

## 人間協調型ロボットにおける RoBE を用いた実時間自己衝突回避\*

瀬戸 文美\*<sup>1</sup>, 小菅 一弘\*<sup>1</sup>, 平田 泰久\*<sup>1,2</sup>Real-Time Self-Collision Avoidance Using RoBE for  
Human-Friendly RobotFumi SETO\*<sup>3</sup>, Kazuhiro KOSUGE and Yasuhisa HIRATA<sup>\*3</sup> Department of Bioengineering and Robotics, Tohoku University,  
6-6-01 Aoba, Aramaki-za, Aoba-ku, Sendai shi, Miyagi, 980-8579 Japan

We have proposed a real-time self-collision avoidance control method for the robot which is used for human-robot cooperation. In this method, we represent the body of the robot by using elastic elements referred to as "RoBE (Representation of Body by Elastic elements)". In this paper, especially, we consider task constraints and environmental constraints during the self-collision avoidance motion, and propose two priority functions for robots to realize the several kinds of tasks in an environment based on the force/moment applied by a human. By using this control algorithm, we could apply the proposed control algorithm to any robot systems used for human-robot cooperation. The proposed motion control algorithm is implemented in a human friendly robot, referred to as "MR Helper", and experiments are done for illustrating the validity of the proposed self-collision avoidance motion.

**Key Words:** Robot, Manipulator, Moving Robot, Mobile Manipulator, Self collision Avoidance, RoBE

## 1. はじめに

近年, 人間との協調作業を前提としたヒューマノイドやモバイルマニピュレータが多数開発されているが, これらのロボットシステムにはそれらが有する多自由度故に, ロボットが自身と衝突する自己衝突が起きる危険性が存在する. 自己衝突の発生によりロボットが破損したり, 転倒したロボットによって周囲に危害が及ぶ恐れがあるため, ロボットが人間と作業空間を共にするには自己衝突問題の解決が必要不可欠である.

人間と協調作業を行うロボットにおける自己衝突問題に対して, 著者らはこれまでにロボットの弾性要素表現 "RoBE (Representation of Body by Elastic elements)" 及び RoBE を用いた自己衝突回避制御手法を提案し, ロボットが単体で運動している場合における回避実験を実現し, その概念の有効性を確認した<sup>(1)</sup>.

ロボットの自己衝突問題に関しては, 本研究以外にも金広らや J. Kuffner らがヒューマノイドロボットの歩行計画問題として取り扱っている<sup>(2)(3)</sup>. また, 自己衝突問題に応用可能な研究としてロボットの障害物回避の研究がある. O. Khatib らは "Elastic strip" の概念を提案し, モバイルマニピュレータの実時間での動作

計画問題に応用した<sup>(4)</sup>. これらの手法は予めロボットの動作計画を行うことができる場合には有効であるが, 協調作業においてロボットの動作は人間から加えられる操作力に基づき実時間で生成されるため, 人間が加える力を予測して動作計画を行うことは困難であり適用は難しい. 実時間での衝突回避については, M. P. Hennessey らは高速走行車において "Virtual bumper" を用いた手法を提案している<sup>(5)</sup>. この手法は2次元平面上の簡単な形状の物体における実時間衝突回避が可能であるが, 人間協調型ロボットのような3次元空間における複雑な形状の物体への適用は困難である.

本研究で提案している RoBE を用いた自己衝突回避制御手法は, 人間と協調作業を行うロボット全般に適用が可能であり, かつ人間との協調作業中においても自己衝突を回避することが可能である. しかしながら, これまで用いていた自己衝突回避動作はロボットの形状や作業状態に依存しており, 様々な協調作業を行う場合や異なるロボットを用いる場合などに同じ回避動作をそのまま適用することは困難である. また人間との協調作業中においては, 自己衝突回避中であってもできるだけ協調作業を継続することが望ましい.

本稿では, これまで提案してきた RoBE を用いた自己衝突回避において, 人間との協調作業に用いられるロボット全般に適用可能かつ協調作業中であっても自

\* 原稿受付 2005年8月22日.

<sup>\*1</sup> 正員, 東北大学大学院工学研究科(☎980-8579 仙台市青葉区荒巻字青葉6-6-01).<sup>\*2</sup> 正員, 科学技術振興事業機構さきがけ研究21(☎332-0012 川口市本町4-1-8).

