

# Novel Collision Detection Index based on Joint Torque Sensors for a Redundant Manipulator

Sang-Duck Lee, Young-Loul Kim and Jae-Bok Song *Member, IEEE*

**Abstract**— Human-robot collision has drawn increasing attention in recent years and collision safety can be improved by successfully detecting collisions between a human and a robot. For a manipulator working in human environments, collisions usually occur at the manipulator body while the robot performs a contact task using its end-effector to interact with the environment. Therefore, both collision force and the force on the end-effector contribute to the external torques which can be estimated from the robot dynamics and the joint torques measured by the joint torque sensors, which means whether or not a collision has occurred cannot be reliably determined using this estimation. In this study, we propose a novel collision detection index to detect collisions independently of the end-effector force of a redundant manipulator equipped with joint torque sensors. Using the null space projection of a redundant manipulator, the collision detection index can be expressed as a function of the torque generated by a collision and the manipulator configuration. The proposed index is verified through various simulations. Simulation results show that collisions can be reliably detected regardless of the presence of the end-effector forces even in situations with external torques contaminated by substantial error.

## I. INTRODUCTION

In many industrial fields, human-robot cooperation has been increasing, and expansion of the market for service robots is expected. Since robot manipulators and service robots share the workspace with humans, robots can harm humans due to human error or malfunctions. For these reasons, the importance of human-robot collision safety has begun to be raised recently. To cope with this problem, several solutions have been proposed, such as the use of skin sensors [1], installation of a mechanism which can absorb impact forces [2, 3], the use of a vision sensor to avoid collisions [4], and the design and control of manipulators based on collision analysis and safety evaluation [5, 6]. Although these solutions have been used to improve collision safety, there are some disadvantages. The use of skin sensors may increase the cost of a robot significantly; adding additional mechanisms is a rather passive solution, which lacks flexibility and would

increase the size and complexity of a robot; vision sensors have limitations, such as blind spots and heavy computational load in dynamic environments; and merely using safety evaluations to design and control systems requires a tradeoff between task performance and collision safety. Meanwhile, improvement of collision safety through collision detection has several advantages; it requires no design changes, no installation of extra sensors or mechanisms is necessary, it has relatively high reliability, and it does not affect task performance. Therefore, several studies have been conducted on collision detection algorithms

A manipulator is usually equipped with a force/torque sensor at its wrist to measure the end-effector force and moment. This sensor is necessary for tasks which require force control or a contact between the end-effector and the environment. However, this sensor suffers from high cost and cannot detect a body collision which occurs at the body parts of a manipulator (not the end-effector). To cope with these problems, some manipulators are equipped with joint torque sensors (JTS) which are mounted at each joint of the manipulator and sense the joint torques delivered to the link. Although the external force (including the moment) acting on the end-effector cannot be directly measured by JTS, this force can be estimated from the dynamic model of a manipulator and JTS-measured joint torques. Unlike the F/T sensor attached to the end-effector, the JTS-based approach can detect a body collision. Some collision detection algorithms used a disturbance observer which required heavy computational load and the use of acceleration sensors which are often noisy and inaccurate [7-9]. Instead of acceleration, the generalized momentum was adopted in [10, 11]. However, these collision detection schemes fail to detect a collision when a body collision occurs together with the force interaction at the end-effector because these two forces cannot be distinguished.

To solve this problem, a novel collision detection index for a redundant manipulator is proposed in this study to detect a body collision regardless of whether or not the end-effector is subject to a force interacting with the environment. The remainder of this paper is organized as follows. Section 2 explains some preliminaries for understanding the proposed collision detection index which is proposed in Sec. 3. Section 4 verifies the proposed collision detection index using Matlab-based simulations. Finally, our conclusions are drawn in Sec. 5.

\*Research supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2007-0056650) and supported by the Human Resources Development Program for Convergence Robot Specialists (Ministry of Knowledge Economy).

