

3차원 고체 유한요소해석의 새로운 후처리 방법

New Methods of Postprocessing for Finite Element Analysis of 3-dimensional Solids

이 재 영*
Lee, Jae Young

.....
요 약

3차원적인 고체의 유한요소해석 결과를 컴퓨터 그래픽스를 이용하여 시각화하는 후처리 방법들을 고안하고, 유한요소해석 소프트웨어의 개발에 응용하여 그 실용성과 효율성을 검토하였다. 이 연구에서는 고체 구조물의 후처리에서 가장 어려운 문제인 입체 내부의 데이터를 표현하는 방법을 중점적으로 다루었으며, 이를 위하여 공간 내부의 절단면을 표시하는 방법, 입체를 절단하여 분리하는 방법, 등가곡면으로 데이터 값의 범위를 표시하는 방법을 제안하였다.

Abstract

New methods of visualizing the data from finite element analyses of 3-dimensional solids were developed in this study. Their efficiency and practicality were examined through their application and implementation into a finite element analysis software. The major effort of the study was to provide a way of representing data inside of volume, which is the most difficult problem in the postprocessing of 3-dimensional solids. Section representation, volume slice and separation, and isosurface representation were proposed for this purpose.

.....

1. 서 론

유한요소해석 후처리의 내용과 방법은 대상 구조물의 종류와 복잡성에 따라서 달라진다. 예를 들어서 뼈대 구조물의 후처리는 전단력선도, 모멘트선도, 변위의 탄성곡선 등 주로 선도화(line drawing)로 이루어지며 평면문제의 경우에는 등용력선도와 같은 등고선의 도화가 후처리의 중심을 이룬다. 쉘 구조물은 그 형상이 입체적이므로

데이터를 표시함에 있어서 은선처리, 은면처리, 음영표현 및 3차원 좌표변환 등의 3차원 그래픽 기능이 추가되어야 한다. 후처리의 관점에서 볼 때 3차원 고체 구조물의 특징은 데이터 공간이 부피를 갖는 입체라는 점이다. 그러므로 3차원 고체 구조물의 후처리에서 필요로하는 요건은 1) 구조물의 입체적인 형상을 식별할 수 있도록 데이터가 표시되어야 하며, 2) 구조물의 투시방향을 임의로 설정하여 데이터를 표시할 수 있어야 하며, 3) 구

* 정회원, 전북대, 농공학과 부교수

이 논문에 대한 토론을 1994년 6월 30일까지 본 학회에 보내주시면 1994년 12월호에 그 결과를 게재하겠습니다.

조물 내부의 데이터를 표시할 수 있어야 하며, 4) 데이터의 공간적인 분포를 입체적으로 나타낼 수 있어야 한다는 것이다. 앞의 두가지 요건과 관련된 문제와 그 해결 방법에 관해서는 이미 이전의 연구^{8),9)}에서 다룬 바 있으며, 이 연구는 뒤의 두가지 요건을 대상으로 하여 이루어졌다.

실제로 3차원 고체 구조물의 유한요소해석 후처리에 있어서 가장 어려운 문제는 내부의 역학적인 거동을 어떻게 가시화하느냐 하는 것이다. 의료화상(medical image)에서 전형적으로 채택되고 있는 체적표현(volume rendering)방식^{2),3),5)}의 데이터구조는 체적 전체의 화상 정보를 체적화소단위(voxel)로 내포하고 있기 때문에 데이터 공간 내부의 정보를 쉽게 추출하여 밖으로 노출시킬 수 있다. 그러나 유한요소해석 전후처리 소프트웨어는 일반적으로 요소와 절점 중심의 데이터 구조를 가지고 있기 때문에 체적표현방식을 적용하는 것이 어려울 뿐만 아니라 매우 비효율적이다. 그러므로 유한요소해석의 후처리에서는 전반적으로 표면표현(surface rendering)방식이 채택되고 있다. 표면표현 방식에 의하면 물체 표면의 정보만 표현의 대상이 되며, 내부의 정보는 무시된다. 평면 구조물이나 셀의 경우에는 데이터 공간 자체가 표면에 국한되어 있기 때문에 표면표현에 의해서 모든 데이터를 시각화할 수 있다.^{1),4),7)} 그러나 3차원 고체 구조물의 경우에는 데이터공간이 부피를 갖고 있으므로 표면표현에 의해서는 구조물 외곽 경계표면의 데이터만이 표현되며 내부의 상태를 나타낼 수가 없다. 따라서 구조물 내부의 데이터를 시각화하기 위해서는 별도의 그래픽 처리와 데이터 가공이 필요하다. 이를 위해서 공간 내부에 절단면을 설정하여 절단면 상의 데이터를 표시하는 알고리즘과 입체 공간을 절단하여 분리함으로써 내부를 노출시켜 데이터를 표현하는 알고리즘을 개발하였다.

한편 등응력선도, 등변형도선도 등과 같이 등고선에 의한 표현 방법만으로는 구조물 내부의 응력이나 변형도 등의 공간적인 분포를 이해하기 쉽게 가시화하기가 어렵다. 이러한 문제를 해결하기 위해서 등가곡면(等價曲面, isosurface)이라는 새로운 개념이 제시된 바 있다. 등가곡면이란 동일한

데이터 값을 갖는 입체 공간내의 점들로 구성된 곡면을 의미한다. 등고선이 2차원적인 데이터 분포를 표시하는 곡선인데 비해서 등가곡면은 3차원적인 데이터의 분포를 표시하는 곡면이라고 할 수 있다. 이 연구에서는 등가곡면을 구하여 데이터를 표현하고 등가곡면 상에 다른 데이터를 매핑하여 표현하는 알고리즘을 개발하였다.

이 연구는 Macintosh 계열 컴퓨터를 위한 유한요소해석 시스템 개발 계획의 일부로 이루어졌다. 이 논문에 제시된 모든 방법과 알고리즘은 이 시스템의 후처리 소프트웨어에 실제로 응용되었으며, 소프트웨어의 실제적인 실행을 통해서 이들 방법과 알고리즘의 효율성과 실용성이 검토되었다.

2. 절단면의 데이터 표시

구조물 내부의 데이터를 표시하기 위해 이 연구에서 제안하는 첫번째 방법은 절단면을 설정하여 그 위에 존재하는 모든 점들의 데이터 값을 표시하는 것이다. 이 방법은 사용자가 지정한 위치와 방향에 따라서 잘단면 패취(patch)를 구성하는 과정과 구성된 절단면 패취위에 데이터를 투사하는 과정으로 이루어지며, 전체적인 실행과정의 흐름도는 그림 1에서 보는 바와 같다. 이 연구에서 개발한 절단면 표시 알고리즘을 각 단계별로 기술하면 다음과 같다.

