

Predicting Human Intent for Cooperative Physical Human-Robot Interaction Tasks

Harsh Maithani, Juan Antonio Corrales, Youcef Mezouar

▶ **To cite this version:**

Harsh Maithani, Juan Antonio Corrales, Youcef Mezouar. Predicting Human Intent for Cooperative Physical Human-Robot Interaction Tasks. 2019 IEEE 15th International Conference on Control and Automation (ICCA), Jul 2019, Edinburgh, United Kingdom. pp.1523-1528, 10.1109/ICCA.2019.8899490 . hal-02422909

HAL Id: hal-02422909
https://uca.hal.science/hal-02422909

Submitted on 23 Dec 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Predicting Human Intent for Cooperative Physical Human-Robot Interaction Tasks

Harsh Maithani, Juan Antonio Corrales Ramon and Youcef Mezouar

***Abstract*— In this paper, a robot assistive Impedance and Admittance control methodology is proposed for a cooperative physical human-robot interaction (pHRI) task. In a pHRI task in which the human is the leader, the robot is a passive follower as the human intention of desired motion and force to be applied are unknown. It is generally difficult to predict the intention of the human leader. Recurrent Neural Networks (RNN) with Long Short-Term Memory (LSTM) units are employed to forecast the position,velocity and force anticipated to be applied by the human. The estimated parameters are integrated into the impedance and admittance controllers via a target impedance model which aids the robot in becoming a proactive partner of the human by sharing the physical load. The same methodology is also applied to the Minimum Jerk model which allows the robot to follow the Minimum Jerk trajectory without knowing the trajectory parameters in advance.**

I. INTRODUCTION

We are living in times where technology has made rapid advancements in every sphere of human life and activity. Robots are becoming omnipresent and it is only a matter of time before they become an essential part of human households. Robots have been used in industrial settings since many decades, however due to safety purposes industrial robots have been traditionally kept isolated from human operators. As such collaboration between humans and robots has been limited. But now there is a growing emphasis on incorporating robots in more intimate environments alongside humans to make full use of both human and robot capabilities by working in teams. This has resulted in the development of new generation robots which are lightweight (e.g KUKA LWR), can be programmed on the fly etc. Some examples where physical human-robot interaction (pHRI) has been used recently are - collaborative object transportation [1], object swinging [2], assistive welding [3], car windshield positioning [4] etc. Impedance Control and Admittance Control are popular control techniques used for physical human-robot interaction tasks as they establish a dynamic relationship between the robot and the environment. Hogan [5] introduced the Impedance Control scheme for robotic manipulators. The equations for both the Impedance and Admittance Control are the same but the inputs/outputs are different. In the Impedance Control scheme the input is displacement/velocity and output is force whereas in the Admittance Control scheme the input is force and output is displacement/velocity.

In the field of computational neuroscience and motor control there has been significant investigation into the mathematical modeling of natural human limb motion, in both unconstrained and constrained environments. The main models that have been found to explain the motion are the Minimum Jerk Model (MJM) [6], the Minimum Torque-Change Model [7] and Fitts Law [8]. Although these models were determined with an individual human, later it was established that they are applicable even in dyadic interactions. An important aspect of pHRI is the component of transparency. The system becomes more transparent if the robot partner behaves like a human and shows predictable behaviour. As such the impedance parameters of the robot controller should be determined to enable transparent cooperation between the human and the robot. It was shown in [9] that the MJM is followed in Human-Human Interaction tasks. [10] used the MJM to determine the human impedance parameters in a pHRI task. In [11] it was demonstrated experimentally that the MJM is applicable even for pHRI tasks. In several works involving Human-Robot Interaction, the efficacy of the proposed controller is measured by determining if it allows the human to follow the Minimum Jerk trajectory naturally [12]. One disadvantage of the MJM model is that it requires



Fig. 1. A cooperative physical Human-Robot Interaction task. The user moves the tool continuously between two points 30 cms apart along a single degree of freedom, using either Impedance or Admittance Control, with a laser pointer attached to the end-effector of the robot. The position and velocity of the end-effector are recorded, as well as the force applied by the human. An RNN with LSTM units is trained over each of the datasets to predict the desired human motion.

