

로봇의 소비 에너지 최소화를 위한
궤도 계획에 관한 연구

*최도영, 권승희, 박인갑, 김형래
건국대학교 전자공학과

A Study on the Minimum - Energy Trajectories
Planning for Industrial Robot

*Choi-Doeyong, Kwon-Synghoe, Park-Inkap, Kim-Hyunglae
Dept. of Electronics, Konkuk University

ABSTRACT

The problem of minimization of energy, consumed by the robot manipulator, is important, notably for larger manipulators, higher working speeds, and loads.

Obviously, the stated problem requires the application of optimal control theory, which is being successfully applied for linear systems and certain classes of nonlinear systems. However, the application of optimal control theory (in energy or time optimization) leads to substantial practical difficulties, so that significant simplifications are usually performed, either in model complexity or by neglecting the existing constraints.

In this paper the problem of obtaining such an optimization method, which would take into account the complete system dynamics and all the constraints is considered. The only method found to be suitable for such a complex optimization should be based on dynamic programming.

1. 서 론

산업용 로봇은 생산성 향상과 생산품의 품질 향상을 위한 산업 자동화에 중요한 수단으로 등장하였다. 이에 따라 고속 작업이 요하고 큰 부하를 다루는 곳에서 로봇 매니퓰레이터가 소비 하는 에너지를 최소화시키는 것이 중요한 문제로 대두되었다.

그러나 로봇 매니퓰레이터는 복잡하고 고도의 비선형 다변수 시스템에 속하고 또한, 메카니즘 자체의 구동기(actuator)의 기구학적, 물리적 특성과 작업 공간에서 다른 물체의 충돌의 위험성을 갖는 등 엄격한 제약 조건 때문에 로봇의 제어는 많은 어려움이 따른다. 이러한 어려움 때문에 로봇의 제어를 위한 알고리즘은 그 구조를 궤도 계획(trjectory planning)과 경로 추적(path tracking)의 두 부분으로 나눌 수 있다.

궤도 계획 단계에서는 로봇의 동역학이 고려되어야 하고 동역학 모델이 정확할수록 로봇이 가지고 있는 능력을 보다 효율적으로 이용할 수 있다. 또한, 최소의 에너지를 소비하는 궤도 계획을 위해서는 선형이나 비선형 시스템에 적용 가능한 최적제어 이론의 응용이 불가피하다. 그러나 최적제어 이론의 응용은 실제적으로 어려움이 있어 제약 조건을 무시하거나 동력학적 모델을 축소하는 등의 간략화가 이루어지고 있다.

본 논문은 완벽한 로봇 매니퓰레이터의 동역학 모델과 모든 제약 조건을 고려하여 공간 상의 주어진 경로에서 최소의 에너지를 소비하는 최적 속도 분포를 구하는 궤도 계획의 문제를 다루었다.

본 분석을 위하여 엄격한 제약 조건을 만족하면서 복잡하고 고도로 비선형인 시스템에 최적화시키기 위해 적합하다고 알려진 Dynamic Programming(D.P)을 이용하였다. 이에 따라 로봇 메카니즘의 모델과 구동기의 모델을 포함시킨 로봇 시스템의 완전한 동역학 모델에 직접적으로 D.P기법을 적용하여 공간 상의 주어진 초기 위치와 최종 위치 간을 선형적으로 이동하며 최소의 에너지를 소비하도록 하는 속도 분포에 관해 연구했다.

2. 본 론

2.1 매니퓰레이터 시스템의 동역학 모델

로봇 매니퓰레이터 시스템의 동역학 모델은 메카니즘 부분의 모델과 매니퓰레이터의 각 조인트를 구동하는 구동기의 모델로 구성된다. 본 논문에서는 improved Newton-Euler 방식의 운동 방정식과 2차 선형 모델로 설계된 DC 모터를 이용하여 완전한 동역학 모델을 얻었다.

