

การวิเคราะห์ปฏิสัมพันธ์ระหว่างการไหล คุณทงุมิ และโครงสร้าง  
ด้วยระเบียบวิธีไฟไนต์เอลิเมนต์



นายอติพงษ์ มาลาทิพย์

ศูนย์วิทยพัทพยากร  
วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต

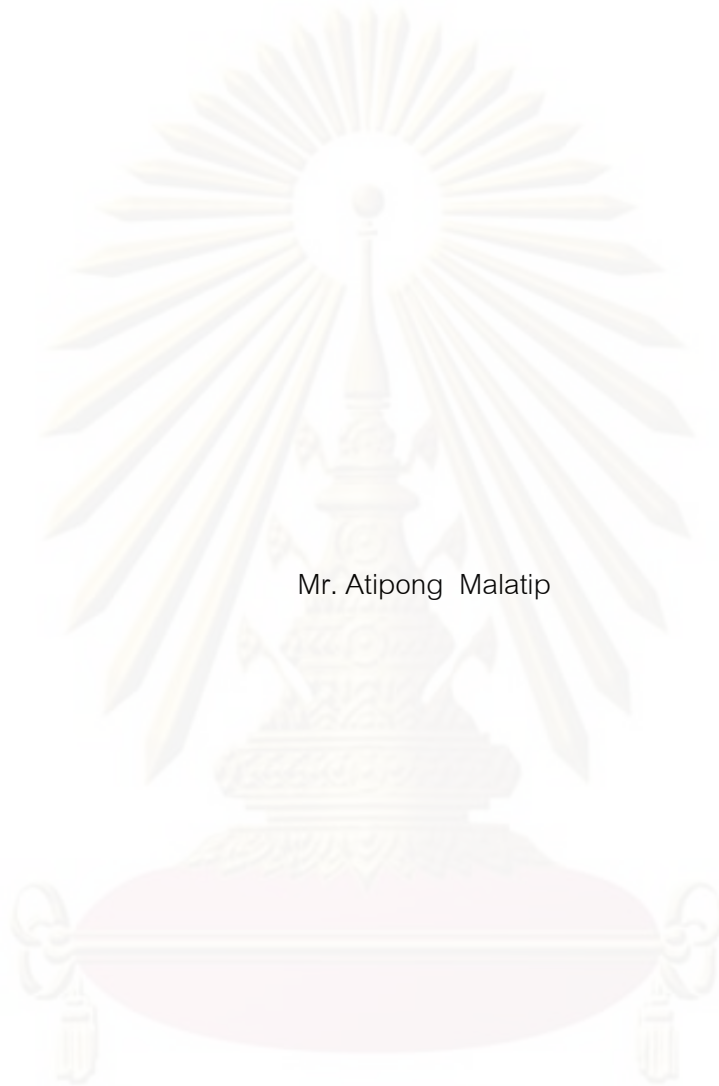
สาขาวิชาวิศวกรรมเครื่องกล ภาควิชาวิศวกรรมเครื่องกล

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2552

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

COMBINED FLUID-THERMAL-STRUCTURAL ANALYSIS BY  
FINITE ELEMENT METHOD



Mr. Atipong Malatip

A Dissertation Submitted in Partial Fulfillment of the Requirements  
for the Degree of Doctor of Philosophy Program in Mechanical Engineering

Department of Mechanical Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2009

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์

การวิเคราะห์ปฏิสัมพันธ์ระหว่างการไหล อุณหภูมิ และ  
โครงสร้างด้วยระเบียบวิธีไฟไนต์เอลิเมนต์

โดย

นายอิทธิพงษ์ มาลาทิพย์

สาขาวิชา

วิศวกรรมเครื่องกล

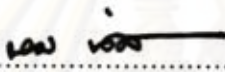
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

ศาสตราจารย์ ดร.ปราโมทย์ เตชะอำไพ

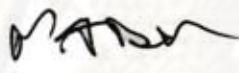
อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

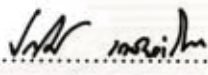
ผู้ช่วยศาสตราจารย์ ดร.นิพนธ์ วรรณโสภาคย์

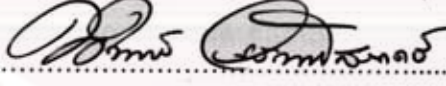
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วน  
หนึ่งของการศึกษาตามหลักสูตรปริญญาตรีบัณฑิต

  
.....คนบดีคณะวิศวกรรมศาสตร์  
(รองศาสตราจารย์ ดร.บุญสม เลิศนันทวงค์)

คณะกรรมการสอบวิทยานิพนธ์

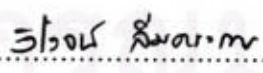
  
..... ประธานกรรมการ  
(ผู้ช่วยศาสตราจารย์ ดร.ตุลย์ มณีวัฒนา)

  
..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก  
(ศาสตราจารย์ ดร.ปราโมทย์ เตชะอำไพ)

  
..... อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม  
(ผู้ช่วยศาสตราจารย์ ดร.นิพนธ์ วรรณโสภาคย์)

  
..... กรรมการ  
(อาจารย์ ดร.ชญญาพันธ์ วิรุฬห์ศรี)

  
..... กรรมการภายนอกมหาวิทยาลัย  
(รองศาสตราจารย์ ดร.วรางค์รัตน์ จันทร์สาโร)

  
..... กรรมการภายนอกมหาวิทยาลัย  
(รองศาสตราจารย์ ดร.วิโรจน์ ลิ้มตระการ)

อธิพงษ์ มาลาทิพย์ : การวิเคราะห์ปฏิสัมพันธ์ระหว่างการไหล อุณหภูมิ และโครงสร้าง ด้วยระเบียบวิธีไฟไนต์เอลิเมนต์. (COMBINED FLUID-THERMAL-STRUCTURAL ANALYSIS BY FINITE ELEMENT METHOD) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ศ. ดร. ปราโมทย์ เดชะอำไพ, อ.ที่ปรึกษาวิทยานิพนธ์ร่วม: ผศ. ดร.นิพนธ์ วรรณโสภาคย์, 190 หน้า.

วิทยานิพนธ์ฉบับนี้นำเสนอระเบียบวิธีสตรูมไลน์อัปวินด์ เพทรอฟ-กาเลอร์คินร่วมกับวิธีการคำนวณแบบแยกส่วน 4 ชั้นตอนเพื่อแก้ปัญหาพร้อมกันระหว่างการไหล อุณหภูมิ และโครงสร้าง โดยเป็นการคำนวณการนำความร้อนในของแข็งและการพาความร้อนในของไหลควบคู่กัน อีกทั้งนำผลที่เกิดขึ้นไปคำนวณหาค่าความเค้นในของแข็งด้วย โดยในส่วนของ การไหล จะวิเคราะห์ด้วยวิธีสตรูมไลน์อัปวินด์ เพทรอฟ-กาเลอร์คินและใช้วิธีกาเลอร์คินในการวิเคราะห์การถ่ายเทความร้อนและความเค้นที่เกิดขึ้นในของแข็ง ในวิธีดังกล่าวได้ใช้เอลิเมนต์สามเหลี่ยมแบบสามจุดต่อกับใช้ฟังก์ชันการประมาณภายในสำหรับความเร็ว ความดันและอุณหภูมิที่อันดับเท่ากันได้ เป็นผลให้สามารถทำความเข้าใจในขั้นตอนของการประดิษฐ์สมการไฟไนต์เอลิเมนต์ได้โดยง่าย อีกทั้งยังเพิ่มประสิทธิภาพของการคำนวณด้วยการคำนวณแบบแยกกันกล่าวคือตัวแปรความเร็ว ความดันและอุณหภูมิจะถูกคำนวณไม่พร้อมกัน

การตรวจสอบความถูกต้องของโปรแกรมคอมพิวเตอร์ทำโดยการนำผลลัพธ์ที่ได้จากการวิเคราะห์ไปเปรียบเทียบกับปัญหาที่มีผลเฉลยแม่นยำตรง ก่อนนำโปรแกรมคอมพิวเตอร์นี้ไปใช้ในการวิเคราะห์ปัญหาที่มีความซับซ้อนมากยิ่งขึ้น ผลลัพธ์ที่ได้จากการวิเคราะห์ปัญหาในวิทยานิพนธ์นี้แสดงให้เห็นถึงประสิทธิภาพของการประยุกต์ใช้ระเบียบวิธีไฟไนต์เอลิเมนต์ในการวิเคราะห์ปัญหาพร้อมกันระหว่างการไหล อุณหภูมิ และโครงสร้าง

ภาควิชา.....วิศวกรรมเครื่องกล.....  
สาขาวิชา.....วิศวกรรมเครื่องกล.....  
ปีการศึกษา...2552

ลายมือชื่อนิสิต อธิพงษ์ มาลาทิพย์  
ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก [Signature]  
ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์ร่วม [Signature]

## 4871846121 : MAJOR MECHANICAL ENGINEERING

KEYWORDS : FRACTIONAL STEP METHOD / FLUID-THERMAL-STRUCTURAL ANALYSIS / SUPG / FINITE ELEMENT METHOD

ATIPONG MALATIP : COMBINED FLUID-THERMAL-STRUCTURAL ANALYSIS BY FINITE ELEMENT METHOD. THESIS ADVISOR : PROF. PRAMOTE DECHAUMPHAI, Ph.D., THESIS CO-ADVISOR : ASST. PROF. NIPHON WANSOPHARK, Ph.D., 190 pp.

A combined streamline upwind Petrov-Galerkin method (SUPG) and fractional four-step algorithm are developed. An integrated fluid-thermal-structural analysis is presented, where heat conduction in a solid is coupled with heat convection in viscous fluid flow inducing the stress in solid. The streamline upwind Petrov-Galerkin method is used for the analysis of viscous thermal flow in the fluid region, while the analyses of heat conduction and thermal stress in the solid region are performed by the Galerkin method. The method uses the three-node triangular element with equal-order interpolation functions for all the variables of the velocity components, pressure, temperature and displacement components to reduce the complexity in deriving the corresponding finite element equations. A fractional four-step solution algorithm is also incorporated to compute the velocities, pressure and temperature separately for improving the computational efficiency.

A corresponding finite element computer program was developed and verified using simple examples that have exact solutions before applying to solve more complex problems. The computational results from several tested problems illustrate the effectiveness of the presented finite element method that can accurately predict the integrated fluid-thermal-structural phenomena.

Department : Mechanical Engineering...	Student's Signature	<i>Atipong Malatip</i>
Field of Study : Mechanical Engineering...	Advisor's Signature	<i>Prate Dechaumphai</i>
Academic Year : 2009.....	Co-Advisor's Signature	<i>Nipphon Wansophark</i>

## กิตติกรรมประกาศ

ผู้วิจัยขอกราบขอบพระคุณ ศาสตราจารย์ ดร.ปราโมทย์ เดชะอำไพ อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก และผู้ช่วยศาสตราจารย์ ดร.นิพนธ์ วรรณโสภาคย์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วมเป็นอย่างสูง ที่ท่านได้ให้ความรู้ คำแนะนำ ตลอดจนข้อคิดที่มีคุณค่ายิ่งในการทำวิจัย นอกจากนี้ท่านยังได้ถ่ายทอดข้อคิดหลายสิ่งหลายอย่างที่มีคุณค่ายิ่งเกี่ยวกับการทำงานและการดำเนินชีวิตของผู้วิจัย

ขอกราบขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.ตุลย์ มณีวัฒนา ประธานกรรมการรองศาสตราจารย์ ดร.วรางศรัตน์ จันทสาโร รองศาสตราจารย์ ดร.วิโรจน์ ลิ้มตระการ และอาจารย์ ดร.ชัญญาพันธ์ วิรุพหิศรี กรรมการ ที่ได้ให้คำแนะนำและถ่ายทอดความรู้ตลอดระยะเวลาในการทำงานวิจัยนี้ ซึ่งทำให้วิทยานิพนธ์ฉบับนี้มีความสมบูรณ์มากยิ่งขึ้น

ขอขอบพระคุณทุน 90 ปี จุฬาลงกรณ์มหาวิทยาลัย ที่ให้การสนับสนุนทางการเงินตลอดระยะเวลาที่ทำงานวิจัยฉบับนี้

ขอขอบพระคุณ อาจารย์ ดร.สุทธศักดิ์ พงศ์ธนาพานิช ที่ได้ถ่ายทอดความรู้ คอยให้คำปรึกษา และประสพการณ์ในทุก ๆ ด้าน ขอขอบคุณ คุณสุธี ไตรวิวัฒนา คุณปริญญา บุญมาเลิศ คุณพัชรี ธีระเอก และคุณพิชฌันทร์ โพธิกุล ซึ่งเป็นผู้ร่วมงานในห้องปฏิบัติการวิจัยกลศาสตร์การคำนวณ สำหรับความช่วยเหลือและกำลังใจตลอดเวลาทำงานวิจัยนี้

ท้ายที่สุดนี้ผู้วิจัยขอกราบขอบพระคุณบิดามารดาที่เป็นกำลังใจและสนับสนุนการศึกษาของผู้วิจัยมาโดยตลอด และคุณปรีชญา มงคลหัตถิ ผู้ที่คอยให้กำลังใจแก่ข้าพเจ้าในการทำงานวิจัยจนสำเร็จลุล่วง อนึ่งประโยชน์และคุณค่าอันใดที่ได้รับจากวิทยานิพนธ์นี้ขอมอบเป็นกตัญญูตานุชาแต่บิดามารดา ครูอาจารย์ ตลอดจนผู้มีพระคุณทุกท่าน

ศูนย์วิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ฎ
สารบัญภาพ.....	ฏ
คำอธิบายสัญลักษณ์และคำย่อ.....	ถ
<b>บทที่ 1 บทนำ.....</b>	<b>1</b>
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	3
1.3 ขอบเขตของการวิจัย.....	3
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	4
1.5 วิธีดำเนินการวิจัย.....	4
1.6 เอกสารและงานวิจัยที่เกี่ยวข้อง.....	5
<b>บทที่ 2 สมการเชิงอนุพันธ์ย่อยสำหรับการไหล การถ่ายเทความร้อน และของแข็ง.....</b>	<b>8</b>
2.1 สมการเชิงอนุพันธ์ย่อยสำหรับการไหล.....	8
2.2 สมการเชิงอนุพันธ์ย่อยสำหรับการถ่ายเทความร้อน.....	9
2.3 สมการเชิงอนุพันธ์ย่อยสำหรับปัญหาของแข็ง.....	11
2.4 ระเบียบวิธีไฟไนต์เอลิเมนต์.....	13
2.4.1 ขั้นตอนทั่วไปของระเบียบวิธีไฟไนต์เอลิเมนต์.....	13
<b>บทที่ 3 ระเบียบวิธีไฟไนต์เอลิเมนต์สำหรับการวิเคราะห์ร่วมกันระหว่าง การไหล อุณหภูมิ และโครงสร้าง.....</b>	<b>17</b>
3.1 ระเบียบวิธีไฟไนต์เอลิเมนต์สำหรับปัญหาการไหลแบบหนืด.....	17
3.1.1 ระเบียบวิธีการแบ่งย่อยช่วงเวลา.....	17

3.1.2	วิธีการคำนวณแบบแยกส่วน.....	19
3.1.3	ระเบียบวิธีสตรึมไลน์อัปวินด์เพทروف-กาเลอร์คิน.....	22
3.1.4	ระเบียบวิธีถ่วงน้ำหนักเศษตักข้าง.....	23
3.2	สมการไฟไนต์เอลิเมนต์สำหรับปัญหาการถ่ายเทความร้อน ทั้งภายในของไหลและโครงสร้าง.....	29
3.3	สมการไฟไนต์เอลิเมนต์สำหรับปัญหาความเค้นในโครงสร้างอัน เนื่องมาจากอุณหภูมิ.....	29
3.4	การประดิษฐ์ไฟไนต์เอลิเมนต์เมตริกซ์.....	32
3.4.1	เอลิเมนต์เมตริกซ์สำหรับปัญหาการไหลแบบหนืด.....	33
3.4.2	เอลิเมนต์เมตริกซ์สำหรับวิเคราะห์ปัญหาความเค้นในโครงสร้าง อันเนื่องมาจากอุณหภูมิ.....	48
<b>บทที่ 4</b>	<b>โปรแกรมคอมพิวเตอร์.....</b>	<b>50</b>
4.1	โปรแกรมคอมพิวเตอร์สำหรับวิเคราะห์ปัญหาการไหลแบบหนืดชนิดอัด ตัวไม่ได้ที่มีการถ่ายเทความร้อนระหว่างของไหลและโครงสร้างทั้งสภาวะ อยู่ตัวและไม่อยู่ตัว.....	50
4.2	รายละเอียดของโปรแกรม.....	52
4.3	ลักษณะของไฟล์ข้อมูลที่โปรแกรมต้องการ.....	52
4.4	โปรแกรมคอมพิวเตอร์สำหรับวิเคราะห์ปัญหาความเค้น อันเนื่องมาจากอุณหภูมิ.....	56
4.5	รายละเอียดของโปรแกรม.....	57
4.6	ลักษณะของไฟล์ข้อมูลที่โปรแกรมต้องการ.....	57
<b>บทที่ 5</b>	<b>การตรวจสอบความถูกต้องของโปรแกรมคอมพิวเตอร์.....</b>	<b>60</b>
5.1	ตรวจสอบโปรแกรมสำหรับวิเคราะห์ปัญหาการไหลแบบหนืดแบบไม่อัด ตัวโดยมีการถ่ายเทความร้อนระหว่างของไหลและโครงสร้างที่สภาวะอยู่ ตัวและไม่อยู่ตัว.....	60
5.1.1	ปัญหาการไหลระหว่างแผ่นคู่ขนานที่สภาวะไม่อยู่ตัว.....	61
5.1.2	ปัญหาการไหลหมุนวนในช่องสี่เหลี่ยม.....	62



5.1.3	ปัญหาการไหลเนื่องจากการพาความร้อนแบบอิสระในช่องปิดรูปสี่เหลี่ยมจัตุรัส.....	66
5.1.4	ปัญหาการถ่ายเทความร้อนผ่านทรงกระบอกร้อน.....	71
5.1.5	ปัญหาการไหลระหว่างแผ่นคู่ขนานพร้อมการถ่ายเทความร้อนกับโครงสร้าง.....	75
5.16	ปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน.....	78
5.2	ตรวจสอบโปรแกรมสำหรับวิเคราะห์ปัญหาความเค้นอันเนื่องมาจากอุณหภูมิ.....	81
5.2.1	ปัญหาแผ่นสี่เหลี่ยมที่มีอุณหภูมิเปลี่ยนแปลงเชิงเส้นตามความยาว.....	81
5.2.2	ปัญหาวงแหวนที่มีอุณหภูมิเปลี่ยนแปลงตามแนวรัศมี.....	83
<b>บทที่ 6</b>	<b>การวิเคราะห์ปัญหาร่วมกันระหว่างการไหล อุณหภูมิและโครงสร้าง.....</b>	<b>87</b>
6.1	ปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน.....	87
6.2	ปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสโดยที่มีผนังนำความร้อน.....	92
6.3	ปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายในช่องทางการไหล.....	109
6.4	ปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีครีบริบายความร้อน.....	114
<b>บทที่ 7</b>	<b>บทสรุป ปัญหาที่พบและข้อเสนอแนะ.....</b>	<b>123</b>
7.1	บทสรุป.....	123
7.2	ปัญหาที่พบในขณะที่ทำวิทยานิพนธ์.....	125
7.3	ข้อเสนอแนะสำหรับงานวิจัยในอนาคต.....	125
	<b>รายการอ้างอิง.....</b>	<b>126</b>
	<b>ภาคผนวก.....</b>	<b>130</b>

ภาคผนวก ก	รายละเอียดของโปรแกรมวิเคราะห์ปัญหาการไหลและการ ถ่ายเทความร้อน.....	131
ภาคผนวก ข	รายละเอียดของโปรแกรมวิเคราะห์ปัญหาความเค้นอันเนื่องมา จากอุณหภูมิ.....	173
ประวัติผู้เขียนวิทยานิพนธ์.....		190



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

## สารบัญตาราง

ตารางที่		หน้า
5.1	การเปรียบเทียบค่าฟังก์ชันการไหลที่ตำแหน่งจุดศูนย์กลางของการหมุนโดย เปรียบเทียบกับวิธีอื่น ๆ.....	66
5.2	การเปรียบเทียบค่านัสเซลล์ทึ่มเบอร์เฉลี่ยที่ผนังร้อนโดยเปรียบเทียบกับวิธีอื่น ๆ.....	71
6.1	ค่าการเคลื่อนตัวสูงสุดในแนวตั้งของของแข็งสำหรับปัญหาการไหลแบบสวน ทางภายในอุปกรณ์แลกเปลี่ยนความร้อน กรณีค่า $K_{sf}$ แตกต่างกัน.....	91
6.2	การเปรียบเทียบค่านัสเซลล์ทึ่มเบอร์เฉลี่ยที่บริเวณผิวรอยต่อ.....	103
6.3	ค่าการเคลื่อนตัวสูงสุดในแนวนอนของของแข็งสำหรับปัญหาการพาความร้อน แบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัส กรณีค่า $K_{sf}$ แตกต่างกัน.....	109
6.4	แสดงค่าความเค้นวอนมิสเซสสูงสุดและค่าการเคลื่อนตัวสูงสุดบนครีระบาย ความร้อนที่ค่า $Re$ แตกต่างกัน.....	114
6.5	แสดงค่าความเค้นวอนมิสเซสสูงสุดและค่าการเคลื่อนตัวสูงสุดบนครีระบาย ความร้อนที่ค่า $K_{sf}$ แตกต่างกัน.....	121

## สารบัญภาพ

ภาพที่		หน้า
2.1	ขอบเขตและเงื่อนไขขอบเขตของการไหลแบบหนืด.....	9
2.2	ขอบเขตและเงื่อนไขขอบเขตของปัญหาการถ่ายเทความร้อน.....	10
2.3	ขอบเขตและเงื่อนไขขอบเขตของปัญหาของแข็ง.....	12
2.4	การแบ่งขอบเขตรูปร่างของปัญหาออกเป็นเอลิเมนต์ย่อย ๆ.....	14
2.5	เอลิเมนต์สามเหลี่ยมแบบ 3 จุดต่อและตัวไม่ทราบค่าที่จุดต่อ.....	14
3.1	ความสัมพันธ์ของตัวแปรสเกลาร์ในช่วงเวลา $t^n \rightarrow t^{n+1}$ .....	18
3.2	ขนาดของความสูงของเอลิเมนต์สามเหลี่ยมแบบสามจุดต่อ.....	22
3.3	การแบ่งลักษณะของปัญหาออกเป็นเอลิเมนต์สามเหลี่ยมแบบสามจุดต่อ.....	23
3.4	เอลิเมนต์สามเหลี่ยมบนที่วางตัวอยู่ในโคออร์ดิเนต x-y.....	33
3.5	การคำนวณพจน์ $\{R_{bi}\}$ ที่บริเวณขอบของโดเมนการไหล.....	43
5.1	ลักษณะของปัญหาการไหลแบบหนืดระหว่างแผ่นคู่ขนาน.....	61
5.2	เปรียบเทียบการกระจายตัวของความเร็วที่ได้จากการคำนวณกับผลเฉลยแม่นยำ ตรงของปัญหาการไหลระหว่างแผ่นคู่ขนานที่ $Re = 100$ ณ เวลาต่าง ๆ.....	62
5.3	ลักษณะของปัญหาการไหลหมุนวนภายในช่องแคบ.....	63
5.4	แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการไหลหมุนวนภายในช่องแคบและ เงื่อนไขขอบเขต.....	63
5.5	สภาวะการไหลสำหรับปัญหาการไหลหมุนวนภายในช่องแคบที่ค่าเรย์โนลด์ เท่ากับ 100.....	64
5.6	การเปรียบเทียบการกระจายตัวของความเร็วที่ค่าเรย์โนลด์เท่ากับ 100 ณ เวลา ต่าง ๆ.....	64
5.7	แสดงเส้นกระแสการไหล เส้นชั้นของความดันและแสดงลักษณะการกระจาย ตัวของความเร็วที่ค่าเรย์โนลด์เท่ากับ (ก) 1,000 (ข) 5,000 และ (ค) 10,000 ตามลำดับ.....	65
5.8	ปัญหาการไหลเนื่องจากการพาความร้อนในช่องสี่เหลี่ยมจัตุรัส.....	67
5.9	การเปรียบเทียบลักษณะการกระจายของความเร็วและอุณหภูมิไร้มิติตลอด แนวแกน x ที่ค่าเรย์เลห์นัมเบอร์เท่ากับ $10^3$ ณ เวลาต่าง ๆ.....	68

ภาพที่	หน้า	
5.10	การเปรียบเทียบลักษณะการกระจายของความเร็วและอุณหภูมิไร้มิติตลอด แนวแกน $x$ ที่ค่าเรย์เลห์นัมเบอร์เท่ากับ $10^4$ ณ เวลาต่าง ๆ.....	69
5.11	การเปรียบเทียบลักษณะการกระจายของความเร็วและอุณหภูมิไร้มิติตลอด แนวแกน $x$ ที่ค่าเรย์เลห์นัมเบอร์เท่ากับ $10^5$ ณ เวลาต่าง ๆ.....	70
5.12	ลักษณะของปัญหาการถ่ายเทความร้อนผ่านทรงกระบอกร้อน.....	72
5.13	แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการถ่ายเทความร้อนผ่านทรงกระบอก ร้อน.....	72
5.14	แสดงเส้นความเร็ว $u$ เส้นกระแสการไหล เส้นชั้นของอุณหภูมิที่เวลาเท่ากับ 10, 20, 30, 50 และ 80.....	73
5.15	การเปรียบเทียบค่าความเร็ว $v$ ที่ตำแหน่ง $Q$ ณ เวลาต่าง ๆ.....	74
5.16	ค่านัสเซลนัมเบอร์ที่ตำแหน่งต่าง ๆ บนผิวของทรงกระบอกร้อน.....	74
5.17	ลักษณะของปัญหาการไหลระหว่างแผ่นคู่ขนานพร้อมการถ่ายเทความร้อนกับ โครงสร้าง.....	75
5.18	แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการไหลระหว่างแผ่นคู่ขนานพร้อมการ ถ่ายเทความร้อนกับโครงสร้าง.....	76
5.19	เวกเตอร์ความเร็วของปัญหาการไหลระหว่างแผ่นคู่ขนานที่มีการถ่ายเทความร้อน กับโครงสร้าง.....	76
5.20	ลักษณะการกระจายของอุณหภูมิของปัญหาการไหลระหว่างแผ่นคู่ขนานพร้อม การถ่ายเทความร้อนกับโครงสร้างที่ค่า $K_{sf}$ ต่าง ๆ.....	77
5.21	การเปรียบเทียบการกระจายตัวของความเร็วและอุณหภูมิที่ได้จากการคำนวณ กับผลเฉลยแม่นยำที่ค่า $K_{sf}$ ต่าง ๆ.....	78
5.22	ลักษณะของปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน..	79
5.23	แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการไหลแบบสวนทางภายในอุปกรณ์ แลกเปลี่ยนความร้อน.....	79
5.24	ลักษณะการกระจายตัวของอุณหภูมิของปัญหาการไหลแบบสวนทางภายใน อุปกรณ์แลกเปลี่ยนความร้อนที่ค่า $K_{sf}$ ต่าง ๆ.....	79
5.25	การเปรียบเทียบค่าอุณหภูมิที่ได้จากการคำนวณกับผลลัพธ์ของ Chen and Han ที่ตำแหน่ง $x = 0.5$ ตลอดแกน $y$ ของปัญหาการไหลแบบสวนทาง ภายในอุปกรณ์แลกเปลี่ยนความร้อนที่ค่า $K_{sf}$ ต่าง ๆ.....	80

ภาพที่	หน้า
5.26	ลักษณะของปัญหาแผ่นสี่เหลี่ยมที่มีอุณหภูมิเปลี่ยนแปลงเชิงเส้นตามความยาว..... 81
5.27	รูปแบบไฟไนต์เอลิเมนต์สำหรับปัญหาแผ่นสี่เหลี่ยมที่มีอุณหภูมิเปลี่ยนแปลงเชิงเส้นตามความยาว..... 82
5.28	ค่าการเคลื่อนตัวเนื่องจากความร้อนในแนวแกน x สำหรับปัญหาแผ่นสี่เหลี่ยมที่มีอุณหภูมิเปลี่ยนแปลงเชิงเส้นตามความยาว..... 82
5.29	ลักษณะของปัญหาวงแหวนที่มีอุณหภูมิเปลี่ยนแปลงตามแนวรัศมี..... 84
5.30	รูปแบบไฟไนต์เอลิเมนต์สำหรับปัญหาวงแหวนที่มีอุณหภูมิเปลี่ยนแปลงตามแนวรัศมี..... 84
5.31	การเปลี่ยนแปลงของค่าการเคลื่อนตัวตามทิศทางรัศมี ในแนวมุม $\theta$ เท่ากับ 0... 85
5.32	ความเค้นในแนวรัศมี $\sigma_r$ ตามทิศทางรัศมีที่มีแนวมุม $\theta$ เท่ากับ 0..... 85
5.33	ความเค้นในทิศ $\theta$ , $\sigma_\theta$ ตามทิศทางรัศมีที่มีแนวมุม $\theta$ เท่ากับ 0..... 86
6.1	รูปแบบไฟไนต์เอลิเมนต์และเงื่อนไขขอบเขตสำหรับวิเคราะห์ความเค้นในปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน..... 88
6.2	ผลการคำนวณสำหรับปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อนในกรณีที่มีค่า $K_{sf} = 1$ ..... 88
6.3	ผลการคำนวณสำหรับปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อนในกรณีที่มีค่า $K_{sf} = 5$ ..... 89
6.4	ผลการคำนวณสำหรับปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อนในกรณีที่มีค่า $K_{sf} = 10$ ..... 90
6.5	ลักษณะของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสโดยที่มีผนังนำความร้อน..... 92
6.6	แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสโดยที่มีผนังนำความร้อน..... 93
6.7	สภาวะการไหลเนื่องจากการพาความร้อนในช่องสี่เหลี่ยมจัตุรัสเมื่อมีผนังนำความร้อนเมื่อค่า $Gr = 10^3$ ..... 94
6.8	สภาวะการไหลเนื่องจากการพาความร้อนในช่องสี่เหลี่ยมจัตุรัสเมื่อมีผนังนำความร้อนเมื่อค่า $Gr = 10^5$ ..... 95

ภาพที่	หน้า
6.9	สภาวะการไหลเนื่องจากการพาความร้อนในช่องสี่เหลี่ยมจัตุรัสเมื่อมีผนังนำความร้อนเมื่อค่า $Gr = 10^7$ ..... 96
6.10	การเปรียบเทียบลักษณะการกระจายของความเร็วและอุณหภูมิไร้มิติตลอดแนวแกน x ที่ค่า $Gr = 10^3$ ณ เวลาต่าง ๆ..... 98
6.11	การเปรียบเทียบลักษณะการกระจายของความเร็วและอุณหภูมิไร้มิติตลอดแนวแกน x ที่ค่า $Gr = 10^4$ ณ เวลาต่าง ๆ..... 99
6.12	การเปรียบเทียบลักษณะการกระจายของความเร็วและอุณหภูมิไร้มิติตลอดแนวแกน x ที่ค่า $Gr = 10^5$ ณ เวลาต่าง ๆ..... 100
6.13	การเปรียบเทียบค่าอุณหภูมิและปริมาณความร้อนที่ผิวรอยต่อโดยเปรียบเทียบกับผลการคำนวณของ Hriberšek & Kuhn ที่ค่า $K_{sf} = 1, 5$ และ $10$ เมื่อค่า $Gr = 10^5$ ..... 101
6.14	การเปรียบเทียบค่าอุณหภูมิและปริมาณความร้อนที่ผิวรอยต่อโดยเปรียบเทียบกับผลการคำนวณของ Hriberšek & Kuhn ที่ค่า $K_{sf} = 1, 5$ และ $10$ เมื่อค่า $Gr = 10^7$ ..... 102
6.15	แบบจำลองไฟไนต์เอลิเมนต์และเงื่อนไขขอบเขตสำหรับวิเคราะห์ความเค้นในปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสโดยที่มีผนังนำความร้อน..... 104
6.16	การกระจายตัวของอุณหภูมิและค่าความเค้นวอนมิสเสสภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณีนี้ที่ $Gr = 10^3$ และ $K_{sf} = 1$ ..... 104
6.17	การกระจายตัวของอุณหภูมิและค่าความเค้นวอนมิสเสสภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณีนี้ที่ $Gr = 10^3$ และ $K_{sf} = 5$ ..... 105
6.18	การกระจายตัวของอุณหภูมิและค่าความเค้นวอนมิสเสสภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณีนี้ที่ $Gr = 10^3$ และ $K_{sf} = 10$ ..... 105
6.19	การกระจายตัวของอุณหภูมิและค่าความเค้นวอนมิสเสสภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณีนี้ที่ $Gr = 10^5$ และ $K_{sf} = 1$ ..... 106

ภาพที่	หน้า	
6.20	การกระจายตัวของอุณหภูมิและค่าความเค้นวอนมิสเซสภายในของแข็งของ ปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี $Gr = 10^5$ และ $K_{sf} = 5$ .....	106
6.21	การกระจายตัวของอุณหภูมิและค่าความเค้นวอนมิสเซสภายในของแข็งของ ปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี $Gr = 10^5$ และ $K_{sf} = 10$ .....	107
6.22	การกระจายตัวของอุณหภูมิและค่าความเค้นวอนมิสเซสภายในของแข็งของ ปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี $Gr = 10^7$ และ $K_{sf} = 1$ .....	107
6.23	การกระจายตัวของอุณหภูมิและค่าความเค้นวอนมิสเซสภายในของแข็งของ ปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี $Gr = 10^7$ และ $K_{sf} = 5$ .....	108
6.24	การกระจายตัวของอุณหภูมิและค่าความเค้นวอนมิสเซสภายในของแข็งของ ปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี $Gr = 10^7$ และ $K_{sf} = 10$ .....	108
6.25	ลักษณะปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายในช่อง ทางการไหล.....	110
6.26	แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวาง ทรงสี่เหลี่ยมภายในช่องทางการไหล.....	110
6.27	สภาวะการไหลของปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยม ภายในช่องทางการไหลที่ค่า $Re = 100$ .....	110
6.28	สภาวะการไหลของปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยม ภายในช่องทางการไหลที่ค่า $Re = 500$ .....	111
6.29	เปรียบเทียบการกระจายตัวของอุณหภูมิที่ผิวของครีบบรรเทาความร้อนทั้งสาม ที่ค่า $Re$ เท่ากับ 100 และ 1,000 กับผลการคำนวณของ Davalath and Bayazitogl.....	112
6.30	รูปแบบไฟไนต์เอลิเมนต์ของครีบบรรเทาความร้อนทั้งสามและเงื่อนไขขอบเขต สำหรับปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายในช่อง ทางการไหล.....	113



ภาพที่	หน้า	
6.31	การกระจายตัวของค่าความเค้นวอนมิสเซส (kPa) บนครีบบระบายความร้อนทั้งสามที่ค่า $Re$ เท่ากับ 100.....	113
6.32	เวกเตอร์แสดงการเคลื่อนตัวของแต่ละจุดบนครีบบระบายความร้อนทั้งสามที่ค่า $Re$ เท่ากับ 100.....	113
6.33	ลักษณะของปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีครีบบระบายความร้อน.....	114
6.34	รูปแบบไฟไนต์เอลิเมนต์ของปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีครีบบระบายความร้อน.....	115
6.35	เส้นกระแสการไหลของปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีครีบบระบายความร้อน.....	115
6.36	เส้นชั้นของอุณหภูมิของปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีครีบบระบายความร้อนโดยที่ค่า $K_{sf}$ ต่าง ๆ.....	116
6.37	เปรียบเทียบการกระจายตัวของอุณหภูมิตลอดผิวด้านบนของครีบบระบายความร้อนทั้งสาม เมื่อเวลาเปลี่ยนแปลงไป ที่ค่า $K_{sf}$ แตกต่างกัน.....	117
6.38	รูปแบบไฟไนต์เอลิเมนต์และเงื่อนไขขอบเขตสำหรับวิเคราะห์การเคลื่อนตัวของปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีครีบบระบายความร้อน.....	119
6.39	แสดงลักษณะการกระจายตัวของอุณหภูมิสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า $K_{sf}$ เท่ากับ 1.....	119
6.40	แสดงการกระจายตัวของค่าความเค้นวอนมิสเซสภายในของแข็งพร้อมกับค่าการเคลื่อนตัวการเสียรูปขนาด 50,000 เท่าสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า $K_{sf}$ เท่ากับ 1.....	120
6.41	แสดงลักษณะการกระจายตัวของอุณหภูมิสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า $K_{sf}$ เท่ากับ 10.....	120
6.42	แสดงการกระจายตัวของค่าความเค้นวอนมิสเซสภายในของแข็งพร้อมกับค่าการเคลื่อนตัวการเสียรูปขนาด 50,000 เท่าสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า $K_{sf}$ เท่ากับ 10.....	120
6.43	แสดงลักษณะการกระจายตัวของอุณหภูมิสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า $K_{sf}$ เท่ากับ 100.....	121

ภาพที่	หน้า
6.44	แสดงการกระจายตัวของค่าความเค้นวอนมิสเซสภายในของแข็งพร้อมกับการเคลื่อนตัวการเสียรูปขนาด 50,000 เท่าสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า $K_{sf}$ เท่ากับ 100..... 121



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

## คำอธิบายสัญลักษณ์และคำย่อ

A	พื้นที่, เมตริกซ์สัมประสิทธิ์
a	ตัวแปรความสูงของช่องการไหล
$a_i$	สัมประสิทธิ์ของฟังก์ชันการประมาณภายใน
B	เมตริกซ์สัมประสิทธิ์
$b_i$	สัมประสิทธิ์ของฟังก์ชันการประมาณภายใน
C	เมตริกซ์สัมประสิทธิ์
c	ค่าความจุความร้อน
$c_i$	สัมประสิทธิ์ของฟังก์ชันการประมาณภายใน
D	ตัวดำเนินการเชิงอนุพันธ์
E	ค่าโมดูลัสของยังส์ (Young's modulus)
F	แรงที่กระทำกับขอบนอกของโครงสร้าง
Gr	กราชอร์ฟนัมเบอร์ (Grashof Number)
$g_i$	ค่าคงที่ของความเร่งเนื่องจากแรงโน้มถ่วง (gravitational acceleration constant)
h	ขนาดโดยของเอลิเมนต์
$K_{sf}$	อัตราส่วนสัมประสิทธิ์การนำความร้อนในของแข็งต่อสัมประสิทธิ์การนำความร้อนในของไหล
k	สัมประสิทธิ์การนำความร้อน
$k_f$	สัมประสิทธิ์การนำความร้อนในของไหล
$k_s$	สัมประสิทธิ์การนำความร้อนในของแข็ง
L	พิกัดธรรมชาติ, ตัวแปรระยะทาง
m	มวล
N	ฟังก์ชันน้ำหนัก

$Nu$	นัสเซิลท์นัมเบอร์ที่ตำแหน่งใด ๆ (local Nusselt number)
$\overline{Nu}$	นัสเซิลท์นัมเบอร์เฉลี่ย (average Nusselt number)
$\hat{n}$	ทิศทางตั้งฉาก
$n_x$	เวกเตอร์หนึ่งหน่วยในทิศทาง $x$
$n_y$	เวกเตอร์หนึ่งหน่วยในทิศทาง $y$
$Pe$	เพกเลตนัมเบอร์ (Peclet number)
$Pr$	พรันด์เทิลนัมเบอร์ (Prandlt number)
$p$	ความดัน
$Q$	อัตราปริมาณความร้อนที่ผลิตได้ต่อหนึ่งหน่วยปริมาตร
$q_i$	ปริมาณความร้อนที่ผ่านขอบในแนวแกน $x$ และ $y$
$q_x$	ปริมาณความร้อนในแนวแกน $x$
$q_y$	ปริมาณความร้อนในแนวแกน $y$
$R$	เศษตกค้าง, โหลดเวกเตอร์
$Ra$	เรย์เลห์นัมเบอร์ (Rayleigh number)
$Re$	เรย์โนลด์นัมเบอร์ (Reynolds number)
$T$	อุณหภูมิ
$T_0$	อุณหภูมิอ้างอิง
$t$	เวลา
$u$	ความเร็วในแนวแกน $x$ หรือ ค่าการเสียรูปในแนวแกน $x$
$u_i$	ความเร็วในแนวแกน $x$ และ $y$
$\hat{u}_i$	ค่าความเร็วชั่วขณะในแนวแกน $x$ และ $y$
$u_i^*$	ค่าความเร็วชั่วขณะในแนวแกน $x$ และ $y$
$v$	ความเร็วในแนวแกน $y$ หรือ ค่าการเสียรูปในแนวแกน $y$
$W$	ฟังก์ชันถ่วงน้ำหนัก

$x$	ระยะในแนวราบ
$x_i$	ระบบพิกัดฉากในแนวแกน $x$ และแกน $y$
$y$	ระยะในแนวตั้ง
$\alpha$	ค่าการการแพร่กระจายความร้อน (thermal diffusivity coefficient) หรือสัมประสิทธิ์การขยายตัวเนื่องจากอุณหภูมิในของแข็ง
$\alpha_f$	ค่าการการแพร่กระจายความร้อนในของไหล
$\alpha_s$	ค่าการการแพร่กระจายความร้อนในของแข็ง
$\alpha_{sf}$	อัตราส่วนของค่าการการแพร่กระจายความร้อนในของแข็งกับของไหล
$\beta$	สัมประสิทธิ์การขยายตัวทางความร้อนในของไหล (coefficient of thermal expansion)
$\Gamma$	ขอบของการไหล
$\gamma$	ความเครียดเฉือน
$\Omega$	โดเมนของการไหล
$\varepsilon$	ค่าความเครียดฉาก (normal strain)
$\phi$	ตัวแปรใด ๆ
$\phi'$	ตัวแปรตามแมนตรง
$\lambda$	ตัวแปรในการปรับขนาดเอลิเมนต์
$\mu$	ความหนืด
$\nu$	อัตราส่วนปัวส์ซง (Poisson's ratio)
$\nu$	สัมประสิทธิ์ความหนืดจลนศาสตร์ (kinematics viscosity)
$\theta$	มุมหรือมุมการไหล
$\rho$	ความหนาแน่น
$\sigma$	ความเค้นในแนวตั้งฉาก
$\tau$	ความเค้นเฉือน

## บทที่ 1

### สมการเชิงอนุพันธ์ย่อยสำหรับการไหล การถ่ายเทความร้อน และของแข็ง

#### 1.1 ความเป็นมาและความสำคัญของปัญหา

การศึกษาปัญหาทางพลศาสตร์ของของไหล (fluid dynamics) เป็นงานวิจัยที่ยังคงได้รับความสนใจอย่างต่อเนื่องจากอดีตจนถึงปัจจุบัน ทั้งด้านการคำนวณ (computation) และด้านการทดลอง (experiment) ซึ่งมีการวิจัยทั้งในหัวข้อของการไหลแบบอัดตัวได้ (compressible flow) และการไหลแบบอัดตัวไม่ได้ (incompressible flow) รวมถึงการไหลแบบราบเรียบ (lamina flow) และการไหลแบบปั่นป่วน (turbulent flow) งานวิจัยในหัวข้อต่าง ๆ เหล่านี้ก่อให้เกิดประโยชน์อย่างมากในการพัฒนางานทางด้านวิศวกรรม

สำหรับงานวิจัยด้านการคำนวณทางพลศาสตร์ของของไหลในหัวข้อการไหลแบบราบเรียบชนิดอัดตัวไม่ได้ นั้น โดยงานวิจัยต้น ๆ จะนำเสนอระเบียบวิธีผลต่างสืบเนื่อง (finite difference method) ซึ่งเป็นระเบียบวิธีที่ประกอบด้วยขั้นตอนที่ไม่ซับซ้อนและง่ายต่อการทำความเข้าใจ แต่อาจก่อให้เกิดความยากลำบากขึ้นได้หากปัญหาที่ทำการวิเคราะห์มีรูปร่างที่ซับซ้อน (complex geometry) เป็นสาเหตุให้ระเบียบวิธีไฟไนต์เอลิเมนต์ (finite element method) ซึ่งสามารถวิเคราะห์ปัญหาที่มีรูปร่างซับซ้อนได้ดี จึงส่งผลให้เป็นหัวข้อในการวิจัยมาอย่างต่อเนื่อง โดยงานวิจัยจำนวนมากจะเสนอวิธีการที่จะแก้สมการนาเวียร์-สโตกส์ (Navier-Stokes) ด้วยระเบียบวิธีการแยกคิด (segregated methods) ซึ่งจะช่วยลดหน่วยความจำของเครื่องคอมพิวเตอร์และเวลาในการคำนวณมากกว่าการแก้ระบบสมการของความเร็วและความดันไปพร้อม ๆ กัน โดยส่วนใหญ่จะแบ่งเป็น 2 แบบ ส่วนแรกเป็นการไหลที่สภาวะอยู่ตัว (steady flow) โดยงานวิจัยจะเสนอวิธีการหาผลลัพท์ของความเร็วและความดันแบบแยกส่วนกัน (velocity-pressure segregated method) [1-4] โดยมีพื้นฐานมาจากการคำนวณแบบ SIMPLE (semi-implicit method for pressure-linked equation) [5] แบบสองเป็นการไหลที่สภาวะไม่อยู่ตัว (unsteady flow) โดยใช้ระเบียบวิธีการคำนวณแบบแยกส่วน (fractional step method) [6-11] โดยทั้งสองแบบที่กล่าวมานั้นจะทำการหาค่าความเร็วที่เหมาะสมได้ จะต้องหาค่าความดันจากสมการที่เกิดจากการรวมกันของสมการอนุรักษ์มวลและสมการอนุรักษ์โมเมนตัม ส่วนวิธีอื่น ๆ ที่สามารถทำนายความถูกต้องในแต่ละช่วงเวลาได้ ยกตัวอย่างเช่น ระเบียบวิธีรุงเงอ-คุททา (Runge-Kutta method) อันดับสามหรือสี่ [9] หรือ ระเบียบวิธีแครงก์-นิโคลสัน (Crank-Nicolson scheme) หรือระเบียบวิธีการทำย้อนกลับอันดับสอง (second order backward method)

ความยุ่งยากซับซ้อนในการนำระเบียบวิธีไฟไนต์เอลิเมนต์มาวิเคราะห์ปัญหาการไหลแบบหนืดแบบไม่อัดตัวในสองมิติในส่วนใหญ่เกิดขึ้นจากพจน์ของอนุพันธ์ย่อยเนื่องจากการพา (convection term) ในระบบสมการนาเวียร์-สโตกส์ ซึ่งอยู่ในรูปแบบไม่เชิงเส้น (nonlinear) นอกจากนี้ยังมีความยุ่งยากจากการที่พจน์ของอนุพันธ์ย่อยที่เกี่ยวข้องกับความเร็วมีอันดับสูงกว่าพจน์ของอนุพันธ์ย่อยที่เกี่ยวข้องกับความดันอยู่หนึ่งอันดับซึ่งหากเลือกใช้เอลิเมนต์ขนาดใหญ่ในบริเวณที่เกิดการเปลี่ยนแปลงของผลลัพธ์สูงหรือเลือกใช้อันดับของฟังก์ชันการประมาณภายในเอลิเมนต์สำหรับความเร็วและความดันที่เท่ากันในระบบสมการนาเวียร์-สโตกส์แล้วจะทำให้ผลลัพธ์ที่ได้เกิดการสั่น (oscillation) และการลู่เข้า (converge) ของคำตอบเป็นไปได้ช้าหรืออาจไม่เกิดการลู่เข้าของคำตอบโดยเฉพาะหากปัญหาที่ทำการวิเคราะห์มีค่าเรย์โนลด์นัมเบอร์ (Reynold number) สูง จากสาเหตุดังกล่าวจึงได้มีการพัฒนาการแก้ปัญหาพจน์การพาในวิธีไฟไนต์เอลิเมนต์นั้นแบ่งได้เป็นสองวิธี วิธีแรกคือวิธีสตรีมไลน์อัปวินด์แบบโดยตรง (direct streamline upwind method) [1-3] วิธีการนี้จะทำการหาทิศทางของกระแสการไหล (streamline direction) และทำการคำนวณพจน์การพาโดยตรงในทิศทางนั้น โดยวิธีดังกล่าวจะให้ความถูกต้องอันดับหนึ่ง (first order accuracy) สำหรับวิธีที่สองคือวิธีสตรีมไลน์อัปวินด์ เพทรอฟ-กาเลอร์คิน (streamline upwind Petrov–Galerkin method, SUPG) [12] วิธีการนี้จะทำการปรับฟังก์ชันถ่วงน้ำหนักเศษตักต่างเพื่อทำการถ่วงน้ำหนักในทิศทางของการไหล โดยวิธีนี้จะให้ความถูกต้องอันดับที่สอง (second order accuracy) โดยทั้งสองวิธีที่กล่าวมานี้จะช่วยลดปัญหาการสั่นของคำตอบได้ อนึ่ง งานวิจัยโดยส่วนใหญ่จะศึกษาพฤติกรรมของของไหลเพียงอย่างเดียว ขณะที่งานวิจัยจำนวนไม่มากนักที่สนใจศึกษาผลกระทบที่เกิดขึ้นระหว่างปัญหาการไหล อุณหภูมิ และโครงสร้างไป พร้อม ๆ กัน การศึกษาผลกระทบที่เกิดขึ้นระหว่างองค์ความรู้ทั้งสามด้านดังกล่าวจะช่วยให้สามารถประยุกต์ใช้งานทางด้านวิศวกรรมที่กว้างขวางมากยิ่งขึ้น

ดังนั้นวิทยานิพนธ์ฉบับนี้จะดำเนินงานวิจัยด้านการคำนวณที่เน้นการทำนายปฏิสัมพันธ์ของการไหลแบบราบเรียบชนิดอัดตัวไม่ได้ ลักษณะการกระจายตัวของอุณหภูมิทั้งในส่วนที่เป็นของไหลและของแข็ง พร้อมทั้งค่าความเค้นและความเครียดที่เกิดขึ้นในโครงสร้างภายใต้อุณหภูมิที่เกิดขึ้นนั้น โดยในส่วนของระเบียบวิธีไฟไนต์เอลิเมนต์ที่ใช้ในการคำนวณพฤติกรรมของการไหลโดยใช้ระเบียบวิธีการคำนวณแบบแยกส่วน (fractional step method) และวิธีการประมาณค่าอันดับสอง ด้วยวิธีแครงก์-นิโคลสัน (Crank-Nicolson scheme) โดยใช้เอลิเมนต์สามเหลี่ยมแบบสามจุดต่อ และใช้ฟังก์ชันถ่วงน้ำหนักด้วยวิธีสตรีมไลน์อัปวินด์ เพทรอฟ-กาเลอร์คิน (SUPG) ลำดับสุดท้ายจะทำการทดสอบกับปัญหาต่าง ๆ ทั้งปัญหาแบบอย่างที่มีผลเฉลยแม่นยำ

ตรง และปัญหาที่มีความซับซ้อนทั้งในด้านของปรากฏการณ์ ของการไหลและรูปร่างของปัญหา (geometry) เพื่อให้มั่นใจว่าเทคนิคที่ถูกนำเสนอในงานวิจัยชิ้นนี้สามารถนำไปใช้วิเคราะห์ปัญหา ในทางปฏิบัติที่มีความซับซ้อนและหลากหลายต่อไป

## 1.2 วัตถุประสงค์ของการวิจัย

- 1.2.1 ประดิษฐ์ระเบียบวิธีการคำนวณแบบแยกส่วนสำหรับเอลิเมนต์สามเหลี่ยมแบบ สามจุดต่อ เพื่อใช้แก้ปัญหาคาร์ไหลแบบราบเรียบชนิดอัดตัวไม่ได้ในสถานะไม่คง ตัว
- 1.2.2 ศึกษาระเบียบวิธีไฟไนต์เอลิเมนต์สำหรับวิเคราะห์ปัญหาการถ่ายเทความร้อนทั้งใน ของไหลและของแข็ง
- 1.2.3 ศึกษาลักษณะความเค้นเนื่องจากความร้อนที่เกิดขึ้นบนโครงสร้าง ในขณะที่ของ ไหลแบบราบเรียบชนิดอัดตัวไม่ได้ไหลผ่านโครงสร้างนั้น
- 1.2.4 ประดิษฐ์โปรแกรมคอมพิวเตอร์ที่สามารถทำนาย 3 ปรากฏการณ์ได้ ดังนี้
  - 1.2.4.1 พฤติกรรมของของไหลแบบราบเรียบชนิดอัดตัวไม่ได้ขณะไหลผ่าน โครงสร้าง
  - 1.2.4.2 ลักษณะการกระจายของอุณหภูมิทั้งในของไหลและบนโครงสร้างขณะที่ มีการถ่ายเทความร้อนกับของไหลที่ไหลผ่าน
  - 1.2.4.3 ลักษณะความเค้นและความเครียดของโครงสร้างขณะมีการถ่ายเทความ ร้อนระหว่างโครงสร้างและของไหลที่ไหลผ่าน

## 1.3 ขอบเขตของการวิจัย

ขอบเขตของการศึกษาที่สอดคล้องกับวัตถุประสงค์ข้างต้น มีดังต่อไปนี้

- 1.3.1 ศึกษาถึงพฤติกรรมของของไหลชนิดอัดตัวไม่ได้ในสถานะไม่คงตัวในสองมิติ
- 1.3.2 ลักษณะการไหลที่พิจารณาเป็นการไหลแบบราบเรียบ



- 1.3.3 พัฒนาระเบียบวิธีการคำนวณแบบแยกส่วนสำหรับเอลิเมนต์สามเหลี่ยมแบบสามจุดต่อ
- 1.3.4 ศึกษาการถ่ายเทความร้อนทั้งในของไหลและโครงสร้างในสองมิติ
- 1.3.5 ศึกษาลักษณะความเค้นเนื่องจากความร้อนในโครงสร้างสองมิติ
- 1.3.6 การเคลื่อนตัวของโครงสร้างจะมีค่าน้อยมากจนไม่ส่งผลกระทบต่อพฤติกรรมของการไหลดังนั้นจะไม่นำมาพิจารณา
- 1.3.7 ความสัมพันธ์ของความเค้นและความเครียดของโครงสร้างมีลักษณะยืดหยุ่นเชิงเส้น

#### 1.4 ประโยชน์ที่ได้รับจากงานวิจัย

- 1.4.1 มีความเข้าใจถึงการใช้ระเบียบวิธีการคำนวณแบบแยกส่วน เพื่อแก้ปัญหาของไหลชนิดอัดตัวไม่ได้ในสภาวะไม่คงตัวในสองมิติ
- 1.4.2 ก่อให้เกิดความรู้ความเข้าใจถึงความสัมพันธ์ระหว่าง 3 สาขาวิชา คือ พลศาสตร์ของไหล การถ่ายเทความร้อน และกลศาสตร์ของแข็ง อันเป็นแนวทางพื้นฐานที่เป็นประโยชน์ต่องานวิจัยระดับสูงต่อไป
- 1.4.3 โปรแกรมคอมพิวเตอร์ที่ประดิษฐ์ขึ้นสามารถทำนายพฤติกรรมการไหลแบบราบเรียบชนิดอัดตัวไม่ได้ การกระจายของอุณหภูมิทั้งในของไหลและของแข็ง และลักษณะของความเค้นและความเครียดอันเนื่องจากความร้อนบนโครงสร้างได้
- 1.4.4 สามารถนำไปสู่การพัฒนาโปรแกรมคอมพิวเตอร์ที่สามารถนำไปใช้แก้ไขปัญหาที่มีรูปร่างหรือเงื่อนไขขอบเขตที่แตกต่างกันได้
- 1.4.5 สามารถนำไปขยายผลเพื่อศึกษาปัญหาการไหลชนิดอื่น ๆ เช่น การไหลแบบปั่นป่วน (turbulent flows) เป็นต้น

#### 1.5 วิธีดำเนินการวิจัย

- 1.5.1 ศึกษาหลักการและทฤษฎีด้านพลศาสตร์ของไหล และระเบียบวิธีเชิงตัวเลขที่เกี่ยวข้องกับงานวิจัยนี้

- 1.5.2 ศึกษาและประดิษฐ์วิธีการคำนวณแบบแยกส่วนสำหรับเอลิเมนต์สามเหลี่ยมแบบสามจุดต่อ
- 1.5.3 ศึกษาลักษณะของของไหลแบบราบเรียบชนิดอัดตัวไม่ได้ในสภาวะไม่คงตัว
- 1.5.4 ศึกษาลักษณะการถ่ายเทความร้อนระหว่างของไหลกับโครงสร้าง
- 1.5.5 ศึกษาลักษณะความเค้นเนื่องมาจากความร้อนที่เกิดขึ้นบนโครงสร้าง
- 1.5.6 ทดสอบความถูกต้องของวิธีที่นำเสนอทั้งหมดข้างต้น ด้วยการเปรียบเทียบผลลัพธ์ที่ได้กับปัญหาที่มีผลเฉลยแม่นยำตรงและปัญหาที่มีความซับซ้อน
- 1.5.7 จัดทำรายงานเพื่อนำเสนอ และสรุปผล

## 1.6 เอกสารและงานวิจัยที่เกี่ยวข้อง

- 1.6.1 Rice, J. G. and Schnipke, R. J. [1] นำวิธีสตรีมไลน์อัปวินด์แบบโดยตรงที่พัฒนาขึ้นไปประยุกต์ใช้เพื่อการวิเคราะห์ปัญหาการไหลแบบหนึ่งชนิดอัดตัวไม่ได้พร้อมทั้งใช้ขั้นตอนการคำนวณความเร็วและความดันแบบแยกกัน
- 1.6.2 Chorin, A. [6,13] นำเสนอระเบียบวิธีการคำนวณแบบแยกกัน (segregated solution method) วิธีนี้จะเริ่มทำการคำนวณหาค่าความเร็วก่อนโดยใช้ค่าความดันที่สมมติขึ้น จากนั้นจะใช้สมการที่เกิดจากการรวมกันของสมการอนุรักษ์มวลและสมการอนุรักษ์โมเมนตัมในการหาค่าของความดัน ซึ่งค่าความดันที่คำนวณได้นี้จะนำกลับไปใช้ในการคำนวณหาค่าความเร็วใหม่ แต่วิธีการคำนวณแบบแยกกันที่เป็นที่รู้จักและเป็นที่ยอมรับกันอย่างแพร่หลายจนถึงปัจจุบันได้แก่วิธีที่เรียกว่า SIMPLE (semi-implicit method for pressure-linked equation) [6] โดยที่วิธีดังกล่าวจะทำการแทนสมการอนุรักษ์โมเมนตัมลงในสมการอนุรักษ์มวลเพื่อสร้างสมการของตัวแก้ไขความดัน (pressure correction equation) ซึ่งขั้นตอนในการคำนวณของวิธี SIMPLE มีดังนี้
1. สมมติการกระจายตัวของความดัน
  2. แก้สมการโมเมนตัมเพื่อหาค่าความเร็วโดยใช้ค่าความดันที่สมมติไว้
  3. คำนวณหาค่าของตัวแก้ไขความดันจากสมการของตัวแก้ไขความดัน

4. คำนวณหาค่าความดันใหม่โดยใช้ตัวแก้ไขความดัน
  5. แก้ไขความเร็วใหม่โดยใช้ค่าของตัวแก้ไขความดัน
  6. นำค่าความดันใหม่ที่ได้กลับไปคำนวณในข้อที่ 2 ใหม่ และทำการคำนวณซ้ำจนกว่าจะเข้าสู่ค่าตอบที่ต้องการ
- 1.6.3 Brooks, A. N. and Hughes, T. J. R. [12] พัฒนาวิธีสตรึมไลน์อัปวินด์ เพทรอพ-กาเลอร์คินสำหรับระเบียบวิธีไฟไนต์เอลิเมนต์ขึ้น วิธีดังกล่าวได้ทำการปรับปรุงฟังก์ชันถ่วงน้ำหนักเศษตค่างโดยใช้ทิศทางของเวกเตอร์ความเร็วในการทำให้ฟังก์ชันดังกล่าวมีการถ่วงน้ำหนักไปในแนวของการไหล โดยที่วิธีการดังกล่าวสามารถให้ความถูกต้องในอันดับ 2 ( $2^{\text{nd}}$  order of accuracy) แต่ยังคงให้ผลคำตอบที่มีการล้นอยู่บ้างในบางปัญหา
- 1.6.4 Choi, H. G., Choi, H. and Yoo J. Y. [7] พัฒนาวิธีการคำนวณแบบแยกส่วน 4 ขั้นตอนไปประยุกต์ใช้วิเคราะห์ปัญหาการไหลแบบหนึ่งชนิดอัดตัวไม่ได้ โดยรวมอุณหภูมิจึงใช้เอลิเมนต์สี่เหลี่ยมสี่จุดต่อ โดยเสนอวิธีการแก้สมการความดันแบบใหม่ ซึ่งผลที่ได้เป็นที่น่าพอใจ
- 1.6.5 Sung, J., Choi, H. G. and Yoo J. Y. [10] ได้ทำการเปรียบเทียบวิธีการคำนวณแบบแยกส่วน 4 ขั้นตอนกับวิธีการคำนวณแบบแยกส่วน 3 ขั้นตอนของ Chorin (Chorin type SIMPLE algorithm) โดยประยุกต์ใช้วิเคราะห์ปัญหาการไหลที่สภาวะไม่คงตัวในของไหลของคลื่นแบบอิสระ
- 1.6.6 Zienkiewicz and Codina [14] นำเสนอวิธีการแยกด้วยคุณลักษณะ (characteristic based split algorithm, CBS algorithm) หรือวิธีที่เรียกกันโดยทั่วไปว่าวิธีซีบีเอสในการจัดการกับพจน์เนื่องจากการพาโดยย้ายพิกัดที่อยู่บนแกนอ้างอิงไปอยู่บนแกน คุณลักษณะซึ่งเป็นแกนที่เคลื่อนที่ไปกับอนุภาคของของไหลบนเส้นทางเดินของของไหล ซึ่งเป็นผลให้พจน์เนื่องจากการพาหายไป จากนั้นจึงทำการประมาณค่าความเร็วและความดันกลับมาอยู่บนแกนอ้างอิงตามเดิม นอกจากนี้ยังได้ประยุกต์ใช้วิธีการคำนวณแบบแยกส่วน ในการหาผลลัพธ์ของความเร็วและความดัน จะเห็นว่าวิธีการดังกล่าวลดการล้นของผลลัพธ์เนื่องจากการพาและสามารถเลือกใช้อันดับของฟังก์ชันการประมาณภายในสำหรับความเร็วและความ

ตันที่เท่ากันได้ ช่วยให้การลดความยุ่งยากในการคำนวณจากการ อินทิเกรตเชิงตัวเลข (numerical integration) และได้ผลลัพธ์ที่ดีขึ้น

- 1.6.7 Massarotti, N., Arpino, F., Lewis, R. W. and Nithiarasu, P. [15] นำเสนอการเปรียบเทียบระหว่างขั้นตอนวิธีโดยชัดแจ้ง (explicit) กับขั้นตอนวิธีโดยปริยาย (implicit) ด้วยวิธีการแยกด้วยคุณลักษณะ (CBS) ผลที่ได้มีความแตกต่างกันระหว่าง 2 วิธี
- 1.6.8 Wansophark, N. and Dechaumphai, P. [3] พัฒนาวิธีสตรึมไลน์อัปวินด์แบบโดยตรงให้สามารถวิเคราะห์ปัญหาการไหลแบบหนึ่งมิติอัดตัวไม่ได้ โดยรวมอุณหภูมิจากและประยุกต์ใช้เทคนิคการปรับขนาดเอลิเมนต์ จากนั้นได้ทำการพัฒนาโปรแกรมที่ประดิษฐ์ขึ้น เพื่อแก้ปัญหาคำนวณการถ่ายเทความร้อนแบบคอนจูเกต [16] ผลที่ได้มีความสอดคล้องกับผลการคำนวณกับงานวิจัยอื่น ๆ เป็นอย่างดี
- 1.6.9 Malatip, A., Wansophark, N. and Dechaumphai, P. [4] นำเสนอวิธีสตรึมไลน์อัปวินด์ เพทรอพ-กาเลอร์คิน (SUPG) โดยรวมกับระเบียบวิธีการแยกกัน (segregated solution method) เพื่อแก้ปัญหาคำนวณการถ่ายเทความร้อนแบบคอนจูเกต แล้วได้ทำการเปรียบเทียบกับผลการคำนวณกับงานวิจัยอื่น ๆ โดยผลที่ได้มีความสอดคล้องกันเป็นอย่างดี
- 1.6.10 Codina, R. and Badia, S. [11] ได้เสนอวิธีแก้สมการนาเวียร์-สโตกส์ โดยรวบรวมแนวทางในการแก้สมการความดันไว้หลายวิธี โดยเสนอวิธีรวมเข้าด้วยกันของพจน์ที่เกี่ยวข้องกับเวลาอันดับหนึ่งและอันดับสอง (first and second order time integration schemes) โดยได้ใช้วิธีการคำนวณแบบแยกส่วน 3 ขั้นตอน ไปประยุกต์ใช้วิเคราะห์ปัญหาการไหลแบบหนึ่งมิติอัดตัวไม่ได้ โดยคำนวณหาค่าความเร็วและความดันแบบแยกกัน
- 1.6.11 Sai, B.V.K.S., Seetharamu, K.N. and Narayana, P.A.A. [17] ได้เสนอวิธีแก้สมการนาเวียร์-สโตกส์ โดยใช้การคำนวณแบบแยกส่วน Chorin (Chorin type SIMPLE algorithm) ซึ่งวิธีการดังกล่าวให้ความถูกต้องอันดับ 1 ( $1^{\text{nd}}$  order of accuracy) แล้วประยุกต์ใช้ระเบียบวิธีไฟไนต์เอลิเมนต์เพื่อวิเคราะห์ปัญหาการไหลเนื่องจากการพาความร้อนแบบอิสระที่สภาวะไม่คงตัว

## บทที่ 2

### สมการเชิงอนุพันธ์ย่อยสำหรับการไหล การถ่ายเทความร้อน และของแข็ง

งานวิจัยนี้จะศึกษาปัญหาปฏิสัมพันธ์ระหว่างของไหล อุณหภูมิ และโครงสร้างในสองมิติโดยใช้ระเบียบวิธีไฟไนต์เอลิเมนต์ การทำความเข้าใจในพฤติกรรมของการไหลของของไหลจะต้องอาศัยความรู้ทฤษฎีพลศาสตร์ของไหล ส่วนการหาลักษณะความเค้นเนื่องจากความร้อนที่เกิดขึ้นในโครงสร้างจะต้องอาศัยทฤษฎีการถ่ายเทความร้อน และทฤษฎีกลศาสตร์ของแข็งเพิ่มเติมด้วย ดังนั้นในบทนี้จะได้กล่าวถึงทฤษฎีพลศาสตร์ของไหล [18] เพื่อหาสมการเชิงอนุพันธ์ย่อยสำหรับปัญหาการไหล ทฤษฎีการถ่ายเทความร้อน [19] เพื่อหาสมการเชิงอนุพันธ์ย่อยสำหรับปัญหาการถ่ายเทความร้อน และทฤษฎีกลศาสตร์ของแข็ง [20] เพื่อหาสมการเชิงอนุพันธ์ย่อยสำหรับปัญหาของแข็ง ตามลำดับ สุดท้ายจะได้นำเสนอขั้นตอนโดยรวมของระเบียบวิธีไฟไนต์เอลิเมนต์ และคุณลักษณะของเอลิเมนต์สามเหลี่ยมแบบสามจุดต่อ

#### 2.1 สมการเชิงอนุพันธ์ย่อยสำหรับการไหล

สำหรับปัญหาการไหลแบบหนืดชนิดอัดตัวไม่ได้ในสภาวะไม่คงตัว (two dimensional unsteady viscous incompressible flows) ประกอบไปด้วยสมการเชิงอนุพันธ์ย่อยที่สอดคล้องกับ: (1) การอนุรักษ์มวล (conservation of mass), (2) การอนุรักษ์โมเมนตัม (conservation of momentum) และ (3) การอนุรักษ์พลังงาน (conservation of energy) ซึ่งประกอบด้วยสมการทั้งหมด 4 สมการในปัญหาแบบสองมิติ ดังนี้

สมการอนุรักษ์มวล

$$\frac{\partial u_i}{\partial x_i} = 0 \quad (2.1)$$

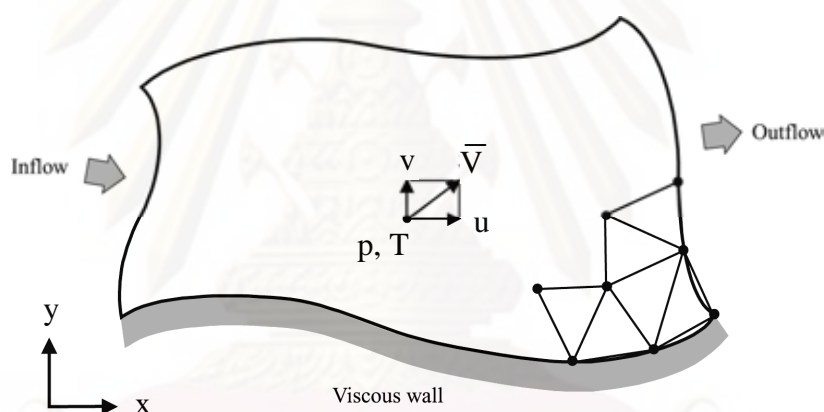
สมการอนุรักษ์โมเมนตัม

$$\frac{\partial u_i}{\partial t} + \frac{\partial}{\partial x_j} (u_j u_i) = -\frac{1}{\rho} \frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j} \left( \nu \frac{\partial u_i}{\partial x_j} \right) - g_i (1 - \beta (T - T_0)) \quad (2.2)$$

สมการอนุรักษ์พลังงาน

$$\frac{\partial T}{\partial t} + \frac{\partial}{\partial x_j} (u_j T) = \frac{\partial}{\partial x_j} \left( \tilde{k} \frac{\partial T}{\partial x_j} \right) + \tilde{Q} \quad (2.3)$$

โดยที่  $i, j = 1, 2$   $x_i$  หมายถึงระบบพิกัดฉากในแนวแกน  $x$  และแกน  $y$  ตามลำดับ  $u_i$  หมายถึงความเร็วในแนวแกน  $x$  และแกน  $y$  ตามลำดับ  $t$  หมายถึงเวลา  $\rho$  หมายถึงความหนาแน่น  $p$  หมายถึงค่าความดัน  $\nu$  หมายถึงค่าสัมประสิทธิ์ความหนืดจลนศาสตร์ (kinematics viscosity;  $\nu = \mu / \rho$ )  $\mu$  หมายถึงค่าความหนืด  $g_i$  หมายถึงค่าคงที่ของความเร่งเนื่องจากแรงโน้มถ่วง (gravitational acceleration constant)  $\beta$  หมายถึงค่าสัมประสิทธิ์การขยายตัวเนื่องจากอุณหภูมิ (coefficient of thermal expansion) ของของไหล  $T$  หมายถึงค่าอุณหภูมิ  $T_0$  หมายถึงอุณหภูมิอ้างอิงที่ของไหลไม่เกิดการลอยตัว  $\alpha$  หมายถึงค่าการการแพร่กระจายความร้อน (thermal diffusivity coefficient;  $\alpha = k / \rho c$ )  $k$  หมายถึงค่าการนำความร้อน  $c$  หมายถึงค่าความจุความร้อน  $Q$  หมายถึงอัตราปริมาณความร้อนที่ผลิตได้เองต่อหนึ่งหน่วยมวล



