

재귀적 방법에 의한 NURBS 보간기

백대균*, 고태조#, 이재원**, 김희솔**

NURBS Interpolator with Recursive Method

Dae Kyun Baek*, Tae Jo Ko#, Jeh Won Lee** and Hee Sool Kim**

ABSTRACT

The purpose of this research is to find a simple and accurate NURBS interpolator for CNC systems such as robot, CMM and CNC machine tools. This paper presents a new design of NURBS interpolator for CNC system. The proposed algorithm used the recursive characteristics of NURBS equation, the previous incremental value and chord length for the sake of a constant chord length. Simulation study was conducted to see the performance of the proposed interpolator with reference-word and reference-pulse method. Consequently, an accurate and simple NURBS interpolator was possible for modern CNC systems.

Key Words : NURBS Interpolator(NURBS 보간기), Parameter update(변수갱신), Recursive(재귀), Reference-pulse interpolator(기준펄스 보간기), Reference-word interpolator(기준워드 보간기), Constant chord length(일정 선분 길이)

1. 서론

현재 자동차, 항공 및 각종 부품의 3차원 형상 모델링을 위하여 자유곡면(Free-formed surface)이 많이 사용되고 있다. 일반적으로 로봇, CNC 공작기계 그리고 3차원 측정기(CMM)에서 이러한 자유곡면의 제어는 CAD에서 모델링한 자유곡면을 CAD/CAM에서 수많은 선형화된 선분으로 나누어 CNC 시스템에 전송하여 수행한다. 이것은 기존의 CNC 시스템이 단지 직선과 원호보간을 주로 지원하고 있기 때문이다. 이와 같이 선형화된 선분을 CNC 시스템에 전달하여 제어할 때 다음과 같은 몇 가지 문제점을 가지고 있다. 먼저, CAD/CAM에서

CNC 시스템으로 데이터를 전송할 때 데이터 손실이 발생할 수 있고, 노이즈가 들어갈 수 있다. 또한 작은 직선으로 이루어진 데이터는 가속 특성으로 인하여 기계운동의 정밀도를 나쁘게 한다.

상기와 같은 문제점을 해결하기 위하여 최근 1개의 블록으로 다양한 종류의 직선 및 곡선을 표현하고, 또한 정의된 1개의 블록을 각각 한번씩의 가속과 감속으로 연속적으로 이동할 수 있는 방식에 관해서 많은 연구가 이루어지고 있다. 이것은 일반적으로 CAD에서 정의된 매개변수 곡선의 정보를 CNC 시스템에 입력하고, 수치제어 장치는 내부적으로 매개변수 곡선을 보간하여 정확한 운동경로를 생성하는 방법이다.

접수일: 2004년 7월 14일; 게재승인일: 2005년 3월 11일

Louisiana State University

* 교신저자: 영남대학교 기계공학부

E-mail : tjko@yu.ac.kr, Tel. (053) 810-2576

** 영남대학교 기계공학부

이와 같은 연구에 많이 사용되고 있는 자유곡선은 매개변수형으로서 Bezier curve, B-Spline, Cubic Spline과 NURBS(Non-Uniform Rational B-Spline) 등이다.¹ 이러한 함수에서 자유곡선의 보간이라고 하는

것은 다음과 같이 매개변수의 증분치 Δu_j 를 적절히 찾는 방법이다.

$$u_{j+1} = u_j + \Delta u_j$$

여기서 u_j 는 현재 매개변수이고, u_{j+1} 은 갱신된 매개변수이다. 곡선의 보간점들은 새로운 매개변수 u_{j+1} 을 함수에 대입하여 구할 수 있다.

이제까지의 보간방법을 살펴보면 1차 또는 2차 Taylor 급수를 이용하여 일정한 선분길이로 보간하는 방법을 많이 채택하고 있다.^{2,5} 가장 대표적인 방법은 1차 Taylor 정리와 자유곡선의 미분함수를 구하여 Δu_j 를 결정하여 일정한 이송속도로 제어하는 방법이다.^{2,3} 그러나 1차 Taylor 정리를 이용하면 절단오차(Truncation error)로 인하여 보간 오차가 많이 발생하므로 이러한 연구에 더하여 2차 Taylor정리를 이용하여 이송속도를 변화하여 자유곡선의 곡면부위에서 오차를 줄이는 방법에 대해 연구하였다.^{4,5} 한편, CNC 공작기계에서는 자유곡선의 보간 중에 일정한 속도로 기계를 제어하면 곡률에 따라 절삭부하가 변하여 가공정밀도가 저하되므로 곡선의 곡률을 고려하여 이송속도를 가변하여 균일한 절삭부하로 가공하는 방법을 이용하기도 한다.⁶ 어떤 방법을 사용하던 자유곡선을 보간하기 위해 1차 Taylor 급수를 이용하면 오차가 많이 발생하게 되고, 2차 Taylor 급수를 이용하면 계산이 복잡해지고 정확하게 가변 이송속도를 얻기가 어렵다.⁷

본 연구에서는 로봇, CNC 공작기계, CMM 등의 CNC 시스템에서 간단한 알고리즘으로 정확하게 일정속도로 자유곡선을 보간하는 새로운 방법을 제안하고자 한다. 일반적인 CNC 시스템에서 자유곡선의 보간은 원호보간^{8,9}과 같이 항상 일정한 선분길이로 보간하는 것이 가장 좋은 방법이다. 원호보간은 곡선의 방정식이 간단하여 보간이 용이하나 NURBS곡선과 같은 매개변수형 곡선은 곡선정도가 다양하고 복잡하여 항상 일정한 길이로 보간하는 것은 어렵다. 따라서 본 연구에서는 자유곡선의 보간도 원호보간과 마찬가지로 항상

