

---

 論 文
 

---

大韓造船學會論文集  
 第 29 卷 第 3 號 1992 年 8 月  
 Transactions of the Society of  
 Naval Architects of Korea  
 Vol. 29, No.3, August 1992

## 컴퓨터 그래픽스 기법을 이용한 선체곡면 가시화 연구

신현경\*, 박규원\*\*

### A Study on the Visualization of Ship Hull using Computer Graphics Techniques

by

H. Shin\* and K.W.Park\*\*

#### 요 약

이 논문에서는 화가의(Painter's) 알고리즘을 사용하여 3차원 자유형상곡면(Three dimensional free form surface)을 가시화하는 방법을 개발하였고, 특히 선체형상은 Open 균일 B-Spline 곡면으로 정의하였다. 컴퓨터그래픽스를 응용하여 와이어프레임, 숨은면제거, 묘영등의 가시화를 위한 프로그램을 완성하였다. 또한 선체곡면의 순정을 검토하기 위해 Gaussian 곡률값을 15등급으로 구분하여 등급별 색(color)처리 하였다.

#### Abstract

This paper outlines the methods of visualizing 3-dimensional free form surfaces employing the Painter's algorithm, especially for the ship hull forms which are defined as open uniform Bi-cubic B-spline surfaces. The computer graphic codes are developed for the transparent wire-frame, the hidden surface removal and the shading visualization techniques, The codes are applied to the ship hull 3-dimensional surface visualization and the color graphic figures are displayed. Also Gaussian curvature is displayed on the color plots of the isoparametric net of the ship hull surface.

#### 1. 서 론

국내 조선소에서의 응용과 컴퓨터 그래픽스를 이  
 용한 울산대학교 전산기지원 선박설계 교과과정을

위하여, 선각곡면의 수식화가 시도되어 1차적인 프로  
 그램을 이미 개발 발표하였다[25][26]. 대한조선학  
 회 '91춘계 연구발표회에서 발표된 "선체형상을 위한  
 기하학적 모델링 연구(I)"에서는 Open knot

발표 : 1991년도 대한조선학회 추계연구발표회('91.11.16)

접수일자 : 1991년 12월 16일, 재접수일자 : 1992년 6월 1일

\* 정희원, 울산대학교 조선 및 해양공학과

\*\* 부산수산대학교 해양산업개발연구소

vectors를 이용한 Bi-cubic B-spline 곡면기법을 사용하여 선체형상을 정의하고 투영와이어 프레임(Transparent wire-frame)방법으로 그것을 가시화 하였으나, 선체의 3차원 자유곡면(Free form surface)을 좀더 효율적으로 가시화하기 위한 그래픽 방법이 이 논문에서 시도되어 그래픽 프로그램의 개발을 완료하였다.

이 논문은 크게 6개절로 구성되며, 2절에서는 “선체형상을 위한 기하학적 모델링 연구( I )”[25]의 내용을 요약하였고, 3절에서는 일반적인 3차원 형상의 가시화에 대하여 소개하였으며, 4절에서는 Painter’s 알고리즘을 사용하여 숨은 면제거, 묘영 및 Gaussian 곡률을 처리하는 방법과 비교적 단순한 3차원 곡면에 대한 응용예를 보였으며 5절에서는 완성된 그래픽 프로그램의 선체형상에 대한 응용을 보였다.

2. 선체곡면의 정의

2.1 B-spline 곡면과 Basis functions

B-spline 곡면의 임의점은 n+1, m+1개의 defining polygon net vertex와 기본함수(Basis functions)의 Cartesian 곱으로 정의된다[12][14].

$$Q(u, w) = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} B_{i,j} N_{i,k}(u) M_{j,l}(w) \quad (1)$$

$$U_{min} \leq u < u_{max}, 2 \leq k \leq n+1$$

$$W_{min} \leq w < w_{max}, 2 \leq l \leq m+1$$

$$N_{i,k}(u) = \begin{cases} 1 & \text{if } x_i \leq u \leq x_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$N_{i,k}(u) = \frac{(u-x_i)N_{i,k-1}(u)}{x_{i+k-1}-x_i} + \frac{(x_{i+k}-u)N_{i+1,k-1}(u)}{x_{i+k}-x_{i+1}}$$

$$M_{j,l}(w) = \begin{cases} 1 & \text{if } y_j \leq w \leq y_{j+1} \\ 0 & \text{otherwise} \end{cases}$$

$$M_{j,l}(w) = \frac{(w-y_j)M_{j,l-1}(w)}{y_{j+l-1}-y_j} + \frac{(y_{j+1}-w)M_{j+1,l-1}(w)}{y_{j+1}-y_{j+1}} \quad (3)$$

여기서 Q는 곡면위의 점이고, B는 곡면 결정을 위한 Vertices이다.