รูปที่ 2.1 ขอบเขตและเงื่อนไขขอบเขตของการไหลแบบหนืด

ระบบสมการเชิงอนุพันธ์ย่อยสำหรับปัญหาการไหล (2.1)–(2.3) นี้ จำเป็นต้องแก้ควบคู่ไปกับเงื่อนไขขอบเขตของปัญหาที่กำหนดให้ ดังแสดงในรูปที่ 2.1 ซึ่งอาจประกอบไปด้วย: (1) การกำหนดขอบเขตของการไหลเข้า (inflow); (2) การกำหนดขอบเขตของผนังแบบหนืด (viscous wall); (3) การกำหนดขอบเขตของการไหลออก (outflow); และ (4) การกำหนดอุณหภูมิที่ขอบเป็นต้น

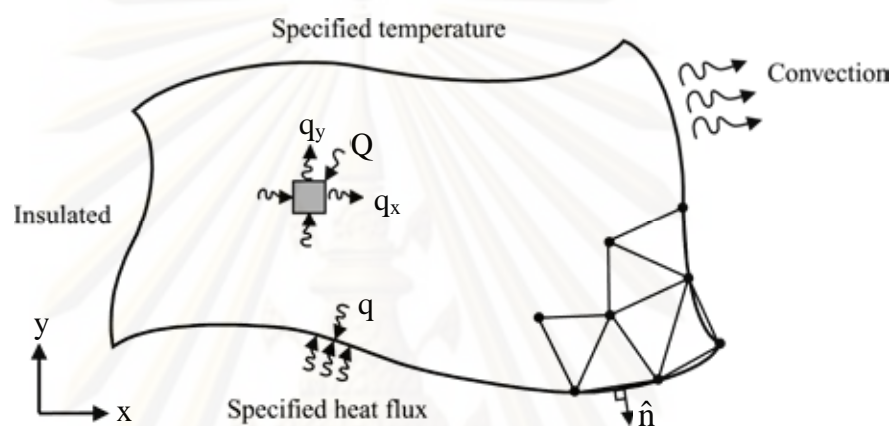
## 2.2 สมการเชิงอนุพันธ์ย่อยสำหรับการถ่ายเทความร้อน

สมการเชิงอนุพันธ์ที่อธิบายความสัมพันธ์ของปริมาณฟลักซ์  $q_i$  ซึ่งผ่านเข้าและออกในแนวแกน  $x$  และ  $y$  ของเอลิเมนต์เล็ก ๆ ที่วางตัวอยู่ ณ ตำแหน่งใด ๆ บนแผ่นระนาบดังแสดงใน

รูปที่ 2.2 คือ

$$\rho c \frac{\partial T}{\partial t} + \frac{\partial q_i}{\partial x_i} - Q = 0 \quad (2.4)$$

โดย  $Q$  แทนอัตราปริมาณความร้อนที่ผลิตได้เอง (internal heat generation) ต่อหนึ่งหน่วยปริมาตร



รูปที่ 2.2 ขอบเขตและเงื่อนไขขอบเขตของปัญหาการถ่ายเทความร้อน

ปริมาณฟลักซ์เหล่านี้ขึ้นอยู่กับความชัน (temperature gradients) ของอุณหภูมิ  $T$  และค่าสัมประสิทธิ์การนำความร้อน (thermal conductivity)  $k$  ของแผ่นระนาบตามกฎฟูริเยร์ (Fourier's law) ดังนี้

$$q_i = -k \frac{\partial T}{\partial x_i} \quad (2.5)$$

แทนสมการ (2.5) ลงในสมการ (2.4) ก่อให้เกิดสมการเชิงอนุพันธ์สำหรับการถ่ายเทความร้อนภายใต้สภาวะอยู่ตัว คือ

$$\rho c \frac{\partial T}{\partial t} + \frac{\partial}{\partial x_i} \left[ k \frac{\partial T}{\partial x_i} \right] + Q = 0 \quad (2.6)$$

ความซับซ้อนของลักษณะการกระจายของอุณหภูมิ  $T(x, y, t)$  ที่เกิดขึ้นจากสมการ (2.6) นี้ขึ้นอยู่กับรูปร่างลักษณะของแผ่นระนาบและเงื่อนไขขอบเขตที่กำหนดให้ เงื่อนไขขอบเขตตลอดขอบของแผ่นบางอาจประกอบด้วย

(ก) การกำหนดอุณหภูมิตลอดขอบ เช่น

$$T(x, y, t) = T_1(x, y, t)$$

(ข) การกำหนดปริมาณฟลักซ์  $q_s$  ที่ผ่านเข้าสู่ขอบ

$$q_i = -q = k \frac{\partial T}{\partial x_i} n_i$$

โดย  $n_i$  แทนทิศทางโคไซน์ (cosines direction) ของเวกเตอร์หนึ่งหน่วย  $\hat{n}$  ที่ตั้งฉากกับขอบนั้น

(ค) การกำหนดว่าขอบเป็นฉนวน ไม่มีปริมาณฟลักซ์ผ่านเข้าหรือออกได้

$$q_i = k \frac{\partial T}{\partial x_i} n_i = 0$$

### 2.3 สมการเชิงอนุพันธ์ย่อยสำหรับปัญหาของแข็ง

สมการเชิงอนุพันธ์ย่อยซึ่งแสดงความสมดุลของแรงในแนวแกน  $x$  และ  $y$  บนแผ่นระนาบ เมื่อไม่คิดน้ำหนักของตัวเอง ดังแสดงในรูปที่ 2.3 คือ

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} = 0 \quad (2.7ก)$$

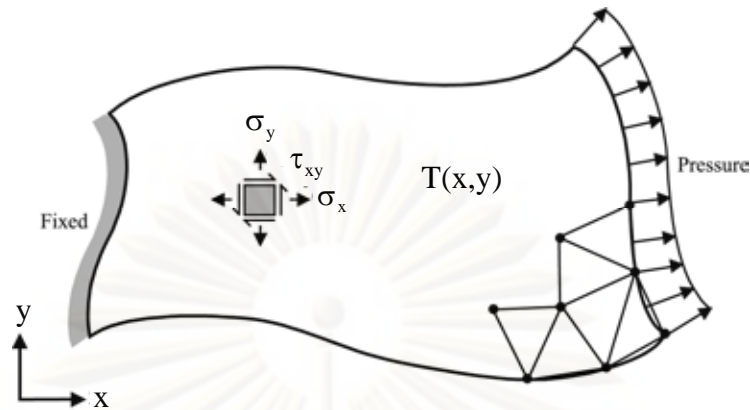
$$\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} = 0 \quad (2.7ข)$$

โดย  $\sigma_x$ ,  $\sigma_y$  แทนค่าความเค้นฉาก (normal stress) ในแนวแกน  $x$  และ  $y$  ตามลำดับ ส่วน  $\tau_{xy}$  แทนค่าความเค้นเฉือน (shearing stress)

ในกรณีของแผ่นบางซึ่งเป็นปัญหาความเค้นในระนาบ (plane stress) ค่าความเค้นในแนวแกน  $z$  จะถูกสมมติให้มีค่าเป็นศูนย์ โดยค่าความเค้นย่อยต่าง ๆ สามารถเขียนให้อยู่ในรูปแบบของค่าความเครียดย่อย (strain components) ได้ดังนี้

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \frac{E}{(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \begin{Bmatrix} \epsilon_x - \alpha(T-T_0) \\ \epsilon_y - \alpha(T-T_0) \\ \gamma_{xy} \end{Bmatrix} \quad (2.8)$$





รูปที่ 2.3 ขอบเขตและเงื่อนไขขอบเขตของปัญหาของแข็ง

โดย  $E$  แทนค่าโมดูลัสของยังส์ (Young's modulus) และ  $\nu$  แทนค่าอัตราส่วนของปัวส์ซง (Poisson's ratio) ส่วน  $\epsilon_x, \epsilon_y$  แทนค่าความเครียดฉาก (normal strain) ในแนวแกน  $x$  และแกน  $y$  ตามลำดับ  $\gamma_{xy}$  แทนค่าความเครียดเฉือน  $\alpha$  แทนสัมประสิทธิ์การขยายตัวเนื่องจากอุณหภูมิ (coefficient of thermal expansion) และ  $T_0$  แทนอุณหภูมิอ้างอิงที่วัสดุไม่เกิดความเค้น (reference temperature for zero stress) เช่นอุณหภูมิห้อง เป็นต้น

ค่าความเครียดย่อยเหล่านี้เขียนให้อยู่ในรูปของค่าการเสียรูป  $u$  และ  $v$  ในแนวแกน  $x$  และ  $y$  ได้คือ

$$\epsilon_x = \frac{\partial u}{\partial x}; \quad \epsilon_y = \frac{\partial v}{\partial y} \quad (2.9)$$

และ 
$$\gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}$$

ซึ่งหมายความว่า ตัวไม่รู้ค่าในปัญหาแผ่นระนาบมีเพียง 2 ค่า คือ  $u$  และ  $v$  เมื่อทราบค่า  $u$  และ  $v$  แล้วจึงสามารถนำไปคำนวณหาค่าความเครียดและความเค้นย่อยต่างๆ ได้ตามลำดับ

สำหรับปัญหาที่ค่าความเครียดในแนวแกน  $Z$  ถูกสมมติให้มีค่าเป็นศูนย์ (plane strain) สมการเชิงอนุพันธ์ย่อย (2.7) และสมการ (2.9) ซึ่งแสดงความสัมพันธ์ระหว่างความเครียดย่อยและค่าการเสียรูป ยังคงใช้ได้เช่นเดิม ยกเว้นความสัมพันธ์ระหว่างค่าความเค้นย่อยและความเครียดย่อยดังแสดงในสมการ (2.8) ต้องเปลี่ยนมาเป็น

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \begin{Bmatrix} \epsilon_x - \alpha(T-T_0) \\ \epsilon_y - \alpha(T-T_0) \\ \gamma_{xy} \end{Bmatrix} \quad (2.10)$$

สำหรับเงื่อนไขขอบเขตโดยทั่วไปดังแสดงในรูปที่ 2.3 ประกอบด้วย การยึดแน่น (fixed) หรือปล่อยให้อิสระ (free) ตลอดขอบต่าง ๆ รวมทั้งอาจกำหนดแรงดัน (pressure) ซึ่งแทนแรงที่กระทำต่อพื้นที่ตลอดขอบนั้น ๆ อีกทั้งยังต้องการผลการกระจายตัวของอุณหภูมิ  $T = T(x, y)$  ที่สามารถคำนวณได้จากสมการ (2.6) ก่อนด้วย ซึ่งผลจากอุณหภูมิ  $T$  นี้เองแฝงตัวอยู่ในสมการ (2.8) และ (2.10) ซึ่งอธิบายลักษณะของความเค้นย่อนั้นเอง

## 2.4 ระเบียบวิธีไฟไนต์เอลิเมนต์

การนำระเบียบวิธีไฟไนต์เอลิเมนต์ (finite element method) มาประยุกต์ใช้ในการวิเคราะห์ปัญหาการถ่ายเทความร้อนและการไหลแบบหนืดแต่ไม่อัดตัวภายใต้สภาวะอยู่ตัว โดยจะเริ่มจากขั้นตอนทั่วไปของระเบียบวิธีไฟไนต์เอลิเมนต์ จากนั้นจะอธิบายรายละเอียดของระเบียบวิธีไฟไนต์เอลิเมนต์และสุดท้ายจะเป็นการประดิษฐ์สมการไฟไนต์เอลิเมนต์เพื่อการวิเคราะห์การถ่ายเทความร้อนและการไหลแบบหนืด

### 2.4.1 ขั้นตอนทั่วไปของระเบียบวิธีไฟไนต์เอลิเมนต์

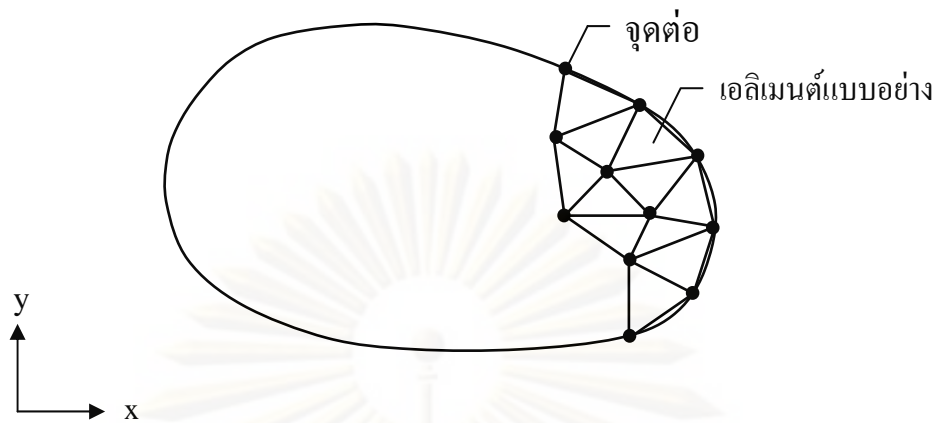
การแก้ปัญหาไฟไนต์เอลิเมนต์โดยวิธีการถ่วงน้ำหนักเศษตักข้างประกอบด้วยขั้นตอนที่สำคัญ 6 ขั้นตอน [20] คือ

ขั้นตอนที่ 1 แบ่งขอบเขตรูปร่างของปัญหาออกเป็นเอลิเมนต์ย่อย ๆ เช่น แบ่งออกเป็นเอลิเมนต์สามเหลี่ยมย่อย ๆ สำหรับปัญหาในสองมิติ ดังรูปที่ 2.4

จากนั้นก็ทำการหาสมการเชิงอนุพันธ์ที่สอดคล้องกับปัญหาที่ต้องการแก้ นั้น โดยสมการเชิงอนุพันธ์ทั่วไปสามารถเขียนให้อยู่ใน

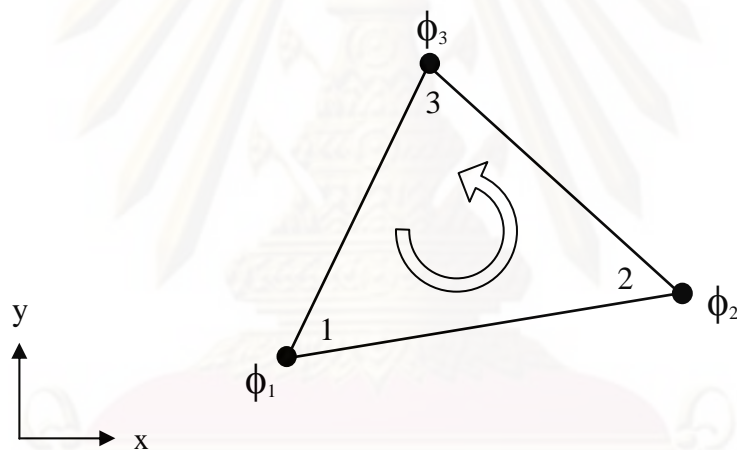
$$D(\phi') = 0 \quad (2.11)$$

โดยที่  $D$  คือตัวดำเนินการเชิงอนุพันธ์ (differential operator) และ  $\phi'$  คือตัวแปรตามแมนตรง



รูปที่ 2.4 การแบ่งขอบเขตรูปร่างของปัญหาออกเป็นเอลิเมนต์ย่อย ๆ

ขั้นตอนที่ 2 พิจารณาลักษณะการกระจายของผลเฉลยโดยประมาณบนเอลิเมนต์



รูปที่ 2.5 เอลิเมนต์สามเหลี่ยมแบบ 3 จุดต่อและตัวไม่ทราบค่าที่จุดต่อ

ยกตัวอย่างเช่น สำหรับเอลิเมนต์ที่ประกอบด้วยสามจุดต่อดังแสดงในรูปที่ 2.5 โดยที่จุดต่อนี้เป็นตำแหน่งของตัวไม่รู้จักค่า  $\phi_1$ ,  $\phi_2$  และ  $\phi_3$  ตัวไม่รู้จักค่าเหล่านี้เป็นคุณสมบัติต่าง ๆ ของการไหล ซึ่งสามารถสร้างสมการอธิบายลักษณะการกระจายของตัวไม่ทราบค่าที่จุดต่อได้ดังนี้

$$\phi(x, y) = N_1(x, y)\phi_1 + N_2(x, y)\phi_2 + N_3(x, y)\phi_3 \quad (2.12)$$

โดย  $N_i(x, y)$ ;  $i = 1, 2, 3$  แทนฟังก์ชันการประมาณภายในเอลิเมนต์ สมการ (2.12) นี้สามารถเขียนให้อยู่ในรูปของเมตริกซ์ได้ดังนี้

$$\begin{aligned}\phi = \phi(x, y) &= [N_1 \quad N_2 \quad N_3] \begin{Bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{Bmatrix} \\ &= [N(x, y)]_{(1 \times 3)} \{\phi\}_{(3 \times 1)}\end{aligned}\quad (2.13)$$

โดยที่  $[N]$  คือ เมทริกซ์ของฟังก์ชันการประมาณภายในเอลิเมนต์

$\{\phi\}$  คือ เวกเตอร์เมทริกซ์ที่ประกอบไปด้วยตัวไม่ทราบค่าที่จุดต่อของเอลิเมนต์นั้น

ฟังก์ชันการประมาณภายในเอลิเมนต์สามเหลี่ยมเป็นแบบเชิงเส้น คือ

$$N_i(x, y) = \frac{1}{2A}(a_i + b_i x + c_i y) \quad i=1, 2, 3 \quad (2.14)$$

โดย  $A$  คือ พื้นที่ของเอลิเมนต์สามเหลี่ยม สามารถคำนวณได้จากโคออร์ดิเนตที่จุดต่อทั้งสามจุด ดังนี้

$$A = \frac{1}{2}[x_2(y_3 - y_1) + x_1(y_2 - y_3) + x_3(y_1 - y_2)] \quad (2.15)$$

$$\begin{aligned}a_1 &= x_2 y_3 - x_3 y_2 & b_1 &= y_2 - y_3 & c_1 &= x_3 - x_2 \\ a_2 &= x_3 y_1 - x_1 y_3 & b_2 &= y_3 - y_1 & c_2 &= x_1 - x_3 \\ a_3 &= x_1 y_2 - x_2 y_1 & b_3 &= y_1 - y_2 & c_3 &= x_2 - x_1\end{aligned}\quad (2.16)$$

ขั้นตอนที่ 3 ประยุกต์ระเบียบวิธีถ่วงน้ำหนักเศษตกค้าง (method of weighted residual) เข้ากับสมการเชิงอนุพันธ์ย่อยเพื่อให้ผลลัพธ์โดยประมาณนั้นมีความคลาดเคลื่อนน้อยที่สุด ซึ่งจะก่อให้เกิดสมการไฟไนต์เอลิเมนต์ (finite element equation) ที่สอดคล้องกัน ซึ่งสามารถเขียนในรูปเมทริกซ์ได้ดังนี้

$$D(\phi) \text{ จะไม่เท่ากับ } 0 \text{ แต่จะเท่ากับ } R$$

โดยที่  $R$  คือเศษตกค้าง (residual) นั้นหมายถึง

$$R = D(\phi) = D\left(\sum_{i=1}^m N_i \phi_i\right) \quad (2.17)$$

โดย  $m$  คือจำนวนจุดต่อของเอลิเมนต์นั้น

จากวิธีกาลิเคอริคิน (Galerkin) ซึ่งมีขั้นตอนโดยเริ่มจากการคูณเศษตกค้าง  $R$  ด้วยฟังก์ชันน้ำหนัก (weighting function;  $W$ ) จากนั้นจึงอินทิเกรตตลอดทั้งโดเมนของเอลิเมนต์แล้วกำหนดผลที่ได้ให้เท่ากับศูนย์นั้นคือ

$$\int_{\Omega} W_i R d\Omega = 0 \quad i = 1, 2, \dots, m \quad (2.18)$$

ขั้นตอนที่ 4 อินทิเกรตทีละส่วน (integrate by part) ซึ่งหากเราแทนสมการ (2.17) ลงในสมการ (2.18) แล้วอินทิเกรตทีละส่วนจะได้

$$\begin{aligned} \int_{\Omega} W_i R d\Omega &= \int_{\Omega^{(e)}} W_i D \left( \sum_{i=1}^m N_i \phi_i \right) d\Omega \\ &= \underbrace{\int_{\Omega^{(e)}} (W_i, N_i, \phi_i) d\Omega}_{\text{พจน์ที่เกี่ยวข้องกับโดเมนของเอลิเมนต์ } \Omega^{(e)}} + \underbrace{\int_{\Gamma^{(e)}} (W_i, N_i, \phi_i) d\Gamma}_{\text{พจน์ที่เกี่ยวข้องกับขอบเขตของเอลิเมนต์ } \Gamma^{(e)}} = 0 \end{aligned}$$

ขั้นตอนที่ 5 แทนพจน์ที่เกี่ยวข้องกับขอบเขตของเอลิเมนต์  $\Gamma^{(e)}$  ด้วยภาวะขอบเขตอื่น ๆ ที่เกี่ยวข้อง ซึ่งจะก่อให้เกิดสมการของเอลิเมนต์ที่สมบูรณ์สำหรับปัญหาที่พิจารณา

ขั้นตอนที่ 6 จากนั้นเขียนสมการของเอลิเมนต์ ซึ่งมีทั้งหมด  $m$  สมการให้อยู่ในรูปของเมตริกซ์ นั่นคือ

$$\begin{bmatrix} \mathbf{K} \end{bmatrix}_{(m \times m)} \begin{Bmatrix} \phi \end{Bmatrix}_{(m \times 1)} = \begin{Bmatrix} \mathbf{F} \end{Bmatrix}_{(m \times 1)} \quad (2.19)$$

โดย  $[\mathbf{K}]$  คือ เอลิเมนต์เมตริกซ์ของความแข็งแรง (element stiffness matrix)  $\{\phi\}$  คือ เวกเตอร์ซึ่งประกอบด้วยตัวไม่รู้ค่าที่จุดต่อต่าง ๆ ของเอลิเมนต์และ  $\{\mathbf{F}\}$  คือ โหลดเวกเตอร์ของเอลิเมนต์นั้น เมื่อได้สมการไฟไนต์เอลิเมนต์ดังเช่นแสดงในสมการ (2.19) แล้วลำดับขั้นตอนต่อไปก็จะทำการรวมสมการของเอลิเมนต์ย่อยเข้าด้วยกันก่อให้เกิดระบบสมการรวมจากนั้นกำหนดค่าที่ขอบเขต แล้วจึงแก้ระบบสมการรวมเพื่อหาค่าผลลัพธ์ที่จุดต่อต่าง ๆ

### บทที่ 3

## ระเบียบวิธีไฟไนต์เอลิเมนต์สำหรับการวิเคราะห์ร่วมกันระหว่าง การไหล อุณหภูมิ และโครงสร้าง

การนำระเบียบวิธีไฟไนต์เอลิเมนต์ (finite element method) มาประยุกต์ใช้ในการวิเคราะห์ปัญหาการไหลแบบหนืดแต่ไม่อันตัวภายใต้สภาวะไม่อยู่ตัว ปัญหาการถ่ายเทความร้อนและกลศาสตร์ของแข็งของโครงสร้าง จะต้องใช้ระเบียบวิธีไฟไนต์เอลิเมนต์ที่เหมาะสมนำมาประยุกต์ใช้กับปัญหาในแต่ละด้าน เพื่อที่จะทำนายพฤติกรรมของปัญหาได้อย่างถูกต้อง ขั้นตอนการหาสมการไฟไนต์เอลิเมนต์ของปัญหาแต่ละด้านดังกล่าวมีรายละเอียดดังนี้

### 3.1 ระเบียบวิธีไฟไนต์เอลิเมนต์สำหรับปัญหาการไหลแบบหนืด

การประดิษฐ์สมการไฟไนต์เอลิเมนต์ โดยภาพรวมแล้วประกอบด้วย 4 ขั้นตอน คือ ทำการแบ่งย่อยช่วงเวลา (time discretization) ระเบียบวิธีการคำนวณแบบแยกส่วน (fractional step method) ระเบียบวิธีสตรีมไลน์อัปวินด์เพโทรฟ-กาเลอร์คิน (streamline upwind/ Petrov-Galerkin formulation; SUPG) และระเบียบวิธีถ่วงน้ำหนักเศษตกค้าง (method of weighted residual) [21]

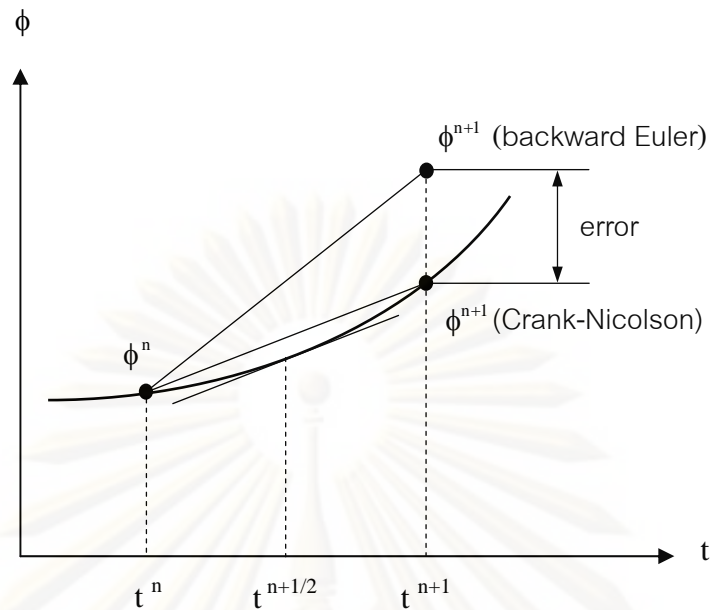
#### 3.1.1 ระเบียบวิธีการแบ่งย่อยช่วงเวลา (time discretization)

เริ่มต้นจากการพิจารณาสมการพาและการแพร่ (convection-diffusion equation) ของตัวแปรสเกลาร์ (scalar variable) ดังแสดงในสมการ (3.1)

$$\frac{\partial \phi}{\partial t} + \frac{\partial}{\partial x_j} (u_j \phi) - \frac{\partial}{\partial x_j} \left( k \frac{\partial \phi}{\partial x_j} \right) - Q = 0 \quad (3.1)$$

ทำการประยุกต์วิธีของความสัมพันธ์เวียนบังเกิด (recurrence relations) [20] เพื่อสร้างความสัมพันธ์ที่เกี่ยวข้องกับเวลากับสมการ (3.1) โดยพิจารณาจากรูป 3.1 ได้ดังต่อไปนี้

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = \theta \left[ \frac{\partial}{\partial x_j} \left( k \frac{\partial \phi}{\partial x_j} \right) - \frac{\partial}{\partial x_j} (u_j \phi) + Q \right]^{n+1} + (1-\theta) \left[ \frac{\partial}{\partial x_j} \left( k \frac{\partial \phi}{\partial x_j} \right) - \frac{\partial}{\partial x_j} (u_j \phi) + Q \right]^n \quad (3.2)$$



รูปที่ 3.1 ความสัมพันธ์ของตัวแปรสเกลาร์ในช่วงเวลา  $t^n \rightarrow t^{n+1}$

จากนั้นใช้ค่า  $\theta$  เท่ากับ 0.5 ซึ่งเรียกว่าวิธีแครงก์-นิโคลสัน (Crank-Nicolson scheme) เพื่อให้เกิดเป็นการประมาณอันดับสอง (second order approximation) [9] ในสมการ (3.2) ได้ดังนี้

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = \frac{1}{2} \left[ \frac{\partial}{\partial x_j} \left( k \frac{\partial \phi}{\partial x_j} \right) - \frac{\partial}{\partial x_j} (u_j \phi) + Q \right]^{n+1} + \frac{1}{2} \left[ \frac{\partial}{\partial x_j} \left( k \frac{\partial \phi}{\partial x_j} \right) - \frac{\partial}{\partial x_j} (u_j \phi) + Q \right]^n \quad (3.3)$$

สามารถจัดรูปใหม่ได้เป็น

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} + \frac{1}{2} \left[ \frac{\partial}{\partial x_j} (u_j^{n+1} \phi^{n+1}) + \frac{\partial}{\partial x_j} (u_j^n \phi^n) \right] = \frac{1}{2} \left[ \frac{\partial}{\partial x_j} \left( k \frac{\partial \phi^{n+1}}{\partial x_j} \right) + \frac{\partial}{\partial x_j} \left( k \frac{\partial \phi^n}{\partial x_j} \right) \right] + \frac{1}{2} [Q^{n+1} + Q^n] \quad (3.4)$$

โดยที่ ตัวห้อย  $j=1, 2$  และ  $Q$  คือพจน์ของแหล่งกำเนิด (source term) ในที่นี้อาจจะเป็นพจน์ของความดัน หรือพจน์ของแรงลอยตัวเนื่องจากอุณหภูมิ ขณะเดียวกันสมการ (3.4) เป็นสมการแบบอย่างของปัญหาการพาและการแพร่ของปริมาณสเกลาร์ใด ๆ ซึ่งได้ทำการสร้างความสัมพันธ์ที่เกี่ยวกับช่วงเวลา โดยสามารถนำไปประยุกต์ใช้กับสมการเชิงอนุพันธ์อื่น ๆ ได้

### 3.1.2 วิธีการคำนวณแบบแยกส่วน (fractional step method)

สมการเชิงอนุพันธ์สำหรับปัญหาการไหลแบบหนืดแบบไม่อัดตัวภายใต้สภาวะไม่อยู่ตัว ที่แสดงในสมการที่ (2.1-2.2) นั้นมีลักษณะคล้ายกับสมการแบบอย่างของปัญหาการพาและการแพร่จึงสามารถนำสมการ (3.4) มาประยุกต์ได้โดยตรงกับสมการเชิงอนุพันธ์ย่อยของการอนุรักษ์โมเมนตัมเพื่อสร้างความสัมพันธ์ที่เกี่ยวข้องกับเวลา และสามารถจัดรูปใหม่ได้ดังนี้

สมการเชิงอนุพันธ์ย่อยของการอนุรักษ์มวล

$$\frac{\partial u_i^{n+1}}{\partial x_i} = 0 \quad (3.5)$$

สมการเชิงอนุพันธ์ย่อยของการอนุรักษ์โมเมนตัม

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{1}{2} \left[ H(u_i^{n+1}) + H(u_i^n) \right] - \frac{1}{2\rho} \left( \frac{\partial p^{n+1}}{\partial x_i} + \frac{\partial p^n}{\partial x_i} \right) \quad (3.6)$$

โดยที่  $H(u_i)$  คือพจน์ที่ประกอบไปด้วยพจน์การพา พจน์การแพร่และพจน์แรงลอยตัวเนื่องจากอุณหภูมิ

วิธีการคำนวณแบบแยกส่วน (fractional step method) นั้นได้ถูกนำเสนอครั้งแรกโดย Chorin [6] โดยขั้นตอนพื้นฐานของวิธีนี้ จะทำการแยกพจน์อัตราการเปลี่ยนแปลงความดันออกจากพจน์ของการพา การแพร่และพจน์อื่น ๆ เนื่องจากแรงภายนอก โดยทำการสมมติค่าความดันและค่าความเร็วในขอบเขตของปัญหาที่สนใจ แล้วคำนวณหาค่าความเร็วจากค่าที่สมมติเพื่อที่จะนำค่าความเร็วที่คำนวณได้ไปหาค่าความดันที่เหมาะสม โดยที่สมการความดันนั้นสามารถหาได้จากการรวมกันของสมการเชิงอนุพันธ์ย่อยของการอนุรักษ์โมเมนตัมและสมการเชิงอนุพันธ์ย่อยของการอนุรักษ์มวล แล้วใช้ค่าความดันที่ได้นั้นสามารถไปคำนวณหาค่าความเร็วที่เหมาะสมได้ ซึ่งวิธีการนี้จะทำให้ค่าความเร็วและความดันมีความสัมพันธ์เป็นไปตามการอนุรักษ์โมเมนตัมและการอนุรักษ์มวล โดยในงานวิจัยนี้จะใช้วิธีเครงก์-นิโคลสันโดยรวมกับวิธีการคำนวณแบบแยกส่วน 4 ขั้นตอนซึ่งได้ถูกพัฒนาโดย [7, 9, 23] โดยมีขั้นตอนดังนี้

ขั้นตอนที่ 1 คำนวณหาค่าความเร็วกลาง (intermediate velocity,  $\hat{u}$ ) โดยใช้ค่าความดันที่สมมติขึ้น ส่วนพจน์การพา พจน์การแพร่และพจน์เนื่องจากแรงลอยตัวเนื่องจากอุณหภูมินั้นจะใช้วิธี



แครงก์-นิโคลสัน (Crank-Nicolson method) ซึ่งเป็นวิธีการประมาณค่าอันดับสอง โดยจะรวมกันระหว่างค่าความเร็วที่คำนวณได้ ( $\hat{u}$ ) และค่าความเร็วที่สมมติขึ้นในตอนแรก ( $u^n$ ) จะได้

$$\frac{\hat{u}_i - u_i^n}{\Delta t} = \frac{1}{2} \left[ H(\hat{u}_i) + H(u_i^n) \right] - \left( \frac{1}{\rho} \frac{\partial p^n}{\partial x_i} \right) \quad (3.7)$$

ขั้นตอนที่ 2 นำครึ่งหนึ่งของพจน์อัตราการเปลี่ยนแปลงความดันของค่าที่สมมติ ลดทอนออกจากค่าความเร็ว  $\hat{u}$  ที่ได้จากขั้นตอนที่แรก แล้วนำไปใช้คำนวณหาค่าความเร็ว  $u^*$  จะได้

$$\frac{u_i^* - \hat{u}_i}{\Delta t} = \frac{1}{2} \left( \frac{1}{\rho} \frac{\partial p^n}{\partial x_i} \right) \quad (3.8)$$

ขั้นตอนที่ 3 คำนวณหาค่าความเร็วที่เหมาะสม ( $u^{n+1}$ ) โดยใช้อัตราการเปลี่ยนแปลงความดันที่เหมาะสม ( $p^{n+1}$ ) และค่าความเร็วที่ได้จากขั้นตอนที่สอง

$$\frac{u_i^{n+1} - u_i^*}{\Delta t} = -\frac{1}{2} \left( \frac{1}{\rho} \frac{\partial p^{n+1}}{\partial x_i} \right) \quad (3.9)$$

โดยที่ความดัน  $p^{n+1}$  เป็นตัวแปรไม่ทราบค่า เพราะฉะนั้นทำการจัดสมการ (3.9) ใหม่ จะได้

$$u_i^{n+1} = u_i^* - \frac{\Delta t}{2} \left( \frac{1}{\rho} \frac{\partial p^{n+1}}{\partial x_i} \right) \quad (3.10)$$

ขั้นตอนที่ 4 นำสมการ (3.10) แทนในสมการเชิงอนุพันธ์ย่อยของการอนุรักษ์มวล (3.5) แล้วทำการจัดรูปใหม่เพื่อคำนวณหา  $p^{n+1}$  จะได้ (โดยจะแสดงในหัวข้อ 3.1.4)

$$\frac{\partial}{\partial x_i} \left( \frac{\partial p^{n+1}}{\partial x_i} \right) = \frac{2\rho}{\Delta t} \left( \frac{\partial u_i^*}{\partial x_i} \right) \quad (3.11)$$

จากผลรวมของสมการ (3.7-3.9) จะได้

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{1}{2} \left[ H(\hat{u}_i) + H(u_i^n) \right] - \frac{1}{2\rho} \left( \frac{\partial p^{n+1}}{\partial x_i} + \frac{\partial p^n}{\partial x_i} \right) \quad (3.12)$$

เนื่องจากสมการ (3.12) มีลักษณะคล้ายกับสมการเชิงอนุพันธ์ย่อยอนุรักษ์โมเมนตัมที่ได้แสดงไว้ในสมการที่ (3.6) โดยสามารถแทนค่า  $\hat{u}$  ด้วย  $u^{n+1}$  ได้โดยตรง ซึ่งสามารถแสดงถึงค่าความถูกต้องอันดับสองของพจน์ที่เกี่ยวข้องกับเวลา ( $2^{nd}$  order accuracy in time) โดยสามารถอธิบายได้จากรวมสมการที่ (3.8-3.9) แล้วทำการจัดรูปจะได้

$$u_i^{n+1} - \hat{u}_i = -\frac{\Delta t}{2\rho} \frac{\partial(p^{n+1} - p^n)}{\partial x_i} \approx -\frac{(\Delta t)^2}{2\rho} \frac{\partial}{\partial x_i} \left( \frac{\partial p}{\partial t} \right) \quad (3.13)$$

สมการที่ (3.13) สามารถละทิ้งพจน์อันดับสูง  $(\Delta t)^2$  แล้วแทนค่าในสมการ (3.12) จะได้

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{1}{2} \left[ H(u_i^{n+1}) + H(u_i^n) \right] - \frac{1}{2\rho} \left( \frac{\partial p^{n+1}}{\partial x_i} + \frac{\partial p^n}{\partial x_i} \right) \quad (3.14)$$

โดยที่ ตัวห้อย  $i, j=1, 2$

ระบบสมการเชิงอนุพันธ์ย่อยนาเวียร์-สโตกส์ อันประกอบไปด้วยสมการเชิงอนุพันธ์ย่อยของการอนุรักษ์มวลและสมการเชิงอนุพันธ์ย่อยของการอนุรักษ์โมเมนตัม ที่แสดงในสมการ (2.1-2.2) โดยประยุกต์ใช้วิธีแครงก์-นิโคลสัน (Crank-Nicolson scheme) และระเบียบวิธีการคำนวณแบบแยกส่วน 4 ขั้นตอนโดยปริยาย (fully implicit four-step fractional method) สามารถเขียนใหม่ได้ดังนี้

$$\begin{aligned} \text{ขั้นตอนที่ 1, } \frac{\hat{u}_i - u_i^n}{\Delta t} + \frac{1}{2} \left( \frac{\partial}{\partial x_j} (\hat{u}_i \hat{u}_j + u_i^n u_j^n) \right) &= - \left( \frac{1}{\rho} \frac{\partial p^n}{\partial x_i} \right) + \frac{1}{2} v \frac{\partial}{\partial x_j} \left( \frac{\partial \hat{u}_i}{\partial x_j} + \frac{\partial u_i^n}{\partial x_j} \right) \\ &+ g_i^n (1 - \beta(T^n - T_0)) \end{aligned} \quad (3.15)$$

$$\text{ขั้นตอนที่ 2, } \frac{u_i^* - \hat{u}_i}{\Delta t} = \frac{1}{2} \left( \frac{1}{\rho} \frac{\partial p^n}{\partial x_i} \right) \quad (3.16)$$

$$\text{ขั้นตอนที่ 3, } \frac{\partial}{\partial x_i} \left( \frac{\partial p^{n+1}}{\partial x_i} \right) = \frac{2\rho}{\Delta t} \left( \frac{\partial u_i^*}{\partial x_i} \right) \quad (3.17)$$

$$\text{ขั้นตอนที่ 4, } \frac{u_i^{n+1} - u_i^*}{\Delta t} = -\frac{1}{2} \left( \frac{1}{\rho} \frac{\partial p^{n+1}}{\partial x_i} \right) \quad (3.18)$$

ในทำนองเดียวกันประยุกต์ใช้วิธีแบ่งย่อยช่วงเวลากับวิธีแครงก์-นิโคลสันเข้ากับสมการเชิงอนุพันธ์ย่อยของการอนุรักษ์พลังงาน ที่แสดงในสมการ (2.3) สามารถเขียนใหม่ได้ดังนี้

$$\begin{aligned} \text{ขั้นตอนที่ 5, } \frac{T^{n+1} - T^n}{\Delta t} + \frac{1}{2} \left( \frac{\partial}{\partial x_j} (T^{n+1} u_j^{n+1}) + \frac{\partial}{\partial x_j} (T^n u_j^n) \right) &= \\ \frac{1}{2} \tilde{k} \frac{\partial}{\partial x_j} \left( \frac{\partial T^{n+1}}{\partial x_j} + \frac{\partial T^n}{\partial x_j} \right) + \tilde{Q}^n \end{aligned} \quad (3.19)$$

### 3.1.3 ระเบียบวิธีสตรีมไลน์อัปวินด์เพทรอฟ-กาเลอร์คิน (SUPG)

สำหรับปัญหาของไหลเราจะเลือกใช้วิธีสตรีมไลน์อัปวินด์เพทรอฟ-กาเลอร์คิน (streamline upwind / Petrov-Galerkin formulation; SUPG) [22] วิธีดังกล่าวได้ทำการปรับปรุงฟังก์ชันถ่วงน้ำหนักเศษตักค้างโดยใช้ทิศทางของเวกเตอร์ความเร็วในการทำให้ฟังก์ชันดังกล่าวมีการถ่วงน้ำหนักไปในแนวของการไหลโดยที่วิธีการดังกล่าวสามารถให้ความถูกต้องในอันดับ 2 ( $2^{\text{nd}}$  order of accuracy) โดยกำหนดให้ฟังก์ชันถ่วงน้ำหนักอยู่ในรูปแบบดังนี้

$$W_\alpha = N_\alpha + \frac{\Delta t_c}{2} u_j \frac{\partial N_\alpha}{\partial x_j} \quad (3.20ก)$$

$$\text{เมื่อ } \Delta t_c = \frac{\sigma h_{\min}}{|U|} \quad (3.20ข)$$

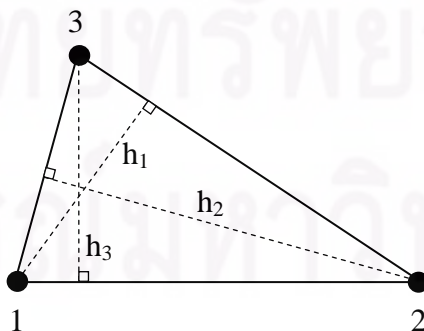
$$\sigma = \coth \frac{Pe}{2} - \frac{2}{Pe} \quad (3.20ค)$$

$$Pe = \frac{|U| h_{\min}}{2\nu} \quad \text{และ} \quad |U| = \sqrt{u^2 + v^2} \quad (3.20ง)$$

$$h_{\min} = \min(h_1, h_2, h_3) \quad (3.20จ)$$

โดยที่  $Pe$  คือ เพกเลตน์มเบอร์ (Peclet number)  $|U|$  คือค่าความเร็วเฉลี่ย และ  $h_{\min}$  คือขนาดความสูงของเอลิเมนต์ที่เล็กที่สุด โดยแสดงในรูปที่ 3.2

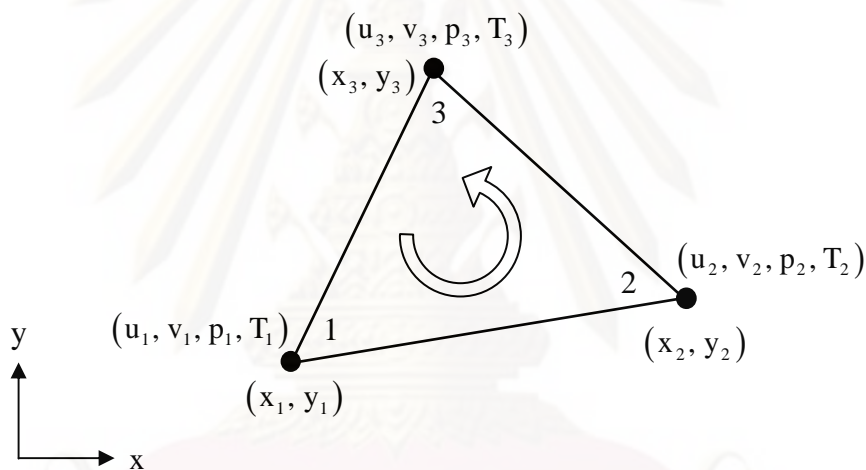
ปัญหาการถ่ายเทความร้อนและกลศาสตร์ของแข็งของโครงสร้างเราจะเลือกใช้ฟังก์ชันน้ำหนักเป็น  $W_i = N_i$  ซึ่งเรียกโดยทั่วไปว่าบับโนฟ-กาเลอร์คิน (Bubnov-Galerkin) โดยที่ฟังก์ชันการประมาณภายในเอลิเมนต์สามเหลี่ยมแสดงไว้ในสมการ (2.14-2.16)



รูปที่ 3.2 ขนาดของความสูงของเอลิเมนต์สามเหลี่ยมแบบสามจุดต่อ

### 3.1.4. ระเบียบวิธีถ่วงน้ำหนักเศษตกค้าง (method of weighted residual)

การนำเอาระเบียบวิธีไฟไนต์เอลิเมนต์มาประยุกต์ใช้ในการแก้ปัญหา ขั้นแรกจะต้องทำการแบ่งขอบเขตของปัญหาออกเป็นเอลิเมนต์ย่อย ๆ โดยในการทำวิทยานิพนธ์ครั้งนี้เลือกใช้เอลิเมนต์แบบสามเหลี่ยมสามจุดต่อ ดังแสดงในรูปที่ 3.3 ซึ่งประกอบไปด้วยตัวไม่ทราบค่าของความเร็วในทิศทางของแกน  $x$  ตัวไม่ทราบค่าของความเร็วในทิศทางของแกน  $y$  ตัวไม่ทราบค่าของความดัน  $p$  และตัวไม่ทราบค่าอุณหภูมิ  $T$  ที่จุดต่อทั้งสามของเอลิเมนต์สามเหลี่ยม จากนั้นกำหนดให้ลักษณะการกระจายของผลเฉลยโดยประมาณของความเร็ว ความดัน และอุณหภูมิบนเอลิเมนต์มีลักษณะดังนี้



รูปที่ 3.3 การแบ่งลักษณะของปัญหาออกเป็นเอลิเมนต์สามเหลี่ยมแบบสามจุดต่อ

$$u(x, y) = \sum N_i(x, y)u_i = [N]\{u\} \quad (3.21ก)$$

$$v(x, y) = \sum N_i(x, y)v_i = [N]\{v\} \quad (3.21ข)$$

$$p(x, y) = \sum N_i(x, y)p_i = [N]\{p\} \quad (3.21ค)$$

$$T(x, y) = \sum N_i(x, y)T_i = [N]\{T\} \quad (3.21ง)$$

ทำการประยุกต์ระเบียบวิธีถ่วงน้ำหนักเศษตกค้าง (method of weighted residuals) ลงในสมการเชิงอนุพันธ์ย่อยของการอนุรักษ์โมเมนตัม (3.15-3.19) จะได้

ขั้นตอนที่ 1, ใช้สำหรับหาค่าความเร็ว  $\hat{u}_i$

$$\begin{aligned} & \int_{\Omega} W_{\alpha} \left( \frac{\hat{u}_i - u_i^n}{\Delta t} \right) d\Omega + \frac{1}{2} \int_{\Omega} W_{\alpha} \left( \frac{\partial}{\partial x_j} (\hat{u}_i \hat{u}_j) + \frac{\partial}{\partial x_j} (u_i^n u_j^n) \right) d\Omega \\ &= - \int_{\Omega} W_{\alpha} \left( \frac{1}{\rho} \frac{\partial p^n}{\partial x_i} \right) d\Omega + \frac{\nu}{2} \int_{\Omega} W_{\alpha} \left( \frac{\partial}{\partial x_j} \left( \frac{\partial \hat{u}_i}{\partial x_j} + \frac{\partial u_i^n}{\partial x_j} \right) \right) d\Omega \\ & \quad + g_i^n \int_{\Omega} W_{\alpha} (1 - \beta(T^n - T_0)) d\Omega \end{aligned} \quad (3.22)$$

พิจารณาพจน์ทางด้านขวามือพจน์ที่ 2 ทำการประยุกต์ใช้ทฤษฎีบทของเกาส์ (Gauss's theorem) แสดงในสมการ (3.22) จะได้

$$\begin{aligned} \frac{\nu}{2} \int_{\Omega} W_{\alpha} \left( \frac{\partial}{\partial x_j} \left( \frac{\partial \hat{u}_i}{\partial x_j} + \frac{\partial u_i^n}{\partial x_j} \right) \right) d\Omega &= \frac{\nu}{2} \int_{\Omega} W_{\alpha} \left( \frac{\partial \hat{u}_i}{\partial x_j} \hat{n}_j + \frac{\partial u_i^n}{\partial x_j} n_j \right) d\Gamma \\ & \quad - \frac{\nu}{2} \int_{\Omega} \frac{\partial W_{\alpha}}{\partial x_j} \left( \frac{\partial \hat{u}_i}{\partial x_j} + \frac{\partial u_i^n}{\partial x_j} \right) d\Omega \end{aligned} \quad (3.23)$$

จากสมการ (3.23) พจน์ความเร็วของเงื่อนไขขอบทางด้านขวามือ เนื่องจากค่าอัตราการเปลี่ยนแปลงความเร็วที่เกิดบริเวณขอบไม่ส่งผลมากในการคำนวณหาค่าความเร็วกลาง  $\hat{u}_i$  จึงสามารถทำการประมาณค่าของความเร็วที่เกิดบริเวณขอบของปัญหาด้วยอนุกรมเทย์เลอร์ได้  $\hat{u}_i \approx u_i^n + O(\Delta t)$  จากนั้นแทนสมการ (3.23) ลงในสมการ (3.22) จะได้

$$\begin{aligned} & \int_{\Omega} W_{\alpha} \left( \frac{\hat{u}_i - u_i^n}{\Delta t} \right) d\Omega + \frac{1}{2} \int_{\Omega} W_{\alpha} \left( \frac{\partial}{\partial x_j} (\hat{u}_i \hat{u}_j + u_i^n u_j^n) \right) d\Omega + \frac{\nu}{2} \int_{\Omega} \frac{\partial W_{\alpha}}{\partial x_j} \left( \frac{\partial \hat{u}_i}{\partial x_j} + \frac{\partial u_i^n}{\partial x_j} \right) d\Omega \\ &= - \frac{1}{\rho} \int_{\Omega} W_{\alpha} \left( \frac{\partial p^n}{\partial x_i} \right) d\Omega + \nu \int_{\Omega} W_{\alpha} \left( \frac{\partial u_i^n}{\partial x_j} \hat{n}_j \right) d\Gamma \\ & \quad + g_i^n \int_{\Omega} W_{\alpha} (1 - \beta(T^n - T_0)) d\Omega \end{aligned} \quad (3.24)$$

ขั้นตอนที่ 2, ใช้สำหรับหาค่าความเร็ว  $u_i^*$

$$\int_{\Omega} W_{\alpha} \left( \frac{u_i^* - \hat{u}_i}{\Delta t} \right) d\Omega = \frac{1}{2\rho} \int_{\Omega} W_{\alpha} \left( \frac{\partial p^n}{\partial x_i} \right) d\Omega \quad (3.25)$$

ขั้นตอนที่ 3, สร้างสมการความดัน

ในการประดิษฐ์สมการไฟไนต์เอลิเมนต์สำหรับความดันนั้น สามารถทำได้โดยเริ่มจากการประยุกต์ใช้ระเบียบวิธีถ่วงน้ำหนักเศษตค่างกับสมการการอนุรักษ์มวล (3.5) และประยุกต์ใช้ทฤษฎีบทของเกาส์ จะได้สมการสำหรับเอลิเมนต์ดังนี้

$$\int_{\Omega} W_{\alpha} \left( \frac{\partial u_i^{n+1}}{\partial x_i} \right) d\Omega = - \int_{\Omega} \frac{\partial W_{\alpha}}{\partial x_i} (u_i^{n+1}) d\Omega + \int_{\Gamma} W_{\alpha} (u_i^{n+1} n_i) d\Gamma = 0 \quad (3.26)$$

จากนั้นนำสมการ (3.10) แทนลงในสมการ (3.26) จะได้

$$- \int_{\Omega} \frac{\partial W_{\alpha}}{\partial x_i} \left( u_i^* - \frac{\Delta t}{2} \left( \frac{1}{\rho} \frac{\partial p^{n+1}}{\partial x_i} \right) \right) d\Omega + \int_{\Gamma} W_{\alpha} (u_i^{n+1} n_i) d\Gamma = 0 \quad (3.27)$$

ทำการจัดรูปใหม่ได้ [7]

$$\int_{\Omega} \frac{\partial W_{\alpha}}{\partial x_i} \left( \frac{\partial p^{n+1}}{\partial x_i} \right) d\Omega = \frac{2\rho}{\Delta t} \int_{\Omega} \frac{\partial W_{\alpha}}{\partial x_i} u_i^* d\Omega - \frac{2\rho}{\Delta t} \int_{\Gamma} W_{\alpha} (u_i^{n+1} n_i) d\Gamma \quad (3.28)$$

จากสมการที่ (3.28) พจน์เงื่อนไขขอบเขตทางด้านขวามือ สามารถแทนค่าความเร็ว  $u_i^{n+1}$  ด้วยค่าความผิดพลาดอันดับสองของพจน์ที่เกี่ยวข้องกับเวลา โดยได้แสดงไว้ในสมการที่ (3.13) ดังนี้  $\hat{u}_i = u_i^{n+1} + O(\Delta t^2)$  จากนั้นละทิ้งพจน์อันดับสูง  $(\Delta t)^2$  แล้วสามารถเขียนสมการสำหรับหาค่าความดันใหม่ได้ดังนี้

$$\int_{\Omega} \frac{\partial W_{\alpha}}{\partial x_i} \left( \frac{\partial p^{n+1}}{\partial x_i} \right) d\Omega = \frac{2\rho}{\Delta t} \int_{\Omega} \frac{\partial W_{\alpha}}{\partial x_i} u_i^* d\Omega - \frac{2\rho}{\Delta t} \int_{\Gamma} W_{\alpha} (\hat{u}_i n_i) d\Gamma \quad (3.29)$$

ขั้นตอนที่ 4, ใช้สำหรับหาค่าความเร็วที่เหมาะสม  $u_i^{n+1}$  โดยใช้ความดันที่คำนวณได้จากขั้นตอนที่ 3

$$\int_{\Omega} W_{\alpha} \left( \frac{u_i^{n+1} - u_i^*}{\Delta t} \right) d\Omega = - \frac{1}{2\rho} \int_{\Omega} W_{\alpha} \left( \frac{\partial p^{n+1}}{\partial x_i} \right) d\Omega \quad (3.30)$$

ขั้นตอนที่ 5, ใช้สำหรับหาค่าอุณหภูมิ

$$\int_{\Omega} W_{\alpha} \left( \frac{T^{n+1} - T^n}{\Delta t} \right) d\Omega + \frac{1}{2} \int_{\Omega} W_{\alpha} \left( \frac{\partial}{\partial x_j} (T^{n+1} u_j^{n+1} + T^n u_j^n) \right) d\Omega =$$

$$\frac{\tilde{k}}{2} \int_{\Omega} W_{\alpha} \left( \frac{\partial}{\partial x_j} \left( \frac{\partial T^{n+1}}{\partial x_j} + \frac{\partial T^n}{\partial x_j} \right) \right) d\Omega + \int_{\Omega} W_{\alpha} \tilde{Q}^n d\Omega \quad (3.31)$$

ทำการประยุกต์ใช้ทฤษฎีบทของเกาส์ในสมการ (3.31) พจน์แรกทางด้านขวามือ แล้วจัดรูปใหม่ได้

$$\int_{\Omega} W_{\alpha} \left( \frac{T^{n+1} - T^n}{\Delta t} \right) d\Omega + \frac{1}{2} \int_{\Omega} W_{\alpha} \left( \frac{\partial}{\partial x_j} (T^{n+1} u_j^{n+1} + T^n u_j^n) \right) d\Omega$$

$$+ \frac{\tilde{k}}{2} \int_{\Omega} \frac{\partial W_{\alpha}}{\partial x_j} \left( \frac{\partial T^{n+1}}{\partial x_j} + \frac{\partial T^n}{\partial x_j} \right) d\Omega = \tilde{k} \int_{\Omega} W_{\alpha} \left( \frac{\partial T^n}{\partial x_j} \right) \hat{n}_j d\Gamma$$

$$+ \int_{\Omega} W_{\alpha} \tilde{Q}^n d\Omega \quad (3.32)$$

จากนั้นแทนค่าฟังก์ชันการประมาณค่าภายในของค่าความเร็ว ความดันและอุณหภูมิที่แสดงในสมการ (3.21ก-ง) ลงในสมการทั้ง 5 ขั้นตอนแล้วทำการจัดรูปจะได้

$$\text{ขั้นตอนที่ 1, } \left( \frac{[M]}{\Delta t} + \frac{1}{2} [H_m] \right) \{ \hat{u}_i \} = \left( \frac{[M]}{\Delta t} - \frac{1}{2} [H_m] \right) \{ u_i^n \}$$

$$- [G_i] \{ p \}^n + \{ R_{gi} \}^n + \{ R_{bi} \}^n \quad (3.33)$$

$$\text{ขั้นตอนที่ 2, } [M] \{ u_i \}^* = [M] \{ \hat{u}_i \} + \frac{\Delta t}{2} [G_i] \{ p \}^n \quad (3.34)$$

$$\text{ขั้นตอนที่ 3, } [K_p] \{ p \}^{n+1} = \{ R_i \}^* + \{ R_b \}^* \quad (3.35)$$

$$\text{ขั้นตอนที่ 4, } [M] \{ u_i \}^{n+1} = [M] \{ u_i \}^* - \frac{\Delta t}{2} [G_i] \{ p \}^{n+1} \quad (3.36)$$

$$\text{ขั้นตอนที่ 5, } \left( \frac{[M]}{\Delta t} + \frac{1}{2} [H_T] \right) \{ T \}^{n+1} = \left( \frac{[M]}{\Delta t} - \frac{1}{2} [H_T] \right) \{ T \}^n$$

$$+ \{ R_c \}^n + \{ R_q \}^n + \{ R_Q \}^n \quad (3.37)$$

โดยที่  $[H_k] = [C] + [K_k]$  เมื่อ ตัวห้อย  $k = m, T$

ดังนั้นในการแก้ปัญหาการไหลแบบหนืดแต่ไม่อันตัวภายใต้สภาวะไม่อยู่ตัว โดยรวม  
 อุณหภูมิ ด้วยระเบียบวิธีการคำนวณแบบแยกส่วน โดยมีขั้นตอนการคำนวณหาค่าตัวแปรอนุพันธ์  
 ที่จุดต่อต่าง ๆ จากเวลา  $t^n$  ไปยัง  $t^{n+1}$  6 ขั้นตอนตามลำดับดังต่อไปนี้

- 1) สมมติค่าเริ่มต้นของค่าความเร็ว ความดัน และอุณหภูมิที่เวลา  $t = t^n$
- 2) คำนวณหาค่าความเร็วกลาง  $\hat{u}_i$  และ  $u_i^*$  จากสมการที่ (3.33) และ (3.34)  
 ตามลำดับ
- 3) คำนวณหาค่าความดัน  $p^{n+1}$  จากสมการที่ (3.35) ที่เวลา  $t = t^n + \Delta t$
- 4) คำนวณหาค่าความเร็วที่ถูกต้อง  $u_i^{n+1}$  จากสมการที่ (3.36)
- 5) คำนวณหาค่าอุณหภูมิ  $T^{n+1}$  จากสมการที่ (3.37) ที่เวลา  $t = t^n + \Delta t$
- 6) กลับไปคำนวณขั้นตอนที่ 1-5 จนได้เวลาตามต้องการ

รายละเอียดของเมตริกซ์ต่าง ๆ ที่ใช้ในการคำนวณในแต่ละขั้นตอน ได้เขียนอยู่ในรูปแบบการของ  
 อินทิเกรตบนเอลิเมนต์และการอินทิเกรตที่ขอบดังต่อไปนี้

$$[M] = \int_{\Omega} \{W\} [N] d\Omega \quad (3.38)$$

$$[C] = \int_{\Omega} \{W\} \left( u_j \left[ \frac{\partial N}{\partial x_j} \right] \right) d\Omega \quad (3.39)$$

$$[K_m] = v \int_{\Omega} \left\{ \frac{\partial W}{\partial x_j} \right\} \left[ \frac{\partial N}{\partial x_j} \right] d\Omega \quad (3.40)$$

$$[K_T] = \tilde{k} \int_{\Omega} \left\{ \frac{\partial W}{\partial x_j} \right\} \left[ \frac{\partial N}{\partial x_j} \right] d\Omega \quad (3.41)$$

$$[G_i] = \frac{1}{\rho} \int_{\Omega} \{W\} \left[ \frac{\partial N}{\partial x_i} \right] d\Omega \quad (3.42)$$



$$[K_p] = \int_{\Omega} \left\{ \frac{\partial W}{\partial x_j} \right\} \left[ \frac{\partial N}{\partial x_j} \right] d\Omega \quad (3.43)$$

$$\{R_{gi}\} = g_i \int_{\Omega} \{W\} (1 - \beta(T - T_0)) d\Omega \quad (3.44)$$

$$\{R_i\} = \frac{2\rho}{\Delta t} \int_{\Omega} \left\{ \frac{\partial W}{\partial x_i} \right\} [N] \{u_i\} d\Omega \quad (3.45)$$

$$\{R_{bi}\} = \nu \int_{\Gamma} \{W\} \left( \frac{\partial u_i}{\partial x_j} \hat{n}_k \right) d\Gamma \quad (3.46)$$

$$\{R_b\} = -\frac{2\rho}{\Delta t} \int_{\Gamma} \{W\} (u_j \hat{n}_k) d\Gamma \quad (3.47)$$

$$\{R_c\} = \tilde{k} \int_{\Gamma} \{W\} \left( \frac{\partial T}{\partial x_j} \hat{n}_k \right) d\Gamma \quad (3.48)$$

$$\{R_q\} = \frac{1}{\rho c} \int_{\Gamma} \{W\} q_s d\Gamma \quad (3.49)$$

$$\{R_Q\} = \frac{1}{\rho c} \int_{\Omega} \{W\} Q d\Omega \quad (3.50)$$

การคำนวณหาตัวแปรอนุรักษ์ของการไหล ด้วยระเบียบวิธีการคำนวณแบบแยกส่วน นั้นเป็นการดำเนินก้าวไปกับเวลา (time marching) โดยเป็นการแก้ระบบสมการแบบปริยาย (implicit) ดังนั้นในระหว่างการคำนวณผลลัพธ์อาจเกิดการลู่ออก (diverged) ได้หากเลือกใช้ ช่วงเวลา (time step)  $\Delta t$  ที่สูงเกินไป จึงจำเป็นต้องมีขีดจำกัดของช่วงเวลาที่ใช้ในการคำนวณโดย ประเมินจาก ช่วงเวลาวิกฤติ (critical time step) [22] ดังต่อไปนี้

$$\Delta t = \min(\Delta t_{\text{conv}}, \Delta t_{\text{diff}}) \quad (3.51)$$

$$\Delta t_{\text{conv}} = \frac{h_{\min}}{|U|}, \quad \Delta t_{\text{diff}} = \frac{h_{\min}^2}{2k} \quad (3.52)$$

โดยที่  $h_{\min}$  คือ ขนาดความสูงของเอลิเมนต์ที่เล็กที่สุด

$|U|$  คือ ความเร็วสัมบูรณ์

### 3.2 สมการไฟไนต์เอลิเมนต์สำหรับปัญหาการถ่ายเทความร้อนทั้งภายในของไหลและโครงสร้าง

สำหรับสมการไฟไนต์เอลิเมนต์ของปัญหาการถ่ายเทความร้อนภายในของไหลและของแข็งนั้น สามารถนำสมการ (3.37) มาใช้ในการวิเคราะห์ปัญหาได้โดยทันที โดยถ้าจะทำการวิเคราะห์ปัญหาการถ่ายเทความร้อนภายในโครงสร้าง ก็เพียงแต่กำหนดให้ค่าความเร็วในแนวแกน  $x$  และ  $y$  ภายในสมการ (3.37) มีค่าเท่ากับศูนย์เท่านั้น ดังนั้นจะเห็นได้ว่าการหาสมการเอลิเมนต์และขั้นตอนที่ใช้ในการคำนวณที่ได้อธิบายไปในหัวข้อที่ 3.1 นั้น ไม่เพียงแต่จะสามารถใช้วิเคราะห์ปัญหาการไหลแบบหนึ่งมิติอัดตัวไม่ได้เท่านั้น ยังสามารถนำมาใช้วิเคราะห์ปัญหาการถ่ายเทความร้อนภายในตัวของไหลเองและภายในโครงสร้างได้โดยทันที ทำให้เกิดความสะดวกในการวิเคราะห์ปัญหาเป็นอย่างมาก

### 3.3 สมการไฟไนต์เอลิเมนต์สำหรับปัญหาความเค้นในโครงสร้างอันเนื่องมาจากอุณหภูมิ

การประดิษฐ์สมการไฟไนต์เอลิเมนต์สำหรับปัญหาความเค้นในโครงสร้างอันเนื่องมาจากอุณหภูมิสามารถกระทำได้โดยใช้วิธีถ่วงน้ำหนักเศษตกค้างเช่นกัน จากสมการเชิงอนุพันธ์แสดงความสมดุลของแรงในสองมิติดังแสดงในบทที่ 2

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} = 0 \quad (2.7ก)$$

$$\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} = 0 \quad (2.7ข)$$

ทำการประยุกต์วิธีถ่วงน้ำหนักเศษตกค้างกับสมการ (2.7ก) ดังนี้

$$\int_{\Omega} \{N\} \left( \frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} \right) d\Omega = 0 \quad (3.53)$$

ทำการประยุกต์ทฤษฎีบทของเกาส์ (Gauss's theorem) เข้ากับสมการ (3.53) จะได้

$$\int_{\Omega} \left( \frac{\partial N}{\partial x} \sigma_x + \frac{\partial N}{\partial y} \tau_{xy} \right) d\Omega = \int_{\Gamma} \{N\} (\sigma_x n_x + \tau_{xy} n_y) d\Gamma \quad (3.54)$$

ในทำนองเดียวกัน ทำการประยุกต์วิธีการถ่วงน้ำหนักเศษตกค้างกับสมการ (2.7ข) จะได้

$$\int_{\Omega} \left( \frac{\partial N}{\partial x} \tau_{xy} + \frac{\partial N}{\partial y} \sigma_y \right) d\Omega = \int_{\Gamma} \{N\} (\tau_{xy} n_x + \sigma_y n_y) d\Gamma \quad (3.55)$$

จากนั้นกำหนดให้แรงที่กระทำกับขอบนอกของโครงสร้าง เช่น ความดัน หรือแรงที่กระทำที่ผิว สามารถเขียนอยู่ในรูปแบบเมตริกซ์ดังนี้

$$\begin{Bmatrix} F_x \\ F_y \end{Bmatrix} = \begin{bmatrix} \sigma_x & \tau_{xy} \\ \tau_{xy} & \sigma_y \end{bmatrix} \begin{Bmatrix} n_x \\ n_y \end{Bmatrix} \quad (3.56)$$

แทนความสัมพันธ์ดังสมการ (3.56) ลงในสมการ (3.54) และ (3.55) แล้วจัดสมการใหม่ ให้อยู่ในรูปแบบเมตริกซ์ได้ดังนี้

$$\int_{\Omega} \begin{bmatrix} \frac{\partial N_i}{\partial x} & 0 & \frac{\partial N_i}{\partial y} \\ 0 & \frac{\partial N_i}{\partial y} & \frac{\partial N_i}{\partial x} \end{bmatrix} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} d\Omega = \int_{\Gamma} \begin{bmatrix} N_i & 0 \\ 0 & N_i \end{bmatrix} \begin{Bmatrix} F_x \\ F_y \end{Bmatrix} d\Gamma \quad (3.57)$$

โดยที่  $i = 1, 2, 3$

ซึ่งสามารถเขียนให้อยู่ในรูปแบบที่เข้าใจได้ง่ายขึ้นดังนี้

$$\int_{\Omega} [B]^T \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} d\Omega = \int_{\Gamma} [N]^T \begin{Bmatrix} F_x \\ F_y \end{Bmatrix} d\Gamma \quad (3.58)$$

โดยที่

$$[B] = \frac{1}{2A} \begin{bmatrix} b_1 & 0 & b_2 & 0 & b_3 & 0 \\ 0 & c_1 & 0 & c_2 & 0 & c_3 \\ c_1 & b_1 & c_2 & b_2 & c_3 & b_3 \end{bmatrix} \quad (3.59)$$

$$[N]_{(2 \times 6)} = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 \end{bmatrix} \quad (3.60)$$

จากความสัมพันธ์ระหว่างความเค้นและความเครียด

$$\begin{Bmatrix} \sigma \end{Bmatrix}_{(3 \times 1)} = [C]_{(3 \times 3)} \begin{Bmatrix} \varepsilon - \varepsilon_0 \end{Bmatrix}_{(3 \times 1)} \quad (3.61)$$

โดย  $\{\varepsilon_0\}$  คือค่าความเครียดขั้นต้นอันเนื่องมาจากอุณหภูมิ

$$\{\varepsilon_0\} = \{\alpha\} (T(x,y) - T_0) \quad (3.62)$$

และเมตริกซ์  $[C]$  และ  $\{\alpha\}$  ก็ขึ้นอยู่กับลักษณะของปัญหาว่าเป็นแบบความเค้นในระนาบ (plane stress) หรือความเครียดในระนาบ (plane strain) ซึ่งในกรณีของความเค้นในระนาบ

$$[C] = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad \text{และ} \quad \{\alpha\} = \begin{Bmatrix} \alpha \\ \alpha \\ 0 \end{Bmatrix} \quad (3.63)$$

และกรณีของความเครียดในระนาบ

$$[C] = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \quad \text{และ} \quad \{\alpha\} = \begin{Bmatrix} \alpha(1+\nu) \\ \alpha(1+\nu) \\ 0 \end{Bmatrix} \quad (3.64)$$

สำหรับความเครียดนั้นก็สามารถเขียนให้อยู่ในรูปแบบของค่าการเคลื่อนตัวได้ดังแสดงในสมการ (2.9)

$$\varepsilon_x = \frac{\partial u}{\partial x}; \quad \varepsilon_y = \frac{\partial v}{\partial y}; \quad \gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \quad (2.9)$$

ซึ่งสามารถเขียนอยู่ในรูปเมตริกซ์ได้ดังนี้

$$\begin{Bmatrix} \varepsilon \end{Bmatrix}_{(3 \times 1)} = \begin{bmatrix} B \end{bmatrix}_{(3 \times 6)} \begin{Bmatrix} \delta \end{Bmatrix}_{(6 \times 1)} \quad (3.65)$$

$$\text{โดยที่} \quad \begin{Bmatrix} \delta \end{Bmatrix}_{(6 \times 1)}^T = [u_1 \quad v_1 \quad u_2 \quad v_2 \quad u_3 \quad v_3] \quad (3.66)$$

แทนค่าสมการ (3.62), (3.65) และ (3.66) ลงในสมการ (3.58) แล้วทำการจัดรูปใหม่จะได้สมการ ไฟไนต์เอลิเมนต์เมตริกซ์ดังนี้

$$\int_{\Omega} [B]^T [C] [B] d\Omega \{\delta\} = \int_{\Omega} [B]^T [C] \{\varepsilon_0\} d\Omega + \int_{\Gamma} [N]^T \{F\} d\Gamma \quad (3.67)$$