2.1.1 Newton-Euler 운동 방정식

n 자유도의 로봇 매니퓰레이터에 대한 운동 방정식은 조인트 공간에서 다음과 같이 주어진다.

$$T = M(q)\ddot{q} + V(q, \dot{q}) + G(q) \quad \dots \dots \dots (1)$$

M(q) : n x n inertial matrix of the manipulator
V(q, q̇) : n x 1 vector of centrifugal and coriolis term
G(q) : n x 1 vector of gravity

2.1.2 DC 모터의 모델

산업용 로봇의 구동기로 널리 이용되는 것이 DC 모터이다. DC 모터가 2차 선형 모델로 설계되면 i번째 조인트의 구동기 모델은 다음의 형태로 고려할 수 있다.

$$\dot{x}_i = C_i x_i + f_i P_i + d_i u_i \quad (i=1, 2, \dots, n) \quad \dots (2)$$

$x_i = [q_i \ \dot{q}_i]^T$ i번째 구동기의 상태 벡터
 $u_i \in R^1$ i번째 구동기의 입력 전압
 $T_i \in R^1$ i번째 구동기에서 얻어지는 구동 토크
 $C_i \in R^{2 \times 2}$, $d_i \in R^2$, $f_i \in R^1$: 구동기 시스템 행렬
 $C_i = [a_{11} \ a_{12}]$ $a_{11}=0$, $a_{12}=1$, $a_{21}=0$, $a_{22}=\alpha$
 $d_i = [0 \ \beta]^T$
 $f_i = [0 \ \gamma]^T$

$$\alpha = -\frac{R_i}{J_i} - \frac{K_{Emi} K_{Mei} N_{1i} N_{2i}}{R_i J_i} = -\frac{C_{ei} C_{mi}}{J_i R_i}$$

$$\beta = \frac{K_{Emi} N_{2i}}{J_i R_i} \quad \gamma = -\frac{1}{J_i}$$

- C_m 모멘트 상수
- C_e Electromotor force
- J_i i번째 조인트의 모터 관성 모멘트
- F_i 마찰력(friction)
- R_i rotor 저항
- K_{Emi} electromechanical 상수
- K_{Mei} mechanolectrical 상수
- N_{1i} 속도 감소비(speed reduction ratio)
- N_{2i} 모멘트 감소비(moment reduction ratio)

2.1.3 완전한 동역학 모델

앞에서 논의한 메카니즘 부분의 상태 벡터에 구동기의 상태 좌표(coordinates)가 포함되고 벡터의 요소(element)가 일치할 경우 즉, i번째 조인트 구동기의 상태 벡터, xi에 일반화 좌표(generalized coordinate), qi와 일반화 속도(generalized velocity) q̇i가 포함될 경우에는 메카니즘 부분의 모델과 구동기의 모델을 포함하는 하나의 완전한 매니퓰레이터 시스템의 동역학 모델을 구할 수 있다. 앞 절의 (1), (2)식을 이용하여 동역학 모델을 구하는 과정은 다음과 같다.
먼저, (1)식의 q̇i와 (2)식의 상태 벡터, xi간의 변환 행렬(transformation matrix)을 도입한다.

$$\begin{aligned} \dot{q}_i &= T_i \dot{x}_i \quad (T_i : i = 1, 2, \dots, n) \dots\dots (3) \\ T &= \text{diag}[T_1, T_2, \dots, T_n] \end{aligned}$$

그리고 (3)식을 (1)식에 대입한다.

$$\begin{aligned} T &= M(q)\ddot{q} + V(q, \dot{q}) + g(q) \\ &= M(q)T\ddot{x} + V(q, \dot{q}) + G(q) \dots\dots (4) \end{aligned}$$

또, (2)식을 (4)식에 대입하면

$$\begin{aligned} T &= M(q)T'(Cx + Ft + Du) + V(q, \dot{q}) + G(q) \\ &= (En - MT') [MT'(Cx + Du) + V + G] \dots\dots (5) \\ &\quad ; En : n \times n \text{의 단위 행렬} \end{aligned}$$

최종적으로 (5)식을 (2)식에 대입하면 다음의 완전한 모델을 얻을 수 있다.

$$\begin{aligned} \ddot{x}_i &= Cx + F(En - MT') [MT'(Cx + Du) + V + G] + Du \\ &= Cx + F(En - MT') [MTCx + V + G] + \\ &\quad (D + F(En - MT') MTD)u \end{aligned}$$

