บทที่ 5

คอมพิวเตอร์และโปรแกรมคอมพิวเตอร์

5.1 อุปกรณ์เครื่องคอมพิวเตอร์และซอฟท์แวร์ (Computer Hardward and Softward)

เนื่องจากการทำวิจัยนี้ เป็นงานที่เกี่ยวข้องกับงานประมวลผลและการคำนวณทางคณิต-ศาสตร์ จึงจำเป็นที่จะต้องใช้คอมพิวเตอร์เป็นเครื่องมือในการทำงานโดยทั้งนี้เครื่องคอมพิวเตอร์อุปกรณ์อื่น ๆ และโปรแกรมสนับสนุนต่าง ๆ ได้ใช้ของบริษัท ดิจิตอล อีควิปเมนต์ จำกัด (DEC) ทั้งหมด ซึ่งองค์ประกอบสำคัญของเครื่องคอมพิวเตอร์แยกออกได้เป็น 2 ส่วนคือ อุปกรณ์เครื่องคอมพิวเตอร์ (Computer Hardware) และซอฟท์แวร์คอมพิวเตอร์ (Software Computer)

5.1.1 อุปกรณ์เครื่องคอมพิวเตอร์ (Computer Hardware)

ได้ใช้คอมพิวเตอร์รุ่น PDP 11/24 ซึ่งมีรายละเอียดต่าง ๆ ดังต่อไปนี้

1. หน่วยประมวลผลกลาง (CPU) KDF11 - UA ซึ่งประกอบด้วยส่วนสำคัญ ๆ คือ
   1. Control Processor Machine
   2. Memory Management Unit (MMU)
   3. Floating - Point Processor (FPP) Option
   4. Commercial Instruction Set Processor (CISP) Option
   5. Two Serial Line Interfaces (SLU)
   6. Line Frequency Interrupt Real Time Clook
   7. Extended UNIBUS (EUB)

2. หน่วยความจำ (Memory) ประกอบด้วยหน่วยความจำหลัก (Main Memory) ซึ่งมีความจุ 512 K-byte และหน่วยความจำรอง (Secondary Memory) คือ Disk Cartridge แบบ RL02K Type 5440 โดยมี Disk Drive แบบ Rl02 เป็นตัวขับเคลื่อนซึ่ง Disk Cartridge RL02K มีคุณสมบัติดังนี้

ขนาดความจุข้อมูล	10.4 Megabyte	ต่อ	1	Cartridge
จำนวนแผ่น Disk	1 แผ่น	ต่อ	1	Cartridge
จำนวนหน้า (Side) ที่ใช้ในการเก็บข้อมูล	2 หน้า	ต่อ	1	แผ่น

จำนวน Track	512	Track	ต่อ 1	หน้า
จำนวน Sector	40	Sector	ต่อ 1	Track
ความจุข้อมูล	256	Byte	ต่อ 1	Sector
อัตราเฉลี่ยในการหาข้อมูล	55	millisecond		

3. หน่วยรับและแสดงผลข้อมูล (Input - Output Unit) หน่วยรับข้อมูลที่ใช้คือ Keyboard ของ Terminal รุ่น VT220 ส่วนหน่วยแสดงผลข้อมูลคือ จอภาพ CRT ของ VT220 และ Line Printer รุ่น LETTER PRINTER 100

5.1.2 ซอฟท์แวร์คอมพิวเตอร์ (Computer Software)

ในที่นี้หมายถึงโปรแกรมและโปรแกรมย่อยต่าง ๆ ที่สนับสนุนให้เครื่องคอมพิวเตอร์ทำงานได้ เช่น โปรแกรมการควบคุมการทำงานของระบบโปรแกรมการสร้างแฟ้มข้อมูล หรือโปรแกรม Compiler เป็นต้น ซึ่งบริษัทผู้ผลิตคอมพิวเตอร์ได้เตรียมไว้ในเครื่อง เพื่อให้ผู้ใช้คอมพิวเตอร์สามารถใช้เครื่องได้อย่างมีประสิทธิภาพยิ่งขึ้น ซึ่งซอฟท์แวร์คอมพิวเตอร์ที่สำคัญสำหรับระบบของคอมพิวเตอร์ PDP 11/24 นี้ได้แก่

1. โปรแกรมควบคุมการปฏิบัติงาน (Operating System) คือ RSX - 11M VERSION 3.2 ซึ่งเป็นโปรแกรมควบคุมการปฏิบัติงานสำหรับระบบคอมพิวเตอร์ซึ่งผู้ใช้คอมพิวเตอร์สามารถทำงานได้พร้อม ๆ กันหลายคนด้วยเครื่องคอมพิวเตอร์เครื่องเดียวกัน (Muti - User System)

2. โปรแกรม File EDITOR คือ ED2 ใช้สำหรับการสร้าง SOURCE FILE

3. โปรแกรม Compiler คือ PDP 11/24 FORTRAN - 77 VERSION 4.0

4. โปรแกรม Task Builder ใช้สำหรับการ Link Object Module ให้เป็นโปรแกรมที่ใช้งานได้ซึ่ง Task Builder ที่ใช้คือ TKB

นอกจากนี้ยังมีโปรแกรมย่อยอื่น ๆ อีกหลายโปรแกรมที่มีอยู่ในคอมพิวเตอร์ที่สามารถเรียกมาทำงานได้ และโปรแกรมย่อยที่เรียกมาใช้ในงานวิจัยได้แก่ DSIMQ , DMINV และ DSINV

สำหรับภาษาระดับสูง (High Level Language) ภาษาที่มี Compiler ให้ใช้ได้ขณะนี้ได้แก่ FORTRAN - 77 , FORTRAN - 4 , BASIC - Plus 2

ซึ่งในการทำวิจัยนี้ใช้ภาษา FORTRAN - 77

5.2 โปรแกรมหาคำตอบสมการที่มีโครงสร้างเป็นแบนด์และแบนด์บอร์เดอร์

5.2.1 วัตถุประสงค์

วัตถุประสงค์ของโปรแกรมนี้ก็คือ ใช้หาคำตอบของสมการที่มีโครงสร้างเป็นแบนด์และแบนด์-บอร์เดอร์โดยให้สามารถแก้สมการขนาดใหญ่สุดและใช้เวลาในการคำนวณน้อยสุดเท่าที่จะเป็นไปได้ ซึ่งทั้งนี้โปรแกรมที่สร้างขึ้นมี 2 เวอร์ชัน (Version) คือ DSBRP และEDSBRP DSBRP มีวัตถุประสงค์เพื่อใช้หาคำตอบของสมการโดยใช้ได้เฉพาะหน่วยความจำที่มีอยู่ภายในเครื่องคอม-พิวเตอร์ แต่ EDSBRP มีวัตถุประสงค์เพื่อใช้หาคำตอบของสมการโดยใช้หน่วยความจำสำรองภายนอกเครื่องคอมพิวเตอร์เป็นหลัก ซึ่งทำให้สามารถหาคำตอบของสมการที่มีขนาดใหญ่ ๆ ได้ อย่างไรก็ดีโปรแกรมทั้งสองเวอร์ชันมีโครงสร้างของโปรแกรมเหมือนกันทุกประการ ดังนั้นในที่นี้จะกล่าวเฉพาะโครงสร้าง และรายละเอียดของโปรแกรม DSBRP เท่านั้น แต่วิธีการใช้โปรแกรมทั้ง 2 จะแสดงให้เห็นแยกจากกัน