เพื่อให้สามารถเข้าใจได้ง่ายต่อการประดิษฐ์โปรแกรมคอมพิวเตอร์ สามารถจัดรูปสมการ (3.67) ให้อยู่ในรูปแบบที่สั้นขึ้นได้ดังนี้

$$\begin{bmatrix} \mathbf{K} \end{bmatrix} \begin{Bmatrix} \delta \end{Bmatrix} = \begin{Bmatrix} \mathbf{F}_0 \end{Bmatrix} + \begin{Bmatrix} \mathbf{F}_T \end{Bmatrix} \quad (3.68)$$

โดยที่ 
$$\begin{bmatrix} \mathbf{K} \end{bmatrix} = \int_{\Omega} \begin{bmatrix} \mathbf{B} \end{bmatrix}^T \begin{bmatrix} \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{B} \end{bmatrix} d\Omega \quad (3.69)$$

$$\begin{Bmatrix} \mathbf{F}_0 \end{Bmatrix} = \int_{\Omega} \begin{bmatrix} \mathbf{B} \end{bmatrix}^T \begin{bmatrix} \mathbf{C} \end{bmatrix} \begin{Bmatrix} \epsilon_0 \end{Bmatrix} d\Omega \quad (3.70)$$

$$\begin{Bmatrix} \mathbf{F}_T \end{Bmatrix} = \int_{\Gamma} \begin{bmatrix} \mathbf{N} \end{bmatrix}^T \begin{Bmatrix} \mathbf{F} \end{Bmatrix} d\Gamma \quad (3.71)$$

ในที่นี้  $\begin{bmatrix} \mathbf{K} \end{bmatrix}$  แทนเมตริกซ์ของความแข็งแกร่ง  $\begin{Bmatrix} \mathbf{F}_0 \end{Bmatrix}$  แทนโหนดเวกเตอร์อันเนื่องมาจากความเค้นเริ่มต้น และ  $\begin{Bmatrix} \mathbf{F}_T \end{Bmatrix}$  แทนโหนดเวกเตอร์ของแรงที่ผิว

### 3.4 การประดิษฐ์ไฟไนต์เอลิเมนต์เมตริกซ์

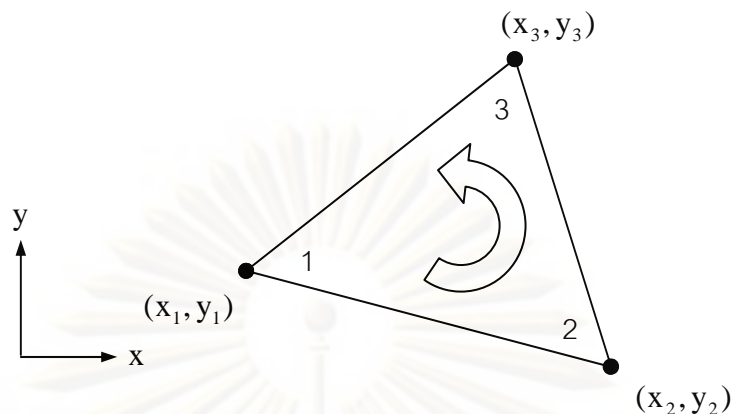
ไฟไนต์เอลิเมนต์เมตริกซ์ต่าง ๆ สามารถเขียนให้อยู่ในรูปแบบปิดได้ ซึ่งจะทำได้สามารถนำไปประดิษฐ์เป็นโปรแกรมคอมพิวเตอร์ได้ต่อเนื่อง โดยเริ่มจากการพิจารณาฟังก์ชันการประมาณภายในเอลิเมนต์ของเอลิเมนต์สามเหลี่ยม ซึ่งมีลักษณะการกระจายแบบแผ่นเรียบ (flat plane) รูปที่ 3.4 แสดงเอลิเมนต์สามเหลี่ยมแบบสามจุดต่อใดๆ ที่วางตัวอยู่ในโคออร์ดิเนต  $x - y$  ฟังก์ชันการประมาณภายในบนเอลิเมนต์สามเหลี่ยมสามารถเขียนให้อยู่ในรูปของโคออร์ดิเนตได้ดังต่อไปนี้

$$N_{\alpha}(x, y) = \frac{1}{2A}(a_{\alpha} + b_{\alpha}x + c_{\alpha}y) \quad , \quad \alpha=1, 2, 3 \quad (3.72)$$

เมื่อ  $A$  แทนพื้นที่ของเอลิเมนต์สามเหลี่ยมซึ่งคำนวณได้จากโคออร์ดิเนตของจุดต่อทั้งสามได้ดังนี้

$$A = \frac{1}{2}[x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2)] \quad (3.73)$$

ค่าสัมประสิทธิ์  $a_{\alpha}, b_{\alpha}, c_{\alpha}$  ในสมการ (3.72) คำนวณได้จาก



รูปที่ 3.4 เอลิเมนต์สามเหลี่ยมบนที่วางตัวอยู่ในโคออร์ดิเนต x-y

$$\begin{aligned}
 a_1 &= x_2 y_3 - x_3 y_2 & b_1 &= y_2 - y_3 & c_1 &= x_3 - x_2 \\
 a_2 &= x_3 y_1 - x_1 y_3 & b_2 &= y_3 - y_1 & c_2 &= x_1 - x_3 \\
 a_3 &= x_1 y_2 - x_2 y_1 & b_3 &= y_1 - y_2 & c_3 &= x_2 - x_1
 \end{aligned} \tag{3.74}$$

ดังนั้นค่าอนุพันธ์ของฟังก์ชันการประมาณภายในจากสมการ (3.72) คือ

$$\frac{\partial N_\alpha}{\partial x} = \frac{b_\alpha}{2A} \quad \text{และ} \quad \frac{\partial N_\alpha}{\partial y} = \frac{c_\alpha}{2A} \tag{3.75}$$

### 3.4.1 เอลิเมนต์เมตริกซ์สำหรับปัญหาการไหลแบบหนืด

ปัญหาการไหลแบบหนืดแต่ไม่อินตัวภายใต้สภาวะไม่อยู่ตัว โดยรวมคุณสมบัติจากที่กล่าวไว้ตอนต้นในสมการ (3.33-3.37) สามารถคำนวณหาเอลิเมนต์เมตริกซ์ต่าง ๆ ได้ดังนี้

ขั้นตอนที่ 1, แทนค่า  $[H_m] = [C] + [K_m]$  ลงในสมการ (3.33) ได้ดังนี้

$$\left( \frac{[M]}{\Delta t} + \frac{1}{2} [[C] + [K_m]] \right) \{ \hat{u}_i \} = \left( \frac{[M]}{\Delta t} - \frac{1}{2} [[C] + [K_m]] \right) \{ u_i^n \} - [G_i] \{ p \}^n + \{ R_{gi} \}^n + \{ R_{bi} \}^n \tag{3.76}$$

พิจารณาเมตริกซ์มวล [M]

$$[M] = \int_A \{W\} [N] dA \quad (3.38)$$

แทนค่าฟังก์ชันน้ำหนัก  $W = N$

$$\begin{aligned} [M] &= \int_A \{N\} [N] dA \\ &= \int_A \begin{Bmatrix} N_1 \\ N_2 \\ N_3 \end{Bmatrix} [N_1 \quad N_2 \quad N_3] dA \\ &= \int_A \begin{bmatrix} N_1 N_1 & N_1 N_2 & N_1 N_3 \\ N_2 N_1 & N_2 N_2 & N_2 N_3 \\ N_3 N_1 & N_3 N_2 & N_3 N_3 \end{bmatrix} dA \end{aligned}$$

การหาค่าอินทิกรัลของพิกัดพื้นที่สามารถคำนวณได้จากสูตร

$$\int_{\Omega} N_1^{\alpha} N_2^{\beta} N_3^{\gamma} d\Omega = \frac{\alpha! \beta! \gamma!}{(\alpha + \beta + \gamma + 2)!} \cdot 2A \quad (3.77)$$

$$\therefore \int_{\Omega} N_i^2 d\Omega = \frac{A}{6} \text{ และ } \int_{\Omega} N_i N_j d\Omega = \frac{A}{12} \text{ จะได้}$$

$$[M] = \frac{A}{12} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \quad (3.78)$$

พิจารณาเมตริกซ์ [C]

$$[C] = \int_{\Omega} \{W\} \left( u_j \left[ \frac{\partial N}{\partial x_j} \right] \right) d\Omega \quad (3.39)$$

แทนค่าฟังก์ชันน้ำหนักในสมการ (3.21ก) ลงในสมการ (3.41)

$$[C] = \int_{\Omega} \left\{ N_{\alpha} + \frac{\Delta t_e}{2} \bar{u}_j \frac{\partial N_{\alpha}}{\partial x_j} \right\}^n \left( u_j \left[ \frac{\partial N}{\partial x_j} \right] \right)^n d\Omega$$

$$[C] = \underbrace{\int_{\Omega} \{N_{\alpha}\}^n \left( u_j \left[ \frac{\partial N}{\partial x_j} \right] \right)^n d\Omega}_{[C_n]} + \underbrace{\frac{\Delta t_e}{2} \int_{\Omega} \left\{ \bar{u}_j \frac{\partial N_{\alpha}}{\partial x_j} \right\}^n \left( u_j \left[ \frac{\partial N}{\partial x_j} \right] \right)^n d\Omega}_{[C_w]}$$

ทำการจัดพจน์ต่าง ๆ แล้วสามารถเขียนให้อยู่ในรูปแบบของผลบวกเมตริกซ์ได้ดังนี้

$$[C] = [C_n] + [C_w] \quad (3.79)$$

โดยที่

$$\begin{aligned} [C_n] &= \int_{\Omega} \{N_{\alpha}\}^n \left( u_j \left[ \frac{\partial N}{\partial x_j} \right] \right)^n d\Omega \\ &= \int_{\Omega} \{N\}^n \left( u \left[ \frac{\partial N}{\partial x} \right] + v \left[ \frac{\partial N}{\partial y} \right] \right)^n d\Omega \\ &= \int_{\Omega} \{N\}^n \left( [N] \{u\} \left[ \frac{\partial N}{\partial x} \right] + [N] \{v\} \left[ \frac{\partial N}{\partial y} \right] \right)^n d\Omega \\ [C_n] &= \frac{1}{24} \left[ \begin{array}{ccc} b_1(2u_1 + u_2 + u_3) & b_2(2u_1 + u_2 + u_3) & b_3(2u_1 + u_2 + u_3) \\ b_1(u_1 + 2u_2 + u_3) & b_2(u_1 + 2u_2 + u_3) & b_3(u_1 + 2u_2 + u_3) \\ b_1(u_1 + u_2 + 2u_3) & b_2(u_1 + u_2 + 2u_3) & b_3(u_1 + u_2 + 2u_3) \end{array} \right]^n + \\ &\quad \left[ \begin{array}{ccc} c_1(2v_1 + v_2 + v_3) & c_2(2v_1 + v_2 + v_3) & c_3(2v_1 + v_2 + v_3) \\ c_1(v_1 + 2v_2 + v_3) & c_2(v_1 + 2v_2 + v_3) & c_3(v_1 + 2v_2 + v_3) \\ c_1(v_1 + v_2 + 2v_3) & c_2(v_1 + v_2 + 2v_3) & c_3(v_1 + v_2 + 2v_3) \end{array} \right]^n \end{aligned} \quad (3.80)$$

$$\begin{aligned} [C_w] &= \frac{\Delta t_e}{2} \int_{\Omega} \left\{ \bar{u}_j \frac{\partial N_{\alpha}}{\partial x_j} \right\}^n \left( u_j \left[ \frac{\partial N}{\partial x_j} \right] \right)^n d\Omega \\ &= \frac{\Delta t_e}{2} \int_{\Omega} \left\{ \bar{u} \frac{\partial N}{\partial x} + \bar{v} \frac{\partial N}{\partial y} \right\}^n \left( u \left[ \frac{\partial N}{\partial x} \right] + v \left[ \frac{\partial N}{\partial y} \right] \right)^n d\Omega \end{aligned}$$



$$\begin{aligned}
[C_w] = & \frac{\Delta t_e}{2} \left[ \underbrace{\int_{\Omega} \bar{u} \left\{ \frac{\partial N}{\partial x} \right\} [N] \{u\} \left[ \frac{\partial N}{\partial x} \right] d\Omega}_{[C_{w1}]} + \underbrace{\int_{\Omega} \bar{u} \left\{ \frac{\partial N}{\partial x} \right\} [N] \{v\} \left[ \frac{\partial N}{\partial y} \right] d\Omega}_{[C_{w2}]} \right. \\
& \left. + \underbrace{\int_{\Omega} \bar{v} \left\{ \frac{\partial N}{\partial y} \right\} [N] \{u\} \left[ \frac{\partial N}{\partial x} \right] d\Omega}_{[C_{w2}]} + \underbrace{\int_{\Omega} \bar{v} \left\{ \frac{\partial N}{\partial y} \right\} [N] \{v\} \left[ \frac{\partial N}{\partial y} \right] d\Omega}_{[C_{w3}]} \right]
\end{aligned}$$

$$\text{เมื่อ } \bar{u} = \frac{(u_1 + u_2 + u_3)}{3}, \quad \bar{v} = \frac{(v_1 + v_2 + v_3)}{3} \quad (3.81)$$

ทำการจัดพจน์ต่าง ๆ แล้วสามารถเขียนให้อยู่ในรูปแบบของผลบวกเมตริกซ์ได้ดังนี้

$$[C_w] = \frac{\Delta t_e}{2} ([C_{w1}] + [C_{w2}] + [C_{w3}] + [C_{w4}]) \quad (3.82)$$

พิจารณาพจน์  $[C_{w1}]$

$$\begin{aligned}
[C_{w1}] &= \int_{\Omega} \bar{u} \left\{ \frac{\partial N}{\partial x} \right\} [N] \{u\} \left[ \frac{\partial N}{\partial x} \right] d\Omega \\
&= \frac{\bar{u}^2}{4A^2} \int_{\Omega} \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \end{Bmatrix} [b_1 \ b_2 \ b_3] d\Omega
\end{aligned}$$

$$[C_{w1}] = \frac{\bar{u}^2}{4A} \begin{bmatrix} b_1^2 & b_1 b_2 & b_1 b_3 \\ b_1 b_2 & b_2^2 & b_2 b_3 \\ b_1 b_3 & b_2 b_3 & b_3^2 \end{bmatrix} \quad (3.82\alpha)$$

พิจารณาพจน์  $[C_{w2}]$

$$\begin{aligned}
[C_{w2}] &= \int_{\Omega} \bar{u} \left\{ \frac{\partial N}{\partial x} \right\} [N] \{v\} \left[ \frac{\partial N}{\partial y} \right] d\Omega \\
&= \frac{\bar{u}\bar{v}}{4A^2} \int_{\Omega} \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \end{Bmatrix} [c_1 \ c_2 \ c_3] d\Omega
\end{aligned}$$

$$[C_{w2}] = \frac{\bar{u}\bar{v}}{4A} \begin{bmatrix} b_1c_1 & b_1c_2 & b_1c_3 \\ b_2c_1 & b_2c_2 & b_2c_3 \\ b_3c_1 & b_3c_2 & b_3c_3 \end{bmatrix} \quad (3.82\text{ข})$$

พิจารณาพจน์  $[C_{w3}]$

$$\begin{aligned} [C_{w3}] &= \int_{\Omega} \bar{v} \left\{ \frac{\partial N}{\partial y} \right\} [N] \{u\} \left[ \frac{\partial N}{\partial x} \right] d\Omega \\ &= \frac{\bar{u}\bar{v}}{4A^2} \int_{\Omega} \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix} [b_1 \quad b_2 \quad b_3] d\Omega \end{aligned}$$

$$[C_{w3}] = \frac{\bar{u}\bar{v}}{4A} \begin{bmatrix} b_1c_1 & b_1c_2 & b_1c_3 \\ b_2c_1 & b_2c_2 & b_2c_3 \\ b_3c_1 & b_3c_2 & b_3c_3 \end{bmatrix} \quad (3.82\text{ค})$$

พิจารณาพจน์  $[C_{w4}]$

$$\begin{aligned} [C_{w4}] &= \int_{\Omega} \bar{v} \left\{ \frac{\partial N}{\partial y} \right\} [N] \{v\} \left[ \frac{\partial N}{\partial y} \right] d\Omega \\ &= \frac{\bar{v}^2}{4A^2} \int_{\Omega} \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix} [c_1 \quad c_2 \quad c_3] d\Omega \end{aligned}$$

$$[C_{w4}] = \frac{\bar{v}^2}{4A} \begin{bmatrix} c_1^2 & c_1c_2 & c_1c_3 \\ c_1c_2 & c_2^2 & c_2c_3 \\ c_1c_3 & c_2c_3 & c_3^2 \end{bmatrix} \quad (3.82\text{ง})$$

พิจารณาเมตริกซ์  $[K_m]$

$$\begin{aligned} [K_m] &= v \int_{\Omega} \left\{ \frac{\partial W}{\partial x_j} \right\} \left[ \frac{\partial N}{\partial x_j} \right] d\Omega \quad (3.40) \\ &= v \int_{\Omega} \left( \left\{ \frac{\partial W}{\partial x} \right\} \left[ \frac{\partial N}{\partial x} \right] + \left\{ \frac{\partial W}{\partial y} \right\} \left[ \frac{\partial N}{\partial y} \right] \right) d\Omega \end{aligned}$$

เนื่องจาก  $\frac{\partial W}{\partial x} = \frac{\partial N}{\partial x}$  และ  $\frac{\partial W}{\partial y} = \frac{\partial N}{\partial y}$

$$\begin{aligned}
 [K_m] &= v \int_{\Omega} \left( \left\{ \frac{\partial N}{\partial x} \right\} \left[ \frac{\partial N}{\partial x} \right] + \left\{ \frac{\partial N}{\partial y} \right\} \left[ \frac{\partial N}{\partial y} \right] \right) d\Omega \\
 &= \frac{v}{4A^2} \int_{\Omega} \left[ \begin{array}{c} b_1 \\ b_2 \\ b_3 \end{array} \right] [b_1 \ b_2 \ b_3] + \left[ \begin{array}{c} c_1 \\ c_2 \\ c_3 \end{array} \right] [c_1 \ c_2 \ c_3] d\Omega \\
 [K_m] &= \frac{v}{4A} \begin{bmatrix} b_1^2 + c_1^2 & b_1 b_2 + c_1 c_2 & b_1 b_3 + c_1 c_3 \\ \text{Sym} & b_2^2 + c_2^2 & b_2 b_3 + c_2 c_3 \\ & & b_3^2 + c_3^2 \end{bmatrix} \quad (3.83)
 \end{aligned}$$

พิจารณาพจน์ด้านขวามือของสมการ (3.76) เมตริกซ์  $[G_i] \{p\}^n$  กำหนดให้มีค่าเท่ากับ  $\{R_{pi}\}^n$  โดยที่  $i = x, y$  พิจารณาไหลดของความดันในทิศทางแกน  $x$

$$\begin{aligned}
 \{R_{px}\}^n &= \frac{1}{\rho} \int_{\Omega} \{W\} \left[ \frac{\partial N}{\partial x_i} \right] d\Omega \cdot \{p\}^n \\
 &= \frac{1}{\rho} \int_{\Omega} \left\{ N_{\alpha} + \frac{\Delta t_e}{2} \bar{u}_j \frac{\partial N_{\alpha}}{\partial x_j} \right\} \left[ \frac{\partial N}{\partial x_i} \right] d\Omega \cdot \{p\}^n \\
 \{R_{px}\}^n &= \underbrace{\frac{1}{\rho} \int_{\Omega} \{N\} \left[ \frac{\partial N}{\partial x} \right] d\Omega \cdot \{p\}^n}_{\{R_{pnx}\}^n} + \underbrace{\frac{\Delta t_e}{2\rho} \int_{\Omega} \left( \bar{u} \left\{ \frac{\partial N}{\partial x} \right\} + \bar{v} \left\{ \frac{\partial N}{\partial y} \right\} \right) \left[ \frac{\partial N}{\partial x} \right] d\Omega \cdot \{p\}^n}_{\{R_{pwx}\}^n}
 \end{aligned}$$

ทำการจัดพจน์ต่าง ๆ ได้ดังนี้

$$\{R_{px}\}^n = \{R_{pnx}\}^n + \{R_{pwx}\}^n \quad (3.84)$$

พิจารณาพจน์  $\{R_{pnx}\}^n$

$$\{R_{pnx}\}^n = \frac{1}{\rho} \int_{\Omega} \{N\} \left[ \frac{\partial N}{\partial x} \right] d\Omega \cdot \{p\}^n$$

$$\begin{aligned}
\{R_{\text{pnx}}\}^n &= \frac{1}{\rho(2A)} \int_{\Omega} \begin{Bmatrix} N_1 \\ N_2 \\ N_3 \end{Bmatrix} [b_1 \quad b_2 \quad b_3] \begin{Bmatrix} p_1 \\ p_2 \\ p_3 \end{Bmatrix}^n d\Omega \\
&= \frac{1}{2\rho A} \int_{\Omega} \begin{bmatrix} N_1 b_1 & N_1 b_2 & N_1 b_3 \\ N_2 b_1 & N_2 b_2 & N_2 b_3 \\ N_3 b_1 & N_3 b_2 & N_3 b_3 \end{bmatrix} \begin{Bmatrix} p_1 \\ p_2 \\ p_3 \end{Bmatrix}^n d\Omega \\
\{R_{\text{pnx}}\}^n &= \frac{1}{6\rho} (b_1 p_1^n + b_2 p_2^n + b_3 p_3^n) \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \tag{3.85}
\end{aligned}$$

พิจารณาพจน์  $\{R_{\text{pwx}}\}^n$

$$\begin{aligned}
\{R_{\text{pwx}}\}^n &= \frac{\Delta t_e}{2\rho} \int_{\Omega} \left( \bar{u} \left\{ \frac{\partial N}{\partial x} \right\} + \bar{v} \left\{ \frac{\partial N}{\partial y} \right\} \right) \left[ \frac{\partial N}{\partial x} \right] d\Omega \cdot \{p\}^n \\
&= \frac{\Delta t_e}{8A^2 \rho} \int_{\Omega} \left( \bar{u}^n \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \end{Bmatrix} + \bar{v}^n \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix} \right) [b_1 \quad b_2 \quad b_3] \begin{Bmatrix} p_1 \\ p_2 \\ p_3 \end{Bmatrix}^n d\Omega \\
&= \frac{\Delta t_e}{8A^2 \rho} \int_{\Omega} (b_1 p_1^n + b_2 p_2^n + b_3 p_3^n) \begin{Bmatrix} b_1 \bar{u}^n + c_1 \bar{v}^n \\ b_2 \bar{u}^n + c_2 \bar{v}^n \\ b_3 \bar{u}^n + c_3 \bar{v}^n \end{Bmatrix} d\Omega \\
\{R_{\text{pwx}}\}^n &= \frac{\Delta t_e}{8A\rho} (b_1 p_1^n + b_2 p_2^n + b_3 p_3^n) \begin{Bmatrix} b_1 \bar{u}^n + c_1 \bar{v}^n \\ b_2 \bar{u}^n + c_2 \bar{v}^n \\ b_3 \bar{u}^n + c_3 \bar{v}^n \end{Bmatrix} \tag{3.86}
\end{aligned}$$

ในทำนองเดียวกัน

$$\{R_{\text{py}}\}^n = \{R_{\text{pny}}\}^n + \{R_{\text{pwy}}\}^n \tag{3.87}$$

พิจารณาพจน์  $\{R_{\text{pnx}}\}^n$  และ  $\{R_{\text{pwx}}\}^n$  จะได้ดังนี้

$$\{R_{\text{pnx}}\}^n = \frac{1}{6\rho} (c_1 p_1^n + c_2 p_2^n + c_3 p_3^n) \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \tag{3.88}$$

$$\{\mathbf{R}_{\text{pwx}}\}^n = \frac{\Delta t_c}{8A\rho} (c_1 p_1^n + c_2 p_2^n + c_3 p_3^n) \begin{Bmatrix} b_1 \bar{u}^n + c_1 \bar{v}^n \\ b_2 \bar{u}^n + c_2 \bar{v}^n \\ b_3 \bar{u}^n + c_3 \bar{v}^n \end{Bmatrix} \quad (3.89)$$

พิจารณาเมตริกซ์  $\{\mathbf{R}_{\text{gi}}\}$

$$\{\mathbf{R}_{\text{gi}}\}^n = \mathbf{g}_i \int_{\Omega} \{\mathbf{W}\} (1 - \beta(T^n - T_0)) d\Omega \quad (3.44)$$

เมื่อ  $\mathbf{g}_x = -\mathbf{g}$

$$\begin{aligned} \{\mathbf{R}_{\text{gx}}\}^n &= \mathbf{g} \int_{\Omega} \{\mathbf{W}\} (\beta T^n - (1 + \beta T_0)) d\Omega \\ &= \mathbf{g} \int_{\Omega} \left\{ N_{\alpha} + \frac{\Delta t_c}{2} \bar{u}_j \frac{\partial N_{\alpha}}{\partial x_j} \right\}^n (\beta T^n - (1 + \beta T_0)) d\Omega \\ &= \underbrace{\mathbf{g} \int_{\Omega} \{\mathbf{N}\} (\beta T^n - (1 + \beta T_0)) d\Omega}_{\{\mathbf{R}_{\text{gnx}}\}^n} \\ &\quad + \underbrace{\frac{\mathbf{g} \Delta t_c}{2} \int_{\Omega} \left\{ \bar{u}_j \frac{\partial N_{\alpha}}{\partial x_j} \right\}^n (\beta T^n - (1 + \beta T_0)) d\Omega}_{\{\mathbf{R}_{\text{gwx}}\}^n} \end{aligned}$$

ทำการจัดพจน์ต่าง ๆ ได้ดังนี้

$$\{\mathbf{R}_{\text{gx}}\}^n = \{\mathbf{R}_{\text{gnx}}\}^n + \{\mathbf{R}_{\text{gwx}}\}^n \quad (3.90)$$

พิจารณาพจน์  $\{\mathbf{R}_{\text{gnx}}\}^n$

$$\begin{aligned} \{\mathbf{R}_{\text{gnx}}\}^n &= \mathbf{g} \int_{\Omega} \{\mathbf{N}\} (\beta T^n - (1 + \beta T_0)) d\Omega \\ &= \mathbf{g} \left[ \beta \int_{\Omega} \{\mathbf{N}\} T d\Omega - \int_{\Omega} \{\mathbf{N}\} (1 + \beta T_0) d\Omega \right] \end{aligned}$$

$$\begin{aligned}
&= g \left[ \beta \int_{\Omega} \begin{Bmatrix} N_1 \\ N_2 \\ N_3 \end{Bmatrix} [N_1 \quad N_2 \quad N_3] \begin{Bmatrix} T_1 \\ T_2 \\ T_3 \end{Bmatrix} d\Omega - \int_{\Omega} \begin{Bmatrix} N_1 \\ N_2 \\ N_3 \end{Bmatrix} (1 + \beta T_o) d\Omega \right] \\
\{R_{gnx}\}^n &= g \left[ \beta \int_{\Omega} \begin{bmatrix} N_1^2 & N_1 N_2 & N_1 N_3 \\ N_1 N_2 & N_2^2 & N_2 N_3 \\ N_1 N_3 & N_2 N_3 & N_3^2 \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \\ T_3 \end{Bmatrix} d\Omega - \int_{\Omega} \begin{Bmatrix} N_1 \\ N_2 \\ N_3 \end{Bmatrix} (1 + \beta T_o) d\Omega \right] \\
\text{เมื่อ } \int_{\Omega} N_i^2 d\Omega &= \frac{A}{6} \text{ และ } \int_{\Omega} N_i N_j d\Omega = \frac{A}{12} \\
\{R_{gnx}\}^n &= g \left[ \frac{\beta A}{12} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \\ T_3 \end{Bmatrix} - \frac{A}{3} (1 + \beta T_o) \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \right] \\
\{R_{gnx}\}^n &= \frac{gA}{12} \left( \beta \begin{Bmatrix} 2T_1^n + T_2^n + T_3^n \\ T_1^n + 2T_2^n + T_3^n \\ 2T_1^n + T_2^n + 2T_3^n \end{Bmatrix} - 4(1 + \beta T_o) \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \right) \quad (3.91)
\end{aligned}$$

พิจารณาพจน์  $\{R_{gwx}\}^n$

$$\begin{aligned}
\{R_{gwx}\}^n &= \frac{g\Delta t_e}{2} \int_{\Omega} \left\{ \bar{u}_j \frac{\partial N_{\alpha}}{\partial x_j} \right\}^n (\beta T^n - (1 + \beta T_o)) d\Omega \\
&= \frac{g\Delta t_e}{2} \int_{\Omega} \left( \bar{u} \left\{ \frac{\partial N_i}{\partial x} \right\} + \bar{v} \left\{ \frac{\partial N_i}{\partial y} \right\} \right)^n (\beta T^n - (1 + \beta T_o)) d\Omega
\end{aligned}$$

$$\{R_{gwx}\}^n = \frac{g\Delta t_e}{2} \left[ \beta \int_{\Omega} \left( \bar{u} \left\{ \frac{\partial N_i}{\partial x} \right\} + \bar{v} \left\{ \frac{\partial N_i}{\partial y} \right\} \right)^n [N] \{T\}^n d\Omega \right.$$

$$\left. - (1 + \beta T_o) \int_{\Omega} \left( \bar{u} \left\{ \frac{\partial N_i}{\partial x} \right\} + \bar{v} \left\{ \frac{\partial N_i}{\partial y} \right\} \right)^n d\Omega \right]$$

$$\{R_{gwx}\}^n = \frac{g\Delta t_e}{2} \left[ \frac{\beta}{2A} \int_{\Omega} \left( \bar{u} \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \end{Bmatrix} + \bar{v} \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix} \right)^n (N_1 T_1 + N_2 T_2 + N_3 T_3)^n d\Omega \right.$$

$$\begin{aligned}
 & -(1+\beta T_0) \frac{1}{2A} \int_{\Omega} \left[ \bar{\mathbf{u}} \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \end{Bmatrix} + \bar{\mathbf{v}} \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix} \right]^n d\Omega \\
 \left. \begin{aligned}
 \{ \mathbf{R}_{\text{gwx}} \}^n &= \frac{g\Delta t_c}{12} \left[ \beta (T_1^n + T_2^n + T_3^n) - 3(1+\beta T_0) \right] \begin{Bmatrix} b_1 \bar{\mathbf{u}}^n + c_1 \bar{\mathbf{v}}^n \\ b_2 \bar{\mathbf{u}}^n + c_2 \bar{\mathbf{v}}^n \\ b_3 \bar{\mathbf{u}}^n + c_3 \bar{\mathbf{v}}^n \end{Bmatrix}
 \end{aligned} \right\} \quad (3.92)
 \end{aligned}$$

ในทำนองเดียวกัน

$$\{ \mathbf{R}_{\text{gy}} \}^n = \{ \mathbf{R}_{\text{gny}} \}^n + \{ \mathbf{R}_{\text{gwy}} \}^n \quad (3.93)$$

$$\{ \mathbf{R}_{\text{gny}} \}^n = \frac{gA}{12} \left( \beta \begin{Bmatrix} 2T_1 + T_2 + T_3 \\ T_1 + 2T_2 + T_3 \\ T_1 + T_2 + 2T_3 \end{Bmatrix}^n - 4(1+\beta T_0) \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \right) \quad (3.94)$$

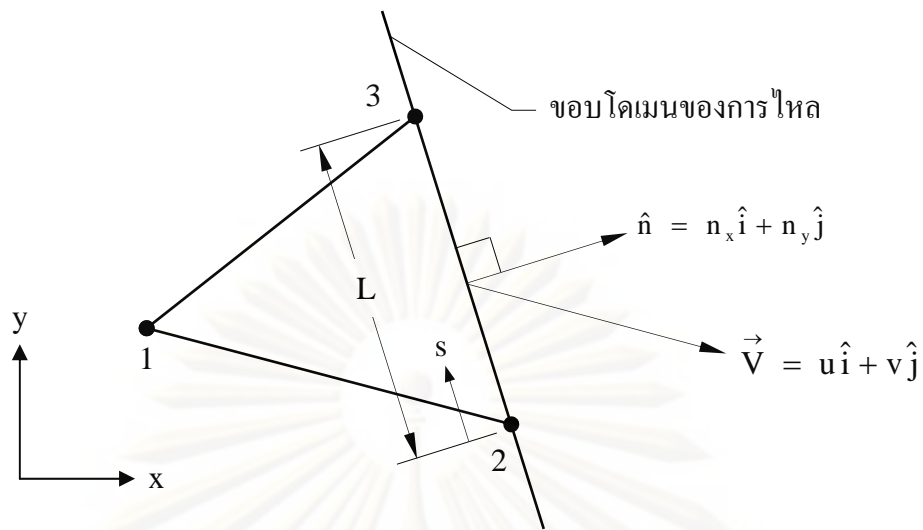
$$\{ \mathbf{R}_{\text{gwy}} \}^n = \frac{g\Delta t_c}{12} \left[ \beta (T_1^n + T_2^n + T_3^n) - 3(1+\beta T_0) \right] \begin{Bmatrix} b_1 \bar{\mathbf{u}}^n + c_1 \bar{\mathbf{v}}^n \\ b_2 \bar{\mathbf{u}}^n + c_2 \bar{\mathbf{v}}^n \\ b_3 \bar{\mathbf{u}}^n + c_3 \bar{\mathbf{v}}^n \end{Bmatrix} \quad (3.95)$$

เนื่องจากเป็นพจน์การอินทิเกรตขอบเขต ดังนั้นภายในขอบเขตของปัญหา พจน์เหล่านี้จะตัดกันหมดไป เหลือที่จะต้องคำนวณเฉพาะบริเวณขอบเขตของโดเมนเท่านั้น หากพิจารณาเอลิเมนต์สามเหลี่ยมดังแสดงในรูปที่ 3.5 ซึ่งมีขอบที่ประกอบด้วยจุดต่อ 2-3 เป็นขอบโดเมนของการไหล และเนื่องจากลักษณะการกระจายของฟังก์ชันความเร็วสำหรับเอลิเมนต์สามเหลี่ยมจะอยู่ในลักษณะเชิงเส้นตามขอบที่ประกอบด้วยจุดต่อ 2-3 ซึ่งมีความยาว  $L$  ดังที่แสดงในรูปที่ 3.5

ดังนั้นพจน์  $\{ \mathbf{R}_{\text{bi}} \}$  ในรูปแบบของสูตรอินทิเกรตสมการ (3.46) ที่สอดคล้องกับขอบคือ

$$\{ \mathbf{R}_{\text{bi}} \}^n = \nu \int_{\Gamma} \{ \mathbf{W} \} \left( \frac{\partial u_i}{\partial x_j} \hat{\mathbf{n}}_k \right) d\Gamma \quad (3.46)$$

แทนฟังก์ชันถ่วงน้ำหนัก  $\mathbf{W} = \mathbf{N}$



รูปที่ 3.5 การคำนวณพจน์  $\{R_{bi}\}$  ที่บริเวณขอบของโดเมนการไหล

$$\begin{aligned}
 \{R_{bu}\}^n &= v \int_0^L \{N\} \left( \frac{\partial u}{\partial x} \hat{n}_x + \frac{\partial u}{\partial y} \hat{n}_y \right) ds \\
 &= v \int_0^L \{N\} \left( \left[ \frac{\partial N}{\partial x} \right] \{u\} \hat{n}_x + \left[ \frac{\partial N}{\partial y} \right] \{u\} \hat{n}_y \right) ds \\
 &= \frac{v}{2A} \int_0^L \begin{Bmatrix} 0 \\ \frac{s}{L} \\ 1 - \frac{s}{L} \end{Bmatrix} \left( \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} \hat{n}_x + \begin{bmatrix} c_1 & c_2 & c_3 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} \hat{n}_y \right) ds \\
 \{R_{bu}\}^n &= \frac{vL}{4A} \left( \begin{Bmatrix} 0 \\ b_1 u_1^n + b_2 u_2^n + b_3 u_3^n \\ b_1 u_1^n + b_2 u_2^n + b_3 u_3^n \end{Bmatrix} \hat{n}_x + \begin{Bmatrix} 0 \\ c_1 u_1^n + c_2 u_2^n + c_3 u_3^n \\ c_1 u_1^n + c_2 u_2^n + c_3 u_3^n \end{Bmatrix} \hat{n}_y \right) \quad (3.96)
 \end{aligned}$$

ในทำนองเดียวกัน

$$\{R_{bv}\} = \frac{vL}{4A} \left( \begin{Bmatrix} 0 \\ b_1 v_1^n + b_2 v_2^n + b_3 v_3^n \\ b_1 v_1^n + b_2 v_2^n + b_3 v_3^n \end{Bmatrix} \hat{n}_x + \begin{Bmatrix} 0 \\ c_1 v_1^n + c_2 v_2^n + c_3 v_3^n \\ c_1 v_1^n + c_2 v_2^n + c_3 v_3^n \end{Bmatrix} \hat{n}_y \right) \quad (3.97)$$



$$\text{ขั้นตอนที่ 2, } [M]\{u_i\}^* = [M]\{\hat{u}_i\} + \frac{\Delta t}{2}[G_i]\{p\}^n \quad (3.34)$$

เอลิเมนต์เมตริกซ์  $[M]$  และ  $[G_i]\{p\}^n = \{R_{pi}\}^n$  ในขั้นตอนที่ 2 สามารถใช้เอลิเมนต์เมตริกซ์ที่คำนวณได้จากขั้นตอนที่ 1 ได้โดยตรง โดยแสดงดังสมการที่ (3.78) และ (3.84-3.89) ตามลำดับ

$$\text{ขั้นตอนที่ 3, } [K_p]\{p\}^{n+1} = \{R_i\}^* + \{R_b\}^* \quad (3.35)$$

พิจารณาพจน์  $[K_p]$

$$\begin{aligned} [K_p] &= \int_{\Omega} \left\{ \frac{\partial W}{\partial x_j} \right\} \left[ \frac{\partial N}{\partial x_j} \right] d\Omega \\ &= \int_{\Omega} \left( \left\{ \frac{\partial W}{\partial x} \right\} \left[ \frac{\partial N}{\partial x} \right] + \left\{ \frac{\partial W}{\partial y} \right\} \left[ \frac{\partial N}{\partial y} \right] \right) d\Omega \end{aligned} \quad (3.43)$$

$$\text{เนื่องจาก } \frac{\partial W}{\partial x} = \frac{\partial N}{\partial x} \quad \text{และ} \quad \frac{\partial W}{\partial y} = \frac{\partial N}{\partial y}$$

$$\begin{aligned} [K_p] &= \int_{\Omega} \left( \left\{ \frac{\partial N}{\partial x} \right\} \left[ \frac{\partial N}{\partial x} \right] + \left\{ \frac{\partial N}{\partial y} \right\} \left[ \frac{\partial N}{\partial y} \right] \right) d\Omega \\ &= \frac{1}{4A^2} \int_{\Omega} \left[ \begin{array}{c} \left\{ b_1 \right\} \\ \left\{ b_2 \right\} \\ \left\{ b_3 \right\} \end{array} \left[ b_1 \quad b_2 \quad b_3 \right] + \left\{ c_1 \right\} \left[ c_1 \quad c_2 \quad c_3 \right] \right] d\Omega \\ [K_p] &= \frac{1}{4A} \begin{bmatrix} b_1^2 + c_1^2 & b_1 b_2 + c_1 c_2 & b_1 b_3 + c_1 c_3 \\ \text{Sym} & b_2^2 + c_2^2 & b_2 b_3 + c_2 c_3 \\ & & b_3^2 + c_3^2 \end{bmatrix} \end{aligned} \quad (3.98)$$

พิจารณาพจน์  $[R_i]$

$$\{R_i\}^* = \frac{2\rho}{\Delta t} \int_{\Omega} \left\{ \frac{\partial W}{\partial x_i} \right\} [N] \{u_i^*\} d\Omega \quad (3.47)$$

$$\{R_u\}^* = \frac{2\rho}{\Delta t} \int_{\Omega} \left\{ \frac{\partial W}{\partial x_i} \right\} [N] \{u_i^*\} d\Omega$$

$$\begin{aligned} \{R_u\}^* &= \frac{2\rho}{\Delta t} \int_{\Omega} \frac{1}{2A} \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \end{Bmatrix} (N_1 u_1^* + N_2 u_2^* + N_3 u_3^*) d\Omega \\ \{R_u\}^* &= \frac{\rho}{3\Delta t} (u_1^* + u_2^* + u_3^*) \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \end{Bmatrix} \end{aligned} \quad (3.99)$$

ในทำนองเดียวกัน

$$\{R_v\}^* = \frac{\rho}{3\Delta t} (v_1^* + v_2^* + v_3^*) \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix} \quad (3.100)$$

พิจารณาพจน์  $\{R_b\}^*$  เนื่องจากเป็นพจน์การอินทิเกรตขอบเขต ทำนองเดียวกับพจน์  $\{R_{bi}\}$  ที่กล่าวไว้ตอนต้น ดังนั้นภายในขอบเขตของปัญหา พจน์เหล่านี้จะตัดกันหมดไป เหลือที่จะต้องคำนวณเฉพาะบริเวณขอบเขตของโดเมนเท่านั้น หากพิจารณา เอลิเมนต์สามเหลี่ยมดังแสดงในรูปที่ 3.5 ซึ่งมีขอบที่ประกอบด้วยจุดต่อ 2-3 เป็นขอบโดเมนของการไหลจะได้

$$\{R_b\}^* = -\frac{2\rho}{\Delta t} \int_{\Gamma} \{W\} (u_j^* \hat{n}_k) d\Gamma \quad (3.47)$$

$$\begin{aligned} \{R_b\}^* &= -\frac{2\rho}{\Delta t} \int_{\Gamma} \{N\} (u^* \cdot \hat{n}_x + v^* \cdot \hat{n}_y) d\Gamma \\ &= -\frac{2\rho}{\Delta t} \int_0^L \begin{Bmatrix} 1 - \frac{s}{L} \\ \frac{s}{L} \end{Bmatrix} \left( \begin{bmatrix} 1 - \frac{s}{L} & \frac{s}{L} \end{bmatrix} \begin{Bmatrix} u_2^* \\ u_3^* \end{Bmatrix} n_x + \begin{bmatrix} 1 - \frac{s}{L} & \frac{s}{L} \end{bmatrix} \begin{Bmatrix} v_2^* \\ v_3^* \end{Bmatrix} n_y \right) ds \\ \{R_b\}^* &= -\frac{2\rho}{\Delta t} \int_0^L \left[ \begin{array}{l} \left( \left( 1 - \frac{s}{L} \right)^2 u_2^* + \left( 1 - \frac{s}{L} \right) \frac{s}{L} u_3^* \right) \\ \left( \left( 1 - \frac{s}{L} \right) \frac{s}{L} u_2^* + \frac{s^2}{L^2} u_3^* \right) \end{array} \right] n_x ds \\ &\quad + \int_0^L \left[ \begin{array}{l} \left( \left( 1 - \frac{s}{L} \right)^2 v_2^* + \left( 1 - \frac{s}{L} \right) \frac{s}{L} v_3^* \right) \\ \left( \left( 1 - \frac{s}{L} \right) \frac{s}{L} v_2^* + \frac{s^2}{L^2} v_3^* \right) \end{array} \right] n_y ds \end{aligned}$$

$$\{R_b\}^* = -\frac{\rho L}{3\Delta t} \left[ \begin{array}{c} \left\{ \begin{array}{c} 2u_2^* + u_3^* \\ u_2^* + 2u_3^* \end{array} \right\} n_x + \left\{ \begin{array}{c} 2v_2^* + v_3^* \\ v_2^* + 2v_3^* \end{array} \right\} n_y \end{array} \right]$$

ดังนั้นพจน์  $\{R_b\}^*$  สำหรับเอลิเมนต์สามเหลี่ยมที่มีขอบที่ประกอบด้วยจุดต่อ 2-3 ซึ่งมีความยาว  $L$  อยู่ติดขอบโดเมนของการไหลคือ

$$\{R_b\}^* = -\frac{\rho L}{3\Delta t} \left[ \begin{array}{c} \left\{ \begin{array}{c} 0 \\ 2u_2^* + u_3^* \\ u_2^* + 2u_3^* \end{array} \right\} n_x + \left\{ \begin{array}{c} 0 \\ 2v_2^* + v_3^* \\ v_2^* + 2v_3^* \end{array} \right\} n_y \end{array} \right] \quad (3.101)$$

$$\text{ขั้นตอนที่ 4, } [M]\{u_i\}^{n+1} = [M]\{u_i\}^* - \frac{\Delta t}{2}[G_i]\{p\}^{n+1} \quad (3.36)$$

พิจารณาพจน์ทางด้านขวามือ  $[G_i]\{p\}^{n+1} = \{R_{pi}\}^{n+1}$  เอลิเมนต์เมตริกซ์ดังกล่าวสามารถประยุกต์ใช้เอลิเมนต์เมตริกซ์ที่ได้จากขั้นตอนที่ 1 โดยแทนค่าความดัน  $\{p\}^n$  ด้วย  $\{p\}^{n+1}$  และค่าความเร็ว  $\{u_i\}^n$  ด้วย  $\{u_i\}^*$  ที่คำนวณได้จากขั้นตอนที่ 2 โดยมีรายละเอียดดังนี้

$$\{R_{px}\}^{n+1} = \{R_{pnx}\}^{n+1} + \{R_{pwx}\}^{n+1} \quad (3.102)$$

โดยที่

$$\{R_{pnx}\}^{n+1} = \frac{1}{6\rho} (b_1 p_1^{n+1} + b_2 p_2^{n+1} + b_3 p_3^{n+1}) \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \quad (3.103)$$

$$\{R_{pwx}\}^n = \frac{\Delta t_c}{8A\rho} (b_1 p_1^{n+1} + b_2 p_2^{n+1} + b_3 p_3^{n+1}) \begin{Bmatrix} b_1 \bar{u}^* + c_1 \bar{v}^* \\ b_2 \bar{u}^* + c_2 \bar{v}^* \\ b_3 \bar{u}^* + c_3 \bar{v}^* \end{Bmatrix} \quad (3.104)$$

ในทำนองเดียวกัน

$$\{R_{py}\}^{n+1} = \{R_{pny}\}^{n+1} + \{R_{pwy}\}^{n+1} \quad (3.105)$$

โดยที่

$$\{R_{pny}\}^{n+1} = \frac{1}{6\rho} (c_1 p_1^{n+1} + c_2 p_2^{n+1} + c_3 p_3^{n+1}) \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \quad (3.106)$$

$$\{\mathbf{R}_{\text{pwy}}\}^{n+1} = \frac{\Delta t_e}{8A\rho} (c_1 p_1^{n+1} + c_2 p_2^{n+1} + c_3 p_3^{n+1}) \begin{Bmatrix} b_1 \bar{u}^* + c_1 \bar{v}^* \\ b_2 \bar{u}^* + c_2 \bar{v}^* \\ b_3 \bar{u}^* + c_3 \bar{v}^* \end{Bmatrix} \quad (3.107)$$

ขั้นตอนที่ 5, แทนค่า  $[\mathbf{H}_T] = [\mathbf{C}] + [\mathbf{K}_T]$  ลงในสมการ (3.37) ได้ดังนี้

$$\begin{aligned} \left( \frac{[\mathbf{M}]}{\Delta t} + \frac{1}{2} [[\mathbf{C}] + [\mathbf{K}_T]] \right) \{\mathbf{T}\}^{n+1} &= \left( \frac{[\mathbf{M}]}{\Delta t} - \frac{1}{2} [[\mathbf{C}] + [\mathbf{K}_T]] \right) \{\mathbf{T}\}^n \\ &+ \{\mathbf{R}_c\}^n + \{\mathbf{R}_q\}^n + \{\mathbf{R}_Q\}^n \end{aligned} \quad (3.108)$$

โดยเอลิเมนต์เมตริกซ์  $[\mathbf{M}]$  และ  $[\mathbf{C}]$  เหมือนในสมการที่ (3.78) และ (3.79-3.82) ตามลำดับ

พิจารณาพจน์  $[\mathbf{K}_T]$

$$[\mathbf{K}_T] = \tilde{k} \int_{\Omega} \left\{ \frac{\partial \mathbf{W}}{\partial x_j} \right\} \left[ \frac{\partial \mathbf{N}}{\partial x_j} \right] d\Omega \quad (3.41)$$

$$[\mathbf{K}_T] = \frac{\tilde{k}}{4A} \begin{bmatrix} b_1^2 + c_1^2 & b_1 b_2 + c_1 c_2 & b_1 b_3 + c_1 c_3 \\ & b_2^2 + c_2^2 & b_2 b_3 + c_2 c_3 \\ \text{Sym} & & b_3^2 + c_3^2 \end{bmatrix} \quad (3.109)$$

สำหรับพจน์  $\{\mathbf{R}_c\}^n$  ทางด้านขวามือของสมการ (3.108) นั้นพบว่าภายในขอบเขตของปัญหาพจน์เหล่านี้จะตัดกันหมดไปจึงไม่จำเป็นต้องหาค่า

พิจารณาพจน์  $\{\mathbf{R}_Q\}^n$

$$\{\mathbf{R}_Q\}^n = \frac{1}{\rho c} \int_{\Omega} \{\mathbf{W}\} Q d\Omega \quad (3.52)$$

$$\begin{aligned} \{\mathbf{R}_Q\}^n &= \frac{1}{\rho c} \int_{\Omega} \left\{ N_\alpha + \frac{\Delta t_e}{2} \bar{u}_j \frac{\partial N_\alpha}{\partial x_j} \right\}^n Q d\Omega \\ &= \frac{1}{\rho c} \int_{\Omega} \{\mathbf{N}\} Q d\Omega + \frac{\Delta t_e}{2\rho c} \int_{\Omega} \left( \bar{u} \frac{\partial \mathbf{N}}{\partial x} + \bar{v} \frac{\partial \mathbf{N}}{\partial y} \right)^n Q d\Omega \end{aligned}$$

$$\{R_Q\}^n = \frac{AQ}{3\rho c} \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} + \frac{Q\Delta t_e}{4\rho c} \begin{Bmatrix} b_1\bar{u}^n + c_1\bar{v}^n \\ b_2\bar{u}^n + c_2\bar{v}^n \\ b_3\bar{u}^n + c_3\bar{v}^n \end{Bmatrix} \quad (3.110)$$

เอลิเมนต์เมตริกซ์สำหรับวิเคราะห์ปัญหาความเค้นในโครงสร้างอันเนื่องมาจากอุณหภูมิ

จากสมการ (3.68)

$$\begin{Bmatrix} K \\ (6 \times 6) \end{Bmatrix} \begin{Bmatrix} \delta \\ (6 \times 1) \end{Bmatrix} = \begin{Bmatrix} F_0 \\ (6 \times 1) \end{Bmatrix} + \begin{Bmatrix} F_T \\ (6 \times 1) \end{Bmatrix} \quad (3.111)$$

โดยที่

$$\begin{Bmatrix} \delta \\ (6 \times 1) \end{Bmatrix} = [u_1 \quad v_1 \quad u_2 \quad v_2 \quad u_3 \quad v_3] \quad (3.112)$$

$$\begin{Bmatrix} K \\ (6 \times 6) \end{Bmatrix} = A t \begin{Bmatrix} B \\ (6 \times 3) \end{Bmatrix}^T \begin{Bmatrix} C \\ (3 \times 3) \end{Bmatrix} \begin{Bmatrix} B \\ (3 \times 6) \end{Bmatrix} \quad (3.113)$$

$$\begin{Bmatrix} F_0 \\ (6 \times 1) \end{Bmatrix} = A t \begin{Bmatrix} B \\ (6 \times 3) \end{Bmatrix}^T \begin{Bmatrix} C \\ (3 \times 3) \end{Bmatrix} \begin{Bmatrix} \epsilon_0 \\ (3 \times 1) \end{Bmatrix} \quad (3.114)$$

$$\begin{Bmatrix} F_T \\ (6 \times 1) \end{Bmatrix} = A t \begin{Bmatrix} B \\ (6 \times 3) \end{Bmatrix}^T \begin{Bmatrix} C \\ (3 \times 3) \end{Bmatrix} \begin{Bmatrix} \alpha \\ (3 \times 1) \end{Bmatrix} \left( \frac{T_1^n + T_2^n + T_3^n}{3} - T_0 \right) \quad (3.115)$$

เอลิเมนต์เมตริกซ์ ดังแสดงในสมการ (3.111-3.115) จะถูกคำนวณสำหรับทุกเอลิเมนต์ จากนั้นจึงนำเมตริกซ์เหล่านี้มารวมกันก่อให้เกิดระบบสมการรวม แล้วแรงภายนอกที่มากระทำจะถูกประยุกต์เข้าไปในระบบสมการรวมนี้ก่อนที่จะแก้สมการรวมดังกล่าวเพื่อหาผลลัพธ์ของการเสียรูปตามจุดต่อต่าง ๆ เมื่อได้ผลลัพธ์ของการเสียรูปของทุกจุดต่อแล้วก็สามารถคำนวณค่าความเค้นย่อยของแต่ละเอลิเมนต์ได้จาก

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \\ (3 \times 1) \end{Bmatrix} = \begin{Bmatrix} C \\ (3 \times 3) \end{Bmatrix} \left[ \begin{Bmatrix} B \\ (3 \times 6) \end{Bmatrix} \begin{Bmatrix} \delta \\ (6 \times 1) \end{Bmatrix} - \begin{Bmatrix} \alpha \\ (3 \times 1) \end{Bmatrix} \left( \frac{T_1^n + T_2^n + T_3^n}{3} - T_0 \right) \right] \quad (3.116)$$

อนึ่ง ความเค้นย่อยตามสมการ (3.116) นี้เป็นค่าความเค้นย่อยของแต่ละเอลิเมนต์ ในทางปฏิบัติ ค่าความเค้นของเอลิเมนต์นี้จะถูกแปลงให้อยู่ในรูปของค่าความเค้นที่จุดต่อเพื่อความสะดวกในการแสดงผลลัพธ์โดยโปรแกรมกราฟิก หลักการแปลงหาค่าความเค้นที่จุดต่อใดจุดต่อหนึ่ง อาจทำได้โดยการเฉลี่ยค่าความเค้นจากทุกเอลิเมนต์ที่ล้อมรอบจุดต่อนั้น

ไฟไนต์เอลิเมนต์เมตริกซ์ต่าง ๆ ที่ได้แสดงในหัวข้อนี้ สามารถนำไปประดิษฐ์เป็นโปรแกรมคอมพิวเตอร์สำหรับการวิเคราะห์ร่วมกันระหว่างการไหล อุณหภูมิ และโครงสร้างได้โดยตรง ซึ่งรายละเอียดของโปรแกรมคอมพิวเตอร์ที่ประดิษฐ์ขึ้นจะได้แสดงไว้ในบทที่ 4



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

## บทที่ 4

### โปรแกรมคอมพิวเตอร์

บทนี้จะกล่าวถึงรายละเอียดและขั้นตอนการคำนวณภายในโปรแกรมคอมพิวเตอร์ที่ประดิษฐ์ขึ้นให้สอดคล้องกับสมการไฟไนต์เอลิเมนต์ที่ได้แสดงไว้ในบทที่ 3 โดยโปรแกรมห้ถูกเขียนขึ้นมาด้วยภาษาฟอร์แทรน 90 (FORTRAN 90) ประกอบไปด้วยโปรแกรมที่สามารถนำไปวิเคราะห์ปัญหาการไหลแบบหนึ่งมิติไม่อัดตัวที่มีการถ่ายเทความร้อนระหว่างของไหลและโครงสร้างทั้งสถานะอยู่ตัวและไม่อยู่ตัว และโปรแกรมวิเคราะห์ปัญหาความเค้นอันเนื่องมาจากอุณหภูมิ โดยรายละเอียดต่าง ๆ ของโปรแกรมห้ดังนี้

#### 4.1 โปรแกรมคอมพิวเตอร์สำหรับวิเคราะห์ปัญหาการไหลแบบหนึ่งมิติอัดตัวไม่ได้ที่มีการถ่ายเทความร้อนระหว่างของไหลและโครงสร้างทั้งสถานะอยู่ตัวและไม่อยู่ตัว

โปรแกรมคอมพิวเตอร์สำหรับวิเคราะห์ปัญหาการไหลแบบหนึ่งมิติอัดตัวไม่ได้ที่มีการถ่ายเทความร้อนระหว่างของไหลและโครงสร้างนั้น ประกอบไปด้วยโปรแกรมหลัก (main program) และ 15 โปรแกรมย่อย (subroutines) โดยมีขั้นตอนการทำงานดังนี้

- 4.1.1 เริ่มต้นการทำงานภายในโปรแกรมหลัก [MAIN PROGRAM] โดยจะเรียกโปรแกรมย่อยแรก [READ\_INPUT] โดยในโปรแกรมย่อยดังกล่าวก็จะทำการอ่านข้อมูลของปัญหา เช่น จำนวนจุดต่อและจำนวนเอลิเมนต์ของปัญหาทั้งหมด, จำนวนรอบในการคำนวณ, ค่าคุณสมบัติต่าง ๆ ของของไหล ได้แก่ค่าความหนาแน่น ค่าความจุความร้อน สัมประสิทธิ์การนำความร้อนในของไหล ค่าความหนืด สัมประสิทธิ์การขยายตัว ค่าอุณหภูมิอ้างอิง ส่วนค่าคุณสมบัติต่าง ๆ ของของแข็ง ได้แก่ค่าความหนาแน่น ค่าความจุความร้อน สัมประสิทธิ์การนำความร้อนในของแข็ง ค่าปริมาณความร้อนที่ผลิตได้เองจากนั้นจะอ่านค่า พิกัดของจุดต่อ เงื่อนไขเริ่มต้นในแต่ละจุดต่อ ซึ่งได้แก่ ค่าความเร็ว  $u$  ในแนวแกน  $x$  ค่าความเร็ว  $v$  ในแนวแกน  $y$  ค่าความดัน  $p$  และค่าอุณหภูมิ  $T$  และหมายเลขจุดต่อที่ประกอบกันขึ้นเป็นเอลิเมนต์ และชนิดของเอลิเมนต์ว่าเป็นของแข็งหรือของไหล เป็นต้น

- 4.1.2 จากนั้นทำการคำนวณหาค่าขนาดของทุก ๆ เอลิเมนต์เพื่อคำนวณหาค่าเวลาที่เหมาะสม (time critical) ที่ใช้ในการคำนวณ จากนั้นก็เริ่มการคำนวณแบบทำซ้ำ โดยเรียกโปรแกรมย่อย [STEP1\_U] เพื่อทำการสร้างเอลิเมนต์เมตริกซ์ เมื่อสร้างเอลิเมนต์เมตริกซ์เสร็จแล้ว ก็ทำการประยุกต์เงื่อนไขขอบเขต และแก้ระบบสมการเพื่อหาค่าความเร็ว  $\hat{u}$  ในทำนองเดียวกันเรียกโปรแกรมย่อย [STEP1\_V] เพื่อคำนวณหาค่าความเร็ว  $\hat{v}$
- 4.1.3 ค่าความเร็วค่าใหม่ที่ได้จากขั้นตอน 4.1.2 จะถูกนำมาใช้ในโปรแกรมย่อย [STEP2\_UV] เพื่อสร้างเอลิเมนต์เมตริกซ์สำหรับหาค่าตัวแปรความเร็ว เมื่อสร้างเอลิเมนต์เมตริกซ์เสร็จ ก็จะทำการหาค่าความเร็ว  $u^*$  และ  $v^*$  ค่าใหม่
- 4.1.4 ค่าความเร็วค่าใหม่ที่ได้จากขั้นตอน 4.1.3 ก็จะถูกนำมาใช้ในโปรแกรมย่อย [STEP3\_P] เพื่อสร้างเอลิเมนต์เมตริกซ์สำหรับหาค่าตัวแปรความดัน เมื่อสร้างเอลิเมนต์เมตริกซ์เสร็จ และได้ประยุกต์เงื่อนไขขอบเขตแล้วก็จะทำการหาค่าความดันค่าใหม่
- 4.1.5 นำค่าความดันค่าใหม่ที่ได้ไปปรับปรุงค่าของความเร็ว โดยเรียกโปรแกรมย่อย [STEP4\_UV]
- 4.1.6 เรียกโปรแกรมย่อย [STEP5\_T] เพื่อทำการสร้างเอลิเมนต์เมตริกซ์ของสมการอนุรักษ์พลังงาน เมื่อสร้างเอลิเมนต์เมตริกซ์เสร็จแล้ว ก็ทำการประยุกต์เงื่อนไขขอบเขต และแก้ระบบสมการเพื่อหาค่าอุณหภูมิค่าใหม่
- 4.1.7 เมื่อทำการปรับปรุงค่าความเร็วเรียบร้อยแล้วก็จะเรียกโปรแกรมย่อย [ERROR] เพื่อตรวจสอบค่าความผิดพลาดที่เกิดขึ้นในแต่ละรอบของการคำนวณซ้ำ
- 4.1.8 จากนั้นจะเรียกโปรแกรมย่อย [WRITE\_OUTPUT] เพื่อทำการเขียนผลลัพธ์ในแต่ละรอบของการคำนวณซ้ำลงไปไฟล์สำรองที่เตรียมไว้



- 4.1.9 สุดท้ายภายในโปรแกรมหลัก [MAIN PROGRAM] จะตรวจสอบการลู่เข้าของผลลัพธ์ โดยถ้าค่าความเคื่อนมีค่าต่ำกว่าที่กำหนดหรือทำการคำนวณครบจำนวนรอบที่ตั้งไว้ก็ให้ดำเนินการตามข้อ 4.1.10 ต่อไป แต่ถ้าไม่ก็ต้องกลับไปทำการคำนวณซ้ำตั้งแต่ข้อ 4.1.2 เป็นต้นไปจนกว่าคำตอบจะลู่เข้า
- 4.1.10 สุดท้ายภายในโปรแกรมหลัก [MAIN PROGRAM] จะตรวจสอบการลู่เข้าของผลลัพธ์ โดยถ้าค่าความเคื่อนมีค่าต่ำกว่าที่กำหนดหรือทำการคำนวณครบจำนวนรอบที่ตั้งไว้ก็ให้ดำเนินการตามข้อ 4.1.10 ต่อไป แต่ถ้าไม่ก็ต้องกลับไปทำการคำนวณซ้ำตั้งแต่ข้อ 4.1.2 เป็นต้นไปจนกว่าคำตอบจะลู่เข้า
- 4.1.11 พิมพ์ค่าผลลัพธ์สุดท้ายที่คำนวณได้ ซึ่งได้แก่ค่าความเร็วในแนวแกนทั้งสอง ค่าความดัน และค่าอุณหภูมิทั้งภายในของไหลและโครงสร้างลงในไฟล์เพื่อนำไปใช้แสดงผลต่อไป

## 4.2 รายละเอียดของโปรแกรม

รายละเอียดของโปรแกรมแสดงไว้ในภาคผนวก ก

## 4.3 ลักษณะของไฟล์ข้อมูลที่โปรแกรมต้องการ

ลักษณะของข้อมูลที่โปรแกรมต้องการ สามารถแบ่งออกเป็น 6 ส่วนย่อยได้ดังนี้

### ส่วนที่ 1

ประโยคอธิบายกำกับลักษณะของไฟล์

บรรทัดแรก

ตัวเลขระบุจำนวนบรรทัดที่เป็นตัวอักษร

บรรทัดต่อไป

ประโยคต่าง ๆ ที่มีจำนวนบรรทัดเท่าที่ระบุไว้

ตัวอย่างเช่น:

1

FINITE ELEMENT DATA FOR FLUID FLOW

**ส่วนที่ 2**

ขนาดของปัญหา พร้อมค่าที่จะใช้ในการคำนวณ

บรรทัดแรก            คำระบุจำนวนจุดต่อ จำนวนเอลิเมนต์ เอลิเมนต์ที่ชอบ รอบการคำนวณ และค่าความผิดพลาดที่ยอมรับได้

บรรทัดต่อไป        ตัวเลขจำนวนจุดต่อ จำนวนเอลิเมนต์ เอลิเมนต์ที่ชอบ

ตัวอย่างเช่น:        NPOIN        NELEM        NBOU

1521            2888            6

หมายเหตุ: จำนวนเอลิเมนต์ที่ชอบ (NBOU) หมายถึง จำนวนเอลิเมนต์ที่อยู่บนขอบเขตของปัญหาซึ่งมีของไหลไหลผ่าน เช่นเอลิเมนต์ที่ชอบทางเข้าของการไหลในปัญหาที่จะพิจารณา เป็นต้น

**ส่วนที่ 3**

คุณสมบัติต่าง ๆ ของของไหล

บรรทัดแรก            คำระบุคุณสมบัติต่าง ๆ

บรรทัดต่อไป        ตัวเลขแสดงค่าความหนาแน่น ค่าความหนืด ค่าสัมประสิทธิ์การขยายตัว ค่าความร้อนจำเพาะ ค่าสัมประสิทธิ์การนำความร้อนของโครงสร้าง และของไหล ตามลำดับ สุดท้ายค่าอุณหภูมิอ้างอิง

ตัวอย่างเช่น:        (FLUID)        DEN        Cv        Kfluid        VIS        TH\_EXP        Tref

1            0.7            1            0.001        0.1            30

(SOLID)        DEN        Cv        Ksolid        Qgen

8030        503        16.27        0.0

**ส่วนที่ 4**

ลักษณะของจุดต่อ

บรรทัดแรก            คำระบุลักษณะของจุดต่อ

บรรทัดต่อไป        ตัวเลขแสดงหมายเลขจุดต่อ เงื่อนไขขอบเขตของความเร็วในทิศทาง x และ y พร้อมกับเงื่อนไขขอบเขตของความดัน ตำแหน่งจุดต่อในแกน x และ y และค่าความเร็วในแกน x, y และค่าความดัน

ตัวอย่างเช่น:

NODE	IBCU	IBCV	IBCP	IBCT	X	Y	U	V	P	T
1	1	1	0	0	-0.2	0.0	0.	0.	0.	0.
2	1	1	0	0	-0.16	0.0	0.	0.	0.	0.
3	1	1	0	0	-0.12	0.0	0.	0.	0.	0.
4	1	1	0	0	-0.08	0.0	0.	0.	0.	0.
5	1	1	0	0	-0.04	0.0	0.	0.	0.	0.
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
745	0	0	0	0	0.791	1.	0.	0.	0.	0.
746	0	0	0	0	0.833	1.	0.	0.	0.	0.
747	0	0	0	0	0.875	1.	0.	0.	0.	0.
748	0	0	0	0	0.916	1.	0.	0.	0.	0.
749	0	0	0	0	0.958	1.	0.	0.	0.	0.
750	0	0	0	0	1.0	1.	0.	0.	0.	0.

หมายเหตุ: เงื่อนไขขอบเขต IBC ของความเร็ว  $u$  ในทิศทาง  $x$ , ความเร็ว  $v$  ในทิศทาง  $y$ , เงื่อนไขขอบเขตของความดันและเงื่อนไขขอบเขตของอุณหภูมิหมายถึง

$IBC = 1$  จุดต่อนั้นถูกกำหนดให้มีค่าตามที่ให้ไว้ และไม่ต้องทำการคำนวณหาค่าที่จุดต่อดังกล่าว

$IBC = 0$  ให้ทำการคำนวณหาค่าที่จุดต่อดังกล่าว

(ถ้าจุดต่อใดเป็นของแข็ง ให้แทนค่าความเร็ว  $u$  และ  $v$  เป็นศูนย์)

### ส่วนที่ 5 ลักษณะของเอลิเมนต์

บรรทัดแรก      คำระบุลักษณะของเอลิเมนต์และชนิดของเอลิเมนต์

บรรทัดต่อ ๆ ไป      หมายเลขของจุดต่อทั้งสามในทิศทางแนวนอนที่ประกอบขึ้นเป็น  
เอลิเมนต์นั้น ๆ

ตัวอย่างเช่น:

ELEMENT NO.	I	J	K	LTYPE1	LTYP2
1	1	8	7	1	0
2	1	2	8	1	0
3	2	9	8	1	0
4	2	3	9	1	0
5	3	10	9	1	0
⋮	⋮	⋮	⋮	⋮	⋮
1388	723	724	748	0	0
1389	724	749	748	0	0
1390	724	725	749	0	0
1391	725	750	749	0	0
1392	725	726	750	0	0

หมายเหตุ: LTYPE1 คือเอลิเมนต์ที่เป็นของแข็ง

หมายถึง LTYPE1 = 1 คือเอลิเมนต์ที่เป็นของแข็งจะใช้ค่าสัมประสิทธิ์  
การนำความร้อนในของแข็ง Ksolid

LTYPE1 = 0 คือเอลิเมนต์ที่เป็นของไหลจะใช้ค่าสัมประสิทธิ์  
การนำความร้อนในของไหล Kfluid

LTYPE2 คือเอลิเมนต์ผลิตความร้อนได้เอง

หมายถึง LTYPE2 = 1 คือเอลิเมนต์ผลิตความร้อนได้เองใช้ค่า Qgen

LTYPE2 = 0 คือเอลิเมนต์ไม่สามารถผลิตความร้อนได้

(ซึ่งค่าต่าง ๆ Ksolid, Kfluid, Qgen อยู่ในส่วนที่ 3)

**ส่วนที่ 6** เอลิเมนต์ที่อยู่ขอบเขตของปัญหาที่มีการไหลไหลผ่านขอบด้านนั้น

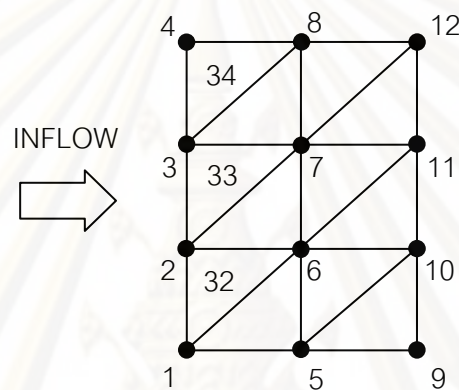
บรรทัดแรก คำระบุเอลิเมนต์ขอบเขต

บรรทัดต่อ ๆ ไป หมายเลขเอลิเมนต์ที่อยู่ขอบดังกล่าว และหมายเลขจุดต่อที่อยู่บน

ขอบของเอลิเมนต์นั้น ๆ (โดยมีจำนวนบรรทัดต้องเท่ากับค่า NBOU  
ที่ให้ไว้ในส่วนที่ 2)

ตัวอย่างเช่น:

NODE1	NODE2	ELEMENT NO.	BOU_TYPE
2	1	32	1
3	2	33	1
4	3	34	1



หมายเหตุ: BOU\_TYPE คือชนิดขอบของเอลิเมนต์ของการไหล

- หมายถึง
- BOU\_TYPE = 1 คือขอบของเอลิเมนต์เป็นช่องทางการไหลเข้า
  - BOU\_TYPE = 2 คือขอบของเอลิเมนต์เป็นช่องทางการไหลออก
  - BOU\_TYPE = 4 คือขอบของเอลิเมนต์เป็นแบบสมมาตร
  - BOU\_TYPE = 5 คือขอบของเอลิเมนต์เป็นผนังหรือกำหนด  
อุณหภูมิตลอดขอบ

#### 4.4 โปรแกรมคอมพิวเตอร์สำหรับวิเคราะห์ปัญหาความเค้นอันเนื่องมาจากอุณหภูมิ

โปรแกรมคอมพิวเตอร์สำหรับวิเคราะห์ปัญหาความเค้นอันเนื่องมาจากอุณหภูมิ ประกอบไปด้วยโปรแกรมหลัก (main program) และ 9 โปรแกรมย่อย (subroutines) โดยมีขั้นตอนในการทำงานดังต่อไปนี้

- 4.4.1 เริ่มต้นการทำงานภายในโปรแกรมหลัก [MAIN PROGRAM] โดยจะเรียกโปรแกรมย่อยแรก [READ\_INPUT] โดยในโปรแกรมย่อยดังกล่าวก็จะทำ

การอ่านข้อมูลของปัญหา เช่น จำนวนจุดต่อและจำนวนเอลิเมนต์ของปัญหา ค่าคุณสมบัติต่าง ๆ ของวัสดุ ตำแหน่งของจุดต่อต่าง ๆ เป็นต้น

- 4.4.2 จากนั้นก็เริ่มการคำนวณ โดยเรียกโปรแกรมย่อย [CST] โดยภายในโปรแกรมย่อยนี้จะเรียกโปรแกรมย่อย [CnK\_ELE] เพื่อทำการสร้างเอลิเมนต์เมตริกซ์ต่าง ๆ
- 4.4.3 เรียกโปรแกรมย่อย [ALL\_LOAD] เพื่อคำนวณเอลิเมนต์เมตริกซ์ของโหลดต่าง ๆ เช่น โหลดจากแรงภายนอก
- 4.4.4 จากนั้นส่งผ่านเอลิเมนต์เมตริกซ์ต่าง ๆ ที่คำนวณได้นี้ไปสร้างเมตริกซ์ใหญ่ของระบบสมการรวมโดยเรียกโปรแกรมย่อย [ASSMP]
- 4.4.5 กำหนดเงื่อนไขขอบเขตลงในระบบสมการรวม เช่น บางจุดต่อจะถูกตรึงแน่นเคลื่อนที่ไม่ได้หรือบางจุดต่อมีแรงภายนอกมากระทำโดยเรียกโปรแกรมย่อย [APPLBC]
- 4.4.6 แก้ระบบสมการรวมเพื่อหาค่าเคลื่อนตัว  $u$  และ  $v$  ที่ทุกจุดต่อโดยเรียกโปรแกรมย่อย [SYMPCG]
- 4.4.7 คำนวณค่าความเค้น  $\sigma_x$ ,  $\sigma_y$  และ  $\tau_{xy}$  ที่ทุกจุดต่อโดยเรียกโปรแกรมย่อย [STRESS]
- 4.4.8 สุดท้ายพิมพ์ค่าผลลัพธ์ที่คำนวณได้ ซึ่งได้แก่ค่าการเคลื่อนตัวของแต่ละจุดต่อในแนวแกนทั้งสอง ค่าความเค้นต่าง ๆ ลงในไฟล์เพื่อนำไปใช้แสดงผลต่อไป

#### 4.5 รายละเอียดของโปรแกรม

รายละเอียดของโปรแกรมแสดงไว้ในภาคผนวก ข

#### 4.6 ลักษณะของไฟล์ข้อมูลที่โปรแกรมต้องการ

ลักษณะข้อมูลที่โปรแกรมวิเคราะห์ความเค้นอันเนื่องมาจากอุณหภูมิต้องการสามารถจำแนกออกเป็น 6 ส่วนย่อยได้ดังต่อไปนี้

**ส่วนที่ 1** ประโยคอธิบายกำกับลักษณะของไฟล์

บรรทัดแรก ตัวเลขระบุจำนวนบรรทัดที่เป็นตัวอักษร

บรรทัดต่อไป ประโยคต่าง ๆ ที่มีจำนวนบรรทัดเท่าที่ระบุไว้

ตัวอย่างเช่น: 1

FINITE ELEMENT DATA FOR THERMAL STRESS

**ส่วนที่ 2** ขนาดของปัญหา พร้อมค่าที่จะใช้ในการคำนวณ

บรรทัดแรก คำระบุจำนวนจุดต่อ จำนวนเอลิเมนต์ และชนิดของปัญหา

บรรทัดต่อไป ตัวเลขจำนวนจุดต่อ จำนวนเอลิเมนต์ ตัวเลขระบุชนิดของปัญหา

ตัวอย่างเช่น: NPOIN NELEM STYPE

1521 2888 1

หมายเหตุ: ชนิดของปัญหามีความหมายดังนี้

STYPE = 0 กำหนดชนิดของปัญหาเป็นความเค้นระนาบ

STYPE = 1 กำหนดชนิดของปัญหาเป็นความเครียดระนาบ

**ส่วนที่ 3** คุณสมบัติต่าง ๆ ของปัญหา

บรรทัดแรก คำระบุคุณสมบัติต่าง ๆ

บรรทัดต่อไป ตัวเลขแสดงค่าโมดูลัสของความยืดหยุ่น อัตราส่วนของปัวส์ซง ระดับอุณหภูมิที่ไม่มีความเค้น สัมประสิทธิ์การขยายตัวเนื่องจากอุณหภูมิ และความหนาของของโครงสร้าง

ตัวอย่างเช่น: ELAS PR Tref ALPHA THICK

7.E+10 0.29 10. 0.2E-6 1.