All the authors are with Université Clermont Auvergne, CNRS, SIGMA Clermont, Institut Pascal, F-63000 Clermont-Ferrand, France. Corresponding author email address: harshmaithani09@gmail.com

knowledge of the trajectory parameters in advance. Also, it is not flexible to accommodate sudden changes in the trajectory being pursued by the human. Furthermore in [13] it was showed that for constrained tasks that involve an exchange of forces, the resulting trajectory complies more with the Minimum Torque-Change model. However inspite of this, the MJM model is used popularly for simple pHRI tasks where the robot controller has to be developed or the performance of the controller studied. In [14] a Fuzzy Inference System was tuned to the MJM to output the damping coefficient to be used in a variable admittance control scheme. In [15] a Reinforcement Learning algorithm was used to determine the damping coefficient for a variable admittance controller while complying with the MJM without any knowledge of the initial and final position or the duration of the motion.

In pHRI tasks generally the human and the robot act as either a leader or a follower. If the human is the leader and the human intent is unknown to the robot then the robot is limited to the role of a passive follower and the robot assistance is limited, which might even obstruct the free motion of the human. Several works have focused on estimating the human intent to make the robot an active follower. In [16] a modified Hidden Markov Model (HMM) was used to determine the human-intended path in a pHRI task. In [19] force and motion data were encoded into a HMM for a joint load transport problem using multiple observations where the partner of the human was a mobile robot. The MJM was used in a pHRI task in [17], and by estimating the unknown parameters in real-time the MJM gave the predicted motion of the human partner. Similarly [18] estimated the time duration of motion in a simplified MJM using an extended Kalman Filter. [20] used an extended Kalman Filter to predict the motion of the human, and the accuracy of predictions was used to fluctuate the role of the robot between a leader and a follower. Radial Basis Function Neural Networks were used in [21] to estimate the human motion intention. This work was extended in [24] to adapt to human motion intention changes during the motion itself. The Learning from Demonstration methodology was used in [22], where Gaussian Mixture Models were utilized to learn the task model and used for a collaborative carrying and positioning of load task, which was then integrated with an adaptive impedance controller.

The RNN-LSTM algorithm [25] has been popular recently for prediction and classification using long time-series data. In [26] RNN-LSTM was used for predicting pedestrian motion in cluttered environments, whereas in [27] it was used for anticipating driver activity. To the best of our knowledge the RNN-LSTM architecture has not been used before for motion estimation/prediction for cooperative human-robot tasks.

In this paper we will consider a cooperative human-robot task in which the human user holds a tool which is mounted to the end-effector of a robot along with a force-torque sensor. The robot is constrained to move along a single cartesian direction. Using either Impedance Control or Admittance Control the user moves the tool between two specified points repetitively according to his own comfortable speed. We record 3 parameters - the position and velocity of motion directly from the robot controllers and the force applied from the force sensor. We were able to generate a huge dataset for each of the 3 recorded parameters (which we refer to as the 'Natural Motion' dataset). Furthermore, using the Minimum Jerk model [6] we can generate another dataset independently. Using a RNN-LSTM architecture we can train one separate neural network over each of the 4 datasets and predict the future values of each variable using input data. These predicted values can be integrated into the controllers to make the robot a proactive partner and share the load of the human in the task.

The contributions of our work are as follows -

1) For periodic tasks, this methodology can be used to encode the motion data.
2) For the Minimum Jerk model, the trajectory parameters such as the distance to be traversed and the time duration are required to be known in advance. Using the trained algorithm over the 'Minimum Jerk Model' dataset, we can eliminate this requirement.
3) By training the algorithm over velocities in the Natural Motion dataset, the reference velocity can be acquired using the trained algorithm instead of the MJM. As the trained algorithm is customized for a particular person, hence the assistance offered by the robot will be natural and intuitive to him.
4) By training the algorithm over applied forces in the Natural Motion dataset, the robot can provide the assistive force itself and lessen the load on the human partner.
5) The trained algorithm for one person can be used as a reference for another person i.e this methodology can be used for training purposes.