2.1 절단면의 위치와 방향 설정

절단면의 위치와 방향은 절단면 위에 있는 3점의 좌표를 지정함으로써 결정된다. 키보드에 의한 수치적인 데이터의 입력으로 사용자가 원하는 구조물 내부의 위치와 방향을 정확하게 지정하기는 매우 어렵다. 절단면을 직관적이며 손쉬운 방법으로 정확하게 설정하기 위해서는 그림 7(a)에서 보는 바와 같이 사용자가 마우스를 이용하여 지시하는 바에 따라서 대화적이며, 도식적인 방법으로 데이터의 입력이 이루어질 수 있어야 한다.

2.2 절단면 기준 좌표계 설정

절단면은 3점의 좌표에 의해서 정의되며, 위에

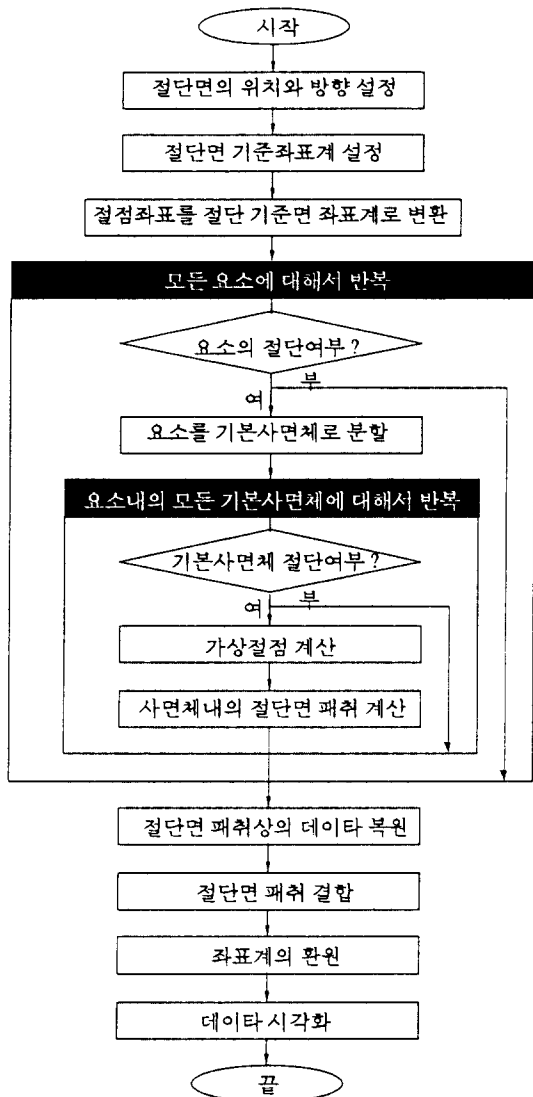


그림 1. 절단면 데이터 표시의 흐름도

기술한 바와 같이 대화적이며 도식적인 방법으로 3점의 좌표가 입력되면 절단면 상의 임의의 한점을 원점으로 하고 절단면에 수직인 방향으로 z축을 갖는 직각좌표계를 설정한다. 3점, p, q 및 r의 좌표를 각각 (x_p, y_p, z_p) , (x_q, y_q, z_q) 및 (x_r, y_r, z_r) 라고 하면 절단면 위에 있는 2개의 벡터

$$\mathbf{v}_1 = \frac{(x_q - x_p)\mathbf{i} + (y_q - y_p)\mathbf{j} + (z_q - z_p)\mathbf{k}}{D} \quad (1)$$

$$\mathbf{v}_2 = \frac{(x_r - x_p)\mathbf{i} + (y_r - y_p)\mathbf{j} + (z_r - z_p)\mathbf{k}}{D} \quad (2)$$

를 정의한다. 여기서

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

이며, $\mathbf{i}, \mathbf{j}, \mathbf{k}$ 는 각각 x, y, z 방향의 단위벡터를 나타낸다. 절단면 기준 좌표계의 좌표축 방향의 단위벡터를 \mathbf{i}', \mathbf{j}' 및 \mathbf{k}' 이라고 하면

$$\mathbf{i}' = \mathbf{v}_1 \quad (3)$$

$$\mathbf{k}' = \mathbf{i}' \times \mathbf{v}_2 \quad (4)$$

$$\mathbf{j}' = \mathbf{k}' \times \mathbf{i}' \quad (5)$$

이다. 여기서 \mathbf{k}' 이 절단면에 수직인 좌표축 방향의 단위벡터이다. 또한 편의상 점 p를 좌표의 원점으로 한다.

2.3 좌표 변환

계산 과정을 간략하게 하기 위해서 모든 절점의 좌표를 절단면 기준 좌표계로 전환한다. 좌표변환을 위한 회전행렬을 벡터 \mathbf{i}', \mathbf{j}' 및 \mathbf{k}' 의 성분에서 결정된다. 즉,

$$\mathbf{i}' = l_1\mathbf{i} + m_1\mathbf{j} + n_1\mathbf{k} \quad (6)$$

$$\mathbf{j}' = l_2\mathbf{i} + m_2\mathbf{j} + n_2\mathbf{k} \quad (7)$$

$$\mathbf{k}' = l_3\mathbf{i} + m_3\mathbf{j} + n_3\mathbf{k} \quad (8)$$

라고 하면, 회전행렬은

$$\mathbf{R} = \begin{bmatrix} l_1 & m_1 & n_1 \\ l_2 & m_2 & n_2 \\ l_3 & m_3 & n_3 \end{bmatrix} \quad (9)$$

이다. 점 p를 회전하여 얻은 좌표를

$$[X \ Y \ Z] = [x_p \ y_p \ z_p]\mathbf{R} \quad (10)$$

라고 하면, 좌표변환 행렬은

$$\mathbf{T} = \begin{pmatrix} l_1 & m_1 & n_1 & 0 \\ l_2 & m_2 & n_2 & 0 \\ l_3 & m_3 & n_3 & 0 \\ -X & -Y & -Z & 1 \end{pmatrix} \quad (11)$$

이다. 이는 점 p의 변환된 좌표가 (0, 0, 0)이 되도록 정해진 행렬이다. 절점 i의 좌표를 (x_i, y_i, z_i)라고 하면, 절단면 기준좌표계로 변환된 좌표(x'_i, y'_i, z'_i)는

$$[x'_i \ y'_i \ z'_i \ 1] = [x_i \ y_i \ z_i \ 1] \mathbf{T} \quad (12)$$

에 의해서 계산된다.