E-mail: setofumi@irs.mech.tohoku.ac.jp

己衝突回避が可能な一般性を有する自己衝突回避動作を提案する。提案する回避動作を人間協調型ロボット“Mr Helper”に適用し、実験を行い有効性を確認する。

## 2. RoBE を用いた実時間自己衝突回避制御

本節では RoBE を用いた実時間自己衝突回避制御手法と、実際のロボットへ適用する手法の概要を述べる。

### 2.1 RoBE (Representation of Body with Elastic elements)

RoBE とは図 1 に示すような中心部分に非弾性領域、表面部分に弾性領域を有する弾性要素 (Elastic element) を用いてロボット形状を仮想的に表現するものである。実際のロボットの運動によりロボットのリンク同士が接近して自己衝突の危険が高まった際には弾性要素間の接触が発生し、その 2 要素間には仮想的な反力が生成される。生成された反力に基づいて実際のロボットの運動を制御することにより、反力の方向、すなわち弾性要素間の接触を回避する方向へロボットが運動し、自己衝突の回避を実現する。

RoBE ではどんな形状の弾性要素も用いることが可能であるが、本研究では実時間性を確保するために計算負荷の軽減を考慮し、球と円柱を用いる。弾性要素間の接触が検出された場合、その 2 要素間には要素間の距離に比例した大きさの反力が仮想的に生成され、その向きは 2 要素の中心を結んだ直線上である。また、接触状態が進んで非弾性領域同士が接触した場合は作用反作用の法則に従い、一方の要素に加えられる力と同じ大きさの反力がもう一方の要素に生成される。そのため、ロボット形状を表現する際にロボットが各弾性要素の有する非弾性領域に含まれるよう弾性要素を配置することにより、いかなる場合においても自己衝突を防ぐことができる。

**2.2 RoBE によるロボット形状の表現** RoBE を用いてロボット形状を表現する際、簡単のためロボット形状を Moving Segment (以下 MS と表す) と Joint Segment (以下 JS と表す) に分割する。ロボットの各関節部分を JS と定義するが、関節であってもその次の関節の姿勢のみを変化させるものは JS と見なさない。また、二つの JS に挟まれたリンク等の部分を MS と定義する。例えば、図 2(a) に示されるマニピュレータは、図 2(b) のように MS 及び JS に分割される。

ロボット形状の表現に用いられる弾性要素は全て、いずれかの MS または JS に属する。また全ての MS 及び JS は、少なくとも一つの弾性要素から構成される。ロボットを MS 及び JS に分割し、各セグメントを弾性要素を用いて表現すると、同一 MS 内の要素間、及びある MS とそれに隣接する JS の要素間には衝突

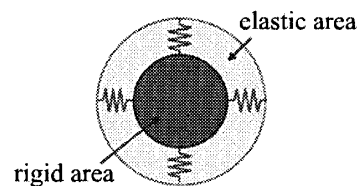


Fig. 1 Elastic Element

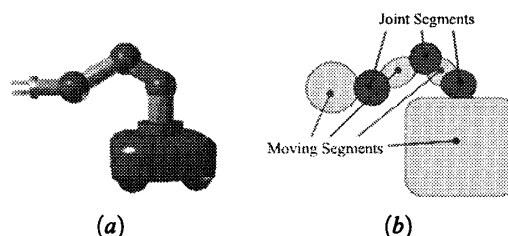


Fig. 2 Dividing Robot into MS/JS

の危険がない。そこで、それ以外の各要素の組に対して要素間の距離を計算し、衝突検出を行う。

## 3. 一般性を有する自己衝突回避動作

本節では人間協調型ロボットを対象とした、RoBE を用いた自己衝突回避制御における一般性を有する自己衝突回避動作の実現手法について述べる。

本回避動作ではロボット上に複数の制御点を配置し、自己衝突回避のための力や人間から加えられる操作力に基づいて各制御点の変位を制御する。得られた変位をもとにロボットの動作を生成し、人間との協調作業及び自己衝突回避を実現する。動作を生成する際、どの制御点の目標変位を優先的に実現させるかを決定する優先度と、自己衝突回避動作と人間からの操作力に基づく動作のどちらをロボットの実際の動作により反映させるかを決定する優先度の、二つの優先度を用いることにより自己衝突回避と協調作業の共存を実現し、作業拘束や環境からの拘束への対応を可能とする。