II. Preliminaries

A. Impedance Control

For a robot the kinematics are written as

$$x(t) = f(q) \tag{1}$$

where $x(t) \in \Re^n$ and $q \in \Re^n$ are positions/orientations in the Cartesian space and coordinates in the joint space respectively. Differentiating (1) with respect to time results in

$$\dot{x}(t) = J(q)\dot{q} \tag{2}$$

where $J(q) \in \Re^{n \times n}$ is the Jacobian matrix. Differentiating (2) with respect to time results in

$$\ddot{x}(t) = \dot{J}(q)q + J(q)\ddot{q} \tag{3}$$

The robot arm dynamics in the joint space for a non-redundant robot are given by

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau - J^T(q)F_r \tag{4}$$

where $M(q) \in \Re^{n \times n}$ is the symmetric bounded positive-definite inertia matrix; $C(q,\dot{q})\dot{q} \in \Re^n$ is the Coriolis and Centrifugal forces;$G(q) \in \Re^n$ is the gravitational force; $\tau \in$

$\Re^n$ is the vector of control input; $F_r \in \Re^n$ denotes the force exerted by the robot on the environment at the end-effector. As we are considering a pHRI task so the environment here is the human, specifically the human hand. In the Cartesian space the robot dynamics can be written as

$$M_x(q)\ddot{x} + C_x(q,\dot{q})\dot{x} + G_x(q) = u - F_r \tag{5}$$

where $M_x(q) = J^{-T}(q)M(q)J^{-1}(q)$,
$C_x(q,\dot{q}) = J^{-T}(q)(C(q,\dot{q}) - M(q)J^{-1}(q)\dot{J}(q))J^{-1}(q)$,
$G_x(q) = J^{-T}(q)G(q)$, $u = J^{-T}(q)\tau$

The most common interaction controllers for physical human-robot tasks are Admittance and Impedance Control. Both controllers need a target impedance model for the robot and differ only with regards to the input and output. The target impedance model can be written as a mass-damper-spring equation

$$M_d(\ddot{x} - \ddot{x}_d) + B_d(\dot{x} - \dot{x}_d) + K_d(x - x_d) = F_d \tag{6}$$

where M_d, B_d, K_d are the virtual inertia,damping and stiffness of the robot respectively. F_d is the desired force and x_d can be interpreted as the rest position of the virtual mass-damper-spring system.

For physical human-robot interaction tasks in which a human arm is in contact with the robot the human arm can similarly be modeled as a mass-damper-spring system

$$M_h\ddot{x}_h + B_h\dot{x}_h + K_h(x_h - x_{hd}) = F_h \tag{7}$$

where M_h, B_h, K_h are the limb inertia,damping and stiffness respectively and F_h is the force applied by the arm at the wrist. The limb impedance values are not fixed and depend from person to person as well as on the task being carried out. x_h is the position of the human wrist in the robot frame and x_{hd} can be interpreted as the desired target position. Discussion on the limb impedance parameters or their calculation are not in the purview of this paper.

The Impedance/Admittance Control scheme in general is a "Reactive" Control Scheme. Predicting or anticipating any of the factors mentioned above for future time steps constitutes "Predictive/Anticipatory" Control. For "Direct Teaching Mode" i.e for the robot to freely follow the motion of the human we can set $K_d = 0$ or $x = x_d$. This essentially eliminates the spring component of the robot impedance model and hence avoids any restoring forces.

In this paper we make the assumption that the human holds the tool rigidly such that the forces are transmitted completely to the robot. We also assume that the hand is close enough to the end effector to assume that the positions, velocities and forces discussed in this paper are of/at the end-effector of the robot. Thus we can assume $x_d = x_{hd}$, $\dot{x}_h = \dot{x}$ and $\ddot{x}_h = \ddot{x}$.

In Admittance Control (popularly called Position-based Impedance Control) the input is the force applied by the environment and the output is displacement/velocity as shown in Fig. 2. In our case the force is applied by the human on the tool and measured by the force-torque sensor i.e $F_r = F_d = -F_h$. The desired position x_d calculated using (6) is passed to the Position Controller of the robot.

