ผลกระทบของระบบหลายพาร์ทิชันในการจัดการงานแบบขนาน โดยใช้เป้าหมายเป็นหลัก

Impacts of Multi-partition Systems on Goal-oriented Parallel Computer Job Scheduling

อมรรัตน์ ประสิทธิ์ศุภโรจน์

Amonrat Prasitsupparote

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา
วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์
มหาวิทยาลัยสงขลานครินทร์

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Engineering in Computer Engineering
Prince of Songkla University
2554

ลิขสิทธิ์ของมหาวิทยาลัยสงขลานครินทร์

ชื่อวิทยานิพนธ์ ผลกระทบของระบบหลายพาร์ทิชันในการจัดการงานแบบขนานโดยใช้เป้าหมายเป็นหลัก

ผู้เขียน นางสาวอมรรัตน์ ประสิทธิ์ศุภโรจน์

สาขาวิชา วิศวกรรมคอมพิวเตอร์

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	**คณะกรรมการสอบ**
…….....	……..ประธานกรรมการ
(ดร.แสงสุรีย์ วสุพงศ์อัยยะ)	(ดร.สมชัย หลิมศิโรรัตน์)
	……..กรรมการ
อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม	(ดร.แสงสุรีย์ วสุพงศ์อัยยะ)
…….....	……..กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.พิชญา ตัณฑัยย์)	(ผู้ช่วยศาสตราจารย์ ดร.พิชญา ตัณฑัยย์)
	……..กรรมการ
	(ดร.เดือนเพ็ญ กชกรจารุพงศ์)

บัณฑิตวิทยาลัย มหาวิทยาลัยสงขลานครินทร์ อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

……..

(ศาสตราจารย์ ดร.อมรรัตน์ พงศ์ดารา)

คณบดีบัณฑิตวิทยาลัย

ชื่อวิทยานิพนธ์ ผลกระทบของระบบหลายพาร์ทิชันในการจัดการงานแบบขนานโดยใช้เป้าหมายเป็นหลัก

ผู้เขียน นางสาวอมรรัตน์ ประสิทธิ์ศุภโรจน์

สาขาวิชา วิศวกรรมคอมพิวเตอร์

ปีการศึกษา 2553

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้นำเสนอการจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันโดยใช้นโยบายแบบใช้เป้าหมายเป็นหลัก ซึ่งนโยบายแบบใช้เป้าหมายเป็นหลักจะค้นหาลำดับของงานที่จะเป็นไปได้ทั้งหมดภายในเวลาที่จำกัดเพื่อกำหนดลำดับของงานที่จะทำให้ระบบมีประสิทธิภาพสูงสุด วิทยานิพนธ์ฉบับนี้นำเสนอการทดสอบนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบหลายพาร์ทิชัน 2 วิธีคือ การผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ และการใช้ฮิวริสทิคในการเลือกพาร์ทิชัน ซึ่งจะทดสอบบนระบบคอมพิวเตอร์แบบ 2-5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน และแบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน โดยใช้ปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน จากการทดสอบพบว่านโยบายแบบใช้เป้าหมายเป็นหลักบนระบบหลายพาร์ทิชันโดยใช้เบสฟิทเป็นฮิวริสทิคในการเลือกพาร์ทิชันให้ประสิทธิภาพดี เมื่อเปรียบเทียบประสิทธิภาพกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งบนระบบแบบพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากัน และมีจำนวนหน่วยประมวลผลไม่เท่ากัน เนื่องจากนโยบายแบบใช้เป้าหมายเป็นหลักสามารถกระจายจำนวนงาน และงานขนาดต่าง ๆ ไปแต่ละพาร์ทิชันได้ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล

คำสำคัญ: แบ็กฟิล, หลายพาร์ทิชัน, ตัวกำหนดการงานแบบขนาน, นโยบายแบบใช้เป้าหมายเป็นหลัก, เบสฟิท

Thesis Title	Impacts of Multi-partition Systems on Goal-oriented Parallel Computer Job Scheduling
Author	Miss Amonrat Prasitsupparote
Major Program	Computer Engineering
Academic Year	2010

ABSTRACT

In this thesis, we focus on applying goal-oriented parallel computer job scheduling policies on multipartition parallel computer systems since several existing systems partition their system resources. Goal-oriented parallel computer job scheduling policies utilizes a combinatorial search technique to find the best schedule within a time limit. Two implementation ideas of multi-partition goal-oriented parallel computer job scheduling policies are proposed and evaluated using an event-driven simulator. The first implementation idea includes the partition selection question into the search space while the second idea selects a partition according to some heuristics resulting in a smaller search space. The experimental results show that the second idea with a best-fit partition selection heuristic performs well under both equal size and non-equal size two-partition systems. Furthermore, the proposed policy also outperforms the two basic priority backfill policies.

KEYWORDS: Multi-partition, job scheduling, goal-oriented, backfill, best-fit

กิตติกรรมประกาศ

ขอขอบพระคุณ ดร.แสงสุรีย์ วสุพงศ์อัยยะ ประธานกรรมการที่ปรึกษางานวิจัยที่ได้เสียสละเวลาในการให้คำปรึกษา แนวคิดในการทำวิจัย รวมถึงการช่วยเหลือแก้ไขปัญหาที่เกี่ยวกับการวิจัย ตลอดจนตรวจสอบและแก้ไขวิทยานิพนธ์ให้ดำเนินไปอย่างลุล่วงสมบูรณ์

ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.พิชญา ตัณฑัยย์ ที่ได้กรุณาให้คำปรึกษา คำแนะนำ และให้ความช่วยเหลือในงานวิจัย ตลอดจนช่วยตรวจทานแก้ไขวิทยานิพนธ์ให้ดำเนินไปด้วยดี

ขอขอบพระคุณ ดร.สมชัย หลิมศิโรรัตน์ ที่ได้กรุณาเสียสละเวลาเป็นประธานกรรมการสอบวิทยานิพนธ์ และตรวจทานแก้ไขวิทยานิพนธ์ให้มีความสมบูรณ์

ขอขอบพระคุณ ดร.เดือนเพ็ญ กชกรจารุพงศ์ ที่กรุณาเสียสละเวลาเป็นกรรมการสอบวิทยานิพนธ์ อีกทั้งตรวจทาน และแก้ไขวิทยานิพนธ์ให้มีความสมบูรณ์ยิ่งขึ้น

ขอขอบพระคุณ บัณฑิตวิทยาลัย มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่ ที่ให้การสนับสนุนทุนในการทำวิจัย และให้ความช่วยเหลือด้านการประสานงานต่าง ๆ

ขอขอบพระคุณ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์ ที่กรุณาให้ทุนผู้ช่วยวิจัยแก่ข้าพเจ้า

ขอขอบพระคุณ คณาจารย์ บุคลากร และนักศึกษาปริญญาโทภาควิชาวิศวกรรมคอมพิวเตอร์ทุกคนที่ได้ให้คำปรึกษา และกำลังใจในการทำงานเป็นอย่างดีเสมอมา

และสุดท้าย ข้าพเจ้าขอน้อมรำลึกถึงพระคุณของ บิดามารดา และครอบครัว ที่ส่งเสริม และสนับสนุนข้าพเจ้าในทุก ๆ เรื่องตลอดมาจนสำเร็จการศึกษา

อมรรัตน์ ประสิทธิ์ศุภโรจน์

สารบัญ

หน้า

สารบัญ .....(6)

รายการตาราง.....(10)

รายการภาพประกอบ.....(13)

สัญลักษณ์คำย่อและตัวย่อ.....(16)

บทที่ 1 บทนำ 1

1.1 วัตถุประสงค์.....3

1.2 ขอบเขตของการวิจัย.....3

1.3 ขั้นตอนและวิธีดำเนินงานวิจัย.....3

1.4 ประโยชน์ที่คาดว่าจะได้รับ 5

1.5 โครงสร้างของรายงาน.....5

บทที่ 2 ทฤษฎีและหลักการ 6

2.1 การจัดกำหนดการบนระบบคอมพิวเตอร์แบบขนาน.....6

2.2 ตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน 7

2.3 การจัดกำหนดการบนระบบคอมพิวเตอร์แบบขนานโดยใช้นโยบายแบบใช้เป้าหมายเป็นหลัก.....12

2.3.1 การจัดเรียงงานในคิวเพื่อสร้างโครงสร้างต้นไม้.....12

2.3.2 อัลกอริทึมสำหรับการค้นหา.....14

2.3.3 รูปแบบของเป้าหมาย.....16

2.3.3.1 รูปแบบเล็กซิคอล (Lexical Model).....17

2.3.3.2 รูปแบบอีควลเทรดออฟ (Equal-Tradeoff Model).....17

2.4 ระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน.....19

บทที่ 3 การออกแบบและวิธีทดสอบระบบ 23

สารบัญ (ต่อ)

หน้า

3.1 การจัดกำหนดการงานแบบขนานโดยใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน....23
3.1.1 การผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้....23
3.1.2 การใช้ฮิวริสทิคในการเลือกพาร์ทิชัน....25
3.1.3 การเลือกพาร์ทิชัน....26
3.2 การทดสอบประสิทธิภาพของนโยบาย....27
3.2.1 โปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน 28
3.2.1.1 ส่วนโปรแกรมหลัก....29
3.2.1.2 ส่วนนโยบายที่ใช้....30
3.2.2 เวิร์คโหลดไฟล์ (Workload File)....30
3.2.3 ค่าที่ใช้วัดประสิทธิภาพ (Measurement)....34
บทที่ 4 ผลการทดสอบและวิเคราะห์ผล....36
4.1 เปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักด้วยการผนวกการเลือกพาร์ทิชันกับการใช้ฮิวริสทิคในการเลือกพาร์ทิชัน....37
4.2 เปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน....39
4.2.1 ประสิทธิภาพบนระบบ 2 พาร์ทิชันแบบจำนวนหน่วยประมวลผลเท่ากัน....39
4.2.1.1 ประสิทธิภาพโดยรวมของระบบ....39
4.2.1.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน....41
4.2.1.3 วิเคราะห์ประสิทธิภาพ....42
4.2.2 ประสิทธิภาพบนระบบ 2 พาร์ทิชันแบบจำนวนหน่วยประมวลผลไม่เท่ากัน....45
4.2.2.1 ประสิทธิภาพโดยรวมของระบบ....45
4.2.2.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน....47

สารบัญ (ต่อ)

หน้า

4.2.2.3 วิเคราะห์ประสิทธิภาพ 49
4.2.3 ประสิทธิภาพบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 53
4.2.3.1 ระบบ 3 พาร์ทิชัน 53
4.2.3.2 ระบบ 4 พาร์ทิชัน 62
4.2.3.3 ระบบ 5 พาร์ทิชัน 71
4.3 ประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักเมื่อเปลี่ยนค่าการจำกัดจำนวนโหนด 81
4.4 เปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด.....83
4.4.1 ประสิทธิภาพบนระบบหลายพาร์ทิชันแบบจำนวนหน่วยประมวลผลเท่ากัน 83
4.4.1.1 ระบบ 2 พาร์ทิชัน 84
4.4.1.2 ระบบ 3 พาร์ทิชัน 90
4.4.1.3 ระบบ 4 พาร์ทิชัน 98
4.4.1.4 ระบบ 5 พาร์ทิชัน 107
4.4.2 ประสิทธิภาพบนระบบหลายพาร์ทิชันแบบจำนวนหน่วยประมวลผลไม่เท่ากัน.....117
4.4.2.1 ประสิทธิภาพโดยรวมของระบบ 117
4.4.2.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน 119
4.4.2.3 วิเคราะห์ประสิทธิภาพ 121
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ 126
5.1 สรุปผล 126
5.2 ผลที่ได้จากการทำวิทยานิพนธ์ชุดนี้.....129
5.3 ข้อเสนอแนะ.....129

สารบัญ (ต่อ)

หน้า

บรรณานุกรม 130

ภาคผนวก 132

ภาคผนวก ก การใช้งานโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน 133

ภาคผนวก ข อภิธานศัพท์ 141

ภาคผนวก ค ผลงานตีพิมพ์เผยแพร่จากวิทยานิพนธ์ 146

ประวัติผู้เขียน 155

รายการตาราง

หน้า

ตารางที่ 2-1 เปรียบเทียบวิธีการจัดเรียงงานในคิวเพื่อสร้างโครงสร้างต้นไม้ 14
ตารางที่ 2-2 เงื่อนไขของรูปแบบแลคซิคอน 17
ตารางที่ 2-3 เงื่อนไขของรูปแบบอีควลเทรดออฟ.....18
ตารางที่ 3-1 ข้อมูลโดยสรุปแต่ละเดือนของเวิร์คโหลดไฟล์ต้นฉบับ 31
ตารางที่ 3-2 ข้อมูลโดยสรุปแต่ละเดือนของเวิร์คโหลดไฟล์หลังจากแก้ไขโดยลดหน่วยประมวลผลลงไม่เกิน 64 สำหรับ 2 พาร์ทิชัน 32
ตารางที่ 3-3 ข้อมูลโดยสรุปแต่ละเดือนของเวิร์คโหลดไฟล์หลังจากแก้ไขโดยลดหน่วยประมวลผลลงไม่เกิน 32 สำหรับ 3, 4 และ 5 พาร์ทิชัน 33
ตารางที่ 4-1 สัญลักษณ์ต่าง ๆ.....36
ตารางที่ 4-2 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 43
ตารางที่ 4-3 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 44
ตารางที่ 4-4 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน 50
ตารางที่ 4-5 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน 51
ตารางที่ 4-6 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน 52
ตารางที่ 4-7 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 59
ตารางที่ 4-8 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 60
ตารางที่ 4-9 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน.....61
ตารางที่ 4-10 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 68

รายการตาราง (ต่อ)

หน้า

ตารางที่ 4-11 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 69
ตารางที่ 4-12 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 70
ตารางที่ 4-13 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 77
ตารางที่ 4-14 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 78
ตารางที่ 4-15 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 79
ตารางที่ 4-16 สัญลักษณ์แทนจำนวนโหนดที่ใช้ในการทดสอบระบบ 81
ตารางที่ 4-17 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 88
ตารางที่ 4-18 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 89
ตารางที่ 4-19 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 95
ตารางที่ 4-20 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 96
ตารางที่ 4-21 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 97
ตารางที่ 4-22 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 104
ตารางที่ 4-23 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 105
ตารางที่ 4-24 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 106

รายการตาราง (ต่อ)

หน้า

ตารางที่ 4-25 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 114
ตารางที่ 4-26 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 115
ตารางที่ 4-27 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด....116
ตารางที่ 4-28 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 122
ตารางที่ 4-29 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด.....123
ตารางที่ 4-30 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 124

รายการภาพประกอบ

หน้า

ภาพประกอบ 2-1 การจำลองการประมวลผลของระบบคอมพิวเตอร์แบบขนาน 7
ภาพประกอบ 2-2 การจัดกำหนดการโดยใช้นโยบายแบบเอฟซีเอฟเอส 9
ภาพประกอบ 2-3 การจัดกำหนดการโดยใช้นโยบายแบบอีซีเอฟซีเอฟเอสแบ็กฟิล.....9
ภาพประกอบ 2-4 การจัดกำหนดการโดยใช้นโยบายแบบคอนเซอร์เวทิฟเอฟซีเอฟเอสแบ็กฟิล 9
ภาพประกอบ 2-5 ต้นไม้สำหรับการค้นหาของงาน 4 งาน 13
ภาพประกอบ 2-6 การค้นหาของอัลกอริทึมแบบดีดีเอส.....15
ภาพประกอบ 2-7 การจำลองการประมวลผลของระบบคอมพิวเตอร์แบบขนานที่มี 2 พาร์ทิชัน 20
ภาพประกอบ 2-8 การส่งงานของตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานแบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากัน 21
ภาพประกอบ 3-1 โครงสร้างต้นไม้แบบผนวกการเลือกพาร์ทิชันขณะมีงานรออยู่ 4 งาน 24
ภาพประกอบ 3-2 โครงสร้างต้นไม้ของงาน 4 งาน ภายใต้การใช้ฮิวริสทิคในการเลือกพาร์ทิชัน ...26
ภาพประกอบ 3-3 การนำงานเข้าสู่ระบบของโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน 28
ภาพประกอบ 3-4 โครงสร้างของโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน 28
ภาพประกอบ 4-1 ประสิทธิภาพการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ และการใช้ฮิวริสทิคในการเลือกพาร์ทิชัน.....38
ภาพประกอบ 4-2 ประสิทธิภาพบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 40
ภาพประกอบ 4-3 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน.....42
ภาพประกอบ 4-4 ประสิทธิภาพบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน 46
ภาพประกอบ 4-5 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน 48
ภาพประกอบ 4-6 ประสิทธิภาพบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 54
ภาพประกอบ 4-7 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน.....56
ภาพประกอบ 4-8 ประสิทธิภาพบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน 62

รายการภาพประกอบ (ต่อ)

หน้า

ภาพประกอบ 4-9 ค่าสูงสุดของการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน....64
ภาพประกอบ 4-10 ค่าเฉลี่ยการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน....65
ภาพประกอบ 4-11 ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน....66
ภาพประกอบ 4-12 ประสิทธิภาพบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน....72
ภาพประกอบ 4-13 ค่าสูงสุดของการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน....74
ภาพประกอบ 4-14 ค่าเฉลี่ยการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน....75
ภาพประกอบ 4-15 ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน....76
ภาพประกอบ 4-16 ประสิทธิภาพเมื่อเปลี่ยนค่าการจำกัดจำนวนโหนด....82
ภาพประกอบ 4-17 ประสิทธิภาพบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด....84
ภาพประกอบ 4-18 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด....86
ภาพประกอบ 4-19 ประสิทธิภาพบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด....91
ภาพประกอบ 4-20 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด....92
ภาพประกอบ 4-21 ประสิทธิภาพบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด....98
ภาพประกอบ 4-22 ค่าเฉลี่ยการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด....100

รายการภาพประกอบ (ต่อ)

หน้า

ภาพประกอบ 4-23 ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 101
ภาพประกอบ 4-24 จำนวนงานที่รอมากกว่า 24 ชั่วโมงแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด.....102
ภาพประกอบ 4-25 ประสิทธิภาพบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด.....108
ภาพประกอบ 4-26 ค่าเฉลี่ยการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด....110
ภาพประกอบ 4-27 ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 111
ภาพประกอบ 4-28 จำนวนงานที่รอมากกว่า 12 ชั่วโมงแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด.....112
ภาพประกอบ 4-29 ประสิทธิภาพบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด.....118
ภาพประกอบ 4-30 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด 120
ภาพประกอบ ก-1 เวิร์คโหลดไฟล์แบบซิมพลิไฟด์.....135
ภาพประกอบ ก-2 เวิร์คโหลดไฟล์แบบเอสดับบิวเอฟ 135
ภาพประกอบ ก-3 ตัวอย่างเอาพุทของโปรแกรม.....140

สัญลักษณ์คำย่อและตัวย่อ

PBS	The Portable Batch System
PCB	Process Control Block
FCFS	First Come First Serve
LXF	Largest Slowdown First
SJF	Short Job First
LDS	Limited Discrepancy Search
ILDS	Improved Limited Discrepancy Search
DDS	Depth-Bounded Discrepancy Search
SWF	Standard Workload Format

บทที่ 1
บทนำ

ในปัจจุบันความก้าวหน้าทางด้านวิทยาศาสตร์ และเทคโนโลยีเพิ่มขึ้นอย่างรวดเร็ว ระบบคอมพิวเตอร์แบบขนาน (Parallel Computer System) เข้ามามีบทบาทกับงานในหลาย ๆ ด้าน เช่น งานด้านวิทยาศาสตร์โดยเฉพาะทางด้านเคมี เภสัชวิทยา งานวิจัยเกี่ยวกับยีนในเซลล์ของสิ่งมีชีวิต งานประมวลผลภาพทางการแพทย์ อุตสาหกรรมการออกแบบยาและเครื่องมือแพทย์ การศึกษาโครงสร้างและการเคลื่อนที่ของโมเลกุล งานควบคุมขีปนาวุธ งานควบคุมทางอวกาศ และงานด้านวิศวกรรมการออกแบบ อุตสาหกรรมการผลิตรถยนต์ เป็นต้น งานด้านต่าง ๆ ที่ได้กล่าวมานั้นจะมีลักษณะพิเศษไม่เหมือนงานอื่น ๆ เช่น มีข้อมูลปริมาณมหาศาลที่จะต้องประมวลผลหรือต้องการความรวดเร็วในการประมวลผล แม้ว่างานวิจัยลักษณะดังกล่าวบางงานสามารถประมวลผลบนเครื่องคอมพิวเตอร์ส่วนบุคคลได้ก็ตาม แต่การประมวลผลก็จะใช้เวลานานกว่าการประมวลผลบนระบบคอมพิวเตอร์แบบขนานหลายเท่า ในบางกรณีเครื่องคอมพิวเตอร์ส่วนบุคคลอาจจะไม่มีประสิทธิภาพเพียงพอที่จะประมวลผลงานดังกล่าวได้

แม้ว่าระบบคอมพิวเตอร์แบบขนานจะมีประสิทธิภาพสูงมาก แต่ระบบส่วนใหญ่ราคาจะสูงตามประสิทธิภาพ (ส่วนมากจะมีราคาระหว่าง 100 – 5,000 ล้านบาท) ดังนั้นการใช้งานระบบดังกล่าวให้ได้ประสิทธิภาพสูงสุดจึงเป็นสิ่งสำคัญ ทั้งนี้เป็นที่ยอมรับกันว่าการตัดสินใจเพื่อเลือกงานเข้าไปประมวลผลบนระบบคอมพิวเตอร์แบบขนานที่เหมาะสมจะช่วยให้ประสิทธิภาพของระบบดีขึ้นได้ ซึ่งตัวจัดกำหนดการงานแบบขนาน (Scheduler) ทำหน้าที่เลือกงานเพื่อเข้าไปประมวลผล ในปัจจุบันตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานเชิงพาณิชย์ที่มีใช้กันในศูนย์ให้บริการระบบประมวลผลแบบขนานต่างๆ ได้แก่ Maui [1], Moab [2], PBS [3], LSF [4] และ LoadLeveler [5] เป็นต้น การทำงานของตัวจัดกำหนดการงานแบบขนานเหล่านี้จะยึดหลักให้ผู้ดูแลระบบ (System Administrator) เป็นผู้ตัดสินใจในการเลือกนโยบาย (Policy) โดยการกำหนดค่าพารามิเตอร์ต่าง ๆ ของตัวจัดกำหนดการงานแบบขนาน (Scheduling Parameters) เพื่อให้ระบบสามารถจัดลำดับงานได้อย่างมีประสิทธิภาพ และตรงตามนโยบายที่ผู้ดูแลระบบกำหนดไว้ ทั้งนี้ในการใช้งานระบบดังกล่าวนี้ต้องการผู้ดูแลระบบที่มีความชำนาญในการเลือกใช้นโยบายให้เหมาะสมกับงานที่นำไปประมวลผล และต้องมีความเชี่ยวชาญในการกำหนดค่าพารามิเตอร์ต่าง ๆ ของนโยบายที่เลือกใช้ให้สอดคล้องกัน

การกำหนดค่าต่างๆ ของตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานไม่ใช่เรื่องง่าย เนื่องจากความต้องการทางประสิทธิภาพบางชุดจะมีความขัดแย้งกันเอง

เช่น หากผู้ดูแลระบบต้องการป้องกันปัญหาการอดอยาก (Starvation) หรือการรอที่นานมาก ระบบจะต้องให้ระดับความสำคัญ (Priority) กับงานที่ต้องการหน่วยประมวลผลมาก และใช้ระยะเวลาในการประมวลผล (Runtime) มากกว่างานอื่น ๆ เนื่องจากงานที่ต้องการหน่วยประมวลผลมาก และใช้ระยะเวลาการประมวลผลนานส่วนใหญ่จะเป็นงานที่ต้องรอเพื่อให้ระบบมีทรัพยากรเพียงพอสำหรับการประมวลผลทำให้งานลักษณะนี้ส่วนใหญ่รอนาน ในขณะเดียวกันระบบก็ต้องการให้งานสั้น ๆ เสร็จเร็ว เนื่องจากงานลักษณะดังกล่าวมีจำนวนมาก และเพื่อความยุติธรรมเนื่องจากงานประมวลผลสั้น ดังนั้นงานดังกล่าวก็ไม่สมควรจะต้องรอนาน ซึ่งการจะทำให้บรรลุเป้าหมายดังกล่าวระบบต้องให้ความสำคัญกับงานที่สั้นมากกว่างานอื่น ๆ จะเห็นได้ว่าการกำหนดความต้องการเพียง 2 สิ่งที่ได้กล่าวไปก็มีความขัดแย้งกันเองแล้ว

การให้ระดับความสำคัญของงานบนระบบคอมพิวเตอร์แบบขนานทั่วไปมีอยู่ 2 แบบกว้าง ๆ คือ การแบ่งระดับความสำคัญตามคิว (Queue-based Priority) ผู้ดูแลระบบต้องกำหนดคิว และแยกแยะแต่ละงานลงแต่ละคิวเอง ส่วนอีกแบบคือการแบ่งระดับความสำคัญตามงาน (Job-based Priority) ผู้ดูแลระบบจะเป็นคนกำหนดระดับความสำคัญของแต่ละงานเอง นโยบายทั้ง 2 แบบดังกล่าวต้องใช้ความชำนาญของผู้ดูแลระบบในการกำหนดค่าต่าง ๆ เอง อีกทั้งเมื่อเปลี่ยนเวิร์คโหลดของระบบจะเป็นการเพิ่มภาระงานของผู้ดูแลระบบในการวิเคราะห์เวิร์คโหลดอีกด้วย

เมื่อไม่นานมานี้ได้มีการนำเสนอนโยบายสำหรับตัวจัดกำหนดการงานแบบขนาน คือ การจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานโดยใช้เป้าหมายเป็นหลัก (Goal-oriented Parallel Computer Job Scheduling) [6] ซึ่งการจัดกำหนดการดังกล่าวจะลดภาระของผู้ดูแลระบบลง โดยการผลักภาระการกำหนดค่าพารามิเตอร์ต่าง ๆ (Parameter Setting) ไปให้ตัวจัดกำหนดการงานแบบขนานเป็นผู้จัดการ ผู้ดูแลระบบเพียงตั้งเป้าหมายของระบบว่าต้องการอะไรเช่น ต้องการให้ระบบไม่เกิดปัญหาการอดอยาก (Stavation) ต้องการให้งานสั้น ๆ เสร็จเร็วโดยไม่ต้องรอนาน เป็นต้น ตัวจัดกำหนดการงานแบบขนานจะค้นหาลำดับของงานให้ได้ตามเป้าหมายที่ตั้งไว้ทำให้ผู้ดูแลระบบไม่จำเป็นต้องกำหนดค่าพารามิเตอร์ต่าง ๆ ของตัวจัดกำหนดการงานแบบขนานเอง

ในงานวิจัย [6] และชิ้นอื่น ๆ ที่ตามมา [7], [8], [9] และ [10] ได้เน้นไปในเรื่องของการนำเสนอประสิทธิภาพนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบพาร์ทิชันเดียว (Single Partition) ทั้งนี้ตัวจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักยังไม่มีการทดสอบบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน (Multi-partition) ซึ่งระบบคอมพิวเตอร์แบบขนานทั่วไปนั้นจะมีทั้งแบบที่มีพาร์ทิชันเดียว และแบบหลายพาร์ทิชัน ซึ่งยังถูกแบ่งย่อยออกเป็นแบบหลายพาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน (Equal Size Partition)

และแบบหลายพาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน (Non-equal Size Partition) ในส่วนของระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันจะกล่าวโดยละเอียดในหัวข้อที่ 2.4

งานวิจัยชิ้นนี้มีแนวคิดที่จะศึกษาถึงประสิทธิภาพของตัวจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักเมื่อทำงานอยู่บนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน โดยจะศึกษาทั้งแบบที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากัน และแบบที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากัน รวมทั้งผลกระทบของจำนวนพาร์ทิชันที่มีต่อประสิทธิภาพของตัวจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักด้วย

1.1 วัตถุประสงค์

1. เพื่อปรับปรุงนโยบายของตัวจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักเพื่อให้ทำงานบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันได้
2. เพื่อนำเสนอประสิทธิภาพของตัวจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน ทั้งแบบที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากัน และมีจำนวนหน่วยประมวลผลไม่เท่ากัน

1.2 ขอบเขตของการวิจัย

1. ออกแบบตัวจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักเพื่อให้ทำงานบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน
2. แสดงประสิทธิภาพการทำงานของตัวจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันแบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากัน
3. แสดงประสิทธิภาพการทำงานของตัวจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันแบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากัน

1.3 ขั้นตอนและวิธีดำเนินงานวิจัย

ขั้นที่ 1: ศึกษาโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน (Simulator) เพื่อใช้เป็นเครื่องมือในการทดสอบนโยบาย

ขั้นที่ 2: ออกแบบ และพัฒนาตัวจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันบนโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน

ขั้นที่ 3: ตรวจสอบระบบการทำงานของส่วนประกอบต่าง ๆ ของตัวจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลัก

ขั้นที่ 4: ทดสอบประสิทธิภาพของการใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน และวิเคราะห์ผล

ขั้นที่ 4.1: เปรียบเทียบ และวิเคราะห์ประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันที่ออกแบบไว้

ขั้นที่ 4.2: เปรียบเทียบ และวิเคราะห์ประสิทธิภาพของการใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันโดยแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากันกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล

ขั้นที่ 4.3: วิเคราะห์ผลประสิทธิภาพของการใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันโดยแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากัน ซึ่งวิเคราะห์ประสิทธิภาพของแต่ละนโยบายแยกตามพาร์ทิชัน

ขั้นที่ 4.4: เปรียบเทียบ และวิเคราะห์ประสิทธิภาพของการใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันโดยแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากันกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล

ขั้นที่ 4.5: วิเคราะห์ผลประสิทธิภาพของการใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากัน โดยวิเคราะห์ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบาย

ขั้นที่ 4.6: เปรียบเทียบ และวิเคราะห์ผลกระทบของการเพิ่มเวลาในการตัดสินใจให้กับนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน

ขั้นที่ 4.7: วิเคราะห์ผลกระทบของนโยบายที่ออกแบบมาเมื่อจำนวนพาร์ทิชันเพิ่มเป็น 3, 4 และ 5 พาร์ทิชัน

ขั้นที่ 4.8: เปรียบเทียบ และวิเคราะห์ผลกระทบในกรณีทดสอบนโยบายที่ออกแบบมากับระบบที่มีความใกล้เคียงกับสถานการณ์จริงกล่าวคือ ตัวจัดกำหนดการงานแบบขนานจะใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด (User Estimate Runtime) ในการตัดสินใจ

ขั้นที่ 5: จัดทำรายงานฉบับสมบูรณ์ และเอกสารต้นฉบับ (Manuscript) เพื่อตีพิมพ์

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ระบบการจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน
2. ประสิทธิภาพ และแนวทางในการจัดการระบบการจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน

1.5 โครงสร้างของรายงาน

ในบทที่ 2 จะกล่าวถึงหลักการต่าง ๆ ที่จำเป็นเกี่ยวกับงานวิจัยชิ้นนี้ โดยบทที่ 3 จะกล่าวถึงนโยบายที่ออกแบบในงานวิจัยชิ้นนี้ และระบบที่ใช้ในการทดสอบนโยบาย บทที่ 4 จะกล่าวถึงผลที่ได้จากการทดสอบ และวิเคราะห์ผลการใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน และบทสุดท้ายจะกล่าวสรุปผลที่ได้จากการวิจัยครั้งนี้พร้อมข้อเสนอแนะเกี่ยวกับงานวิจัยนี้

บทที่ 2
ทฤษฎีและหลักการ

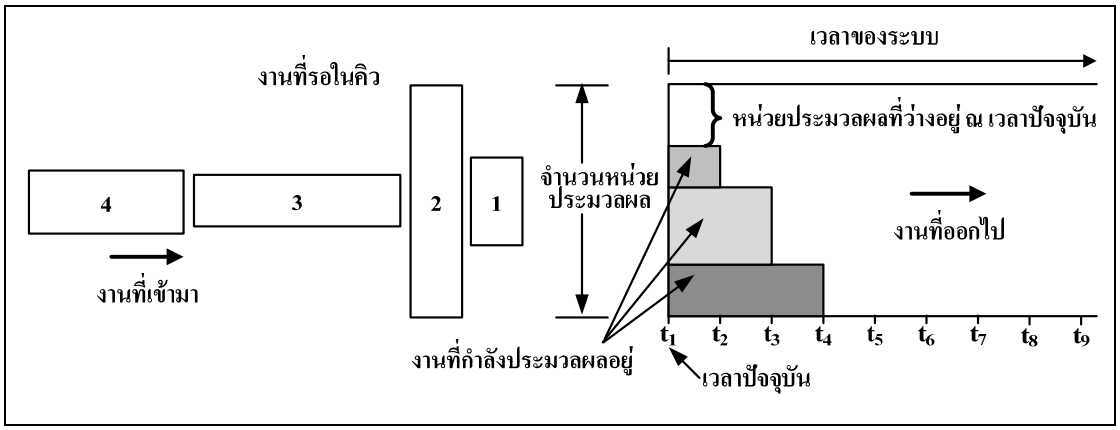
ทฤษฎีและหลักการที่จะนำมาใช้ในงานวิจัยชิ้นนี้ ประกอบด้วยรายละเอียดเกี่ยวกับการจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน (Parallel Computer Job Scheduling) ตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน (Parallel Computer Job Scheduler) การจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานโดยใช้เป้าหมายเป็นหลัก (Goal-oriented Parallel Computer Job Scheduling) ซึ่งเป็นนโยบายที่จะนำมาพัฒนาต่อเพื่อให้รองรับกับระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน โดยจะอธิบายถึงองค์ประกอบหลัก 3 ส่วนด้วยกันคือ การจัดเรียงงานในคิวเพื่อสร้างโครงสร้างต้นไม้ (Branching Heuristic) อัลกอริทึมสำหรับการค้นหา (Search Algorithm) และรูปแบบของเป้าหมาย (Objective Model) หัวข้อสุดท้ายในบทนี้จะกล่าวถึงระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน

2.1 การจัดกำหนดการบนระบบคอมพิวเตอร์แบบขนาน

เมื่อผู้ใช้ต้องการประมวลผลงานของตนเองบนระบบคอมพิวเตอร์แบบขนาน ผู้ใช้จะต้องส่งงานมายังระบบพร้อมทั้งระบุจำนวนทรัพยากรต่าง ๆ ที่งานต้องการใช้ส่งมาด้วยเช่น จำนวนหน่วยประมวลผลที่ต้องการ จำนวนหน่วยความจำที่ต้องการใช้ เวลาที่ผู้ใช้คาดว่างานจะใช้ในการประมวลผลจนเสร็จ เป็นต้น ทั้งนี้งานที่เข้ามาในระบบจะได้รับการแจกจ่ายทรัพยากรตามที่ต้องการ และสั่งให้เริ่มประมวลผลบนหน่วยประมวลผล ซึ่งมีตัวจัดกำหนดการงานแบบขนานเป็นผู้ที่ตัดสินใจเลือกงานที่รออยู่มาประมวลผล โดยปกติตัวจัดกำหนดการงานแบบขนานจะทำงานเมื่อมีงานใหม่เข้าสู่ระบบหรือมีงานเก่าที่ประมวลผลเสร็จแล้วออกจากระบบไป เมื่อมีงานเข้ามายังระบบและงานดังกล่าวไม่ได้ถูกเลือกให้ประมวลผล งานดังกล่าวจะไปรอในคิวการรอ (Waiting Queue) จนกว่าจะถูกเลือกโดยตัวจัดกำหนดการงานแบบขนานให้ไปประมวลผล ซึ่งตัวจัดกำหนดการงานแบบขนานจะเลือกงานไปประมวลผลโดยยึดตามนโยบายการจัดกำหนดการงานแบบขนาน (Scheduling Policy) ที่ระบบเลือกใช้ในการจัดกำหนดการงานแบบขนาน ในส่วนของนโยบายจะอธิบายอย่างละเอียดในหัวข้อต่อไป

ภาพประกอบที่ 2-1 แสดงการจำลองการประมวลผลของระบบคอมพิวเตอร์แบบขนานโดยแสดงเป็นแบบสองมิติ ทั้งนี้กำหนดให้แนวตั้งแทนจำนวนหน่วยประมวลผล และแนวนอนเป็นเวลา โดยกำหนดให้เวลาที่ t_1 เป็นเวลาปัจจุบัน จากภาพประกอบที่ 2-1 จะเห็นว่า ณ

เวลาปัจจุบันมีงานกำลังประมวลผลอยู่ 3 งาน มีหน่วยประมวลผลว่างอยู่จำนวนหนึ่ง และมีงานรอการประมวลผลอยู่ 4 งาน คือ งานหมายเลข 1, 2, 3 และ 4 ตามลำดับ โดยในรูปจะเรียงงานตามลำดับก่อนหลังที่งานถูกส่งเข้ามายังระบบ กล่าวคืองานที่เข้ามาทีหลังก็จะเข้ามาต่อท้ายคิวไปเรื่อย ๆ การเลือกงานที่รออยู่มาประมวลผลบนหน่วยประมวลผลที่ว่างจะขึ้นอยู่กับนโยบายการจัดกำหนดการ หัวข้อต่อไปจะกล่าวถึงตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน



ภาพประกอบ 2-1 การจำลองการประมวลผลของระบบคอมพิวเตอร์แบบขนาน

2.2 ตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน

ระบบคอมพิวเตอร์แบบขนานในปัจจุบันส่วนใหญ่จะมีการใช้นโยบายการจัดกำหนดการงานแบบขนานแบบนอนพรีเอ็มพ์ทีพ (Non-Preemptive) กล่าวคือหากงานใดได้รับสิทธิ์ให้ประมวลผล หรือกำลังประมวลผลอยู่ งานนั้นจะได้สิทธิ์ในการประมวลผลจนกว่าจะเสร็จโดยไม่มีอะไรมาขัดจังหวะ เนื่องจากการใช้นโยบายการจัดกำหนดการงานแบบขนานแบบพรีเอ็มพ์ทีพ (Preemptive) ต้องทำคอนเท็กท์สวิทชิง (Context switching) ซึ่งก่อให้เกิดโอเวอร์เฮด (Overhead) ตามมา อีกทั้งการย้ายงานบนระบบคอมพิวเตอร์แบบขนานจะต้องทำคอนเท็กท์สวิทชิงทุกหน่วยประมวลผลของระบบ ทำให้ระบบคอมพิวเตอร์แบบขนานที่ใช้จริงส่วนใหญ่ไม่นิยมใช้นโยบายแบบพรีเอ็มพ์ทีพ ดังนั้นในงานวิจัยชิ้นนี้จะมุ่งไปที่ระบบแบบนอนพรีเอ็มพ์ทีพเท่านั้น

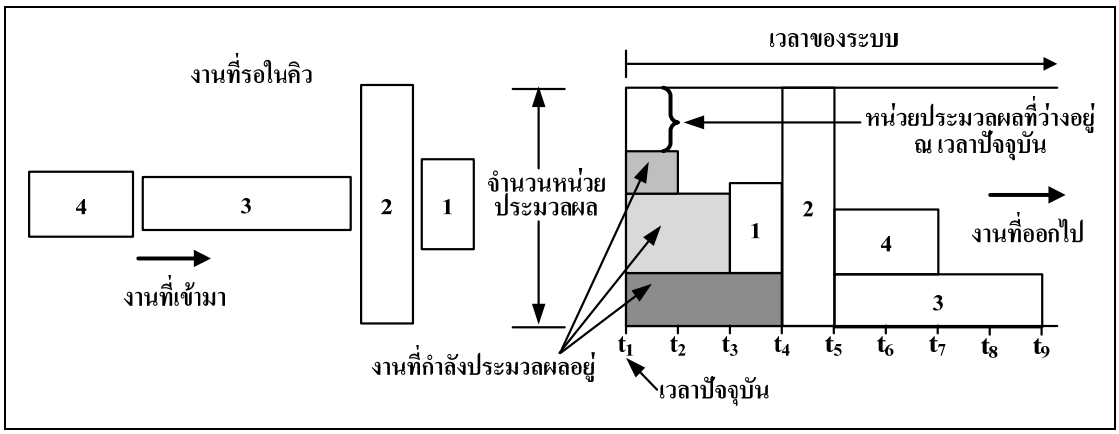
ตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานส่วนใหญ่ใช้หลักการทำงานแบบกำหนดระดับความสำคัญกล่าวคือ ตัวจัดกำหนดการงานแบบขนานจะเรียงลำดับงานที่มีอยู่ตามระดับความสำคัญที่กำหนดไว้ล่วงหน้า และพิจารณางานเพื่อให้เริ่ม

ประมวลผลตามลำดับดังกล่าว นอกจากนี้จะมีการเสริมเทคนิคที่ชื่อว่าแบ็กฟิล (Backfill) เพื่อช่วยเพิ่มประสิทธิภาพ ทั้งนี้การกำหนดระดับความสำคัญของงานมีหลักการกว้าง ๆ อยู่ 2 แบบคือ การแบ่งระดับความสำคัญตามคิว และการแบ่งระดับความสำคัญตามงาน

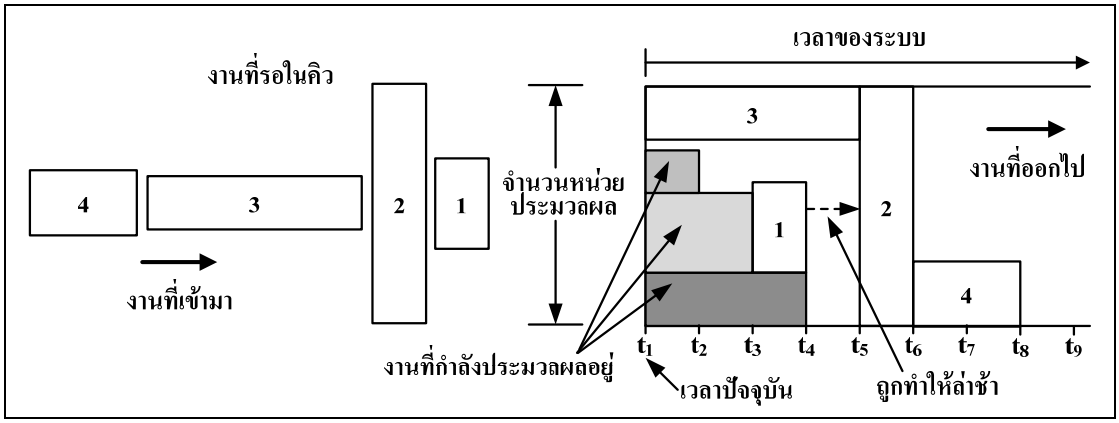
การแบ่งระดับความสำคัญตามคิว (เช่น นโยบายแบบ LSF [4]) ทำได้โดยผู้ดูแลระบบเป็นผู้กำหนดจำนวนคิว และระดับความสำคัญของแต่ละคิว โดยวิเคราะห์จากลักษณะงานที่อยู่บนระบบเช่น ความต้องการหน่วยประมวลผลของงาน หรือระยะเวลาการประมวลผลของงาน และแยกงานแต่ละชนิดไปแต่ละคิว ตัวจัดกำหนดการงานแบบขนานจะพิจารณางานมาประมวลผลตามระดับความสำคัญของคิว ข้อเสียของนโยบายแบบแบ่งระดับความสำคัญตามคิวคือ เมื่อระบบเปลี่ยนเวิร์คโหลดผู้ดูแลระบบต้องวิเคราะห์เวิร์คโหลดใหม่เพื่อจะกำหนดข้อมูลของแต่ละคิวใหม่ให้ถูกต้อง หากผู้ดูแลระบบวิเคราะห์เวิร์คโหลดผิดพลาดอาจส่งผลกระทบต่อประสิทธิภาพของระบบเช่น ปัญหาการอดอยาก จะเห็นได้ว่าการใช้งานตัวจัดกำหนดการงานแบบขนานลักษณะดังกล่าวต้องอาศัยความสามารถ และความชำนาญของผู้ดูแลระบบในการกำหนดระดับความสำคัญ

การแบ่งระดับความสำคัญตามงาน (เช่น นโยบายแบบ Maui [1]) ซึ่งตัวจัดกำหนดการงานแบบขนานลักษณะนี้ จัดระดับความสำคัญของแต่ละงานในระบบโดยอาจใช้เป็นการใช้ค่าวัดเพียงค่าเดียวเช่น เรียงตามลำดับก่อนหลัง หรือใช้การรวมตัววัดหลาย ๆ ชนิดแบบสมการความสำคัญถ่วงน้ำหนัก (Weighted Priority Function) ถึงแม้ว่านโยบายเช่นนี้จะมีความยืดหยุ่นกว่านโยบายแบบแบ่งระดับความสำคัญตามคิวก็ตาม นโยบายลักษณะนี้ก็จะสร้างปัญหาในเรื่องของการหาค่าน้ำหนักของสมการ เนื่องจากหน่วยวัดที่ใช้อาจจะไม่เท่ากันเช่น การใช้ค่าระยะเวลาการประมวลผล และค่าความต้องการหน่วยประมวลผลของงาน จะเห็นได้ว่า 2 ค่าดังกล่าวมีหน่วยวัดต่างกัน ดังนั้นภาระการหาค่าน้ำหนักที่ใช้งานจะตกอยู่ที่ผู้ดูแลระบบ

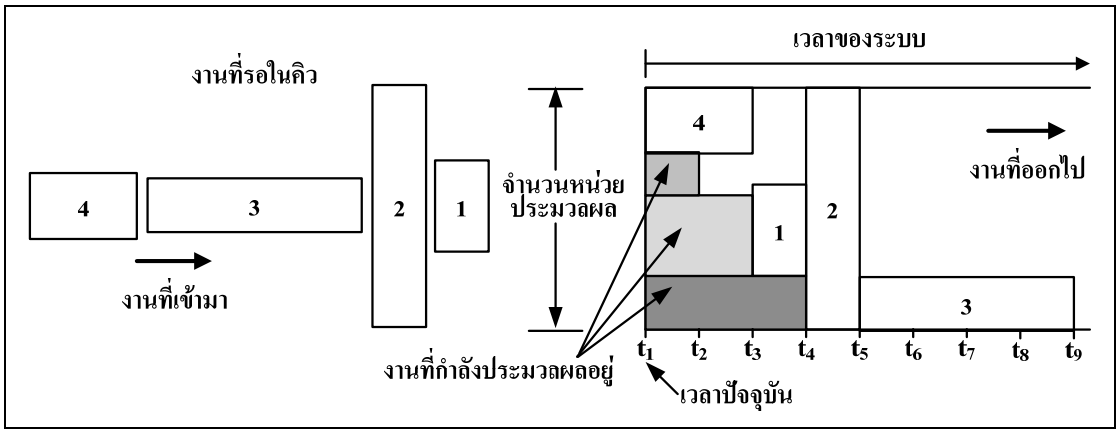
ตัวจัดกำหนดการงานแบบขนานเชิงพาณิชย์ทั่วไปจะจัดค่าเริ่มต้นเป็นนโยบายแบบเอฟซีเอฟเอส (First Come First Serve: FCFS) ซึ่งนโยบายดังกล่าวจะเรียงระดับความสำคัญของงานโดยให้ความสำคัญกับงานที่มาก่อน ดังนั้นตัวจัดกำหนดการงานแบบขนานจะพิจารณางานตามลำดับก่อนหลังที่งานเข้ามาในระบบ ถ้างานใดถูกส่งเข้ามาก่อนก็จะมีสิทธิ์ได้ไปประมวลผลก่อน ซึ่งก็น่าจะยุติธรรมแต่หากมีงานที่ต้องการหน่วยประมวลผลจำนวนมากเข้ามาสู่ระบบก่อนงานที่ต้องการหน่วยประมวลผลน้อย และระบบในขณะนั้นมีทรัพยากรไม่เพียงพอให้งานที่ต้องการหน่วยประมวลผลจำนวนมากทำงานได้ งานที่ต้องการหน่วยประมวลผลจำนวนน้อยที่ตามมาก็จะต้องรอ เนื่องจากตนเองเข้าสู่ระบบหลังจากงานที่ต้องการหน่วยประมวลผลมาก แม้ว่าทรัพยากรที่ว่างอยู่ปัจจุบันจะเพียงพอให้งานที่ต้องการหน่วยประมวลผลน้อยทำงานได้ก็ตาม



ภาพประกอบ 2-2 การจัดกำหนดการโดยใช้นโยบายแบบเอฟซีเอฟเอส



ภาพประกอบ 2-3 การจัดกำหนดการโดยใช้นโยบายแบบอีซีเอฟซีเอฟเอสแบ็กฟิล



ภาพประกอบ 2-4 การจัดกำหนดการโดยใช้นโยบายแบบคอนเซอร์เวทิฟเอฟซีเอฟเอสแบ็กฟิล

เพื่อให้เห็นตัวอย่างของปัญหาที่เกิดจากนโยบายเอฟซีเอฟเอส ภาพประกอบที่ 2-2 แสดงการจัดกำหนดการงานแบบขนานโดยใช้นโยบายแบบเอฟซีเอฟเอส กำหนดให้แนวตั้งแทนจำนวนหน่วยประมวลผล ส่วนในแนวนอนเป็นเวลาของระบบ และกำหนดให้เวลาที่ t_1 เป็นเวลาปัจจุบัน ณ เวลาปัจจุบันจะเห็นได้ว่ามีงานกำลังประมวลผลอยู่ 3 งาน และมีหน่วยประมวลผลว่างอยู่จำนวนหนึ่ง ที่เวลา t_1 มีงานรอการประมวลผลอยู่ 4 งาน คือ งานหมายเลข 1, 2, 3 และ 4 ตามลำดับ โดยในรูปจะเรียงงานตามลำดับก่อนหลังที่งานถูกส่งเข้ามายังระบบ นั้นคืองานที่เข้ามาทีหลังก็จะเข้ามาต่อท้ายคิวไปเรื่อย ๆ ตัวจัดกำหนดการงานแบบขนานจะเลือกงานหมายเลข 1 ออกมาจากคิวก่อน แต่จะเห็นว่างานหมายเลข 1 ไม่สามารถเริ่มประมวลผลได้ที่เวลา t_1 เพราะงานมีขนาดใหญ่เกินพื้นที่ว่างที่เหลืออยู่ของระบบที่เวลา t_1 ดังนั้นงานหมายเลข 1 จึงต้องรอให้งานที่กำลังประมวลผลอยู่ในระบบบางงานเสร็จก่อน งานหมายเลข 1 จึงจะเริ่มประมวลผลได้ งานหมายเลข 2 ก็ต้องรอให้งานทั้งหมดในระบบประมวลผลจนเสร็จก่อนจึงจะเริ่มประมวลผลได้เพราะงานหมายเลข 2 ต้องการจำนวนหน่วยประมวลผลทั้งหมดของระบบ และงานหมายเลข 3 และ 4 ต้องรอให้งานหมายเลข 2 ประมวลผลให้เสร็จก่อน จะเห็นได้ว่าการจัดกำหนดการงานแบบเอฟซีเอฟเอสไม่ยุติธรรมสำหรับงานหมายเลข 3 และ 4 ที่เป็นงานสั้นแต่ต้องรอให้งานที่กำลังประมวลผลอยู่ และงานหมายเลข 1 และ 2 ประมวลผลให้เสร็จก่อน ซึ่งส่งผลให้ค่าการใช้ประโยชน์ (Utilization) ของระบบต่ำ ดังนั้นจึงมีผู้นำเสนอการเพิ่มค่าการใช้ประโยชน์ของระบบโดยใช้เทคนิคแบ็กฟิล [12]

เทคนิคแบ็กฟิลจะอนุญาตให้งานที่ไม่ได้รับสิทธิ์ในการใช้ทรัพยากร ณ เวลาปัจจุบันของระบบสามารถใช้ทรัพยากรได้หากระบบมีทรัพยากรเหลือเพียงพอสำหรับงานนั้น และการประมวลผลของงานนี้จะต้องไม่ทำให้งานที่ได้รับสิทธิ์ในการใช้ทรัพยากร (Reservation) ต้องล่าช้าออกไป (Delay) จะเห็นได้ว่าเทคนิคแบ็กฟิลทำให้งานที่มีระดับความสำคัญต่ำกว่าสามารถเข้าไปใช้ทรัพยากรที่ว่างอยู่ได้โดยไม่กระทบต่องานที่มีระดับความสำคัญสูงกว่า ส่งผลให้ระบบมีค่าการใช้ประโยชน์สูงขึ้น อีกทั้งยังช่วยป้องกันปัญหาการอดอยากของงานที่ต้องการหน่วยประมวลผลจำนวนมากได้อีกด้วย

การป้องกันไม่ให้งานที่ได้รับสิทธิ์ในการใช้ทรัพยากรต้องล่าช้าออกไปของเทคนิคแบ็กฟิลแบ่งออกเป็น 2 ลักษณะคือ แบบที่คำนึงถึงงานแรกเพียงงานเดียวดังภาพประกอบที่ 2-3 และแบบที่คำนึงถึงงานทุกงานที่มีระดับความสำคัญสูงกว่าดังภาพประกอบที่ 2-4

ภาพประกอบที่ 2-3 นำเสนอการใช้นโยบายแบบเอฟซีเอฟเอสแบ็กฟิลที่คำนึงถึงงานแรกเพียงงานเดียวซึ่งมีอีกชื่อหนึ่งว่าอีซี (Easy) ดังนั้นตัวจัดกำหนดการงานแบบขนานจะเริ่มต้นใช้นโยบายแบบเอฟซีเอฟเอส งานหมายเลข 1 ถูกนำมาพิจารณาให้ได้รับสิทธิ์ในการใช้ทรัพยากรก่อนแต่เนื่องจากงานหมายเลข 1 ไม่สามารถเริ่มประมวลผลได้ที่เวลา t_1 ดังนั้นงานหมายเลข 1 จะ

เป็นงานที่ได้รับสิทธิ์จองใช้ทรัพยากร ณ เวลา t_3 หลังจากนั้นงานหมายเลข 2 ก็ถูกพิจารณา เนื่องจากงานหมายเลข 2 ต้องการจำนวนหน่วยประมวลผลทั้งหมดของระบบ ทำให้งานหมายเลข 2 น่าจะทำงานได้ที่เวลา t_4 ต่อมาตัวจัดกำหนดการงานแบบขนานจะนำงานหมายเลข 3 มาพิจารณา งานหมายเลข 3 สามารถใช้งานทรัพยากร ณ เวลาปัจจุบันได้ โดยไม่ทำให้เวลาที่จองไว้สำหรับงานหมายเลข 1 เลื่อนออกไป ถึงแม้ว่าการกระทำดังกล่าวจะทำให้งานหมายเลข 2 ได้เริ่ม ณ เวลา t_5 ก็ตาม การใช้เทคนิคอีซีแบ็กฟิลจะอนุญาตให้งานหมายเลข 3 ถูกแบ็กฟิลมาใช้ทรัพยากร ณ เวลา t_1 ได้ จะเห็นได้ว่าเทคนิคอีซีแบ็กฟิลจะให้ความสำคัญกับงานแรกที่ได้รับสิทธิ์จองใช้ทรัพยากรเท่านั้น ในทางตรงข้ามเทคนิคที่ให้ความสำคัญกับงานที่มีระดับความสำคัญสูงกว่าทุกตัวตามลำดับความสำคัญจะเรียกว่าคอนเซอร์เวทิฟ (Conservative) ดังภาพประกอบที่ 2-4 ตัวจัดกำหนดการงานแบบขนานจะเลือกงานหมายเลข 1 มาพิจารณา แต่งานหมายเลข 1 ไม่สามารถประมวลผลได้ที่เวลาปัจจุบัน ดังนั้นงานหมายเลข 1 จะได้รับสิทธิ์จองใช้ทรัพยากรไว้ก่อน ณ เวลา t_3 หลังจากนั้นตัวจัดกำหนดการงานแบบขนานจะพิจารณางานหมายเลข 2 ซึ่งก็ไม่สามารถเริ่มงานได้จึงได้รับสิทธิ์จองใช้ทรัพยากรไว้ ณ เวลา t_4 แล้วพิจารณางานหมายเลข 3 ซึ่งสามารถใช้ทรัพยากร ณ เวลาปัจจุบัน (t_1) ได้แต่การแบ็กฟิลงานหมายเลข 3 จะทำให้เวลาที่จองไว้สำหรับงานหมายเลข 2 ต้องล่าช้าออกไปจึงทำให้งานหมายเลข 3 ไม่สามารถถูกแบ็กฟิลได้ ดังนั้นงานหมายเลข 3 จะได้รับสิทธิ์จองใช้ทรัพยากรไว้ ณ เวลา t_5 ต่อมาตัวจัดกำหนดการงานแบบขนานก็พิจารณางานหมายเลข 4 ซึ่งจะเห็นได้ว่างานหมายเลข 4 สามารถใช้งานทรัพยากร ณ เวลาปัจจุบันได้ โดยไม่ทำให้เวลาที่จองไว้สำหรับงานทุกตัวที่มีความสำคัญสูงกว่าต้องล่าช้าออกไป ทำให้ระบบแบ็กฟิลงานหมายเลข 4 ที่เวลาปัจจุบัน

จากภาพประกอบที่ 2-2, 2-3 และ 2-4 จะเห็นได้ว่าการใช้นโยบายที่มีแบ็กฟิลร่วมด้วยจะทำให้ค่าการใช้ประโยชน์ของระบบเพิ่มขึ้นจากการใช้นโยบายแบบเอฟซีเอฟเอสเพียงอย่างเดียว เมื่อเปรียบเทียบการใช้เทคนิคแบ็กฟิลทั้ง 2 แบบคือแบบอีซี และแบบคอนเซอร์เวทิฟ งานวิจัย [12] ได้นำเสนอให้เห็นว่าเทคนิคอีซีแบ็กฟิลให้ประสิทธิภาพดีกว่าเทคนิคคอนเซอร์เวทิฟแบ็กฟิล เนื่องจากเทคนิคอีซีแบ็กฟิลมีความยืดหยุ่นมากกว่าเพราะสนใจงานแรกที่ได้รับสิทธิ์จองใช้ทรัพยากรเพียงงานเดียวเท่านั้นที่ต้องไม่ถูกทำให้ล่าช้า ในงานวิจัยชิ้นนี้จะเปรียบเทียบประสิทธิภาพกับเทคนิคแบ็กฟิลแบบอีซี

