



Drive-by-Wireless Teleoperation with Network QoS Adaptation

Zeashan Hameed Khan, Jean-Marc Thiriet, Denis Genon-Catalot

► To cite this version:

Zeashan Hameed Khan, Jean-Marc Thiriet, Denis Genon-Catalot. Drive-by-Wireless Teleoperation with Network QoS Adaptation. International Journal of Engineering Science and Technology, 2011, 2 (2), pp.160-169. hal-00560419

HAL Id: hal-00560419

<https://hal.science/hal-00560419>

Submitted on 28 Jan 2011

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.

Drive-by-Wireless Teleoperation with Network QoS Adaptation

Zeashan H. Khan¹, Jean Marc Thiriet²

Control Systems Department,
GIPSA-lab, France

engrzee@gmail.com¹, jean-marc.thiriet@ujf-grenoble.fr²

Denis Genon-Catalot

LCIS, Valence

Denis.genon-catalot@iut-valence.fr

Abstract—This paper describes an adaptation scheme for drive-by-wireless teleoperation of an electric vehicle. The performance variation due to the varying time delay and packet losses is investigated and catered for by using an adaptive gain scheduling as well as varying network load in the co-design framework. A support vector machine (SVM) predictor is proposed for network delay and packet loss prediction while, fuzzy based estimation and control is employed for QoS evaluation and corresponding reconfiguration for video packet rate and controller gain in order to ensure passivity and improved transparency.

Keywords—Networked teleoperation; co-design problem; drive-by-wireless; supervised learning; QoS estimation

I. INTRODUCTION

In drive-by-wireless teleoperation, a remote vehicle is controlled with a wireless connection as opposed to traditional electrical, hydraulic and mechanical methods as presented in [1]. The control and video information is exchanged between the driver (master) and the vehicle (slave) bilaterally during teleoperation. Long distance teleoperation employs hybrid/heterogeneous communication network to transport command and feedback data between the operator and the slave system as direct control is not possible in scenarios where the mobile teleoperator (slave) is located in hard to reach or dangerous remote areas.

In addition to mobility, a wireless network eliminates hard connections between one or more control and sensor modules which results in simple design and lowered installation and maintenance costs. Moreover, for multi-vehicle case, a wireless network is more effective than point-to-point communication links as proposed in [2]. However, as per [3], wireless networks induce additional delays and information loss as they could be easily perturbed by the environmental effects and interference due to multipath effects or due to other communicating entities. [4] and [5] emphasize that the drive-by-wireless systems require multi-domain knowledge including communication networks, vehicle dynamics and bilateral teleoperation with force feedback for better haptic sensation.

In the literature, generally, bilateral teleoperation utilizes master-slave pair which communicates over a communication network. Despite its interesting features, bilateral teleoperation has some limitations and performance dependencies over several factors. The number of tasks they can perform as compared to human are also limited, since the dexterity of teleoperator is poorer

than the human dexterity as noted in [6]. This even worsens and sometimes destabilize with the added time delays. In addition to stability, bilateral teleoperator are supposed to provide sufficient transparency despite the presence of mechanical nonlinearities and communication imperfections as discussed in [7]. We consider only passivity based architecture for bilateral teleoperation of drive-by-wireless applications, which utilize energy concepts and impose the passivity requirement for each individual sub-system. However, this approach is too conservative, in the sense that it does not guarantee good performance under all operating conditions as argued in [8]. This justifies the intelligent adaptation of control gains with the operating conditions to maximize the performance and satisfy all control objectives.

In modeling a communication channel only time delay is considered during analysis and full knowledge of communication protocol with its flexibilities, limitations and QoS aspects are rarely addressed by the control engineers. However, it is interesting to investigate that even if the wave variables are exchanged between master and slave, giving a notion of importance to this information or using some network based tactics can improve the quality of tracking and performance in bilateral teleoperation for drive-by-wireless application. This requires a co-design approach applied to robotic systems as shown in [9] and [10].

II. CONTROLLER DESIGN FOR BILATERAL TELEOPERATION

The classical architecture for bilateral teleoperation, proposed by [11] was based on the scattering and passivity theory, which assures robustness against the network delays. This architecture neither guarantees the position tracking in stationary conditions nor force detection during the functioning of the system. In [12], the authors expanded the Anderson's work to guarantee stability under time varying delays but it produces significant transient oscillations.

In [13], the traditional configuration based on passivity is utilized and a position control loop is added on the master/slave side to track the position and force as shown in Fig. 1. In [14], a detail survey of techniques in bilateral teleoperation is presented. The single degree of freedom (DOF) master/slave dynamics with position control loop in standard notation are given as:

$$M_m \ddot{x}_m + B_m \dot{x}_m = F_h + F_m \quad (1)$$

$$M_s \ddot{x}_s + B_{s1} \dot{x}_s = F_s - F_e \quad (2)$$

where x_m is the velocity of the master, F_h and F_m constitute the control couple applied to the motors at the master/slave, M_m , M_s are the inertias, B_m , B_{sl} are the viscous frictions of master and slave, F_h , F_e are the reaction couple from the operator and the environment and x_m , x_s are the respective positions. $F_{feed} = K(x_m(t-T)-x_s)$ and $F_{back} = K(x_s(t-T)-x_m)$ are the position controllers for the slave and master loop respectively. We utilize the scattering transformation to assure the passivity of the system in the presence of constant time delays so that the characteristics that describe the channel are similar to those of a transmission line without losses. The transformation used in Anderson and Spong (1989) is as under:

$$U_m = \frac{1}{\sqrt{2b}}(F_m + b\dot{x}_m), \quad V_m = \frac{1}{\sqrt{2b}}(F_m - b\dot{x}_m) \quad (3)$$

$$U_s = \frac{1}{\sqrt{2b}}(F_s + b\dot{x}_{sd}), \quad V_s = \frac{1}{\sqrt{2b}}(F_s - b\dot{x}_{sd}) \quad (4)$$

The scattering variables (u_m, u_s, v_m, v_s) are transmitted across the delay line instead of the original velocities and forces. The transient error is dependent on delay while in the steady state position tracking $e(t) = x_m(0) - x_s(0)$ is dependent on the initial position difference even when there is no packet loss. However, with packet losses, the performance of the control loop deteriorates even more. The position tracking error is defined as $e = x_m(t-T) - x_s(t)$, where $x_m(t-T)$ is the delayed master position received on the slave side. For the stability analysis, it is assumed that the human operator and the environment model are passive systems, bounded by known functions of the master and slave velocities. All signals are assumed to belong to the extended l_{2e} space and $x_m, x_s = 0$ for $t < 0$.