初めに、RoBE を適用されたロボット上に複数の制御点を配置する手法を述べ、次に優先度を用いて自己衝突回避動作を生成する手法について述べる。

**3.1 制御点の配置** 本研究では議論を簡単にするため、対象としてロボットの手先の作業空間の自由度が  $l$ 、有する自由度が  $m$  のマニピュレータを 1 本搭載したモバイルマニピュレータを考える。

前章で述べたようにモバイルマニピュレータを JS と MS に分割し、各セグメントを弾性要素で表現することにより、RoBE を適用する。その際、図 3(a) に示すように、ロボットのボディを表す MS を  $MS_{body}$ 、ボディとマニピュレータを接続する JS を  $JS_{body}$  とし、

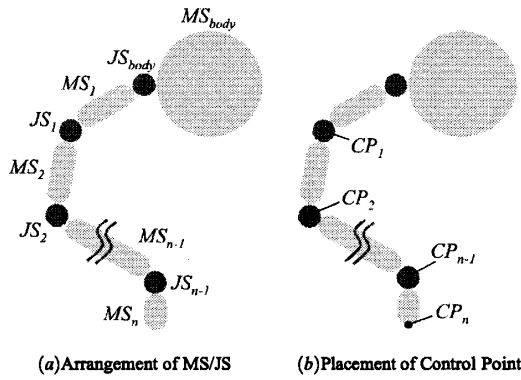


Fig. 3 Setting of Control Points

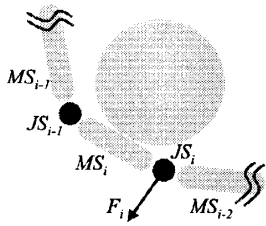


Fig. 4 Force applied to Control Point

そこからマニピュレータの手先に向かって順に  $MS_1$ ,  $JS_1$ ,  $MS_2$ ,  $JS_2$ ,  $\dots$ ,  $JS_{n-1}$ ,  $MS_n$  とする。

制御点はロボット上に任意に複数配置することが可能であるが、ロボットの有する自由度には限りがあるため 1 リンク上に複数の制御点を配置しても全ての目標変位を実現することは不可能である。そのため、本稿では図 3(b) に示すように  $JS_i$  の中点に制御点  $i(CP_i)$  を配置する。ただし、 $i=n$  のときは  $MS_n$  の先端の点、すなわちマニピュレータの手先と一致するように  $CP_n$  を配置する。

自己衝突回避制御によって  $MS_i$ ,  $JS_i$  に属する弾性要素に発生した反力は、図 4 に示すように、 $CP_i$  に加わる力  $F_{avoid_i} \in \mathbf{R}^l$  へと等価的に変換されるものとする。同様に、人間からロボットに加えられる操作力もその部位に応じて  $CP_i$  に加わる力  $F_{human_i} \in \mathbf{R}^l$  へと等価的に変換されるものとする。

**3.2 各制御点の目標変位の実現優先度** 自己衝突回避制御により  $CP_i$  に発生した力  $F_{avoid_i}$  に対して、 $CP_i$  の自己衝突回避のための位置  $X_{avoid_i} \in \mathbf{R}^l$  を以下の特性を持つように動的に制御する。

$$F_{avoid_i} = M_i \Delta \ddot{X}_{avoid_i} + D_i \Delta \dot{X}_{avoid_i} \quad (1)$$

$$\Delta X_{avoid_i} = X_{avoid_i} - X_i^d \quad (2)$$

$M_i$ ,  $D_i \in \mathbf{R}^{l \times l}$  は  $CP_i$  の見かけの慣性行列、粘性係数行列、 $X_i^d \in \mathbf{R}^l$  は  $CP_i$  の位置の目標軌道である。求まった  $\Delta X_{avoid_i}$  より、自己衝突回避のため

各制御点に発生した位置の変位の集合  $\Delta X_{avoid} = [\Delta X_{avoid_1}, \Delta X_{avoid_2}, \dots, \Delta X_{avoid_n}]^T \in \mathbf{R}^L$  を求める。  $L$  は各制御点の自由度の総和であり、 $L = l \times n$  である。