Sang-Duck Lee is with the School of Mechanical Eng., Korea University, Seoul, Korea (e-mail: ysdchris@korea.ac.kr)

Young-Loul Kim is with the School of Mechanical Eng., Korea University, Seoul, Korea (e-mail: kimloul@korea.ac.kr)

Jae-Bok Song (corresponding author) is a Professor of the School of Mechanical Eng., Korea University, Seoul, Korea (Tel.: +82 2 3290 3363; fax: +82 2 3290 3757; e-mail: jbsong@korea.ac.kr)

## II. PRELIMINARIES

Consider the following equation of motion of a manipulator

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + \tau_{ext} = \tau_m \quad (1)$$

where  $M(q)$  is the inertia matrix,  $C(q, \dot{q})$  is the matrix containing Coriolis and centrifugal terms,  $g(q)$  is the gravity vector, and  $\tau_m$  is the actuator torque vector. The external torque vector  $\tau_{ext}$  is described by

$$\tau_{ext} = \tau_e + \tau_c \quad (2)$$

where  $\tau_e$  is the external torque vector generated by a task force  $F_e$  acting on the end-effector and  $\tau_c$  is the external torque vector induced by a collision force  $F_c$  which is acting on parts other than the end-effector, as shown in Fig. 2. We assume that a collision (generating  $F_c$ ) occurs while a manipulator is performing a task (using  $F_e$ ).

If  $\tau_c$  can be measured or computed, we can detect a human-robot collision. However, the joint torque sensors embedded at each joint of a manipulator can measure only the actuator torque  $\tau_m$ . Therefore, the total external torque  $\tau_{ext}$ , which contains  $\tau_c$ , should be estimated as follows:

$$\tau_{ext} = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) - \tau_m \quad (3)$$

However, acceleration measurements are often noisy and inaccurate, and thus a residual observer is used in this study. The residual  $r$ , used to estimate  $\tau_{ext}$ , is given by

$$r(t) = K \left[ \int_0^t (C(q, \dot{q})^T \dot{q} - g(q) + \tau_m - r) dt - M(q)\dot{q} \right] \quad (4)$$

where  $K$  is the cutoff frequency of the observer [8]. Note that the residual can be computed using the joint torques  $\tau_m$  measured by the joint torque sensors, and the joint positions and velocities obtained from encoders; it does not require acceleration or complex computations (e.g., inverse matrix). Thus, this algorithm can be applied to any robot equipped with joint torque sensors. If the control period of the manipulator is fast enough,  $r$  follows  $\tau_{ext}$  quite well [11], and thus we can assume that  $r = \tau_{ext}$ . In the next section, we discuss how  $\tau_c$  can be computed from  $\tau_{ext}$ .

## III. COLLISION DETECTION INDEX

### A. Null space projection of joint torque

Consider a linear mapping between the joint torque space  $S_\tau$  and the end-effector force space  $S_F$ . This mapping can be described by  $J^T$  or  $(J^T)^+$ , where  $J$  is the  $(m \times n)$  Jacobian matrix, and the superscript  $+$  denotes the pseudo-inverse operation defined as  $A^+ = (A^T A)^{-1} A^T$ , as shown in Fig. 1. For a redundant manipulator (i.e.,  $n > m$ ), a null space of  $(J^T)^+$  can be defined as  $S_n = \{ \tau_n \mid (J^T)^+ \tau_n = 0_F \}$ . [14] The range space of  $J^T$  is defined as  $S_e = \{ \tau_e \mid \tau_e = J^T F_e \}$ , where  $F_e$  in  $S_F$  is the  $(m \times 1)$  external force

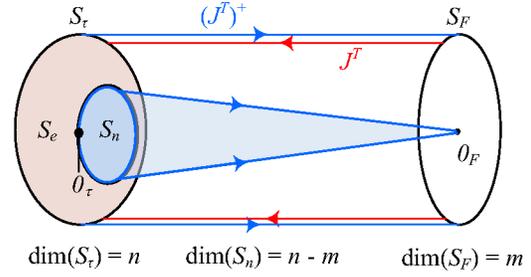


Figure 1. Mapping between joint torque space and end-effector force space.

vector applied to the end-effector, and  $\tau_e$  is the  $(n \times 1)$  joint torque vector generated by  $F_e$ .

