Robotica (2005) volume 23, pp. 177–187. © 2005 Cambridge University Press
doi:10.1017/S0263574704000608 Printed in the United Kingdom

Magister-P; a 6-URS parallel haptic device with open control architecture

Jose M. Sabater*, Jose M. Azorin*, Rafael Aracil†
and Roque J. Saltaren†

(Received in Final Form: June 2, 2004)

SUMMARY

This paper presents a new haptic device based on a parallel structure that can be used as a master interface in a teleoperation or haptic control architecture. The basic idea of a haptic device is to serve force and/or position reflection to the operator; at the same time that is being used by the human operator to input the required commands. The original mechanical structure of the presented system implies important advantages over other existing devices. The mechanism is a modification of the 6-d.o.f. Gough platform where the linear actuators have been replaced by cable-driven pantographs. Avoiding the use of reduction gears by means of cable transmission allows a wide sensing bandwidth. Some experimental indices comparing the performance of the presented device are presented. The paper shows the geometrical model and the kinematic analysis used on the control algorithms of this interface. The hardware and software architectures used on the system, and the control schemes implemented on a multi-axis board, are detailed. This setup provides an open control architecture that allows the implementation and experimentation of several bilateral control schemes. The integration of the haptic device in a teleoperation simulator is shown. This simulator includes virtual robotic slaves and its dynamic interaction with the virtual environment. Finally, the results obtained in the virtual objects manipulation experiments are shown. A classical force-position bilateral control scheme was used for these experiments.

KEYWORDS: Haptic devices; Parallel robots; Bilateral control; Teleoperation; Mechanical design.

1. INTRODUCTION

In a telerobotic system the operator wants to receive as much useful information of the remote environment as possible. The interaction between operators and robots to carry out a task was described by Goertz.[1] Several authors, like Hannaford,[2] Burdea,[3] or Sayers[4] have shown that if the contact force of the slave robot with the environment is reflected to the operator, the tasks could be more efficiently carried out. Tactile or kinesthetic exploration of remote or virtual environments need a mechanical interface that allows the operator to interact with the remote zone. The haptic interfaces are used wherever the using of only visual feedback can introduce unacceptable manipulation errors. Besides, one of the main objective of a haptic device is to be transparent to the operator, so the own inertia, fricction and weight of the device must not be felt when using the interface.[5]

On other hand, the increase of the computational capacity of new processors has renewed research interest in parallel robots. The special characteristics of this kind of robots make them suitable for many applications, as for example, machine tools,[6] robot manipulators[7] or climbing service robots.[8] Especially interesting is the application of parallel structures as haptic devices. Several examples, like the Gough based platforms for surgery applications,[9] 3 degrees of freedom spherical mechanisms,[10] cable driven mechanisms[11] or mechanisms with legs of several links,[12] HapticMaster[13] have been presented in the last few years. All these devices try to exploit the special characteristics of parallel structures, like low inertia, high rigidity, compactness, precise resolution and high load/power ratio, as compared with serial mechanisms. However, some of these parallel haptic devices that have been developed still have disadvantages, such as small workspace or narrow bandwidth.

The haptic interface presented in this paper, named Magister-P (Spanish acronym: *Maestro Genérico para Interfaz en estructuras Teleoperadas*) (Fig. 1) (Patent N° P200302351), is a modification of a 6 degrees of freedom Gough-Stewart platform where the linear actuators have been replaced by cable-driven pantographs. The movable base of the designed joystick is the inferior one, thus the resulting design becomes an ergonomic device, showing to the operator a free collision workspace (as the mechanical structure stays overhead), and at the same time, the static balancing torques required to “mask to the operator” the gravity effects over its own links are decreased. Besides that, the device shows an equilibrium position when no forces are acting on the system. Because of the special mechanical configuration of the joints, such a haptic device has several advantages: It has 6 programmable degrees of freedom, a large workspace, low inertia, high bandwidth, and easy use. The use of a non slip-cable transmission between the actuator axis and the rotation of one of the links of the pantograph (marked with 1 in Fig. 1) increase the mentioned advantages

* Corresponding author: Jose M. Sabater. E-mail: j.sabater@umh.es
Departamento de Ingenieria de Sistemas Industriales, Universidad Miguel Hernandez, Elche (Alicante), 03202 (Spain).
† DISAM, ETSII, Universidad Politecnica de Madrid, Madrid, 28006 (Spain).

https://doi.org/10.1017/S0263574704000608 Published online by Cambridge University Press

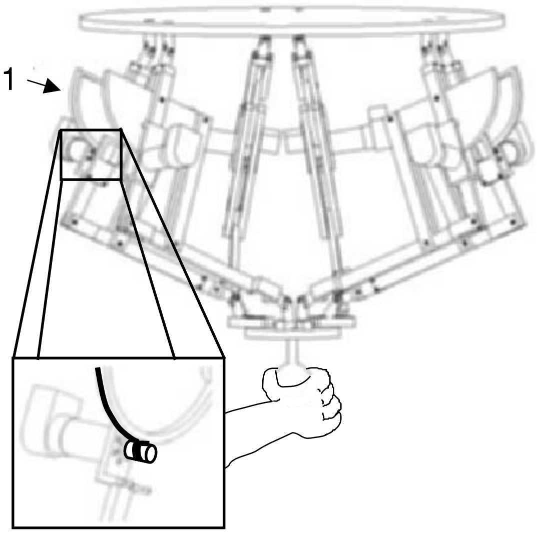


Fig. 1. CAD model of the designed prototype.

of parallel mechanisms as haptic devices, because of the absence of gearbox transmission.

With the aim of testing the designed haptic device, a complete virtual robotic teleoperation environment has been developed. The dynamic model of any robotic slave (serial or parallel) can be modelled on this simulator. The use of virtual simulators provides a powerful tool for the realization of teleoperated tasks. Burdea[3] shows different application areas of the virtual reality with teleoperation environments. For example, geometrical and kinematic simulations allow an increase of the performance of aiding-tools in several applications, like the mechanisms CAD design of Hollerbach et al.,[14] the augmented reality on poor visual feedback environments studied by Lin and Kuo,[15] predictive displays on communication time delays environments[16] or in tools for the training of operators.[17] Nevertheless, adding a full dynamic environment that includes the complete dynamic model of the slave device and its interaction with other objects allows one to realize more complex simulations, for example to follow a path inside an environment where a force field is acting over all the links of the slave. The main characteristic of this simulator is to include an open loop model of the slave, so that any bilateral control teleoperation scheme can be implemented between the master and the slave.

In this work a new open control haptic interface is presented. This interface includes a mechanical device, a hardware setup and the algorithms and libraries developed to use this interface in any haptic or teleoperated system. Besides, an application of peg-in-hole insertion task using this interface with a virtual haptic environment is shown. The arrangement of two open control subsystems (the operator-master subsystem and the slave-environment one) allows an experimentation of control strategies with 6 degrees of freedom devices. In this paper a classical force-position scheme is used.

The remainder of the paper is organized as follows: Firstly, the kinematic analysis of the presented device is presented. The inverse kinematic problem (IKP) and the forward kinematic problem (FKP) are solved, and a geometrical relation between the Cartesian generalized forces and the joint forces is obtained. Section 3 shows the system architecture and explains the implementation of different work-modes of the Magister-P, working as an impedance display (rendering forces to the operator) or working as an admittance display (rendering motion to the operator). Some performance characteristics of the developed interface are shown in section 4. The next section shows one application of the haptic interface working with a teleoperated tasks simulator, when a general scheme of the developed software tool is reviewed. Such simulator software allows one to generate virtual worlds where the dynamic objects can be manipulated, and the reaction forces are calculated. Finally, some results obtained, when using a force-position scheme between the haptic interface and the simulator are shown.

2. KINEMATIC ANALYSIS

The kinematic analysis of the proposed 6-URS (Universal-Rotational-Spherical) platform is based on a modification of the algorithms of the 6-UPS (Universal-Prismatic-Spherical) platform.

2.1. *Framework*

For the kinematic analysis of the platform, only 13 parts have been considered, because the adding of the transmission links of the pantographs does not give additional information to the kinematic model and extends the vectorial equations. This assumption must be reviewed for a dynamical modelling. Fig. 2 shows the used model.

