

## 보완 골격 알고리즘을 이용한 구난로봇의 자체 충돌감지/회피

이원석<sup>\*1)</sup> · 홍성일<sup>1)</sup> · 박규현<sup>1)</sup> · 강윤식<sup>1)</sup><sup>1)</sup> 국방과학연구소 제5기술연구본부 국방무인기술센터Self-Collision Detection/Avoidance for a Rescue Robot  
by Modified Skeleton AlgorithmWonsuk Lee<sup>\*1)</sup> · Seongil Hong<sup>1)</sup> · Gyuhyun Park<sup>1)</sup> · Younsik Kang<sup>1)</sup><sup>1)</sup> Defense Unmanned Technology Center, The 5th Research and Development Institute, Agency for Defense Development, Korea

(Received 18 November 2014 / Revised 13 May 2015 / Accepted 3 July 2015)

## ABSTRACT

This paper handles self-collision avoidance for a rescue robot with redundant manipulators. In order to detect all available self-collisions in advance, minimum distances between arbitrary robot parts should be monitored in real-time. For the minimum distance estimation, we suggest a modified method from a previous skeleton algorithm which has less computation burden and realize collision avoidance based on a potential function using the proposed algorithm. The resultant command by collision avoidance should not disturb a given primary task, so null-space of joint solution from a CLIK is utilized for collision avoidance by a gradient projection method.

Key Words : Rescue Robot(구난로봇), Self-Collision Detection/Avoidance(자체충돌 감지/회피),  
Modified Skeleton Algorithm(보완 골격 알고리즘)

## 기 호 설 명

 $P_i$  :  $i$  링크의 충돌 예상 지점 $S_i$  :  $i$  링크의 시작점 $E_i$  :  $i$  링크의 끝점 $w_i$  :  $i$  링크 벡터 위 임의의 점 계산을 위한 가중치 $u_i$  :  $i$  링크의 단위벡터 $x$  : 작업공간 임무상태변수 $q$  : 조인트 상태변수 $J$  : 자코비안 행렬  $J \in R^{(m \times n)}$  $J^*$  : 자코비안 의사 역행렬 $W$  : 조인트 가중치 $K$  : 역기구학 계인 행렬 $h$  : 임의의 포텐셜 함수 $D_{ij}$  :  $i, j$  링크사이의 최소거리 $V_{rep, i, j}$  : 충돌회피 포텐셜 함수 $F_{rep}$  : 충돌회피 반력 $F_{max}$  : 반력 최대값 $p_c$  : 충돌 가능지점

\* Corresponding author, E-mail: wsblues82@gmail.com

Copyright © The Korea Institute of Military Science and Technology

### 1. 서론

최근 로봇 발전에 힘입어, 재난 현장 등 사람의 접근이 어려운 환경에 로봇을 활용하려는 연구가 활발한 가운데, 본 논문에서는 이러한 요구에 적합하도록 Fig. 1과 같은 반 휴머노이드(상체 : 양팔 매니플레이터, 하체 : 트랙형태의 가변형상플랫폼)형태의 구난로봇을 다루고자 한다. 제안된 구난로봇은 Fig. 2에 묘사되어 있는 것처럼 그리퍼를 제외하고 총 20자유도를 가지고 있고 이를 활용하여 전장의 사람을 구난하는 연구를 진행하고 있다<sup>[1,2]</sup>. 이와 같이 여유자유도가 많은 로봇에게 일반적으로 임의의 임무를 부여할 경우, 그리퍼를 기준으로 궤적을 생성하고 이를 역기구학 문제로 조인트 궤적을 산출하기 때문에, 셀프모션 등으로 인해 양팔과 다른 부위와의 충돌이 발생할 수 있다. 이를 방지하기 위해서는, 로봇의 모든 부위의 상태정보를 기반으로 충돌을 미리 예상하고 이에 대한 회피 명령을 생성해야 한다.

