Bilateral Teleoperation of Mobile Robot over Delayed Communication Network: Implementation.

Oscar. M. Palafox, Dongjun Lee and Mark W. Spong
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
1308 W. Main St. Urbana IL 61801
d-lee@ieee.org, {pomartin,mspong}@uiuc.edu

Ivan Lopez and Chaouki T. Abdallah
Electrical & Computer Engineering Department
University of New Mexico
Albuquerque, NM 87131-0001
{ilopez,chaouki}@ece.unm.edu

Abstract— In a previous paper we proposed a bilateral teleoperation framework of a wheeled mobile robot over communication channel with constant time delay. In this paper we present experimental results. Our goal is to illustrate and validate the properties of the proposed scheme as well as to present practical implementation issues and the adopted solutions. In particular, the bilaterally teleoperated system is passive and the system is stable in the presence of time delay. Internet has been used as the communication channel and a buffer has been implemented to maintain a constant time delay and to handle packet order.

I. INTRODUCTION.

Many applications required robotics agents to cover large work spaces. Surveillance, rescue [1] and exploration [2] are examples of such applications. Mobile robots are the typical solution to the automation of such tasks.

The teleoperation of mobile robots is an extension of the aforementioned applications useful when the task requires human intelligence in the control loop due to unstructured or unknown environments. Bilateral control provides another dimension to the teleoperation enabling the operator to interact with the remote environment by force feedback [3], complementing other sensory feedback (i.e. visual) specially when they are either unavailable or occluded (i.e. obstacle in the back of the camera while moving backwards) and the task relies on mechanical interaction with the environment (i.e. pushing or handling an object).

In our approach, we consider a car-driving metaphor to teleoperate the mobile robot with force feedback. This metaphor consists in using \( q_1 \) as linear velocity reference (Figure 1), as if it were a gas pedal, and \( q_2 \) as heading angle reference as if it were a driving wheel. To achieve this metaphor, a basic requirement in our framework is the coordination between \( q_1 \) and \( \nu \) (linear velocity control) and between \( q_2 \) and \( \phi \) (heading angle control). For \((q_2,\phi)\)-coordination many schemes are readily available [5], [6]. However, this is not the case for \((q_1,\nu)\)-coordination because those passivity based teleoperation schemes are usually based on the open loop passivity of the master/slave system with power as supply rate and mechanical power is a function of velocity rather than position. Then, the problem we tackle is to redefine the passivity requirements of the teleoperation loop to incorporate the use of velocity and position information to construct the linear velocity and heading angle references for the slave robot such that we can guarantee passivity of the controller when we have force feedback in a delayed communications channel.

To solve this problem, we proposed in [4] a local control to modify the passivity of the master system with a supply rate that contains position information. In this paper, we present the experimental results for the implementation of this scheme on an Internet based teleoperation set up.

![Figure 1](image-url)

The problem formulation is presented section II, the control scheme is reviewed in section III, experimental set up is described in section IV. Section V contains experimental results and discussion, conclusions and future work can be found in section 6.
II. PROBLEM FORMULATION.

A. System Modeling

We consider the dynamics of a differential wheeled mobile robot (with nonholonomic constraints [7]) given by:

$$D(x) \begin{pmatrix} \frac{\dot{v}}{\phi} \\ \frac{\dot{\phi}}{\phi} \end{pmatrix} + Q(x, \dot{x}) \begin{pmatrix} \frac{\dot{v}}{\phi} \\ \frac{\dot{\phi}}{\phi} \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \begin{pmatrix} \delta_1 \\ \delta_2 \end{pmatrix}$$  \(1\)

where \(v\) and \(\phi\) are the linear velocity and the heading angle of the slave mobile robot, \(x := (x, \eta, \phi, \theta_1, \theta_2)\) is the 5-degrees of freedom (5-DOF) configuration of the mobile robot with \((x, \eta, \phi)\) and \((\theta_1, \theta_2)\) being the position of the cart and the rotation of the right/left wheels, respectively, \((u_1, u_2) := \left(\frac{1}{h}(u_1 + u_2), \frac{1}{h}(u_1 - u_2)\right)\) are the controls with \((u_1, u_2)\) being the angular torques directly applied on the right and left wheels, \(h\) being the radius of the wheels and \((\delta_1, \delta_2)\) are the external force/torque acting on the cart (Figure 1). The inertia matrix \(D(x)\) is symmetric and positive-definite, and \(\frac{d}{dt} D(x) - 2Q(x, \dot{x})\) is skew symmetric. The evolution of the 5-DOF dynamics can be computed by solving the reduced 2-DOF (1) and the following linear robotic dynamics:

