한국 CAD/CAM 학회 논문집
제5권 제1호 2000년 3월 pp. 95-101

학술논문

수정된 Lo의 요소망 생성 알고리즘을 이용한 자동 삼각 요소망 생성 소프트웨어의 개발

김병옥*, 양현익**

The Development of the Automatic Triangular Mesh Generation Software Using Modified Lo's Algorithm

Kim, B.-O.* and Yang, H.-I.**

ABSTRACT

For last two decades numerous automatic mesh generation algorithms for various two dimensional objects have been introduced continuously and among them triangular mesh generation schemes have been majority because of efficiency and controllability. In our study, an existing triangular mesh generation algorithm developed by Lo is totally modified to more improve node distribution, element shape, and objects shape independency. It is composed of node generation part and element generation part. In order to find a suitable node position within geometry, the suggested algorithm searches desirable positions of points within boundary and optimizes node position to generate comparatively well-shaped elements. More over, the suggested algorithm handles various complex two dimensional objects and its meshing speed shows superiority to those of the existing triangulation mesh generation algorithms. It is fully automated in a sense of constructing object boundary and hence can be directly used as an independent meshing software.

Key words : Mesh generation, Triangulation, Node distribution

1. 서 론

설계해석 및 평가에서 유한 요소 해석은 중요한 자리를 매김하고 있다. 그러나 유한요소 해석에 걸리는 시간은 전체 해석과정의 많은 부분을 차지하고 있다. 따라서 유한 요소 해석 시 해석에 필요한 정보를 자동으로 생성하는 과정이 설계 효율을 좌우하는 관건이 되고 있다. 이를 해결하기 위하여 유한요소를 자동으로 생성하기 위한 연구가 1970년대 초반부터 활발히 진행되어 그 중 일부는 상용화되기도 하였지만 보다 효율적인 방법을 개발하기 위하여 계속 많은 연구가 진행되고 있다.

유한 요소의 자동 생성 방법으로는 위상 분해법, 절점 연결법, 기하 분할법, 격자 이용법, 사상법, 적응 사상법이 있다[1]. 여기서 요소의 생성 시 관건이 되는 사항은 요소가 균형적인 형상을 갖추고 있는가 하는 점이다. 형상이 좋지 못한 요소는 해석 시 많은 오차를 야기할 수 있기 때문이다. 이러한 점에서 절점 연결법은 다른 유한 요소 생성법들 보다 이러한 요구 조건에 가장 적합한 특성을 지니고 있다.

절점 연결법으로 물체의 경계와 내부에 절점을 생성하는 방법은 Cavendish의 절점 생성법[2], Lo의 절점 생성법[4-6,10], Lee의 절점 생성법[8], Simada의 절점 생성법[9]들이 있으며 일부는 상용화 되었으나, 보다 효율적인 방법을 개발하기 위하여 연구가 활발히 진행중이다.

본 연구에서는 절점 연결법들 중에서 Lo의 절점 생성 알고리즘을 수정하여 임의의 2차원 형상을 가지는 물체에 대해 요소망을 자동으로 생성할 수 있는 소프트웨어를 개발하였다. 수정된 알고리즘은 기존의 Lo의 알고리즘보다 더 균형된 삼각형 요소망을 생성하게 함으로써 높은 효용성을 보였다.

*학생회원, 한양대학교 기계공학과
**종신회원, 한양대학교 기계산업공학부

2. 절점 생성법

2.1 절점 생성법의 기반 및 수정된 알고리즘

본 연구에서는 Lo의 절점 생성 알고리즘을 수정하여 사용하였다. Lo의 절점 생성 알고리즘이란 원하는 요소와 같은 간격을 갖는 수평선들을 물체의 경계와 교차시킨 후 물체 내부의 각각 선분에 다시 요소의 간격으로 절점을 생성시키는 방법을 말한다. 이 때 수평선들을 물체의 경계와 교차시킬 경우는 식 (1)을, 물체 내부에 절점을 생성시킬 경우는 식 (2)를 이용한다.

$$l' = \frac{Ymax - Ymin}{int\left(\frac{Ymax - Ymin}{l}\right)} \quad (1)$$

l : 초기설정 요소 길이
l' : 수정된 요소 길이.
$int(\ )$: 정수화 함수.
$Ymax$: 경계의 Y의 최대 값.
$Ymin$: 경계의 Y의 최소 값.

$$l'' = \frac{Xmax - Xmin}{int\left(\frac{Xmax - Xmin}{l}\right)} \quad (2)$$

l : 초기설정 요소 길이
l'' : 수정된 요소길이.
$int(\ )$: 정수화 함수.
$Xmax$: 수평선과 교차한 경계의 X의 최대 값.
$Xmin$: 수평선과 교차한 경계의 X의 최소 값.

