Implementation of a virtual tennis entertainment system with haptic behaviour

K-S Hsu¹*, M-Y Cheng² and M-G Her³
¹Department of Automation Engineering, Kao Yuan Institute of Technology, Kaohsiung, Taiwan
²Department of Electrical Engineering, National Cheng Kung University, Kaohsiung, Taiwan
³Department of Mechanical Engineering, Tatung University, Tatung, Taiwan

Abstract: For most of the virtual reality systems, one of the major aims is to provide a vivid interaction platform between the human operators and the haptic devices. Through the user interface, a skillful operator can control the haptic devices to accomplish relatively complicated jobs in real-time. Generally, the main components of a virtual reality system include dynamic simulations, haptic devices and the user interface, which is composed of virtual environments and visual equipment. This study focuses on developing a virtual tennis entertainment system with haptic behaviour. A parallel-type robot and a serial-type robot are employed as the haptic device handlers in this study, in which they are controlled directly by the operator’s arm through the user interface. The operator can sense the change in virtual environment provided by dynamic simulations. In addition, the human operator can ‘see’ the change of environment during operation in real-time through the screen. A virtual spring model and a virtual damper model were constructed to simulate the process of tennis playing in this study. Experimental results verify the feasibility of the proposed virtual tennis entertainment system.

Keywords: haptic device behaviour, virtual reality, direct-drive arm robot, parallel robot

NOTATION

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>coefficient of viscous damping of the virtual environment</td>
</tr>
<tr>
<td>b₁</td>
<td>damper constant</td>
</tr>
<tr>
<td>e(t)</td>
<td>error between the target and the actual response</td>
</tr>
<tr>
<td>fₑ</td>
<td>force imposed on the handler actuator by the environment</td>
</tr>
<tr>
<td>fₕ</td>
<td>force imposed on the handler actuator by the operator</td>
</tr>
<tr>
<td>G</td>
<td>closed-loop transfer function of the handler</td>
</tr>
<tr>
<td>Gₒₒₕ</td>
<td>open-loop transfer function of the 2-DOF direct-drive robot</td>
</tr>
<tr>
<td>Kₜ</td>
<td>compensator transfer function for fₕ</td>
</tr>
<tr>
<td>k</td>
<td>stiffness of the spring</td>
</tr>
<tr>
<td>kₐ</td>
<td>gain constant of the D controller</td>
</tr>
<tr>
<td>kᵢ</td>
<td>gain constant of the I controller</td>
</tr>
<tr>
<td>kᵣ</td>
<td>gain constant of the P controller</td>
</tr>
<tr>
<td>m</td>
<td>mass of a ball</td>
</tr>
</tbody>
</table>

1 INTRODUCTION

Due to rapid progress in computer technology and haptic devices, virtual reality systems have enjoyed a great success and have been applied to various fields such as education, entertainment, industry and the arts. Recently, several studies concerning the application of robots in virtual reality systems have been reported [1–11]. Kazerooni and Her [1] proposed a control scheme
for haptic interfaces. A two-degree-of-freedom (2-DOF) electric haptic interface was built to verify their approach. Other researchers studied the virtual environment of a cutting process [2–4] and the idea of virtual walls [5, 6]. A common feature for these studies is that they all intended to provide the operators with an environment that seemed as if they were in the real world.

How to provide vivid interactions between the human operators and the virtual environments is crucial in developing a successful virtual reality system. Human operators interact with the virtual reality systems through the senses of sight, hearing and touch. In order to make dynamic simulations of touch between the human operators and the virtual objects more realistic, the haptic devices are controlled in such a way as to increase the sense of reality so that the human operators can actually ‘feel’ interactions with the virtual environment. Moreover, by adding visual effects, the sense of reality can be further increased. Virtual reality systems are often employed in applications such as telerobotics and manpower machines [7, 8]. In a telerobotic system, the human operator manipulates the master robot through equipment with a sense of touch. The motion of the master robot is converted into the motion commands of the slave robot, in which the slave robot executes these commands to mimic the motion of the master robot. On the other hand, the slave robot also transmits the information obtained from the interactions between the slave robot and the virtual environment back to the master robot, so that the operator can ‘feel’ or ‘see’ what the operation process is really like. Telerobotic systems can also be used in a scenario where the force to be applied exceeds the manoeuvring capability of the human operator; namely the force exerted by the human operator is inadequate and needs to be reinforced. Other applications of telerobotic systems include operations in hazardous environments. Human operators can sit in a control room that is far from danger and manipulate a slave robot to execute a certain task [9–11]. This kind of technology can also be applied to teaching simulators. It can avoid any danger caused by improper operations and result in a significant cost saving for the maintenance of facilities.