2.4 요소의 절단 여부 판정

절단면이 요소를 절단하는지의 여부는 절단면 기준좌표계로 환산된 요소내의 절점 좌표를 검사하여 판정할 수 있다. 절단면상의 모든점의 변환된 z'축방향 좌표는 0이다. 먼저 모든 절점을 z'좌표의 부호에 따라서 +, - 및 0의 세가지로 분류한다. 요소내 모든 절점의 부호가 1) 모두 0 또는 +일 경우에는 요소가 절단면보다 위에 있고, 2) 모두 0 또는 -일 경우 절단면보다 아래쪽에 있으며, 3) 일부는 +, 일부는 -일 경우에는 절단면에 의해 절단됨을 알 수 있다. 이와같이 요소의 위치를 1), 2) 및 3)의 세 유형으로 분류한다. 1) 또는 2)의 경우에서 1개 이상의 절점의 부호가 0인 경우에 이 요소의 일부가 절단면에 접촉하고 있으며, 이 경우는 별도로 처리한다.

2.5 요소의 분할

앞에서 기술한 바와 같이 절단면에 의해 절단되는 요소로 판정되면, 이 요소를 4절점의 기본사면체로 분할한다. 요소의 형상 및 절점수에 따른 기본사면체의 구성은 그림 2에서 보는 바와 같이, 이와같이 기본사면체로 분할하는 이유는 절단면 패취를 간략하게 구하고, 인접한 요소간에 절단면 패취가 연속되도록 하기 위한 것이다. 모든 요소를 일단 기본사면체로 분할하면, 요소의 형태나 절점수와 관계없이 동일한 알고리즘에 의해서 절단면 패취를 구할 수 있다.

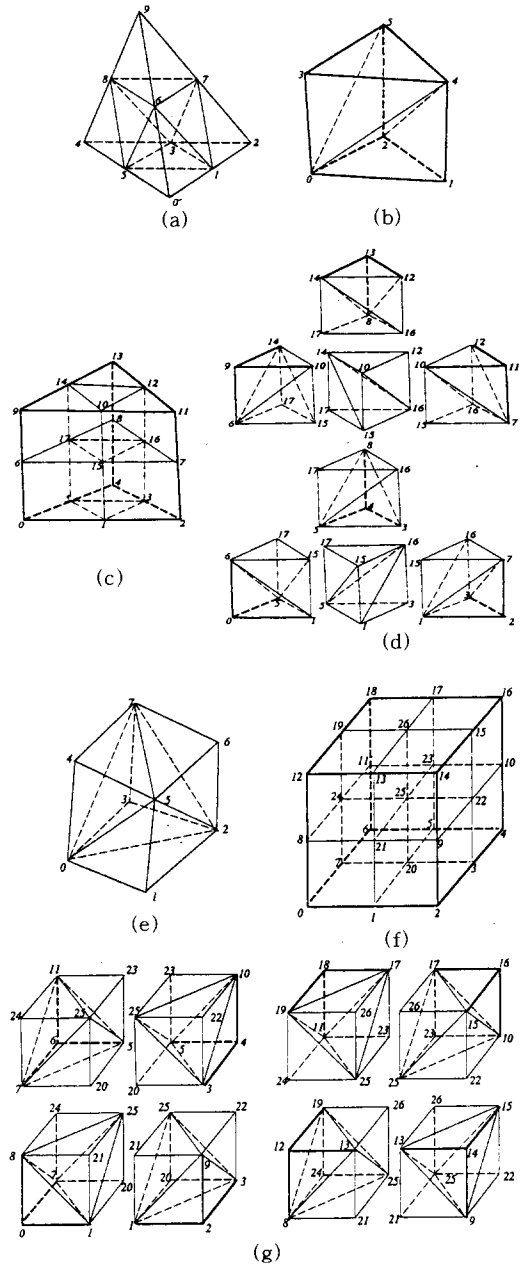


그림 2. 요소의 절단을 위한 기본사면체로의 분할

- (a) 10절점 사변형 요소
- (b) 6절점 프리즘 요소
- (c) 15절점 프리즘 요소
- (d) 15절점 프리즘 요소(분할그룹)
- (e) 8절점 육면체요소
- (f) 10절점 육면체요소
- (g) 20절점 육면체요소(분할 그룹)

2.6 기본사면체의 절단면 패취 계산

먼저 기본사면체가 절단 기준면에 의해 절단되는지의 여부를 판정한다. 요소의 경우와 마찬가지로 기본사면체내의 절점의 부호를 검사함으로써 절단 여부를 판정할 수 있다. 절단되는 것으로 판정된 기본사면체에 대해서만 다음에 기술한 과정을 거쳐서 절단면 패취를 구한다. 먼저 각 절점을 z'좌표 크기 순으로 나열한 다음에 절단되는 형태를 그림 3과 같이 분류한다. 즉 z'의 값이 큰 절점부터 번호를 1, 2, 3, 4로 부여하고, 다음과 같이 세 가지 경우로 나누어서 절단면 패취를 형성한다.

1) 절점 1만 절단 기준면보다 위에 있는 경우
 기본사면체를 (1, a, b, c.)의 사면체와 (a, b, c, 2, 3, 4)의 프리즘으로 분할한다. 절단면 패취(a, b, c)를 형성한다.

2) 절점 1과 2가 절단면보다 위에 있는 경우
 (1, a, b, 2, c, d)와 (a, 3, c, b, 4, d)의 두 프리즘으로 분할한다. 절단면 패취(a, b, c, d)를 형성한다.

3) 꼭지점 1, 2 및 3이 절단면보다 위에 있는 경우
 (1, 2, 3, a, b, c)의 프리즘과 (a, b, c, 4)의 사면체로 분할한다. 절단면 패취(a, b, c)를 형성한다.

여기서 a, b, c 또는 d는 절단면 위에 새로 만들어진 가상절점을 표시하며, 이는 절단면 양쪽의 절점 좌표를 보간하여 구한다. 예를 들어서 절점 i와 j를 연결하는 모서리 상의 가상절점 좌표의 계산을 위한 보간값은

$$r = \frac{|z_i|}{|z_i| + |z_j|} \tag{13}$$

이며, 가상절점을 q라고 하면

$$x'_q = (1-r)x'_i + rx'_j \tag{14}$$

$$y'_q = (1-r)y'_i + ry'_j \tag{15}$$

이고, 가상절점이 절단면 상에 위치하므로 $z'_q = 0$ 이다. 이 연구에서는 편의상 절점 i와 j를 가상절점 q의 모절점(母節點)이라고 명명하였다.

