A novel approach to compensate delay in communication by predicting teleoperator behaviour using deep learning and reinforcement learning to control telepresence robot

Fawad Naseer[1], Muhammad Nasir Khan[1], Akhtar Rasool[2], and Nafees Ayub[3]

[1]The University of Lahore
[2]Beijing Institute of Technology
[3]Government College University Faisalabad

January 2, 2023

Abstract

Robots with telepresence capabilities are typically employed for tasks where human presence is not feasible due to geography, safety risks like fire or radiation exposure, or other factors like any epidemic disease. Time delay is a significant consideration in controlling a telepresence robot. This study proposes a deep learning-based approach to compensate for the delay by predicting the behaviour of the teleoperator. We integrate a recurrent neural network (RNN) based on the Long Short-Term Memory (LSTM) architecture with the reinforcement learning-based Deep Deterministic Policy Gradient (DDPG) algorithm. The proposed method predicts the teleoperator's angular and linear controlling commands by using data gathered by embedded sensors on the specially designed and built telepresence robot. Simulations and experiments assess the operation of the proposed technique in Gazebo simulation and MATLAB with ROS integration, which shows 2.3% better response in the presence of static and dynamic obstacles.

A novel approach to compensate delay in communication by predicting teleoperator behaviour using deep learning and reinforcement learning to control telepresence robot

Fawad Naseer[1,], Muhammad Nasir Khan[1], Akhtar Rasool[2], and Nafees Ayub[3]

[1] *Electrical Engineering Department, The University of Lahore, Lahore, Pakistan*

[2] *Mechanical Engineering Department, Beijing Institute of Technology, Beijing, China*

[3] *Computer Science Department, Government College University Faisalabad, Faisalabad, Pakistan*

Email: fawadn.84@gmail.com

Robots with telepresence capabilities are typically employed for tasks where human presence is not feasible due to geography, safety risks like fire or radiation exposure, or other factors like any epidemic disease. Time delay is a significant consideration in controlling a telepresence robot. This study proposes a deep learning-based approach to compensate for the delay by predicting the behaviour of the teleoperator. We integrate a recurrent neural network (RNN) based on the Long Short-Term Memory (LSTM) architecture with the reinforcement learning-based Deep Deterministic Policy Gradient (DDPG) algorithm. The proposed method predicts the teleoperator's angular and linear controlling commands by using data gathered by embedded sensors on the specially designed and built telepresence robot. Simulations and experiments assess the operation of the proposed technique in Gazebo simulation and MATLAB with ROS integration, which shows 2.3% better response in the presence of static and dynamic obstacles.

Introduction: Healthcare paramedical staff spends significant time and effort in the sensitive and critically vital field of patient interaction. Robotics has been employed extensively in the healthcare environment for at last few years. The Puma 560 robot performed neurosurgical biopsies in 1985, making it the first robotic surgery [1-3]. The market for surgical robots has grown remarkably in recent years and is expected to reach $15 billion by 2030, with a compound annual growth rate of 17.1%.

ICT and robotic solutions have been developed over the past decade and are legitimate support that could enable persons to survive freely. Robots like Double robot from Double robotics and Nao from Softbank are a few examples. Increasing attention towards telepresence robots has been seen recently in creating for treating patients. Before testing the robot in a real-life environment, several researchers conduct brief robot assessments in a lab setting. However, other works use focus groups or laboratory experiments to analyze the system [4-5]. The telepresence robot generalized system is shown in Figure 1.

Fig 1: Generalized Telepresence Robot Framework

This study proposes a strategy for predicting teleoperator behaviour while controlling the telepresence robot by using a recurrent neural model built on the architecture of LSTM [6] and integrate it with the DDPG framework [7]. The goal is to create a single model that can handle all these distinct types of data from embedded sensors, whether they are raw data or not. Additionally, this model is used to demonstrate the significance of data considering the circumstances the telepresence robot will face. Thus, each entity should specify a control signal like angular and linear velocity.