그러나 이러한 Lo의 절점 생성법은 Fig. 1과 같이 동그라미 영역에 절점이 생성되지 않아 나중에 생성되는 요소의 질이 떨어지게 된다.

Lo는 이러한 점을 보완하기 위하여 Cavendish의 Smoothing Process를 사용하였다. Smoothing Process

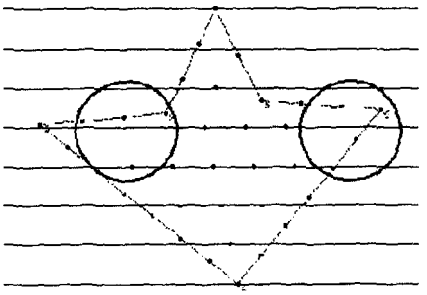
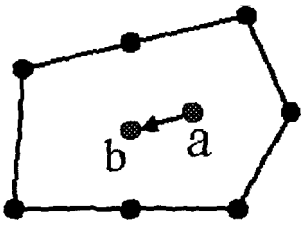


Fig. 1. Result of the mesh points generation using Lo's mesh generation algorithm.



a : 전환전의 절점
b : 전환후의 절점

Fig. 2. Smoothing process.

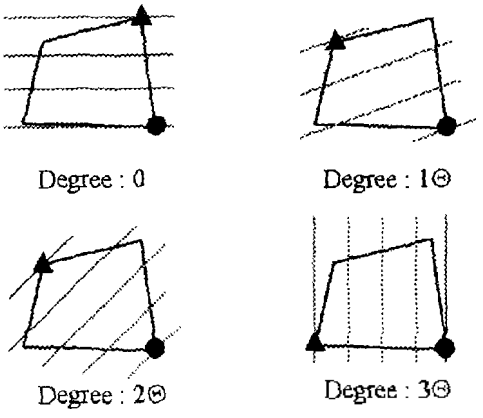
란 Fig. 2와 같이 임의의 절점을 중심으로 주위의 절점들로 이루어진 다각형의 중심으로 절점을 이동시키는 방법을 말한다.

그러나 별 모양과 같은 오목한 형상의 경우 생성되는 절점의 수가 적어서 생성된 요소의 길이가 초기에 설정한 요소길이보다 상당히 길어지는 결과를 가져오게 된다.

따라서 본 연구에서는 이러한 문제점을 해결하기 위해여 대상으로 하는 물체를 수평으로만 나누면서 절점을 생성시키지 않고 Fig. 3에서 보듯이 일정 각도(Θ)로 변화시키면서 물체를 나누는 작업을 수행하면서 절점을 생성시키는 방법을 제시하고자 한다.

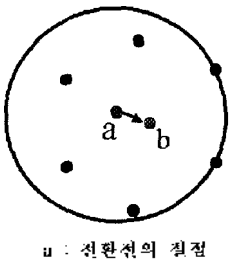
여기서 ▲표시는 좌표전환후의 Y의 최대 값이며 ●표시는 최소 값이다.

이 때, 기존에 생긴 절점들을 저장한 상태에서 새로 생성된 절점들을 저장하게 된다. 그리하여 Fig. 4과 같이 기존에 생성된 절점들과 새로 생성된 절점간의 거리측정을 한 후 새로 생성된 절점을 중심으



Θ : 분할선의 각도

Fig. 3. Searching mesh points by varying line angle.



a : 전환전의 절점
b : 전환후의 절점

Fig. 4. Generation of new mesh points.

로 초기설정 요소길이의 일정 범주 안에 있는 절점들을 모아 절점들 간의 중점으로 절점을 이동시킨다. 이 과정을 반복하면서 기존에 생성된 절점들과 전환전의 절점은 상쇄되어진다.