Using Euler parameters to represent the orientation, each body need seven generalized coordinates, leading to 91 generalized coordinates for the 13 parts to completely define the display. The sum of the constraints imposed by the spherical, the rotational and the universal joints, and the constraints imposed by the normalization of the Euler parameters, give a total of 85 constraints. The difference are the degrees of freedom of the mechanism (see table I).

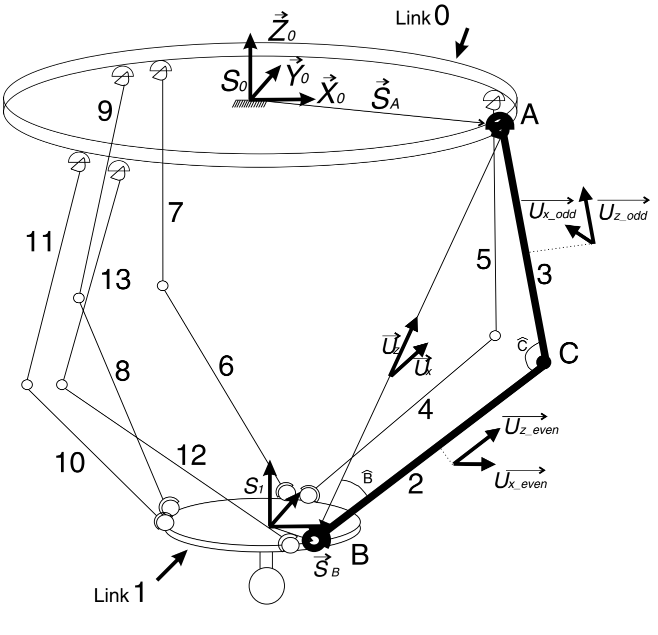


Fig. 2. Geometrical model.

https://doi.org/10.1017/S0263574704000608 Published online by Cambridge University Press

Table I. Kinematic model of 6 URS platform.

Generalized coordinates	
13 bodies	$13 \times 7 = 91$
Constraints	
Spherical	$6 \times 3 = 18$
Universal	$6 \times 4 = 24$
Rotational	$6 \times 5 = 30$
Normalization of Euler parameters	$13 \times 1 = 13$
Sum:	**85**
Degrees of freedom	$91 - 85 = 6$

The movement of the device is defined by the time variation of the angle $\hat{C}$ between the even and odd links. This is equivalent to the imposition of actuators constraints to the rotational joints, hence the six degrees of freedom will be completely determined.

To solve the kinematics of the mechanism, a reference system must be assigned to each link.

The upper base, that is fixed, is named as "*link 0*" and its reference system $\mathbf{S_0}$ (considered inertial). The $\mathbf{S_0^{A_n}}$ vectors localize the application points of the universal joints A_n and are referenced to $\mathbf{S_0}$ by the expression

$$\mathbf{S_0^{A_n}} = \begin{bmatrix} r_{\text{base}} \cos\left((n-1)\frac{\pi}{3} + \frac{\delta_0}{2}\right) \\ r_{\text{base}} \sin\left((n-1)\frac{\pi}{3} + \frac{\delta_0}{2}\right) \\ 0 \end{bmatrix}^T \quad n = 1, 3, 5$$
$$\mathbf{S_0^{A_n}} = \begin{bmatrix} r_{\text{base}} \cos\left((n-2)\frac{\pi}{3} - \frac{\delta_0}{2}\right) \\ r_{\text{base}} \sin\left((n-2)\frac{\pi}{3} - \frac{\delta_0}{2}\right) \\ 0 \end{bmatrix}^T \quad n = 2, 4, 6 \qquad (1)$$

where δ_0 defines the angle between two consecutive joints and r_{base} is the separation radius of the joints from the origin of the reference frame.

The lower base or joystick is named as "*link 1*" and its reference system S_1. The $\mathbf{S_1^{\prime B_n}}$ vectors localize the application points of the spherical joints B_n and are related to $\mathbf{S_1}$ by the expression

$$\mathbf{S_1^{\prime B_n}} = \begin{bmatrix} r_{\text{joystick}} \cos\left((n-2)\frac{\pi}{3} + \frac{\delta_1}{2}\right) \\ r_{\text{joystick}} \sin\left((n-2)\frac{\pi}{3} + \frac{\delta_1}{2}\right) \\ 0 \end{bmatrix}^T \quad n = 1, 3, 5$$
$$\mathbf{S_1^{\prime B_n}} = \begin{bmatrix} r_{\text{joystick}} \cos\left((n-3)\frac{\pi}{3} - \frac{\delta_1}{2}\right) \\ r_{\text{joystick}} \sin\left((n-3)\frac{\pi}{3} - \frac{\delta_1}{2}\right) \\ 0 \end{bmatrix}^T \quad n = 2, 4, 6 \qquad (2)$$

where δ_1 and r_{joystick} are defined as for *link 0*.

2.2. *Inverse kinematics*

To find the unique solution of the IKP problem, the distance between the A_n and B_n anchor points is used.

Given the position and orientation of the "*link 1*" joystick, by a vector $\mathbf{q_1} = [\mathbf{r_1}, \mathbf{p_1}]^T$ where $\mathbf{r_1} = [x_1,\ y_1,\ z_1]^T$ is the Cartesian position and $\mathbf{p_1} = [e_0,\ e_1,\ e_2,\ e_3]^T$ are the Euler parameters, or $\mathbf{q_1} = [\mathbf{r_1},\ \alpha_1,\ \beta_1,\ \gamma_1]^T$ if the orientation is given with the 313 Euler angles. The distance between the universal and spherical joints can easily be obtained by:

$$\mathbf{r_{A_nB_n}} = \mathbf{r_1} + \mathbf{A_1}\mathbf{s_1^{\prime B_n}} - \mathbf{s_0^{B_n}} \qquad (3)$$

where $\mathbf{A_1}$ is the rotation matrix given the orientation of "*link 1*". Getting the norm of $\mathbf{r_{A_nB_n}}$, the solution is shown by equation (4) (see Fig. 2).

$$\hat{C}_n = \arccos\left(\frac{(\overline{BC})^2 + (\overline{AC})^2 - \text{norm}\left(\mathbf{r_{A_nB_n}}\right)^2}{2(\overline{BC})(\overline{AC})}\right) \qquad (4)$$

2.3. *Forward kinematics*

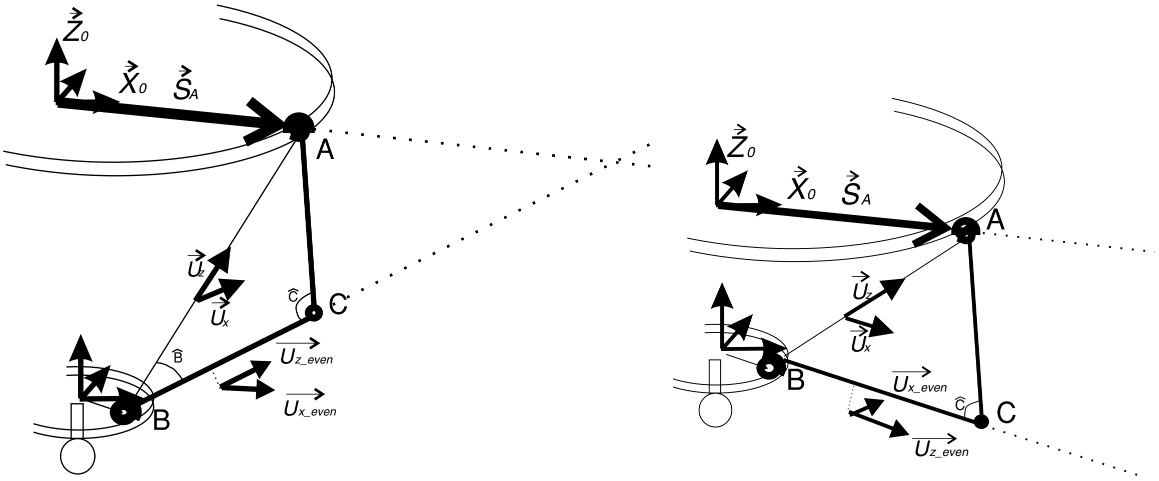
To solve the forward kinematics (FKP) of a URS platform is to establish the relations between the command variables of the angles $\hat{C}$ and the resultant position of the end effector.