แต่การใช้เทคนิคแบ็กฟิลนั้นตัวจัดกำหนดการงานแบบขนานจะต้องรู้ความต้องการหน่วยประมวลผลของงาน และระยะเวลาการประมวลผลของงานช่วงที่งานถูกส่งเข้ามาในระบบเพื่อใช้ในการคำนวณว่างานที่ส่งเข้ามาจะสามารถแบ็กฟิลได้หรือไม่ ในระบบจริงนั้นตัวจัดกำหนดการงานแบบขนานจะไม่ทราบค่าระยะเวลาในการประมวลผลของงาน แต่จะทราบเพียง

ค่าประมาณระยะเวลาประมวลผลที่ผู้ใช้เป็นผู้กำหนดขณะส่งงานเข้าสู่ระบบ ซึ่งส่วนใหญ่ค่าประมาณดังกล่าวจะมากกว่าค่าระยะเวลาการประมวลผลจริงทำให้การแบ็กฟิลโดยใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดอาจจะเกิดการผิดพลาดมากกว่าการใช้ค่าระยะเวลาการประมวลผลจริง

2.3 การจัดกำหนดการบนระบบคอมพิวเตอร์แบบขนานโดยใช้นโยบายแบบใช้เป้าหมายเป็นหลัก

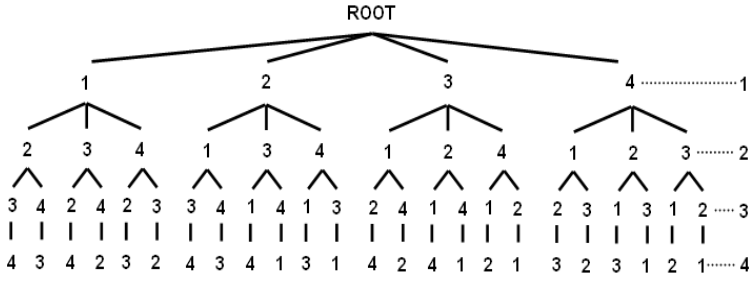
เมื่อไม่นานมานี้ได้มีผู้นำเสนอนโยบายแบบใช้เป้าหมายเป็นหลักสำหรับระบบคอมพิวเตอร์แบบขนาน [6] ซึ่งสามารถช่วยลดภาระของผู้ดูแลระบบลงได้ระดับหนึ่ง กล่าวคือจากเดิมที่จะใช้นโยบายแบบกำหนดระดับความสำคัญของงานที่จะได้รับการพิจารณาเพื่อประมวลผลร่วมกับเทคนิคแบ็กฟิลก็เปลี่ยนเป็นให้ผู้ดูแลระบบสามารถกำหนดเป้าหมายที่ต้องการแทน จากนั้นตัวจัดกำหนดการงานแบบขนานจะจัดการกับพารามิเตอร์ต่าง ๆ ที่จำเป็น เพื่อให้ได้กำหนดการงานแบบขนาน (Schedule) ที่ดีที่สุดตามนโยบายที่วางไว้ ทั้งนี้นโยบายดังกล่าวได้ประยุกต์ใช้อัลกอริทึมสำหรับการค้นหามาช่วยในการค้นหากำหนดการงานแบบขนานที่ดี และเป็นไปตามนโยบายที่กำหนดไว้ โดยมองปัญหาการค้นหากำหนดการงานแบบขนานที่ดีเป็นปัญหาของการค้นหาการจัดเรียงลำดับที่เหมาะสมตามเป้าหมายที่กำหนดแทนการให้ผู้ดูแลระบบมากำหนดระดับความสำคัญเอง เนื่องจากในเวลาใดเวลาหนึ่งนั้น ตัวจัดกำหนดการงานแบบขนานจะมีจำนวนงานที่รอประมวลผลอยู่จำนวนหนึ่ง หากนำงานเหล่านี้มาจัดเรียงเป็นโครงสร้างต้นไม้ กำหนดการงานที่เป็นไปได้ทั้งหมดคือเส้นทางที่ค้นหาได้ทั้งหมดจากโครงสร้างต้นไม้ ตัวจัดกำหนดการงานแบบขนานเพียงค้นหากำหนดการงานแบบขนานที่เหมาะสมในขณะนั้นเพื่อใช้ประกอบการตัดสินใจ

การใช้นโยบายแบบใช้เป้าหมายเป็นหลักจะประกอบด้วย 3 ส่วนหลัก ๆ คือ การจัดเรียงงานในคิวเพื่อสร้างต้นไม้สำหรับการค้นหา อัลกอริทึมสำหรับการค้นหา และรูปแบบของเป้าหมาย นโยบายแบบใช้เป้าหมายเป็นหลักจะนำงานที่รออยู่ในคิวมาสร้างเป็นโครงสร้างต้นไม้ไว้สำหรับค้นหาเส้นทาง ซึ่งเส้นทางที่ได้จะหมายถึงการจัดกำหนดการงานแบบขนาน และคำนวณค่าของเส้นทางที่ได้เพื่อเปรียบเทียบกับเส้นทางที่ถูกเก็บไว้ โดยจะเก็บเส้นทางที่ดีที่สุดเพียงเส้นทางเดียวเท่านั้น ระบบจะค้นหา และเปรียบเทียบเช่นนี้ไปเรื่อย ๆ จนกว่าจะค้นหาครบทุกเส้นทางหรือค้นหาจนถึงเวลาที่ระบบได้กำหนดไว้

2.3.1 การจัดเรียงงานในคิวเพื่อสร้างโครงสร้างต้นไม้

เมื่องานถูกส่งเข้ามาในระบบ งานจะต้องเข้ามารอในคิวเพื่อให้ตัวจัดกำหนดการงานแบบขนานตัดสินใจว่างานใดจะได้ใช้ทรัพยากรตามนโยบายที่กำหนด สำหรับนโยบายแบบใช้

เป้าหมายเป็นหลักจะนำงานทั้งหมดในคิวมาสร้างเป็นโครงสร้างต้นไม้ดังแสดงในภาพประกอบที่ 2-5 ซึ่งแสดงตัวอย่างของโครงสร้างต้นไม้เมื่อมีงาน 4 งานรออยู่ในระบบ



ภาพประกอบ 2-5 ต้นไม้สำหรับการค้นหาของงาน 4 งาน

ภาพประกอบที่ 2-5 แสดงการจัดเรียงงาน 4 งานซึ่งแทนด้วยหมายเลข 1 ถึง 4 ตามลำดับก่อนหลังที่งานเข้าสู่ระบบ ส่วนบนสุดของต้นไม้จะเรียกว่า โหนดราก (Root Node) และส่วนปลายสุดของต้นไม้ที่ไม่มีโหนดใดมาต่อนั้นจะเรียกว่า โหนดใบ (Leaf Node) เส้นทางจากโหนดรากจนถึงโหนดใบมี 4 ระดับซึ่งเท่ากับระดับความลึกของต้นไม้ (จำนวนงานที่รออยู่ในระบบขณะนั้นนั่นเอง) ทั้งนี้การสร้างโครงสร้างต้นไม้ต้องคำนึงถึงการจัดเรียงงานจากซ้ายไปขวา ในภาพประกอบที่ 2-5 งานถูกจัดเรียงโดยใช้การเรียงตามลำดับก่อนหลังของงานจากซ้ายไปขวา จะเห็นได้ว่างานแรกที่อยู่โหนดซ้ายสุดที่ระดับที่ 1 คืองานหมายเลข 1 งานโหนดถัดไปคืองานหมายเลข 2, 3 และ 4 จากนั้นในระดับความลึกที่ 2 จะเห็นว่ามีโครงสร้างต้นไม้ย่อย (Sub Tree) 4 โครงสร้าง พิจารณาโครงสร้างต้นไม้ย่อยทางซ้ายสุดก่อนจะเห็นได้ว่าโหนดจะเรียงจากงานหมายเลข 2, 3 และ 4 ซึ่งเป็นการเรียงจากซ้ายไปขวาตามลำดับก่อนหลังที่งานเข้าสู่ระบบ กระบวนการนี้ทำไปเรื่อย ๆ จนครบทุกระดับความลึกของโครงสร้างต้นไม้ จะเห็นว่าโครงสร้างต้นไม้มีเส้นทาง (Path) ที่เป็นไปได้ทั้งสิ้น 24 เส้นทางจากโหนดรากถึงโหนดใบ ซึ่งการนำงาน 4 งานมาจัดลำดับก็เป็นไปได้เพียง 24 ลำดับนี้เท่านั้น ซึ่งตัวจัดกำหนดการงานแบบขนานสามารถค้นหาลำดับที่เหมาะสมกับนโยบายได้จากโครงสร้างต้นไม้ดังกล่าว

งานวิจัย [13] ซึ่งนำเสนอนโยบายแบบใช้เป้าหมายเป็นหลักได้แสดงประสิทธิภาพของวิธีการจัดเรียงงานสำหรับสร้างโครงสร้างต้นไม้แบบเอฟซีเอฟเอส และแบบเอลเอกซ์เอฟ (Largest Slowdown First: LXF) ซึ่งเรียงโดยใช้ค่าสโลว์ดาวน์จากมากไปหาน้อย ซึ่งสรุปได้ดังตารางที่ 2-1 แสดงการเปรียบเทียบวิธีการจัดเรียงงานในคิวเพื่อสร้างโครงสร้างต้นไม้แบบเอฟซีเอฟเอส และแบบเอลเอกซ์เอฟ

ดังนั้นในงานวิจัยชิ้นนี้จะเลือกใช้การจัดเรียงงานในคิวเพื่อสร้างโครงสร้างต้นไม้แบบเอลเอกซ์เอฟเป็นวิธีสำหรับการจัดเรียงงานในคิวเพื่อมาสร้างโครงสร้างต้นไม้เท่านั้น

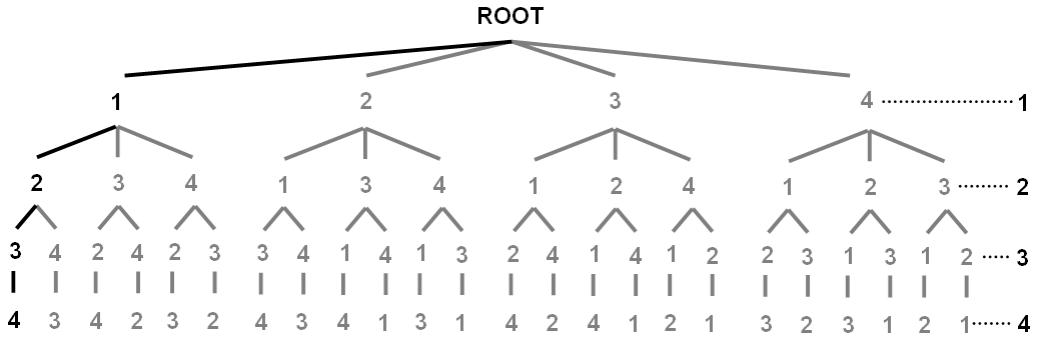
ตารางที่ 2-1 เปรียบเทียบวิธีการจัดเรียงงานในคิวเพื่อสร้างโครงสร้างต้นไม้

วิธีการจัดเรียงงาน	**การจัดเรียงงานขึ้นอยู่กับ**	**ข้อดี**	**ข้อเสีย**
เอฟซีเอฟเอส	ลำดับที่งานถูกส่งเข้ามาในระบบ	ไม่เสียเวลาในการคำนวณลำดับของงาน	ประสิทธิภาพที่ได้จากการจัดเรียงแย่กว่าการจัดเรียงแบบเอลเอกซ์เอฟ
เอลเอกซ์เอฟ	ค่าสโลว์ดาวน์ของงาน	ประสิทธิภาพที่ได้จากการจัดเรียงดีกว่าการจัดเรียงแบบเอฟซีเอฟเอส	ต้องเสียเวลาในการคำนวณค่าสโลว์ดาวน์

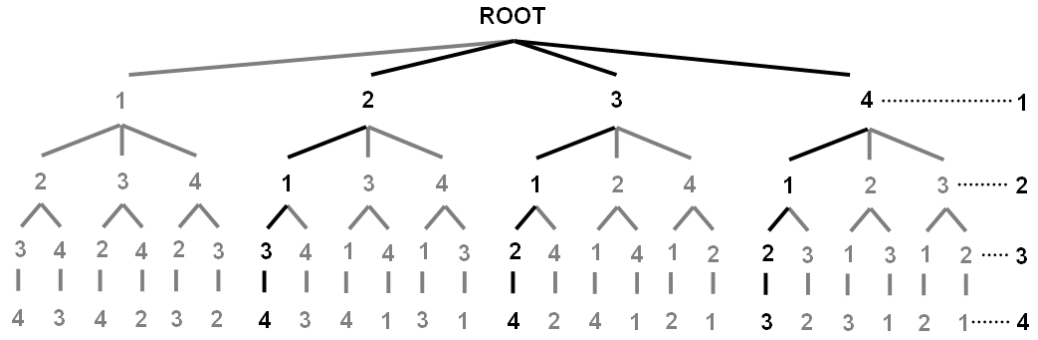
2.3.2 อัลกอริทึมสำหรับการค้นหา

เมื่อได้โครงสร้างต้นไม้มาแล้วก็ต้องมีการค้นหาการเรียงลำดับที่เหมาะสม ทั้งนี้ในการค้นหานั้นจะมีผลกระทบต่อประสิทธิภาพของตัวจัดกำหนดการงานแบบขนาน เนื่องจากหากเสียเวลาการค้นหาที่เดิมนาน ๆ ก็จะทำให้ประสิทธิภาพของตัวจัดกำหนดการงานแบบขนานลดลง เนื่องจากต้นไม้จะมีขนาดใหญ่ขึ้นตามจำนวนงานเช่น จำนวนงานในคิว 8 งาน เส้นทางที่เป็นไปได้ทั้งหมดมีถึง 40,000 เส้นทาง และหากเพิ่มจำนวนงานในคิวเป็น 10 งาน เส้นทางที่เป็นไปได้สูงถึง 3.5 ล้านเส้นทาง จะเห็นได้ว่าจำนวนงานในคิวเพิ่มขึ้นเพียงเล็กน้อยแต่จำนวนเส้นทางของโครงสร้างต้นไม้กลับเพิ่มขึ้นเป็นทวีคูณ นั่นคือจำนวนงานในคิวส่งผลโดยตรงต่อขนาดของต้นไม้ เนื่องจากจำนวนเส้นทางของโครงสร้างต้นไม้ที่เพิ่มขึ้นจะทำให้ขนาดของต้นไม้ใหญ่ขึ้น ดังนั้นตัวค้นหาต้องสามารถค้นหาได้หลาย ๆ ส่วนของต้นไม้เพื่อหาการเรียงลำดับงานที่เหมาะสมกับนโยบายปัจจุบันให้มากที่สุด นโยบายแบบใช้เป้าหมายเป็นหลักที่นำเสนอใน [13] ได้ทดสอบอัลกอริทึมการค้นหาที่อาศัยหลักการดิสครีแพนซี (Discrepancy) โดยมองว่าด้านซ้ายสุดของโครงสร้างต้นไม้จะเป็นด้านที่ถูก กล่าวคือด้านที่คิดว่าจะค้นหาพบเส้นทางที่ถูกต้อง ดังนั้นควรเริ่มค้นหาจากด้านซ้ายก่อน และค่อย ๆ ขยับไปด้านขวา อัลกอริทึมที่อาศัยหลักการดิสครีแพนซีถูกนำเสนอไว้ 2 อัลกอริทึมคือแอลดีเอส (Limited Discrepancy Search: LDS) [14], [15] จะเน้นการค้นหาลงไปทางซ้ายเท่านั้นเพราะคิดว่าจะเป็นทางที่ถูกต้อง และดีดีเอส (Depth-Bounded

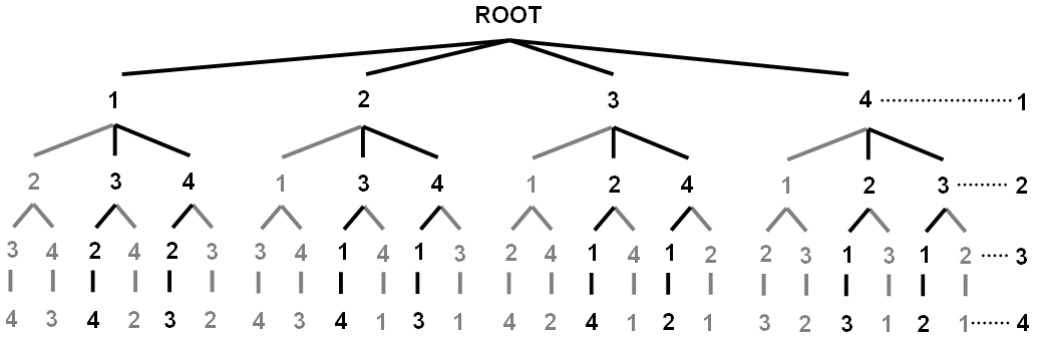
Discrepancy Search: DDS) [16] ซึ่งจะค้นหาไปทางซ้าย และขวาบ้างเพื่อจะได้ค้นหาลงในต้นไม้ย่อยอื่น ๆ พร้อมกันไปด้วยดังภาพประกอบที่ 2-6



(ก) ดีดีเอสรอบที่ 0



(ข) ดีดีเอสรอบที่ 1



(ค) ดีดีเอสรอบที่ 2

ภาพประกอบ 2-6 การค้นหาของอัลกอริทึมแบบดีดีเอส

ภาพประกอบที่ 2-6 แสดงเส้นทางที่ได้จากการค้นหาของอัลกอริทึมแบบดีดีเอส ในรอบต่าง ๆ จะเห็นได้ว่ารอบที่ 0 จะค้นหาเฉพาะทางซ้ายเท่านั้นจึงได้เส้นทางซ้ายสุดเพียงเส้นทางเดียวคือ 1-2-3-4 (ภาพประกอบ 2-6 (ก)) รอบที่ 1 จะค้นหาลงมาทางขวา 1 ระดับจากโหนดราก และค้นหาลงไปทางซ้าย ซึ่งการค้นหามาทางด้านขวานี้จะทำให้การค้นหาได้เส้นทางจากต้นไม้ย่อยอื่น และไม่ได้เส้นทางซ้ำกับเส้นทางเดิมที่ได้ค้นหาไปแล้ว ทำให้รอบนี้ได้เส้นทาง 3 เส้นทาง คือ 2-1-3-4, 3-1-2-4 และ 4-1-2-3 (ภาพประกอบ 2-6 (ข)) ต่อมาในรอบที่ 2 จะค้นหาลงไปทุกต้นไม้ย่อยโดยลงไปทางขวา 1 ระดับ และค้นหาลงไปทางซ้าย ซึ่งจะได้ทั้งหมด 8 เส้นทางคือ 1-3-2-4, 1-4-2-3, 2-3-1-4, 2-4-1-3, 3-2-1-4, 3-4-1-2, 4-2-1-3 และ 4-3-1-2 (ภาพประกอบ 2-6 (ค)) จะเห็นได้ว่าเส้นทางที่ค้นหาได้ในแต่ละรอบไม่ซ้ำกันเลย อีกทั้งในแต่ละรอบนั้นเส้นทางที่ได้ในลำดับแรก ๆ ก็ไม่ซ้ำกันอีกด้วย เนื่องจากเส้นทางที่ได้จะหมายถึงการจัดเรียงลำดับงาน หากการจัดเรียงลำดับงานแตกต่างกันตั้งแต่ลำดับแรกก็จะทำให้เกิดความแตกต่างของลำดับงานที่ได้ สำหรับการจัดกำหนดการงานแบบขนานนั้นงานในลำดับแรก ๆ จะมีผลอย่างมากต่องานในลำดับถัด ๆ ไป ดังนั้นในงานวิจัย [13] สรุปว่าอัลกอริทึมสำหรับการค้นหาแบบดีดีเอสเหมาะสำหรับนโยบายแบบใช้เป้าหมายเป็นหลัก ซึ่งงานวิจัยชิ้นนี้เลือกที่จะใช้เฉพาะอัลกอริทึมสำหรับการค้นหาแบบดีดีเอสเท่านั้น

เนื่องจากในหลายกรณีการค้นหาไม่สามารถค้นหาได้จนครบทุกเส้นทางของโครงสร้างต้นไม้จึงต้องมีการกำหนดเงื่อนไขสำหรับหยุดการค้นหา ซึ่งในงานวิจัย [13] ใช้จำนวนโหนดที่ค้นหาพบเป็นเงื่อนไขสำหรับหยุดการค้นหา ดังนั้นในงานวิจัยชิ้นนี้จะจำกัดจำนวนโหนดที่ค้นหา แต่จะทดสอบประสิทธิภาพของนโยบายเมื่อขยายเวลาการค้นหา โดยทำการทดสอบตั้งแต่ 2,000 ถึง 128,000 โหนด เมื่อค้นหาเส้นทางได้แต่ละเส้นทางตัวจัดกำหนดการงานแบบขนานจะต้องคำนวณค่าของเส้นทางที่ได้ตามเป้าหมายที่วางไว้ว่าเป็นเส้นทางที่ดีกว่าเส้นทางที่เจอก่อนหน้าหรือไม่ ดังนั้นในหัวข้อถัดไปจะกล่าวถึงวิธีการคำนวณค่าของเส้นทางตามเป้าหมายที่วางไว้

2.3.3 รูปแบบของเป้าหมาย

หลังจากค้นหาลงไปจนถึงโหนดใบก็จะได้มา 1 เส้นทาง โดยที่แต่ละเส้นทางคือกำหนดการงานแบบขนาน ต่อมาตัวจัดกำหนดการงานแบบขนานจะประเมินว่ากำหนดการงานแบบขนานที่ได้มาดีหรือไม่ โดยเป้าหมายที่กำหนดไว้จะเป็นตัวบอกว่าต้องคำนวณอย่างไร หลังจากนั้นเมื่อตัวจัดกำหนดการงานแบบขนานพิจารณาเส้นทางอื่นจนถึงโหนดใบก็จะสามารถคำนวณค่าเป้าหมายเพื่อนำมาเปรียบเทียบกับค่าเป้าหมายของเส้นทางแรกที่พบ และจะจดจำเส้นทางที่ได้ค่า

เป้าหมายที่ดีที่สุดไว้เพียงเส้นทางเดียวเพื่อเปรียบเทียบกับเส้นทางอื่น ๆ ต่อไป งานวิจัยที่นำเสนอนโยบายแบบใช้เป้าหมายเป็นหลักได้ศึกษารูปแบบของเป้าหมายโดยละเอียดใน [17] ดังนี้

2.3.3.1 รูปแบบเล็กซิคอล (Lexical Model)

สัญลักษณ์ที่ใช้คือ $Lexical(x \rightarrow y)$ โดยที่ x คือเป้าหมายแรก และ y คือเป้าหมายที่สอง รูปแบบเล็กซิคอลจะให้ความสำคัญกับเป้าหมายแรกมากกว่าเป้าหมายที่สอง ซึ่งสามารถเขียนเป็นรหัสเทียม (Pseudocode) ได้ดังตารางที่ 2-2

ตารางที่ 2-2 เงื่อนไขของรูปแบบแลคซิคอน

$if\ (x_1\ better\ x_2)\ then$ $\quad S_1\ better\ S_2$ $if\ (x_1\ =\ x_2)$ && $(y_1\ better\ y_2)\ then$ $\quad S_1\ better\ S_2$
กำหนดให้: S_1 คือเส้นทางที่ 1 S_2 คือเส้นทางที่ 2 x_1 คือค่าเป้าหมายแรกของเส้นทางที่ 1 x_2 คือค่าเป้าหมายแรกของเส้นทางที่ 2 y_1 คือค่าเป้าหมายที่สองของเส้นทางที่ 1 y_2 คือค่าเป้าหมายที่สองของเส้นทางที่ 2

จะเห็นได้ว่าตัวจัดกำหนดการงานแบบขนานสามารถพิจารณาได้ว่าเส้นทางใดดีกว่าตามเงื่อนไขในตารางที่ 2-2 ซึ่งตัวจัดกำหนดการงานแบบขนานจะเก็บเพียงเส้นที่ดีที่สุด ตั้งแต่เริ่มต้นหาเพียงเส้นทางเดียว รูปแบบเล็กซิคอลจึงขึ้นอยู่กับเป้าหมายแรกมากกว่า กล่าวคือผู้ดูแลระบบต้องการให้ระบบมีประสิทธิภาพด้านใดด้านหนึ่งมากกว่าอีกด้านหนึ่ง

2.3.3.2 รูปแบบอีควลเทรดออฟ (Equal-Tradeoff Model)

สัญลักษณ์ที่ใช้คือ $Tradeoff(x{:}y)$ โดยที่ x คือเป้าหมายแรก และ y คือเป้าหมายที่สอง รูปแบบอีควลเทรดออฟจะให้ความสำคัญกับเป้าหมายแรกเท่ากับเป้าหมายที่สอง นั่นคือทุกเป้าหมายมีความสำคัญเท่ากัน ดังนั้นตัวจัดกำหนดการงานแบบขนานสามารถคำนวณค่าของเส้นทางแต่ละเส้นทางได้ตามเงื่อนไขในตารางที่ 2-3

ตารางที่ 2-3 เงื่อนไขของรูปแบบอีควลเทรดออฟ

$if\ (x_1\ better\ x_2)\ \&\&\ (y_1\ better\ y_2)\ then$ $S_1\ better\ S_2$ $if\ (x_1\ better\ x_2)\ \&\&\ (y_1\ =\ y_2)\ then$ $S_1\ better\ S_2$ $if\ (x_1\ =\ x_2)\ \&\&\ (y_1\ better\ y_2)\ then$ $S_1\ better\ S_2$ $if\ ((x_1\ better\ x_2)\ \&\&\ (y_1\ worse\ y_2))\ \&\&\ ((I_1\ -\ I_2)\ >\ R_1)\ then$ $S_1\ better\ S_2$ $if\ ((x_1\ worse\ x_2)\ \&\&\ (y_1\ better\ y_2))\ \&\&\ ((I_2\ -\ I_1)\ >\ R_2)\ then$ $S_2\ better\ S_1$
กำหนดให้: S_1 คือเส้นทางที่ 1 S_2 คือเส้นทางที่ 2 x_1 คือค่าเป้าหมายแรกของเส้นทางที่ 1 x_2 คือค่าเป้าหมายแรกของเส้นทางที่ 2 y_1 คือค่าเป้าหมายที่สองของเส้นทางที่ 1 y_2 คือค่าเป้าหมายที่สองของเส้นทางที่ 2 R_1 คือค่าอ้างอิงของเป้าหมายแรก R_2 คือค่าอ้างอิงของเป้าหมายที่สอง I_1 คืออัตราส่วนการดีกว่าหรือแย่กว่าของค่าเป้าหมายแรก $\frac{x_2}{x_1}$ I_2 คืออัตราส่วนการดีกว่าหรือแย่กว่าของค่าเป้าหมายที่สอง $\frac{y_2}{y_1}$

จากตารางที่ 2-3 จะเห็นได้ว่าเมื่อค่าเป้าหมายตัวใดตัวหนึ่งดีกว่าแต่อีกตัวแย่กว่า จะต้องคำนวณส่วนต่างของอัตราการดีกว่าต่ออัตราการแย่กว่า โดยเปรียบเทียบกับค่าอ้างอิง (Reference Value) จึงจะสรุปได้ว่าเส้นทางใดดีกว่า ซึ่งการคำนวณค่าอ้างอิงนั้นจะแตกต่างกันไปตามชุด (Version) ที่ใช้ ซึ่งในงานวิจัย [6] ได้นำเสนอการใช้ค่าอ้างอิงของรูปแบบอีควลเทรดออฟไว้ 3 ชุดคือ ชุดเอ (Version A), ชุดบี (Version B) และชุดซี (Version C) และได้เปรียบเทียบให้เห็นถึงประสิทธิภาพของทั้ง 3 ชุด สรุปได้ว่าชุดเอจะมีประสิทธิภาพดีพอ ๆ กับชุดอื่น ๆ อีกทั้งไม่จำเป็นต้องเสียเวลาในการคำนวณค่าอ้างอิงมากนัก ซึ่งค่าอ้างอิงของชุดเอจะคำนวณจากค่าของกำหนดการงานแบบขนานที่ตัวจัดกำหนดการงานแบบขนานเก็บไว้ของแต่ละเป้าหมาย

นอกจากนี้งานวิจัย [17] ได้นำเสนอประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักเมื่อผู้ดูแลระบบต้องการให้ระบบไม่เกิดปัญหาการอดอยาก และต้องการให้งานสั้นไม่ต้อง

รอนาน ซึ่งได้เปรียบเทียบประสิทธิภาพของการใช้เป้าหมาย และรูปแบบของเป้าหมายแบบต่าง ๆ โดยสามารถสรุปได้ว่ารูปแบบของเป้าหมายแบบอีควลเทรดออฟชุดเอโดยใช้เป้าหมายเป็นค่าผลรวมการรอที่มากเกินกำหนด (Total Excessive Wait Time: Tw) และค่าเฉลี่ยสโลว์ดาวน์ (Average Slowdown: avgX) ให้ประสิทธิภาพได้ตรงตามที่ผู้ดูแลระบบต้องการมากที่สุด ดังนั้นในงานวิจัยชิ้นนี้จึงเลือกใช้รูปแบบอีควลเทรดออฟชุดเอโดยใช้เป้าหมายเป็นค่าผลรวมการรอที่มากเกินกำหนด และค่าเฉลี่ยสโลว์ดาวน์ ในการทดสอบนโยบายแบบใช้เป้าหมายเป็นหลักแบบหลายพาร์ทิชัน สามารถเขียนเป็นสัญลักษณ์ได้ดังนี้ Tradeoff(Tw : avgX)

ค่าการรอที่มากเกินกำหนดถูกนำเสนอไว้ในงานวิจัย [6], [7], [8], [9], [13] และ [17] โดยสามารถคำนวณได้จากค่าการรอ (Waitting Time) ที่มากเกินค่าที่กำหนด (Threshold : ω) ซึ่งค่าที่กำหนดมี 2 แบบคือแบบกำหนดไว้ก่อน (Fixed Threshold) และแบบเปลี่ยนแปลงตามค่าการรอในคิว (Dynamic Threshold) ใน [13] ได้แสดงให้เห็นว่าการใช้ค่าที่กำหนดแบบเปลี่ยนแปลงตามค่าการรอในคิวจะทำให้นโยบายแบบใช้เป้าหมายเป็นหลักมีความยืดหยุ่นตามสถานการณ์ และเวิร์คโหลด จึงส่งผลให้มีประสิทธิภาพมากกว่าการใช้ค่าที่กำหนดแบบกำหนดไว้ก่อน ดังนั้นในงานวิจัยนี้จะเลือกใช้ค่าที่กำหนดแบบเปลี่ยนแปลงตามค่าการรอในคิวโดยคำนวณจากค่าการรอในคิวของงานแต่ละงาน ค่าการรอสูงสุดจากค่าการรอทั้งหมดของแต่ละงานในคิวคือค่าที่กำหนด

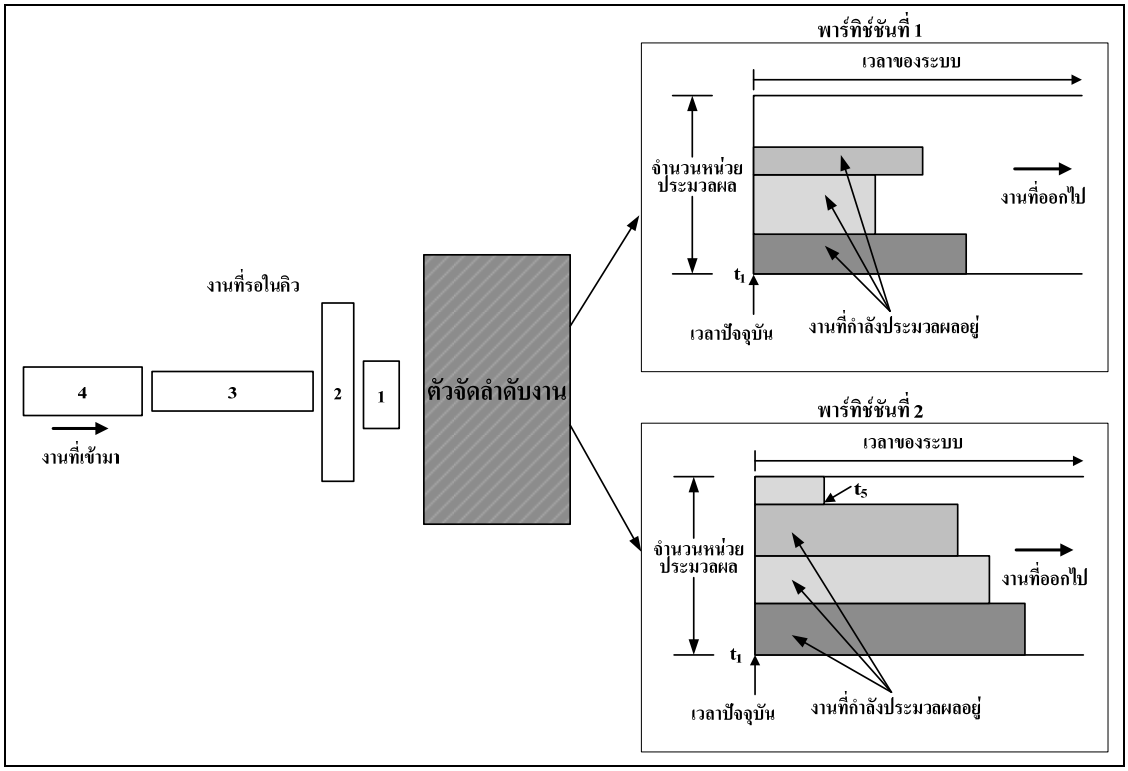
เนื่องจากงานวิจัยชิ้นนี้จะทดสอบนโยบายสำหรับระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน ดังนั้นในหัวข้อถัดไปจะกล่าวถึงระบบคอมพิวเตอร์แบบขนานหลายพาร์ทิชัน

2.4 ระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน

ระบบคอมพิวเตอร์แบบขนานทั่วไปสามารถแบ่งได้เป็น 2 แบบคือแบบพาร์ทิชันเดียว เช่น ระบบคลัสเตอร์ (Cluster System) และแบบหลายพาร์ทิชัน เช่น ระบบกริด (Grid System) โดยส่วนใหญ่ระบบคลัสเตอร์มักจะอยู่ในสถานที่เดียวกันเชื่อมต่อกันผ่านเครือข่ายความเร็วสูงภายในเพื่อให้ง่ายต่อการจัดการ แต่ระบบคลัสเตอร์ก็สามารถทำให้เป็นแบบหลายพาร์ทิชันได้เช่นกัน ระบบหลายพาร์ทิชันส่วนใหญ่จะใช้เป็นระบบกริดซึ่งระบบอาจอยู่หรือไม่อยู่ในสถานที่เดียวกันก็ได้ โดยแต่ละพาร์ทิชันเรียกว่าไซท์ (Site) หรือ โฮสท์ (Host)

ตัวอย่างการประมวลผลของระบบคอมพิวเตอร์แบบขนานเมื่อระบบมี 2 พาร์ทิชัน แสดงในภาพประกอบที่ 2-7 กำหนดให้แนวตั้งแทนจำนวนหน่วยประมวลผล ส่วนในแนวนอนเป็นเวลาของระบบ และกำหนดให้เวลาที่ t_1 เป็นเวลาปัจจุบัน ณ เวลาปัจจุบันจะเห็นได้ว่ามีงานกำลังประมวลผลอยู่ 3 งานที่พาร์ทิชันที่ 1 และมีหน่วยประมวลผลเหลืออยู่จำนวนหนึ่ง แต่พาร์ทิชันที่ 2 มีงานประมวลผลอยู่ 4 งาน และไม่มีหน่วยประมวลผลเหลืออยู่เลยจนกว่าจะถึงเวลา t_5 ที่เวลา t_1 มี

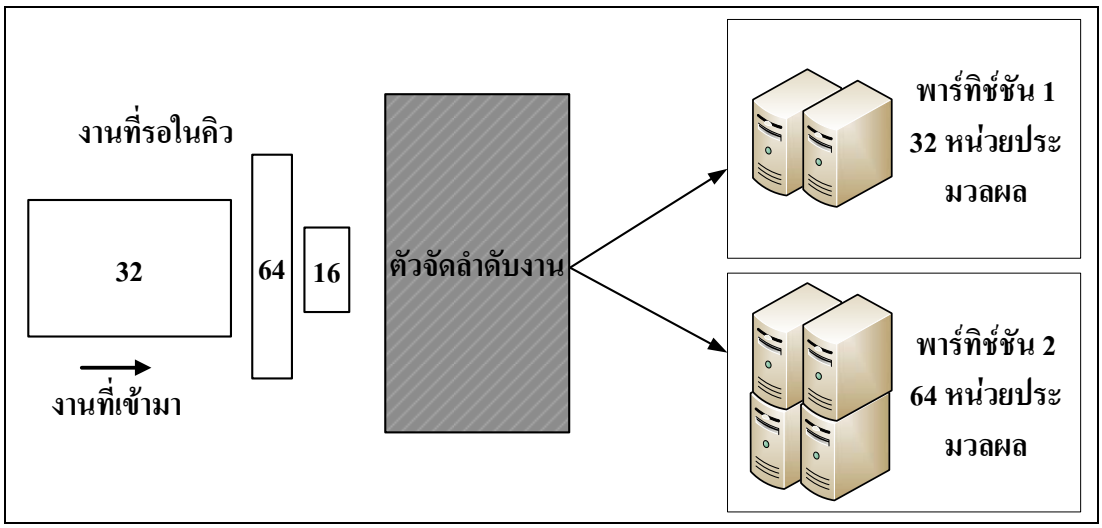
งานรอการประมวลผลอยู่ 4 งาน คือ งานหมายเลข 1, 2, 3 และ 4 ตามลำดับ โดยในรูปจะเรียงงานตามลำดับก่อนหลังที่งานถูกส่งเข้ามายังระบบ กล่าวคืองานที่เข้ามาหลังก็จะเข้ามาต่อท้ายคิวไปเรื่อย ๆ ตัวจัดกำหนดการงานแบบขนานจะต้องมีข้อมูลจำนวนหน่วยประมวลผลที่ว่างอยู่ของแต่ละพาร์ทิชันที่เวลาต่าง ๆ เนื่องจากตัวจัดกำหนดการงานแบบขนานจะใช้ข้อมูลดังกล่าวในการเลือกงานไปประมวลผลบนพาร์ทิชันใด และ ณ เวลาใด ซึ่งจะเห็นได้ว่าทั้งระบบจะมีตัวจัดกำหนดการงานแบบขนานเพียงตัวเดียว และมีคิวงานเพียงคิวเดียวเท่านั้น ซึ่งตัวจัดกำหนดการงานแบบขนานในลักษณะนี้เรียกว่า ตัวจัดกำหนดการงานแบบขนานแบบส่วนกลาง (Central Global Scheduler) [18] ซึ่งจะคอยจัดการกับงานจากคิวซึ่งมีเพียงคิวเดียวทั้งระบบซึ่งเรียกว่า คิวแบบส่วนกลาง (Central Global Queue) นอกจากนี้แต่ละพาร์ทิชันอาจจะมีจำนวนทรัพยากรที่ไม่เท่ากันได้ ดังตัวอย่างในภาพประกอบที่ 2-8



ภาพประกอบ 2-7 การจำลองการประมวลผลของระบบคอมพิวเตอร์แบบขนานที่มี 2 พาร์ทิชัน

ภาพประกอบที่ 2-8 แสดงตัวอย่างของระบบแบบ 2 พาร์ทิชันที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากัน โดยที่พาร์ทิชันแรกมีจำนวนหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2 จะมีจำนวนหน่วยประมวลผล 64 โหนด เมื่องานถูกส่งเข้าสู่ระบบจะต้องมารออยู่ที่คิว

แบบส่วนกลางเพื่อรอตัวจัดกำหนดการงานแบบขนานแบบส่วนกลางจัดส่งไปประมวลผล จะเห็นได้ว่างานลำดับแรกของคิวต้องการหน่วยประมวลผล 16 โหนด ดังนั้นงานลำดับแรกสามารถประมวลผลได้ทั้ง 2 พาร์ทิชัน ซึ่งตัวจัดกำหนดการงานแบบขนานจะเลือกเพียงพาร์ทิชันเดียวเท่านั้นสำหรับงานลำดับแรก ส่วนงานในลำดับที่สองของคิวจะสามารถประมวลผลได้เฉพาะพาร์ทิชันที่ 2 เท่านั้นเพราะงานในลำดับที่สองต้องการหน่วยประมวลผลทั้งหมดของพาร์ทิชันที่ 2 ส่วนงานในลำดับที่ 3 ของคิวจะสามารถประมวลผลได้ทั้ง 2 พาร์ทิชันเช่นเดียวกับงานลำดับแรกของคิว ซึ่งงานในลำดับที่ 3 จะประมวลผลที่พาร์ทิชันใดจะขึ้นอยู่กับนโยบายการเลือกพาร์ทิชันที่ระบบกำหนดไว้



ภาพประกอบ 2-8 การส่งงานของตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานแบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากัน

ในปัจจุบันระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันมีการใช้ตัวจัดกำหนดการงานแบบขนาน และคิวหลายแบบ [18] เช่น การใช้ตัวจัดกำหนดการงานแบบขนานเฉพาะภายในพาร์ทิชัน (Local Scheduler) ร่วมกับการใช้คิวแบบภายในพาร์ทิชัน (Local Queue) เป็นต้น ซึ่งระบบลักษณะนี้จะมีคิวหลายระดับ อาจทำให้เกิดความยุ่งยาก และซับซ้อนในการจัดการทรัพยากรได้ งานวิจัยอีกจำนวนหนึ่งได้ศึกษาเฉพาะระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันที่มีการใช้ตัวจัดกำหนดการงานแบบขนานแบบส่วนกลาง และคิวแบบส่วนกลาง [19], [20] ซึ่งระบบดังกล่าวมีข้อดีในเรื่องของการจัดการ เนื่องจากเป็นคิวเดียว และการจัดการต่าง ๆ ทำจากศูนย์กลาง เพื่อเป็นการทดสอบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบ

หลายพาร์ทิชันในเบื้องต้นจะทำการศึกษาเพียงระบบที่ใช้ตัวจัดกำหนดการงานแบบขนานแบบส่วนกลาง และคิวแบบส่วนกลางเท่านั้น

บทที่ 3
การออกแบบและวิธีทดสอบระบบ

ในบทนี้จะกล่าวถึงวิธีการปรับปรุงนโยบายแบบใช้เป้าหมายเป็นหลักให้สามารถทำงานได้บนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน และกล่าวถึงการทดสอบนโยบายดังกล่าว ซึ่งประกอบด้วยโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน คุณลักษณะของเวิร์คโหลดไฟล์ และขั้นตอนการทดสอบประสิทธิภาพนโยบาย

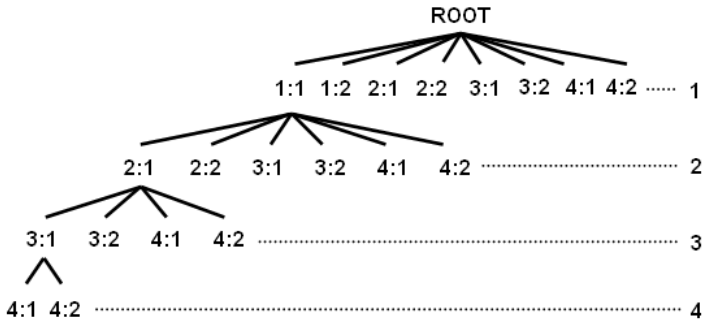
3.1 การจัดกำหนดการงานแบบขนานโดยใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน

ในหัวข้อที่ 2.3 ได้กล่าวถึงตัวจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบพาร์ทิชันเดียว แต่งานวิจัยชิ้นนี้จะออกแบบ และทดสอบการใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน ดังนั้นตัวจัดกำหนดการงานแบบขนานจำเป็นต้องบอกได้ว่างานแต่ละงานในคิวจะเริ่มประมวลผลเมื่อใด และต้องไปประมวลผลที่พาร์ทิชันไหน ซึ่งบนระบบคอมพิวเตอร์แบบขนานแบบพาร์ทิชันเดียวตัวจัดกำหนดการงานแบบขนานจำเป็นต้องบอกเพียงว่างานแต่ละงานในคิวจะเริ่มประมวลผลได้เมื่อใดเท่านั้น การเลือกพาร์ทิชันสำหรับงานแต่ละงานจึงเป็นคำถามที่เพิ่มขึ้นมาของตัวจัดกำหนดการงานแบบขนานเมื่อทำงานบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน ในงานวิจัยชิ้นนี้เสนอแนวทางการออกแบบนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน 2 วิธีคือ การผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ (Large Search Space) และการใช้ฮิวริสทิคในการเลือกพาร์ทิชัน (Small Search Space) ซึ่งจะกล่าวโดยละเอียดในหัวข้อ 3.1.1 และ 3.1.2 ตามลำดับ

3.1.1 การผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้

โครงสร้างต้นไม้ที่ใช้สำหรับการค้นหาการกำหนดการที่ดีบนระบบคอมพิวเตอร์แบบขนานแบบพาร์ทิชันเดียวนั้น แต่ละโหนดในโครงสร้างต้นไม้จะมีข้อมูลเพียงอย่างเดียวคือหมายเลขของงาน (Job Id) ที่ตัวจัดกำหนดการจะพิจารณาแต่การผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้จะนำเอาหมายเลขของงาน และพาร์ทิชันที่งานไปประมวลผลมารวมอยู่ในโหนดเดียวกันทำให้แต่ละโหนดจะมีข้อมูล 2 ส่วนด้วยกัน นอกจากนั้นจำนวนโหนดในแต่ละระดับความ

ลึกจะเพิ่มขึ้นจากจำนวนโหนดบนระบบพาร์ทิชันเดียวเท่ากับจำนวนพาร์ทิชันที่ระบบใช้ ดังภาพประกอบที่ 3-1 แสดงตัวอย่างโครงสร้างต้นไม้แบบผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้บนระบบ 2 พาร์ทิชัน โดยเป็นตัวอย่างของระบบขณะมีงานรออยู่ 4 งาน เส้นจุดด้านขวาแสดงถึงระดับความลึกของโครงสร้างต้นไม้จะเห็นได้ว่ามีตั้งแต่ระดับความลึกที่ 1 ถึง 4 ทุกโหนดที่ระดับความลึก 1 แสดงไว้ในภาพประกอบที่ 3-1 เนื่องจากโครงสร้างต้นไม้มีขนาดใหญ่มากที่ระดับความลึก 2 ถึง 4 จึงแสดงเฉพาะโครงสร้างต้นไม้ย่อยแรกด้านซ้ายสุดเท่านั้นเพื่อให้เห็นชัดเจน จะเห็นได้ว่าที่ระดับความลึกเดียวกันเมื่อเปรียบเทียบกับภาพประกอบที่ 2-6 (โครงสร้างต้นไม้ของระบบแบบพาร์ทิชันเดียว) จำนวนโหนดจะเพิ่มขึ้นจากเดิมถึง 2 เท่า เพราะว่าโครงสร้างต้นไม้ใหม่นี้ผนวกตัวเลือกพาร์ทิชันไว้ด้วย เนื่องจากมี 2 พาร์ทิชันให้เลือกทำให้จำนวนโหนดในแต่ละระดับความลึกเพิ่มขึ้นเป็นสองเท่า



ภาพประกอบ 3-1 โครงสร้างต้นไม้แบบผนวกการเลือกพาร์ทิชันขณะมีงานรออยู่ 4 งาน

ในภาพประกอบที่ 3-1 ใช้การจัดเรียงงานในคิวแบบเอฟซีเอฟเอสเพื่อสร้างโครงสร้างต้นไม้สำหรับการค้นหา วิธีการจัดเรียงงานในคิวเพื่อสร้างโครงสร้างต้นไม้สำหรับการค้นหานั้นได้กล่าวไว้แล้วในหัวข้อ 2.3.1 เนื่องจากโครงสร้างต้นไม้มีการจัดเรียงงานในคิวแบบเอฟซีเอฟเอสทำให้เส้นทางที่ได้จากการค้นหาจะเลือกพาร์ทิชันแรกเสมอ อีกทั้งโครงสร้างต้นไม้จากการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้นั้นจะสร้างโครงสร้างต้นไม้ที่ใหญ่มากทำให้เสียเวลาการค้นหา ดังนั้นควรเลือกพาร์ทิชันก่อนที่จะสร้างโครงสร้างต้นไม้เพื่อลดขนาดของโครงสร้างต้นไม้ ซึ่งการเลือกพาร์ทิชันควรจะเลือกอย่างไร เลือกพาร์ทิชันใดถึงจะเหมาะสมจึงเป็นปัญหาอย่างมาก การใช้ฮิวริสทิคในการเลือกพาร์ทิชันเป็นวิธีหนึ่งที่สามารถช่วยแก้ปัญหาการเลือกพาร์ทิชันได้ ในหัวข้อถัดไปจะกล่าวถึงการใช้ฮิวริสทิคในการเลือกพาร์ทิชัน

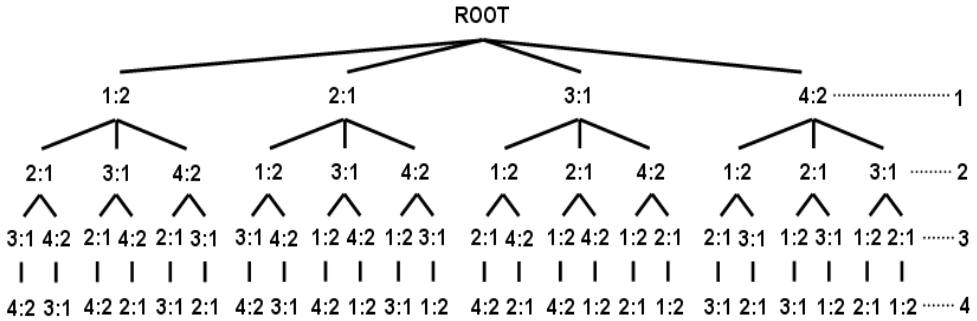
3.1.2 การใช้ฮิวริสทิคในการเลือกพาร์ทิชัน

ดังที่ได้กล่าวไว้แล้วในหัวข้อ 3.1.1 การผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้จะทำให้โครงสร้างต้นไม้มีขนาดใหญ่มาก และอาจทำให้ไม่สามารถค้นหาเส้นทางได้ครบทั้งหมดของโครงสร้างต้นไม้ อีกทั้งทำให้เสียเวลาในการค้นหาเส้นทาง ดังนั้นในหัวข้อนี้จะนำเสนอวิธีการที่สองสำหรับนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันคือการใช้ฮิวริสทิคในการเลือกพาร์ทิชัน ซึ่งจะเลือกพาร์ทิชันไว้ก่อนที่จะสร้างโครงสร้างต้นไม้ทำให้โครงสร้างต้นไม้ที่ได้มีขนาดเท่ากับโครงสร้างต้นไม้สำหรับการค้นหาแบบพาร์ทิชันเดียวในขณะที่แต่ละโหนดของต้นไม้จะยังคงมีข้อมูล 2 ส่วนเหมือนกับการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้

ในงานวิจัยชิ้นนี้จะใช้ฮิวริสทิคแบบเบสฟิท (Best-fit) ซึ่งเป็นวิธีที่ง่าย และใช้เวลาในการคำนวณน้อยเพียง $O(n)$ โดยเบสฟิทจะส่งงานไปที่พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเหลือพอดีกับความต้องการหน่วยประมวลผลของงานนั้นมากที่สุด ในกรณีที่มีพาร์ทิชันที่พอดีกับงานมากกว่า 1 พาร์ทิชัน เบสฟิทจะเลือกพาร์ทิชันที่ทำให้งานดังกล่าวสามารถเริ่มต้นประมวลผลได้เร็วที่สุด หากมีพาร์ทิชันที่ทำให้งานเริ่มต้นประมวลผลได้เท่ากันอีก เบสฟิทก็จะเลือกพาร์ทิชันจากลำดับของพาร์ทิชันเช่น ที่เวลา t_1 ตัวจัดกำหนดการงานแบบขนานเลือกงานที่ต้องการหน่วยประมวลผล 6 โหนดออกมาจากคิวเพื่อนำมาสร้างโครงสร้างต้นไม้ และที่เวลา t_1 พาร์ทิชันที่ 1 เหลือหน่วยประมวลผล 9 โหนด พาร์ทิชันที่ 2 เหลือหน่วยประมวลผล 7 โหนด พาร์ทิชันที่ 3 ไม่เหลือหน่วยประมวลผลเลยแต่ที่เวลา t_5 จะเหลือหน่วยประมวลผล 6 โหนด ภายใต้ฮิวริสทิคแบบเบสฟิทจะต้องจัดให้งานดังกล่าวไปที่พาร์ทิชันที่ 2 เพราะจากทั้ง 3 พาร์ทิชันมีเพียงพาร์ทิชันที่ 1 และ 2 เท่านั้นที่สามารถเริ่มต้นประมวลผลงานได้ที่เวลา t_1 แต่ที่พาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผลเหลือพอดีกับงานดังกล่าวมากกว่าพาร์ทิชันที่ 1 เมื่อเลือกพาร์ทิชันให้กับงานได้แล้วต่อไปตัวจัดกำหนดการงานแบบขนานก็จะนำหมายเลขของงาน และหมายเลขพาร์ทิชันที่เบสฟิทกำหนดให้ไปสร้างเป็นโหนดของโครงสร้างต้นไม้สำหรับการค้นหาดังภาพประกอบที่ 3-2

ภาพประกอบที่ 3-2 แสดงตัวอย่างของโครงสร้างต้นไม้ของงาน 4 งาน ภายใต้การใช้ฮิวริสทิคในการเลือกพาร์ทิชันบนระบบคอมพิวเตอร์แบบขนานแบบ 2 พาร์ทิชัน เส้นจุดด้านขวาแสดงถึงระดับความลึกของโครงสร้างต้นไม้จะเห็นได้ว่ามีตั้งแต่ระดับความลึกที่ 1 ถึง 4 และโหนดแต่ละโหนดยังคงมีข้อมูลอยู่ 2 ส่วนเช่นเดียวกับการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ แต่ขนาดของโครงสร้างต้นไม้จะเล็กกว่าเพราะงานแต่ละงานถูกกำหนดพาร์ทิชันไว้ตั้งแต่ก่อนจะสร้างโครงสร้างต้นไม้จึงทำให้จำนวนโหนดที่ระดับความลึกที่ 1 เมื่อเปรียบเทียบกับภาพประกอบที่ 3-1 (การผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้) ซึ่งขนาดของโครงสร้างต้นไม้จะเท่ากับ

โครงสร้างต้นไม้ของระบบคอมพิวเตอร์แบบขนานแบบพาร์ทิชันเดียว (ภาพประกอบที่ 2-5) ดังนั้น จำนวนพาร์ทิชัน และขนาดของต้นไม้จะไม่มีผลต่อการค้นหาเหมือนกับการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ ในบางกรณีการใช้ฮิวริสทิคในการเลือกพาร์ทิชันอาจมีประสิทธิภาพแย่กว่าการให้ระบบเป็นตัวเลือกพาร์ทิชันก็เป็นได้ ซึ่งวิธีการเลือกพาร์ทิชันนั้นจะกล่าวโดยละเอียดในหัวข้อ 3.1.3 ทั้งนี้ผลกระทบของการเลือกพาร์ทิชันต่อประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักจะเป็นงานที่จะดำเนินการในอนาคต



ภาพประกอบ 3-2 โครงสร้างต้นไม้ของงาน 4 งาน ภายใต้การใช้ฮิวริสทิคในการเลือกพาร์ทิชัน

3.1.3 การเลือกพาร์ทิชัน

ในหัวข้อที่ 3.1.2 ได้กล่าวถึงการทดสอบนโยบายแบบใช้เป้าหมายเป็นหลักโดยใช้ฮิวริสทิคในการเลือกพาร์ทิชันไว้ก่อนที่จะสร้างโครงสร้างต้นไม้ จะเห็นได้ว่าการเลือกใช้ฮิวริสทิคมีผลต่อประสิทธิภาพของระบบมากเพราะฮิวริสทิคที่เลือกใช้คือวิธีการเลือกพาร์ทิชันให้แก่งานแต่ละงาน หากฮิวริสทิคที่เลือกใช้กำหนดให้งานประมวลผลที่พาร์ทิชันใดพาร์ทิชันหนึ่งมากกว่าอีกพาร์ทิชันหนึ่งก็จะทำให้พาร์ทิชันที่ต้องประมวลผลงานจำนวนมากมีโหลด (Load) สูงกว่าพาร์ทิชันที่ได้รับงานไปประมวลผลน้อย ทำให้ระบบไม่สมดุล (Load Imbalance) และส่งผลต่อประสิทธิภาพของระบบ ดังนั้นการเลือกพาร์ทิชันที่ดีต้องทำให้งานแต่ละงานสามารถกระจายไปประมวลผลได้ทุกพาร์ทิชันใกล้เคียงกันมากที่สุด ในงานวิจัย [18] ได้นำเสนอการใช้ฮิวริสทิคเพื่อเลือกพาร์ทิชันที่จะต้องไปประมวลผลโดยพิจารณาจากทรัพยากรของระบบ และการประมาณค่าการรอ โดยนำเสนอการเลือกไว้ 4 แบบคือ

1. เลือกพาร์ทิชันที่ทำให้งานมีค่าการรอน้อยที่สุด (The Less – Job Wait Time) วิธีดังกล่าวจะคำนวณค่าการรอของงานเมื่อไปประมวลผลที่พาร์ทิชันที่เลือกไว้ โดยเลือกพาร์ทิชันจากค่าการรอที่ได้น้อยที่สุดเมื่องานไปประมวลผลที่พาร์ทิชันนั้น จะเห็นได้ว่าวิธีการ

เลือกโดยใช้ค่าการรอน้อยที่สุดนี้เหมือนกับการใช้ฮิวริสทิคแบบเบสฟิทเพราะเบสฟิทจะเลือกพาร์ทิชันที่ทำให้งานสามารถเริ่มต้นประมวลผลได้เร็วที่สุดก่อน ในกรณีที่มีพาร์ทิชันที่พอดีกับงานดังกล่าวมากกว่า 1 พาร์ทิชัน เบสฟิทจะเลือกพาร์ทิชันที่ทำให้งานเริ่มประมวลผลได้เร็วที่สุด กล่าวคือเลือกพาร์ทิชันที่ทำให้ค่าการรอของงานน้อยลงนั้นเอง ในงานงานวิจัย [18] ได้นำเสนอว่าการเลือกพาร์ทิชันโดยทำให้งานมีค่าการรอน้อยที่สุดจะทำให้ค่าการรอของงาน การใช้ทรัพยากรของระบบ จำนวนงาน และขนาดของงานแต่ละพาร์ทิชันดีขึ้น