**ส่วนที่ 4** ลักษณะของจุดต่อ

บรรทัดแรก คำระบุลักษณะของจุดต่อ

บรรทัดต่อ ๆ ไป ตัวเลขแสดงหมายเลขจุดต่อ เงื่อนไขขอบเขตในทิศทาง x และ y ตำแหน่งจุดต่อในแกน x และ y ค่าระยะการเคลื่อนตัวในแกน x และ y ค่าแรงดันจากภายนอกในแกน x และ y และค่าอุณหภูมิของจุดต่อนั้น

ตัวอย่างเช่น:

NODE	IBCU	IBCV	X	Y	U	V	SX	SY	T
1	1	0	1.033	0.	0.	0.	0.	0.	30.
2	0	0	1.066	0.	0.	0.	0.	0.	30.
3	1	1	1.1	0.	0.	0.	0.	0.	30.
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
51	1	0	1.82	0.	0.	0.	0.	0.	50.
52	0	0	1.833	0.	0.	0.	0.	0.	50.

หมายเหตุ: เงื่อนไขขอบเขต IBC ในทิศทาง x หรือ y

หมายถึง

$IBC = 1$  จุดต่อนั้นถูกตรึงไม่ให้เคลื่อนที่ในทิศนั้น

$IBC = 0$  คำนวณหาการเคลื่อนตัวที่จุดต่อดังกล่าวในทิศนั้น

### ส่วนที่ 5 ลักษณะของเอลิเมนต์

บรรทัดแรก ค่าระบุลักษณะของเอลิเมนต์

บรรทัดต่อ ๆ ไป หมายเลขเอลิเมนต์ และหมายเลขของจุดต่อทั้งหกในทิศทวนเข็มนาฬิกาที่ประกอบขึ้นเป็นเอลิเมนต์นั้น ๆ

ตัวอย่างเช่น:	ELEM	I	J	K
	1	42	79	87
	2	79	45	95
	3	45	68	11
	⋮	⋮	⋮	⋮
	119	32	69	77
	120	69	35	85



## บทที่ 5

### การตรวจสอบความถูกต้องของโปรแกรมคอมพิวเตอร์

ในบทนี้จะเป็นการตรวจสอบโปรแกรมคอมพิวเตอร์ที่ประดิษฐ์ขึ้น เพื่อใช้ในการวิเคราะห์ปัญหาการไหลแบบหนืดแบบไม่อัดตัวโดยมีการถ่ายเทความร้อนระหว่างของไหลและโครงสร้างที่สภาวะอยู่ตัวและไม่อยู่ตัว และการวิเคราะห์ปัญหาความเค้นอันเนื่องมาจากอุณหภูมิ โดยจะนำโปรแกรมที่ประดิษฐ์ขึ้นไปทดสอบกับปัญหาพื้นฐานที่มีผลเฉลยแน่นอนตรงและปัญหาที่มีผู้ทำการคำนวณอื่น ๆ ดังรายละเอียดต่อไปนี้

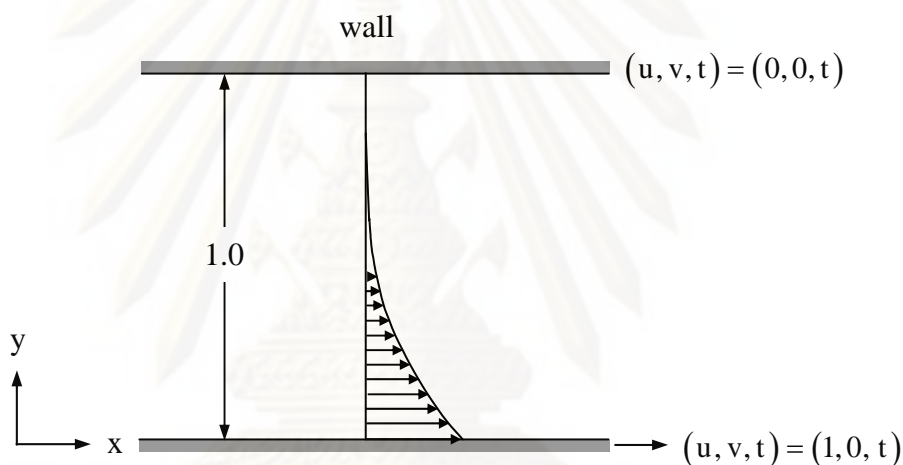
#### 5.1 ตรวจสอบโปรแกรมสำหรับวิเคราะห์ปัญหาการไหลแบบหนืดแบบไม่อัดตัวโดยมีการถ่ายเทความร้อนระหว่างของไหลและโครงสร้างที่สภาวะอยู่ตัวและไม่อยู่ตัว

หัวข้อนี้จะแบ่งการตรวจสอบออกเป็น 2 ส่วนย่อย ๆ คือส่วนแรกจะเป็นการตรวจสอบความถูกต้องกับปัญหาการไหลแบบหนืดแบบไม่อัดตัวโดยมีการถ่ายเทความร้อน โดยมีตัวอย่างปัญหาที่ใช้ในการตรวจสอบ 4 ตัวอย่างด้วยกัน ปัญหาแรกเป็นปัญหาการไหลระหว่างแผ่นคู่ขนานที่สภาวะไม่อยู่ตัว (Unsteady Couette flow) ซึ่งเป็นปัญหาที่มีผลเฉลยแน่นอนตรง ปัญหาที่สองปัญหาการไหลหมุนวนในช่องสี่เหลี่ยม (Unsteady cavity flow) ปัญหาที่สามปัญหาการไหลเนื่องจากการพาความร้อนแบบอิสระในช่องปิดรูปสี่เหลี่ยมจัตุรัส (Unsteady free convection in square enclosure) และปัญหาที่สี่เป็นปัญหาการถ่ายเทความร้อนผ่านทรงกระบอกกลม (Transient flow over a heated circular cylinder) โดยปัญหาที่ 2, 3, และ 4 เป็นปัญหาที่ใช้ในการตรวจสอบผลการคำนวณกับผลการวิเคราะห์ด้วยวิธีอื่น ส่วนที่สองจะเป็นการตรวจสอบความถูกต้องกับการไหลแบบหนืดแบบไม่อัดตัวโดยมีการถ่ายเทความร้อนระหว่างของไหลผ่านโครงสร้างที่สภาวะอยู่ตัวและไม่อยู่ตัว โดยมีตัวอย่างปัญหาที่ใช้ในการตรวจสอบ 2 ตัวอย่างด้วยกัน ปัญหาแรกเป็นปัญหาการไหลระหว่างแผ่นคู่ขนานพร้อมการถ่ายเทความร้อนกับโครงสร้าง (conjugate Couette flow problem in parallel plate channel) ซึ่งเป็นปัญหาที่มีผลเฉลยแน่นอนตรง ส่วนปัญหาที่สองเป็นปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน (conjugate counter flow heat exchanger) โดยจะได้ตรวจสอบผลการคำนวณกับผลการวิเคราะห์ด้วยวิธีอื่น

### 5.1.1 ปัญหาการไหลระหว่างแผ่นคู่ขนานที่สภาวะไม่อยู่ตัว

(Unsteady Couette flow)

ลักษณะของปัญหาการไหลระหว่างแผ่นคู่ขนานเนื่องจากความหนืดเป็นปัญหาการไหลแบบหนืดแต่ไม่อัดตัวระหว่างแผ่นเรียบสองแผ่น ซึ่งเป็นปัญหาที่สามารถใช้ในการตรวจสอบความถูกต้องเบื้องต้นของสมการไฟไนต์เอลิเมนต์ระหว่างโปรแกรมคอมพิวเตอร์ที่ประดิษฐ์ขึ้นกับผลเฉลยแม่นยำตรงได้ ลักษณะของปัญหานี้ได้แสดงในรูปที่ 5.1 โดยที่แผ่นล่างกำหนดให้มีการเคลื่อนที่ในแนวแกน  $x$  ด้วยความเร็ว  $u$  เท่ากับ 1.0 ที่เวลาใด ๆ ในขณะที่แผ่นบนกำหนดให้ไม่มีการเคลื่อนที่ โดยกำหนดให้ระยะห่างระหว่างแผ่นคู่ขนานเท่ากับ 1.0



รูปที่ 5.1 ลักษณะของปัญหาการไหลแบบหนืดระหว่างแผ่นคู่ขนาน

รูปแบบการกระจายตัวของความเร็วตามแนวแกน  $y$  ที่เวลาต่าง ๆ สำหรับปัญหานี้สามารถหาผลเฉลยแม่นยำตรงได้จาก White [24] โดยมีสมการดังนี้

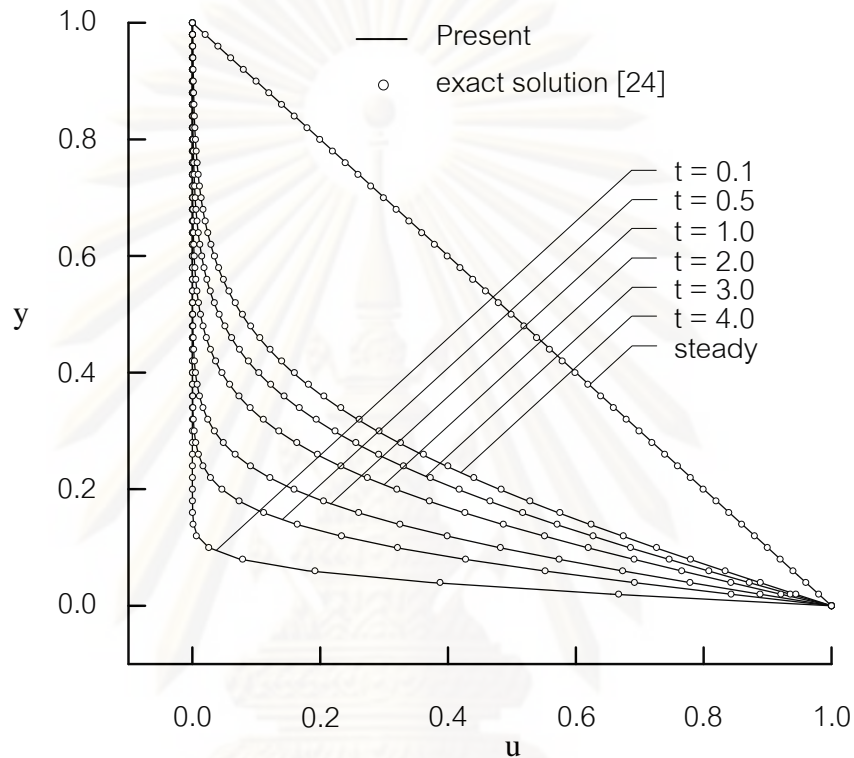
$$\frac{u}{U_0} = 1 - \operatorname{erf}\left(\frac{y}{2\sqrt{vt}}\right) = \operatorname{erfc}\left(\frac{y}{2\sqrt{vt}}\right) \quad (5.1)$$

เมื่อ

$$\operatorname{erfc}(\beta) = \frac{2}{\sqrt{\pi}} \int_0^\beta e^{-x^2} dx \quad (5.2)$$

ปัญหาดังกล่าวถูกนำไปวิเคราะห์โดยใช้โปรแกรมคอมพิวเตอร์ที่ได้ประดิษฐ์ขึ้น โดยเริ่มจากการสร้างรูปแบบจำลองทางไฟไนต์เอลิเมนต์โดยแบ่งโดเมนของปัญหาออกเป็น 2,601 จุดต่อ และ 5,000 เอลิเมนต์ โดยกำหนดให้ความดันมีค่าเท่ากับศูนย์ตลอดขอบในแนวตั้งทางด้าน

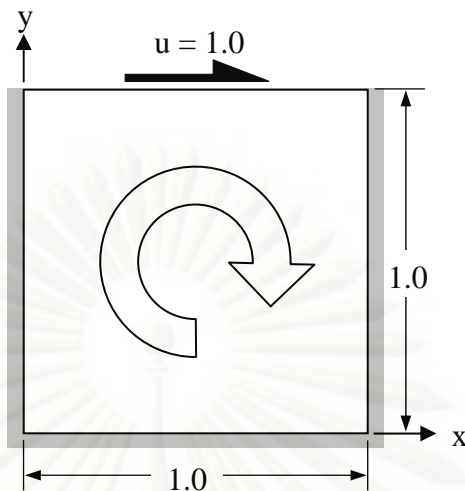
ซ้ายและขวาของโดเมนการคำนวณ ที่ค่าเรย์โนลด์นัมเบอร์,  $Re$  เท่ากับ 100 จากนั้นทำการเปรียบเทียบค่าความเร็วตามแนวแกน  $y$  กับผลเฉลยแม่นยำตรงที่เวลา 0.1, 0.5, 1.0, 2.0, 3.0, 4.0 และที่สภาวะอยู่ตัว โดยแสดงดังรูปที่ 5.2 ซึ่งผลที่ได้มีความสอดคล้องกันเป็นอย่างดี



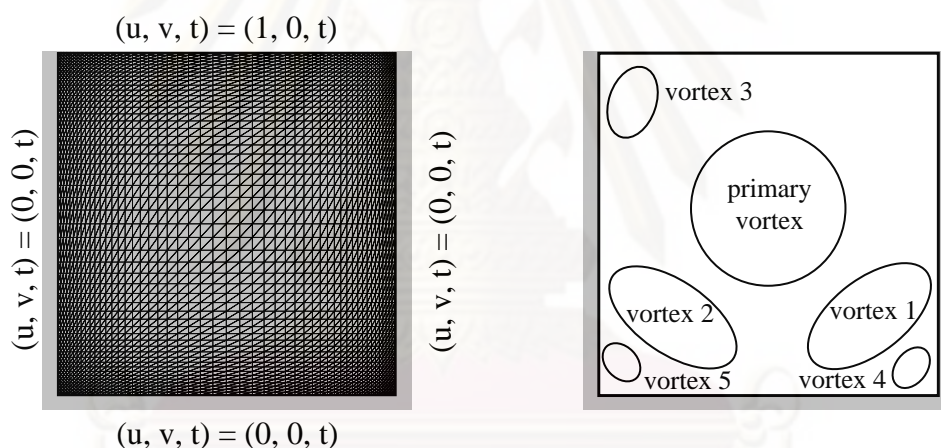
รูปที่ 5.2 เปรียบเทียบการกระจายตัวของความเร็วที่ได้จากการคำนวณกับผลเฉลยแม่นยำตรงของปัญหาการไหลระหว่างแผ่นคู่ขนานที่  $Re = 100$  ณ เวลาต่าง ๆ

### 5.1.2 ปัญหาการไหลหมุนวนในช่องสี่เหลี่ยม (Unsteady cavity flow)

ปัญหาการไหลหมุนวนภายในช่องสี่เหลี่ยมจัตุรัสขนาด  $1 \times 1$  เป็นอีกปัญหาหนึ่งที่น่าสนใจเพื่อทดสอบความเที่ยงตรงของโปรแกรม โดยกำหนดให้ตลอดขอบด้านบนมีความเร็วในแนวแกน  $x$  เท่ากับ 1 ณ เวลาใด ๆ ส่วนอีกสามด้านที่เหลือนั้นถูกกำหนดให้มีความเร็วในแนวแกนทั้งสองมีค่าเท่ากับศูนย์ ดังนั้นของไหลภายในช่องสี่เหลี่ยมจะเกิดการหมุนวนในทิศตามเข็มนาฬิกา ดังแสดงในรูปที่ 5.3 ซึ่งลักษณะของการไหลจะขึ้นอยู่กับค่าเรย์โนลด์นัมเบอร์ รูปแบบจำลองไฟไนต์เอลิเมนต์ดังกล่าวประกอบไปด้วย 2,601 จุดต่อ และ 5,000 เอลิเมนต์ ดังแสดงในรูปที่ 5.4



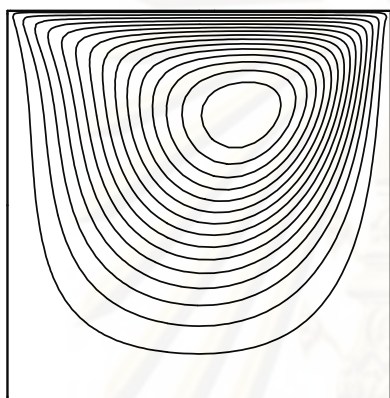
รูปที่ 5.3 ลักษณะของปัญหาการไหลหมุนวนภายในช่องแคบ



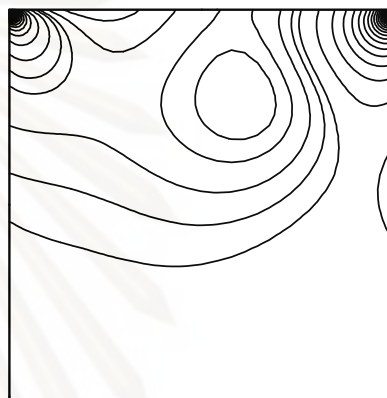
รูปที่ 5.4 แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการไหลหมุนวนภายในช่องแคบ และเงื่อนไขขอบเขต

จากนั้นทำการวิเคราะห์ปัญหาดังกล่าวที่ค่าเรย์โนลด์เท่ากับ 100 ผลลัพธ์ที่ได้จากการวิเคราะห์ปัญหาได้แสดงในรูปที่ 5.5 โดยรูปที่ 5.5(ก-ข) แสดงเส้นกระแสการไหล และเส้นชั้นของความดันตามลำดับ สำหรับรูปที่ 5.6 ได้แสดงลักษณะการกระจายตัวของความเร็ว  $u$  ที่ตำแหน่ง  $x$  เท่ากับ 0.5 ตลอด  $y$  ใด ๆ ที่ค่าเรย์โนลด์ 100 ที่เวลาเท่ากับ 1, 2, 3, 4 โดยทำการเปรียบเทียบกับผลการคำนวณของ Yagawa et al. [25] และที่สถานะคงตัว ทำการเปรียบเทียบกับผลการคำนวณของ Ghia et al. [26] ซึ่งผลการคำนวณที่ได้มีความสอดคล้องกันเป็นอย่างดี รูปที่ 5.7 แสดงเส้นกระแสการไหล เส้นชั้นของความดันและแสดงลักษณะการกระจายตัวของความเร็ว  $u$  ที่ตำแหน่ง  $x =$

0.5 ตลอด  $y$  ใด ๆ และการกระจายตัวของความเร็ว  $v$  ที่ตำแหน่ง  $y = 0.5$  ตลอด  $x$  ใด ๆ ที่ค่าเรย์โนลด์นัมเบอร์เท่ากับ 1,000, 5,000 และ 10,000 ตามลำดับ โดยทำการเปรียบเทียบกับผลการคำนวณของ Ghia et al. [26] ที่สภาวะอยู่ตัว จากตารางที่ 5.1 แสดงการเปรียบเทียบค่าฟังก์ชันการไหล (stream function) ที่ตำแหน่งจุดศูนย์กลางของการหมุน ณ ตำแหน่งต่าง ๆ ที่แสดงในรูปที่ 5.4 ทางด้านขวามือ โดยเปรียบเทียบกับผลการคำนวณของ Ghia, et al. [26] และ Choi, et al. [6] ผลที่ได้สอดคล้องกันเป็นอย่างดี

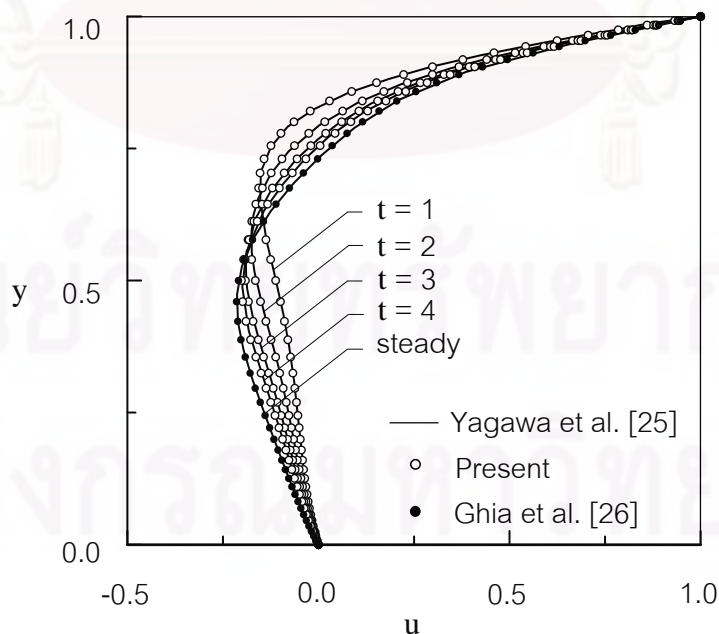


(ก) เส้นกระแสการไหล

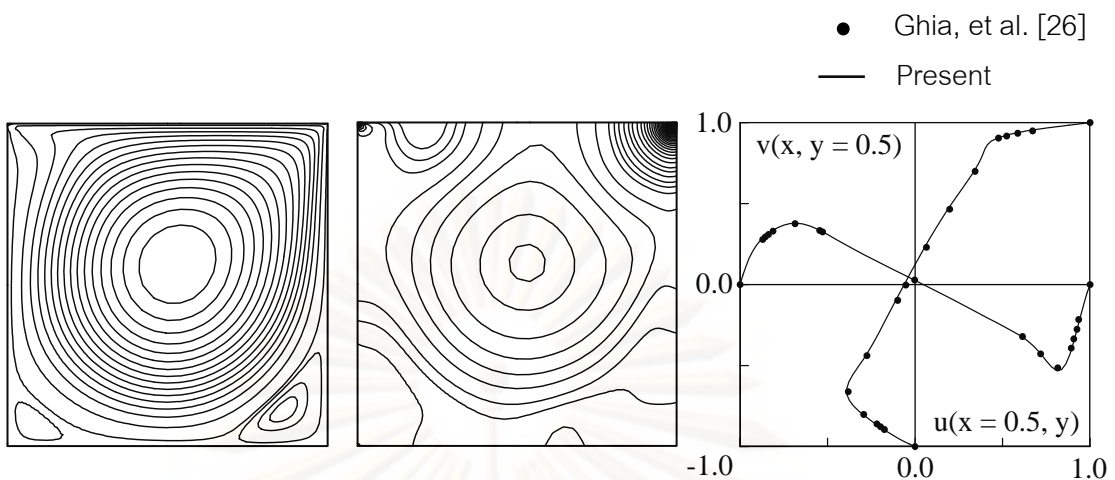


(ข) เส้นชั้นของความดัน

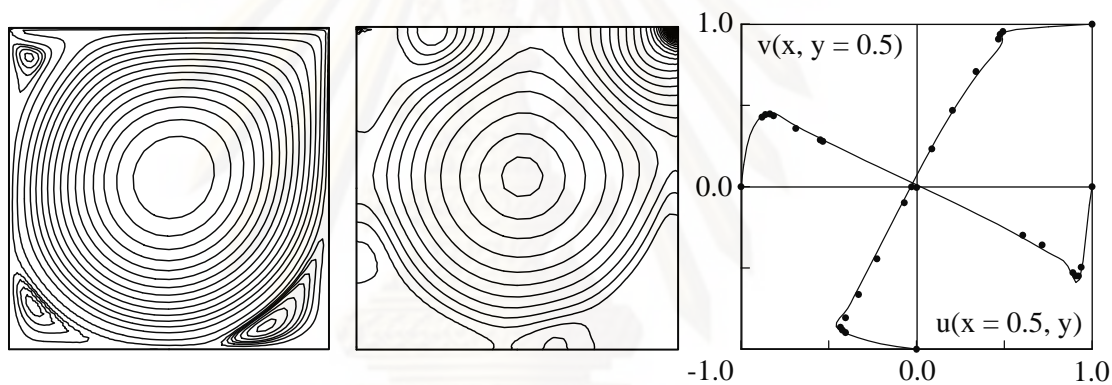
รูปที่ 5.5 สภาวะการไหลสำหรับปัญหาการไหลหมุนวนภายในช่องแคบที่ค่าเรย์โนลด์เท่ากับ 100



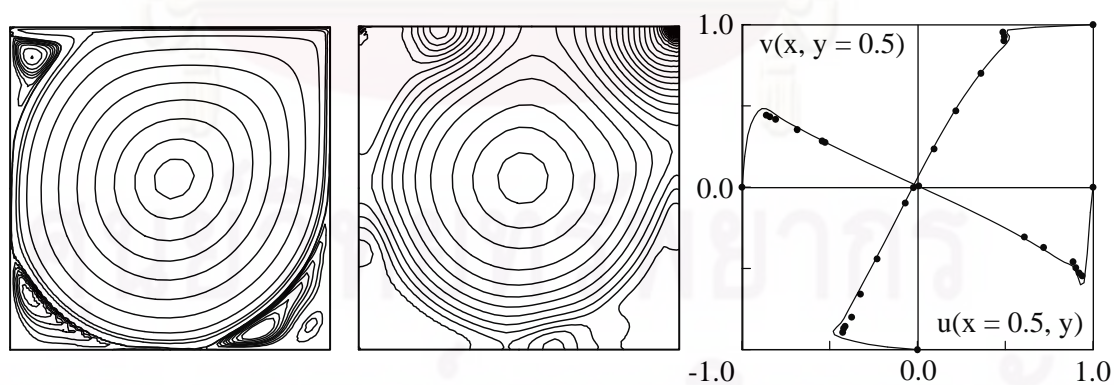
รูปที่ 5.6 การเปรียบเทียบการกระจายตัวของความเร็วที่ค่าเรย์โนลด์เท่ากับ 100 ณ ที่เวลาต่าง ๆ



(ก)  $Re = 1,000$



(ข)  $Re = 5,000$



(ค)  $Re = 10,000$

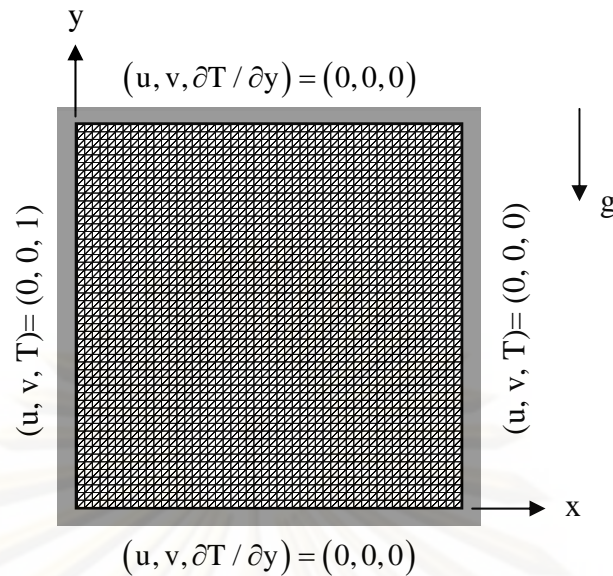
รูปที่ 5.7 แสดงเส้นกระแสการไหล เส้นชั้นของความดันและแสดงลักษณะการกระจายตัวของความเร็วที่ค่าเรย์โนลด์เท่ากับ (ก) 1,000 (ข) 5,000 และ (ค) 10,000 ตามลำดับ

ตารางที่ 5.1 การเปรียบเทียบค่าฟังก์ชันการไหลที่ตำแหน่งจุดศูนย์กลางของการหมุนโดย  
เปรียบเทียบกับวิธีอื่น ๆ

	Re	Ghia, et al. [26]	Choi, et al. [6]	Present
Primary vortex	400	-0.1139	-0.1135	-0.1140
	1,000	-0.1179	-0.1207	-0.1194
	5,000	-0.1190	-0.1255	-0.1248
	10,000	-0.1197	-0.1257	-0.1259
Vortex 1	400	6.423 E-4	6.010 E-4	6.043 E-4
	1,000	1.751 E-3	1.682 E-3	1.666 E-3
	5,000	3.083 E-3	3.749 E-3	3.118 E-3
	10,000	3.418 E-3	4.493 E-3	3.230 E-3
Vortex 2	400	1.419 E-5	1.030 E-5	1.411 E-5
	1,000	2.311 E-4	2.205 E-4	2.451 E-4
	5,000	1.361 E-3	1.334 E-3	1.488 E-3
	10,000	1.518 E-3	1.579 E-3	1.822 E-3
Vortex 3	5,000	1.456 E-3	1.288 E-3	1.390 E-3
	10,000	2.421 E-3	2.372 E-3	2.600 E-3

### 5.1.3 ปัญหาการไหลเนื่องจากการพาความร้อนแบบอิสระในช่องปิดรูปสี่เหลี่ยมจัตุรัส (Unsteady free convection in square enclosure)

ในปัญหานี้จะทำการเปรียบเทียบผลลัพธ์เชิงตัวเลขกับนักวิจัยท่านอื่นเพื่อทดสอบความเที่ยงตรงของโปรแกรม โดยปัญหานี้ประกอบไปด้วยช่องสี่เหลี่ยมจัตุรัสขนาดกว้าง 1x1 ภายในบรรจุของไหลอยู่ ผนังด้านบนและผนังด้านล่างเป็นฉนวน ส่วนผนังด้านซ้ายมีอุณหภูมิสูงเท่ากับ 1 ในขณะที่ผนังด้านขวามีอุณหภูมิต่ำเท่ากับ 0 ซึ่งความแตกต่างของอุณหภูมิจะทำให้ของไหลเกิดการหมุนวนขึ้นในช่องสี่เหลี่ยมจัตุรัส และมีแรงเนื่องจากแรงโน้มถ่วงดังแสดงในรูปที่ 5.8 รวมทั้งแสดงรูปแบบไฟไนต์เอลิเมนต์ซึ่งประกอบไปด้วย 2,601 จุดต่อ และ 5,000 เอลิเมนต์



รูปที่ 5.8 ปัญหาการไหลเนื่องจากการพาความร้อนในช่องสี่เหลี่ยมจัตุรัส

ในขั้นตอนการวิเคราะห์สภาวะการไหลสำหรับปัญหานี้ได้กำหนดให้พรีนัมเบอร์ (Prandtl Number;  $Pr$ ) มีค่าเท่ากับ 0.7 ค่าพรีนัมเบอร์นี้แสดงอัตราส่วนของการแพร่กระจายของโมเมนตัม (momentum diffusivity) กับการแพร่กระจายของพลังงานความร้อน (thermal diffusivity) [20] ดังนี้

$$Pr = \frac{\mu c}{k} = \frac{\nu}{\alpha} \quad (5.3)$$

โดยที่  $\alpha$  แทนค่าการแพร่กระจายของความร้อนของของไหล ซึ่งเป็นอัตราส่วนของการนำความร้อนกับความสามารถในการจุความร้อนดังสมการ

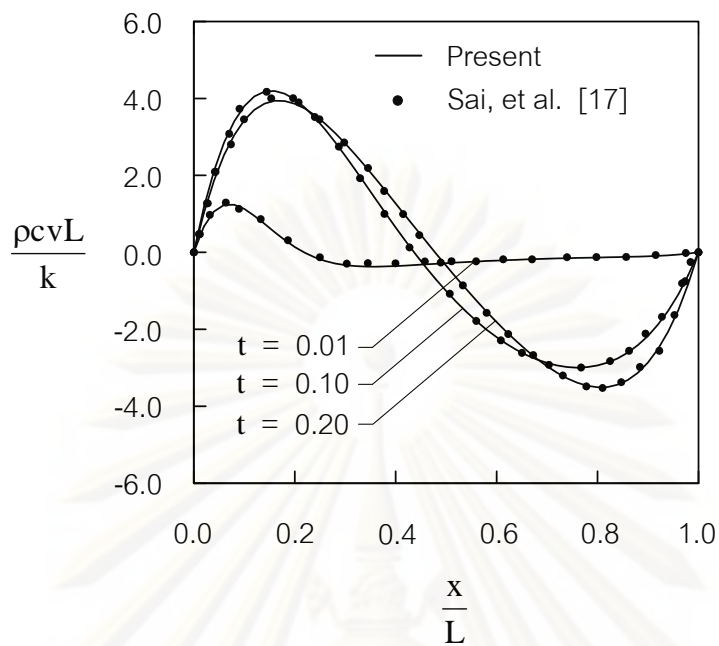
$$\alpha = \frac{k}{\rho c} \quad (5.4)$$

และกำหนดให้ค่าเรย์เลห์นัมเบอร์ (Rayleigh number;  $Ra$ ) มีค่าเท่ากับ  $10^3$ ,  $10^4$  และ  $10^5$  ค่าเรย์เลห์นัมเบอร์จะแสดงอัตราส่วนของแรงลอยตัวเนื่องจากอุณหภูมิกับแรงของความหนืดในของไหลดังนี้

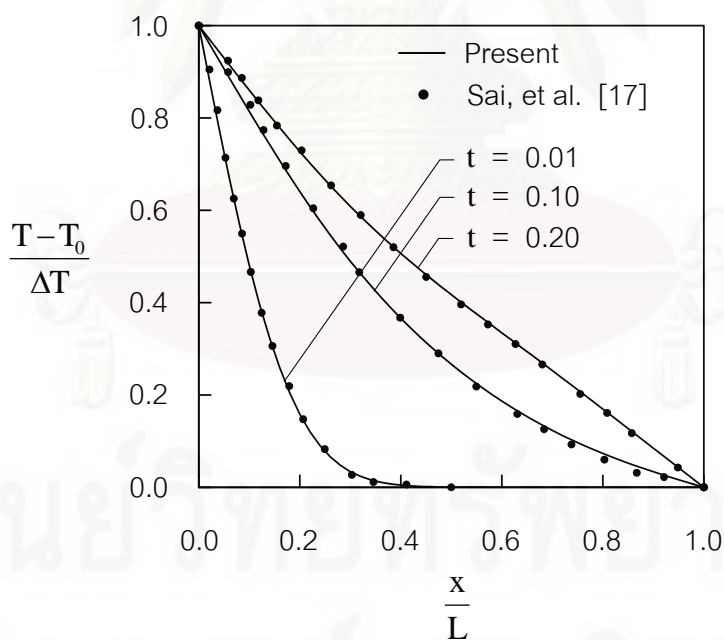
$$Ra = \frac{g\beta\Delta TL^3}{\nu\alpha} \quad (5.5)$$

โดยที่  $L$  แทนค่าความยาวเฉพาะ (characteristic length) ของปัญหา ในงานวิจัยนี้จะใช้ค่า  $L$  เท่ากับ 1 จากนั้นทำการเปรียบเทียบผลลัพธ์ที่คำนวณได้ในกรณีของค่า  $Ra$  เท่ากับ  $10^3$ ,  $10^4$  และ  $10^5$  ณ เวลาต่าง ๆ ดังแสดงในรูปที่ 5.9, 5.10 และ 5.11 ตามลำดับ โดยจะแสดงการเปรียบเทียบค่าความเร็ว  $v$  ตลอดแนวแกน  $x$  ที่ระดับกึ่งกลางความสูงของช่องสี่เหลี่ยมจัตุรัส และแสดงการเปรียบเทียบของอุณหภูมิ  $T$  ตลอดแนวเดียวกัน โดยทำการเปรียบเทียบับผลการคำนวณของ Sai, et al. [17] ผลลัพธ์ที่ได้เป็นที่น่าพอใจ



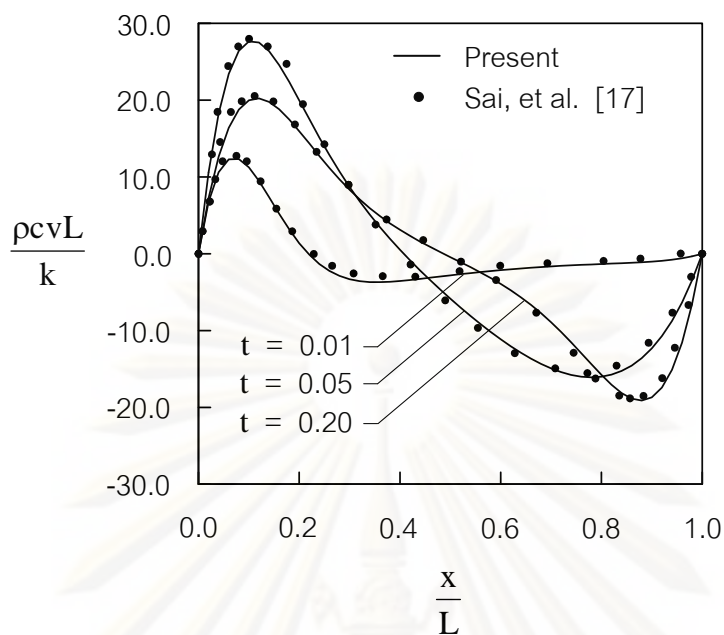


(ก) ความเร็วไร้มิติ

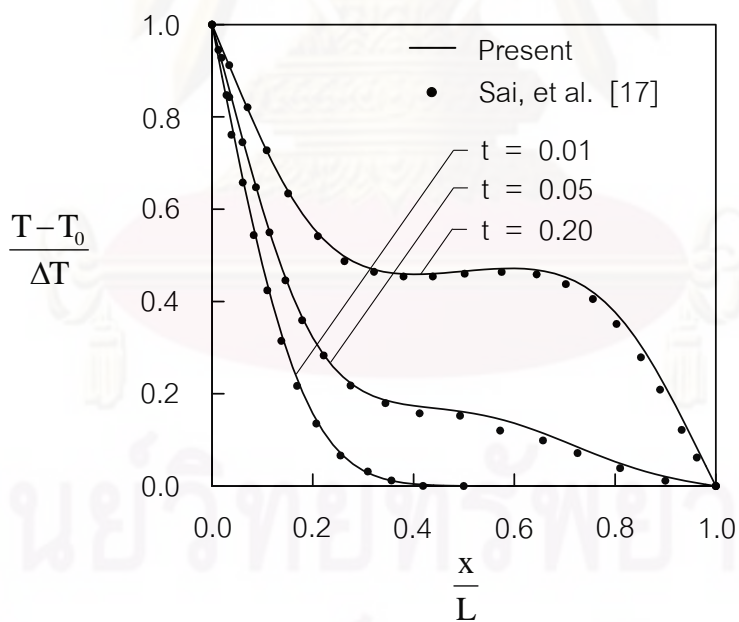


(ข) อุณหภูมิไร้มิติ

รูปที่ 5.9 การเปรียบเทียบลักษณะการกระจายของความเร็วจึงและอุณหภูมิไร้มิติตลอดแนวแกน  $x$  ที่ค่าเรย์เลห์นัมเบอร์เท่ากับ  $10^3$  ณ เวลาต่างๆ

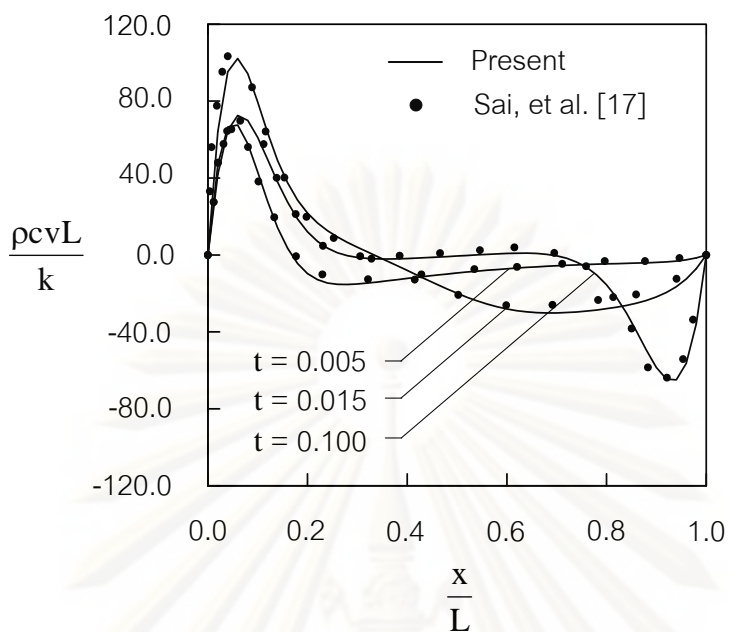


(ก) ความเร็วไร้มิติ

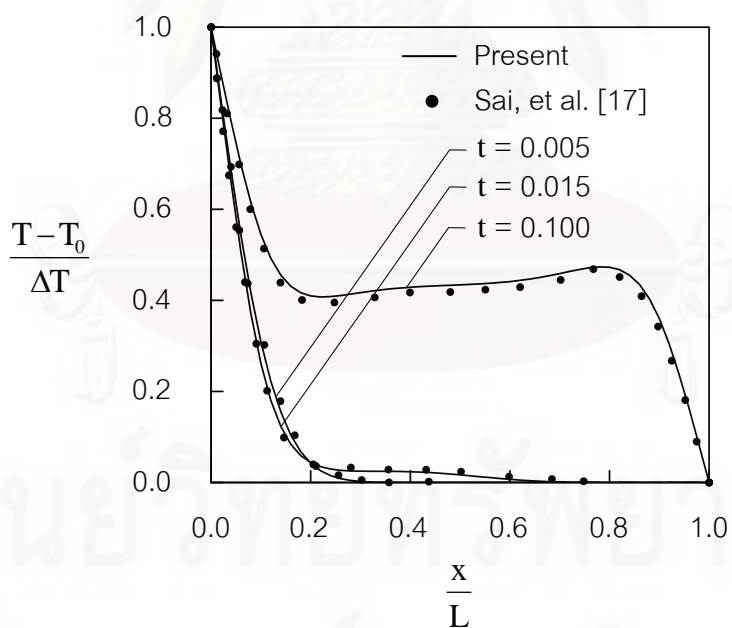


(ข) อุณหภูมิไร้มิติ

รูปที่ 5.10 การเปรียบเทียบลักษณะการกระจายของความเร็วจึงและอุณหภูมิไร้มิติตลอดแนวแกน  $x$  ที่ค่าเรย์เลห์นัมเบอร์เท่ากับ  $10^4$  ณ เวลาต่าง ๆ



(ก) ความเร็วไร้มิติ



(ข) อุณหภูมิไร้มิติ

รูปที่ 5.11 การเปรียบเทียบลักษณะการกระจายของความเร็วและอุณหภูมิไร้มิติ  
ตลอดแนวแกน  $x$  ที่ค่าเรย์เลห์นัมเบอร์เท่ากับ  $10^5$  ณ เวลาต่าง ๆ

จากตารางที่ 5.2 แสดงการเปรียบเทียบค่าตัวเลขที่นมเบอร์เฉลี่ยที่ผนังร้อน,  $\overline{Nu}_{x=0}$  โดยเปรียบเทียบกับผลการทดลอง [27] และผลจากการคำนวณต่าง ๆ [6, 17, 28] ซึ่งผลที่ได้ให้ค่าที่สอดคล้องกันเป็นอย่างดี

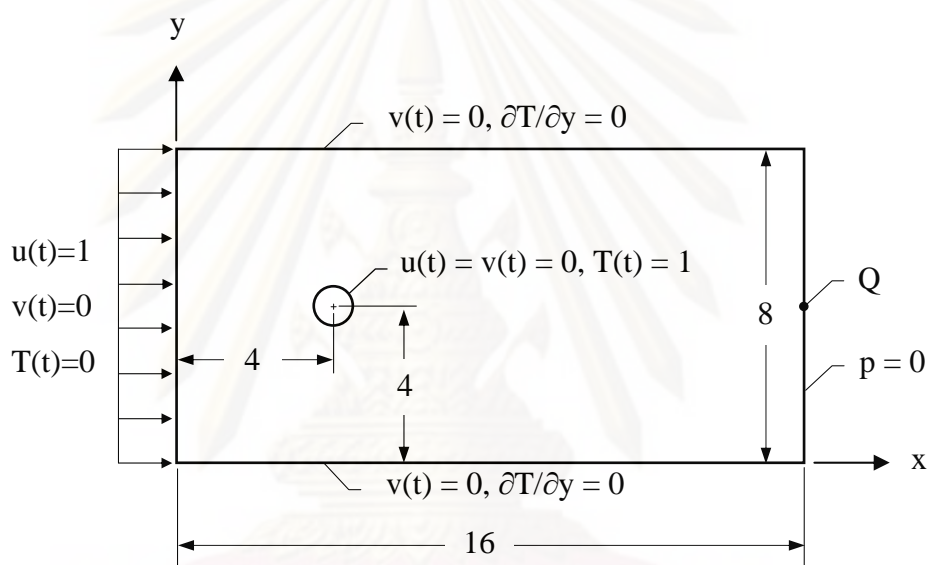
ตารางที่ 5.2 การเปรียบเทียบค่าตัวเลขที่นมเบอร์เฉลี่ยที่ผนังร้อนโดยเปรียบเทียบกับวิธีอื่น ๆ

ค่าตัวเลขที่นมเบอร์เฉลี่ยที่ผนังร้อน (เปอร์เซ็นต์ความคลาดเคลื่อนเปรียบเทียบกับ [27])			
Ra	$10^3$	$10^4$	$10^5$
de Vahl Davis [27] (Benchmark)	1.117	2.238	4.509
Choi, et al. [6]	1.143 (2.33%)	2.264 (1.16%)	4.530 (0.47%)
Sai, et al. [17]	1.131 (1.25%)	2.289 (2.28%)	4.687 (3.95%)
Leal, et al. [28]	1.118 (0.09%)	2.248 (0.44%)	4.562 (1.18%)
Present	1.117 (0.00%)	2.234 (0.18%)	4.466 (0.95%)

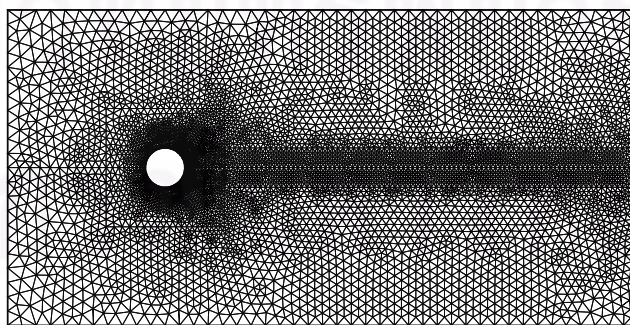
#### 5.1.4 ปัญหาการถ่ายเทความร้อนผ่านทรงกระบอกกลม (Transient flow over a heated circular cylinder)

ในปัญหานี้เป็นอีกหนึ่งปัญหาที่ใช้ในการตรวจสอบความถูกต้องสำหรับปัญหาการถ่ายเทความร้อนผ่านทรงกระบอกที่สภาวะไม่อยู่ตัว ลักษณะโดยทั่วไปของปัญหานี้เมื่อการไหลมีค่าเรย์โนลด์นัมเบอร์ไม่มากนักปัญหาดังกล่าวจะเป็นการไหลที่สภาวะอยู่ตัวและจะเกิดการไหลหมุนวนขึ้นบริเวณด้านหลังของวัตถุทรงกระบอก เมื่อเพิ่มค่าเรย์โนลด์นัมเบอร์จนถึงค่าหนึ่งขนาดของการไหลหมุนวนนั้นจะมีขนาดใหญ่ขึ้น แต่ถ้าเพิ่มค่าเรย์โนลด์นัมเบอร์มากขึ้นเรื่อย ๆ จนถึงค่าหนึ่งปัญหาดังกล่าวจะเป็นการไหลที่สภาวะไม่อยู่ตัวและจะเกิดการไหลบริเวณด้านหลังของทรงกระบอกเป็นการไหลหมุนวนแบบซ้ำไปซ้ำมา ส่วนในงานวิจัยนี้จะศึกษาการไหลที่สภาวะไม่อยู่ตัวที่ค่าเรย์โนลด์นัมเบอร์,  $Re$  เท่ากับ 100 และค่าแพรนด์เทิลนัมเบอร์,  $Pr$  เท่ากับ 0.71 โดยทำการเปรียบเทียบกับผลลัพธ์เชิงตัวเลขจากนักวิจัยท่านอื่น ๆ ลักษณะของปัญหานี้ประกอบไปด้วยช่องทางการไหลขนาด  $8 \times 16$  โดยขอบทางด้านซ้ายมีมีของไหลผ่านเข้ามาด้วยอุณหภูมิต่ำและค่อย ๆ ไหลผ่านวัตถุทรงกระบอกที่มีอุณหภูมิสูงจนเกิดการแลกเปลี่ยนความร้อนแล้วไหลออกทางขอบด้านขวามือ ส่วนขอบด้านบนและด้านล่างเป็นแบบสมมาตรดังแสดงในรูปที่ 5.12 รูป

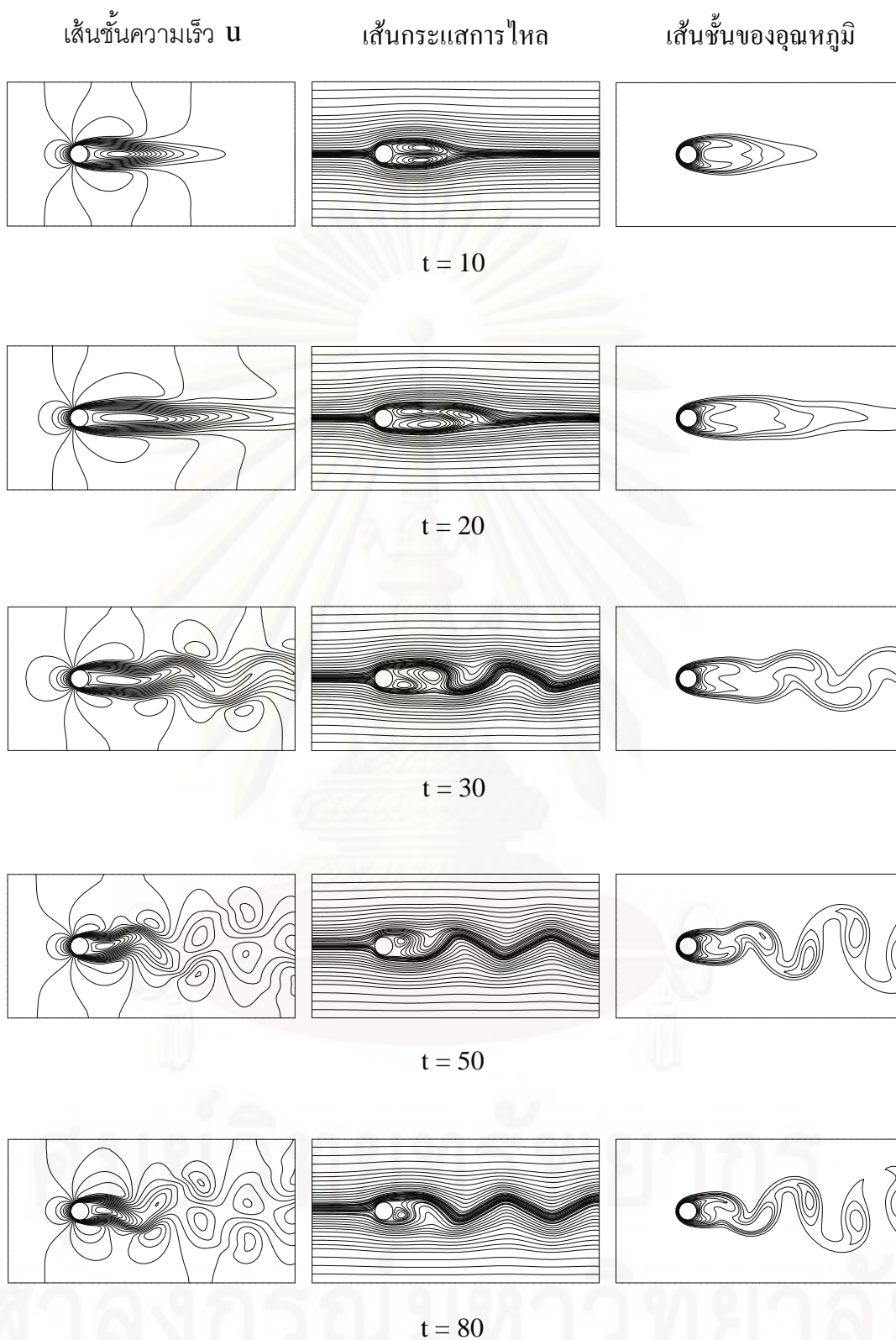
แบบจำลองไฟไนต์เอลิเมนต์ดังกล่าวประกอบไปด้วย 6,485 จุดต่อ และ 12,734 เอลิเมนต์ ดังแสดงในรูปที่ 5.13 จากนั้นทำการวิเคราะห์ปัญหาดังกล่าวที่ค่าเรย์โนลด์เท่ากับ 100 ผลลัพธ์ที่ได้จากการวิเคราะห์ปัญหาได้แสดงในรูปที่ 5.14 โดยรูปที่ 5.14 แสดงเส้นกระแสความเร็ว  $u$  เส้นกระแสการไหล และเส้นชั้นของอุณหภูมิเรียงจากด้านซ้ายไปขวาตามลำดับ ที่เวลาเท่ากับ 10, 20, 30, 50 และ 80 จากนั้นทำการเปรียบเทียบค่าความเร็ว  $v$  ที่ตำแหน่งกึ่งกลางของช่องทางการไหล ออกตรงตำแหน่งจุด  $Q$  ที่แสดงในรูปที่ 5.12 ขณะที่เวลาเปลี่ยนแปลงไปกับผลการคำนวณของ Malan, et al. [29] และ Sampaio, et al. [30] ดังแสดงในรูปที่ 5.15



รูปที่ 5.12 ลักษณะของปัญหาการถ่ายเทความร้อนผ่านทรงกระบอกกลม



รูปที่ 5.13 แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการถ่ายเทความร้อนผ่านทรงกระบอกกลม

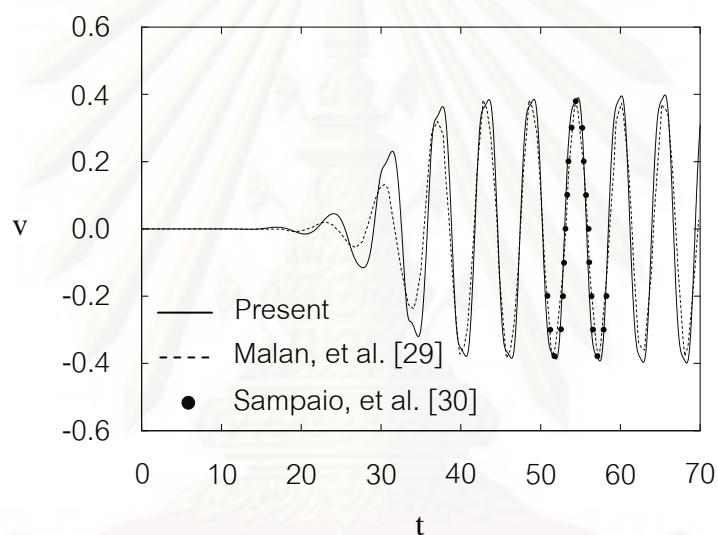


รูปที่ 5.14 แสดงเส้นความเร็ว  $u$  เส้นกระแสการไหล เส้นชั้นของอุณหภูมิ  
 ที่เวลาเท่ากับ 10, 20, 30, 50 และ 80

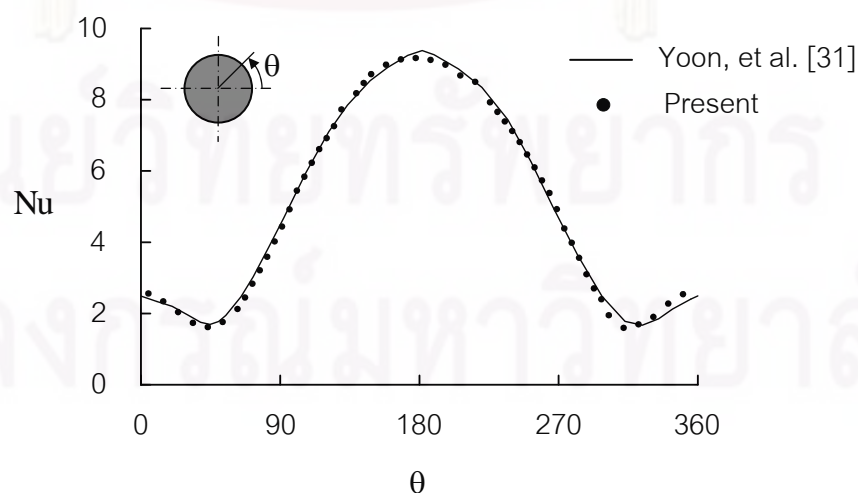
รูปที่ 5.16 แสดงค่าตัวเลขนัมเบอร์ที่ตำแหน่งต่าง ๆ บนผิวของทรงกระบอกร้อน โดยนำมาเปรียบเทียบกับผลการคำนวณของ Yoon, et al. [31] ผลลัพธ์ที่ได้สอดคล้องกันเป็นอย่างดี หลังจากนั้นจึงคำนวณหาค่าเฉลี่ยของตัวเลขนัมเบอร์  $\overline{Nu}$  ตลอดขอบผิวทรงกระบอกมีค่าเท่ากับ 5.058 แล้วนำมาเปรียบเทียบกับ Yoon, et al. แสดงค่าเฉลี่ยของตัวเลขนัมเบอร์ไว้ในรูปของสมการ (5.6-5.7) โดยเปอร์เซ็นต์ความคลาดเคลื่อนมีค่าเท่ากับ 1.36%

$$\overline{Nu} = 0.082Re^{0.5} + 0.734Re^\lambda \quad (5.6)$$

โดยที่  $\lambda = 0.05 + 0.226Re^{0.085} \quad (5.7)$



รูปที่ 5.15 การเปรียบเทียบค่าความเร็ว  $v$  ที่ตำแหน่ง  $Q$  ณ เวลาต่าง ๆ



รูปที่ 5.16 ค่าตัวเลขนัมเบอร์ที่ตำแหน่งต่าง ๆ บนผิวของทรงกระบอกร้อน

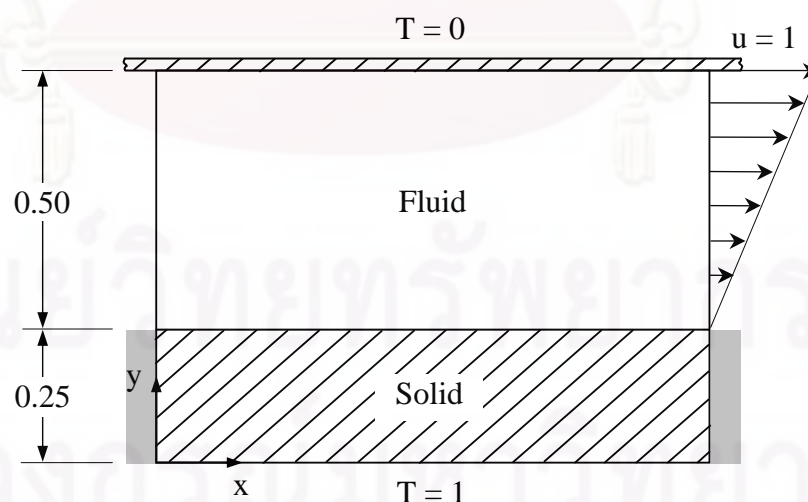
### 5.1.5 ปัญหาการไหลระหว่างแผ่นคู่ขนานพร้อมการถ่ายเทความร้อนกับโครงสร้าง (Conjugate Couette flow problem in parallel plate channel)

สำหรับปัญหาการไหลระหว่างแผ่นคู่ขนานที่มีการถ่ายเทความร้อนกับโครงสร้างนี้เป็นปัญหาพื้นฐานที่ใช้ในการตรวจสอบโปรแกรมคอมพิวเตอร์ที่ประดิษฐ์ขึ้นสำหรับปัญหาการไหลโดยมีการถ่ายเทความร้อนกับโครงสร้าง ได้แสดงในรูปที่ 5.17 โดยที่แผ่นด้านบนเป็นแผ่นบางที่มีอุณหภูมิต่ำ ( $T = 0$ ) แล้วมีการเคลื่อนที่ของความเร็วในแนวแกน  $x$  เท่ากับ 1 ในขณะที่แผ่นด้านล่างเป็นผนังที่มีความหนาเท่ากับ 0.25 และไม่มีการเคลื่อนที่ โดยที่ขอบด้านล่างของผนัง ( $y = 0$ ) มีอุณหภูมิสูง ( $T = 1$ ) และกำหนดให้ระยะห่างระหว่างแผ่นคู่ขนานเท่ากับ 0.5 โดยลักษณะของปัญหานี้สามารถหาผลเฉลยแน่นอนตรงได้ [24] โดยผลเฉลยแน่นอนตรงของค่าความเร็วและอุณหภูมิเป็นดังนี้

$$u(y) = 2y - 0.5 \quad (5.8)$$

$$T(y) = \frac{2K_{sf}}{2K_{sf} + 1} (1.5 - y) \quad (5.9)$$

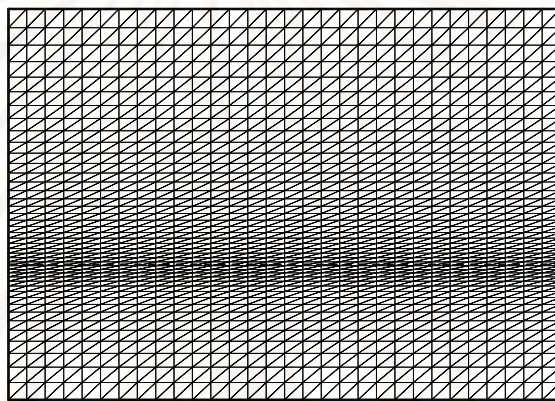
โดยที่  $K_{sf} = \frac{k_s}{k_f} = \frac{\text{สัมประสิทธิ์การนำความร้อนในโครงสร้าง}}{\text{สัมประสิทธิ์การนำความร้อนในของไหล}}$



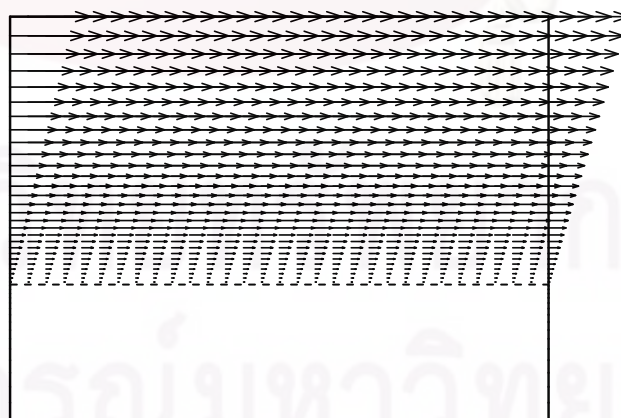
รูปที่ 5.17 ลักษณะของปัญหาการไหลระหว่างแผ่นคู่ขนานพร้อมการถ่ายเทความร้อนกับโครงสร้าง



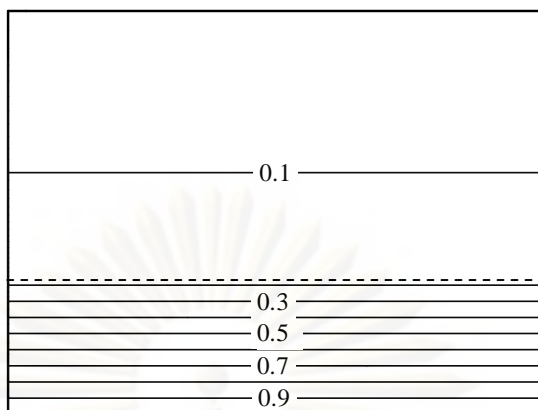
ปัญหาดังกล่าวถูกนำไปวิเคราะห์โดยใช้โปรแกรมคอมพิวเตอร์ที่ประดิษฐ์ขึ้นโดยเริ่มจากการสร้างรูปแบบจำลองไฟไนต์เอลิเมนต์ดังแสดงในรูปที่ 5.18 ซึ่งประกอบไปด้วย 1,364 จุดต่อและ 2,580 เอลิเมนต์ โดยกำหนดให้ความดันมีค่าเท่ากับศูนย์ตลอดขอบในแนวตั้งทางด้านซ้ายและขวาของโดเมนการคำนวณ โดยผลลัพธ์ที่ได้จากการคำนวณจะมีลักษณะการกระจายตัวของเวกเตอร์ความเร็วดังแสดงในรูปที่ 5.19 และการกระจายตัวของอุณหภูมิที่ค่า  $K_{sf}$  เท่ากับ 0.1, 1 และ 10 ตามลำดับ ดังแสดงในรูปที่ 5.20 จากรูปจะเห็นได้ว่าเมื่ออัตราส่วนของสัมประสิทธิ์การถ่ายเทความร้อนมีค่าเพิ่มขึ้นการถ่ายเทความร้อนจะดียิ่งขึ้น



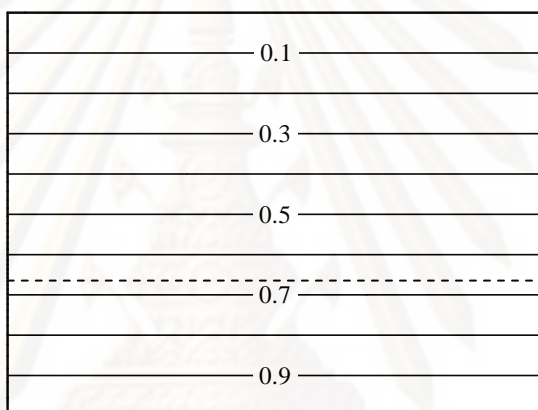
รูปที่ 5.18 แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการไหลระหว่างแผ่นคู่ขนาน พร้อมการถ่ายเทความร้อนกับโครงสร้าง



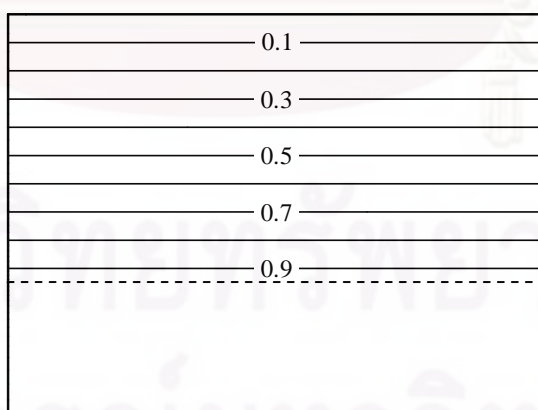
รูปที่ 5.19 เวกเตอร์ความเร็วของปัญหาการไหลระหว่างแผ่นคู่ขนาน ที่มีการถ่ายเทความร้อนกับโครงสร้าง