5.2.2 โครงสร้างของโปรแกรม

กำหนดให้ DSBRP (Directed Solution by Recursive Partitioning) เป็นชื่อของโปรแกรมที่ใช้หาคำตอบของสมการที่มีรูปแบนด์และแบนด์บอร์เดอร์ เพราะว่าเมตริกซ์สัมประสิทธิ์ของสมการที่จะหาคำตอบเป็นเมตริกซ์สมมาตรแถบและสมมาตรแบนด์บอร์เดอร์ ซึ่งเมตริกซ์ทั้งสองชนิดนี้เป็นลักษณะพิเศษของเมตริกซ์สมมาตร เพื่อเป็นการเพิ่มความสามารถของโปรแกรมจึงได้สร้างโปรแกรมนี้ให้สามารถหาคำตอบของเมตริกซ์สมมาตรได้ด้วย

เมื่อพิจารณาถึงวิธีการหาคำตอบโดยตรงในบทที่ 3 พบว่าการหาคำตอบโดยวิธีนี้มี 2 ขั้นตอนคือ การทอนสมการ (Forward) และการแทนค่ากลับ (Backward) โดยที่การทอนสมการเป็นขั้นตอนการกำจัดค่าเวคเตอร์ X_1 ซึ่งเป็นผลทำให้ขนาดของสมการเล็กลง จนสามารถหาค่าคำตอบได้ส่วนการแทนค่ากลับเป็นขั้นตอนการแทนค่ากลับเพื่อหาเวคเตอร์ X_1 ไล่ย้อนกลับขึ้นไปจนกระทั่งได้ค่าของตัวไม่ทราบค่าทุกตัว

สมการที่ใช้ในขั้นตอนของการทอนสมการมีดังนี้

เมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์สมมาตร

$$\left[A_{22} - A_{21}A_{11}^{-1}A_{12} \right] X_2 = \left[U_2 - A_{21}A_{11}^{-1}U_1 \right]$$

เมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์สมมาตรแถบ

$$\begin{bmatrix} A_{22} - A_{21}A_{11}^{-1}A_{12} & A_{23} \\ A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} U_2 - A_{21}A_{11}^{-1}U_1 \\ U_3 \end{bmatrix}$$

เมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์สมมาตรแบนด์บอร์เดอร์

$$\begin{bmatrix} A_{22} - A_{21}A_{11}^{-1}A_{12} & A_{23} & A_{24} - A_{21}A_{11}^{-1}A_{14} \\ A_{32} & A_{33} & A_{34} \\ A_{42} - A_{41}A_{11}^{-1}A_{12} & A_{43} & A_{44} - A_{41}A_{11}^{-1}A_{14} \end{bmatrix} \begin{bmatrix} X_2 \\ X_3 \\ X_4 \end{bmatrix} = \begin{bmatrix} U_2 - A_{21}A_{11}^{-1}U_1 \\ U_3 \\ U_4 - A_{41}A_{11}^{-1}U_1 \end{bmatrix}$$

จะเห็นว่าในขั้นตอนของการทอนสมการของเมตริกซ์สัมประสิทธิ์ชนิดต่าง ๆ มีการหานิพจน์ $A_{22} - A_{21}A_{11}^{-1}A_{12}$ และ $U_2 - A_{21}A_{11}^{-1}U_1$ เหมือนกัน จึงให้โปรแกรมย่อยชื่อ FDSBP (Forward of Directed Solution by Partitioning and Recursive Partitioning) เป็นโปรแกรมย่อยใช้ในการคำนวณนิพจน์ $A_{22} - A_{21}A_{11}^{-1}A_{12}$ และ $U_2 - A_{21}A_{11}^{-1}U_1$

นอกจากจะหาค่าของ $A_{22} - A_{21}A_{11}^{-1}A_{12}$ และ $U_2 - A_{21}A_{11}^{-1}U_1$ แล้ว การหาคำตอบของสมการที่มีรูปแบบแบนด์บอร์เดอร์ยังมีการหาค่าของ $A_{42} - A_{41}A_{11}^{-1}A_{12}$, $A_{44} - A_{41}A_{11}^{-1}A_{14}$, $A_{24} - A_{21}A_{11}^{-1}A_{14}$ และ $U_4 - A_{41}A_{11}^{-1}U_1$ จึงให้โปรแกรมย่อยชื่อ FDSBB (Forward of Directed Solution of Banded - Bordered Matrix) เป็นโปรแกรมหาค่าของนิพจน์ $A_{42} - A_{41}A_{11}^{-1}A_{12}$, $A_{44} - A_{41}A_{11}^{-1}A_{14}$ และ $U_4 - A_{41}A_{11}^{-1}U_1$

ในขั้นตอนของการแทนค่ากลับมีการหาค่าของ X_1 ดังนี้

เมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์สมมาตร

$$X_1 = A_{11}^{-1}U_1 - A_{11}^{-1}A_{12}X_2$$

เมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์สมมาตรแถบ

$$X_1 = A_{11}^{-1}U_1 - A_{11}^{-1}A_{12}X_2$$

เมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์สมมาตรแบนด์บอร์เดอร์

$$X_1 = A_{11}^{-1}U_1 - A_{11}^{-1}A_{12}X_2 - A_{11}^{-1}A_{14}X_4$$

จะพบว่าไม่ว่าเมตริกซ์สัมประสิทธิ์เป็นชนิดใดจะต้องมีการหานิพจน์ $A_{11}^{-1}U_{11}$ แต่เมื่อย้อนกลับไปดูในขั้นตอนการทอนสมการในส่วนของโปรแกรมย่อย FDSBP ในการหา $U_2 - A_{21}A_{11}^{-1}U_1$ ถ้าเก็บค่าของ $A_{11}^{-1}U_1$ ใช้ในขณะการคำนวณจะทำให้ไม่ต้องหาใหม่ จึงให้โปรแกรมย่อย FDSBP เก็บค่าของ $A_{11}^{-1}U_1$ ไว้ด้วย

ให้โปรแกรมย่อยชื่อ BDSBB (Backward Directed Solution of Banded - Bordered Matrix) เป็นโปรแกรมใช้หาค่าของนิพจน์ $- A_{11}^{-1}A_{12}X_2 - A_{11}^{-1}A_{14}X_4$

ดังนั้นโปรแกรม DSBRP จะมีโปรแกรมย่อย 3 โปรแกรมคือ

1. FDSBP
2. FDSBB
3. BDSBB

โปรแกรมย่อยแต่ละโปรแกรมมีหน้าที่ดังนี้

โปรแกรม FDSBP ใช้คำนวณหา

1. $A_{22} - A_{21}A_{11}^{-1}A_{12}$
2. $U_2 - A_{21}A_{11}^{-1}U_1$
3. $A_{11}^{-1}U_1$

โปรแกรม FDSBB ใช้คำนวณหา

1. $A_{42} - A_{41}A_{11}^{-1}A_{12}$
2. $A_{44} - A_{41}A_{11}^{-1}A_{14}$
3. $U_4 - A_{41}A_{11}^{-1}U_1$

โปรแกรม BDSBB ใช้คำนวณ

$$-A_{11}^{-1}A_{12}X_2 - A_{11}^{-1}A_{14}X_4$$

5.2.3 รายละเอียดของโปรแกรม

5.2.3.1 ขั้นตอนการทำงานของโปรแกรม

ขั้นตอนการทำงานของโปรแกรม DSBRP โดยสังเขปได้แสดงไว้ในรูปที่ 5.1 ส่วนผังการทำงานโดยละเอียดอยู่ในภาคผนวก ก.1 หน้าที่ของโปรแกรมนี้ก็คือ ตรวจสอบว่าเมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์สมมาตรชนิดใดซึ่งมีโอกาสเป็นไปได้สามแบบคือ เมตริกซ์สมมาตรเมตริกซ์สมมาตรแถบและเมตริกซ์สมมาตรแบนด์บอร์เดอร์ แล้วใช้วิธีการสำหรับหาคำตอบของระบบสมการชนิดนั้นหาคำตอบซึ่งมี 3 วิธีคือ

