論文94-31B-4-5

다작업 로보트 매니퓰레이터의 최적 시간 경로 계획을 위한 미소 유전알고리듬의 적용

(Planning a Minimum Time Path for Multi-task Robot Manipulator using Micro-Genetic Algorithm)

金 容 虎*, 沈 貴 寶**, 趙 鉉 讚***, 全 洪 兌*

(Yong Ho Kim, Kwee Bo Sim, Hyun Chan Cho and Hong Tae Jeon)

要 約

본 논문에서는 로보트 매니퓰레이터의 최적시간경로 계획을 위해 최적화 알고리듬의 일종인 미소유전알고리듬을 사용한다. 로보트 시스템이 요구되는 작업을 최소 시간내에 수행할수 있도록 해주는 최적 경로 계획은 생산성 극대화에 중요한 역할을 한다. 그러나 대부분의 방법들은 아직도 엄청난 계산량이 요구되며 최적의 값을 찾는다는 보장도 없다. 이러한 어려운 점을 극복하기 위한 한가지 방법으로 미소 유전 알고리듬을 이용해서 최소 시간 경로 계획 문제를 해결한다. 제안된 방식의 효용성은 2 d.o.f 평면 로보트 메니퓰레이터에 모의 실험함으로서 입증된다.

Abstract

In this paper, Micro-Genetic algorithms(μ-GAs) is proposed on a minimum-time path planning for robot manipulator, which is a kind of optimization algorithm. The minimum-time path planning, which can allow the robot system to perform the demanded tasks with a minimum execution time, may be of consequence to improve the productivity. But most of the methods proposed till now suffers from a significant computation burden and can't often find the optimul values. One way to overcome such difficulties is to apply the Micro-Genetic Algorithms, which can allow to find the optimul values, to the minimum-time problem. This paper propose an approach for solving the minimum-time path planning by using Micro-Genetic Algorithms. The effectiveness of the proposed method is demonstrated using the 2 d.o.f plannar Robot manipulator.

* 正會員, 中央大學校 電子工學科
(Dept. of Elec. Eng., Chungang Univ.)
** 正會員, 中央大學校 制御計測工學科
(Dept. of Control & Instrumentation Engineering, Chungang Univ.)
*** 正會員, 韓國技術教育大學校 電子工學科
(Dept. of Electronic Engineering, KITE)
接受日字 : 1993年 9月 28日

I. 서 론

산업용 로보트는 다수의 링크들과 조인트들로 구성되어 있으며 공장자동화에서 중추적인 역할을 담당하는 컴퓨터로 제어되는 기계적인 매니퓰레이터 시스템이다. 이러한 매니퓰레이터는 프로그램의 변환을 통하여 다양한 작업을 할수있기 때문에 FMS의 중요한 work cell로 인식되고 있다.

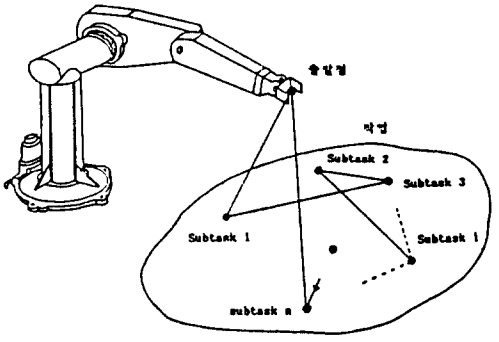
일반적으로 산업용 로보트가 수행하는 작업(drilling,

welding, painting, assembling, 그리고 운반작업 등)들은 여러개의 소작업(subtask)들과 그 소작업들 사이에서의 로보트의 주행경로(traveling path)들로 구성되어 있으며, 다음 두 종류로 대분될 수 있다.

첫째, 각 소작업들 사이에 수행 순위 (order of execution)가 미리 규정되어 있는 작업 (그림 1).

둘째, 수행 순위가 규정되어 있지 않은 소작업들로 구성된 작업.

