

# 매개변수를 이용한 Labview<sup>®</sup>기반의 3축 SCARA로봇의 이종모션 제어 알고리즘

정원지<sup>+</sup>, 주지훈\*, 이기상<sup>++</sup>

(논문접수일 2008. 9. 22, 심사완료일 2009. 1. 16)

## Hybrid Motion Blending Algorithm of 3-Axis SCARA Robot based on Labview<sup>®</sup> using Parametric Interpolation

Won-Jee Chung<sup>+</sup>, Ji-Hun Ju\*, Kee Sang Lee<sup>++</sup>

### Abstract

In order to implement continuous-path motion on a robot, it is necessary to blend one joint motion to another joint motion near a via point in a trapezoidal form of joint velocity. First, the velocity superposition using parametric interpolation is proposed. Hybrid motion blending is defined as the blending of different two type's motions such as blending of joint motion with linear motion, in the neighborhood of a via point. Second, hybrid motion blending algorithm is proposed based on velocity superposition using parametric interpolation. By using a 3-axis SCARA (Selective Compliance Assembly Robot Arm) robot with LabVIEW<sup>®</sup> controller<sup>(1)</sup>, the velocity superposition algorithm using parametric interpolation is shown to result in less vibration, compared with PTP(Point-To-Point) motion and Kim's algorithm<sup>(2)</sup>. Moreover, the hybrid motion algorithm is implemented on the robot using LabVIEW<sup>®</sup><sup>(1)</sup> programming, which is confirmed by showing the end-effector path of joint-linear hybrid motion.

**Key Words** : Velocity superposition(속도 중첩), parametric interpolation(매개변수 보간), SCARA robot(스카라 로봇), Hybrid motion blending(이종모션 블렌딩), Continuous Path(연속 관절 운동)

\* 교신저자, 창원대학교 기계설계학과 (jujh87@hanmail.net)

주소: 641-773 경남 창원시 사림동 9번지

+ 창원대 기계설계공학과

++ (주)아진에스텍

### 1. 서론

이 논문의 목적은 산업용 로봇의 고속 핸들링을 위한 연속 관절 운동에 신속하고 부드러운 관절 연속운동을 위한 모션블렌딩 알고리즘을 개발하는 것이다. 현대 로봇의 이동방식은 크게 PTP(Point to Point)모션과 연속 관절 운동CP(Continuous Path)모션으로 나눌 수 있다. 초기의 산업용 로봇은 대량 생산 라인에서 단순 반복 작업을 위주로 하였기 때문에 PTP모션만으로 충분하였다. 그러나 현대에는 단순 반복 작업용 로봇, 용접용 로봇뿐만 아니라 공작기계에 결합하여 다양한 기능을 수행할 수 있는 산업용 로봇을 필요로 하고 있다.

따라서 앞으로 다양한 분야의 로봇 수요 증가에 대응하기 위해서는 고속 핸들링을 위한 연속 관절 운동(CP)을 필수적으로 구현할 수 있어야 한다. 연속 관절 운동 구현 시 부드럽고 신속한 핸들링이 가능하며, 소음 및 진동 감소 또한 가능하다.

기본적인 로봇의 CP 동작은 Fig. 1와 같이 시작점에서 경유점, 경유점에서 목표점까지의 각각의 사다리꼴 속도 선도를 중첩하여 구현한다. 관절보간 동작을 블렌딩 하는 방법은 현재 이동 명령 신호의 속도 선도와 다음 명령 신호의 속도 선도를 시간 축을 중심으로 중첩을 시켜 현재 명령 신호의 경유점 부근에서 부드럽게 블렌딩 시키는 것이다.

CP 구현으로 인해 이동시간 단축 및 진동 저감의 효과를 얻을 수 있다. 기존의 CP방식은 단순 속도 중첩을 이용한 것으로, 여러 가지 속도 중첩의 경우를 고려하여 알고리즘이 만들어졌기 때문에 식의 수가 많으며 로봇의 실제 구현시 어려운 점이 있다. 따라서, 본 논문은 매개변수  $u(t)$ 를 이용하여 보다 쉽게 CP를 구현 할 수 있는 방법을 제안한다.