Proposed Methodology: One of the deep learning approaches known as RNN automatically chooses the proper attributes from the practice cases. By storing a wealth of past data in its internal state, RNN is suitable for data processing and has exceptional potential in time-series forecasting. The basic configuration of an LSTM memory cell consisting of the long-term state component (C_t) and the short-term state component (h_t).

Input, forget, control, and output gates comprise LSTM's basic architecture. The input gate is what decides which data to transmit towards the cell and is described in Equation (1):

$$i_t = \sigma W_i \ \times \ h_{(t-1)} x(t) + \ b_i \ (1)$$

The bias vector and weight matrix are represented by b and W in the above equation. Tanh is applied to level the values in the range of [-1 to 1]. The proposed approach is to increase the cumulative future reward R_t, which is defined as Equation (2):

$$O_t = \ \sigma W_o \ \times \ h_{t-1} x\,(t) + b_o (2)$$

With gamma ranging from (0-1]. Under the state s_t and actiona_t, the estimation of R_t is defined as the value function in Equation (3):

$$R_t = \ r_t + \ \Upsilon . r_{t+1} + \ \Upsilon^2 . r_{t+2} + \ldots = \ \sum_{k=0}^{\infty} \Upsilon^k r_{t+k} (3)$$

To find the best action value $P^*(s_t, \ a_t)$, that is usually the major of all policies π. Afterwards, the optimal policy selects the action as Equation (4) to comprehensively train the optimal action value.

$P^{\pi}(s_t,\ a_t) = \ \mathbb{E}_{\pi}[R_t|s_t,\ a_t] = \ \mathbb{E}_{\pi}[\sum_{k=0}^{\infty} \Upsilon^k r_{t+k}|s_t,\ a_t](4)$

This study recommends deep reinforcement learning for teleoperator-controlling behaviour prediction (DRL). Modified LSTM is primarily used to predict the transmitted commands' linear and angular velocity and turning angle. The anticipated output of LSTM is then forwarded to the DDPG algorithm. The suggested workflow is shown in Figure 2.

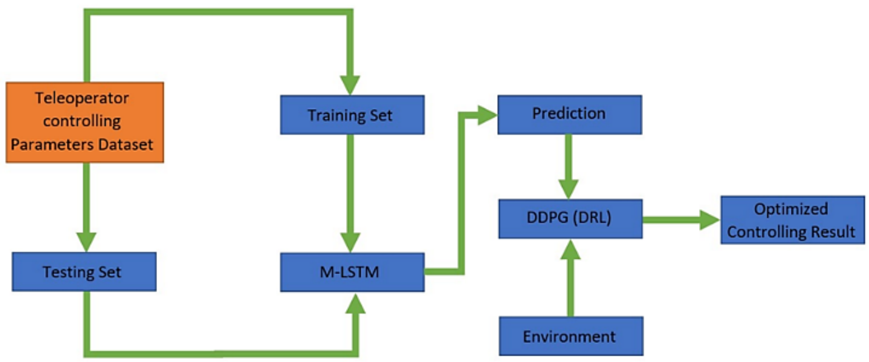


Fig 2 : Proposed Approach Framework to Control Telepresence Robot during Delay in Communication

The hybrid network divides into an online and targets part in the proposed approach structure comparable to DDPG alone. Although the two networks have the same topology, their update patterns are different. Figure 3 illustrates the hybrid network topology, where an LSTM component along with the decision component make up the main network. The main network's job is identifying the best policy for converting the state-to-action mapping.