산업용 로보트가 상기 2부류 작업들을 수행하는데 소요되는 작업 시간은 공히 각 소작업의 수행시간들과 소작업 사이에서의 주행시간들의 합이다. 따라서 이 작업시간들의 최소화 (즉, 소작업 수행시간과 주행 시간의 최소화)는 로보트 생산성 극대화에 크게 기여 할 것이다. 그러나 많은 예에서 각 소작업의 수행시간은 일정하다고 간주 할 수 있기 때문에 작업시간의 최소화는 주행시간의 최소화와 일치 한다고 생각할 수 있다.



수행순서 : 출발점 → 소작업 1 → 소작업 3 → 소작업 2 → ··· → 소작업 i → ··· → 소작업 n → 출발점

그림 1. 수행 순위가 규정된 n개의 소작업 들로 구성된 작업

Fig. 1. A Task consisting of n subtasks with the prespecified execution order.

작업시간의 최소화 (즉, 주행시간의 최소화)는 앞서 분류한 두 종류의 작업에 따라 각기 다르게 진행되어야 한다. 전자의 경우, 즉 작업 순서가 미리 규정된 경우에는 그 주행경로들이 정해져 있기 때문에 기존의 연구 결과([1] , [2])들에 의해 해결될 수 있다.

그러나 후자의 경우에는 각 소작업들 사이에서의 주행시간의 최소화 뿐만 아니라 주행경로의 선정(혹은, 각 소작업의 수행순위 결정)이 전체 작업 시간의 최소화에 큰 영향을 미친다. 따라서 후자의 작업인 경우에는 각 소작업 사이에서의 주행시간의 최소화 뿐만 아니라 최소시간 측면에서의 경로선정(selection of path)에 관한 연구가 동시에 이루어져야 한다.

본 논문에서는 각 소작업에서 로보트의 주행경로 (즉, 주행 순위)가 정해져 있지 않은 경우, 그 작업시간의 최소화를 얻기 위한 경로 선정에 관한 방법이 제시 된다. 최적의 주행 경로를 얻기 위해 제안된 방법은 자연계의 진화 이론을 바탕으로 이루어진 것으로, 최근에 최적화 및 탐색문제, 제어규칙 찾는문제 그리고 신경회로망 학습문제등 여러부문에 많이 응용되고 있는 유전알고리듬(Genetic Algorithm)이다. [4] [6] [7]

본 논문의 구성은 다음과 같다. 2 장에서는 유전알고리듬을, 3 장에서는 각 소작업 사이에서의 최소 주행시간을 결정하기 위한 방법이, 그리고 4장에서는 최적의 주행경로를 얻기위한 유전알고리듬의 적용방법이 설명된다. 5장에서는 제안된 방법의 컴퓨터 모의 실험결과가 제시되고 6장에서는 결론 및 향후 연구방향이 논의 된다.

Ⅱ. 유전 알고리듬 (Genetic Algorithm)

1970년대초 John Holland에 의해 본격적으로 연구되기 시작한 유전 알고리듬은 자연생태계의 진화과정에서 관찰된 몇가지 처리과정에서 '적자생존(Survival of the fittest)'의 원리를 컴퓨터 알고리듬과 결합시켜 정립된 최적화(optimization) 알고리듬이다. [4] [5]

유전알고리듬은 자연생태계의 진화매카니듬을 모방하였는데 실제로 자연계의 진화과정이 모두 밝혀져 있지는 않지만 중요한 몇가지는 알려져 있어 유전알고리듬은 이러한 진화과정에서 일부 관찰된것을 사용하였다. [6] 또한 유전알고리듬을 문제해결에 적용 시키기 위한 2가지 매카니듬이 있다.

첫째, 염색체상에 문제의 해를 부호화 하는 방법

둘째, 문제속에서 염색체의 가치 측정을 하는 평가(적합도)함수 설정 방법

첫번째의 경우 부호화 하는 기술은 문제마다 다르며 부호화는 bit 스트링으로 수행한다. 두번째의 경우 평가함수는 유전 알고리듬과 해결해야 할 문제를 결합시켜주며 또한 염색체를 입력으로 받아 해결될 문제에 대한 염색체의 수행 측정이 되는 값을 제공한다. 평가함수는 환경이 자연진화에 하는 것과 같은 역할을 유전알고리듬에 주며 환경과 개개의 상호작용은 적합도의 측정이고 염색체와 평가함수의 상호작용은 재생산시 유전알고리듬이 사용하는 적합도의 측정

을 제공한다.