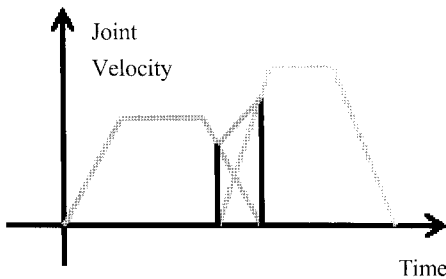


Fig. 1 Superposition of velocity

### 2. 매개변수를 이용한 속도 중첩

이종모션 블렌딩을 하기 위해서는 먼저 매개변수를 이용하여 동일한 가/감속 시간을 가지는 속도선도를 만들어야 한다. 이 속도 선도를 이용하여 속도 중첩을 통해 모션 블렌딩이 구현되며, 매개변수를 이용하여 이종 모션 블렌딩을 구현할 수 있다.

각 축의 이동 량이 서로 다른 경우에 매개변수를 이용하여 동일한 속도 선도를 모든 축에 적용하기 위해서는 동일한 가/감속시간과 등속시간을 가져야 하는 제약조건이 따른다. 이 제약 조건을 만족시킬 수 있도록 가/감속 시간 및 등속시간을 동일하게 하는 계산 방법을 설명하면 다음과 같다.

우선 식 (1)을 이용하여 각 축이 가/감속 구간 없이 지정된 속도로 이동할 때 걸리는 시간을 계산한다. 그 방법은 먼저 속도 선도를 가속시간, 감속시간, 등속시간을 동일하게 하여 시작점이 1이고 목표점이 0인 매개변수  $u$ 를 도입하여 모든 축에 대하여 동일한 매개변수를 이용한 속도 선도를 구한다.

$$t_i = \frac{|P_c^i - P_s^i|}{V_{max}^i \times \text{speedfactor}} \quad (\text{speedfactor} \in [0,1]) \quad (1)$$

여기서  $i$ 는 관절의 수,  $V_{max}^i$ 는  $i$ 번째 관절이 낼 수 있는 최대 이동속력,  $P_c^i$ 는 종점,  $P_s^i$ 는 시점을 나타낸다. 그리고 Speedfactor는 프로그램에 의해 지정되어 0에서 1까지의 값을 갖는 변수이다. 예를 들어 이동속도가 최대속도의 70%가 되도록 프로그램 되었다면 Speedfactor는 0.7이 된다. (Speedfactor는 실제 로봇제어 시 모터 설정 값을 따른다.)

$$t_a = \text{MAX} \left( \frac{|P_c^i - P_s^i|}{t \cdot (a_{max}^i \times \text{accelfactor})} \right) \quad (2)$$

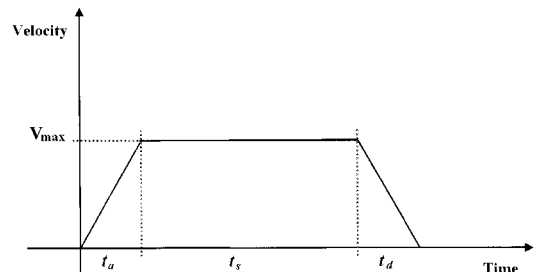


Fig. 2 Trapezoidal velocity 선도

$$t_d = MAX\left(\frac{|P_e^i - P_s^i|}{tt \cdot (d^i_{max} \times \text{decelfactor})}\right) \quad (3)$$

$$t_s = tt - \frac{(t_a + t_d)}{2} \quad (4)$$

여기서  $a^i_{max}$ 는  $i$ 번째 관절이 낼 수 있는 최대 가속도이고,  $d^i_{max}$ 는  $i$ 번째 관절이 낼 수 있는 최대 감속도이다.  $tt$ 는 전체 이동 거리를  $V_{max}$ 로 이동한 전체이동시간이다. **AccelFactor**는 프로그램에 의해 지정되는 값으로서 0에서 1의 범위를 갖는다.