일정한 선분길이로 보간하여 CNC 시스템을 정밀하게 제어할 수 있는 보간기를 제안하고자 한다. 이 방법은 Taylor급수나 미분방정식을 이용하지 않고 NURBS 방정식 자체를 이용하여 항상 일정한 길이로 보간하게 된다. 즉, 현재의 매개변수 증분치(Δu_j)를 구하기 위해 그전의 증분치와 곡선의 길이를 이용하여 비례식으로 증분치를 구하고, 다시 NURBS 방정식을 이용하여 증분치를 개선하여 최종 증분치를 구하는 방법을 채택하였다. 본 연구에서 개발한 보간 알고리즘으로 기준펄스 보간방식⁸과 기준워드 보간방식⁹에 대해 시뮬레이션하여 보간기의 성능을 검증하였다.

2. NURBS 보간기 설계

2.1 NURBS 곡선의 정의¹

일반적으로 곡선이나 곡면을 나타내기 위하여 여러 가지 수학적 모델 중에서 Cubic-spline, Coons, Bezier, B-spline 등이 대표적이다. 그 중에서 B-spline은 형상의 정의가 간편하며, 조정점의 이동에 따른 형상의 변화가 전체 곡선이 아닌 조정점 근처에 국부적으로 작용하고, 곡면이 2차 미분이 가능하여 매끄럽다는 장점을 가지고 있다. B-spline의 정의는 다음과 같다.

$$P(u) = \sum_{i=0}^n B_i \cdot N_{i,k}(u) \quad u_{\min} \leq u < u_{\max}, \quad 1 \leq k \leq n \quad (1)$$

여기서 $P(u)$ 는 스플라인 위치 벡터, B_i 는 조정점 벡터이며, $N_{i,k}(u)$ 는 B-spline의 기저함수로서 다음과 같이 정의된다.

$$N_{i,0}(u) = \begin{cases} 1, & u_i \leq u \leq u_{i+1} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$$N_{i,k}(u) = \frac{u - u_i}{u_{i+k} - u_i} \cdot N_{i,k-1}(u) + \frac{u_{i+k+1} - u}{u_{i+k+1} - u_{i+1}} \cdot N_{i+1,k-1}(u) \quad (3)$$

여기서 절점벡터는 $U = \{u_0, \dots, u_m\}$ 와 같고, u_i 는 절점이며, $u_i \leq u_{i+1}$, $i = 0, \dots, m-1$ 의

성질을 갖는다. 기저함수는 재귀적(Recursive)인 성질을 가지고 있는데 기저함수의 재귀적 특성을 보기 위해 2차 기저함수의 예를 Fig. 1에 나타내었다.¹

이상과 같이 정의된 일반적인 B-spline에 해당 조정점으로 곡선을 끌어당기는 가중치 h_i 와 다양한 곡선을 표현하기 위해 유리함수(rational function)로 표현한 일반적인 NURBS는 다음과 같이 표시된다.

$$P(u) = \frac{\sum_{i=0}^n B_i \cdot h_i \cdot N_{i,k}(u)}{\sum_{i=0}^n h_i \cdot N_{i,k}(u)} = \sum B_i R_{i,k}(u) \quad (4)$$

여기서
$$R_{i,k}(u) = \frac{h_i \cdot N_{i,k}(u)}{\sum_{i=0}^n h_i \cdot N_{i,k}(u)}, \quad h_i \geq 0$$

이 때 두 조정점간의 가중치를 조정함으로써 기존의 수치제어 장치에 적용되고 있는 직선 및 원호를 정의할 수 있다. 이상의 NURBS곡선의 정의와 특성에서 살펴본 것처럼 NURBS곡선은 기저함수의 재귀적인 성질에 의해 곡선이 부드럽게 변하며 가중치에 의해 곡면이 조정점으로 쏠리는 정도를 결정할 수 있다.

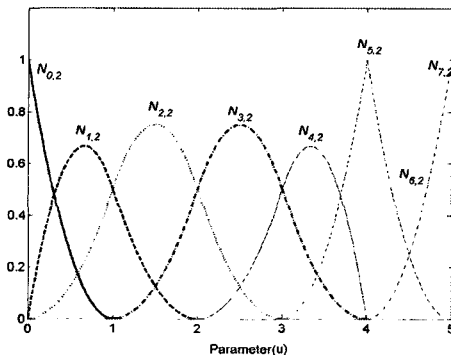


Fig. 1 The second-degree basis functions, $U=\{0,0,0,1,2,3,4,4,5,5,5\}$.

2.2 NURBS 보간알고리즘

2.1에서 살펴본 NURBS 곡선의 고유한 성질을 이용하여 일정한 선분길이를 보간할 수 있도록 매

개변수 u_j 를 구할 수 있다. 먼저 NURBS 곡선의 특성을 살펴보기 위해 매개변수 증분치 Δu_j 값을 균일한 간격 0.02로 설정하여 NURBS곡선을 구하여 일정 영역만을 확대하여 Fig. 2에 xy평면상에 나타내었다. 사용된 조정점, 가중치 그리고 절점벡터는 임의의 값이다.