John Holland가 상기의 과정을 컴퓨터 알고리듬과 결합시켜 정립한 유전알고리듬은 다음과 같이 구성된다.

[단계 1] 10 진수 인자 1과 0으로 구성된 스트링형으로 재표현하기 위한 스트링의 길이를 선정함.

[단계 2] 단계 1의 스트링 을 이용해 초기집단(population)을 구성함.

[단계 3] 집단의 각 스트링 을 10진수 인자로 변환하고 적합도를 계산

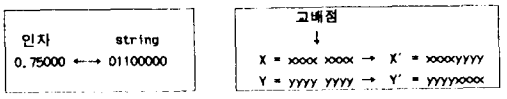
[단계 4] 적합도에 따라 해당 스트링 들을 선택함.

[단계 5] String 간의 교배와 돌연변이등의 진화과정을 거쳐 새로운 집단을 구성함.

[단계 6] 단계 3 이하의 과정을 반복함. (적합도가 수렴 할 때까지)

상기과정의 각 단계에 대한 설명은 다음과 같다.

그림 2.(a)는 단계 1 에서의 인자의 스트링 표현과 그 역 표현(즉 스트링 표현으로부터 인자로의 표현)을 보여주고 있으며 단계 3에서의 적합도는 함수에 대한 최적 정도를 의미한다. 그리고 단계 4에서의 스트링 선택은 적합도의 높은 정도에 의해서 이루어지며 선택방법은 Roulette 선택과 Deterministic sampling 선택 방법이 있다. 그림 2.(b)는 8비트인 두 스트링 X와 Y의 교배를 보여주고 있다. 이때 교배점(crossing site)은 임의 난수로 선택하게 된다. 돌연변이는 교배와 동시 행해지며, 임의 비트를 선태하여 비트전환으로 이루어진다(단계5). 한 예로 X = 1001 일때 임의로 선택된 비트가 3번째이면 돌연변이를 한 새로운 X' 는 1011이 된다.



(a) 인자와 스트링 (b) 스트링 X와 Y의 교배

그림 2. 인자의 스트링 재표현과 그 교배

Fig. 2. string representation of parameter and crossover.

이 유전알고리듬은 미지수 함수 Y=G(x)의 최적해를 발견하는 모의진화(Simulated evolution) 형의 탐색(Search) 알고리듬의 성격을 갖으며 다음과 같은 특성을 갖는다.[4]

첫째, 유전 알고리듬은 해공간(solution space)에서 단일 해를 사용하는 것이 아니라 해집단(population of solution space)을 이용한다.

둘째, 유전이론은 'blind' 하다. 즉 최적화 함수 정보(미분가능성,연속성 등)를 필요치 않는다.

세째, 유전 알고리듬은 global 한 최적해의 발견을 가능케 한다.

상기 유전 알고리듬은 일반적으로 대규모의 유전자 집단을 사용하는데 특히 단순 유전 알고리듬(SGA: Simple Genetic Algorithms)에서는 유전자 집단의 크기가 30에서 200개 정도이다. 따라서 유전 알고리듬의 계산을 위한 시간소비량은 크다고 할 수 있다. 계산시간은 적은 집단의 선택으로 줄일 수 있는데 이를 위해 제안된 방법이 '미소 유전알고리듬(μ-GA: Micro-Genetic Algorithms)이다.[5]

본 논문에서는 로보트 매니퓰레이터의 최적의 작업경로를 구하기 위해 μ-GA를 수정하여 적용한다. μ-GA를 이용한 최적시간경로 결정 수행순위가 규정되어 있지 않은 소작업들로 구성된 작업의 경우 소작업들은 다음과같은 성질을 갖는다고 가정한다.

(1) 각 소작업의 수행 우선 순위는 동일하다.

(2) 임의의 두 소작업 사이에서의 수행경로는 직선경로이다.

(3) 모든 소작업들을 수행한 후 매니퓰레이터는 처음 출발점 소작업으로 귀환한다.

상기 작업을 수행하기 위해 매니퓰레이터가 선택할 수 있는 주행 경로들은 많이 있지만 최소 시간 측면에서는 그 경로들은 한정되어 있다. 이 한정된 경로, 즉 최적 시간 측면에서의 경로를 결정하기 위해 본 절에서는 앞에서 제안된 μ-GA를 도입한다.