同様に人間がロボットに加える操作力  $F_{human_i}$  に対して、 $CP_i$  の人間からの操作力に基づく位置  $X_{human_i} \in \mathbf{R}^l$  を以下の特性を持つように動的に制御する。

$$F_{human_i} = M_i \Delta \ddot{X}_{human_i} + D_i \Delta \dot{X}_{human_i} \quad (3)$$

$$\Delta X_{human_i} = X_{human_i} - X_i^d \quad (4)$$

求まった  $\Delta X_{human_i}$  より、人間からの操作力に基づいて各制御点に発生した位置の変位の集合  $\Delta X_{human} = [\Delta X_{human_1}, \Delta X_{human_2}, \dots, \Delta X_{human_n}]^T \in \mathbf{R}^L$  を求める。

以上までで得られた  $\Delta X_{avoid}$ ,  $\Delta X_{human}$  それぞれから、辻らの提案した仮想アームの逆運動学解法<sup>(6)</sup>を用いて、自己衝突回避のために発生した関節角変位  $\Delta \theta_{avoid} = [\Delta \theta_{avoid_1}, \Delta \theta_{avoid_2}, \dots, \Delta \theta_{avoid_m}]^T \in \mathbf{R}^m$  と人間からの操作力によって発生した関節角変位  $\Delta \theta_{human} = [\Delta \theta_{human_1}, \Delta \theta_{human_2}, \dots, \Delta \theta_{human_m}]^T \in \mathbf{R}^m$  を、以下の評価関数  $Q$  が最小となるように求める。

$$Q(\Delta \theta_*) = (\Delta X^d - J \Delta X_*)^T W_* (\Delta X^d - J \Delta X_*) \quad (5)$$

$*$  = avoid/human,  $\Delta X^d = [\Delta X_1^d, \Delta X_2^d, \dots, \Delta X_n^d] \in \mathbf{R}^L$  である。  $J = [J^1, J^2, \dots, J^n] \in \mathbf{R}^{L \times m}$  であり、 $J^i \in \mathbf{R}^{m \times m}$  はマニピュレータの  $CP_i$  までのヤコビ行列である。

対角行列である重み行列  $W_* \in \mathbf{R}^{L \times L}$  が提案する一つの優先度であり、各制御点の目標変位の実現優先度である。  $W_*$  の要素  $w_{*j}^i$  ( $w_{*j}^i < 0$ ,  $1 < i < n$ ,  $1 < j < l$ ) は  $\Delta X_{*j}$  の  $j$  番目の要素に対する重み係数であり、 $W_*$  全体に対する  $w_{*j}^i$  の比率を大きくすることにより、 $\Delta X_{*j}$  の  $j$  成分が実現されやすくなる。この優先度の値を各制御点における自己衝突の危険度や作業状態に応じて変更することにより、人間との協調作業における操作性の向上や協調作業中における自己衝突回避、作業拘束や環境から受ける拘束などを扱うことが可能となる。

本稿では操作性及び自己衝突の回避を第一に考え、 $CP_i$  の  $j$  方向に加わる力  $F_{*j}^i$  に基づき、 $w_{*j}^i$  を以下の式を用いて求める。

$$w_{*j}^i = \begin{cases} \frac{|F_{*j}^i|}{\sum_{i=1}^n \sum_{j=1}^l F_{*j}^i} & \text{if } F_{*j}^i \neq 0 \\ 0.001 & \text{if } F_{*j}^i = 0 \end{cases} \quad (6)$$

$F_{*j}^i = 0$  の際の  $w_{*j}^i$  の値は、 $F_{*j}^i \neq 0$  の際の  $w_{*j}^i$  の値と比較して十分小さくする必要があるため、 $w_{*j}^i = 0.001$  とした。ここで、 $F_{*j}^i$  が大きいほど  $CP_i$  の  $j$  方向における自己衝突の危険性が高い、または  $CP_i$  の  $j$  方向へ人

間が操作しようとしていると考えられる。そのため上式を用いることにより、 $F_*^i$  の値が大きくなるほどその方向への変位の実現優先度が上がり、自己衝突回避や人間との協調作業を効果的に行うことが可能となる。

