Proceedings of the 2004 IEEE
International Conference on Robotics & Automation
New Orleans, LA • April 2004

Singularity-free Load Distribution Algorithms for a 6 DOF Parallel Haptic Device

Hyung Wook Kim
Department of Electronics Engineering,
Hanyang University,
Ansan, Korea
Email: hwkim@incorl.hanyang.ac.kr

Jae Hoon Lee, Byung-Ju Yi, and Il Hong Suh
School of Electrical Engineering and
Computer Science, Hanyang University,
Ansan, Korea
Email: robot@ihanyang.ac.kr
{bj, ihsuh}@hanyang.ac.kr

Abstract— **It is known that parallel type mechanisms have many singularities than serial type mechanisms. In haptic application, these singularities deteriorate the force reflecting performance. Moreover, different from general manipulators, haptic systems can't avoid the singular point, because they are operated by the user's random motion command.**

Although many singularity–free load distribution algorithms for kinematically redundant manipulators have been proposed, singularity-free load distribution algorithms for parallel haptic application have not been deeply discussed. In this paper, various singularity–free load distribution algorithms that are appropriate to parallel haptic system will be discussed. A new 6 DOF parallel mechanism equipped with four sub–chains is employed as a test device. To cope with singularity problem, four task–priority algorithms and redundant actuation algorithm are discussed and compared through simulation, and also experiments have been performed to show the effectiveness of those algorithms at the presence of singularity.

I. Introduction

Since master-slave system was proposed by Goertz in the 1950's [1], many researchers have developed various types of haptic display 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 magnetic levitation haptic interface by Berkelman *et al.* [4], etc. Although many haptic devices have been developed as above, the force reflection capability is still confined to three or less DOF, which is not enough to display the reality in virtual environment. Moreover, precise and hard contact feeling is also hard to realize.

Parallel mechanisms have advantages over serial mechanisms in aspect of high structural stiffness, low inertia, and high force bandwidth. However, haptic devices having parallel structure are bulky, have relatively small workspace, and have multiple forward kinematic solutions. Also they require large power due to floating actuators [2], [6]–[8]. To overcome these difficulties, Gosselin [16], Lee *et al.* [14], [15], Pierrot [17], and Tsai [18] proposed new parallel mechanisms, which are light by employing non–floating actuators and has relatively large workspace as compared to the previous parallel mechanisms.

Although parallel mechanism has many advantages as a haptic device that requires high structural stiffness, high force bandwidth, and high force dynamic range, it is known that parallel mechanisms have many singularities that degrade the force reflection performance in haptic applications. At singular points, the haptic system can't generate the desired reflection force completely, and moreover, torque saturation happens. Thus, singularity–free algorithm to maintain system performance that generates the reflection force precisely needs to be investigated.

Many singularity–free algorithms for kinematically redundant mechanisms have been investigated extensively [9]–[13]. However, researches on singularity–free algorithms for parallel type haptic devices have been rare. In this paper, four kinds of singularity-free load distribution algorithms will be examined to cope with the singularity problem of parallel haptic systems. Also, as a new scheme to overcome the limitation of these algorithms, a new design including redundant actuation is proposed.

This paper is organized as follows. Section II shows the overview of task-priority algorithms that are used to overcome the singularity problem, and the analysis for each algorithm is performed in section III. In section IV, experimental results are discussed and some conclusions are reported in section V.

II. Singularity–free Algorithms

In general, parallel mechanisms have quite a few singular points in their workspace. Singularity causes serious problems such as torque saturation, undesirable motion, and breakdown of systems. More importantly, the singular point can't be avoided in haptic system because the haptic device is operated by the user's random motion command. For serial type manipulators, several motion planning algorithms have been reported to avoid singularities by using kinematic redundancy. However, singularity–free load distribution algorithms for parallel device have not been deeply discussed yet. Thus, singularity-free load distribution algorithms adequate to parallel devices need to be investigated.