Fig. 2에서 보는 것처럼 일정한 매개변수 증분치로 시뮬레이션하였을 때 곡률 반경이 큰 부분에서는 선분의 길이가 길고, 곡률반경이 작은 부위에서는 선분의 길이가 짧아진다. 곡선에서 임의의 두 점 $P(u_{j+1})$ 와 $P(u_j)$ 사이의 거리를 s_j 라 할 때 식 (5)와 같다.

$$s_j = |P(u_{j+1}) - P(u_j)| \quad (5)$$

이때 선분길이의 변화율(%)은 식 (6)과 같이 정의할 수 있다.

$$dl_r = \frac{|s_j - s_{j-1}|}{s_j} \times 100 \quad (6)$$

식 (6)을 이용하여 Δu_j 가 0.02로 균일할 때, 선분의 길이 변화율을 나타내어보면 Fig. 3과 같다. Fig. 3에서 보는 것과 같이 각 매개변수 사이의 곡선의 길이는 급격한 차이는 없고 완만하게 조금씩 변한다. 동일한 매개변수의 증분치에서 앞 선분과 뒤 선분의 길이의 비는 거의 10%이내에 있고 아주 곡률이 심한 부분에서도 30%이내에 있는 것을 알 수 있다. 즉, NURBS의 재귀적인 성질에 의해 Δu_j 가 균일할 때 현재의 코드길이 s_j 는 그 앞의 s_{j-1} 과 길이와 유사하게 된다.

본 연구에서는 이러한 NURBS 식의 고유한 성질을 이용하여 보간하게 된다. 즉, 일정한 매개변수 증분치에서 NURBS의 현재 선분길이는 그 선분의 전후의 길이와 유사하다는 성질을 이용하여 매개변수 증분치를 구하게 된다.

먼저 임의의 증분치 Δu_j 를 구하는 방법을 설명하면 다음과 같다. Fig. 4에서 보는 것과 같이 목적의 선분길이는 기계의 최소지령단위(BLU)와 동일한 값 d 로 둔다. 그럼에서 보는 것과 같이 앞 선 증분치 Δu_{j-1} 로 보간된 선분의 길이는 s_{j-1}

이다. 매개변수 증분치 Δu_j 를 구하기 위해 Fig. 4

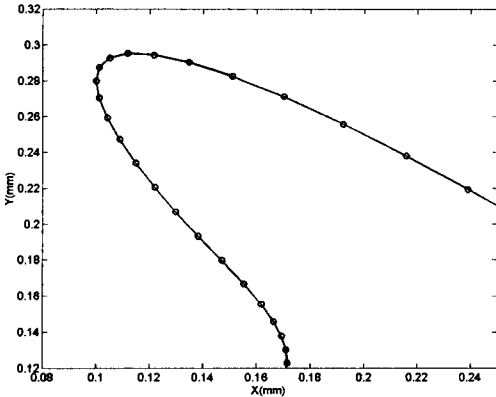


Fig. 2 The example of NURBS in xy plain.

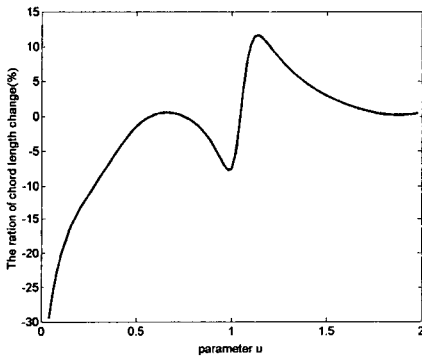


Fig. 3 The ratio of the length difference between two adjacent chords.

에서 보는 것과 같이 Δu_{j1} 에서 Δu_{jm} 까지 증분치를 n_1 번 갱신하여 최종의 증분치($\Delta u_{jm} = \Delta u_j$)를 구한다. 이때 매번 증분치를 갱신할 때마다 보간된 선분의 길이가 d 에 가까워지도록 한다.

따라서 인접한 선분의 길이는 조금씩 변한다는 NURBS의 성질에 따라 Δu_j 를 구하기 위한 첫 번째 임시 증분치 Δu_{j1} 은 다음과 같은 비례식을 이용할 수 있다.

$$\Delta u_{j-1} : s_{j-1} = \Delta u_{j1} : d, \quad j > 1 \quad (7)$$

여기서 $s_{j-1} = |P(u_j) - P(u_{j-1})|$

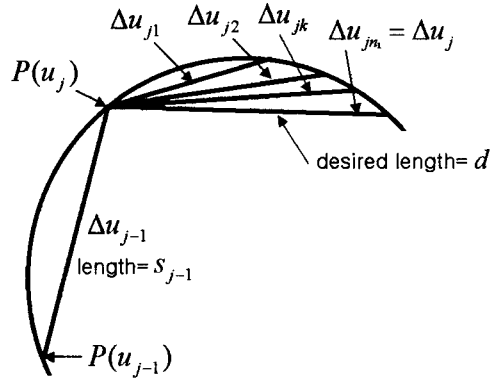


Fig. 4 The concept of the interpolation.

식(7)에서 지수 $j1$ 은 j 를 구하기 위한 첫 번째 지수라는 의미이다. 이 식에서 보는 것과 같이 앞 선 보간에서 Δu_{j-1} 로 s_{j-1} 의 선분의 길이를 구 하였으므로 선분의 길이가 d 가 되기 위해서는 갱신된 Δu_{j1} 이 필요하다. 식 (7)을 다시 쓰면 다음과 같다.

$$\Delta u_{j1} = \frac{d \cdot \Delta u_{j-1}}{s_{j-1}} \quad j > 1 \quad (8)$$

임시 증분치 Δu_{j1} 으로 구한 곡선의 선분길이는 구하고자 하는 길이 d 와 오차가 발생하므로 식 (8)과 마찬가지로 개선된 증분치를 Δu_{j2} 부터 Δu_{jk} 까지 차례대로 구할 수 있다. 이 식을 일반화시키게 되면 증분치 Δu_j 를 구하기 위한 개선된 증분치 Δu_{jk} 의 식은 다음과 같다.

$$\Delta u_{jk} = \frac{d \cdot \Delta u_{j(k-1)}}{s_{j(k-1)}} \quad j > 1, \quad k = 2 \cdots n_1 \quad (9)$$

여기서 $s_{j(k-1)} = |P[u_{j(k-1)}] - P(u_j)|$,

$$u_{j(k-1)} = u_j + \Delta u_{j(k-1)}$$

따라서 최종 구하고자 하는 증분치는

$\Delta u_j = \Delta u_{j_1}$ 이다. 이때 n_1 은 반복적인 계산의 회수를 결정하는 변수로서, n_1 이 크면 계산시간은 많이 걸리나 정밀도가 증가하게 된다.

