

## 論 文

大 韓 造 船 學 會 誌  
 第22卷 第3號 1985年 9月  
 Journal of the Society of  
 Naval Architects of Korea  
 Vol. 22, No. 3, September 1985

### B-Spline 을 이용한 선체표면의 표현에 관한 연구

윤명호\* · 서승완\* · 김원돈\* · 김광욱\*

### Representation of Ship Hull Surface Using B-Spline

by

B.H. Yoon\* · S.W. Sur\* · W.D. Kim\* · K.W. Kim\*

#### Abstract

This paper deals with the representation and fairing of ship hull using a B-spline surface which is known as a 3-D model in computer aided free-form design.

The first part of this paper is devoted to the fundamental concepts of B-spline theory and its geometrical characteristics.

In its second part are described the facilities for efficient application, fast rendering of B-spline surface, and intersection of B-spline surfaces.

The examples are presented with actual application to a real ship in the last part.

And also a new concept of evaluating the surface fairness by the Gaussian curvature isolines is briefly discussed and its examples are presented.

#### 1. 서 론

1915년 Taylor가 그의 standard ship form을 개발하기 위해 W.L.과 frame을 4차 및 5차의 다항식으로 표시하여 사용한 이후 선체표면을 수학적으로 표현하는 연구는 꾸준히 지속되어 왔다. 특히 전산기의 출현과 아울러 이러한 노력은 더욱 가속되어 이들을 이용한 많은 선박설계용 program package들이 개발되어 사용되어 오고 있다.

이러한 일련의 연구는 선체를 표현하는 방법으로서 3개의 직교면에서의 단면곡선들을 정의하여 이를 곡선들로서 선체표면을 근사하게 표현하는 방법(surface

approximation)이었다. 이 방법은 3차원의 선체를 2차원의 곡선으로서 정의하므로 이미 이러한 방법에 익숙해진 설계자에게는 사용하거나 이해하기가 쉽고, 지금까지 개발된 많은 이론들을 쉽게 이용할 수 있다는 장점이 있다.

그러나 종래의 이러한 방법으로는 순정한 표면(smooth surface)을 얻기 위해 소위 'Cross-Fairing'이라는 단순하고도 지루한 작업을 거쳐야 할뿐 아니라 이 방법은 선체표면의 근사한 표현일 뿐이지 수학적 의미에서 선체의 정확한 표현은 아니라는 본질적 취약점을 내포하므로 특히 선수미부의 곡이 심한 부분 등에서는 선체표면의 정확한 경의가 어렵다는 단점을 가진다.

따라서 최근에 와서는 CAGD(Computer Aided Geo-

\* 接受日字：1985年 3月 25日，再接受日字：1985年 6月 7日

\* 正會員，韓國機械研究所 大德船舶分所

metric Design) 및 computer graphics 등의 관련기술의 급속한 발달과 아울러 이를 취약점을 해결하기 위한 시도로서 선체 표면을 곡선이 아닌 곡면으로서 직접 정의하려는 연구가 활발히 진행되고 있다.<sup>1,2,3,4,5,6,7)</sup>

이는 종래의 방법과는 달리 선체표면을 완전한 수식을 사용하여 정의하므로 보다 정확하게 정의할 수 있으며, 이들 수식을 이용하여 선체표면의 여러가지 기하학적 성질들을 (surface normal, curvature 등) 쉽게 계산할 수 있으므로 설계단계에서 필요한 유체역학적인 계산이나 유체동역학적인 해석등을 수행할 수 있다는 장점이 있다.<sup>(8)</sup>

또한 선체가 3차원으로 직접 정의되므로 cross-fairing의 필요성을 제거할 수 있을 뿐만 아니라 순정도 (fairness)의 검토를 위한 여러가지 수학적 방법의 폐 어링을 행 할수도 있다. (1, 5, 8, 9)

일반적으로 임의의 곡면을 수학적으로 표현하기 위한 이론으로는 1967년에 발표된 Coons Patch 곡면 이후 Bézier 곡면, B-spline 곡면 등의 많은 이론이 있으며, 이들은 각자 특정분야에 응용되어 널리 사용되고 있다.<sup>(10)</sup> 그러나 어느 경우이든 이들이 효과적으로 사용되기 위해서는 다음과 같은 몇가지 요구를 만족시켜야 한다.<sup>(11)</sup>

—간단한 수학적 표현으로 곡선과 곡면을 정의할 수 있어야 한다.

