Proceedings of the 1998 IEEE
International Conference on Robotics & Automation
Leuven, Belgium • May 1998

Internet-Based Remote Teleoperation *

Kevin Brady

System Engineering and Analysis Group
MIT Lincoln Laboratory
244 Wood Street
Lexington, MA 02173-9108, USA

Tzyh-Jong Tarn

Systems Science and Mathematics Dept.
Washington University
1 Brookings Avenue
St. Louis, MO 63130, USA

Abstract

A new method for controlling telerobots over vast distances, where communication propagation delays exist, is presented. Such delays are potentially destabilizing, and certainly degrade the human teleoperator's intuition and performance. Applications include internet-based robotic systems, as well as underwater and space-based systems. A canonical state space formulation is presented, taking into account the time-varying non-deterministic nature of the control and observation delays. A model of the delay characteristics for the communication medium is also derived. Using the state space framework a general purpose supervisory architecture is developed, allowing the projection of human "intelligence" to the remote environment via the telerobot. Dynamics of the robotic system, as well as the delay characteristics of the communication medium, become part of the design process. The design criteria of transparency, generality, and safety have been met and successfully tested in an experimental setup between Albuquerque, New Mexico and Washington University's Center for Robotics and Automation.

1 Introduction

The remote control of telerobotic manipulators has gained considerable attention in recent years. Issues concerning communication channels, communication propagation delay, bandwidth limitations, and telepresence have all been dealt with to varying degree. In particular, great interest has been generated by the ubiquitous internet as a viable medium for remote systems. Most remote control architectures fall into one of three approaches: predictive displays/control, bilateral control, and teleprogramming.

Book et al [3] have used the internet as a high-level supervisory controller. The remote operator has actual and simulated work environments provided graphically via the internet. The return channel is limited to verbal instructions that are actually enacted by the operators local to the experiment.

Rovetta et al [8] used an eclectic mix of communication media for performing Telesurgery. The idea is to use a local surgeon to perform the more general aspects of the surgery, while a remote specialist takes over for the more specific parts of the surgery. The communication media include a modem for the transfer of the robot commands, a satellite channel for the image transmission, and the internet for documenting the experiment and exchanging written communications. The remote surgeon's commands were entered via a keyboard during the July 1993 experiment. The commands 'telecontrolled' the robot's displacements.

Anderson [1] uses the internet as the medium for his SMART (Sequential Modular Architecture for Robotics and Telerobotics) architecture. This bilateral modular controller is based on earlier work guaranteeing the passivity of the architecture for a given delay.

Wakita et al [10] suggest a combination of intelligent visual monitoring and a canonical set of high level commands as a means of "Intelligently Monitoring" remote telerobots. In an August 1995 experiment they performed an internet based experiment between the ETL lab at Tsukuba, Japan and the Jet Propulsion Laboratory (JPL) in Los Angeles, USA. They note the importance of bandwidth in design considerations. Due to this limitation they note that higher level commands (requiring less bandwidth) is ideal, though the abstraction of the command must be low enough to allow the easy intervention of the teleoperator into the telerobot's work.

Oboe and Fiorini [7] also suggests a bilateral ar-

*Research Partially supported under Oak Ridge National Laboratories Contract No. SU662-49 and by Sandia National Laboratories Contract No. AC-3752-C.

chitecture for remote teleoperation. They discuss and take into account the time-varying nature of the internet and the nonlinear dynamics of the telerobot. An experiment is presented comprising of a hardware based master, and virtual software slave manipulator

Bejczy et al [2] suggest use of the "phantom robot" for remote teleoperation. The idea is to superimpose a predictive robotic model over the live delayed video feed from the remote system. Human "intelligence" is used to bridge any gaps in knowledge or model discrepancies.

Funda and Paul [5] suggest the concept of "Teleprogramming" for dealing with controlling time-delayed telerobots. The approach is based on creating a set of symbolic instructions for accomodating the discrepancy between the world and local model. The telerobot executes instructions until their completion, unless an unexpected occurrence such as a collision intervenes. In such a case the telerobot awaits further instructions from the teleoperator who has a force-reflective / graphical model of the remote workcell.

