A Remote Handling Rate-Position Controller for Telemanipulating in a Large Workspace

Jorge Barrio^a, Manuel Ferre^{a,*}, Francisco Suárez-Ruiz^a, Rafael Aracil^a

^a*Centre for Automation and Robotics UPM-CSIC
Calle José Gutiérrez Abascal, n.2, 28006 Madrid, Spain*

Abstract

This paper presents a new Haptic Rate-Position controller, which allows manipulating a slave robot in a large workspace using a small haptic device. This control algorithm is very effective when the master device is much smaller than the slave device. Haptic information is displayed to the user so as to be informed when a change in the operation mode occurs. This controller allows performing tasks in a large remote workspace by using a haptic device with a reduced workspace such as Phantom. Experimental results have been carried out using a slave robot from Kraft Telerobotics and a commercial haptic interface as a master device. A curvature path following task has been simulated using the proposed controller which was compared with the force-position control algorithm. Results obtained show that higher accuracy is obtained when the proposed method is used, spending a similar amount of time to perform the task.

Keywords: remote handling, telemanipulation, haptic interfaces

1. Introduction

Previous works [1, 2] have found that depending on the requirements in a teleoperation task, there is a specific configuration that offers better performance. Two control modes are usually used for guiding a remote robot: position control and rate control.

Several works have been carried out to determine how task performance is affected by the control mode. In [3] the authors found out that the position control can be 1.5 times faster than the rate control when the master and slave workspaces are similar. In contrast, the rate control reaches better performance when the slave workspace is larger than the master's.

In general, position control has been proven suitable for tasks where short and precise movements are involved. Moreover, rate control has shown better performance for tasks that involve long and precise movements in an extremely rigid environment [4]. Manual operation of a crane can be an example of rate control. The crane itself is the slave device which is commanded by the operator using several joysticks (usually one for degree of freedom). These joysticks serve as master devices. Movements of the joysticks define the speed and direction of the different crane degrees of freedom. On the other hand, position control is frequently applied in robotics applications where movements of the slave are expected to imitate the movements

executed by the master device. As mentioned, this kind of control is better when the master and slave have similar workspace, which means a direct kinematic relation between the master and slave devices.

At the present time, the current commercialized teleoperation systems do not allow combining position and rate control. In fact, when there is a substantial difference between the device workspace, rate control or position control workspace indexing is used. Indexing the workspace of the master consists of performing unlinking when the master reaches its mechanical limit. The master device is then relocated to a new position where it will permit the guidance process to continue. The problem with indexing is that it generates disorientation on the operator due to the changes in the references frames. Productivity of the system is thereby affected due to the downtime in getting accustomed to the new references.

Although some rate-position approaches have already been developed for mobile robots [5] or virtual haptic applications [6], there is no development specially designed for telemanipulation systems. Other typical scaling and indexing methods require the user to press a button in order to swap from one method to another, in contrast with the proposed algorithm. Due to these features, the proposed method is considered more intuitive for users since they can shift between control modes more naturally.

The present paper is organized as follows; Section 2 explains the proposed method in detail, Section 3 describes the developed test-bed for assessing the method, Section 4 presents the results obtained from the experiments carried out and finally, conclusions are provided in Section 5.

*Corresponding author

Email addresses: jordi.barrio@upm.es (Jorge Barrio),
m.ferre@upm.es (Manuel Ferre), fa.suarez@upm.es (Francisco
Suárez-Ruiz), rafael.aracil@upm.es (Rafael Aracil)

2. Description of the Rate-Position Algorithm

A new algorithm which is able to swap from position control to rate control was designed. Only haptic information is used for informing the operator about the transitions from position to rate control. Pressing a button while tasks are carried out is then not required.

Haptic workspace has been divided into position and rate control areas as shown in Fig. 1. It means that accuracy is obtained when the slave robot is manipulated using the position control and large displacements can be carried out using the rate control. Therefore, accuracy can be gained in a large workspace by using a small haptic master.

Figure 1: Definition of the position and rate control areas within a haptic interface

2.1. States of the Control Algorithm

Different states have been defined in order to implement transitions from position to rate control and vice-versa. In Fig. 2, it is depicted what the main states are and how the controller evolves between them. Some states display haptic information to the user notifying about any change occurred in the control mode. Additionally, state transitions have been defined to assure the system stability.

Figure 2: State machine diagram used for implementing the Position-Rate control

2.1.1. State: 'Go to Center'

This state displays forces to bring the user to the workspace center. It is usually activated after entering the position workspace coming from the rate control area. During this transition, the slave robot holds its position and it is not affected by the master displacements.

In addition, this state is also important for the initialization of the system, since it allows to define the zero point of the master. During this initialization, the master is held in the center for around 5s in order for the user to know where the zero point is.