In this study, two robots are used to simulate the haptic devices. In general, an ideal haptic device must possess characteristics such as low friction, low inertia, high sensitivity and large bandwidth. However, it is unlikely that all the requirements can be satisfied without any modification. For instance, under some circumstances, the driving motor of a robot may have a high gear ratio; it would be difficult to drive the motor by simply applying some external force. To overcome this difficulty, extra information provided by the force sensor is necessary. Force sensors are used to detect the touch force between the haptic device and the human operator, in which the measured force signal is converted into the position command of the robot via an analogue-to-digital (A/D) converter. The two robots used to simulate the haptic devices in this study belong to the type consisting of direct-drive robots and gear-retard robots. The advantage of a direct-drive robot is its quick mobility, but it also suffers from several drawbacks such as a smaller force feedback and less stability. On the other hand, the advantage of a gear-retard robot is that it has a larger force feedback and is more stable, but its movement is slow.

This paper is organized as follows. A brief introduction to the mathematical model of the virtual tennis entertainment system developed in this study is provided in section 2. Section 3 discusses the details of the experimental set-up. The virtual environment developed in this study is addressed in section 4. Experimental results and conclusions are given in sections 5 and 6 respectively.

2 MATHEMATICAL MODEL OF THE VIRTUAL TENNIS ENTERTAINMENT SYSTEM

The aim of this study is to develop a virtual tennis entertainment system. Therefore the models that describe the dynamics of the components in the virtual tennis entertainment system must be derived in advance. This virtual system model consists of three parts, the human arm, a haptic device and the virtual environment. In the following, the mathematical model for each component will be derived.

2.1 Dynamics of the human arm and a model of the haptic device

The block diagram of a human-in-the-loop system containing a human arm and a haptic interface device is illustrated in Fig. 1, where $m_h$ is the human muscle force, $f_h$ the force imposed on the haptic device, $T_h$ the transfer function of the human arm dynamics, $S_h$ the sensitivity of the human force to the handler position, $G$ the closed-loop transfer function of the handler actuator, $K_h$ the compensator transfer function for $f_h, v$ the desired position and $p$ the position of the handler.

In general, the action behaviour of a human arm can be classified into three types:

(a) when the hand does not carry any load, namely the arm is moving freely in the space;
(b) when the hand is carrying a heavy load, the human arm can be viewed as a torque control system;
(c) when the hand is carrying a light load, the human arm can be viewed as an impedance control system.

Nonetheless, when someone is pushing an object, the behaviour of his or her arm is likely to contain all three types of actions. Hence it is unreasonable to divide the
human arm dynamics into any specific type of action. This idea can be seen clearly from the block diagram in Fig. 1, in which the human arm’s muscle not only involves pushing an object but also carries the whole arm. Therefore, by subtracting the force corresponding to the dynamics of the human arm, the net force $f_h$ applied to the haptic device by the human arm can be obtained.

In Fig. 1, the transfer function $T_h$ is used to describe the dynamics of a human arm [2]. In order to derive $T_h$, an experiment is conducted to obtain the frequency response of $T_h$. In the experiment, first an operator is asked to grasp a handler of the robot without exerting any force and then the operator gradually increases the grasping force to follow the robot arm’s movement until the operator cannot keep grasping any more. Note that the robot arm performs a sinusoid-function-like movement and its associated frequency is gradually increased. In this way, the input positions and the force magnitudes measured from the force sensor, namely the frequency response of $T_h$, can be recorded. Three human operators were asked to perform the experiment; the corresponding Bode plots for these three operators are shown in Fig. 2. Based on these Bode plots, a curve-fitting technique is used to obtain a transfer function that best fits the frequency response of $T_h$. The obtained transfer function for the human arm dynamics is expressed as

$$T_h = 0.25s^2 + 2s + 6$$

The obtained transfer function for the human arm dynamics is expressed as

$$T_h = 0.25s^2 + 2s + 6$$

![Fig. 1](image1.png)  
**Fig. 1** Block diagram of a human-in-the-loop system containing a human arm and a haptic device

