

유한 요소법을 위한 메쉬의 자동생성

박 준 영



●1960년생
●CAD/CAM을 전공하였으며, 메쉬의 자동생성법 및 Geometric Modeling에 관심을 가지고 있다.

1. 머리말

유한 요소법에서 어떤 물체의 외형으로부터 메쉬를 생성하는 단계는 매우 중요하다. 메쉬 생성이란 물체의 내부를 이미 정해진 요소로 나누는 기법을 말하는데 유한 요소법이 적용되기 이전에 반드시 필요한 과정이다. 이 과정은 주어진 영역에서만 행하여 지고 알고자 하는 물리적 현상을 묘사하는데 필요한 미분 방정식의 조건들과 일치하도록 행하여 진다. 메쉬 생성에 쓰이는 요소들의 모양으로는 삼각형과 사각형이 2차원에서 사용되고, 썰기모양과 사면체가 3차원에서 사용된다. 그 중에서 2차원에서는 삼각형이 3차원에서는 사면체 요소가 주로 사용되는데 그 이유는 이러한 요소들이 복잡한 물체들을 잘 묘사하기 때문이다.

메쉬는 물체의 내부와 외부를 근사적으로 묘사한 것이므로 그 물체와 메쉬와의 차이는 주어진 오차를 넘지 않아야 한다. 메쉬 생성은 1960년대까지 수동으로 이루어졌다. 그 이후로, 복잡한 물체를 간단한 요소로 나누는 데 컴퓨터가 사용되어 왔다. 또한, 메쉬의 구조를 가시화하고 규명하는데 메쉬 생성 프로그램들이 사용되어 왔고 메쉬 생성 과정을 자동화하기 위하여 많은 알고리즘이 개발되었다. (2,3,5,8,12,13,14)

이러한 알고리즘들에서는 메쉬 생성 이후,

고르기 과정(smoothing process)이라는 반복적인 방법을 통해서 작은 각도를 가지고 있는 요소들을 없앤다. 고르기 과정을 사용하더라도 이 알고리즘들은 메쉬안에 있는 제일 작은 각도의 하한값(lower bound)을 항상 보장할 수는 없고 메쉬를 향상시키기 위하여 때로는 인위적인 간섭이 필요하다. 유한 요소법에서 결과에 대한 오차는 메쉬안에 있는 제일 작은 각도에 반비례하므로,⁽¹¹⁾ 그 값이 하한값을 갖도록 하는 것은 중요하다. 이 글에서는 메쉬안에 있는 제일 작은 각도의 하한값을 보장할 수 있는 메쉬 생성 알고리즘을 개발하였다.

최근에, 메쉬 요소의 모양을 가능한 한 정삼각형에 가깝게 함으로써 메쉬를 향상시키는 몇 가지 시도가 있었다. 그 중 몇 알고리즘은 메쉬 안에 있는 제일 작은 각도의 하한값을 갖게 하였다. 예를 들면, Rivara,⁽¹⁰⁾는 메쉬 안에 있는 제일 작은 각도의 하한값은 초기 삼각화(initial triangulation)에서 제일 작은 각도의 반 임을 증명하였다. 이때의 메쉬 생성은 각 삼각형의 가장 긴 변을 이등분함으로써 이루어졌다. 요소의 한 변이 둘 이상의 인접한 요소들과 공유되지 않아야 하는 일치성(conformity) 문제를 해결하기 위하여, 각 삼각형의 가장 긴 변을 이등분하는 과정을 메쉬의 일치성이 이루어질 때까지 반복되어진다. Field⁽⁶⁾는 Delaunay 삼각화를 이용하여 메쉬를 향상시키는 알고리즘을 개발하였다. 이 알고리즘은 지