—Smoothness를 만족시키기 위한 충분한 연속성을 가져야 한다.

—다양한 형상의 geometry를 표현할 수 있어야 한다.

— 표현된 geometry를 쉽게 수정할 수 있어야 한다.

본 논문에서는 위와 같은 관점 하에 Coons Patch 곡면, Bézier 곡면, B-spline 곡면에 대한 타당성 검토를 수행하였으며, 그 결과 선체를 표현하기 위한 최적의 이론으로서 B-spline 곡면을택하였다. (Table 1 참조)

따라서 본고는 먼저 B-spline 곡면의 이론 및 여러 가지 특징들에 대해 고찰하고, 이를 효과적으로 응용하기 위해 필요한 여러가지 기능들에 대해 기술하였다. 또 이들을 실제 선체표면의 정의에 적용시키는 방법 및 적용시킨 결과를 제시하고 선체표면 각 점에서의 Gauss 곡률을 계산하여 이를 정면도 상에 등곡률도로 표시함으로서 정의된 선체의 fairness를 확인해 볼 수 있는 방법을 제시하였다.

## 2. B-Spline 곡면

$u$ ,  $v$ 의 2 변수방향(parametric direction)으로 각각  $(m+1)$ ,  $(n+1)$ 개의 control vertex를 가지는 B-spline 곡면상의 한점  $P(u, v)$ 는 control vertex  $V_{ij}$ 와 B-spline 기분합수  $N_{iM}, N_{jN}$ 에 의해 정의된다.

$$P(u, v) = \sum_{i=0}^m \sum_{j=0}^n V_{ij} \cdot N_{iM}(u) \cdot N_{jN}(v) \quad (1)$$

또 knot vector  $[t_i, t_{i+1}, \dots, t_{i+k}]$ 를 가지는  $(k-1)$  차(order  $k$ )의 B-spline 기본함수  $N_{ik}(t)$ 는 다음과 같은 순환식(recursive formula)에 의해 결정된다. <sup>(12)</sup>

$$N_{i+1}(t) = \begin{cases} 1, & \text{for } t_i \leq t \leq t_{i+1} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

**Table 1.** Comparison of various surface techniques

	장 점	단 점
Coons Patch surface	<ul style="list-style-type: none"> <li>Mathematical simplicity</li> </ul>	<ul style="list-style-type: none"> <li>Patch Corner에 시의 tangent vector 및 twist vector 가 요구됨</li> <li>Local control이 어려움</li> <li>불연속면의 표현이 불가능</li> </ul>
Bézier surface	<ul style="list-style-type: none"> <li>곡면의 형상을 쉽게 예측할 수 있다.</li> <li>충분한 smoothness를 가진다.</li> </ul>	<ul style="list-style-type: none"> <li>Interpolation이 불가능</li> <li>Local control이 어려움</li> <li>Control vertex의 갯수가 증가함에 따라 order가 증가해야 한다.</li> <li>불연속면의 표현이 불가능</li> </ul>
B-Spline surface	<ul style="list-style-type: none"> <li>곡면의 형상을 쉽게 예측할 수 있다.</li> <li>Local control이 가능</li> <li>불연속면의 정의가 가능</li> <li>Interpolation이 가능</li> <li>충분한 smoothness를 가진다.</li> </ul>	<ul style="list-style-type: none"> <li>Mathematical complexity</li> </ul>

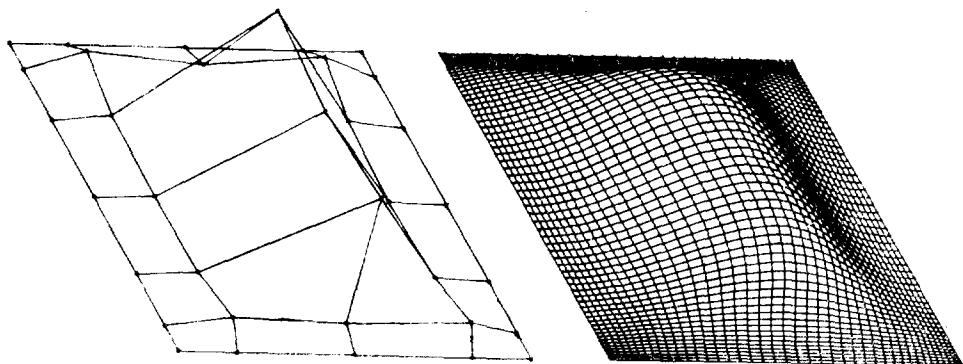


Fig. 1 B-spline surface and control vertex

$$N_{i,k}(t) = \frac{t - t_i}{t_{i+k-1} - t_i} \cdot N_{i,k-1}(t) + \\ \frac{t_{i+k} - t}{t_{i+k} - t_{i+1}} \cdot N_{i+1,k-1}(t), \text{ for } k > 1$$