Several methods for geometric calculation of forward kinematic of 6 d.o.f. parallel platforms can be found in specific literature. Some of them allow one to obtain the possible solutions through the use of polynomials that result of the geometric modelling of the kinematic chains of the platform, for example, in Merlet[18] the 16 possible solutions for a 6 d.o.f. platform are calculated. Lazard and Merlet,[19] and Danescu and Dahan[20] prove that the direct kinematics of the Gough-Steward platform have 12 possible solutions. Nair[21] suggests a systematic method to obtain the minimal polynomial equations for certain cases of parallel platforms. With this method he obtains a solution of polynomials of 8 degrees, although for a general 6 d.o.f. robot, the method of Nair arrives at polynomials of 144 degrees. For the general case of a 6 d.o.f. platform, Lazard and Merlet prove that the maximum number of solutions is 40. On other hand, Dasgupta[22] suggests an approximation that reduces the problem to three equations plus one more equation of constraints that must be satisfied by the triplet of solutions and must be resolved by numerical methods.

In this section a numerical method based on the initial estimation of the generalized coordinate vector q_i will be proposed. In general, a 6-d.o.f. URS parallel platform is formed by 13 links that constitute the arms and the movable base. Then the generalized coordinates vector will be represented as: $\mathbf{q} = [\mathbf{q_1}, \mathbf{q_2}, \mathbf{q_3}, \ldots, \mathbf{q_{13}}]^T_{91\times1}$, where $\mathbf{q_1}$ is the generalized coordinate system of the end effector and $\mathbf{q_2}, \mathbf{q_3}, \ldots, \mathbf{q_{13}}$ correspond to the links that form the arms. In general, each link is defined by a generalized coordinate system where: $\mathbf{q_i} = [\mathbf{r_{A_iB_i}}, \mathbf{p_i}]^T$ with $\mathbf{r_{A_iB_i}} = [x_i,\ y_i,\ z_i]^T$ and the Euler parameters: $\mathbf{p} = [e_0^i,\ e_1^i,\ e_2^i,\ e_3^i]^T = [e_0, \mathbf{e}^T]$.

The first step to assign the reference systems to the even and odd links, an auxiliary reference system composed by

https://doi.org/10.1017/S0263574704000608 Published online by Cambridge University Press



(a) Case I

(b) Case II

Fig. 3. Two possible configurations for solving the inverse kinematics.

the vectors $\mathbf{u_z}$ and $\mathbf{u_x}$ is obtained.

$$\mathbf{u_z^n} = \frac{\mathbf{r_{A_nB_n}}}{\left\|\mathbf{r_{A_nB_n}}\right\|}, \quad \mathbf{u_x^n} = \frac{\mathbf{u_z^n} \times \mathbf{s_0^{An}}}{\left\|\mathbf{u_z^n} \times \mathbf{s_0^{An}}\right\|} \tag{5}$$

and the angles $\hat{B}_n$ and $\hat{C}_n$ of each arm are solved using the cosine theorem.

Vectors $\mathbf{u_{z\_even}}$ are part of the reference systems of the even links; 2, 4, 6, 8, 10 and 12, and are obtained by a rotation of the vector $\mathbf{u_z}$ around vector $\mathbf{u_x}$.

$$\mathbf{u^n_{z\_even}} = (\mathrm{Rot}(\mathbf{u_x}, (\hat{B}_n)) \times \mathbf{u_z} \tag{6}$$

To obtain the vectors $\mathbf{u_{x\_even}}$, two possible situations must be considered. Fig. 3 shows that the third component of $\mathbf{u_{z\_even}}$ can be positive or negative (considering null the z coordinate of the B_n points), meaning that the vector $\mathbf{u_{x\_even}}$ lays above or beyond the vector $\mathbf{s_0^{An}}$ (whose third component is zero). So the expressions to get $\mathbf{u_{x\_even}}$ are

$$\mathbf{u}^n_{\mathbf{x\_even}} = \frac{\mathbf{u}^n_{\mathbf{z\_even}} \times \mathbf{s_0^{An}}}{\left\|\mathbf{u}^n_{\mathbf{z\_even}} \times \mathbf{s_0^{A_n}}\right\|}, \quad \text{if } \mathbf{u(3)_{z\_even}} \geq 0 \tag{7}$$

or

$$\mathbf{u^n_{x\_even}} = \frac{\mathbf{s_0^{An}} \times \mathbf{u^n_{z\_even}}}{\left\|\mathbf{s_0^{An}} \times \mathbf{u^n_{z\_even}}\right\|}, \quad \text{if } \mathbf{u(3)_{z\_even}} < 0 \tag{8}$$

$$\mathbf{u^n_{y\_even}} = \mathbf{u}^n_{\mathbf{z\_even}} \times \mathbf{u^n_{x\_even}} \tag{9}$$

Vectors $\mathbf{u_{z\_odd}}$ are part of the reference systems of the odds links; 3, 5, 7, 9, 11 and 13, and are obtained rotating $(\pi - \hat{C}_n)$ the vector $\mathbf{u_{z\_even}}$ around vector $\mathbf{u_{x\_even}}$.

$$\mathbf{u^n_{z\_odd}} = (\mathrm{Rot}\ (\mathbf{u_{x\_even}}, (\pi - \hat{C}_n)) \times \mathbf{u_{z\_even}} \tag{10}$$

$$\mathbf{u^n_{x\_odd}} = \frac{\mathbf{u^n_{z\_odd}} \times \mathbf{s_0^{An}}}{\left\|\mathbf{u^n_{z\_odd}} \times \mathbf{s_0^{A_n}}\right\|} \tag{11}$$

$$\mathbf{u^n_{y\_odd}} = \mathbf{u^n_{z\_odd}} \times \mathbf{u^n_{x\_odd}} \tag{12}$$

Once each link has its own reference system, the description of the kinematic chain of the 6-URS device, is based on the constraint vector:

$$\boldsymbol{\phi}(\mathbf{q}, t) = \begin{bmatrix} \boldsymbol{\phi}^k(\mathbf{q}) \\ \boldsymbol{\phi}^D(\mathbf{q}, \mathbf{C}(t)) \\ \boldsymbol{\phi}^P(\mathbf{q}) \end{bmatrix}_{91\times 1} = 0 \tag{13}$$

where $\boldsymbol{\phi}^k(\mathbf{q})_{72\times 1} = 0$ is the vector of the 72 holonomic constraints imposed by the universal (4), rotational (5) and spherical (3) joints. $\boldsymbol{\phi}^D(\mathbf{q}, t)_{6\times 1}$ is a vector of 6 constraints imposed by the actuators, that in this case are functions of the command joint variables for which the forward kinematics will be calculated. $\boldsymbol{\phi}^P(\mathbf{q})_{13\times 1}$ is a vector of 13 constraints due to the norm of the Euler parameters.

Besides the constraint vector, the differential relation between the generalized coordinates is expressed by the Jacobian matrix of the constraint vector, that in this device is given by the following 91 × 91 matrix.

$$\boldsymbol{\phi}_q(\mathbf{q}) = \begin{bmatrix} \boldsymbol{\phi}_q^K \\ \boldsymbol{\phi}_q^D \\ \boldsymbol{\phi}_q^P \end{bmatrix}_{91\times 91} \tag{14}$$

where $\boldsymbol{\phi}_q^K$ are the differential terms of the holonomic constraints, $\boldsymbol{\phi}_q^D$ are the differential terms of the constraint actuators, and $\boldsymbol{\phi}_q^P$ are the normalization terms. All the terms of this Jacobian matrix are calculated using the following primitives:

$$\begin{aligned} \boldsymbol{\phi}_{qi}^k &= \left[\boldsymbol{\phi}_{ri}^k, \boldsymbol{\phi}_{pi}^k\right] = \left[\boldsymbol{\phi}_{ri}^k, 2\boldsymbol{\phi}_{\pi i}^k \mathbf{G_i}\right] \\ \boldsymbol{\phi}_{qi}^D &= \left[\boldsymbol{\phi}_{ri}^D, \boldsymbol{\phi}_{pi}^D\right] = \left[\boldsymbol{\phi}_{ri}^D, 2\boldsymbol{\phi}_{\pi i}^D \mathbf{G_i}\right] \\ \boldsymbol{\phi}_{qi}^P &= \left[\mathbf{0}, 2\mathbf{p}_\mathbf{i}^T\right] \end{aligned} \tag{15}$$

where the $\mathbf{G_i}$ is the Euler matrix defined in Haug[23] as $\mathbf{G_i} = [-e, -\tilde{\mathbf{e}} + e_0\mathbf{I}_{3\times 3}]$, and the terms for the displacement (ϕ_r) and orientation (ϕ_π) are obtained with the expressions of the

https://doi.org/10.1017/S0263574704000608 Published online by Cambridge University Press

Table II. Differential terms.

