

Internet-Based Teleoperation Using Wave Variables With Prediction

Saghir Munir and Wayne J. Book, *Fellow, IEEE*

Abstract—Wave-based teleoperation has been previously attempted over the Internet, however, performance rapidly deteriorates with increasing delay. This paper focuses on the use of a modified Smith predictor, a Kalman filter and an energy regulator to improve the performance of a wave-based teleoperator. This technique is further extended for use over the Internet, where the time delay is varying and unpredictable. It is shown that the resulting system is stable even if there are large uncertainties in the model of the remote system (used in prediction). Successful experimental results using this technique for teleoperation in a master-slave arrangement over the Internet, where the control signal is streamed between Atlanta (Georgia) and Tokyo (Japan), are also given.

Index Terms—Internet, prediction, remote bilateral teleoperation, time delay, wave variables.

I. INTRODUCTION

THE USE OF THE Internet has exploded over the last decade. However, it is still primarily used for emailing, web surfing, stock trading, online shopping etc. Recently several researchers have tried to use the Internet to control physical systems in an open-loop arrangement, meaning that the reference and not the control signal propagates through the net. Even more recently, researchers, have tried to establish bilateral teleoperation via the Internet, which requires transmitting the control signal through the network, exposing the system control loop to the varying time delay of a packet switched network. Time delay, however, occurs in every electro-mechanical system. In most cases it is not noticeable, yet in other cases it can render the system unstable as in the case of teleoperation over the Internet. Several techniques have been tried to compensate for this effect, such as a time forward observer developed for a supervisory control over the Internet by Brady and Tarn ([3], [4] and [16]), a position-based force-feedback scheme implemented by Oboe and Fiorini [13] and a wave variable based technique developed by Niemeyer and Slotine [12].

All techniques but the wave-based formalism require knowledge of the remote plant, which almost always has uncertainties. The stability of such techniques is hinged on how precisely the remote plant is known and modeled. Wave-based techniques do

not require knowledge of the remote plant, however, they suffer from poor performance for delays significantly larger than the time constant(s) of the system.

Wave variables were first introduced by Anderson and Spong ([1] and [2]) and were later presented in a more intuitive, physically motivated, passivity-based formalism by Niemeyer and Slotine ([8]–[11]). Later the use of wave variables was extended to variable delay, as in the case of the Internet ([12] and [17]). However, performance degradation with increased time delay still remains a serious issue. The predictive technique summarized in this paper has previously been presented by the same authors in [7] for a constant delay. This paper shall focus on extending its use to the Internet, where the delay is varying.

Theoretical results are experimentally validated using a teleoperator which relies on the internet to establish a link between the slave and master systems. In this arrangement both the master and slave systems are located in Atlanta, Georgia, however, the control signal is rebounded from Tokyo, Japan. This not only introduces a varying time delay, but the variations are large, erratic and the total closed-loop distance traversed by the control signal exceeds the circumference of the earth (which is approximately 40 000 km).

Section II summarizes how wave variables work (for the unfamiliar reader). Section III deals with the development of a wave-based predictor and an energy regulator. Section IV highlights how time delay varies over the Internet and which protocol should be used for streaming real-time control signals. Section V extends the predictive wave-based technique for use over the Internet. In Section VI experimental results of a teleoperation experiment in a master-slave arrangement over the Internet are given. Finally, the paper is concluded in Section VII.

II. BACKGROUND ON WAVE VARIABLES

In this section, we shall briefly summarize how wave variables work. Consider a single DOF bilateral teleoperator. In this arrangement, the master manipulator is a single DOF crank mechanism bilaterally coupled to a similar slave mechanism. The equation of motion of the master manipulator is

$$J_m \ddot{\theta}_m + b_m \dot{\theta}_m = \tau_m \quad (1)$$

where J_m is the crank inertia, b_m is the crank damping, $\ddot{\theta}_m$ is the crank acceleration, $\dot{\theta}_m$ is the crank velocity, and τ_m the applied torque. The subscript m denotes variables representing the master manipulator. Similarly, the equation of motion for the slave manipulator can be written by interchanging the subscript

Received October 2, 2001; revised January 25, 2002. Recommended by Guest Editor B. Siciliano. This paper was supported by Intel Corporation.

S. Munir was with the George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA. He is now with Intel Corporation, Santa Clara, CA 95052 USA.

W. J. Book is with the George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: wayne.book@me.gatech.edu).

Publisher Item Identifier S 1083-4435(02)05518-7.

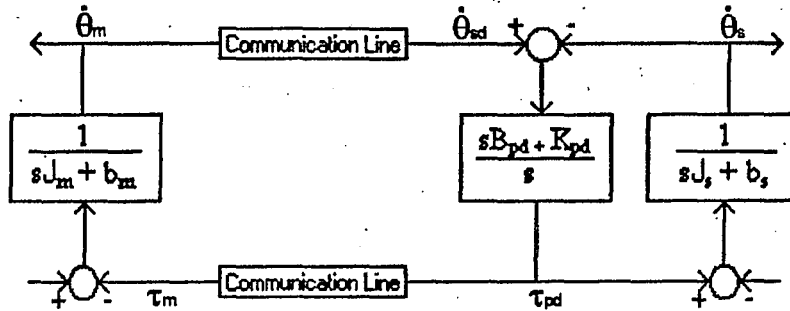


Fig. 1. Single DOF bilateral teleoperator.

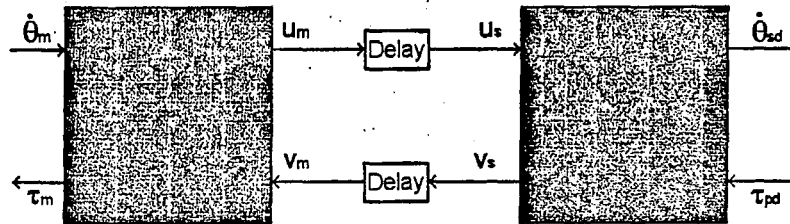


Fig. 2. Wave-based communication, by transforming velocity-force variables to wave variables before transmission and then back to velocity-force variable after transmission. Note: The gray boxes indicate wave transformations.