In Fig. 1,  $S_e$  and  $S_n$  are the subspaces of  $S_\tau$  and they share the common origin  $\theta_\tau$ . Since the space  $S_\tau$  consists of  $S_e$ ,  $S_n$ , and their linear combination, a vector  $\tau$  in  $S_\tau$  can be described by

$$\tau = \tau_e + \tau_n \quad (5)$$

where  $\tau_e$  and  $\tau_n$  are the vectors in  $S_e$  and  $S_n$ , respectively.

As discussed above, the relationship between the end-effector force and the joint torque is given by

$$\tau_e = J^T F_e \quad (6)$$

For a redundant manipulator (i.e.,  $n > m$ ), (6) can be written as

$$F_e = (J^T)^+ \tau_e \quad (7)$$

and it can be rewritten as

$$F_e = (J^T)^+ \tau \quad (8)$$

by (5) and the relation of  $(J^T)^+ \tau_n = 0$ . Then, substituting (8) into (6) yields

$$\tau_e = J^T (J^T)^+ \tau \quad (9)$$

where  $J^T (J^T)^+$  is the projection matrix for the projection from  $S_\tau$  into  $S_e$ . From (5) and (9),  $\tau_n$  can be given by

$$\tau_n = \tau - \tau_e = (I - J^T (J^T)^+) \tau \quad (10)$$

where  $(I - J^T (J^T)^+)$  is the projection matrix for the projection from  $S_\tau$  into  $S_n$ .

The inner product of the two vectors  $\tau_e$  and  $\tau_n$  is

$$\begin{aligned} \tau_e^T \tau_n &= (J^T (J^T)^+ \tau)^T (I - J^T (J^T)^+) \tau \\ &= (((J^T)^+ \tau)^T J) (I - J^T (J^T)^+) \tau \\ &= ((J^T)^+ \tau)^T (J - J J^T (J J^T)^{-1} J) \tau = 0 \end{aligned} \quad (11)$$

Therefore, the two vector spaces  $S_e$  and  $S_n$  are orthogonal to each other. The vector  $\tau_e$  can be projected into  $S_n$  by (10) because  $S_e$  is a subspace of  $S_\tau$ . Since a vector projected into a

vector space that is orthogonal to its own space always results in the zero vector, we conclude

$$0 = (I - J^T (J^T)^+) \tau_e \quad (12)$$

This means that  $\tau_e$  does not affect the result of projecting a vector into  $\tau_n$ , and this fact leads to the derivation of the collision detection index in the next section.

### B. Collision detection index

Figure 2(a) shows the situation in which a human-robot collision occurs (with a force  $F_c$ ) while a manipulator is performing a contact task (with a force  $F_e$ ) using the end-effector. In the case of force interaction such as force control, collisions between a human and a robot occur more frequently in the parts other than the end-effector because the end-effector's position is purposefully controlled to be relatively static during the performance of a task unlike the other parts of the manipulator. Therefore, we assume that the collision force  $F_c$  acts on the body parts other than the end-effector.

Combining (2) with (6) yields

$$\tau_{ext} = J^T F_e + \tau_c \quad (13)$$

Therefore, collision detection based on only  $\tau_{ext}$  may lead to false detection because  $\tau_{ext}$  can have a nonzero value due to  $F_e$  even when  $\tau_c = 0$  (i.e., no body collision). Therefore, we need to develop a collision detection index which is affected only by  $F_c$  (i.e., body collision) and is independent of  $F_e$ . From (2), (10) and (12), the collision detection index is defined as

$$\begin{aligned} N &= (I - J^T (J^T)^+) \tau_{ext} \\ &= (I - J^T (J^T)^+) \tau_e + (I - J^T (J^T)^+) \tau_c \\ &= (I - J^T (J^T)^+) \tau_c \end{aligned} \quad (14)$$

The collision detection index  $N$  is a function of  $\tau_c$  (and thus  $F_c$ )