금까지도 널리 쓰여지고 있는 Laplacian 고르기 과정을 좀더 발전 시킨 것이다. Laplacian 고르기 과정에서는 노우드(node) i 는 노우드 i 를 둘러 싸고 있는 노우드들의 좌표의 평균으로 다시 정의된다. 새로 정의된 노우드가 Delaunay 삼각화의 조건을 만족하지 않는 경우에 그 조건을 만족할 수 있도록 Field의 알고리즘이 사용된다. 그러나 이 방법은 메쉬 향상을 항상 보장하지는 못한다. Baker, Grosse and Rafferty⁽¹⁾들이 하한값을 갖게 하는 삼각형 메쉬를 생성하는 알고리즘을 2차원에서 개발하였다. 알고리즘 적용 후 메쉬 안에 있는 모든 각도는 13° 와 90° 사이에 놓이게 된다. 이 알고리즘에서는 메쉬가 생성될 부분에 정사각형 격자를 올려 놓은 다음 물체 내부에 놓인 정사각형들에 대해서는 대각선을 이용하여 삼각형 메쉬를 생성하였다. 그 다음 물체 주변에 놓인 사각형들에 대한 메쉬 생성 방법은 따로 개발되었다. 이 기법을 사용하면 물체 주변에 작은 삼각형들이 생성되는 단점이 있다.

2. 이차원에서의 메쉬 생성법

주어진 물체의 외형을 바탕으로 메쉬를 생성하기 위해서는 노우드를 생성하고 그 노우드를 바탕으로 삼각형 메쉬를 만드는 기법이 필요하다. 이 절에서는 노우드를 생성하고 그 노우드를 바탕으로 삼각형 메쉬를 만드는 기본적인 요소들을 설명하기로 한다.

2.1 NE(Node on Edge)

임의의 삼각형에서 다음과 같이 간단하면서도 중요한 관찰을 할 수 있다.

관찰

- i) 어떠한 삼각형이라도 60° 보다 크거나 같은 작은 적어도 하나 이상 있다.
- ii) 한 삼각형에서, 가장 긴 변(LE; longest edge)을 마주보는 각도는 항상 60° 보다 크거나 같다.

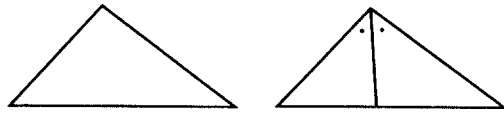


그림 1 NE의 적용예

위와 같은 관찰로부터 다음과 같은 사실을 알 수 있다. 한 삼각형의 모든 각도가 30° 보다 크다고 할 때, 가장 큰 각을 이등분하는 선분으로 그 삼각형을 둘로 나눈다면 나뉘진 두 삼각형들에서의 모든 각도도 30° 보다 크거나 같다. 이와 같은 요령으로 메쉬를 생성하는 기법을 만들려면, 메쉬의 일치성을 만족하는가를 고려해야 한다. 메쉬의 일치성을 유지하기 위해서는 LE를 마주보는 각도를 이등분하는 기법은 LE가 가장자리변(BE; boundary edge)일 때에만 적용되어야 한다. 이러한 작용을 NE(Node on Edge)라고 하며 파스칼 프로그램 언어에서는 프로그램 I 과 같이 행하여 진다.

프로그램 I

Procedure NE

{*NE finds a boundary node location and completes the triangulation.*}

Begin

If LE=BE then

Bisect the angle which faces LE

End{*NE*}

2.2 NI(Node in Interior)

LE가 내부변(IE; interior edge)일 때는 LE를 공유하는 두 삼각형에 NE를 적용하면 메쉬의 일치성을 만족하지 못하는 경우가 그림 2와 같이 생기게 된다.

이러한 경우의 노우드 생성을 설명하기 위해

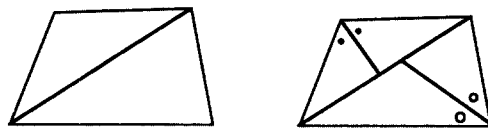


그림 2 NE의 내부변에 대한 적용예

여, 다음과 같은 용어를 정의한다.

정의 1

한 사각형이 대각선을 이용하여 두 개의 삼각형으로 나뉘어진 경우를 생각해 보자. 만일 그 대각선이 두 개로 나뉘어진 사각형에서 LE이라면 IE를 마주보는 두 각은 60° 보다 크거나 같다. 그러한 사각형을 QUAD라고 정의한다. QUAD의 모양은 항상 볼록꼴(convex)을 이룬다.