	ϕ_{ri}	ϕ_{rj}	$\phi_{\pi_i'}$	$\phi_{\pi_j'}$
$\phi^{d1}(a_i, a_j)$	0	0	$a_j'^T A_j^T A_i \tilde{a}_i'$	$a_i'^T A_i^T A_j \tilde{a}_j'$
$\phi^{s}(P_i, P_j)$	$-$ I	I	$A_i \tilde{s}_i'^P$	$-A_j \tilde{s}_j'^P$

table II for the case of a parallelism constraint $\boldsymbol{\phi}^{d1}(\mathbf{a_i}, \mathbf{a_j})$ or a spherical constraint $\boldsymbol{\phi}^{s}(\mathbf{P_i}, \mathbf{P_j})$.

As mentioned before, to calculate the forward kinematic solution we start from an approximated generalized coordinates vector $\mathbf{q_i}$, and the command values $\hat{\mathbf{C}}_\mathbf{n}(t)$ (in case the forward kinematic solution depends only on the command variable). For these effects, it is common to use the Newton-Raphson method.

$$\begin{aligned} \boldsymbol{\phi}_q \triangle \mathbf{q}^{(\mathrm{j})} &= -\boldsymbol{\phi}\left(\mathbf{q}^{(\mathrm{j})}, t\right) \\ \mathbf{q}^{(\mathrm{j+1})} &= \mathbf{q}^{(\mathrm{j})} + \triangle \mathbf{q}^{(\mathrm{j})} \end{aligned} \tag{16}$$

where $\boldsymbol{\phi}_q$ is the Jacobian of the vector of constraints and $\mathbf{q}^{(\mathrm{j+1})}$ is the next time step of the FKP problem.

2.4. *Cartesian forces to joint forces mapping*

In order to obtain the forces supplied to the operator, the relation between the joint forces exerted by the actuators and the Cartesian force supplied at the end-effector ("*link1*") must be known. This relation must be evaluated for each pose of the device.

Using the reference frames explained before, if $[\mathbf{f}, \mathbf{n}]^T$ are the forces and torques on the "*link 1*", and $\mathbf{N_n}$ are the torques generated by the actuators on the rotational joints,

$$\mathbf{f} = \sum_{n=1}^{6} \mathbf{u_{y\_even}} \times \overline{BC} \times \mathbf{N_n} \tag{17}$$

$$\mathbf{n} = \sum_{n=1}^{6} \left(s_1^{Bn} \mathbf{u_{y\_even}}\right) \times \overline{BC} \times \mathbf{N_n} \tag{18}$$

the relation between the applied forces on the actuators and the Cartesian forces on "*link 1*" is expressed in equation (19).

$$\begin{aligned} \begin{bmatrix} \mathbf{f} \\ \mathbf{n} \end{bmatrix} &= (\mathbf{J}^T)^{-1}\mathbf{F} \\ &= \begin{bmatrix} \mathbf{u}^1_{\mathbf{y\_even}} & \mathbf{u}^2_{\mathbf{y\_even}} & \cdots & \mathbf{u}^n_{\mathbf{y\_even}} \\ \mathbf{s}_1^{\mathbf{B1}} \times \mathbf{u}^1_{\mathbf{y\_even}} & \mathbf{s}_1^{\mathbf{B2}} \times \mathbf{u}^2_{\mathbf{y\_even}} & \cdots & \mathbf{s}_1^{\mathbf{Bn}} \times \mathbf{u}^n_{\mathbf{y\_even}} \end{bmatrix} \mathbf{F} \end{aligned} \tag{19}$$

3. OPEN CONTROL ARCHITECTURE

Some of the more relevant characteristics of the designed device are the ability of working as an impedance or as an admittance display, and to present an open control programmable architecture that allows the implementing of any bilateral control scheme. To obtain those characteristics, a open hardware architecture and several software tools have been developed. This section shows the functioning of the Magister-P and the system architecture of the whole interface.

3.1. *Hardware scheme*

The system hardware architecture has been built on a general purpose multiaxis board (dSpace model DS1103PPC),[24] that is equipped with a real time Motorola PowerPC 604e processor and with a slave-DSP subsystem based on the Texas Instruments TMS320F240 DSP microcontroller. The board is allocated on the ISA bus of the workstation computer. All the simulation software and the visualization interface are programmed over the workstation PC. Communications between the PC-Workstation and the PowerPC 604e microprocessor are made using the dSpace Clib libraries, which control the access to the DS1103PPC memory. Both the lecture of the optical encoders, the calculation of the control signal, and the management of the linear servo amplifiers are carried out in the PowerPC microprocessor. Fig. 4 shows the scheme of the designed architecture for the control of the haptic interface.

The six 24 V DC motors are managed by 4-Quadrants linear controller power amplifiers. Those amplifiers are set on current control configuration mode (torque control), so they supply a proportional torque to the command voltage.

The forces and torques applied by the operator are obtained by a six-d.o.f. force/torque sensor (JR3 sensor) located on the end-effector (joystick link), and the signal is read by the own communication board.

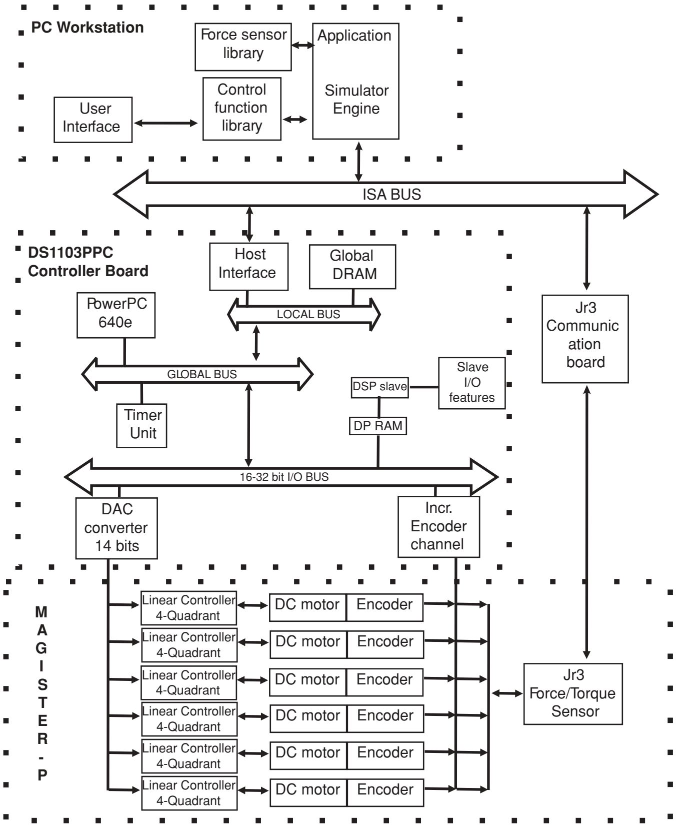


Fig. 4. Magister-P Open Control Architecture.

https://doi.org/10.1017/S0263574704000608 Published online by Cambridge University Press

3.2. Software tools

An ANSI-C library (MBlib – multibody library) has been developed for the implementation of the control algorithms in the multiaxis board. The kinematic model explained in section 2 is programmed in this library. All the code for the necessary matricial operations of the previous algorithms have been also included in MBlib library. Additional *C* or *C*++ libraries cannot be used because of the PowerPC compiler tool.