1. วิธีการหาคำตอบโดยตรงโดยการแบ่งส่วน
2. วิธีการหาคำตอบโดยตรงโดยวิธีรีเดอร์ซีฟพาทิชันแบบที่ 1
3. วิธีการหาคำตอบโดยตรงโดยวิธีรีเดอร์ซีฟพาทิชันแบบที่ 2

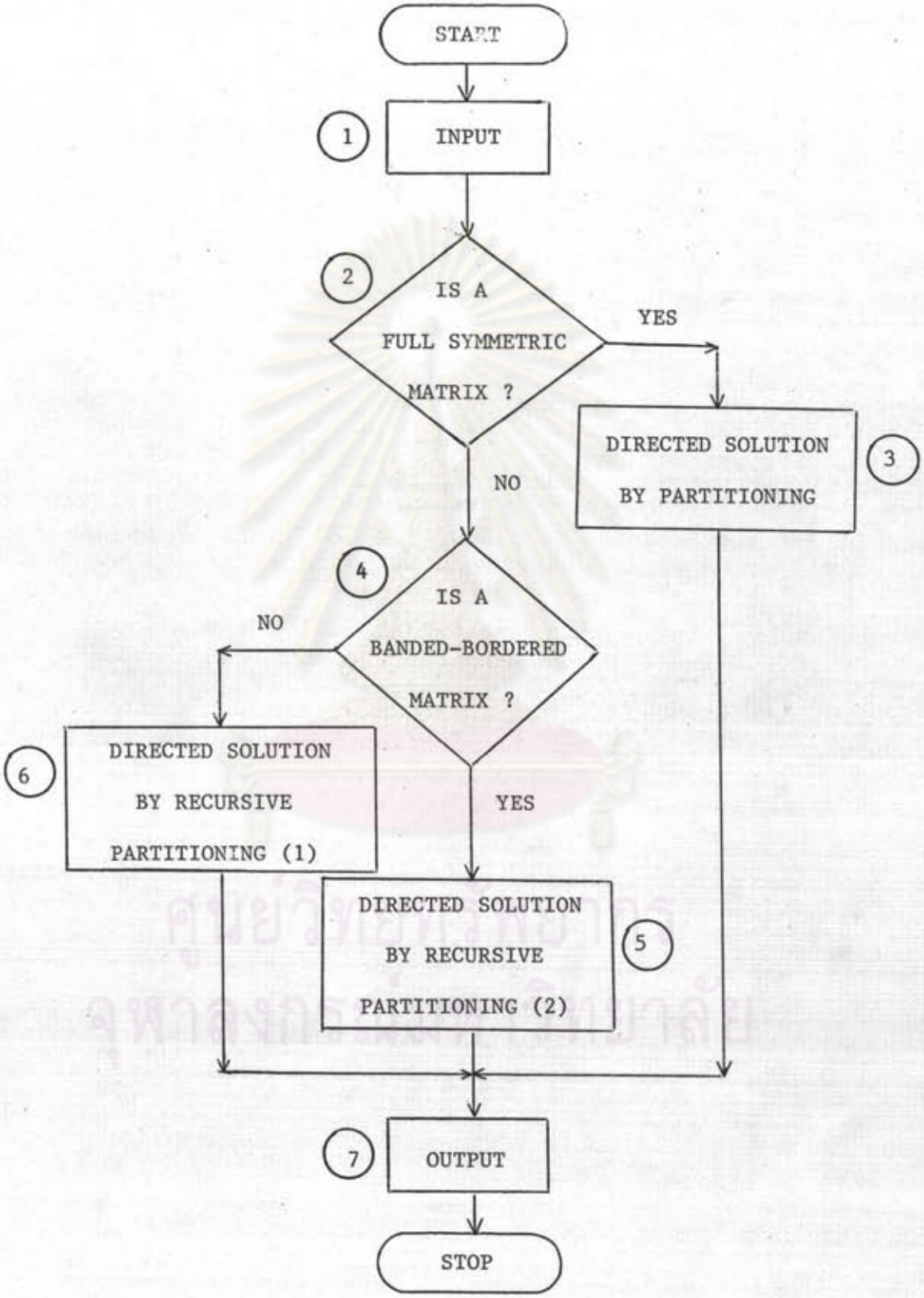
ขั้นตอนต่าง ๆ ของโปรแกรม DSBRP ตามรูปที่ 5.1 มีดังนี้

ขั้นตอนที่ 1 เป็นข้อมูลที่จะนำเข้าสู่คอมพิวเตอร์ซึ่งเป็นค่าตัวเลขของข้อมูลที่จะต้องนำเข้า