Conway et al [4] present methods for "teleautonomous technology" or "teleautomation". Their motivation is to project intelligence over a distance, blending autonomy with human intelligence. Various teleautomation "tools" are introduced, most notable is the time-clutch. The time-clutch gives the teleoperator the ability to engage and disengage the remote manipulator temporally, giving greater flexibility and efficiency in completing some task.

No discussion of remote teleoperation is complete without discussion of the ambitious project by the German Aerospace Research Establishment — ROTEX [6]. This arm flew on the Space Shuttle Columbia in 1993 where it tested the viability of using ground based supervision for space-based tasks. ROTEX was an eclectic approach to remote teleoperation. It used a n-step ahead discrete Kalman filter for prediction. Active compliance was used to provide desirable dynamic response. Telesensorprogramming is introduced for facilitating human interaction with the arm.

In Section 2 the time-varying delays and bandwidth limitations for remote systems are described. Using these descriptions a state space model for controlling such systems is presented. Section 3 builds an architecture around the state space model from Section 2. This architecture relies on the teleoperator supplying a trajectory to the telerobot which executes this trajectory semi-autonomously. The teleoperator only intervenes in cases of unexpected circumstances. Section 4 presents the Experimental Results of this architecture, and Section 5 provides concluding remarks and directions for future research.

2 State Space Formulation

2.1 Delay and Bandwidth Limitations

The goal of this subsection is to describe the nature of the propagation delay that is incurred, describe its relationship to the bandwidth, and come up with a delayed model for the state space system. The unidirectional delay is designated $h(t)$. It is only for illustrative purposes and may denote the delay in either the forward or backward direction. The delay $h(t)$ may be broken down into three components as follows:

$$h(t) \quad = \quad h_n + \overline{h}_d(t) + h_b(t) \qquad (1)$$

where:

- h_n is the nominal propagation delay. It represents the time that it takes for the signal to physically propagate without disturbance from its source to its destination across the communication medium. Its value may be experimentally determined and is non time-varying.

- h_d is the disturbance delay. It represents the deviation from the expected delay that results from unknown disturbances or even loss of information. h_d is non-deterministic and time-varying, though its characteristics (ie. variance) may be known a priori.

- $\overline{h}_d$ is a step function that is based on the function h_d. Since the data exchange is discrete the only time that the disturbance delay is relevant is when the information actually propagates. This is reflected in the function $\overline{h}_d$.

- h_b is the bandwidth delay. Since information is exchanged at finite rate b across the communication medium a corresponding time-varying delay will be incurred whose form is known as seen in Figure 1. The delay h_b is a sawtooth function and is bounded as follows:

$$\overline{h}_b \quad = \quad \frac{1}{b} \geq h_b(t) \geq 0$$

The choice of b is the first important design consideration of our system. A large value of b enables a greater exchange of information, as well as reducing the effective delay of the system. On the other hand, there may be practical limitations to just how large a value may be chosen. Choosing b too large may overload the communication system, resulting in lost data and possibly an untimely shutdown.

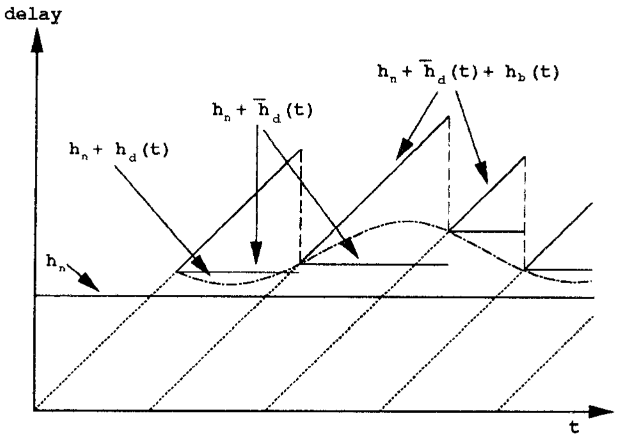

Figure 1: Delay Components

Figure 1 shows the breakdown of the uni-directional delay into its constituent parts. Note that the graph is cumulative by nature. The delay in the observer direction (information flowing from the telerobot side to the teleoperator side) is handled differently than the delay in the controller direction (information flowing from the teleoperator side to the telerobot side). The reason for the discrepancy is rather simple:

- In the observer direction it simply makes sense to apply received information to the observer model as soon as it is received.
- In the controller direction it is important for there to be a static delay between the time the control is sent from the teleoperator to the time it is applied by the telerobot. The static delay provides a consistency in scheduling so that a causal nature is established between teleoperator commands and the predictive display resulting in greater telepresence. This delay offset is designated h_c. In other words, if the control information arrives early it should not be applied until h_c seconds after it was generated by the teleoperator.

