

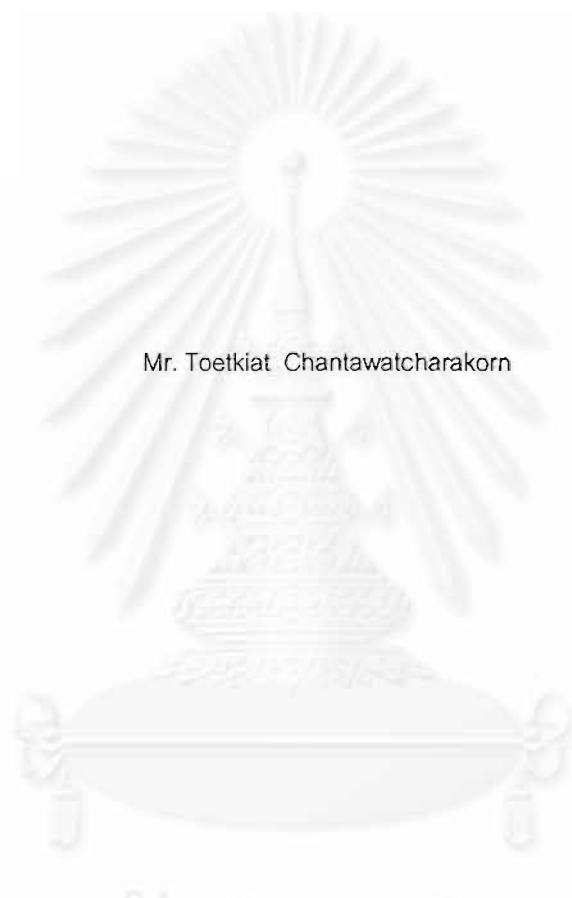
โปรแกรมฟอนต์เอเลเมนต์ที่ปรับขนาดขึ้นส่วนอัตโนมัติ



นายเทอดเกียรติ จันทวีชรากร

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย
วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมโยธา ภาควิชาวิศวกรรมโยธา
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2542
ISBN 974-334-056-4
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

AN AUTOMATIC ADAPTIVE REMESHING FINITE ELEMENT PROGRAM



Mr. Toetkiat Chantawatcharakorn

สถาบันวิทยบริการ

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering in Civil Engineering
Department of Civil Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 1999

ISBN 974-334-056-4

หัวข้อวิทยานิพนธ์

โปรแกรมไฟไนต์เอลิเมนต์ที่ปรับขนาดชิ้นส่วนอัดโนมัติ

โดย

นาย เทอดเกียรติ จันทร์ชรากร

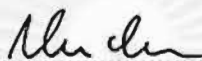
ภาควิชา

วิศวกรรมโยธา

อาจารย์ที่ปรึกษา

ศาสตราจารย์ ดร. ทักษิณ เทพชาตรี


คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้บัณฑิตวิทยาลัยฉบับนี้เป็นส่วนหนึ่งของการ
ศึกษาตามหลักสูตรปริญญาโทบัณฑิต



คณบดีคณะวิศวกรรมศาสตร์

(ศาสตราจารย์ ดร. สมศักดิ์ ปัญญาแก้ว)

คณะกรรมการสอบวิทยานิพนธ์



ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. เริงเดชา รัชตโพธิ์)

ประธานกรรมการ



อาจารย์ที่ปรึกษา
(ศาสตราจารย์ ดร. ทักษิณ เทพชาตรี)

อาจารย์ที่ปรึกษา



กรรมการ
(อาจารย์ ดร. ชัชชาติ สิทธิพันธุ์)

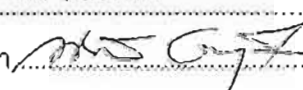
กรรมการ

นายเทอดเกียรติ จันทร์ชรากร : โปรแกรมไฟไนต์เอลิเมนต์ที่ปรับขนาดชิ้นส่วนอัตโนมัติ (AN AUTOMATIC ADAPTIVE REMESHING FINITE ELEMENT PROGRAM) อ. ที่ปรึกษา: ศาสตราจารย์ ดร. ทักษิณ เทพชาติศรี 93 หน้า. ISBN 974-334-056-4.

การวิเคราะห์ด้วยระเบียบวิธีไฟไนต์เอลิเมนต์ให้มีการกระจายความคลาดเคลื่อนออกไปเท่า ๆ กันตลอดทั้งโดเมนของปัญหาจะทำให้ ได้ผลเฉลยที่ดีที่สุดเทียบกับโครงข่ายที่มีระดับชั้นความเร็วเท่ากัน ซึ่งการที่จะกระจายความคลาดเคลื่อนนี้จำเป็นต้องใช้เวลาและแรงงานจำนวนมากในการเตรียมข้อมูล ดังนั้นในงานวิจัยนี้จึงเสนอการพัฒนาโปรแกรมไฟไนต์เอลิเมนต์ที่ปรับขนาดชิ้นส่วนอัตโนมัติ ซึ่งจะทำการสร้างชิ้นส่วนด้วยวิธีการสร้างสามเหลี่ยมเดลอนโดยใช้ฟรอนท์คืบหน้า(Advancing Front Delaunay Triangulation) และทำการประมาณค่าความคลาดเคลื่อนของผลการวิเคราะห์จากการเปรียบเทียบกับผลเฉลยที่ได้จากวิธีการฟื้นตัวเป็นหย่อม (Patch Recovery) พร้อมทั้งปรับขนาดของชิ้นส่วนให้สอดคล้องกับความคลาดเคลื่อน ด้วยการเพิ่มจุดลงในบริเวณที่มีความคลาดเคลื่อนสูงให้โดยอัตโนมัติ ผลการวิจัยแสดงให้เห็นว่าสามารถลดการเตรียมข้อมูลลงได้มาก และได้ผลการวิเคราะห์ที่มีความคลาดเคลื่อนอยู่ในช่วงที่ยอมรับได้โดยมีความคลาดเคลื่อนสัมพัทธ์ในช่วง 1 - 5 เปอร์เซ็นต์

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา วิศวกรรมโยธา
สาขาวิชา วิศวกรรมโยธา
ปีการศึกษา 2542

ลายมือชื่อนิสิต เสนอถึงท่านที่ปรึกษา
ลายมือชื่ออาจารย์ที่ปรึกษา 

4070290221 : MAJOR CIVIL ENGINEERING

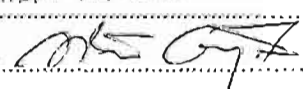
KEY WORD: FEM / ADAPTIVE / MESH GENERATION / DELAUNAY TRIANGULATION

TOETKIAT CHANTAWATCHARAKORN : THESIS TITLE. AN AUTOMATIC ADAPTIVE REMESHING FINITE ELEMENT PROGRAM THESIS ADVISOR : PROF. THAKSIN THEPCHATRI Ph.D., 93 pp. ISBN 974-334-056-4.

An analysis with the finite element method gives the best solution when errors are distributed equally to the whole domain. Commonly, it is time-consuming and requires a great amount of work in preparing data for a complicated problem. This thesis proposes "An Automatic Adaptive Remeshing Finite Element Program" that generates meshes by Advancing Front Delaunay Triangulation and estimates the errors of the analysis by comparing with the results obtained from the Patch Recovery method. The program will self-adjust the given mesh size to the suitable mesh size corresponding to the errors by inserting nodes in the high error zones. Results of the study show that the developed program facilitates the data preparation and yields results with a relative error of 1 - 5 percent.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา วิศวกรรมโยธา
สาขาวิชา วิศวกรรมโยธา
ปีการศึกษา 2542

ลายมือชื่อนิสิต เพ็ญเกียรติ ไกรธรรมา
ลายมือชื่ออาจารย์ที่ปรึกษา 



กิตติกรรมประกาศ

ผู้เขียนขอกราบขอบพระคุณศาสตราจารย์ ดร. ทักษิณ เทพชาตรี อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ได้สละเวลาอันมีค่า แนะนำ ตรวจสอบ แก้ไขข้อบกพร่องต่าง ๆ ที่เกิดขึ้นในการวิจัย พร้อมทั้งช่วยเหลือในการติดต่อและให้ข้อมูลอันเป็นประโยชน์ ขอกราบขอบพระคุณผู้ช่วยศาสตราจารย์ ดร. เริงเดชา รัชตโพธิ์ และ อาจารย์ ดร. ชัชชาติ สิทธิพันธุ์ ที่ได้กรุณาให้ความสนใจในการเป็นคณะกรรมการตรวจสอบวิทยานิพนธ์ ขอกราบขอบพระคุณอาจารย์ ดร. บุญชัย อุกฤษฏชน ที่กรุณาเอื้อเฟื้อข้อมูลที่เป็นประโยชน์จำนวนมาก ขอขอบคุณเจ้าหน้าที่ห้องสมุดและเจ้าหน้าที่ประจำเครื่องถ่ายเอกสารประจำห้องสมุดต่าง ๆ ที่ให้ความสะดวกในการยืมคืนหนังสือ และยืมคืนต่างสถานศึกษา รวมทั้งให้ความสะดวกในการถ่ายสำเนาข้อมูลต่าง ๆ

สุดท้ายนี้ผู้เขียนขอขอบพระคุณ บิดา มารดา และ ครูบาอาจารย์ทุกท่าน ที่ให้การ ดูแล อบรม สั่งสอน ประสิทธิ์ประสาทวิชาความรู้ต่าง ๆ ให้กับผู้เขียนตลอดมากระทั่งถึงปัจจุบัน



สารบัญ

หน้า

บทคัดย่อภาษาไทย	ง
บทคัดย่อภาษาอังกฤษ	จ
กิตติกรรมประกาศ	ฉ
สารบัญ	ช
สารบัญภาพ	ฅ
บทที่ 1 บทนำ	
1.1 ความนำ	1
1.2 งานวิจัยที่เกี่ยวข้อง	2
1.3 วัตถุประสงค์ของงานวิจัย	4
1.4 ขอบเขตการศึกษา	4
1.5 ผลที่คาดว่าจะได้รับ	4
บทที่ 2 แนวทางและทฤษฎีที่ใช้ในการวิจัย	
2.1 ความนำ	5
2.2 การประมาณค่าคลาดเคลื่อน	5
2.3 การประมาณค่าผลเฉลี่ยจากผลของระเบียบวิธีไฟไนต์เอลิเมนต์	8
2.4 การประมาณขนาดของชิ้นส่วน	12
2.5 การสร้างชิ้นส่วนโดยอัตโนมัติ	13
2.6 การแก้ระบบสมการ	15
บทที่ 3 การพัฒนาโปรแกรม	
3.1 ภาษาคอมพิวเตอร์ที่ใช้ในโปรแกรม	24
3.2 การเลือกใช้โครงสร้างข้อมูล	25
3.3 ชนิดของข้อมูล	26
3.4 โครงสร้างและลำดับการทำงานของโปรแกรม	28
3.5 การแสดงผล	30

สารบัญ (ต่อ)

หน้า

บทที่ 4 ตัวอย่างการวิเคราะห์ผล	
4.1 ตัวอย่างที่ 1 ปัญหาผ่านบางรับแรงดึง	39
4.2 ตัวอย่างที่ 2 ปัญหาผ่านวงกลมบางรับแรงกระทำเป็นจุด	42
บทที่ 5 สรุปผลการวิจัย	
5.1 สรุปผลการวิจัย	84
5.2 ข้อเสนอแนะอื่น ๆ	85
รายการอ้างอิง	86
ภาคผนวก ก การใช้งานโปรแกรม	90
ภาคผนวก ข รายละเอียดของรหัสคำนวณ	92
ประวัติผู้เขียน	93

สารบัญภาพ

หน้า

รูปที่ 2.3.1	ตำแหน่งแก๊สในหย่อมของจุดต่อที่ต้องการประมาณค่า	20
รูปที่ 2.3.2	การประมาณค่าตอบจากตำแหน่งแก๊สในชั้นส่วนเชิงเส้น 1 มิติ	20
รูปที่ 2.3.3	ลักษณะของชั้นส่วนบริเวณขอบของปัญหา	21
รูปที่ 2.3.4	ลักษณะของหย่อมที่ใช้ในการประมาณค่าหน่วยแรงที่จุดต่อที่ขอบ	21
รูปที่ 2.3.5	ลักษณะของจุดต่อที่หาหย่อมภายในได้ยาก	21
รูปที่ 2.5.1	โครงข่ายโคโนนอยและสามเหลี่ยมเดอลอน	22
รูปที่ 2.5.2	คุณสมบัติมุมเล็กสุดที่ใหญ่ที่สุด	22
รูปที่ 2.5.3	ขั้นตอนการสร้างสามเหลี่ยมด้วยการเชื่อมต่อจุด	23
รูปที่ 2.5.4	ขั้นตอนการเพิ่มจุดควบคุมกับการสร้างสามเหลี่ยมเดอลอน	23
รูปที่ 3.2.1	โครงสร้างข้อมูลชนิดแถวลำดับ	31
รูปที่ 3.2.2	โครงสร้างข้อมูลชนิดลิงค์ลิสต์	31
รูปที่ 3.2.3	การจัดเก็บข้อมูลในโครงสร้างข้อมูล ADT	32
รูปที่ 3.3.1	การจัดเก็บข้อมูลของเมตริกซ์	33
รูปที่ 3.4.1	โครงสร้างการทำงานของโปรแกรมหลัก	34
รูปที่ 3.4.2	โครงสร้างการทำงานของการสร้างชั้นส่วนสามเหลี่ยม	35
รูปที่ 3.4.3	โครงสร้างการทำงานของการหาจุดที่เหมาะสมกับเส้นในการสร้างสามเหลี่ยม	36
รูปที่ 3.4.4	โครงสร้างการทำงานของการตรวจสอบการเพิ่มจุด	37
รูปที่ 3.4.5	โครงสร้างการทำงานของการปรับขนาดเส้นขอบและกำหนดระยะห่างของจุดภายใน	38
รูปที่ 4.1.1	ปัญหาแผ่นบางรับแรงดึงและแบบจำลองการวิเคราะห์	44
รูปที่ 4.1.2	โครงข่ายชั้นส่วนสามเหลี่ยมในปัญหาแผ่นบางรับแรงดึง	45
รูปที่ 4.1.3	การกระจายของหน่วยแรงในทิศทาง x ในปัญหาแผ่นบางรับแรงดึง	46
รูปที่ 4.1.4	การกระจายของหน่วยแรงในทิศทาง y ในปัญหาแผ่นบางรับแรงดึง	47
รูปที่ 4.1.5	การกระจายของหน่วยแรงเฉือนในปัญหาแผ่นบางรับแรงดึง	48
รูปที่ 4.1.6	กราฟแสดงค่าหน่วยแรงในทิศทาง x ตามแนว AB ในปัญหาแผ่นบางรับแรงดึง	49
รูปที่ 4.1.7	กราฟแสดงค่าหน่วยแรงในทิศทาง y ตามแนว AB ในปัญหาแผ่นบางรับแรงดึง	50
รูปที่ 4.1.8	กราฟแสดงค่าหน่วยแรงเฉือนตามแนว AB ในปัญหาแผ่นบางรับแรงดึง	51
รูปที่ 4.1.9	โครงข่ายชั้นส่วนสามเหลี่ยมจากโครงข่ายเริ่มต้นที่หยดเกินไป ในปัญหาแผ่นบางรับแรงดึง	52
รูปที่ 4.1.10	โครงข่ายชั้นส่วนสามเหลี่ยมจากโครงข่ายเริ่มต้นที่ละเอียดเกินไป	53
รูปที่ 4.1.11	การกระจายของหน่วยแรงในโครงข่ายสุดท้าย จากการกำหนดโครงข่ายเริ่มต้นละเอียดเกินไป	54
รูปที่ 4.1.12	โครงข่ายชั้นส่วนสามเหลี่ยมที่ได้จากการปรับขนาดโดยตรง	55
รูปที่ 4.1.13	การกระจายของหน่วยแรงในโครงข่ายสุดท้าย จากการปรับขนาดโดยตรง	56

สารบัญภาพ (ต่อ)

	หน้า
รูปที่ 4.1.14	โครงข่ายชั้นสามเหลี่ยม จากการลดความคลาดเคลื่อนยอมให้ 57
รูปที่ 4.1.15	การกระจายของหน่วยแรงในโครงข่ายสุดท้าย จากการลดความคลาดเคลื่อนยอมให้ 58
รูปที่ 4.1.16	เปรียบเทียบหน่วยแรงในทิศทาง x ในโครงข่ายสุดท้ายจากการวิเคราะห์ด้วยข้อกำหนดต่าง ๆ 59
รูปที่ 4.1.17	เปรียบเทียบหน่วยแรงในทิศทาง y ในโครงข่ายสุดท้ายจากการวิเคราะห์ด้วยข้อกำหนดต่าง ๆ 59
รูปที่ 4.1.18	เปรียบเทียบหน่วยแรงเฉือนในโครงข่ายสุดท้ายจากการวิเคราะห์ด้วยข้อกำหนดต่าง ๆ 59
รูปที่ 4.2.1	ปัญหาแผ่นวงกลมบางรับแรงกระทำเป็นจุด และแบบจำลองการวิเคราะห์ 60
รูปที่ 4.2.2	โครงข่ายชั้นสามเหลี่ยมในปัญหาแผ่นวงกลมบาง (จำลองแรงกระทำเป็นจุด) 61
รูปที่ 4.2.3	การกระจายของหน่วยแรงในทิศทาง x ในปัญหาแผ่นวงกลมบาง (จำลองแรงกระทำเป็นจุด) 62
รูปที่ 4.2.4	การกระจายของหน่วยแรงในทิศทาง y ในปัญหาแผ่นวงกลมบาง (จำลองแรงกระทำเป็นจุด) 63
รูปที่ 4.2.5	การกระจายของหน่วยแรงเฉือนในปัญหาแผ่นวงกลมบาง (จำลองแรงกระทำเป็นจุด) 64
รูปที่ 4.2.6	กราฟแสดงค่าหน่วยแรงในทิศทาง y ตามแนว AB ในปัญหาแผ่นวงกลมบาง (จำลองแรงกระทำเป็นจุด) 65
รูปที่ 4.2.7	แบบจำลองการวิเคราะห์ปัญหาแผ่นวงกลมบาง (จำลองเป็นแรงแผ่กระจาย) 66
รูปที่ 4.2.8	โครงข่ายชั้นสามเหลี่ยมในปัญหาแผ่นวงกลมบาง (จำลองเป็นแรงแผ่กระจาย) 67
รูปที่ 4.2.9	การกระจายของหน่วยแรงในทิศทาง x ในปัญหาแผ่นวงกลมบาง (จำลองเป็นแรงแผ่กระจาย) 68
รูปที่ 4.2.10	การกระจายของหน่วยแรงในทิศทาง y ในปัญหาแผ่นวงกลมบาง (จำลองเป็นแรงแผ่กระจาย) 69
รูปที่ 4.2.11	การกระจายของหน่วยแรงเฉือนในปัญหาแผ่นวงกลมบาง (จำลองเป็นแรงแผ่กระจาย) 70
รูปที่ 4.2.12	กราฟแสดงค่าหน่วยแรงในทิศทาง y ตามแนว AB ในปัญหาแผ่นวงกลมบาง (จำลองเป็นแรงแผ่กระจาย) 71
รูปที่ 4.2.13	เปรียบเทียบรายละเอียดโครงข่ายจากแรงกระทำเป็นจุดและแรงแผ่กระจาย 72
รูปที่ 4.2.14	โครงข่ายชั้นสามเหลี่ยมที่ได้จากการปรับขนาดโดยตรง และจำลองเป็นแรงแผ่กระจาย 73
รูปที่ 4.2.15	การกระจายของหน่วยแรงในโครงข่ายสุดท้าย จากการปรับขนาดโดยตรง และจำลองเป็นแรงแผ่กระจาย 74
รูปที่ 4.2.16	โครงข่ายชั้นสามเหลี่ยมจากการจำลองแรงเป็นรูปสามเหลี่ยม 75
รูปที่ 4.2.17	การกระจายของหน่วยแรงในโครงข่ายสุดท้าย จากการจำลองแรงเป็นรูปสามเหลี่ยม 76

สารบัญญภาพ (ต่อ)

หน้า

รูปที่ 4.2.18	โครงข่ายขึ้นสามเหลี่ยมที่ได้จากการปรับขนาดโดยตรง และจำลองแรงเป็นรูปสามเหลี่ยม	77
รูปที่ 4.2.19	การกระจายของหน่วยแรงในโครงข่ายสุดท้าย จากการปรับขนาดโดยตรง และจำลองแรงเป็นรูปสามเหลี่ยม	78
รูปที่ 4.2.20	โครงข่ายขึ้นสามเหลี่ยมที่ได้จากการจำลองแรงเป็นรูปสามเหลี่ยม และลดความคลาดเคลื่อนยอมให้	79
รูปที่ 4.2.21	การกระจายของหน่วยแรงในโครงข่ายสุดท้าย จากการจำลองแรงเป็นรูปสามเหลี่ยม และลดความคลาดเคลื่อนยอมให้	80
รูปที่ 4.2.22	โครงข่ายขึ้นสามเหลี่ยมที่ได้จากการปรับขนาดโดยตรง จำลองแรงเป็นรูปสามเหลี่ยม และลดความคลาดเคลื่อนยอมให้	81
รูปที่ 4.2.23	การกระจายของหน่วยแรงในโครงข่ายสุดท้าย จากการปรับขนาดโดยตรง จำลองแรงเป็นรูปสามเหลี่ยม และลดความคลาดเคลื่อนยอมให้	82
รูปที่ 4.2.24	เปรียบเทียบผลการวิเคราะห์จากการเปลี่ยนข้อกำหนดต่าง ๆ	83
รูปที่ ผ.ก.1	ตัวอย่างแฟ้มนำเข้าข้อมูล	91

บทที่ 1

บทนำ



1.1 ความนำ

การวิเคราะห์ปัญหาด้วยระเบียบวิธีไฟไนต์เอลิเมนต์ เป็นการวิเคราะห์โดยการสมมติปัญหาที่ต่อเนื่องให้มีลักษณะเป็นชิ้นส่วนย่อย ๆ ประกอบกัน (discretization) และใช้สมการคณิตศาสตร์จำลองพฤติกรรมของชิ้นส่วนย่อยเหล่านั้น ร่วมกับการวิเคราะห์พฤติกรรมของชิ้นส่วนทั้งหมดร่วมกัน เราสามารถหาค่าประมาณของผลเฉลยของปัญหาต่อเนื่องได้ โดยความแม่นยำของผลเฉลยนั้นขึ้นอยู่กับแบบจำลองทางคณิตศาสตร์ และ ขนาดของชิ้นส่วนที่ประกอบกัน ยิ่งแบบจำลองคณิตศาสตร์มีความสอดคล้องกับพฤติกรรมของชิ้นส่วนมากเท่าไร ความแม่นยำในการเป็นตัวแทนของชิ้นส่วนก็จะดีมากขึ้นเท่านั้น ในขณะที่การเพิ่มจำนวนชิ้นส่วนซึ่งเป็นการลดขนาดของช่วงการประมาณ ก็จะลดความคลาดเคลื่อน ที่เกิดจากการประมาณปัญหาต่อเนื่องด้วยชิ้นส่วน

ในการเพิ่มความแม่นยำของผลเฉลย ทำได้โดยการเพิ่มความซับซ้อนของสมการคณิตศาสตร์ เพื่อให้สอดคล้องกับพฤติกรรมมากขึ้น หรือโดยการเพิ่มจำนวนของชิ้นส่วน จากทั้ง 2 วิธีนี้ การเพิ่มความสอดคล้องของสมการคณิตศาสตร์นั้นทำได้ยาก ทั้งในด้านการวิเคราะห์หาสมการคณิตศาสตร์ที่เหมาะสม และการนำมาเขียนรหัสคำนวณ ดังนั้นจึงใช้สมการคณิตศาสตร์ที่สลับซับซ้อนเฉพาะในปัญหาที่ต้องการความแม่นยำของคำตอบมาก ๆ เท่านั้น ในโครงการทั่ว ๆ ไปนิยมใช้สมการคณิตศาสตร์ที่ไม่ซับซ้อนมากนัก แต่ใช้ชิ้นส่วนจำนวนมากแทน

ในปัญหาหนึ่ง ๆ จะมีค่าของผลเฉลยแตกต่างกันไปตามโดเมน (Domain) ของปัญหา ในบริเวณใดที่มีอัตราการเปลี่ยนแปลงของผลเฉลยต่อพื้นที่มาก ก็จะมี ความคลาดเคลื่อนของผลเฉลยอยู่มากเช่นกัน ดังนั้นในการที่จะวิเคราะห์ปัญหาให้มีประสิทธิภาพที่สุด จำเป็นต้องกระจายความคลาดเคลื่อนเหล่านี้ให้มีค่าใกล้เคียงกันตลอดทั้งโดเมนของปัญหา ซึ่งทำได้โดยการใช้ชิ้นส่วนที่มีขนาดเหมาะสมกับพื้นที่ของปัญหา โดยใช้ชิ้นส่วนขนาดเล็กในพื้นที่ที่มีความคลาดเคลื่อนมาก และใช้ชิ้นส่วนขนาดใหญ่ในบริเวณที่มีความคลาดเคลื่อนน้อย การที่จะสร้างชิ้นส่วนให้ได้ดังที่กล่าวมานี้ จำเป็นต้องมีการเตรียมข้อมูลและทำการคำนวณหลายรอบ ซึ่งต้องใช้แรงงานและเวลาจำนวนมาก

วิทยานิพนธ์นี้มุ่งพัฒนาโปรแกรมไฟไนต์เอลิเมนต์ที่ปรับขนาดชิ้นส่วนอัตโนมัติ ซึ่งจะประมาณค่าคลาดเคลื่อนของผลเฉลย และทำการเพิ่มจำนวนชิ้นส่วนในบริเวณที่มีความคลาดเคลื่อนมากให้โดยอัตโนมัติ มีผลให้ลดปริมาณการเตรียมข้อมูลลงได้มาก และได้คำตอบที่รวดเร็วมีความคลาดเคลื่อนในขนาดที่ยอมรับได้

1.2 งานวิจัยที่เกี่ยวข้อง

การปรับขนาดชิ้นส่วนในการวิเคราะห์ด้วยระเบียบวิธีไฟไนต์เอลิเมนต์นั้น มีขั้นตอนที่สำคัญคือการหาความคลาดเคลื่อนของผลเฉลย เพื่อจะได้ลดขนาดของชิ้นส่วนในบริเวณที่มีความคลาดเคลื่อนมาก วิธีดั้งเดิมซึ่งมีค่าใช้จ่ายสูงจะทำโดยการเปรียบเทียบผลเฉลยจากการวิเคราะห์ซ้ำเมื่อมีการลดขนาดชิ้นส่วนตลอดทั้งโดเมนของปัญหา

ในช่วงปี 1975 - 1986 Babuska^[1,2] และคณะ ได้คิดค้นและพัฒนาการประมาณค่าคลาดเคลื่อน โดยวิธีเศษตกค้างของแรง ซึ่งเป็นการประมาณค่าคลาดเคลื่อนจากการ อินทิเกรตค่ากำลังสองของเศษตกค้างของหน่วยแรงในสมการอนุพันธ์ และกำลังสองของผลต่างของหน่วยแรงระหว่างชิ้นส่วน วิธีของ Babuska นี้ต้องใช้คณิตศาสตร์ที่ยุ่งยากในการพิสูจน์และคำนวณหาค่าคลาดเคลื่อน จึงไม่สะดวกในการใช้งาน

ต่อมาในปี 1987 Zienkiewicz^[3,4] ซึ่งเป็นหนึ่งในคณะผู้ร่วมงานของ Babuska ได้ร่วมกับ Zhu พัฒนาวิธีการประมาณค่าหน่วยแรงที่ดีขึ้น จากหน่วยแรงที่ได้จากไฟไนต์เอลิเมนต์ และประมาณค่าคลาดเคลื่อนจากการเปรียบเทียบหน่วยแรงทั้งสอง หลักการของวิธีนี้คือ ค่าหน่วยแรงที่ได้จากระเบียบวิธีไฟไนต์เอลิเมนต์ จะมีค่าไม่ต่อเนื่องกันระหว่างชิ้นส่วน ดังนั้นถ้ามีการปรับเรียบให้หน่วยแรงมีความต่อเนื่องกันก็จะได้คำตอบที่ดีขึ้น การปรับเรียบใช้วิธีกำลังสองน้อยสุด (Least square) วิธีนี้ใช้หลักการง่าย ๆ และมีการคำนวณไม่ยุ่งยากนัก เหมาะสำหรับการเขียนรหัสคำนวณ อย่างไรก็ตาม ค่าประมาณของหน่วยแรงที่ได้จากวิธีนี้มีความถูกต้องน้อย และมีการลู่ออกคำตอบซ้ำ โดยเฉพาะในแบบจำลองคณิตศาสตร์ที่มีระดับชั้นของพหุนามสูง

ปี 1990 Zienkiewicz และ Zhu^[5,6,7,8] ยังคงร่วมกันพัฒนาการประมาณค่าหน่วยแรงให้มีความถูกต้องมากขึ้น และมีอัตราการลู่ออกคำตอบเร็วขึ้น โดยใช้หลักการของตำแหน่งลู่ออกยิ่งยวด (Superconvergent Points) หรือตำแหน่งเกาส์ (Gauss point) ในแต่ละชิ้นส่วน ซึ่งมีค่าหน่วยแรงที่มีความแม่นยำกว่าในตำแหน่งอื่น ๆ และมีอัตราการลู่ออกคำตอบมากกว่าระดับชั้นของพหุนามในสมการคณิตศาสตร์ 1 ระดับ ($O(h^{p+1})$) โดยวิธีนี้การประมาณค่าหน่วยแรงที่ตำแหน่งใด ๆ ก็ได้จากการประมาณจากหน่วยแรงที่ตำแหน่งเกาส์รอบ ๆ ตำแหน่งที่ต้องการประมาณนั้น โดยใช้วิธีกำลังสองน้อยสุด วิธีนี้มีหลักการที่สามารถเข้าใจได้ง่ายและมีความแม่นยำมากพอสมควร ประกอบกับการคำนวณไม่มากนักจึงเป็นที่นิยมใช้กันอย่างแพร่หลายจนกระทั่งถึงปัจจุบัน

ในระยะหลัง ๆ มีนักวิจัยหลายกลุ่มพยายามที่จะพัฒนาวิธีของ Zienkiewicz และ Zhu ให้ประมาณค่าหน่วยแรงได้แม่นยำขึ้นโดยเฉพาะในบริเวณขอบเขตของปัญหา โดยการเพิ่มข้อกำหนดในเรื่องความสมดุลของหน่วยแรงเพิ่มเติมเข้าไปในวิธีของ Zienkiewicz และ Zhu ดังเช่น Blacker, Belytschko (1994)^[9] ได้เพิ่มความสมดุลของแรงเชิงปริมาตร (body forces) และ แรงเชิงผิว (traction) ในขณะที่ Wiberg, Adulwahab (1993), (1994)^[10,11] นอกจากเพิ่มพจน์ของแรงทั้ง 2 นี้แล้วยังเพิ่มข้อกำหนดในเรื่องเงื่อนไขขอบเขตด้านการเคลื่อนที่ หรือเงื่อนไขค่าขอบเขตจำเป็น (Essential boundary conditions) เข้าไปในสมการด้วย ต่อมา Lee T, Park, Lee S.W. (1997)^[12] ได้เพิ่มความสมดุลของแรงในลักษณะคล้ายกับ Blacker, Belytschko แต่อยู่ในรูปของพลังงานเสมือน จึงสามารถนำมาประยุกต์กับชิ้นส่วนที่มีลักษณะโค้งเช่นโครงสร้างเปลือกบางได้ อย่างไรก็ตามในการเพิ่มพจน์ของความสมดุลนี้ ถึงแม้จะทำให้คำตอบมีความแม่นยำมากขึ้น แต่ก็ส่งผลให้มีการคำนวณมากขึ้นตามไปด้วยเช่นกัน

นอกจากการประมาณค่าความคลาดเคลื่อนแล้ว ยังมีขั้นตอนที่สำคัญอีกขั้นตอนหนึ่งในกระบวนการปรับขนาดชิ้นส่วนคือการสร้างชิ้นส่วนโดยอัตโนมัติ (mesh generation) กระบวนการนี้สามารถแบ่งออกเป็นประเภทใหญ่ ๆ ได้ 2 ประเภทคือ ประเภทแรกมุ่งที่จะลดการเตรียมข้อมูลของมนุษย์แต่เพียงอย่างเดียว ไม่ได้ต้องการเพื่อใช้ในกระบวนการปรับขนาดชิ้นส่วนแต่อย่างใด การสร้างชิ้นส่วนจึงนิยมใช้วิธีแปลงรูปโดเมนของปัญหาให้มาอยู่ในรูปเรขาคณิตอย่างง่าย⁽¹³⁾ จากนั้นสร้างชิ้นส่วนในรูปอย่างง่ายนี้ แล้วจึงแปลงกลับไปอยู่ในสภาพเริ่มต้น วิธีนี้มีข้อดีคือสามารถสร้างชิ้นส่วนที่มีรูปร่างสามเหลี่ยมหรือสี่เหลี่ยมก็ได้ และได้ชิ้นส่วนที่มีรูปร่างค่อนข้างดีมีขนาดต่อเนื่องกัน แต่มีข้อเสียคือใช้ได้เฉพาะกับโดเมนที่มีรูปร่างไม่ซับซ้อนมากนัก และไม่สามารถสร้างชิ้นส่วนให้มีขนาดแตกต่างออกไปในแต่ละบริเวณตามที่กำหนดได้ จึงไม่เหมาะสมอย่างยิ่งที่จะนำมาใช้ในกระบวนการปรับขนาดชิ้นส่วน

สำหรับประเภทที่ 2 เป็นการสร้างชิ้นส่วนโดยการเพิ่มชิ้นส่วนลงในโดเมนโดยตรง จึงสามารถที่จะกระจายขนาดของชิ้นส่วนให้แตกต่างกันออกไปในแต่ละพื้นที่ได้ ฉะนั้นเราจึงสนใจเฉพาะวิธีในประเภทที่ 2 นี้เท่านั้นดังจะได้กล่าวถึงต่อไป

ต้นไม้แปดภาค (Octree)⁽¹⁴⁾ หรือ ต้นไม้สี่ภาค (Quadtree) เป็นการสร้างชิ้นส่วนโดยการแบ่งโดเมนของปัญหาออกเป็นภาค 4 ภาคหรือ 8 ภาคตามแต่มิติของปัญหา และในแต่ละภาคเหล่านั้นก็จะทำการแบ่งเป็นภาคย่อยต่อไปเป็นลำดับ จนกระทั่งมีขนาดเท่ากับขนาดของชิ้นส่วนที่เราต้องการ จากนั้นจะเชื่อมชิ้นส่วนเหล่านั้นให้มีความต่อเนื่องกันและสอดคล้องกับขอบเขตของปัญหา วิธีนี้ค่อนข้างยุ่งยากในการสร้างความต่อเนื่องระหว่างชิ้นส่วน และได้ชิ้นส่วนที่มีรูปร่างไม่ค่อยดีนักจึงไม่เป็นที่นิยมใช้

วิธีต่อมาคือ วิธีการสร้างฟรอนท์คืบหน้า (Advancing Front technique)^(15, 16, 17) เป็นการสร้างชิ้นส่วนโดยการเชื่อมขอบเขตของปัญหาทั้งหมดด้วยเส้นเส้นหนึ่งเรียกว่าฟรอนท์ (Front) จากนั้นสร้างชิ้นส่วนจากขอบเขตเข้ามาพร้อมทั้งปรับขนาดของฟรอนท์ใหม่โดยการกันชิ้นส่วนที่เพิ่งสร้างให้ออกไปนอกฟรอนท์ ทำดังนี้ซ้ำไปจนกระทั่งหมดก็จะได้ชิ้นส่วนเต็มทั้งพื้นที่ การกำหนดขนาดของชิ้นส่วนในวิธีนี้ทำในขณะที่กำลังสร้างชิ้นส่วนนั้น ๆ โดยการตรวจสอบจากขนาดที่ได้กำหนดไว้ก่อนในบริเวณนั้น วิธีนี้มีข้อดีคือ เนื่องจากชิ้นส่วนเริ่มจากขอบเขตและมีการสร้างต่อไปทีละชิ้นส่วน จึงมั่นใจได้ว่าชิ้นส่วนจะไม่ออกนอกขอบเขตหรือซ้อนทับกัน นอกจากนี้ยังสามารถพัฒนาต่อไปเพื่อสร้างชิ้นส่วนชนิดสี่เหลี่ยมได้ โดยการกำหนดจำนวนชิ้นส่วนที่ขอบเขต และรูปร่างทางเรขาคณิตของชิ้นส่วนอย่างเคร่งครัด

การสร้างสามเหลี่ยมเดลอาน (Delaunay triangulation)^(18, 19, 20, 21) เป็นวิธีเก่าแก่ที่มีจุดประสงค์เพื่อเชื่อมต่อดจุดต่าง ๆ เข้าด้วยกันเป็นรูปสามเหลี่ยมจำนวนมากที่ไม่ซ้อนทับกันเท่านั้น จึงไม่สามารถรับประกันได้ว่าชิ้นส่วนที่สร้างขึ้นจะไม่ออกนอกขอบเขตของปัญหา ในระยะหลังได้มีผู้พัฒนาวิธีการต่าง ๆ ขึ้นมากมายจนกระทั่งสามารถกำหนดได้ว่าให้สร้างชิ้นส่วนขึ้นเฉพาะภายในขอบเขตเท่านั้น การกำหนดขนาดของชิ้นส่วนขึ้นอยู่กับจุดที่สร้างขึ้น โดยกำหนดระยะห่างของจุดตามความต้องการในแต่ละพื้นที่นั้น วิธีนี้มีข้อดีคือตำแหน่ง ขนาด และรูปร่างของชิ้นส่วนขึ้นอยู่กับวิธีการสร้างจุดที่แยกออกมาต่างหาก ในระยะหลังสามารถพัฒนาให้สร้างชิ้นส่วนสามเหลี่ยมที่มีรูปร่างดี

มากตลอดทั้งโดเมน มีการกระจายขนาดของชิ้นส่วนตามที่ต้องการ และใช้เวลาในการสร้างน้อย ข้อเสียของวิธีนี้คือไม่สามารถสร้างชิ้นส่วนที่เหลื่อมได้โดยตรง ต้องใช้การแปลงรูปจากชิ้นส่วนสามเหลี่ยม^[22]

1.3 วัตถุประสงค์ของงานวิจัย

1. ศึกษาการประมาณค่าความคลาดเคลื่อนของผลเฉลยที่ได้จากระเบียบวิธีไฟไนต์เอลิเมนต์
2. ศึกษาการสร้างชิ้นส่วนโดยอัตโนมัติสำหรับวิเคราะห์ด้วยระเบียบวิธีไฟไนต์เอลิเมนต์
3. พัฒนาโปรแกรมไฟไนต์เอลิเมนต์ที่ปรับขนาดชิ้นส่วน โดยอาศัยกระบวนการทั้ง 2 ข้างต้น

1.4 ขอบเขตการศึกษา

1. ศึกษาการวิเคราะห์โครงสร้างบนระนาบ (plane stress , plane strain)
2. ใช้ชิ้นส่วนที่มีรูปร่างเป็นรูปสามเหลี่ยม
3. วิเคราะห์ปัญหาสถิตศาสตร์ (static)
4. โครงสร้างรับเฉพาะแรงทางกล (mechanical forces)
5. วัสดุอยู่ในช่วงยืดหยุ่น
6. วัสดุมีคุณสมบัติเหมือนกันตลอดทั้งชิ้นส่วน
7. วัสดุมีคุณสมบัติเหมือนกันทุกทิศทาง

1.5 ผลที่คาดว่าจะได้รับ

1. เข้าใจกระบวนการของการประมาณค่าความคลาดเคลื่อนของผลเฉลยจากระเบียบวิธีไฟไนต์เอลิเมนต์
2. เข้าใจหลักการสร้างชิ้นส่วนโดยการใช้โปรแกรมคอมพิวเตอร์
3. เข้าใจขั้นตอนการทำงานของกระบวนการปรับขนาดชิ้นส่วน
4. ได้โปรแกรมคอมพิวเตอร์สำหรับใช้งานและศึกษาพัฒนาต่อไป

บทที่ 2

แนวทางและทฤษฎีที่ใช้ในการวิจัย

2.1 ความนำ

ในการปรับขนาดของชิ้นส่วนให้มีความสอดคล้องกับปัญหาใด ๆ นั้นมีขั้นตอนใหญ่ ๆ ตามลำดับดังต่อไปนี้ หลังจากได้ผลเฉลยจากระเบียบวิธีไฟไนต์เอลิเมนต์ในระบบชิ้นส่วนที่มีลักษณะหยาบซึ่งเป็นระบบเริ่มต้นแล้ว จึงประมาณค่าคลาดเคลื่อนในตำแหน่งต่าง ๆ ของปัญหาจากผลเฉลยที่ได้ จากนั้นคำนวณหาขนาดของชิ้นส่วนที่เหมาะสมกับขนาดความคลาดเคลื่อนพร้อมทั้งสร้างชิ้นส่วนขึ้นใหม่ตามขนาดที่คำนวณได้ แล้วจึงสร้างสถิติเฟนสเมตริกซ์เพื่อคำนวณหาผลเฉลยต่อไป กระทำซ้ำตามขั้นตอนที่กล่าวมาจนกระทั่งค่าประมาณความคลาดเคลื่อนของผลเฉลยมีขนาดอยู่ในช่วงที่ยอมรับได้ จึงใช้ผลเฉลยสุดท้ายนั้นเป็นผลเฉลยของระเบียบวิธีไฟไนต์เอลิเมนต์เพื่อใช้ประโยชน์ต่อไป จากนั้นเราจะได้กล่าวถึงแต่ละขั้นตอนโดยละเอียด

2.2 การประมาณค่าคลาดเคลื่อน

โดยปกติการคำนวณความคลาดเคลื่อนของวิธีการใด ๆ ก็ตามจะทำได้โดยการเทียบคำตอบที่ได้จากวิธีการนั้น ๆ กับผลเฉลยแม่นยำตรง เช่น

$$e = u - u_h \tag{2.2.1a}$$

$$e_\sigma = \sigma - \sigma_h \tag{2.2.1b}$$

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

- โดยที่ e : ค่าความคลาดเคลื่อนของการกระจัด
 u : ค่าการกระจัดแม่นยำตรง
 u_h : ค่าการกระจัดที่ได้จากการวิเคราะห์
 e_σ : ค่าความคลาดเคลื่อนของหน่วยแรง
 σ : ค่าหน่วยแรงแม่นยำตรง
 σ_h : ค่าหน่วยแรงที่ได้จากการวิเคราะห์

แต่ในกระบวนการปรับขนาดชิ้นส่วนนี้เราไม่ทราบผลเฉลยแม่นยำตรงของปัญหา จึงไม่สามารถหาความคลาดเคลื่อนตามสมการ (2.2.1) ได้ ต้องประมาณจากการเปรียบเทียบกับผลเฉลยอื่นที่มีความถูกต้องกว่าผล

เฉลี่ยจากระเบียบวิธีไฟไนต์เอลิเมนต์ดังแสดงในสมการ (2.2.2) สำหรับการคำนวณหาค่าเฉลี่ยที่ดีขึ้นนี้เราจะได้กล่าวถึงโดยละเอียดในหัวข้อถัดไป

$$e' = u' - u_h \quad (2.2.2a)$$

$$e'_\sigma = \sigma' - \sigma_h \quad (2.2.2b)$$

- โดยที่
- e' : ค่าประมาณความคลาดเคลื่อนของการกระจัด
 - u' : ค่าการกระจัดที่มีความถูกต้องกว่าผลจากระเบียบวิธีไฟไนต์เอลิเมนต์
 - u_h : ค่าการกระจัดจากระเบียบวิธีไฟไนต์เอลิเมนต์
 - e'_σ : ค่าประมาณความคลาดเคลื่อนของหน่วยแรง
 - σ' : ค่าหน่วยแรงที่มีความถูกต้องกว่าผลจากระเบียบวิธีไฟไนต์เอลิเมนต์
 - σ_h : ค่าหน่วยแรงจากระเบียบวิธีไฟไนต์เอลิเมนต์

จากสมการ (2.2.1) และ (2.2.2) จะเห็นว่าเราวัดความคลาดเคลื่อนเป็นสเกลาร์ในขณะที่คำตอบที่ได้จากระเบียบวิธีไฟไนต์เอลิเมนต์มีลักษณะเป็นเวกเตอร์ ซึ่งมีความไม่สะดวกในการนำมาใช้งาน ดังนั้นจึงต้องมีการนำรูปแบบอื่นที่มีลักษณะเป็นสเกลาร์มาใช้แทนเวกเตอร์ของความคลาดเคลื่อน เช่น ในรูปของพลังงานของความคลาดเคลื่อนดังในสมการ (2.2.3)

$$\|e\| = \left(\int_{\Omega} e_\sigma^T D^{-1} e_\sigma d\Omega \right)^{\frac{1}{2}} \quad (2.2.3)$$

- โดยที่
- $\|e\|$: นอร์มพลังงานของความคลาดเคลื่อน
 - e_σ : เวกเตอร์ความคลาดเคลื่อนของหน่วยแรง
 - D : เมตริกซ์ความสัมพันธ์ระหว่างความเค้นและความเครียด $\sigma = D\varepsilon$
 - Ω : โดเมนที่พิจารณา เช่น เฉพาะที่(Local) หรือ โดยรวม(Global)