2.2 D.P를 이용한 에너지 최소화 문제의 분석

본 절에서는 복잡하고 비선형 다변수 시스템에 속하는 모보트 매니퓰레이터의 완전한 동역학 모델과 모든 제약 조건을 고려하여 공간 상의 주어진 초기 위치와 최종 위치를 직선 경로로 이동하면서 최소의 에너지를 소비하는 최적 속도 분포를 D.P기법을 응용하여 구하는 궤도 계획의 문제를 다루도록 하겠다. D.P기법을 적용하기 전에 다음의 과정이 필요하다.

2.2.1 동역학 모델의 이산화.

주어진 이동 시간 t (0 < t < T)를 균등한 N개의 시간 간격으로 나누어 각 구간의 미소 시간을 Δt라 하자. Δt가 아주 작다면 제어 신호를 소구간 상수 함수(piecewise-constant function)로 근사할 수 있다.

$$\begin{aligned} t &= 0, \Delta t, 2\Delta t, \dots, (N-1)\Delta t \\ t &= k\Delta t \quad (k=0, 1, \dots, (N-2), (N-1)) \dots\dots (6) \end{aligned}$$

앞 절에서 한 n 자유도를 가진 모보트 매니퓰레이터의 동역학 모델을 이산화하여 차등 방정식의 형태로 나타내는 과정은 다음과 같다.

$$\begin{aligned} \dot{x}(t) &= \hat{A} + \hat{B}u(t) \\ \frac{x(t+\Delta t) - x(t)}{\Delta t} &= \hat{A}(x(t)) + \hat{B}(x(t))u(t) \end{aligned}$$

또는, $x(t+\Delta t) = x(t) + \Delta t(\hat{A}(x(t)) + \hat{B}(x(t))u(t))$
(6)식에 따라 $x(k+1) = x(k) + \Delta t(\hat{A}(k)) + \hat{B}(x(k))u(k) \dots\dots (7)$

2.2.2 성능 지수의 이산화

일반적인 형태의 성능 지수는 다음과 같다 $J = h(x(t_f)) + \int_{t_0}^{t_f} g(x(t), u(t)) dt \dots\dots (8)$

(2.2.1)과 유사한 방법으로 (8)식을 근사시키면 $J = h(x(N \Delta t)) + \sum_{k=0}^{N-2} \int_{k\Delta t}^{(k+1)\Delta t} g dt + \dots + \int_{(N-1)\Delta t}^{N\Delta t} g dt$
 $= h(x(N)) + \Delta t \sum_{k=0}^{N-2} g(x(k), u(k)) \dots\dots (9)$

(8)식이 구동기의 소비 에너지에 대한 성능 지수로서 고려될

경우는 Joule의 효과에 기인한 전체 낭비(dissipate) 전력과 구동기에서 메카니즘에 전달되는 전체 전력의 합으로 구성된다. 따라서, 성능 지수는 다음과 같이 표현된다.

$$J = \sum_{t_0}^{t_f} \int_{t_0}^{t_f} u_1 u_1 dt = \int_{t_0}^{t_f} (u^T Q_1 u + x^T Q_2 u) dt \dots\dots (10)$$

- n : 자유도 수
- Ii : rotor 전류
- Ri : i번째 모터의 rotor 저항
- $Q_1 \in R^{n \times n} = \text{diag}((1/R, 1/R, \dots, 1/R))$
- $Q_2 \in R^{1 \times n} = q_{21} \dots q_{2n} ; q_{2i} = -K_{MEi} N_{Vi} / R_i$
- K_{MEi} : i번째 모터의 mechanical-electrical상수
- N_{Vi} : 각 감속비

(10)식도 위와 같은 방법으로 이산화된다. 즉 $J = \int_0^{\Delta t} (u^T Q_1 u + x^T Q_2 u) dt + \int_{\Delta t}^{2\Delta t} (u^T Q_1 u + x^T Q_2 u) dt + \dots$
 $= \Delta t \sum_{k=0}^{N-2} (u^T(k) Q_1 u(k) + x^T(k) Q_2 u(k))$