m with the subscript s for the slave. In order for one side to track the other, a proportional differential (PD) controller is derived such that

$$\tau_m = -\tau_{pd} \quad (2)$$

$$\tau_s = \tau_{pd} \quad (3)$$

and

$$\tau_{pd} = K_{pd}(\theta_m - \theta_s) + B_{pd}(\dot{\theta}_m - \dot{\theta}_s) \quad (4)$$

where τ_{pd} is the torque generated by the PD controller, K_{pd} is the proportional gain, and B_{pd} the derivative gain. This arrangement is schematically represented in Fig. 1. This system performs well for as long as there is no delay in the loop. However, as the delay increases, performance starts to degrade and the system very quickly becomes unstable. This problem is attributed to the nonpassive nature of the communication link, where it can be seen that torque and velocity have multiplicative dependence on the instantaneous power-input (to the communication line) defined as

$$P_{in} = \dot{\theta}_m^T \tau_m - \dot{\theta}_{sd}^T \tau_{pd}. \quad (5)$$

This dependence can be eliminated using the wave transformations

$$\begin{aligned} u_m(t) &= A_w \dot{\theta}_m(t) + B_w \tau_m(t) \\ v_s(t) &= C_w \dot{\theta}_{sd}(t) - D_w \tau_{pd}(t) \end{aligned} \quad (6)$$

and

$$\begin{aligned} v_m(t) &= C_w \dot{\theta}_m(t) - D_w \tau_m(t) \\ u_s(t) &= A_w \dot{\theta}_{sd}(t) + B_w \tau_{pd}(t) \end{aligned} \quad (7)$$

where A_w , B_w , C_w and D_w are $n \times n$ scaling matrices (n being the number of degrees of freedom of the teleoperator) which satisfy¹ the following equations

$$A_w = C_w$$

$$B_w = D_w$$

$$I = 2A_w B_w. \quad (8)$$

Here, wave variables (u and v), rather than power variables (τ and θ), are transmitted across the communication line (i.e., the communication line is replaced with that shown in Fig. 2). Using the above wave transformations it can be shown that the net power-in (5) no longer has a multiplicative dependence on torque and velocity, but rather has an additive dependence. Hence, when signals are temporarily delayed in the communication link, the line does not appear to be nonpassive. For better illustration consider the power flow into the left side on the wave junction shown in Fig. 2. As pointed out by Niemeyer [9] the power-flow is now given as

$$P = \dot{\theta}_m^T \tau_m = \frac{1}{2} u_m^T u_m - \frac{1}{2} v_m^T v_m. \quad (9)$$

Mathematically expressing the energy balance for this system yields

$$\begin{aligned} \int_0^t P d\tau &= \int_0^t \dot{\theta}_m^T \tau_m d\tau \\ &= E_{store}(t) - E_{store}(0) + \int_0^t P_{diss} d\tau \end{aligned} \quad (10)$$

¹The reader is referred to the author's Ph.D. dissertation [6] for a more detailed proof of this analysis.

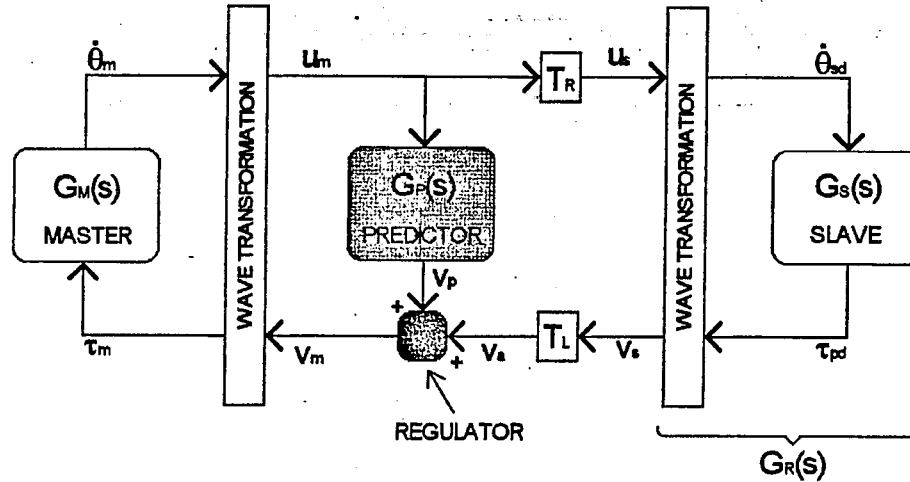


Fig. 3. Possible arrangement of a predictor incorporated inside the wave junction.

where P_{diss} is the power dissipated, $E(0)$ is the initial energy stored, and $E(t)$ is the energy stored at time t . Substituting (9) into (10) leads to

$$\int_0^t \frac{1}{2} v_m^T v_m d\tau = \int_0^t \frac{1}{2} u_m^T u_m d\tau - E_{store}(t) + E_{store}(0) - \int_0^t P_{diss} d\tau \quad (11)$$

which simplifies to

$$\int_0^t \frac{1}{2} v_m^T v_m d\tau \leq \int_0^t \frac{1}{2} u_m^T u_m d\tau + E_{store}(0). \quad (12)$$

Hence, the system is passive if the energy in the outgoing wave is greater than or equal to the energy in the returning wave. This would always be true for as long as the energy dissipated is positive (i.e., the remote system itself is passive). Should the returning wave get delayed, then energy is only temporarily stored in the communication line and released later, still satisfying the passivity condition. Hence, the need to eliminate the multiplicative dependence of what flows through the communication line on power-flow becomes obvious.

III. WAVE PREDICTION AND REGULATION

A possible arrangement of a Smith-type (see [14] and [15]) wave predictor is shown in Fig. 3. Here $G_M(s)$ is the transfer function of the master manipulator, $G_S(s)$ is the transfer function of the slave and PD controller combined, and $G_P(s)$ is the transfer function of the predictor. The two rectangular boxes represent wave transformations and $G_R(s)$ is the combined transfer function of the entire right-hand side (i.e., the slave, the PD controller, and the wave transformation). T_R and T_L represent the time delays in the right and left directions respectively. The box marked as the *REGULATOR* for now can be assumed to be a summing junction.

It can be shown that position information is encoded in the integral of the wave signals u and v . Hence, if the predictor is not designed with care, these integrals might not be preserved, leading to a nonzero steady-state error. Writing out an expression for position difference across the wave junction yields

$$\Delta\theta(t) = \theta_m(t) - \theta_{sd}(t) \quad (13)$$

which, written in terms of wave variables becomes

$$\Delta\theta(t) = \frac{1}{2} A_w^{-1} \int_0^t (u_m + v_m - u_s - v_s) d\tau. \quad (14)$$

Taking the Laplace transform of this and expanding, yields

$$\Delta\Theta(s) = \frac{1}{2s} A_w^{-1} (1 - e^{-sT_R} + G_P(s)) U_m(s) - \frac{1}{2s} A_w^{-1} (1 - e^{sT_L}) V_s(s). \quad (15)$$

For zero steady-state error

$$\lim_{t \rightarrow \infty} \Delta\theta(t) = \lim_{s \rightarrow 0} s \Delta\Theta(s) = 0. \quad (16)$$

Since at steady state the wave signals decay to zero, it is required that

$$\lim_{s \rightarrow 0} G_P(s) = 0. \quad (17)$$

In order to ensure passivity, the predictor must not increase the total return energy of the system. Hence, it is required that

$$\int_0^t \frac{1}{2} v_a^T v_a d\tau \geq \int_0^t \frac{1}{2} v_m^T v_m d\tau. \quad (18)$$

This condition is explicitly enforced by the *REGULATOR* (discussed later in this section). Fig. 4 shows a possible arrangement of a predictor. In this arrangement a Kalman filter first estimates the internal state of the plant, which is delayed by an amount T_T , where

$$T_T = T_R + T_L. \quad (19)$$

The time forward observer then uses the output of the Kalman filter to march the state T_T seconds into the future and computes the output y_p , which is then used to obtain a prediction signal v_p according to

$$V_p(s) = (1 - e^{sT_T}) Y(s). \quad (20)$$

Notice that this expression satisfies the tracking condition given by (17).