นำความคลาดเคลื่อนที่ได้ในสมการ (2.2.3) มาเปรียบเทียบกับนอร์มพลังงานความเครียด(Strain energy norm) ในสมการ (2.2.4) จะได้ความคลาดเคลื่อนสัมพัทธ์(Relative error)ดังในสมการ (2.2.5)

$$\|U\| = \left(\int_{\Omega} \sigma^T D^{-1} \sigma d\Omega \right)^{\frac{1}{2}} \quad (2.2.4)$$

$$\eta = \frac{\|e\|}{\|U\|} = \frac{\left(\int_{\Omega} e_\sigma^T D^{-1} e_\sigma d\Omega \right)^{\frac{1}{2}}}{\left(\int_{\Omega} \sigma^T D^{-1} \sigma d\Omega \right)^{\frac{1}{2}}} \quad (2.2.5)$$

โดยที่ $\|U\|$: นอร์มพลังงานความเครียด
 η : ความคลาดเคลื่อนสัมพัทธ์

แทนค่าหน่วยแรงและความคลาดเคลื่อนในสมการ (2.2.5) ด้วยค่าประมาณจากสมการ (2.2.2) จะได้ค่าประมาณความคลาดเคลื่อนสัมพัทธ์และนอร์มพลังงานของค่าต่าง ๆ ดังนี้

$$\|e^*\| = \left(\int_{\Omega} e_{\sigma}^{*T} D^{-1} e_{\sigma}^* d\Omega \right)^{\frac{1}{2}} \quad (2.2.6a)$$

$$\|U^*\| = \left(\int_{\Omega} \sigma^{*T} D^{-1} \sigma^* d\Omega \right)^{\frac{1}{2}} \quad (2.2.6b)$$

$$\|U\| \approx \sqrt{\|U^*\|^2 + \|e^*\|^2} \quad (2.2.6c)$$

$$\eta \approx \eta^* = \frac{\|e^*\|}{\sqrt{\|U^*\|^2 + \|e^*\|^2}} \quad (2.2.6d)$$

นอกจากการแสดงผลในรูปของนอร์มพลังงานแล้วอาจใช้นอร์มในรูปอื่นแทนได้ เช่น ในงานวิจัยนี้เลือกใช้ แอลทูนอร์ม (L_2 norm) ซึ่งมีรูปแบบในลักษณะเดียวกับนอร์มพลังงาน แต่ไม่มีพจน์ของเมตริกซ์ความสัมพันธ (D) ทำให้ปริมาณการคำนวณลดลง ค่านอร์มต่าง ๆ ในสมการ (2.2.3) และ (2.2.6) สามารถเขียนในรูปแอลทูนอร์มได้ ดังในสมการ (2.2.7) และ (2.2.8)

$$\|e\|_2 = \left(\int_{\Omega} e_{\sigma}^T e_{\sigma} d\Omega \right)^{\frac{1}{2}} \quad (2.2.7a)$$

$$\|U\|_2 = \left(\int_{\Omega} \sigma^T \sigma d\Omega \right)^{\frac{1}{2}} \quad (2.2.7b)$$

$$\eta = \frac{\|e\|_2}{\|U\|_2} \quad (2.2.7c)$$

$$\|e^*\|_2 = \left(\int_{\Omega} e_{\sigma}^{*T} e_{\sigma}^* d\Omega \right)^{\frac{1}{2}} \quad (2.2.8a)$$

$$\|U^*\|_2 = \left(\int_{\Omega} \sigma^{*T} \sigma^* d\Omega \right)^{\frac{1}{2}} \quad (2.2.8b)$$

$$\|U\|_2 \approx \sqrt{\|U^*\|_2^2 + \|e^*\|_2^2} \quad (2.2.8c)$$

$$\eta \approx \eta^* = \frac{\|e^*\|_2}{\sqrt{\|U^*\|_2^2 + \|e^*\|_2^2}} \quad (2.2.8d)$$

เพื่อเป็นการลดการคำนวณลง งานวิจัยนี้เลือกที่จะใช้ความคลาดเคลื่อนสัมพัทธ์โดยประมาณ ดังในสมการ (2.2.9) แทนสมการ (2.2.8d)

$$\eta^* = \frac{\|e^*\|_2}{\|U^*\|_2} \quad (2.2.9)$$

เมื่อได้ค่าประมาณของความคลาดเคลื่อนสัมพัทธ์โดยรวมแล้วนั้นไปเปรียบเทียบกับความคลาดเคลื่อนสัมพัทธ์ที่ยอมรับได้ ถ้ามีความคลาดเคลื่อนมากกว่าก็คำนวณหาขนาดของชิ้นส่วนที่สอดคล้องกับความคลาดเคลื่อนเฉพาะทีละส่วนต่อไป

2.3 การประมาณค่าผลเฉลยจากผลของระเบียบวิธีไฟไนต์เอลิเมนต์

ดังที่กล่าวไว้ในหัวข้อ 2.2 ว่าการประมาณค่าความคลาดเคลื่อน จะประมาณจากผลเฉลยของระเบียบวิธีไฟไนต์เอลิเมนต์เทียบกับผลเฉลยที่แม่นยำกว่า ซึ่งในหัวข้อนี้จะได้กล่าวถึงรายละเอียดในการประมาณค่าหน่วยแรงให้มีความแม่นยำขึ้น โดยวิธีการฟื้นตัวเป็นหย่อม(Patch Recovery)^[5, 6, 7, 8] เพื่อใช้เปรียบเทียบหาความคลาดเคลื่อน

โดยธรรมชาติของระเบียบวิธีไฟไนต์เอลิเมนต์ หน่วยแรงที่ได้จะมีความคลาดเคลื่อนแตกต่างกันไปในแต่ละส่วนของชิ้นส่วน โดยที่ตำแหน่งเกาส์จะมีค่าหน่วยแรงแม่นยำกว่าในตำแหน่งอื่น ๆ ^[23] ดังนั้นเราจึงสามารถประมาณค่าหน่วยแรงที่จุดต่อได้แม่นยำขึ้นโดยประมาณจากหน่วยแรงที่ตำแหน่งเกาส์รอบจุดต่อนั้น ดังแสดงในรูป 2.3.1 สำหรับวิธีที่ใช้ในการประมาณค่านั้นทำได้หลายวิธี แต่ที่ค่อนข้างยืดหยุ่นและได้ค่าที่แม่นยำคือวิธีกำลังสองน้อยสุดดังมีรายละเอียดต่อไปนี้

สมมติให้หน่วยแรงโดยประมาณ(σ^*)เป็นหน่วยแรงที่ได้จากการประมาณค่าจากตำแหน่งเกาส์ซึ่งอยู่ในรูปของพหุนาม ในกรณี 1 มิติจะมีลักษณะเป็นเส้นดังแสดงในรูป 2.3.2 และลักษณะของเส้นจะขึ้นอยู่กับการกำลังและค่าคงตัวของพหุนาม สำหรับกรณี 2 มิติหน่วยแรงโดยประมาณมีลักษณะเป็นพื้นผิวมีค่าดังในสมการ (2.3.1a) หรือในรูปเมตริกซ์ดังสมการ (2.3.1b)

$$\sigma^* = a_0 + a_1x + a_2y + a_3xy + a_4x^2 + \dots \quad (2.3.1a)$$

$$\sigma^* = [p]\{a\} \quad (2.3.1b)$$

โดยที่ $[p]$: ฟังก์ชันฐาน(Base Function) อยู่ในรูปตัวแปรของพหุนาม $[1 \ x \ y \ xy \ x^2 \dots]$

$\{a\}$: ค่าคงตัวของพหุนามซึ่งเป็นตัวไม่รู้ค่าที่ต้องคำนวณหา $\{a_0 \ a_1 \ a_2 \ a_3 \ a_4 \dots\}$

หลักของวิธีกำลังสองน้อยสุดคือต้องหาค่าคงตัว $\{a\}$ ให้หน่วยแรงโดยประมาณมีความแตกต่างจากจุดที่ประมาณมาน้อยที่สุด ในกรณี 2 มิติเศษตกค้างของกำลังสองของความแตกต่างระหว่างหน่วยแรงโดยประมาณและหน่วยแรงที่ตำแหน่งเกาส์สามารถแสดงได้ดังในสมการ (2.3.2)

$$R(a) = \sum_{i=1}^n (\sigma_h(x_i, y_i) - \sigma^*(x_i, y_i))^2 \quad (2.3.2a)$$

$$R(a) = \sum_{i=1}^n (\sigma_h(x_i, y_i) - [p(x_i, y_i)]\{a\})^2 \quad (2.3.2b)$$

โดยที่ $R(a)$: เศษตกค้างของกำลังสองของความแตกต่างระหว่างหน่วยแรง

σ_h : หน่วยแรงที่ตำแหน่งเกาส์

σ^* : หน่วยแรงโดยประมาณ

n : จำนวนตำแหน่งเกาส์ทั้งหมดในหย่อม

ในการหาค่าคงตัวเพื่อให้ได้เศษตกค้างต่ำสุด ทำได้โดยกำหนดค่าอนุพันธ์ของเศษตกค้างให้เท่ากับศูนย์ โดยหาอนุพันธ์เทียบกับค่าคงตัวแต่ละตัว จะได้สมการซึ่งมีจำนวนเท่ากับจำนวนของค่าคงตัวดังนี้

$$\sum_{i=1}^n [p(x_i, y_i)]^T [p(x_i, y_i)]\{a\} = \sum_{i=1}^n \sigma_h(x_i, y_i) [p(x_i, y_i)]^T \quad (2.3.3)$$

ในงานวิจัยนี้ใช้ฟังก์ชันฐานเป็นพหุนามกำลัง 1 โดยเลือกใช้ $p = [1 \ x \ y \ xy]$ ดังนั้นสามารถแจงสมการ (2.3.3) ให้อยู่ในรูปของ เมตริกซ์ได้ดังนี้

$$\begin{bmatrix} n & \sum x_i & \sum y_i & \sum x_i y_i \\ \sum x_i & \sum x_i^2 & \sum x_i y_i & \sum x_i^2 y_i \\ \sum y_i & \sum x_i y_i & \sum y_i^2 & \sum x_i y_i^2 \\ \sum x_i y_i & \sum x_i^2 y_i & \sum x_i y_i^2 & \sum x_i^2 y_i^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} \sum \sigma_{hi} \\ \sum x_i \sigma_{hi} \\ \sum y_i \sigma_{hi} \\ \sum x_i y_i \sigma_{hi} \end{bmatrix} \quad (2.3.4a)$$

$$[A]\{a\} = \{b\} \quad (2.3.4b)$$

$$\{a\} = [A]^{-1}\{b\} \quad (2.3.5)$$

$$\sigma^*(x_i, y_i) = [p(x_i, y_i)]\{a\} \quad (2.3.6)$$

หลังจากคำนวณหาค่าคงตัวจากสมการ (2.3.5) ได้แล้ว สามารถคำนวณหน่วยแรงที่จุดต่อได้โดยการแทนพิกัดที่จุดต่อลงในสมการ (2.3.6) และเนื่องจากในปัญหามatrix มีหน่วยแรง 3 ค่าดังนั้นในแต่ละจุดต่อจึงต้องคำนวณหาค่าคงตัว 3 ครั้งโดยในแต่ละครั้งจะเปลี่ยนแปลงเฉพาะค่าทางด้านขวาของสมการ (2.3.4) เท่านั้น ถ้าแก้สมการด้วยการแก้สมการโดยตรง (Direct Solver) การแก้สมการในครั้งที่ 2 และ 3 ก็เพียงแต่แทนค่ากลับเท่านั้นไม่ต้องแก้สมการใหม่

ข้อควรระวังสำหรับวิธีนี้ก็คือ จำนวนจุดเกาส์ในแต่ละหอย่อมของจุดต่อจะต้องมากกว่าจำนวนตัวแปรในสมการ (2.3.4) ในที่นี้ต้องมีมากกว่า 4 จุดมิฉะนั้นเมทริกซ์สัมประสิทธิ์ ([A]) ในสมการ (2.3.4) จะมีลักษณะเป็นเอกฐาน (Singular) ลักษณะเช่นนี้จะเกิดขึ้นในบริเวณขอบของปัญหาดังจุดต่อหมายเลข 1,3,4,5,6 ในรูป 2.3.3 ซึ่งมีวิธีแก้ปัญหาได้ 2 วิธีคือ

- 1.) ลดจำนวนพจน์ของพหุนามลง เช่น ให้ $[p] = [1 \ x \ y]$ หรือ $[p] = [1]$
- 2.) ประมาณค่าหน่วยแรงของจุดต่อนั้นจากหอย่อมของจุดต่ออื่นดังเช่นในรูป 2.3.4

เนื่องจากหน่วยแรงที่ได้จากระเบียบวิธีไฟไนต์เอลิเมนต์ในชิ้นส่วนที่อยู่บริเวณขอบมีความคลาดเคลื่อนสูงเมื่อเทียบกับชิ้นส่วนด้านใน งานวิจัยนี้จึงเลือกใช้วิธีที่ 2 ในการประมาณค่าหน่วยแรงที่จุดต่อบริเวณขอบ ยกเว้นในกรณีพิเศษที่มีลักษณะดังแสดงในรูป 2.3.5 ซึ่งการหาหอย่อมที่จะใช้ประมาณค่าทำได้ลำบาก จึงใช้วิธีที่ 1 แทนโดยให้จำนวนของพหุนามเท่ากับ 1

หลังจากที่ได้คำนวณหน่วยแรงที่จุดต่อทุกจุดแล้ว ประมาณค่าหน่วยแรงที่ตำแหน่งใด ๆ ในชิ้นส่วนด้วยฟังก์ชันสัณฐาน (Shape function) ตัวเดียวกับที่ใช้ในการหาค่าการกระจัดในระเบียบวิธีไฟไนต์เอลิเมนต์ดังในสมการ (2.3.7)

$$\{\sigma^*(x, y)\} = [N(x, y)]\{\sigma_n^*\} \quad (2.3.7)$$

โดยที่ σ_n^* : หน่วยแรงที่ตำแหน่งจุดต่อ
 σ^* : หน่วยแรงที่ตำแหน่งใด ๆ
 N : ฟังก์ชันสัณฐานในระเบียบวิธีไฟไนต์เอลิเมนต์

จากนั้นคำนวณแอมพลิจูดของความเค้นและความคลาดเคลื่อน ทั้งในส่วนเฉพาะที่ในสมการ (2.3.8) และโดยรวมทั้งปัญหาในสมการ (2.3.9)

$$\|U^*\|_{2l} = \left(\int_{\Omega} (N\sigma^*)^T (N\sigma^*) d\Omega \right)^{\frac{1}{2}} \quad (2.3.8a)$$

$$\|e^*\|_{2l} = \left(\int_{\Omega} (N\sigma^* - \sigma_n^*)^T (N\sigma^* - \sigma_n^*) d\Omega \right)^{\frac{1}{2}} \quad (2.3.8b)$$

$$\|e^*\|_{2g} = \left(\sum_{i=1}^{Nc} \|e^*\|_{2li}^2 \right)^{\frac{1}{2}} \quad (2.3.9a)$$

$$\|U^*\|_{2g} = \left(\sum_{i=1}^{Ne} \|U^*\|_{2li}^2 \right)^{\frac{1}{2}} \quad (2.3.9b)$$

$$\eta_g^* = \frac{\|e^*\|_{2g}}{\|U^*\|_{2g}} \quad (2.3.9c)$$

โดยที่ $\|U^*\|_{2l}$: แอลทูนอร์มของความเค้นเฉพาะที่ในชั้นส่วน
 $\|e^*\|_{2l}$: แอลทูนอร์มของความคลาดเคลื่อนเฉพาะที่ในชั้นส่วน
 $\|U^*\|_{2g}$: แอลทูนอร์มของความเค้นรวม
 $\|e^*\|_{2g}$: แอลทูนอร์มของความคลาดเคลื่อนรวม
 N : ฟังก์ชันสัญญาณ
 Ne : จำนวนชั้นส่วนทั้งหมดในระบบ

เนื่องจากในงานวิจัยนี้ใช้ชั้นส่วนที่มีฟังก์ชันฐานมีระดับชั้นของพหุนามเท่ากับ 1 ซึ่งมีจำนวนจุดต่อน้อยกว่าจำนวนชั้นส่วนมาก ดังนั้นจะเป็นการประหยัดหน่วยความจำและเวลาในการคำนวณลง ถ้าคำนวณนอร์มเฉพาะที่ต่าง ๆ ที่จุดต่อแทนการคำนวณในแต่ละชั้นส่วน และคำนวณนอร์มโดยรวมจากการรวมจากทุกจุดต่อดังในสมการ (2.3.10) และ (2.3.11)

$$\|U^*\|_{2l} = \left((\sigma^{*T} \sigma^*) \cdot \frac{1}{3} \cdot \sum_{i=1}^n area_i \right)^{\frac{1}{2}} \cdot \frac{5}{12} \quad (2.3.10a)$$

$$\|e^*\|_{2l} = \left(\frac{1}{3} \cdot \sum_{i=1}^n ((\sigma^* - \sigma_{hi})^T (\sigma^* - \sigma_{hi})) \cdot area_i \right)^{\frac{1}{2}} \cdot \frac{5}{12} \quad (2.3.10b)$$

$$\|U^*\|_{2g} = \left(\sum_{i=1}^{Nn} \|U^*\|_{2li}^2 \right)^{\frac{1}{2}} \quad (2.3.11a)$$

$$\|e^*\|_{2g} = \left(\sum_{i=1}^{Nn} \|e^*\|_{2li}^2 \right)^{\frac{1}{2}} \quad (2.3.11b)$$

$$\eta_g^* = \frac{\|e^*\|_{2g}}{\|U^*\|_{2g}} \quad (2.3.11c)$$

โดยที่ $\|U^*\|_{2l}$: แอลทูนอร์มของความเค้นเฉพาะที่ที่จุดต่อ
 $\|e^*\|_{2l}$: แอลทูนอร์มของความคลาดเคลื่อนเฉพาะที่ที่จุดต่อ
 $\|U^*\|_{2g}$: แอลทูนอร์มของความเค้นรวม

$\|e^*\|_{2g}$: แอลทูนอร์มของความคลาดเคลื่อนรวม

$area$: พื้นที่ของแต่ละชั้นส่วนในหอย่อม

n : จำนวนชั้นส่วนในหอย่อม

Nh : จำนวนจุดต่อทั้งหมดในระบบ

จากความคลาดเคลื่อนที่ได้นี้นำมาคำนวณหาขนาดที่เหมาะสมของชั้นส่วนดังจะได้กล่าวถึงรายละเอียดในหัวข้อถัดไป

2.4 การประมาณขนาดของชั้นส่วน

เนื่องจากความคลาดเคลื่อนแปรตามขนาดของชั้นส่วนยกกำลังด้วยกำลังของพหุนามในฟังก์ชันฐานดังในสมการ (2.4.1) ดังนั้นถ้าเรารู้ขนาดของชั้นส่วนและความคลาดเคลื่อนในปัจจุบัน เราสามารถคำนวณขนาดของชั้นส่วนที่มีความคลาดเคลื่อนเท่ากับความคลาดเคลื่อนที่ยอมให้ได้ดังในสมการ (2.4.2)

$$\|e\| = Ch^p \quad (2.4.1)$$

โดยที่ C : ค่าคงตัวที่ไม่รู้ค่า
 h : ขนาดของชั้นส่วน
 p : กำลังของพหุนามในฟังก์ชันฐาน

$$h_{new} = h_{exist} \cdot \left(\frac{\|e\|_a}{\|e\|_{exist}} \right)^{\frac{1}{p}} \quad (2.4.2a)$$

ในกรณี $p = 1$
$$h_{new} = h_{exist} \cdot \frac{\|e\|_a}{\|e\|_{exist}} \quad (2.4.2b)$$

โดยที่ h_{new} : ขนาดของชั้นส่วนที่จะสร้างขึ้นใหม่

h_{exist} : ขนาดของชั้นส่วนในปัจจุบัน

$\|e\|_a$: ความคลาดเคลื่อนที่ยอมให้

$\|e\|_{exist}$: ความคลาดเคลื่อนในปัจจุบัน

สำหรับการหาความคลาดเคลื่อนที่ยอมให้ นั้น จะคำนวณจากพื้นฐานที่ว่าในการเปรียบเทียบระบบที่มีชั้นส่วนจำนวนเท่ากันแล้ว ระบบที่มีการกระจายความคลาดเคลื่อนออกไปเท่า ๆ กันทุกจุดจะให้คำตอบที่ดีที่สุด^[29] ดังนั้นเราสามารถคำนวณค่าความคลาดเคลื่อนเฉพาะที่ที่ยอมให้ในแต่ละจุดต่อ โดยให้มีขนาดเท่า ๆ กันดังแสดงในสมการ (2.4.3)

$$\|e\|_{no} = \eta_o \frac{\|U^*\|_{2g}}{\sqrt{Nn}} \quad (2.4.3)$$

โดยที่ $\|e\|_{no}$: ความคลาดเคลื่อนที่ยอมให้ในแต่ละจุดต่อ
 Nn : จำนวนจุดต่อทั้งหมดในระบบ
 η_o : ความคลาดเคลื่อนสัมพัทธ์โดยรวมที่ยอมให้

ในกรณีที่คำนวณความคลาดเคลื่อนเฉพาะที่โดยอ้างอิงจากชิ้นส่วน ความคลาดเคลื่อนที่ยอมให้ในแต่ละชิ้นส่วนจะคำนวณได้จากสมการ (2.4.4)

$$\|e\|_{ea} = \eta_o \frac{\|U^*\|_{2g}}{\sqrt{Ne}} \quad (2.4.4)$$

โดยที่ $\|e\|_{ea}$: ความคลาดเคลื่อนที่ยอมให้ในแต่ละชิ้นส่วน
 Ne : จำนวนชิ้นส่วนทั้งหมดในระบบ

2.5 การสร้างชิ้นส่วนโดยอัตโนมัติ

ในงานวิจัยนี้เลือกวิธีการสร้างสามเหลี่ยมเดอลอนตามแนวทางของ Karamete , Tokdemir , Ger^[25] และใช้วิธีการสร้างจุดตามแนวทางของ Weatherill , Hassan^[26] ดังรายละเอียดต่อไปนี้

การสร้างสามเหลี่ยมเดอลอนคือการเชื่อมต่อจุดทุกจุดที่มีอยู่ ให้เกิดเป็นโครงข่ายตาข่ายของสามเหลี่ยมที่ไม่มีารซ้อนทับกันและไม่มีจุดใดที่ไม่เป็นจุดยอดของสามเหลี่ยม โดยมีพื้นฐานดังนี้คือจุดทุกจุดจะมีพื้นที่ที่อยู่ใกล้ตัวมากที่สุด เรียกเส้นที่แบ่งพื้นที่ของจุดใด ๆ กับพื้นที่ของจุดข้างเคียงว่าโครงข่ายโวโรนอย(Voronoi Tesselation , Voronoi Tile) ดังแสดงในรูป 2.5.1 และถ้าลากเส้นจากจุดทุกจุดไปตั้งฉากยังเส้นเหล่านี้ที่ล้อมรอบจุดอยู่ ก็จะได้โครงข่ายสามเหลี่ยมเดอลอน จะเห็นว่าการสร้างโครงข่ายโวโรนอยโดยตรงด้วยคอมพิวเตอร์ทำได้ยาก ดังนั้นการสร้างสามเหลี่ยมเดอลอนจะใช้คุณสมบัติอื่น ๆ ของโครงข่ายแทน เช่น

คุณสมบัติมุมเล็กสุดที่ใหญ่ที่สุด(Max-Min Angle property) : มีหลักว่าสี่เหลี่ยมใด ๆ ที่มีมุมภายในทุกมุมเล็กกว่า π เส้นทแยงมุมที่แบ่งสี่เหลี่ยมนี้เป็นสามเหลี่ยมเดอลอน 2 รูปคือเส้นทแยงมุมที่ทำให้มุมที่เล็กที่สุดใน 6 มุมมีขนาดใหญ่กว่าที่เกิดจากเส้นทแยงมุมอีกเส้นหนึ่ง ดังแสดงในรูป 2.5.2 คุณสมบัติข้อนี้นำไปใช้ในกระบวนการกลับเส้นทแยงมุมในการสร้างสามเหลี่ยมเดอลอนตามแนวทางของหลาย ๆ คน^[27, 28, 29]

คุณสมบัติวงกลมว่าง(Empty Circle) : จากรูป 2.5.1 จะมีจุดที่เกิดจากเส้นแบ่งพื้นที่ 3 เส้นมาบรรจบกัน ซึ่งถ้าใช้เป็นจุดศูนย์กลางของวงกลมจะได้วงกลมที่ผ่านจุดยอดมุมของสามเหลี่ยมเดอลอนและไม่มีจุดใด ๆ อยู่ในวงกลมนี้ คุณสมบัติข้อนี้เองที่นำมาใช้ในงานวิจัยโดยการตรวจสอบหาจุดที่สร้างวงกลมได้มีรัศมีเล็กที่สุดมาใช้เป็นจุดยอดของสามเหลี่ยม นอกจากคุณสมบัติ 2 ข้อนี้แล้วโครงข่ายไวโรนอยยังมีคุณสมบัติอื่น ๆ อีกมากถ้ามีความสนใจสามารถศึกษาได้จากงานของ Lawson⁽³⁰⁾

สำหรับการสร้างสามเหลี่ยมตามแนวทางของ Karamete และคณะนั้นเรียกว่า การสร้างสามเหลี่ยมเดอลอนโดยใช้ฟรอนท์ที่คืบหน้า (Advancing Front Delaunay Triangulation) มีรายละเอียดดังแสดงในรูป 2.5.3 โดยเริ่มจากการเชื่อมต่อจุดทุกจุดที่อยู่บนขอบของปัญหาไล่ตามกันไปทีละจุด เรียกเส้นที่เชื่อมจุดเหล่านี้ว่าฟรอนท์ ในรูปที่ 2.5.3(1) ฟรอนท์คือเส้นที่เชื่อมจุด 1 3 5 4 2 1 และเรียกเส้นที่เชื่อมจุด 2 จุดซึ่งจะใช้เป็นฐานของสามเหลี่ยมว่า เส้นกำเนิด (Generating line) เลือกเส้นกำเนิดเส้นหนึ่งจากฟรอนท์ในที่นี้เลือกเส้น $\overline{13}$ จากนั้นตรวจสอบจุดที่อยู่ในฟรอนท์ว่าจุดใดที่สอดคล้องกับคุณสมบัติของสามเหลี่ยมเดอลอน ก็เชื่อมต่อจุดนั้นเข้ากับเส้นกำเนิดเป็นสามเหลี่ยมในที่นี้คือจุดหมายเลข 6 ทำการปรับปรุงฟรอนท์ใหม่โดยการเพิ่มเส้น $\overline{16}$ และ $\overline{63}$ พร้อมทั้งลบเส้น $\overline{13}$ ออก ก็จะเป็นการกันสามเหลี่ยมที่สร้างขึ้นใหม่ออกจากฟรอนท์ เลือกเส้นกำเนิดเส้นใหม่และทำซ้ำเช่นนี้เรื่อยไปจนกระทั่งไม่มีเส้นกำเนิดเหลืออยู่ในฟรอนท์ก็จะเสร็จสิ้นการเชื่อมโยงจุดเป็นสามเหลี่ยมเดอลอน วิธีนี้มีข้อดีคือสามเหลี่ยมที่สร้างจะอยู่ใน ฟรอนท์เท่านั้น จึงสามารถใช้กับปัญหาที่มีช่องว่างภายในได้ ดังแสดงในรูป 2.5.4

จะเห็นว่าวิธีการที่ได้กล่าวมาเป็นการเชื่อมต่อจุดต่าง ๆ ที่มีอยู่แล้วเข้าด้วยกัน โดยไม่ได้เหยียดถึงการสร้างจุดต่าง ๆ เหล่านี้เลย ต่อไปนี้จะกล่าวถึงรายละเอียดในการสร้างจุดตามแนวทางของ Weatherill และคณะ ซึ่งใช้ควบคู่กับการสร้างสามเหลี่ยมเดอลอนแต่มีขั้นตอนแสดงในรูป 2.5.4 โดยจะกำหนดระยะห่างระหว่างจุดไว้ทุกจุด เมื่อมีการเพิ่มจุดใด ๆ จะต้องมีใกล้จุดที่มีอยู่แล้วเกินกว่าระยะที่กำหนดนี้ หลักในการสร้างจุดก็คือหลังจากเชื่อมจุดที่ขอบเข้าด้วยกันหมดแล้วดังรูป 2.5.4(1) ก็จะเพิ่มจุดลงที่กึ่งกลางของแต่ละสามเหลี่ยมโดยต้องไม่แย้งกับข้อกำหนดเรื่องระยะห่าง ทั้งกับจุดที่มีอยู่เดิมและจุดที่เพิ่มเข้าไปใหม่ด้วยกัน ดังในรูป 2.5.4(2) สำหรับระยะห่างที่จะกำหนดให้กับจุดใหม่ก็ได้จากการเฉลี่ยจากจุดยอดของสามเหลี่ยม เมื่อเพิ่มจุดเรียบร้อยแล้วสร้างสามเหลี่ยมขึ้นใหม่จากจุดเดิมที่มีอยู่แล้วร่วมกับจุดที่เพิ่มเข้าไป ทำซ้ำจนกระทั่งไม่สามารถเพิ่มจุดใด ๆ ได้อีกก็จะได้สามเหลี่ยมเต็มพื้นที่มีขนาดแปรตามระยะห่างที่ได้กำหนดไว้ ข้อดีของการสร้างจุดวิธีนี้คือสามารถกำหนดความถี่ห่างของจุดในแต่ละพื้นที่ได้ง่ายด้วยการกำหนดระยะห่างให้กับแต่ละจุด

จะเห็นว่าในการสร้างชิ้นส่วนนั้น มีขั้นตอนที่ต้องใช้มากคือการค้นหาตำแหน่งจุดที่จะใช้เป็นจุดยอดของสามเหลี่ยม ซึ่งเกี่ยวข้องโดยตรงกับการจัดการข้อมูล เราเลือกโครงสร้างข้อมูลตามแนวทางของ Peraire และคณะ⁽³¹⁾ เรียกโครงสร้างชนิดนี้ว่าต้นไม้เชิงตัวเลขสลับ ADT (Alternate Digital Tree) เป็นโครงสร้างชนิดแผนภูมิต้นไม้ (Tree) ชนิดหนึ่ง ซึ่งช่วยค้นหาข้อมูลที่จักเพาะลงไปในแต่ละพื้นที่ ดังจะได้กล่าวถึงรายละเอียดต่อไปในบทของการพัฒนาโปรแกรม หัวข้อโครงสร้างข้อมูล

2.6 การแก้ระบบสมการ

โดยปกติระเบียบวิธีไฟไนต์เอลิเมนต์ต้องมีการแก้สมการหลายตัวแปรอยู่แล้ว ซึ่งมีวิธีการแก้สมการอยู่ 2 วิธีใหญ่ ๆ คือการแก้สมการโดยตรง(Direct Solver) และ การทำซ้ำ(Iterative Solver) การแก้สมการโดยตรงมีข้อดีคือ ในการแก้ปัญหหลาย ๆ ปัญหาที่มีเมตริกซ์สัมประสิทธิ์ตัวเดียวกัน สามารถทำได้รวดเร็วเพราะแก้สมการเพียงครั้งเดียว และสามารถประยุกต์กับการแก้ปัญหาที่แบ่งเป็นโครงสร้างย่อยได้ (Substructure) นอกจากนี้ถ้าระบบสมการมีขนาดไม่ใหญ่เกินไปประกอบกับการเตรียมข้อมูลที่ดี สามารถจัดเรียงลำดับของชิ้นส่วนหรือจุดต่อให้เมตริกซ์ของสมการมีสมาชิกรวมอยู่ที่บริเวณแนวทแยงได้ ก็จะทำให้การแก้ระบบสมการมีประสิทธิภาพสูง แต่ในกรณีของกระบวนการปรับขนาดชิ้นส่วน จะมีการเพิ่มจำนวนชิ้นส่วนและจุดต่อโดยไม่คำนึงถึงลำดับของระบบสมการทำให้การแก้ระบบสมการมีประสิทธิภาพต่ำ การแก้ปัญหาทำได้โดยใช้โปรแกรมมาช่วยจัดเรียงลำดับของจุดต่อและชิ้นส่วน ซึ่งมีด้วยกันหลายวิธี อย่างไรก็ตามการจัดเรียงมีความยุ่งยากและต้องใช้เวลามาก นอกจากนี้ก็ยังคงมีข้อจำกัดอยู่ ดังในระบบสมการขนาดใหญ่ซึ่งถึงแม้ว่าจะมีการจัดเรียงข้อมูลอย่างดีแล้วก็ตาม การแก้สมการโดยตรงก็ยังคงไม่มีประสิทธิภาพอยู่ดี การแก้สมการด้วยวิธีกระทำซ้ำเป็นอีกทางเลือกหนึ่งที่มีความเหมาะสม ข้อเสียของวิธีการนี้คือในหลายวิธียังลู่เข้าสู่คำตอบค่อนข้างช้าทำให้เกิดการสะสมของความคลาดเคลื่อน และในกรณีที่เมตริกซ์มีลักษณะเป็นอิลคอนดิชัน (Ill-Conditioned) ผลที่ได้จะไม่ลู่เข้า ทำให้หาคำตอบไม่ได้ แต่ในกรณีทั่ว ๆ ไปเมื่อเทียบกับการแก้สมการโดยตรงแล้วก็ยังคงมีความเหมาะสมกว่า เพราะในกระบวนการปรับขนาดมีการแก้สมการของปัญหาเดียวกันหลายครั้ง แต่มีระบบสมการเปลี่ยนไปทุกครั้ง ซึ่งถ้าใช้การแก้สมการโดยตรงก็ต้องแก้สมการใหม่ทุกครั้ง ในขณะที่วิธีการทำซ้ำสามารถนำคำตอบที่ได้จากการแก้สมการครั้งก่อน ๆ มาใช้เป็นคำตอบเริ่มต้นในกระบวนการได้

ในงานวิจัยนี้ เลือกการแก้สมการด้วยวิธีทำซ้ำชนิด เกรเดียนต์สังยุคแบบมีเงื่อนไขล่วงหน้า PCG (Preconditioned Conjugate Gradient) ^[32 - 40] ซึ่งเป็นการพัฒนามาจากการหาค่าต่ำสุดของฟังก์ชัน(Minimization Problem) ในสมการ (2.6.1)

$$F(x) = \frac{1}{2} \langle Ax, x \rangle - \langle b, x \rangle \quad (2.6.1)$$

จะได้ $F'(x) = \text{grad } F(x) = Ax - b = 0$

โดยที่ $A \in R^{l \times l}$ และมีค่าบวกแน่นอน (Positive Definite) (PSD)

$$b \in R^l$$

\langle, \rangle : ผลคูณเชิงสเกลาร์ (Scalar product หรือ Dot product)

$F(x)$: ฟังก์ชันที่ต้องการหาค่าต่ำสุด

และถ้าให้ $f(\alpha) = F(x + \alpha p)$ (2.6.2)

โดยที่ p : เวกเตอร์ทิศทางในการปรับแก้เวกเตอร์ x หรือเรียกว่าเกรเดียนต์ (gradient)

α : เป็นค่าคงตัวเพื่อปรับขนาดเวกเตอร์ปรับแก้ทิศทาง p

เราสามารถแจกฟังก์ชันในสมการ (2.6.2) ได้ดังในสมการ (2.6.3) ซึ่งจะมีความสมมูลกับสมการ (2.6.1) โดยมีคำตอบเท่ากับ $x + \alpha p$ และเมื่อหาค่าต่ำสุดโดยอนุพันธ์ของฟังก์ชันในสมการ (2.6.2) เทียบกับ α จะได้ค่าคงตัวที่เหมาะสมที่สุดสำหรับเวกเตอร์ปรับแก้ทิศทางใด ๆ ดังในสมการ (2.6.4)

$$f(\alpha) = F(x) + \alpha \langle Ax - b, p \rangle + \alpha^2 \frac{1}{2} \langle Ap, p \rangle \quad (2.6.3)$$

$$\alpha = \frac{\langle r, p \rangle}{\langle Ap, p \rangle} \quad (2.6.4)$$

โดยที่ r : เชนตกค้างของสมการมีค่าเท่ากับ $b - Ax$

α : ค่าคงตัวที่เหมาะสมที่สุด(optimal) สำหรับเวกเตอร์ปรับแก้ทิศทาง p

ถ้าเลือกเวกเตอร์ปรับแก้ทิศทางให้เท่ากับเชนตกค้างของสมการ $p = r = b - Ax$ เราเรียกทิศทางปรับแก้ว่าการลงชันสุด (Steepest descent) ซึ่งสามารถหาค่า x ได้ดังในกระบวนการในสมการ (2.6.5)

เริ่มต้น :

$$x^0 = \text{arbitrary}, r^0 = b - Ax^0 \quad (2.6.5a)$$

ทำซ้ำ : $m = 0, 1, 2, \dots$

$$x^{m+1} = x^m + \alpha_m r^m \quad (2.6.5b)$$

$$r^{m+1} = b - Ax^{m+1} = r^m - \alpha_m Ar^m \quad (2.6.5c)$$

โดยที่
$$\alpha_m = \frac{\langle r^m, r^m \rangle}{\langle r^m, Ar^m \rangle} \quad (2.6.5d)$$

จะเห็นว่าเวกเตอร์ปรับแก้ทิศทางจะเปลี่ยนไปในแต่ละรอบของการกระทำ โดยไม่คำนึงถึงผลของการปรับแก้ครั้งก่อน ๆ ดังนั้นเพื่อเร่งการลู่เข้าโดยการรวมทิศทางที่ต้องปรับแก้ของรอบที่แล้ว ๆ มาเข้าด้วยกันจะได้เวกเตอร์ปรับแก้ทิศทางในแต่ละรอบดังในสมการ (2.6.6)

$$\rho^{m+1} = r^m + \beta_{m+1} \rho^m \quad (2.6.6)$$

และถ้าเลือกเวกเตอร์ปรับแก้ที่มีลักษณะเป็นสังยุค (Conjugate) ของกันและกันเทียบกับ A แล้วทำการคูณเชิงสเกลาร์ทั้ง 2 ข้างของสมการ (2.6.6) ด้วย $A\rho^m$ ก็สามารถหาค่า ρ ได้ดังในสมการ (2.6.7)

$$0 = \langle r^m, A\rho^m \rangle + \beta_{m+1} \langle \rho^m, A\rho^m \rangle \quad (2.6.7a)$$

$$\beta_{m+1} = - \frac{\langle r^m, A\rho^m \rangle}{\langle \rho^m, A\rho^m \rangle} \quad (2.6.7b)$$

$$\rho^{m+1} = r^m - \frac{\langle r^m, A\rho^m \rangle}{\langle \rho^m, A\rho^m \rangle} \rho^m \quad (2.6.7c)$$

และสามารถหาค่า x ได้ดังกระบวนการในสมการ (2.6.8)

เริ่มต้น :

$$x^0 = \text{arbitrary} , \rho^0 = r^0 = b - Ax^0 \quad (2.6.8a)$$

ทำซ้ำ : $m = 0, 1, 2, \dots, n$

$$x^{m+1} = x^m + \alpha_m \rho^m \quad (2.6.8b)$$

$$r^{m+1} = r^m - \alpha_m A \rho^m \quad (2.6.8c)$$

โดยที่
$$\alpha_m = \frac{\langle r^m, \rho^m \rangle}{\langle \rho^m, A \rho^m \rangle} \quad (2.6.8d)$$

$$\rho^{m+1} = r^m + \beta_{m+1} \rho^m \quad (2.6.8e)$$

โดยที่
$$\beta_{m+1} = -\frac{\langle r^m, A \rho^m \rangle}{\langle \rho^m, A \rho^m \rangle} \quad (2.6.8f)$$

ในทางทฤษฎีแล้ว วิธีนี้จะเข้าสู่ค่าตอบภายในจำนวน n รอบของการกระทำซ้ำ โดยที่ n คือขนาดของเมตริกซ์ ซึ่งมีค่าเท่ากับจำนวนเวกเตอร์ทั้งหมดที่เป็นสิ่งยุคกัน แต่ในความเป็นจริงจำนวนครั้งที่ใช้อาจเกินกว่าค่านี้ได้ ทั้งนี้เนื่องมาจากความคลาดเคลื่อนสะสมที่เกิดขึ้นในเวกเตอร์ปรับแก้ทิศทาง ทำให้ไม่เป็นสิ่งยุคกันจริง

ในการเร่งอัตราการเข้าสู่ของการทำซ้ำให้เร็วขึ้น สามารถทำได้โดยการแปลงระบบสมการเดิมให้เป็นระบบสมการใหม่ด้วยการคูณระบบสมการเดิมด้วยเมตริกซ์ที่เรียกว่าเมตริกซ์ปรับสภาพล่วงหน้า (Precondition matrix) ดังมีรูปแบบแสดงในสมการ (2.6.9)

$$P^{-1} Ax = P^{-1} b \quad (2.6.9a)$$

$$P^{-1/2} A P^{-1/2} P^{1/2} x = P^{-1/2} b \quad (2.6.9b)$$

โดยทั่วไปจะเลือกรูปแบบในสมการ (2.6.9b) เพราะระบบสมการใหม่จะยังคงสภาพสมมาตร โดยจะได้ระบบสมการใหม่ดังนี้

$$\hat{A} \hat{x} = \hat{b} \quad (2.6.10a)$$

$$\hat{A} = P^{-1/2} A P^{-1/2} \quad (2.6.10b)$$

$$\hat{x} = P^{1/2} x \quad (2.6.10c)$$

$$\hat{b} = P^{-1/2} b \quad (2.6.10d)$$

$$\hat{r} = \hat{b} - \hat{A} \hat{x} = P^{1/2} r \quad (2.6.10e)$$

และเมื่อนำเมตริกซ์ปรับสภาพล่วงหน้ามาใช้ร่วมกับวิธีเกรเดียนต์สังยุค (Conjugate Gradient) ในสมการ (2.6.8) ก็จะได้วิธี PCG ดังมีขั้นตอนแสดงในสมการ (2.6.11)

เริ่มต้น :

$$x^0 = \text{arbitrary}, \rho^0 = r^0 = b - Ax^0 \quad (2.6.11a)$$

ทำซ้ำ : $m = 0, 1, 2, \dots, n$

$$x^{m+1} = x^m + \alpha_m \rho^m \quad (2.6.11b)$$

$$r^{m+1} = r^m - \alpha_m A \rho^m \quad (2.6.11c)$$

โดยที่
$$\alpha_m = \frac{\langle r^m, \rho^m \rangle}{\langle \rho^m, A \rho^m \rangle} \quad (2.6.11d)$$

$$\rho^{m+1} = P^{-1} r^{m+1} + \beta_{m+1} \rho^m \quad (2.6.11e)$$

โดยที่
$$\beta_{m+1} = -\frac{\langle P^{-1} r^{m+1}, A \rho^m \rangle}{\langle \rho^m, A \rho^m \rangle} \quad (2.6.11f)$$

หรืออาจเลือกใช้ค่าตัวคูณในสมการ (2.6.11g) และ (2.6.11h) แทนในสมการ (2.6.11d) และ (2.6.11f) ซึ่งสมมูลกันได้

$$\alpha_m = \frac{\langle P^{-1} r^m, r^m \rangle}{\langle \rho^m, A \rho^m \rangle} \quad (2.6.11g)$$

$$\beta_{m+1} = \frac{\langle P^{-1} r^{m+1}, r^{m+1} \rangle}{\langle P^{-1} r^m, r^m \rangle} \quad (2.6.11h)$$

เมตริกซ์ P ที่จะนำมาใช้เป็นเมตริกซ์ปรับสภาพล่วงหน้าที่ดีนั้นต้องทำให้ $P^{-1} A$ มีค่าใกล้เคียงเมตริกซ์เอกลักษณ์ I (Identity matrix) มากที่สุด ในขณะที่เดียวกันจะต้องหาค่าผกผัน (Inverse) ได้ง่ายเมื่อเทียบกับเมตริกซ์ A นอกจากนี้เมื่อใช้กับวิธี PCG ก็ต้องมีลักษณะเป็นบวกแน่นอนด้วย ในที่นี้เราเลือกใช้เมตริกซ์แปลงรูป ICC (Incomplete Cholesky Matrix) ซึ่งโดยปกติวิธีโคเลสกี้ (Cholesky) เป็นวิธีการแก้สมการโดยตรงโดยการแปลงรูปเมตริกซ์สมประสิทธิ์ A ให้อยู่ในรูปอย่างง่ายดังในสมการ (2.6.12)

$$A = u^T D^{-1} u \quad (2.6.12)$$

โดยที่ u : เป็นเมตริกซ์สามเหลี่ยมบน (Upper Triangular Matrix)

D : เป็นสมาชิกในแนวทแยงของเมตริกซ์ u (Diagonal Matrix)