2. เลือกพาร์ทิชันที่มีจำนวนงานในคิวน้อยที่สุด (The Less – Jobs In Queue) วิธีดังกล่าวจะคำนวณจำนวนงานในคิวแบบภายในพาร์ทิชัน ถ้าพาร์ทิชันใดมีจำนวนงานในคิวน้อยที่สุดก็จะกำหนดพาร์ทิชันนั้นให้กับงานดังกล่าว
3. เลือกพาร์ทิชันที่มีจำนวนงานประมวลผลอยู่น้อยที่สุด (The Less - Work Left) วิธีดังกล่าวจะคำนวณจำนวนงานที่กำลังประมวลผลของแต่ละพาร์ทิชัน ถ้าพาร์ทิชันใดมีจำนวนงานที่กำลังประมวลผลอยู่น้อยที่สุดก็จะกำหนดพาร์ทิชันนั้นให้กับงานดังกล่าว
4. เลือกพาร์ทิชันที่มีจำนวนการส่งงานไปน้อยที่สุด (The Less – Submitted Jobs) วิธีดังกล่าวจะเลือกพาร์ทิชันจากจำนวนงานที่ถูกกำหนดไปแต่ละพาร์ทิชัน ถ้าพาร์ทิชันใดมีจำนวนงานที่ถูกกำหนดให้น้อยที่สุดก็จะเลือกพาร์ทิชันนั้นให้กับงานดังกล่าว ซึ่งวิธีนี้ไม่ได้พิจารณาถึงข้อมูลของงานแต่ละงาน (ระยะเวลาการประมวลผล จำนวนความต้องการหน่วยประมวลผล) จึงทำให้ประสิทธิภาพของการเลือกพาร์ทิชันแบบนี้แย่ที่สุดใน 4 แบบที่นำเสนอมา

จะเห็นได้ว่าการเลือกพาร์ทิชันนั้นสามารถทำได้หลายวิธีขึ้นอยู่กับว่าจะให้ความสนใจไปที่เรื่องใด ในงานวิจัยชิ้นนี้จะไม่ศึกษาถึงในส่วนของวิธีการพาร์ทิชันแบบใดจึงจะเหมาะสมต่อระบบ แต่จะใช้เฉพาะเบสฟิทเท่านั้นในการเลือกพาร์ทิชัน ในหัวข้อถัดไปจะกล่าวถึงวิธีการทดสอบประสิทธิภาพของนโยบาย

3.2 การทดสอบประสิทธิภาพของนโยบาย

การทดสอบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันจะทดสอบโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน โดยใช้เวิร์คโหลดไฟล์จากระบบคอมพิวเตอร์แบบขนานเชิงพาณิชย์ที่ใช้ทำงานอยู่จริงมาเป็นอินพุท (Input) ของระบบ และวัดประสิทธิภาพของนโยบายโดยใช้ค่าสูงสุดของการรอ ค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (Bounded Slowdown) ค่า

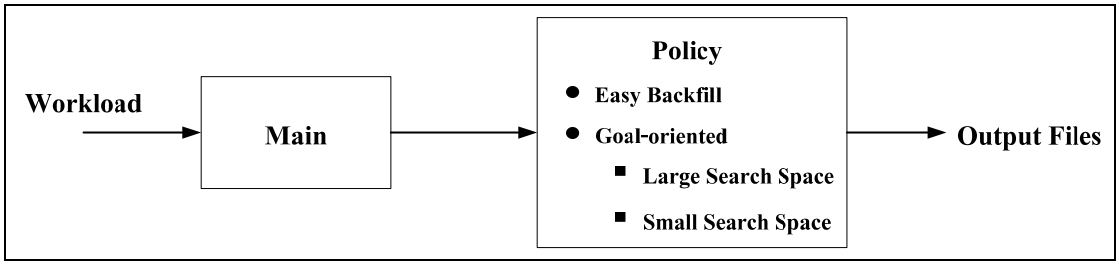
98 เปอร์เซ็นไทล์ของการรอ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง งานวิจัยชิ้นนี้ทำการทดสอบบนเครื่องคอมพิวเตอร์ส่วนบุคคลที่มีหน่วยประมวลผลแบบ 4 คอร์ (Quad Core) ความเร็วหน่วยประมวลผลที่ 2.66 GHz และมีหน่วยความจำ 4 GB บนระบบปฏิบัติการ Microsoft Window XP

3.2.1 โปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน

นโยบายทุกชนิดจะถูกทดสอบประสิทธิภาพบนโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน ซึ่งถูกพัฒนาโดยภาษาจาวา 1.4 (Java Version 1.4) เพื่อให้โปรแกรมจำลองดังกล่าวทำงานใกล้เคียงกับระบบจริงมากที่สุด ในการจำลองแต่ละครั้งจะมีช่วงเวลาที่เป็นวอร์มอัพ (Warm-up) และคูลดาวน์ (Cool-down) ดังภาพประกอบที่ 3-3 ส่วนวอร์มอัพเพื่อทำให้ระบบเริ่มต้นจากการมีงานอยู่ในระบบโดยจะทำการป้อนงานของช่วง 1 สัปดาห์ก่อนหน้าเดือนที่ต้องการประเมินผลประสิทธิภาพเข้ามาประมวลผลก่อน และตามด้วยงานในเดือนที่ต้องการประเมินประสิทธิภาพ หลังจากนั้นหากงานในช่วงเดือนที่ประเมินประสิทธิภาพยังอยู่ในระบบก็จะส่งในเดือนถัดไปเข้ามาในระบบเพื่อทำให้ระบบสมจริง และระบบจะหยุดเมื่องานของเดือนที่ประเมินผลออกจากระบบไปหมดแล้ว

Warm Up	ช่วงเดือนที่ทำการประเมินประสิทธิภาพ	Cool Down

ภาพประกอบ 3-3 การนำงานเข้าสู่ระบบของโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน



ภาพประกอบ 3-4 โครงสร้างของโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน

โดยตัวโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานจะถูกแบ่งออกเป็น 2 ส่วนใหญ่ ๆ ดังภาพประกอบที่ 3-4 คือ ส่วนโปรแกรมหลัก (Main Program) และส่วนนโยบายที่ใช้ (Policy)

3.2.1.1 ส่วนโปรแกรมหลัก

ส่วนโปรแกรมหลักจะทำหน้าที่รับข้อมูลอินพุทซึ่งก็คือเวิร์คโหลดไฟล์ ผู้ใช้สามารถกำหนดค่าเริ่มต้นต่าง ๆ ของระบบได้โดยผ่านทางพารามิเตอร์ขณะสั่งรันโปรแกรม ซึ่งพารามิเตอร์ที่สามารถกำหนดได้มีดังนี้

- กำหนดเวิร์คโหลดไฟล์ที่จะใช้ และรูปแบบของเวิร์คโหลดไฟล์ ในโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานที่พัฒนาขึ้นจะมีการใช้เวิร์คโหลดไฟล์ 2 รูปแบบ ซึ่งจะกล่าวโดยละเอียดในภาคผนวก ก
- นโยบายที่จะใช้ในการทดสอบมี 2 แบบคือ
  - นโยบายแบบเอฟซีเอฟเอส และเอลเอกซ์เอฟร่วมกับเทคนิคอีซีแบ็กฟิล
  - นโยบายแบบใช้เป้าหมายเป็นหลักจะต้องมีการกำหนดพารามิเตอร์ 3 ค่าด้วยกันคือ
    - การจัดเรียงงานจากซ้ายไปขวาเพื่อสร้างต้นไม้ ซึ่งโปรแกรมจำลองที่พัฒนาขึ้นมานี้ จะสามารถเลือกการจัดเรียงงานได้ 2 แบบคือ เอฟซีเอฟเอส และเอลเอกซ์เอฟ
    - การกำหนดรูปแบบของเป้าหมาย และเป้าหมายที่จะใช้ ซึ่งโปรแกรมจำลองที่พัฒนาขึ้นมานี้สามารถเลือกรูปแบบของเป้าหมายได้ 2 รูปแบบคือ รูปแบบเล็กซิคอล และรูปแบบอีควลเทรดออฟ ในส่วนของเป้าหมายจะขึ้นอยู่กับรูปแบบที่ใช้ ซึ่งดูเพิ่มเติมได้ในภาคผนวก ก
    - การจำกัดจำนวนโหนดที่ค้นหา ซึ่งจำนวนโหนดที่ใช้ในงานวิจัยชิ้นนี้อยู่ระหว่าง 2,000 ถึง 128,000 โหนด
- กำหนดจำนวนพาร์ทิชันที่จะใช้ ซึ่งระบบรองรับได้ไม่เกิน 5 พาร์ทิชัน และกำหนดจำนวนหน่วยประมวลของแต่ละพาร์ทิชันได้
- สามารถกำหนดได้ว่าจะให้ตัวจัดกำหนดการงานแบบขนานใช้ค่าระยะเวลาการประมวลผลจริง (Actual Runtime) หรือค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด
- กำหนดช่วงของงานที่จะนำมาประมวลผลได้ ซึ่งในงานวิจัยนี้จะวัดประสิทธิภาพรายเดือน

จะเห็นได้ว่าในส่วนของโปรแกรมหลักจะใช้สำหรับกำหนดค่าเริ่มต้นทั่วไปที่จำเป็นต่อระบบคอมพิวเตอร์แบบขนานที่จำลองขึ้นมา ส่วนการจัดกำหนดการงานแบบขนานตาม

นโยบาย และการตัดสินใจของตัวจัดกำหนดการงานแบบขนานนั้นจะขึ้นอยู่กับนโยบายที่ใช้ ซึ่งจะกล่าวในหัวข้อ 3.2.1.2

3.2.1.2 ส่วนนโยบายที่ใช้

โปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานที่พัฒนาไว้ในงานวิจัยชิ้นนี้รองรับนโยบาย 2 ตัวด้วยกันดังภาพประกอบที่ 3-4 คือ

1. นโยบายแบบเอฟซีเอฟเอส และเอลเอกซ์เอฟร่วมกับเทคนิคแบ็กฟิล
2. นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน ซึ่งแบ่งออกเป็น 2 วิธีคือ
   2.1 การผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้
   2.2 การใช้ฮิวริสทิคในการเลือกพาร์ทิชัน

ในส่วนนโยบายที่จะใช้สามารถพัฒนาเพิ่มได้เอง โดยสร้างเป็นคลาส (Class) แยกออกมาซึ่งจะไม่กระทบในส่วนอื่น ๆ หลังจากงานถูกประมวลผลเสร็จ เอาพุทที่ได้จากโปรแกรมจะเป็นข้อความจำนวน 9 คอลัมน์ ซึ่งจะประกอบไปด้วยข้อมูลของงานนั้น ๆ เช่นเริ่มประมวลผลเมื่อใด เป็นต้น ทั้งนี้รายละเอียดแสดงไว้ในภาคผนวก ก หัวข้อถัดไปจะกล่าวถึงเวิร์คโหลดไฟล์ซึ่งเป็นอินพุทของโปรแกรมจำลองที่พัฒนาขึ้น

3.2.2 เวิร์คโหลดไฟล์ (Workload File)

เวิร์คโหลดไฟล์เป็นอินพุทสำหรับโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน ในงานวิจัยชิ้นนี้นำเวิร์คโหลดไฟล์มาจากศูนย์ซุปเปอร์คอมพิวเตอร์ มหาวิทยาลัยอิลลินอยส์ (NCSA Illinois) ระบบของศูนย์เป็นลินุกซ์คลัสเตอร์ (Linux Cluster) ที่ใช้หน่วยประมวลผล Intel Itanium (IA-64) แบบ 2 คอร์ (Dual Processor) โดยมีหน่วยประมวลผลทั้งหมด 128 โหนด เวิร์คโหลดไฟล์ที่นำมาทดสอบอยู่ในช่วงเดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 โดยแบ่งได้ 2 ช่วงตามข้อกำหนดของระยะเวลาในการประมวลผลคือช่วงเดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนพฤศจิกายนปี ค.ศ. 2003 งานจะประมวลผลได้ไม่เกิน 12 ชั่วโมง ส่วนในช่วงเดือนธันวาคมปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 งานจะประมวลผลได้ไม่เกิน 24 ชั่วโมง ในตารางที่ 3-1 ถึง 3-3 แสดงเปอร์เซ็นต์ความต้องการหน่วยประมวลผล (Processor Demand) ของแต่ละเดือน จำนวนผู้ใช้ที่ส่งงานมาในเดือนนั้น ๆ จำนวนงานในแต่ละเดือน และขนาดของงาน (Job size) ในแต่ละเดือนโดยแสดงเป็นค่าเฉลี่ย ค่ามัธยฐาน (Median) และค่าสูงสุด ที่คอลัมน์ขนาดของงาน N จะหมายถึงความต้องการหน่วยประมวลผล ถ้า

N มีค่ามากจะหมายถึงงานที่มีขนาดใหญ่ และ T หมายถึงเวลาที่งานใช้ในการประมวลผล ถ้า T มีค่ามากจะหมายถึงงานที่ยาว

ตารางที่ 3-1 ข้อมูลโดยสรุปแต่ละเดือนของเวิร์คโหลดไฟล์ต้นฉบับ

Month	Proc. Demand	#User	#Jobs	Job size (NT)		
				Avg.	Med.	Max.
6/03	81.96%	73	2191	34.47	0.82	960.00
7/03	89.07%	68	1400	60.59	1.25	1536.00
8/03	79.08%	73	3221	23.38	0.00	1536.00
9/03	71.90%	74	3057	21.68	0.10	912.00
10/03	70.84%	75	4149	16.26	0.36	912.00
11/03	72.69%	81	3443	19.46	0.74	1536.00
12/03	74.26%	61	3521	20.08	1.08	1151.99
1/04	73.11%	53	3156	22.06	5.14	1920.00
2/04	73.77%	73	3969	16.56	0.33	1824.00
3/04	74.83%	70	3466	20.56	0.01	1832.82

ตารางที่ 3-1 แสดงข้อมูลโดยสรุปแต่ละเดือนของเวิร์คโหลดไฟล์ต้นฉบับ เปอร์เซ็นต์ความต้องการหน่วยประมวลผลแต่ละเดือน ส่วนใหญ่ประมาณ 70-80% ยกเว้นในเดือนกรกฎาคมปี ค.ศ. 2003 ซึ่งมีเปอร์เซ็นต์ความต้องการหน่วยประมวลผลสูงที่สุด (89.08%) ในส่วนของจำนวนงานในแต่ละเดือนส่วนใหญ่จะอยู่ที่ประมาณ 3,000 - 4,000 งาน ยกเว้นจำนวนงานของเดือนมิถุนายนปี ค.ศ. 2003 และเดือนกรกฎาคมปี ค.ศ. 2003 ที่น้อยกว่า 3,000 งาน ส่วนขนาดของงานจะแสดงด้วยค่าเฉลี่ย ค่ามัธยฐาน และค่าสูงสุด ซึ่งจะเห็นได้ว่าค่าเฉลี่ยของขนาดของงานส่วนมากประมาณ 16 – 23 โหนดแต่ที่เดือนมิถุนายนปี ค.ศ. 2003 และเดือนกรกฎาคมปี ค.ศ. 2003 จะมีค่าเฉลี่ยของขนาดของงานสูงกว่าเดือนอื่น ๆ กล่าวคือในเดือนดังกล่าวมีบางช่วงที่ค่าเฉลี่ยของขนาดของงานมีค่ามากทำให้ค่าเฉลี่ยที่ได้สูงตามไปด้วย ค่ามัธยฐานของขนาดของงานส่วนใหญ่จะไม่เกิน 1 โหนดยกเว้นเดือนมกราคมปี ค.ศ. 2004 ซึ่งมีค่ามัธยฐานถึง 5.14 โหนดแสดงว่างานเดือนมกราคมปี ค.ศ. 2004 ส่วนใหญ่จะเป็นงานที่ใหญ่ และยาว

จากที่ได้กล่าวไว้แล้วว่าการทดสอบนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน จะมีการทดสอบทั้งแบบแต่ละพาร์ทิชันมีจำนวนหน่วย

ประมวลผลเท่ากัน และแบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากันบนระบบ 2 พาร์ทิชันที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เกิน 64 โหนด ทั้งนี้แต่ละงานจะต้องสามารถทำงานได้ใน 1 พาร์ทิชัน ทำให้ต้องมีการปรับเวิร์คโหลดให้แต่ละงานสามารถใช้หน่วยประมวลผลได้ไม่เกิน 64 โหนด การปรับงานแต่ละงานในเวิร์คโหลดให้ใช้หน่วยประมวลผลได้ไม่เกิน 64 โหนดดังสมการที่ 2-1

$$N_{new} = \min(64, N_{old}) \quad (2\text{-}1)$$

- N_{new} คือค่าความต้องการหน่วยประมวลผลของงานที่ได้ใหม่
- N_{old} คือค่าความต้องการหน่วยประมวลผลของงานเดิม

ตารางที่ 3-2 ข้อมูลโดยสรุปแต่ละเดือนของเวิร์คโหลดไฟล์หลังจากแก้ไขโดยลดหน่วยประมวลผลลงไม่เกิน 64 สำหรับ 2 พาร์ทิชัน

Month	Proc. Demand		#User	#Jobs	Job size (NT)		
	N64, 64	N32, 64			Avg.	Med.	Max.
6/03	80.32%	107.09%	73	2191	33.78	0.82	768.00
7/03	73.68%	98.25%	68	1400	50.12	1.17	768.00
8/03	69.71%	92.94%	73	3221	20.61	0.00	768.00
9/03	66.61%	88.82%	74	3057	20.08	0.10	768.00
10/03	69.53%	92.71%	75	4149	15.96	0.36	768.00
11/03	67.74%	90.32%	81	3443	18.13	0.74	768.00
12/03	72.94%	97.25%	61	3521	19.73	1.08	1151.99
1/04	69.94%	93.26%	53	3156	21.11	5.14	1536.00
2/04	71.13%	94.84%	73	3969	15.97	0.33	1536.00
3/04	71.95%	95.94%	70	3466	19.77	0.01	1536.00

ตารางที่ 3-2 แสดงข้อมูลโดยสรุปแต่ละเดือนของเวิร์คโหลดไฟล์หลังจากแก้ไขเพื่อใช้ในการทดสอบประสิทธิภาพของนโยบายที่นำเสนอ และนโยบายอื่น ๆ บนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เกิน 64 โหนด หลังจากแก้ไขเวิร์คโหลดไฟล์แล้วจะเห็นได้ว่าในแต่ละเดือนยังคงมีแนวโน้มของขนาดของงานเหมือนในตารางที่ 3-1 แต่ในบางเดือนจะมีขนาดของ

งานเพิ่มขึ้นเล็กน้อย ส่วนเดือนมกราคมปี ค.ศ. 2004 ยังคงมีงานขนาดใหญ่ และยาวเช่นเดิม แต่เปอร์เซ็นต์ความต้องการหน่วยประมวลผลแต่ละเดือนส่วนใหญ่จะลดลงจากตารางที่ 3-1

นอกจากการทดสอบระบบแบบ 2 พาร์ทิชันที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เกิน 64 โหนด ในงานวิจัยชิ้นนี้ยังทดสอบระบบโดยเปลี่ยนจำนวนพาร์ทิชันตั้งแต่ 3 ถึง 5 พาร์ทิชันและแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เกิน 32 โหนด ซึ่งงานแต่ละงานจะต้องทำงานได้ใน 1 พาร์ทิชัน ทำให้ต้องมีการปรับเวิร์คโหลดให้งานแต่ละงานใช้หน่วยประมวลผลได้ไม่เกิน 32 โหนดดังสมการที่ 2-2

$$N_{new} = \min(32,\ N_{old}) \tag{2-2}$$

ตารางที่ 3-3 ข้อมูลโดยสรุปแต่ละเดือนของเวิร์คโหลดไฟล์หลังจากแก้ไขโดยลดหน่วยประมวลผลลงไม่เกิน 32 สำหรับ 3, 4 และ 5 พาร์ทิชัน

Month	Proc. Demand			#User	#Jobs	Job size (NT)		
	3 Part.	4 Part.	5 Part.			Avg.	Med.	Max.
6/03	89.75%	67.31%	53.85%	73	2191	28.31	0.82	384.00
7/03	66.78%	50.09%	40.07%	68	1400	34.07	1.09	384.00
8/03	71.54%	53.65%	42.92%	73	3221	15.86	0.00	384.00
9/03	70.09%	52.56%	42.05%	74	3057	15.85	0.10	384.00
10/03	79.92%	59.94%	47.95%	75	4149	13.76	0.35	384.00
11/03	68.63%	51.47%	41.18%	81	3443	13.78	0.68	384.00
12/03	80.65%	60.48%	48.39%	61	3521	16.36	1.07	768.00
1/04	80.35%	60.26%	48.21%	53	3156	18.18	5.13	768.00
2/04	85.56%	64.17%	51.34%	73	3969	14.40	0.33	768.00
3/04	87.25%	65.44%	52.35%	70	3466	17.98	0.01	768.00

ตารางที่ 3-3 แสดงข้อมูลโดยสรุปแต่ละเดือนของเวิร์คโหลดไฟล์หลังจากแก้ไขเพื่อใช้ในการทดสอบประสิทธิภาพของนโยบายที่นำเสนอ และนโยบายอื่น ๆ บนระบบ 3, 4 และ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันที่ 32 โหนด หลังจากแก้ไขเวิร์คโหลดไฟล์แล้ว จะเห็นได้ว่าในแต่ละเดือนยังคงมีแนวโน้มของขนาดของงานในแต่ละเดือนเหมือนในตารางที่ 3-1 และ 3-2 แต่ขนาดของงานในแต่ละเดือนลดลงจากตารางที่ 3-1 อย่างเห็นได้ชัด ในส่วนของเปอร์เซ็นต์

ความต้องการหน่วยประมวลผลแต่ละเดือน จะเห็นได้ว่าเมื่อจำนวนพาร์ทิชันเพิ่มขึ้นเปอร์เซ็นต์ความต้องการหน่วยประมวลผลแต่ละเดือนจะลดลง เนื่องมาจากเมื่อเพิ่มจำนวนพาร์ทิชันระบบก็จะมีทรัพยากรเพิ่มขึ้นด้วย

รูปแบบของเวิร์คโหลดไฟล์มีหลายแบบซึ่งสามารถดูได้จาก [21] แต่โปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานที่พัฒนาขึ้นในหัวข้อ 3.2.1 สามารถใช้เวิร์คโหลดไฟล์ได้ 2 รูปแบบเท่านั้นคือ แบบเอสดับบิวเอฟ (Standard Workload Format : SWF) [22] มีทั้งหมด 18 คอลัมน์ และแบบซิมพลิไฟด์ (Simplified) มีทั้งหมด 6 คอลัมน์ แต่เมื่อโปรแกรมทำงานจริงจะใช้เพียง 6 คอลัมน์ที่เป็นแบบซิมพลิไฟด์ ซึ่งตัวเวิร์คโหลดไฟล์ที่ใช้ทดสอบนำมาจากศูนย์ซุปเปอร์คอมพิวเตอร์เป็นแบบเอสดับบิวเอฟ ดังนั้นโปรแกรมต้องแปลงเวิร์คโหลดไฟล์ที่เข้ามาให้อยู่ในรูปแบบซิมพลิไฟด์ ซึ่งจะกล่าวโดยละเอียดในส่วนภาคผนวก ก

3.2.3 ค่าที่ใช้วัดประสิทธิภาพ (Measurement)

ในปัจจุบันการวัดประสิทธิภาพของระบบคอมพิวเตอร์แบบขนานจะขึ้นอยู่กับเวิร์คโหลดที่นำมาใช้ [23] และ [24] เนื่องจากลักษณะของงานในเวิร์คโหลดทำให้การวัดประสิทธิภาพไม่มีค่าที่ใช้วัดที่แน่นอน ส่วนใหญ่จะใช้ค่าเวลาตอบสนอง (Response Time) เวลาการรอ การใช้ประโยชน์ ปริมาณงาน เป็นต้น ต่อมามีผู้เสนอการใช้ค่าสโลว์ดาวน์ (Slowdown) [25] ดังสมการที่ 2-3

$$Slowdown = \frac{wait\ time+runtime}{runtime} \tag{2-3}$$

จากสมการจะเห็นได้ว่าค่าสโลว์ดาวน์จะขึ้นกับค่าการรอ ซึ่งค่าการรอจะคำนวนจากเวลาที่งานถูกส่งเข้ามาในระบบแล้วยังไม่ได้รับการประมวลผล ดังนั้นถ้าค่าการรอสูงหรือมากแสดงว่างานต้องรออยู่ในระบบนาน หากงานต้องรออยู่ในระบบนานมาก ๆ ก็อาจเกิดปัญหาอดหยากได้ ดังนั้นในการใช้ค่าการรอหรือค่าสโลว์ดาวน์ในการวัดประสิทธิภาพของระบบควรจะได้ค่าต่ำหรือน้อยจึงจะทำให้ระบบมีประสิทธิภาพดี ข้อเสียของค่าสโลว์ดาวน์คืองานสั้นจะมีผลกระทบต่อระบบตัวอย่างเช่น งานใช้เวลาการประมวลผล 1 วินาที และมีค่าการรอเท่ากับ 20 นาที จะทำให้ได้ค่าสโลว์ดาวน์เท่ากับ 1201 วินาที จะเห็นได้ว่างานสั้นที่มีค่าการรอนานส่งผลต่อค่าสโลว์ดาวน์ที่ได้ งานวิจัย [26] ได้นำเสนอการลดผลกระทบของงานสั้นต่อค่าสโลว์ดาวน์โดยการกำหนดงานสั้นที่จะนำมาคำนวณค่าสโลว์ดาวน์โดยเรียกว่า เบาน์เด็ดสโลว์ดาวน์ (Bounded

Slowdown) ซึ่งงานที่มีค่าระยะเวลาการประมวลผลไม่เกินค่าเบาน์เด็ด (Bounded) จะใช้ค่าเบาน์เด็ดเป็นตัวหารดังสมการที่ 2-4

$$Bounded\ Slowdown = \frac{wait\ time + runtime}{\max(runtime, bounded)} \quad (2\text{-}4)$$

ตัวอย่างเช่นงานที่ต้องใช้เวลาการประมวลผล 10 วินาที และมีค่าการรอเท่ากับ 1200 วินาที (20 นาที) จะทำให้ได้ค่าสโลว์ดาวน์เท่ากับ 121 วินาที ในทางกลับกันเมื่อคิดค่าเบาน์เด็ดสโลว์ดาวน์โดยกำหนดค่าเบาน์เด็ดเป็น 60 วินาที (1 นาที) จะได้ค่าเบาน์เด็ดสโลว์ดาวน์เท่ากับ 20.17 วินาที จะเห็นได้ว่าน้อยกว่าค่าสโลว์ดาวน์ถึง 100.83 วินาที งานวิจัย [27] ได้ทดสอบการใช้ค่าต่าง ๆ ในการวัดประสิทธิภาพคือ ค่าเวลาตอบสนอง ค่าสโลว์ดาวน์ และค่าเบาน์เด็ดสโลว์ดาวน์ ผลที่ได้แสดงให้เห็นว่าการใช้ค่าเบาน์เด็ดสโลว์ดาวน์ในการวัดประสิทธิภาพของระบบสามารถแสดงประสิทธิภาพของระบบที่แท้จริงได้ดีกว่าค่าอื่น ๆ

ในงานวิจัยชิ้นนี้ต้องการให้ระบบมีประสิทธิภาพ 2 อย่างด้วยกันคือ ไม่ต้องการให้งานรอนานจนเกิดปัญหาอดอยาก และต้องการให้งานสั้นไม่ต้องรอนาน จะเห็นได้ว่าความต้องการทั้ง 2 ประการขัดแย้งกัน ซึ่งความต้องการแต่ละตัวต้องใช้ตัววัดประสิทธิภาพที่แตกต่างกันกล่าวคือ การป้องกันปัญหาอดอยากสามารถทำได้โดยการทำให้จำนวนงานที่รอเกินกำหนดมีจำนวนน้อยที่สุด ค่าสูงสุดของการรอน้อยที่สุด และค่าเฉลี่ยการรอน้อยที่สุด ส่วนการทำให้งานสั้นไม่ต้องรอนานนั้นสามารถทำได้โดยทำให้ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์น้อยที่สุด ดังนั้นการวัดประสิทธิภาพของนโยบายในงานวิจัยชิ้นนี้จะใช้ 3 ค่าคือ ค่าการรอ จำนวนงานที่รอเกินกำหนด และค่าเบาน์เด็ดสโลว์ดาวน์ ในการวัดประสิทธิภาพด้วยค่าจำนวนงานที่รอเกินกำหนดจะใช้จำนวนงานที่รอเกิน 24 ชั่วโมง ส่วนค่าเบาน์เด็ดสโลว์ดาวน์จะกำหนดค่าเบาน์เด็ดเป็น 1 นาที

บทที่ 4
ผลการทดสอบและวิเคราะห์ผล

ในบทนี้จะกล่าวถึงผลการทดสอบ และวิเคราะห์ผลการทำงานของนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบหลายพาร์ทิชันในกรณีต่าง ๆ ทดสอบโดยใช้โปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน

ตารางที่ 4-1 สัญลักษณ์ต่าง ๆ

สัญลักษณ์	ความหมาย
FCFS-backfill	นโยบายแบบเอฟซีเอฟเอสร่วมกับเทคนิคแบ็กฟิล
LXF-backfill	นโยบายแบบเอลเอกซ์เอฟร่วมกับเทคนิคแบ็กฟิล
Tradeoff	นโยบายแบบใช้เป้าหมายเป็นหลักที่ใช้รูปแบบของเป้าหมายเป็นอีควลเทรดออฟ
Goal-oriented	นโยบายแบบใช้เป้าหมายเป็นหลัก
hr	ชั่วโมง
s	วินาที
98th wait	98 เปอร์เซ็นไทล์ของการรอ
Tw	ค่าผลรวมการรอที่มากเกินกำหนด
avgX	ค่าเฉลี่ยสโลว์ดาวน์
#jobs wait > 24hrs	จำนวนงานที่รอมากกว่า 24 ชั่วโมง
NT	ขนาดของงาน
N	จำนวนหน่วยประมวลผลที่ใช้ของงาน
T	ระยะเวลาที่ใช้จริงในการประมวลผลของงาน
R	ระยะเวลาที่ใช้ในการประมวลผลของงานซึ่งถูกประมาณค่าโดยผู้ใช้
large search space	การผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้
small search space	การใช้ฮิวริสทิคในการเลือกพาร์ทิชัน

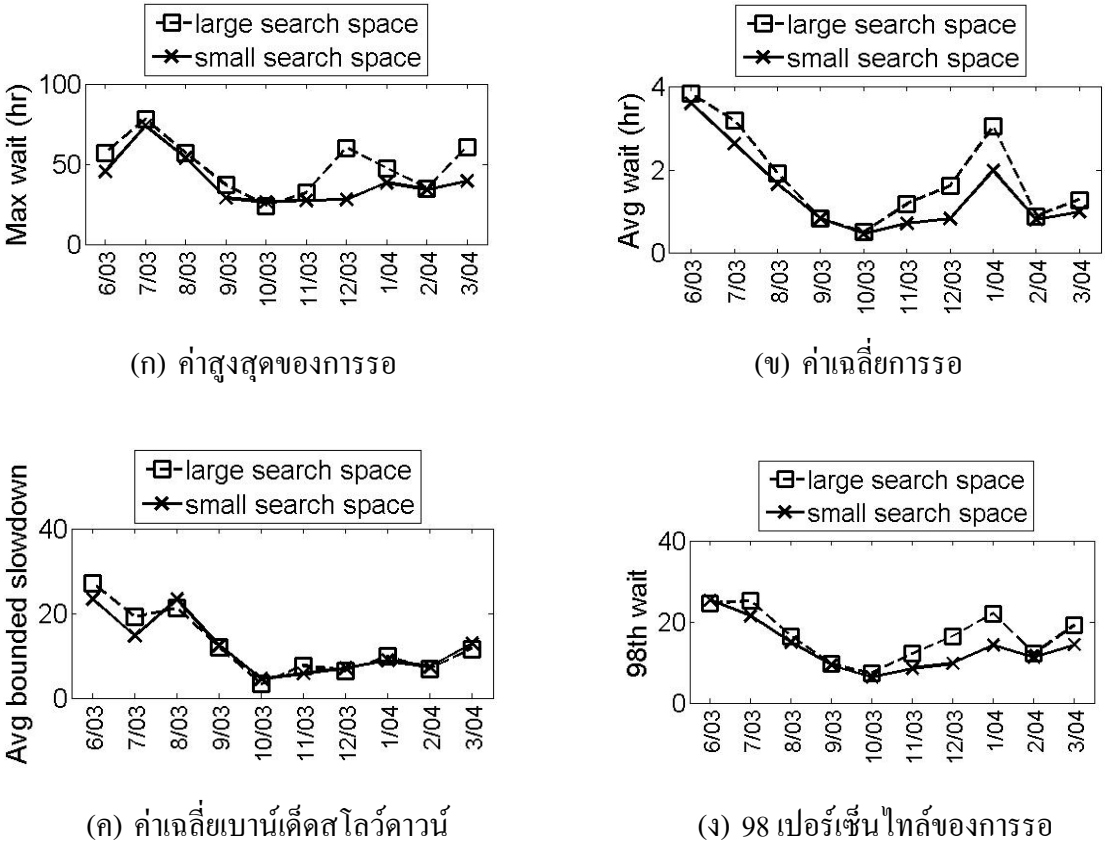
การเปรียบเทียบประสิทธิภาพจะเริ่มจากเปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักที่ออกแบบไว้ในหัวข้อ 3.1 หลังจากนั้นจะทำการเปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบหลายพาร์ทิชันทั้งที่เป็นแบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากัน และแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากัน นอกจากนี้ยังมีการนำเสนอประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักเมื่อเพิ่มระยะเวลาในการค้นหา เมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด และเมื่อจำนวนพาร์ทิชันเพิ่มขึ้น ในตารางที่ 4-1 แสดงสัญลักษณ์ต่าง ๆ และความหมายที่ใช้สำหรับบทนี้

4.1 เปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักด้วยการผนวกการเลือกพาร์ทิชันกับการใช้ฮิวริสทิคในการเลือกพาร์ทิชัน

หัวข้อที่ 3.1 เสนอแนวทางการปรับนโยบายแบบใช้เป้าหมายเป็นหลักให้ทำงานบนระบบหลายพาร์ทิชัน 2 วิธีคือ การผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ และการใช้ฮิวริสทิคในการเลือกพาร์ทิชัน ในหัวข้อนี้จะเปรียบเทียบและนำเสนอผลการใช้นโยบายทั้งสองบนระบบคอมพิวเตอร์แบบขนาน 2 พาร์ทิชันแบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผล 64 โหนดเท่ากัน การทดสอบในหัวข้อนี้ใช้เวิร์คโหลดไฟล์ซึ่งแสดงรายละเอียดไว้ในตารางที่ 3-2

ภาพประกอบที่ 4-1 แสดงผลการเปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักด้วยการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้กับการใช้ฮิวริสทิคในการเลือกพาร์ทิชันแบบเบสฟิท โดยเปรียบเทียบประสิทธิภาพจากระบบคอมพิวเตอร์แบบขนานที่มีจำนวนหน่วยประมวลผลเท่ากันที่ 64 โหนด และแสดงเป็นกราฟประสิทธิภาพแต่ละเดือนจำนวน 10 เดือนตั้งแต่เดือนมิถุนายนปีค.ศ. 2003 ถึงเดือนมีนาคมปีค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งแสดงประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่าการใช้ฮิวริสทิคในการเลือกพาร์ทิชันให้ค่าสูงสุดการรอ (ภาพประกอบที่ 4-1 (ก)) ค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-1 (ข)) และค่า 98 เปอร์เซ็นไทล์ของการรอ (ภาพประกอบที่ 4-1 (ง)) ดีกว่าหรือเท่ากับการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ทุกเดือน แต่ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ภาพประกอบที่ 4-1 (ค)) ของการใช้ฮิวริสทิคในการเลือกพาร์ทิชันให้ค่าดีกว่าการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ในเดือนส่วนใหญ่ ยกเว้นเดือนสิงหาคมปี ค.ศ. 2003 และเดือนมกราคมปี ค.ศ. 2003 เนื่องมาจากเดือนสิงหาคมปี ค.ศ. 2003 และเดือนมกราคมปี ค.ศ. 2004 มีงานสั้น ๆ บางกลุ่มจากการใช้ฮิวริสทิคในการเลือกพาร์ทิชันต้องรอนานกว่าการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ ซึ่งส่งผลกระทบต่อระบบเพียงเล็กน้อยเท่านั้น จะเห็นได้จาก

ค่าเฉลี่ยการรอจากการใช้ฮิวริสทิคในการเลือกพาร์ทิชันให้ค่าดีกว่าการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้



(ก) ค่าสูงสุดของการรอ

(ข) ค่าเฉลี่ยการรอ

(ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์

(ง) 98 เปอร์เซ็นไทล์ของการรอ

ภาพประกอบ 4-1 ประสิทธิภาพการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ และการใช้ฮิวริสทิคในการเลือกพาร์ทิชัน

การใช้ฮิวริสทิคในการเลือกพาร์ทิชันให้ค่าดีกว่าการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้เพราะว่าขนาดพื้นที่ในการค้นหาของการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ใหญ่กว่าการใช้ฮิวริสทิคในการเลือกพาร์ทิชันเท่ากับจำนวนเท่าของจำนวนพาร์ทิชันของระบบ ส่งผลให้การค้นหาด้วยการผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้อาจจะค้นหาไม่ครอบคลุมพื้นที่หลากหลายได้เท่ากับการใช้ฮิวริสทิคในการเลือกพาร์ทิชัน เนื่องจากครบกำหนดเวลาก่อน

หัวข้อต่อไปจะทำการเปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ซึ่งการทดสอบนโยบายแบบใช้เป้าหมายเป็นหลักในหัวข้อถัด ๆ ไปจะใช้ฮิวริสทิคในการเลือกพาร์ทิชันเท่านั้น

4.2 เปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน

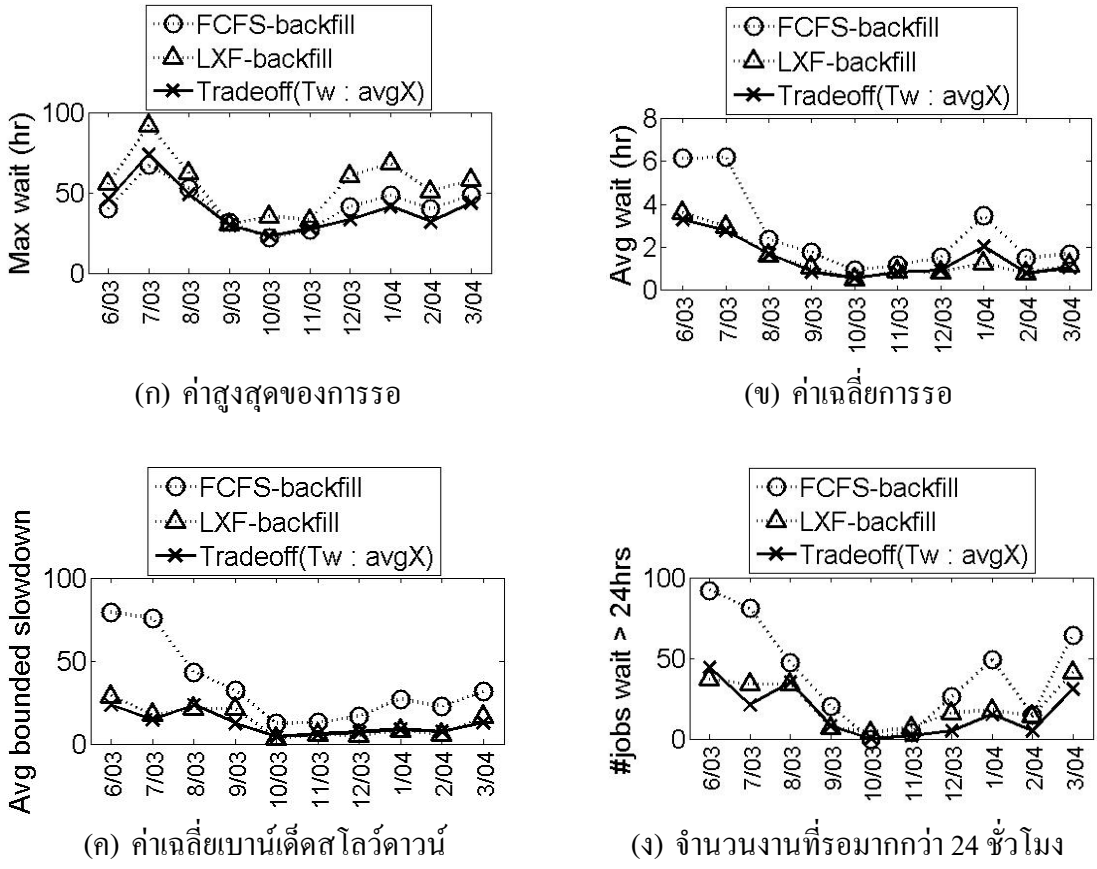
ในหัวข้อนี้จะแสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน โดยทำการทดสอบบนระบบ 2 พาร์ทิชันทั้งแบบที่มีจำนวนหน่วยประมวลผลเท่ากัน (หัวข้อ 4.2.1) และแบบที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน (หัวข้อ 4.2.2) และทดสอบแบบหลายพาร์ทิชันโดยเพิ่มจำนวนพาร์ทิชันเป็น 3 ถึง 5 พาร์ทิชัน (หัวข้อ 4.2.3) ทั้งนี้นโยบายแบบใช้เป้าหมายเป็นหลักที่นำมาทดสอบคือ แบบการใช้ฮิวริสทิคในการเลือกพาร์ทิชันโดยใช้เบสฟีทเป็นฮิวริสทิคในการเลือกพาร์ทิชัน เนื่องจากให้ประสิทธิภาพการทำงานที่ดีกว่าดังได้แสดงในหัวข้อ 4.1

4.2.1 ประสิทธิภาพบนระบบ 2 พาร์ทิชันแบบจำนวนหน่วยประมวลผลเท่ากัน

หัวข้อนี้จะแสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันคือ 64 โหนด การทดสอบในหัวข้อนี้ใช้เวิร์คโหลดไฟล์ ซึ่งแสดงรายละเอียดไว้ในตารางที่ 3-2 ผลที่ได้จากการทดสอบนโยบายจะแสดงเป็น 3 หัวข้อย่อยคือ ประสิทธิภาพโดยรวมของระบบ (หัวข้อ 4.2.1.1) ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน (หัวข้อ 4.2.1.2) และวิเคราะห์ประสิทธิภาพ (หัวข้อ 4.2.1.3)

4.2.1.1 ประสิทธิภาพโดยรวมของระบบ

หัวข้อนี้จะแสดงการเปรียบเทียบประสิทธิภาพโดยรวมทั้งระบบของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน



(ก) ค่าสูงสุดของการรอ (ข) ค่าเฉลี่ยการรอ

(ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ง) จำนวนงานที่รอมากกว่า 24 ชั่วโมง

ภาพประกอบ 4-2 ประสิทธิภาพบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

ภาพประกอบที่ 4-2 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน ขนาด 64 โหนด และแสดงเป็นกราฟประสิทธิภาพแต่ละเดือนจำนวน 10 เดือนตั้งแต่เดือนมิถุนายนปีค.ศ. 2003 ถึงเดือนมีนาคมปีค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งแสดงประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่านโยบายแบบเอฟซีเอฟเอสแบ็กฟิลให้ค่าสูงสุดการรอดีที่สุด (ภาพประกอบที่ 4-2 (ก)) ในขณะที่นโยบายแบบเอลเอกซ์เอฟแบ็กฟิลให้ค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-2 (ข)) และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ภาพประกอบที่ 4-2 (ค)) ดีที่สุด จะเห็นได้ว่านโยบายทั้งสองตัวทำงานดีในเป้าหมายอันใดอันหนึ่งเท่านั้น แต่นโยบายแบบใช้เป้าหมายเป็นหลักให้ค่าประสิทธิภาพการทำงานทั้งสามดีทั้งหมด เมื่อมองลึกไปถึงจำนวนงานที่ต้องรอนานกว่า 24 ชั่วโมง (ภาพประกอบที่ 4-2 (ง)) จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีเช่นกัน และในบางเดือนประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักดีกว่าประสิทธิภาพที่ดีที่สุดของนโยบายแบบ

ให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเช่น ประสิทธิภาพจากค่าสูงสุดของการรอในเดือนธันวาคมปีค.ศ. 2003

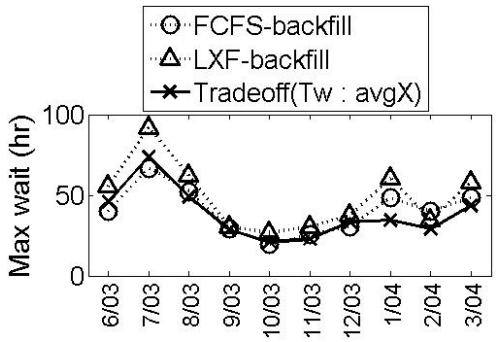
โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-2 นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าสูงสุดการรอ และค่าเฉลี่ยการรอ แม้ว่าตัววัดประสิทธิภาพทั้ง 2 จะขัดแย้งกันก็ตาม

4.2.1.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน

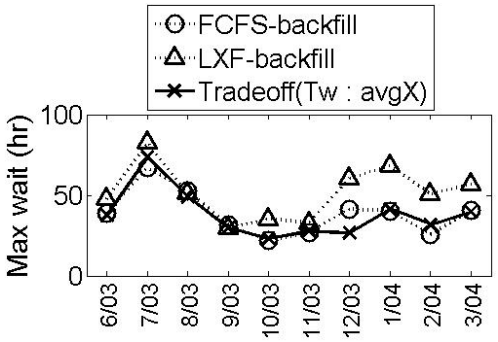
จากหัวข้อ 4.2.1.1 ได้เปรียบเทียบประสิทธิภาพโดยภาพรวมของทั้งระบบ โดยแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ดังนั้นหัวข้อนี้จะวิเคราะห์หาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ประสิทธิภาพของแต่ละนโยบายบนแต่ละพาร์ทิชัน ดังแสดงในภาพประกอบที่ 4-3 ทั้งนี้แนวนอนแสดงเดือนที่วัด และแนวตั้งแสดงประสิทธิภาพต่าง ๆ อันได้แก่ ค่าสูงสุดการรอ ค่าเฉลี่ยการรอ และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แต่ละค่าจะแสดงเป็น 2 ภาพ โดยภาพประกอบที่ 4-3 ก, ค, จ แสดงผลของค่าดังกล่าวของแต่ละนโยบายบนพาร์ทิชันแรก ในขณะที่ภาพประกอบที่ 4-3 ข, ง, ฉ แสดงค่าที่วัดได้ของแต่ละนโยบายบนพาร์ทิชันที่ 2

จากภาพประกอบที่ 4-3 จะเห็นว่าเมื่อวิเคราะห์ประสิทธิภาพแต่ละพาร์ทิชัน นโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลยังคงให้ค่าประสิทธิภาพที่ขัดแย้งกัน กล่าวคือ นโยบายแบบเอฟซีเอฟเอสแบ็กฟิลให้ค่าสูงสุดการรอดีกว่านโยบายแบบเอลเอกซ์เอฟแบ็กฟิล (ภาพประกอบที่ 4-3 (ก, ข)) ในขณะที่นโยบายแบบเอลเอกซ์เอฟแบ็กฟิลให้ค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-3 (ค, ง)) และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ภาพประกอบที่ 4-3 (จ, ฉ)) ดีกว่านโยบายแบบเอฟซีเอฟเอสแบ็กฟิล ส่วนนโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีของทั้ง 3 ประสิทธิภาพที่แสดงบนทั้ง 2 พาร์ทิชัน

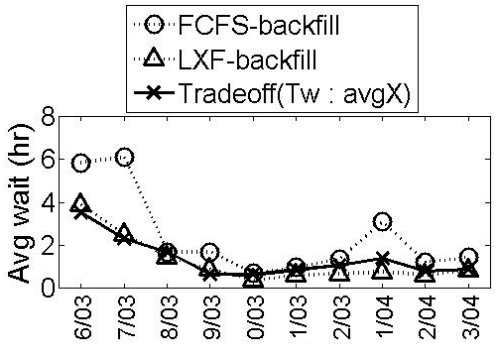
โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-3 เมื่อวิเคราะห์ประสิทธิภาพแต่ละพาร์ทิชันนโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าสูงสุดการรอ และค่าเฉลี่ยการรอ จึงส่งผลให้ประสิทธิภาพโดยภาพรวมของทั้งระบบ (หัวข้อ 4.2.1.1) ดีกว่าด้วย



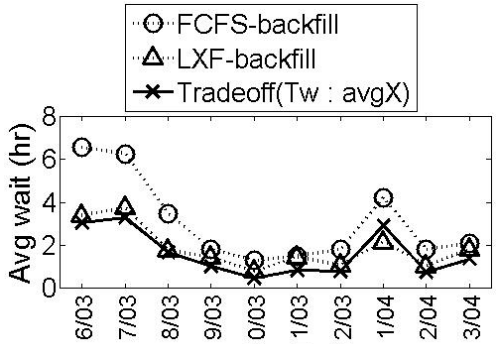
(ก) ค่าสูงสุดของการรอของพาร์ทิชันที่ 1



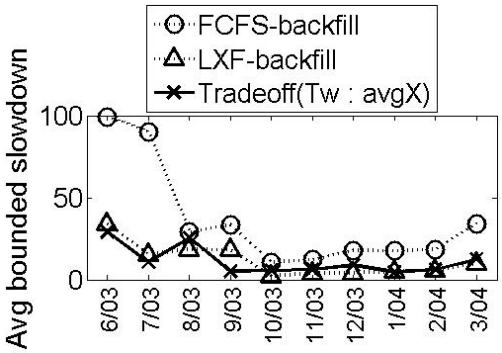
(ข) ค่าสูงสุดของการรอของพาร์ทิชันที่ 2



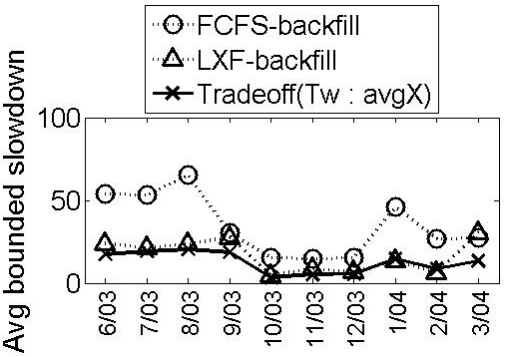
(ค) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 1



(ง) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 2



(จ) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 1



(ฉ) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 2

ภาพประกอบ 4-3 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

4.2.1.3 วิเคราะห์ประสิทธิภาพ

จากหัวข้อที่ 4.2.1.1 และ 4.2.1.2 ก่อนหน้าได้แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล

ในหัวข้อนี้จะตรวจสอบหาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ถึงจำนวนงาน และปริมาณงานในแต่ละพาร์ทิชัน ดังแสดงในตารางที่ 4-2 จำนวนงานที่ถูกส่งไปประมวลผลในแต่ละพาร์ทิชันของแต่ละนโยบาย และตารางที่ 4-3 ลักษณะของงาน (จำนวนหน่วยประมวลผล × เวลาที่ใช้) ซึ่งแสดงค่าเฉลี่ยมีหน่วยเป็น จำนวนหน่วยประมวลผลชั่วโมง (Node-hour) เพื่อให้ง่ายต่อการวิเคราะห์ผลในแต่ละตารางของแต่ละนโยบายจะเพิ่มคอลัมน์ Diff. สำหรับแสดงค่าผลต่างทั้งสองพาร์ทิชันของแต่ละนโยบาย ตัวหนาจะแสดงค่าที่ดีที่สุดของคอลัมน์ Diff. ในแต่ละเดือน

ตารางที่ 4-2 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

Number of jobs									
	FCFS-backfill			LXF-backfill			Goal-oriented		
Partition / Month	1st	2nd	Diff.	1st	2nd	Diff.	1st	2nd	Diff.
6/03	1,235	956	279	1,043	1,148	105	1,138	1,053	**85**
7/03	854	546	308	889	511	378	747	653	**94**
8/03	1,962	1,259	703	1,588	1,633	**45**	1,821	1,400	421
9/03	1,797	1,260	537	2,034	1,023	1,011	1,411	1,646	**235**
10/03	2,524	1,625	899	2,597	1,552	1,045	1,976	2,173	**197**
11/03	2,306	1,137	1,169	2,282	1,161	1,121	2,114	1,329	**785**
12/03	2,134	1,387	747	2,042	1,479	563	1,588	1,933	**345**
1/04	2,128	1,028	1,100	2,076	1,080	996	1,817	1,339	**478**
2/04	1,893	2,076	**183**	2,292	1,677	615	2,200	1,769	431
3/04	2,134	1,332	**802**	2,373	1,093	1,280	2,269	1,197	1,072

จากตารางที่ 4-2 จะเห็นว่านโยบายแบบเอลเอกซ์เอฟแบ็กฟิลมีจำนวนงานประมวลผลบนพาร์ทิชันทั้งสองใกล้เคียงกันมากกว่านโยบายอื่น ๆ เฉพาะเดือนสิงหาคมปี ค.ศ. 2003 ส่วนนโยบายแบบเอฟซีเอฟเอสแบ็กฟิลมีจำนวนงานประมวลผลบนพาร์ทิชันทั้งสองใกล้เคียงกันมากกว่านโยบายอื่น ๆ เฉพาะเดือนกุมภาพันธ์ และมีนาคมปี ค.ศ. 2003 แต่จำนวนงานจาก

นโยบายแบบใช้เป้าหมายเป็นหลักของทั้งสองเดือนนี้ใกล้เคียงกันไม่ต่างจากนโยบายแบบเอฟซีเอฟเอสแบ็กฟิลมากนัก ในเดือนอื่น ๆ นโยบายแบบใช้เป้าหมายเป็นหลักมีจำนวนงานประมวลผลบนพาร์ทิชันทั้งสองใกล้เคียงกันมากกว่านโยบายอื่น ๆ อย่างเห็นได้ชัด เนื่องจากระบบที่ทดสอบเป็นแบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากัน ดังนั้นการกระจายจำนวนงานไปทั้ง 2 พาร์ทิชันแบบเท่า ๆ กันจะช่วยให้ประสิทธิภาพโดยรวมดีขึ้น โดยภาพรวมแล้วจะเห็นได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักสามารถกระจายจำนวนงานไปแต่ละพาร์ทิชันได้เท่า ๆ กันมากกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองที่ใช้ในการทดสอบ

ตารางที่ 4-3 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

Average job size (NT) in node-hour									
	FCFS-backfill			LXF-backfill			Goal-oriented		
Partition / Month	1st	2nd	Diff.	1st	2nd	Diff.	1st	2nd	Diff.
6/03	30.1	38.6	8.5	35.6	32.1	3.5	33.0	34.6	**1.6**
7/03	40.4	65.4	25.0	39.1	69.4	30.3	46.5	54.2	**7.7**
8/03	16.6	26.9	10.3	21.0	20.2	**0.8**	18.2	23.7	5.5
9/03	17.5	23.8	6.3	15.2	29.7	14.5	21.7	18.7	**3.0**
10/03	13.6	19.6	6.0	12.6	21.6	9.0	17.5	14.6	**2.9**
11/03	14.2	26.2	12.0	14.2	25.9	11.7	14.5	23.9	**9.4**
12/03	16.3	25.0	8.7	17.3	23.1	5.8	21.5	18.3	**3.2**
1/04	17.6	28.3	10.7	17.7	27.7	10.0	18.9	24.1	**5.2**
2/04	17.3	14.8	2.5	14.3	18.3	4.0	14.9	17.3	**2.4**
3/04	16.6	24.8	**8.2**	14.9	30.3	15.4	15.6	27.6	12.0

จากตารางที่ 4-3 จะเห็นว่านโยบายแบบเอลเอกซ์เอฟแบ็กฟิลกระจายงานลักษณะต่าง ๆ ไปพาร์ทิชันทั้งสองใกล้เคียงกันมากกว่านโยบายอื่น ๆ เฉพาะเดือนสิงหาคมปี ค.ศ. 2003 และนโยบายแบบเอฟซีเอฟเอสแบ็กฟิลกระจายงานลักษณะต่าง ๆ ไปพาร์ทิชันทั้งสองใกล้เคียงกันมากกว่านโยบายอื่น ๆ เฉพาะเดือนมีนาคมปี ค.ศ. 2004 ส่วนเดือนอื่น ๆ นโยบายแบบใช้เป้าหมาย

เป็นหลักสามารถกระจายงานลักษณะต่าง ๆ ไปพาร์ทิชันทั้งสองใกล้เคียงกันมากกว่านโยบายอื่น ๆ จะเห็นได้ว่าความสามารถในการกระจายงานลักษณะต่าง ๆ ของแต่ละนโยบายจะเหมือนกับความสามารถในการกระจายจำนวนงานที่ตาราง 4-2 กล่าวคือนโยบายแบบใช้เป้าหมายเป็นหลักยังคงสามารถกระจายงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชันได้ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองที่ใช้ในการทดสอบ ซึ่งการกระจายงานขนาดเล็ก และใหญ่จะปะปนกันไปทั้ง 2 พาร์ทิชันจะช่วยให้งานถูกจัดบนพื้นที่ว่างบนทั้ง 2 พาร์ทิชัน การกระทำเช่นนี้จะช่วยให้ประสิทธิภาพโดยรวมของระบบดีขึ้น

โดยสรุปนโยบายแบบใช้เป้าหมายเป็นหลักสามารถกระจายจำนวนงาน และงานลักษณะต่าง ๆ ไปยังพาร์ทิชันทั้งสองเท่า ๆ กัน เนื่องจากทั้งสองพาร์ทิชันมีขนาดเท่ากันการกระจายงานและจำนวนงานลักษณะดังกล่าวช่วยให้นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทุกประสิทธิภาพที่วัดถึงแม้บางประสิทธิภาพจะขัดแย้งกันเองก็ตาม

