

곡선의 효율적 자료 교환

김혁진*

An Effective Data Exchange of Curves

Hyeock-Jin Kim*

요 약

현재 많은 종류의 컴퓨터 그래픽스 및 CAD 시스템들이 개발되어 있고, 앞으로도 그 추세는 응용분야가 다양한 만큼 더욱 증가될 전망이다. 그러나 그 종류가 많아 질 수록 이들 시스템간에 자료 교환 문제가 발생하게 된다. 이러한 문제들을 해결하기 위한 표준들이 IGES(Initial Graphics Exchange Specification)와 STEP(STandard for the Exchange of Product model data)등에서 이루어지고 있다. 그럼에도 불구하고 이 기종 시스템간의 자료 교환이 신뢰성 있게 효율적으로 이루어지지 못하여 만족스럽지 못한 실정이다. 특히, 기하학적 모델링 시스템에서 기본이 되는 곡선 및 곡면의 경우, 개발자들이 서로 다른 방식으로 구현하기 때문에 이 기종간에 자료 교환을 충족시키지 못하고 있다.

본 논문은 이 기종간의 자료 교환에서의 곡선 변환(curve conversion)에 관한 연구로서, 곡선의 형상 모양이 덜 중요한 환경에서 4가지 형태의 서로 다른 곡선들간에 곡선을 효율적으로 변환하고 그때의 전송되는 자료들을 구체적으로 분석하였다.

Abstract

Recently computer graphics and CAD systems of many kinds have been developed, and this trends will be more increased in the future because the application fields are diverse. However, data exchange problems between these systems will be occurred as the system kinds increase. Standards for solving these problems are made by IGES and STEP, etc. Nevertheless, it is unsatisfactory now because data exchange between different systems is not running well with reliability and efficiency. Because vendors implement with different ways, data exchange of the curves and surfaces between different systems which are based in the geometric modeling system, does not satisfy.

This paper is a research for curve conversion in the data exchange between different systems. Also this paper analyzed the transferring data which are effectively conversed the curves among the four different types of curves in the less important curve shape environment.

* 청운대학교 전자계산학과 전임강사

** 본 논문은 청운대학교 1998년도 학술연구조성비 지원으로 연구되었음.

논문접수: 98.8.28. 심사완료: 98.9.20.

I. 서론

오늘날 컴퓨터 관련 기술이 빠르게 발전을 하고 있고, 기업들은 점점 통합되는 세계시장과 제품정보의 공유를 통하여 컴퓨터를 이용한 설계와 제조 분야에 관심을 갖고 연구하고 있다. 이러한 시대적 변화에서 인터넷 환경은 전세계의 분산된 정보를 쉽고 빠르게 접근할 수 있는 매우 효과적인 정보 공유의 형태를 제공한다. 컴퓨터 그래픽스 및 CAD/CAM/CAE나 네트워크를 중심으로 한 netCAD등의 도입이 확대됨에 따라 다양한 시스템들이 개발되고 있고, 앞으로도 CAGD 분야에서의 그 추세는 응용분야가 다양한 만큼 더욱 증가될 전망이다. 그러나 그 종류가 많아 질 수록 이들 시스템간에 자료 교환 문제가 발생하게 된다. 정보의 공유 차원에서 임의의 한 시스템에서 얻은 자료를 또 다른 시스템에서 같은 의미의 자료로 사용하기 위해서는 가능한 한 변형 없이 자료 교환(data exchange)이 이루어 져야 한다. 이러한 아이디어의 구현은 이상보다는 좋은 결과를 얻지 못하였다. 그러므로 이러한 문제들을 해결하기 위한 노력들이 CADEX(CAD Exchange), CAD*I(CAD*Interfaces, ESPRIT 322), IGES (Initial Graphics Exchange Specification)와 STEP (STandard for the Exchange of Product model data)등에서 이루어지고 있지만, 이 기종 시스템간의 자료 교환이 만족스럽게 이루어지진 못하고 있는 실정이다(2),[12],[16].

일반적으로 기하학적 모델링 시스템에서 기본이 되는 곡선 및 곡면의 경우, 개발사들이 서로 다른 방식으로 구현하기 때문에 피할 수 없는 자료의 변형이 이루어지게 된다.[16],[17] 최근에는 보통 표준에서 채택한 NURBS를 쓰지만, 표준 이전의 시스템 등에서는 Bézier나 B-spline등을 사용하였다. 그러므로 표준 이전의 시스템에서 만든 자료를 최근의 시스템에서 쓰기 위해서라도 곡선변환이 요구되고, 이 반대의 경우에도 필요하게 된다.