เมื่อต้องการหาคำตอบของสมการก็แทนค่ากลับจากสมการ (2.6.12) ได้โดยง่าย สำหรับวิธีการแปลงรูปเมตริกซ์มีกระบวนการดังแสดงในสมการ (2.6.13)

initial : u = upper triangular matrix of A

for $r = 1, 2, \dots, n-1$

for $i = r + 1, \dots, n$

$$u_{ij} = u_{ij} / u_{rr} \quad (2.6.13a)$$

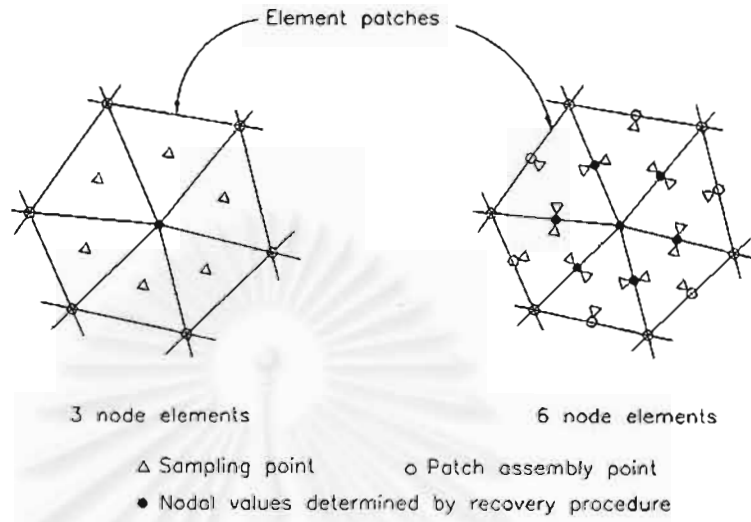
for $j = i + 1, \dots, n$

$$u_{ij} = u_{ij} - u_{ij} \cdot u_{ri} / u_{rr} \quad (2.6.13b)$$

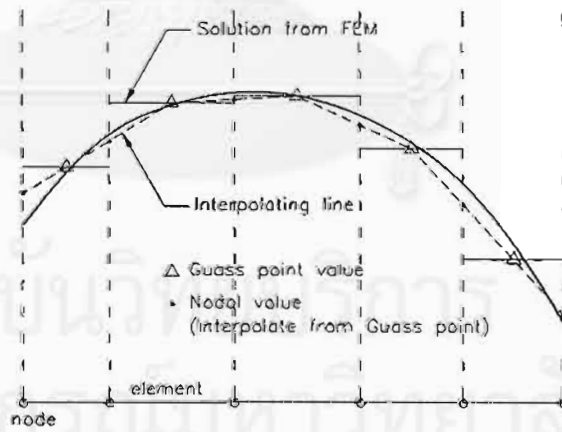
โดยที่ n : ขนาดของเมตริกซ์

ในกรณีของ ICC ก็จะมีลักษณะเดียวกันเพียงแต่จะดำเนินการเฉพาะกับสมาชิกที่มีค่าตั้งแต่เริ่มต้นเท่านั้น ซึ่งเป็นการลดปริมาณการคำนวณลงอย่างมาก โดยเฉพาะในเมตริกซ์ขนาดใหญ่ที่มีจำนวนสมาชิกน้อย และถ้าเมตริกซ์มีสมาชิกครบทุกตำแหน่ง เมตริกซ์แปลงรูปที่ได้จาก ICC ก็จะเหมือนกับในวิธีโคเลสกี้ทุกประการ ข้อควรระวังสำหรับวิธีการนี้คือ ในการนำมาใช้ร่วมกับระเบียบวิธีไฟไนต์เอลิเมนต์ เมื่อชิ้นส่วนมีรูปร่างไม่ดีคือมีอัตราส่วนด้านสั้นต่อด้านยาวต่ำมาก ๆ หรือในกรณีที่เมตริกซ์มีขนาดใหญ่มาก ๆ เช่นในปัญหา 3 มิติ เมตริกซ์แปลงรูปที่ได้จะไม่มีลักษณะเป็นบวกแน่นอน จำเป็นต้องอาศัยการปรับแก้อื่น ๆ ซึ่งสามารถศึกษาเพิ่มเติมได้ในเอกสารอ้างอิง^[32-40]

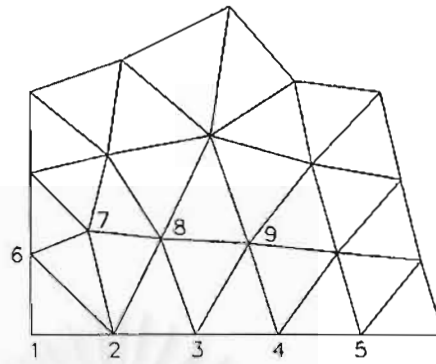
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



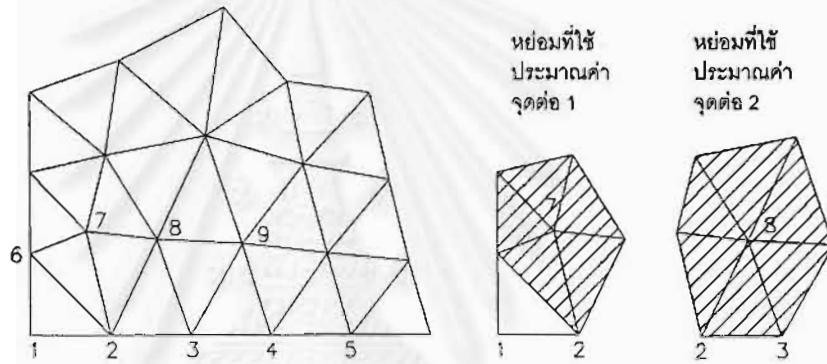
รูปที่ 2.3.1 ตำแหน่งเกาส์ในหย่อมของจุดต่อที่ต้องการประมาณค่า



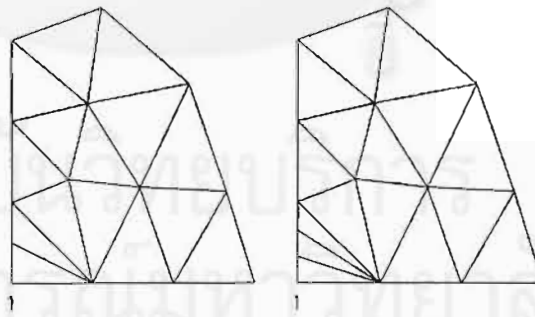
รูปที่ 2.3.2 การประมาณค่าตอบจากตำแหน่งเกาส์ในชั้นส่วนเชิงเส้น 1 มิติ



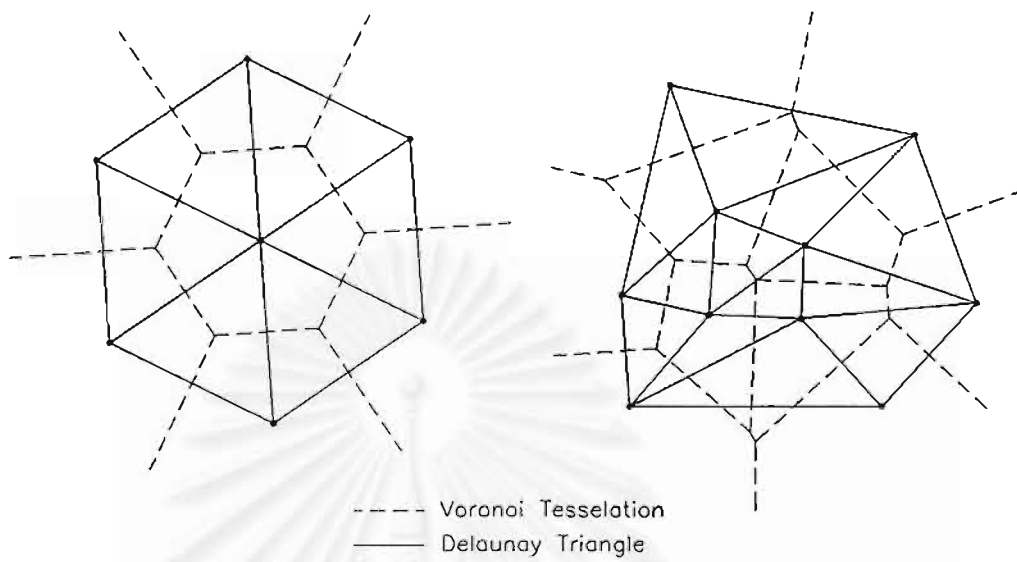
รูปที่ 2.3.3 ลักษณะของชิ้นส่วนบริเวณขอบของปัญหา



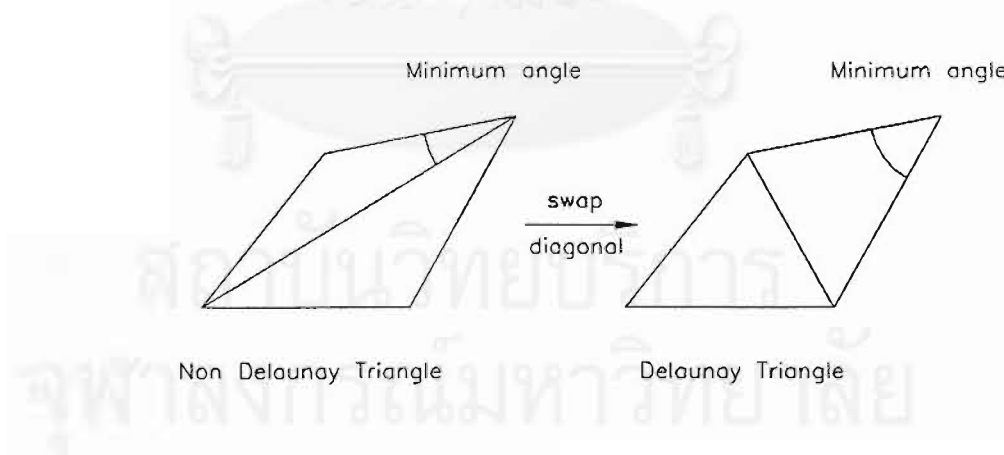
รูปที่ 2.3.4 ลักษณะของห่อหมที่ใช้ในการประมาณค่าหน่วยแรงที่จุดต่อที่ขอบ



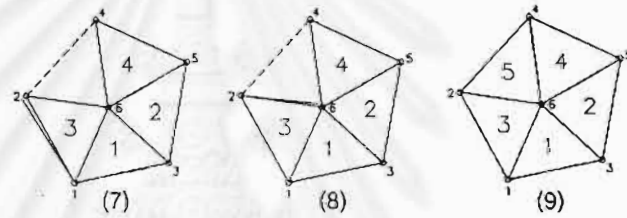
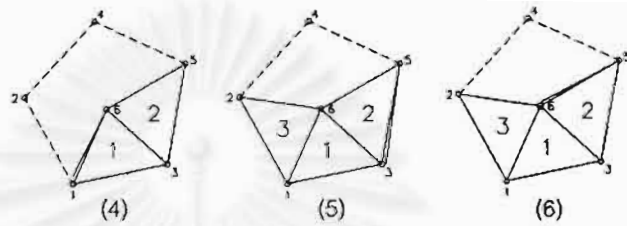
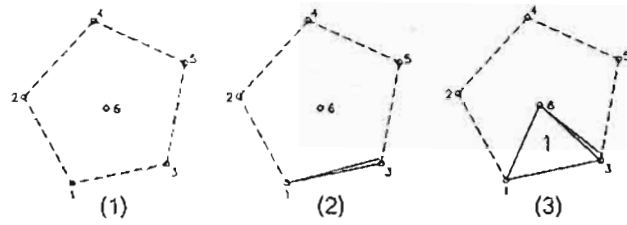
รูปที่ 2.3.5 ลักษณะของจุดต่อที่หาห่อหมภายในได้ยาก



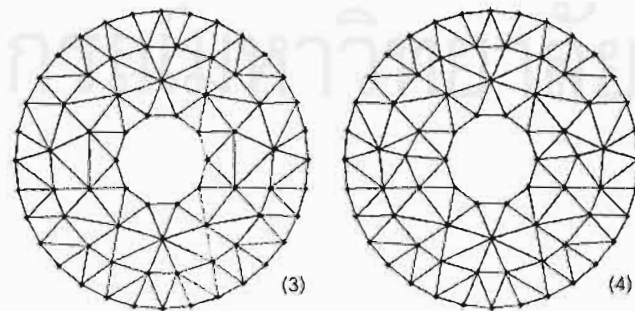
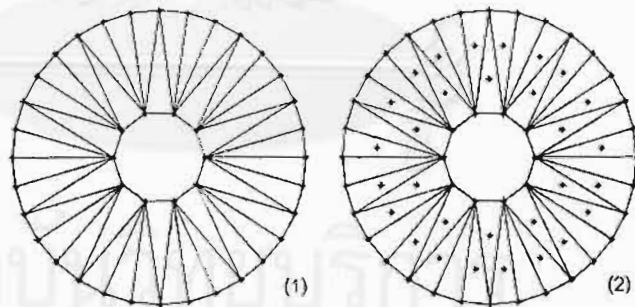
รูปที่ 2.5.1 โครงข่ายโวโรนอยและสามเหลี่ยมเดอลอน



รูปที่ 2.5.2 คุณสมบัติมุมเล็กสุดที่ใหญ่ที่สุด



รูปที่ 2.5.3 ขั้นตอนการสร้างสามเหลี่ยมด้วยการเชื่อมต่อจุด



รูปที่ 2.5.4 ขั้นตอนการเพิ่มจุดควบคู่กับการสร้างสามเหลี่ยมเดอลอนเน

การพัฒนาโปรแกรม

3.1 ภาษาคอมพิวเตอร์ที่ใช้ในโปรแกรม

การเลือกใช้ภาษาที่เหมาะสมกับลักษณะของงานจะส่งผลให้สามารถพัฒนาโปรแกรมได้รวดเร็ว ได้โปรแกรมที่มีประสิทธิภาพ ในงานวิจัยนี้เลือกใช้ภาษาC++ ซึ่งมีจุดเด่นต่าง ๆ ที่เหมาะสมกับกระบวนการปรับขนาดขึ้นส่วนดังต่อไปนี้

ประสิทธิภาพของโปรแกรม ภาษาC++ เป็นภาษาที่ประมวลผลได้รวดเร็วที่สุดภาษาหนึ่งในบรรดาภาษาสูงด้วยกันจะเป็นรองเพียง ภาษาฟอร์แทรน และ ภาษาC เพียงเล็กน้อยเท่านั้น จึงมีความเหมาะสมกับการพัฒนาโปรแกรมไฟไนต์เอลเมนต์ ซึ่งมีการคำนวณและขั้นตอนต่าง ๆ จำนวนมาก

การจองหน่วยความจำแบบพลวัต (Dynamic memory allocation) เนื่องจากกระบวนการปรับขนาดขึ้นส่วนมีการขยายขนาดของข้อมูลชนิดต่าง ๆ ตลอดเวลา การจัดสรรหน่วยความจำให้มีขนาดเหมาะสมกับข้อมูลแต่ละชนิดในขณะใด ๆ ย่อมต้องมีประสิทธิภาพกว่าการจองหน่วยความจำปริมาณมาก ๆ ให้กับข้อมูลทุกชนิดตั้งแต่เริ่มต้นโดยไม่สอดคล้องกับหน่วยความจำที่ใช้จริงในขณะใช้งาน

การเรียกใช้ฟังก์ชันชนิดเรียกตัวเองได้ (Recursive function) ถึงแม้ว่าฟังก์ชันเรียกตัวเองจะทำให้ประสิทธิภาพของโปรแกรมหดลง แต่ในสถานการณ์ที่ไม่รู้จำนวนรอบของการกระทำซ้ำที่เกิดจากการแยกการกระทำซ้ำลงในแต่ละสาขาของข้อมูลที่มีจำนวนไม่แน่นอน เช่น ในกระบวนการค้นหาจุดต่อในพื้นที่ที่กำหนดในโครงสร้างข้อมูลADT หรือการตรวจสอบการเพิ่มจุดในชิ้นส่วนโดยรอบชิ้นส่วนที่พิจารณานั้น ถ้าไม่ใช้ฟังก์ชันเรียกตัวเองก็จะทำให้รหัสคำนวณมีความเยิ่นเย้ออย่างมาก มีผลให้ออกาสผิดพลาดมากขึ้นและอ่านทำความเข้าใจโปรแกรมได้ยาก

การสร้างข้อมูลชนิดใหม่ การสร้างข้อมูลชนิดใหม่ ๆ จะส่งผลให้วิธีการเขียนและโครงสร้างของโปรแกรมมีความหลากหลาย การเขียนรหัสคำนวณมีลักษณะเป็นระเบียบมากขึ้น รหัสคำนวณมีความกระชับ ลดจำนวนตัวแปรที่อาจต้องส่งผ่านให้กับฟังก์ชันจำนวน 10-12 ตัวเหลือเพียง 3-4 ตัว เพิ่มความเข้าใจในโปรแกรมขึ้นอย่างมาก ส่งผลให้การตรวจสอบ ปรับปรุงแก้ไขหรือพัฒนาโปรแกรมต่อไปทำได้ง่าย

การเข้าถึงตำแหน่งของหน่วยความจำโดยตัวชี้ (Pointer) ตัวชี้เป็นข้อมูลชนิดหนึ่งซึ่งอ้างอิงถึงตำแหน่งของหน่วยความจำ สามารถตอบสนองต่อคำสั่งและดำเนินการใด ๆ ได้เหมือนกับสิ่งที่อ้างอิงไปถึง ซึ่งจะเป็นอะไรก็ได้

- เช่น ข้อมูลชนิดต่าง ๆ ฟังก์ชัน โครงสร้างข้อมูล หรือข้อมูลชนิดใหม่ ๆ ที่สร้างขึ้นมา ฯลฯ จึงส่งผลให้โปรแกรมมีประสิทธิภาพมากขึ้น ลดจำนวนของหน่วยความจำที่ใช้เก็บข้อมูลลง และสามารถเข้าถึงข้อมูลต่าง ๆ ได้อย่างรวดเร็ว

จะเห็นว่ามีการใช้คุณสมบัติต่าง ๆ ของภาษาC++ เป็นจำนวนมาก ดังนั้นในการปรับปรุงโปรแกรมเป็นภาษาอื่นควรคำนึงถึงคุณสมบัติเหล่านี้ด้วย เพื่อไม่ให้เป็นการยากลำบากเกินไปในการเขียนรหัสคำนวณใหม่ให้สอดคล้องกับรหัสเดิมที่มีอยู่

3.2 การเลือกใช้โครงสร้างข้อมูล

โครงสร้างข้อมูลที่ใช้ในงานวิจัยนี้แบ่งเป็นประเภทใหญ่ ๆ ได้ 3 ประเภทดังมีรายละเอียดต่อไปนี้

1.) โครงสร้างข้อมูลชนิดแถวลำดับ (Array) เป็นโครงสร้างข้อมูลที่จองหน่วยความจำขนาดใหญ่ มีหน่วยความจำของแต่ละข้อมูลอยู่ติดกันจึงสามารถเข้าถึงข้อมูลแต่ละตัวได้อย่างรวดเร็ว แต่โดยปกติโครงสร้างข้อมูลชนิดนี้ต้องจองหน่วยความจำล่วงหน้าและเปลี่ยนแปลงขนาดไม่ได้ (Static memory allocation) จึงไม่เหมาะกับกระบวนการปรับขนาดนักเพราะในขณะที่จองหน่วยความจำยังไม่ทราบขนาดของข้อมูล แต่เนื่องจากความรวดเร็วในการอ้างอิงถึงข้อมูลมีความจำเป็น เช่น ในการดำเนินการต่าง ๆ เกี่ยวกับเมตริกซ์ จึงต้องจำจองโครงสร้างข้อมูลชนิดนี้โดยการจองหน่วยความจำขนาดใหญ่แบบพลวัตแทน ดังเช่นที่ใช้กับข้อมูลชนิด vector การอ้างอิงถึงแต่ละข้อมูลก็อ้างอิงถึงตำแหน่งที่ห่างออกจากหน่วยความจำแรกไปเท่ากับหมายเลขข้อมูลคูณด้วยขนาดของข้อมูลนั้น ลักษณะของโครงสร้างข้อมูลชนิดนี้สามารถแสดงได้ดังในรูปที่ 3.2.1

2.) โครงสร้างข้อมูลชนิดลิงค์ลิสต์ (Linklist) เป็นโครงสร้างข้อมูลที่แต่ละข้อมูลอยู่กระจัดกระจายกันออกไป โดยมีการเชื่อมโยงข้อมูลถึงกันข้อมูลต่อข้อมูล ทำให้ไม่สามารถเข้าถึงข้อมูลตัวใดตัวหนึ่งได้ ต้องสืบค้นจากข้อมูลตัวแรกไล่ไปเรื่อย ๆ จนกระทั่งพบตัวที่ต้องการ แต่ข้อดีของโครงสร้างข้อมูลชนิดนี้คือมีความยืดหยุ่นสูงในการเพิ่มลดข้อมูล และถ้าใช้กับกระบวนการที่ต้องกระทำกับข้อมูลทุกตัวไล่กันไป ก็จะมีประสิทธิภาพไม่ยิ่งหย่อนไปกว่าโครงสร้างข้อมูลชนิดแถวลำดับแต่อย่างใด ชนิดของข้อมูลที่เหมาะสมกับโครงสร้างชนิดนี้คือข้อมูลที่มีการเพิ่มหรือลดจำนวนตลอดเวลา และการนำข้อมูลมาใช้ก็มีการเรียงลำดับตั้งแต่ต้นจนจบ ตัวอย่างของชนิดข้อมูลที่ใช้ได้แก่ ข้อมูลชนิด front ซึ่งจำนวนข้อมูลมีการเปลี่ยนแปลงตลอดเวลาและการใช้ข้อมูลก็เลือกใช้ตัวใดก่อนก็ได้ไม่จำเป็นต้องจำเพาะลงไป ลักษณะของโครงสร้างข้อมูลชนิดนี้แสดงได้ดังในรูปที่ 3.2.2

3.) โครงสร้างข้อมูลชนิดต้นไม้เชิงตัวเลขสลับ ADT (Alternate Digital Tree) พื้นฐานของโครงสร้างข้อมูลชนิดนี้มาจากโครงสร้างข้อมูลชนิดต้นไม้ทวิภาค (binary tree) ซึ่งแต่ละข้อมูลจะเก็บตำแหน่งของข้อมูลอื่นอีก 2 ตำแหน่งโดยมีเงื่อนไขในการเก็บข้อมูลต่างกันออกไป เช่น ถ้าต้องการเก็บข้อมูลเพื่อความสะดวกในการค้นหาข้อมูลที่มีค่าอยู่ในช่วงที่ต้องการ ก็อาจตั้งเงื่อนไขว่าถ้าข้อมูลที่เพิ่มมีค่ามากกว่าให้เก็บในฝั่งซ้าย ด้านย่อยกว่าให้เก็บในฝั่งขวา ดังนั้นถ้าต้องการหาข้อมูลที่มีค่ามากกว่าข้อมูลในตำแหน่งใด ข้อมูลที่เป็นบริวารในฝั่งขวาทั้งหมดของข้อมูลในตำแหน่งนั้นก็ไม่ต้องพิจารณา ส่วน ADT จะแตกต่างจากต้นไม้ทวิภาคตรงที่การค้นหาข้อมูลจะเป็นการค้นหาที่อ้างอิง

ตำแหน่งทางเรขาคณิต ดังนั้นในแต่ละข้อมูลจะมีการเก็บตำแหน่งพื้นที่ครอบครองของตัวเองไว้ด้วยเพื่อใช้ในการอ้างอิงกับตำแหน่งที่ต้องการค้นหา และข้อมูลที่เป็นบริวารซึ่งอยู่ในพื้นที่นั้นก็มีพื้นที่ครอบครองที่แบ่งจากพื้นที่ครอบครองของข้อมูลต้น โดยมีหลักในการเก็บข้อมูลดังนี้ รูป 3.2.3 ประกอบ จุดแรกที่เก็บเข้าไปในโครงสร้างจะถือว่าครอบครองพื้นที่ทั้งหมด เมื่อมีการเพิ่มจุดต่อ ๆ มาพื้นที่ที่ครอบครองโดยจุดนั้นก็จะถูกแบ่งให้เล็กลงและให้เป็นพื้นที่ครอบครองของจุดที่เพิ่มเข้ามาโดยที่พื้นที่ของจุดต้นไม่มีการเปลี่ยนแปลง ดังเช่นจุดที่ 2 ในรูป 3.2.3 ก็จะมีพื้นที่ครอบครองเป็นผืนผ้าของพื้นที่ทั้งหมดและตัวชี้ทางผืนผ้าของจุด 1 ก็จะอ้างอิงไปยังข้อมูลของจุด 2 เมื่อเพิ่มจุด 3 ซึ่งอยู่ในพื้นที่ของจุด 2 พื้นที่ครอบครองของจุด 3 ก็จะเป็นครึ่งหนึ่งของพื้นที่ครอบครองของจุด 2 คือพื้นที่ 1 ใน 4 ทางผืนผ้าบนของพื้นที่ทั้งหมดพร้อมทั้งเชื่อมโยงตำแหน่งข้อมูลของจุด 3 ไปเก็บยังผืนผ้าของจุด 2 จุดที่ 4 ก็มีลักษณะเช่นเดียวกับจุด 3 สำหรับจุดที่ 5 เป็นจุดแรกที่มีตำแหน่งอยู่บนผืนผ้าของพื้นที่ ซึ่งเป็นพื้นที่ที่แบ่งจากจุด 1 และยังไม่มีการครอบครอง จุด 5 ก็จะครอบครองพื้นที่ผืนผ้าทั้งหมดและเชื่อมโยงตำแหน่งข้อมูลของจุด 5 ไปเก็บยังผืนผ้าของจุด 1 เมื่อมีการจัดเก็บข้อมูลครบทั้งหมด ทุก ๆ จุดก็จะมีพื้นที่ครอบครองของตนเอง และเมื่อต้องการค้นหาจุดที่อยู่ในพื้นที่ที่สนใจก็ตรวจสอบจากพื้นที่ครอบครองของจุดไล่ตามลำดับลงมา โดยตรวจสอบเฉพาะจุดที่มีพื้นที่ครอบครองซ้อนทับกับพื้นที่ที่สนใจเท่านั้น สำหรับในจุดที่พื้นที่ไม่ซ้อนทับกันก็ไม่จำเป็นต้องพิจารณาจุดนั้น ๆ และจุดบริวารทั้งหมดของจุดนั้นที่เชื่อมโยงไปถึง เช่นถ้าพื้นที่ที่สนใจคือพื้นที่ผืนผ้าของพื้นที่ทั้งหมด จำนวนจุดที่ไม่ต้องพิจารณาก็คือทุกจุดที่อยู่ในผืนผ้าของจุด 1 ในแผนภูมิดังกล่าวในรูป 3.2.3 ยิ่งพื้นที่สนใจมีขนาดเล็กเท่าใดจำนวนจุดที่ไม่ต้องพิจารณาก็ยิ่งมีมากขึ้น

สำหรับโครงสร้างข้อมูลชนิด ADT ที่ใช้ในงานวิจัยนี้จะแตกต่างจากที่กล่าวมาเล็กน้อยคือในแต่ละจุดจะเชื่อมโยงไปยังจุดบริวาร 4 จุดในแต่ละภาคของพื้นที่ ทำให้ไม่ต้องคำนวณหาลำดับขั้นและทิศทางของการแบ่งพื้นที่ในแต่ละจุด

3.3 ชนิดของข้อมูล

ชนิดของข้อมูลที่ใช้ในโปรแกรมมีจำนวนมากภายในที่นี้จะกล่าวถึงเฉพาะที่สำคัญที่ช่วยให้เข้าใจความสัมพันธ์และการทำงานของโปรแกรมเท่านั้น ดังมีรายละเอียดดังนี้

eline : เป็นข้อมูลที่แทนเส้นเชื่อมระหว่างจุดต่อของชิ้นส่วนโดยมีข้อมูลต่าง ๆ ดังนี้

1.) ตัวชี้ที่ชี้ไปยังตำแหน่งของจุดต่อที่ต้นและปลายของเส้นเชื่อม
2.) ตัวชี้ที่ชี้ไปยังชิ้นส่วนทางด้านซ้ายและขวา เพื่อความสะดวกในการอ้างอิงไปยังชิ้นส่วนต่าง ๆ ผ่านทางเส้นเชื่อมจุด จะมีค่าเป็น 0 เมื่อไม่มีชิ้นส่วนทางด้านนั้น

3.) ตัวชี้ที่ชี้ไปยังสมาชิกในฟรอนท์ที่อ้างอิงกลับมา เพื่อประโยชน์ในการปรับปรุงฟรอนท์เมื่อมีการสร้างชิ้นส่วน จะชี้ไปยังตำแหน่ง 0 ถ้าเส้นเชื่อมไม่ได้อยู่ในฟรอนท์

enode : เป็นข้อมูลที่แทนจุดต่อของชิ้นส่วน มีข้อมูลที่เก็บคือ

1.) ค่าพิกัด x, y ของจุดต่อ

2.) ค่าระยะห่างของจุดที่กำหนดไว้เพื่อใช้ตรวจสอบในการสร้างจุดใหม่ และในบางครั้งใช้เก็บหมายเลขของจุดต่อเพื่อใช้อ้างอิงไปยังสมาชิกใน vector

3.) ลิงค์ลิสต์ของ eline ที่เชื่อมกับจุดนี้ทั้งหมด มีประโยชน์เพื่อใช้ในการตรวจสอบการเชื่อมโยงไปยังจุดอื่น ๆ และใช้ค้นหาชิ้นส่วนที่อยู่รอบจุดในกรณีคำนวณหน่วยแรงที่จุดด้วยวิธีการพื้นตัวเป็นหย่อม การที่เก็บเป็นลิงค์ลิสต์เพราะจำนวนเส้นที่เชื่อมจุดต่อในแต่ละจุดมีจำนวนไม่เท่ากัน การเก็บเป็นลิงค์ลิสต์จะประหยัดเนื้อที่ที่สุด การเก็บข้อมูลจะเก็บเป็นตัวชี้ไปยังตำแหน่งของเส้นเชื่อม และมีการเรียงลำดับในลักษณะทวนเข็มนาฬิกา

element : เป็นข้อมูลแทนชิ้นส่วน โดยมีข้อมูลที่สำคัญคือ

1.) flag สำหรับใช้ตรวจสอบการเพิ่มจุดต่อที่กึ่งกลางชิ้นส่วน โดยตรวจสอบว่าที่ชิ้นส่วนโดยรอบมีการเพิ่มจุดหรือยัง และมีตำแหน่งใกล้เคียงไปหรือไม่ นอกจากนี้บางครั้งยังใช้เก็บหมายเลขของชิ้นส่วนเพื่อใช้ในการอ้างอิง

2) ตัวชี้ไปยัง eline ที่เป็นด้านทั้ง 3 ของชิ้นส่วน การที่เก็บ eline แทน enode เพราะมีความสะดวกกว่าในการอ้างอิงไปยังชิ้นส่วนอื่น ๆ เนื่องจากข้อมูลใน eline มีน้อยกว่าทำให้อ้างอิงได้เร็วขึ้น

vector : เป็นข้อมูลที่จำลองโครงสร้างข้อมูลชนิดแถวลำดับ คือมีการเข้าถึงข้อมูลโดยการอ้างอิงหมายเลขของข้อมูลโดยตรง เพื่อให้มีความสะดวกและรวดเร็วในการแก้สมการหลายตัวแปรที่จำเป็นต้องมีการคูณเชิงสเกลาร์และการคูณระหว่างเมตริกซ์และเวกเตอร์

symmat : เป็นข้อมูลที่แทนเมตริกซ์ชนิดสมมาตร การเก็บข้อมูลจะเก็บข้อมูลชนิด dia_ele ซึ่งเป็นข้อมูลที่แทนค่าสมาชิกในแนวทแยงของเมตริกซ์ โดยจะเก็บข้อมูลเป็นเวกเตอร์ทำให้เข้าถึงสมาชิกในแนวทแยงของเมตริกซ์ได้อย่างรวดเร็ว

dia_ele : เป็นข้อมูลที่อยู่ใน symmat จะเก็บค่าสมาชิกในแนวทแยงของเมตริกซ์พร้อมทั้งหมายเลขประจำแถว และเก็บลิงค์ลิสต์ที่เชื่อมโยงไปยังสมาชิกในแนวนอนและแนวตั้งของเมตริกซ์ที่สัมพันธ์กับ dia_ele คือ hor_ele และ ver_ele

hor_ele : เป็นข้อมูลที่เก็บค่าสมาชิกของเมตริกซ์และเก็บหมายเลขสดมภ์ของสมาชิกพร้อมทั้งเชื่อมโยงไปยังสมาชิกตัวอื่น ๆ ที่อยู่ในแถวเดียวกัน

ver_ele : เป็นข้อมูลที่เก็บตัวชี้ที่ชี้ไปยังข้อมูลสมาชิกใน hor_ele และเก็บหมายเลขแถวของสมาชิกนั้น ๆ พร้อมทั้งเชื่อมโยงไปยังสมาชิกอื่นที่อยู่ในสดมภ์เดียวกัน สาเหตุที่ต้องเก็บข้อมูลชนิดนี้ก็เพื่อความสะดวกรวดเร็วในการอ้างอิงถึงสมาชิกในการคูณกับเวกเตอร์

ความสัมพันธ์ของข้อมูลชนิด symmat , dia_ele , hor_ele และ ver_ele สามารถแสดงได้ดังในรูป 3.3.1 การเก็บข้อมูลของเมตริกซ์ในลักษณะนี้จะประหยัดหน่วยความจำได้มากและการลดรูปของ ICC ก็ทำได้อย่างมีประสิทธิภาพเพราะเป็นกระบวนการที่ต้องกระทำเรียงไปตามลำดับในแต่ละสมาชิกอยู่แล้ว การคูณกับเวกเตอร์ก็เช่นกัน สามารถอ้างอิงไปยังสมาชิกในเวกเตอร์ได้โดยตรงโดยใช้หมายเลขประจำแถวและสดมภ์ที่เก็บอยู่ในแต่ละสมาชิก

boundary : เป็นข้อมูลแรกที่จะเก็บข้อมูลที่ได้จากการนำเข้าจากแฟ้มข้อมูล โดยเป็นข้อมูลของขอบเขตของปัญหา ซึ่งมีข้อมูลย่อยต่อไปนี้

- 1.) ค่าพิกัด x, y ของตำแหน่งต้นและปลายของเส้นขอบ
- 2.) รัศมีของเส้นขอบในกรณีที่เป็นเส้นโค้ง จะมีค่า 0 ถ้าเป็นเส้นตรง มีค่าเป็นบวกเมื่อจุดศูนย์กลางของเส้นโค้งอยู่ทางด้านซ้ายของเส้นขอบ มีค่าเป็นลบเมื่อจุดศูนย์กลางของเส้นโค้งอยู่ทางด้านขวาของเส้นขอบ เพราะฉะนั้นถ้าเส้นขอบของปัญหามีขนาดใหญ่กว่าครึ่งวงกลมจำเป็นต้องมีการแบ่งเป็น 2 เส้น
- 3.) จำนวนของเส้นย่อยที่จะแบ่งจากเส้นขอบ ทั้งนี้เพื่อความกระชับในการเตรียมข้อมูลและสะดวกในการปรับขนาดของชิ้นส่วน
- 4.) ตัวชี้ที่ไปยังข้อมูลชนิด `bound_load` ซึ่งจะเก็บข้อมูลของแรงที่กระทำต่อขอบเขตของปัญหาโดยจะแยกเป็นแรงที่กระทำเป็นจุดและแรงที่กระจายตลอดแนวขอบ
- 5.) ตัวชี้ที่ไปยังข้อมูลชนิด `bound_support` ซึ่งจะเก็บข้อมูลของที่รองรับที่ขอบเขตของปัญหา มีทั้งที่รองรับที่เป็นจุดและรองรับตลอดแนวของเส้นขอบ

`b_line` : เป็นข้อมูลที่ผสมผสานระหว่าง `boundary` และ `eline` คือมีชนิดข้อมูลทั้งหมดของ `eline` และมีชนิดข้อมูลใน `boundary` ด้วยยกเว้นจำนวนที่จะแบ่งเป็นเส้นย่อยและค่าพิกัด x, y ซึ่งจะเป็นตัวชี้ไปยัง `enode` แทน ทั้งนี้ข้อมูลชนิดนี้มีเพื่อใช้เป็นฟรอนต์เริ่มต้นในกระบวนการเชื่อมต่อดังกล่าวเป็นสามเหลี่ยมหลังการเพิ่มจุด และใช้สร้างขอบของปัญหาใหม่เมื่อมีการลดขนาดของชิ้นส่วนลง

นอกจากข้อมูลที่ได้กล่าวมาแล้วยังมีข้อมูลชนิดอื่นอีกมาก แต่จะเว้นไว้ไม่กล่าวถึงในที่นี้เพราะสามารถศึกษาจากรหัสคำนวณเองได้ และไม่มีส่วนช่วยในการเข้าใจความสัมพันธ์และลำดับการทำงานของโปรแกรม

3.4 โครงสร้างและลำดับการทำงานของโปรแกรม

โครงสร้างการทำงานของโปรแกรมทั้งในส่วนโปรแกรมหลักและโปรแกรมย่อย สามารถติดตามได้ในรูปที่ 3.4.1 - 3.4.5 โดยมีรายละเอียดของลำดับการทำงานดังนี้

โปรแกรมหลัก (รูปที่ 3.4.1) เริ่มจากนำเข้าข้อมูลขอบเขตของปัญหา แรงกระทำและที่รองรับต่าง ๆ เข้ามาเก็บในลิสต์ของ `boundary` จากนั้นแบ่งขอบเขตปัญหาเป็นเส้นย่อย ๆ เก็บพิกัดของจุดไว้ใน ADT และเก็บเส้นเชื่อมโยงจุดพร้อมทั้งแรงกระทำและที่รองรับไว้ในลิสต์ของ `b_line` จากข้อมูลใน ADT และ `b_line` นำมาสร้างชิ้นส่วนสามเหลี่ยมจะได้สามเหลี่ยมที่มีขนาดสอดคล้องกับระยะห่างของจุดที่ขอบ เก็บข้อมูลของชิ้นส่วนที่ได้ในลิสต์ของ `element` ต่อมาทำการคำนวณหาการกระจัดของแต่ละจุดต่อโดยการสร้างสติเฟเนสจากแต่ละชิ้นส่วน และแก้สมการด้วยวิธีเกรเดียนต์สังยุคแบบมีเงื่อนไขล่วงหน้า เก็บการกระจัดที่ได้ใน `vector u` เมื่อได้การกระจัดมาแล้วคำนวณหน่วยแรง ค่าคลาดเคลื่อนและขนาดของชิ้นส่วนที่สอดคล้องกับค่าคลาดเคลื่อนในแต่ละจุดต่อจากหอย่อมของชิ้นส่วนที่สัมพันธ์กับจุดต่ออื่น ๆ แสดงผลที่คำนวณได้ทางแฟ้มข้อมูล จากนั้นตรวจสอบว่าความคลาดเคลื่อนรวมทั้งหมดมีค่าเกินกว่าที่ยอมให้หรือไม่ ถ้าไม่เกินก็หยุดการทำงาน แต่ถ้าเกินก็ปรับขนาดของชิ้นส่วนใหม่โดยนำข้อมูลจากลิสต์

ของ b_line มาสร้างขอบของปัญหาใหม่เก็บไว้ในลิสต์ของ boundary ก็จะได้ขอบของปัญหาตัวใหม่สำหรับวิเคราะห์ต่อไป กระทำซ้ำในลักษณะเดิมจนกระทั่งค่าคลาดเคลื่อนมีค่าต่ำกว่าที่ยอมให้ก็หยุดการทำงาน

การสร้างชิ้นส่วนสามเหลี่ยม (รูปที่ 3.4.2) หลังจากได้ข้อมูลในลิสต์ของ b_line และ ADT แล้ว สร้างพรีอนท์เริ่มต้นจาก b_line และเลือกเส้นเส้นหนึ่งจากในพรีอนท์ ทำการค้นหาจุดที่จะเป็นจุดยอดของสามเหลี่ยมร่วมกับเส้นนั้นพร้อมทั้งเชื่อมโยงจุดเป็นชิ้นส่วนสามเหลี่ยม เก็บชิ้นส่วนที่ได้ในลิสต์ของ element ทำการปรับปรุงพรีอนท์และดำเนินการกับเส้นอื่น ๆ จนกระทั่งไม่เหลือเส้นใด ๆ ในพรีอนท์อีก ก็ตรวจสอบในแต่ละชิ้นส่วนว่าสามารถเพิ่มจุดได้หรือไม่ ถ้าได้ก็เพิ่มจุดลงใน ADT จนกระทั่งครบทุกชิ้นส่วน จากนั้นสร้างชิ้นส่วนและเพิ่มจุดในลักษณะเดิมจนกระทั่งไม่สามารถเพิ่มจุดใด ๆ ได้อีก ก็จะส่งคืนการควบคุมให้แก่โปรแกรมหลัก

การหาจุดที่เหมาะสมสำหรับแต่ละเส้นในการสร้างสามเหลี่ยม (รูปที่ 3.4.3) เริ่มจากการกำหนดขนาดของพื้นที่ค้นหาให้มีขนาดเริ่มต้นเท่ากับขนาดของเส้นที่จะค้นหา นั้น จากนั้นจึงค้นหาจุดทั้งหมดที่อยู่ในพื้นที่นี้จาก ADT ตรวจสอบจุดที่ได้ทั้งหมดเพื่อหาจุดที่สร้างวงกลมได้รัศมีต่ำสุด แล้วนำรัศมีที่ได้ไปตรวจสอบกับขนาดของพื้นที่ค้นหา ถ้าเส้นผ่านศูนย์กลางของวงกลมมีขนาดใหญ่กว่าด้านของพื้นที่ค้นหาแสดงว่าอาจมีจุดใดจุดหนึ่งนอกพื้นที่ค้นหาที่อยู่ในรัศมีของวงกลม ดังนั้นขยายขนาดพื้นที่เป็น 2 เท่าและทำการค้นหาและตรวจสอบในลักษณะเดิมจนกระทั่งรัศมีที่ได้เล็กกว่าพื้นที่ค้นหา ก็ส่งคืนตำแหน่งจุดที่ได้ให้แก่โปรแกรมหลัก

การตรวจสอบการเพิ่มจุดในแต่ละชิ้นส่วน (รูปที่ 3.4.4) เริ่มจากตรวจสอบว่าชิ้นส่วนมีขนาดใหญ่พอให้เพิ่มจุดที่กึ่งกลางได้หรือไม่ ถ้าไม่ได้ให้ส่งค่า ไม่ คืนแก่โปรแกรมหลัก แต่ถ้าได้ต้องตรวจสอบชิ้นส่วนที่อยู่รอบ ๆ ว่ามีการเพิ่มจุดใหม่ที่อยู่ในระยะกำหนดหรือไม่โดยมีหลักดังนี้ เริ่มจากเส้นด้านซ้ายของชิ้นส่วน ตรวจสอบชิ้นส่วนที่ติดกันที่กึ่งกลางของชิ้นส่วนอยู่ห่างจากจุดที่จะเพิ่มมากกว่าระยะที่กำหนดหรือไม่ ถ้ามากกว่าก็ตรวจสอบด้านที่เหลือต่อไป แต่ถ้าน้อยกว่าต้องตรวจสอบต่อไปว่ามีการเพิ่มจุดแล้วหรือไม่ ถ้ามีก็ส่งคืนค่า ไม่ แก่โปรแกรมหลัก แต่ถ้าไม่มีก็ต้องตรวจสอบกับชิ้นส่วนอื่น ๆ ที่ติดกับชิ้นส่วนทางด้านซ้ายนี้จนกว่าจะพบว่าไม่มีจุดอยู่ในระยะที่กำหนดก็ส่งคืนค่า ไม่ แก่โปรแกรมหลัก หรือจนกระทั่งพบว่ากึ่งกลางของชิ้นส่วนอยู่นอกระยะที่กำหนดก็ตรวจสอบกับชิ้นส่วนอื่นต่อไป ตรวจสอบในลักษณะเดียวกันนี้กับเส้นทางด้านขวาและด้านล่างของชิ้นส่วนที่ตรวจสอบ ถ้าตรวจสอบผ่านหมดส่งคืนค่า ได้ แก่โปรแกรมหลัก

การปรับปรุงขนาดของชิ้นส่วนที่ขอบปัญหาและกำหนดระยะห่างของจุดภายใน (รูปที่ 3.4.5) เริ่มจากการสร้าง ADT ตัวใหม่พร้อมทั้งสร้างลิสต์ของ boundary ใหม่แทนของเดิมโดยใช้ข้อมูลจาก b_line และขนาดของชิ้นส่วนที่ได้คำนวณไว้ จากนั้นแบ่งเส้นขอบในลิสต์ของ boundary เป็นเส้นย่อยโดยเก็บพิกัดใน ADT ตัวใหม่และเก็บข้อมูลของขอบเขตในลิสต์ของ b_line ขึ้นต่อมานำจุดภายในจาก ADT ตัวเก่ามาเพิ่มให้กับ ADT ตัวใหม่พร้อมทั้งกำหนดขนาดจากที่ได้คำนวณไว้ เมื่อถ่ายโอนข้อมูลหมดแล้วคืนหน่วยความจำของ ADT ตัวเก่าแล้วอ้างอิงมายัง ADT ตัวใหม่แทน สุดท้ายคืนการควบคุมให้แก่โปรแกรมหลัก