일반적으로 매니퓰레이터가 주어진 작업을 최소 시간내에 완료 하기 위해서는 모든 소작업 들의 수행시간과 소작업 들 사이에서의 전체주행 시간의 최소화가 필요하다. 그러나 각 소작업의 수행 시간과 임의의 두 소작업들 사이에서의 최소 주행시간을 규정할 수 있다면, 주어진 최적 시간 경로 설정 문제는 소작업들의 수행 순서(order of execution)를 결정하는 문제로 변환 된다. 더구나 두 소작업 사이에서의 최소 주행 시간을 거리 변수로 치환 한다면 상기 문제는 기존의 TSP 문제로 간주할 수 있으며 문제해결에 유전 알고리듬의 도입을 가능케 한다.

유전 알고리듬에 의한 문제 해결을 위해 두 소작업 사이에서의 최소 주행 시간 결정 과정이 다음에 설명된다.

Ⅲ. 각 소작업 사이에서의 최소 주행 시간 결정

작업 공간내에 위치한 i번째 소작업(i=1,2,…,n)의

위치(position) 와 방위(orientation) 는 다음과 같은 4×4 Homogeneous transformation matrix 에 의해 표현될 수 있다.

$$H_i(\in R_{4\times4})=\begin{bmatrix} ni & s_i & a_i & p_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

여기에서 $n_i(=[n_{xi}, n_{yi}, n_{zi}]^T)$, $s_i(=[s_{xi}, s_{yi}, s_{zi}]^T)$, 그리고 $a_i(=[a_{xi}, a_{yi}, a_{zi}]^T)$ 는 각각 방위를 나타내는 단위 normal, slide, approach 벡터들이며 $p_i(=[p_{xi}, p_{yi}, p_{zi}]^T)$ 는 위치 백터이다.

i 번째 소작업을 수행하는 동안, end-effector 의 카르테시안 주행속도 (Cartesian travelling velocity) 성분은 0 으로 간주한다. 따라서 i번째 소작업 수행 후 매니퓰레이터가 일정한 위치 및 방위 속도를 갖고서 직선 경로를 수행 하기 위해서는 일정한 가속구간이 필요하며 i+1 번째 소작업을 수행하기 직전에는 직선 주행속도의 감속을 위한 감속 구간이 요구된다. 이러한 가속 및 감속구간을 설정하기 위해 중간 경로점들을 다음과 같이 설정한다. (그림 3)

$$\begin{aligned} H_{ai} &= H_i + k(H_{i+1} - H_i) \\ &= [n_{ai}, a_{ai}, s_{ai}, p_{ai}] \end{aligned} \tag{2}$$

$$\begin{aligned} H_{di} &= H_{i+1} + k(H_{i+1} - H_i) \\ &= [n_{di}, a_{di}, s_{di}, p_{di}] \end{aligned} \tag{3}$$

여기에서 k($0\le k\le 1$) 은 감/가속 구간의 결정을 위한 스칼라 값이며 구간 H_{ai}-H_i 는 가속 구간 그리고 구간 H_{di}-H_i 는 감속 구간이 된다.

한편 각 구간에서 매니퓰레이터 end-effector 의 주행 시간들을 t_{ai}, t_{li}, 그리고 t_{di} 라 하면 각 구간에서의 end-effector 의 카테르시안 선형 속도와 가속도 성분들을 다음과 같이 얻어진다.

$$V_{ai} = \frac{V_{li}}{t_{ai}}t, \quad V_{ai} = \frac{V_{li}}{t_{ai}}t, \quad \text{(가속구간)} \tag{4}$$

$$V_{li} = \frac{P_{di} - P_{ai}}{t_{li}}, \quad V_{li} = 0 \quad \text{(선형구간)} \tag{5}$$

$$V_{di} = \frac{V_{li}}{t_{di}}(t - t_{ai} - t_{li}), \quad V_{di} = \frac{V_{li}}{t_{ai}} \text{(감속구간)} \tag{6}$$