충돌 감지는 로봇 연구에서 주요 이슈이기 때문에 많은 연구들이 진행되어 왔다. 로봇 매니플레이터의 충돌을 미리 예상하기 위해서는 로봇 부위간의 거리를 계산하고 실시간으로 충돌가능성에 대해 모니터링 해야 한다. 이러한 연구로서, Gilbert는 로봇의 각 부위를 실제 형상에 유사한 다면체로 고려하고 각 다면체 표면 간의 최소거리를 계산하는 방법을 제시하였다<sup>[3]</sup>. 이 방법은 각 부위의 형상을 반영한 만큼 거리계산에 대한 정밀도가 좋으나, 계산부담이 크다는 단점이 있다. Santis는 계산부담을 줄이기 위해, 각 부위를 골격형태의 링크로 고려하여 이를 벡터로 표현하고, 임의의 링크 간의 충돌 가능한 지점을 계산함으로써, 최소거리를 산출하는 골격 알고리즘(SA : Skeleton Algorithm)을 제안하였다<sup>[4]</sup>. 각 부위의 부피는 최소거리에 대한 기준치를 다르게 설정함으로써 고려될 수 있도록 하였다. 이 방법은 다면체 기반 알고리즘보다 충돌거리에 대한 정밀도는 떨어지지만, 충돌감지에 대해 보수적으로 접근할 경우 적은 계산부담을 갖는 유용한 알고리즘이라 할 수 있다. 따라서, 본 논문에서는 구난로봇의 자체충돌감지를 위해 SA를 적용하였다. 하지만 기존에 제안된 SA는 경우에 따라 최소거리를 산출하지 못하는 오류가 발견되어 이를 보완한 MSA(Modified skeleton algorithm)를 제안하고 이로부터 포텐셜 함수 기반의 충돌회피를 구현하였다.

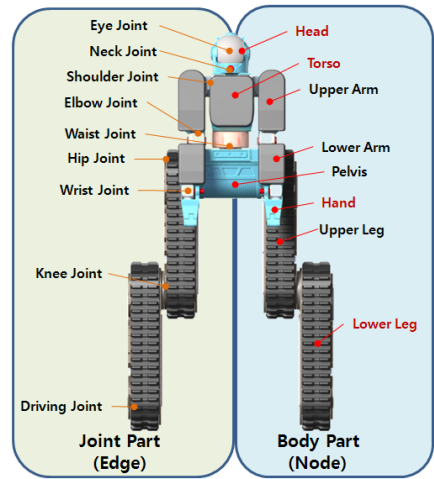


Fig. 1. Configuration of rescue robot

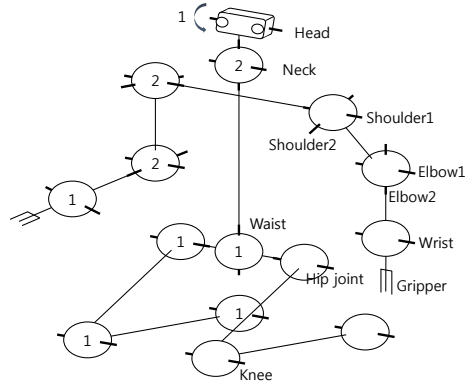


Fig. 2. DOF description of rescue robot

### 2. 자체충돌감지 알고리즘

#### 2.1 Skeleton Algorithm(SA)

SA는 로봇의 기구학을 기반으로 각 파트를 링크의 뼈대로 간주하여 Fig. 3와 같이 임의의 두 링크에 대한 충돌 가능 지점을 계산한다. 링크의 시작점( $S$ )과 끝점( $E$ )을 알고 있는 경우, 링크 위의 임의의 모든 점은 식 (1)처럼 링크의 시작점과 단위벡터( $u_i$ )의 조합으로 표현가능하다. 여기서  $i, j$ 는 임의의 두 벡터를 위한 인덱스이며,  $w_i$ 는 링크 위의 길이 가중치를 나타내는 스칼라이다. SA는 임의의 두 링크의 충돌을 벡터 기반으로 계산하기 때문에 링크 밖의 점이 충돌 가능 지점으로 계산될 수 있다. 하지만, 임의의 점은

항상 링크 위에 존재해야하기 때문에 가중치에 대한 조건을 식 (3)와 같이 제한하도록 한다<sup>[4]</sup>.

$$P_i = S_i + w_i u_i \quad (1)$$

$$u_i = \frac{E_i - S_i}{\|E_i - S_i\|} \quad (2)$$

$$\frac{w_i}{\|E_i - S_i\|} \in \{0,1\} \quad (3)$$

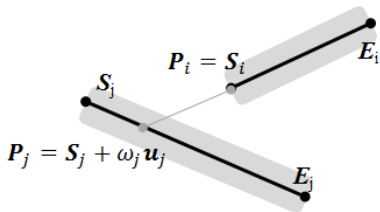


Fig. 3. Concept of possible collision points between arbitrary two links

본 논문에서 다루는 구난로봇의 링크는 총 13개로 분류하여 적용하도록 한다( $i \in \{1, 2, \dots, 12, 13\}$ ). 3차원 공간상에서 임의의 두 링크의 가장 짧은 거리를 갖는 벡터는 두 링크와 직교한다는 사실( $(P_i - P_j) \cdot u_i = 0$ ,  $(P_i - P_j) \cdot u_j = 0$ )에 근거하여 위 식을 활용하면, 두 링크의 충돌 가능 지점(충돌하거나 혹은 최소거리의 두 포인트)을 계산할 수 있다. 이렇게 계산된 충돌 가능 지점의 가중치 값은 식 (4)처럼 표현가능하며, 이를 대입하여 위치 값을 계산할 수 있다<sup>[4]</sup>.