QUAD에서는 방사형 삼각화(star shape triangulation)의 결과가 30° 의 하한값을 갖도록 내부의 한 점을 생성할 수 있다. 이러한 작용을 NI(node in interior)라고 하자. NI가 적용될 때 만족되어야 할 조건이 있다. 그 조건을 그림 3을 예로 들어 설명하기로 하자.

NI를 적용하기 전에 필요한 조건은 그림 3에서 다음과 같이 나타낼 수 있다.

$$\theta_4, \theta_5 \geq 30 \quad (1)$$

이 때 L_1 은 $\max(L_1, L_2, L_3, L_4)$ 의 관계식으로 표시된다. 위에 설명된 조건을 삼각형에 있는 각 변의 관계를 이용하여 다시 표시하면 다음과 같다.

$$\frac{d_1^2 + L_3^2 - L_2^2}{d_1 L_3} \geq \sqrt{3}, \quad \frac{d_2^2 + L_3^2 - L_1^2}{d_2 L_3} \geq \sqrt{3} \quad (2)$$

위와 같은 필요 조건을 가지고 NI는 파스칼로

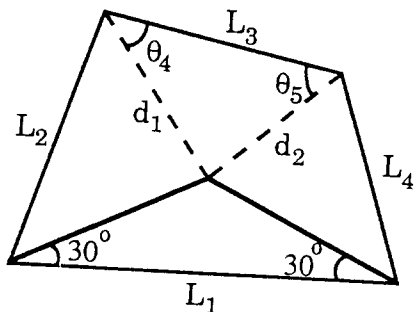


그림 3 NI의 필요조건

표시된 프로그램 2와 같이 작용한다.

프로그램 II

Procedure NI

{*NI finds an interior node location and completes the triangulation*}

Begin

If (LE=IE) and (θ_4, θ_5 in QUAD 30°)

then

Find PCP with 30°

End : {*NI*}

그림 4는 사각형의 한 대각선이 그 사각형 내에서 가장 긴 변일 경우에 NI를 적용하는 방법을 예를 든 것이고, 이 때 사각형의 외변 중 가장 긴 변을 밑변에 위치한 것이다.

NI의 작용을 확인하기 위해서는 NI적용 후 모든 각도들이 30° 보다 커지는 것을 보여야 한다. 또한 NI의 필요 조건이 만족되지 않는 경우에는 θ_4 혹은 θ_5 만이 30° 보다 작아질 수 있다는 것을 보일 수 있다. 자세한 증명은 참고문헌 (9)에 되어있다.

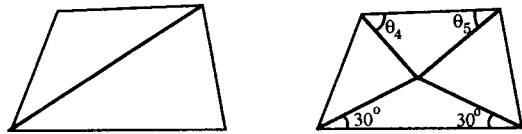


그림 4 NI의 적용예

2.3 ES(Edge Swapping)

LE가 내부변(IE)인 경우에는 QUAD의 내부에 있는 선분을 다른 대각선으로 교체함으로써 가장 작은 각도를 증가시킬 수 있다. 이러한 작용을 ES(edge swapping)이라고 한다.

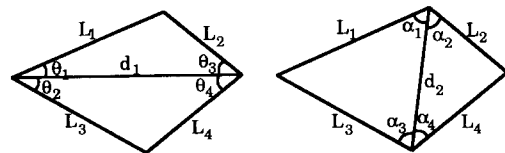


그림 5 ES의 적용예

이 때에 꼭지점, 선분, 삼각형의 숫자는 변하지 않는다. 또한, 이 작용은 NI의 필요 조건이 만족되지 않을 때에도 사용된다.

그림 5에서, d_1 이 LE이고 $\theta_i (i=1...4)$ 가 30° 보다 크면, 대각선을 교체하는 작용(d_1 대신 d_2 를 사용)으로 메쉬의 질을 향상시킬 수 있다. 이 때 다음과 같은 필요조건이 만족되어야 한다.

$$\min(\theta_1, \theta_2, \theta_3, \theta_4) < \min(a_1, a_2, a_3, a_4)$$

위에서 설명한 필요 조건이 만족될 때, ES가 파스칼로 표시된 프로그램 3과 같이 작용한다.