2.2.3 제약 조건의 양자화

D.P기법에서 일반적인 제어 변수와 상태 변수의 양자화는 다음 (11)식의 부등 제약 조건식에서 전체 허용 구간 내의 값들을 일정한 간격으로 양자화하여 격자 점(grid point)을 구하였다.

$$\begin{aligned} V &< u(t) < W \\ R &< x(t) < S \end{aligned} \dots\dots (11)$$

그러나 이와 같은 양자화는 격자 점간의 간격이 좁아야 보다 정확한 최적 제어 가능하지만 그에 따른 메모리의 증가와 계산량이 많아지는 단점이 생긴다. 따라서 이와 같은 방법과 다르게 격자 점을 구해보면 다음과 같다.

n자유도를 가진 매니퓰레이터의 end effector가 카테시안 공간상의 초기 위치 Xⁱ와 최종 위치 X^f간을 정해진 경로로 이동한다 하자. 이동 경로는 다음의 파라미터 방정식으로 구한다.

$$X_C = R(\lambda(t)) \dots\dots (12)$$

- X_C : 카테시안 좌표 벡터
- R : 주어진 벡터 함수
- [0,1]인 스칼라 파라미터
- $R(0) = X_C^i$
- $R(1) = X_C^f$

(12)식은 작업 공간에서 주어진 초기위치와 최종위치간의 궤도 함수로서 주어진 위치 간을 직선 궤도로 이동할 경우 (11)식은 다음과 같이 표현된다.

$$X_C(t) = X_C^i + \lambda(t) (X_C^f - X_C^i) \dots\dots (13)$$

따라서, λ(t)값에 따라 카테시안 공간 상의 위치가 결정 된다. 일반적으로 초기 분포(initial distribution)값을 정의하는 것이 보다 정밀한 제어 및 메모리나 계산상의 효율성등 여러 잇점을 얻을 수 있다. 주로 0과 1간의 선형 함수를 정의하는 것이 가장 바람직 하다.

또한, (13)식의 시간 t는 (6)식에 따라 다음으로 이산화 된다.

$$X_C^j(k) = X_C^i + \lambda(k) (X_C^f - X_C^i) \dots\dots (14)$$

(14)식에서, λ(k)의 임의의 k값에 따른 λ(k)값을 기준으로 하여 고정된 증분 Δλ를 더하거나 빼어서 여러 λ(k)값을 구한다.

즉, $\lambda(k) = \lambda(k) + (i - ((M+1)/2)) \quad i = 1, 2, \dots, M \dots\dots (15)$

컴퓨터 해석에 의하면 M이 7과 10 사이 값이 선택될 때가 효율적이다. (15)식에서 구한 값들을 (14)식에 대입하면 X_C^j(k), i=1, ..., M을 얻을 수 있다.

따라서, 임의의 단계를 N-라 할때 카테시안 공간 상의 좌표 벡터, X_C^j(N-K), i=1, ..., M를 구할 수 있고, 그와 일치하는 모보트 매니퓰레이터의 조인트 공간 상의 좌표 벡터 q^j(N-K)는 다음과 같이 inverse kinematics식에 의해 구해진다.

$$q^j(N-K) = f^{-1}(X_C^j(N-K)) \dots\dots (16)$$

결국 (16)식에 따라 모든 단계에 대한 조인트 공간의 좌표 벡터를 구하면 n자유도를 가진 모보트 매니퓰레이터의 상태 벡터는 다음의 계산 절차로 구할 수 있다.