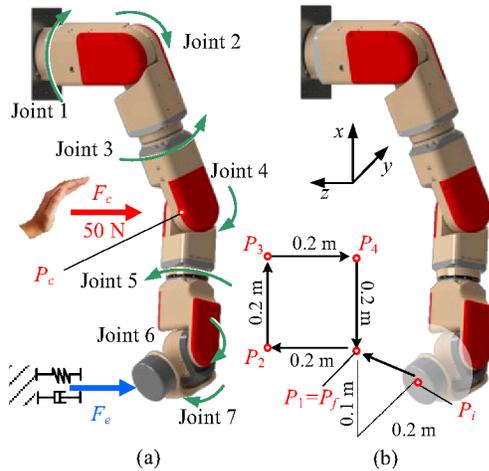


Figure 2. Simulation conditions: (a) applied forces, and (b) trajectory of end-effector.

and  $J$  (i.e., manipulator configuration), and is fundamentally unrelated to  $\tau_e$ . Thus, body collisions during force interaction at the end-effector can be detected by observing the value of each element of  $N$ .

## IV. VERIFICATION

The collision detection index  $N$  proposed in the prevision section is verified by MATLAB-based simulations. This simulation is conducted using a 7 DOF manipulator developed in our laboratory. It has a reachable distance of 0.8 m, and weighs about 14 kg with a payload of 7 kg.

In the simulation, collisions occur at the manipulator body while the manipulator is doing a contact task using its end-effector. Some error is added to  $\tau_{ext}$  in order to investigate the change in performance that may occur in an actual experiment. Since  $N$  is a function of  $J$  and  $\tau_c$  in (12), the collision detection performance is affected by the manipulator configuration, the magnitude of  $F_c$ , and the level of error in  $\tau_{ext}$ .

### A. Simulation conditions

Figure 2 shows the simulation conditions and external forces. An external continuous reaction force  $F_e$  is applied to a 7 DOF manipulator.  $P_c$  represents the position where  $F_c$  is applied, and we assume that a collision occurs at link 3 of the manipulator. Fig. 2(b) illustrates the trajectory of the end-effector during a task. After moving -0.2 m in the  $y$  direction and 0.1 m in the  $x$  direction from the initial end-effector position  $P_i$ , the end-effector draws a square of size 0.2 m through  $P_1$ ,  $P_2$ ,  $P_3$ , and  $P_4$ , and reaches the final position  $P_f$ . The execution time of the simulation is 12 sec, and it takes 2 sec to move from the current position to the next position (e.g.,  $P_1$  to  $P_2$ ). The collision occurs between 6 and 8 sec. Angular positions of each joint along the end-effector trajectory are listed in Table I.

The components  $F_{ex}$ ,  $F_{ey}$ ,  $F_{ez}$ ,  $M_{ex}$ ,  $M_{ey}$ , and  $M_{ez}$  of the interaction force  $F_e$  have the shape of a trigonometric function as shown in Fig. 3(a). The magnitudes of  $F_{ex}$ ,  $F_{ey}$ , and  $F_{ez}$  are 50 N, and their periods are 5 s, 3 s, and 2 s, respectively. The magnitudes of  $M_{ex}$ ,  $M_{ey}$ , and  $M_{ez}$  are 10 Nm, and their periods are 5 s, 3 s, and 2 s, respectively. On the other hand, the components  $F_{cx}$ ,  $F_{cy}$ , and  $F_{cz}$  of the collision force  $F_c$  are 50 N at the intervals of 6 to 6.2 sec, 6.9 to 7.1 sec, and 7.8 to 8.0 sec, respectively, which represents the physical conditions created when three collisions occur, as shown in Fig. 3(b). Note that 50 N is the collision force threshold above which humans

TABLE I. ANGULAR POSITION OF JOINTS

| Joint   | Angular position of joints |         |         |         |         |         |
|---------|----------------------------|---------|---------|---------|---------|---------|
|         | $P_i$                      | $P_1$   | $P_2$   | $P_3$   | $P_4$   | $P_f$   |
| Joint 1 | 5 °                        | 11.3 °  | 11.5 °  | 1.2 °   | -0.1 °  | 10.9 °  |
| Joint 2 | 5 °                        | 11.7 °  | 1.7 °   | 6.9 °   | 17.7 °  | 11.1 °  |
| Joint 3 | -5 °                       | -18.9 ° | -20.9 ° | -20.1 ° | -25.5 ° | -26.9 ° |
| Joint 4 | 5 °                        | 38.3 °  | 38.2 °  | 72.9 °  | 73 °    | 38.6 °  |
| Joint 5 | -5 °                       | 12.1 °  | -5.1 °  | -1.9 °  | 11.4 °  | 11.1 °  |
| Joint 6 | 5 °                        | -32.8 ° | -35.9 ° | -58.9 ° | -55.3 ° | -33.1 ° |
| Joint 7 | -5 °                       | -6.3 °  | 9.3 °   | 6.27 °  | -3.6 °  | -7.4 °  |

