Available online at www.sciencedirect.com
SCIENCE DIRECT®
Mechatronics 15 (2005) 403–422
MECHATRONICS

Comparative study and experimental verification of singular-free algorithms for a 6 DOF parallel haptic device

Hyung Wook Kim [a], Jae Hoon Lee [b], Il Hong Suh [c], Byung-Ju Yi [b,*]

[a] *Department of Electronics Engineering, Hanyang University, 1271 Sa 1 Dong, Ansan Si, Gyeonggi Do, 426-791, Republic of Korea*
[b] *School of Electrical Engineering and Computer Science, Hanyang University, 1271 Sa 1 Dong, Ansan Si, Gyeonggi Do 426-791, Republic of Korea*
[c] *Graduate School of Information and Communications, Hanyang University, 17 Haengdang Dong, Seongdong Gu, Seoul 133-791, Republic of Korea*

Received 21 August 2003; accepted 19 October 2004

Abstract

It is well known that there exist many more singularities in parallel type mechanisms compared to serial type mechanisms. In haptic applications, these singularities deteriorate the force reflecting performance. Furthermore, as opposed to general manipulators, haptic systems cannot avoid the singular point, since they are operated by the user's random motion command.

Although many singularity-free algorithms for kinematically redundant manipulators have been proposed, singularity-free algorithms for parallel haptic application have not been extensively discussed. In this paper, various singularity-free algorithms that are appropriate for parallel haptic systems will be discussed. A new 6 degree-of-freedom (DOF) parallel mechanism equipped with four sub-chains is designed and employed as the test device. To cope with the singularity problem, four task-priority algorithms and a redundant actuation algorithm are

* Corresponding author. Fax: +82 31 416 6416.
E-mail address: bj@hanyang.ac.kr (B.-J. Yi).

0957-4158/$ - see front matter
doi:10.1016/j.mechatronics.2004.10.005

introduced and compared through simulation. In addition, experiments have been performed to show the effectiveness of those algorithms at the presence of singularity.

Keywords: Haptic device; Parallel structure; Singularity; Redundant actuation

1. Introduction

Since the master–slave system was initially proposed by Goertz in the 1950s [1], many researchers have developed various types of haptic displays, such as an exoskeleton type master arm by M. Bergamasco [5], PHANToM by Massie and Salisbury [3], MagLev Wrist by Ellis in CMU [2], and a magnetic levitation haptic interface by Berkelman et al. [4]. Despite the development of many haptic devices as mentioned above, the force reflection capability is still confined to three or less DOF, which is not sufficient to display the reality in virtual environments. Additionally, precise and hard contact feeling is also hard to realize with these devices.

Advantages of parallel mechanisms over serial mechanisms include high structural stiffness, low inertia, and high force bandwidth. However, haptic devices having parallel structure are heavy, have a relatively small workspace, and have multiple forward kinematic solutions. These devices also require a large amount of power due to floating actuators [2,6–8]. In order to overcome these difficulties, Gosselin [16], Lee et al. [14,15], Pierrot [17], and Tsai [18] proposed new mechanisms that employed non-floating actuators. These mechanisms were light and had a relatively large workspace as compared to parallel mechanisms.

Even though parallel mechanisms have many advantages for haptic devices requiring high structural stiffness, high force bandwidth and high force dynamic range, it has been reported that parallel mechanisms contain more singular configurations than typical serial mechanisms does. This is due to kinematic interaction among chains and distribution of actuators to each chain. Thus, identification of the singularity for every parallel mechanism should be made prior to its operation. However, there are still many singular configurations that are not still being identified. Also, differently from usual industrial robot manipulators controlled by off-line programming, the singular points cannot be avoided in haptic operation due to the random motion generated by the human operator, even though the singular points of parallel haptic devices are totally identified. At singular points, the haptic system cannot generate the reflection force completely, and moreover, torque saturation occurs. Thus, there is a need to investigate a singularity-free algorithm that maintains system performance and precisely generates the reflection force.

Research has been conducted with a variety of focuses in regards to singularity-free algorithms for kinematically redundant mechanisms [9–13], although research specific to singularity-free algorithms for parallel type haptic devices has been limited. In this paper, four kinds of singularity-free algorithms will be examined that cope with the singularity problem of parallel haptic systems. Also, in an attempt

to overcome these limitations, a new design including redundant actuation is proposed.

This paper is organized as follows: Section 2 shows the overview of task-priority algorithms used to overcome the singularity problem, the kinematics of a redundantly actuated 6 DOF parallel haptic device is discussed in Section 3. In Sections 4 and 5, simulation and experimental results are given and Section 6 reports overall conclusions.

2. Singularity-free algorithms

In general, parallel mechanisms contain quite a few singular points within their workspace. Singularity causes serious problems such as torque saturation, undesirable motion, and the breakdown of systems. Of more significant concern is the fact that the singular point cannot be avoided in haptic systems due to the operation of the haptic device by the user's random motion command. Several motion planning algorithms have been reported in order to avoid singularities by using kinematically redundant manipulators. However, singularity-free algorithms for parallel devices have not yet been deeply discussed. Thus, there is a need to investigate singularity-free algorithms adequate to parallel devices.

Fig. 1 shows three examples of singular configurations of a parallel haptic mechanism [14]. Fig. 1(a) shows a singular position where the top plate is parallel to one of the upper chains. Fig. 1(b) shows a singular position where the system touches the boundary of the workspace, and Fig. 1(c) shows an algorithmic singularity caused by an internal kinematic problem.

Another important concept of algorithms is that of a task-priority algorithm. This was introduced by Maciejewski et al. [10] and Nakamura et al. [11] and was applied to the control of kinematically redundant manipulators. According to the task priority, the task having a higher priority is performed first and the task having a lower priority will be performed successively. In this paper, the previously developed kinematic task-priority algorithms will be modified to force-based task-priority algorithms to cope with the singularity problem of parallel haptic system.

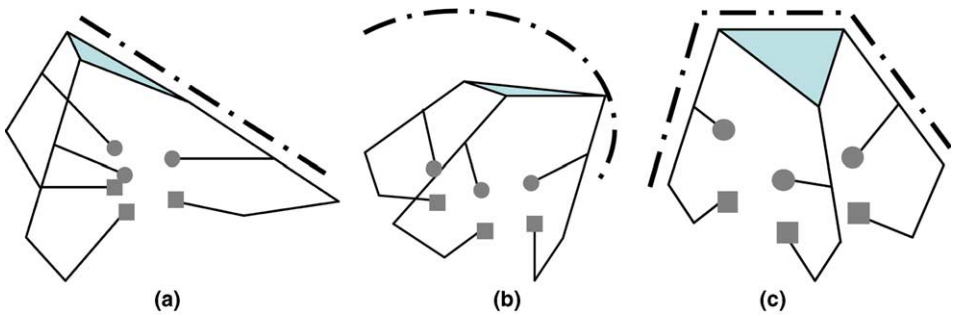

Fig. 1. Singular configurations.

2.1. Nakamura's algorithm

To begin with, Nakamura's task priority algorithm [11] is considered. The subtask having the first priority will be specified using the first manipulation force, $\boldsymbol{T}_{u1} \in \mathbf{R}^{m_1}$, and the secondary priority subtask will be specified using the second manipulation force, $\boldsymbol{T}_{u2} \in \mathbf{R}^{m_2}$.

The output force $\boldsymbol{T}_{u1} \in \mathbf{R}^{m_1}$ is chosen as the first manipulating force and the output moment $\boldsymbol{T}_{u2} \in \mathbf{R}^{m_2}$ is chosen as the second manipulating force. The force relationship between the joint torque $\boldsymbol{T}_A \in \mathbf{R}^n$ and the first manipulating force vector can expressed as follows:

$$\boldsymbol{T}_{u1} = \boldsymbol{J}_1 \boldsymbol{T}_A, \tag{1}$$

where $\boldsymbol{J}_1$ denotes the Jacobian matrix relating the first manipulating force to the joint torque vector.