식 (9)는 현재의 증분치 Δu_j 를 구하는 방법이다. 그런데 최초의 증분치 Δu_0 는 앞선 증분치와 선분의 길이가 없으므로 다른 방법으로 찾아야 한다. NURBS 곡선의 전체길이의 합은 각 조종점들간의 길이의 합보다는 작다는 원리를 이용하면, 최초의 매개변수 증분치는 다음과 같이 구할 수 있다. 먼저 조정점들로 이루어진 선분의 전체 길이 합을 구하면 다음과 같다.

$$l = \sum_{i=0}^{n-1} |B_{i+1} - B_i| \quad (10)$$

절점의 최대값 u_{max} 와 최소값 u_{min} 의 차이를 u_s 라 할 때, 먼저 곡선에서 초기 증분치 Δu_0 를 구하기 위한 첫 번째 임시 증분치 Δu_{01} 은 다음과 같은 비례식을 이용할 수 있다.

$$u_s : l = \Delta u_{01} : d \quad (11)$$

즉, 곡선의 전체길이를 조정점간의 선분의 전체길이의 합으로 가정하여 초기 증분치를 구하기 위한 임시 증분치를 계산한다. 식 (11)을 다시 쓰면

$$\Delta u_{01} = \frac{u_s d}{l} \quad (12)$$

임시 증분치 Δu_{01} 으로 보간점을 구하여 NURBS 식에서 계산한 점을 $P(u_{01})$ 라 할 때 선분의 길이는 다음과 같다.

$$s_{01} = |P(u_{01}) - B_0| \quad (13)$$

초기 증분치 Δu_0 를 구하기 위한 개선된 증분치 Δu_{0k} 의 일반화된 식은 식 (12)와 마찬가지로 다음과 같이 쓸 수 있다.

$$\Delta u_{0k} = \frac{d \cdot \Delta u_{0(k-1)}}{s_{0(k-1)}} \quad k = 2 \cdot \cdot \cdot m_1 \quad (14)$$

여기서 $s_{0(k-1)} = |P[u_{0(k-1)}] - B_0|$,

$$u_{0(k-1)} = u_{min} + \Delta u_{0(k-1)}$$

식 (14)에서 최종 구하고자 하는 초기 증분치 $\Delta u_0 = \Delta u_{0m_1}$ 이다. 그리고 m_1 은 초기 증분치를 구하기 위한 반복적인 계산의 회수를 결정하는 변수이다. m_1 은 식 (9)의 n_1 보다 더 큰 값을 가져야 한다. 왜냐 하면 곡선의 길이를 가정하여 계산한 식 (12)에서 오차가 크므로 곡선을 정확하게 보간하기 위해 n_1 보다는 더 크게 할 필요가 있다. 3장의 시뮬레이션에서 파라미터 n_1 과 m_1 의 결정 방법을 설명한다.

식 (9)를 이용하여 구하는 증분치 Δu_j 는 기계의 운동 중에 구해지므로 반복적인 계산의 회수를 결정하는 변수 n_1 은 계산시간을 고려하여 결정해야 한다. 만약 n_1 이 2이면 식 (8)은 앞선 곡선의 길이를 이용하므로 곡선의 길이를 구하기 위해 실제로 NURBS 방정식을 한 번 계산한다. 즉, NURBS 방정식을 이용하는 회수는 $(n_1 - 1)$ 번이다.

이상과 같이 기저함수의 재귀적 성질을 이용하는 본 보간기는 기존의 NURBS 보간기처럼 미분이나 다른 조작이 필요없이 NURBS 곡선방정식 자체를 이용하여 보간한다. 그러나 초기 증분치를 구하는 식 (14)는 앞선 곡선의 길이와 증분치가 없으므로 조정점으로부터 곡선의 길이를 구하였으므로 약간의 오차가 발생한다. 변수 n_1 과 마찬가지로 변수 m_1 도 보간 전에 값을 설정하여 NURBS 곡선을 보간할 수 있다. 그러나 m_1 을 설정하지 않고, 첫 번째 곡선의 길이가 설정한 오차 범위내에 들어가도록 아래의 2.2.1절과 같은 알고리즘으로 식 (14)를 대신하여 Δu_0 를 계산할 수 있다.

2.2.1 알고리즘에 의한 초기 증분치 결정

step1: 첫 번째 선분길이가 목적의 선분길이에 대한 오차범위, $e(\%)$ 를 결정하고, 식 (12)를 이용하여 Δu_{01} 을 계산한다.

step2: 변수 k 에 초기값 k=2 을 설정한다.

step3: 다음 식으로 증분치를 계산한다.

$$\Delta u_{0k} = \frac{d \cdot \Delta u_{0(k-1)}}{s_{0(k-1)}}$$

여기서 $s_{0(k-1)} = |P[u_{0(k-1)}] - B_0|$ and

$$u_{0(k-1)} = u_{min} + \Delta u_{0(k-1)}$$

step4: $u_{0k} = u_{\min} + \Delta u_{0k}$ 를 이용하여 보간점 $P(u_{0k})$ 을 구하고 곡선의 길이 s_{0k} 를 구한다.

step5 : 만약 $\left| \frac{d-s_{0k}}{d} \right| \times 100 \geq e$ 이면 $k=k+1$ 을 수행한 후 step3으로 간다.