프로그램 III

Procedure ES

{*ES improves the mesh quality by swapping an edge*}

Begin

If LE=BE and $\min\{\theta_1, \theta_2, \theta_3, \theta_4\} < \min(a_1, a_2, a_3, a_4)$ then Swap diagonal in QUAD

End ; {*ES*}

2.4 NR(Node Relocation)

지금까지 설명한 것들(NE, NI and ES) 중에서 NE와 NI만이 노우드를 생성하는데 필요하다. 그러므로 원하는 밀도의 메쉬를 얻기 위해서는 이 두 작용이 반드시 필요하다. NE가 적용될 때에는 LE가 BE이기만 하면 된다. 그러나 NI가 적용되기 위해서는 조건이 필요하다.

NI의 필요 조건이 만족되지 않을 때에 NI를 적용하는 방법을 살펴보기로 하자. 앞에서 설명한 바와 마찬가지로, NI가 적용될 때에는 QUAD 안에 있는 각도들 중에서 30° 의 하한값을 만족하지 못하는 각도의 개수는 많아야 하나이다. NI가 적용된 후 하한값을 만족하지 못하는 경우를 다루기 위하여 보조정리(lemma) 1과 2가 필요하다. 이들 보조정리를 설명하기 위해서 다음과 같은 정의가 필요하다.

정의 2

NI를 이용하여 네 개의 삼각형으로 나뉘어진 QUAD를 가정하자. 외부의 네 변 중 가장 긴 것을 아래쪽에 위치 했을 때 네 삼각형을 상대적인 위치에 따라서 BT (bottom triangle), TT(top triangle), RT (right triangle), LT(left triangle)로 정의한다.

그림 6은 AD가 QUAD의 외부에 있는 네 변 중 가장 긴 변인 경우에 BT, TT, RT, LT를 나타낸 것이다.

보조정리 1과 2를 설명하기 위해서 일반적인 사각형에 대한 다음과 같은 관찰을 할 수 있다.

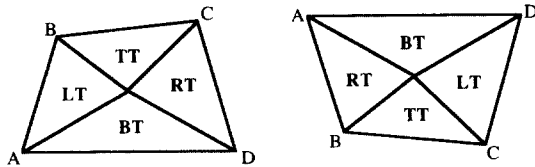


그림 6 QUAD내에서의 NI적용예

관찰

임의의 사각형에서, 가장 긴 변의 양 끝에 접하는 두 각중 적어도 하나는 90° 보다 작다.

이 관찰은 다음 그림을 이용하여 명확하게 알 수 있다. 그림 7에서 AB를 QUAD ABCD

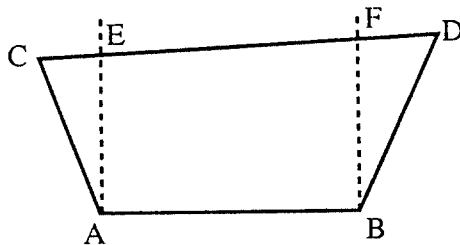


그림 7 LE의 양 끝각이 90° 보다 큰 사각형의 예

의 **LE**라고 하고 $\angle CAB$ 와 $\angle DBA$ 가 90° 보다 크다고 가정하자. 그러면 \overline{CD} 는 \overline{AB} 보다 길게 되므로 앞서한 가정은 모순이다.

— 보조정리 1 —

모든 각도가 30° 보다 큰 **QUAD**가 있다고 가정하자. **NI**를 적용한 후에 30° 의 하한값을 만족하지 못할 때, **TT**의 외부 선분의 수직 이등분선 상에 새로운 노우드를 생성할 수 있다. 이 때에 **BT**의 가장 작은 각도를 30° 되게 만들면 **TT**의 각도들은 15° 보다 크거나 같게 되고 **QUAD** 내에 있는 다른 모든 각도는 30° 보다 크게 된다.

