

구조해석과 설계과정의 통합시스템에 관하여

Toward a System of Integrating Structural Analysis and Design Procedures

이 주 성* 오 석 진**
Lee, Joo-Sung Oh, Seuk-Jin

ABSTRACT

This paper is concerned with the integrated computer program system aiming at efficiently performing the structural analysis and design. The developed computer program system is introduced and applied to a simple two-dimensional structure to show the general concept of the integrated system. Some design modifications and re-analyses are illustrated including local mesh refinement. These show the efficiency in doing design modification and analysis.

1. 서 론

잘 알려진 바와 같이 선박이나 해양구조물과 같은 복잡한 구조물의 설계에 있어서 구조해석은 중요한 과정 중 하나이다. 이러한 구조물의 구조설계는 통상적으로 선급규칙 (Classification Society)에서 정하는 설계코드 (Design Code)에 따라 초기 설계를 수행하고 이어서 유한요소해석 프로그램을 이용한 전체 및 국부구조해석과 그 결과에 기초하여 상세설계가 따르게 되며, 이러한 과정은 구조적 안전성 측면에서 만족할 만한 결과가 얻어질 때까지 반복하게 된다. 현재 실무현장에서는 상용화된 구조해석 프로그램을 사용하고 있고 이에 는 대개 전처리과정과 후처리과정 (pre-processor and post-processor) 이 포함되어 있기는 하지만 구조해석과 설계과정은 서로 별개로 나뉘어져 있어서 두과정 사이의 정보교환이 기본적으로 수작업으로 이루어지고 있기 때문에 최종적인 설계결과를 얻기까지는 상당히 많은 시간과 노력을 필요로 한다. 최근 컴퓨터의 급속한 발달로 워크스테이션은 물론 개인컴퓨터 (personal coputer:PC)에서도 계산속도가 대단히 빠르기 때문에 한 워크스테이션이나 PC 내에서 위의 두과정을 어렵지 않게 처리할 수 있게 되었다. 따라서 지금보다 구조해석과 설계과정을 하나의 시스템 내에서 관리할 수 있는 통합시스템 (integrated system)의 개발은 설계에 소요되는 시간과 노력의 감소에 큰 효과를 볼 수 있을 것이다. 이러한 통합시스템은 그 효율화를 위해서 GUI (graphic user interface) 의 개념을 도입해야 할 것이다 [1]. 또한 최근 구조물의 형태가 다양해 지고 있는 추세이므로 이러한 통합시스템은 구조물의 형태 마다 구조해석 및 설계전용으로 사용할 수 있는 시스템으로서 유용하게 사용될 것이다.

* 울산대학교 조선 및 해양공학과 교수
** 울산대학교 조선 및 해양공학과 석사과정

구조해석과 설계에는 일반적으로 다음의 과정들이 포함된다.

- (1) 설계과정
- (2) 구조해석과정
- (3) 안전성평가 과정

설계과정은 초기구조설계와 상세설계과정이 포함된다. 구조해석과정에는 유한요소를 생성하고 구조해석을 수행하며 그 결과를 표현하는 내용이 포함되는데, 이는 잘 알려진 바와같이 전처리과정, 후처리과정 그리고 유한요소방정식을 구성하고 그 해를 구하는 과정을 포함한다. 마지막 것은 Solver라고 불리우는데, 상용화된 프로그램이나 개발된 다른 프로그램을 이용할 수 있을 것이다. 안전성평가는 확정적 또는 확률적인 평가방법을 적용하여 수행하게 된다. 통합시스템은 이러한 여러 과정들을 하나의 시스템 내에서 관리함으로써 설계에 관련된 일련의 과정들을 효율적으로 수행하는 것으로 정의할 수 있겠다.

