

การเร่งความเร็วของการสร้างภาพตัดขวางที่ใช้อัลกอริธึมเชิงพีชคณิต
สำหรับวิเคราะห์ทางเดินแสงแบบกรวยด้วยระบบประมวลผลแบบคลัสเตอร์

RAPID CONE-BEAM SIMULTANEOUS ALGEBRAIC
RECONSTRUCTION TECHNIQUE BY CLUSTERING SYSTEM



โกสินทร์ กาฬรัตน์
KOSIN KALARAT

รพ.
119667
2649

เลขหมู่.....
เลขทะเบียน 63496
วัน,เดือน,ปี 29 ส.ค. 2549

b. 11638138
1.....

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2549
ISBN 974 - 15 - 2200 - 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**RAPID CONE-BEAM SIMULTANEOUS ALGEBRAIC
RECONSTRUCTION TECHNIQUE BY CLUSTERING SYSTEM**



**A THESIS SUBMITTED IN PARTIAL FULFILMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER ENGINEERING IN ELECTRONICS ENGINEERING
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2006

ISBN 974 - 15 - 2200 - 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2006

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์ การเร่งความเร็วของการสร้างภาพตัดขวางที่ใช้อัลกอริทึมเชิงพีชคณิต
สำหรับวิเคราะห์ทางเดินแสงแบบกรวยด้วยระบบประมวลผลแบบ-
คลัสเตอร์

นักศึกษา นายโกสินทร์ กาพรัตน์

รหัสนักศึกษา 47060402

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา วิศวกรรมอิเล็กทรอนิกส์

พ.ศ. 2549

อาจารย์ผู้ควบคุมวิทยานิพนธ์ รศ.ดร. มนัส สังวรศิลป์

อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม ศศ.ดร. ชูชาติ ปิณฑวิรุจน์

บทคัดย่อ

บทความนี้นำเสนอการเร่งความเร็วในการสร้างภาพตัดขวางที่ใช้อัลกอริทึมเชิงพีชคณิต
สำหรับวิเคราะห์ทางเดินแสงแบบกรวยด้วยระบบประมวลผลแบบคลัสเตอร์ ซึ่งในการ
ขบวนการสร้างภาพตัดขวางสำหรับวิเคราะห์ทางเดินแสงแบบกรวยโดยใช้อัลกอริทึมเชิงพีชคณิต
นั้น จะต้องใช้ระยะเวลาในการประมวลผลเพื่อที่จะให้เป็นภาพตัดขวางเป็นเวลาค่อนข้างนาน
ดังนั้นในบทความวิจัยนี้จึงได้นำเอาหลักการของระบบประมวลผลแบบขนานบนคอมพิวเตอร์
หลายเครื่องที่เรียกว่าระบบคลัสเตอร์และขบวนการสร้างภาพตัดขวางสำหรับวิเคราะห์ทางเดินแสง
แบบกรวยโดยใช้อัลกอริทึมเชิงพีชคณิต เพื่อให้การสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิต
ทำได้รวดเร็ว สำหรับการแบ่งงานในระบบประมวลผลแบบขนาน มีแบ่งงานโดยใช้วิธีราวโรบินด์
(Round robin algorithm) เพื่อกระจายงานออกไปยังโหนดต่าง ๆ ในเพื่อทำการการสร้าง
ภาพตัดขวาง ซึ่งจะช่วยแก้ปัญหาเนื่องจากการประมวลผลในแต่ละรอบนั้นข้อมูลของแต่ละรอบจะ
ขึ้นต่อกัน อีกทั้งวิธีนี้ทำให้การทำงานรวมของระบบทั้งหมดนั้นทำงานไปพร้อมกัน ทำให้เหมาะ
สำหรับการประมวลผลที่ต้องการกระบวนการทำซ้ำ

Thesis Title	RAPID CONE-BEAM SIMULTANEOUS ALGEBRAIC RECONSTRUCTION TECHNIQUE BY CLUSTERING SYSTEM
Student	Mr.KOSIN KALARAT
Student ID.	47060402
Degree	Master of Engineering
Programme	Electronics Engineering
Year	2006
Thesis Advisor	Assoc. Prof. Dr. Manas Sangworasil
Thesis Co-advisor	Asst. Prof. Dr. Chuchart Pinthaviruj

ABSTRACT

An important problem in image processing is to construct a cross section of an object from several images of its trans-axial projection. However, the time consuming and the complexity of the reconstruction process are the crucial problems. In addition, the reconstruction process requires very high performance of the computer. Therefore, in this paper, a concept of parallel programming method is employed to speed up a Cone-Beam Simultaneous Algebraic Reconstruction Technique. The scheduling process in the clustering system is improved. The appropriate amount of work is distributed to each computer (node) in the clustering system using a round-robin algorithm scheduling scheme. Our proposed system works successfully with decreasing the reconstruction time up to 78 percent referred to the normal image reconstruction form projection performed on a single computer.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้ถูกลงได้อย่างดี ด้วยความช่วยเหลือและกาสนับสนุนจากบุคคลหลายๆ ท่าน ซึ่งผู้เขียนขอขอบพระคุณทุกๆ ท่านต่อไปนี้

ขอขอบพระคุณ คุณพ่อ คุณแม่ ที่ให้ความสนับสนุนด้านทุนการศึกษาตลอดจนคำแนะนำ และกำลังใจในการทำวิทยานิพนธ์ฉบับนี้ด้วยดีเสมอมา

ขอขอบพระคุณ รศ.ดร.มนัสสัจวารศิลป์ อาจารย์ที่ปรึกษาผู้ซึ่งคอยให้คำปรึกษา และให้ความสนับสนุนทางด้านความรู้และแนวทางในการทำวิทยานิพนธ์ ตลอดระยะเวลาการศึกษารวมถึงเอื้อเฟื้อสนับสนุนเครื่องมือที่ใช้ในการทำงานวิจัย ซึ่งผู้เขียนรู้สึกซาบซึ้งในความเมตตาของท่านเป็นอย่างสูง

ขอขอบพระคุณ ผศ.ดร.ชูชาติ ปิณฑวิรุจน์ อาจารย์ที่ปรึกษาร่วม ผู้ซึ่งให้คำปรึกษา และคำแนะนำต่างๆเกี่ยวกับงานวิจัย

ขอขอบพระคุณ สำนักวิจัยและบริการคอมพิวเตอร์ ที่ให้บริการระบบประมวลผลขนานแบบคลัสเตอร์ และ คุณนพรัตน์ พันธุ์เสนา ที่ได้ช่วยอธิบายในเรื่องพื้นฐานเกี่ยวกับการประมวลผลขนาน ทำให้วิทยานิพนธ์ฉบับนี้ไม่ต้องเริ่มต้นจากศูนย์

และสุดท้าย ขอขอบพระคุณ เพื่อนๆ พี่ๆ น้องๆ ในห้องวิจัยที่ได้แบ่งปันความรู้หลายๆอย่างตลอดจนช่วยเหลือในการจัดทำวิทยานิพนธ์ฉบับนี้

คุณค่าและประโยชน์อันพึงมีจากวิทยานิพนธ์ฉบับนี้ ผู้วิจัยขอบอแด่ผู้มีพระคุณทุกท่าน

โกสินทร์ กาพรัตน์

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VIII
สารบัญรูป	IX
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของงานวิจัย	2
1.3 สมมติฐานของการศึกษา	2
1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในงานวิจัย	2
1.5 ขอบเขตของงานวิจัย	3
บทที่ 2 ภาพตัดขวางของวัตถุและคณิตศาสตร์ที่เกี่ยวข้อง.....	5
2.1 บทนำ.....	5
2.2 วิธีการสร้างภาพตัดขวาง	5
2.3 ภาพถ่ายเอ็กซ์เรย์.....	6
2.3.1 ประวัติการค้นพบเอ็กซ์เรย์	6
2.3.2 การเกิดรังสีเอ็กซ์	7
2.3.3 ประโยชน์ของรังสีเอ็กซ์เรย์ทางการแพทย์.....	9
2.4 คณิตศาสตร์ที่เกี่ยวข้องกับคอมพิวเตอร์ถ่ายภาพตัดขวาง	10
2.4.1 การหาค่าสัมประสิทธิ์การคูณกลับ.....	10
2.4.2 การอินทิเกรตในแนวเส้น (Line Integrals) และการ โปรเจกชัน (Projections)	14
2.4.3 การแปลงเรดอน (Radon Transform)	17
2.5 อัลกอริทึมของการสร้างภาพตัดขวาง.....	18
2.6 การสร้างภาพตัดขวางที่มีทางเดินแสงแบบขนาน.....	18
2.6.1 วิธีการสร้างภาพตัดขวางโดยใช้การแปลงฟูเรียร์.....	19
2.6.2 วิธีการสร้างภาพตัดขวางโดยใช้วิธีแบคโปรเจกชัน.....	24

สารบัญ (ต่อ)

	หน้า
2.6.3 วิธีการสร้างภาพตัดขวางโดยใช้วิธีฟิลเตอร์แบบ โปรเจกชัน.....	27
2.7 ทฤษฎีพื้นฐานของอัลกอริทึมแบบพีชคณิต.....	30
2.7.1 ชนิดของอัลกอริทึมแบบพีชคณิต.....	33
2.7.2 ทฤษฎีของคาร์แมค(Kacmarz Method).....	35
2.7.3 การคำนวณหาค่าแฟกเตอร์น้ำหนัก.....	39
2.8 ทำไมอัลกอริทึมแบบพีชคณิตจึงต้องการจำนวน โปรเจกชัน น้อยกว่าวิธีฟิลเตอร์แบบ โปรเจกชัน.....	40
2.9 โปรเจกชันและแบ็คโปรเจกชันใน SART.....	41
2.9.1 SART โปรเจกชัน.....	42
2.9.2 SART แบ็คโปรเจกชัน.....	42
บทที่ 3 อัลกอริทึมของการสร้างภาพตัดขวาง.....	43
3.1 บทนำ.....	43
3.2 การสร้างภาพตัดขวางแบบพัด.....	43
3.2.1 มุมระหว่างเส้นแสงเท่ากัน.....	43
3.2.2 ระยะระหว่างเส้นแสงเท่ากัน.....	47
3.3 การสร้างภาพตัดขวางแบบกรวย.....	50
3.4 โปรเจกชันของวัตถุ 3 มิติ.....	51
3.5 ฟิลเตอร์แบ็คโปรเจกชันแบบ 3 มิติ.....	54
3.6 ART และ SART แบบ 3 มิติ.....	56
บทที่ 4 การประมวลผลแบบขนานและ โครงสร้างของระบบคลัสเตอร์.....	61
4.1 บทนำ.....	61
4.2 การประมวลผลแบบขนาน.....	62
4.2.1 การประยุกต์ใช้การประมวลผลแบบขนาน.....	62
4.3 การจำแนกประเภทของคอมพิวเตอร์.....	64
4.3.1 การจำแนกตามงานที่ประยุกต์ใช้.....	64
4.3.2 การจำแนกตามลักษณะของเครื่องในระบบ.....	65

สารบัญ (ต่อ)

	หน้า
4.4 องค์ประกอบและการสร้างระบบคลัสเตอร์.....	66
4.4.1 ระบบเครือข่ายความเร็วสูง.....	66
4.4.2 โปรแกรมบริการการสื่อสาร.....	68
4.4.3 ระบบปฏิบัติการ.....	68
4.4.4 คลัสเตอร์มีคเคิลเวร์.....	68
4.4.5 โปรแกรมสำหรับสร้างสถานะแวดล้อมของโปรแกรมแบบขนาน.....	69
4.5 การสื่อสารระหว่างงานด้วยวิธีส่งข้อความ.....	70
4.5.1 การสื่อสารแบบจุดต่อจุด.....	73
4.5.2 การสื่อสารเป็นกลุ่ม.....	74
4.6 สรุป.....	75
บทที่ 5 โปรแกรมแบบขนาน.....	77
5.1 บทนำ.....	77
5.2 รูปแบบของการพัฒนาโปรแกรมแบบขนาน.....	77
5.2.1 การสร้างโปรแกรมแบบขนาน โดยอัตโนมัติ.....	78
5.2.2 การสร้างโปรแกรมแบบขนาน โดยใช้คลังของชุดคำสั่งแบบขนาน.....	78
5.2.3 การสร้างโปรแกรมแบบขนานด้วยตัวเอง.....	79
5.3 วิธีการออกแบบอัลกอริทึมของโปรแกรมแบบขนาน.....	79
5.3.1 ขั้นตอนการแบ่งงาน.....	79
5.3.2 ขั้นตอนออกแบบการสื่อสาร.....	80
5.3.3 ขั้นตอนรวมกลุ่มงาน.....	84
5.3.4 ขั้นตอนการกำหนดงาน.....	85
5.4 กลวิธีการ โปรแกรมแบบขนาน.....	85
บทที่ 6 การทดลองและผลการทดลอง.....	89
6.1 บทนำ.....	89
6.2 ระบบคลัสเตอร์ที่ใช้ในการทดลอง.....	89
6.3 การออกแบบโปรแกรมการสร้างภาพตัดขวาง.....	90
6.4 การวัดประสิทธิภาพของการสร้างภาพตัดขวางบนระบบคลัสเตอร์.....	92
6.5 การทดลองและผลการทดลอง.....	94
6.5.1 ผลการทดลองเปรียบเทียบกับอุดมคติ.....	94

สารบัญ (ต่อ)

	หน้า
6.5.2 การทดลองเปรียบเทียบเมื่อขนาดก่อนปริมาตรต่างกัน.....	96
6.5.3 การสร้างภาพตัดขวางจากภาพเอ็กซ์เรย์.....	98
บทที่ 7 สรุปผลและวิจารณ์ผลการทดลอง.....	104
เอกสารอ้างอิง.....	106
ประวัติผู้เขียน.....	109



สารบัญตาราง

ตารางที่	หน้า
2.1 ส่วนประกอบและพิกัดของวงรีในเฟ้นทอมรูปศีรษะ.....	17
3.1 พารามิเตอร์รูปทรงเรขาคณิตสำหรับสร้าง Shepp-Logan head phantom 3 มิติ	52



สารบัญรูป

รูปที่	หน้า
2.1 หลักการสร้างภาพตัดขวางโดยวิธีการส่งผ่าน การแผ่ และการสะท้อน.....	4
2.2 फिल्मเอ็กซ์เรย์มือแบบที่เรินต์เกนค้นพบ โดยบังเอิญ	7
2.3 เครื่องทำฟลูออโรสโคปชนิดที่ใช้เครื่องปรับความเข้มของแสงและโทรทัศน์วงจรปิด	10
2.4 ลำรังสีเอ็กซ์ที่ฉายไปยังวัตถุที่มีเนื้อเดียวกันตลอด	11
2.5 ฉายรังสีไปยังวัตถุที่มีค่าสัมประสิทธิ์การดูดกลืนหลายค่า	12
2.6 ฉายรังสีไปยังวัตถุที่มีค่าสัมประสิทธิ์การดูดกลืนหลายค่าและมีการฉายรังสีหลายทิศทาง	13
2.7 แสดงภาพตัดขวางของร่างกายมนุษย์	14
2.8 โปรเจกชันของวัตถุ $f(x, y)$ ที่มุม θ	15
2.9 แพนทอมรูปสี่เหลี่ยมของ Shepp และ Logan	16
2.10 เรขาคณิตของการแปลงเรดอน	18
2.11 ทฤษฎีบทของฟูเรียร์สไลด์แสดงความสัมพันธ์ระหว่างการแปลงฟูเรียร์ของโปรเจกชันกับ การแปลงฟูเรียร์ของภาพ.....	19
2.12 Block diagram แสดงวิธีการสร้างภาพตัดขวางโดยใช้การแปลงฟูเรียร์.....	22
2.13 (a) ราวสเตอร์เชิงขั้ว (b) ราวสเตอร์เชิงขั้วมุมฉาก.....	23
2.14 การประมาณค่าจุด Q	23
2.15 โปรเจกชันของฟังก์ชัน 2 มิติ.....	24
2.16 Block diagram แสดงวิธีการสร้างภาพตัดขวางโดยใช้ฟิลเตอร์แบ็คโปรเจกชัน.....	27
2.17 ภาพแสดงผลตอบสนองทางความถี่ของ Ram-Lak Filter.....	29
2.18 ภาพแสดงรูปแบบของการฉายแสงและการคำนวณค่าแฟกเตอร์น้ำหนัก.....	31
2.19 ภาพแสดงการใช้ทฤษฎีของคาร์แมคในการแก้ระบบสมการ 2 ตัวแปร.....	36
2.20 ภาพแสดง Hyperplane ของสมการ $\vec{h}_1 \cdot \vec{f} = p_1$	37
2.21 ภาพแสดงการคำนวณหาค่าแฟกเตอร์น้ำหนัก.....	38
2.22 ภาพแสดงตัวอย่างการคำนวณหาค่าแฟกเตอร์น้ำหนัก.....	40
2.23 การประมาณค่าแบบเชิงเส้น.....	42
3.1 เส้นทางเดินแสง.....	43
3.2 ข้อมูลโปรเจกชันเมื่อพิจารณาทางเดินแสงแบบพัดที่มีระยะห่างระหว่างหัวรับเท่ากัน.....	44
3.3 ข้อมูลโปรเจกชันเมื่อพิจารณาทางเดินแสงแบบพัดที่มีระยะห่างระหว่างหัวรับเท่ากัน.....	45
3.4 ข้อมูลโปรเจกชันเมื่อพิจารณาทางเดินแสงแบบพัดที่มีระยะห่างระหว่างหัวรับเท่ากัน.....	48

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.5 เรขาคณิตของทางเดินแสงแบบพัดที่พิจารณาระยะห่างระหว่างหัวรับเท่ากัน.....	48
3.6 การสร้างภาพตัดขวางจากวัตถุ 3 มิติ.....	50
3.7 การพิจารณาทางเดินแสงแบบกรวยในรูปแบบทางเดินแสงแบบพัด.....	50
3.8 ความสัมพันธ์ของการหมุนของระบบโคออร์ดิเนต.....	51
3.9 Shepp and Logan head phantom แบบ 3 มิติสำหรับทดสอบกระบวนการ การสร้างภาพแบบกรวย.....	52
3.10 ภาพแสดงระบบพิกัด (t, s)	55
3.11 บล็อกไดอะแกรมของอัลกอริทึมเชิงพีชคณิต.....	56
3.12 ภาพแสดงเส้นทางเดินแสงแบบต่าง ๆ ในข้อมูลเชิงปริมาตร.....	58
3.13 โปรแกรมการทำงาน ART แบบเดิม.....	58
3.14 โปรแกรมการทำงานของ SART แบบเดิม.....	59
3.15 ภาพแสดงข้อมูลเชิงปริมาตรบนเส้นทางเดินแสง.....	59
4.1 ความสัมพันธ์ระหว่างวิทยาศาสตร์เชิงทฤษฎี การทดลอง และเชิงคำนวณ.....	63
4.2 องค์ประกอบของระบบคลัสเตอร์.....	66
4.3 การเชื่อมต่อเครื่องในระบบคลัสเตอร์เข้ากับระบบเครือข่ายส่วนตัว.....	67
4.4 การเชื่อมต่อระบบเครือข่ายแบบศูนย์เซิร์ฟเวอร์.....	67
4.5 แต่ละเครื่องในระบบคลัสเตอร์มีหน่วยความจำของตัวเอง.....	70
4.6 ลักษณะรหัสต้นฉบับซึ่งมีโครงสร้างแบบโปรแกรมเดี่ยวหลายข้อมูล.....	71
4.7 การจัดกลุ่มงานใน MPI.....	72
4.8 การส่งและรับข้อมูล MPI.....	73
4.9 ลักษณะการทำงานของ MPI_Bcast.....	74
4.10 ลักษณะการทำงานของ MPI_Scatter.....	74
4.11 ลักษณะการทำงานของ MPI_Gather.....	75
4.12 ลักษณะการทำงานของ MPI_Reduce.....	75
5.1 ขั้นตอนการทำงานของการสร้างโปรแกรมแบบขนานโดยอัตโนมัติ.....	78
5.2 ขั้นตอนการสร้างโปรแกรมแบบขนานโดยใช้คลังของชุดคำสั่งแบบขนาน.....	79
5.3 ขั้นตอนการสร้างโปรแกรมแบบขนานด้วยตัวเอง.....	79
5.4 การแบ่งขอบเขตข้อมูลของปัญหา.....	80

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.5 การแบ่งขอบเขตหน้าที่ของปัญหา.....	80
5.6 แผนผังการจำแนกประเภทของระบบคอมพิวเตอร์.....	81
5.7 ลักษณะการสื่อสารในการคำนวณไฟในตึกฟิฟเฟอร์นดในสองมิติ.....	82
5.8 การสื่อสารเพื่อรวมค่าจากงานที่เป็นผู้จัดการ.....	83
5.9 ตาราง (Grid) ซึ่งสร้างจากวิธีวิเคราะห์ไฟในตึกอัสซิเมนต์.....	83
5.10 การรวมกลุ่มงาน ในการประมวลผลแบบขนานเพื่อลดเวลาในการสื่อสาร.....	84
6.1 การแบ่งข้อมูลอินพุตตามความสูงของโปรเจกชัน.....	91
6.2 ข้อมูลที่ทำการแบ่งจะถูกส่งไปยังเครื่องเพื่อประมวลผล แต่ละโหนด.....	91
6.3 การทำงานของระบบศูนย์รวมงาน (Work pool).....	92
6.4 แผนภูมิแสดงเวลาที่ลดลงของการทดลองเปรียบเทียบกับระบบอุดมคติ.....	94
6.5 แผนภูมิแสดงอัตราการเพิ่มความเร็วยของการทดลองเปรียบเทียบกับระบบอุดมคติ.....	95
6.6 แผนภูมิแสดงประสิทธิภาพของระบบที่ใช้ในการทดลองเปรียบเทียบกับระบบอุดมคติ.....	95
6.7 ขนาดของปริมาณที่แบ่ง ในการทดลอง.....	96
6.8 แผนภูมิแสดงเวลาของระบบที่ใช้ในการสร้างภาพเมื่อปริมาณต่างกัน.....	96
6.9 แผนภูมิแสดงความเร็วของระบบที่ใช้ในการสร้างภาพเมื่อปริมาณต่างกัน.....	97
6.10 แผนภูมิแสดงค่าประสิทธิภาพของระบบที่ใช้ในการสร้างภาพเมื่อปริมาณต่างกัน.....	97
6.11 ตัวอย่างของข้อมูลอินพุตซึ่งเป็นรูปเอ็กซ์เรย์ รูปเท้าขวาที่มุมต่าง ๆ กัน.....	98
6.12 ภาพผลลัพธ์ซึ่งเป็นภาพตัดขวางรูปเท้าขวาที่สไลด์ต่างกันการสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบคลัสเตอร์.....	99
6.13 แผนภูมิแสดงเวลาที่ลดลงของการทดลองการสร้างภาพตัดขวางด้วยภาพถ่ายเอ็กซ์เรย์.....	100
6.14 แผนภูมิแสดงอัตราการเพิ่มความเร็วยของการสร้างภาพตัดขวางด้วยภาพถ่ายเอ็กซ์เรย์.....	100
6.15 แผนภูมิแสดงค่าประสิทธิภาพของระบบในการสร้างภาพตัดขวางด้วยภาพถ่ายเอ็กซ์เรย์.....	100
6.16 ข้อมูลผลลัพธ์การสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบคลัสเตอร์ โดยแสดงผลด้วยการสร้างภาพเชิงพื้นผิวด้วยวิธีมาร์ชชิงคิวบ์.....	101

สารบัญรูป (ต่อ)

รูปที่	หน้า
6.17 ข้อมูลผลลัพธ์การสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับวิเคราะห์ ทางเดินแสงแบบกรวยบนระบบคลัสเตอร์ โดยแสดงผลด้วยการสร้างภาพเชิงปริมาตร.....	101
6.18 ข้อมูลภาพผลลัพธ์ของภาพตัดขวางของรูปแท่งทาสไลด์ที่ 124 เปรียบเทียบกัน	102
6.19 ข้อมูลภาพผลลัพธ์เมื่อประยุกต์ใช้การสร้างภาพเชิงพื้นผิวของรูปแท่งทาสไลด์ด้วยวิธีมาร์ชชิง - คิวบ์เปรียบเทียบกัน.....	102
6.20 ข้อมูลภาพผลลัพธ์เมื่อประยุกต์ใช้การสร้างภาพเชิงปริมาตรของรูปแท่งทาสไลด์เปรียบเทียบ กัน.....	103



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันการสร้างภาพตัดขวางจากภาพถ่ายเอ็กซเรย์ มีไว้เพื่อช่วยแพทย์ในการวินิจฉัยโรค ซึ่งภาพตัดขวางนั้นสร้างมาจากภาพถ่ายเอ็กซเรย์ที่มุมต่าง ๆ รอบ วัตถุ หลังจากนั้นก็นำมาประมวลผลโดยใช้อัลกอริทึมในการสร้างภาพกลับ (Reconstruction Algorithm) เพื่อจะทำให้ข้อมูลที่อยู่ในโคเมนของภาพเอ็กซเรย์สองมิติมาอยู่ใน โคเมนของภาพสามมิติ ซึ่งอัลกอริทึมในการสร้างภาพกลับนั้นมีหลายงานวิจัยได้นำเสนอวิธีการในการสร้างภาพกลับ ซึ่งมีข้อดีข้อเสียแตกต่างกัน

ในงานวิจัยฉบับนี้มุ่งเน้นเกี่ยวกับการสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตที่มีสำหรับวิเคราะห์ทางเคมแสงแบบกรวย เนื่องจากอัลกอริทึมในการสร้างภาพกลับโดยวิธีนี้มีข้อดีหลายประการ อาทิเช่น เป็นอัลกอริทึมที่ใช้ในการสร้างภาพตัดขวางที่มีคุณภาพดีกว่าวิธีฟิลเตอร์แบ็ค โปรเจกชันเมื่อมีจำนวนมุมของข้อมูลมีจำนวนจำกัด และความถูกต้องของข้อมูลเอทพุท เป็นต้น แต่อย่างไรก็ตาม การสร้างภาพตัดขวางโดยวิธีนี้ก็ยังมีข้อเสียอยู่คือ ข้อเสียด้านเวลา ซึ่งการสร้างภาพตัดขวางโดยใช้อัลกอริทึมทางพีชคณิตนี้จะสิ้นเปลืองเวลาในการคำนวณและต้องการความสามารถในการประมวลผลของคอมพิวเตอร์ที่สูงมาก ดังนั้นในวิทยานิพนธ์ฉบับนี้จึงได้นำเสนอการสร้างภาพตัดขวางที่ใช้อัลกอริทึมเชิงพีชคณิตสำหรับวิเคราะห์ทางเคมแสงแบบกรวยโดยประมวลผลในระบบคลัสเตอร์ ซึ่งระบบคลัสเตอร์เป็นคอมพิวเตอร์แบบขนานความเร็วสูงนั้นจะสามารถช่วยเร่งความเร็วซึ่งเป็นข้อดีในการสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเคมแสงแบบกรวย

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

1. เพื่อพัฒนาให้การสร้างภาพตัดขวางทางการแพทย์ทำได้รวดเร็วขึ้นด้วยการประมวลผลบนระบบคลัสเตอร์
2. เพื่อแนะนำการสร้างและใช้งานระบบคลัสเตอร์อย่างง่าย โดยใช้ทรัพยากรที่มีอยู่ในห้องปฏิบัติการ
3. เพื่อที่จะสามารถนำหลักการที่ได้นำเสนอไปประยุกต์ใช้กับการประมวลผลอื่นๆที่ต้องใช้เวลาานได้ต่อไป

1.3 สมมติฐานของการศึกษา

การสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยเป็นอัลกอริทึมในกระบวนการการสร้างภาพกลับที่มีคุณภาพดีและมีข้อดีหลายประการ แต่กระบวนการสร้างภาพตัดขวางในวิธีนี้ต้องใช้เวลาในการประมวลผลค่อนข้างมาก ดังนั้นวิธีที่จะลดข้อด้อยของการสร้างภาพตัดขวางด้วยวิธีนี้จึงต้องทำให้เวลาในการประมวลผลสำหรับการสร้างภาพตัดขวางด้วยวิธีนี้ลดลง ซึ่งในงานวิจัยฉบับนี้ได้นำเสนอการสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยซึ่งมีข้อด้อยทางด้านเวลา มาทำงานร่วมกับระบบคลัสเตอร์ซึ่งเป็นระบบประมวลผลแบบขนานเพื่อเพิ่มความรวดเร็วเพิ่มมากขึ้น

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในงานวิจัย

ในงานวิจัยนี้ทำการแบ่งการคำนวณที่ใช้ในการสร้างภาพออกเป็นส่วนย่อยแล้วทำการกระจายไปให้เครื่องต่างๆในระบบคลัสเตอร์ประมวลผลอย่างพร้อมๆกัน เมื่อทุกโหนดประมวลผลเสร็จเรียบร้อยแล้ว ก็จะทำการส่งข้อมูลผลลัพธ์มาเก็บในเครื่องเพื่อแสดงผลในเครื่องที่กำหนดไว้ในระบบ ซึ่งในวิทยานิพนธ์ฉบับนี้ต้องคำนึงในการแบ่งงานอย่างมากเนื่องจาก การสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตนั้น ต้องใช้การประมวลผลแบบทำซ้ำ เพราะฉะนั้นในการประมวลผลในแต่ละรอบ ข้อมูลนั้นจะมีความเกี่ยวเนื่องซึ่งกันและกัน ดังนั้นจึงต้องทำการออกแบบระบบคลัสเตอร์ให้มีความสามารถในการที่จะทำงานพร้อม ๆ กันทุกโหนด

เนื่องจากงานที่ถูกแบ่งในการสร้างภาพตัดขวางไม่จำเป็นต้องมีการสื่อสารระหว่างกลุ่มงานด้วยกัน การออกแบบการสื่อสารจึงไม่มีความยุ่งยาก ซึ่งในวิทยานิพนธ์ฉบับนี้ใช้หลักการของศูนย์รวมงาน (Work pool) และใช้การสมดุลงาน (Load Balancing) เป็นแบบศูนย์กลางคำสั่ง (Centralize Dynamic load balancing) ซึ่งระบบนี้ประกอบไปด้วยคอมพิวเตอร์หลาย ๆ เครื่องเชื่อมต่อกันผ่านทางเครือข่ายความเร็วสูง และมีคอมพิวเตอร์หนึ่งทำหน้าที่เป็นผู้กระจายงานเรียกว่า Manager และทำงานไปพร้อมๆกับเครื่องอื่น ๆ ซึ่งเรียกว่า Worker และ นั่นคือ เครื่อง Manager จะเป็นผู้กำหนดงานให้กับเครื่อง Worker ทุกๆตัวและเมื่อแต่ละเครื่อง Worker ทำงานเสร็จแล้ว Worker นั้นจะการร้องขอเพื่อของงานใหม่จากเครื่อง Manager ซึ่งลำดับของคำสั่งที่ใช้กันนั้นจะมีลักษณะเป็นการวนรอบของการสั่งงาน ซึ่งเรียกการแบ่งงานแบบนี้ว่า ราวด์-โรบิน (Round robin algorithm)

1.5 ขอบเขตการวิจัย

ในงานวิจัยนี้มุ่งเน้นที่การเพิ่มความเร็วของการสร้างภาพตัดขวางทางการแพทย์ โดยกระจายงานย่อยๆไปทำการประมวลผลบนแต่ละเครื่องในระบบคลัสเตอร์ โดยรายละเอียดต่างๆภายในวิทยานิพนธ์ฉบับนี้ได้จัดแบ่งเนื้อหาออกเป็น 7 บทซึ่งแต่ละบทมีหัวข้อและเนื้อหาดังต่อไปนี้

บทที่ 1 บทนำ

อธิบายถึงความเป็นมาและความสำคัญของปัญหาตลอดจนวัตถุประสงค์และขอบเขตของวิทยานิพนธ์

บทที่ 2 ภาพตัดขวางของวัตถุและคณิตศาสตร์ที่เกี่ยวข้อง

กล่าวถึงการสร้างภาพตัดขวางภาพถ่ายเอ็กซเรย์ และคณิตศาสตร์ที่เกี่ยวข้องกับการสร้างภาพตัดขวางรวมทั้งอัลกอริธึมในการสร้างภาพตัดขวางโดยวิธีต่างๆ ได้แก่ วิธีการแปลงฟูเรียร์ วิธีเบ็คโปรเจกชัน วิธีฟิลเตอร์เบ็คโปรเจกชัน และวิธีเชิงพีชคณิต สำหรับทางเดินแสงแบบขนาน

บทที่ 3 การสร้างภาพตัดขวางที่มีทางเดินแสงแบบกรวย

กล่าวถึงการสร้างภาพตัดขวางรวมทั้งอัลกอริธึมในการสร้างภาพตัดขวางโดยวิธีต่างๆ ได้แก่ วิธีการแปลงฟูเรียร์ วิธีเบ็คโปรเจกชัน วิธีฟิลเตอร์เบ็คโปรเจกชัน และวิธีเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวย

บทที่ 4 การประมวลผลแบบขนานและโครงสร้างของระบบคลัสเตอร์

อธิบายถึงที่มาของการประมวลผลแบบขนาน การประยุกต์ใช้การประมวลผลแบบขนานในงานต่างๆ ประเภทของระบบคอมพิวเตอร์และระบบการประมวลผลแบบขนาน ตลอดจนโครงสร้างของระบบคลัสเตอร์

บทที่ 5 การโปรแกรมแบบขนาน

กล่าวถึงลักษณะของโปรแกรมแบบขนาน รูปแบบและวิธีการสร้างโปรแกรมแบบขนานซึ่งประกอบด้วยการแบ่งงาน ขั้นตอนออกแบบการสื่อสาร ขั้นตอนรวมกลุ่มงาน และขั้นตอนการกำหนดงานเป็นขั้นตอนสุดท้าย

บทที่ 6 การทดลองและผลการทดลอง

ในการทดลองได้กล่าวถึงวิธีการวัดประสิทธิภาพของโปรแกรมแบบขนาน วิธีการทดลองและผลการทดลอง

บทที่ 7 สรุปผลและแนวทางการพัฒนา
กล่าวถึงบทสรุปของวิทยานิพนธ์และแนวทางการวิจัยต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

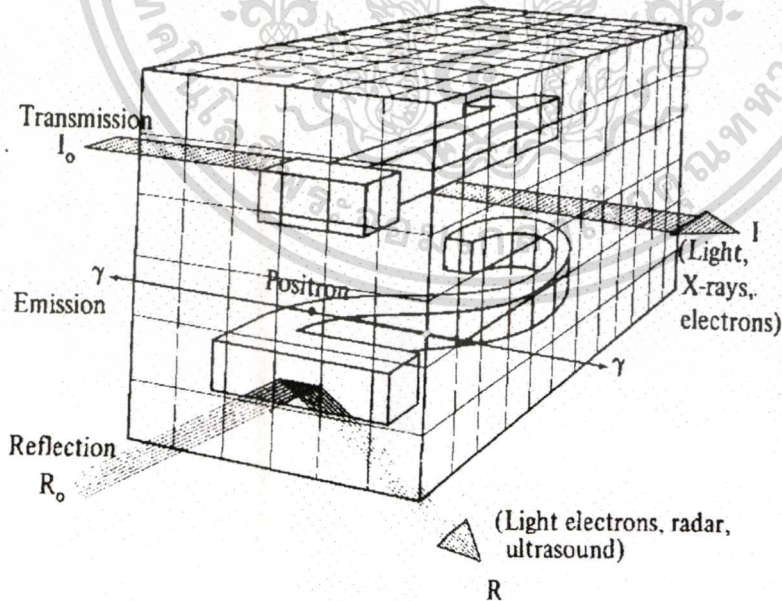
ภาพตัดขวางของวัตถุและคณิตศาสตร์ที่เกี่ยวข้อง

2.1 บทนำ

ในบทนี้จะกล่าวถึงการสร้างภาพตัดขวางด้วยวิธีการต่างๆ ภาพถ่ายเอ็กซเรย์ คณิตศาสตร์ที่เกี่ยวข้องกับการสร้างภาพตัดขวาง ซึ่งจะแบ่งออกเป็น การหาค่าสัมประสิทธิ์การดูดกลืน การอินทิเกรตในแนวเส้นและการ โปรเจกชัน และการแปลงเรดอน รวมถึงคณิตศาสตร์ที่ใช้ในการคำนวณในการสร้างภาพตัดขวางและการสร้างภาพตัดขวางสำหรับทางเดินแสงแบบขนานด้วยอัลกอริทึมในแบบต่างๆ

2.2 วิธีการสร้างภาพตัดขวาง

ปัญหาที่สำคัญอย่างหนึ่งในการประมวลผลเชิงภาพ (Image Processing) คือ การสร้างภาพตัดขวางของวัตถุโดยไม่ต้องทำลายวัตถุนั้นตัวอย่างเช่น ในรูปที่ 2.1 พิจารณารูปสี่เหลี่ยมทรงลูกบาศก์กั้นภายในประกอบด้วยเลขหนึ่งและสอง วิธีการที่ง่ายที่สุดเพื่อที่จะรู้ว่าภายในสี่เหลี่ยมคั้นนี้ประกอบด้วยอะไรบ้าง คือ การผ่าวัตถุตามแนวขวางให้ผ่านส่วนที่ต้องการดู อย่างไรก็ตาม ในทางปฏิบัติในกรณีต่างๆ เช่น การวินิจฉัยโรคทางการแพทย์ (Medical diagnosis), การสังเกตการณ์ทางดาราศาสตร์ (Astronomical observation), การทดสอบในทางอุตสาหกรรมโดยไม่ต้องทำลาย (Industrial nondestructive testing) เป็นต้น จะไม่สามารถใช้วิธีการนี้ได้



รูปที่ 2.1 หลักการสร้างภาพตัดขวางโดยวิธีการส่งผ่าน การแผ่ และการสะท้อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการสร้างภาพตัดขวางโดยไม่ต้องทำลายวัตถุ แบ่งออกเป็นหลักใหญ่ๆ 3 วิธี คือ การส่งผ่าน(Transmission) การแผ่ (Emission) และการสะท้อน (Reflection) ในการส่งผ่านเราใช้รังสีเอ็กซ์ ลำอเล็กตรอน ความร้อนหรือแสง หรือพลังงานใดๆก็ได้ที่ประพฤติตนตามกฎหมายของการดูดซึม (Absorption law) โดยพลังงานที่ผ่านเข้าไปในวัตถุถูกดูดซึมในส่วนต่างๆของวัตถุไม่เท่ากัน พลังงานส่วนที่ทะลุออกมา เรียกว่า โปรเจกชัน และภาพตัดขวางที่ได้จากวิธีการนี้ คือ ภาพตัดขวางจากเครื่องเอ็กซ์เรย์คอมพิวเตอร์ ในวิธีการแผ่จะใช้การแผ่รังสีที่มีกัมมันตภาพรังสีเป็นส่วนประกอบ เมื่อสารนี้ไปยังอวัยวะที่สนใจก็จะแผ่รังสีแกมมาในทิศทางที่แตกต่างกัน ความแตกต่างของเวลาที่ใช้ในการเดินทางของรังสีแกมมาที่ผ่านตำแหน่งเดียวกัน จะสามารถนำไปประมวลผลหาตำแหน่งของจุดนั้นได้ ภาพตัดขวางที่ได้จากวิธีนี้คือ ภาพตัดขวางจากเครื่องโพซิตรอนอิมิตชันโทโมกราฟี (Positron Emission Tomography หรือ P.E.T) และ เครื่องซิงเกิลโฟตอนอิมิตชันโทโมกราฟี (Single Photon Emission Computed Tomography หรือ S.P.E.C.T) ส่วนการสะท้อนจะใช้คุณสมบัติการสะท้อนของผิวหน้าของวัตถุมาตรวจสอบหาลักษณะของวัตถุได้ แหล่งพลังงานที่ใช้ เช่น แสง ลำอเล็กตรอน เรดาร์ หรือ อุลตราโซนิค ภาพตัดขวางที่ได้จากวิธีนี้คือ ภาพตัดขวางจากเครื่องอัลตราซาวด์ (Ultrasound)

2.3 ภาพถ่ายเอ็กซ์เรย์

รังสีเอ็กซ์หรือที่เรียกกันทั่วๆ ไปอย่างแพร่หลายว่า เอ็กซ์เรย์ เป็นพลังงานรูปหนึ่งเช่นเดียวกับแสงสว่าง แต่เป็นชนิดที่เราไม่สามารถมองเห็นได้ด้วยตาเปล่า รังสีเอ็กซ์นี้ไม่ได้เกิดเองตามธรรมชาติ มนุษย์ต้องสร้างขึ้นโดยใช้เครื่องมือชนิดหนึ่ง เรียกว่า เครื่องเอ็กซ์เรย์

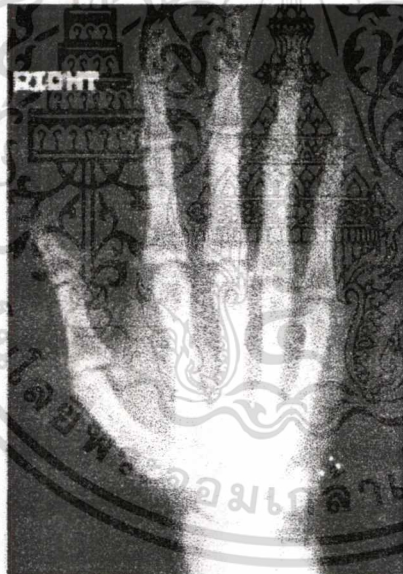
เมื่อเราป่วยไข้ เราได้ยินเสมอๆว่าแพทย์สั่งให้เราไปเอ็กซ์เรย์ เพื่อหาสาเหตุของการป่วย พนักงานรังสีวิทยา ก็จะพาเราเข้าห้องเอ็กซ์เรย์ ฉายรังสีเอ็กซ์ทะลุตัวเราไปถูกฟิล์มเอ็กซ์เรย์ ซึ่งเหมือนกับถ่ายรูปแต่ แผ่นใหญ่กว่า แม้รังสีเอ็กซ์จะผ่านทะลุตัวเราแต่เราก็ไม่รู้สึอะไรเลย ต่อมาเจ้าหน้าที่ก็นำฟิล์มไปล้างตามกรรมวิธีในห้องมืด ไม่ช้าก็ได้ภาพที่มองเห็นเนื้อหนังกระดูก คับ ไต ลำไส้ และอื่น ๆ ปรากฏอยู่บนฟิล์มเอ็กซ์เรย์ แพทย์รังสีซึ่งเป็นผู้เชี่ยวชาญในการใช้รังสีเอ็กซ์ทางแพทย์ จะเป็นผู้อ่านฟิล์มเหล่านั้น เพื่อทำรายงานการวินิจฉัยว่า เรามีโรคหรือสิ่งผิดปกติใดๆอยู่ในร่างกายหรือไม่ เมื่อแพทย์ผู้รักษาได้อ่านรายงานของแพทย์รังสีแล้ว ก็จะรักษาโรคของเราได้ถูกต้องด้วยเหตุนี้ จึงนับได้ว่ารังสีวิทยามีประโยชน์มากมาย และจำเป็นอย่างยิ่งในการแพทย์แผนปัจจุบัน

2.3.1 ประวัติการค้นพบเอ็กซ์เรย์[1]

มีการค้นพบรังสีเอ็กซ์ เมื่อวันที่ 8 พฤศจิกายน พ.ศ. 2438 โดยศาสตราจารย์วิลเฮล์ม คอนราด เรินด์เกน(Wilhelm Konrad Roenigen, ค.ศ. 1845-1923 ชาวเยอรมัน) อาจารย์สอนวิชาฟิสิกส์ แห่งมหาวิทยาลัย วูร์ซบูร์ก (Wurzberg) ประเทศเยอรมนี ในเวลาใกล้ ๆ กัน เบกเคอเรล (Antonic

Becquerel, ค.ศ. 1852-1908 ชาวฝรั่งเศส) ค้นพบว่า แร่ยูเรเนียมมีกัมมันตภาพรังสีและมาตามคูรี (Madame Marie Curie, ค.ศ. 1867-1934 ชาวฝรั่งเศส) ค้นพบแร่กัมมันตรังสี เรเดียม

รังสีที่เรินต์เกนค้นพบนี้เขาเรียกว่า รังสีเอ็กซ์ หรือ เอ็กซ์เรย์ (X-ray) หมายความว่า เป็นรังสีใหม่ที่ซึ่งไม่มีใครรู้จัก ในภายหลังจึงมีผู้เรียกรังสีเอ็กซ์นี้ว่า รังสีเรินต์เกน เพื่อเป็นเกียรติแก่ผู้ค้นพบ รังสีเรินต์เกนมีคุณสมบัติทางกายภาพหลายอย่างที่ทำให้มีลักษณะที่ทั้งเหมือนและแตกต่างจากรังสีอื่นๆ เรินต์เกนพบว่า รังสีเอ็กซ์สามารถฉายทะลุวัตถุทึบแสง เช่น ร่างกายมนุษย์ได้ เขาจึงลองเอารังสีเอ็กซ์มาใช้ประโยชน์ทางการแพทย์ โดยใช้รังสีเอ็กซ์ผ่านมือคน เนื้อ เอ็น และกระดูกในมือคน ค้นพบรังสีเอาไว้บางส่วนปล่อยให้ผ่านไปบางส่วน ส่วนเนื้อและเอ็นที่รังสีได้น้อยมากก็มีรังสีออกมาเยอะ กระดูกที่รังสีได้มากก็มีรังสีผ่านออกมาน้อย รังสีทั้งหมดที่ผ่านมือออกมาจะมีความเข้มต่อพื้นที่ไม่เท่ากัน ทำให้เกิดเป็นรูปแบบ (pattern) ของความเข้มของรังสีในรูปมือขึ้น เมื่อเอารังสีที่มีรูปแบบแล้วนี้ไปกระทบกับวัสดุที่ไวต่อแสง เช่น ฟิล์มถ่ายภาพหรือกระดาษอัดรูป (photographic material) แล้วนำไปล้างด้วยน้ำยาล้างรูป ก็จะเกิดภาพของมือที่มีกระดูกซ่อนอยู่ในเนื้อ แพทย์จึงสามารถบอกได้ว่ากระดูกข้างในมือนั้นหักหรือไม่ โดยไม่จำเป็นต้องผ่าเอาเนื้อที่หุ้มกระดูกออกมาดังแสดงในรูปที่ 2.2