전체 총 이동시간(  $t_{total}$  )는 다음과 같이 주어진다.

$$t_{total} = t_a + t_s + t_d \quad (5)$$

만약  $tt < \frac{(t_a + t_d)}{2}$ ,  $t_s$ 이면 위 식으로부터  $t_s$ 는 음수를 가지게 되므로 이런 경우에는  $t_s$ 를 0으로 설정하고 속도 선도를 삼각형으로 변경한다.  $t_s$ 를 0으로 설정하였으므로 위 식에서 계산한  $t_a$ ,  $t_d$ 도 아래와 같이 정의 할 수 있다.

$$t_a + t_d = 2tt \quad (6)$$

$a^i_{max} \times \text{accelfactor} = a^i$  라 가정하면 식 (2)를 다음과 같이 나타낼 수 있다.

$$t_a = \frac{|P_e^i - P_s^i|}{tt \cdot a^i} \quad (7)$$

여기서  $i$ 는 가속시간이 가장 긴 축을 나타낸다. 또한 감속 구간 역시  $d^i_{max} \times \text{decelfactor} = d^i$  라 가정할 경우, 아래 식과 같이 나타 낼 수 있다.

$$t_d = \frac{|P_e^i - P_s^i|}{tt \cdot d^i} \quad (8)$$

여기서  $j$ 는 감속시간이 가장 긴축을 나타낸다. 그리고 식 (6), (7)과 (8)을 이용하여 아래 식을 구 할 수 있다.

$$t_a = \frac{2 \cdot |P_e^i - P_s^i|}{\sqrt{2a^i (|P_e^i - P_s^i| d^j + |P_e^j - P_s^j| a^i)}} \quad (9)$$

$$t_d = \frac{2 \cdot |P_e^j - P_s^j|}{\sqrt{2d^j (|P_e^i - P_s^i| d^j + |P_e^j - P_s^j| a^i)}} \quad (10)$$

식 (9), (10)을 이용하여 모든 축의 가/감속시간을 가장 긴 시간으로 동일하게 할 수 있다. 가장 긴 시간으로 정의 하는 이유는 짧은 시간에 맞출 경우 한계 가/감속도를 초과 할 수 있기 때문이다.

동일한 가/감속시간의 속도 선도를 구하면 매개변수를 이용하여 속도 프로파일 기본 식을 다음과 같이 정의 할 수 있다.

$$P^i(u(t)) = P_e^i - u(t)(P_e^i - P_s^i), \quad u \in [0,1] \quad (11)$$

(여기서,  $i$ : 축 번호,)

여기서  $u(t)$ 는 매개변수로 가/감속 속도 중점의 핵심적인 역할을 한다. 매개변수  $u(t)$ 는 식 (12)와 같이 정의 되며, 시점이 1이 되고 중점이 0이 되어 전체 이동거리에 대한 이동한 이동거리 비를 뜻하며, 이를 이용하여 모든 축에 대하여 동일한 속도 선도를 구할 수 있다.

$$u(t) = 1 - \frac{L}{l} \quad (12)$$

여기서  $L$ 은 Fig. 3에서 보듯이 전체 이동 거리이며,  $l$ 은 이동한 이동거리이다.

전체 이동거리는 간략하게 아래와 같이 나타낼 수 있다.

$$L = \frac{1}{2}v \cdot t_a + v \cdot t_s + \frac{1}{2}v \cdot t_d \quad (13)$$

여기서  $v$ 는 등속 구간의 속도이며, 매개변수  $u(t)$ 는 아래 가속 구간, 등속 구간 그리고 감속 구간으로 나뉘진다.