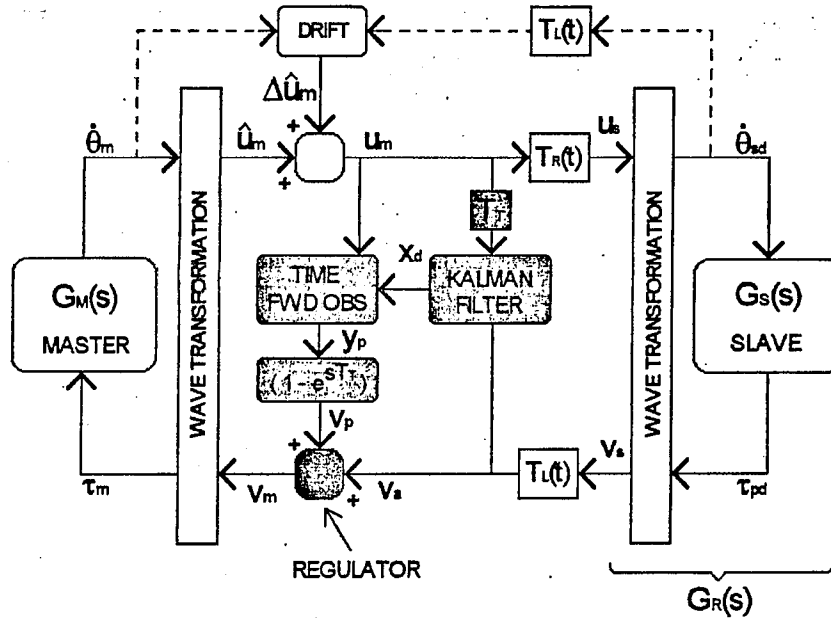


Fig. 4. Overall prediction scheme.

A. Predictor Implementation

The entire right-hand side of the system (marked $G_R(s)$ in Fig. 4) can be represented as

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu_m(t - T_R) \\ v_s(t) &= Cx(t) + Du_m(t - T_R). \end{aligned} \quad (21)$$

Using the variable $x_d(t) = x(t - T_L)$, the above expression can be written as

$$\begin{aligned} \dot{x}_d(t) &= Ax_d(t) + Bu_m(t - T_T) \\ v_a(t) &= v_s(t - T_L) = Cx_d(t) + Du_m(t - T_T). \end{aligned} \quad (22)$$

Hence, the entire right-hand side plant can be viewed as if it was driven by a control signal delayed T_T units of time and has an internal state $x_d(t)$. This state is estimated using a Kalman filter. The time forward observer now generates a predicted state vector $x_p(t)$ (corresponding to the current input $u_m(t)$), from the delayed state vector $x_d(t)$ (which corresponds to the delayed input $u_m(t - T_T)$) according to

$$x_p(t) = e^{AT_T} x_d(t) + \int_{t-T_T}^t e^{A(t-\tau)} Bu_m(\tau) d\tau \quad (23)$$

and finally the new output is computed as

$$y_p(t) = Cx_p(t) + Du_m(t). \quad (24)$$

The integral term in (23) can be computed according to the following state-space model

$$\begin{aligned} \dot{z}(t) &= Az(t) + Bu_m(t) \\ g(t) &= z(t) - e^{AT_T} z(t - T_T) \end{aligned} \quad (25)$$

where it can be shown that

$$g(t) = \int_{t-T_T}^t e^{A(t-\tau)} Bu_m(\tau) d\tau. \quad (26)$$

In this arrangement the predictor does not require any knowledge of the initial conditions and the Kalman filter will eventually converge to the correct internal state of the slave as viewed on the left side of the communication link. Since the internal state of the slave is directly effected when the slave interacts with the environment and the predictor relies on the Kalman filter to estimate the internal state of the slave, no measurements of forces exerted by the remote environment onto the slave are needed.

B. Regulation

For passivity, the condition depicted by (18) must to be met. In other words the predictor must not increase the total energy contained in the returning wave v_m . This condition can be explicitly enforced through the use of a filter, which we refer to as a regulator. First we define $v_t(t)$ as the sum of the returning wave $v_a(t)$ and the prediction $v_p(t)$

$$v_t(t) = v_a(t) + v_p(t). \quad (27)$$

The goal is to minimize the "distance-to-go" defined as

$$D_{tg}(t) = \int_0^t (v_t(\tau) - v_m(\tau)) d\tau. \quad (28)$$

For this purpose we define an energy reservoir

$$E_r(t) = \int_0^t (v_a^T(\tau)v_a(\tau) - v_m^T(\tau)v_m(\tau)) d\tau \quad (29)$$

which keeps track of the energy extracted by the regulator. The control law which computes v_m in order to drive $D_{tg}(t)$ to zero based on the energy contained in the reservoir is then given by

$$v_m(t) = \alpha (1 - e^{-\beta E_r(t)}) D_{tg}(t) \quad (30)$$

where α and β are both positive constant tuning parameters. Given that

$$1 > (1 - e^{-\beta E_r(t)}) > 0 \quad (31)$$

the output $v_m(t)$ and the distance $D_{tg}(t)$ are always of the same sign. At steady state, when the transients have decayed and there is no forced input

$$\lim_{t \rightarrow \infty} v_p(t) = \lim_{t \rightarrow \infty} v_a(t) = 0 \Rightarrow \lim_{t \rightarrow \infty} v_t(t) = 0. \quad (32)$$

Hence, under such condition, from (28)

$$\frac{d}{dt} D_{tg}(t) = -v_m(t) \quad (33)$$

given that D_{tg} and v_m are of the same sign, implies that

$$D_{tg}(t) \rightarrow 0. \quad (34)$$

If during regulation, the energy reserve approaches 0 then

$$(1 - e^{-\beta E_r(t)}) \rightarrow 0 \quad (35)$$

choking the output wave v_m , while simultaneously increasing the energy reserve. At startup it would take a little time for the reservoir to build up, the size of which is governed by β while α determines how fast $D_{tg}(t)$ decays. Choosing α and β to both be positive ensures that the energy reservoir (29) is kept positive.