이상과 같이 정의되는 B-spline 곡면은 다음과 같은 특성을 갖는다.

(1) B-spline 곡면은 2변수방향으로 각각  $(m-1)$ ,  $(n-1)$ 의 차수를 갖는 부분 다항식(piecewise polynomial)으로 정의되므로 각 방향으로  $(m-2)$ ,  $(n-2)$  차의 연속성을 갖는다. (Fig. 1)

(2) 식 ②에서 보듯이 B-spline 기본함수의 차수는 control vertex의 개수와 무관하므로 복잡한 형상의 도

형도 저차수의 B-spline 곡면으로 정의할 수 있다. (Fig. 2)

(3) B-spline 기본함수는 ‘Local Support’의 특징을 가지므로 도형의 수정을 용이하게 할 수 있다. 즉, 식 ②에서 보듯이 B-spline 기본함수  $N_{i,k}(t)$ 는  $k$ 개의 구간에서만 값을 가지고 나머지 구간에서는 0이므로 곡선의 경우 1 vertex를 변화시키면  $k$ 개의 곡선 구간에만 영향을 미치고, 나머지 구간에 대해서는 영향을 미치지 않는다. (Fig. 3) 곡면의 경우도 마찬가지이다.

(4) 각종 control vertex를 정의함으로서 너클을 가지는 불연속곡면을 정의할 수 있다. (Fig. 4)

(5) 식 ①에 의해 정의된 곡면은 Fig. 1에서 보는

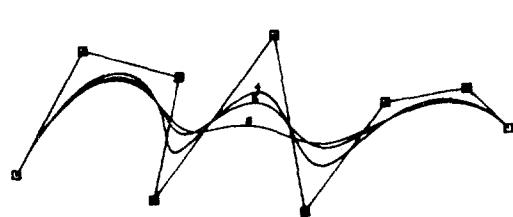


Fig. 2 B-spline curves of order 4, 5, 6

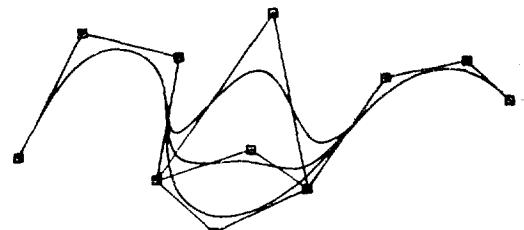


Fig. 3 Local control of B-spline curves

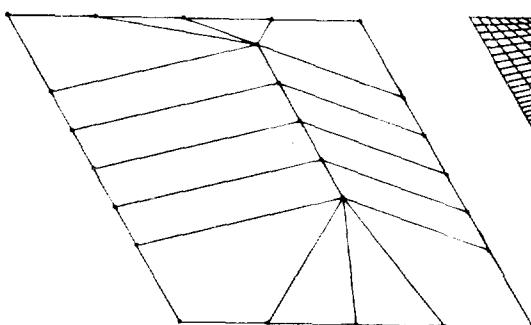


Fig. 4 B-spline surface with hard chine

바와 같이 control vertex에 의해 만들어지는 다면체에 convex한 성질을 가지므로 control vertex에 의해 정의되는 곡면의 형상을 쉽게 예측할 수 있다.

### 3. 효과적 응용을 위한 기능

#### 3.1. 곡면분할에 의한 곡면의 계산

대화식 작업방법에 의해 곡면을 형성하고 이를 도면으로 확인하고 필요한 수정을 하기 위해서는 빠른 시간내에 곡면을 계산하고 이를 화면상에 그려볼 수 있어야 한다.

텐서곱 곡면인 B-spline 곡면상의 점들을 계산하여 이를 디스플레이하기 위한 방법으로서 곡면분할(patch subdivision) 방법을 택하였다.

