Proceedings of the 2003 IEEE
International Conference on Robotics & Automation
Taipei, Taiwan, September 14-19, 2003

Virtual Fixture Architectures for Telemanipulation

Jake J. Abbott and Allison M. Okamura
Department of Mechanical Engineering
The Johns Hopkins University, Baltimore, MD 21218
{jake.abbott, aokamura}@jhu.edu

***Abstract*— A forbidden-region virtual fixture (FRVF) is a computer-generated constraint that displays position or force limitations to a robot manipulator or operator, in order to prevent motion into forbidden regions of the workspace. We compare nine FRVFs on each of four common telemanipulator control architectures: position forward, position exchange, position forward/force feedback, and position exchange/force feedback. A one-degree-of-freedom telemanipulation system was used in an experiment designed to simulate users working near a known forbidden region. The metrics of tracking, safety, and submittance were used to analyze the performance of the system with six different users. The results indicate that different FRVF architectures perform best for each of the three metrics. No single FRVF scheme is the best over all metrics, so selection of an FRVF architecture should be an application-dependent weighting of the three metrics. Across all control architectures, the results indicate that a very strong FRVF at the slave device in combination with no FRVF at the master device leads to poor telepresence.**

I. INTRODUCTION

Telemanipulation is the direct human control of a robotic manipulator, where the operator and the manipulator are at different locations. It usually refers to a master/slave system, where the user operates a robotic manipulandum that is similar to the slave manipulator, and the slave emulates the behavior of the master. Telemanipulation is used in cases where the movements or forces of the user must be amplified or attenuated at the slave, and in situations where it is impractical or unsafe for a user to be at the same location of the slave.

The performance of a telemanipulation system is typically judged by three criteria: stability, tracking, and transparency. As with any control system, stability is a fundamental property that is essential for any viable system. Ideally, a telemanipulator would remain stable regardless of how the human operator behaves, the properties of the slave's environment, and noise and modeling errors in the system. "Tracking" refers to the geometric correspondence between the master and slave devices. Good tracking is needed to translate movement at the master device into identical movement at the slave device. "Transparency" is traditionally a measure of how well the impedance felt at the slave is reflected to the master. "Impedance" refers to the relationship between position (and its time derivatives) and force. A telemanipulator may have either good tracking or good transparency without necessarily having both. If a system does have both good tracking and transparency, it is sometimes described as creating "telepresence," meaning that movements and forces experienced at the master and slave devices are identical.

One focus of current research in telemanipulation is in the field of minimally-invasive surgery (MIS). In addition to traditional telemanipulation, we are interested in the application of *virtual fixtures* for operator assistance in MIS tasks. A virtual fixture is a constraint, implemented in software, that attempts to force a robot's movement along desired paths or prevent a robot from moving into forbidden regions. The potential benefit of virtual fixtures is safer and faster operation. Virtual fixtures attempt to capitalize on the accuracy of robotic systems, while maintaining a degree of operator control.

The goals of telemanipulator design all revolve around giving the user the highest possible control over the slave. In contrast, the goal of virtual fixture design is to remove some control from the user. Because these goals generally conflict with one another, it is not obvious how to best implement virtual fixtures on a telemanipulation system.

A. Previous Work in Telemanipulation

Early work in telemanipulation led to the two-port-network representation of a telemanipulator [5], [6], [18]. Recent work on analysis and design of telemanipulation controllers has used the two-port representation almost exclusively. Increasing transparency in a telemanipulator while retaining stability is a common research topic [3], [4], [21], [22]. Work has been done applying robust control techniques to telemanipulation [3], [4], [9], [13], [21], [23], and analyzing stability and transparency in the presence of time delays [1], [8], [10], [12], [13], [23]. Some nonrobust techniques have been developed that achieve "perfect" telepresence in a system that can be exactly modeled [24], but these techniques require accurate acceleration measurements not available in practice. The same type of nonrobust techniques have also been applied in the presence of time-delays [25]. Previous work has also considered position/rate control, where the master and slave devices work on different geometric scales [20], [23]. Recent research considers telemanipulation systems with various combinations of impedance-type and admittance-type master and slave devices [7].

B. Previous Work in Virtual Fixtures

The term "virtual fixture" refers to a general class of guidance modes that help a robotic manipulator perform a task by limiting its movement into restricted regions and/or influencing its movement along desired paths. As their name implies, forbidden-region virtual fixtures (FRVF) [17] prohibit the motion of a robot manipulator into forbidden regions of geometric or configuration space.

