visual foresight for planning robot motion," in *ICRA '17*.

[25] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *CoRR*, 2015. [Online]. Available: http://arxiv.org/abs/1509.02971

[26] P. Mirowski, R. Pascanu, F. Viola, H. Soyer, A. J. Ballard, A. Banino, M. Denil, R. Goroshin, L. Sifre, K. Kavukcuoglu, D. Kumaran, and R. Hadsell, "Learning to navigate in complex environments," *CoRR*, 2016. [Online]. Available: http://arxiv.org/abs/1611.03673

[27] J. Zhang, J. T. Springenberg, J. Boedecker, and W. Burgard, "Deep reinforcement learning with successor features for navigation across similar environments," in *IROS '17*.

[28] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi, "Target-driven visual navigation in indoor scenes using deep reinforcement learning," in *ICRA '17*.

[29] M. Pfeiffer, S. Shukla, M. Turchetta, C. Cadena, A. Krause, R. Siegwart, and J. Nieto, "Reinforced imitation: Sample efficient deep reinforcement learning for mapless navigation by leveraging prior demonstrations," *IEEE RAL '18*.

[30] Y. Zhu, Z. Wang, J. Merel, A. A. Rusu, T. Erez, S. Cabi, S. Tunyasuvunakool, J. Kramr, R. Hadsell, N. de Freitas, and N. Heess.

[31] M. P. Deisenroth, G. Neumann, and J. Peters, "A survey on policy search for robotics," *Foundations and Trends in Robotics*, 2013.

[32] F. Ebert, C. Finn, S. Dasari, A. Xie, A. X. Lee, and S. Levine, "Visual foresight: Model-based deep reinforcement learning for vision-based robotic control," *CoRR*, 2018. [Online]. Available: http://arxiv.org/abs/1812.00568

[33] D. Meger, J. C. G. Higuera, A. Xu, P. Gigure, and G. Dudek, "Learning legged swimming gaits from experience," in *ICRA '15*.

[34] H. Durrant-Whyte, N. Roy, and P. Abbeel, *Learning to Control a Low-Cost Manipulator Using Data-Efficient Reinforcement Learning*.

[35] K. Lee, K. Saigol, and E. Theodorou, "Safe end-to-end imitation learning for model predictive control," *CoRR*, 2018. [Online]. Available: http://arxiv.org/abs/1803.10231

[36] B. Amos, I. Jimenez, J. Sacks, B. Boots, and J. Z. Kolter, "Differentiable MPC for End-to-end Planning and Control," 2018.

[37] I. Clavera, A. Nagabandi, R. S. Fearing, P. Abbeel, S. Levine, and C. Finn, "Learning to adapt: Meta-learning for model-based control." [Online]. Available: http://arxiv.org/abs/1803.11347

[38] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*. MIT Press, 1998.

[39] J. Peters and S. Schaal, "Reinforcement learning of motor skills with policy gradients," *Neural Networks '08*.

[40] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput. 1997*.

[41] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *ICML'10*.

[42] S. Schaal, "Learning from demonstration," in *NIPS'96*.

[43] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *ICML '16*.