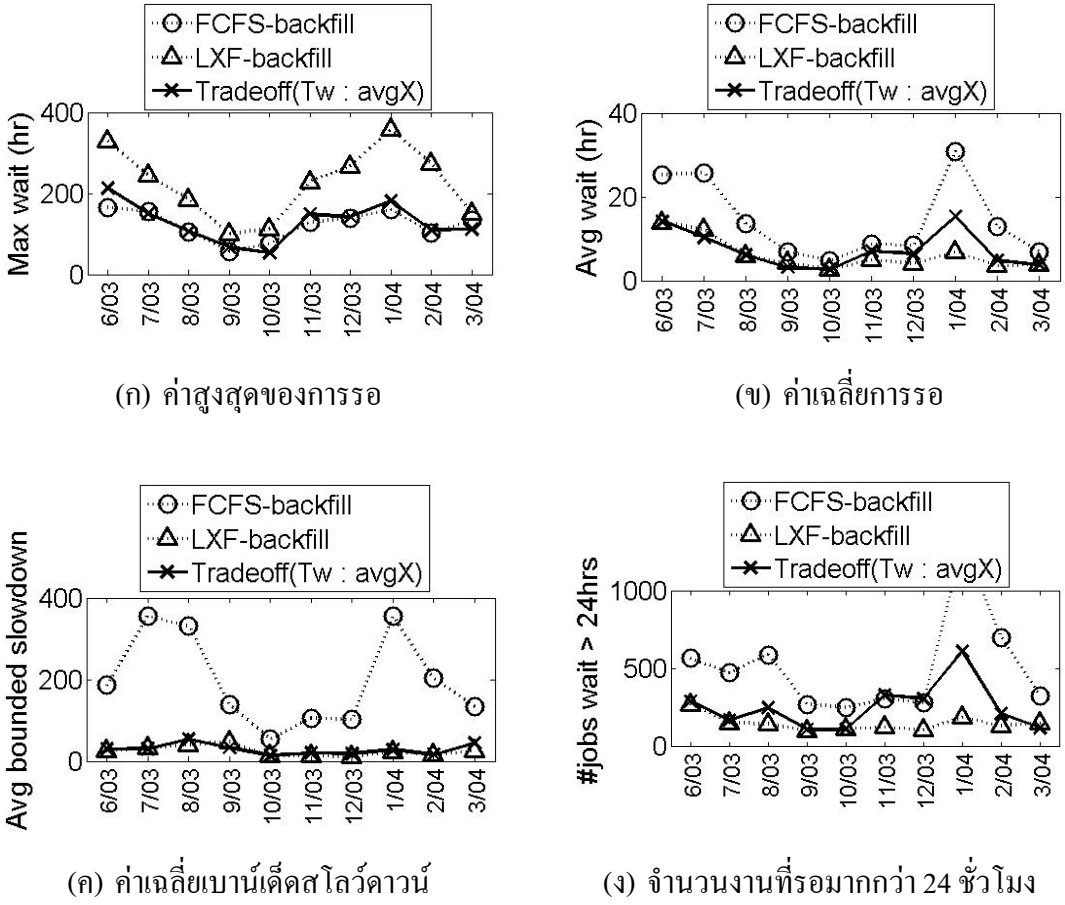
ในหัวข้อต่อไปจะนำเสนอ และวิเคราะห์ประสิทธิภาพทุก ๆ นโยบายบนระบบคอมพิวเตอร์แบบขนานแบบ 2 พาร์ทิชันเมื่อแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากัน

4.2.2 ประสิทธิภาพบนระบบ 2 พาร์ทิชันแบบจำนวนหน่วยประมวลผลไม่เท่ากัน

จากหัวข้อ 4.2.1 แสดงให้เห็นถึงประสิทธิภาพบนระบบคอมพิวเตอร์แบบขนาน 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน ในหัวข้อนี้จะแสดงให้เห็นถึงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน โดยให้พาร์ทิชันแรกมีจำนวนหน่วยประมวลผล 32 โหนดและพาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผล 64 โหนด การทดสอบในหัวข้อนี้ใช้เวิร์คโหลดไฟล์ซึ่งแสดงรายละเอียดไว้ในตารางที่ 3-2 ผลที่ได้จากการทดสอบนโยบายจะแสดงเป็น 3 หัวข้อย่อยคือ ประสิทธิภาพโดยรวมของระบบ (หัวข้อ 4.2.2.1) ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน (หัวข้อ 4.2.2.2) และวิเคราะห์ประสิทธิภาพ (หัวข้อ 4.2.2.3)

4.2.2.1 ประสิทธิภาพโดยรวมของระบบ

หัวข้อนี้จะแสดงการเปรียบเทียบประสิทธิภาพโดยรวมทั้งระบบของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน



(ก) ค่าสูงสุดของการรอ (ข) ค่าเฉลี่ยการรอ

(ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ง) จำนวนงานที่รอมากกว่า 24 ชั่วโมง

ภาพประกอบ 4-4 ประสิทธิภาพบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน

ภาพประกอบที่ 4-4 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน โดยพาร์ทิชันแรกมีจำนวนหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผล 64 โหนด ซึ่งแสดงเป็นกราฟประสิทธิภาพแต่ละเดือนจำนวน 10 เดือนตั้งแต่เดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งเป็นประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่านโยบายแบบเอฟซีเอฟเอสแบ็กฟิลให้ค่าสูงสุดการรอดีที่สุด (ภาพประกอบที่ 4-4 (ก)) ในขณะที่นโยบายแบบเอลเอกซ์เอฟแบ็กฟิลให้ค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-4 (ข)) และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ภาพประกอบที่ 4-4 (ค)) ดีที่สุด จะเห็นได้ว่านโยบายทั้งสองตัวทำงานดีในเป้าหมายอันใดอันหนึ่งเท่านั้น ซึ่งเหมือนประสิทธิภาพจากการทดสอบแบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากัน (หัวข้อ 4.2.1) แต่นโยบายแบบใช้เป้าหมายเป็นหลักให้ค่า

ประสิทธิภาพการทำงานทั้งสามเป้าหมายดีทั้งหมด เมื่อมองลึกไปถึงจำนวนงานที่ต้องรอนานกว่า 24 ชั่วโมง (ภาพประกอบที่ 4-4 (ง)) จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีเช่นกัน

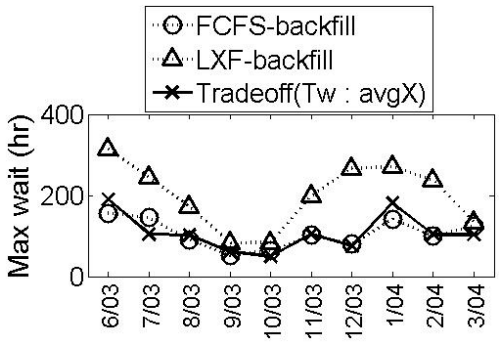
โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-4 นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าสูงสุดการรอ และค่าเฉลี่ยการรอ แม้ว่าตัววัดประสิทธิภาพทั้ง 2 จะขัดแย้งกันก็ตาม

4.2.2.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน

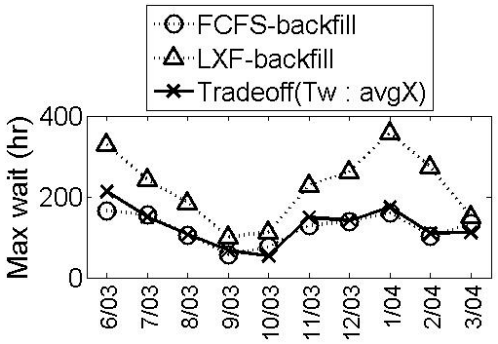
จากหัวข้อ 4.2.2.1 ได้เปรียบเทียบประสิทธิภาพโดยภาพรวมของทั้งระบบ โดยแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ดังนั้นหัวข้อนี้จะวิเคราะห์หาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ประสิทธิภาพของแต่ละนโยบายบนแต่ละพาร์ทิชัน ดังแสดงในภาพประกอบที่ 4-5 ทั้งนี้แนวนอนแสดงเดือนที่วัด และแนวตั้งแสดงประสิทธิภาพต่าง ๆ อันได้แก่ ค่าสูงสุดการรอ ค่าเฉลี่ยการรอ และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แต่ละค่าจะแสดงเป็น 2 ภาพ โดยภาพประกอบที่ 4-5 ก, ค, จ แสดงผลของค่าดังกล่าวของแต่ละนโยบายบนพาร์ทิชันแรก ในขณะที่ภาพประกอบที่ 4-5 ข, ง, ฉ แสดงค่าที่วัดได้ของแต่ละนโยบายบนพาร์ทิชันที่ 2

จากภาพประกอบที่ 4-5 จะเห็นว่าเมื่อแยกแสดงผลแต่ละพาร์ทิชัน นโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองยังคงให้ค่าที่ขัดแย้งกัน ส่วนนโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีของทั้ง 3 ประสิทธิภาพที่แสดงบนทั้ง 2 พาร์ทิชันเช่นเดียวกับแบบรวมทั้งระบบในหัวข้อ 4.2.2.1

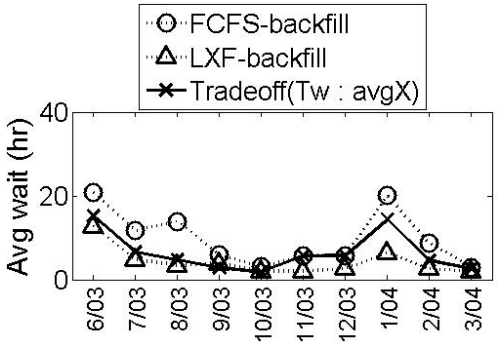
จากประสิทธิภาพที่แสดงในภาพประกอบที่ 4-4 และ 4-5 จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล แม้จะทำงานบนระบบหลายพาร์ทิชันที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากันก็ตาม



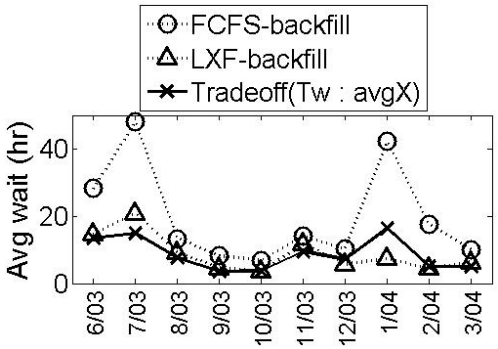
(ก) ค่าสูงสุดของการรอของพาร์ทิชันที่ 1



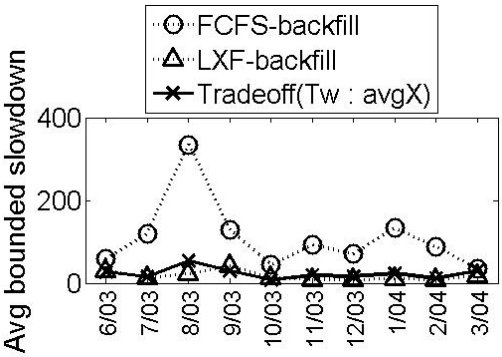
(ข) ค่าสูงสุดของการรอของพาร์ทิชันที่ 2



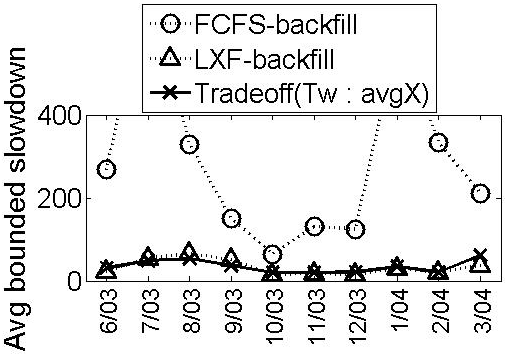
(ค) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 1



(ง) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 2



(จ) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 1



(ฉ) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 2

ภาพประกอบ 4-5 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน

4.2.2.3 วิเคราะห์ประสิทธิภาพ

จากหัวข้อที่ 4.2.2.1 และ 4.2.2.2 ก่อนหน้าได้แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ในหัวข้อนี้จะตรวจสอบหาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ถึงจำนวนงาน และปริมาณงานในแต่ละพาร์ทิชันโดยตารางที่ 4-4 แสดงจำนวนงานที่ถูกส่งไปประมวลผลในแต่ละพาร์ทิชันของแต่ละนโยบาย ตารางที่ 4-5 แสดงลักษณะของงาน (จำนวนหน่วยประมวลผล × เวลาที่ใช้) ซึ่งแสดงค่าเฉลี่ยมีหน่วยเป็น จำนวนหน่วยประมวลผลชั่วโมง และตารางที่ 4-6 แสดงค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานมีหน่วยเป็น จำนวนหน่วยประมวลผล (Node) เนื่องจากระบบที่ใช้ทดสอบเป็นระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน โดยที่พาร์ทิชันแรกมีจำนวนหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผล 64 โหนด นั่นคือพาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผลเป็น 2 เท่าของพาร์ทิชันแรก ดังนั้นในตารางที่ 4-3 ถึง 4-5 จะเพิ่มคอลัมน์ 1st/2nd ซึ่งใช้แสดงอัตราส่วนระหว่างพาร์ทิชันที่ 1 กับพาร์ทิชันที่ 2 ซึ่งพาร์ทิชันแรกควรจะประมวลผลงานเล็ก และสั้นจำนวนมาก ส่วนพาร์ทิชันที่ 2 ควรจะประมวลผลงานใหญ่ และยาวร่วมกับงานเล็กบางส่วนเพื่อให้ระบบมีการกระจายโหลดที่สมดุล ดังนั้นค่าอัตราส่วนจำนวนงานระหว่างพาร์ทิชันที่ 1 กับพาร์ทิชันที่ 2 ควรจะใกล้เคียง 1 แต่ค่าอัตราส่วนลักษณะของงาน และค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานระหว่างพาร์ทิชันที่ 1 กับพาร์ทิชันที่ 2 ควรจะใกล้เคียง 0.5 โดยที่ค่าอัตราส่วนจำนวนงานที่เป็นตัวหนาจะหมายถึงค่าที่ใกล้เคียง 1 ส่วนค่าอัตราส่วนลักษณะของงาน และค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานที่เป็นตัวหนาจะหมายถึงค่าที่ใกล้เคียง 0.5

จากตารางที่ 4-4 จะเห็นได้ว่าในเดือนมิถุนายนปี ค.ศ. 2003 จำนวนงานแต่ละพาร์ทิชันของทั้ง 3 นโยบายใกล้เคียงกันมาก เดือนตุลาคมปี ค.ศ. 2003 และเดือนกุมภาพันธ์ปี ค.ศ. 2004 จำนวนงานแต่ละพาร์ทิชันของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลใกล้เคียงกันมากกว่านโยบายแบบใช้เป้าหมายเป็นหลัก ส่วนนโยบายแบบใช้เป้าหมายเป็นหลักจะมีจำนวนงานแต่ละพาร์ทิชันใกล้เคียงกันมากกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองในเดือนสิงหาคม เดือนกันยายน และเดือนธันวาคมปี ค.ศ. 2003 เดือนมกราคมปี ค.ศ. 2004 ซึ่งจะเห็นได้ชัดเจนที่เดือนสิงหาคม และเดือนกันยายนปี ค.ศ. 2003

จากการวิเคราะห์จำนวนงานแต่ละพาร์ทิชันสรุปได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักสามารถกระจายจำนวนงานไปแต่ละพาร์ทิชันได้ใกล้เคียงกัน ต่อไปจะวิเคราะห์ถึงลักษณะของงานในแต่ละพาร์ทิชันดังตารางที่ 4-5

ตารางที่ 4-4 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน

Number of jobs									
	FCFS-backfill			LXF-backfill			Goal-oriented		
Partition / Month	1st	2nd	1st/2nd	1st	2nd	1st/2nd	1st	2nd	1st/2nd
6/03	875	1,316	0.66	854	1,337	0.64	848	1,343	0.63
7/03	862	538	1.60	753	647	1.16	780	620	1.26
8/03	1,255	1,966	0.64	1,814	1,407	1.29	1,671	1,550	**1.08**
9/03	1,704	1,353	1.26	1,298	1,759	0.74	1,586	1,471	**1.08**
10/03	2,326	1,823	1.28	2,134	2,015	**1.06**	2,275	1,874	1.21
11/03	2,227	1,216	1.83	2,408	1,035	2.33	2,147	1,296	1.66
12/03	1,538	1,983	0.78	1,810	1,711	**1.06**	1,786	1,735	**1.03**
1/04	1,611	1,545	**1.04**	1,300	1,856	0.70	1,658	1,498	**1.11**
2/04	2,098	1,871	**1.12**	2,001	1,968	**1.02**	1,764	2,205	0.80
3/04	1,534	1,932	0.79	2,064	1,402	1.47	1,906	1,560	1.22

จากตารางที่ 4-5 จะเห็นได้ว่าทั้ง 3 นโยบายในทุกเดือนงานที่พาร์ทิชันแรกจะเป็นงานขนาดเล็กกว่าพาร์ทิชันที่ 2 ในบางเดือนงานที่พาร์ทิชันแรกมีขนาดเล็กกว่าพาร์ทิชันที่ 2 ถึง 4 เท่าเช่น เดือนกรกฎาคมปี ค.ศ. 2003 แต่จะเห็นได้ว่าจำนวนงานที่ตาราง 4-4 ของเดือนนี้พาร์ทิชันแรกมีจำนวนมากกว่าพาร์ทิชันที่ 2 นั่นแสดงว่างานที่ประมวลผลบนพาร์ทิชันแรกเป็นงานขนาดเล็กเกือบทั้งหมด เมื่อวิเคราะห์ถึงอัตราส่วนของขนาดงานในพาร์ทิชันที่ 1 ต่อพาร์ทิชันที่ 2 แล้วพบว่าเดือนกันยายน เดือนตุลาคม และเดือนธันวาคมปี ค.ศ. 2003 เดือนกุมภาพันธ์ปี ค.ศ. 2004 ลักษณะของงานจากนโยบายแบบเอลเอกซ์เอฟแบ็กฟิลบนพาร์ทิชันแรกน้อยกว่าพาร์ทิชันที่ 2 ประมาณครึ่งหนึ่ง ส่วนนโยบายแบบเอฟซีเอฟเอสแบ็กฟิลจะมีขนาดของงานบนพาร์ทิชันแรกน้อยกว่าพาร์ทิชันที่ 2 ประมาณครึ่งหนึ่งเฉพาะเดือนธันวาคมปี ค.ศ. 2003 เดือนมกราคม และเดือนกุมภาพันธ์ปี ค.ศ. 2004 แต่นโยบายแบบใช้เป้าหมายเป็นหลักจะมีขนาดของงานบนพาร์ทิชันแรกน้อยกว่าพาร์ทิชันที่ 2 ประมาณครึ่งหนึ่งเกือบทุกเดือน ยกเว้นเดือนมิถุนายน เดือนกรกฎาคม และเดือนพฤศจิกายนปี ค.ศ. 2003

จากการวิเคราะห์ลักษณะของงานแต่ละพาร์ทิชันสรุปได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักสามารถกระจายงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชัน โดยที่พาร์ทิชันแรกจะมี

งานขนาดเล็กกว่าพาร์ทิชันที่ 2 เป็นอัตราส่วน 1:2 โดยประมาณแสดงว่านโยบายแบบใช้เป้าหมายเป็นหลักจะเลือกส่งงานขนาดเล็กไปที่พาร์ทิชันแรก ซึ่งเหมือนกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล แต่นโยบายแบบใช้เป้าหมายเป็นหลักจะเลือกส่งงานขนาดเล็กบางงานไปที่พาร์ทิชันที่ 2 ด้วย ซึ่งดูได้จากจำนวนงาน (ตารางที่ 4-4) และลักษณะของงาน (ตารางที่ 4-5) ต่อไปจะวิเคราะห์ถึงค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันดังตารางที่ 4-6

ตารางที่ 4-5 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน

Average job size (NT) in node-hour									
	FCFS-backfill			LXF-backfill			Goal-oriented		
Partition / Month	1st	2nd	1st/2nd	1st	2nd	1st/2nd	1st	2nd	1st/2nd
6/03	28.6	37.2	0.77	28.7	37.1	0.77	30.0	36.2	0.83
7/03	22.9	93.7	0.24	22.9	81.8	0.28	26.1	80.3	0.33
8/03	17.5	22.6	0.77	11.3	32.7	0.35	13.0	28.8	**0.45**
9/03	10.7	31.9	0.34	14.6	24.1	**0.61**	12.1	28.7	**0.42**
10/03	9.4	24.3	0.39	10.6	21.7	**0.49**	9.5	23.8	**0.40**
11/03	8.2	36.3	0.23	7.4	43.0	0.17	8.4	34.2	0.25
12/03	14.8	23.5	**0.63**	12.9	26.9	**0.48**	12.4	27.3	**0.45**
1/04	14.1	28.4	**0.50**	17.3	23.8	0.73	13.8	29.2	**0.47**
2/04	9.9	22.8	**0.43**	10.6	21.4	**0.50**	12.2	19.0	**0.64**
3/04	15.2	23.4	0.65	11.0	32.7	0.34	11.9	29.4	**0.40**

จากตารางที่ 4-6 จะเห็นได้ว่าทั้ง 3 นโยบายในทุกเดือนงานที่พาร์ทิชันแรกจะเป็นงานที่ต้องการหน่วยประมวลผลน้อยกว่าพาร์ทิชันที่ 2 ในบางเดือนงานที่พาร์ทิชันแรกต้องการหน่วยประมวลผลน้อยกว่าพาร์ทิชันที่ 2 ถึง 4 เท่าเช่น เดือนกรกฎาคมปี ค.ศ. 2003 ซึ่งจะเห็นได้ว่าลักษณะของงานในตารางที่ 4-5 ทั้ง 3 นโยบายงานที่พาร์ทิชันแรกจะเป็นงานขนาดเล็กกว่าพาร์ทิชันที่ 2 เช่นกัน เมื่อวิเคราะห์ถึงอัตราส่วนของขนาดงานในพาร์ทิชันที่ 1 ต่อพาร์ทิชันที่ 2 แล้วพบว่าเดือนมิถุนายน เดือนสิงหาคม และเดือนธันวาคมปี ค.ศ. 2003 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานจากนโยบายแบบเอฟซีเอฟเอสแบ็กฟิลบนพาร์ทิชันแรกน้อยกว่าพาร์ทิชันที่ 2 ประมาณครึ่งหนึ่ง แต่ในเดือนมิถุนายน และเดือนกันยายนปี ค.ศ. 2003 เดือนกุมภาพันธ์ปี ค.ศ. 2004 ค่าเฉลี่ย

หน่วยประมวลผลที่ใช้ของงานจากนโยบายแบบเอลเอกซ์เอฟแบ็กฟิลบนพาร์ทิชันแรกน้อยกว่าพาร์ทิชันที่ 2 ประมาณครึ่งหนึ่ง ส่วนนโยบายแบบใช้เป้าหมายเป็นหลักจะมีค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานบนพาร์ทิชันแรกน้อยกว่าพาร์ทิชันที่ 2 ประมาณครึ่งหนึ่งในเดือนมิถุนายน เดือนตุลาคม เดือนพฤศจิกายน และเดือนธันวาคมปี ค.ศ. 2003

จากการวิเคราะห์ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานแต่ละพาร์ทิชันสรุปได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักสามารถทำให้งานในแต่ละพาร์ทิชันใช้หน่วยประมวลผลได้ใกล้เคียงกันในบางเดือนเช่นเดียวกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล

ตารางที่ 4-6 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน

Average N									
	FCFS-backfill			LXF-backfill			Goal-oriented		
Partition / Month	1st	2nd	1st/2nd	1st	2nd	1st/2nd	1st	2nd	1st/2nd
6/03	6.6	13.8	**0.48**	6.6	13.7	**0.48**	7.4	13.1	**0.56**
7/03	8.6	35.0	0.25	7.5	31.8	0.24	9.4	30.5	0.31
8/03	4.5	7.2	**0.63**	3.2	9.9	0.32	3.4	9.1	0.37
9/03	4.4	12.2	0.36	5.2	9.9	**0.53**	4.4	11.6	0.38
10/03	2.9	7.4	0.39	2.8	7.1	0.39	3.2	6.9	**0.46**
11/03	2.9	10.8	0.27	2.8	12.4	0.23	3.2	10.0	**0.32**
12/03	2.9	7.2	**0.40**	2.3	8.5	0.27	3.2	7.5	**0.43**
1/04	4.6	16.2	0.28	4.2	14.5	0.29	8.2	12.6	0.65
2/04	2.2	7.5	0.29	2.7	6.6	**0.41**	2.5	6.4	0.39
3/04	2.2	7.8	0.28	2.6	9.4	0.28	2.6	8.7	0.30

ตารางที่ 4-4, 4-5 และ 4-6 แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักสามารถกระจายงานได้ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเช่นเดียวกับระบบที่มีจำนวนหน่วยประมวลผลเท่ากัน (หัวข้อ 4.2.1) โดยที่บนพาร์ทิชันที่ 1 งานของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล และนโยบายแบบใช้เป้าหมายเป็นหลักจะเป็นงานขนาดเล็ก ส่วนบนพาร์ทิชันที่ 2 งานของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลจะเป็นงานขนาดใหญ่ แต่งานของนโยบายแบบใช้เป้าหมายเป็นหลักจะเป็นงานขนาดใหญ่ และงาน

ขนาดเล็กบางงาน จากลักษณะงานในแต่ละพาร์ทิชันดังกล่าวจะเห็นได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักจะส่งงานเล็กไปที่พาร์ทิชันที่เล็กกว่า และส่งงานขนาดใหญ่รวมกับงานเล็กบางงานไปที่พาร์ทิชันใหญ่กว่า การจัดการดังกล่าวจะช่วยให้งานถูกจัดไว้บนพื้นที่ว่างบนทั้ง 2 พาร์ทิชันดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ดังจะเห็นได้จากค่าเฉลี่ยการรอของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนพาร์ทิชันที่ 1 (ภาพประกอบที่ 4-5 (ค)) ต่ำกว่าพาร์ทิชันที่ 2 (ภาพประกอบที่ 4-5 (ง))

จากที่ได้กล่าวมาในหัวข้อ 4.2.1 และ 4.2.2 แสดงให้เห็นว่าประสิทธิภาพบนระบบพาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันจำนวน 2 พาร์ทิชัน และไม่เท่ากันจำนวน 2 พาร์ทิชันของนโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเหมือนบนระบบแบบพาร์ทิชันเดียวใน [6] และ [13] หัวข้อต่อไปจะนำเสนอและวิเคราะห์ประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักเมื่อเพิ่มจำนวนพาร์ทิชันตั้งแต่ 3 ถึง 5 พาร์ทิชันบนระบบหลายพาร์ทิชันแบบจำนวนหน่วยประมวลผลเท่ากัน

4.2.3 ประสิทธิภาพบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

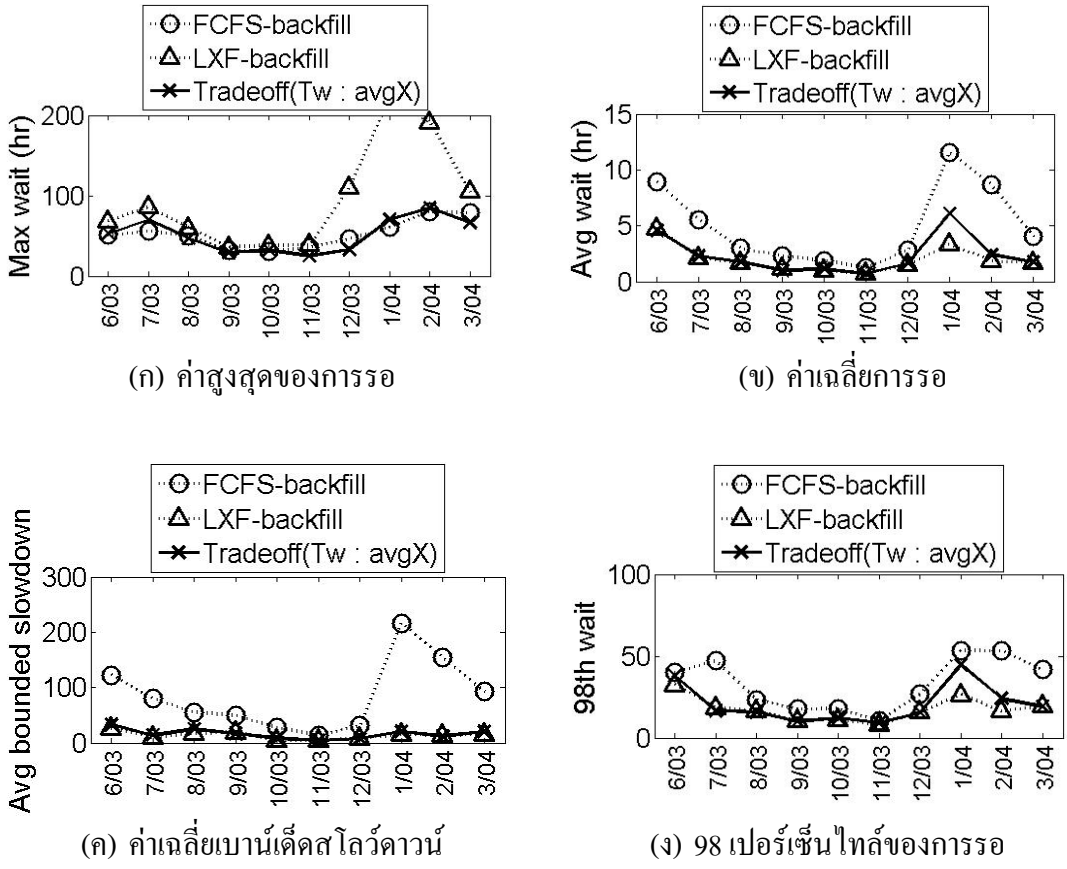
หัวข้อนี้จะทดสอบและวิเคราะห์ประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบจำนวนหน่วยประมวลผลเท่ากันคือ 32 โหนด โดยจะเพิ่มจำนวนพาร์ทิชันตั้งแต่ 3 ถึง 5 พาร์ทิชัน การทดสอบในหัวข้อนี้ใช้เวิร์คโหลดไฟล์ซึ่งแสดงรายละเอียดไว้ในตารางที่ 3-3

4.2.3.1 ระบบ 3 พาร์ทิชัน

ในหัวข้อนี้จะแสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 3 พาร์ทิชันซึ่งมีจำนวนหน่วยประมวลผลเท่ากันคือ 32 โหนด โดยใช้เวิร์คโหลดในตารางที่ 3-3 จะเห็นได้ว่าค่าโหลดส่วนใหญ่ของระบบ 3 พาร์ทิชันจะสูงกว่าค่าโหลดในเวิร์คโหลดต้นฉบับ (ตารางที่ 3-1) ผลที่ได้จากการทดสอบนโยบายจะแสดงเป็น 3 หัวข้อย่อยคือ ประสิทธิภาพโดยรวมของระบบ (หัวข้อ 4.2.3.1.1) ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน (หัวข้อ 4.2.3.1.2) และวิเคราะห์ประสิทธิภาพ (หัวข้อ 4.2.3.1.3)

4.2.3.1.1 ประสิทธิภาพโดยรวมของระบบ

หัวข้อนี้จะแสดงการเปรียบเทียบประสิทธิภาพโดยรวมทั้งระบบของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน ดังแสดงในภาพประกอบที่ 4-6



(ก) ค่าสูงสุดของการรอ (ข) ค่าเฉลี่ยการรอ

(ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ง) 98 เปอร์เซ็นไทล์ของการรอ

ภาพประกอบ 4-6 ประสิทธิภาพบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

ภาพประกอบที่ 4-6 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน โดยแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากับ 32 โหนด และแสดงเป็นกราฟประสิทธิภาพแต่ละเดือนจำนวน 10 เดือนตั้งแต่เดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งแสดงประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่านโยบายแบบเอฟซีเอฟเอสแบ็กฟิลให้

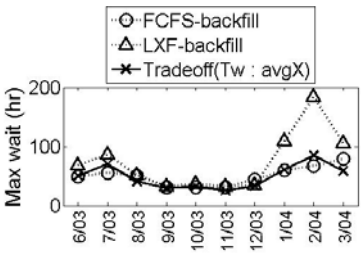
ค่าสูงสุดการรอ (ภาพประกอบที่ 4-6 (ก)) ดีกว่านโยบายแบบเอลเอกซ์เอฟแบ็กฟิลทุกเดือน ในขณะที่นโยบายแบบเอลเอกซ์เอฟแบ็กฟิลให้ค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-6 (ข)) ดีกว่านโยบายแบบเอฟซีเอฟเอสแบ็กฟิลทุกเดือนเช่นกัน จะเห็นได้ว่านโยบายทั้งสองตัวทำงานดีในเป้าหมายอันใดอันหนึ่งเท่านั้น แต่นโยบายแบบใช้เป้าหมายเป็นหลักให้ค่าประสิทธิภาพการทำงานทั้งสามเป้าหมายดีทั้งหมด เมื่อดูที่ 98 เปอร์เซ็นไทล์ของการรอ (ภาพประกอบที่ 4-6 (ง)) จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีเช่นกัน

โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-6 นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าสูงสุดการรอและค่าเฉลี่ยการรอ แม้ว่าตัววัดประสิทธิภาพทั้ง 2 จะขัดแย้งกันก็ตาม

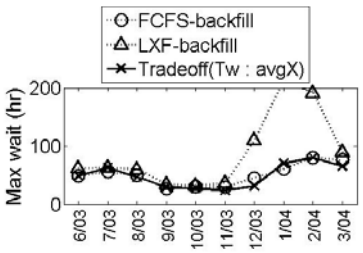
4.2.3.1.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน

จากหัวข้อ 4.2.3.1.1 ได้เปรียบเทียบประสิทธิภาพโดยภาพรวมของทั้งระบบ โดยแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ดังนั้นหัวข้อนี้จะวิเคราะห์หาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก ดังแสดงในภาพประกอบที่ 4-7 ทั้งนี้แนวนอนแสดงเดือนที่วัด และแนวตั้งแสดงประสิทธิภาพต่าง ๆ อันได้แก่ ค่าสูงสุดการรอ ค่าเฉลี่ยการรอ และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แต่ละค่าจะแสดงเป็น 3 ภาพ โดยภาพประกอบที่ 4-7 ก, ข, ค แสดงผลของค่าสูงสุดการรอของแต่ละนโยบายแต่ละพาร์ทิชัน ในขณะที่ภาพประกอบที่ 4-7 ง, จ, ฉ แสดงผลของค่าเฉลี่ยการรอของแต่ละนโยบายแต่ละพาร์ทิชัน และภาพประกอบที่ 4-7 ช, ซ, ฌ แสดงผลของค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของแต่ละนโยบายแต่ละพาร์ทิชัน

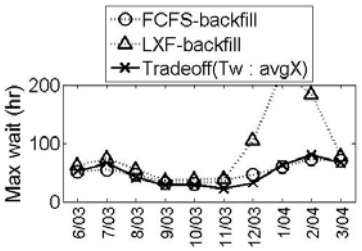
จากภาพประกอบที่ 4-7 จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีของทั้ง 3 ประสิทธิภาพที่แสดงบนทั้ง 3 พาร์ทิชัน ยกเว้นค่าเฉลี่ยการรอเดือนมกราคมปี ค.ศ. 2004 บนพาร์ทิชันที่ 1 เท่านั้นที่นโยบายแบบใช้เป้าหมายเป็นหลักให้ค่าแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ซึ่งแย่กว่าเพียงเล็กน้อยเท่านั้น เมื่อดูภาพรวมของระบบที่ภาพประกอบ 4-6 แล้วพบว่านโยบายแบบใช้เป้าหมายเป็นหลักอยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง



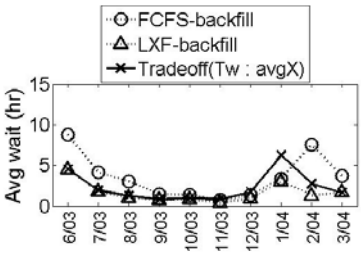
(ก) ค่าสูงสุดของการรอของพาร์ทิชันที่ 1



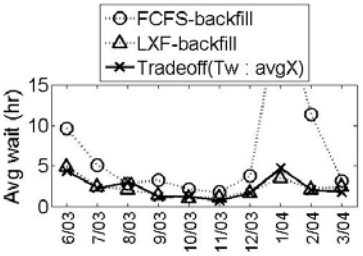
(ข) ค่าสูงสุดของการรอของพาร์ทิชันที่ 2



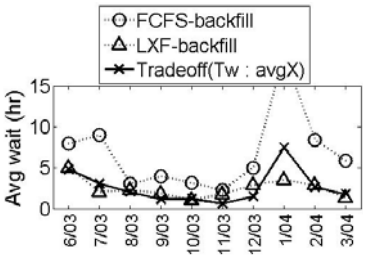
(ค) ค่าสูงสุดของการรอของพาร์ทิชันที่ 3



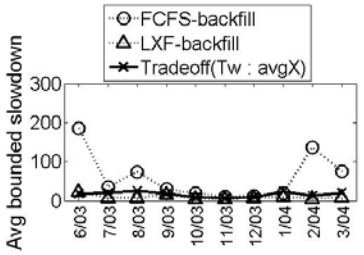
(ง) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 1



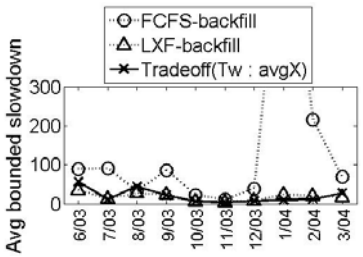
(จ) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 2



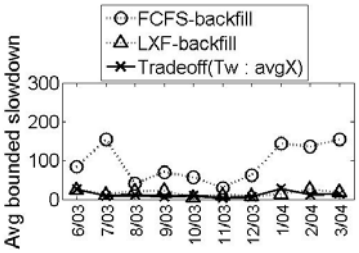
(ฉ) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 3



(ช) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 1



(ซ) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 2



(ฌ) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 3

ภาพประกอบ 4-7 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

จากประสิทธิภาพที่แสดงในภาพประกอบที่ 4-6 และ 4-7 จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล แม้จะทำงานบนระบบ 3 พาร์ทิชันที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากันก็ตาม โดยให้ประสิทธิภาพที่ดีทั้งค่าสูงสุดการรอ ค่าเฉลี่ยการรอ และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์

4.2.3.1.3 วิเคราะห์ประสิทธิภาพ

จากหัวข้อที่ 4.2.3.1.1 และ 4.2.3.1.2 ก่อนหน้าได้แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ในหัวข้อนี้จะตรวจสอบหาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ถึงจำนวนงาน และปริมาณงานในแต่ละพาร์ทิชันโดยตารางที่ 4-7 แสดงจำนวนงานที่ถูกส่งไปประมวลผลในแต่ละพาร์ทิชันของแต่ละนโยบาย ในขณะที่ตารางที่ 4-8 แสดงลักษณะของงาน (จำนวนหน่วยประมวลผล × เวลาที่ใช้) ซึ่งแสดงค่าเฉลี่ยมีหน่วยเป็น จำนวนหน่วยประมวลผลชั่วโมง และตารางที่ 4-9 แสดงค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานมีหน่วยเป็นจำนวนหน่วยประมวลผล เพื่อให้ง่ายต่อการวิเคราะห์ค่าในแต่ละพาร์ทิชันจึงใช้ค่าความแปรปรวน (Variance) เพิ่มมาแต่ละนโยบาย ซึ่งค่าความแปรปรวนจะสามารถแสดงการกระจายของค่าในแต่ละพาร์ทิชันของแต่ละนโยบาย ถ้าค่าความแปรปรวนน้อยแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นใกล้เคียงกันมาก แต่ถ้าค่าความแปรปรวนมากแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นห่างกันมาก

จากตารางที่ 4-7, 4-8 และ 4-9 จะเห็นได้ว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งเอฟซีเอฟเอส และเอลเอกซ์เอฟมีจำนวนงานประมวลผลบนพาร์ทิชันที่ 1 มากกว่าพาร์ทิชันอื่น ๆ ในขณะที่ขนาดของงานในตารางที่ 4-8 และ 4-9 พาร์ทิชันแรกของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเป็นงานที่มีขนาดเล็กกว่าพาร์ทิชันอื่น ๆ อย่างเช่นในเดือนพฤศจิกายนปี ค.ศ. 2003 จำนวนงานในพาร์ทิชันแรกมากกว่าพาร์ทิชันอื่น ๆ ถึง 3 เท่า แต่ขนาดของงาน และหน่วยประมวลผลที่งานต้องการบนพาร์ทิชันแรกกลับน้อยกว่าพาร์ทิชันอื่น ๆ 3 ถึง 4 เท่า นั่นแสดงว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลจะส่งงานที่ต้องการหน่วยประมวลผลน้อยไปประมวลผลบนพาร์ทิชันแรกก่อน ซึ่งต่างจากนโยบายแบบใช้เป้าหมายเป็นหลักที่สามารถกระจายจำนวนงาน งานลักษณะต่าง ๆ และความต้องการหน่วยประมวลผลของงานไปแต่ละพาร์ทิชันได้ใกล้เคียงกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยดูได้จากค่าความแปรปรวนของนโยบายแบบใช้เป้าหมายเป็นหลักที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเกือบทุกเดือน ยกเว้นเดือนมิถุนายน และเดือนสิงหาคมปี ค.ศ. 2003 จำนวนงานของนโยบายแบบใช้เป้าหมายเป็นหลักที่เดือนมิถุนายนปี ค.ศ. 2003 แย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลแต่เมื่อวิเคราะห์ถึงลักษณะของงาน และความต้องการหน่วยประมวลผลของงานแล้ว พบว่าค่าในแต่ละพาร์ทิชันใกล้เคียงกันมากกว่านโยบายแบบเอฟซีเอฟเอสแบ็กฟิล ส่วนเดือนสิงหาคมปี ค.ศ. 2003 ทั้งจำนวนงาน ลักษณะของงาน และความต้องการหน่วยประมวลผลของงานจากนโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่านโยบาย

แบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล แต่จะเห็นได้ว่าประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักของเดือนสิงหาคมปี ค.ศ. 2003 ในภาพประกอบที่ 4-6 ซึ่งเป็นแบบรวมทั้งระบบ และภาพประกอบที่ 4-7 แบบแยกแต่ละพาร์ทิชันมีค่าใกล้เคียงกับค่าที่ดีที่สุดหรืออยู่ระหว่างค่าจากนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง

จากผลการทดสอบในภาพประกอบที่ 4-6 ซึ่งเป็นแบบรวมทั้งระบบ และแบบแยกแต่ละพาร์ทิชันในภาพประกอบที่ 4-7 จะเห็นได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ซึ่งเหมือนกับระบบแบบพาร์ทิชันเดียว และระบบแบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน (หัวข้อ 4.2.1) อีกทั้งนโยบายแบบใช้เป้าหมายเป็นหลักยังคงสามารถกระจายจำนวนงาน และชนิดของงานไปแต่ละพาร์ทิชันได้ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยจะส่งงานที่มีขนาดเล็ก และใหญ่ไปทุกพาร์ทิชันให้ใกล้เคียงกันมากที่สุด ซึ่งต่างจากนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลที่เลือกจะส่งงานไปที่พาร์ทิชันแรกก่อน ดังจะเห็นได้จากจำนวนงาน (ตารางที่ 4-7) ลักษณะของงาน (ตารางที่ 4-8) และความต้องการหน่วยประมวลผลของงาน (ตารางที่ 4-9) การที่จำนวนงานแต่ละพาร์ทิชันมีจำนวนใกล้เคียงกัน และลักษณะของงานแต่ละพาร์ทิชันมีขนาดใกล้เคียงกันสามารถบอกได้ว่างานถูกกระจายไปประมวลผลแต่ละพาร์ทิชันได้เท่า ๆ กัน ซึ่งจะช่วยให้ประสิทธิภาพโดยรวมดีขึ้น

ตารางที่ 4-7 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

Number of jobs												
	FCFS-backfill				LXF-backfill				Goal-oriented			
Partition / Month	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance
6/03	789	878	524	22,606.9	877	517	797	23,822.2	749	655	787	**3,078.2**
7/03	563	570	267	19,941.6	526	439	435	**1,762.9**	606	472	322	13,456.9
8/03	1,665	741	815	175,750.2	1,349	915	957	**38,198.2**	1,852	573	796	311,189.6
9/03	1,840	831	386	370,024.7	1,523	1,006	528	165,088.7	1,435	932	690	**96,288.7**
10/03	2,428	846	875	546,152.7	1,843	1,249	1,057	111,944.0	1,471	1,461	1,217	**13,794.7**
11/03	2,172	609	662	525,097.6	2,251	712	480	617,642.9	1,051	1,304	1,088	**12,448.2**
12/03	1,862	817	842	237,005.6	1,840	1,065	616	255,600.2	1,403	968	1,150	**31,817.6**
1/04	1,744	707	705	239,432.7	1,178	1,082	896	13,704.0	961	1,158	1,037	**6,580.7**
2/04	1,864	893	1,212	163,300.7	2,022	1,155	792	266,262.0	1,279	1,743	947	**106,570.7**
3/04	1,420	1,191	855	53,840.2	1,112	791	1,563	100,269.6	1,373	1,099	994	**25,526.9**

ตารางที่ 4-8 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

Average job size (NT) in node-hour												
	FCFS-backfill				LXF-backfill				Goal-oriented			
Partition / Month	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance
6/03	26.5	23.0	39.8	52.4	24.3	40.7	24.7	58.3	27.4	31.9	26.2	**6.0**
7/03	29.9	27.1	57.7	190.8	30.2	38.0	34.7	**10.2**	27.8	32.6	48.1	75.0
8/03	10.6	22.3	20.8	27.0	13.1	18.6	17.2	**5.4**	8.8	32.8	20.1	96.1
9/03	8.4	19.7	43.0	207.5	9.7	16.4	32.4	90.7	11.2	16.7	24.3	**28.8**
10/03	7.9	21.9	22.1	44.2	10.6	15.4	17.5	8.3	12.9	13.3	15.3	**1.1**
11/03	6.7	28.1	23.8	85.4	6.4	23.0	34.8	135.7	15.8	11.9	14.0	**2.5**
12/03	10.4	23.4	22.7	35.6	10.4	18.1	31.2	73.7	13.5	20.0	16.7	**7.0**
1/04	11.5	27.1	25.7	49.7	17.1	16.8	21.3	4.2	20.6	16.3	18.1	**3.1**
2/04	10.9	20.3	15.4	14.7	9.7	15.9	24.3	35.8	14.9	11.0	20.0	**13.6**
3/04	13.8	18.1	24.9	20.9	19.7	25.8	12.8	28.2	14.6	18.9	21.7	**8.5**

ตารางที่ 4-9 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

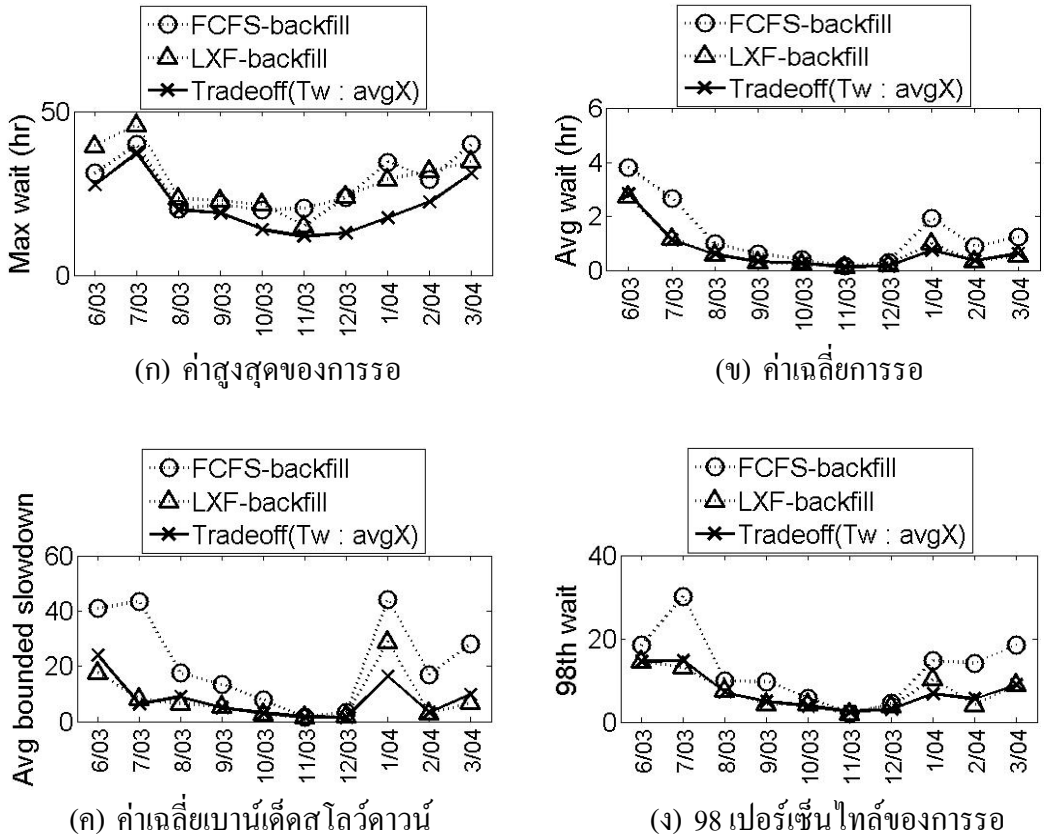
Average N												
	FCFS-backfill				LXF-backfill				Goal-oriented			
Partition / Month	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance
6/03	8.9	7.6	11.3	2.35	8.6	11.2	7.9	2.02	8.6	9.2	9.2	**0.08**
7/03	10.6	13.0	19.6	14.48	12.3	16.4	11.5	4.61	10.8	15.0	15.6	**4.56**
8/03	3.6	6.2	6.3	**1.56**	3.4	6.5	5.5	1.67	3.1	8.6	6.3	5.09
9/03	4.5	7.4	14.3	16.90	4.7	7.1	10.7	6.08	4.9	7.2	8.9	**2.69**
10/03	2.9	6.2	6.6	2.75	3.6	4.7	5.3	0.50	4.0	4.3	4.8	**0.11**
11/03	2.8	6.8	8.9	6.40	2.7	6.4	11.6	13.33	5.9	3.8	4.5	**0.76**
12/03	3.3	5.0	6.3	1.51	3.1	5.7	6.0	1.70	4.4	5.3	3.7	**0.43**
1/04	2.9	18.0	17.1	47.83	10.1	10.4	7.6	**1.58**	12.2	5.7	11.1	8.07
2/04	2.6	5.9	4.9	**1.91**	2.7	3.9	7.7	4.54	4.1	2.7	6.5	2.46
3/04	4.0	3.7	6.1	1.14	4.1	6.0	3.8	0.95	3.9	4.8	4.7	**0.16**

4.2.3.2 ระบบ 4 พาร์ทิชัน

ในหัวข้อนี้จะแสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 4 พาร์ทิชันซึ่งมีจำนวนหน่วยประมวลผลเท่ากันคือ 32 โหนด โดยใช้เวิร์คโหลดในตารางที่ 3-3 จะเห็นได้ว่าค่าโหลดของระบบ 4 พาร์ทิชันจะลดต่ำกว่าค่าโหลดในเวิร์คโหลดต้นฉบับเล็กน้อย (ตารางที่ 3-1) ผลที่ได้จากการทดสอบนโยบายจะแสดงเป็น 3 หัวข้อย่อยคือ ประสิทธิภาพโดยรวมของระบบ (หัวข้อ 4.2.3.2.1) ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน (หัวข้อ 4.2.3.2.2) และวิเคราะห์ประสิทธิภาพ (หัวข้อ 4.2.3.2.3)

4.2.3.2.1 ประสิทธิภาพโดยรวมของระบบ

หัวข้อนี้จะแสดงการเปรียบเทียบประสิทธิภาพโดยรวมทั้งระบบของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน



(ก) ค่าสูงสุดของการรอ (ข) ค่าเฉลี่ยการรอ

(ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ง) 98 เปอร์เซ็นไทล์ของการรอ

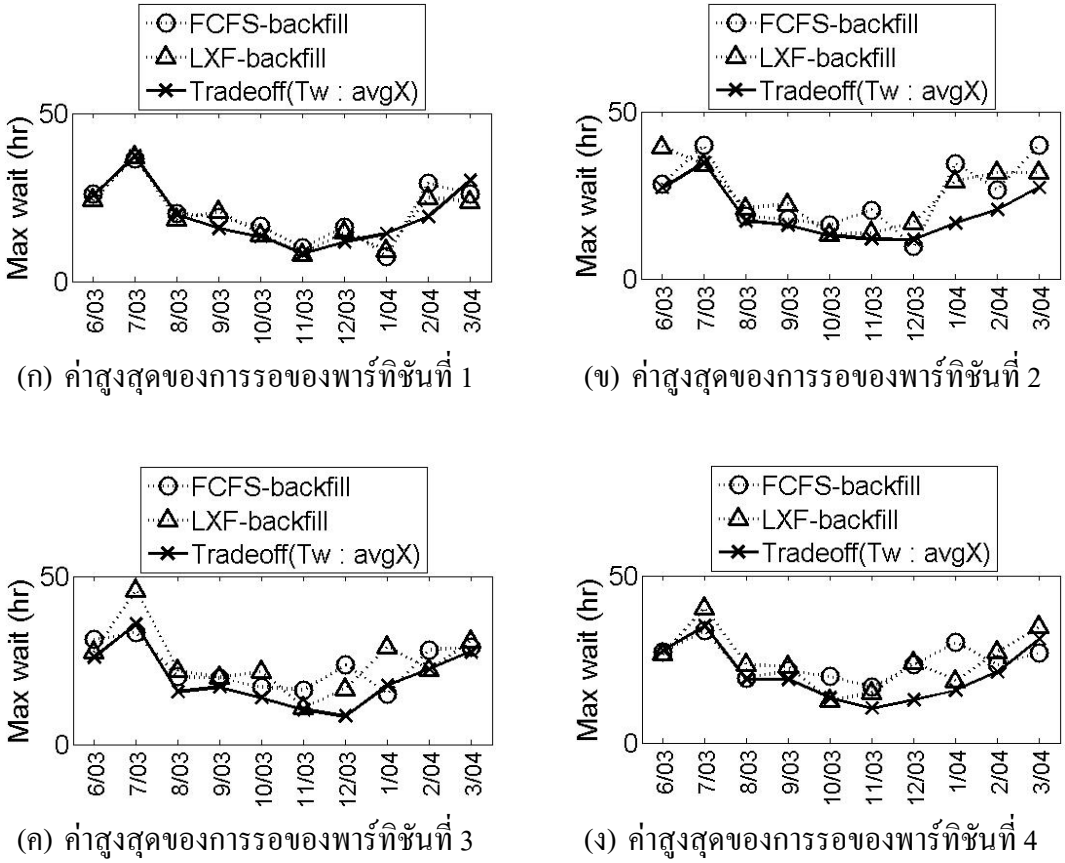
ภาพประกอบ 4-8 ประสิทธิภาพบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

ภาพประกอบที่ 4-8 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน โดยมีจำนวนหน่วยประมวลผลแต่ละพาร์ทิชันเท่ากับ 32 โหนด และแสดงเป็นกราฟประสิทธิภาพแต่ละเดือนจำนวน 10 เดือน ตั้งแต่เดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งเป็นประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่านโยบายแบบเอฟซีเอฟเอสแบ็กฟิลให้ค่าสูงสุดการรอ (ภาพประกอบที่ 4-8 (ก)) ดีกว่านโยบายแบบเอลเอกซ์เอฟแบ็กฟิลเกือบทุกเดือน แต่นโยบายแบบเอลเอกซ์เอฟแบ็กฟิลให้ค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-8 (ข)) ดีกว่านโยบายแบบเอฟซีเอฟเอสแบ็กฟิลทุกเดือน จะเห็นได้ว่านโยบายทั้งสองตัวทำงานดีในเป้าหมายอันใดอันหนึ่งเท่านั้น แต่นโยบายแบบใช้เป้าหมายเป็นหลักให้ค่าประสิทธิภาพการทำงานทั้งสามเป้าหมายดีทั้งหมด เมื่อดูที่ 98 เปอร์เซ็นไทล์ของการรอ (ภาพประกอบที่ 4-8 (ง)) จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีเช่นกัน โดยเฉพาะค่าสูงสุดการรอของนโยบายแบบใช้เป้าหมายเป็นหลัก ซึ่งให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองอย่างเห็นได้ชัด

โดยสรุปจากผลการทดสอบดังแสดงในภาพประกอบที่ 4-8 นโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเมื่อระบบเป็นแบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน โดยให้ประสิทธิภาพที่ดีทั้งค่าสูงสุดการรอ และค่าเฉลี่ยการรอ แม้ว่าตัววัดประสิทธิภาพทั้ง 2 จะขัดแย้งกันก็ตาม

4.2.3.2.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน

จากหัวข้อ 4.2.3.2.1 ได้เปรียบเทียบประสิทธิภาพโดยภาพรวมของทั้งระบบ โดยแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ดังนั้นหัวข้อนี้จะวิเคราะห์หาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ประสิทธิภาพของแต่ละนโยบายบนแต่ละพาร์ทิชัน ดังแสดงในภาพประกอบที่ 4-9, 4-10 และ 4-11 ทั้งนี้แนวนอนแสดงเดือนที่วัด และแนวตั้งแสดงประสิทธิภาพต่าง ๆ อันได้แก่ ค่าสูงสุดการรอ ค่าเฉลี่ยการรอ และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ แต่ละค่าจะแสดงเป็น 4 ภาพ โดยภาพประกอบที่ 4-9 แสดงผลของค่าสูงสุดการรอของแต่ละนโยบายแต่ละพาร์ทิชัน ในขณะที่ภาพประกอบที่ 4-10 แสดงผลของค่าเฉลี่ยการรอของแต่ละนโยบายแต่ละพาร์ทิชัน และภาพประกอบที่ 4-11 แสดงผลของค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของแต่ละนโยบายแต่ละพาร์ทิชัน

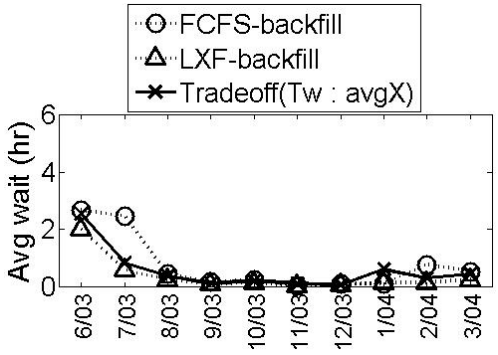


(ก) ค่าสูงสุดของการรอของพาร์ทิชันที่ 1 (ข) ค่าสูงสุดของการรอของพาร์ทิชันที่ 2

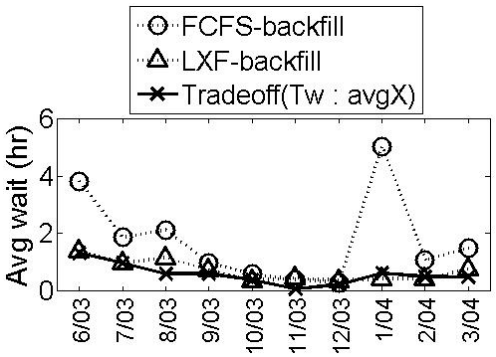
(ค) ค่าสูงสุดของการรอของพาร์ทิชันที่ 3 (ง) ค่าสูงสุดของการรอของพาร์ทิชันที่ 4