3.5 การแสดงผล

การแสดงผลของโปรแกรมอยู่ในรูปแฟ้มข้อมูล (file) 2 ประเภทคือ

1.) แฟ้มข้อมูลที่เป็นตัวหนังสือที่สามารถเปิดอ่านและแก้ไขข้อมูลได้ด้วยโปรแกรมแก้ไขคำ (Text editor) ทั่วไป แฟ้มประเภทนี้มี 2 แฟ้มคือ

1.1) แฟ้มแสดงตำแหน่งของจุดต่าง ๆ และแสดงความสัมพันธ์ระหว่างชิ้นส่วนและจุดต่อ เพื่อนำไปปรับปรุงให้เป็นข้อมูลสำหรับโปรแกรมอื่นเพื่อเปรียบเทียบผลต่อไป

1.2) แฟ้มข้อมูลแสดงผลการคำนวณของแต่ละจุด โดยจะแสดงหมายเลขจุดต่อ การการจัดหน่วยแรง และความคลาดเคลื่อนในจุดต่อนั้น ๆ

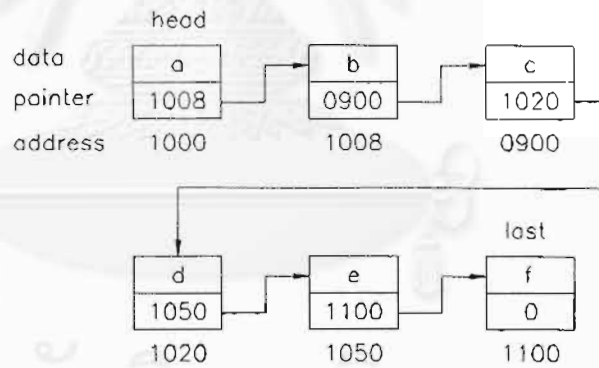
2.) แฟ้มข้อมูลแลกเปลี่ยนประเภทกราฟฟิก เพื่อใช้ในกระบวนการแสดงผลทางกราฟฟิกโดยโปรแกรม CAD ต่าง ๆ ในที่นี้เลือกใช้แฟ้มมาตรฐานประเภทแฟ้มแลกเปลี่ยนข้อมูล DXF (Data Exchange File) ซึ่งจะมี 2 ประเภทคือ ประเภทที่เป็นแฟ้มตัวอักษร (Text file) และแฟ้มตัวเลขฐานสอง (Binary file) แต่จะเลือกใช้แฟ้มประเภทตัวอักษรเพื่อความสะดวกในการแก้ไขตรวจสอบด้วยโปรแกรมแก้ไขคำ โดยจะแสดงโครงข่ายของชิ้นส่วน การจัดหมายเลขจุดต่อ หมายเลขชิ้นส่วน และเส้นคอนทัวร์ของหน่วยแรงต่าง ๆ ในการตรวจสอบผลการแสดงก็สามารถเปิดหรือปิดเลเยอร์ในโปรแกรม CAD เพื่อพิจารณาเฉพาะส่วนที่สนใจได้ สำหรับรูปแบบของแฟ้มมาตรฐานชนิดนี้สามารถศึกษาได้ในคู่มือของโปรแกรม CAD ทั่ว ๆ ไป

data	a	b	c	d	e		char
address	1000	1001	1002	1003	1004		

data	1	2	3	4	5		int
address	1000	1002	1004	1006	1008		

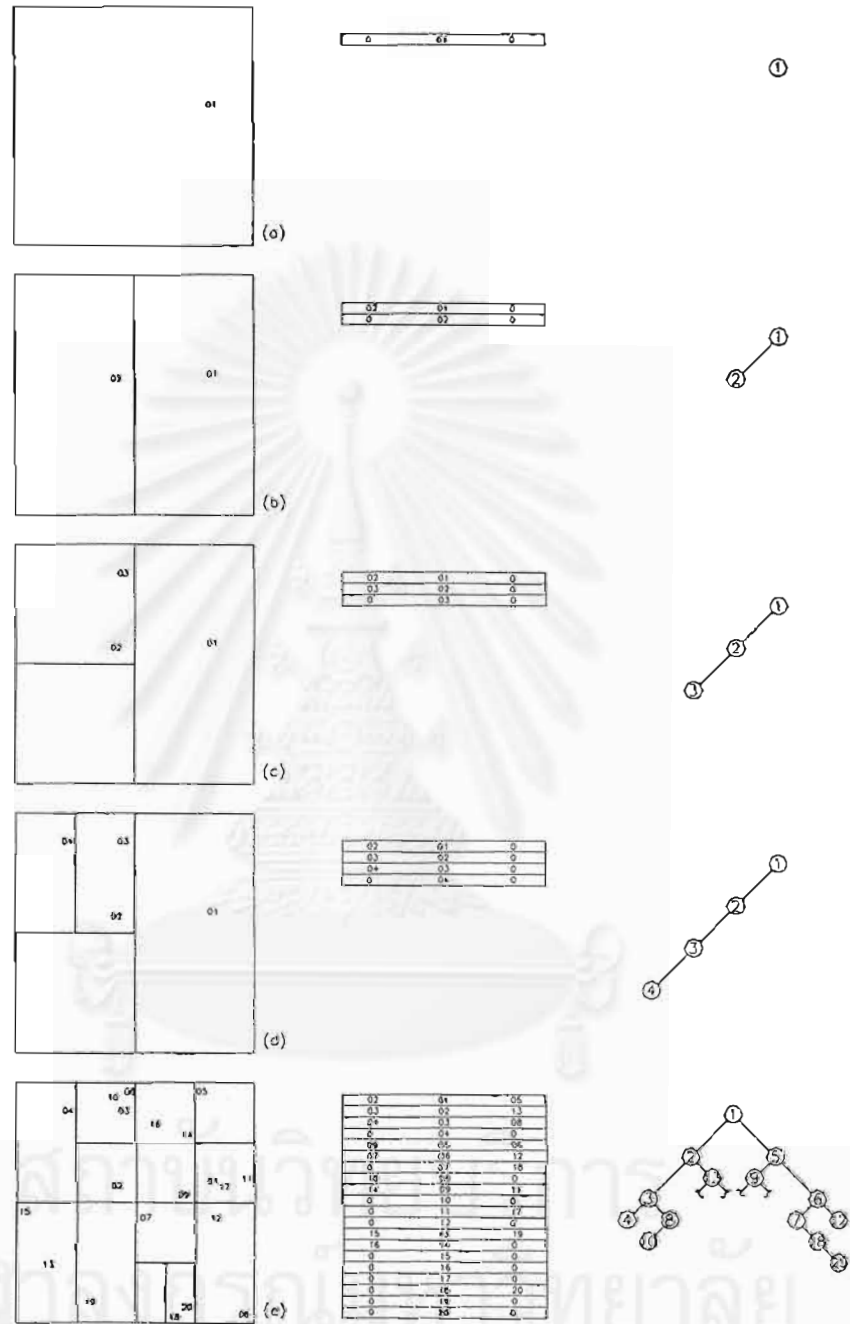
data	1	2	3	4	5		long int
address	1000	1004	1008	1012	1016		

รูปที่ 3.2.1 โครงสร้างข้อมูลชนิดแถวลำดับ

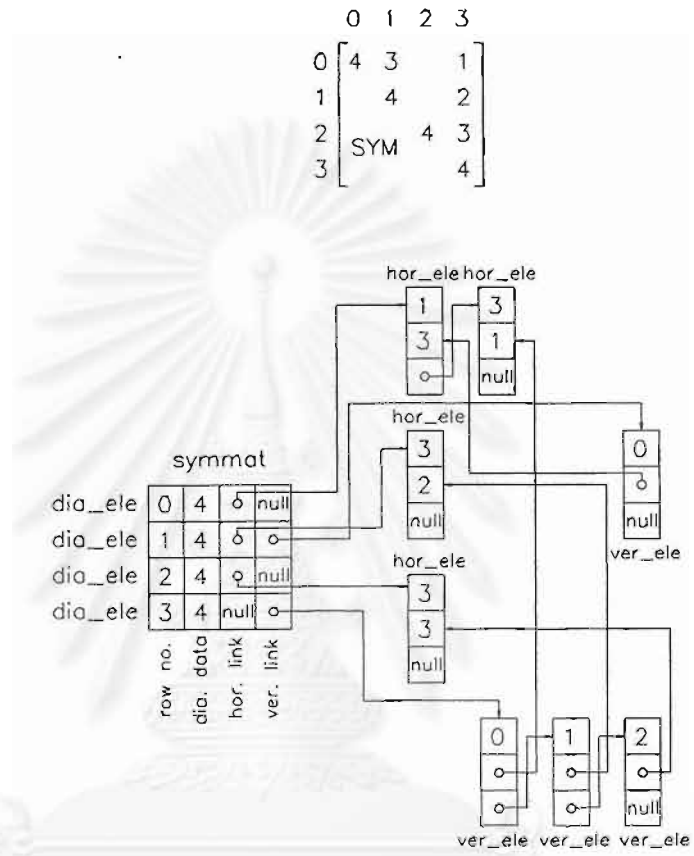


รูปที่ 3.2.2 โครงสร้างข้อมูลชนิดลิงคิลิสต์

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

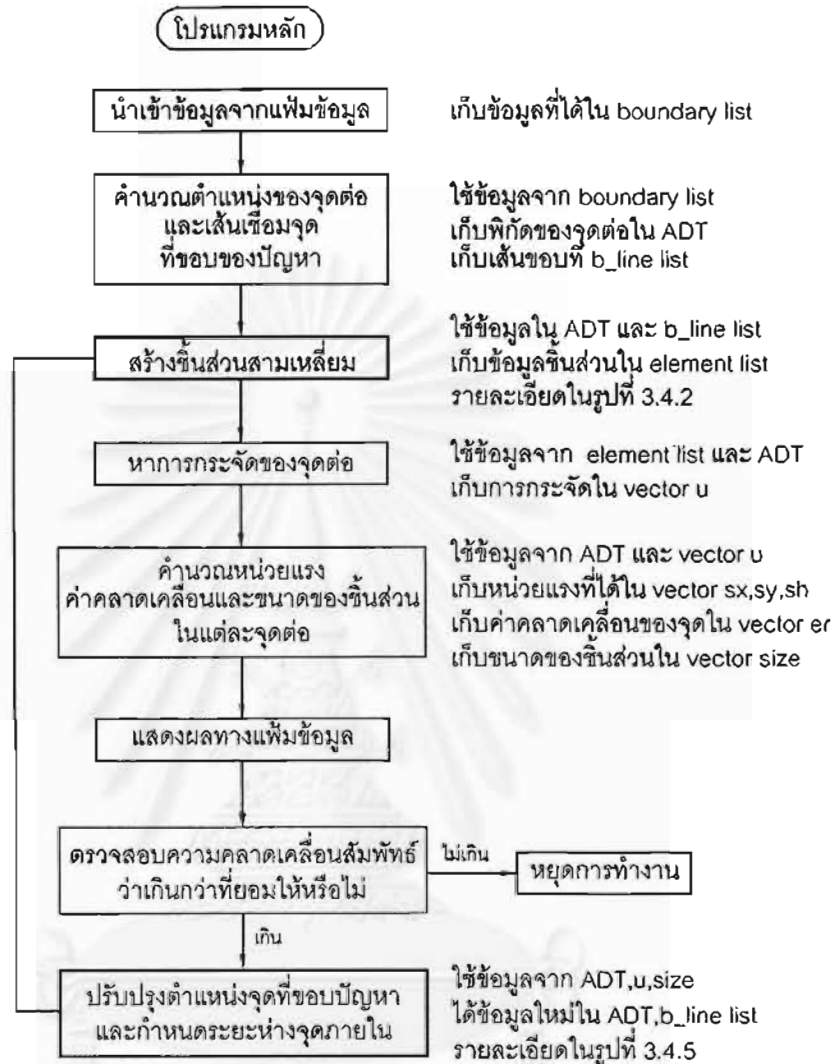


รูปที่ 3.2.3 การจัดเก็บข้อมูลในโครงสร้างข้อมูล ADT

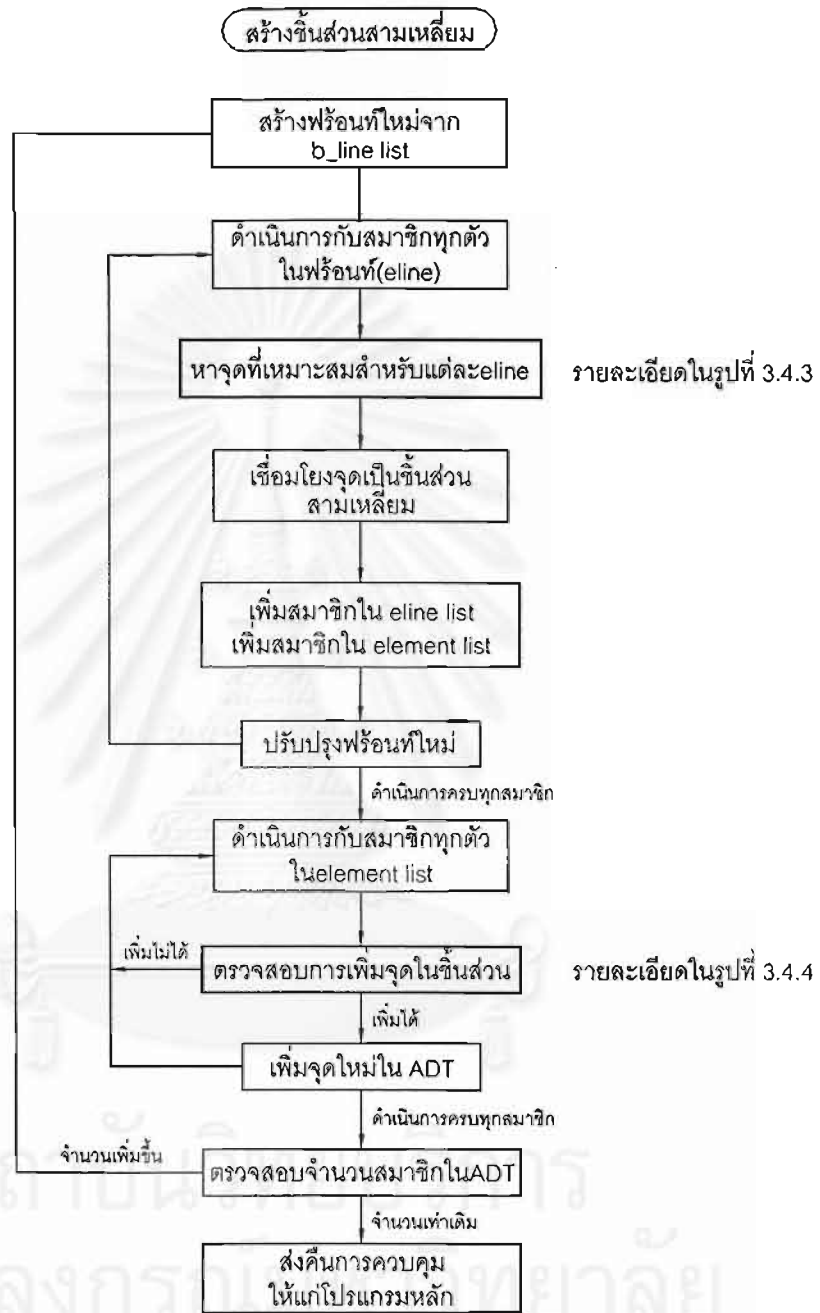


รูปที่ 3.3.1 การจัดเก็บข้อมูลของเมตริกซ์

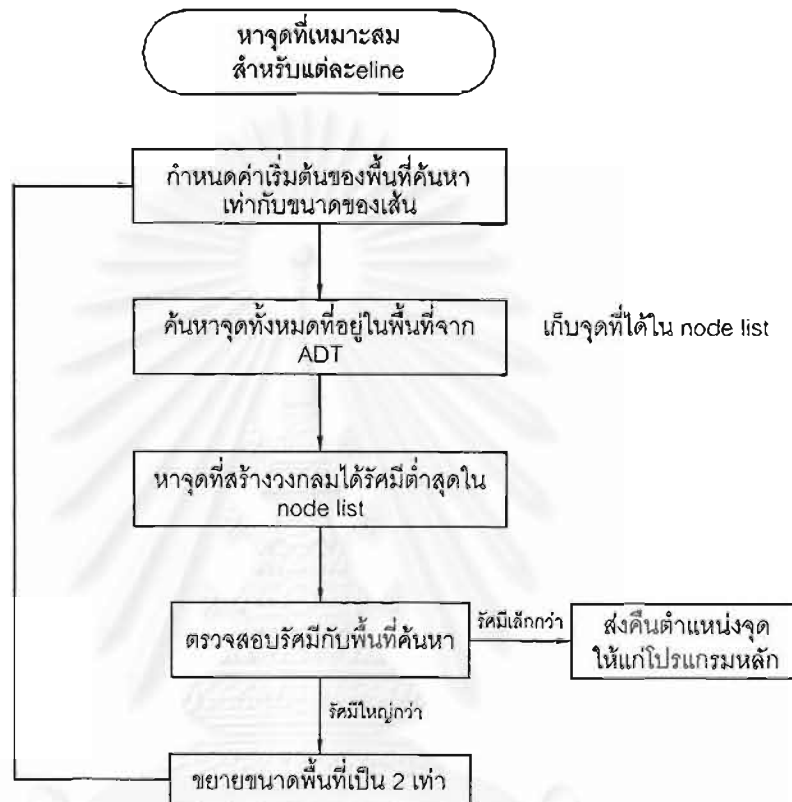
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



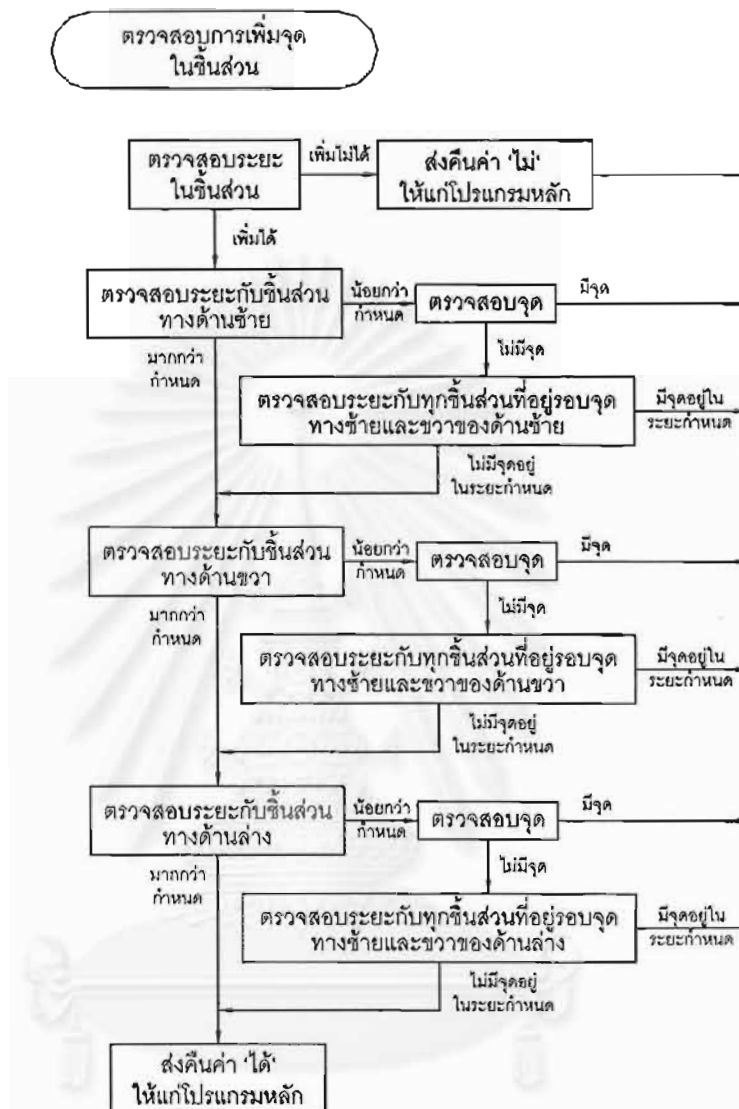
รูปที่ 3.4.1 โครงสร้างการทำงานของโปรแกรมหลัก



รูปที่ 3.4.2 โครงสร้างการทำงานของการสร้างชิ้นส่วนสามเหลี่ยม



รูปที่ 3.4.3 โครงสร้างการทำงานของการทำงานหาจุดที่เหมาะสมกับเส้นในการสร้างสามเหลี่ยม



รูปที่ 3.4.4 โครงสร้างการทำงานของ การตรวจสอบการเพิ่มจุด

ปรับปรุงตำแหน่งจุดที่ขอบปัญหา
และกำหนดระยะห่างจุดภายใน



รูปที่ 3.4.5 โครงสร้างการทำงานของ การปรับขนาดเส้นขอบและกำหนดระยะห่างของจุดภายใน

ตัวอย่างการวิเคราะห์ผล

4.1 ตัวอย่างที่ 1 ปัญหาแผ่นบางรับแรงดึง

ตัวอย่างที่ 1 เป็นการวิเคราะห์ปัญหาแผ่นบางขนาดเป็นอนันต์มีรูเจาะที่กึ่งกลางแผ่น และรับแรงดึงแผ่กระจายที่ปลายทั้ง 2 ด้าน เนื่องจากเป็นปัญหาที่มีขนาดเป็นอนันต์จึงลดรูปให้มีขนาดนับได้ดังแสดงในรูปที่ 4.1.1 คือ มีขนาดกว้าง 20 ซม. ยาว 40 ซม.หนา 0.1 ซม. มีรูเจาะที่กึ่งกลางขนาดเส้นผ่านศูนย์กลาง 4 ซม. รับแรงแผ่กระจายตามแนวยาวที่ปลายทั้งสองด้านมีขนาด 1000 กก.ต่อตร.ซม. วัสดุมีโมดูลัสยืดหยุ่น 2×10^6 กก.ต่อตร.ซม. อัตราส่วนปัวซองเท่ากับ 0.3 สามารถสร้างแบบจำลองสำหรับวิเคราะห์ผลโดยอาศัยหลักสมมาตรได้ดังแสดงในรูปด้านขวาของรูปที่ 4.1.1

ถ้ากำหนดความคลาดเคลื่อนสัมพัทธ์ที่ยอมให้เท่ากับ 0.01 จะได้ผลการวิเคราะห์ทั้งหมด 6 รอบ โดยมีโครงข่ายชิ้นส่วนในแต่ละรอบดังแสดงในรูปที่ 4.1.2 จะเห็นว่ามีการลู่เข้าค่อนข้างช้าทั้งนี้เนื่องจากในกระบวนการปรับขนาดในงานวิจัยนี้ใช้วิธีการเพิ่มจุดเข้าไปในบริเวณที่มีความคลาดเคลื่อนมาก โดยไม่มีการลดจำนวนจุดในบริเวณที่มีความคลาดเคลื่อนน้อยกว่าที่ยอมให้ เพราะฉะนั้นเพื่อเป็นการป้องกันการเพิ่มจุดมากเกินไปในบริเวณที่ไม่จำเป็น ในการปรับขนาดชิ้นส่วนในรอบถัดไป ถ้าค่าคลาดเคลื่อนในปัจจุบันมากกว่าค่าคลาดเคลื่อนที่ยอมให้เกินกว่า 2 เท่า จะกำหนดให้เป้าหมายของความคลาดเคลื่อนในรอบถัดไปมีขนาดเท่ากับค่าเฉลี่ยของค่าคลาดเคลื่อนในปัจจุบันและค่าคลาดเคลื่อนที่ยอมให้ ดังเช่น ค่าคลาดเคลื่อนสัมพัทธ์ในรอบที่ 1 มีค่า 0.059 เพราะฉะนั้นจะคำนวณขนาดของชิ้นส่วนโดยมีเป้าหมายของค่าคลาดเคลื่อนในรอบถัดไปเท่ากับ 0.035 และทำดังนี้ซ้ำจนกระทั่งค่าคลาดเคลื่อนที่ได้มีค่าไม่เกิน 2 เท่าของค่าคลาดเคลื่อนที่ยอมให้ ดังในรอบที่ 4 ค่าคลาดเคลื่อนเท่ากับ 0.0199 ก็จะต้องเป้าหมายการปรับขนาดชิ้นส่วนให้มีค่าเท่ากับค่าคลาดเคลื่อนที่ยอมให้ ในตัวอย่างนี้โครงข่ายสุดท้ายมีขนาดความคลาดเคลื่อนสัมพัทธ์เท่ากับ 0.0087 ซึ่งต่ำกว่าที่ยอมให้พอสมควร เนื่องจากการแบ่งชิ้นส่วนที่น้อยที่สุดคือแบ่งเป็นครึ่งหนึ่งของชิ้นส่วนไม่ได้มีการรวมกันแล้วแบ่งเฉลี่ย เช่นรวม 2 ชิ้นส่วนที่ติดกันแล้วแบ่งเป็น 3 ชิ้น ดังนั้นในชิ้นส่วนที่มีการปรับขนาดใหม่จะมีขนาดที่เล็กกว่าที่คำนวณเสมอ แต่ในขั้นตอนต้น ๆ ของการปรับขนาด ความคลาดเคลื่อนสัมพัทธ์ในแต่ละรอบก็ยังคงมีค่ามากกว่าความคลาดเคลื่อนสัมพัทธ์เป้าหมาย ทั้งนี้เพราะโครงข่ายของชิ้นส่วนยังมีขนาดหยาบ ทำให้ความคลาดเคลื่อนโดยประมาณที่คำนวณได้มีค่าน้อยกว่าค่าจริงมาก จึงคำนวณเป้าหมายของความคลาดเคลื่อนมากเกินไป แต่เมื่อโครงข่ายละเอียดมากขึ้นความคลาดเคลื่อนสัมพัทธ์ก็จะใกล้เคียงเป้าหมายมากขึ้น

สำหรับการกระจายของหน่วยแรงต่าง ๆ สามารถแสดงได้ในรูปที่ 4.1.3 - 4.1.5 ซึ่งจะมีความต่อเนื่องของเส้นคอนทัวร์ในทุกโครงข่ายชิ้นส่วนเพราะใช้ผลจากหน่วยแรงที่จุดต่อที่ได้มีการประมาณจากชิ้นส่วนโดยรอบเรียบร้อยแล้ว และจะพบว่าเส้นคอนทัวร์ของหน่วยแรงต่าง ๆ ที่บริเวณภายในมีลักษณะเรียบกว่าบริเวณขอบ ซึ่งเป็นปกติของ

ระเบียบวิธีไฟไนต์เอลิเมนต์ที่มีความคลาดเคลื่อนมากตามขอบของปัญหา แม้ว่าจะมีการประมาณจากตำแหน่งเกาส์จากหอย่อมชิ้นส่วนภายในแล้วก็ตาม แสดงให้เห็นว่าการประมาณค่าที่จุดต่อด้วยวิธีนี้ยังคงให้ค่าหน่วยแรงที่ขอบได้ไม่ดีมากนัก

การเปรียบเทียบกับผลเฉลยแม่นยำตรงจากการเปรียบเทียบหน่วยแรงที่เกิดตามแนว AB ในรูปที่ 4.1.1 ได้ผลการเปรียบเทียบในแต่ละโครงข่ายดังแสดงในรูปที่ 4.1.6 - 4.1.8 ซึ่งจะมีลักษณะลู่เข้าสู่ค่าตอบแม่นยำตรงมากขึ้นเมื่อโครงข่ายของชิ้นส่วนมีความละเอียดมากขึ้น และในโครงข่ายสุดท้ายผลเฉลยที่ได้มีความใกล้เคียงกับผลเฉลยแม่นยำตรงในระดับที่น่าพอใจโดยมีค่าคลาดเคลื่อนของหน่วยแรงในทิศทาง y ที่ตำแหน่ง A เท่ากับ 0.44% จะเห็นว่าหน่วยแรงที่ได้ไม่ลู่เข้าสู่ในลักษณะที่มีขอบเขตบนหรือล่างแต่จะสลับไปมาไม่แน่นอน เพราะเป็นผลจากการประมาณจากชิ้นส่วนโดยรอบไม่ใช่ผลจากระเบียบวิธีไฟไนต์เอลิเมนต์โดยตรง

สังเกตได้ว่าหน่วยแรงในทิศทาง x จะมีค่าลดลงมากกว่าผลเฉลยแม่นยำตรงอย่างเห็นได้ชัดหลังจากผ่านค่าสูงสุด และจะลู่เข้าสู่ค่าศูนย์ในบริเวณใกล้จุด B ทั้งนี้เกิดจากการจำลองแผ่นบางขนาดอนันต์ด้วยแผ่นบางที่มีขนาดนับได้ซึ่งจะมีหน่วยแรงในทิศทาง x เท่ากับ 0 ที่ขอบ ในขณะที่แผ่นบางขนาดอนันต์ไม่มีขอบ เมื่อค่าหน่วยแรงลดลงถึงค่า ๆ หนึ่งก็คงที่ไปตลอด ลักษณะเช่นนี้เกิดขึ้นกับหน่วยแรงในทิศทาง y เช่นกันโดยที่จุด B หน่วยแรงที่ได้จะมีค่าต่ำกว่าผลเฉลยแม่นยำตรงเล็กน้อยแต่ผลกระทบจะเกิดขึ้นในบริเวณแคบ ๆ ไม่มากเหมือนกับหน่วยแรงในทิศทาง x สำหรับหน่วยแรงเฉือนจะไม่สังเกตเห็นเพราะผลเฉลยมีค่าเท่ากันทั้งแผ่นบางขนาดอนันต์และแผ่นบางมีขนาด

การวิเคราะห์ปัญหาโดยกำหนดโครงข่ายเริ่มต้นอื่น ๆ การกำหนดโครงข่ายเริ่มต้นต้องมีความสอดคล้องกับความคลาดเคลื่อนยอมให้ในแต่ละปัญหา และมีความละเอียดเพียงพอให้โปรแกรมสามารถประมาณค่าที่จุดต่อต่าง ๆ ได้ ถ้ามีการกำหนดโครงข่ายที่หยาบเกินไปดังแสดงในรูปที่ 4.1.9 จะทำให้ไม่มีจุดต่อใด ๆ อยู่ภายใน จึงไม่สามารถประมาณค่าหน่วยแรงที่จุดต่อจากหอย่อมชิ้นส่วนภายในได้ ต้องประมาณค่าหน่วยแรงที่จุดต่อให้เท่ากับที่ตำแหน่งเกาส์ ผลก็คือความคลาดเคลื่อนที่เทียบจากตำแหน่งเกาส์มีค่าเป็น 0 เพราะฉะนั้นผลการวิเคราะห์จึงไม่ลู่เข้าเพราะไม่สามารถคำนวณหาความคลาดเคลื่อนได้ ในขณะที่การเพิ่มความละเอียดของโครงข่ายเริ่มต้น จะทำให้จำนวนรอบที่ใช้ในการวิเคราะห์ลดลง เพราะส่งผลโดยตรงต่อความคลาดเคลื่อนเริ่มต้น และเมื่อโครงข่ายมีความละเอียดมากขึ้นการประมาณค่าความคลาดเคลื่อนก็แม่นยำขึ้น ทำให้สามารถปรับขนาดชิ้นส่วนเข้าสู่เป้าหมายได้เร็ว แต่ถ้ากำหนดโครงข่ายเริ่มต้นละเอียดเกินไปดังแสดงในรูปที่ 4.1.10 โปรแกรมจะไม่สามารถจัดการให้มีการกระจายความคลาดเคลื่อนออกไปเท่า ๆ กันได้ เพราะไม่มีกระบวนการเพิ่มขนาดของชิ้นส่วน ส่งผลให้ความคลาดเคลื่อนโดยรวมไม่อยู่ในตำแหน่งเหมาะสมที่สุด (Optimal) คือ ความคลาดเคลื่อนรวมมีค่าน้อยในขณะที่ความคลาดเคลื่อนเฉพาะที่ในหลาย ๆ บริเวณยังคงมีค่าสูง โปรแกรมจึงหยุดการทำงานก่อนโดยไม่มีการลดขนาดของชิ้นส่วนในบริเวณที่น่าจะลดขนาดลงได้อีก ในรูปจะเห็นว่าที่บริเวณกลาง ๆ จะมีความหนาแน่นของชิ้นส่วนมากเพราะว่าโครงข่ายเริ่มต้นในตัวอย่างนี้มีค่าใกล้เคียงกับขนาดชิ้นส่วนที่ความคลาดเคลื่อนที่ยอมให้แต่ก็ยังไม่เพียงพอ ดังนั้นเมื่อมีการปรับขนาดจึงได้มีการแบ่งครึ่งชิ้นส่วนในบริเวณนี้เป็นจำนวนมากทำให้ได้ชิ้นส่วนที่มีความละเอียดมากเกินไป

ดังที่ได้กล่าวแล้วว่ากระบวนการปรับขนาดในงานวิจัยนี้จะตั้งเป้าหมายของการปรับขนาดไปที่ค่าเฉลี่ยระหว่างค่าคลาดเคลื่อนในปัจจุบันและค่าคลาดเคลื่อนยอมให้ เพื่อหลีกเลี่ยงการเพิ่มจุดต่อในบริเวณที่ไม่จำเป็น แต่ในตัวอย่างนี้จะทำการวิเคราะห์โดยปรับขนาดสู่ความคลาดเคลื่อนยอมให้โดยตรง เพื่อเปรียบเทียบกับผลที่ได้วิเคราะห์ไว้แต่แรก โครงข่ายที่ได้จากการวิเคราะห์โดยปรับขนาดโดยตรงสามารถแสดงได้ดังในรูปที่ 4.1.12 จะเห็นว่ามีการรู้เข้าของผลการวิเคราะห์ที่เร็วมาก แต่ในโครงข่ายสุดท้ายจะมีการกระจายของชิ้นส่วนที่ไม่ค่อยต่อเนื่องมากนัก ดังที่ได้กล่าวแล้วว่าการเพิ่มจำนวนชิ้นส่วนที่น้อยที่สุดคือการแบ่งครึ่งชิ้นส่วน เมื่อโครงข่ายมีขนาดของชิ้นส่วนที่ใกล้เคียงกับขนาดโครงข่ายที่ความคลาดเคลื่อนที่ยอมให้ในบริเวณกว้าง เมื่อมีการปรับขนาดจะทำให้ได้ชิ้นส่วนที่มีขนาดละเอียดเกินไปในบริเวณนั้น ๆ สำหรับการปรับขนาดโดยการตั้งเป้าหมายไปที่ค่าเฉลี่ยจะค่อย ๆ ปรับขนาดชิ้นส่วนให้มีขนาดเล็กลงพอประมาณจึงสามารถควบคุมการปรับขนาดในแต่ละบริเวณได้ดีกว่า แต่ถ้าสามารถควบคุมการปรับขนาดชิ้นส่วนเป็นบริเวณเฉพาะได้ โดยการจัดกลุ่มชิ้นส่วนใกล้เคียงที่มีความคลาดเคลื่อนเท่า ๆ กันแล้วสร้างจุดต่อใหม่ทั้งหมดโดยไม่ใช้จุดต่อเดิมเลย ก็จะสามารถปรับขนาดโดยตรงได้อย่างมีประสิทธิภาพ คือ ได้โครงข่ายชิ้นส่วนที่มีการกระจายขนาดอย่างต่อเนื่องเพราะสามารถเพิ่มหรือลดขนาดของชิ้นส่วนได้อย่างอิสระ และมีการรู้เข้าของผลการวิเคราะห์ที่รวดเร็วกว่าในงานวิจัยนี้มาก ผลของการรู้เข้าที่รวดเร็วของการปรับขนาดโดยตรงแสดงให้เห็นว่าหลักการในการปรับขนาดสามารถใช้ได้อย่างมีประสิทธิภาพ แต่ที่ไม่สามารถปรับขนาดเข้าสู่ค่าคลาดเคลื่อนที่ยอมให้ภายในรอบเดียวก็เพราะความคลาดเคลื่อนโดยประมาณที่คำนวณได้มีความคลาดเคลื่อนจากความคลาดเคลื่อนที่เกิดขึ้นจริง

ในตัวอย่างนี้ยังได้ทำการวิเคราะห์โดยการลดขนาดของความคลาดเคลื่อนที่ยอมให้ลงอีกเพื่อเปรียบเทียบกับผลที่ได้วิเคราะห์ไว้ โดยกำหนดความคลาดเคลื่อนที่ยอมให้เท่ากับ 0.005 จะได้ผลการวิเคราะห์ทั้งหมด 8 รอบโดยแสดงโครงข่ายบางส่วนที่ได้จากการวิเคราะห์ในรูปที่ 4.1.14 ผลการวิเคราะห์แสดงให้เห็นว่าถึงแม้ว่าผลที่ได้จะมีความคลาดเคลื่อนลดลงแต่ก็ส่งผลให้จำนวนจุดต่อและชิ้นส่วนเพิ่มขึ้นอีกมากเช่นกัน ในตัวอย่างนี้ลดความคลาดเคลื่อนสัมพัทธ์ลงครึ่งหนึ่ง แต่จำนวนจุดต่อและชิ้นส่วนเพิ่มขึ้นประมาณ 3.6 เท่า ทำให้ต้องใช้เวลาในการสร้างชิ้นส่วนและวิเคราะห์ผลเพิ่มขึ้นอีกมาก ดังนั้นในการกำหนดความคลาดเคลื่อนยอมให้ควรที่จะพิจารณาลักษณะของปัญหาให้ดี ในตัวอย่างนี้หน่วยแรงมีลักษณะค่อนข้างคงที่เป็นบริเวณกว้างทำให้ความคลาดเคลื่อนโดยรวมน้อยจึงสามารถกำหนดความคลาดเคลื่อนสัมพัทธ์ที่ยอมให้ได้ค่อนข้างต่ำคือ 0.005 แต่ถ้าเป็นปัญหาที่หน่วยแรงมีการเปลี่ยนแปลงตลอดทั้งพื้นที่ การกำหนดความคลาดเคลื่อนยอมให้ต่ำ ๆ อาจทำให้ปัญหามีขนาดใหญ่เกินไปจนกระทั่งผลการวิเคราะห์ซึ่งใช้วิธีการแก้สมการโดยการทำซ้ำ มีการสะสมความคลาดเคลื่อนมากจนผลที่ได้มีความคลาดเคลื่อนสะสมสูงกว่าความคลาดเคลื่อนที่ยอมให้โดยไม่รู้ตัว ดังนั้นควรกำหนดความคลาดเคลื่อนสัมพัทธ์ยอมให้อยู่ในช่วง 1-5 เปอร์เซ็นต์ ถ้าต้องการผลการวิเคราะห์ที่มีความแม่นยำมากกว่านี้ ควรใช้ชิ้นส่วนที่มีกำลังของพหุนามสูงขึ้น

ผลการวิเคราะห์ในโครงข่ายสุดท้ายจากการกำหนดเงื่อนไขในการวิเคราะห์ต่าง ๆ เทียบกับผลเฉลยแม่นยำตรงนำมาแสดงในรูปที่ 4.1.16 – 4.1.18

4.2 ตัวอย่างที่ 2 ปัญหาแผ่นวงกลมบางรับแรงกระทำเป็นจุด

ในตัวอย่างนี้เป็นปัญหาแผ่นวงกลมบางขนาดเส้นผ่านศูนย์กลาง 2 ซม. รับแรงกระทำเป็นจุด 2 แรงในแนวแกน y กระทำในทิศทางตรงกันข้าม ขนาดของแรงเท่ากับ 100 กก. วัสดุมีโมดูลัสยืดหยุ่น $2 \cdot 10^6$ กก.ต่อตร.ซม. และอัตราส่วนปัวซองเท่ากับ 0.3 ดังแสดงในรูปที่ 4.2.1 และโดยอาศัยคุณสมบัติด้านความสมมาตร สามารถจำลองปัญหาได้ดังในรูปขวาของรูปที่ 4.2.1 การเปรียบเทียบกับผลเฉลยแม่นยำตรงจะเปรียบเทียบหน่วยแรงในทิศทาง y ที่เกิดตามแนว AB ในรูปที่ 4.2.1 ผลการวิเคราะห์และเปรียบเทียบแสดงดังในรูปที่ 4.2.2 - 4.2.6 โดยกำหนดความคลาดเคลื่อนสัมพัทธ์เท่ากับ 0.25 ผลปรากฏว่าไม่ลู่เข้าสู่ค่าตอบ ทั้งนี้เนื่องจากในบริเวณที่แรงกระทำมีความคลาดเคลื่อนเป็นอนันต์ เมื่อปรับขนาดโดยปรับไปที่ค่าเฉลี่ยจึงมีการปรับขนาดเฉพาะในบริเวณนี้เท่านั้น โดยไม่มีการปรับขนาดในบริเวณอื่น ๆ แต่อย่างไรก็ตามผลเฉลยในส่วนอื่น ๆ ไม่ได้ขึ้น และเมื่อมีการปรับขนาดของชิ้นส่วนถึงค่า ๆ หนึ่งซึ่งขนาดของชิ้นส่วนมีขนาดเล็กมากเกินกว่าที่ได้กำหนดไว้ในโปรแกรมให้สามารถจัดการได้ โปรแกรมจะแจ้งข้อความเตือนและหยุดการทำงาน ในการวิเคราะห์ด้วยการปรับขนาดโดยตรงก็มีปัญหาในลักษณะเดียวกันเพราะที่บริเวณแรงกระทำมีความคลาดเคลื่อนมากกว่าในบริเวณอื่นมากจนทำให้การลดขนาดชิ้นส่วนในบริเวณอื่น ๆ ไม่มีผลต่อความคลาดเคลื่อนโดยรวม การแก้ปัญหาก็ทำได้โดยการจำลองแรงกระทำเป็นจุดให้มีลักษณะเป็นแรงกระทำแผ่กระจาย โดยกำหนดระยะแรงกระทำเท่ากับ 4 องศาจากแนวแรงเดิม ในที่นี้มีระยะเท่ากับ 0.0698 ซม. จากแนวแกน y ซึ่งอาจจะกำหนดให้มากขึ้นหรือน้อยลงกว่านี้ได้ขึ้นอยู่กับผู้ทำการวิเคราะห์ ที่กำหนดระยะขนาดนี้เพื่อความสะดวกในการป้อนข้อมูลให้ชิ้นส่วนมีขนาดไม่แตกต่างกันมากเกินไป ถ้าจะกำหนดระยะของแรงกระทำให้น้อยกว่านี้ก็ควรที่จะเพิ่มขอบเขตของปัญหาให้มากขึ้นและแบ่งขอบเขตย่อยในแต่ละขอบเขตให้มีความใกล้เคียงกันกับขอบเขตข้างเคียง โดยที่ขอบเขตทั้งหมดต้องไม่มากจนกระทั่งโครงข่ายเริ่มต้นมีความละเอียดมากเกินไปเมื่อเทียบกับความคลาดเคลื่อนยอมให้ผลการวิเคราะห์ปรากฏว่าการลู่เข้าในแต่ละรอบมีลักษณะปกติ เมื่อกำหนดความคลาดเคลื่อนสัมพัทธ์ยอมให้เท่ากับ 0.05 จะได้ผลการวิเคราะห์และผลการเปรียบเทียบกับผลเฉลยแม่นยำตรงดังแสดงในรูปที่ 4.2.8 - 4.2.12 ซึ่งมีความน่าพอใจว่าการวิเคราะห์ด้วยแรงกระทำเป็นจุดมาก โดยที่ตำแหน่ง A มีค่าคลาดเคลื่อน 1.15 % สำหรับรายละเอียดของโครงข่ายที่ได้จากการวิเคราะห์ทั้ง 2 แบบสามารถเปรียบเทียบกันได้ดังแสดงในรูปที่ 4.2.13 ผลการวิเคราะห์ที่น่าสนใจอีกอย่างในตัวอย่างนี้คือความคลาดเคลื่อนสัมพัทธ์ในโครงข่ายเริ่มต้นมีค่าต่ำกว่าในโครงข่ายในรอบที่ 2 ทั้งนี้มีสาเหตุมาจากโครงข่ายเริ่มต้นมีขนาดของชิ้นส่วนหยาบ จนค่าประมาณของความคลาดเคลื่อนสัมพัทธ์ที่คำนวณได้มีความคลาดเคลื่อนมาก จึงทำให้ความคลาดเคลื่อนสัมพัทธ์ที่คำนวณได้มีค่าต่ำกว่าความคลาดเคลื่อนในรอบที่ 2