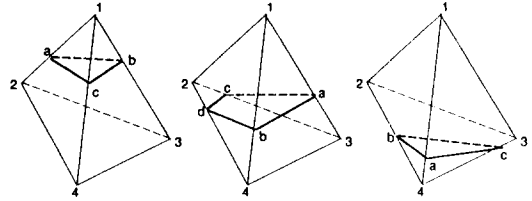


그림 3. 기본사면체의 절단면 패취

2.7 절단면 패취 상의 데이터 복원

유한요소해석을 통해서 계산되는 응력, 변형도, 변위 등은 절점에서의 값이며 이들 값을 보간하여 요소내의 데이터 값의 분포를 추정한다. 절단면 패취상의 데이터 분포는 패취를 형성하는 가상절점의 데이터 값에 의해서 정의된다. 가상절점의 데이터 값은 모절점의 값을 보간하여 구한다. 즉 모절점 i와 j의 데이터 값을 각각 u_i 와 u_j 라고 하면, 가상절점 q의 데이터 값은

$$u_q = (1-r) u_i + r u_j \tag{16}$$

이며, 여기서 r은 식(13)으로 구한 보간값이다.

2.8 절단면 패취의 결합

각 요소를 절단하는 절단면 패취를 구한 다음에는 이들을 결합하여 연속적인 절단평면을 구성한다. 패취의 결합은 패취간에 공통된 가상절점을 통해서 이루어진다. 여기서 중요한 것은 모든 패취의 방향이 일치하도록 패취내의 가상절점의 번호를 부여하는 것이다. 패취의 방향은 이에 수직인 벡터의 방향으로 정의되며, 벡터의 방향은 가상절점의 번호에 따라서 일관성있게 오른손 나사, 또는 왼손 나사의 방향으로 정한다. 인접한 패취는 서로 공통된 가상절점의 번호를 적절히 부여함으로써 방향을 일치시킬 수 있으며, 순환 알고리즘을 적용하여 모든 패취의 방향이 일치하도록 가상절점의 번호를 부여할 수 있다. 즉 절단면 패취의 가상 절점을 차례로 연결한 선분이 패취의 방향에서 봤을 때 시계방향(또는 반시계방향)의 고리를 만들도록 일관성있게 번호를 부여한다.

2.9 좌표계의 환원

모든 절단면 패취가 계산되면 좌표계는 절단면 기준좌표계에서 원래의 좌표계로 환원되어야 한다. 좌표의 환원은 변환된 절점 및 가상절점의 좌표에 식(11)의 행렬 **T**를 곱하여 변환 이전의 좌표를 계산함으로써 이루어진다.

2.10 절단면 상의 데이터 시각화

절단면 패취를 결합하여 형성된 절단면 상의 데이터 값 분포는 가상절점의 데이터 값을 각 패취별로 보간하여 추정하고, 그에 따라서 등응력선도나 또는 절단면의 변형 등 해석 결과를 도식적으로 시각화한다. 앞에서 기술한 바와 같이 절단면 패취가 상호 부합되도록 결합되었다면 절단면상의 연속적인 등고선 표시 뿐만 아니라 부드러운 음영표시가 쉽게 이루어질 수 있다.

절단면의 위치와 방향을 쉽게 알아볼 수 있도록 하기 위해서는 구조물의 외곽 경계표면과 함께 절단면의 데이터가 표시되는 것이 바람직하다. 구조물의 경계표면 표시는 그림 7(b)의 예에서 보는 바와 같이 투명시⁶⁾로 하거나, 절단면 뒷부분만의 반투명시 또는 그물망표시(wireframe representation)로 할 수 있다.

3. 구조물 체적의 절단 및 분리

앞절에 기술한 바와 같이 데이터 표시가 완료되면 이와 관련된 모든 기억 공간이 해제된다. 즉 절단면 패취는 일시적으로 만들어진 가상의 객체(object)며, 데이터 표시가 완료됨과 동시에 소멸된다. 이와는 달리 데이터 공간의 일부를 절단하여 독립된 데이터공간으로 분리하여 처리함으로써 후처리를 일층 더 편리하게 진행시킬 수 있다. 이 연구에서 고안한 이 방법에 의하여 생성된 절단면은 구조물의 경계 표면을 형성하여 후처리가 진행되는 동안 존속한다. 이 경우에 있어서도 절단면을 형성하는 과정은 앞의 절단면 데이터 표시를 위한 절단면 형성 과정과 동일하다. 그러나 새로운 데이터 공간의 형성 및 분리 과정이 추가되며 데이터의 구성이 달라진다. 이 방법의 전체적인 흐름도는 그림 4와 같으며, 절단면의 형성과정

을 다시 기술할 필요는 없으므로, 여기서는 추가적으로 소요되는 과정만을 기술하였다.

3.1 요소의 절단

체적을 절단하여 분리하기 위해서는 먼저 절단면에 걸쳐있는 요소를 절단하여 독립된 요소들로 분리하여야 한다. 앞에 기술한 절단면 데이터 표시의 경우와 동일한 방법으로 체적내에 있는 모든 요소에 대해서 절단 여부를 판정하고, 기본사면체로 분할한다. 그림 3에서 보는 바와 같이 기본사면