$$M \ddot{q} = \eta + f$$  \(3\)

where \(q = [q_1, q_2]^T, \eta = [\eta_1, \eta_2]^T\) and \(f = [f_1, f_2]^T \in \mathbb{R}^2\) are the configuration of the joystick, control torque, and human force, respectively. Similar to (1), \(M\) is symmetric and positive-definite. Here, note that the dynamics (3) does not have Coriolis terms, as it is linear. We assume the communication structure as shown in Figure 1, where the forward (i.e. master to slave) and backward (i.e. slave to master) communications are subject to (finite) constant time-delays, \(\tau_1 \geq 0\) and \(\tau_2 \geq 0\), respectively.

B. Passivity with redefined input-output pair.

The slave mobile robot possesses the following energetic passivity property [8], [9]:

$$\int_0^T \begin{pmatrix} u_1 + \delta_1 \\ u_2 + \delta_2 \end{pmatrix}^T \begin{pmatrix} \dot{v} \\ \dot{\phi} \end{pmatrix} dt = k_s(t) - k_s(0) \geq -k_s(0)$$  \(4\)

\(\forall T\), where \(k_s(t) := \frac{1}{2} \begin{pmatrix} \dot{v} \\ \dot{\phi} \end{pmatrix} D(x) \begin{pmatrix} \dot{v} \\ \dot{\phi} \end{pmatrix}\) is the slave kinetic energy. Using \(r := \dot{\phi} + \lambda q_1\) where \(\lambda \geq 0\) proposed in [4], and suppose that the master control \(\eta\) is designed s.t. \(\eta := \eta' + \bar{\eta}\), where \(\eta'\) is a local control defined by \(\eta' := [-b_1 q_1 + k_1 q_1, b_2 q_2]^T\) with \(b_1, b_2, k_1 > 0\), and \(\bar{\eta} := [\bar{\eta}_1, \bar{\eta}_2]^T\) is an additional control to achieve the car driving metaphor (to be designed later). Then, for the controlled master dynamics given by

$$M \ddot{q} + \begin{pmatrix} b_1 q_1 + k_1 q_1 \\ b_2 q_2 \end{pmatrix} = \begin{pmatrix} \bar{\eta}_1 + f_1 \\ \bar{\eta}_2 + f_2 \end{pmatrix}\)  \(5\)

we arrive at the following modified passivity condition:

**Proposition 1 [4].** There exists an upper-bound \(\lambda \geq 0\) s.t. for all \(\lambda \leq \bar{\lambda}\), the locally-controlled master satisfies the following modified passivity: \(\forall T \geq 0,\)

$$\int_0^T (\bar{\eta}_1 + f_1)^T \begin{pmatrix} r \\ \bar{\eta}_2 + f_2 \end{pmatrix} dt = k_{ml}(T) - k_{ml}(0) \geq -k_{ml}(0),$$  \(6\)

where \(k_{ml}(t)\) is a positive function as long as \(\lambda \leq \bar{\lambda}\). For a proof, please consult [4].

III. CONTROL DESIGN.

Once the modified passivity (4) and (6) is guaranteed, \(r\) can be thought of just as a velocity satisfying its energetic passivity (i.e. \(r\) and (6) as \(v\) and (4)). Thus, the \((r, v)\)-coordination problem can also be thought of as a standard velocity-coordination problem of teleoperation. Also, the \((q_2, \phi)\)-coordination is merely a standard position-coordination problem of teleoperation. Therefore, for both the linear velocity and heading angle controls, many passivity-based bilateral teleoperation schemes can be applied (e.g. [6], [10], [11]). In this paper we used the scheme proposed in [6], as it can explicitly enforce the coordination with relatively simple PD-control structure over delayed communication network.