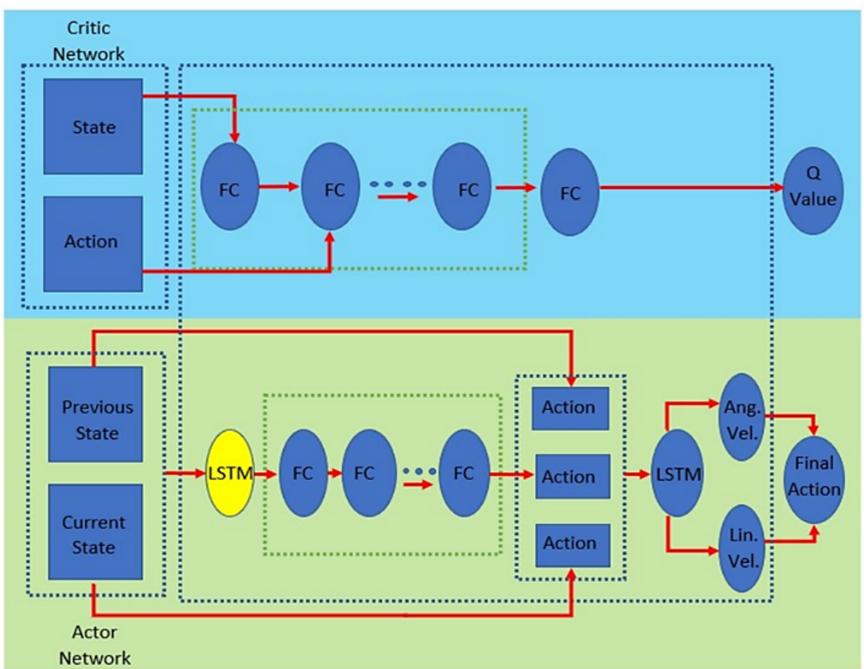


Fig 3: Proposed hybrid network framework

The algorithm for predicting teleoperator behaviour using LSTM and DDPG to control a telepresence robot during delayed communication is as follows:

ALGORITHM 1: A hybrid approach of integrating LSTM and DDPG to predict teleoperator behaviour

define and declare the prediction function of teleoperator behaviour

// preprocess data

processed_data = preprocess_data(data)

// train LSTM network

lstm_model = train_lstm(processed_data)

// make predictions using LSTM network

teleoperator_actions = lstm_model.predict(processed_data)

// train DDPG algorithm

ddpg_model = train_ddpg(processed_data)

// use DDPG algorithm to choose actions for telepresence robot

telepresenceRobot_actions= ddpg_model.choose_actions(teleoperator

// update LSTM and DDPG models with new data DDPG algorithm to choose actions for telepresence robot

lstm_model.update(new_data)

ddpg_model.update(new_data)

// use reinforcement learning to reward DDPG algorithm for positive outcomes

ddpg_model.reward(positive_outcomes)

// continuously update telepresence robot actions in real-time based on predicted teleoperator actions and current state

while True:

teleoperator _actions = lstm_model.predict(current_data)

telepresenceRobot_actions =ddpg_model.choose_actions(teleoperator

execute_actions(telepresenceRobot_actions)

end while

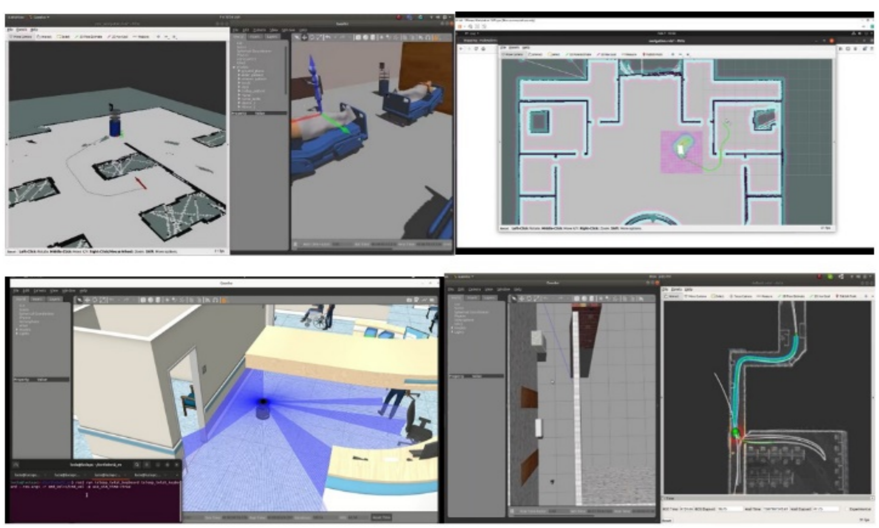
The above algorithm outlines the basic steps for predicting teleoperator behaviour using LSTM and DDPG to control an autonomous car. It includes preprocessing the data, training the LSTM and DDPG models, making predictions using the LSTM model, choosing actions for the telepresence robot using the DDPG model, continuously updating the models with new data, and using reinforcement learning to reward the DDPG model for positive outcomes. The telepresence robot's actions are continuously updated in a loop based on the predicted teleoperator actions and the current state of the robot.