รูปที่ 2.2 แสดงภาพฟิล์มเอ็กซ์เรย์มือแบบที่เรินต์เกนค้นพบโดยบังเอิญ

2.3.2 การเกิดรังสีเอ็กซ์

รังสีเอ็กซ์มิได้เกิดขึ้นโดยธรรมชาติ แต่เป็นรังสีที่มนุษย์สร้างขึ้น โดยใช้ปรากฏการณ์อันหนึ่งที่ภาษาเยอรมันเรียกว่า เบรมม์ชตราห์ลุง (bremmstrahlung) ซึ่งแปลว่า สกัดกันหรือทำให้ช้าลงหรือทำให้หยุด (brakingradiation) เพราะรังสีเอ็กซ์ส่วนใหญ่ เกิดขึ้นจากการที่อิเล็กตรอน ซึ่งเป็นวัสดุที่มีมวลและมีน้ำหนัก (อิเล็กตรอน 1 ตัวมีมวล 9.11×10^{-28} กรัม) วิ่งไปกระทบกับโลหะทั้งสเดน

และถูกทั้งสแตนด์สแตนด์ไว้ จนวิ่งช้าลงหรือจนหยุด ทำให้อิเล็กตรอนคายพลังงานจลน์ (kinetic energy) ของมันออกมา ตามกฎที่ว่าพลังงานย่อมไม่สูญหาย กล่าวคือ เมื่ออิเล็กตรอนกำลังวิ่ง มีพลังงาน 2 รูป คือ พลังงานศักย์ (potential energy) และพลังงานจลน์พอถูกทั้งสแตนด์สแตนด์ให้หยุดจะเหลือแต่พลังงานศักย์ ส่วนพลังงานจลน์ไม่สูญหาย แต่เปลี่ยนรูปไปเป็น พลังงานใหม่อีก 2 รูป คือ ส่วนใหญ่ (มากกว่า 99%) เป็นความร้อน และ ส่วนน้อย (น้อยกว่า 1%) เป็นพลังงานในรูปของคลื่นแม่เหล็กไฟฟ้าที่เรียกว่า รังสีเอ็กซ์ รังสีเอ็กซ์ที่เกิดขึ้นนี้เป็นส่วนผสมของรังสีเอ็กซ์ที่มีความยาวคลื่นต่างๆกัน ตั้งแต่ความยาวคลื่นสั้นที่สุดซึ่งมีพลังงานสูงสุดที่เกิดจากอิเล็กตรอนที่ถูกทำให้หยุด และความยาวคลื่นปานกลางขนาดต่างๆ ไปจนถึงความยาวคลื่นที่ยาวมากๆ ซึ่งเป็นรังสีเอ็กซ์ที่มีพลังงานต่ำ และยังมีพลังงานจลน์เหลืออยู่ ส่วนผสมของรังสีเอ็กซ์นี้เรียกว่า เอ็กซ์เรย์สเปกตรัม (X-ray spectrum) รังสีเอ็กซ์ที่มีคลื่นสั้นมีพลังงานสูง จึงมีอำนาจทะลุทะลวงสูงกว่ารังสีเอ็กซ์ที่มีความยาวคลื่นยาว

รังสีเอ็กซ์ยังเกิดขึ้นได้อีกวิธีหนึ่ง คือ เมื่ออิเล็กตรอนหลายวงที่วิ่งวนรอบนิวเคลียสของอะตอมกระโดดจากวงหนึ่งที่มีระดับพลังงานสูงกว่า ไปสู่วงอื่นที่มีระดับพลังงานต่ำกว่า จะคายพลังงานออกมาในรูปของรังสีเอ็กซ์ ที่มีพลังงานเฉพาะและคงที่ แล้วแต่ลักษณะอะตอมของธาตุหนึ่ง ๆ รังสีเอ็กซ์ที่ได้มาด้วยวิธีนี้ จึงเรียกว่า รังสีลักษณะเฉพาะ (Characteristic radiation)

ในการเกิดรังสีเอ็กซ์นี้อาศัยองค์ประกอบสำคัญ 5 ประการ คือ

1. ทำให้เกิดอนุภาคอิเล็กตรอน หรือการแยกอิเล็กตรอนออกจากอะตอมของโลหะ

อิเล็กตรอนจะเกิดขึ้น หรือแยกจากอะตอมของโลหะที่ทำเป็นไส้หลอดเอ็กซ์เรย์ โดยการที่เราผ่านกระแสไฟฟ้า เข้าไปในไส้หลอด (Filament) จนกระทั่งไส้หลอดร้อนขึ้น ประมาณ 2000°C หรือมากกว่านั้น จะมีผลทำให้เกิดเทอร์มิออนิกอิมิชัน (Thermionic Emission) คือ อิเล็กตรอนหลุดออกจาก เซลล์หรือวงโคจรของมันเมื่อถูกความร้อน อิเล็กตรอนที่หลุดออกมา นี้จะมอดกันอยู่รอบๆ ผิวหน้าของโลหะ จนกลายเป็นกลุ่ม (Cloud) เรียกว่า Space charge สาเหตุที่อิเล็กตรอนไม่สามารถหลุดพ้น ไปจากผิวหน้าโลหะ เพราะว่ามีแรงดึงดูดระหว่าง อิเล็กตรอนและอะตอม ของโลหะยังมีอยู่

2. การทำให้อนุภาคอิเล็กตรอนหลุด และเคลื่อนที่ไปด้วยความเร็วสูง

วิธีที่จะทำให้อนุภาคอิเล็กตรอนหลุดจากอะตอมไปได้ คือ ต้องหาสิ่งที่มีพลังงาน หรือแรงดึงดูด มากกว่าแรงดึงดูดระหว่างอิเล็กตรอนกับ อะตอมของโลหะ มาดึงดูดอิเล็กตรอนตั้งนั้นก็คือ ไฟฟ้าแรงสูง (High voltage) ไฟฟ้าแรงสูงนี้ นอกจากจะทำให้เกิดอิเล็กตรอนหลุดจากผิวหน้าของโลหะแล้ว ยังทำให้อิเล็กตรอนเคลื่อนที่หรือวิ่งไปด้วยความเร็วสูง กล่าวอีกอย่างหนึ่งคือ ต้องทำให้ไฟฟ้าระหว่างขั้วบวกและขั้วลบให้ต่างกัน ความต่างศักย์นี้ ถ้ายังมีค่ามากขึ้นเท่าใด ก็ยิ่งทำให้อิเล็กตรอนหลุดจากผิวหน้าของไส้หลอด ได้มาก และวิ่งไปด้วยความเร็วสูงขึ้น ความต่างศักย์ 100 กิโลโวลต์ จะทำให้อิเล็กตรอนมีความเร็วประมาณ 165 กิโลเมตรต่อวินาที

3. การทำให้เส้นทางที่อิเล็กตรอนวิ่งผ่านไป ต้องไม่มีสิ่งกีดขวาง

จึงจะทำให้ไอเล็กตรอนวิ่งไปด้วยความเร็วสูง ดังนั้นภายในหลอดเอ็กซ์เรย์ จึงเป็นสุญญากาศ เพราะถ้ามีอากาศ หรือก๊าซอยู่ภายในหลอด แม้เพียงเล็กน้อยก็ตาม ความเร็วของไอเล็กตรอนจะลดลง เนื่องจากเกิดไอออนไนเซชัน (Ionization)

4. การทำให้ไอเล็กตรอนมีความเข้มข้น (Concentration of electron)

จะต้องมีวิธีการที่จะให้ไอเล็กตรอนเคลื่อนที่ไปในแนวทิศทางเดียวกัน คือ พุ่งไปหาจุดโฟกัสของเป้าด้วยปริมาณ (ความเข้มข้น) ที่มากพอสมควร ซึ่งทำให้ได้โดยการใช้เครื่องมือที่เรียกว่า “Electron focusing device” คอยควบคุม ให้ไอเล็กตรอนส่วนใหญ่ ไปตกในบริเวณจุดโฟกัสของเป้า

5. การทำให้ไอเล็กตรอนหยุดวิ่งในทันทีทันใด

การที่จะทำให้ไอเล็กตรอนที่วิ่งมาด้วยความเร็วสูงนี้หยุดวิ่งในทันทีทันใด ทำได้โดยหาวัตถุมารกั้น หรือขวางไอเล็กตรอนวัตถุนั้นก็คือ เป้า (Target) ผลที่เกิดขึ้น คือ เกิดการชน หรือ กระทบกันระหว่างไอเล็กตรอนกับเป้าอย่างแรง ไอเล็กตรอนดังกล่าว จะไปชนไอเล็กตรอนที่อยู่ในวงโคจร ของอะตอมของเป้านั้น ให้หลุดกระเด็นออกนอกวงโคจร และจะมีไอเล็กตรอนจากวงโคจรอื่นๆ ที่อยู่ถัดไป วิ่งเข้ามาแทน แต่เนื่องจาก พลังงาน ของไอเล็กตรอนในแต่ละวงโคจร จะไม่เท่ากัน ดังนั้นจึงมีการคายพลังงานส่วนเกินออกมา ในรูปของเอ็กซ์เรย์และความร้อน (ส่วนใหญ่จะเป็นความร้อนมากกว่า) กล่าวคือ จากพลังงานของไอเล็กตรอนทั้งหมด ที่วิ่ง ไปสู่เป้านั้น 99.8% จะเปลี่ยนเป็นความร้อน และ 0.2 % เป็นเอ็กซ์เรย์

2.3.3 ประโยชน์ของรังสีเอ็กซ์เรย์ทางการแพทย์ [1]

รังสีเอ็กซ์ เมื่อฉายทะลุอวัยวะที่ต้องการตรวจแล้ว จะเกิดเป็นรูปแบบขึ้นบนฟิล์มเอ็กซ์เรย์ เมื่อนำฟิล์มเอ็กซ์เรย์ไปล้างตามกรรมวิธี จะได้ภาพทั้งภายนอกและภายในของอวัยวะ เช่น กระดูกที่ฝังอยู่ในเนื้อ หรือ แผลฉีก โรคที่อยู่ในเนื้อปอด ทำให้วินิจฉัยโรคได้โดยไม่ต้องผ่าอวัยวะนั้นเข้าไปดูภายใน ถ้าฉายรังสีที่มีรูปแบบแล้วนี้ไปถูกกระจกที่ฉาบด้วยแบเรียมแพลตทินไซอะไนด์ หรือ แคลเซียมทังสเตนในห้องมืด ก็จะเห็นภาพของอวัยวะภายในได้ทันที แต่ภาพนี้ไม่ค่อยจะชัดเจน เพราะแสงเรืองที่เกิดขึ้นนั้นอ่อนมาก จึงต้องดูในห้องมืด และรังสีแพทย์ผู้ตรวจต้องปิดตาหรืออยู่ในห้องมืดเสียก่อนราว 20 นาที จนตาคุ้นกับความมืด จึงจะดูเห็น การตรวจด้วยวิธีนี้เรียกว่า ฟลูออโรสโคปี (fluoroscopy) ซึ่งในปัจจุบันนี้ มีผู้นำภาพจากแสงเรืองๆในห้องมืดนี้มาฉายผ่านเครื่องปรับความเข้มของแสงที่เรียกว่า อิมเมจอินเทนซิไฟเออร์ (image intensifier) แล้วนำสัญญาณภาพไปฉายออกทางจอโทรทัศน์ ทำให้รังสีแพทย์ทำการตรวจอวัยวะภายในโดยการทำให้ฟลูออโรสโคปีในห้องที่มีแสงสว่างธรรมดาได้ และชัดเจนกว่าการทำในห้องมืด การตรวจด้วยวิธีนี้ แพทย์รังสีจะเห็นการเคลื่อนไหวของอวัยวะภายในได้สะดวก ตัวอย่างที่เห็นได้ชัดเจน คือ การตรวจกระเพาะอาหาร ถ้าใส่ และถ่วงน้ำดี รูปที่ 2.3 แสดงถึงเครื่องทำฟลูออโรสโคปีชนิดที่ใช้เครื่องปรับความเข้มของแสงและโทรทัศน์วงจรปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อวัยวะบางอย่างที่ตรวจพบได้ไม่ชัดเจน แพทย์อาจทำให้เห็นชัดเจนขึ้นได้โดยใช้สารทึบแสงบางอย่างสารเหล่านี้พอจะจำแนกได้ ดังนี้

1. สารทึบแสงที่ทึบน้อยกว่าเนื้อ ได้แก่ ก๊าซต่างๆ เช่น อากาศ คาร์บอนไดออกไซด์ และไนโตรสออกไซด์
2. สารทึบแสงที่ทึบมากกว่าเนื้อ ได้แก่ สารประกอบไอโอดีน และแบเรียมซัลเฟต (barium sulphate)



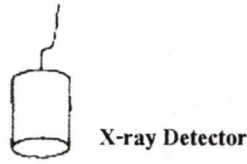
รูปที่ 2.3 เครื่องทำฟลูออโรสโคปชนิดที่ใช้เครื่องปรับความเข้มของแสงและโทรทัศน์วงจรปิด

2.4 กลนิตศาสตร์ที่เกี่ยวข้องกับคอมพิวเตอร์ถ่ายภาพตัดขวาง

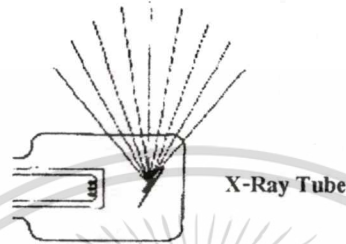
2.4.1 การหาค่าสัมประสิทธิ์การดูดกลืน

จุดมุ่งหมายของคอมพิวเตอร์ถ่ายภาพตัดขวาง คือ การแมปค่าสัมประสิทธิ์การดูดกลืนของการฉายรังสีเอ็กซ์ไปยังวัตถุ ซึ่งเราจะฉายรังสีเอ็กซ์ไปยังวัตถุ และตรวจจับรังสีเอ็กซ์ที่ถูกดูดกลืน ซึ่งค่าของรังสีเอ็กซ์ที่ถูกดูดกลืน จะเรียกว่า ข้อมูลโปรเจกชัน ซึ่งการหาค่าสัมประสิทธิ์การดูดกลืน แบ่งออกเป็น 3 กรณี คือ

2.4.1.1 กรณีที่มีค่าสัมประสิทธิ์การดูดกลืน 1 ค่า และการฉายแสงไปยังวัตถุที่มีเนื้อเดียวกันตลอด



X-ray Detector



X-Ray Tube

รูปที่ 2.4 ลำรังสีเอ็กซ์ที่ฉายไปยังวัตถุที่มีเนื้อเดียวกันตลอด

พิจารณารูปที่ 2.4 ถ้าเราทำการฉายรังสีเอ็กซ์ไปยังวัตถุ จะได้ความสัมพันธ์ระหว่าง $N, \Delta N, \Delta x$ และ μ ดังสมการที่ 2.1

$$\frac{\Delta N}{N} \cdot \frac{1}{\Delta x} = -\mu \quad (2.1)$$

ซึ่ง N คือ จำนวนโฟตอนที่ออกจากวัตถุ

N_0 คือ จำนวนโฟตอนที่ฉายเข้าไปในวัตถุ

μ คือ ค่าสัมประสิทธิ์การดูดกลืน

x คือ ความหนาของวัตถุ

จากนั้น จึงหาปริมาตร โดยให้ Δx มีค่าเข้าใกล้ 0 จะได้ดังสมการที่ 2.2

$$\frac{1}{N} dN = -\mu dx \quad (2.2)$$

อินทิเกรตทั้ง 2 ข้างของสมการ จะได้

$$\int_{N_0}^N \frac{dN}{N} = -\mu \int_0^x dx \quad (2.3)$$

จัดรูปสมการใหม่ โดยการเอาลอการิธึมคูณทั้ง 2 ข้าง จะได้

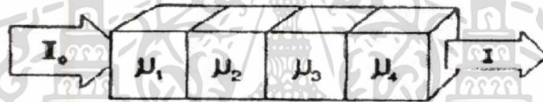
$$\ln N - \ln N_0 = -\mu x \quad (2.4)$$

จัดสมการใหม่ จะได้ความสัมพันธ์ระหว่างโฟตอนที่ฉายไปยังวัตถุ กับโฟตอนที่ออกจากวัตถุ ดังสมการที่ 2.5

$$N = N_0 e^{-\mu x} \quad (2.5)$$

2.4.1.2 กรณีที่มีค่าสัมประสิทธิ์การดูดกลืนหลายค่า แต่มีการฉายแสงไปยังวัตถุเพียง 1

ครั้ง



รูปที่ 2.5 ฉายรังสีไปยังวัตถุที่มีค่าสัมประสิทธิ์การดูดกลืนหลายค่า

พิจารณารูปที่ 2.5 เป็นการฉายรังสีเอ็กซ์ผ่านชิ้นส่วนของเนื้อเยื่อซึ่งมีค่าสัมประสิทธิ์การดูดกลืน 4 ค่า คือ μ_1 , μ_2 , μ_3 และ μ_4 จะได้ความสัมพันธ์ระหว่าง I กับ I_0 ดังสมการที่ 2.8

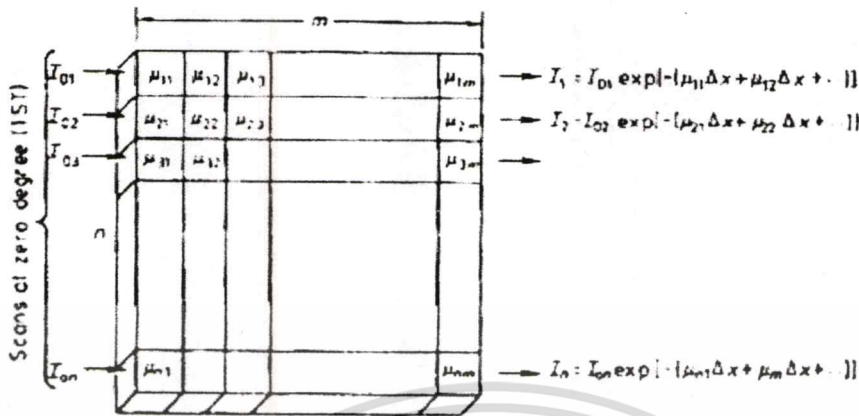
$$I = I_0 \exp[-(\mu_1 + \mu_2 + \mu_3 + \mu_4)x] \quad (2.6)$$

ซึ่ง x คือ ระยะทางของลำอเล็กตรอน จากนั้นจึงใช้ลอการิธึม คูณทั้งสองข้าง จะได้

$$p(x) = \sum_{i=1}^4 \mu_i \quad (2.7)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1.3 กรณีที่มีค่าสัมประสิทธิ์การดูดกลืนหลายค่า และมีการฉายรังสีหลายทิศทาง



รูปที่ 2.6 ฉายรังสีไปยังวัตถุที่มีค่าสัมประสิทธิ์การดูดกลืนหลายค่าและมีการฉายรังสีหลายทิศทาง

พิจารณารูปที่ 2.6 จะได้ความสัมพันธ์ระหว่างโฟตอนที่ฉายไปยังวัตถุกับโฟตอนที่ออกจากวัตถุ ดังสมการที่ 2.8

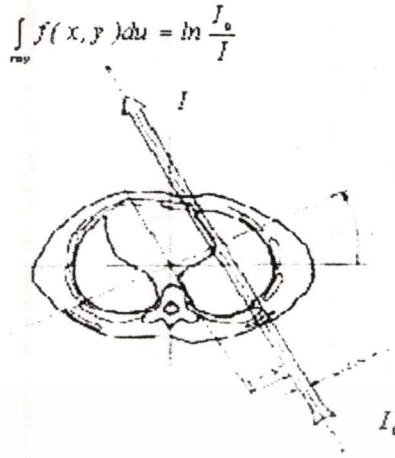
$$\begin{aligned}
 I_1 &= I_{01} \exp[-(\mu_{11}\Delta x + \mu_{12}\Delta x + \mu_{13}\Delta x + \dots + \mu_{1m}\Delta x)] \\
 I_2 &= I_{02} \exp[-(\mu_{21}\Delta x + \mu_{22}\Delta x + \mu_{23}\Delta x + \dots + \mu_{2m}\Delta x)] \\
 &\vdots \\
 I_n &= I_{0n} \exp[-(\mu_{n1}\Delta x + \mu_{n2}\Delta x + \mu_{n3}\Delta x + \dots + \mu_{nm}\Delta x)]
 \end{aligned}
 \tag{2.8}$$

จัดรูปสมการใหม่ จะได้

$$p_i(x) = \mu_{i1} + \mu_{i2} + \mu_{i3} + \dots + \mu_{im}
 \tag{2.9}$$

ต่อมาพิจารณารูปที่ 2.7 เป็นภาพตัดขวางของร่างกายมนุษย์ ซึ่งจะถือว่าเป็นวัตถุที่มีสารหลายชนิดประกอบกัน จะได้ความสัมพันธ์ระหว่าง I_0 กับ I ดังสมการที่ 2.10

$$I = I_0 \exp \left[- \int_{ray} f(x, y) du \right]
 \tag{2.10}$$



รูปที่ 2.7 แสดงภาพตัดขวางของร่างกายมนุษย์

จากนั้น จึงจัดรูปสมการใหม่ จะได้สมการที่ 2.11

$$\int_{ray} f(x, y) du = \ln \frac{I_0}{I} \quad (2.11)$$

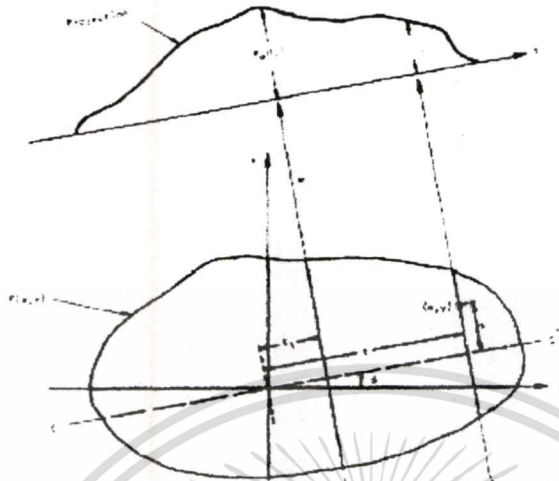
- ซึ่ง I_0 คือ จำนวนโฟตอนที่ฉายเข้าไปในวัตถุจากด้าน A
 I คือ จำนวนโฟตอนที่ออกจากวัตถุทางด้าน B
 du คือ ระยะทางตลอดความยาวของรังสี
 $f(x, y)$ คือ ค่าสัมประสิทธิ์การดูดกลืน

อย่างไรก็ตาม ในคอมพิวเตอร์ถ่ายภาพตัดขวางนั้น การส่งค่าสัมประสิทธิ์การดูดกลืน $f(x, y)$ นั้น เราจะต้องทำการฉายรังสีไปยังวัตถุมากกว่า 1 มุม จากการทำมุมแหล่งกำเนิดรังสี เอ็กซ์และตัวตรวจจับนั้น จะได้ข้อมูลโปรเจกชันที่มุมแตกต่างกัน ซึ่งข้อมูลโปรเจกชันนี้เราจะนำไปใช้ในการสร้างภาพตัดขวางของวัตถุ

2.4.2 การอินทิเกรตในแนวเส้น (Line Integrals) และการโปรเจกชัน (Projections)

การอินทิเกรตในแนวเส้น เป็นการแสดงถึงการอินทิเกรตของตัวแปรของวัตถุตลอดแนวเส้น จากรูปที่ 2.8 แสดงถึงใช้ระบบพิกัด (Coordinate system) อธิบายการอินทิเกรตในแนวเส้นและการโปรเจกชัน จากภาพ สมมติให้ $f(x, y)$ คือวัตถุรูปร่าง 2 มิติ และสมการของเส้น AB คือ

$$x \cos \theta + y \sin \theta = t \quad (2.12)$$



รูปที่ 2.8 โปรเจกชันของวัตถุ $f(x, y)$ ที่มุม θ

จากความสัมพันธ์ใน (2.12) จะได้นิยามของการอินทิเกรตในแนวเส้น คือ

$$P_\theta(t) = \int_{(\theta, t) \text{ line}} f(x, y) ds \quad (2.13)$$

ใช้เคลด้า ฟังก์ชัน เขียนใหม่ได้เป็น

$$P_\theta(t) = \int_{-\alpha}^{\alpha} \int_{-\alpha}^{\alpha} f(x, y) \delta(x \cos \theta + y \sin \theta - t) dx dy \quad (2.14)$$

การจำลองการโปรเจกชันในคอมพิวเตอร์จะใช้รูปภาพดังในรูปที่ 2.9 เรียกว่า Shepp and Logan หรือแฟนทอมรูปศีรษะ (head phantom) ซึ่งจะนำไปใช้ในการทดสอบอัลกอริทึมในการสร้างภาพกลับจากภาพตัดขวาง จากรูปที่ 2.9 (a) จะเห็นว่ารูปแฟนทอมนี้ประกอบด้วยวงรีจำนวน 10 รูป ตำแหน่งและส่วนประกอบอื่นๆของวงรีทั้งหมดได้แสดงไว้ในรูปที่ 2.9 (b) และตารางที่ 2.5 [14]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(a)



(b)

รูปที่ 2.9 แพนทอมรูปครีมาของ Spepp และ Logan

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 ส่วนประกอบและพิกัดของวงรีในแผนทอรูปสี่รียะ

Center Coordinate	Major Axis	Min Axis	Rotation Angle	Reflective Index
(0,0)	0.92	0.69	90	2.0
(0, -0.0184)	0.874	0.6624	90	-0.98
(0.22,0)	0.31	0.11	72	-0.02
(-0.22,0)	0.41	0.16	108	0.01
(0,0.35)	0.25	0.21	90	0.01
(0,0.1)	0.046	0.046	0	0.01
(0, -0.1)	0.046	0.046	0	0.01
(-0.08, -0.605)	0.046	0.023	0	0.01
(0, -0.605)	0.023	0.023	0	0.01
(0.06, -0.605)	0.046	0.023	-90	0.01

2.4.3 การแปลงเรดอน (Radon Transform)

การแปลงเรดอนของฟังก์ชัน $f(x, y)$ เขียนแทนด้วย $g(s, \theta)$ ซึ่งนิยามคือการอินทิเกรตในแนวเส้นตลอดเส้น ทึ่มุม θ และระยะ s จากจุดกำเนิด ดังแสดงในรูปที่ 2.10 สามารถเขียนเป็นสมการทางคณิตศาสตร์ คือ

$$g(s, \theta) = Rf = \iint_{-\infty}^{\infty} f(x, y) \delta(x \cos \theta + y \sin \theta - s) dx dy, \quad (2.15)$$

$$-\infty < s < \infty, 0 \leq \theta < \pi$$

สัญลักษณ์ R คือ ตัวกระทำการแปลงเรดอน (Radon transform operator) การแปลงเรดอนของฟังก์ชัน $f(x, y)$ หรือ $g(s, \theta)$ สามารถเขียนสมการใหม่ให้อยู่ในระบบพิกัดการหมุน (s, u) (Rotating coordinate system) ซึ่ง

$$s = x \cos \theta + y \sin \theta$$

$$u = -x \sin \theta + y \cos \theta$$

หรือ

$$x = s \cos \theta - u \sin \theta$$

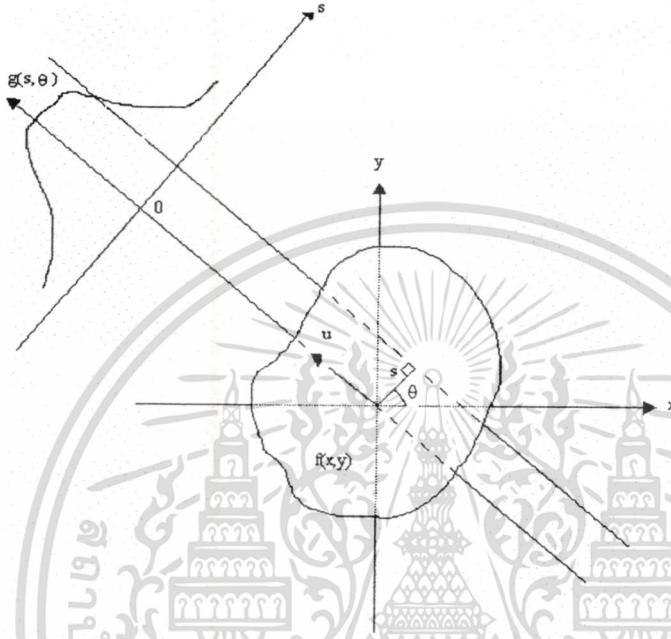
$$y = s \sin \theta + u \cos \theta$$

(2.16)

จากสมการที่ (2.15) สามารถเขียนใหม่ได้เป็น

$$g(s, \theta) = \int_{-\infty}^{\infty} f(s \cos \theta - u \sin \theta, s \sin \theta + u \cos \theta) du \quad (2.17)$$

$$-\infty < s < \infty, 0 \leq \theta < \pi$$



รูปที่ 2.10 เรขาคณิตของการแปลงเรดอน

2.5 อัลกอริธึมของการสร้างภาพตัดขวาง

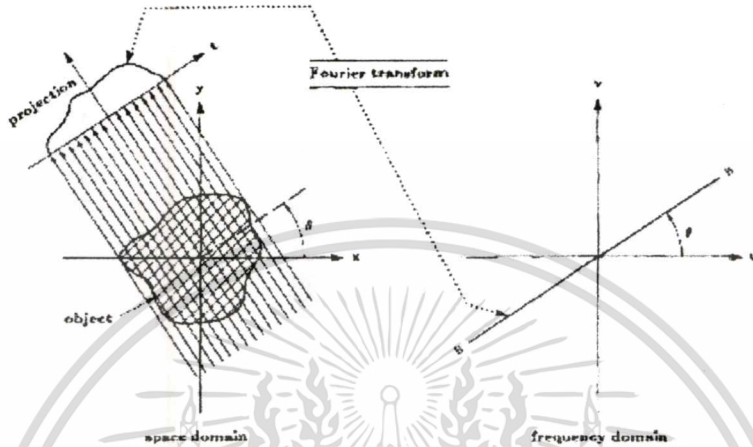
จากที่กล่าวมาข้างต้น ได้ทราบถึงประวัติความเป็นมา ส่วนประกอบและการทำงานของคอมพิวเตอร์ถ่ายภาพตัดขวาง และคณิตศาสตร์ที่เกี่ยวข้องการสร้างภาพตัดขวางไปแล้ว ต่อไปจะกล่าวถึงอัลกอริธึมที่ใช้ในการสร้างภาพตัดขวาง ในวิธีการสร้างภาพตัดขวางนั้นจะสามารถแบ่งออกเป็น 2 ประเภทใหญ่ๆ ได้เป็น การสร้างภาพตัดขวางที่มีทางเดินแสงแบบขนาน (Parallel-Beam) และ การสร้างภาพตัดขวางที่มีทางเดินแสงแบบรูปกรวย (Cone-Beam)

2.6 การสร้างภาพตัดขวางที่มีทางเดินแสงแบบขนาน

อัลกอริธึมที่ใช้ในการสร้างภาพตัดขวางที่มีทางเดินแสงแบบขนานนั้นประกอบด้วยหลายวิธี ได้แก่ วิธีการแปลงฟูเรียร์, วิธีแบค โปรเจกชัน และวิธีพิวคเตอร์แบค โปรเจกชัน

2.6.1 วิธีการสร้างภาพตัดขวางโดยใช้การแปลงฟูเรียร์

เป็นวิธีการสร้างภาพตัดขวางโดยใช้นิยามจากทฤษฎีบทของฟูเรียร์สไลด์ (The Fourier Slice Theorem) ซึ่งระบุไว้ว่า เมื่อเราทำการหาโปรเจกชันของภาพ $f(x,y)$ ที่มุมต่าง $\theta_1, \theta_2, \dots, \theta_k$ แล้วทำการแปลงฟูเรียร์โปรเจกชันเหล่านั้น จะมีค่าเท่ากับการแปลงฟูเรียร์สองมิติของภาพตามแนวเส้นที่ทำมุม $\theta_1, \theta_2, \dots, \theta_k$ เช่นกัน ดังรูปที่ 2.11



รูปที่ 2.11 ทฤษฎีบทของฟูเรียร์สไลด์แสดงความสัมพันธ์ระหว่างการแปลงฟูเรียร์ของโปรเจกชันกับการแปลงฟูเรียร์ของภาพ[14]

เราจะทำการพิสูจน์ทฤษฎีบทของฟูเรียร์สไลด์ได้โดยเริ่มต้นจากการนิยามของการแปลงฟูเรียร์ 2 มิติของภาพ $f(x,y)$ จะได้

$$F(u,v) = \int_{-\alpha}^{\alpha} \int_{-\alpha}^{\alpha} f(x,y) e^{-j2\pi(ux+vy)} dx dy \quad (2.18)$$

และให้ $S_{\theta}(w)$ คือการแปลงฟูเรียร์ของข้อมูลโปรเจกชันที่มุม θ (P_{θ})

$$S_{\theta}(w) = \int_{-\alpha}^{\alpha} P_{\theta}(t) e^{-j2\pi wt} dt \quad (2.19)$$

พิจารณาการแปลงฟูเรียร์ของวัตถุตลอดเส้นในโดเมนความถี่ โดยให้ $v = 0$ จะได้

$$F(u,0) = \int_{-\alpha}^{\alpha} \int_{-\alpha}^{\alpha} f(x,y) e^{-j2\pi ux} dx dy \quad (2.20)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่เนื่องจากตัวประกอบเฟส (Phase factor) จะไม่ขึ้นอยู่กับ y เราสามารถแบ่งการอินทิเกรตออกเป็น 2 ส่วนคือ

$$F(u,0) = \int_{-\alpha}^{\alpha} \left[\int_{-\alpha}^{\alpha} f(x,y) dy \right] e^{-j2\pi ux} dx \quad (2.21)$$

จะเห็นว่าเทอมที่อยู่ในวงเล็บเหมือนกับสมการของโปรเจกชันตลอดแนวอนที่ค่า x คงที่ซึ่งก็คือการโปรเจกชันภาพ $f(x,y)$ ที่มุม $\theta = 0$

$$P_{\theta=0}(x) = \int_{-\infty}^{\infty} f(x,y) dy \quad (2.22)$$

เอา (3.5) แทนไปใน (3.4) จะได้

$$F(u,0) = \int_{-\infty}^{\infty} P_{\theta=0}(x) e^{-j2\pi ux} dx \quad (2.23)$$

ทางด้านขวามือ แสดงถึงการแปลงฟูเรียร์ 1 มิติ ของโปรเจกชัน $P_{\theta=0}$ ดังนั้นเราจะได้ความสัมพันธ์ระหว่างโปรเจกชันในแนวตั้งกับการแปลง 2 มิติ ของ ฟังก์ชันวัตถุ คือ

$$F(u,0) = S_{\theta=0}(u) \quad (2.24)$$

จะเห็นว่าเป็นรูปแบบอย่างง่ายของทฤษฎีบทของฟูเรียร์สไลด์ ผลที่ได้นี้จะไม่ขึ้นอยู่กับการหมุนระหว่างวัตถุกับระบบพิกัด เราสามารถพิสูจน์ทฤษฎีบทของฟูเรียร์สไลด์ที่มุมอื่นได้โดยให้ $f(t,s)$ คือฟังก์ชัน $f(x,y)$ ในระบบพิกัดหมุน (Rotated Coordinate System) ดังในรูป 2.11 พิกัด (t,s) สัมพันธ์กับพิกัด (x,y) โดยความสัมพันธ์

$$\begin{bmatrix} t \\ s \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (2.25)$$

ซึ่งในระบบพิกัด (t,s) เราจะสามารถเขียนโปรเจกชันตลอดแนวเส้นของค่าคงที่ t ได้เป็น

$$P_\theta(t) = \int_{-\infty}^{\infty} f(t, s) ds \quad (2.26)$$

จากสมการที่ (2.19) การแปลงฟูเรียร์

$$S_\theta(w) = \int_{-\infty}^{\infty} P_\theta(t) e^{-j2\pi w t} dt \quad (2.19)$$

เอานิยามของโปรเจกชันไปแทนในสมการที่ (2.19) จะได้

$$S_\theta(w) = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} f(t, s) ds \right] e^{-j2\pi w t} dt \quad (2.27)$$

จากนั้นเอาผลที่ได้มาแปลงให้อยู่ในระบบพิกัด (x, y) โดยใช้ความสัมพันธ์ในสมการ (2.25) จะได้

$$S_\theta(w) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi w (x \cos \theta + y \sin \theta)} dx dy \quad (2.28)$$

ทางด้านขวาของสมการ คือการแปลงฟูเรียร์ 2 มิติ ที่ความถี่ $u = w \cos \theta, v = w \sin \theta$

$$S_\theta(w) = F(w, \theta) = F(w \cos \theta, w \sin \theta) \quad (2.29)$$

จากทั้งหมดที่ผ่านมาแสดงให้เห็นถึงโปรเจกชันของฟังก์ชันวัตถุที่มุม $\theta_1, \theta_2, \dots, \theta_k$ และการแปลงฟูเรียร์ของแต่ละโปรเจกชันนั้น เราสามารถหาค่าของ $F(u, v)$ ได้ถ้าจำนวนโปรเจกชันมีค่าเป็นอนันต์ ดังนั้นควรจะรู้ค่า $F(u, v)$ ในทุกๆจุดในระนาบ uv จากนั้นจึงทำการแปลงกลับฟูเรียร์เพื่อที่จะหาค่าฟังก์ชันวัตถุ $f(x, y)$

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{j2\pi(ux + vy)} dudv \quad (2.30)$$

ถ้าฟังก์ชัน $f(x, y)$ มีขอบเขต คือ $-\frac{A}{2} < x < \frac{A}{2}$ และ $-\frac{A}{2} < y < \frac{A}{2}$ ดังนั้นสามารถเขียนสมการที่ (2.30) ใหม่ได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f(x, y) = \frac{1}{A^2} \sum_m \sum_n F\left(\frac{m}{A}, \frac{n}{A}\right) e^{j2\pi((m/A)x + (n/A)y)}, \quad (2.31)$$

$$-\frac{A}{2} < x < \frac{A}{2}, -\frac{A}{2} < y < \frac{A}{2}$$

ในทางปฏิบัติ เราจะรู้ว่าส่วนประกอบของฟูรีเยร์มีจำนวนจำกัด สามารถเขียนใหม่ได้เป็น

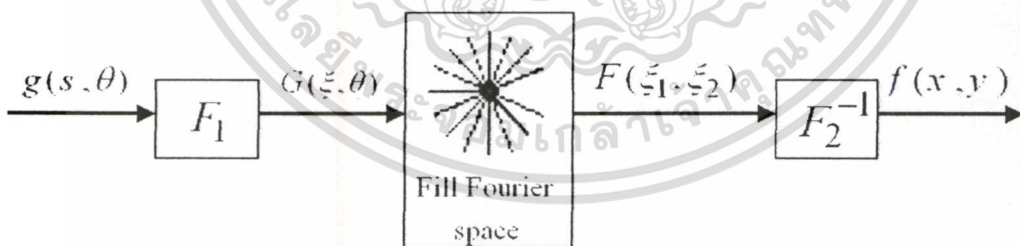
$$f(x, y) = \frac{1}{A^2} \sum_{m=-N/2}^{N/2} \sum_{n=-N/2}^{N/2} F\left(\frac{m}{A}, \frac{n}{A}\right) e^{j2\pi((m/A)x + (n/A)y)}, \quad (2.32)$$

$$-\frac{A}{2} < x < \frac{A}{2}, -\frac{A}{2} < y < \frac{A}{2}$$

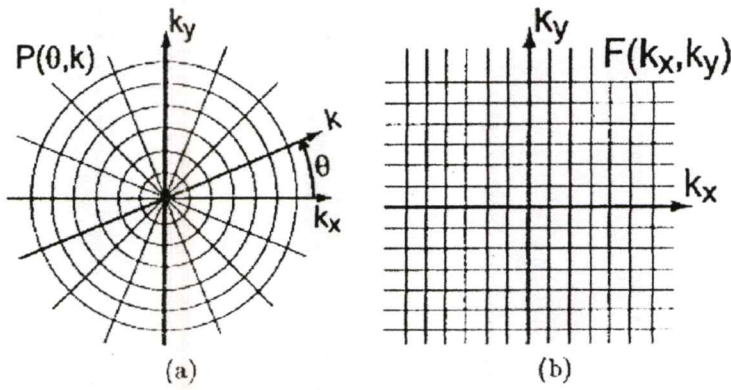
โดยสมมติว่า N เป็นจำนวนเต็มคู่ เราจะเห็นว่ารายละเอียด(Resolution)ของภาพที่สร้างได้จะถูกกำหนดโดย N สมการ (2.32) สามารถถูกคำนวณได้แบบเร็วโดยใช้อัลกอริทึมการแปลงฟูรีเยร์เร็ว (FFT) ถ้าเรารู้สัมประสิทธิ์ฟูรีเยร์จำนวน N^2 ของ $F(m/A, n/A)$

จากทฤษฎีข้างต้นนี้เราสามารถแบ่งขั้นตอนวิธีการสร้างภาพกลับด้วยวิธีการแปลงฟูรีเยร์ออกเป็น 3 ขั้นตอน คือ

- 1) นำข้อมูลโปรเจกชันไปแปลงฟูรีเยร์ 1 มิติ จะได้ $F_1\{p_\theta(r)\} = P_\theta(k)$
- 2) วาง $P_\theta(k)$ ในราสเตอร์เชิงขั้ว(polar raster) จะได้ $P(k, \theta)$ จากนั้นจึงประมาณค่าให้อยู่ในราสเตอร์มุมฉาก(rectangular raster) ดังรูปที่ 2.13
- 3) ทำการแปลงกลับฟูรีเยร์ 2 มิติของ $F(k_x, k_y)$ จะได้ $f(x, y)$

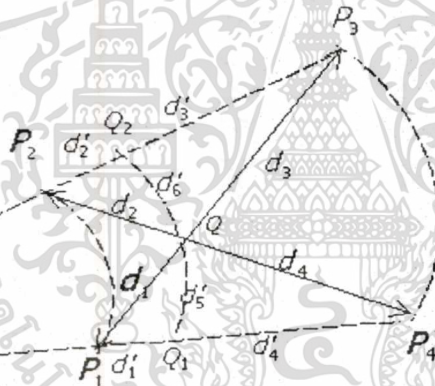


รูปที่ 2.12 Block diagram แสดงวิธีการสร้างภาพตัดขวางโดยใช้การแปลงฟูรีเยร์



รูปที่ 2.13 (a) รัสเตอร์เชิงขั้ว (b) รัสเตอร์เชิงขั้วมุมฉาก

การประมาณค่าในช่วงจากรัสเตอร์เชิงขั้วไปยังรัสเตอร์มุมฉาก สามารถทำได้หลายวิธี จากรูปที่ 2.14 จะทำการหาจุด Q จาก P_1, P_2, P_3, P_4 ด้วยวิธีดังต่อไปนี้



รูปที่ 2.14 การประมาณค่าจุด Q

1) การหาจุดที่ใกล้ที่สุด

$$F(Q) = F(P_k) \quad k : \min_i \{d_i\} = d_k \quad (2.33)$$

2) การประมาณค่าในช่วงเชิงเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$F(Q) = \frac{\sum_{k=1}^4 F(P_k) / d_k}{\sum_{k=1}^4 \frac{1}{d_k}} \quad (2.34)$$

3) การประมาณค่าในช่วงแบบไม่เชิงเส้น

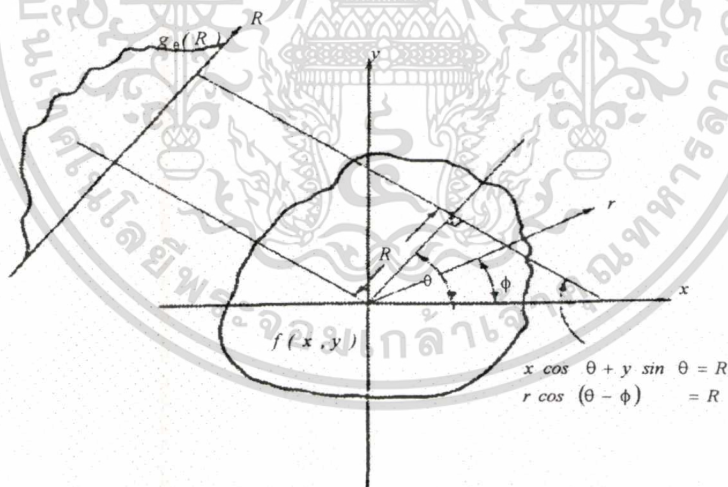
$$\begin{aligned} F(Q) &= \frac{F(Q_1)/d'_5 + F(Q_2)/d'_6}{(1/d'_5) + (1/d'_6)} \\ &= \frac{F(Q_1)/d'_6 + F(Q_2)/d'_5}{d'_5 + d'_6} \end{aligned} \quad (2.35)$$

ซึ่ง

$$F(Q_1) = \frac{F(P_1)/d'_4 + F(P_4)/d'_1}{d'_1 + d'_4}, \quad F(Q_2) = \frac{F(P_2)/d'_3 + F(P_3)/d'_2}{d'_2 + d'_3} \quad (2.36)$$

2.6.2 วิธีการสร้างภาพตัดขวางโดยใช้วิธีแบคโปรเจกชัน

การสร้างภาพตัดขวางโดยใช้วิธีแบคโปรเจกชันนั้นเป็นวิธีที่ง่ายที่สุด ความยุ่งยากทางคณิตศาสตร์มีน้อย โดยมีคูห์ล(Kuhl) และเอคเวิร์ดส์(Edwards) [11,12] เป็นสองคนแรกที่น่าวิธีนี้ไปสร้างภาพตัดขวางของผู้ป่วยได้สำเร็จ



รูปที่ 2.15 โปรเจกชันของฟังก์ชัน 2 มิติ

จากรูปที่ 2.15 และนิยามของโปรเจกชันที่กล่าวมาแล้วในบทที่ 2 จะได้นิยามของแบคโปรเจกชันของโปรเจกชันที่มุม θ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$b_{\theta}(x, y) = \int_0^{\pi} g_{\theta}(R) \delta(x \cos \theta + y \sin \theta - R) dR \quad (2.37)$$