본 논문은 이러한 자료 교환의 필요성에 의하여,

J.S.M. Vergeest[16]의 곡선 변환 연구를 확장하였다. 기하학 영역에서의 곡선변환(curve conversion)을 구현하고 실제로 전송되어지는 자료들을 수치적 예로서 제시·분석하였다. 다양한 곡선들간의 교환 관점에서 조망하여 연구함으로써, 앞으로 그래픽스 및 CAD 시스템 등 곡선의 형상 모양이 덜 중요한 환경에서 곡선변환이 효율적으로 이루어질 수 있는 기본적인 접근방식을 제공한다.

본 논문의 구성은 다음과 같다. 제2절에서는 생산 자료 교환(product data exchange) 인터페이스인 IGES와 STEP 및 곡선 근사 변환에 관련된 연구들에 대해서 간단히 살펴본다. 제3절에서는 전체적인 곡선의 종류를 4개의 곡선으로 분류하고, 이들 곡선들을 각각 정의하고 속성을 알아본다. 제4절에서는 제3절의 곡선들을 변환한 결과들을 예로서 보여주고, 마지막으로 제5절에서는 본 논문에 대한 결론과 추후연구에 대해 언급한다.

II. 자료 교환

1. IGES와 STEP

IGES(Initial Graphics Exchange Specification)나 STEP(STandard for the Exchange of Product model data)등 이 기종간의 자료교환을 위한 모델이(그림2.1)에 도시되어 있다.

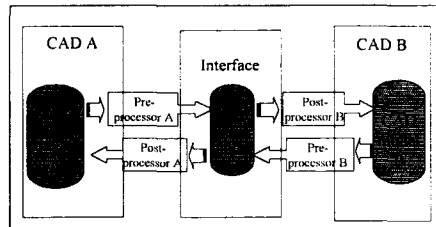


그림2.1. 자료교환 모델

대부분의 CAD시스템에서는 어느 특정 CAD시스템에 의존하지 않는 효율적인 자료 교환 방법을 모색하여 왔다. 서로 다른 CAD/CAM 시스템간에 형상 정보를 왜곡되지 않게 전달하는 것의 중요성에 따라 여러 가지

중립 자료 형식이 제안되었다. 그 중 IGES는 STEP 이전의 자료 교환을 위한 U.S. 국가 표준으로, ANSI Y14.26M 위원회에서 만든 IGES1.0(1980년)을 시초로 하여 여러 번의 발전을 거듭하여 현재 IGES6.x이 개발되어있다[17].

자료 교환은 간단한 형상 엔티티 일수록 지원이 잘되지만 B-Spline과 같은 복잡한 형상 엔티티 일수록 잘 안되며 홀(hole) 형상 등과 같은 존재하지 않거나 불합치적인 것들은 더욱 지원이 되지 않는다.

기존 CAD 시스템간의 자료 교환은 제품 형상을 중심으로 정의한 IGES 파일을 이용해 왔다. 그러나 제품의 해석 및 생산에 필요한 정보는 표준이 정립되지 않아 이 기종 시스템간에 정보 교환이 원활히 이루어지지 않고 있다[2],[12].

STEP은 1984년부터 ISO TC184/ SC4/ WG1(at NIST)에서 제정하여 1994년 12월 12개 Part가 국제 표준 ISO10303으로 채택되었고, 그 이후에 4개 Part가 더 추가되었다. STEP에서 모든 정보들은 정보 모델링 언어인 EXPRESS로 표현된다[19]. 산업의 각 응용분야에서 사용되는 응용프로그램들은 STEP의 응용 프로토타입을 사용함으로써 다른 시스템과의 자료 교환을 제공한다.

곡선 및 곡면의 경우, IGES나 STEP은 NURBS 내에서만 이루어진 자료 교환 표준이지만 본 논문에서는 전체적인 일반 곡선형태들 간의 교환을 다룬다.