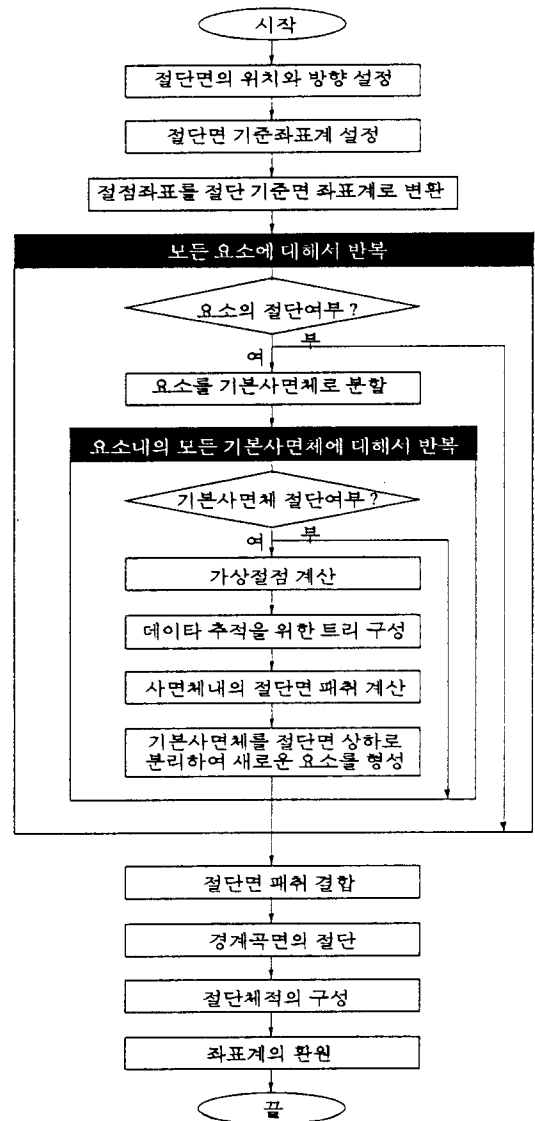


그림 4. 체적의 절단 및 분리 계산의 흐름도

체가 절단면 패취에 의해서 상하로 절단되면 2개의 새로운 요소를 형성한다. 즉 기본사면체가 절단되면 두개의 사면체요소, 또는 하나의 사면체요소와 하나의 프리즘 요소를 형성한다. 이들 요소는 절단에 의해서 새로 형성된 체적에 분산되어 포함되며 후처리가 진행되는 동안 실제의 요소와 동일하게 취급된다.

3.2 경계곡면의 절단

체적을 절단하면 그 체적의 경계곡면도 절단되어 두개이상의 새로운 곡면을 형성하게 되며, 이들은 절단에 의해 새로 만들어진 체적의 경계곡면을 이루게 된다. 경계곡면의 절단이란 각 곡면을 표현하는 그물망을 절단하고, 분리시켜서 새로운 그물망을 형성하는 과정을 의미한다. 요소의 절단 과정에서 절단된 요소의 모서리를 결합하여 요소 경계면의 삼각형 또는 사변형을 만들고, 앞에서 기술한 절단면 패취의 결합과 동일한 방법으로 곡면패취를 형성하면 경계곡면의 그물망이 완성된다.

3.3 체적의 절단

구조물은 하나 또는 여러개의 체적(volume)으로 구성되며, 후처리 과정에서 이들은 각기 독립된 데이터 공간으로 취급된다. 체적을 구성하는 기본 데이터 구조는 다음과 같다.

```
struct volumes {
    short numOfNode, numOfElem, numOfSurf;
    struct nodes **node;
    struct elems **elem;
    struct surfs **surf;
    struct volumes *vol;
};
```

여기서 numOfNode, numOfElem 및 numOfSurf는 각각 체적에 포함된 절점, 요소 및 경계곡면의 수이며, node, elem 및 surf는 각각 체적에 포함된 절점, 요소 및 경계곡면들의 포인터 변수이고, vol은 체적이 분할되어 나온 모체적의 포인터 변수이다.

구조물 체적의 절단이란 요소를 절단하고, 절단

면 패취를 만들고, 경계곡면을 절단하여 새로운 체적을 구성하여 독립된 데이터 공간을 형성하는 과정을 의미한다. 체적의 절단을 위해서 절단부위의 부피내에 포함된 절단된 요소들과 절단되지 않은 요소들간의 연결성을 검사하여 절단면의 양쪽으로 분류하여 새로운 체적에 포함시킨다. 또한 절단면 양쪽의 체적에 대해서 새로운 경계곡면의 리스트를 만든다. 이들 경계곡면에는 일부 기존의 경계곡면(절단되지 않은)과 앞에서 만들어진 절단면 패취 및 분할되어 새로 만들어진 경계곡면 등이 포함된다. 절단면 패취는 일시적인 가상의 객체가 아니라 새로운 체적의 경계곡면으로 존속된다.

3.4 체적의 분리

체적을 절단하는 목적은 새로 구성된 체적들을 분리하여 독립적으로 이동, 회전, 축소, 확대 등 기하학적인 연산을 실행함으로써 역학적인 거동을 효과적으로 시각화하는데 있다. 체적을 분리하기 위해서는 절단면 패취 양쪽의 공통적인 절점들을 복제하여 분리된 평면 패취를 구성하여야 한다. 절단면 패취상의 모든 절점은 가상절점이며, 이들을 분리하기 위해서는 모절점을 추적할 수 있도록 데이터 구조가 이루어져야 한다.

3.5 절점의 데이터 구조

가상 절점의 데이터 구조는 모절점을 추적하여 원래의 데이터로부터 가상절점의 데이터를 계산할 수 있도록 이루어져야 한다. 왜냐하면 후처리를 위한 데이터 값은 체적을 절단한 이후에 보조 기억장치로부터 읽어들이게 되는데, 가상절점이 아닌 실제의 절점에 대한 절점의 값만 존재하며, 가상절점의 값은 실제 절점의 값으로부터 추적하여야 하기 때문이다. 이와같은 추적을 가능하게 하기 위하여 절점의 데이터는 모절점과 이의 보간값을 포함한다. 절점의 데이터 구조는 다음과 같다.

```

struct nodes {
    short number;
    float xyz[3],value[3],r;
    struct bounds *bound;
    struct loads *load;
    struct displ *displ;
    struct nodes *nodei,*nodej;
}
    
```

여기서 number는 절점의 번호, xyz 및 value는 각각 절점의 좌표와 절점의 데이터 값(변위, 응력 등)이고, r은 모절점의 보간값이며, bound, load 및 displ은 각각 절점에 부여된 경계조건, 하중조건, 초기변위조건 등이다. nodei는 절단면윗쪽(절단면 기준좌표계에서)의 모절점에 대한 포인터 변수이며, nodej는 절단면 아랫쪽의 모절점에 대한 포인터 변수이다.

3.6 가상절점의 데이터 값 추적

절단면 패취의 가상절점으로 부터 트리를 거슬러 올라가면 실제의 절점에 도달한다. 여기서부터 다시 데이터 값을 순차적으로 보간하여 내려오면 절단면 패취상의 가상절점의 데이터 값이 계산된다. 즉 체적을 절단한 이후에도 절점의 응력, 변형도, 변위등의 데이터를 보조기억장치로 부터 읽어 들여서 절단면 패취상의 데이터 분포를 추적할 수 있다. 이는 다음과 같이 간단한 순환 알고리즘에 의해 실행된다.

```

TraceNodeValue(node,dof)
struct nodes *node;
short dof;
{
    if(node->nodei==NULL)
        return;
    TraceNodeValue(node->nodei,dof);
    TraceNodeValue(node->nodej,dof);
    node->value[dof]=node->nodei->value[dof]
        *(1-node->r)+node->nodej->value[dof]*node->r;
}
    
```

3.7 절단 분리된 체적의 데이터 시각화

체적을 절단하고 분리한 이후에는 개개의 체적에 포함된 절단면 패취와 경계곡면의 기하학적인 형상에 바탕을 두고 구조물의 변형 전후 형태나 또는 각종 데이터의 시각화가 이루어진다. 앞에서 기술한 바와 같이 절단면 패취 및 절단에 의해서 분리된 경계곡면에 포함된 가상절점의 데이터 값

을 계산하고, 일상적인 보간법에 의해서 곡면상의 데이터 분포를 추정하여 등고선 표시, 음영표시 등의 데이터 시각화를 실행한다.