$b_{\theta}(x, y)$ คือ ตัวกระทำแบคโปรเจกชัน (Backprojection operator) ของโปรเจกชัน $g_{\theta}(R)$ ที่มุม θ ดังนั้น การสร้างภาพกลับ (Reconstruction) จากวิธีแบคโปรเจกชัน คือ ผลรวมของแบคโปรเจกชันในทุกๆมุม จะได้

$$\begin{aligned} f_b(x, y) &= \int_0^{\pi} b_{\theta}(x, y) d\theta \\ &= \int_0^{\pi} \int_{-\alpha}^{\alpha} g_{\theta}(R) \delta(x \cos \theta + y \sin \theta - R) dR d\theta \\ &= \int_0^{\pi} d\theta \int_{-\alpha}^{\alpha} g_{\theta}(R) \delta(x \cos \theta + y \sin \theta - R) dR \end{aligned} \quad (2.38)$$

$f_b(x, y)$ คือ ภาพที่ได้จากการสร้างภาพกลับ ซึ่งการสร้างภาพกลับแบบนี้มักจะเรียกว่า ภาพถ่ายเอ็กซ์เรย์ของชั้นของเนื้อเยื่อตามความลึกที่เลือกไว้ (Laminogram) ซึ่งผลลัพธ์ที่ได้จากวิธีแบคโปรเจกชัน ($f_b(x, y)$) นี้ จะไม่ใช่วัตถุต้นฉบับที่นำมาสร้างภาพกลับ ($f(x, y)$) ซึ่งจะอธิบายได้ดังนี้

เริ่มต้น พิจารณาคุณสมบัติของเดลต้าฟังก์ชัน คือ

$$\delta(r) = \begin{cases} 1 & \text{if } r = 0 \\ 0 & \text{otherwise} \end{cases} \quad (2.39)$$

สมมติให้เดลต้าฟังก์ชันที่จุดกำเนิดคือฟังก์ชันรูปภาพต้นฉบับ $f(x, y)$ จะได้

$$f(x, y) = \delta(x) \delta(y) \rightarrow \frac{\delta(r)}{\pi r} \quad (2.40)$$

จากพิกัดเชิงมุมจาก $x \cos \theta + y \sin \theta = R$ และพิกัดเชิงขั้ว $r \cos(\theta - \phi) = R$ จะสามารถเขียนโปรเจกชัน $g_{\theta}(R)$ ได้เป็น

$$\begin{aligned}
 g_{\theta}(R) &= \int_{-\alpha}^{\alpha} \int_{-\alpha}^{\alpha} f(x, y) \delta(x \cos \theta + y \sin \theta - R) dx dy \\
 &= \int_{-\alpha}^{\alpha} \int_{-\alpha}^{\alpha} f(r, \phi) \delta(r \cos(\theta - \phi) - R) r dr d\phi
 \end{aligned}
 \tag{2.41}$$

นำสมการที่ (2.40) ไปแทนลงในสมการ (2.41) จะได้

$$g_{\theta}(R) = \int_0^{2\pi} \int_0^{\infty} \frac{\delta(r)}{\pi r} \delta(r \cos(\theta - \phi) - R) r dr d\phi
 \tag{2.42}$$

จัดรูปสมการใหม่ จะได้

$$g_{\theta}(R) = \int_0^{\pi} \int_{-\infty}^{\infty} \frac{\delta(r)}{\pi} \delta[r \cos(\theta - \phi) - R] dr d\phi
 \tag{2.43}$$

จาก $\int_0^{\alpha} dr \delta(r) f(r) = f(0)$ จะได้

$$g_{\theta}(R) = \int_0^{\pi} d\phi \frac{\delta[R]}{\pi} = \delta(R)
 \tag{2.44}$$

แสดงว่าโปรเจกชันของเคลด้าฟังก์ชันก็คือ เคลด้าฟังก์ชันเช่นกัน ถ้าเคลด้าฟังก์ชันเหล่านี้ถูกแบคโปรเจก จะได้ผลตอบสนองอิมพัลส์ คือ

$$\begin{aligned}
 h_b(r) &= \int_0^{\pi} d\theta \int_{-\infty}^{\infty} \delta(R) \delta(r \cos(\theta - \phi) - R) dR \\
 &= \int_0^{\pi} \delta(r \cos(\theta - \phi)) d\theta
 \end{aligned}
 \tag{2.45}$$

จากเคลด้าฟังก์ชันของฟังก์ชันของตัวแปร

$$\delta[f(x)] = \sum \frac{\delta(x - x_n)}{|f'(x_n)|}
 \tag{2.46}$$

ซึ่ง x_n คือ รากของ $f(x)$ จะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f(\phi) = r \cos(\theta - \phi), \quad \phi = \frac{\pi}{2} + \theta$$

$$f'(\phi) = r \sin(\theta - \phi) \Big|_{\phi = \frac{\pi}{2} + \theta} = r \quad (2.47)$$

ดังนั้น

$$h_b(r) = \int_0^{\pi} \frac{\delta\left(\phi - \left(\frac{\pi}{2} + \theta\right)\right)}{r} = \frac{1}{r} \quad (2.48)$$

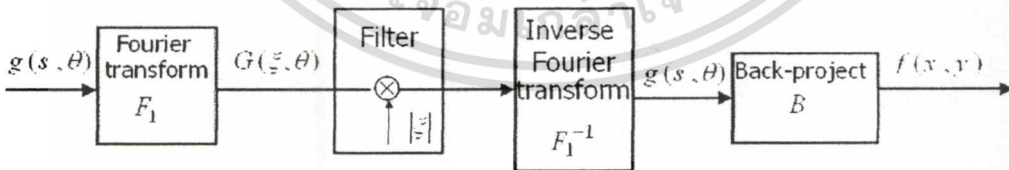
จะได้ การสร้างภาพกลับด้วยวิธีแบคโปรเจกชัน คือ

$$f_b(x, y) = f(x, y) * * \frac{1}{r} \quad (2.49)$$

** คือ คอนโวลูชัน 2 มิติ และภาพที่ได้จากวิธีนี้จะเบลอ สามารถแก้ปัญหานี้ได้ด้วยการใช้วิธีฟิลเตอร์ก่อนที่แบคโปรเจกชัน ซึ่งจะกล่าวถึงในหัวข้อต่อไป

2.6.3 วิธีการสร้างภาพตัดขวางโดยใช้วิธีฟิลเตอร์แบคโปรเจกชัน (FBP)

วิธีฟิลเตอร์แบคโปรเจกชันนี้เป็นวิธีที่เป็นที่นิยมใช้กันอย่างแพร่หลายในการสร้างภาพตัดขวาง ทั้งนี้เนื่องจากเป็นวิธีที่ได้รับการปรับปรุงจากวิธีแบคโปรเจกชันในการลดผลของ $\frac{1}{r}$ ซึ่งวิธีนี้จะใช้ความรู้เกี่ยวกับทฤษฎีบทของฟูเรียร์สไลด์ช่วยในการพิสูจน์



รูปที่ 2.16 Block diagram แสดงวิธีการสร้างภาพตัดขวางโดยใช้วิธีฟิลเตอร์แบคโปรเจกชัน

จากทฤษฎีบทของฟูเรียร์สไลด์ ภาพ $f(x, y)$ จะสามารถหามาได้จากการแปลงฟูเรียร์ย้อนกลับ

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{j2\pi(ux+vy)} dudv \quad (2.50)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าให้ (w, θ) เป็นระบบพิกัดแบบโพลาร์ (Polar Coordinate) ในระนาบ uv เราสามารถเขียนสมการ (2.50) ใหม่ได้เป็น

$$\begin{aligned} f(x, y) &= \int_0^{2\pi} \int_0^{\infty} F(w, \theta) e^{j2\pi w(x \cos \theta + y \sin \theta)} w dw d\theta \\ f(x, y) &= \int_0^{\pi} \int_0^{\infty} F(w, \theta) e^{j2\pi w(x \cos \theta + y \sin \theta)} w dw d\theta \\ &\quad + \int_0^{\pi} \int_0^{\infty} F(w, \theta + \pi) e^{j2\pi w(x \cos(\theta + \pi) + y \sin(\theta + \pi))} w dw d\theta \end{aligned} \quad (2.51)$$

จากนั้นใช้คุณสมบัติ

$$F(w, \theta + \pi) = F(-w, \theta) \quad (2.52)$$

สมการ (2.51) สามารถเขียนใหม่ได้เป็น

$$\begin{aligned} f(x, y) &= \int_0^{\pi} \left[\int_{-\infty}^{\infty} F(w, \theta) |w| e^{j2\pi w t} dw \right] d\theta \\ &= \int_0^{\pi} \left[\int_{-\infty}^{\infty} S_{\theta}(w) |w| e^{j2\pi w t} dw \right] d\theta \end{aligned} \quad (2.53)$$

โดยที่

$$t = x \cos \theta + y \sin \theta \quad (2.54)$$

และเราสามารถเขียนสมการ(2.53)ใหม่ได้เป็น

$$f(x, y) = \int_0^{\pi} Q_{\theta}(x \cos \theta + y \sin \theta) d\theta \quad (2.55)$$

โดยที่

$$Q_{\theta}(t) = \int_{-\infty}^{\infty} S_{\theta}(w) |w| e^{j2\pi w t} dw \quad (2.56)$$

จากสมการข้างบนจะเห็นว่า ในการสร้างภาพตัดขวางเราสามารถทำได้โดยการนำข้อมูลแต่ละโปรเจกชัน $P_{\theta}(t)$ มาทำการฟิลเตอร์ ที่มีผลตอบสนองทางความถี่(Frequency Response) เท่ากับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$|w|$ ดังสมการ (2.56) จากนั้นจึงนำค่าที่ได้มาทำการแบคโปรเจกชันดังสมการ (2.55) โดยที่ฟิลเตอร์ในสมการ (2.56) เรียกว่า Ram-Lak Filter

พารามิเตอร์ w เป็นตัวแปรความถี่ โดยหลักการแล้วอินทิกรัลในสมการ(2.56) ต้องทำสำหรับตลอดทุกย่านความถี่ ในทางปฏิบัติพลังงานที่อยู่ในส่วนประกอบ (Components) การแปลงฟูเรียร์เหนือความถี่ค่าหนึ่งจะมีค่าน้อย ดังนั้นการโปรเจกชันจึงถูกทำในแถบความถี่ที่จำกัด (Bandlimited) โดยให้ W แทนค่าความถี่ซึ่งถ้าเกินจากนี้พลังงานสำหรับการโปรเจกชันจะมีค่าน้อยมาก

เราสมมติว่าข้อมูลโปรเจกชันถูกสุ่มด้วยช่วงของการสุ่มเท่ากับ τ ชม. เราสมมติว่าไม่มีปัญหาเรื่อง Aliasing ซึ่งกล่าวเป็นนัยว่าในโดเมนของการแปลง (Transform Domain) โปรเจกชันไม่มีพลังงานนอกย่านความถี่ $(-W, W)$ โดยที่

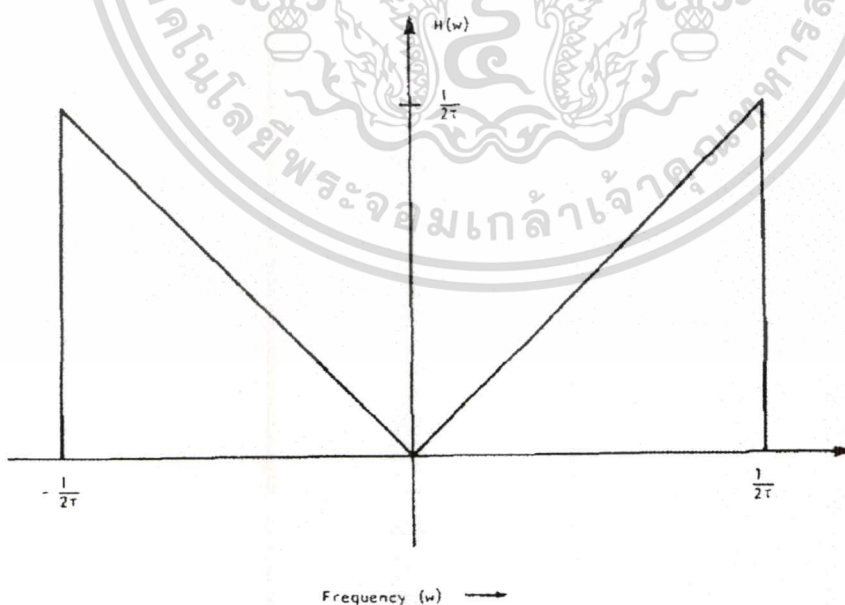
$$W = \frac{1}{2\tau} \quad \text{ไซเคิลต่อชม.} \quad (2.57)$$

เราสามารถเขียนสมการ Ram-Lak Filter ได้เป็น

$$H(w) = |w| b_w(w) \quad (2.58)$$

โดยที่

$$b_w(w) = \begin{cases} 1 & |w| \leq \frac{1}{2\tau} \\ 0, & \text{Otherwise} \end{cases} \quad (2.59)$$



รูปที่ 2.17 ภาพแสดงผลตอบสนองทางความถี่ของ Ram-Lak Filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 ทฤษฎีพื้นฐานของอัลกอริธึมแบบพีชคณิต

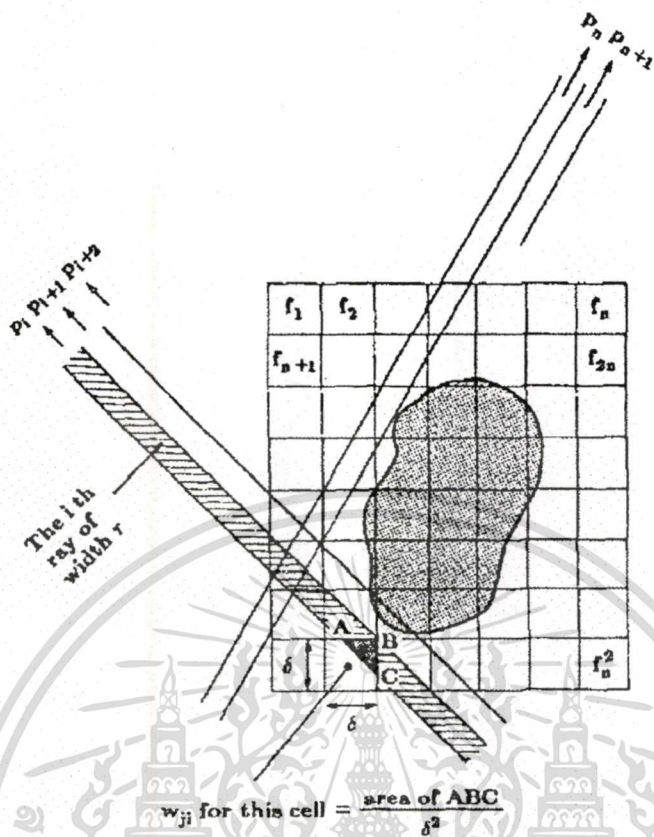
ในการสร้างภาพตัดขวางโดยใช้อัลกอริธึมแบบพีชคณิตนั้นจะสมมติให้วัตถุที่ต้องการสร้างกลับ(ภาพตัดขวางใน 2 มิติ หรือ ปริมาตรใน 3 มิติ) เป็นเมตริกของตัวแปรที่ไม่ทราบค่า และจากนั้นทำการแก้สมการทางพีชคณิตเพื่อหาค่าตัวแปรที่ไม่ทราบค่าจากค่าโปรเจกชันที่วัดได้

ในที่นี้เราจะพิจารณาการสร้างกลับของข้อมูลใน 2 มิติก่อน โดยที่เราจะสมมติให้ภาพประกอบด้วยเซตของเซลล์ $f(x,y)$ โดยที่ในแต่ละเซลล์ฟังก์ชัน $f(x,y)$ เป็นค่าคงที่ กำหนดให้ f_j เป็นค่าคงที่ที่เซลล์ที่ j^{th} และให้ N เป็นจำนวนทั้งหมดของเซลล์ ให้ p_i ค่าแทนโปรเจกชันของรังสีเส้นที่ i^{th}

$$\sum_{j=1}^N w_{ij} f_j = p_i, \quad i=1,2,\dots,M \quad (2.60)$$

โดยที่ M คือจำนวนรังสีทั้งหมด (ในทุกโปรเจกชัน) และ w_{ij} คือค่าแฟกเตอร์น้ำหนักที่แทนผลของเซลล์ที่ j^{th} ที่มีต่อรังสีเส้นที่ i^{th} โดยแฟกเตอร์ w_{ij} มีค่าเท่ากับสัดส่วนของพื้นที่ของเซลล์ที่ j^{th} ที่ตัดกับรังสี i^{th} ดังแสดงให้ดูหนึ่งเซลล์ในรูปที่ 2.18

ถ้า N และจำนวนของรังสีทั้งหมดมีค่าน้อย เราสามารถใช้ทฤษฎีเมทริกซ์ต่างๆ ไปในการแปลงกลับ(Invert)ระบบสมการใน(2.60) อย่างไรก็ตามในทางปฏิบัติ N อาจจะมีค่ามาก (ในกรณีของภาพขนาด $256 \times 256 = 65536$) และโดยมากที่จำนวนรังสีทั้งหมด(M)จะมีขนาดใหญ่พอๆกับถ้า M และ N มีค่าดังกล่าว ขนาดของเมทริกซ์ $[w_{ij}]$ ใน (4.1) คือ 65536×65536 ซึ่งไม่สามารถใช้การแปลงกลับเมทริกซ์ได้เลย



รูปที่ 2.18 รูปแบบการฉายแสงและการคำนวณค่าแพคเตอร์น้ำหนัก

สำหรับ M และ N มีค่ามาก มีเทคนิคการทำซ้ำหรือ iterative ที่น่าสนใจที่ใช้ในการแก้สมการ (2.60) เทคนิคนี้เสนอโดย Kaczmarz [2] และได้อธิบายโดยละเอียดภายหลังโดย Tanabe เพื่อที่จะอธิบายขั้นตอนการคำนวณที่ยากในเทคนิคนี้ ขั้นแรกเราจะเขียนสมการ (2.60) ใหม่ให้อยู่ในรูปแบบที่ขยายออกดังนี้

$$\begin{aligned}
 w_{11}f_1 + w_{12}f_2 + \dots + w_{1N}f_N &= p_1 \\
 w_{21}f_1 + w_{22}f_2 + \dots + w_{2N}f_N &= p_2 \\
 &\vdots \\
 w_{M1}f_1 + w_{M2}f_2 + \dots + w_{MN}f_N &= p_M
 \end{aligned}
 \tag{2.61}$$

ภาพที่ถูกแทนด้วย (f_1, f_2, \dots, f_N) จะถูกพิจารณาว่าเป็นจุดในสเปซ(Space) N มิติ ในสเปซดังกล่าวแต่ละสมการจะแทน Hyperplane เมื่อระบบสมการสามารถหาผลเฉลยที่เป็นหนึ่งเดียว (Unique Solution) จุดตัดของ Hyperplane ทั้งหมดจะเป็นจุดจุดเดียวและเป็นค่าผลเฉลยของสมการ (Solution)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในวิธีของ Kaczmarz เราต้องทำการเดาค่าเริ่มต้น กำหนดให้ค่าเริ่มต้นเป็น $f_1^{(0)}, f_2^{(0)}, \dots, f_N^{(0)}$ และสามารถเขียนแทนแบบเวกเตอร์ได้เป็น $\vec{f}^{(0)}$ ในสเปซ (Space) N มิติ ในวิทยานิพนธ์ฉบับนี้ได้กำหนดให้ค่าเริ่มต้นมีค่าเป็นศูนย์ทั้งหมด โดยค่าเริ่มต้นนี้จะถูกโปรเจกไปบน Hyperplane ซึ่งก็คือสมการแรกใน(2.61)ได้เป็น $\vec{f}^{(1)}$ จากนั้น $\vec{f}^{(1)}$ จะถูกโปรเจกไปบน Hyperplane ที่แทนด้วยสมการที่สองใน(2.61)จะได้เป็น $\vec{f}^{(2)}$ และทำต่อไปเรื่อยๆ โดยที่ $\vec{f}^{(i-1)}$ จะถูกโปรเจกไปบน Hyperplane ที่แทนด้วยสมการที่ i^{th} ใน (2.61) จะได้เป็น $\vec{f}^{(i)}$ ซึ่งขบวนการนี้เราสามารถเขียนแทนในรูปของสมการคณิตศาสตร์ได้ดังนี้

$$\vec{f}^{(i)} = \vec{f}^{(i-1)} - \frac{(\vec{w}_i \cdot \vec{f}^{(i-1)} - p_i)}{\vec{w}_i \cdot \vec{w}_i} \vec{w}_i \quad (2.62)$$

โดยที่ $\vec{w}_j = (w_{j1}, w_{j2}, \dots, w_{jN})$ และ $\vec{w}_i \cdot \vec{w}_j$ เป็นผลคูณเวกเตอร์ (Dot product) ของ \vec{w}_j กับตัวเอง

ในการประยุกต์ใช้งานจริงจำเป็นต้องมีจำนวนของโปรเจกชันจำนวนมากและภาพที่ต้องการสร้างกลับคืนมีขนาดใหญ่ทำให้เกิดความยุ่งยากที่จะใช้สมการ(2.62) ในการคำนวณเนื่องมาจากปัญหาในเรื่องการคำนวณ เนื้อที่หน่วยความจำและการที่จะเรียกสัมประสิทธิ์น้ำหนัก w_{ji} พิจารณากรณีที่เราจะสร้างภาพขนาด 128×128 จากข้อมูล 150 โปรเจกชัน โดยแต่ละโปรเจกชันมี 150 รังสี (Ray) จำนวนของสัมประสิทธิ์น้ำหนัก w_{ji} ทั้งหมดที่ใช้คือ $128 \times 128 \times 150 \times 150$ ($\approx 2.7 \times 10^8$) ซึ่งเป็นจำนวนมหาศาล

เพื่อที่จะแก้ปัญหาค่าความยุ่งยากในการคำนวณ(2.62) วิธีทางพีชคณิตอื่นๆได้ถูกนำเสนอส่วนใหญ่จะเป็นการประมาณของสมการ (2.62) เพื่อที่จะวิเคราะห์การประมาณที่สามารถคำนวณได้ง่าย เราจะจัด (2.62) ใหม่ในรูปแบบที่ต่างไปคือ

$$f_m^{(j)} = f_m^{(j-1)} + \left[(p_j - q_j) / \sum_{k=1}^N w_{jk}^2 \right] w_{jm} \quad (2.63)$$

โดยที่

$$q_j = \vec{f}^{(j-1)} \cdot \vec{w}_j = \sum_{k=1}^N f_k^{(j-1)} w_{jk} \quad (2.64)$$

สมการเหล่านี้กล่าวว่าเมื่อเราโปรเจกค่าตอบที่ $(j-1)^{\text{th}}$ ไปบน Hyperplane ที่ j^{th} (สมการที่ j^{th} ใน(2)) การเปลี่ยนแปลง $\Delta f_m^{(j)}$ ในค่าของเซลล์ที่ m^{th} คือ

$$\Delta f_m^{(j)} = f_m^{(j)} - f_m^{(j-1)} = \left[(p_j - q_j) / \sum_{k=1}^N w_{jk}^2 \right] w_{jm} \quad (2.65)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขณะที่ p_j คือผลบวกเรย์ (Ray Sum) ที่วัดได้ตามแนวรังสีที่ j^{th} q_j คือผลบวกเรย์ (Ray Sum) ที่คำนวณได้ตามแนวรังสีเดียวกันที่ใช้คำตอบ (Solution) ที่ $(j-1)^{\text{th}}$ สำหรับระดับเทาของภาพค่าที่แก้ไข (Correction) Δf_i สำหรับเซลล์ที่ i^{th} สามารถได้จากการคำนวณผลต่างระหว่างผลบวกเรย์ (Ray Sum) ที่วัดได้และผลบวกเรย์ที่คำนวณได้ ค่าความแตกต่างจะถูกนอร์มอลไลซ์ด้วย $\sum_{k=1}^N w_{jk}^2$ และถูกกำหนด (Assigned) ให้กับเซลล์ทุกเซลล์ตามแนวรังสี j^{th} ค่าที่กำหนดให้แต่ละเซลล์จะถูกสเกลด้วยค่า w_{ij} ที่สอดคล้องกัน

2.7.1 ชนิดของอัลกอริธึมแบบพีชคณิต

อัลกอริธึมแบบพีชคณิตสามารถแบ่งออกเป็น 3 ชนิดด้วยกันคือ ART (Algebraic Reconstruction Techniques), SIRT (Simultaneous Iterative Reconstructive Technique) และ SART (Simultaneous Algebraic Reconstruction Techniques)

2.7.1.1 ART (Algebraic Reconstruction Techniques)

วิธี ART นี้ได้ถูกนำเสนอสำหรับงานด้านการเอกซเรย์คอมพิวเตอร์ (CT) โดย Gordon [17] ในวิธีนี้ค่าที่แก้ไข (Correction) จะถูกใช้ปรับปรุงกับทุกๆ เซลล์ที่รังสีที่ i^{th} ผ่านก่อนที่จะค่าที่แก้ไขในรังสีถัดไป ดังนั้นเราสามารถเขียนสมการค่าโปรเจกชันของผลเฉลยได้เป็น

$$\hat{p}_i = \sum_{j=1}^N w_{ij} f_j \quad i = 1, 2, \dots, M \quad (2.66)$$

โดยที่ M เป็นจำนวนของรังสีทั้งหมด และ N เป็นจำนวนของเซลล์ขั้นตอนของการคำนวณมีดังนี้

- สำหรับอิตเทอเรชันที่ k^{th}

1. สำหรับรังสีที่ i^{th} สามารถคำนวณค่าที่แก้ไข (Correction) ได้จาก $\Delta p_i = p_i - \hat{p}_i$
2. คำนวณค่าสัมประสิทธิ์ค่าที่แก้ไขจาก $\alpha_i = \sum_{j=1}^N w_{ij}^2$
3. ทำการปรับปรุงค่าของผลเฉลยในแต่ละเซลล์ที่ j^{th} จาก

$$\hat{f}_j^{\text{new}} = \hat{f}_j^{\text{old}} + \lambda \frac{w_{ij} \Delta p_i}{\alpha_i} \quad (2.67)$$

โดยที่ค่า λ เป็นค่า Relaxation factor

4. ทำซ้ำตั้งแต่ข้อที่ 1-3 จนครบทุกๆ รังสี เป็นการเสร็จการคำนวณอิตเทอเรชันที่ k^{th}

2.7.1.2 SIRT (Simultaneous iterative Reconstruction Techniques)

Gilbert ได้เป็นผู้ที่นำเสนอวิธีนี้ โดยในวิธี SIRT ค่าในแต่ละเซลล์ของผลเฉลยจะถูกปรับปรุง หลังจากการคำนวณค่าที่แก้ไข ในแต่ละเส้นรังสีเสร็จเรียบร้อยแล้ว ซึ่งในอัลกอริทึมนี้จะมีลักษณะ คล้ายกับวิธี ART แต่จะค่าที่แก้ไขจะถูกใช้ในการปรับปรุงทีหลัง

ขั้นตอนของการคำนวณมีดังนี้

- สำหรับอิตเทอเรชันที่ k^{th}

1. สำหรับรังสีที่ i^{th} สามารถคำนวณค่าที่แก้ไข (Correction) ได้จาก $\Delta p_i = p_i - \hat{p}_i$

2. คำนวณค่าสัมประสิทธิ์ค่าที่แก้ไขจาก $\alpha_i = \sum_{j=1}^N w_{ij}^2$

3. ทำการทำซ้ำในข้อที่ 1 และ 2 จนครบทุกเส้นรังสี

- กำหนดให้ N_{c_j} เป็นจำนวนเส้นรังสีที่ผ่านเซลล์ที่ j^{th} และตรงกันกับ w_{ij} และ Δp_i
- สำหรับแต่ละเซลล์ที่ j^{th} จะทำการปรับปรุงค่าของผลเฉลยด้วยผลรวมเฉลี่ยของค่าที่แก้ไขดังสมการ

$$\hat{f}_j^{new} = \hat{f}_j^{old} + \lambda \frac{\sum_{i=1}^{N_{c_j}} w_{ij} \Delta p_i}{\sum_{i=1}^{N_{c_j}} w_{ij}} \quad (2.68)$$

เป็นการเสร็จการคำนวณอิตเทอเรชันที่ k^{th}

2.7.1.3 SART (Simultaneous Algebraic Reconstruction Techniques)

Anderson และ Kak ได้เป็นผู้ที่นำเสนอวิธีนี้ขึ้น ซึ่งเป็นการรวมกันระหว่างวิธี ART และ วิธี SIRT โดยพบว่าให้ประสิทธิภาพที่ดีกว่าทั้งสองวิธีที่กล่าวมาแล้ว ในวิธี SART นี้จะมีการปรับปรุง ค่าของผลเฉลยจากค่าที่แก้ไขค่า คล้ายกับวิธีการ ART แต่จะมีโครงสร้างคล้ายกับวิธี SIRT

ขั้นตอนของการคำนวณมีดังนี้

- สำหรับมุมของการฉายรังสีที่ θ

1. สำหรับรังสีที่ i_θ^{th} สามารถคำนวณค่าที่แก้ไข (Correction) ได้จาก $\Delta p_{i\theta} = p_{i\theta} - \hat{p}_{i\theta}$

2. คำนวณค่าสัมประสิทธิ์ค่าที่แก้ไขจาก $\alpha_{i\theta} = \sum_{j=1}^N w_{ij\theta}^2$

3. ทำการทำซ้ำในข้อที่ 1 และ 2 จนครบทุกเส้นรังสีที่อยู่ในมุมของการฉายรังสีที่ θ

- กำหนดให้ N_{c_j} เป็นจำนวนเส้นรังสีที่ผ่านเซลล์ที่ j^{th} และตรงกันกับ $w_{ij\theta}$ และ $\Delta p_{i\theta}$
- สำหรับแต่ละเซลล์ที่ j^{th} จะทำการปรับปรุงค่าของผลเฉลยด้วยผลรวมเฉลี่ยของค่าที่แก้ไขดังสมการ

$$\hat{f}_j^{new} = \hat{f}_j^{old} + \lambda \frac{\sum_{i=1}^{N_c} w_{ij0} \Delta p_{i0}}{\sum_{i=1}^{N_c} w_{ij0}} \quad (2.69)$$

- ทำซ้ำในขั้นตอนข้างต้นจนครบทุกมุมของการฉายรังสี เป็นการเสร็จการคำนวณอิทธิพลที่ k^{th}

2.7.2 ทฤษฎีของคาร์แมค (Kaczmarz Method)

ทฤษฎีของคาร์แมค(Kaczmarz Method) เป็นทฤษฎีที่ใช้ในการหาผลเฉลยของระบบสมการเชิงเส้นที่มีรูปแบบดังสมการ (2.70)

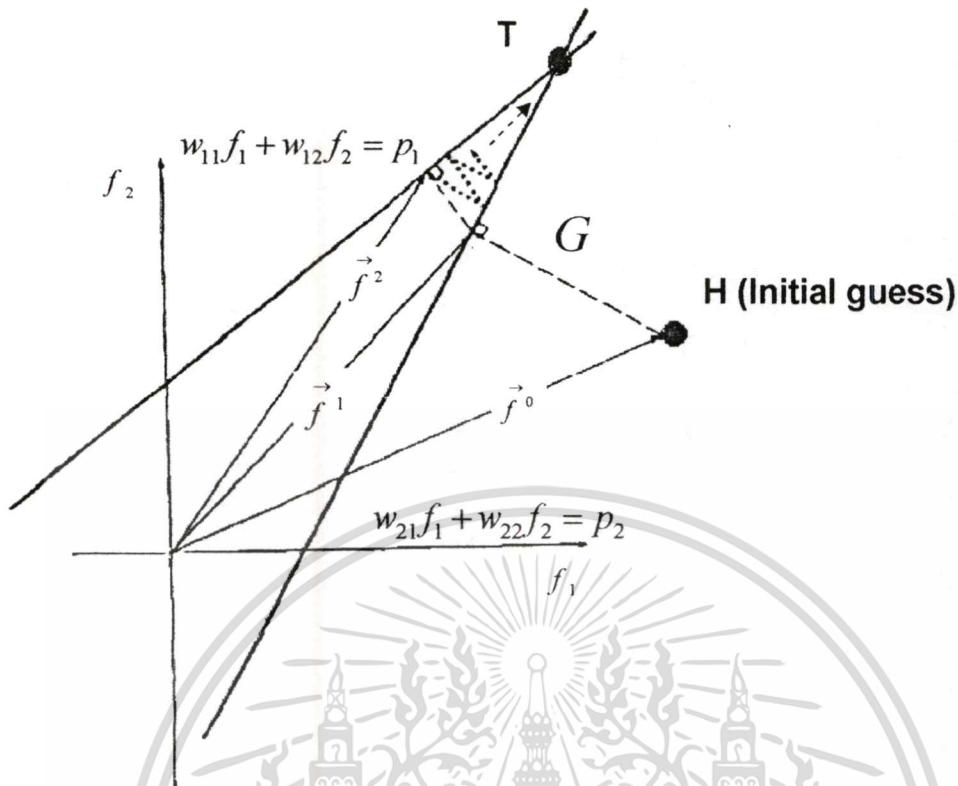
$$\begin{aligned} w_{11}f_1 + w_{12}f_2 + \dots + w_{1N}f_N &= p_1 \\ w_{21}f_1 + w_{22}f_2 + \dots + w_{2N}f_N &= p_2 \\ \vdots & \\ w_{M1}f_1 + w_{M2}f_2 + \dots + w_{MN}f_N &= p_M \end{aligned} \quad (2.70)$$

โดยผลเฉลยที่ถูกแทนด้วย (f_1, f_2, \dots, f_N) จะถูกพิจารณาว่าเป็นจุดในสเปซ(Space) N มิติ ในสเปซดังกล่าวแต่ละสมการจะแทน Hyperplane เมื่อระบบสมการสามารถหาผลเฉลยที่เป็นหนึ่งเดียว (Unique Solution) จุดตัดของ Hyperplane ทั้งหมดจะเป็นจุดจุดเดียวและเป็นค่าผลเฉลยของสมการ (Solution) เพื่อที่จะให้เห็นชัดถึงแนวคิดดังกล่าวเราได้แสดงให้เห็นในภาพ 4.2 ซึ่งเป็นกรณีที่เรามีตัวแปรแค่สองตัว f_1 และ f_2 และมีความสัมพันธ์ดังนี้

$$\begin{aligned} w_{11}f_1 + w_{12}f_2 &= p_1 \\ w_{21}f_1 + w_{22}f_2 &= p_2 \end{aligned} \quad (2.71)$$

ในวิธีของ Kaczmarz เราต้องทำการเดาคำตอบเบื้องต้น ให้การเดานี้คือ $f_1^{(0)}, f_2^{(0)}, \dots, f_N^{(0)}$ การเดาสามารถเขียนแทนแบบเวกเตอร์ได้เป็น $\vec{f}^{(0)}$ ในในสเปซ (Space) N มิติ การเดาเบื้องต้นนี้จะถูกโปรเจกไปบน Hyperplane ซึ่งก็คือสมการแรกใน (2.71) ได้เป็น $\vec{f}^{(1)}$ ดังแสดงในรูป 2.19 สำหรับกรณีสองมิติ การคำนวณในขั้นตอนแรกนี้จะทำโดยใช้สูตร

$$\vec{f}^{(1)} = \vec{f}^{(0)} - \frac{(\vec{w}_1 \cdot \vec{f}^{(0)} - p_1)}{\vec{w}_1 \cdot \vec{w}_1} \vec{w}_1 \quad (2.72)$$



รูปที่ 2.19 ภาพแสดงการใช้ทฤษฎีของคาร์แมคในการแก้ระบบสมการ 2 ตัวแปร

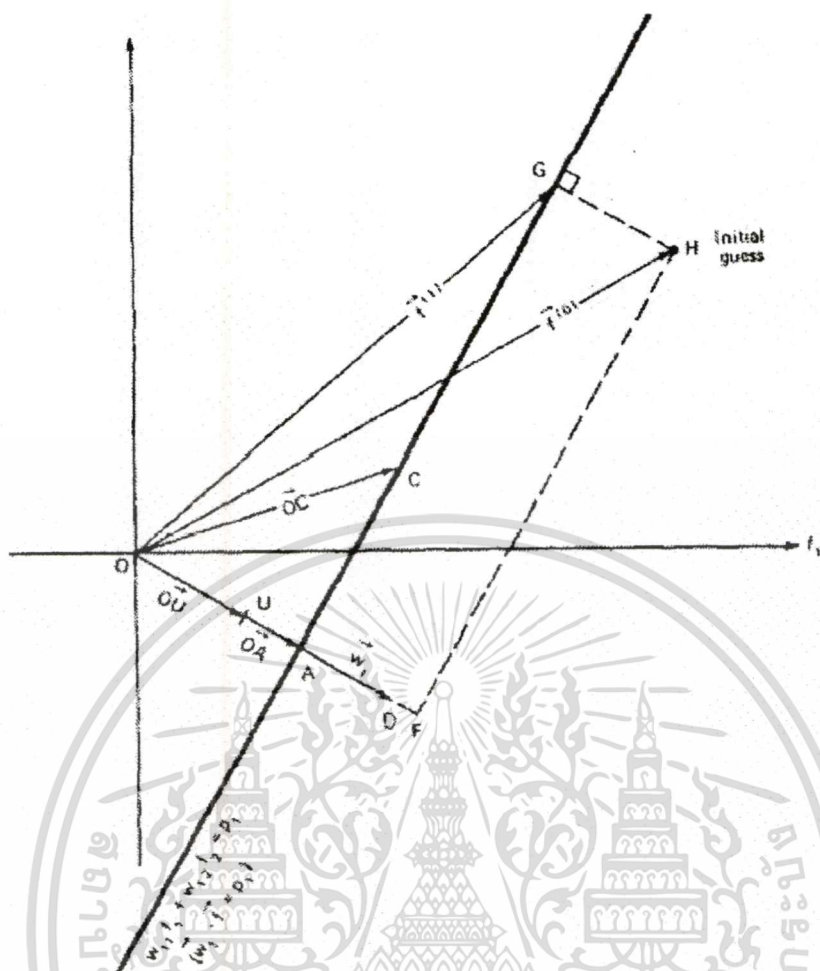
โดยที่ $\vec{w}_1 = (w_{11}, w_{12})$ คือสัมประสิทธิ์ของสมการแรกใน (2.71) การพิสูจน์ (2.72) ทำได้โดยการเขียน (2.71) ใหม่ดังนี้

$$\vec{w}_1 \cdot \vec{f} = p_1 \quad (2.73)$$

โดย Hyperplane ของสมการนี้ตั้งฉากกับเวกเตอร์ \vec{w}_1 ดังแสดงในรูป 2.20 สมการนี้กล่าวหาโปรเจกชันของเวกเตอร์ \vec{OC} (สำหรับจุดใดๆใน Hyperplane) บนเวกเตอร์ \vec{w}_1 ที่มีความยาวคงที่เป็นเวกเตอร์หนึ่งหน่วย \vec{OU} บน \vec{w}_1 คือ

$$\vec{OU} = \frac{\vec{w}_1}{\sqrt{\vec{w}_1 \cdot \vec{w}_1}} \quad (2.74)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 ภาพแสดง Hyperplane ของสมการ $\vec{w}_1 \cdot \vec{f} = p_1$

และระยะที่ตั้งฉากของ Hyperplane ถึงจุดกำเนิดเท่ากับความยาว \vec{OA} ในรูป 4.20 ซึ่งคือ $\vec{OC} \cdot \vec{OU}$

$$|\vec{OA}| = \vec{OC} \cdot \vec{OU} = \frac{1}{\sqrt{\vec{w}_1 \cdot \vec{w}_1}} \left(\sqrt{\vec{w}_1 \cdot \vec{w}_1} \vec{OU} \cdot \vec{OC} \right) \quad (2.75)$$

$$= \frac{1}{\sqrt{\vec{w}_1 \cdot \vec{w}_1}} (\vec{w}_1 \cdot \vec{f}) = \frac{p_1}{\sqrt{\vec{w}_1 \cdot \vec{w}_1}}$$

ในการหา $\vec{f}^{(1)}$ เราต้องทำการลบ $\vec{f}^{(0)}$ จาก เวกเตอร์ \vec{HG}

$$\vec{f}^{(1)} = \vec{f}^{(0)} - \vec{HG} \quad (2.76)$$

ความยาวของเวกเตอร์ \vec{HG} คือ

$$|\vec{HG}| = |\vec{OF}| - |\vec{OA}| = \vec{f}^{(0)} \cdot \vec{OU} - |\vec{OA}| \quad (2.77)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แทนค่า (2.74) และ (2.74) ลงใน (2.77) เราได้

$$|\vec{HG}| = \frac{\vec{f}^{(0)} \cdot \vec{w}_1 - p_1}{\sqrt{\vec{w}_1 \cdot \vec{w}_1}} \quad (2.78)$$

เนื่องจากทิศทางของ \vec{GH} เหมือนกับเวกเตอร์หนึ่งหน่วย \vec{OU} เราสามารถเขียน

$$\vec{HG} = |\vec{HG}| \vec{OU} = \frac{\vec{f}^{(0)} \cdot \vec{w}_1 - p_1}{\sqrt{\vec{w}_1 \cdot \vec{w}_1}} \quad (2.79)$$

แทนค่า (2.79) ลงใน (2.76) เราจะได้ (2.72)

หลังจากได้ $\vec{f}^{(1)}$ เราทำการโปรเจกต์ $\vec{f}^{(1)}$ บน Hyperplane ซึ่งคือสมการที่สองของ (2.70) ซึ่งจะได้เป็น $\vec{f}^{(2)}$ ดังแสดงในรูป 2.19 ขบวนการ Process ก็ทำซ้ำสำหรับ Hyperplane ต่อๆ ไป การโปรเจกต์บน Hyperplane ที่ j^{th} สามารถได้จากการโปรเจกต์บน Hyperplane ที่ $(j-1)^{\text{th}}$ ดังนี้

$$\vec{f}^{(j)} = \vec{f}^{(j-1)} - \frac{(\vec{w}_j \cdot \vec{f}^{(j-1)} - p_j)}{\vec{w}_j \cdot \vec{w}_j} \vec{w}_j \quad (2.80)$$

โดยที่ $\vec{w}_j = (w_{j1}, w_{j2}, \dots, w_{jN})$

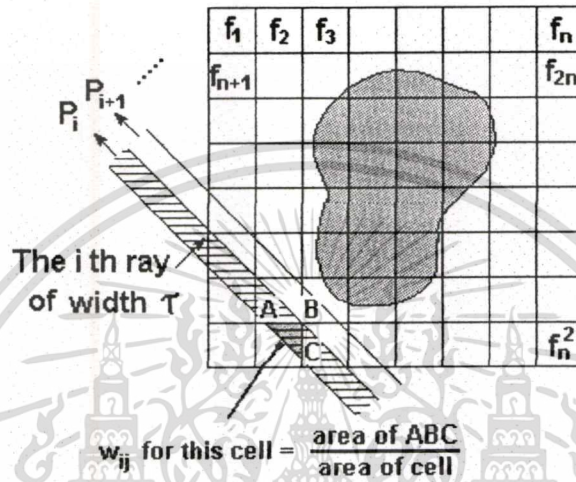
ขบวนการ (Process) ของการทำโปรเจกต์บน Hyperplane ต่างกันจะดำเนินต่อไปเรื่อยๆ จนกระทั่งเราได้ $\vec{f}^{(M)}$ ซึ่งคือการโปรเจกต์ของสมการสุดท้ายของ (2.70) จากนั้นก็จะทำการวนซ้ำโดยการโปรเจกต์ $\vec{f}^{(M)}$ ไปบน Hyperplane อันแรก ตัวอย่างของสองมิติแสดงในรูป 2.19 การโปรเจกต์ $\vec{f}^{(2)}$ บน Hyperplane อันแรกได้เป็น $\vec{f}^{(3)}$ กระบวนการ (Process) จะดำเนินต่อไปจนกระทั่งทั้งหมดของ M Hyperplane ได้มีการวนผ่าน (Cycled) เพื่อได้เป็น $\vec{f}^{(2M)}$ การทำซ้ำครั้งที่สองก็เกิดขึ้นจากการโปรเจกต์ $\vec{f}^{(2M)}$ ไปบน Hyperplane อันแรก เป็นเช่นนี้เรื่อยๆ ไป Tanabe [16] ได้แสดงว่าถ้าค่าของสมการที่เป็นหนึ่งเดียว (Unique Solution) \vec{f}_s ของระบบใน (2.70) สามารถหาได้แล้ว

$$\lim_{k \rightarrow \infty} \vec{f}^{(kM)} = \vec{f}_s \quad (2.81)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.3 การคำนวณหาค่าแฟกเตอร์น้ำหนัก(Weigthing factor)

ในการสร้างภาพตัดขวางโดยใช้อัลกอริทึมแบบพีชคณิตนั้น การคำนวณหาค่าแฟกเตอร์น้ำหนักมีความสำคัญเป็นอย่างมาก ซึ่งในการสร้างภาพตัดขวางที่มีทางเดินแสงแบบขนานกับการสร้างภาพตัดขวางที่มีทางเดินแสงรูปกรวยนั้นก็มีส่วนที่ต่างกันเฉพาะการหาค่าแฟกเตอร์น้ำหนัก ส่วนการคำนวณในช่วงของการอิทธิพลเรชันจะมีหลักการเหมือนกัน ในวิทยานิพนธ์นี้จะกล่าวถึงเฉพาะการคำนวณค่าแฟกเตอร์น้ำหนักที่ใช้กับทางเดินแสงแบบขนาน