2. 근사 변환 방법

기하학적 표현이 서로 다른 시스템간의 곡선 및 곡면의 교환에서 가능한 3가지 시나리오가 있다[15]. 첫째, 수학적으로 정확한 변환이 가능하다. 두 시스템 모두 같은 기하학적·매개변수적으로 동일한 경우에 해당된다. floating point roundoff error만이 발생하는 경우로, 예를 들면, B-spline과 piecewise power basis 형태(monomial) 간의 변환이다. 두 번째, 매개변수적으로는 변하고 기하학적으로는 정확한 교환이다. 삼각함수에 의한 매개변수화된 썸플 같은 경우이다. 셋째, 주어진 공차(tolerance)내에서 가장 좋은 곡선에 근사하는 방법으로 수학적으로는 정확성이 없다. 예를 들어, 임의의 degree가 가능한 시스템에서 3차 곡선만 가능한 시스템으로 변환한다면 더 높은 degree로 근사해야 한다.

서로 다른 기하학적 모델링 시스템을 사용하는 곡선

및 곡면 자료 교환에서 같은 기저 함수를 사용하는 곡선 및 곡면 내에서 근사 변환(approximate conversion) 알고리즘들이 종종 사용된다. 즉, 서로 다른 표현 방식을 사용하는 시스템이나 같은 표현 방식을 지원하더라도 최대 지원하는 degree가 다른 시스템간의 곡선 및 곡면의 자료 교환을 할 때에는 자료를 받아들이는 시스템에 맞게 변환을 해야 한다. 곡선 자료를 저장하는 표현방식 중 Non-Rational B-spline인 경우 basis 변환으로 간단한 연산에 의해 같은 degree의 다른 곡선으로 변환이 가능하다. 그러나 시스템이 지원하는 최대 degree가 입력 곡선의 degree보다 낮은 경우 주어진 공차내에서 시스템이 지원하는 degree로 근사해야 한다. Rational B-spline을 지원하지 않는 시스템에 입력된 Rational B-spline으로 근사해야 한다. Rational B-spline은 원호 근사를 위해 주로 사용되며, 곡면의 경우 Rational B-spline으로 표현된 곡면은 회전곡면이 대부분이다. 그러므로 회전 곡면인지를 검사하여 회전 곡면이라면 일반 Rational B-spline인 경우와 다른 근사과정을 거치는 것이 바람직하다.

다음은 이런 시스템에 적용할 수 있는 근사 변환 방법에 관한 문헌들이다[1], [3], [4], [5], [7], [8], [9], [11], [13], [14].

J.E. Hoelzle[7]은 integral polynomial curve의 구간적 보간을 위한 최적 knot placement scheme을 고안했다. 미리 정의한 오차범위를 만족시키기 위한 최소의 knot 수를 결정한다. Hoelzle 알고리즘은 짝수 order(홀수 degree) 곡선을 이용하는 구간적 Hermite 보간을 사용한다.

L. Dannenberg와 H. Nowacki[3]는 1985년 보간 하는 구간적 다항식의 degree가 $2ic+1$ 보다 큰 경우에, integral polynomial curve의 Hoelzle 방법을 확장하였다. 여기서 ic은 knot들에서 요구된 연속성 order이다. 즉, 높은 차수의 다항식 곡면을 낮은 차수의 다항식 곡면으로 근사 변환 하는 내용으로서 체계적으로 기존의 알고리즘을 통합한 알고리즘을 제시하였다.

Hoschek[8]는 1987년 매개 변수화와 GCn 조건을 적정한 변환을 사용하여 spline 곡선을 conversion 하는 방법을 제안했다. 즉, Bernstein basis를 사용하는 높은 order integral spline 곡선을 낮은 order spline 세그먼트로 근사 하는 알고리즘이다.

N.M. Patrikalakis[13]는 nonuniform rational B-spline을 더 낮은 degree nonuniform integral

B-spline으로 approximate conversion하는 알고리즘을 제안했다.

L. Bardis와 N.M. Patrikalakis[1]는 위의 approximate conversion을 곡선에서 NUBS 곡면으로 근사 변환하였다.

J. Hoschek와 F.J. Schneider, P. Wassum[9]는 Hoschek의 알고리즘을 GC1-와 GC2 spline 곡면으로 확장하였다.

III. 곡선의 정의

본 논문에서는 곡선의 일반성을 고려하여 변환을 위한 곡선의 범주를 다음과 같이 분류한다.