IV. TIME DELAY ON THE INTERNET

So far it is assumed that the time delay is constant in both directions. However, this is not true in the case of the Internet. Therefore, it is necessary to find out what kind of delays are involved as well as some other characteristics of Internet-based communication (which for the most part depend on the network protocol used to stream data).

Currently, the two most popular protocols used to transmit data over the net are the transport control protocol (TCP/IP) and the user datagram protocol (UDP). TCP provides a point-to-point channel for applications that require reliable communication. It is a higher-level protocol that manages to robustly string together data packets, sorting them and retransmitting them as necessary to reliably transmit data. Further, TCP/IP is confirmation based. The top of Fig. 5 shows the cross Atlantic round trip time delay between Georgia Tech, Atlanta, and Georgia Tech Loraine, in Metz (France), using TCP/IP protocol over a period of 100 s. Data points are generated 10 ms apart. The experiment was carried out on a typical workday during mid afternoon. It is apparent that the delay varies substantially, ranging from a minimum value of 100 ms to as much as 3000 ms. The bottom portion of Fig. 5 shows the transmitted and received sine waves for a period of 10 s, sampled at 10 ms intervals. Although no information is lost in TCP/IP based communication, it is evident from the figure that data sampled at different points in time gets lumped together along the way and arrives simultaneously at the destination. Hence, the shape of the sine wave is not preserved. Thus making TCP/IP based communication rather unfavorable for real-time control.

UDP protocol provides communication that is not guaranteed between two applications on the network. Unlike TCP/IP which is connection based, UDP is not. Rather, it sends independent packets of data, called datagrams, from one application to another. In UDP based connections data is packed into packets

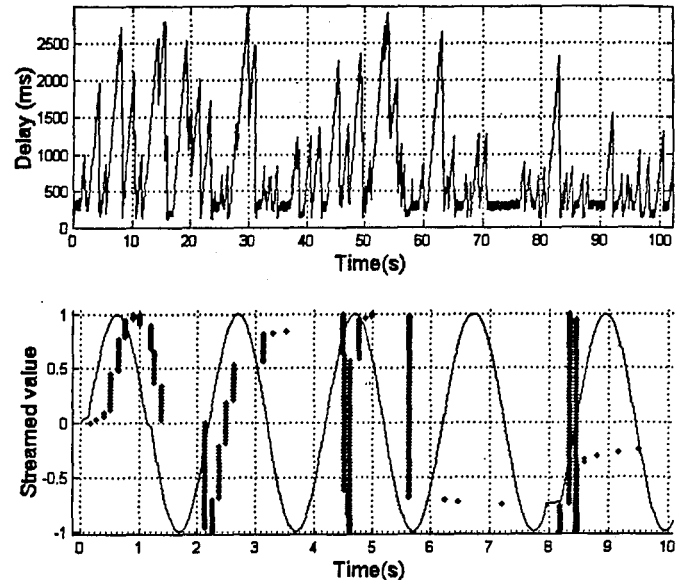


Fig. 5. Top: Cross Atlantic round trip time delay between Georgia Tech, Atlanta and Metz, France using TCP protocol. Bottom: Transmitted and received sine wave, sampled at 10 ms.

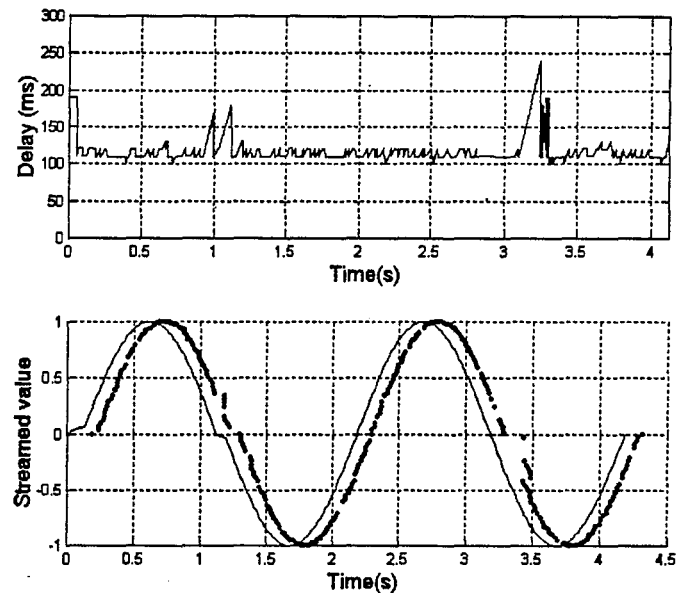


Fig. 6. Top: Cross Atlantic round trip time delay between Georgia Tech, Atlanta and Metz, France using UDP protocol. Bottom: Transmitted and received sine wave, sampled at 10 ms.

called datagrams, addressed (like an envelope) and then transmitted. Much like sending a letter through mail, UDP is not confirmation based and the order of arrival is not guaranteed. The overhead of retransmitting data is eliminated which comes at the expense of some data getting lost and not arriving at the destination at all. The top of Fig. 6 shows the cross Atlantic round trip delay between Georgia Tech, Atlanta, and Georgia Tech Loraine, in Metz (France), using UDP protocol. Notice that the fluctuations are a lot less, ranging from a minimum value of 100 ms to a maximum of 250 ms with the average being 116 ms and a standard deviation of less than 11 ms. Although only 4 s of streaming results are shown, this trend was also confirmed

over a much longer period of time. The bottom portion of Fig. 6 shows the transmitted and received sine wave over the same 4 s period. Notice this time the shape of the signal is preserved and the received wave closely tracks the transmitted wave with a lag time equaling the delay observed in the top graph.

However, notice that some data arrives out of order as can be seen around the 3.5 s mark where the top graph shows a sudden spike in the delay. A few datagrams are also observed to arrive simultaneously and some 12 to 16 % of information is lost along the way.

It should be emphasized that this experiment was conducted in January of 2000. The experiment was repeated over a year later using a C++ application under similar conditions. It was determined that although the round trip delay still has a similar lower bound, the standard deviations had decreased by over a factor of 2 and the data loss rates were below 2%. This can be attributed to two factors. First C++ was used instead of JAVA, which allowed for a smaller sample time and a higher resolution with which the round trip delay could be measured. Secondly, there was a rapid improvement in the Internet infrastructure in the U.S. and France during this time.

Given what is known about TCP/IP and UDP protocols, the protocol of choice for real-time control is UDP. This is because a consistent sample rate with lower fluctuations can be maintained with UDP. This consistency in sampling, is what makes UDP the protocol of choice.