만약 $\left| \frac{d-s_{0k}}{d} \right| \times 100 < e$ 이면

$\Delta u_{0k} = \Delta u_0$ 로 초기 증분치를 결정한다.

2.3 기준 워드(Sampled data) 보간 방식

기준 워드 보간방식⁹은 최근에 CNC공작기계가 고속으로 발전함에 따라 많이 채택되고 있다. 이 방식은 x, y, z 좌표와 각 축의 속도를 지령하는 방식이다. 2.2절의 보간 알고리즘을 이용하여 매개변수의 증분치 Δu_j 를 계산할 수 있고, 매개변수 $u_{j+1} = u_j + \Delta u_j$ 를 식 (4)에 대입하여 NURBS곡선의 각 위치점을 계산할 수 있다. 따라서 현재의 좌표 $P(u_j)$ 에서 다음 좌표 $P(u_{j+1})$ 는 아래의 식 (15)와 같이 연속적으로 구할 수 있다.

$$P(u_{j+1}) = [X(j+1), Y(j+1), Z(j+1)] ,$$

$$u_{j+1} = u_j + \Delta u_j ; \quad (15)$$

각 위치에서 최초 속도가 V 일 때 각 축의 속도는 다음과 같다.

$$V_x(j+1) = \frac{V \cdot [X(j+1) - X(j)]}{DS(j+1)}$$

$$V_y(j+1) = \frac{V \cdot [Y(j+1) - Y(j)]}{DS(j+1)}$$

$$V_z(j+1) = \frac{V \cdot [Z(j+1) - Z(j)]}{DS(j+1)} \quad (16)$$

여기서 $DS(j+1) = \sqrt{[X(j+1) - X(j)]^2 + [Y(j+1) - Y(j)]^2 + [Z(j+1) - Z(j)]^2}$

2.4 기준 펄스 보간 방식(Reference pulse)

기준 펄스 보간방식^{8,10,11}은 각 축에 펄스를 지령하여 펄스 주파수로 속도를 제어한다. CNC 시스템의 각 축은 한 펄스에 1 BLU만큼 움직인다. 2.2절의 보간 알고리즘을 이용하여 매개변수의 증분치 Δu_j 를 계산할 수 있고, 매개변수를 식 (4)를

이용하여 NURBS곡선의 각 위치점을 구할 수 있다. 현재의 x, y, z 좌표와 다음의 좌표를 비교하여 차이가 0.5BLU 이상이면 각 축에 펄스를 지령하도록 한다.

현재 좌표 $P(u_j)$ 에서 위치이동은 벡터를 이용하여 다음과 같이 쓸 수 있다.

$$dP(u_{i+1}) = [dX, dY, dZ] \quad (17)$$

여기서 dX, dY, dZ 는 각 축의 단위이동을 나타낸다. 만약 +1이면 양의 방향으로 이동하고, -1이면 음의 방향으로 이동하고, 0이면 이동하지 않는다. 현재의 위치에서 증분치 Δu_j 로 NURBS곡선의 다음 좌표 $P(u_{j+1})$ 를 구하여 다음과 같은 규칙으로 각 축의 단위이동을 구한다.

- if $X(j+1) \geq X(j)$ and $|X(j+1) - X(j)| \geq 0.5BLU$,
then $dX = +1$
- if $X(j+1) \leq X(j)$ and $|X(j+1) - X(j)| \geq 0.5BLU$,
then $dX = -1$
- if $Y(j+1) \geq Y(j)$ and $|Y(j+1) - Y(j)| \geq 0.5BLU$,
then $dY = +1$
- if $Y(j+1) \leq Y(j)$ and $|Y(j+1) - Y(j)| \geq 0.5BLU$,
then $dY = -1$
- if $Z(j+1) \geq Z(j)$ and $|Z(j+1) - Z(j)| \geq 0.5BLU$,
then $dZ = +1$
- if $Z(j+1) \leq Z(j)$ and $|Z(j+1) - Z(j)| \geq 0.5BLU$,
then $dZ = -1$

이상과 같은 논리를 적용하면 각 축의 단위 이동은 다음과 같은 7가지의 경우가 발생한다.

- $(X + dX, Y, Z)$
- $(X, Y + dY, Z)$
- $(X, Y, Z + dZ)$
- $(X + dX, Y + dY, Z)$
- $(X + dX, Y, Z + dZ)$
- $(X, Y + dY, Z + dZ)$
- $(X + dX, Y + dY, Z + dZ)$

이러한 규칙으로 단위이동을 위한 펄스를 발생시키고, 연속적으로 그 다음 증분치를 구하고 보간

점을 찾는다.