In [19], FRVFs were implemented as impedance surfaces on the master to assist in peg-in-hole tasks. In [16], FRVFs were implemented on the remote slave by rejecting master commands into the forbidden region. In [17], virtual fixtures were implemented on both the master and slave manipulators, using a variety of geometries, to help guide the remote manipulator in a predetermined task. Virtual fixtures have also been used in Human-Machine Collaborative Systems, such as Cobots [14] and the Johns Hopkins University Steady Hand Robot [2]. These collaborative systems are not telemanipulators because the human and robot simultaneously act on a single end-effector.

C. Goals of Experiment

To the best of the authors' knowledge, no previous research on telemanipulation with virtual fixtures investigates whether implementing virtual fixtures on the master or slave side (or both) leads to the most desirable system behavior. Also, no research has compared how a given virtual-fixturing methodology works with multiple control architectures.

The aim of this research is to compare different combinations of master and slave FRVFs with common telemanipulation control architectures, and to determine which combinations lead to the most desirable system behavior. The controller architecture used with a telemanipulator is usually dictated by hardware (actuators and sensors available). Thus, it is desirable to know which FRVF architecture is the "best" to use with a given control architecture, using qualitative metrics such as "tracking", "safety," and "submittance," which are defined quantitatively in Section V-A.

The control systems and virtual-fixture architectures used in this study are all simple, and for any stable system including environment and static human input the equilibrium position and force of the telemanipulator can be found analytically. This being said, a psychophysical study must be conducted to determine the effects of virtual fixtures when performing tasks at the threshold of human perception.

II. EXPERIMENTAL SETUP

The experimental setup used in this research consists of two Haptic Paddles [15], configured for telemanipulation (Figure 1). For each haptic paddle in this experiment, the motor was updated with a digital encoder, and an Entran ELFS-T3E-10N $\pm 10N$ load cell was added for direct force measurement. Also, the smooth-shaft capstan drive was replaced with a threaded shaft, giving more consistent behavior of the drive.

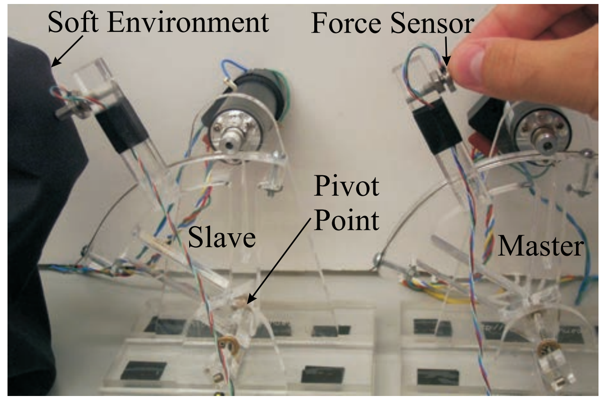

Fig. 1. Experimental system.

The modified Haptic Paddles are impedance-type devices, with high backdrivability and low mass. A local velocity-feedback loop is implemented on both devices to stabilize the system at higher position gains, but the impedance-type nature is retained. Linear models of both devices were found empirically. In Laplace form, the linear model of the master is $\frac{968}{s(s+16.3)}$, and the linear model of the slave is $\frac{742}{s(s+17.2)}$, where the input to the system is the voltage applied to the motor amplifiers, and the output is a rotation in radians. These models show the real differences between two "identical" plants. On the master device, an actuator voltage of $1V$ corresponds to a static force of 1.9 Newtons. On the slave device, $1V = 1.6N$. The master and slave are geometrically identical, with a paddle rotation of one radian corresponding to a motion at the load cell with an arclength of 115 mm.

A compliant environment is used in the experiment (Section V-B). The compliant environment is built from a soft sponge bound with a thick rubber band on its surface. This gives an environmental stiffness of approximately $K_{en} = 0.41 \frac{N}{mm}$.

III. TELEMANIPULATOR CONTROLLER ARCHITECTURES

Four controller architectures are considered in this experiment. These four controllers are shown in Figure 2 in their two-port representations, with generalized effort (e, force) and flow (f, velocity) variables. Figure 2(a) shows the position-forward (PF) controller. In this control mode the slave tracks the master with a simple position servo, and the master is not actuated.

Figure 2(b) shows the position-exchange (PE) controller. Here, the slave tracks the master's position, and the master simultaneously tracks the slave's position. With this controller, all forces fed back to master are generated from the position error between the master and slave. This adds a viscous feel to the master as it leads the slave.