Case 1:  $0 \leq t < t_a$

$$u(t) = 1 - \frac{t^2}{t_a(t_a + 2t_s + t_d)} \quad (14)$$

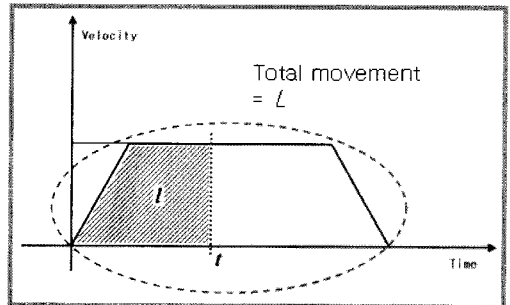


Fig. 3 Meaning of  $l$  and  $L$

Case 2:  $t_a \leq t < t_{total} - t_d$

$$u(t) = 1 - \frac{2t - t_a}{(t_a + 2t_s + t_d)} \quad (15)$$

Case 3:  $t_{total} - t_d \leq t \leq t_{total}$

$$u(t) = 1 - \frac{(t_{total} - t)^2}{t_d(t_a + 2t_s + t_d)} \quad (16)$$

식 (14), (15) 그리고 (16)을 이용하여 이동시간 및 이동거리를 구할 수 있다.

그리고 매개변수를 이용하여 각 구간별 속도 선도를 구할 수 있다.

①  $0 \leq t < t_{bs}$

$$P^i(u_1(t)) = P_v^i - u_1(t)(P_v^i - P_f^i) \quad (17)$$

②  $t_{bs} \leq t < t_{be}$

$$P^i(u_1(t_1), u_2(t_2)) = P_f^i - u_1(t_1)(P_f^i - P_c^i) - u_2(t_2)(P_c^i - P_v^i) \quad (18)$$

③  $t_{be} \leq t \leq t_{total}$

$$P^i(u_2(t)) = P_v^i - u_2(t)(P_v^i - P_c^i) \quad (19)$$

여기서  $P_v^i$  는 경유점 (via point)를 의미 한다. 위 식 (17), (18) 그리고 (19)는 각 각 Fig. 4 의 ①, ② 그리고 ③에 대응 된다. 시점에서 속도 중첩 전까지, 속도 중첩 구간 그리고 속도 중첩 후에 각 각 대응됨을 의미한다.

기존의 연구 되어진 조건을 고려한 속도 중첩 방식<sup>(1)</sup>과 제안한 방식을 비교 했을 때 매개변수  $u$ 를 도입하여 식을 전개

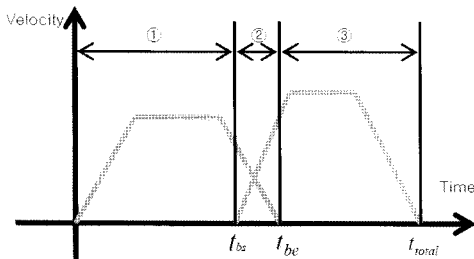


Fig. 4 Concept of velocity superposition

하였기 때문에 식이 간결하고 경우의 수를 따지지 않아도 되는 유리한 점을 가지게 된다.(단 중첩시간은 가/감속 시간 중 짧은 시간을 따르게 한다.)

### 3. 매개변수를 이용한 이중 모션 블랜딩

로봇의 모션은 관절 보간, 직선 보간, 원호보간 이루어진다. 일반 적인 CP는 동일한 모션, 관절 - 관절 보간, 직선 - 직선 보간, 원호 - 원호 보간으로 이루어진다. 이것은 서로 다른 운동일 경우 적용이 힘들다. 하지만, 제안한 매개변수를 이용한 이중 모션 블랜딩의 경우 서로 다른 운동을 쉽게 블랜딩 할 수 있다.

한 점에서 시작하여 목표점으로 관절 보간으로 사다리꼴 속도선도를 따라 이동할 때 목표점 부근에서 다음의 관절 보간과 블랜딩 하는 방법은 기본적으로 현재 이동 신호의 속도 선도와 다음 신호의 속도 선도를 시간 축을 중심으로 중첩을 시켜 현재 신호의 목표점 부근에서 부드럽게 블랜딩 시키는 것이다.

이중 모션 블랜딩은 서로 다른 두 가지 모션을 경유점 부근에서 속도 중첩을 하여 신속하고 부드러운 동작을 만들어 내는 것이다. 일반적인 로봇의 보간 방식은 관절보간, 직선 보간, 원호 보간이 대표적이다.

각 운동 방식을 속도 선도 기본 식 (11)을 이용하여 정의 할 수 있다.