The general solution of Eq. (1) is expressed as

$$\boldsymbol{T}_A = \boldsymbol{J}_1^+ \boldsymbol{T}_{u1} + \{\boldsymbol{I} - \boldsymbol{J}_1^+ \boldsymbol{J}_1\}\boldsymbol{z}, \tag{2}$$

where the matrix $\boldsymbol{J}_1^+$ is the pseudoinverse of $\boldsymbol{J}_1$ and $\boldsymbol{z}$ is an arbitrary vector that satisfies some secondary requirement. The m_2-dimensional secondary task is specified in the following form:

$$\boldsymbol{T}_{u2} = \boldsymbol{J}_2 \boldsymbol{T}_A, \tag{3}$$

where $\boldsymbol{J}_2 \in \mathbf{R}^{m_2 \times n}$ is the Jacobian matrix for the secondary task. Minimizing the secondary task error $\|\boldsymbol{T}_{u2} - \boldsymbol{J}_2 \boldsymbol{T}_A\|_2$ in the least square sense, yields the least square solution $\boldsymbol{z}$ as [11]:

$$\boldsymbol{z} = \widetilde{\boldsymbol{J}}_2^+ \{\boldsymbol{T}_{u2} - \boldsymbol{J}_2 \boldsymbol{J}_1^+ \boldsymbol{T}_{u1}\} + \{\boldsymbol{I} - \widetilde{\boldsymbol{J}}_2^+ \widetilde{\boldsymbol{J}}_2\}\boldsymbol{x}, \tag{4}$$

where $\widetilde{\boldsymbol{J}}_2 = \boldsymbol{J}_2\{\boldsymbol{I} - \boldsymbol{J}_1^+ \boldsymbol{J}_1\}$ and $\boldsymbol{x}$ is an arbitrary vector.

Thus, the general solution of $\boldsymbol{T}_A$ is expressed as

$$\boldsymbol{T}_A = \boldsymbol{J}_1^+ \boldsymbol{T}_{u1} + \widetilde{\boldsymbol{J}}_2^+ \{\boldsymbol{T}_{u2} - \boldsymbol{J}_2 \boldsymbol{J}_1^+ \boldsymbol{T}_{u1}\} + \{\boldsymbol{I} - \widetilde{\boldsymbol{J}}_2^+ \widetilde{\boldsymbol{J}}_2\}\boldsymbol{x}. \tag{5}$$

since $\{\boldsymbol{I} - \boldsymbol{J}_1^+ \boldsymbol{J}_1\}$ is idempotent.

2.2. Chiaverini's algorithm

In general, there are two kinds of singularities when solving inverse kinematics. One is the kinematic singularity and the other is algorithmic singularity [12]. The kinematic singularity occurs in the following case:

$$\operatorname{rank}(\boldsymbol{J}_1) < m_1 \quad \text{or} \quad \operatorname{rank}(\boldsymbol{J}_2) < m_2. \tag{6}$$

Whereas the algorithmic singularity occurs in either

$$\mathbf{R}(\boldsymbol{J}_1^+) \cap \mathbf{R}(\boldsymbol{J}_2^+) \neq \emptyset \quad \text{or} \quad \mathbf{N}(\boldsymbol{J}_1) \cap \mathbf{N}(\boldsymbol{J}_2) \neq \emptyset \tag{7}$$

when $m_2 \leqslant n - m_1$ [12]. In Eq. (7), $\mathbf{R}(*)$ and $\mathbf{N}(*)$ represent the range space and null space of matrix $*$, respectively.

The kinematic singularity exists as the fundamental problem when solving inverse kinematics; alternatively the algorithmic singularity can be eliminated or changed according to the characteristics of the employed algorithm since it is considered an artificial part. Regardless of the kind of singularities, it should be noted that the general solution of Eq. (5) is not acceptable near singularities [12].

To eliminate the algorithmic singularities existing in Nakamura's method, Chiaverini [12] modified Eq. (5) as follows:

$$\boldsymbol{T}_A = \boldsymbol{J}_1^+ \boldsymbol{T}_{u1} + \{\boldsymbol{I} - \boldsymbol{J}_1^+ \boldsymbol{J}_1\}\{\boldsymbol{J}_2^+ \boldsymbol{T}_{u2} + \{\boldsymbol{I} - \boldsymbol{J}_2^+ \boldsymbol{J}_2\}\boldsymbol{z}\}. \tag{8}$$

Although Nakamura's algorithm which is given in Eq. (5) has algorithmic singularities, the primary and the secondary tasks do not have any task error in the normal case. On the contrary, Chiaverini's algorithm which is given in Eq. (8) has no algorithmic singularity, but it always has the secondary task error with the exception of the case when $\boldsymbol{J}_2\boldsymbol{J}_1^+ = 0$. Thus, there exists trade-offs between the two algorithms.

2.3. Choi's algorithm

By using the advantage that Chiaverini's algorithm has no algorithmic singularity, Choi et al. [13] proposed an algorithm that reduces the secondary task error found in Eq. (8). Chiaverini's algorithm was modified to the following equation:

$$\boldsymbol{T}_A = \boldsymbol{J}_W^+ \boldsymbol{T}_{u1} + \{\boldsymbol{I} - \boldsymbol{J}_W^+ \boldsymbol{J}_1\}\{\boldsymbol{J}_2^+ \boldsymbol{T}_{u2} + \{\boldsymbol{I} - \boldsymbol{J}_2^+ \boldsymbol{J}_2\}\boldsymbol{z}\}, \tag{9}$$

where $\boldsymbol{J}_W^+ = \boldsymbol{W}^{-1}\boldsymbol{J}_1^{\mathrm{T}}(\boldsymbol{J}_1\boldsymbol{W}^{-1}\boldsymbol{J}_1^{\mathrm{T}})^{-1}$.

If the weight matrix is chosen to be a positive definite matrix as follows:

$$\boldsymbol{W} = \boldsymbol{J}_1^{\mathrm{T}}\boldsymbol{J}_1 + \boldsymbol{J}_2^{\mathrm{T}}\boldsymbol{J}_2 + \epsilon\boldsymbol{I} > 0, \tag{10}$$

it will have no algorithmic singularities, although Eq. (9) tend to contaminate the performance of the secondary task with a small positive number ϵ.

2.4. Damped least square method

The singular value decomposition theorem states that for any matrix $\boldsymbol{A} \in \mathbf{R}^{m\times n}$, there exists orthogonal matrices $\boldsymbol{U} \in \mathbf{R}^{m\times m}$ and $\boldsymbol{V} \in \mathbf{R}^{n\times n}$ such that

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}, \tag{11}$$

where the matrix $\boldsymbol{\Sigma} \in \mathbf{R}^{m\times n}$ has the block matrix form

$$\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{S} & 0 \\ 0 & 0 \end{pmatrix}, \tag{12}$$

and $\boldsymbol{S} = \mathrm{diag}(\sigma_1, \ldots, \sigma_r) \in \mathbf{R}^{r\times r}$ and $r = \mathrm{rank}(\boldsymbol{A}\mathrm{f})$.

The pseudoinverse can then be given by

$$\boldsymbol{A}^+ = \boldsymbol{V}\boldsymbol{\Sigma}^+\boldsymbol{U}^{\mathrm{T}}, \tag{13}$$

where

$$\boldsymbol{\Sigma}^{+} = \begin{pmatrix} \boldsymbol{S}^{-1} & \boldsymbol{0} \\ 0 & 0 \end{pmatrix}.$$

Wampler [9] initially proposed the damped least-square method, given by

$$\boldsymbol{A}_{\alpha}^{+} = \boldsymbol{V}\boldsymbol{\Sigma}_{\alpha}\boldsymbol{U}^{\mathrm{T}}, \tag{14}$$

where

$$\boldsymbol{\Sigma}_{\alpha} = \begin{pmatrix} \boldsymbol{S}_{\alpha} \\ 0 \end{pmatrix} \in \mathbf{R}^{n\times m} \tag{15}$$

and $\boldsymbol{S}_{\alpha} = \operatorname{diag}\left(\frac{\sigma_1}{\sigma_1^2+\alpha^2}, \ldots, \frac{\sigma_m}{\sigma_m^2+\alpha^2}\right) \in \mathbf{R}^{m\times m}$.

