Transparency Enhancement in Teleoperation: An Improved Model-Free Predictor for Varying Network Delay in Telerobotic Application

Nithya Sridhar, Rolif Lima, Utsav Rai, Vismay Vakharia, Kaushik Das and Balamuralidhar P

***Abstract*— This work proposes a useful modification to an existing model-free predictor for handling network delays in teleoperation. This modification consists of an adaptively varying parameter that depends on delay at the current instant and the latest prediction error terms, which was not in use in the existing method. The conception of the proposed modification was obtained through theoretical analysis and later implemented for a mobile robotic platform. It was observed that the modified predictor adapted well to the varying delay and improved the system transparency by a maximum of 86% and by 15% in average. Additionally, this work also proposes a method to compensate for missing data using the same model-free predictor, showing a maximum transparency improvement by 12%.**

I. INTRODUCTION

In the recent past, unilateral teleoperation has been employed in special applications such as space exploration, underwater navigation, hazardous material handling, etc. However, most of them are not completely real-time and consist of a command-execute-feedback type of operation. With the advent of computational convenience and advances in virtual/augmented reality; wider network bandwidth and availability; bilateral teleoperation, involving real-time control has become a budding area in the present time. However, problem areas with respect to remote environment model, human operator model [1], network delay and packet loss are explored and researched extensively till date. Out of these challenges, the current work attempts to address the challenge faced in bilateral teleoperation due to network delay and missing data. Many studies and works on bilateral teleoperation consider stability as the topmost property to be conserved as delay in the network lead to oscillations and instabilities in a bilateral system [2]. This led to the implementation of many conservative approaches based on passivity and wave variable transformation in the beginning. These are very old techniques but can be seen to be improved and utilised even in the recent works [3], [4] [5]. Though these techniques have seen to preserve stability of the system, they have also been conservative in the sense of limiting the performance of the system. This drove the research community to move towards predictive approaches that can inherently preserve system stability and improve performance, provided the individual local and remote systems are stable. This introduced the concept of transparency in bilateral teleoperation defined as the ability of the system to perceive the remote environment with minimal lag/delay. Hence, these methods aimed at improving the transparency of the system. It is also imperative to mention the definition of synchronisation in the teleoperation domain, which is the ability of the robot to track the undelayed master command. Transparency was essentially considered more influential than synchronisation since the human operator was the sole decision maker and decrease in transparency could tamper with the system stability.

Nithya Sridhar, Rolif Lima, Utsav Rai, Vismay Vakharia, Kaushik Das and Balamuralidhar P are with the Embedded Systems and Robotics Reserach Group of Tata Consultancy Services, Bangalore-560066, India. {sridhar.nithya, rolif.lima, rai.utsav, vismay.vakharia, kaushik.da, balamurali.p}@tcs.com

A. *Predictive Approaches*

In teleoperation, without delay compensation, the human operator will perceive the past states of the robot operated in the remote environment. This may cause stability issues in the whole system. However, if the operator is able to perceive the undelayed states of the robot in the current time, 100% transparency is achieved and thus the operator's next course of actions will not cause instability issues due to delay. In the real world, 100% transparency is difficult to achieve. Hence, if the delayed states from the remote side is used to predict or estimate the current states using the model of the robot or through some other means, it would improve the transparency. By this, the operator is made to perceive the estimated states instead of the actual delayed states. This is also called as predictive display, which projects the states at current time using the received delayed states through the communication channel, thereby attempting to improve the transparency and decision making of the operator. This concept is illustrated in Fig. 1.

A survey on predictive control approaches was given by Uddin et al. [6] with larger emphasis to three predictive control approaches for teleoperation namely Smith Predictor (SP), Model-based Predictor(MBP) and Generalised Predictive Control (GPC), also known as model-mediated