決定された  $W_*$  を用いて、 $\Delta\theta_*$  は以下の式を用いて求められる。

$$\Delta\theta_* = (J^T W_* J)^{-1} J^T W_* \Delta X^d \quad (7)$$

**3.3 ロボットの実際の動作へ反映する優先度** 前節までで求められた  $\Delta\theta_{avoid}$ 、 $\Delta\theta_{human}$  から実際にロボットに与える関節角変位  $\Delta\theta = [\Delta\theta_1, \Delta\theta_2, \dots, \Delta\theta_m]^T \in \mathbf{R}^m$  を以下の式を用いて求める。

$$\Delta\theta_i = \alpha \cdot \Delta\theta_{avoid_i} + (1 - \alpha) \cdot \Delta\theta_{human_i} \quad (8)$$

ここで、 $\alpha$  ( $0 \leq \alpha \leq 1$ ) は今回提案する二つ目の優先度であり、自己衝突回避のための動作と人間からの操作力に基づく動作、どちらの動作をロボットの実際の動作にどれだけ反映させるかを決定するものである。本稿では、以下の式を用いて  $\alpha$  を決定する。

$$\alpha = \frac{\left| \sum_{i=1}^n \sum_{j=1}^l F_{avoid_j}^i \right|}{\left| \sum_{i=1}^n \sum_{j=1}^l F_{avoid_j}^i + \sum_{i=1}^n \sum_{j=1}^l F_{human_j}^i \right|} \quad (9)$$

上式を用いることにより、自己衝突の危険がないときには  $\alpha = 0$  となり、ロボットは人間からの操作力に基づいた動作を行う。自己衝突の危険が高まるにつれて  $\alpha$  の値も大きくなり、自己衝突回避動作が反映される度合いが大きくなる。それにより、状況に応じた協調作業や自己衝突回避が可能となり、操作性の向上や操作者の安全性が高い自己衝突回避が実現できる。

## 4. 実験

**4.1 シミュレーション** 提案する回避動作を用いて自己衝突を回避することが可能かどうか確認するため、7自由度のマニピュレータを有するモバイルマニピュレータに提案する手法を適用し、計算機シミュレーションを行った。この7自由度マニピュレータ上には図5に示すように3点の制御点を配置した。

シミュレーション結果の一例を図6に示す。図中の矢印はマニピュレータの手先に加えた力の方向、円柱はモバイルマニピュレータの移動ベースを表現する弾性要素を示す。図6より、提案した手法の適用によってモバイルマニピュレータにおいて人間との協調作業中における自己衝突回避を実現し、さらに回避中においても回避に用いられていない自由度を用いて人間の操作力に基づく動作を継続していることが分かる。

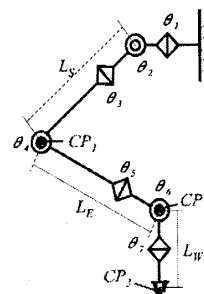
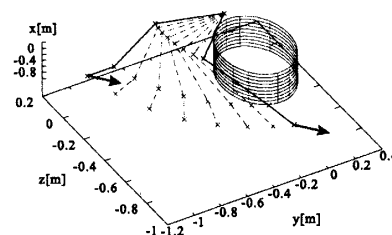
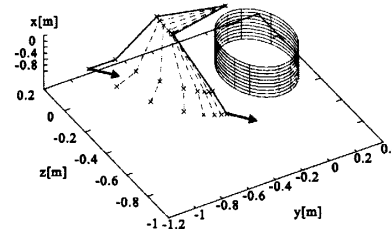


Fig. 5 ControlPoint for 7-DOF Manipulator



(a) Without Avoidance Control



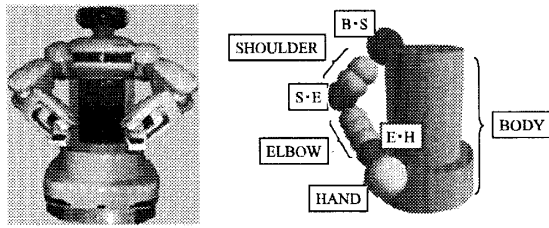
(b) With Avoidance Control

Fig. 6 Simulation Result

また、本シミュレーションでは Intel Pentium M(1.3GHz) を搭載した PC を用いたが、この PC において提案する自己衝突回避制御の 1 周期分の計算にかかる処理時間を求めたところ、約 0.06[ms] と求められた。この結果より、提案する自己衝突回避制御手法及び回避動作は、実時間性を必要とする人間協調型ロボットの運用に十分適用可能であると言える。