Programming and management of the control algorithms on the DS1103PPC board is done using the dSpace RTlib libraries and the cited MBlib libraries.

RTlib. The resources of the DS1103PPC board are managed by the dSpace Ind. RTlib library. The functions are written on *ANSI-C* and control the configuration and the read/write operations on the several resources of the board.

MBlib. The MBlib library holds the necessary matricial functions and the joint primitives of the multibody algorithms that solve the IKP and the FKP problems of the 6-URS platforms. The Jacobian matrix of the equation (19) is also computed. The library is written in *ANSI-C*.

For the communication between an external application and the hardware of the DS1103PPC board, the Clib library is needed. This library is proportioned by the board manufacturer. The communication is managed by the DSP slave of the board. The process to establish such communication channel is the next one:

(i) Register a client application on the board DSP.
(ii) Select a board where the real-time task is being executed.
(iii) Allocate board resources for the interchange of variables between the dSpace board and the client application.
(iv) Access the real-time board variables using the existent functions. (The names of the variables must mate with the ones programmed on the board).
(v) Free the assigned board resources.
(vi) Unregister the application.

Finally, some tools for the compilation (Microtec PowerPC C Compiler) and for the program management on the DS1103PPC board (Down1103 program from dSpace Inc.) are needed.

3.3. Impedance mode

Impedance displays are defined as devices that can supply information about its motion and at the same time they can supply to the operator a vector of generalized forces. To use a mechanical device as an impedance display, the next requirements must be fulfilled:

- to obtain the position and orientation expressed on the Cartesian space, the device must have position sensors, and the forward kinematic problem must be solved.
- to supply a generalized force vector to the operator, the geometrical Jacobian matrix must be evaluated, and the actuators must be controlled by simple open control loop amplifiers.

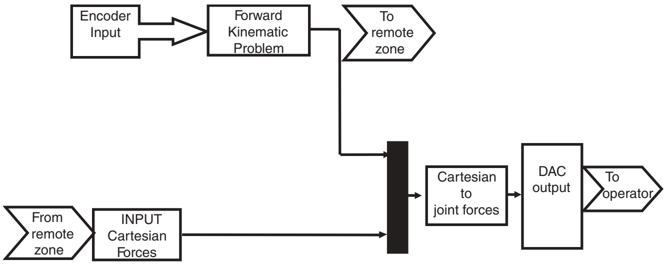


Fig. 5. Simplified Scheme of the impedance mode (open loop).

Fig. 5 shows a simplified scheme of the *impedance mode* work-mode (open loop control) programmed on the Magister-P. In this mode, the display is set on open loop mode, and all their control signals are accessible from the outside. This is the widespread control mode in the greater part of the haptic devices, and this work-mode allows the implementation of the classic bilateral scheme force-position (force-reflection FR), and others that are derived from this.

3.4. Admittance mode

Admittance displays are defined as devices that can supply information about the generalized force vector exerted by the operator; at the same time, they can supply to the operator the Cartesian position and/or orientation of the remote environment. To use a mechanical device as an admittance display, the next requirements must be fulfilled:

- to obtain the generalized force vector, the device must have force sensors. The widespread options are a six-d.o.f. force sensor on the end-effector link or the lecture of the motor intensity, and to use the simplified model of the DC motor, where the motor torque is proportional to this intensity. The Jacobian matrix converts those torques to generalized forces on the end-effector link.
- to render the remote environment positions to the operator, a position loop must be closed on the device. The dynamic model of the display is used to obtain the controllers to establish this loop. The mechanical structure must be stiff enough to supply those forces to the operator.

Fig. 6 shows a simplified scheme of the *admittance mode* work-mode. This mode allows the implementation of the position-position (Position-reflection PR) bilateral control scheme.

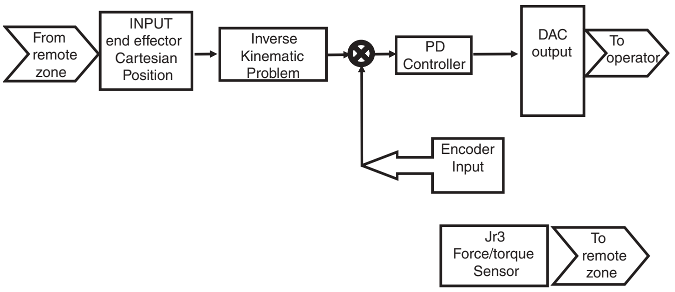


Fig. 6. Simplified Scheme of the admittance mode (position loop).

4. GLOBAL PERFORMANCE AND CHARACTERISTICS

To characterize the performance of a haptic display is not an easy task, due to its bi-directional nature; it is capable of both a reading and writing input to and from a human user. Several indices can be used to compare different devices with some similar characteristics, but frequently those indices point on a very particular feature. In specialized literature several of those indices can be found; some of them are used to characterize the performance of parallel devices, as the Ozaki global pay-load index (GPI),[25] the global conditioning index (GCI) defined by Gosselin and Angeles,[26] or the constant orientation workspace (COW) defined by Yoon and Ryu.[27] Colgate and Brown[28] suggested the dynamic range of achievable impedance (Z-width) as a measure of performance. Other indices have been considered for more general features of the haptic interfaces, like the resolution, precision, bandwidth or structural response. Nevertheless, the wide variety of the existent designs makes it difficult to establish some comparisons.

This section tries to show some characteristics of the presented device; its mechanical dimensions, workspace and force reflection bandwidth. A more complex analysis will be made in future works.

4.1. Mechanical dimensions

Table III shows the main mechanical features of the developed prototype.

4.2. Performance parameters

- Workspace. The COW is defined as the 3D region that can be obtained by the end-effector when the mobile platform is kept at a constant orientation. Therefore, large values of COW will give a wide range of motion. Fig. 7 shows a picture of the experimental workspace of the prototype. To maximize this workspace, and avoid the joint limit imposed by the spherical joint, the point of application of these joints is allocated at 45° with the plane of the end-effector.

Table III. Mechanical Characteristics.

Geometric data	
Upper base radius (r_{base})	250 mm
Joystick base radius (r_{joystick})	100 mm
Angle between spherical joints (δ_1)	15°
Angle between Universal joints (δ_0)	20°
Even links length	188 mm
Odd links length	220 mm
Cable transmission rate	13:1
Maximum angle of the pantograph	135°
Minimum angle of the pantograph	40°
Mass Data	
Joystick mass	0.29 Kg
Composite mass of the even links	0.06 Kg
Composite mass of the odd links (+ motor)	0.45 Kg
Main inertias of the end effector	$kgmm^2$ I_x 236.16 I_y 236.17 I_y 431.85
Main inertias of the even links	$kgmm^2$ I_x 42.04 I_y 139.21 I_z 146.10
Main inertias of the even links	$kgmm^2$ I_x 2246.35 I_y 2206.94 I_z 1139.79

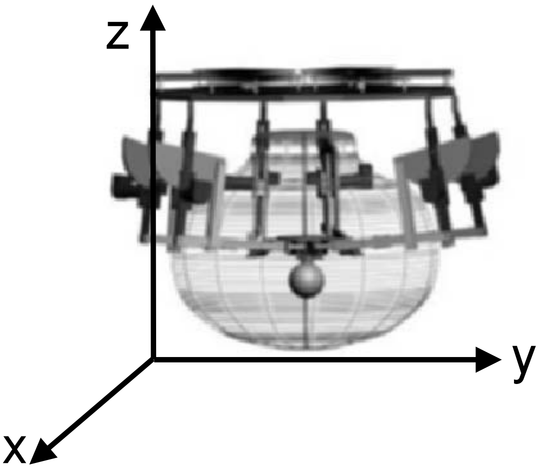


Fig. 7. Constant Orientation Workspace at $\beta = 0°$.