4. 등가 곡면(iso-surface) 표시

등고선이 평면이나 곡면상의 동일한 데이터 값을 갖는 점들을 연결한 곡선이라고 하면, 등가곡

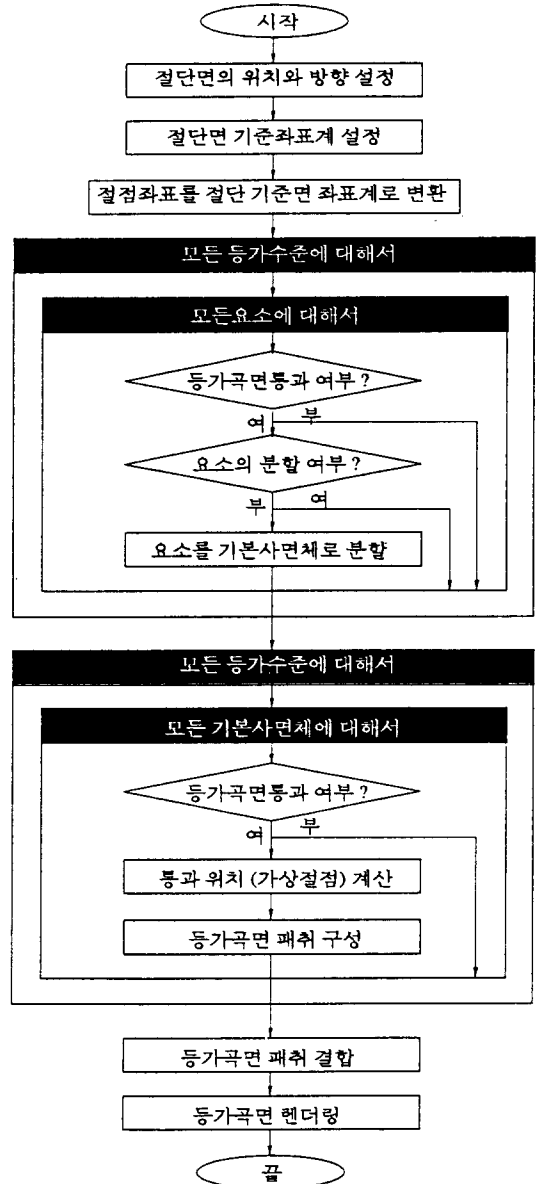


그림 5. 등가곡면 표시의 흐름도

면은 입체적인 데이터 공간에서 동일한 데이터 값을 갖는 점들로 구성된 연속적인 곡면이라고 할 수 있다. 입체 구조물의 후처리에 있어서 등고선이 경계곡면의 데이터 값만을 표시하는데 반해서 등가곡면은 입체 공간내부에서 데이터 값의 공간적인 분포를 직접적으로 표시한다. 구조물 체적내에 발생하는 응력이나 변형도 등의 분포는 여러개의 등가곡면을 입체적으로 표현함으로써 이해하기 쉽게 나타낼 수 있다. 그러나 이 방법은 데이터 값의 변화가 완만하게 분포되어 있고, 연속성이 양호한 경우에 한해서만 효과적이다. 그렇지 못한 경우에는 등가곡면이 매우 복잡하게 표시되어 데이터의 전체적인 경향을 파악하기가 어렵게 된다.

4.1 등가 수준의 결정

등가수준이란 개개의 등가곡면이 나타내는 데이터 값의 크기를 의미한다. 등가수준은 등가곡면의 수와 배치방법에 따라서 결정된다. n개의 등가곡면을 등배치한다면 i번째 등가곡면의 등가수준은 $v_i = (v_{max} - v_{min})^i / (n+1)$ 이다. 여기서 v_{max} 와 v_{min} 는 각각 데이터 값의 최대 및 최소치를 의미한다. 데이터의 분포에 따라서는 등가곡면을 부등배치하거나 또는 각 등가곡면의 값을 지정하는 것이 적합할 수도 있다. 예를 들어서 데이터의 변화가 국부적으로 급격할 경우에는 등가곡면을 등배치하면 한 부분에 밀집되어 구조물 전체의 경향을 적절히 표현하지 못하게 된다. 이때에는 등가곡면을 적당한 비율로 부등배치함으로써 시각적인 효과를 높일 수 있다.

4.2 등가곡면 통과 여부 판정

모든 등가곡면에 대해서 순차적으로 어떤 요소를 통과하는지 여부를 검사한다. 이는 요소내의 절점의 데이터 값과 등가수준을 비교하여 결정할 수 있다. 즉 요소내의 데이터 값의 최소치와 최대치를 각각 v_{min}^e 및 v_{max}^e 라고하면, i번째 등가곡면이 요소를 통과하는 조건은 $v_{min}^e \leq v_i \leq v_{max}^e$ 이다. 모든 요소에 대해서 이 조건이 만족되는지 여부를 검사하여, 등가곡면이 통과하는 요소들을 선별하고, 이들 요소에 대해서만 다음에 기술한 바와 같은 등가곡면 패취의 계산 과정을 실행한다.

4.3 요소의 분할

하나 이상의 등가곡면이 통과하는 것으로 판정된 요소는 그림 2에서 보는 바와 같이 4개의 절점을 갖는 기본사면체로 분할한다. 분할 방법은 앞에 기술한 절단면 데이터 표시의 경우와 동일하다. 이와 같이 기본사면체로 분할하는 이유는 등가곡면의 계산을 간단하게 하고, 등가곡면의 연속성을 보장하기 위한 것이다.