위에서 구한 q^j(N-K), i=1, ..., M과 q^j(N-K+1), j=1, ..., M의 각각에 대해 다음과 같이 조인트 속도가 결정된다.

$$\begin{aligned} \dot{q}^m(N-K) &= (q^j(N-K+1) - q^j(N-K)) / t \\ i &= 1, \dots, M, \quad (j = 1, \dots, M) \\ m &= (i-1)M + j \end{aligned}$$

따라서 상태 벡터는,

$$x^{m(N-K)} = [q^{T(N-K)} \quad q^{mT(N-K)}] \quad (m = 1, \dots, M)$$
 결론적으로, 모든 단계에서의 상태 벡터 값들이 구해졌고, 그에 따라 이들 값을 (7)식의 동력학 모델에서 구한 제어 벡터 $u(k)$ 에 대입하여 각각의 상태 벡터에 대한 제어 벡터를 구해 낸다.

$$u^{mK}(K) = [\hat{B}(x^m(K))\hat{B}(x^m(K))]^{-1} \hat{B}^T(x^m(K)) [x^E(K+1) - x^m(K)] / \Delta t - \hat{A}(x^m(K)) \dots \dots \dots (17)$$

즉, 일반적인 D.P 기법의 K단계에 대한 $u(k)$ 와 $X(k)$ 의 양자와 된 값을 (7)식에 대입, $k+1$ 단계의 $X(k+1)$ 을 구하는 대신, 이미 알고 있는 각각의 $X(k)$ 와 $X(k+1)$ 를 차례로 (17)식에 대입, 그에 따른 모든 제어 벡터를 구해낸다.

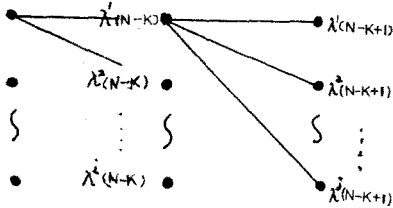


그림 1

단, 그림 1 과 같이 제어 벡터 $u(k)$ 는 $X(N-J)$ 상태를 구성한 $(N-J)$ 값이 $X(N-J+1)$ 상태를 구성한 $(N-J+1)$ 값과 대응 될 경우에만 구할 수 있다. 그러므로, 첨자 1과 m 은 다음과 같다.
 $l = (m^* - 1)M + 1, \dots, m^*M$
 $m^* = \lceil (m-1)/M \rceil$

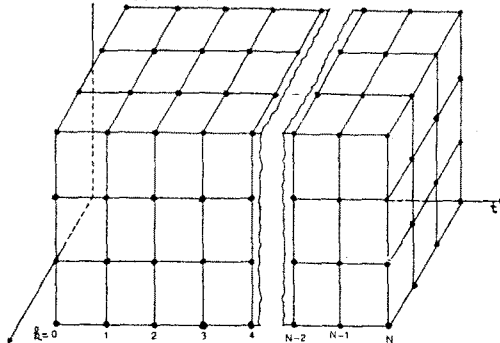


그림 2 상태 벡터와 제어 벡터의 격자점

2.3 D.P기법의 적용

(2.2.1 - 3)에서 제시된 내용을 근거하여 본절에서는 각각의 미소 시간에서의 최적 궤도(trjectory) $x^1, x^2(1), \dots, x^F(N-1), x^F$ 를 따라 시스템을 이동시키는 최적 제어 정책(Optimal control policy) $u(x^1(0), x^2(1), 0), u(x^2(1), x^2(2), 1), \dots, u(x^F(N-2), x^F(N-3), N-2), u(x^F(N-1), x^F(N), N-1)$ 을 다음과 같은 계산 절차에 따라 구해 보도록 하겠다.

우선 최종 상태 벡터 $x^F(N)$ 에 이르는 비용(cost)을 J_{NN} 이라 정의하자.

$$J_{NN}(x(N)) = h(x^F(N))$$

$K = N-1$ 로 놓고 상태 벡터 $x^{m(N-1)}$ 와 $x^F(N)$ 을 2.2.3에서 논한 방식으로 대응, 그에 따른 제어 벡터 $u^{m(N-1)}$ 을 구한다. 그중 최소 비용을 갖는 제어 벡터가 대응된 상태 벡터에 대한 최적 제어 정책이 된다.