- Bézier curve
- rational Bézier curve
- B-spline curve
- NURBS curve

nth-degree Bézier curve의 정의는 다음과 같다.[6],[15],[18]

$$a(u) = \sum_{i=0}^n B_{i,n}(u) P_i \quad 0 \leq u \leq 1 \quad (3.1)$$

여기서 {P_i}은 control polygon을 형성하는 control point이고, {B_{i,n}(u)}은 Bernstein polynomial로서

$$B_{i,n}(u) = \frac{n!}{i!(n-i)!} u^i (1-u)^{n-i} \quad (3.2)$$

이다.

nth-degree rational Bézier curve의 정의는 다음과 같다[15].

$$b(u) = \frac{\sum_{i=0}^n B_{i,n}(u) w_i P_i}{\sum_{i=0}^n B_{i,n}(u) w_i} \quad 0 \leq u \leq 1 \quad (3.3)$$

여기서 {w_i}은 weights이다.

pth-degree B-spline curve의 정의는 다음과 같다.[6],[15],[18]

$$c(u) = \sum_{i=0}^n N_{i,p}(u) P_i \quad a \leq u \leq b \quad (3.4)$$

여기서 {P_i}는 control polygon을 만드는 control point이고, {N_{i,p}(u)}은 degree p(order p+1)인 B-spline basis function으로 다음과 같이 정의한다.

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

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u)$$

여기서, u_i는 knot vector를 이루는

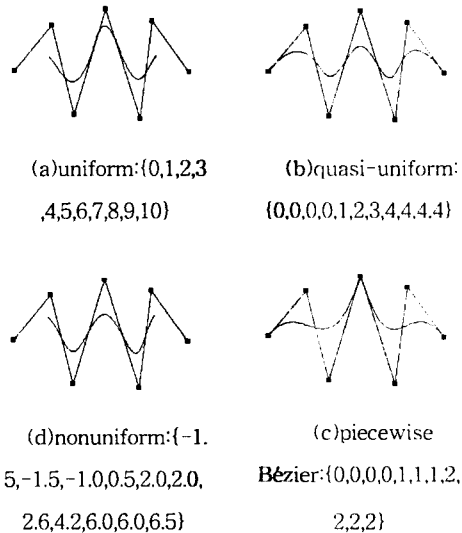


그림3.1. STEP의 곡선 형태

non-decreasing순서를 갖는 실수의 knot들로서 다음과 같이 정의한다

$$U = \{u_0, \dots, u_m\}, \quad u_i \leq u_{i+1}, \quad i = 0, 1, \dots, m-1.$$

nonperiodic과 nonuniform인 경우의 knot vector에서 처음과 끝의 a와 b knot 각각의 수는 p+1번의 반복을 갖는다.

$$U = \{a, \dots, a, u_{p+1}, \dots, u_{m-p-1}, b, \dots, b\}$$

여기서 degree는 p, control point 수는 n, knot

vector 수가 m 이면, $m = n + p + 1$ 이 성립된다. 또, $n = p$ 이고 $U = \{0, \dots, 0, 1, \dots, 1\}$ 이면 $c(u)$ 는 piecewise polynomial curve이며 Bézier curve가 된다. $c(a) = P_0$ 이고 $c(b) = P_n$ 일 때, endpoint interpolation 속성을 갖는다. 다른 속성으로는 affine invariance, convex hull property, local modification scheme 등의 특성이 있다.

p th-degree NURBS(Non-Uniform Rational B-Spline) curve의 정의는 다음과 같다[6],[15].

$$d(u) = \frac{\sum_{i=0}^n N_{i,p}(u) w_i P_i}{\sum_{i=0}^n N_{i,p}(u) w_i} \quad a \leq u \leq b \quad (3.5)$$

여기서 $\{w_i\}$ 은 weights, $\{P_i\}$ 는 control polygon 을 만드는 control point, $\{N_{i,p}(u)\}$ 은 nonuniform knot vector

$$U = \{a, \dots, a, u_{p+1}, \dots, u_{m-p-1}, b, \dots, b\}$$

에서 정의된 p th-degree인 B-spline basis function이다.