Note that the value of the delay in Eq. (1) is different depending on if it is in the observer or controller direction. As stated in the above paragraph, it is practical to apply the observed information as it comes in. Mathematically, this means that the delay in the observer direction (h_o) is defined as follows:

$$h_o(t) \quad = \quad h(t)$$

The delay in the control direction is evaluated slightly differently. First, a control delay h_c must be determined as a design consideration. As mentioned earlier, h_c's value represents the ideal difference in time between when a control is applied at the teleoperator and at the telerobot side of the system. It makes sense to choose its value such that it is at least as large as the maximum value of the deterministic part of the delay model:

$$h_c \quad \geq \quad h_n + \bar{h}_b$$

Once the choice of h_c is made a function $\Delta h_c(t)$ can be used to model when $h(t)$ exceeds h_c:

$$\Delta h_c(t) \quad = \quad \begin{cases} h(t) - h_c & h(t) > h_c \\ 0 & h(t) \leq h_c \end{cases}$$

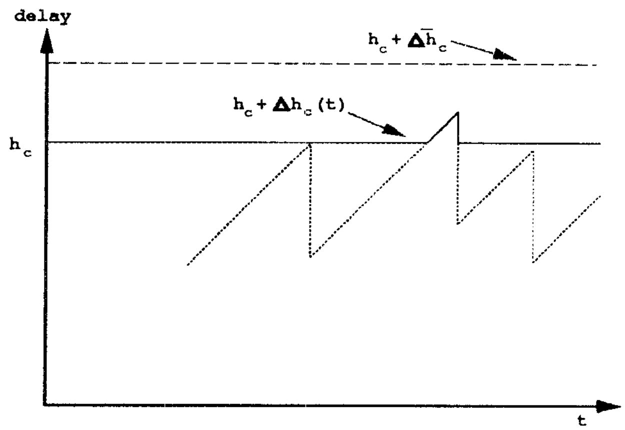

Figure 2: Composite Delay

$h_c + \Delta h_c(t)$ now represents the time difference between when a control is generated by the teleoperator and enacted by the telerobot. The value of $\Delta h_c(t)$ is bounded by $\Delta\bar{h}_c$. If this were not the case the system would effectively be open-loop. The solid line in Figure 2 represents this delay, while the dotted line represents the cumulative delay from Figure 1.

2.2 State Space Derivation

The specific details of the state space model have been presented by the authors in the literature before [9] and will only be presented briefly here. The dynamic model of a robot is governed by a nonlinear dynamic equation. In the following approach nonlinear feedback linearization (NFL) decoupling has been used locally by the telerobot so that the state prediction may be carried out on a set of linear equations. The literature is rich with examples of the use of NFL and the reader is directed to [11] for further direction.

The representation for a time-delayed system consists of the following equations:

$$\dot{x}(t) = Ax(t) + B_1 u_1(t) + B_2 u_2(t - h_c - \Delta h_c(t))$$
$$y(t) = \sum_{j=1}^{q} C_j x(t - h_{o_j}(t))$$

where $x(t)$ is the joint space vector of dimension n and $y(t)$ is the task space vector of dimension m. It is assumed that $u_1(t)$, $u_2(t)$, and $y(t)$ are measureable, though $x(t)$ is not. Matrices A, B_1, B_2, C are time invariant and of appropriate dimension. The delay components $(\Delta h_c(t), h_{o_j}(t))$ are non-deterministic, time-varying, non-negative, and bounded $\forall j, \forall t$:

$$0 \le h_{o_j}(t) \le \overline{h}_{o_j}$$
$$0 \le \Delta h_c(t) \le \Delta \overline{h}_c$$

The state space system represented by the state equations have the following initial conditions:

$$x(t) = \phi(t) \qquad \forall t \in [-h_c, 0]$$
$$u_2(t) = \psi_2(t) \qquad \forall t \in [-h_c, 0]$$

Note that the state feedback is of the following form:

$$\overline{u}_1(t + h_c|t) = Fx(t + h_c|t) + Ld(t + h_c|t)$$
$$u_2(t) = Ld_S(t)$$

where $d()$ is the nominal desired trajectory and $d_S()$ is the supervisory augmentation to this nominal trajectory. Note, therefore, that $\overline{u}_1$ is generated locally whereas u_2 is supplied by the teleoperator in real-time.