Figure 2(c) shows the position-forward/force-feedback (PFFF) controller. The slave tracks the master's position, and the force felt between the slave and its environment is fed back to the master actuator. This is accomplished by commanding the appropriate voltage to the master actuator to create the desired force between the master and the human operator in a static situation. This controller reflects slave/environment forces "perfectly" to the master in a static situation, limited in practice by the resolution of the load cells and D/A card, as well as the calibration of the load cells. The PFFF controller feels identical to the PF controller when the slave interacts with a compliant environment of impedance $Z_e = 0$ (when the slave is free), but provides better telepresence than the PF controller when $Z_e \neq 0$.

Figure 2(d) shows the final controller, the position-exchange/force-feedback (PEFF) controller. In this control mode, the slave tracks the master's position, while the actuation of the master is the sum of a servo to the slave's position, as well as the reflected force felt between the slave and its environment. This controller combines the features of the PE and the PFFF controllers. The PEFF feels like the PE controller when $Z_e = 0$, but gives additional force feedback when $Z_e \neq 0$, making the environment feel stiffer than it really is.

IV. MASTER AND SLAVE VIRTUAL FIXTURES

There are a number of ways to implement FRVFs on a telemanipulation system, and it is not obvious which FRVF method is the best to use, given all the characteristics a viable system must possess. One method is to implement the virtual fixtures with impedance methods. Here the FRVF is represented by a hyperplane with a specified stiffness k_{VF}, and movement through the hyperplane results in an actuated force $F_a = k_{VF}\Delta$, where Δ is the normal distance of movement through the hyperplane. A second way to implement a FRVF is to disallow the slave to follow any movements of the master that are normal to the FRVF hyperplane when the master is in the forbidden region. A third way to implement FRVF is to scale down the movements of the master normal to the virtual fixture by a scaling gain k_{scale}, where $0 \leq k_{scale} \leq 1$. The second virtual-fixturing method discussed is a special case of this third method, when the normal components are scaled down to zero ($k_{scale} = 0$). The first two FRVF methods are illustrated in Figure 3(a) and Figure 3(b), respectively.

In this study, four levels of FRVF will be considered: soft, hard, infinite, and none (the control case). A soft FRVF is implemented as an impedance-type fixture with

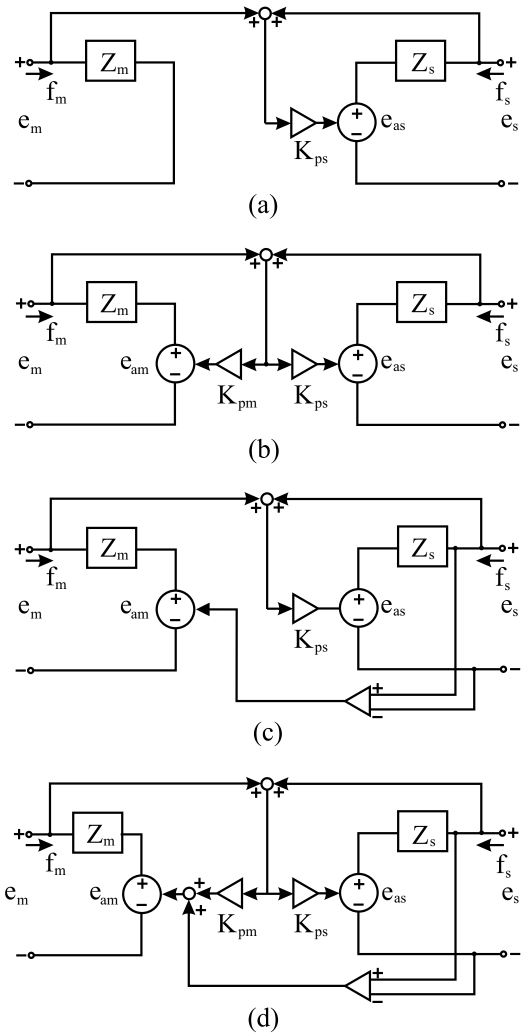

Fig. 2. Telemanipulation Controller Architectures (Two-Port Representations): (a) position forward (PF), (b) position exchange (PE), (c) position forward/force feedback (PFFF), and (d) position exchange/force feedback (PEFF).