이 방법은 Cavendish가 제안한 Smoothing process와 비슷해 보이기는 하나 Smoothing process는 새로운 절점을 생성시키지 않아 초기설정 요소길이보다 현저히 길어진 요소를 생성시키게 되는 반면 본 연구에서 제안한 방법은 새로운 절점을 생성시키면서 수행함으로 해서 초기설정 요소길이가 변하지 않으면서도 후에 생성시키는 요소망 또한 균형되게 만들 수 있다는 차이점이 있다.

한편, 무한히 각을 변화시키면서 절점을 생성하는 것이 아니라 절점 생성 시 각각의 절점들과 요소길이의 일정거리 안에 있는 절점들 간의 거리가 초기설정 요소길이의 일정 범주 안에 들 때까지 위의 과정을 반복 수행하도록 하여 생성하는 요소망의 질을 높이도록 하였다. 그 결과는 Fig. 5와 같다.

Fig. 5에서 보면 알 수 있듯이 Fig. 1과 비교해서 문제영역이 보완된 것을 알 수 있다.

여기서, 나누는 각도(θ)가 작으면 후에 생성시키는 요소망의 질이 높아지기는 하나 절점을 찾는 시간이 다소 걸리는 경향이 있으며 각도(θ)가 크면 요소 망의 질은 비교적 떨어지는 반면 속도는 향상되는 경향을 보인다.

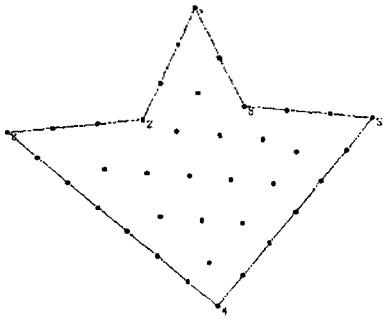


Fig. 5. Result of the mesh points generation using modified Lo's mesh generation algorithm.

2.2 절점 생성 결과

Fig. 6은 위에서 언급한 알고리즘을 사용하여 절점을 생성한 결과들의 예이다.

3. 삼각화 알고리즘

3.1 Delaunay-Watson의 삼각화 알고리즘 및 수정된 알고리즘

본 연구에서는 Delaunay-Watson의 알고리즘을 수

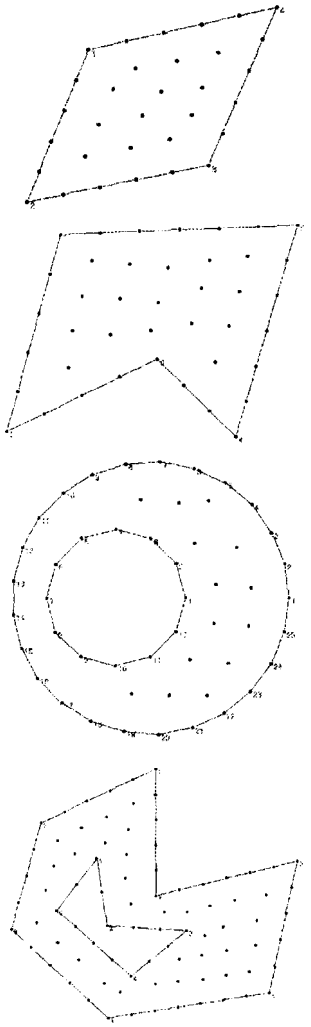
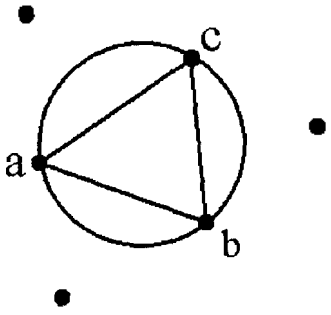


Fig. 6. Generation of inner nodes in various shapes.



a, b, c : 삼각형을 이루는 절점

Fig. 7. Delaunay-Watson's triangulation algorithm.

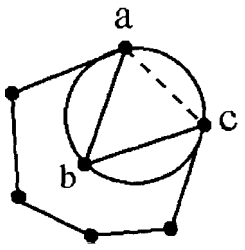
정하여 사용하였으며 기본 개념은 다음과 같다.