이 논문은 위에서 언급한 통합시스템의 자세한 내용을 다루기 보다는 통합시스템의 개념을 다루고 있는데, 이를 위해서 본 연구진이 개발하고 있는 통합시스템을 소개하고 이의 적용을 보여주고자 한다. 본 연구의 통합시스템은 기본적으로 GUI의 개념을 도입함으로써 구조물의 기하학적 형상이나 구조치수 등을 손쉽게 수정하고 구조해석을 수행할 수 있도록 구성되어 있으며 상용화된 Solver를 활용할 수 있도록 구축되어 있다. 본 연구의 통합시스템은 간단한 2차원 구조물에 적용하여 통합시스템의 기본개념과 개발된 프로그램의 몇가지 기능을 보여 주었다.

2. 통합시스템의 일반

본연구에서 개발하고 있는 통합시스템을 ISAD: Integrated Computer Program System for Analysis and Design of Structures 라 하였다. ISAD는 기본적인 3차원 그래픽 기능을 가지고 있고 Multi-Window System을 채택하고 있다. 그림 1은 ISAD에서 Window의 배치를 보여 주고 있는데 Drawing Area와 Tool Area로 크게 나뉘어져 있다. Drawing Area에는 구조물의 전체적인 모양과 세부적인 모양을 나타내고 또한 구조해석결과를 표현하며 구조물의 기하학적 형상을 변형시키고 필요에 따라 부재의 치수변경을 하게 된다. Tool Area는 말 그 자체대로 내장된 기능을 사용하기 위해 명령을 기능을 선택하는 Menu Plate들이 나열되어 있다. 모든 명령은 Window의 상단에 있는 Menu Bar에서 Menu를 선택거나 Tool Area의 Menu Plate에서 필요한 기능을 선택함으로써 이루어 지도록 구성하였는데, 이로서 사용자가 시스템의 기능을 쉽게 이해하고 사용할 수 있다. 그림 2는 통합시스템의 전반적인 흐름을 보여 주고 있는데, 이는 구조해석결과를 바탕으로 구조신뢰성해석을 수행하여 안전성을 평가하는 과정 등 필요한 과정들을 삽입할 수 있다.

다루는 구조물은 초기구조설계결과로부터 6개 또는 8개의 Keynode를 갖는 Region으로 자동적으로 구성되고 요소자동생성과정 (automatic mesh generation process) 을 거쳐서 요소가 생성되기 때문에 이를 준비하는 시간을 상당히 절약할 수 있을 것이다. 또한 Region의 형상이나 생성된 유한요소의 크기는 Mouse를 이용해서 Keynode들을 이동시키거나 필요에 따라 부재를 추가 또는 제거 할 수 있기 때문에 이의 과정을 용이하게 수행할 수 있다. 그림 3에 두 Region의 형태를 나타내었고, 그 형상함수는 유한요소법에 관한 많은 서적에서 볼 수 있다 (예로서 참고문헌 2, 3 등).

본연구의 통합프로그램 시스템은 매킨토시의 Micro-Soft Fortran Compiler 환경에서 개발하였고 [4], 그래픽스 툴은 매킨토시 ROM에 내장된 Toolbox Routine을 이용하였다 [5].

3. 적용 예

개발된 통합시스템, ISAD의 적용 예를 보여주기위해 그림 4에 보인 외팔보를 구조 모델로 선택하였다. 하중은 자유단에 5 ton이 작용하고 있다. 여기에서 편의상 어떤 과정을 거쳐서 초기구조설계가 수행되었다고 가정한다.

2.1 초기설계에 대한 구조해석

그림 4의 외팔보를 그림 5와 같이 두개의 8 Keynode Region으로 구성하였다. Region의 각 변에 8개씩의 절점을 갖도록 3절점 선형 삼각형 요소로 분할한 결과를 그림 5(a)에 보여 주었다 (물론 분할과정에서 Tool Area에서 다른 형태의 요소를 선택하여 분할할 수 있다). 잘 알려진 바와같이 구조해석의 결과는 변형된 형상이나 응력의 분포 또는 응력이 높은 부분이나 안전성이 낮은 부분(critical part)을 나타 낼 수 있는데, 여기에서는 편의상 응력이 어느 수준 이상이 되는 부분을 표현하였다. 이를 Critical Part라 하였고, 이 Critical Part는 여기에서 편의상 평면응력상태에서 아래의 식으로 주어지는 등가응력이 재료의 항복응력의 80% 이상이 되는 경우로 정의 하였다.