Fig. 1 shows three examples of singular configuration 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 another singular position where the system touches the boundary of the workspace. Fig. 1(c) shows an algorithmic singularity caused by internal kinematic problem.

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

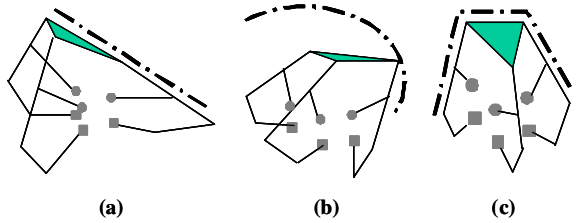

Fig. 1. Singular configurations

A. Inverse Jacobian

The force relationship between the joint torque $\mathbf{T}_A \in \mathbf{R}^n$ and the output force/torque vector $\mathbf{T}_u \in \mathbf{R}^m$ is expressed as

$$\mathbf{T}_u = \mathbf{J}\mathbf{T}_A, \tag{1}$$

where $\mathbf{J}$ denotes the Jacobian matrix. If m is equal to n, the inverse relation of (1) is given by

$$\mathbf{T}_A = \mathbf{J}^{-1}\mathbf{T}_u, \tag{2}$$

assuming that $\mathbf{J}$ is invertible.

B. Nakamura's Algorithm

To begin with, consider Nakamura's task priority algorithm [11]. The subtask having the first priority will be specified using the first manipulation force and the secondary priority subtask will be specified using the second manipulation force.

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

$$\mathbf{T}_{u1} = \mathbf{J}_1\mathbf{T}_A, \tag{3}$$

where $\mathbf{J}_1$ denotes the Jacobian matrix that relates the first manipulating force to the joint torque vector.

The general solution of (3) is expressed as

$$\mathbf{T}_A = \mathbf{J}_1^+\mathbf{T}_{u1} + \{\mathbf{I} - \mathbf{J}_1^+\mathbf{J}_1\}\mathbf{z}, \tag{4}$$

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

$$\mathbf{T}_{u2} = \mathbf{J}_2\mathbf{T}_A, \tag{5}$$

where $\mathbf{J}_2 \in \mathbf{R}^{m_2 \times n}$ is the Jacobian matrix for the secondary task. If we try to minimize the secondary task error $\|\mathbf{T}_{u2} - \mathbf{J}_2\mathbf{T}_A\|_2$ in the least square sense, then the least square solution $\mathbf{z}$ is obtained as follow [11]:

$$\mathbf{z} = \tilde{\mathbf{J}}_2^+\{\mathbf{T}_{u2} - \mathbf{J}_2\mathbf{J}_1^+\mathbf{T}_{u1}\} + \{\mathbf{I} - \tilde{\mathbf{J}}_2^+\tilde{\mathbf{J}}_2\}\mathbf{x}, \tag{6}$$

where $\tilde{\mathbf{J}}_2 = \mathbf{J}_2\{\mathbf{I} - \mathbf{J}_1^+\mathbf{J}_1\}$.

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

$$\begin{aligned}\mathbf{T}_A \;=\; & \mathbf{J}_1^+\mathbf{T}_{u1} + \tilde{\mathbf{J}}_2^+\{\mathbf{T}_{u2} - \mathbf{J}_2\mathbf{J}_1^+\mathbf{T}_{u1}\} + \\ & \{\mathbf{I} - \tilde{\mathbf{J}}_2^+\tilde{\mathbf{J}}_2\}\mathbf{x},\end{aligned} \tag{7}$$

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

C. Chiaverini's Algorithm

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

$$rank(\mathbf{J}_1) < m_1 \quad \text{or} \quad rank(\mathbf{J}_2) < m_2. \tag{8}$$

The algorithmic singularity occurs in either [12]

$$\mathbf{R}(\mathbf{J}_1^+) \cap \mathbf{R}(\mathbf{J}_2^+) \neq \emptyset \tag{9}$$

or

$$\mathbf{N}(\mathbf{J}_1) \cap \mathbf{N}(\mathbf{J}_2) \neq \emptyset, \tag{10}$$

when $m_2 \leq n - m_1$.