관절 보간은 다음과 같이 정의 된다.

$$J(u(t)) = J_c - u(t)(J_c - J_s), u \in [0,1] \quad (20)$$

여기서  $J(u(t))$ 는 관절의 위치 값을 뜻하며 아랫 첨자  $e$ 는 종점을  $s$ 는 시점을 의미한다. 즉,  $J(u(t))$ 는 관절각의 목표점까지의 남은 이동 량이 된다.

같은 방식으로 직선 보간은 다음과 같이 정의 된다.

$$L(u(t)) = P_c - u(t)(P_c - P_s), u \in [0,1] \quad (21)$$

여기서  $L(u(t))$  는 직선 보간을 의미하며, 실제 로봇 끝단의 시점에서 종점까지 직선으로 움직일 때의 남은 이동 량이 된다.  $P_c$  와  $P_s$  는 직교 좌표계에서 직선 이동할 시 끝단의 위치 값으로 시점과 종점을 의미한다.

실제 로봇의 제어를 위해서는 관절의 값으로 변환하여야 한다. 직선에 따른 관절 값을 역기구학을 통하여 구하면 다음과 같이 나타 낼 수 있다.

$$\begin{aligned} J(u(t)) &= \text{InvKin}(L(u(t))) \\ &= \text{InvKin}(P_c - u(t)(P_c - P_s)), \quad u \in [0,1] \end{aligned} \quad (22)$$

여기서  $\text{InvKin}(L(u(t)))$  는 직선보간에 따른 역기구학 변환을 의미한다.

원호 보간도 같은 방법으로 정의하면, 아래 식과 같이 나타낼 수 있다.  $\text{InvKin}(C(u(t)))$ 는 원호보간에 따른 역기구학 변환을 의미한다.  $k$ 는 원호중심에서 원호와 이루는 수직벡터를 뜻한다.

$$\begin{aligned} C(u(t)) &= P_c + (P_s - P_c) \cos((1-u(t))\Delta\theta) \\ &\quad + k \times (P_s - P_c) \sin((1-u(t))\Delta\theta) \end{aligned} \quad (23)$$

$$J(u(t)) = \text{InvKin}(C(u(t))), \quad u \in [0,1] \quad (24)$$

위 식 (20) ~ (24)을 이용하여 이중모션 블렌딩을 위한 관계식을 아래와 같이 6가지로 정의 할 수 있다.

관절 - 직선 블렌딩

$$\begin{aligned} J_B^i(u_k(t_k), u_{k-1}(t_{k-1})) \\ = \text{InvKin}(L(u_k(t_k)))^i - u_{k-1}(t_{k-1})(J_{k-1,e}^i - J_{k-1,s}^i) \end{aligned} \quad (25)$$

관절 - 원호 블렌딩

$$\begin{aligned} J_B^i(u_k(t_k), u_{k-1}(t_{k-1})) \\ = \text{InvKin}(C(u_k(t_k)))^i + u_{k-1}(t_{k-1})(J_{k-1,e}^i - J_{k-1,s}^i) \end{aligned} \quad (26)$$

직선 - 원호 블렌딩

$$\begin{aligned} J_B^i(u_k(t_k), u_{k-1}(t_{k-1})) \\ = \text{InvKin}(C(u_k(t_k)))^i + \text{InvKin}(L(u_{k-1}(t_{k-1})))^i - J_{k-1,e}^i \end{aligned} \quad (27)$$

원호 - 직선 블렌딩

$$\begin{aligned} J_B^i(u_k(t_k), u_{k-1}(t_{k-1})) \\ = \text{InvKin}(L(u_k(t_k)))^i + \text{InvKin}(C(u_{k-1}(t_{k-1})))^i - J_{k-1,e}^i \end{aligned} \quad (28)$$

원호 - 관절 블렌딩

$$\begin{aligned} J_B^i(u_k(t_k), u_{k-1}(t_{k-1})) \\ = J^i(u_k(t_k)) + \text{InvKin}(C(u_{k-1}(t_{k-1})))^i - J_{k-1,e}^i \end{aligned} \quad (29)$$