**4.2 実験** 本節では図7(a)に示される人間協調型ロボット“Mr Helper”<sup>(7)</sup>を用いて、人間との協調作業中における自己衝突回避実験を行った。Mr Helper は全方向移動ベースとボディ、及びボディに取り付けられた2台の7自由度マニピュレータ（三菱重工製 PA-10）から構成される。マニピュレータの手首部には六軸力センサ（Nitta 製 JR3 90M31）が取り付けられており、人間からの操作力の検出に用いられる。また、本実験においてすべての制御アルゴリズムは1台の PC(CPU: Intel Pentium IV(2.4GHz)) を用いて実現した（サンプリング周期: 512Hz）。

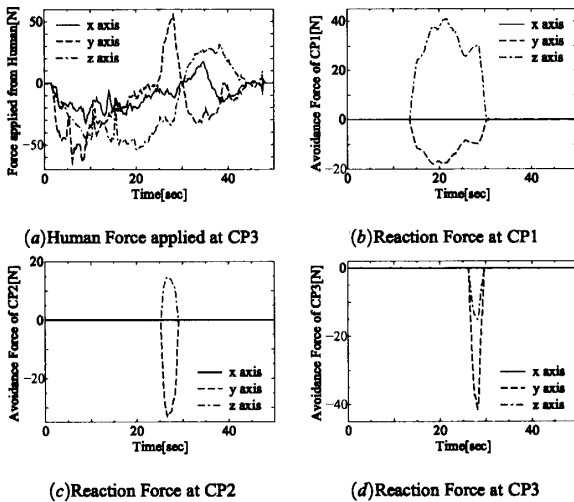
Mr Helper には図7(b)に示すように予め RoBE を適用し、Mr Helper に搭載されているマニピュレータ



(a) Mr Helper

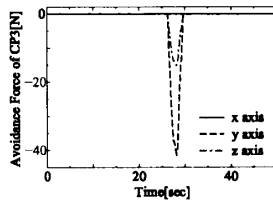
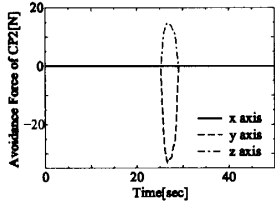
(b) Mr Helper Applying RoBE

Fig. 7 Mr Helper



(a) Human Force applied at CP3

(b) Reaction Force at CP1



(c) Reaction Force at CP2

(d) Reaction Force at CP3

Fig. 8 Force applied to Manipulator

には先のシミュレーションと同様に3点の制御点を配置した。本実験では簡単のため、Mr Helperのボディ及び右腕に対して自己衝突回避制御を適用した。人間から加えられる操作力は手首の力センサのみに加えられるものとし、 $CP_3$ に加わる力へと変換される。また、 $M_i$ ,  $D_i$ の値は操作者が操作しやすい値を実験的に求め、 $M_i = 5.0[\text{kg}]$ ,  $D_i = 10.0[\text{Ns/m}]$ とした。

本実験では操作者がMr Helperのマニピュレータに根本( $CP_1$ )から順に手先( $CP_3$ )までボディに巻き付くような操作力を加えた場合、自己衝突を回避しつつ、巻き付き動作を完了できるかどうか確認した。

実験結果を図8、図9及び図10に示す。図8(a)はマニピュレータの手先( $CP_3$ )に加わる人間の操作力の変化を、図8(b), (c), (d)はそれぞれ $CP_1$ ,  $CP_2$ ,  $CP_3$ に生成された自己衝突回避のための反力の変化を示す。図8より、自己衝突の危険のない範囲ではマニピュレータが人間の操作力に基づいて運動し、マニピュレータ上の各制御点における自己衝突の危険が高まるとその点において反力が生成されていることがわかる。その結果、各制御点が自己衝突を回避する方向へ運動し、人間が操作力を加え続けているにもかかわらず自己衝突回避が実現されている。