이에 대한 연구는 Lane, Riesenfeld, Cohen, 그리고 Catmull 등에 의해서 계속 진행되어 왔다. 이중에서 Catmull<sup>(13)</sup>이 제시한 방법이 계산시간이 가장 적게 걸리는 알고리즘이지만, 곡면이 반드시 3차의 균등 B-spline이어야 한다는 제한조건이 있다.

또 Cohen<sup>(14)</sup>이 제시한 Oslo-알고리즘은 일반적인 형내인 불균등(non-uniform) B-spline에 적용할 수 있다 는 장점을 가지나 3차원 곡면의 분할에 대한 명확한 알고리즘을 제시하지 못하였다.

따라서 본 연구에서는 Oslo-알고리즘을 기초로 하여 다음과 같은 기능을 추가한 새로운 알고리즘을 개발하였다.

##### —3차원 곡면으로의 확장

—분할깊이(subdivision depth)에 따른 Knot vector를 자동적으로 계산하는 기능을 추가함으로서 새로운 Knot vector를 지정해 주어야 하는 번거로움을 해결 —B-spline 곡면의 교차문제를 해결하기 위해 B-spline 곡면을 Bézier 곡면으로 변환할 수 있는 기능

불균등 B-spline 곡선의 분할을 하기 위해서는 해당되는 변수 구간내에 새로운 Knot들을 추가함으로서 새로운 control vertex 및 B-spline 기본함수(discrete B-spline)가 정의되어 본래의 곡선과 동일한 B-spline 곡선이 정의된다.

이 과정을 반복하면 새로이 생성되는 control vertex에 의한 convex hull은 B-spline 곡선에 접근하게 된다.

2차원의 경우를 확장하여 3차원 곡면에 적용하면, 여러개의 분할된 곡면에 의해서 B-spline 곡면이 정의된다.

이와같은 곡면분할에 의해서 곡면을 계산하여 디스플레이하는 방법은 앞으로 추가되어야 할 비가시선 및 비가시면 제거를(hidden line/surface elimination) 효과적으로 처리할 수 있는 기초가 된다.

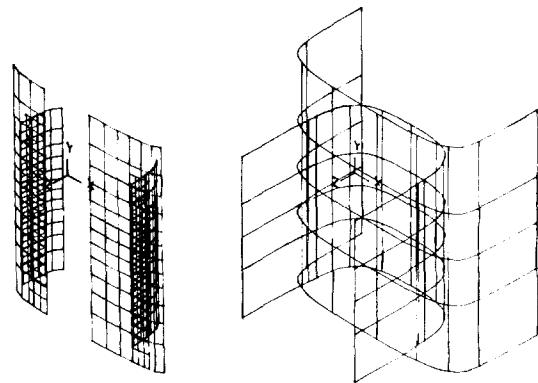


Fig. 5 Intersection of 2 B-spline surfaces

#### 3.2. B-spline 곡면의 교차(Intersection)

선체표면의 정의가 끝난후에 일어지는 W.L., Body-plan 등의 단면도는 선체와 평면간의 교차에 의해서 구해진다. 본 연구에서는 이와같은 교차문제를 해결하고자 두 개의 B-spline 곡면간의 교차선을 찾아내는 방법을 제시하고 이를 프로그램화 하였다. 일반적으로 교차의 문제는 수치적 방법(numerical approach)이나 기하학적 방법(geometrical approach)에 의해서 해결할 수 있다. <sup>(15, 16)</sup> 수치적 방법은 Newton-Raphson 방식에 의한 비선형 연립방정식의 해를 구하는 것과 최소자승법에 의한 최소거리를 구하는 문제로 귀착되며 계산상의 어려움과 함께 완전한 교차를 찾지 못한다는 결점을 갖고 있다.

본 연구에서는 기하학적 방법의 일종인 곡면분할에 의한 ‘divide-and-conquer’ 형태의 방법을 채택하여 빠른 시간내에 교차를 찾을 수 있는 다음과 같은 알고리즘을 개발하였다. (Fig. 5)

(교차 알고리즘의 개요)

- 1단계 : Separability test를 줄이기 위해 B-spline 곡면을 Bézier 곡면으로 변환한다.
- 2단계 : Bézier 형태로 변환된 각 patch들의 minimax-box간에 separability test를 수행함으로서 교차점으로 접근해 간다.
- 3단계 : 교차하는 patch들을 분할하여 quad-tree의 형태로 표현한다.
- 4단계 : 2단계와 3단계를 반복수행하여 box의 두께가 충분히 작아지면 이를 사면형의 평면으로 간주하여 평면간의 교차를 구한다.