Once the haptic device reaches the workspace center, this state is held for 0.5s in order to stabilize the master device and synchronise both robots. As synchronization error is close to zero, the controller automatically goes to the next state: *Position Control*.

2.1.2. State: 'Position Control'

In this state the user can manipulate the slave robot in position and perceive the interaction forces with the environment. The accuracy of the movements can be adjusted according to the features of both robots and the task.

The controller checks whether the position of the master device is within the position control area. The distance from the zero point of the master device to the current position is calculated and compared with the radius of position workspace, see equation (1). In case the user goes beyond the position workspace, the controller moves to the next state: *Vibration phase*.

$$\begin{aligned} d &= \sqrt{x^2 + y^2 + z^2} \\ d &\leq R_{pos} \end{aligned} \quad (1)$$

2.1.3. State: 'Vibratory phase'

This *vibratory phase* informs the user when a transition from position to rate control occurs. It generates a vibratory stimulus to inform the user that a new operation mode is activated. The use of haptic information avoids using buttons to switch from position to rate control, making the teleoperation more natural. Furthermore, the vibratory stimulus plays a key role since the user knows that a new operation mode is being used, avoiding unexpected behaviours of the system. A damped oscillatory signal is used for generating the vibratory stimulus, where A is the amplitude, C the decay rate parameter and w is the angular frequency of the signal.

$$F_{vib} = A \cdot e^{-C \cdot t} \cdot \sin(wt) \quad (2)$$

2.1.4. State: 'Rate Control'

This state allows telemanipulating the slave robot using rate commands. A force feedback proportional to the rate command is displayed to the user according to equation (3). Parameter K describes the spring used for displaying force feedback, d is the current position distance

measured from the center of position workspace, R_{pos} is the radius of the sphere which bounds the position control and D_{hys} is the distance which is considered as hysteresis area.

According to the equation (4), the further the user is from the position sphere, the faster the slave robot moves.

$$F_{rate} = \begin{cases} K_f(d - R_{pos} - D_{hys}) & \text{if } d > R_{pos} + D_{hys} \\ 0 & \text{if } d \leq R_{pos} + D_{hys} \end{cases} \quad (3)$$

$$\dot{x}_{rate} = K_{vel}(d - R_{pos} - D_{hys}) \quad (4)$$

As you can see from Fig. 1 a hysteresis area is defined in order to avoid force glitches in accidentally changing the controller from rate control to position control or vice-versa. In this area, no velocity commands are sent to the slave robot and no force feedback is displayed to the user.

Controlling the slave robot in velocity requires the user to keep the master robot in the rate control area. In case a change of direction in velocity is required, the user will have to move the master robot out of the sphere position.

2.1.5. State: 'Rate Collision'

A collision with the remote environment when the slave robot is guided in rate control mode can be very dangerous if there is any kind of force feedback which makes it possible to implement safety measures or inform the man-in-the-loop that a collision has occurred. In case of rate collision, typically, the rate commands, which are being sent to the slave, are immediately stopped and a significant opposition force is displayed to the user so as to be led to the center of the workspace to come back to the position control mode.

The main goal of this state is to bring the user to the workspace center so that the controller can be swapped to a position control mode operation. This will allow interacting with the remote environment more safely. Thus, when the haptic device is next to the center of the workspace area, the controller will move to the next state 'Go to Center', before reaching a position control mode.

3. Experimental Design

Figure 3: Experimental set-up

A test-bed has been designed in order to evaluate the proposed rate-position algorithm. A commercial haptic

device, Phantom Omni, has been used for guiding a GRIPS slave robot of Kraft Telerobotics Inc. The slave robot has a complex configuration and numerical methods has been adopted to solve the inverse kinematics [7, 8]. The master movements are scaled 1:3 in relation to the slave when the rate-position controller is activated. A simple scenario has been developed where the slave robot has to follow the centreline of a trail, see Fig. 3. This scenario requires accurate and precise movements, which is an excellent manner to assess the performance of the proposed algorithm. Besides, the performance of welding and cutting with remote handling technologies are of interest for on-site manufacturing and maintenance of the thick-wall structure of ITER [9].

4. Results obtained

This section shows the results obtained when the new rate-position control algorithm was implemented. A curvature following path task was carried out using both the new rate-position controller and force-position control algorithms. The user had to follow a bend line with the maximum accuracy. The results obtained are compared between them in order to properly assess the real advantages provided by each method.

Since usually these kinds of telemanipulation tasks are carried out for advanced users with the proper training, this experiment was also performed for an experienced teleoperator from our lab. This person performed the same task ten times with each control algorithm. The most important parameters which describe the task such as the time spent and the trajectory were recorded, making it possible to assess the accuracy achieved in each case.