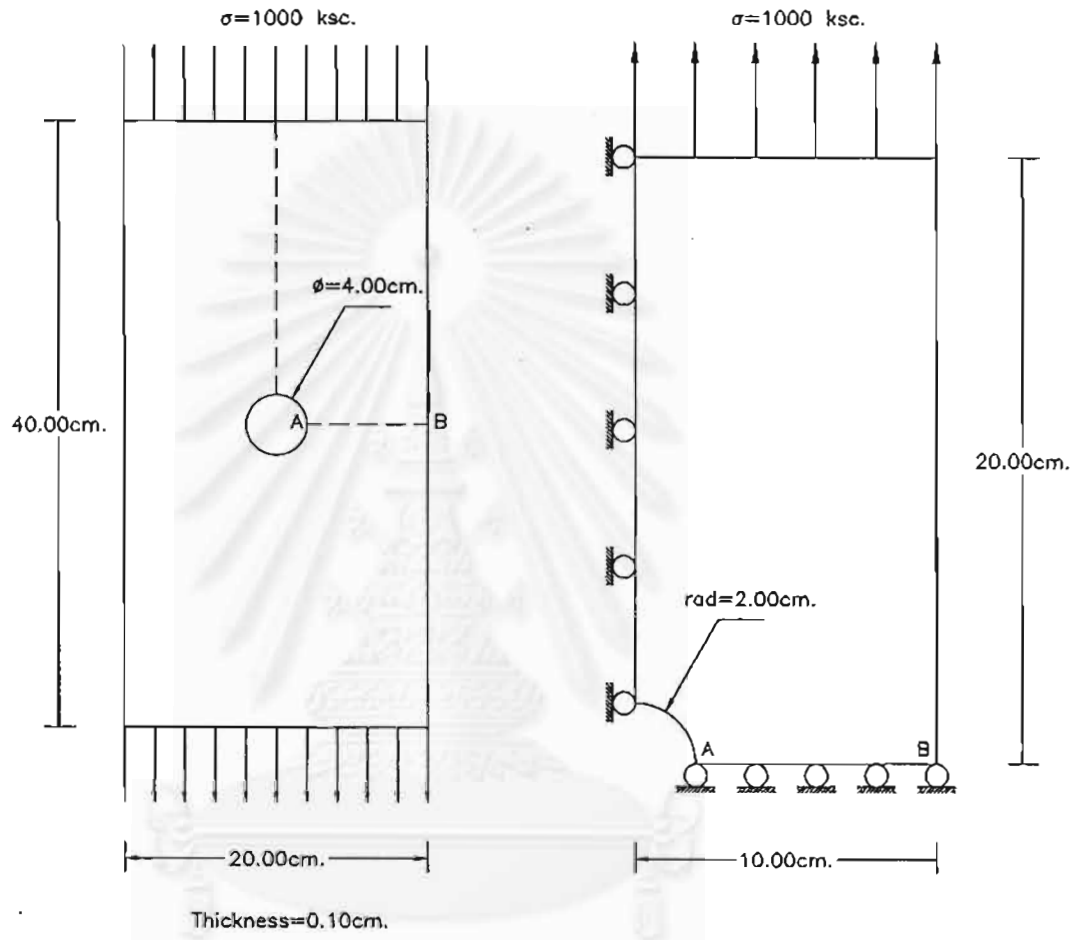
ในตัวอย่างนี้ได้ทำการวิเคราะห์ด้วยการปรับขนาดโดยตรงเพื่อเปรียบเทียบกับ โดยแสดงโครงข่ายที่ได้จากการวิเคราะห์ในรูปที่ 4.2.14 จะเห็นว่ามีการลู่เข้าเร็วมากเหมือนกับในตัวอย่างที่ 1 คือมีการวิเคราะห์เพียง 3 รอบเท่านั้น แต่จำนวนจุดต่อและชิ้นส่วนจะมากกว่าการปรับขนาดสู่ค่าเฉลี่ยมาก ถ้าสังเกตในโครงข่ายของชิ้นส่วนจะพบว่าในบริเวณกลางและล่างของพื้นที่มีขนาดและการกระจายของชิ้นส่วนไม่ต่างจากการปรับขนาดสู่ค่าเฉลี่ยมากนัก ในขณะที่ในบริเวณด้านบนของพื้นที่มีการลดขนาดของชิ้นส่วนมากเกินไป ที่เป็นเช่นนั้นเพราะว่าโครงข่ายเริ่มต้นในตัวอย่างนี้มีความคลาดเคลื่อนของความคลาดเคลื่อนสัมพัทธ์มาก ทำให้ความคลาดเคลื่อนสัมพัทธ์ในโครงข่ายที่ 2 มีค่าแตกต่างจากความคลาดเคลื่อนสัมพัทธ์ยอมให้พอสมควร เมื่อมีการปรับขนาดในรอบสุดท้ายจึงได้โครงข่ายที่ไม่ละเอียดมากเกินไปนักในบริเวณกลางและส่วนล่างของพื้นที่ ในขณะที่บริเวณส่วนบนมีความแม่นยำของการประมาณค่าคลาดเคลื่อนในโครงข่ายเริ่มต้นต่ำ ทำให้ขนาดของชิ้นส่วนในโครงข่ายที่ 2 มีขนาดหยาบเกินไป เมื่อปรับขนาดครั้ง

สุดท้ายจึงได้โครงข่ายที่ละเอียดเท่า ๆ กันเป็นบริเวณกว้าง ดังนั้นในการพัฒนาต่อไปควรจะศึกษาความคลาดเคลื่อนขนาดต่าง ๆ จากหลาย ๆ ตัวอย่างว่าความคลาดเคลื่อนขนาดใหญ่ที่การปรับขนาดโดยตรงจะให้ผลการปรับขนาดในรอบถัดไปมีความเหมาะสมสุด จากนั้นเมื่อนำมาใช้งานก็ให้ทำการปรับขนาดขึ้นส่วนจากโครงข่ายเริ่มต้นมายังโครงข่ายที่ได้ศึกษาแล้วว่าดีที่สุดก่อน จึงปรับขนาดให้สอดคล้องกับความคลาดเคลื่อนที่ยอมให้ที่ต้องการ ก็จะได้นาฬิกาของชิ้นส่วนที่เหมาะสม และ วิเคราะห์ปัญหาได้อย่างมีประสิทธิภาพ (ลดจำนวนรอบของการวิเคราะห์ให้น้อยที่สุด)

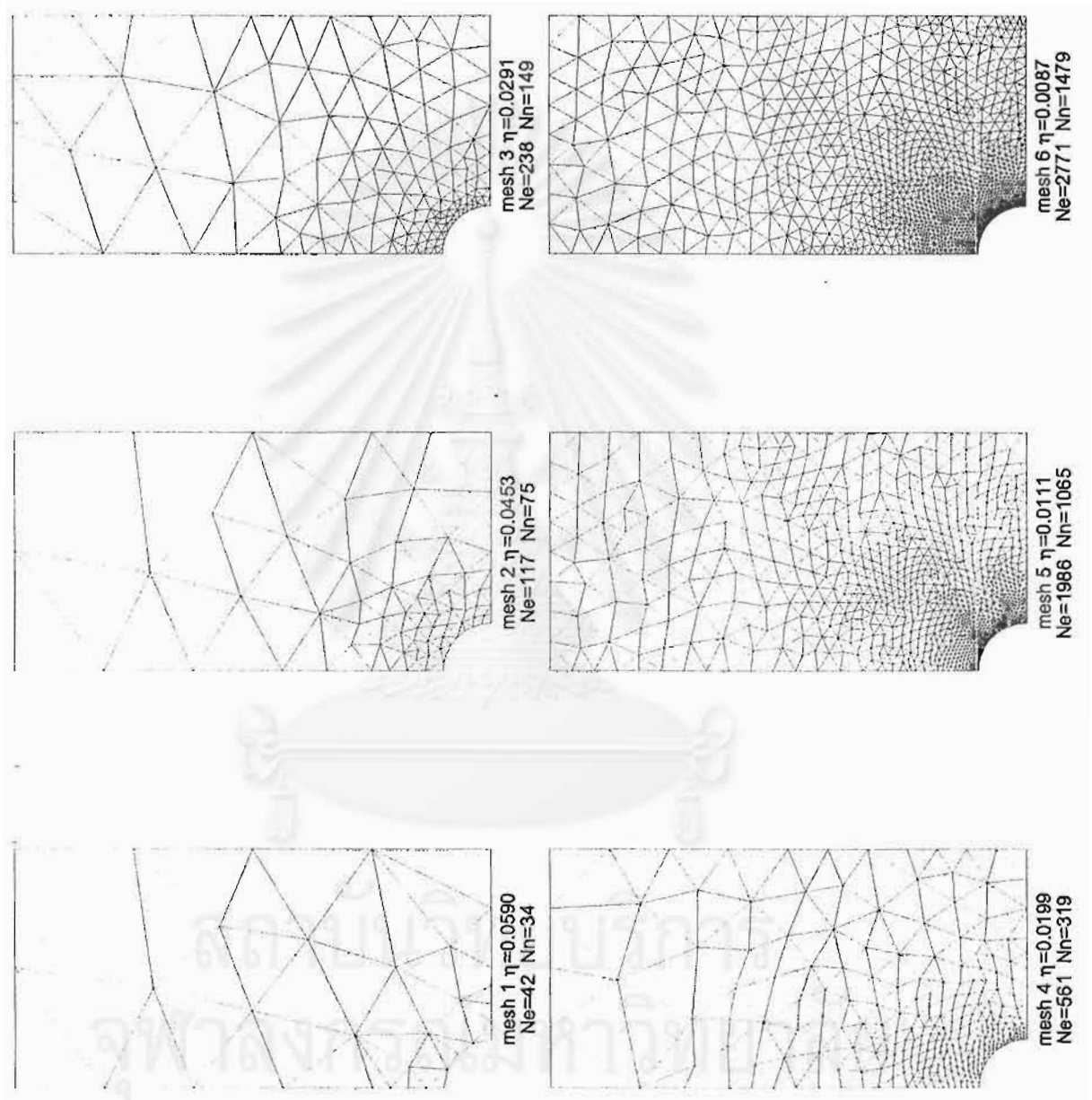
ในการจำลองแรงกระทำเป็นจุดให้เป็นแรงแผ่กระจายขนาดเท่ากันจะสามารถลดความคลาดเคลื่อนที่จุดที่แรงกระทำได้ แต่จะพบว่าที่บริเวณรอยต่อระหว่างจุดสิ้นสุดของแรงแผ่กระจายและชิ้นส่วนปกติที่ไม่มีแรงกระทำจะมีความคลาดเคลื่อนจำนวนหนึ่งซึ่งมากกว่าตำแหน่งโดยรอบ สังเกตได้จากโครงข่ายของชิ้นส่วนที่จะมีการปรับขนาดในบริเวณนี้มากกว่าบริเวณอื่น ๆ และการกระจายของหน่วยแรงจะมีการรวมของเส้นคอนทอร์ที่บริเวณนี้ ในตัวอย่างนี้จึงได้จำลองแรงกระทำให้แผ่กระจายเป็นรูปสามเหลี่ยมเพื่อลดความแตกต่างของแรงกระทำที่รอยต่อของจุดสิ้นสุดแรงกระทำ ดังมีโครงข่ายของการวิเคราะห์แสดงในรูปที่ 4.2.16 ซึ่งสามารถลดความคลาดเคลื่อนที่บริเวณรอยต่อได้ดีขึ้น แต่ผลการวิเคราะห์ในตำแหน่งที่ห่างออกมาไม่แตกต่างจากการวิเคราะห์ด้วยแรงแผ่กระจายสม่ำเสมอมากนัก สำหรับการปรับขนาดโดยตรงกับแรงแผ่กระจายเป็นรูปสามเหลี่ยมจะมีลักษณะคล้ายกับในแรงแผ่กระจายสม่ำเสมอ คือมีการลู่เข้าของการวิเคราะห์ที่ค่อนข้างเร็ว และมีจำนวนจุดต่อและชิ้นส่วนมากกว่าการปรับขนาดสุ่มค่าเฉลี่ยมาก โดยมีโครงข่ายที่ละเอียดเกินในบริเวณส่วนบนของพื้นที่ ผลการวิเคราะห์แสดงได้ในรูปที่ 4.2.18 และ 4.2.19

การลดความคลาดเคลื่อนสัมพัทธ์ที่ยอมให้ ในตัวอย่างนี้ได้ลดให้เหลือ 0.02 โดยวิเคราะห์แรงแผ่กระจายเป็นรูปสามเหลี่ยม ผลปรากฏว่ามีจำนวนจุดต่อพอ ๆ กับในตัวอย่างที่ 1 ที่กำหนดความคลาดเคลื่อนสัมพัทธ์ยอมให้เท่ากับ 0.005 เป็นการยืนยันว่าการกำหนดความคลาดเคลื่อนสัมพัทธ์ยอมให้ต้องคำนึงถึงลักษณะของปัญหาให้ดี ถ้าปัญหามีลักษณะซับซ้อนอาจจะวิเคราะห์คร่าว ๆ ที่ความคลาดเคลื่อนยอมให้สูง ๆ ก่อน จากนั้นจึงวิเคราะห์โดยละเอียดด้วยการลดความคลาดเคลื่อนที่ยอมให้ และอาจเพิ่มขอบเขตในบริเวณที่ต้องพิจารณาเป็นพิเศษเพื่อเพิ่มอัตราการลู่เข้าของการวิเคราะห์ได้ ผลการวิเคราะห์ด้วยการลดความคลาดเคลื่อนยอมให้แสดงในรูปที่ 4.2.20 – 4.2.21 เมื่อวิเคราะห์ด้วยการปรับขนาดโดยตรงที่ความคลาดเคลื่อนยอมให้เดียวกันพบว่า ความคลาดเคลื่อนยอมให้มีขนาดน้อยเกินไปในขณะที่ความคลาดเคลื่อนโดยประมาณในโครงข่ายเริ่มต้นที่บริเวณส่วนบนของพื้นที่มีความแม่นยำต่ำมาก ๆ เมื่อมีการปรับขนาดจึงทำให้โครงข่ายมีความแตกต่างของชิ้นส่วนมากในบริเวณใกล้เคียงกัน การสร้างชิ้นส่วนในรอบถัดไปจึงมีปัญหาไม่สามารถสร้างได้ จึงต้องมีการแบ่งขอบเขตเริ่มต้นให้มีขนาดของขอบเขตย่อยเล็กลง ในตัวอย่างนี้แบ่งเป็นครึ่งหนึ่งของข้อมูลเดิม ได้ผลการวิเคราะห์ 3 รอบดังแสดงในรูปที่ 4.2.22 – 4.2.28 แต่เนื่องจากการเพิ่มความละเอียดให้กับโครงข่ายเริ่มต้นทำให้โครงข่ายที่ 2 มีโครงข่ายที่ใกล้เคียงกับโครงข่ายสุดท้ายมากขึ้น เมื่อปรับขนาดครั้งสุดท้ายจึงได้โครงข่ายที่มีความละเอียดมากกระจายอยู่ทั่วทั้งบริเวณ หากเปรียบเทียบกับ การปรับขนาดสุ่มค่าเฉลี่ยจะมีจำนวนจุดต่อมากถึง 1.6 เท่าในขณะที่ความคลาดเคลื่อนที่ยอมให้มีขนาดเท่า ๆ กัน

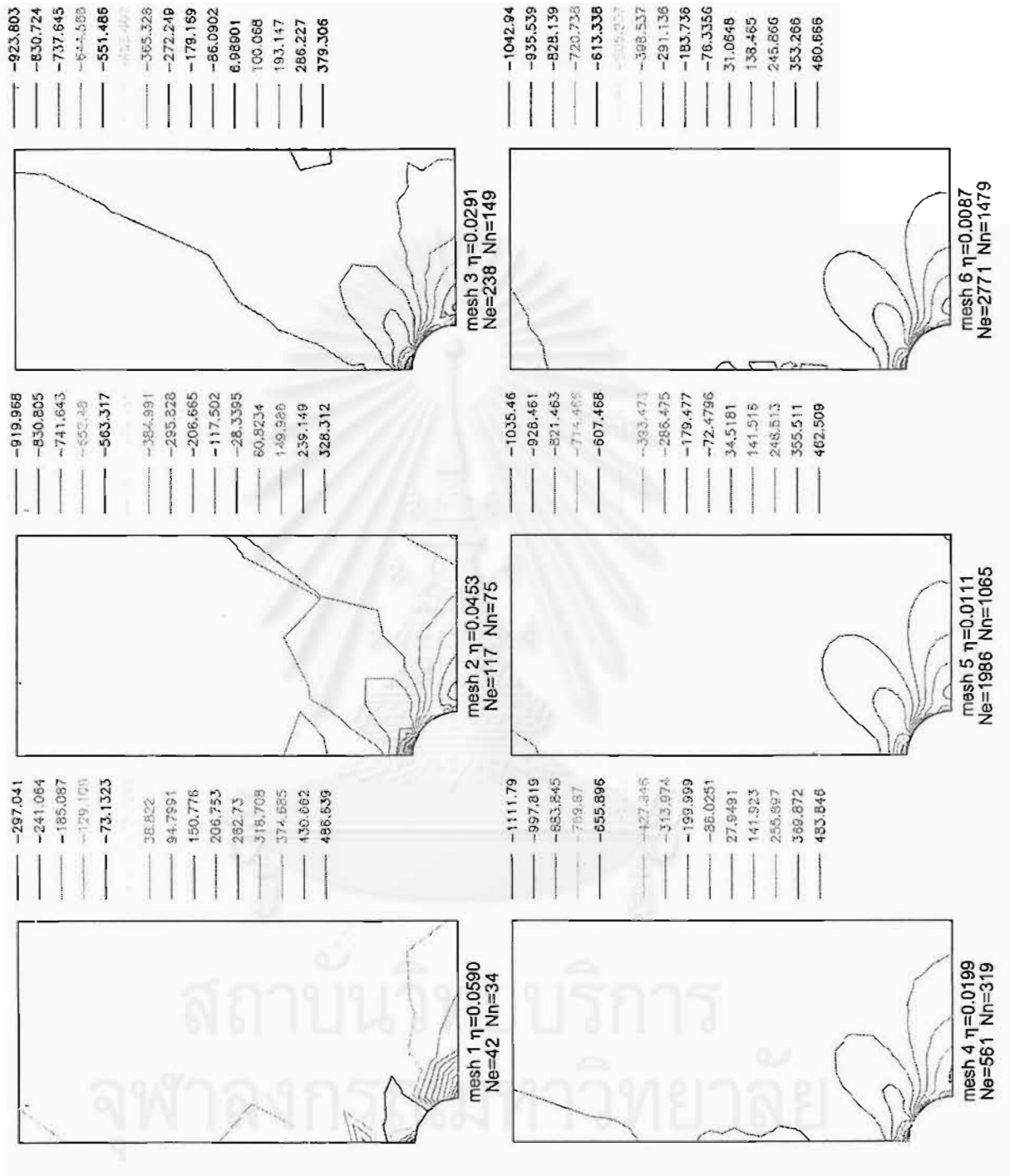
ผลการวิเคราะห์ในโครงข่ายสุดท้ายเทียบกับผลเฉลยแม่นยำตรงเมื่อกำหนดเงื่อนไขการวิเคราะห์ต่าง ๆ สามารถแสดงได้ในรูปที่ 4.1.24



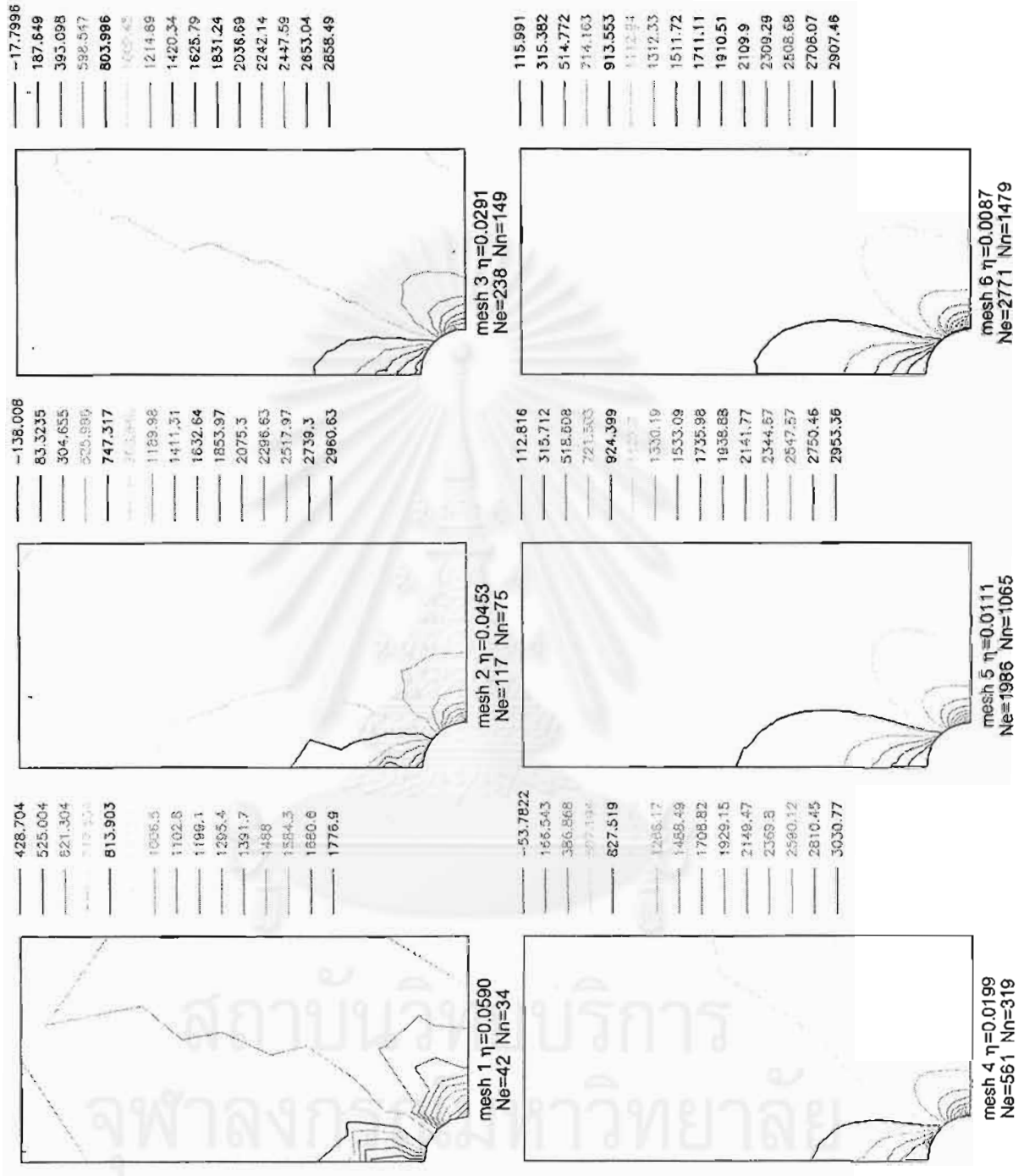
รูปที่ 4.1.1 ปัญหาแผ่นบางรับแรงดึงและแบบจำลองการวิเคราะห์



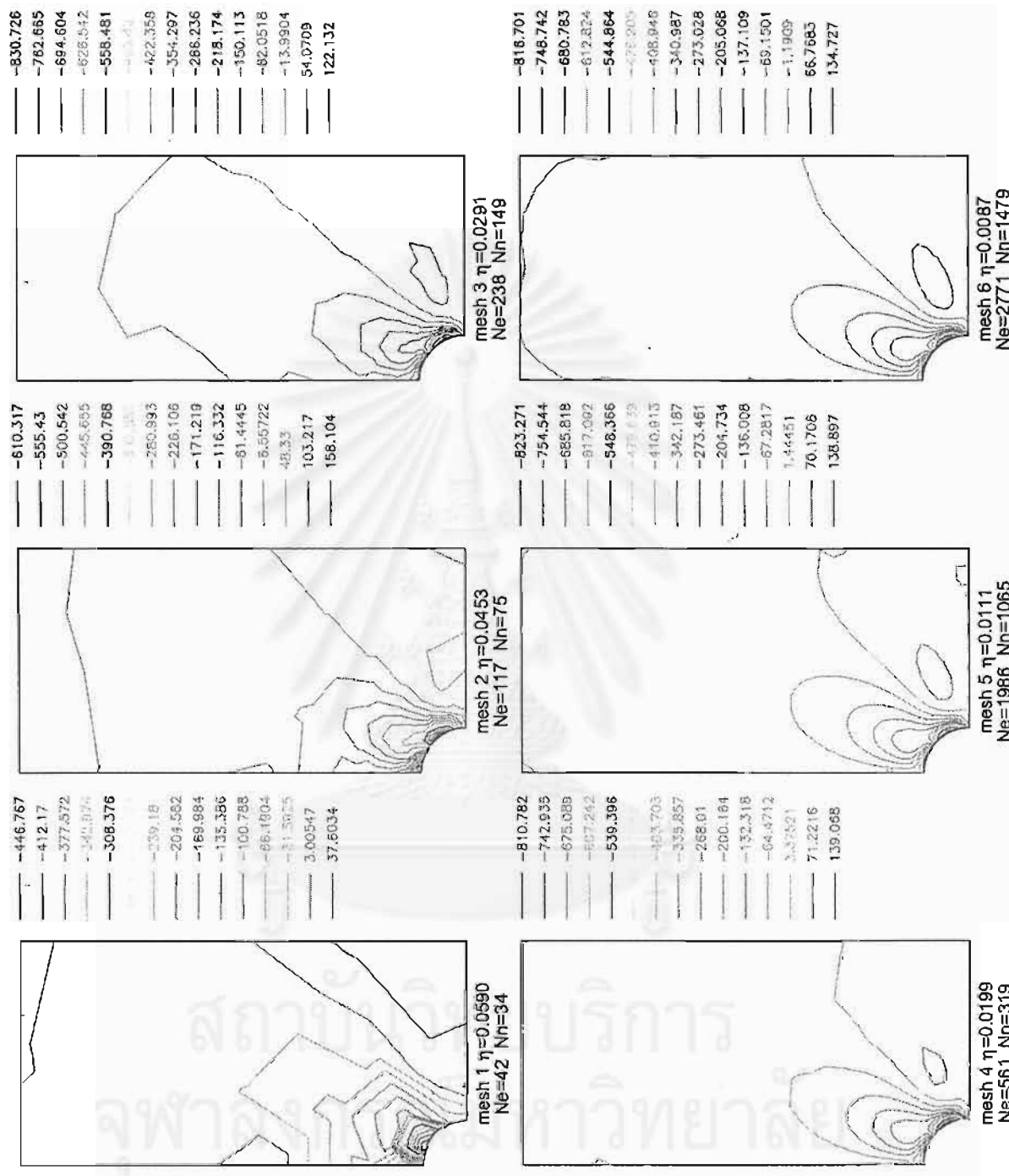
รูปที่ 4.1.2 โครงข่ายชิ้นส่วนสามเหลี่ยมในปัญหาแผ่นบางรับแรงดึง



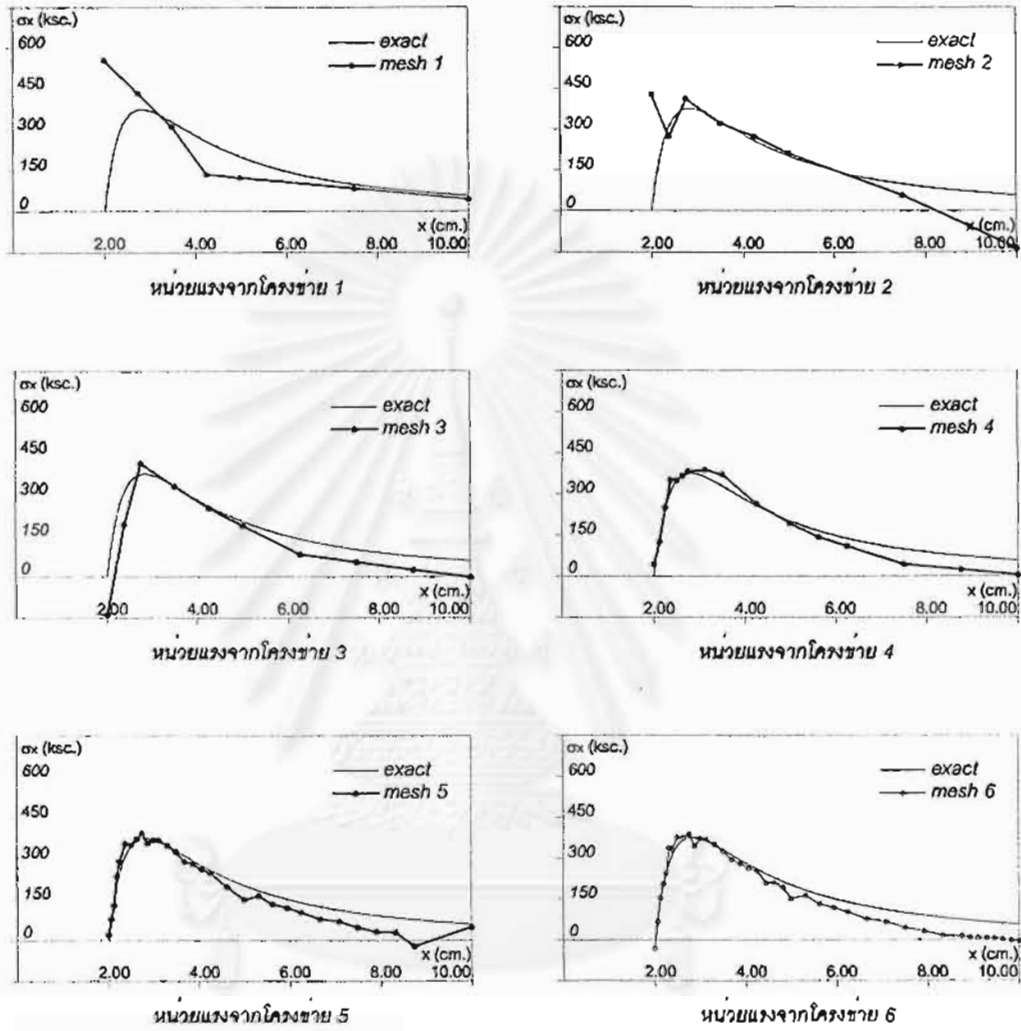
รูปที่ 4.1.3 การกระจายของหน่วยแรงในทิศทาง x ในปัญหาแผ่นบางรับแรงดึง



รูปที่ 4.1.4 การกระจายของหน่วยแรงในทิศทาง y ในปัญหาแผ่นบางรับแรงดึง

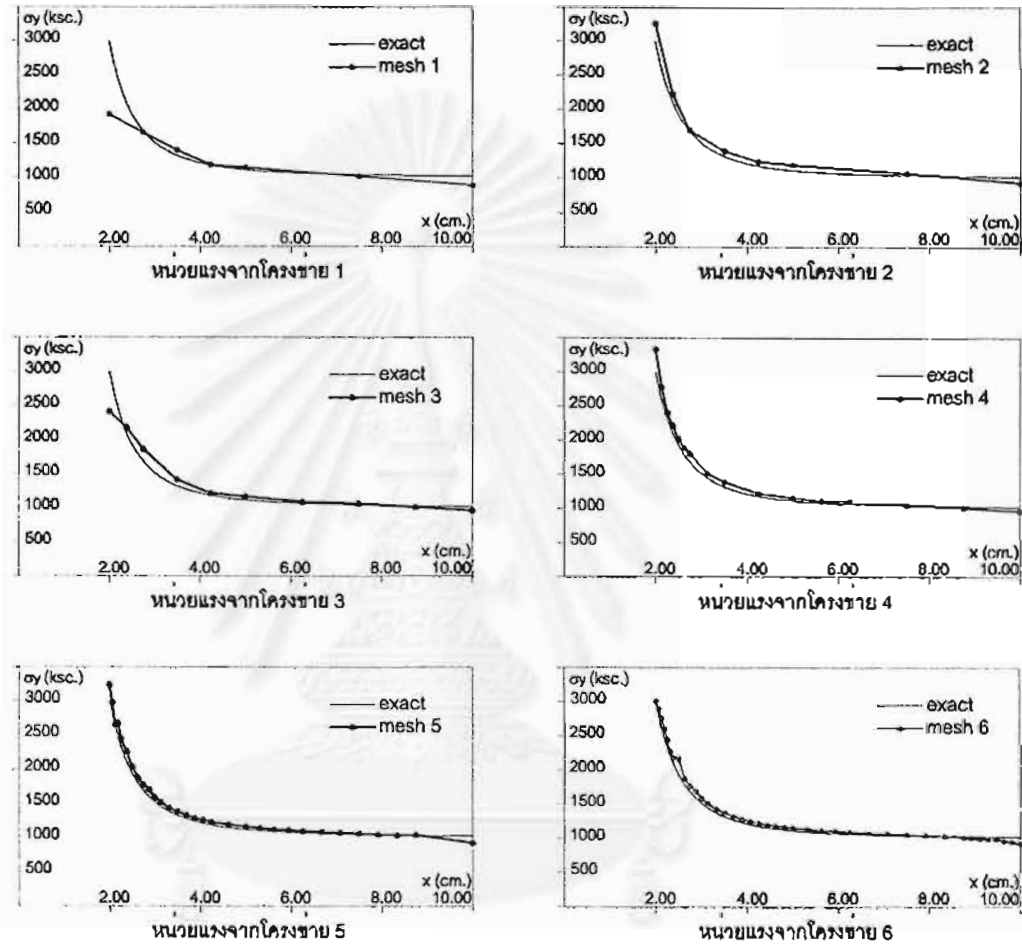


รูปที่ 4.1.5 การกระจายของหน่วยแรงเฉือนในปัญหาผ่านบางกับแรงดึง

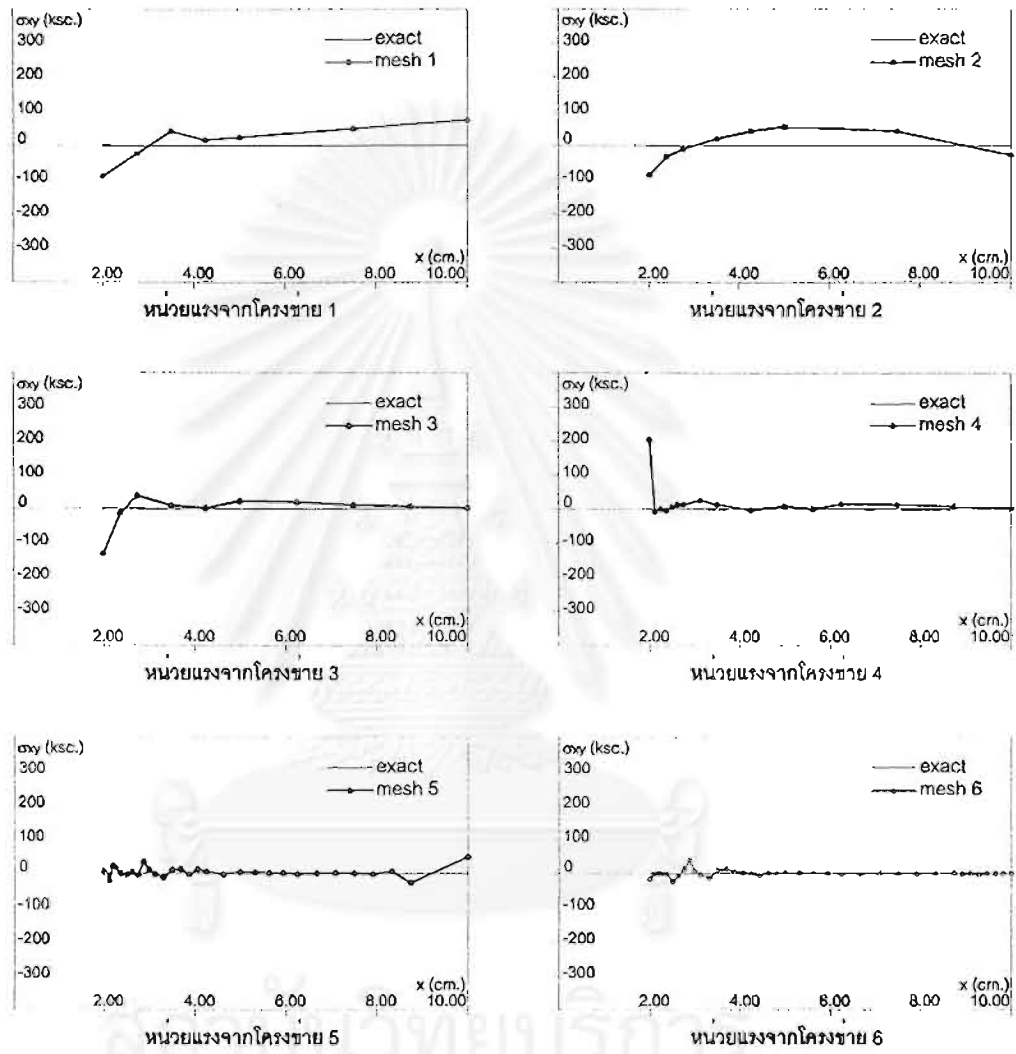


สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

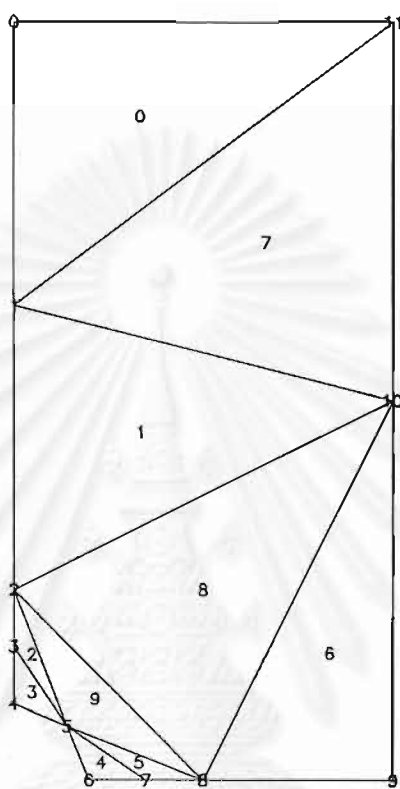
รูปที่ 4.1.6 กราฟแสดงค่านหน่วยแรงในทิศทาง x ตามแนว AB
ในปัญหาแผ่นบางรับแรงดึง



รูปที่ 4.1.7 กราฟแสดงค่าหน่วยแรงในทิศทาง y ตามแนว AB
ในปัญหาแผ่นบางรับแรงดึง

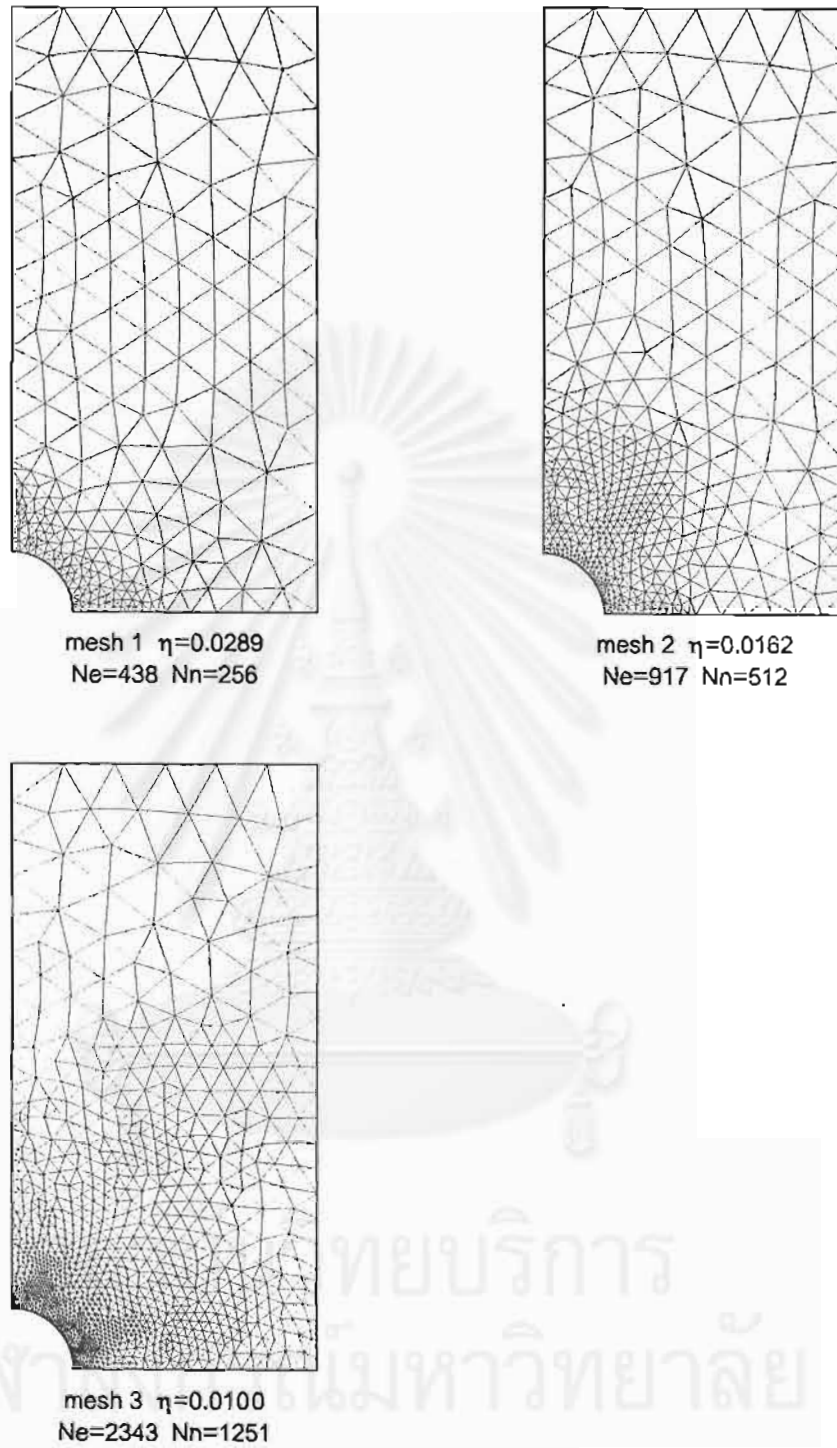


รูปที่ 4.1.8 กราฟแสดงค่าหน่วยแรงเฉือนตามแนว AB
ในปัญหาแผ่นบางรับแรงดิ่ง

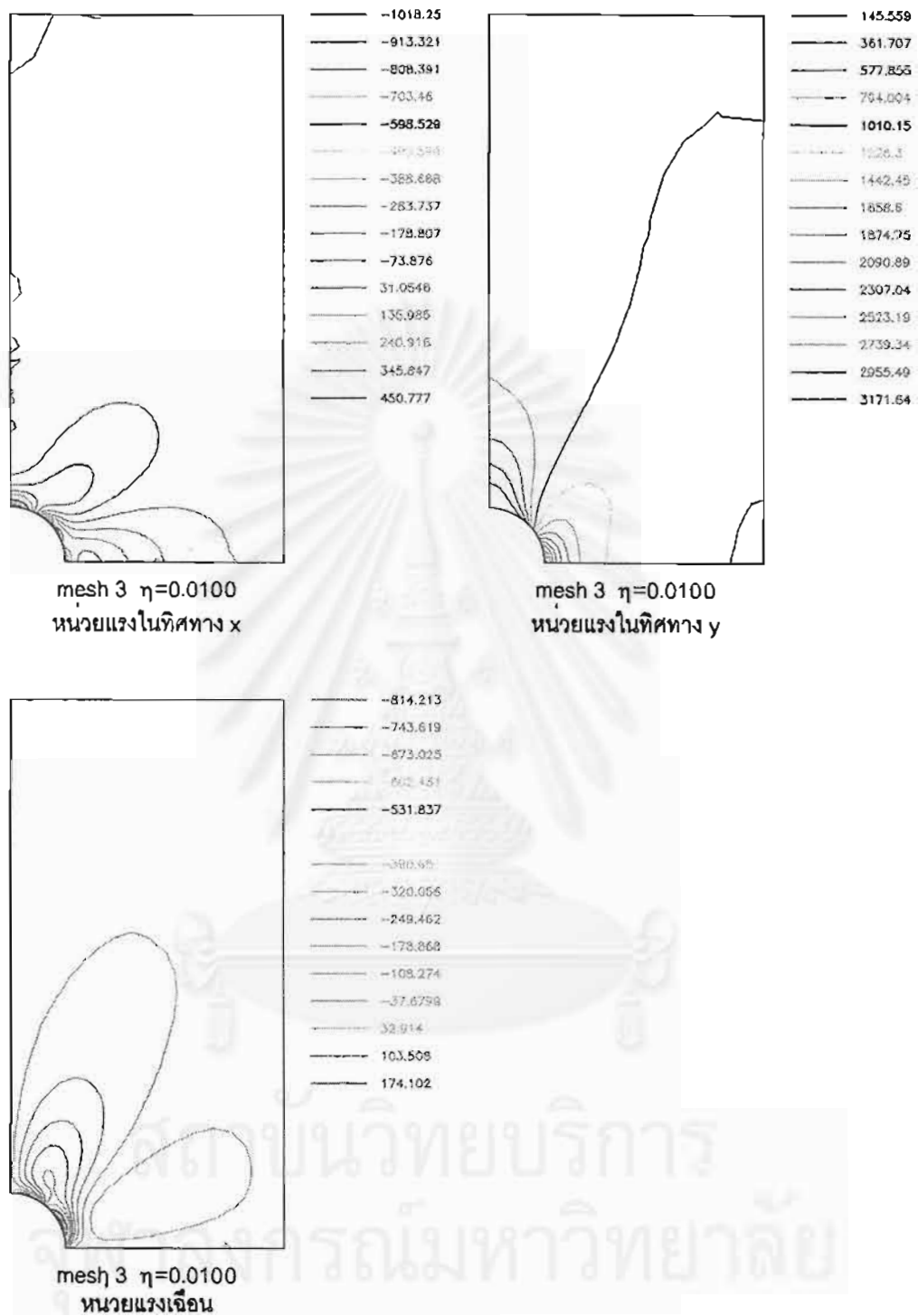


รูปที่ 4.1.9 โครงข่ายชิ้นส่วนสามเหลี่ยมจากโครงข่ายเริ่มต้นที่หยาบเกินไป
ในปัญหาแผนบางรับแรงดึง