여기에서 V_{ai}, V_{li}, V_{di}는 각 구간에서의 3×1 선형 속도 벡터들이며, V_{ai}, V_{li}, V_{di}는 3×1 선형 가속도 벡터들이다. 마찬가지로 각 구간에서의 각 속도와 가속도들은 다음과 같이 유도된다.

$$\omega_{ai} = \frac{\omega_{li}}{t_{ai}}t, \quad \dot{\omega}_{ai} = \frac{\omega_{li}}{t_{ai}} \qquad \text{(가속구간)} \tag{7}$$

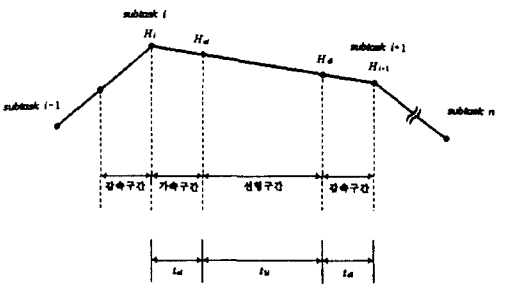


그림 3. 가속 및 감속 구간의 설정
Fig. 3. Establishment of the acceleration and deceleratin sections.

$$\omega_{li} = \frac{\theta_{li}}{t_{ai}}\rho_{li}, \quad \dot{\omega}_{li} = 0 \qquad \text{(선형구간)} \tag{8}$$

$$\omega_{di} = \frac{\omega_{li}}{t_{di}}(t - t_{ai} - t_{li}), \quad \dot{\omega}_{di} = \frac{\omega_{li}}{t_{di}} \text{ (감속구간)} \tag{9}$$

여기에서 ω_{ai}, ω_{li}, ω_{di} 는 각 구간에서의 3x1 각속도를 나타내며 ω_{ai}, ω_{li}, 그리고 ω_{di}는 3x1 각 가속도 벡터이다. 또한 각 θ_{li}와 회전 기준 벡터 $\rho_{li}(\in R^3)$ 는 다음식에 의해 정의된다. [11]

$$\theta_{li} = \left\{\left[(n_{di})^T n_{ai} + (s_{di})^T s_{ai} + (a_{di})^T a_{ai} - 1\right]\right\} \tag{10}$$

$$\rho_{li} = \frac{1}{2\sin\theta_{li}}\begin{bmatrix} (a_{di})^T s_{ai} - (s_{di})^T a_{ai} \\ (n_{di})^T a_{ai} - (a_{di})^T n_{ai} \\ (s_{di})^T n_{ai} - (n_{di})^T s_{ai} \end{bmatrix} \tag{11}$$

한편 매니퓰레이터는 제한된 구동력을 가지므로 end-effector의 선형속도와 가속도 그리고 각속도와 각가속도는 다음과 같은 제한치를 갖는다고 가정할 수 있다.

$$\|V(t)\| \le K_v \tag{12a}$$

$$\|\dot{V}(t)\| \le K_a \tag{12b}$$

$$\|\omega(t)\| \le K_w \tag{12c}$$

$$\|\dot{\omega}(t)\| \le K_{wa} \tag{12d}$$

상기 제한값과 식 (4)부터 (9)를 이용해 i번째 소작업과 (i+1)번째 소작업 사이에서의 최소 주행 시간 t_i^* 는 다음과 같이 구할 수 있다.

$$t_{li}^* = \max\left\{\|P_{di} - P_{ai}\|/k_v, \ \|\theta_i\|/k_{wa}\right\} \tag{13}$$

$$t_{ai}^* = t_{di}^* = \left\{\|P_{li}^*\|/k_a, \ \|\omega_{li}^*\|/k_{wa}\right\} \tag{14}$$

$$t_i^* = t_{ai}^* + t_{li}^* + t_{di}^* \tag{15}$$

여기에서 $(\cdot)^*$은 최소 시간을 의미하며 다음과 같이 된다.

$$P_{1i}^* = (P_{di} - P_{ai})/t_{1i}^*, \quad \omega_{1i}^* = |\theta_{1i}|/t_{1i}^* \tag{16}$$

따라서, 로보트가 수행해야 할 소작업의 갯수가 결정되면 각 소작업간의 최소 주행시간이 상기의 식들로 부터 구해지게 된다. 각 소작업간의 최소 주행시간을 사용하여 다음의 유전알고리듬을 통해 최소 주행경로를 결정하게된다.