รูปที่ 2.21 ภาพแสดงการคำนวณหาค่าแฟกเตอร์น้ำหนัก

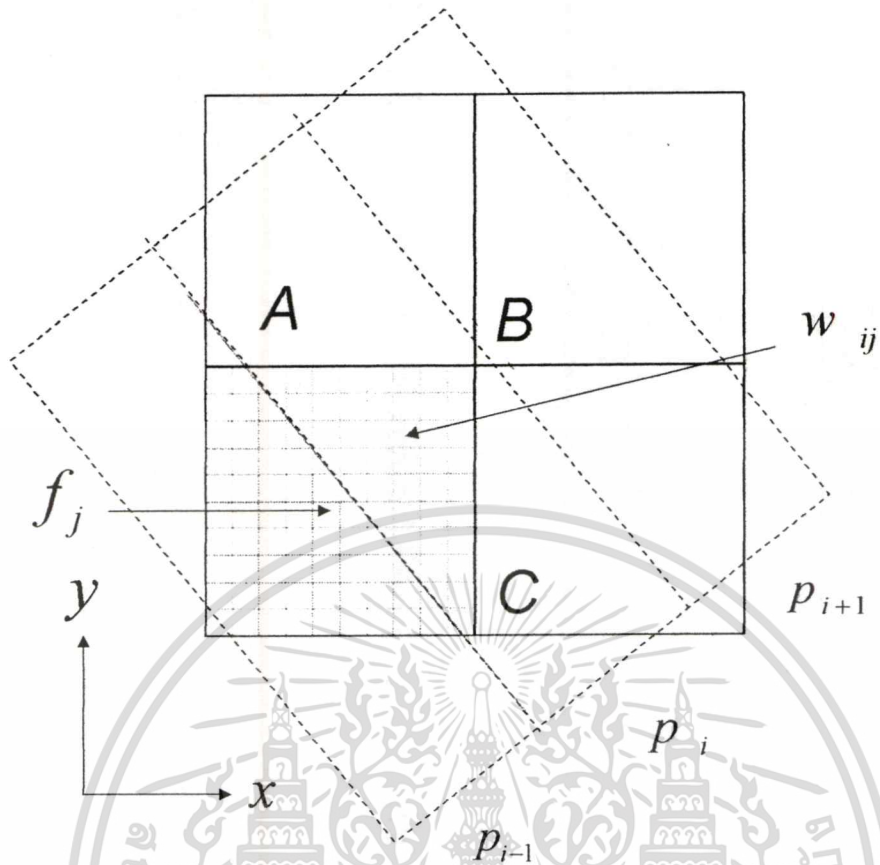
จากรูปที่ 4.21 จะเห็นว่าค่าแฟกเตอร์น้ำหนัก(w_{ij}) จะเป็นอัตราส่วนระหว่างพื้นที่ของเซลล์ที่ j ที่แสงเส้นที่ i ผ่านกับพื้นที่ของเซลล์ที่ j ดังสมการ

$$w_{ij} = \frac{\text{area of } ABC}{\text{area of cell}} \tag{2.82}$$

ซึ่งจะเห็นว่า การที่จะคำนวณหาพื้นที่ของสามเหลี่ยม ABC ของแต่ละเซลล์ที่แสงผ่านนั้น เป็นการคำนวณที่ยุ่งยากในวิทยานิพนธ์นี้จึงได้ทำการคำนวณหาค่าแฟกเตอร์น้ำหนักโดยทำการแบ่งเซลล์แต่ละเซลล์ออกเป็นเซลล์ย่อยๆ แล้วทำการคำนวณหาว่าเซลล์ย่อยแต่ละเซลล์อยู่ในเส้นทางที่แสงผ่านหรือไม่

$$w_{ij} = \frac{\text{จำนวนของเซลล์ย่อยที่อยู่ในเส้นทางเดินแสง}}{\text{จำนวนของเซลล์ย่อยทั้งหมด}} \tag{2.83}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 ภาพแสดงตัวอย่างการคำนวณหาค่าพิกเซลน้ำหนัก

จากรูปที่ 2.22 จะเห็นว่าเซลล์จะถูกแบ่งเป็น 100 เซลล์ย่อย โดยจะมี 48 เซลล์ย่อยที่อยู่ในเส้นทางเดินแสง ดังนั้นเซลล์นี้มีค่าพิกเซลน้ำหนักก็เท่ากับ $(48/100) = 0.48$

นอกจากนี้ค่าพิกเซลน้ำหนักยังสามารถที่จะคำนวณค่าไว้ล่วงหน้าแล้วเก็บเป็นไฟล์ไว้เพื่อใช้ในการสร้างภาพตัดขวางได้โดยไม่ต้องคำนวณใหม่ทุกครั้ง ซึ่งจะช่วยลดเวลาในการคำนวณลงได้

2.8 ทำไมอัลกอริธึมแบบพีซีซีดีจึงต้องการจำนวนโปรเจกชันน้อยกว่าวิธีฟลเตอร์แบคโปรเจกชัน

คำถามนี้ได้รับการตอบโดย Guan และ Gordon [18] สำหรับกรณีของการสร้างภาพกลับแบบ 2 มิติโดยใช้ลำแสงแบบขนาน และได้มีการอธิบายถึงในกรณีของการสร้างภาพกลับแบบ 3 มิติโดย Mueller ใน [19]

การสร้างภาพกลับของภาพที่เป็นสี่เหลี่ยมจัตุรัสขนาด $n \times n$ ดังนั้นจำนวนเซลล์ทั้งหมดของภาพจะเป็น $N = n^2$ โดยทั่วไปเราจะสมมติให้ขอบเขตของการสร้างภาพกลับเป็นวงกลม โดยพื้นที่นอกวงกลมถือว่าไม่มีข้อมูล ในกรณีนี้เราจะมี $(1/4)\pi n^2$ เซลล์ที่ไม่ทราบค่าและข้อมูล โปรเจกชัน n ค่าต่อโปรเจกชันหนึ่งมุม จากสมการ (2.62) ในการจะหาผลเฉลยของสมการ จำนวน โปรเจกชันของภาพ M_{ART} ต้องการเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$M_{ART} = \frac{(1/4)\pi n^2}{n} = \frac{\pi n}{4} = 0.785 \cdot n \quad (2.84)$$

จากนั้นมาดูจำนวน โปรเจกชันที่ต้องการสำหรับวิธีฟิลเตอร์แบคโปรเจกชัน (FBP) ค่าระยะห่างของการสุ่มข้อมูลในโดเมนความถี่ (Fourier space) ที่น้อยที่สุดเป็น $\Delta\omega = 1/nT_g$ และค่าความถี่สูงสุดเป็น $\omega_{\max} = 1/(2T_g)$ แต่เนื่องจากการสุ่มในระบบพิกัดเชิงขั้ว (polar sampling) ในโดเมนความถี่จึงทำให้ความหนาแน่นในการสุ่มข้อมูลลดลงเมื่อออกห่างจากแกน จากอัตราการสุ่มข้อมูลที่น้อยที่สุด $\Delta\omega$ จะเป็นตัวกำหนดขอบเขตในการคำนวณค่ามุมระหว่างโปรเจกชันที่น้อยที่สุดที่ต้องการสำหรับวิธีฟิลเตอร์แบคโปรเจกชัน ดังสมการ

$$\Delta\phi_p = \frac{\Delta\omega}{\omega_{\max}} = \frac{2T_g}{nT_g} = \frac{2}{n} \quad (2.85)$$

ดังนั้นจำนวน โปรเจกชันที่ต้องการสำหรับวิธีฟิลเตอร์แบคโปรเจกชันเป็น

$$M_{FBP} = \frac{\pi}{\Delta\phi_p} = \frac{\pi n}{2} = 1.57 \cdot n \quad (2.86)$$

จะเห็นว่าจำนวน โปรเจกชันที่ต้องการสำหรับวิธีฟิลเตอร์แบคโปรเจกชันเป็นเป็นสองเท่าของจำนวน โปรเจกชันที่ต้องการสำหรับวิธีART

ในการสร้างภาพกลับแบบสามมิติ ปริมาตรที่ต้องการสร้างกลับสี่เหลี่ยมลูกบาศก์มีขนาด $n \times n \times n$ ดังนั้นจำนวนของเซลล์ทั้งหมดของปริมาตร จะเป็น $N = n^3$ เช่นเดียวกันโดยทั่วไปแล้วจะสมมติให้ขอบเขตของการสร้างภาพกลับเป็นทรงกลม เราจะมีเซลล์ที่ไม่ทราบค่า $(1/6)\pi n^3$ เซลล์และข้อมูลโปรเจกชันจำนวน $(1/4)\pi n^2$ ต่อโปรเจกชันหนึ่งมุม ดังนั้นจำนวนโปรเจกชันที่ต้องการ M_{ART3D} เป็น

$$M_{ART3D} = \frac{(1/6)\pi n^3}{(1/4)\pi n^2} = 0.67n \quad (2.87)$$

2.9 โปรเจกชันและแบ็คโปรเจกชันใน SART

ในหัวข้อนี้กล่าวถึง ขั้นตอนการโปรเจกชันและแบ็คโปรเจกชันของกระบวนการเชิงพีชคณิต SART มีขั้นตอนการหาค่าผิดพลาดจากโปรเจกชันที่คำนวณได้กับโปรเจกชันต้นฉบับของเส้นแสงในโปรเจกชันมุมเท่ากับ θ จากนั้นนำค่าผิดพลาดดังกล่าวไปปรับปรุงให้กับภาพสำหรับมุม θ จนครบทุกมุมโปรเจกชัน แล้วจึงคำนวณค่าความผิดพลาดดังกล่าวไปปรับปรุงให้กับภาพ MSE^k สำหรับอิทเทอเรชันที่ k กระบวนการก็จะวนทำซ้ำทั้งหมดจนกว่าผลต่างของค่าผิดพลาดเฉลี่ย MSE^k กับ MSE^{k-1} จะมีค่าต่ำกว่าค่า pre-specified stopping criteria(per) ที่กำหนดไว้

$$f_j^k(\theta) = f_j^{k-1}(\theta) + \lambda \frac{p_{i\theta} - \sum_{s=0}^l f^{k-1}(s, i, \theta)}{l} \tag{2.88}$$

2.9.1 SART โปรเจกชัน

การโปรเจกชันสามารถทำได้โดยนำค่าของวัตถุหรือภาพของแสงเส้นที่ i^{th} ผ่านมารวมกัน แต่เนื่องจากข้อมูลของวัตถุหรือภาพนั้นเป็นข้อมูลแบบสุมที่อยู่ในระบบพิกัดฉาก (Cartesian-coordinate) ทำให้ตำแหน่งการสุมของเส้นแสงไม่อยู่ในตำแหน่งเดียวกัน จึงต้องใช้การประมาณค่าให้กับตำแหน่งดังกล่าว จากรูปที่ 2.23(ก) เป็นการประมาณค่าแบบเชิงเส้น (Linear-Interpolation) จากข้อมูลภาพ $f_{x,y}$ ให้กับแสงเส้นที่ i^{th} โดยนำค่าข้อมูลภาพรอบๆ $f_{x,y}$ คือ f_{x_0,y_0} , f_{x_1,y_0} , f_{x_0,y_1} และ f_{x_1,y_1} มาเฉลี่ยตามแกนอนให้ที่จุด A สมการที่ (2.89) และจุด B สมการที่ (2.90) แล้วจึงนำไปเฉลี่ยตามแกนตั้งคือ $f_{x,y}$ ตามสมการที่ (2.91)

$$A = (f_{x_1,y_0} - f_{x_0,y_0}) \cdot (x - x_0) + f_{x_0,y_0} \tag{2.89}$$

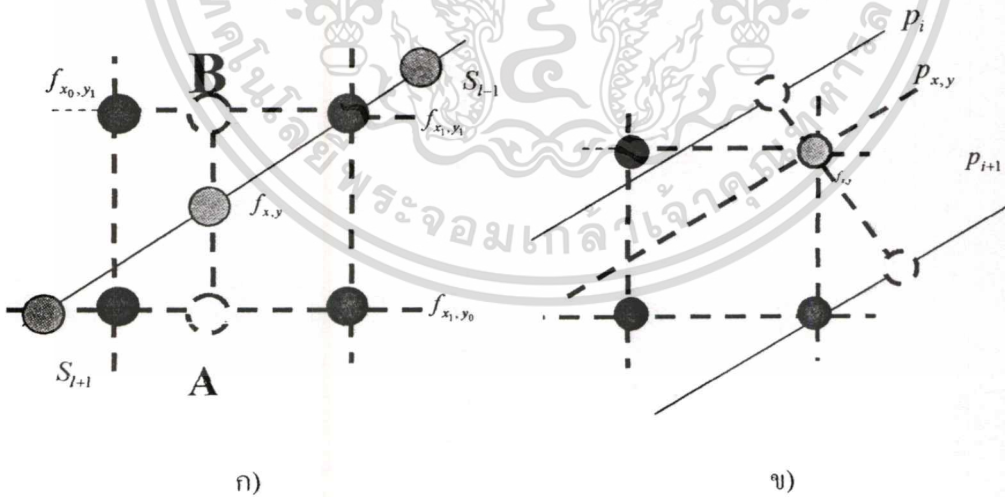
$$B = (f_{x_1,y_1} - f_{x_0,y_1}) \cdot (x - x_0) + f_{x_0,y_1} \tag{2.90}$$

$$f_{x,y} = (B - A) \cdot (y - y_0) + A \tag{2.91}$$

2.9.2 SART เบ็คโปรเจกชัน

เบ็คโปรเจกชันเป็นการนำข้อมูลจากเส้นแสง $p_{x,y}$ ไปใส่ให้ที่ภาพ $f_{x,y}$ พิจารณารูปที่ 2.23(ข) หาได้จากค่าเฉลี่ยค่าจากเส้นแสง p_i กับ p_{i+1} ดังสมการที่ 2.91

$$p_{x,y} = (p_{i+1} - p_i) \cdot (x \cos \theta + y \sin \theta - i) + p_i \tag{2.92}$$



รูปที่ 2.23 การประมาณค่าแบบเชิงเส้น

- ก) จากภาพสู่เส้นทาง
- ข) จากเส้นทางสู่ภาพ

บทที่ 3

อัลกอริทึมของการสร้างภาพตัดขวางแบบกรวย

3.1 บทนำ

อัลกอริทึมที่ใช้ในการสร้างภาพตัดขวางสำหรับทางเดินแสงแบบกรวยในวิทยานิพนธ์นี้จะกล่าวถึงเฉพาะวิธีเฟลแคม(Feldkemp Cone-beam) ซึ่งพิจารณาทางเดินแสงแบบกรวย ซึ่งจากที่ได้กล่าวมาแล้วเป็นกระบวนการสร้างภาพตัดขวางแบบ 2 มิติที่สามารถสร้างภาพตัดขวางของวัตถุได้ครั้งละระนาบ ถ้าวัตถุมีข้อมูลเป็นแบบปริมาตรก็สามารถทำได้โดยการแบ่งวัตถุออกเป็นหลายๆระนาบแล้วสร้างภาพตัดขวางทีละระนาบ แต่การสร้างภาพตัดขวางแบบขนานนี้ยังไม่สามารถนำไปใช้กับข้อมูลจริงได้ ดังนั้นจึงมีการพิจารณาทางเดินของแสงใหม่ให้ใกล้เคียงกับความเป็นจริงคือคือทางเดินแสงแบบพัดและทางเดินแสงแบบกรวยดังรูปที่ 3.1



รูปที่ 3.1 เส้นทางเดินแสง

3.2 การสร้างภาพตัดขวางแบบพัด(Fan-beam reconstruction)

กระบวนการสร้างภาพตัดขวางโดยใช้อัลกอริทึมแบบ 3 มิติ(cone-beam reconstruction) พิจารณาทางเดินแสงเป็นแบบกรวย ซึ่งมีลักษณะเหมือนคัตที่ขยขึ้น- ลมรอบจุดหมุนที่ตำแหน่งแหล่งกำเนิดแสง (point source) เพื่อให้เกิดความเข้าใจจึงต้องกล่าวถึงวิธีการสร้างภาพแบบพัด (fan-beam reconstruction) [14] เสียก่อนซึ่งถูกแบ่งออกเป็น 2 แบบคือ

3.2.1 มุมระหว่างเส้นแสงเท่ากัน(Equiangular ray)

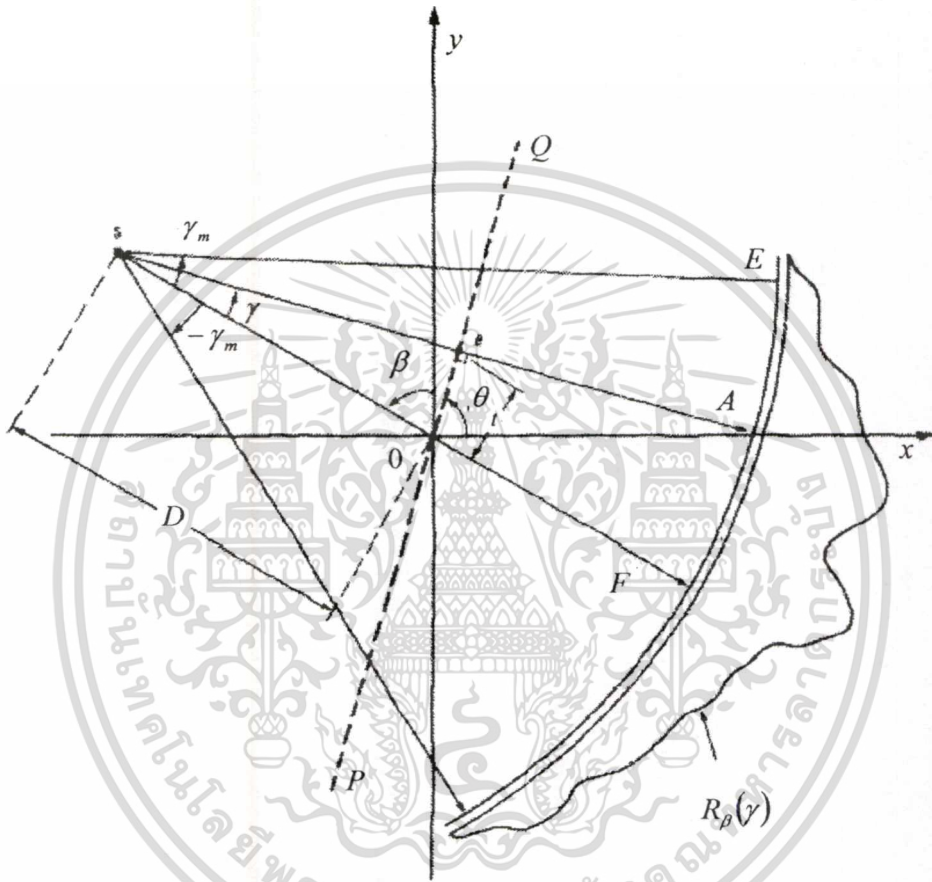
กำหนดให้ $R_\beta(s)$ เป็นข้อมูลโปรเจกชันที่มีทางเดินแสงเป็นแบบพัดดังรูปที่ 3.2 ซึ่ง β เป็นมุมที่ทำกับแกน y พิจารณาเส้นแสง \overline{SA} ถ้าโปรเจกชันถูกสร้างโดยพิจารณาทางเดินแสงเป็นแบบขนานนั่นคือ \overline{SA} เป็นเส้นแสงเส้นหนึ่งของโปรเจกชัน $p_\theta(t)$ มีมุมเท่ากับ θ ตำแหน่ง t

$$\theta = \beta + \gamma$$

$$t = D \sin \gamma \quad (3.1)$$

โดย D คือระยะห่างระหว่างแหล่งกำเนิดแสง S ถึงจุดกำเนิด O ความสัมพันธ์ในสมการที่ (3.1) ถูกนำมาใช้ในการพิจารณาให้อยู่ในรูปแบบเดียวกันกับทางเดินแสงแบบขนาน

$$f(x, y) = \int_{-t_m}^{\pi} \int_{0}^{t_m} p_{\theta}(t) \cdot h(s \cos \theta + y \sin \theta) dt d\theta \quad (3.2)$$



รูปที่ 3.2 ข้อมูลโปรเจกชันเมื่อพิจารณาทางเดินแสงแบบพัดที่มีระยะห่างระหว่างหัวรับเท่ากัน (Equally space collinear detectors)

โดยที่ t_m คือขอบเขตบนแกนดีเทคเตอร์ สมการ (3.2) นี้เป็นสมการการสร้างภาพตัดขวางวิธีฟิลเตอร์แบ็คโปรเจกชันโดยพิจารณาทางเดินแสงแบบขนาน ดังนั้นช่วงมุมโปรเจกชันจึงอยู่ระหว่าง 0 ถึง 180 องศา อย่างไรก็ตามการสร้างภาพตัดขวางจากโปรเจกชันที่เป็นแบบพัดนั้นจำเป็นต้องใช้ช่วงมุมที่มากกว่าคือระหว่าง 0 ถึง 360 องศา ดังนั้นเขียนสมการ (3.2) ได้ใหม่คือ

$$f(x, y) = \frac{1}{2} \int_{0}^{\pi} \int_{-t_m}^{t_m} p_{\theta}(t) \cdot h(s \cos \theta + y \sin \theta) dt d\theta \quad (3.3)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f(r, \phi) = \frac{1}{2} \int_{\gamma - \gamma_m}^{2\pi - \gamma} \int_{\gamma}^{\gamma_m} R_\beta(\gamma) h(r \cos(\beta + \gamma - \phi) - D \sin \gamma) D \cos \gamma d\gamma d\beta \quad (3.7)$$

พารามิเตอร์ในฟังก์ชัน h เขียนได้เป็น

$$r \cos(\beta + \gamma - \phi) - D \sin \gamma = r \cos(\beta - \phi) \cos \gamma - (r \sin(\beta - \phi) + D) \sin \gamma \quad (3.8)$$

L คือระยะห่างระหว่างแหล่งกำเนิดแสง S ถึงพิกัด (x, y) หรือ (r, ϕ) ในระบบพิกัดทรงกลมเช่น จุด C ในรูป 3.3 ซึ่ง L เป็นฟังก์ชันของ 3 ตัวแปรคือ r, ϕ, β และ γ' เป็นมุมของเส้นแสงที่ผ่านพิกัด (r, ϕ) ซึ่งเขียนความสัมพันธ์ได้คือ

$$\begin{aligned} L \cos \gamma' &= D + r \sin(\beta - \phi) \\ L \sin \gamma' &= r \cos(\beta - \phi) \end{aligned} \quad (3.9)$$

$$L(r, \phi, \beta) = \sqrt{[D + r \sin(\beta - \phi)]^2 + [r \cos(\beta - \phi)]^2} \quad (3.10)$$

และ

$$r' = \tan^{-1} \left[\frac{r \cos(\beta - \phi)}{D + r \sin(\beta - \phi)} \right] \quad (3.11)$$

นำสมการ (3.9) แทนใน (3.8) ได้

$$r \cos(\beta + \gamma - \phi) - D \sin \gamma = L \sin(\gamma' - \gamma) \quad (3.12)$$

และนำไปแทนใน (3.7) จะได้

$$f(r, \phi) = \frac{1}{2} \int_{\gamma - \gamma_m}^{2\pi - \gamma} \int_{\gamma}^{\gamma_m} R_\beta(\gamma) h(L \sin(\gamma' - \gamma)) D \cos \gamma d\gamma d\beta \quad (3.13)$$

สำหรับฟังก์ชัน h เป็นฟังก์ชันอินทรีย์ ซึ่ง $h(t)$ เป็นการแปลงกลับฟูเรียร์ของ $|\omega|$ ในโดเมนของความถี่

$$h(t) = \int_{-\infty}^{\infty} |\omega| e^{j2\pi t \omega} d\omega \quad (3.14)$$

เพราะฉะนั้น

$$h(L \sin \gamma) = \int_{-\infty}^{\infty} |\omega| e^{j2\pi L \sin \gamma \omega} d\omega \quad (3.15)$$

กำหนด ω' ให้เป็น

$$\omega' = \frac{\omega L \sin \gamma}{\gamma} \quad (3.16)$$

สามารถเขียนได้เป็น

$$\begin{aligned} h(L \sin \gamma) &= \left(\frac{\gamma}{L \sin \gamma} \right)^2 \int_{-\infty}^{\infty} |\omega| e^{j2\pi \omega' \gamma} d\omega \\ &= \left(\frac{\gamma}{L \sin \gamma} \right)^2 h(\gamma) \end{aligned} \quad (3.17)$$

สมการที่ (3.13) เขียนได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f(r, \phi) = \int_0^{2\pi} \frac{1}{L^2} \int_{-\gamma}^{\gamma} R_\beta(\gamma) g(\gamma' - \gamma) D \cos \gamma d\gamma d\beta \quad (3.18)$$

โดย

$$g(\gamma) = \frac{1}{2} \left(\frac{\gamma}{\sin \gamma} \right)^2 h(\gamma) \quad (3.19)$$

ดังนั้นสมการการสร้างภาพตัดขวางวิธีฟิลเตอร์เบ็ค โปรเจกชันทางเดินแสงแบบพัด โดยพิจารณามุมระหว่างเส้นแสงเท่ากัน เขียนสมการที่ (3.18) ได้เป็น

$$f(r, \phi) = \int_0^{2\pi} \frac{1}{L^2} Q_\beta(\gamma') d\beta \quad (3.18)$$

โดยที่

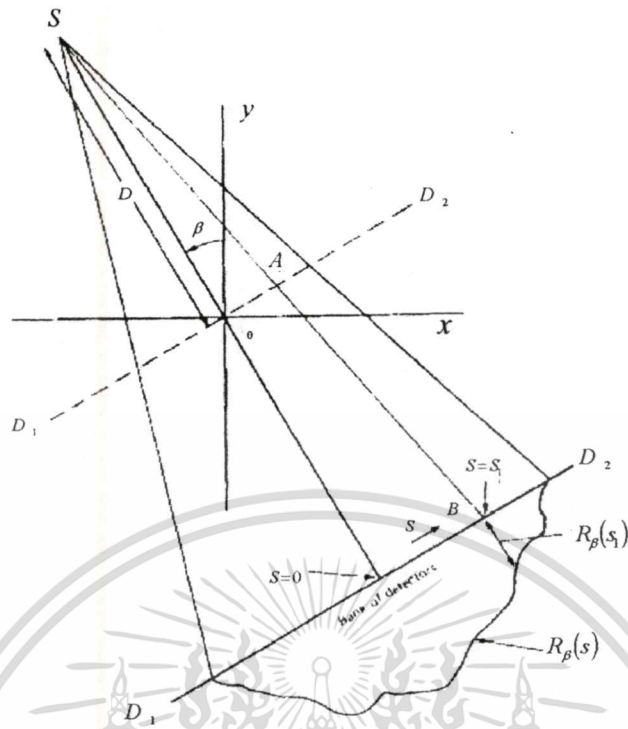
$$Q_\beta(\gamma) = R'_\beta(\gamma') * g(\gamma) \quad (3.19)$$

$$R'_\beta(\gamma') = R_\beta(\gamma) \cdot D \cdot \cos \gamma \quad (3.20)$$

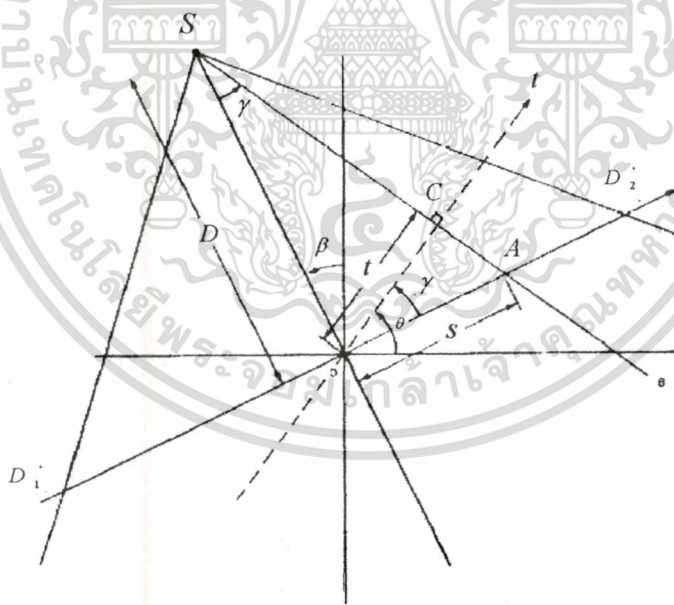
3.2.2 ระยะห่างระหว่างหัวรับเท่ากัน (Equally space collinear detectors)

กำหนดให้ $R_\beta(s)$ เป็น โปรเจกชันแบบพัด (fan-beam projection) โดยที่มุมภายในระหว่างเส้นแสงห่างเท่าๆกัน จากรูปที่ 3.4 ให้ $\overline{D_1 D_2}$ เป็นแกนคิเทคเตอร์เสมือนตัดผ่านจุดหมุนหรือจุดกำเนิด ถ้าพิจารณาเส้นแสงจากแหล่งกำเนิดแสงไปยังฉากรับ \overline{SB} ซึ่งผ่านจุด A บนแกน $\overline{D_1 D_2}$ เช่นเดียวกับจุด B บน $\overline{D_1 D_2}$ จากรูปที่ 3.3 แกนคิเทคเตอร์สมมติ $\overline{D_1 D_2}$ พิจารณาเส้นแสง \overline{SA} พบว่าค่า s มีระยะเท่ากับ \overline{OA} และถ้าสมมติการสร้างข้อมูลโปรเจกชันพิจารณาทางเดินแสงเป็นแบบขนานเส้นแสง \overline{SA} ก็คือเส้นแสงตำแหน่งที่ t ทำมุม θ หรือ $p_\theta(t)$ ดังนั้นเขียนสมการความสัมพันธ์ระหว่าง β กับ t สำหรับการพิจารณาทางเดินแสงแบบขนาน (parallel-beam projection) ดังสมการที่ (3.21)

$$\begin{aligned} t &= s \cos \gamma & \theta &= \beta + \gamma \\ t &= \frac{sD}{\sqrt{D^2 + s^2}} & \theta &= \beta + \tan^{-1} \frac{s}{D} \end{aligned} \quad (3.21)$$



รูปที่ 3.4 ข้อมูลโปรเจกชันเมื่อพิจารณาทางเดินแสงแบบพืดที่มีระยะห่างระหว่างหัวรับเท่ากัน (Equally space collinear detectors)



รูปที่ 3.5 เวกเตอร์ของทางเดินแสงแบบพืดที่พิจารณาระยะห่างระหว่างหัวรับเท่ากัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่มุม AOC เท่ากันกับมุม OSC และ D คือระยะห่างระหว่างแหล่งกำเนิดแสงที่จุด S ถึงจุดกำเนิด O จากสมการโปรเจกชันแบบขนานเขียนสมการการสร้างภาพตัดขวางแบบฟิลเตอร์เบ็คโปรเจกชันได้คือ

$$f(r, \phi) = \frac{1}{2} \int_0^{2\pi} \int_{-t_m}^{t_m} p_\theta(t) h(r \cos(\theta - \phi) - t) dt d\theta \quad (3.22)$$

เมื่อ $f(r, \phi)$ คือภาพที่สร้างได้ในระบบพิกัดเชิงขั้ว (Polar coordinates) ให้แทนความสัมพันธ์สมการ (3.21) ในสมการ (3.22) จะได้

$$f(r, \phi) = \frac{1}{2} \int_{-\tan(s_m/D)}^{2\pi - \tan(s_m/D)} \int_{-s_m}^{s_m} p_{\beta+\gamma} \left(\frac{sD}{\sqrt{D^2 + s^2}} \right) \cdot h \left(r \cos \left(\beta + \tan^{-1} \left(\frac{s}{D} \right) - \phi \right) - \frac{sD}{\sqrt{D^2 + s^2}} \right) \frac{D^3}{(D^2 + s^2)^{3/2}} ds d\beta \quad (3.23)$$

โดยที่

$$dt d\theta = \frac{D^3}{(D^2 + s^2)^{3/2}} ds d\beta \quad (3.24)$$

จากสมการที่ (3.23) s_m คือค่าที่มากที่สุดของ s ของแต่ละโปรเจกชันและเท่ากับ t_m ของโปรเจกชันแบบขนาน $-\tan(s_m/D)$ และ $2\pi - \tan^{-1}(s_m/D)$ เป็นมุมอยู่ในช่วง 0 ถึง 360 องศา เขียนความสัมพันธ์ระหว่างข้อมูลโปรเจกชันแบบตัดกับแบบขนานได้เป็น

$$R_\beta(s) = p_{\beta+\gamma} \left(\frac{sD}{\sqrt{D^2 + s^2}} \right) \quad (3.25)$$

แทน $R_\beta(s)$ โปรเจกชันแบบตัดในสมการ (3.23) เขียนสมการได้เป็น

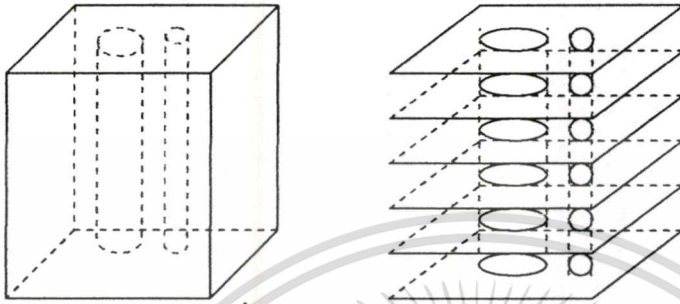
$$f(r, \phi) = \frac{1}{2} \int_{-\tan(s_m/D)}^{2\pi - \tan(s_m/D)} \int_{-s_m}^{s_m} R_\beta(s) \cdot h \left(r \cos \left(\beta + \tan^{-1} \left(\frac{s}{D} \right) - \phi \right) - \frac{sD}{\sqrt{D^2 + s^2}} \right) \frac{D^3}{(D^2 + s^2)^{3/2}} ds d\beta \quad (3.26)$$

เทอมของฟิลเตอร์ h เขียนใหม่ได้เป็น

$$\begin{aligned} r \cos \left(\beta + \tan^{-1} \left(\frac{s}{D} \right) - \phi \right) - \frac{sD}{\sqrt{D^2 + s^2}} \\ = r \cos(\beta - \phi) \frac{D}{\sqrt{D^2 + s^2}} - (D + r \sin(\beta - \phi)) \frac{s}{\sqrt{D^2 + s^2}} \end{aligned} \quad (3.27)$$

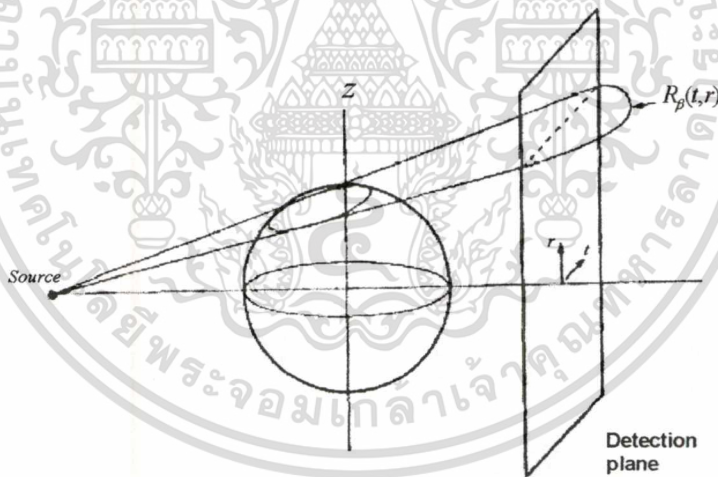
3.3 การสร้างภาพตัดขวางแบบกรวย (Cone-beam reconstruction)

การสร้างภาพจากวัตถุที่เป็น 3 มิติจากวิธีการสร้างภาพตัดขวางโดยพิจารณาทางเดินแสงแบบขนานและแบบพัดนั้นสามารถทำได้โดยการสร้างภาพตัดขวางของวัตถุแบบ 3 มิติทีละสไลด์หรือทีละ 2 มิติ แล้วค่อยนำภาพตัดขวางแต่ละสไลด์มารวมกันดังรูปที่ 3.6



รูปที่ 3.6 การสร้างภาพตัดขวางจากวัตถุ 3 มิติ

ในการสร้างภาพตัดขวางที่พิจารณาทางเดินแสงแบบกรวย ใช้การพิจารณาเช่นเดียวกับทางเดินแสงแบบพัด[14] คือแทนทางเดินแสงแบบพัดที่ทำมุมเอียงกับระนาบ (x,y) ของภาพแต่ละสไลด์มีจุดหมุนอยู่ที่ตำแหน่งแหล่งกำเนิดแสงดังรูป 3.7



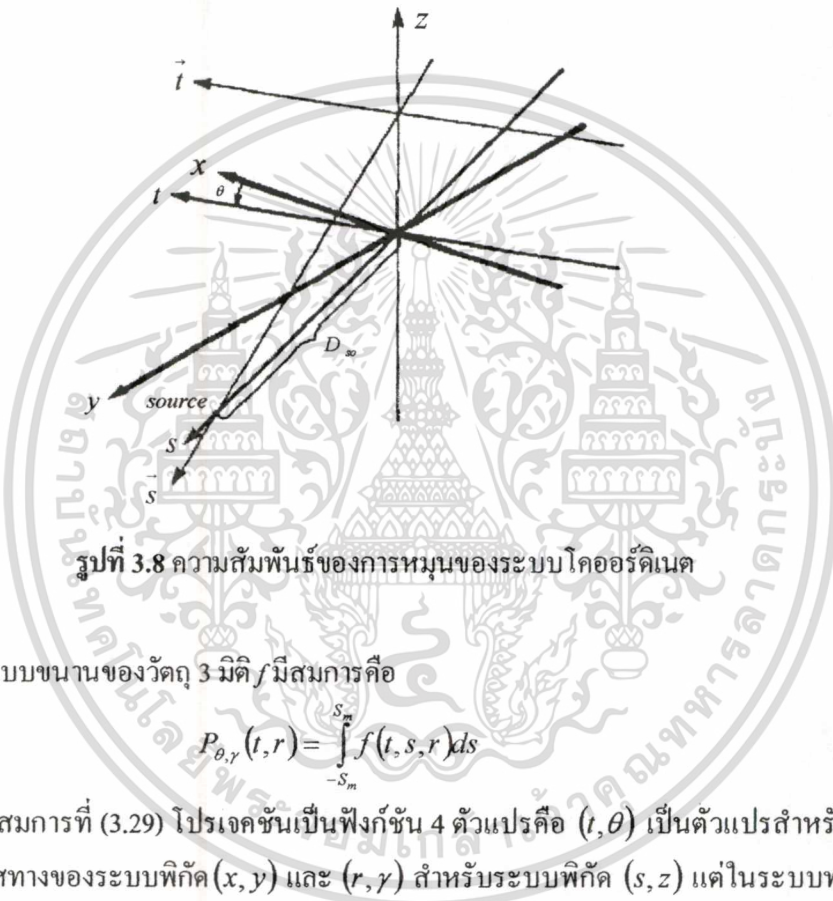
รูปที่ 3.7 การพิจารณาทางเดินแสงแบบกรวยในรูปแบบทางเดินแสงแบบพัด

ข้อดีของการฉายแสงแบบกรวยคือลดเวลาในการทำงานให้น้อยลงซึ่งมีผลให้ผู้ป่วยได้รับปริมาณรังสีเอ็กซ์น้อยลงด้วย เนื่องจากในแต่ละมุมจะฉายรังสีเอ็กซ์เพียงครั้งเดียวโดยใช้ตัวกำเนิดแสงเพียงอันเดียว จากรูปที่ 3.7 $R_p(p, \zeta)$ เป็นโปรเจกชันที่ได้จากหัววัดรังสีเอ็กซ์ดูวงวางเรียงเป็นระนาบ 2 มิติ ซึ่งโปรเจกชัน $R_p(p, \zeta)$ นี้เป็นฟังก์ชันของมุม β และตำแหน่งบนระนาบ 2 มิติ (p, ζ)

3.4 โปรเจกชันของวัตถุ 3 มิติ(Three-Dimension Projection)

โปรเจกชันของวัตถุ 3 มิติประกอบด้วย 2 ส่วนคือ โคออร์ดิเนตของทางเดินแสง(t, s, r) และ โคออร์ดิเนตของวัตถุ (x, y, z) มีความสัมพันธ์กันจากการหมุนตั้งรูปที่ 3.8 (coordinate-rotation) ที่มี 2 การหมุนคือรอบแกน z ของลำดับมุมการฉายแสงเท่ากับ θ และที่จุดกำเนิดแสงรอบแกน t เป็นมุมเท่ากับ γ ซึ่งเขียนสมการความสัมพันธ์ในฟังก์ชันของเมตริก [3,14] คือ

$$\begin{bmatrix} t \\ s \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \gamma & \sin \gamma \\ 0 & -\sin \gamma & \cos \gamma \end{bmatrix} \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (3.28)$$



รูปที่ 3.8 ความสัมพันธ์ของการหมุนของระบบโคออร์ดิเนต

โปรเจกชันแบบขนานของวัตถุ 3 มิติ f มีสมการคือ

$$P_{\theta, \gamma}(t, r) = \int_{-s_m}^{s_m} f(t, s, r) ds \quad (3.29)$$

จากสมการที่ (3.29) โปรเจกชันเป็นฟังก์ชัน 4 ตัวแปรคือ (t, θ) เป็นตัวแปรสำหรับระบบขนาดและทิศทางของระบบพิกัด (x, y) และ (r, γ) สำหรับระบบพิกัด (s, z) แต่ในระบบทางเดินแสงแบบกรวย (cone-beam system) แหล่งกำเนิดแสงจะถูกหมุนด้วยมุมเท่ากับ β และผลรวมของแสงที่วัดได้แทนด้วย $R_\beta(p', \zeta')$ โดยที่

$$p = \frac{p' D_{SO}}{D_{SO} + D_{DE}} \quad \zeta = \frac{\zeta' D_{SO}}{D_{SO} + D_{DE}} \quad (3.30)$$

เมื่อ D_{SO} คือระยะห่างระหว่างจุดหมุน (Center of rotation) ถึงแหล่งกำเนิดแสง และ D_{DE} คือระยะห่างระหว่างจุดหมุนถึงฉากรับซึ่งเขียนความสัมพันธ์ของตัวแปรในระบบทางเดินแสงทั้งสองได้คือ

$$t = p \frac{D_{so}}{\sqrt{D_{so}^2 + \zeta^2}} \quad (3.31)$$

$$\theta = \beta + \tan^{-1} \left[\frac{p}{D_{so}} \right] \quad (3.32)$$

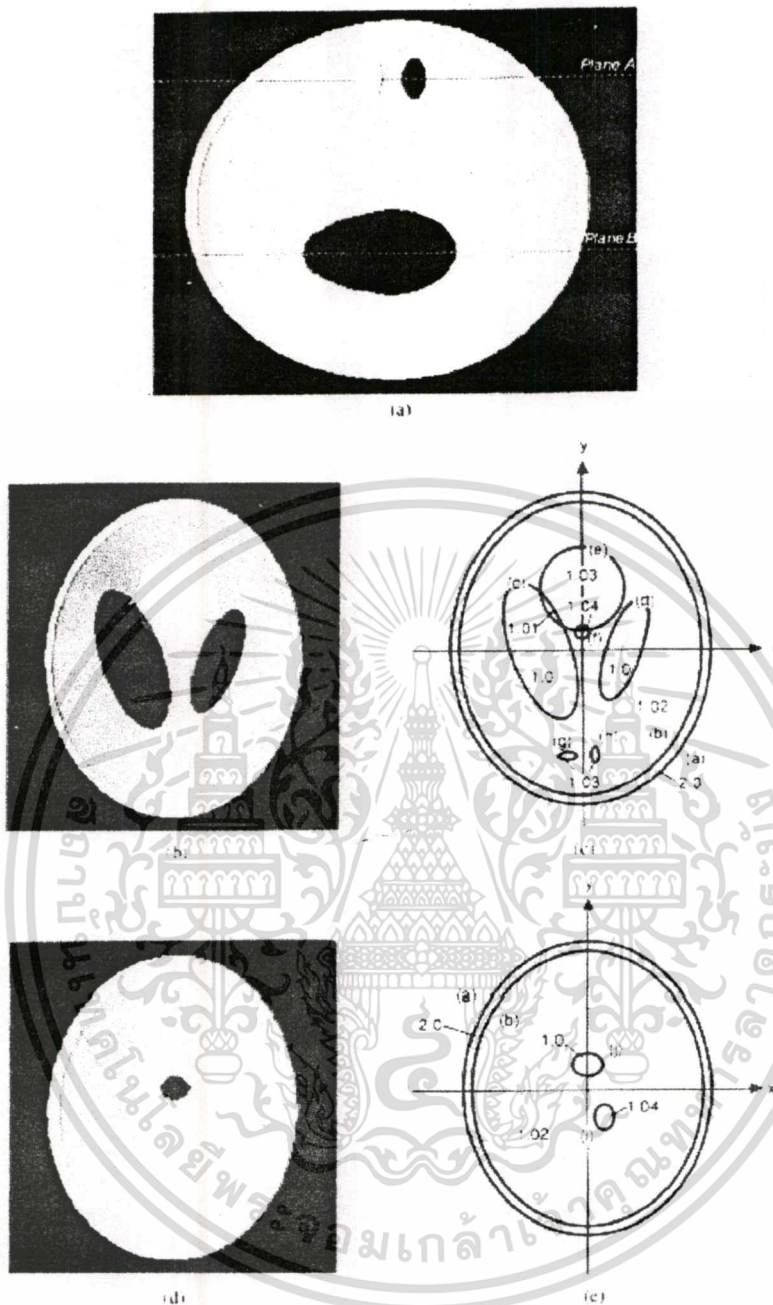
$$r = \zeta \frac{D_{so}}{\sqrt{D_{so}^2 + \zeta^2}} \quad (3.33)$$

โดยที่ r และ ζ ทำหน้าที่ระบุตำแหน่งของระนาบแสงแบบพัด (Tilted fan) ที่ผ่านวัตถุ

ในการทดสอบกระบวนการสร้างภาพตัดขวางแบบกรวยนี้จะใช้แบบจำลอง Shepp and Logan head phantom แบบ 3 มิติที่ประกอบกันจากวัตถุรูปทรงรี ตารางที่ 3.1 แสดงขนาดและตำแหน่งของรูปทรงรีต่างๆ ดังรูปที่ 3.9

ตารางที่ 3.1 พารามิเตอร์รูปทรงเรขาคณิตสำหรับสร้าง Shepp-Logan head phantom แบบ 3 มิติ

Ellipsoid	Coordinate of The Center(x, y, z)	Axis Lengths (A, B, C)	Rotation Angle β (deg)	Gray Level ρ
a	(0, 0, 0)	(0.69, 0.92, 0.9)	0	2.0
b	(0, 0, 0)	(0.6624, 0.874, 0.88)	0	-0.98
c	(-0.22, 0, -0.25)	(0.41, 0.16, 0.21)	108	-0.02
d	(0.22, 0, -0.25)	(0.31, 0.11, 0.22)	72	-0.02
e	(0, 0.1, -0.25)	(0.046, 0.046, 0.046)	0	0.02
f	(0, 0.1, -0.25)	(0.046, 0.046, 0.046)	0	0.02
g	(-0.8, -0.65, -0.25)	(0.046, 0.023, 0.023)	0	0.01
h	(0.06, -0.065, -0.25)	(0.046, 0.023, 0.023)	90	0.01
i	(0.06, -0.105, 0.625)	(0.56, 0.04, 0.1)	90	0.02
j	(0, 0.1, -0.625)	(0.056, 0.056, 0.1)	0	-0.02