$$\sigma_{eq} = \sqrt{\sigma_x^2 + \sigma_y^2 - \sigma_x \sigma_y + 3\tau_{xy}^2}$$

2.2 구조형상의 변형

그림 5에서 보는 바와같이 고정단에서 높은 응력이 작용하고 있는데, 이근처에서 응력을 낮추기 위해서는 Bracket를 추가하는 것이 자연스러운 일이다. 이는 곧 구조형상을 변형 시키는 것이고, 다음의 몇가지 방법으로 구현할 수 있다.

(1) Modification-1 : Keynode 의 이동에 의한 구조형상의 변형

위에서 기술한 대로 구조형상을 변형 하는 방법은 그림 5 (a)에서 Keynode 9와 10을 연직방향의 적절한 위치로 이동시킴으로서 Bracket를 추가하는 것이 되도록 하는 것이다. 그림 6(a)는 이 과정을 거친 후 Region의 정의를 보여주는데, Keynode 13역시 적절한 위치로 이동시켰다. 그림 6(b)는 구조형상을 변형시킨 후 유한요소 생성결과와 Critical Part를 보여 주는데, 쉽게 예측할 수 있듯이 외팔보의 고정단 근처에서의 응력이 많이 완화되어 강도가 향상 되었음을 볼 수 있다.

(2) Modification-2 : Region 의 추가에 의한 구조형상의 변형

구조형상을 변형하는 또 다른 방법으로서 Region을 관심있는 부분 근처에 추가하는 것이다. 그림 7(a)는 8-Keynode Region을 고정단 근처에 추가로 정의함으로서 그림 6(a)와 같은 형상으로 변형한 것인데, 그림 7(b)에 생성된 요소의 형태와 구조해석의 결과로서 Critical Part들을 보여 주었다. 그림 8은 그림 7과 마찬가지로 Region을 하나 더 추가한 것인데, 8-Keynode Region 대신에 6-Keynode Region으로 정의한 것이 다른 점이다. 생성된 요소의 형태와 Critical Part들을 앞에서와 마찬가지로 나타내었다. 그림 6-8에서 구조해석의 결과는 유사함을 볼 수 있는데, 외팔보의 Main Part (그림 4 자체)와 Bracket의 연결부분에서 여전히 높은 응력이 작용하고 있어서 이를 더 완화시킬 목적으로 삼각형 Bracket 대신에 원형 Bracket 를 추가할 수 있을 것이다. 그림 9는 원형 Bracket 를 6-Keynode Region 으로 정의한 경우를 보여주는데, 예상과는 달리 큰 응력완화효과가 없는 것을 볼 수 있다. 그러나 이러한 구조형상을 변형시키는 과정은 구조해석 자체를 보여주는 것이 목적이 아니라, 구조해석을 수행하고 그 결과를 참고하여 설계자가 원하는 바대로 구조형상을 변형시키는 일예를 보여주는 것이다.

2.3 요소의 국부적 상세분할 (local mesh refinement)

앞절에서는 한 Region 내에서 생성되는 요소의 크기가 같도록 자동분할한 것인데, 구조해석결과로부터 더 관심있는 부분에서 보다 자세한 응력분포상태를 알고자하거나 또는 전체구조해석의 결과를 이용해서 국부구조부재에 대한 상세구조해석을 수행하고자 할 때 국부적으로 요소를 상세히 분할해야 할 필요가 있다. 이를 위한 손쉬운 방법은 Region을 정의하는 Keynode들을 관심있는 부분으로 이동시켜서 그 부분 근처를 더 작은 요소로 분할하는 것이다. 그림 10은 그 예로서, 그림 9에서 외팔보의 Main Part와 Bracket의 연결부분 근처로 Keynode들을 이동시켜서 재분할하는 것을 보여주고 있다. 이로써 관심있는 부분에서의 응력분포를 더 자세히 분석할 수 있을 것이다.