$$w_{ic} = \frac{(S_j - S_i)^T (u_i - \zeta u_j)}{(1 - \zeta^2)} \quad (4)$$

$$w_{jc} = \frac{w_{ic} - u_i^T (S_j - S_i)}{\zeta}$$

$$\zeta = u_i^T u_j$$

만약, 임의의 두 벡터가 교차하지 않는 경우에는 계산된 가중치의 범위가 그 조건을 벗어나기 때문에, 이 경우에는 상대 링크에 가까운 끝점을 충돌 가능 지점으로 선언한다. 따라서, SA를 활용하여 특정 포인트가 아닌 링크 내 모든 점에 대한 충돌 가능성을 계산할 수 있다.

## 2.2 Modified Skeleton Algorithm(MSA)

앞서 소개된 SA는 임의의 두 링크의 벡터를 고려하여 충돌 가능지점을 계산 후 최소거리를 산출하는 방식으로써, 일반적인 충돌상황에 대한 거리계산에 유용하다. 하지만, 이러한 충돌가능지점을 기반으로 접근하는 경우, 예외의 상황이 발생하게 된다. Fig. 4은 이러한 상황을 2차원으로 단순화시켜 표현하였으며, 점 S, E는 각각 링크의 시작점과 끝점을 나타낸다. 기존 SA를 기반으로 충돌가능지점을 계산할 경우, 우선 두 링크의 충돌 지점으로 C가 산출되어 최소거리가 A1이 되지만, 링크 내에 존재해야하는 제한으로 인해  $S_i, S_j$ 가 각각  $i, j$ 링크의 충돌 가능 지점으로 결정되어 최소거리가 A1'이 된다. 하지만, 그림에서 확인되듯이 이 경우에는 A1'보다  $S_i$ 과 j링크와의 직선거리(A2)가 실질적으로 더 짧기 때문에 SA에 오류가 있음을 알 수 있다. 따라서, 이를 보완하기 위해 두 벡터간의 직선거리를 추가로 계산하고 SA와 비교하는 로직을 구성해야 한다. 직선거리는 내적을 이용한 점과 선분과의 거리 식을 적용하여 산출 가능하며, 임의의 링크의 양끝점과 상대링크와의 거리를 계산하도록 한다. Fig. 4 경우의 직선거리는 다음과 같이 계산된다.

$$A2 = \|S_j S_i\| (1 - (u_{S_j E_j} \cdot u_{S_i S_j})^2) \quad (5)$$

$u$ 는 단위벡터를 의미한다. 이때, 직선거리에 상응하는 링크위의 점은 다음 식을 이용하여 계산 가능하다.

$$\alpha = \arcsin\left(\frac{A2}{D1}\right) \quad (6)$$

$$D2 = D1 \cos(\alpha)$$

$$P_i = S_i + D2(\pm u_i)$$

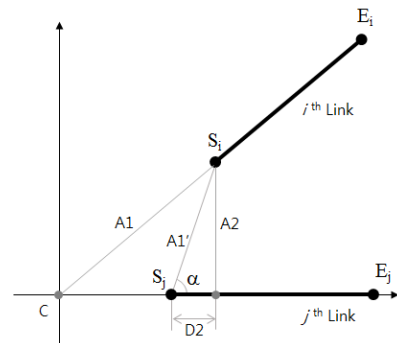


Fig. 4. Example case of SA error

결국, Fig. 5에 표현된 블록다이어그램처럼, 임의의 두 링크에 대해 계산된 직선거리(A2)와 SA로 계산된 값(A1)을 비교하여 최종적으로 실제최소거리(cDist)를 선정하고 이 값을 미리 선정한 기준치와 비교하여 충돌위험이 판단되는 경우 충돌회피를 하게 된다. 이때, 기준치는 팔, 몸통, 다리 등의 부위가 다르기 때문에, 링크마다 상응하는 기준치를 다르게 설정하도록 한다.