If α is found to be much less than the smallest nonzero singular value of $\boldsymbol{J}$, then $\boldsymbol{T}_{A\alpha}$ is approximately the minimum norm solution. As a singular value approaches zero, the associated component of $\boldsymbol{S}_{\alpha}$ reaches a maximum when $\sigma = \alpha$ and then decreases rapidly to zero. Additionally, the size of the solution, $\|\boldsymbol{T}_{\alpha}\|$, decreases monotonically as α increases. This is a useful fact that can be exploited to find solutions subject to joint torque limits.

3. Four legged parallel haptic device involving redundant actuators

It is well known that the advantage of employing redundant actuation is two-fold. First, large payloads can be achieved, and secondly singularities can be minimized through the use of redundant actuation. In this paper, a four legged parallel mechanism with redundant actuators is proposed as a means to resolve the singularity problem occurring in haptic operation.

3.1. Geometric description

The developed four legged 6 DOF parallel haptic device is shown in Fig. 2, the kinematic structure is depicted in Fig. 3. This haptic device consists of a top plate, eight actuators located under the base, and four parallel chains connecting the top plate to the eight actuators. To avoid interference between the motor bodies and the links, the upper actuators (M1) are placed under the base plate and the actuating torques of these upper actuators are transmitted to the first joint through a bevel gear set.

To further describe this device, let {B} represent the base frame fixed to the ground with its origin at the center of the base, and {T} represent the local frame fixed to the top plate with its origin at the center of the top frame. Each of the four ball-socket joints (${}^{m}\boldsymbol{C}$, $m = 1$–4) of the top plate is placed on the circle of radius (R_T) with a separation of 90° from each other. Similarly, four pairs of actuators are placed on the ground, also with a separation of 90°. Each of the four actuator pairs consist

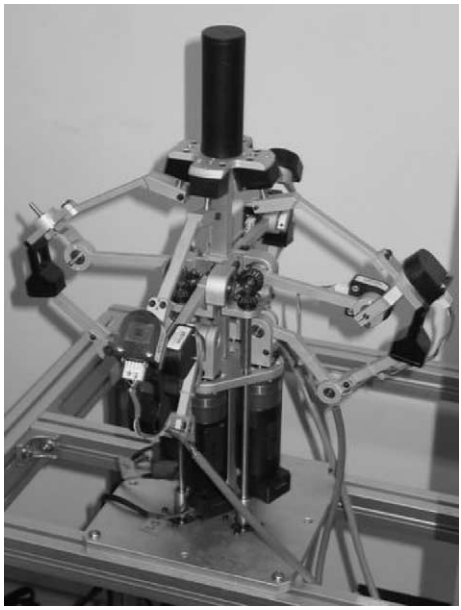

Fig. 2. Four-legged parallel haptic device.

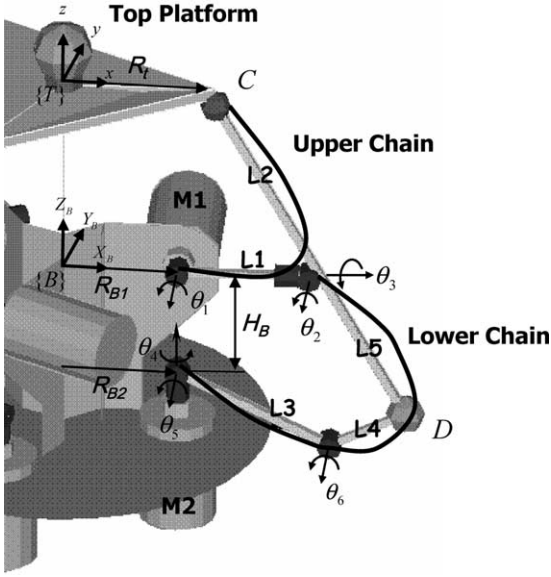

Fig. 3. Kinematic structure of a four-legged haptic device.

of the upper actuator (M1), placed on the circle of radius (R_{B1}), and the lower actuator (M2), placed on the circle of radius (R_{B2}) vertical to the ground. The distance from the upper actuator to the lower actuator, in the z-direction, is denoted by H_B.

The first leg is aligned with x-axis of base frame $\{B\}$, and other three legs are located successively with a separation of 90° from the first leg in counterclockwise direction. Each leg consists of an upper closed-chain and a lower closed-chain as shown in Fig. 3. The upper chains connect the upper actuators (M1) to the top plate through the ball–socket joints (${}^m\boldsymbol{C}$, $m = 1$–4). The lower chains connect the lower actuators (M2) to the upper actuators (M1) through the lower ball–socket joints (${}^m\boldsymbol{D}$, $m = 1$–4). Thus, the upper actuators support gravity loads and generate the Z directional motion while the lower actuators generate the X and Y directional motions, respectively. Interaction of the four chains generates the rotational motion of the upper plate.

The position vector of the end-effector is defined as

$$\boldsymbol{u} = (x_t, y_t, z_t, \theta_x, \theta_y, \theta_z)^{\mathrm{T}}, \tag{16}$$

where (x_t, y_t, z_t) represents the position of the top plate's origin, and $(\theta_x, \theta_y, \theta_z)$ denotes $\hat{x}_t$–$\hat{y}_t$–$\hat{z}_t$ Euler angles equivalent to $[{}^{\mathrm{T}}_{B}\mathbf{R}]$. This can be expressed by

$$[{}^{\mathrm{T}}_{B}\mathbf{R}] = [\mathrm{Rot}(\hat{x}_t, \theta_x)][\mathrm{Rot}(\hat{y}_t, \theta_y)][\mathrm{Rot}(\hat{z}_t, \theta_z)]. \tag{17}$$

3.2. Forward/inverse kinematics

In general, parallel mechanisms have a multitude of forward kinematic solutions; our 6 DOF parallel haptic device has an eighth-order polynomial. Two additional encoders were placed at the passive joints of each upper chain (θ_2, θ_3) to obtain a unique forward solution.

The inverse kinematic solution of each upper chain can be calculated from the position vectors of the upper ball-socket joints (${}^m\boldsymbol{C}$, $m = 1$–4), which are given from the position and orientation of the top plate. In the same manner, the inverse kinematic solution of each lower chain can be obtained from position vectors of the lower ball-socket joints (${}^m\boldsymbol{D}$, $m = 1$–4), which are obtained from the forward kinematics of the upper chain.

3.3. First-order kinematic modeling

In the following subsection, the first-order KIC (kinematic influence coefficient) that relates the operational velocity ($\dot{\boldsymbol{u}}$) to the active joint velocity ($\dot{\boldsymbol{\theta}}_A$) is described. The position vector of the four upper ball-socket joints is denoted as

$$\boldsymbol{C} = ({}^1\boldsymbol{C}^{\mathrm{T}}, {}^2\boldsymbol{C}^{\mathrm{T}}, {}^3\boldsymbol{C}^{\mathrm{T}}, {}^4\boldsymbol{C}^{\mathrm{T}})^{\mathrm{T}} \tag{18}$$

with respect to the global reference frame, where ${}^m\boldsymbol{C} = ({}^m x_c, {}^m y_c, {}^m z_c)^{\mathrm{T}}$.