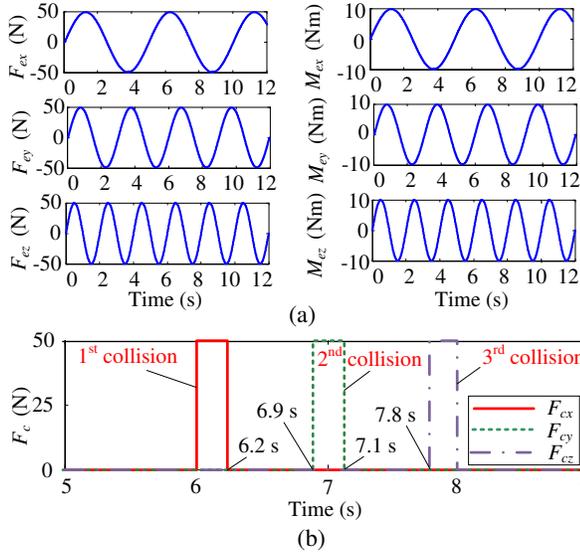


Figure 3. Applied forces versus time: (a) interaction force  $F_e$  and (b) collision force  $F_c$

begin to feel pain [15]. Since the dominant factor that determines human injuries caused by human-robot collision is the collision force, the moment-related components of  $F_c$  are set to 0.

When  $F_e$  and  $F_c$  shown in Fig. 3 are applied to the manipulator,  $\tau_e$  is calculated from (5), and  $\tau_c$  can be calculated from

$$\tau_c = J_{03}^T F_c \quad (15)$$

where  $J_{03}$  is the Jacobian matrix defined by the relationship between the velocity of  $P_c$  and the angular velocity of the first three joints (i.e., joints 1, 2, and 3 in Fig. 2).

### B. Simulation results

The computed components of  $\tau_{ext}$  that correspond to joint 1, joint 2, and joint 3, are shown in Fig. 4, but the other components (i.e., joint 4 to 7) of  $\tau_{ext}$  are omitted since a collision acting on  $P_c$  mainly affects the first three joints [11]. As shown in Fig. 4,  $\tau_{ext}$  shows non-zero values even before collisions because the interaction force  $F_e$  makes a contribution to  $\tau_{ext}$ . Therefore, it is difficult to determine if a collision has happened based on whether  $\tau_{ext}$  exceeds the threshold or not.

As verified in (12), the null-space projection of  $\tau_{ext}$  (i.e.,  $N$ ) does not depend on  $\tau_e$ . Figure 5 shows the elements of the index  $N$  with time. Three elements of  $N$  are shown in Fig. 5; these are chosen arbitrarily since the rows of  $N$  behave similarly. The solid line indicates the elements of  $N$  when collisions occur, and the dotted line indicates elements when no collision occurs. If a collision has not occurred,  $\tau_c$  is zero, and thus every element of  $N$  is zero as well. On the other hand, over the intervals of collisions, the elements of  $N$  have a non-zero value. Note that a contact task is performed during the simulation, which results in a non-zero value of  $F_e$ , but the elements of  $N$  are kept zero in the range over which no