즉, 일 단계의 초기 상태 $x^m(N-1), m=1, \dots, M$ 을 갖는 $(N-1)\Delta t \leq t \leq N\Delta t$ 구간의 비용은 다음과 같이 정의 한다.

$$J_{N-1, N}(x^m(N-1), u^{m(N-1)}) = g_4(x^m(N-1), u^{m(N-1)}) + h(x^F(N)) = g_4(x^m(N-1), u^{m(N-1)}) + J_{NN}(x(N))$$

최종 상태 벡터는 입력으로 주어지기 때문에 모든 $x(N)$ 상태 벡터는 한 값으로 정의된다. 즉,
 $x^F(N) \triangleq x^m(N) \triangleq x(N)$

따라서, 일 단계의 최소 비용은 다음과 같다.

$$J_{N-1, N}^*(x^m(N-1)) = \min_{u^{m(N-1)}} \{g_4(x^m(N-1), u^{m(N-1)}) + J_{NN}(x(N))\} \dots \dots \dots (18)$$

$u^{m(N-1)}$ 의 제어 벡터는 (2.2.3)의 (17)식에 의해 $x^{m(N-1)}$ 과 $x(N)$ 에 따라 결정되므로 (18)식에서 구해지는 최적 제어 벡터는 $u^*(x^m(N-1), x^F(N), N-1)$ 로 정의한다.
 $k = N-2$ 일때인 이 단계의 과정에서 $(N-2)\Delta t \leq t \leq N\Delta t$ 구간의 비용은 다음과 같다.

$$J_{N-2, N}(x^m(N-2), u^{m(N-2)}, u^{m(N-1)}) = g_4(x^m(N-2), u^{m(N-2)}) + g_4(x^m(N-1), u^{m(N-1)}) + J_{N,N}(x(N)) = g_4(x^m(N-2), u^{m(N-2)}) + J_{N-1, N}(x^m(N-1), u^{m(N-1)})$$

따라서, 마지막 두 구간 동안의 최적 제어 정책은 다음 식으로 구해진다.

$$J_{N-2, N}^*(x^m(N-2)) = \min_{u^{m(N-2)}, u^{m(N-1)}} \{g_4(x^m(N-2), u^{m(N-2)}(N-2)) + J_{N-1, N}(x^m(N-1), u^{m(N-1)})\} \dots (19)$$

(19)식의 이 단계 과정에 대해 최적성 원리는 이 단계의 초기 상태 벡터 $x(N-2)$ 와 초기 제어 벡터 $u(N-2)$ 가 무엇이든간에 남아 있는 제어 벡터 $u(N-1)$ 은 $x(N-2)$ 의 적용에 따른 결과인 $x(N-1)$ 에 따라 최적화가 이루어져야 함을 말해 준다. 결국, 최적성 원리에 의해 마지막 두 구간의 최소 비용은 다음과 같다.

$$J_{N-2, N}^*(x^m(N-2)) = \min_{u^{m(N-2)}} \{g_4(x^m(N-2), u^{m(N-2)}) + J_{N-1, N}^*(x^m(N-1))\}$$

이와같은 방법으로 임의의 초기 상태, $x^m(N-K)$ 를 갖고 K 단계 과정인 마지막 K구간의 최소 비용은 다음과 같이 결정된다.

$$J_{N-K, N}^*(x^m(N-K)) = \min_{u^{m(N-K)}, \dots, u^{m(N-1)}} \{h(x(N)) + \sum_{k=N-K}^{N-1} g_4(x(N), u(k))\}$$

즉 (K-1)단계 정책에 대한 최소 비용 $J_{N-K+1, N}^*$ 을 알면 임의의 K 단계 정책에 대한 최소 비용 $J_{N-K, N}^*$ 을 구할 수 있음을 보여 준다.

N 단계 최적화 과정에서 시스템의 초기 상태는 입력으로 주어진다. 그러나, 최종 상태와는 달리 (2.2.1)에서 논의된 것처럼 앞 단계의 조인트 위치 값과의 관계 때문에 각각의 초기 조인트 위치의 벡터 요소는 모두 일치하지만 조인트 속도는 모두 일치하는 것은 아니다. 그러나, 초기 속도와 최종 속도는 모두 0 이므로 일단은 각각의 조인트 속도 모두 0의 상태에서 출발한다. 결국, 전체 이동 시간 $[t^1, t^F]$ 동안 최소 비용을 얻을 수 있는 최적 제어 벡터 $u(x(0), 0)$ 을 얻을 수 있다.