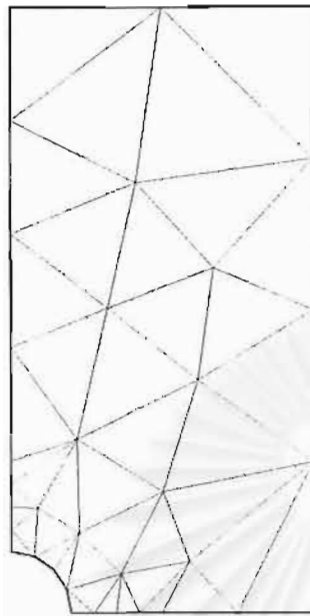
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



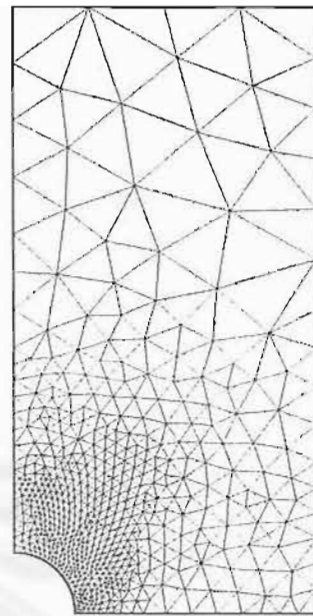
รูปที่ 4.1.10 โครงข่ายชิ้นส่วนสามเหลี่ยมจากโครงข่ายเริ่มต้นที่ละเอียดเกินไป



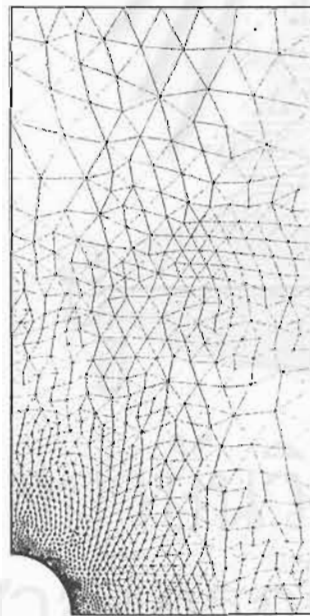
รูปที่ 4.1.11 การกระจายของหน่วยแรงในโครงข่ายสุดท้าย
จากการกำหนดโครงข่ายเริ่มต้นละเอียดเกินไป



mesh 1 $\eta=0.0590$
Ne=42 Nn=34

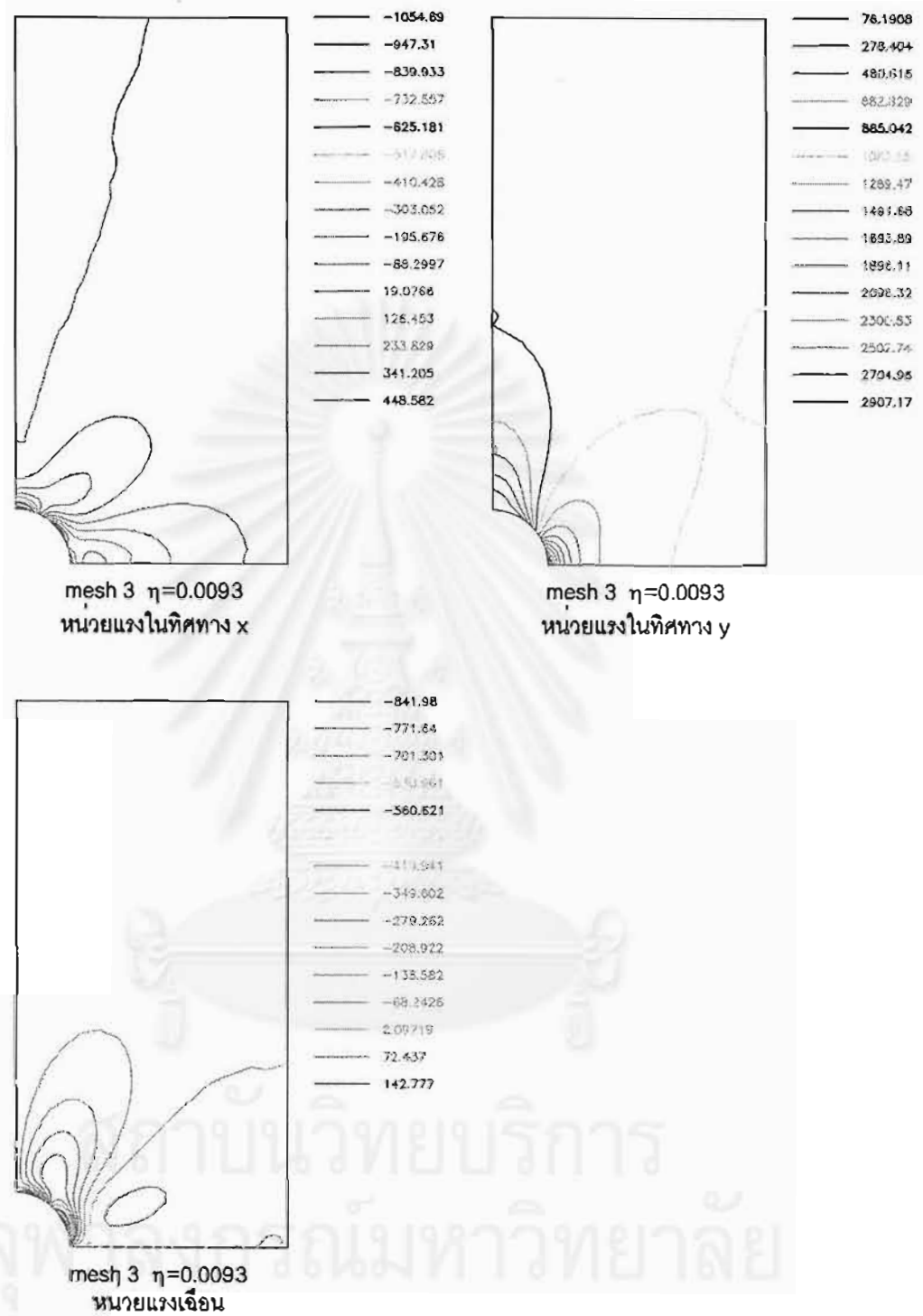


mesh 2 $\eta=0.0149$
Ne=1173 Nn=631

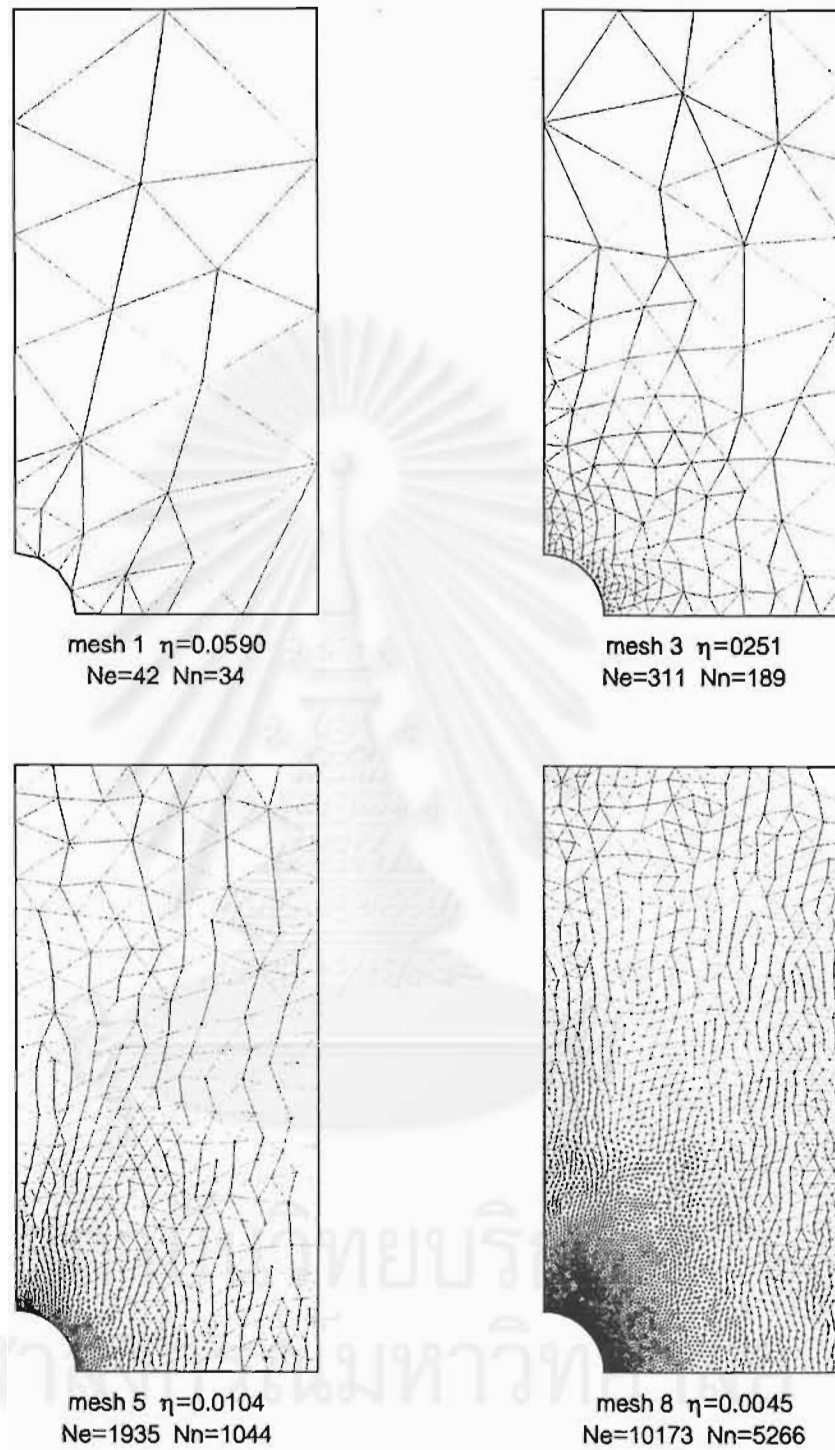


mesh 3 $\eta=0.0093$
Ne=2446 Nn=1310

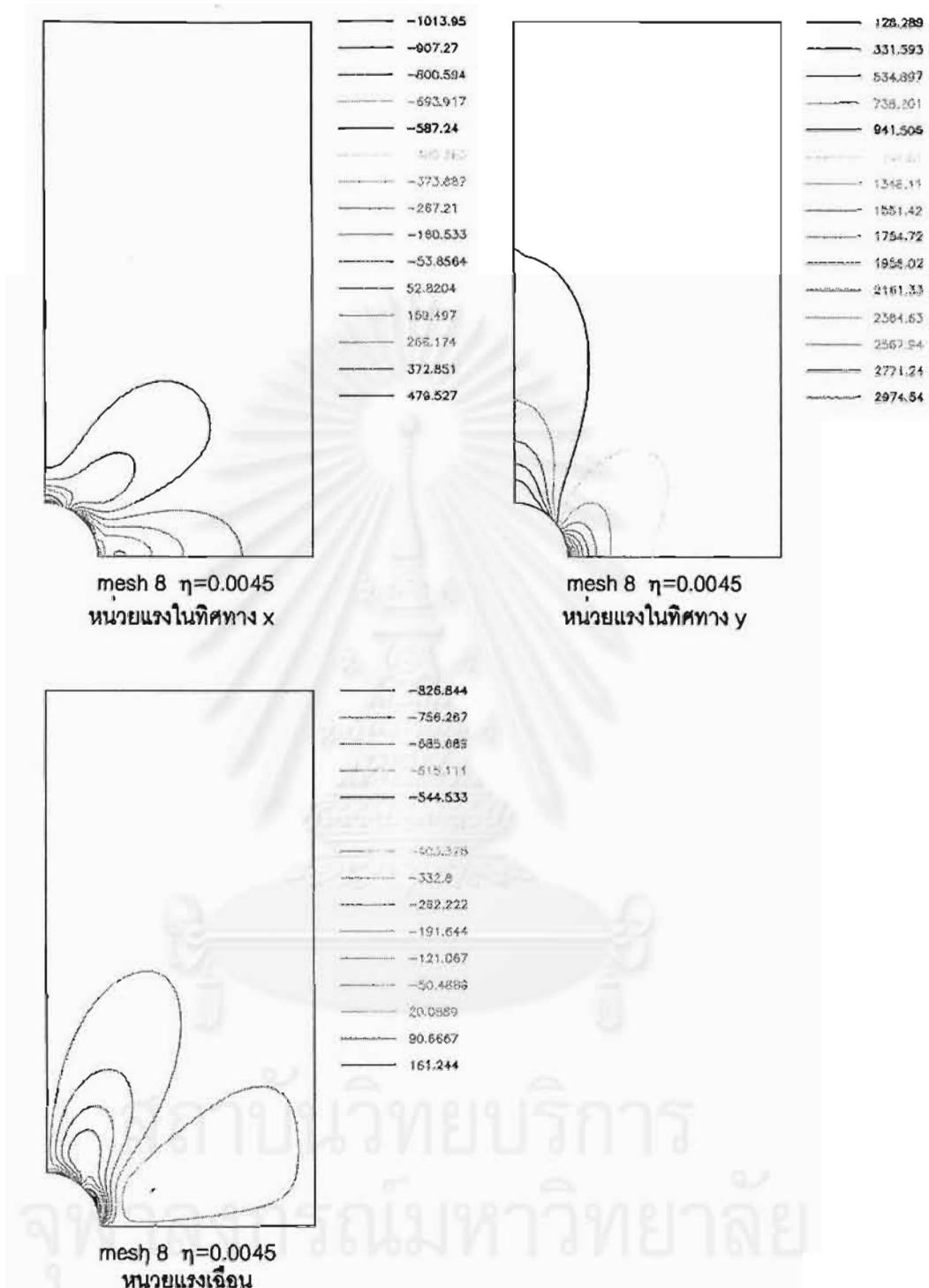
รูปที่ 4.1.12 โครงข่ายชิ้นส่วนสามเหลี่ยมที่ได้จากการปรับขนาดโดยตรง



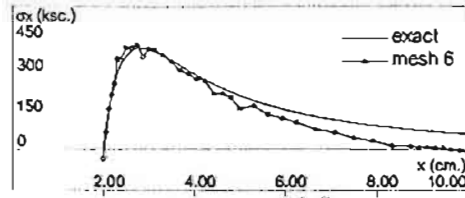
รูปที่ 4.1.13 การกระจายของหน่วยแรงในโครงข่ายสุดท้าย
จากการปรับขนาดโดยตรง



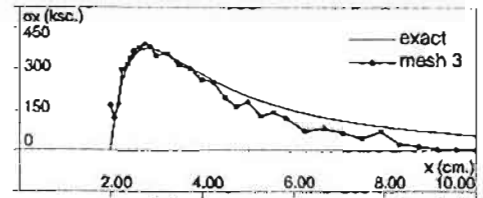
รูปที่ 4.1.14 โครงข่ายชิ้นส่วนสามเหลี่ยม
จากการลดความคลาดเคลื่อนยอมให้



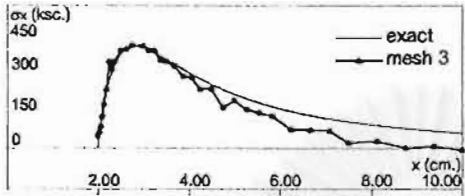
รูปที่ 4.1.15 การกระจายของหน่วยแรงในโครงข่ายสุดท้าย จากการลดความคลาดเคลื่อนยอมให้



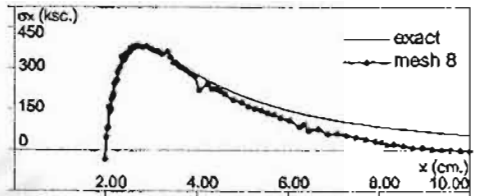
1.) ผลการวิเคราะห์เริ่มต้น



2.) ผลการวิเคราะห์เมื่อเพิ่มความละเอียดของโครงข่ายเริ่มต้น

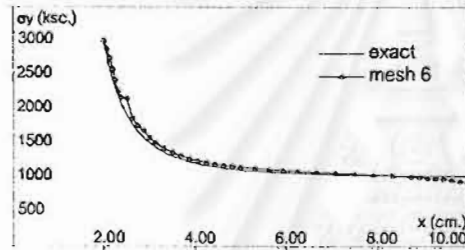


3.) ผลการวิเคราะห์เมื่อตั้งเป้าหมายสุดท้ายอัตโนมัติ

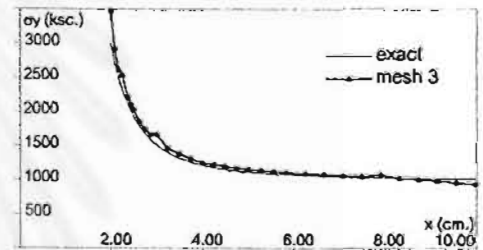


4.) ผลการวิเคราะห์เมื่อลดความคลาดเคลื่อนสัมพัทธ์อัตโนมัติ

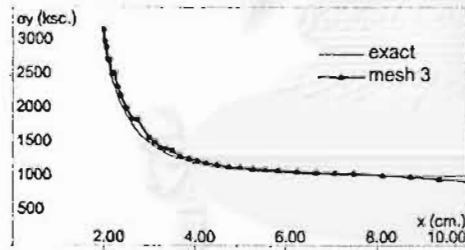
รูปที่ 4.1.16 เปรียบเทียบหน่วยแรงในทิศทาง x , ในโครงข่ายสุดท้ายจากการวิเคราะห์ด้วยข้อกำหนดต่าง ๆ



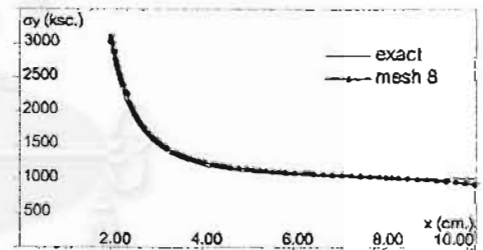
1.) ผลการวิเคราะห์เริ่มต้น



2.) ผลการวิเคราะห์เมื่อเพิ่มความละเอียดของโครงข่ายเริ่มต้น

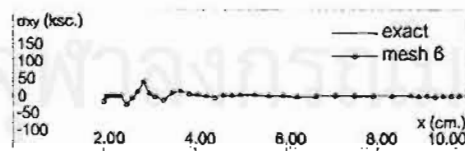


3.) ผลการวิเคราะห์เมื่อตั้งเป้าหมายสุดท้ายอัตโนมัติ

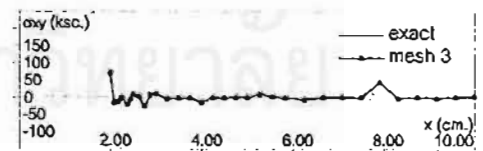


4.) ผลการวิเคราะห์เมื่อลดความคลาดเคลื่อนสัมพัทธ์อัตโนมัติ

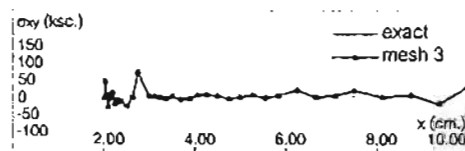
รูปที่ 4.1.17 เปรียบเทียบหน่วยแรงในทิศทาง y , ในโครงข่ายสุดท้ายจากการวิเคราะห์ด้วยข้อกำหนดต่าง ๆ



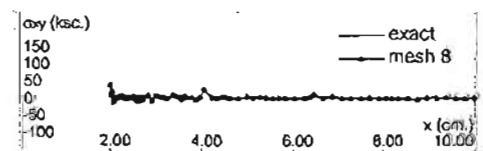
1.) ผลการวิเคราะห์เริ่มต้น



2.) ผลการวิเคราะห์เมื่อเพิ่มความละเอียดของโครงข่ายเริ่มต้น

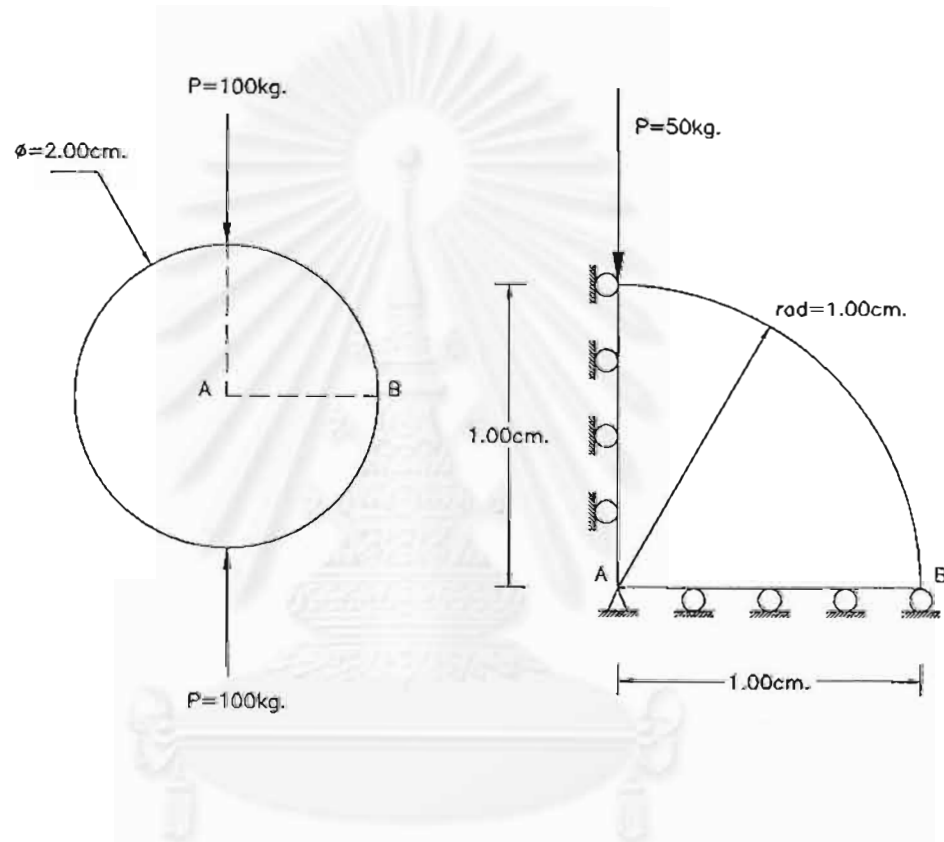


3.) ผลการวิเคราะห์เมื่อตั้งเป้าหมายสุดท้ายอัตโนมัติ

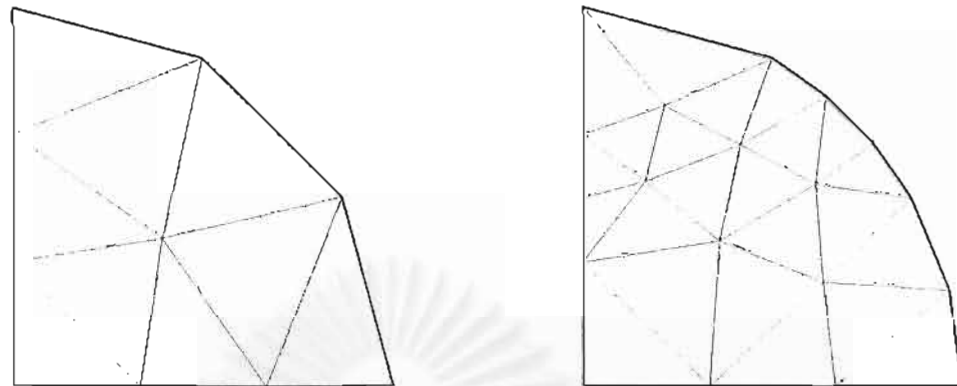


4.) ผลการวิเคราะห์เมื่อลดความคลาดเคลื่อนสัมพัทธ์อัตโนมัติ

รูปที่ 4.1.18 เปรียบเทียบหน่วยแรงเฉือนในโครงข่ายสุดท้ายจากการวิเคราะห์ด้วยข้อกำหนดต่าง ๆ

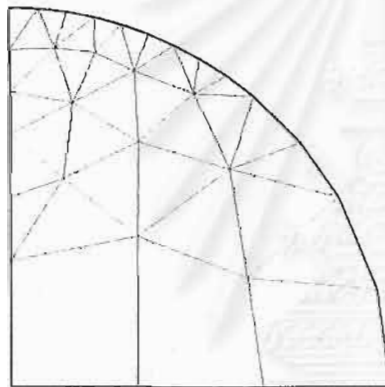


รูปที่ 4.2.1 ปัญหาแผ่นวงกลมบางรับแรงกระทำเป็นจุด และแบบจำลองการวิเคราะห์

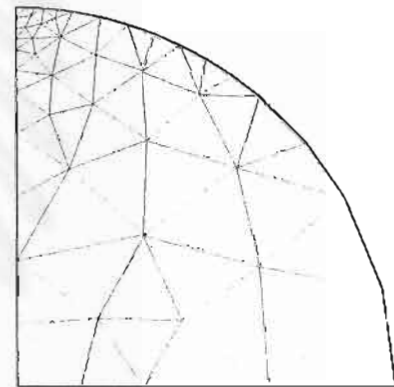


mesh 1 $\eta=0.3587$
Ne=9 Nn=10

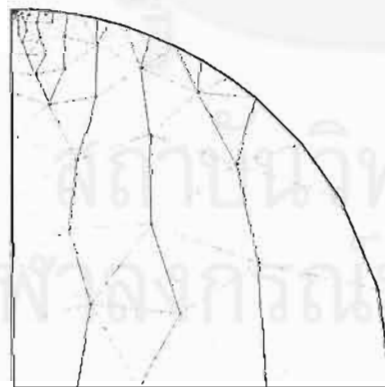
mesh 2 $\eta=0.4175$
Ne=23 Nn=19



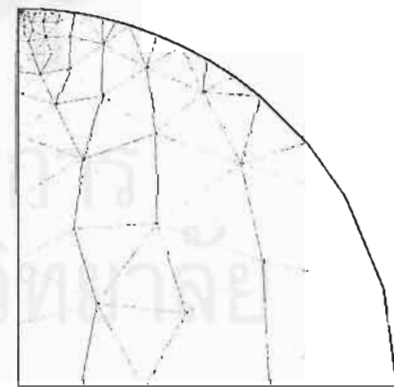
mesh 3 $\eta=0.3622$
Ne=40 Nn=32



mesh 4 $\eta=0.3143$
Ne=69 Nn=51

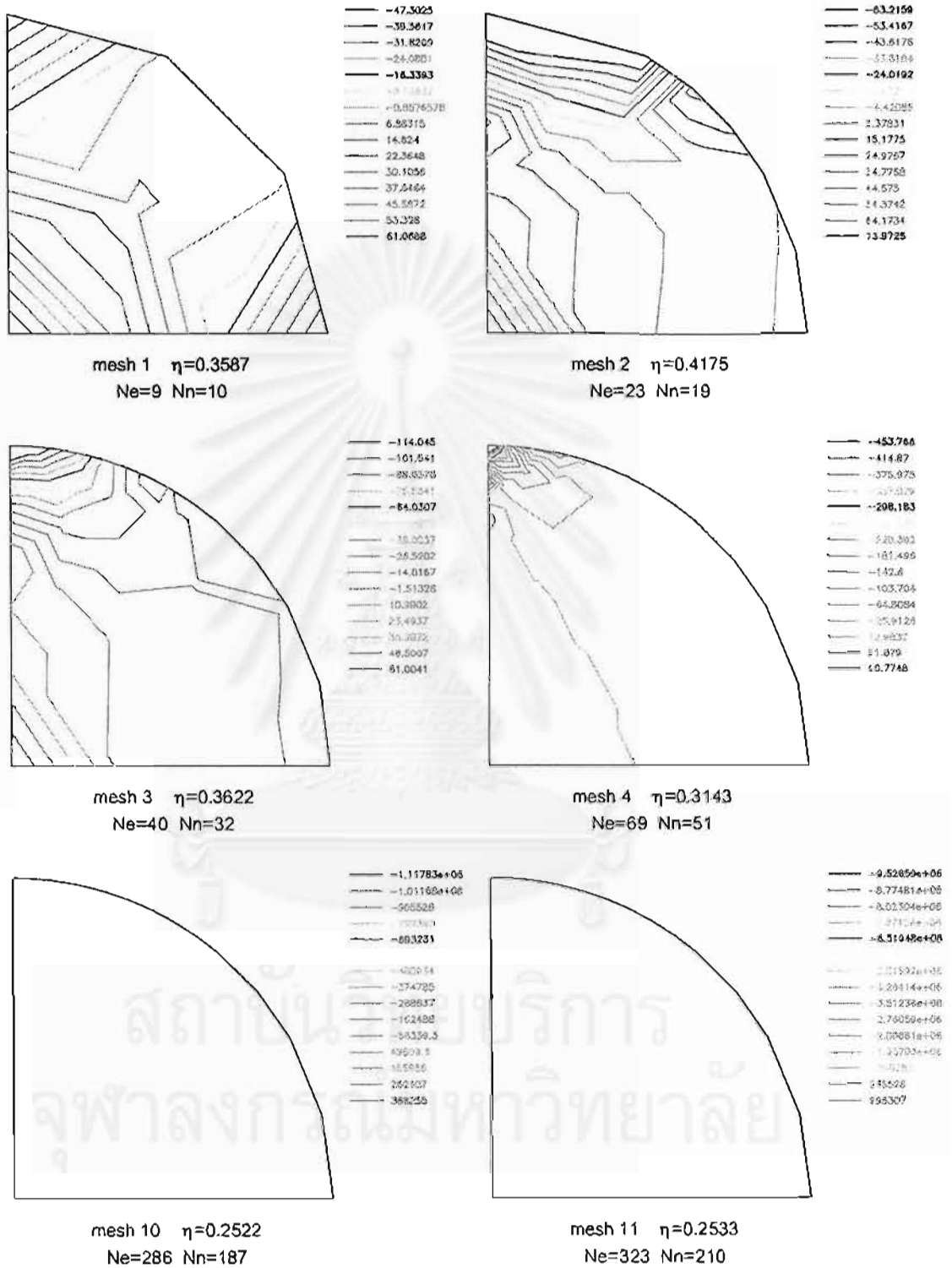


mesh 10 $\eta=0.2522$
Ne=286 Nn=187

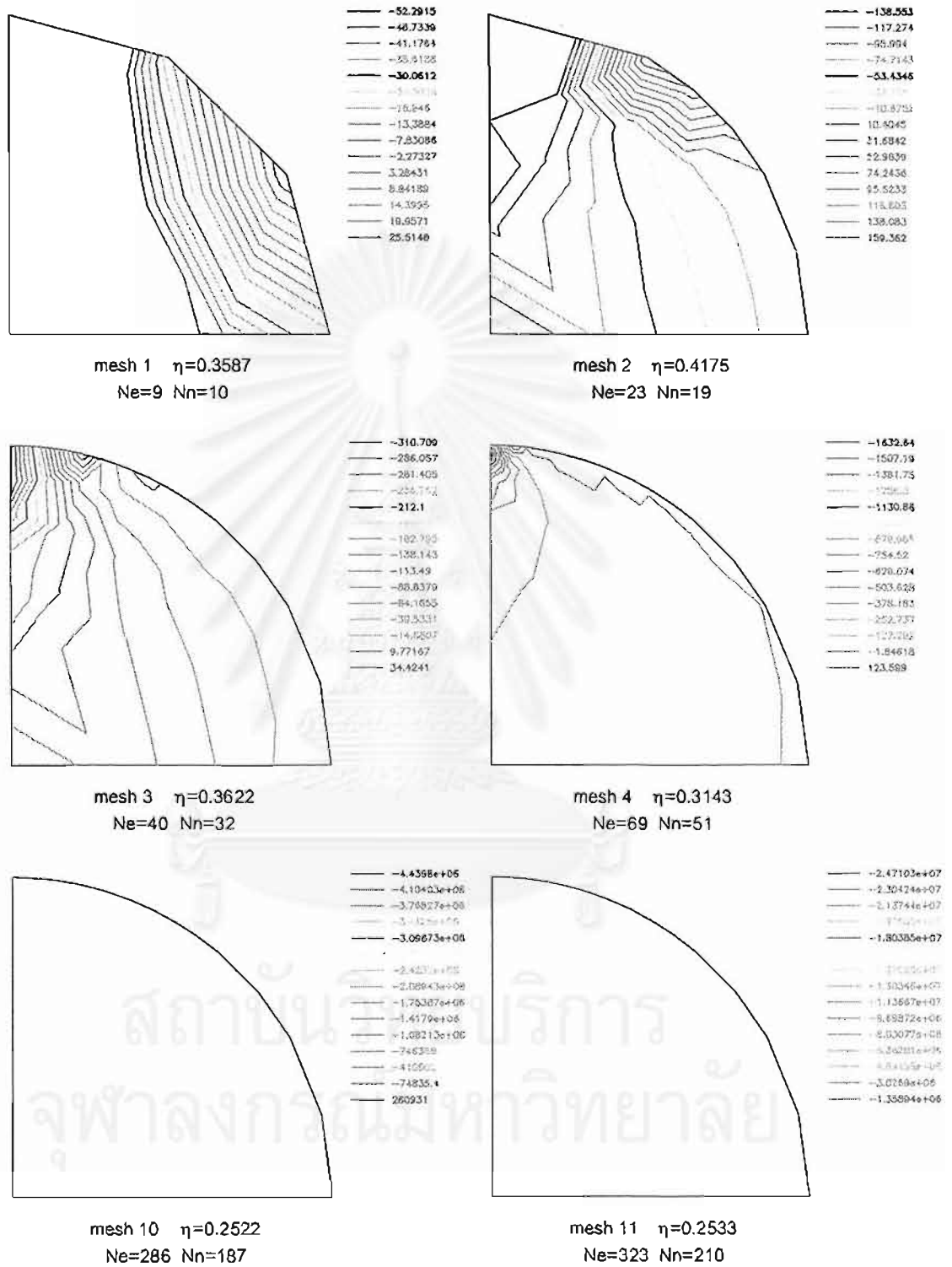


mesh 11 $\eta=0.2533$
Ne=323 Nn=210

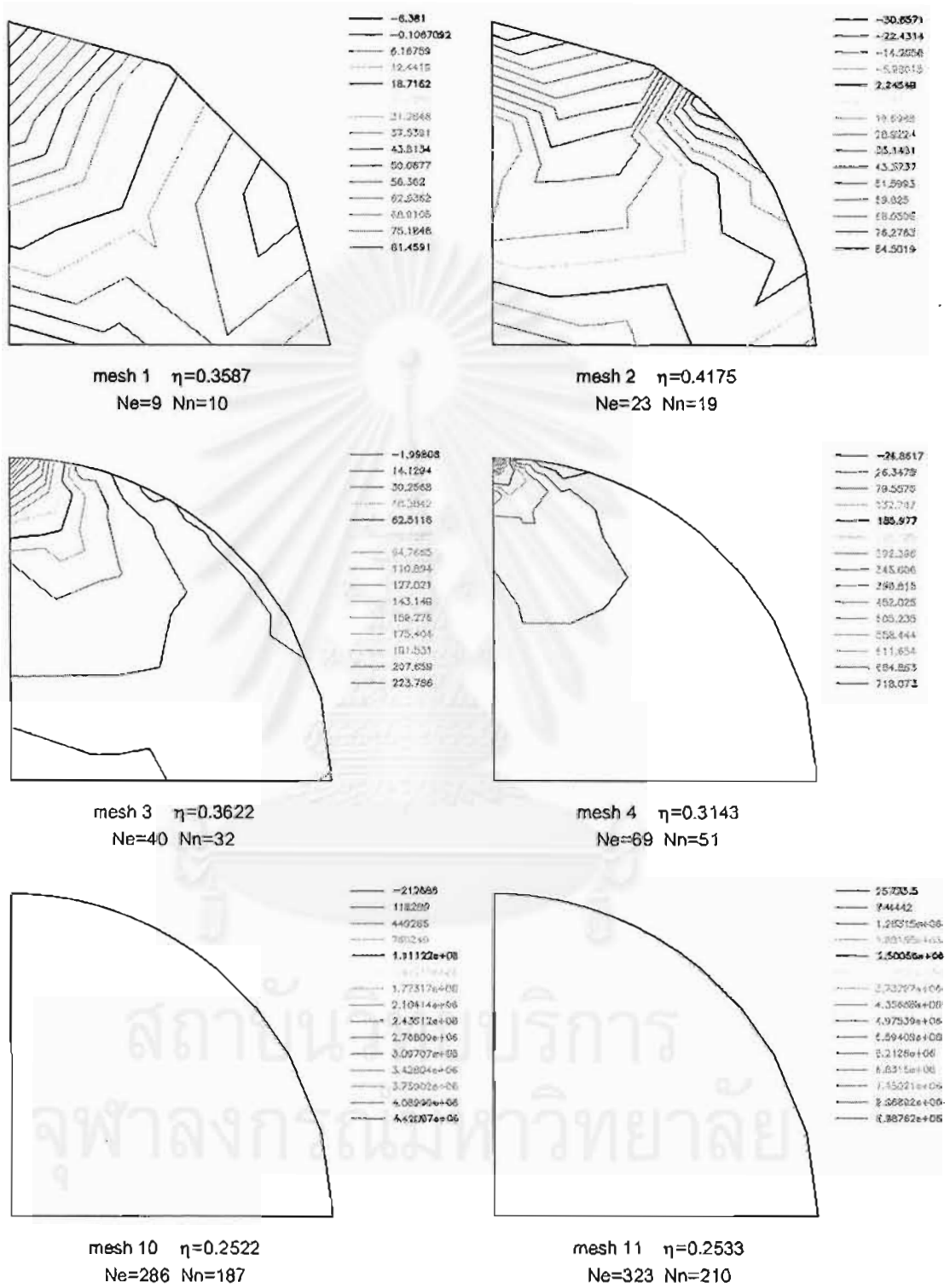
รูปที่ 4.2.2 โครงข่ายชิ้นส่วนสามเหลี่ยมในปัญหาแผ่นวงกลมบาง
(จำลองแรงกระทำเป็นจุด)



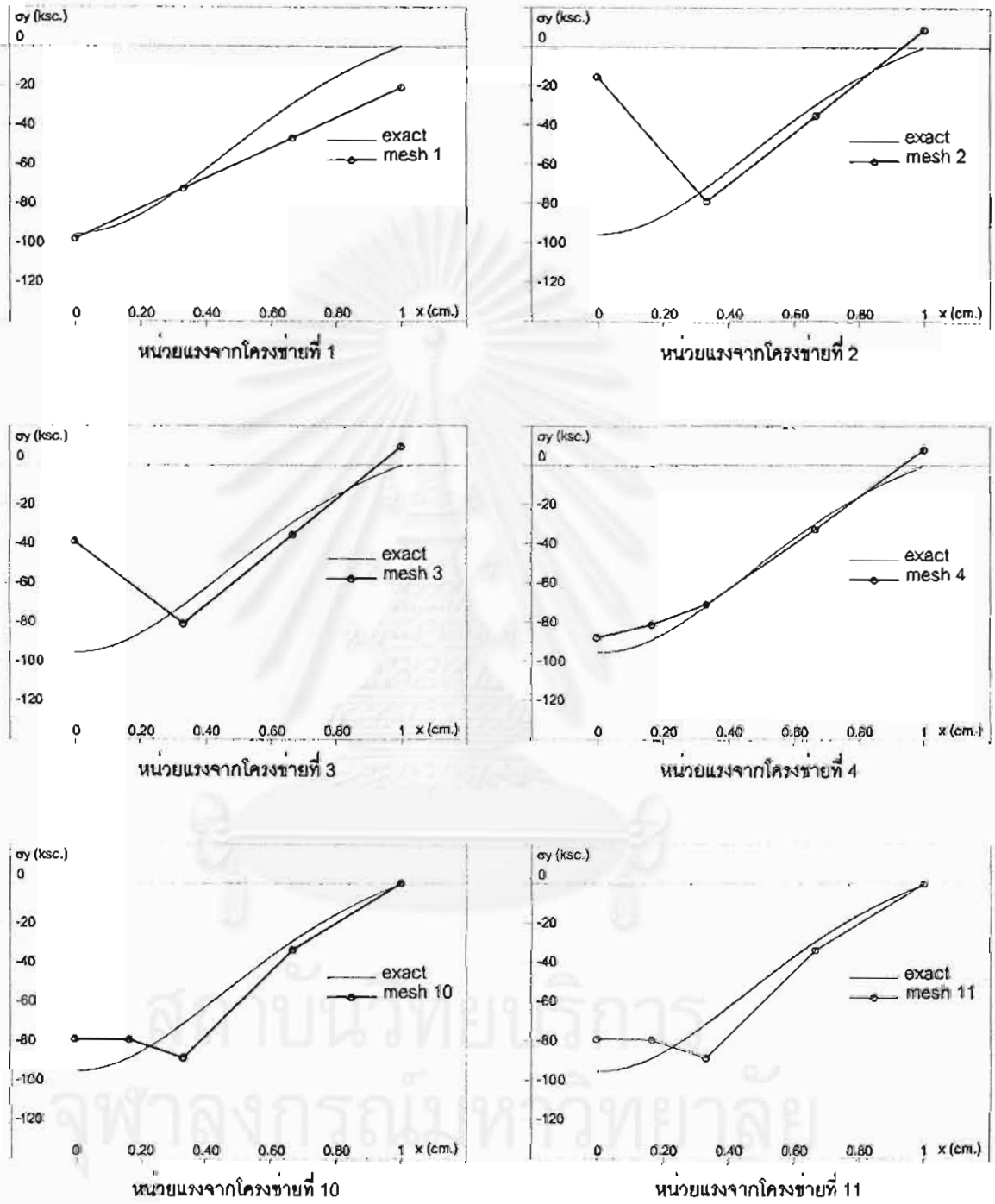
รูปที่ 4.2.3 การกระจายของหน่วยแรงในทิศทาง x ในปัญหาแผ่นวงกลมบาง (จำลองแรงกระทำเป็นจุด)



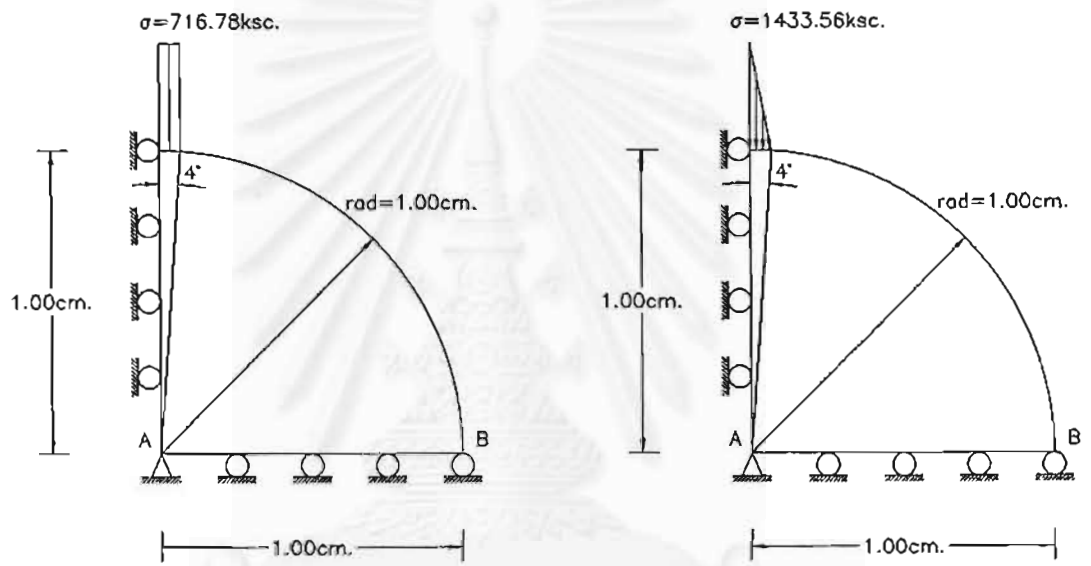
รูปที่ 4.2.4 การกระจายของหน่วยแรงในทิศทาง y ในปัญหาแผ่นวงกลมบาง (จำลองแรงกระทำเป็นจุด)



รูปที่ 4.2.5 การกระจายของหน่วยแรงเฉือน ในปัญหาแผ่นวงกลมบาง (จำลองแรงกระทำเป็นจุด)