2.2 Vertex 산출 - Pseudoinverse Matrix 방법

Offset으로부터 선체를 표현하기 위해서는 생성된 곡면이 offset을 지나가도록 하는 vertex들을 산출해야 한다. 그러나 [Q]가 정방행렬이 아닌 경우에는 pseudoinverse matrix 방법을 사용해야 한다[14][25][26].

\* B-spline 곡면식의 Matrix 형태는

$$r[Q] = r[N]_{n+1} [B]_{m+1} [M]_s \quad \text{이다.} \quad (4)$$

\* Vertices 산출

$$[B] = [N^*]^{-1} [N]^T [Q] [M]^T [M^*]^{-1} \quad (5)$$

$$[N^*] = [N]^T [N], [M^*] = [M][M]^T$$

[N\*]<sup>-1</sup> : pseudoinverse matrix of [N]

[M\*]<sup>-1</sup> : pseudoinverse matrix of [M]

2.3 선체 곡면 모델링

선체를 곡면으로 표현하기 위해서는 같은 숫자의 격자모양의 좌표가 필요하며 이를 위해서 주어진 선형의 offset과 선도로부터 Grid offset을 산출하였다. 이 offset을 사용하여 2.2절에 제시된 방법으로 vertex를 산출하고 2.1절에 제시된 B-spline곡면 방정식으로부터 선체를 작은 곡면 조각(patch)들로 구성하여 가시화하였다[25][26].

2.4 곡면의 순정 검토

선체를 수식적으로 처리하게되면 곡면의 곡률을 계산하여 순정도를 검사할 수 있다. 일반적으로 선체 곡면의 특성을 잘 나타내 주는 Gaussian곡률이 사용되고 있다[1][4][6][8][25].

$$K = k_{min} \cdot k_{max} \quad \text{여기서 } A = [Q_u \times Q_w] \cdot Q_{uu} \\ = \frac{AC - B^2}{|Q_u \times Q_w|} \quad B = [Q_u \times Q_w] \cdot Q_{uw} \quad (6) \\ C = [Q_u \times Q_w] \cdot Q_{ww}$$

3. 3차원 곡면의 가시화

3차원 물체를 표현하는 기하학적 방법에 따라 투영와이어프레임(Transparent wire-frame), 숨은면 제거(Hidden surface removal)와 묘영(Shading)으로 구분할 수 있다.

현재 상용화되어 있는 CAD시스템들은 이 3가지 방법을 이용하고 있다.

**3.1 투영 와이어 프레임 (Transparent wire-frame)**

투영와이어 프레임은 물체(Object)들을 단순히 직선(line)으로만 연결하여 그리기 때문에 프로그래밍하기가 매우 간단하며, 수행 속도 또한 빨라 흔히 사용하고 있는 방법이지만, 물체가 복잡해짐에 따라 선(line)들의 간섭으로 인해 형상들을 구분하기가 어렵다.

**3.2 숨은면 제거(Hidden surface removal)**

숨은면을 제거하는 방법으로는 플레인 방정식방법(Plane equation method)과 물체-공간 (Object-space)방법이 있다[22].

플레인 방정식법은 어떤 특정한 평면이 어느쪽에 있는가(visibility test)를 결정하고 숨은면의 교차점(intersection point)을 찾아 숨은면을 제거하는 기법이고, 물체-공간법은 모델자체(공간에서의 물체의 병렬)에 기초하여 결정한다. 물체-공간방법이 효율적이지만 각각의 모든 픽셀(pixel)에 대한 시각적 정보를 기억하기 위해서는 많은 양의 메모리를 요구한다. 이 기법을 z-buffer알고리즘이라 부른다[13][20][21][22].

z-buffer 알고리즘은 실패없이 수행되나 메모리 집중과 계산량 증가로 PC에서는 잘 쓰이지 않는다. z-buffer기법을 약간 변형한 것이 Painter's 알고리즘이며 이는 메모리를 적게 필요로 한다[13][20][22].