NURBS는 현재 국가 및 국제 표준으로 geometric information을 취급하는 representation, design 및 자료 교환 등을 위한 산업계 표준으로서도 많이 사용되고 있으며, NURBS 알고리즘은 빠르고 numerically stable한 성질을 갖고 있다.

knot vector는 크게 clamped와 unclamped 두 그룹으로 나눌 수 있고, 또 두 그룹 각각은 knot spacing에 따라 uniform과 nonuniform으로 나눈다 [15]. 예를 들면,

-clamped uniform: $\{0,0,0,0,1,2,3,4,4,4,4\}$,

$\{-0.5,-0.5,-0.5,1,2.5,4,4,4\}$

-clamped nonuniform: $\{0,0,0,2,3,6,7,7,7\}$,

$\{0,0,0,0,1,2,2,3,4,5,5,5,5\}$

-unclamped uniform: $\{-3,-2,-1,0,1,2,3$

$,4,5\}$, $\{0,1,2,3,4,5,6,7,8,9,10,11\}$

-unclamped nonuniform: $\{0,0,1,2,3,4\}$,

$\{-2,-1,0,4,5,6,7\}$ 이다. clamped와 unclamped는 처음과 마지막 값이 degree+1 만큼 multiplicity를 갖는지에 따라 구분한다.

STEP에서 정의한 NURBS의 곡선 및 곡면은 knot vector에 의한 것으로, uniform type에 따라 다음의 4가지로 구분한다[16].

- uniform(unclamped uniform)
- quasi-uniform(clamped uniform)
- piecewise Bézier(equally spaced)
- nonuniform(otherwise)

{그림3.1}은 STEP에서 제공하는 knot vector에 따른 곡선들의 특징을 보여준다.

IV. 곡선변환

곡선 및 곡면을 변환하기 위한 기술적인 요구사항은 다음과 같다[16].

- generality of curves
 - the number of different entities must be kept to a minimum
 - no exchange of redundant data
 - high numerical stability
 - automatic implementation
- 각 곡선들간의 관계를 살펴보자.

먼저, n th-degree인 rational Bézier curve $b(u)$ 에서 weight w_i 가 0이 아닌 모두 같은 값이라면 식(3.4)와 같이 $a(u) = b(u)$ 인 Bézier curve로 유도된다.

식(3.3)은 모든 i 에 대해서 $w_i > 0$ 이라면 n th-degree rational Bézier curve를 다음과 같이 정의 할 수 있다.

$$b(u) = \sum_{i=0}^n R_{i,n}(u) P_i \quad 0 \leq u \leq 1 \quad (4.1)$$

여기서 $R_{i,n}(u)$ 는

$$R_{i,n}(u) = \frac{B_{i,n}(u) w_i}{\sum_{j=0}^n B_{j,n}(u) w_j}$$

인 n th-degree rational Bézier curve의 rational basis function이다. 이 곡선은 $R_{0,n}(0) = R_{n,n}(1) = 1$ 인 곡선이며, 식(3.2)의 Bernstein polynomial과의 관계는 모든 i 에 대해서 $w_i > 1$ 이면 모든 i 에 대해서 $R_{i,n}(u) = B_{i,n}(u)$ 이 성립한다. 즉, $B_{i,n}(u)$ 는 $R_{i,n}(u)$ 의 특별한 경우이다.

식(3.5)와 같은 pth-degree인 B-spline에서 $n = p$ 이고 knot vector가 $\{0, \dots, 0, 1, \dots, 1\}$ 이면 $c(u)$ 는 piecewise polynomial curve이며 Bézier curve가 된다. B-spline은 $c(a)=P_0$ 이고 $c(b)=P_n$ 인 endpoint interpolation 속성을 갖는다.

pth-degree인 NURBS 식(3.6)에서 $a=0$ $b=1$ 이고, 모든 i 에 대해서 $w_i > 0$ 이면

$$d(u) = \sum_{i=0}^n Q_{i,p}(u)P_i \quad a \leq u \leq b \quad (4.2)$$

으로 다시 쓸 수 있다. 여기서 $Q_{i,p}(u)$ 는

$$Q_{i,p}(u) = \frac{N_{i,p}(u)w_i}{\sum_{j=0}^n N_{j,p}(u)w_j}$$

인 rational basis function으로, $u \in [0, 1]$ 에서 piecewise rational function이다. 이 곡선은 $Q_{0,p}(0) = Q_{n,p}(1) = 1$ 인 곡선이며, 모든 i 에 대해서 $w_i = 1$ 이면 모든 i 에 대해서 $Q_{i,p}(u) = N_{i,p}(u)$ 이 성립한다. 즉, $N_{i,p}(u)$ 는 $Q_{i,p}(u)$ 의 특별한 경우이다. $a \neq 0$ 인 상황에서 모든 i 에 대해서 $w_i = a$ 이면 모든 i 에 대해서 $Q_{i,p}(u) = N_{i,p}(u)$ 이다.