![Fig. 2](image2.png)  
**Fig. 2** Magnitude plot of the operator arm dynamics: +, first operator; ○, second operator; *, third operator; solid line represents the theoretical approximation using the curve-fitting technique
On the other hand, based on Fig. 1, it is found that

\[ p = (S_h + K_h G) f_h \]  

(2)

A parallel robot and a 2-DOF direct-drive robot are adopted as the haptic devices in this study. The transfer function \( S_h \) of the parallel robot is different from that of the 2-DOF direct-drive robot. The parallel robot used in this study is non-back-drivable. In other words, only the controller can change the position of the handler actuator. The force of the human arm, which is exerted on the handler actuator, cannot change the position of the handler actuator. Therefore, when deriving the model of the experimental system, the transfer function \( S_h \) for the parallel robot can be ignored. On the other hand, the 2-DOF direct-drive robot is back-drivable; hence \( S_h \) is significant compared to \( G \). In other words, \( S_h \) for the 2-DOF direct-drive robot cannot be neglected.

2.2 Model of the virtual system

Combining the dynamics of the virtual environment with the models for the human arm and the haptic device, a model can be constructed for the virtual system as shown in Fig. 3, where \( f_e \) is the force imposed on the handler actuator by the environment, \( T_e \) the transfer function of the environment dynamics, \( K_h \) the compensator transfer function for \( f_h \), \( K_e \) the compensator transfer function for \( f_e \) and \( S_h \) the sensitivity function of the operator force to the handler position.

In practice, each object has its own physical characteristics, such as elasticity, rigidity, destroying stress, strain and so on. These physical characteristics can be described by simply using a spring model or a damper model. As a result, the transfer functions for every component of the virtual environments can be obtained by applying the Laplace transform to the spring models and the damper models. Details concerning the transfer functions of the virtual environments developed in this study are addressed in the following. Based on Fig. 3, the desired position \( v \) and the position of the handler \( p \) can be expressed respectively as

\[ v = K_h f_h - K_e f_e \]  

(3)

\[ p = G v + S_h f_h \]  

(4)

Substituting equation (3) into equation (4) yields

\[ p = G (K_h f_h - K_e f_e) + S_h f_h \]  

(5)

where

\[ f_e = T_e p \]

After some mathematical manipulations, equations (3) to (5) give

\[ \frac{f_e}{f_h} = \frac{S_h T_e + G T_e K_h}{1 + G T_e K_e} \]

(6)

Note that the aim of a virtual reality system is to make \( f_h \) and \( f_e \) as close as possible. To accomplish this, the values of \( K_h \) and \( K_e \) in equation (6) need to be regulated. If the gain constant of the force compensator exceeds a specified level, the system may become unstable.

2.3 Model of the virtual system with two haptic devices

If there are two haptic devices used in the same virtual reality system, the block diagram of the whole system can be redrawn as shown in Fig. 4. The positions of the handlers for the two haptic devices can be obtained through the same transfer function \( T_e \). In addition, the position command for each haptic device can be obtained via its corresponding compensator.
3 EXPERIMENTAL SET-UP

The experimental set-up of the virtual reality system for simulating the tennis ball hitting system, the environment and the paddle is shown in Fig. 5. Details concerning the experimental set-up are given as follows.

3.1 2-DOF direct-drive robot

The 2-DOF direct-drive robot developed in this study (Fig. 5) is driven by two direct-drive a.c. servomotors. The PCL-816 A/D converter card converts the analogue signals measured from a force sensor into digital signals (position), while the D/A converter converts digital signals into analogue signals that are used as the commands for a.c. servomotors. The position outputs of a.c. servomotors are obtained via the PCL-833 quadrature encoder card and used as the feedback signals of the position loop. In contrast, the position of the manipulator can be obtained by simply computing the manipulator kinematics. In addition, a two-channel force sensor is mounted on the end-effector of the manipulator. The force sensor is used to measure the touch force between the manipulator and the human operator, in which the measured force signal is amplified via the PCL-816 and is fed back to the force compensator. The control kernel is a Pentium 133 personal computer and all of the control algorithms are implemented using TURBO C++.

The block diagram of the 2-DOF direct-drive robot is illustrated in Fig. 6, where $G_{\text{dop}}$ is the open-loop transfer function of the 2-DOF direct-drive robot, PID represents the proportional-integral-derivative (PID) control-
ler and $S_{hd}$ is the transfer function of sensitivity of the 2-DOF direct-drive robot. By adjusting the gain constants of the PID controller appropriately, a satisfactory performance can often be achieved. Nonetheless, the effects $S_{hd}$ caused by the force of the human arm exerted on the robot’s position must be considered if better performances of the closed-loop control are desired.