(ก) เส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 0.1



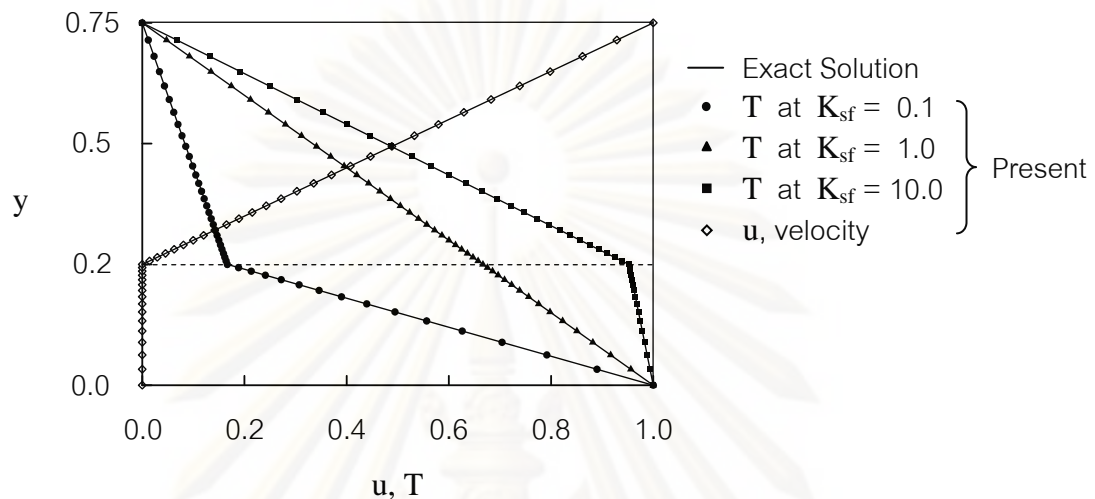
(ข) เส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 1



(ค) เส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 10

รูปที่ 5.20 ลักษณะการกระจายของอุณหภูมิของปัญหาการไหลระหว่างแผ่นคู่ขนาน พร้อมการถ่ายเทความร้อนกับโครงสร้างที่ค่า  $K_{sf}$  ต่าง ๆ

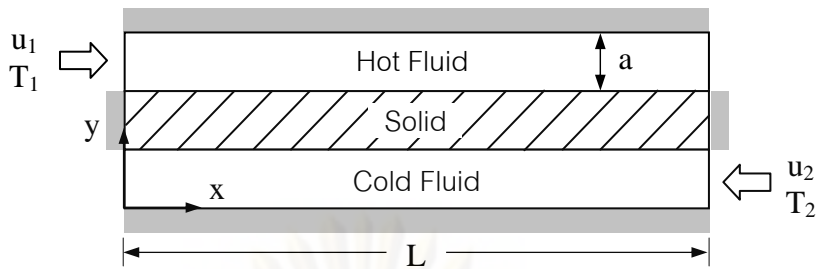
ในรูปที่ 5.21 เป็นการเปรียบเทียบผลลัพธ์ที่ได้จากการคำนวณกับผลเฉลยแม่นยำตรงที่ค่า  $K_{sf}$  เท่ากับ 0.1, 1 และ 10 โดยผลลัพธ์ที่ได้มีค่าความผิดพลาดของความเร็วและค่าของอุณหภูมิมีค่าเท่ากับ 0.01% และ 0.04% ตามลำดับ ซึ่งให้ผลเป็นที่น่าพอใจ



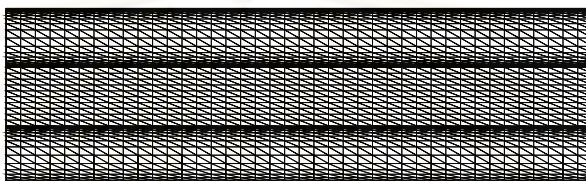
รูปที่ 5.21 การเปรียบเทียบการกระจายตัวของความเร็วและอุณหภูมิที่ได้จากการคำนวณกับผลเฉลยแม่นยำตรงที่ค่า  $K_{sf}$  ต่าง ๆ

### 5.1.6 ปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน (Conjugate counter flow heat exchanger)

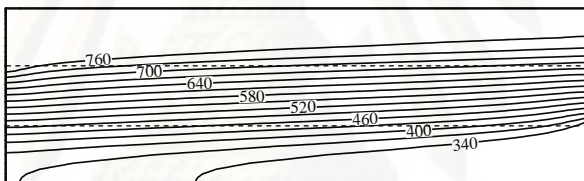
สำหรับปัญหานี้เป็นปัญหาสำหรับการตรวจสอบความถูกต้องของโปรแกรมคอมพิวเตอร์ที่พัฒนาขึ้น โดยจะนำผลลัพธ์ที่ได้จากการคำนวณมาเปรียบเทียบกับผลการคำนวณด้วยระเบียบวิธีเชิงตัวเลขของ Chen and Han [32] โดยมีลักษณะของปัญหาเป็นอุปกรณ์แลกเปลี่ยนความร้อนที่มีการไหลสวนทางกันและมีแผ่นเหล็กเป็นตัวกลางในการแลกเปลี่ยนความร้อนโดยมีปลายทั้งสองเป็นฉนวน สำหรับช่องทางไหลทั้งสองมีทิศทางการไหลสวนทางกัน มีขอบด้านบนและด้านล่างเป็นฉนวน ดังแสดงในรูปที่ 5.22 ช่องทางการไหลทั้งสองมีขนาดเท่ากับความหนาของแผ่นเหล็ก  $a$  เท่ากับ 0.1 และ  $L$  มีค่าเท่ากับ 1.0 โดยที่มีช่องทางการไหลข้างบนเป็นของไหลที่มีอุณหภูมิสูงไหลเข้ามาด้วยความเร็ว  $u_1 = 0.2$  อุณหภูมิ  $T_1 = 800$  และค่าเรย์โนลด์  $Re = 133$  ส่วนช่องทางการไหลข้างล่างเป็นของไหลที่มีอุณหภูมิต่ำไหลเข้ามาด้วยความเร็ว  $u_2 = 0.1$  อุณหภูมิ  $T_2 = 300$  และค่าเรย์โนลด์  $Re = 66$  ปัญหานี้จะคิดค่า ฟรันด์เทิลนัมเบอร์  $Pr = 0.75$  และอัตราส่วนของการแพร่กระจายความร้อนระหว่างของแข็งและของไหล  $\alpha_{sf} = K_{sf}$  จากนั้นทำการสร้างรูปแบบจำลองทางไฟไนต์เอลิเมนต์ซึ่งประกอบไปด้วย 1,763 จุดต่อ และ 3,360 เอลิเมนต์ ดังแสดงในรูปที่ 5.23



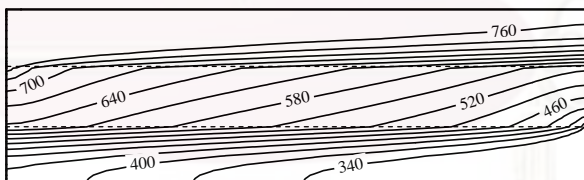
รูปที่ 5.22 ลักษณะของปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน



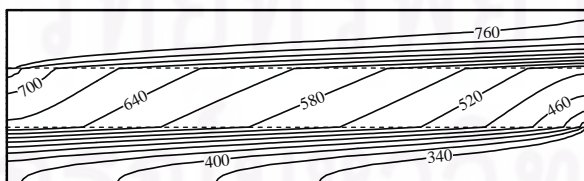
รูปที่ 5.23 แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน



(ก) เส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 1



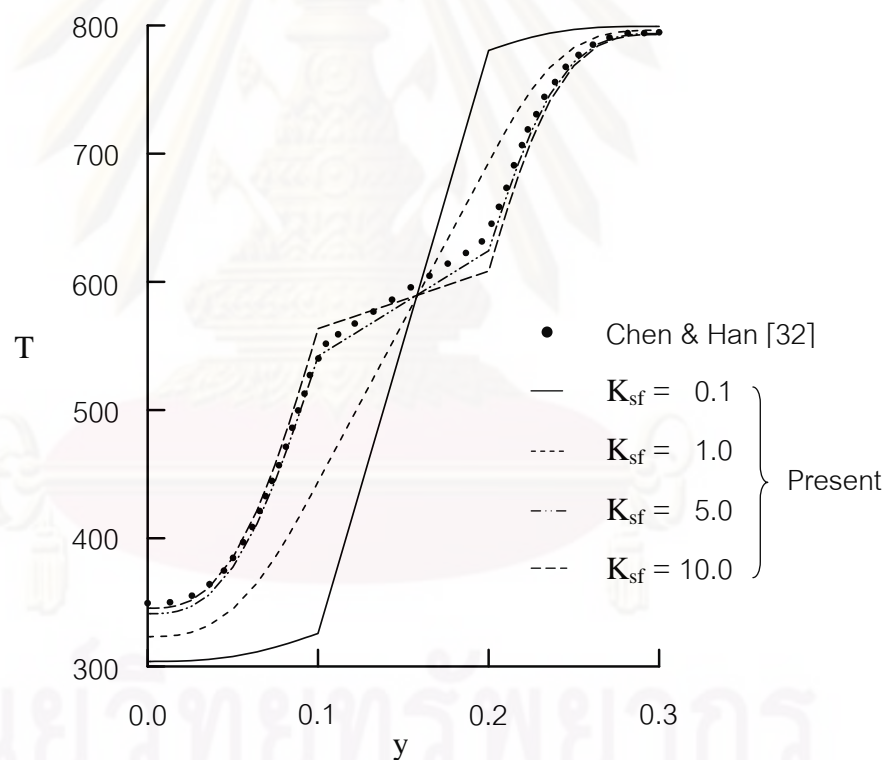
(ข) เส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 5



(ค) เส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 10

รูปที่ 5.24 ลักษณะการกระจายตัวของอุณหภูมิของปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อนที่มีการไหลสวนทางกันที่ค่า  $K_{sf}$  ต่าง ๆ

รูปที่ 5.25 แสดงการเปรียบเทียบการกระจายตัวของอุณหภูมิกับผลการคำนวณของ Chen and Han ในปัญหานี้จะพบว่า การถ่ายเทความร้อนภายในบริเวณของแข็งเกิดการเปลี่ยนแปลงอุณหภูมิที่มีลักษณะเป็นเส้นตรง เนื่องจากมีการถ่ายเทความร้อนแบบการแพร่กระจาย ส่วนในบริเวณของไหลการเปลี่ยนแปลงอุณหภูมิมักมีลักษณะเป็นเส้นโค้งเนื่องจากการถ่ายเทความร้อนแบบการพาความร้อน จากผลการคำนวณเมื่อค่าอัตราส่วนของสัมประสิทธิ์การถ่ายเทความร้อน  $K_{sf}$  เพิ่มขึ้นการแลกเปลี่ยนความร้อนระหว่างของเหลวจะดีขึ้นโดยสามารถอธิบายได้ในรูปที่ 5.25 ซึ่งแสดงการกระจายของอุณหภูมิที่ตำแหน่ง กึ่งกลางของปัญหา ( $x = 0.5$ ) ตลอดแนวแกน  $y$  ที่ค่า  $K_{sf}$  เท่ากับ 0.1, 1, 5 และ 10 โดยทำการเปรียบเทียบกับผลการคำนวณของ Chen and Han [32] ที่ค่า  $K_{sf} = 5$  พบว่าผลที่ได้มีความสอดคล้องกัน



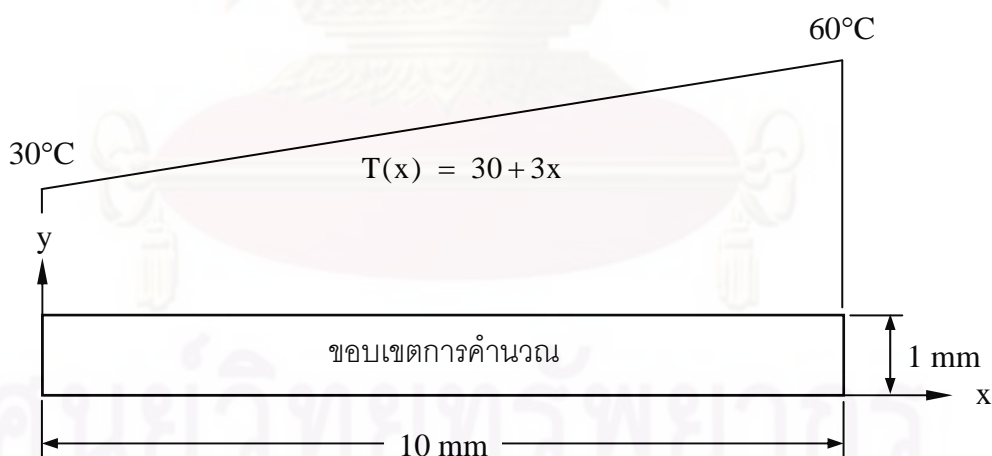
รูปที่ 5.25 การเปรียบเทียบค่าอุณหภูมิที่ได้จากการคำนวณกับผลลัพธ์ของ Chen and Han ที่ตำแหน่ง  $x = 0.5$  ตลอดแกน  $y$  ของปัญหาการถ่ายเทความร้อนแบบคอนจูเกต ในอุปกรณ์แลกเปลี่ยนความร้อนที่มีการไหลสวนทางกันที่ค่า  $K_{sf}$  ต่าง ๆ

## 5.2 ตรวจสอบโปรแกรมสำหรับวิเคราะห์ปัญหาความเค้นอันเนื่องมาจากอุณหภูมิ

ในหัวข้อต่อไปนี้จะทำการตรวจสอบความถูกต้องของโปรแกรมวิเคราะห์ปัญหาความเค้นอันเนื่องมาจากอุณหภูมิ โดยทั้งสองปัญหาที่จะนำมาตรวจสอบความถูกต้องนั้นจะทำการแทนค่าอุณหภูมิที่ถูกต้องลงไปยังโดเมนของปัญหาด้วย เนื่องจากโปรแกรมที่ได้ประดิษฐ์ขึ้นนี้ใช้สำหรับนำค่าของอุณหภูมิและแรงที่กระทำจากภายนอกเพื่อมาคำนวณหาค่าการเคลื่อนตัวเท่านั้น จากนั้นจึงนำค่าการเคลื่อนตัวที่ได้มาคำนวณหาค่าความเค้นเฉือนและค่าความเค้นในแนวแกน  $x$  และ  $y$  โดยจะทำการทดสอบกับปัญหาที่มีผลเฉลยแม่นยำตรง 2 ปัญหาดังนี้

### 5.2.1 ปัญหาแผ่นสี่เหลี่ยมที่มีอุณหภูมิเปลี่ยนแปลงเชิงเส้นตามความยาว

ลักษณะของปัญหาเป็นแผ่นสี่เหลี่ยมขนาด  $10 \times 1$  mm มีอุณหภูมิที่ขึ้นกับระยะทาง  $x$  ด้วยความสัมพันธ์เชิงเส้น  $T(x) = 30 + 3x$  โดยตลอดขอบทางซ้ายจะมีอุณหภูมิเท่ากับ  $30^\circ\text{C}$  ส่วนตลอดขอบทางด้านขวามีอุณหภูมิเท่ากับ  $60^\circ\text{C}$  ดังแสดงในรูปที่ 5.26 ส่วนขอบทางด้านซ้ายจะไม่มี การเคลื่อนที่ทางแกน  $x$  และที่จุด  $(0,0)$  กำหนดให้ถูกต้องกับผนัง ส่วนคุณสมบัติทางกลของแผ่นสี่เหลี่ยมนี้คือ ค่าโมดูลัสของความยืดหยุ่น  $E = 71,705.5 \text{ N/mm}^2$  ค่าอัตราส่วนปัวส์ซง  $\nu = 0.29$  และสัมประสิทธิ์การขยายตัว  $\alpha = 1.27 \times 10^{-5} \text{ mm/mm}^\circ\text{C}$



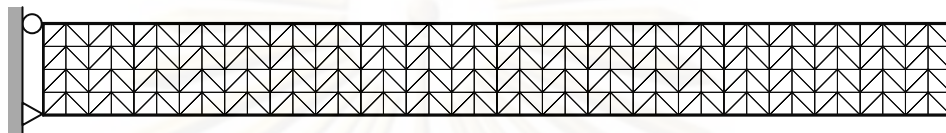
รูปที่ 5.26 ลักษณะของปัญหาแผ่นสี่เหลี่ยมที่มีอุณหภูมิเปลี่ยนแปลงเชิงเส้นตามความยาว

ปัญหานี้มีผลเฉลยแม่นยำตรงดังนี้ [33]

$$u(x) = \alpha \left( 30x + \frac{3x^2}{2} \right) \quad (5.10)$$

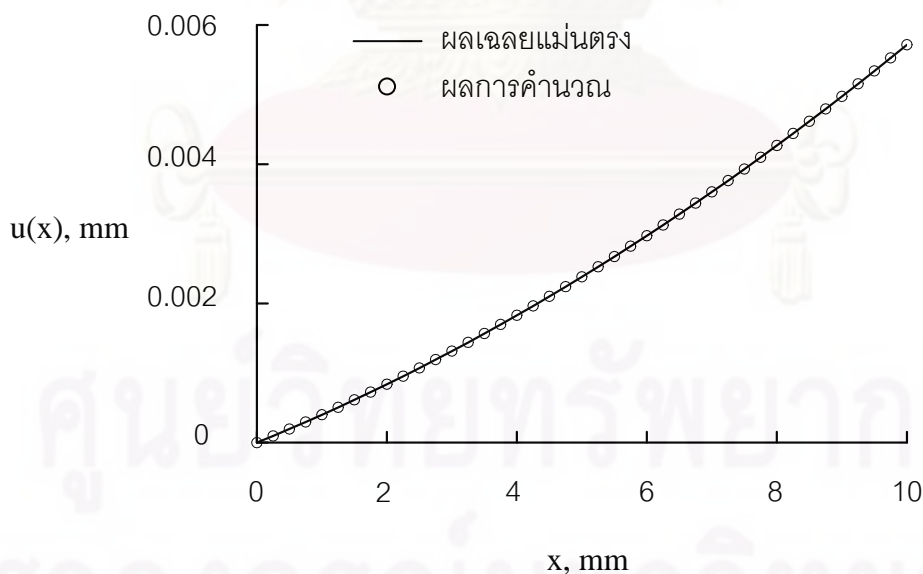
โดย  $u(x)$  คือค่าการเคลื่อนตัวในแนวแกน  $x$

การคำนวณค่าการเคลื่อนตัวบนแผ่นสี่เหลี่ยมดังที่แสดงข้างต้นโดยใช้โปรแกรมที่ได้ประดิษฐ์ขึ้นนี้ จะเริ่มต้นจากการแบ่งขอบเขตของปัญหาออกเป็นเอลิเมนต์ย่อย ๆ จำนวน 205 จุด ต่อ 320 เอลิเมนต์ ดังแสดงในรูปที่ 5.27 จากนั้นทำการประยุกต์เงื่อนไขขอบเขตตามที่กล่าวข้างต้น และกำหนดอุณหภูมิขณะที่ยึดไม่มีความเค้น  $T_0$  เท่ากับ  $0^{\circ}\text{C}$  แล้วทำการคำนวณหาค่าการเคลื่อนตัวที่เกิดขึ้น



รูปที่ 5.27 รูปแบบไฟไนต์เอลิเมนต์สำหรับปัญหาแผ่นสี่เหลี่ยมที่มีอุณหภูมิเปลี่ยนแปลงเชิงเส้นตามความยาว

รูปที่ 5.28 เป็นการเปรียบเทียบระยะการเคลื่อนตัวในแนวแกน  $x$  ที่คำนวณได้จากโปรแกรมกับผลเฉลยแม่นยำ พบว่ามีค่าความผิดพลาดเท่ากับ 0.123%



รูปที่ 5.28 ค่าการเคลื่อนตัวเนื่องจากความร้อนในแนวแกน  $x$  สำหรับปัญหาแผ่นสี่เหลี่ยมที่มีอุณหภูมิเปลี่ยนแปลงเชิงเส้นตามความยาว

## 5.2.2 ปัญหาวงแหวนที่มีอุณหภูมิเปลี่ยนแปลงตามแนวรัศมี

ลักษณะของปัญหาเป็นวงแหวนที่มีรัศมีภายใน  $a$  เท่ากับ 100 mm รัศมีภายนอก  $b$  เท่ากับ 200 mm และวงแหวนมีอุณหภูมิเป็นความสัมพันธ์เชิงเส้นตรงกับรัศมี  $r$  ดังนี้

$$T(r) = 110 - 0.4r \quad (5.11)$$

โดยอุณหภูมิตลอดขอบด้านในเท่ากับ  $30^{\circ}\text{C}$  และขอบด้านนอกเท่ากับ  $70^{\circ}\text{C}$  ดังแสดงในรูปที่ 5.29 โดยที่ค่าสัมประสิทธิ์ทางกลของวงแหวนคือ ค่าโมดูลัสของความยืดหยุ่น  $E = 71,705.5 \text{ N/mm}^2$  ค่าอัตราส่วนปัวส์ซง  $\nu = 0.29$  สัมประสิทธิ์การขยายตัว  $\alpha = 1.27 \times 10^{-5} \text{ mm/mm-}^{\circ}\text{C}$  และมีอุณหภูมิที่วัสดุไม่มีความเค้น  $T_0$  เท่ากับ  $0^{\circ}\text{C}$  ปัญหานี้สามารถหาผลเฉลยแม่นยำตรงสำหรับกรณีความเค้นในระนาบได้ดังนี้ [33]

$$u(r) = \frac{\alpha}{r} \left( 110r^2 - \frac{0.4}{3} \left( (1+\nu)(r^3 - a^3) + \frac{b^3 - a^3}{b^2 - a^2} ((1-\nu)r^2 + (1+\nu)a^2) \right) \right) \quad (5.12)$$

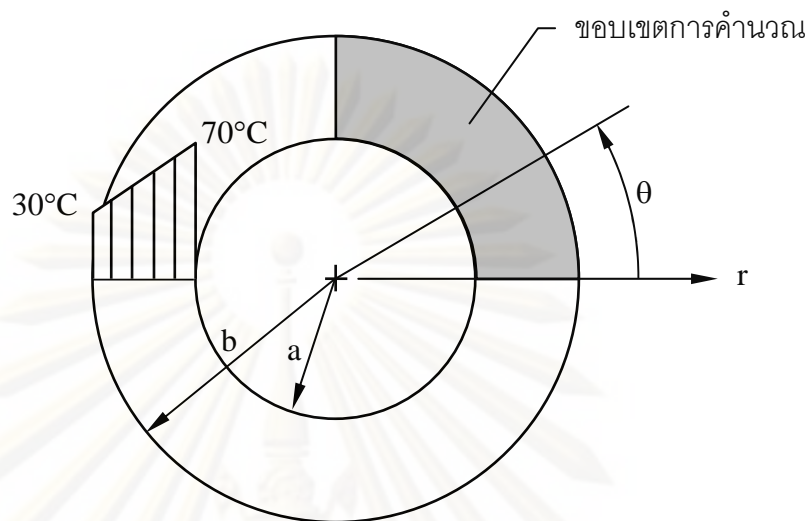
$$\sigma_r(r) = \frac{0.4 E \alpha}{3 r^2} \left( r^3 - a^3 - \frac{b^3 - a^3}{b^2 - a^2} (r^2 - a^2) \right) \quad (5.13)$$

$$\sigma_\theta(r) = \frac{0.4 E \alpha}{3 r^2} \left( 2r^3 + a^3 - \frac{b^3 - a^3}{b^2 - a^2} (r^2 + a^2) \right) \quad (5.14)$$

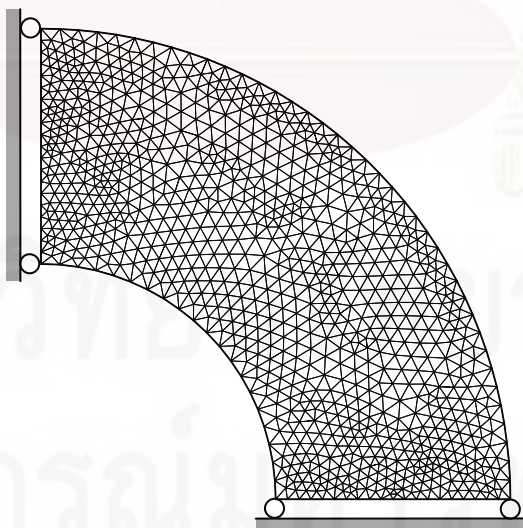
โดย  $u(r)$  คือค่าการเคลื่อนตัวในแนวรัศมี

เนื่องจากลักษณะของปัญหามีความสมมาตร ดังนั้นสามารถใช้ขอบเขตในการคำนวณเพียง 1 ใน 4 ของวงแหวนได้ ดังแสดงในรูปที่ 5.29 การวิเคราะห์ปัญหาจะเริ่มจากการแบ่งขอบเขตของปัญหาออกเป็นเอลิเมนต์ย่อย ๆ ซึ่งประกอบไปด้วย 966 จุดต่อ 1,831 เอลิเมนต์ ดังแสดงในรูปที่ 5.30 จากนั้นประยุกต์เงื่อนไขขอบเขตโดยกำหนดให้ตลอดขอบด้านล่างไม่มีการเคลื่อนตัวในแนวแกน  $y$  และตลอดขอบทางด้านซ้ายกำหนดให้ไม่มีการเคลื่อนตัวในแกน  $x$  รวมทั้งกำหนดอุณหภูมิ  $T(r)$  ตลอดทั้งขอบเขตการคำนวณ แล้วทำการคำนวณหาค่าการเคลื่อนตัวที่เกิดขึ้น ค่าการเคลื่อนตัวในแนวรัศมี ความเค้นในแนวรัศมี และในแนว  $\theta$  ที่คำนวณได้จากโปรแกรมที่ประดิษฐ์ขึ้น ได้นำมาแสดงเปรียบเทียบกับผลเฉลยแม่นยำตรงจากสมการ (5.12)–(5.14) ดังแสดงในรูปที่ 5.31–5.33 ซึ่งมีค่าความผิดพลาดเท่ากับ 0.026%, 1.96% และ 2.93% ตามลำดับ

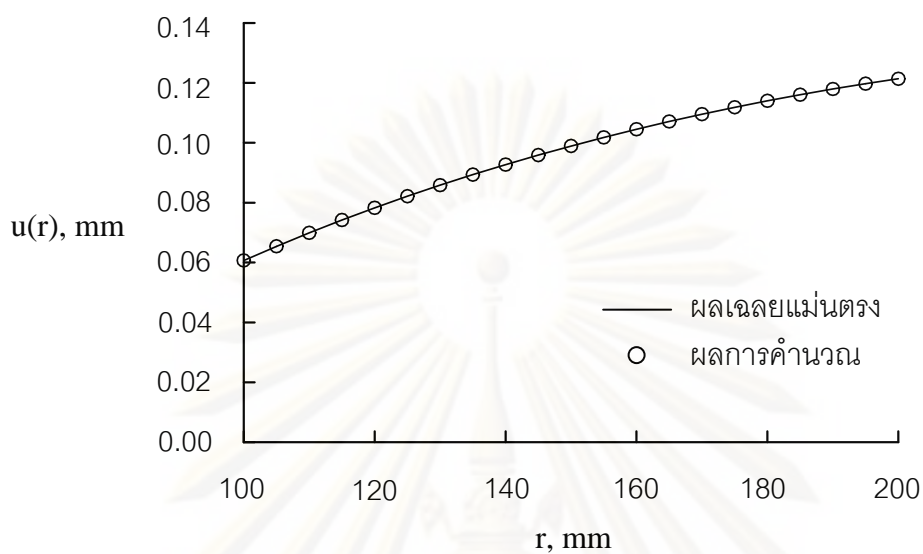




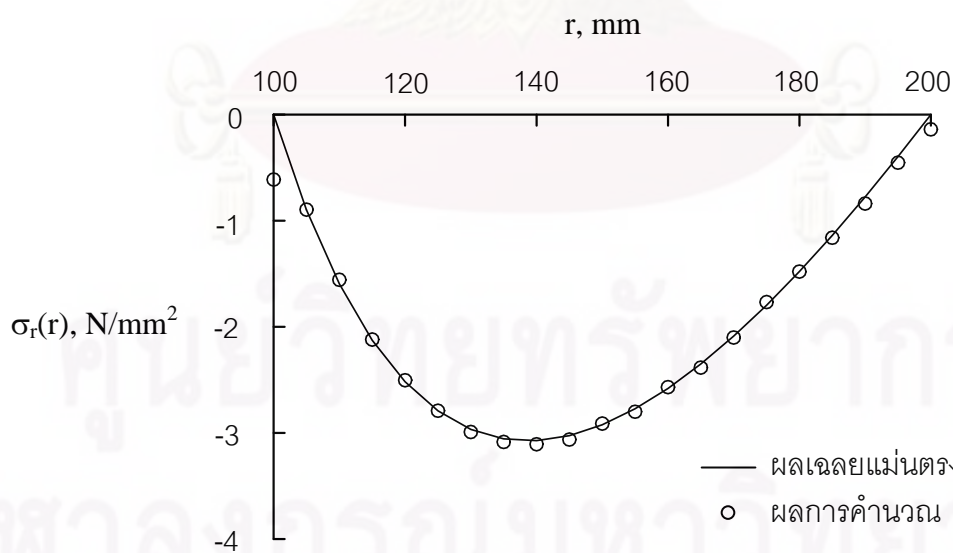
รูปที่ 5.29 ลักษณะของปัญหาวงแหวนที่มีอุณหภูมิเปลี่ยนแปลงตามแนวรัศมี



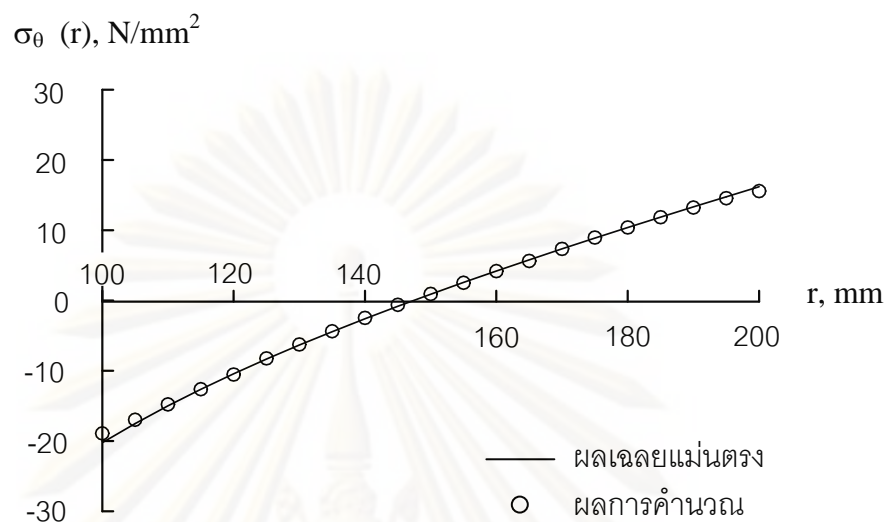
รูปที่ 5.30 รูปแบบไฟไนต์เอลิเมนต์สำหรับปัญหาวงแหวนที่มีอุณหภูมิเปลี่ยนแปลงตามแนวรัศมี



รูปที่ 5.31 การเปลี่ยนแปลงของค่าการเคลื่อนตัวตามทิศทางรัศมี ในแนวมุม  $\theta$  เท่ากับ 0



รูปที่ 5.32 ความเค้นในแนวรัศมี  $\sigma_r$  ตามทิศทางรัศมีที่มีแนวมุม  $\theta$  เท่ากับ 0



รูปที่ 5.33 ความเค้นในทิศ  $\theta$ ,  $\sigma_\theta$  ตามทิศทางรัศมีที่มีแนวมุม  $\theta$  เท่ากับ 0

## บทที่ 6

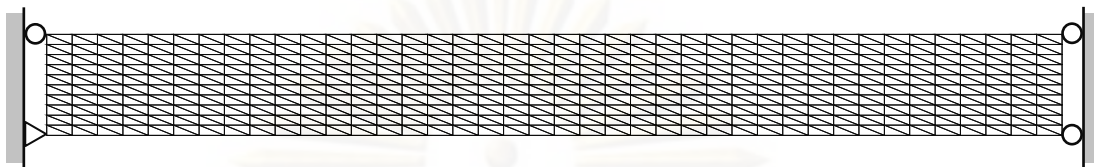
### การวิเคราะห์ปัญหาพร้อมกันระหว่างการไหล อุณหภูมิ และโครงสร้าง

ผลกระทบที่เกิดร่วมกันระหว่างการไหล อุณหภูมิ และโครงสร้าง เป็นพฤติกรรมที่พบได้ในอุปกรณ์แลกเปลี่ยนความร้อน (heat exchanger) หรือการทำความเย็นบนแผงวงจรรีเลย์ทรอนิกส์ (electronics board cooling) เป็นต้น ความเข้าใจในพฤติกรรมเหล่านี้จะช่วยให้การออกแบบมีประสิทธิภาพมากขึ้น ในหัวข้อนี้จะได้แสดงถึงขั้นตอนที่ใช้ในการวิเคราะห์ปัญหาพร้อมกันระหว่างการไหล อุณหภูมิ และโครงสร้าง โดยจะเริ่มจากการคำนวณหาพฤติกรรมของการไหลและการถ่ายเทความร้อนไปพร้อม ๆ กัน หลังจากนั้นนำค่าการกระจายตัวของอุณหภูมิที่คำนวณได้จากขั้นตอนก่อนนี้กับแรงอันเนื่องมาจากความดันของของไหลที่กระทำกับโครงสร้างมาคำนวณหาค่าความเค้นภายในโครงสร้างเป็นลำดับสุดท้าย ซึ่งในส่วนถัดไปจะแสดงถึงตัวอย่างที่ใช้ในการวิเคราะห์ทั้งหมด 4 ตัวอย่างด้วยกัน โดยตัวอย่างแรกเป็นปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน (conjugate counter flow heat exchanger) โดยปัญหานี้ได้วิเคราะห์หาค่าการกระจายตัวของอุณหภูมิและความดันที่จุดต่อต่าง ๆ แล้วในบทที่ 5 ปัญหาที่สองคือปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสโดยที่มีผนังนำความร้อน (conjugate natural convection in a square cavity with a conducting wall) ปัญหาที่สามคือปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายในช่องทางการไหล (forced convection cooling across rectangular blocks) และปัญหาสุดท้ายเป็นปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีครีระบายความร้อน (free convection in channel with fins)

#### 6.1 ปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน (conjugate counter flow heat exchanger)

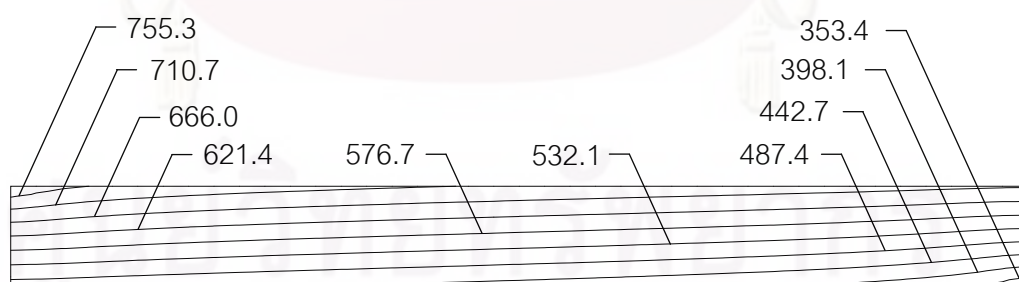
ลักษณะของปัญหาดังกล่าวได้แสดงไว้แล้วในรูปที่ 5.22 และได้ทำการวิเคราะห์หาพฤติกรรมของการไหลและการถ่ายเทความร้อนในกรณีที่มีค่าอัตราส่วนสัมประสิทธิ์การนำความร้อนระหว่างของแข็งและของไหลมีค่าต่าง ๆ กันเป็นที่เรียบร้อยแล้วอีกเช่นเดียวกัน ดังแสดงในรูปที่ 5.24(ก-ค) ในหัวข้อนี้จะได้นำเอาค่าการกระจายตัวของอุณหภูมิและความดันที่ช่องไหลกระทำกับของแข็งมาทำการวิเคราะห์หาค่าความเค้นที่เกิดขึ้น โดยกำหนดให้ของแข็งที่จะทำการวิเคราะห์มีค่าคุณสมบัติดังต่อไปนี้ มีค่าโมดูลัสของความยืดหยุ่น  $7 \times 10^{10}$  N/m<sup>2</sup> ค่าสัมประสิทธิ์การขยายตัว

$6 \times 10^{-6}/^{\circ}\text{C}$  ค่าอัตราส่วนปัวส์ซงเท่ากับ 0.3 และอุณหภูมิขณะที่ไม่มีความเค้นคือที่ 300 K รูปที่ 6.1 แสดงรูปแบบไฟไนต์เอลิเมนต์และเงื่อนไขขอบเขตของการวิเคราะห์ที่ปัญหาความเค้นอันเนื่องมาจากอุณหภูมิซึ่งประกอบไปด้วย 451 จุดต่อ 800 เอลิเมนต์



รูปที่ 6.1 รูปแบบไฟไนต์เอลิเมนต์และเงื่อนไขขอบเขตสำหรับการวิเคราะห์ความเค้นในปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน

เมื่อกำหนดอุณหภูมิภายในจุดต่อและค่าความดันที่กระทำกับผิวของของแข็งที่คำนวณได้จากบทที่ 5 แล้ว ต่อไปก็ทำการคำนวณหาค่าความเค้น โดยในรูปที่ 6.2 (ก-ค) แสดงลักษณะการกระจายตัวของอุณหภูมิ (K) ค่าความเค้นวอนมิสเชส (Von Mises stress, MPa) และค่าการเคลื่อนตัวพร้อมกับการเสียรูปขนาด 100 เท่าที่เกิดขึ้นภายในของแข็ง ในกรณีที่มีค่าอัตราส่วนของสัมประสิทธิ์การถ่ายเทความร้อนระหว่างของแข็งและของไหล,  $K_{sf} = 1$  ตามลำดับสำหรับลักษณะการกระจายตัวของอุณหภูมิ ค่าความเค้นวอนมิสเชสและค่าการเคลื่อนตัวพร้อมกับการเสียรูปที่เกิดขึ้นภายในของแข็ง ในกรณีที่มีค่า  $K_{sf} = 5$  และ 10 ได้แสดงในรูปที่ 6.3 และ 6.4 ตามลำดับ

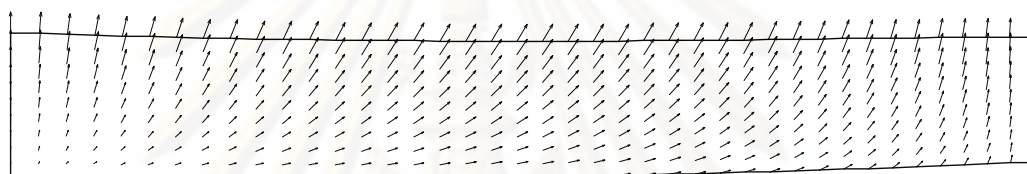


(ก) การกระจายตัวของอุณหภูมิ (K)

รูปที่ 6.2 ผลการคำนวณสำหรับปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อนในกรณีที่มีค่า  $K_{sf} = 1$

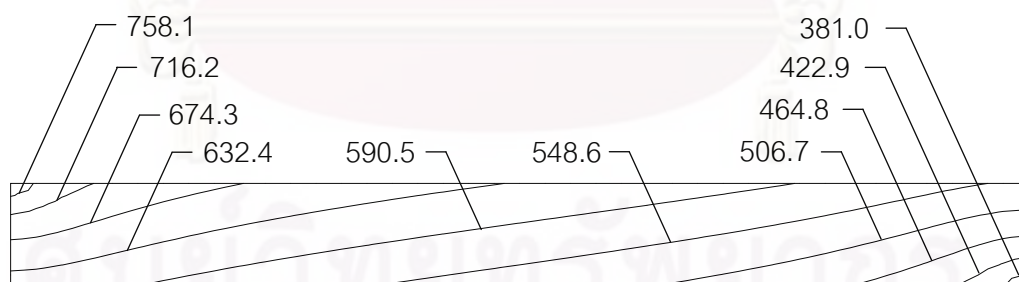


(ข) ค่าความเค้นเฉือนในของแข็ง (MPa)



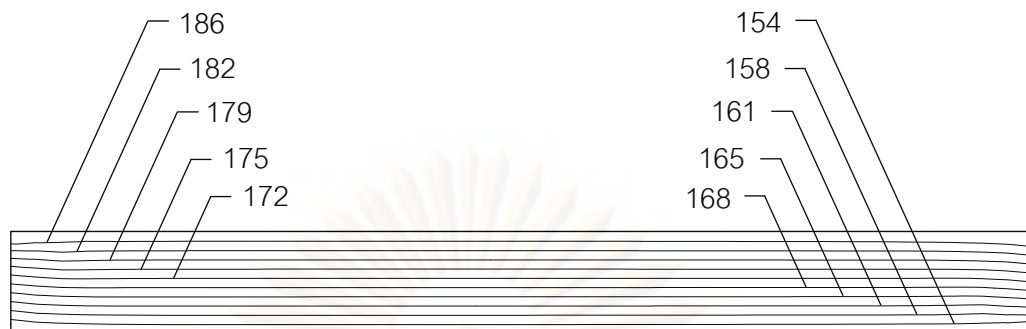
(ค) ค่าการเคลื่อนตัวพร้อมกับการเสียรูปขนาด 100 เท่าที่เกิดขึ้นภายในของแข็ง

รูปที่ 6.2(ต่อ) ผลการคำนวณสำหรับปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อนในกรณีที่มีค่า  $K_{sf} = 1$

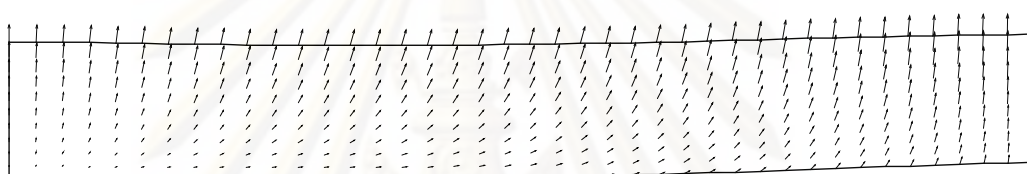


(ก) การกระจายตัวของอุณหภูมิ (K)

รูปที่ 6.3 ผลการคำนวณสำหรับปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อนในกรณีที่มีค่า  $K_{sf} = 5$

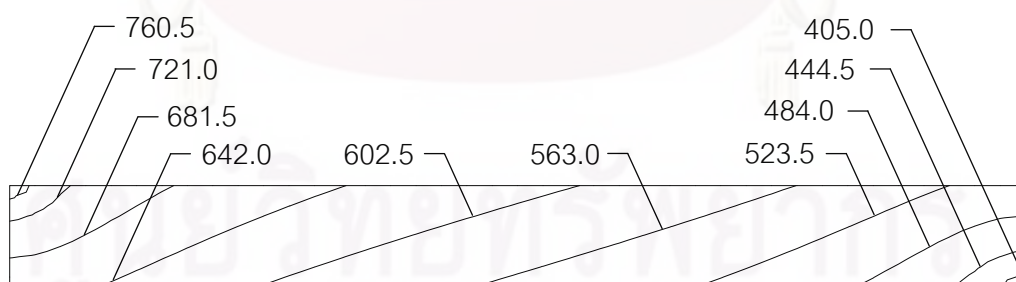


(ข) ค่าความเค้นเฉือนสูงสุดในของแข็ง (MPa)



(ค) ค่าการเคลื่อนตัวพร้อมกับการเสียรูปขนาด 100 เท่าที่เกิดขึ้นภายในของแข็ง

รูปที่ 6.3(ต่อ) ผลการคำนวณสำหรับปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อนในกรณีที่มีค่า  $K_{sf} = 5$

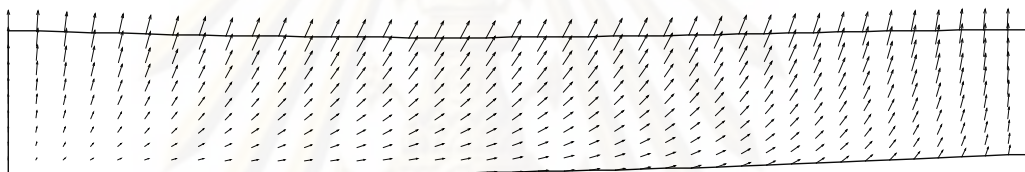


(ก) การกระจายตัวของอุณหภูมิ (K)

รูปที่ 6.4 ผลการคำนวณสำหรับปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อนในกรณีที่มีค่า  $K_{sf} = 10$



(ข) ค่าความเค้นวอนมิสเสสภายในของแข็ง (MPa)



(ค) ค่าการเคลื่อนตัวพร้อมกับการเสียรูปขนาด 100 เท่าที่เกิดขึ้นภายในของแข็ง

รูปที่ 6.4(ต่อ) ผลการคำนวณสำหรับปัญหาการไหลแบบสวนทางภายในอุปกรณ์  
แลกเปลี่ยนความร้อนในกรณีที่ว่า  $K_{sf} = 10$

ส่วนค่าการเคลื่อนตัวสูงสุดในแนวตั้งของของแข็งในกรณีที่ว่า  $K_{sf}$  มีค่าแตกต่างกันได้แสดงไว้ในตารางที่ 6.1 ซึ่งเห็นได้ว่ามีระยะการเคลื่อนตัวที่น้อยมาก จึงอาจกล่าวได้ว่าไม่มีผลกระทบต่อรูปแบบของการไหล

ตารางที่ 6.1 ค่าการเคลื่อนตัวสูงสุดในแนวตั้งของของแข็งสำหรับปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน กรณีค่า  $K_{sf}$  แตกต่างกัน

	อัตราส่วนสัมประสิทธิ์การนำความร้อน, $K_{sf}$		
	1	5	10
ค่าการเคลื่อนตัวสูงสุดในแนวตั้ง (mm)	0.4055	0.4111	0.4021



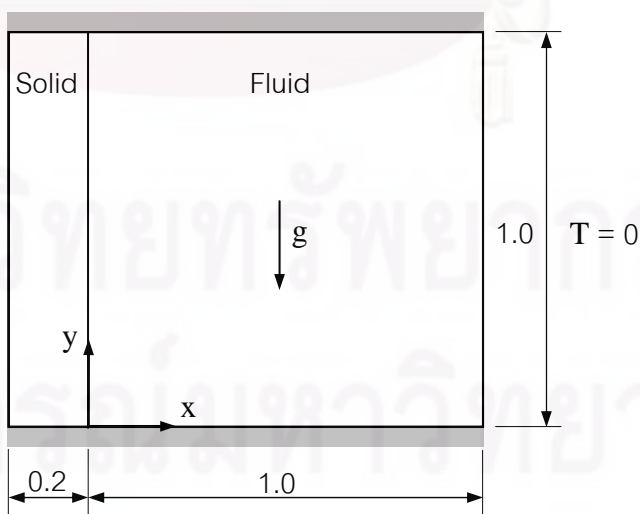
6.2 ปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสโดยที่มีผนังนำความร้อน (Conjugate natural convection in a square cavity with a conducting wall)

ในปัญหานี้จะทำการเปรียบเทียบผลลัพธ์เชิงตัวเลขของ Hriberšek & Kuhn [34] เพื่อตรวจสอบความถูกต้องของโปรแกรม ลักษณะของปัญหาประกอบไปด้วยช่องสี่เหลี่ยมจัตุรัสขนาดกว้าง 1x1 ซึ่งภายในบรรจุของไหลอยู่ และมีผนังนำความร้อนหนา 0.2 ผนังด้านบนและด้านล่างเป็นฉนวน ผนังด้านซ้ายมีอุณหภูมิสูงเท่ากับ 1 ในขณะที่ผนังด้านขวามีอุณหภูมิต่ำเท่ากับ 0 ซึ่งความแตกต่างของอุณหภูมิจะทำให้ของไหลเกิดการหมุนวนขึ้นในช่องสี่เหลี่ยมจัตุรัส ปัญหานี้จะคิดแรงเนื่องจากแรงโน้มถ่วง ดังแสดงในรูปที่ 6.5 จากนั้นทำการสร้างรูปแบบจำลองไฟไนต์เอลิเมนต์ประกอบไปด้วย 3,111 จุดต่อและ 6,000 เอลิเมนต์แสดงในรูปที่ 6.6

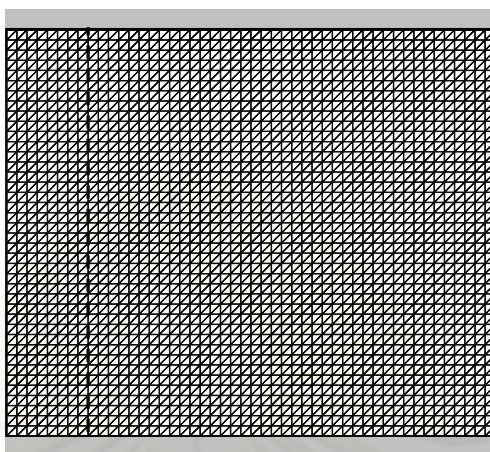
ในขั้นตอนการวิเคราะห์สภาวะการไหลสำหรับปัญหานี้ได้กำหนดให้พรันด์เทิลนัมเบอร์ (Prandtl Number;  $Pr$ ) มีค่าเท่ากับ 0.71 และอัตราส่วนของการแพร่กระจายความร้อนระหว่างของแข็งและของไหล  $\alpha_{sf} = K_{sf}$  ขณะเดียวกันกำหนดให้ค่ากราชอฟนัมเบอร์ (Grashof Number;  $Gr$ ) มีค่าเท่ากับ  $10^3, 10^4, 10^5, 10^6$  และ  $10^7$  โดยค่ากราชอฟนัมเบอร์คืออัตราส่วนของแรงลอยตัวเนื่องจากอุณหภูมิต่างกับความหนืดในของไหล ดังนี้

$$Gr = \frac{g\beta\rho^2\Delta TL^3}{\mu^2} \tag{6.1}$$

โดยที่  $L$  แทนความยาวเฉพาะ (Characteristic Length) โดยมีค่าเท่ากับ 1

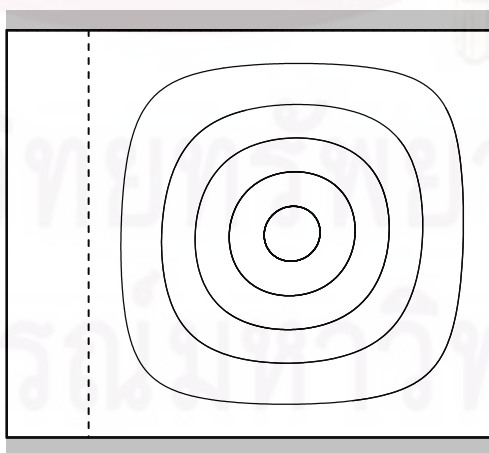


รูปที่ 6.5 ลักษณะของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสโดยที่มีผนังนำความร้อน

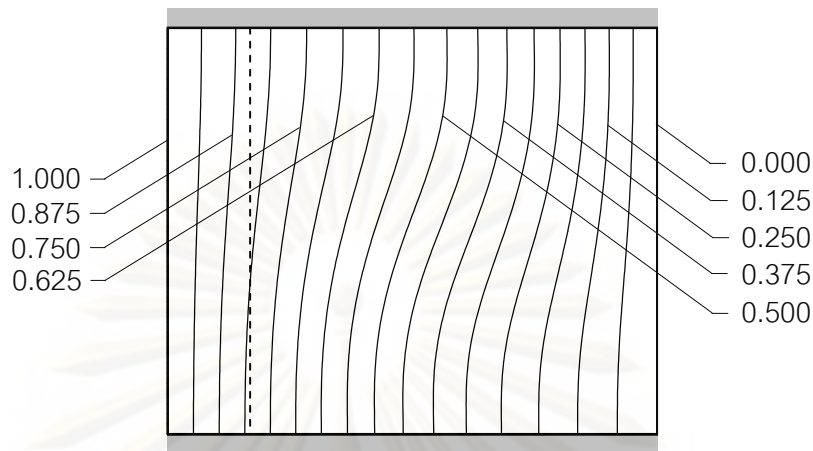


รูปที่ 6.6 แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการพาความร้อนแบบอิสระ  
ในช่องปิดสี่เหลี่ยมจัตุรัสโดยที่มีผนังนำความร้อน

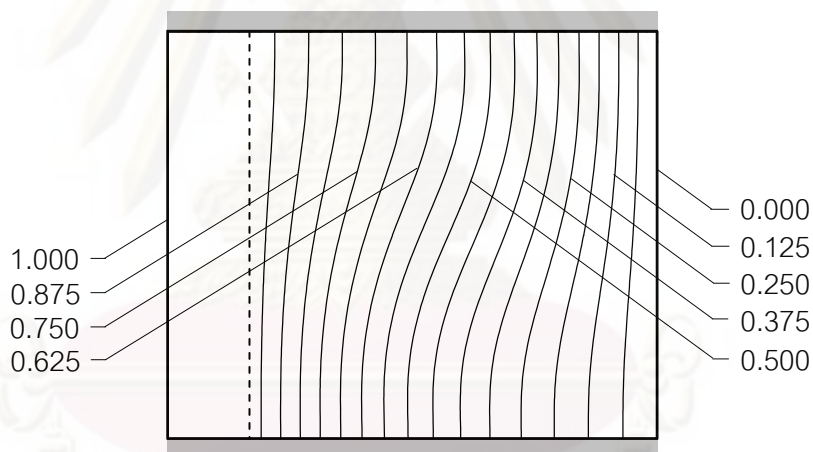
ผลลัพธ์ที่ได้จากการคำนวณสำหรับปัญหาดังกล่าวแสดงในรูปที่ 6.7 (ก-ค) ในรูปที่ 6.7 (ก) แสดงเส้นกระแสการไหล ซึ่งอธิบายถึงการไหลหมุนวนของของไหลในทิศตามเข็มนาฬิกาอันเนื่องมาจากของไหลเกิดการลอยตัวขึ้นในบริเวณผนังด้านซ้ายที่มีอุณหภูมิสูงและลอยตัวลงในบริเวณผนังด้านขวาที่มีอุณหภูมิต่ำโดยผลที่ได้จากค่าอัตราส่วนสัมประสิทธิ์การนำความร้อนในของแข็งต่อสัมประสิทธิ์การนำความร้อนในของไหล  $K_{sf}$  ที่ค่าต่าง ๆ นั้นผลที่ได้ไม่แตกต่างกันมากนัก รูปที่ 6.8 (ข-ค) จะแสดงถึงเส้นชั้นของอุณหภูมิ (Temperature Contours) ที่ค่า  $Gr = 10^3$  ขณะที่อัตราส่วนสัมประสิทธิ์การนำความร้อนในของแข็งต่อสัมประสิทธิ์การนำความร้อนในของไหล  $K_{sf}$  เท่ากับ 1 และ 10 ตามลำดับ



(ก) เส้นกระแสการไหล



(ข) เส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 1

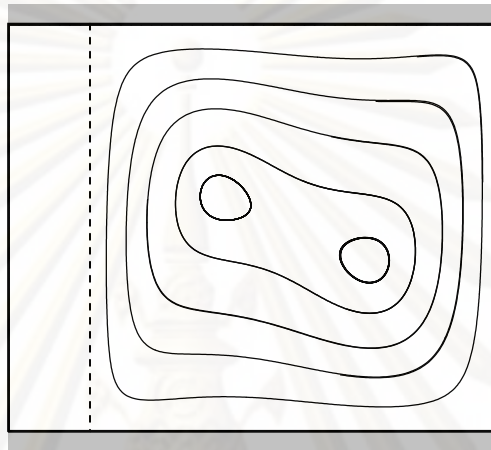


(ค) เส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 10

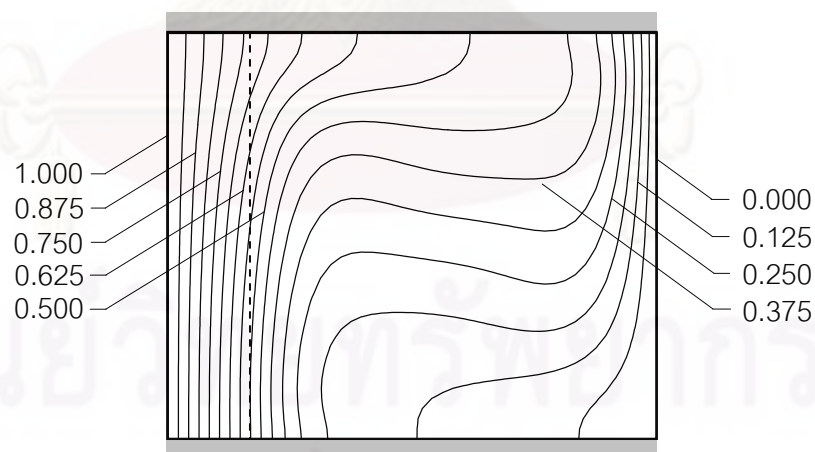
รูปที่ 6.7 สภาวะการไหลเนื่องจากการพาความร้อนในช่องสี่เหลี่ยมจัตุรัส  
เมื่อมีผนังนำความร้อนเมื่อค่า  $Gr = 10^3$

จากนั้นทำการวิเคราะห์สภาวะการไหลสำหรับปัญหานี้โดยการทำซ้ำโดยเพิ่มค่ากราชอร์ฟนัมเบอร์  $Gr$  มีค่าสูงขึ้นเป็น  $10^5$  ซึ่งหมายถึงแรงลอยตัวเนื่องจากความแตกต่างของอุณหภูมิเพิ่มขึ้นเป็น 100 เท่าของกรณีที่แล้วหรือค่าความหนืดของของไหลลดลงไป 100 เท่า ซึ่งก่อให้เกิดผลลัพธ์ของสภาวะการไหลดังแสดงในรูปที่ 6.8

ในรูปที่ 6.8 (ก) จะพบว่ามีการไหลหมุนวนในทิศตามเข็มนาฬิกาเพิ่มขึ้นเป็น 2 วง และค่าเส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 1 และ 10 นั้นบางและแคบลง ดังแสดงในรูป 6.8 (ข-ค) ในทำนองเดียวกันทำการเพิ่มค่ากราชอร์ฟนัมเบอร์  $Gr$  ให้ค่าสูงขึ้นเป็น  $10^7$  โดยแสดงค่าเส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 1, 5 และ 10 ตามลำดับ ดังแสดงในรูปที่ 6.9

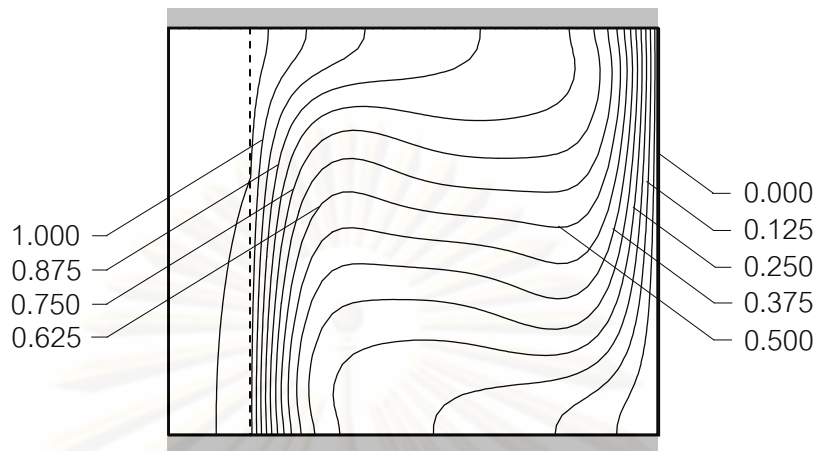


(ก) เส้นกระแสการไหล



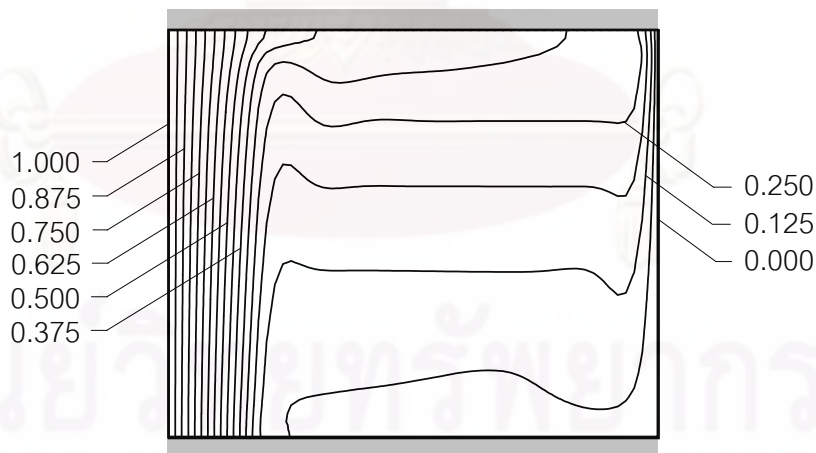
(ข) เส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 1

รูปที่ 6.8 สภาวะการไหลเนื่องจากการพาความร้อนในช่องสี่เหลี่ยมจัตุรัส  
เมื่อมีผนังนำความร้อนเมื่อค่า  $Gr = 10^5$



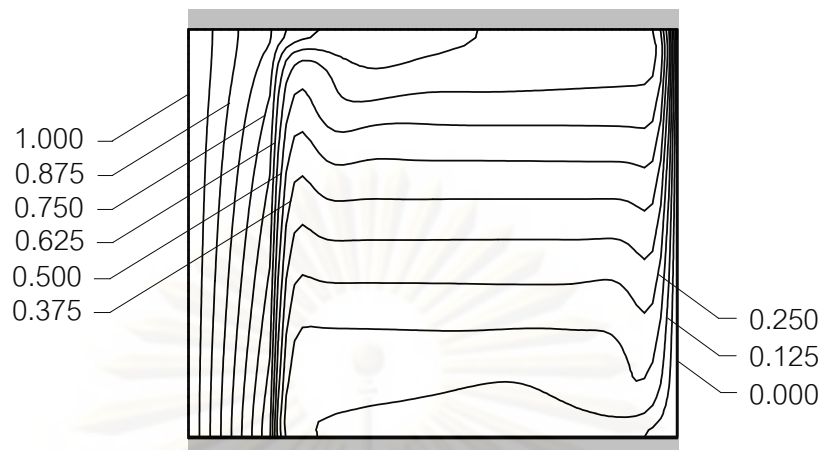
(ค) เส้นชั้นของอุณหภูมิต่ำที่  $K_{sf}$  เท่ากับ 10

รูปที่ 6.8(ต่อ) สภาวะการไหลเนื่องจากการพาความร้อนในช่องสี่เหลี่ยมจัตุรัส  
เมื่อมีผนังนำความร้อนเมื่อค่า  $Gr = 10^5$

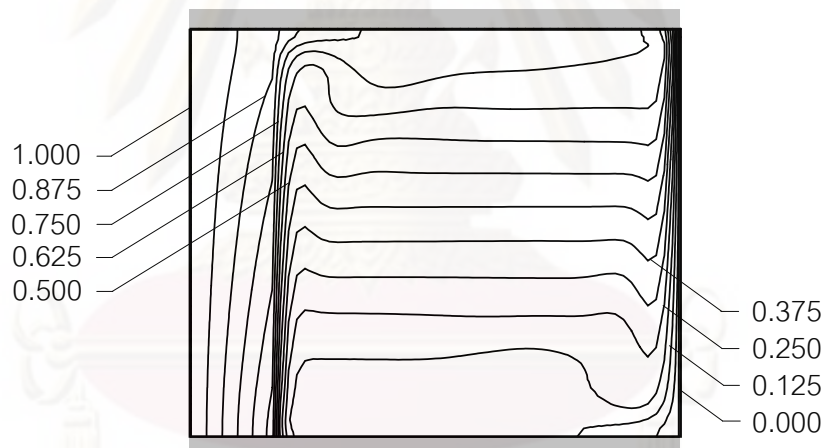


(ก) เส้นชั้นของอุณหภูมิต่ำที่  $K_{sf}$  เท่ากับ 1

รูปที่ 6.9 สภาวะการไหลเนื่องจากการพาความร้อนในช่องสี่เหลี่ยมจัตุรัส  
เมื่อมีผนังนำความร้อนเมื่อค่า  $Gr = 10^7$



(ข) เส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 5

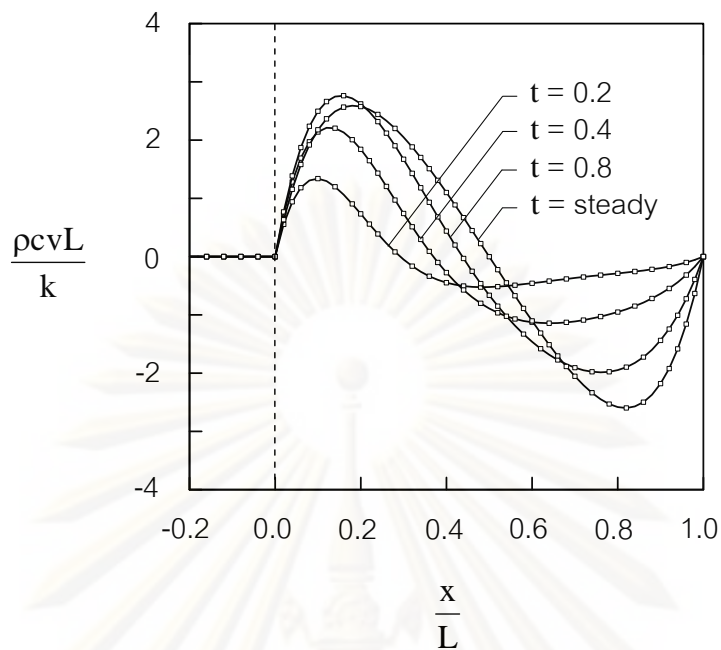


(ค) เส้นชั้นของอุณหภูมิที่  $K_{sf}$  เท่ากับ 10

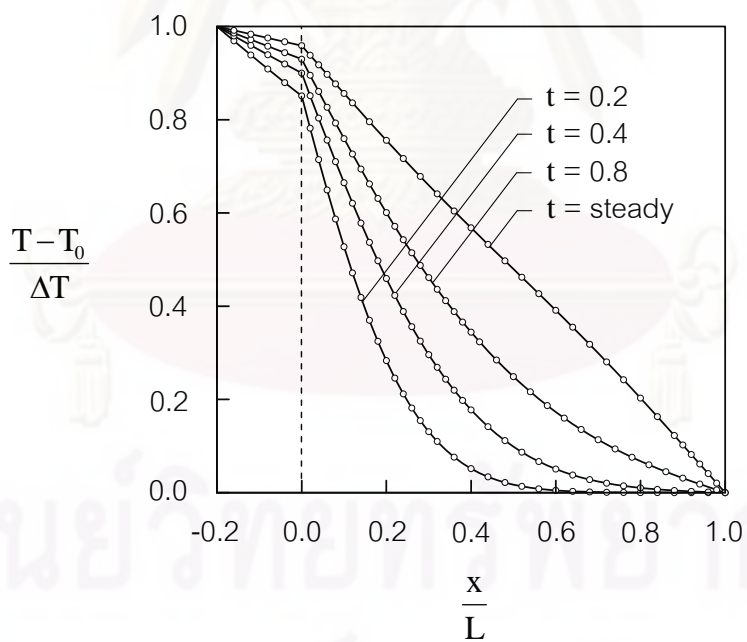
รูปที่ 6.9(ต่อ) สถานะการไหลเนื่องจากการพาความร้อนในช่องสี่เหลี่ยมจัตุรัส

เมื่อมีพจน์นำความร้อนเมื่อค่า  $Gr = 10^7$

จากนั้นแสดงการกระจายความเร็วและอุณหภูมิไว้มิติตลอดแนวแกน  $x$  ที่ระดับกึ่งกลางความสูงของช่องสี่เหลี่ยมจัตุรัสที่ค่า  $Gr$  เท่ากับ  $10^3$ ,  $10^4$  และ  $10^5$  ณ เวลาต่าง ๆ ดังแสดงในรูปที่ 6.10, 6.11 และ 6.12 ตามลำดับ

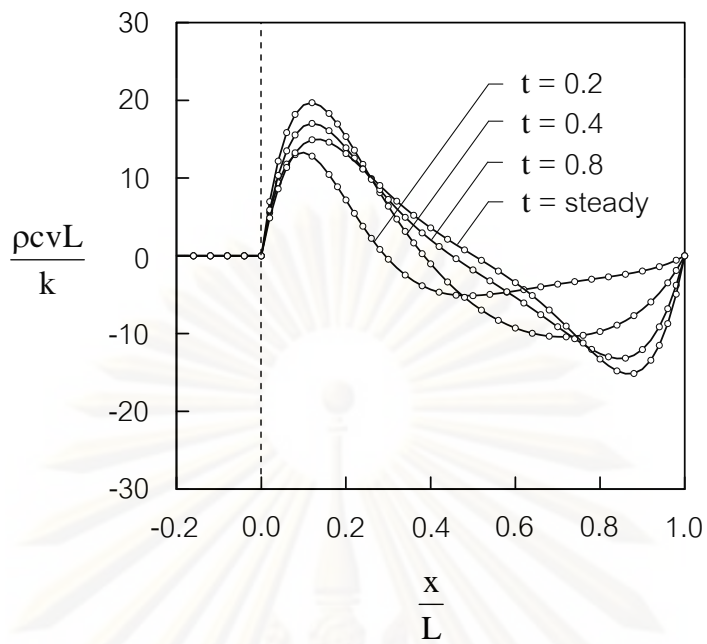


(ก) ความเร็วไร้มิติ

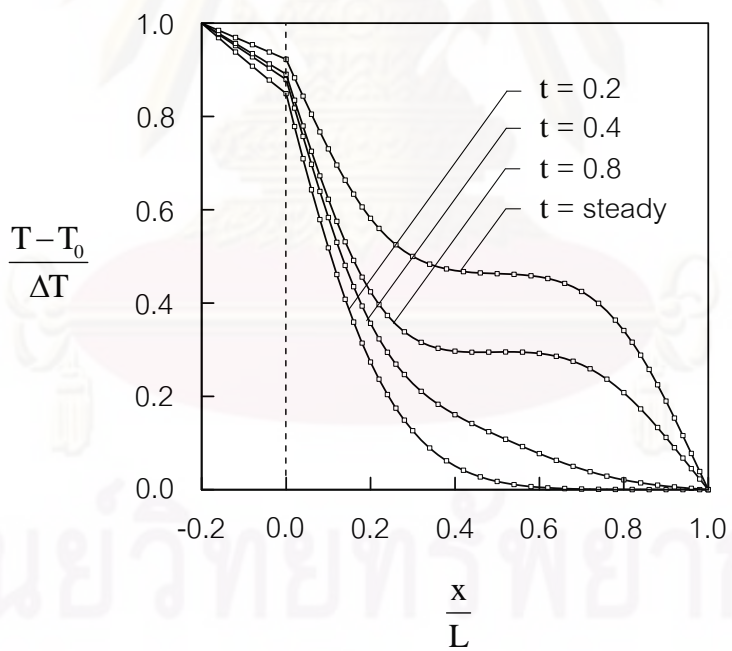


(ข) อุณหภูมิไร้มิติ

รูปที่ 6.10 การเปรียบเทียบลักษณะการกระจายของความเร็วจึงและอุณหภูมิไร้มิติตลอดแนวแกน  $x$  ที่ค่า  $Gr = 10^3$  ณ เวลาต่าง ๆ