국부적으로 요소를 상세히 분할하는 또 하나의 방법은 국부적 상세분할법 (local mesh refinement technique)을 적용하여 관심있는 부분 근처를 보다 더 작은 요소로 분할하는 것이다. 그림 11은 참고문헌 6에서 적용한 국부분할방법으로 그림 9 또는 10과 같은 형상을 갖는 외팔보의 고정단 상하부분과 Main Part-Bracket 연결부분을 상세분할한 결과를 보여 주고 있다. 여기에 Error Estimation법을 적용한다면 보다 부드러운 요소 분할결과를 얻을 수 있다.

본연구의 통합프로그램시스템인 ISAD는 위에서 기술한 내용 외에도 Mouse로 선택한 Region의 치수를 변형하는 기능들을 포함하고 있는데, 이러한 통합시스템은 설계를 담당하는 실무자가 구조해석과 설계를 하나의 시스템 내에서 편리하게 수행할 수 있을 것이다.

3. 결 언

본논문은 구조해석과 설계과정을 하나의 시스템으로 통합하는 통합시스템의 개념과 개발된 프로그램을 소개하였다. 이러한 통합시스템의 개념을 적용하면 실무를 담당하는 설계자가 구조해석과 설계를 수행하는 시간과 노력을 절약할 수 있어서 이의 수행에 상당한 이점을 가질 수 있다. 또한 이러한 통합프로그램시스템은 학부생 또는 대학원 생들이 구조해석과 설계에 대한 내용을 쉽게 익힐 수 있도록 하는 교육용으로도 유용하게 사용될 수 있다. 본연구에서 개발된 프로그램을 실제 구조물에 대해 적용하는 것은 추후 우수한 학회 등을 통해 발표할 예정이다.

후 기

이 연구는 89년도 한국과학재단 특정기초연구비 지원에 의한결과이다
(과제번호 : KOSEF 91-02-00-06).

참고문헌

1. Rogers, D.F. and Adams, J.A., *Mathematical Elements for Computer Graphics*, McGraw-Hill, 1989
2. Zienkiewicz, O.C. and Taylor, R.L., *The Finite Element Method*, 4th ed., McGraw-Hill International Editions, 1989
3. 임상전, 곽병만, 이주성, *유한요소법입문*, 동명사, 1985
4. *Microsoft FORTRAN Compiler for the Apple Macintosh*, Microsoft Corporation, 1985
5. Apple Computer Inc., *Inside Macintosh: vol.I-III*, Addison-Wesley, 1985
6. Rivara, M.C., "A Gid Generator Based on 4-triangles Conforming Mesh-Refinement Algorithms", *Intl. J. for Numerical methods in Engineering*, Vol.24, 1987, pp.1343-1354