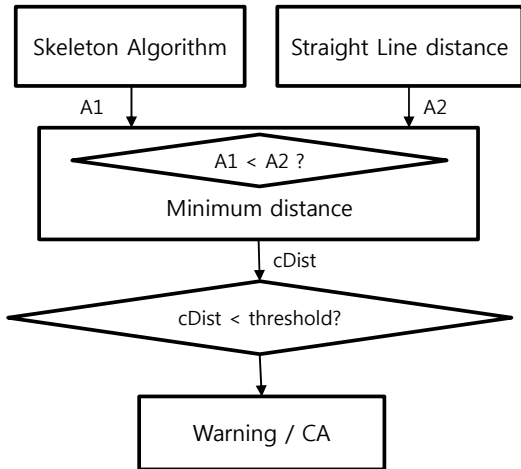


Fig. 5. MSA Blockdiagram

제안된 MSA를 활용하여 구난로봇의 자체충돌을 감지하기 위해 양팔을 기준링크로 하여 전체 링크와의 충돌을 순차적으로 스캔하도록 한다. Fig. 6는 오른팔을 기준으로 스캔하는 경우에 대한 패턴을 보여주며, 왼팔 경우에도 같은 패턴으로 적용된다.

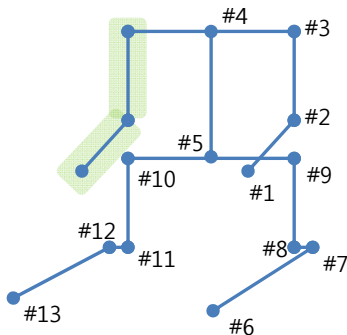


Fig. 6. Scan pattern for right arm collision

MSA로 기본적인 충돌 상황에 대한 시뮬레이션을 수행하여, 예상되는 지점에 대한 충돌이 잘 감지되는

것을 확인하였으며, Fig. 7에서와 같이 충돌 가능한 모든 경우, 즉 조인트-조인트 충돌, 조인트-링크 충돌, 링크-링크 충돌 등에 대해 모두 감지 가능한 것을 확인하였다. 빨간색, 파란색 삼각형은 양팔과 다른 링크와 충돌이 감지될 때 예상되는 지점을 나타내며, 예측한 값과 일치함을 확인하였다. Fig. 8은 SA 오류로 인한 결과를 보여준다. Fig. 7(b)와 동일한 상황에서 SA로 감지하는 경우 실제 최소거리에 해당되는 지점이 아닌 다른 값을 산출하는 것을 확인할 수 있다.

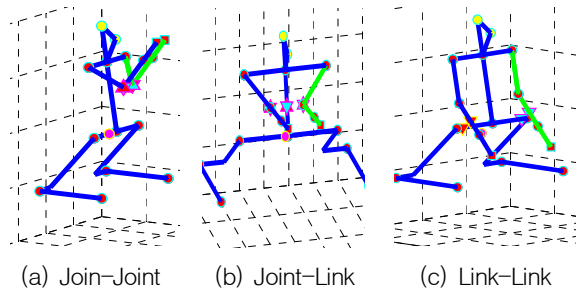


Fig. 7. All self-collision cases

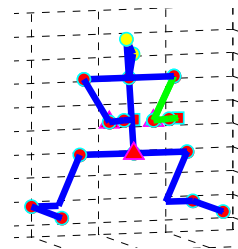


Fig. 8. Example of SA error

### 3. 역기구학 제어

일반적으로 로봇의 매니플레이터 운동은 다음과 같이 조인트의 비선형 함수로 표현된다.

$$\mathbf{x} = f(\mathbf{q}) \tag{7}$$

여기서,  $\mathbf{x}$ 는 작업공간에서의 임무상태변수이며,  $\mathbf{q}$ 는 조인트 영역의 변수들을 의미한다. 로봇의 임무는 작업공간 상에서 주어지기 때문에 이에 대한 조인트 해를 구하는데 있어, 식 (7)은 유용하지 않다. 따라서, 식 (7)를 자코비안(Jacobian)을 이용하여 속도나 가속도 레

벨로 재유도하여 이를 이용한 역기구학 문제로 해를 찾는다. 매니퓰레이터를 보다 부드럽게 제어하기 위해 본 논문에서는 속도 레벨에서도 연속성이 유지되어야 하기 때문에 다음과 같이 가속도 레벨의 2차역기구학 식을 적용하도록 한다.

$$\ddot{\mathbf{x}} = \mathbf{J}\ddot{\mathbf{q}} + \dot{\mathbf{J}}\dot{\mathbf{q}} \quad (8)$$

$$\ddot{\mathbf{q}} = \mathbf{J}^+(\ddot{\mathbf{x}} - \dot{\mathbf{J}}\dot{\mathbf{q}}) \quad (9)$$

여기서,  $\mathbf{J} \in R^{(m \times n)}$ 이며,  $m, n$ 은 각각 주어진 임무 수와 조인트의 자유도를 나타내며, +는 자코비안의 의사역행렬(Pseudo inverse)을 의미한다. 일반적으로 로봇의 자유도 수가 주어진 임무 수보다 많기 때문에, 여유 자유도를 갖는 문제가 된다. 이 경우, 여러 해가 가능하며, 이 중에서 조인트의 변화, 즉 조인트 속도가 최소가 되도록  $g = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{W} \dot{\mathbf{q}}$ 의 최소노름(Minimum norm)