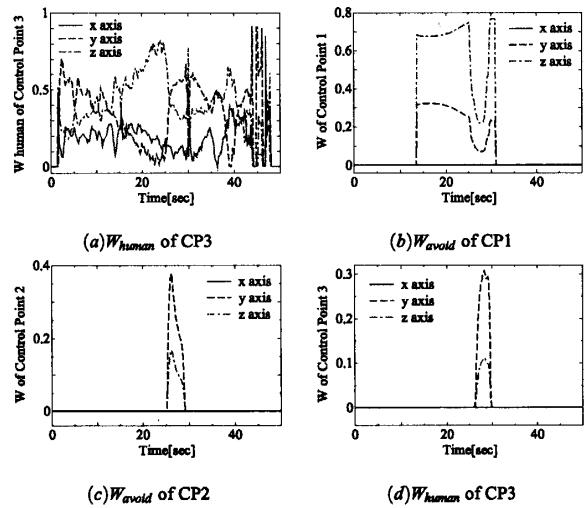
(a)  $W_{human}$  of CP3(b)  $W_{avoid}$  of CP1(c)  $W_{avoid}$  of CP2(d)  $W_{human}$  of CP3

Fig. 9 Value of Priority "W"

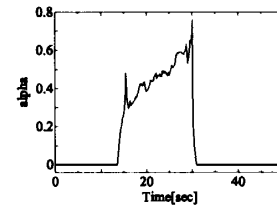


Fig. 10 Value of Priority "α"

また、図9(a)は $CP_3$ の人間の操作力に基づく目標変位の実現に対する優先度 $w_{human}^3$ の変化を、図9(b), (c), (d)はそれぞれ $CP_1$ ,  $CP_2$ ,  $CP_3$ の自己衝突回避のために発生した目標変位の実現に対する優先度 $w_{avoid}^1$ ,  $w_{avoid}^2$ ,  $w_{avoid}^3$ の変化をそれぞれ示す。図8(a)と図9(a)から、その方向における人間が加える操作力が大きいほどその方向の目標変位実現に対する優先度が高くなり、人間の加える操作力が大きい方向へロボットが容易に運動可能となっていることがわかる。さらに図8(b), (c), (d)及び図9(b), (c), (d)から、ある制御点における自己衝突の危険度が高まると、その点における目標変位実現に対する優先度が高くなり、自己衝突の危険度が高い制御点の変位が他の制御点の変位と比較して優先的に実現され、限られた自由度での効果的な自己衝突回避が実現される。

また、図10は自己衝突回避のための動作をロボットの実際の動作にどれだけ反映させるかを決定する優先度 $\alpha$ の変化を示す。図8、図10より自己衝突の危険が高まるにつれて $\alpha$ の値が大きくなり、その結果、Mr Helperの実際の動作における自己衝突回避動作の占める割合が大きくなり、状況に応じた自己衝突回避が実現されていることがわかる。