$k_{VF} = 100$. This corresponds to an actuator voltage equal to 100 times the depth through the virtual fixture, in radians. The soft FRVF gives a compliant feel as the virtual fixture is penetrated. The hard virtual fixture corresponds to $k_{VF} = 500$, defined as before. Qualitatively, the hard virtual fixture appears to the user to have almost no compliance. For the infinite FRVF, the master motion through and normal to the FRVF hyperplane is scaled by $k_{scale} = 0$ before being commanded to the slave, disallowing any movement of the slave through the FRVF. The control case of no FRVF and the soft FRVF are implemented on both the master and slave. The infinite FRVF is only implemented on the slave, by definition. The hard FRVF is only implemented on the master, because initial trials showed that the hard FRVF implemented on the slave

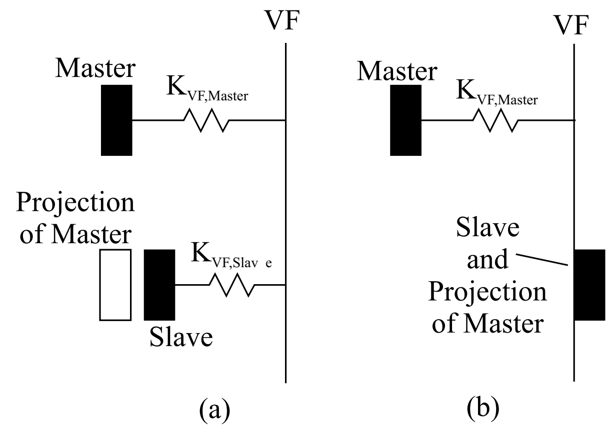

Fig. 3. Forbidden-Region Virtual Fixtures (FRVF): Impedance-type FRVF at master and (a) impedance-type FRVF at slave and (b) infinite-type FRVF at slave (motion scaling). The forbidden region is left of the virtual fixture.

could lead to unstable vibrations "against" the virtual fixture. Three types of FRVF on the slave and three types on the master give a total of nine FRVF combinations used in this study.

V. EXPERIMENT

A. Metrics

The qualitative metrics "tracking," "safety," and "submittance," need to be defined quantitatively for the purpose of analysis . *Tracking* is measured here as the inverse of the worst-case position error between the master and slave devices. To make the user feel that his/her movements are being directly recreated at the slave device, we would like the absolute value of the position error between the master and slave to be as small as possible, leading to good *Tracking*.

The purpose of the FRVF is to prevent the slave device from entering forbidden regions. From a safety perspective, it is inconsequential if the master device enters a projection of these same forbidden regions. For this reason, *Safety* is defined as the inverse of the maximum penetration of the FRVF hyperplane by the slave device. No negative penetrations are considered here, so two systems that never penetrate the FRVF on the slave side are both considered equally safe, regardless of which system came closest to the forbidden region.

We define the final metric, *Submittance*, to quantify the ability of the user to move the slave to a desired position, regardless of any virtual fixture on the master. When implementing FRVFs on a telemanipulation system, certain combinations of master and slave fixtures, designed to prevent movement into the forbidden region, can actually prevent the slave from ever reaching the fixture. This phenomenon is only seen when there is a disturbance load on the slave. There are some circumstances where reaching the FRVF is necessary, such as when a virtual fixture is used to limit as well as act as a guide for depth of cutting, and a system with good *Submittance* will allow this. *Submittance* is used in this experiment to quantify whether there is any master position that will lead to the desired slave position, and if not, to determine how close to the desired position the slave is moved.

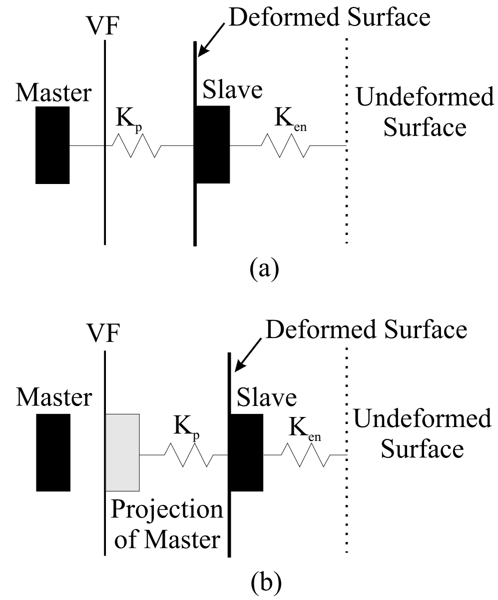

Fig. 4. Visualization of Submittance: (a) Impedance-type FRVF at slave (high level of submittance) and (b) Infinite FRVF at slave (low level of submittance). The forbidden region is left of the virtual fixture.