인  $\mathbf{J}^+ = \mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}$ 을 적용하여 해를 찾는다.  $\mathbf{W}$ 는 각 조인트의 가중치이며, 이에 따라 조인트별로 모션제약이 가능하다. 여유자유도가 있는 문제는 기본 임무를 위반하지 않는 해의 공간 안에서 불필요하거나 의도치 않는 모션이 가능한 해가 존재할 수 있기 때문에, 이를 구속하기 위해 기본 임무의 영공간(Null space)을 활용하여 다른 임무를 부여하거나 구배투사법(Gradient projection method)을 반영한다<sup>5-7)</sup>. 본 논문에서는 충돌 회피를 주요 이슈로 다루고 있기 때문에 다음과 같이 구배투사법을 이용하여 이를 접근하도록 한다.

$$\ddot{\mathbf{q}} = \mathbf{J}^+(\ddot{\mathbf{x}}_d - \dot{\mathbf{J}}\dot{\mathbf{q}}) + (\mathbf{I} - \mathbf{J}^+\mathbf{J})\nabla h \quad (10)$$

여기서,  $\mathbf{I}$ 는 단위행렬,  $h$ 는 임의의 함수가 되며, 본 논문에서는 충돌회피를 위한 포텐셜 함수를 적용하도록 하며 이에 대한 함수정의는 다음 장에 소개된다.

위와 같은 역기구학은 개루프(open-loop) 형태를 띄고 있어, 초기오차나 적분오차 등에 의한 에러를 보상 해주지 못한다. 따라서, 오차를 0으로 수렴시키기 위해 다음과 같이 위치와 속도오차에 대한 PD제어기가 반영된 CLIK(Closed-Loop Inverse Kinematics)을 적용한다<sup>5-7)</sup>.

$$\ddot{\mathbf{q}} = \mathbf{J}^+(\ddot{\mathbf{x}}_d + \mathbf{K}_D\dot{\mathbf{e}} + \mathbf{K}_P\mathbf{e} - \dot{\mathbf{J}}\dot{\mathbf{q}}) + (\mathbf{I} - \mathbf{J}^+\mathbf{J})\nabla h \quad (11)$$

여기서,  $\mathbf{e} = \mathbf{x}_d - \mathbf{x}$ 이며,  $\mathbf{K}_D, \mathbf{K}_P$ 는 각각 양한정행렬 (Positive definite matrix) 형태의 계수를 나타낸다.

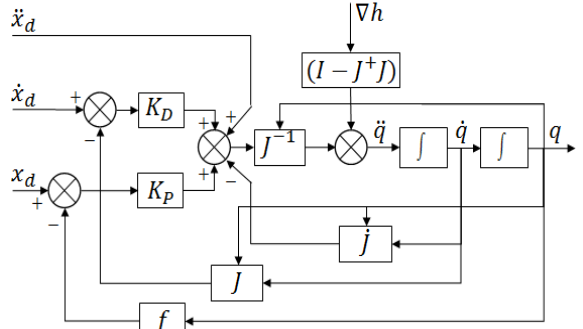


Fig. 9. Block-diagram

#### 4. 충돌회피

MSA로 충돌감지가 되면, 양팔이 충돌을 회피할 수 있도록 조인트 명령이 생성되어야 한다. 이를 위해 다음과 같이 포텐셜 함수를 기반으로 반력(Repulsive force)을 형성하였다<sup>9)</sup>.

$$V_{rep,i,j}(D_{ij}) = \begin{cases} -\frac{F_{max}}{3D_0^2}(D_{ij} - D_0)^3 & \forall D_{ij} \leq D_0 \\ 0 & \forall D_{ij} > D_0 \end{cases} \quad (12)$$

여기서,  $D_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\|$ 이며,  $D_0$ 는 상대링크와의 부피를 고려한 기준치이고,  $F_{max}$ 는 반력의 정도를 조절할 수 있는 디자인 파라미터이다. 포텐셜 함수기반의 반력은 CLIK 해의 영공간(Null space)에 반영하기 위해 조인트 변수에 대한 편미분으로 유도 가능하지만, 포텐셜함수가 조인트변수로 표현되어 있지 않기 때문에 체인룰(Chain rule)을 기반으로 다음과 같이 자코비안(Jacobian)을 이용한 식으로 변형하여 적용하도록 한다.