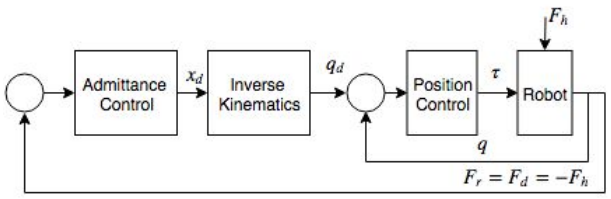


Fig. 2. Admittance Control (Position-based Impedance Control)

In Impedance Control (Torque-based Impedance Control) the input is displacement/velocity and output is force as shown in Fig. 3. The equation of the dynamics becomes

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau + J^T(q)(-F_r + F_h) \tag{8}$$

where $F_r = F_d$

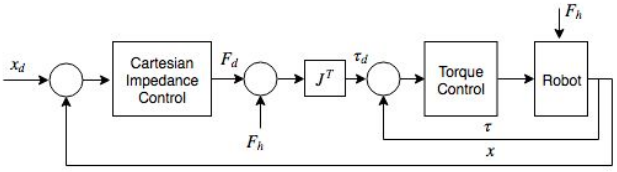


Fig. 3. Torque-based Impedance Control

From the previous equations we can see that in a robot Admittance/Impedance Controller we can

1) Vary the Impedance parameters M_d, B_d, K_d of the robot model. This is known as Impedance Shaping. The parameters can be varied according to the limb impedance parameters M_h, B_h, K_h or $\dot{x}/\ddot{x}/F_h/\Delta F_h$ or some heuristic.
2) Vary the force applied by the robot F_r as a part of a higher level control strategy.
3) Amplify the force applied by the user by varying η in $F_r = \eta F_h$ (using Impedance Control). This is sometimes called Power Assist and is used specially for exoskeletons and assistive / rehabilitative robots.
4) Estimate the intended motion x_d of the human user if the robot is the follower (one of the objectives of this paper). This is known as motion estimation or human intention estimation. x_d can also be a reference trajectory from trajectory planners or for tracking tasks.
5) Use a reference $\dot{x}_d$ from models such as Minimum Jerk model or Minimum Torque Change model that are known to be representative of natural human limb motion.

B. Minimum Jerk Trajectory

In [6] it was shown that during natural point to point motion the human arm tends to minimize the cost function of mean-square jerk. The Minimum Jerk trajectory for a one degree of freedom motion can be expressed as

$$x(t) = x_0 + (x_f - x_0)(6\zeta^5 - 15\zeta^4 + 10\zeta^3) \tag{9}$$

where t_f is the duration of motion. It is assumed that the velocity is zero both at the beginning and the end of the movement ($\dot{x}_0 = \dot{x}_f = 0$) where x_0 is the initial hand position coordinates and x_f is the final position respectively, $\zeta = t/t_f$, $0 \leq \zeta \leq 1$. The velocity at each time instant corresponding to the MJM can be computed through numeric differentiation of the above equation.

The velocity given at each time instant by the MJM can be taken as a desired velocity for a robot-assisted Impedance/Admittance Control scheme. If the end effector follows the desired velocity during the task then it can provide a natural human-like feeling to the user while performing the task and ensure high transparency. The Minimum Jerk model is used in many works that have focused on robot assistance schemes such as in [14], [15].

C. Long Short-Term Memory Model

Recurrent Neural Networks are a powerful type of neural networks that are used for processing sequential data especially temporal data as they have an internal memory. As such they are useful for making predictions using time-series data. RNNs can be improved by using what are called LSTM units, whose structure maintains memory over time by utilizing multiple gates to control what information to keep and what to forget. Each LSTM unit has an input gate (i), output gate (o), and forget gate (f), each gate serving the role as their name suggests. It is known RNNs are unable to efficiently backpropagate error when the sequence becomes too long due to the problem of vanishing and exploding gradients. However LSTM units are effective in handling this issue due their ability to maintain a more constant error that can be backpropagated through time. As a result, RNNs with LSTM units are able to learn long-term dependencies within data sequences that were not possible only with RNNs. The LSTM formulation described in [25] is used here. Given an input sequence $\bar{x}_1, \bar{x}_2, ..., \bar{x}_t$ LSTM maps the input sequence to a sequence of hidden states $h_1, h_2, ..., h_t$ (which are also the outputs) by passing information through a combination of gates:

The Input gate (for updating the cell) is

$$i_t = \sigma_g(W_i \bar{x}_t + R_i h_{t-1} + b_i) \tag{10}$$

The Forget gate (for reseting the cell/forgetting) is

$$f_t = \sigma_g(W_f \bar{x}_t + R_f h_{t-1} + b_f) \tag{11}$$

The Cell candidate (for adding information to the cell) is

$$g_t = \sigma_c(W_g \bar{x}_t + R_g h_{t-1} + b_g) \tag{12}$$

where σ_c is the state activation function which here is $\sigma_c = tanh(\bar{x})$

The Output gate is

$$o_t = \sigma_g(W_o \bar{x}_t + R_o h_{t-1} + b_o) \tag{13}$$

σ_g is the gate activation function which here is the sigmoid function i.e $\sigma(x) = (1 + e^{-\bar{x}})^{-1}$

The Memory Cell state at timestep t is

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{14}$$

Here $\odot$ is the Hadamard product (element-wise multiplication of vectors). The memory cell selectively retains information from previous timesteps by controlling what to remember via the forget gate f_t.

The Hidden state (also called Output state) at time step t is

$$h_t = o_t \odot \sigma_c(c_t) \tag{15}$$

The Hidden State is passed as input to the next timestep. W_i, W_f, W_g are the learnable input weights, R_i, R_f, R_g are the learnable recurrent weights and b_i, b_f, b_g are the learnable bias.

The sigmoid function is a good activation function, and hence is used for the 3 gates - In, Out and Forget in the LSTM units as it outputs a value between 0 and 1. However for the memory cell, the values should be able to increase or decrease which is not possible with the sigmoid function as the output is always non-negative, hence we use the hyperbolic tangent function (tanh) as the activation function for the memory cell.

Using memory cells and hidden states LSTM units are able to retain information and stacking of numerous LSTM units is possible by using the h_t values at each layer as inputs to the next layer.

III. Experiments

Our experimental apparatus consisted of a KUKA LWR 4+ robot with 7 degrees of freedom, with a 6-axis ATI Gamma force-torque sensor mounted at the end-effector. We did not utilize the redundancy of the robot. The force-torque sensor provided a 6-dimensional wrench in the sensor frame at 1000 Hz. The joint encoders of the robot, through forward and differential kinematics computed the position and linear velocity of the end-effector of the robot at 500 Hz. The high level program was written in MATLAB and connected to the network via ROS (using the MATLAB Robotics Toolbox). A laser pointer was attached rigidly to a cylindrical tool mounted below the force-torque sensor. The laser pointer enabled the user to track the motion of the end-effector. Even though the sensors had different sampling rates we recorded the data at the same rate as we generated an individual dataset for each of the variables - position,velocity and force.

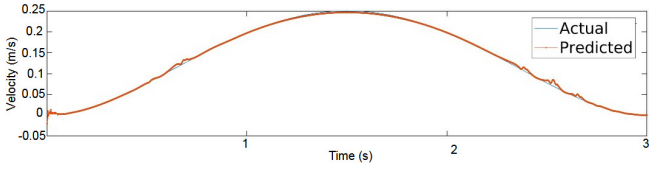


Fig. 4. Prediction of future velocity using the RNN-LSTM trained on the Minimum Jerk velocity dataset. The controller was Admittance Control. Here the RNN-LSTM is tested on a Minimum Jerk trajectory with $x_0 = t_0 = 0$, $x_f = 0.4$ m and $t_f = 3$ seconds. The RMSE was 0.003. These predicted values can be taken as the desired velocity $\dot{x}_d$ in the impedance model.