Figure 4 illustrates how a system can lose *Submittance*. In Figure 4(a), the slave is tracking the master with an actuator force proportional to the position error between them. The slave is also interacting with a compliant environment with stiffness K_{en}, and this environmental force tends to increase the error between the master and the slave as the slave moves into the compliant environment. If the master and slave FRVFs are both of the impedance type, and the master FRVF is not too stiff, it is possible to command the slave to the depth of the fixture by moving the master past the fixture. In Figure 4(b), an infinite virtual fixture is implemented on the slave side as described in the previous section. Regardless of the type of FRVF on the master, the slave can never reach the fixture. To quantify this phenomenon, in our experiment *Submittance* is defined as the inverse of the minimum distance between the slave and its goal when the slave fails to reach its goal. Whenever the slave reaches its goal, the minimum distance used in the calculation of *Submittance* is set to zero.

B. ANOVA Experiment

An experiment was conducted to quantitatively compare the performance of systems with varying control architectures and FRVF methodologies, using the three metrics described above. The experiment was designed to simulate a scenario where the user knows the location of forbidden regions, and test how well FRVFs help the user perform tasks safely. A mixed-effects analysis of variance (ANOVA) experiment was conducted. Four control architectures, nine FRVF combinations, and two tasks give a total of 72 levels of the fixed-effect factor, representing all possible combinations. Six subjects were used in the experiment, giving six levels to the random-effect factor. Although the three metrics used are defined as inverses of measured distances, the actual ANOVA is performed on the measured distances themselves.

The user is asked to perform two different tasks. The first task, *Touch*, simulates the user attempting to work near a forbidden region without entering it. In this task, the subject was asked to move the slave forward and touch the surface of the compliant environment with the slave device, but to not depress the environment at all. The subject was instructed to stop and retract the slave when it was determined that the surface had been touched, using all visual and haptic information available.

The second task, *Depress*, simulates a FRVF to limit depth of cutting, needle insertion, etc. In this task, the subject was asked to move the slave forward and depress the surface of the compliant environment to a depth equivalent to half of the threaded portion of the load cell on the slave (4 mm), where the FRVF was placed. The subject was also shown an example of the slave depressing the surface to the correct depth. The subject was instructed to stop and retract the slave when it was determined that the slave had depressed the surface to the correct depth, using all visual and haptic information available.

In both the *Touch* and the *Depress* tasks, the subjects were told that if the slave device did not reach the desired position the trial would be repeated. The trial was actually only repeated if the master device failed to reach the correct position. This was done because some FRVF combinations forbid the slave from reaching the correct depth. In these cases, haptic cues to the user prematurely indicate that the correct depth has been reached.

Each subject was asked to perform each of the 72 control/FRVF/task combinations three times, giving a total of 216 data runs assigned randomly to each subject, with 1296 data sets in total for the experiment. The average time to complete all the trials was approximately 40 minutes per subject. Subjects were allowed rest at any time. From each of the data sets, *Tracking* and *Safety* were calculated; *Submittance* was only calculated for the *Depress* task, because it only becomes an issue when the slave is loaded.

For reduced complexity, only one set of position, velocity-feedback, and force gains is used for each control architecture, but steps were taken to make these systems "equivalent." First, the same local velocity-feedback was used on the master and slave for every controller ($K_v = 0.2\frac{V}{rad/s}$). Second, whenever position information is used, a position gain of $K_p = 50\frac{V}{rad}$ is used. The choice of identical position gains makes the PE and PEFF controllers feel more sluggish than the PF and PFFF controllers, but it creates systems that have identical steady-state position errors to loads on the slave. The position gain chosen gives good position tracking, while still generating a stable system with a smooth feel. Finally, whenever force-feedback is used, it is unity force-feedback. This means that the proper voltage is given to the master motor to make the force seen at the master load cell equal that seen at the slave load cell (in a static situation).

VI. RESULTS AND DISCUSSION

Figure 5 shows the results of Tukey's Method of Pairwise Comparisons [11] – a test to determine if two data sets are significantly different from one another – for each of the four control architectures. Although forbidding any negative distances for *Safety* and *Submittance* slightly harms the normal distribution of the data, ANOVA is robust to this [11]. For each controller, the results are shown for *Tracking* and *Safety* for both the *Touch* and *Depress* tasks, and *Submittance* for the *Depress* task. The vertical bars indicate FRVF pairs that statistically are not significantly different from one another, with $p = 0.05$. Any FRVF pair that belongs to multiple groupings indicates a questionable result. For these cases, a more sensitive test is needed to sufficiently differentiate the groupings.

The results for *Tracking* were the same across all four controllers. They indicate that an infinite slave FRVF with no master FRVF gives undesirable tracking, and all other FRVF pairs give equally good tracking. This is most clearly seen in the PF and PFFF controllers, indicating that position feedback to the master can improve tracking with an infinite slave FRVF and no master FRVF. This makes sense; in the limit as the position gain at the master becomes very large, the master's position is unable to deviate from the slave's, regardless of the virtual fixture used at the slave.