ภาพประกอบ 4-9 ค่าสูงสุดของการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

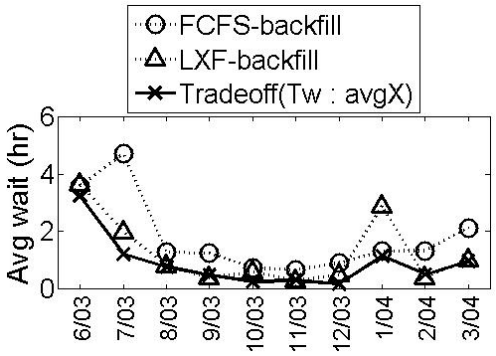
จากภาพประกอบที่ 4-9, 4-10 และ 4-11 จะเห็นว่าค่าประสิทธิภาพที่ได้จากการแยกแต่ละพาร์ทิชันของนโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีของทั้ง 3 ประสิทธิภาพที่แสดงบนทั้ง 4 พาร์ทิชัน ยกเว้นค่าสูงสุดของการรอบนพาร์ทิชันที่ 1 ที่เดือนมกราคม และมีนาคมปี ค.ศ. 2004 ซึ่งนโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล เช่นเดียวกับค่าเฉลี่ยการรอบนพาร์ทิชันที่ 1 ที่เดือนมกราคม ปี ค.ศ. 2004 ที่นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล แต่จะเห็นได้ว่าประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่าเฉพาะบนพาร์ทิชันที่ 1 เท่านั้น เนื่องมาจากนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลจะส่งงานเล็กและสั้นไปที่พาร์ทิชันที่ 1 แต่นโยบายแบบใช้เป้าหมายเป็นหลักจะพยายามกระจายงานไปทุก ๆ พาร์ทิชันให้เท่า ๆ กันจึงอาจทำให้พาร์ทิชันที่ 1 มีงานขนาดใหญ่อยู่ด้วยทำให้ค่าสูงสุดการรอ และค่าเฉลี่ยการรอสูงกว่า



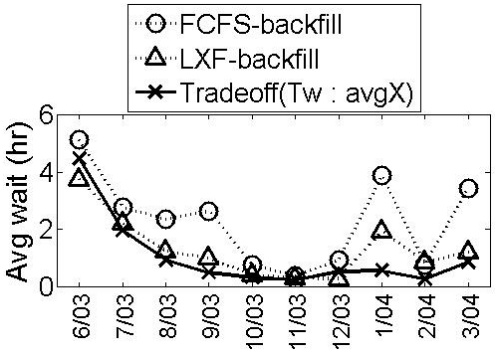
(ก) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 1



(ข) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 2



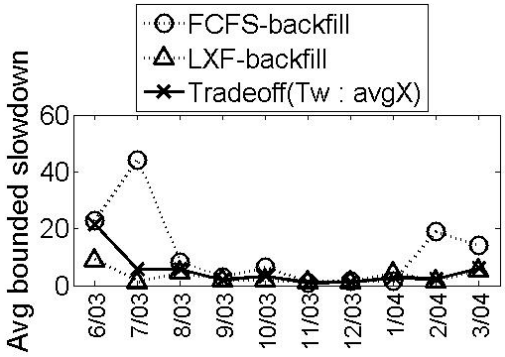
(ค) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 3



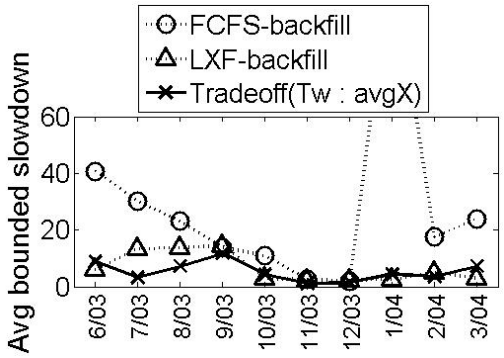
(ง) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 4

ภาพประกอบ 4-10 ค่าเฉลี่ยการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

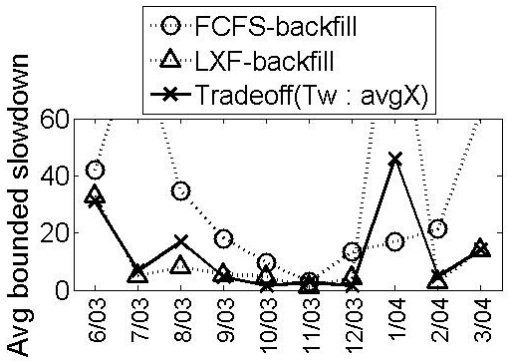
จากประสิทธิภาพที่แสดงในภาพประกอบที่ 4-8 ซึ่งเป็นแบบรวมทั้งระบบ และภาพประกอบที่ 4-9, 4-10 และ 4-11 ซึ่งเป็นแบบแยกพาร์ทิชันจะเห็นได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ถึงแม้จำนวนพาร์ทิชันจะเพิ่มขึ้นเป็น 4 พาร์ทิชันแล้วก็ตาม โดยให้ประสิทธิภาพที่ดีทั้งค่าสูงสุดการรอ ค่าเฉลี่ยการรอ และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์



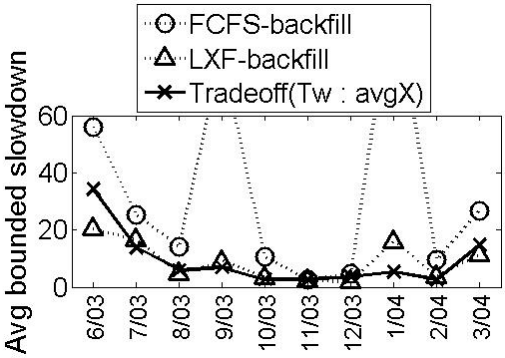
(ก) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 1



(ข) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 2



(ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 3



(ง) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 4

ภาพประกอบ 4-11 ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

4.2.3.2.3 วิเคราะห์ประสิทธิภาพ

จากหัวข้อที่ 4.2.3.2.1 และ 4.2.3.2.2 ก่อนหน้าได้แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ในหัวข้อนี้จะตรวจสอบหาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ถึงจำนวนงาน และปริมาณงานในแต่ละพาร์ทิชันโดยตารางที่ 4-10 แสดงจำนวนงานที่ถูกส่งไปประมวลผลในแต่ละพาร์ทิชันของแต่ละนโยบาย ในขณะที่ตารางที่ 4-11 แสดงลักษณะของงาน (จำนวนหน่วยประมวลผล × เวลาที่ใช้) มีหน่วยเป็น จำนวนหน่วยประมวลผลชั่วโมง และตารางที่ 4-12 แสดงค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานมีหน่วยเป็น จำนวนหน่วยประมวลผล เพื่อให้ง่ายต่อการวิเคราะห์ค่าในแต่ละพาร์ทิชันจึงใช้ค่าความแปรปรวนเพิ่มมาแต่ละนโยบาย ซึ่งค่าความแปรปรวนจะสามารถแสดงการกระจายของค่าในแต่ละพาร์ทิชันของแต่ละ

นโยบาย ถ้าค่าความแปรปรวนน้อยแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นใกล้เคียงกันมาก แต่ถ้าค่าความแปรปรวนมากแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นห่างกันมาก

จากตารางที่ 4-10, 4-11 และ 4-12 จะเห็นได้ว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งเอฟซีเอฟเอส และเอลเอกซ์เอฟมีจำนวนงานประมวลผลบนพาร์ทิชันที่ 1 มากกว่าพาร์ทิชันอื่น ๆ ในขณะที่ขนาดของงานในตารางที่ 4-11 และ 4-12 ที่พาร์ทิชันแรกของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเป็นงานที่มีขนาดเล็กกว่าพาร์ทิชันอื่น ๆ ซึ่งจะเห็นได้ชัดเจนในเดือนพฤศจิกายนปี ค.ศ. 2003 จำนวนงานในพาร์ทิชันแรกมากกว่าพาร์ทิชันอื่น ๆ ประมาณ 4 เท่า แต่ขนาดของงาน และหน่วยประมวลผลที่งานต้องการบนพาร์ทิชันแรกกลับน้อยกว่าพาร์ทิชันอื่น ๆ ประมาณ 6 เท่า นั่นแสดงว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลจะส่งงานที่ต้องการหน่วยประมวลผลน้อยไปประมวลผลบนพาร์ทิชันแรกก่อน ซึ่งต่างจากนโยบายแบบใช้เป้าหมายเป็นหลักที่สามารถกระจายจำนวนงาน งานลักษณะต่าง ๆ และความต้องการหน่วยประมวลผลของงานไปแต่ละพาร์ทิชันได้ใกล้เคียงกันมากกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลอย่างเช่น เดือนพฤศจิกายนปี ค.ศ. 2003 จะเห็นได้ว่าทั้งจำนวนงาน งานลักษณะต่าง ๆ และความต้องการหน่วยประมวลผลของงานจากนโยบายแบบใช้เป้าหมายเป็นหลักแต่ละพาร์ทิชันใกล้เคียงกันมากกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองมาก เมื่อวิเคราะห์ค่าความแปรปรวน พบว่านโยบายแบบใช้เป้าหมายเป็นหลักดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเกือบทุกเดือน ซึ่งเดือนที่ไม่ดีกว่าก็ยังอยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง

ความสามารถในการกระจายงานของนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันยังคงสามารถกระจายงานขนาดต่าง ๆ ไปแต่ละพาร์ทิชันให้เท่า ๆ กันได้ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง ดังจะเห็นได้จากตารางลักษณะงาน (ตารางที่ 4-11) และตารางค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงาน (ตารางที่ 4-12) โดยเฉพาะค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานแต่ละพาร์ทิชันแสดงให้เห็นได้ชัดเจนว่าค่าในแต่ละพาร์ทิชันใกล้เคียงกันมากกว่ามาก

ตารางที่ 4-10 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

Number of jobs															
	FCFS-backfill					LXF-backfill					Goal-oriented				
Partition / Month	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance
6/03	448	708	546	489	9,771.2	619	461	621	490	**5,325.7**	644	480	649	418	10,235.2
7/03	564	395	203	238	20,493.5	562	388	181	269	20,377.5	431	450	298	221	**8,976.5**
8/03	1,757	398	869	197	361,428.2	1,854	461	548	358	371,148.7	1,189	716	855	461	**69,053.2**
9/03	2,019	438	318	282	528,135.2	1,869	605	371	212	426,364.7	1,393	566	695	403	**142,481.7**
10/03	2,630	725	418	376	863,768.7	2,556	694	460	439	778,885.7	1,785	815	935	614	**199,530.2**
11/03	2,402	400	250	391	795,355.7	2,446	349	265	383	839,517.2	1,138	1,124	478	703	**79,387.7**
12/03	2,228	542	507	244	618,743.2	1,825	577	571	548	297,634.7	1,370	785	997	369	**130,983.7**
1/04	1,630	795	353	378	266,583.5	1,482	622	675	377	172,719.5	659	758	919	820	**8,930.5**
2/04	1,808	788	553	820	232,444.2	1,932	800	725	512	305,538.2	1,013	890	860	1,206	**18,516.2**
3/04	1,829	614	820	203	358,139.3	2,068	548	497	353	486,314.3	1,034	1,080	717	635	**37,395.3**

ตารางที่ 4-11 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

Average job size (NT) in node-hour															
	FCFS-backfill					LXF-backfill					Goal-oriented				
Partition / Month	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance
6/03	38.6	22.6	26.4	29.3	34.9	28.0	34.3	24.5	28.0	**12.5**	25.7	33.6	22.0	36.1	32.7
7/03	25.1	30.7	59.7	39.2	172.6	24.2	31.5	69.9	34.3	312.1	33.0	27.9	38.9	42.2	**30.1**
8/03	7.5	37.4	12.7	61.3	459.9	7.0	30.5	21.5	34.4	111.0	11.3	19.3	14.2	25.4	**28.7**
9/03	5.8	29.7	40.6	38.2	189.3	6.7	21.5	34.7	47.3	228.1	8.6	24.6	17.2	26.2	**48.6**
10/03	6.1	19.6	35.6	31.9	133.7	6.4	20.3	32.0	27.4	93.6	8.5	18.1	14.5	22.1	**25.0**
11/03	5.2	34.0	48.2	24.1	244.8	5.0	43.1	43.5	22.3	257.2	12.0	11.3	23.5	14.0	**23.9**
12/03	7.6	26.1	24.1	58.4	338.7	9.5	25.5	21.4	24.5	**40.6**	12.1	19.3	12.5	36.4	97.0
1/04	10.6	19.6	35.5	31.6	97.1	11.8	24.7	18.1	32.5	59.1	23.6	20.9	14.5	15.4	**14.3**
2/04	9.4	18.1	22.8	16.2	23.2	8.9	17.4	18.0	25.6	35.0	15.3	14.8	15.9	12.3	**1.9**
3/04	10.2	25.9	16.3	70.5	558.7	8.5	29.5	30.9	37.3	117.2	15.9	14.3	23.5	21.4	**14.4**

ตารางที่ 4-12 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

Average N															
	FCFS-backfill					LXF-backfill					Goal-oriented				
Partition / Month	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance
6/03	11.0	8.0	7.5	10.1	2.09	7.7	10.6	8.3	9.9	1.37	7.9	9.3	8.5	10.9	**1.27**
7/03	9.5	13.4	21.4	15.2	18.44	9.2	13.0	21.6	16.9	21.17	12.8	11.6	14.2	16.5	**3.32**
8/03	2.8	9.4	4.6	15.0	22.39	2.5	7.9	6.9	9.8	7.18	3.6	5.6	4.0	8.5	**3.71**
9/03	3.2	11.0	14.2	14.7	21.14	3.2	10.2	10.7	17.8	26.68	4.3	8.3	7.9	9.4	**3.66**
10/03	2.4	6.0	8.4	10.3	8.70	2.6	5.7	8.7	8.4	6.05	2.9	5.6	4.2	7.2	**2.56**
11/03	2.1	9.3	15.2	9.0	21.53	2.2	10.5	13.8	9.2	17.89	4.2	3.6	7.3	5.3	**1.99**
12/03	2.4	6.4	8.5	9.7	7.70	2.9	5.9	6.5	5.8	1.95	3.0	4.4	4.8	8.6	**4.30**
1/04	2.6	16.0	16.2	19.2	41.21	3.1	6.9	16.9	25.6	77.28	8.7	8.4	11.1	9.2	**1.10**
2/04	2.6	5.0	6.6	4.7	2.03	2.1	4.8	5.8	7.8	4.22	4.0	4.6	5.1	2.9	**0.67**
3/04	2.4	6.6	4.9	13.8	17.99	2.2	6.8	8.1	8.1	5.88	4.1	3.4	5.8	5.0	**0.82**

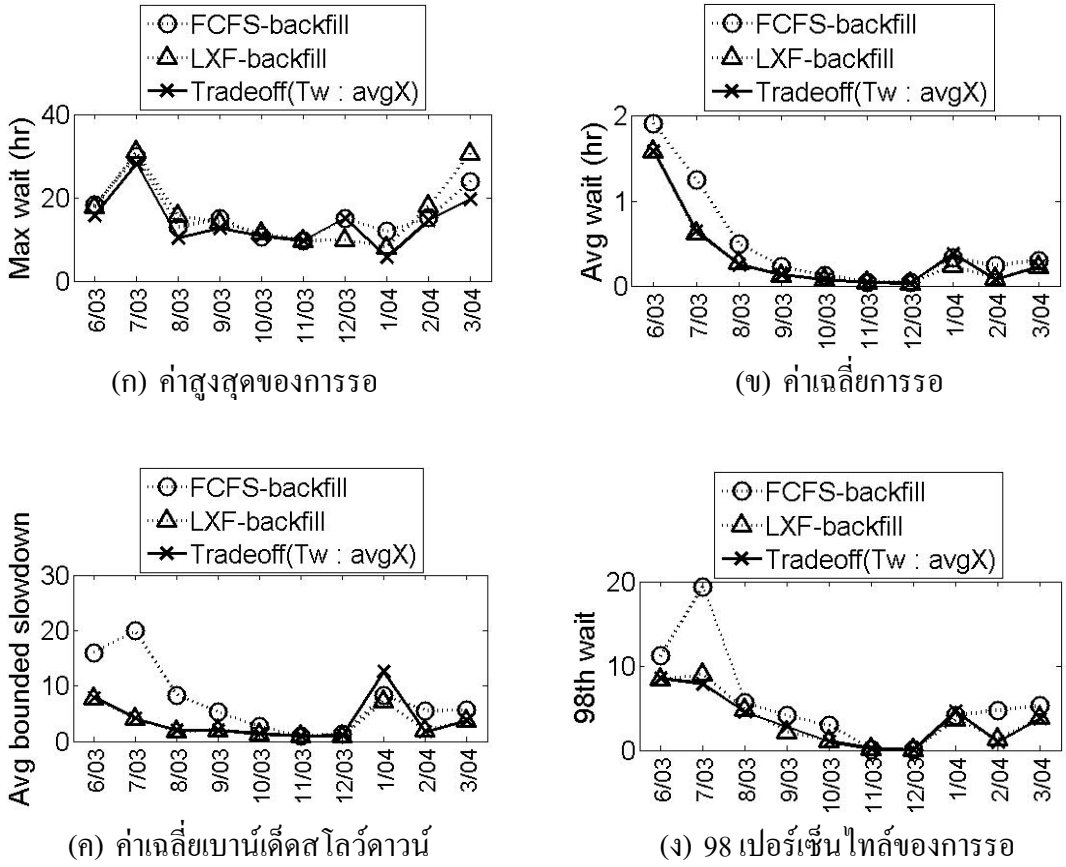
จากผลการทดสอบในภาพประกอบที่ 4-8 ซึ่งเป็นแบบรวมทั้งระบบ และแบบแยกแต่ละพาร์ทิชันในภาพประกอบที่ 4-9, 4-10 และ 4-11 จะเห็นได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลซึ่งเหมือนกับระบบแบบพาร์ทิชันเดียว ระบบแบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน (หัวข้อ 4.2.1) และระบบแบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน (หัวข้อ 4.2.3.1) อีกทั้งนโยบายแบบใช้เป้าหมายเป็นหลักยังคงสามารถกระจายจำนวนงาน และชนิดของงานไปแต่ละพาร์ทิชันได้ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยจะส่งงานไปทุกพาร์ทิชันให้ใกล้เคียงกันมากที่สุด ซึ่งต่างจากนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลที่เลือกจะส่งงานไปที่พาร์ทิชันแรกก่อน ดังจะเห็นได้จากจำนวนงาน (ตารางที่ 4-10) ลักษณะของงาน (ตารางที่ 4-11) และความต้องการหน่วยประมวลผลของงาน (ตารางที่ 4-12) ในหัวข้อต่อไปจะนำเสนอ และวิเคราะห์ประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักเมื่อระบบเป็นแบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

4.2.3.3 ระบบ 5 พาร์ทิชัน

ในหัวข้อนี้จะแสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 5 พาร์ทิชันซึ่งมีจำนวนหน่วยประมวลผลเท่ากันคือ 32 โหนด โดยใช้เวิร์คโหลดในตารางที่ 3-3 จะเห็นได้ว่าค่าโหลดของระบบ 5 พาร์ทิชันจะลดต่ำกว่าค่าโหลดในเวิร์คโหลดต้นฉบับมาก (ตารางที่ 3-1) ซึ่งลดลงประมาณ 50% จากค่าโหลดในเวิร์คโหลดต้นฉบับ ผลที่ได้จากการทดสอบนโยบายจะแสดงเป็น 3 หัวข้อย่อยคือ ประสิทธิภาพโดยรวมของระบบ (หัวข้อ 4.2.3.3.1) ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน (หัวข้อ 4.2.3.3.2) และวิเคราะห์ประสิทธิภาพ (หัวข้อ 4.2.3.3.3)

4.2.3.3.1 ประสิทธิภาพโดยรวมของระบบ

หัวข้อนี้จะแสดงการเปรียบเทียบประสิทธิภาพโดยรวมทั้งระบบของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน ดังแสดงในภาพประกอบที่ 4-12



(ก) ค่าสูงสุดของการรอ (ข) ค่าเฉลี่ยการรอ

(ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ง) 98 เปอร์เซ็นไทล์ของการรอ

ภาพประกอบ 4-12 ประสิทธิภาพบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

ภาพประกอบที่ 4-12 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน โดยมีจำนวนหน่วยประมวลผลแต่ละพาร์ทิชันเท่ากับ 32 โหนด และแสดงเป็นกราฟประสิทธิภาพแต่ละเดือนจำนวน 10 เดือนตั้งแต่เดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งเป็นประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่าค่าสูงสุดการรอ (ภาพประกอบที่ 4-12 (ก)) และค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-12 (ข)) ของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองมีค่าใกล้เคียงกันมาก แต่นโยบายแบบเอฟซีเอฟเอสแบ็กฟิลยังคงให้ค่าสูงสุดการรอดีกว่านโยบายแบบเอลเอกซ์เอฟแบ็กฟิล โดยดีกว่าเพียงเล็กน้อยเท่านั้น ขณะที่นโยบายแบบเอลเอกซ์เอฟแบ็กฟิลยังคงให้ค่าเฉลี่ยการรอดีกว่านโยบายแบบเอฟซีเอฟเอสแบ็กฟิล โดยดีกว่าเพียงเล็กน้อยเช่นกัน จะเห็นได้ว่านโยบายทั้งสองตัวทำงานดีในเป้าหมายอันใดอันหนึ่งเท่านั้น แต่นโยบายแบบใช้เป้าหมายเป็นหลักให้ค่าประสิทธิภาพการทำงานทั้งค่าสูงสุดการรอ ค่าเฉลี่ยการรอ

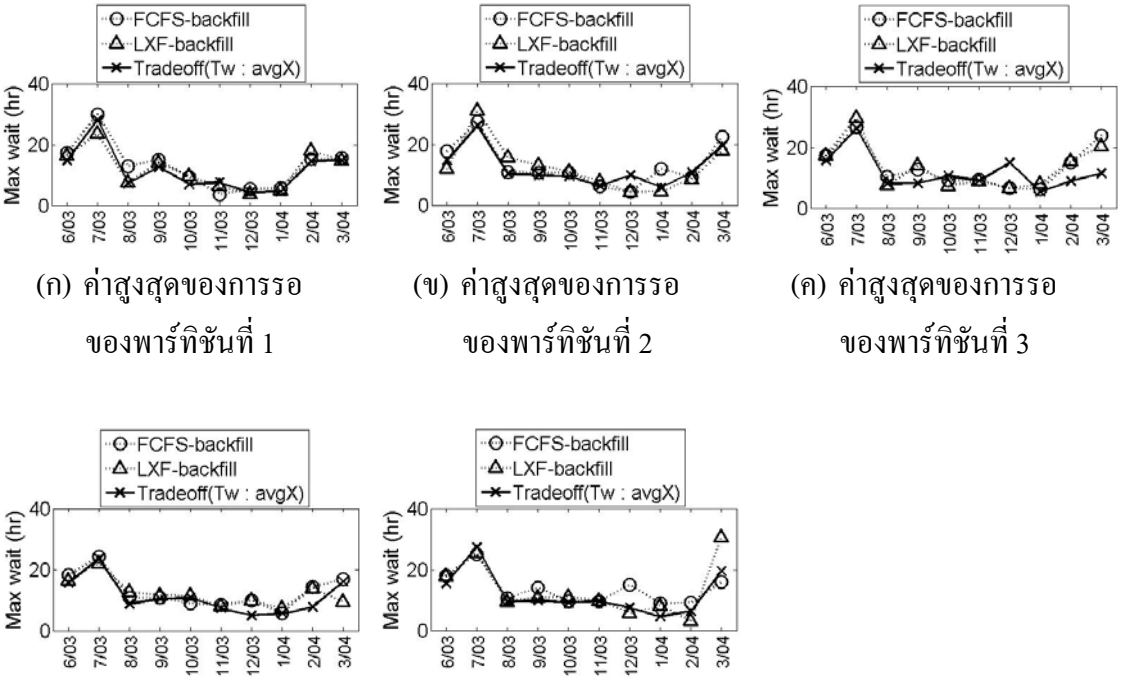
ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และค่า 98 เปอร์เซ็นไทล์ของการรอ ดีหรือใกล้เคียงค่าที่ดีที่สุด โดยจะอยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง ยกเว้นค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ในเดือนมกราคมปี ค.ศ. 2004 ซึ่งค่าจากนโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล อันเนื่องมาจากกลุ่มงานสั้นบางกลุ่มที่ต้องรอนาน แต่จะเห็นได้ว่าค่าเฉลี่ยการรอของเดือนนี้จากนโยบายแบบใช้เป้าหมายเป็นหลักยังคงอยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง

โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-12 นโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเมื่อระบบเป็นแบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน โดยให้ประสิทธิภาพที่ดีทั้งค่าสูงสุดการรอ และค่าเฉลี่ยการรอ แม้ว่าตัววัดประสิทธิภาพทั้ง 2 จะขัดแย้งกันก็ตาม

4.2.3.3.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน

จากหัวข้อ 4.2.3.3.1 ได้เปรียบเทียบประสิทธิภาพโดยภาพรวมของทั้งระบบ โดยแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ดังนั้นหัวข้อนี้จะวิเคราะห์หาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ประสิทธิภาพของแต่ละนโยบายบนแต่ละพาร์ทิชัน ดังแสดงในภาพประกอบที่ 4-13, 4-14 และ 4-15 ทั้งนี้แนวนอนแสดงเดือนที่วัด และแนวตั้งแสดงประสิทธิภาพต่าง ๆ อันได้แก่ ค่าสูงสุดการรอ ค่าเฉลี่ยการรอ และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ แต่ละค่าจะแสดงเป็น 5 ภาพ โดยภาพประกอบที่ 4-13 แสดงผลของค่าสูงสุดการรอของแต่ละนโยบายแต่ละพาร์ทิชัน ในขณะที่ภาพประกอบที่ 4-14 แสดงผลของค่าเฉลี่ยการรอของแต่ละนโยบายแต่ละพาร์ทิชัน และภาพประกอบที่ 4-15 แสดงผลของค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของแต่ละนโยบายแต่ละพาร์ทิชัน

จากภาพประกอบที่ 4-13 จะเห็นว่าเมื่อแยกแสดงแต่ละพาร์ทิชัน นโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีหรือใกล้เคียงค่าที่ดี ยกเว้นเดือนธันวาคมปี ค.ศ. 2003 บนพาร์ทิชันที่ 2 และ 3 และเดือนกรกฎาคมปี ค.ศ. 2003 บนพาร์ทิชันที่ 5 ซึ่งนโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล เมื่อดูถึงประสิทธิภาพของค่าสูงสุดการรอของทั้งระบบในภาพประกอบที่ 4-12 จะเห็นได้ว่าประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักดีกว่าหรืออยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง



(ก) ค่าสูงสุดของการรอของพาร์ทิชันที่ 1 (ข) ค่าสูงสุดของการรอของพาร์ทิชันที่ 2 (ค) ค่าสูงสุดของการรอของพาร์ทิชันที่ 3

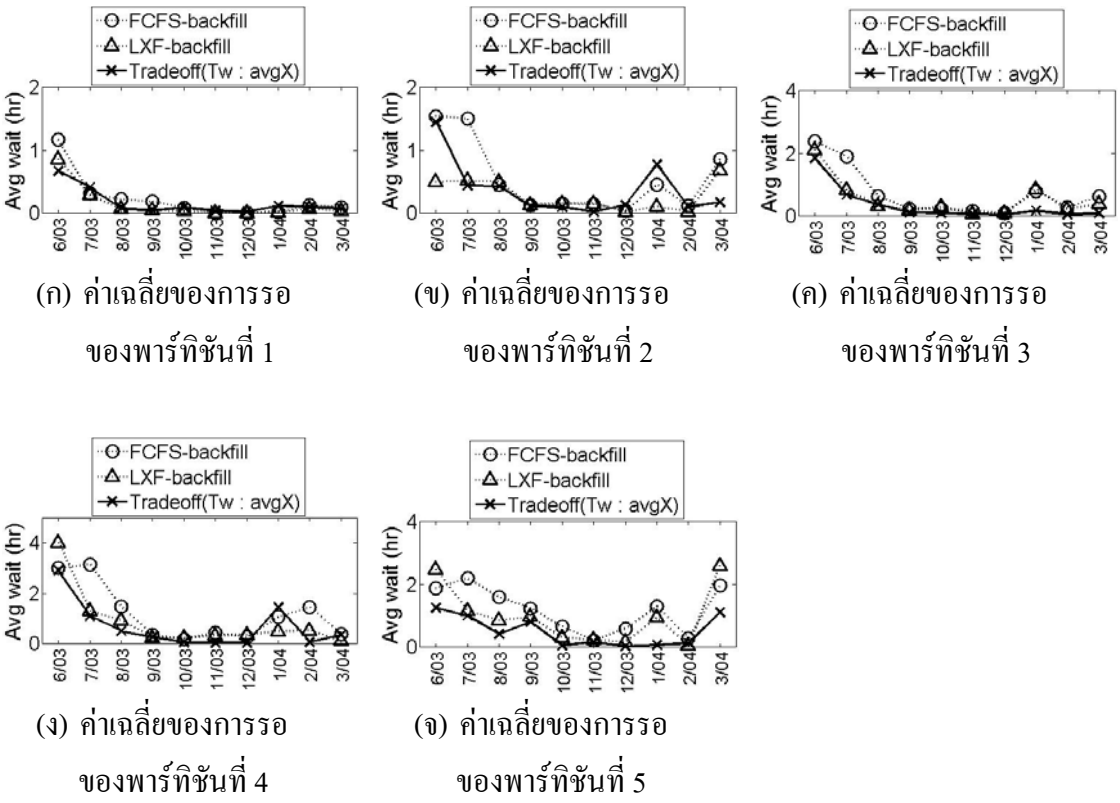
(ง) ค่าสูงสุดของการรอของพาร์ทิชันที่ 4 (จ) ค่าสูงสุดของการรอของพาร์ทิชันที่ 5

ภาพประกอบ 4-13 ค่าสูงสุดของการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

จากภาพประกอบที่ 4-14 จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีเกือบทุกเดือน ยกเว้นเดือนมกราคมปี ค.ศ. 2004 ที่พาร์ทิชัน 2 และ 4 ซึ่งนโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลอยู่เพียงเล็กน้อยเท่านั้น เมื่อดูถึงประสิทธิภาพของค่าเฉลี่ยการรอของทั้งระบบในภาพประกอบที่ 4-12 จะเห็นได้ว่าประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักดีกว่าหรืออยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง

จากภาพประกอบที่ 4-15 จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีเกือบทุกเดือน ยกเว้นเดือนมกราคมปี ค.ศ. 2004 ที่พาร์ทิชัน 2 และ 4 ซึ่งนโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลมาก เมื่อดูถึงประสิทธิภาพของค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของทั้งระบบในภาพประกอบที่ 4-12 จะเห็นได้ว่าเฉพาะเดือนมกราคมปี ค.ศ. 2004 ประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล

ทั้งสองเล็กน้อย จึงทำให้ค่าในแต่ละพาร์ทิชันแย่ตามไปด้วยเพราะกลุ่มงานสั้นบางกลุ่มที่ต้องรอนานทำให้ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แย่กว่าด้วย

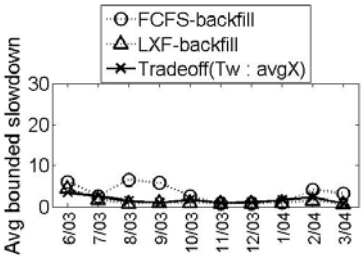


(ก) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 1 (ข) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 2 (ค) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 3

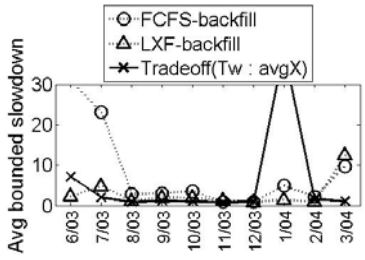
(ง) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 4 (จ) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 5

ภาพประกอบ 4-14 ค่าเฉลี่ยการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

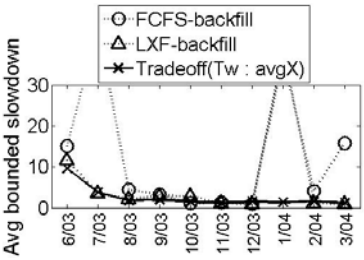
โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-12 ซึ่งเป็นแบบรวมทั้งระบบ และภาพประกอบที่ 4-13, 4-14 และ 4-15 ซึ่งเป็นแบบแยกพาร์ทิชันจะเห็นได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลถึงแม้จำนวนพาร์ทิชันจะเพิ่มขึ้นเป็น 5 พาร์ทิชันแล้วก็ตาม โดยให้ประสิทธิภาพที่ดีทั้งค่าสูงสุดการรอ ค่าเฉลี่ยการรอ และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์



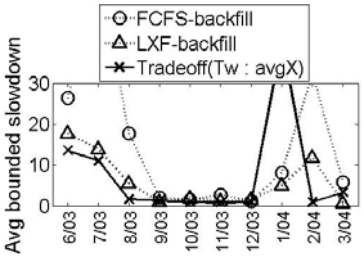
(ก) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 1



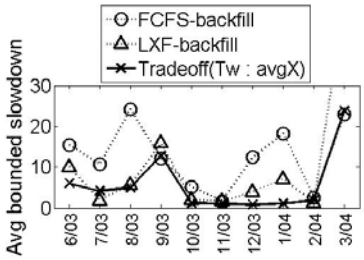
(ข) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 2



(ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 3



(ง) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 4



(จ) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 5

ภาพประกอบ 4-15 ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

4.2.3.3.3 วิเคราะห์ประสิทธิภาพ

จากหัวข้อที่ 4.2.3.3.1 และ 4.2.3.3.2 ก่อนหน้าได้แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ในหัวข้อนี้จะตรวจสอบหาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ถึงจำนวนงาน และปริมาณงานในแต่ละพาร์ทิชันโดยตารางที่ 4-13 แสดงจำนวนงานที่ถูกส่งไปประมวลผลในแต่ละพาร์ทิชันของแต่ละนโยบาย ในขณะที่ตารางที่ 4-14 แสดงลักษณะของงาน (จำนวนหน่วยประมวลผล × เวลาที่ใช้) มีหน่วยเป็น จำนวนหน่วยประมวลผลชั่วโมง และตารางที่ 4-15 แสดงค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานมีหน่วยเป็น จำนวนหน่วยประมวลผล เพื่อให้ง่ายต่อการวิเคราะห์ค่าในแต่ละพาร์ทิชันจึงใช้ค่าความแปรปรวนเพิ่มมาแต่ละนโยบาย ซึ่งค่าความแปรปรวนจะสามารถแสดงการกระจายของค่าในแต่ละพาร์ทิชันของแต่ละนโยบาย ถ้าค่าความแปรปรวนน้อยแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นใกล้เคียงกันมาก แต่ถ้าค่าความแปรปรวนมากแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นห่างกันมาก

ตารางที่ 4-13 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

Number of jobs																		
	FCFS-backfill						LXF-backfill						Goal-oriented					
Partition / Month	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance
6/03	695	261	509	357	369	22,748.2	780	425	531	223	232	42,888.6	390	491	584	299	427	**9,174.2**
7/03	612	355	208	143	82	35,801.2	559	353	195	151	142	25,216.0	404	369	238	217	172	**8,138.8**
8/03	2,053	301	353	252	262	497,439.8	1,969	311	411	294	236	441,953.4	1,632	320	398	337	534	**249,597.0**
9/03	2,035	465	297	123	137	522,101.4	1,976	489	327	119	146	483,410.6	1,546	563	446	300	202	**233,551.0**
10/03	2,977	565	230	224	153	1,173,079.8	3,008	554	241	216	130	1,206,755.4	1,552	596	690	523	788	**138,330.6**
11/03	2,623	234	303	135	148	939,194.6	2,559	281	321	140	142	879,879.4	1,038	1,198	365	654	188	**147,616.6**
12/03	2,523	447	247	148	156	838,619.8	2,532	442	245	150	152	846,505.8	856	320	682	618	1,045	**58,944.2**
1/04	1,803	540	446	137	230	364,185.4	1,773	547	462	127	247	348,250.6	1,029	611	484	326	706	**55,812.6**
2/04	2,079	880	463	276	271	462,007.0	2,112	958	279	256	364	500,717.8	855	682	1,081	710	641	**25,819.8**
3/04	2,410	369	217	350	120	745,123.8	2,319	416	272	342	117	670,564.6	1,282	673	826	290	395	**123,244.6**

ตารางที่ 4-14 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

Average job size (NT) in node-hour																		
	FCFS-backfill						LXF-backfill						Goal-oriented					
Partition / Month	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance
6/03	21.2	49.3	24.4	32.2	28.5	96.4	18.8	29.0	23.3	53.0	46.9	179.5	37.1	26.7	20.8	39.6	24.5	**53.6**
7/03	19.1	32.3	47.5	54.8	83.4	475.9	22.8	33.0	51.5	44.0	46.7	107.4	30.8	31.2	42.9	32.5	37.5	**21.4**
8/03	5.5	43.5	27.9	37.6	28.2	167.5	5.8	40.7	23.7	32.5	32.8	142.4	6.9	41.1	25.5	27.8	13.3	**142.0**
9/03	5.2	23.7	37.9	68.7	52.4	485.9	5.5	22.7	34.4	68.9	48.3	468.7	7.1	20.7	24.7	25.1	35.8	**85.7**
10/03	5.0	22.8	51.8	45.0	47.8	318.7	5.0	22.8	47.6	47.2	60.0	394.0	8.3	24.3	15.4	18.6	11.9	**30.3**
11/03	4.3	55.4	33.3	56.6	36.2	361.2	4.5	45.6	30.9	54.8	39.1	293.7	11.8	10.0	25.6	13.4	27.1	**52.7**
12/03	6.7	29.5	44.0	68.2	42.4	403.8	6.7	29.7	44.3	68.2	42.8	404.6	13.7	39.0	14.8	17.9	11.7	**99.9**
1/04	9.3	25.1	23.5	72.9	28.4	464.2	9.6	24.9	22.5	69.5	30.4	409.9	16.2	21.1	22.0	21.7	14.3	**10.1**
2/04	8.3	13.9	26.1	33.0	24.2	78.4	8.1	12.9	43.9	34.9	17.8	185.6	16.2	17.2	11.8	15.2	12.5	**4.4**
3/04	6.9	36.3	59.1	30.7	72.7	524.7	7.1	34.4	48.1	30.8	67.8	401.8	11.9	20.7	16.4	33.9	24.9	**56.8**

ตารางที่ 4-15 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน

Average N																		
	FCFS-backfill						LXF-backfill						Goal-oriented					
Partition / Month	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance
6/03	7.1	13.3	8.3	10.5	8.9	4.59	6.5	8.9	9.0	13.9	12.7	7.35	10.5	8.6	7.7	11.8	7.8	**2.60**
7/03	8.7	13.4	17.9	22.0	20.8	24.23	9.3	14.3	16.5	20.4	14.7	12.89	12.4	13.7	14.4	13.5	12.9	**0.47**
8/03	2.2	8.9	7.3	11.0	11.6	11.34	2.3	9.0	6.7	9.5	12.0	10.68	2.5	8.3	7.2	7.3	6.6	**4.06**
9/03	3.3	8.6	12.6	24.2	18.7	54.08	3.0	9.1	12.0	24.1	18.8	54.45	3.5	7.4	9.0	11.2	14.8	**14.23**
10/03	2.2	6.5	11.6	11.1	17.7	27.20	2.1	6.6	12.9	11.1	19.8	35.68	3.1	5.6	5.5	5.6	4.1	**1.03**
11/03	2.0	13.1	11.8	20.3	9.8	34.64	2.1	11.2	11.0	19.4	10.4	30.00	4.3	3.2	8.2	5.2	7.3	**3.45**
12/03	2.2	5.7	11.4	15.4	14.6	26.30	2.2	5.5	11.5	16.0	14.7	28.29	5.0	10.0	3.9	4.0	2.8	**6.39**
1/04	2.3	9.9	22.4	27.1	28.9	106.60	2.3	9.3	21.4	27.4	29.6	111.33	5.2	13.7	7.2	29.0	4.5	**83.48**
2/04	2.3	4.1	7.4	9.8	5.8	6.73	2.2	3.6	11.1	11.9	4.9	15.90	4.0	4.3	3.8	4.1	4.2	**0.03**
3/04	2.0	9.6	11.2	6.1	19.9	35.71	2.0	9.4	9.9	5.3	18.1	29.29	2.5	5.4	4.5	6.4	7.3	**2.73**

จากตารางที่ 4-13, 4-14 และ 4-15 จะเห็นได้ว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งเอฟซีเอฟเอส และเอลเอกซ์เอฟมีจำนวนงานประมวลผลบนพาร์ทิชันที่ 1 มากกว่าพาร์ทิชันอื่น ๆ ทุกเดือน ในขณะที่ขนาดของงานในตารางที่ 4-14 และ 4-15 ที่พาร์ทิชันแรกของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเป็นงานที่มีขนาดเล็กกว่าพาร์ทิชันอื่น ๆ นั่นแสดงว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลจะส่งงานที่ต้องการหน่วยประมวลผลน้อยไปประมวลผลบนพาร์ทิชันแรกก่อน ซึ่งต่างจากนโยบายแบบใช้เป้าหมายเป็นหลักที่สามารถกระจายจำนวนงาน งานลักษณะต่าง ๆ และความต้องการหน่วยประมวลผลของงานไปแต่ละพาร์ทิชันได้ใกล้เคียงกันมากกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล เมื่อวิเคราะห์ค่าความแปรปรวนจะเห็นได้ชัดเจนว่าค่าแต่ละพาร์ทิชันของนโยบายแบบใช้เป้าหมายเป็นหลักใกล้เคียงกันมากกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทุกเดือน

จากผลการทดสอบในภาพประกอบที่ 4-12 ซึ่งเป็นแบบรวมทั้งระบบ และแบบแยกแต่ละพาร์ทิชันในภาพประกอบที่ 4-13, 4-14 และ 4-15 จะเห็นได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ซึ่งเหมือนกับระบบแบบพาร์ทิชันเดียว ระบบแบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน (หัวข้อ 4.2.1) ระบบแบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน (หัวข้อ 4.2.3.1) และระบบแบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน (หัวข้อ 4.2.3.2) อีกทั้งนโยบายแบบใช้เป้าหมายเป็นหลักยังคงสามารถกระจายจำนวนงาน และงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชันได้ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลโดยจะส่งงานไปทุกพาร์ทิชันให้ใกล้เคียงกันมากที่สุด ซึ่งต่างจากนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ที่เลือกจะส่งงานไปที่พาร์ทิชันแรกก่อน ดังจะเห็นได้จากจำนวนงาน (ตารางที่ 4-13) ลักษณะของงาน (ตารางที่ 4-14) และความต้องการหน่วยประมวลผลของงาน (ตารางที่ 4-15)

จากที่ได้กล่าวมาในหัวข้อ 4.2.1 และ 4.2.3 แสดงให้เห็นถึงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบพาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อเพิ่มจำนวนพาร์ทิชันตั้งแต่ 2 ถึง 5 พาร์ทิชัน นโยบายแบบใช้เป้าหมายเป็นหลักยังคงดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเพียงแต่ไม่ชัดเจนเท่ากับระบบแบบ 2 พาร์ทิชัน อันเนื่องมาจากโหลดของระบบที่ลดลง และทรัพยากรที่มีเพิ่มขึ้น กล่าวคือเมื่อระบบเป็น 5 พาร์ทิชันจำนวนหน่วยประมวลผลรวมของทั้งระบบจะเท่ากับ 160 โหนด ซึ่งเวิร์คโหลดต้นฉบับ (ตารางที่ 3-1) ที่นำมาทดสอบมีหน่วยประมวลผลเพียง 128 โหนดเท่านั้น ในงานวิจัย [6] และ [13] ที่นำเสนอประสิทธิภาพของนโยบาย

แบบใช้เป้าหมายเป็นหลักได้กล่าวว่านโยบายแบบใช้เป้าหมายเป็นหลักจะมีประสิทธิภาพสูงก็ต่อเมื่อระบบมีโหลดที่สูง อีกทั้งประสิทธิภาพจากนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองจะใกล้เคียงกันมากขึ้นเมื่อจำนวนพาร์ทิชันที่ใช้ทดสอบเพิ่มขึ้นทำให้ประสิทธิภาพจากนโยบายแบบใช้เป้าหมายเป็นหลักในบางเดือนใกล้เคียงกับค่าที่แย่ของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ในหัวข้อต่อไปจะนำเสนอ และวิเคราะห์ประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักเมื่อเปลี่ยนค่าการจำกัดจำนวนโหนด

4.3 ประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักเมื่อเปลี่ยนค่าการจำกัดจำนวนโหนด

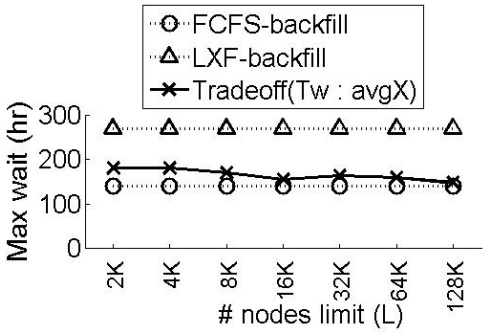
การใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานใน [6] ได้กล่าวไว้ว่าหากระบบสามารถให้เวลาในการค้นหามากขึ้นประสิทธิภาพที่ได้จะดีขึ้นหรือไม่แย่ไปกว่าเดิม ดังนั้นในหัวข้อนี้จะกล่าวถึงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักเมื่อเพิ่มเวลาที่ให้กับการค้นหาเพิ่มขึ้น ซึ่งเวลาในงานวิจัยชิ้นนี้คือจำนวนโหนดที่ค้นหาพบ ในตารางที่ 4-16 จะแสดงสัญลักษณ์แทนจำนวนโหนดที่จะใช้ทดสอบในหัวข้อนี้ โดยเริ่มตั้งแต่ 2,000 ถึง 128,000 โหนด ทั้งนี้นโยบายแบบใช้เป้าหมายเป็นหลักที่นำมาทดสอบคือแบบการใช้เบสฟีทเป็นฮิวริสทิคในการเลือกพาร์ทิชัน เนื่องจากให้ประสิทธิภาพการทำงานที่ดีกว่าดังแสดงในหัวข้อ 4.1

ตารางที่ 4-16 สัญลักษณ์แทนจำนวนโหนดที่ใช้ในการทดสอบระบบ

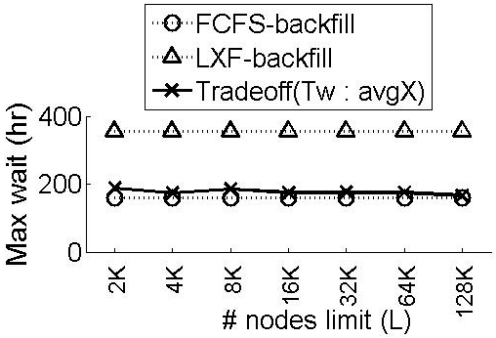
สัญลักษณ์	2K	4K	8K	16K	32K	64K	128K
จำนวนโหนด	2,000	4,000	8,000	16,000	32,000	64,000	128,000

ภาพประกอบที่ 4-16 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักเมื่อเพิ่มจำนวนโหนดในการค้นหา โดยทำการทดสอบบนระบบคอมพิวเตอร์แบบขนานแบบ 2 พาร์ทิชัน โดยแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากัน โดยพาร์ทิชันที่ 1 มีจำนวนหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผล 64 โหนด ซึ่งแสดงเฉพาะเดือนมกราคมปี ค.ศ. 2004 เท่านั้น แกนแนวนอนแสดงจำนวนโหนดที่ใช้ในการทดสอบ ส่วนแกนแนวตั้งแสดงประสิทธิภาพการทำงาน ข้อมูลจากภาพประกอบที่ 4-16 (ก), (ข) แสดงค่าสูงสุดของการรอ จะเห็นว่าที่พาร์ทิชันแรก (ภาพประกอบที่ 4-16 (ก)) เมื่อจำนวนโหนดที่ใช้เพิ่มขึ้นประสิทธิภาพจะดีขึ้นเรื่อย ๆ ซึ่งประสิทธิภาพที่ดีขึ้นนี้ไม่ส่งผลกระทบต่อประสิทธิภาพในพาร์ทิชันที่ 2 (ภาพประกอบที่ 4-16 (ข)) และเมื่อพิจารณาค่าเฉลี่ยการรอพบว่าเมื่อจำนวนโหนดที่ใช้เพิ่มขึ้น

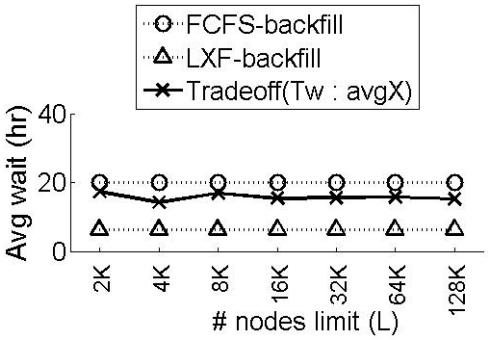
ประสิทธิภาพก็ดีขึ้นด้วยซึ่งจะเห็นได้ชัดในพาร์ทิชันแรก (ภาพประกอบที่ 4-16 (ค)) อีกทั้งเมื่อพิจารณาจำนวนงานที่รอมากกว่า 24 ชั่วโมง (ภาพประกอบที่ 4-16 (จ), (ฉ)) พบว่าเมื่อจำนวนโหนดที่ใช้เพิ่มขึ้นจำนวนงานที่รอมากกว่า 24 ชั่วโมงจะปรับเปลี่ยนเล็กน้อยทั้ง 2 พาร์ทิชัน



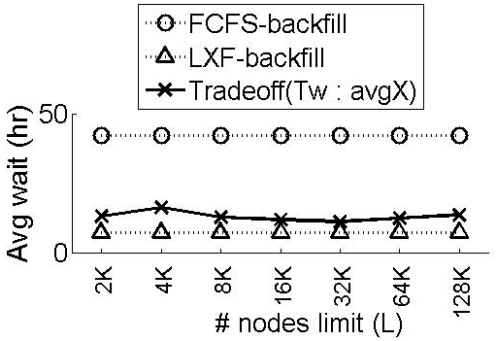
(ก) ค่าสูงสุดของการรอของพาร์ทิชันที่ 1



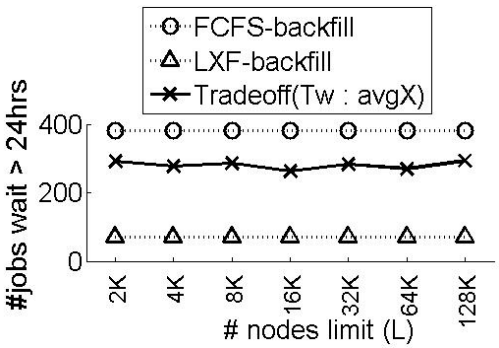
(ข) ค่าสูงสุดของการรอของพาร์ทิชันที่ 2



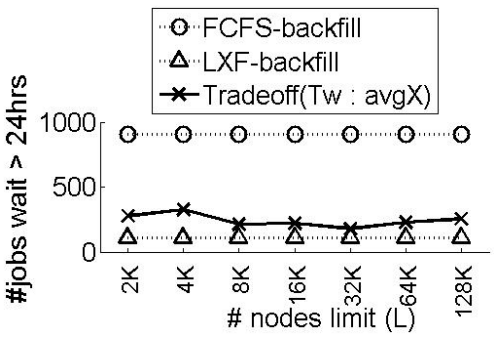
(ค) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 1



(ง) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 2



(จ) จำนวนงานที่รอมากกว่า 24 ชั่วโมง ของพาร์ทิชันที่ 1



(ฉ) จำนวนงานที่รอมากกว่า 24 ชั่วโมง ของพาร์ทิชันที่ 2

ภาพประกอบ 4-16 ประสิทธิภาพเมื่อเปลี่ยนค่าการจำกัดจำนวนโหนด

จากผลการทดสอบสรุปได้ว่าหากให้เวลาสำหรับการค้นหาเส้นทางของนโยบายแบบใช้เป้าหมายเป็นหลักเพิ่มขึ้นตัวจัดกำหนดการจะสามารถค้นหาเส้นทางที่เหมาะสมที่สุดได้ ซึ่งส่งผลโดยตรงต่อประสิทธิภาพของนโยบาย ดังเช่นในภาพประกอบ 4-16 ที่ทำให้ได้ค่าสูงสุดการรอลดลงเรื่อย ๆ เมื่อเพิ่มจำนวนโหนดที่ใช้ การทดสอบจากหัวข้อก่อนหน้าที่ได้กล่าวมาทั้งหมดรวมทั้งหัวข้อนี้จะใช้ค่าระยะเวลาการประมวลผลจริง ซึ่งในหัวข้อถัดไปจะนำเสนอ และวิเคราะห์ประสิทธิภาพทุก ๆ นโยบายบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันโดยใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

4.4 เปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

หัวข้อที่ 4.1 ถึง 4.3 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักโดยใช้ระยะเวลาการประมวลผลจริง เพื่อทดสอบประสิทธิภาพของนโยบายว่ายังสามารถทำงานได้ดีเช่นเดียวกับระบบแบบพาร์ทิชันเดียวหรือไม่ เมื่อระบบที่ใช้ทดสอบเป็นแบบหลายพาร์ทิชันที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากัน และไม่เท่ากัน ซึ่งในระบบคอมพิวเตอร์แบบขนานที่ใช้งานจริงในปัจจุบัน ตัวจัดกำหนดการจะไม่สามารถรู้ค่าระยะเวลาการประมวลผลจริงของงานได้เลยจนกว่างานนั้นจะถูกประมวลผลเสร็จ โดยจะรู้เพียงค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดในช่วงส่งงานเข้าสู่ระบบเท่านั้น ดังนั้นในหัวข้อนี้จะทดสอบประสิทธิภาพของนโยบายที่ออกแบบเมื่อตัวจัดกำหนดการงานแบบขนานใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดแทนค่าระยะเวลาการประมวลผลจริงในการตัดสินใจ โดยทดสอบบนระบบที่มีจำนวนหน่วยประมวลผลเท่ากัน (หัวข้อ 4.4.1) จำนวน 2 ถึง 5 พาร์ทิชัน และแบบที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน (หัวข้อ 4.4.2) ทั้งนี้นโยบายแบบใช้เป้าหมายเป็นหลักที่นำมาทดสอบคือแบบการใช้ฮิวริสทิคในการเลือกพาร์ทิชันโดยใช้เบสฟิทในการเลือกพาร์ทิชัน เนื่องจากให้ประสิทธิภาพการทำงานที่ดีกว่าดังได้แสดงในหัวข้อ 4.1 และจำกัดจำนวนโหนดในการค้นหาเท่ากับ 6,000 โหนด

4.4.1 ประสิทธิภาพบนระบบหลายพาร์ทิชันแบบจำนวนหน่วยประมวลผลเท่ากัน

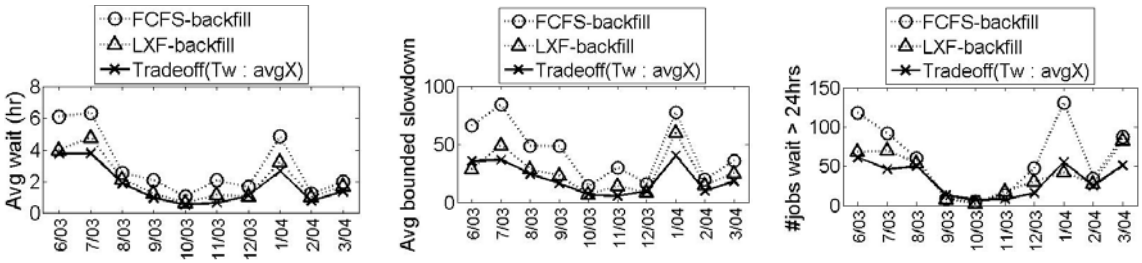
หัวข้อนี้จะทดสอบ และวิเคราะห์ประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนานที่มีจำนวนหน่วยประมวลผลเท่ากัน ซึ่งจะทดสอบโดยใช้จำนวนพาร์ทิชันตั้งแต่ 2 ถึง 5 พาร์ทิชัน

4.4.1.1 ระบบ 2 พาร์ทิชัน

ในหัวข้อนี้จะแสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันคือ 64 โหนด การทดสอบในหัวข้อนี้ใช้เวิร์คโหลดไฟล์ซึ่งแสดงรายละเอียดไว้ในตารางที่ 3-2 และทำการทดสอบเช่นเดียวกับหัวข้อที่ 4.2.1 เพียงแต่เปลี่ยนค่าระยะเวลาการประมวลผลที่ใช้จากค่าระยะเวลาการประมวลผลจริงเป็นค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด ผลที่ได้จากการทดสอบนโยบายจะแสดงเป็น 3 หัวข้อย่อยคือ ประสิทธิภาพโดยรวมของระบบ (หัวข้อ 4.4.1.1.1) ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน (หัวข้อ 4.4.1.1.2) และวิเคราะห์ประสิทธิภาพ (หัวข้อ 4.4.1.1.3)

4.4.1.1.1 ประสิทธิภาพโดยรวมของระบบ

หัวข้อนี้จะแสดงการเปรียบเทียบประสิทธิภาพโดยรวมทั้งระบบของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน โดยใช้ค่าระยะเวลาการประมวลผลเป็นค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด ดังแสดงในภาพประกอบที่ 4-17



(ก) ค่าเฉลี่ยการรอ (ข) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ค) จำนวนงานที่รอมากกว่า 24 ชั่วโมง