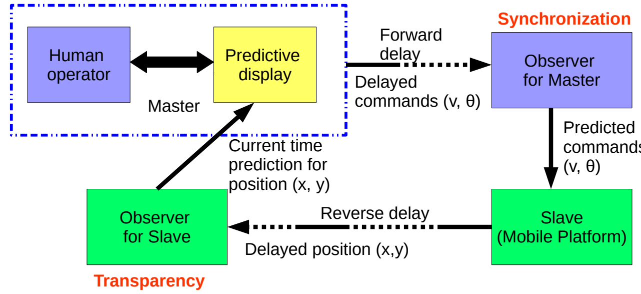

Fig. 1. System block diagram

control. Similarly, Shen et al. [7] proposed a novel cascaded observer for motion prediction of robot in the presence of system uncertainties. However, the authors assume prior model information of master and slave models.Though these techniques are effective, they required accurate model of the slave system, master system including the human operator dynamics and environment model, which can be difficult to capture in the real-world. Kebria et al. [8] used a filtering mechanism for prediction which consisted of 3 tuning parameters. Zheng et al. [9] proposed a model-free prediction approach for teleoperation of a ground vehicle. The advantage of using this approach was the presence of only one tuning parameter without the need for master or slave model. With a tiny refinement, this model-free predictor was extensively analysed and was employed in teleoperated ground vehicles in recent works [10], [11]. The current work aims to address some of the prevailing drawbacks of this model-free predictor and improve the prediction for better transparency.

II. Problem Statement and Contributions

The model-free predictor in the recent work [11] has its own advantages over model-mediated techniques for delay compensation. However, prior analysis of network delays and state errors due to the network, is required to select the value of the tuning parameter in the prediction equation. Hence, this study focuses on closing this gap by adapting to the instantaneous delay and prediction error terms, to facilitate real-time online prediction. The contributions of this work in the teleoperation domain are multi-fold,

- The study analyses a recent model-free predictor developed for teleoperation of vehicles and attempts to improve it, in terms of accuracy, online implementation and larger delay handling, by making the tuning parameter adaptive. This was achieved by using theoretical support and was implemented for a real network.
- In addition to delay handling, the study attempts to solve the issue of missing data due to various network issues in the communication channel.
- It attempts to show the significance of transparency in teleoperation and also provides a quantitative analysis for the same.

Section III presents the existing model-free predictor design, need for its improvement and proposes a modified predictor with potential reduction in prediction error. It also presents a way to handle missing data issue in teleoperation. Section IV discusses the details of the case study for which the proposed modification is implemented along with the means of quantifying transparency. Later, Section V quantifies and discusses the results of delay compensation and solution for missing data for various scenarios in terms of system transparency. Conclusion and future work is presented in section VI.

III. Predictor Design and Improvement

A model-free predictor, inspired from sliding mode control methodology, was developed by Tandon et al. [12] for hardware-in-loop testing of automobile engine, which later evolved and was expanded to teleoperation of ground vehicles in [9]. The predictor was given by,

$$\dot{y_p}(t) = \dot{y_r}(t - d(t)) - \mu(y_p(t - d(t)) - y_r(t - d(t)), \quad (1)$$

where, y_r and $\dot{y_r}$ are the delayed signals received from the remote location; $y_p(t - d(t))$ is the local state value in a previous instant corresponding to the observed delay, and $\dot{y_p}$ is the prediction made for the current time with μ as the only tuning parameter.

Though extensive frequency domain analysis and tuning guidelines were presented in connection with tuning μ, there was a considerable scope for improvement in the predictor performance for real-time implementation. This could be achieved if μ could adapt to the current time delay and the latest prediction error terms. This following sections shows the need for adapting μ and later proposes an adaptive law for the same.

A. Need for Adaptive μ

The original predictor [12] in (1) assumes its operating point on the sliding surface, implying that the sliding surface value is zero for the given μ value. However, with the inclusion of delay, the derivative of the prediction error becomes,

$$\dot{e}(t) = \dot{y_p}(t) - \dot{y_r}(t). \quad (2)$$

Substituting (1) in (2),

$$\dot{e}(t) = -\mu e(t - d(t)) + \dot{y_r}(t - d(t)) - \dot{y_r}(t). \quad (3)$$

Taking the error due to delay as,

$$e_{nw}(t) = -y_r(t) + y_r(t - d(t)), \quad (4)$$

(3) becomes,

$$\dot{e}(t) + \mu e(t - d(t)) = \dot{e}_{nw}(t). \quad (5)$$

Taking Laplace transform on both sides,

$$\frac{e(s)}{e_{nw}(s)} = \frac{s}{s + \mu e^{-ds}}, \quad (6)$$

where the error due to network delay (or the coupling error [10]) is the input to the system and the prediction error is the output of the system. The finite time convergence was proved [12] for constant delay and stability range was established [13] for varying delay.

In reality, the system will not be on the sliding surface due to a lot of factors. Hence, if a small value, ϵ is considered for the sliding surface, (5) would become,

$$\dot{e}(t) + \mu e(t - d(t)) - \epsilon(t - d(t)) = \dot{e}_{nw}(t). \quad (7)$$

Some prediction results with choice of different μ values in the stable range are shown in Figure 2. This shows the difference in predictions between the proposed method, μ_{max} for the maximum delay in the network and 50% of μ_{max}. With prior knowledge of the maximum delay in the network, the case of μ_{max} performed at par with adaptive case. Assuming no knowledge of the maximum

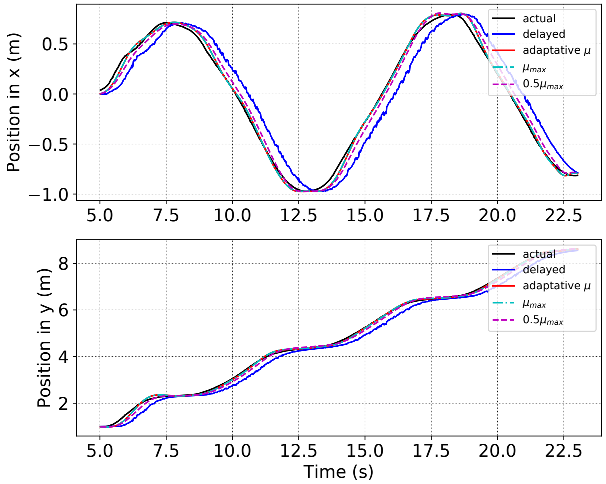

Fig. 2. Comparison of constant and adaptive μ performance

network delay (online prediction), a conservative value of $0.5\mu_{max}$ was chosen, which performed fairly well with scope for improvement. With the proposed technique, which was able to adapt to instantaneous delay and prediction error terms, there was a 17% reduction in prediction error compared to the above approach. This shows the adaptive techniques capability to be implemented for online prediction without any prior information or analysis of the network delay or the coupling error. Also, under/over assumption of maximum delay in the network would impact the predictor's performance badly if constant μ value had to be used for varying delay cases.

Hence, there is a definite need for the tunable parameter μ to adapt with respect to ϵ, besides, with the changing delay $d(t)$, for online implementation. The above equation (7) can thus be written as,

$$\dot{e}(t) + (\mu + \Delta\mu(\epsilon, t))e(t - d(t) = \dot{e}_{nw}(t), \tag{8}$$

where $\Delta\mu$ would attempt to compensate for the residue ϵ of the sliding surface in the past, corresponding to the delay.

Apart from the scope available for improvement in the predictor design, there were some special requirements pertaining to telerobotics area taken up in the current work, as compared to teleoperation of road vehicles carried out by Zheng et al. [9]. These requirements are listed,

(1) The accuracy expected in the current application is of centimeter level whereas the cited study showed accuracy improvement in the order of square meters and degrees.
(2) This application calls for online prediction of the delayed states, which does not have prior knowledge of maximum delay in the network and the error between the local and remote signals of the same state due to network delay. These information were used for tuning μ [11] in the previous work for offline prediction.
(3) A maximum delay of 600 ms was considered in the previous work, for testing, however this study aims to develop an improved predictor which can handle delays of up to 1 s.

In one respect, these mentioned requirements could also be viewed as the gaps present in the existing technique. The aim of the current work was to attempt to address these gaps.

B. Proposed Modification

Keeping in mind the role of instantaneous time delay on μ from (6), and the effect of sliding surface's past residue ϵ on $\Delta\mu$, the following adaptive law is proposed,

$$\mu_{adapt} = \mu_c(d) + \mu_f(\epsilon(t - d), \mu_{max}(d)), \tag{9}$$

where $\mu_c(d)$ is the course tuned component of the parameter with respect to the instantaneous delay, $d(t)$, and μ_f is the fine tuned component that is a function of ϵ and the maximum value of μ from Lambert W function [14]. Both these components are given by,

$$\mu_c = \frac{2}{3}\mu_{max}(d), \tag{10a}$$

$$\dot{\mu}_f = -\frac{\epsilon(t - d)\mu_{max}(d)}{K}, \tag{10b}$$

where $\mu_{max}(d)$ is the maximum value of μ corresponding to the current time delay for which the system is stable, obtained from Lambert W function [14] and from (6). K is a tuning variable that also determines the step change in μ_f for a particular sample time. Here, $\mu_{max}(d)$ is also considered in the equation to make the step change proportional to the current delay's maximum allowable μ value for stable operation. Also, from (7) and (8), it is seen that increase in ϵ will reduce the μ_f value to compensate for the prediction error and hence the negative sign in (10b).

The minimum and maximum values of μ are restricted to $\frac{1}{3}\mu_{max}(d)$ and $\mu_{max}(d)$ respectively to enable stable operation in this study. Hence, μ_f is given a variation limit of $\frac{1}{3}\mu_{max}(d)$ or approximately 33.3% of $\mu_{max}(d)$ on either side to adjust for the past ϵ value in the delay span. These specific limits were chosen with the insights provided by the studies [9], [10] and their tuning guidelines [11], [13].

C. Missing Data Issue

During network controlled operations, it is common to have missing data issues due to packet loss or depending on network protocols used. While the previous work [11] proposed to compensate for delay with the predictor, it did not attempt to compensate for missing data in a particular instant. With this in mind, the above predictor is modified slightly to compensate for missing data and is given by,

$$\dot{y_p}(t) = \dot{y_r}(t-d-m) - \mu(y_p(t-d) - y_r(t-d-m)), \tag{11}$$

where m represents the instances of data loss from the current time, which makes $\dot{y_r}(t-d-m)$ and $y_r(t-d-m)$ as the only set of usable states for prediction with respect to $y_p(t-d)$ with delay, $d(t-m)$ representing the delay corresponding to the latest available states. This simple law would be tested for its efficacy in section V.

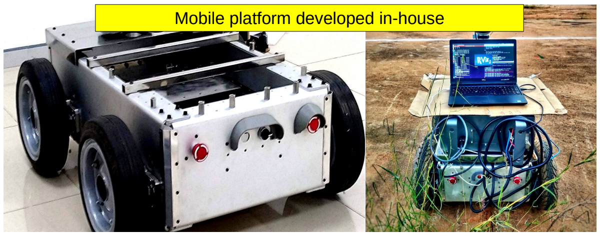

Fig. 3. Mobile platform

IV. Predictor Implementation on Mobile Robot

A. Master Slave System

Though the developed predictor could be used for any tele-system covered under its bandwidth [11], it is implemented in a mobile robot moving in a constrained space. The system used for testing the efficacy of the predictor is a mobile platform shown in Figure 3 that is developed for a humanoid robot, operated by a human from a remote location. The human operator relays two commands to move the platform, namely the longitudinal speed and the yaw angle of the platform. Since, it moves in an indoor/constrained environment with many stationary objects and moving humans in the vicinity, the maximum velocity is limited to 1 m/s, however for testing purpose, it is 1.5 m/s. A humanoid robot will later be integrated with the mobile platform and the predictor will be used for estimating the undelayed positions of various joints and elements of the robot. This kind of application demands accuracy in the order of a few centimeters in the predictor for efficient real-time control.

B. Transparency Metric

Transparency is defined as the ability of a human operator to perceive (see, feel and smell) the remote environment with a virtual absence of delay and missing data in the communication network. Transparency is a very important parameter not only with respect to system stability but also with respect to operator friendliness in a real-time teleoperation. This property can be quantified as,

$$T_r = \frac{1}{n}\sum_{i=1}^{n}\left(1 - \frac{|m_{di} - m_{ai}|}{|m_{ai}|}\right)100, \qquad (12)$$

where m_{di} is the delayed/ predicted value and m_{ai} is the undelayed value at the i^{th} instant, and n is the total number of data points. The transparency T_r will be 100% when the error between actual and the delayed signals are zero, which is never true in a network control system due to the coupling error. However, this metric would be used to analyse the transparency with and without the proposed predictor.

V. Results

For validating the proposed scheme, simulations were run initially, with the robotic system in Gazebo along with artificial delay injection. This data was utilised to choose an appropriate value of K in (5). Depending on the sample time used for simulation and with systematic trial and error

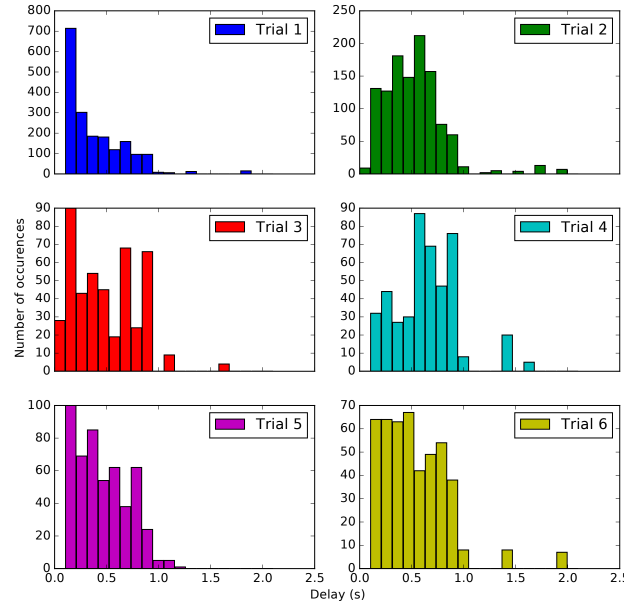

Fig. 4. Delays for different scenarios

approach, K was chosen to be 1 for the sample time of 20 ms. It is important to note that, this prior simulation exercise is the only requirement to tune K. Once a suitable value is chosen, there is no need for real network data analysis in the later stage. The algorithm with its adaptive law, is ready for implementation in the physical network.

A. Implementation on the Physical Network

The experiments on the physical network was carried out using Amazon Web Services (AWS) to establish a connection link between the master in Ayodhya, Uttar Pradesh, India and the slave in Bangalore, Karnataka, India with a physical distance of about 1900 km. Before implementation, the systems on both the locations were time synchronized for forward and return delay calculations. There were six scenarios/trials with different maneuvers implemented, for which the return delay pattern is shown in Fig. 4. The slave system was a mobile platform in the Gazebo environment and the human operator in the master location controlled the slave system by giving velocity and yaw commands through keyboard interface. The video streaming of slave environment was carried out in the master location. The control architecture was built using Robotic Operating System (ROS) platform. A snapshot taken from the master's system is shown in Figure 5.

B. Prediction with time delay

The proposed predictor in (1) was tested for its efficacy for 6 scenarios and a comparison of the results with and without the use of predictor is presented in table I. These two cases were compared in terms of transparency given in (12).

From table I, it is seen that the mean transparency improvement with the usage of the proposed predictor computes to 15% and the maximum improvement is shown at 86%, thereby, ascertaining the predictor's efficacy. A visual

TABLE I
TRANSPARENCY IMPROVEMENT WITH PREDICTION FOR TIME DELAY

Trail No.	Transparency (x,y), % With prediction	Transparency (x,y), % Without prediction	T_r improvement (x,y), %
1	(94,99)	(79,91)	(19,9)
2	(94,99)	(83,98)	(13,1)
3	(95,98)	(51,95)	(86,3)
4	(99,95)	(89,89)	(11,7)
5	(86,95)	(82,86)	(5,9)
6	(97,98)	(82,91)	(23,7)

representation of its performance is shown in Figure 6 for one of the scenarios for both x and y position of the mobile platform. This trial consisted of delays as high as 1 s. This figure visually shows the accuracy of its prediction with respect to the black line, which is the actual undelayed signal. Hence, the proposed method performs well for the specified application. A magnified plot for another trial is shown in Fig. 7 to present the comparison between the distorted delayed signal and the predicted signal that is close to the actual undelayed state. This visually shows the transparency improvement with the use of the modified model-free predictor.

C. Prediction with time delay and missing data

The proposed predictor in (11) was tested for its efficacy for two scenarios with different data transmission probabilities. For example, a 80% data probability would mean 20% loss in data or 20% of missing data. A comparison of the results with and without the use of data compensation is presented in table II. In testing, when there was a data loss in a particular instant, the previous state values were considered for that instant to make the delayed signal. When data compensation was not in use, the prediction, $\dot{y}_p$ in the previous instant was considered for extrapolation during a missing data case.

The two cases are compared in terms of transparency given in (12). Further, Figures 8 and 9 show the performance comparison of the data loss compensation against its absence.

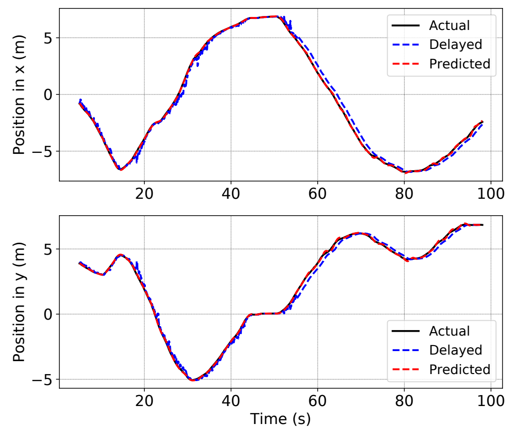

Fig. 6. With the predictor for delay compensation

Due to missing data, the performance of the prediction without compensation was oscillatory, however this issue was mitigated with compensation. This is clearly visible in the zoomed plot in Fig. 9. Also, transparency improvement is higher about 4.6% on an average with compensation, which shows its efficacy. It is also interesting to note that there is a decrease in transparency for one of the cases in trial 2 with the usage of predictor and without data loss compensation.

TABLE II
TRANSPARENCY IMPROVEMENT WITH PREDICTION FOR TIME DELAY, WITH AND WITHOUT MISSING DATA COMPENSATION

Trial No.	Data Probability	T_r improvement (x,y), % With comp.	T_r improvement (x,y), % Without comp.
1	0.8	(19.5,8.0)	(7.3,7.7)
1	0.6	(17.8,9.1)	(17.1,3.0)
1	0.4	(20.0,12.0)	(18.2,8.1)
2	0.8	(7.9,10.5)	(5.0,9.8)
2	0.6	(10.1,11.2)	(4.6,10.2)
2	0.4	(10.1,11.0)	(-13.0,10.0)

Fig. 5. Teleoperation implementation

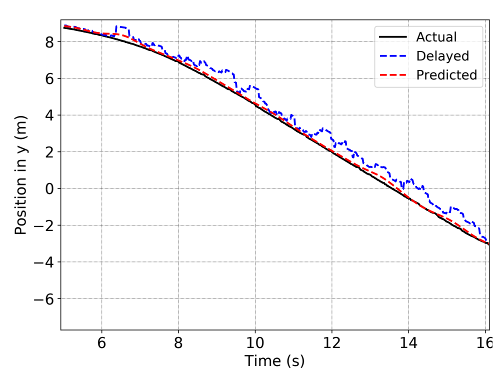

Fig. 7. Zoomed plot: Distorted delay signal vs predictor output

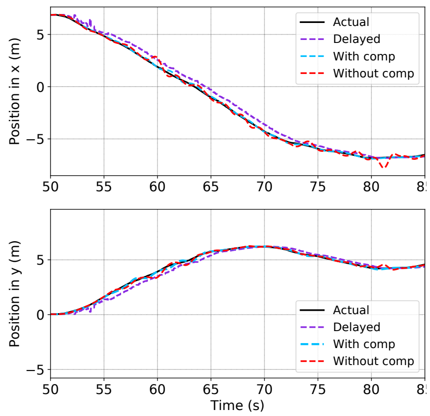

Fig. 8. With and without data loss compensation

VI. Conclusion and Future Work

A useful model-free predictor proposed in an existing work for handling network delays, was modified in this work in order to avoid prior data analysis and enable online implementation of the predictor. The modification consisted of an adapting predictor parameter, μ that was tuned from data analysis in the previous work. Need for this adaptive technique was explicated and the proposed technique was implemented for real network data. It was shown to improve the transparency by 15% on average with a maximum improvement of 86% in the evaluated cases.

In addition, this work proposed a simple technique attempting to solve the missing data issue, which was absent in the previous work. This was also tested in the given robotic application with physical network delay (varying). This led to an additional transparency improvement of 4.6% on average with a maximum of 12%.

This work assumed data and video streaming to be synchronised and both the delays to be the same at a particular instant. However, in real time, there may be little/less synchronisation between the two, which will be considered in future work. Further, it is planned to extend this work for a humanoid robot. Though, this work was implemented in the real network, the robotic system was simulated in Gazebo environment, hence, it is only fruitful to deploy it for the real robotic system in the near future.