**3.3 묘영(Shading)**

정확한 묘영처리를 하기 위해서는 광원의 위치와 물체 형상에 따른 빛의 강도(intensity)를 계산하여 명암 처리를 하여야 하기 때문에 물체 형상의 복잡성에 따라 계산 시간이 증가한다.

Artist 입장에서는 보다 사실감 있게 형상을 재현하여야 하기 때문에 계산 시간이 그리 중요하지 않겠지만, 엔지니어나 설계자 입장에서는 단순히 원하는 형상대로 되었는가를 빠른 시간에 확인하는 것이 목적이므로 계산 시간은 매우 중요한 문제이며, 또한 사용중인 PC의 color모니터에서 표현할 수 있는 해상도와 색(color)의 한계도 고려되어야 한다. 따라서 숨은면 제거의 문제를 변형한 형태로 각 가시면을 단순한 색으로 칠함으로써 묘영처리를 하였다.

**4. 화가의 알고리즘**

(Painter's Algorithm or Priority fill)

Painter's 알고리즘은 각각의 가시면(polygon)들

을 관찰자(viewer)로 부터 가장 멀리 있는 것 부터 순서대로 그려나가 가장 가까운 부분을 나중에 그리는 방법으로, 가까이 있는 것이 멀리 있는 것을 덮기(covering)때문에 멀리 있는 가려진 부분을 제거하는 기법이다. 이는 화가들이 canvas에 그림을 그리는 방법과 같다[13][22].

가시면들은 단순히 정렬(sorting)만 하여 가시면을 채우기(filling)때문에 교차문제(Intersection Problem)를 위한 계산이 필요없이 효과적이거나 film recorder나 Pen plotter 등에서 사용할 수가 없다 [13].

**4.1 알고리즘(Algorithm)**

Painter's 알고리즘은 다음과 같다[13]

- a. 가시면을 정렬(sorting)하기 위해 가시면의 평균 깊이(depth)를 구한다. x-y평면의 법선 방향에 관찰자가 놓여 있는 경우 평균깊이(mean depth)의 z좌표는 다음과 같이 계산 된다.

$$Z_a = \frac{1}{n} \sum_{i=1}^n Z_i$$

여기서 n은 가시면의 vertices의 수이다.

- b. 가시면들을 정렬한다.  
(정렬하기 전에 가시면들의 z좌표가 계산되어야 한다.)
- c. 관찰자로부터 가장 먼 부분을 순서대로 그려 나간다.

**4.2 곡면에 대한 응용**

숨은면 제거와 묘영 문제는 다각형(가시면)내부와 경계선의 색을 어떻게 처리하는가에 달려 있으며 다음과 같다[19].

**4.2.1 숨은면 제거(Hidden surface removal)**

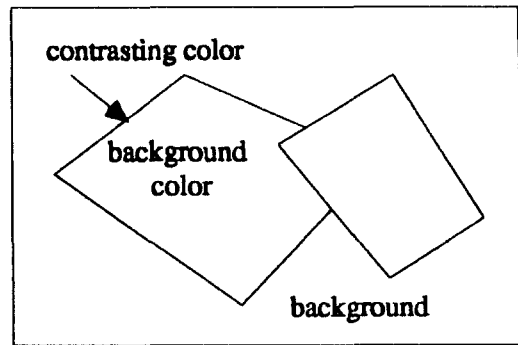


Fig. 1 Hidden surface removal

다각형 내부를 배경색(background color)으로 채운 다음 경계(boundary)선의 색을 배경색의 대조색(contrasting color)으로 그린다(Fig. 1).

4.2.2 묘영(Shading)

다각형 내부를 대조색으로 채운 다음 경계선의 색을 배경색으로 그린다(Fig. 2).

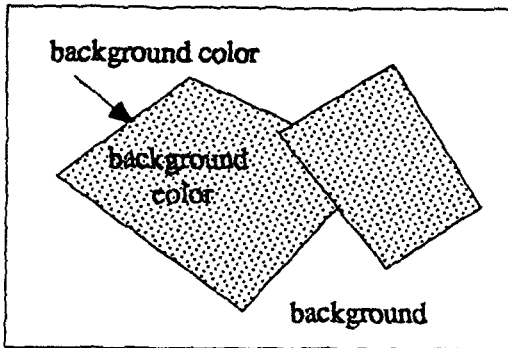


Fig. 2 Shading

4.2.3 Gaussian 곡률

다각형 내부를 Gaussian 곡률에 따라 색깔을 등급하여 다각형 내부를 채우고 경계선을 배경색과 동일한 색으로 칠한다(Fig. 3).