위의 보조정리를 이용하여 **NR**(node relocation)는 파스칼로 표시된 프로그램 4와 같이 정의된다.

— 프로그램 IV —

Procedure NR {*NR finds a new node location when NI fails*}

Begin

If(LE = IE) and $(\min(\theta_4, \theta_5)$ in **QUAD** after **NI** < 30°) **then**

If(Top Edge of **TT**) = (LE in the **QUAD** defined by the edge) **then**

Find the intersection between perpendicular bisection line of the Top Edge of **TT** and 30° supporting line of Bottom Edge of **BT**

Else

Begin

Find second **LE**

LE := Second **LE**

End

End ; {*NR*}

이와 같은 **NR**를 이용하면, **NI**를 적용한 후 30° 의 하한값을 만족하지 않는 삼각형은 다음의 보조정리를 이용하여 없앨 수 있다.

— 보조정리 2 —

NI적용 후 30° 의 하한값을 만족하지 못하는 **QUAD**가 있다고 가정하자. **TT**의 외부변이 그 변을 공유하는 두 삼각형을 합한 사각형 형태에서 가장 긴 변이면, **NR**과 **ES**를 적용함으로써 모든 각도가 30° 의 하한값을 만족하도록 할 수 있다.

보조정리 1과 보조정리 2를 이용하면, **NI**적용 후 30° 의 하한값을 만족하지 않는 경우를 해결할 수 있다. **TT**의 외부변이 그 변을 공유하는 두 삼각형을 합한 사각형 형태에서 가장 긴 변이면, **NR**과 **ES**를 적용함으로써 모든 각도가 30° 의 하한값을 만족하도록 할 수 있다. 그림 8은 **NI**가 하한값을 만족하지 못하는 경우를 수정하는 것을 나타낸다. 그림 8(a)에서 **QUAD**의 한 각도 만이 30° 보다 작다. 그림 8(b)와 (c)에서와 같이 **NR**과 **ES**를 적용하면, 모든 삼각형이 하한값을 만족하도록 할 수 있다.

2.5 이차원에서의 메쉬 생성 알고리즘

지금까지 설명된 네 가지 기본 작용을 이용

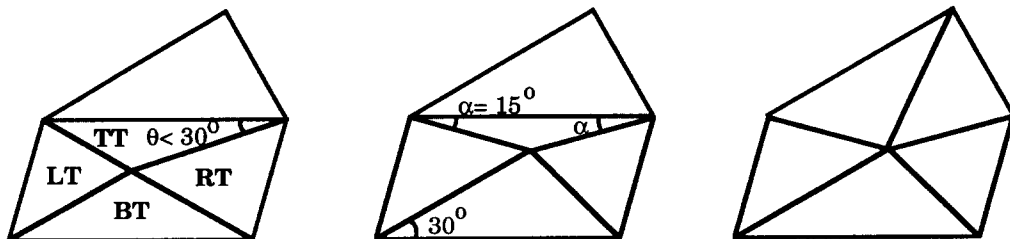


그림 8 NR의 적용예

하여, 이차원에서의 메쉬 생성 알고리즘이 프로그램 5와 같이 개발되었다.

프로그램 V

```

Program Uniform Mesh in 2D ;
{*Generates a uniform mesh with a user
specified element size from an initial tri-
angulation of the object. Element size is
defined by a user specified edge length.*}
Begin
Sort Edges in the current data struc-
ture ;
If (length of LE) < (length of pre-defined
edge) then
    exit
else if LE=BE then
begin
    Apply NE ;
    Update Data Structure ;
    Uniform Mesh in 2D ;
end
else if LE=IE then
begin
    Find QUAD defined by LE ;
If pre-condition of ES=true then
begin
    Apply ES
    Update Data Structure ;
end
else begin
    Find PCP using NI ;
if pre0condition of NI=true then
        Uniform Mesh in 2-D ;
else begin
        Apply NR and ES ;
        Update Data Structure ;
        Uniform Mesh in 2-D ;
end ; {*else*}
end ; {*else*}
    
```

프로그램 5에서의 입출력은 다음과 같다.