직선 - 관절 블렌딩

$$\begin{aligned} J_B^i(u_k(t_k), u_{k-1}(t_{k-1})) \\ = J^i(u_k(t_k)) + \text{InvKin}(L(u_{k-1}(t_{k-1})))^i - J_{k-1,e}^i \end{aligned} \quad (30)$$

## 4. 측정 및 실험

앞 절에서 우리는 매개변수를 이용한 블렌딩 방법을 제안하였다. 기존의 방식<sup>(2)</sup>과 PTP 방식을 성능 평가를 통하여 제안한 매개변수를 이용한 방식을 증명하고자 한다. 실제 로봇의 운동 시, 빠르고 부드러운 동작의 척도는 진동과 이동 시간이다. 따라서 성능 평가는 진동 측정과 이동 시간을 한다. 성능평가를 위하여 자체 제작한 3-axis-SCARA robot을 이용하여 실험 및 측정을 하였다.

성능 평가에 앞서 기존알고리즘을 간단히 설명한다면 속도 중첩 시 발생하는 경우의 수를 고려한 것이다. 기존의 알고리즘은 발생할 수 있는 경우의 수를 고려하여 알고리즘이 만들어 졌기 때문에 경우의 수가 많고 연산이 복잡하고 연산 시간이 길다.

기존의 방식은 Fig. 6와 같이 여러 가지 경우의 수를 고려하여야 한다. 첫 번째 구간의 감속시간이 두 번째 가속시간보다 짧을 경우와 클 경우, 방향이 다를 경우, 최대 속도가 앞의 구간이 클 경우와 작을 경우 등을 고려하여 중첩을 하여야 한다. 따라서 프로그램 시 각 각의 경우에 따른 프로그램을 하여야 하기에 복잡하고 프로그래밍이 어려워진다. 하지만 제안한 방식은 매개변수를 이용하여 속도 선도를 작성하여 중첩 하므로 간단히 프로그램 할 수 있다.

제안한 방식과 기존의 방식의 성능 평가를 위한 제어기는 Fig. 7의 LabVIEW®의 Motion device<sup>(1)</sup> 사용하였으며, 프로그램은 LabVIEW®의 Graphical Programming 언어<sup>(3)</sup>를 사용하여 실험하였다. 그리고 제안한 블렌딩 방법을 검증하기 위하여 진동 측정 장비는 Zonic®사의 Medallion을 사용하였으며, 측정은 Fig. 8과 같이 측정 환경을 설정하고 로봇의 고정된 지지부에 센서를 부착하여 측정하였다.