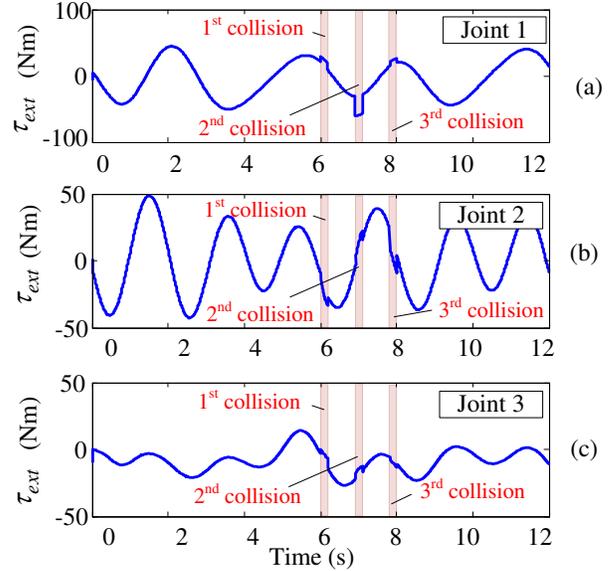


Figure 4. Simulation results for external torque: (a) 1<sup>st</sup> joint, (b) 2<sup>nd</sup> joint, and (c) 3<sup>rd</sup> joint.

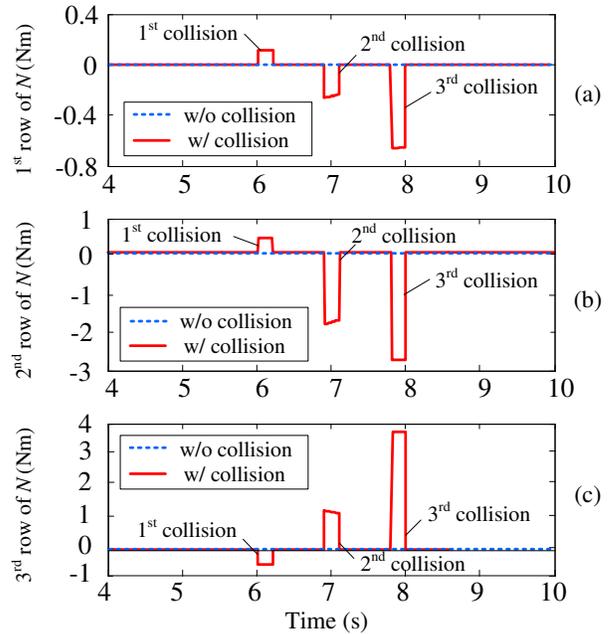


Figure 5. Developed collision detection index  $N$ : (a) 1<sup>st</sup> row of  $N$ , (b) 2<sup>nd</sup> row of  $N$ , and (c) 3<sup>rd</sup> row of  $N$ .

collision occurs. Therefore, we can detect collision by observing the magnitudes of the elements of  $N$  even when a contact task is conducted.

In an actual system,  $\tau_{ext}$  is computed based on the equation of motion of a manipulator and the torques measured by the joint torque sensors. Therefore,  $\tau_{ext}$  may exhibit some error due to the uncertainty of the manipulator and torque measurement. These errors in  $\tau_{ext}$  then affect the value of  $N$ , and thus the collision detection performance may be degraded. Figure 6 shows each element of  $N$  with respect to time, when the error in each element of  $\tau_{ext}$  is set to 5% of the measurable range of

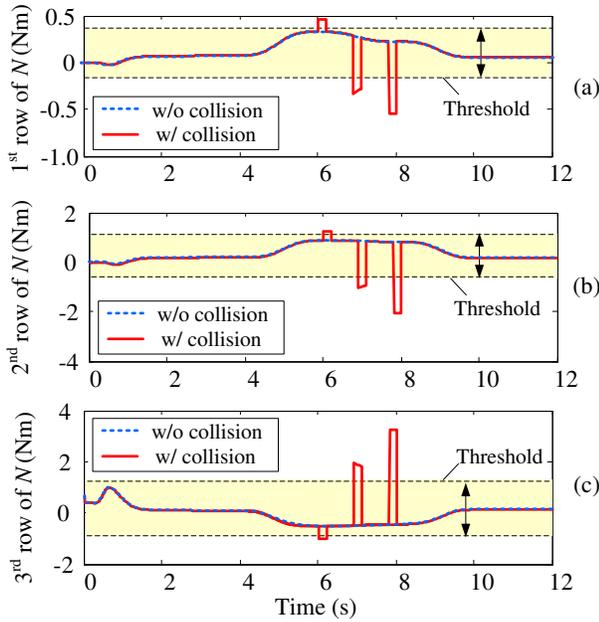


Figure 6. Collision detection index  $N$  with 5% error of  $\tau_{ext}$ : (a) 1<sup>st</sup> row of  $N$ , (b) 2<sup>nd</sup> row of  $N$ , and (c) 3<sup>rd</sup> row of  $N$ .