3. 보간성능 시뮬레이션

3.1 기준워드 보간방식

본 연구에서 제안한 보간기의 성능을 검증하기 위하여 Fig. 2의 NURBS곡선에 대해 시뮬레이션을 수행하였다. 곡률이 작은 부분에서 실제로는 제어가 어렵지만 본 연구에서는 새로운 알고리즘을 제안하므로 얼마나 정확하게 선분의 길이가 보간되는지를 시뮬레이션하고자 하였다. 따라서 본 연구에서 제안한 보간기로 기준 워드 방식으로 보간 선분길이를 $1\ \mu\text{m}$ 로 가정하여 시뮬레이션하였다. 2.2절에서 제안한 방법으로 식 (9)에서 $n_1=2$ 로 설정하고 식 (14)에서 $m_1=2$ 로 설정하여 NURBS보간을 수행하여 Fig. 5에 나타내었다. 각 점을 보간하기 위해 NURBS 방정식 이용회수는 $(n_1 - 1=1)$ 로 한 번 이용하였고, 첫 번째 보간점을 찾기 위해 NURBS 방정식을 이용한 회수도 $(m_1 - 1=1)$ 로 설정하였다. 즉, NURBS 방정식을 한번 이용하여 곡선의 길이 s_{1l} 을 구한 후 증분치 $\Delta u_j = \Delta u_{j2}$ 를 구하였다. Fig. 5는 3차원에서 살펴본 보간 결과이고, Fig. 6(a)는 xy평면에서 살펴본 보간 결과이다. Fig. 6(b)는 원래의 NURBS곡선과 보간한 데이터 사이의 오차를 보기 위해 곡률이 가장 작은 부분을 확대하여 xy평면에 대해 나타낸 것이다. 그림에서 보는 것과 같이 원래의 NURBS곡선의 반경이 약 $1\ \mu\text{m}$ 에서도 NURBS곡선을 한 번 이용하여 증분치를 구하였음에도 불구하고 보간이 잘 되었음을 알 수 있다.

다음으로 각 선분의 길이를 구하여 보간기의 성능을 살펴보았다. 매개변수의 증분치 Δu_j 를 0.02로 균일하게 하였을 때 선분의 길이를 구하여 보면 Fig. 7과 같다. Fig. 7에서 보는 것과 같이 각 선분의 길이는 크게 변한다. 최고 큰 선분의 길이는 $5\ \mu\text{m}$ 이고, 가장 작은 선분의 길이는 거의 0에 가깝다. 이와는 반대로 Fig. 8은 본 연구에서 제안한 방법으로 보간한 선분의 길이를 구하여 나타낸 것이다. 그림에서 보는 것과 같이 선분의 길이가 $1\ \mu\text{m}$ 로 보간되었음을 알 수 있다.

본 시뮬레이션에 사용된 NURBS 곡선의 반경이 보간 선분길이인 $1\ \mu\text{m}$ 수준임에도 불구하고 정

확하게 보간되었음을 알 수 있다. Fig. 8에서 보면 오차가 작지만 곡률이 $1\ \mu\text{m}$ 인 부근에서 약간의

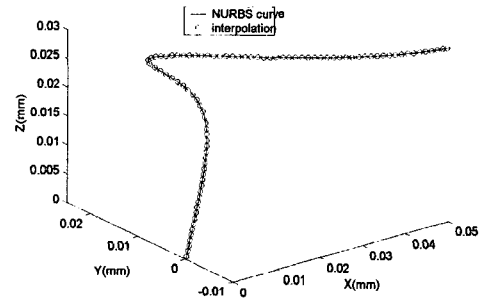
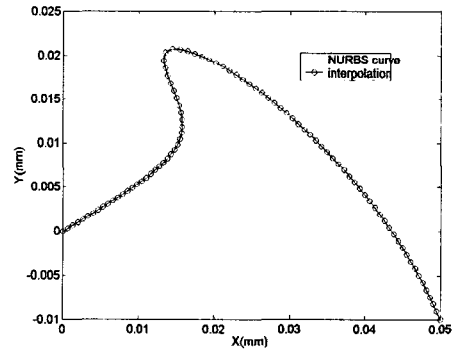
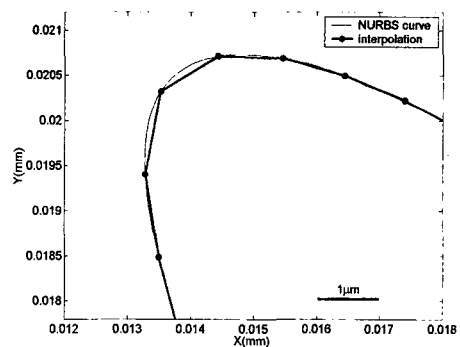


Fig. 5 The 3-dimensional interpolation of the curve.



(a) overall view of the interpolation of the curve



(b) scaling up the smallest curvature

Fig. 6 The interpolation of the curve in xy plane.

오차가 발생함을 알 수 있다. 더욱 정확하게 보간하기 위해 식 (9)의 m_1 값을 변화시켜 선분의 길이를 구하여 Fig. 9에 나타내었다. $n_1=1$ 일 때는 NURBS 곡선의 방정식을 이용하지 않고 식 (8)에서 보는 것과 같이 앞선 곡선의 길이(s_{j-1})와 앞선 증분치(Δu_{j-1}) 만을 이용하여 현재의 증분치(Δu_j)를 구하였다. $m_1=2$ 일 때는 NURBS 곡선의 방정식을 한 번 이용하였고, $m_1=3$ 일 때는 두 번 이용하였다. 그림에서 보는 것처럼 $m_1=3$ 일 때는 곡률이 $1 \mu\text{m}$ 인 부분에서도 선분의 길이가 $1 \pm 0.02 \mu\text{m}$ 이내로 보간되었다. 따라서 곡선의 길이가 점차적으로 변한다는 NURBS곡선의 기저함수의 특성을 이용하여 비례식과 두 번의 NURBS방정식을 이용하여 중간 보간점을 찾은 후 아주 정확한 보간점을 찾을 수 있었다.