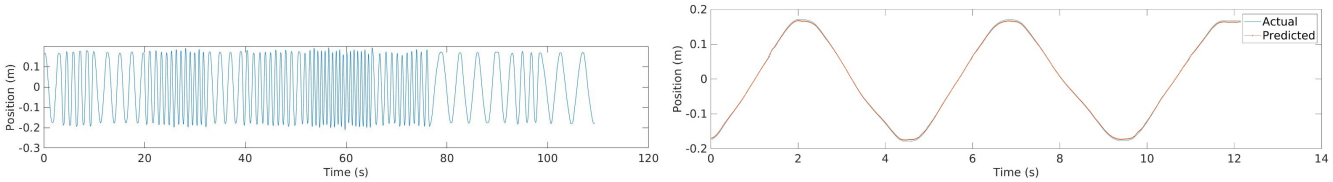


Fig. 5. The figure on the left shows the position values in the Natural Motion training dataset. The controller was Admittance Control. More than 30 cycles were recorded with varying speeds in each cycle. The figure on the right shows the prediction of future positions using the RNN-LSTM trained on the position values of the Natural Motion dataset. The RMSE was 0.002. These predicted values can be taken as the desired position x_d in the impedance model.

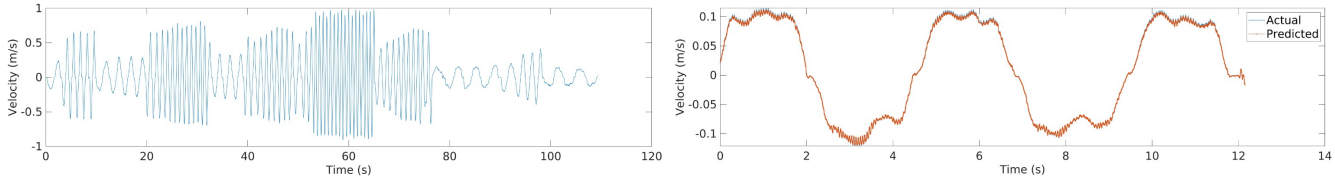


Fig. 6. The figure on the left shows the velocity values in the Natural Motion training dataset. The controller was Admittance Control. The dataset had a mixture of cycles with some cycles being fast movements while some were slow. The figure on the right shows the prediction of future velocities using the RNN-LSTM trained on the velocity values of the Natural Motion dataset. The RMSE was 0.0045. These predicted values can be taken as the desired velocity $\dot{x}_d$ in the impedance model.

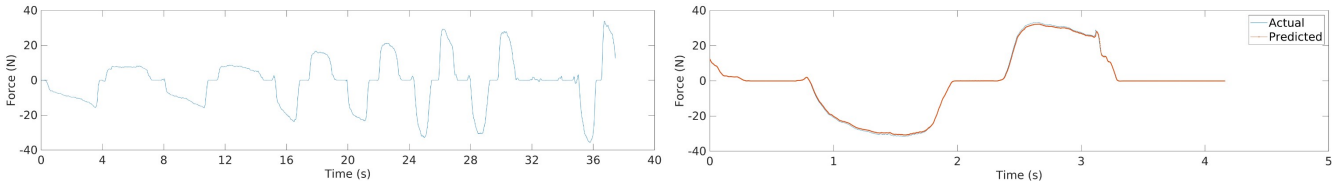


Fig. 7. The figure on the left shows the force values in the Natural Motion training dataset. The controller was Impedance Control. The figure on the right shows the prediction of future force values using the RNN-LSTM trained on the force values of the Natural Motion dataset. The RMSE was 0.3. These predicted values can be taken as the $F_r = F_d$ in (8).

For Impedance Control the controller used was the in-built Cartesian Impedance Controller with the following control law

$$\tau = J^T(K_d(x - x_d) + B_d(b_d)) + f_{dynamics}(q, \dot{q}, \ddot{q}) \quad (16)$$

where $q \in \Re^n$ is the joint position vector, K_d is the stiffness matrix in the end-effector frame,B_d is the damping matrix in the end-effector frame (b_d is the damping ratio), x and x_d have the same notation as before, in the global frame. The translational stiffness $K_x, K_y, K_z \in [0.01, 5000]$ N/m and rotational stiffness $K_{A_z}, K_{B_y}, K_{C_x} \in [0.01, 300]$ N/m-rad

For collecting the datasets we needed the laser pointer (i.e the tool) to be facing downwards vertically all the time, so we set the Rotation Matrix of the pose to be constant, the values of $K_{A_z}, K_{B_y}, K_{C_x}$ were set to 300 N/m-rad. The value of damping was set to b_d = 0.7 and $x_d = x$.