- Bandwidth. On the design process of a haptic device, the sensing and motor-control bandwidth plays an important role. The sensing bandwidth refers to the frequency with which tactile or kinesthetic stimuli are sensed, and the control bandwidth refers to the rapidity with which humans can respond. The sensing and control loops are asymmetric, meaning that the input (or sensing) bandwidth is much larger than the output (or control) bandwidth. The present prototype can obtain a sensing bandwidth of over 320 Hz, that is the bandwidth beyond which the human fingers cannot discriminate two consecutive force input signals. Thus this device can be used as a kinesthetic display supplying manipulation forces to the operator, or as a haptic display supplying tactile forces.
- Peak Force. The forces supplied to the operator must be according to the sensing human capacity. The maximum continuous force an operator can support without feeling fatigue is near 20 N.[3] For example, the peak output force of the well-known Phantom is 10 N,[29] and continuous force (without actuator overheating) is only 1.5 N. The present prototype is able to supply peak forces of 16.65 N on each pantograph, given a total force of 99.9 N in the vertical axis of the end-effector. This characteristic allows the use of the device in the position loop shown before.

5. INTEGRATION WITH A VIRTUAL TELEOPERATION ENVIRONMENT

To test the real performance of the presented device, a virtual teleoperated dynamic environments simulator have been developed. Among the main characteristics of this virtual simulator are the possibility of modelling a complete dynamic model of a robotic slave and its interaction with

https://doi.org/10.1017/S0263574704000608 Published online by Cambridge University Press

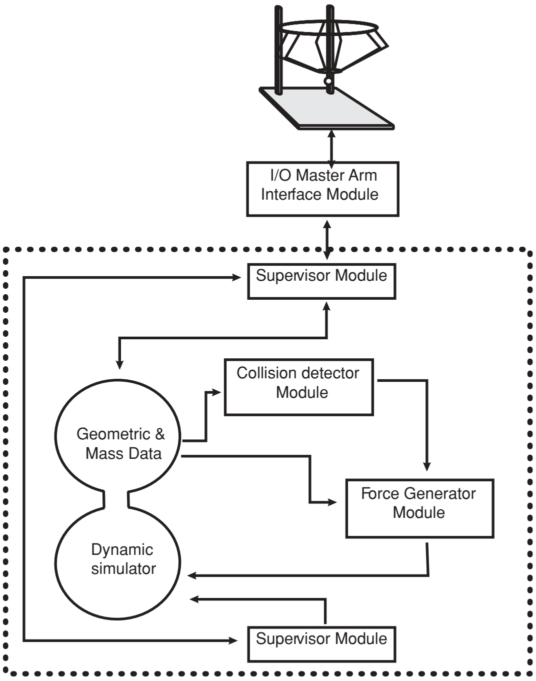


Fig. 8. Virtual Teleoperation Simulator System Architecture.

other dynamic objects that are in a virtual situation. Another important feature for the experimentation with teleoperated schemes is the capability of controlling the communication time between the local zone and the remote virtual zone, so that the virtual zone is controlled by a discrete time interval integration step. This way the communication with the master device can be included on the step simulation.

5.1. Architecture of the telerobotic simulator

This subsection concerns the basis of implementing a dynamic simulator for teleoperation tasks. Fig. 8 shows a general scheme of the developed architecture. The system is designed to be used as a haptic environment generator or as a help tool for real-teleoperation tasks. To provide independence from the devices connected to the system, each mechanical device is connected to the system by an *I/O Interface Module* that contains the input/oputput functions.

The characteristics and main modules of the developed simulator are briefly reviewed here. A more detailed explanation of this simulator being used with a Phantom device can be found in Sabater.[30]

- **Development and Graphic Environment**. The developed tool has been implemented in *C*++, using OpenGL graphic libraries. The user interface has been programmed with Qt libraries,[31] ensuring the portability of the tool. In fact, we have tested the software under Microsoft *Visual C*++ and Suse 8 Linux. OpenGL libraries give us the possibility of changing the point of view of the scene using the mouse movements.
- **Geometric and Mass Database module**. The first step to build a simulator is to establish a database containing the geometrical and inertial parameters of the bodies that appear in the virtual world. A rigid body has various properties that are needed on a simulation loop. Some of them change with time: The position vector of the body's reference system; the linear velocity; the orientation of a body, represented by a quaternion or by a rotation matrix; and the angular velocity vector.
  Some other properties are considered as invariant: the mass of the body; the center of mass expressed on the body reference system; and the inertia matrix expressed on the body reference system.
  Once the single body is full defined, we provide the interface to define kinematic chains using the Denavit-Hartenberg (DH) notation. Hierarchical relations between the linkable objects are also defined. Joint libraries containing rotational and prismatic primitives have also been defined. This way, the implementation of a new slave robot inside the VR environment only requires the definition of the DH and mass parameters.
- **I/O Master Arm module**. The platform has been considered for the experimentation and testing of open architectures, so any robotic device that has its own interface could be "connected" to the system. We have develop a generic class (*dhInterface*) from which we can derive any particular interface.
- **Supervisor module**. The supervisor modules perform administration and data management tasks.
- **Collision module**. The collision module has the role of obtaining the possible pairs of bodies that are colliding. The input information for this module is the geometric model of the objects and its actual positions. The collision detection algorithm is based on a AABB (Axis Aligned Bounding Boxes) representation of the 3D polyhedra of the scene, in which the tree hierarchy is constructed from boxes bounding the primitives associated with them. The boxes' axes are aligned to the axis of the object's local-coordinate system. This module stores the information on structures containing the contact position, the normal vector and the penetration depth of each pair of colliding geometries. The module that comes with ODE free software[32] is being used as collision module.
- **Contact-Forces Generator module**. Using the contact points provided by the collision module and the masses and dynamic parameters stored on the simulator, the force generator module calculates the reaction forces to the interaction of the environment elements. To compute the contact forces on each point involves solving a Linear Complementary Problem (LCP) that is obtained from the current contact configuration of the system.[33]
- **Dynamic Engine**. Once the contact forces (and associated torques) have been added to the net external force-torque pair acting on each object, the dynamic engine module is responsible for the actualization of the geometric and dynamic data of the objects in the environment. The key of this module is the solution of the movement equations of the system. A first order Euler integrator is used.

5.2. Force-position bilateral control scheme implemented

This subsection explains the one-step loop simulation that takes place in the FR scheme implementation of the experiments show on next subsection. Fig. 9 follows the loop simulation of the developed software. A 0.005 seconds

https://doi.org/10.1017/S0263574704000608 Published online by Cambridge University Press