Fig. 5 3-axis-SCARA robot

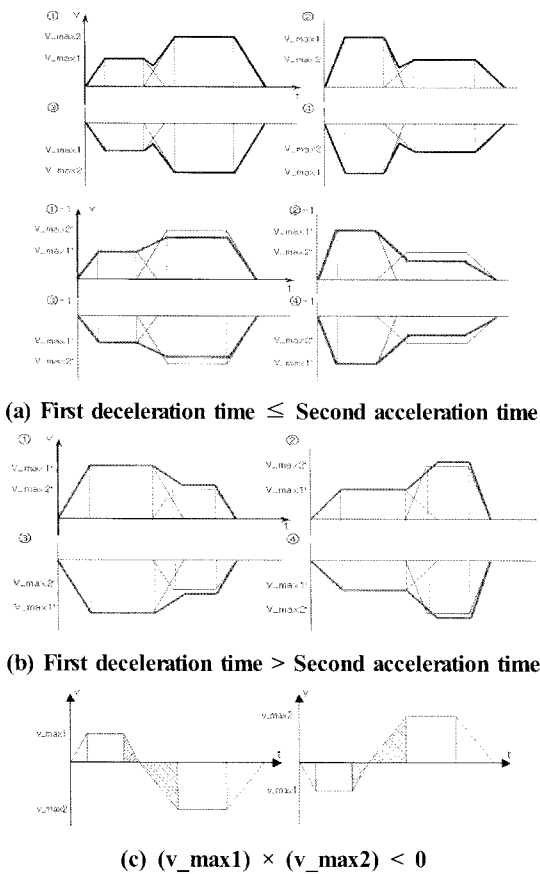


Fig. 6 8 cases of superposition for Kim's algorithm



Fig. 7 LabVIEW® Motion device

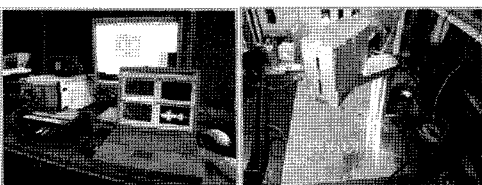


Fig. 8 Measuring equipment and Target Model

성능평가는 기존의 방식과 기본적인 PTP를 진동과 시간에 관하여 실험하였다. 그리고 비교 동작과 조건은 Fig. 9와 같이 가장 기본적인 동작인 두 직선이동을 하였으며, 이는 속도 중첩에 관한 성능을 보기 위함이다.

그 결과, 매개변수를 이용한 블랜딩의 경우 기존방식에 비해 진동이 1/2 정도로 감소하였으며, 시간은 유사하게 측정되었다. 그리고 PTP에 비해 1/100 정도로 감소하였음을 알 수 있다. 기존의 Kim's 방식과 제안한 방식의 진동측정 값의 크기 차이는 크지 않지만 발생하는 원인은 제어기의 동일 조건에서 연산시 버퍼 사용량에 따른 시간지연에 따른 차이로 발생된다. 그리고 진동 측정은 z축 이동이 없으므로 x, y축 방향을 측정하였다.

이종모션의 구현은 성능평가의 같은 조건으로 대표적인 관절 - 직선 블랜딩을 구현하였다. 궤적은 끝단에 펜을 이용하여 궤적을 보았다. 그 결과 Fig. 10과 같이 경유점에서 블랜딩이 발생하였으며 또한 추가적으로 Fig. 11~13에서 보듯이 관절 - 직선 보간 구현됨을 확인하였으며, 직선-원호-직선, 관절-원호-관절과 직선-관절-직선 등을 추가 확인하였다.

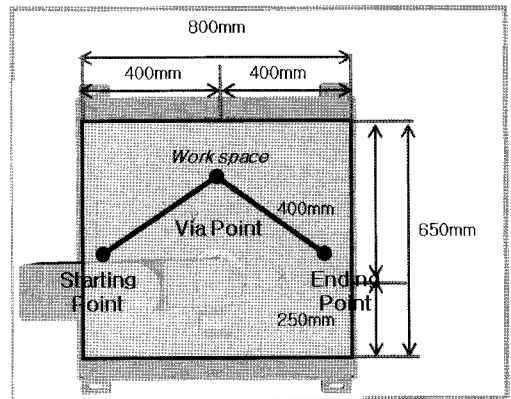


Fig. 9 Test Condition and Workspace

Table 1 Vibration accelerations of 3 algorithms

	g[RMS] in x-coord	g[RMS] in y-coord	Time(s)
PTP	0.2585	0.7558	6.3
기존	0.0052	0.0107	3.1
매개변수	0.0023	0.0064	3.0