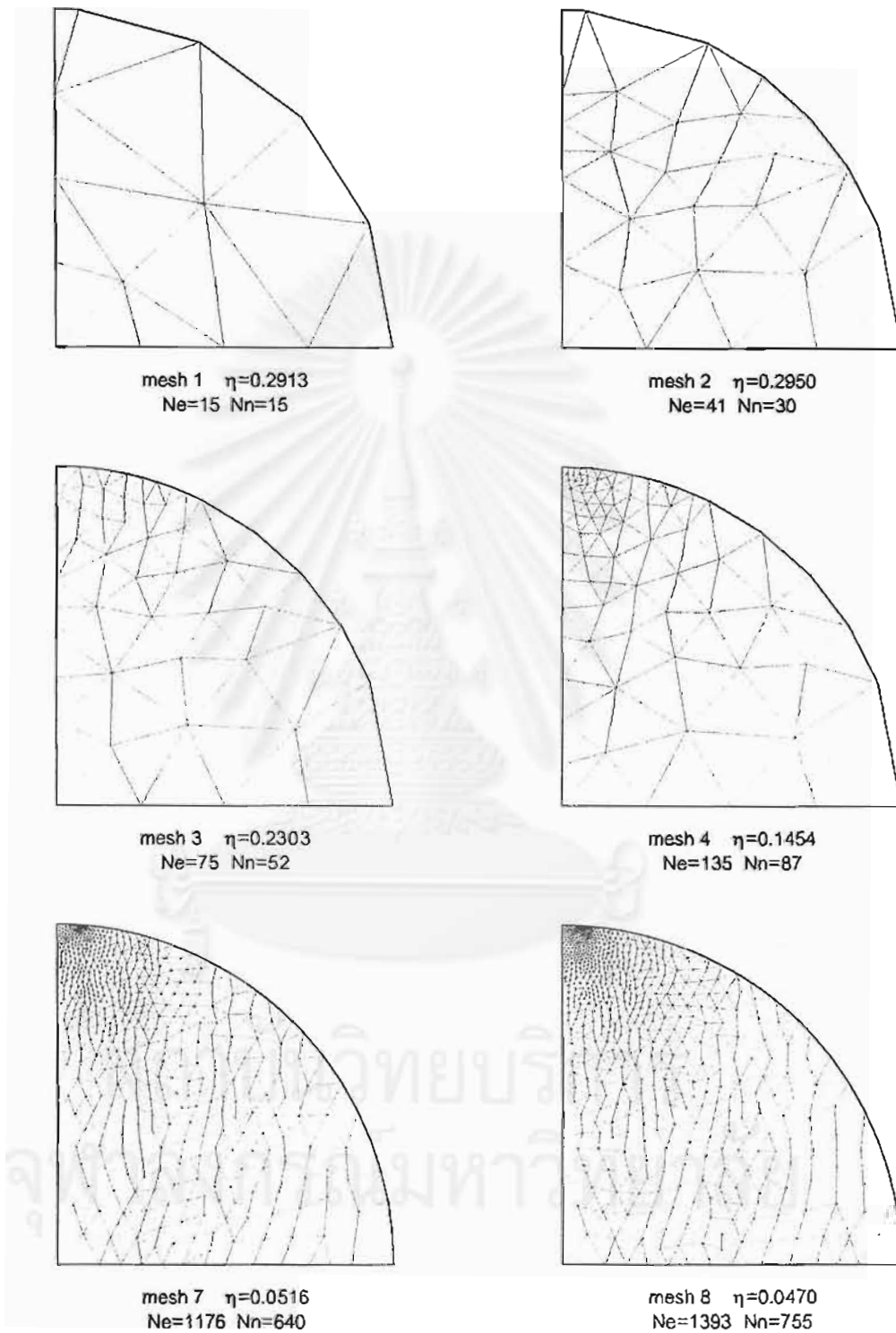
รูปที่ 4.2.6 กราฟแสดงค่าน้อยแรงในทิศทาง y ตามแนว AB ในปัญหาแผ่นวงกลมบาง (จำลองแรงกระทำเป็นจุด)



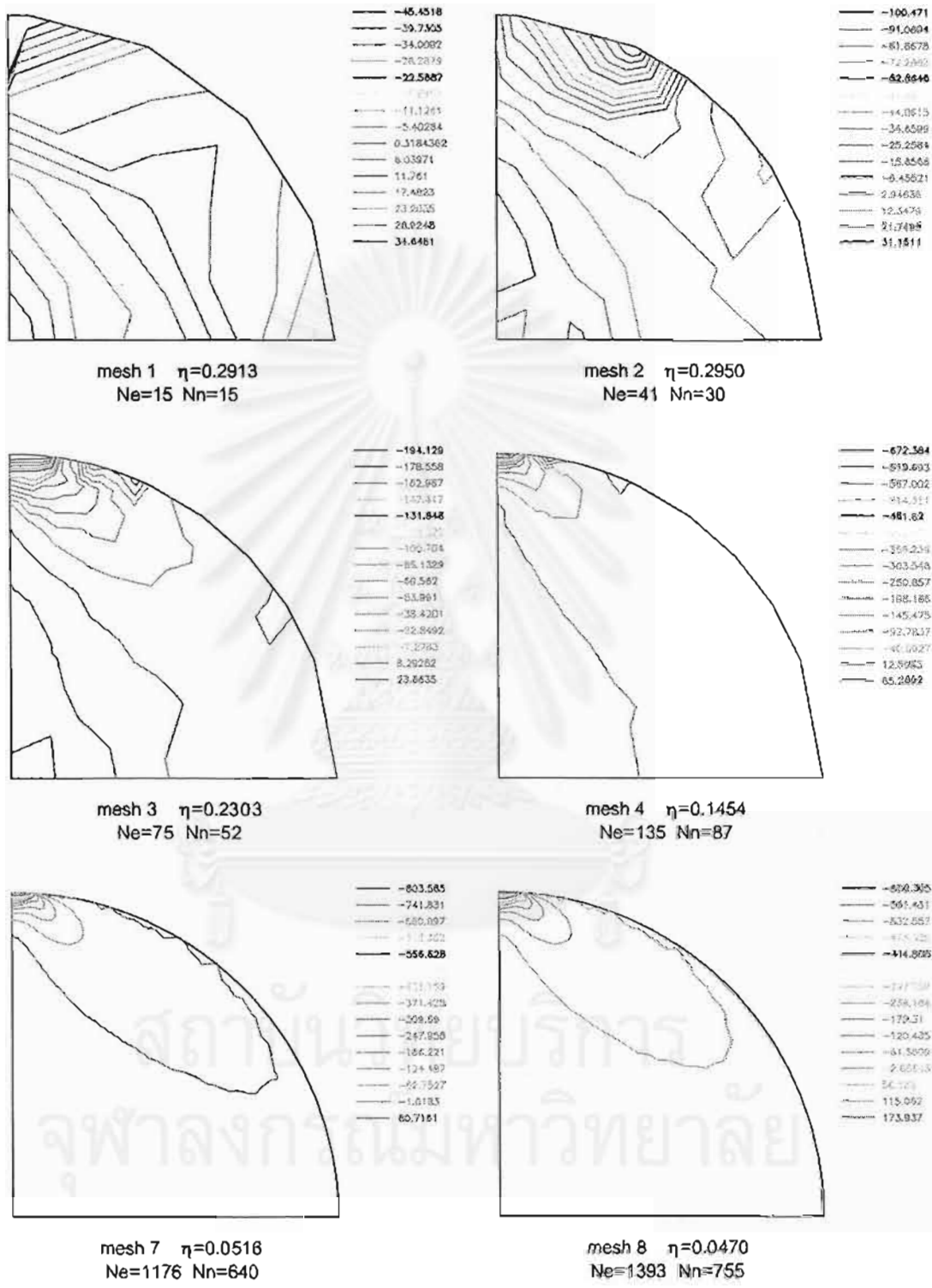
1.) แรงแผ่กระจายเท่ากันตลอดแนวกระทำ

2.) แรงแผ่กระจายเป็นรูปสามเหลี่ยม

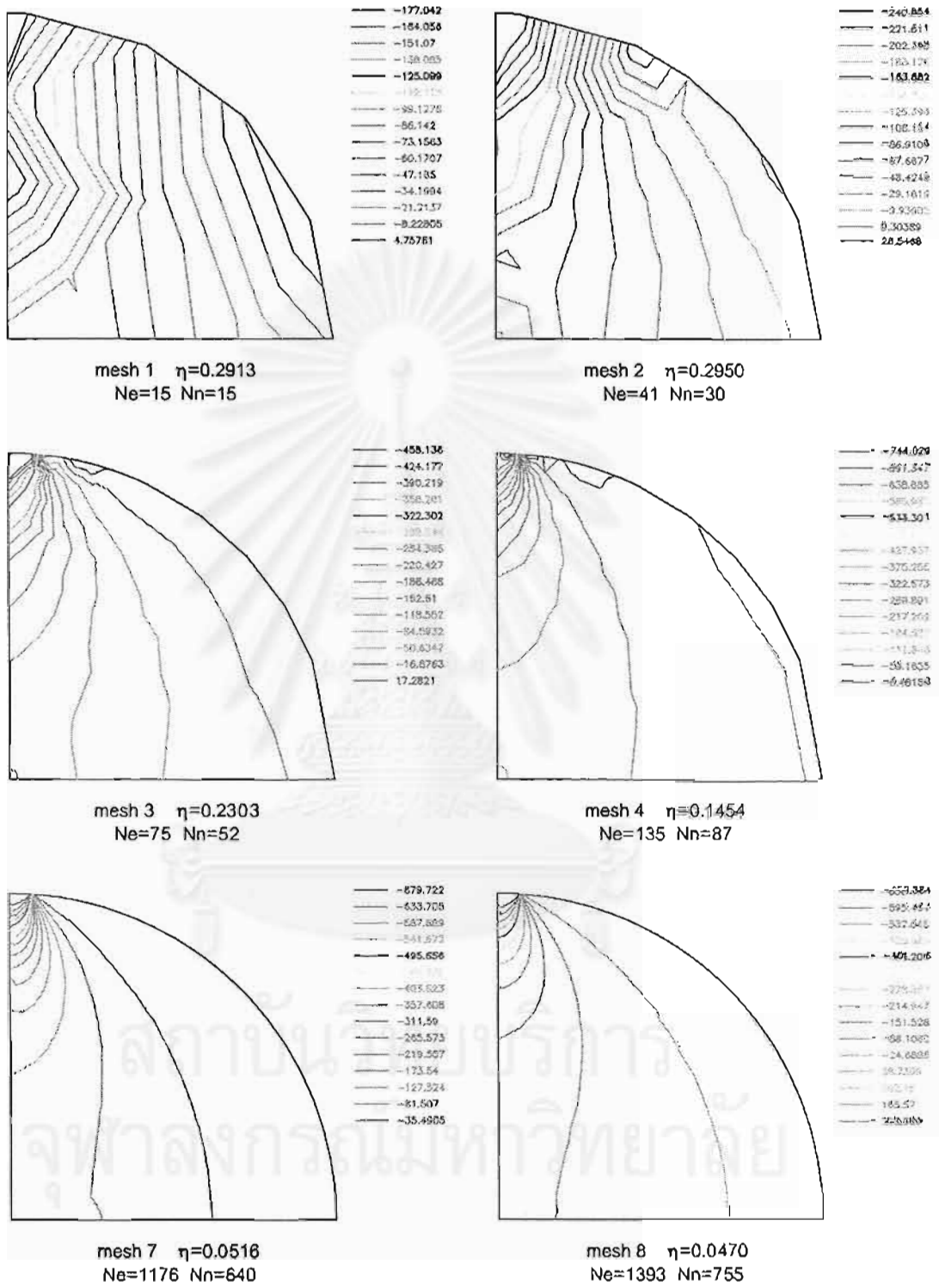
รูปที่ 4.2.7 แบบจำลองการวิเคราะห์ปัญหาแผ่นวงกลมบาง (จำลองเป็นแรงแผ่กระจาย)



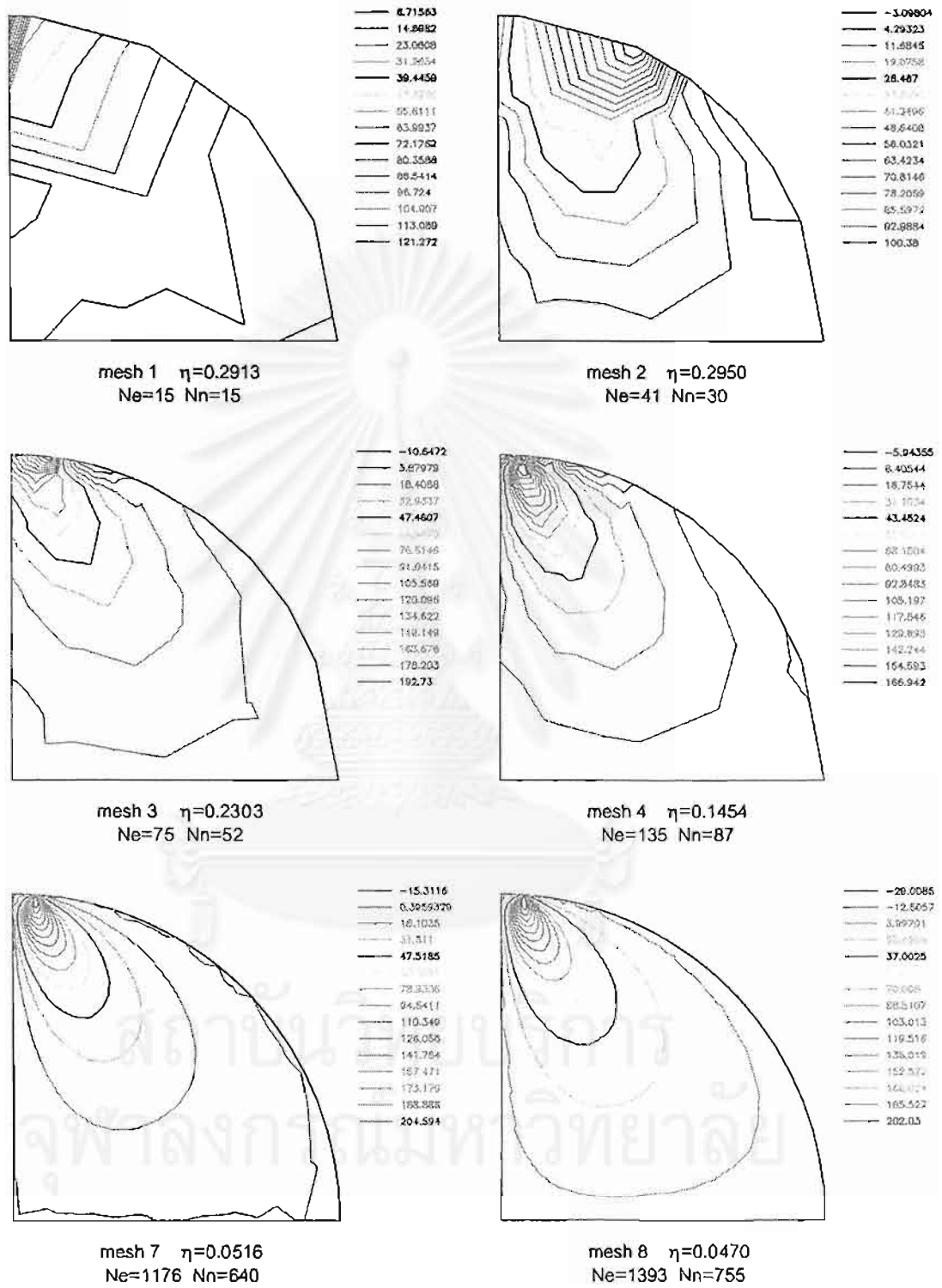
รูปที่ 4.2.8 โครงข่ายชิ้นส่วนสามเหลี่ยมในปัญหาแผ่นวงกลมบาง
(จำลองเป็นแรงแผ่กระจาย)



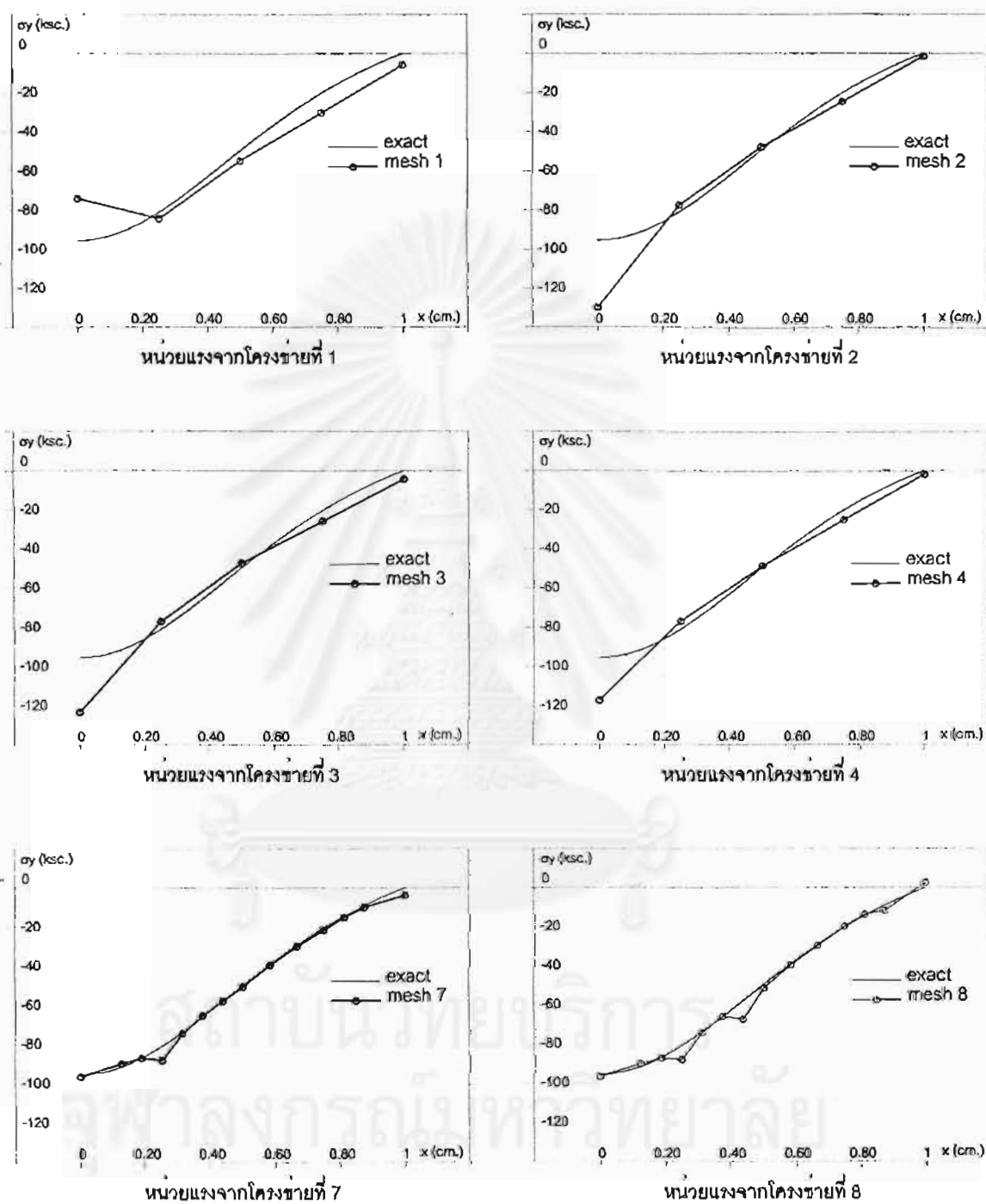
รูปที่ 4.2.9 การกระจายของหน่วยแรงในทิศทาง x ในปัญหาแผ่นวงกลมบาง (จำลองเป็นแรงแผ่กระจาย)



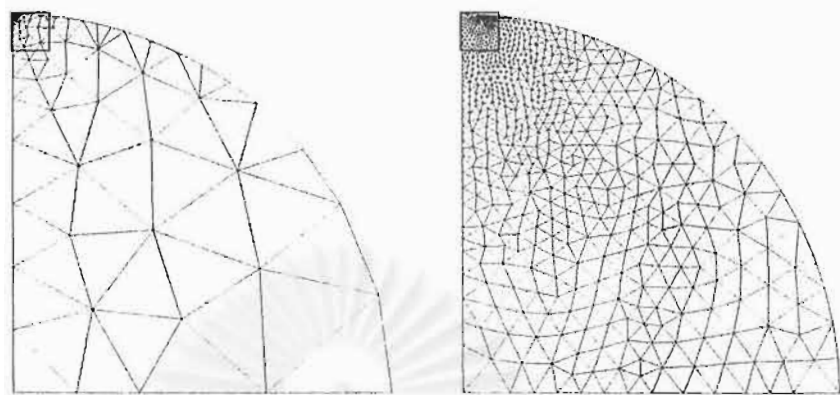
รูปที่ 4.2.10 การกระจายของหน่วยแรงในทิศทาง y ในปัญหาแผ่นวงกลมบาง (จำลองเป็นแรงแผ่กระจาย)



รูปที่ 4.2.11 การกระจายของหน่วยแรงเฉือน ในปัญหาแผ่นวงกลมบาง (จำลองเป็นแรงแผ่กระจาย)

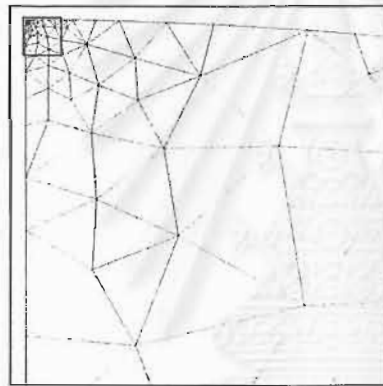


รูปที่ 4.2.12 กราฟแสดงค่าหน่วยแรงในทิศทาง y , ตามแนว AB ในปัญหาแผ่นวงกลมบาง (จำลองเป็นแรงแผ่กระจาย)

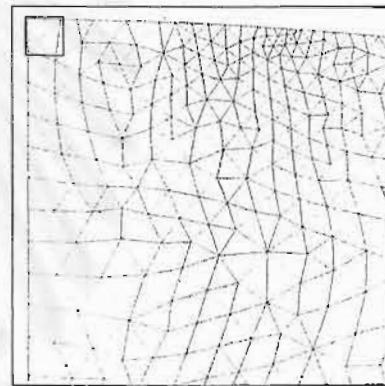


โครงข่ายสุดท้าย (แรงกระทำเป็นจุด)

โครงข่ายสุดท้าย (แรงแม่กระจาย)



ภาพขยาย 10 เท่า



ภาพขยาย 10 เท่า

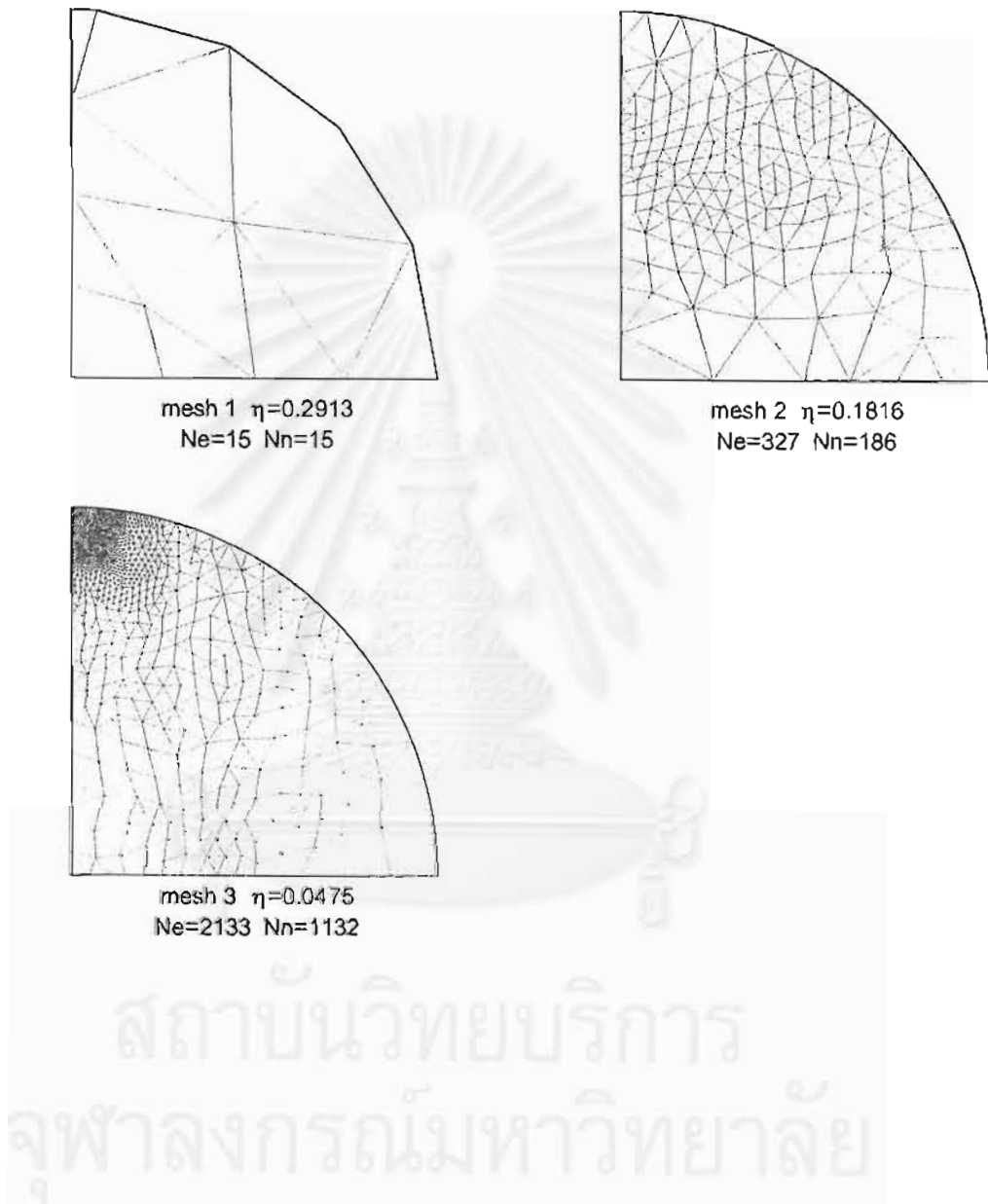


ภาพขยาย 100 เท่า

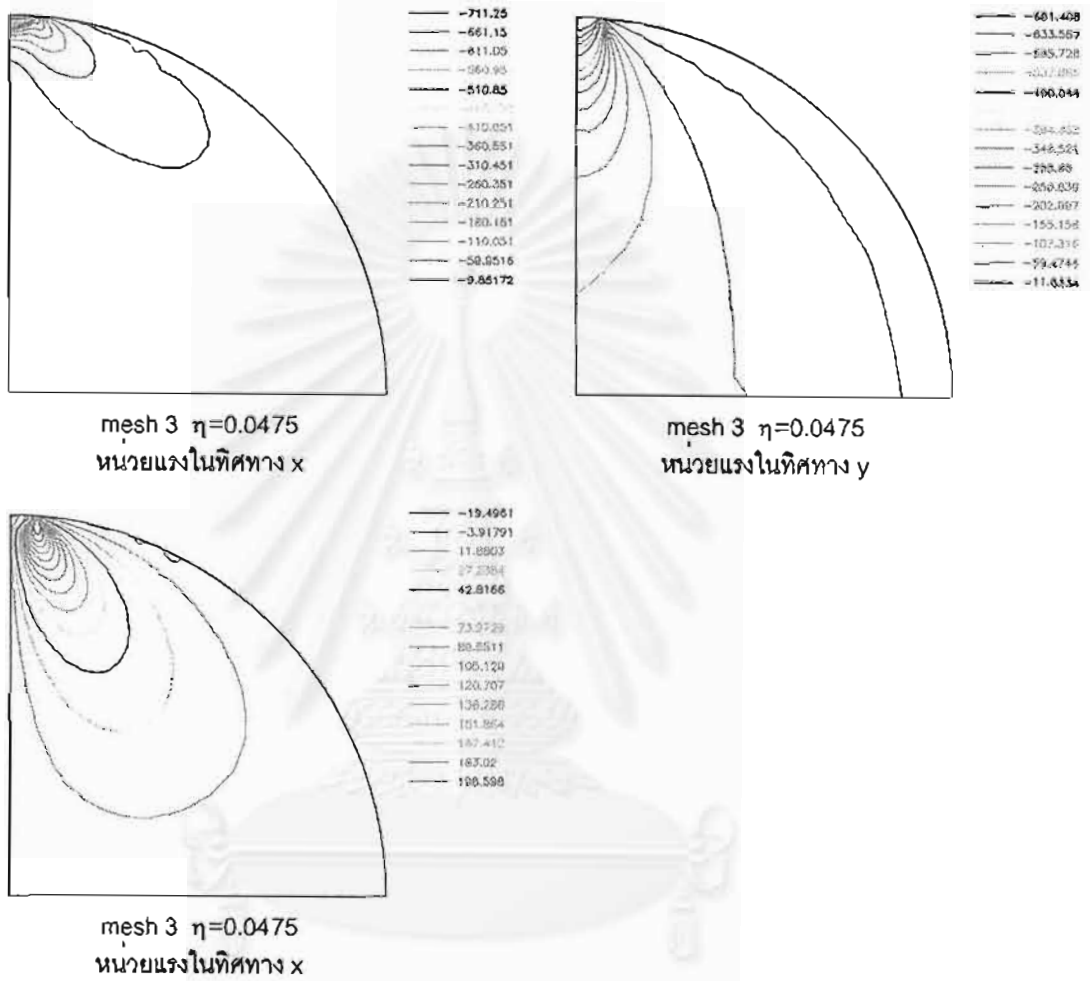


ภาพขยาย 100 เท่า

รูปที่ 4.2.13 เปรียบเทียบรายละเอียดโครงข่ายจากแรงกระทำเป็นจุดและแรงแม่กระจาย

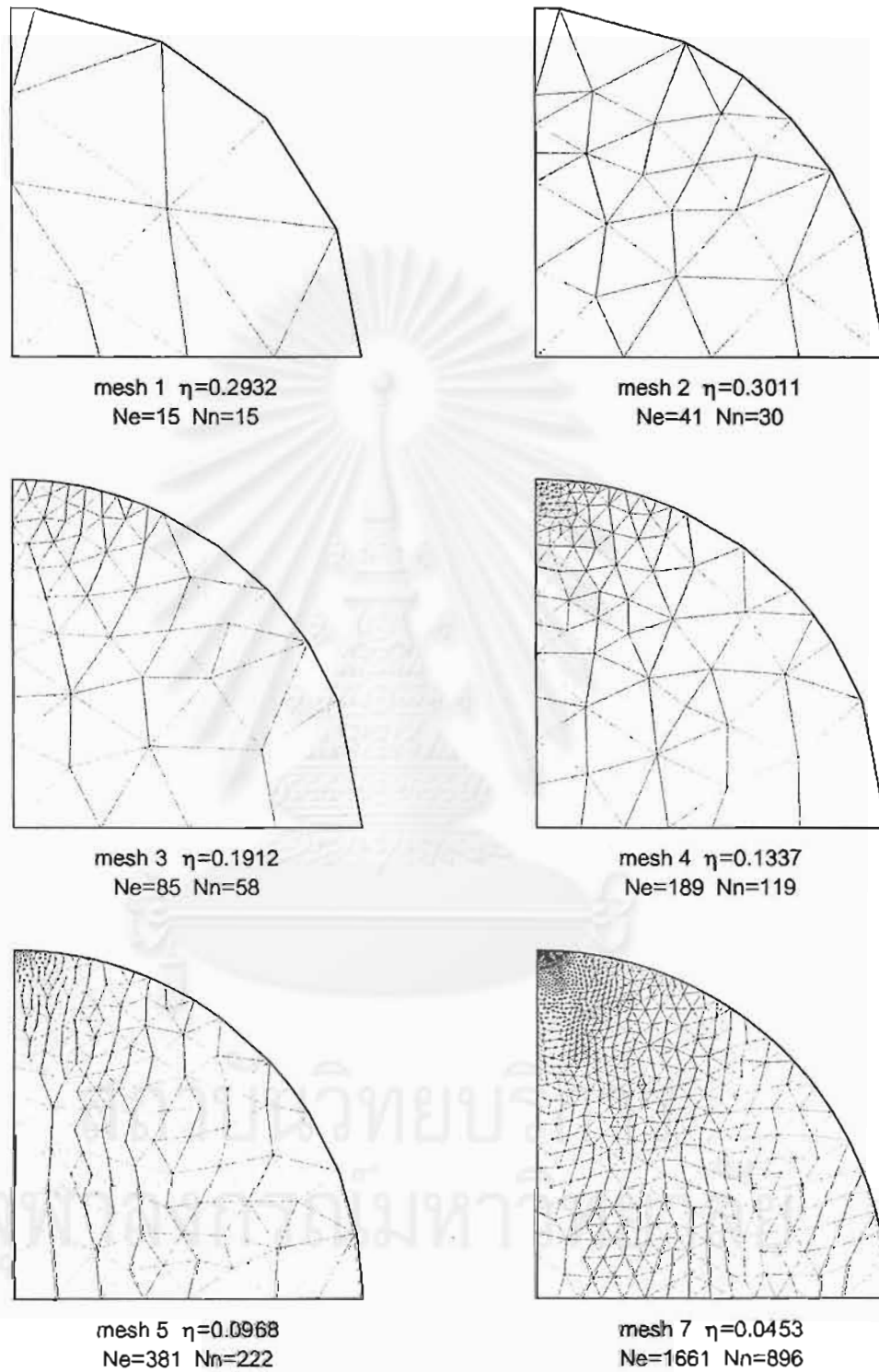


รูปที่ 4.2.14 โครงข่ายชิ้นส่วนสามเหลี่ยมที่ได้จากการปรับขนาดโดยตรง
และจำลองเป็นแรงแผ่กระจาย

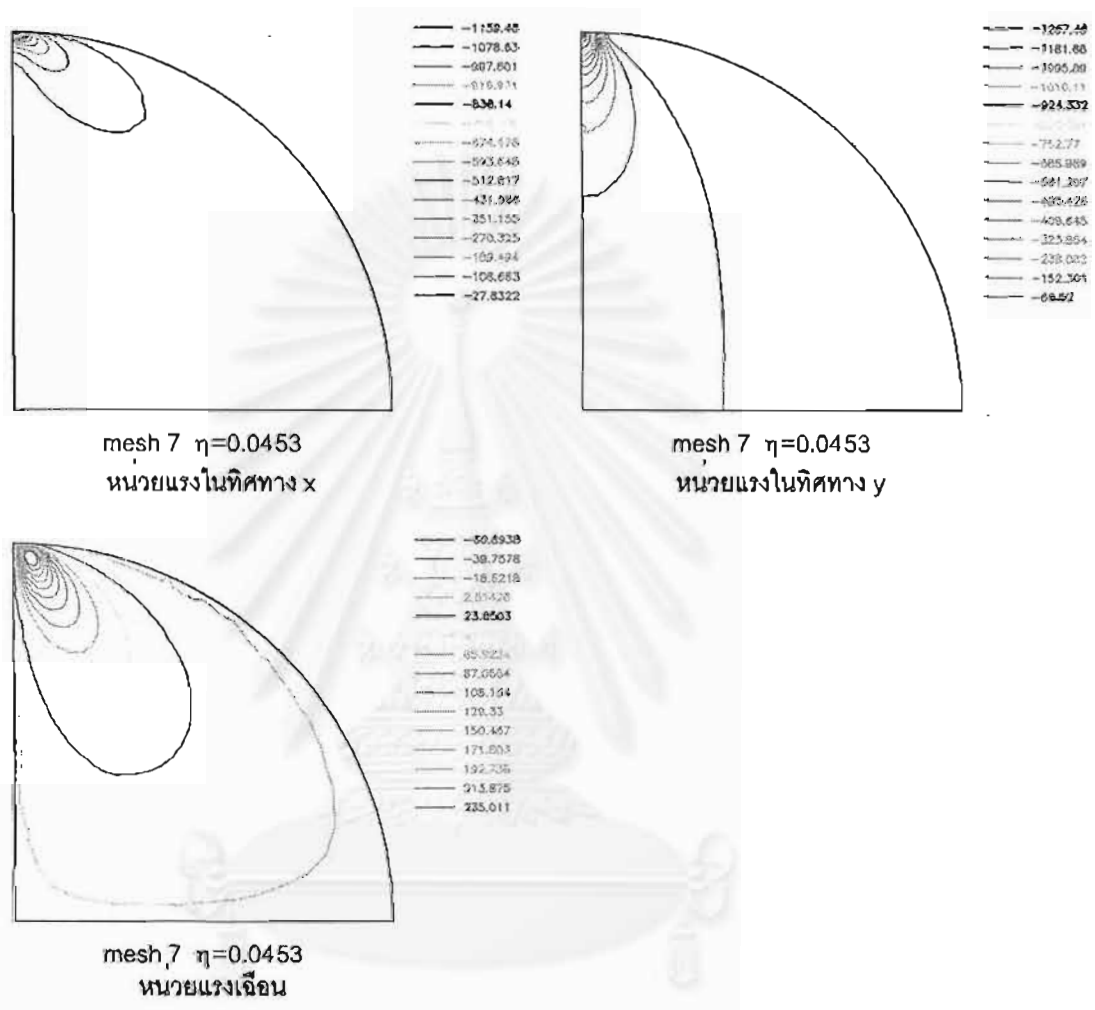


สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 4.2.15 การกระจายของหน่วยแรงในโครงข่ายสุดท้ายจากการปรับขนาดโดยตรง และจำลองเป็นแรงแมกระจาย

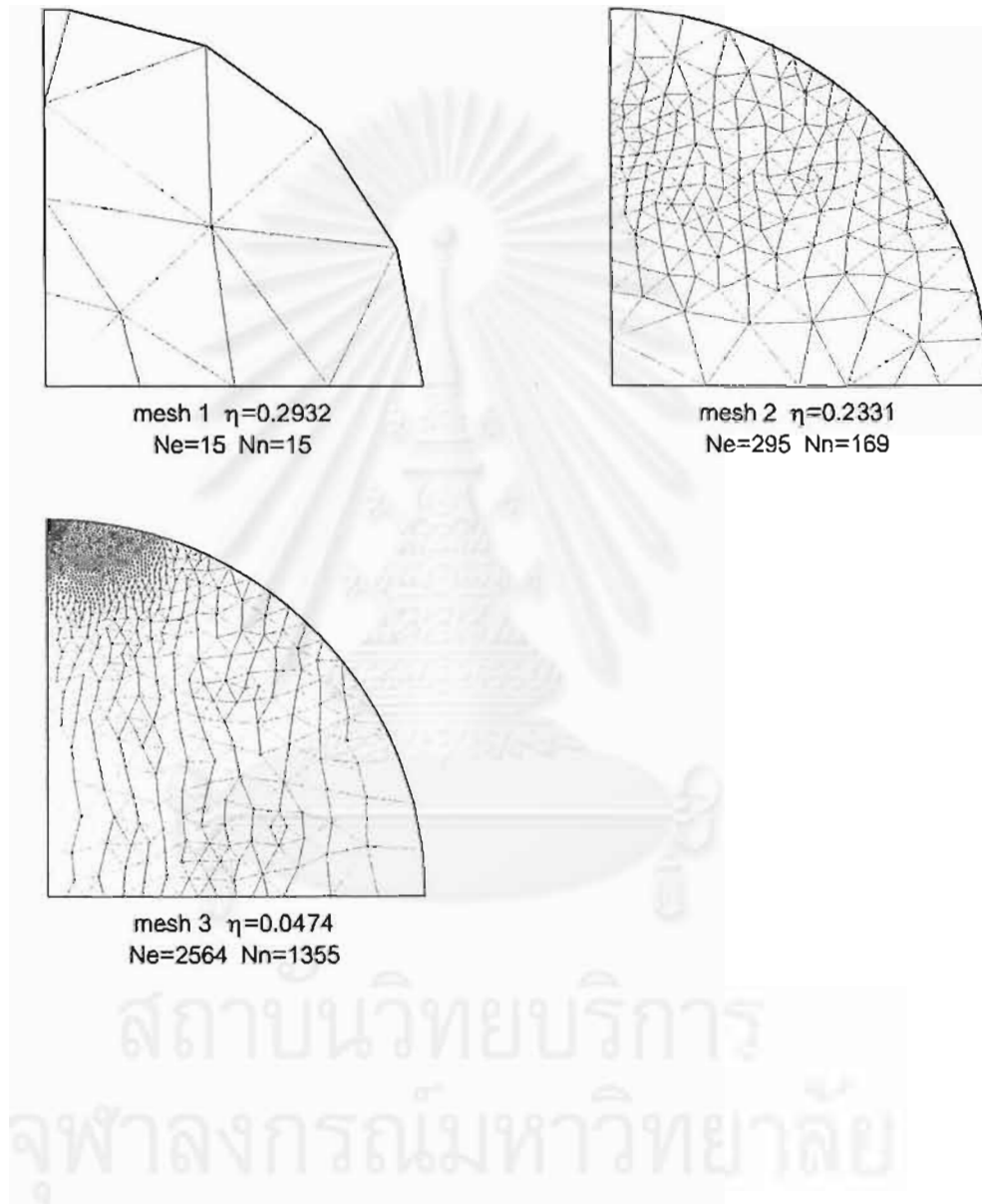


รูปที่ 4.2.16 โครงข่ายชิ้นส่วนสามเหลี่ยมจากการจำลองแรงเป็นรูปสามเหลี่ยม

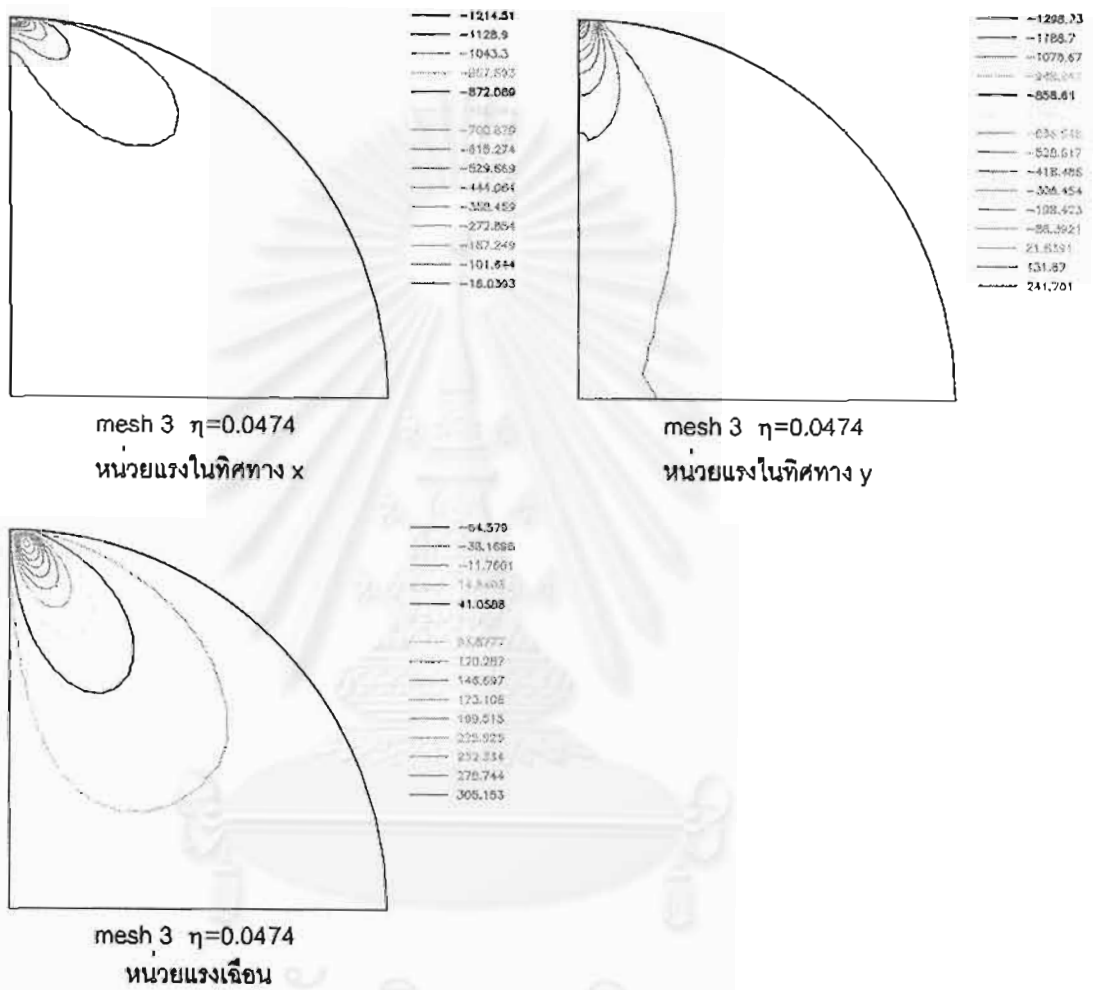


สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 4.2.17 การกระจายของหน่วยแรงในโครงข่ายสุดท้าย
จากการจำลองแรงเป็นรูปสามเหลี่ยม