### 4. 선체표면에의 적용

일반적으로 곡면을 이용하여 선체를 정의하는 방법

으로는 2가지를 생각할 수 있다. <sup>(4)</sup> 첫째는 선체를 여러개의 작은 곡면조각(patch)의 조합으로 간주하여 각각을 정의한 후 이들을 완전한 선체로 합성하는 방법으로 합성할 때 충분한 연속성을 갖도록 해야한다는 어려움이 있다. 둘째는 선체를 연속성을 갖도록 합성된 하나의 곡면(composite surface)으로 정의하는 방법으로 전자의 방법에 비해 부분적인 수정은 용이치 않으나 합성의 필요가 없어 간단하다는 장점이 있다.

본 연구에서는 후자의 방법을 택하였으며 2변수방향으로 각각 차수가 3인 bicubic B-spline 곡면을 사용함으로서 초기 control vertex를 구하기 위한 보간법의 사용을 가능하게 하였다.

#### 4.1. 초기 Control vertex의 결정

초기 lines 및 offset를 가지고 선체표면을 정의하고자 할 때 최초의 control vertex를 구하는 문제는 아주 중요하다. 즉, 오프셋포인트를 지나고 원하는 선체에 근사하는 선체를 결정할 수 있는 control vertex를 구할 수 있다면 이를 수정하여 선체의 모양을 변화시킴으로서 원하는 선체를 정의할 수 있다.

이 문제를 해결하는 방법으로는 2가지의 방법을 생각할 수 있다. 첫번째는  $u$ ,  $v$ 의 2변수평면상에 평행한 control vertex 조합을 배치한 후 이들을 화면상에서 변화시켜 원하는 control vertex가 얻어질 때까지 반복하는 방법으로서 <sup>(4)</sup> 3차원의 control vertex를 2차원 평면상에서 처리해야 하므로 조작하기가 힘들다는 단점이 있다.

다른 하나의 방법은 주어진 오프셋 포인트를 보간하

는 control vertex를 수치적으로 구하는 방법으로 이는 결국 ①식의 역행렬을 구하는 문제로 귀착된다. 이 문제를 효율적으로 해결하는 기법은 몇 가지가 있으나 어느것을 사용하든 너클이 존재하는 불연속면을 처리할 수 없다는 단점이 있다.

본 논문에서는 기본적으로 후자의 방법을 택하여 'Barsky'와 'Greenberg'가 제시한 방법을<sup>(17, 18)</sup> 이용하여 초기 control vertex를 구하였으며 이의 단점을 해결하기 위한 방법으로서 곡면접합(surface blending)의 기법을 사용하였다.

즉, 너클라인이 있는 선체의 경우 너클라인을 경계로 선체표면을 나누어 각각의 선체에 대해 control vertex를 구한 후 이들을 접합함으로서 완전한 control vertex를 구하였다.

따라서 이상과 같은 과정을 위해서 기본적으로는 W.L.이 입력되고 중앙부 부근의 선저부에서 발생하기 쉬운 변곡의 경향을 최소화 하기 위해 bottom tangent line이 입력된다.

#### 4.2. 선체표면의 수정

B-spline곡면의 모양을 결정짓는 것은 control vertex이다. 따라서 4.1의 과정을 거쳐 결정된 control vertex로 만들어진 선체의 수정은 control vertex의 수정을 통하여 이루어질 수 있다. 그러나 이들 점들은 3차원 공간상의 점이므로 이들을 3차원 공간에서 직접 수정하기란 어려운 일이다.