$$F_{rep}(D_{ij}) = \frac{\partial V_{rep}(D_{ij})}{\partial \mathbf{q}} = \frac{\partial V_{rep}}{\partial D_{ij}} \frac{\partial D_{ij}}{\partial \mathbf{p}_c} \frac{\partial \mathbf{p}_c}{\partial \mathbf{q}} \quad (13)$$

여기서,  $\mathbf{p}_c$ 는 충돌 가능지점으로 MSA에서 계산된  $\mathbf{P}_i, \mathbf{P}_j$ 를 의미하며, 충돌지점에 대한 자코비안( $\partial \mathbf{p}_j / \partial \mathbf{q}$ )은 순기구학(Forward kinematics)을 기반으로 계산가능하다.

따라서, 식 (13)을 식 (11)의  $\nabla h$ 에 적용하면, 기본 임무를 수행하면서 충돌이 예상되는 경우, 기본 임무를 방해하지 않으면서 여유자유도를 활용하여 충돌회피를 할 수 있게 된다.

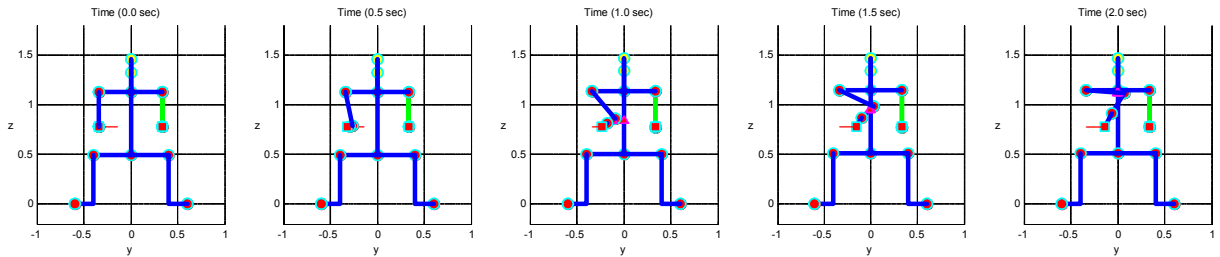
### 5. 수치 시물레이션

제안된 충돌감지 및 회피 알고리즘을 검증하기 위해 수치 시물레이션을 수행하였다. 고려된 시나리오에서 목 조인트는 고려되지 않았으며, 요구되는 하체의 양쪽 모션이 동일하기 때문에, 왼쪽 다리 조인트의 입력 값은 계산하지 않고 오른쪽 다리 조인트 값과 동일하게 적용되었다. 조인트 할당 번호는 Table 1와 같으며 어깨와 팔꿈치의 숫자는 Fig. 2에 표기된 조인트를 나타낸다.

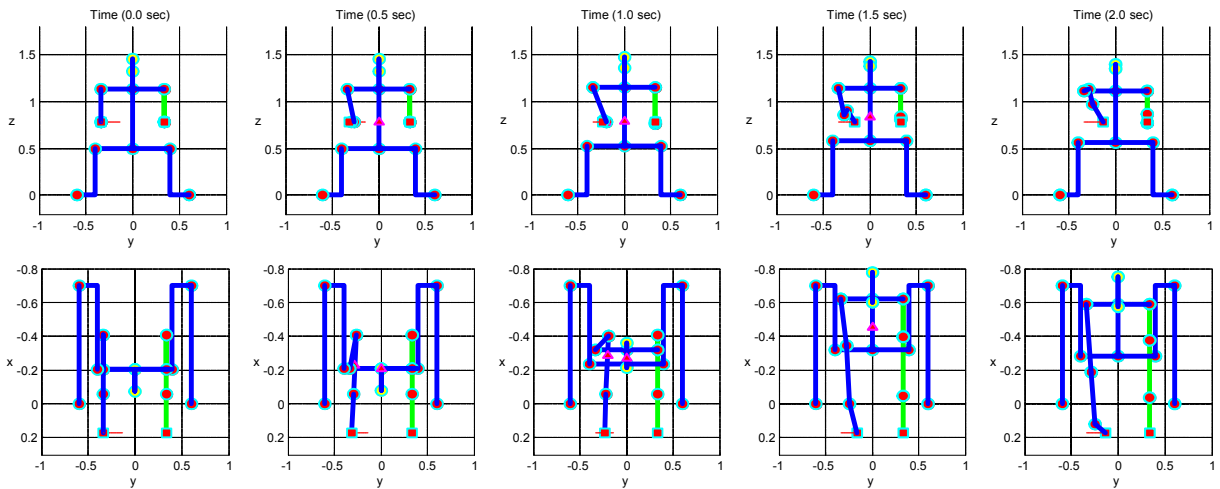
Table 1. Joint number

조인트 번호	설명	조인트 번호	설명
1	허리	8	왼쪽 어깨2
2	오른쪽 어깨1	9	왼쪽 팔꿈치1
3	오른쪽 어깨2	10	왼쪽 팔꿈치2
4	오른쪽 팔꿈치1	11	왼쪽 손목
5	오른쪽 팔꿈치2	12	오른쪽 골반
6	오른쪽 손목	13	오른쪽 무릎
7	왼쪽 어깨1		