これらより、提案する回避動作は人間協調型ロボッ

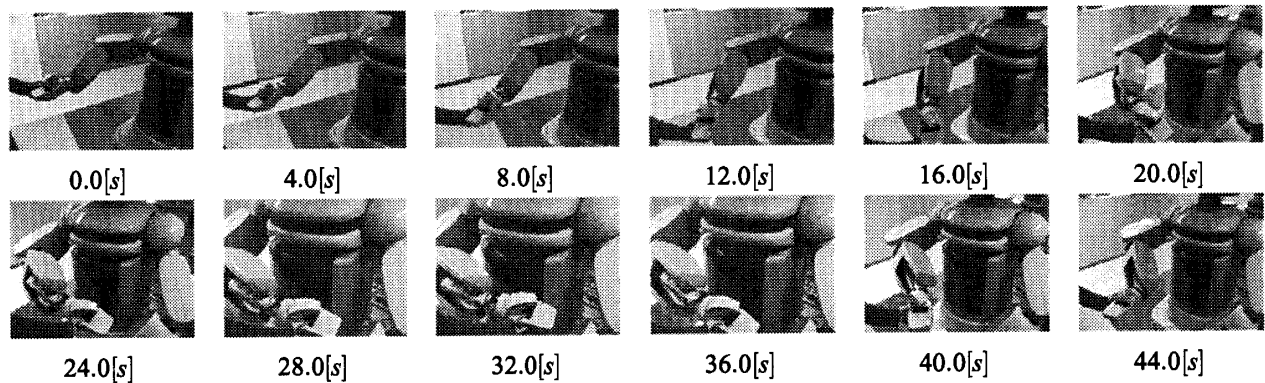


Fig. 11 Example of Experiment

トの自己衝突回避において十分有効であると言える。

また、実験の様子を図 11 に示す。0.0[s]～16.0[s]の間では、Mr Helper は人間が加える操作力に従って手先位置を変化させ、人間の意図に基づいた運動を行っている。しかし、20.0[s]にて Mr Helper の上腕、すなわち  $CP_1$  がボディに接近し、自己衝突を起こす危険が高まったために反力が  $CP_1$  に生成され、上腕がそれ以上自己衝突を起こす方向へは運動せず、上腕における自己衝突回避が実現されている。さらに人間が Mr Helper のボディにマニピュレータが巻き付くような操作力を加え続けた結果、20.0[s]～28.0[s]では、自己衝突に用いられていない前腕及び手先部分を用いて Mr Helper が人間からの操作力に基づいた動作を継続している。しかし 28.0[s]ではロボットの前腕、すなわち  $CP_2$  もボディに接近し、自己衝突を起こす危険が高まったために  $CP_1$  に加えて  $CP_2$  においても反力が発生し、前腕もそれ以上自己衝突を起こす方向へ運動せず、この点においても自己衝突回避が実現されている。さらに人間が操作力を加え続けた結果、Mr Helper の手先、すなわち  $CP_3$  が運動して巻き付き動作を継続し、32.0[s]ではマニピュレータ全体を用いたボディへの巻き付き動作が実現されている。ここで人間はさらに Mr Helper の手先に操作力を加えているが、32.0[s]で見られる以上にはマニピュレータは運動せず、全ての制御点において自己衝突が回避されている。巻き付き動作の完了後、人間が巻き付きを解除する方向、すなわち自己衝突を回避する方向へ力を加えた 36.0[s]～44.0[s]では、再び人間の操作力に基づく動作を実現している。

本実験では冗長自由度を有するマニピュレータを用いたが、冗長自由度を有さないマニピュレータにおいても提案する手法を適用することにより、限られた自由度を適切に用いて自己衝突を回避しつつ、人間からの操作力に基づく動作を実現することが可能である。

## 5. 結 言

本稿では RoBE を用いた自己衝突回避制御において、人間協調型ロボットシステム全般に適用可能かつ協調動作中においても自己衝突回避が可能な、一般性を有する自己衝突回避動作を“Mr Helper”に適用し、実験を行うことによってその有効性を確認した。

また、提案した手法は移動ベースの運動を含めた場合でも適用が可能である。その際、移動ベースが加わることによりロボットの運動の範囲が格段に広がるため、様々な回避動作が生成可能であると考えられる。そのため、作業拘束や環境から受ける拘束に応じてどのように優先度を変化させ、状況に応じた適切な回避動作を生成するかが今後の課題として挙げられる。

## 文 献

- (1) Seto, F. et al. “Real-time Self-collision Avoidance System for Robots using RoBE”. *International Journal of Humanoid Robotics*, Vol. 1, No. 3, pp. 533–550, 2004.
- (2) Kanehiro, F. et al. “Development of Real-Time Self Interference Checker of Humanoid Robot”. *Proceedings of The 19th Annual Conference of the Robotics Society of Japan*, pp. 3–4, 2001.
- (3) Kuffner, J. et al. “Self-Collision Detection and Prevention for Humanoid Robot”. *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2265–2270, 2002.
- (4) Khatib, O. et al. “Robots in Human Environments: Basic Autonomous Capabilities”. *The International Journal of Robotics Research*, Vol. 18, No. 7, pp. 684–696, 1999.
- (5) Hennessey, M. P. et al. “Sensor-based virtual bumpers for collision avoidance: configuration issues”. *SPIE Photonics 95 Conference, Collision Avoidance and Automated Traffic Management Sensors*, Vol. 2592, pp. 48–59, 1995.
- (6) Tsuji, T. et al. “Instantaneous Inverse Kinematics Solution for Redundant Manipulators Based on Virtual Arms and Its Application to Winding Control”. *JSME International Journal Series C*, Vol. 38, No. 1, pp. 87–93, 1995.
- (7) Kosuge, K. et al. “Mobile Robot Helper”. *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 583–588, 2000.