Experiment Setup and Result Discussion: In this section, the experimental setup and its results are both explained with the proposed approach of controlling the telepresence robot during the delayed communication with the teleoperator.

The setup consists of a custom-manufactured telepresence robot and a remote-controlled setup of a teleoperator, as shown in Figure 4. The telepresence robot is powered by two DC-geared motors of 200 watts. The telepresence robot was equipped in the actual experiments with two Lidar sensors at various heights with a maximum measurement range of ten meters.

Fig 4: Custom built telepresence robot

The model is trained in four steps using our suggested multi-stage training mechanism-based technique. Using the Gazebo simulation framework using ROS integration for Lidar sensor configuration, we created four training scenarios, as shown in Figure 5. Each of the four simulation environments corresponds to one of the three training phases.



‘

Fig 5 : Simulation of Four different Environment using Gazebo and integration of ROS for Lidar sensor configuration

The proposed framework is tested using hardware-in-the-loop in the MATLAB and Simulink to organize and control the telepresence robot. During the simulation, a user uses the teleoperator controlling device connected to a ROS node and the ROS node updates at 1 kHz. The simulation runs at a sample rate of 150 Hz and in real-time. Each communication channel has time-varying delays, with$\mu_t = 0.5$ seconds and $\sigma_t^2 = 0.10$seconds implemented. In Fig. 9, the agents fluctuate slightly as they avoid the obstacles, which is a benefit of the proposed teleoperation approach.

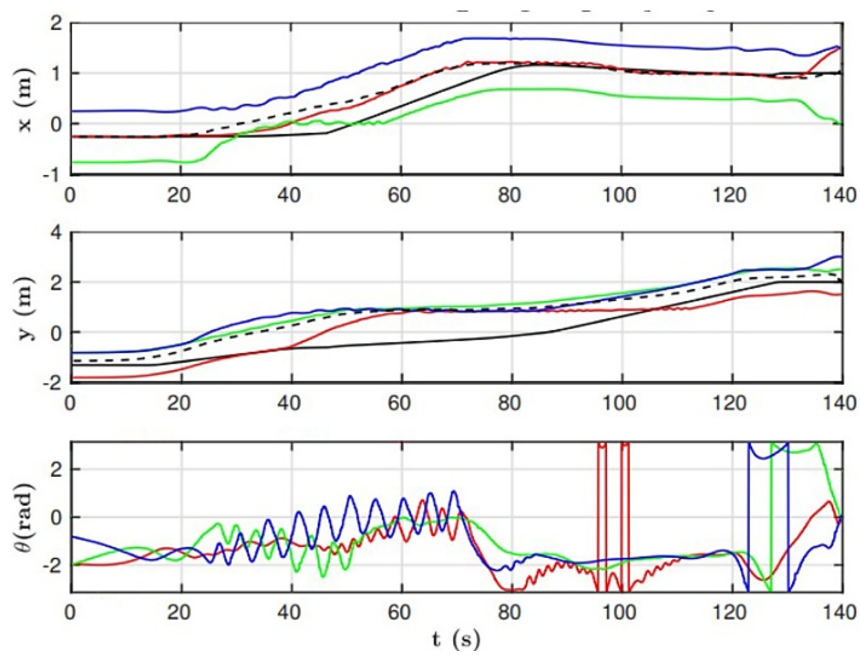
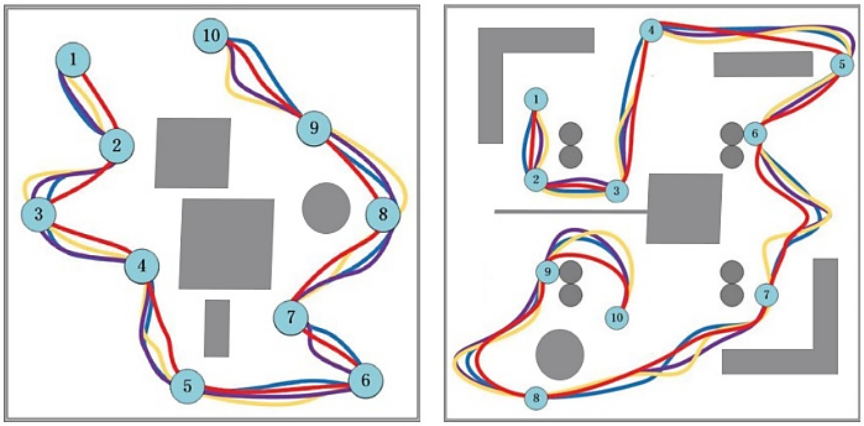