그러므로 이들을  $xy$ ,  $yz$ ,  $zx$ 의 3직교면에 투영시킨 후 2차원 평면에서 수정함으로서 선체곡면의 수정을

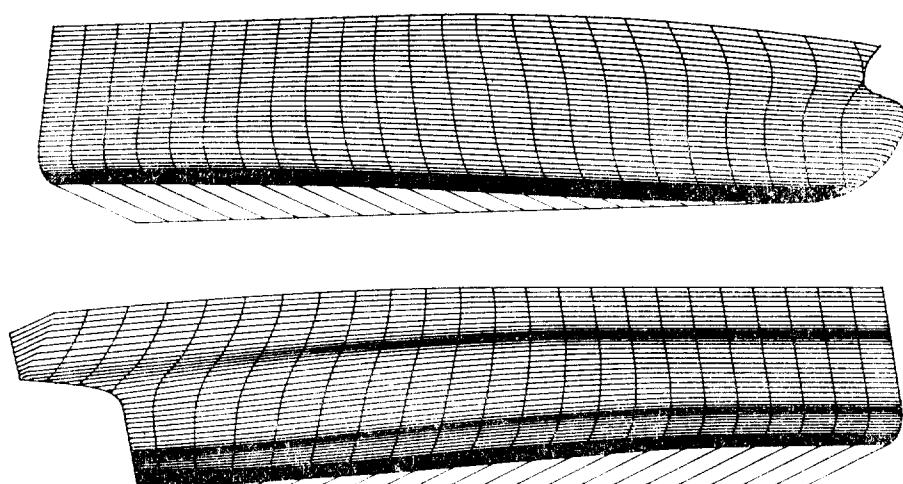


Fig. 6 Isometric view of B-spline hull surface

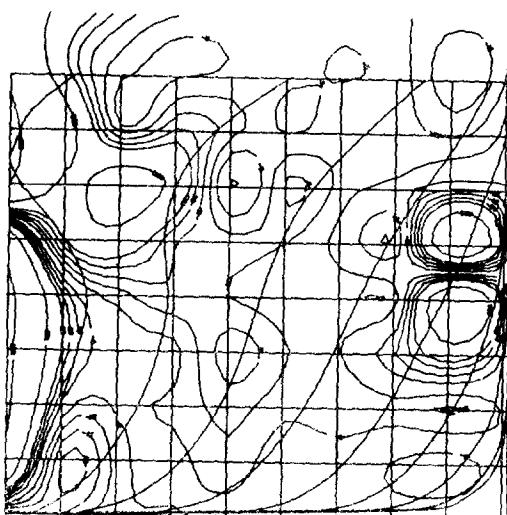


Fig. 7a

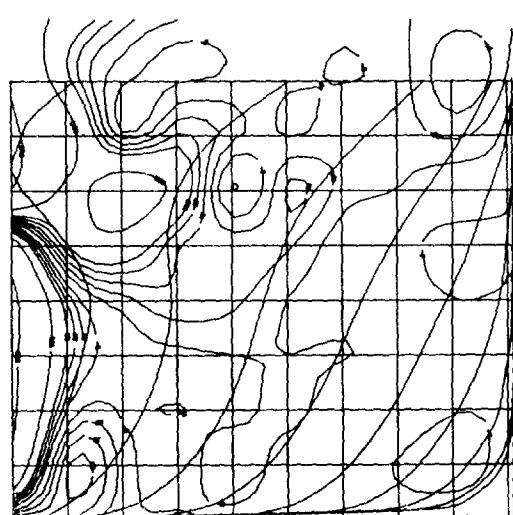
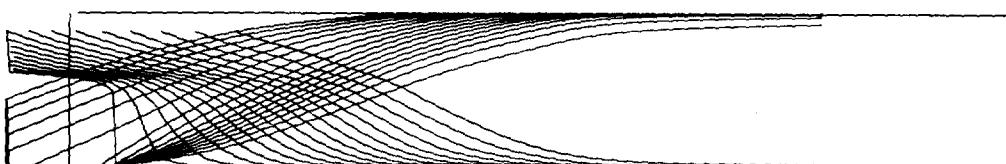
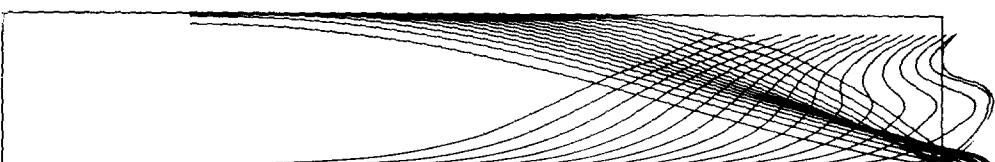
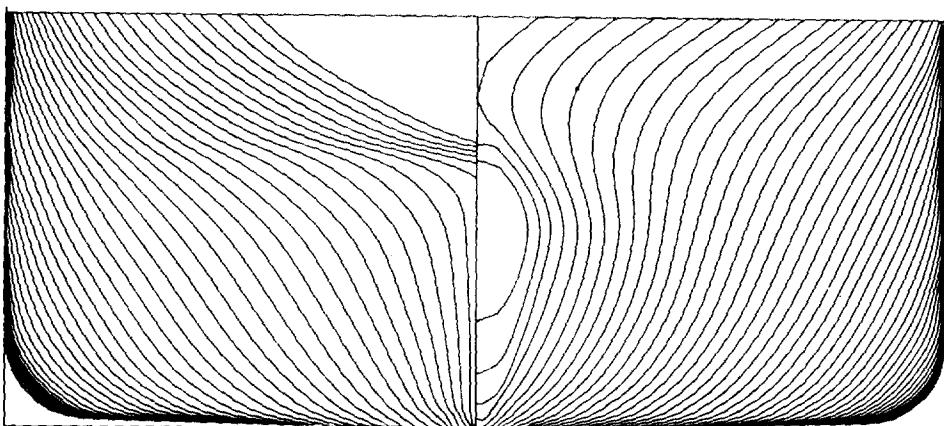