III. STABILITY AND PERFORMANCE IN PASSIVE BILATERAL TELEOPERATION

The stability of the position controller is proposed with a Lyapunov function which puts the condition proven in Chopra et al. (2004) that:

$$K_{pass} < \sqrt{\frac{b_m \cdot b_{sl}}{\tau^2}} \quad (5)$$

Where K_{pass} is the gain limit which should not be exceeded in order to respect the passivity.

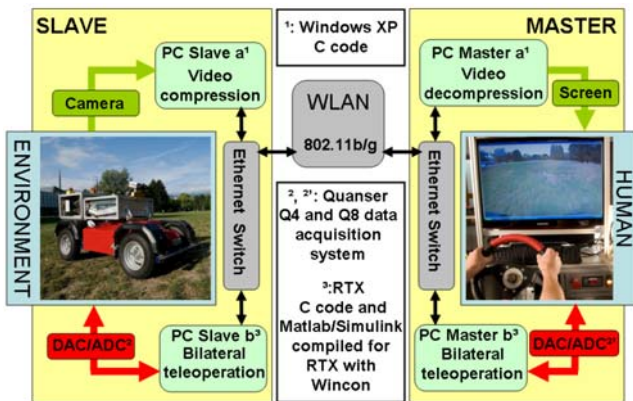


Figure 1. NeCS-Car Embedded System and Communication Architecture.

Using Barbalat's lemma it was shown that the tracking error can be rewritten in (6) as found in [5]:

$$e = x_m(t) - x_s(t) - \int_{t-T}^t \dot{x}_m(\tau) d\tau \quad (6)$$

It is important to note that the stability of this architecture is proven by taking into account a fixed delay assumption. However, the simulation results prove that the system is stable under limited packet losses, whereas the tracking performance is degraded with increasing packet losses. The transparency objective performance metric requires that the impedance perceived by the human and the environment impedance must be equal as shown in [19], i.e. $Z_h = Z$. In addition, the position and force at the master interface and the slave are ideally required to be equal. However, it depends on the similarity between master and slave devices, gain adjustments and delays introduced by the network. The stiffness decreases with the increased delay and packet loss rate. The mechanical impedance Z_t is the transmitted impedance of the slave as seen by the human operator i.e. Z . In most cases this impedance can be sufficiently well approximated by an LTI system and hence can be formulated as the transfer function shown above. Thus, with increased delay and packet losses, environment impedance Z_e decreases and affects the transparency.

IV. NECS-CAR EXPERIMENTAL SETUP

The NeCS-Car is a dedicated platform for teleoperation funded by the Networked Control System (NeCS) team at the Control System department of GIPSA-lab. A remote operator can drive the car via a hybrid (Ethernet + WLAN) networked communication by observing the video and force feedback.

A. Hardware

The overall system architecture is shown in Fig. 1. The embedded PC hosts 2 operating systems, each having their own network card and can be considered as 2 separate PCs for video and control system. For communication between controller, image processor and IP video cameras, Ethernet is used as the embedded network which is connected with a 100 Mbps switch linking the Slave with the Master via WLAN router.

B. Communication

Some time critical data is exchanged between master and slave because this 200 Kg vehicle can be driven up to a maximum speed of 10 m/s on uneven ground. The control data (speed, position, brake, etc) is sent over UDP. The packet size for control information is 83 bytes (125 bytes with header) of data sent with a sampling rate of 1000 Hz (1 Mbps). About 20% of the traffic sent over UDP is lost. For video data, both TCP/IP or UDP/IP protocol and different compression algorithms can be tested. IP camera with MPEG-4 compression can also be connected to the slave Ethernet switch to generate traffic perturbations (data rate of 3.1 Mbps over TCP/IP measured while driving) over the network. For control data reconstruction due to packet loss, our approach is similar to [18] where scattering transform energy is exchanged to continuously monitor the passivity condition in order to choose between the hold last value (HLS) and zeroing. In this way, passivity of the communication block is conserved. NeCS-Car

parameters used are as follows: $M_m = 0.0284 \text{ N.m}^2/\text{rd}$, $M_{s1} = 3.25 \text{ N.m}^2/\text{rd}$, $B_m = 0.0817 \text{ N.m.s/rd}$, $B_{s1} = 5.6833 \text{ N.m.s/rd}$. The sampling time $T_s = 0.001 \text{ s}$.

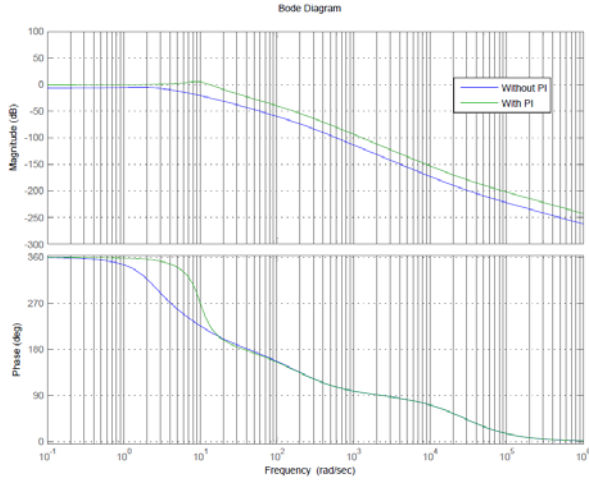


Figure 2. Bode plot with and without PI Controller.

A PI controller is used to improve the reversibility with $K_p = 10$ and $T_i = 0.2$. The improvement in response is witnessed by the bode plot where larger bandwidth is available due to higher cutoff frequency with PI controller as shown in Fig. 2.

V. ANALYSIS OF TRANSPARENCE IN DRIVE BY WIRELESS APPLICATIONS

Transparency refers to the good feeling of the task felt by the operator. Ideally, the operator feeling to sense the ground and environment should be the same as if he/she is directly interacting at the teleoperator site. This concept is difficult to model and defined as a mathematical formulation for quantitative performance measure. However, there are several approaches in the literature today e.g. in [29], [30] and [31]. The classical approach takes into account the impedance matching between the master and the slave [31]. In [32], an extended transparency condition is defined based on trackability and immersivity.