시물레이션을 위한 기본임무는 양손의 위치궤적에 대한 임무를 반영하여, CLIK기반으로 전체 조인트의



(a) Results without collision avoidance



(b) Results with collision avoidance

Fig. 10. Comparison between with and without collision avoidance

궤적을 계산하고, 충돌회피를 위한 조인트 궤적명령은 식 (11)와 같이 기본임무의 영공간에 구배투사법으로 부여하였다.

시나리오는, 초기 준비 자세에서 왼손은 초기위치를 유지하고, 오른손은 2초 동안 몸통 쪽으로 0.2 m 수평 이동시키도록 궤적을 설정하였다. Fig. 10에는 충돌회피가 고려되지 않은 경우(a)와 고려된 경우(b)가 비교되어 있다. 붉은선은 요구되는 오른손 궤적을 의미하며, 붉은 ▲은 충돌이 감지될 때 예상되는 지점을 나타낸 것이다. 해석에 용이하도록, Fig. 10(b)는 정면도와 평면도를 모두 나타내었다. Fig. 10(a)에서 보여지듯이, 오른손이 궤적을 추종하면서 0.5초에 팔꿈치와 몸통과의 충돌이 감지되지만, 충돌회피가 되지 않아 몸통과 간섭이 예상된다. 하지만, Fig. 10(b)에서는 감지된 이후에, 충돌회피 명령이 반영되어, 팔꿈치의 모션에 변화가 생겨 몸통과의 간섭을 최소화하는 볼 수 있다. 또한, 기본 임무인 오른손 궤적을 추종성능을 확인하기 위해 Fig. 11에 양손의 위치궤적에 대한 기본임무 추종오차를 상세히 나타내었다. (R), (L)은 오른손과 왼손을 의미하며, x,y,z 축 모두에 대해서 그 오차가 매우 작다. Fig. 12에서 위그림은 오른팔과 임의 링크간의 최소거리를 나타내며, 아래그림은 충돌회피를 위해 식 (13)에 의한 반력변화를 보여주고 있다. 위 그림에서 굵은 붉은 선은 최소거리 기준치(0.2 m)를 의미하며, 0.4초부터 기준치를 침범하면서 반력이 반영되는 것을 확인할 수 있다. rS1, rS2, rE1, rE2는 각각 2,3,4,5번 관절을 의미한다. Fig. 13는 Fig. 10(b)에 묘사된 모션에 대한 명령결과로서, 기본임무를 수행하기 위한 가속도 명령과 기본임무의 영공간에 투영된 충돌회피에 의한 가속도 명령이 조합된 최종 명령을 보여준다. 수치를 살펴보면, 일부 관절에 다소 큰 값이 산출된 것을 확인할 수 있는데, 이는 충돌회피로 인한 명령이 다소 크고, 그에 대한 순간적인 모션변화로 다른 조인트들의 명령이 영향을 받는 것임을 알 수 있다. 이러한 경향을 줄이기 위해서는 식 (12)에서의  $F_{max}$  값과 최소거리 기준치에 대한 트레이드오프를 통해 재설정해야하며, 향후 연구에서는 실제 설계치를 반영한 가속도레벨을 정의하여, 디자인 파라미터를 선정할 계획이다.