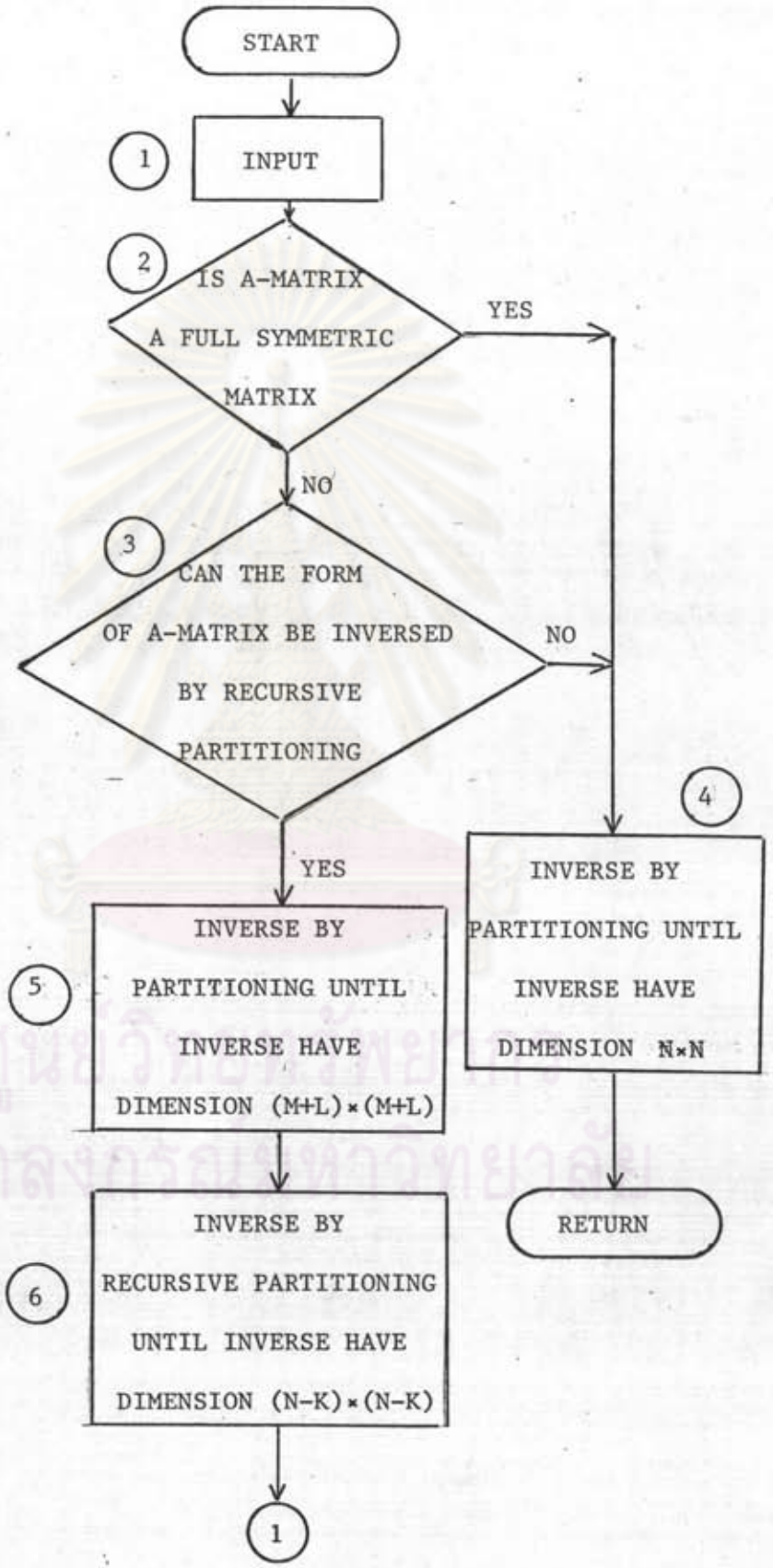
1. ขนาดของระบบสมการ (N)
2. ความกว้างแถบของเมตริกซ์สัมประสิทธิ์ (L)
3. ความกว้างบอร์เดอร์ของเมตริกซ์สัมประสิทธิ์ (K)
4. ขนาดของเมตริกซ์ย่อย A_{11} ซึ่งเป็นเมตริกซ์ที่จะต้องหาส่วนกลับ (M)
5. ค่าธาตุมูลในส่วนของสามเหลี่ยมล่างเฉพาะในแถบและบอร์เดอร์ของเมตริกซ์สัมประสิทธิ์
6. ค่าธาตุมูลของเวคเตอร์ค่าคงที่

ขั้นตอนที่ 2 ตรวจสอบความเป็นเมตริกซ์สมมาตรเต็ม

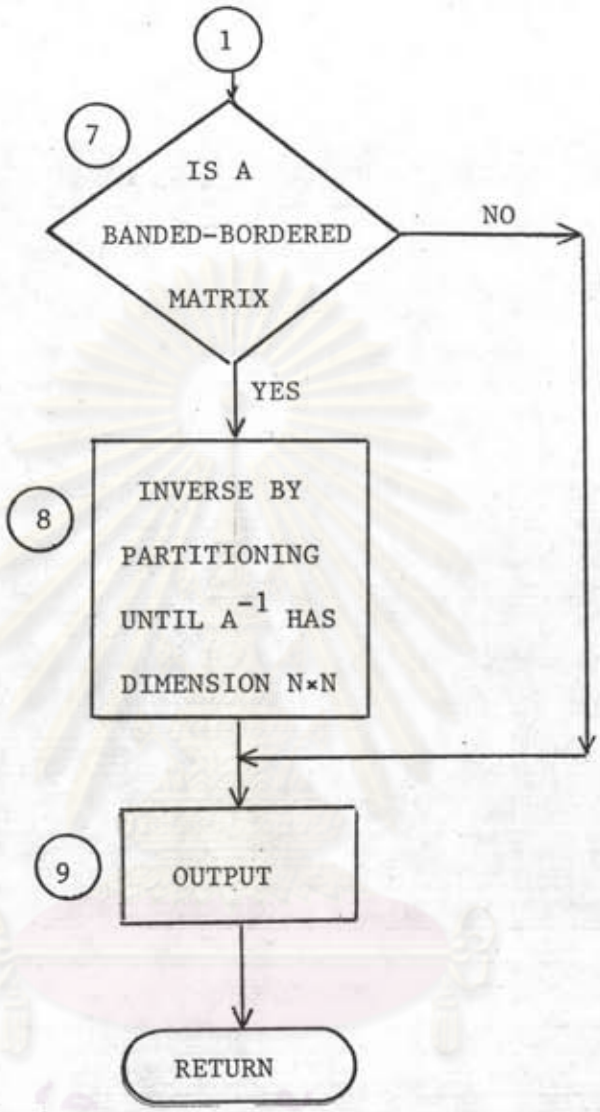
เป็นขั้นตอนที่ใช้ตรวจสอบว่าเมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์สมมาตรเต็มหรือไม่ โดยเปรียบเทียบว่าขนาดของความกว้างแถบเท่ากับขนาดของเมตริกซ์หรือไม่ ถ้าเท่าก็แสดงว่าเป็นเมตริกซ์สมมาตรเต็มแล้วดำเนินตามขั้นตอนที่ 3 คือ ถ้าไม่เท่าก็แสดงว่าเป็นเมตริกซ์สมมาตรแถบหรือสมมาตรแบนด์บอร์เดอร์ก็ดำเนินการตามขั้นตอนที่ 4



รูป 5.1 แผนผังแสดงขั้นตอนการทำงานโดยสังเขปของโปรแกรม DSBRP



รูป 5.2 แผนผังแสดงขั้นตอนการทำงานโดยสังเขปของโปรแกรม KRINV



รูป 5.2 (ต่อ)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ขั้นตอนที่ 3 หาคำตอบโดยตรงโดยการแบ่งส่วน

เป็นขั้นตอนที่ใช้หาคำตอบของสมการที่มีเมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์สมมาตรโดยวิธีการแบ่งส่วนซึ่งมีการดำเนินการดังนี้

1. หาจำนวนรอบที่จะใช้ในการลดขนาดของสมการ จนสามารถหาค่าของตัวไม่ทราบค่า

2. ดำเนินในขั้นตอนของการทอนสมการโดยเรียกโปรแกรมย่อย FDSBP เท่าจำนวนรอบที่หาได้ในข้อ 1

3. หาเมตริกซ์ส่วนกลับของเมตริกซ์สัมประสิทธิ์ที่ถูกลดขนาดแล้ว และนำไปคูณกับเวคเตอร์ค่าคงที่ที่ถูกลดขนาดแล้วได้ค่าของตัวไม่ทราบค่า

4. ดำเนินขั้นตอนของการแทนค่ากลับโดยเรียกโปรแกรมย่อย BDSBB เท่าจำนวนรอบที่หาได้ในข้อที่ 1 ซึ่งจะได้เวคเตอร์ของตัวไม่ทราบค่าขนาดใหญ่ขึ้นเรื่อย ๆ จนได้เวคเตอร์ของตัวไม่ทราบค่าทั้งตัว