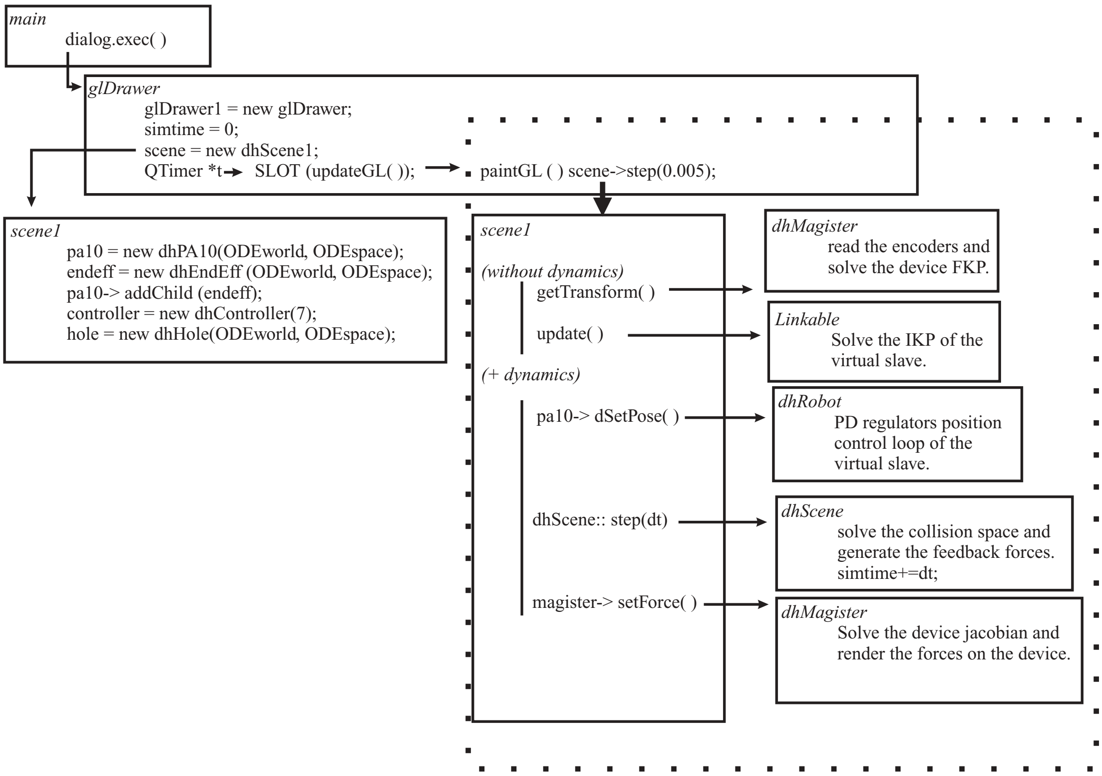

main
dialog.exec( )
glDrawer
glDrawer1 = new glDrawer;
simtime = 0;
scene = new dhScene1;
QTimer *t→ SLOT (updateGL( ));
paintGL ( ) scene->step(0.005);
scene1
pa10 = new dhPA10(ODEworld, ODEspace);
endeff = new dhEndEff (ODEworld, ODEspace);
pa10-> addChild (endeff);
controller = new dhController(7);
hole = new dhHole(ODEworld, ODEspace);
scene1
(without dynamics)
getTransform( )
update( )
(+ dynamics)
pa10-> dSetPose( )
dhScene:: step(dt)
magister-> setForce( )
dhMagister
read the encoders and solve the device FKP.
Linkable
Solve the IKP of the virtual slave.
dhRobot
PD regulators position control loop of the virtual slave.
dhScene
solve the collision space and generate the feedback forces.
simtime+=dt;
dhMagister
Solve the device jacobian and render the forces on the device.

Fig. 9. Schedule of developed software.

timer is implemented on the *glDrawer1* object. This timer commands the simulation loop (dashed box). The first step is to read the Cartesian position of the input master device, and to solve the inverse kinematics of the slave. We use the Clib library to interact with the mechanical device. Next, if the dynamics of the simulator is activated, the position control loop of the slave is solved (in this case using a decoupled control architecture with a simple PD controller on each joint). Once the positions, velocities and accelerations of all bodies are updated, the collision module is called, and one integration step is solved using the Euler integrator. The forces calculated are rendered using the *I/O module* of the master device.

The modularity of this schedule easily allows to insert blocks on the communication channels (for example delays), showing the capacity of the developed system to be used as a platform to test new control schemes.

The main characteristics of this architecture are:

- easy interface with the mechanical devices
- the *I/O Master Arm Interface* allows for the decoupling of the force servo loop from other loops (graphics, simulation, etc.)
- ready architecture for extension to other mechanical devices (master or slaves)
- ready architecture for extension to other control schemes
- user-friend interface

5.3. *Experimental results*

Using the haptic device presented in this paper and the dynamic simulator of the subsection 5.1, several experiments have been made. This subsection shows a peg-in-hole insertion. The bilateral scheme used is the force-position explained above. Some values of the critical simulation parameters are expressed on table IV. The accepted error on the iterative Newton-Raphson method have a direct relation with the cycle time of the dSpace board, so this value must be adjusted in order to be synchronized with the simulation step time.

Table IV. Values for the simulation.

Force reflection factor	0.1
Newton-Raphson Error	0.001
Max. Cycle time	0.01 sec.
Simulation step	0.05 sec.

Fig. 10 shows the results of the simulation. Two separate insertions of the peg were made, (between the 8–10 s and the 11–14 s). The plane marked with the horizontal line in $z = 0.21$ m represents the plane where the hole is allocated (Fig. 11). Under this height, the peg is in the hole. At the bottom part of the figure, the Z forces fed back to the operator are plotted. These forces are produced by the friction of the peg with the walls of the hole during the experiment. When the peg is going up, the forces are negative, and when the peg is out of the hole, these forces are null.

6. CONCLUSION

The design and development of an "open control" six degrees of freedom parallel master has been presented (Fig. 12).

https://doi.org/10.1017/S0263574704000608 Published online by Cambridge University Press

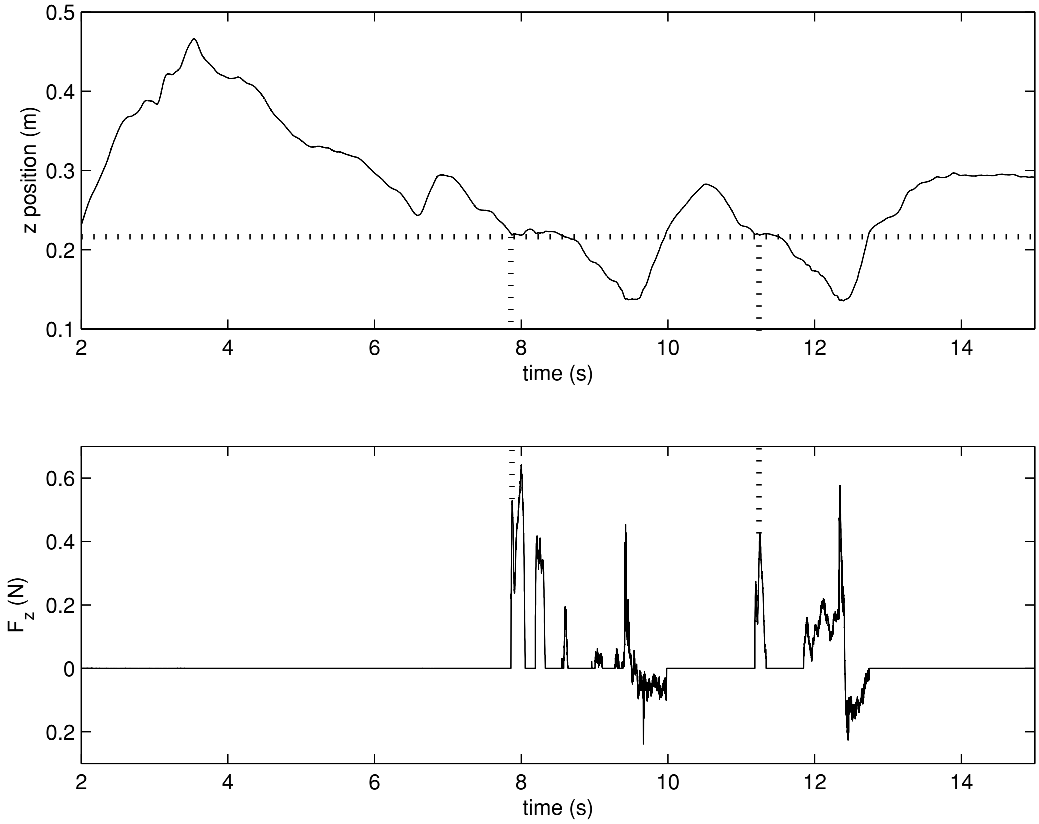


Fig. 10. Peg-in-hole task: position z (top part) and F_z force (bottom part).

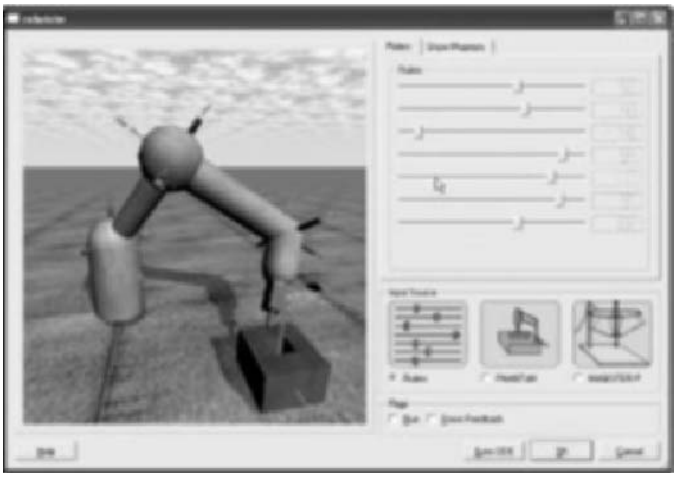


Fig. 11. Virtual Teleoperation Simulator interface.