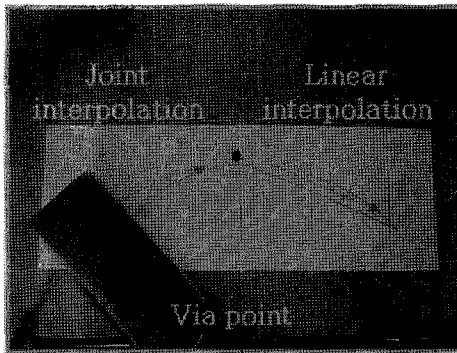


Fig. 10 Result of test motion

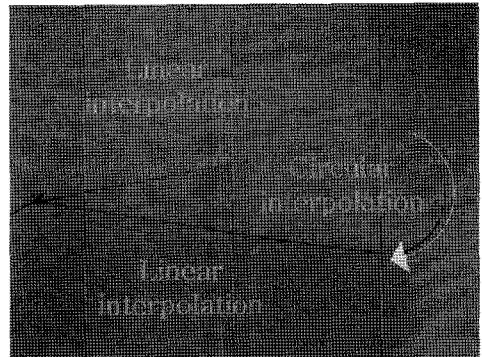


Fig. 11 Linear - Circular - Linear interpolation

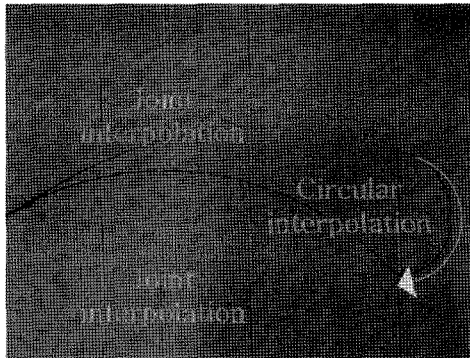


Fig. 12 Joint - Circular - Joint interpolation

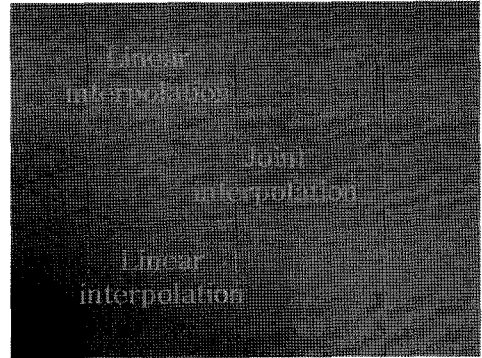


Fig. 13 Linear - Joint - Linear interpolation

## 5. 결론

본 연구에서는 매개변수를 이용하여 속도 선도를 정의하고 이를 이용한 속도 증첩방식을 제안하였다. 또한, 자체 제작한 3축 SCARA 형태의 로봇을 이용하여 모션 블렌딩을 구현하였으며, 매개변수를 이용함으로써 서로 다른 이종 모션 또한 간편하게 블렌딩 할 수 있는 방안을 제안하였다.

그리고 제안한 매개변수를 이용한 블렌딩의 성능 검증은 진동 측정 및 비교를 통하였으며, PTP 방식과 기존 방식에 비해 그 우수성을 확인하였다.

## 후 기

본 연구는 산업자원부와 한국산업기술재단의 지역혁신인력양성사업으로 수행된 연구결과임.

이 연구에 참여자는 [2단계 BK 21사업]의 지원비를 받았습니다.

## 참 고 문 헌

- (1) Kim, D. Y., "Development of a new weaving Algorithm using a Bezier Spline and A study on the Realization of CP (Continuous Path) Motion with Jerk Continuity," Master of Engineering treatise, Changwon National University, Republic of Korea.
- (2) National Instruments, 2002, *LabVIEW Motion control Fundamentals*, Republic of Korea.
- (3) National Instruments, 2004, *LabVIEW Fundamentals*, Republic of Korea.
- (4) Fu, K. S. Gonzalez, R. C., and Lee, C.S.G, 1987, *Robotics*, McGraw-Hill, New York, pp. 163~189.
- (5) Ju, J. H., "Monte-Carlo Simulation Technique for Error Analysis of 3-axis SCARA Robot using Observability," *Journal of the KSMTE*, Vol. 17, No. 4, pp. 8~14.

- (6) Kim, J. H., "Prototyping and Visualization Techniques of 3-axis SCARA Robot Using DOE and LabVIEW®," *Journal of the KSMTE*, Vol. 17, No. 3, pp. 40~46.
- (7) Kim, J. H., "Prototyping and Visualization Techniques of 3-axis SCARA Robot Using DOE and LabVIEW®," *Journal of the KSMTE*, Vol. 17, No. 3, pp. 40~46.
- (8) Shin, W. J., "A Study on the Error Compensation of Three-DOF Translational Parallel Manipulator", *Journal of the KSMTE*, Vol. 13, No. 3, pp. 44~52.
- (9) Thomas, M. and Tesar, D., 1982, "Dynamic Modeling of Serial Manipulator Arms," *Transactions of the ASME*, Vol. 104, pp. 218~228.