ขั้นตอนที่ 4 ตรวจสอบความเป็นเมตริกซ์แบนด์บอร์เดอร์

เป็นขั้นตอนที่ใช้ตรวจสอบว่าเมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์แบนด์บอร์เดอร์หรือไม่ โดยตรวจสอบจากค่าของความกว้างบอร์เดอร์ ถ้าความกว้างบอร์เดอร์เท่ากับศูนย์แสดงว่าเป็นเมตริกซ์สมมาตรแถบก็ไปดำเนินการตามขั้นตอนที่ 6 ถ้าความกว้างบอร์เดอร์ไม่เท่ากับศูนย์แสดงว่าเป็นเมตริกซ์สมมาตรแบนด์บอร์เดอร์ ก็ไปดำเนินการตามขั้นตอนที่ 5

ขั้นตอนที่ 5 การหาคำตอบโดยตรงโดยวิธีรีเคอร์ซีฟพาทิชันแบบที่ 1

เป็นขั้นตอนที่ใช้หาคำตอบของสมการที่มีเมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์สมมาตรแบนด์บอร์เดอร์โดยวิธีรีเคอร์ซีฟพาทิชัน ซึ่งมีการดำเนินการดังนี้

1. หาจำนวนรอบที่จะต้องใช้ในการลดขนาดของสมการจนสามารถหาค่าของตัวไม่ทราบค่าได้โดยตรง

2. ดำเนินขั้นตอนของการทอนสมการโดยเรียกโปรแกรม FDSBP และโปรแกรม FDSBB เท่าจำนวนรอบที่หาได้ในข้อที่ 1

3. หาเมตริกซ์ส่วนกลับของเมตริกซ์สัมประสิทธิ์ซึ่งถูกลดขนาดแล้ว และนำไปคูณกับเวคเตอร์ค่าคงที่ที่ถูกลดขนาดแล้วจะได้ค่าของตัวไม่ทราบค่า

4. ดำเนินขั้นตอนแทนค่ากลับโดยเรียกโปรแกรมย่อย BDSBB มาใช้เท่าจำนวนรอบ

ที่หาได้ในข้อที่ 1 ซึ่งจะได้เวคเตอร์ของตัวไม่ทราบค่าขนาดใหญ่ขึ้นเรื่อย ๆ จนได้เวคเตอร์ของตัวไม่ทราบค่าทั้งตัว

ขั้นตอนที่ 6 การหาคำตอบโดยตรงโดยวิธีรีเคอร์ซีฟพาทิชันแบบที่ 2

เป็นขั้นตอนที่ใช้หาคำตอบของสมการที่มีเมตริกซ์สัมประสิทธิ์เป็นเมตริกซ์สมมาตรแถบโดยวิธีรีเคอร์ซีฟพาทิชันซึ่งมีการดำเนินการดังนี้

1. หาจำนวนรอบที่จะต้องใช้ในการลดขนาดของสมการจนสามารถหาค่าของตัวไม่ทราบค่าได้โดยตรง

2. ดำเนินขั้นตอนของการทอนสมการโดยเรียกโปรแกรม FDSBP มาใช้เท่าจำนวนรอบที่หาได้ในข้อ 1

3. หาเมตริกซ์ส่วนกลับของเมตริกซ์สัมประสิทธิ์ซึ่งถูกลดขนาดแล้ว และนำไปคูณกับเวคเตอร์ค่าคงที่ที่ถูกลดขนาดแล้วเช่นเดียวกัน จะได้ค่าของตัวไม่ทราบค่า

4. ดำเนินขั้นตอนแทนค่ากลับโดยเรียกโปรแกรม BDSBB มาใช้เท่าจำนวนรอบที่หาได้ในข้อที่ 1 ซึ่งในแต่ละรอบจะได้เวคเตอร์ของตัวไม่ทราบค่ามีขนาดเพิ่มขึ้นเรื่อย ๆ จนได้คำตอบของสมการทั้งหมด

ขั้นตอนที่ 7 ผลลัพธ์

ผลลัพธ์ที่ได้คือค่าของตัวไม่ทราบค่าของระบบสมการ

5.2.4 วิธีการใช้โปรแกรม

5.2.4.1 วิธีการใช้โปรแกรมย่อย DSBRP

วัตถุประสงค์

เพื่อหาคำตอบโดยตรงจากระบบสมการที่มีรูปแบบแบนด์หรือแบนด์บอร์เดอร์ โดยวิธีรีเคอร์ซีฟพาทิชัน

วิธีการใช้

CALL DSBRP (A,N,L,K,X,U,H,M)

เมื่อ A คือเวคเตอร์ที่เก็บค่าธาตุมูลของเมตริกซ์สัมประสิทธิ์เฉพาะในแถบหรือบอร์เดอร์เท่านั้น เวคเตอร์ มีขนาด $\frac{1}{2}[N(N+1) - (N-L-K)(N-L-K+1)]$

N คือขนาดของระบบสมการ

L คือความกว้างแถบ

K คือความบอร์เดอร์

X คือเวคเตอร์ของตัวไม่ทราบค่า มีขนาด N

U คือเวคเตอร์ของตัวคงที่ มีขนาด N

H คือเวคเตอร์ช่วยในการคำนวณ มีขนาด $\frac{M(M+1)}{2}$

M คือขนาดของเวคเตอร์ย่อยของ X ซึ่งถูกกำจัดในแต่ละรอบ ขนาดของ M อยู่ในช่วง $1 \leq M \leq L$

ไปรแกรมย่อยที่ต้องใช้

FDSBP , FDSBB , BDSBB , DSINV , DMFDS

5.2.4.2 วิธีการใช้ไปรแกรมย่อย EDSBRP

วัตถุประสงค์

เพื่อหาคำตอบโดยตรงของระบบสมการที่มีรูปแบบแบนด์หรือแบนด์บอร์เดอร์ โดยวิธีรีเดอร์ซีฟพาทิชัน เหมือน DSBRP แต่ EDSBRP ใช้วิธีเก็บค่าธาตุมูลของเวคเตอร์ A ลงในจานแม่เหล็ก

วิธีการใช้

CALL EDSBRP (N,L,K,X,U,H,M)

เมื่อ N คือขนาดของระบบสมการ

L คือความกว้างแถบ

K คือความกว้างบอร์เดอร์

X คือเวคเตอร์ของตัวไม่ทราบค่า มีขนาด N

U คือเวคเตอร์ของตัวคงที่ มีขนาด N

H คือเวคเตอร์ช่วยในการคำนวณ มีขนาด $\frac{M(M+1)}{2}$

M คือขนาดของเวคเตอร์ย่อยของ X ซึ่งถูกกำจัดในแต่ละรอบ ขนาดของ M อยู่ในช่วง $1 \leq M \leq L$

ไปรแกรมย่อยที่ต้องใช้

EFDSBP , EFDSBB , EBDSBB , DSINV , DMFSD

หมายเหตุ

ค่าธาตุมูลของเวคเตอร์ A จะต้องนำไปเก็บในแฟ้มข้อมูล (Data File) ซึ่งมีลักษณะของการจัดข้อมูลแบบเรียงตามลำดับ (Sequential Organization) การเข้าหาข้อมูลเป็นแบบการเข้าถึงโดยตรง (Direct Access) และรูปแบบการเก็บข้อมูลเป็นแบบ Unformatted โดยที่แต่ละระเบียน (Record) จะเก็บค่าธาตุมูล 1 ค่า หลังจากการคำนวณแล้วค่าธาตุมูลของเวคเตอร์ A จะเปลี่ยนไป