4.4 등가곡면 통과 위치의 계산

기본사면체의 6개 모서리에 대하여 등가곡면이 통과하는 위치를 구한다. 개개의 등가곡면의 값이 모서리 양끝의 절점의 데이터값 사이에 있는지의 여부를 판정하여 등가곡면의 통과 여부를 결정한다. 등가곡면이 통과할 경우에는 선형보간에 의해서 등가곡면이 통과하는 위치를 계산하여 가상절점을 만든다. 예를 들어서 그림 6에서 보는 바와 같이 양쪽 절점의 좌표를 각각 (x_1, y_1, z_1) 및 (x_2, y_2, z_2) 라고 하고, 데이터의 값을 v_1 및 v_2 라고 하면, 데이터 값 v_p 를 나타내는 등가곡면의 가상절점 좌표는

$$x_p = x_1 + \frac{v_p - v_1}{v_2 - v_1}(x_2 - x_1) \tag{17}$$

에 의해서 간단하게 계산된다. 또한 동일한 데이터 포인트들을 연결한 모서리상에서 등가곡면의 위치는 소속된 사면체에 관계없이 일정한 값으로 계산된다. 이는 등가곡면의 연속성을 의미한다. 그러나 데이터 값의 분포가 불연속적이라면 등가곡면도 불연속적이어야 하기 때문에 별도의 처리가 필요하다.

4.5 등가곡면 패취의 구성

특정한 데이터 값에 해당되는 가상절점들이 계산되면, 이들을 연결하여 사각형 또는 삼각형을 구성한다. 사각형의 경우에는 그림 6에서 보는 바와 같이 반으로 분할하여 두개의 삼각형을 만든다. 이렇게 만들어진 삼각형은 등가곡면을 구성하는 삼각형 패취가 된다. 삼각형 패취의 데이터 구조는 삼각형에 포함된 가상절점과 이들 상호간의 위상관계로 이루어진다.

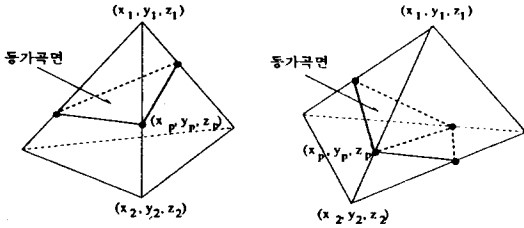


그림 6. 등가곡면 패취의 구성

4.6 등가곡면 패취와 결합

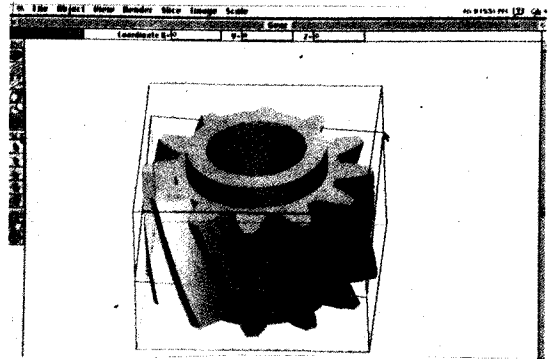
등가곡면이 통과하는 모든 기본사면체에 대해 서 위의 두 과정을 실행하여 형성된 삼각형 패취들의 리스트를 만들고 이들을 결합하여 연속적인 등가곡면을 구성한다. 이때 하나의 데이터 값을 나타내는 등가곡면은 연속적인 단일 곡면패취로 이루어질 수도 있지만, 데이터 공간의 형상과 데이터의 불연속성 등에 따라서 불연속적인 여러개의 곡면패취로 나뉘어서 형성될 수도 있다.

4.7 등가곡면 표시

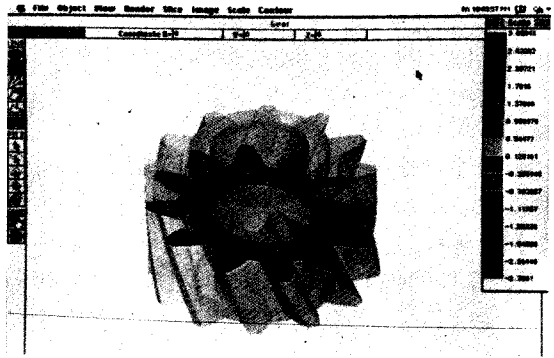
등가곡면이 완성되면 그림 8(a)에서 보는 바와 같이 등가수준에 대응되는 색을 정하여 음영표현한다. 등가곡면을 구성하는 개개의 패취별로 음영표현을 하면 완만하고 부드러운 음영표시가 이루어지지 않는다. 모든 가상절점에서 이를 공유하는 패취들의 법선 벡터를 평균하여 음영표시를 함으로써 이러한 결함을 보완할 수 있다.

등가곡면 상에 또 다른 데이터의 분포를 등고선으로 표현할 수 있다. 그림 8(b)의 예에서 보는 바와 같이 동일한 변위의 수준을 나타내는 등가곡면 위에 등응력선도를 표시할 수 있다. 이는 3.6절에 기술된 가상절점의 데이터 값 추적 방법을 이용하여 등가곡면상의 데이터 분포를 추정하여 등고선을 도화함으로써 실행된다.

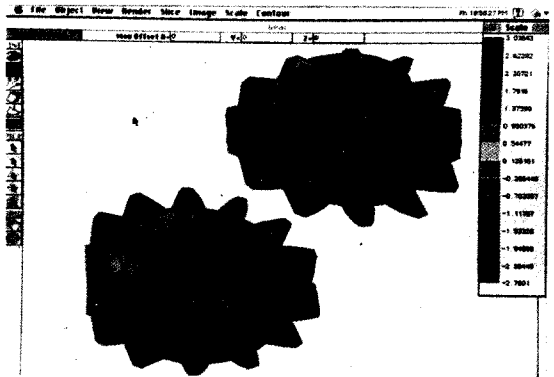
등가곡면의 공간적인 위치를 쉽게 이해할 수 있도록 하기 위해서는 등가곡면을 구조물의 경계곡면과 함께 나타나는 것이 바람직하다. 경계곡면은 그림 8(a)에서 보는 바와 같이 반투명체로 표현하거나, 그물망 또는 윤곽선으로 표현할 수 있으며, 그림 8(c)와 같이 등가곡면에 의해 절단된 구조물의 형상을 표시할 수도 있다.



(a)



(b)



(c)

그림 7. 체적의 절단 및 분리에 의한 해석 결과의 표시
 (a) 대화적, 도식적 방법에 의한 절단면의 설정
 (b) 절단면 상의 데이터 표시
 (c) 절단 및 분리에 의한 데이터 표시

5. 실용성 및 효율성 검토

이 연구는 Macintosh 계열 컴퓨터를 위한 유한요소해석 시스템 개발 계획의 일부로 이루어졌다.