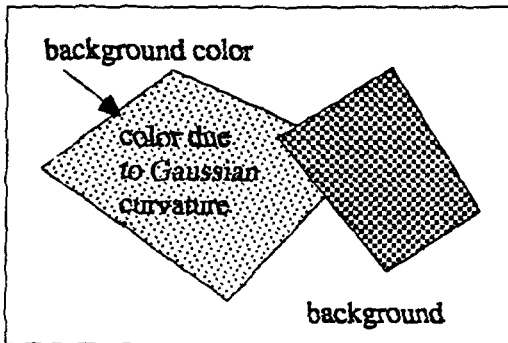


Fig. 3 Color filling due to Gaussian Curvature

4.2.4 단순 3차원 곡면에 대한 응용

이 논문에서는 개발된 3차원 곡면 가시화 그래픽 프로그램 논리의 적합성(validity)을 보여 주기 위해 비교적 단순한 곡면을 사용하여 투영 와이어 프레임, 숨은면 제거 및 묘영에 의한 결과를 비교할 수 있도록 하였다.

Fig. 4는 곡면에 대한 와이어 프레임, Fig. 5는 곡면에 대한 숨은면 제거, Fig. 6는 곡면에 대한 묘영, Fig. 7는 곡면에 대한 Gaussian 곡률이다.

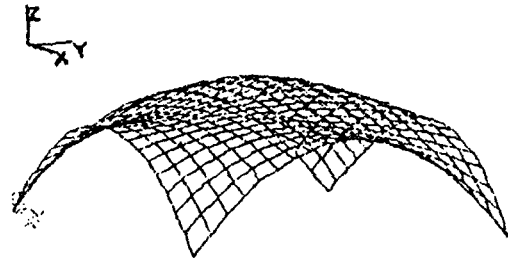


Fig. 4 Transparent wire-frame for surface

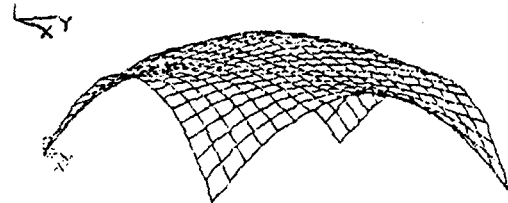


Fig. 5 Hidden surface removal for surface

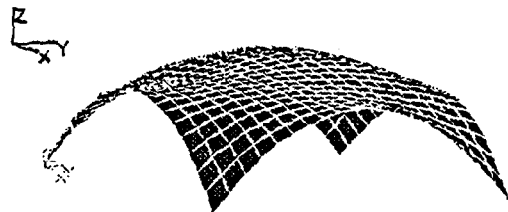


Fig. 6 Shading for surface



Fig. 7 Gaussian Curvature for surface

5. 선체 곡면에 대한 응용

2절에서의 B-spline 곡면생성과 3, 4절에서의 곡면의 가시화를 위해 Micro-Soft Fortran Compiler[27]를 사용하여 Graphic 프로그램을 개발, 선체에 적용하였으며 프로그램의 순서도는 Fig. 8과 같다. 사용된 hardware는 PC 386DX이며 VGA card와 co-processor를 사용하였다. 또한 컬러 프린터로는 해사 기술연구소의 Tecktronix4693DX를 사용하였다.