The kinematic singularity is the fundamental problem in solving inverse kinematics, but the algorithmic singularity is an artificial part and can be eliminated or changed according to the characteristics of the employed algorithm. Whatever the kinds of singularities, it should be noted that the general solution of (7) is not acceptable near singularities [12].

To eliminate the algorithmic singularities existing in Nakamura's method, Chiaverini [12] modified (7) as the following form

$$\mathbf{T}_A = \mathbf{J}_1^+\mathbf{T}_{u1} + \{\mathbf{I} - \mathbf{J}_1^+\mathbf{J}_1\}\{\mathbf{J}_2^+\mathbf{T}_{u2} + \{\mathbf{I} - \mathbf{J}_2^+\mathbf{J}_2\}\mathbf{z}\}. \tag{11}$$

Although Nakamura's algorithm given in (7) has algorithmic singularities, the primary and the secondary tasks do not have any task error in normal case. On the contrary, Chiaverini's algorithm given in (11) has no algorithmic singularity, but it always has the secondary task error except when $\mathbf{J}_2\mathbf{J}_1^+ = 0$. Thus, there are trade-off between the two algorithms.

D. Choi's Algorithm

By using the advantage that the Chiaverini's algorithm has no algorithmic singularity, Choi *et al.* [13] proposed an algorithm that reduces the secondary task error found in (11). The Chiaverini's algorithm is modified to the following equation.

$$\mathbf{T}_A = \mathbf{J}_W^+ \mathbf{T}_{u1} + \{\mathbf{I} - \mathbf{J}_W^+ \mathbf{J}_1\}\{\mathbf{J}_2^+ \mathbf{T}_{u2} + \{\mathbf{I} - \mathbf{J}_2^+ \mathbf{J}_2\}\mathbf{z}\}, \quad (12)$$

where $\mathbf{J}_W^+ = \mathbf{W}^{-1}\mathbf{J}_1^T(\mathbf{J}_1\mathbf{W}^{-1}\mathbf{J}_1^T)^{-1}$. If the weight matrix is chosen to be a positive definite matrix as follows

$$\mathbf{W} = \mathbf{J}_1^T\mathbf{J}_1 + \mathbf{J}_2^T\mathbf{J}_2 + \epsilon\mathbf{I} > 0, \quad (13)$$

it has no algorithmic singularities, though (12) more of less contaminates the performance of the secondary task with small positive number ϵ.

E. Damped Least Square Method

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

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \quad (14)$$

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

$$\mathbf{\Sigma} = \begin{pmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \quad (15)$$

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

Then, the pseudoinverse is given by

$$\mathbf{A}^+ = \mathbf{V}\mathbf{\Sigma}^+\mathbf{U}^T, \quad (16)$$

where

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

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

$$\mathbf{A}_\alpha^+ = \mathbf{V}\mathbf{\Sigma}_\alpha\mathbf{U}^T, \quad (17)$$

where

$$\mathbf{\Sigma}_\alpha = \begin{pmatrix} \mathbf{S}_\alpha \\ \mathbf{0} \end{pmatrix} \in \mathbf{R}^{n\times m} \quad (18)$$

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

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

F. Redundant Actuation Algorithm

When the number of actuators is greater than that of output(i.e., $n > m$), the general solution of (1) is given by

$$\mathbf{T}_A = \mathbf{J}^+\mathbf{T}_u + \{\mathbf{I} - \mathbf{J}^+\mathbf{J}\}\epsilon, \quad (19)$$

where $\mathbf{J}^+$ is a pseudo-inverse solution of $\mathbf{J}$ and the second term denotes a homogeneous solution. The first particular solution will be employed in the simulation study.