The position vector of the mth upper contact point is expressed as

$${}^m\boldsymbol{C} = \boldsymbol{u}_t + {}^m\boldsymbol{r}_c, \tag{19}$$

where ${}^m\boldsymbol{r}_c = [{}^{\mathrm{T}}_{B}\mathbf{R}]{}^m\boldsymbol{r}_c^{(\text{top plate})}$.

Differentiating ${}^m\boldsymbol{C}$ with respect to time results in

$$ {}^m\dot{\boldsymbol{C}} = \dot{\boldsymbol{u}}_t + \boldsymbol{\omega} \times {}^m\boldsymbol{r}_c, \tag{20} $$

where $\dot{\boldsymbol{u}}_t = (\dot{x}_t, \dot{y}_t, \dot{z}_t)^{\mathrm{T}}$, $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)^{\mathrm{T}}$, and ${}^m\boldsymbol{r}_c = ({}^mr_{cx}, {}^mr_{cy}, {}^mr_{cz})^{\mathrm{T}}$. Eq. (20) can be written in a matrix form as

$$ {}^m\dot{\boldsymbol{C}} = [{}^m\boldsymbol{G}_u^c]\,\dot{\boldsymbol{u}}, \tag{21} $$

where

$$ [{}^m\boldsymbol{G}_u^c] = \begin{bmatrix} 1 & 0 & 0 & 0 & {}^mr_{cz} & -{}^mr_{cy} \\ 0 & 1 & 0 & -{}^mr_{cz} & 0 & {}^mr_{cx} \\ 0 & 0 & 1 & {}^mr_{cy} & -{}^mr_{cx} & 0 \end{bmatrix}, $$

and $\dot{\boldsymbol{u}} = (\dot{x}_t, \dot{y}_t, \dot{z}_t, \omega_x, \omega_y, \omega_z)^{\mathrm{T}}$.

At this point, the relationship between $\dot{\boldsymbol{u}}$ and $\dot{\boldsymbol{C}}$ can be described as

$$ \dot{\boldsymbol{C}} = [\boldsymbol{G}_u^c]\,\dot{\boldsymbol{u}}, \tag{22} $$

where $[\boldsymbol{G}_u^c] = \left[\, [{}^1\boldsymbol{G}_u^c]^{\mathrm{T}} \quad [{}^2\boldsymbol{G}_u^c]^{\mathrm{T}} \quad [{}^3\boldsymbol{G}_u^c]^{\mathrm{T}} \quad [{}^4\boldsymbol{G}_u^c]^{\mathrm{T}} \,\right]^{\mathrm{T}}$.

The open chain kinematics of the mth upper chain is described as

$$ {}^m\dot{\boldsymbol{C}} = [{}^m\boldsymbol{G}_{u\theta}^c]\;{}^m\dot{\boldsymbol{\theta}}_u, \tag{23} $$

where ${}^m\dot{\boldsymbol{\theta}}_u$ is the joint velocity vector of the mth upper chain and $[{}^m\boldsymbol{G}_{u\theta}^c]$ is a Jacobian matrix which relates the joint velocity of the mth upper chain to the velocity of the upper contact point.

The only situation that $[{}^m\boldsymbol{G}_{u\theta}^c]$ will be singular is that the first and second links of the upper chain are parallel to each other. Since the robot is designed in such a way as to keep the unfolded configuration of the upper chain by the limited motion range of the lower chain, the first and second links of the upper chain will never be parallel. So, the fact that the Jacobian matrix $[{}^m\boldsymbol{G}_{u\theta}^c]$ is invertible is always guaranteed by the mechanism design.

The first-order inverse kinematics of Eq. (23) is obtained as

$$ {}^m\dot{\boldsymbol{\theta}}_u = [{}^m\boldsymbol{G}_c^{u\theta}]\;{}^m\dot{\boldsymbol{C}}, \tag{24} $$

where $[{}^m\boldsymbol{G}_c^{u\theta}] = [{}^m\boldsymbol{G}_{u\theta}^c]^{-1}$.

Congregating m from 1 to 4 yields

$$ \dot{\boldsymbol{\theta}}_u = [\boldsymbol{G}_c^{u\theta}]\;\dot{\boldsymbol{C}}, \tag{25} $$

where $\dot{\boldsymbol{\theta}}_u = \left([{}^1\dot{\boldsymbol{\theta}}_u]^{\mathrm{T}}\; [{}^2\dot{\boldsymbol{\theta}}_u]^{\mathrm{T}}\; [{}^3\dot{\boldsymbol{\theta}}_u]^{\mathrm{T}}\; [{}^4\dot{\boldsymbol{\theta}}_u]^{\mathrm{T}} \right)^{\mathrm{T}}$ and

$$ [\boldsymbol{G}_c^{u\theta}] = \begin{bmatrix} [{}^1\boldsymbol{G}_c^{u\theta}] & 0 & 0 & 0 \\ 0 & [{}^2\boldsymbol{G}_c^{u\theta}] & 0 & 0 \\ 0 & 0 & [{}^3\boldsymbol{G}_c^{u\theta}] & 0 \\ 0 & 0 & 0 & [{}^4\boldsymbol{G}_c^{u\theta}] \end{bmatrix}. $$

3.4. Internal kinematics

As shown in Fig. 4, the velocity vector of the mth lower ball-socket joints is described as

$$ {}^{m}\dot{\boldsymbol{D}} = [{}^{m}\boldsymbol{G}^{D}_{u\theta}]\,{}^{m}\dot{\boldsymbol{\theta}}_{u} = [{}^{m}\boldsymbol{G}^{D}_{l\theta}]\,{}^{m}\dot{\boldsymbol{\theta}}_{l}, \tag{26} $$

where ${}^{m}\dot{\boldsymbol{\theta}}_{l}$ is the joint velocity vector of the mth lower chain, and $[{}^{m}\boldsymbol{G}^{D}_{u\theta}]$ and $[{}^{m}\boldsymbol{G}^{D}_{l\theta}]$ represent the first-order KICs for the upper and the lower branch, respectively.

From Eq. (26), the first-order kinematic relationship between the upper and lower chains can be obtained as follows:

$$ {}^{m}\dot{\boldsymbol{\theta}}_{l} = [{}^{m}\boldsymbol{G}^{D}_{l\theta}]^{-1}[{}^{m}\boldsymbol{G}^{D}_{u\theta}]\,{}^{m}\dot{\boldsymbol{\theta}}_{u} = [{}^{m}\boldsymbol{G}^{l\theta}_{u\theta}]\,{}^{m}\dot{\boldsymbol{\theta}}_{u}, \quad m = 1\text{–}4. \tag{27} $$

Note that the two active joints of the mth chain are ${}^{m}\dot{\theta}_{1}$ located at the upper chain and ${}^{m}\dot{\theta}_{4}$ located at the lower chain. By using the row-column selection of Eq. (27), the first-order relationship between the active joint velocity (${}^{m}\dot{\boldsymbol{\theta}}_{A} : {}^{m}\dot{\theta}_{1}, {}^{m}\dot{\theta}_{4},\ m = 1\text{–}4$) and the joint velocity ($\dot{\boldsymbol{\theta}}_{u}$) of the upper chains can be expressed as

$$ \dot{\boldsymbol{\theta}}_{A} = [\boldsymbol{G}^{A}_{u\theta}]\,\dot{\boldsymbol{\theta}}_{u}, \tag{28} $$

where

$$ [\boldsymbol{G}^{A}_{u\theta}] = \begin{bmatrix} 1 & 0 & 0 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ [{}^{1}\boldsymbol{G}^{l\theta}_{u\theta}]_{1;} & & & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 1 & 0 & 0 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & [{}^{2}\boldsymbol{G}^{l\theta}_{u\theta}]_{1;} & & & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 1 & 0 & 0 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & [{}^{3}\boldsymbol{G}^{l\theta}_{u\theta}]_{1;} & & & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & 1 & 0 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & [{}^{4}\boldsymbol{G}^{l\theta}_{u\theta}]_{1;} & & \end{bmatrix} \in \mathbf{R}^{8\times 12}, $$