The results for *Safety* are the same for the PF, PE, and PFFF controllers. They indicate that for improved safety, a FRVF should be used at the master side if an impedance-type FRVF is being implemented at the slave. Otherwise, all other FRVF schemes are equally safe. This is because a master with no FRVF can easily "pull" the slave through an impedance-type FRVF some finite distance before the FRVF generates enough actuator force to stop the slave. The results for the PEFF controller indicate that all FRVFs

Touch Tracking	*Depress Tracking*	*Touch Safety*	*Depress Safety*	*Depress Submittance*
FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave
H/0	H/0	S/I	S/I	0/S
S/S	H/S	H/S	H/S	0/0
S/I	H/I	H/I	H/I	S/S
0/0	S/0	0/I	H/0	S/0
H/S	S/S	H/0	0/I	H/0
H/I	S/I	S/S	S/S	H/S
S/0	0/0	S/0	S/0	H/I
0/S	0/S	0/S	0/S	0/I
0/I	0/I	0/0	0/0	S/I

(a) Position-Forward Control Architecture

Touch Tracking	*Depress Tracking*	*Touch Safety*	*Depress Safety*	*Depress Submittance*
FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave
H/I	H/0	S/I	S/I	0/0
H/0	H/S	H/S	H/S	0/S
0/0	H/I	H/I	H/I	S/S
0/I	S/S	H/0	H/0	S/0
S/0	S/0	0/I	0/I	H/S
0/S	S/I	S/S	S/S	H/0
S/I	0/0	S/0	S/0	0/I
S/S	0/S	0/S	0/S	S/I
H/S	0/I	0/0	0/0	H/I

(b) Position-Exchange Control Architecture

Touch Tracking	*Depress Tracking*	*Touch Safety*	*Depress Safety*	*Depress Submittance*
FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave
S/S	H/0	S/I	S/I	0/0
S/0	H/S	H/S	H/S	0/S
S/I	S/S	H/I	H/I	S/0
H/0	H/I	H/0	H/0	S/S
H/S	S/0	0/I	0/I	H/S
0/0	S/I	S/S	S/S	H/0
0/S	0/0	S/0	S/0	S/I
H/I	0/S	0/S	0/S	H/I
0/I	0/I	0/0	0/0	0/I

(c) Position-Forward/Force-Feedback Control Architecture

Touch Tracking	*Depress Tracking*	*Touch Safety*	*Depress Safety*	*Depress Submittance*
FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave	FRVF Master/Slave
S/0	H/I	0/I	S/I	0/0
H/0	S/0	H/0	S/S	0/S
S/S	H/S	S/I	H/I	S/0
0/0	H/0	H/S	H/S	S/S
0/I	S/I	H/I	S/0	H/0
H/I	S/S	S/S	0/I	H/S
S/I	0/0	S/0	H/0	0/I
0/S	0/S	0/S	0/S	S/I
H/S	0/I	0/0	0/0	H/I

(d) Position-Exchange/Force-Feedback Control Architecture

Fig. 5. Statistical Results using Tukey's Method of Pairwise Comparisons. FRVF pairs are listed in descending order (best performance in each category

are equally safe. This is probably due to the large amount of haptic cues given to the user.

The results for safety are somewhat counter-intuitive. The statistics show that a system with an infinite slave FRVF is not significantly safer than many other systems, even though the only way to guarantee that the slave never crosses into a forbidden region is with an infinite slave FRVF. The results for a hard master FRVF are questionable for similar reasons. The reason for this result is that the *Touch* and *Depress* tasks did not capture every scenario the telemanipulator may experience. Recall that the experiment was designed to simulate a situation where the user knows where the forbidden regions are. In the *Touch* experiment, the user was instructed to touch the surface of the compliant environment, but to not depress the surface at all. Because of this instruction, the user used visual cues to help the slave approach the environment slowly, so almost any FRVF scheme worked to create a safe system. This experiment did not test scenarios where either the user did not have good visual cues or did not realize the environment was a forbidden region. In these two scenarios, the user could move quickly into a FRVF, and an infinite slave FRVF and/or a hard master FRVF would probably give safer results than other FRVF schemes.