V. STABILITY FOR A VARYING DELAY

Given that the standard deviation in the delay is rather small when streaming data using UDP protocol, we shall treat the variation in the delay as a perturbation. In other words the delay in the right moving wave can be written as

$$T_R(t) = T_R + \Delta T_R(t) \quad (36)$$

and the delay in the left moving wave can be written as

$$T_L(t) = T_L + \Delta T_L(t) \quad (37)$$

where ΔT_R and ΔT_L are perturbations in the delay. In order to aid the following analysis consider Fig. 7, which is a compact representation of the right side of the system. Here, $G_R(s)$ is the transfer function of the entire right side of the plant as previously defined in Fig. 4. The perturbation in delay causes a distortion in the wave signals u_m and v_s as they pass through the communication link. Hence, we can model the communication process as

$$u_s(t) = u_m(t - T_R) + \Delta u_m(t) \quad (38)$$

and

$$v_m(t) = v_s(t - T_L) + \Delta v_s(t). \quad (39)$$

For the system to remain passive under a varying delay, the energy returned must be less than the energy sent

$$\frac{1}{2} \int_0^t u_m^T u_m d\tau - \frac{1}{2} \int_0^t v_m^T v_m d\tau > 0. \quad (40)$$

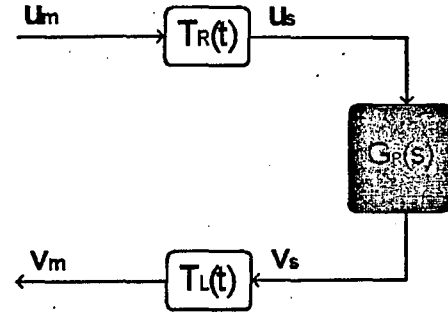


Fig. 7. Compact representation of the right side.

However, we know that the energy dissipated in the remote system is

$$E_d(t) = \frac{1}{2} \int_0^t u_s^T u_s d\tau - \frac{1}{2} \int_0^t v_s^T v_s d\tau. \quad (41)$$

Because the delay in the communication line is constantly varying, the line could compress or stretch the wave signal in such a way so as to increase or decrease the power in the outgoing wave. Hence, the energy contained in the right moving wave u_m is given by

$$\frac{1}{2} \int_0^t u_m^T u_m d\tau = \frac{1}{2} \int_0^t u_s^T u_s d\tau + E_{T_R}(t) - E_R(t) \quad (42)$$

while the energy contained in the left moving wave v_s is given by

$$\frac{1}{2} \int_0^t v_s^T v_s d\tau = \frac{1}{2} \int_0^t v_m^T v_m d\tau + E_{T_L}(t) - E_L(t). \quad (43)$$

Here, $E_{T_R}(t)$ and $E_{T_L}(t)$ are the instantaneous energies stored in the communication line and $E_R(t)$ and $E_L(t)$ are the energy increases or decreases caused by the varying delay. Using (41), (42), and (43), (40) becomes

$$E_d(t) + E_{T_R}(t) + E_{T_L}(t) - E_R(t) - E_L(t) > 0. \quad (44)$$

Given that the remote system is passive, $E_d(t)$ will grow more and more positive with time, while $E_{T_R}(t)$ and $E_{T_L}(t)$ are always positive. Hence, it can be concluded that the system is stable if the remote plant $G_R(s)$ is passive and it dissipates enough energy such that (44) is satisfied.

A. Correcting for the Position Error Due to a Varying Delay

Given that position information is encoded in the integral of the wave signal, a varying delay will introduce a nonzero steady-state error between the master and the slave. This is because the integral of the wave signals is not preserved as the signals pass through the communication medium. Niyemmer and Slotine tried to overcome this problem by transmitting the integral of the wave signals through the communication medium and then reconstructing the original signal using a specially designed filter. However, we can not use such a scheme in this arrangement as the nonlinear dynamics of the reconstruction filter can not be easily factored into the prediction scheme. Therefore, we shall opt for a different method, where a minor correction $\Delta \hat{u}_m$ is added to the right moving wave \hat{u}_m (here, the

"hat" denotes the original uncorrected signal) to ensure a zero steady-state error. Rewriting the expression similar to (14), for the position difference as viewed on the left side of the communication line

$$\Delta\theta_2(t) = \theta_m(t) - \theta_{sd}(t - T_L(t)) \quad (45)$$

and expanding yields

$$\Delta\theta_2(t) = \frac{1}{2}A_w^{-1} \int_0^t (\hat{u}_m(\tau) + v_m(\tau) - u_s(\tau - T_L(\tau)) - v_s(\tau - T_L(\tau))) d\tau. \quad (46)$$

Since we want to correct the right moving wave by adding a corrective term

$$u_m(t) = \hat{u}_m(t) + \Delta\hat{u}_m(t). \quad (47)$$

Assuming $T_L \approx T_R$, the following approximations can be made:

$$u_s(t - T_L(t)) = u_m(t - 2T_L) + \omega_1(t) \quad (48)$$

and

$$v_s(t - T_L(t)) = v_a(t) + \omega_2(t) \quad (49)$$

where ω_1 and ω_2 are assumed to be small perturbations caused by the variations in the delays. Equation (46) can now be written as

$$\begin{aligned} \Delta\theta_2 = & \frac{1}{2}A_w^{-1} \int_{t-2T_L}^t u_m(\tau) d\tau \\ & + \frac{1}{2}A_w^{-1} \int_0^t (v_m(\tau) - v_a(\tau)) d\tau \\ & - \frac{1}{2}A_w^{-1} \int_0^t \Delta\hat{u}_m(\tau) d\tau \end{aligned} \quad (50)$$

where the perturbations ω_1 and ω_2 have been absorbed into $\Delta\hat{u}_m$. Under ideal conditions $\Delta\hat{u}_m = 0$. Now define the drift error d as

$$d(t) = \text{expected } \Delta\theta_2 - \text{actual } \Delta\theta_2 \quad (51)$$

which leads to

$$\begin{aligned} d(t) = & \frac{1}{2}A_w^{-1} \int_{t-2T_L}^t u_m(\tau) \\ & + \frac{1}{2}A_w^{-1} \int_0^t (v_m(\tau) - v_a(\tau)) d\tau \\ & - [\theta_m(t) - \theta_{sd}(t - T_L(t))]. \end{aligned} \quad (52)$$