III. Simulation Results

At singular configuration, parallel haptic device can't generate the desired 6 DOF force/moment completely and also torque saturation may happen. In such case, the user may choose the priority of tasks according to special purpose and condition. In this paper, 3 DOF force and 3 DOF moment are chosen to be the first and second priority task, respectively.

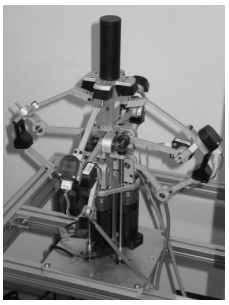

Fig. 2. 6 DOF haptic device

A 6 DOF haptic device shown in Fig. 2, which consists of four chains, is chosen as an examplary device. Each chain is driven by two actuators. The performances of three singularity–free algorithms and redundant actuation algorithm are tested for this device. Firstly, actuator torques required to perform the given task are computed. Then, the output force and moment are recalculated by (3) and (5) to check the task error. In all simulations, the primary task is selected to display –10N along the Z axis and the secondary task is selected to yield zero moment.

Table I shows the simulation results when the five algorithms are applied to the parallel haptic device, which is actuated only by three chains' actuators(i.e., 6 actuators). The system is in the neighborhood of a singular configuration as shown in Fig. 1(a) (0cm, 0cm, 12cm, 0°, 40°, 0°). In such 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 there are some errors in the secondary task.

Table II shows the simulation result of the 4 chained parallel haptic device, which is driven by eight motors. Actuator saturation is not found at the same singular configuration. Furthermore, in such redundantly actuated case 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 advantage in aspect of singularity avoidance and saving input norm.

TABLE I

COMPARISON OF SINGULARITY-FREE ALGORITHMS (CASE OF FIG. 1(A))

Position	X [cm]	Y [cm]	Z [cm]	Roll [Degree]	Pitch [Degree]	Yaw [Degree]
	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 II

SIMULATION RESULT OF THE 4 CHAINED HAPTIC DEVICE

Position	X [cm]	Y [cm]	Z [cm]	Roll [Degree]	Pitch [Degree]	Yaw [Degree]
	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 trend can be explained as follows. The surplus actuators play a role of avoiding singularities by abundant sources existing in the column space that relates the joint actuators to the output forces. Moreover, the nonlinear geometry incorporated with the redundant actuation generates larger reflection forces even with small amount of torque.

IV. EXPERIMENTAL RESULTS

The haptic device shown in Fig. 2 is driven by eight dc motors(model : 3863A048C, MiniMotor SA) with encoders(model : HEDM 5500, MiniMotor SA). The link parameters of the developed haptic device are as follows : $l_i(i = 1, \cdots, 5)$=(0.094, 0.131, 0.075, 0.075, 0.045)m. Fig. 3 shows the structure of haptic controller. To implement the controller of the haptic device, a Pentium III(800MHz) computer with real time kernel(RTX) [19] based on Windows NT is used and the control program is coded by C++ language. The control output and sensor feedback data are updated every 2msec.

The objective of experiment is to show that actuator saturation can be avoided near or at singular configurations by using task–priority algorithms as explained in section 2, and that redundant actuation can overcome the singularity problem without any auxiliary algorithms such as those task–priority algorithms, and that it consumes smaller torque than the minimum actuation mode using only six motors.

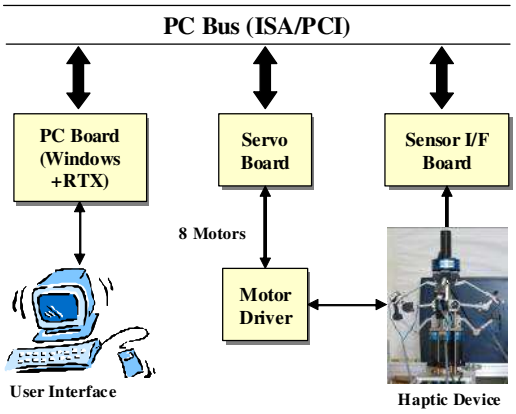

Fig. 3. Block diagram of haptic device controller

Choi's, SVD Damping algorithm, and redundant actuation algorithm are tested in this experiment. First experiment is conducted for minimum actuation mode (six motors) and the second experiment is performed for redundantly actuated mode (eight motors). Primary and secondary tasks are chosen to display –10N along the Z axis and zero moments, respectively.