To determine if the infinite slave FRVF and/or the hard master FRVF are significantly safer than other FRVF schemes, additional experiments must be conducted. Two possibilities would be to modify the *Touch* experiment by obstructing the user's visual cues from the slave, or to instruct the user to do the *Depress* experiment, but to set the FRVF at the surface of the environment. These tests may reveal that the infinite slave FRVF and/or the hard are safer overall, but they could possibly also change the *Tracking* and *Submittance* results for these FRVFs. In fact, intuitively, these two FRVF schemes should give the worst *Submittance* of any of the FRVF architectures.

A quick glance at the *Submittance* results indicate that *Safety* and *Submittance* are inversely related to one another. This is intuitive, when *Submittance* is a measure of user control, and *Safety* is a measure of the lack of user control. The *Submittance* results are the same for all controllers. A FRVF architecture with no master FRVF and an impedance-type slave FRVF gives the most *Submittance*. A FRVF architecture with a soft master FRVF and an impedance-type slave FRVF gives the next-highest *Submittance*. Finally, any FRVF scheme with either a hard master FRVF or infinite slave FRVF leads to equally poor *Submittance*. In addition, there is also a distiction between the two most schemes with the highest *Submittance* for the PFFF controller. Here, no FRVFs appear to give better *Submittance* than a FRVF scheme with no master FRVF and a soft slave FRVF.

Because the *Safety* and *Submittance* metrics are inversely related, no single FRVF method is obviously the "best" overall for use with a given control architecture. The final choice of FRVF method should be made with a specific application in mind, as a balance of *Tracking*, *Safety*, and *Submittance*.

VII. CONCLUSIONS

Thirty-six different systems (consisting of nine FRVF methods and four control architectures) were implemented on a 1-DOF teleoperated system. A six-subject ANOVA experiment was created to quantify the performance of the systems during two tasks that simulate a user working near a known forbidden region, with the intent of finding the best FRVF architecture for a given controller.

The experimental results show that performance is generally the same across all control architectures. They indicate that for good tracking, a telemanipulator should not use a FRVF architecture that has an infinite FRVF at the slave with no FRVF at the master. For safe operation, any telemanipulator that has an impedance-type FRVF at the slave should also have a FRVF at the master. For the best submittance, a system should be configured with an impedance-type FRVF at the slave, and no FRVF (or possibly a soft FRVF) at the master. Rather than making a conclusion on the "best" FRVF architecture overall, the desired application of the telemanipulator should be taken into account when choosing how to balance the systems's tracking, safety, and submittance.

In the future, we will investigate three different extensions of the research presented here. First, will analytically study the stability of telemanipulators interacting with virtual fixtures. Vibration against an FRVF should be prevented, regardless of the human operator's actions or environment encountered by the slave. Second, we will experiment with virtual fixtures on telemanipulation systems where one or both of the master and slave devices is of the admittance type (nonbackdrivable). There is reason to believe that this could lead to systems with desirable characteristics, i.e. nearly perfect position tracking may be possible for a large range of environmental impedances. Finally, we will extend the results of this research to higher degrees of freedom. Many results from $1DOF$ will apply directly to $3DOF$, but higher degrees of freedom will add increased complexity to the problem. For example, in a $1DOF$ system master and slave movements are *always* normal to the FRVF, but with higher degrees of freedom this will not be the case.

VIII. ACKNOWLEDGEMENTS

We would like to thank Jason Nolin and Evan Murphy for their contributions. This material is based on work supported by the National Science Foundation, grant #ITR-0205318.

IX. REFERENCES

[1] R. J. Anderson and M. W. Spong. Bilateral Control of Teleoperators with Time Delay. *IEEE Trans. Automatic Control*, 34(5):494-501, 1989.

[2] A. Bettini, S. Lang, A. Okamura, and G. Hager. Vision Assisted Control for Manipulation Using Virtual Fixtures. *IEEE/RSJ Intl. Conf. Intelligent Robots and Systems*, 1171-1176, 2001.

[3] M. C. Çavuşoğlu, A. Sherman, F. Tendick. Bilateral Controller Design for Telemanipulation in Soft Environments. *IEEE Intl. Conf. Robotics and Automation*, 1045-1052, 2001.

[4] J. E. Colgate. Robust Impedance Shaping Telemanipulation. *IEEE Trans. Robotics and Automation*, 9(4):374-384, 1993.

[5] B. Hannaford. A Design Framework for Teleoperators with Kinesthetic Feedback. *IEEE Trans. Robotics and Automation*, 5(4):426-434, 1989.

[6] B. Hannaford. Stability and Performance Tradeoffs In Bi-Lateral Telemanipulation. *IEEE Intl. Conf. Robotics and Automation*, 1764-1767, 1989.