Fig 6 : Comparison of succession rate of different approaches

We have illustrated the pathways of the navigation challenges, as shown in Figure 7, to illustrate the benefits of our approach more clearly. The graph clearly shows that our proposed method produces softer paths while the paths produced by the comparative methods oscillate to be different degrees.

Conclusion: We proposed a novel approach to control the telepresence robot during delayed signals by integrating LSTM with the DDPG model. It utilizes supervised and reinforcement learning to combine the indication and assessment signals. The proposed hybrid technique uses RNN in addition to the off-policy actor-critic architecture to identify the best dynamic treatments. The comprehensive experiments on the real-world manufactured telepresence robot generate a dataset by multiple traversing of the same path in a healthcare environment. The proposed approach showed appreciative results in simulation experiments compared to other methods. After the data generation, our proposed approach was used and revealed that the suggested method could boost controllability by up to 2.3% and offer more control during the lack of communication or commanding signals.



(a) Env 2 (b) Env 4

Fig 7 : Simulation result of controlling and navigating in multiple different environments with obstacles.

Acknowledgments: We are thankful to University of Lahore for providing us the equipped laboratory and my supervisor to complete this research.

Conflict of interest: The authors declare no conflict of interest.

Data availability statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. Pedersen I, Reid S, Aspevig K. Developing social robots for aging populations: A literature review of recent academic sources. Sociology Compass. 2018 Apr 26;12(6):e12585.
2. Wang M, Pan C, Ray PK. Technology Entrepreneurship in Developing Countries: Role of Telepresence Robots in Healthcare. IEEE Engineering Management Review. 2021 Mar;49(1):20–6.
3. Wang J, Huang H, Li K, Li J. Towards the Unified Principles for Level 5 Autonomous Vehicles. Engineering. 2021 Jan;
4. Zhou Z, Zhu P, Zeng Z, Xiao J, Lu H, Zhou Z. Robot navigation in a crowd by integrating deep reinforcement learning and online planning. Applied Intelligence. 2022 Mar 17.
5. Zong C, Ji Z, Tian L, Zhang Y. Distributed Multi-Robot Formation Control Based on Bipartite Consensus With Time-Varying Delays. IEEE Access. 2019;7:144790–8.
6. Van Houdt G, Mosquera C, Nápoles G. A review on the long short-term memory model. Artificial Intelligence Review. 2020 May 13.
7. Xu J, Zhang H, Qiu J. A deep deterministic policy gradient algorithm based on averaged state-action estimation. Computers and Electrical Engineering. 2022 Jul;101:108015.