Following [6], we design the linear velocity control \((\bar{\eta}_1, u_1)\) s.t.

$$\bar{\eta}_1(t) := -k_v (v(t) - v(t - \tau_2)), \quad (7)$$

$$u_1(t) := -k_v (v(t) - r(t - \tau_1)), \quad (8)$$

where \(\tau_1 \geq 0, \tau_2 \geq 0\) are the (constant) forward/backward communication-delays, and \(k_v > 0\) is the control gain. We also design the heading angle control \((\bar{\eta}_2, u_2)\) to be

$$\bar{\eta}_2(t) := -b_\phi (\phi(t) - \phi(t - \tau_2)) - b_q \dot{q}_2(t) - k_\phi (\phi(t) - \phi(t - \tau_2)), \quad (9)$$

$$u_2(t) := -b_\phi (\phi(t) - \dot{q}_2(t - \tau_2)) - b_q \dot{q}_2(t) - k_\phi (\phi(t) - \dot{q}_2(t - \tau_2)), \quad (10)$$

where \(b_\phi > 0, b_q > 0\) are the PD-control gains, and \(b_d > 0\) is the dissipation. Following [6], to enforce passivity of the delayed P-action (i.e. with \(k_\phi\), the dissipation gain \(b_d\) in (9)-(10) is set to be

$$b_d \geq \frac{\tau_1 + \tau_2}{2} k_\phi. \quad (11)$$

Note, from (7)-(10), that each control consists of the local sensing (e.g. \(\dot{q}_2(t), q_2(t)\) in (9)) and the delayed information directly received from the delayed communication channel (e.g. \(\dot{\phi}(t - \tau_2), \phi(t - \tau_2)\) in (9)). In this sense the design controls are distributed.

**Theorem 1 [4].** Consider the slave mobile robot (1) and the locally controlled master (5) with the modified passivity condition (6) under control (7)-(10).

1) The closed-loop teleoperator is passive in the sense that \(\exists \) a finite \(d\) s.t. \(\forall T \geq 0,\)

$$\int_0^T |(f_1 r + f_2 \dot{q}_2) + (\delta_1 v + \delta_2 \phi)| dt \geq -d^2, \quad (12)$$
2) Suppose that the human operator and the slave environment are passive in the sense that \( \exists \) finite \( d_1, d_2 \) s.t.

\[
\int_0^T (f_1 r + f_2 \dot{q}_2) dt \leq d_1^2, \quad \int_0^T (\delta_1 v + \delta_2 \dot{\phi}) dt \leq -d_2^2,
\]

\( \forall T \geq 0 \). Then, \( q_1(t), \dot{q}_1(t), q_2(t), v(t), \dot{\phi}(t) \) are all bounded \( \forall t \geq 0 \). Also, the coordination errors \( r(t) - v(t) \) and \( q_2(t) - \phi(t) \) are bounded \( \forall t \geq 0 \).

3) Suppose that \( (q_1(t), \dot{q}_1(t), v(t), \dot{\phi}(t), q_2(t), \dot{q}_2(t), \phi(t)) \rightarrow 0 \).

Then, \( f_2(t) \rightarrow -k_0(q_2(t) - \phi(t)) \rightarrow -\delta_2 \) (i.e. heading-angle torque reflection). Also, if \( \delta(t) \rightarrow 0 \), \( f_1(t) \rightarrow \frac{1}{2} v(t) \) (i.e. human can perceive the slave linear velocity), and, if \( v(t) \rightarrow 0 \), \( f_1(t) \rightarrow \frac{k_1 + \delta k_v}{\lambda k_v} \delta(t) \) (i.e. linear-force reflection).

Please refer to [4] for a proof.

IV. EXPERIMENTAL SETUP.

A. Master Site.

The master site is located in the Robotics Lab of the Coordinated Science Laboratory at University of Illinois at Urbana-Champaign. We used a commercial haptic device (PHANToM® Desktop™, figure 3(a)) as master device connected to a desktop PC running Windows XP® Professional. We used the 2-DOF (distance from the origin as actuated DOF of the master haptic joystick. We set \( \lambda \) to be small enough by trial and error so that the system behaves well. The master local control loop runs at 1 kHz. We use Internet as the communication channel using UDP. We chose UDP since it is a protocol better suited for real time control given that it does not increases the time delay waiting for packets to arrive. The master robot is connected to Internet using a wired ethernet connection while the slave robot is connected using wireless connection. Three implementation problems have been taken care of to make the communication possible: 1) time-varying-delay [12], [13], [14], 2) packet reordering and packet loss [13], [16], and 3) wireless disconnection [14]; all of this while preserving passivity [13], [17]. The explanation of how we solved this problems in our scheme is presented in the implementation section. We performed experiments using wired and wireless connections and found that the wireless communications suffers from severe disconnections that can last up to 10 seconds. This problem produces a deviation in the states of master and slave that causes a radius reduction problem [19] when communication is regain (i.e. the master and slaves jump to try to catch up with each other after communication is recovered which may be an undesirable behavior when carrying out a delicate teleoperation task).