the joint torque sensor. The simulations are carried out under the same conditions described in Fig. 2 and Fig. 3. Unlike the simulation results described in Fig. 5, each element of  $N$  in Fig. 6 has a non-zero value even when no collision occurs. That is, with  $\tau_{ext}$  is contaminated with error, collisions cannot be detected just by observing whether an element of  $N$  is zero or not. Instead, collisions ought to be detected by observing whether the elements of  $N$  exceed a predefined threshold or not. Fig. 6 shows that collisions can be effectively detected for a 5% error level in  $\tau_{ext}$  by properly selecting a threshold level.

Fig. 7 shows each element of  $N$  with respect to time when the error level in  $\tau_{ext}$  is set to 10% of the measurable range of the joint torque sensor. Similarly to Fig. 6, each element of  $N$  has a non-zero value even in the absence of a collision. The threshold of  $N$  for collision detection was increased according to an increase in the error level of  $\tau_{ext}$ , but the second collision could not be detected because the values of all the elements of  $N$  were below the threshold. However, such situations with a very large error will not arise very often during regular operations of a manipulator. Therefore, by employing the proposed index  $N$  for a redundant manipulator, it is possible to detect collisions of a manipulator conducting contact tasks.

Another issue is how to detect a collision at the end-effector during its free motion without a contact with the environment because the proposed index is not affected by the end-effector force. To cope with this problem, we recommend the use of the proposed collision detection index along with the collision detection method described in [10]. In general, the geometry of the environment at the start of force interaction is known. Therefore, collisions not only during the performance of a task but also during the free motion can be detected by switching the collision detection method from the one described in [10] to the proposed method when a contact task is expected.

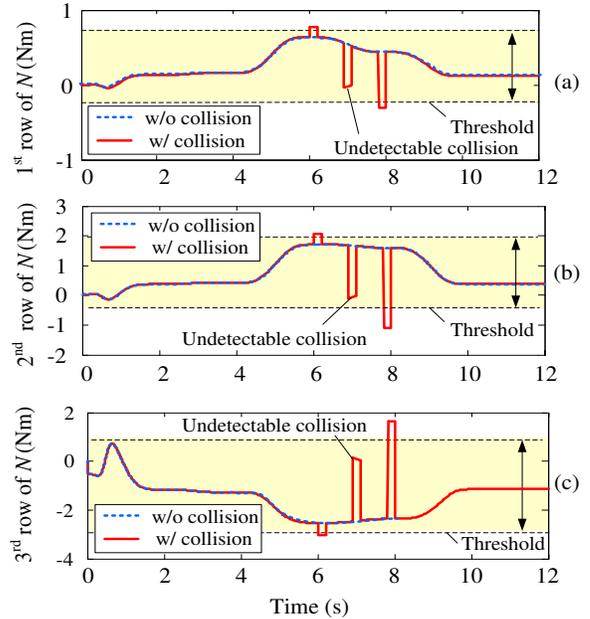


Figure 7. Collision detection index  $N$  with 10% error of  $\tau_{ext}$ : (a) 1<sup>st</sup> row of  $N$ , (b) 2<sup>nd</sup> row of  $N$ , and (c) 3<sup>rd</sup> row of  $N$ .

## V. CONCLUSION

In this study, a collision detection index was proposed to detect a body collision of a redundant manipulator equipped with joint torque sensors when the end-effector is interacting with the environment. The proposed index was verified based on simulations in which various levels of error in the external torque are considered. The following conclusions are drawn from these results:

1) The proposed collision detection index is fundamentally unrelated to the joint torques generated by the end-effector force interacting with the environment. Thus, a body collision can be detected regardless of the frequency and magnitude of the collision force.