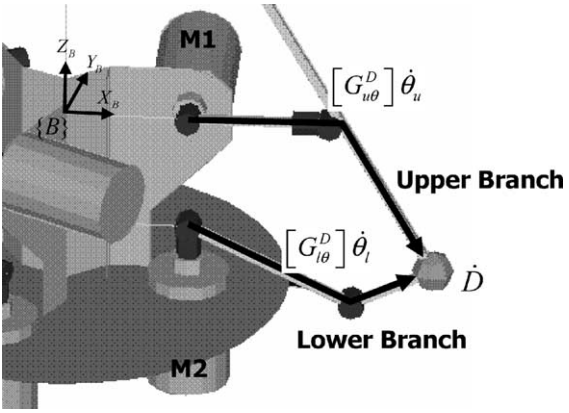

Fig. 4. Description of the lower ball-socket joint.

$$\dot{\boldsymbol{\theta}}_A = [{}^1\theta_1\ {}^1\theta_4\ {}^2\theta_1\ {}^2\theta_4\ {}^3\theta_1\ {}^3\theta_4\ {}^4\theta_1\ {}^4\theta_4\ ]^{\mathrm{T}} \in \mathbf{R}^8$$

and

$$\dot{\boldsymbol{\theta}}_u = [{}^1\theta_1\ {}^1\theta_2\ {}^1\theta_3\ {}^2\theta_1\ {}^2\theta_2\ {}^2\theta_3\ {}^3\theta_1\ {}^3\theta_2\ {}^3\theta_3\ {}^4\theta_1\ {}^4\theta_2\ {}^4\theta_3]^{\mathrm{T}} \in \mathbf{R}^{12}.$$

Substituting Eq. (24) into Eq. (28) yields the following relationship between the velocity of upper contact points and the velocity of active joints

$$\dot{\boldsymbol{\theta}}_A = [\boldsymbol{G}_c^A]\ \dot{\boldsymbol{C}}, \tag{29}$$

where $[\boldsymbol{G}_c^A] = [\boldsymbol{G}_{u\theta}^A][\boldsymbol{G}_c^{u\theta}]$.

Furthermore, by substituting Eq. (22) into Eq. (29), the relationship between the velocity of end-effector and the velocity of active joint is obtained as

$$\dot{\boldsymbol{\theta}}_A = [\boldsymbol{G}_u^A]\ \dot{\boldsymbol{u}}, \tag{30}$$

where $[\boldsymbol{G}_u^A] = [\boldsymbol{G}_c^A][\boldsymbol{G}_u^c]$, $\dot{\boldsymbol{\theta}}_A \in \mathbf{R}^8$, $\dot{\boldsymbol{u}} \in \mathbf{R}^6$, and $[\boldsymbol{G}_u^A] \in \mathbf{R}^{8\times 6}$.

By the duality relation, the relationship between the operational forces and the joint torques is described as

$$\boldsymbol{T}_u = [\boldsymbol{G}_u^A]^{\mathrm{T}}\ \boldsymbol{T}_A, \tag{31}$$

where $\boldsymbol{T}_u \in \mathbf{R}^6$ and $\boldsymbol{T}_A \in \mathbf{R}^8$ are the output force vector at the end-effector and the input torque vector of active joints, respectively.

When we employ task-priority algorithms, the first three rows of $[\boldsymbol{G}_u^A]^{\mathrm{T}}$ correspond to $\boldsymbol{J}_1$ of Eq. (1) and the last three rows correspond to $\boldsymbol{J}_2$ of Eq. (3). When we employ the redundant actuation algorithm, the general solution will be employed, which is given by

$$\boldsymbol{T}_A = ([\boldsymbol{G}_u^A]^{\mathrm{T}})^{+}\boldsymbol{T}_u + (\boldsymbol{I} - ([\boldsymbol{G}_u^A]^{\mathrm{T}})^{+}[\boldsymbol{G}_u^A]^{\mathrm{T}})\boldsymbol{\epsilon} \tag{32}$$

where $\boldsymbol{\epsilon}$ is an arbitrary vector.

In the general solution of Eq. (32), the first term is a pseudoinverse solution and the second term denotes a null space solution. In the simulation, only the pseudoinverse solution was employed.

4. Simulation results

In a singular configuration, a parallel haptic device cannot generate the desired 6 DOF force/moment completely and in addition, torque saturation may occur. In such cases, the user may choose the priority of tasks according to particular purpose and condition. In this paper, 3 DOF forces and 3 DOF moments are chosen to be the first and second priority tasks, respectively.

At the outset, the performances of three singularity-free algorithms and the redundant actuation algorithm are tested by simulation. Initially, actuator torques required to perform the given task are computed. Subsequently, the output forces and moments are recalculated using Eqs. (1) and (3) to check the task error. In all simulations, the primary task is selected to display -10 N along the Z-axis and the secondary task is selected to yield zero moments.

The simulation results of the four algorithms being applied to the four-legged parallel haptic device, and actuated only by six motors, is shown in Table 1. The system is in the neighborhood of a singular configuration as shown in Fig. 1(a) (0 cm, 0 cm, 12 cm, 0°, 40°, 0°). In such a configuration, the second link of the first chain and the top plate are parallel. It can be observed that the actuator torques of Inverse Jacobian and Nakamura's algorithms are saturated. On the other hand, Chiaverini's, Choi's, and SVD damping algorithms do not generate actuator saturation, but errors still remain in the secondary task.

Table 2 shows the simulation result of the four legged parallel haptic device, which was driven by eight motors. Actuator saturation was not found at the same singular configuration. Furthermore, the torque norm is smaller than the other three algorithms (Chiaverini:Choi:SVD damping:Redundant actuation = 0.806:0.801:1.815:0.783). Thus, redundant actuation has advantages in the aspects of singularity avoidance and input saving.

Table 1
Comparison of singularity-free algorithms (case of Fig. 1(a))

Position	X [cm]	Y [cm]	Z [cm]	Roll [°]	Pitch [°]	Yaw [°]
	0.0	0.0	12.0	0.0	40.0	0.0
Desired force/moment	0.0 [N]	0.0 [N]	−10.0 [N]	0.0 [Nm]	0.0 [Nm]	0.0 [Nm]
Method	Inverse Jacobian					
Torque [Nm]	−35.45	33.31	35.77	−126.69	−0.35	124.73
Output force	0.0	0.0	−10.0	0.0	0.0	0.0
Method	Nakamura					
Torque [Nm]	−35.45	33.31	35.77	−126.69	−0.35	124.73
Output force	0.0	0.0	−10.0	0.0	0.0	0.0
Method	Chiaverini					
Torque [Nm]	−0.45	0.18	−0.12	−0.05	−0.61	−0.16
Output force	0.0	0.0	−10.0	−0.032	−0.199	0.004
Method	Choi					
Torque [Nm]	−0.45	0.19	−0.13	−0.07	−0.59	−0.18
Output force	0.0	0.0	−10.0	−0.035	−0.196	0.007
Method	SVD damping					
Torque [Nm]	−0.67	0.66	−0.14	−1.49	−0.35	0.22
Output force	0.0	0.0	−10.0	−0.086	−0.113	0.072

Table 2
Simulation result of the four-legged haptic device

Position	X [cm]	Y [cm]	Z [cm]	Roll [°]	Pitch [°]	Yaw [°]		
	0.0	0.0	12.0	0.0	40.0	0.0		
Desired force	0.0 [N]	0.0 [N]	−10.0 [N]	0.0 [Nm]	0.0 [Nm]	0.0 [Nm]		
Method	Redundant actuation							
Torque [Nm]	−0.075	0.0	−0.534	−0.135	0.0	0.0	−0.534	0.135
Output force	0.0	0.0	−10.0	0.0	0.0	0.0		

These phenomena can be explained in two ways. First, the surplus actuators play a role of avoiding singularities by the existence of abundant sources in the column space relating the joint actuators to the output forces. Also, the nonlinear geometry incorporated with the redundant actuation generates large reflection forces.