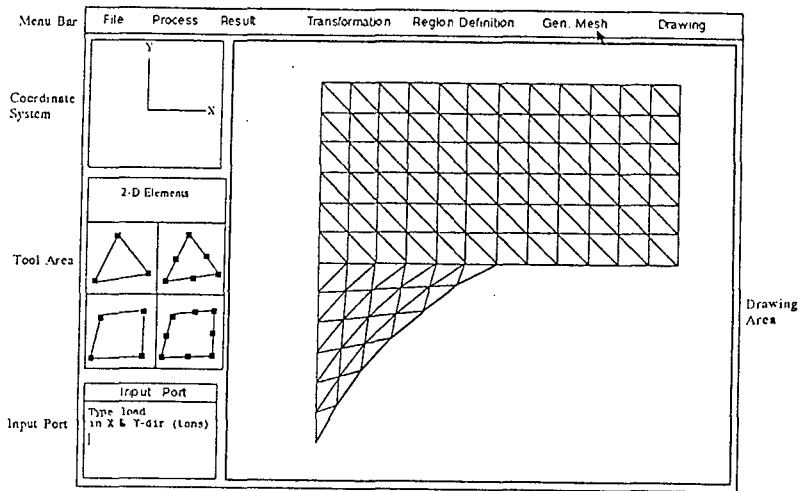


Figure 1. Window arrangement in ISAD

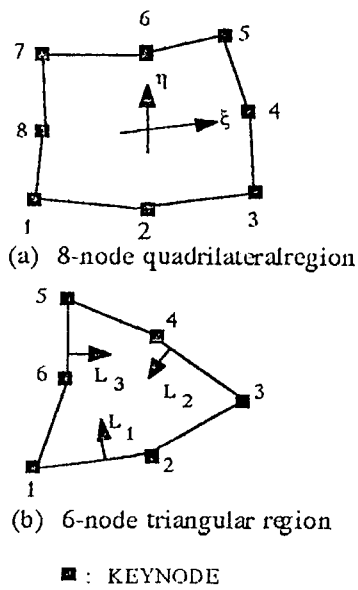


Figure 2. Types of regions used

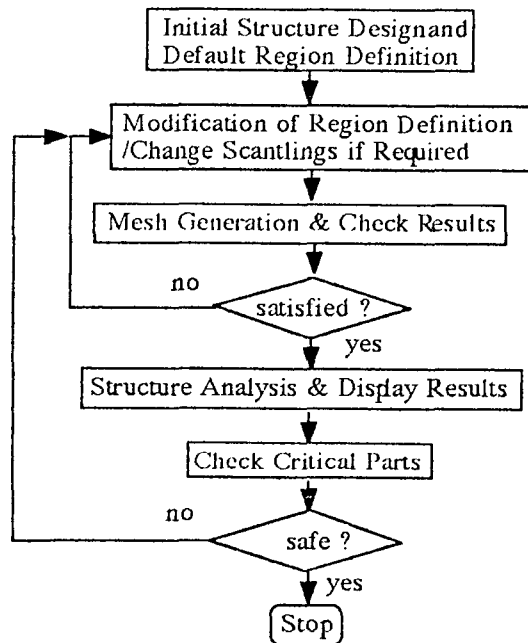


Figure 3. General flow of ISAD

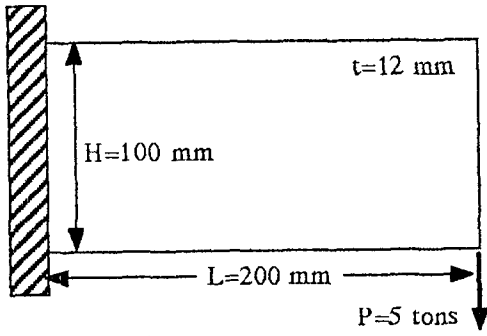
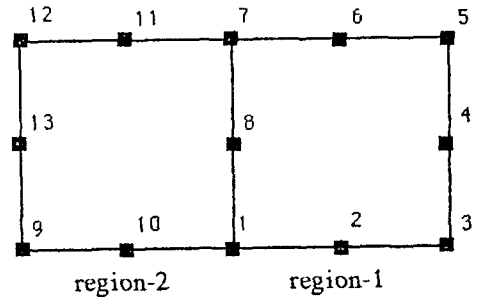
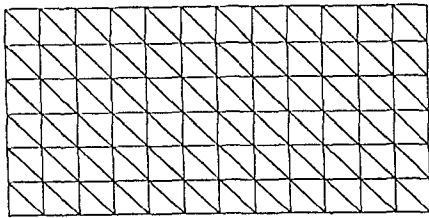