[7] K. Hashtrudi-Zaad and S. E. Salcudean. Analysis of Control Architectures for Teleoperation Systems with Impedance/Admittance Master and Slave Manipulators. *Intl. J. Robotics Research*, 20(6):419-445, 2001.

[8] K. Hashtrudi-Zaad and S. E. Salcudean. Transparency in Time-Delayed Systems and the Effect of Local Force Feedback for Transparent Teleoperation. *IEEE Trans. Robotics and Automation*, 18(1):108-114, 2002.

[9] H. Kazerooni, T.-I. Tsay, and K. Hollerbach. A Controller Design Framework for Telerobotic Systems. *IEEE Trans. Control Systems Technology*, 1(1):50-62, 1993.

[10] W. S. Kim, B. Hannaford, and A. K. Bejczy. Force-Reflection and Shared Compliant Control in Operating Telemanipulators with Time Delay. *IEEE Trans. Robotics and Automation*, 8(2):176-185, 1992.

[11] D. G. Kleinbaum, L. L. Kupper, K. E. Muller, and A. Nizam. *Applied Regression Analysis and Other Multivariable Methods*, chapter 17, pp. 447–450. Duxbury Press, Pacific Grove, California, 1998.

[12] D. A. Lawrence. Stability and Transparency in Bilateral Teleoperation. *IEEE Trans. Robotics and Automation*, 9(5):624-637, 1993.

[13] G. M. H. Leung, B. A. Francis, and J. Apkarian. Bilateral controller for teleoperators with time delay via μ-synthesis. *IEEE Trans. Robotics and Automation*,11(1):105-116, 1995.

[14] C. A. Moore, M. A. Peshkin, and J. E. Colgate. Cobot Implementation of 3D Virtual Surfaces. *IEEE Intl. Conf. Robotics and Automation*, 3242-3247, 2002.

[15] A. M. Okamura, C. Richard, and M. R. Cutkosky. Feeling is believing: Using a Force-Feedback Joystick to Teach Dynamic Systems. *ASEE J. Engineering Education*, 92(3):345-349, 2002.

[16] S. Park, R. D. Howe, and D. F. Torchiana. Virtual fixtures for robotic cardiac surgery. *Fourth Intl. Conf. on Medical Image Computing and Computer-Assisted Intervention*, 2001.

[17] S. Payandeh and Z. Stanisic. On Application of Virtual Fixtures as an Aid for Telemanipulation and Training. *Proc. 10th Symp. On Haptic Interfaces For Virtual Envir. and Teleoperator Systs.*, 2002.

[18] G. J. Raju, G. C. Verghese, and T. B. Sheridan. Design Issues in 2-Port Network Models of Bilateral Remote Manipulation. *IEEE Intl. Conf. Robotics and Automation*, 1316-1321, 1989.

[19] L. Rosenberg. Virtual Fixtures: Perceptual Tools for Telerobotic Manipulation. *Proc. IEEE Virtual Reality International Symposium*, 76-82, 1993.

[20] S. E. Salcudean, M. Zhu, W.-H. Zhu, and K. Hashtrudi-Zaad. Transparent Bilateral Teleoperation Under Position and Rate Control. *Intl. J. Robotics Research*, 19(12):1185-1202, 2000.

[21] A. Sherman, M. C. Çavuşoğlu, and F. Tendick. Comparison of Teleoperator Control Architectures for Palpation Task. In *Proc. ASME Dynamic Systems and Control Division, part of the ASME Intl. Mechanical Congress and Exposition (IMECE 2000)*, volume 2, 1261-1268, November 2000.

[22] J. E. Speich, K. Fite, and M. Goldfarb. A Method for Simultaneously Increasing Transparency and Stability Robustness in Bilateral Telemanipulation. *IEEE Intl. Conf. Robotics and Automation*, 2671-2676, 2000.

[23] J. Yan and S. E. Salcudean. Teleoperation Controller Design Using H_∞-Optimization with Application to Motion-Scaling. *IEEE Trans. Control Systems Technology*, 4(3):244-258, 1996.

[24] Y. Yokokohji and T. Yoshikawa. Bilateral Control of Master-Slave Manipulators for Ideal Kinesthetic Coupling – Formulation and Experiment. *IEEE Trans. Robotics and Automation*, 10(5):605-620, 1994.

[25] Y. Yokokohji, T. Imaida, and T. Yoshikawa. Bilateral Teleoperation Under Time-Varying Communication Delay. *IEEE/RJS Intl. Conf. Intelligent Robots and Systems*, 1854-1859, 1999.