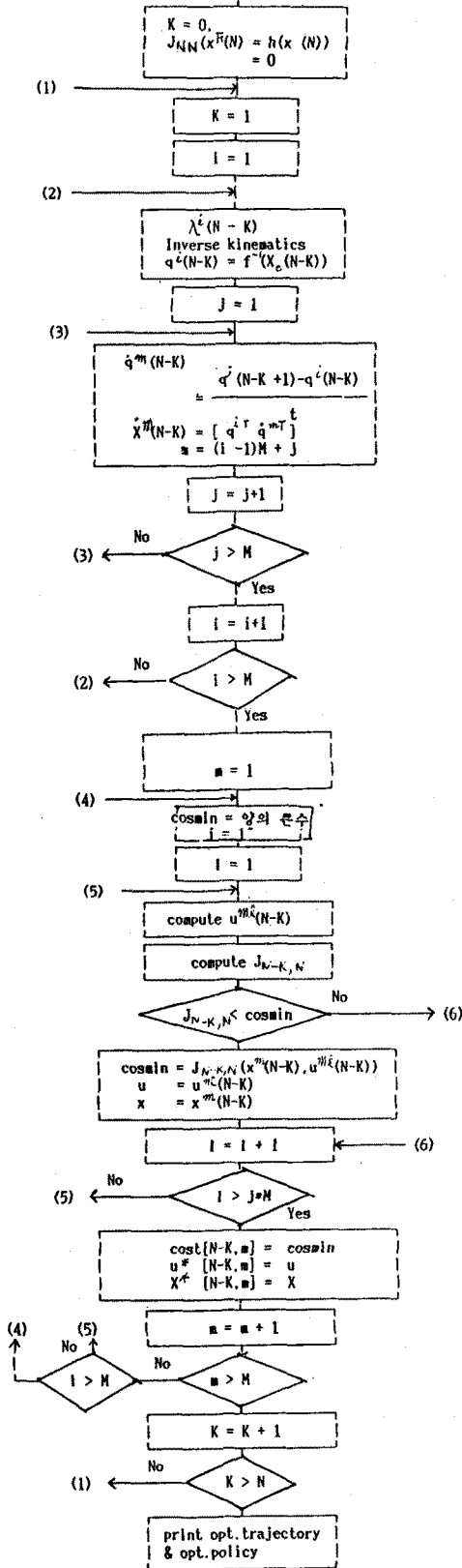
이상과 같이 순환(recurrence) 방식으로 일 단계에서부터 N 단계까지 구한 최소 비용과 최적 제어 정책을 가지고 최적 궤도(optimal trajectory)와 최적 제어 정책은 뒤로 부터 재구성한다. 즉, 시스템의 초기 상태 $x^1(0)$ 로부터 N 단계에서 구한 최적 제어 벡터중 한 값이 선택되고 선택된 $u^*(x^1(0), x^1(1), 0)$ 를 구성하는 $x^{m^*}(1)$ 으로부터 $u^{m^*}(x^{m^*}(1), x^{m^*}(2), 1)$ 이 구해진다.

여기서 m^* 과 1은 앞의 순환방식으로 구한 M 개의 최적 제어 벡터와 최적 상태 벡터 중 선택된 수치이다. 이와 같은 방법으로 계속 구해 나가면 $x^E(N-1)$ 에 따른 $u^{m^*}(x^{m^*}(N-1), x^{m^*}(N), N-1)$ 까지 결정된다.

따라서, 전체 이동 시간에서의 최적 제어 정책과 최적 궤도 가 구해진다.

2.4 최적 속도 분포 계산을 위한 알고리즘

입력 데이터 :	
초기 상태 벡터	= x^1
최종 상태 벡터	= x^F
미소 시간	= Δt
단계 수(Number of Stage)	= N
스칼라 증분	= $\Delta \lambda$
벡터 함수(vectorial function)	= R
매니퓰레이터와 구동기의 입력 데이터	



3. 컴퓨터 해석 및 고찰

본 논문은 DC 모터로 구동되는 n 자유도의 매니퓰레이터가 공간상의 정해진 위치 간을 최소의 에너지를 소비하며 이동하는 문제에 대하여 연구하였다.