2차원의 경우, Delaunay-Watson의 알고리즘의 기본 개념은 물체의 내부에 생성된 절점들 중에서 Fig. 7과 같이 일직선상에 있지 않은 임의의 세 개의 점으로 구성된 삼각형에서 그 외접원 내에 다른 절점이 포함되어 있지 않으면 삼각형을 이룬 각각의 세 점은 Delaunay 삼각형의 조건을 만족하여 삼각형 요소망을 생성시키는 알고리즘을 말한다.

이러한 Delaunay-Watson[1]의 삼각화 알고리즘은 대상 물체의 내부에 생성된 절점들로 이루어질 수 있는 삼각형들의 집합중에서 가장 등변형에 가까운 삼각형 요소망을 이룰 수 있는 장점을 지니고 있으나 Fig. 8에서 보듯이 대상으로 하는 물체의 경계가 오목(concave)이면 어긋난 삼각형 요소망을 생성시키지는 단점과 요소망을 생성하는데 방대한 계산량을 필요로 하여 장시간을 요한다는 단점을 지니고 있다.

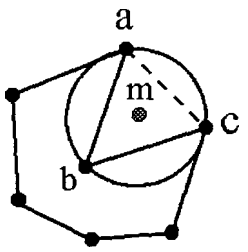
본 연구에서는 Delaunay-Watson의 삼각화 알고리즘이 지니고 있는 이 두 가지 문제점을 수정하여 삼각 요소망을 생성토록 하였다.

첫째로 오목한 모양의 경계의 경우, Fig. 9에서와



a, b, c : 삼각형을 이루는 절점

Fig. 8. Case of concave boundary.



a, b, c : 삼각형을 이루는 절점
m : 삼각형의 중점

Fig. 9. Determination of trianglulation at concave boundary.

같이 요소망을 이루는 삼각형의 중점이 대상으로 하는 물체의 내부에 존재할 시에만 삼각형 요소 망을 형성함으로써 해결하였다.

삼각형의 중점이 물체의 내부인지 외부인지는 Fig. 10과 같이 요소 망을 이루는 삼각형의 중점을 포함하고 X축과 평행한 임의 가상 선을 그어 교차의 정도를 평가하였다. 교차된 횟수가 홀수이면 내부를 교차된 횟수가 0이거나 짝수이면 외부를 나타내게 된다.

두 번째로 발생하는 삼각형 요소망 생성에 장시간을 요하는 문제점은 물체에 생성된 모든 점들을 비교함으로 해서 발생한다. 이 문제점을 해결하기 위하여 본 연구에서는 다음과 같은 해결 방법을 개발하였다.

절점 생성 후 삼각형 요소를 이루는 점들은 임의의 점을 중심으로 초기 설정 요소 길이(initial mesh length)보다 많이 벗어나지 못한다. 따라서 생성된 모든 절점들을 비교하지 않고 주위에 있는 절점들만을 따로 저장해 두었다가 비교를 수행함으로 해서 요소 생성 소요 시간을 현저히 줄였다.

Fig. 11과 같이 이상적인 삼각형 요소인 정삼각형의 경우 임의의 점을 중심으로 절점은 6개가 존재하게 된다. 따라서 생성된 모든 절점들을 비교하지 않고 주위의 최대 6개만을 비교한다. 따라서, 삼각화시의 계산속도가 현저히 줄어드는 것을 알 수 있었고,

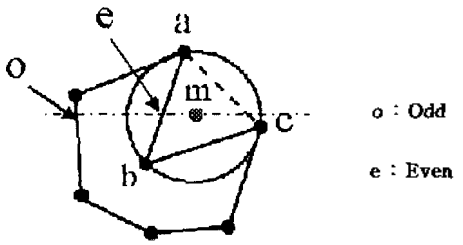


Fig. 10. Checking center point in triangle.

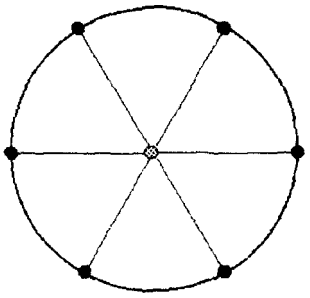


Fig. 11. Number of mesh points for an ideal triangle.