입력 : i) 꼭지점들을 이용하여 삼각형으로 나뉘어진 다각형. (가정 : 삼각형으로 나뉘어진 다각형의 모든 각도가 30°보다 크거나 같다.)

ii) 메쉬 내에서 가장 큰 삼각형 요소를 결정하는데 사용 되는 선분의 길이.

출력 : 모든 각도가 30°보다 크거나 같고 모든 선분의 길이가 주어진 선분의 길이 보다 짧은 삼각형 메쉬.

3. 맺음말

이 글에서는 가장 작은 각도의 하한값을 보장하는 이 차원에서의 메쉬 생성 알고리즘이 개발되었다. 지금까지 유한 요소법에서는 어떠한 알고리즘도 메쉬질의 정도를 보장할 수 없었기 때문에 고르기 과정이 항상 필요하였다.

이 알고리즘은 또한 적은 메쉬생성(adaptive mesh generation)에도 쓰일 수 있는데, 이러한 메쉬 생성법에서는 초기 메쉬를 생성한 다음 유한 요소법의 오차를 줄이기 위해 노우드를 하나씩 첨가하거나 제거한다. 따라서 여기서 소개된 알고리즘은 메쉬의 밀도를 조절하는데에도 사용된다.

참고문헌

- (1) Baker, B.S., Grosse, E. and Rafferty, C.S., 1988, "Non-Obtuse Triangulation of Polygons," Discrete and Computational Geometry, 3, pp. 147~168.
- (2) Bykat, A., 1983, "Design of a Recursive, Shape Controlling Mesh Generator," International Journal for Numerical Methods in Engineering, Vol. 19, pp. 1375~1390.
- (3) Cavendish, J.C., Field, D.A., and Frey, W. H., 1985, "An Approach to Automatic Three-Dimensional Finite Element Mesh Generation," International Journal for Numerical Methods in Engineering, Vol. 21,

- pp. 329~347.
- (4) Chew, P.L., 1989, "Guaranteed-Quality Triangular Meshes," TR 89-983, Dept. of Computer Science, Cornell University, Apr.
- (5) Cook, W.A. and Oakes, W.R., 1982, "Mapping Methods for Generating Three-Dimensional Meshes," Computers in Mechanical Engineering, pp. 67~72.
- (6) Field, D.A., 1987, "A Flexible Delaunay Triangulation Algorithm," Research Publication GMR-5675.
- (7) Ho-Le, K., 1988, "Finite Element Mesh Generation Methods : A Review and Classification," Computer-Aided Design, Vol. 20, No. 1 Jan/Feb.
- (8) Lo, S.H., 1985, "A New Mesh Generation Scheme for Arbitrary Planar Domains," International Journal for Numerical Methods in Engineering, Vol. 21, pp. 1403~1426.
- (9) Park, J.Y., 1991, "Mesh Generation with Quasi-Equilateral Triangulation," Ph.D Thesis, Industrial and Operations Engineering Department, The University of Michigan, Ann Arbor, MI.
- (10) Rivara, M.C., 1987, "A Grid Generator Based on 4-Triangles Conforming Mesh-Refinement Algorithms," International Journal for Numerical Methods in Engineering, Vol. 24, pp. 1343~1354.
- (11) Strang, G. and Fix, G., 1976, An Analysis of the Finite element Method, Prentice-Hall.
- (12) Woo, T.C. and Thomasma, T., 1984, "An Algorithm for Generating Solid Elements in Objects with Holes," Computers and Structures, Vol. 18, No. 2, pp. 333~342.
- (13) Wordenweber, B., 1984, "Finite Element Mesh Generation," Computer-Aided Design, Vol. 16, No. 5, pp. 285~291.
- (14) Yerry, M.A. and Shephard, M.S., 1984, "Automatic Three-Dimensional Mesh Generation by the Modified-Octree Technique," International Journal for Numerical Methods in Engineering, Vol. 20, pp. 1965~1990. 