รูปที่ 3.9 Shepp and Logan head phantom แบบ 3 มิติสำหรับการทดสอบกระบวนการสร้างภาพ

แบบกรวย

(a) Shepp and Logan head phantom แบบ 3 มิติ

(b) ภาพตัดขวาง A สไลด์ที่ 98

(c) รูปเรขาคณิตในภาพตัดขวาง A สไลด์ที่ 98

(d) ภาพตัดขวาง B สไลด์ที่ 208

(e) รูปเรขาคณิตในภาพตัดขวาง B สไลด์ที่ 208

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 FeldKamp หรือ ฟิลเตอร์เบ็คโปรเจกชันแบบ 3 มิติ (3D Filtered Backprojection)

การสร้างภาพกลับวิธีฟิลเตอร์เบ็คโปรเจกชันแบบ 3 มิติ [4, 14] นั้นมีวิธีการเช่นเดียวกับกับวิธีฟิลเตอร์เบ็คโปรเจกชันแบบขนาน แต่ต่างกันที่เส้นทางเดินของแสง ซึ่งสามารถพิจารณาทางเดินแสงให้เป็นรูปพัด (Fan-beam) ในระนาบ 2 มิติ แล้วนำระนาบดังกล่าวหมุนรอบแกน p โดยจุดหมุนอยู่ที่แหล่งกำเนิดแสง

เริ่มแรกให้พิจารณาสมการการสร้างภาพกลับแบบพัด (Fan-beam reconstruction) ในระบบพิกัด (r, ϕ) คือ

$$g(r, \phi) = \frac{1}{2} \int_0^{2\pi} \frac{1}{U^2} \int_{-\infty}^{\infty} R_\beta(p) h(p'-p) \frac{D_{so}}{\sqrt{D_{so}^2 + p^2}} dp d\beta \quad (3.34)$$

$$p' = \frac{D_{so} r \cos(\beta - \phi)}{D_{so} + r \sin(\beta - \phi)} \quad h(p) = \int_{-w}^w |\omega| \cdot e^{j\omega p} d\omega \quad (3.35)$$

$$U(r, \phi, \beta) = \frac{D_{so} + r \sin(\beta - \phi)}{D_{so}} \quad (3.36)$$

เขียนสมการความสัมพันธ์ (r, ϕ) ให้อยู่ในระบบโคออร์ดิเนตการหมุน (t, s) ได้คือ

$$t = x \cos \beta + y \sin \beta \quad s = y \cos \beta - x \sin \beta \quad (3.37)$$

$$x = r \cos \phi \quad y = r \sin \phi \quad (3.38)$$

$$p' = \frac{D_{so}}{D_{so} - s} \quad U(x, y, z) = \frac{D_{so} - s}{D_{so}} \quad (3.39)$$

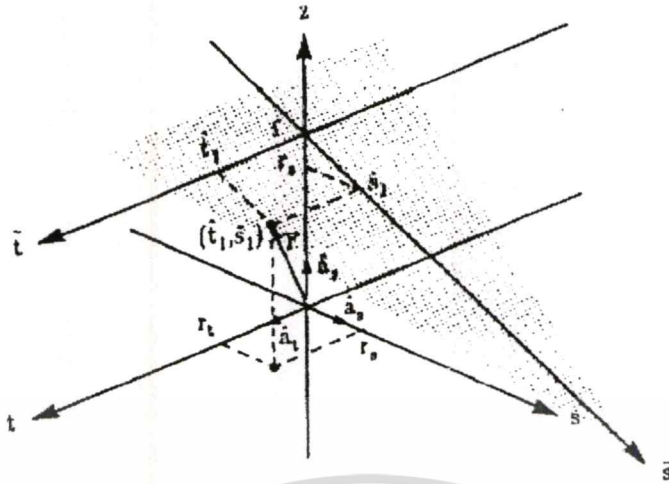
สมการสร้างภาพแบบพัดในระบบโคออร์ดิเนต (t, s) เขียนได้เป็น

$$g(t, s) = \frac{1}{2} \int_0^{2\pi} \frac{D_{so}^2}{(D_{so} - s)^2} \int_{-\infty}^{\infty} R_\beta(p, \zeta) h\left(\frac{D_{so} \cdot t}{D_{so} - s} - p\right) \frac{D_{so}}{\sqrt{D_{so}^2 + p^2}} dp d\beta \quad (3.40)$$

เมื่อแสงเปลี่ยนระนาบไปดังรูปที่ 3.10 ทำให้ไม่สามารถใช้สมการที่ (3.40) ได้เนื่องจากระยะทางของแสงเปลี่ยนไป เขียนสมการสร้างภาพแบบพัดสำหรับ (t', s') คือ

$$g(t', s') = \frac{1}{2} \int_0^{2\pi} \frac{D'_{so}{}^2}{(D'_{so} - s')^2} \int_{-\infty}^{\infty} R_{\beta'}(p, \zeta) h\left(\frac{D'_{so} \cdot t'}{D'_{so} - s'} - p\right) \frac{D'_{so}}{\sqrt{D'_{so}{}^2 + p^2}} dp d\beta' \quad (3.41)$$

โดยที่ ζ คือความสูงของแสงแบบพัดบนแกนหมุนและ $d\beta'$ หาได้จาก



รูปที่ 3.10 ภาพแสดงระบบพิกัด (t, s)

$$D_{SO} d\beta = D'_{SO} d\beta' \quad d\beta' = \frac{D_{SO} d\beta}{D'_{SO}} \tag{3.42}$$

$$t = t' \quad s' = D'_{SO} \frac{s}{D_{SO}} \quad \zeta = D_{SO} \frac{z}{D_{SO} - s} \tag{3.43}$$

$$D'_{SO} = \sqrt{D_{SO}^2 + \zeta^2} \tag{3.44}$$

แทนค่า D'_{SO} ด้วย D_{SO} และ β' ด้วย β เขียนสมการ(3.41) สำหรับทุกระนาบแสงได้เป็น

$$g(t, s) = \frac{1}{2} \int_0^{2\pi} \frac{D_{SO}^2}{(D_{SO} - s)^2} \int_{-\infty}^{\infty} R_{\beta'}(p, \zeta) h\left(\frac{D_{SO} \cdot t}{D_{SO} - s} - p\right) \frac{D_{SO}}{\sqrt{D_{SO}^2 + \zeta^2 + p^2}} dp d\beta \tag{3.45}$$

โดยขั้นตอนของฟิลเตอร์เบ็ค โปรเจกชันแบบ 3 มิติดังต่อไปนี้

1. นำโปรเจกชัน $R_{\beta'}(p, \zeta)$ คูณด้วยฟังก์ชัน $\frac{D_{SO}}{\sqrt{D_{SO}^2 + \zeta^2 + p^2}}$ เพื่อหาค่า คือ

$$R'(p, \zeta, \beta) = R(p, \zeta, \beta) \frac{D_{SO}}{\sqrt{D_{SO}^2 + \zeta^2 + p^2}} \tag{3.46}$$

2. คอนโวลูชันโปรเจกชัน $R'(p, \zeta)$ กับ $\frac{1}{2} h(p)$ ได้ค่า $Q_{\beta}(p, \zeta)$ คือ

$$Q(p, \zeta, \beta) = R'(p, \zeta, \beta) * \frac{1}{2} h(p) \tag{3.47}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. แเบ็คโพรเจกชันด้วยค่าแฟกเตอร์นำหน้าตามระยะทางของเส้นแสงสำหรับระบบโคออร์ดิเนตแบบ 3 มิติ

$$g(x, y, z) = \int_0^{2\pi} \frac{D_{so}^2}{(D_{so} - s)^2} Q_{\beta'}(p, \zeta, \beta) d\beta \quad (3.48)$$

หรือ

$$f(x, y, z) = \frac{1}{2} \int_0^{2\pi} \frac{D_{so}^2}{(D_{so} - s)^2} \int_{-\infty}^{\infty} R(p, \zeta, \beta) h\left(\frac{D_{so} \cdot t}{D_{so} - s} - p\right) \frac{D_{so}}{\sqrt{D_{so}^2 + \zeta^2 + p^2}} dp d\beta \quad (3.48)$$

3.6 ART และ SART แบบ 3 มิติ (3D Algebraic Reconstruction Algorithm)

อัลกอริธึมการสร้างภาพแบบ 3 มิติสามารถพิจารณาได้เช่นเดียวกับฟิลเตอร์แบ็คโพรเจกชันแบบ 3 มิติ คือพิจารณาทางเดินแสงแบบพัดที่ถูยกยขึ้น-ลง (Tilted fan-beam) โดยที่กระบวนการยังเหมือนเดิมกับแบบขนานดังที่กล่าวไปแล้วในบทที่ 2 นั่นคืออัลกอริธึมเชิงพีชคณิตมีขั้นตอนคือ



รูปที่ 3.11 บล็อกไดอะแกรมของอัลกอริธึมเชิงพีชคณิต

หลักการทำงานของอัลกอริธึมเชิงพีชคณิต แบบ 3 มิติจะเหมือนกับแบบ 2 มิติ แต่ต่างกันที่เราจะพิจารณาข้อมูลภาพเป็นปริมาตรหรือภาพหลายๆสไลด์แทน โดยสมการได้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$v_j^{(k)} = v_j^{(k-1)} + \frac{\left(p_i - \sum_{h=1}^N v_n^{(k-1)} w_{in} \right)}{\sum_{n=1}^N w_{in}^2} w_{ij} \quad (3.49)$$

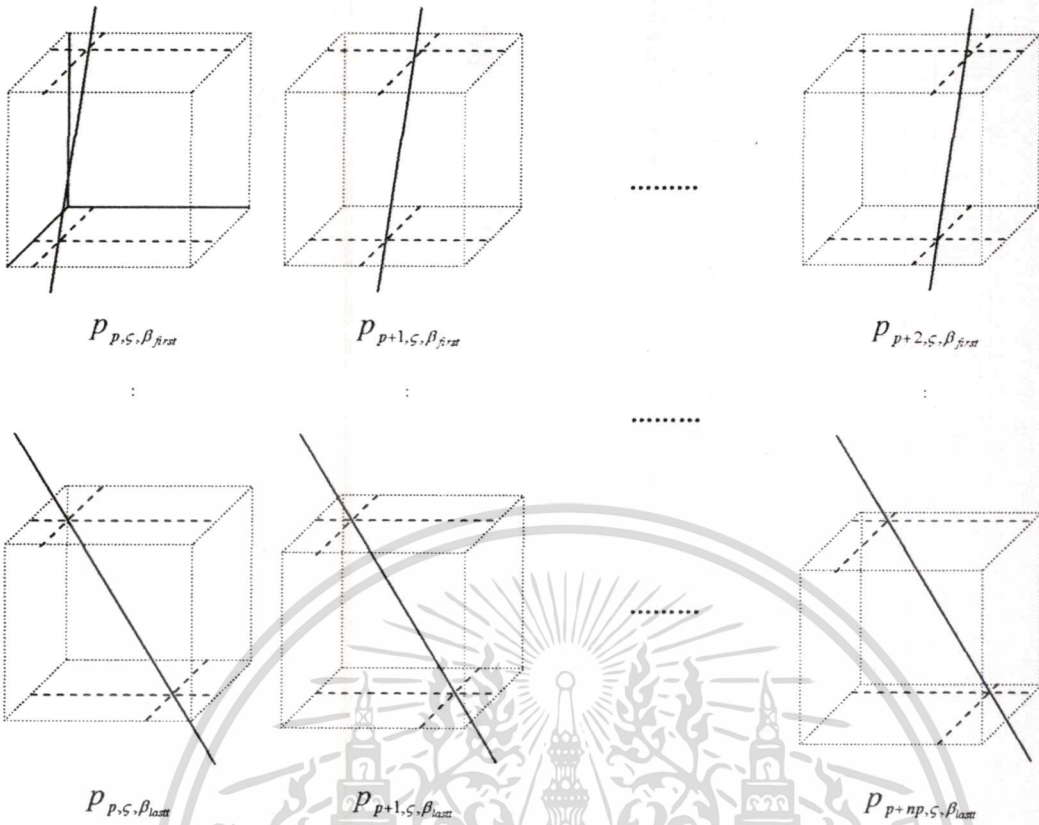
หรือเขียนให้อยู่ในระบบพิกัด (x, y, z) และ (p, ζ, β) คือ

$$v_{x,y,z}^{(k)} = v_{x,y,z}^{(k-1)} + \lambda \frac{\left(P_{p,\zeta,\beta} - \sum_{c=1}^{nz} \sum_{b=1}^{ny} \sum_{a=1}^{nx} v_{a,b,c}^{k-1} w_{p,\zeta,\beta,a,b,c} \right)}{\sum_{c=1}^{nz} \sum_{b=1}^{ny} \sum_{a=1}^{nx} w_{p,\zeta,\beta,a,b,c}^2} w_{p,\zeta,\beta,x,y,z} \quad (3.50)$$

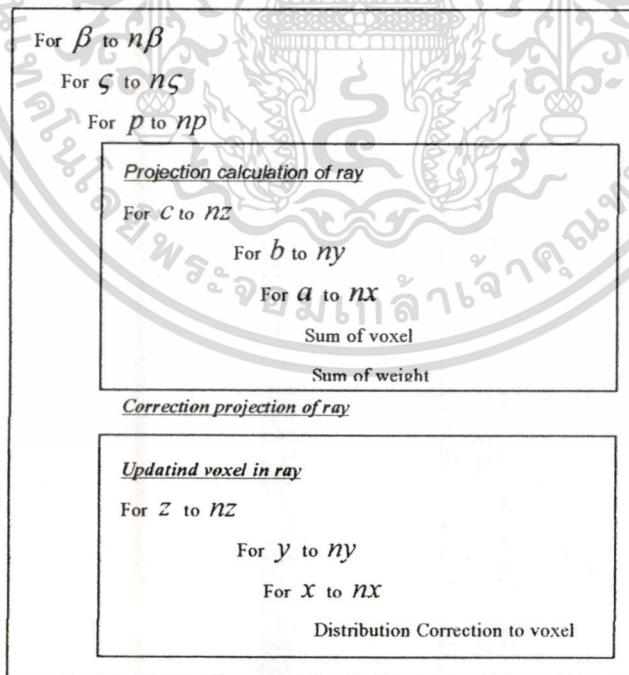
$$v_{x,y,z}^{(k)} = v_{x,y,z}^{(k-1)} + \lambda \frac{\sum_{p,\zeta,\beta \in P_\phi} \left(\frac{P_{p,\zeta,\beta} - \sum_{c=1}^{nz} \sum_{b=1}^{ny} \sum_{a=1}^{nx} v_{a,b,c}^{k-1} w_{p,\zeta,\beta,a,b,c}}{\sum_{c=1}^{nz} \sum_{b=1}^{ny} \sum_{a=1}^{nx} w_{p,\zeta,\beta,a,b,c}^2} \right)}{\sum_{p,\zeta,\beta \in P_\phi} w_{p,\zeta,\beta,x,y,z}} w_{p,\zeta,\beta,x,y,z} \quad (3.51)$$

จากสมการที่ (3.50) และ (3.51) เป็นสมการการสร้างภาพกลับด้วยวิธีเชิงพีชคณิตแบบ 3 มิติที่อธิบายด้วยระบบโคออร์ดิเนต $(p, \zeta, \beta, x, y, z)$ พบว่าฟังก์ชัน $w_{p,\zeta,\beta,x,y,z}$ มีขนาดใหญ่มาก ตัวอย่างเช่น ถ้าต้องการสร้างภาพขนาด 64×64 พิกเซล จำนวน 64 สไลด์หรือ $64 \times 64 \times 64$ วอกเซล จากข้อมูลโปรเจกชันขนาด 64×64 จำนวน 24 มุม ดังนั้นขนาดของ $w_{p,\zeta,\beta,x,y,z}$ คือ คำนวณค่าของวอกเซลต่าง ๆ $64 \times 64 \times 64 \times 64 \times 64 \times 24 = 25,769,803,776$ หรือประมาณ 25.769 G-byte เนื่องจากต้องเก็บค่าแอมพลิจูดของแต่ละเส้นแสงเป็นปริมาตรตั้งรูปที่ 3.12 และขั้นตอนการคำนวณแอมพลิจูดจะยิ่งเพิ่มขึ้นเป็นเท่าตัวเมื่อใช้การคำนวณแบบแบ่งเซลล์ย่อยดังที่กล่าวไปในบทที่ 2 หัวข้อที่ 2.7.3 เช่นถ้าแบ่งเซลล์ย่อยเป็น 100 เซลล์ย่อย ดังนั้นขั้นตอนการคำนวณจะเท่ากับ $25,769,803,776 \times 100$ ดังนั้นนำไปใช้งานจริงสำหรับข้อมูลที่มีขนาดใหญ่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

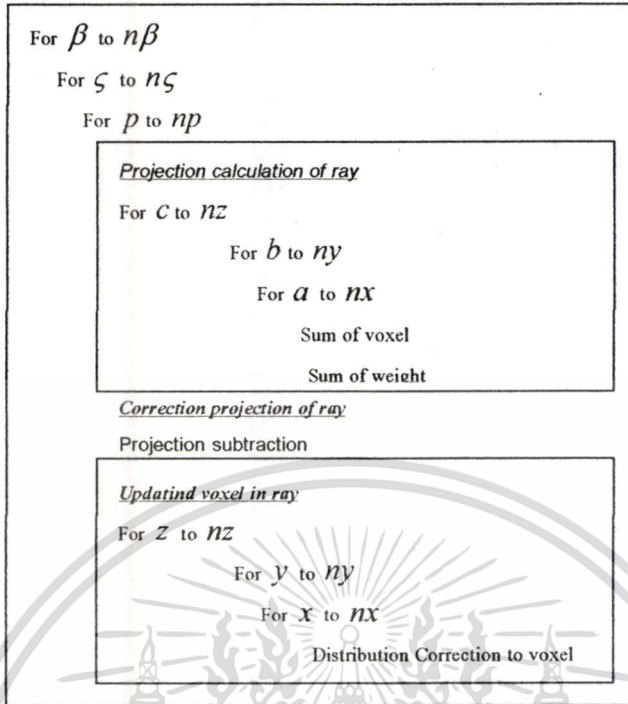


รูปที่ 3.12 ภาพแสดงเส้นทางเดินแสงแบบต่างๆ ในข้อมูลเชิงปริมาตร



รูปที่ 3.13 โปรแกรมการทำงานของ ART แบบเดิม

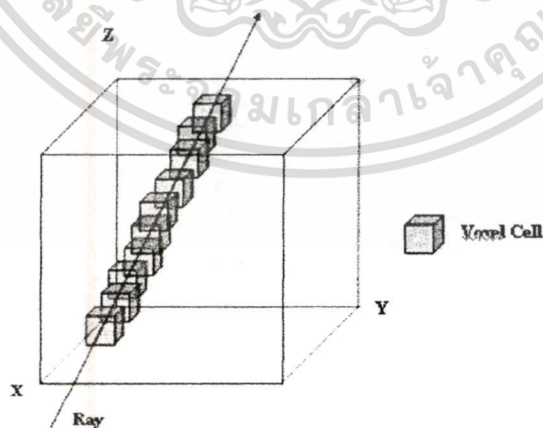
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 โปรแกรมการทำงานของ SART แบบเดิม

จากรูปที่ 3.13 และรูปที่ 3.14 แสดงลำดับการทำงานทั้งหมดของ ART และ SART แบบ 3 มิติ ซึ่งมีความยุ่งยากและซับซ้อนเป็นอย่างมาก จำนวนการทำงานทั้งหมดเขียนได้เป็น

$$\begin{aligned}
 O(ART_{org}) = & O(n_x \cdot n_y \cdot n_z \cdot n_p \cdot n_\zeta \cdot n_\beta)_{Projection} \\
 & + O(n_p \cdot n_\zeta \cdot n_\beta)_{Correction} \\
 & + O(n_x \cdot n_y \cdot n_z \cdot n_p \cdot n_\zeta \cdot n_\beta)_{Back Projection}
 \end{aligned}
 \tag{3.52}$$



$P_{p,\zeta,\beta}$

รูปที่ 3.15 ภาพแสดงข้อมูลเชิงปริมาตรบนเส้นทางเดินแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พิจารณาทางเดินแสงดังรูปที่ 3.15 จะเห็นบริเวณที่แสงผ่านในข้อมูลปริมาตรนั้นมีอยู่เพียงส่วนหนึ่งเท่านั้น ดังนั้นการเก็บข้อมูลค่าแฟกเตอร์น้ำหนักจึงไม่จำเป็นต้องเก็บทุกตำแหน่ง ซึ่งอาจจะพิจารณาในรูปของฟังก์ชันได้เพื่อลดจำนวนการทำงานและพื้นที่ข้อมูลที่ไม่จำเป็นให้น้อยลง

$$\begin{aligned} O(ART_{org}) &= O(R \cdot np \cdot n\zeta \cdot n\beta)_{Projection} \\ &+ O(np \cdot n\zeta \cdot n\beta)_{Correction} \\ &+ O(nR \cdot np \cdot n\zeta \cdot n\beta)_{Back Projection} \end{aligned} \quad (3.52)$$

โดยที่ m คือค่าดัชนีตำแหน่งของค่าแฟกเตอร์น้ำหนักชั้นตอนคำนวณค่าโปรเจกชันช่วง

$0 < m < nm$ เมื่อ nm เป็นจำนวนค่าดัชนีของเส้นแสง $p_{p,\zeta,\beta}$ และ $nm \ll nx \cdot xy \cdot xz$

r คือค่าดัชนีตำแหน่งของค่าแฟกเตอร์น้ำหนักชั้นตอนแบ็คโปรเจกชันช่วง $0 < r < nr$

เมื่อ nr เป็นจำนวนค่าดัชนีของเส้นแสง $p_{p,\zeta,\beta}$ และ $nr \ll nx \cdot ny \cdot nz$

$$v_{x(r),y(r),z(r)}^{(k)} = v_{x(r),y(r),z(r)}^{(k-1)} + \lambda \frac{\left(p_{p,\zeta,\beta} - \sum_{m=1}^{nm} v_{a(m),b(m),c(m)}^{(k-1)} W_{p,\zeta,\beta,m} \right)}{\sum_{m=1}^{nm} W_{p,\zeta,\beta,m}^2} W_{p,\zeta,\beta,r} \quad (3.53)$$

$$v_{x(r),y(r),z(r)}^{(k)} = v_{x(r),y(r),z(r)}^{(k-1)} + \lambda \frac{\sum_{p_{p,\zeta,\beta} \in P\Phi} \left(p_{p,\zeta,\beta} - \sum_{m=1}^{nm} v_{a(m),b(m),c(m)}^{(k-1)} W_{p,\zeta,\beta,m} \right)}{\sum_{p_{p,\zeta,\beta} \in P\Phi} \sum_{m=1}^{nm} W_{p,\zeta,\beta,m}} W_{p,\zeta,\beta,r} \quad (3.54)$$

บทที่ 4

การประมวลผลแบบขนานและโครงสร้างของระบบคลัสเตอร์

4.1 บทนำ

แนวความคิดของระบบคลัสเตอร์เกิดขึ้นในปี ค.ศ. 1960 โดยบริษัท IBM ต้องการเชื่อมเมนเฟรม (Mainframe) ขนาดใหญ่เข้าด้วยกันเพื่อใช้ในเชิงการค้า แต่ในตอนนั้นระบบคลัสเตอร์ยังไม่เป็นที่รู้จักอย่างแพร่หลายเนื่องจากข้อจำกัดทางด้านราคาและความแพร่หลายของฮาร์ดแวร์ (Hardware) นอกจากนี้ยังขาดเครื่องมือ (Tools) มาตรฐานสำหรับการเชื่อมต่อเครื่องในระบบเข้าด้วยกัน จนในช่วงกลางปี 1980 เป็นช่วงที่เทคโนโลยีระบบคลัสเตอร์สามารถเกิดขึ้นได้เนื่องจากสามารถสร้างไมโครโปรเซสเซอร์ที่มีประสิทธิภาพได้ และเกิดระบบเครือข่ายความเร็วสูงขึ้น ซึ่งเครื่องคอมพิวเตอร์ในระบบเครือข่ายสามารถสื่อสารกันได้ภายในเวลาไม่เกินหนึ่งมิลลิวินาที นอกจากนี้ความต้องการความเร็วในการประมวลผลงานทางวิทยาศาสตร์และวิศวกรรมศาสตร์ซึ่งมีมากขึ้นอย่างต่อเนื่องก็มีส่วนผลักดันให้เกิดเทคโนโลยีคลัสเตอร์ขึ้น

ระบบคลัสเตอร์ได้เกิดขึ้นอย่างเป็นทางการเป็นรูปธรรมในปี 1994 เมื่อ Thomas Sterling และ Don Becker จาก CESDIS (The Center of Excellence in Space Data and Information Sciences) ต้องการที่จะวิเคราะห์ข้อมูลจากอวกาศที่มีจำนวนมากและซับซ้อนซึ่งต้องใช้เครื่องคอมพิวเตอร์ที่สามารถประมวลผลได้เร็วมาก ๆ และในขณะนั้นมีเพียงซูเปอร์คอมพิวเตอร์ซึ่งมีราคาแพงแต่เนื่องจากงบประมาณที่มีอย่างจำกัดทำให้ Sterling และ Becker มีแนวความคิดที่จะใช้ระบบคลัสเตอร์มาทำการประมวลผล โดยได้สร้างระบบคลัสเตอร์จากเครื่องคอมพิวเตอร์ซึ่งใช้ชิพ Intel DX4 เป็นหน่วยประมวลผลกลาง (CPU) และส่วนประกอบอื่น ๆ ซึ่งสามารถหาได้ง่ายจากท้องตลาดและได้ตั้งชื่อระบบคลัสเตอร์นี้ว่า Beowulf (Sterling and Becker, 1994)

ระบบคลัสเตอร์มักจะเป็นระบบที่แต่ละหน่วยประมวลผลหรือ โหนดการคำนวณ (Computing node) อยู่ใกล้กัน หรืออยู่ในพื้นที่ใกล้เคียงกัน แล้วทำการเชื่อมต่อระหว่างโหนดต่อโหนดหรือแต่ละหน่วยประมวลผลผ่านทางเครือข่าย เช่น Gigabit Ethernet, Myrinet, InfiniBand, Quadrics หรือ เครือข่ายรูปแบบอื่น ๆ ซึ่งต้องเป็นเทคโนโลยีเครือข่ายที่มีความเร็วสูง และค่า Latency Time ต่ำ เพื่อให้การติดต่อสื่อสารระหว่างกันสามารถเป็นไปได้อย่างรวดเร็ว และสูญเสียเวลาในเครือข่ายน้อยที่สุดเสมือนว่าทุกหน่วยประมวลผลอยู่ใกล้กันมากจนเกือบจะเหมือนอยู่บนแผงวงจรรวม Mother Board เดียวกัน ทำให้การส่งผ่านข้อความเพื่อแลกเปลี่ยนข้อมูลต่าง ๆ ระหว่างคำนวณเป็นไปได้อย่างรวดเร็ว

4.2 การประมวลผลแบบขนาน

การประมวลผลแบบขนานเป็นวิธีการในการแบ่งปัญหาขนาดใหญ่ออกเป็นปัญหาย่อยๆ แล้วทำการแก้ปัญหาเหล่านั้น ไปพร้อมๆ กัน ในปัจจุบันระบบประมวลผลแบบขนานถูกใช้อย่างแพร่หลายทั้งในการคำนวณทางวิทยาศาสตร์และการประยุกต์ใช้กับงานทั่วไป จากความต้องการความสามารถในการประมวลผลที่มากขึ้นนี้ทำให้มีความต้องการระบบคอมพิวเตอร์ที่มีความเร็วสูงขึ้น ราคาถูกลง และมีการพัฒนาอย่างต่อเนื่อง และในการพัฒนาระบบคอมพิวเตอร์แบบขนานถูกแบ่งออกเป็นสองกลุ่มคือ

1. ระบบคอมพิวเตอร์ขนานแบบหลายหน่วยประมวลผล (Massively Parallel Processors (MPPs)) ซึ่งเป็นระบบที่มีความสามารถสูงที่สุดในปัจจุบัน โดยระบบคอมพิวเตอร์นี้อาจประกอบด้วยหน่วยประมวลผลกลางตั้งแต่หนึ่งร้อยถึงหนึ่งพันหน่วยซึ่งหน่วยประมวลผลกลางเชื่อมต่อกันผ่านระบบบัส (BUS) ปัจจุบันระบบ MPP ถูกใช้ในการแก้ปัญหาที่ต้องการความเร็วในการประมวลผลมากๆ หลายอย่าง เช่น การสร้างแบบจำลองสภาพภูมิอากาศของโลก และ การออกแบบยาเป็นต้น

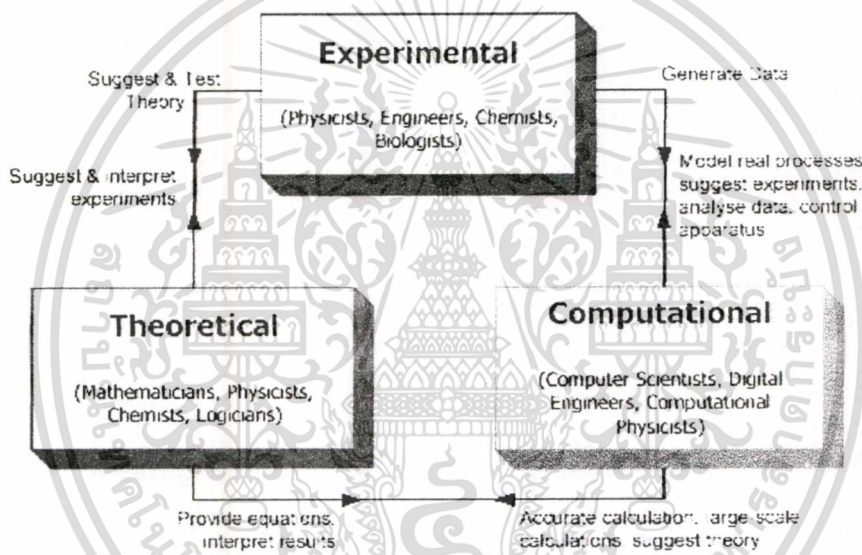
2. ระบบคอมพิวเตอร์แบบกระจาย (Distributed Computing) ซึ่งเป็นกระบวนการที่เครื่องคอมพิวเตอร์หลายๆ เครื่องเชื่อมต่อกันผ่านระบบเครือข่ายเพื่อแก้ปัญหาขนาดใหญ่โดยประสิทธิภาพของระบบคอมพิวเตอร์แบบนี้อาจจะเทียบเท่ากับหรือมากกว่าระบบซูเปอร์คอมพิวเตอร์ขนาดใหญ่ได้ ระบบระบบคอมพิวเตอร์ขนานแบบหลายหน่วยประมวลผลที่ได้กล่าวถึงในตอนแรกสามารถนำมาเชื่อมต่อกันผ่านทางระบบเครือข่ายเพื่อใช้ในลักษณะของระบบระบบคอมพิวเตอร์แบบกระจายเพื่อทำให้มีประสิทธิภาพในการคำนวณมากขึ้น

4.2.1 การประยุกต์ใช้การประมวลผลแบบขนาน

ในอดีตการแก้ปัญหาทางวิทยาศาสตร์มีสาขาวิชาที่สัมพันธ์กันสองสาขาคือวิทยาศาสตร์เชิงทฤษฎี (Theoretical Science) และ วิทยาศาสตร์เชิงการทดลอง (Experimental Science) นักวิทยาศาสตร์เชิงทฤษฎีจะใช้ทฤษฎีทางวิทยาศาสตร์สร้างแบบจำลองทางคณิตศาสตร์และสมมติฐานของปัญหาจากนั้น นักวิทยาศาสตร์เชิงการทดลองนำแบบจำลองและสมมติฐานของปัญหาไปสร้างการทดลองเพื่อพิสูจน์แบบจำลองและสมมติฐาน ถ้าหากการทดลองเป็นไปตามสมมติฐานและแบบจำลองถูกต้องแสดงว่าสามารถแก้ปัญหานั้นๆ ได้ แต่ถ้าหากการทดลองล้มเหลว นักวิทยาศาสตร์เชิงทฤษฎีจะนำแบบจำลองและสมมติฐานกลับไปแก้ไขและนำกลับมาทดลองอีกครั้ง ปัญหาในปัจจุบันมีความซับซ้อนมาก งานหลายอย่างไม่สามารถเสี่ยงต่อการล้มเหลวได้ เนื่องจากข้อจำกัดทางด้านงบประมาณ เวลา และความปลอดภัย เมื่อระบบคอมพิวเตอร์ถูกพัฒนาขึ้นมาจึงเกิดสาขาวิชาวิทยาศาสตร์เชิงคำนวณ (Computational Science) ขึ้นมาเพื่อช่วยในการแก้ปัญหาทางวิทยาศาสตร์ที่ซับซ้อนให้เป็นไปได้ไปอย่างมีประสิทธิภาพ ความสัมพันธ์ของวิทยาศาสตร์ทั้งสาม

สาขานี้แสดงดังรูปที่ 4.1 เมื่อนักวิทยาศาสตร์เชิงทฤษฎีพัฒนาแบบจำลองทางคณิตศาสตร์ของปัญหาขึ้นมาจะส่งแบบจำลองนั้นให้แก่นักวิทยาศาสตร์เชิงคำนวณเพื่อสร้างโปรแกรมเพื่อแก้ปัญหานั้นแบบเชิงตัวเลข (Numerically) ส่วนนักวิทยาศาสตร์เชิงการทดลองจะให้คำแนะนำเกี่ยวกับคุณสมบัติในความเป็นจริงต่างๆที่ใช้ในการแก้ปัญหานั้นแก่นักวิทยาศาสตร์เชิงคำนวณเพื่อใช้เป็นข้อมูลในการแก้ปัญหานั้น ข้อดีของการใช้คอมพิวเตอร์แก้ปัญหานั้นทางวิทยาศาสตร์มีดังนี้

1. ใช้งบประมาณน้อยกว่า และทำได้รวดเร็วกว่าการทดลองในห้องปฏิบัติการ
2. คอมพิวเตอร์สามารถแก้ปัญหานั้นได้หลายเงื่อนไขกว่าการใช้เฉพาะเครื่องมือที่มีอยู่ในห้องปฏิบัติการ
3. ข้อจำกัดของการแก้ปัญหานั้นด้วยการคำนวณมีเพียงแต่ความเร็วและจำนวนหน่วยความจำของคอมพิวเตอร์ แต่การทดลองในห้องปฏิบัติการจะมีข้อจำกัดในทางปฏิบัติมากกว่า



รูปที่ 4.1 ความสัมพันธ์ระหว่างวิทยาศาสตร์เชิงทฤษฎี การทดลอง และเชิงคำนวณ

การประยุกต์ใช้ระบบการประมวลผลแบบขนานสามารถแบ่งออกได้เป็นห้ากลุ่มใหญ่ๆคือ

4.2.1.1 แบบจำลองของการทำนายและการจำลองแบบ (Predictive Modelling and Simulations)

แบบจำลองหลายมิติ (Multidimensional Modeling) ของอากาศ สิ่งแวดล้อมของโลก อวกาศและเศรษฐกิจโลก การสร้างแบบจำลองเพื่อทำนายเหตุการณ์เหล่านี้ถูกทำโดยการจำลองด้วยคอมพิวเตอร์ซึ่งการคำนวณต้องการระบบที่มีความเร็วในการประมวลผลอย่างต่ำ 1 GFLOPS (Giga-floating point operation per second) หรือหนึ่งพันล้านการปฏิบัติงานแบบทศนิยมต่อหนึ่งวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.2 การออกแบบทางวิศวกรรมและระบบอัตโนมัติ (Engineering Design and Automation)

ระบบคอมพิวเตอร์แบบขนานถูกใช้ในการวิเคราะห์และออกแบบทางวิศวกรรมเช่น การวิเคราะห์ไฟไนต์อีลีเมนต์ (Finite-Element Analysis) พลศาสตร์เชิงคำนวณ (Computational Aerodynamic) เป็นต้น

4.2.1.3 การค้นหาแหล่งพลังงาน (Energy Resources Exploration)

คอมพิวเตอร์มีบทบาทสำคัญในการค้นหาน้ำมันและก๊าซธรรมชาติทั้งในด้านการขุดหา และการจัดการควบคุมระบบ

4.2.1.4 ด้านการแพทย์ การทหาร และการวิจัยพื้นฐาน (Medical, Military and Basic Research)

ในด้านการแพทย์ คอมพิวเตอร์ความเร็วสูงจะถูกใช้ในการสร้างภาพทางการแพทย์ การออกแบบหัวใจเทียม การวินิจฉัยโรคตับ และการศึกษาทางด้านพันธุวิศวกรรม สำหรับด้านการทหารคอมพิวเตอร์ความเร็วสูงจะถูกใช้ในการออกแบบอาวุธ และการทำสงครามด้วยอุปกรณ์อิเล็กทรอนิกส์

4.2.1.5 การสร้างภาพ (Visualization)

เทคนิคซึ่งใช้ในอุตสาหกรรมหนังเช่น The Matrix และ Toy Story ซึ่งการคำนวณสำหรับสร้างภาพต้องใช้คอมพิวเตอร์ความเร็วสูงสำหรับสร้างภาพเสมือนจริงจำนวนมาก ในหนังทั้งสองเรื่องใช้ระบบคลัสเตอร์ในการประมวลผล นอกจากนั้นการสร้างภาพเสมือนจริงจากข้อมูลทางวิทยาศาสตร์ก็เป็นอีกงานหนึ่งซึ่งต้องใช้คอมพิวเตอร์ความเร็วสูงในการประมวลผลเพื่อสร้างภาพ

4.3 การจำแนกประเภทของคอมพิวเตอร์

โดยทั่วไปแล้วสามารถจำแนกชนิดของระบบคลัสเตอร์ออกตามกฎเกณฑ์ต่างๆดังนี้[10]

4.3.1 จำแนกตามงานที่ประยุกต์ใช้ (Application Target)

ระบบคลัสเตอร์สามารถนำเอาไปประยุกต์ใช้งานได้หลายอย่าง ไม่ว่าจะเป็นงานทางด้านการคำนวณหรืองานทางด้านการเป็นเครื่องแม่ข่ายให้บริการงานต่างๆ โดยสามารถแบ่งย่อยได้เป็นสองประเภทคือ

- ระบบคลัสเตอร์แบบประสิทธิภาพสูง (High Performance (HP) Clusters) ระบบนี้มักจะนำมาประยุกต์ใช้ในการคำนวณทางด้านวิทยาศาสตร์และคณิตศาสตร์ ระบบคลัสเตอร์แบบนี้จะถูกสร้างขึ้นมาให้มีความรวดเร็วในการคำนวณมากที่สุด ประสิทธิภาพของหน่วยประมวลผล

จะต้องสูงเพียงพอ อีกทั้งเครือข่ายที่ใช้ในการเชื่อมต่อต้องมีคุณภาพดีมาก ประสิทธิภาพในการคำนวณจึงจะสูงตามไปด้วย

- ระบบคลัสเตอร์แบบเสถียรภาพสูง (High Availability (HA) Clusters) ระบบคลัสเตอร์แบบนี้จะเน้นไปทางด้านเครื่องแม่ข่ายที่ให้บริการงานต่าง ๆ เช่น ให้บริการเป็นเว็บเซิร์ฟเวอร์ (Web Server) หรือให้บริการพื้นที่เก็บข้อมูลบนเครือข่าย (Storage Server) เพื่อจะทำให้มั่นใจได้ว่าผู้ใช้งานทั่ว ๆ ไป จะสามารถเข้าถึงทรัพยากรต่าง ๆ ได้ตลอดเวลา การให้บริการไม่ได้ขึ้นอยู่กับเครื่องเซิร์ฟเวอร์เครื่องใดเครื่องหนึ่ง

4.3.2 จำแนกตามลักษณะของเครื่องในระบบ

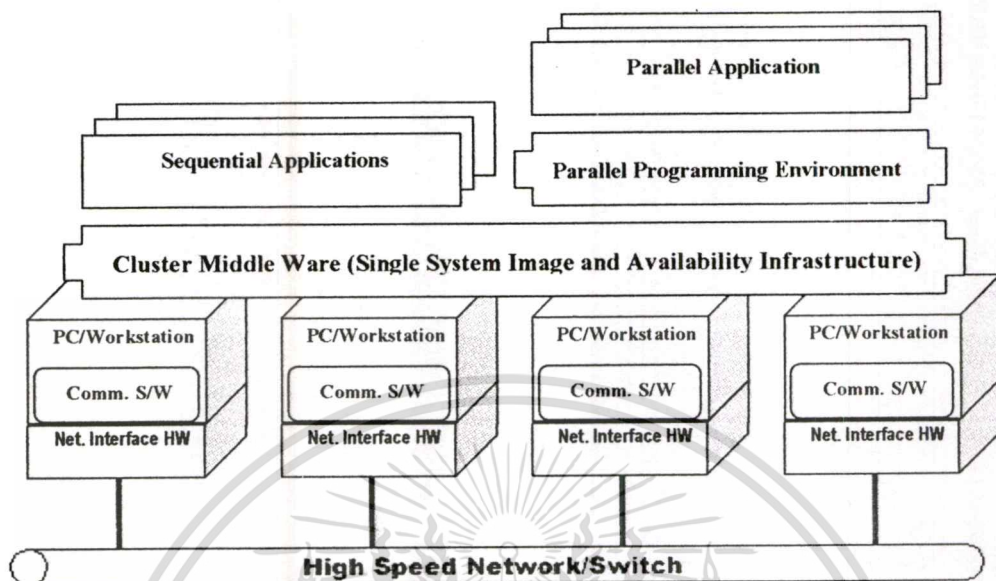
ระบบคลัสเตอร์เกิดจากการเชื่อมต่อเครื่องคอมพิวเตอร์หลายๆเครื่องเข้าด้วยกัน โดยเครื่องคอมพิวเตอร์เหล่านี้จะมีลักษณะเหมือนกันทั้งหมดหรือไม่เหมือนกันเลยก็ได้ การจำแนกนี้สามารถแบ่งระบบคลัสเตอร์ออกเป็นสองพวกคือ

- ระบบคลัสเตอร์แบบเนื้อเดียว (Homogeneous Cluster) ระบบคลัสเตอร์แบบนี้เป็นระบบที่มีองค์ประกอบทางด้านฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในแต่ละเครื่องเหมือนกันทั้งหมด ได้แก่ หน่วยประมวลผลกลาง ชนิดและขนาดของหน่วยความจำ ชนิดและขนาดของฮาร์ดดิสก์ และชนิดของระบบปฏิบัติการเป็นต้น โดยระบบคลัสเตอร์นี้เป็นแบบที่นิยมสร้างกันเนื่องจากการบริหารจัดการระบบสามารถทำได้ง่ายสะดวก นอกจากนั้นการเขียน โปรแกรมเพื่อทำการประมวลผลบนระบบคลัสเตอร์สามารถเขียนในครั้งเดียวแล้วทำงานได้กับทุกๆเครื่องในระบบ

- ระบบคลัสเตอร์แบบเนื้อผสม (Heterogeneous Cluster) ระบบคลัสเตอร์แบบนี้เป็นระบบที่มีความยืดหยุ่นสูง โดยสามารถสร้างจากเครื่องแบบใดก็ได้ที่สนับสนุนการประมวลผลแบบขนาน แต่ปัญหาของระบบนี้คือการสร้าง โปรแกรมสำหรับประมวลผลแบบขนานจะมีความยุ่งยากและซับซ้อนมากขึ้นเนื่องจากต้องทำการสร้าง โปรแกรมที่สามารถประมวลผลเฉพาะกับเครื่องหนึ่งๆ เช่นเครื่องที่ใช้ระบบปฏิบัติการลินุกซ์ (Linux) และระบบที่ใช้ไมโครซอฟท์วินโดวส์ (Microsoft Windows) เป็นระบบปฏิบัติการ โปรแกรมที่สร้างขึ้นมาอาจไม่สามารถใช้ด้วยกันได้วิธีแก้ปัญหอย่างหนึ่งคือสร้าง โปรแกรมโดยใช้ตัวแปรภาษาที่สนับสนุนมาตรฐานเช่น ANSI C เป็นต้น หรืออาจสร้าง โปรแกรมที่สามารถบนประมวลผลได้บนทุกระบบโดยที่ใช้รหัสต้นฉบับ (Source Code) ตัวเดียวกัน เช่นภาษาจาวา (Java) เป็นต้น

4.4 องค์ประกอบและการสร้างระบบคลัสเตอร์

องค์ประกอบของระบบคลัสเตอร์แสดงดังรูปที่ 4.2



รูปที่ 4.2 องค์ประกอบของระบบคลัสเตอร์

รายละเอียดของแต่ละองค์ประกอบมีดังนี้

4.4.1 ระบบเครือข่ายความเร็วสูง (High Speed Network)

การสื่อสารระหว่างแต่ละเครื่องในระบบคลัสเตอร์จะผ่านระบบเครือข่ายความเร็วสูง ซึ่งอุปกรณ์เครือข่ายแต่ละชนิดจะมีความเร็วและราคาแตกต่างกันไป ตัวอย่างอุปกรณ์เครือข่ายที่สามารถใช้สร้างระบบคลัสเตอร์ได้มีดังนี้

- Ethernet ในปัจจุบันอุปกรณ์ Ethernet ได้ถูกพัฒนาให้มีความเร็วในการรับส่งข้อมูลสูงมากขึ้นจนถึงระดับ Gigabit Ethernet หรือ 10 Gigabit Ethernet คือ มีความเร็วในการส่งผ่านข้อมูลประมาณ 1-10 พันล้านบิตต่อวินาที

- Myrinet มีความเร็วในการส่งผ่านข้อมูล 2 พันล้านบิตต่อวินาที (Gigabit per second) และมีค่า Latency Time ต่ำกว่าเครือข่ายแบบ Ethernet มาก แต่ก็มีราคาแพงกว่า Ethernet มาก ในระบบคลัสเตอร์ที่ต้องการความเร็วในการส่งข้อมูลสูงมาก ๆ มักใช้ระบบเครือข่ายนี้ในการสื่อสารระหว่างเครื่อง