Define the augmented variable $\overline{x}(t + h_c|t) \approx x(t + h_c), \overline{y}(t)$ as follows:

$$\overline{x}(t + h_c|t) = e^{Ah_c}[x(t) + \int_t^{t+h_c} e^{A(t-\theta)} \bullet [B_1\overline{u}_1(\theta|t) + B_2 u_2(\theta - h_c)]d\theta]$$
$$\overline{y}(t) = y(t) + \sum_{j=1}^{n} C_j e^{-Ah_{o_j}(t)} \int_{t-h_{o_j}(t)}^{t+h_c} e^{A(t-\theta)} \bullet [B_1\overline{u}_1(\theta|t) + B_2 u_2(\theta - h_c)]d\theta$$

resulting in the following canonical system:

$$\dot{\overline{x}}(t + h_c|t) = A\overline{x}(t + h_c|t) + B_1\overline{u}_1(t + h_c|t) + B_2 u_2(t) + Qv(t)$$
$$\overline{y}(t) = \overline{C}(t)\overline{x}(t + h_c|t)$$

where:

$$\overline{C}(t) = \sum_{j=1}^{q} C_j e^{-A(h_c + h_{o_j}(t))}$$
$$Q = e^{Ah_c}B_2$$
$$v(t) = u_2(t - h_c - \Delta h_c(t)) - u_2(t - h_c)$$

and the usual assumptions for complete controllability and observability are assumed. An observer of the augmented variable $\overline{x}(t + h_c|t)$ is introduced as:

$$\dot{w}(t + h_c|t) = D(t)w(t + h_c|t) + B_1\overline{u}_1(t + h_c|t) + B_2 u_2(t) + E(t)\overline{y}(t)$$

Finally, a weighting between the observer $w()$ and the output of a dynamic simulator $r()$ is combined linearly:

$$z(t + h_c|t) = \alpha(t)w(t + h_c|t) + \beta(t)r(t + h_c|t)$$

to obtain the state index $z()$ for the predictive display and task planner as explained in the following section.

3 Architecture

Figure 3 exhibits the underlying structure of the remote telerobotic architecture presented in this paper. A trajectory is submitted to the planner for execution by the telerobot. Execution proceeds normally, except where intervening supervisory commands are provided by the human teleoperator. The difficulty lies in the fact that there is a random, time-varying communication propagation delay between the teleoperator and telerobot. In most applications synchonicity in time is a crucial problem and is dealt with here through use of an Event Based Planner [11]. It is noted that Event Based Planning, while integral to the architecture, is not necessary for understanding the delay architecture.

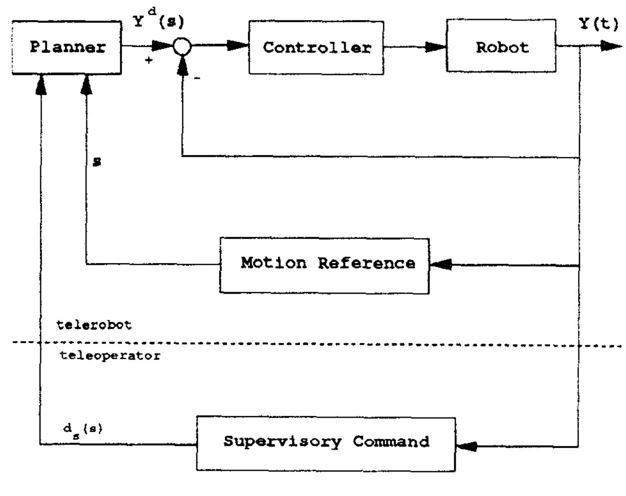

Figure 3: Supervisory Control

Figure 4 provides more detail on the delay architecture idealized in Figure 3. The results of a state predictor are combined with the results of a dynamic simulator. The resulting state $z()$ is used for providing the predicted state of the telerobot in the simulated TGRIP workspace. The Human Interface also contains a visual feed (delayed) from the remote workspace, as well as a joystick input to generate an appropriate supervisory command $d_S()$. Note that the nominal trajectory $d()$ is always generated locally. It is only the supervisory command that is generated, sent, and implemented in a delayed fashion.