5. Experimental method and results

The proposed 6 DOF parallel haptic device and its PC-based control system are shown in Fig. 5(a) and (b), respectively. The haptic device is driven by eight dc motors (model: 3863A048C, MiniMotor SA) with encoders (model: HEDM 5500, MiniMotor SA). The kinematic parameters of the developed haptic device are as follows: $R_t = 0.03$ m, $R_{B1} = 0.03$ m, $R_{B2} = 0.0345$ m, and l_i $(i = 1,\ldots,5) = (0.094, 0.131, 0.075, 0.075, 0.045)$ m. To implement a haptic device controller, a Pentium III (800 MHz) computer with real time kernel (RTX) [19] based on Windows NT was used and the control program was coded by the C++ language. The control output and sensor feedback data were updated every 2 ms.

The objective of this experiment was to show that actuator saturation can be avoided near or at singular configurations by using task-priority algorithms as explained in Section 2. Additionally, this objective includes showing how redundant actuation can overcome the singularity problem without any auxiliary algorithms such as task-priority algorithms and consumes smaller torques than the minimum actuation mode using only six motors.

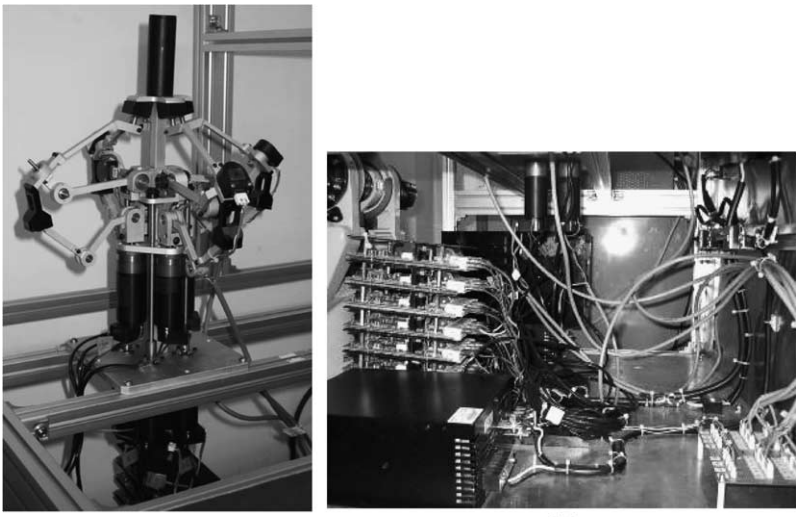

Fig. 5. Proposed haptic device and PC-based control system.

Choi's, SVD damping algorithm and redundant actuation algorithm were tested in this experiment. The first experiment was conducted for the minimum actuated mode (six motors that are distributed to base locations of the legs 1–3) and the second experiment was performed for the redundant actuated mode (eight motors). Primary and secondary tasks are chosen to display -10 N along the Z axis and zero moments, respectively.

The experimental setup is shown in Fig. 6. This setup consists of a 6 DOF force sensor mounted on the top plate of the haptic device with a gripper mounted onto it. While the operator moves the gripper of the haptic device up and down two or three times, the position of the top plate passes through the area of the singular position (0 cm, 0 cm, 12 cm, 0°, 40°, 0°). Forces were measured continuously to check that the given task was properly performed.

Since the motion of a haptic device is generated by a human operator, it is impossible to move the haptic device along the exact trajectory during each experiment. However, noting that the torques needed to perform given tasks rapidly grew as the system configuration approaches the singularity, this phenomenon was used as a means to check whether the device approached a singular point.

Fig. 7 shows the trajectory of position and orientation of the haptic device when Choi's algorithm is applied. The haptic device moves up and down while the pitch angle is maintained as close to 40° as possible. Meanwhile, excessive torques which may cause mechanical damage are generated at the singular position, as shown in the simulation results. For safety of haptic device, we designed the upper ball-socket joints of the top plate in such a way that the singular position is unreachable

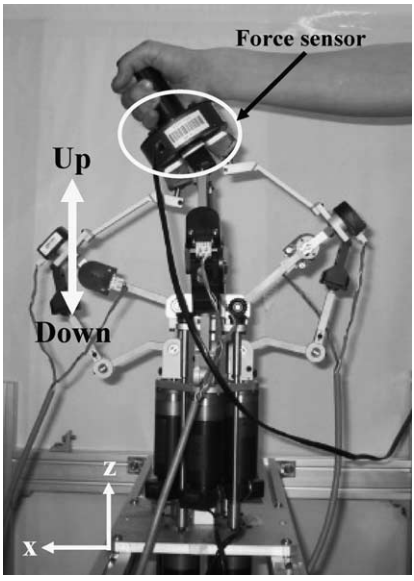

Fig. 6. Experimental setup.

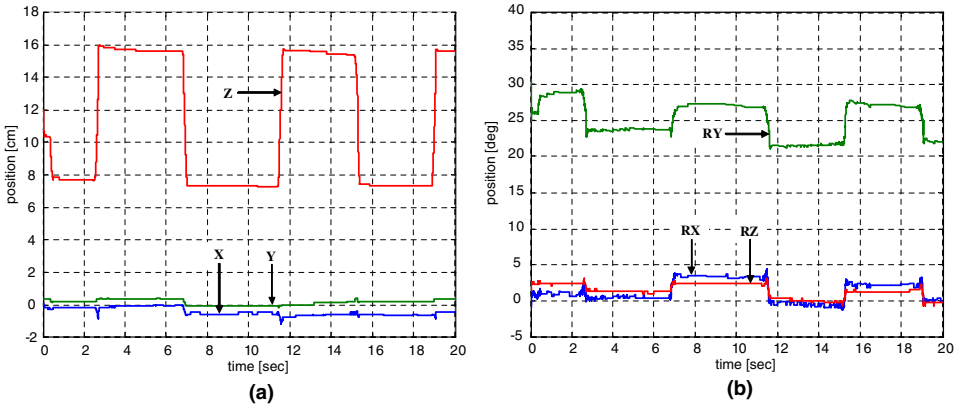

Fig. 7. Trajectory for Choi's algorithm: (a) position and (b) orientation.

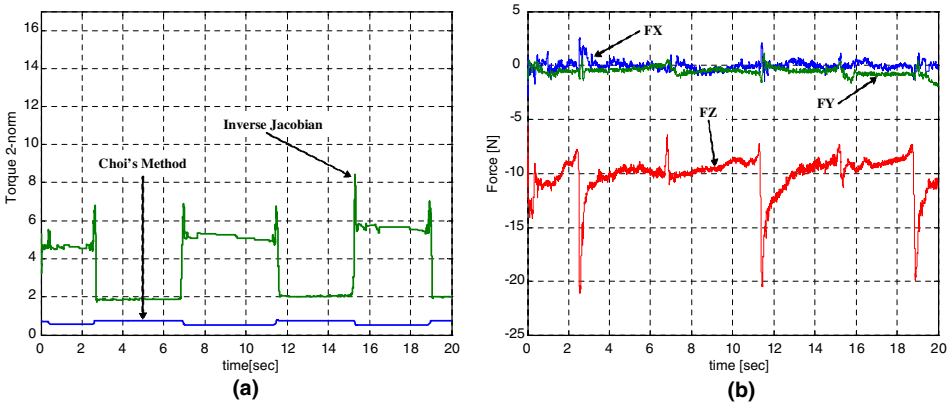

Fig. 8. Torque 2-norm and measured forces for Choi's algorithm: (a) torque 2-norm and (b) measured forces.

mechanically. As a result, the maximum reachable angle around the x and y axis is limited to 35° in the Cartesian space. That was the reason why 35° was the maximum reachable angle according to the posture of haptic device in the experiments.