또한, 개발한 Dynamic Programming 알고리즘을 다음의 DC모터로 구동되는 2-link 매니퓰레이터에 적용하여 에너지 최소화 문제를 고찰하였다.

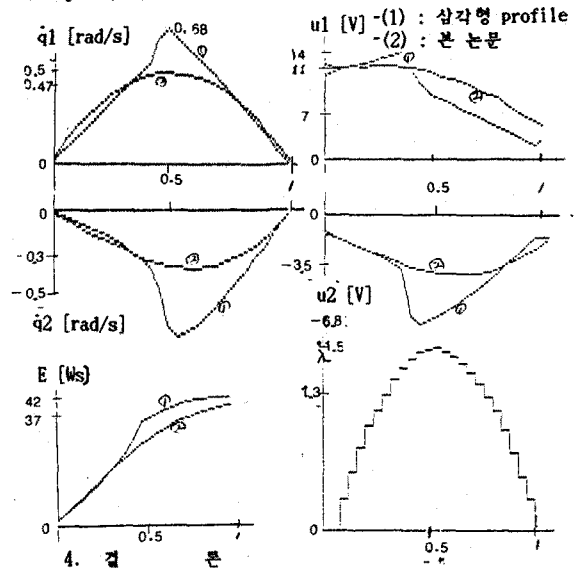
(2.1.1)에서 논의한 DC 모터의 α, β, γ 는 $-10.8, 1.42, -0.66$ 이고 각 링크의 말단에 모든 질량이 존재하는, 각 링크의 무게가 1 [Kg]이고 링크 길이가 1 [m]인 매니퓰레이터를 선정하였다. 매니퓰레이터의 운동 방정식은 Improved Newton-Euler 방식을 이용하였고, 매니퓰레이터의 이동 경로는 다음으로 주었다.

$$x_c = x_i + \lambda(x_f - x_i)$$

$$x_i^1 = [0.75 \ 1.25]^T, \quad x_f^1 = [0.5 \ 1.5]^T$$

또한, 초기 위치에서 최종 위치로의 이동 시간은 1초로 가정하였고 단계(stage) 수를 25로 잡아서 이산화된 각 구간의 시간은 0.04초로 정하였다.

위의 데이터를 이용하여 본 논문에서 개발한 내용과 삼각형(triangular) 속도 프로파일과 비교한 결과는 다음과 같다.



본 논문은 모보트 매니퓰레이터의 완전한 동역학 모델을 고려하여 주어진 초기 위치와 최종 위치 간을 최소의 에너지로 이동하도록 하는 궤도 계획의 문제를 Dynamic Programming 기법을 이용하여 연구하였다.

본 논문에서 개발한 알고리즘은 실제적으로 주어진 업무에 대한 off-line 궤도 계획에 이용할 수 있고 또한, 고속 작업을 요하고 큰 부하를 다루는 모보트 매니퓰레이터에 대해 에너지 소비를 최적화 할 뿐만 아니라 구동기와 메카니즘 자체에 무리가 가는 것을 최소화해 줌으로서 모보트의 효율적 이용과 아울러 생산성 향상이 기대된다.

4. REFERENCE

[1] Bellman, R., Dynamic Programming, Princeton University Press, 1957.
 [2] Bellman, R., Applied Dynamic Programming, Princeton University Press, 1962.
 [3] Bellman, R. & Kalaba, Dynamic Programming and Modern Control Theory, Academic Press NewYork and London, 1965.
 [4] Vukobratovic, M. & Kiricanski, M., "A Method for the Optimal Synthesis of Manipulation Robot Trajectories", Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME, Vol. 104, June 1982, pp.188 - 193.
 [5] Singh, S. & Leu, M.C., "Optimal Trajectory Planning and Control of Robotic Manipulators", Japan - U.S.A. Symposium on Flexible Automation, 1986, pp.169 - 177.
 [6] Donald E. Kirk, Optimal Control Theory An Introduction, Prentice Hall, INC. Englewood Cliffs, New Jersey, 1970, pp.53 - 98.