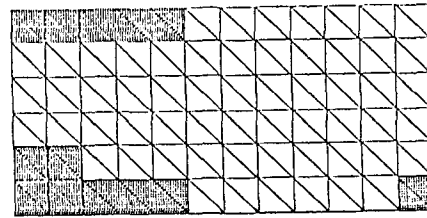
Figure 4. Cantilever loaded at free end



(a) region definition

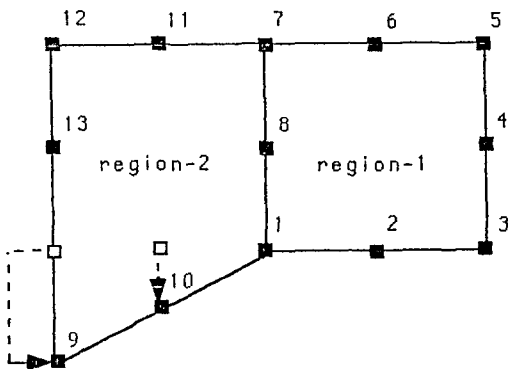


(b) generated mesh

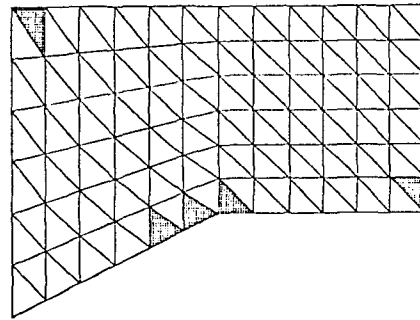


(c) structural analysis results: critical parts

Figure 5. Analysis of Initial Design

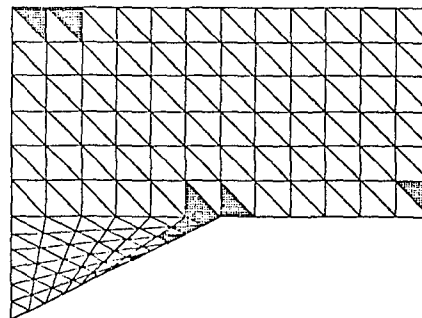
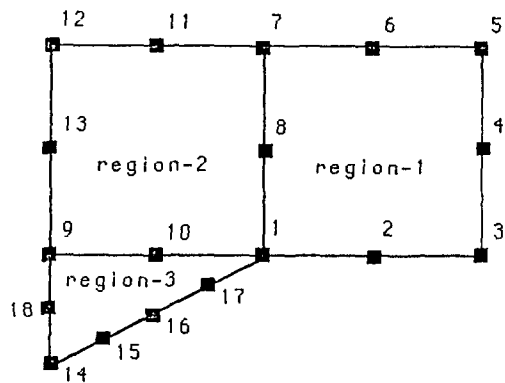


(a) region definition



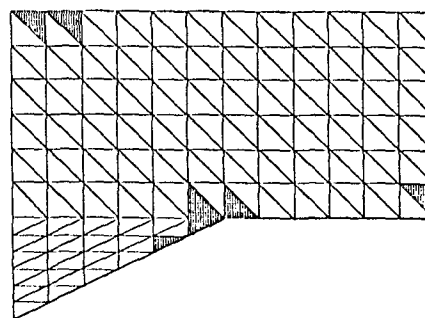
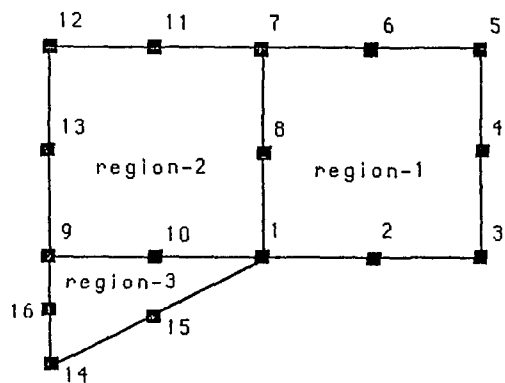
(b) structural analysis results: critical parts