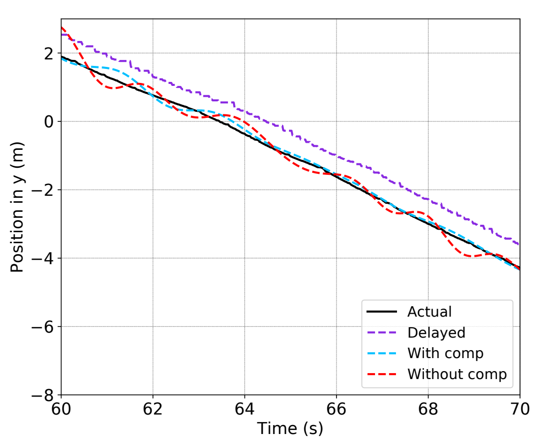

Fig. 9. Missing data compensation: Zoomed in plot

References

[1] M. Esfandiari, K. Nazari, and F. Farahmand, "Robust adaptive bilateral control of teleoperation systems with uncertain parameters and unmodeled dynamics," in *2017 5th RSI International Conference on Robotics and Mechatronics (ICRoM)*. IEEE, 2017, pp. 95–100.

[2] G. Niemeyer and J.-J. Slotine, "Stable adaptive teleoperation," *IEEE Journal of oceanic engineering*, vol. 16, no. 1, pp. 152–162, 1991.

[3] C. Yang, X. Wang, Z. Li, Y. Li, and C.-Y. Su, "Teleoperation control based on combination of wave variable and neural networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 8, pp. 2125–2136, 2016.

[4] P. M. Kebria, A. Khosravi, S. Nahavandi, Z. Najdovski, and S. J. Hilton, "Neural network adaptive control of teleoperation systems with uncertainties and time-varying delay," in *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2018, pp. 252–257.

[5] J. Guo, C. Liu, and P. Poignet, "A scaled bilateral teleoperation system for robotic-assisted surgery with time delay," *Journal of Intelligent & Robotic Systems*, vol. 95, no. 1, pp. 165–192, 2019.

[6] R. Uddin and J. Ryu, "Predictive control approaches for bilateral teleoperation," *Annual Reviews in Control*, vol. 42, pp. 82–99, 2016.

[7] S. Shen, A. Song, T. Li, and H. Li, "Time delay compensation for nonlinear bilateral teleoperation: A motion prediction approach," *Transactions of the Institute of Measurement and Control*, vol. 41, no. 16, pp. 4488–4498, 2019.

[8] P. M. Kebria, A. Khosravi, S. Nahavandi, P. Shi, and R. Alizadehsani, "Robust adaptive control scheme for teleoperation systems with delay and uncertainties," *IEEE transactions on cybernetics*, 2019.

[9] Y. Zheng, M. J. Brudnak, P. Jayakumar, J. L. Stein, and T. Ersal, "An experimental evaluation of a model-free predictor framework in teleoperated vehicles," *IFAC-PapersOnLine*, vol. 49, no. 10, pp. 157–164, 2016.

[10] X. Ge, Y. Zheng, M. J. Brudnak, P. Jayakumar, J. L. Stein, and T. Ersal, "Performance analysis of a model-free predictor for delay compensation in networked systems," *IFAC-PapersOnLine*, vol. 48, no. 12, pp. 434–439, 2015.

[11] Y. Zheng, M. J. Brudnak, P. Jayakumar, J. L. Stein, and T. Ersal, "A delay compensation framework for predicting heading in teleoperated ground vehicles," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 5, pp. 2365–2376, 2019.

[12] A. Tandon, M. J. Brudnak, J. L. Stein, and T. Ersal, "An observer based framework to improve fidelity in internet-distributed hardware-in-the-loop simulations," in *Dynamic Systems and Control Conference*, vol. 56130. American Society of Mechanical Engineers, 2013, p. V002T21A004.

[13] Y. Zheng, M. J. Brudnak, P. Jayakumar, J. L. Stein, and T. Ersal, "A predictor-based framework for delay compensation in networked closed-loop systems," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 5, pp. 2482–2493, 2018.

[14] F. M. Asl and A. G. Ulsoy, "Analysis of a system of linear delay differential equations," *J. Dyn. Sys., Meas., Control*, vol. 125, no. 2, pp. 215–223, 2003.