ภาพประกอบ 4-17 ประสิทธิภาพบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

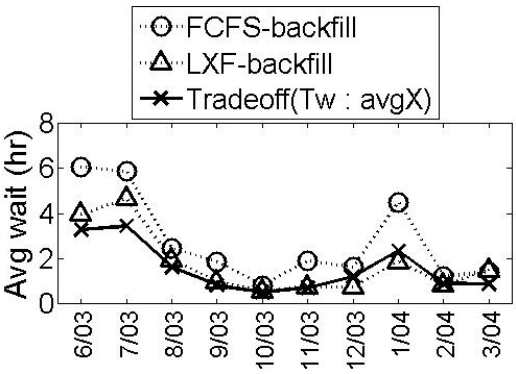
ภาพประกอบที่ 4-17 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วย

ประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด โดยมีจำนวนหน่วยประมวลผลแต่ละพาร์ทิชันเท่ากับ 64 โหนด และแสดงเป็นกราฟจำนวน 10 เดือนตั้งแต่เดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งแสดงประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่านโยบายแบบเอลเอกซ์เอฟแบ็กฟิลให้ประสิทธิภาพที่ดีกว่านโยบายแบบเอฟซีเอฟเอสแบ็กฟิลทั้งค่าเฉลี่ยการรอค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แต่จะมีจำนวนงานที่รอมากกว่า 24 ชั่วโมงเท่ากันในบางเดือน ซึ่งนโยบายแบบใช้เป้าหมายเป็นหลักยังคงมีประสิทธิภาพที่ดีกว่าหรือใกล้เคียงค่าที่ดีของทุกประสิทธิภาพที่แสดงในภาพประกอบที่ 4-17

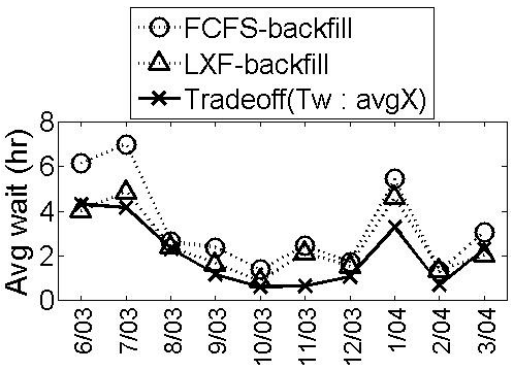
โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-17 นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แม้จะใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดแล้วก็ตาม เช่นเดียวกับบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันที่ใช้ค่าระยะเวลาการประมวลผลจริงดังภาพประกอบที่ 4-2

4.4.1.1.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน

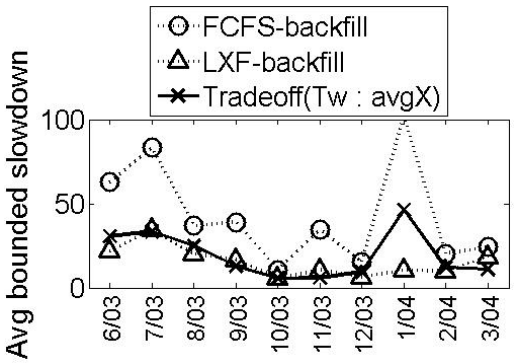
จากหัวข้อ 4.4.1.1.1 ได้เปรียบเทียบประสิทธิภาพโดยภาพรวมของทั้งระบบ โดยแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ดังนั้นหัวข้อนี้จะวิเคราะห์หาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ประสิทธิภาพของแต่ละนโยบายบนแต่ละพาร์ทิชัน ดังแสดงในภาพประกอบที่ 4-18 ทั้งนี้แนวนอนแสดงเดือนที่วัด และแนวตั้งแสดงประสิทธิภาพต่าง ๆ อันได้แก่ ค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แต่ละค่าจะแสดงเป็น 2 ภาพ โดยภาพประกอบที่ 4-18 ก, ค, จ แสดงผลของค่าดังกล่าวของแต่ละนโยบายบนพาร์ทิชันแรก ในขณะที่ภาพประกอบที่ 4-18 ข, ง, ฉ แสดงค่าที่วัดได้ของแต่ละนโยบายบนพาร์ทิชันที่ 2



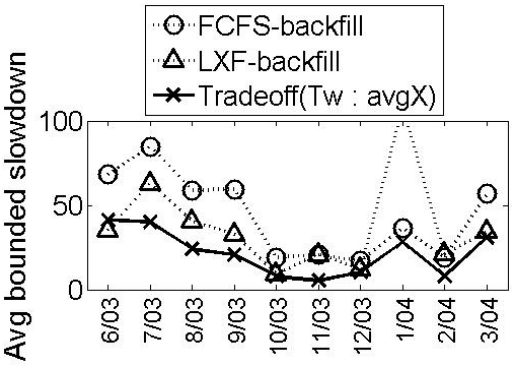
(ก) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 1



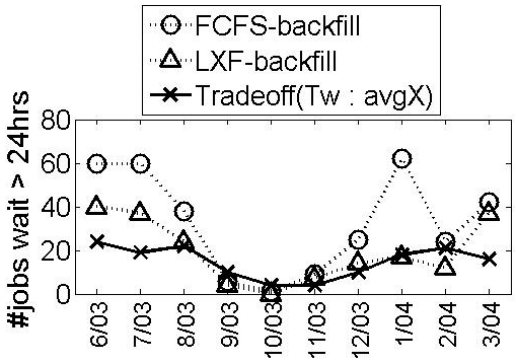
(ข) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 2



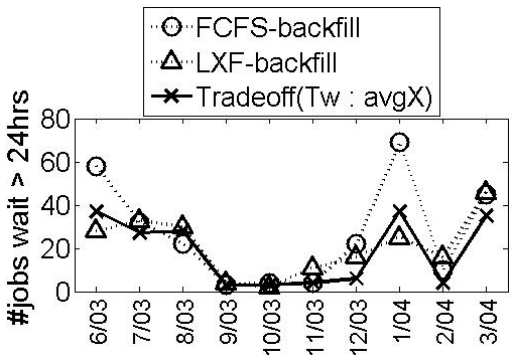
(ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 1



(ง) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 2



(จ) จำนวนงานที่รอมากกว่า 24 ชั่วโมง ของพาร์ทิชันที่ 1



(ฉ) จำนวนงานที่รอมากกว่า 24 ชั่วโมง ของพาร์ทิชันที่ 2

ภาพประกอบ 4-18 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

จากภาพประกอบที่ 4-18 จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีของค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-18 (ก), (ข)) และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ภาพประกอบที่ 4-18 (ค), (ง)) ทั้ง 2 พาร์ทิชัน อีกทั้งจำนวนงานที่รอมากกว่า 24 ชั่วโมง (ภาพประกอบที่ 4-18 (จ), (ฉ)) ก็แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักสามารถทำให้งานที่ต้องรอมากกว่า 24 ชั่วโมงน้อยกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทุกเดือน

โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-17 ซึ่งเป็นแบบรวมทั้งระบบ และ 4-18 ซึ่งเป็นแบบแยกพาร์ทิชัน นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แม้จะทำงานบนระบบ 2 พาร์ทิชัน และใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด เช่นเดียวกับบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันที่ใช้ค่าระยะเวลาการประมวลผลจริงดังภาพประกอบที่ 4-2 และ 4-3

4.4.1.1.3 วิเคราะห์ประสิทธิภาพ

จากหัวข้อที่ 4.4.1.1.1 และ 4.4.1.1.2 ก่อนหน้าได้แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ในหัวข้อนี้จะตรวจสอบหาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ถึงจำนวนงาน และปริมาณงานในแต่ละพาร์ทิชัน ดังแสดงในตารางที่ 4-17 จำนวนงานที่ถูกส่งไปประมวลผลในแต่ละพาร์ทิชันของแต่ละนโยบาย และตารางที่ 4-18 ลักษณะของงาน (จำนวนหน่วยประมวลผล × เวลาที่ใช้) ซึ่งแสดงค่าเฉลี่ยมีหน่วยเป็น จำนวนหน่วยประมวลผลชั่วโมง เพื่อให้ง่ายต่อการวิเคราะห์ผลในแต่ละตารางของแต่ละนโยบายจะเพิ่มคอลัมน์ Diff. สำหรับแสดงค่าผลต่างทั้งสองพาร์ทิชันของแต่ละนโยบาย ตัวหนาจะแสดงค่าที่ดีที่สุดของคอลัมน์ Diff. ในแต่ละเดือน

จากตารางที่ 4-17 จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักมีจำนวนงานประมวลผลบนพาร์ทิชันทั้งสองใกล้เคียงกันมากกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเพียง 3 เดือนเท่านั้นคือ เดือนตุลาคม เดือนพฤศจิกายน และเดือนธันวาคมปี ค.ศ. 2003 ส่วนเดือนอื่น ๆ นโยบายแบบใช้เป้าหมายเป็นหลักมีจำนวนงานแต่ละพาร์ทิชันใกล้เคียงกันมากกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล 1 นโยบาย แต่แย่กว่าอีกนโยบาย

กล่าวคืออยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง ยกเว้นเดือนสิงหาคมปี ค.ศ. 2003 และเดือนมกราคมปี ค.ศ. 2004 ซึ่งจำนวนงานแต่ละพาร์ทิชันจากนโยบายแบบใช้เป้าหมายเป็นหลักแตกต่างกันมาก เมื่อวิเคราะห์ถึงลักษณะงาน (ตารางที่ 4-18) ของ 2 เดือนนี้ พบว่าพาร์ทิชันที่ 2 จะเป็นงานขนาดใหญ่ที่แตกต่างจากพาร์ทิชันแรกมาก เช่นเดียวกับลักษณะงานจากนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองที่จะมีงานขนาดใหญ่ที่พาร์ทิชันที่ 2 เพียงแต่ลักษณะงานจากนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองใกล้เคียงกันมากกว่าลักษณะงานจากนโยบายแบบใช้เป้าหมายเป็นหลัก ส่วนลักษณะงานในเดือนอื่น ๆ จะมีแนวโน้มไปในทางเดียวกับจำนวนงาน กล่าวคือลักษณะงานจากนโยบายแบบใช้เป้าหมายเป็นหลักจะใกล้เคียงกันมากกว่าหรืออยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง

ตารางที่ 4-17 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Number of jobs									
	FCFS-backfill			LXF-backfill			Goal-oriented		
Partition / Month	1st	2nd	Diff.	1st	2nd	Diff.	1st	2nd	Diff.
6/03	886	1,305	419	1,058	1,133	**75**	1,145	1,046	99
7/03	799	601	198	678	722	**44**	735	665	70
8/03	1,495	1,726	**231**	1,848	1,373	475	1,942	1,279	663
9/03	1,662	1,395	**267**	1,779	1,278	501	1,747	1,310	437
10/03	2,382	1,767	615	2,626	1,523	1,103	2,261	1,888	**373**
11/03	2,321	1,122	1,199	2,375	1,068	1,307	1,764	1,679	**85**
12/03	1,996	1,525	471	2,152	1,369	783	1,813	1,708	**105**
1/04	1,964	1,192	772	1,566	1,590	**24**	2,094	1,062	1,032
2/04	2,020	1,949	**71**	2,290	1,679	611	1,886	2,083	197
3/04	2,226	1,240	986	1,994	1,472	**522**	2,228	1,238	990

ตารางที่ 4-18 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Average job size (NT) in node-hour									
	FCFS-backfill			LXF-backfill			Goal-oriented		
Partition / Month	1st	2nd	Diff.	1st	2nd	Diff.	1st	2nd	Diff.
6/03	42.0	28.2	13.8	34.8	32.8	**2.0**	32.2	35.6	3.4
7/03	43.6	58.9	15.3	53.5	46.9	6.6	48.7	51.6	**2.9**
8/03	22.6	18.9	**3.7**	17.8	24.4	6.6	16.7	26.6	9.9
9/03	19.1	21.2	**2.1**	17.6	23.5	5.9	17.3	23.8	6.5
10/03	13.9	18.8	4.9	12.8	21.4	8.6	15.1	17.0	**1.9**
11/03	14.1	26.5	12.4	13.4	28.6	15.2	18.1	18.2	**0.1**
12/03	17.6	22.5	4.9	15.8	25.9	10.1	19.7	19.8	**0.1**
1/04	18.6	25.3	6.7	22.8	19.4	**3.4**	17.0	29.2	12.2
2/04	15.8	16.1	0.3	13.7	19.0	5.3	16.1	15.9	**0.2**
3/04	15.5	27.5	12.0	16.7	23.9	**7.2**	16.3	26.1	9.8

โดยสรุปความสามารถในการกระจายจำนวนงาน และงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชันให้เท่า ๆ กันของนโยบายแบบใช้เป้าหมายเป็นหลักดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองเพียงเล็กน้อยเท่านั้น ซึ่งจะดีกว่าค่าที่แย่ที่สุดของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเป็นส่วนใหญ่

เมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน นโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ซึ่งเหมือนกับระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าระยะเวลาการประมวลผลจริง (หัวข้อ 4.2.1) แต่การกระจายจำนวนงาน และงานลักษณะต่าง ๆ ของนโยบายแบบใช้เป้าหมายเป็นหลักดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเพียงเล็กน้อยเท่านั้น ทำให้ประสิทธิภาพที่ได้ไม่

ชัดเจนมากเหมือนการใช้ค่าระยะเวลาการประมวลผลจริง ในหัวข้อต่อไปจะนำเสนอ และวิเคราะห์ประสิทธิภาพเมื่อระบบเป็นแบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน และใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

4.4.1.2 ระบบ 3 พาร์ทิชัน

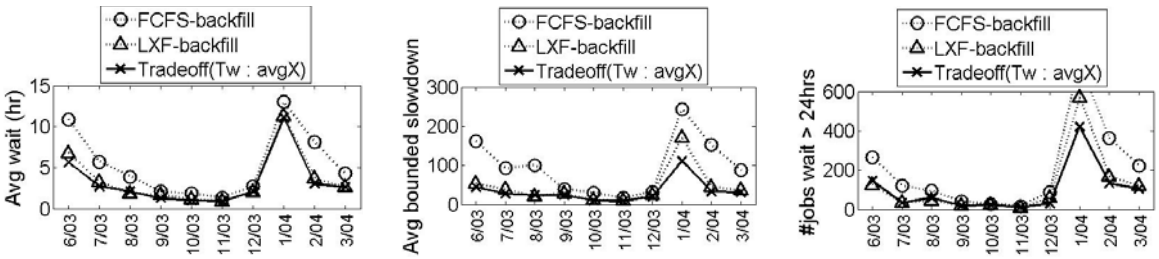
ในหัวข้อนี้จะแสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันคือ 32 โหนด การทดสอบในหัวข้อนี้ใช้เวิร์คโหลดไฟล์ซึ่งแสดงรายละเอียดไว้ในตารางที่ 3-3 จะเห็นได้ว่าค่าโหลดส่วนใหญ่ของระบบ 3 พาร์ทิช์ชันจะสูงกว่าค่าโหลดในเวิร์คโหลดต้นฉบับ (ตารางที่ 3-1) การทดสอบในหัวข้อนี้กระทำเช่นเดียวกับหัวข้อที่ 4.2.3.1 เพียงแต่เปลี่ยนค่าระยะเวลาการประมวลผลที่ใช้จากค่าระยะเวลาการประมวลผลจริงเป็นค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด ผลที่ได้จากการทดสอบนโยบายจะแสดงเป็น 3 หัวข้อย่อยคือ ประสิทธิภาพโดยรวมของระบบ (หัวข้อ 4.4.1.2.1) ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน (หัวข้อ 4.4.1.2.2) และวิเคราะห์ประสิทธิภาพ (หัวข้อ 4.4.1.2.3)

4.4.1.2.1 ประสิทธิภาพโดยรวมของระบบ

หัวข้อนี้จะแสดงการเปรียบเทียบประสิทธิภาพโดยรวมทั้งระบบของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน โดยใช้ค่าระยะเวลาการประมวลผลเป็นค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดดังแสดงในภาพประกอบที่ 4-19

ภาพประกอบที่ 4-19 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน เมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด โดยมีจำนวนหน่วยประมวลผลแต่ละพาร์ทิชันเท่ากับ 32 โหนด และแสดงเป็นกราฟจำนวน 10 เดือนตั้งแต่เดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งแสดงประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-19 (ก)) และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ภาพประกอบที่ 4-19 (ข)) ดีกว่าหรือเท่ากับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล เมื่อพิจารณาจำนวนงานที่รอมากกว่า 24 ชั่วโมง (ภาพประกอบที่ 4-19 (ค)) พบว่านโยบาย

แบบใช้เป้าหมายเป็นหลักยังคงดีกว่าหรือเท่ากับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล



(ก) ค่าเฉลี่ยการรอ (ข) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ค) จำนวนงานที่รอมากกว่า 24 ชั่วโมง

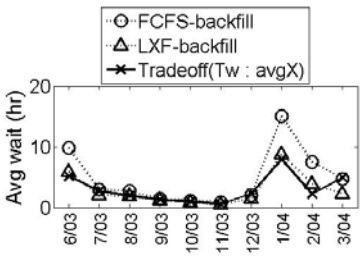
ภาพประกอบ 4-19 ประสิทธิภาพบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-19 นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แม้จะใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดแล้วก็ตาม เช่นเดียวกับบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันที่ใช้ค่าระยะเวลาการประมวลผลจริงดังภาพประกอบที่ 4-6

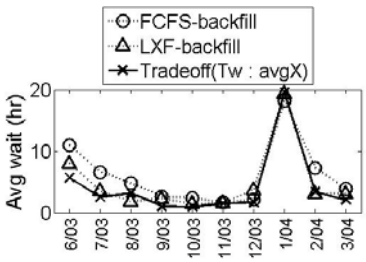
4.4.1.2.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน

จากหัวข้อ 4.4.1.2.1 ได้เปรียบเทียบประสิทธิภาพโดยภาพรวมของทั้งระบบ โดยแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ดังนั้นหัวข้อนี้จะวิเคราะห์หาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ประสิทธิภาพของแต่ละนโยบายบนแต่ละพาร์ทิชัน ดังแสดงในภาพประกอบที่ 4-20 ทั้งนี้แนวนอนแสดงเดือนที่วัด และแนวตั้งแสดงประสิทธิภาพต่าง ๆ อันได้แก่ ค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แต่ละค่าจะแสดงเป็น 3 ภาพ โดยภาพประกอบที่ 4-20 ก, ข, ค แสดงผลของค่าเฉลี่ยการรอของแต่ละนโยบายแต่ละพาร์ทิชัน ในขณะที่ภาพประกอบที่ 4-20 ง, จ, ฉ แสดงผล

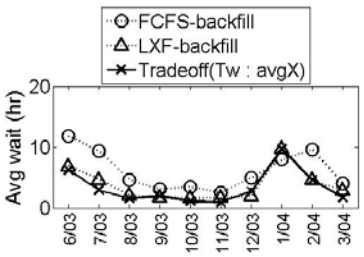
ของค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของแต่ละนโยบายแต่ละพาร์ทิชัน และภาพประกอบที่ 4-20 ช, ซ, ฌ แสดงผลของจำนวนงานที่รอมากกว่า 24 ชั่วโมงของแต่ละนโยบายแต่ละพาร์ทิชัน



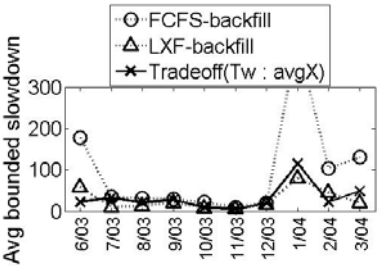
(ก) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 1



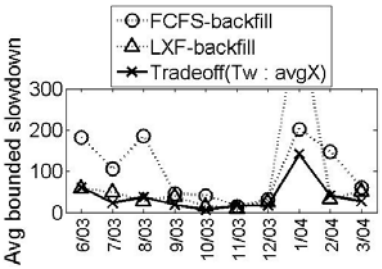
(ข) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 2



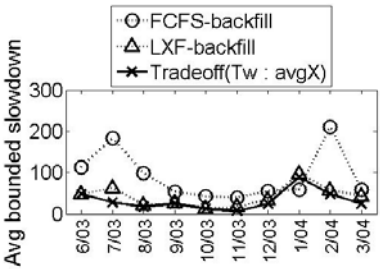
(ค) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 3



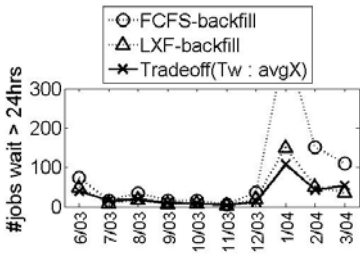
(ง) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 1



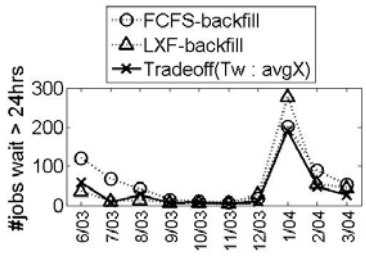
(จ) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 2



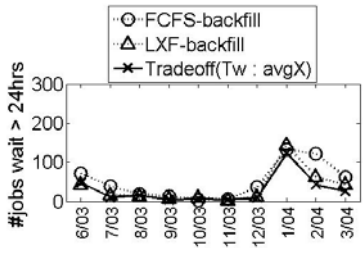
(ฉ) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 3



(ช) จำนวนงานที่รอมากกว่า 24 ชั่วโมงของพาร์ทิชันที่ 1



(ซ) จำนวนงานที่รอมากกว่า 24 ชั่วโมงของพาร์ทิชันที่ 2



(ฌ) จำนวนงานที่รอมากกว่า 24 ชั่วโมงของพาร์ทิชันที่ 3

ภาพประกอบ 4-20 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

จากภาพประกอบที่ 4-20 จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีของค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-20 (ก), (ข), (ค)) และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ภาพประกอบที่ 4-20 (ง), (จ), (ฉ)) ทั้ง 3 พาร์ทิชัน อีกทั้งจำนวนงานที่รอมากกว่า 24 ชั่วโมง (ภาพประกอบที่ 4-20 (ช), (ซ), (ฌ)) ก็แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักสามารถทำให้งานที่ต้องรอมากกว่า 24 ชั่วโมงน้อยกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทุกเดือน

โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-19 ซึ่งเป็นแบบรวมทั้งระบบ และ 4-20 ซึ่งเป็นแบบแยกพาร์ทิชัน นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แม้จะทำงานบนระบบ 3 พาร์ทิชัน และใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด เช่นเดียวกับบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันที่ใช้ค่าระยะเวลาการประมวลผลจริงดังภาพประกอบที่ 4-6 และ 4-7

4.4.1.2.3 วิเคราะห์ประสิทธิภาพ

จากหัวข้อที่ 4.4.1.2.1 และ 4.4.1.2.2 ก่อนหน้าได้แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ในหัวข้อนี้จะตรวจสอบหาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ถึงจำนวนงาน และปริมาณงานในแต่ละพาร์ทิชัน ดังแสดงในตารางที่ 4-19 จำนวนงานที่ถูกส่งไปประมวลผลในแต่ละพาร์ทิชันของแต่ละนโยบาย ตารางที่ 4-20 ลักษณะของงาน (จำนวนหน่วยประมวลผล × เวลาที่ใช้) ซึ่งแสดงค่าเฉลี่ยมีหน่วยเป็น จำนวนหน่วยประมวลผล ชั่วโมง และตารางที่ 4-21 แสดงค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานมีหน่วยเป็น จำนวนหน่วยประมวลผล เพื่อให้ง่ายต่อการวิเคราะห์ค่าในแต่ละพาร์ทิชันจึงใช้ค่าความแปรปรวนเพิ่มมาแต่ละนโยบาย ซึ่งค่าความแปรปรวนสามารถแสดงการกระจายของค่าในแต่ละพาร์ทิชันของแต่ละนโยบาย ถ้าค่าความแปรปรวนน้อยแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นใกล้เคียงกันมาก แต่ถ้าค่าความแปรปรวนมากแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นห่างกันมาก

จากตารางที่ 4-19, 4-20 และ 4-21 จะเห็นได้ว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งเอฟซีเอฟเอส และเอลเอกซ์เอฟมีจำนวนงานประมวลผลบนพาร์ทิชันที่ 1 มากกว่าพาร์ทิชันอื่น ๆ ในขณะที่ขนาดของงานในตารางที่ 4-20 และ 4-21 ที่พาร์ทิชันแรกของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเป็นงานที่มีขนาดเล็กกว่าพาร์ทิชันอื่น ๆ

นั่นแสดงว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลจะส่งงานขนาดเล็กไปประมวลผลบนพาร์ทิชันแรกก่อน จึงทำให้พาร์ทิชันแรกมีจำนวนงานมากแต่มีค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานน้อย ซึ่งต่างจากนโยบายแบบใช้เป้าหมายเป็นหลักที่สามารถกระจายจำนวนงาน งานลักษณะต่าง ๆ และความต้องการหน่วยประมวลผลของงานไปแต่ละพาร์ทิชันได้ใกล้เคียงกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ตัวอย่างเช่นเดือนตุลาคมปี ค.ศ. 2003 ค่าจากนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งจำนวนงาน ลักษณะงาน และความต้องการหน่วยประมวลผลของงานแต่ละพาร์ทิชันจะเห็นได้ชัดเจนว่าค่าแตกต่างกันมาก แต่ค่าจากนโยบายแบบใช้เป้าหมายเป็นหลักแต่ละพาร์ทิชันกลับให้ค่าที่ใกล้เคียงกันมาก จากการวิเคราะห์ค่าของแต่ละนโยบายในทุกเดือนพบว่าเดือนกันยายนปี ค.ศ. 2003 และเดือนกุมภาพันธ์ปี ค.ศ. 2004 ค่าจำนวนงาน ลักษณะงาน และความต้องการหน่วยประมวลผลของงานแต่ละพาร์ทิชันจากนโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ส่วนเดือนอื่น ๆ นโยบายแบบใช้เป้าหมายเป็นหลักดีกว่าหรืออยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง

โดยสรุปความสามารถในการกระจายจำนวนงาน และงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชันให้เท่า ๆ กันของนโยบายแบบใช้เป้าหมายเป็นหลักดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองเพียงเล็กน้อยเท่านั้น ซึ่งจะดีกว่าค่าที่แย่ที่สุดของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเป็นส่วนใหญ่

เมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน นโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ซึ่งเหมือนกับระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าระยะเวลาการประมวลผลจริง (หัวข้อ 4.2.3.1) แต่การกระจายจำนวนงาน และงานลักษณะต่าง ๆ ของนโยบายแบบใช้เป้าหมายเป็นหลักดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเพียงเล็กน้อยเท่านั้น ทำให้ประสิทธิภาพที่ได้ไม่ชัดเจนมากเหมือนการใช้ค่าระยะเวลาการประมวลผลจริง ในหัวข้อต่อไปจะนำเสนอ และวิเคราะห์ประสิทธิภาพเมื่อระบบเป็นแบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน และใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

ตารางที่ 4-19 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Number of jobs												
	FCFS-backfill				LXF-backfill				Goal-oriented			
Partition / Month	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance
6/03	624	956	611	25,490.9	755	554	882	18,234.9	726	814	651	**4,437.6**
7/03	550	578	272	19,078.2	538	500	362	5,718.2	470	484	446	**246.2**
8/03	1,290	1,103	828	36,004.2	1,131	1,098	992	**3,516.2**	1,161	834	1,226	29,424.2
9/03	1,531	942	584	152,432.7	1,333	1,024	700	**66,794.0**	1,376	1,142	539	124,326.0
10/03	2,365	1,081	703	505,976.0	2,004	1,205	940	204,524.7	1,560	1,247	1,342	**17,168.7**
11/03	2,018	723	702	378,813.6	2,130	587	726	485,709.6	1,447	865	1,131	**56,592.9**
12/03	1,370	1,427	724	101,641.6	1,465	751	1,305	93,590.2	1,061	1,448	1,012	**38,029.6**
1/04	1,405	591	1,160	116,264.7	1,417	669	1,070	93,412.7	1,345	683	1,128	**75,928.7**
2/04	1,519	1,075	1,375	34,208.0	1,269	1,534	1,166	**24,028.7**	2,023	1,126	820	260,606.0
3/04	1,357	901	1,208	36,042.9	1,314	1,042	1,110	**13,358.2**	906	1,140	1,420	44,150.2

ตารางที่ 4-20 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Average job size (NT) in node-hour												
	FCFS-backfill				LXF-backfill				Goal-oriented			
Partition / Month	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance
6/03	33.5	21.5	33.7	32.5	27.5	37.7	23.1	37.4	28.5	25.2	32.0	**7.7**
7/03	29.3	27.4	58.0	196.0	30.0	32.0	42.9	32.1	35.1	32.1	35.1	**2.0**
8/03	13.8	15.2	20.0	7.0	15.9	15.2	16.6	**0.3**	15.6	20.7	12.8	10.7
9/03	10.1	17.4	28.5	57.2	12.0	15.7	23.5	**23.0**	11.2	13.6	32.5	90.7
10/03	8.1	16.9	27.9	65.6	9.6	15.1	20.8	20.9	11.9	15.2	14.5	**2.0**
11/03	7.4	23.7	21.8	53.0	6.7	29.8	21.5	91.3	10.7	19.7	13.2	**14.4**
12/03	14.6	12.6	27.1	41.2	13.9	24.3	14.6	22.5	18.0	13.2	19.2	**6.7**
1/04	14.6	31.7	15.6	61.4	13.9	29.1	17.1	**42.8**	14.1	30.7	15.4	56.8
2/04	12.8	17.3	13.9	3.7	14.9	12.1	16.8	**3.7**	9.2	16.6	24.3	38.0
3/04	15.1	23.9	16.8	14.5	16.1	20.5	17.9	**3.3**	23.6	18.2	14.2	14.8

ตารางที่ 4-21 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 3 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

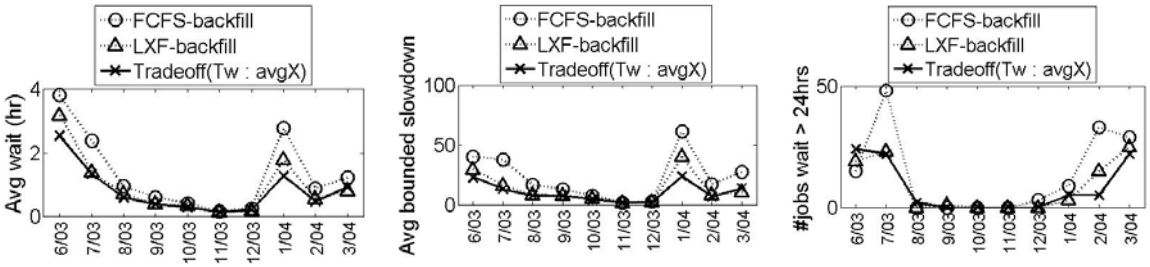
Average N												
	FCFS-backfill				LXF-backfill				Goal-oriented			
Partition / Month	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance	1st	2nd	3rd	Variance
6/03	9.2	7.5	11.1	2.16	9.2	10.5	7.8	1.22	8.6	8.1	10.4	**0.98**
7/03	12.1	11.5	19.5	13.24	11.2	14.2	15.1	2.78	13.2	13.6	13.2	**0.04**
8/03	4.6	4.8	5.3	**0.09**	4.9	4.4	5.3	0.14	4.6	6.1	4.3	0.62
9/03	4.9	7.3	9.6	3.68	5.9	5.8	8.8	**1.94**	5.9	4.8	11.8	9.45
10/03	2.9	5.5	7.5	3.55	3.1	5.2	5.9	1.42	4.0	4.8	4.4	**0.11**
11/03	3.0	6.1	8.1	4.40	2.6	8.3	7.8	6.64	3.6	6.1	5.0	**1.05**
12/03	3.7	4.0	6.4	1.46	3.3	7.3	4.0	3.04	5.0	3.8	4.6	**0.25**
1/04	10.7	12.1	6.6	**5.45**	5.2	17.0	10.4	23.32	9.1	14.8	6.6	11.78
2/04	3.7	3.4	4.9	**0.42**	3.9	3.3	5.1	0.56	3.1	5.4	4.6	0.91
3/04	4.3	4.8	4.2	**0.07**	3.9	4.2	5.2	0.31	6.1	4.5	3.2	1.41

4.4.1.3 ระบบ 4 พาร์ทิชัน

ในหัวข้อนี้จะแสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบ 4 พาร์ทิชันแบบจำนวนหน่วยประมวลผลเท่ากันคือ 32 โหนด การทดสอบในหัวข้อนี้ใช้เวิร์คโหลดไฟล์ซึ่งแสดงรายละเอียดไว้ในตารางที่ 3-3 จะเห็นได้ว่าค่าโหลดของระบบ 4 พาร์ทิช์ชันจะลดต่ำกว่าค่าโหลดในเวิร์คโหลดต้นฉบับเล็กน้อย (ตารางที่ 3-1) การทดสอบในหัวข้อนี้กระทำเช่นเดียวกับหัวข้อที่ 4.2.3.2 เพียงแต่เปลี่ยนค่าระยะเวลาการประมวลผลที่ใช้จากค่าระยะเวลาการประมวลผลจริงเป็นค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด ผลที่ได้จากการทดสอบนโยบายจะแสดงเป็น 3 หัวข้อย่อยคือ ประสิทธิภาพโดยรวมของระบบ (หัวข้อ 4.4.1.3.1) ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน (หัวข้อ 4.4.1.3.2) และวิเคราะห์ประสิทธิภาพ (หัวข้อ 4.4.1.3.3)

4.4.1.3.1 ประสิทธิภาพโดยรวมของระบบ

หัวข้อนี้จะแสดงการเปรียบเทียบประสิทธิภาพโดยรวมทั้งระบบของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน โดยใช้ค่าระยะเวลาการประมวลผลเป็นค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด ดังแสดงในภาพประกอบที่ 4-21



(ก) ค่าเฉลี่ยการรอ (ข) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ค) จำนวนงานที่รอมากกว่า 24 ชั่วโมง

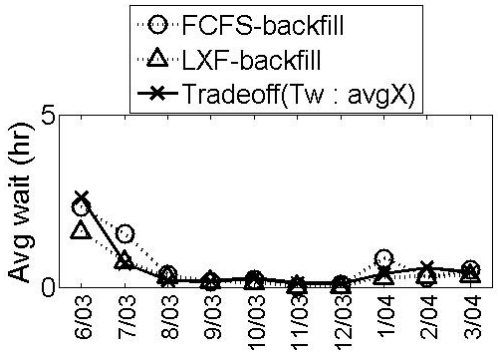
ภาพประกอบ 4-21 ประสิทธิภาพบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

ภาพประกอบที่ 4-21 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด โดยมีจำนวนหน่วยประมวลผลแต่ละพาร์ทิชันเท่ากับ 32 โหนด และแสดงเป็นกราฟจำนวน 10 เดือนตั้งแต่เดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งแสดงประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-21 (ก)) และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ภาพประกอบที่ 4-21 (ข)) ดีกว่าหรือเท่ากับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล เมื่อพิจารณาจำนวนงานที่รอมากกว่า 24 ชั่วโมง (ภาพประกอบที่ 4-21 (ค)) พบว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงดีกว่าหรือเท่ากับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ยกเว้นเดือนมิถุนายนปี ค.ศ. 2003 ซึ่งจำนวนงานของนโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเล็กน้อย

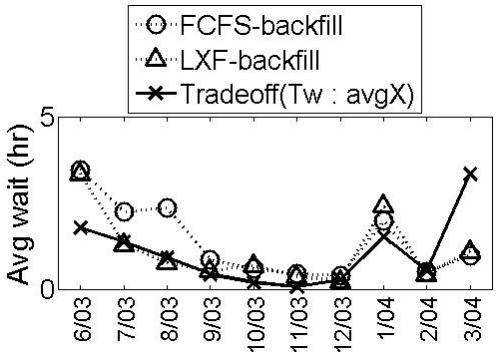
โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-21 นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แม้จะใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดแล้วก็ตาม เช่นเดียวกับบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันที่ใช้ค่าระยะเวลาการประมวลผลจริงดังภาพประกอบที่ 4-8

4.4.1.3.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน

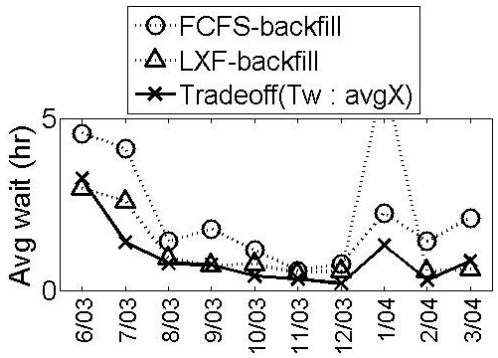
จากหัวข้อ 4.4.1.3.1 ได้เปรียบเทียบประสิทธิภาพโดยภาพรวมของทั้งระบบ โดยแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ดังนั้นหัวข้อนี้จะวิเคราะห์หาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ประสิทธิภาพของแต่ละนโยบายบนแต่ละพาร์ทิชัน ดังแสดงในภาพประกอบที่ 4-22, 4-23 และ 4-24 ทั้งนี้แนวนอนแสดงเดือนที่วัด และแนวตั้งแสดงประสิทธิภาพต่าง ๆ อันได้แก่ ค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แต่ละค่าจะแสดงเป็น 4 ภาพ โดยภาพประกอบที่ 4-22 แสดงผลของค่าเฉลี่ยการรอของแต่ละนโยบายแต่ละพาร์ทิชัน ในขณะที่ภาพประกอบที่ 4-23 แสดงผลของค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของแต่ละนโยบายแต่ละพาร์ทิชัน และภาพประกอบที่ 4-24 แสดงผลของจำนวนงานที่รอมากกว่า 24 ชั่วโมงของแต่ละนโยบายแต่ละพาร์ทิชัน



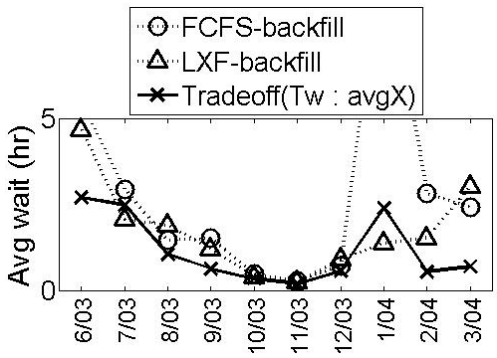
(ก) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 1



(ข) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 2



(ค) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 3

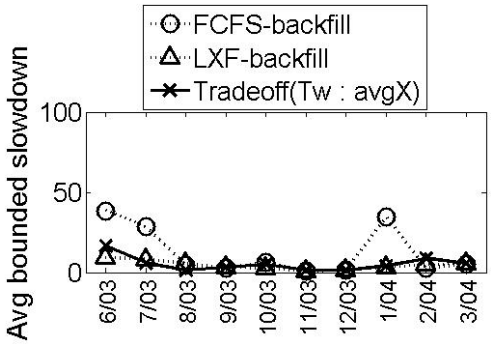


(ง) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 4

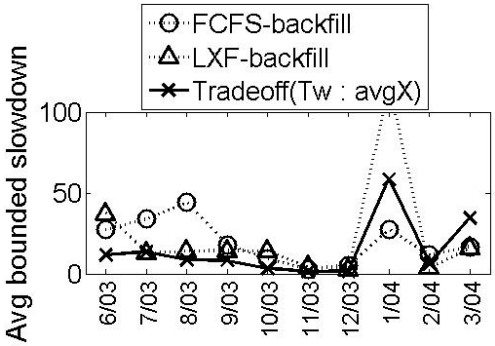
ภาพประกอบ 4-22 ค่าเฉลี่ยการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

จากภาพประกอบที่ 4-22, 4-23 และ 4-24 จะเห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีของทั้ง 3 ประสิทธิภาพที่แสดงบนทั้ง 4 พาร์ทิชัน ยกเว้นค่าเฉลี่ยการรอ และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์เดือนมีนาคมปี ค.ศ. 2004 บนพาร์ทิชันที่ 2 ที่นโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล แต่เมื่อดูภาพรวมทั้งระบบในภาพประกอบที่ 4-21 (ก), (ข) จะเห็นได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล แสดงว่าค่าที่แย่กว่าในบางพาร์ทิชันไม่ได้ส่งผลถึงทั้งระบบ ส่วนจำนวนงานที่รอมากกว่า 24 ชั่วโมงที่เดือนมีนาคมปี ค.ศ. 2004 บนพาร์ทิชันที่ 2 และเดือนมิถุนายนปี ค.ศ. 2003 บนพาร์ทิชันที่ 2 นโยบายแบบใช้เป้าหมายเป็นหลักจะมีจำนวนงานมากกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ซึ่งผล

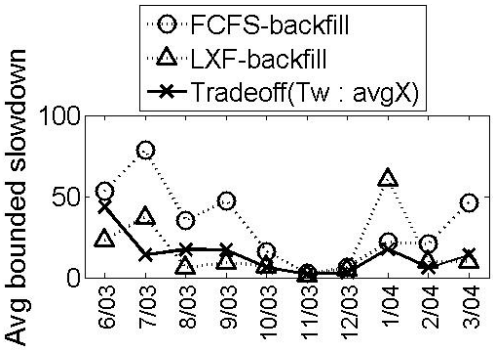
ของเดือนมีนาคมปี ค.ศ. 2004 บนพาร์ทิชันที่ 2 ไม่ได้ส่งผลถึงภาพรวมของทั้งระบบในภาพประกอบที่ 4-21 (ค)



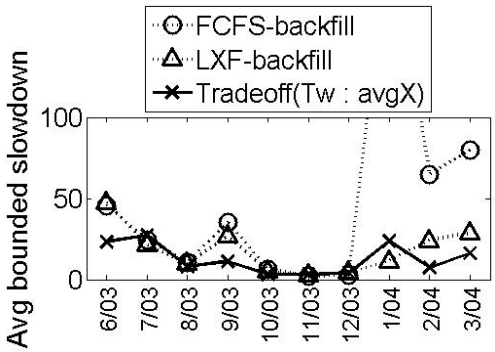
(ก) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 1



(ข) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 2



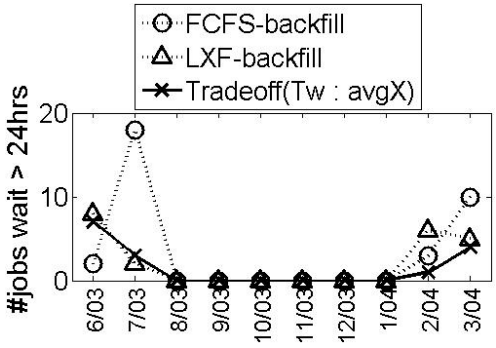
(ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 3



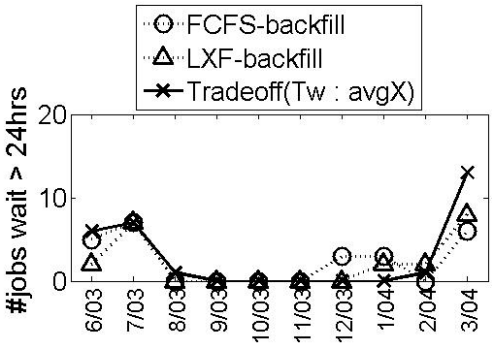
(ง) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 4

ภาพประกอบ 4-23 ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

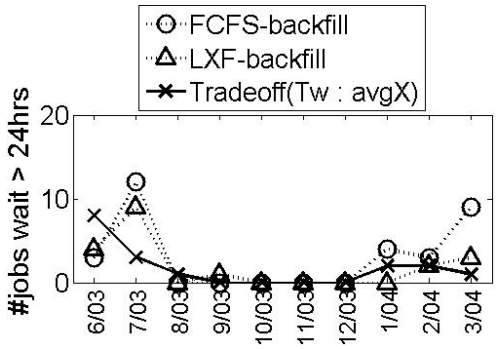
โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-21 ซึ่งเป็นแบบรวมทั้งระบบ และ 4-22, 4-23, 4-24 ซึ่งเป็นแบบแยกพาร์ทิชัน นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แม้จะทำงานบนระบบ 4 พาร์ทิชัน และใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดเช่นเดียวกับบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันที่ใช้ค่าระยะเวลาการประมวลผลจริงดังภาพประกอบที่ 4-8, 4-9, 4-10 และ 4-11



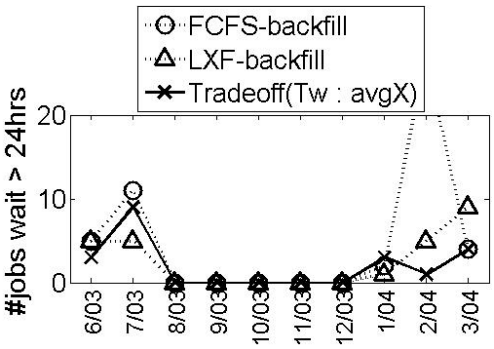
(ก) จำนวนงานที่รอมากกว่า 24 ชั่วโมง ของพาร์ทิชันที่ 1



(ข) จำนวนงานที่รอมากกว่า 24 ชั่วโมง ของพาร์ทิชันที่ 2



(ค) จำนวนงานที่รอมากกว่า 24 ชั่วโมง ของพาร์ทิชันที่ 3



(ง) จำนวนงานที่รอมากกว่า 24 ชั่วโมง ของพาร์ทิชันที่ 4

ภาพประกอบ 4-24 จำนวนงานที่รอมากกว่า 24 ชั่วโมงแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

4.4.1.3.3 วิเคราะห์ประสิทธิภาพ

จากหัวข้อที่ 4.4.1.3.1 และ 4.4.1.3.2 ก่อนหน้าได้แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ในหัวข้อนี้จะตรวจสอบหาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ถึงจำนวนงาน และปริมาณงานในแต่ละพาร์ทิชัน ดังแสดงในตารางที่ 4-22 จำนวนงานที่ถูกส่งไปประมวลผลในแต่ละพาร์ทิชันของแต่ละนโยบาย ตารางที่ 4-23 ลักษณะของงาน (จำนวนหน่วยประมวลผล × เวลาที่ใช้) ซึ่งแสดงค่าเฉลี่ยมีหน่วยเป็น จำนวนหน่วยประมวลผล ชั่วโมง และตารางที่ 4-24 แสดงค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานมีหน่วยเป็น จำนวนหน่วย

ประมวลผล เพื่อให้ง่ายต่อการวิเคราะห์ค่าในแต่ละพาร์ทิชันจึงใช้ค่าความแปรปรวนเพิ่มมาแต่ละนโยบาย ซึ่งค่าความแปรปรวนสามารถแสดงการกระจายของค่าในแต่ละพาร์ทิชันของแต่ละนโยบาย ถ้าค่าความแปรปรวนน้อยแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นใกล้เคียงกันมาก แต่ถ้าค่าความแปรปรวนมากแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นห่างกันมาก

จากตารางที่ 4-23, 4-24 และ 4-25 จะเห็นได้ว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งเอฟซีเอฟเอส และเอลเอกซ์เอฟมีจำนวนงานประมวลผลบนพาร์ทิชันที่ 1 มากกว่าพาร์ทิชันอื่น ๆ ในขณะที่ขนาดของงานในตารางที่ 4-24 และ 4-25 ที่พาร์ทิชันแรกของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเป็นงานที่มีขนาดเล็กกว่าพาร์ทิชันอื่น ๆ อย่างเห็นได้ชัดเช่น ในเดือนธันวาคมปี ค.ศ. 2003 จำนวนงานในพาร์ทิชันแรกมากกว่าพาร์ทิชันอื่น ๆ ถึง 3 เท่า แต่ขนาดของงาน และจำนวนหน่วยประมวลผลที่งานต้องการบนพาร์ทิชันแรกกลับน้อยกว่าพาร์ทิชันอื่น ๆ 3 ถึง 4 เท่า นั่นแสดงว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลจะส่งงานที่ต้องการหน่วยประมวลผลน้อยไปประมวลผลบนพาร์ทิชันแรกก่อน ซึ่งต่างจากนโยบายแบบใช้เป้าหมายเป็นหลักที่สามารถกระจายจำนวนงาน งานลักษณะต่าง ๆ และความต้องการหน่วยประมวลผลของงานไปแต่ละพาร์ทิชันได้ใกล้เคียงกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยดูได้จากค่าความแปรปรวนของนโยบายแบบใช้เป้าหมายเป็นหลักที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเกือบทุกเดือน ยกเว้นจำนวนงานในเดือนมิถุนายนปี ค.ศ. 2003 ลักษณะงาน และความต้องการหน่วยประมวลผลของงานในเดือนพฤศจิกายนปี ค.ศ. 2003 และเดือนมีนาคมปี ค.ศ. 2004 ถึงแม้ว่านโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล แต่นโยบายแบบใช้เป้าหมายเป็นหลักก็ดีกว่าค่าที่แย่ของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล กล่าวคือนโยบายแบบใช้เป้าหมายเป็นหลักจะอยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง

โดยสรุปความสามารถในการกระจายจำนวนงาน และงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชันให้เท่า ๆ กันของนโยบายแบบใช้เป้าหมายเป็นหลักดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองเพียงเล็กน้อยเท่านั้น ซึ่งจะดีกว่าค่าที่แย่ที่สุดของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล

ตารางที่ 4-22 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Number of jobs															
	FCFS-backfill					LXF-backfill					Goal-oriented				
Partition / Month	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance
6/03	609	669	468	445	8,842.7	474	573	633	511	**3,673.7**	598	652	510	431	7,112.2
7/03	554	408	222	216	19,830.0	597	372	221	210	24,433.5	509	350	304	237	**10,041.5**
8/03	1,763	328	772	358	336,547.7	1,498	849	500	374	190,242.7	1,365	558	838	460	**123,680.7**
9/03	1,877	577	348	255	426,468.7	1,887	578	370	222	436,181.2	1,494	750	421	392	**197,282.2**
10/03	2,660	670	330	489	892,242.7	2,516	714	433	486	740,046.7	1,545	1,210	759	635	**131,720.2**
11/03	2,339	379	268	457	732,918.2	2,403	405	222	413	798,681.2	1,196	1,344	215	688	**198,189.7**
12/03	2,264	617	285	355	653,568.7	2,207	737	357	220	622,626.7	1,653	723	828	317	**235,462.7**
1/04	1,640	488	510	518	241,521.0	1,538	783	437	398	209,455.5	1,066	745	715	630	**27,355.5**
2/04	1,723	1,181	432	633	253,140.7	1,676	966	811	516	181,967.2	967	850	1,267	885	**26,965.7**
3/04	1,626	604	674	562	193,880.8	1,726	516	880	344	283,694.8	1,197	336	833	1,100	**111,576.3**

ตารางที่ 4-23 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Average job size (NT) in node-hour															
	FCFS-backfill					LXF-backfill					Goal-oriented				
Partition / Month	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance
6/03	27.5	23.7	32.2	32.2	12.7	35.4	27.6	23.5	28.4	18.3	28.8	23.7	31.3	31.0	**9.3**
7/03	24.9	30.9	56.3	40.8	141.1	22.8	34.7	55.9	41.9	143.7	27.5	36.1	38.0	40.1	**22.9**
8/03	7.3	45.6	14.8	32.9	226.0	8.8	16.5	22.5	33.7	82.8	9.9	25.9	13.9	25.1	**48.3**
9/03	6.5	21.1	39.4	40.7	199.1	6.3	21.3	35.6	49.9	263.2	8.3	17.1	30.4	26.5	**73.6**
10/03	5.8	21.8	45.0	24.8	194.3	6.3	20.6	34.0	24.5	99.3	9.4	12.7	19.7	19.3	**19.2**
11/03	5.2	36.3	43.4	21.5	**215.1**	5.1	34.5	54.7	21.8	328.4	11.5	9.2	53.9	14.2	338.1
12/03	7.3	24.0	50.0	33.9	240.2	7.6	19.0	38.4	59.2	385.4	9.1	21.5	16.2	43.0	**160.1**
1/04	10.5	32.9	25.6	21.5	65.7	11.8	18.7	31.9	26.6	58.5	14.8	20.6	19.3	19.8	**5.1**
2/04	9.9	11.9	28.5	21.7	56.7	10.0	15.2	15.8	25.0	29.2	14.3	17.2	10.9	16.9	**6.4**
3/04	11.2	26.0	21.3	25.0	**34.3**	10.6	31.4	15.2	41.9	157.4	12.8	47.8	20.3	12.7	208.0

ตารางที่ 4-24 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Average N															
	FCFS-backfill					LXF-backfill					Goal-oriented				
Partition / Month	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance	1st	2nd	3rd	4th	Variance
6/03	7.6	7.9	10.1	11.4	2.47	9.9	9.4	7.0	10.0	1.49	10.3	7.3	8.8	9.8	**1.31**
7/03	10.0	12.4	20.4	16.3	15.60	9.3	14.0	18.6	17.9	13.71	11.0	15.2	12.7	16.2	**4.19**
8/03	2.7	11.2	5.1	9.0	10.94	2.9	5.3	7.1	8.7	4.65	3.3	6.9	4.4	8.1	**3.66**
9/03	3.4	8.8	13.6	14.7	20.00	3.4	8.6	13.3	16.4	24.16	4.1	6.0	11.7	11.1	**10.58**
10/03	2.5	6.2	10.8	7.7	8.92	2.6	5.7	8.4	8.2	5.51	3.4	3.8	5.1	6.7	**1.66**
11/03	2.2	9.9	13.9	7.8	**17.82**	2.1	9.6	16.5	8.3	26.16	4.1	3.3	15.6	5.2	24.82
12/03	2.3	6.4	9.9	9.7	9.53	2.5	4.6	10.3	12.9	17.60	3.1	5.3	4.9	8.3	**3.49**
1/04	3.7	10.0	12.0	24.7	58.19	3.2	12.8	21.7	13.7	43.04	5.3	14.4	6.8	13.7	**16.34**
2/04	2.3	3.0	9.2	7.2	8.26	2.7	3.4	5.3	7.5	3.47	5.7	3.7	3.3	3.7	**0.88**
3/04	2.8	6.4	5.9	5.2	**1.91**	2.6	5.9	4.1	12.3	13.67	3.5	11.1	4.7	3.1	10.43

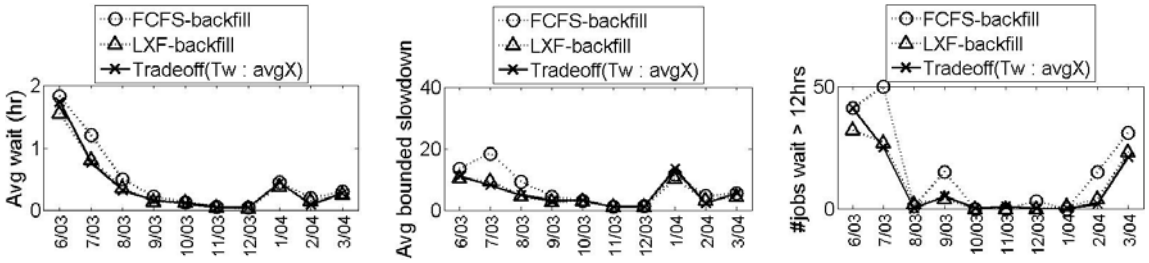
เมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดบนระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน นโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ซึ่งเหมือนกับระบบ 4 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าระยะเวลาการประมวลผลจริง (หัวข้อ 4.2.3.2) แต่การกระจายจำนวนงาน และงานลักษณะต่าง ๆ ของนโยบายแบบใช้เป้าหมายเป็นหลักดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเพียงเล็กน้อยเท่านั้น ทำให้ประสิทธิภาพที่ได้ไม่ชัดเจนมากเหมือนการใช้ค่าระยะเวลาการประมวลผลจริง ในหัวข้อต่อไปจะนำเสนอ และวิเคราะห์ประสิทธิภาพเมื่อระบบเป็นแบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน และใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

4.4.1.4 ระบบ 5 พาร์ทิชัน

ในหัวข้อนี้จะแสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบ 5 พาร์ทิชันแบบจำนวนหน่วยประมวลผลเท่ากันคือ 32 โหนด การทดสอบในหัวข้อนี้ใช้เวิร์คโหลดไฟล์ซึ่งแสดงรายละเอียดไว้ในตารางที่ 3-3 จะเห็นได้ว่าค่าโหลดของระบบ 5 พาร์ทิช์ชันจะต่ำกว่าค่าโหลดในเวิร์คโหลดต้นฉบับมาก (ตารางที่ 3-1) ทำให้ในหลาย ๆ เดือนไม่มีจำนวนงานที่รอมากกว่า 24 ชั่วโมง ดังนั้นจึงพิจารณาจำนวนงานที่มากกว่า 12 ชั่วโมงแทน การทดสอบในหัวข้อนี้กระทำเช่นเดียวกับหัวข้อที่ 4.2.3.3 เพียงแต่เปลี่ยนค่าระยะเวลาการประมวลผลที่ใช้จากค่าค่าระยะเวลาการประมวลผลจริงเป็นค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด ผลที่ได้จากการทดสอบนโยบายจะแสดงเป็น 3 หัวข้อย่อยคือ ประสิทธิภาพโดยรวมของระบบ (หัวข้อ 4.4.1.4.1) ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน (หัวข้อ 4.4.1.4.2) และวิเคราะห์ประสิทธิภาพ (หัวข้อ 4.4.1.4.3)

4.4.1.4.1 ประสิทธิภาพโดยรวมของระบบ

หัวข้อนี้จะแสดงการเปรียบเทียบประสิทธิภาพโดยรวมทั้งระบบของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน โดยใช้ค่าระยะเวลาการประมวลผลเป็นค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด ดังแสดงในภาพประกอบที่ 4-25



(ก) ค่าเฉลี่ยการรอ (ข) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ค) จำนวนงานที่รอมากกว่า 12 ชั่วโมง