Fig. 4 shows the experimental setup. A 6 DOF force sensor is mounted on the top plate of the haptic device and a gripper is mounted on 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 neighborhood of a singular position(0cm, 0cm, 12cm, 0°, 40°, 0°). Forces are measured continuously to check that the desired reflection force is properly felt. The given task is to display forces (FX, FY, FZ)=(0N, 0N, –10N).

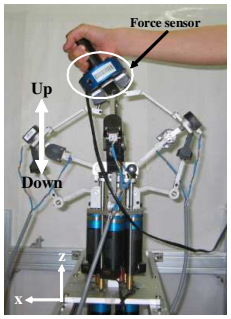

Fig. 4. Experimental setup

Since the motion of a haptic device is generated by the human operator, it's impossible to move the haptic device along exactly the same trajectory at each experiment. However, noting that the torque values needed to perform the given task grow rapidly as the system configuration approaches the singularity, this phenomenon will be used as a means to check whether the haptic device approaches a singular point or not.

Figs. 5 and 6 show the trajectories 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 as 40°. Fig. 7 displays the torque norms of Choi's and Inverse Jacobian algorithms. It can be seen that the peak value of the torque 2–norm of Inverse Jacobian algorithm is increased abruptly up to 8.5, when the haptic device passes through a singular position(i.e., 12cm in the Z-coordinate). On the other hand, the Choi's algorithm does not reveal this kind of the saturation problem.

It is shown that the measured forces given in Fig. 8 are distributed around (0N, 0N, –10N). Thus, the performance of Choi's algorithm is satisfactory.

The jerk observed in the FZ plot of Fig. 8 is due to the change of motion direction. When the haptic device starts upward motion, instantaneous force peak over –10N is measured, because the given task is to display –10N along the downward Z direction. On the other hand, when the haptic device moves downward, smaller force than –10N is measured.

Figs. 9, 10, 11, and 12 show the experimental results when the SVD Damping algorithm is employed. And, Figs. 13, 14, 15, and 16 show the experimental results when the haptic device is operated in redundant actuation mode. Experimental process is the same with that of Choi's algorithm. As it can be seen in Fig. 11 and Fig. 15, the peak value of the torque 2–norm of the Inverse Jacobian algorithm is large. Three experiments described as above show that torques are saturated near singularity when only Inverse Jacobian algorithm is used. However, by using task-priority algorithms and redundant actuation, torque saturation can be avoided and the given task can be fully achievable.

Fig. 17 shows the comparison of torque 2–norms for the three algorithms. As it can be seen in the figure, redundant actuation requires the smallest torque among the three methods. And, this result agrees well with the simulation result.

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

1) Inverse Jacobian and Nakamura's algorithm can't be directly applicable to reflect the desired forces and moments because actuator torques are saturated at the neighborhood of singularities.
2) 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.
3) 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 load distribution methods based on minimum actuation. Thus, redundant actuation can be suggested as a good solution to singularity–free haptic operation.