B. Slave Site.

The slave mobile robot is located at the Network Control Systems Laboratory at the University of New Mexico (UNM). The slave robot is a differentially-driven vehicle as shown in figure 1. It was built in UNM’s lab and it consist of DC motors attached to each wheel. The motors were driven with Advanced Motion Controls® power drivers, and powered by two 12 Volts batteries. A PCMCIA data acquisition card, DAQ 6024E from National Instruments™, was used to interface the laptop computer to the drivers and other electronic peripherals of the robot. The software programs used to acquire the state’s measurements from the encoders and to apply control signals to the motors, as well as to implement the communication routines, were developed in LabView®, another National Instruments™ product. Encoders were used to calculate \( \dot{\phi} \) and \( \dot{\theta} \) through algebraic kinematic relationships. The laptop was running standard Windows XP Professional and it was connected to the Internet through a WLAN Card. The program took advantage of the multi-threading capability of LabView®.
that allowed us to execute several routines in “parallel”\(^1\). The program was composed of two loops, one for the UDP communication process and one for the state measurement and the control action calculation, thus avoiding any bottlenecks in the individual processes executions.

The mechanical model of the robot was complemented with the gains obtained from the digital readings of the encoders, the frequency-to-voltage converters gains, and the DAC gains. We get then the following equations for the volts applied to each motor amplifier:

\[
\text{Volt}_r = \frac{\tau_r}{286000} - 0.3, \quad (14)
\]

\[
\text{Volt}_l = \frac{\tau_l}{266000} - 0.3, \quad (15)
\]

where \(\text{Volt}_r\) and \(\text{Volt}_l\) are the voltages applied to the right and left motor amplifiers, respectively. It is important to notice that these huge denominators required us to scale the controller gains in the haptic device control law. This explains the large controller gains used in the next section.

### C. Implementation.

The implementation can be schematically depicted as in Figure 2. The implemented control scheme is divided into blocks as follows: 1) a communications block that includes: a) UDP sockets read and write functions to socket connected to the opposite side of the loop, c) packet order check, each packet is assigned an increasing number as identification which is checked upon arrival, if it is greater than the last one received then it is accepted, otherwise it is dropped; d) disconnection condition checkpoint, if no packet is received at any given time nothing is stored in the buffer (only "bilateral" disconnection, i.e. no information flows in any direction, is considered since it is the only type of disconnection noticed in our experiments); e) the buffer, when the buffer reaches a size of two samples (this can happen when there are 8 packets lost consecutively) zeros are stored to maintain at least two samples available, this case is when communication loss is assumed, similarly if the buffer size increases above 10 samples the last sample received is dropped until the controller dequeues one sample making space for a new sample in the buffer; and 2) the controller that contains: a) implementation of equations (7-10), b) display and save data module, references sent from the master and the information received from the slave are stored in a text file in the master's computer.

The time varying delay is reduced to a constant time delay by implementing a buffer [20] that stores the last 10 samples received. The buffer "samples the Internet" (i.e. the master's socket is read) every 60 ms. The buffer "samples the Internet" (i.e. the master’s socket is read) every 60 ms. To obtained the rate to sample the Internet, the time delay between master and slave sites was experimentally measured and found to be 36 ms round trip in average. Then, we added a computing time margin. The length of the buffer is a compromise between the amount of variation in the time varying delay that is needed to be compensated and the amount of constant time delay that will be added by the buffer. So, the buffer should be long enough to keep some information stored at all time regardless of the delay variations and, at the same time, short enough to have a small as possible constant time delay. In our case, the constant time delay obtained is 600 [ms].