คำสั่งที่ใช้ในการเปิดแฟ้มข้อมูลให้มีคุณสมบัติดังกล่าว คือ

OPEN (UNIT = N , FILE = 'FILENAME' , STATUS = 'UNKNOWN' , ORGANIZATION = 'SEQUENTIAL' , ACCESS = 'DIRECT' , FORM = 'UNFORMATTED')

โดยที่ n คือ Logical Unit Number ผู้ใช้กำหนดได้เอง

FILENAME คือ ชื่อของแฟ้มข้อมูลที่เชื่อมต่อกับโปรแกรม Unit n ผู้ใช้กำหนดได้เอง

5.3 โปรแกรมหาเมตริกซ์ส่วนกลับของเมตริกซ์สมมาตรแถบและสมมาตรแบนด์บอร์เดอร์

การหาเมตริกซ์ส่วนกลับของเมตริกซ์สมมารถแถบและสมมาตรแบนด์บอร์เดอร์โดยวิธีรีเดอร์ซีฟพาทิชัน จะกระทำโดยจัดสร้างโปรแกรมคอมพิวเตอร์ 3 โปรแกรม คือโปรแกรม KRINV โปรแกรม KRINV1 และโปรแกรม EKRINV เพื่อที่ให้ผู้ใช้สามารถเลือกใช้โปรแกรมให้เหมาะกับงาน เมื่อต้องการหาเมตริกซ์ส่วนกลับที่มีขนาดเล็ก ก็เลือกใช้โปรแกรมที่ใช้หน่วยความจำภายในซึ่งการคำนวณจะรวดเร็ว การใช้หน่วยความจำภายในมีโปรแกรมให้เลือก 2 โปรแกรมคือ KRINV และ KRINV1 โดยโปรแกรม KRINV1 จะใช้จำนวนหน่วยความจำมากกว่าแต่ใช้เวลาน้อยกว่า เมื่อต้องการหาเมตริกซ์ส่วนกลับที่มีขนาดใหญ่เกินกว่าที่หน่วยความจำภายในจะเก็บได้ ก็ใช้โปรแกรม EKRINV ซึ่งนำค่าของธาตุมูลของเวคเตอร์ไปเก็บในจานบันทึกแม่เหล็ก โปรแกรมนี้มีข้อดีคือ สามารถหาเมตริกซ์ส่วนกลับได้ขนาดใหญ่กว่า แต่ข้อเสียคือใช้เวลามากกว่าโดยที่โปรแกรมทั้งสามมีโครงสร้างเหมือนกัน จึงกล่าวเฉพาะรายละเอียดและโครงสร้างของโปรแกรม KRINV แต่วิธีการใช้โปรแกรมจะแสดงแยกจากกัน

5.3.1 วัตถุประสงค์

หาเมตริกซ์ส่วนกลับของเมตริกซ์สมมาตรแถบและเมตริกซ์สมมาตรแบนด์บอร์เดอร์โดยใช้คอมพิวเตอร์ขนาดเล็กหาเมตริกซ์ส่วนกลับให้ได้ขนาดใหญ่ที่สุดโดยใช้เวลาคำนวณพอสมควร.

เนื่องจากสามารถถือได้ว่า เมตริกซ์สมมาตรทั่วไป เป็น เมตริกซ์สมมาตรแถบที่มีความกว้าง แถบ เท่ากับขนาดของ เมตริกซ์ ดังนั้นหากต้องการใช้โปรแกรมนี้หาส่วนกลับของ เมตริกซ์ดังกล่าว ก็ย่อมกระทำได้ แต่จากการพยายามให้ใช้หน่วยความจำกลางน้อยที่สุดที่จะ เป็นไปได้อาจต้องใช้ เวลามากกว่าโปรแกรมที่มีใช้อยู่

5.3.2 โครงสร้างของโปรแกรม

โปรแกรม KRINV จะประกอบด้วยโปรแกรมย่อย 3 โปรแกรมด้วยกันคือ RIBPN , KIBRP และ KISBP

เมื่อพิจารณาถึงวิธีการหา เมตริกซ์ส่วนกลับโดยการใช้เมตริกซ์ย่อยในบทที่ 4 จะพบว่าการหา เมตริกซ์ส่วนกลับสามารถหาได้สองวิธี คือ โดยวิธีการแบ่งส่วนและโดยวิธีรีเคอร์ซีฟ-พาทิชัน

การหา เมตริกซ์ส่วนกลับโดยวิธีการแบ่งส่วนมีการคำนวณหา เมตริกซ์ย่อยของ เมตริกซ์ B ดังนี้

1. $B_{22} = (A_{22} - A_{21}A_{11}^{-1}A_{12})^{-1}$
2. $B_{21} = - B_{22}A_{21}A_{11}^{-1}$
3. $B_{11} = A_{11}^{-1} - A_{11}^{-1}A_{12}B_{21}$

โปรแกรมย่อย RIBPN เป็นโปรแกรมที่ใช้คำนวนหาค่าของ $B_{22} = (A_{22} - A_{21}A_{11}^{-1}A_{12})^{-1}$, $B_{21} = - B_{22}A_{21}A_{11}^{-1}$ และ $B_{11} = A_{11}^{-1} - A_{11}^{-1}A_{12}B_{21}$

การหาส่วนกลับโดยวิธีรีเคอร์ซีฟพาทิชันมีการคำนวณหา เมตริกซ์ย่อยของ เมตริกซ์ B ดังนี้

1. $B_{33} = (A_{22} - A_{32}C_{22}A_{23})^{-1}$
2. $B_{31} = - B_{33}A_{32}C_{21}$
3. $B_{32} = - B_{33}A_{32}C_{22}$
4. $B_{11} = C_{11} - C_{12}A_{23}B_{31}$
5. $B_{21} = C_{21} - C_{22}A_{23}B_{31}$
6. $B_{22} = C_{22} - C_{22}A_{23}B_{32}$

โปรแกรมย่อย เป็นโปรแกรมใช้หา เมตริกซ์ย่อยของ เมตริกซ์ โดยวิธีรีเคอร์-ซีฟพาทิชันมีการหาค่าของ $B_{33} = (A_{33} - A_{32}C_{22}A_{23})^{-1}$, $B_{31} = - B_{33}A_{32}C_{21}$, $B_{32} = - B_{33}A_{32}C_{22}$, $B_{11} = C_{11} - C_{12}A_{23}B_{31}$, $B_{21} = C_{21} - C_{22}A_{23}B_{31}$ และ $B_{22} = C_{22} - C_{22}A_{33}B_{32}$ ส่วนโปรแกรมย่อย KISBP เป็นโปรแกรมที่ใช้หา เมตริกซ์ส่วนกลับ

ของเมตริกซ์สมมาตรโดยวิธีการแบ่งส่วนเพื่อขยายขนาดของ A_{11}^{-1} จากขนาด s × s ไปเป็นขนาด t × t เมื่อ t s เมื่อ t เป็นขนาดของเมตริกซ์ส่วนกลับใหม่ s เป็นขนาดของเมตริกซ์ส่วนกลับเดิมโดยโปรแกรมนี้จะเรียกโปรแกรม RIBRP มาใช้