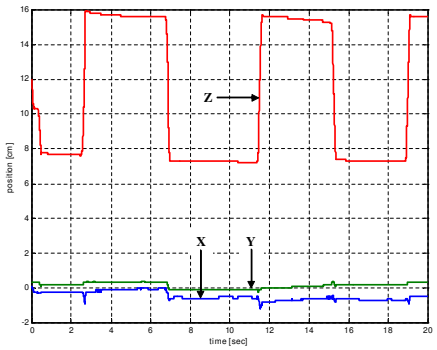

Fig. 5. Trajectory for Choi's algorithm : Position

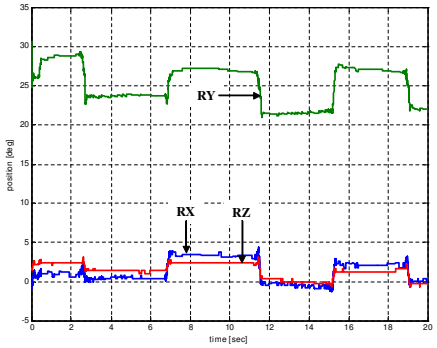

Fig. 6. Trajectory for Choi's algorithm : Orientation

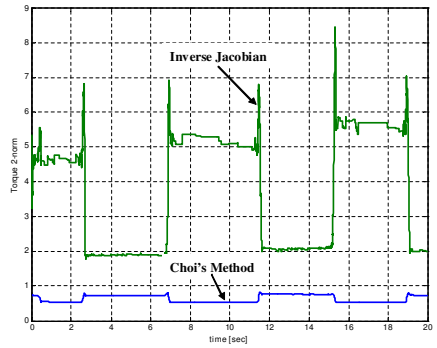

Fig. 7. Torque 2–norm for Choi's algorithm

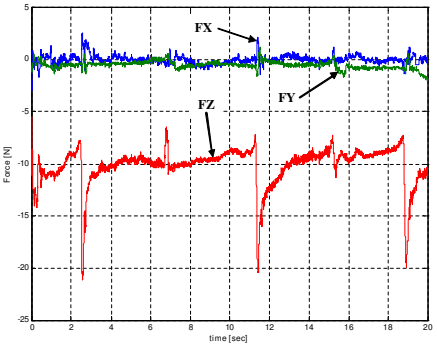

Fig. 8. Measured force for Choi's algorithm

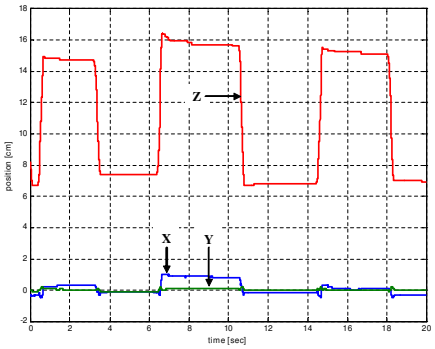

Fig. 9. Trajectory for SVD Damping algorithm : Position

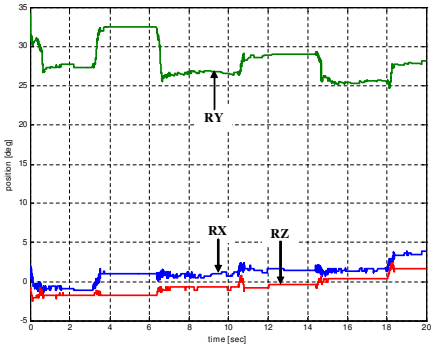

Fig. 10. Trajectory for SVD Damping algorithm : Orientation

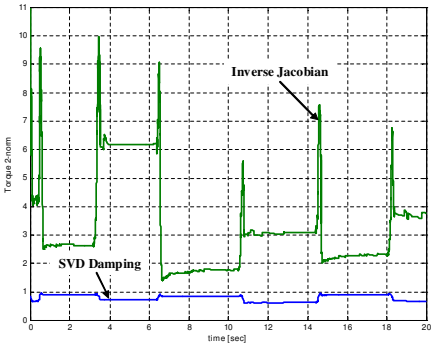

Fig. 11. Torque 2–norm for SVD Damping algorithm

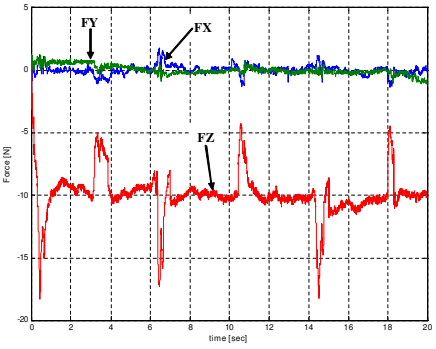

Fig. 12. Measured force for SVD Damping algorithm

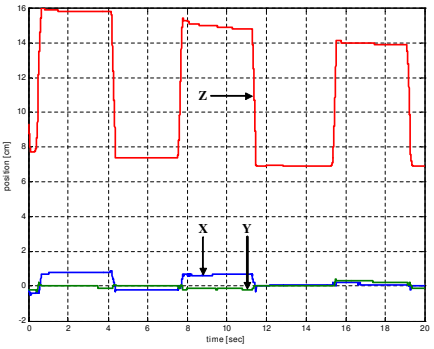

Fig. 13. Trajectory for redundant actuation algorithm : Position

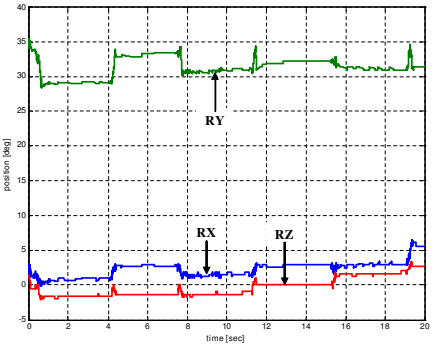

Fig. 14. Trajectory for redundant actuation algorithm : Orientation

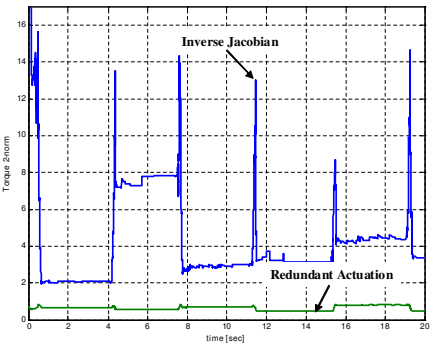

Fig. 15. Torque 2–norm for redundant actuation algorithm