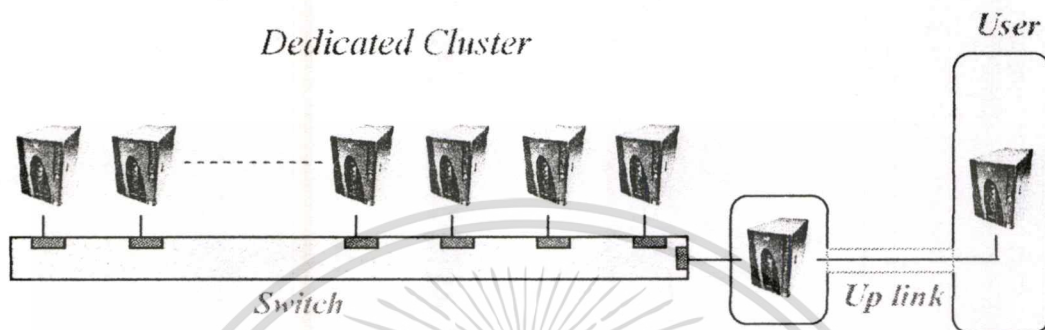
- Quadrics มีความเร็วในการส่งข้อมูลอยู่ที่ 340-900 MB/second หรือประมาณ 2.65-7 Gbit/sec และค่า Latency time มีค่าต่ำมาก

- InfinBand เป็นเทคโนโลยีที่มีความเร็วในการสื่อสารข้อมูลสูงมากถึง 5 Gbit/sec สำหรับการ์ดรุ่นที่มีความเร็ว 4x อีกทั้งค่า Latency Time มีค่าต่ำมาก น้อยกว่า 10 microsecond

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

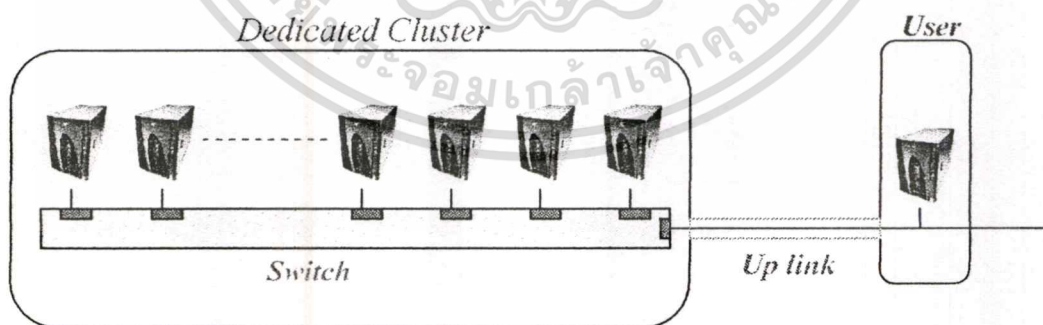
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อแต่ละเครื่องเข้ากับระบบเครือข่ายอาจทำได้ดังรูปที่ 4.3 โดยมีเครื่องเดียวที่เชื่อมต่อกับระบบภายนอกซึ่งอาจเป็นระบบอินเทอร์เน็ตก็ได้ ส่วนเครื่องที่เหลือต่อเข้ากับ Hub หรือ Switching ธรรมดา ลักษณะการเชื่อมต่อแบบรูปที่ 4.4 มีข้อดีคือมีความปลอดภัยสูง ผู้ใช้จะต้องผ่านเครื่องที่ต่อกับเครือข่ายภายนอกก่อนถึงจะสามารถเข้าถึงเครื่องที่อยู่ในเครือข่ายภายในได้



รูปที่ 4.3 การเชื่อมต่อเครื่องในระบบคลัสเตอร์เข้ากับระบบเครือข่ายส่วนตัว (Private Network)

ถึงแม้ว่าระบบในรูปที่ 4.3 จะมีความปลอดภัยสูงแต่ในแง่การประยุกต์ใช้กับงานบางอย่างเช่นงานที่ต้องการเข้าถึงแต่ละเครื่องโดยตรงอาจทำได้ไม่สะดวกเช่นงานเกี่ยวกับศูนย์รวมเซิร์ฟเวอร์ (Server Farm) ในการทำอีเมลล์เซิร์ฟเวอร์ที่มีความคงทนสูง (High Availability Email Server) หรือ เว็บเซิร์ฟเวอร์ที่มีความคงทนสูง (High Availability Web Server) สามารถทำการเชื่อมต่อแต่ละเครื่องเข้ากับระบบเครือข่ายได้ดังรูปที่ 4.4 ซึ่งแต่ละเครื่องเชื่อมต่อกันผ่านระบบเครือข่ายในขณะที่ทุกๆเครื่องก็เชื่อมต่อกับระบบภายนอกด้วยเช่นกัน



รูปที่ 4.4 การเชื่อมต่อระบบเครือข่ายแบบศูนย์รวมเซิร์ฟเวอร์

4.4.2 โปรแกรมบริการการสื่อสาร (Communication Software)

เนื่องจากในวิชานี้พันธบัตรนี้ใช้ MPI เป็น โปรแกรมสำหรับสร้างสภาวะแวดล้อมของการโปรแกรมแบบขนาน (Parallel Programming Environment) ซึ่งจะกล่าวถึงในหัวข้อที่ 4.4.5 โดย MPI สนับสนุนโปรแกรมบริการการสื่อสารสองตัวคือ RSH (Remote Shell) และ SSH (Secure Shell) โปรแกรมทั้งสองตัวนี้ทำหน้าที่จัดการไฟล์และประมวลผลโปรแกรมระหว่างเครื่องได้ Remote Shell จะถูกติดตั้งมาให้ในระบบปฏิบัติการแบบยูนิกซ์ (UNIX) ส่วนมาก แต่มีข้อบกพร่องในเรื่องระบบรักษาความปลอดภัย ในระบบปฏิบัติการแบบยูนิกซ์ยุคใหม่มักไม่แนะนำให้ใช้ Remote Shell เช่นในระบบปฏิบัติการลินุกซ์ (LINUX) ส่วนมากจะมี Secure Shell ติดตั้งมาให้ซึ่งมีความปลอดภัยมากกว่า Remote Shell เนื่องจากมีการเข้ารหัสข้อมูลก่อนทำการส่งข้อมูล ถ้าระบบคลัสเตอร์ไม่ได้เชื่อมต่อกับระบบอินเทอร์เน็ตก็เป็นไปได้ว่า Remote Shell สามารถใช้ได้อย่างปลอดภัย

4.4.3 ระบบปฏิบัติการ (Operating System)

ระบบปฏิบัติการที่สามารถใช้ระบบเครือข่ายได้ส่วนมากสามารถนำมาสร้างเป็นระบบคลัสเตอร์ได้ เช่น ไมโครซอฟท์วินโดวส์ เอ็นที (Microsoft Windows NT) ไมโครซอฟท์วินโดวส์สองพัน (Microsoft Windows 2000) ลินุกซ์ (Linux) ซัน โซลาริส (Sun Solaris) เป็นต้น ระบบปฏิบัติการอื่นๆที่เป็นระบบยูนิกซ์สามารถใช้ร่วมกับระบบลินุกซ์เพื่อทำเป็นระบบคลัสเตอร์ได้ง่าย เนื่องจากมีหลักการออกแบบและโครงสร้างของระบบปฏิบัติการใกล้เคียงกัน

4.4.4 คลัสเตอร์มิดเดิลแวร์ (Cluster Middle Ware)

คลัสเตอร์มิดเดิลแวร์เป็นโปรแกรมที่ทำให้ผู้ใช้ระบบคลัสเตอร์รู้สึกว่าการใช้ระบบเดี่ยวเช่นเดียวกับที่ใช้เครื่องคอมพิวเตอร์ส่วนบุคคลอยู่ โปรแกรมส่วนคลัสเตอร์มิดเดิลแวร์นี้จะต้องทำที่ระดับแก่น (Kernel) ของระบบปฏิบัติการเพื่อที่จะสามารถจัดการถ่ายโอนโปรเซส (Process) ที่เกิดขึ้นไปสู่เครื่องอื่นๆในระบบได้ ตัวอย่างของคลัสเตอร์มิดเดิลแวร์นี้คือ

- BPROC (Beowulf-distributed process space) ซึ่งเป็นโปรแกรมส่วนที่เพิ่มเข้าไปในแก่นของระบบปฏิบัติการลินุกซ์ โดยพัฒนาจากผู้สร้างระบบ Beowulf ในโครงการ CESDIS ของ NASA
- MOSIX เป็นโปรแกรมส่วนที่เพิ่มเข้าไปในส่วนของแก่นของระบบปฏิบัติการเช่นเดียวกับ BPROC ถูกพัฒนาโดยมหาวิทยาลัย Hebrew University of Jerusalem
- Solaris-MC เป็นระบบปฏิบัติการที่สนับสนุนการกระจายงานในระบบคลัสเตอร์ถูกพัฒนาโดย Sun Labs

เนื่องจากว่าระบบคลัสเตอร์ที่ใช้ในวิทยานิพนธ์ฉบับนี้เป็นระบบคลัสเตอร์ที่สร้างขึ้น เฉพาะงาน การกระจายงานอาจไม่จำเป็นต้องใช้คลัสเตอร์มิคเคิลแวร์ แต่การกระจายงานจะทำใน ชั้นของสภาวะแวดล้อมของการ โปรแกรมแบบขนานเช่น MPI หรือ PVM

4.4.5 โปรแกรมสำหรับสร้างสภาวะแวดล้อมของการโปรแกรมแบบขนาน (Parallel Programming Environment)

โปรแกรมสำหรับสร้างสภาวะแวดล้อมของการ โปรแกรมแบบขนานมีหน้าที่ทำให้ โปรแกรมแบบขนานที่ประมวลผลผ่านสภาวะแวดล้อมนี้สามารถกระจายงานได้อย่างอัตโนมัติ ตามที่ผู้สร้างโปรแกรมได้ออกแบบไว้ สภาวะแวดล้อมของการ โปรแกรมแบบขนานแบ่งออกเป็น สองส่วนคือ

4.4.5.1 ส่วนของโปรแกรมสร้างสภาวะแวดล้อมแบบขนาน

ส่วนของโปรแกรมสภาวะแวดล้อมนี้ทำหน้าที่ในการเชื่อมต่อระหว่างเครื่องในระบบ คลัสเตอร์ เมื่อทำการประมวลผลโปรแกรมแบบขนานที่ออกแบบโดยใช้คลังชุดคำสั่ง (Library) ของการ โปรแกรมแบบขนานนี้ สภาวะแวดล้อมแบบขนานจะทำการส่งงานที่ผู้สร้าง โปรแกรม กำหนด ไปประมวลผลบนเครื่องที่เลือกไว้โดยอัตโนมัติ หรือถ้าใน โปรแกรมที่สร้างขึ้น ไม่ได้ระบุ ชื่อของเครื่องที่ต้องการให้ประมวลผล สภาวะแวดล้อมแบบขนานนี้จะทำการกระจายงานให้โดย อัตโนมัติคล้ายกับหน้าที่ของคลัสเตอร์มิคเคิลแวร์ แต่ส่วนของสภาวะแวดล้อมแบบขนานนี้มีความ ยืดหยุ่นและประสิทธิภาพมากกว่า แต่ก็สร้างได้ยากกว่าเนื่องจากขึ้นอยู่กับโปรแกรมประยุกต์ที่ สร้างขึ้น

4.4.5.2 ส่วนคลังชุดคำสั่ง (Library) ของการโปรแกรมแบบขนาน

โปรแกรมแบบขนานจำเป็นต้องทำการสื่อสารผ่านระบบเครือข่าย ทางเลือกหนึ่งที่ สะดวกรวดเร็วและมีประสิทธิภาพคือใช้คลังชุดคำสั่งแบบขนานที่มีผู้พัฒนาไว้แล้ว การกระจาย งานอาจประกอบด้วยการกระจายให้เครื่องทุกเครื่องในระบบหรือกระจายสู่บางเครื่อง หรือกระจาย สู่กลุ่มของเครื่องที่มีระบบปฏิบัติการเดียวกันหรือกลุ่มของเครื่องที่ได้ออกแบบไว้ก่อน นอกจากนั้น คลังชุดคำสั่งยังได้จัดเตรียมวิธีการส่งข้อมูลชนิดต่างๆเช่น จำนวนเต็ม ทศนิยม หรือแม้แต่ข้อมูลที่มี ความสลับซับซ้อนมากๆเช่นข้อมูลในหน่วยความจำที่ไม่อยู่ติดเป็นกลุ่มเดียวกันเป็นต้น

โปรแกรมในส่วนของสภาวะแวดล้อมของการ โปรแกรมแบบขนานที่นิยมใช้ในงานวิจัย มีสองตัวคือ

- Parallel Virtual Machine (PVM)ถูกพัฒนาโดย Oak Ridge National Laboratory มี ลักษณะเป็นเครื่องเสมือน (Virtual Machine) เพื่อใช้สำหรับประมวลผลโปรแกรมแบบขนาน เริ่มแรกผู้ใช้ต้องทำการเพิ่มเครื่องในระบบคลัสเตอร์เข้าไปในส่วนที่เป็นเครื่องเสมือนจากนั้นเมื่อทำ

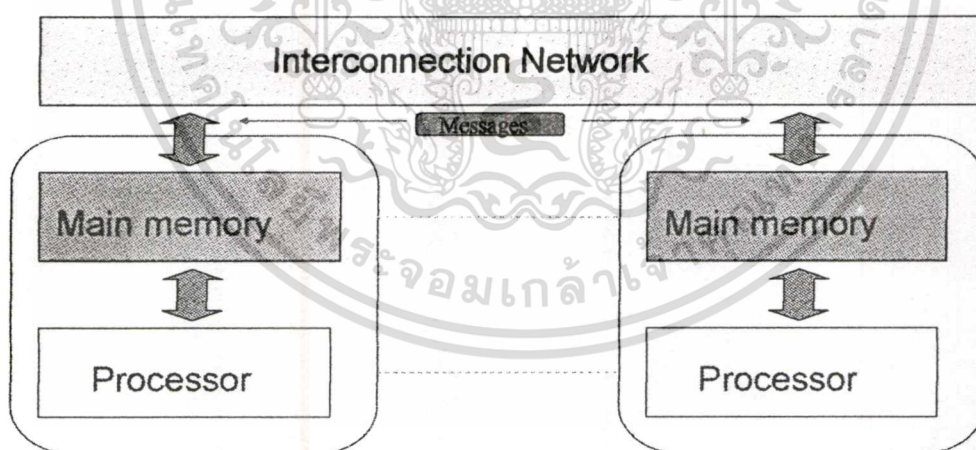
การประมวลผลโปรแกรมส่วนที่เป็นเครื่องเสมือนจะทำการกระจายงานไปทำบนเครื่องที่ต้องการตามที่ได้ออกแบบไว้ในโปรแกรม

- Message Passing Interface (MPI) เป็นมาตรฐานของการประมวลผลแบบขนานในระบบคลัสเตอร์ ซึ่งสนับสนุนภาษา C และ Fortran ในวิทยานิพนธ์ฉบับนี้ใช้ MPICH ซึ่งเป็น คลังชุดคำสั่งของการส่งข้อมูลที่สร้างตามมาตรฐาน MPI ลักษณะของการใช้คลังชุดคำสั่งและการโปรแกรมแบบขนานในลักษณะต่างๆมีดังนี้

4.5 การสื่อสารระหว่างงานด้วยวิธีส่งข้อความ (Message Passing)

จากบทที่ 4 ได้กล่าวถึงโครงสร้างของคอมพิวเตอร์แบบขนานซึ่งระบบคลัสเตอร์เป็นระบบคอมพิวเตอร์แบบขนานที่สื่อสารระหว่างงานด้วยการส่งข้อความ และแต่ละเครื่องมีหน่วยความจำเป็นของตัวเอง (DMMP) ซึ่งแสดงดังรูปที่ 4.5 วิธีการส่งข้อความในระบบแบบหลายคอมพิวเตอร์ (Multicomputer) สามารถทำได้โดย

- ออกแบบตัวแปลภาษาแบบพิเศษซึ่งสนับสนุนการประมวลผลแบบขนาน
- เพิ่มไวยากรณ์ (Syntax) ในตัวแปลภาษาที่มีอยู่แล้วให้สนับสนุนการสื่อสารแบบส่งข้อความ
- ใช้ตัวแปลภาษาที่มีอยู่แล้วร่วมกับคลังชุดคำสั่ง (Library) ซึ่งสนับสนุนการสื่อสารแบบส่งข้อความ



รูปที่ 4.5 แต่ละเครื่องในระบบคลัสเตอร์มีหน่วยความจำของตัวเอง

ตัวแปลภาษาที่สนับสนุนการสื่อสารแบบส่งข้อความได้แก่ OCCAM ซึ่งออกแบบมาเพื่อใช้กับหน่วยประมวลผลที่เรียกว่า Transputer ส่วนตัวแปลภาษาที่มีการเพิ่มไวยากรณ์ของการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สื่อสารด้วยการส่งข้อความได้แก่ C C++ และ Fortran M ในวิทยานิพนธ์ฉบับนี้ใช้คลังชุดคำสั่งของการสื่อสารแบบส่งข้อความร่วมกับตัวแปรภาษาที่มีอยู่แล้วโดยคลังชุดคำสั่งนี้มีชื่อว่า MPI (Message Passing Interface) คลังชุดคำสั่งอีกตัวหนึ่งที่นิยมใช้กันคือ PVM (Parallel Virtual Machine)

ก่อนที่ระบบคอมพิวเตอร์จะทำการประมวลผลงานจะต้องมีการสร้างโปรเซสขึ้น โดยโปรเซสจะเกิดจากระบบปฏิบัติการ การสร้างโปรเซสในระบบคลัสเตอร์จะถูกทำโดยระบบปฏิบัติการของแต่ละหน่วยประมวลผลเช่นเดียวกันกับระบบคอมพิวเตอร์อื่นๆ การควบคุมการสร้างโปรเซสจะทำจากเครื่องที่ถูกกำหนดเป็นผู้จัดการ(Manager) หรือเครื่องที่ทำการส่งประมวลผลโปรแกรม โดยทั่วไปการสร้างงานในระบบคลัสเตอร์ที่ใช้คลังชุดคำสั่งในการสื่อสารระหว่างงานจะมีสองแบบคือ

- การสร้างโปรเซสแบบกำหนดไว้ก่อน (Static Process Creation) วิธีการสร้างโปรเซสแบบนี้จะใช้ในคลังชุดคำสั่ง MPI ซึ่งการสร้างโปรเซสจะเกิดตอนเริ่มตั้งให้ประมวลผลโปรแกรม
- การสร้างโปรเซสแบบกำหนดในขณะโปรแกรมทำงาน (Dynamic Process Creation) ซึ่งการสร้างโปรเซสในลักษณะนี้จะเกิดเมื่อใดก็ได้เมื่อโปรเซสที่เป็นผู้จัดการยังทำงานอยู่ การสร้างโปรเซสลักษณะนี้จะใช้ในคลังชุดคำสั่ง PVM

ใน MPI จะใช้แบบจำลองโปรแกรมแบบโปรแกรมเดี่ยวหลายข้อมูล (Single Program Multiple Data) หรือ SPMD ซึ่งทุกๆ โปรเซสที่ถูกสร้างในระบบคลัสเตอร์จะประมวลผลโปรแกรมเดียวกันแต่จะสามารถควบคุมให้โปรเซสที่อยู่ในหน่วยประมวลผลที่ต่างกันสามารถประมวลผลในส่วนของโปรแกรมที่ต่างกันได้ ดังแสดงในรูปที่ 4.6

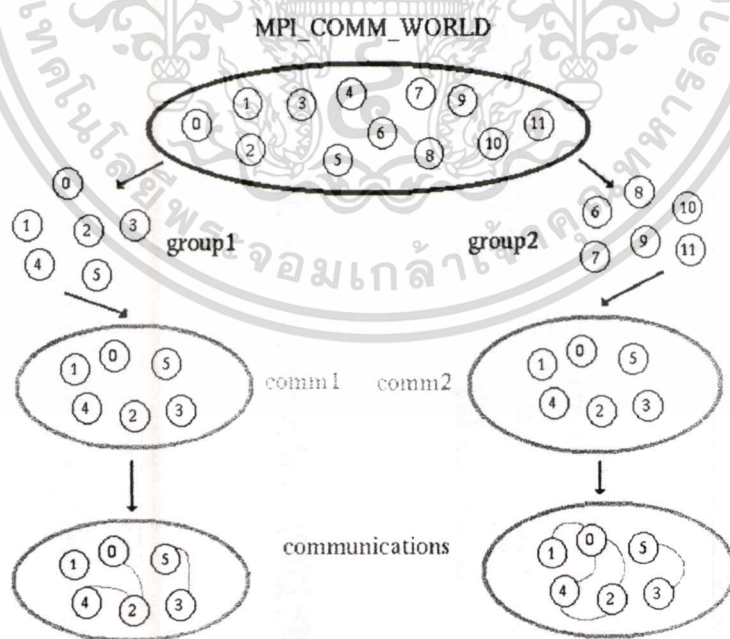
```
int main(int argc, char *argv[])
{
    if(tid == 0){
        ... Manager Code ...
    }else{
        ... Worker Code ...
    }
}
```

รูปที่ 4.6 ลักษณะรหัสต้นฉบับซึ่งมีโครงสร้างแบบโปรแกรมเดี่ยวหลายข้อมูล

จากรูปที่ 4.6 เป็นรหัสต้นฉบับของโปรแกรมแบบขนาน ค่า tid ที่ได้จากคำสั่งคำสั่ง MPI จะบอก ว่าโปรแกรมกำลังประมวลผลอยู่บน โพรเซสใด ถ้า tid มีค่าเท่ากับศูนย์แสดงว่าโปรแกรมกำลังถูก ประมวลผลอยู่บน โพรเซสที่ทำหน้าที่เป็นผู้จัดการ ส่วนโพรเซสที่มีค่า tid อื่นๆจะเป็น โพรเซสที่มี หน้าที่เป็นคณงาน จากคุณสมบัติของค่า tid ดังกล่าวนี้อาจทำให้สามารถใส่รหัสต้นฉบับของ โปรแกรมแบบขนานสำหรับงานต่างๆในระบบคลัสเตอร์ไว้ในโปรแกรมเดียวได้โดยใช้ประโยค เงื่อนไข if else เป็นตัวแบ่งแยก

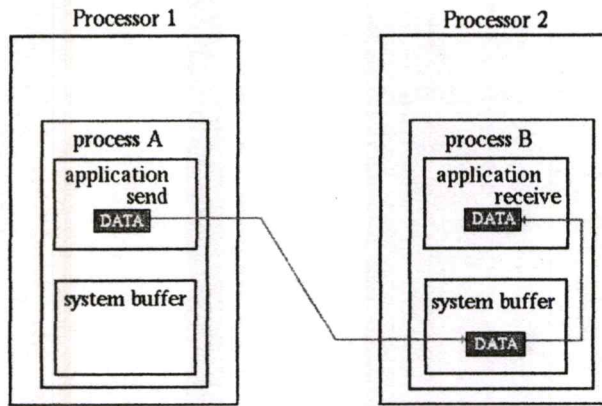
งานใน MPI สามารถจัดรวมกลุ่มได้ด้วยสิ่งที่เรียกว่า COMMUNICATOR งานทั้งหมด จะถูกระบุด้วยตัวแปร MPI_COMM_WORLD ผู้สร้างโปรแกรมสามารถกำหนดกลุ่มของงาน เพิ่มเติมเพื่อให้สามารถจัดการงานเหล่านั้น ได้ง่ายขึ้นดังแสดงในรูปที่ 4.7

จากหัวข้อที่ 5.3.2 ซึ่งเป็นการออกแบบลักษณะการสื่อสารระหว่างงานในระบบคลัส เตอร์ MPI ได้จัดเตรียมคำสั่งคำสั่งสำหรับการส่งและรับข้อมูลลักษณะต่างๆ ซึ่งสามารถ ประยุกต์ใช้ใน โปรแกรมแบบขนานที่ออกแบบไว้ ลักษณะของการส่งและรับข้อมูลของ MPI เป็น ดังรูปที่ 4.8 ซึ่งเป็นการส่งข้อมูลจากโพรเซส A ของหน่วยประมวลผล 1 ไปสู่โพรเซส B ในหน่วย ประมวลผล 2 การส่งข้อมูลจะส่งโดยตรงจากตัวแปรของโปรแกรมส่วนการรับข้อมูลจะทำผ่านตัว เก็บข้อมูลของระบบ (Buffer) ก่อนแล้วจึงคัดลอกสู่ตัวแปรของโปรแกรมอีกครั้งหนึ่งเพื่อให้ สามารถสื่อสารแบบไม่ประสานกันได้ (Asynchronous) การสื่อสารของ MPI สามารถแบ่งได้เป็น การสื่อสารแบบจุดต่อจุด (Point-Point Communication) และแบบสื่อสารเป็นกลุ่ม (Collective Communication)



รูปที่ 4.7 การจัดกลุ่มงานใน MPI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Path of a message buffered at the receiving process

รูปที่ 4.8 การส่งและรับข้อมูลของ MPI

4.5.1 การสื่อสารแบบจุดต่อจุด (Point-Point Communication)

การสื่อสารแบบจุดต่อจุดนี้เป็นพื้นฐานของระบบสื่อสารแบบส่งข้อความ ซึ่งข้อความจะถูกส่งระหว่างโปรเซสสองโปรเซส โดยด้านหนึ่งเป็นผู้ส่งและอีกด้านหนึ่งเป็นผู้รับ คลังชุดคำสั่งที่สนับสนุนการส่งข้อความลักษณะนี้แบ่งเป็นสองกลุ่มคือการส่งและรับข้อความแบบมีการยืนยัน และการส่งและรับข้อความแบบไม่มีการยืนยัน

4.5.1.1 การส่งและรับข้อความแบบมีการยืนยัน (Blocking Send and Blocking Receive)

การส่งแบบนี้จะเป็นการส่งข้อมูลที่มีความแน่นอน ขั้นตอนการส่งจะสิ้นสุดและคืนการทำงานให้กับโปรแกรมเมอร์เมื่อผู้รับได้รับข้อมูลเรียบร้อยแล้ว ข้อควรระวังในการใช้การส่งข้อความด้วยวิธีนี้คือการเกิดเหตุการณ์ที่อ็อกตาย (Dead Lock) ซึ่งเป็นเหตุการณ์ที่งานรอการส่งข้อมูลตลอดไปโดยไม่มีการคืนการทำงานให้โปรแกรม ทำให้ไม่สามารถทำงานเสร็จได้ คำสั่งของ MPI ในกลุ่มคำสั่งนี้ได้แก่ MPI_SEND และ MPI_RECV

4.5.1.2 การส่งและรับข้อความแบบไม่มีการยืนยัน (Non-Blocking Send and Non-Blocking Receive)

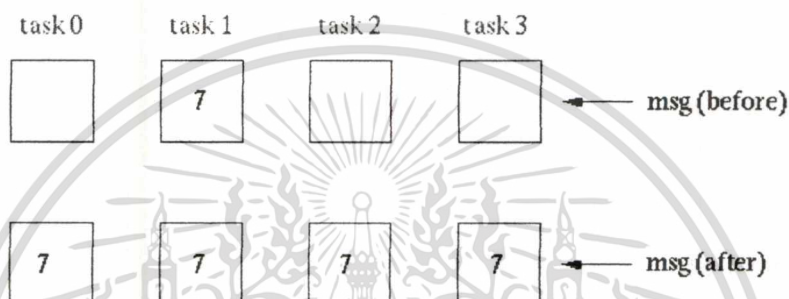
การส่งและรับข้อความแบบไม่มีการยืนยันจะถูกทำเพื่อปรับปรุงประสิทธิภาพของการส่งข้อความ การส่งข้อความแบบไม่มีการยืนยันนี้จะไม่รอให้ฝ่ายรับข้อความได้รับข้อความก่อนเหมือนการส่งข้อมูลแบบมีการยืนยัน โดยจะคืนการทำงานให้โปรแกรมทันที การส่งข้อความลักษณะนี้มีความเสี่ยงในความปลอดภัยเมื่อมีการเขียนทับตัวแปรของงานที่ต้องการส่งข้อความและรับข้อความก่อนที่ข้อความจะถูกส่งไปจริง MPI จึงมีคำสั่งที่ตรวจสอบความสำเร็จของการส่งและรับด้วยคำสั่ง MPI_WAIT และ MPI_TEST เพื่อตรวจสอบสถานะการส่งและรับข้อความก่อนที่จะมี

การเขียนข้อมูลทับข้อมูลเก่า ตัวอย่างคำสั่งของการส่งและรับข้อความแบบไม่มีการยืนยันคือ MPI_ISEND และ MPI_IRECV

4.5.2 สื่อการเป็นกลุ่ม (Collective Communication)

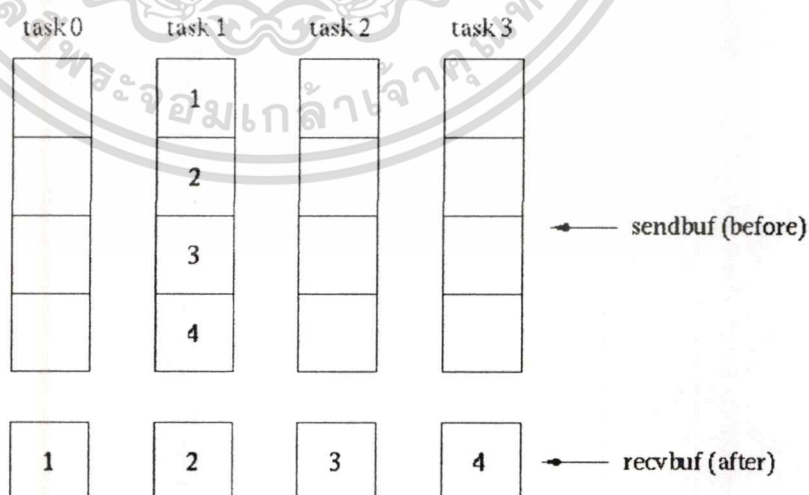
การสื่อสารแบบเป็นกลุ่มจะเป็นการสื่อสารของทุกโพรเซสในกลุ่ม ตัวอย่างคำสั่งของการสื่อสารแบบเป็นกลุ่มคือ

- MPI_Bcast เป็นคำสั่งในการกระจายค่าในตัวแปร (msg) ไปให้ทุกๆโพรเซสในกลุ่ม ดังแสดงในรูปที่ 4.9



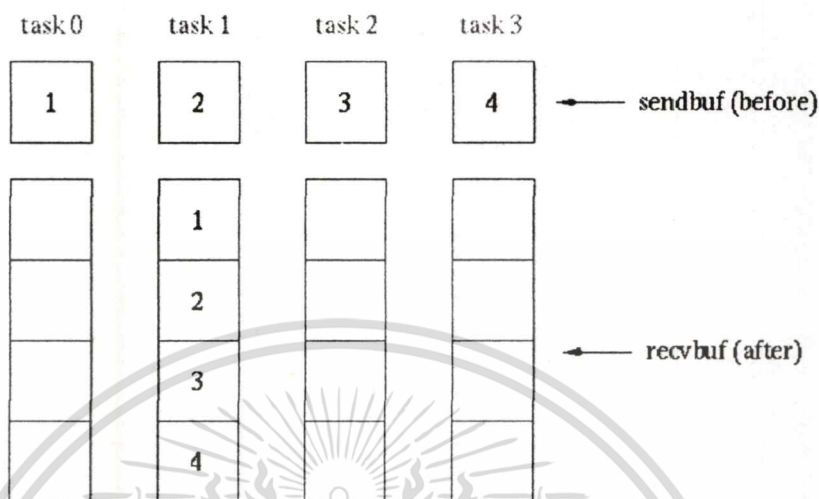
รูปที่ 4.9 ลักษณะการทำงานของ MPI_Bcast

- MPI_Scatter เป็นคำสั่งในการกระจายค่าในอะเรย์ (Array) สำหรับส่งข้อมูล (sendbuf) ไปสู่ตัวแปรของทุกๆโพรเซสในกลุ่ม (recvbuf) ตามลำดับตำแหน่งของอะเรย์ที่ใช้ส่งนั้น ดังแสดงในรูปที่ 4.10



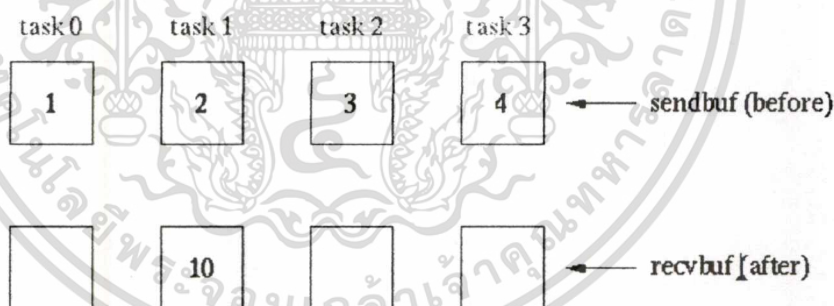
รูปที่ 4.10 ลักษณะการทำงานของ MPI_Scatter

- MPI_Gather เป็นการเก็บรวบรวมค่าในตัวแปร (sendbuf) ของโปรเซสในกลุ่มมาใส่ในอะเรย์ (recvbuf) ของโปรเซสที่เรียกใช้คำสั่งตามลำดับงานดังแสดงในรูปที่ 4.11



รูปที่ 4.11 ลักษณะการทำงานของ MPI_Gather

- MPI_Reduce เป็นการรวมค่าที่เก็บอยู่ในตัวแปร (sendbuf) ของทุกๆ โปรเซสมาเก็บในตัวแปร (recvbuf) ของโปรเซสที่เรียกใช้คำสั่งนี้ดังแสดงในรูปที่ 4.12



รูปที่ 4.12 ลักษณะการทำงานของ MPI_Reduce

4.6 สรุป

ในบทนี้ได้กล่าวถึงการทำงานของระบบคอมพิวเตอร์ การประยุกต์ใช้การประมวลผลแบบขนาน และองค์ประกอบที่จำเป็นในการสร้างระบบคลัสเตอร์ ซึ่งประกอบด้วยระบบเครือข่ายความเร็วสูงและการเชื่อมต่อแบบต่างๆ โปรแกรมสำหรับการสื่อสาร ระบบปฏิบัติการ คลัสเตอร์มีดเคิลแวร์ และ โปรแกรมสร้างสภาวะแวดล้อมของการ โปรแกรมแบบขนาน

ในบทต่อไปจะกล่าวถึงการเขียนโปรแกรมแบบขนานลักษณะต่างๆ และการสร้าง
 ภาพตัดขวางโดยที่ใช้อัลกอริทึมเชิงพีชคณิตสำหรับวิเคราะห์ทางเดินแสงแบบกรวยด้วยระบบคลัส
 เทอร์โดยจะนำเสนอเป็นลำดับต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

โปรแกรมแบบขนาน

5.1 บทนำ

การประมวลผลแบบขนานนอกจากต้องมีระบบคอมพิวเตอร์แบบขนานแล้วยังจำเป็นต้องมีโปรแกรมแบบขนานเพื่อใช้สำหรับประมวลผลบนระบบคอมพิวเตอร์แบบขนาน ลักษณะของโปรแกรมแบบขนานจะขึ้นอยู่กับลักษณะของระบบคอมพิวเตอร์แบบขนาน โดยโปรแกรมแบบขนานสำหรับระบบคอมพิวเตอร์ขนานแบบหนึ่งๆอาจไม่สามารถทำงานได้บนระบบคอมพิวเตอร์ขนานแบบอื่น

การสร้างโปรแกรมแบบขนานจะมีความยากง่ายแตกต่างกันไปขึ้นอยู่กับการประยุกต์ใช้งาน เครื่องมือที่ใช้ในการสร้างโปรแกรมแบบขนานและชนิดของระบบคอมพิวเตอร์แบบขนาน นอกจากนี้ประสิทธิภาพของโปรแกรมแบบขนานของงานหนึ่งๆอาจไม่เท่ากันเมื่อสร้างโปรแกรมแบบขนานนี้ด้วยวิธีที่ต่างกัน

การสร้างโปรแกรมแบบขนานมีสองวิธี

1. การสร้างโปรแกรมแบบขนานโดยนัย (Implicit Parallelism) วิธีนี้สามารถทำได้ง่ายโดยใช้ภาษาแบบขนานหรือตัวแปลภาษา (Compiler) แบบขนาน แต่วิธีนี้ผู้สร้างโปรแกรม (Programmer) ไม่สามารถควบคุมการจัดการ การแบ่งส่วนคำนวณ และการแบ่งข้อมูลได้ ซึ่งโดยทั่วไปแล้วจะมีประสิทธิภาพต่ำกว่าการสร้างโปรแกรมแบบขนานโดยตรง (Explicit Parallelism)

2. การสร้างโปรแกรมแบบขนานโดยตรง วิธีนี้ผู้สร้างโปรแกรมต้องทำการจัดแบ่งการทำงานเองทั้งหมด ซึ่งรูปแบบ และวิธีการแบ่งงานในแต่ละการประยุกต์ใช้จะไม่เหมือนกัน ซึ่งจะทำให้มีความยากง่ายแตกต่างกันด้วย แต่การใช้วิธีการแบ่งงานที่เหมาะสมกับการประยุกต์ใช้จะทำให้ได้ประสิทธิภาพในการประมวลผลสูงกว่าการสร้างโปรแกรมแบบขนานโดยนัย

5.2 รูปแบบของการพัฒนาโปรแกรมแบบขนาน

ก่อนที่จะสร้างโปรแกรมแบบขนานส่วนมากผู้สร้างโปรแกรมมักจะสร้างโปรแกรมในแบบลำดับ (Sequential Program) ซึ่งเป็นโปรแกรมที่ใช้ประมวลผลในเครื่องคอมพิวเตอร์แบบธรรมดาให้สามารถทำงานได้อย่างถูกต้องก่อน จากนั้นจะพัฒนาโปรแกรมแบบลำดับนี้ไปเป็นโปรแกรมแบบขนานได้จากวิธีเหล่านี้

5.2.1 การสร้างโปรแกรมแบบขนานโดยอัตโนมัติ (Automatic Parallelism)

การสร้างโปรแกรมแบบขนานด้วยวิธีนี้เป็นวิธีที่ง่ายที่สุด แต่ก็จะมีประสิทธิภาพต่ำที่สุด โดยหน้าที่ของการสร้างโปรแกรมแบบขนานจะเป็นหน้าที่ของตัวแปรภาษาซึ่งตัวแปรภาษาจะตรวจสอบรหัสต้นฉบับ (Source Code) ซึ่งอาจประกอบด้วยส่วนของรหัสที่มีการวนซ้ำ และการประมวลผลกับข้อมูลที่ซ้ำๆกัน โดยตัวแปรภาษานี้จะทำการเปลี่ยนรหัสต้นฉบับนี้ไปเป็นรหัสที่สนับสนุนการประมวลผลแบบขนานจากนั้นใช้ตัวแปรภาษาเปลี่ยนรหัสที่สนับสนุนการประมวลผลแบบขนานนี้ไปเป็นโปรแกรมแบบขนาน ข้อจำกัดของการสร้างโปรแกรมแบบขนานด้วยวิธีนี้ขึ้นอยู่กับเทคโนโลยีที่ตัวแปรภาษาใช้ในการเปลี่ยนรหัสต้นฉบับเป็นรหัสที่สนับสนุนการประมวลผลแบบขนาน ขั้นตอนการทำงานของการสร้างโปรแกรมแบบขนานโดยอัตโนมัติซึ่งแสดงดังรูปที่ 5.1



รูปที่ 5.1 ขั้นตอนการทำงานของการสร้างโปรแกรมแบบขนานโดยอัตโนมัติ

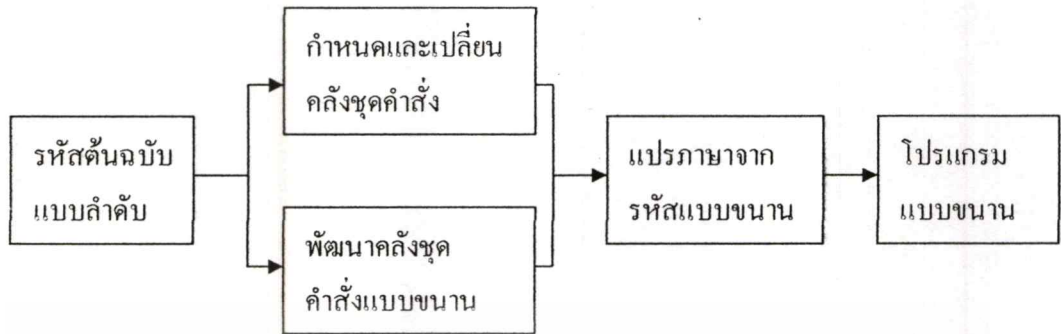
5.2.2 การสร้างโปรแกรมแบบขนานโดยใช้คลังของชุดคำสั่งแบบขนาน (Parallel Library)

วิธีนี้เป็นวิธีที่มีประสิทธิภาพดีกว่าวิธีแรก โดยใช้คลังของชุดคำสั่งแบบขนานที่พัฒนาไว้แล้วเช่น คลังชุดคำสั่งของการแปลงอนุกรมฟูเรียร์ (Fourier) แบบขนาน และการคูณเมตริกแบบขนาน เป็นต้น โดยการเปลี่ยนชุดคำสั่งจากรหัสต้นฉบับเป็นรหัสที่สนับสนุนการประมวลผลแบบขนานจะถูกทำโดยผู้สร้างโปรแกรมแบบขนาน จากนั้นใช้ตัวแปรภาษาทำการเปลี่ยนรหัสที่สนับสนุนการประมวลผลแบบขนานไปเป็นโปรแกรมแบบขนาน ขั้นตอนการสร้างโปรแกรมแบบขนานโดยใช้คลังของชุดคำสั่งแบบขนานแสดงดังรูปที่ 5.2 การใช้คลังของชุดคำสั่งแบบขนานสามารถสร้างโปรแกรมแบบขนานในสองรูปแบบ

- ใช้ควบคุมโครงสร้างของโปรแกรมแบบขนานในการกระจายงานไปตามส่วนต่างๆ ของหน่วยการประมวลผล (Processing Elements)
- ใช้ในการทดแทนชุดคำสั่งต้นฉบับที่สามารถเปลี่ยนการทำงานเป็นแบบขนานได้ โดยใช้คลังชุดคำสั่งที่สนับสนุนการทำงานแบบขนานที่พัฒนาขึ้น

ในวิทยานิพนธ์ฉบับนี้ใช้วิธีการสร้างโปรแกรมแบบขนานโดยใช้คลังของชุดคำสั่งแบบขนาน ซึ่งใช้เพื่อควบคุมโครงสร้างของโปรแกรมแบบขนานในการกระจายงานไปตามโพรเซส (Processes)

ต่างๆที่อยู่ในหน่วยการประมวลผล (Processors) แต่ไม่ได้ใช้คลังชุดคำสั่งในการเปลี่ยนรหัส
ต้นฉบับเป็นรหัสแบบขนาน



รูปที่ 5.2 ขั้นตอนการสร้างโปรแกรมแบบขนานโดยใช้คลังของชุดคำสั่งแบบขนาน

5.2.3 การสร้างโปรแกรมแบบขนานด้วยตัวเอง (Major Recoding)

การสร้างโปรแกรมแบบขนานด้วยวิธีนี้เป็นแบบที่ซับซ้อนมากที่สุด ผู้สร้างโปรแกรมสามารถเลือกตัวแปลภาษาใดก็ได้ที่ต้องการ รวมถึงสามารถเลือกรูปแบบและโปรแกรมที่ใช้ในการสื่อสาร แต่วิธีนี้นับเป็นวิธีที่ยากและเสียเวลามากที่สุด ขั้นตอนการสร้างโปรแกรมแบบขนานด้วยตัวเองแสดงดังรูปที่ 5.3



รูปที่ 5.3 ขั้นตอนการสร้างโปรแกรมแบบขนานด้วยตัวเอง

5.3 วิธีการออกแบบอัลกอริทึมของโปรแกรมแบบขนาน (Methodical Design of Parallel Algorithms)

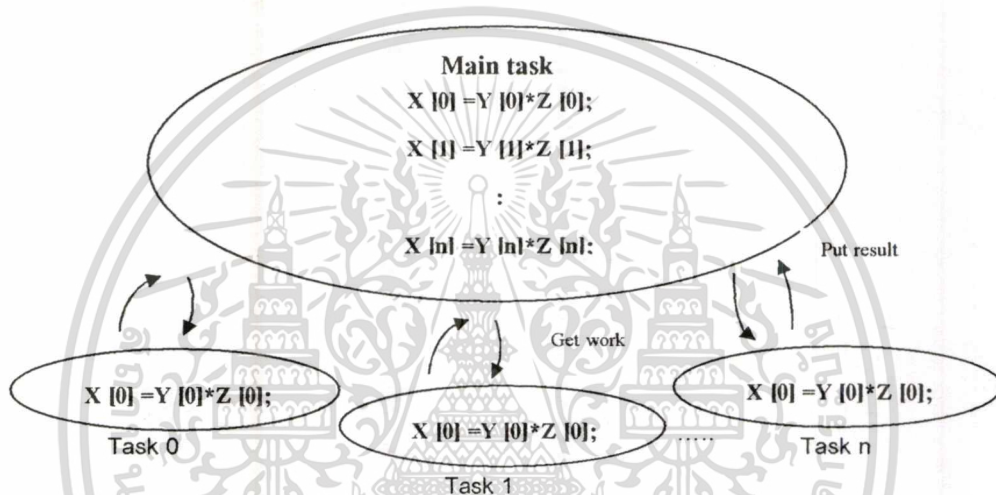
การออกแบบอัลกอริทึมของโปรแกรมแบบขนานไม่มีข้อกำหนดตายตัว Ian Foster ได้แนะนำวิธีการออกแบบอัลกอริทึมแบบขนานของโปรแกรมไว้สี่ขั้นตอน คือ

- ขั้นตอนการแบ่งงาน (Partitioning)
- ขั้นตอนออกแบบการสื่อสาร (Communication)
- ขั้นตอนการรวมกลุ่มงาน (Agglomeration)
- ขั้นตอนกำหนดงาน (Mapping)