Ⅳ. μ-GA에 의한 최소 시간 경로 결정

앞서 언급 했듯이 식(15)에 의해 결정된 임의의 두 소작업 사이에서의 최소주행시간을 거리변수로 바꾸어 생각하면, 주어진 문제는 기존의 TSP 문제와 유사하게 된다. 따라서 본 절에서는 문제 해결을 위해 μ-GA을 이용하여 주어진 작업공간에서 최적시간경로를 결정 한다.

문제해결을 위해서 다음과 같은 선택이 필요하다

첫째, 문제 해결을 위한 스트링 표현은 다음과 같은 '인접표현(adjacency representation)'과 10진수 스트링을 사용한다.

소 작 업	1 2 3 4 5
경로 스트링	5 4 1 3 2
작 업 순 서	1 → 5 → 2 → 4 → 3 (→ 1)

그림 4. 인접표현에의한 스트링 구성
Fig. 4. string composition by adjacent representation.

둘째, 소작업에 순서가 있으므로 순서를 고려한 Greedy 교배를 사용한다. [1]

문제해결을 위해 수정 적용된 μ-GA의 단계는 다음과 같다.

[단계 1] 초기 집단의 크기를 임의로 5개를 선택하고 이후에는 4개는 임의로 선택하고 1개는 이전 의 탐색으로부터 나온 우수한 것으로 선택한다.(이때 스트링은 10진수로 표현하며 한개의 스트 링내에 중복된 숫자가 나오지 않도록 선택한다.)

[단계 2] 적합도를 계산한 후 제일 우수한 것을 구한 다음 이 스트링을 5번째 스트링으로 두고 다음 세대로 넘긴다.(엘리트 규칙)

[단계 3] 남은 4개의 스트링을 재생산하기 위해 임의로 배열하여 8쌍의 짝을 지운다. (이때 같은 스트링 끼리의 짝짓기는 피한다.)

[단계 4], 교배를 시킨다.(Greedy crossover 사용)

[단계 5] 수렴성을 검사한 후 수렴하면 [단계 1] 로, 수렴하지 않으면 [단계 2] 로 간다.

한편, [단계2] 에서의 적합도는 집단내에서 진화대상의 요구가 되는 스트링을 선택하는 기준이 되며 다음과 같이 설정한다.

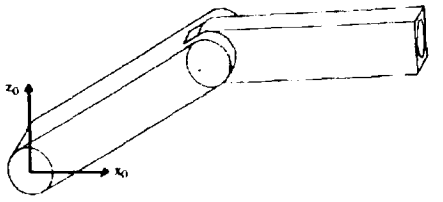
$$Fintness = \frac{1}{\sum_{i=1}^{n} t_i^*} \tag{17}$$

여기에서 n은 경로의 수이고 t_i^*는 식(15)에서 구한 i번째 소작업과 i+1번째 소작업 사이에서의 최소 주행시간이다.

Ⅴ. 실험 결과

본 논문에서는 2자유도의 평면 로보트 매니퓰레이터를 사용했으며 μ-GA가 최적의 경로를 찾는가를 시험하기 위해 20개의 정방형 으로 위치시킨 소작업을 선택하였고 실제적인 작업을 고려하여 임의로 위치시킨 30개의 소작업을 선택하였다.

다음 그림 5는 실험에 사용된 2자유도의 평면 로보트 매니퓰레이터이다.



	Link 1	Link 2
Length	0.53m	0.47m
Mass	2.5Kg	2.0Kg

그림 5. 2축 평면 로보트 매니퓰레이터
Fig. 5. Planner Robot manipulator of 2 d.o.f.