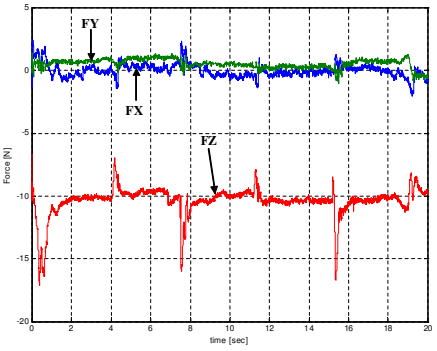

Fig. 16. Measured force for redundant actuation algorithm

V. Conclusions

In this paper, various singularity–free load distribution algorithms for operation of parallel haptic device are investigated. Although Choi's method shows the best performance among the three load distribution algorithms near singular positions in simulation work, the performances of the proposed algorithms are not distinguishable. So, any load distribution algorithm may be applicable in real haptic application. As an alternative, redundant actuation can be suggested as a good solution to singularity–free haptic operation. The effectiveness of the load distribution algorithms and redundant actuation algorithm is shown by experiment.

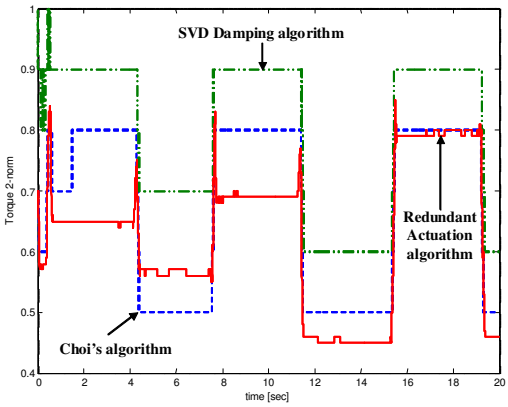

Fig. 17. Comparison of torque 2–norms

Acknowledgment

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

References

[1] R. C. Goertz, "Fundamentals of General–Purpose Remote Manipulators," Journal of Nucleonics, Vol. 10, No. 11, pp. 36–42, 1952.