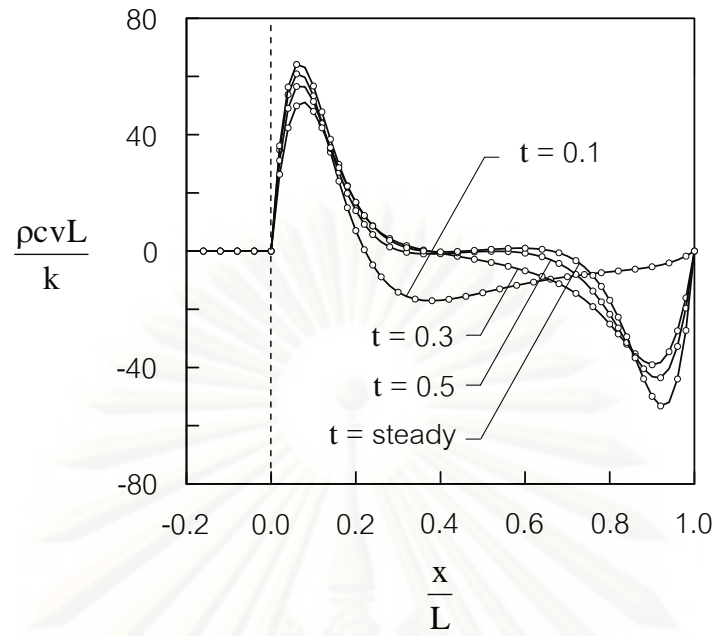
(ก) ความเร็วไร้มิติ



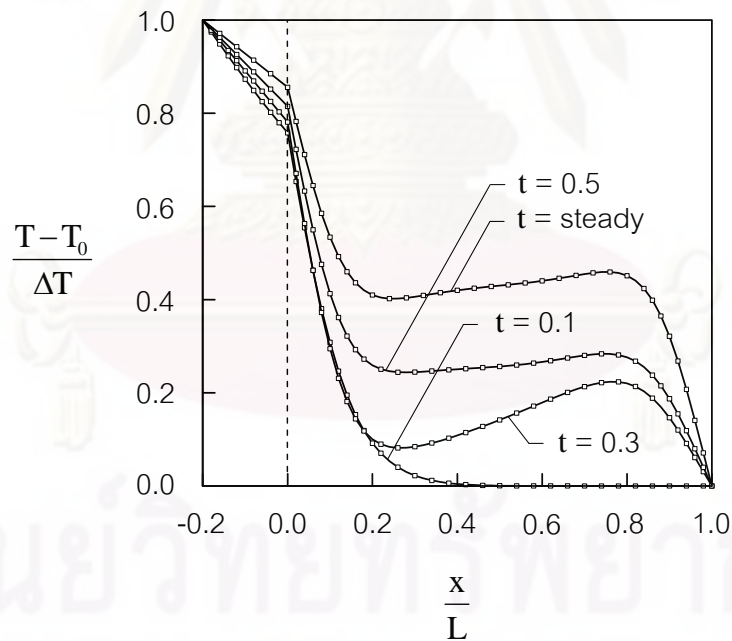
(ข) อุณหภูมิไร้มิติ

รูปที่ 6.11 การเปรียบเทียบลักษณะการกระจายของความเร็วจึงและอุณหภูมิไร้มิติตลอดแนวแกน  $x$  ที่ค่า  $Gr = 10^4$  ณ เวลาต่าง ๆ





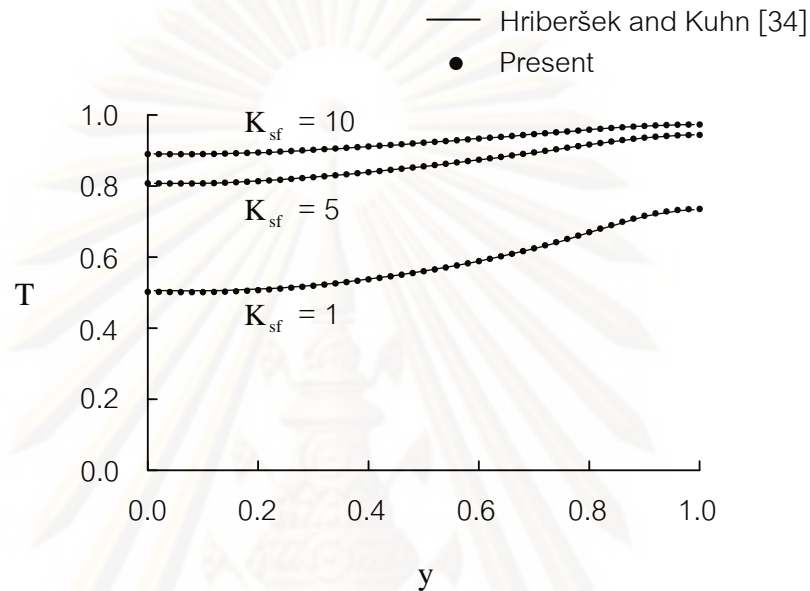
(ก) ความเร็วไร้มิติ



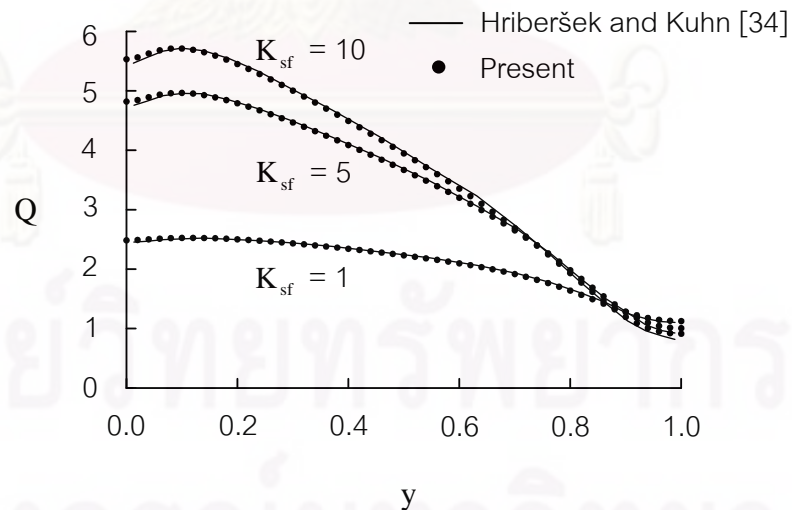
(ข) อุณหภูมิไร้มิติ

รูปที่ 6.12 การเปรียบเทียบลักษณะการกระจายของความเร็วจึงและอุณหภูมิไร้มิติ  
ตลอดแนวแกน  $x$  ที่ค่า  $Gr = 10^5$  ณ เวลาต่าง ๆ

จากนั้นทำการตรวจสอบความถูกต้องโดยเปรียบเทียบกับผลลัพธ์ที่ได้กับ Hriberšek & Kuhn [34] โดยทำการเปรียบเทียบค่าอุณหภูมิและปริมาณความร้อน (Heat Fluxes) ที่บริเวณผิวรอยต่อระหว่างของแข็งและของไหล ( $x = 0.0$ ) ตลอดแนวแกน  $y$  ที่ค่า  $Gr = 10^5$  และ  $10^7$  ดังแสดงในรูปที่ 6.13 - 6.14 ตามลำดับ

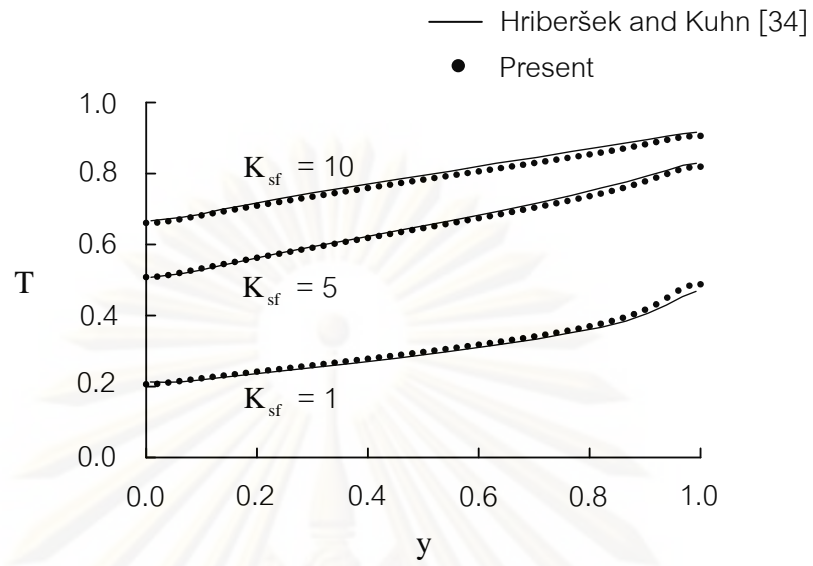


(ก) อุณหภูมิที่ผิวรอยต่อ

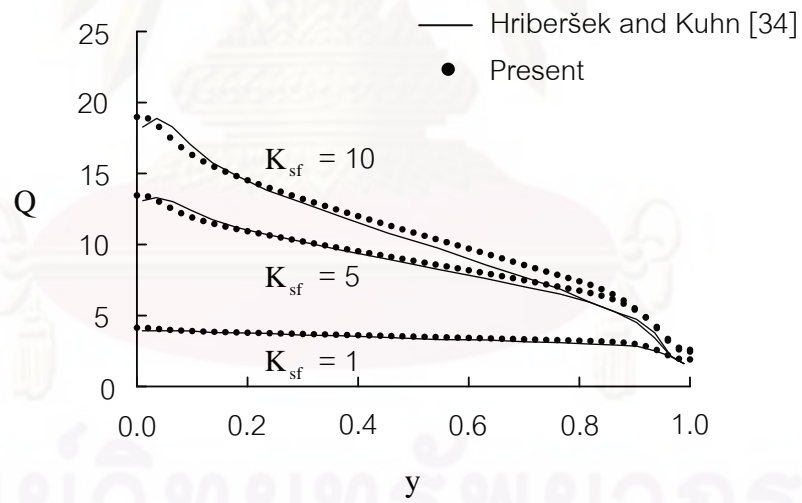


(ข) ปริมาณความร้อนที่ผิวรอยต่อ

รูปที่ 6.13 การเปรียบเทียบอุณหภูมิและปริมาณความร้อนที่ผิวรอยต่อโดยเปรียบเทียบกับผลการคำนวณของ Hriberšek & Kuhn ที่ค่า  $K_{sf} = 1, 5$  และ  $10$  เมื่อค่า  $Gr = 10^5$



(ก) อุณหภูมิที่ผิวรอยต่อ



(ข) ปริมาณความร้อนที่ผิวรอยต่อ

รูปที่ 6.14 การเปรียบเทียบอุณหภูมิและปริมาณความร้อนที่ผิวรอยต่อโดยเปรียบเทียบกับผลการคำนวณของ Hriberšek & Kuhn ที่ค่า  $K_{sf} = 1, 5$  และ 10 เมื่อค่า  $Gr = 10^7$

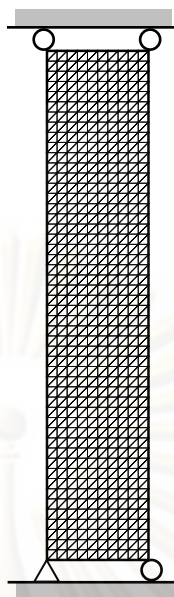
จากตารางที่ 6.2 แสดงการเปรียบเทียบค่าตัวเลขนัมเบอร์เฉลี่ยที่บริเวณผิวรอยต่อระหว่างของแข็งและของไหล,  $\overline{Nu}_{x=0.0}$  โดยเปรียบเทียบกับผลการคำนวณของ Hriberšek & Kuhn [34] ผลที่ได้ก็ให้ค่าเป็นที่น่าพอใจ

ตารางที่ 6.2 การเปรียบเทียบค่าตัวเลขนัมเบอร์เฉลี่ยที่บริเวณผิวรอยต่อ

Gr	ค่าตัวเลขนัมเบอร์เฉลี่ยที่บริเวณผิวรอยต่อระหว่างของแข็งและของไหล			
	Conductivity ratio, $K_{sf}$	1	5	10
$10^3$	Hriberšek and Kuhn [34]	0.87	1.02	1.04
	Present	0.87 (0.0%)	1.02 (0.0%)	1.04 (0.0%)
$10^4$	Hriberšek and Kuhn [34]	1.35	1.83	1.92
	Present	1.35 (0.0%)	1.83 (0.0%)	1.91 (0.52%)
$10^5$	Hriberšek and Kuhn [34]	2.08	3.42	3.72
	Present	2.08 (0.0%)	3.40 (0.58%)	3.70 (0.54%)
$10^6$	Hriberšek and Kuhn [34]	2.87	5.88	6.78
	Present	2.83 (1.39%)	5.80 (1.36%)	6.69 (1.33%)
$10^7$	Hriberšek and Kuhn [34]	3.53	9.07	11.25
	Present	3.45 (2.27%)	8.73 (3.75%)	10.88 (3.29%)

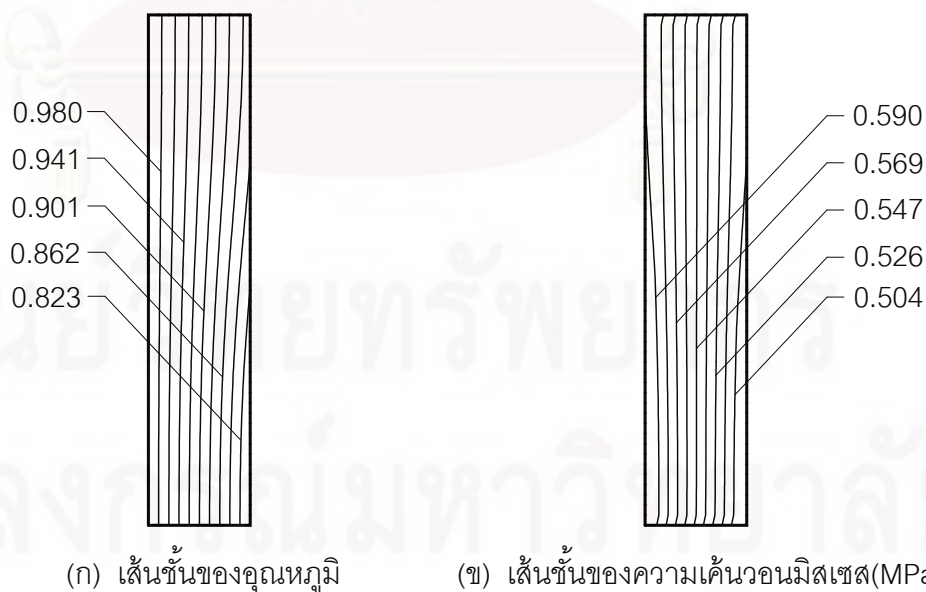
หมายเหตุ : ค่าตัวเลขนัมเบอร์เฉลี่ยที่บริเวณผิวรอยต่อ (ค่าความคลาดเคลื่อน % โดยเปรียบเทียบกับ Hriberšek & Kuhn [34])

เมื่อวิเคราะห์การไหลและการถ่ายเทความร้อนทั้งภายในของไหลและของแข็งเสร็จเรียบร้อยแล้ว ขั้นตอนต่อไปก็นำค่าการกระจายตัวของอุณหภูมิที่ได้และค่าความดันมาคำนวณหาค่าความเค้นที่เกิดขึ้นภายในของแข็ง โดยจะทำการวิเคราะห์ที่ค่า  $Gr = 10^3, 10^5,$  และ  $10^7$  ณ ค่า  $K_{sf}$  ต่าง ๆ กัน โดยกำหนดให้ของแข็งที่จะทำการวิเคราะห์มีค่าคุณสมบัติดังต่อไปนี้ มีค่าโมดูลัสของความยืดหยุ่น  $7 \times 10^{10} \text{ N/m}^2$  ค่าสัมประสิทธิ์การขยายตัว  $6 \times 10^{-6} / ^\circ\text{C}$  ค่าอัตราส่วนปัวส์ซองเท่ากับ 0.3 และอุณหภูมิขณะที่ไม่มีความเค้นคือที่  $0^\circ\text{C}$  สำหรับรูปแบบไฟไนต์เอลิเมนต์และเงื่อนไขขอบเขตของการวิเคราะห์ปัญหาความเค้นอันเนื่องมาจากอุณหภูมิสำหรับตัวอย่างนี้ได้แสดงในรูปที่ 6.15 ซึ่งประกอบไปด้วย 561 จุดต่อ 1,000 เอลิเมนต์

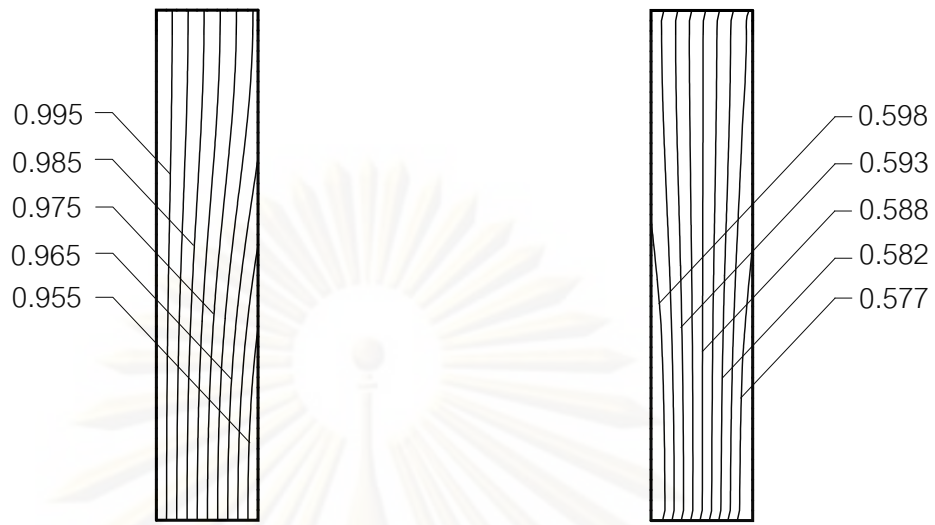


รูปที่ 6.15 แบบจำลองไฟไนต์เอลิเมนต์และเงื่อนไขขอบเขตสำหรับวิเคราะห์ความเค้นในปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสโดยที่มีผนังนำความร้อน

รูปที่ 6.16 - 6.18 แสดงลักษณะการกระจายตัวของอุณหภูมิ และค่าความเค้นวอนนิสเชส(MPa) ในกรณีที่มีการเปลี่ยนแปลงค่าอัตราส่วนของสัมประสิทธิ์การถ่ายเทความร้อน  $K_{sf}$  เท่ากับ 1, 5 และ 10 ตามลำดับ โดยคำนวณที่ค่า  $Gr$  เท่ากับ  $10^3$  ในทำนองเดียวกันแสดงลักษณะการกระจายตัวของอุณหภูมิ และค่าความเค้นวอนนิสเชสที่ค่า  $Gr$  เท่ากับ  $10^5$  และ  $10^7$  ดังแสดงในรูปที่ 6.19 - 6.24 ตามลำดับ

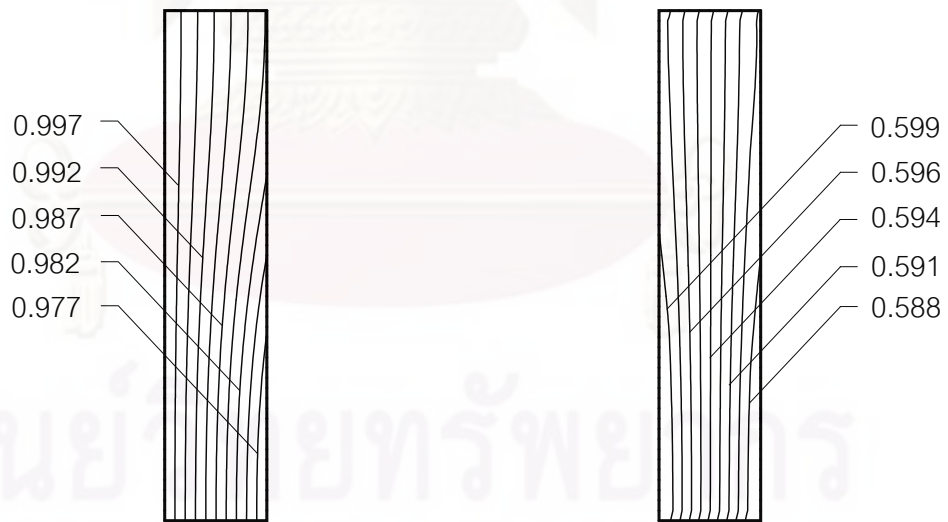


รูปที่ 6.16 การกระจายตัวของอุณหภูมิและค่าความเค้นวอนนิสเชสภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี  $Gr = 10^3$  และ  $K_{sf} = 1$



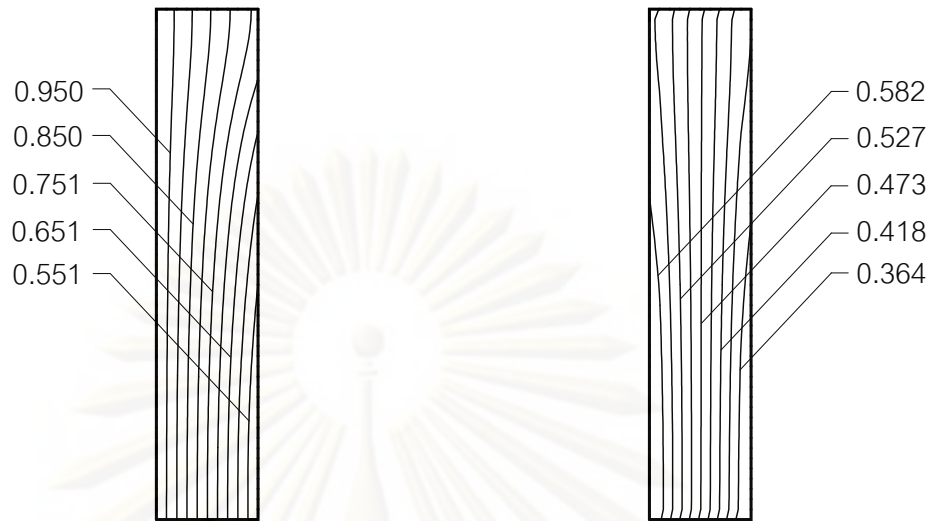
(ก) เส้นชั้นของอุณหภูมิจากบนลงล่าง (ข) เส้นชั้นของความเค้นวอนมีสเซส(MPa) จากบนลงล่าง

รูปที่ 6.17 การกระจายตัวของอุณหภูมิจากบนลงล่างและค่าความเค้นวอนมีสเซสภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี  $Gr = 10^3$  และ  $K_{sf} = 5$



(ก) เส้นชั้นของอุณหภูมิจากบนลงล่าง (ข) เส้นชั้นของความเค้นวอนมีสเซส(MPa) จากบนลงล่าง

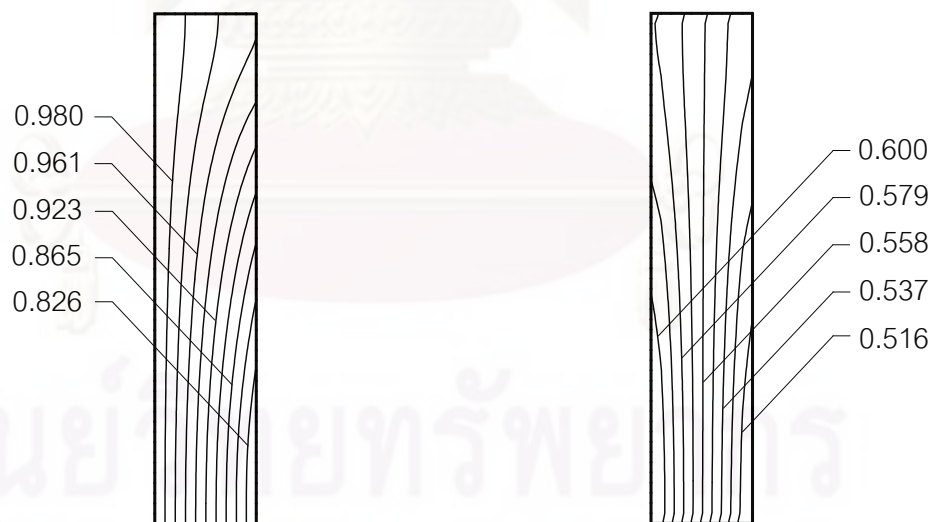
รูปที่ 6.18 การกระจายตัวของอุณหภูมิจากบนลงล่างและค่าความเค้นวอนมีสเซสภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี  $Gr = 10^3$  และ  $K_{sf} = 10$



(ก) เส้นชั้นของอุณหภูมิ

(ข) เส้นชั้นของความเค้นอนมิสเซส(MPa)

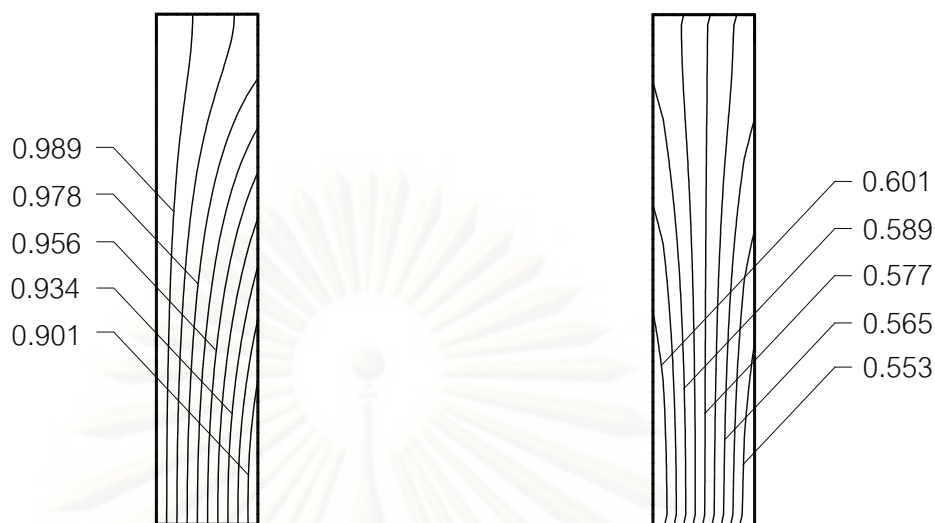
รูปที่ 6.19 การกระจายตัวของอุณหภูมิและค่าความเค้นอนมิสเซสภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี  $Gr = 10^5$  และ  $K_{sf} = 1$



(ก) เส้นชั้นของอุณหภูมิ

(ข) เส้นชั้นของความเค้นอนมิสเซส(MPa)

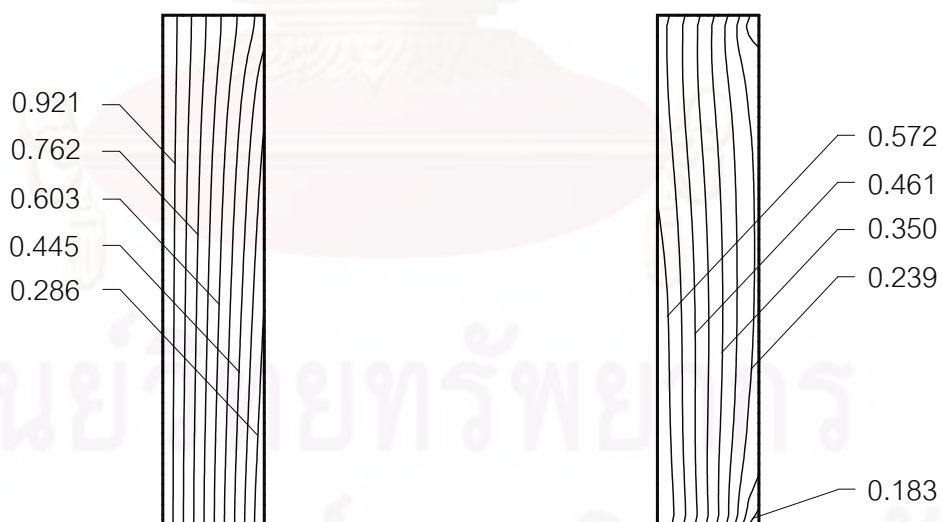
รูปที่ 6.20 การกระจายตัวของอุณหภูมิและค่าความเค้นอนมิสเซสภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี  $Gr = 10^5$  และ  $K_{sf} = 5$



(ก) เส้นชั้นของอุณหภูมิ

(ข) เส้นชั้นของความเค้นวอนมิสเชส(MPa)

รูปที่ 6.21 การกระจายตัวของอุณหภูมิและค่าความเค้นวอนมิสเชสภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี  $Gr = 10^5$  และ  $K_{sf} = 10$

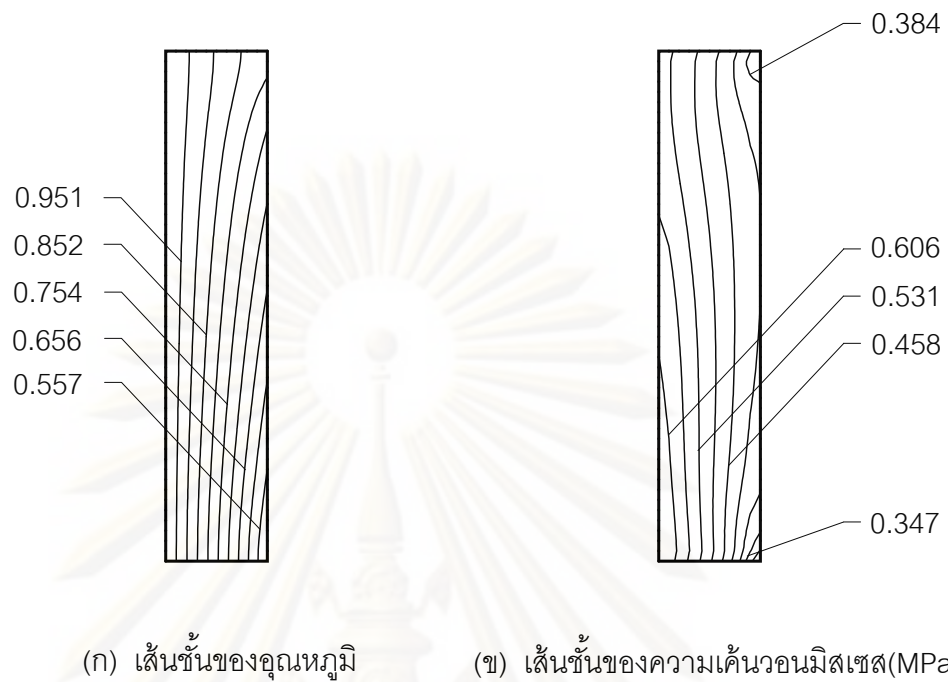


(ก) เส้นชั้นของอุณหภูมิ

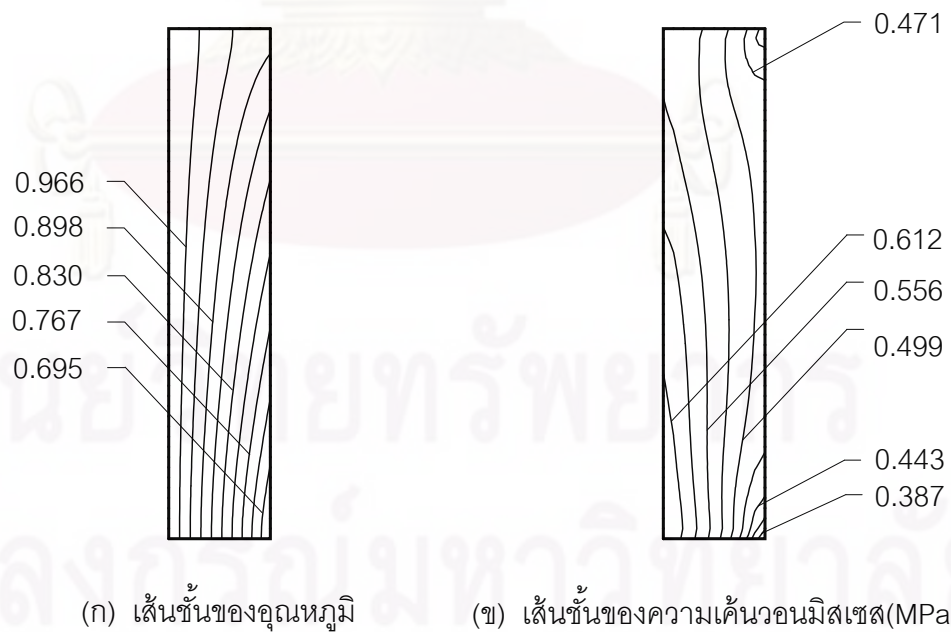
(ข) เส้นชั้นของความเค้นวอนมิสเชส(MPa)

รูปที่ 6.22 การกระจายตัวของอุณหภูมิและค่าความเค้นวอนมิสเชสภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี  $Gr = 10^7$  และ  $K_{sf} = 1$





รูปที่ 6.23 การกระจายตัวของอุณหภูมิตัวและค่าความเค้นวอนมิสเซลภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี  $Gr = 10^7$  และ  $K_{sf} = 5$



รูปที่ 6.24 การกระจายตัวของอุณหภูมิตัวและค่าความเค้นวอนมิสเซลภายในของแข็งของปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสกรณี  $Gr = 10^7$  และ  $K_{sf} = 10$

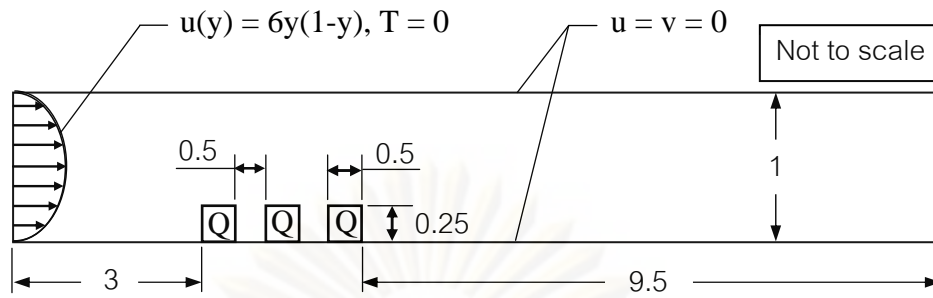
ส่วนค่าการเคลื่อนตัวสูงสุดในแนวนอนของของแข็งในกรณีที่มีค่า  $K_{sf}$  มีค่าแตกต่างกันได้แสดงไว้ในตารางที่ 6.3 จะพบว่าระยะการเคลื่อนตัวมีค่าที่น้อยมาก จึงอาจกล่าวได้ว่าไม่มีผลกระทบต่อรูปแบบของการไหล

ตารางที่ 6.3 ค่าการเคลื่อนตัวสูงสุดในแนวนอนของของแข็งสำหรับปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัส กรณีค่า  $K_{sf}$  แตกต่างกัน

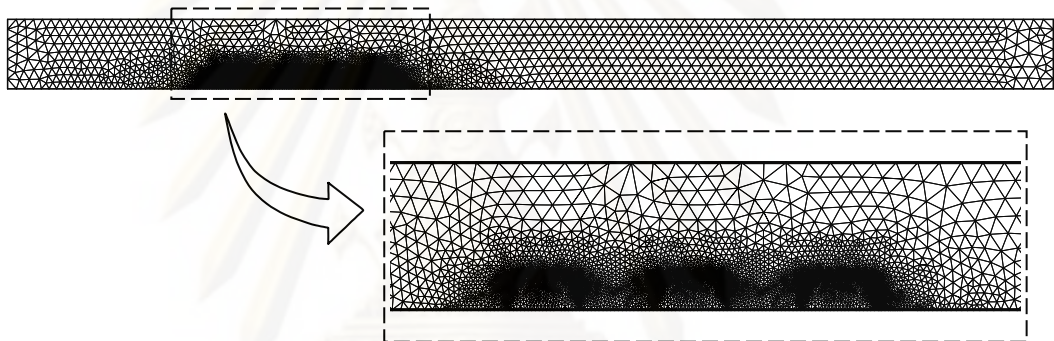
Gr	ค่าการเคลื่อนตัวสูงสุดในแนวนอน (mm)		
	$K_{sf} = 1$	$K_{sf} = 5$	$K_{sf} = 10$
$10^3$	$2.06 \times 10^{-3}$	$2.19 \times 10^{-3}$	$2.21 \times 10^{-3}$
$10^5$	$1.88 \times 10^{-3}$	$2.14 \times 10^{-3}$	$2.18 \times 10^{-3}$
$10^7$	$1.51 \times 10^{-3}$	$1.90 \times 10^{-3}$	$2.03 \times 10^{-3}$

### 6.3 ปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายในช่องทางการไหล (forced convection cooling across rectangular blocks)

ลักษณะของปัญหาจะเป็นช่องการไหลยาวซึ่งมีของไหลที่ปรับตัวเต็มที่แล้วไหลเข้ามาทางด้านซ้าย และภายในช่องการไหลนี้จะมีครีบบระบายความร้อนเป็นทรงสี่เหลี่ยมวางอยู่ที่ผนังด้านล่างจำนวนสามอัน โดยแต่ละอันจะมีการผลิตความร้อนขึ้นเอง ดังแสดงในรูปที่ 6.25 พร้อมทั้งแสดงขนาดของปัญหาและเงื่อนไขขอบเขตที่ใช้ในการวิเคราะห์ด้วย สำหรับคุณสมบัติของการไหลนั้นกำหนดให้ของไหลมีค่า  $Re = 100, 500$  และ  $1,000$  (คำนวณเทียบกับขนาดช่องการไหลเข้า) โดยมีค่า  $Pr = 0.7$  อัตราส่วนของการแพร่กระจายความร้อนระหว่างของแข็งและของไหล  $\alpha_{sf} = K_{sf} = 10$  ส่วนบริเวณครีบบระบายความร้อนทรงสี่เหลี่ยมกำหนดให้มีการผลิตความร้อนต่อปริมาตร  $Q = 8$  จากนั้นทำการวิเคราะห์ปัญหาโดยการสร้างรูปแบบไฟไนต์เอลิเมนต์ประกอบไปด้วย 5,653 จุดต่อ 10,933 เอลิเมนต์ ดังแสดงในรูปที่ 6.26 จากนั้นรูปที่ 6.27 แสดงเส้นชั้นของความดัน เส้นกระแสการไหลและแสดงเส้นชั้นของอุณหภูมิที่ค่า  $Re = 100$  ตามลำดับ ในทำนองเดียวกันที่ค่า  $Re = 500$  ดังแสดงในรูปที่ 6.28 จากผลที่ได้เมื่อพิจารณาค่าอุณหภูมิไรหน่วยสูงสุดจะเกิดขึ้นบริเวณครีบบระบายอันที่ 3 นับจากซ้ายมือมีค่าเท่ากับ 0.2813 ดังแสดงในรูปที่ 6.27(ค) และค่าอุณหภูมิไรหน่วยสูงสุดในรูปที่ 6.28(ค) มีค่าเท่ากับ 0.1633 จะเห็นได้ว่าเมื่อทำการเพิ่มค่าเรย์โนลด์ให้สูงขึ้นก็จะสามารถระบายความร้อนออกจากครีบบระบายความร้อนได้ดียิ่งขึ้น



รูปที่ 6.25 ลักษณะปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายในช่องทางการไหล



รูปที่ 6.26 แบบจำลองไฟไนต์เอลิเมนต์ของปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายในช่องทางการไหล

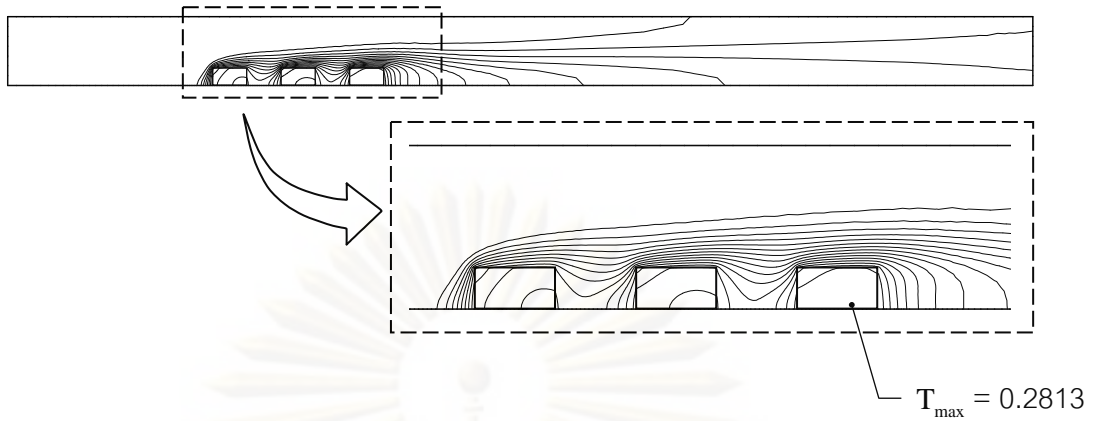


(ก) เส้นชั้นของความดัน



(ข) เส้นกระแสการไหล

รูปที่ 6.27 สภาวะการไหลของปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายในช่องทางการไหลที่ค่า  $Re = 100$



(ค) เส้นชั้นของอุณหภูมิ

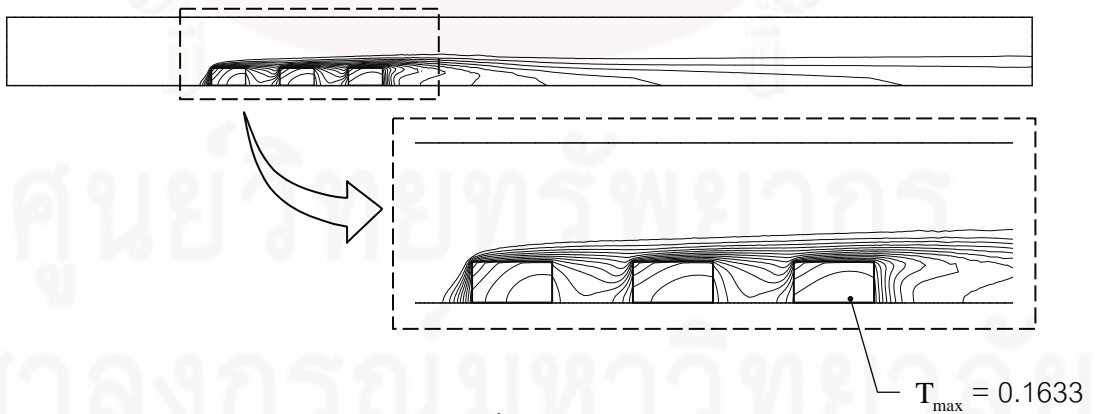
รูปที่ 6.27(ต่อ) สภาวะการไหลของปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายในช่องทางการไหลที่ค่า  $Re = 100$



(ก) เส้นชั้นของความดัน



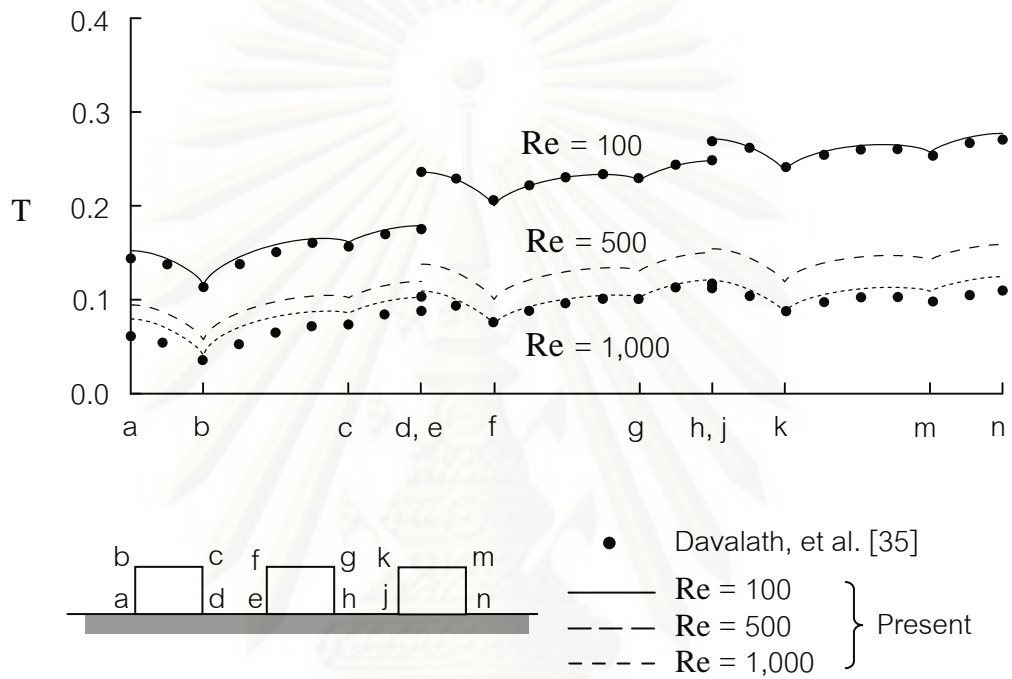
(ข) เส้นกระแสการไหล



(ค) เส้นชั้นของอุณหภูมิ

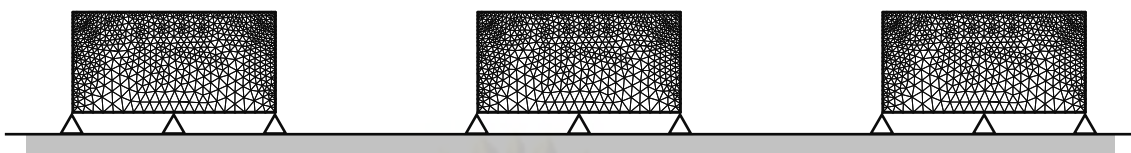
รูปที่ 6.28 สภาวะการไหลของปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายในช่องทางการไหลที่ค่า  $Re = 500$

ผลการคำนวณในตัวอย่งนี้จะได้นำไปเปรียบเทียบกับผลการคำนวณของ Davalath and Bayazitoglu [35] โดยจะทำการเปรียบเทียบการกระจายตัวของอุณหภูมิที่ผิวของครีกระบายความร้อนทั้งสามเฉพาะที่ค่า Re เท่ากับ 100 และ 1,000 เท่านั้น ดังแสดงในรูปที่ 6.29 ผลที่ได้ให้ค่าสอดคล้องกัน



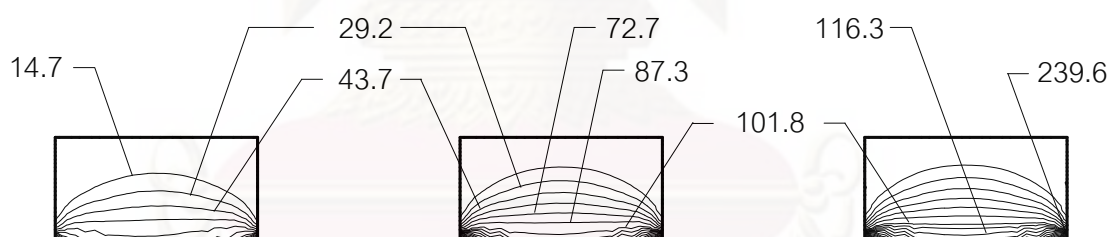
รูปที่ 6.29 เปรียบเทียบการกระจายตัวของอุณหภูมิที่ผิวของครีกระบายความร้อนทั้ง 3 ที่ค่า Re เท่ากับ 100 และ 1,000 กับผลการคำนวณของ Davalath and Bayazitoglu

สุดท้ายนำผลลัพธ์ที่ได้มาใช้เป็นเงื่อนไขขอบเขตเพื่อทำการคำนวณหาค่าการเคลื่อนตัวและค่าความเค้นที่เกิดขึ้นบนครีกระบายความร้อนทั้งสาม โดยรูปที่ 6.30 แสดงรูปแบบไฟไนต์เอลิเมนต์ในส่วนที่เป็นของแข็งที่จะใช้ในการคำนวณ ซึ่งประกอบไปด้วย 2,073 จุดต่อ 3,780 เอลิเมนต์ โดยกำหนดคุณสมบัติของของแข็งดังนี้ ค่าโมดูลัสของความยืดหยุ่นเท่ากับ  $7 \times 10^{10}$  N/m<sup>2</sup> ค่าสัมประสิทธิ์การขยายตัว  $6 \times 10^{-6}/^{\circ}\text{C}$  ค่าอัตราส่วนปัวซองเท่ากับ 0.3 และอุณหภูมิขณะที่ไม่มีความเค้นคือที่ 0°C

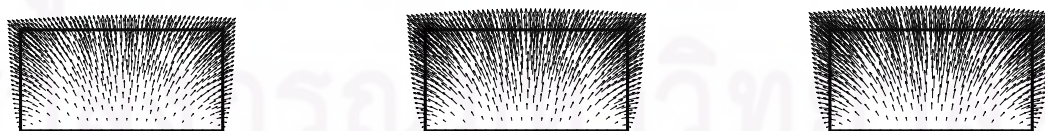


รูปที่ 6.30 รูปแบบไฟไนต์เอลิเมนต์ของครีกระบายความร้อนทั้งสามและเงื่อนไขขอบเขต  
สำหรับปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายใน  
ช่องทางการไหล

จากนั้นทำการวิเคราะห์เพื่อหาค่าการเคลื่อนตัว แล้วทำการคำนวณหาค่าความเค้นตามลำดับ รูป  
ที่ 6.31 แสดงการกระจายตัวของค่าความเค้นวอนมิสเซสบนครีกระบายความร้อนทั้งสาม โดยมี  
ค่าสูงสุดเท่ากับ 239.6 kPa ส่วนรูปที่ 6.32 นั้นแสดงเวกเตอร์การเคลื่อนตัวของแต่ละจุดต่อ โดยค่า  
การเคลื่อนตัวสูงสุดมีค่าเท่ากับ  $5.65 \times 10^{-4}$  mm ในทำนองเดียวกันสามารถแสดงค่าความเค้นวอน  
มิสเซสสูงสุดและค่าการเคลื่อนตัวสูงสุดได้ดังตารางที่ 6.4 จากตารางพบว่าระยะการเคลื่อนตัวมี  
ค่าน้อยมาก จึงอาจกล่าวได้ว่าไม่มีผลกระทบต่อรูปแบบของการไหล



รูปที่ 6.31 การกระจายตัวของค่าความเค้นวอนมิสเซส (kPa) บนครีกระบายความร้อน  
ทั้งสามที่ค่า Re เท่ากับ 100



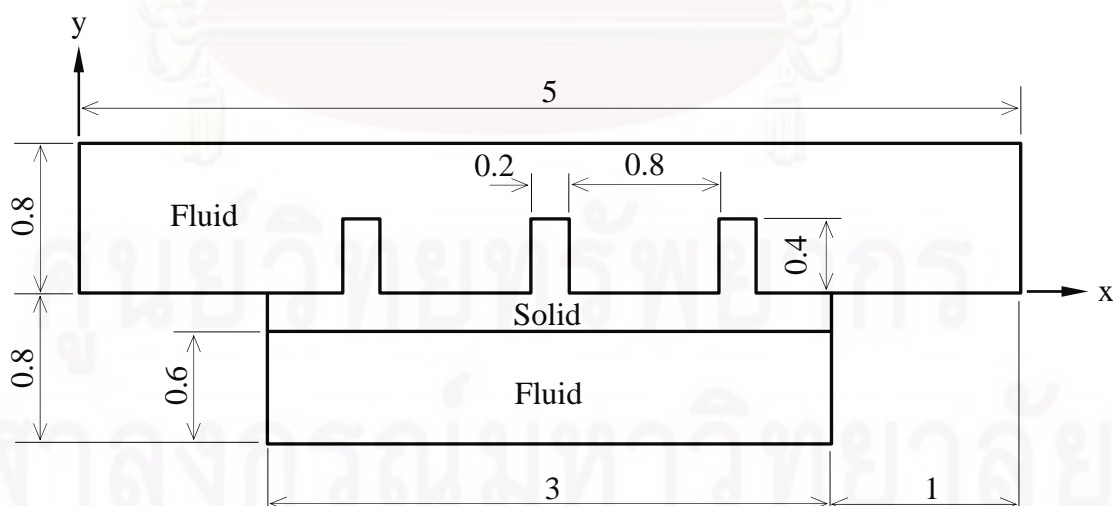
รูปที่ 6.32 เวกเตอร์แสดงการเคลื่อนตัวของแต่ละจุดต่อบนครีกระบายความร้อนทั้งสาม  
ที่ค่า Re เท่ากับ 100

ตารางที่ 6.4 แสดงค่าความเค้นวอนนิสเซสสูงสุดและค่าการเคลื่อนตัวสูงสุดบนครีบริบาย ความร้อนที่ค่า Re แตกต่างกัน

Re	ค่าความเค้นวอนนิสเซสสูงสุด (kPa)	ค่าการเคลื่อนตัวสูงสุด (mm)
100	239.6	$5.65 \times 10^{-4}$
500	134.8	$3.18 \times 10^{-4}$
1,000	104.6	$2.47 \times 10^{-4}$

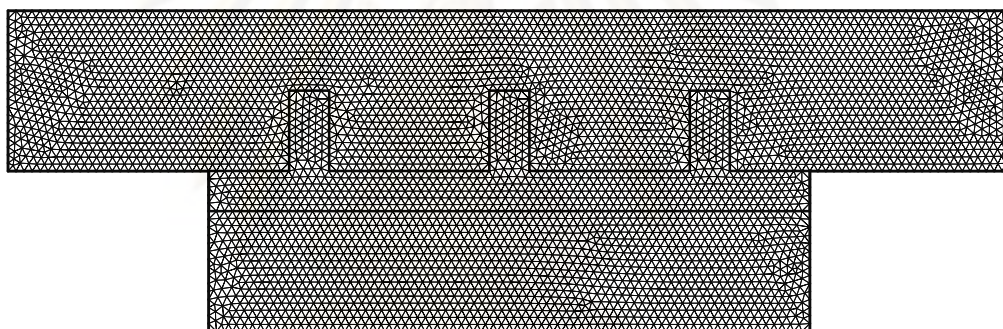
#### 6.4 ปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีครีบริบายความร้อน (free convection in channel with fins) [36]

ปัญหาสุดท้ายของหัวข้อนี้สร้างขึ้นมาเพื่อให้มีความซับซ้อนมากยิ่งขึ้น โดยเป็นการวิเคราะห์ปัญหาการพาความร้อนแบบอิสระ โดยลักษณะของปัญหาประกอบไปด้วยช่องการไหล 2 ส่วนได้แก่ด้านบนและด้านล่าง โดยชั้นกลางด้วยของแข็งที่มีครีบริบายความร้อนอยู่ สำหรับรายละเอียดของขนาดต่าง ๆ ได้แสดงในรูปที่ 6.33 โดยกำหนดให้พื้นผิวล่างสุด ( $y=-0.8$ ) มีอุณหภูมิเท่ากับ 1 และกำหนดให้พื้นผิวด้านบนสุด ( $y=0.8$ ) มีอุณหภูมิเท่ากับ 0 ส่วนขอบด้านซ้ายและขวาของช่องการไหลทั้งด้านบนและด้านล่างรวมทั้งส่วนที่เป็นของแข็งถูกกำหนดให้เป็นผนังแบบฉนวน

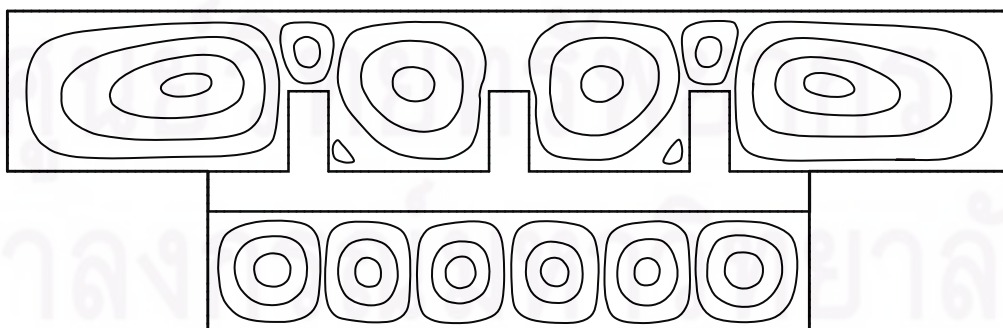


รูปที่ 6.33 ลักษณะของปัญหาการพาความร้อนแบบอิสระภายในช่องการไหล ที่มีครีบริบายความร้อน

การวิเคราะห์ปัญหาในช่วงแรกจะเริ่มจากการสร้างรูปแบบไฟไนต์เอลิเมนต์ดังแสดงในรูปที่ 6.34 ซึ่งประกอบไปด้วย 4,742 จุดต่อ 9,152 เอลิเมนต์ แล้วทำการกำหนดคุณสมบัติของของไหลและของแข็งดังนี้ กำหนดให้ของไหลมีค่า  $Pr = 0.71$  อัตราส่วนของการแพร่กระจายความร้อนระหว่างของแข็งและของไหล  $\alpha_{sf} = K_{sf}$  และค่าสัมประสิทธิ์การขยายตัวทางความร้อนของของไหล,  $\beta = 100$  และอัตราส่วนระหว่างสัมประสิทธิ์การนำความร้อนของของแข็งและของไหล,  $K_{sf}$  เท่ากับ 1, 10 และ 100 โดยรูปที่ 6.35 แสดงเส้นกระแสการไหลที่ได้จากการคำนวณที่ค่า  $K_{sf}$  เท่ากับ 10 ส่วนที่ค่า  $K_{sf}$  อื่น ๆ ผลที่ได้มีค่าไม่แตกต่างกันมากนัก ส่วนลักษณะการกระจายตัวของอุณหภูมินั้นได้แสดงในรูปที่ 6.36(ก-ค) ที่ค่า  $K_{sf}$  ต่าง ๆ หลังจากนั้นแสดงการกระจายตัวของอุณหภูมิที่บริเวณตลอดผิวด้านบนของครีระบายความร้อนทั้งสาม ที่ ณ เวลาแตกต่างกันที่ค่า  $K_{sf}$  เท่ากับ 1, 10 และ 100 ดังแสดงไว้ในรูปที่ 6.37(ก-ค) ตามลำดับ

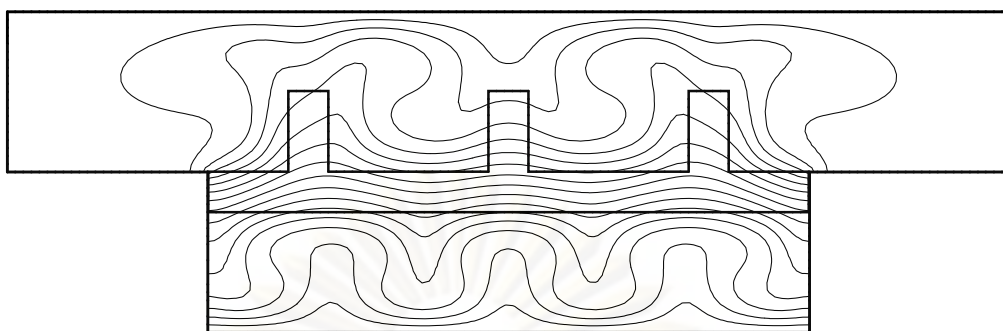


รูปที่ 6.34 รูปแบบไฟไนต์เอลิเมนต์ของปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีครีระบายความร้อน

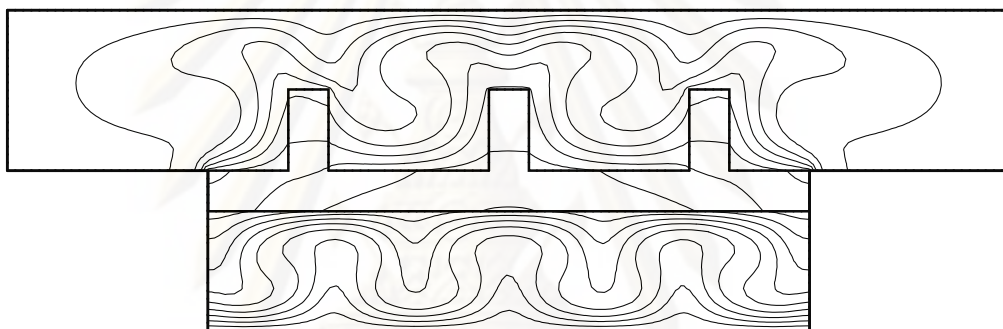


รูปที่ 6.35 เส้นกระแสการไหลของปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีครีระบายความร้อน

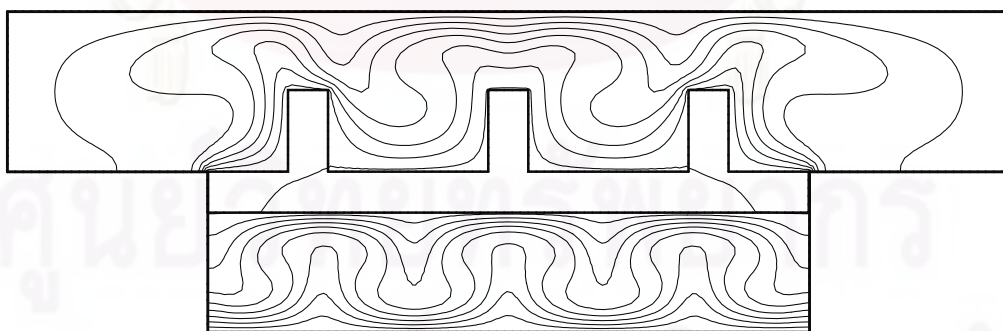




(ก) เส้นชั้นของอุณหภูมิที่ค่า  $K_{sf} = 1$

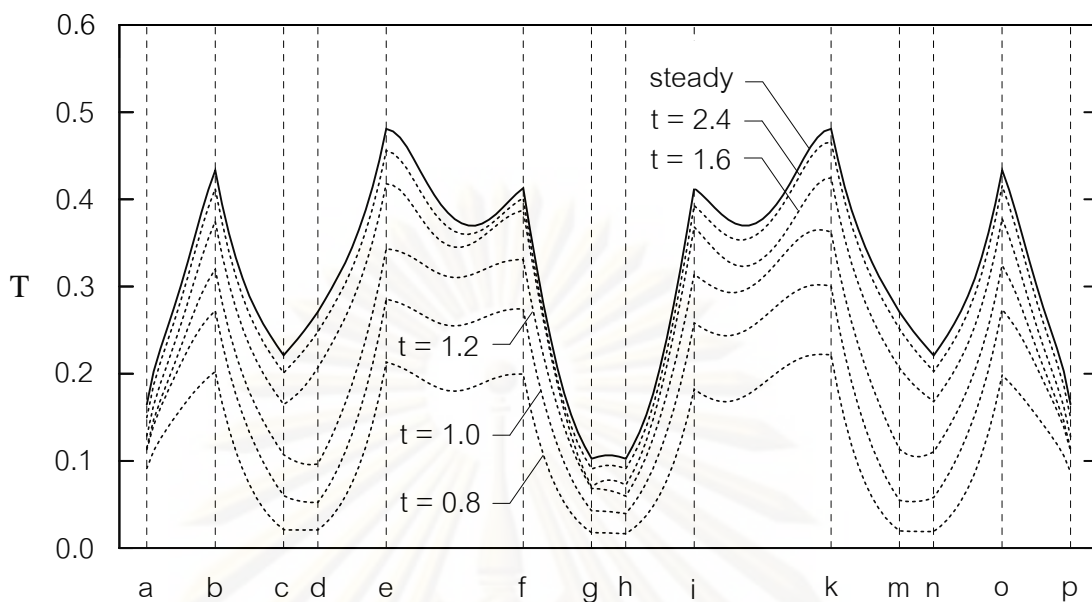


(ข) เส้นชั้นของอุณหภูมิที่ค่า  $K_{sf} = 10$

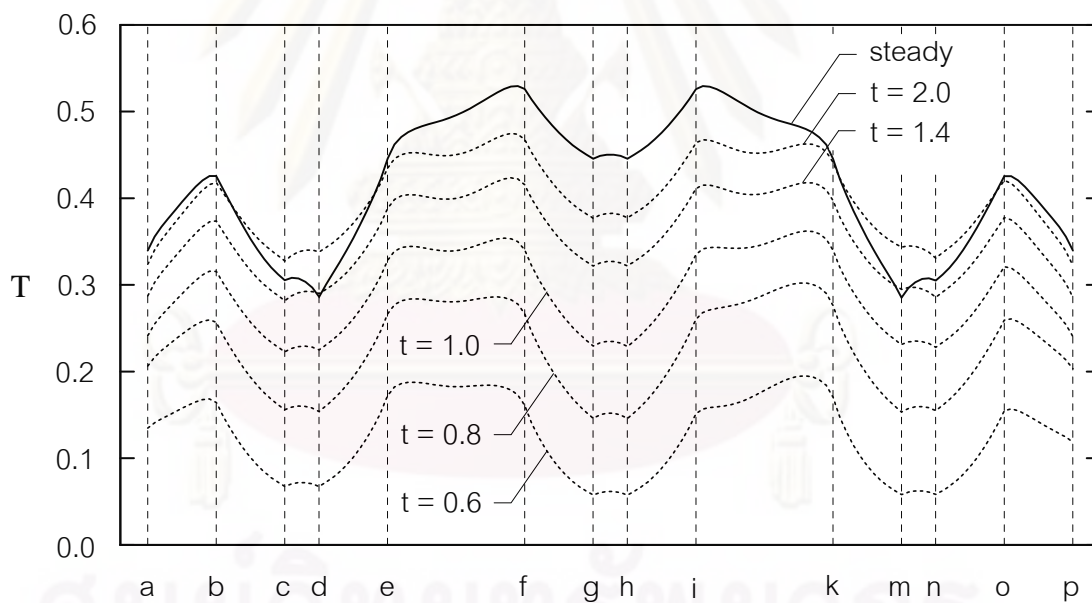


(ค) เส้นชั้นของอุณหภูมิที่ค่า  $K_{sf} = 100$

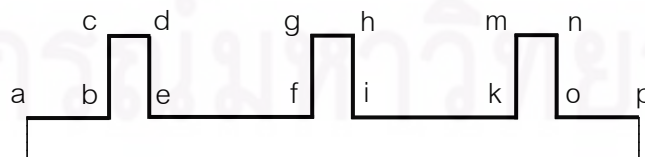
รูปที่ 6.36 เส้นชั้นของอุณหภูมิของปัญหาการพาความร้อนแบบอิสระภายในช่องการไหล  
ที่มีครีระบายความร้อนโดยที่ค่า  $K_{sf}$  ต่าง ๆ



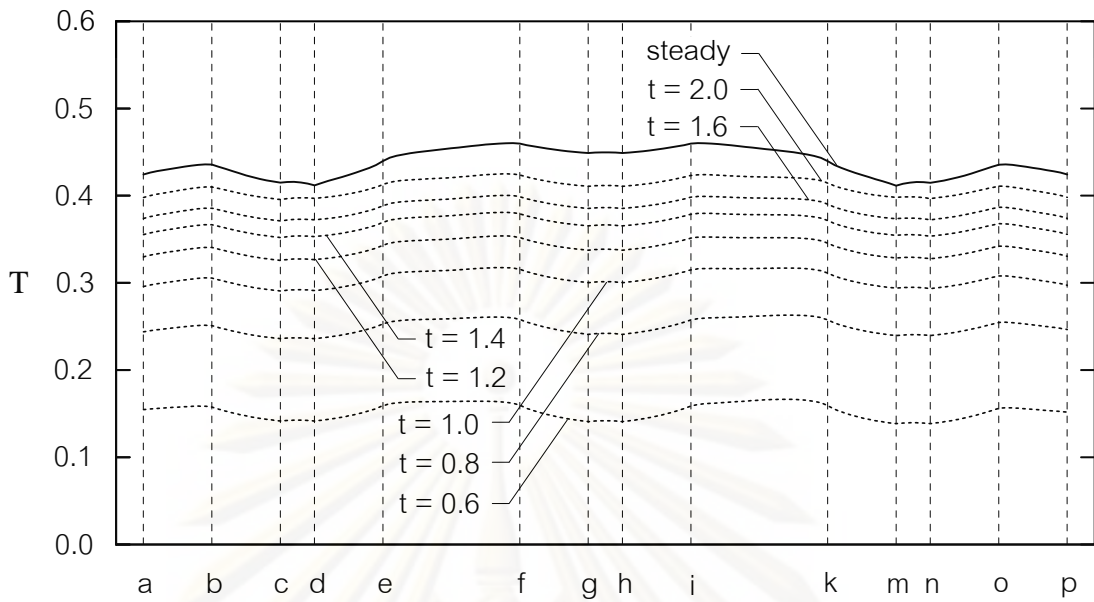
(ก) การกระจายตัวของอุณหภูมิที่ค่า  $K_{sf} = 1$



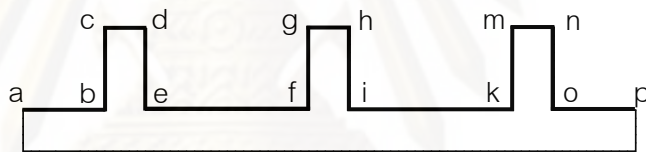
(ข) การกระจายตัวของอุณหภูมิที่ค่า  $K_{sf} = 10$



รูปที่ 6.37 เปรียบเทียบการกระจายตัวของอุณหภูมิตลอดผิวด้านบนของครีป  
ระบายความร้อนทั้งสาม เมื่อเวลาเปลี่ยนแปลงไป ที่ค่า  $K_{sf}$  ต่างต่างกัน



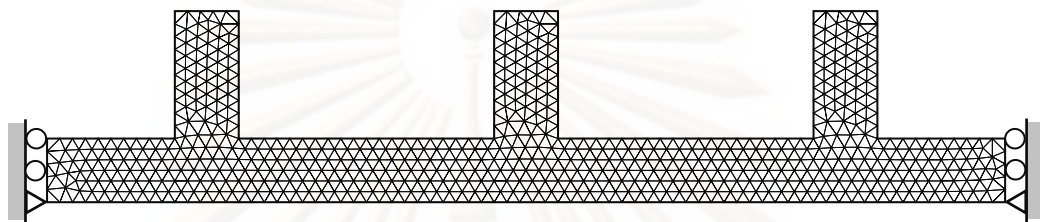
(ค) การกระจายตัวของอุณหภูมิที่ค่า  $K_{sf} = 100$



รูปที่ 6.37(ต่อ) เปรียบเทียบการกระจายตัวของอุณหภูมิตลอดผิวด้านบนของครีบริบายความร้อนทั้งสาม เมื่อเวลาเปลี่ยนแปลงไปที่ค่า  $K_{sf}$  แตกต่างกัน

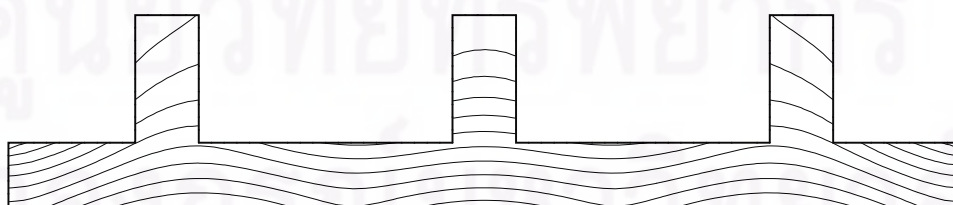
จากผลลัพธ์ที่ได้ข้างต้นนั้นจะเห็นได้ว่าปัญหาดังกล่าวมีลักษณะสมมาตรที่ตำแหน่งกึ่งกลางของปัญหา ( $x = 2.5$ ) จึงส่งผลให้ลักษณะการกระจายตัวของเส้นกระแสการไหล เส้นชั้นของอุณหภูมิ และการกระจายตัวของอุณหภูมิจึงมีลักษณะสมมาตรตามไปด้วย ในขณะที่ค่าอัตราส่วนของสัมประสิทธิ์การนำความร้อนระหว่างของแข็งและของไหล  $K_{sf}$  เปลี่ยนแปลงไป ที่ได้แสดงไว้แล้วในรูปที่ 6.35-6.37 จากนั้นเมื่อพิจารณารูป 6.37(ก-ค) พบว่าเมื่อค่าอัตราส่วนของสัมประสิทธิ์การนำความร้อนระหว่างของแข็งและของไหลคงที่ ขณะที่เวลาเปลี่ยนแปลงไปอุณหภูมิที่ผิวครีบริบายความร้อนเพิ่มสูงขึ้นเรื่อยๆ จนเริ่มเข้าสู่สภาวะอยู่ตัว (steady state) ที่ค่าหนึ่ง หนึ่งขณะเดียวกันเมื่อพิจารณาการกระจายตัวของอุณหภูมิตลอดผิวของครีบริบายความร้อนที่สภาวะอยู่ตัวในขณะที่  $K_{sf}$  เปลี่ยนแปลงไปจะเห็นได้ว่ายิ่งค่า  $K_{sf}$  ยิ่งสูงขึ้นอุณหภูมิตลอดผิวของครีบริบายความร้อนก็จะมีค่าต่ำเมื่อเทียบกับค่า  $K_{sf}$  ที่น้อยกว่า