Figure 6. Modification-1: moving keynodes



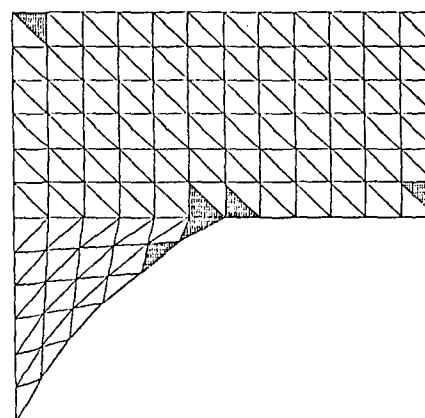
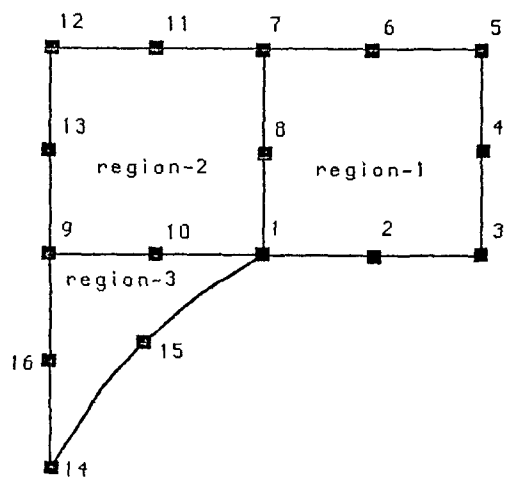
(a) region definition
Figure 7. Modification-2: adding 8-node region

(b) structural analysis results: critical parts



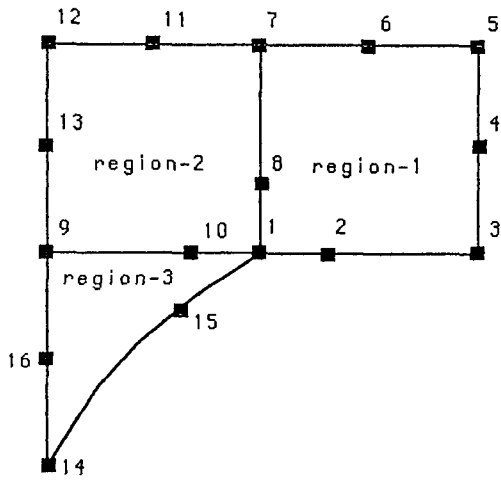
(a) region definition
Figure 8. Modification-2: adding 6-node region

(b) structural analysis results: critical parts

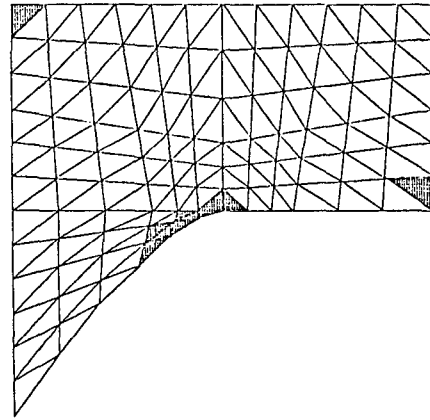


(a) region definition
Figure 9. Modification-2: adding rounded shape region

(b) structural analysis results: critical parts



(a) region definition



(b) structural analysis results: critical parts

Figure 10. Local mesh refinement by moving keynodes

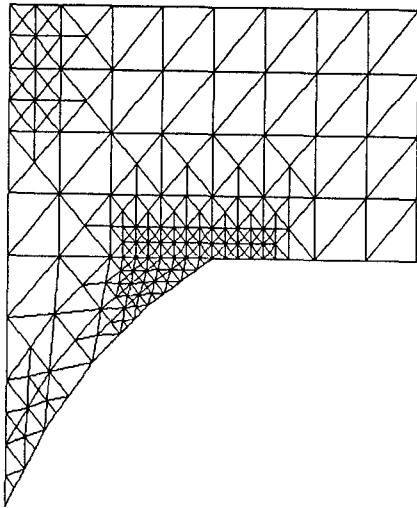


Figure 11. Local mesh refinement by refinement technique in reference 4