Fig. 7b

**Fig. 7** Gaussian curvature isolines

- a) With unfair regions near middlepart
- b) After fairing

**Fig. 8** Lines plan from B-spline hull surface

행 할 수 있다.

#### 4.3. Fairness의 검토

정의된 선체표면의 fairness를 검토하기 위한 방법으로 'Gauss 곡률'이 사용된다. 이는 곡면상의 한점에서의 최대곡률과 최소곡률의 합으로 표시되며, 그점에서의 곡의 정도를 나타내는 척도로 사용된다.<sup>(10)</sup> 즉

$$\text{Gauss 곡률} : K = \frac{L \cdot N - M^2}{E \cdot G - F^2} \quad (3)$$

여기에서

$$E = \left( \frac{\partial x}{\partial u} \right)^2 + \left( \frac{\partial y}{\partial u} \right)^2 + \left( \frac{\partial z}{\partial u} \right)^2$$

$$G = \left( \frac{\partial x}{\partial v} \right)^2 + \left( \frac{\partial y}{\partial v} \right)^2 + \left( \frac{\partial z}{\partial v} \right)^2$$

$$F = \frac{\partial x}{\partial u} \cdot \frac{\partial x}{\partial v} + \frac{\partial y}{\partial u} \cdot \frac{\partial y}{\partial v} + \frac{\partial z}{\partial u} \cdot \frac{\partial z}{\partial v}$$

$$L = X \cdot \frac{\partial^2 x}{\partial u^2} + Y \cdot \frac{\partial^2 y}{\partial u^2} + Z \cdot \frac{\partial^2 z}{\partial u^2}$$

$$N = X \cdot \frac{\partial^2 x}{\partial v^2} + Y \cdot \frac{\partial^2 y}{\partial v^2} + Z \cdot \frac{\partial^2 z}{\partial v^2}$$

$$M = X \cdot \frac{\partial^2 x}{\partial u \cdot \partial v} + Y \cdot \frac{\partial^2 y}{\partial u \cdot \partial v} + Z \cdot \frac{\partial^2 z}{\partial u \cdot \partial v}$$

$X, Y, Z$ : Direction cosine of the normal to surface  
따라서 선체표면 혹은 정면도 상에서 동일 Gauss 곡률을 갖는 절을 연결한 '등 Gauss 곡률도'를 그려봄으로서 선체표면의 fairness를 확인해 볼 수 있다.  
(Fig. 7)

#### 4.4. 단면도 및 offset

최종적인 선체표면의 정의가 끝난 후 일어지는 W.L., bodyplan, B.T. 등의 단면도는 정의된 선체와 평면간의 교차에 의해 구해진다.

또 선체 표면상의 모든점에 대한 offset 및 normal vector의 계산도 가능하므로 이를 이용한 선박의 제작 단계를 수행할 수 있다.

### 5. 결 론

B-spline 곡면을 이용한 수학적 선체정의 방법이 제시되었다. 이는 종래의 방법과는 달리 선체를 완전한 수식으로 정의함으로서 선수미등의 폭이 심한 부분도 정확하게 정의할 수 있으며 cross-fairing의 필요성을 제거하므로 초기설계시 발생하는 lines의 수정을 빠르고 용이하게 처리할 수 있다.