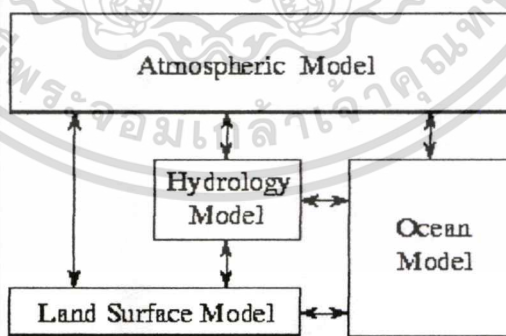
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.1 ขั้นตอนการแบ่งงาน

การแบ่งงานเป็นหัวใจของการประมวลผลแบบขนาน ซึ่งเป็นการแยกการคำนวณ ออกเป็นส่วนย่อย การแบ่งงานสามารถทำได้สองแบบคือแบ่งข้อมูลของปัญหาออกเป็น ส่วนย่อย (Domain Decomposition) แบบที่สองคือแบ่งปัญหาตามหน้าที่ของการคำนวณออกเป็น ส่วนย่อย (Functional Decomposition) ตัวอย่างของการแบ่งข้อมูลของปัญหาแสดงดังรูปที่ 5.4 ซึ่งเป็นการ แบ่งข้อมูลในการคำนวณออกเป็น ส่วนๆ ส่วนการแบ่งงานตามหน้าที่ของปัญหาแสดงดังรูปที่ 5.5 ซึ่งเป็นตัวอย่างของการสร้างแบบจำลองของสภาพภูมิอากาศ ซึ่งประกอบด้วยงานย่อยๆคือการสร้าง แบบจำลองของทะเล พื้นดิน และก๊าซ โดยทั่วไปแล้วการแบ่งข้อมูลจะมีความยืดหยุ่นและสามารถ ทำได้ง่ายกว่าการแบ่งงานตามหน้าที่



รูปที่ 5.4 การแบ่งขอบเขตข้อมูลของปัญหา



รูปที่ 5.5 การแบ่งขอบเขตหน้าที่ของปัญหา

หรือสามารถแบ่งงานโดยการกำหนดลักษณะการติดต่อกัน โดยจำแนกตามจำนวนของชุดคำสั่งและ ข้อมูลได้ดังนี้

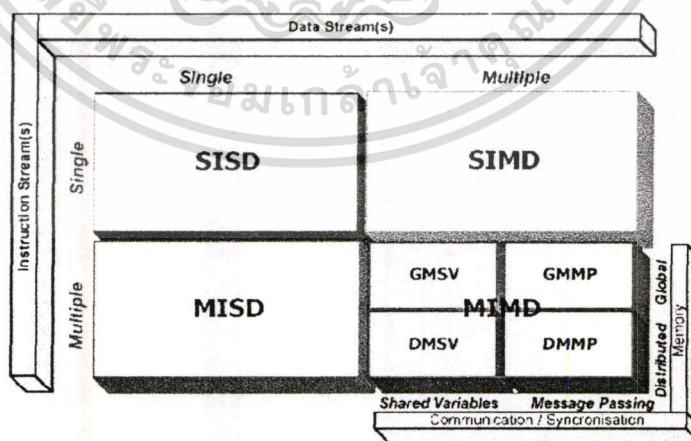
- SISD : ชุดคำสั่งเดี่ยว ข้อมูลเดี่ยว (Single Instruction Stream / Single Data Stream)
- SIMD: ชุดคำสั่งเดี่ยว หลายข้อมูล (Single Instruction Stream / Multiple Data Stream)
- MISD: หลายชุดคำสั่ง ข้อมูลเดี่ยว (Multiple Instruction Stream / Single Data Stream) ยังไม่มีระบบแบบนี้เกิดขึ้นจริง
- MIMD: หลายชุดคำสั่ง หลายข้อมูล (Multiple Instruction Stream / Multiple Data Stream)

ตัวอย่างของระบบชุดคำสั่งเดี่ยวข้อมูลเดี่ยวคือเครื่องคอมพิวเตอร์ส่วนบุคคล (Personal Computer) ซึ่งชุดคำสั่งจะถูกประมวลผลตามลำดับแต่อาจจะมีการซ้อนทับ (Overlap) ของชุดคำสั่งในขั้นตอนประมวลผลได้โดยใช้หลักการของท่อลำเลียง (Pipelining) ระบบคอมพิวเตอร์แบบขนานจะจัดอยู่ในพวกระบบหลายชุดคำสั่งหลายข้อมูลซึ่งอาจเป็นระบบหลายหน่วยประมวลผล (Multiprocessor) และระบบหลายเครื่องคอมพิวเตอร์ (Multiple Computer)

ระบบคอมพิวเตอร์แบบหลายชุดคำสั่งหลายข้อมูลนี้สามารถแบ่งย่อยได้สี่แบบคือ

- GMSV: หน่วยความจำรวม ตัวแปรร่วม (Global Memory / Shared Variables)
- GMMP: หน่วยความจำรวม ส่งข้อความ (Global Memory / Message Passing) ยังไม่มีระบบแบบนี้เกิดขึ้นจริง
- DMSV: หน่วยความจำแบบกระจาย ตัวแปรร่วม (Distributed Memory / Shared Variables)
- DMMP: หน่วยความจำแบบกระจาย ส่งข้อความ (Distributed memory / Message Passing)

แผนผังการจำแนกประเภทของระบบคอมพิวเตอร์แสดงดังรูปที่ 5.6



รูปที่ 5.6 แผนผังการจำแนกประเภทของระบบคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบคอมพิวเตอร์แบบขนานที่ใช้ในวิทยานิพนธ์ฉบับนี้เป็นระบบแบบ DMMP ซึ่งเป็นระบบที่ใช้เครื่องคอมพิวเตอร์มาทำการเชื่อมต่อกันผ่านระบบเครือข่ายความเร็วสูงและทำงานร่วมกันเสมือนเป็นระบบเดียว โดยระบบชนิดนี้เรียกว่าระบบคลัสเตอร์

5.3.2 ขั้นตอนนอกแบบการสื่อสาร

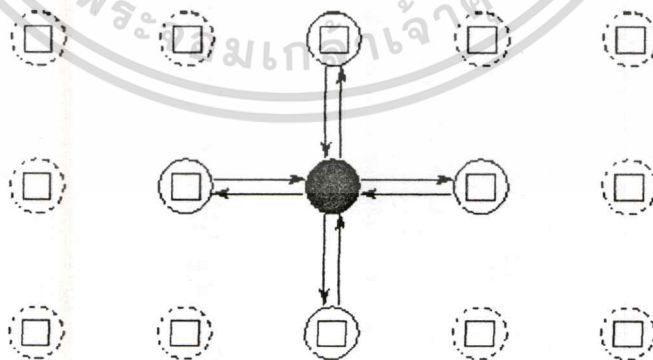
ขั้นตอนการสื่อสารจะถูกทำโดยพิจารณาการสื่อสารระหว่างงานที่แบ่งออกไป ในงานบางอย่าง งานย่อยที่ถูกแบ่งไม่สามารถประมวลผลได้ในเวลาเดียวกันเนื่องจากต้องอาศัยข้อมูลจากงานย่อยอื่นเพื่อใช้ในการประมวลผล ดังนั้นจึงต้องมีการสื่อสารระหว่างงานย่อย การเลือกวิธีการสื่อสารที่เหมาะสมจะทำให้การประมวลผลมีประสิทธิภาพมากยิ่งขึ้น การออกแบบขั้นตอนการสื่อสารจะแบ่งเป็นสองขั้นตอน ขั้นตอนแรกออกแบบโครงสร้างของการเชื่อมโยงว่างานใดต้องการข้อมูลและงานใดทำหน้าที่ให้ข้อมูล ขั้นตอนที่สองคือกำหนดโครงสร้างของข้อความ (Message) ที่ใช้ในการส่งและรับของแต่ละการเชื่อมโยง ลักษณะของการสื่อสารอาจแบ่งได้เป็นสี่แบบดังนี้

5.3.2.1 การสื่อสารระยะใกล้ และการสื่อสารแบบครอบคลุม (Local and Global Communication)

การสื่อสารระยะใกล้คือการสื่อสารที่เกิดขึ้นระหว่างงานจำนวนน้อย เช่นการวิเคราะห์เชิงเลข (Numerical Analysis) ในวิธีไฟไนต์ดิฟเฟอเรนซ์ของ Jacobi ซึ่งการหาค่า $X_{i,j}^{(t+1)}$ ทำได้จากสมการ 5.1 โดยใช้ข้อมูลห่าจุดในการคำนวณ

$$X_{i,j}^{(t+1)} = \frac{4X_{i,j}^{(t)} + X_{i-1,j}^{(t)} + X_{i+1,j}^{(t)} + X_{i,j-1}^{(t)} + X_{i,j+1}^{(t)}}{8} \quad (5.1)$$

งานแต่ละงานทำหน้าที่หาค่าของ $X_{i,j}^{(t+1)}$ ที่ตำแหน่งต่างๆกัน และในขณะเดียวกันก็ทำหน้าที่ส่งค่าของตัวเองให้กับงานอื่นด้วยดังรูปที่ 5.7

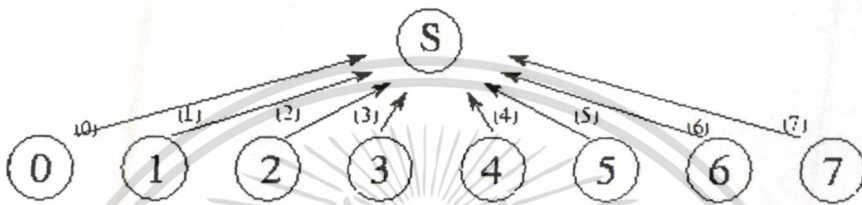


รูปที่ 5.7 ลักษณะการสื่อสารในการคำนวณไฟไนต์ดิฟเฟอเรนซ์ในสองมิติ

การสื่อสารแบบคอบคลุมจะเป็นการสื่อสารในลักษณะที่งานๆหนึ่งต้องติดต่อกับงานหลายๆงานเช่นการคำนวณหาค่าผลรวมดังสมการ 5.2

$$S = \sum_{i=0}^{N-1} X_i \quad (5.2)$$

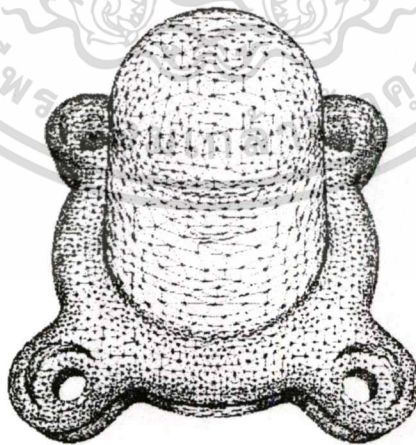
การคำนวณแบบขนานจะเกิดจากงานที่เป็นผู้จัดการ(Manager) จะรับค่า X_i จากงานที่เป็นคนงาน (Worker) เพื่อทำการหาค่า S โดยการสื่อสารจะมีลักษณะดังรูปที่ 5.8



รูปที่ 5.8 การสื่อสารเพื่อรวมค่าจากงานที่เป็นผู้จัดการ

5.3.2.2 การสื่อสารแบบไม่มีโครงสร้างและการสื่อสารแบบเปลี่ยนแปลงได้ (Unstructured and Dynamic Communication)

ในหัวข้อที่ผ่านมาเป็นการสื่อสารแบบคงที่และมีโครงสร้างซึ่งไม่มีการเปลี่ยนแปลงรูปแบบการสื่อสารตามเวลาเช่นการสื่อสารในรูปแบบตารางและแบบโครงสร้างต้นไม้ การสื่อสารแบบไม่มีโครงสร้างและสามารถเปลี่ยนแปลงได้จะใช้ในปัญหาเช่นการวิเคราะห์ไฟไนต์อีลิเมนต์ ซึ่งแสดงดังรูปที่ 5.9



รูปที่ 5.9 ตาราง (Grid) ซึ่งสร้างจากวิธีวิเคราะห์ไฟไนต์อีลิเมนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปร่างของตารางที่สร้างจากวิธีไฟไนต์อีลิเมนต์นี้จะไม่มีรูปแบบที่แน่นอนทำให้ไม่สามารถออกแบบการสื่อสารไว้ก่อนได้ วิธีแบ่งงานอาจทำได้โดยกำหนดค่าให้แก่แต่ละจุดต่อ (Vertex) ของเส้นตารางเป็นงานหนึ่งงาน และการสื่อสารจะทำตามจุดต่อตารางที่แบ่งไว้

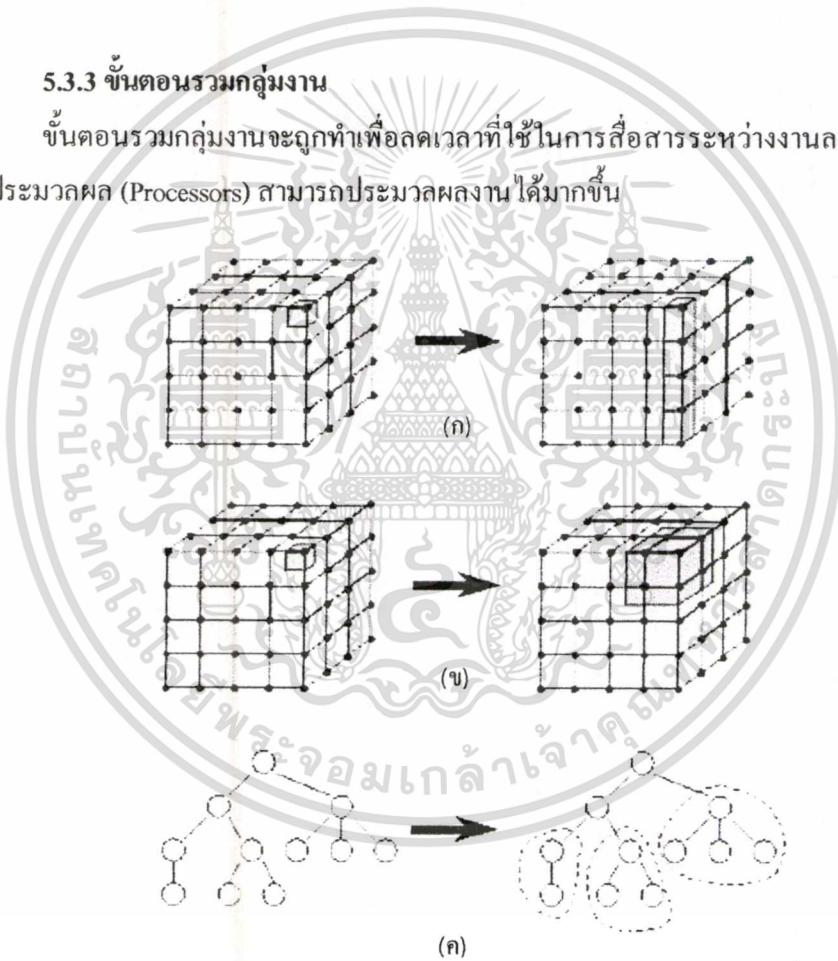
5.3.2.3 การสื่อสารแบบประสานกันและการสื่อสารแบบไม่ประสานกัน

(Synchronous and Asynchronous Communication)

การสื่อสารแบบประสานกัน ผู้ส่งและผู้รับรู้ว่าจะเกิดการสื่อสารกันก่อนจะเกิดการสื่อสารกันจริง ส่วนการสื่อสารแบบไม่ประสานกันผู้ส่งจะไม่ว่าเมื่อใดผู้รับจะต้องการข้อมูล ผู้รับข้อมูลจะต้องทำการร้องขอข้อมูล โดยตรงจากผู้ส่งข้อมูลเมื่อต้องการ

5.3.3 ขั้นตอนรวมกลุ่มงาน

ขั้นตอนรวมกลุ่มงานจะถูกทำเพื่อลดเวลาที่ใช้ในการสื่อสารระหว่างงานลงและทำให้หน่วยประมวลผล (Processors) สามารถประมวลผลงานได้มากขึ้น



รูปที่ 5.10 การรวมกลุ่มงานในการประมวลผลแบบขนานเพื่อลดเวลาในการสื่อสาร

การรวมกลุ่มงานอาจทำได้หลายลักษณะขึ้นอยู่กับประเภทของงานเช่นจากรูปที่ 5.10 ก. เป็นการรวมกลุ่มงานจากงานในตารางสามมิติเป็นงานในตารางสองมิติ ข. เป็นการรวมกลุ่มงานจาก

งานในตารางสามมิติเป็นงานที่มีกลุ่มใหญ่ขึ้น ในสามมิติ ค. เป็นการรวมกลุ่มงานในรูปแบบแผนภูมิ
ต้นไม้เป็นต้น

5.3.4 ขั้นตอนกำหนดงาน

การกำหนดงานจะถูกทำเป็นขั้นตอนสุดท้ายในการออกแบบอัลกอริทึมแบบขนาน ในคอมพิวเตอร์แบบขนาน การกำหนดงานจะต่างจากระบบคอมพิวเตอร์ที่มีหน่วยประมวลผลเดี่ยว (Uniprocessor) โดยระบบปฏิบัติการจะเป็นผู้กำหนดและจัดลำดับงาน (Scheduling) ให้กับหน่วยประมวลผลเพื่อทำการประมวลผล ในคอมพิวเตอร์แบบขนานการกำหนดงานอาจทำโดยอัตโนมัติจากระบบปฏิบัติการแบบขนานหรืออาจกำหนดจากผู้สร้างโปรแกรมโดยตรง การกำหนดงานเป็นขั้นตอนที่ซับซ้อนซึ่งวิธีการกำหนดงานที่ดีจะทำให้ใช้เวลาในการประมวลผลงานน้อยและมีความยืดหยุ่นในการใช้งาน หลักการในการกำหนดงานที่ดีมีสองข้อคือ

- กำหนดงานที่สามารถประมวลผลพร้อมๆกันได้ให้ทำงานในหน่วยประมวลผลต่างกัน
- กำหนดงานที่ต้องมีการสื่อสารกันบ่อยให้อยู่ในหน่วยประมวลผลเดียวกัน

การกำหนดงานอาจไม่มีรูปแบบที่แน่นอนขึ้นอยู่กับประเภทของงานและลักษณะคอมพิวเตอร์แบบขนาน ในระบบที่งานย่อยมีปริมาณงานไม่เท่ากันหรือระบบคอมพิวเตอร์แบบขนานเป็นแบบเนื้อผสม (Heterogeneous) ซึ่งความสามารถในการประมวลผลของแต่ละหน่วยประมวลผลอาจไม่เท่ากันจะทำให้วิธีการกำหนดงานมีความยากและซับซ้อนขึ้นไปอีก

5.4 กลวิธีการโปรแกรมแบบขนาน

การสร้างโปรแกรมแบบลำดับขั้น (Sequential Algorithm) บนเครื่องที่มีเพียงหน่วยประมวลผลเพียงเป็นสิ่งที่ไม่ยุ่งยากหรือสลับซับซ้อนมากมายนัก เพราะลำดับการทำงานของโปรแกรมเป็นไปอย่างตรงไปตรงมากับความต้องการหรือทฤษฎีและวิธีการที่อธิบายไว้แล้ว การทำงานของโปรแกรมใช้เพียงหน่วยประมวลผลเดียว การเข้าใช้งานหน่วยความจำก็สามารถเข้าถึงและใช้งานได้โดยไม่มีปัจจัยอย่างอื่นเข้ามาเกี่ยวข้อง การสื่อสารระหว่างโปรเซสที่ทำงานอยู่ก็มักจะไม่เกิดขึ้นในวิธีการการโปรแกรมแบบนี้ ทำให้การออกแบบโปรแกรมสามารถทำได้ง่าย ซึ่งแตกต่างจากกลวิธีการสร้างโปรแกรมแบบขนานเพราะปัจจัยในเรื่องการใช้งานหน่วยประมวลผล การเข้าใช้งานหน่วยความจำและการสื่อสารระหว่างโปรเซสจะเริ่มเข้ามามีบทบาทในการทำงานของโปรแกรมมากขึ้น อีกทั้งถ้าหากต้องการให้โปรแกรมทำงานได้อย่างมีประสิทธิภาพแล้วต้องคำนึงถึงปัจจัยอื่น ๆ ที่มาเกี่ยวข้องด้วย

ดังนั้นในการสร้างโปรแกรมแบบขนานจึงได้มีผู้นำเสนอเทคนิคและวิธีการต่าง ๆ หลายอย่างในการสร้างโปรแกรมขึ้นมา เพื่อให้การทำงานของโปรแกรมเป็นไปอย่างเหมาะสมและมีประสิทธิภาพ Lester (1993:13) ได้นำเสนอวิธีการต่าง ๆ โดยรวมไว้ทั้งหมด 6 วิธีการดังนี้

- การรวมกลุ่มชุดข้อมูลที่มีความเหมือนกัน (Data Parallelism) ซึ่งเป็นวิธีการที่สามารถใช้งานได้โดยทั่วไป หลักการของการสร้างโปรแกรมแบบนี้ จะทำการรวมเอาข้อมูลอินพุต (Input) ที่มีความเหมือนกันเข้าไว้ด้วยกันเป็นกลุ่ม หลังจากนั้นทำการประมวลผลแยกเป็นกลุ่ม ๆ ไป เมื่อข้อมูลที่อยู่ในกลุ่มเดียวกันมีความสัมพันธ์กันแล้ว ทำให้เกิดความง่ายในการคำนวณมากยิ่งขึ้น

- การแบ่งกลุ่มข้อมูลที่อยู่ใกล้กันได้ด้วยกัน (Data Partitioning) ข้อมูลอินพุตที่อยู่ใกล้กันจะมีความสัมพันธ์อยู่แล้วส่วนหนึ่ง ดังนั้นจึงสามารถที่จะรวมกลุ่มชุดของข้อมูลที่อยู่ใกล้กันไว้ด้วยกันได้ วิธีนี้ทำให้เกิดความสะดวกในการแบ่งกลุ่มของข้อมูลมากยิ่งขึ้น เพราะเพียงพิจารณาข้อมูลที่อยู่ติดกัน แยกเป็นส่วน ๆ เท่านั้น เมื่อทำการแยกข้อมูลได้แล้วก็ทำการประมวลผลข้อมูลกลุ่มต่าง ๆ ไปพร้อมกันตามหลักการของการประมวลผลแบบขนาน

วิธีการนี้เหมาะกับการคำนวณบนระบบประมวลผลแบบขนานแบบหลายเครื่องประมวลผลอย่างเช่นระบบคลัสเตอร์เป็นอย่างมากเพราะแต่ละเครื่องจะมีหน่วยความจำเป็นของตัวเองและเมื่อทำการคำนวณแต่ละเครื่องหรือแต่ละโหนดจะเกี่ยวข้องกับชุดข้อมูลที่ถูกแบ่งออกมาอย่างชัดเจน การเกี่ยวข้องกันกับชุดข้อมูลข้างเคียงมีน้อย ทำให้เกิดการแลกเปลี่ยนข้อมูลระหว่างโหนดน้อยลงตามไปด้วย มีผลทำให้เวลาที่สูญเสียไปในการสื่อสารน้อย เวลารวมในการทำงานก็ลดลงด้วย

- การทำงานแยกกัน โดยอิสระ (Relaxed Algorithm) ถ้าหากแต่ละหน่วยประมวลผลในระบบสามารถทำงานแยกจากกันได้โดยอิสระ โดยที่ไม่ต้องรอข้อมูลผลลัพธ์จากหน่วยประมวลผลหรือโปรเซสข้างเคียงและไม่ต้องสื่อสารหรือตรวจทานความถูกต้องกับหน่วยประมวลผลหรือโปรเซสข้างเคียง และไม่ต้องสื่อสารหรือตรวจทานความถูกต้องกับหน่วยประมวลผลอื่น ๆ ทำให้ไม่ต้องสูญเสียเวลาในการสื่อสารหรือการรอในระหว่างการทำงานเลย หลักการนี้สามารถนำไปประยุกต์ใช้ได้ทั้งระบบแบบหลายหน่วยประมวลผลและหลายเครื่องประมวลผล

- การทำงานพร้อมกันในแต่ละรอบการทำงาน (Synchronous Iteration) ในการคำนวณบางประเภทอาจจะต้องมีการทำงานไปพร้อม ๆ กันในแต่ละรอบการทำงาน วิธีการนี้จะให้แต่ละโปรเซสทำการตรวจสอบการทำงานของโปรเซสข้างเคียงอื่น ๆ ในแต่ละรอบเพื่อให้คำตอบที่ได้มีค่าถูกต้อง การทำงานไปพร้อม ๆ กันจะลดความเร็วในการประมวลผลลงเล็กน้อย และเหมาะกับระบบที่เป็นแบบหลายหน่วยประมวลผลมากกว่าระบบที่เป็นแบบหลายเครื่องประมวลผล เพราะจะต้องมีการสื่อสารระหว่างกันค่อนข้างมาก

- การกระจายการทำงาน วิธีการนี้จะทำการรวมศูนย์งานไว้ที่เดียวเรียกว่า Work Pool เครื่องอื่น ๆ หรือโหนดอื่น ๆ ในระบบจะถูกมองว่าเป็นผู้ทำงานหรือ Worker ซึ่งผู้ทำงานนี้จะต้องเข้ามารับงาน

จาก Work Pool ไปทำงานกระทั่งงานที่อยู่ใน Pool หมดลง การทำงานแบบนี้อาจกล่าวง่าย ๆ ว่าเป็นการทำงานแบบรวมศูนย์เพราะทุก Worker จะทำการติดต่อและร้องของานจากที่เดียวกัน

- การส่งต่อค่าผลลัพธ์ (Pipelined Computation) เทคนิคนี้ได้อาศัยรูปแบบของการส่งต่อค่าข้อมูลผลลัพธ์จากโปรเซสหนึ่ง ไปยังอีกโปรเซสหนึ่งในระบบเป็นลำดับเรื่อยไป ซึ่งการคำนวณงานบางชนิดต้องใช้รูปแบบการคำนวณงานแบบนี้ การส่งต่อค่าผลลัพธ์จะสามารถใช้ได้ดีทั้งระบบแบบหลายหน่วยประมวลผลและระบบแบบหลายเครื่องประมวลผล

เพื่อให้โปรแกรมทำงานอย่างมีประสิทธิภาพสูงสุด เมื่อจะทำการออกแบบโปรแกรมนั้น Lester (1993:15) ได้เสนองัยปัญหาต่าง ๆ ที่ควรคำนึง รวมทั้งปัญหาที่ควรหลีกเลี่ยงไม่ให้เกิดขึ้นในโปรแกรม ปัญหาสำคัญ ๆ มีดังนี้

- ความคับคั่งในการเข้าใช้งานหน่วยความจำ (Memory Contention) ปัญหานี้เกิดจากการเข้าใช้ตัวแปรร่วม (Global variable) ในหน่วยความจำพร้อมกันของหน่วยประมวลผล เมื่อหน่วยประมวลผลหนึ่งใช้งานหน่วยความจำอยู่ หน่วยประมวลผลอื่น ๆ จะไม่สามารถเข้าใช้งานได้ทำให้เกิดการหยุดรอของหน่วยประมวลผลขณะที่กำลังคำนวณ ซึ่งจะทำให้สูญเสียเวลาไปกับการรอนี้ แต่ปัญหานี้มักจะเกิดในระบบหลายหน่วยประมวลผลที่ใช้หน่วยความจำรวมมากกว่าส่วนระบบแบบหลายเครื่องประมวลผลมักไม่เป็นปัญหา

- การมีส่วนของโปรแกรมที่ทำงานแบบลำดับมากเกินไป (Excessive Sequential Code) ซึ่งอาจจะเกิดจากตอนที่ออกแบบโปรแกรมนั้นผู้เขียนโปรแกรมอาจจะสร้างโปรแกรมขึ้นมาจากวิธีการแบบลำดับ และยังใช้วิธีการคิดแบบเดิมในการ โปรแกรมแบบขนานอยู่ทำให้โปรแกรมที่ถูกสร้างขึ้น ไม่ได้ทำงานไปอย่างพร้อม ๆ กันอย่างแท้จริง และประสิทธิภาพของโปรแกรมก็ลดลง

- การสูญเสียเวลาไปในการสร้างโปรเซส (Process Creation Time) ในการทำงานแบบขนานจะต้องมีเวลาส่วนหนึ่งที่สูญเสียไปเพราะการสร้างโปรเซสขึ้นมาใหม่ เพราะฉะนั้นการสร้างโปรเซสควรรวสร้างขึ้นมาเท่าที่จำเป็นเท่านั้น เพราะบางทีการสร้างโปรเซสขึ้นมามากเกินไปก็ไม่ได้ช่วยทำให้การทำงานของโปรแกรมเร็วขึ้นแต่อย่างใด

- การสูญเสียเวลาในการสื่อสาร (Communication Delay) ปัญหานี้จะเกิดขึ้นกับระบบแบบหลายเครื่องประมวลผลค่อนข้างมาก เพราะแต่ละหน่วยประมวลผลจะต้องทำการติดต่อกันผ่านทาง การส่งข้อความข้ามไปมาระหว่างโปรเซสที่ทำงานอยู่ในระบบ ดังนั้นเมื่อทำการออกแบบโปรแกรมแล้วต้องให้มีการสื่อสารระหว่างโปรเซสน้อยที่สุด และขนาดของข้อมูลที่รับและส่งนั้นควรมีขนาดเล็กมากที่สุดด้วยเพื่อให้ เสียเวลาในการสื่อสารน้อยที่สุดนั่นเอง

- การสูญเสียเวลาในการตรวจทางสถานะ (Synchronization Delay) เมื่อมีการออกแบบโปรแกรมให้แต่ละโปรเซสต้องทำการตรวจทางสถานะซึ่งกันและกันแล้ว จะต้องใช้เวลาส่วนหนึ่งสูญเสียไปเพื่อ

การนี้ ดังนั้น ทางที่ดีถ้าหากลดการตรวจทานสถานะไปได้หรือ ไม่มีเลขก็จะทำให้โปรแกรมที่ได้มี ประสิทธิภาพดีขึ้นด้วย

- การกระจายงานไม่สมดุล หรือไม่เหมาะสม (Load Imbalance) เกิดจากการที่มีวิธีการ กระจายการทำงาน ไม่เหมาะสม ทำให้มีหน่วยประมวลผลบางส่วนเท่านั้นที่มีโอกาสได้ทำงาน ส่วน หน่วยประมวลผลอื่นอาจจะอยู่ในสถานะรอเพียงอย่างเดียว ดังนั้นเพื่อแก้ปัญหานี้จะต้องมีวิธีการที่ เหมาะสมในการกระจายงานไปยังหน่วยประมวลผลในระบบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การทดลองและผลการทดลอง

6.1 บทนำ

ในบทนี้จะกล่าวถึงการทดลองและผลการทดลองของการสร้างภาพตัดขวางอัลกอริธึมทางพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบคลัสเตอร์ โดยแบ่งการทดลองออกเป็นสองส่วน ส่วนแรกวัดเวลาที่ใช้ในการทดลองสร้างภาพตัดขวางบนระบบคลัสเตอร์จากแบบจำลองทางคอมพิวเตอร์ และในส่วนที่สองจะเป็นการสร้างภาพตัดขวางจากข้อมูลของภาพถ่ายเอ็กซเรย์จริง

6.2 ระบบคลัสเตอร์ที่ใช้ในการทดลอง

ระบบคลัสเตอร์เป็นระบบที่มีโครงสร้างแบบหลายหน่วยประมวลผล และมีหน่วยความจำกระจาย(Distributed Memory)

- โครงสร้างของระบบทางฮาร์ดแวร์

โครงสร้างทางด้านฮาร์ดแวร์ของระบบที่ใช้ในการคำนวณนั้น เป็นคลัสเตอร์แบบเนื้อเดียว (Homogenous Cluster) คือจะมีหน่วยประมวลผล ปริมาณหน่วยความจำ และส่วนประกอบอื่น ๆ เหมือนกัน ระบบที่สร้างขึ้นมานี้ประกอบด้วยเครื่อง PC X86 โดยใช้เครื่องคอมพิวเตอร์รุ่น HPx2100 จำนวน 5 เครื่อง ชนิดที่มีสองหน่วยประมวลผลในหนึ่งเครื่อง หน่วยประมวลผล (CPU) เป็น Intel Xeon 2.4 GHz หน่วยความจำชนิด ECC จำนวน 1 GB เชื่อมต่อกันด้วยเครือข่ายภายในความเร็ว 1 Gbit/sec

- โครงสร้างของระบบทางซอฟต์แวร์

ระบบคลัสเตอร์ที่ใช้ในการทดลองนี้ใช้ระบบปฏิบัติการ Suse Linux 8.0 และใช้ gcc เป็นตัวแปลภาษาและได้ติดตั้งมาพร้อมกับตัว Linux อยู่แล้ว ในงานวิจัยนี้แนะนำให้ใช้ระบบคลัสเตอร์ที่มีระบบปฏิบัติการเป็นลินุกซ์เนื่องจาก

- เป็นระบบปฏิบัติการที่แจกให้ฟรี
- สามารถทำเป็นระบบคลัสเตอร์ได้จากซอฟต์แวร์และไลบรารีที่มีแจกให้ฟรีบนระบบอินเทอร์เน็ต โดยที่ไม่ต้องเสียค่าใช้จ่ายในส่วนซอฟต์แวร์เพิ่มเติมในการสร้างระบบคลัสเตอร์
- มีประสิทธิภาพและความแน่นอนสูง

- มีการพัฒนาอย่างต่อเนื่อง
- มีความยืดหยุ่นในการปรับเปลี่ยนระบบสูง
- โปรแกรมที่พัฒนาบนระบบลินุกซ์สามารถย้ายไปทำงานบนระบบยูนิกซ์ตระกูลอื่นได้ง่าย
- ระบบคลัสเตอร์ส่วนมากใช้ลินุกซ์เป็นระบบปฏิบัติการ ซึ่งเมื่อมีปัญหาสามารถขอคำปรึกษาจากผู้ดูแลระบบเหล่านั้นได้
- โปรแกรมที่เกี่ยวกับระบบคลัสเตอร์ส่วนมากพัฒนาบนระบบลินุกซ์ สามารถแก้ไขแก่น (Kemel) ของระบบปฏิบัติการเพื่อปรับปรุงโครงสร้างของระบบปฏิบัติการได้ และสามารถพัฒนาระบบคลัสเตอร์โดยเพิ่มระบบการทำงานเข้าไปในแก่นของระบบปฏิบัติการโดยตรงได้

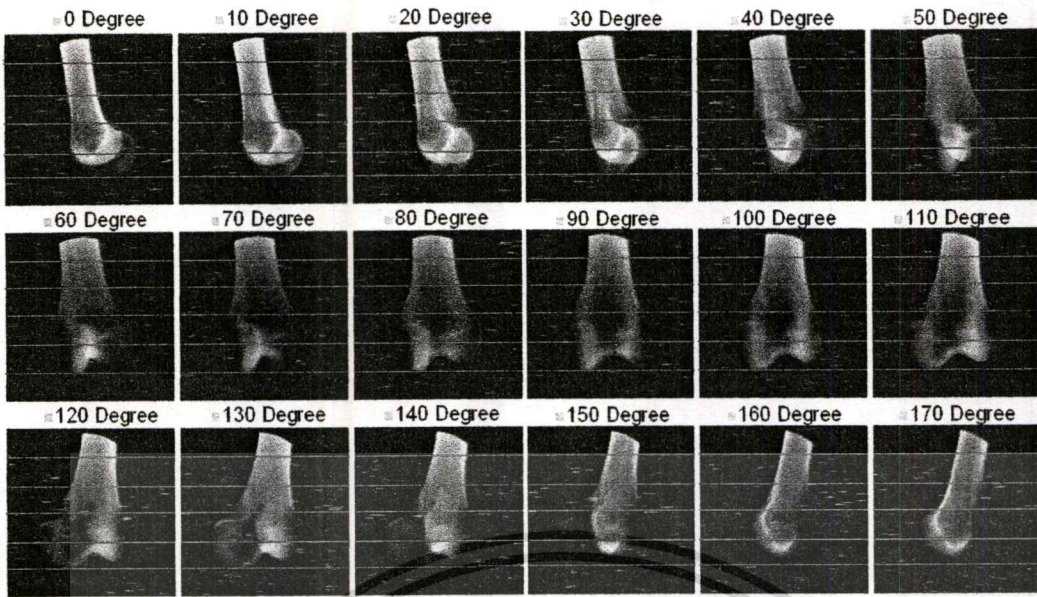
และในส่วนของซอฟต์แวร์ที่ทำหน้าที่เป็นตัวสร้างสถานะแวดล้อมแบบขนานนั้น ในวิทยานิพนธ์ฉบับนี้เลือกใช้ MPICH ซึ่งเป็นคลังชุดคำสั่งที่ถูกพัฒนาขึ้นตามมาตรฐาน MPI ซึ่งมีชุดคำสั่งต่าง ๆ ให้เลือกใช้งานอย่างเพียงพอ อีกทั้งยังสามารถใช้งานได้หลากหลายระบบปฏิบัติการ

6.3 การออกแบบโปรแกรมการสร้างภาพตัดขวาง

ในวิทยานิพนธ์ฉบับนี้เป็นการพัฒนาการสร้างภาพตัดขวางโดยวิธีอัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบคลัสเตอร์เพื่อเพิ่มความเร็วในการสร้างภาพตัดขวางได้รวดเร็วยิ่งขึ้น ในวิธีการที่นำเสนอใช้หลักการที่ว่ากระบวนการสร้างภาพตัดขวางสามารถแบ่งย่อยข้อมูลในการประมวลผลได้แม้ข้อมูลต่าง ๆ ที่ใช้ในการประมวลผลนั้นจะขึ้นต่อกันในแต่ละส่วนของการทำงาน และ ไม่มีความจำเป็นต้องมีการสื่อสารระหว่างงานย่อย (Embarrassingly Parallel Computations)

6.3.1 การแบ่งงานในการสร้างภาพตัดขวาง

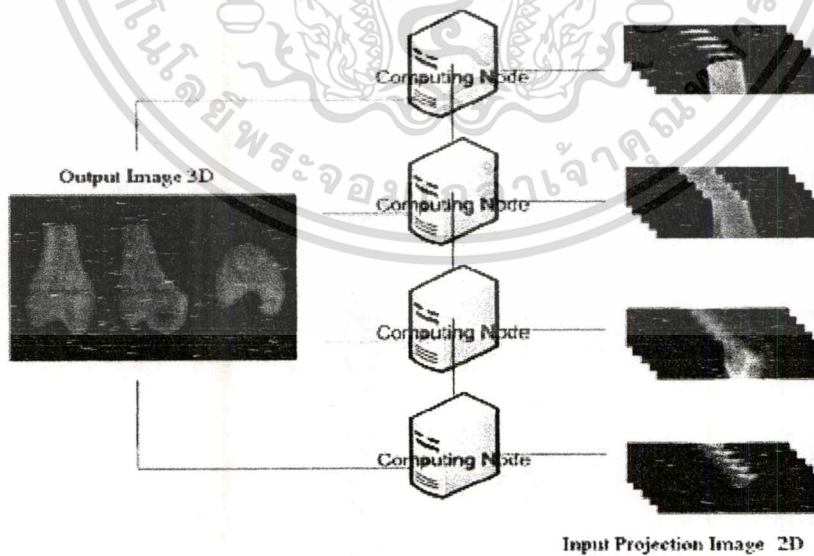
เนื่องจากระบบคอมพิวเตอร์แบบคลัสเตอร์ในการทดลองเป็นแบบคลัสเตอร์แบบเนื้อเดียว และมีหน่วยความจำเป็นแบบรวมศูนย์กลาง ในวิทยานิพนธ์ฉบับนี้ได้ใช้การแบ่งงานที่เป็นการแบ่งข้อมูลออกเป็นส่วนย่อย ๆ ตามความสูงของโปรเจกชันคิงรูปที่ 6.1 และความสูงของกอนปริมาตร ซึ่งจัดสรรให้แต่ละโหนดมีข้อมูลที่รับผิดชอบต่างกัน และมีข้อมูลบัฟเฟอร์กลางในการตรวจสอบการเปลี่ยนแปลงเมื่อโหนดต่างๆประมวลผลดังนั้นการทำงานของแต่ละโหนด(Node) จึงสามารถมีเวลาในการทำงานใกล้เคียงกัน เนื่องจากความเร็วของหน่วยประมวลผลมีความเร็วใกล้เคียงกัน ดังนั้นจึงสามารถแบ่งงานให้เครื่องละเท่าๆกันได้ ทำให้ระบบทำงานไปพร้อมๆกัน



รูปที่ 6.1 การแบ่งข้อมูลอินพุตตามความสูงของโปรเจกชัน

6.3.2 การรวมกลุ่มงาน

การรวมกลุ่มของงานในการการสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวย เนื่องจากการสร้างภาพตัดขวางด้วยวิธีนี้เป็นกระบวนการแบบวิธีการทำซ้ำ และข้อมูลในแต่ละขั้นต่อนั้นขึ้นต่อกัน ดังนั้นเมื่อแบ่งข้อมูลตามความรับผิดชอบของแต่ละหน่วยประมวลผลแล้ว การกำหนดงาน (Task) จะเป็นการส่งคำสั่งและส่งกระจายออกไปทุกโหนด เพื่อให้ทุกหน่วยประมวลผลได้ทราบว่าในขณะนี้ แต่ละโหนดกำลังทำงานอะไรอยู่ เพื่อให้ทุกโหนดทำข้อมูลที่รับผิดชอบอยู่จนเสร็จสิ้น ก่อนจะทำในคำสั่งต่อไป



รูปที่ 6.2 ข้อมูลที่ทำการแบ่งจะถูกส่งไปยังเครื่องเพื่อประมวลผล แต่ละ โหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.3 การสื่อสารในระบบ

เนื่องจากงานที่ถูกแบ่งในการสร้างภาพตัดขวางไม่จำเป็นต้องมีการสื่อสารระหว่างกลุ่มงานด้วยกัน การออกแบบการสื่อสารจึงไม่มีความยุ่งยาก ซึ่งในวิทยานิพนธ์ฉบับนี้ใช้หลักการของศูนย์รวมงาน (Work pool) และใช้การสมดุลงาน (Load Balancing) เป็นแบบศูนย์กลางคำสั่ง (Centralize Dynamic load balancing) ซึ่งระบบนี้ประกอบไปด้วยคอมพิวเตอร์หลาย ๆ เครื่อง เชื่อมต่อกันผ่านทางเครือข่ายความเร็วสูง และมีคอมพิวเตอร์หนึ่งทำหน้าที่เป็นผู้กระจายงานเรียกว่า Manager และทำงานไปพร้อมกับเครื่องอื่น ๆ ซึ่งเรียกว่า Worker และ นั่นคือ เครื่อง Manager จะเป็นผู้กำหนดงานให้กับเครื่อง Worker ทุกๆตัว และเมื่อแต่ละเครื่อง Worker ทำงานเสร็จแล้ว Worker นั้นจะการร้องขอเพื่อของงานใหม่จากเครื่อง Manager ซึ่งลำดับของคำสั่งที่ใช้ในนั้นจะมีลักษณะเป็นการวนรอบของการสั่งงาน ซึ่งเรียกการแบ่งงานแบบนี้ว่า ราวด์-โรบิน (Round robin algorithm) ซึ่งระบบราวด์-โรบินนี้ จะทำงานเหมือนกับระบบปฏิบัติการ (Operating System) ในเครื่องประมวลผลเดี่ยว ตัวอย่างของอัลกอริทึมแบบราวด์-โรบินแสดงประกอบรูปที่ 6.3 คือ หากมีโหนดทั้งหมด 6 โหนด แต่จำนวนงานมี 20 ชิ้น ดังนั้น โหนดที่ 0 จะรับผิดชอบ 0, 6, 12 และ 18; โหนดที่ 1 จะรับผิดชอบ 1, 7, 13, 19; โหนดที่ 2 จะรับผิดชอบ 2, 8, 14; โหนดที่ 3 จะรับผิดชอบ 3, 9, 15; โหนดที่ 4 จะรับผิดชอบ 4, 10, 16; โหนดที่ 5 จะรับผิดชอบ 5, 11, 17;



รูปที่ 6.3 การทำงานของระบบศูนย์รวมงาน (Work pool)

6.4 การวัดประสิทธิภาพของการสร้างภาพตัดขวางบนระบบคลัสเตอร์

การวัดเวลาที่ใช้ในการสร้างภาพเชิงปริมาตรในวิทยานิพนธ์ฉบับนี้จะทำการวัดเวลาที่ใช้ในการสร้างภาพจริงๆ เท่านั้นซึ่งจะไม่รวมไปถึงเวลาที่ใช้ในการอ่านภาพมาเก็บไว้ในหน่วยความจำและการตั้งค่าระบบ (Initialization) ของ MPI ซึ่งกระบวนการทั้งสองจะถูกทำครั้งเดียวในการสร้างภาพ เมื่อต้องการสร้างภาพในมุมมองใหม่จะไม่ต้องทำการกระบวนการทั้งสองอีก แนวโน้มของเวลาที่ใช้ในการสร้างภาพด้วยวิธีการที่นำเสนอในวิทยานิพนธ์นี้เป็นดังสมการ 6.1

$$T = \frac{100}{p} \quad (6.1)$$

เมื่อ T คือเปอร์เซ็นต์ของเวลาที่ใช้ในการสร้างภาพ

p คือจำนวนเครื่องที่ใช้ประมวลผลในระบบคลัสเตอร์

เวลาที่ใช้ในการประมวลผลจะมีลักษณะตามสมการ 6.2

$$T_p = t_{comp} + t_{comm} \quad (6.2)$$

เมื่อ T_p คือเวลาที่ใช้ในการสร้างภาพ

t_{comp} คือเวลาที่ใช้ในการประมวลผล

t_{comm} คือเวลาที่ใช้ในการสื่อสารระหว่างโพรเซส

ระบบคลัสเตอร์เป็นระบบที่การสื่อสารระหว่างโพรเซสจะทำผ่านระบบเครือข่ายซึ่งความเร็วของการสื่อสารจะขึ้นอยู่กับชนิดและคุณภาพของอุปกรณ์ระบบเครือข่าย อุปกรณ์เครือข่ายที่ดีควรจะทำให้ t_{comm} มีค่าน้อยๆ เพราะเมื่อเกิดการสื่อสารระหว่างหลายๆ โพรเซสในระบบคลัสเตอร์ t_{comm} จะมีค่ามากขึ้นตามจำนวนของการสื่อสารเหล่านั้นด้วยซึ่งจะส่งผลให้เวลาที่ใช้ในการสร้างภาพมากขึ้น กราฟในอุดมคติของเวลาที่ใช้ในการสร้างภาพเทียบกับจำนวนเครื่องที่ใช้ในการประมวลผลบนระบบคลัสเตอร์ซึ่งให้ $t_{comm} = 0$ แสดงดังรูปที่ 6.4