ภาพประกอบ 4-25 ประสิทธิภาพบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

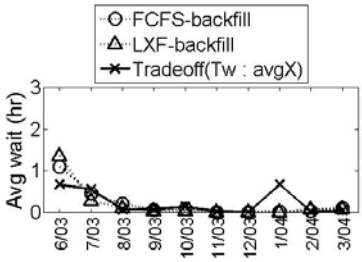
ภาพประกอบที่ 4-25 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน เมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด โดยมีจำนวนหน่วยประมวลผลแต่ละพาร์ทิชันเท่ากับ 32 โหนด และแสดงเป็นกราฟจำนวน 10 เดือนตั้งแต่เดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งแสดงประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่านโยบายแบบเอลเอกซ์เอฟแบ็กฟิลให้ค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-25 (ก)) และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ภาพประกอบที่ 4-25 (ข)) ดีกว่าหรือเท่ากับนโยบายแบบเอฟซีเอฟเอสแบ็กฟิล ซึ่งจะเห็นได้ว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองให้ค่าเฉลี่ยการรอ และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ที่ใกล้เคียงกันมากและเท่ากันในบางเดือน ซึ่งนโยบายแบบใช้เป้าหมายเป็นหลักสามารถให้ค่าเฉลี่ยการรอ และค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ดีกว่าหรือใกล้เคียงค่าที่ดีที่สุด ยกเว้นค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของเดือนมกราคมปี ค.ศ. 2004 ที่ค่าจากนโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล อันเนื่องมาจากกลุ่มงานสั้นบางกลุ่มเพราะเมื่อวิเคราะห์ค่าเฉลี่ยการรอจะเห็นได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักมีค่าใกล้เคียงกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล เมื่อพิจารณาจำนวนงานที่รอมากกว่า 12 ชั่วโมง (ภาพประกอบที่ 4-25 (ค)) พบว่านโยบายแบบใช้เป้าหมายเป็นหลักมีจำนวนงานที่รอมากกว่า 12 ชั่วโมงน้อยกว่าหรือเท่ากับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล

โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-25 นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 12 ชั่วโมงแม้จะใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดแล้วก็ตาม เช่นเดียวกับบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันที่ใช้ค่าระยะเวลาการประมวลผลจริงดังภาพประกอบที่ 4-12

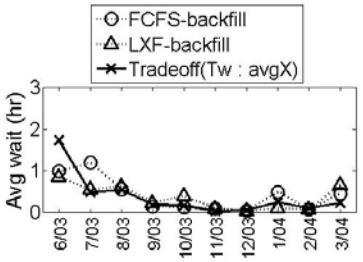
4.4.1.4.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน

จากหัวข้อ 4.4.1.4.1 ได้เปรียบเทียบประสิทธิภาพโดยภาพรวมของทั้งระบบ โดยแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ดังนั้นหัวข้อนี้จะวิเคราะห์หาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ประสิทธิภาพของแต่ละนโยบายบนแต่ละพาร์ทิชัน ดังแสดงในภาพประกอบที่ 4-26, 4-27 และ 4-28 ทั้งนี้แนวนอนแสดงเดือนที่วัด และแนวตั้งแสดงประสิทธิภาพต่าง ๆ อันได้แก่ ค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 12 ชั่วโมง แต่ละค่าจะแสดงเป็น 5 ภาพ โดยภาพประกอบที่ 4-26 แสดงผลของค่าเฉลี่ยการรอของแต่ละนโยบายแต่ละพาร์ทิชัน ในขณะที่ภาพประกอบที่ 4-27 แสดงผลของค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของแต่ละนโยบายแต่ละพาร์ทิชัน และภาพประกอบที่ 4-28 แสดงผลของจำนวนงานที่รอมากกว่า 12 ชั่วโมงของแต่ละนโยบายแต่ละพาร์ทิชัน

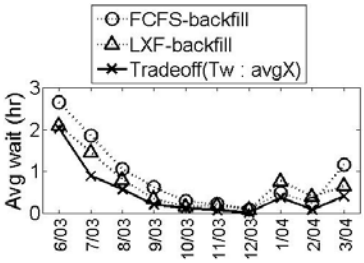
จากภาพประกอบที่ 4-26 จะเห็นว่าค่าเฉลี่ยการรอของนโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีของทั้ง 3 ประสิทธิภาพที่แสดงบนทั้ง 5 พาร์ทิชัน ยกเว้นพาร์ทิชันที่ 1 เดือนมกราคมปี ค.ศ. 2004 พาร์ทิชันที่ 2 เดือนมิถุนายนปี ค.ศ. 2003 และพาร์ทิชันที่ 4 เดือนมีนาคมปี ค.ศ. 2004 ที่นโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล แต่จะเห็นได้ว่าในภาพรวมทั้งระบบในภาพประกอบที่ 4-25 นโยบายแบบใช้เป้าหมายเป็นหลักจะมีค่าอยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง



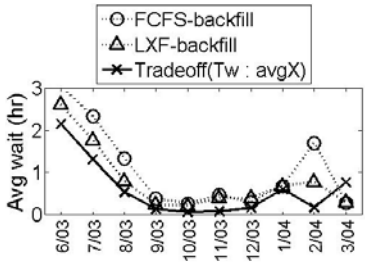
(ก) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 1



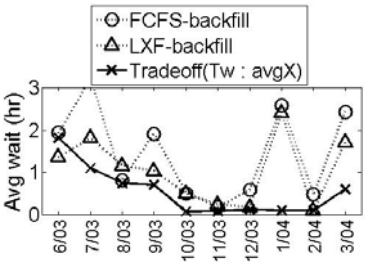
(ข) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 2



(ค) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 3



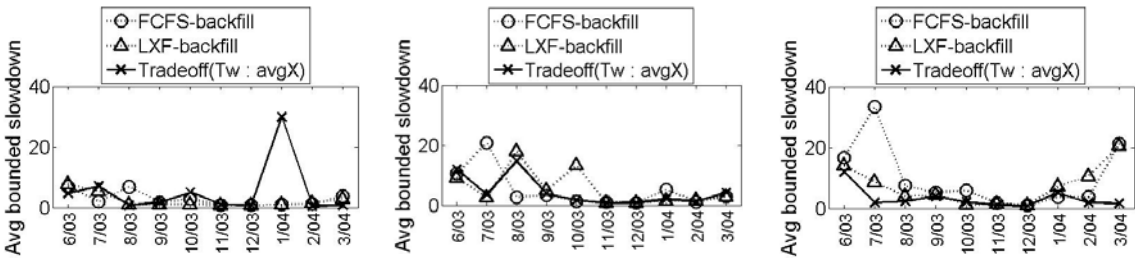
(ง) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 4



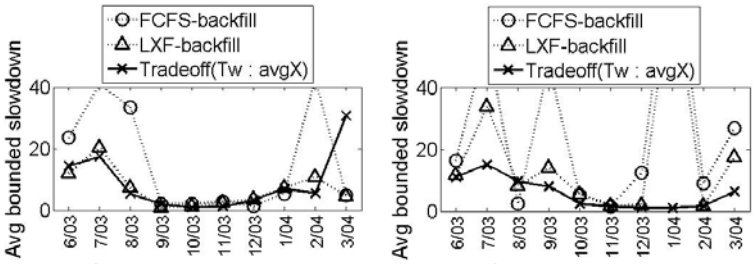
(จ) ค่าเฉลี่ยของการรอของพาร์ทิชันที่ 5

ภาพประกอบ 4-26 ค่าเฉลี่ยการรอแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

จากภาพประกอบที่ 4-27 จะเห็นว่าค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของนโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีของทั้ง 3 ประสิทธิภาพที่แสดงบนทั้ง 5 พาร์ทิชัน ยกเว้นพาร์ทิชันที่ 1 เดือนมกราคมปี ค.ศ. 2004 และพาร์ทิชันที่ 4 เดือนมีนาคมปี ค.ศ. 2004 ที่นโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล แต่จะเห็นได้ว่าในภาพรวมทั้งระบบในภาพประกอบที่ 4-25 นโยบายแบบใช้เป้าหมายเป็นหลักที่เดือนมีนาคมปี ค.ศ. 2004 จะมีค่าอยู่ระหว่างนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง ส่วนเดือนมกราคมปี ค.ศ. 2004 นโยบายแบบใช้เป้าหมายเป็นหลักจะมีค่ามากกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลอยู่เล็กน้อย ดังที่ได้กล่าวไว้แล้วในหัวข้อ 4.4.1.4.1



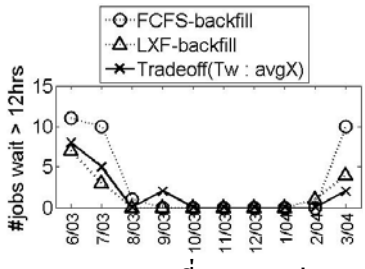
(ก) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 1 (ข) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 2 (ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 3



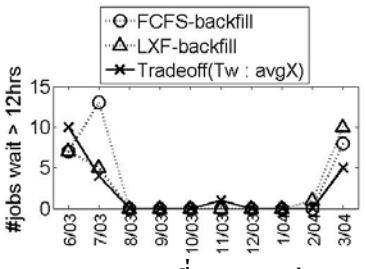
(ง) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 4 (จ) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 5

ภาพประกอบ 4-27 ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์แยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

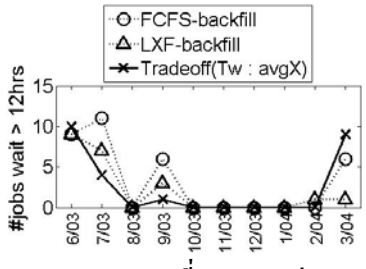
จากภาพประกอบที่ 4-28 จะเห็นว่าจำนวนงานที่รอมากกว่า 12 ชั่วโมงของนโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีของทั้ง 3 ประสิทธิภาพที่แสดงบนทั้ง 5 พาร์ทิชัน ยกเว้นพาร์ทิชันที่ 2 เดือนมิถุนายนปี ค.ศ. 2003 และพาร์ทิชันที่ 3 เดือนมีนาคมปี ค.ศ. 2004 ที่นโยบายแบบใช้เป้าหมายเป็นหลักแย่กว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล แต่จะเห็นได้ว่าในภาพรวมทั้งระบบในภาพประกอบที่ 4-25 นโยบายแบบใช้เป้าหมายเป็นหลักจะมีค่าใกล้เคียงกับค่าที่ดีที่สุดของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล



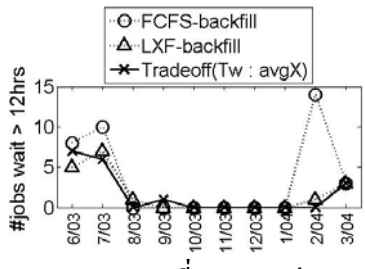
(ก) จำนวนงานที่รอมากกว่า 12 ชั่วโมงของพาร์ทิชันที่ 1



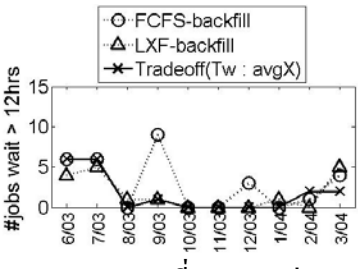
(ข) จำนวนงานที่รอมากกว่า 12 ชั่วโมงของพาร์ทิชันที่ 2



(ค) จำนวนงานที่รอมากกว่า 12 ชั่วโมงของพาร์ทิชันที่ 3



(ง) จำนวนงานที่รอมากกว่า 12 ชั่วโมงของพาร์ทิชันที่ 4



(จ) จำนวนงานที่รอมากกว่า 12 ชั่วโมงของพาร์ทิชันที่ 5

ภาพประกอบ 4-28 จำนวนงานที่รอมากกว่า 12 ชั่วโมงแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-25 ซึ่งเป็นแบบรวมทั้งระบบ และ 4-26, 4-27, 4-28 ซึ่งเป็นแบบแยกพาร์ทิชัน นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 12 ชั่วโมง แม้จะทำงานบนระบบ 5 พาร์ทิชัน และใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดเช่นเดียวกับบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันที่ใช้ค่าระยะเวลาการประมวลผลจริงดังภาพประกอบที่ 4-12, 4-13, 4-14 และ 4-15

4.4.1.4.3 วิเคราะห์ประสิทธิภาพ

จากหัวข้อที่ 4.4.1.4.1 และ 4.4.1.4.2 ก่อนหน้าได้แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล

ในหัวข้อนี้จะตรวจสอบหาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ถึงจำนวนงาน และปริมาณงานในแต่ละพาร์ทิชัน ดังแสดงในตารางที่ 4-25 จำนวนงานที่ถูกส่งไปประมวลผลในแต่ละพาร์ทิชันของแต่ละนโยบาย ตารางที่ 4-26 ลักษณะของงาน (จำนวนหน่วยประมวลผล × เวลาที่ใช้) ซึ่งแสดงค่าเฉลี่ยมีหน่วยเป็น จำนวนหน่วยประมวลผลชั่วโมง และตารางที่ 4-27 แสดงค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานมีหน่วยเป็น จำนวนหน่วยประมวลผล เพื่อให้ง่ายต่อการวิเคราะห์ค่าในแต่ละพาร์ทิชันจึงใช้ค่าความแปรปรวนเพิ่มมาแต่ละนโยบาย ซึ่งค่าความแปรปรวนสามารถแสดงการกระจายของค่าในแต่ละพาร์ทิชันของแต่ละนโยบาย ถ้าค่าความแปรปรวนน้อยแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นใกล้เคียงกันมาก แต่ถ้าค่าความแปรปรวนมากแสดงว่าค่าในแต่ละพาร์ทิชันของนโยบายนั้นห่างกันมาก

จากตารางที่ 4-25, 4-26 และ 4-27 จะเห็นได้ว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งเอฟซีเอฟเอส และเอลเอกซ์เอฟมีจำนวนงานประมวลผลบนพาร์ทิชันที่ 1 มากกว่าพาร์ทิชันอื่น ๆ ทุกเดือนอย่างเห็นได้ชัด ในขณะที่ขนาดของงานในตารางที่ 4-26 และ 4-27 ที่พาร์ทิชันแรกของนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเป็นงานที่มีขนาดเล็กกว่าพาร์ทิชันอื่น ๆ ทุกเดือนอย่างชัดเจนเช่น ในเดือนตุลาคมปี ค.ศ. 2003 จำนวนงานในพาร์ทิชันแรกมากกว่าพาร์ทิชันอื่น ๆ ถึง 6 เท่า แต่ขนาดของงาน และจำนวนหน่วยประมวลผลที่งานต้องการบนพาร์ทิชันแรกกลับน้อยกว่าพาร์ทิชันอื่น ๆ 5 ถึง 8 เท่า นั่นแสดงว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลจะส่งงานที่ต้องการหน่วยประมวลผลน้อยไปประมวลผลบนพาร์ทิชันแรกก่อน ซึ่งต่างจากนโยบายแบบใช้เป้าหมายเป็นหลักที่สามารถกระจายจำนวนงาน งานลักษณะต่าง ๆ และความต้องการหน่วยประมวลผลของงานไปแต่ละพาร์ทิชันได้ใกล้เคียงกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยดูได้จากค่าความแปรปรวนของนโยบายแบบใช้เป้าหมายเป็นหลักที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทุกเดือน

โดยสรุปความสามารถในการกระจายจำนวนงาน และงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชันให้เท่า ๆ กันของนโยบายแบบใช้เป้าหมายเป็นหลักดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองอย่างชัดเจน ถึงแม้ว่านโยบายแบบใช้เป้าหมายเป็นหลักจะเลือกส่งงานขนาดเล็กไปพาร์ทิชันใด พาร์ทิชันหนึ่งมากกว่าพาร์ทิชันอื่น ๆ ซึ่งดูได้จากจำนวนงาน (ตารางที่ 4-25) และขนาดของงาน (ตารางที่ 4-25) แต่นโยบายแบบใช้เป้าหมายเป็นหลักจะสามารถกระจายจำนวนงาน และงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชันได้ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล

ตารางที่ 4-25 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Number of jobs																		
	FCFS-backfill						LXF-backfill						Goal-oriented					
Partition / Month	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance
6/03	767	299	458	331	336	29,963.0	768	425	440	246	312	32,362.6	371	487	577	427	329	**7,642.6**
7/03	612	362	208	136	82	36,414.4	609	323	171	195	102	32,176.0	420	407	172	220	181	**12,158.8**
8/03	1,974	253	337	352	305	443,243.0	2,050	453	359	217	142	505,775.0	1,631	620	405	301	264	**258,783.0**
9/03	2,100	440	252	124	141	566,662.2	2,079	455	265	119	139	552,784.6	1,555	536	520	262	184	**241,834.2**
10/03	3,000	541	253	202	153	1,195,612.6	2,961	588	254	197	149	1,159,190.2	1,638	500	653	538	820	**175,691.4**
11/03	2,604	304	224	140	171	920,279.8	2,620	323	223	142	135	937,199.4	1,269	882	396	589	307	**123,084.2**
12/03	2,522	450	244	149	156	837,913.8	2,516	443	243	171	148	831,442.2	1,354	618	646	319	584	**119,177.0**
1/04	1,794	562	347	124	329	357,247.8	1,782	599	313	120	342	354,318.2	1,165	388	548	443	612	**77,359.8**
2/04	2,047	932	473	238	279	453,291.0	2,083	959	424	243	260	482,880.6	706	679	1,112	724	748	**25,821.8**
3/04	2,379	365	202	417	103	723,107.4	2,400	398	192	387	89	742,065.4	1,438	794	611	390	233	**175,071.8**

ตารางที่ 4-26 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Average job size (NT) in node-hour																		
	FCFS-backfill						LXF-backfill						Goal-oriented					
Partition / Month	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance
6/03	19.2	43.7	26.8	33.8	32.1	65.4	19.6	30.5	27.1	46.3	34.4	77.9	38.5	26.9	21.5	26.2	33.5	**35.7**
7/03	19.9	30.8	47.3	56.5	83.5	483.5	19.8	36.6	58.7	35.6	67.1	291.9	29.9	27.2	60.9	33.9	33.9	**147.4**
8/03	5.8	52.3	29.6	27.0	23.1	222.0	5.4	28.2	28.4	44.6	51.7	256.2	7.0	21.0	25.0	32.4	25.6	**71.2**
9/03	4.7	25.8	45.7	65.3	53.5	459.5	4.9	25.6	42.0	68.3	53.2	482.3	7.2	21.2	21.1	27.9	41.1	**121.1**
10/03	5.0	24.2	45.8	49.5	48.1	302.8	5.1	21.7	45.0	53.1	49.9	344.1	7.5	29.4	16.7	17.9	11.6	**54.7**
11/03	4.4	42.4	45.5	57.1	28.9	325.0	4.4	39.2	44.9	55.0	41.0	294.0	9.5	13.1	23.7	15.1	17.9	**22.9**
12/03	6.7	29.3	44.5	67.7	42.4	399.8	6.7	29.4	45.0	59.9	44.0	322.7	11.0	19.3	16.3	32.4	17.0	**50.9**
1/04	9.3	24.6	30.8	79.6	19.3	599.2	9.3	21.4	38.3	75.9	20.0	546.7	12.3	35.8	21.8	17.6	15.5	**67.3**
2/04	8.2	13.4	24.5	40.4	24.0	122.6	8.2	13.0	26.9	40.1	25.3	126.3	18.0	18.3	10.7	14.3	13.0	**8.6**
3/04	6.9	37.0	64.6	24.8	86.8	810.0	6.9	35.4	69.1	25.7	94.7	991.6	10.5	18.2	23.5	21.3	43.3	**118.8**

ตารางที่ 4-27 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Average N																		
	FCFS-backfill						LXF-backfill						Goal-oriented					
Partition / Month	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance	1st	2nd	3rd	4th	5th	Variance
6/03	6.2	12.7	9.2	10.1	10.5	4.46	6.4	9.5	9.5	13.1	10.4	4.60	10.2	8.7	7.9	8.4	10.6	**1.11**
7/03	8.6	13.4	16.9	21.8	25.2	34.73	8.8	14.2	18.3	17.2	21.7	18.86	12.6	12.7	17.6	13.7	11.9	**4.13**
8/03	2.3	10.4	8.9	8.1	8.9	**7.90**	2.3	6.5	8.1	11.1	19.4	32.67	2.6	4.8	7.6	8.0	11.2	**8.61**
9/03	3.1	8.9	15.1	24.3	19.5	56.44	2.9	9.7	14.9	24.1	19.4	54.70	3.6	8.6	8.0	12.0	12.9	**10.91**
10/03	2.1	6.7	12.2	11.9	16.9	25.87	2.1	6.4	12.2	11.5	18.0	29.29	2.9	6.6	5.6	5.6	4.2	**1.67**
11/03	2.0	10.0	15.9	20.1	8.5	38.92	2.0	9.7	16.0	19.1	10.4	34.54	3.6	3.9	8.1	5.7	5.0	**2.59**
12/03	2.2	5.7	11.5	15.3	14.6	26.15	2.2	5.6	11.6	14.0	14.9	24.44	3.1	6.1	4.5	8.0	3.7	**3.15**
1/04	2.2	10.1	20.5	27.5	29.6	108.62	2.2	10.2	19.3	27.7	30.5	112.50	8.3	13.9	8.9	14.9	5.5	**12.62**
2/04	2.0	3.6	7.8	12.5	7.0	13.31	2.1	3.5	9.5	10.2	6.6	10.20	4.6	4.7	3.1	4.5	3.9	**0.36**
3/04	2.0	8.9	13.5	5.1	22.9	53.35	2.0	10.0	13.1	5.4	21.2	43.93	2.5	4.8	5.6	5.2	10.1	**6.13**

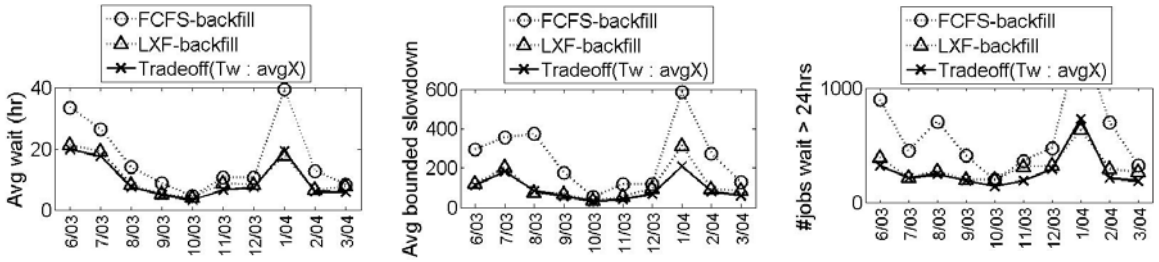
เมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดบนระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากัน นโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ซึ่งเหมือนกับระบบ 5 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันเมื่อใช้ค่าระยะเวลาการประมวลผลจริง (หัวข้อ 4.2.3.3) แต่ประสิทธิภาพที่ได้ไม่ชัดเจนมากเหมือนการใช้ค่าระยะเวลาการประมวลผลจริง และมีกลุ่มงานสั้นบางกลุ่มที่ต้องรอนานทำให้ประสิทธิภาพค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ในเดือนมกราคมปี ค.ศ. 2004 ของนโยบายแบบใช้เป้าหมายเป็นหลักสูงกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเล็กน้อย ในหัวข้อต่อไปจะนำเสนอ และวิเคราะห์ประสิทธิภาพเมื่อระบบเป็นแบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน และใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

4.4.2 ประสิทธิภาพบนระบบหลายพาร์ทิชันแบบจำนวนหน่วยประมวลผลไม่เท่ากัน

หัวข้อนี้จะทดสอบ และวิเคราะห์ประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบ 2 พาร์ทิชันแบบจำนวนหน่วยประมวลผลไม่เท่ากัน โดยให้พาร์ทิชันแรกมีหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2 มีหน่วยประมวลผล 64 โหนด การทดสอบในหัวข้อนี้ใช้เวิร์คโหลดไฟล์ซึ่งแสดงรายละเอียดไว้ในตารางที่ 3-2 และทดสอบเช่นเดียวกับหัวข้อที่ 4.2.2 เพียงแต่เปลี่ยนค่าระยะเวลาการประมวลผลที่ใช้จากค่าระยะเวลาการประมวลผลจริงเป็นค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด ผลที่ได้จากการทดสอบนโยบายจะแสดงเป็น 3 หัวข้อย่อยคือ ประสิทธิภาพโดยรวมของระบบ (หัวข้อ 4.4.2.1) ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน (หัวข้อ 4.4.2.2) และวิเคราะห์ประสิทธิภาพ (หัวข้อ 4.4.2.3)

4.4.2.1 ประสิทธิภาพโดยรวมของระบบ

หัวข้อนี้จะแสดงการเปรียบเทียบประสิทธิภาพโดยรวมทั้งระบบของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบคอมพิวเตอร์แบบขนาน 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน โดยใช้ค่าระยะเวลาการประมวลผลเป็นค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด ดังแสดงในภาพประกอบที่ 4-29



(ก) ค่าเฉลี่ยการรอ (ข) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ค) จำนวนงานที่รอมากกว่า 24 ชั่วโมง

ภาพประกอบ 4-29 ประสิทธิภาพบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

ภาพประกอบที่ 4-29 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน เมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด โดยพาร์ทิชันแรกมีจำนวนหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผล 64 โหนด ซึ่งแสดงเป็นกราฟจำนวน 10 เดือนตั้งแต่เดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งแสดงประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่านโยบายแบบเอลเอกซ์เอฟแบ็กฟิลให้ค่าเฉลี่ยการรอ (ภาพประกอบที่ 4-29 (ก)) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ (ภาพประกอบที่ 4-29 (ข)) และจำนวนงานที่รอมากกว่า 24 ชั่วโมง (ภาพประกอบที่ 4-29 (ค)) ดีกว่านโยบายแบบเอฟซีเอฟเอสแบ็กฟิลทุกเดือน แต่จะเห็นได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ค่าดีกว่าหรือเท่ากับค่าจากนโยบายแบบเอลเอกซ์เอฟแบ็กฟิลทุกประสิทธิภาพที่แสดงไว้

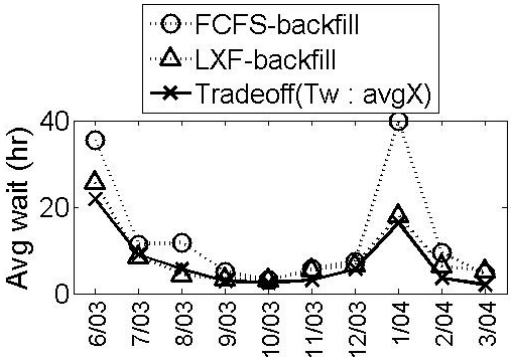
โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-29 นโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แม้จะใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดแล้วก็ตาม เช่นเดียวกับบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันที่ใช้ค่าระยะเวลาการประมวลผลจริงดังภาพประกอบที่ 4-4

4.4.2.2 ประสิทธิภาพเมื่อแยกแต่ละพาร์ทิชัน

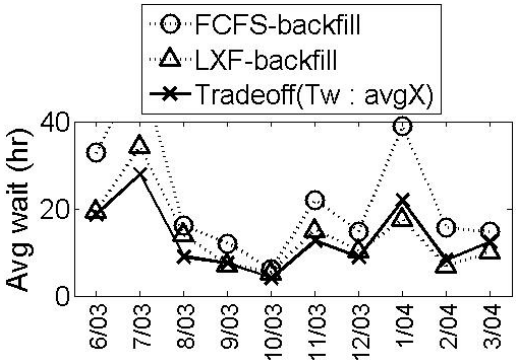
จากหัวข้อ 4.4.2.1 ได้เปรียบเทียบประสิทธิภาพโดยภาพรวมของทั้งระบบ โดยแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ดังนั้นหัวข้อนี้จะวิเคราะห์หาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ประสิทธิภาพของแต่ละนโยบายบนแต่ละพาร์ทิชัน ดังแสดงในภาพประกอบที่ 4-30 ทั้งนี้แนวนอนแสดงเดือนที่วัด และแนวตั้งแสดงประสิทธิภาพต่าง ๆ อันได้แก่ ค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แต่ละค่าจะแสดงเป็น 2 ภาพ โดยภาพประกอบที่ 4-30 ก, ค, จ แสดงผลของค่าดังกล่าวของแต่ละนโยบายบนพาร์ทิชันแรก ในขณะที่ภาพประกอบที่ 4-30 ข, ง, ฉ แสดงค่าที่วัดได้ของแต่ละนโยบายบนพาร์ทิชันที่ 2

ภาพประกอบที่ 4-30 แสดงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักและนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน เมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด และแยกแสดงประสิทธิภาพแต่ละพาร์ทิชัน โดยพาร์ทิชันแรกมีจำนวนหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผล 64 โหนด ซึ่งแสดงเป็นกราฟจำนวน 10 เดือนตั้งแต่เดือนมิถุนายนปี ค.ศ. 2003 ถึงเดือนมีนาคมปี ค.ศ. 2004 แกนแนวนอนแสดงชื่อเดือนแต่ละเดือน ส่วนแกนแนวตั้งแสดงประสิทธิภาพการทำงาน ข้อมูลจากภาพแสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพดีที่สุดหรือใกล้เคียงค่าที่ดีของทั้ง 3 ประสิทธิภาพที่แสดงบนทั้ง 2 พาร์ทิชัน

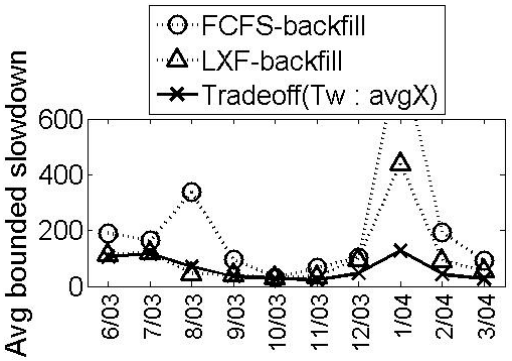
โดยสรุปจากผลการทดสอบแสดงในภาพประกอบที่ 4-29 ซึ่งเป็นแบบรวมทั้งระบบ และ 4-30 ซึ่งเป็นแบบแยกพาร์ทิชัน นโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล โดยให้ประสิทธิภาพที่ดีทั้งค่าเฉลี่ยการรอ ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ และจำนวนงานที่รอมากกว่า 24 ชั่วโมง แม้จะทำงานบนระบบ 2 พาร์ทิชัน และใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด เช่นเดียวกับบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน และใช้ค่าระยะเวลาการประมวลผลจริงดังภาพประกอบที่ 4-4 และ 4-5



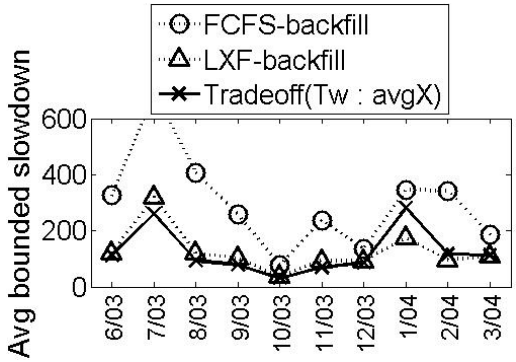
(ก) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 1



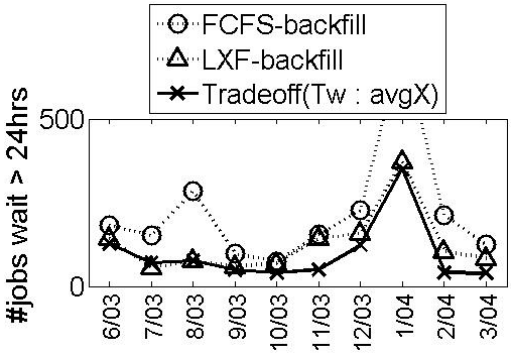
(ข) ค่าเฉลี่ยการรอของพาร์ทิชันที่ 2



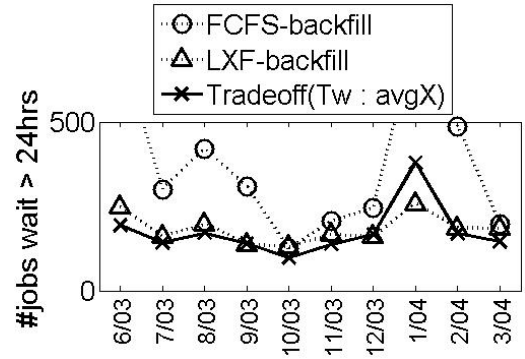
(ค) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 1



(ง) ค่าเฉลี่ยเบาน์เด็ดสโลว์ดาวน์ของพาร์ทิชันที่ 2



(จ) จำนวนงานที่รอมากกว่า 24 ชั่วโมงของพาร์ทิชันที่ 1



(ฉ) จำนวนงานที่รอมากกว่า 24 ชั่วโมงของพาร์ทิชันที่ 2

ภาพประกอบ 4-30 ประสิทธิภาพแยกตามพาร์ทิชันของแต่ละนโยบายบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

4.4.2.3 วิเคราะห์ประสิทธิภาพ

จากหัวข้อที่ 4.4.2.1 และ 4.4.2.2 ก่อนหน้าได้แสดงให้เห็นว่านโยบายแบบใช้เป้าหมายเป็นหลักให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ในหัวข้อนี้จะตรวจสอบหาสาเหตุของการได้ประสิทธิภาพที่ดีของนโยบายแบบใช้เป้าหมายเป็นหลัก โดยวิเคราะห์ถึงจำนวนงาน และปริมาณงานในแต่ละพาร์ทิชัน ดังแสดงในตารางที่ 4-28 จำนวนงานที่ถูกส่งไปประมวลผลในแต่ละพาร์ทิชันของแต่ละนโยบาย ตารางที่ 4-29 ลักษณะของงาน (จำนวนหน่วยประมวลผล × เวลาที่ใช้) ซึ่งแสดงค่าเฉลี่ยมีหน่วยเป็น จำนวนหน่วยประมวลผลชั่วโมง และตารางที่ 4-30 แสดงค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานมีหน่วยเป็น จำนวนหน่วยประมวลผล เนื่องจากระบบที่ใช้ทดสอบเป็นระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน โดยที่พาร์ทิชันแรกมีจำนวนหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผล 64 โหนด เพื่อให้ง่ายต่อการวิเคราะห์ค่าในแต่ละพาร์ทิชันจึงเพิ่มคอลัมน์ 1st/2nd ในตารางที่ 4-3 ถึง 4-5 ซึ่งใช้แสดงอัตราส่วนระหว่างพาร์ทิชันที่ 1 กับพาร์ทิชันที่ 2 ซึ่งพาร์ทิชันแรกควรจะประมวลผลงานเล็ก และสั้นจำนวนมาก ส่วนพาร์ทิชันที่ 2 ควรจะประมวลผลงานใหญ่ และยาวร่วมกับงานเล็กบางส่วนเพื่อให้ระบบมีการกระจายโหลดที่สมดุล ดังนั้นค่าอัตราส่วนจำนวนงานระหว่างพาร์ทิชันที่ 1 กับพาร์ทิชันที่ 2 ควรจะใกล้เคียง 1 แต่ค่าอัตราส่วนลักษณะของงาน และค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานระหว่างพาร์ทิชันที่ 1 กับพาร์ทิชันที่ 2 ควรจะใกล้เคียง 0.5 โดยที่ค่าอัตราส่วนจำนวนงานที่เป็นตัวหนาจะหมายถึงค่าที่ใกล้เคียง 1 ส่วนค่าอัตราส่วนลักษณะของงาน และค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานที่เป็นตัวหนาจะหมายถึงค่าที่ใกล้เคียง 0.5

จากตารางที่ 4-28 จะเห็นว่าจำนวนงานในแต่ละพาร์ทิชันจากนโยบายแบบเอฟซีเอฟเอสแบ็กฟิลใกล้เคียงกันเฉพาะเดือนสิงหาคม และเดือนกันยายนปี ค.ศ. 2003 เดือนมกราคมปี ค.ศ. 2004 นโยบายแบบเอลเอกซ์เอฟแบ็กฟิลจะมีจำนวนงานในแต่ละพาร์ทิชันใกล้เคียงกันเฉพาะเดือนตุลาคมปี ค.ศ. 2003 เดือนมกราคม และเดือนมีนาคมปี ค.ศ. 2004 ส่วนนโยบายแบบใช้เป้าหมายเป็นหลักจะมีจำนวนงานในแต่ละพาร์ทิชันใกล้เคียงกันเฉพาะเดือนกันยายน และเดือนธันวาคมปี ค.ศ. 2003 เดือนกุมภาพันธ์ปี ค.ศ. 2004 จากทั้ง 3 นโยบายจะเห็นได้ว่าไม่มีนโยบายใดสามารถทำให้จำนวนงานแต่ละพาร์ทิชันใกล้เคียงกันได้ดีกว่า ต่อไปจะวิเคราะห์ถึงลักษณะของงานในแต่ละพาร์ทิชันดังตารางที่ 4-29

ตารางที่ 4-28 จำนวนงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Number of jobs									
	FCFS-backfill			LXF-backfill			Goal-oriented		
Partition / Month	1st	2nd	1st/2nd	1st	2nd	1st/2nd	1st	2nd	1st/2nd
6/03	483	1,708	0.28	657	1,534	0.43	781	1,410	0.55
7/03	886	514	1.72	820	580	1.41	760	640	1.19
8/03	1,557	1,664	**0.94**	1,923	1,298	1.48	1,458	1,763	0.83
9/03	1,507	1,550	**0.97**	1,653	1,404	1.18	1,595	1,462	**1.09**
10/03	2,268	1,881	1.21	2,210	1,939	**1.14**	2,329	1,820	1.28
11/03	2,395	1,048	2.29	2,269	1,174	1.93	2,180	1,263	1.73
12/03	1,976	1,545	1.28	1,892	1,629	1.16	1,677	1,844	**0.91**
1/04	1,560	1,596	**0.98**	1,669	1,487	**1.12**	1,475	1,681	0.88
2/04	1,849	2,120	0.87	1,662	2,307	0.72	1,927	2,042	**0.94**
3/04	2,164	1,302	1.66	1,648	1,818	**0.91**	2,159	1,307	1.65

จากตารางที่ 4-29 จะเห็นว่าขนาดของงานจากทั้ง 3 นโยบายที่พาร์ทิชันแรกน้อยกว่าพาร์ทิชันที่ 2 แต่จำนวนงานแต่ละพาร์ทิชันใกล้เคียงกัน แสดงว่างานในพาร์ทิชันแรกเป็นงานขนาดเล็กเป็นส่วนใหญ่ ยกเว้นเดือนมิถุนายนปี ค.ศ. 2003 เพียงเดือนเดียวเท่านั้น ขนาดของงานจากนโยบายแบบเอฟซีเอฟเอสแบ็กฟิลในเดือนสิงหาคม เดือนกันยายน เดือนตุลาคม และเดือนธันวาคมปี ค.ศ. 2003 เดือนมกราคม และเดือนกุมภาพันธ์ปี ค.ศ. 2004 บนพาร์ทิชันแรกมีขนาดเล็กกว่าพาร์ทิชันที่ 2 ครึ่งหนึ่งโดยประมาณ เดือนตุลาคม และเดือนธันวาคมปี ค.ศ. 2003 เดือนมกราคม ปี ค.ศ. 2004 จากนโยบายแบบเอลเอกซ์เอฟแบ็กฟิลจะมีขนาดของงานบนพาร์ทิชันแรกเล็กกว่าพาร์ทิชันที่ 2 ประมาณครึ่งหนึ่ง ส่วนนโยบายแบบใช้เป้าหมายเป็นหลักจะมีขนาดของงานบนพาร์ทิชันแรกเล็กกว่าพาร์ทิชันที่ 2 ประมาณครึ่งหนึ่งเฉพาะเดือนสิงหาคม และเดือนธันวาคมปี ค.ศ. 2003 เดือนมกราคม และเดือนกุมภาพันธ์ปี ค.ศ. 2004 ซึ่งจะเห็นได้ว่าทั้ง 3 นโยบายสามารถกระจายงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชันได้ใกล้เคียงกัน ต่อไปจะวิเคราะห์ถึงค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันดังตารางที่ 4-30

ตารางที่ 4-29 ลักษณะของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Average job size (NT) in node-hour									
	FCFS-backfill			LXF-backfill			Goal-oriented		
Partition / Month	1st	2nd	1st/2nd	1st	2nd	1st/2nd	1st	2nd	1st/2nd
6/03	51.7	28.7	1.80	37.0	32.4	1.14	33.1	34.1	0.97
7/03	22.4	98.0	0.23	22.0	89.9	0.24	24.5	80.5	0.30
8/03	14.2	26.6	**0.53**	10.2	36.0	0.28	15.0	25.2	**0.60**
9/03	13.0	27.0	**0.48**	11.5	30.1	0.38	11.5	29.5	0.39
10/03	9.8	23.4	**0.42**	10.0	22.7	**0.44**	9.4	24.4	0.39
11/03	7.9	41.4	0.19	8.3	37.1	0.22	7.5	36.4	0.21
12/03	11.9	29.7	**0.40**	12.2	28.5	**0.43**	13.2	25.7	**0.51**
1/04	14.4	27.6	**0.52**	13.2	29.9	**0.44**	14.3	27.1	**0.53**
2/04	11.9	19.5	**0.61**	13.3	17.9	0.74	11.1	20.5	**0.54**
3/04	10.8	34.7	0.31	15.4	23.7	0.65	10.5	35.1	0.30

จากตารางที่ 4-30 จะเห็นว่าทั้ง 3 นโยบายงานที่พาร์ทิชันแรกจะเป็นงานที่ต้องการหน่วยประมวลผลน้อยกว่าพาร์ทิชันที่ 2 เป็นส่วนใหญ่ ซึ่งในบางเดือนงานที่พาร์ทิชันแรกต้องการหน่วยประมวลผลน้อยกว่าพาร์ทิชันที่ 2 ถึง 4 เท่าเช่น เดือนกรกฎาคมปี ค.ศ. 2003 เมื่อวิเคราะห์ถึงอัตราส่วนของขนาดงานในพาร์ทิชันที่ 1 ต่อพาร์ทิชันที่ 2 แล้วพบว่าเดือนสิงหาคม และเดือนตุลาคมปี ค.ศ. 2003 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานจากนโยบายแบบเอฟซีเอฟเอสแบ็กฟิลบนพาร์ทิชันแรกน้อยกว่าพาร์ทิชันที่ 2 ประมาณครึ่งหนึ่ง แต่ในเดือนตุลาคมปี ค.ศ. 2003 เดือนกุมภาพันธ์ปี ค.ศ. 2004 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานจากนโยบายแบบเอลเอกซ์เอฟแบ็กฟิลบนพาร์ทิชันแรกน้อยกว่าพาร์ทิชันที่ 2 ประมาณครึ่งหนึ่ง ส่วนนโยบายแบบใช้เป้าหมายเป็นหลักจะมีค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานบนพาร์ทิชันแรกน้อยกว่าพาร์ทิชันที่ 2 ประมาณครึ่งหนึ่งในเดือนสิงหาคม และเดือนตุลาคมปี ค.ศ. 2003 เดือนมกราคม และเดือนกุมภาพันธ์ปี ค.ศ. 2004 ซึ่งจะเห็นได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักสามารถทำให้ความต้องการหน่วยประมวลผลของงานบนพาร์ทิชันแรกน้อยกว่าพาร์ทิชันที่เป็นอัตราส่วนประมาณ 1:2

ตารางที่ 4-30 ค่าเฉลี่ยหน่วยประมวลผลที่ใช้ของงานในแต่ละพาร์ทิชันบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด

Average N									
	FCFS-backfill			LXF-backfill			Goal-oriented		
Partition / Month	1st	2nd	1st/2nd	1st	2nd	1st/2nd	1st	2nd	1st/2nd
6/03	10.5	11.0	0.95	7.9	12.2	0.65	8.7	12.1	0.72
7/03	8.5	36.4	0.23	8.8	32.8	0.27	9.0	30.3	0.30
8/03	3.8	8.3	**0.46**	3.0	10.8	0.28	3.9	8.0	**0.49**
9/03	3.6	12.0	0.30	4.0	12.4	0.32	4.1	12.0	0.34
10/03	3.1	7.0	**0.44**	3.3	6.7	**0.49**	3.1	7.2	**0.43**
11/03	2.9	12.2	0.24	3.2	10.5	0.30	2.8	10.7	0.26
12/03	2.9	8.4	0.35	3.0	8.0	0.38	2.8	7.6	0.37
1/04	7.9	12.6	0.63	9.1	11.7	0.78	5.8	14.2	**0.41**
2/04	2.2	6.8	0.32	3.0	5.9	**0.51**	2.9	6.4	**0.45**
3/04	2.4	10.3	0.23	2.7	7.7	0.35	2.4	10.2	0.24

โดยสรุปความสามารถในการกระจายจำนวนงาน และงานลักษณะต่าง ๆ ของนโยบายแบบใช้เป้าหมายเป็นหลักดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสอง กล่าวคือนโยบายแบบใช้เป้าหมายเป็นหลักจะเลือกส่งงานขนาดเล็กไปพาร์ทิชันที่เล็กกว่าและส่งงานขนาดใหญ่ไปพาร์ทิชันที่ใหญ่กว่า โดยที่งานที่สามารถประมวลผลได้ทั้งสองพาร์ทิชันก็จะถูกส่งไปที่พาร์ทิชันที่มีโหลดน้อยกว่า เพื่อให้ระบบมีโหลดที่สมดุล ซึ่งจะทำให้ประสิทธิภาพของระบบดีขึ้น

เมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากัน นโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ซึ่งเหมือนกับระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเมื่อใช้ค่าระยะเวลาการประมวลผลจริง (หัวข้อ 4.2.2) แต่ประสิทธิภาพที่ได้ไม่ชัดเจนมากเหมือนการใช้ค่าระยะเวลาการประมวลผลจริง

จากที่ได้กล่าวมาในหัวข้อ 4.4 แสดงให้เห็นถึงประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบหลาย

พาร์ทิชันที่มีจำนวนหน่วยประมวลผลเท่ากันจำนวน 2 ถึง 5 พาร์ทิชัน และจำนวนหน่วยประมวลผลไม่เท่ากันเฉพาะ 2 พาร์ทิชัน เมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด สรุปได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ประสิทธิภาพดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเพียงแต่ไม่เห็นชัดเจนมากเหมือนการใช้ค่าระยะเวลาการประมวลผลจริง (หัวข้อที่ 4.2) อีกทั้งความสามารถกระจายจำนวนงาน และงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชันของนโยบายแบบใช้เป้าหมายเป็นหลักยังคงดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล กล่าวคือบนระบบที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากันทั้งจำนวนงาน และขนาดของงานแต่ละพาร์ทิชันจะใกล้เคียงกันมากกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ส่วนบนระบบที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากัน นโยบายแบบใช้เป้าหมายเป็นหลักจะเลือกส่งงานขนาดเล็กไปพาร์ทิชันที่เล็กกว่า และส่งงานขนาดใหญ่ไปพาร์ทิชันที่ใหญ่กว่าโดยจะทำให้จำนวนงาน และโหลดแต่ละพาร์ทิชันใกล้เคียงกันมากที่สุด

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงการสรุปผล ข้อเสนอแนะที่ได้จากการดำเนินการทำงานวิจัย ตลอดจนปัญหา และอุปสรรคที่เกิดขึ้นขณะทำงานวิจัย และให้ข้อเสนอแนะแก่ผู้สนใจที่จะนำงานวิจัยชิ้นนี้ไปพัฒนาต่อ

5.1 สรุปผล

งานวิจัยชิ้นนี้นำเสนอประสิทธิภาพของตัวจัดกำหนดการแบบขนานโดยใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน ทั้งแบบที่แต่ละพาร์ทิชันมีขนาดเท่ากันจำนวน 2 ถึง 5 พาร์ทิชัน และไม่เท่ากันจำนวน 2 พาร์ทิชัน โดยปรับปรุงนโยบายของตัวจัดกำหนดการแบบขนานโดยใช้เป้าหมายเป็นหลักเพื่อให้ทำงานบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันได้ ซึ่งทำการทดสอบ เปรียบเทียบ และวิเคราะห์ประสิทธิภาพของการใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบหลายพาร์ทิชัน 4 กรณีคือ

1. เปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันทั้ง 2 วิธีคือแบบผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ และแบบการใช้ฮิวริสทิคในการเลือกพาร์ทิชันสรุปได้ว่าแบบการใช้ฮิวริสทิคในการเลือกพาร์ทิชันโดยใช้เบสฟิทเป็นฮิวริสทิคมีประสิทธิภาพดีกว่าแบบผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้อันเนื่องมาจากแบบผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้จะทำให้พื้นที่สำหรับค้นหาการจัดกำหนดการงานแบบขนานที่เหมาะสมใหญ่ขึ้น ส่งผลให้บางครั้งการค้นหาการจัดลำดับที่ค้นหาพบในเวลาที่กำหนดไม่ใช่การจัดลำดับงานที่ดี กล่าวคือเป็นการจัดลำดับงานที่ไม่ทำให้ระบบมีประสิทธิภาพดี
2. เปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลได้แก่ (FCFS-backfill และ LXF-backfill) บนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน โดยแบ่งการทดสอบเป็น 2 แบบคือ
    2.1 แบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากันได้ทดสอบจำนวนพาร์ทิชัน 2 ถึง 5 พาร์ทิชัน ซึ่งระบบ 2 พาร์ทิชันแต่ละพาร์ทิชันจะมีจำนวนหน่วยประมวลผล 64 โหนด และระบบ 3 ถึง 5 พาร์ทิชันแต่ละพาร์ทิชันจะมี

จำนวนหน่วยประมวลผล 32 โหนด จากการทดสอบพบว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเมื่อจำนวนพาร์ทิชันน้อย แต่เมื่อจำนวนพาร์ทิชันเพิ่มขึ้นนโยบายแบบใช้เป้าหมายเป็นหลักยังคงมีประสิทธิภาพดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลแต่จะไม่ชัดเจนเท่ากับเมื่อระบบมีจำนวนพาร์ทิชันน้อย นั่นก็เพราะโหลดของระบบที่ลดลงเมื่อจำนวนพาร์ทิชันเพิ่มขึ้น อีกทั้งนโยบายแบบใช้เป้าหมายเป็นหลักยังสามารถกระจายจำนวนงาน และงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชันได้ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล ซึ่งประสิทธิภาพการกระจายงานของนโยบายแบบใช้เป้าหมายเป็นหลักจะเห็นได้ชัดเมื่อระบบมีจำนวนพาร์ทิชันน้อย แต่เมื่อระบบมีจำนวนพาร์ทิชันเพิ่มขึ้นการกระจายงานไปแต่ละพาร์ทิชันยังคงดีกว่าในเดือนที่โหลดยังคงสูงระดับหนึ่งเท่านั้น

2.2 แบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากันได้ทดสอบบนระบบ 2 พาร์ทิชันเท่านั้น โดยพาร์ทิชันแรกมีจำนวนหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผล 64 โหนด จากการทดสอบพบว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเช่นเดียวกับระบบที่แต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากัน ส่วนการกระจายงานไปแต่ละพาร์ทิชันพบว่านโยบายแบบใช้เป้าหมายเป็นหลักสามารถกระจายจำนวนงาน และงานลักษณะต่าง ๆ ไปแต่ละพาร์ทิชันได้ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเนื่องจากนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลจะส่งงานขนาดเล็กไปพาร์ทิชันที่ 1 และงานขนาดใหญ่ไปพาร์ทิชันที่ 2 ทำให้จำนวนงานในพาร์ทิชันที่ 1 มีมากกว่าพาร์ทิชันที่ 2 มาก ซึ่งต่างจากนโยบายแบบใช้เป้าหมายเป็นหลักที่จะส่งงานใหญ่ และงานเล็กบางงานไปที่พาร์ทิชันที่ 2 ด้วย ทำให้จำนวนงาน และลักษณะของงานในพาร์ทิชันที่ 1 และ 2 ใกล้เคียงกันมากกว่า

3. วิเคราะห์ประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชันเมื่อให้เวลาในการตัดสินใจของตัวจัดกำหนดการงานแบบขนานเพิ่มขึ้น ได้ทดสอบบนระบบ 2 พาร์ทิชันที่มีจำนวนหน่วยประมวลผลไม่เท่ากันเท่านั้น โดยพาร์ทิชันแรกมีจำนวนหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2

มีจำนวนหน่วยประมวลผล 64 โหนด จากการทดสอบพบว่าเมื่อให้เวลาในการตัดสินใจเพิ่มขึ้นประสิทธิภาพที่ได้ของระบบก็จะดีขึ้นด้วย เมื่อวิเคราะห์ประสิทธิภาพของแต่ละพาร์ทิชันพบว่าเมื่อพาร์ทิชันแรกมีประสิทธิภาพดีขึ้น พาร์ทิชันที่ 2 ยังคงมีประสิทธิภาพคงที่หรือใกล้เคียงค่าเดิม

4. เปรียบเทียบประสิทธิภาพของนโยบายแบบใช้เป้าหมายเป็นหลักกับนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลบนระบบหลายพาร์ทิชันเมื่อใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นกำหนด โดยทดสอบ 2 แบบคือ

    4.1 แบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลเท่ากันได้ทำการทดสอบจำนวนพาร์ทิชัน 2 ถึง 5 พาร์ทิชัน โดยที่บนระบบ 2 พาร์ทิชันแต่ละพาร์ทิชันจะมีจำนวนหน่วยประมวลผล 64 โหนด และระบบ 3 ถึง 5 พาร์ทิชันแต่ละพาร์ทิชันจะมีจำนวนหน่วยประมวลผล 32 โหนด จากการทดสอบพบว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลทั้งสองมีค่าประสิทธิภาพต่าง ๆ ใกล้เคียงกันมากเมื่อจำนวนพาร์ทิชันเพิ่มขึ้น แต่นโยบายแบบใช้เป้าหมายเป็นหลักยังคงให้ค่าประสิทธิภาพต่าง ๆ ใกล้เคียงค่าที่ดีที่สุด สามารถสรุปได้ว่านโยบายแบบใช้เป้าหมายเป็นหลักมีประสิทธิภาพดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิล อีกทั้งนโยบายแบบใช้เป้าหมายเป็นหลักสามารถกระจายงานลักษณะต่าง ๆ และจำนวนงานไปแต่ละพาร์ทิชันได้ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลเพราะนโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลจะส่งงานไปให้พาร์ทิชันแรกเป็นลำดับแรก ซึ่งจะเห็นชัดเจนจากตารางของจำนวนงานและลักษณะของงานของ 4 และ 5 พาร์ทิชัน

    4.2 แบบแต่ละพาร์ทิชันมีจำนวนหน่วยประมวลผลไม่เท่ากันได้ทดสอบบนระบบ 2 พาร์ทิชันเท่านั้น โดยพาร์ทิชันแรกมีจำนวนหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผล 64 โหนด จากการทดสอบพบว่านโยบายแบบใช้เป้าหมายเป็นหลักยังคงมีประสิทธิภาพที่ดีกว่านโยบายแบบให้ลำดับความสำคัญร่วมกับเทคนิคแบ็กฟิลหรือใกล้เคียงค่าที่ดีทุกประสิทธิภาพที่แสดงไว้ แต่ความสามารถในการกระจายงานไปแต่ละพาร์ทิชันไม่ชัดเจนเหมือนการใช้ค่าระยะเวลาการประมวลผลจริง

5.2 ผลที่ได้จากการทำวิทยานิพนธ์ชุดนี้

ผู้ทำวิทยานิพนธ์ได้ศึกษาและวิเคราะห์ระบบการจัดกำหนดการงานแบบขนาน โดยใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน โดยนำเสนอประสิทธิภาพ และแนวทางในการจัดการระบบการจัดกำหนดการงานแบบขนานโดยใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน

5.3 ข้อเสนอแนะ

งานวิจัยชิ้นนี้นำเสนอแนวคิดของการใช้นโยบายแบบใช้เป้าหมายเป็นหลักบนระบบคอมพิวเตอร์แบบขนานแบบหลายพาร์ทิชัน แนวคิดในงานวิจัยนี้สามารถนำไปประยุกต์ใช้งานกับระบบกริด หรือระบบแบบกระจายได้ โดยใช้หลักการของตัวจัดกำหนดการงานแบบขนานแบบส่วนกลาง แต่แนวคิดที่เสนอในงานวิจัยนี้ยังไม่มีการคำนึงถึงการส่งข้อมูลไปมาระหว่างโฮสท์หรือพาร์ทิชัน ดังนั้นการนำไปประยุกต์ใช้ในระบบดังกล่าวควรคำนึงถึงประเด็นการส่งข้อมูลระหว่างส่วนต่าง ๆ ด้วย การทดสอบแนวคิดจะทดสอบเป็นรายเดือน ซึ่งแต่ละเดือนลักษณะของงานมีความแตกต่างกันแต่ข้อมูลทั้งหมดก็มาจากแหล่งเดียวกัน ผู้สนใจสามารถพัฒนาต่อโดยทำการทดสอบกับเวิร์คโหลดจากหลายแหล่ง นอกจากนี้ส่วนของฮิวริสทิคในการเลือกพาร์ทิชันสามารถส่งผลกระทบต่อประสิทธิภาพของระบบได้ ในงานวิจัยชิ้นนี้ใช้เพียงเบสฟิทเป็นฮิวริสทิคในการเลือกพาร์ทิชัน ผู้สนใจสามารถทดสอบประสิทธิภาพของแนวคิดในงานวิจัยชิ้นนี้โดยการปรับเปลี่ยนการใช้ฮิวริสทิคในการเลือกพาร์ทิชันได้

บรรณานุกรม

[1] *Maui Scheduler*, http://www.supercluster.org/maui/.

[2] *MOAB*, http://www.supercluster.org/moab.

[3] *PBS Scheduler*, http://www.nas.nasa.gov/Software/PBS/.

[4] *Platform LSF*, http://www.platform.com/products/LSFfamily/.

[5] S. Kannan, M. Roberts, P. Mayes, D. Brelsford, and J.F. Skovira, "Workload Management with Loadleveler", IBM, Nov. 2001.

[6] S.-H. Chiang & S. Vasupongayya, "Design and potential performance of goal-oriented job scheduling policies for parallel computer workloads", *IEEE Tran. on Parallel and Distributed Systems*, pp. 1642-1656, 2008.

[7] S. Vasupongayya, "Goal-oriented parallel job scheduling: A revisit", *Proc. of the 2nd UBU-Research*, Ubonratchathani, Thailand, July 2008.

[8] S. Vasupongayya, "Achieving fair share objectives via goal-oriented parallel computer job scheduling policies", *Proc. WASET ICCSE'09*, Bangkok, Thailand, December 25-27, 2009.

[9] S. Vasupongayya, "Impact of User Runtime Estimates on Achieving Fair Share Objectives", *Proc. TISD*, Nong Khai, Thailand, March 4-6, 2010.

[10] S. Vasupongayya, "Impact of Workloads on Fair Share Policies", *Proc. ANSCSE14*, Chiang Rai, Thailand, March 23-26, 2010.

[11] D. Lifka, "The ANL/IBM SP scheduling system", *In Workshop on JSSPP*, 1995.

[12] A.W. Mu'alem and D.G. Feitelson, "Utilization, Predictability, Workloads, and User Runtime Estimates in Scheduling the IBM SP2 with Backfilling", *IEEE Trans. Parallel and Distributed System*, vol.12, no.6, pp. 529–543, June 2001.