그림 6은 20개의 정방형 소작업의 경로를 찾기 시작할때의 경로이고 그림 7은 μ-GA가 찾은 최적 경로를 나타내고 있다. 이때 최적의 경로가 1개 이상일

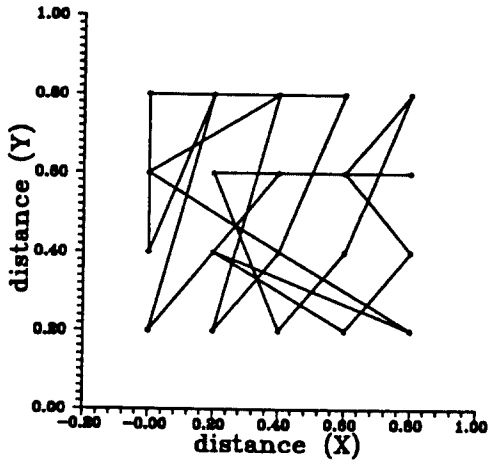


그림 6. 정방형 소작업의 초기 경로
Fig. 6. Initial path of qubic-substasks.

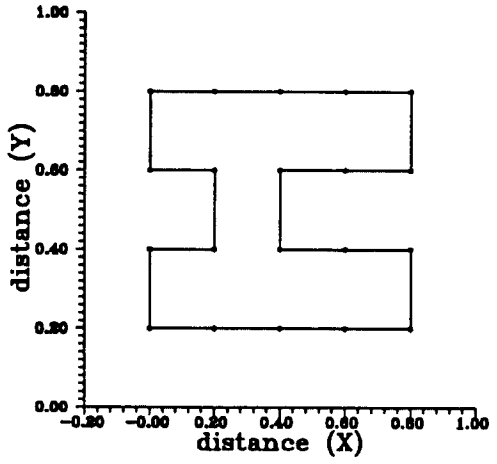


그림 7. 정방형 소작업의 최적 경로
Fig. 7. Optimal path of qubic-substasks.

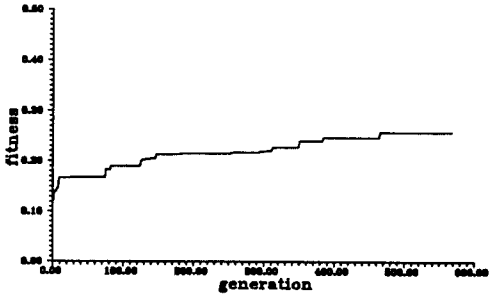


그림 8. 정방형 소작업의 적합도
Fig. 8. Fitness of qubic-substask.

수 있다.

그림 8은 적합도값과 Generation과의 관계를 보여주고 있는데 약 500 Generation에서 최적의 경로를 찾음을 알 수있다. 500 Generation이라 할지라도 집단수가 5개에 불과하므로 소요된 계산시간은 매우 작다. 그림 9는 임의의 30개 소작업의 경로를 찾기 시작할때의 경로이며 그림 10은 μ-GA가 찾은 최적 경로중의 하나를 보여주고 있다. 그림 11은 적합도값과 Generation과의 관계를 나타내고 있으며 약

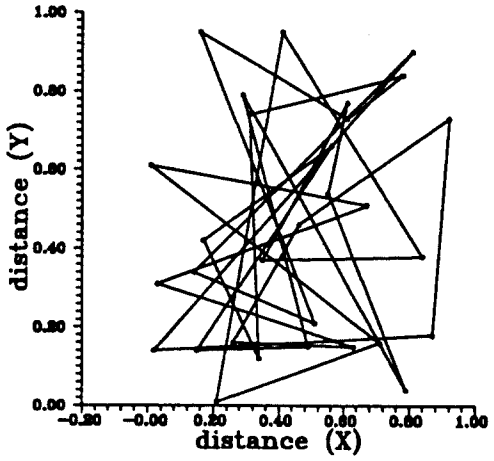


그림 9. 임의로 위치한 소작업 30개에 대한 초기 경로
Fig. 9. Initial path of randomly plased 30 subtasks.

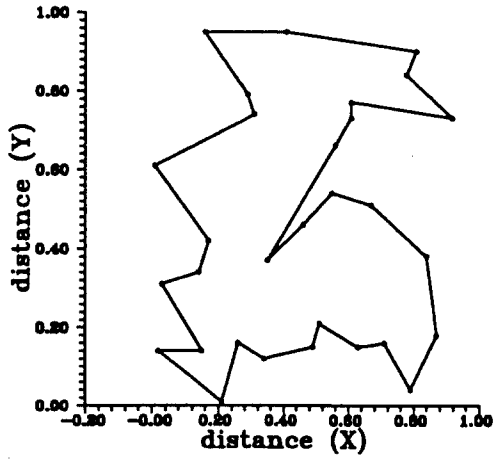


그림 10. 임의로 위치한 소작업 30개에 대한 최적 경로
Fig. 10. Optimal path of randomly plased 30 subtasks.