Detailed procedures of deriving $S_{hd}$ for the 2-DOF direct-drive robot can be found in the paper by Her et al. [2] but are omitted here. Experiments for obtaining the frequency response of $S_{hd}$ are performed and the results are illustrated in Fig. 7. By using a curve-fitting technique, a curve (i.e. $S_{hd}$) that best fits the obtained frequency responses (the dots in Fig. 7) can be found, in which the obtained transfer function of sensitivity $S_{hd}$ is expressed as

$$S_{hd} = \frac{13.1}{s^2 + 32.4s + 492.46}$$

(7)

### 3.2 Parallel-type robot

The parallel-type robot developed in this study is shown in Fig. 8. The moving platform has three degrees of freedom ($X$, $Y$, and $Z$) and is always parallel to the base. A Pentium 200 personal computer equipped with a DSP-LC/DSP motion control card (made by MEI Company) is used as the control kernel. In addition, all of the control algorithms are implemented using Borland C++ 3.0. The parallel robot is powered by an ABS-coordinate a.c. servomotor, while the retarder motor is the harmonic drive of a CSF series with a retard ratio of 1:100.

The force exerted by the operator is measured using a force sensor, in which an amplifier is used to process and magnify the measured force data. The measured force data are then transmitted (using the PCL-816 A/D converter card) to the computer as the position command of the parallel robot. The block diagram of
the parallel robot is shown in Fig. 9, where $G_{\text{top}}$ is the open-loop transfer function of the parallel robot. Similar to the case of the 2-DOF direct-drive robot, a satisfactory performance of the parallel robot can be achieved by adjusting the gain constants of the PID controller appropriately. As mentioned previously, the parallel robot is not affected by the sensitivity $S_h$ at all; hence $S_h$ can be ignored in the parallel robot system.

### 3.3 Controller design

Despite the fact that many novel control methodologies have been proposed and gained significant progress in recent years, the conventional PID controller is still very popular among industrial users. With a relatively simple structure, the PID controller is easy to implement and operate compared with the new generations of controllers, which often have complex structures. Since the control algorithm for the PID controller is simple, its computation time is short, making it particularly useful in implementing real-time control tasks. In addition, even under circumstances where the plant model is not known, the PID controller can still produce a satisfactory performance by adjusting its gain constants appropriately. The general form of a PID controller is expressed as

$$u(t) = k_p e(t) + k_d \dot{e}(t) + k_i \int e(t) \, dt \tag{8}$$

where $u(t)$ is the control command, $e(t)$ the error between the target and the actual response, $k_p$ the gain constant of the P controller, $k_d$ the gain constant of the D controller and $k_i$ the gain constant of the I controller. Taking the Laplace transform of equation (8) will yield

$$U(s) = \left(k_p + k_d s + \frac{k_i}{s}\right) E(s) \tag{9}$$

### 4 VIRTUAL ENVIRONMENT

The virtual environment developed in this study will be discussed in the following.

#### 4.1 Equipment set-up of the virtual environment

The set-up of the equipment of the virtual environment is shown in Fig. 10. The virtual environment is simulated using the control computer of the 2-DOF direct-drive robot. Since the kinematics of the parallel-type robot is complex, the computation time for representing it in the virtual environment is considerably long. An 8255 digital input/output (I/O) card is responsible for the position data transmission between the 2-DOF direct-drive robot and the parallel robot. The position data of the parallel robot are sent to the control computer of the 2-DOF direct-drive robot, while the position data of the 2-DOF direct-drive robot are obtained from the PCL-833 quadrature encoder card. Based on the obtained encoder data, it is possible to determine whether the virtual paddle and the tennis ball contacted each other or not. If the virtual paddle of the parallel robot is touched, the virtual force is sent to the control computer of the parallel robot by the control computer of the 2-DOF direct-drive robot. If the virtual paddle of the 2-DOF direct-drive robot is touched, the virtual force is sent back to itself in real-time. In this virtual process, the position data of the virtual environment are sent to a 486 DX2 66 personal computer for real-time animation.