In Fig. 8(a), the variation of torque 2-norms are displayed for Choi's and Inverse Jacobian algorithms. It can be seen that the peak value of torque 2-norm of Inverse Jacobian algorithm is increased abruptly up to 8.5 when the Z-directional position becomes approximately 12 cm. On the other hand, Choi's algorithm does not reveal this kind of saturation problem.

The given task was to display forces $(FX, FY, FZ) = (0\text{ N}, 0\text{ N}, -10\text{ N})$. It is shown that the measured forces given in Fig. 8(b) are distributed around (0 N, 0N, −10 N). Thus, the performance of Choi's algorithm is satisfactory.

The jerk observed in the FZ plot of Fig. 8(b) is due to the change in direction of motion. When the haptic device starts in an upward motion, instantaneous force

peak over −10 N is measured because the given task is to display −10 N along the downward Z direction. On the other hand, when the haptic device moves downward, a lesser force below −10 N is measured.

Figs. 9 and 10 show the experimental results when the SVD Damping algorithm is employed. The experimental results when the haptic device is operated in redundant actuation mode are shown in Figs. 11 and 12. Experimental processes were the same as those of Choi's algorithm. As it can be seen in Figs. 10(a) and 12(a), torque 2-norm's peak value of Inverse Jacobian algorithm is larger, when the pitch angle is close to 40°. The three experiments described above show that torques are saturated near singularity only when Inverse Jacobian algorithm is used. However, by using task-priority algorithms and redundant actuation algorithm, torque saturation can be avoided and thus the given task can be fully achieved.

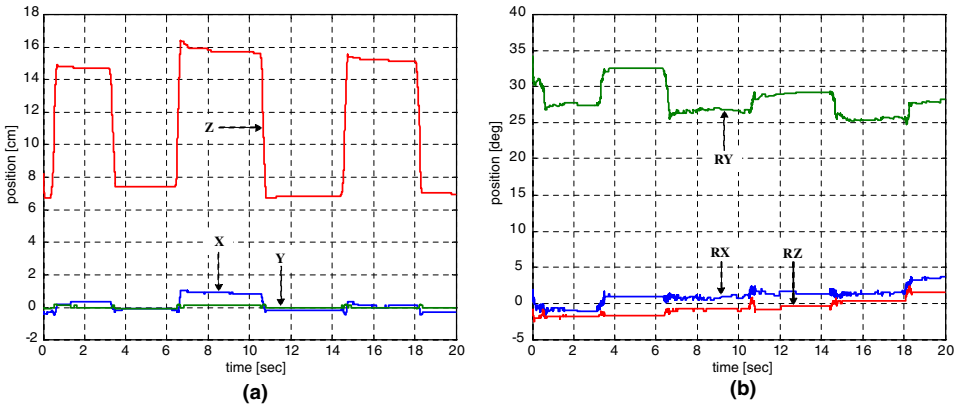

Fig. 9. Trajectory for SVD Damping algorithm: (a) position and (b) orientation.

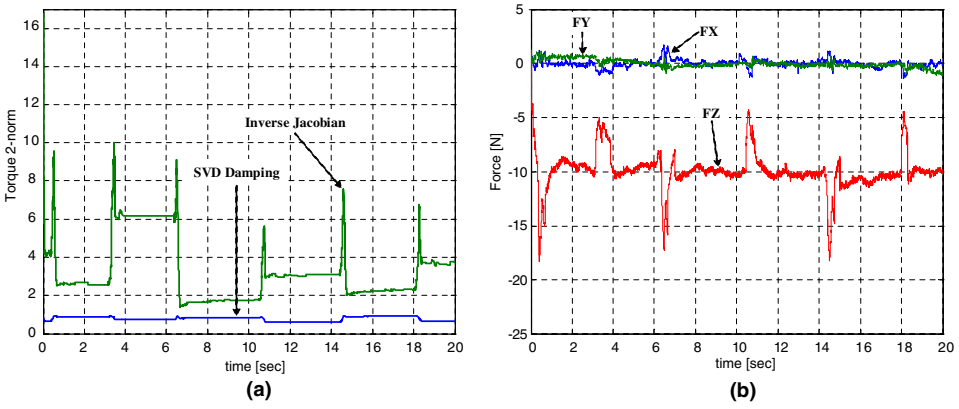

Fig. 10. Torque 2-norm and measured forces for SVD Damping algorithm: (a) torque 2-norm and (b) measured forces.

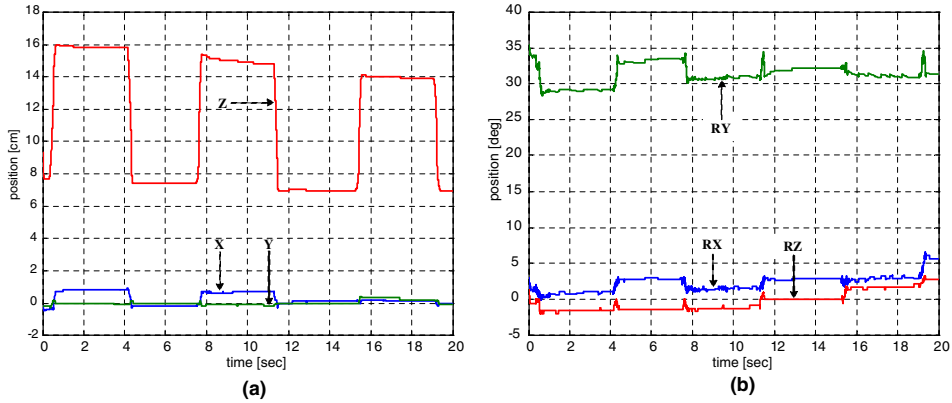

Fig. 11. Trajectory for redundant actuation algorithm: (a) position and (b) orientation.

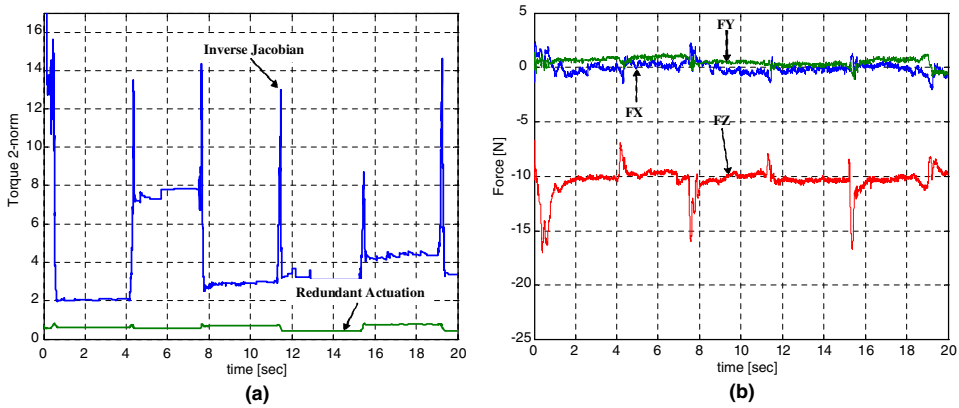

Fig. 12. Torque 2-norm and measured forces for redundant actuation algorithm: (a) torque 2-norm and (b) measured forces.

Fig. 13 shows the comparison of torque 2-norms for the three algorithms. As shown in the figure, redundant actuation requires the smallest torque among the three methods. This result clearly agrees with the simulation result.