5.3.3 รายละเอียดของโปรแกรม

5.3.3.1 ขั้นตอนการทำงานของโปรแกรม KRINV

ขั้นตอนการทำงานของโปรแกรม KRINV โดยสังเขปได้แสดงไว้ในรูปที่ 5.2 ซึ่งหน้าที่หลักของโปรแกรมนี้ก็คือ ตรวจสอบว่าเมตริกซ์ที่จะหาเมตริกซ์ส่วนกลับเป็นเมตริกซ์ชนิดใด แล้วหาส่วนกลับตามวิธีของเมตริกซ์ชนิดนั้นโดยมีขั้นตอนดังนี้

ขั้นตอนที่ 1

เป็นการนำข้อมูลเข้าสู่คอมพิวเตอร์ซึ่งเป็นค่าตัวเลขของ

1. ขนาดของเมตริกซ์ (N)
2. ความกว้างของแถบ (L)
3. ความกว้างของบอร์เดอร์ (K)
4. ขนาดของเมตริกซ์ย่อยที่จะต้องหาเมตริกซ์ส่วนกลับ (M)

ขั้นตอนที่ 2

เป็นขั้นตอนที่จะตรวจสอบว่าเมตริกซ์ที่จะหาเมตริกซ์ส่วนกลับเป็นเมตริกซ์สมมาตรเต็มหรือไม่ โดยเปรียบเทียบว่าขนาดของความกว้างแถบเท่ากับขนาดของเมตริกซ์หรือไม่ ถ้าเท่ากันก็แสดงว่าเป็นเมตริกซ์สมมาตรเต็มแล้วดำเนินการตามขั้นตอนที่ 4 ถ้าขนาดของความกว้างแถบไม่เท่ากับขนาดของเมตริกซ์ ก็ให้ดำเนินการตามขั้นตอนที่ 3

ขั้นตอนที่ 3

เป็นขั้นตอนที่จะตรวจสอบว่าเมตริกซ์สมมาตรแถบหรือเมตริกซ์สมมาตรแบนด์บอร์เดอร์นั้น มีรูปแบบที่จะสามารถหาส่วนกลับโดยวิธีรีเคอร์ซีฟพาทิชันหรือไม่ เพราะการหาเมตริกซ์ส่วนกลับโดยวิธีรีเคอร์ซีฟพาทิชันนั้นจะต้องหาส่วนกลับโดยวิธีการแบ่งส่วนก่อน เพื่อให้ได้เมตริกซ์ส่วนกลับมีขนาดใหญ่กว่าความกว้างของแถบเสียก่อน จึงจะสามารถแบ่งเมตริกซ์ออกเป็นสามระดับดังรูป 4.5 แล้วใช้วิธีรีเคอร์ซีฟพาทิชันหาเมตริกซ์ส่วนกลับต่อไป ดังนั้นถ้าเมตริกซ์มีความกว้างแถบมาก การหาเมตริกซ์ส่วนกลับด้วยวิธีการแบ่งส่วนในครั้งแรกจะทำให้ส่วนที่เหลือไม่สามารถ

แบ่งออกได้เป็นสามระดับ เพื่อใช้วิธีรีเดอร์ซีฟพาทิชันได้ เมตริกซ์ที่มีลักษณะเช่นนี้จะต้องใช้วิธีการแบ่งส่วนหาเมตริกซ์ส่วนกลับตามขั้นตอนที่ 4

ขั้นตอนที่ 4

เป็นการหาเมตริกซ์ส่วนกลับโดยวิธีการแบ่งส่วนจนได้เมตริกซ์ส่วนกลับที่ต้องการ ซึ่งมีการดำเนินการดังนี้

1. หาจำนวนรอบที่ต้องใช้ในการหาเมตริกซ์ส่วนกลับ จนได้เมตริกซ์ส่วนกลับที่ต้องการ

2. ดำเนินการหาเมตริกซ์ส่วนกลับโดยเรียกโปรแกรม RIBPN เท่าจำนวนรอบที่หาได้ในข้อ 1 ก็จะได้เมตริกซ์ส่วนกลับในขนาดที่ต้องการ

ขั้นตอนที่ 5

เมื่อเมตริกซ์สมมาตรแถบหรือสมมาตรแบนด์บอร์เดอร์นี้มีรูปแบบที่สามารถหาส่วนกลับโดยวิธีรีเดอร์ซีฟพาทิชันได้ ขั้นตอนนี้ก็เป็นการหาเมตริกซ์ส่วนกลับโดยการแบ่งส่วน เพื่อให้ได้เมตริกซ์ส่วนกลับมีขนาดใหญ่กว่าความกว้างแถบแล้วใช้วิธีรีเดอร์ซีฟพาทิชันหาเมตริกซ์ส่วนกลับในขั้นตอนต่อไป การดำเนินงานในขั้นตอนนี้จะเรียกโปรแกรม RIBPN มาใช้ 1 ครั้ง

ขั้นตอนที่ 6

เป็นการหาเมตริกซ์ส่วนกลับโดยวิธีรีเดอร์ซีฟพาทิชัน จนได้เมตริกซ์ส่วนกลับในขนาดที่ต้องการ ซึ่งมีการดำเนินการดังนี้

1. หาจำนวนรอบที่ใช้วิธีรีเดอร์ซีฟพาทิชัน จนได้เมตริกซ์ส่วนกลับขนาดที่ต้องการ

2. ดำเนินการหาเมตริกซ์ส่วนกลับโดยเรียกโปรแกรม KIBRP มาใช้เท่าจำนวนรอบที่หาได้ในข้อ 1

ขั้นตอนที่ 7

เป็นการตรวจสอบว่าเมตริกซ์เป็นเมตริกซ์สมมาตรแถบหรือสมมาตรแบนด์บอร์เดอร์ โดยตรวจสอบว่าความกว้างบอร์เดอร์มากกว่าศูนย์หรือไม่ ถ้าไม่มากกว่าก็แสดงว่าเป็นเมตริกซ์สมมาตรแถบแล้วดำเนินตามขั้นตอนที่ 9 แต่ถ้ามากกว่าศูนย์ก็แสดงว่าเป็นเมตริกซ์สมมาตรแบนด์บอร์เดอร์ก็ต้องดำเนินการในขั้นตอนที่ 8

ขั้นตอนที่ 8

เป็นการหาเมตริกซ์ส่วนกลับโดยวิธีการแบ่งส่วน จนได้เมตริกซ์ส่วนกลับที่ต้องการซึ่งมีการดำเนินการดังนี้

1. หาจำนวนรอบที่ต้องใช้ในการหาเมตริกซ์ส่วนกลับ จนได้เมตริกซ์ส่วนกลับที่ต้องการ

2. ดำเนินการหาเมตริกซ์ส่วนกลับโดยการแบ่งส่วนโดยเรียกโปรแกรม RIBPN เท่าจำนวนรอบที่หาได้ในข้อ 1 ก็จะได้เมตริกซ์ส่วนกลับที่ต้องการแล้วดำเนินตามขั้นตอนที่ 9