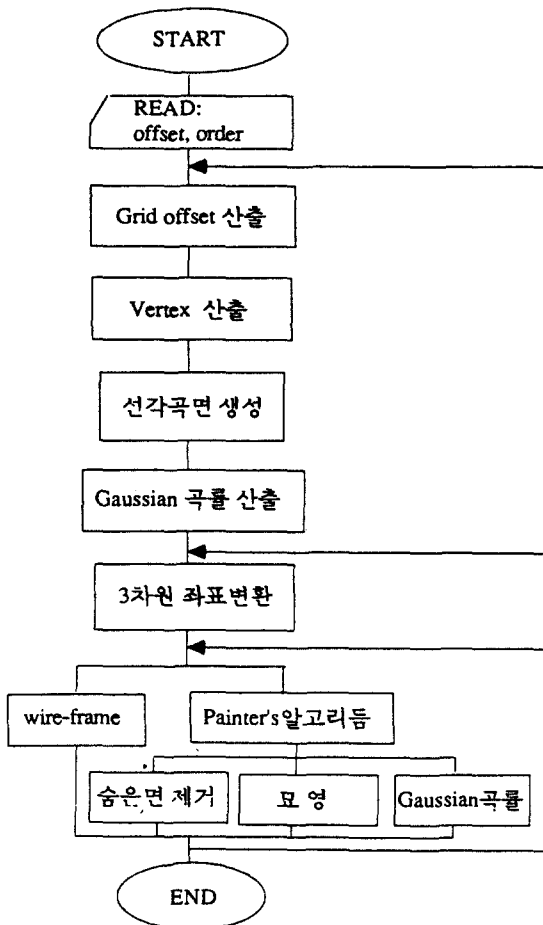


Fig. 8 그래픽 프로그램 순서도

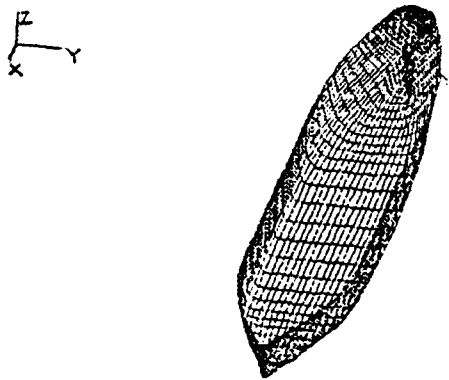


Fig. 9 Transparent wire-frame for ship hull form

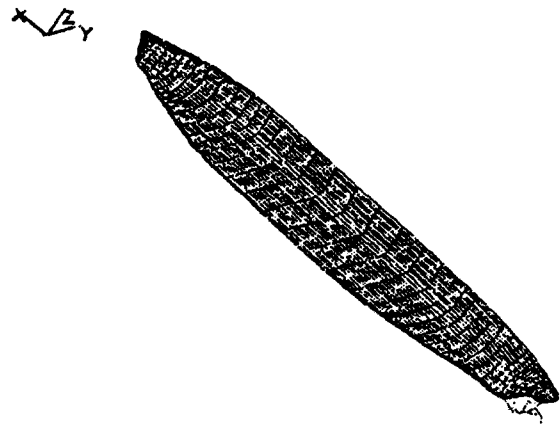


Fig. 10 Hidden surface removal for ship hull form

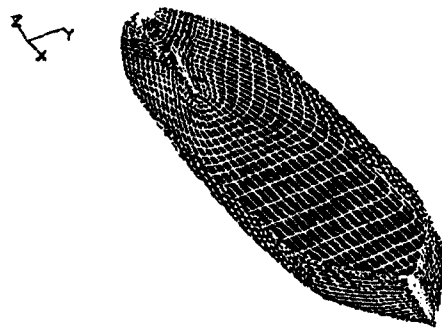


Fig. 11 Shading for ship hull form

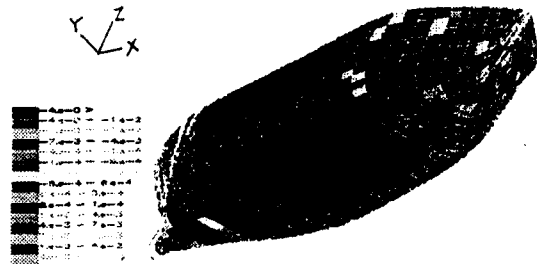


Fig. 12 Gaussian Curvature for ship hull form

Fig. 9는 3.1절의 방법으로 투영 와이어프레임 방법을 시도한 결과이다.

Fig. 10은 4.2.1절의 방법으로 숨은면 제거를 시도하고 있으며

Fig. 11은 4.2.2절의 방법으로 묘영처리를 하였다.

Fig. 12는 선체곡면의 Gaussian곡률 값을 15등급으로 구분하여 등급별 색처리로서 Painter's 알고리즘을 사용하여 가시화하였다.

## 6. 결 론

Open 균일 Bi-cubic B-spline 곡면과 Computer Graphics의 기법을 사용하여, 3차원 선체 곡면 처리를 위한 울산대학교의 독자적인 프로그램을 개발하였고 조선소 및 학부과정에서도 이용이 가능하도록 PC용 프로그램을 완성하였다.