นอกจากการวัดเวลาที่ใช้ในการประมวลผลแล้วยังสามารถวัดประสิทธิภาพได้จากค่าอัตราการเพิ่มของความเร็ว (Speedup) ของการสร้างภาพได้จากสมการ 6.3

$$S_p = \frac{T_1}{T_p} \quad (6.3)$$

เมื่อ S_p คืออัตราการเพิ่มของความเร็วที่ใช้ในการสร้างภาพ

T_1 เวลาที่ใช้ในการสร้างภาพบนเครื่องหนึ่งเครื่องด้วยอัลกอริทึมแบบลำดับ

T_p เวลาที่ใช้ในการสร้างภาพบนเครื่องจำนวน p เครื่องด้วยอัลกอริทึมแบบขนานบนระบบคลัสเตอร์

กราฟในอุดมคติของอัตราการเพิ่มความเร็วในการสร้างภาพเทียบกับจำนวนเครื่องที่ใช้ในการประมวลผลบนระบบคลัสเตอร์ แสดงดังรูปที่ 6.6 และค่าสุดท้ายสำหรับการวัดประสิทธิภาพคือค่าประสิทธิภาพ (Efficiency) ของการสร้างภาพได้ดังสมการ 6.4

$$E_p = \frac{S_p}{p} = \frac{T_1}{pT_p} = 1 \quad (6.4)$$

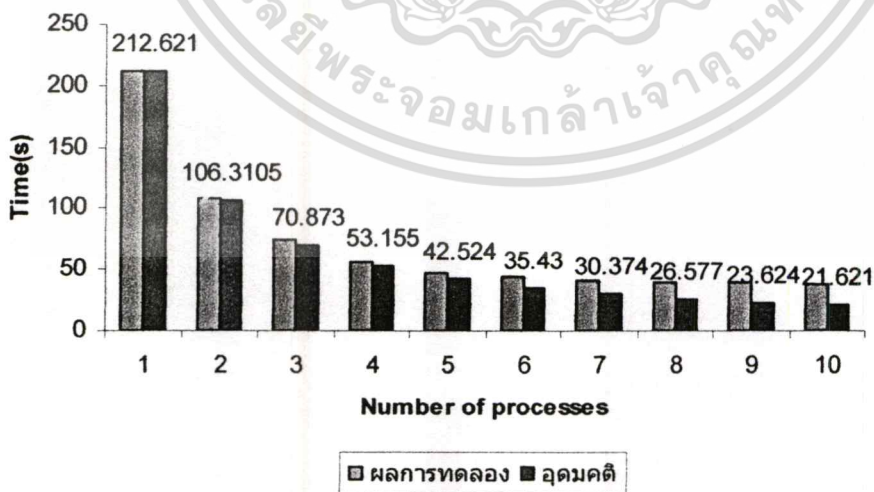
กราฟในอุดมคติของประสิทธิภาพในการสร้างภาพเทียบกับจำนวนเครื่องที่ใช้ในการประมวลผลบนระบบคลัสเตอร์ แสดงดังรูปที่ 6.6

6.5 การทดลองและผลการทดลอง

ในวิทยานิพนธ์นี้ได้ทำการทดลองโดยมุ่งความสนใจไปที่การสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวย โดยจะมีการทดลองเปรียบเทียบทางด้านเวลาและความถูกต้องระหว่าง การสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบประมวลผลเกี่ยวกับการสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบคลัสเตอร์ และสุดท้ายคือแสดงผลในการสร้างภาพตัดขวางโดยใช้ภาพเอ็กซ์เรย์

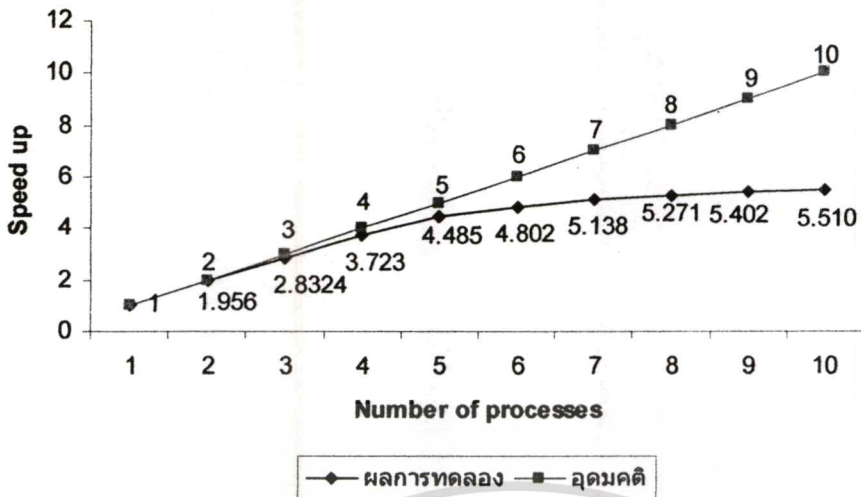
6.5.1 ผลการทดลองเปรียบเทียบกับอุดมคติ

การทดลองในหัวข้อนี้จะเป็นการวัดประสิทธิภาพของระบบที่ใช้ในการทดลองเปรียบเทียบกับระบบที่เป็นอุดมคติโดยข้อมูลเริ่มต้นคือ ข้อมูลโปรเจกชัน ของสร้างภาพตัดขวางจากแบบจำลองทางคอมพิวเตอร์ที่มีขนาดของโปรเจกชันเท่ากับ 256x256 พิกเซล จำนวน 40 มุม ในการทดลองจะได้ผลลัพธ์เป็นภาพตัดขวางขนาดปริมาตร 256x256x256 voxels โดยเปรียบเทียบเวลาระหว่างการทำงานของโปรแกรมบนระบบคลัสเตอร์และการทำงานของโปรแกรมบนระบบคลัสเตอร์อุดมคติ

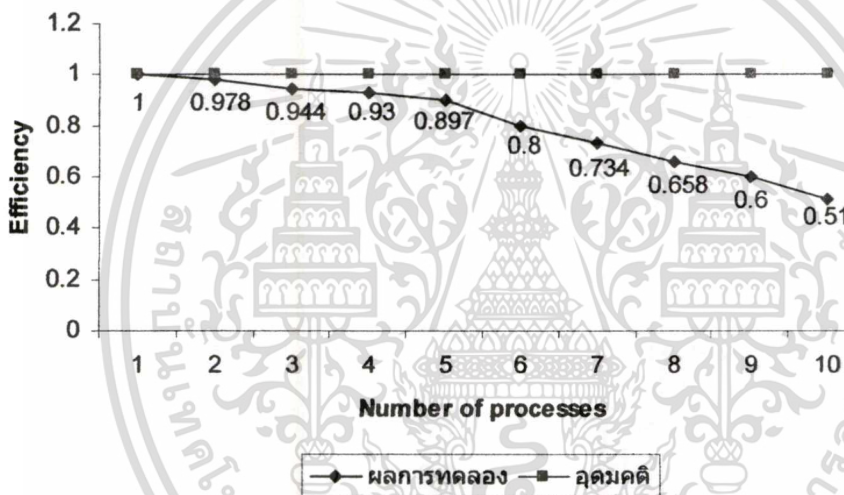


รูปที่ 6.4 แผนภูมิแสดงเวลาที่ลดลงของการทดลองเปรียบเทียบกับระบบอุดมคติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.5 แผนภูมิแสดงอัตราการเพิ่มความเร็วของการทดลองเปรียบเทียบกับระบบอุดมคติ



รูปที่ 6.6 แผนภูมิแสดงอัตราการเพิ่มความเร็วของการทดลองเปรียบเทียบกับระบบอุดมคติ

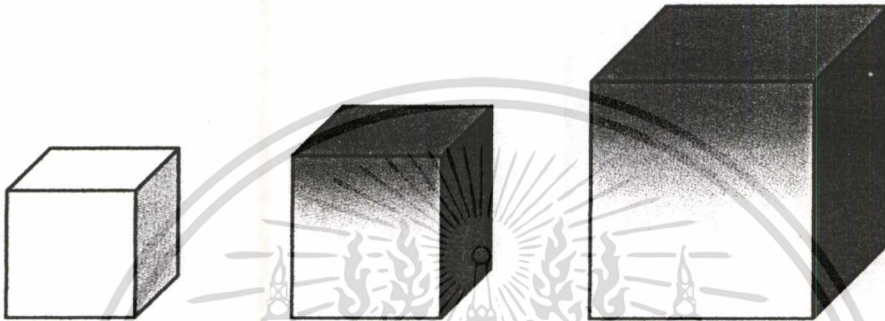
แผนภูมิที่ 6.5-6.7 เวลาที่จำนวนโปรเซส (Number of processes) หมายถึง เนื่องจากวิทยานิพนธ์นี้ใช้การแบ่งกลุ่มของข้อมูล ดังนั้นจำนวนโปรเซสคือจำนวนของกลุ่มข้อมูลที่ใช่ประมวลผลบนระบบคลัสเตอร์

จากผลการทดลอง จะเห็นได้ว่าเมื่อจำนวนของโปรเซสเพิ่มขึ้นความเร็วในการสร้างภาพตัดขวางจะเพิ่มขึ้น จนถึงระดับหนึ่งความเร็วของระบบจะเริ่มคงที่ ซึ่งเป็นผลเนื่องมาจากเวลาที่ใช้ในการติดต่อสื่อสารกันระหว่างโปรเซสเพิ่มมากขึ้น ซึ่งจากแผนภูมิจะเห็นได้ว่าแนวโน้มเมื่อแบ่งโปรเซสออกมามากกว่าจำนวนโหนด จะทำให้ความเร็วที่เพิ่มขึ้นเริ่มคงที่ ซึ่งเป็นไปตามสมการที่ 6.2 ดังนั้น ในการทดลองต่อไปจึงขอแสดงการแบ่งจำนวนโปรเซสเท่ากับจำนวนโหนด

6.5.2 การทดลองเปรียบเทียบเมื่อขนาดก้อนปริมาตรต่างกัน

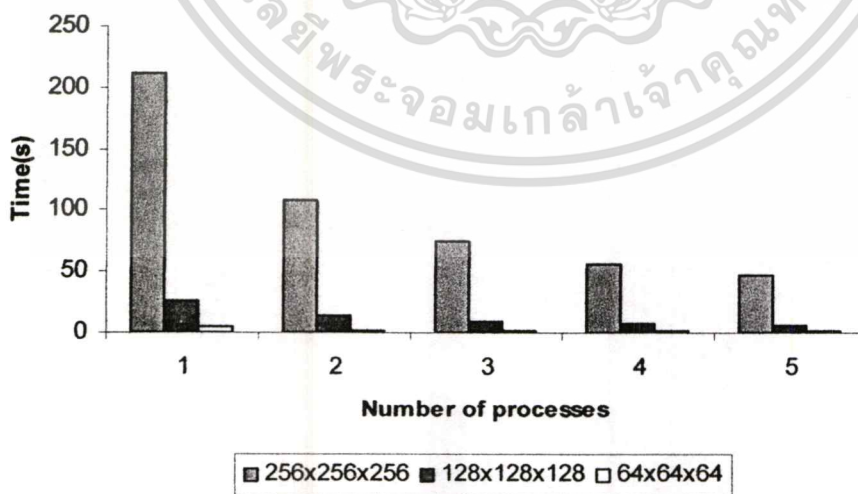
เงื่อนไขอีกอย่างหนึ่งที่น่าสนใจในการทดลองคือ ขนาดของก้อนปริมาตรที่จะทำการสร้างภาพกลับ เพื่อจะทดสอบผลกระทบที่มีต่อความเร็วในการทำงานของระบบ โดยมีเงื่อนไขการทดลองดังนี้

ตัวแปรต้น ขนาดของก้อนปริมาตรที่จะทำการสร้างภาพกลับ
 ตัวแปรตาม เวลาที่ใช้ในการประมวลผล
 ตัวแปรควบคุม อิทธิพลเรชั่น = 3, ขนาดโปรเจ็คชัน = 40 โปรเจ็คชัน
 ขนาดของปริมาตรที่แบ่งจะมีทั้งหมด 4 ขนาด ดังตาราง



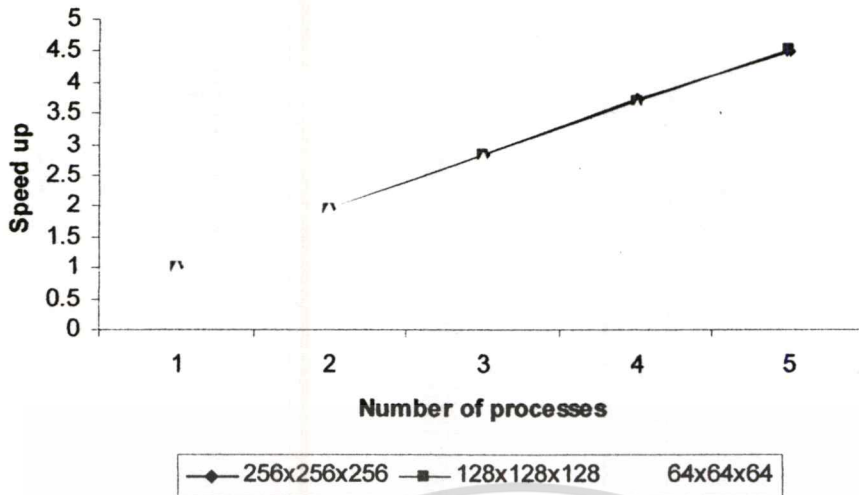
ขนาด (Voxels)	Kbytes
64	256
128	2048
256	16384

รูปที่ 6.7 ขนาดของปริมาตรที่แบ่งในการทดลอง

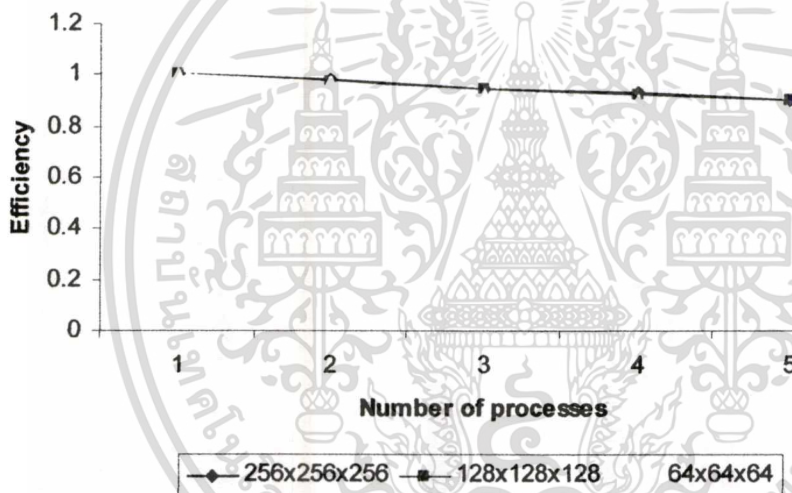


รูปที่ 6.8 แผนภูมิแสดงเวลาของระบบที่ใช้ในการสร้างภาพเมื่อปริมาตรต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.9 แผนภูมิแสดงความเร็วของระบบที่ใช้ในการสร้างภาพเมื่อปริมาตรต่างกัน



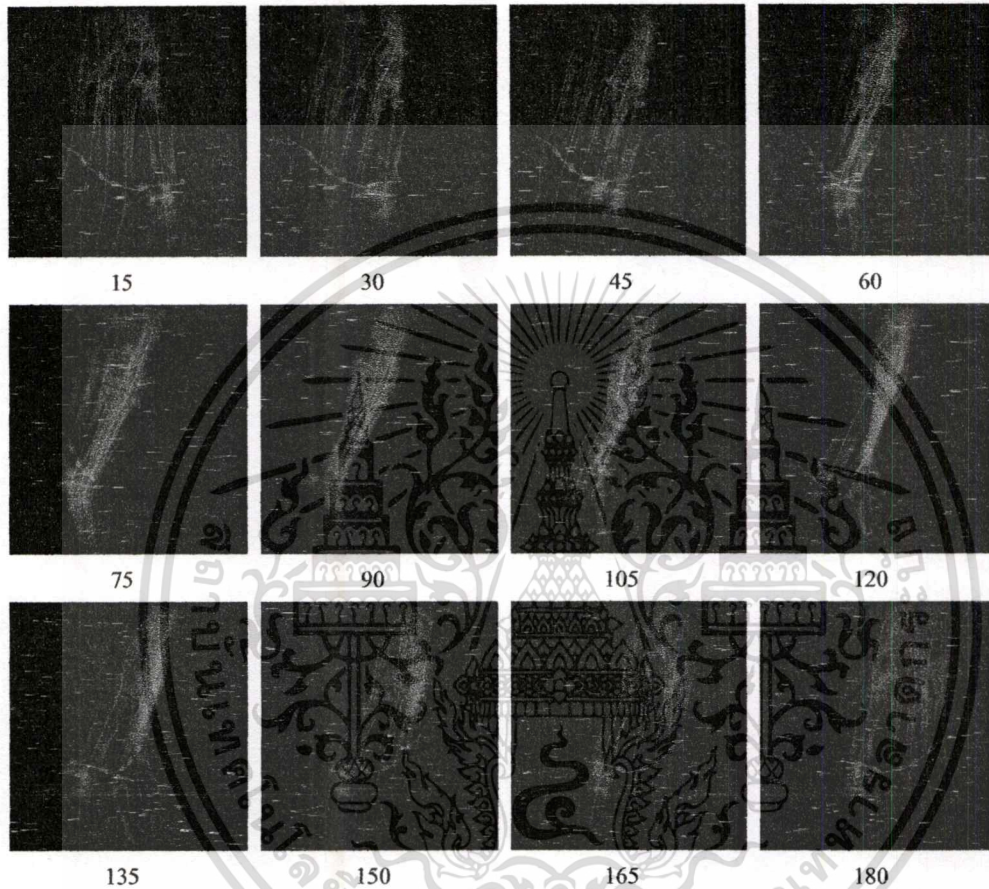
รูปที่ 6.10 แผนภูมิแสดงค่าประสิทธิภาพของระบบที่ใช้ในการสร้างภาพเมื่อปริมาตรต่างกัน

แผนภูมิที่ 6.8-6.10 เวลาที่จำนวนโปรเซส (Number of processes) หมายถึง เนื่องจากวิทยานิพนธ์นี้ใช้การแบ่งกลุ่มของข้อมูล ดังนั้นจำนวนโปรเซสคือจำนวนของกลุ่มข้อมูลที่ใช้ประมวลผลบนระบบคลัสเตอร์

ซึ่งจากผลการทดลองพบว่า ผลกระทบทางขนาดของปริมาตรนั้น ในด้านเวลา เมื่อก่อนปริมาตรขนาดเล็กลง การประมวลผลก็จะใช้เวลาน้อย ส่วนผลด้านความเร็วและค่าประสิทธิภาพ โดยการพิจารณาจากแผนภูมิด้านความเร็ว และค่าประสิทธิภาพ จะเห็นได้ว่ากราฟมีค่าใกล้เคียงกันมาก

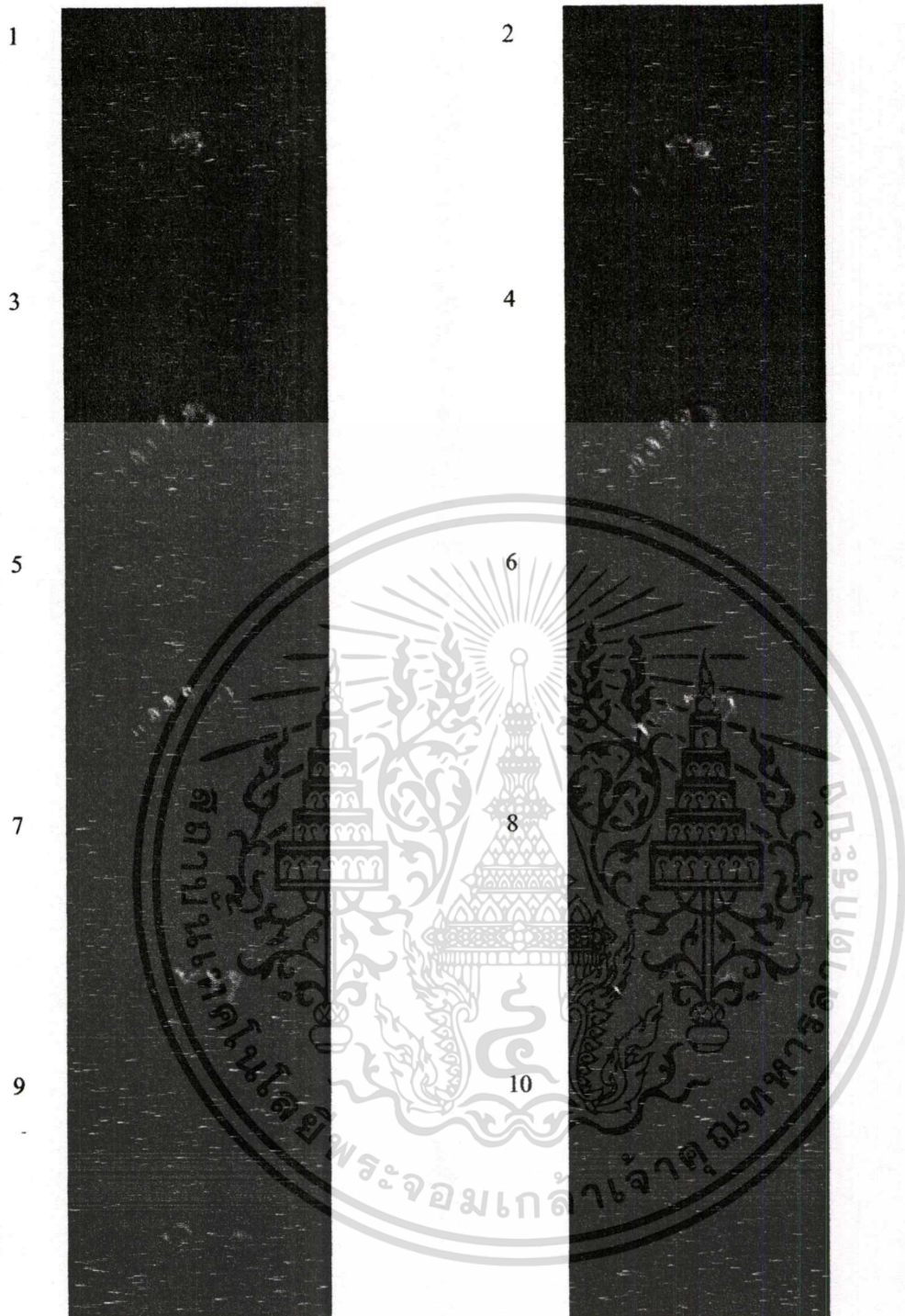
6.5.3 การสร้างภาพตัดขวางจากภาพถ่ายเอ็กซเรย์

ในการสร้างภาพตัดขวางจากภาพถ่ายเอ็กซเรย์นั้นงานวิจัยนี้ได้ทำการทดลองกับชุดภาพถ่ายเอ็กซเรย์ขนาด 256x256 พิกเซล ของเท้าขวา ซึ่งข้อมูลเริ่มต้นเป็นภาพเอ็กซเรย์จำนวน 60 มุม และทำการสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบคลัสเตอร์ ในการประมวลผลจำนวน 5 อิทเทอเรชัน



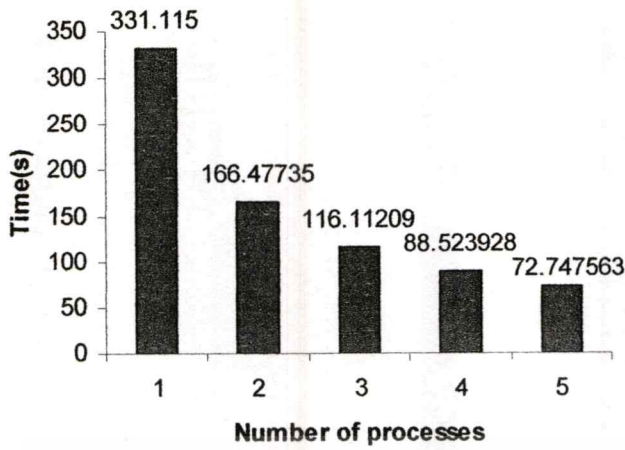
รูปที่ 6.11 ตัวอย่างของข้อมูลอินพุตซึ่งเป็นรูปเอ็กซเรย์ รูปเท้าขวาที่มุมต่าง ๆ กัน

ในการทดลองได้ทำการเปรียบเทียบการสร้างภาพตัดขวางจากภาพถ่ายเอ็กซเรย์ในรูปที่ 6.18 ถึงรูปที่ 6.20 โดยวิธีฟิลเตอร์เบ็คโปรเจกชัน (FBP) และวิธี SART 5 iteration ที่ประมวลผลบนระบบคลัสเตอร์ หลังจากได้ภาพผลลัพธ์ซึ่งเป็นข้อมูลเมตริกซ์เชิงปริมาตร แล้วจึงนำข้อมูลเมตริกซ์เชิงปริมาตรที่ได้มาสร้างเป็นภาพ 3 มิติ ด้วยการสร้างภาพเชิงพื้นผิวด้วยวิธีมาร์ชชิงคิวบ์ และการสร้างภาพเชิงปริมาตร

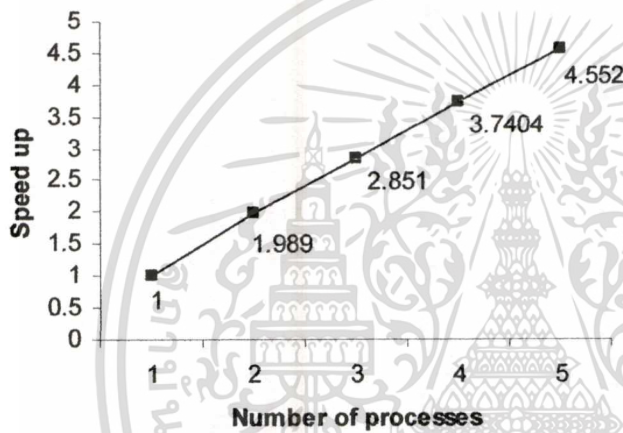


รูปที่ 6.12 ภาพผลลัพธ์ซึ่งเป็นภาพตัดขวางรูปเท้าขวาที่สไลด์ต่างกันการสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบคลัสเตอร์

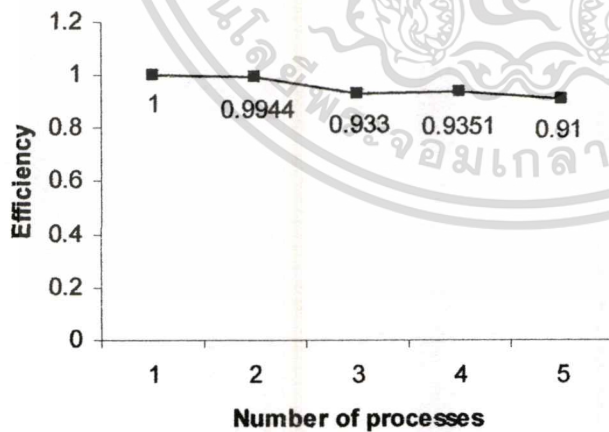
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.13 แผนภูมิแสดงเวลาที่ลดลงของการทดลองการสร้างภาพตัดขวางด้วยภาพถ่ายเอ็กซเรย์



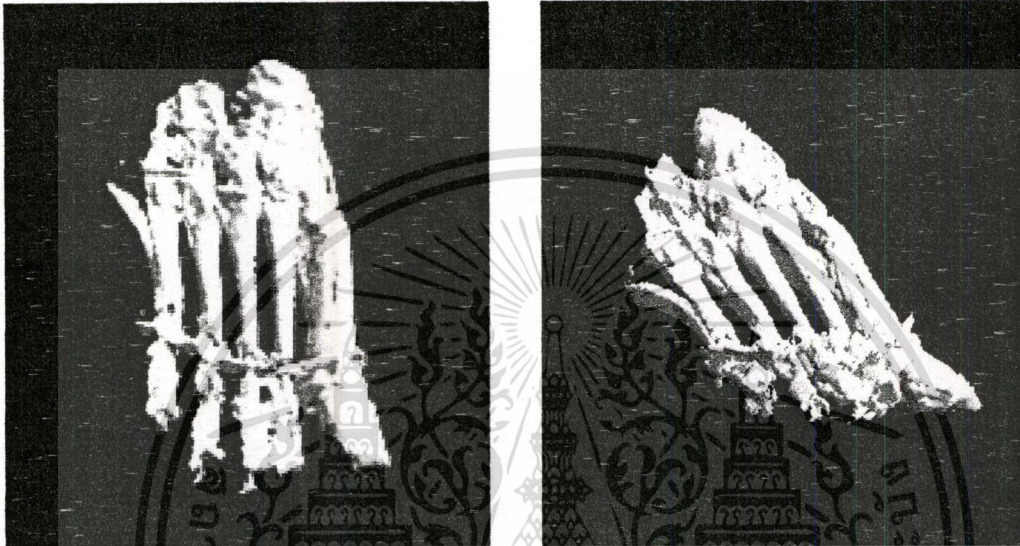
รูปที่ 6.14 แผนภูมิแสดงอัตราการเพิ่มความเร็วของการสร้างภาพตัดขวางด้วยภาพถ่ายเอ็กซเรย์



รูปที่ 6.15 แผนภูมิแสดงค่าประสิทธิภาพของระบบในการสร้างภาพตัดขวางด้วยภาพถ่ายเอ็กซเรย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการทดลองจะเห็นได้ว่าในรูปที่ 6.12 นั้นแสดงให้เห็นถึงภาพตัดขวางของรูปเท้าขวา ซึ่งมีลักษณะของภาพตัดขวางค่อนข้างสมบูรณ์และมีสัญญาณรบกวนน้อย ซึ่งตรงตามคุณสมบัติของการสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิต ส่วนในเรื่องของเวลานั้น ได้นำมาใช้ประมวลผลบนระบบคลัสเตอร์ ซึ่งเวลาในการประมวลผลลดลง 78 เปอร์เซ็นต์เมื่อเปรียบเทียบกับเครื่องประมวลผลเดี่ยว และมีความเร็วเพิ่มขึ้นประมาณ 4.5 เท่าและมีค่าประสิทธิภาพลดลงเนื่องจากจำนวนโปรเซสเพิ่มขึ้นจึงมีการสื่อสารมากขึ้น

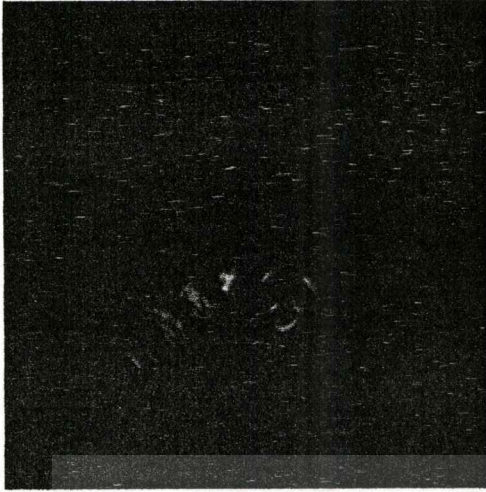


รูปที่ 6.16 ข้อมูลผลลัพธ์การสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบคลัสเตอร์ โดยแสดงผลด้วยการสร้างภาพเชิงพื้นผิวด้วยวิธีมาร์ชชิงคิวบ์

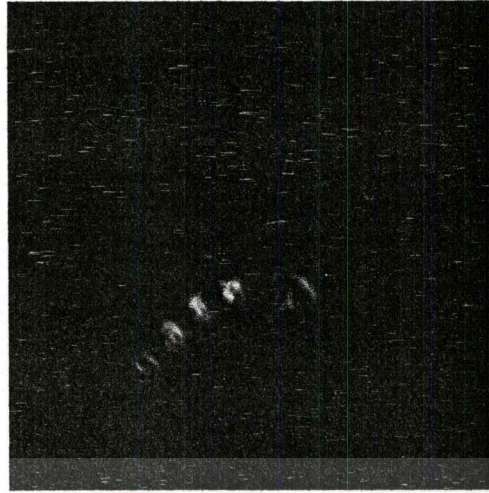


รูปที่ 6.17 ข้อมูลผลลัพธ์การสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับวิเคราะห์ทางเดินแสงแบบกรวยบนระบบคลัสเตอร์ โดยแสดงผลด้วยการสร้างภาพเชิงปริมาตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก)

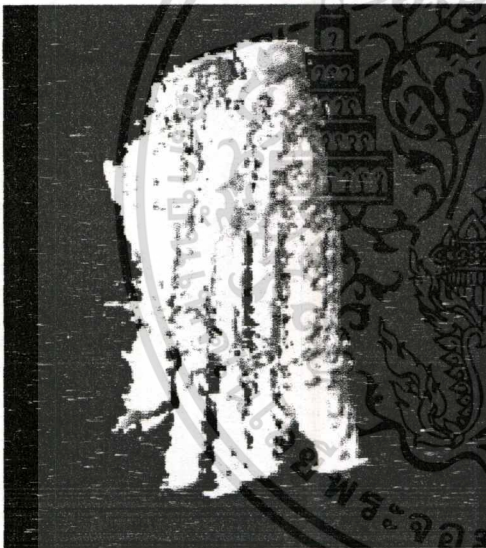


(ข)

รูปที่ 6.18 ข้อมูลภาพผลลัพธ์ของภาพตัดขวางของรูปเท้าขวาสไลด์ที่ 124 เปรียบเทียบกัน

(ก) ภาพผลลัพธ์จากการสร้างภาพตัดขวางโดยใช้วิธีฟิลเตอร์แบบคโปรเจกชัน

(ข) ภาพผลลัพธ์จากการสร้างภาพตัดขวางโดย SART บนระบบคลัสเตอร์



(ก)



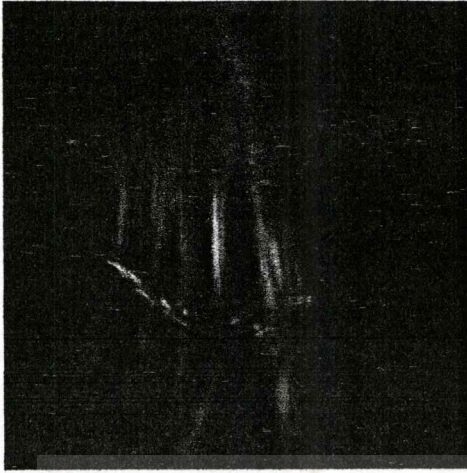
(ข)

รูปที่ 6.19 ข้อมูลภาพผลลัพธ์เมื่อประยุกต์ใช้การสร้างภาพเชิงพื้นผิวของรูปเท้าขวาศด้วยวิธีมาร์ชชิง -

คิวบ์เปรียบเทียบกัน

(ก) ภาพผลลัพธ์จากการสร้างภาพตัดขวางโดยใช้วิธีฟิลเตอร์แบบคโปรเจกชัน

(ข) ภาพผลลัพธ์จากการสร้างภาพตัดขวางโดย SART บนระบบคลัสเตอร์



(ก)



(ข)

รูปที่ 6.20 ข้อมูลภาพผลลัพธ์เมื่อประยุกต์ใช้การสร้างภาพเชิงปริมาตรของรูปเท้าวาเปรียบเทียบกัน

(ก) ภาพผลลัพธ์จากการสร้างภาพตัดขวางโดยใช้วิธีฟิลเตอร์เบ็ค โปรเจคชัน

(ข) ภาพผลลัพธ์จากการสร้างภาพตัดขวางโดย SART บนระบบคลัสเตอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

สรุปและวิจารณ์ผลการทดลอง

สรุปและวิจารณ์ผลการทดลอง

งานวิจัยในวิทยานิพนธ์ฉบับนี้เสนอการเร่งความเร็วในการสร้างภาพตัดขวางจากภาพถ่ายเอ็กซ์เรย์โดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับวิเคราะห์ทางเดินแสงแบบกรวยบนระบบประมวลผลแบบขนาน เปรียบเทียบกับการสร้างภาพตัดขวางสำหรับวิเคราะห์ทางเดินแสงแบบกรวยโดยใช้อัลกอริทึมเชิงพีชคณิตวิธีเดิม ซึ่งในการขบวนการสร้างภาพตัดขวางสำหรับวิเคราะห์ทางเดินแสงแบบกรวยโดยใช้อัลกอริทึมเชิงพีชคณิตวิธีเดิมนั้น จะต้องใช้ระยะเวลาในการประมวลผลเพื่อที่จะให้เป็นภาพตัดขวางเป็นเวลาค่อนข้างนาน ดังนั้นในบทความวิจัยนี้จึงได้นำเอาหลักการของระบบประมวลผลแบบขนานบนคอมพิวเตอร์หลายเครื่องที่เรียกว่าระบบคลัสเตอร์และขบวนการสร้างภาพตัดขวางสำหรับวิเคราะห์ทางเดินแสงแบบกรวยโดยใช้อัลกอริทึมเชิงพีชคณิตเพื่อให้การสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตทำได้เร็วยิ่งขึ้น สำหรับการแบ่งงานในระบบประมวลผลแบบขนาน มีแบ่งงานโดยใช้วิธีราด-โรบิน(Round robin algorithm) เพื่อกระจายในการทำงานของการสร้างภาพตัดขวางในวิธีนี้ทำให้การทำงานรวมของระบบทั้งหมดเสร็จภายในเวลาใกล้เคียงกัน จากผลการทดลองระบบการคำนวณในงานวิจัยนี้สามารถลดเวลาในการคำนวณลงได้ 78 เปอร์เซ็นต์เมื่อเปรียบเทียบกับวิธีการสร้างภาพตัดขวางสำหรับวิเคราะห์ทางเดินแสงแบบกรวยโดยใช้อัลกอริทึมเชิงพีชคณิตบนระบบประมวลผล บนเครื่องประมวลผลเดี่ยวโดยสามารถแยกการทดลองออกเป็น 2 ส่วน

ส่วนแรกเป็นการทดลองสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบคลัสเตอร์จากแบบจำลองทางคอมพิวเตอร์ (ซึ่งได้ใช้แบบจำลองของ Shepp and Logan Phantom) โดยเปรียบเทียบด้านเวลา จากผลการทดลองจะสามารถสรุปได้คือ การสร้างภาพตัดขวางโดยใช้อัลกอริทึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบคลัสเตอร์นั้นสามารถเร่งความเร็วในการสร้างภาพตัดขวาง และการเปรียบเทียบเวลาระหว่างในการทำงานของระบบคลัสเตอร์ในอุดมคติ กับเวลาในการทำงานในระบบที่ทดลองจริง พบว่าเมื่อจำนวน โพรเซสเพิ่มขึ้นความเร็วในการสร้างภาพตัดขวางจะเพิ่มขึ้น จนถึงระดับหนึ่งความเร็วของระบบจะเริ่มคงที่ ซึ่งเป็นผลเนื่องมาจากเวลาที่ใช้ในการติดต่อสื่อสารกันระหว่างโพรเซสเพิ่มมากขึ้น ซึ่งจากการทดลองจะเห็นได้ว่าเมื่อ แบ่งโพรเซสออกมามากกว่าจำนวน โหนด จะทำให้ความเร็วที่เพิ่มขึ้นเริ่มคงที่ ในส่วนที่สองเป็นการทดลองเพื่อให้ทราบถึงผลกระทบเมื่อเปลี่ยนแปลงขนาดของปริมาตรที่จะทำการสร้างภาพตัดขวาง จากผลการทดลองจะเห็นได้ว่าการเปลี่ยนแปลงขนาดของปริมาตรนั้นมีผลกระทบทางด้านความเร็วค่อนข้างน้อยในการสร้างภาพตัดขวาง

สุดท้ายเป็นการประยุกต์การสร้างภาพตัดขวางจากภาพถ่ายเอ็กซเรย์ โดยในส่วนนี้ได้ทำการทดลองกับภาพถ่ายเอ็กซเรย์เท้าขวามนุษย์ในมุมที่ต่าง ๆ กัน จำนวน 60 ภาพ จากนั้นจึงนำมาประมวลผลโดยใช้กระบวนการการสร้างภาพตัดขวางโดยใช้อัลกอริธึมเชิงพีชคณิตสำหรับทางเดินแสงแบบกรวยบนระบบคลัสเตอร์ แล้วจึงนำภาพตัดขวางทั้งหมดมาสร้างข้อมูลเมตริกซ์เชิงปริมาตร จากนั้นจึงสร้างภาพ 3 มิติโดยวิธีการสร้างภาพเชิงพื้นผิวด้วยวิธีมาร์ชชิงคิวบ์ และการสร้างภาพเชิงปริมาตร ซึ่งจะเห็นได้ว่ามีสัญญาณรบกวนในภาพน้อยตามคุณสมบัติของการสร้างภาพตัดขวางโดยใช้อัลกอริธึมเชิงพีชคณิต อีกทั้งจะไม่มีปัญหาความถูกต้องในเรื่องระหว่างชั้นข้อมูลเนื่องจากระบบทำพร้อมๆ กันไป และการประมวลผลบนระบบคลัสเตอร์นั้นช่วยลดข้อด้อยเรื่องความเร็วในการประมวลผลของวิธีการสร้างภาพตัดขวางโดยใช้วิธีนี้คือ ช่วยทำให้ความเร็วในการประมวลผลเพิ่มขึ้น



เอกสารอ้างอิง

- [1] ทนาริป์ นิลแก้ว, ชูชาติ ปิณเขวิรุจน์, มนัส สัจจวิธิลปี และ กิตติพล ชิตสกุล, 2544. “การสร้างภาพ สามมิติจากภาพถ่ายเอ็กซ์เรย์” sohk 1220-1225. การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 24. กรุงเทพฯ : คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
- [2] S. Kaczmarz, “Angenaherte Auflosung Von Systemen Linearer Gleichungen,” *Bull. Acad. Pol. Sci. Lett. A*, vol. 6-8A, pp. 355-357, 1937.
- [3] H. Guan, and R. Gordon, “Computed Tomography using Algebraic Reconstruction Techniques (ART) with Different Projection Access Schemes: A Comparison Study Under Practical Situations,” *Phys. Med. Biol.*, No.41, pp.1727-1746, 1996.
- [4] Feldkamp, L. A., Davis, L. and Kress, K., “Practical Cone-Beam Algorithm,” *J. Opt.Soc.Am.* 1, pp 612-619.
- [5] C. Thompson and T. Peters, "A fractional address accumulator for fast backprojection," *IEEE Transactions on Nuclear Science*, vol.NS-28, no. 4, pp. 3648-3650, 1981.
- [6] R. Hartz, D. Bristow, and N. Mullani, "A real time TOFPET slice backproject engine using dual Am 29116 microprocessors," *IEEE Transactions on Nuclear Science*, vol. NS-32, pp. 839-842, February 1985.
- [7] Iskender Agi, P.J. Hurst, and K.W. Current, "An image processing IC for backprojection and spatial histogramming in a pipelined array," *IEEE Journal of Solid State Circuits*, vol. 28, no. 3, pp. 210- 220, 1993.
- [8] C.M. Chen, S.Y. Lee, and Z.H. Cho, "A parallel implementation of 3D CT image reconstruction on hypercube multiprocessor," *IEEE Transactions on Nuclear Science*, vol. 37, no. 3, pp. 1333 - 1346, 1990.
- [9] Mueller, K.; Yagel, R; “Rapid 3-D cone-beam reconstruction with the simultaneous algebraic reconstruction technique (SART) using 2-D texture mapping hardware” *Medical Imaging, IEEE Transactions on*, Volume: 19, Issue: 12, Dec. 2000 pp:1227 -1237
- [10] Petersen, J., “ Introduction to Programming on Distributed Memory Multiprocessor”, *Computer Physics Communications*, vol. 73, no. 1-3, pp.72-92, 1999.

- [11] Potmesil, M. and Hoffert, E. M., "The Pixel Machine: A parallel Image Computer", *Comput. Graphics*, vol. 23, no. 3, pp. 69-78, 1989.
- [12] Buyya R. "High Performance Cluster Computing: Architectures and Systems", Volume 1. Prentice Hall PTR. 1999.
- [13] MPICH Project. "MPICH-A Portable Implementation of MPI." , [Online] Available: <http://www-unix.mcs.anl.gov/mpi/mpich/>
- [14] A.C. Kak, and M. Slaney, "*Principles of Computerized Tomographic Imaging*," IEEE Press, NY, 1988.
- [15] A.H. Andersen, and A.C. Kak, "Simultaneous Algebraic Reconstruction Technique (SART): A Superior Implementation of the ART Algorithm," *Ultrason. Imaging*, vol. 6, pp. 81-94, Jan. 1984.
- [16] R. Gordon, et. al., "Algebraic Reconstruction Techniques (ART) for three dimensional electron microscopy and X-ray photography," *J. Theor. Biol.*, vol. 29, pp. 471-481, 1970.
- [17] P. Gilbert, "Iterative Method for the Reconstruction of Three Dimensional Objects from their Projection," *J. Theor. Biol.*, vol. 36, pp. 105-117, 1972
- [18] Michael J. Quinn "Parallel Programming in C with MPI and OpenMP" ISBN 007-282256-2

ภาคผนวก ก.

ผลงานวิจัยที่ได้รับการตีพิมพ์

1. Kosin Kalarat, Bunyarit Laksanapanai, Chuchart Pintavirooj, Manas Sangworasil, “*FAST CONE-BEAM TOMOGRAPHY ON CLUSTERING SYSTEM.*”, Proceeding of The 2005 ECTI International Conference, Vol 1, May 12-13,2005. pp.335-338.
2. Kosin Kalarat, Walita Narkbuakaew, Chuchart Pintavirooj, Manas Sangworasil “*Rapid Simultaneous Algebraic Reconstruction Technique(SART) for Cone-Beam Geometry on Clustering System.*” Proceeding of IEEE Tencon’05, Melbourne, Australia, pp.84, November 22-24, 2005.



ประวัติผู้เขียน

นายโกสินทร์ กาหรัศน์ เกิดเมื่อวันที่ 21 พฤษภาคม 2525 ที่จังหวัดสุราษฎร์ธานี สำเร็จการศึกษาระดับมัธยมปลายจากโรงเรียนสุราษฎร์ธานี จังหวัดสุราษฎร์ธานี สำเร็จการศึกษาปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมอิเล็กทรอนิกส์ จากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2547



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้