For Admittance Control we used the in-built Joint Position Controller. The desired virtual impedance Values were M_d=1 kg, B_d = 10 Ns/m and K_d = 10 N/m.

We generated multiple datasets for validating our RNN-LSTM architecture. We refer to the first dataset as the 'Natural Motion' dataset as it comprised of natural motions of the human. For the Natural Motion dataset we asked the user to move the tool continuously between two specified points along a single degree of freedom. The distance between the two points was 30 cms. The user was asked to move the tool at varying speeds in each cycle. The position and velocity of the end-effector was recorded from the robot controllers and the force applied was measured using the force-torque sensor. Two such 'Natural Motion' datasets were created, one using Impedance Control and the other with Admittance Control.

The last dataset was that of a large number of trajectories according to the MJM. For the 'Minimum Jerk' velocity dataset, we generated trajectories with $x_0 = t_0 = 0$ and varied the value of $x_f \in [0.1, 0.5]$ with step size of 0.1, and $t_f \in [1, 10]$ with a time-step of 0.1.

We took 90 percent of the complete datasets as the training dataset. The entire architecture consisted of 4 layers- an input layer, an LSTM layer, a fully connected layer, and a regression layer. The number of features was 1 as we had only a single variable for each dataset. Similarly as only 1 output was expected, the number of responses was 1. The number of hidden units was taken as 200 and the maximum number of epochs was set to 250.

The results of the predictions of the LSTM on the training set (10 percent of the complete datasets) is shown in Figures (4)-(7). The predictions are made for the immediate next time-step. The prediction accuracy is summarized in Table I. It is visible that the proposed methodology is accurate in

predicting the parameters. As a future work the immediate next step would be to study the increase in robot assistance using the proposed methodology. It could also be interesting to study the variation of robot impedance parameters with prediction of the human intention to facilitate cooperative pHRI tasks.

TABLE I

PREDICTION ACCURACY

Parameter	Root Mean Square Error
Velocity (m/s) (Minimum Jerk dataset)	0.003
Position (m)	0.002
Velocity (m/s) (Natural Motion dataset)	0.0045
Force (N)	0.3

IV. CONCLUSION

In this paper a new methodology has been proposed to predict the position, velocity and force applied by the human in a cooperative physical human-robot interaction task. The forecasted parameters can be integrated into the Impedance Control scheme and make the robot a proactive partner in sharing the physical load of the human. The proposed methodology uses a RNN-LSTM architecture and can accurately forecast parameters after being trained on a sample dataset. Accuracy of the predictions on the test dataset confirm the efficacy of the proposed method.

ACKNOWLEDGMENT

This work is partially supported by the Interreg Sudoe Programme and the European Regional Development Fund (ERDF) through the CoMManDIA project (SOE2/P1/F0638).

REFERENCES

[1] Dumora, Julie, et al. "Experimental study on haptic communication of a human in a shared human-robot collaborative task." Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on. IEEE, 2012.

[2] Donner, Philine, et al. "Human-robot cooperative object swinging." Robotics and Automation (ICRA), 2013 IEEE International Conference on. IEEE, 2013.

[3] Erden, M.S. and Mari, B., 2011. Assisting manual welding with robot. Robotics and Computer-Integrated Manufacturing, 27(4), pp.818-828.

[4] Wojtara, T., Uchihara, M., Murayama, H., Shimoda, S., Sakai, S., Fujimoto, H. and Kimura, H., 2009. Human robot collaboration in precise positioning of a three-dimensional object. Automatica, 45(2), pp.333- 342.

[5] Hogan, Neville. "Impedance control: An approach to manipulation." American Control Conference, 1984. IEEE, 1984.

[6] Flash, Tamar, and Neville Hogan. "The coordination of arm movements: an experimentally confirmed mathematical model." Journal of neuroscience 5.7 (1985): 1688-1703.