또한 입의의 선체표면에서의 기하학적 성질들(surface normal, curvature)을 간단히 계산할 수 있으므로 이를 선체저항, 선체운동등의 유체동력학적인 해석의 자료로 사용할 수 있다.

나아가서 본고는 정의된 선체의 fairness를 검토하기 위하여 Gauss 곡률을 이용한 새로운 개념의 폐어링 기법을 제시함으로서 수학적 방법의 폐어링에 대한 가능성을 확인하였다.

앞으로 CAD/CAM, computer graphics등의 관련기술을 토대로 본 연구의 결과를 확장시키기 위한 많은 노력이 계속되어야 할 것으로 믿는다.

### 참 고 문 헌

- [1] Izumida, K. and Matida, Y., "Ship hull definition by Surface Techniques for Production Use", ICCAS 79, 1979.
- [2] Munchmeyer, F.C. and Schubert, C. and Nowachi, H., "Interactive Design of Fair Hull Surfaces Using B-splines", ICCAS 79, 1979.
- [3] Rogers, D.F. and Satterfield, S.G., "Dynamic B-spline Surfaces", ICCAS 82, 1982.
- [4] Rogers, D.F. and Satterfield, S.G. and Rodriguez, F.A., "Ship Hulls, B-spline Surfaces, and CAD/CAM", IEEE CG&A, 1983.
- [5] Munchmeyer, F.C., "Mathematical Ship Lines and Surfaces", Marine Technology, Vol. 19, No. 3, 1982.
- [6] Fog, N.G., "Creative Definition and Fairing of Ship Hulls Using B-spline Surface" CAD, Vol. 16, No. 4, 1984.
- [7] Catley, D. and Okan, M.B. and Whitley, Cl, "Unique Mathematical Definition of a Hull Surface, its Manipulation and Interrogation", WEMT 84, 1984.
- [8] Odabasi, A.Y. and Fitzsimmons, P.A. and Wright, B.D., "A System Hydrodynamic Assessment and Design System for Single Screw Ship Forms", PRADS 83, 1983.
- [9] Dills, J.C. and Rogers, D.F., "Color Graphics and Ship Hull Surface Curvature", ICCAS 82, 1982.
- [10] Faux, I.D. and Pratt, J.J., "Computational Geometry For Design and Manufacture", Ellis Horwood Limited, 1981.
- [11] Barnhill, R.E. and Boehm, W., "Surfaces in Computer Aided Geometric Design", North-Holland Publishing Co., 1983.
- [12] Barnhill, R.E. and Riesenfeld, R.F. (eds), "Com-

- puter Aided Geometric Design”, Academic Press, New York, 1974.
- [13] Catmull, E. and Clark, J., “Recursively Generated B-spline Surfaces on Arbitrary Topological Meshes”, CAD, Vol. 10, No. 6, 1978.
- [14] Cohen, E. and Lyche, T. and Riesenfeld, R.F., “Discrete B-spline and Subdivision Techniques in Computer-Aided Geometric Design and Computer Graphics”, Computer Graphics and Image Processing, Vol. 14, No. 2, 1980.
- [15] Peng, Q.S., “An Algorithm for Finding the Intersection Lines Between two B-spline Surfaces”, CAD, Vol. 16, No. 3, 1982.
- [16] Carlson, W.E., “An Algorithm and Data Structure for 3D Object Synthesis Using Surface Patch Intersections”, Computer Graphics, Vol. 16, No. 3, 1982.
- [17] Barsky, B.A. and Greenberg, D.P., “Determining a Set of B-spline Control Vertices to Generate an Interpolating Surface”, Computer Graphics and Image Processing, Vol. 14, 1980.
- [18] Wu, S.C. and Abel, J.F. and Greenberg, D.P., “An Interactive Computer Graphics Approach to Surface Representation”, Communications of ACM, Vol. 20, N. 10, 1977.
- [19] Nuchmeyer, F.C., “Application of Differential Geometry to Ship Design”, ICCAS 82, 1982.
- [20] Foley, J.D. and Van Dam, A., “Fundamentals of Interactive Computer Graphics”, Addison-Wesley Publishing Company, 1982.
- [21] Rogers, D.F. and Adams, J.A., “Mathematical Elements For Computer Graphics”, McGraw-Hill, New York, 1976.