숨은면 제거와 묘영처리를 위해 Painter's 알고리즘을 선체곡면에 응용하여 좋은 결과를 얻었다.

서로 교차되지 않은 곡면들의 경우에 Painter's 알고리즘이 효율적인 방법이라고 생각되나, 서로 교차될 가능성이 있는 곡면을 표현할 때는 정확한 표현이 얻어지지 않을 수 있으므로 다른 알고리즘과 병행하여 사용되어야 할 것이다.

## 7. 후 기

해사기술연구소의 이규열 단장, 한순홍 박사, 강원수, 김준호 연구원 등의 협조로 3차원 선형곡면의 컬러 프린팅이 가능하였고 이에 감사드린다.

## 참 고 문 헌

- [1] 윤병호, 서승환, 김원돈, 김광욱, "B-spline을 이용한 선체표현에 관한 연구", 대한조선학회지 제 22 권 제 3 호, 1985.
- [2] 이규열, 강원수, "선형 변환에 의한 최적 초기 선형 설계 기법에 관한 연구", 대한조선학회지, 제 24 권 제 2 호, 1987.
- [3] 강원수, 이규열, "Rational Cubic Spline을 이용한 선형정의", 대한조선학회 춘계 연구발표집, 1989.
- [4] 김원돈, 남종호, "직접순정법에 의한 초기 선형의 3차원 모델링 프로그램 개발", 해사기술연구소 연구보고서, UCE419-1257. D, 1989.
- [5] 강원수, 이규열, 김용철, "Modified Cubic Spline에 의한 선체형상의 수치적 표현", 대한조선학회지, 제 27 권 제 1 호, 1990.
- [6] 김원돈, 남종호, 김광욱, "선형 모델링을 위한 직접 순정법에 관한 연구", 대한조선학회 춘계 연구 발표회, 1990.
- [7] 강사원, 김수영 "B-spline form parameter방법에 의한 선형 생성", 대한조선학회 추계연구발표회, 1990.
- [8] 김광욱, 김원돈, 남종호, "선체모델링에 있어서 구조면의 정의 및 표현", 대한조선학회 추계연구발표회, 1990.
- [9] Barsky, "Determining a Set of B-spline Control Vertices to Generate an Interpolating Surface", Computer Graphics and image processing 14, 1980.
- [10] Parson, Michael G. "Microcomputer Software for Computer-Aided Ship Design", Marine Technology Vol.24, 1987.
- [11] Welsh, M. Bsc, "Preliminary Ship Design Using a Micro-Based System", North East Coast Institution of Engineer & Shipbuilders, 1987.
- [12] Yamaguchi, "Curves and Surfaces in Computer Aided Geometric Design", Springer-Verlag, 1988.
- [13] Dewey, "Computer Graphic for Engineers", HARDER & ROW, 1988.
- [14] Rogers & Adams, "Mathematical Elements for Computer Graphics", McGRAW-HILL, 1990.
- [15] Roy A., "Computer Graphics", McGRAW HILL, 1986.
- [16] Spath. H. "Spline Algorithms for Curves and Surface", Utilitas Mathematica Publishing Incorporated, 1974.
- [17] Prenter, P.M. "Splines and Variational methods", WILEY, 1975.
- [18] Sunmicro system, "SunCore Reference Manual", 1986.
- [19] Sutherland, Sproull, "Device-Independent GRAPHICS" McGraw Hill, 1985.
- [20] Roger, D.F. "Procedural Element for Computer Graphics", McGraw Hill, 1985.
- [21] Harrington, "Computer Graphics", McGraw Hill, 1991.
- [22] 류성렬, "C Cad 그래픽 프로그래밍", 세웅, 1991.
- [23] 류성렬, "C 애니메이션, 시뮬레이션 프로그래밍" 세웅, 1991.
- [24] Daniel L. Ryan, "Computer Aided Graphics and Design", Marcel Dekker, Inc, 1985.
- [25] 신현경, 박규원, "선체형상을 위한 기하학적 모델링 연구(I)", 대한조선학회 춘계 연구 발표회, 1991.
- [26] 신현경, 박규원, "Open 균일 B-spline곡면을 이용한 선체곡면 표현에 관한 연구", 대한조선학회 논문집 28권 2호, 1991.
- [27] Microsoft, "Microsoft Fortran", Ver5.0, Vol 1-4, Microsoft Corporation, 1989.