Table 1. Comparison of calculation time

	계산횟수
Delaunay-Watson	n*(n-1)*(n-2)*n
Modified	n*(n-1)*6*5*6

n : 절점의 개수

이에 대한 Delaunay-Watson의 삼각화 알고리즘과의 계산 횟수의 비교는 Table 1과 같다.

Delaunay-Watson의 계산횟수는 내부에 생성된 n개의 절점들 중에서 3개의 점을 임의로 선택하고[n*(n-1)*(n-2)] 이 점들로 이루어진 외접원 내에 다른 절점들의 존재여부를 판단하여[*n] 얻어진다. 본 연구에서 제안한 방법의 계산횟수는 내부에 생성된 절점들 중에서 임의의 한점을 선택하고[n] 이 절점의 주위에 있는 절점들을 찾는다[*(n-1)]. 그리고, 찾은 절점들 중에서 두 점을 선택해서[*6*5] 초기에 선택한 한 점과 후에 선택한 두점으로 이루어진 외접원내에 다른 주위 절점의 존재여부를 판단하여 얻어진다[*6].

3.2 요소망 생성 결과

Fig. 12는 요소망 생성의 결과이다. 여기서 절점 생성시와 요소망 생성 후의 절점의 수가 틀린 것은 프로그램의 속도 향상을 위하여 절점세분화(Refinement)를 수행하였기 때문이다.

4. 소프트웨어의 개발

4.1 소프트웨어의 구성

본 연구에서 개발한 소프트웨어는 Visual Basic 5.0으로 작성하였으며 Microsoft Window 95상에서 운영된다. 최소한의 사용자 입력만이 필요하도록 함으로서 사용자가 사용하기 편리하도록 작성하였으며 각 부분별 소스 코드를 참고 또는 수정사용하기 편

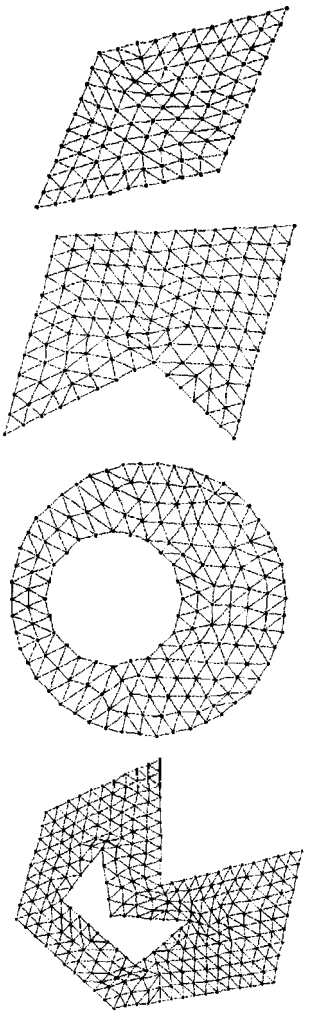


Fig. 12. Final mesh generation in various shapes.

하도록 각 단계를 모듈화 하였다.

본 연구에서 개발한 소프트웨어는 크게 다섯 단계로 이루어진다. 첫 단계는 방향에 관계없이 키보드 혹은 마우스를 통하여 경계를 입력한다. 두 번째 단계에서는 원하는 요소의 크기를 입력한다. 다음 단계로 입력받은 두 가지 정보, 즉 경계의 정보와 원하는 요소의 크기를 가지고 대상 물체의 경계 내부에 절점을 생성시키게 된다. 그 다음 단계에서는 물체의 내부에 생성된 절점들로 삼각형의 요소망을 생성하게 된다. 마지막 단계로서 생성된 절점 및 경계의 위치, 소요 시간, 작동 횟수 등을 프로그램 상에서 출력한다.

4.2 소프트웨어의 실행

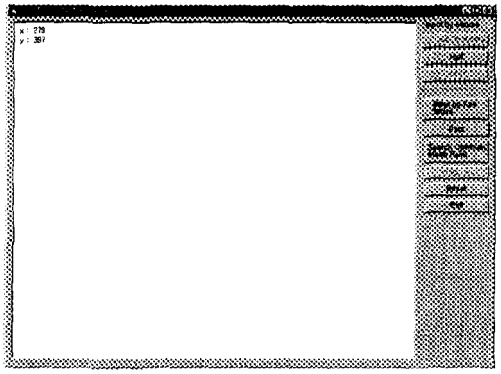


Fig. 13. Initial window.