However, the drift $d(t)$ is also given by

$$d(t) = \frac{1}{2}A_w^{-1} \int_0^t \Delta\hat{u}_m(\tau) d\tau. \quad (53)$$

The goal now is to drive the drift $d(t)$ to zero. For this purpose let us define a second energy reservoir which keeps track of how much energy is dissipated in the slave system, as

$$E_d(t) = \int_0^t (\hat{u}_m^T \hat{u}_m - v_m^T v_m) d\tau. \quad (54)$$

Then, let the correction term be computed as

$$\Delta\hat{u}_m = -\gamma (1 - e^{-\delta E_d(t)}) A_w d(t) \quad (55)$$

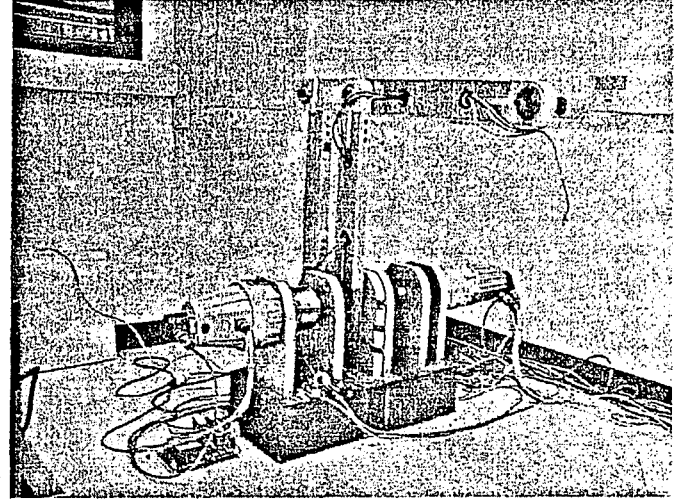


Fig. 8. The 2-DOF parallel link haptic robot, acting as the master.

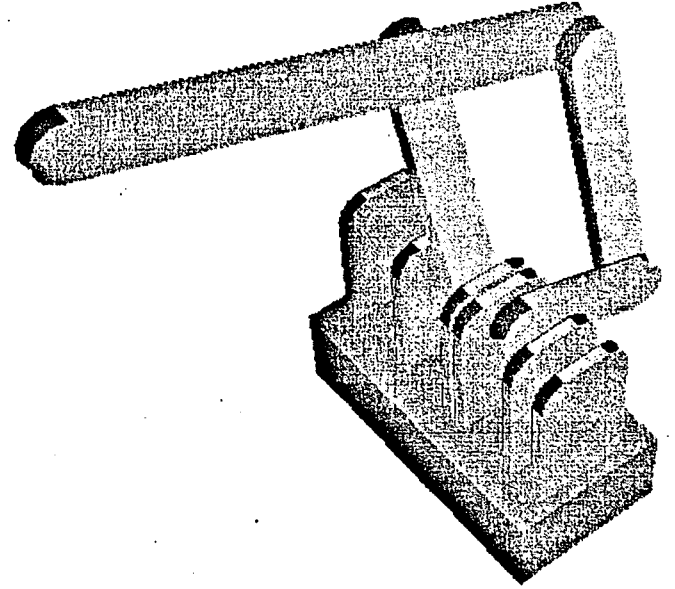


Fig. 9. The 2-DOF parallel link simulated robot, acting as the slave.

where both γ and δ are positive constants. From (53)

$$\frac{d}{dt}d(t) = \frac{1}{2}A_w^{-1} \Delta\hat{u}_m. \quad (56)$$

Plugging (55) into the above expression yields

$$\frac{d}{dt}d(t) = -\frac{1}{2}\gamma (1 - e^{-\delta E_d(t)}) d(t). \quad (57)$$

Given that the remote system is dissipative, implies that $E_d(t) \geq 0$ and

$$0 < (1 - e^{-\delta E_d(t)}) < 1. \quad (58)$$

This implies that $d(t)$ and its derivative are always of the opposite signs, hence

$$d(t) \rightarrow 0. \quad (59)$$

Note, that the above control law is very similar to that used in the regulator, however, the correction term computed is very

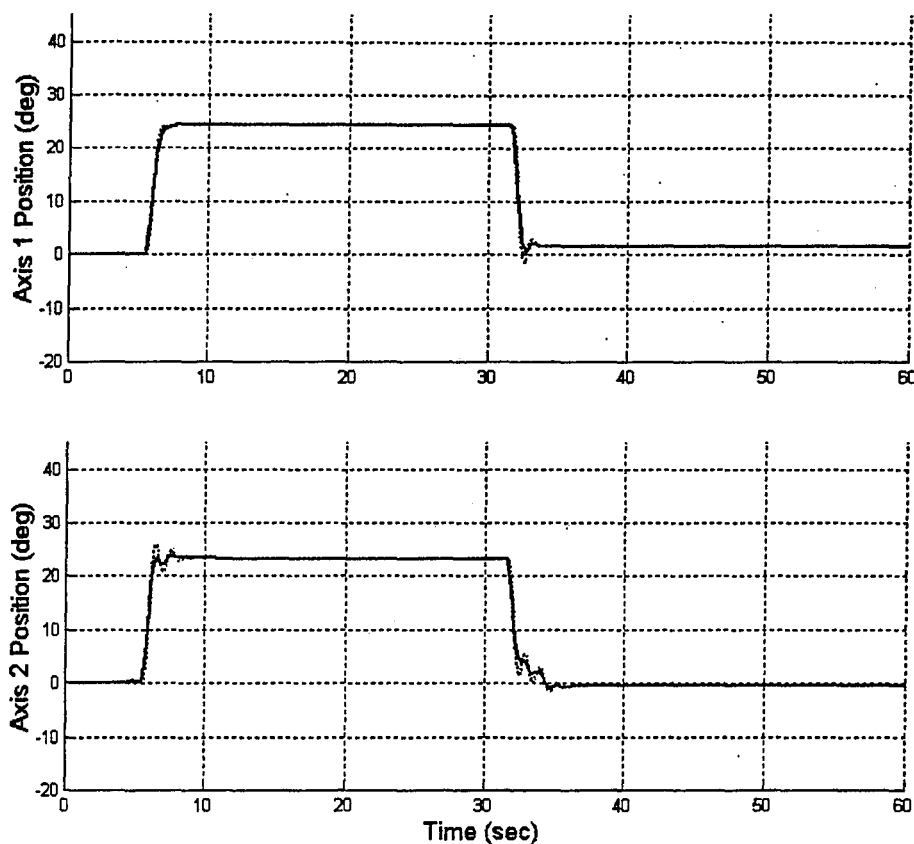


Fig. 10. Normal teleoperation (no delay). Solid lines represent the master, dashed lines represent the slave.

small given that the variations in the delay are assumed to be small. It should be noted that in this scheme the energy dissipated E_d could temporarily become negative (due to the environment acting on the slave) or large over time. Hence, the software should bound the value of E_d between 0 and some small positive number. The overall control scheme is shown in Fig. 4.