실제 NURBS 보간에서는 최소곡률이 기계 정밀도보다는 크기 때문에 $m_1=2$ 로 하여 NURBS방정식을 한 번만 이용해도 충분히 정밀하게 보간할 수 있다. 그러나 초기 증분치를 구하기 위해 $m_1=2$ 로 하여 NURBS 방정식을 한 번 이용하면 Fig. 8 과 9에서 보는 것과 같이 초기 보간점에서 약간의 오차가 발생한다.

따라서 2.2.1절에서 제시한 탐색알고리즘을 이용하여 초기 증분치를 시뮬레이션하여 보았다. Fig. 5와 6에서 초기 증분치가 보간길이 d 의 1% 이내에 들어가기 위하여 알고리즘에서 NURBS 방정식을 이용한 회수를 확인한 결과는 2번 이었다. 이때 식 (10)으로부터 l 은 0.157mm 이고 보간 후 NURBS 곡선의 전체길이를 계산해 본 결과는 0.089mm 였다. NURBS 곡선의 길이와 조정점을 다양하여 변화시키며 오차범위를 1%로 설정하여 2.2.1절의 알고리즘으로 시뮬레이션 해본 결과 NURBS방정식 이용회수는 한 두번 이내였다. 조정점들의 함으로 구한 곡선의 길이와 NURBS 곡선의 전체 길이 차이가 커도 NURBS 방정식의 이용회수는 크지 않다는 것을 알 수 있었다. 따라서 본 연구에서 제한한 방법으로 다음과 같이 2가지 방법으로 보간하면 고속으로 정밀하게 NURBS 곡선을 보간할 수 있다.

1. 초기 증분치를 구하기 위한 식 (14)에서 $m_1=3$ 을 설정하고, 두 번째 이후의 증분치를 구하기 위한 식 (9)에서 $m_1=2$ 를 설정하여 보간한다.

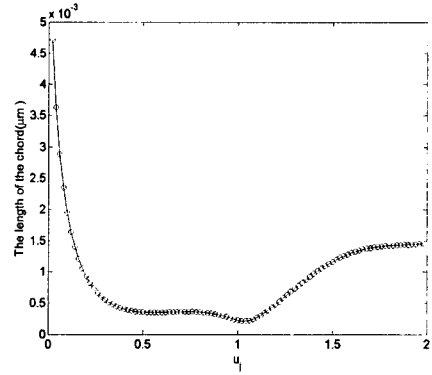


Fig. 7 The length of the chord with the constant incremental parameter($\Delta u_j=0.02$).

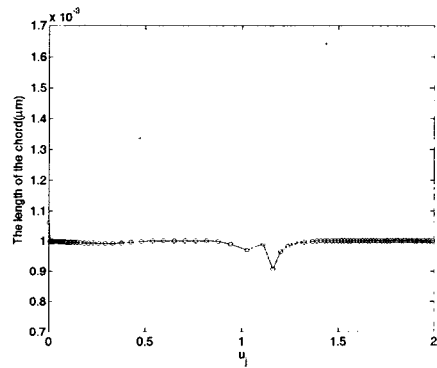


Fig. 8 The length of the chord with the interpolation($n_1=2, m_1=2$).

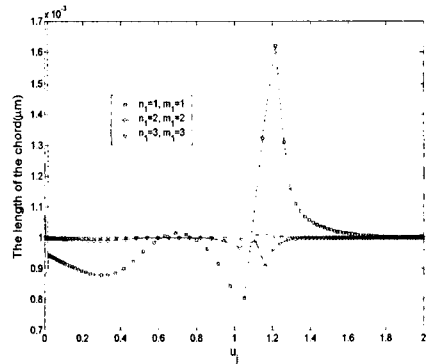


Fig. 9 The length of the chord with the interpolation varying the parameter.

2. 초기 증분치를 구하기 위한 알고리즘에서 오차 범위(1%이내)를 설정하고, 식 (9)에서 $n_1=2$ 를 설정하여 보간한다.

3.2 기준펄스 보간방식

본 연구에서 제시한 방법으로 2.4절의 기준펄스 방식에 대해 시뮬레이션하여 보았다. 기준펄스 보간 방식에서는 $m_1=3$ 으로 하고, $n_1=2$ 로 하여 Fig. 2의 NURBS곡선과 원호형상의 NURBS곡선을 보간하였다. 기준펄스 보간방식에서 최소지령단위 (BLU)를 $1 \mu\text{m}$ 로 설정하여 시뮬레이션하였다. Fig.

2의 NURBS곡선을 기준펄스 방식으로 보간하여 Fig. 10과 Fig. 11에 나타내었다. Fig. 10은 기준펄스 보간방식에 의한 3차원 보간을 나타내고, Fig. 11은 보간된 결과를 xy평면에서 나타내었다. Fig. 11에서 보는 것과 같이 보간 오차는 0.5 BLU 이내 에 있는 것을 알 수 있다. 원호 형상의 NURBS 곡선에 대해 기준 펄스 방식으로 보간하여 원래의 곡선과 함께 Fig. 12에 나타내었다. 그림에서 보는 것과 같이 원호형상의 NURBS 곡선의 보간 결과는 일반원호 보간과 동일한 결과를 나타냄을 알 수 있다. 즉, 본 연구에서 제안한 보간 방법은 정확하게 일정 선분길이를 구할 수 있으므로, 기준펄스 보간 방식에 적용하기 용이하고 0.5BLU 이내의 오차로 보간 할 수 있다.