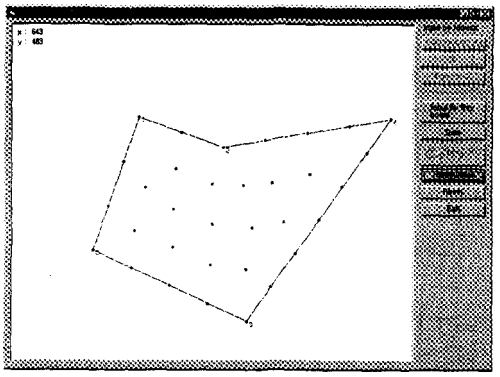


Fig. 16. Generating inner nodes.

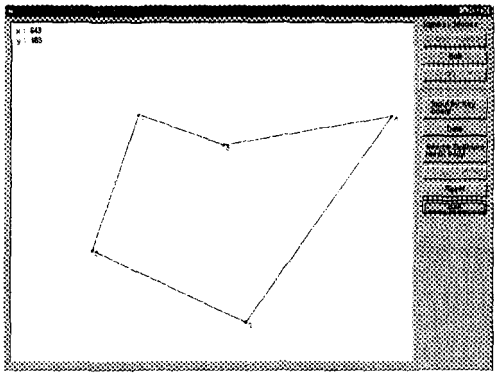


Fig. 14. Input of boundary points.

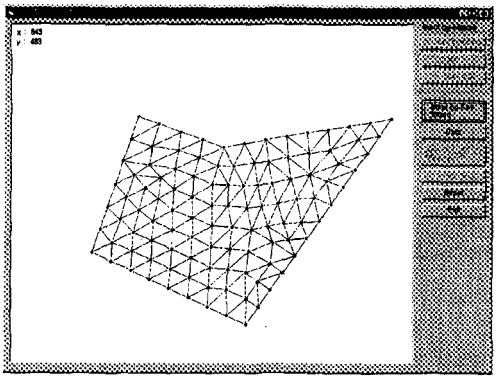


Fig. 17. Generating mesh.

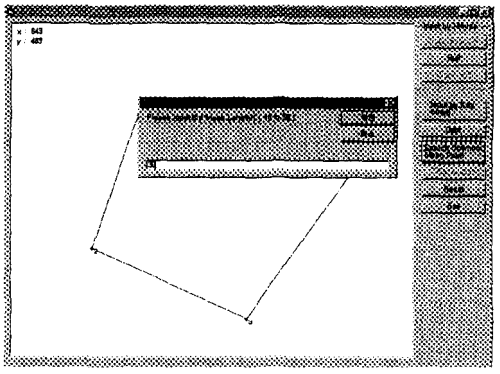


Fig. 15. Input of the desired mesh length.

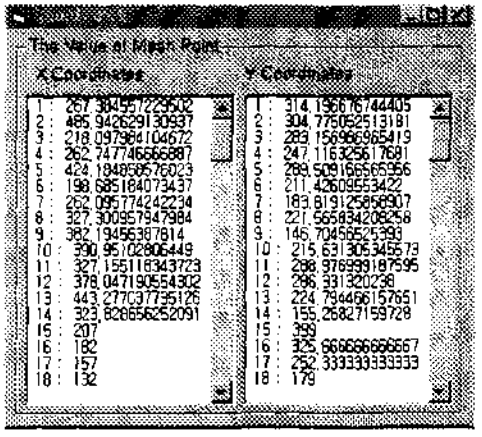


Fig. 18. Mesh Data

다음 그림들은 소프트웨어를 통해 자동으로 요소망을 생성하는 과정을 나타낸다. Fig. 13은 프로그램의 시작화면이며, Fig. 14는 경계를 입력한 상태를 나타내고 있다. Fig. 15는 경계입력 후 요소의 길이를 입력하는 창이며, Fig. 16은 대상 물체의 경계내부에 절점을 생성한 결과이고 Fig. 17은 요소망 생성의 결과이다. Fig. 18은 각 종 데이터 값을 나타내며, Fig. 19는 절점들의 좌표 값이다. Fig. 20은 사용자 편의를 돕기 위하여 키보드로 입력하는 창을 보이고 있다.