#### 4.2 Dynamic model of the virtual environment

An application for haptic devices based on virtual reality is explored in this study. The virtual environment developed in this study describes the dynamic of a ball, a paddle and walls in the virtual space. Thus, it is mainly composed of the paddle (active object), the ball (passive object), the walls (only in a single robot simulation) and other parameters such as the spring constant and the damper constant. It is no surprise that the ball and the paddle are the main focus. The paddle is a virtual tool that is conceptually linked to the actual robot manipulator handled by the human operator. According to
Newton’s law of motion, a ball moves freely in space unless an external force is applied to it; namely its acceleration is always zero. After the ball is hit, it moves along a straight line to the wall at a constant speed. Figure 11 shows the virtual environment.

The paddle consists of a spring and a damper. According to Newton’s second law of motion, the above system can be described as

$$\sum F = ma$$

(10)

To derive the dynamic model describing paddle 1 in Fig. 11, equation (10) can be substituted into the virtual mass–damper–spring system to obtain

$$m\ddot{y} = -b_1(\dot{y} - \dot{p}_1) - k_1(y - p_1)$$

(11)

where $p_1$ is the displacement of paddle 1, $y$ the displacement of a ball, $m$ the mass of a ball, $k_1$ the spring constant and $b_1$ the damper constant.

In this study, one constraint is added to equation (11). The unilateral constraint $\gamma_1$ ensures that the force to the end-effector is exerted only when the ball and the paddle touch each other. Similarly, $\gamma_1$ ensures that the damper does not exert any force during the period that the ball does not have any contact with the paddle. This constraint is defined as

$$m\ddot{y} = a_1[\gamma_1 b_1(\dot{y} - \dot{p}_1) - k_1(y - p_1)]$$

(12)

where

$$a_1 = \begin{cases} 1, & y < p_1 \\ 0, & y \geq p_1 \end{cases}$$

$$\gamma_1 = \begin{cases} 1, & \dot{y} > \dot{p}_1 \\ 0, & \dot{y} \leq \dot{p}_1 \end{cases}$$

Taking the Laplace transform of equation (12) yields

$$(ms^2 + a_1\gamma_1 b_1 s + a_1 k_1)Y(s) = (a_1\gamma_1 b_1 s + a_1 k_1)P_1(s)$$

(13)

Therefore the transfer function between $Y(s)$ and $P_1(s)$ is

$$T_{p_1}^e(s) = \frac{Y(s)}{P_1(s)} = \frac{a_1(\gamma_1 b_1 s + k_1)}{ms^2 + a_1\gamma_1 b_1 s + a_1 k_1}$$

(14)

Based on equation (14),

$$f_e = m\ddot{y} = a_1[\gamma_1 b_1(\dot{y} - \dot{p}_1) - k_1(y - p_1)]$$

(15)

In addition, according to Fig. 3,

$$f_e = T_{p_1}$$

(16)

Hence, the transfer function between $f_e$ and $p_1$ can be obtained as

$$T_e = \frac{ms^2a_1(\gamma_1 b_1 s + k_1)}{ms^2 + a_1\gamma_1 b_1 s + a_1 k_1}$$

(17)

where $T_e$ is the dynamic model of the virtual environment.
Similarly, the transfer function for paddle 2 \( p_2 \) in Fig. 11 can be derived, which is

\[
T_e = \frac{ms^2\gamma_2b_2s + k_2}{ms^2 + \alpha_2\gamma_2b_2s + \alpha_2k_2}
\]  

(18)

where

\[
\alpha_2 = \begin{cases} 
1, & y > p_2 \\
0, & y \leq p_2 
\end{cases}
\]

\[
\gamma_2 = \begin{cases} 
1, & \dot{y} < p_2 \\
0, & \dot{y} \geq p_2 
\end{cases}
\]

5 EXPERIMENTAL RESULTS

To evaluate the performance of the proposed approach, two robots are controlled to execute a single ball-game program. The resulting force curves for these two robots are recorded and analysed.

To avoid the performance degrading due to heavy data transmission, a 486DX2-66 personal computer is responsible for all the real-time animation communication tasks. Experimental results for a single ball-game are illustrated in Figs 12 to 17.