[7] Uno, Yoji, Mitsuo Kawato, and Rika Suzuki. "Formation and control of optimal trajectory in human multijoint arm movement." Biological cybernetics 61.2 (1989): 89-101.

[8] Fitts, Paul M. "The information capacity of the human motor system in controlling the amplitude of movement." Journal of experimental psychology 47.6 (1954): 381.

[9] Ikeura, Ryojun, and Kazuki Mizutani. "Control of robot cooperating with human motion." Proc. of IEEE International Workshop on Robotics and Human Communication. 1998.

[10] Rahman, M. M., R. Ikeura, and K. Mizutani. "Investigating the impedance characteristic of human arm for development of robots to co-operate with human operators." Systems, Man, and Cybernetics, 1999. IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on. Vol. 2. IEEE, 1999.

[11] Seki, Hirokazu, and Susumu Tadakuma. "Minimum jerk control of power assisting robot on human arm behavior characteristic." Systems, Man and Cybernetics, 2004 IEEE International Conference on. Vol. 1. IEEE, 2004.

[12] Ikeura, Ryojun, Tomoki Moriguchi, and Kazuki Mizutani. "Optimal variable impedance control for a robot and its application to lifting an object with a human." Robot and Human Interactive Communication, 2002. Proceedings. 11th IEEE International Workshop on. IEEE, 2002.

[13] Ohta, Ken, et al. "On the trajectory formation of the human arm constrained by the external environment." Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on. Vol. 2. IEEE, 2003.

[14] Dimeas, Fotios, and Nikos Aspragathos. "Fuzzy learning variable admittance control for human-robot cooperation." Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on. IEEE, 2014.

[15] Dimeas, Fotios, and Nikos Aspragathos. "Reinforcement learning of variable admittance control for human-robot co-manipulation." Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on. IEEE, 2015.

[16] Yamada, Yoji, et al. "Construction of a human/robot coexistence system based on a model of human will-intention and desire." Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on. Vol. 4. IEEE, 1999.

[17] Maeda, Yusuke, Takayuki Hara, and Tamio Arai. "Human-robot cooperative manipulation with motion estimation." Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on. Vol. 4. Ieee, 2001.

[18] Corteville, Brecht, et al. "Human-inspired robot assistant for fast point-to-point movements." Robotics and Automation, 2007 IEEE International Conference on. IEEE, 2007.

[19] Medina, Josmt al. "An experience-driven robotic assistant acquiring human knowledge to improve haptic cooperation." Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on. IEEE, 2011.

[20] Thobbi, Anand, Ye Gu, and Weihua Sheng. "Using human motion estimation for human-robot cooperative manipulation." Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on. IEEE, 2011.

[21] Ge, Shuzhi Sam, Yanan Li, and Hongsheng He. "Neural-network-based human intention estimation for physical human-robot interaction." Ubiquitous Robots and Ambient Intelligence (URAI), 2011 8th International Conference on. IEEE, 2011.

[22] Gribovskaya, Elena, Abderrahmane Kheddar, and Aude Billard. "Motion learning and adaptive impedance for robot control during physical interaction with humans." Robotics and Automation (ICRA), 2011 IEEE International Conference on. IEEE, 2011.

[23] Berger, Erik, et al. "Inferring guidance information in cooperative human-robot tasks." Humanoid Robots (Humanoids), 2013 13th IEEE-RAS International Conference on. IEEE, 2013.

[24] Li, Yanan, and Shuzhi Sam Ge. "Human?robot collaboration based on motion intention estimation." IEEE/ASME Transactions on Mechatronics 19.3 (2014): 1007-1014.

[25] Hochreiter, Sepp, and Jrgen Schmidhuber. "Long short-term memory." Neural computation 9.8 (1997): 1735-1780.

[26] Pfeiffer, Mark, et al. "A data-driven model for interaction-aware pedestrian motion prediction in object cluttered environments." 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018.

[27] Jain, Ashesh, et al. "Recurrent neural networks for driver activity anticipation via sensory-fusion architecture." Robotics and Automation (ICRA), 2016 IEEE International Conference on. IEEE, 2016.