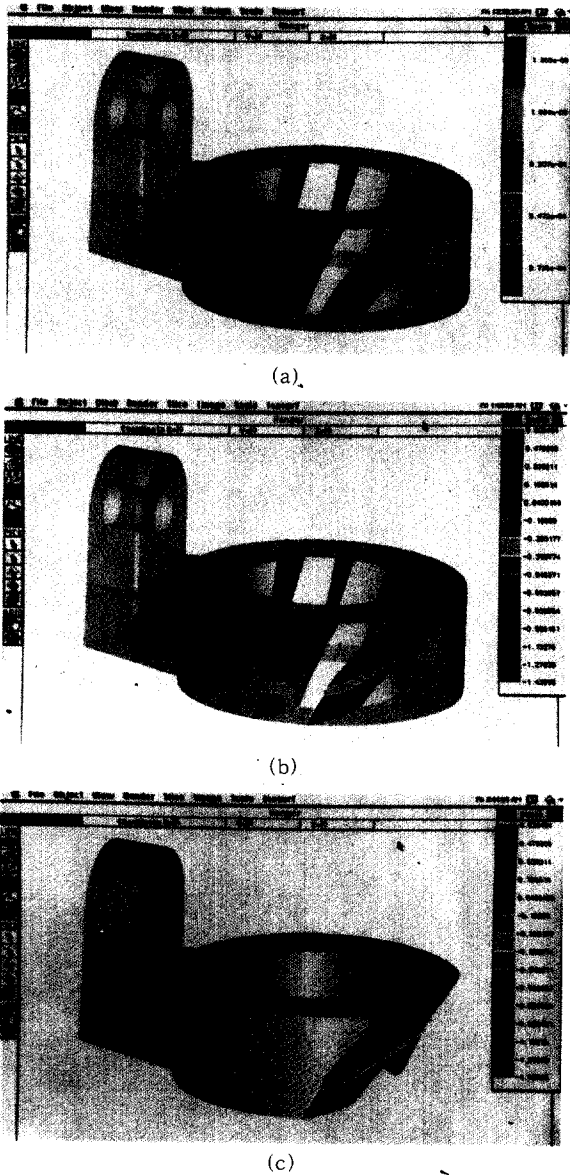


그림 8. 등가곡면에 의한 해석 결과의 표시
 (a) 등변위를 나타내는 등가곡면
 (b) 등가곡면상에 매핑된 등응력선도
 (c) 등가곡면에 의해 절단된 표면에 매핑된 등응력선도

앞에서 기술한 절단면 표시 방법, 체적의 절단, 분리 방법 및 등가곡면 표시방법은 이 시스템의 후처리 소프트웨어에 실제로 응용되었으며, 소프트웨어의 실제적인 실행을 통해서 이들 방법과 알고

리즘의 효율성과 실용성이 검토되었다. 또한 여러 가지 구조물의 형태와 여러 종류의 요소를 조합한 해석 예를 설정하여 이들 방법을 실행함으로써 이들 방법과 알고리즘이 보편성을 지니고 있는지를 검증하였다. 그림 7과 8은 그중 일부를 예시한 것이다.

6. 결 론

3차원 고체의 유한요소해석 결과를 시각화하는 방법으로서 절단면 표시 방법, 체적의 절단 및 분리 방법, 등가곡면 표시방법을 개발하였으며, 이들 방법을 이용한 후처리 소프트웨어를 개발하였다. 실제의 후처리 과정에서 이들 방법은 도식적으로 표현하기 힘든 입체 내부의 역학적인 거동을 편리하게 분석하고 쉽게 이해할 수 있도록 시각화하는 방법으로서 매우 유용하였다. 또한 소프트웨어적인 측면에서 직관적이며, 간편한 사용자 인터페이스를 가능하게 하였다.

절단면 표시방법은 시각화하고자 하는 데이터가 절점과 요소에 할당된 이후에 사용자가 설정한 절단면 위에 데이터를 표시하는 수단으로 편리하게 이용될 수 있으며, 체적의 절단, 분리 기능은 데이터 공간을 실제적으로 절단하고 분리시켜서 구조물의 여러가지 역학적인 거동을 관찰할 수 있는 이점을 지니고 있다. 등가곡면 표시방법은 데이터의 공간적인 분포를 쉽게 이해할 수 있도록 가시화하는 방법으로서 매우 효과적이었다. 여러 가지 구조물 형상과 요소를 대상으로 한 해석 예를 설정하여 검증한 바에 의하면 이들 알고리즘은 비교적 간단하면서도 보편 타당성을 지니고 있음이 입증되었으며, 따라서 향후 3차원 구조물해석을 위한 유한요소해석 후처리 소프트웨어의 개발에 널리 적용될 수 있을 것으로 기대된다.

감사의 글

본 논문은 1992년 1월부터 12월까지 과학기술처 특정연구개발과제의 연구비지원에 의해 연구되었으며 이에 깊은 감사를 드립니다.

참 고 문 헌

1. Abel, J.F., A.R. Ingraffea, W. McGuire, and D.P. Greenberg, Interactive color graphical postprocessing as a unifying influence in numerical analysis research, *Finite Elements in Analysis and Design*, Vol.2, pp1-17, 1986.
2. Crawford S.L., and T.C. Fall, Projection pursuit techniques for the visualization of high dimensional datasets, *Visualization in Scientific Computing*, IEEE Computer Society Press, pp94-108, 1990.
3. Doctor, L.J., J.G. Torborg, Display techniques for octree encoded objects, *IEEE Computer Graphics and Applications*, Vol.1, No.3, pp29-38, 1981.
4. Forcade, T., Engineering visualization, *Computer Graphics World*, Vol.15, No.11, pp37-44, 1992.
5. Giertsen, C., Volume visualization of sparse irregular meshes, *IEEE Computer Graphics and Applications*, Vol.12, No.2, pp40-48, 1992.
6. Kay, D.S., and D.P., Greenberg, Transparency for computer synthesized images, *Computer Graphics*, Vol.13, pp158-164, 1979.
7. Singh, C., and D. Sarcar, A simple and fast algorithm for the plotting of contours using quadrilateral meshes, *Finite Elements in Analysis and Design*, Vol.7, pp217-228, 1991.
8. Thalmann, D., *Scientific Visualization and Graphic Simulation*, John Wiley, 1990.
9. 이재영, 마이크로컴퓨터를 이용한 유한요소해석의 효율적 Postprocessing, 전산구조공학, 3권 2호, pp117-128, 1990.

(접수일자 : 1993. 8. 12)