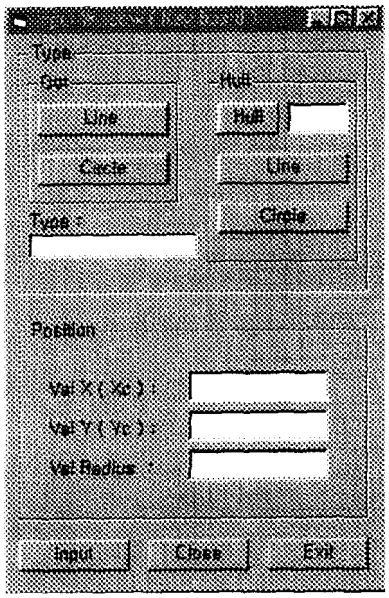


Fig. 19. Coordinates X, Y of the generated mesh points.

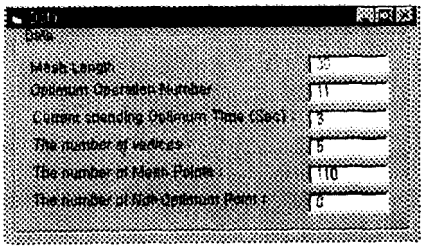


Fig. 20. Window of the various data.

5. 결 론

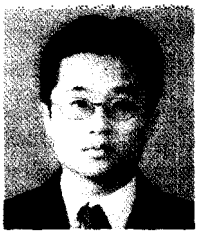
본 연구에서는 Lo의 알고리즘을 수정하여 보다 좋은 형상을 지니는 요소를 생성시키는 방법을 제시하였고, 삼각화 시 발생하는 문제점을 보완 수정하였다.

개발된 알고리즘을 통하여 최소의 정보, 즉 경계의 위치 및 요소의 크기만을 입력하면 해석하고자하는 이차원 형상의 내부에 균형된 요소를 보다나은 속도로 자동 생성토록 하였다. 이에 따라, 유한 요소 해석 시 가장 많은 시간과 노력이 요구되는 요소 망의 생성을 자동화함으로서 해석 시의 노력을 덜어줄 수 있다고 판단된다.

향후 본 소프트웨어에 3차원 형상에 대한 모듈이 추가되어진다면, 해석을 수행하는 분야에 생산성 및 능률향상에 기여할 수 있을 것으로 기대되어 진다.

참고문헌

1. 이건우, "컴퓨터 그래픽과 CAD", 영지문화사.
2. Visual Basic 5.0 Bible, 인포북.
3. Visual Basic 5.0 그대로 따라하기, 혜지원.
4. S. H. Lo, "A new mesh generation scheme for arbitrary planar domains", *International Journal for Numerical Methods in Engineering*, Vol. 21, 1403-1426, 1985.
5. S. H. Lo, "Delaunay triangulation of non-convex planar domains", *International Journal for Numerical Methods in Engineering*, Vol. 38, 943-954, 1995.
6. S. H. Lo, "Automatic mesh generation and adaption by using contours", *International Journal for Numerical Methods in Engineering*, Vol. 24, 237-357, 1987.
7. J. C. Cavendish, "Automatic triangularation of arbitrary planar domains for the finite element method", *International Journal for Numerical Methods in Engineering*, Vol. 9, 679-696, 1974.
8. Y. T. Lee, "Automatic finite element generation based on constructive solid geometry", 1983.
9. K. Shinada and D. C. Gossard, "Computational methods for physically based FE mesh generation", *Human Aspects in Computer Integrated Manufacturing*, North-Holland, 1992.
10. S. H. Lo, A. Y. T. Leung and Y. K. Cheung, "Automatic finite element mesh generation", Beijing, China, 931-937, 1982.



김 병 욱

1997년 한양대학교 기계공학과 학사
1999년 한양대학교 기계공학과 석사
1999년~현재 한양대학교 기계공학과 박사 과정
관심분야: CAD, CAM, Mesh generation

양 현 익

1984년 한양대학교 기계설계학과 학사
1986년 Columbia University 기계공학과 석사
1991년 Columbia University 기계공학과 박사
1991년~1995년 현대전자 산업전자연구소 책임연구원
1995년~현재 한양대학교 공학대학 기계산업공학부 조교수
관심분야: CAD, CAM, CAE, 유한요소생성론