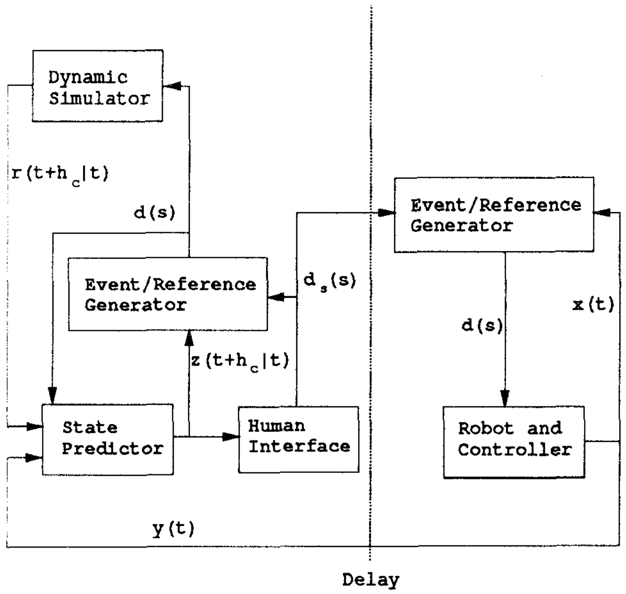

Figure 4: Time-Delayed Architecture

4 Experimental Results

During a plenary address of the 1997 IEEE International Conference on Robotics and Automation in Albuquerque, NM a live demonstration of this remote architecture was performed. The teleoperator in Albuquerque controlled a PUMA robot located in Washington University's Center for Robotics and Automation St. Louis, MO (approximately 1500 kilometers distant). Communication was at 2 Hz via a UDP connection. An audio and visual interface using SGI's Inperson was augmented by a predictive interface using Deneb's TGRIP robotic workspace simulator.

The experiment begins with the teleoperator choosing an appropriate data exchange rate and static delay (h_c was 1.5 seconds for the experiment). Next, the teleoperator chooses a set point for the robot to move to. The telerobot ran autonomously in the remote environment while the teleoperator observed over the visual (delayed) and predictive interfaces. The predictive interface provided the teleoperator with enough notice of unforseen circumstances (i.e. collisions) to take evasive actions. Upon deciding that intervention was needed the teleoperator used a joystick interface to semi-autonomously work with the remote telerobot to avoid an object. Once the teleoperator released the joystick the telerobot returned to the nominal trajectory and completed the task at hand.

Figure 5 illustrates the various positional rms errors using the following communication schemes to illustrate the efficacy of the method used in this paper. A move / avoid / recover operation is used for this illustration.

- Use the time-forward observer on the teleoperator side. Supervisory commands are implemented h_c seconds after being sent by the teleoperator.
- A Just in Time (JIT) communication paradigm, where supervisory commands are implemented as they arrive at the robot, rather than waiting h_c seconds.
- Same as the JIT paradigm, but without prediction.

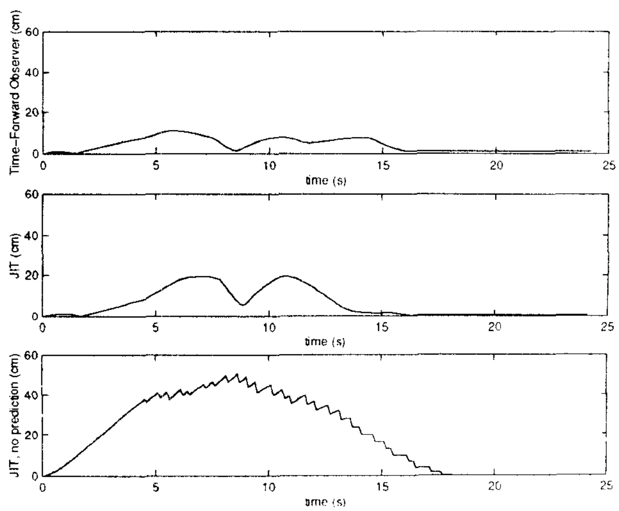

Figure 5: Avoid positional errors

It is obvious that without prediction it is difficult to obtain a reliable estimation for the state of the remote telerobot. Degradation also results in the second case due to the irregularity in arrival of the supervisory commands. It should be noted that using JIT is a reasonable alternative in designs where information must be supplied as soon as possible, rather than being queued as in this paper. The tradeoff, though, is a degradation in expected state as seen in Figure 5.