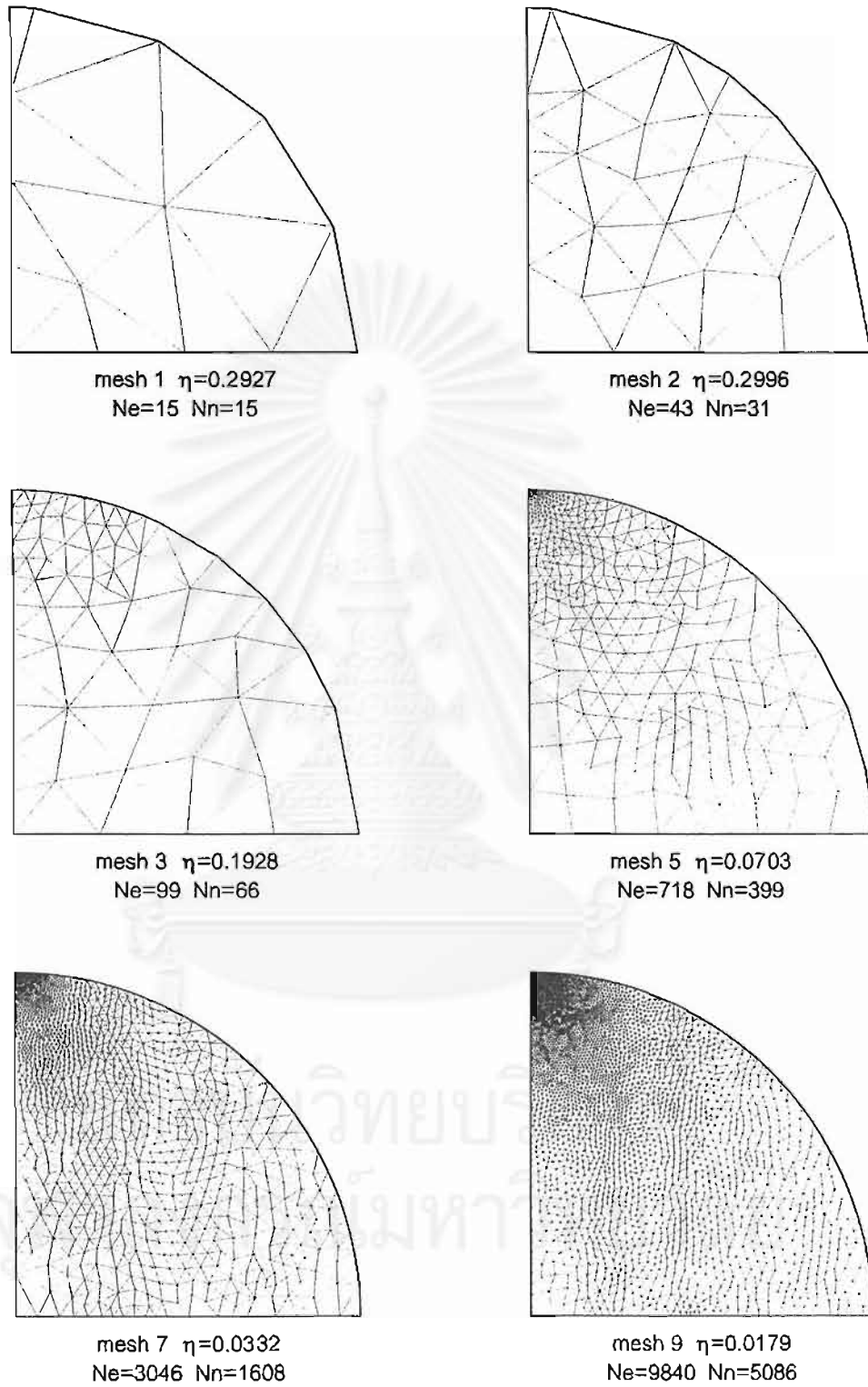


รูปที่ 4.2.18 โครงข่ายชิ้นส่วนสามเหลี่ยมที่ได้จากการปรับขนาดโดยตรง
และจำลองแรงแป็นรูปสามเหลี่ยม

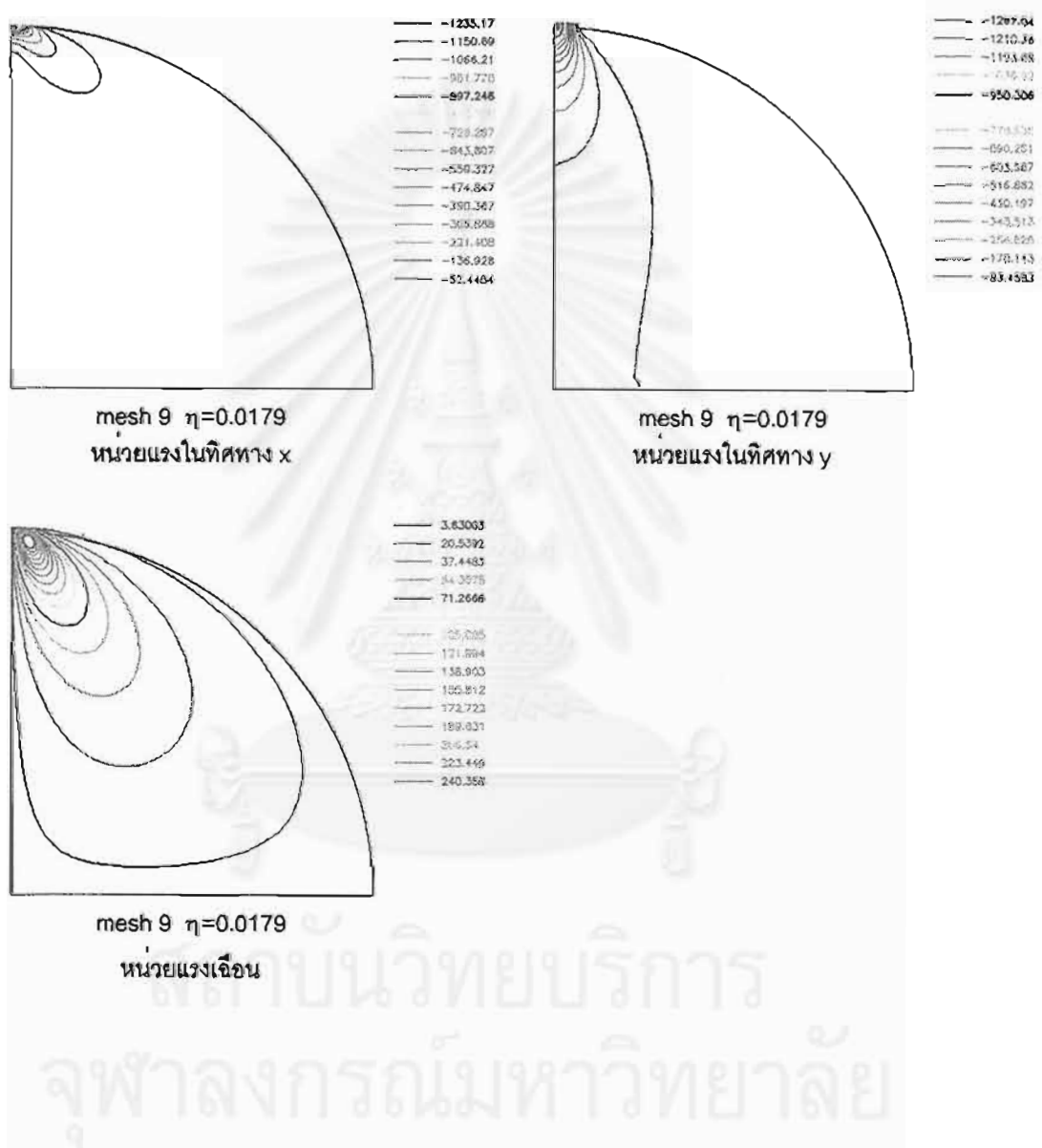


สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

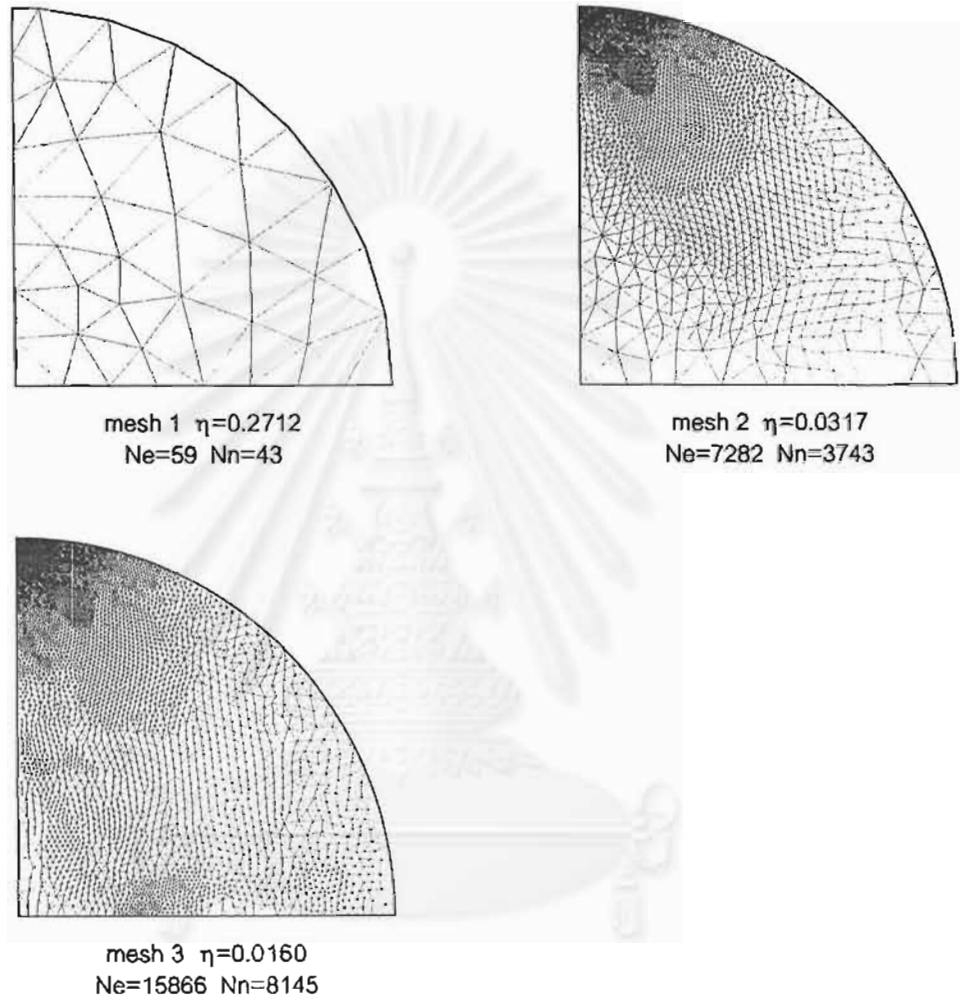
รูปที่ 4.2.19 การกระจายของหน่วยแรงในโครงข่ายสุดท้าย
จากการปรับขนาดโดยตรง และจำลองแรงเป็นรูปสามเหลี่ยม



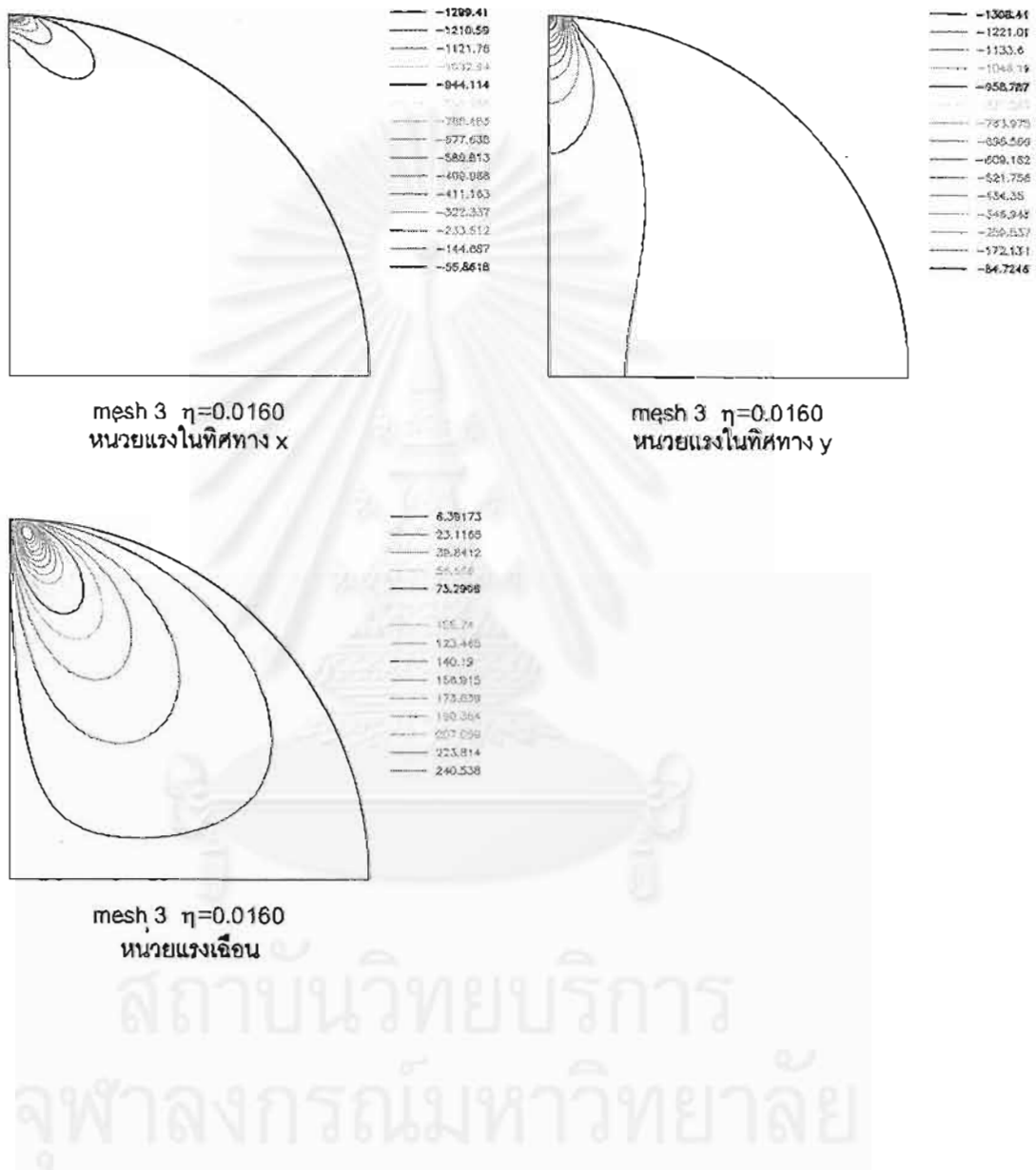
รูปที่ 4.2.20 โครงข่ายชิ้นส่วนสามเหลี่ยมที่ได้จากการจำลองแรงเป็นรูปสามเหลี่ยม และลดความคลาดเคลื่อนยอมให้



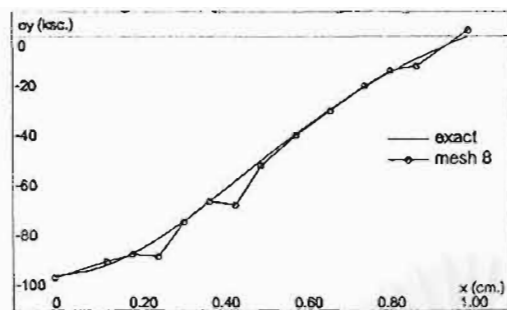
รูปที่ 4.2.21 การกระจายของหน่วยแรงในโครงข่ายสุดท้าย จากภาวะจำลองแรงเป็นรูปสามเหลี่ยม และลดความคลาดเคลื่อนยอมให้



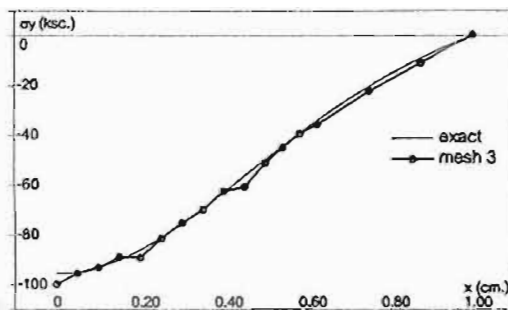
รูปที่ 4.2.22 โครงข่ายชิ้นส่วนสามเหลี่ยมที่ได้จากการปรับขนาดโดยตรง
จำกัดแรงเป็นรูปสามเหลี่ยม และลดความคลาดเคลื่อนยอมให้



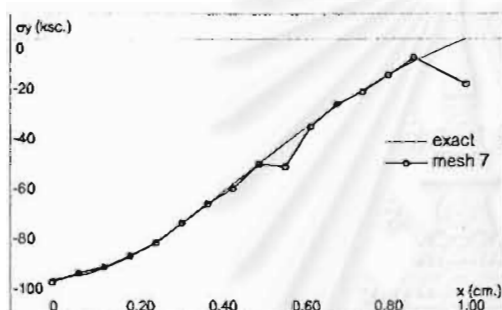
รูปที่ 4.2.23 การกระจายของหน่วยแรงในโครงข่ายสุดท้าย จากการปรับขนาดโดยตรง
จำลองแรงเป็นรูปสามเหลี่ยม และลดความคลาดเคลื่อนยอมให้



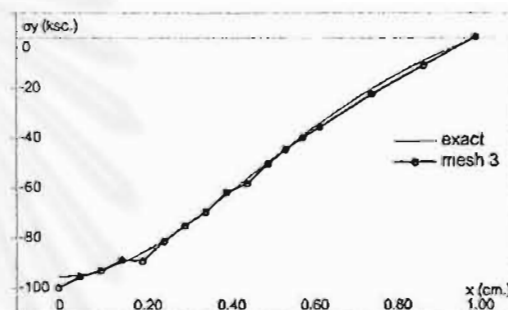
1.) แรงกระทำสามเหลี่ยมและกำหนดเป้าหมายสู่ค่าเฉลี่ย



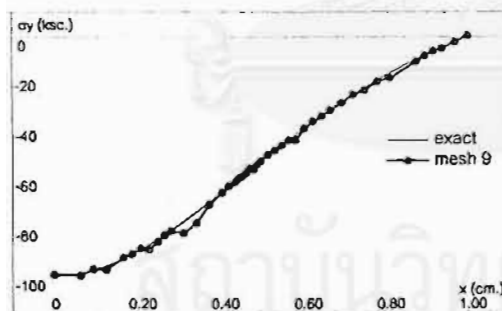
2.) แรงกระทำสามเหลี่ยมและกำหนดเป้าหมายสู่ค่าขอมโน



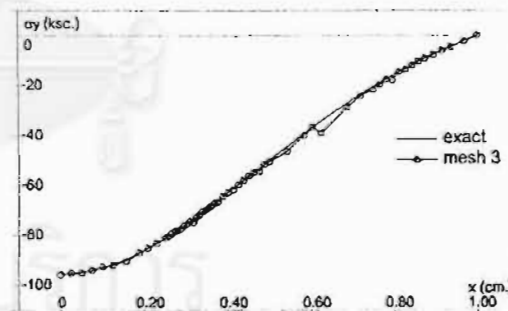
3.) แรงกระทำเป็นรูปสามเหลี่ยมและกำหนดเป้าหมายสู่ค่าเฉลี่ย



4.) แรงกระทำเป็นรูปสามเหลี่ยมและกำหนดเป้าหมายสู่ค่าขอมโน



5.) แรงกระทำเป็นรูปสามเหลี่ยม กำหนดเป้าหมายสู่ค่าเฉลี่ย และ ลดความคลาดเคลื่อนสัมพัทธ์ขอมโน



6.) แรงกระทำเป็นรูปสามเหลี่ยม กำหนดเป้าหมายสู่ค่าขอมโน และ ลดความคลาดเคลื่อนสัมพัทธ์ขอมโน

รูปที่ 4.2.24 เปรียบเทียบผลการวิเคราะห์จากการเปลี่ยนข้อกำหนดต่าง ๆ

สรุปผลการวิจัย

5.1 สรุปผลการวิจัย

กระบวนการปรับขนาดชิ้นส่วนที่ใช้ในงานวิจัยนี้ ใช้การประมาณค่าคลาดเคลื่อนจากการเปรียบเทียบผลที่ได้จากระเบียบวิธีไฟไนต์เอลิเมนต์กับค่าประมาณของผลเฉลยที่ได้จากวิธีการฟินิตเป็นหย่อม การสร้างชิ้นส่วนสามเหลี่ยมใช้วิธีการสร้างสามเหลี่ยมเดอลอนโดยวิธีฟรอนท์คืบหน้า และปรับขนาดชิ้นส่วนโดยการเพิ่มจุดลงในบริเวณพื้นที่ที่มีความคลาดเคลื่อนมาก โดยคำนวณความหนาแน่นของจุดจากความคลาดเคลื่อนโดยประมาณ ผลที่ได้สามารถสรุปเป็นข้อ ๆ ได้ดังนี้

การประมาณค่าคลาดเคลื่อน ได้ผลค่อนข้างดีในพื้นที่ภายในของปัญหา และจะมีความคลาดเคลื่อนมากขึ้นในบริเวณขอบ แต่ก็ไม่มีผลมากนักในกระบวนการปรับขนาดในงานวิจัยนี้ซึ่งมีกำลังของพหุนามในฟังก์ชันฐานต่ำ ถ้าใช้กับชิ้นส่วนที่มีกำลังของพหุนามสูงอาจจะต้องพิจารณาเรื่องความแม่นยำให้มากขึ้น

การสร้างชิ้นส่วนสามเหลี่ยม การสร้างชิ้นส่วนยังเป็นกระบวนการที่ใช้เวลานานที่สุดในโปรแกรมโดยใช้เวลาประมาณ 2 ใน 3 ของเวลาทั้งหมด และมีแนวโน้มจะมากขึ้นเมื่อจำนวนชิ้นส่วนมากขึ้น การพัฒนาให้เร็วขึ้นอาจทำได้โดยการใช้วิธีสร้างชิ้นส่วนอื่นร่วมด้วย เช่น วิธีกลับเส้นทแยงมุมซึ่งจะค่อนข้างช้าในตอนเริ่มต้นของกระบวนการ แต่จะเร็วขึ้นในตอนท้าย ตรงข้ามกับที่ใช้ในงานวิจัยซึ่งจะเร็วมากในตอนเริ่มต้นและจะเริ่มทำงานช้าช้อนในตอนท้าย ๆ ของกระบวนการ ดังนั้นการนำมาใช้ร่วมกันจะเพิ่มประสิทธิภาพขึ้นอีกมาก

การปรับขนาดชิ้นส่วนโดยการเพิ่มจุด สามารถให้ผลการวิเคราะห์สุดท้ายที่ค่อนข้างน่าพอใจ แต่การควบคุมขนาดของชิ้นส่วนยังไม่ได้มากนัก ถ้าสามารถควบคุมให้มีการจัดกลุ่มชิ้นส่วน แล้วสร้างจุดในบริเวณนั้นทั้งหมดโดยไม่ใช้จุดเดิมเลยจะควบคุมขนาดของชิ้นส่วนได้ดีกว่านี้ และเร่งอัตราการลู่เข้าได้มากขึ้น

ชิ้นส่วนที่ใช้ ในงานวิจัยนี้ใช้ชิ้นส่วนสามเหลี่ยมที่มีความเครียดคงที่ จึงสามารถคำนวณเมตริกซ์สติเฟเนสได้โดยตรงไม่ต้องอินทิเกรตเชิงเลข การคำนวณค่าต่าง ๆ ทำได้ง่ายทำให้เขียนรหัสคำนวณได้กระชับ แต่การประมาณค่าของชิ้นส่วนค่อนข้างหยาบทำให้ต้องใช้ชิ้นส่วนจำนวนมาก ส่งผลโดยตรงต่อการสร้างชิ้นส่วนและจำนวนสมการ ดังนั้นถ้าใช้ชิ้นส่วนอื่นที่มีกำลังของพหุนามสูงกว่านี้ก็จะได้ผลเฉลยที่ขึ้นและรวดเร็วกว่านี้

โปรแกรมที่ได้ สามารถวิเคราะห์ผลได้ดีในระดับหนึ่ง ทั้งนี้วัตถุประสงค์หลักของโปรแกรมคือเพื่อศึกษาความเป็นไปได้ในการพัฒนาโปรแกรมไฟไนต์เอลิเมนต์ที่ปรับขนาดชิ้นส่วนอัตโนมัติ ดังนั้นการจะนำไปใช้งานจำเป็นต้อง

ต้องมีการปรับปรุงและทดสอบโปรแกรมให้มากกว่านี้ จึงจะน่าเชื่อถือเพียงพอต่อการใช้งาน และถึงแม้ว่าโปรแกรมไฟไนต์เอเลเมนต์ที่ปรับขนาดชิ้นส่วนอัตโนมัติจะสามารถปรับขนาดให้เหมาะสมกับความคลาดเคลื่อนและให้ผลเฉลยที่มีความคลาดเคลื่อนในช่วงที่ยอมให้ก็ตาม ผู้ใช้งานก็ยังคงจำเป็นต้องมีความรู้ เพื่อกำหนดค่าคลาดเคลื่อนที่ยอมให้ ให้เหมาะสมกับปัญหาซึ่งจะแตกต่างกันไป สามารถกำหนดขอบเขตเริ่มต้นของปัญหาได้อย่างมีประสิทธิภาพ สร้างแบบจำลองที่ใช้ในการวิเคราะห์ได้เหมาะสมกับปัญหา และสามารถตรวจสอบผลที่ได้ว่ามีความน่าเชื่อถือเพียงใด

5.2 ข้อเสนอนั้นอื่น ๆ

การใช้กับโครงสร้างอื่น ๆ เช่น ในโครงสร้างแผ่นพื้นสามารถนำการสร้างชิ้นส่วนไปใช้ได้เลยแต่ต้องศึกษาเพิ่มเติมในเรื่องการหาความคลาดเคลื่อน สำหรับโครงสร้างเปลือกบางถ้าสามารถประมาณระดับของตำแหน่งภายในได้ก็สามารถนำการสร้างชิ้นส่วนไปปรับใช้ได้ไม่ยาก

การประยุกต์ใช้กับการวิเคราะห์ปัญหาโดยการแบ่งเป็นโครงสร้างย่อย ถ้าสามารถควบคุมการแบ่งโซนของปัญหาได้ก็สามารถนำไปประยุกต์ใช้ได้เช่นกันโดยการเพิ่มการแก้สมการด้วยวิธีการแก้สมการโดยตรง

การประยุกต์ใช้กับชิ้นส่วนอื่น ๆ ที่ไม่ใช่ชิ้นส่วนสามเหลี่ยม สามารถทำได้เช่นกันโดยการแปลงรูปจากชิ้นส่วนสามเหลี่ยมที่ได้นี้ ซึ่งในปัจจุบันนี้ก็มีผู้พัฒนาวิธีการแปลงรูปได้หลายวิธีแล้ว

รายการอ้างอิง

1. Kelly D. W. , De J. P. , Gago S. R. , Zienkiewicz O. C. , Babuska I. *A Posteriori Error Analysis and Adaptive Processes in the Finite Element Method : Part I - Error Analysis* Int. J. Numer. Meth. in Eng , (1983) , vol.19 , pp.1593-1619
2. Kelly D. W. , De J. P. , Gago S. R. , Zienkiewicz O. C. , Babuska I. *A Posteriori Error Analysis and Adaptive Processes in the Finite Element Method : Part II - Adaptive Mesh Refinement* Int. J. Numer. Meth. in Eng , (1983) , vol.19 , pp.1621-1656
3. Zienkiewicz O.C. , Zhu J.Z. *A Simple Error Estimator and Adaptive Procedure for Practical Engineering Analysis* Int. J. Numer. Meth. in Eng , (1987) , vol.24 , pp.337-357
4. Ainsworth M. , Zhu J.Z. , Craig A.W. , Zienkiewicz O.C. *Analysis of the Zienkiewicz-Zhu A-Posteriori Error Estimator in the Finite Element Method* Int. J. Numer. Meth. in Eng , (1989) , vol.28 , pp.2161-2174
5. Zhu J. Z. , Zienkiewicz O. C. *Superconvergence Recovery Technique and a Posteriori Error Estimators* Int. J. Numer. Meth. in Eng , (1990) , vol.30 , pp.1321-1339
6. Zienkiewicz O.C. , Zhu J.Z. *Adaptivity and Mesh Generation* Int. J. Numer. Meth. in Eng , (1991) , vol.32 , pp.783-810
7. Zienkiewicz O.C. , Zhu J.Z. *The Superconvergent Patch Recovery and a Posteriori Error Estimates. Part I: The Recovery Technique* Int. J. Numer. Meth. in Eng , (1992) , vol.33 , pp.1331-1364
8. Zienkiewicz O.C. , Zhu J.Z. *The Superconvergent Patch Recovery and a Posteriori Error Estimates. Part II: Error Estimates and Adaptivity* Int. J. Numer. Meth. in Eng , (1992) , vol.33 , pp.1365-1382
9. Blacker T. , Belytschko T. *Superconvergent Patch Recovery with Equilibrium and Conjoint Interpolant Enhancements* Int. J. Numer. Meth. in Eng , (1994) , vol.37 , pp.517-536
10. Wiberg N. -E. , Abdulwahab WF. *Patch Recovery Based on Superconvergent Derivatives and Equilibrium* Int. J. Numer. Meth. in Eng , (1993) , vol.36 , pp.2703-2724
11. Wiberg N. -E. , Abdulwahab WF. , Ziukas S. *Enhanced Superconvergent Patch Recovery Incorporating Equilibrium and Boundary Conditions* Int. J. Numer. Meth. in Eng , (1994) , vol.37 , pp.3417-3440
12. Lee T. , Park H. C. , Lee S. W. *A Superconvergent Stress Recovery Technique with Equilibrium Constraint* Int. J. Numer. Meth. in Eng , (1997) , vol.40 , pp.1139-1160
13. Herrmann L. R. *Laplacian - Isoparametric Grid Generation Scheme* J. Eng. Mech. Div. , (1976) , October , pp.749-759

14. Bachmann P. L. , Wittchen S. L. , Shephard M. S. , Grice K. R. , Yerry M. A. *Robust, Geometrically Based, Automatic Two Dimensional Mesh Generation* Int. J. Numer. Meth. in Eng , (1987) , vol.24 , pp.1043-1078
15. Lo S. H. *A New Mesh Generation Scheme for Arbitrary Planar Domains* Int. J. Numer. Meth. in Eng , (1985) , vol.21 , pp.1403-1426
16. Peraire J. , Peiro J. , Farmaggia L. , Morgan K. , Zienkiewicz O. C. *Finite Element Euler Computations in Three Dimensions* Int. J. Numer. Meth. in Eng , (1988) , vol.26 , pp.2135-2159
17. Zhu J. Z. , Zienkiewicz O. C. , Hinton E. , Wu J. *A New Approach to the Development of Automatic Quadrilateral Mesh Generation* Int. J. Numer. Meth. in Eng , (1991) , vol.32 , pp.849-866
18. Cavendish J. C. *Automatic Triangulation of Arbitrary Planar Domain for the Finite Element Method* int. J. Numer. Meth. In Eng , (1974) , vol.8 , pp.679-696
19. Lo S. H. *Delaunay Triangulation of Non-Convex Planar Domains* Int. J. Numer. Meth. in Eng , (1989) , vol.28 , pp.2695-2707
20. Borouchaki H. , George P. L. *Aspects of 2-D Delaunay Mesh Generation* Int. J. Numer. Meth. in Eng , (1997) , vol.40 , pp.1957-1975
21. Alexander F. *Almost Regular Delaunay - Triangulations* Int. J. Numer. Meth. in Eng , (1997) , vol.40 , pp.4595-4610
22. Johnston B. P. , Sullivan J. M. , Kwasnik J. R. and A. *Automatic Conversion of Triangular Finite Element Meshes to Quadrilateral Elements* Int. J. Numer. Meth. in Eng , (1991) , vol.31 , pp.67-84
23. Mackinnon R. J. , Carrey G. F. *Superconvergent Derivatives: A Taylor Series Analysis* Int. J. Numer. Meth. in Eng , (1989) , vol.28 , pp.489-509
24. Mar A. , Hicks M. A. *A Benchmark Computational Study of Finite Element Error Estimation* Int. J. Numer. Meth. in Eng , (1996) , vol.39 , pp.3969-3983
25. Karamete B. K. , Tokdemir T. , Ger M. *Unstructured Grid Generation and a Simple Triangulation Algorithm for Arbitraries Using Object Oriented Programming* Int. J. Numer. Meth. In Eng , (1997) , vol.40 , pp.251-268
26. Weatherill N. P. , Hassan O. *Efficient Three - Dimensional Delaunay Triangulation with Automatic Point Creation and Imposed Boundary Constraints* Int. J. Numer. Meth. in Eng , (1994) , vol.37 , pp.2005-2039
27. Bowyer A. *Computing Dirichlet Tessellations* The Computer Journal , (1981) , vol.24 , No.2 , pp.162-172 ,
28. Floriani L. De , Falcidieno B. , Pienovi C. *Delaunay-based Representation of Surfaces Defined Over Arbitrarily Shaped Domains* Computer vision , graphics , and image processing , (1985) , vol.32 , pp.127-140

29. Sloan S.W. *A Fast Algorithm for Constructing Delaunay Triangulations in the Plane* Advances in Engineering Software , (1987) , vol.9 , pp.34-55
30. Lawson C.L. *Software for C^1 Surface Interpolation*
31. Bonet J. , Peraire J. *An Alternating Digital Tree (ADT) Algorithm for 3D Geometric Searching and Intersection Problems* Int. J. Numer. Meth. in Eng , (1991) , vol.31 , pp.1-17
32. Hackbusch W. *Iterative Solution of Large Sparse Systems of Equations* Springer-Verlag NewYork , (1994)
33. Ajiz M.A. , Jennings A. *A Robust Incomplete Cholesky-Conjugate Gradient Algorithm* Int. J. Numer. Meth. in Eng , (1984) , vol.20 , pp.949-966
34. Schmit Jr. A. L. , Lai Y.C. *Structural Optimization Based on Preconditioned Conjugate Gradient Analysis Methods* Int. J. Numer. Meth. in Eng , (1994) , vol.33 , pp.943-964
35. Dickinson J.K. , Forsyth P.A. *Preconditioned Conjugate Gradient Methods for Three-Dimensional Linear Elasticity* Int. J. Numer. Meth. in Eng , (1994) , vol.37 , pp.2211-2234
36. Baigakov V.E. , Kuhn G. *High-Performance Multilevel Iterative Aggregation Solver for Large Finite-Element Structural Analysis Problems* Int. J. Numer. Meth. in Eng , (1985) , vol.38 , pp.3529-3544
37. St Georges P. , Warzee G. , Beauwens R. , Notay Y. *High-Performance PCG Solvers for FEM Structural Analysis* Int. J. Numer. Meth. in Eng , (1996) , vol.39 , pp.1313-1340
38. Fish J. , Guttai R. *Adaptive Solver for the P-Version of Finite Element Method* Int. J. Numer. Meth. in Eng , (1997) , vol.40 , pp.1767-1784
39. Hladik L. , Reed M.B. , Swoboda G. *Robust Preconditioners for Linear Elasticity FEM Analysis* Int. J. Numer. Meth. in Eng , (1997) , vol.40 , pp.2109-2127
40. Graham E. , Forsyth P.A. *Preconditioning Methods for Very Ill-Conditioned Three-Dimensional Analysis* Int. J. Numer. Meth. in Eng , (1999) , vol.44 , pp.77-99
41. Joun M. S. , Lee M. C. *Quadrilateral Finite-Element Generation and Mesh Quality Control for Metal Forming Simulation* Int. J. Numer. Meth. in Eng , (1997) , vol.40 , pp.4059-4075
42. Lee C. K. , Hobbs R. E. *On Using Different Finite Elements with an Automatic Adaptive Refinement Procedure for the Solution of 2-D Stress Analysis Problem* Int. J. Numer. Meth. In Eng , (1997) , vol.40 , pp.4547-4576
43. Blacker T. D. , Stephenson M. B. *Paving: A New Approach to Automated Quadrilateral Mesh Generation* Int. J. Numer. Meth. in Eng , (1991) , vol.32 , pp.811-847
44. Khamanysch A. , Kuprat A. *Anisotropic Smoothing and Solution Adaption for Unstructured Grids* Int. J. Numer. Meth. in Eng , (1996) , vol.39 , pp.3163-3174



ภาคผนวก

ภาคผนวก ก

การใช้งานโปรแกรม

เนื่องจากโปรแกรมมีการทำงานโดยอาศัยการนำเข้าข้อมูลจากแฟ้มข้อมูลเป็นหลัก ดังนั้นแฟ้มข้อมูลจึงเป็นส่วนที่สำคัญที่สุดสำหรับผู้ใช้งาน ซึ่งมีรายละเอียดของแฟ้มดังนี้ โดยยกตัวอย่างแฟ้มจากการวิเคราะห์ปัญหาในตัวอย่างที่ 1 ในบทที่ 4 ดังแสดงในรูปที่ ผ.ก.1 โดยเพิ่มหมายเลขบรรทัดเพื่อความสะดวกในการอ้างอิง

ในส่วนแรกของแฟ้มบรรทัดที่ 1 - 6 จะเป็นการกำหนดลักษณะของปัญหา ความคลาดเคลื่อนสัมพัทธ์ที่ยอมรับ และคุณสมบัติทางวัสดุของปัญหาซึ่งสามารถเข้าใจได้โดยง่าย ลำดับต่อมาในบรรทัดที่ 8 จะเป็นพิกัดที่มุมล่างซ้ายและมุมบนขวาของพื้นที่ที่จะกำหนดให้ ADT เพื่อใช้แบ่งให้กับแต่ละจุดต่อไป ข้อสังเกตคือที่มุมบนขวาดังกล่าวอยู่เลยตำแหน่งบนขวาสุดของปัญหาออกไป ในขณะที่ที่มุมล่างซ้ายสามารถกำหนดให้ซ้อนทับกับตำแหน่งล่างซ้ายของปัญหาได้

ส่วนต่อมาในบรรทัดที่ 10 เป็นจำนวนของขอบเขตปัญหาและในบรรทัดที่ 12 - 18 เป็นข้อมูลของแต่ละขอบเขตโดยมีข้อมูลเรียงตามลำดับดังนี้ 1.หมายเลขประจำขอบเขตโดยต้องเริ่มจาก 0 ไล่ตามกันไป 2. ค่าพิกัดในแนวแกน x ของจุดเริ่มต้น 3. ค่าพิกัดในแนวแกน y ของจุดเริ่มต้น 4. ค่าพิกัดในแนวแกน x ของจุดปลาย 5. ค่าพิกัดในแนวแกน y ของจุดปลาย 6. รัศมีของเส้นขอบเขต มีค่าเป็นบวกเมื่อจุดศูนย์กลางอยู่ทางด้านซ้ายของเส้น มีค่าเป็นลบเมื่อจุดศูนย์กลางอยู่ทางด้านขวาของเส้น และจะมีค่า 0 ถ้าเป็นเส้นตรง 7. จำนวนขอบเขตย่อยที่จะแบ่งจากขอบเขตนั้น ๆ

ต่อมาเป็นข้อมูลของที่รองรับโดยในบรรทัดที่ 20 เป็นจำนวนของที่รองรับ และมีรายละเอียดของที่รองรับในบรรทัดที่ 22 - 25 โดยมีข้อมูลของที่รองรับเรียงตามลำดับดังนี้ 1. เป็นหมายเลขเส้นขอบเขตที่จะอ้างอิงถึง 2. ชนิดของที่รองรับมีค่า p ถ้ามีการรองรับเฉพาะที่จุดต้นของขอบเขต และมีค่าเป็น 1 ถ้ามีการรองรับตลอดทั้งเส้นขอบเขต 3. การรองรับในทิศทาง x มีค่าเป็น 1 ถ้าไม่อนุญาตให้มีการเคลื่อนที่ และมีค่าเป็น 0 ถ้าอนุญาตให้เคลื่อนที่ในแนวแกน x ได้ 4. การรองรับในทิศทาง y มีค่าเป็น 1 ถ้าไม่อนุญาตให้มีการเคลื่อนที่ และมีค่าเป็น 0 ถ้าอนุญาตให้เคลื่อนที่ได้ในแนวแกน y

ต่อมาเป็นข้อมูลของแรงกระทำโดยในบรรทัดที่ 27 เป็นจำนวนของแรงกระทำ และบรรทัดที่ 29 เป็นรายละเอียดของแรงกระทำโดยมีข้อมูลเรียงตามลำดับดังนี้ 1. หมายเลขของขอบเขตที่อ้างอิงถึง 2. ชนิดของแรงกระทำ มีค่าเป็น p ถ้าแรงกระทำเป็นจุดกระทำเฉพาะที่จุดต้นของขอบเขต และมีค่าเป็น 1 ถ้าแรงกระทำเป็นแรงแผ่กระจายตลอดทั้งขอบเขต 3. และ 4. เป็นขนาดของแรงกระทำในทิศทาง x และ y ที่จุดต้นของขอบเขต 5. และ 6. เป็นขนาดของแรงกระทำในทิศทาง x และ y ที่จุดปลายของขอบเขต ถ้าชนิดของข้อมูลเป็น p ข้อมูลที่ 5 และ 6 ต้องไม่มี

ในส่วนสุดท้ายของแฟ้มในบรรทัดที่ 31 เป็นข้อมูลสำหรับการแสดงผลทางกราฟฟิกโดยกำหนดขนาดของตัวหนังสือและมาตราส่วนในการแสดงผลการเปลี่ยนตำแหน่ง

ในการใช้งาน โปรแกรมจะให้ป้อนชื่อแฟ้มข้อมูลสำหรับนำเข้า และหลังจากที่ทำการวิเคราะห์แล้ว โปรแกรมก็จะให้ป้อนชื่อแฟ้มส่งออก โดยในแต่ละรอบของโครงข่ายชิ้นส่วนจะส่งออกแฟ้มข้อมูล 3 แฟ้มคือ 1. แฟ้มข้อมูลทางเรขาคณิตของจุดและชิ้นส่วน 2. แฟ้มข้อมูลแสดงผลการวิเคราะห์ของแต่ละจุด 3. แฟ้มแลกเปลี่ยนข้อมูลทางด้านกราฟิก สำหรับการให้ข้อมูลใน 2 แฟ้มแรกสามารถเปิดใช้ได้ด้วยโปรแกรมแก้ไขค่าทั่วไป ส่วนแฟ้มข้อมูลที่ 3. สามารถนำเข้าแสดงผลในโปรแกรม CAD ทั่วไป เช่นถ้าใช้โปรแกรม AutoCAD ก็ให้นำเข้าแฟ้มข้อมูลโดยใช้คำสั่ง dxflg จากนั้นก็สามารถจัดการกับข้อมูลได้เหมือนกับข้อมูลที่เขียนขึ้นจาก AutoCAD ทุกประการ

```

1 Plane type<plane stress=0 , plane strain=1>
2 0
3 Expected Error
4 0.01
5 Material properties <E nu thickness >
6 2e6 0.3 0.1
7 Limit of search <lowerleft (x,y) upperright (x,y)>
8 0 0 10.01 20.01
9 Number of boundary
10 7
11 BoundaryNo. (x1,y1) (x2,y2) radius segment
12 0 0 20 0 5 0 4
13 1 0 5 0 2 0 4
14 2 0 2 2 0 -2 4
15 3 2 0 5 0 0 4
16 4 5 0 10 0 0 2
17 5 10 0 10 20 0 4
18 6 10 20 0 20 0 2
19 Number of support
20 4
21 BoundaryNo. (Point/Line<p/l>) ( x,y <fix/free:1/0>)
22 0 1 1 0
23 1 1 1 0
24 3 1 0 1
25 4 1 0 1
26 Number of load
27 1
28 BoundaryNo. (Point/Line) (x1,y1) (x2,y2) <if point neglect x2,y2>
29 6 1 0 100 0 100
30 Text size & Displacement scale
31 0.02 50

```

รูปที่ ผ.ก.1 ตัวอย่างแฟ้มนำเข้าข้อมูล

ภาคผนวก ข

รายละเอียดของรหัสคำนวณ

ในการเขียนรหัสคำนวณด้วยภาษา C++ จะแบ่งประเภทแฟ้มข้อมูลเป็น 2 ชนิดคือแฟ้มส่วนหัว มีชื่อแฟ้มลงท้ายด้วย .h และแฟ้มโปรแกรมมีชื่อแฟ้มลงท้ายด้วย .cpp ในงานวิจัยนี้ได้รวมแฟ้มโปรแกรมเข้าในแฟ้มส่วนหัวเพื่อลดการพิมพ์ในการคอมไพล์โปรแกรม ดังนั้นถ้าต้องการความเร็วในการคอมไพล์สามารถแยกส่วนดำเนินการ (implementation) ในแฟ้มส่วนหัวออกเป็นแฟ้มโปรแกรมในชื่อเดียวกันแล้วใช้คำสั่ง `#include "filename"` ในการอ้างอิงถึงแฟ้มส่วนหัว สำหรับรายละเอียดของรหัสคำนวณทั้งหมดได้รวบรวมไว้ในแผ่นดิสเกตประกอบวิทยานิพนธ์



ประวัติผู้เขียน

นาย เทอดเกียรติ จันทวัชรกร เกิดเมื่อวันที่ 8 กันยายน พ.ศ. 2517 ที่จังหวัดสุพรรณบุรี เข้ารับการศึกษา
 ชั้นประถมและมัธยมต้น ที่โรงเรียนสตรีชาติเดช และเบญจมาชชุทิศ จังหวัดจันทบุรี ศึกษาต่อชั้นมัธยมปลายที่โรงเรียน
 เตรียมอุดมศึกษา กรุงเทพฯ สำเร็จการศึกษาปริญญาวิศวกรรมศาสตรบัณฑิต จาก จุฬาลงกรณ์มหาวิทยาลัย ในปีการ
 ศึกษา 2538 และเข้าศึกษาต่อในระดับปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมโยธา จุฬาลงกรณ์
 มหาวิทยาลัย ในปีการศึกษา 2540