In most cases, the H-matrix is used to describe the transparency which relates the output variables slave velocity (V_s) and reflected force (F_h) with the input variables reflected force (F_e) and master velocity (V_m), as:

$$H = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \quad (7)$$

Where, $h_{11} = \left. \frac{F_h}{V_m} \right|_{F_e=0}$ is the master impedance Z_m , $h_{21} = \left. \frac{V_s}{F_e} \right|_{V_m=0}$ is the velocity gain (G_v) from master to slave, $h_{12} = \left. \frac{F_h}{F_e} \right|_{V_m=0}$ is the force Gain (G_F) and $h_{22} = \left. \frac{V_s}{F_e} \right|_{V_m=0}$ is the slave admittance (Y_s).

For optimal transparency, $G_F = G_v = 1$ which result in $V_m = V_s$ and $F_h = F_e$. Also, it is required that $Z_m = 0$ and finally the slave should be unaffected by the external forces ($Y_s = 0$). However, these ideal requirements are rarely satisfied due to the sensor and actuator non-linearities and hardware limitations.

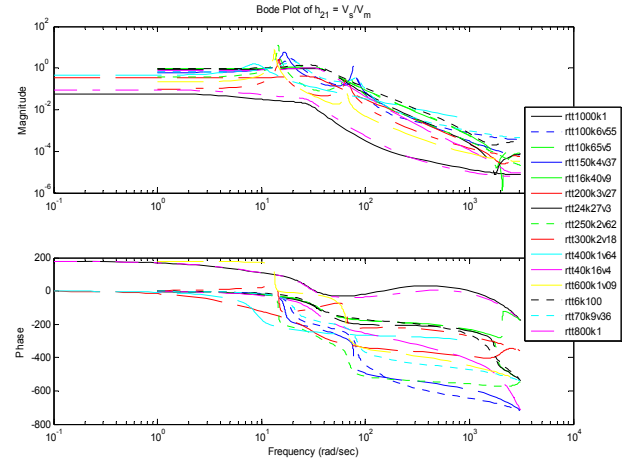


Figure 3. Bode plot of h_{21} with variation in delay.

In Fig. 3, variation of gain and phase with frequency is shown. As delay increases, the cutoff frequency decreases so as the stability margins. This fact is more visible from Fig. 4, 5, 6 and 7.

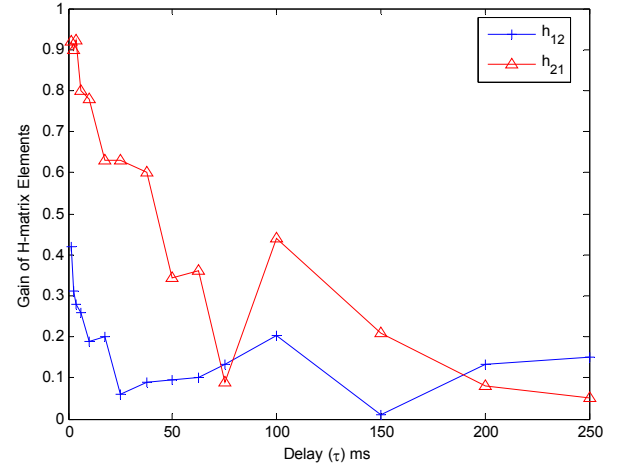


Figure 4. Gain variations with delay for h_{12} and h_{21} .

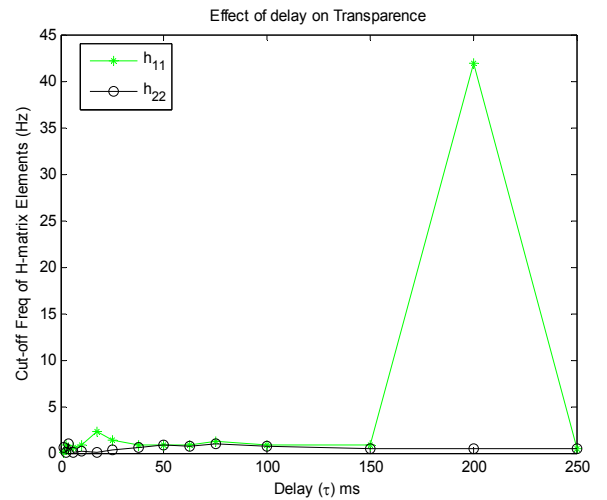
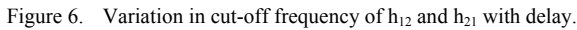


Figure 5. Variation in cut-off frequency of h_{11} and h_{22} with delay.

forecasting as in [21]. Support Vector Machine (SVM) has the property to transform a training vector into a high dimensional



The diagram illustrates the proposed network control architecture, divided into two main sections: a high-level control loop and a detailed network component view.

High-level Control Loop:

- Block 1: Low Pass Filtering** receives an external input r_f and provides a signal to Block 2 and Block 3.
- Block 2: QoS Estimation** receives Pkt_loss from Block 1 and provides a signal to Block 5.
- Block 3: Gain Scheduler** receives a signal from Block 1 and provides a signal to Block 4.
- Block 4: QoS Estimation** receives a signal from Block 3 and provides a signal to Block 5.
- Block 5: Network QoS Control** receives signals from Block 2 and Block 4.

Detailed Network Component View (Yellow Shaded Area):

- Human:** Interacts with the MASTER component.
- MASTER:** A green block that receives signals F_1 and F_2 from the Human and the Network. It outputs x_m to the Scattering Transformation block.
- Scattering Transformation (Left):** A yellow block that receives x_m and outputs U_m to the Network. It also receives F_3 and F_4 from the Network.
- Network (6):** A central purple block that receives U_m and U_s from the Scattering Transformation blocks. It outputs x_n to the Scattering Transformation (Left) and x_s to the Scattering Transformation (Right).
- Scattering Transformation (Right):** A yellow block that receives x_s and outputs U_s to the Network. It also receives F_5 and F_6 from the Network.
- SLAVE:** A green block that receives U_s from the Scattering Transformation (Right) and outputs x_e to the Environment. It also receives F_7 and F_8 from the Environment.
- Environment:** Interacts with the SLAVE component.

External Flows:

- r_f : Input to Block 1.
- Pkt_loss : Signal from Block 1 to Block 2.
- r_{net} : Input to Block 6.
- $Video_pkt_rate$: Input to Block 6.