2) With the proposed collision detection index, a collision can be detected using only the joint torque sensors without an extra force/torque sensor. Therefore, the proposed collision detection index is very economical.

In a future study, the proposed collision index will be verified experimentally using the 7 DOF manipulator.

## REFERENCES

- [1] V. Lumelsky and E. Cheung, Real-time collision avoidance in teleoperated whole-sensitive robot arm manipulator, *IEEE Trans. on Systems, Man, and Cybernetics*, 1993, vol. 23, no.1, pp. 194-203.
- [2] N. Lauzier, and C. Gosselin, "Series Clutch Actuators for Safe Physical Human-Robot Interaction," in *IEEE Int. Conf. on Robotics and Automation*, Shanghai, China, 2011, pp. 5401-5406.
- [3] J. J. Park, H. S. Kim, and J. B. Song, Safe Robot Arm with Safe Joint Mechanism using Nonlinear Spring System for Collision Safety, in *IEEE Int. Conf. on Robotics and Automation*, Kobe, Japan, 2009, pp. 3371-3376.
- [4] F. Flacco, T. Kroger, A. D. Luca, and O. Khatib, "A depth space approach to human-robot collision avoidance," in *IEEE Int. Conf. on Robotics and Automation*, Saint Paul, USA, 2012, pp. 338-345.

- [5] S. Haddadin, S. Haddadin, A. Khoury, T. Rokahr, S. Parusel, R. Burgkart, A. Bicchi, and A. Albu-Schäffer, "A truly safely moving robot has to know what injury it may cause," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Vilamoura, Algarve, Portugal, 2012, pp. 5406–5413.
- [6] S. D. Lee, B. S. Kim, and J. B. Song, "Human–robot collision model with effective mass and manipulability for design of a spatial manipulator," in *Advanced Robotics*, vol. 27, no.3, 2013, pp.189-198.
- [7] S. Takakura, T. Murakami, and K. Ohnishi, "An Approach to Collision Detection and Recovery Motion in Industrial Robot," in *15th Annual Conf. of IEEE Industrial Electronics Society*, 1998, pp. 421-426.
- [8] Y. Yamada, Y. Hirasawa, S. Huang, Y. Umetani, and K. Suita, "Human-Robot Contact in th Safeguarding Space," *IEEE/ASME Trans. on Mechatronics*, vol. 2, no. 4, 1997, pp. 230-236.
- [9] W. E. Dixon, I. D. Walker, D. W. Dawson, and J. P. Hartranf, "Fault Detection for Robot Manipulators with Parametric Uncertainty: A Prediction-error-based Approach," in *IEEE Trans. On Robotics and Automation*, vol. 16, no. 6, 2000, pp. 689-699.
- [10] A. D. Luca and R. Mattone, "Actuator Fault Detection and Isolation using Generalized Momenta," in *IEEE Int. Conf. on Robotics and Automation*, Tahipei, Taiwan, 2003, pp. 634-639.
- [11] A. De Luca and R. Mattone, "Sensorless robot collision detection and hybrid force/motion control," in *IEEE Int. Conf. on Robotics and Automation*, 2005, pp. 1011–1016.
- [12] C. N. Cho, J. H. Kim, Y. L. Kim, J. B. Song, and J. H. Kyung, "Collision Detection Algorithm to Distinguish Between Intended Contact and Unexpected Collision," in *Advanced Robotics*, vol. 26, no. 16, 2012, pp.1825-1840.
- [13] J. H. Kim, Y. L. Kim, and J. B. Song, "Collision Detection Algorithm to Distinguish between Intended Contact and Unexpected Collision," in *int. Conf. on Ubiquitous Robots and Ambient Intelligence*, 2010, pp. 221-224.
- [14] O. Khatib, "Inertial Properties in Robotic Manipulation: An Object-level Framework," in *Int. Journal of Robotics Research*, vol. 14, no. 1, 1995, pp. 19-36.
- [15] Y. Yamada, Y. Hirasawa, S.Y. Huang and Y. Umetani, "Fail-safe human/robot contact in the safety space," in *IEEE Int. Workshop on Robot and Human Communication*, 1996, pp. 59-64.