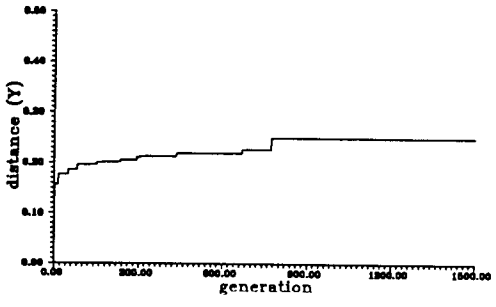


그림 11. 임의의 소작업의 30개에 대한 적합도
Fig. 11. Fitness of randomly plased 30 subtasks.

800번 정도에서 적합도값이 수렴하므로 최적 경로일 가능성이 있음을 알 수 있다.

Ⅵ. 결론 및 앞으로의 연구방향

최적의 경로를 찾는 문제는 일반적으로 시간소비가 매우 큰 작업이라고 할 수 있다. 또한 경로를 찾았다 하더라도 최적인지는 의문의 여지가 있다. 이 두가지의 문제를 해결하기 위해 본 논문에서는 μ-GA과 발견적 교배(Heuristic crossover)방식 중에서 Greedy 교배법을 사용하여 만족할만한 결과를 얻었다. 향후, 2대 이상의 협력 로보트 작업의 최적 경로를 찾는데 있어, 협력 로보트들의 충돌을 회피하기 위한 방법과 더불어 사용되면 좋은 결과가 나올것으로 생각된다.

參 考 文 獻

[1] J.Y.S.Luh and C.S.Lin, "Optimum path Planning for Mechanical Manipulators," *Trans.* of the ASME, vol.102, June, 1981, pp.142-151.

[2] J.E.Bobrow, S.Dubowsky and J.S.Gibson, "Time Optimal control of robot manipulators along specified paths," Int. Journal of Robotics Research, vol.4, no.3, pp.3-17, 1985.

[3] K.S.Fu, R.C.Gonzalez, and C.S.G.Lee, Robotics control, sensing, vision and Intelligence, McGrow-Hill.

[4] D.E.Goldberg, Genetic Algorithms in Search, Optimization & Machine Learning, Addison-Wesley Publishing Company, 1989.

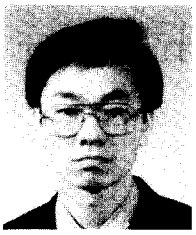
[5] Krishnakumar.K, "Micro-Genetic Algorithms for stationary and non-stationary function optnmization," *Proc.* SPIE Intell. Cont. adapt. Syst., 1196, 289-296, Nov.7-8, Philadelphia, Pa., 1989.

[6] Yuval Davidor, Genetic Algorithms and Robotics, A Heuristic Strategy for Optimization, World Scientific, 1991.

[7] L.Davis, Handbook of Genetic Algorithms, Van Nostrand ReinHold, NewYork, 1991.

[8] 박세희, 김용호, 심귀보, 전홍태, "유전알고리듬을 이용한 퍼지 규칙 베이스의 자동생성", 전자공학회지, vol.29-B, pp.60-67, 1992.

著 者 紹 介



金 容 虎(正會員)
1962年 3月 4日生. 1986年 광운대학교 전자공학과 졸업. 1988年 중앙대 대학원 전자공학과(석사). 1992年 ~ 현재 중앙대 대학원 전자공학과 박사과정. 주관심 분야는 Robotics, Fuzzy & Neural network, Optimization theory(Genetic algorithms,etc) 등

•

趙 鉉 讚(正會員) 第 18卷 第 10號 參照
현재 한국기술교육대학 조교수

沈 貴 寶(正會員) 第 29卷 第 1號 參照
현재 중앙대학교 제어계측공학과 조교수

•

全 洪 兌(正會員) 第 29卷 第 1號 參照
현재 중앙대학교 전자공학과 부교수