สุดท้ายเป็นการนำผลลัพธ์ที่ได้มาคำนวณหาค่าการเคลื่อนตัวภายในค้ำระบายนความร้อนทั้ง 3 โดยรูปที่ 6.38 เป็นรูปแบบไฟไนต์เอลิเมนต์ของส่วนที่เป็นของแข็งซึ่งประกอบไปด้วย 726 จุดต่อ 1,230 เอลิเมนต์ และส่วนของเงื่อนไขขอบเขตนั้นกำหนดให้จุดต่อที่มุมด้านล่างทั้งสองฝั่งถูกตรึงให้อยู่กับที่ ส่วนจุดต่ออื่น ๆ ตลอดขอบด้านซ้ายและด้านขวากำหนดให้เคลื่อนตัวในแนวตั้งเท่านั้น

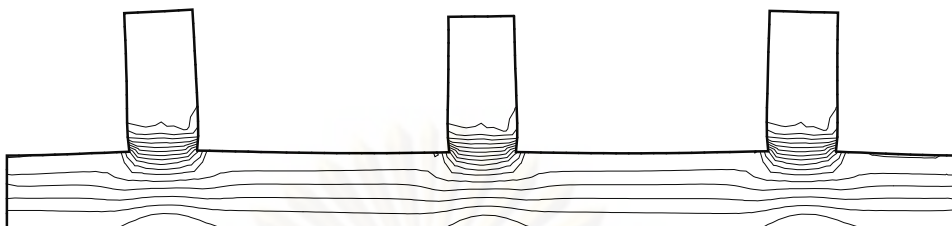


รูปที่ 6.38 รูปแบบไฟไนต์เอลิเมนต์และเงื่อนไขขอบเขตสำหรับวิเคราะห์การเคลื่อนตัวของปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีค้ำระบายนความร้อน

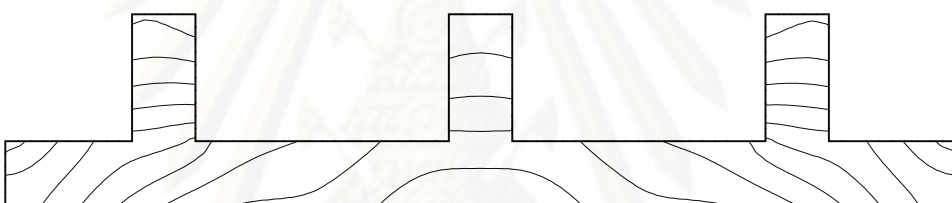
รูปที่ 6.39 แสดงลักษณะการกระจายตัวของอุณหภูมิพร้อมกับเวกเตอร์แสดงทิศทางการเคลื่อนตัวของแต่ละจุดต่อในขณะที่ค่าอัตราส่วนของสัมประสิทธิ์การนำความร้อนระหว่างของแข็งและของไหล  $K_{sf}$  มีค่าเท่ากับ 1 ในขณะที่เดียวกันรูปที่ 6.40 แสดงการกระจายตัวของค่าความเค้น von-mises ภายในของแข็งพร้อมกับค่าการเคลื่อนตัวการเสียรูปขนาด 50,000 เท่า จากนั้นทำซ้ำอีกครั้งโดยแสดงค่าต่าง ๆ ที่กล่าวมาแล้วข้างต้นแต่เพิ่มค่าอัตราส่วนของสัมประสิทธิ์การถ่ายเทความร้อนเป็น 10 และ 100 โดยสามารถแสดงได้ในรูปที่ 6.41 - 6.44 ตามลำดับ



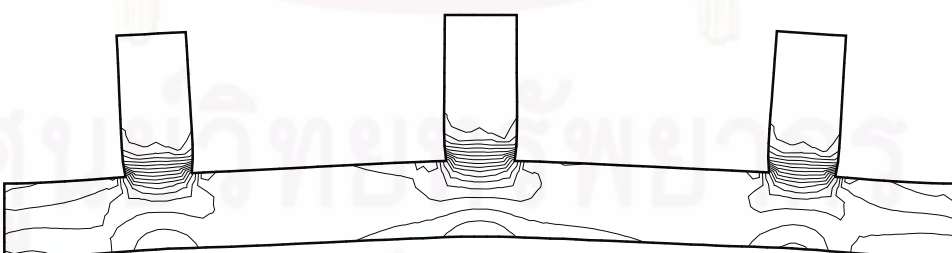
รูปที่ 6.39 แสดงลักษณะการกระจายตัวของอุณหภูมิพร้อมสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า  $K_{sf}$  เท่ากับ 1



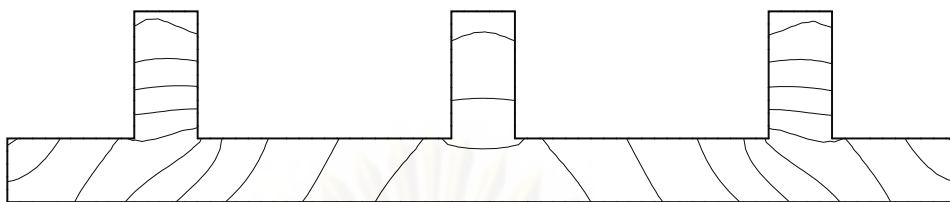
รูปที่ 6.40 แสดงการกระจายตัวของค่าความเค้นวอนมิสเซลภายในของแข็งพร้อมกับค่าการเคลื่อนตัวการเสียรูปขนาด 50,000 เท่าสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า  $K_{sf}$  เท่ากับ 1



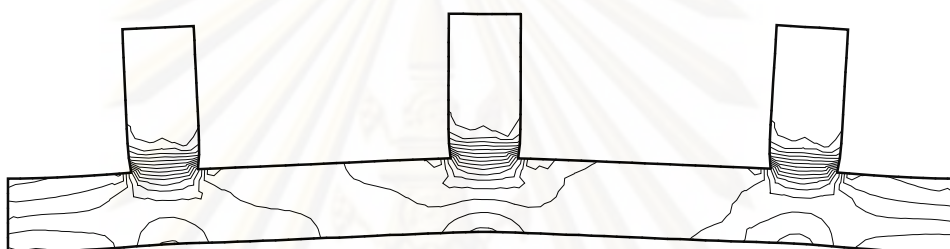
รูปที่ 6.41 แสดงลักษณะการกระจายตัวของอุณหภูมิพร้อมสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า  $K_{sf}$  เท่ากับ 10



รูปที่ 6.42 แสดงการกระจายตัวของค่าความเค้นวอนมิสเซลภายในของแข็งพร้อมกับค่าการเคลื่อนตัวการเสียรูปขนาด 50,000 เท่าสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า  $K_{sf}$  เท่ากับ 10



รูปที่ 6.43 แสดงลักษณะการกระจายตัวของอุณหภูมิพร้อมสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า  $K_{sf}$  เท่ากับ 100



รูปที่ 6.44 แสดงการกระจายตัวของค่าความเค้นวอนมิสเซสภายในของแข็งพร้อมกันค่าการเคลื่อนตัวการเสียรูปขนาด 50,000 เท่าสำหรับกรณีการพาความร้อนแบบอิสระที่ค่า  $K_{sf}$  เท่ากับ 100

นอกจากนี้สามารถแสดงค่าความเค้นวอนมิสเซสสูงสุดและค่าการเคลื่อนตัวสูงสุดได้ดังตารางที่ 6.5 จากตารางพบว่าระยะการเคลื่อนตัวมีค่าที่น้อยมาก จึงอาจกล่าวได้ว่าไม่มีผลกระทบต่อรูปแบบของการไหล

ตารางที่ 6.5 แสดงค่าความเค้นวอนมิสเซสสูงสุดและค่าการเคลื่อนตัวสูงสุดบนค้ำรับระบายความร้อนที่ค่า  $K_{sf}$  ต่างกัน

$K_{sf}$	ค่าความเค้นวอนมิสเซสสูงสุด (kPa)	ค่าการเคลื่อนตัวสูงสุด (mm)
1	278.5	$1.895 \times 10^{-3}$
5	242.6	$2.083 \times 10^{-3}$
10	240.6	$3.310 \times 10^{-3}$
100	230.8	$3.142 \times 10^{-3}$

จากตัวอย่างทั้งหมด 4 ตัวอย่างที่กล่าวมาในบทนี้ได้แสดงให้เห็นถึงการใช้ระเบียบวิธีไฟไนต์เอลิเมนต์มาประยุกต์ใช้ เพื่อการวิเคราะห์ร่วมกันระหว่างปัญหาของไหล ของแข็ง และการถ่ายเทความร้อนทั้งภายในของไหล และโครงสร้าง อีกทั้งยังสามารถหาความเค้นและระยะการเคลื่อนตัวที่เพิ่มขึ้นในของแข็ง ซึ่งเป็นการรวมองค์ความรู้ในหลาย ๆ ด้านมาทำการวิเคราะห์พร้อมกัน ทำให้สามารถเข้าใจถึงพฤติกรรมที่มีผลเกี่ยวของกันของปัญหาทั้งสามสาขาได้เป็นอย่างดี



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

## บทที่ 7

### บทสรุป ปัญหาที่พบและข้อเสนอแนะ

#### 7.1 บทสรุป

วิทยานิพนธ์นี้จะเน้นระเบียบวิธีการคำนวณแบบแยกส่วน 4 ขั้นตอนร่วมกับการใช้ฟังก์ชันถ่วงน้ำหนักด้วยวิธีสตรึมไลน์อัปวินด์ เพทรอพ-กาเลอริคิน โดยใช้เอลิเมนต์สามเหลี่ยมแบบสามจุดต่อ และใช้วิธีดังกล่าวในการวิเคราะห์ปัญหาพร้อมกันระหว่างของไหล ของแข็งและการถ่ายเทความร้อน ซึ่งในส่วนของปัญหาการไหลนั้นจะกำหนดให้เป็นการไหลแบบหนืดชนิดอัดตัวไม่ได้ในสองมิติเท่านั้น ส่วนปัญหาในของแข็งนั้นจะกำหนดให้เป็นปัญหาความเครียดระนาบ เพื่อให้สอดคล้องกับปัญหาในสองมิติ

ในการวิเคราะห์ปัญหาพร้อมกันระหว่างของไหล ของแข็งและการถ่ายเทความร้อนด้วยระเบียบวิธีไฟไนต์เอลิเมนต์จำเป็นที่จะต้องมีความรู้พื้นฐานในเรื่องของสมการเชิงอนุพันธ์ที่เกี่ยวข้อง ดังนั้นในบทที่ 2 จึงได้แสดงถึงระบบสมการเชิงอนุพันธ์สำหรับปัญหาการไหลแบบหนืดที่สภาวะไม่คงตัวในสองมิติ ซึ่งประกอบไปด้วยสมการเชิงอนุพันธ์ย่อยได้แก่ สมการอนุรักษ์มวล สมการอนุรักษ์โมเมนตัมในแนวแกน  $x$  และ  $y$  และสมการอนุรักษ์พลังงาน ตามลำดับ จากนั้นก็แสดงถึงสมการเชิงอนุพันธ์ของปัญหาการถ่ายเทความร้อน และการหาค่าความเค้นอันเนื่องมาจากอุณหภูมิภายในของแข็ง อีกทั้งอธิบายถึงขั้นตอนทั่วไปของระเบียบวิธีไฟไนต์เอลิเมนต์ แต่ในความเป็นจริงนั้นการวิเคราะห์ปัญหาการไหลโดยทั่ว ๆ ไปนั้นมีความยุ่งยากซับซ้อนเป็นอย่างมาก ซึ่งปัญหาสำคัญประการหนึ่งที่ทำให้เกิดความยุ่งยากในการวิเคราะห์ปัญหาการไหลก็คือความมีลักษณะไม่เชิงเส้นของพจน์การพาในสมการอนุรักษ์โมเมนตัม ในขณะเดียวกันต้องการศึกษาการไหลทั้งแบบที่มีทั้งสภาวะอยู่ตัวและไม่อยู่ตัว ซึ่งในบทที่ 3 จึงอธิบายถึงรายละเอียดของวิธีการคำนวณแบบแยกส่วน 4 ขั้นตอนที่สามารถนำมาใช้ในการวิเคราะห์ทั้งปัญหาที่มีทั้งสภาวะอยู่ตัวและไม่อยู่ตัว รวมทั้งใช้ฟังก์ชันถ่วงน้ำหนักด้วยวิธีสตรึมไลน์อัปวินด์ เพทรอพ-กาเลอริคินที่นิยมนำมาใช้ในการวิเคราะห์ปัญหาการไหล เพื่อลดการสั่นของคำตอบ อีกทั้งยังได้แสดงขั้นตอนในการประดิษฐ์สมการไฟไนต์เอลิเมนต์ที่สอดคล้องกับสมการเชิงอนุพันธ์ของปัญหาการไหล สมการเชิงอนุพันธ์สำหรับการถ่ายเทความร้อนทั้งภายในของไหลและโครงสร้าง และสมการเชิงอนุพันธ์สำหรับปัญหาความเค้นในโครงสร้างอันเกิดเนื่องมาจากอุณหภูมิ ในขณะเดียวกันระเบียบวิธีที่ใช้ในการวิเคราะห์ปัญหาการไหล อุณหภูมิและโครงสร้างที่นำเสนอในวิทยานิพนธ์นี้ยังสามารถ



ที่จะใช้ฟังก์ชันการประมาณภายในของความเร็ว ความดันและอุณหภูมิที่อันดับเท่ากันได้ (equal-order interpolation function) ทำให้การประดิษฐ์เอลิเมนต์เมตริกซ์และไฟไนต์เอลิเมนต์โปรแกรมคอมพิวเตอร์สามารถที่จะทำความเข้าใจได้ง่าย สำหรับเอลิเมนต์เมตริกซ์ที่ถูกสร้างขึ้นในบทที่ 3 นี้จะอยู่ในรูปแบบที่สามารถนำไปประดิษฐ์เป็นโปรแกรมคอมพิวเตอร์ได้โดยตรง

ในบทที่ 4 เป็นการอธิบายโปรแกรมคอมพิวเตอร์ที่ประดิษฐ์ขึ้น โดยนำเสนอถึงรูปแบบของไฟล์ข้อมูลเข้าที่จะต้องเตรียมเพื่อทำการคำนวณ จากนั้นโปรแกรมคอมพิวเตอร์ที่อธิบายในบทที่ 4 นี้ก็ได้รับการตรวจสอบความถูกต้องในบทที่ 5 โดยนำผลลัพธ์ที่ได้จากการวิเคราะห์ด้วยไฟไนต์เอลิเมนต์โปรแกรมคอมพิวเตอร์ไปเปรียบเทียบกับผลเฉลยแม่นยำตรง หรือปัญหาที่มีผู้ทำการคำนวณมาก่อน โดยปัญหาที่นำมาใช้ในการตรวจสอบแบ่งออกเป็น 3 ส่วนคือ ส่วนแรกเป็นการตรวจสอบความถูกต้องกับปัญหาการไหลแบบหนืดแบบไม่อัดตัวโดยมีการถ่ายเทความร้อน โดยมีตัวอย่างปัญหาที่ใช้ในการตรวจสอบ 4 ตัวอย่างด้วยกัน ปัญหาแรกเป็นปัญหาการไหลระหว่างแผ่นคู่ขนานที่สภาวะไม่อยู่ตัว ปัญหาการไหลหมุนวนในช่องสี่เหลี่ยม ปัญหาการไหลเนื่องจากการพาความร้อนแบบอิสระในช่องปิดรูปสี่เหลี่ยมจัตุรัส และปัญหาการถ่ายเทความร้อนผ่านทรงกระบอกร้อน ส่วนที่สองจะเป็นการตรวจสอบความถูกต้องกับการไหลแบบหนืดแบบไม่อัดตัวโดยมีการถ่ายเทความร้อนระหว่างของไหลผ่านโครงสร้างที่สภาวะอยู่ตัวและไม่อยู่ตัว โดยมีตัวอย่างปัญหาที่ใช้ในการตรวจสอบ 2 ตัวอย่างด้วยกัน ปัญหาแรกเป็นปัญหาการไหลระหว่างแผ่นคู่ขนานพร้อมการถ่ายเทความร้อนกับโครงสร้างและปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน ส่วนที่สามจะเป็นการตรวจสอบความถูกต้องของปัญหาความเค้นอันเนื่องมาจากอุณหภูมิโดยมีตัวอย่างปัญหาที่ใช้ในการตรวจสอบ 2 ตัวอย่างด้วยกัน ปัญหาแรกเป็นปัญหาแผ่นสี่เหลี่ยมที่มีอุณหภูมิเปลี่ยนแปลงเชิงเส้นตามความยาว และปัญหาวงแหวนที่มีอุณหภูมิเปลี่ยนแปลงตามแนวรัศมี หลังจากมีความมั่นใจในความถูกต้องของโปรแกรมที่ประดิษฐ์ขึ้นแล้ว จึงนำไปวิเคราะห์ปัญหาพร้อมกันระหว่างของไหล ของแข็งและการถ่ายเทความร้อน ดังแสดงในบทที่ 6 โดยปัญหาที่นำมาทำการวิเคราะห์มีทั้งหมด 4 ปัญหาด้วยกัน คือ ปัญหาการไหลแบบสวนทางภายในอุปกรณ์แลกเปลี่ยนความร้อน ปัญหาการพาความร้อนแบบอิสระในช่องปิดสี่เหลี่ยมจัตุรัสโดยที่มีผนังนำความร้อน ปัญหาการถ่ายเทความร้อนผ่านสิ่งกีดขวางทรงสี่เหลี่ยมภายในช่องทางการไหล และปัญหาการพาความร้อนแบบอิสระภายในช่องการไหลที่มีคิรีระบายความร้อน

กล่าวโดยสรุปได้ว่าวิทยานิพนธ์นี้สามารถนำระเบียบวิธีไฟไนต์เอลิเมนต์เพื่อวิเคราะห์ปัญหาพร้อมกันระหว่างปัญหาของไหล อุณหภูมิและโครงสร้าง ทำให้สามารถเข้าใจถึงพฤติกรรมที่มีผลเกี่ยวข้องกันของปัญหาทั้งสามสาขาได้เป็นอย่างดี

## 7.2 ปัญหาที่พบในขณะที่ทำวิทยานิพนธ์

ปัญหาสำคัญที่พบในขณะที่ทำวิทยานิพนธ์ก็คือ การทำความเข้าใจกับระเบียบวิธีการคำนวณแบบแยกส่วน 4 ขั้นตอนร่วมกับวิธีสตรึมไลน์อัลวินด์เพทรอพ-กาลเลอร์คิน เนื่องจากมีความซับซ้อนในการประดิษฐ์คอมพิวเตอร์โปรแกรม อีกปัญหาที่สำคัญอีกปัญหาหนึ่งคือการแก้ระบบสมการไฟไนต์เอลิเมนต์ในขั้นตอนที่ 1 และขั้นตอนที่ 5 ต้องใช้เวลาในการแก้ระบบสมการขนาดใหญ่ ประกอบกับผลการคำนวณที่ได้เป็นการคำนวณแบบก้าวไปตามเวลา จึงทำให้ต้องใช้เวลาในการคำนวณมาก

## 7.3 ข้อเสนอแนะสำหรับงานวิจัยในอนาคต

การพัฒนางานวิจัยต่อจากวิทยานิพนธ์นี้มีอยู่ด้วยกันหลายแนวทาง ยกตัวอย่างเช่น ในส่วนของการวิเคราะห์ปัญหาพร้อมกันระหว่างของไหล ของแข็งและการถ่ายเทความร้อนนั้น อาจจะทำกรวิเคราะห์โดยสมมติให้ของแข็งเกิดการเสียรูปมากขึ้นจนส่งผลต่อเนื่องไปถึงพฤติกรรมการไหล ทำให้เกิดการปฏิสัมพันธ์กันอย่างครบถ้วนระหว่างของไหล ของแข็งและการถ่ายเทความร้อน อาจพัฒนาโปรแกรมคอมพิวเตอร์ให้สามารถวิเคราะห์ปัญหาไหลและการถ่ายเทความร้อนในสภาวะปั่นป่วน (Turbulent Problem) ได้ สุดท้ายถ้ามีการปรับปรุงวิธีในการแก้ระบบสมการขนาดใหญ่ให้ใช้เวลาในการคำนวณน้อยลงจะทำให้สามารถศึกษาและวิเคราะห์ปัญหาการไหลที่มีขนาดใหญ่และมีความซับซ้อนได้เพิ่มมากขึ้น

## รายการอ้างอิง

- [1] Rice, J.G. and Schnipke, R.J. An equal-order velocity-pressure formulation that does not exhibit spurious pressure modes. Computer Methods in Applied Mechanics and Engineering 58 (1986): 135-149.
- [2] Choi, H.G. and Yoo, J.Y. Streamline upwind scheme for the segregated formulation of the Navier-Stokes equation. Numerical Heat Transfer, Part B 25 (1994): 145-161.
- [3] Wansophark, N. and Dechaumphai, P. Combined adaptive meshing technique and segregated finite element algorithm for analysis of free and forced convection heat transfer. Finite Elements in Analysis and Design 40 (2004): 645-663.
- [4] Malatip, A., Wansophark, N. and Dechaumphai, P. Combined streamline upwind Petrov–Galerkin method and segregated finite element algorithm for conjugate heat transfer problems. Journal of Mechanical Science and Technology 20 (2006): 1741-1752.
- [5] Patankar, S.V. and Spalding, D.B. A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-Dimensional Parabolic Flows. International Journal of Heat and Mass Transfer 15 (1972)
- [6] Chorin, A.J. Numerical solution of the Navier-Stokes equations. Mathematics of Computation 22 (1968): 745-762.
- [7] Choi, H.G., Choi, H. and Yoo J.Y. A fractional four-step finite element formulation of the unsteady incompressible Navier-Stokes equations using SUPG and linear equal-order element methods. Computer Methods in Applied Mechanics and Engineering 143 (1996): 333-348.
- [8] Kjellgren, P. A semi-implicit fractional step finite element method for viscous incompressible flows. Computational Mechanics 20 (1997): 541-550.
- [9] Ferziger, J. H. and Peric, M. Computational Methods for Fluid Dynamics. Tokyo: Springer, 1999.

- [10] Sung, J., Choi, H.G. and Yoo, J.Y. Time-accurate computation of unsteady free surface flows using an ALE-segregated equal-order FEM. Computer Methods in Applied Mechanics and Engineering 190 (2000): 1425-1440.
- [11] Codina, R. and Badia, S. On some pressure segregation method of fractional-step type for the finite element approximation of incompressible flow problems. Computer Methods in Applied Mechanics and Engineering 195 (2006): 2900-2918.
- [12] Brooks, A.N. and Hughes, T.J.R. Streamlines upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. Computer Methods in Applied Mechanics and Engineering 32 (1982): 199-259.
- [13] Chorin, A.J. On Convergence of Discrete Approximation to the Navier-Stokes Equations. Mathematics of Computation 23 (1969): 341-353.
- [14] Zienkiewicz, O.C. and Codina, R. A General Algorithm for Compressible and Incompressible Flow-Part I. The Split, Characteristic-Based Scheme. International Journal for Numerical Methods in Fluids 20 (1995): 869-885.
- [15] Massarotti, N., Arpino, F., Lewis, R.W. and Nithiarasu, P. Explicit and semi-implicit CBS procedures for incompressible viscous flows. International Journal for Numerical Methods in Engineering. 66 (2006): 1618-1640.
- [16] Wansophark, N., Malatip, A. and Dechaumphai, P. Streamline upwind finite element method for conjugate heat transfer problems. Acta Mechanica Sinica Published online: 26 (2005).
- [17] Sai, B.V.K.S., Seetharamu, K.N. and Narayana, P.A.A. Solution of transient laminar natural convection in a square cavity by an explicit finite element scheme. Numerical Heat Transfer, Part A 25 (1994): 593-609.
- [18] Anderson, J.D. Computational fluid dynamics. Singapore: McGraw-Hill, 1995.
- [19] Incropera, F.P. and De Witt, D.P. Fundamentals of heat and mass transfer. 4<sup>th</sup> ed. Singapore: John Wiley & Sons, 1996.
- [20] Timoshenko, S.P. and Goodier, J.N. Theory of elasticity. 3<sup>rd</sup> ed. Singapore: McGraw-Hill, 1970.

- [21] ปราโมทย์ เดชะอำไพ. ไฟไนต์เอลิเมนต์ในงานวิศวกรรม. พิมพ์ครั้งที่ 4. กรุงเทพมหานคร: สำนักพิมพ์จุฬาลงกรณ์มหาวิทยาลัย, 2550.
- [22] Zienkiewicz, O.C., Taylor, R.L. and Nithiarasu, P. The Finite Element Method for Fluid Dynamics. Sixth Edition, Butterworth-Heinemann, 2005.
- [23] Choi, H.G., and Moin, P. Effect of the computational time step on numerical solutions of turbulent flow. Journal of Computational Physics 113 (1994): 1-4.
- [24] White, F.M. Viscous fluid flow. 2<sup>nd</sup> ed. New York: McGraw-Hill, 1991.
- [25] Yagawa, G., and Shirazaki, M. Parallel Computing for Incompressible Flow using a Nodal-Based Method. Computational Mechanics 23 (1999): 209-217.
- [26] Ghia, U., Ghia, K.N., and Shin, C.T. High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method. Journal of Computational Physics 48 (1982): 387-411.
- [27] de Vahl Davis, G. Natural Convection of Air in a Square Cavity: A Benchmark Numerical Solution. International Journal for Numerical Methods in Fluids 3 (1983): 249-264.
- [28] Leal, M.A., Machado, H.A., and Cotta, R.M. Integral Transform Solutions of Transient Natural Convection in Enclosures with Variable Fluid Properties. International Journal of Heat and Mass Transfer 43 (2000): 3977-3990.
- [29] Malan, A.G., Lewis, R.W., and Nithiarasu, P. An Improved Unsteady, Unstructured, Artificial Compressibility, finite Volume Scheme for Viscous Incompressible Flows: Part II. Application. International Journal for Numerical Methods in Engineering 54 (2002): 715-729.
- [30] de Sampaio, P.A.B., Lyra, P.R.M., Morgan, K., and Weatherill, N.P. Petrov-Galerkin Solutions of the Incompressible Navier-Stokes Equations in Primitive Variables with Adaptive Remeshing. Computer Methods in Applied Mechanics and Engineering 106 (1993): 143-178.
- [31] Yoon, H.S., Lee, J.B., and Chun, H.H. A numerical study on the fluid flow and heat transfer around a circular cylinder near a moving wall. International Journal of Heat and Mass Transfer 50 (2007): 3507-3520.

- [32] Chen, X. and Han, P. A note on the solution of conjugate heat transfer problems using SIMPLE-like algorithms. International Journal of Heat and Fluid Flow 21 (2000): 463-467.
- [33] วิโรจน์ ลิ่มตระการ. ระเบียบวิธีไฟไนต์เอลิเมนต์สำหรับปฏิสัมพันธ์ระหว่างการไหลความเร็วสูงและโครงสร้าง. วิทยานิพนธ์ปริญญาดุษฎีบัณฑิต, ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2545.
- [34] Hriberšek, M. and Kuhn, G. Conjugate heat transfer by boundary-domain integral method. Engineering Analysis with Boundary Element 24 (2000): 297-305.
- [35] Davalath, J. and Bayazitoglu, Y. Forced convection cooling across rectangular blocks. Journal of Heat Transfer 109 (1987): 321-328.
- [36] นิพนธ์ วรรณโสภาคย์. ระเบียบวิธีสตรึมไลน์อัปวินด์ไฟไนต์เอลิเมนต์และการปรับขนาดสำหรับการวิเคราะห์ห้วงกันระหว่างการไหล อุณหภูมิและโครงสร้าง. วิทยานิพนธ์ปริญญาดุษฎีบัณฑิต, ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2549.



ภาคผนวก

ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย





```

REAL(8)                                :: absu, peclet, alp

uold = u
rxele = 0.0D0
ryele = 0.0D0
sysrx = 0.0D0
sysry = 0.0D0
sysk = 0.0D0
axele = 0.0D0
ayele = 0.0D0

!-----
! Loop over the number of elements
!-----

DO ie = 1,nelem

!-----
! Find element local coordinates:
!-----

ii = intmat(ie,1)
jj = intmat(ie,2)
kk = intmat(ie,3)

xg1 = coord(ii,1)
xg2 = coord(jj,1)
xg3 = coord(kk,1)

yg1 = coord(ii,2)
yg2 = coord(jj,2)
yg3 = coord(kk,2)

area = 0.50D0*(xg2*(yg3-yg1)+xg1*(yg2-yg3)+xg3*(yg1-yg2))

IF(area <= 0.) WRITE(6,5) ie
5 FORMAT(/,'   !!! ERROR !!! ELEMENT NO.', I5,           &
         ' HAS NEGATIVE OR ZERO area ', /,             &
         ' --- check F.E. MODEL FOR NODAL COORDINATES', &
         ' AND ELEMENT NODAL CONNECTIONS ---'         )
IF(area <= 0.) STOP

b1 = yg2 - yg3
b2 = yg3 - yg1
b3 = yg1 - yg2

c1 = xg3 - xg2
c2 = xg1 - xg3
c3 = xg2 - xg1

!-----
! Setup Uba , Vba
!-----

uba = (u(ii)+u(jj)+u(kk))/3.0D0
vba = (v(ii)+v(jj)+v(kk))/3.0D0

!-----
! Element Size
!-----

IF(weight == 1) THEN
  absu = SQRT((uba*uba) + (vba*vba))
  peclet = absu*hele(ie)*den/(2.0D0*vis)
  alp = 1.0D0/(DTANH(peclet)) - 1.0D0/peclet
  dte(ie) = alp*hele(ie)/absu
  if(peclet == 0.0D0 .or. dte(ie) >= dt(ie)) dte(ie) = dt(ie)
ENDIF

```

```

=====
! Setup [M] matrix: (Mass term)
=====

Mele(1,1) = area/6.0D0
Mele(1,2) = area/12.0D0
Mele(1,3) = area/12.0D0

Mele(2,1) = area/12.0D0
Mele(2,2) = area/6.0D0
Mele(2,3) = area/12.0D0

Mele(3,1) = area/12.0D0
Mele(3,2) = area/12.0D0
Mele(3,3) = area/6.0D0

! Setup [C] matrix: (Convection term)
=====

!-----
! Setup [Cn] matrix
!-----

canu = uba/6.0D0
canv = vba/6.0D0

Cn(1,1) = canu*b1 + canv*c1
Cn(1,2) = canu*b2 + canv*c2
Cn(1,3) = canu*b3 + canv*c3

Cn(2,1) = Cn(1,1)
Cn(2,2) = Cn(1,2)
Cn(2,3) = Cn(1,3)

Cn(3,1) = Cn(1,1)
Cn(3,2) = Cn(1,2)
Cn(3,3) = Cn(1,3)

!-----
! Setup [Cw] matrix
!-----

caw1 = uba*uba/(8.0D0*area)
caw23 = uba*vba/(8.0D0*area)
caw4 = vba*vba/(8.0D0*area)

Cw(1,1) = caw1*b1*b1 + caw23*(b1*c1 + b1*c1) + caw4*c1*c1
Cw(1,2) = caw1*b1*b2 + caw23*(b1*c2 + b2*c1) + caw4*c1*c2
Cw(1,3) = caw1*b1*b3 + caw23*(b1*c3 + b3*c1) + caw4*c1*c3

Cw(2,1) = Cw(1,2)
Cw(2,2) = caw1*b2*b2 + caw23*(b2*c2 + b2*c2) + caw4*c2*c2
Cw(2,3) = caw1*b2*b3 + caw23*(b2*c3 + b3*c2) + caw4*c2*c3

Cw(3,1) = Cw(1,3)
Cw(3,2) = Cw(2,3)
Cw(3,3) = caw1*b3*b3 + caw23*(b3*c3 + b3*c3) + caw4*c3*c3

!-----
! Setup [C] matrix : [Cn] + [Cw]
!-----

Cele = Cn + dte(ie)*Cw

! Setup [K] matrix: (Diffusion term)
=====

IF(ltype(ie,1) == 0) then          ! For Fluid

Kele(1,1) = ((b1*b1)+(c1*c1))*vis/(4.0D0*area*den)
Kele(1,2) = ((b1*b2)+(c1*c2))*vis/(4.0D0*area*den)

```

```

Kele(1,3) = ((b1*b3)+(c1*c3))*vis/(4.0D0*area*den)

Kele(2,2) = ((b2*b2)+(c2*c2))*vis/(4.0D0*area*den)
Kele(2,3) = ((b2*b3)+(c2*c3))*vis/(4.0D0*area*den)
Kele(3,3) = ((b3*b3)+(c3*c3))*vis/(4.0D0*area*den)

Kele(2,1) = Kele(1,2)
Kele(3,1) = Kele(1,3)
Kele(3,2) = Kele(2,3)

ENDIF

IF(ltype(ie,1) == 1) then          ! For Solid

Kele = 0.0D0

ENDIF

!=====
! Setup [rpx] matrix:
!=====

rpnx(1) = -(b1*p(ii) + b2*p(jj) + b3*p(kk))/(6.0D0*den)
rpnx(2) = rpnx(1)
rpnx(3) = rpnx(1)

rpwx(1) = -(b1*uba + c1*vba)*(b1*p(ii) + b2*p(jj) + b3*p(kk))/(8.0D0*den*area)
rpwx(2) = -(b2*uba + c2*vba)*(b1*p(ii) + b2*p(jj) + b3*p(kk))/(8.0D0*den*area)
rpwx(3) = -(b3*uba + c3*vba)*(b1*p(ii) + b2*p(jj) + b3*p(kk))/(8.0D0*den*area)

rpx = rpnx + dte(ie)*rpwx

!rpx = 0.0D0

!=====
! Setup [rgx] matrix:
!=====

!coefn    = 0.0D0
!coefw    = 0.0D0

!rgnx(1) = coefn*(expan*( 2.0D0*t(ii) + t(jj) + t(kk)) - 4.0D0*(1. + (expan*tref)))
!rgnx(2) = coefn*(expan*(t(ii) + 2.0D0*t(jj) + t(kk) ) - 4.0D0*(1. + (expan*tref)))
!rgnx(3) = coefn*(expan*(t(ii) + t(jj) + 2.0D0*t(kk) ) - 4.0D0*(1. + (expan*tref)))

!rgwx(1) = coefw*(b1*uba + c1*vba)/2.0D0
!rgwx(2) = coefw*(b2*uba + c2*vba)/2.0D0
!rgwx(3) = coefw*(b3*uba + c3*vba)/2.0D0

!rgx = rgnx + dte(ie)*rgwx

rgx = 0.0D0

!=====
! Setup [rbu] matrix
!=====

rbu = 0.0D0
rb1 = 0.0D0
rb2 = 0.0D0

!-----
! Check boundary
!-----

DO i = 1,nbou

IF(ie == intbou(i,3)) THEN

    bu = b1*u(ii) + b2*u(jj) + b3*u(kk)
    cu = c1*u(ii) + c2*u(jj) + c3*u(kk)

```

```

!-----
! To apply boundary conditions
!           Inflow      => edge = 1
!           Outflow     => edge = 2
!           Symmetry     => edge = 4
!-----

IF(intbou(i,4) == 1 .or. 2 .or. 4) THEN

    ib = intbou(i,1)
    jb = intbou(i,2)

    xb1 = coord(ib,1)
    xb2 = coord(jb,1)

    yb1 = coord(ib,2)
    yb2 = coord(jb,2)

    unx = yb2 - yb1
    uny = xb1 - xb2

    IF(ib == ii) THEN
        rbl(1) = bu*unx + cu*uny
        rbl(2) = bu*unx + cu*uny
        rbl(3) = 0.0D0
    ENDIF

    IF(ib == jj) THEN
        rbl(1) = 0.0D0
        rbl(2) = bu*unx + cu*uny
        rbl(3) = bu*unx + cu*uny
    ENDIF

    IF(ib == kk) THEN
        rbl(1) = bu*unx + cu*uny
        rbl(2) = 0.0D0
        rbl(3) = bu*unx + cu*uny
    ENDIF

ENDIF

!-----
! To apply wall boundary conditions
!           Wall => edge = 5
!-----

IF(intbou(i,4) == 5) THEN

    ib = intbou(i,1)
    jb = intbou(i,2)

    xb1 = coord(ib,1)
    xb2 = coord(jb,1)

    yb1 = coord(ib,2)
    yb2 = coord(jb,2)

    unx = yb2 - yb1
    uny = xb1 - xb2

    IF(ib == ii) THEN
        rb2(1) = bu*unx + cu*uny
        rb2(2) = bu*unx + cu*uny
        rb2(3) = 0.0D0
    ENDIF

    IF(ib == jj) THEN
        rb2(1) = 0.0D0
        rb2(2) = bu*unx + cu*uny
        rb2(3) = bu*unx + cu*uny
    ENDIF

    IF(ib == kk) THEN

```

```

        rb2(1) = bu*unx + cu*uny
        rb2(2) = 0.0D0
        rb2(3) = bu*unx + cu*uny
    ENDIF

ENDIF

rbu = rb1 + rb2

! For Fluid
    IF(ltype(ie,1) == 0) then
        rbu = vis*rbu/(4.0D0*area*den)
    ENDIF

! For Solid
    IF(ltype(ie,1) == 1) then
        rbu = 0.0D0
    ENDIF

ENDIF

ENDDO

!=====
! Setup matrix : [Am]{x}* = [An]{x} + {R}
!=====

Aelem = Mele/dt(ie) + (Cele + Kele)/2.0D0
Aelen = Mele/dt(ie) - (Cele + Kele)/2.0D0

axele = Aelem

!-----
! Setup {Rau}
!-----

rau(1) = Aelen(1,1)*u(ii) + Aelen(1,2)*u(jj) + Aelen(1,3)*u(kk)
rau(2) = Aelen(2,1)*u(ii) + Aelen(2,2)*u(jj) + Aelen(2,3)*u(kk)
rau(3) = Aelen(3,1)*u(ii) + Aelen(3,2)*u(jj) + Aelen(3,3)*u(kk)

!-----
! Sum term in x:
!-----

rxele = rau + rpx + rgx + rbu

!-----
!ASSMBLE() ! Subroutine for assembling the element equations
!-----

DO i = 1,3
    DO j = 1,3
        ii = intmat(ie,i)
        jj = intmat(ie,j)
        sysk(ii,jj) = sysk(ii,jj) + axele(i,j)
    END DO
END DO

DO i = 1,3
    ii = intmat(ie,i)
    sysrx(ii) = sysrx(ii) + rxele(i)
END DO

!-----

END DO

```

```

!-----
CALL APPLYBC(ibcu, u, sysrx)

CALL unsympcg(sysk, sysrx, u, npoi)

!-----

END SUBROUTINE STEP1_U

!=====

SUBROUTINE STEP1_V() ! Subroutine for solve u* & v* -velocities ; Step 1

USE VARIABLE
USE SOLVE

IMPLICIT NONE

REAL(8), DIMENSION(3)      :: rpy, rpny, rpwy, rpx, rpnx, rpwx
REAL(8), DIMENSION(3)      :: rgy, rgny, rgwy, rgx, rgnx, rgwx
REAL(8), DIMENSION(3)      :: rau, rav, rbu, rbv, Keu, Kev
REAL(8), DIMENSION(3)      :: rb1, rb2
REAL(8), DIMENSION(3,3)    :: axele, ayele
REAL(8), DIMENSION(3,3)    :: Mele, Cele, Cn, Cw, Kele, Aelem, Aelen
REAL(8)                     :: sum, coefn, coefw
REAL(8)                     :: canu, canv, caw1, caw23, caw4
REAL(8)                     :: uba, vba, ubao, vbao
REAL(8)                     :: canuo, canvo, caw1o, caw23o, caw4o
REAL(8)                     :: unx, uny, xb1, xb2, yb1, yb2
REAL(8)                     :: absu, peclet, alp
integer(4)                  :: d, rc
integer, DIMENSION(npoi)    :: INDX !integer vector (n)

void = v
rxele = 0.0D0
ryele = 0.0D0
sysrx = 0.0D0
sysry = 0.0D0
sysk = 0.0D0
axele = 0.0D0
ayele = 0.0D0

!-----
! Loop over the number of elements
!-----

DO ie = 1,nelem

!-----
! Find element local coordinates:
!-----

ii = intmat(ie,1)
jj = intmat(ie,2)
kk = intmat(ie,3)

xg1 = coord(ii,1)
xg2 = coord(jj,1)
xg3 = coord(kk,1)

yg1 = coord(ii,2)
yg2 = coord(jj,2)
yg3 = coord(kk,2)

area = 0.5D0*(xg2*(yg3-yg1)+xg1*(yg2-yg3)+xg3*(yg1-yg2))

IF(area <= 0.) WRITE(6,5) ie
5 FORMAT(/,'   !!! ERROR !!! ELEMENT NO.', I5,           &
' HAS NEGATIVE OR ZERO area ', /,                      &
' --- check F.E. MODEL FOR NODAL COORDINATES', &
' AND ELEMENT NODAL CONNECTIONS ---' )
IF(area <= 0.) STOP

```

```

b1 = yg2 - yg3
b2 = yg3 - yg1
b3 = yg1 - yg2

c1 = xg3 - xg2
c2 = xg1 - xg3
c3 = xg2 - xg1

!-----
! Setup Uba , Vba
!-----

uba = (uold(ii)+uold(jj)+uold(kk))/3.0D0
vba = (v(ii)+v(jj)+v(kk))/3.0D0

!=====
! Setup [M] matrix: (Mass term)
!=====

Mele(1,1) = area/6.0D0
Mele(1,2) = area/12.0D0
Mele(1,3) = area/12.0D0

Mele(2,1) = area/12.0D0
Mele(2,2) = area/6.0D0
Mele(2,3) = area/12.0D0

Mele(3,1) = area/12.0D0
Mele(3,2) = area/12.0D0
Mele(3,3) = area/6.0D0

!=====
! Setup [C] matrix: (Convection term)
!=====

!-----
! Setup [Cn] matrix
!-----

canu = uba/6.0D0
canv = vba/6.0D0

Cn(1,1) = canu*b1 + canv*c1
Cn(1,2) = canu*b2 + canv*c2
Cn(1,3) = canu*b3 + canv*c3

Cn(2,1) = Cn(1,1)
Cn(2,2) = Cn(1,2)
Cn(2,3) = Cn(1,3)

Cn(3,1) = Cn(1,1)
Cn(3,2) = Cn(1,2)
Cn(3,3) = Cn(1,3)

!-----
! Setup [Cw] matrix
!-----

caw1 = uba*uba/(8.0D0*area)
caw23 = uba*vba/(8.0D0*area)
caw4 = vba*vba/(8.0D0*area)

Cw(1,1) = caw1*b1*b1 + caw23*(b1*c1 + b1*c1) + caw4*c1*c1
Cw(1,2) = caw1*b1*b2 + caw23*(b1*c2 + b2*c1) + caw4*c1*c2
Cw(1,3) = caw1*b1*b3 + caw23*(b1*c3 + b3*c1) + caw4*c1*c3

Cw(2,1) = Cw(1,2)
Cw(2,2) = caw1*b2*b2 + caw23*(b2*c2 + b2*c2) + caw4*c2*c2
Cw(2,3) = caw1*b2*b3 + caw23*(b2*c3 + b3*c2) + caw4*c2*c3

Cw(3,1) = Cw(1,3)
Cw(3,2) = Cw(2,3)
Cw(3,3) = caw1*b3*b3 + caw23*(b3*c3 + b3*c3) + caw4*c3*c3

```

```

!-----
! Setup [C] matrix : [Cn] + [Cw]
!-----

Cele = Cn + dte(ie)*Cw/2.0D0

!=====
! Setup [K] matrix: (Diffusion term)
!=====

IF(ltype(ie,1) == 0) then          ! For Fluid

Kele(1,1) = ((b1*b1)+(c1*c1))*vis/(4.0D0*area*den)
Kele(1,2) = ((b1*b2)+(c1*c2))*vis/(4.0D0*area*den)
Kele(1,3) = ((b1*b3)+(c1*c3))*vis/(4.0D0*area*den)

Kele(2,2) = ((b2*b2)+(c2*c2))*vis/(4.0D0*area*den)
Kele(2,3) = ((b2*b3)+(c2*c3))*vis/(4.0D0*area*den)
Kele(3,3) = ((b3*b3)+(c3*c3))*vis/(4.0D0*area*den)

Kele(2,1) = Kele(1,2)
Kele(3,1) = Kele(1,3)
Kele(3,2) = Kele(2,3)

ENDIF

IF(ltype(ie,1) == 1) then          ! For Solid

Kele = 0.0D0

ENDIF

!=====
! Setup [rpy] matrix:
!=====

rpny(1) = -(c1*p(ii) + c2*p(jj) + c3*p(kk))/(6.0D0*den)
rpny(2) = rpny(1)
rpny(3) = rpny(1)

rpwy(1) = -(b1*uba + c1*vba)*(c1*p(ii) + c2*p(jj) + c3*p(kk))/(8.0D0*den*area)
rpwy(2) = -(b2*uba + c2*vba)*(c1*p(ii) + c2*p(jj) + c3*p(kk))/(8.0D0*den*area)
rpwy(3) = -(b3*uba + c3*vba)*(c1*p(ii) + c2*p(jj) + c3*p(kk))/(8.0D0*den*area)

rpy = rpny + dte(ie)*rpwy/2.0D0

!rpy = 0.0D0

!=====
! Setup [rgy] matrix:
!=====

coefn  = 10.0D0*area/12.0D0
coefw  = 10.0D0*(expan*(t(ii) + t(jj) + t(kk)) - 3.0D0*(1.0D0 + expan*tref))/12.0D0

rgny(1) = coefn*(expan*( 2.0D0*t(ii) + t(jj) + t(kk) ) - 4.0D0*(1.0D0 + (expan*tref)))
rgny(2) = coefn*(expan*( t(ii) + 2.0D0*t(jj) + t(kk) ) - 4.0D0*(1.0D0 + (expan*tref)))
rgny(3) = coefn*(expan*( t(ii) + t(jj) + 2.0D0*t(kk) ) - 4.0D0*(1.0D0 + (expan*tref)))

rgwy(1) = coefw*(b1*uba + c1*vba)
rgwy(2) = coefw*(b2*uba + c2*vba)
rgwy(3) = coefw*(b3*uba + c3*vba)

rgy = rgny + dte(ie)*rgwy/2.0D0

!rgy = 0.0D0

!-----

IF(ltype(ie,1) == 1) then          ! For Solid

```



```

rgy = 0.0D0

ENDIF

!-----

!=====
! Setup [rbv] matrix
!=====

rbv = 0.0D0
rb1 = 0.0D0
rb2 = 0.0D0

!-----
! Check boundary
!-----

DO i = 1,nbou
  IF(ie == intbou(i,3)) THEN

      bv = b1*v(ii) + b2*v(jj) + b3*v(kk)
      cv = c1*v(ii) + c2*v(jj) + c3*v(kk)

!-----
! To apply boundary conditions
!           Inflow           => edge = 1
!           Outflow          => edge = 2
!           Symmetry         => edge = 4
!-----

  IF(intbou(i,4) == 1 .or. 2 .or. 4) THEN

      ib = intbou(i,1)
      jb = intbou(i,2)

      xb1 = coord(ib,1)
      xb2 = coord(jb,1)

      yb1 = coord(ib,2)
      yb2 = coord(jb,2)

      unx = yb2 - yb1
      uny = xb1 - xb2

      IF(ib == ii) THEN
          rbl(1) = bv*unx + cv*uny
          rbl(2) = bv*unx + cv*uny
          rbl(3) = 0.0D0
      ENDIF

      IF(ib == jj) THEN
          rbl(1) = 0.0D0
          rbl(2) = bv*unx + cv*uny
          rbl(3) = bv*unx + cv*uny
      ENDIF

      IF(ib == kk) THEN
          rbl(1) = bv*unx + cv*uny
          rbl(2) = 0.0D0
          rbl(3) = bv*unx + cv*uny
      ENDIF

  ENDIF

  IF(intbou(i,4) == 5) THEN

!-----
! To apply wall boundary conditions
!           Wall => edge = 5
!-----


```

```

      ib = intbou(i,1)
      jb = intbou(i,2)

      xb1 = coord(ib,1)
      xb2 = coord(jb,1)

      yb1 = coord(ib,2)
      yb2 = coord(jb,2)

      unx = yb2 - yb1
      uny = xb1 - xb2

      IF(ib == ii) THEN
        rb2(1) = bv*unx + cv*uny
        rb2(2) = bv*unx + cv*uny
        rb2(3) = 0.0D0
      ENDIF

      IF(ib == jj) THEN
        rb2(1) = 0.0D0
        rb2(2) = bv*unx + cv*uny
        rb2(3) = bv*unx + cv*uny
      ENDIF

      IF(ib == kk) THEN
        rb2(1) = bv*unx + cv*uny
        rb2(2) = 0.0D0
        rb2(3) = bv*unx + cv*uny
      ENDIF

!rb2 = 0.0D0

      ENDIF

      rbv = rb1 + rb2

      ! For Fluid
        IF(ltype(ie,1) == 0) then
          rbv = vis*rbv/(4.0D0*area*den)
        ENDIF

      ! For Solid
        IF(ltype(ie,1) == 1) then
          ENDIF

      ENDIF

      ENDDO

      !=====
      ! Setup matrix : [Am]{x}* = [An]{x} + {R}
      !=====

      Aelem = Mele/dt(ie) + (Cele + Kele)/2.0D0
      Aelen = Mele/dt(ie) - (Cele + Kele)/2.0D0

      axele = Aelem

      !-----
      ! Setup {Rav}
      !-----

      rav(1) = Aelen(1,1)*v(ii) + Aelen(1,2)*v(jj) + Aelen(1,3)*v(kk)
      rav(2) = Aelen(2,1)*v(ii) + Aelen(2,2)*v(jj) + Aelen(2,3)*v(kk)
      rav(3) = Aelen(3,1)*v(ii) + Aelen(3,2)*v(jj) + Aelen(3,3)*v(kk)

      !-----
      ! Sum term in y:

```

```

!-----
ryele = rav + rpy + rgy + rbv

!-----
!ASSMBLE() ! Subroutine for assembling the element equations
!-----

DO i = 1,3
  DO j = 1,3
    ii = intmat(ie,i)
    jj = intmat(ie,j)
    sysk(ii,jj) = sysk(ii,jj) + axele(i,j)
  END DO
END DO

DO i = 1,3
  ii = intmat(ie,i)
  sysry(ii) = sysry(ii) + ryele(i)
END DO

!-----

END DO

!-----

CALL APPLYBC(ibcv, v, sysry)

CALL unsympcg(sysk, sysry, v, npoi)

!-----

END SUBROUTINE STEP1_V

!=====

SUBROUTINE STEP2_UV() ! Subroutine for solve u(n+1) & v(n+1); Step 3

USE VARIABLE

IMPLICIT NONE

REAL(8), DIMENSION(3)      :: Rdpv, Rdpny, Rdpwy, Rdpv, Rdpnx, Rdpwx
REAL(8), DIMENSION(3)      :: Rux, Ruy
REAL(8)                     :: uba, vba, absu, peclet, alp
REAL(8), DIMENSION(npoi)   :: dig

uhat = u
vhat = v
sysrx = 0.0D0
sysry = 0.0D0
dig = 0.0D0
rxele = 0.0D0
ryele = 0.0D0

!-----
! Loop over the number of elements
!-----

DO ie = 1,nelem

!-----
! Find element local coordinates:
!-----

ii = intmat(ie,1)
jj = intmat(ie,2)
kk = intmat(ie,3)
xg1 = coord(ii,1)
xg2 = coord(jj,1)
xg3 = coord(kk,1)

```

```

yg1 = coord(ii,2)
yg2 = coord(jj,2)
yg3 = coord(kk,2)

area = 0.5D0*(xg2*(yg3-yg1)+xg1*(yg2-yg3)+xg3*(yg1-yg2))

IF(area <= 0.) WRITE(6,5) ie
5 FORMAT(/,'    !!! ERROR !!! ELEMENT NO.', I5,           &
        ' HAS NEGATIVE OR ZERO area ', /,           &
        ' --- check F.E. MODEL FOR NODAL COORDINATES', &
        ' AND ELEMENT NODAL CONNECTIONS ---'      )
IF(area <= 0.) STOP

b1 = yg2 - yg3
b2 = yg3 - yg1
b3 = yg1 - yg2

c1 = xg3 - xg2
c2 = xg1 - xg3
c3 = xg2 - xg1

!-----
! Setup Uba , Vba
!-----

uba = (uold(ii)+uold(jj)+uold(kk))/3.0D0
vba = (vold(ii)+vold(jj)+vold(kk))/3.0D0

!=====
! Setup [Rdpx] matrix:
!=====

Rdpx(1) = (b1*p(ii)+b2*p(jj)+b3*p(kk))/(6.0D0*den)
Rdpx(2) = Rdpx(1)
Rdpx(3) = Rdpx(1)

Rdpxw(1) = (b1*uba + c1*vba)*(b1*p(ii)+b2*p(jj)+b3*p(kk))/(8.0D0*den*area)
Rdpxw(2) = (b2*uba + c2*vba)*(b1*p(ii)+b2*p(jj)+b3*p(kk))/(8.0D0*den*area)
Rdpxw(3) = (b3*uba + c3*vba)*(b1*p(ii)+b2*p(jj)+b3*p(kk))/(8.0D0*den*area)

Rdpx = Rdpx + dte(ie)*Rdpxw

!=====
! Setup [Rdpy] matrix:
!=====

Rdpy(1) = (c1*p(ii)+c2*p(jj)+c3*p(kk))/(6.0D0*den)
Rdpy(2) = Rdpy(1)
Rdpy(3) = Rdpy(1)

Rdpyw(1) = (b1*uba + c1*vba)*(c1*p(ii)+c2*p(jj)+c3*p(kk))/(8.0D0*den*area)
Rdpyw(2) = (b2*uba + c2*vba)*(c1*p(ii)+c2*p(jj)+c3*p(kk))/(8.0D0*den*area)
Rdpyw(3) = (b3*uba + c3*vba)*(c1*p(ii)+c2*p(jj)+c3*p(kk))/(8.0D0*den*area)

Rdpy = Rdpy + dte(ie)*Rdpyw

!-----
! Sum term in x:
!-----

rxle = Rdpx*dt(ie)*3.0D0/(area*2.0D0)

!-----
! Sum term in y:
!-----

ryle = Rdpy*dt(ie)*3.0D0/(area*2.0D0)

!-----

IF(ltype(ie,1) == 1) then      ! For Solid

rxle = 0.0D0

```

```

ryele = 0.0D0

ENDIF
!-----

!-----
! Apply BC on element
!-----

      DO i = 1,3
        ii = intmat(ie,i)
        sysrx(ii) = sysrx(ii) + rxele(i)
        sysry(ii) = sysry(ii) + ryele(i)
        dig(ii)   = dig(ii)   + 1.0D0
      END DO

END DO

DO i = 1, npoi
      sysrx(i) = sysrx(i)/dig(i)
      sysry(i) = sysry(i)/dig(i)
END DO

!-----
! Evaluate the new value of velocities
!-----

DO i = 1, npoi
      IF(ibcu(i) == 1) THEN
        u(i) = uold(i)
      ELSE
        u(i) = uhat(i) + sysrx(i)
      ENDIF

      IF(ibcv(i) == 1) THEN
        v(i) = vold(i)
      ELSE
        v(i) = vhat(i) + sysry(i)
      ENDIF
END DO

END SUBROUTINE STEP2_UV

!=====

SUBROUTINE STEP3_P() ! Subroutine for solve Pressure

USE VARIABLE
USE SOLVE

IMPLICIT NONE

REAL(8)          :: cru, crv, xb1,  xb2,  yb1,  yb2
REAL(8)          :: unx, uny
REAL(8), DIMENSION(3,3) :: Kele
REAL(8), DIMENSION(3)   :: rpu, rpv, rb
REAL(8)          :: sum

ustar = u
vstar = v
pold = p
aele = 0.0D0
sysk = 0.0D0
rele = 0.0D0
sysr = 0.0D0

DO ie = 1,nelem ! Loop over the number of elements:

```

```

ii = intmat(ie,1)
jj = intmat(ie,2)
kk = intmat(ie,3)

xg1 = coord(ii,1)
xg2 = coord(jj,1)
xg3 = coord(kk,1)

yg1 = coord(ii,2)
yg2 = coord(jj,2)
yg3 = coord(kk,2)

area = 0.50D0*(xg2*(yg3-yg1)+xg1*(yg2-yg3)+xg3*(yg1-yg2))

IF(area <= 0.) WRITE(6,5) ie
5 FORMAT(/,'   !!! ERROR !!! ELEMENT NO.', I5,           &
' HAS NEGATIVE OR ZERO area ', /,           &
' --- check F.E. MODEL FOR NODAL COORDINATES', &
' AND ELEMENT NODAL CONNECTIONS ---'      )

IF(area <= 0.) STOP

b1 = yg2 - yg3
b2 = yg3 - yg1
b3 = yg1 - yg2

c1 = xg3 - xg2
c2 = xg1 - xg3
c3 = xg2 - xg1

!=====
! Setup [K] matrix: (Diffusion term)
!=====

Kele(1,1) = ((b1*b1)+(c1*c1))/(4.0D0*area)
Kele(1,2) = ((b1*b2)+(c1*c2))/(4.0D0*area)
Kele(1,3) = ((b1*b3)+(c1*c3))/(4.0D0*area)

Kele(2,2) = ((b2*b2)+(c2*c2))/(4.0D0*area)
Kele(2,3) = ((b2*b3)+(c2*c3))/(4.0D0*area)
Kele(3,3) = ((b3*b3)+(c3*c3))/(4.0D0*area)

Kele(2,1) = Kele(1,2)
Kele(3,1) = Kele(1,3)
Kele(3,2) = Kele(2,3)

!-----
IF(ltype(ie,1) == 1) then           ! For Solid

Kele = 0.0D0

Kele(1,1) = 1E-14
Kele(2,2) = 1E-14
Kele(3,3) = 1E-14

ENDIF
!-----
!-----
! Setup [ru] matrix
!-----

cru   = (ustar(ii)+ustar(jj)+ustar(kk))*den/(6.0D0*dt(ie))
rpu(1) = cru*b1
rpu(2) = cru*b2
rpu(3) = cru*b3

!-----
! Setup [rv] matrix
!-----

crv   = (vstar(ii)+vstar(jj)+vstar(kk))*den/(6.0D0*dt(ie))
rpv(1) = crv*c1

```

```

rpv(2) = crv*c2
rpv(3) = crv*c3
!-----
! Setup [rb] matrix
!-----

rb = 0.0D0

DO i = 1,nbou

IF(ie == intbou(i,3)) THEN

!-----
! To apply boundary conditions
!           Inflow      => edge = 1
!           Outflow     => edge = 2
!           Symmetry    => edge = 4
!-----

IF(intbou(i,4) == 1 .or. 2 .or. 40) THEN

    ib = intbou(i,1)
    jb = intbou(i,2)

    xb1 = coord(ib,1)
    xb2 = coord(jb,1)

    yb1 = coord(ib,2)
    yb2 = coord(jb,2)

    unx = yb2 - yb1
    uny = xb1 - xb2

IF(ib == ii) THEN

rb(1) = ((uhat(ib)/3.0D0)+(uhat(jb)/6.0D0))*unx+((vhat(ib)/3.0D0)+(vhat(jb)/6.0D0))*uny
rb(2) = ((uhat(ib)/6.0D0)+(uhat(jb)/3.0D0))*unx+((vhat(ib)/6.0D0)+(vhat(jb)/3.0D0))*uny
rb(3) = 0.0D0

ENDIF

IF(ib == jj) THEN

rb(1) = 0.0D0
rb(2) = ((uhat(ib)/3.0D0)+(uhat(jb)/6.0D0))*unx+((vhat(ib)/3.0D0)+(vhat(jb)/6.0D0))*uny
rb(3) = ((uhat(ib)/6.0D0)+(uhat(jb)/3.0D0))*unx+((vhat(ib)/6.0D0)+(vhat(jb)/3.0D0))*uny

ENDIF

IF(ib == kk) THEN

rb(1) = ((uhat(ib)/6.0D0)+(uhat(jb)/3.0D0))*unx+((vhat(ib)/6.0D0)+(vhat(jb)/3.0D0))*uny
rb(2) = 0.0D0
rb(3) = ((uhat(ib)/3.0D0)+(uhat(jb)/6.0D0))*unx+((vhat(ib)/3.0D0)+(vhat(jb)/6.0D0))*uny

ENDIF

ENDIF

rb = den*rb/dt(ie)

ENDIF

ENDDO

!-----

aele = Kele

rele = (rpu + rpv - rb)*2.0D0

!-----
!ASSMBLE() ! Subroutine for assembling the element equations
!-----

```

```

DO i = 1,3
  DO j = 1,3
    ii = intmat(ie,i)
    jj = intmat(ie,j)
    sysk(ii,jj) = sysk(ii,jj) + aele(i,j)
  END DO
END DO

DO i = 1,3
  ii = intmat(ie,i)
  sysr(ii) = sysr(ii) + rele(i)
END DO

!-----
END DO

!-----

CALL APPLYBC(ibcp, p, sysr)

CALL sympcg(sysk, sysr, p, npoi)

!-----

DO i = 1, npoi
  IF (ABS(p(i)) < 1.E-8) THEN
    p(i) = 0.0D0
  ENDIF
END DO

END SUBROUTINE STEP3_P

!=====

SUBROUTINE STEP4_UV() ! Subroutine for solve u(n+1) & v(n+1); Step 3

USE VARIABLE

IMPLICIT NONE

REAL(8), DIMENSION(3)      :: Rdpv, Rdpny, Rdpwy, Rdpv, Rdpnx, Rdpwx
REAL(8), DIMENSION(3)      :: Rux, Ruy
REAL(8)                     :: uba, vba, absu, peclet, alp
REAL(8), DIMENSION(npoi)   :: dig

sysrx = 0.0D0
sysry = 0.0D0
dig    = 0.0D0
rxele  = 0.0D0
ryele  = 0.0D0

!-----
! Loop over the number of elements
!-----

DO ie = 1, nelem

!-----
! Find element local coordinates:
!-----

ii = intmat(ie,1)
jj = intmat(ie,2)
kk = intmat(ie,3)

xg1 = coord(ii,1)
xg2 = coord(jj,1)
xg3 = coord(kk,1)

yg1 = coord(ii,2)
yg2 = coord(jj,2)

```



```

yg3 = coord(kk,2)

area = 0.50D0*(xg2*(yg3-yg1)+xg1*(yg2-yg3)+xg3*(yg1-yg2))

IF(area <= 0.) WRITE(6,5) ie
5 FORMAT(/,'    !!! ERROR !!! ELEMENT NO.', I5,           &
' HAS NEGATIVE OR ZERO area ', /,                       &
' --- check F.E. MODEL FOR NODAL COORDINATES', &
' AND ELEMENT NODAL CONNECTIONS ---' )
IF(area <= 0.) STOP

b1 = yg2 - yg3
b2 = yg3 - yg1
b3 = yg1 - yg2

c1 = xg3 - xg2
c2 = xg1 - xg3
c3 = xg2 - xg1

!-----
! Setup Uba , Vba
!-----

uba = (ustar(ii)+ustar(jj)+ustar(kk))/3.0D0
vba = (vstar(ii)+vstar(jj)+vstar(kk))/3.0D0

!=====
! Setup [Rdpx] matrix:
!=====

Rdpx(1) = -(b1*p(ii)+b2*p(jj)+b3*p(kk))/(6.0D0*den)
Rdpx(2) = Rdpx(1)
Rdpx(3) = Rdpx(1)

Rdpx(1) = -(b1*uba + c1*vba)*(b1*p(ii)+b2*p(jj)+b3*p(kk))/(8.0D0*den*area)
Rdpx(2) = -(b2*uba + c2*vba)*(b1*p(ii)+b2*p(jj)+b3*p(kk))/(8.0D0*den*area)
Rdpx(3) = -(b3*uba + c3*vba)*(b1*p(ii)+b2*p(jj)+b3*p(kk))/(8.0D0*den*area)

Rdpx = Rdpx + dte(ie)*Rdpx

!=====
! Setup [Rdpy] matrix:
!=====

Rdpy(1) = -(c1*p(ii)+c2*p(jj)+c3*p(kk))/(6.0D0*den)
Rdpy(2) = Rdpy(1)
Rdpy(3) = Rdpy(1)

Rdpy(1) = -(b1*uba + c1*vba)*(c1*p(ii)+c2*p(jj)+c3*p(kk))/(8.0D0*den*area)
Rdpy(2) = -(b2*uba + c2*vba)*(c1*p(ii)+c2*p(jj)+c3*p(kk))/(8.0D0*den*area)
Rdpy(3) = -(b3*uba + c3*vba)*(c1*p(ii)+c2*p(jj)+c3*p(kk))/(8.0D0*den*area)

Rdpy = Rdpy + dte(ie)*Rdpy

!-----
! Sum term in x:
!-----

rxle = Rdpx*dt(ie)*3.0D0/(area*2.0D0)

!-----
! Sum term in y:
!-----

ryle = Rdpy*dt(ie)*3.0D0/(area*2.0D0)

!-----
IF(ltype(ie,1) == 1) then           ! For Solid

rxle = 0.0D0
ryle = 0.0D0

ENDIF
!-----

```

```

!-----
! Apply BC on element
!-----

      DO i = 1,3
          ii = intmat(ie,i)

          sysrx(ii) = sysrx(ii) + rxele(i)
          sysry(ii) = sysry(ii) + ryele(i)
          dig(ii)   = dig(ii)   + 1.0D0

      END DO

END DO

      DO i = 1, npoi

          sysrx(i) = sysrx(i)/dig(i)
          sysry(i) = sysry(i)/dig(i)

      END DO

!-----
! Evaluate the new value of velocities
!-----

DO i = 1, npoi

      IF(ibcu(i) == 1) THEN
          u(i) = uold(i)
      ELSE
          u(i) = ustar(i) + sysrx(i)
      ENDIF

      IF(ibcv(i) == 1) THEN
          v(i) = vold(i)
      ELSE
          v(i) = vstar(i) + sysry(i)
      ENDIF

      uv(i) = SQRT( u(i)*u(i) + v(i)*v(i) )

END DO

END SUBROUTINE STEP4_UV

!=====
SUBROUTINE STEP5_T() ! Subroutine for solve Temperature

USE VARIABLE
USE SOLVE

IMPLICIT NONE

REAL(8)                :: sum, error
REAL(8), DIMENSION(3) :: ele, tb
INTEGER(4)             :: it, niter
INTEGER(4)             :: mm, ic, ir, ieq
REAL(8), DIMENSION(3,3) :: Mele, Cele, Cn, Cw, Kele, Aelem, Aelen
REAL(8), DIMENSION(3,3) :: Celeo, Cno, Cwo
REAL(8), DIMENSION(3)  :: qq, qqn, qqw, qs, qsn, qsw, rat, rbt
REAL(8), DIMENSION(3)  :: rb1, rb2
REAL(8)                :: cqqn, cqqw, qqs, LL
REAL(8)                :: canu, canv, caw1, caw23, caw4
REAL(8)                :: canuo, canvo, caw1o, caw23o, caw4o
REAL(8)                :: uba, vba, ubao, vbao
REAL(8)                :: unx, uny, xb1,  xb2,  yb1,  yb2
REAL(8)                :: absu, peclet, alp

REAL(8), DIMENSION(3,3) :: Mn, Mw

told = t

```

```

aelem = 0.0D0
aelen = 0.0D0
rele  = 0.0D0
sysr  = 0.0D0
sysk  = 0.0D0

!
! Loop over the number of elements
!

DO ie = 1,nelem

!-----
! Find element local coordinates:
!-----

ii = intmat(ie,1)
jj = intmat(ie,2)
kk = intmat(ie,3)

xg1 = coord(ii,1)
xg2 = coord(jj,1)
xg3 = coord(kk,1)

yg1 = coord(ii,2)
yg2 = coord(jj,2)
yg3 = coord(kk,2)

area = 0.5D0*(xg2*(yg3-yg1)+xg1*(yg2-yg3)+xg3*(yg1-yg2))

IF(area <= 0.) WRITE(6,5) ie
5 FORMAT(/,'    !!! ERROR !!! ELEMENT NO.', I5,           &
          ' HAS NEGATIVE OR ZERO area ', /,           &
          ' --- check F.E. MODEL FOR NODAL COORDINATES', &
          ' AND ELEMENT NODAL CONNECTIONS ---'      )
IF(area <= 0.) STOP

b1 = yg2 - yg3
b2 = yg3 - yg1
b3 = yg1 - yg2

c1 = xg3 - xg2
c2 = xg1 - xg3
c3 = xg2 - xg1

!-----
! Setup Uba , Vba
!-----

uba = (u(ii)+u(jj)+u(kk))/3.0D0
vba = (v(ii)+v(jj)+v(kk))/3.0D0

ubao = (uold(ii)+uold(jj)+uold(kk))/3.0D0
vbao = (vold(ii)+vold(jj)+vold(kk))/3.0D0

!-----
! Element Size
!-----

IF(weight == 1) THEN

  absu = SQRT((uba*uba) + (vba*vba))
  peclet = absu*hele(ie)*den*spec/(2.0D0*condf)
  !
  alp = 1.0D0/(DTANH(peclet)) - 1.0D0/peclet
  alp = 1.0D0/(DTANH(peclet/2.0D0)) - 2.0D0/peclet
  dte(ie) = alp*hele(ie)/absu
  IF(peclet == 0.0D0 .or. dte(ie) >= dt(ie)) dte(ie) = dt(ie)
  IF(ltype(ie,1) == 1) dte(ie) = dt(ie)    ! For Solid

ENDIF

ENDIF