곡선들의 변환에서 교환되어야 할 실제 자료는 3D control point (polygon, P_i), point 수(n), degree(d), weight(w), knot vector(k)등이고, 그 중에서도 기본적으로 전송되는 자료는 3D control point와 그의 개수이다. 이후에 설명하는 내용에서는 3D control point는 언급하지 않는다. 나머지 자료들은 곡선의 정의에 따라 선택적으로 전송될 수가 있다. 효율적인 변환을 위해서 가능한 한 불필요한 자료는 전송하지 않고, default 값을 자동 생성하여 전송되는 자료의 량을 최소화한다.

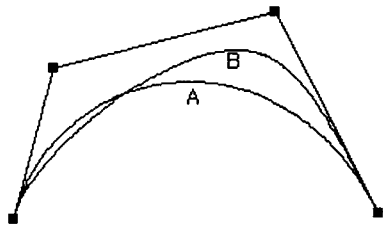


그림4.1. cubic Bézier curve

degree는 control point 개수로 생성하는 방법과 시스템이나 사용자가 요구하는 degree로 결정하는 2가지 방법으로 얻을 수 있다. knot의 default는 quasi-

uniform으로 생성한다. 4 곡선간의 서로 곡선 변환하는 모든 경우는 다음과 같다.

```

void curveConversion()
{
// subscript f : from curve
// d : degree of curve
// k : knot vector
// w : weights
switch(ConversionType)
{
case (from Bézier to RBézier) :
    d = (n-1);
    weight_generator();
    return;
case (from Bézier to B-spline) :
    d = df or input;
    knot_generator();
    return;
case (from Bézier to NURBS) :
    d = df or input;
    weight_generator();
    knot_generator();
    return;
case (from RBézier to Bézier) :
    d = (n-1);
    return;
case (from RBézier to B-spline):
    d = df or input;
    knot_generator();
    return;
case (from RBézier to NURBS):
    d = df or input;
    w = wf;
    knot_generator();
    return;
case (from B-spline to Bézier) :
    d = (n-1);
    return;
case (from B-spline to rational Bézier) :
    d = (n-1);
}
}
    
```

```

weight_generator():
return:
case (from B-spline to NURBS) :
d = df:
weight_generator():
k = kf:
return:
case (from NURBS to Bézier) :
d = (n-1):
return:
case (from NURBS to rational Bézier) :
d = (n-1):
w = wf:
return:
case (from NURBS to B-spline) :
d = df:
k = kf:
return:
}

void knot_generator()
{
// create quasi-uniform
}

void weight_generator()
{
// create wi = 1, i=0,...,n-1
}
    
```

B-spline	Bézier	4	3		
	rational Bézier	4	3	{1,1,1,1}	
	NURBS	4	3	{1,1,1,1}	{0,0,0,0, 33,0.66, 1,1,1}
NURBS	Bézier	4	3		
	rational Bézier	4	3	{1,1,3,1}	
	B-spline	4	3		{0,0,0,0, 33,0.66, 1,1,1}

각 곡선들간의 실제 변환되는 예들을 살펴보자.

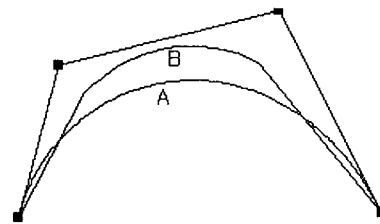


그림 4.2. knot vector {0,0,0,0,33,0.66,1,1,1}의 B-spline curve

from	to	number	Degree	Weights	Knots
Bézier	rational Bézier	4	3	{1,1,1,1}	
	B-spline	4	3		{0,0,0,0, 1,1,1,1}
	NURBS	4	3	{1,1,1,1}	{0,0,0,0, 1,1,1,1}
R Bézier	Bézier	4	3		
	B-spline	4	3		{0,0,0,0, 1,1,1,1}
	NURBS	4	3	{1,1,3,1}	{0,0,0,0, 1,1,1,1}

표 4.1. 3차 곡선의 예

그림4.1의 A곡선은 cubic Bézier curve를 나타낸 것이고 이 곡선을 rational Bézier curve, B-spline curve 그리고 NURBS curve로 변환하였을 때에도 같은 곡선의 결과를 얻는다.

그림4.1에서 B곡선은 w2=3인 cubic rational Bézier curve이고, 이 곡선을 Bézier curve과 B-spline curve으로 변환하면 A곡선을, NURBS로 변환하면 이 곡선과 같은 B곡선을 얻는다.

그림4.2에서 B곡선은 knot vector를 {0,0,0,0,3, 3,0.66,1,1,1}로 정의했을 때의 B-spline curve을 보여준다. 이 곡선을 cubic Bézier curve나 cubic rational Bézier curve로 변환하면 A곡선으로, NURBS로 변환하면 B-spline과 같은 A곡선으로 변환된다.

그림4.3에서 N곡선은 weight를 {1,1,3,1}, knot

vector를 $\{0,0,0,0.33,0.66,1,1,1\}$ 로 정의한 NURBS 곡선이다. 이 N곡선을 Bézier curve, rational Bézier curve와 B-spline곡선으로 각각

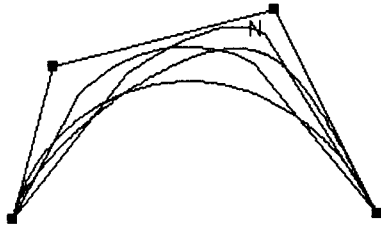


그림4.3. NURBS로부터 변환된 곡선들

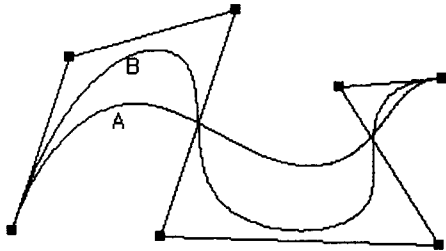


그림4.4. 7개의 control point로 정의된 곡선

변환한 곡선을 나타낸 것이다. 그러나, [그림4.1]의 A곡선을 NURBS로, weight를 $\{1,1,1,1\}$, knot vector를 $\{0,0,0,0,1,1,1,1\}$ 로 정의한 NURBS를 Bézier curve, rational Bézier curve와 B-spline곡선으로 각각 변환하여도 동일한 형상의 곡선을 얻었다.

$\{0,0,0,0.33,0.66,1,1,1\}$ 의 B-spline curve

표.4.1은 위의 곡선들이 변환 될 때 전송되거나 생성된 자료의 예를 보인 것이다.

그림4.4의 A곡선은 6차 Bézier curve이고, 이 곡선을 default weight를 생성하여 rational Bézier curve로 변환하였을 때에도 같은 A곡선을 얻었다.

또 B곡선은 default knot를 생성하여 3차 B-spline곡선을, 여기에 default weight를 생성하여 3차 NURBS로 변환한 결과이다. 마찬가지로 역 변환하였을 때에도 같은 결과를 얻는다. 표.4.2는 [그림4.4]의 A와 B곡선의 실제 전송되거나 생성된 자료의 예를 보인 것이다.

표.4.3)에 곡선이 변환될 때 전송 또는 생성되는 일반적인 자료 상태를 정리하였다.

표4.2 6차 곡선으로부터 변환되는 예

from	to	number	Degree	Weights	Knots
Bézier	rational Bézier	7	6	$\{1,1,1,1,1,1,1\}$	
	B-spline	7	3		$\{0,0,0,0,1,2,3,4,4,4,4\}$
	NURBS	7	3	$\{1,1,1,1,1,1,1\}$	$\{0,0,0,0,1,2,3,4,4,4,4\}$
RBézier	Bézier	7	6		
	B-spline	7	3		$\{0,0,0,0,1,2,3,4,4,4,4\}$
B-spline	NURBS	7	3	$\{1,1,1,1,1,1,1\}$	$\{0,0,0,0,1,2,3,4,4,4,4\}$
	rational Bézier	7	3	$\{1,1,1,1,1,1,1\}$	
	Bézier	7	6		
NURBS	B-spline	7	3		$\{0,0,0,0,1,2,3,4,4,4,4\}$
	rational Bézier	7	3	$\{1,1,1,1,1,1,1\}$	
	Bézier	7	6		

V. 결론

기하학적 모델링 시스템에서 기본이 되는 4종류 곡선 및 곡면의 개념과 그들간의 자료 교환을 간략하게 소개하였고 특히, 곡선의 형상 모양보다는 수치적 자료들이 더 중요시되는 기하학 부분에서의 전체 곡선 및 곡면의 효율적인 변환을 모색하였다.

IGES나 STEP에서의 곡선/곡면의 교환은 NURBS 내에서 이루어진 것이지만 본 연구는 전체적인 범주로 곡선 교환을 확장하여 일반화하였다. 또 쉽게 곡면으로도 확장할 수 있다. geometry 변환에서는 일반성과 효율성이 강조되는데, default값을 생성하여 가능한 전송자료를 최소화함으로써 효율성을 고려하였다. 실제 전송되는 자료들을 예로 보임으로서 이 요구 조건을

검증하였다. 또한, 곡선별로 분석 자료를 제공하므로 효율적인 conversion methodology 연구에 도움이 될 것으로 사료된다.

표 4.3. 곡선변환 자료 상태

from	to	Control points/it's number	Degree	Weights	Knots
Bézier	rational Bézier	○	default	default	
	B-spline	○	△		Quasi
	NURBS	○	△	default	Quasi
RBézier	Bézier	○	default	loss	
	B-spline	○	△	loss	Quasi
	NURBS	○	△	○	Quasi
B-spline	Bézier	○	default		loss
	rational Bézier	○	default	default	loss
	NURBS	○	○	default	○
NURBS	Bézier	○	default	loss	loss
	rational Bézier	○	default	○	loss
	B-spline	○	○	loss	○

Legend ○:전송되는 자료, default:기본값으로 생성,
 △:전송되거나 선택가능,
 loss:상실, Quasi:Quasi-uniform 생성

향후 연구 과제로는, degree reduction과 degree elevation 방법들, 그리고 knot insertion과 knot deletion등과 같은 더 효율적이고 깊이 있는 근사 교환 (approximate conversion)이나 정확한 교환 연구가 요구된다. 또, 기하학적으로 복잡한 모양의 곡선 및 곡면의 정확한 교환 문제해결이 숙제로 남는다.

참고문헌

- [1] Bardis, L. and Patrikalakis, N.M., Approximate conversion of rational B-spline patches, CAGD6, 189-204, 1989
- [2] Bloor, M.S. and Owen, J., CAD/CAM product-data exchange : the next step, CAD23(4), 237-243, 1991
- [3] Dannenberg, L. and Nowacki, H., Approximate conversion of surface representations with polynomial bases, CAGD2, 123-131, 1985
- [4] Eck, M., Degree reduction of Bézier curves, CAD10, 237-251, 1993
- [5] Eck, M., Least squares degree reduction of Bézier curves, CAD27, 845-851, 1995
- [6] Farin, G., Curves and Surfaces for Computer Aided Geometric Design-A Practical Guide, 3rd Edition, Academic Press, 1993
- [7] Hoelzle, J.E., Knot placement for piecewise polynomial approximation of curves, CAD15(5), 296-296, 1983
- [8] Hoschek, J., Approximate conversion of spline curves, CAGD4, 171-181, 1987
- [9] Hoschek, J.; Schneider, F.J. and Wassum, P., Optimal approximate conversion of spline surfaces, CAGD6, 293-306, 1989
- [10] Howard, T.L.J., et.al., A Introduction to PHIGS and PHIGS PLUS, Addison-Wesley, 1991
- [11] Park, Y.; Choi, U. J. and Kimn, H. J. , Approximate conversion of Bézier curves, Bull. Australian Math. Soc., Vol. 51, No. 1, pp.153-162, 1995
- [12] Pasquill, K.G., Practical uses of IGES, CAD20(6),358-359, 1988
- [13] Patrikalakis, N.M., Approximate conversion of rational splines, CAGD6, 155-165, 1989
- [14] Piegl, L. and Tiller, W., Algorithm for degree reduction of B-spline curves, CAD27(2), 101-110, 1995

- [15] Piegl, L. and Tiller, W., The NURBS book, 2nd ed., Springer, 1997
- [16] Vergeest, J.S.M., CAD surface data exchange using STEP, CAD23(4), 269-281, 1991
- [17] Willson, P.R., A short history of CAD data transfer standards, IEEE CG&A, 64-67, 1987
- [18] 김혁진 외, PC 그래픽스 도구로서의 3중첩 구간적 Bézier 3차 곡선의 구성, 한국정보과학회 논문지20(2), 1993
- [19] ISO TC184/SC4/WG4 Part42:The EXPRESS language reference manual

저자 소개



김혁진
1989년 아주대학교 대학원
전자계산학과
1991-1995년 아주대학교
대학원 컴퓨터공학과
박사과정
1992-1997년 김천대학
사무자동화과 조교수
1997년-현 청운대학교
전자계산학과 전임강사