ขั้นตอนที่ 9

แสดงผลลัพธ์คือเมตริกซ์ส่วนกลับของเมตริกซ์ที่ต้องการ

5.3.4 วิธีการใช้โปรแกรม

5.3.4.1 วิธีการใช้โปรแกรมย่อย KRINV

วัตถุประสงค์

เพื่อหาเมตริกซ์ส่วนกลับของเมตริกซ์สมมาตร เมตริกซ์สมมาตรแถบ และเมตริกซ์สมมาตรแบนด์บอร์เดอร์ เมื่อเปรียบเทียบกับ KRINV1 แล้ว KRINV ใช้เนื้อที่หน่วยความจำน้อยกว่า แต่ใช้เวลาในการทำงานมากกว่า

วิธีการใช้

CALL KRINV (A,N,L,K,H,M)

เมื่อ A เวคเตอร์ที่ใช้เก็บค่าธาตุมูลในส่วนสามเหลี่ยมล่างของเมตริกซ์มีขนาด $\frac{N(N+1)}{2}$ หลังการคำนวณ ค่าธาตุมูลเก็บในเวคเตอร์ A จะหายไปและถูกแทนที่ด้วยค่าธาตุมูลในส่วนสามเหลี่ยมล่างของเมตริกซ์ส่วนกลับที่ต้องการ

N ขนาดของเมตริกซ์ที่ต้องการหาส่วนกลับ

L ความกว้างแถบ ถ้าเป็นเมตริกซ์สมมาตร L = N, ถ้าเป็นเมตริกซ์สมมาตรแถบและเมตริกซ์สมมาตรแบนด์บอร์เดอร์ L = ความกว้างของแถบ

K ความกว้างบอร์เดอร์ ถ้าเป็นเมตริกซ์สมมาตรและเมตริกซ์สมมาตรแถบ K = 0 ถ้าเป็นเมตริกซ์สมมาตรแบนด์บอร์เดอร์ K = ความกว้างของบอร์เดอร์

H เวคเตอร์ช่วยในการคำนวณ

กรณีเมตริกซ์สมมาตร H มีขนาด $M(N-1) + \frac{M(M+1)}{2}$

กรณีเมตริกซ์สมมาตรแถบ H มีขนาด $ML + \frac{M(M+1)}{2}$

ถ้า A เป็นเมตริกซ์สมมาตรแบนด์บอร์เดอร์มี 2 กรณี คือ

ถ้า $K \geqslant M$ H มีขนาด $M(N-1) + \frac{M(M+1)}{2}$

ถ้า $K < M$ H มีขนาด $K(N-1) + \frac{M(M+1)}{2}$

M ขนาดของเมตริกซ์ย่อยที่จะต้องหาส่วนกลับในแต่ละรอบ ผู้ใช้กำหนดขนาดของ M ได้ในช่วง $1 \leqslant M \leqslant L$

ไปรแกรมย่อยที่ต้องใช้

KIBRP , KISBP , RIBPN , SINV , MFSD

หมายเหตุ

เวคเตอร์ A จะต้องมีค่าธาตุมูลทั้งหมดในส่วนสามเหลี่ยมล่างของเมตริกซ์ ในกรณีที่เป็นเมตริกซ์สมมาตรแถบหรือเมตริกซ์สมมาตรแบนด์บอร์เดอร์ และมีการเก็บค่าธาตุมูลเฉพาะในแถบหรือในบอร์เดอร์เท่านั้น ก่อนที่จะใช้ไปรแกรมย่อยนี้ได้ จะต้องเรียกใช้ไปรแกรมย่อย PFETP โดย CALL PFETP (A,N,L,K) ก่อน เพื่อจัดตำแหน่งธาตุมูลที่เก็บในเวคเตอร์ A ให้อยู่ในลักษณะเก็บทุกค่าในส่วนของสามเหลี่ยมล่าง

5.3.4.2 วิธีการใช้ไปรแกรมย่อย KRINV1

วัตถุประสงค์

เพื่อหาเมตริกซ์ส่วนกลับของเมตริกซ์สมมาตร เมตริกซ์สมมาตรแถบ และเมตริกซ์สมมาตรแบนด์บอร์เดอร์ เมื่อเปรียบเทียบกับ KRINV แล้ว KRINV1 ใช้เนื้อที่หน่วยความจำมากกว่า แต่การทำงานจะใช้เวลาน้อยกว่า

วิธีการใช้

CALL KRINV1 (A,N,L,K,H,M,NDX)

เมื่อ A คือเวคเตอร์ที่ใช้เก็บค่าธาตุมูลในส่วนสามเหลี่ยมล่างของเมตริกซ์ มีขนาด $\frac{N(N+1)}{2}$ หลังการคำนวณ ค่าธาตุมูลเดิมในเวคเตอร์ A จะหายไป และ

ถูกแทนที่ด้วยค่าธาตุมูลในส่วนสามเหลี่ยมล่างของเมตริกซ์ส่วนกลับที่ต้องการ

N ขนาดของเมตริกซ์ที่ต้องการหาส่วนกลับ

L ความกว้างแถบ ถ้าเป็นเมตริกซ์สมมาตร L = N, ถ้าเป็นเมตริกซ์สมมาตรแถบ และหาเมตริกซ์สมมาตรแบนด์บอร์เดอร์ L = ความกว้างของแถบ

K ความกว้างบอร์เดอร์ ถ้าเป็นเมตริกซ์สมมาตรและเมตริกซ์สมมาตรแถบ K = 0 ถ้าเป็นเมตริกซ์สมมาตรแบนด์บอร์เดอร์ K = ความกว้างของบอร์เดอร์

H คือเวคเตอร์ช่วยในการคำนวณ

เมตริกซ์สมมาตร H มีขนาด $M(N-1) + \frac{M(M+1)}{2}$

เมตริกซ์สมมาตรแถบ H มีขนาด $ML + \frac{M(M+1)}{2}$

เมตริกซ์สมมาตรแบนด์บอร์เดอร์ มี 2 กรณี

$K \geqslant M$ H มีขนาด $M(N-1) + \frac{M(M+1)}{2}$

$K < M$ H มีขนาด $K(N-1) + \frac{M(M+1)}{2}$

M ขนาดของเมตริกซ์ย่อยที่จะต้องหาส่วนกลับในแต่ละรอบ ผู้ใช้กำหนดขนาดของ M ได้ในช่วง $1 \leq M \leqslant L$

NDX เวคเตอร์ช่วยในการคำนวณ มีขนาด N

ไปรแกรมย่อยที่ต้องใช้

KIBRP1 , KISBP1 , RIBPN1 , SINV , MFSD

หมายเหตุ

เวคเตอร์ A จะต้องมีค่าธาตุมูลทั้งหมดในส่วนสามเหลี่ยมล่างของเมตริกซ์ ในกรณีที่เป็นเมตริกซ์สมมาตรแถบหรือเมตริกซ์สมมาตรแบนด์บอร์เดอร์ และมีการเก็บค่าธาตุมูลเฉพาะในแถบหรือ ในบอร์เดอร์เท่านั้น ก่อนที่จะใช้ไปรแกรมย่อยนี้ได้จะต้องเรียกใช้ไปรแกรมย่อย PFETP โดย CALL PFETP (A,N,L,K) ก่อน เพื่อจัดตำแหน่งธาตุมูลที่เก็บในเวคเตอร์ A ให้อยู่ในลักษณะเก็บทุกค่าในส่วนของสามเหลี่ยมล่าง