Figure 7. Co-adaptation strategy for bilateral Teleoperation.

(theoretically infinite) dimensional space by the function. SVM finds a linear separating hyper plane with the maximal margin in this higher dimensional space. In [17], a new use of SVM as a prediction tool is proposed. In this work, the idea has been modified to an adaptive (one sample ahead) prediction tool, for an online implementation.

B. Prediction quality and performance

There are several hyper-parameters of SVM learning machine module that need to be arbitrary chosen in order to ensure the best possible quality and performance of prediction. These parameters are:

- **Kernel type** - For the kernel type, RBF (Radial Basis Function) is preferred since other studies as [16], [24] and [15] have shown that it is the most suitable one for the prediction problem.
- **c** – Cost parameter, is a tradeoff between model flatness against tolerance of deviations larger than epsilon in optimization formulation.
- **ϵ** – Epsilon parameter which determines the zone of insensitivity of cost function, $\epsilon=0.1$ is found sufficient for our case.
- **κ** – MSE check horizon - Since the proposed module constantly adapts it selves to current network conditions, it is necessary to choose an appropriate horizon of past samples which will be used to calculate current MSE error.
- **α** – MSE (Mean Square Error) check threshold - Past samples determined by MSE check horizon are compared with the same (in terms of time) predicted samples and the SVM model is retrained if MSE is bigger than the MSE check threshold.
- **λ** – Training horizon is the number of past samples used to train the SVM model.
- **μ** – Error threshold for SVM training is chosen as 2 ms for our case which corresponds to the sampling rate of 1 ms.

A. Delay and Packet loss Prediction

The first block of the co-design architecture comprises of delay and extracting the packet loss information from it. The support vector machine (SVM) is a popular method for chaotic time series

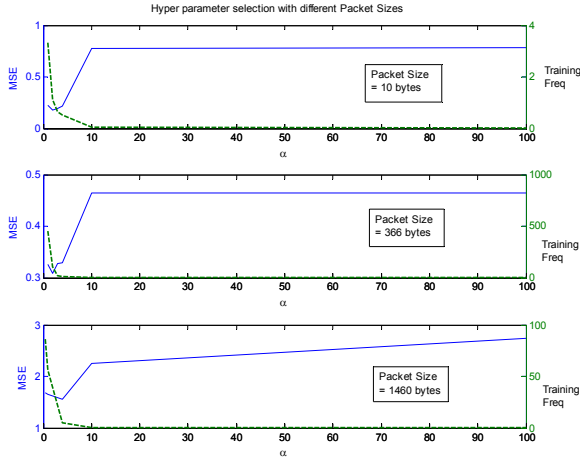


Figure 8. Fuzzy Packet rate surface

In Fig. 9, a selection of key hyper parameters for different packet sizes is shown. Thus, MSE and training frequency are plotted against MSE threshold (α). When large values of ' α ' are allowed, it means that the MSE may go higher and the training frequency will be lower and vice versa. As seen from the Fig. 9, for different packet sizes (10 bytes, 366 bytes and 1460 bytes), the optimum MSE threshold is quite close which means that SVM prediction can be used for variable packet sizes.

VII. QUALITY OF SERVICE ESTIMATION

The second block is the QoS module to estimate QoS from delay and packet loss. The QoS is defined as the necessary network resources made available for a particular application, so it is dependent on the application requirements. In literature, several criterions are available for QoS estimation e.g. by evaluating the received signal strength (RSS), signal to noise ratio (SNR), bit error rate (BER), available bandwidth, packet loss rate etc. Our approach is to use fuzzy inference for estimation of QoS. This will result in reconfiguring the network flows as well as the controller parameters in order to achieve appropriate gain values for position control, thus ensuring improved quality of control (QoC) in bilateral teleoperation.

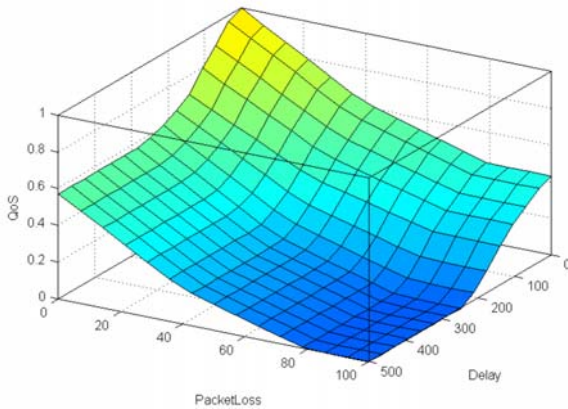


Figure 9. Fuzzy QoS surface

Fuzzy inference system operates with fuzzy sets (F), characterized by a membership function (μ) which gives the degree of similarity of x to F. Thus, FIS is capable of approximating any continuous function with an arbitrary bound B as in [27]. In engineering, the most widely used are the rule-based FIS of Mamdani or Takagi Sugeno type. In this work, only triangular MFs are used in Sugeno FIS for delay (DL) and packet loss (PL) as the two inputs of QoS fuzzy inference block. The range of delay varies from 0 to 500 ms, for packet loss it is 0 to 100 and for QoS it is scaled between 0 and 1. The fuzzy rule base (reduced) for QoS estimation is shown in Table 1, where QoS has more weight for packet loss than delay. This is because of the fact that the information loss has a severe impact on transparency and stability as compared to delay.

TABLE I. FUZZY RULE BASE INSIDE THE QoS ESTIMATOR WITH 2 INPUTS

Rule	DL	PL	QoS	α
1	NE	LW	EX	1
2	NE	HH	GD	1
3	NE	VH	BD	1
4	SL	NE	EX	0.5
5	LG	NE	GD	0.5
6	VL	NE	BD	0.5

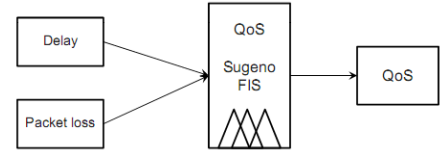


Figure 10. QoS Fuzzy Inference system (FIS)

As noted from above that the simple rules are chosen to ensure rapid decision and low computation for online use. The two input variables are given the states as none (NE), Small (SL), Large (LG) and Very Large (VL) for delay. Low (LW), High (HH), Very High (VH) for packet loss and the QoS output is marked with Excellent (EX), Good (GD) and Bad (BD) levels.

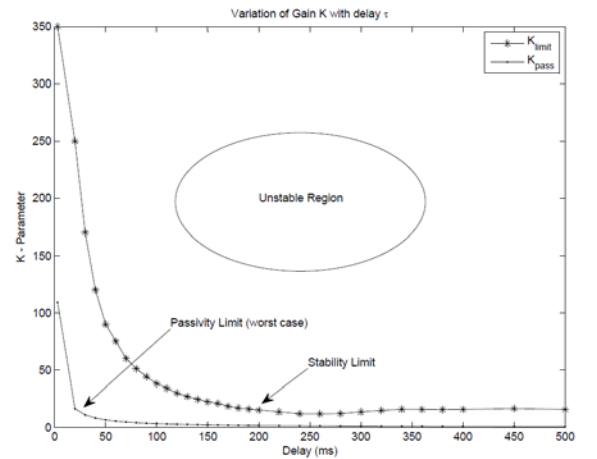


Figure 11. Variation in Controller Gain with delay

Fig. 9 shows the nonlinear surface representing the effect of delay and packet loss on QoS.

A. Gain Scheduling

The third block in the co-design architecture is the gain scheduling. With increasing delay, the gain for the position control loop needs to be decreased in order to respect passivity and therefore stability. The gain is adapted from the relation between delay t and gain K as shown in Fig. 5. The outer curve K_{limit} shows the values of gain where stability is lost as experienced in real time. Next, the inner curve gives the limit imposed by the passivity condition. The practical values of K on the system which takes into account the worst case value of B_{s1} are taken as about 50% of the K_{pass} . Worst case damping values are used in the passivity criteria to make sure that the passivity is respected at all operating conditions.

The variation in passivity condition is dependent on the time delay and varying values of master and slave damping B_m and B_{s1} respectively. B_{s1} varies due to the nonlinearities e.g. backlash (0.2328 rad), dry friction etc in the Rack and Pinion Gearset (RPG) as well as due to wheels in contact with the environment. Whereas, B_{s1} has different values for different drivers. The worst case values for B_{s1} and B_m are estimated as 1.3123 N.m.s/rd (without wheels) and 0.0817 N.m.s/rd (free steering) respectively.

B. Quality of Control Estimation

The fourth block is used to estimate QoC and the tracking performance. It is a combined sum of position, velocity and force errors as follows:

$$QoC_e = 1 - (Gqos_F * \epsilon_F) + (Gqos_x * \epsilon_x) + (Gqos_v * \epsilon_v) \quad (7)$$

where, $Gqos_F$, $Gqos_x$ and $Gqos_v$ are arbitrary scaling weights for errors in force (F), position (x) and velocity (v) as per design objectives and for our experiments, they are taken as 0.001, 1 and 0.1 respectively to normalize the tracking errors as given in Eqs. (8), (9) and (10) below.

$$\epsilon_x = x_s - x_m \quad (8)$$

$$\epsilon_v = v_s - v_m \quad (9)$$

$$\epsilon_F = F_e - \Gamma_m \quad (10)$$

Where, Γ_m is the applied motor torque resulting from the input force/velocity at the master end.

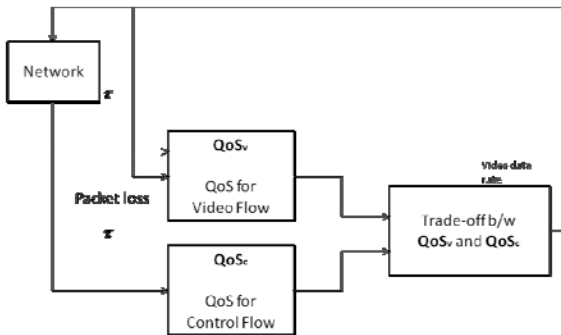


Figure 12. Variation in Controller Gain with delay

The simplified co-design architecture for implementation is shown in Fig. 12. The delay is taken as $\tau_v = \tau_c + \tau_p$, where τ_v and τ_c are the delays in video and control flows respectively. τ_p is the fixed delay in the video flow due to processing except the communication delay on the network i.e. video acquisition, coding, packetization, reception of packets, decoding and video display. This is because τ_c is easy to measure on the master station. The QoC function is a performance measure for the control and tracking performance as well as the time domain on-line transparency which is given in Eq. (7). The tradeoff between QoS_v and QoS_e is given as:

$$QoC = p * QoS_e * (1 - p) * QoC_e \quad (11)$$

where,

$$p = \frac{1}{1 + \left| \dot{X}_m \right|^2}$$

and $|\dot{X}_m|$ is the master's velocity. When $\dot{X}_m \rightarrow 0$, QoS_e will have a unitary value while QoC_e will approach to zero. QoS_e is the

QoS of the control flow dominated by the delay, while QoC_e is based on scaled control errors. This is important to consider because in the steady driving, when there is no control action applied from the driver, there is no need to evaluate QoC. Thus, the only parameter of interest will be QoS which is a direct function of delay. The time varying gain which is the output of controller block is normalized ($K = K/100$) and passed through a low pass filter $G_k = \frac{1}{5s + 1}$, to avoid noisy

switching of the controller. The time constant of the controller has double frequency as compared to the highest frequency of the slave dynamics to avoid aliasing (Slave dynamic model pole $p_s = -0.0907$ rad/s as compared to the controller filter pole $p_k = 0.2$ rad/s).

C. Network QoS Control

The fifth block in the co-design architecture is the Fuzzy QoS management block. The input variables includes QoS, QoC and the distance between the Master/Slave stations, while the output is the video packet rate as shown in Fig. 8. All these I/O variables are ranged between 0 and 1 except the distance which is taken from 0 to 100 m.

To control the delay experienced in the control loop, the video traffic can be varied. This variation can be performed in the steps of 10 packets/sec to vary from 10 packets (121 kbps) to 1000 packets (12.1 Mbps). The non-linear relation between delay and video traffic with the maximum bound is shown in Fig. 8, which is used in the simulation. This delay curve is just used as a reference for simulating the effect of video rate on delay; otherwise, it depends on environment type and time of experimentation. The maximum video rate does not exceeds the (one-way) delay of 250 ms. Delay exceeding 500 ms is considered as the communication failure and as a result the NeCS-Car safety loop stops the car immediately.

As shown in Table 3, the input variables are marked with none (NE), Good (GD), Bad (BD), Average (AV) and Very Large (VL). While, the output has levels such as High (HH), Low (LW), Medium (MD) for packet rate. The packet rate FIS surface is shown in Fig. 13. It can be noted that the role of distance is limited in decision as given by a single rule which states that when the slave is farther than d, packet rate just can't

be made better as the link capacity is limited. This is because the distance measure (GPS or estimated) has large errors as compared to WLAN range and vehicle speed, therefore not suitable for online QoS management. Knowing that the QoC is the actual objective of our design, all QoC rules have double weight in the decision.

TABLE II. MSE CHECK THRESHOLD VERSUS QUALITY AND PERFORMANCE OF SVM PREDICTION (PACKET SIZE 1460 BYTES, TRACE LENGTH 10000 SAMPLES)

c	ϵ	α	κ	λ	Quality (MSE)	Quality (NMSE)	Performance (% trainings per sample)
1	0.1	0.5	16	100	1.6837	0.8147	86.32
1	0.1	1	16	100	1.6602	0.8033	56.69
1	0.1	4	16	100	1.5512	0.7506	5.08
1	0.1	10	16	100	2.2586	1.0929	0.32
1	0.1	100	16	100	2.7448	1.3281	0.01

D. Network Block

The sixth block in the architecture is the network architecture and the chosen protocol. In recent technologies, QoS classes are available to be mapped on flows and ports for guaranteed bandwidth and delay characteristics e.g. 802.11e, 802.1p/q, 802.16 etc.

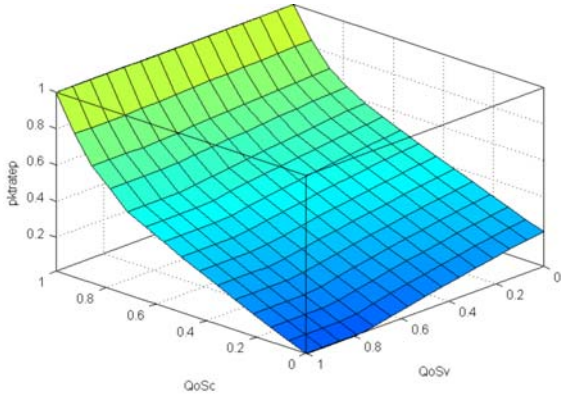


Figure 13. Fuzzy Packet rate surface

In addition, whether the network is wired or wireless also affects the performance. In wireless networks, interference and multipath phenomenon is troublesome.

TABLE III. FUZZY RULE BASE INSIDE THE PR CONTROLLER

Rule	QoS	QoC	Distance	PR	A
1	NE	GD	NE	HH	1
2	NE	BD	NE	LW	1
3	NE	AV	NE	MD	1
4	GD	NE	NE	HH	0.5
5	BD	NE	NE	LW	0.5
6	AV	NE	NE	MD	0.5
7	NE	NE	VL	LW	1

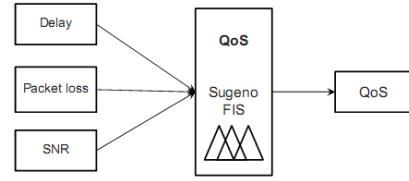


Figure 14. Fuzzy Inference system (FIS) for Packet rate generation

Dual band routers offer redundant frequency channels and two different SSIDs to switch networks in case of degraded QoS over one. External flows are simulated to vary network load and observe performance changes.

E. Co-Adaptation with QoS

The co-design architecture is implemented on NeCS-Car for real time performance. Fig. 15 shows the comparison of adaptation of video data rate with variations in delay and packet losses with the case where fixed video data rate ($\phi_{fix} = 10Mbps$) is sent over the network. This test is performed in the static conditions and inside the lab. At 55 sec and 105 sec, two external video flows of ($\phi_{ext1} = \phi_{ext2} = 3Mbps$) over TCP/IP are added to simulate the network perturbations. This results in low QoS, due to increased delay and packet loss detected by the QoS module. Thus, the QoS controller generates the respective gain K and video packet rate f_n for the bilateral teleoperation. The result proves the corresponding decrease in video rate and improvement in QoC as shown in Fig. 16. It can be noted that in case of fixed data rate, the external perturbations would result in very low gain for position controller and poor tracking performance. The improved performance in QoC is obtained at the cost of deterioration in video quality due to decreased video data rate. However, this degradation is not annoying for the operator (up to a limit defined by human perception) as seen in the quality of video context.

In the second part, we carried out dynamic testing in a u-shaped path, where the vehicle comes back towards the master after around 60 sec as shown in Fig.17. As can be noted, the distance between the master and slave increases, due to the decreasing QoS_c and increasing delay and packet losses, the change in video data rate (above the nominal data rate) reduces to zero at 50 sec. At this time the return path started and the network QoS starts improving again. While the rising time constant for the data rate is quite larger as compared to the decreasing time constant, the video rate mounts quite slowly, offering lowest possible delay for the control flow. Due to poor network conditions, the vehicle stopped at 100 sec as the delay rises instantly and the safety loop comes into action to stop the car. A v-shaped trend in the video rate can be noted which exactly reflects the NeCS-Car movement away from the base station and towards it when QoS is improved which results in higher video rate. The position, velocity and force errors are presented in the sub-figure which corresponds to the QoS as reflected from the packet loss curve.

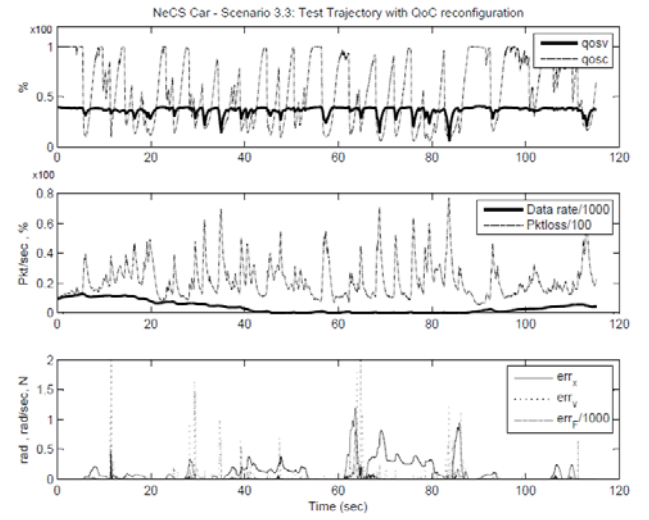


Figure 17. Video Load Management (Dynamic Case)

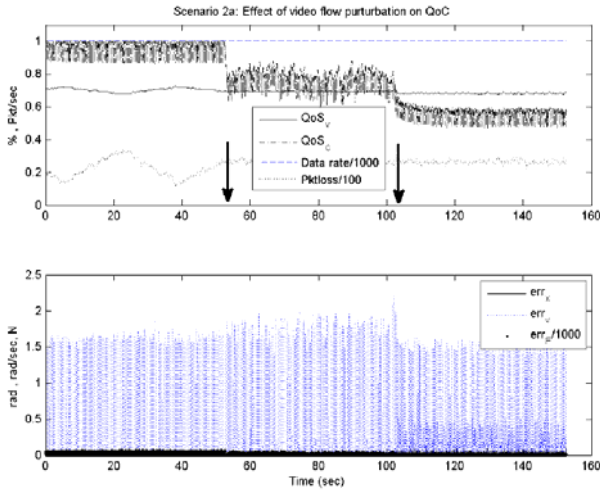


Figure 15. Effect of video flow perturbation (static case).

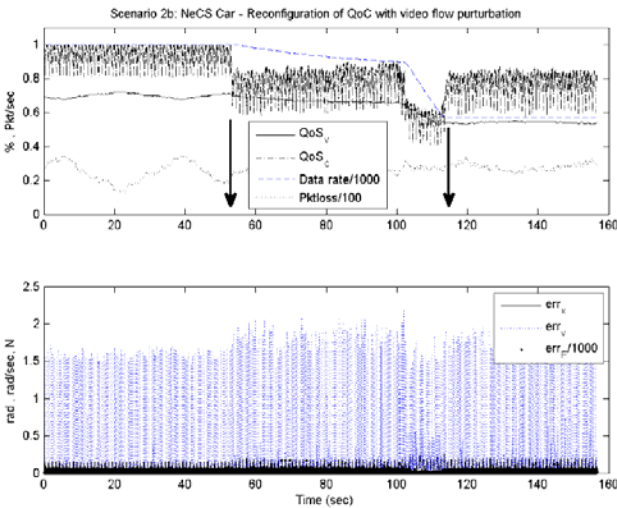


Figure 16. Reconfiguration of video load as per QoS and QoC

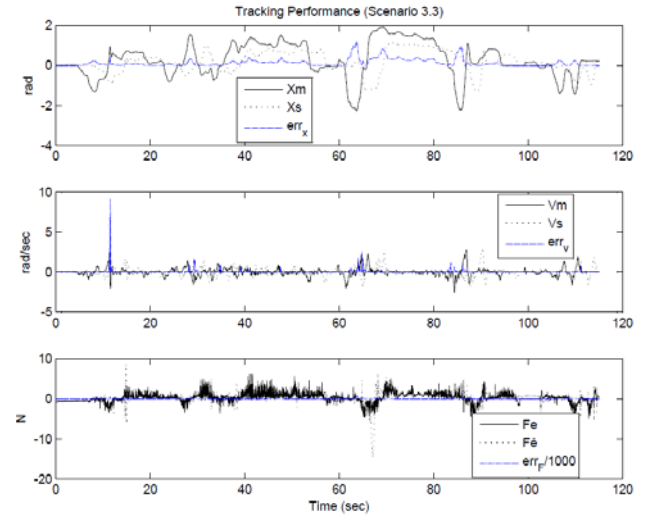


Figure 18. Tracking performance with QoS management.

Fig. 18 shows the tracking performance for the dynamic scenario. It can be seen that the control errors are notable at around 60 sec when a sudden drop in QoS was experienced due to the farthest distance between the master and slave. At most of other times the tracking performance is good as the master position is faithfully followed by the slave.

F. Effect of video packet rate on image information

The video is transmitted at 25 fps with varying image information in bits per pixel which affects directly the video data rate per frame sent over network.

In Fig. 19(a), original image is shown. Whereas, in Fig. 19(b), 1000 pk/s video data is sent which corresponds to 12 Mbps over the network. In the subsequent Fig. 19(c), (d) and (e), the information per frame is decreased further up to 500, 200 and 90 pk/s resulting in data rates of 6, 2.4 and 1.08 Mbps respectively.

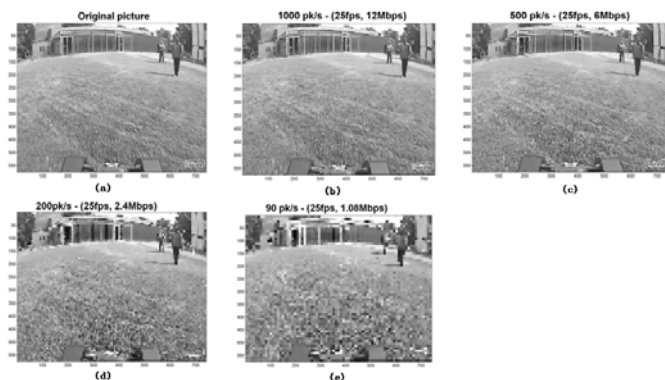


Figure 19. Fuzzy Packet rate surface

The results shows that even with minimum bandwidth, the remote driver is able to differentiate between the obstacles in order to avoid it. Thus, varying wireless network load strategy in case of lowered QoS can be effectively applied to drive-by-wireless applications.

VIII. CONCLUSION

This paper presents a co-design approach to cater for the effect of degradation of network QoS on the control performance in bilateral teleoperation of drive-by-wireless applications. A fuzzy logic controller is used to manage the video flows (which as a result effects the network delay and its variation), in order to guarantee the stability by respecting the passivity criteria. The video rate adaptation is found to improve the tracking performance. However, more analysis will be worked out on the aspect of transparency in the future work. An extension is also possible by considering internet for teleoperation of drive-by-wireless application.

ACKNOWLEDGMENT

This work has been supported by the Higher Education Commission (HEC) of Pakistan under OSS-06 Program. The authors would like to thank Jonathon Dumon and other technical staff of GIPSA-lab for providing adequate experimental setup.