Packet reordering has been treated in previous works, see for example [13], [17]. In our case, the problem of packet reordering is solved in a simple way. Any packet that is not received in order is dropped, i.e. if packet \(p_i\) arrives at time \(t_i\) then any packet \(p_k\) for \(k < i\) will be dropped if it is received at any time \(t_k > t_i\). This avoids time swap. So, after packet \(p_j\) is received only packets \(p_j\) with \(j > i\) are used to be stored in the buffer. The case when disconnections occur is also taken into consideration. This is also useful in general, since disconnection may occur independently of the communication channel for different reasons. Whenever the communication is lost, (which may occur to a wireless connection due to weakening and loss of signal), both slave and master robots used all the data in their buffers and since nothing else is being received the buffers are filled with zeros. This preserves passivity as was shown in [17] when using wave variable to encode the transmitted signals. In our case, this prevents unwanted behavior when the communication is restored. We consider that this preserves passivity as long as the damping gains \(b_S\) and \(b_D\) are large enough to passify.

\(^1\)While not truly parallel, this approach allowed the program to allocate CPU time to every subroutine in a balanced form so that no subroutine had to wait for the completion of another making it appear parallel for slow systems such as ours.

Fig. 2. Implementation diagram.
\(k_{phi}\), since in practice: no infinitely fast switching may occur (communication loss do not occur very often), the inputs are bounded. However, a bound on the damping gains as a function of \(k_{phi}\) to preserve passivity under arbitrary switching is currently being investigated. Since zeros are being used in the absence of communication (i.e. all received variables are set to zero), then \((\dot{q}_1, \dot{q}_2, q_1, q_2 - q_0) \to 0\) (where \(q_0\) is arbitrary set point.) in words, the master robot moves toward the "home set point" in the center of its working space where the references that would be sent to the slave are zero. In the meantime, in the slave side, we also set all receiving variables (which would be the references if there were communications with the master) to zero and so \((\dot{\phi}, \dot{\theta}) \to 0\). This ensures that when communication is regained, both master and slave are in the same state such that the radius reduction problem may be reduced or eliminated.

V. RESULTS.

Experiments to demonstrate the behavior of the system under disconnection and normal communication stages were performed and the results are shown in the following subsections. For the experiments, the controller was tuned with the final gains set as follows: \(K_{rv} = 45.0, B_{\phi \theta} = 2 \times 10^5\) and \(K_{\phi \theta} = 2.7 \times 10^6\).

A. Teleoperation with disconnection events.

The disconnection events were found when using a wireless card. This problem inspired us to compensate for such events. The mechanism of compensation is described in section (IV-C).

The experiment consists of driving the slave robot around an obstacle. Four disconnection events were forced by manually shutting down the communications to exaggerate the effects and show the response of the system to these events. These disconnections occur at times \(t = 12, 31, 42, 46, 64\) seconds. These disconnection can be observed as thin impulses in Figure (4(a)) The reconnection happened at time \(t = 20, 41, 53\) seconds as can be better appreciated in figure (4(a)). And it is noticeable that no considerable overshoot occurred after the communication was reestablished. It can be seen that disconnection also occurred naturally (i.e. not manually forced) at times 23, 42 seconds, with respective reconnection times at 23+, 42+. Experiments with wired ethernet connection were also performed but are omitted due to space constraints.

VI. CONCLUSIONS AND FUTURE WORK.

We have proposed an approach to teleoperate a mobile robot considering its kinematic differences with a linear joystick like master device while implementing force feedback and considering constant time delay. Also, we presented practical implementation issues (e.g. time varying delay) and the solutions we found suitable to cope with them (e.g. buffer). An advantage of our approach is the use of only encoder sensors to measure the states of the slave robot. Extending this approach to general nonlinear master devices is a topic for future work.
Fig. 5. Teleoperation with disconnection. The force is a linearly scaled version of the corresponding error signal. No overshoot is found after reconnection of communications.

communication loss (i.e. either the master or the slave still receives information but not the other one), is a topic of ongoing research in our lab.

ACKNOWLEDGMENTS.

The authors would like to thank ONR’s support through grant N000140510186 as well as NSF grants IIS-0233374, NFS-0233205 and ANI-0312611. Also, the research of first and fourth authors was partially funded by CoNaCyT, Mexico.

REFERENCES