[13] S. Vasupongayya, S.-H. Chiang, and B. Massey, "Search-Based Job Scheduling for Parallel Computer Workloads", *Proc. IEEE Int'l Conf. Cluster Computing (CLUSTER '05)*, Sept. 2005.

[14] W.D. Harvey and M.L. Ginsberg, "Limited Discrepancy Search", Proc. *14th Int'l Joint Conf. Artificial Intelligence (IJCAI '95)*, Aug. 1995.

[15] R.E. Korf, "Improved Limited Discrepancy Search", *Proc. 13th Nat'l Conf. Artificial Intelligence (AAAI '96)*, pp. 209-215, Aug. 1996.

[16] T. Walsh, "Depth-Bounded Discrepancy Search", *Proc. 15th Int'l Joint Conf. Artificial Intelligence (IJCAI '97)*, vol. 2, pp. 1388-1393, Aug. 1997.

[17] S. Vasupongayya and S.-H. Chiang, "Multi-Objective Models for Scheduling Jobs on Parallel Computer Systems", *Proc. IEEE Int'l Conf. Cluster Computing (CLUSTER '06)*, Sept. 2006.

[18] F. Guim, J.Corbalan, "A Job Self-Scheduling Policy for HPC Infrastructures", *In Job Scheduling Strategies for Parallel Processing, Lect. Notes Comput. Sci.*, vol. 4942, pp. 51-75, 2008.

[19] C. Ernemann, V. Hamscher and R. Yahyapour. "Benefits of global grid computing for job scheduling", *5th IEEE/ACM International Workshop on Grid Computing*, 2004.

[20] B. Schroeder and M. Harchol-Balter. "Evaluation of task assignment policies for supercomputing servers: The case for load unbalancing and fairness", *Cluster Computing 2004*, 2004.

[21] *Parallel Workloads Archive*, http://www.cs.huji.ac.il/labs/parallel/workload/models.html.

[22] *The Standard Workload Format*, http://www.cs.huji.ac.il/labs/parallel/workload/swf.html.

[23] M. Calzarossa and G. Serazzi, "Workload characterization: a survey", *Proc. IEEE 81(8)*, pp. 1136-1150, Aug 1993.

[24] V. Lo, J. Mache and K. Windisch, "A comparative study of real workload traces and synthetic workload models for parallel job scheduling", *In Job Scheduling Strategies for Parallel Processing, Lect. Notes Comput. Sci.*, vol. 1459, pp. 25-47, Springer Verlag, 1998.

[25] D.G. Feitelson, "Metrics for Parallel Job Scheduling and Their Convergence", *In Job Scheduling Strategies for Parallel Processing, Lect. Notes Comput. Sci.*, vol. 2221, pp. 188-205, Springer Verlag, 2001.

[26] D. G. Feitelson, L. Rudolph, U. Schwiegelshohn, K. C. Sevcik, and P. Wong, "Theory and practice in parallel job scheduling", *In Job Scheduling Strategies for Parallel Processing. Lect. Notes Comput. Sci.*, vol. 1291, pp. 1-34, Springer Verlag, 1997.

[27] D.G. Feitelson and L. Rudolph, "Metrics and Benchmarking for Parallel Job Scheduling", *In Job Scheduling Strategies for Parallel Processing, Lect. Notes Comput. Sci.*, vol. 1459, pp. 1-24, Springer Verlag, 1998.

ภาคผนวก

ภาคผนวก ก

การใช้งานโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน

การใช้งานโปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน

การทดสอบในวิทยานิพนธ์ชิ้นนี้จะทดสอบโดยใช้โปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานเท่านั้น โดยนำโปรแกรมที่พัฒนาโดย Chuyong Fu ในปี ค.ศ. 2003 ตัวโปรแกรมสามารถทำงานกับนโยบายแบบเอฟซีเอฟเอส และอีซีเอฟซีเอฟเอสแบ็กฟิลได้เท่านั้น ทั้งหมดทำงานบนระบบแบบพาร์ทิชันเดียว ดังนั้นต้องแก้ไขปรับปรุงโดยการเพิ่มนโยบายแบบใช้เป้าหมายเป็นหลัก และนโยบายแบบอีซีเบสฟิทเอฟซีเอฟเอสแบ็กฟิลทั้งเอฟซีเอฟเอสและเอลเอกซ์เอฟ อีกทั้งต้องปรับปรุงให้รองรับระบบแบบพาร์ทิชันเดียวและระบบแบบหลายพาร์ทิชัน การใช้งานโปรแกรมจะแบ่งออกเป็น 3 ส่วนด้วยกันคือ รูปแบบของเวิร์คโหลดไฟล์ (อินพุท), วิธีรันโปรแกรม และเอาพุทของโปรแกรม

ก1. รูปแบบของเวิร์คโหลดไฟล์

โปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานที่ใช้ทดสอบจะใช้เวิร์คโหลดไฟล์เป็นอินพุทของโปรแกรม ทั้งนี้ตัวข้อมูลในเวิร์คโหลดไฟล์จะถูกอ่านเข้ามาเก็บไว้ในบัฟเฟอร์ (Buffer) เพื่อจำลองการทำงานของระบบ โปรแกรมจะจำลองเวลาขึ้นมาเป็นเวลาของระบบจำลอง เมื่อเวลาเดินไปถึงเวลาที่งานเข้ามาในระบบ งานก็จะถูกนำเข้ามาให้ตัวจัดกำหนดการงานแบบขนานเลือกว่าจะสามารถประมวลผลได้หรือไม่ ณ เวลาปัจจุบันของโปรแกรม โปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนานสามารถใช้เวิร์คโหลดไฟล์ได้ 2 รูปแบบคือ แบบซิมพลิไฟด์ ซึ่งมี 6 คอลัมน์ดังภาพประกอบที่ ก-1 และแบบเอสดับบิวเอฟซึ่งมี 18 คอลัมน์ดังภาพประกอบที่ ก-2 โดยสามารถกำหนดรูปแบบของไฟล์ผ่านทางพารามิเตอร์ขณะสั่งรันโปรแกรม ในตัวโปรแกรมจำลองจะใช้ข้อมูลเพียง 6 คอลัมน์ซึ่งเป็นแบบซิมพลิไฟด์ ดังนั้นต้องมีการแปลงเวิร์คโหลดไฟล์แบบเอสดับบิวเอฟให้เป็นแบบซิมพลิไฟด์ ในส่วนของการแปลงนี้มีอยู่แล้วในตัวโปรแกรม โดยจะเลือกใช้เฉพาะบางคอลัมน์จากไฟล์แบบเอสดับบิวเอฟ ส่วนในคอลัมน์อื่น ๆ ที่ไม่ได้กล่าวถึงจะไม่ได้ใช้งาน คอลัมน์ที่ใช้มีดังนี้

- คอลัมน์ที่ 1 เป็นหมายเลขของงาน
- คอลัมน์ที่ 2 เป็นเวลาที่งานเข้ามาในระบบ ซึ่งอยู่ในรูปแบบของยูนิกซ์ไทม์ (Unix Time)
- คอลัมน์ที่ 4 เป็นค่าระยะเวลาการประมวลผลจริงของงานมีหน่วยเป็นวินาที
- คอลัมน์ที่ 5 เป็นจำนวนหน่วยประมวลผลที่งานต้องการใช้

- คอลัมน์ที่ 9 เป็นค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนดมีหน่วยเป็นวินาที
- คอลัมน์ที่ 10 เป็นค่าสถานะว่างานนี้ได้รับการประมวลผลหรือไม่ ซึ่งจะไม่ถูกใช้ในโปรแกรมขณะนำงานมาจัดลำดับ แต่ที่ต้องกำหนดให้โปรแกรมเพราะต้องกำหนดให้ถูกต้องตามรูปแบบไฟล์แบบซิมพลิไฟด์

```
1 0 937 306 35100 1
2 164472 75 17 300 1
3 197154 35268 306 35100 0
4 310448 29493 64 64800 1
5 310541 29063 64 64800 1
6 310684 33707 64 64800 0
7 319091 47451 16 54000 1
8 319101 49724 16 54000 1
9 324746 61578 1 64800 1
10 324747 61578 32 64800 1
11 324748 61578 302 64800 1
12 324749 61578 132 64800 1
13 324750 61578 132 64800 1
14 324751 61578 162 64800 1
15 324752 61578 100 64800 1
16 324753 61578 100 64800 1
```

ภาพประกอบ ก-1 เวิร์คโหลดไฟล์แบบซิมพลิไฟด์

```
1 0 12 40 32 -1 377286 32 600 -1 -1 1 -1 1 1 -1 -1 -1
2 75 1 304 32 -1 97494 32 300 -1 -1 1 -1 1 1 -1 -1 -1
3 880 19 304 32 -1 184118 32 300 -1 -1 1 -1 1 1 -1 -1 -1
4 23784 12 7113 32 -1 5221062 32 8100 -1 -1 2 -1 2 1 -1 -1 -1
5 23936 7 31 56 -1 -1 56 28800 -1 -1 3 -1 3 1 -1 -1 -1
6 24074 7 11829 56 -1 1283231 56 28800 -1 -1 3 -1 3 1 -1 -1 -1
7 27923 10 33511 16 -1 4878995 16 43200 -1 -1 4 -1 4 1 -1 -1 -1
8 28704 179 17188 20 -1 5366456 20 21600 -1 -1 5 -1 5 1 -1 -1 -1
9 31546 4 5370 8 -1 200051 8 43200 -1 -1 6 -1 6 1 -1 -1 -1
10 32013 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 1 -1 -1 -1
11 32359 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 1 -1 -1 -1
12 32491 20 139 1 -1 14233 1 600 -1 -1 7 -1 7 1 -1 -1 -1
13 32557 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 1 -1 -1 -1
14 32645 22 428 2 -1 8601 2 600 -1 -1 7 -1 7 1 -1 -1 -1
15 33526 7 40 28 -1 92580 28 180 -1 -1 1 -1 1 1 -1 -1 -1
16 33527 2684 148 32 -1 9472 32 1200 -1 -1 8 -1 8 1 -1 -1 -1
```

ภาพประกอบ ก-2 เวิร์คโหลดไฟล์แบบเอสดับบิวเอฟ

ก2. วิธีรันโปรแกรม

เมื่อผู้ใช้กำหนดนโยบายที่จะใช้สำหรับโปรแกรมจำลองระบบคอมพิวเตอร์แบบขนานผ่านทางโปรแกรมหลักแล้ว โปรแกรมจะสร้างตัวจัดกำหนดการงานแบบขนานที่ใช้นโยบายสำหรับกำหนดลำดับงานตามที่ระบบกำหนดไว้ เมื่อเวลาของโปรแกรมตรงกับเวลาที่งานถูกส่งเข้ามาสู่ระบบ และมีงานอยู่ในคิวอย่างน้อย 1 งานระบบก็จะเริ่มจัดลำดับงานตามนโยบายที่ใช้จนกว่างานที่ถูกอ่านมาจากเวิร์คโหลดไฟล์จะหมดหรือเกินช่วงเวลาของงานที่ได้กำหนดไว้ การรันโปรแกรมต้องมีการกำหนดค่าพารามิเตอร์ ซึ่งพารามิเตอร์มีทั้งแบบที่จำเป็นต้องใช้ทุกครั้งที่รันโปรแกรมและพารามิเตอร์เสริมเพื่อใช้กำหนดค่าบางค่าเพิ่มเติมจากที่โปรแกรมกำหนดให้ดังนี้

ก2.1 รูปแบบคำสั่ง

java SchDriver [-f < filename>] <-t policy> [-a] [-e] [-d] [-r runtime] <-h partition> [-m start end] [-l start end]

- -f filename กำหนดชื่อเวิร์คโหลดไฟล์ถ้าไม่มีการกำหนดมาให้โปรแกรมจะกำหนดให้เป็น "job.log"
- -t policy กำหนดว่าจะใช้นโยบายอะไร ซึ่งการใช้นโยบายต่าง ๆ จะมีพารามิเตอร์ที่ต้องใช้ควบคู่กันไปแตกต่างกันดังนี้
  - "FCFS" คือนโยบายแบบเอฟซีเอฟเอส
  - "EASY" คือนโยบายแบบเอฟซีเอฟเอสร่วมกับเทคนิคอีซีแบ็กฟิล
  - "EBF" <fcfs | lxf> เลือกใช้นโยบายแบบเอฟซีเอฟเอสหรือเอลเอกซ์เอฟร่วมกับเทคนิคอีซีแบ็กฟิลและใช้วิธีการเลือกพาร์ทิชันเป็นเบสฟิท
  - "DDS" <fcfs | lxf> [objective] <node limit> เป็นการใช้นโยบายแบบใช้เป้าหมายเป็นหลักแบบหลายพาร์ทิชันวิธีผนวกการเลือกพาร์ทิชันเข้ากับโครงสร้างต้นไม้ ซึ่งสามารถเลือกใช้ 2 แบบคือ
    - fcfs | lxf เป็นวิธีการจัดเรียงงานในคิวเพื่อสร้างต้นไม้ ซึ่งในโปรแกรมได้พัฒนาไว้ 2 วิธีคือ เอฟซีเอฟเอสและเอลเอกซ์เอฟ
    - objective เป็นการกำหนดรูปแบบของเป้าหมายและเป้าหมายที่จะใช้ ซึ่งในโปรแกรมได้พัฒนาเป้าหมายไว้ 7 ชุดดังนี้
      - ถ้าไม่ใช้โปรแกรมจะกำหนดให้เป็นการใช้เป้าหมายเดียวคือการทำให้ค่าสูงสุดของการรอต่ำที่สุด
      - "avgWait" เป็นการใช้เป้าหมายเดียวคือค่าเฉลี่ยการรอ

- o "lexical" เป็นแบบ 2 เป้าหมายโดยใช้รูปแบบของเป้าหมายเป็นเล็กซิคอลและใช้เป้าหมายคือการทำให้ค่าสูงสุดของการรอต่ำที่สุดกับการทำให้ค่าเฉลี่ยการรอต่ำที่สุด
- o "tradeoffA" เป็นแบบ 2 เป้าหมายโดยใช้รูปแบบของเป้าหมายเป็นอีควลเทรดออฟและใช้เป้าหมายคือการทำให้ค่าสูงสุดของการรอต่ำที่สุดกับการทำให้ค่าเฉลี่ยการรอต่ำที่สุด
- o "lexical[maxW-avgX]" เป็นแบบ 2 เป้าหมายโดยใช้รูปแบบของเป้าหมายเป็นเล็กซิคอลและใช้เป้าหมายคือการทำให้ค่าสูงสุดของการรอต่ำที่สุดกับการทำให้ค่าเฉลี่ยสโลว์ดาวน์ต่ำที่สุด
- o "lexical[Tw-avgX]" เป็นแบบ 2 เป้าหมายโดยใช้รูปแบบของเป้าหมายเป็นเล็กซิคอลและใช้เป้าหมายคือการทำให้ค่าผลรวมการรอที่มากเกินกำหนดต่ำที่สุดกับการทำให้ค่าเฉลี่ยสโลว์ดาวน์ต่ำที่สุด
- o "tradeoffA[Tw-avgX]" เป็นแบบ 2 เป้าหมายโดยใช้รูปแบบของเป้าหมายเป็นอีควลเทรดออฟและใช้เป้าหมายคือการทำให้ค่าผลรวมการรอที่มากเกินกำหนดต่ำที่สุดกับการทำให้ค่าเฉลี่ยสโลว์ดาวน์ต่ำที่สุด

- node limit จำนวนโหนดที่ค้นหาพบ ซึ่งสามารถใส่ได้ตั้งแต่ 2,000 – 128,000 ต้องใส่เป็นตัวเลขจำนวนเต็มที่ไม่มีจุลภาคเช่น 2000, 4000 เป็นต้น
- "GoalM2" <fcfs | lxf> <objective> <node limit> เป็นการใช้นโยบายแบบใช้เป้าหมายเป็นหลักแบบหลายพาร์ทิชันด้วยการใช้ฮิวริสทิคในการเลือกพาร์ทิชันในการเลือกพาร์ทิชันโดยใช้ฮิวริสทิคในการเลือกพาร์ทิชันเป็นเบสฟิท
- fcfs | lxf เป็นวิธีการจัดเรียงงานในคิวเพื่อสร้างต้นไม้ ซึ่งในโปรแกรมได้พัฒนาไว้ 2 วิธีคือเอฟซีเอฟเอสและเอลเอกซ์เอฟ
- objective เป็นการกำหนดรูปแบบของเป้าหมายและเป้าหมายที่จะใช้ ซึ่งในโปรแกรมได้พัฒนาเป้าหมายไว้ชุดเดียวคือ
  - o "tradeoffA[Tw-avgX]" เป็นแบบ 2 เป้าหมายโดยใช้รูปแบบของเป้าหมายคืออีควลเทรดออฟและใช้เป้าหมายคือการทำให้ค่า

ผลรวมการรอที่มากเกินกำหนดต่ำที่สุดกับการทำให้ค่าเฉลี่ยสโลว์ดาวน์ต่ำที่สุด

- node limit จำนวนโหนดที่ค้นหาพบ ซึ่งสามารถใส่ได้ตั้งแต่ 2,000 – 128,000 ต้องใส่เป็นตัวเลขจำนวนเต็มที่ไม่มีจุลภาคเช่น 2000, 4000 เป็นต้น

- -a กำหนดว่าจะใช้เวิร์คโหลดไฟล์แบบใด ถ้าไม่กำหนดโปรแกรมจะกำหนดให้เป็นแบบซิมพลิไฟด์ แต่ถ้าใช้พารามิเตอร์นี้จะหมายถึงให้ใช้เวิร์คโหลดไฟล์แบบเอสดับบิวเอฟ
- -e กำหนดว่าจะให้ตัวจัดกำหนดการงานแบบขนานใช้ค่าระยะเวลาการประมวลผลแบบใด ถ้ากำหนดพารามิเตอร์นี้ขณะรันโปรแกรมตัวจัดกำหนดการงานแบบขนานจะใช้ค่าระยะเวลาการประมวลผลจริง แต่ถ้าไม่กำหนดไว้โปรแกรมจะกำหนดให้ใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด
- -d ให้แสดงการดีบัก (Debug) โปรแกรมออกมาทางหน้าจอ
- -r runtime กำหนดให้ตัวจัดกำหนดการใช้ค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด และกำหนดค่าสูงสุดของระยะเวลาการประมวลผลของงาน ซึ่งงานแต่ละงานจะใช้เวลาได้ไม่เกินค่าที่กำหนดไว้ โดยใส่เป็นตัวเลขจำนวนเต็มที่ไม่มีจุลภาค
- -h partition กำหนดจำนวนพาร์ทิชันของระบบจำเป็นต้องใช้ทุกครั้งและต้องใส่พารามิเตอร์ตามหลังอีก 2 ค่าคือ
  - –p maxprocessor กำหนดจำนวนหน่วยประมวลผลของพาร์ทิชันที่ใหญ่ที่สุด
  - –c processor กำหนดจำนวนหน่วยประมวลผลของแต่ละพาร์ทิชันเช่น –c 32 64 หมายความว่าพาร์ทิชันแรกมีจำนวนหน่วยประมวลผล 32 โหนด และพาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผล 64 โหนด
- -m start end กำหนดช่วงของงานที่จะประมวลผลโดยใช้ค่าหมายเลขของงาน ซึ่งต้องกำหนดตัวเลขเริ่มต้นและตัวเลขสุดท้าย ถ้าไม่มีการกำหนดโปรแกรมจะประมวลผลทุกงานที่อ่านได้จากเวิร์คโหลดไฟล์
- -l start end กำหนดช่วงของงานที่จะประมวลผลโดยใช้ค่าเวลาที่งานถูกส่งเข้ามาในระบบ ซึ่งต้องกำหนดตัวเลขเริ่มต้นและตัวเลขสุดท้าย ถ้าไม่มีการกำหนดโปรแกรมจะประมวลผลทุกงานที่อ่านได้จากเวิร์คโหลดไฟล์

ก2.2 ตัวอย่างการรันโปรแกรม และความหมาย

- คำสั่ง: java SchDriver -f "titan.dat" -t "GoalM2" "lxf" "tradeoffA[Tw-avgX]" 4000 -a -h 2 -p 64 -c 32 64 -l 4660633 7943832
- ความหมาย: ใช้นโยบายแบบใช้เป้าหมายเป็นหลักแบบหลายพาร์ทิชันมีการจัดเรียงงานของโครงสร้างต้นไม้แบบเอลเอกซ์เอฟ ใช้รูปแบบของเป้าหมายเป็นอีควลเทรดออฟและตั้งเป้าหมายคือ การทำให้ค่าผลรวมการรอที่มากเกินกำหนดต่ำที่สุดกับการทำให้ค่าเฉลี่ยสโลว์ดาวน์ต่ำที่สุด มีการกำหนดจำนวนโหนดที่ค้นหาพบเท่ากับ 4,000 โหนด เวิร์คโหลดไฟล์ที่ใช้เป็นแบบเอสดับบิวเอฟ และกำหนดให้ระบบที่จำลองขึ้นมามี 2 พาร์ทิชัน พาร์ทิชันที่ 1 มีจำนวนหน่วยประมวลผลเท่ากับ 32 โหนด พาร์ทิชันที่ 2 มีจำนวนหน่วยประมวลผลเท่ากับ 64 โหนด ซึ่งพาร์ทิชันที่ใหญ่ที่สุดมีจำนวนหน่วยประมวลผลเท่ากับ 64 โหนด กำหนดให้ระบบเริ่มประมวลผลงานในเวิร์คโหลดไฟล์ตั้งแต่เวลา 4660633 ถึง 7943832 ของเวลาที่งานถูกส่งเข้ามาในระบบ

ก3. เอาพุทของโปรแกรม

เอาพุทที่ได้จากโปรแกรมจะเป็นข้อความจำนวน 9 คอลัมน์แต่ละคอลัมน์จะแสดงข้อมูลดังนี้

- คอลัมน์ที่ 1 แสดงหมายเลขของงาน
- คอลัมน์ที่ 2 แสดงหมายเลขของพาร์ทิชันที่งานไปประมวลผล
- คอลัมน์ที่ 3 แสดงเวลาที่งานถูกส่งเข้ามาสู่ระบบซึ่งอยู่ในรูปแบบของยูนิกซ์ไทม์
- คอลัมน์ที่ 4 แสดงค่าระยะเวลาการประมวลผลจริงเป็นหน่วยวินาที
- คอลัมน์ที่ 5 แสดงความต้องการหน่วยประมวลผลของงาน
- คอลัมน์ที่ 6 แสดงเวลาที่งานเริ่มต้นประมวลผลซึ่งอยู่ในรูปแบบของยูนิกซ์ไทม์
- คอลัมน์ที่ 7 แสดงเวลาที่น้อยที่สุดระหว่างค่าระยะเวลาการประมวลผลจริงกับค่าประมาณระยะเวลาประมวลผลซึ่งผู้ใช้เป็นผู้กำหนด ซึ่งอยู่ในรูปแบบของยูนิกซ์ไทม์
- คอลัมน์ที่ 8 บอกให้ทราบว่างานได้รับการประมวลผลหรือไม่ 0 จะหมายถึงไม่ได้รับการประมวลผล และ 1 จะหมายถึงได้รับการประมวลผลแล้ว
- คอลัมน์ที่ 9 แสดงเวลาที่งานประมวลผลเสร็จ ซึ่งอยู่ในรูปแบบของยูนิกซ์ไทม์

ในส่วนของเอาพุทนี้ผู้ใช้สามารถแก้ไขรูปแบบของเอาพุทที่ออกมาได้ตามต้องการ ซึ่งจะไม่ส่งผลกระทบถึงในส่วนอื่น ๆ ของโปรแกรม โดยไปแก้ที่คลาส Policy เมธอด preprocess ของโปรแกรม

```
17511 0 15375829 13 1 15375829 13 1 15375842
17512 0 15375904 12 1 15375904 12 1 15375916
17500 1 15369480 4156 1 15372308 4156 1 15376464
17515 1 15377149 13 1 15377149 13 1 15377162
17516 1 15377266 11 1 15377266 11 1 15377277
17445 0 15342061 35503 2 15342061 35503 1 15377564
17462 0 15350530 28045 10 15350530 28045 1 15378575
17519 1 15378773 13 1 15378773 13 1 15378786
17483 0 15361733 17851 1 15361733 17851 1 15379584
17524 1 15380619 18 1 15380619 18 1 15380637
```

ภาพประกอบ ก-3 ตัวอย่างเอาพุทของโปรแกรม

ภาคผนวก ข
อภิธานศัพท์

อภิธานศัพท์

B

Backfill: เทคนิคสำหรับการจัดกำหนดการงานโดยอนุญาติให้งานที่มีลำดับความสำคัญต่ำกว่าสามารถใช้ทรัพยากรของระบบได้ ถ้าระบบมีทรัพยากรเหลือพอสำหรับงาน ๆ นั้น โดยงานต้องมีความต้องการหน่วยประมวลผลและระยะเวลาประมวลผลน้อยกว่าหรือเท่ากับทรัพยากรที่เหลืออยู่ ณ เวลานั้น

Best-fit: การเลือกพาร์ทิชันโดยเลือกพาร์ทิชันจากพาร์ทิชันที่มีจำนวนหน่วยประมวลผลเหลือพอดีกับความต้องการของงาน ๆ นั้นมากที่สุด

Bounded Slowdown: ค่าสโลว์ดาวน์ที่ลดผลกระทบของงานที่ใช้ระยะเวลาการประมวลผลน้อยแต่ต้องรอนานโดยการกำหนดค่าระยะเวลาการประมวลผลที่จะใช้เป็นตัวหาร

Branching Heuristic: ลำดับของการจัดเรียงงานในคิวเพื่อสร้างโครงสร้างต้นไม้สำหรับการค้นหาซึ่งขึ้นอยู่กับฮิวริสทิคที่เลือกใช้

C

Conservative Backfill: เทคนิคแบ็กฟิลที่ให้ความสำคัญกับงานที่มีระดับความสำคัญสูงกว่าทุกตัวตามลำดับความสำคัญ

Context switching: การที่หน่วยประมวลผลสลับไปประมวลผลงานอื่น ทำให้ต้องบันทึกสถานะปัจจุบันของงานไว้ในหน่วยความจำ

D

DDS (Depth-Bounded Discrepancy Search): วิธีการค้นหาประเภทหนึ่งที่อาศัยดิสครีแพนซีในการค้นหาเป็นวิธีที่เหมาะกับการค้นหาในพื้นที่การค้นหาขนาดใหญ่

Discrepancy: จุดที่ใช้สำหรับตัดสินใจว่าจะทำการค้นหาลึกลงไปเมื่อเปรียบเทียบกับฮิวริสทิคที่ใช้สำหรับการค้นหา

Dynamic Threshold: ค่าที่กำหนดของระบบโดยจะเปลี่ยนแปลงตามค่าการรอในคิวใช้สำหรับคำนวณค่าการรอที่มากเกินกำหนด (Excessive Wait Time)

E

Easy Backfill: เทคนิคแบ็กฟิลที่ให้ความสำคัญกับงานที่มีระดับความสำคัญสูงกว่างานแรกเพียงงานเดียวเท่านั้น

Equal Size Partition: ประเภทหนึ่งของระบบหลายพาร์ทิชัน โดยที่แต่ละพาร์ทิชันจะมีจำนวนหน่วยประมวลผลเท่ากัน

Excessive Wait Time: เป็นค่าที่ใช้ในการวัดประสิทธิภาพของระบบชนิดหนึ่ง โดยคำนวณจากค่าการรอที่มากเกินกว่าค่าที่ระบบกำหนดไว้

F

FCFS (First Come First Serve): เป็นนโยบายสำหรับจัดกำหนดการงานที่กำหนดลำดับความสำคัญของงานตามลำดับที่งานถูกส่งเข้ามาสู่ระบบ

G

Goal-oriented: เป็นนโยบายสำหรับจัดกำหนดการงานแบบขนานที่สามารถกำหนดลำดับความสำคัญของงานได้เองโดยไม่ต้องให้ผู้ดูแลระบบเป็นผู้กำหนดระดับความสำคัญของงาน

H

Heuristic: การใช้จิตสำนึกในการแก้ปัญหาหรือการเลือก

J

Job-based Priority: การให้ระดับความสำคัญของงานโดยการแบ่งระดับความสำคัญตามงาน

Job Scheduler: ตัวจัดกำหนดการงานในคิว เพื่อจัดลำดับของงานในคิวให้เข้าไปใช้ทรัพยากรของระบบ

Job Scheduling: การจัดกำหนดการงานของระบบคอมพิวเตอร์

L

LDS (Limited Discrepancy Search): วิธีการค้นหาประเภทหนึ่งที่อาศัยดิสครีแพนซีในการค้นหา

LXF (Largest Slowdown First): เป็นนโยบายสำหรับจัดกำหนดการงานที่กำหนดลำดับความสำคัญของงานตามลำดับจากค่าสโลว์ดาวน์จากมากไปหาน้อย

M

Multi-partition: ระบบหลายพาร์ทิชัน ซึ่งเป็นระบบประเภทหนึ่งในระบบคอมพิวเตอร์แบบขนาน

N

Non-equal Size Partition: ประเภทหนึ่งของระบบหลายพาร์ทิชัน โดยที่แต่ละพาร์ทิชันจะมีจำนวนหน่วยประมวลผลไม่เท่ากัน

Non-preemptive: ชนิดของนโยบายสำหรับจัดกำหนดการงานที่จะให้สิทธิ์กับงานที่กำลังประมวลผลอยู่จะได้สิทธิ์ในการประมวลผลจนกว่าจะเสร็จ

P

PCB (Process Control Block): บล็อกที่เก็บสถานะการประมวลผลของงาน

Policy: นโยบายสำหรับตัวจัดกำหนดการงานใช้ในการจัดลำดับงาน

Q

Queue-based Priority: การให้ระดับความสำคัญของงานโดยการแบ่งระดับความสำคัญตามคิว

R

Reservation: สิทธิ์ในการใช้ทรัพยากรของระบบ

Runtime: ระยะเวลาที่ใช้ในการประมวลผลของงาน

S

Simplified: รูปแบบของเวิร์คโหลดไฟล์มีทั้งหมด 6 คอลัมน์

Simulator: โปรแกรมจำลองตัวจัดกำหนดการงานแบบขนานบนระบบคอมพิวเตอร์แบบขนาน

SJF (Short Job First): เป็นนโยบายสำหรับจัดกำหนดการงานที่กำหนดลำดับความสำคัญของงานตามลำดับจากจำนวนความต้องการหน่วยประมวลผลของงานจากน้อยไปหามาก

Slowdown: ค่าการรอของงานที่นำค่าระยะเวลาประมวลผลของงานมาคำนวณด้วย

Starvation: ปัญหาการอดอยากคือการที่งานต้องอยู่ในคิวนานมากแล้วยังไม่ได้สิทธิ์ให้ใช้ทรัพยากรของระบบ

SWF (Standard Workload Format): รูปแบบของเวิร์คโหลดไฟล์มีทั้งหมด 18 คอลัมน์

U

User Estimate Runtime: ระยะเวลาที่ใช้ในการประมวลผลของงานซึ่งถูกประมาณค่าโดยผู้ใช้

Utilization: การใช้ประโยชน์ของระบบ

ภาคผนวก ค
ผลงานตีพิมพ์เผยแพร่จากวิทยานิพนธ์

The 7th International Joint Conference on Computer Science and Software Engineering

(JCSSE2010)

May 12-14, 2010, at Ramkhamhaeng University, Bangkok, Thailand.

Department of Computer Science
Faculty of Science
Ramkhamhaeng University
Bangkok, Thailand
ISBN 978-616-513-223-7

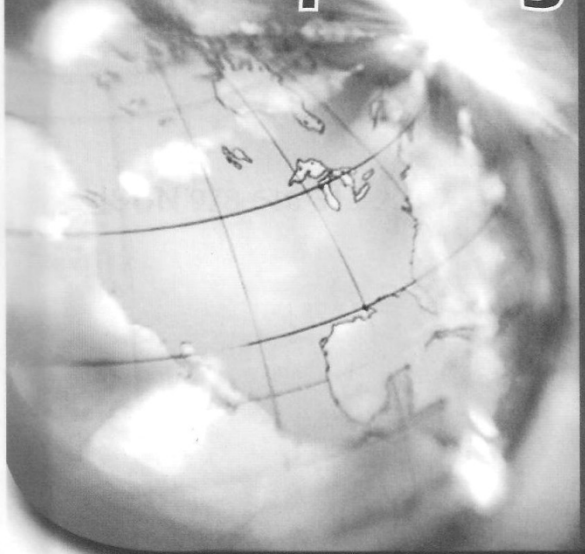
JCSSE 2010

Vol.1

"Computing for Change"

Proceedings of the 7th International Joint Conference on Computer Science and Software Engineering

May 12-14, 2010 Bangkok, Thailand



Impact of Multi-partition Systems on Goal-oriented Parallel Computer Job Scheduling Policies

Amonrat Prasitsupparote Sangsuree Vasupongayya*
Department of Computer Engineering
Prince of Songkla University
Hat Yai, Songkhla, 90110, Thailand
*e-mail: vsangsur@coe.psu.ac.th

Abstract

This work is focusing on applying goal-oriented parallel computer job scheduling policies on multi-partition parallel computer systems since several existing systems partition their system resources. Goal-oriented parallel computer job scheduling policies utilizes a combinatorial search technique to find the best schedule within a time limit. Two implementation ideas of multi-partition goal-oriented parallel computer job scheduling policies are proposed and evaluated using an event-driven simulator. The first implementation idea includes the partition selection question into the search space while the second idea selects a partition according to some heuristics resulting in a smaller search space. The experimental results show that the second idea with a best-fit partition selection heuristic performs well under both equal-size and non-equal-size two-partition systems. Furthermore, the proposed policy also outperforms the two basic priority backfill policies.

Keywords: Multi-partition, job scheduling, goal-oriented, backfill, best-fit

1. Introduction

Parallel computer systems have recently increased their impacts on many applications such as drug designs, weather predictions, disaster predictions, scientific research simulations and modeling. A typical production parallel computer system can simultaneously service many users. To run a parallel computer job, a user usually submits his/her job to the job scheduler along with his/her job requirements such as the required amount of memory, the required amount of processors and the job estimated runtime. Most production parallel computer systems are typically non-preemptive. Under such schemes, a job will not be interrupted once it is executed. The scheduler is activated upon each job arrival and departure to select a set of waiting jobs for executions according to a pre-defined scheduling policy. All jobs in the system will eventually be executed unless they are canceled by their owner.

Typical non-preemptive parallel computer job scheduling policies are priority based backfilling policies. This set of policies considers jobs for executions according to some priorities. The priority is usually a linear combination of some job measures defined by the system administrator. For example, the job wait time may be used to allow the older jobs to have a higher priority than the younger jobs. The job slowdown may be used to allow the shorter jobs to have a higher priority than the longer jobs. A linear combination is usually a weighted function.

In many situations, however, a set of required objectives can be conflicting with each other. To prevent starvation, for example, difficult jobs (i.e., large jobs, long jobs and large-and-long jobs) must have a high priority because these jobs are likely to be delayed. To minimize average wait time, on the other hand, small and short jobs must have a high priority because majority of jobs are in this group. As can be seen, these two objectives are conflicting with each other. Most priority based scheduling policies can achieve one or the other but not both.

Goal-oriented parallel computer job scheduling policies [1] were proposed recently to reduce the system administrator tasks of adjusting and tuning low-level scheduling parameters (e.g., a set of weighted values) for performances. The goal-oriented policy employs a discrepancy-based complete search technique [2] to find a 'good' solution in a limited time according to a given set of objectives. It has been shown that the goal-oriented parallel job scheduling policies are good at finding a schedule that compromises a set of conflicting objectives [3, 4].

In this work, the goal-oriented parallel computer job scheduling policies are further evaluated under multi-partition parallel computer systems which is still lacking in previous works. Under a single partition system, all processors in the system are available

to the user. Thus, a job can be as large as the entire system. And, the scheduler only question is whether this job should be scheduled now. Under a multi-partition system, however, the processors are partitioned and a job must fit into one partition. Thus, the job can only be as large as the capacity of one partition. Therefore, a multi-partition job scheduler must answer an additional question that is on which partition this job should be scheduled.

Since several parallel computer systems partition their system resources and the original goal-oriented parallel computer job scheduling polices have not been tested on such systems, this work is focusing on modifying the original goal-oriented parallel computer job scheduling policies to handle multi-partition parallel computer systems. Both an equal-size and a non-equal size partition system are evaluated.

The remaining of this paper is organized as follow. Section 2 describes both the original and the proposed goal-oriented policies. Section 3 explains the methodologies for evaluating the proposed policies. Section 4 gives results and discussions while conclusions are given in Section 5.

2. Goal-oriented parallel job scheduling

The original goal-oriented policies are described in Section 2.1 while the proposed multi-partition goal-oriented policies are described in Section 2.2.

2.1 Single-partition goal-oriented policies

The idea of a goal-oriented parallel computer job scheduling is to reduce the difficulty of operating a parallel computer job scheduler. Instead of tuning low-level scheduling parameters for performances, the goal-oriented parallel computer job scheduler allows the system administrator to define a set of high-level scheduling objectives such as preventing starvations and improving average wait times. Then, the scheduler automatically searches the space of all possible schedules to find a 'good' schedule according to the objectives within a time limit.

To achieve this, all candidate schedules at each scheduling decision point are organized into a search tree according to some branching heuristic (i.e., the order of the jobs from left to right). Figure 1 shows an example search tree of four waiting jobs (numbered 1 to 4 in their arriving order) using the FCFS (i.e., first-come-first-served) branching heuristic. In the tree, each node represents a job while each path from the root to a leaf (e.g., 1-2-3-4) represents an order. The scheduler evaluates jobs for scheduling in the given order of each path. At each node, the earliest time that enough resources will be available to execute the current job is computed (according to the resources allocated to currently executing jobs and the waiting jobs considered prior to the current job). Note that the order of jobs considered for scheduling is not necessarily the same as the order the jobs started.

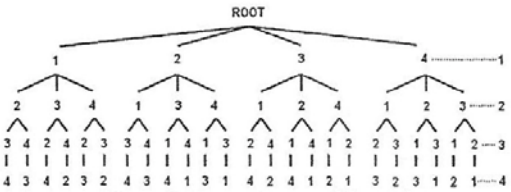


Figure 1. A search tree of four jobs

Once reaching a leaf node (i.e., search the whole path) the objective score is calculated and compared with the score of the best solution found so far. If the current solution is 'better' than the best solution found so far, the current solution is kept as the best solution. Otherwise, the current solution is discarded. The searching process repeats until the time limit is reached. The time limit can be described as the number of nodes visited or the actual search time. To find a good schedule, the search technique [2] employed by the goal-oriented policy is designed to search various different portions of the space of all possible schedules. Furthermore, allowing more search time will guarantee to always find an equally good or a better solution.

2.2 Multi-partition goal-oriented policies

Under a single partition system, the scheduler only needs to find out whether this job should be scheduled now. Under a multi-partition system, however, the scheduler must also answer on which partition this job should be scheduled. In this work, two implementation ideas of the multi-partition goal-oriented scheduling policy are proposed.

The first idea is to build a large search tree. Under this scheme, each node in the tree will have two pieces of information that is (1) which partition should this job be assigned and (2) when this job should start on the assigned partition. Figure 2 shows a partial search tree of four waiting jobs on a two-partition parallel computer system. The number of nodes in the tree increases linearly with the number of partitions. That is, the number of nodes on a two-partition system is twice as much as those on a single-partition system. For example, at the 1st-level there are 8 nodes (i.e., 1:1, 1:2, 2:1, 2:2, 3:1, 3:2, 4:1, 4:2) instead of 4 nodes for a 4-job search tree. At each node, the first number indicates the job identification number while the second number indicates the partition number that the job will be scheduled on. The doted lines on the right-hand side indicate the depth of the tree. At depth 1, all nodes (i.e., 8 nodes)

are shown. At depth 2-4, only the first subtree (i.e., the left-most subtree at each depth) is shown. As seen in the figure, the FCFS branching heuristic is used to order the jobs from left to right. And, the 1st partition always comes before the 2nd partition.

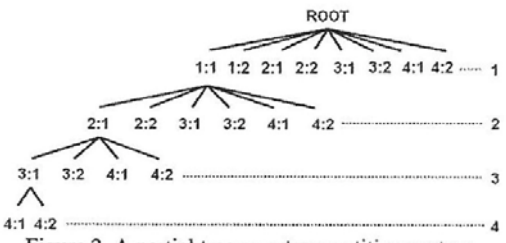


Figure 2. A partial tree on a two partition system

The second implementation idea aims to keep the search space small by first selecting the partition for each job using some heuristic. Thus, the search space is the same size as that of the original goal-oriented policies on a single partition system.

3. Experimental design

Two implementation ideas of the multi-partition goal-oriented parallel computer job scheduling policies discussed in Section 2.2 are evaluated using an event-driven simulator. The workload used in the previous works [1,3,4] is also used as input to the simulator. However, the workload is collected from a single partition system. Therefore, the workload is modified to be a two-partition system. The modification details are given in Section 3.1. The multi-partition goal-oriented policies are evaluated against two backfill policies whose details are given in Section 3.2. Section 3.3 provides performance measures and simulator settings.

3.1 Workload

Two multi-partition systems are tested in this work. One is an equal size partition each of which contains 64 computational nodes. Another one is a non-equal size partition i.e., the 1st partition contains 32 computational nodes while the 2nd partition contains 64 computation nodes. However, the original workload (i.e., IA-64) was running on a single partition of 128 dual-processor node system. Since each job has to be executed completely on one partition, it cannot request more than 64 computational nodes. Therefore, the original number of requested nodes of each job is modified to be the minimum between 64 and the original number of requested nodes.

Table 1 shows the characteristic of the modified workload including demand (Proc. demand), number of users (#users), number of jobs (#jobs) and job size (i.e., NT: node-hour) which is given in three measures (i.e., average (Avg.), median (Med.), and maximum (Max.)).

Table 1. Information of the modified IA-64 workload

Month	Proc. demand	#users	#jobs	Job size (NT)		
				Avg.	Med.	Max.
6/03	80%	73	2191	33.78	0.8	768.0
7/03	74%	68	1400	50.12	1.1	768.0
8/03	70%	73	3221	20.61	0.0	768.0
9/03	67%	74	3057	20.08	0.1	768.0
10/03	70%	75	4149	15.96	0.3	768.0
11/03	68%	81	3443	18.13	0.7	768.0
12/03	73%	61	3521	19.73	1.0	1151.0
1/04	70%	53	3156	21.11	5.1	1536.0
2/04	71%	73	3969	15.97	0.3	1536.0
3/04	72%	70	3466	19.77	0.0	1536.0

3.2 Policies

The most widely used non-preemptive parallel computer job scheduling policies are priority scheduling policies with backfilling techniques. Backfilling techniques allows lower priority jobs to start on idle resources as long as their executions do not delay the scheduled start time of the oldest waiting job [5]. In this work, the proposed multi-partition goal-oriented parallel computer job scheduling policies are evaluated against Best-fit-First-Come-First-Sever-backfill (i.e., FCFS-backfill) and Best-fit-Largest-Slowdown-First-backfill (i.e., LXF-backfill) policies.

The idea of the two backfill policies is similar to their single partition counterparts. The only difference is that each job will be assigned a partition using a best fit heuristic. That is, the waiting jobs are ordered using an appropriated priority value (either submit time for FCFS or slowdown for LXF). The best fit heuristic selects the partition which is best fit the considered job. That is, when the job can start on both partitions, the job will be assigned the partition with the least number of available nodes that is large enough to execute the job. For example, a two-partition system with 10 available computational nodes on the 1st partition and 8 available computational nodes on the 2nd partition. When a 5-node job arrives, it will be assigned to the 2nd partition because the job is best fitted in the 2nd partition.

Two versions of the multi-partition goal-oriented job scheduling policies (i.e., Tradeoff(Tw:avgX)) described in Section 2 are evaluated. The partition selection heuristic of the second implementation idea is best-fit similar to that of both backfill policies. To limit the search time during each scheduling decision, a number of node limit is used as a stopping criteria. In this study, the number of node limit is 4000 nodes which take only a few ten milliseconds.

3.3 Performance measures and simulator settings

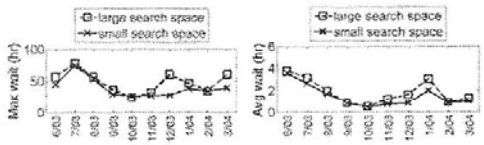
Several scheduling performance measures widely used in the field [6,7,8,9,10] are studied. The monthly performances are measured from the jobs submitted during the month. To be realistic, the simulator is loading with jobs submitting during the last week of the previous month and the jobs from the next month continue to arrive until the jobs from the measured month are all started. To study the full potential of the proposed policies, the scheduler uses actual job runtime information (i.e., accurate runtime information) to make scheduling decisions however the scheduler does not have the job arrival information ahead of time (i.e., online setting). The impact of user runtime estimates are left for future works.

4. Results and discussions

First, the scheduling performances of the two proposed multi-partition goal-oriented parallel computer job scheduling policies are presented in Section 4.1. Second, the performances of the multi-partition goal-oriented parallel computer job scheduling policies are compared against those of the two priority backfilling policies in Section 4.2. Section 4.3 shows the performances of all policies under a non-equal-size two-partition parallel computer system.

4.1 Impact of the search space size

Since the two multi-partition goal-oriented policy implementation ideas proposed in this work result in different size of the search space, this section investigates the impact of the search space size. Figure 3 presents the overall scheduling performances of the two ideas on an equal-size two-partition system each of which contains 64 computational nodes. Only the maximum and the average wait measures are presented because the average bounded slowdown performance is similar to that of the average wait.



(a) Maximum wait (b) Average wait

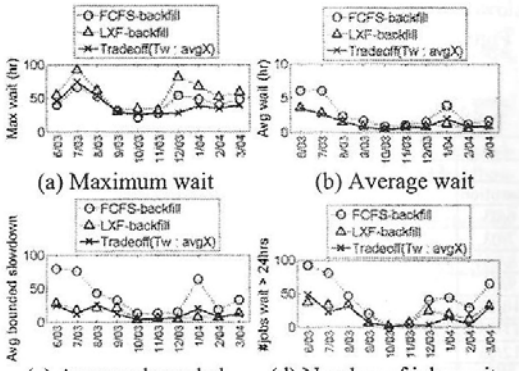
Figure 3 Performance of goal-oriented policies

According to Figure 3, the second implementation idea (i.e., small search space) outperforms the first idea on all measures on all months studied. Even though the performance differences on most months are small, the performance differences are clearly shown on 12/03 and 1/04 months. This is caused by the number of waiting jobs which is directly related to the size of the search space. The search space is double under the first idea causing the searching process to stop (i.e., exceeding the node limit) before reaching a good solution.

4.2 Goal-oriented versus priority backfill policies

In this section, the performances of the second idea of the multi-partition goal-oriented policy are compared with those of the two backfill policies. Figure 4(a)-(d) shows the maximum wait, the average wait, the average bounded slowdown and the number of jobs waiting longer than 24 hours under Best-fit-FCFS-backfill (denoted FCFS-backfill), Best-fit-LXF-backfill (denoted LXF-backfill) and the goal-oriented policy (denoted Tradeoff(Tw:avgX)) under an equal-size two-partition system.

As expected, the FCFS-backfill policy provides good maximum wait performances and poor average performances (i.e., both average wait and average bounded slowdown). The LXF-backfill policy has totally opposite performances. Thus, the results confirm that priority backfill policies still cannot achieve good performances on all three measures simultaneously on all months studied. The goal-oriented policy however can achieve the best or close to the best performances on all measures even when it is operated on an equal-size two-partition system.



(a) Maximum wait (b) Average wait

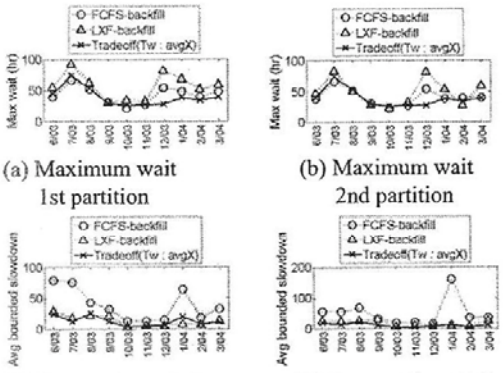
(c) Average bounded slowdown (d) Number of jobs wait longer than 24 hours

Figure 4 Performances of priority backfill versus those of the proposed goal-oriented policies

To analyze the detail performances of each policy, Figure 5 shows the performances of the jobs executed on each partition during the measured month. The jobs are assigned the partition according to the best-fit heuristic. Thus, each policy will result in different number of jobs. Table 2 shows the characteristics of jobs executed on each partition under each policy.

According to Figure 5, the performance trends on each partition are similar to the trends observed on

the overall graphs (Figure 4). That is, the multi-partition goal-oriented policy still produces good scheduling performances on each partition. Note that the average bounded slowdown on the 2nd partition can be higher than that of the overall graph because the values are calculated from a different set of jobs. That is, the value shows in Figure 5(d) is calculated from a subset of jobs used in Figure 4(b) because only the performances of jobs executed on the 2nd partition are used in Figure 5(d) while the performances of all jobs of the month are used in Figure 4(b).



(a) Maximum wait 1st partition (b) Maximum wait 2nd partition

(c) Average bounded slowdown 1st partition (d) Average bounded slowdown 2nd partition

Figure 5 Performances of jobs on each partition under an equal two-partition system

Table 2. Information of jobs on each partition

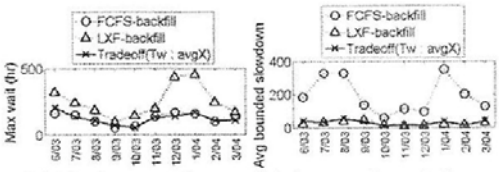
Number of jobs						
	FCFS-backfill		LXF-backfill		goal-oriented	
partition	1st	2nd	1st	2nd	1st	2nd
6/03	1268	923	1078	1113	1281	910
7/03	879	521	905	495	795	605
8/03	2034	1187	1677	1544	2199	1022
9/03	1921	1136	2143	914	1607	1450
10/03	2801	1348	3044	1105	1935	2214
11/03	2545	898	2409	1034	1635	1808
12/03	2441	1080	2405	1116	1581	1940
1/04	2241	915	2133	1023	1692	1464
2/04	2810	1159	2689	1280	1902	2067
3/04	2409	1057	2399	1067	1541	1925
Average job size (NT) in node-hour						
	FCFS-backfill		LXF-backfill		goal-oriented	
partition	1st	2nd	1st	2nd	1st	2nd
6/03	29.3	39.9	34.4	33.1	29.1	40.3
7/03	39.2	68.4	38.4	71.5	42.9	59.4
8/03	15.9	28.5	19.8	21.3	14.9	32.7
9/03	16.3	26.3	14.0	34.3	18.6	21.6
10/03	12.4	23.1	10.9	29.8	17.7	14.3
11/03	12.7	33.3	13.7	28.2	18.4	17.8
12/03	14.6	31.2	14.5	30.9	21.5	18.2
1/04	16.8	31.4	17.1	29.2	21.4	20.6
2/04	11.4	26.8	11.8	24.5	16.3	15.5
3/04	14.5	31.6	14.8	30.9	21.6	18.2

According to Table 2, the two backfill policies usually arrange more jobs in the 1st partition while the goal-oriented policy tends to evenly distribute jobs to both partitions. The average job size in node-hour information shows that the two backfill policies send large jobs to the 2nd partition while the small jobs are mostly executed on the 1st partition. The goal-oriented policy, however, sends similar jobs to both partitions. Since there are equal resources available on both partitions (each with 64 computational nodes), evenly distributing jobs to each partition reduces the chance of letting a few jobs wait for a long time resulting in good maximum wait time performances. The average wait time (Figure 4(b)) and the number of jobs waiting longer than 24 hour information (Figure 4(d)) demonstrate that the goal-oriented policy achieves good maximum wait performance and also provides good average performances.

4.3 Impact of non-equal size partitions

Results reported in the previous two sections are on an equal-size two-partition system each of which provides 64 computational nodes. To study the impact of non-equal-size systems, all policies are evaluated under a two-partition system where the 1st partition contains 32 computational nodes and the 2nd partition contains 64 computational nodes.

Figure 6 shows the performances under a non-equal-size two-partition system of both backfill policies and the 2nd idea of the multi-partition goal-oriented policy. Figure 6(a) shows the maximum wait performance while Figure 6(b) shows the average bounded slowdown performances. The average wait performances (not show) have similar trends with those of the average bounded slowdown.



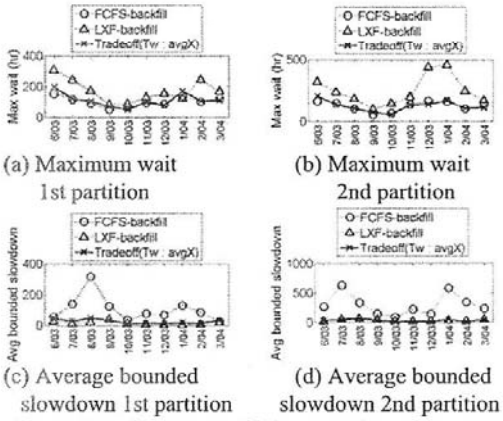
(a) Maximum wait (b) Average bounded slowdown

Figure 6 Performances under a non-equal size system

The results observed here are similar to those under an equal-size partition system. That is, the FCFS-backfill policy achieves good maximum wait performances on all months while the LXF-backfill policy achieves good average bounded slowdown performances on all months. The goal-oriented policy achieves the best or close to the best on both measures under a non-equal-size two-partition system.

Figure 7 shows the maximum wait and average

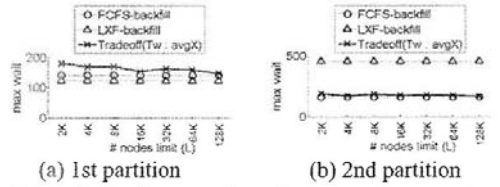
bounded slowdown performances of the jobs executed on each partition. The trends of the average bounded slowdown performances on both partitions are similar to that of the overall graph (see Figure 6(b)). The trend of the maximum wait performance of the jobs executed on the 2nd partition (i.e., the 64 computational node partition) is also similar to that of the overall graph (Figure 6(a)). However, the trend of the maximum wait performance of the 1st partition (i.e., the 32 computational node partition) is slightly different from that observed in the overall graph and in the 2nd partition, especially in January 2004 month. That is, the performances of all three policies on the 1st partition are similar on January 2004 month while the overall graph (Figure 6(a)) and the 2nd partition graph (Figure 7(b)) clearly show that the FCFS-backfill policy produces a significantly worse performance than the other two policies.



(a) Maximum wait 1st partition (b) Maximum wait 2nd partition

(c) Average bounded slowdown 1st partition (d) Average bounded slowdown 2nd partition

Figure 7 Performances of jobs on each partition under a non-equal two-partition system.



(a) 1st partition (b) 2nd partition

Figure 8 Maximum wait performances versus node limit on January 2004

Figure 8 shows the maximum wait performance of the goal-oriented policy on January 2004 workload on a non-equal-size two-partition system when varying the node limit from 2000 to 128000 nodes. The performances of both backfill policies are given for comparison purposes. Figure 8(a) clearly shows that the goal-oriented policy produces a smaller maximum wait value (i.e., a better solution) as the node limit increases (i.e., longer search time). While, Figure 8(b) shows that the better performance on the 1st partition does not degrade the performance of the 2nd partition.

5. Conclusions

Two implementation ideas of multi-partition goal-oriented parallel job scheduling policy are proposed in this work. The first implementation idea enlarges the search space by including the partition selection into the search tree. The second implementation idea keeps the same search space as that in the original goal-oriented policy by first assigning the partition according to some heuristic. The experimental results show that applying a simple best-fit partition selection heuristic to the original goal-oriented policy (i.e., the second implementation idea) produces the best or closed to the best performances in comparison with the two priority backfill policies. The results are observed on both equal-size and non-equal-size two-partition systems. Furthermore, allowing longer search time will result in an equally good or a better performance.

6. References

[1] S.-H. Chiang & S. Vasupongayya, "Design and potential performance of goal-oriented job scheduling policies for parallel computer workloads". IEEE Tran. on Parallel and Distributed Systems. 19(12):1642-1656, 2008.

[2] T. Walsh, "Depth-bounded discrepancy search", proc. of IJCAI, 1997.

[3] S. Vasupongayya, "Goal-oriented parallel job scheduling: A revisit", Proc. of the 2nd UBU-Research, Ubonratchathani, Thailand, July 2008.

[4] S. Vasupongayya, "Achieving fair share objectives via goal-oriented parallel computer job scheduling policies", Proc. WASET ICCSE'09, Bangkok, Thailand, December 25-27, 2009.

[5] D. Lifka. The ANL/IBM SP scheduling system. In Workshop on JSSPP, 1995.

[6] S.-H. Chiang, A. Arpaci-Dusseau & M. Vernon, "The impact of more accurate request runtimes on production job scheduling performance". In Lecture Notes in Computer Science (2537):103-127, 2002.

[7] S.-H. Chiang & C. Fu, "Benefit of limited time-sharing in the presence of very large parallel jobs". Proc. of IEEE IPDPS, 2005.

[8] S.-H. Chiang & M. Vernon, "Production job scheduling for parallel shared memory systems". Proc. of IEEE IPDPS, 2001.

[9] D. Talby & D. Feitelson, "Supporting priorities and improving utilization of the IBM SP2 scheduler using slack-based backfilling". Proc. of IPPS. 1999.

[10] D. Talby & D. Feitelson, "Improving and stabilizing parallel computer performance using adaptive backfilling". Proc. of IEEE IPDPS, 2005.

ประวัติผู้เขียน

ชื่อ สกุล นางสาวอมรรัตน์ ประสิทธิ์ศุภโรจน์

รหัสประจำตัวนักศึกษา 5110120058

วุฒิการศึกษา

วุฒิ	ชื่อสถาบัน	ปีที่สำเร็จการศึกษา
วิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)	มหาวิทยาลัยทักษิณ	2548

ทุนการศึกษา (ที่ได้รับในระหว่างการศึกษา)

- ทุนผู้ช่วยวิจัย คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์ ประจำปีการศึกษา 2551
- ทุนผู้ช่วยสอน มหาวิทยาลัยสงขลานครินทร์ ประจำปีการศึกษา 2553

การตีพิมพ์เผยแพร่ผลงาน

- Amonrat Prasitsupparote and Sangsuree Vasupongayya, "Impact of Multi-partition Systems on Goal-oriented Parallel Computer Job Scheduling Policies," In Proceedings of 7th International Joint Conference on Computer Science and Software Engineering (JCSSE2010), Bangkok, Thailand, 12th – 14th May 2010, pages 232-237.