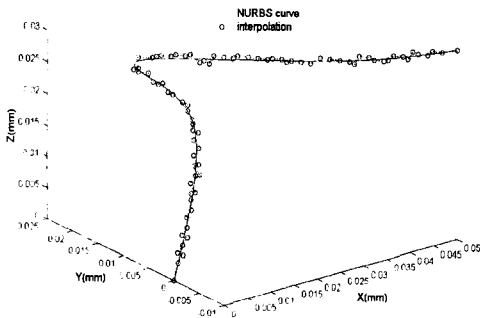


Fig. 10 The 3-dimensional interpolation of the curve with reference-pulse interpolation.

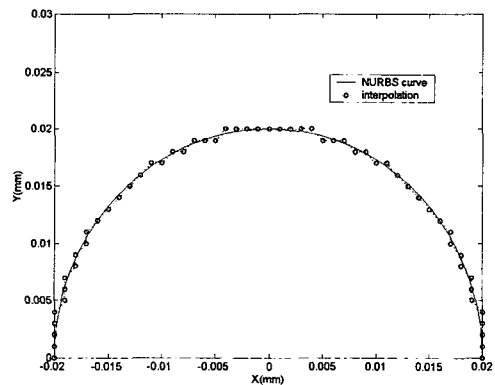


Fig. 12 The interpolation of the circular NURBS curve with reference-pulse interpolation.

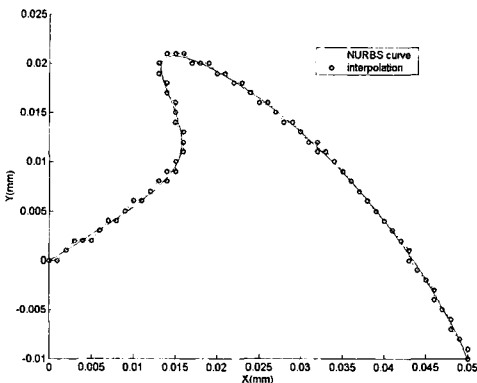


Fig. 11 The interpolation of the curve in xy plane with reference-pulse interpolation.

4. 결론

본 연구에서는 자유곡선의 수치제어를 위하여 고속, 고정도로 보간할 수 있는 새로운 NURBS 보간기를 제안하였다. 이것은 NURBS 곡선의 재귀적인 성질을 이용하여 일정선분의 길이를 갖도록 한 보간기이다. 즉, 현재의 매개 증분치를 구하기 위해 바로 앞의 매개 증분치와 선분의 길이를 이용하여 비례식으로 임시의(첫 번째) 증분치를 구하고, 다음으로 임시의 증분치를 이용하여 NURBS 방정식으로부터 곡선의 길이를 구한 후에 오차를 줄일 수 있도록 증분치를 연속적으로 갱신하여 최종 증분치를 구하는 방식이다. 본 연구에

서 제안한 방법은 NURBS곡선의 미분방정식을 사용하지 않고 NURBS방정식을 1-2회 이용하여 고속, 고정도로 보간할 수 있다. 본 연구에서 제안한 알고리즘을 기준워드 보간방식과 기준펄스 보간방식으로 시뮬레이션하였을 때 곡선의 최소반경이 기계의 최소지령단위 수준에서 정확하게 일정 선분길이로 보간할 수 있었다. 따라서 본 연구에서 제안한 방법으로 각 종 CNC시스템에서 고속, 고정도로 NURBS 곡선을 제어할 수 있다.

후기

본 연구는 산업자원부의 지역혁신 인력양성사업의 결과로 수행되었음.

참고문헌

1. Piegl, L., Tiller, W., "The NURBS Book," Springer, 1997.
2. Koren, Y., Lo, C. C., Shpitalni, M., "CNC Interpolators: Algorithms and Analysis," ASME Manufacturing Science and Engineering, PED-64, pp. 83-92, 1993.
3. Zhang, Q. G., Greenway, R. B., "Development and Implementation of a NURBS Curve Motion Interpolator," Robotics and Computer-Integrated Manufacturing, Vol. 14, No. 1, pp. 27-36, 1998.
4. Yang, D. C. H., Kong, T., "Parametric interpolator Verses Linear Interpolator for Precision CNC Machining," Computer-Aided Design, Vol. 26, No. 3, pp. 225-234. 1994.
5. Yeh, S. S., Hsu, P. L., "The Speed-Controlled Interpolator for Machining Parametric Curves," Computer-Aided Design, Vol. 31, No. 5, pp. 349-357. 1999.
6. Tikhon, M., Ko, T. J., Lee, S. H., Kim, H. S., "NURBS Interpolator for Constant Material Removal Rate in Open NC Machine Tools," International Journal of Machine Tools & Manufacture, Vol. 44, No. 2-3, pp. 237-245, 2004.
7. Farouki, R. T., Tsai, Y. F., "Exact Taylor Series Coefficients for Variable-Feedrate CNC Curve Interpolators," Computer-Aided Design, Vol. 33, No. 2, pp. 155-165, 2001.
8. Koren, Y., Masory, O., "Reference-Pulse Circular Interpolators for CNC System," ASME Journal of Engineering for Industry, Vol. 103, No. 1, pp. 131-136, 1981.
9. Koren, Y. Masory, O., "Reference-Word Circular Interpolators for CNC System," ASME J. of Engineering for Industry, Vol. 104, No. 4, pp. 400-405, 1982.
10. Hong, W., Yang, M., "NURBS interpolation algorithm for CNC machines," Journal of KSPE, Vol. 17, No. 12, pp. 115-120, 2000.
11. Kiritsis, D., "High Precision Interpolation Algorithm For 3D Parametric Curves Generation," Computer-Aided Design, Vol. 26, No. 11, pp. 850-856. 1994.