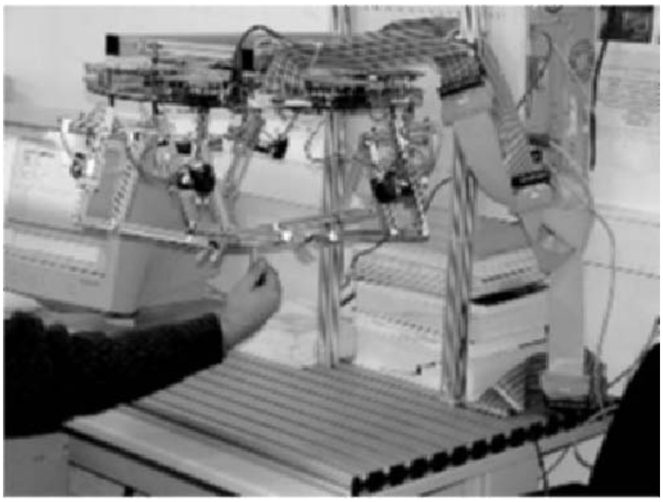


Fig. 12. Actual version of the Magister-P Prototype.

The device has 6-URS kinematic chains and is able to work either as an impedance or as an admittance device. A smart formulation based on the multibody dynamics theory has been developed. These algorithms have been used to obtain of the control schemes of the device.

The availability of "open" master and slave devices (a virtual slave) allows the implementation and experimentation with different bilateral control schemes, an interesting tool for future developments.

References

1. R. C. Goertz, "Fundamentals of general-purpose remote manipulators," *Nucleonics* **10**(11), 36–42 (1952).
2. B. Hannaford, "Stability and performace tradeoffs in bilateral telemanipulation," *Proc. IEEE Int. Conf. on Robotics and Automation* (1989) pp. 1764–1767.
3. G. Burdea, *Force and Touch Feedback for Virtual Reality* (New York, Wiley, 1996).
4. C. Sayers, *Remote Control Robotics* (Berlin, Springer-Verlag, 1998).

https://doi.org/10.1017/S0263574704000608 Published online by Cambridge University Press

5. M. Ueberle and M. Buss, "Design, control, and evaluation of a new 6 dof haptic device," *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (October 2–4 2002) pp. 1–11 (cd. version) Laussane.
6. Robotool project, "Advanced kinematics for manufacturing applications,"http://www.ifw.uni-hannover.de/robotool/pages/mainpage.htm (November, 2001). CEC project BE97-4177.
7. T. Brogardh, "Design of high performance parallel arm robots for industrial applications," *Proccedings of a Symposium Commemorating the Legacy, Works, and Life of Sir Robert Ball*, University of Cambridge (July 9–11 2000) pp. 1–11 (cd. version).
8. R. Aracil, R. Saltarén, J. M. Sabater, O. Reinoso and L. M. Jimenez, "Modelling, simulation and conception of parallel climbing robots for construction and service," *Proc. of the 2nd Int. Workshop and Conference on Climbing and Walking Robots (CLAWAR)* (1999) **Vol. 1**, pp. 253–265.
9. G. Brandt, A. Zimolong, L. Carrat, P. Merloz, H. W. Staudte and K. Lavallee, M. Radermacher and G. Rau, "Crigos: A compact robot for image guided orthopaedic interventions," *IEEE Trans. Inform. Technol. Biomed.* **3**, 252–260 (Dec. 1999).
10. L. Birglen, C. M. Gosselin, N. Pouliot, B. Monsarrat and T. Laliberté, "Shade, a new 3-dof haptic device," *IEEE Transactions on Robotics and Automation* **18**(2), 166–175 (2002).
11. P. Arcara, L. Di Stefano, S. Mattoccia, C. Melchiorri and G. Vassura, "Perception of depth information by means of a wire-actuated haptic interface," *Proc. IEEE Int. Conf. Robotics and Automation, San Francisco* (2000) pp. 3443–3448.
12. Y. Tsumaki, H. Naruse, D. N. Nenchev and M. Uchiyama, "Design of a compact 6-dof haptic interface," *Proc. IEEE Int. Conf. Robotics and Automation* (1998) pp. 2580–2585.
13. H. Iwata, "Artificial reality with force-feedback: Development of desktop virtual space with compact master manipulator," *Computer Graphics* **24**(4), 165–170 (1990).
14. J. M. Hollerbach, E. Cohen, W. Thompson, R. Freier, D. Johnson, Ali Nahvi, Don Nelson and T. V. Thompson, "Haptic interfacing for virtual prototyping of mechanical cad designs," *Proceedings of the 1997 ASME Design Engineering Technical Conferences* (1997) pp. 1–8.
15. Q. Lin and C. Kuo, "Virtual tele-operation of underwater robots," *Proc. of IEEE Int. Conf. Robotics and Automation, Alburquerque* (1997), **Vol. 1**, pp. 1022–1027.
16. A. K. Bejczy, W. Kim and S. Venema, "The phantom robot: Predictive display for teleoperation with time delay," *Proceedings of IEEE Int. Conf. Robotics and Automation Cincinaty* (1990) pp. 546–551.
17. G. C. Burdea, "The synergy between virtual reality and robotics," *IEEE Transactions on Robotics and Automation* **15** (3), 400–410 (1990).
18. J. P. Merlet, "Direct kinematics and assembly modes of parallel manipulators," *Int. J. Robotics Research* **11**(2), 150–162 (1992).
19. D. Lazard, "Stewart platform and gröbner basis," *Symposium on Advances in Robot Kinematics* (1992) ARK, **Vol. 13**(2), pp. 136–142.
20. G. Danescu and M. Dahan, "Note sur le probleme de prolègomènes n° 5. Prolègomènes n° 6: Problemes de geometrie pour les mecanismes," *Les Nouvelles INRIA*, 2–5 (1994).
21. P. Nair, "On the foward kinematics of parallel manipulators" *Int. J. Robotics Research* **13**(2), 171–188 (1994).
22. B. Dasgupta and T. S. Mruthyunjaya, "A canonical formulation of the direct position kinematics for a general 6-6 stewart platform," *Mechanism and Machine Theory* **29**(6), 819–827 (1994).
23. E. J. Haug, *Computer aided Kinematics and Dynamics of Mechanical Systems* (Allyn and Bacon, 1989).
24. dSpace GMBH, *DS1103PPC Controller Board. Installation and Configuration Guide*. (DSpace GMBH, 2002).
25. H. Ozaki, H. Wang, X. Liu and F. Gao, "The atlas of the payload capability for design of 2-dof planar parallel manipulators," *Proc. IEEE Inter. Conf. Systems Man and Cybernetics* (1996) pp. 1483–1488.
26. C. Gosselin and J. Angeles, "A global performance index for the kinematic optimization of robotic manipulator," *Trans. ASME J. Mech. Des.* **113**, 220–226 (1991).
27. Jungwon Yoon and Jeha Ryu, "Design, fabrication and evaluation of a new haptic device using a parallel mechanism," *IEEE/ASME Transactions on Mechatronics* **6**(3), 221–233 (2001).
28. J. E. Colgate and J. M. Brown, "Factors affecting the z-width of a haptic display," *Proc. IEEE Int. Conf. on Robotics and Automation, San Diego, USA* (May, 1994), **Vol. 1**, pp. 3205–3210.
29. T. H. Massie and K. Salisbury, "The phantom haptic interface: a device for probing virtual objects," *Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems* (Nov., 1994) pp. 295–301.
30. J. M. Sabater et al. "Dynamic virtual environment to test teleoperated systems with time delay communications," *Journal of Robotic Systems* (Accepted for publication, 2004).
31. Trolltech, Qt 3.1 whitepaper, http://www.trolltech.com/products/qt/whitepaper/qt-whitepaper.html, **1**, 1–50.
32. R. Smith, *Open Dynamics Engine v0.03 user guide.* **In**: http://q12.org/ode/, 2001.
33. G. Murilo, *Dynamic Simulations of Multibody Systems* (Berlin, Springer-Verlag, 2001).

https://doi.org/10.1017/S0263574704000608 Published online by Cambridge University Press