[2] R. E. Ellis, O. M. Ismaeil, and M. G. Lipsett, "Design and Evaluation of High–Performance Haptic Interface," Robotica, Vol. 4, 1996.

[3] T. Massie and K. Salisbury, "PHANToM Haptic Interface : A Device for Probing Virtual Objects," ASME Journal of Dynamic System and Control, pp. 295–299, 1994.

[4] P. J. Berkelman, R. L. Hollis, and S. E. Salcudean, "Interacting with Virtual Environments using a Magnetic Levitation Haptic Interface," Proc. of IEEE Int. Conf. on Intelligent Robots and Systems, pp. 117–122, 1995.

[5] M. Bergamasco, B. Allotta, L. Bosio, L. Ferretti, G. Parrini, G. M. Prisco, F. Salsedo, and G. Sartini, "An Arm Exoskeleton System for Teleoperation and Virtual Environments Applications," Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 1449–1454, 1994.

[6] M. Ishii and M. Sato, "A 3D Spatial Interface Device Using Tensed Strings," Presence–Teleoperators and Virtual Environments, MIT Press, Vol. 3, No. 1, pp. 81–86, 1994.

[7] J. M. Hollerbarch, "Some Current Issues in Haptic Research," Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 757–762, 2000.

[8] J. H. Lee, B. –J. Yi, S. –R. Oh, and I. H. Suh, "Optimal Design of a Five–Bar Finger with Redundant Actuation," Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 2068–2074, 1998.

[9] C. W. Wampler, "Manipulator Inverse Kinematic Solutions Based on Vector Formulations and Damped Least-Squares Methods," IEEE Trans. on Systems, Man, and Cybernetics, Vol. SMC–16, No. 1, pp. 93–101, 1986.

[10] A. A. Maciejewski and C. A. Klein, "Obstacle Avoidnace for Kinematically Redundant Manipulators in Dynamically Varying Environments," Int. Journal of Robotics Research, Vol. 4, No. 3, pp. 109–117, 1985.

[11] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-Priority based Redundancy Control of Robot Manipulators," Int. Journal of Robotics Research, Vol. 6, No. 2, pp. 3–15, 1987.

[12] S. Chiaverini, "Singularity–robust Task–priority Redundancy Resolution for Real-time Kinematic Control of Robot Manipulators," IEEE Trans. on Robotics and Automation, Vol. 13, No. 3, pp. 398–410, 1997.

[13] Y. J. Choi, W. K. Chung, Y. W. Oh, S. –R. Oh, and I. H. Suh, "On the Task Priority Manipulation Scheme with High Execution Performance for a Robotic Manipulator," Int. Conf. on Advanced Robotics, 2001.

[14] J. H. Lee, K. S. Eom, B. –J. Yi, and I. H. Suh, "Design of A New 6–DOF Parallel Haptic Device," Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 886–891, 2001.

[15] H. W. Kim, K. S. Eom, I. H. Suh, and B. –J. Yi, "A Transparency-optimized Control for a New 6–DOF Parallel-structured Haptic Device," Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 2331–2336, 2001.

[16] C. M. Gosselin, J. F. Allan, and T. Lalibert, "A New Architecture for a High–performance 6-DOF Parallel Mechanism," Proc. of IFTOMM 10th World Congress on the Theory of Machine and Mechanisms, pp. 1140–1145, 1999.

[17] F. Pierrot, F. Marquet, C. Olivier, and T. Gil, "H4 Parallel Robot : Modeling, Design and Preliminary Experiments," Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 3256–3261, 2001.

[18] L. –W. Tsai, G. C. Walsh, and R. E. Stamper, "Kinematics of a Novel Three DOF Translational Platform," Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 3446–3451, 1996.

[19] RTX 4.3 Release Notes, VenturCom, Inc., 1999.