4.1. Accuracy and trajectory analysis

Figure 4: Slave robot trajectories for ten trials. The reference trajectory is: **—**

Fig. 4a shows the trajectories when the rate-position control is used. They are very close to the reference curve which should be followed by the operator. In addition, the trajectories are very smooth without jumps or any movements away from the reference.

On the contrary, the path trajectories obtained when the task is performed using the force-position control algorithm have more irregularities and oscillations, see Fig. 4b,

than the ones obtained with the rate-position control, see Fig. 4a. In general, the force-position control algorithm shows that all the trajectories are further away from the reference line, displaying bigger errors and much less accuracy.

A detailed analysis of the accuracy obtained by each control algorithm has been carried out. The average error obtained with each method is depicted in Fig. 5. The average error of the rate-position control method is 4.03mm which is significantly lower than the average error of 11.6mm obtained in the force-position control. It means that the error generated using the new rate-position controller is a 34.64% of the error shown in the force-position controller. In addition to this, the standard deviation of the rate-position control method is 4.2mm in comparison with the 9.9mm of the force-position control mode.

These results clearly show the new rate-position algorithm is better than the classic force-position control in terms of accuracy.

Figure 5: Comparison of the error and time spent with Rate-Position control vs Force-Position control

4.2. Time spent

Fig. 5 also shows the average time required depending on the control algorithm used. Although the average time is very similar regardless of the controller used, the force-control algorithm is slightly faster 36.77s in comparison to the 38.15s required to complete the task when the new rate-position method is used. The standard deviation is also higher in the rate-position controller 7.05s versus 4.93s in the position controller.

According to the results obtained, it seems to be that the time required to swap from some internal states of rate-position control algorithm to another explain why a bit more time is necessary to finalise and complete the tasks properly. Some internal states require to resynchronise master and slave devices properly before continuing with the normal operation. This can take some time affecting the general velocity to perform a task.

5. Conclusions

A haptic rate-position algorithm is presented in this paper. This method is especially designed for the teleoperation of a slave robot in a large workspace, using a master robot with a much smaller workspace. The master

robot is able to control the slave robot by using both position and rate control modes, swapping from one method to another. The user is aware of the control mode used due to the haptic information displayed.

A welding task has been emulated following a bend track with a slave robot. The same experiment has been carried out several times using both the proposed rate-control algorithm and classic position controller. Analysing the result obtained, the rate-position control algorithm has proven to be 65.36% much accurate than the classic control algorithm. It is due to the fact that two control modes are defined (position and rate control) in the proposed method to control the entire workspace. However, in terms of time, the classic position control seems to be slightly faster since it is not necessary to change between two operation modes.

To conclude, the main strengths of this method are: the possibility to control a large workspace using a haptic desktop master device using a reduced space and the great accuracy achieved due to the two operation modes.

Acknowledgement

This work was carried out under the EFDA Goal Oriented Training Programme (WP10-GOT- GOTRH) and the financial support of TECHNOFUSION programme from Comunidad de Madrid, which are greatly acknowledged. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

References

- [1] A. Kazi, Operator Performance in Surgical Telem Manipulation, Presence: Teleoperators and Virtual Environments 10 (5) (2001) 495–510.
- [2] O. Ben-Porat, et al., Control Design and Task Performance in Endoscopic Teleoperation, Presence: Teleoperators and Virtual Environments 9 (3) (2000) 256–267.
- [3] W. S. Kim, F. Tendick, S. R. Ellis, L. W. Stark, A comparison of position and rate control for telem manipulations with consideration of manipulator system dynamics, IEEE Journal on Robotics and Automation 3 (5) (1987) 426–436.
- [4] M. E. Kontz, W. J. Book, Position/rate haptic control of a hydraulic forklift, in: Proceedings ASME International Mechanical Engineering Congress and Exposition, Washington, DC, USA, 2003, pp. 801–808.
- [5] I. Farkhatdinov, J.-H. Ryu, Hybrid position-position and position-speed command strategy for the bilateral teleoperation of a mobile robot, in: Control, Automation and Systems, 2007. International Conference on, 2007, pp. 2442–2447.
- [6] L. Dominjon, et al., The "bubble" technique: Interacting with large virtual environments using haptic devices with limited workspace, in: Eurohaptics Conference, 2005, pp. 639–640.
- [7] M. Meredith, S. Maddock, Adapting motion capture data using weighted real-time inverse kinematics, Comput. Entertain. 3 (1) (2005) 5–5.
- [8] K. Waldron, J. Schmiedeler, Kinematics, in: B. Siciliano, O. Khatib (Eds.), Springer Handbook of Robotics, Springer, 2008, Ch. 1, pp. 9–33.
- [9] M. Onozuka, et al., Manufacturing and maintenance technologies developed for a thick-wall structure of the ITER vacuum vessel, Fusion Engineering and Design 55 (4) (2001) 397–410.