From the simulation and experimental results described in the above two sections, some valid conclusions can be drawn as follows:

(1) Inverse Jacobian and Nakamura's algorithm cannot be directly applicable to reflect the desired forces and moments because actuator torques are saturated at the neighborhood of singularities.
(2) Among the task-priority algorithms employing six actuators, Chiaverini's algorithm has the secondary task error. Choi's algorithm is shown to lessen the secondary task error. However, based on the experimental result, their differences are not distinguishable and the secondary error does not affect the primary

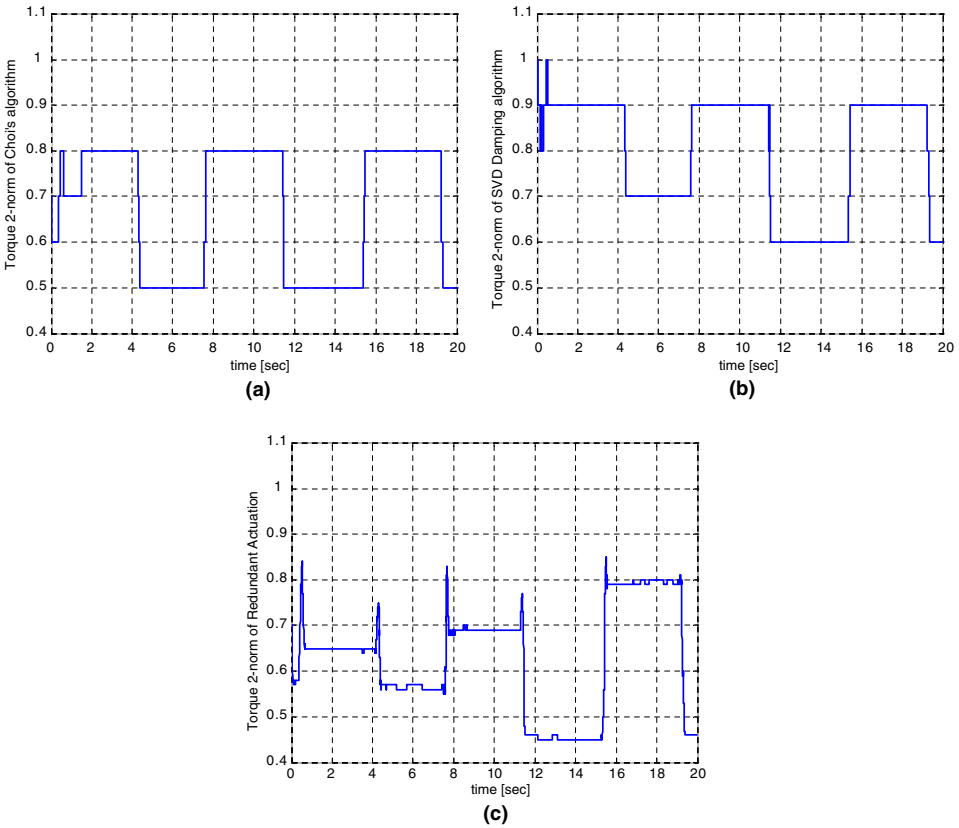

Fig. 13. Comparison of torque 2-norms for the same trajectory: (a) Choi's algorithm and (b) SVD damping algorithm and (c) redundant actuation algorithm.

haptic task that much (pure force reflection capability) in real experimentation. The performance of SVD algorithm is almost the same as the other algorithms employing six actuators.

(3) Task-priority methods such as Chiaverini's, Choi's, and SVD damping algorithms do not cause actuator torque saturation even at the neighborhood of singular position. Choi's algorithm shows better performance than the other two algorithms, but the difference is not distinctive. Thus, those three algorithms may be applicable in real haptic application. Nevertheless, those algorithms based on six actuators have its limitation in aspect of exact force reflection.

(4) However, the algorithms based on six actuators have its limitation in aspect of exact force reflection. A new design including redundant actuation provides robustness to singularities dramatically. And, smaller torques are required to perform the given task as compared to the minimum actuation case. Thus, redundant actuation can be suggested as a good solution to singularity-free haptic operation.

6. Conclusions

In this paper, various singularity-free algorithms for operation of a parallel haptic device were investigated. Although Choi's method resulted in the best performance among the three task-priority algorithms near singular positions in the simulations, the performances of the proposed algorithms were not distinguishable. Therefore, any task-priority algorithm may be applicable to real haptic applications. As an alternative, a new design including redundant actuation was proposed. This mechanism has a singularity-free structure owing to the redundant actuation. The effectiveness of the task-priority algorithms and redundant actuation algorithm is substantiated by the experiments described in this paper.

Acknowledgement

This study was supported by a grant of the Korea Health 21 R&D Project, Ministry of Health and Welfare, Republic of Korea (02-PJ3-PG6-EV04-0003).

References

[1] Goertz RC. Fundamentals of general-purpose remote manipulators. J Nucleon 1952;10(11):36–42.
[2] Ellis RE, Ismaeil OM, Lipsett MG. Design and evaluation of high-performance haptic interface. Robotica 1996;4.
[3] Massie T, Salisbury K. PHANToM haptic interface: a device for probing virtual objects. ASME J Dynam Syst Contr 1994:295–9.
[4] Berkelman PJ, Hollis RL, Salcudean SE. Interacting with virtual environments using a magnetic levitation haptic interface. Proc IEEE Internat Conf Intell Robots Syst 1995:117–22.
[5] Bergamasco M, Allotta B, Bosio L, Ferretti L, Parrini G, Prisco GM et al. An arm exoskeleton system for teleoperation and virtual environments applications. Proc IEEE Internat Conf Robot Autom 1994:1449–54.
[6] Ishii M, Sato M. A 3D spatial interface device using tensed strings. Presence-teleoperators and virtual environments, 3(1). Cambridege, MA: MIT Press; 1994. p. 81–6.
[7] Hollerbarch JM. Some current issues in haptic research. In Proceedings of the IEEE International Conference on Robotics and Automation; 2000. p. 757–62.
[8] Lee JH, Yi B-J, Oh S-R. Suh IH. Optimal design of a five-bar finger with redundant actuation. In: Proceedings of the IEEE International Conference on Robotics and Automation; 1998. p. 2068–74.
[9] Wampler CW. Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods. IEEE Trans Syst Man Cybernet 1986;SMC-16(1):93–101.
[10] Maciejewski AA, Klein CA. Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments. Int J Robot Res 1985;4(3):109–17.
[11] Nakamura Y, Hanafusa H, Yoshikawa T. Task-priority based redundancy control of robot manipulators. Int J Robot Res 1987;6(2):3–15.
[12] Chiaverini S. Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. IEEE Trans Robot Automat 1997;13(3):398–410.
[13] Choi YJ, Chung WK, Oh YW, Oh S-R, Suh IH. On the task priority manipulation scheme with high execution performance for a robotic manipulator. International Conference on Advanced Robotics; 2001.
[14] Lee JH, Eom KS, Yi B-J, Suh IH. Design of a new 6-DOF parallel haptic device. In: Proceedings of the IEEE International Conference on Robotics and Automation; 2001. p. 886–891.

[15] Kim HW, Eom KS, Suh IH, Yi B-J. A transparency-optimized control for a new 6 DOF parallel-structured haptic device. In: Proceedings of the IEEE International Conference on Robotics and Automation; 2001. p. 2331–6.
[16] Gosselin, CM, Allan JF, Lalibert T. A new architecture for a high-performance 6-DOF parallel mechanism. In: Proceedings of the IFTOMM Tenth World Congress on the Theory of Machine and Mechanisms; 1999. p. 1140–5.
[17] Pierrot F, Marquet F, Olivier C, Gil T. H4 Parallel robot: modeling, design and preliminary experiments. In: Proceedings of the IEEE International Conference on Robotics and Automation; 2001. 3256–61.
[18] Tsai L-W, Walsh GC, Stamper RE. Kinematics of a novel three DOF translational platform. In Proceedings of the IEEE International Conference on Robotics and Automation; 1996. p. 3446–51.
[19] RTX 4.3 Release notes, VenturCom, Inc.; 1999.