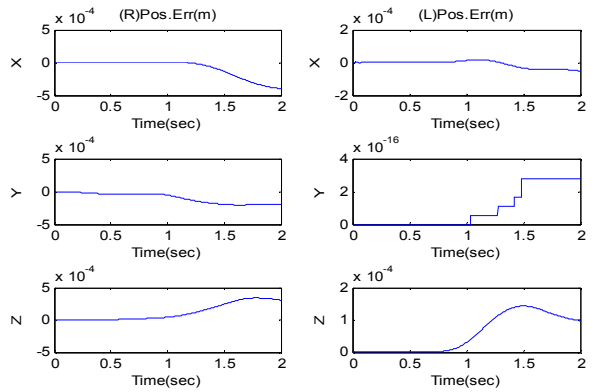


Fig. 11. Error of both hands for basic task

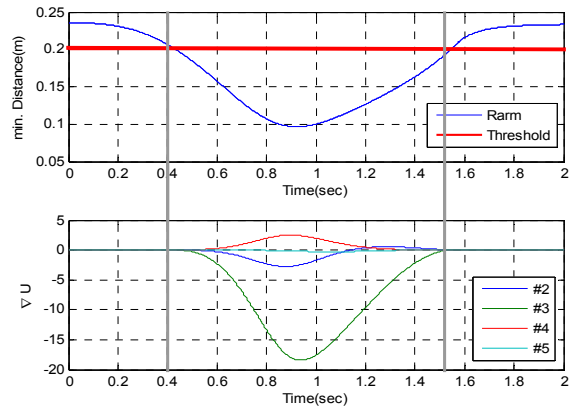


Fig. 12. Minimum distance between any two links and force by collision avoidance

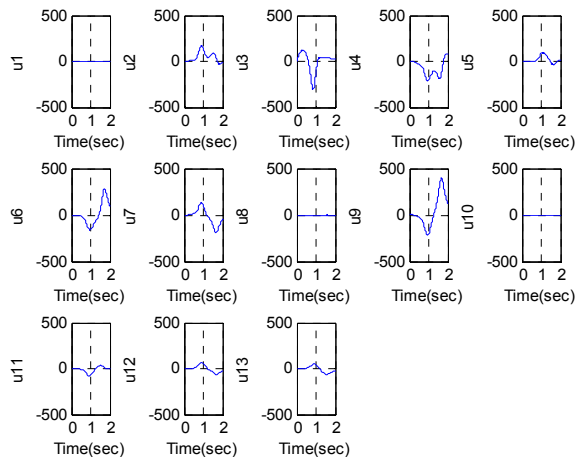


Fig. 13. Joint commands reflecting basic task and collision avoidance

## 6. 결 론

본 논문에서는 로봇의 자체 충돌을 미리 감지하기 위해 충돌예상지점의 거리를 계산하는 기존의 골격 알고리즘의 경우에 따른 문제점을 분석하고 이를 보완한 새로운 알고리즘을 제안하였다. 제안된 알고리즘은 모든 가능한 충돌 경우에 대해 감지하는 것을 확인하였으며, 이를 기반으로 충돌회피를 구현하되, 기본 임무를 왜곡하지 않도록 CLIK의 영공간에 구배투사법으로 충돌회피 명령을 반영하였다. 수치 시뮬레이션을 거쳐 알고리즘을 검증하였으며, 충돌회피로 인한 조인트 명령이 다소 크게 산출되었으나, 이는 향후 실제 구난로봇의 물리적 특성을 반영하여 충돌회피의 디자인 파라미터를 튜닝하여 보완 할 계획이다.

## References

- [1] S. Hong, W. Lee, S. Kang, Y. Park, "Kinematic Control of Rescue Robot for Managing Multiple Tasks," GSDC Annual Conference Proceedings, DCC, Daejeon, 2013.
- [2] W. Lee, S. Hong, B. Choi, Y. Kang, "Motion Control for Resuce Robot by Whole-Body Control based on Task-Priority," KIMST Annual Conference Proceedings, ICCjeju, Jeju, 2014.
- [3] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi, "A Fast Procedure for Computing the Distance between Complex Objects in Three-Dimensional Space," IEEE Journal of Robotics and Automation, Vol. 4, No. 2, pp. 193-203, 1998.
- [4] A. D. Santis, A. Albu-Schaffer, C. Ott, B. Siciliano, and G. Hirzinger, "The Skeleton Algorithm for Self-Collision Avoidance of a Humanoid Manipulator," in Proc. of the 2007 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2007.
- [5] B. Sciliano and J. Slotine, "A General Framework for Managing Multiple Tasks in highly Reduandant Robotic Systems," Proc. of Int. Conf. on Advanced Robotics, Pisa, Italy, pp. 1211-1216, 1991.
- [6] B. Sciliano, "Kinematic Control of Redundant Robot Manipulators : A Tutorial," Journal of Intelligent and Robotic Systems, Vol. 3, pp. 201-212, 1990.
- [7] B. Sciliano, L. Sciavicco, L. Villani, and G. Oriolo, "Robotics : Modeling, Planning and Control," Springer, 2009.
- [8] B. Sciliano, "A Closed-Loop Inverse Kinematic Scheme for On-Line Joint-Based Robot Control," Robotica, Vol. 8, pp. 231-243, 1990.
- [9] A. Dietrich, T. Wimbock, A. Albu-Schaffer, and G. Hirzinger, "Integration of Reactive, Torque-Based Self-Collision Avoidance into a Task Hierarchy," IEEE Transaction on Robotics, Vol. 28, No. 6, 2012.