REFERENCES

- [1] P. Donoghue, G. Larson and S. Dondoe, "Extending technology to drive by wire control", Aerospace and Electronics Conference, NAECON 1994., Proceedings of the IEEE National Conference, 386-394 vol.1. doi: 10.1109/NAECON.1994.332873.
- [2] L. Ma and K. Schilling, "Survey on bilateral teleoperation of mobile robots." Proceedings of 13th IASTED International Conference on Robotics and Applications, 2007, pp. 489-494.
- [3] M. Bertoluzzo, G. Buj and A. Zuccollo, "Design of drive-by-wire communication network for an industrial vehicle," Industrial Informatics, INDIN '04. 2nd IEEE International Conference on, 2004, pp. 155-160. doi:10.1109/INDIN.2004.1417320.
- [4] O. Martinez-Palafox, D. Lee, M.W. Spong, I. Lopez and C. Abdullah, "Bilateral teleoperation of mobile robot over delayed communication network: Implementation." Proceedings of the IEEE/RSJ International Conference on Intelligent robots and systems, pp. 4193-4199, 2006.

- [5] L. Zhang, C. ZhiXin and W. Jia, "A networked teleoperation system for mobile robot with wireless serial communication," IEEE International Conference on Intelligent Computing and Intelligent Systems, pp. 885-889, 2009.
- [6] Y. Yokokohji, T. Imaida and Y. Iida, "Bilateral teleoperation: Towards fine manipulation with large time delay," 2001, Springer-Verlag Berlin.
- [7] S. Salcudean, "Control of teleoperation and haptic interfaces. Control problems in robotics and automation," Springer-Verlag LNCIS 230, pp. 51-66, 1998.
- [8] B. Hannaford and J. H. Ryu, "Time domain passivity control of haptic interfaces," IEEE Transactions on Robotics and Automation, Vol. 18, 2002.
- [9] Z.H. Khan, D. Genon-Catalot and J.M. Thiriet, "A 2-tier wireless network architecture for diagnosis and monitoring applications." IEEE-CCNC 2009, Las Vegas, USA.
- [10] M.B. Gaid, D. Robert, O. Senane, A. Suret and D. Simon, "Co-design Approaches for dependable Networked Control Systems," 2010, Chapter- Computing Aware Control, pp. 70-71. Wiley.
- [11] R. J. Anderson and M.W. Spong, "Bilateral control of teleoperation with time delay," IEEE Transactions on Automatic Control, 1989, Vol. 34, pp. 494-501.
- [12] G. Niemeyer and J.J.E. Slotine, "Stable adaptive teleoperation", IEEE Journal of Oceanic Engineering, vol. 16, pp. 152-162.
- [13] N. Chopra, M. W. Spong, R. Ortega and N.E. Barabanov, "Position and force tracking in bilateral teleoperation," Advances in Communication Control Networks, 2004, pp. 269-280.
- [14] P.F. Hokayem and M.W. Spong, "Bilateral teleoperation: An historical Survey," Automatica, Vol. 42, Issue 12, pp. 2035-2057, 2006.
- [15] L. J. Cao, E. Francis and H. Tay, "Support vector machine with adaptive parameters in nancial time series forecasting," IEEE Transactions on Neural Networks, 2003, Vol. 14.
- [16] N. Cristianini and J. S. Taylor, "An introduction to support vector machines and other kernel-based learning methods," 2000.
- [17] H. Feng, Y. Shu, S. Wang and M. Ma, "SVM based models for predicting WLAN traffic", 2006.
- [18] S. Hirche, P. Hinterseer, E. Steinbach and M. Buss, "Towards deadband control in networked teleoperation systems." In Proceedings IFAC World Congress 2005, Prague, Czech Republic.
- [19] D.A. Lawrence, "Stability and transparency in bilateral teleoperation." IEEE Transaction in Robotics and Automation, 1993, Vol. 9, pp. 624-637.
- [20] Z. Li, W. Wang and Y. Jiang, "Managing quality of control and requirement-of-bandwidth in networked control systems via fuzzy bandwidth scheduling." International Journal of Control, Automation, and Systems, Vol. 7(2), pp. 289-296, 2009.
- [21] S. Mukherjee, E. Osuna and F. Girosi, "Nonlinear prediction of chaotic time series using support vector machine," 1997.
- [22] S.A. Munir, Y. Bin, R. Biao and M. Jian, "Fuzzy logic based congestion estimation for qos in wireless sensor network," IEEE International Symposium on Wireless Communications and Networking, pp. 4336-4341, 2007.
- [23] L. Pirmez, F.C. Delicato, P. Pires, A. Mostardinha and N. de Rezende, "Applying fuzzy logic for decisionmaking on wireless sensor networks", 2007, IEEE International Symposium on Fuzzy Systems, vol.16.
- [24] J. Suykens, T. van Gestel, J. de Brabanter, B. de Moor, and J. Vandewalle, "Least squares support vector machines," 2002.
- [25] A.L. Wilson, A. Lenaghan and R. Malyan, "Optimizing wireless access network selection to maintain QoS in heterogeneous wireless environments". Proceedings of the Wireless Personal Multimedia Communications (WPMC 2005), Aalborg, Denmark.
- [26] F. Xia, S. Li and Y. Sun, "Neural network based feedback scheduler for networked control system with flexible workload," 2005, Lecture Notes in Computer Science, Springer-Verlag, Berlin, 3611, pp. 242-251.
- [27] L.A. Zadeh, "A computational approach to fuzzy quantifiers in natural languages," Computers and Mathematics, Vol. 9, pp. 149-184, 1983.

- [28] W. Zhang, "Stability Analysis of Networked Control Systems," Ph.D. thesis, Dept. of Electrical Engineering and Computer Science, Case Western Reserve University, 2001.
- [29] M. Zhu, S. Salcudean, "Achieving transparency for teleoperator systems under position and rate control", IEEE International Conference On Robotics and Automation pp. 7–12, 1995.
- [30] D. A. Lawrence, "Stability and transparency in bilateral teleoperation", IEEE Transaction in Robotics and Automation 9, 624–637, 1993.
- [31] K. B. Fite, L. Shao and M. Goldfarb, "Loop Shaping for Transparency and Stability Robustness in Bilateral Telemanipulation", IEEE Transaction on Robotics and Automation 20, 620–624, 2004.
- [32] J. Kim and P. H. Chang, "Extended transparency as a quantitative performance measure in bilateral teleoperation", IEEE/RSJ International conference on Intelligent Robots and Systems San Diego, CA, USA .