In the first experiment, the parameters of the virtual environment are set to \( k = 60 \text{ (N/m)}, b = 0 \text{ (N s/m)} \) and

![Fig. 12](image1.png)

**Fig. 12** Force curve of the 2-DOF direct-drive robot with \( k = 60 \text{ N/m}, b = 0 \text{ N s/m} \) and \( m = 0.4 \text{ kg} \)

![Fig. 13](image2.png)

**Fig. 13** Force curve of the parallel robot with \( k = 60 \text{ N/m}, b = 0 \text{ N s/m} \) and \( m = 0.4 \text{ kg} \)
According to Fig. 12, when the paddle hits the ball in the virtual environment, it is found that the force $f_e$ (red line) imposed on the handler actuator by the environment is close to the force imposed on the haptic device $f_h$ (blue line). For the parallel robot, since it is non-back-drivable, the operator’s forces must be converted into motion commands first, and then the robot moves by following the motion commands. Hence in Fig. 13, its force curve is somewhat discontinuous. When the value of parameter $k$ is changed to 150 N/m, it is found that the time duration for the ball touching the paddle is short and the maximal contact force is large (Figs 14 and 15). In this case, the operator may feel as if a hard paddle is handled. These results agree with the theoretic analysis. In the third experiment, damping effects ($b = 6 \text{ Ns/m}$) are added to the system and the resulting force change is shown in Fig. 16. Because of

$m = 0.4 (\text{kg})$. According to Fig. 12, when the paddle hits the ball in the virtual environment, it is found that the force $f_e$ (red line) imposed on the handler actuator by the environment is close to the force imposed on the haptic device $f_h$ (blue line). For the parallel robot, since it is non-back-drivable, the operator’s forces must be converted into motion commands first, and then the robot moves by following the motion commands. Hence in Fig. 13, its force curve is somewhat discontinuous. When the value of parameter $k$ is changed to 150 N/m, it is found that the time duration for the ball touching the paddle is short and the maximal contact force is large (Figs 14 and 15). In this case, the operator may feel as if a hard paddle is handled. These results agree with the theoretic analysis. In the third experiment, damping effects ($b = 6 \text{ Ns/m}$) are added to the system and the resulting force change is shown in Fig. 16. Because of

$m = 0.4 (\text{kg})$. According to Fig. 12, when the paddle hits the ball in the virtual environment, it is found that the force $f_e$ (red line) imposed on the handler actuator by the environment is close to the force imposed on the haptic device $f_h$ (blue line). For the parallel robot, since it is non-back-drivable, the operator’s forces must be converted into motion commands first, and then the robot moves by following the motion commands. Hence in Fig. 13, its force curve is somewhat discontinuous. When the value of parameter $k$ is changed to 150 N/m, it is found that the time duration for the ball touching the paddle is short and the maximal contact force is large (Figs 14 and 15). In this case, the operator may feel as if a hard paddle is handled. These results agree with the theoretic analysis. In the third experiment, damping effects ($b = 6 \text{ Ns/m}$) are added to the system and the resulting force change is shown in Fig. 16. Because of
the added damper, the force curve exhibits a discontinuity. Especially for the parallel robot shown in Fig. 17, the error between \( f_d \) (red line) and \( f_h \) (blue line) is larger than that for the 2-DOF direct-drive robot. The reason is that the restriction on the hardware structure of the robot does not allow the robot to have a drastic change in the force curve.

In the experiment, the ball will move forward and touch the paddle of the 2-DOF direct-drive robot first; thus the fictitious force of the 2-DOF direct-drive robot is generated more quickly than that of the parallel robot. In addition, the force curve of the 2-DOF direct-drive robot is smoother and the resulting error is smaller compared with that of the parallel robot. Note that the data for constructing a virtual reality system are calculated via the control computer of the 2-DOF direct-drive robot. Because the sampling times for the two computers (one for the parallel robot and the other for the 2-DOF direct-drive robot) are different, some time delay will result during data transmission. Therefore, the parallel robot’s force curve is discontinuous and its associated error is large.

6 CONCLUSION

A virtual tennis entertainment system with haptic behaviour is developed in this study. A 2-DOF direct-drive robot and a parallel robot are used as the haptic devices, where a virtual reality system with haptic behaviour is constructed to simulate a tennis ball hitting system. The main components of the virtual reality system developed in this study include the user interface, dynamic simulations and the robot control scheme. The user interface is implemented using a combination of a virtual environment and a standard graphical user interface. Telerobotics for the tennis ball hitting system and the environment are simulated by the haptic virtual system, which allows the operator to sense the actual force feedback from a virtual environment as would be experienced from a real environment. Experiments for hitting a tennis ball with variable damper ratios are conducted to validate the theoretical developments. It is found that the experimental results are in a good agreement with the theoretical analysis.

REFERENCES