5 Conclusions

This paper has presented a promising architecture for controlling remote telerobots. The goals of safety, efficiency, telepresence, and transparency have been met. Its effectiveness has been tested, including a live demonstration during a Plenary address at the 1997 IEEE Conference on Robotics and Automation in Albuquerque, New Mexico. Such an architecture has ready application to remote controlled space-based and underwater robots. Additionally, it is highly relevant to the nascent field of internet-based control.

There are a number of contributions presented in this paper. First, a description of the delays inherent in communication channels is discussed, and its relationship with bandwidth is analyzed. Secondly, a comprehensive state space model is presented taking into account the time-varying nature of the delay. Finally, a 'human in the loop' architecture based on the state space model is illustrated. The architecture is flexible enough to allow the robot to execute commands autonomously, while retaining the ability for the teleoperator to intervene in certain circumstances. The velocity, acceleration, and force feedback of the robotic system are not limited due to the delayed nature of the communication channel, but by the physical limits of the system.

The ideal of this research area depends on the virtual immersion of a human operator in the remote workcell. Future research, therefore, depends on the development of more sophisticated vision systems to extract the relevant data from the environment and provide it to the teleoperator. Haptic and force reflective data should also be presented to augment these means.

References

[1] Robert J. Anderson, "SMART: A Modular Control Architecture for Telerobotics", *IEEE Robotics and Automation Society Magazine*, March 1995.

[2] Antal K. Bejczy, Won S. Kim, and Steven C. Venema, "The Phantom Robot: Predictive Displays for Teleoperation with Time Delay", *IEEE International Conference on Robotics and Automation Proceedings*, Cincinatti, OH, pages 546-551, May 1990.

[3] Wayne J. Book, Hobson Lane, Lonnie J. Love, David P. Magee, and Klaus Obergfell, "A Novel Teleoperated Long-Reach Manipulator Testbed and its Remote Capabilities via the Internet" *IEEE International Conference on Robotics and Automation Proceedings*, pages 1036-1041, April 1996.

[4] Lynn Conway, Richard A. Bolz, and Michael W. Walker, "Teleautonomous Systems: Projecting and Coordinating Intelligent Action at a Distance", *IEEE Transactions on Robotics and Automation*, Vol. 6, No. 2, pages 146-158, April 1990.

[5] Janez Funda and Richard P. Paul, "Remote Control of a Robotic System by Teleprogramming", *IEEE International Conference on Robotics and Automation*, Sacramento, CA, April 1991.

[6] Gerd Hirzinger, Bernhard Brunner, Johannes Dietrich, and Johann Heindl, "Sensor-Based Space Robotics - ROTEX and its Telerobotic Features", *IEEE Transactions on Robotics and Automation*, vol. 9, no. 5, pages 649-663, October 1993.

[7] Roberto Oboe and Paolo Fiorini, "A Design and Control Environment for Internet-Based Telerobotics", *International Journal of Robotics Research* , Spring 1998.

[8] Alberto Rovetta, Francesca Cosmi, and Lorenzo Molinari Tosatti, "Teleoperator Response in a Touch Task with Different Display Conditions", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 25, No. 5, pages 878-881, May 1995.

[9] Tzyh-Jong Tarn and Kevin J. Brady, "A Framework for the Control o f Time-Delayed Telerobotic Systems", *SYROCO Proceedings*, Nantes, France, September 3-5, 1997 .

[10] Y. Wakita, S. Hirai, K. Machida, K. Ogimoto, T. Ikoto, P. Backes, and S. Peters, "Applications of Intelligent Monitoring for Super Long Distance Teleoperation", *IROS Proceedings*, Osaka, Japan, November 1996.

[11] Ning Xi, Tzyh-Jong Tarn, and Antal K. Bejczy, "Intelligent Planning and Control for Multi-Robot Coordination — An Event-Based Approach", *IEEE Transactions on Robotics and Automation*, vol. 12, no. 3, pages 439-452, June 1996.