VI. EXPERIMENTAL RESULTS

The experimental setup consists of a master manipulator bilaterally coupled to a kinematically similar device over the Internet. The master manipulator is a 2-DOF parallel link haptic robot (custom built at Georgia Tech). This system is shown in Fig. 8. It is powered by two geared dc motors at the base and is controlled by an Intel based PC. The user can hold the tip of the manipulator, where strain gauges measure the applied forces and can back drive the system.

The slave manipulator is a real-time simulation of the exact same device running on a separate computer. The output of the simulator is viewed via a graphical user interface (GUI) written in OpenGL, shown in Fig. 9.

The control loop and the slave simulation run at a rate of 1 kHz. This means that all the required states are also streamed through the network at a rate of 1 kHz. Under limited bandwidth, the system can be made to run at 250 Hz, to prevent data loss during streaming. Both systems are connected to the Internet using an Ethernet adapter, hence, bandwidth does not seem to

be an issue. Since the master manipulator and the real-time simulation of the slave are present in the same room, time delay of the Internet is introduced by rebounding the streamed control signal from a remote site (in Japan).

Fig. 10 shows plots for the joint angles under regular teleoperation with no delay, but the control signal is still streamed through the Internet. In this experimental run the tip of the master manipulator is moved forward and up and is kept there for approximately 20–30 s. Following this the master manipulator is moved back to the home position.

Fig. 11 shows the results for a wave-based teleoperation with no prediction for an average round trip delay of over 364 ms. Without some form of delay compensation this control loop is highly unstable and is not shown. In this experiment both the master and slave systems were setup in Atlanta, but the control signal was rebounded from Tokyo, Japan. *Making the closed-loop distance larger than the circumference of Earth.* Again the user moves the master manipulator in a similar fashion. Notice that the slave system experiences a lot of oscillations. If during the experiment the user did not hold on firmly, the master manipulator too would oscillate. Finally Fig. 12 shows the same experiment repeated using the predictive wave-based technique discussed in this paper under the same exact conditions. Notice the significant improvement over regular wave-based teleoperation. The settling time is significantly improved and the user experiences almost no reflections (oscillations). Furthermore, it was seen that the steady-state error between the master and slave systems was less than 0.001 rads.

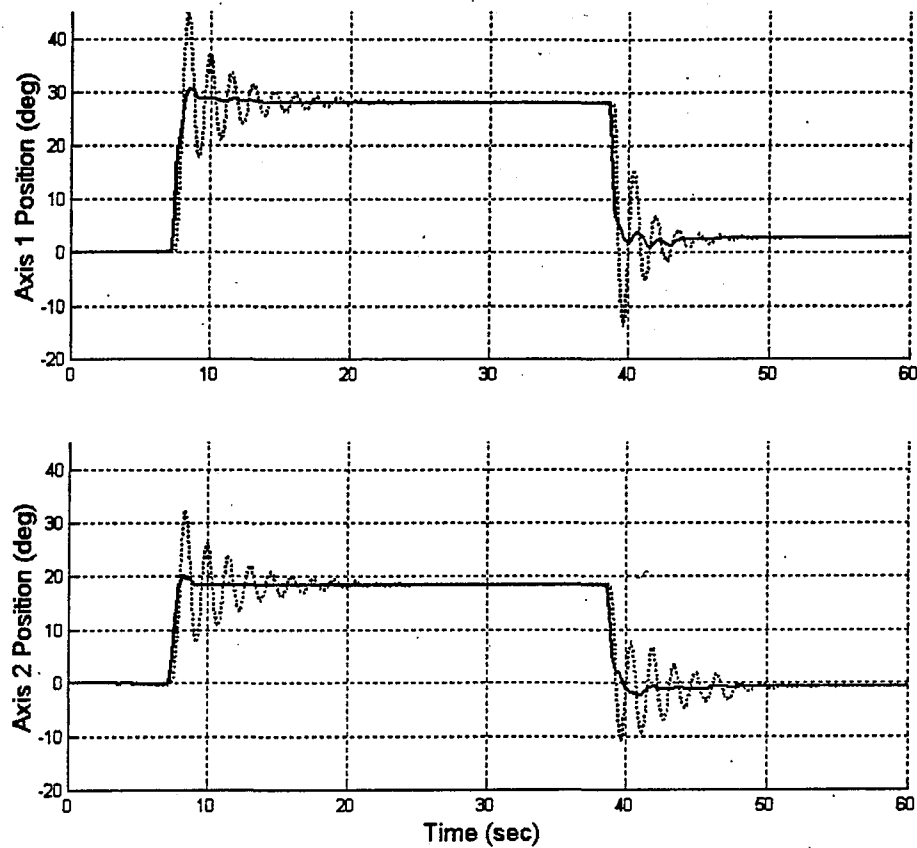


Fig. 11. Wave-based teleoperation (control signal rebounded from Tokyo, Japan). Solid lines represent the master, dashed lines represent the slave.

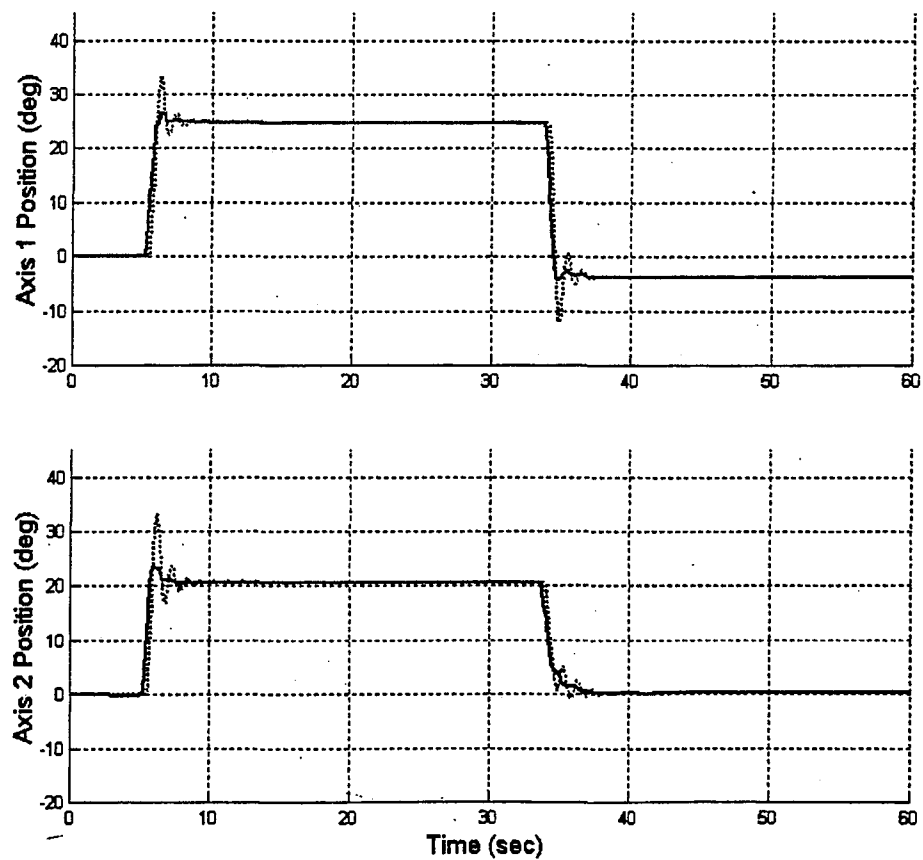


Fig. 12. Wave-based teleoperation with prediction (control signal rebounded from Tokyo, Japan). Solid lines represent the master, dashed lines represent the slave.

VII. CONCLUSION

It can be argued that this technique is very robust to model uncertainties because passivity is explicitly enforced by the regulator. If there is a model mismatch, predicting in the wave domain while regulating the return power-flow ensures that passivity is always maintained. Predicting outside the wave domain would still provide good performance, however, in such a case power-flow can not be monitored in a straightforward manner, hence, a mismatched plant model could destabilize the system.

It was noticed during experimental trials that this system is most effective when the delay in the loop is significantly larger than the time constant of the system. However, for smaller delays this scheme still shows an improvement over regular wave-based teleoperation.

During our experimental trials a virtual wall was also programmed in the slave simulation. It was noticed that if the master manipulator is driven past the wall and then released, the slave manipulator is able to pull back the tip of the master past the wall boundary. This is because the net effect of the predictor over the entire run is zero. Meaning that whatever is added at time t , is subtracted at time $t + T_T$. Hence, the predictor only helps to improve the settling time of the slave and master manipulators under free motion, while still allowing the user to feel the remote environment at steady state.

ACKNOWLEDGMENT

The help of Professor F. Matsuno and K. Ito at the Tokyo Institute of Technology with conducting live experiments over the Internet between Atlanta and Tokyo is greatly appreciated.

REFERENCES

- [1] R. J. Anderson and M. W. Spong, "Bilateral control of teleoperators with time delay," *IEEE Trans. Automat. Contr.*, vol. 34, pp. 494-501, May 1989.
- [2] R. J. Anderson, "SMART Class Notes," Intelligent Systems & Robotics Center, Sandia National Laboratories, Livermore, CA, 1994.
- [3] K. Brady, "Time-delayed control of telerobotic manipulators," Ph.D. dissertation, Dept. Syst. Sci. Math., Univ. Washington, Seattle, 1997.
- [4] K. Brady and T.-J. Tarn, "Internet-based remote teleoperation," in *Proc. IEEE Int. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 65-70.
- [5] C. A. Desoer and M. Vidyasagar, *Feedback Systems: Input-Output Properties*, NY: Academic, 1975.
- [6] S. Munir, "Internet-based teleoperation," Ph.D. dissertation, George W. Woodruff Sch. Mech. Eng., Georgia Inst. Technol., Atlanta, 2001.
- [7] S. Munir and W. Book, "Wave-based teleoperation with prediction," in *Proc. American Controls Conf.*, Arlington, VA, June 2001, pp. 4605-4611.
- [8] G. Niemeyer and J.-J. E. Slotine, "Stable adaptive teleoperation," *IEEE Trans. Automat. Contr.*, vol. 36, pp. 152-162, Jan. 1991.
- [9] G. Niemeyer, "Using wave variables in time delayed force reflecting teleoperation," Ph.D. dissertation, Dept. Aeronautics Astronautics, Mass. Inst. Technol., Cambridge, 1996.

- [10] G. Niemeyer and J.-J. E. Slotine, "Designing force reflecting teleoperators with large time delays to appear as virtual tools," in *Proc. IEEE Int. Conf. Robotics and Automation*, Albuquerque, NM, Apr. 1997, pp. 2212-2218.
- [11] —, "Using wave variables for system analysis and robot control," in *IEEE Int. Conf. Robotics and Automation*, Albuquerque, NM, Apr. 1997, pp. 1619-1625.
- [12] —, "Toward force-reflecting teleoperation over the internet," in *Proc. IEEE Int. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 1909-1915.
- [13] R. Oboe and P. Fiorini, "A design and control environment for internet-based telerobotics," *Int. J. Robot. Res.*, vol. 17, no. 4, pp. 433-499, Apr. 1998.
- [14] Z. Palmor, "Stability properties of smith dead-time compensator controllers," *Int. J. Control*, vol. 32, no. 6, pp. 937-949, 1980.
- [15] O. J. M. Smith, "Closer control of loops with dead time," *Chem. Eng. Prog.*, vol. 53, no. 5, pp. 217-219, May 1957.
- [16] T.-J. Tarn and K. Brady, "A framework for the control of time-delayed telerobotic systems," in *Proc. IFAC Robot Control*, Nantes, France, 1997, pp. 599-604.
- [17] Y. Yokokohji, T. Imaida, and T. Yoshikawa, "Bilateral teleoperation under time-varying communication delay," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, vol. 3, Oct. 1999, pp. 1854-1859.



Saghir Munir was born in Limassol, Cyprus, in 1971. He received the B.S. and M.S. degrees from the University of California, Irvine, and the Ph.D. degree from the Georgia Institute of Technology, Atlanta, in 1996, 1997, and 2001, respectively. His Ph.D. research focused on the control of time-delayed robotic systems over packet switched networks.

Currently, he is with Intel Corporation, Santa Clara, CA, where he writes automation and image processing software for photolithography applica-

tions. His research interests include rigid body dynamics, controls and software design.

Dr. Munir is the recipient of the Holmes Fellowship and the Intel Fellowship.



Wayne J. Book (M'74-SM'94-F'96) was born in San Angelo, TX, in 1946. He received the B.S.M.E. degree from the University of Texas at Austin, in 1969, and the M.S. and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1971 and 1974, all in mechanical engineering.

He is the HUSCO/Ramirez Distinguished Professor in fluid power and motion control in the George W. Woodruff School of Mechanical Engineering at the Georgia Institute of Technology, Atlanta, where he teaches and researches in the

areas of system dynamics, controls, robotics, design, and fluid power systems. He published extensively in these areas and holds several patents on robotics and related inventions. He is a cofounder of CAMotion, Inc., a producer of advanced motion control software.

Dr. Book received the Georgia Institute of Technology's award for Outstanding Faculty Leadership for the development of graduate research assistants. He is a Fellow of the American Society of Mechanical Engineers (ASME).