```

```

=====
! Setup [M] matrix: (Mass term)
=====

Mele(1,1) = area/6.0D0
Mele(1,2) = area/12.0D0
Mele(1,3) = area/12.0D0

Mele(2,1) = area/12.0D0
Mele(2,2) = area/6.0D0
Mele(2,3) = area/12.0D0

Mele(3,1) = area/12.0D0
Mele(3,2) = area/12.0D0
Mele(3,3) = area/6.0D0

=====
! Setup [C] matrix: (Convection term)
=====

!-----
! Setup [Cn+1] matrix
!-----

canu = uba/6.0D0
canv = vba/6.0D0

Cn(1,1) = canu*b1 + canv*c1
Cn(1,2) = canu*b2 + canv*c2
Cn(1,3) = canu*b3 + canv*c3

Cn(2,1) = Cn(1,1)
Cn(2,2) = Cn(1,2)
Cn(2,3) = Cn(1,3)

Cn(3,1) = Cn(1,1)
Cn(3,2) = Cn(1,2)
Cn(3,3) = Cn(1,3)

!-----
! Setup [Cn] matrix
!-----

canuo = ubao/6.0D0
canvo = vbao/6.0D0

Cno(1,1) = canuo*b1 + canvo*c1
Cno(1,2) = canuo*b2 + canvo*c2
Cno(1,3) = canuo*b3 + canvo*c3

Cno(2,1) = Cno(1,1)
Cno(2,2) = Cno(1,2)
Cno(2,3) = Cno(1,3)

Cno(3,1) = Cno(1,1)
Cno(3,2) = Cno(1,2)
Cno(3,3) = Cno(1,3)

!-----
! Setup [Cw]n+1 matrix
!-----

caw1 = uba*uba/(8.0D0*area)
caw23 = uba*vba/(8.0D0*area)
caw4 = vba*vba/(8.0D0*area)

Cw(1,1) = caw1*b1*b1 + caw23*(b1*c1 + b1*c1) + caw4*c1*c1
Cw(1,2) = caw1*b1*b2 + caw23*(b1*c2 + b2*c1) + caw4*c1*c2
Cw(1,3) = caw1*b1*b3 + caw23*(b1*c3 + b3*c1) + caw4*c1*c3

Cw(2,1) = Cw(1,2)
Cw(2,2) = caw1*b2*b2 + caw23*(b2*c2 + b2*c2) + caw4*c2*c2
Cw(2,3) = caw1*b2*b3 + caw23*(b2*c3 + b3*c2) + caw4*c2*c3

```

```

Cw(3,1) = Cw(1,3)
Cw(3,2) = Cw(2,3)
Cw(3,3) = caw1*b3*b3 + caw23*(b3*c3 + b3*c3) + caw4*c3*c3

!-----
! Setup [Cw]n matrix
!-----

caw1o = ubao*ubao/(8.0D0*area)
caw23o = ubao*vbao/(8.0D0*area)
caw4o = vbao*vbao/(8.0D0*area)

Cwo(1,1) = caw1o*b1*b1 + caw23o*(b1*c1 + b1*c1) + caw4o*c1*c1
Cwo(1,2) = caw1o*b1*b2 + caw23o*(b1*c2 + b2*c1) + caw4o*c1*c2
Cwo(1,3) = caw1o*b1*b3 + caw23o*(b1*c3 + b3*c1) + caw4o*c1*c3

Cwo(2,1) = Cwo(1,2)
Cwo(2,2) = caw1o*b2*b2 + caw23o*(b2*c2 + b2*c2) + caw4o*c2*c2
Cwo(2,3) = caw1o*b2*b3 + caw23o*(b2*c3 + b3*c2) + caw4o*c2*c3

Cwo(3,1) = Cwo(1,3)
Cwo(3,2) = Cwo(2,3)
Cwo(3,3) = caw1o*b3*b3 + caw23o*(b3*c3 + b3*c3) + caw4o*c3*c3

!-----
! Setup [C] matrix : [Cn] + [Cw]
!-----

Cele = Cn + dte(ie)*Cw
Celeo = Cno + dte(ie)*Cwo

!-----
! Setup [K]
!-----

ks = conds/(dens*specs)
kf = condff/(den*spec)

IF(ltype(ie,1) == 1) then          ! For Solid

Kele(1,1) = ((b1*b1)+(c1*c1))*ks/(4.0D0*area)
Kele(1,2) = ((b1*b2)+(c1*c2))*ks/(4.0D0*area)
Kele(1,3) = ((b1*b3)+(c1*c3))*ks/(4.0D0*area)

Kele(2,2) = ((b2*b2)+(c2*c2))*ks/(4.0D0*area)
Kele(2,3) = ((b2*b3)+(c2*c3))*ks/(4.0D0*area)
Kele(3,3) = ((b3*b3)+(c3*c3))*ks/(4.0D0*area)

Kele(2,1) = Kele(1,2)
Kele(3,1) = Kele(1,3)
Kele(3,2) = Kele(2,3)

ENDIF

IF(ltype(ie,1) == 0) then        ! For Fluid

Kele(1,1) = ((b1*b1)+(c1*c1))*kf/(4.0D0*area)
Kele(1,2) = ((b1*b2)+(c1*c2))*kf/(4.0D0*area)
Kele(1,3) = ((b1*b3)+(c1*c3))*kf/(4.0D0*area)

Kele(2,2) = ((b2*b2)+(c2*c2))*kf/(4.0D0*area)
Kele(2,3) = ((b2*b3)+(c2*c3))*kf/(4.0D0*area)
Kele(3,3) = ((b3*b3)+(c3*c3))*kf/(4.0D0*area)

Kele(2,1) = Kele(1,2)
Kele(3,1) = Kele(1,3)
Kele(3,2) = Kele(2,3)

ENDIF

!-----
!-----

```

```

! Setup [RQ] matrix
!-----

qq = 0.0D0
qs = 0.0D0

If(ltype(ie,1) == 1 .and. ltype(ie,2) == 1) then      ! For Solid

    cqqn = q*area/(3.0D0*specs)
    qqn(1) = cqqn
    qqn(2) = cqqn
    qqn(3) = cqqn

    cqqw = q*area/specs
    qqw(1) = cqqw*(b1*uba + c1*vba)
    qqw(2) = cqqw*(b2*uba + c2*vba)
    qqw(3) = cqqw*(b3*uba + c3*vba)

    qq = qqn + dte(ie)*qqw

    qq = 0.0D0

ENDIF

If(ltype(ie,1) == 0 .and. ltype(ie,2) == 1) then      ! For Fluid

    cqqn = q*area/(3.0D0*spec)
    qqn(1) = cqqn*1.0D0
    qqn(2) = cqqn*1.0D0
    qqn(3) = cqqn*1.0D0

    cqqw = q*area/spec
    qqw(1) = cqqw*(b1*uba + c1*vba)
    qqw(2) = cqqw*(b2*uba + c2*vba)
    qqw(3) = cqqw*(b3*uba + c3*vba)

    qq = qqn + dte(ie)*qqw

    qq = 0.0D0

ENDIF

!-----
! Setup [Qs] matrix: (fluid)
!-----

qqqs = 0.0D0

!=====
! Setup matrix : [A]{T}(n+1) = [A]{T}(n) + {f}
!=====

Aelem = Mele/dt(ie) + (Cele + Kele)/2.0D0
Aelen = Mele/dt(ie) - (Cele + Kele)/2.0D0
aele = Aelem

!-----
! Setup [A]{T}(n) matrix:
!-----

rat(1) = Aelen(1,1)*t(ii) + Aelen(1,2)*t(jj) + Aelen(1,3)*t(kk)
rat(2) = Aelen(2,1)*t(ii) + Aelen(2,2)*t(jj) + Aelen(2,3)*t(kk)
rat(3) = Aelen(3,1)*t(ii) + Aelen(3,2)*t(jj) + Aelen(3,3)*t(kk)

!=====
! Setup [rbt] matrix
!=====

rbt = 0.0D0
rb1 = 0.0D0
rb2 = 0.0D0

```

```

!-----
! Check boundary
!-----

DO i = 1,nbou

IF(ie == intbou(i,3)) THEN

    bt = b1*t(ii) + b2*t(jj) + b3*t(kk)
    ct = c1*t(ii) + c2*t(jj) + c3*t(kk)

!-----
! To apply boundary conditions
!           Inflow      => edge = 1
!           Outflow     => edge = 2
!           Symmetry    => edge = 4
!-----

IF(intbou(i,4) == 1 .or. 2 .or. 40) THEN

    ib = intbou(i,1)
    jb = intbou(i,2)

    xb1 = coord(ib,1)
    xb2 = coord(jb,1)

    yb1 = coord(ib,2)
    yb2 = coord(jb,2)

    unx = yb2 - yb1
    uny = xb1 - xb2

    IF(ib == ii) THEN
        rbl(1) = bt*unx + ct*uny
        rbl(2) = bt*unx + ct*uny
        rbl(3) = 0.0D0
    ENDIF

    IF(ib == jj) THEN
        rbl(1) = 0.0D0
        rbl(2) = bt*unx + ct*uny
        rbl(3) = bt*unx + ct*uny
    ENDIF

    IF(ib == kk) THEN
        rbl(1) = bt*unx + ct*uny
        rbl(2) = 0.0D0
        rbl(3) = bt*unx + ct*uny
    ENDIF

ENDIF

ENDIF

!-----
! To apply wall boundary conditions
!           Wall => edge = 5
!-----

IF(intbou(i,4) == 5) THEN

    ib = intbou(i,1)
    jb = intbou(i,2)

    xb1 = coord(ib,1)
    xb2 = coord(jb,1)

    yb1 = coord(ib,2)
    yb2 = coord(jb,2)

    unx = yb2 - yb1
    uny = xb1 - xb2

    IF(ib == ii) THEN
        rb2(1) = bt*unx + ct*uny
        rb2(2) = bt*unx + ct*uny
    ENDIF

ENDIF

```

```

        rb2(3) = 0.0D0
    ENDIF

    IF(ib == jj) THEN
        rb2(1) = 0.0D0
        rb2(2) = bt*unx + ct*uny
        rb2(3) = bt*unx + ct*uny
    ENDIF

    IF(ib == kk) THEN
        rb2(1) = bt*unx + ct*uny
        rb2(2) = 0.0D0
        rb2(3) = bt*unx + ct*uny
    ENDIF

ENDIF

rbt = rb1 + rb2

    IF(ltype(ie,1) == 0) then      ! For Fluid
        rbt = kf*rbt/(4.0D0*area)
    ENDIF

    IF(ltype(ie,1) == 1) then      ! For Solid
        rbt = ks*rbt/(4.0D0*area)
    ENDIF

ENDIF
ENDDO

!-----
! Setup [r] matrix:
!-----

rele = rat + qq + rbt

!-----
!ASSMBLE() ! Subroutine for assembling the element equations
!-----

DO i = 1,3
    DO      j = 1,3
        ii = intmat(ie,i)
        jj = intmat(ie,j)
        sysk(ii,jj) = sysk(ii,jj) + aele(i,j)
    END DO
END DO

DO
    DO      i = 1,3
        ii = intmat(ie,i)
        sysr(ii) = sysr(ii) + rele(i)
    END DO
END DO

!-----
END DO
!-----

CALL APPLYBC(ibct, t, sysr)

CALL unsympcgt(sysk, sysr, t, npoi)

!-----

END SUBROUTINE STEP5_T

```



```

=====
SUBROUTINE ERROR()

USE VARIABLE

IMPLICIT NONE

!-----
! check for convergence
!-----
sumu = 0.0D0
sumv = 0.0D0
sump = 0.0D0
sumt = 0.0D0

sumdu = 0.0D0
sumdv = 0.0D0
sumdp = 0.0D0
sumdt = 0.0D0

DO i = 1,npoi
  sumdu = sumdu + ABS(uold(i)-u(i))
  sumu  = sumu  + ABS(u(i))

  sumdv = sumdv + ABS(vold(i)-v(i))
  sumv  = sumv  + ABS(v(i))

  sumdp = sumdp + ABS(pold(i)-p(i))
  sump  = sump  + ABS(p(i))

  sumdt = sumdt + ABS(told(i)-t(i))
  sumt  = sumt  + ABS(t(i))
END DO

erroru = sumdu/sumu
errorv = sumdv/sumv
errorp = sumdp/sump
errort = sumdt/sumt

WRITE(6,360) iter, erroru, errorv, errorp, errort
WRITE(11,360) iter, erroru, errorv, errorp, errort
360 FORMAT('iter = ', I6, 3X, E15.9, 3X, &
          E15.9, 3X, E15.9, 3X, E15.9)

END SUBROUTINE ERROR

=====!

SUBROUTINE WRITE_OUTPUT()

USE VARIABLE

IMPLICIT NONE

CHARACTER(len=1)  :: unit, dec, hun, thou, tthou, hthou
INTEGER(4)        :: leng, freq, number
INTEGER(4)        :: nodemax, elemx, bbb
INTEGER(4)        :: bound
REAL(8)           :: x, y
REAL(8)           :: xb1, xb2, xb3, yb1, yb2, yb3, LL
REAL(8)           :: unx, uny, dTx, dTy, xc, yc
REAL(8)           :: T1, T2, T3, xx, yy, deg

freq = mod(iter,res)

if(freq == 0 .or. iter == 1) then

number = iter

  n = int(number/100000)
  hthou = CHAR(n + 48)

```

```

number = number - n*100000

    m = INT(number/10000)
tthou = CHAR(m + 48)

number = number - m*10000

    l = INT(number/1000)
thou = CHAR(l + 48)

number = number - l*1000

    k = INT(number/100)
hun = CHAR(k + 48)

number = number - k*100

    j = INT(number/10)
dec = CHAR(j + 48)

    i = mod(number,10)
unit = CHAR(i + 48)

leng = len_trim(name1) - 2

!-----
OPEN(UNIT=16,
FILE=name1(1:leng)//'_'//hthou//tthou//thou//hun//dec//unit//'.plt',STATUS='unknown',I
OSTAT=ierror)

    WRITE(16,220)
    220 FORMAT ( 'VARIABLES = "X-CO","Y-CO","U","V","UV","P","T" ' )

    WRITE(16,240) npoi, nelelem
    240 FORMAT('ZONE N=', I6,',E=',I6, ',F=FEPOINT,ET=TRIANGLE')

DO i = 1,npoi
    WRITE(16,260) (coord(i,k),k=1,2),u(i),v(i),uv(i),p(i),t(i)
    260 FORMAT(2F12.6, 2X, E15.6, 2X, E15.6, 2X, E15.6, 2X, E15.6, 2X, E15.6)
    END DO

DO i = 1,nelelem
    WRITE(16,280) (intmat(i,j),j=1,3)
    280 FORMAT(3I6)
    END DO

CLOSE(UNIT=16, STATUS='keep')

!=====
!  WRITE INPUT FOR THERMAL STRESS
!=====

OPEN(UNIT=17,
FILE=name1(1:leng)//'_'//hthou//tthou//thou//hun//dec//unit//'.s',STATUS='unknown',IOS
TAT=ierror)

bbb = 0.

DO ie = 1,nelelem
    IF( ltype(ie,1) == 1) THEN
        DO j = 1,3
            if(intmat(ie,j) >= bbb) nodemax = intmat(ie,j)
            bbb = nodemax
        ENDDO
        elemax = ie
    ENDIF
ENDDO

    WRITE(17,282)
    282 FORMAT ( '1',/,'Data for',/,' npoin', 5X,'nelem',5X,'nboun',5X,'stype' )

    WRITE(17,283) nodemax, elemax, 1, 0

```

```

283 FORMAT(I6,4X,I6,2X,I6,3X,I6)

WRITE(17,285)
285 FORMAT ( ' E',10X,'Pr',7X,'Tref',6X,'Alpha',5X,'Thick' )

WRITE(17,286)
286 FORMAT ( ' 7E10      0.3      0      6E-6      1' )

WRITE(17,287)
287  FORMAT      (      '      NODE',5X,      'ibcx',5X,'ibcy',10X,      'x',      12X,
      'y',9X,'disx',6X,'disy',10X,'sxx',12X,'syy',14X,'temp' )

DO i = 1,nodemax

      WRITE(17,288) i, 0, 0, (coord(i,k),k=1,2), 0, 0, p(i), p(i), t(i)
288      FORMAT(I6,3X,I6,3X,I6,4X,      F12.8,2X,F12.8,      I6,5X,I6,3X,
E15.6,2X,E15.6,2X,E15.6)

END DO

      WRITE(17,289)
289  FORMAT ( '      IE',10X,'I',10X,'J',10X,'K')

DO i = 1,elemax
      WRITE(17,290) i, (intmat(i,j),j=1,3)
290  FORMAT(I6, 5X, I6, 5X, I6,5X, I6)
END DO

CLOSE(UNIT=17, STATUS='keep')

!-----
WRITE(6,310) iter
310 FORMAT(/, ' write out file for iteration #',I6 )

ENDIF

END SUBROUTINE WRITE_OUTPUT

!=====
SUBROUTINE READ_INPUT()

USE VARIABLE

IMPLICIT NONE

REAL(8)      :: x, y
INTEGER(4)   :: ib1, ib2

WRITE(6,10)
10  FORMAT(/, ' PLEASE ENTER INPUT FILE NAME:',/)
READ(5, '(A)') name1

OPEN(UNIT=7, FILE=name1, STATUS='OLD', ACTION='READ', IOSTAT=ierror)

!-----
! Read input data
!-----

READ(7,*) nlines
DO i = 1,nlines
      READ(7,1) text
1  FORMAT(20A4)
END DO

READ(7,1) text
READ(7,*) npoi, nelem, nbou, nflux

ALLOCATE( intmat(nelem,3) ,intbou(nbou,4)      ,coord(npoi,2)
)
ALLOCATE( sysk(npoi,npoi) , intbouq(nflux,3) , nodeid(nflux)      )
ALLOCATE( ibcu(npoi) , ibcv(npoi) , ibcp(npoi), ibce(nelem)      )

```

```

ALLOCATE(      u(npoi) ,      v(npoi)      ,      p(npoi) ,      t(npoi)      )
ALLOCATE(      du(npoi) ,      dv(npoi)      ,      dp(npoi) ,      uv(npoi)      )
ALLOCATE(      uold(npoi) ,      vold(npoi)      ,      pold(npoi) ,      told(npoi)      )
ALLOCATE(      sysrx(npoi) ,      sysry(npoi)      ,      sysr(npoi) ,      ibct(npoi)      )
ALLOCATE(      bflux(nflux) ,      esup2(npoi+1)      ,      chke(nelem) ,      elemq(nflux)      )
ALLOCATE(ltype(nelem,2)
)
ALLOCATE(      pbar(npoi) ,      digp(npoi)      )
ALLOCATE(      unew(npoi) ,      vnew(npoi) ,      pnew(npoi) ,      tnew(npoi)      )
ALLOCATE(      EE(npoi) ,      PP(npoi) ,      ENew(npoi) ,      PPnew(npoi) ,      EB(npoi)      )
ALLOCATE(sysrold(npoi) ,      comb(nelem)
)
ALLOCATE(      ustar(npoi) ,      vstar(npoi) ,      uhat(npoi) ,      vhat(npoi)      )
ALLOCATE(      dte(nelem) ,      dt(nelem),      hele(nelem)      )

!-----
! Read Fluid properties
!-----

READ(7,1) text
READ(7,*) den, spec, condf, vis, expans, tref

!-----
! Read Solid properties
!-----

READ(7,1) text
READ(7,*) dens, specs, conds, q

!-----
! Read nodal coordinates, boundary conditions, their values
!-----

READ(7,1) text

DO ip = 1,npoi
  READ(7,*) i,ibcu(i),ibcv(i),ibcp(i),ibct(i),(coord(i,k),k=1,2),u(i),v(i),p(i),t(i))
  IF(i /= ip) WRITE(6,40) ip
  40 FORMAT(/, ' NODE NO.', I5, ' IN DATA FILE IS MISSING')
  IF(i /= ip) STOP
END DO

!-----
! Read element nodal connection
!-----

READ(7,1) text

DO ie = 1,nelem
  READ(7,*) i,(intmat(i,j),j=1,3),(ltype(i,k),k=1,2)
  ibce = 0
  IF(i /= ie) WRITE(6,50) ie
  50 FORMAT(/, ' ELEMENT NO.', I5, ' IN DATA FILE IS MISSING')
  IF(i /= ie) STOP
END DO

!-----
! Read inflow, outflow, wall boundary element
!-----

READ(7,1) text
IF(nbou /= 0.) THEN
  DO ie = 1,nbou
    READ(7,*) (intbou(ie,k), k=1,4)
    ! IF(i /= 0) ibce(i) = 1
  END DO
END IF

CLOSE(UNIT=7, STATUS='keep')

END SUBROUTINE READ_INPUT

!=====
SUBROUTINE T_CRITICAL() ! Calculation time critical

```

```

USE VARIABLE
IMPLICIT NONE

REAL(8)          :: uba, vba
REAL(8)          :: absu, hele1, hele2, hele3
REAL(8)          :: ds1, ds2, ds3
REAL(8)          :: dta,dtm,dtb,bbb
REAL(8), DIMENSION(nelem) :: cc

DO ie = 1,nelem ! Loop over the number of elements:

!-----
! Find element local coordinates:
!-----

ii = intmat(ie,1)
jj = intmat(ie,2)
kk = intmat(ie,3)

xg1 = coord(ii,1)
xg2 = coord(jj,1)
xg3 = coord(kk,1)

yg1 = coord(ii,2)
yg2 = coord(jj,2)
yg3 = coord(kk,2)

area = 0.5D0*(xg2*(yg3-yg1)+xg1*(yg2-yg3)+xg3*(yg1-yg2))

IF(area <= 0.) WRITE(6,5) ie
5 FORMAT(/,'   !!! ERROR !!! ELEMENT NO.', I5,           &
' HAS NEGATIVE OR ZERO area ', /,           &
' --- check F.E. MODEL FOR NODAL COORDINATES', &
' AND ELEMENT NODAL CONNECTIONS ---'      )
IF(area <= 0.) STOP

b1 = yg2 - yg3
b2 = yg3 - yg1
b3 = yg1 - yg2

c1 = xg3 - xg2
c2 = xg1 - xg3
c3 = xg2 - xg1

!-----
! Setup dt on element ; dte
!-----

uba = (u(ii)+u(jj)+u(kk))/3.0D0
vba = (v(ii)+v(jj)+v(kk))/3.0D0

absu = SQRT((uba*uba) + (vba*vba))

!-----
! Element Size
!-----

ds1 = SQRT((xg1-xg2)*(xg1-xg2) + (yg1-yg2)*(yg1-yg2))
hele1 = 2.0D0*area/ds1

ds2 = SQRT((xg2-xg3)*(xg2-xg3) + (yg2-yg3)*(yg2-yg3))
hele2 = 2.0D0*area/ds2

ds3 = SQRT((xg3-xg1)*(xg3-xg1) + (yg3-yg1)*(yg3-yg1))
hele3 = 2.0D0*area/ds3

!-----
! Element Size Mean
!-----

```

```

hele(ie) = (hele1 + hele2 + hele3)/3.0D0

!-----

dta = hele(ie)/absu

dtm = hele(ie)*hele(ie)*den/(2.0D0*vis)

!-----
! time critical
!-----

dte(ie) = min(dta, dtm) ! solve u, v, p

END DO

!-----
! time critical min
!-----

bbb = 1E20

DO ie = 1,nelem
dte = min(bbb,dte(ie))

bbb = dte

END DO

dte = bbb

!-----
! Print out title
!-----

WRITE(6,20) npoi, nelem, dte
20 FORMAT(/, ' THE FINITE ELEMENT MODEL CONSISTS OF :',/, &
' NUMBER OF NODES = ', I6, /, &
' NUMBER OF ELEMENTS = ', I6, /, &
' TIME CRITICAL MINIMUM = ', E6.1, /)

!-----

state = 1 ! state = 1 for steady and Transint
weight = 1 ! Use Weight SUPG

!=====
! Transint
!=====

IF (state == 1) then

WRITE(6,32)
32 FORMAT(/, ' Enter time interval (dt) = ')
READ(5,*) ccc

DO ie = 1, nelem
dt(ie) = ccc
END DO

dte = dt

END IF

WRITE(6,38)
38 FORMAT(/, ' How frequently to write down the result? ')
READ(5,*) res

nstep = 1E9

END SUBROUTINE T_CRITICAL

!=====

```

```

SUBROUTINE APPLYBC(IBC, UVP, SYSR1) ! Subroutine for apply boundary condition

USE VARIABLE

!
! Apply boundary conditions before solving for nodal
! with condition codes of:
!   0 = Free to change
!   1 = Fixed as specified
!
!
IMPLICIT NONE

INTEGER(4)                :: Ieq, Ir, Ic
INTEGER(4), DIMENSION(npoi) :: Ibc
REAL(8), DIMENSION(npoi)  :: Uvp, Sysr1

!-----
! Apply boundary conditions for nodal
!-----

outer: DO Ieq = 1, npoi
    IF(IBC(Ieq) == 0) CYCLE outer

    inner: DO Ir = 1, npoi
        IF(Ir == Ieq) CYCLE inner
        SYSR1(Ir) = SYSR1(Ir) - SYSK(Ir, Ieq)*UVP(Ieq)
        SYSK(Ir, Ieq) = 0.0D0
    END DO inner

    DO Ic = 1, npoi
        SYSK(Ieq, Ic) = 0.0D0
    END DO

    SYSK(Ieq, Ieq) = 1.0D0
    SYSR1(Ieq) = UVP(Ieq)

END DO outer

END SUBROUTINE APPLYBC

!-----

```

ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

```

MODULE VARIABLE

IMPLICIT NONE

CHARACTER(len=20)                :: name1, name2, name3, name4

INTEGER(2)                        :: sthour, stminute, stsecond, sthund  !
Collect Start time
INTEGER(2)                        :: enhour, enminute, ensecond, enhund  !
Collect End time

INTEGER(4), ALLOCATABLE, DIMENSION(:,:) :: intmat                        !
Node connection on element
INTEGER(4), ALLOCATABLE, DIMENSION(:,:) :: intbou                        !
Define boundary inflow, outflow, wall
INTEGER(4), ALLOCATABLE, DIMENSION(:,:) :: intbouq                       !
Define boundary heat flux
INTEGER(4), ALLOCATABLE, DIMENSION(:)  :: ibcu,   ibcv, ibcp, ibct      !
boundary condition for variable
INTEGER(4), ALLOCATABLE, DIMENSION(:)  :: ibcuu,   ibcvv, ibcPPP, ibctt
INTEGER(4), ALLOCATABLE, DIMENSION(:,:) :: ltype, ktype
INTEGER(4), ALLOCATABLE, DIMENSION(:)  :: ibce,   nodeid, chke        !
boundary condition
INTEGER(4), ALLOCATABLE, DIMENSION(:)  :: esup1, esup2, elemq        !
for elements surrounding node
INTEGER(4), DIMENSION(20)              :: text

REAL(8), ALLOCATABLE, DIMENSION(:,:)   :: coord                        !
Coordinate of each node
REAL(8), ALLOCATABLE, DIMENSION(:,:)   :: sysk
REAL(8), ALLOCATABLE, DIMENSION(:)     :: u      , v      , p      , t      , uv    !
Primitive variable
REAL(8), ALLOCATABLE, DIMENSION(:)     :: uhat  , vhat  , ustar , vstar
REAL(8), ALLOCATABLE, DIMENSION(:)     :: du    , dv    , dp
REAL(8), ALLOCATABLE, DIMENSION(:)     :: uold  , vold  , pold  , told    !
Primitive variable (old)
REAL(8), ALLOCATABLE, DIMENSION(:)     :: dte   , dt    , hele

REAL(8), ALLOCATABLE, DIMENSION(:)     :: unew, vnew, pnew, tnew
REAL(8), ALLOCATABLE, DIMENSION(:)     :: EE, PP, EEnew, PPnew, EB
REAL(8), ALLOCATABLE, DIMENSION(:)     :: sysrx, sysry, sysr,  bflux
REAL(8), ALLOCATABLE, DIMENSION(:)     :: tbar, digt, pbar, digp, sysrold
REAL(8), ALLOCATABLE, DIMENSION(:)     :: comb

INTEGER(4)                              :: npoi, neleM, ierror, istor, iter, edge,
cb
INTEGER(4)                              :: i,    j,    k,    ii,  jj,  kk
INTEGER(4)                              :: n,    m,    l
INTEGER(4)                              :: ip,   ie,   ib,   jb,  kb,  st,  ed
INTEGER(4)                              :: nlines, nbou, nflux, ipoil, check, nstep,
res
INTEGER(4)                              :: state, weight, Ty_FLUX, chk

REAL(8), DIMENSION(3,3)                 :: aele
REAL(8), DIMENSION(3)                   :: rxele, ryele, rele
REAL(8)                                  :: sumu, sumv, sump, sumt
REAL(8)                                  :: sumdu, sumdv, sumdp, sumdt
REAL(8)                                  :: erroru, errorv, errorp, errorT
REAL(8)                                  :: aaa, area
REAL(8)                                  :: conds, condf, spec, specs, expan, tref,
den, dens, vis
REAL(8)                                  :: xg1,   xg2,   xg3,   yg1,  yg2,  yg3
REAL(8)                                  :: b1,    b2,    b3,    c1,   c2,   c3
REAL(8)                                  :: f1,    f2,    f3,    uj,   vj
REAL(8)                                  :: dtc
REAL(8)                                  :: q, ks, kf, ccc
REAL(8)                                  :: bu, cu, bv, cv, bt, ct, buv

END MODULE VARIABLE

!-----

```



```

Module SOLVE

implicit none

contains

!
! Subroutine preconditioned conjugate gradient method
!

subroutine unsympcg(a, b, x, n) ! Solve Velocity ; U,V

implicit none

integer(4)          :: maxit, iter, count
integer(4)          :: i, j, n
real(8)             :: del, dnew, tol, bottom
real(8)             :: alpha, dold, beta, check, thresh
real(8), dimension(n,n) :: a
real(8), dimension(n)  :: q, r, d, pre, s, x, b, ax, atb

integer(4), allocatable, dimension(:) :: ija, ijat, ijata
real(8), allocatable, dimension(:) :: sa, sat, sata

  iter = 1
  tol = 1.d-6
  maxit = 10000
  thresh = 1.e-10

! count = 0
!do i = 1,n
!  do j = 1,n
!    if(i/=j) then
!      if(a(i,j) /= 0.) count = count + 1
!    end if
!  end do
!end do
!count = count + (n+1)

count = 1000000

allocate( sa(count), ija(count))
allocate( sat(count), ijat(count))
allocate(sata(count), ijata(count))

!-----
! Convert square matrix into row-indexed sparse
!-----
call sprsin( a,n,n,thresh,count, sa, ija)
call sprsint(a,n,n,thresh,count,sat,ijat)

!-----
! Compute At*A
!-----
call sprstm(sat,ijat,sat,ijat,sata,ijata,thresh,count)

!-----
! Compute At*b
!-----
call sprsax(sat,ijat,b,atb,n,count)

!-----
! Construct precondition matrix P (but this will find P inverse)
!-----
do i = 1,n
  pre(i) = 1./abs(sata(i))
end do

!-----
! Find Ro
!-----

```

```

call sprsax(sata,ijata,x,ax,n,count)

do i = 1,n
  r(i) = atb(i) - ax(i)
end do

!-----
! Find Do
!-----
do i = 1,n
  d(i) = pre(i)*r(i)
end do

!-----
! Find D-New
!-----
dnew = 0.0D0
do i = 1,n
  dnew = dnew + r(i)*d(i)
end do

del = dnew

do
  if(iter >= maxit.OR.dnew <= (tol*tol*del)) exit

!-----
! Find Q(i+1)
!-----

call sprsax(sata,ijata,d,q,n,count)

!-----
! Find Alpha
!-----
bottom = 0.0D0
do i = 1,n
  bottom = bottom + d(i)*q(i)
end do

alpha = dnew/bottom

do i = 1,n
  x(i) = x(i) + alpha*d(i)
end do

!-----
! Find R(i+1)
!-----
check = mod(iter,50)

if(check == 0.) then

call sprsax(sata,ijata,x,ax,n,count)

do i = 1,n
  r(i) = atb(i) - ax(i)
end do

else
  do i = 1,n
    r(i) = r(i) - alpha*q(i)
  end do
endif

!-----
! Find S(i+1)
!-----
do i = 1,n
  s(i) = pre(i)*r(i)
end do

```

```

!-----
! Find beta
!-----
dold = dnew
dnew = 0.

do i = 1,n
  dnew = dnew + r(i)*s(i)
end do

beta = dnew/dold

do i = 1,n
  d(i) = s(i) + beta*d(i)
end do

iter = iter + 1

end do

end subroutine unsympcg

!-----
---

subroutine unsympcgt(a, b, x, n) ! Solve Temperature ; T

implicit none

integer(4)          :: maxit, iter, count
integer(4)          :: i, j, n
real(8)            :: del, dnew, tol, bottom
real(8)            :: alpha, dold, beta, check, thresh
real(8), dimension(n,n) :: a
real(8), dimension(n)  :: q, r, d, pre, s, x, b, ax, atb
integer(4), allocatable, dimension(:) :: ija, ijat, ijata
real(8), allocatable, dimension(:) :: sa, sat, sata

  iter = 1
  tol = 1.d-6
  maxit = 20000
  thresh = 1.e-10

! count = 0
!do i = 1,n
!  do j = 1,n
!    if(i/=j) then
!      if(a(i,j) /= 0.) count = count + 1
!    end if
!  end do
!end do
!count = count + (n+1)

count = 1000000

allocate( sa(count), ija(count))
allocate( sat(count), ijat(count))
allocate(sata(count), ijata(count))

!-----
! Convert square matrix into row-indexed sparse
!-----
call sprsin( a,n,n,thresh,count, sa, ija)
call sprsint(a,n,n,thresh,count,sat,ijat)

!-----
! Compute At*A
!-----
call sprstm(sat,ijat,sat,ijat,sata,ijata,thresh,count)

!-----

```

```

! Compute At*b
!-----
call sprsax(sat,ijat,b,atb,n,count)

!-----
! Construct precondition matrix P (but this will find P inverse)
!-----
do i = 1,n
  pre(i) = 1./abs(sata(i))
end do

!-----
! Find Ro
!-----
call sprsax(sata,ijata,x,ax,n,count)

do i = 1,n
  r(i) = atb(i) - ax(i)
end do

!-----
! Find Do
!-----
do i = 1,n
  d(i) = pre(i)*r(i)
end do

!-----
! Find D-New
!-----
dnew = 0.0D0
do i = 1,n
  dnew = dnew + r(i)*d(i)
end do

del = dnew

do

if(iter >= maxit.OR.dnew <= (tol*tol*del)) exit

!-----
! Find Q(i+1)
!-----

call sprsax(sata,ijata,d,q,n,count)

!-----
! Find Alpha
!-----
bottom = 0.0D0
do i = 1,n
  bottom = bottom + d(i)*q(i)
end do

alpha = dnew/bottom

do i = 1,n
  x(i) = x(i) + alpha*d(i)
end do

!-----
! Find R(i+1)
!-----
check = mod(iter,50)

if(check == 0.) then

call sprsax(sata,ijata,x,ax,n,count)

do i = 1,n
  r(i) = atb(i) - ax(i)
end do

```

```

else
    do i = 1,n
        r(i) = r(i) - alpha*q(i)
    end do

endif

!-----
! Find S(i+1)
!-----
do i = 1,n
    s(i) = pre(i)*r(i)
end do

!-----
! Find beta
!-----
dold = dnew
dnew = 0.

do i = 1,n
    dnew = dnew + r(i)*s(i)
end do

beta = dnew/dold

do i = 1,n
    d(i) = s(i) + beta*d(i)
end do

iter = iter + 1

end do

end subroutine unsympcgt

!-----

subroutine sympcg(a, b, x, n) ! Solve Pressure ; P

implicit none

integer(4)                :: maxit, iter, count
integer(4)                :: i, j, n
real(8)                  :: del, dnew, tol, bottom
real(8)                  :: alpha, dold, beta, check, thresh
real(8), dimension(n,n) :: a
real(8), dimension(n)   :: q, r, d, pre, s, x, b, ax

integer(4), allocatable, dimension(:) :: ija
real(8), allocatable, dimension(:)   :: sa

    iter = 1
    tol = 1.d-6
    maxit = 50000
    thresh = 1.e-10

    count = 100000

!do i = 1,n
! do j = 1,n
! if(i/=j) then
! if(a(i,j) /= 0.) count = count + 1
! end if
! end do
!end do

!count = count + (n+1)

allocate(sa(count), ija(count))

!-----

```

```

! Construct precondition matrix P (but this will find P inverse)
!-----
do i = 1,n
  pre(i) = 1./abs(a(i,i))
end do

!-----
! Convert square matrix into row-indexed sparse
!-----
call sprsin(a,n,n,thresh,count,sa,ija)

!-----
! Find Ro
!-----

call sprsax(sa,ija,x,ax,n,count)

do i = 1,n
  r(i) = b(i) - ax(i)
end do

!-----
! Find Do
!-----
do i = 1,n
  d(i) = pre(i)*r(i)
end do

!-----
! Find D-New
!-----
dnew = 0.0D0
do i = 1,n
  dnew = dnew + r(i)*d(i)
end do

del = dnew

do

if(iter >= maxit.OR.dnew <= (tol*tol*del)) exit

!-----
! Find Q(i+1)
!-----

call sprsax(sa,ija,d,q,n,count)

!-----
! Find Alpha
!-----
bottom = 0.0D0
do i = 1,n
  bottom = bottom + d(i)*q(i)
end do

alpha = dnew/bottom

do i = 1,n
  x(i) = x(i) + alpha*d(i)
end do

!-----
! Find R(i+1)
!-----
check = mod(iter,50)

if(check == 0.0D0) then

call sprsax(sa,ija,x,ax,n,count)

do i = 1,n
  r(i) = b(i) - ax(i)
end do

```

```

else
    do i = 1,n
        r(i) = r(i) - alpha*q(i)
    end do

endif

!-----
! Find S(i+1)
!-----
do i = 1,n
    s(i) = pre(i)*r(i)
end do

!-----
! Find beta
!-----
dold = dnew
dnew = 0.0D0

do i = 1,n
    dnew = dnew + r(i)*s(i)
end do

beta = dnew/dold

do i = 1,n
    d(i) = s(i) + beta*d(i)
end do

iter = iter + 1

end do

end subroutine sympcg

!-----

subroutine sprsin(a,n,np,thresh,nmax,sa,ija)

implicit none

integer(4)          :: n, nmax, np
integer(4)          :: i, j, k
integer(4), dimension(nmax) :: ija

real(8)            :: thresh
real(8), dimension(np,np) :: a
real(8), dimension(nmax) :: sa

do j = 1,n
    sa(j) = a(j,j)
enddo
ija(1) = n+2
k = n+1
do i = 1,n
    do j = 1,n
        if (abs(a(i,j)) >= thresh) then
            if (i /= j) then
                k = k+1
                if (k > nmax) pause 'nmax too small in sprsin'
                sa(k) = a(i,j)
                ija(k) = j
            endif
        endif
    enddo
    ija(i+1) = k+1
enddo

END subroutine sprsin

!-----

```

```

subroutine sprsax(sa,ija,x,b,n,nmax)

!
! Multiply a matrix in sparse storage by a vector x(1:n)
! and the result is a vector b(1:n)
!

implicit none

integer(4)                :: n, nmax, i, k
integer(4), dimension(nmax) :: ija

real(8), dimension(n)     :: b, x
real(8), dimension(nmax)  :: sa

if (ija(1) /= n+2) pause 'mismatched vector and matrix in sprsax'
do i = 1,n
  b(i) = sa(i) * x(i)
  do k = ija(i),ija(i+1)-1
    b(i) = b(i) + sa(k) * x(ija(k))
  enddo
enddo

END subroutine sprsax

!-----

subroutine sprstm(sa,ija,sb,ijb,sc,ijc,thresh,nmax)

!
! Multiply At*A
!
! Input for sa must be At
! Input for sb must be At
! output of sc will be At*A
!

implicit none

integer(4)                :: n, nmax
integer(4)                :: i,ijma,ijmb,j,k,ma,mb,mbb

integer(4), dimension(nmax) :: ija, ijb, ijc
real(8)  , dimension(nmax)  :: sa, sb, sc

real(8)                :: sum, thresh

if(ija(1) /= ijb(1)) pause 'sprstm sizes do not match'
k = ija(1)
ijc(1) = k
do i = 1,ija(1)-2
  do j = 1,ijb(1)-2
    if(i==j) then
      sum = sa(i)*sb(j)
    else
      sum = 0.d0
    endif
    mb = ijb(j)
    do ma = ija(i),ija(i+1)-1
      ijma = ija(ma)
      if(ijma == j) then
        sum = sum + sa(ma)*sb(j)
      else
        2      if(mb < ijb(j+1)) then
              ijmb = ijb(mb)
              if(ijmb == i) then
                sum = sum + sa(i)*sb(mb)
                mb = mb + 1
                goto 2
              else if(ijmb < ijma) then
                mb = mb + 1
                goto 2
              else if(ijmb == ijma) then

```



```

        sum = sum + sa(ma)*sb(mb)
        mb = mb + 1
        goto 2
    endif
endif
endif
enddo

do mbb = mb,ijb(j+1)-1
    if(ijb(mbb) == i) then
        sum = sum + sa(i)*sb(mbb)
    endif
enddo

if(i == j) then
    sc(i) = sum
else if(abs(sum) > thresh) then
    if(k > nmax) pause 'sprstm: nmax to small'
    sc(k) = sum
    ijc(k) = j
    k = k + 1
endif
enddo
ijc(i+1) = k
enddo

end subroutine sprstm

!-----

subroutine sprsint(a,n,np,thresh,nmax,sa,ija)

implicit none

integer(4)                :: n, nmax, np
integer(4)                :: i, j, k
integer(4), dimension(nmax) :: ija

real(8)                  :: thresh
real(8), dimension(np,np) :: a
real(8), dimension(nmax) :: sa

do j = 1,n
    sa(j) = a(j,j)
enddo
ija(1) = n+2
k = n+1
do j = 1,n
    do i = 1,n
        if (abs(a(i,j)) >= thresh) then
            if (i /= j) then
                k = k+1
                if (k > nmax) pause 'nmax too small in sprsin'
                sa(k) = a(i,j)
                ija(k) = i
            endif
        endif
    enddo
    ija(j+1) = k+1
enddo

END subroutine sprsint

!-----

end module solve

```

**ภาคผนวก ข (Appendix B)**  
**รายละเอียดของโปรแกรมวิเคราะห์ปัญหาความเค้น**  
**อันเนื่องมาจากอุณหภูมิ**

```

MODULE SOLID

IMPLICIT NONE

CHARACTER(len=20)          :: name1

INTEGER(4), ALLOCATABLE, DIMENSION(:,:) :: intmat      ! Node connection on element
INTEGER(4), ALLOCATABLE, DIMENSION(:)   :: ibcx, ibcy  ! boundary condition for
variable
INTEGER(4), DIMENSION(20)              :: text          ! buffer for reading

REAL(8), ALLOCATABLE, DIMENSION(:,:)    :: coord      !
Coordinate of each node
REAL(8), ALLOCATABLE, DIMENSION(:)      :: p, pnw, displx, displx, vms
REAL(8), ALLOCATABLE, DIMENSION(:)      :: sysf, sysfb, dig, temp, sxx, syy, sxy, one

INTEGER(4)                             :: neq, npoi, nelem, iter, ierror, ip, ie
INTEGER(4)                             :: i, j, k, ii, jj, kk
integer(4)                              :: ll, mm, nn
INTEGER(4)                             :: nlines, iline, nstep, ic, res, l
integer(4)                              :: nside, nboun

REAL(8)                                 :: xg1, xg2, xg3, yg1, yg2, yg3, theta
REAL(8)                                 :: b1, b2, b3, c1, c2, c3
REAL(8)                                 :: uj, vj, zero
REAL(8)                                 :: area, x, y, sum
REAL(8)                                 :: elas, pr, tref, alpha, thick, lfa
!real(8)                                :: c11, c12, c21, c22, c33
INTEGER(4)                              :: auto

REAL(8), DIMENSION(6,6)                 :: kele
REAL(8), DIMENSION(3,3)                 :: cmat
REAL(8), DIMENSION(3,6)                 :: bmat
REAL(8), DIMENSION(3)                   :: lmat
REAL(8), DIMENSION(6)                   :: r, fbar, rsur, rtemp

integer(4)                              :: tnode, stype
integer(4), allocatable, dimension(:,:) :: number
integer(4), allocatable, dimension(:,:,:) :: acheck

real(8), allocatable, dimension(:,:)    :: sysk

CONTAINS

!-----
SUBROUTINE MAIN()

use solve
IMPLICIT NONE

real(8)  :: error
integer(4) :: niter, iter

    call READ_INPUT()
    call CST()
    call sympcg(sysk, sysf, p, neq)
    call STRESS()
    call WRITE_OUTPUT()

END SUBROUTINE MAIN

```

```

=====
SUBROUTINE READ_INPUT()

!use solid
IMPLICIT NONE

!-----
! auto = 1 ==> for auto read input file
! auto = 0 ==> for manual read input file
!-----

auto = 1

!-----
! For Read Input (Auto)
!-----

IF (auto == 1) THEN

    name1 = 'K10_E5_007540.s'
    OPEN(UNIT=7, FILE=name1, STATUS='OLD', ACTION='READ', IOSTAT=ierror)

ENDIF

!-----
! For Read Input
!-----

IF (auto == 0) THEN

WRITE(6,10)
10 FORMAT(/, ' PLEASE ENTER INPUT FILE NAME:',/)
READ(5, '(A)') name1

OPEN(UNIT=7, FILE=name1, STATUS='OLD', ACTION='READ', IOSTAT=ierror)

ENDIF

!-----
! Read input data
!-----

READ(7,*) nlines

DO iline = 1,nlines
    READ(7,1) text
    1 FORMAT(20A4)
END DO

READ(7,1) text
READ(7,*) tnode, nelem, nboun, stype

npoi = tnode
neq = 2*npoi

ALLOCATE(intmat(nelem,3), coord(npoi,2))
ALLOCATE( ibcx(npoi), ibcy(npoi), displx(npoi), disp1y(npoi), temp(tnode))
ALLOCATE( sxx(npoi), syy(npoi), sxy(npoi), one(npoi), vms(npoi))
ALLOCATE( p(neq), pnw(neq), sysf(neq), sysfb(neq), dig(neq))
ALLOCATE( number(tnode,tnode), acheck(tnode,tnode,2))
allocate( sysk(neq,neq))

READ(7,1) text
READ(7,*) elas, pr, tref, alpha, thick
READ(7,1) text
DO ip = 1,tnode

    READ(7,*) i, ibcx(i), ibcy(i), (coord(i,k), k=1,2), displx(i), disp1y(i), sxx(i),
syy(i), temp(i)
    IF(i /= ip) WRITE(6,40) ip
    40 FORMAT(/, ' NODE NO.', I5, ' IN DATA FILE IS MISSING')
    IF(i /= ip) STOP
END DO

```

```

READ(7,1) TEXT
DO ie = 1,nelem
  READ(7,*) i, (intmat(i,j), j=1,3)
  IF(i /= ie) WRITE(6,50) ie
  50 FORMAT(/, ' ELEMENT NO.', I5, ' IN DATA FILE IS MISSING')
  IF(i /= ie) STOP
END DO

CLOSE(UNIT=7, STATUS='keep')

!-----

WRITE(6,60) tnode, nelem
60 FORMAT(/, ' THE FINITE ELEMENT MODEL CONSISTS OF :',/, &
          '     NUMBER OF TIP NODES           =', I6, '/', &
          '     NUMBER OF ELEMENTS           =', I6, '/')

END SUBROUTINE READ_INPUT

!=====

SUBROUTINE CST()

IMPLICIT NONE
real(8), dimension(6)      :: tele, ibce
integer(4)                 :: ieq, ir

  sysf = 0.
  sysk = 0.

!
! Loop over the number of elements:
!

const: DO ie = 1,nelem

!
! Find element local coordinates:
!

ii = intmat(ie,1)
jj = intmat(ie,2)
kk = intmat(ie,3)
!ll = intmat(ie,4)
!mm = intmat(ie,5)
!nn = intmat(ie,6)

xg1 = coord(ii,1)
xg2 = coord(jj,1)
xg3 = coord(kk,1)

yg1 = coord(ii,2)
yg2 = coord(jj,2)
yg3 = coord(kk,2)

area = 0.5*(xg2*(yg3-yg1)+xg1*(yg2-yg3)+xg3*(yg1-yg2))

b1 = (yg2 - yg3)
b2 = (yg3 - yg1)
b3 = (yg1 - yg2)

c1 = (xg3 - xg2)
c2 = (xg1 - xg3)
c3 = (xg2 - xg1)

!
! Setup & clear matrices:
!

  kele = 0.
  ibce = 0.
  rsur = 0.
  rtemp = 0.
  r = 0.

```

```

if (ibcx(ii) == 1) ibce(1) = 1
if (ibcx(jj) == 1) ibce(2) = 1
if (ibcx(kk) == 1) ibce(3) = 1

if (ibcy(ii) == 1) ibce(4) = 1
if (ibcy(jj) == 1) ibce(5) = 1
if (ibcy(kk) == 1) ibce(6) = 1

!-----
! Set the value for
! the coefficient of diffusion term
!-----

call CnK_ELE()

!-----
! Compute all loads
!-----

call ALL_LOAD()

!-----
! Assemble element
!-----

call asmp()

!-----

END DO const

!-----

call applbc()

!-----

END SUBROUTINE CST

!=====

subroutine applbc()
!use solid
implicit none
integer(4) :: ieq, ir !, ic

outer1: do ieq = 1,npoi

    if(ibcx(ieq) == 0) cycle outer1

    inner1: do ir = 1,neq
!       if(ir == ieq) cycle inner1
        sysk(ir,2*ieq-1) = 0.0D0
    enddo inner1

    do ic = 1,neq
        sysk(2*ieq-1,ic) = 0.0D0
    enddo

    sysk(2*ieq-1,2*ieq-1) = 1.0D0
    sysf(2*ieq-1) = 0.0D0

    enddo outer1

outer: do ieq = 1,npoi

    if(ibcy(ieq) == 0) cycle outer

    inner: do ir = 1,neq

```

```

!           if(ir == ieq) cycle inner
             sysk(ir,2*ieq) = 0.0D0
        enddo inner

        do ic = 1,neq
             sysk(2*ieq,ic) = 0.0D0
        enddo

        sysk(2*ieq,2*ieq) = 1.0D0
             sysf(2*ieq)      = 0.0D0

        enddo outer

end subroutine applbc

!=====
SUBROUTINE CnK_ELE()

!use solid
IMPLICIT NONE

real(8), dimension(3,6)      :: cb

cmat = 0.0D0
bmat = 0.0D0
lmat = 0.0D0
kele = 0.0D0

!-----
! Setup Metrix C      : stype = 0 (plane stress)
!                     : stype = 1 (plane strain)
!-----

if (stype == 0) then
    cmat(1,1) =      elas / (1 - pr*pr)
    cmat(1,2) =      pr * elas / (1 - pr*pr)
    cmat(3,3) = (1 - pr) * elas / (1 - pr*pr)/2.0D0
    lmat(1)   = alpha
endif

if (stype == 1) then
    cmat(1,1) =      (1 - pr) * elas / (1 + pr) / (1 - 2.0D0*pr)
    cmat(1,2) =      pr * elas / (1 + pr) / (1 - 2.0D0*pr)
    cmat(3,3) = (1 - 2.0D0*pr) * elas / (1 + pr) / (1 - 2.0D0*pr)/2.0D0
    lmat(1)   = alpha*(1+pr)
endif

cmat(2,2) = cmat(1,1)
cmat(2,1) = cmat(1,2)
lmat(2)   = lmat(1)

!-----
! Setup Metrix B
!-----

bmat(1,1) = b1
bmat(1,3) = b2
bmat(1,5) = b3

bmat(2,2) = c1
bmat(2,4) = c2
bmat(2,6) = c3

bmat(3,1) = c1
bmat(3,2) = b1
bmat(3,3) = c2
bmat(3,4) = b2
bmat(3,5) = c3
bmat(3,6) = b3

bmat = bmat/area/2.0D0

```

```

!-----
! Setup Metrix Kele
!-----

kele = thick*area*(matmul(transpose(bmat),matmul(cmat,bmat)))

!cb = matmul(cmat,bmat)

!kele = thick*area*(matmul(transpose(bmat),cb))

!-----
!kele = 1.0D0
!-----

END SUBROUTINE CnK_ELE

!=====

SUBROUTINE ASMP()

!use solid
IMPLICIT NONE

!
! Assembling system load vector
!
! Contribution of coefficients associated with psi
!

DO i = 1,3

  do j = 1,3

    ii = 2*intmat(ie,i)-1
    jj = 2*intmat(ie,j)-1
    ll = 2*i-1
    mm = 2*j-1

    sysk(ii , jj ) = sysk(ii , jj ) + kele(ll , mm )
    sysk(ii+1 , jj ) = sysk(ii+1 , jj ) + kele(ll+1 , mm )
    sysk(ii , jj+1) = sysk(ii , jj+1) + kele(ll , mm+1)
    sysk(ii+1 , jj+1) = sysk(ii+1 , jj+1) + kele(ll+1 , mm+1)

  enddo

  sysf(ii ) = sysf(ii ) + r(ll )
  sysf(ii+1) = sysf(ii+1) + r(ll+1)

END DO

END SUBROUTINE ASMP

!=====

SUBROUTINE WRITE_OUTPUT()

!use solid
IMPLICIT NONE

INTEGER(4)          :: leng

leng = len_trim(name1) - 2

OPEN(UNIT=16, FILE=name1(1:leng)//'.plt',STATUS='unknown',IOSTAT=ierror)

  WRITE(16,220)
  220 FORMAT ( 'VARIABLES = "X-CO","Y-CO",
"i","p(i)","p(i+1)","sxx(i)","syy(i)","vms(i)","temp(i)"' )

  WRITE(16,240) npoi, nelelem
  240 FORMAT('ZONE N=', I6,',E=', I6, ',F=FEPOINT,ET=TRIANGLE')

DO i = 1,npoi

```

```

vms(i) = sqrt((sxx(i)-syy(i))*(sxx(i)-syy(i)) + sxx(i)*sxx(i) + syy(i)*syy(i) +
6*sxy(i)*sxy(i)) / sqrt(2.)

WRITE(16,260) (coord(i,k),k=1,2), i, p(2*i-1), p(2*i), sxx(i), syy(i), vms(i),
temp(i)
260 FORMAT(2F12.6,2X, I6,2X, 6E16.6)
END DO

DO i = 1,nelem
WRITE(16,280) (intmat(i,j),j=1,3)
280 FORMAT(3I6)
END DO

CLOSE(UNIT=16, STATUS='keep')

!-----
END SUBROUTINE WRITE_OUTPUT

!=====
SUBROUTINE ALL_LOAD()

!use solid
IMPLICIT NONE

real(8)          :: dx, dy, dl, SL, tavg, Q

rsur = 0.0D0
r     = 0.0D0

!-----
! surface load
!-----
if ((sxx(ii) /= 0.) .and. (sxx(jj) /= 0.)) then
dx = xg2 - xg1
dy = yg2 - yg1
dl = sqrt(dx*dx + dy*dy)

SL = (sxx(ii) + sxx(jj))/2.0D0

rsur(1) = SL*thick*dl/2.0D0
rsur(3) = SL*thick*dl/2.0D0
endif

if ((sxx(jj) /= 0.) .and. (sxx(kk) /= 0.)) then
dx = xg3 - xg2
dy = yg3 - yg2
dl = sqrt(dx*dx + dy*dy)

SL = (sxx(jj) + sxx(kk))/2.0D0

rsur(3) = SL*thick*dl/2.0D0
rsur(5) = SL*thick*dl/2.0D0
endif

if ((sxx(kk) /= 0.) .and. (sxx(ii) /= 0.)) then
dx = xg3 - xg1
dy = yg3 - yg1
dl = sqrt(dx*dx + dy*dy)

SL = (sxx(kk) + sxx(ii))/2.0D0

rsur(1) = SL*thick*dl/2.0D0
rsur(5) = SL*thick*dl/2.0D0
endif

if ((syy(ii) /= 0.) .and. (syy(jj) /= 0.)) then
dx = xg2 - xg1
dy = yg2 - yg1
dl = sqrt(dx*dx + dy*dy)

SL = (syy(ii) + syy(jj))/2.0D0

```



```

        rsur(2) = SL*thick*dl/2.0D0
        rsur(4) = SL*thick*dl/2.0D0
    endif

    if ((syy(jj) /= 0.) .and. (syy(kk) /= 0.)) then
        dx = xg3 - xg2
        dy = yg3 - yg2
        dl = sqrt(dx*dx + dy*dy)

        SL = (syy(jj) + syy(kk))/2.0D0

        rsur(4) = SL*thick*dl/2.0D0
        rsur(6) = SL*thick*dl/2.0D0
    endif

    if ((syy(kk) /= 0.) .and. (syy(ii) /= 0.)) then
        dx = xg3 - xg1
        dy = yg3 - yg1
        dl = sqrt(dx*dx + dy*dy)

        SL = (syy(kk) + syy(ii))/2.0D0

        rsur(2) = SL*thick*dl/2.0D0
        rsur(6) = SL*thick*dl/2.0D0
    endif

!-----
! average element temperature
!-----

tavg = (temp(ii) + temp(jj) + temp(kk))/3.0D0

Q = tavg - tref

!-----
! temp load
!-----

!rtemp = matmul(transpose(bmat),matmul(cmat,lmat))*Q*thick*area
rtemp = area*thick*Q*(matmul(matmul(transpose(bmat),cmat),lmat))

!-----
! total load = surface load + temp load
!-----

r = rsur + rtemp

!-----
!r = 1
!-----

END SUBROUTINE ALL_LOAD

!=====
SUBROUTINE STRESS()

IMPLICIT NONE

real(8) :: tavg
real(8), dimension(3) :: st
real(8), dimension(6) :: tele
real(8), dimension(3,6) :: b

cmat = 0.0D0
bmat = 0.0D0
lmat = 0.0D0
kele = 0.0D0
sxx = 0.0D0
syy = 0.0D0
sxy = 0.0D0
one = 0.0D0

```

```

do ie=1,nelem

  ii = intmat(ie,1)
  jj = intmat(ie,2)
  kk = intmat(ie,3)

  xg1 = coord(ii,1)
  xg2 = coord(jj,1)
  xg3 = coord(kk,1)

  yg1 = coord(ii,2)
  yg2 = coord(jj,2)
  yg3 = coord(kk,2)

  area = 0.5*(xg2*(yg3-yg1)+xg1*(yg2-yg3)+xg3*(yg1-yg2))

  b1 = (yg2 - yg3)
  b2 = (yg3 - yg1)
  b3 = (yg1 - yg2)

  c1 = (xg3 - xg2)
  c2 = (xg1 - xg3)
  c3 = (xg2 - xg1)

  !-----
  ! Setup Elasticity Metrix C      : stype = 0 (plane stress)
  !                               : stype = 1 (plane strain)
  !-----

  if (stype == 0) then
    cmat(1,1) =      elas / (1 - pr*pr)
    cmat(1,2) =      pr * elas / (1 - pr*pr)
    cmat(3,3) = (1 - pr) * elas / (1 - pr*pr)/2.0D0
    lmat(1)   = alpha
  endif

  if (stype == 1) then
    cmat(1,1) = (1 - pr) * elas / (1 + pr) / (1 - 2.0D0*pr)
    cmat(1,2) =      pr * elas / (1 + pr) / (1 - 2.0D0*pr)
    cmat(3,3) = (1 - 2.0D0*pr) * elas / (1 + pr) / (1 - 2.0D0*pr)/2.0D0
    lmat(1)   = alpha*(1+pr)
  endif

  cmat(2,2) = cmat(1,1)
  cmat(2,1) = cmat(1,2)
  lmat(2)   = lmat(1)

  !-----
  ! Setup Metrix B
  !-----

  bmat(1,1) = b1
  bmat(1,3) = b2
  bmat(1,5) = b3

  bmat(2,2) = c1
  bmat(2,4) = c2
  bmat(2,6) = c3

  bmat(3,1) = c1
  bmat(3,2) = b1
  bmat(3,3) = c2
  bmat(3,4) = b2
  bmat(3,5) = c3
  bmat(3,6) = b3

  bmat = bmat/area/2.0D0

  tele(1) = p(2*ii-1)
  tele(2) = p(2*ii)
  tele(3) = p(2*jj-1)
  tele(4) = p(2*jj)
  tele(5) = p(2*kk-1)

```

```

tele(6) = p(2*kk)

tavg = (temp(ii) + temp(jj) + temp(kk))/3.0D0

st = 0.0D0

st = matmul(cmat,(matmul(bmat,tele) - lmat*(tavg - tref)))

sxx(ii) = sxx(ii) + st(1)
sxx(jj) = sxx(jj) + st(1)
sxx(kk) = sxx(kk) + st(1)

syy(ii) = syy(ii) + st(2)
syy(jj) = syy(jj) + st(2)
syy(kk) = syy(kk) + st(2)

sxy(ii) = sxy(ii) + st(3)
sxy(jj) = sxy(jj) + st(3)
sxy(kk) = sxy(kk) + st(3)

one(ii) = one(ii) + 1.
one(jj) = one(jj) + 1.
one(kk) = one(kk) + 1.

enddo

do ip=1,tnode
  if (one(ip) == 0.) write(6,11) ip
  11 format(' *** warning *** No stress contribution at node', i5)
  if (one(ip) == 0.) one(ip) = 1.0D0
  sxx(ip) = sxx(ip) / one(ip)
  syy(ip) = syy(ip) / one(ip)
  sxy(ip) = sxy(ip) / one(ip)
enddo

END SUBROUTINE STRESS

!=====
END MODULE SOLID

!-----

PROGRAM SOLID_STRESS

use SOLID

IMPLICIT NONE

CALL MAIN()

stop

END PROGRAM SOLID_STRESS

!-----

```

```

Module solve

implicit none

contains

!
! Subroutine preconditioned conjugate gradient method
!

subroutine unsympcg(a, b, x, n)

implicit none

integer(4)          :: maxit, iter, count
integer(4)          :: i, j, n
real(8)             :: del, dnew, tol, bottom
real(8)             :: alpha, dold, beta, check, thresh
real(8), dimension(n,n) :: a
real(8), dimension(n)  :: q, r, d, pre, s, x, b, ax, atb

integer(4), allocatable, dimension(:) :: ija, ijat, ijata
real(8), allocatable, dimension(:) :: sa, sat, sata

    iter = 1
    tol = 1.d-6
    maxit = 50000
    thresh = 1.e-10

    count = 1000000

    write(6,*) count

    allocate( sa(count), ija(count))
    allocate( sat(count), ijat(count))
    allocate(sata(count), ijata(count))

!-----
! Convert square matrix into row-indexed sparse
!-----
    call sprsin( a,n,n,thresh,count, sa, ija)
    call sprsint(a,n,n,thresh,count,sat,ijat)

!-----
! Compute At*A
!-----
    call sprstm(sat,ijat,sat,ijat,sata,ijata,thresh,count)

!-----
! Compute At*b
!-----
    call sprsax(sat,ijat,b,atb,n,count)

!-----
! Construct precondition matrix P (but this will find P inverse)
!-----
    do i = 1,n
        pre(i) = 1./abs(sata(i))
    end do

!-----
! Find Ro
!-----
    call sprsax(sata,ijata,x,ax,n,count)
    do i = 1,n
        r(i) = atb(i) - ax(i)
    end do

!-----
! Find Do
!-----
    do i = 1,n
        d(i) = pre(i)*r(i)
    end do

```

```

!-----
! Find D-New
!-----
dnew = 0.
do i = 1,n
  dnew = dnew + r(i)*d(i)
end do

del = dnew

Do
  if(iter >= maxit.OR.dnew <= (tol*tol*del)) exit

!-----
! Find Q(i+1)
!-----
  call sprsax(sata,ijata,d,q,n,count)

!-----
! Find Alpha
!-----
  bottom = 0.
  do i = 1,n
    bottom = bottom + d(i)*q(i)
  end do

  alpha = dnew/bottom
  do i = 1,n
    x(i) = x(i) + alpha*d(i)
  end do

!-----
! Find R(i+1)
!-----
  check = mod(iter,50)
  if(check == 0.) then
    call sprsax(sata,ijata,x,ax,n,count)
    do i = 1,n
      r(i) = atb(i) - ax(i)
    end do
  else
    do i = 1,n
      r(i) = r(i) - alpha*q(i)
    end do
  endif

!-----
! Find S(i+1)
!-----
  do i = 1,n
    s(i) = pre(i)*r(i)
  end do

!-----
! Find beta
!-----
  dold = dnew
  dnew = 0.
  do i = 1,n
    dnew = dnew + r(i)*s(i)
  end do

  beta = dnew/dold
  do i = 1,n
    d(i) = s(i) + beta*d(i)
  end do

  iter = iter + 1

End do

end subroutine unsympcg

```

```

!-----
subroutine sympcg(a, b, x, n)

implicit none

integer(4)          :: maxit, iter, count
integer(4)          :: i, j, n
real(8)            :: del, dnew, tol, bottom
real(8)            :: alpha, dold, beta, check, thresh
real(8), dimension(n,n) :: a
real(8), dimension(n)  :: q, r, d, pre, s, x, b, ax

integer(4), allocatable, dimension(:) :: ija
real(8),   allocatable, dimension(:) :: sa

  iter = 1
  tol = 1.d-6
  maxit = 50000
  thresh = 1.e-10

  count = 10000000

  allocate(sa(count), ija(count))

!-----
! Construct precondition matrix P (but this will find P inverse)
!-----
do i = 1,n
  pre(i) = 1./abs(a(i,i))
end do

!-----
! Convert square matrix into row-indexed sparse
!-----
call sprsin(a,n,n,thresh,count,sa,ija)

!-----
! Find Ro
!-----
call sprsax(sa,ija,x,ax,n,count)
do i = 1,n
  r(i) = b(i) - ax(i)
end do

!-----
! Find Do
!-----
do i = 1,n
  d(i) = pre(i)*r(i)
end do

!-----
! Find D-New
!-----
dnew = 0.
do i = 1,n
  dnew = dnew + r(i)*d(i)
end do

del = dnew

Do
  if(iter >= maxit.OR.dnew <= (tol*tol*del)) exit
!-----
! Find Q(i+1)
!-----
  call sprsax(sa,ija,d,q,n,count)

!-----
! Find Alpha
!-----
  bottom = 0.

```

```

do i = 1,n
  bottom = bottom + d(i)*q(i)
end do

alpha = dnew/bottom
do i = 1,n
  x(i) = x(i) + alpha*d(i)
end do

!-----
! Find R(i+1)
!-----
  check = mod(iter,50)
  if(check == 0.) then
    call sprsax(sa,ija,x,ax,n,count)
    do i = 1,n
      r(i) = b(i) - ax(i)
    end do
  else
    do i = 1,n
      r(i) = r(i) - alpha*q(i)
    end do
  endif

!-----
! Find S(i+1)
!-----
do i = 1,n
  s(i) = pre(i)*r(i)
end do

!-----
! Find beta
!-----
dold = dnew
dnew = 0.
do i = 1,n
  dnew = dnew + r(i)*s(i)
end do

beta = dnew/dold
do i = 1,n
  d(i) = s(i) + beta*d(i)
end do

iter = iter + 1

End do

end subroutine sympcg

!-----
subroutine sprsin(a,n,np,thresh,nmax,sa,ija)
implicit none

integer(4)          :: n, nmax, np
integer(4)          :: i, j, k
integer(4), dimension(nmax) :: ija

real(8)            :: thresh
real(8), dimension(np,np) :: a
real(8), dimension(nmax) :: sa

do j = 1,n
  sa(j) = a(j,j)
enddo

ija(1) = n+2
k = n+1
do i = 1,n

```

```

do j = 1,n
  if (abs(a(i,j)) >= thresh) then
    if (i /= j) then
      k = k+1
      if (k > nmax) then
        write(6,*) k, nmax !, 'nmax too small in sprsin'
        pause 'nmax too small in sprsin'
      endif
      sa(k) = a(i,j)
      ija(k) = j
    endif
  endif
enddo
ija(i+1) = k+1
enddo

END subroutine sprsin

!-----
subroutine sprsax(sa,ija,x,b,n,nmax)

! Multiply a matrix in sparse storage by a vector x(1:n)
! and the result is a vector b(1:n)

implicit none

integer(4)                :: n, nmax, i, k
integer(4), dimension(nmax) :: ija

real(8), dimension(n)     :: b, x
real(8), dimension(nmax)  :: sa

if (ija(1) /= n+2) pause 'mismatched vector and matrix in sprsax'
do i = 1,n
  b(i) = sa(i) * x(i)
  do k = ija(i),ija(i+1)-1
    b(i) = b(i) + sa(k) * x(ija(k))
  enddo
enddo

END subroutine sprsax

!-----
subroutine sprstm(sa,ija,sb,ijb,sc,ijc,thresh,nmax)

!
! Multiply At*A
!
! Input for sa must be At
! Input for sb must be At
! output of sc will be At*A
!

implicit none

integer(4)                :: n, nmax
integer(4)                :: i,ijma,ijmb,j,k,ma,mb,mbb

integer(4), dimension(nmax) :: ija, ijb, ijc
real(8) , dimension(nmax)  :: sa, sb, sc

real(8)                   :: sum, thresh

if(ija(1) /= ijb(1)) pause 'sprstm sizes do not match'
k = ija(1)
ijc(1) = k
do i = 1,ija(1)-2
  do j = 1,ijb(1)-2
    if(i==j) then
      sum = sa(i)*sb(j)
    else
      sum = 0.d0
    endif
  enddo
enddo

```



```

endif
mb = ijb(j)
do ma = ija(i), ija(i+1)-1
  ijma = ija(ma)
  if(ijma == j) then
    sum = sum + sa(ma)*sb(j)
  else
2    if(mb < ijb(j+1)) then
      ijmb = ijb(mb)
      if(ijmb == i) then
        sum = sum + sa(i)*sb(mb)
        mb = mb + 1
        goto 2
      else if(ijmb < ijma) then
        mb = mb + 1
        goto 2
      else if(ijmb == ijma) then
        sum = sum + sa(ma)*sb(mb)
        mb = mb + 1
        goto 2
      endif
    endif
  endif
enddo

do mbb = mb, ijb(j+1)-1
  if(ijb(mbb) == i) then
    sum = sum + sa(i)*sb(mbb)
  endif
enddo

if(i == j) then
  sc(i) = sum
else if(abs(sum) > thresh) then
  if(k > nmax) pause 'sprstm: nmax to small'
  sc(k) = sum
  ijc(k) = j
  k = k + 1
endif
enddo
ijc(i+1) = k
enddo

end subroutine sprstm

!-----

subroutine sprsint(a,n,np,thresh,nmax,sa,ija)

implicit none

integer(4)          :: n, nmax, np
integer(4)          :: i, j, k
integer(4), dimension(nmax) :: ija

real(8)            :: thresh
real(8), dimension(np,np) :: a
real(8), dimension(nmax) :: sa

do j = 1,n
  sa(j) = a(j,j)
enddo
ija(1) = n+2
k = n+1
do j = 1,n
  do i = 1,n
    if (abs(a(i,j)) >= thresh) then
      if (i /= j) then
        k = k+1
        if (k > nmax) pause 'nmax too small in sprsin'
        sa(k) = a(i,j)
        ija(k) = i
      endif
    endif
  enddo
enddo

```

```
        enddo  
        ija(j+1) = k+1  
    enddo  
  
    END subroutine sprsint  
  
    !-----  
  
end module solve
```



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

## ประวัติผู้เขียนวิทยานิพนธ์

นายอิพงษ์ มาลาทิพย์ เกิดเมื่อวันที่ 12 เดือนพฤษภาคม พุทธศักราช 2521 จังหวัดสุพรรณบุรี สำเร็จการศึกษาปริญญาวิศวกรรมศาสตรบัณฑิตจากภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ เมื่อปีการศึกษา 2544 สำเร็จการศึกษาปริญญาวิศวกรรมศาสตรมหาบัณฑิตจากภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อปีการศึกษา 2547 และเข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรดุษฎีบัณฑิต ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อปีการศึกษา 2548



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย