



รายงานการวิจัย

การพัฒนาอัลกอริทึมการค้นหาแบบตาบู่เชิงปรับตัวหลายวิถีและ
การใช้งานในปัญหาการหาค่าเหมาะที่สุด
(Development of the Multipath Adaptive Tabu Search (MATS)
Algorithms and Applications in Optimization Problems)

คณะผู้วิจัย

หัวหน้าโครงการ

ศาสตราจารย์ นาวาอากาศโท ดร.สรารุฒิ สุจิตจร

สาขาวิชาวิศวกรรมไฟฟ้า

สำนักวิชาวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีสุรนารี

ได้รับทุนอุดหนุนการวิจัยจากมหาวิทยาลัยเทคโนโลยีสุรนารี ปีงบประมาณ พ.ศ. 2552

ผลงานวิจัยเป็นความรับผิดชอบของหัวหน้าโครงการวิจัยแต่เพียงผู้เดียว

พฤษภาคม 2553

กิตติกรรมประกาศ

ขอขอบคุณ คุณจักรกฤษณ์ เกลือบวัง เป็นอย่างยิ่งในความช่วยเหลือด้านต่าง ๆ และ
ขอขอบคุณมหาวิทยาลัยเทคโนโลยีสุรนารี สำหรับการสนับสนุนงบประมาณการวิจัย

ผู้วิจัย

พฤษภาคม 2553

บทคัดย่อ

งานวิจัยนี้นำเสนอการพัฒนาอัลกอริธึมการค้นหาแบบตามเชิงปรับตัวแบบเดิมให้เป็นอัลกอริธึมค้นหาหลายเส้นทาง อาศัยวิธีบริหารจัดการหน่วยค้นหาให้ดำเนินงานได้อย่างมีประสิทธิภาพบนคอมพิวเตอร์แบบหน่วยประมวลผลเดี่ยว วิธีบริหารจัดการดังกล่าวประกอบด้วยกลไกหลัก 3 ประการ ได้แก่ กลไกแบ่งปริภูมิ (partitioning mechanism) ทำหน้าที่แบ่งปริภูมิการค้นหาทั้งหมดออกเป็นปริภูมิต่อๆ แล้วคัดเลือกคำตอบเริ่มต้นภายในปริภูมิต่อนั้น เพื่อจัดให้กับหน่วยค้นหาต่าง ๆ กลไกลำดับการค้นหา (sequencing mechanism) ทำหน้าที่กำกับให้หน่วยค้นหาดำเนินงานตามหลักการแบ่งเวลา และกลไกยกเลิกการค้นหา (discarding mechanism) ทำหน้าที่กำจัดหน่วยค้นหาที่ด้อยคุณภาพออกจากวงรอบการค้นหา เพื่อให้เวลาการค้นหารวมลดลง การทดสอบสมรรถนะของอัลกอริธึมที่นำเสนอ ดำเนินการกับปัญหาการหาค่าเหมาะที่สุดเชิงฟังก์ชันสามชนิด ได้แก่ ฟังก์ชัน โบฮาเชฟสกี (Bohachevsky's function) ฟังก์ชัน ราสทริกิน (Rastrigin's function) และฟังก์ชันชีเคิลฟอกซ์โฮล (Shekel's foxholes function) จากผลการทดสอบเฉลี่ยจากจำนวน 50 ครั้ง ชี้ชัดว่าอัลกอริธึมใหม่ที่นำเสนอสามารถค้นพบคำตอบวงกว้างได้รวดเร็วกว่าอัลกอริธึม ATS แบบเดิม สามารถประหยัดเวลาการค้นหาโดยเฉลี่ยอยู่ที่ประมาณ 26.26% อัลกอริธึมที่พัฒนาขึ้นได้รับการนำไปประยุกต์อย่างประสบความสำเร็จกับปัญหาการหาค่าเหมาะที่สุดทางวิศวกรรมด้านการระบุเอกลักษณ์ระบบพลวัตสำหรับระบบเพนดูลัมผวนกรวด

Abstract

The research proposes the improvement made to the conventional adaptive tabu search (ATS) to achieve the multipath ATS or MATS. The approach used is to incorporate some management techniques to the ATS, and the new MATS can be effectively run on a single CPU platform. The said management composes of 3 mechanisms namely partitioning mechanism (PM), sequencing mechanism (SM) and discarding mechanism (DM). The PM subdivides the entire search space into a number of sub-search-spaces, and assigns initial solutions to the ATS paths belonging to those sub-search-spaces. The SM organizes all the paths to run on the time-sharing basis. The DM identifies and discards the low-quality search paths such that the overall search time could be decreased. The performance of the proposed MATS is assessed against 3 surface optimization problems including Bohachevsky's, Rastrigin's and Shekel's foxholes functions. As an average result obtained from 50 trials, the new algorithm performs faster than the conventional ATS does in that the average search-time savings are about 26.26%. The developed algorithm has been successfully applied to an engineering optimization problem namely the dynamical system identification of the cart-plus-pendulum system.

สารบัญ

หน้า

กิตติกรรมประกาศ.....	ก
บทคัดย่อภาษาไทย.....	ข
บทคัดย่อภาษาอังกฤษ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญภาพ.....	ช
บทที่ 1 บทนำ.....	1
1.1 ความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	1
1.3 ขอบเขตของการวิจัย.....	1
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.5 ข้อตกลงเบื้องต้นของการวิจัย.....	2
บทที่ 2 ปรัชญ่วรรณกรรมและอัลกอริธึมการค้นหาแบบตามูเชิงปรับตัว.....	3
2.1 กล่าวนำ.....	3
2.2 ปรัชญ่วรรณกรรมที่เกี่ยวข้องกับการค้นหาแบบตามู.....	3
2.3 อัลกอริธึมการค้นหาแบบตามูเชิงปรับตัว.....	9
2.4 ผลและอภิปราย.....	12
บทที่ 3 อัลกอริธึมการค้นหาแบบตามูเชิงปรับตัวหลายวิถี.....	15
3.1 กล่าวนำ.....	15
3.2 การค้นหาแบบตามูเชิงปรับตัวหลายวิถี (MATS).....	15
3.2.1 กลไกแบ่งปริภูมิการค้นหา (PM).....	17
3.2.2 กลไกลำดับการค้นหา (SM).....	20
3.2.3 กลไกยกเลิกการค้นหา (DM).....	22
3.3 การประเมินสมรรถนะของอัลกอริธึม.....	26
3.3.1 การตั้งค่าพารามิเตอร์ของ MATS.....	26
3.3.2 ผลและอภิปรายผล.....	31
3.3.3 กราฟการลู่เข้าหาคำตอบวงกว้าง.....	35

สารบัญ(ต่อ)

หน้า

3.4 สรุป.....	36
บทที่ 4 การประยุกต์ด้านระบบเอกลักษณ์ระบบพลวัต	37
4.1 กล่าวนำ.....	37
4.2 ระบบ CPP.....	37
4.3 การระบุเอกลักษณ์ระบบ CPP.....	40
4.3.1 การเตรียมข้อมูลจากผลการทดลองจริง.....	41
4.3.2 การระบุเอกลักษณ์ระบบพลวัตของ CPP ด้วย MATS.....	41
4.3.3 ผลการระบุเอกลักษณ์ระบบ CPP จากการค้นหาด้วย MATS 15 เส้นทาง	45
บทที่ 5 สรุป.....	50
5.1 สรุปผลการวิจัย.....	50
5.2 ข้อเสนอแนะ	50
เอกสารอ้างอิง	52
ภาคผนวก ก.	55
ประวัติผู้วิจัย	56

สารบัญตาราง

หน้า

ตารางที่ 2.1 ฟังก์ชันพื้นผิวที่ใช้ทดสอบอัลกอริทึม	12
ตารางที่ 2.2 พารามิเตอร์ของ ATS ที่ใช้ทดสอบ	13
ตารางที่ 2.3 ผลการทดสอบสมรรถนะของการค้นหาแบบตามเชิงปรับตัวที่เสนอ	14
ตารางที่ 3.1 พารามิเตอร์ของ MATS ภาค PM ที่ใช้ทดสอบ	28
ตารางที่ 3.2 พารามิเตอร์ของ DM ใน MATS ที่ใช้ทดสอบ	29
ตารางที่ 3.3 เวลาการค้นหาเฉลี่ยของ ATS และ MATS.....	31
ตารางที่ 3.4 จำนวนรอบการค้นหาเฉลี่ยของ ATSแบบเดี่ยว และ MATS.....	32
ตารางที่ 3.5 เปอร์เซ็นต์อัตราส่วนความเร็วปรับปรุง	32
ตารางที่ 3.6 ประสิทธิภาพการจำนวนรอบการค้นหาเฉลี่ยของ ATSแบบเดี่ยว และ MATS	32
ตารางที่ 4.1 คำตอบแทนและปริภูมิการค้นหาของ MATS#15	44
ตารางที่ 4.2 ตัวอย่างปริภูมิการค้นหาย่อยของ ATS#1 #2 #4 #14 และ #15.....	44
ตารางที่ 4.3 การตั้งค่าพารามิเตอร์ DM	45
ตารางที่ 4.4 คำตอบที่ได้จากการค้นหาด้วย MATS#15	46

สารบัญภาพ

หน้า

รูปที่ 2.1	อัลกอริทึมการค้นหา (ก) แบบตามคู่ดั้งเดิม (ข) แบบตามเชิงปรับตัวที่เสนอ (ค) กลไกย้อนรอยการค้นหา และ (ง) กลไกปรับรัศมีการค้นหา.....	10
รูปที่ 2.2	เส้นทางการค้นหาและสองกลไกหลักของการค้นหาแบบตามเชิงปรับตัวที่เสนอ	11
รูปที่ 3.1	ขั้นตอนการทำงานของ MATS.....	16
รูปที่ 3.2	ขั้นตอนการทำงานของ PM.....	17
รูปที่ 3.3	ตัวอย่างการแบ่งปริภูมิการค้นหาและการตั้งชื่อเส้นทางการค้นหา (ก) สองเส้นทาง (ข) สี่เส้นทาง (ค) แปรคเส้นทาง และ(ง) สิบหกเส้นทาง	18
รูปที่ 3.4	การแบ่งปริภูมิการค้นหาเป็นส่วนหนึ่งของฟังก์ชันซีเคลฟอกซ์ไฮด (ก) แผนที่เส้นระดับสองมิติ และ (ข) แนวการแบ่งปริภูมีย่อยบนพื้นผิว.....	19
รูปที่ 3.5	ลักษณะการทำงานของ SM (ก) ทำงานแบบแบ่งเวลาในวงรอบการค้นหาของ MATS และ(ข) ขั้นตอนการทำงานของ SM.....	21
รูปที่ 3.6	ตัวอย่างการทำงานของ SM ใน MATSแบบ 4 เส้นทางค้นหาที่ยังไม่มี DM (ก) จังหวะเวลา (timing) ของการเกิดคำตอบทั้งหมดของ MATS และ (ข) ลำดับคำตอบของแต่ละ ATSy่อย บนปริภูมิการค้นหา.....	21
รูปที่ 3.7	ขั้นตอนการทำงานของกลไกยกเลิกการค้นหา.....	23
รูปที่ 3.8	ตัวอย่างการทำงานของ SM ใน MATS แบบ 4 เส้นทางค้นหา (ก) เส้นทางการค้นหาตามพิภคของลำดับคำตอบแต่ละ ATSy่อย และ (ข) ลำดับคำตอบของแต่ละ ATSy่อย.....	24
รูปที่ 3.9	ตัวอย่าง PM พร้อมการตั้งชื่อหน่วยค้นหา (ก) 32 เส้นทาง และ (ข) 64 เส้นทาง	27
รูปที่ 3.10	กราฟแท่งแสดงการประหยัดเวลาค้นหาเป็น % เมื่อใช้ MATS กับปัญหา (ก) BF (ข) RF และ (ค) SF.....	34
รูปที่ 3.11	กราฟลู่อู่เข้าหาคำตอบวงกว้างของ ATS และ MATS ทั้ง 6 ชนิด กับปัญหา BF	35
รูปที่ 4.1	แผนภาพแสดงระบบ CPP.....	37
รูปที่ 4.2	การเตรียมข้อมูลคิสำหรับารระบุเอกลักษณ์ระบบ.....	41
รูปที่ 4.3	ขั้นตอนการคำนวณค่า SSE	42
รูปที่ 4.4	กราฟการลู่อู่เข้าหาผลเฉลยของ MATS กับปัญหาการระบุเอกลักษณ์ระบบ พลวัต CPP	45

สารบัญภาพ(ต่อ)

หน้า

รูปที่ 4.5	ผลการระบุเอกลักษณ์ ที่ค่า $SSE = 3.1891$	46
รูปที่ 4.6	เปรียบเทียบสัญญาณจากการวัดและการระบุเอกลักษณ์ ตลอดทั้งช่วงข้อมูล 37.5 วินาที จำนวน 250 จุด แต่ละจุดห่างกัน 0.15 วินาที (ก) ช่วงข้อมูลที่ 1 (ข) ช่วงข้อมูลที่ 2 (ค) ช่วงข้อมูลที่ 3 (ง) ช่วงข้อมูลที่ 4 (จ) ช่วงข้อมูลที่ 5 และ (ฉ) ช่วงข้อมูลที่ 6 โดยช่วงข้อมูลที่ 2 ถึง 5 สำหรับการระบุเอกลักษณ์ ส่วนช่วงข้อมูลที่ 1 และ 6 สำหรับการประเมินความถูกต้อง.....	47

บทที่ 1

บทนำ

1.1 ความสำคัญของปัญหา

ผ่านมาหลายทศวรรษ เทคนิคการหาค่าที่เหมาะสมที่สุด (optimization techniques) เป็นเครื่องมืออันทรงพลังและมีประสิทธิภาพมากในการหาผลเฉลยที่ดีที่สุด สำหรับปัญหาที่ซับซ้อนทางด้านวิศวกรรมศาสตร์และวิทยาศาสตร์ คณะผู้วิจัยสาขาวิชาวิศวกรรมไฟฟ้า มหาวิทยาลัยเทคโนโลยีสุรนารี ได้เล็งเห็นถึงคุณค่าและความสำคัญของเทคนิคดังกล่าว จึงได้เริ่มปรับปรุงพัฒนาเทคนิคการค้นหาแบบตาบู่ (tabu search, TS) (Glover, 1986) ดั้งเดิม จนกระทั่ง พ.ศ. 2547 ได้กลายเป็นการค้นหาแบบตาบู่เชิงปรับตัว (adaptive tabu search, ATS) (Puangdownreong, D., Kulworawanichpong, T., Sujitjorn, S., 2004) ที่มีความสามารถในการค้นหาคำตอบได้รวดเร็วกว่าการค้นหาแบบตาบู่ดั้งเดิม โดยยืนยันศักยภาพได้จากบทความและวารสารที่เผยแพร่ระดับชาติและระดับนานาชาติมาแล้ว และเพื่อสานต่อแนวทางการพัฒนาเครื่องมืออันทรงพลังนี้ให้มีสมรรถนะที่สูงขึ้นต่อไป จึงเกิดเป็นหัวข้อวิจัยภายใต้ชื่อ “การพัฒนาอัลกอริธึมการค้นหาแบบตาบู่เชิงปรับตัวหลายวิถี” หรือ ชื่อภาษาอังกฤษ “development of multipath adaptive tabu search” หรือ ชื่อย่อภาษาอังกฤษว่า “MATS” โดยผู้วิจัยคาดหวังว่าอัลกอริธึมใหม่นี้จะทำความเร็วในการค้นหาคำตอบได้ดีกว่าการค้นหาแบบตาบู่เชิงปรับตัว และเพื่อให้เป็นทางเลือกที่ดีกว่าสำหรับนำไปประยุกต์ใช้ในการแก้ปัญหาที่ซับซ้อนในทางวิศวกรรมศาสตร์ด้านต่าง ๆ ต่อไป

1.2 วัตถุประสงค์ของการวิจัย

- เพื่อพัฒนาอัลกอริธึมการค้นหาแบบตาบู่เชิงปรับตัวหลายวิถี (multipath ATS หรือ MATS)
- เพื่อประยุกต์อัลกอริธึมการค้นหาแบบตาบู่เชิงปรับตัวหลายวิถี (MATS) กับงานด้านการระบุเอกลักษณ์ระบบรบกวนพหุคูณ

1.3 ขอบเขตของการวิจัย

- พัฒนาอัลกอริธึมการค้นหาแบบตาบู่เชิงปรับตัวหลายวิถีเพื่อแก้ปัญหาค่าที่เหมาะสมที่สุด
- ทดสอบสมรรถนะอัลกอริธึมการค้นหาแบบตาบู่เชิงปรับตัวหลายวิถีเปรียบเทียบกับอัลกอริธึมการค้นหาแบบตาบู่เชิงปรับตัว โดยทดสอบกับฟังก์ชันรูปแบบพิเศษที่มีหลายจุดต่ำสุด ได้แก่ ฟังก์ชัน โบซาเซฟสกีและฟังก์ชันราสทริกิน เป็นต้น
- วิเคราะห์หาแนวทางการปรับระบุเอกลักษณ์ระบบรบกวนพหุคูณด้วยการค้นหาแบบตาบู่เชิงปรับตัวหลายวิถี

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- ได้อัลกอริทึมการค้นหาแบบตาบุงเชิงปรับตัวหลายวิถี

1.5 ข้อตกลงเบื้องต้นของการวิจัย

- อัลกอริทึมที่จะพัฒนาขึ้นให้ดำเนินการตามลำดับขั้น (sequential)
- ปฏิบัติการค้นหาสามารถถูกแบ่งเป็นปริภูมิต่อๆได้
- ขอบเขตของปริภูมิการค้นหาจะถูกกำหนดขึ้นเพื่อใช้ประโยชน์ในการหาผลเฉลยตั้งต้นเมื่อเริ่มต้นเส้นทางการค้นหาต่อๆกัน

บทที่ 2

ปริทัศน์วรรณกรรมและอัลกอริธึมการค้นหาแบบตาบู่เชิงปรับตัว

2.1 กล่าวนำ

บทที่ 2 นี้ นำเสนอการปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้องกับการค้นหาแบบตาบู่ เพื่อเป็นแนวทางการพัฒนาอัลกอริธึมการค้นหาแบบตาบู่เชิงปรับตัวหลายวิธีต่อไป โดยสามารถแบ่งออกได้เป็นหกแนวทางได้แก่ การค้นหาแบบตาบู่เชิงปรับตัว การค้นหาแบบตาบู่เชิงปฏิบัติการ การค้นหาแบบตาบู่เชิงขนาน การค้นหาแบบตาบู่เชิงดัดแปลง การค้นหาแบบตาบู่เชิงความน่าจะเป็น และการค้นหาแบบตาบู่เชิงผสมผสาน โดยใช้ฐานข้อมูลที่เป็นแหล่งสะสมรายงานวิจัยและวรรณกรรมที่เกี่ยวข้องทางวิศวกรรมศาสตร์และเทคโนโลยี เช่น ฐานข้อมูล IEEE SpringerLink ScienceDirect และ Google เป็นต้น หัวข้อสุดท้ายของบทนี้ได้ทบทวนการค้นหาแบบตาบู่เชิงปรับตัวไว้ด้วย การปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้องมีรายละเอียดดังต่อไปนี้

2.2 ปริทัศน์วรรณกรรมที่เกี่ยวข้องกับการค้นหาแบบตาบู่

การค้นหาแบบตาบู่ (tabu search, TS) (Glover, 1986, 1989, 1990) เป็นวิธีการค้นหาคำตอบที่เหมาะสมที่สุดด้วยคอมพิวเตอร์ ในปัญหาแบบไม่ต่อเนื่องเชิงผสมผสาน การค้นหาแบบตาบู่อาศัยกระบวนการค้นหาคำตอบค่าใกล้เคียง (neighborhood search) และรายการตาบู่ (tabu list, TL) ซึ่งจะทำหน้าที่เก็บคำตอบในอดีต คำตอบใน TL สามารถใช้ประโยชน์ในกรณีการค้นหาเกิดการติดล็อกโดยคำตอบวงแคบเฉพาะถิ่น (local solution) โดยอาศัยเกณฑ์ปรารถนา (aspiration criteria, AC) กระบวนการค้นหาจะเคลื่อนที่แบบเคทีทอร์มินิสติก ซึ่งถูกกำหนดไว้อย่างมีระเบียบแบบแผน โดยทั่วไปจะถูกกำกับไว้ด้วยเงื่อนไขความคงอยู่ล่าสุด (recency) และเงื่อนไขความซ้ำซาก (frequency) (Glover, 1989; Glover and Laguna, 1997) ปัจจุบันพบว่ามีรูปแบบอันหลากหลายของ TS ดังที่จะทบทวนต่อไปนี้

การค้นหาแบบตาบู่เชิงปฏิบัติการ (reactive tabu search, RTS) (Battiti and Tecchiolli, 1994) มีการดัดแปลงจากการค้นหาแบบตาบู่ดั้งเดิม TS (tabu search) อยู่สองประการ ประการที่หนึ่งได้มีการดัดแปลงเทคนิคการค้นหาคำตอบเฉพาะถิ่นที่มีการเคลื่อนที่ ในรอบการคำนวณถัดไปจะเลือกจากคำตอบที่ดีที่สุดของค่าใกล้เคียงของจุดในรอบการค้นหาปัจจุบัน คำตอบที่ดีที่สุดของการค้นหาหมายถึงคำตอบที่ให้ค่าฟังก์ชันวัตถุประสงค์ต่ำที่สุด และประการที่สองเป็นการหลีกเลี่ยงการวนซ้ำ (cycle avoidance) เป็นกระบวนการป้องกันการย้อนกลับไปดำเนินการซ้ำส่วนที่เพิ่งจะดำเนินการไป โดยอาศัยรายการตาบู่ ส่วนกลไกปฏิบัติการ (reaction mechanism) เป็นกลไกปรับเปลี่ยนความยาวของรายการตาบู่ให้เหมาะสม ซึ่งกลไกปฏิบัติการนี้จะทำงานเมื่อพบว่าค่าฟังก์ชันวัตถุประสงค์ไม่ได้

ถูกปรับปรุงมาเป็นเวลานาน ซึ่งเป็นปรากฏการณ์ติดกับดัก (local trap) ดังนั้นกลไกปฏิบัติการนี้จึงช่วยการค้นหาแบบตามูเชิงปฏิกิริยาให้มีคุณสมบัติการหนีล๊อคได้เป็นอย่างดี การค้นหาแบบตามูเชิงปฏิกิริยา (Wassan, N., 2006) พัฒนาจาก RTS ของแบตติตีและเทคคิโอลี (R. Battiti and G. Tecchioli, 1994) เฉพาะในส่วนกลไกหนีล๊อค (escape mechanism) โดยการเปลี่ยนจากกลไกการจัดตำแหน่งใหม่แบบสุ่ม (random repositioning) ไปเป็นรูปแบบการแลกเปลี่ยนที่ 2 (2-exchanges scheme) ขณะที่ยังคงรูปแบบการแลกเปลี่ยนที่ 1 (1-exchanges scheme) ไว้อยู่ ทั้งรูปแบบการแลกเปลี่ยนที่ 1 และ รูปแบบการแลกเปลี่ยนที่ 2 เป็นเทคนิควิธีของกระบวนการค้นหาค่าใกล้เคียง ซึ่งช่วยเสริมกลยุทธ์ความหลากหลาย (diversification strategy) ให้กับกระบวนการค้นหาค่าตอบได้

การค้นหาแบบตามูเชิงขนาน (parallel tabu search, PTS) (Crainic, T.G., Toulouse, M., and Gendreau, M., 1996) เป็นการขนานการค้นหาแบบตามูเชิงลำดับ (sequential tabu search) หลายการค้นหาเข้าด้วยกัน มีหน่วยความจำส่วนกลางทำหน้าที่ติดต่อสื่อสารระหว่างหน่วยการค้นหา และตรวจจับค่าต่ำสุดที่เป็นคำตอบกว้าง ระหว่างที่การค้นหาหน่วยต่าง ๆ กำลังดำเนินการอยู่ การค้นหาแบบตามูเชิงขนาน (Talbi, E.G., Hafidi, Z. and Geib, J-M. 1998) เป็นการขนานการค้นหาแบบตามูจำนวนมากว่าหนึ่งการค้นหามัลติโพรเซสเซอร์เดียวกัน ให้ทำงานแบบพร้อมกัน โดยแต่ละการค้นหาที่มีกลไกการขนานแบบปรับตัวได้ (adaptive parallelism) ทำหน้าที่ปรับระดับการขนาน (parallelism degree) ให้เหมาะกับภาระงานของระบบ การค้นหาแบบตามูที่นำมาขนานกันจะทำงานอย่างอิสระโดยไม่ต้องมีการติดต่อสื่อสารกัน และเริ่มต้นด้วยคำตอบแรกที่แตกต่างกันรวมถึงพารามิเตอร์อื่นก็อาจต่างกันด้วย เช่น ความยาวของรายการตามู และจำนวนรอบการคำนวณสูงสุด นอกจากนี้ที่กล่าวนี้แล้วยังมีการค้นหาแบบตามูเชิงขนาน (Sait, S. M., Youssef H., Barada H. R., and Ahmad A.-Y., 2000) ซึ่งประกอบด้วย 3 ส่วนที่สำคัญ คือ ส่วนที่หนึ่งกระบวนการหลัก (master process) ส่วนที่สองผู้ปฏิบัติการค้นหาแบบตามู (tabu search workers, TSWs) และส่วนที่สามผู้ปฏิบัติการบัญชีรายชื่อ (candidate list workers, CLWs) อัลกอริทึมนี้ทำงานพร้อมกันอยู่สองระดับได้แก่ ระดับบน คือ ระดับกระบวนการค้นหาแบบตามู ซึ่งกระบวนการหลักจะทำการกำหนดจำนวนหน่วยผู้ปฏิบัติการค้นหาแบบตามู TSWs แล้วกำหนดคำตอบเริ่มต้นของแต่ละหน่วยผู้ปฏิบัติการค้นหาแบบตามูด้วยค่าเดียวกัน ส่วนระดับล่าง คือ ระดับการสร้างบัญชีรายชื่อคำตอบใกล้เคียงเฉพาะถิ่น (local neighborhood search) ระดับนี้หน่วยผู้ปฏิบัติการค้นหาแบบตามู TSWs ที่ถูกสร้างขึ้นแล้วในระดับบนแต่ละตัวจะอาศัยผู้ปฏิบัติการบัญชีรายชื่อ CLWs จำนวนมาก เพื่อค้นหาค่าตอบค่าใกล้เคียงของคำตอบปัจจุบัน โดยจะกำหนดคำตอบเริ่มต้น ช่วงการค้นหาที่ไม่ทับซ้อนกัน และพารามิเตอร์อื่น ๆ ที่จำเป็นให้กับ CLW แต่ละตัว คำตอบที่ดีที่สุดของ TSWs จะถูกคัดเลือกจากตัวประมวลผลหลัก (master) เพื่อกำหนดเป็นค่าเริ่มต้นในรอบการค้นหาถัดไปให้กับแต่ละ TSWs ดำเนินการค้นหาต่อไปจนครบรอบการค้นหาที่กำหนดไว้หรือพบคำตอบที่พอใจตาม

ที่ตั้งไว้ อัลกอริธึมนี้ถูกนำไปใช้แก้ปัญหาการจัดวางเซลล์มาตรฐานภายในวงจรรวมขนาดใหญ่มาก (VLSI standard-cell placement problem) ได้ผลเป็นที่น่าพอใจ

การค้นหาแบบตาบู่เชิงตัดแปดแปลง (modified tabu search, MoTS) (Nowicki, E. and Smutnicki, C., 2005) ชนิดนี้เป็นอัลกอริธึมที่พัฒนาขึ้นมาโดยอิงปรากฏการณ์หุบเขาขนาดใหญ่ (big valley phenomenon) ร่วมกับการใช้เทคนิคการเชื่อมเส้นทางใหม่ (path relinking technique) เพื่อสร้างคุณสมบัติการค้นหาคำตอบใกล้เคียงแบบใหม่ อัลกอริธึมนี้ได้ถูกนำไปใช้แก้ปัญหาการจัดตารางการทำงานของโรงงาน (job shop scheduling problem) ได้ผลเป็นที่น่าพอใจ การค้นหาแบบตาบู่เชิงตัดแปดแปลง (Zhang, D., Fu, Z. and, Zhang, L. C., 2006) ชนิดนี้ กำหนดพารามิเตอร์ความยาวรายการตาบู่ ด้วยกระบวนการผ่าเหล่า (mutation operation) ของจินตคณิตอัลกอริธึมซึ่งช่วยเพิ่มความสามารถการค้นหาคำตอบกว้าง และได้มีการประยุกต์ใช้อัลกอริธึมนี้กับการแก้ปัญหาการลดกำลังสูญเสียในระบบจำหน่ายขนาดใหญ่ สามารถทำให้สมรรถนะการค้นหาชนิดนี้ดีกว่าการค้นหาแบบตาบู่ดั้งเดิม การค้นหาแบบตาบู่เชิงตัดแปดแปลง (Glover F., 2006) ชนิดนี้ เป็นการพัฒนากระบวนการบรานซ์และบาวด์ (branch and bound procedure) ของโกลเวอร์ ที่เสนอไว้ครั้งแรกเมื่อ ค.ศ.1978 (Glover F., 1978) โดยการแทนที่หน่วยความจำการค้นหาแบบต้นไม้ (tree search memory) ด้วยรายงานหน่วยความจำแบบปรับตัวได้ของการค้นหาแบบตาบู่และได้ใช้กลยุทธ์ร่วมแบบใหม่ ที่มีความอ่อนตัวมากกว่ากลไกในกระบวนการบรานซ์และบาวด์ นอกจากนี้ยังได้ปรับปรุงกระบวนการเข้มข้น (intensification procedure) และกระบวนการหลากหลาย (diversification procedure) โดยอิงแนวคิดของการค้นหาแบบกระจาย (scatter search) การวิเคราะห์ความถี่ (frequency analysis) และความแปรปรวนของหน่วยความจำปรับตัวที่เรียกว่า หน่วยความจำที่ฝังแบบจำลอง (model embedded memory) อัลกอริธึมนี้ถูกนำไปใช้แก้ปัญหาการโปรแกรมจำนวนเต็มผสม (mixed integer programming, MIP) ได้ผลเป็นที่น่าพอใจ

การค้นหาแบบตาบู่เชิงความน่าจะเป็น (probabilistic tabu search, PrTS) (Xu, J., Chiu, S.Y. and Glover, F., 1997) อัลกอริธึมการค้นหาแบบตาบู่แบบดั้งเดิมพิจารณาทิศทางการเคลื่อนตัวของการค้นหา (move) เพียงพิจารณาจากรายการตาบู่และค่าฟังก์ชันวัตถุประสงค์ต่ำสุดของคำตอบตัวเลือก แต่ PrTS แตกต่างตรงที่ประยุกต์ใช้ความน่าจะเป็นเข้าร่วมกับการเลือกคำตอบถัดไปด้วย จึงช่วยแก้ไขข้อผิดพลาดของหน่วยความจำตาบู่ที่มีสัญญาณรบกวน กลยุทธ์รายการคำตอบทางเลือกและกลยุทธ์ความเข้มข้นที่อาศัยการนำคำตอบที่เหมาะสมกลับมาพิจารณาใหม่ เทคนิคนี้จึงมีชื่อเรียกว่าการคัดเลือกทิศทางการเคลื่อนตัวเชิงความน่าจะเป็น (probabilistic move selection) และเป็นที่มาของชื่ออัลกอริธึมนี้นั่นเอง อัลกอริธึมนี้ได้ถูกนำไปใช้ปัญหาการออกแบบเครือข่ายสื่อสารทางไกล ได้ผลเป็นที่น่าพอใจ การค้นหาแบบตาบู่เชิงความน่าจะเป็น (Kochetov, Y.A. and Goncharov, E.N., 2000) เป็นอัลกอริธึมที่ใช้กลไกการสร้างคำตอบใกล้เคียงแบบสุ่มที่มีค่าขีดเริ่มคงตัว นอกจากนี้ยังประยุกต์กฎเชิงปรับตัวเพื่อจัดการความยาวรายการตาบู่ อัลกอริธึมนี้จะสร้าง

ลูกโซ่มาคอฟแบบไม่เป็นรายคาบจำนวนจำกัดจากเซตที่เหมาะสมภายในเซตคำตอบที่ได้ คุณสมบัติเช่นนี้ยืนยันการลู่เข้าหาผลเฉลยเชิงเส้นกำกับของคำตอบที่ดีที่สุดที่ได้ ซึ่งจะนำไปสู่การพบคำตอบทั่วไปที่ความต้องการอัลกอริธึมนี้ได้ถูกนำไปใช้แก้ปัญหาการจัดวางตำแหน่งเครื่องอำนวยความสะดวกแบบหลายช่วง (multi-stage uncapacitated facility location problem) ได้ผลเป็นที่น่าพอใจ การค้นหาแบบตามูเชิงความน่าจะเป็น ยังสามารถนำมาประยุกต์ได้กับปัญหาการกระจายของต้นไม้ ในลักษณะของอัลกอริธึมที่ใช้ปฏิบัติการก่อนกระบวนการ สำหรับหาค่าความน่าจะเป็นของเซตค่ายอดต่าง ๆ ที่สัมพันธ์กับคำตอบที่เหมาะสมและใช้ค่าความน่าจะเป็นนี้ กำหนดขนาดของคำตอบค่าใกล้เคียงให้เหมาะสม อัลกอริธึมนี้ได้ถูกนำไปแก้ปัญหาทั่วไปของต้นไม้กระจายตัวน้อยที่สุด (generalized minimum spanning tree problem)

การค้นหาแบบตามูเชิงผสมผสาน (hybrid tabu search, HTS) (Ongsakul, W. and Bhasaputra, P., 2002) มีโครงสร้างหลักเป็นการค้นหาแบบตามูธรรมดา และมีอัลกอริธึมการอบอุ่นจำลอง (simulated annealing algorithm) ทำหน้าที่สร้างคำตอบค่าใกล้เคียง (neighborhood solution) ให้กับการค้นหาแบบตามูหลัก นอกจากนี้ยังอาศัยเงื่อนไขการยอมรับเชิงความน่าจะเป็น (probabilistic acceptance criterion) ของอัลกอริธึมการอบอุ่นมาทดแทน ระดับความปรารถนา (aspiration level) ของอัลกอริธึมการค้นหาแบบตามูอีกด้วย รวมทั้งแผนการอบอุ่น (cooling schedule) และรายชื่อตามูได้รับประยุกต์ใช้ในอัลกอริธึมนี้ด้วย การค้นหาแบบตามูเชิงผสมผสาน (Gong, D., Zhou, Y., Guo, X., Xiaoping M, and Ming Li, 2002) ชนิดนี้ประกอบขึ้นจากอัลกอริธึมหลักสองชนิด คือ จินติกอัลกอริธึม และอัลกอริธึมการค้นหาแบบตามู ปกติแล้วจินติกอัลกอริธึมจะประกอบด้วยกระบวนการลดปริภูมิการค้นหาและกระบวนการคงความหลากหลายของวิวัฒนาการในประชากร ซึ่งกระบวนการทั้งสองมีแนวทางที่ขัดแย้งกัน (trade-off) จึงมักก่อให้เกิดปัญหาการหาค่าที่เหมาะสม ดังนั้นเพื่อลดความขัดแย้งดังกล่าวจึงอาศัยการค้นหาแบบตามูมาหาค่าที่เหมาะสม การค้นหาแบบตามูเชิงผสมผสาน (Mori, H. and Tani, H., 2003) ชนิดนี้ เป็นการพัฒนาการค้นหาแบบตามูเชิงขนาน (parallel tabu search, PTS) โดยอาศัยการหาค่าเหมาะสมที่สุดแบบออร์ดิแนล (ordinal optimization, OO) มาช่วยสร้างคำตอบค่าใกล้เคียง (neighborhood) ให้กับ PTS ทำการเลือกในรูปแบบของความน่าจะเป็น การค้นหาแบบตามูเชิงผสมผสาน (Kalinli, A., 2004) ชนิดนี้ ประกอบขึ้นจากอัลกอริธึมการค้นหาแบบตามูและจินติกอัลกอริธึม โดยการค้นหาแบบตามูจำนวนมากกว่าหนึ่งกระบวนการทำงานไปพร้อม ๆ กันแล้วส่งผลการค้นหาไปให้จินติกอัลกอริธึมผ่านกระบวนการไขว้ข้าม (crossover operation) เพื่อสร้างผลการค้นหาใหม่ ๆ จนได้ผลการค้นหาสุดท้ายในที่สุด การค้นหาแบบตามูเชิงผสมผสาน (Swamkar, R. and Tiwari, M.K., 2004) ชนิดนี้มีโครงสร้างหลักเป็นอัลกอริธึมการอบอุ่นจำลอง และมีการค้นหาแบบตามูทำหน้าที่สร้างหน่วยความจำระยะสั้น สำหรับจดจำคำตอบที่เพิ่งพบในช่วงเวลาหนึ่ง เป็นการป้องกันการติดล็อกค่าต่ำสุดเฉพาะถิ่น แต่ธรรมชาติของการค้นหาแบบตามูเป็นเชิงกำหนด (deterministic) มักประสบ

กับปัญหาการวนซ้ำ ซึ่งก็สามารถหลีกเลี่ยงปัญหานี้ด้วยคุณสมบัติทางสโตแคสติก (stochastic characteristic) ของอัลกอริทึมการอบอุ่น ดังนั้นเมื่อรวมเอาคุณสมบัติที่ดีของทั้งสองอัลกอริทึมเข้าด้วยกัน แล้วนำไปประยุกต์ใช้กับปัญหากระบวนการผลิตแบบอบอุ่นตัวจึงให้ผลการค้นหาที่ค่อนข้างน่าพอใจ การค้นหาแบบตามูเชิงผสมผสาน (Liu, Y., Liu, Y., Wang, L. and Chen, K., 2005) ชนิดนี้มีการค้นหาแบบตามู เป็นโครงข่ายงานหลักที่มีการเสริมสมรรถนะการหาค่าคำตอบด้วยปฏิบัติการเค-มีนส์ (K-means operation) ซึ่งเป็นวิธีการในอัลกอริทึม เค-มีนส์ (K-means algorithm) และมีการเพิ่มศักยภาพการสร้างคำตอบค่าใกล้เคียงให้กับการค้นหาแบบตามูด้วยปฏิบัติการผ่าเหล่า (mutation operation) ซึ่งเป็นวิธีการในจินตนาการอัลกอริทึม การค้นหาแบบตามูเชิงผสมผสาน (Pedroso, J. P. and Kubo, M., 2005) ชนิดนี้ มีการค้นหาแบบตามูเป็นโครงข่ายงานหลัก ซึ่งคงไว้เฉพาะหน่วยความจำระยะสั้น (short term memory) และผสมผสานอัลกอริทึมตรึงและผ่อนคลาย (relax-and-fix-one-product) เข้ามาเสริมสมรรถนะความหลากหลายและเพื่อช่วยปลดปล่อยการติดขัด เมื่อการค้นหาแบบตามูดำเนินการไปจนพบว่าคำตอบไม่ได้รับการปรับปรุงในช่วงระยะเวลาหนึ่งแล้วจะมีกลไกไปกระตุ้นให้อัลกอริทึมตรึงและผ่อนคลายเริ่มทำงาน หลักการทำงานของอัลกอริทึมตรึงและผ่อนคลายจะเริ่มต้นด้วยการไปลบรายการตามู ที่บันทึกไว้ออกบางส่วนหรือทั้งหมด หลังจากนั้นจึงเริ่มสร้างคำตอบค่าใกล้เคียงขึ้นมาใหม่เพื่อปรับปรุงโอกาสของการพบคำตอบที่ดีขึ้น แล้วทำการเคลื่อนตัวไปยังคำตอบที่ดีที่สุด เมื่อการค้นหาเริ่มเข้าสู่ภาวะปกติ อัลกอริทึมตรึงและผ่อนคลายหยุดทำงาน การค้นหาจะกลับมาอยู่ในโครงสร้างการค้นหาแบบตามูเช่นเดิม การค้นหาแบบตามูเชิงผสมผสาน (Chen, J-S., Pan, J. C-H. and Wu, C-K., 2006) ชนิดนี้ แตกต่างจาก TS ดั้งเดิม คือ เมื่อกระบวนการค้นหาไม่พบคำตอบที่ดีขึ้นจากค่าใกล้เคียงในช่วงเวลาหนึ่ง กฎการจ่าย (dispatching rule) ที่เป็นวิธีการศึกษาสำนึก (heuristic) ของปัญหาเฉพาะด้านจะเข้าไปสร้างบริเวณคำตอบใหม่แทน ทำให้ HTS สามารถหนีการติดขัดและเคลื่อนตัวไปหาคำตอบที่ดีขึ้นได้ พารามิเตอร์ที่สำคัญเจ็ดค่าของอัลกอริทึมนี้ ได้แก่ คำตอบเริ่มต้น การค้นหาคำตอบค่าใกล้เคียง การเลือกทิศการเคลื่อนที่ รายการตามู คำตอบที่ดีที่สุด กฎการหยุด และวิธีการผสมผสาน วิธีการผสมผสานในอัลกอริทึมนี้ จะกระทำกับคำตอบล่าสุดที่ไม่ได้รับการปรับปรุงมานานถึงระยะเวลาหนึ่ง โดยการแบ่งคำตอบนั้นออกเป็นสามส่วนแต่ละส่วนมีจำนวนบิตเป็นไปตามการสุ่ม หลังจากนั้นจึงทำการผสมผสานทำนองเดียวกับกระบวนการจินตนาการ ก็จะได้คำตอบใหม่ที่มีคุณสมบัติในการติดขัดได้ดี การค้นหาแบบตามูเชิงผสมผสาน (Victoire, T. A. A. and Jeyakumar, A. E., 2006) ชนิดนี้ เกิดจากรวมกันของสามอัลกอริทึม ได้แก่ การค้นหาแบบตามู การหาค่าเหมาะที่สุดแบบฝูงอนุภาค (particle swarm optimization, PSO) และเทคนิคการโปรแกรมกำลังสองเชิงลำดับ (sequential quadratic programming technique, SQP) เพื่อแก้ปัญหาการวางแผนการผลิตไฟฟ้า (unit commitment problem, UCP) TS ทำหน้าที่แก้ปัญหาโปรแกรมย่อยแบบผสมผสานของ UCP โดยสร้างรูปแบบการรบกวนแบบสุ่ม (random perturbation scheme) และสร้างตารางการพันธกรณีเริ่มต้น (initial feasible commitment

schedule) ส่วน PSO และ SQP ใช้แก้ปัญหาโปรแกรมย่อยแบบไม่เชิงเส้นของ UCP การค้นหาแบบตามเชิงผสมผสาน (Tahir, M. A., Bouridane A. and Kurugollu, F., 2007) ชนิดนี้อาศัยแนวคิดแบบการค้นหาแบบตามมาช่วยกระบวนการเลือกคุณลักษณะและกระบวนการถ่วงน้ำหนักภายในการคัดแยก K-NN (K-nearest-neighbor classifier) ซึ่งเป็นการคัดแยกที่นิยมใช้ในเทคโนโลยีการตรวจรู้รูปแบบ (pattern recognition)

การค้นหาแบบตามเชิงปรับตัว (adaptive tabu search, ATS) (Kolahan, F. and Liang, M., 1998) ชนิดนี้ได้เสนอกลไกการสร้างคำตอบค่าใกล้เคียงแบบปรับตัว (adaptive neighborhood generation, ANG) ที่สามารถปรับย่านการสลับที่ (transposition range) ให้เหมาะสมกับการปรับปรุงฟังก์ชันวัตถุประสงค์ (objective improvement) อยู่เสมอ คุณสมบัติที่แตกต่างจากการค้นหาแบบตามดั้งเดิมมีอยู่สองประการ คือ การปรับเปลี่ยนย่านการสลับที่ตามการปรับปรุงฟังก์ชันวัตถุประสงค์ และการลดย่านการสลับที่สองด้าน (double-ended transposition range) อัลกอริทึมนี้ได้ถูกนำไปใช้หาค่าเหมาะที่สุดในแบบได้อย่างเสียอย่าง (best trade-off) ระหว่างเป้าหมายกระบวนการผลิตแบบตรงเวลาพอดี (just-in-time production) กับภาวะกดดันช่วงเวลาการผลิต (processing time compression) และค่าใช้จ่ายที่เพิ่มขึ้น (extension costs) ผลการทดสอบอัตราการใช้และคุณภาพของคำตอบ เมื่อเปรียบเทียบกับวิธีการสร้างคำตอบค่าใกล้เคียงวิธีอื่น พบว่า ATS ชนิดนี้ให้ผลดีกว่าอย่างเห็นได้ชัด การค้นหาแบบตามเชิงปรับตัว (Franca, P. M., Sosa, N. M., and Pureza, V., 1999) ประกอบด้วยสองกลไกหลักคือ กลไกการเริ่มต้นคำตอบ (initial solution) ซึ่งใช้การศึกษาสำนึกแบบสร้างเสริม (constructive heuristics) ที่อาศัยทั้งน้ำหนัก (weight) และระยะทาง (distance) ในการหาค่าที่เหมาะสม ส่วนกลไกที่สอง คือ กลไกการสร้างคำตอบค่าใกล้เคียง (neighborhood generation) อาศัยการศึกษาสำนึกแบบการค้นหาเฉพาะถิ่น (local search heuristics) ซึ่งผนวกเทคนิคการสลับคู่ (pairwise interchange) และเทคนิคการเติม (insertion) มาเสริมสมรรถนะ อัลกอริทึมนี้ได้้นำไปใช้แก้ปัญหาการจัดกลุ่มลูกค้า (clustering problem) ได้ผลการทดสอบเป็นที่น่าพอใจ การค้นหาแบบตามเชิงปรับตัวฉบับของคณะวิจัยมหาวิทยาลัยเทคโนโลยีสุรนารี ได้เสนอสองกลไกหลักเพื่อการปรับตัวหนีการติดกับและการปรับตัวให้มีโอกาสพบคำตอบวงกว้างให้รวดเร็วขึ้น สองกลไกดังกล่าวก็คือกลไกย้อนรอยการค้นหา (back-tracking mechanism) และกลไกปรับรัศมีการค้นหา (adaptive radius mechanism) ถ้าพิจารณาเชิงยุทธศาสตร์ของการค้นหาแล้วการเสริมกลไกทั้งสองเปรียบได้กับการเสริมความแข็งแรงให้ทั้งกลยุทธ์ความหลากหลายและกลยุทธ์ความเข้มข้น กับการค้นหาแบบตามดั้งเดิม ซึ่งภายหลังจากที่เพิ่มกลไกทั้งสองเข้าไปแล้ว ส่งผลทำให้การค้นหาแบบตามเชิงปรับตัวใหม่นี้ได้มีสมรรถนะการค้นหาที่ดีขึ้นอย่างน่าพอใจ และยังสามารถประยุกต์ใช้ในการแก้ปัญหาสำหรับงานที่มีความซับซ้อนสูง เช่น ระบบควบคุมระบบไฟฟ้ากำลัง และระบบไฟฟ้าแรงสูง เป็นต้น รายละเอียดของอัลกอริทึมการค้นหาแบบตามเชิงปรับตัวฉบับนี้ได้รับการทบทวนไว้ในหัวข้อถัดไป

ผลการปรับวัตรณกรรมดังกล่าวมาแล้วข้างต้น สามารถแบ่งได้เป็นสองกลุ่มใหญ่ ๆ ตามจำนวนตัวประมวลผลที่ถูกรูปร่างได้แก่ กลุ่มอัลกอริทึมการค้นหาแบบตามูเชิงอนุกรม (sequential tabu search algorithm) ที่มีการใช้งานตัวประมวลผลเพียงตัวเดียว และกลุ่มอัลกอริทึมการค้นหาแบบตามูเชิงขนาน (parallel tabu search algorithm) ซึ่งใช้ตัวประมวลผลมากกว่าหนึ่งในการดำเนินการค้นหาไปพร้อม ๆ กัน เพื่อเพิ่มความเร็วการค้นหาให้สูงขึ้น เทคนิคที่มักพบประยุกต์ใช้ในทั้งสองกลุ่มได้แก่เทคนิคการสร้างคำตอบเริ่มต้น เทคนิคสร้างคำตอบใกล้เคียง เทคนิคการเคลื่อนที่ เทคนิคการบริหารรายการตามู และเทคนิคการเริ่มต้นใหม่ เป็นต้น ซึ่งเทคนิคเหล่านี้เป็นเสมือนเครื่องมือในสองกล่องเครื่องมือของอัลกอริทึมการค้นหาแบบตามู นั่นคือกลยุทธ์ความหลากหลาย (diversification strategy) และกลยุทธ์ความเข้มข้น (intensification strategy) นั่นเอง และการค้นหาแบบตามูเชิงปรับตัวฉบับที่เสนอ โดยผู้วิจัยของมหาวิทยาลัยเทคโนโลยีสุรนารีก็จัดอยู่ในกลุ่มการค้นหาแบบตามูเชิงอนุกรม เนื่องจากใช้ตัวประมวลผลเดี่ยวด้วยเช่นกัน

2.3 อัลกอริทึมการค้นหาแบบตามูเชิงปรับตัว

การค้นหาแบบตามูเชิงปรับตัวฉบับนี้ ถูกพัฒนาต่อจากอัลกอริทึมการค้นหาแบบตามูดั้งเดิม โดยเพิ่มกลไกย้อนรอยการค้นหา (back-tracking mechanism) และกลไกปรับรัศมีการค้นหา (adaptive radius mechanism) กลไกย้อนรอยการค้นหาเป็นกระบวนการค้นคืนคำตอบที่เคยพบแล้วและอยู่ในรายการตามู มาเป็นคำตอบเริ่มต้นตัวใหม่แทนการหาคำตอบถัดไปด้วยกระบวนการปกติ ทั้งนี้เนื่องจากบางโอกาสกระบวนการค้นหาที่ไม่สามารถหาคำตอบถัดไปที่ดีกว่าเดิมได้ หมายความว่า การค้นหากำลังอยู่ในสถานการณ์ติดกับดัก (entrapment) นั่นเอง คณะผู้วิจัยมหาวิทยาลัยเทคโนโลยีสุรนารี จึงเสนอกลไกย้อนรอยนี้มาเพื่อแก้ปัญหาดังกล่าว โดยตั้งเงื่อนไขการเรียกใช้กลไกย้อนรอยนี้ด้วยจำนวนรอบสูงสุดที่ไม่สามารถหาคำตอบถัดไปที่ดีกว่าได้ ก็ให้มีการเรียกใช้กลไกดังกล่าว เพื่อเปิดโอกาสให้การค้นหาได้เปลี่ยนพื้นที่การค้นหาใหม่ได้ ส่วนกลไกปรับรัศมีการค้นหา จะทำหน้าที่ปรับรัศมีการค้นหาให้สั้นลงโดยยังรักษาจำนวนคำตอบใกล้เคียงที่จะถูกสุ่มเลือกภายในรัศมีการค้นหาเพื่อเป็นคำตอบถัดไปไว้เท่าเดิม ซึ่งเปรียบเสมือนการเพิ่มความสามารถของการมองเห็นคำตอบวงกว้างให้ดีขึ้นอีกด้วย

การค้นหาแบบตามูที่เสนอนี้ เริ่มต้นการค้นหาด้วยการสุ่มคำตอบแรกภายในปริภูมิการค้นหาแล้วกำหนดให้เป็นศูนย์กลางเพื่อหาคำตอบถัดไปในระยะรัศมีที่กำหนด คำตอบถัดไปจะเป็นคำตอบที่ให้ค่าการปรับปรุงฟังก์ชันวัตถุประสงค์ที่สุด ในจำนวนที่สุ่มได้ การดำเนินการค้นหาจะดำเนินการวนซ้ำจนกระทั่งการปรับปรุงฟังก์ชันวัตถุประสงค์ถึงค่าขีดเริ่มที่กำหนดไว้ของกลไกปรับรัศมี รัศมีการค้นหาจะหดสั้นเข้าตามลำดับที่ตั้งไว้เพื่อให้พื้นที่การสุ่มคำตอบถัดไปแคบลง เป็นการเพิ่มโอกาสและเพิ่มความเร็วการพบคำตอบวงแคบเฉพาะถิ่นอีกด้วย เมื่อการค้นหาเริ่มถึง

The generic TS algorithms can be summarized as follows:

STEP 0: Initialization

Generate an initial solution, neighbourhood; set best solution, AC, TC, TL and iteration counter.

STEP 1: Iteration

Generate search space and possible solutions. Evaluate cost values for all solutions belonging to the neighbourhood. If the current best solution from the neighbourhood has a lower cost than the best-solution, replace the best-solution by the current best and update the TL by storing (tabuing) the (previous) best-solution, otherwise the best-solution remains unchanged and the current best solution becomes tabu instead. If the search has not been able to improve the best-solution for a certain time, go to STEP 3. Update counter.

STEP 2: Termination

Exit with the global optimum solution if the TC is (are) met, otherwise go to STEP 1.

STEP 3: Aspiration

Perform algorithmically according to the AC. Update TL, and counter. Go to STEP 1.

(ก)

The proposed ATS algorithms can be summarized as follows:

STEP 0: Initialization

Generate an initial solution, neighbourhood; set best solution, AC, TC, TL and iteration counter.

STEP 1: Iteration

Generate search space and possible solutions. Evaluate cost values for all solutions belonging to the neighbourhood. If the current best-solution from the neighbourhood has a lower cost than the best-solution, replace the best solution by the current best and update the TL by storing (tabuing) the (previous) best-solution, otherwise the best-solution remains unchanged and the current best solution becomes tabu instead. If the search has not been able to improve the best-solution for a certain time, go to STEP 3. Update counter.

STEP 2: Termination

Exit with the global optimum solution if the TC is (are) met, otherwise go to STEP 1.

STEP 3: Aspiration

Perform algorithmically according to the AC. If deadlock occurs, invoke the BT mechanism. If the cost value of the current best solution is lower than the preset cost, invoke the AR mechanism. Update TL, and counter. Go to STEP 1.

(ข)

The proposed BT mechanism can be summarized as follows:

STEP 1: Loading

Load a solution in TL

STEP 2: Setting

Set the solution from STEP 1 to be next solution of ATS.

STEP 3: Return to ATS

Exit BT and return to ATS.

(ค)

The proposed AR mechanism can be summarized as follows:

STEP 1: Reduce search radius

Search radius is reduced by step of percentage of the previous search radius with monitoring to the cost function value

STEP 2: Exit AR.

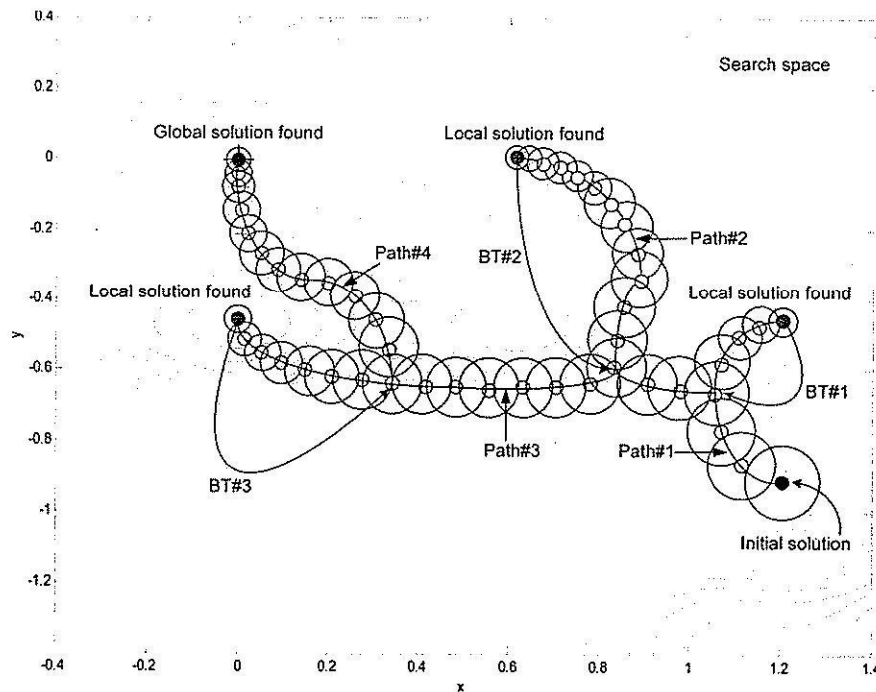
If value of the cost function is not under the AR criterion, the current search radius is replaced by the default search radius. It means ATS exit AR.

(ง)

รูปที่ 2.1 อัลกอริทึมการค้นหา (ก) แบบตาบูคั้งเดิม (ข) แบบตาบูเชิงปรับตัวที่เสนอ

(ค) กลไกย้อนรอยการค้นหา และ (ง) กลไกปรับรัศมีการค้นหา

ขีดจำกัดในการปรับปรุงฟังก์ชันวัตถุประสงค์ หรือการค้นหายุ่งใกล้คำตอบวงแคบเฉพาะถิ่นนานพอสมควรกลไกย้อนรอยการค้นหาก็จะถูกกระตุ้นให้ทำงานอีก เพื่อปลดการติดกับดักท้องถิ่น (local trap) วิธีการที่กลไกย้อนรอยทำให้การค้นหาลุดจากการติดกับดัก คือ การกำหนดคำตอบถัดไป (next solution) จากคำตอบในรายการตาม (tabu list) มาหนึ่งคำตอบซึ่งเป็นคำตอบที่การค้นหาค้นพบมาแล้วในอดีต ทั้งนี้เพื่อเป็นการเปลี่ยนทิศทางการค้นหา และเปลี่ยนตำแหน่งการค้นหาเพื่อหนีล่อไปในตัวด้วย ภายหลังจากการค้นหาหนีล่อเฉพาะถิ่นได้ ก็จะเข้าสู่สภาวะการทำงานปกติ และมุ่งค้นหาคำตอบเฉพาะถิ่นที่ดีขึ้นและพบคำตอบเฉพาะถิ่นที่จะกลายเป็นคำตอบทั่วไป (global solution) ในที่สุด ผู้อ่านสามารถศึกษาพัฒนาการของการค้นหาแบบตามจนเป็นการค้นหาแบบตามที่เสนอได้จากอัลกอริธึมใน รูปที่ 2.1 ร่วมกับภาพเส้นทางการค้นหาของ ATS ที่เสนอ ดังที่แสดงในรูปที่ 2.2 จะช่วยให้เข้าใจง่ายขึ้นได้



รูปที่ 2.2 เส้นทางการค้นหาและสองกลไกหลักของการค้นหาแบบตามเชิงปรับตัวที่เสนอ




พิจารณารูปที่ 2.2 อัลกอริธึมการค้นหาแบบตามเชิงปรับตัวที่เสนอ เริ่มต้นการค้นหาด้วยคำตอบเริ่มต้น (initial solution) ที่อยู่ตอนล่างซ้ายของปริภูมิการค้นหา คำตอบที่ได้รับการคัดเลือกให้เป็นคำตอบถัดไปอยู่ทางตอนบนเฉียงไปทางซ้ายของคำตอบเดิม เมื่อผ่านการค้นหาไปได้ประมาณสี่รอบได้มีการเรียกใช้กลไกปรับรัศมีการค้นหาหรือ AR ร่วมด้วยสังเกตจากขนาดของรัศมีการค้นหาที่มีแนวโน้มนั่นลงตามลำดับตามเส้นทางการค้นหาที่ 1 (Path#1) จนกระทั่งการค้นหาดำเนินไปจนสุดเส้นทางนี้ก็จะพบคำตอบท้องถิ่นซึ่งตั้งอยู่เหนือคำตอบเริ่มต้น ดังแสดงในรูปที่ 2.2 การวนซ้ำของคำตอบก็จะเกิดขึ้นเนื่องจากค่าฟังก์ชันวัตถุประสงค์ไม่ได้รับการปรับปรุง จนกระทั่ง

จำนวนการวนซ้ำถึงเกณฑ์ที่กำหนดไว้ กลไกย้อนรอยการค้นหาก็จะถูกเรียก ซึ่งจากรูปที่ 2 ก็จะเป็นการเรียกกลไกนี้เป็นครั้งแรก BT#1 ซึ่งผลการเรียกใช้กลไกย้อนรอยจะได้คำตอบถัดไป จากหนึ่งในรายการคาบูกิที่เคยผ่านมาซึ่งสิ่งที่เห็นตามรูปที่ 2.2 จะเป็นคำตอบที่อยู่ระหว่างคำตอบเริ่มต้นกับคำตอบเฉพาะถิ่นที่ 1 หลังจากนั้นกระบวนการค้นหาคาบูกิก็ดำเนินไปตามปกติ ซึ่งได้คำตอบถัดไปเคลื่อนตัวไปทางซ้ายในแนวระดับแล้ววกขึ้นทางตอนบนได้ระยะหนึ่ง กลไกปรับรัศมีก็ถูกเรียกใช้ตามเกณฑ์อีก จนกระทั่งการค้นหาเดินทางไปพบคำตอบเฉพาะถิ่นที่ 2 และการวนซ้ำก็เกิดขึ้นจนต้องเรียกกลไกย้อนรอยเป็นครั้งที่ 2 จึงส่งคำตอบเก่าที่มีคุณภาพมาให้อยู่ใกล้ ๆ กับคำตอบเริ่มต้น และหลังจากนั้น ATS ก็ดำเนินการค้นหาต่อไปเรื่อย ๆ ตามกลไกดังกล่าว จนกระทั่งพบคำตอบเฉพาะถิ่นที่ 3 และในที่สุดคำตอบวงกว้าง (global solution) ถูกพบบริเวณบนซ้ายของปริภูมิการค้นหา เกณฑ์ยุติการค้นหา (termination criteria, TC) จึงสั่งหยุดการค้นหาแล้วจึงรายงานคำตอบวงกว้างพร้อมค่าฟังก์ชันวัตถุประสงค์ให้ทราบ ถือเป็นการสิ้นสุดกระบวนการค้นหา

2.4 ผลและอภิปราย

อัลกอริธึมการค้นหาแบบคาบูกิเชิงปรับตัวฉบับนี้ ได้ถูกโปรแกรมด้วย MATLABTM เพื่อศึกษาสมรรถนะการค้นหา โดยให้ค้นหาคำตอบวงกว้างของสามฟังก์ชันในปัญหาการหาค่าเหมาะที่สุดเชิงพื้นผิว ได้แก่ ฟังก์ชันโบฮาเชฟสกี - bohachevsky function (BF) ฟังก์ชันราสตริกิน - rastrigin function (RF) และฟังก์ชันชีเคิลฟอกซ์โฮล - shekel's foxholes function (SF) รายละเอียดคุณสมบัติของฟังก์ชันทดสอบแสดงในตารางที่ 2.1

ตารางที่ 2.1 ฟังก์ชันพื้นผิวที่ใช้ทดสอบอัลกอริธึม

Surface names	Surface functions	Search spaces	Sketches
Bohachevsky	$f(x, y) = x^2 + 2y^2 - 0.3 \cos(3\pi x) - 0.4 \cos(4\pi y) + 0.7$ $f_{\min}(0, 0) = 0$	$[-2, 2]$	
Rastrigin	$f(x, y) = x^2 + y^2 - 10 \cos(2\pi x) - 10 \cos(2\pi y) + 20$ $f_{\min}(0, 0) = 0$	$[-2, 2]$	
Shekel's foxholes	$f(x_1, x_2) = \left[\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})^6} \right]^{-1}$ where $a_{ij} = \begin{pmatrix} -32 & -16 & 0 & 16 & 32 & -32 & \dots & 0 & 16 & 32 \\ -32 & -32 & -32 & -32 & -32 & -16 & \dots & 32 & 32 & 32 \end{pmatrix}$ $f_{\min}(-32, -32) = 1$	$[-40, 40]$	

ฟังก์ชันโบฮาเชฟสกี (Bohachevsky's function) หรือ ฟังก์ชัน BF เป็นฟังก์ชันที่มีตัวแปรสองตัว คือ x และ y สมการฟังก์ชัน BF ปรากฏคำตอบวงแคบเฉพาะถิ่นจำนวนมาก แต่จะมีคำตอบวงกว้างหรือคำตอบ x และ y ที่ทำให้ $f(x, y)$ มีค่าน้อยที่สุดมีอยู่เพียงคำตอบเดียว คือ $x = y = 0$

ที่ให้ค่า $f(x, y) = 0$ ฟังก์ชันที่สอง คือ ฟังก์ชันราสตริกิน (Rastrigin's function) หรือ ฟังก์ชัน RF เป็นฟังก์ชันที่มีตัวแปรสองตัวแปร เช่นเดียวกับฟังก์ชัน BF และมีค่าตอบวงกว้างอยู่ที่ตำแหน่ง $x = y = 0$ และ $f(x, y) = 0$ แตกต่างจาก ฟังก์ชัน BF ตรงที่พื้นผิวมีความลึกและชันมากกว่า ฟังก์ชันที่สามได้แก่ ฟังก์ชันชีเคิลฟอกซ์โฮล (Shekel's foxholes function) หรือ ฟังก์ชัน SF เป็นฟังก์ชันที่มีตัวแปรสองตัว คือ x_1 และ x_2 และมีค่าฟังก์ชัน $f(x_1, x_2)$ ซึ่งพบว่าฟังก์ชัน SF ปรากฏคำตอบวงแคบเฉพาะถิ่นจำนวนมากแต่จะมีคำตอบวงกว้างหรือคำตอบ x_1 และ x_2 ที่ทำให้ $f(x_1, x_2)$ มีค่าน้อยที่สุดมีเพียงคำตอบเดียวคือ $x_1 = x_2 = -32$ ซึ่งจะทำให้ $f(x_1, x_2) = 0.998$ พารามิเตอร์ของอัลกอริทึมการค้นหาแบบตามูเชิงปรับตัวที่ใช้ในการทดสอบแสดงรายละเอียดดังในตารางที่ 2.2

ตารางที่ 2.2 พารามิเตอร์ของ ATS ที่ใช้ทดสอบ

Test functions	ATS parameters									
	search space	search radius	no. of neighbours	BT		AR			TC	
				Re_{max}	k^{th}	stage I	stage II	stage III	count _{max}	J_{min}
BF	[2,-2]	0.2 (5%)	30	5	5	J<0.1, R=0.002	J<0.001, R=0.0001	-	10,000	1×10^{-9}
RF	[2,-2]	0.2 (5%)	30	5	5	J<0.1, R=0.002	J<0.001, R=0.0001	-	10,000	1×10^{-8}
SF	[40,-40]	0.8 (1%)	30	5	5	J<5, R=0.2	J<2, R=0.1	-	10,000	0.999

รัศมีการค้นหาค้นควรตั้งไว้ไม่เกิน 20 เปอร์เซ็นต์ของปริภูมิการค้นหา เช่น ฟังก์ชัน RF กำหนดไว้ 5 เปอร์เซ็นต์และฟังก์ชัน SF กำหนดไว้ 1 เปอร์เซ็นต์ของปริภูมิการค้นหา เป็นต้น จำนวนคำตอบใกล้เคียงของสามฟังก์ชันเลือกใช้ 30 ค่า ส่วนจำนวนการซ้ำของคำตอบสูงสุดก่อนการเรียกใช้กลไกย้อนรอยการค้นหาหรือ Re_{max} ตั้งไว้ที่ 5 ครั้งทั้งสามฟังก์ชัน และลำดับคำตอบย้อนหลังในรายการตามูเชิงเมื่อมีการเรียกใช้กลไกย้อนรอย หรือ k^{th} backward selection ก็ตั้งไว้ที่ลำดับที่ 5 ย้อนขึ้นไปจากคำตอบล่าสุดทั้งสามฟังก์ชันเช่นกัน ทั้งนี้เป็นไปตามคำแนะนำการตั้งค่าอัลกอริทึมการค้นหาแบบตามูเชิงปรับตัว (Sujitjorn et al., 2006) กลไกปรับรัศมีการค้นหาส่วนใหญ่จะใช้แบบสองระดับ ดังแสดงในตารางที่ 2.2

การตั้งเงื่อนไขยุติการค้นหาหรือ TC กำหนดให้มีค่าใกล้เคียงกับค่าฟังก์ชันวัตถุประสงค์ ณ คำตอบวงกว้างของฟังก์ชันนั้น ๆ ตัวอย่างกรณีของฟังก์ชัน BF ณ คำตอบวงกว้างจะให้ค่าฟังก์ชันวัตถุประสงค์ประมาณศูนย์จึงตั้ง $TC < 1 \times 10^{-9}$ ส่วนฟังก์ชัน RF ณ คำตอบวงกว้างจะให้ค่าฟังก์ชันวัตถุประสงค์ประมาณศูนย์เช่นเดียวกัน แต่เลือกตั้ง $TC < 1 \times 10^{-8}$ เพื่อให้ง่ายต่อการค้นหาและ

จำนวนรอบการค้นหาเฉลี่ยของ RF ไม่สูงมากนัก และของฟังก์ชัน SF ณ ค่าตอบวงกว้างจะให้ค่าฟังก์ชันวัตถุประสงค์ประมาณหนึ่ง แต่เลือกตั้ง $TC < 0.999$

การทดสอบดำเนินงานบนเครื่องคอมพิวเตอร์ตั้งโต๊ะ Intel Celeron(R) ความเร็ว 2.6 จิกกะเฮิร์ตซ์ ซึ่งมีหน่วยความจำ RAM ขนาด 238 เมกกะไบต์ หน่วยความจำ HDD ขนาด 30 จิกกะไบต์ การทดสอบแต่ละครั้งจะทำการสุ่มเลือกคำตอบเริ่มต้นภายในปริภูมิการค้นหาหนึ่งคำตอบแล้วปล่อยให้ ATS ทำการค้นหาจนพบเงื่อนไขยุติการค้นหา แล้วบันทึกค่าเวลาการค้นหากับจำนวนรอบการค้นหาที่ใช้ไป ทดสอบจำนวน 50 ครั้งแล้วหาค่าเฉลี่ย ได้ผลการทดสอบเป็นเวลาการค้นหาเฉลี่ย และจำนวนรอบการค้นหาเฉลี่ย แสดงอยู่ในตารางที่ 2.3 ซึ่งจากผลการทดสอบที่ได้รายงานนี้ทั้งหมดล้วน ได้จากเกณฑ์ยุติการค้นหาแบบพอใจ ค่าฟังก์ชันวัตถุประสงค์ไม่ได้เป็นแบบจำนวนรอบการค้นหาสูงสุด ซึ่งแสดงให้เห็นถึงคุณภาพคำตอบวงกว้างที่คืนนั่นเอง

ตารางที่ 2.3 ผลการทดสอบสมรรถนะของการค้นหาแบบตามเชิงปรับตัวที่เสนอ

	Search results	Test functions		
		BF	RF	SF
①	Search time (sec)	5.3940	8.2067	2.2837
②	Search rounds	556.06	759.80	108.22
①/②	(sec/round)	0.0097	0.0108	0.0211

แถวที่ 1 ของตารางที่ 2.3 แสดงค่าเวลาการค้นหาเฉลี่ยของฟังก์ชัน BF RF และ SF ตามลำดับ ส่วนค่าจำนวนรอบการค้นหาเฉลี่ยแสดงอยู่ในแถวที่ 2 เมื่อพิจารณาผลการทดสอบที่ได้ตามตารางที่ 2.3 พบว่าฟังก์ชัน SF จะใช้เวลาการค้นหาคำตอบวงกว้างน้อยที่สุด คือ 2.2837 วินาที ด้วยรอบการค้นหาเฉลี่ย 108.22 รอบ ถัดขึ้นไปเป็นฟังก์ชัน BF ใช้เวลาการค้นหาคำตอบวงกว้างที่ 5.3940 วินาที ด้วยรอบการค้นหาเฉลี่ย 556.06 รอบ และฟังก์ชันสุดท้าย RF ใช้เวลาการค้นหาคำตอบวงกว้างมากที่สุด คือ 8.2067 วินาที ด้วยรอบการค้นหาเฉลี่ย 759.80 รอบ

แถวสุดท้ายแถวที่ 3 ของตารางที่ 2.3 เป็นค่าที่ได้จากการคำนวณภายในแต่ละหลัก นั่นคือนำค่าเวลาการค้นหาเฉลี่ยจากแถวที่ 1 มารวด้วยค่าจำนวนรอบการค้นหาเฉลี่ยจากแถวที่ 2 ผลลัพธ์ที่ได้จะเป็นเวลาเฉลี่ยที่ใช้ในการค้นหาคำตอบแต่ละรอบการค้นหาของแต่ละฟังก์ชันทดสอบ ซึ่งแสดงให้เห็นระดับความยากง่ายของการประเมินฟังก์ชันแต่ละชนิด หากค่าดังกล่าวสูงแสดงว่าตัวประมวลผลต้องใช้เวลาประมวลผลมาก ในทางตรงกันข้ามหากค่าดังกล่าวน้อยย่อมแสดงให้เห็นว่าตัวประมวลผลใช้เวลาการประเมินค่าฟังก์ชันดังกล่าวรวดเร็ว นั่นเอง ถ้าพิจารณาตามแถวที่ 3 ของตารางที่ 2.3 ได้แก่ ฟังก์ชัน SF ใช้เวลาการค้นหาเฉลี่ย 0.0211 วินาที/รอบ ถัดไปเป็นฟังก์ชัน RF ด้วยเวลาการค้นหาเฉลี่ย 0.0108 วินาที/รอบ และฟังก์ชันที่ประเมินฟังก์ชันวัตถุประสงค์ง่ายที่สุดได้แก่ BF ใช้เวลาการค้นหาเฉลี่ย 0.0097 วินาที/รอบ

บทที่ 3

อัลกอริธึมการค้นหาแบบตามูเชิงปรับตัวหลายวิถี

3.1 กล่าวนำ

การค้นหาแบบตามูเชิงปรับตัวหลายวิถี (multi-path ATS หรือ MATS) ที่ได้ดำเนินการพัฒนาต่อออกนี้ หมายถึงการค้นหาค่าเหมาะที่สุดที่ดำเนินการบนคอมพิวเตอร์เครื่องเดียวมีซีพียูเพียงตัวเดียว โดยทำการแบ่งปริภูมิการค้นหาออกเป็นปริภูมิการค้นหาย่อย ด้วยกลไกแบ่งปริภูมิการค้นหา (partitioning mechanism หรือ PM) แล้วใช้การค้นหาแบบตามูเชิงปรับตัว (ATS) ทำการค้นหาคำตอบในแต่ละปริภูมิการค้นหาย่อยตามกลไกลำดับ (sequencing mechanism หรือ SM) โดยแต่ละลำดับจะทำการค้นหาในช่วงเวลาที่สอดคล้องกับการแบ่งเวลา (time sharing) เมื่อกระบวนการค้นหาคำเนินไประยะเวลาหนึ่งกลไกยกเลิกการค้นหา (discarding mechanism หรือ DM) จะทำการยกเลิกการค้นหาบางปริภูมิย่อย ที่ค่าฟังก์ชันวัตถุประสงค์ไม่ได้รับการปรับปรุงมานาน โดยคำตอบสุดท้ายก่อนการยกเลิกการค้นหายังคงเก็บไว้เพื่อป้องกันการย้อนกลับมาค้นหาซ้ำอีก (cycling) กลไกยกเลิกการค้นหาจะดำเนินไปจนกระทั่งเหลือเพียงหนึ่งวิถีการค้นหา เพื่อช่วยเพิ่มความเร็วของการค้นหาและเป็นการรักษากลยุทธ์ความเข้มข้นของการค้นหาแบบตามูไว้อีกด้วย

3.2 การค้นหาแบบตามูเชิงปรับตัวหลายวิถี (MATS)

อัลกอริธึมการค้นหาแบบตามูเชิงปรับตัวหลายวิถี หรือ MATS เป็นการพัฒนาต่อออกอัลกอริธึมการค้นหาแบบตามูเชิงปรับตัว (ATS) ซึ่งต่อไปจะเรียกว่า ATS แบบเดี่ยว ดังได้พบทวนไว้ก่อนหน้านี้แล้ว ซึ่งเป็นหนึ่งพัฒนาการของอัลกอริธึมการค้นหาแบบตามู (TS) การพัฒนา MATS ดำเนินการภายใต้โจทย์ที่ว่า จะทำอย่างไรให้อัลกอริธึมการค้นหาแบบตามูเชิงปรับตัวหลายหน่วยค้นหา ดำเนินงานในหลาย ๆ พื้นที่ของปริภูมิการค้นหา โดยมีเป้าหมายให้ใช้เวลาการค้นหาคำตอบวงกว้างลดลงเมื่อเปรียบเทียบกับ ATS แบบเดี่ยว ภายใต้เงื่อนไขว่าต้องดำเนินการได้ด้วยตัวประมวลผลเดี่ยว ผู้อ่านอาจสงสัยถึงเหตุผลที่ต้องตั้งเงื่อนไขดังกล่าว ทั้งนี้เพื่อความสะดวกในการประยุกต์ใช้อัลกอริธึมนี้กับเครื่องคอมพิวเตอร์พกพาหรือเครื่องคอมพิวเตอร์ตั้งโต๊ะแบบตัวประมวลผลเดี่ยว ซึ่งมีการใช้งานกันอยู่แพร่หลายนั่นเอง การตอบ โจทย์ข้อนี้ผู้วิจัยอาศัยกลยุทธ์ความร่วมมือในการค้นหาคำตอบวงกว้างของหลาย ๆ หน่วยค้นหา ภายใต้การทำงานของตัวประมวลผลเดี่ยวตามวิธีการแบ่งเวลา (time sharing) ทำให้เกิดจำนวนเส้นทางการค้นหาของ ATS ย่อย ๆ จำนวนมากกว่าหนึ่งเส้นทางการค้นหา ขณะที่ ATS แบบเดี่ยว มีเพียงหนึ่งเส้นทางการค้นหาการทำงานของ ATS ย่อย ของ MATS ทำให้เวลาที่ใช้ดำเนินงานตามเส้นทางการค้นหานั้นมากกว่า ATS แบบเดี่ยว จากการที่ต้องคอยเวลาซึ่งกันและกันของ ATS ย่อย แต่ประโยชน์ที่เกิดขึ้นภายใต้

MATS เป็นการเพิ่มความหลากหลายในการสำรวจปริภูมิการค้นหา อันถือเป็นกลยุทธ์ที่สำคัญแบบหนึ่งในการเพิ่มโอกาสการพบคำตอบคุณภาพสูง กลยุทธ์ดังกล่าวมีการเรียกขานว่ากลยุทธ์ความหลากหลาย (diversification strategy) หรือ กลยุทธ์เชิงสำรวจ (exploration strategy) แต่การทำให้ MATS ใช้เวลาการค้นหารวมสั้นกว่า ATS แบบเดียว จำเป็นต้องมีกลยุทธ์ทางการบริหารจัดการที่ดี ดังที่จะได้อธิบายต่อไป

SM ทำหน้าที่ลำดับการค้นหาให้กับแต่ละหน่วยค้นหา ทำให้การค้นหาทั้งหมดต้องอยู่ภายใต้การประมวลผลเชิงลำดับ และ DM ทำหน้าที่ประเมินโอกาสหรือแนวโน้มที่ ATS ย่อย แต่ละหน่วยจะพบคำตอบคุณภาพสูง และสั่งยกเลิก ATS ย่อย หน่วยใดที่ถูกประเมินว่ามีโอกาสน้อยที่จะพบคำตอบที่มีคุณภาพตามที่ต้องการ

The generic MATS algorithms can be summarized as follows:

STEP 0: Initialization (PM, SM and ATS)
Decompose the search domain, initialize all ATS paths.
Release all search paths to perform freely on the original search space.

STEP 1: Iteration. (SM and ATS)
For a parallel processing platform, arrange ATS paths in multi-threading manner, otherwise in sequential manner. Perform search using ATS.

STEP 2: Aspiration (DM)
If the discarding criterion (DC) is met, decrease the corresponding ATS path.

STEP 3: Termination (ATS)
Exit with the global optimum solution if the TC is (are) met, otherwise go to STEP 1.

รูปที่ 3.1 ขั้นตอนการทำงานของ MATS

การทำงานของ DM อย่างมีประสิทธิภาพซึ่งขึ้นอยู่กับกรอบการออกแบบด้วย จะส่งผลให้เวลาการค้นหารวมไม่ยาวนาน และคาดหวังว่าจะใช้เวลาสั้นกว่า ATS แบบเดียว DM ทำหน้าที่ตัดทอนเส้นทางการค้นหาของ ATS ย่อย ไปเรื่อย ๆ จนเหลือเพียงเส้นทางเดียวที่ทำการค้นหาต่อไปจนพบคำตอบกว้างแต่แนวทางของ DM มิได้จำกัดแต่เพียงเท่านั้น ในบางสถานการณ์ของการประยุกต์การตัดทอนจำนวนเส้นทางการค้นหาของ ATS การคงค้างให้มีเส้นทางของ ATS เหลืออยู่มากกว่า 1 เส้นทาง เพื่อแข่งขันกันค้นหาคำตอบ อาจส่งผลที่ได้ต่อการพบคำตอบอย่างรวดเร็ว MATS อาจได้รับการเขียนแสดงเป็นขั้นตอนโดยย่อได้ดังรูปที่ 3.1 เกณฑ์ยุติการค้นหา หรือ TC มีสองชนิด ได้แก่ ชนิดที่หนึ่งใช้จำนวนรอบการค้นหาสูงสุด หรือ count_max จะยุติการค้นหาของ ATS ย่อย ทุกตัวในการกำกับของ MATS เมื่อสิ้นสุดการคำนวณในรอบการค้นหา count_max ส่วน TC ชนิดที่สอง คือ คำตอบที่ได้ของ ATS ย่อย ตัวใดตัวหนึ่งอยู่ในเกณฑ์ที่ยอมรับได้ MATS ก็จะสั่ง ATS ย่อย ทุกตัวในการกำกับการค้นหาลง TC ชนิดนี้เป็นคุณสมบัติติดตัวของ ATS แบบเดียว อยู่แล้ว และยังคง

ใช้การได้กับ ATSy่อย รวมถึง MATS อีกด้วย นั่นหมายความว่า ATSy่อย ทุกตัวในกำกับของ MATS สามารถยุติการค้นหารวมของ MATS ได้ทันทีที่เงื่อนไขนี้ถูกพบโดยไม่จำเป็นต้องรอให้สิ้นสุดรอบการค้นหารวมของ MATS และไม่จำกัดว่าการยุติการค้นหาคือต้องเหลือเพียงหน่วยค้นหาเดียว ดังนั้นเกณฑ์ยุติการค้นหาหรือ TC ชนิดที่สอง จึงมีบทบาทสำคัญอย่างยิ่งของ MATS ในการลดเวลาการค้นหารวมและรักษาคุณภาพคำตอบที่พบไว้ได้ เนื่องจาก ATSy่อย ทุกตัวสามารถยุติการค้นหาของ MATS ได้ทุกเมื่อที่พบคำตอบคุณภาพสูงตามที่ต้องการ ผลการทดสอบที่จะได้กล่าวถึงรายละเอียดในหัวข้อถัดไปนั้น ก็ล้วนได้จาก TC ชนิดที่สองนี้

3.2.1 กลไกแบ่งปริภูมิการค้นหา (PM)

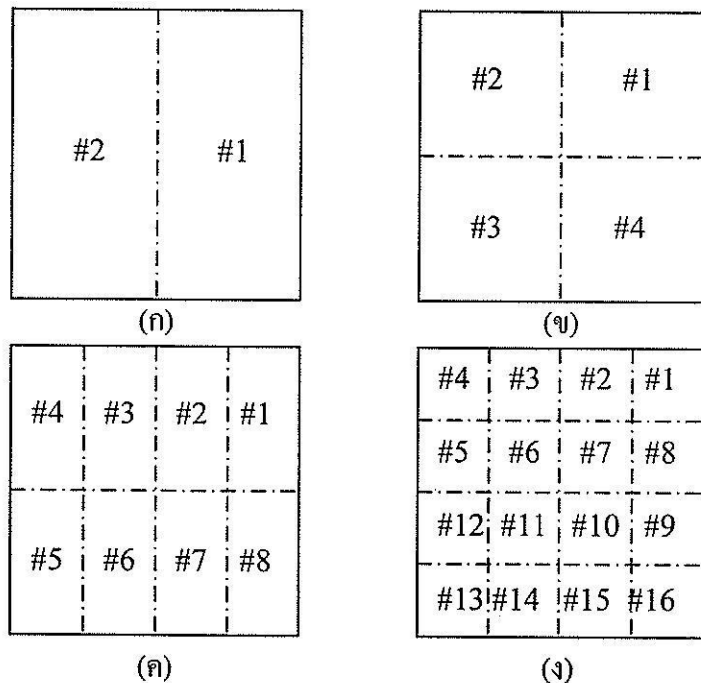
กลไกแบ่งปริภูมิการค้นหา หรือ PM ให้การกระจายคำตอบเริ่มต้นของแต่ละหน่วยค้นหาให้ทั่วถึงทั้งปริภูมิการค้นหา เพื่อเพิ่มโอกาสในการพบปริภูมีย่อยที่มีคำตอบวงกว้างอยู่ภายใน PM เริ่มต้นด้วยการแบ่งปริภูมิการค้นหาทั้งหมดออกเป็นปริภูมิการค้นหาย่อย ๆ จากนั้นจึงสร้างคำตอบเริ่มต้นที่ไม่ซ้ำกันให้แก่แต่ละปริภูมีย่อย ขั้นตอนการทำงานของ PM แสดงไว้ในรูปที่ 3.2

PM procedures	
STEP 1: Loading parameters	Load number of search paths, N , and original search space.
STEP 2: Partitioning	Partition the search space into N sub-search-spaces.
STEP 3: Initial solution generating	Generate initial solutions for each sub-search-space and remove partitions.

รูปที่ 3.2 ขั้นตอนการทำงานของ PM

รูปแบบหนึ่งที่เป็นไปได้ของ PM กับปัญหาแบบสองมิติที่ปริภูมิการค้นหามีรูปทรงสี่เหลี่ยม การแบ่งปริภูมิการค้นหาสามารถทำได้โดยใช้เส้นแนว (grid) ทั้งแบบแนวตั้ง (vertical grid) และแบบแนวนอน (horizontal grid) หรือจะใช้สองแบบรวมกันก็ได้ ดังรูปที่ 3.3 รูปที่ 3.3 (ก) แสดงการแบ่งปริภูมิการค้นหาออกเป็นสองส่วนเท่า ๆ กัน โดยใช้เส้นแนวตั้งแบ่งออกเป็นปริภูมิการค้นหาย่อยทางขวาให้ชื่อว่า ปริภูมิการค้นหาย่อยที่ 1 หรือ #1 และปริภูมิการค้นหาย่อยทางซ้ายมือคือปริภูมิการค้นหาย่อยที่ 2 หรือ #2 การแบ่งแบบสองส่วนนี้สามารถทำได้ด้วยวิธีการอื่น เช่น ใช้เส้นแนวนอนแบ่งเป็นส่วนบนกับส่วนล่าง หรือจะใช้เส้นแนวทแยงมุมจากมุมบนซ้ายลงมามุมล่างขวาได้ส่วนบนขวากับส่วนล่างซ้าย เป็นต้น รูปที่ 3.3 (ข) แสดงตัวอย่างการแบ่งปริภูมิการค้นหาออกเป็นสี่ส่วนเท่า ๆ กัน ด้วยการเพิ่มเติมเส้นแนวนอนให้กับรูปที่ 3.3 (ก) ก็จะได้ปริภูมิการค้นหาย่อยจำนวนสี่ส่วนเท่า ๆ กันตามต้องการ หากเพิ่มเส้นแนวตั้งจำนวนสองเส้นทางด้านซ้ายและขวาให้กับรูปที่ 3.3 (ข) จะได้ปริภูมิการค้นหาย่อยจำนวนแปดส่วนเท่า ๆ กัน ดังแสดงในรูปที่ 3.3 (ค)

และ รูปที่ 3.3 (ง) แสดงตัวอย่างการแบ่งปริภูมิการค้นหาออกเป็นสิบหกส่วนเท่า ๆ กัน ตามแบบที่ได้อธิบายข้างต้น อย่างไรก็ตามวิธีการแบ่งปริภูมิการค้นหาดังเดิมออกเป็นปริภูมิย่อย มิได้จำกัดแค่เพียงวิธีที่อธิบาย อาจแบ่งด้วยวิธีการอื่นที่เหมาะสมกับปัญหาได้ ภายหลังจากการแบ่งปริภูมิการค้นหาขั้นต่อไป PM สร้างคำตอบเริ่มต้นด้วยวิธีการสุ่มคำตอบภายในแต่ละปริภูมิย่อย และถอดถอนสภาพปริภูมิย่อยออก ดังนั้น ATSy่อย แต่ละหน่วยจึงมีอิสระในการเคลื่อนตัวตามวิถีการค้นหาไปทั่วปริภูมิดั้งเดิม ATSy่อย ที่ถูกจัดตั้งในปริภูมิย่อยที่มีคำตอบวงกว้าง หรืออาจเป็นปริภูมิย่อยที่อยู่ใกล้ชิดกับคำตอบวงกว้างก็จะสามารถเข้าถึงคำตอบวงกว้างได้รวดเร็ว อย่างไรก็ตามการสร้างคำตอบเริ่มต้นมิได้จำกัดแค่เพียงการสุ่ม ผู้ใช้สามารถเลือกใช้วิธีวิสัยสามัญ (heuristic) กำหนดคำตอบเริ่มต้นให้เหมาะสมกับปัญหาได้

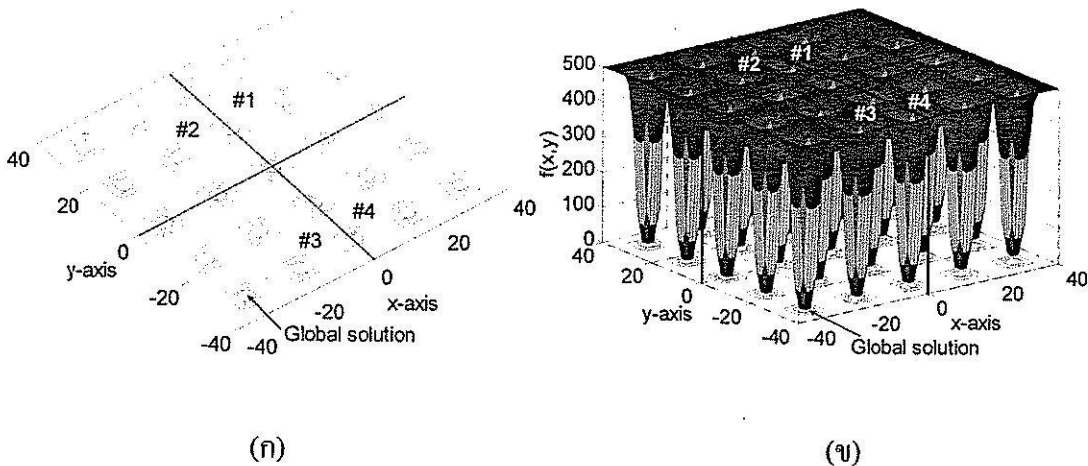


รูปที่ 3.3 ตัวอย่างการแบ่งปริภูมิการค้นหาและการตั้งชื่อเส้นทางการค้นหา

(ก) สองเส้นทาง (ข) สี่เส้นทาง (ค) แปดเส้นทาง และ(ง) สิบหกเส้นทาง

ภาพสองมิติและสามมิติของการแบ่งปริภูมิการค้นหาออกเป็นสี่ส่วนของฟังก์ชันเชลคฟอกซ์โฮล (shekel's foxholes function หรือ SF) แสดงดังรูปที่ 3.4 เป็นตัวอย่างการทำงานตามกลไก PM ปริภูมิ $SF \pm 40$ ทั้งแกน x และ y ดังรูปที่ 3.4 (ก) ที่อาจเขียนในรูปพิกัดของปริภูมิการค้นหาว่า [ขอบเขตบนตัวแปรที่ 1 ขอบเขตบนตัวแปรที่ 2 ; ขอบเขตล่างตัวแปรที่ 1 ขอบเขตล่างตัวแปรที่ 2] นั่นคือ $[40 \ 40; -40 \ -40]$ สมมุติว่า ต้องการแบ่งปริภูมิ SF ออกเป็นสี่ปริภูมิการค้นหาย่อยจึงทำการแบ่งด้วยเส้นแนวนอนและเส้นแนวตั้งอย่างละหนึ่งเส้น อยู่กึ่งกลางของทั้งสองแกน เส้นแบ่งจึงตัดกันเป็นมุมฉากที่จุด $(0,0)$ จึงเกิดเป็นสี่ปริภูมิการค้นหาย่อยที่มีพื้นที่เท่า ๆ กัน ปรากฏบนภาพ

นายสองมิติดังรูปที่ 3.4 (ก) และกำหนดชื่อเป็นปริภูมิการค้นหาย่อยที่ 1 (#1) ปริภูมิการค้นหาย่อยที่ 2 (#2) ปริภูมิการค้นหาย่อยที่ 3 (#3) และปริภูมิการค้นหาย่อยที่ 4 (#4) ที่มีขอบเขตเป็น $[40 \ 40; 0 \ 0]$ $[0 \ 40; -40 \ 0]$ $[0 \ 0; -40 \ -40]$ และ $[40 \ 0; 0 \ -40]$ สำหรับปริภูมิย่อย #1 #2 #3 และ #4 ตามลำดับ และอาจสังเกตเห็นว่าคำตอบวงกว้างปรากฏอยู่ใน #3 กระบวนการสุ่มเลือกคำตอบเริ่มต้นของ PM กรณีของ #3 มีโอกาสมากที่คำตอบเริ่มต้นนั้นจะเป็นคำตอบที่ใกล้เคียงกับคำตอบวงกว้างหรืออาจเป็นคำตอบวงกว้างเลยก็เป็นไปได้ นั่นหมายความว่ายิ่งเรามีจำนวนหน่วยค้นหาเพิ่มขึ้นปริภูมิการค้นหา ก็จะยิ่งถูกแบ่งมากขึ้นตามไปด้วย ทำให้คำตอบวงกว้างมีโอกาสจะถูกคัดเลือกให้เป็นคำตอบเริ่มต้นของปริภูมิการค้นหาย่อยหนึ่งนั้นมีมากขึ้นตามไปด้วย เท่ากับเป็นผลดีต่อ MATS นั้นเอง ถ้าพิจารณาการแบ่งปริภูมิตามรูปที่ 3.3 โดยสมมติให้เป็นปริภูมิของ SF ดังรูปที่ 3.4 กรณีที่จำนวนหน่วยค้นหาเป็น 2 พิจารณาที่รูป 3.3 (ก) พบว่าคำตอบวงกว้างอยู่ใน #2 ถ้าจำนวนหน่วยค้นหาเพิ่มเป็น 4 พิจารณาที่รูป 3.3 (ข) พบว่าคำตอบวงกว้างอยู่ใน #3 เช่นเดิม ถ้าจำนวนหน่วยค้นหาเพิ่มเป็น 8 พิจารณาที่รูป 3.3 (ค) พบว่าคำตอบวงกว้างอยู่ใน #5 และกรณีสุดท้ายเมื่อจำนวนหน่วยค้นหาเพิ่มเป็น 16 พิจารณาที่รูป 3.3 (ง) พบว่าคำตอบวงกว้างอยู่ใน #13 ถ้าเรียงตามขนาดปริภูมิย่อยที่บรรจุคำตอบวงกว้างไว้จากใหญ่ไปหาเล็ก จะเรียงได้ดังนี้ #2 #3 #5 และ #13 ซึ่งเป็นของการแบ่งปริภูมิการค้นหาแบบ 2 4 8 และ 16 หน่วยค้นหาตามลำดับ แสดงให้เห็นว่ายิ่งจำนวนหน่วยค้นหาเพิ่มขึ้นขนาดของปริภูมิย่อยที่มีคำตอบวงกว้างอยู่ก็ยิ่งเล็กลง ทำให้มีโอกาสได้คำตอบเริ่มต้นเป็นคำตอบวงกว้างหรือค่าใกล้เคียงคำตอบวงกว้างมากยิ่งขึ้นนั่นเอง



รูปที่ 3.4 การแบ่งปริภูมิการค้นหาเป็นส่วนๆ ของฟังก์ชันซีเคิลฟอกซ์โฮล
(ก) แผนที่เส้นระดับสองมิติ และ (ข) แนวการแบ่งปริภูมิย่อยบนพื้นผิว

อย่างไรก็ตาม การเพิ่มจำนวนหน่วยค้นหาให้มากขึ้นไปก็มีขอบเขตจำกัด ตัวอย่างปัญหาการหาค่าเหมาะที่สุดแบบสองมิติที่มีลักษณะปริภูมิค้นหาเป็นรูปสี่เหลี่ยม ดังรูปที่ 3.3 จำนวนหน่วยค้นหาสูงสุดที่จะเป็นไปได้สำหรับปัญหานี้จะเป็น ดังสมการที่ (3.1)

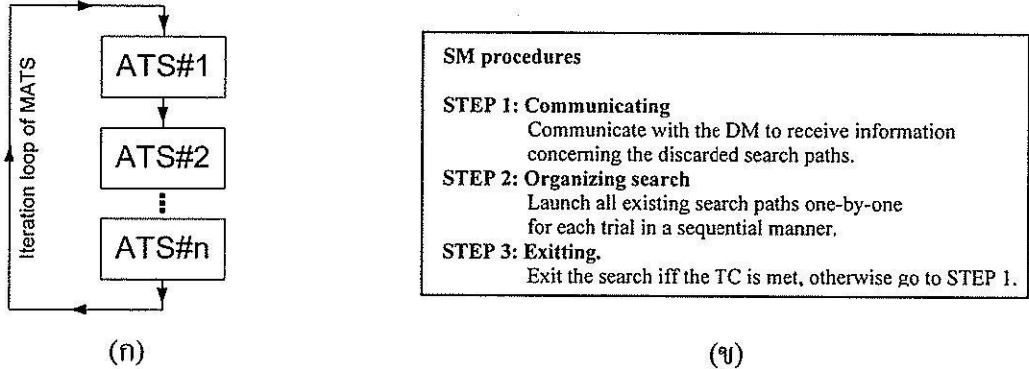
$$N_{\max} = \left(\frac{l}{2R} \right)^2 \quad (3.1)$$

โดยที่ N_{\max} คือ จำนวนหน่วยค้นหาสูงสุดที่แบ่งได้ l คือ ความยาวด้านหนึ่งของปริภูมิค้นหาที่เป็นสี่เหลี่ยมจัตุรัส มีหน่วยเป็น เมตร (m) และรัศมีการค้นหา R ของ ATS มีหน่วยเป็น เมตร (m) ผู้อ่านพึงระลึกไว้เสมอว่า ATS จะคัดเลือกค่าใกล้เคียงภายในวงกลมรัศมี R โดยมีคำตอบปัจจุบันทำหน้าที่เป็นจุดศูนย์กลางวงกลม ตัวอย่างปริภูมิปัญหาหนึ่งเป็นสี่เหลี่ยมจัตุรัสขนาด 30 ซม. \times 30 ซม. และ ATS มีรัศมีการค้นหา $R = 1.5$ ซม. ดังนั้น จะได้ N_{\max} เท่ากับ 100 การเพิ่มจำนวนหน่วยค้นหาที่สูงกว่า N_{\max} ย่อมไม่ก่อให้เกิดประโยชน์ต่อ MATS แต่อย่างไร

3.2.2 กลไกลำดับการค้นหา (SM)

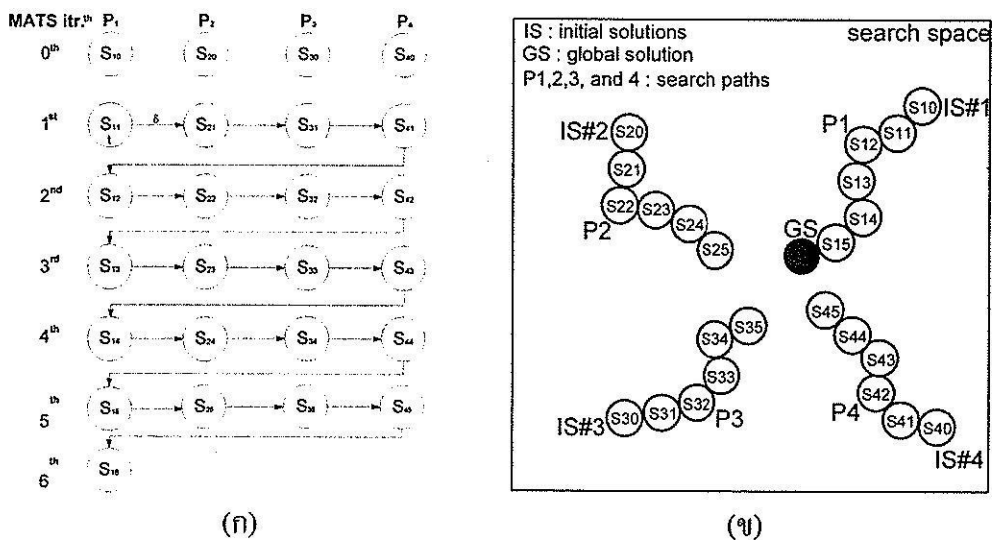
กลไกลำดับการค้นหา - sequencing mechanism (SM) ทำหน้าที่สร้างลำดับการทำงานของ ATSย่อย แต่ละเส้นทาง ให้กับตัวประมวลผลดำเนินการประมวลผลค่าต่าง ๆ ตามคำสั่งภายในของ ATSย่อย นั้น ๆ รูปแบบการจัดสรรเวลาสำหรับการประมวลผลให้กับ ATSย่อย มีได้หลายรูปแบบ เช่น รูปแบบที่หนึ่ง มีการจัดสรรจำนวนรอบการค้นหาครั้งที่เท่ากันทุกหน่วยค้นหาตลอดการค้นหาของ MATS ถือได้ว่าเป็นแบบตายตัว (static) และรูปแบบที่สองถือเป็นแบบเคลื่อนไหว (dynamic) มีการจัดสรรจำนวนรอบการค้นหาไม่คงที่ขึ้นกับสถานการณ์การค้นหาของแต่ละ ATSย่อย สำหรับงานวิจัยนี้เลือกศึกษา SM ในรูปแบบที่หนึ่งด้วยการตั้งค่าจำนวนรอบการค้นหาสำหรับแต่ละ ATSย่อย แบบหนึ่งรอบการค้นหา เปรียบเทียบความแตกต่างกับกรณีของ ATSแบบเดี่ยว ที่ไม่จำเป็นต้องใช้ SM เนื่องจากตัวประมวลผลสามารถดำเนินการสร้างลำดับคำตอบให้กับ ATSแบบเดี่ยว ได้อย่างต่อเนื่องปราศจากการถูกขัดจังหวะเพื่อให้ไปทำงานอย่างอื่น เพราะมีเพียงเส้นทางการค้นหาเดียว การค้นหาตามหลักการแบ่งเวลา (time sharing) ของ SM แสดงดังรูปที่ 3.5 (ก) แสดงลักษณะการทำงานของ ATSย่อย ใน MATS เมื่อ ATS#1 ดำเนินการค้นหาเสร็จสิ้นไปหนึ่งรอบการค้นหาแล้ว ก็จะหยุดเพื่อรอคิวการค้นหาในรอบถัดไป ช่วงเวลาถัดไป SM ตั้งให้ ATS#2 ดำเนินการจนเสร็จกระบวนการค้นหาหนึ่งรอบ และสั่งให้หยุดรอคิวรอบการค้นหาถัดไป ต่อจากนั้นจึงจัดคิวการทำงานของ ATS#3 ATS#4 จนถึง ATS#n ในลักษณะเช่นนี้ไปเรื่อย ๆ จากนั้นจึงเริ่มต้นที่ ATS#1 ใหม่ ตามด้วย ATS#2 ต่อไปจนถึง ATS#n ตามลำดับจนครบรอบการค้นหาที่ 2 ของ MATS รอบการค้นหาของ MATS จะวนซ้ำเช่นนี้จนกระทั่ง เมื่อมีการเรียกใช้กลไก

ยกเลิกการค้นหาหรือ DM จะทำให้จำนวนของหน่วยค้นหาในวงรอบการค้นหาของ MATS ลดลง ทำให้ SM ต้องตรวจสอบจำนวนหน่วยค้นหาที่เหลืออยู่ ต่อจากนั้นจึงเลื่อนลำดับช่วงเวลาการค้นหาของ ATSy่อย ที่อยู่ถัดไปขึ้นมาแทนที่หน่วยค้นหาซึ่งถูกยกเลิกการค้นหาจาก DM ไปแล้ว ขั้นตอนการทำงานของ SM สามารถแจกแจงรายละเอียดได้ดังรูปที่ 3.5 (ข)



รูปที่ 3.5 ลักษณะการทำงานของ SM (ก) ทำงานแบบแบ่งเวลาในวงรอบการค้นหาของ MATS และ (ข) ขั้นตอนการทำงานของ SM

เพื่อความเข้าใจถึงหลักการและวิธีการทำงานของ SM ใน MATS มากขึ้น จึงขออธิบายด้วยตัวอย่าง MATS แบบ 4 เส้นทางค้นหาหรือ MATS#4 ซึ่งประกอบด้วย 4 ATSy่อย เมื่อ PM จัดหาคำตอบเริ่มต้น ได้แก่ IS#1=S10 IS#2=S20 IS#3=S30 และ IS#4=S40 สำหรับ ATs#1 ATs#2 ATs#3 และ ATs#4 ตามลำดับ ดังแสดงในรูปที่ 3.6 (ก) เรียบร้อยแล้วนั้น ATSy่อย แต่ละเส้นทางก็จะเริ่มสร้างลำดับคำตอบของตัวเองต่อไปอย่างอิสระภายใต้การกำกับ หรือให้จังหวะการทำงานของ SM



รูปที่ 3.6 ตัวอย่างการทำงานของ SM ใน MATSแบบ 4 เส้นทางค้นหาที่ยังไม่มี DM (ก) จังหวะเวลา (timing) ของการเกิดคำตอบทั้งหมดของ MATS และ (ข) ลำดับคำตอบของแต่ละ ATSy่อย บนปริภูมิการค้นหา

พิจารณารูปที่ 3.6 (ก) เส้นทางการค้นหาของ ATS#1 หรือ P_1 หมายถึงเส้นที่เชื่อมต่อกิจักบนปริภูมิการค้นหาของคำตอบที่อยู่ติดกันทั้งหมดของ ATS#1 ใน P_1 ประกอบด้วยคำตอบที่พบโดย ATS#1 เรียงกันตามลำดับเวลา ตั้งแต่เริ่มต้นจนถึงสิ้นสุดการค้นหา ตัวอย่างลำดับคำตอบ ตามรูปที่ 3.6 (ก) ของ P_1 ทำการสร้างลำดับคำตอบภายในวงรอบการค้นหาของ MATS อยู่ทั้งหมด 6 รอบการค้นหา ได้คำตอบในแนวคอลัมน์ที่ 1 ด้านซ้ายสุดแสดงได้ดังนี้ $\{S_{10}, S_{11}, S_{12}, S_{13}, S_{14}, S_{15}, S_{16}\}$ ซึ่ง S_{10} คือ คำตอบเริ่มต้น ตามด้วย S_{11} เป็นคำตอบถัดไปซึ่งเกิดขึ้นในรอบการค้นหาที่ 1 ของ MATS แล้วก็เป็น S_{12} ไปเรื่อย ๆ จนกระทั่งได้คำตอบสุดท้ายของ P_1 คือ S_{16} ซึ่งได้ถูกสมมุติให้เป็นคำตอบวงกว้าง (global solution หรือ GS) ของตัวอย่างนี้ และได้สมมุติให้เส้นทางการค้นหา P_1 เป็นตัวพบคำตอบวงกว้าง (ดูรูปที่ 3.6 ประกอบ) ลำดับคำตอบของ P_2 พิจารณาคอลัมน์ที่ 2 จากซ้ายในรูปที่ 3.6 (ก) เรียงจากบนลงล่าง เห็นได้ว่าลำดับคำตอบของ P_2 คือ $\{S_{20}, S_{21}, S_{22}, S_{23}, S_{24}, S_{25}\}$ มี S_{20} คือ คำตอบเริ่มต้น แล้วตามด้วย S_{21} ไปเรื่อย ๆ จนได้ S_{25} เป็นคำตอบสุดท้ายของ P_2 เมื่อพิจารณาการทำงานของอัลกอริทึมร่วมกับความเข้าใจในกลไก SM อาจเห็นว่า ช่วงเริ่มต้นการค้นหาของ MATS มี ATSย่อย อยู่ครบทั้งสี่ตัว ($n=4$ สำหรับตัวอย่างนี้) โดยหลังจากที่ PM จัดสรรคำตอบเริ่มต้นให้กับ ATSย่อย จนครบทุกตัวเรียงลำดับจากซ้ายไปขวาตั้งแต่ P_1 จนถึง P_4 ทำให้แต่ละเส้นทางการค้นหาได้คำตอบค้นหาเรียบร้อยแล้ว ดังรูปที่ 3.6 (ก) แถวที่ 1 หรือ วงรอบการค้นหาที่ศูนย์ของ MATS นั้นเอง เมื่อขึ้นวงรอบการค้นหาที่ 1 ของ MATS SM จะอนุญาตให้ ATS#1 ดำเนินการค้นหาจนได้คำตอบมาหนึ่งค่าที่เป็นคำตอบดีที่สุดในวงรอบการค้นหาขึ้น ก็จะสั่งให้ ATS#1 หยุดรอ แล้วจึงอนุญาตให้ ATS#2 ทำการค้นหาคำตอบของตัวเองอีกหนึ่งรอบ จากนั้นสั่งให้ ATS#2 หยุดรออีก แล้วจึงค่อยอนุญาตให้ ATS#3 ทำการค้นหาคำตอบอีกหนึ่งรอบ จากนั้นสั่งให้ ATS#3 หยุดรออีก แล้วจึงค่อยอนุญาตให้ ATS#4 ทำการค้นหาคำตอบของตัวเองอีกหนึ่งรอบ จากนั้นสั่งให้ ATS#4 หยุดรออีกเป็นอันสิ้นสุดวงรอบการค้นหาที่ 1 ของ MATS เมื่อ MATS ขึ้นวงรอบการค้นหาที่ 2 SM จะวนกลับไปอนุญาตให้ ATS#1 ทำการค้นหาคำตอบถัดไปต่ออีกหนึ่งรอบจนได้ S_{12} แล้วตามด้วย ATS#2 ATS#3 และ ATS#4 ตามลำดับ ก่อนที่วนกลับไปเริ่มต้น ATS#1 เพื่อขึ้นรอบการค้นหาของ MATS รอบใหม่ เป็นเช่นนี้ ไปจนถึงรอบการค้นหาที่ 6 ATS#1 จึงพบกับคำตอบวงกว้างหรือค่าใกล้เคียงคำตอบวงกว้าง ซึ่งสอดคล้องกับหนึ่งในเงื่อนไขยุติการค้นหา การดำเนินงานในลักษณะดังกล่าวนี้เป็นไปตามหลักการแบ่งเวลาดำเนินงาน ซึ่งเป็นวิธีการพื้นฐานสำหรับระบบที่มีตัวประมวลผลเพียงตัวเดียว

3.2.3 กลไกยกเลิกการค้นหา (DM)

DM ช่วยกำจัดบางหน่วยค้นหาที่ถูกประเมินสภาพว่ามีแนวโน้มจะพบคำตอบวงกว้างได้น้อยกว่าหน่วยค้นหาอื่น เช่น บางหน่วยค้นหาอาจเกิดการล็อกโดยคำตอบวงแคบเฉพาะถิ่น ซึ่งจะ

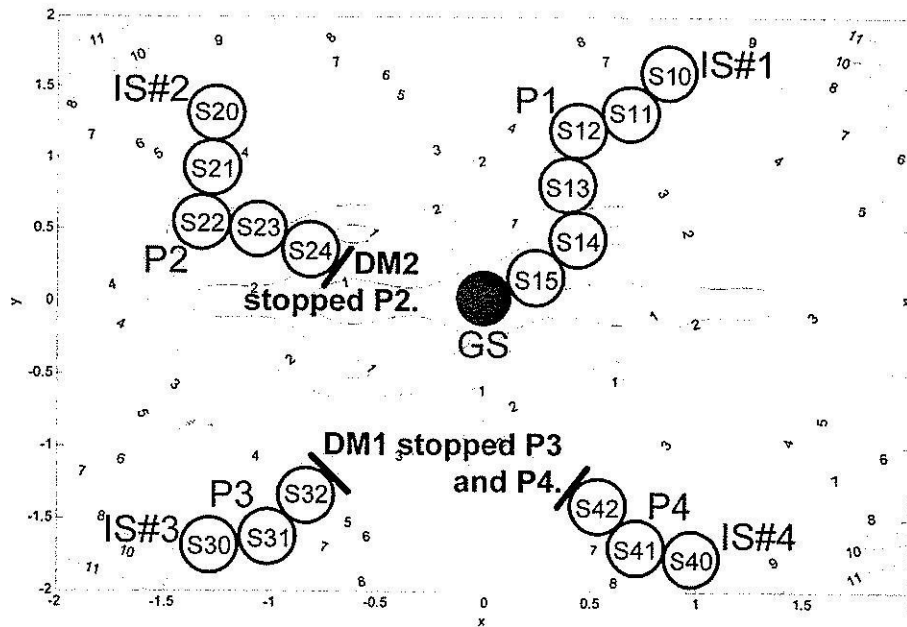
ส่งผลให้การลู่เข้าหาผลเฉลยของเส้นทางอื่นอาจเข้าไปด้วย เป็นต้น ขั้นตอนการทำงานของ DM แสดงได้ในรูปที่ 3.7

STEP 1: Initializing	Load cost values of the current best-solutions of all active search paths.
STEP 2: Sorting	Do min-max sorting of the cost values.
STEP 3: Discarding.	Keep the search paths from the top to the middle of the sorted list as active, terminate the rest of the search paths.
STEP 4: Transferring	Transfer information and search control to the SM.

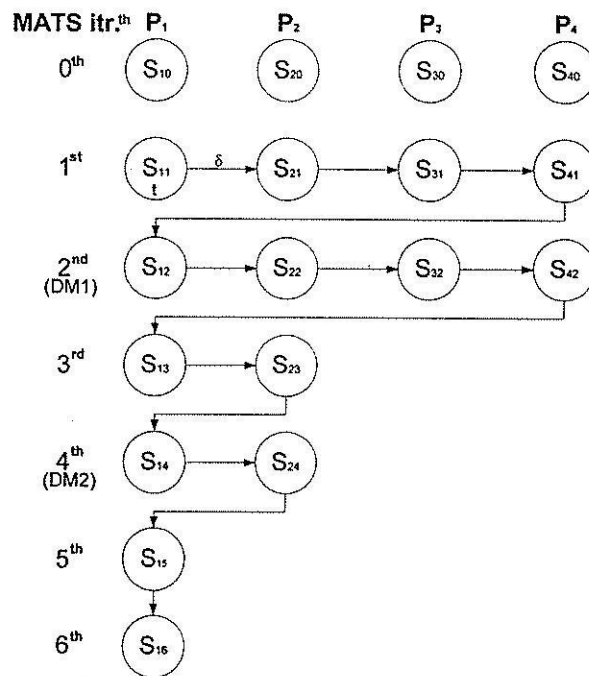
รูปที่ 3.7 ขั้นตอนการทำงานของกลไกยกเลิกการค้นหา

MATS เรียกใช้งาน DM ตามเกณฑ์จำนวนรอบการค้นหาที่กำหนดไว้ล่วงหน้า และสามารถเรียกใช้ DM ได้มากกว่าหนึ่งครั้ง การตัดทอนจำนวนหน่วยค้นหาให้น้อยลงเป็นความอ่อนตัวที่ผู้ใช้สามารถปรับแต่งได้ ซึ่งอาจไม่จำเป็นต้องลดลงครึ่งหนึ่งของจำนวนเดิมดังที่นำเสนอในที่นี้ เมื่อมีการเรียกใช้ DM ขั้นตอนที่ 1 DM จะนำค่าฟังก์ชันวัตถุประสงค์ของ ATSy่อย แต่ละตัวที่ยังมีสิทธิ์ทำงานอยู่ในวงรอบการค้นหาของ MATS มาเก็บไว้ โดยสิทธิ์ดังกล่าวถูกกำหนดด้วยค่าสถานะของ flag นั่นคือ ถ้าสถานะของ flag = 1 หมายถึง ATSy่อย นั้นมีสิทธิ์ทำงานใน MATS ในทางตรงกันข้าม ถ้าสถานะของ flag = 0 หมายถึง ATSy่อย นั้นหมดสิทธิ์ทำงานใน MATS นั่นเอง DM ทำการเรียงลำดับจากค่าฟังก์ชันวัตถุประสงค์น้อยไปหาค่าฟังก์ชันวัตถุประสงค์มาก (สำหรับปัญหาการหาค่าต่ำสุด) และยกเลิกสิทธิ์การทำงานในวงรอบการค้นหาของ MATS ให้กับหน่วยค้นหาที่มีค่าฟังก์ชันวัตถุประสงค์มากที่สุดและมากรองลงมาตามลำดับเป็นจำนวนหนึ่ง ซึ่งอาจเลือกการลดจำนวนหน่วยค้นหาลงทีละกึ่งหนึ่งของจำนวนหน่วยค้นหาที่มีอยู่ขณะนั้นก็ได้ ส่วนหน่วยค้นหาที่เหลือยังคงมีสิทธิ์ดำเนินการค้นหาได้ต่อไปในวงรอบการค้นหาของ MATS การเรียกใช้ DM ครั้งที่สอง หรือ DM2 สามารถทำได้หากจำนวนหน่วยค้นหาที่เหลืออยู่มากกว่าหนึ่งเส้นทาง โดยต้องกำหนดค่ารอบการค้นหาที่เรียกใช้ DM2 ให้มากกว่ารอบของการเรียกใช้ DM1 นั่นเอง ส่วนจะมากกว่าเท่าไรก็ขึ้นอยู่กับความเหมาะสมของแต่ละปัญหา

เพื่อให้เข้าใจการทำงานของ DM ได้ดียิ่งขึ้น ต่อจากนี้ให้พิจารณารูปที่ 3.8 ซึ่งแสดงเส้นทางการค้นหาทั้งสี่เส้นทางของ MATS บนพื้นผิวของฟังก์ชัน BF ที่มีเส้นระดับ (contour) พร้อมตัวเลขกำกับเพื่อความชัดเจนยิ่งขึ้น เส้นทางค้นหาดังกล่าวเกิดขึ้นจากการวางตัวของคำตอบที่พบในแต่ละรอบการค้นหาของ ATSy่อย ช่วงเริ่มต้นการค้นหาหรือรอบการค้นหาที่ศูนย์ของ MATS PM จัดสรรคำตอบเริ่มต้นสำหรับ ATSy่อย #1 คือ S_{10} มีค่าฟังก์ชันวัตถุประสงค์ประมาณ 7 ของ ATSy่อย #2 คือ S_{20} มีค่าประมาณ 8 ของ ATSy่อย #3 คือ S_{30} มีค่าประมาณ 8 และของ ATSy่อย #4 คือ S_{40} มีค่าประมาณ



(ก)



(ข)

รูปที่ 3.8 ตัวอย่างการทำงานของ SM ใน MATS แบบ 4 เส้นทางค้นหา (ก) เส้นทางการค้นหาตาม
 พิกัดของลำดับคำตอบแต่ละ ATSy่อย และ (ข) ลำดับคำตอบของแต่ละ ATSy่อย

8.5 ตามลำดับ พอขึ้นรอบการค้นหาที่ 1 MATS เริ่มต้นการค้นหาที่ P_1 จนได้ S_{11} ซึ่งค่าฟังก์ชันวัตถุประสงค์ของ S_{11} หรือ $f(S_{11}) \approx 4.5$ ถัดไป MATS จะทำการค้นหาคำตอบใน P_2 จนได้ S_{21} ซึ่งมีค่า $f(S_{21}) \approx 4$ ถัดจากนี้จะเป็นการค้นหาที่ P_3 ทำให้ได้ S_{31} ด้วยค่า $f(S_{31}) \approx 7$ และคิวสุดท้ายของรอบการค้นหาที่ 1 เป็นของ P_4 ทำให้ได้ S_{41} ด้วยค่า $f(S_{41}) \approx 7$ ภาพลำดับการสร้างคำตอบเหล่านี้ได้แสดงรายละเอียดไว้แล้วในรูปที่ 3.8 (ข) เมื่อขึ้นรอบการค้นหาที่ 2 MATS ก็ดำเนินการสร้างลำดับคำตอบต่อไปตามปกติจนครบทุกหน่วยค้นหา ช่วงท้ายรอบการค้นหาที่ 2 ซึ่ง MATS ได้กำหนดให้มีการเรียกใช้ DM ครั้งแรก หรือ DM1 ตามขั้นตอนของ DM ที่ได้อธิบายไว้ ได้ผลการเรียงลำดับค่าฟังก์ชันวัตถุประสงค์จากน้อยไปมากของทุกหน่วยค้นหาความสัมพันธ์ที่ (3.2)

$$f(S_{22}) \approx 3 < f(S_{12}) \approx 4 < f(S_{42}) \approx 4.5 < f(S_{32}) \approx 5 \quad (3.2)$$

ถ้าหากว่าพิจารณาคุณภาพของคำตอบตามค่าฟังก์ชันวัตถุประสงค์จะพบว่า คำตอบคุณภาพดีที่สุด คือ 3 ที่ ATS#2 พบ รองลงมาเป็นของ ATS#1 ได้ค่าประมาณ 4 ถัดลงมาเป็นของ ATS#4 ให้ค่าอยู่ที่ 4.5 และที่คุณภาพด้อยที่สุดเป็นของ ATS#3 ให้ค่าประมาณ 5 และตามรูปแบบการลดจำนวนหน่วยค้นหาที่กำหนดไว้คือลดลงทีละครั้งหนึ่งของจำนวนที่มีอยู่ สองหน่วยค้นหาที่ถูกยกเลิกคือหน่วยค้นหาที่ 3 และ 4 สังกะจากรูปที่ 3.8(ข) รอบการค้นหาที่ 2 เป็นรอบการค้นหาสุดท้ายของ P_3 และ P_4 ทำให้ P_3 ได้ลำดับคำตอบคือ $\{S_{30}, S_{31}, S_{32}\}$ และของ P_4 คือ $\{S_{40}, S_{41}, S_{42}\}$ จากนี้ไปจึงเหลือเพียงสองหน่วยค้นหา ได้แก่ ATS#1 และ ATS#2 ที่สลับกันทำงานด้วยตัวประมวลผลเดียว ทำให้เวลาคอยลดลงตามไปด้วยนั่นเอง ขณะนี้จำนวนหน่วยค้นหาที่เหลืออยู่สองเส้นทาง หมายความว่าเราสามารถเรียกใช้ DM ได้อีกหนึ่งครั้งนั่นเองเพื่อลดจำนวนหน่วยค้นหาให้เหลือเพียงหนึ่งเดียว และได้ถูกกำหนดไว้ที่รอบการค้นหาที่ 4 สอดคล้องกับข้อเสนอแนะการตั้งค่า DM ที่ได้กล่าวไว้ในตอนต้นว่า การเรียกใช้ DM สามารถทำได้มากกว่าหนึ่งครั้ง หากจำนวนหน่วยค้นหา ยังคงเหลืออยู่มากกว่าหนึ่งเส้นทาง ดังนั้นเมื่อ MATS ดำเนินการค้นหาคำตอบ S_{14} และ S_{24} แล้วค่าฟังก์ชันวัตถุประสงค์ของทั้งสองคำตอบจึงถูกนำมาเปรียบเทียบกัน นั่นคือ $f(S_{14}) \approx 1.5 < f(S_{24}) \approx 2$ ถ้าพิจารณาตามเกณฑ์ของ DM ที่ตั้งไว้ จึงต้องยกเลิกการค้นหาของ ATS#2 แล้วปล่อยให้ ATS#1 ทำการค้นหาต่อไปจนกระทั่งสิ้นสุดรอบการค้นหาที่ 6 เงื่อนไขยุติการค้นหาถูกพบเพราะได้คำตอบที่ให้ค่าฟังก์ชันวัตถุประสงค์ใกล้เคียงคำตอบวงกว้าง ตามที่ตัวอย่างนี้เลือกใช้พื้นที่ผิวฟังก์ชัน BF ซึ่งทราบแน่ชัดแล้วว่าคำตอบวงกว้างอยู่ที่พิกัด (0,0) ด้วยค่าฟังก์ชันเท่ากับศูนย์ MATS จึงหยุดการค้นหาด้วยเหตุผลดังกล่าว การอธิบายข้างต้นเป็นการอธิบายร่วมกับการยกตัวอย่างเชิงตัวเลขพร้อมภาพประกอบจากพื้นที่ผิวฟังก์ชันจริงที่ใช้ทดสอบ มุ่งเน้นให้ผู้อ่านเข้าใจการทำงานของ DM และเห็นภาพกระบวนการประเมินคุณภาพหน่วยค้นหาด้วยเกณฑ์คุณภาพคำตอบขณะเรียกใช้ DM ให้ชัดเจนนั่นเอง

ถึงตอนนี้เราสามารถประมวลภาพรวมของ MATS ทั้งสามกลไกได้ดังนี้ PM ช่วยกระจายคำตอบเริ่มต้นสำหรับหน่วยค้นหาให้ทั่วถึงทั้งปริภูมิการค้นหา ทั้งนี้เพื่อลดโอกาสเกิดปรากฏการณ์ “ตกสำรวจ” ในบางพื้นที่แล้วจะส่งผลให้ไม่พบคำตอบวงกว้างที่ต้องการหรือต้องใช้เวลาค้นหาที่ยาวนาน เมื่อเทียบเคียงกับกลยุทธ์ของการค้นหาแบบตาบอดดั้งเดิมแล้ว PM จัดอยู่ในกลยุทธ์ความหลากหลายนั่นเอง ส่วน SM ช่วยทำให้ MATS เป็นอัลกอริธึมการค้นหาแบบอนุกรมแต่คิดในเชิงขนาน และสามารถทำงานได้ทั้งบนตัวประมวลผลเดี่ยวและหลายตัวประมวลผล (หากใช้งานกับคอมพิวเตอร์ที่มีหลายตัวประมวลผล เพื่อสมรรถนะสูงสุดควรโปรแกรมแบบขนาน) แม้ว่าจะทำให้เกิดเวลาคอยส่งผลให้เวลาค้นหามากขึ้นบ้าง แต่ก็ยังมีบางกลไกช่วยชดเชยได้ กลไกดังกล่าวคือ DM ที่จะคอยคัดเลือกให้เหลือเพียงหน่วยค้นหาที่จำเป็นในวงรอบการค้นหาของ MATS DM อาจจัดอยู่ในกลยุทธ์ความเข้มข้น โดยยังคำนึงถึงคุณสมบัติความหลากหลายไม่ให้สูญเสียไปมากนัก แต่เมื่อลองเปรียบเทียบกับจินตีสอัลกอริธึม (GAs) แล้ว DM จะมีความคล้ายคลึงกับกระบวนการคัดเลือก (selection) ของ GAs ที่คัดเลือกผู้สืบสายพันธุ์ที่มีความทนทานและความแข็งแรงเท่านั้น เพื่อความอยู่รอดในธรรมชาติได้นั่นเอง หัวข้อต่อไปเป็นการประเมินสมรรถนะของ MATS เทียบกับ ATS แบบเดี่ยว

3.3 การประเมินสมรรถนะของอัลกอริธึม

การวิเคราะห์สมรรถนะในการค้นหาผลเฉลยของการค้นหาแบบตาบอดเชิงปรับตัวหลายวิถี หรือ MATS ดำเนินการทดสอบกับปัญหาการหาค่าเหมาะที่สุดเชิงพื้นผิว (surface optimization) จำนวนสามฟังก์ชัน ได้แก่ ฟังก์ชัน โบฮาเชฟสกี (bohachevsky function หรือ BF) ฟังก์ชันราสตริกิน (rastrigin function หรือ RF) และฟังก์ชันชีเคิลฟอกซ์โฮล (shekel's foxholes function หรือ SF) ซึ่งมีคำตอบวงแคบเฉพาะถิ่นจำนวนมากเป็นอุปสรรคสำหรับการค้นหาคำตอบวงกว้างที่มีเพียงค่าเดียว รายละเอียดคุณสมบัติของทั้งสามฟังก์ชันทดสอบแสดงไว้แล้วในตารางที่ 2.1 ส่วนพารามิเตอร์ของอัลกอริธึมการค้นหาแบบตาบอดเชิงปรับตัวทั้งแบบเส้นทางเดียวและแบบหลายเส้นทางการค้นหา ใช้การตั้งค่าไว้เหมือนกันเพื่อความเป็นธรรมในการเปรียบเทียบสมรรถนะระหว่าง ATS แบบเดี่ยว กับ MATS ซึ่งรายละเอียดการตั้งค่าพารามิเตอร์ต่าง ๆ แสดงไว้ในตารางที่ 2.2

3.3.1 การตั้งค่าพารามิเตอร์ของ MATS

พารามิเตอร์ของ MATS หมายถึง พารามิเตอร์ต่าง ๆ ของกลไกภายในของ MATS ซึ่งก็คือ PM SM และ DM เนื้อหาส่วนนี้ต้องการศึกษาผลของพารามิเตอร์ของ PM และ DM ต่อสมรรถนะการค้นหาของ MATS ส่วน SM คงค่าจำนวนรอบการค้นหาของ ATS ย่อย ไว้ที่หนึ่งรอบตลอดการค้นหาของ MATS

A. PM

การตั้งค่าพารามิเตอร์ของ PM สำหรับปัญหาการหาค่าเหมาะที่สุดเชิงพื้นที่ที่ใช้ทดสอบนี้ อิงความเท่ากันของขนาดพื้นที่เชิงเรขาคณิตเป็นสำคัญ ซึ่งสามารถดำเนินการได้โดยสะดวกเพราะ ปัญหาที่ทดสอบเป็นสองมิติ การตั้งชื่อปริภูมิย่อยได้เรียงลำดับแบบพวนเข็มนาฬิกา ร่วมกับแบบ ขวาไปซ้ายและแบบบนลงล่างประกอบกัน ตัวอย่างการแบ่งปริภูมิการค้นหาพร้อมวิธีการตั้งชื่อ หน่วยค้นหา สำหรับปัญหาในกรอบสองมิติสี่เหลี่ยม กรณี 2 4 8 และ 16 หน่วยค้นหา สามารถดูได้ ในรูปที่ 3.3 ส่วนกรณีของ 32 และ 64 หน่วยค้นหา ดูได้ในรูปที่ 3.9 ปริภูมิการค้นหาย่อยที่แบ่งได้ ด้วยการทำงานของ PM แสดงอยู่ในตารางที่ 3.1 ฟังก์ชัน BF และ ฟังก์ชัน RF มีปริภูมิการค้นหาที่ เหมือนกันจึงได้ผลการแบ่งปริภูมิการค้นหาเหมือนกันด้วย ตัวอย่างผลการแบ่งปริภูมิการค้นหา ของ RF กรณี MATS แบบสองเส้นทางการค้นหา หรือ MATS#2 ด้วยปริภูมิการค้นหา $[2 \ 2; -2 \ -2]$ นั้นได้ ปริภูมิการค้นหาย่อยที่ 1 หรือ #1 คือ $[2 \ 2; 0 \ -2]$ ซึ่งหมายความว่า มีขอบเขตของปริภูมิการ ค้นหาตามแนวแกน x อยู่ในช่วง $[0, 2]$ และมีขอบเขตของปริภูมิการค้นหาตามแนวแกน y อยู่ใน ช่วง $[-2, 2]$ ส่วนของปริภูมิการค้นหาย่อยที่ 2 หรือ #2 คือ $[0 \ 2; -2 \ -2]$ มีความหมายทำนอง เดียวกับปริภูมิการค้นหาย่อยที่ 1 ก็คือขอบเขตของปริภูมิการค้นหาตามแนวแกน x อยู่ในช่วง $[-2, 0]$ และมีขอบเขตของปริภูมิการค้นหาตามแนวแกน y อยู่ในช่วง $[-2, 2]$ เป็นต้น

#8	#7	#6	#5	#4	#3	#2	#1
#9	#10	#11	#12	#13	#14	#15	#16
#24	#23	#22	#21	#20	#19	#18	#17
#25	#26	#27	#28	#29	#30	#31	#32

(ก)

#8	#7	#6	#5	#4	#3	#2	#1
#9	#10	#11	#12	#13	#14	#15	#16
#24	#23	#22	#21	#20	#19	#18	#17
#25	#26	#27	#28	#29	#30	#31	#32
#40	#39	#38	#37	#36	#35	#34	#33
#41	#42	#43	#44	#45	#46	#47	#48
#56	#55	#54	#53	#52	#51	#50	#49
#57	#58	#59	#60	#61	#62	#63	#64

(ข)

รูปที่ 3.9 ตัวอย่าง PM พร้อมการตั้งชื่อหน่วยค้นหา (ก) 32 เส้นทาง และ (ข) 64 เส้นทาง

ตารางที่ 3.1 พารามิเตอร์ของ MATS ภาค PM ที่ใช้ทดสอบ

Test functions	Sub-search-spaces from PM					
	MATS#2	MATS#4	MATS#8	MATS#16	MATS#32	MATS#64
BF and RF [0 2; -2 -2]	#1[2 2;0 -2] #2[0 2;-2 -2]	#1[2 2;0 0] #2[0 2;-2 0] #3[0 0;-2 -2] #4[2 0;0 -2]	#1[2 2;1 0] #2[1 2;0 0] #3[0 2;-1 0] #4[-1 2;-2 0] #5[-1 0;-2 -2] #6[0 0;-1 -2] #7[1 0;0 -2] #8[2 0;1 -2]	#1[2 2;1 1] #2[1 2;0 1] #3[0 2;-1 1] #4[-1 2;-2 1] #5[-1 1;-2 0] #6[0 1;-1 0] #7[1 1;0 0] #8[2 1;1 0]	#1[2 2;1.5 1] #2[1.5 2;1 1] #3[1 2;0.5 1] #4[0.5 2;0 1] #5[0 2;-0.5 1] #6[-0.5 2;-1 1] #7[-1 2;-1.5 1] #8[-1.5 2;-2 1]	#1[2 2;1.5 1.5] #2[1.5 2;1 1.5] #3[1 2;0.5 1.5] #4[0.5 2;0 1.5] #5[0 2;-0.5 1.5] #6[-0.5 2;-1 1.5] #7[-1 2;-1.5 1.5] #8[-1.5 2;-2 1.5]
SF [40 40; -40 -40]	#1[40 40;0 -40] #2[0 40;-40 -40]	#1[40 40;0 0] #2[0 40;-40 0] #3[40 0;0 -40] #4[0 0;-40 -40]	#1[40 40;20 0] #2[20 40;0 0] #3[0 40;-20 0] #4[-20 40;-40 0] #5[40 0;20 -40] #6[20 0;0 -40] #7[0 0;-20 -40] #8[-20 0;-40 -40]	#1[40 40;20 20] #2[20 40;0 20] #3[0 40;-20 20] #4[-20 40;-40 20] #5[20 20;-40 0] #6[0 20;-20 0] #7[20 20;0 0] #8[40 20;20 0]	#1[40 40;30 20] #2[30 40;20 20] #3[20 40;10 20] #4[10 40;0 20] #5[0 40;-10 20] #6[-10 40;-20 20] #7[-20 40;-30 20] #8[-30 40;-40 20]	#1[40 40;30 30] #2[30 40;20 30] #3[20 40;10 30] #4[10 40;0 30] #5[0 40;-10 30] #6[-10 40;-20 30] #7[-20 40;-30 30] #8[-30 40;-40 30]
				the rest are omitted.	the rest are omitted.	the rest are omitted.

B. SM

พารามิเตอร์ภายใน SM ของ MATS คือ จำนวนรอบที่ ATSy่อย แต่ละหน่วยค้นหาจะดำเนินการได้ในแต่ละรอบการค้นหาของ MATS ในงานวิจัยนี้จำนวนรอบของ ATSy่อย กำหนดให้คงที่ตลอดการค้นหาของ MATS และให้จำนวนรอบการค้นหาของ ATSy่อย แต่ละหน่วยค้นหาทำงานครบละหนึ่งรอบ ในแต่ละรอบการค้นหาของ MATS จนกระทั่ง MATS ยุติการค้นหา

C. DM

พารามิเตอร์ DM ของ MATS มีหน้าที่ลดจำนวนหน่วยค้นหา ATSy่อย ในกำกับของ MATS ให้เหลือน้อยที่สุด แต่ยังคงพยายามรักษาโอกาสการค้นหาคำตอบวงกว้างให้พบได้ในเวลาที่รวดเร็ว ดำเนินการ โดยการยกเลิกการทำงานของ ATSy่อย บางเส้นทางที่ถูกประเมินแล้วว่าไม่มีประสิทธิภาพการค้นหาคำตอบวงกว้างต่ำกว่าออกจากวงรอบการค้นหาของ MATS สำหรับเกณฑ์ที่ใช้ในการประเมินประสิทธิภาพการค้นหาของหน่วยค้นหา ใช้การพิจารณาค่าฟังก์ชันวัตถุประสงค์ที่ดีที่สุดของแต่ละหน่วยค้นหา ATSy่อย ค้นพบ ณ รอบการค้นหาที่มีการเรียกใช้ DM จึงเป็นวิธีการประเมินคุณภาพล่าสุด (recency) จำนวนครั้งการเรียกใช้ DM ขึ้นกับจำนวนหน่วยค้นหา ATSy่อย ในกำกับของ MATS และรูปแบบการลดจำนวนหน่วยค้นหา สามารถแบ่งได้เป็นสองลักษณะใหญ่ ๆ ตามอัตราการลดลง ได้แก่ รูปแบบการลดจำนวนหน่วยค้นหาอย่างค่อยเป็นค่อยไป และรูปแบบการลดจำนวนหน่วยค้นหาอย่างทันทีทันใด

รูปแบบหนึ่งของการลดจำนวนอย่างค่อยเป็นค่อยไป คือ การลดจำนวนลงคราวละกึ่งหนึ่ง จนกระทั่งเหลือเพียงหน่วยค้นหาเดียว ดังแสดงในตารางที่ 3.2 ตัวอย่างเช่น MATS#4 ที่ทำการค้นหามนฟังก์ชัน BF มีการเรียกใช้ DM จำนวน 2 ครั้ง โดยครั้งที่ 1 ลดจำนวนหน่วยค้นหาจาก 4 เหลือ 2 และ ครั้งที่ 2 ครั้งสุดท้าย ลดจำนวนหน่วยค้นหาจาก 2 เหลือ 1 หน่วยค้นหา อีกตัวอย่างหนึ่ง

สำหรับรูปแบบการลดจำนวนหน่วยค้นหาอย่างค่อยเป็นค่อยไปได้แก่ MATS#8 ที่ทำการค้นหาบนฟังก์ชัน SF มีการเรียกใช้ DM จำนวน 3 ครั้ง โดยครั้งที่ 1 ลดจำนวนหน่วยค้นหาจาก 8 เหลือ 4 ส่วนการเรียกใช้ DM ครั้งที่ 2 ลดจาก 4 เหลือ 2 และ ครั้งที่ 3 ครั้งสุดท้าย ทำการลดจำนวนหน่วยค้นหาจาก 2 เหลือ 1 หน่วยค้นหาในที่สุด

ส่วนรูปแบบการลดจำนวนหน่วยค้นหาอย่างทันทีทันใดเป็นการเรียกใช้ DM หนึ่งครั้งแล้วทำการลดจำนวนหน่วยค้นหาเหลือเพียงหน่วยค้นหาเดียว รูปแบบนี้มักใช้งานได้ดีกับ MATS ที่มีจำนวนหน่วยค้นหาค่อนข้างมาก ดังแสดงในตารางที่ 3.2 ตัวอย่างเช่น MATS#32 ที่ทำการค้นหาบนฟังก์ชัน SF มีการเรียกใช้ DM เพียง 1 ครั้ง แล้วลดจำนวนหน่วยค้นหาจาก 32 เหลือ 1 หน่วยค้นหา ส่วนอีกตัวอย่างหนึ่งได้แก่ MATS#64 ที่ทำการค้นหาบนฟังก์ชัน RF มีการเรียกใช้ DM เพียง 1 ครั้ง แล้วลดจำนวนหน่วยค้นหาจาก 64 เหลือ 1 หน่วยค้นหา

การเลือกใช้รูปแบบของการลดจำนวนหน่วยค้นหาขึ้นอยู่กับลักษณะและธรรมชาติของปัญหาแต่ละชนิด ผู้วิจัยแนะนำให้เริ่มต้นด้วยรูปแบบการลดจำนวนอย่างค่อยเป็นไปก่อน แต่หากไม่สามารถปรับปรุงสมรรถนะการค้นหาได้จึงค่อยเปลี่ยนการตั้งค่า DM เป็นรูปแบบการลดจำนวนหน่วยค้นหาอย่างทันทีทันใดจะเหมาะสมกว่า

ตารางที่ 3.2 พารามิเตอร์ของ DM ใน MATS ที่ใช้ทดสอบ

Test functions	q th Iteration at which the DM is invoked and the number of discarded path(s)																		
	MATS																		
	#2		#4		#8			#16				#32					#64		
	1 st	1 st	2 nd	1 st	2 nd	3 rd	1 st	2 nd	3 rd	4 th	1 st	2 nd	3 rd	4 th	5 th	1 st	2 nd	3 rd	
BF	q th itr.	5	5	10	5	10	15	5	10	15	20	5	10	15	20	25	1	3	5
	no. dis. pth.	1	2	1	4	2	1	8	4	2	1	16	8	4	2	1	32	16	15
RF	q th itr.	5	50	-	2	4	6	2	4	6	8	2	4	6	8	10	1	-	-
	no. dis. pth.	1	3	-	4	2	1	8	4	2	1	16	8	4	2	1	63	-	-
SF	q th itr.	5	2	4	2	4	6	2	4	6	8	1	-	-	-	-	1	-	-
	no. dis. pth.	1	2	1	4	2	1	8	4	2	1	31	-	-	-	-	63	-	-

รูปแบบการลดจำนวนหน่วยค้นหาแบบค่อยเป็นค่อยไปตามตารางที่ 3.2 สามารถอธิบายแนวทางการตั้งค่ารอบการเรียกใช้ DM ได้จากตัวอย่างดังต่อไปนี้ กรณีของ MATS#2 ที่ทำการค้นหาคำตอบวงกว้างของฟังก์ชัน BF เป็นแบบสองหน่วยค้นหา และต้องมีการเรียกใช้ DM จำนวนหนึ่งครั้ง เพื่อลดจำนวนหน่วยค้นหาจากสองหน่วยค้นหาให้เหลือหนึ่งหน่วยค้นหา เนื่องจากผลการทดสอบด้วย ATS แบบเดี่ยว ตามตารางที่ 2.3 แถวที่ 2 ให้ค่าเฉลี่ยประมาณ 556 รอบการค้นหา จึงจะพบคำตอบวงกว้าง ประกอบกับ MATS มีสองหน่วยค้นหาสลับกันทำงาน จึงควรตั้งค่ารอบการเรียกใช้ DM ให้ต่ำกว่าครึ่งหนึ่งของค่าเฉลี่ยรอบการค้นหาที่ได้จาก ATS แบบเดี่ยว นั่นคือควรตั้งรอบการเรียกใช้ DM ให้ก่อนรอบการค้นหาที่ 278 เพื่อจะได้มีโอกาสทำความเร็วการค้นหาคำตอบ

วงกว้างได้ดีกว่า ATSแบบเดี่ยว นั่นเอง ในที่นี้จึงกำหนดให้มีการเรียกใช้ DM ในรอบการค้นหาที่ 5 อย่างไรก็ตามการตั้งค่ารอบการเรียกใช้ DM ยิ่งต่ำกว่าค่าเฉลี่ยรอบการค้นหาของ ATSแบบเดี่ยวยิ่งมีโอกาสใช้เวลาการค้นหาบ่อยก็จริงตามความรู้สึกทั่วไป ซึ่งความเป็นจริงแล้วจะขึ้นกับลักษณะเฉพาะของแต่ละปัญหาแต่ตัวอย่างอีกทีหนึ่ง

ตัวอย่างถัดไป เป็นการตั้งค่าการเรียกใช้ MATS#4 ซึ่งมีสี่หน่วยค้นหาหรือสี่เส้นทางการค้นหา นั่นเอง ตามตารางที่ 3.2 จะมีการเรียกใช้ DM จำนวนสองครั้ง เพื่อลดจำนวนหน่วยค้นหาจากสี่ให้เหลือสองในการเรียกใช้ DM ครั้งที่หนึ่ง และลดจำนวนหน่วยค้นหาที่เหลือจากสองหน่วยค้นหาเป็นหนึ่งหน่วยค้นหาในการเรียกใช้ DM ครั้งที่สองซึ่งเป็นครั้งสุดท้าย อาศัยข้อมูลจำนวนรอบการค้นหาเฉลี่ยของ ATSแบบเดี่ยว เช่นเดียวกับการตั้งค่าของ MATS#2 ที่ได้นำเสนอไปแล้วข้างต้น กรณีของ MATS#4 มีสี่เส้นทางจึงควรตั้งรอบการเรียกใช้ DM ทั้งสองครั้งที่ทำให้ผลรวมรอบการค้นหาทั้งหมดน้อยกว่าค่าเฉลี่ยรอบการค้นหาของ ATSแบบเดี่ยว นั่นคือประมาณ 556 รอบการค้นหา แนวทางการตั้งค่ารอบการเรียกใช้ DM สามารถสรุปได้ตามสมการที่ (3.3) ข้างล่างนี้

$$\sum_{i=1}^n dm_i \times n_i < \overline{SR}_{ATS} \quad (3.3)$$

โดยที่ dm_i คือ รอบการค้นหาที่มีการเรียกใช้ DM ครั้งที่ i n_i คือ จำนวนหน่วยค้นหาก่อนการเรียกใช้ DM ครั้งที่ i และ \overline{SR}_{ATS} คือ ค่าเฉลี่ยรอบการค้นหาที่พบคำตอบวงกว้างของ ATSแบบเดี่ยว กรณีของ MATS#2 ที่ค้นหาบนฟังก์ชัน BF มีการเรียกใช้ DM หนึ่งครั้ง ณ รอบการค้นหาที่ 5 จึงได้ค่า $dm_1 = 5$, $n_1 = 2$ และ $\overline{SR}_{ATS} = 556$ และเมื่อแทนค่าเหล่านี้ลงในสมการที่ (3.3) ยังคงทำให้สมการเป็นจริง ดังนี้

$$5 \times 2 < 556 \quad (3.4)$$

กรณีของ MATS#4 ที่ค้นหาบนฟังก์ชัน BF มีการเรียกใช้ DM สองครั้ง ณ รอบการค้นหาที่ 5 และ รอบการค้นหาที่ 10 ตามลำดับ จึงได้ค่า $dm_1 = 5$, $dm_2 = 10$, $\overline{SR}_{ATS} = 556$, $n_1 = 4$, และ $n_2 = 2$ เมื่อแทนค่าเหล่านี้ลงในสมการที่ (3.3) จะได้ว่า

$$5 \times 4 + 10 \times 2 < 556 \quad (3.5)$$

3.3.2 ผลและอภิปรายผล

โปรแกรมการค้นหาแบบตุงเชิงปรับตัวหลายวิธีได้รับการพัฒนาขึ้นด้วย MATLAB™ ดำเนินการทดสอบกับปัญหาการหาค่าเหมาะที่สุดเชิงพื้นผิวทั้งสามฟังก์ชัน บนเครื่องคอมพิวเตอร์ Intel Celeron(R) ความเร็ว 2.6 จิกกะเฮิรตซ์ ซึ่งมีหน่วยความจำ RAM ขนาด 256 เมกกะไบต์ หน่วยความจำ HDD ขนาด 30 จิกกะไบต์ จำนวน 50 ครั้ง แล้วหาค่าเฉลี่ยได้ผลการทดสอบเป็นเวลา การค้นหาเฉลี่ย และจำนวนรอบการค้นหาเฉลี่ย แสดงอยู่ในตารางที่ 3.3 และ 3.4 ตามลำดับ ส่วน ตารางที่ 3.5 เป็นการนำเสนอข้อมูลจากตารางที่ 3.3 ในรูปแบบเปอร์เซ็นต์การประหยัดเวลาค้นหา คำนวณตามความสัมพันธ์ที่ (3.6) และตารางที่ 3.6 เป็นการนำเสนอข้อมูลจากตารางที่ 3.4 ใน รูปแบบจำนวนรอบการค้นหาเฉลี่ยของ MATS สมมูลกับ ATSแบบเดี่ยว เพื่อยืนยันความถูกต้อง ของผลเวลาการค้นหาเฉลี่ยที่ได้

$$\text{Percent savings in search time} = \frac{(\text{average search time of ATS} - \text{average search time of MATS})}{\text{average search time of ATS}} \times 100 \quad (3.6)$$

จำนวนรอบการค้นหาเฉลี่ยของ MATS สมมูลกับ ATSแบบเดี่ยว หรือ $Count_{MATSeq}$ แสดง จำนวนรอบการค้นหาของ MATS ซึ่งช่อนจำนวนรอบการค้นหาของ ATS ย่อย หลายหน่วยค้นหา ไว้ภายในหากนำจำนวนรอบการค้นหาของ ATS ย่อย ต่าง ๆ มาเรียงต่อกันเราจะได้ $Count_{MATSeq}$ ที่เทียบเท่าจำนวนรอบการค้นหาของ ATS ย่อย ทั้งหมดของ MATS ซึ่งอยู่ในฐานเวลาเดียวกัน กับ $Count_{ATS}$ จึงสามารถเปรียบเทียบกันได้อย่างสมเหตุสมผล ค่า $Count_{MATSeq}$ แสดงดังตารางที่ 3.5 คำนวณได้ตามความสัมพันธ์ (3.7) โดยอาศัยข้อมูลการเรียกใช้ DM จากตารางที่ 3.2 ประกอบด้วย no. dis. pth. (number of discarded search paths) ซึ่งหมายถึงจำนวนหน่วยค้นหาที่ถูก ยกเลิกเมื่อมีการเรียกใช้ DM แต่ละครั้ง แสดงด้วยค่า n_i และค่า k พิจารณาจากจำนวนการ เรียกใช้ DM นั้นเอง

$$Count_{MATSeq} = n_0 \cdot Count_{MATS} + \sum_{i=1}^k n_i \cdot d_i, \quad 1 < k \leq N \quad \text{และ} \quad N = \sum_{i=0}^k n_i \quad (3.7)$$

ตารางที่ 3.3 เวลาการค้นหาเฉลี่ยของ ATS และ MATS

Functions	Average search time in seconds from 50 trials						
	ATS	MATS					
	1-path	# 2	# 4	# 8	# 16	# 32	# 64
BF	5.3940	5.0516	4.8901	2.3435	2.3309	4.1309	3.3029
RF	8.2067	6.5863	6.9925	7.2175	4.5316	5.0078	7.3221
SF	2.2837	1.8818	1.7697	1.8134	1.3918	1.7944	1.8431

ตารางที่ 3.4 จำนวนรอบการค้นหาเฉลี่ยของ ATSแบบเดี่ยว และ MATS

Functions	Average search rounds from 50 trials						
	ATS	MATS					
	1-path	# 2	# 4	# 8	# 16	# 32	# 64
BF	556.06	530.20	507.94	186.82	142.70	202.34	173.76
RF	759.80	685.40	603.78	713.26	327.32	313.48	200.32
SF	108.22	79.84	72.96	64.62	30.62	54.04	36.16

ตารางที่ 3.5 เปอร์เซนต์การประหยัดเวลาค้นหา

Functions	Percent savings in search time						
	ATS	MATS					
	1-path	# 2	# 4	# 8	# 16	# 32	# 64
BF	0.0000	6.35	9.34	56.55	56.79	23.42	38.77
RF	0.0000	19.74	14.80	12.05	44.78	38.98	10.78
SF	0.0000	17.60	22.51	20.59	39.06	21.43	19.29

ตารางที่ 3.6 จำนวนรอบการค้นหาเฉลี่ยของ MATS สมมูลกับ ATSแบบเดี่ยว

Functions	Average search rounds from 50 trials equivalent to a single ATS						
	ATS	MATS					
	1-path	# 2	# 4	# 8	# 16	# 32	# 64
BF	556.06	535.20	527.94	241.82	272.70	487.34	328.76
RF	759.80	690.40	753.78	735.26	379.32	427.48	263.32
SF	108.22	84.84	80.96	86.62	82.62	85.04	99.16

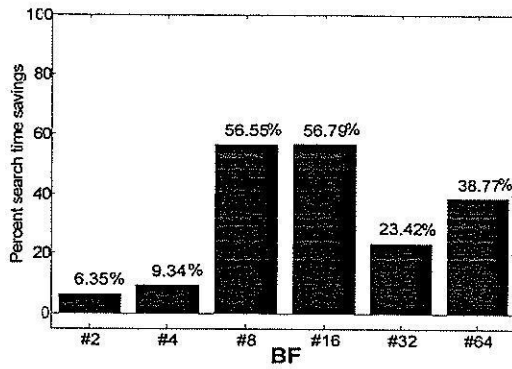
ค่าเวลาการค้นหาเฉลี่ยที่ของ ATSแบบเดี่ยว และ MATS แต่ละค่าที่แสดงในตารางที่ 3.3 ได้จากการทดสอบจำนวน 50 ครั้ง แล้วหาค่าเฉลี่ย ค่าของ ATSแบบเดี่ยวในคอลัมน์ที่สองจะทำหน้าที่เป็นค่าอ้างอิงสำหรับการปรับตั้งค่าพารามิเตอร์ของ MATS จนกว่าจะได้ค่าที่พอใจ นั่นคือ ค่าเวลาการค้นหาเฉลี่ยของ MATS ควรต้องต่ำกว่าของ ATSแบบเดี่ยว นั่นเอง สมาชิกของ MATS ที่สร้างขึ้นสำหรับงานวิจัยนี้มีอยู่ด้วย 6 ชนิด ได้แก่ MATS#2 MATS#4 MATS#8 MATS#16 MATS#32 และ MATS#64 ซึ่งตัวเลขหลังสัญลักษณ์ # หมายถึง จำนวนหน่วยค้นหาหรือจำนวนเส้นทางการค้นหาของ MATS ตัวอย่าง เช่น MATS#2 ก็คือ MATS แบบสองหน่วยค้นหาหรือแบบสองเส้นทางการค้นหานั่นเอง ค่าเวลาการค้นหาเฉลี่ยของ MATS ที่ได้เกือบทุกชนิดในทุกฟังก์ชันตามตารางที่ 3.3 พบว่าให้ค่าเวลาการค้นหาเฉลี่ยต่ำกว่าของ ATSแบบเดี่ยว ทั้งสิ้น ตัวอย่างการอ่านค่าเวลาการค้นหาเฉลี่ยจากตารางที่ 3.3 สำหรับกรณีของฟังก์ชัน RF เวลาการค้นหาค่าตอบวงกว้างเฉลี่ยของ ATSแบบเดี่ยว อยู่ที่ 8.2067 วินาที ส่วนของ MATS#2 #4 #8 #16 #32 และ #64 ใช้เวลาการค้นหาค่าตอบวงกว้างเฉลี่ยอยู่ที่ 6.5863 6.9925 7.2175 4.5316 5.0078 และ 7.3221 วินาที ตามลำดับ ซึ่งทุกตัวล้วนน้อยกว่าของ ATSแบบเดี่ยว โดย MATS#16 ใช้เวลาการค้นหาน้อย

ที่สุด จึงถือว่าหาค่าตอบวงกว้างของ RF ได้รวดเร็วที่สุด หากมีการคัดเลือกตัวแทน MATS เพื่อทำการค้นหาค่าตอบวงกว้างสำหรับฟังก์ชัน RF หรือ ฟังก์ชันที่มีลักษณะพื้นผิวคล้ายคลึงกับฟังก์ชัน RF จึงแนะนำให้เลือก MATS#16 เพื่อทำการค้นหาจะเหมาะสมที่สุด

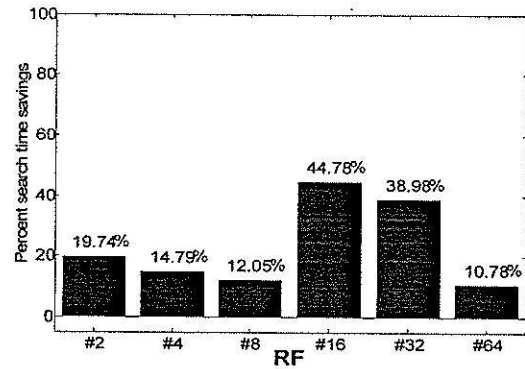
ตารางที่ 3.4 แสดงจำนวนรอบการค้นหาเฉลี่ยของ MATS จากการทดสอบ เป็นค่าที่ใช้ตรวจสอบความถูกต้องของเวลาการค้นหาที่ MATS ใช้ค่าจำนวนรอบการค้นหาเฉลี่ยของ ATS แบบเดี่ยว ในคอลัมน์ที่ 1 ถูกใช้เป็นแนวทางการปรับตั้งค่าพารามิเตอร์การเรียกใช้ DM ของ MATS ด้วยเป้าหมายที่จะลดจำนวนรอบการค้นหาของ MATS เพื่อลดเวลาการค้นหาของ MATS ให้น้อยกว่าของ ATS แบบเดี่ยว และพบว่า MATS ทุกชนิดและทุกฟังก์ชันใช้จำนวนรอบการค้นหาน้อยกว่าของ ATS แบบเดี่ยว

ตารางที่ 3.5 แสดงเปอร์เซ็นต์การประหยัดเวลาค้นหาจากการใช้ MATS เมื่อเปรียบเทียบกับ ATS โดยการแปลงข้อมูลเวลาค้นหาเฉลี่ยจากตารางที่ 3.4 ผ่านการคำนวณตามสมการที่ (3.6) สามารถสังเกตเห็นจากตารางว่า ข้อมูลทุกค่าเป็นบวก หมายความว่าเวลาค้นหาค่าตอบวงกว้างของ MATS น้อยกว่า ของ ATS แบบเดี่ยว จากตารางที่ 3.6 คอลัมน์ที่ 2 จากซ้ายซึ่งเป็นผลของ ATS แบบเดี่ยว จึงแสดงผลด้วยค่าศูนย์ ข้อมูลทั้งหมดยืนยันว่า MATS มีสมรรถนะการค้นหาที่ดีกว่า ATS แบบเดี่ยว ตัวอย่าง เช่น กรณีของฟังก์ชัน BF MATS#16 สามารถลดทอนเวลาค้นหาลงได้ 56.79% หรือกล่าวอีกนัยหนึ่งได้ว่า MATS มีความเร็วเป็นสองเท่าของ ATS แบบเดี่ยว ในการค้นหาค่าตอบของปัญหา BF ส่วน MATS#2 สามารถประหยัดเวลาค้นหาได้ไม่มากนักเพียง 6.35% เท่านั้น ข้อมูลในตารางที่ 3.5 ได้รับการนำเสนอในรูปกราฟแท่งดังรูปที่ 3.10

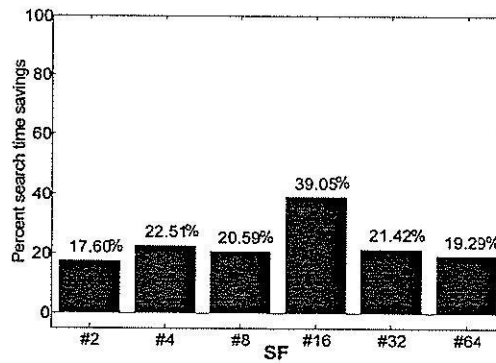
ตารางที่ 3.6 แสดงจำนวนรอบการค้นหาเฉลี่ยของ MATS ที่สมมูลกับ ATS แบบเดี่ยว เป็นค่าที่ได้จากการแปลงจำนวนรอบการค้นหาเฉลี่ยของ MATS จากการทดสอบมาอยู่ในฐานอ้างอิงเดียวกันกับจำนวนรอบของ ATS แบบเดี่ยว เพื่อการเปรียบเทียบสมรรถนะ พิจารณา MATS#4 ปัญหา BF เป็นตัวอย่างหนึ่ง ตารางที่ 3.6 แสดงจำนวนรอบการค้นหา 527.94 รอบ เทียบเท่า ATS ซึ่งมีที่มาจากจำนวนรอบการค้นหา 507.94 รอบที่แสดงในตารางที่ 3.4 บวกด้วยจำนวนรอบการค้นหาที่ ATS ย่อย ใช้ไปก่อนที่ DM จะสั่งให้ยุติ จากข้อมูลการตั้งค่าพารามิเตอร์ของ DM ที่กล่าวผ่านมาแล้ว DM ถูกเรียกใช้ในรอบการค้นหาที่ 5 และ 10 ซึ่งสั่งยุติการค้นหา 2 และ 1 หน่วยการค้นหาย่อยตามลำดับ ดังนั้นจึงได้ข้อมูลจากผลการคำนวณว่า $(5 \times 2) + (10 \times 1) + 507.94 = 527.94$ ดังนั้นเป็นต้น จำนวนรอบการค้นหาสมมูล 527.94 รอบนี้ยังคงน้อยกว่า 556.06 รอบในการทำงานของ ATS แบบเดี่ยว (ดูตารางที่ 3.4) ดังนั้นจึงช่วยยืนยันได้ดีว่าเวลาค้นหาเฉลี่ยโดย MATS (4.8091 วินาที-ตารางที่ 3.3) น้อยกว่าของกรณี ATS (5.3940 วินาที-ตารางที่ 3.3) มีเหตุผลถูกต้องตามความเป็นจริง ผู้อ่านสามารถใช้หลักการนี้วิเคราะห์ข้อมูลในกรณีอื่น ๆ ได้



(ก)



(ข)



(ค)

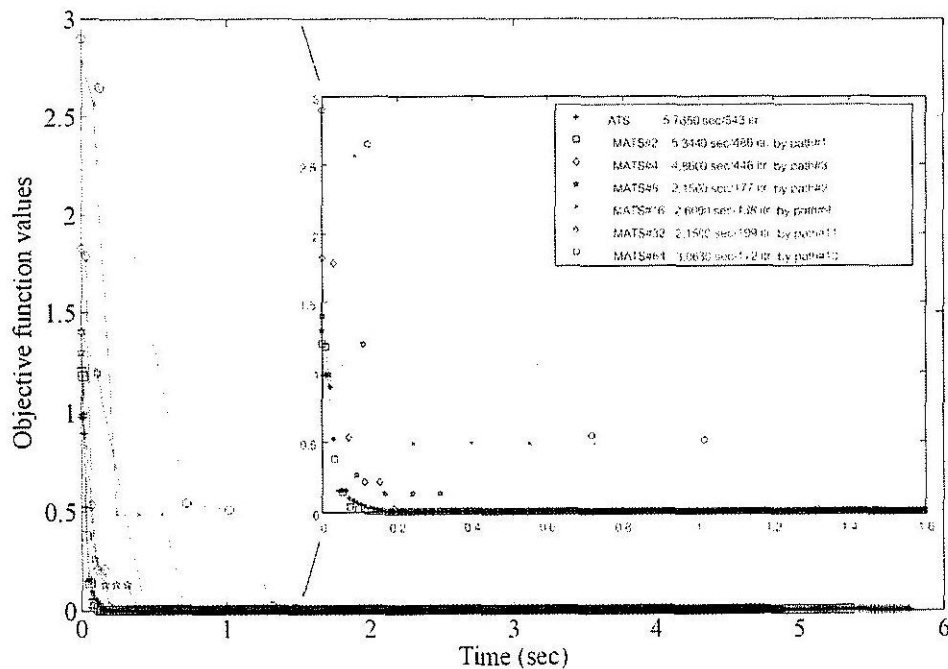
รูปที่ 3.10 กราฟแท่งแสดงการประหยัดเวลาค้นหาเป็น % เมื่อใช้ MATS กับปัญหา

(ก) BF (ข) RF และ (ค) SF

ผลการทดสอบสมรรถนะของ MATS เทียบกับของ ATS แบบเดี่ยว เมื่อพิจารณาการประหยัดเวลาการค้นหาดังที่แสดงในภาพรวมไว้ด้วยกราฟแท่งในรูปที่ 3.10 พบว่าปัญหา BF RF และ SF ทั้งสามชนิด ตั้งแต่แบบสองหน่วยค้นหาจนถึงหกสิบสี่หน่วยค้นหา MATS สามารถทำความเร็วการค้นหาคำตอบวงกว้างได้ดีกว่า ATS แบบเดี่ยว กรณีของ BF MATS#16 สามารถค้นหาได้เร็วที่สุดด้วยเวลาการค้นหาเฉลี่ยที่สั้นกว่าของ ATS แบบเดี่ยวถึง 56.79% และที่น้อยที่สุดเป็นของ MATS#2 ซึ่งประหยัดเวลาลงได้ 6.35% เท่านั้น กรณีของ RF MATS#16 ทำเวลาได้เร็วที่สุดในการค้นหาคำตอบ นั่นคือใช้เวลาสั้นกว่า ATS ถึง 44.78% ขณะที่ MATS#64 ลดเวลาการค้นหาลงได้เพียง 10.78% และกรณีของ SF MATS#16 อีกเช่นกันที่ทำงานด้วยความรวดเร็วที่สุดนั่นคือ ลดเวลาการค้นหาลงได้ 39.05% และที่น้อยที่สุดยังคงเป็นของ MATS#2 ซึ่งทำได้เพียง 17.60% ในการประหยัดเวลาการค้นหา

3.3.3 กราฟการเข้าสู่หาคำตอบวงกว้าง

คุณสมบัติการเข้าสู่หาคำตอบวงกว้างของ ATS ได้รับการพิสูจน์เป็นมาก่อนแล้ว (Puangdownreong et al, 2004) ดังนั้น MATS ที่ใช้ ATS เป็นหน่วยค้นหาหลักเพียงอย่างเดียว จึงยังคงคุณสมบัติการเข้าสู่หาคำตอบวงกว้างอยู่อย่างครบถ้วน เพื่อเป็นการยืนยันผล หัวข้อนี้จึงแสดงกราฟค่าฟังก์ชันวัตถุประสงค์เชิงเลขของปัญหา BF เป็นตัวอย่างให้เห็นจริงว่า ค่าฟังก์ชันวัตถุประสงค์เข้าสู่ค่าต่ำสุดจริง ภายในเวลาจำกัด สำหรับของ MATS ได้รับการทดสอบกับปัญหา BF แสดงดังรูปที่ 3.11 ภาพขยายตรงกลางภาพใหญ่ยังต้องการอธิบายสื่อถึงผู้อ่านให้เข้าใจผลของ SM ที่ทำให้เกิดเวลาคอย ซึ่งทำให้ระยะห่างของคำตอบในช่วงแรกยาวนานกว่าช่วงท้าย ๆ เมื่อมีการเรียกใช้ DM เป็นระยะ ๆ ช่วงห่างของคำตอบในลำดับคำตอบของ MATS ก็ค่อย ๆ ลดลง จนเหลือเวลาเฉลี่ยที่ ATSy่อย หรือ ATSแบบเดี่ยว ใช้ในการค้นหาคำตอบแต่ละรอบ



รูปที่ 3.11 กราฟการเข้าสู่หาคำตอบวงกว้างของ ATS และ MATS ทั้ง 6 ชนิด กับปัญหา BF

ATS เริ่มต้นการค้นหาด้วยค่าฟังก์ชันวัตถุประสงค์ 0.9740 เมื่อดำเนินการค้นหาต่อไปจนถึงรอบการค้นหาที่ 543 จึงพบกับคำตอบด้วยค่าฟังก์ชันวัตถุประสงค์ 3.3177×10^{-10} ซึ่งสอดคล้องกับเงื่อนไขยุติการค้นหา โดยใช้เวลาค้นหาไปทั้งสิ้น 5.7650 วินาที แต่ละคำตอบห่างกันประมาณ $t = 5.7650$ วินาที / 543 รอบการค้นหา ≈ 0.0106 วินาที

MATS แบบ 2 เส้นทางค้นหา มีหน่วยค้นหาที่ 1 เป็นผู้พบคำตอบวงกว้าง โดยเริ่มต้นการค้นหาด้วยค่าฟังก์ชันวัตถุประสงค์ 1.2018 จนกระทั่งรอบการค้นหาที่ 486 จึงพบกับคำตอบวงกว้างด้วยค่าฟังก์ชันวัตถุประสงค์ 6.3224×10^{-10} โดยใช้เวลาค้นหาไปทั้งสิ้น 5.3440 วินาที

MATS แบบ 4 เส้นทางค้นหา มีหน่วยค้นหาที่ 3 เป็นผู้พบคำตอบวงกว้าง โดยเริ่มค้นหาด้วยค่าฟังก์ชันวัตถุประสงค์ 1.8234 จนกระทั่งรอบการค้นหาที่ 446 จึงพบกับคำตอบวงกว้างด้วยค่าฟังก์ชันวัตถุประสงค์ 6.6243×10^{-10} โดยใช้เวลาค้นหาไปทั้งสิ้น 4.8600 วินาที

MATS แบบ 8 เส้นทางค้นหา มีหน่วยค้นหาที่ 2 เป็นผู้พบคำตอบวงกว้าง โดยเริ่มค้นหาด้วยค่าฟังก์ชันวัตถุประสงค์ 1.2972 จนกระทั่งรอบการค้นหาที่ 177 จึงพบกับคำตอบวงกว้างด้วยค่าฟังก์ชันวัตถุประสงค์ 1.8157×10^{-10} โดยใช้เวลาค้นหาไปทั้งสิ้น 2.1560 วินาที

MATS แบบ 16 เส้นทางค้นหา มีหน่วยค้นหาที่ 9 เป็นผู้พบคำตอบวงกว้าง โดยเริ่มค้นหาด้วยค่าฟังก์ชันวัตถุประสงค์ 2.7822 จนกระทั่งรอบการค้นหาที่ 138 จึงพบกับคำตอบวงกว้างด้วยค่าฟังก์ชันวัตถุประสงค์ 1.7153×10^{-10} โดยใช้เวลาค้นหาไปทั้งสิ้น 2.6090 วินาที

MATS แบบ 32 เส้นทางค้นหา มีหน่วยค้นหาที่ 11 เป็นผู้พบคำตอบวงกว้าง โดยเริ่มค้นหาด้วยค่าฟังก์ชันวัตถุประสงค์ 1.3982 จนกระทั่งรอบการค้นหาที่ 199 จึงพบกับคำตอบวงกว้างด้วยค่าฟังก์ชันวัตถุประสงค์ 6.6220×10^{-10} โดยใช้เวลาค้นหาไปทั้งสิ้น 2.1500 วินาที

MATS แบบ 64 เส้นทางค้นหา มีหน่วยค้นหาที่ 13 เป็นผู้พบคำตอบวงกว้าง โดยเริ่มค้นหาด้วยค่าฟังก์ชันวัตถุประสงค์ 2.8961 จนกระทั่งรอบการค้นหาที่ 172 จึงพบกับคำตอบวงกว้างด้วยค่าฟังก์ชันวัตถุประสงค์ 2.9307×10^{-10} โดยใช้เวลาค้นหาไปทั้งสิ้น 3.0630 วินาที ผลของ DM ต่อเวลาการค้นหาสามารถสังเกตได้จากระยะห่างของคำตอบที่อยู่ติดกันบนเส้นกราฟการลู่เข้าหาคำตอบวงกว้าง ดังแสดงในรูปที่ 3.11 เส้นบนสุดเป็นของ MATS แบบ 64 เส้นทางค้นหา เป็นการยืนยันว่า MATS#64 ใช้เวลานานในการคอยการจัดลำดับการค้นหา จนทำให้ประสิทธิภาพในการค้นหาคำตอบไม่ดีเท่าที่ควร

3.4 สรุป

การพัฒนาต่อฮอลอกริซึม ATS เกิดขึ้นเป็น ATS แบบหลายวิถี หรือ MATS ที่อาศัยการจัดการที่ดีต่อการทำงานของ ATS โดยยึดหลักกลยุทธ์ความหลากหลาย กลยุทธ์ความเข้มข้น การแบ่งเวลาและความคงอยู่ล่าสุด MATS มีกลไกย่อย ได้แก่ PM SM และ DM ดังที่ได้อธิบายรายละเอียดผ่านมาแล้ว จากผลการประเมินสมรรถนะของ MATS เทียบกับ ATS แบบเดี่ยว ด้วยปัญหาการหาค่าเหมาะที่สุดเชิงพื้นผิวทั้งสามฟังก์ชัน ที่ได้กล่าวถึงรายละเอียดไปแล้วในตอนต้น จึงสามารถกล่าวได้ว่าฮอลอกริซึมการค้นหาแบบคาบเชิงปรับตัวหลายวิถีที่ได้พัฒนาขึ้นใหม่นี้มีประสิทธิภาพการค้นหาที่สูงขึ้นกว่าฮอลอกริซึมการค้นหาแบบคาบเชิงปรับตัวแบบเดิมจริง ในกรณีที่ดีที่สุด MATS สามารถประหยัดเวลาการค้นหาได้ถึง 56% โดยประมาณ ในกรณีที่เลวที่สุด ประหยัดเวลาการค้นหาได้ 6% โดยประมาณ ทั้งนี้เพื่อยืนยันถึงความสามารถของ MATS หัวข้อถัดไปเป็นการประยุกต์ใช้ MATS เพื่อการระบุเอกลักษณ์ระบบพลวัตของระบบเพนดูลัมผวนกรรณ (cart plus pendulum system หรือ CPP)

บทที่ 4

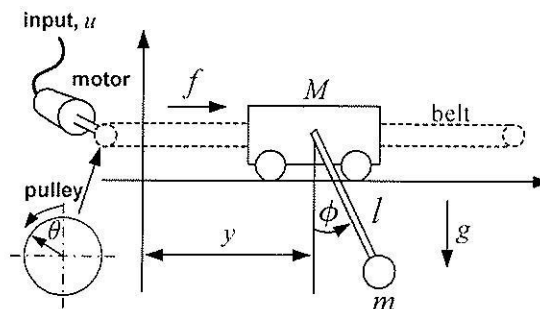
การประยุกต์ด้านระบบเอกลักษณ์ระบบพลวัต

4.1 กล่าวนำ

หัวข้อนี้อธิบายถึงการประยุกต์ใช้งานอัลกอริธึม MATS สำหรับการระบุเอกลักษณ์ระบบพลวัตซึ่งเป็นระบบลูกตุ้มนาฬิกาผกผัน (cart plus pendulum หรือ CPP) โดยมุ่งเน้นที่การระบุเอกลักษณ์แบบจำลองที่ใช้ประมาณค่าแรงกระทำต่อรถ (f) แรงดังกล่าวมีความไม่เป็นเชิงเส้นและได้รับผลกระทบจากตัวแปรพลวัตจำนวนมาก แบบจำลองทางคณิตศาสตร์ที่ถูกต้องแม่นยำมีลักษณะเป็นสมการอนุพันธ์บางส่วนหรือ PDE ที่มีความยุ่งยากมากในการพัฒนาตลอดจนมีข้อจำกัดต่อการนำไปใช้งานทางการควบคุม ดังนั้น งานวิจัยนี้จึงนำเสนอแบบจำลองทางคณิตศาสตร์ที่ประมาณแรง f ดังกล่าวตลอดจนการระบุเอกลักษณ์ของแบบจำลอง ดังรายละเอียดปรากฏในบทนี้

4.2 ระบบ CPP

ระบบ CPP ดังแผนภาพในรูปที่ 4.1 มีส่วนประกอบที่สำคัญ คือ ลูกตุ้ม (pendulum) ที่ต่อเข้ากับรถ (cart) ลูกตุ้มแกว่งไกวอยู่ในระนาบ x, y อย่างอิสระตามการเคลื่อนที่ของรถที่ถูกขับด้วยแรง f (นิวตัน) แรงดังกล่าวกำเนิดขึ้นจากการเคลื่อนที่ของมอเตอร์และส่งผ่านไปตามสายพาน โดยมีมุม ϕ คือ มุมการแกว่งไกวของลูกตุ้ม (เรเดียน) u คือ อินพุตของมอเตอร์ (โวลต์) y คือ ตำแหน่งของรถ (เมตร) M คือ มวลของรถ (กิโลกรัม) m คือ มวลของลูกตุ้ม (กิโลกรัม) l คือ ความยาวของก้านลูกตุ้ม (เมตร) ซึ่งพิจารณาว่าก้านลูกตุ้มเบามาก และ g คือ ความเร่งเนื่องจากแรงโน้มถ่วง (เมตร/วินาที²)



รูปที่ 4.1 แผนภาพแสดงระบบ CPP

แบบจำลองทางคณิตศาสตร์ของระบบ CPP ในงานวิจัยนี้มีรูปแบบไม่เป็นเชิงเส้น พัฒนาโดยอ้างอิงสมการของลากรานจ์ ซึ่งลากรานเจียน (L) นั้นเป็นผลต่างระหว่างพลังงานจลน์ และ

พลังงานศักย์ของระบบ (Bloch, Leonard and Marden, 2000) ดังแสดงในสมการที่ (4.1)

$$L(\phi, y, \dot{\phi}, \dot{y}) \triangleq K(\phi, y, \dot{\phi}, \dot{y}) - V(\phi) \quad (4.1)$$

เมื่อ $L(\cdot)$ คือ ลากรางเจียน $K(\cdot)$ คือ พลังงานจลน์ $V(\cdot)$ คือ พลังงานศักย์ พลังงานจลน์ $K(\cdot)$ เป็นผลรวมของพลังงานจลน์ของรถ (K_{cart}) และพลังงานจลน์ของลูกตุ้ม (K_{pend}) กล่าวคือ

$$K(\phi, y, \dot{\phi}, \dot{y}) = K_{cart} + K_{pend} \quad (4.2)$$

$$K(\phi, y, \dot{\phi}, \dot{y}) = \frac{1}{2} M (v_{cart})^2 + \frac{1}{2} m (v_{pend})^2 \quad (4.3)$$

เมื่อ $v_{cart} = \dot{y}$ ขณะที่ $v_{pend} = [v_{pend_horizontal}, v_{pend_vertical}]$ ซึ่งจะต้องพิจารณาดังนี้ เมื่อกำหนดให้ลูกตุ้มแกว่งไกวอยู่ด้านล่างของรถ $v_{pend} = [\dot{y} + l \cos(\phi)\dot{\phi}, l \sin(\phi)\dot{\phi}]$ และเมื่อกำหนดให้ลูกตุ้มแกว่งไกวอยู่ด้านบนของรถ $v_{pend} = [\dot{y} + l \cos(\phi)\dot{\phi}, -l \sin(\phi)\dot{\phi}]$ ดังนั้นสมการที่ (4.3) สามารถเขียนได้ใหม่ดังสมการที่ (4.4) และ (4.5) ซึ่งสามารถใช้ได้กับกรณีที่ลูกตุ้มแกว่งไกวอยู่ด้านล่างและด้านบนของรถ

$$K(\phi, y, \dot{\phi}, \dot{y}) = \frac{1}{2} M (\dot{y})^2 + \frac{1}{2} m [(\dot{y})^2 + 2l\phi\dot{y}\cos(\phi) + l^2(\dot{\phi})^2] \quad (4.4)$$

$$K(\phi, y, \dot{\phi}, \dot{y}) = \frac{1}{2} \begin{bmatrix} \dot{\phi} & \dot{y} \end{bmatrix} \begin{bmatrix} ml^2 & ml \cos(\phi) \\ ml \cos(\phi) & M + m \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{y} \end{bmatrix} \quad (4.5)$$

สำหรับพลังงานศักย์ $V(\cdot)$ เป็นผลรวมของพลังงานศักย์ของรถและพลังงานศักย์ของลูกตุ้ม แต่เนื่องจากพลังงานศักย์ของรถมีค่าเท่ากับศูนย์ (รถอยู่ในระดับอ้างอิง) ในกรณีที่ลูกตุ้มแกว่งไกวอยู่ด้านบนของรถ พลังงานศักย์อาจแสดงได้ดังสมการที่ (4.6) และ (4.7) สำหรับกรณีที่ลูกตุ้มอยู่ทางด้านล่างของรถ

$$V(\phi) = -mgl \cos(\phi) \quad (4.6)$$

$$V(\phi) = mgl \cos(\phi) \quad (4.7)$$

กำหนดให้ $\alpha = ml^2$, $\beta = ml$, $\gamma = (M + m)$ และ $D = mgl$ เมื่อลูกตุ้มแกว่งไกวอยู่ด้านล่างของรถและ $D = -mgl$ เมื่อลูกตุ้มแกว่งไกวอยู่ด้านบนของรถ ลากรานเจียน $L(\cdot)$ อาจเขียนแสดงได้ดังสมการที่ (4.8)

$$L(\phi, y, \dot{\phi}, \dot{y}) = \frac{1}{2} \left[\alpha (\dot{\phi})^2 + 2\beta \cos(\phi) \dot{\phi} \dot{y} + \gamma (\dot{y})^2 \right] \pm D \cos(\phi) \quad (4.8)$$

ดังนั้นอาจเขียนแสดงโมเมนต์เชิงมุม p_ϕ และโมเมนต์เชิงเส้น p_y ได้ดังสมการที่ (4.9) และ (4.10) ตามลำดับ

$$p_\phi = \frac{\partial L}{\partial \dot{\phi}} = \alpha \dot{\phi} + \beta \cos(\phi) \dot{y} \quad (4.9)$$

$$p_y = \frac{\partial L}{\partial \dot{y}} = \gamma \dot{y} + \beta \cos(\phi) \dot{\phi} \quad (4.10)$$

สมการการเคลื่อนที่ของลากรานจ์สำหรับระบบ CPP ประกอบด้วยสองสมการ คือ (4.11) และ (4.12)

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\phi}} - \frac{\partial L}{\partial \phi} = 0 \quad (4.11)$$

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{y}} = f \quad (4.12)$$

ซึ่งสามารถเขียนแสดงใหม่ได้ดังสมการที่ (4.13) และ (4.14) ตามลำดับ

$$\alpha \ddot{\phi} + \beta \cos(\phi) \ddot{y} + D \sin(\phi) = 0 \quad (4.13)$$

$$\gamma \ddot{y} + \beta \cos(\phi) \ddot{\phi} - \beta (\dot{\phi})^2 \sin(\phi) = f \quad (4.14)$$

เมื่อแทนค่า α β γ และ D ลงในสมการที่ (4.13) และ (4.14) และพิจารณากรณีที่ลูกตุ้มแกว่งไกวอยู่ด้านล่างของตัวรถ จะได้แบบจำลองทางคณิตศาสตร์ของระบบ CPP ในรูปของสมการอนุพันธ์ไม่เชิงเส้นดังนี้

$$\ddot{\phi} = \frac{f \cos(\phi) + 0.5ml \sin(2\phi)(\dot{\phi})^2 + (M + m)g \sin(\phi)}{I[m \cos^2(\phi) - (M + m)]} \quad (4.15)$$

$$\ddot{y} = \frac{f + 0.5mg \sin(2\phi) + ml \sin(\phi)(\dot{\phi})^2}{[(M + m) - m \cos^2(\phi)]} \quad (4.16)$$

4.3 การระบุเอกลักษณ์ระบบ CPP

การระบุเอกลักษณ์ระบบ CPP สำหรับงานวิจัยนี้ อาศัยแบบจำลองทางคณิตศาสตร์ของระบบ CPP ตามสมการที่ (4.15) และ (4.16) และการประมาณแรง f ด้วยฟังก์ชันไม่เป็นเชิงเส้นตามสมการที่ (4.17) ทั้งนี้ค่า θ, y, ϕ และ u สามารถอ่านได้จากตัวตรวจรู้ที่ติดตั้งบนชุดทดลองที่แสดงด้วยแผนภาพตามรูปที่ (4.1)

$$f(\theta, y, \phi, u) = A\ddot{\theta} + B\dot{\theta} + C\theta + D\ddot{y} + E\dot{y} + Fy + G\ddot{\phi} + H\dot{\phi} + I\phi + \dots \\ \dots Je^{Ku} + Lu^2 + Mu + Nu^{-1} + O \quad (4.17)$$

โดยที่ A, B และ C สะท้อนความสัมพันธ์ของการขจัด ความเร็ว และความเร่งเชิงมุมของโรเตอร์ต่อแรงกระทำต่อรถ f

D, E และ F สะท้อนความสัมพันธ์ของการขจัด ความเร็ว และความเร่งเชิงเส้นของรถต่อแรงกระทำต่อรถ f

G, H และ I สะท้อนความสัมพันธ์ของการขจัด ความเร็ว และความเร่งเชิงมุมของเพนควิล์มต่อแรงกระทำต่อรถ f

J, K, L, M และ N สะท้อนความสัมพันธ์ของแรงดันอินพุตของมอเตอร์ไฟฟ้ากระแสตรงต่อแรงกระทำต่อรถ f

O แทนค่าคงค้างใด ๆ (arbitrary residual) ต่อแรงกระทำต่อรถ f

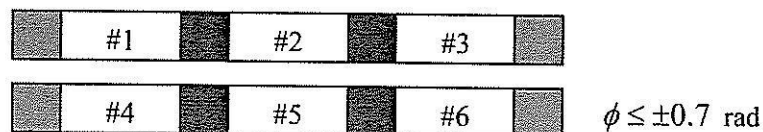
รูปแบบของแรง f ที่สมมุติขึ้นนี้ เป็นเชิงทฤษฎีตามการพิจารณาบนข้อเท็จจริงที่แรง f ได้รับการถ่ายทอดผ่านสายพานอ่อนตัว จึงมีปฏิสัมพันธ์กับส่วนประกอบต่าง ๆ ในระบบ รูปแบบดังกล่าวนี้ สามารถทดแทนแบบจำลองในกลุ่มอนุกรมอนันต์ (infinite series) ได้

การระบุเอกลักษณ์ระบบ CPP ด้วยการประมาณแรง f ที่ไม่เป็นเชิงเส้น พึ่งพาข้อมูลจากสองส่วน คือ จากการทดลองระบบจริงและจากการคำนวณเชิงตัวเลข ซึ่งความเหมาะสมของแบบจำลองกล่องเทาที่ได้ จะพิจารณาจากผลรวมกำลังสองค่าคลาดเคลื่อน (sum of square error หรือ SSE) ทำหน้าที่เป็นตัวชี้ ยังมีค่า SSE ต่ำ แบบจำลองยิ่งเหมาะสมมากกับข้อมูลที่ได้ออกมาทดลอง

4.3.1 การเตรียมข้อมูลจากผลการทดลองจริง

ตามหลักการระบุเอกลักษณ์ระบบ ก่อนนำข้อมูลที่ได้จากการทดลองเข้ากระบวนการระบุเอกลักษณ์ ควรตัดข้อมูลส่วนต้นและส่วนปลายออก ให้เหลือเฉพาะส่วนกลางที่มากพอสำหรับดำเนินการระบุเอกลักษณ์และการตรวจสอบความถูกต้องของแบบจำลอง (validation) ซึ่งโดยทั่วไป ข้อมูลที่จะใช้ระบุเอกลักษณ์จะใช้ประมาณร้อยละ 60 - 80 ส่วนที่เหลือใช้ประเมินความแม่นยำของแบบจำลอง

ข้อมูลสำหรับการระบุเอกลักษณ์ครั้งนี้ ได้จากการทดลองที่มีมุมแกว่งไกวของลูกตุ้ม $\phi \leq \pm 0.7$ เรเดียน ข้อมูลดิบที่บันทึกได้ถูกตัดส่วนหัวและท้ายออกตามหลักการเตรียมข้อมูลทางสถิติ แล้วนำข้อมูลดิบที่ได้แต่ละชุดมาแบ่งออกเป็นสามช่วง ช่วงละ 250 ค่า แต่ละช่วงมีระยะห่างของข้อมูล 50 ค่า ด้วยอัตราสุ่มซ้ำตัวอย่าง $T_s=0.15$ วินาที ต่อ 1 ตัวอย่าง ทำให้ได้ข้อมูลทั้งหมด 6 ช่วง ช่วงละ 250 ค่า ดังแผนภาพแทนชุดข้อมูลรูปที่ 4.2



รูปที่ 4.2 การเตรียมข้อมูลดิบสำหรับการระบุเอกลักษณ์ระบบ

ข้อมูลช่วงที่ 2-5 จำนวน 4 ช่วง จากทั้งหมด 6 ช่วง ซึ่งคิดเป็นร้อยละ 66.67 ของข้อมูลทั้งหมด ถูกใช้เพื่อการระบุเอกลักษณ์ (identification) ส่วนข้อมูลที่เหลือได้แก่ช่วงที่ 1 และ 6 ซึ่งคิดเป็นร้อยละ 33.33 ของข้อมูลทั้งหมด ถูกใช้เพื่อการประเมิน (validation) ความแม่นยำของแบบจำลองสอดคล้องกับหลักการแบ่งข้อมูลเพื่อการระบุเอกลักษณ์

4.3.2 การระบุเอกลักษณ์ระบบพลวัตของ CPP ด้วย MATS

การระบุเอกลักษณ์ระบบดำเนินการด้วยอัลกอริทึม MATS ในการประเมินค่าฟังก์ชันวัตถุประสงค์ต้องคำนวณเพื่อการจำลองผลระบบ CPP การแก้สมการที่ (4.15) และ (4.16) อาศัยระเบียบวิธีรุงกูดอันดับ 4 ดังปรากฏใน Mathew, J.H. and Fink, K.D., (2004) ร่วมกับข้อมูลที่ได้จากการทดลองที่ผ่านการเตรียมดังรูปที่ 4.2 เพื่อหาค่าของ $\theta, \dot{\theta}, \ddot{\theta}, \phi, \dot{\phi}, \ddot{\phi}, y, \dot{y}, \ddot{y}, u$ โดยการประมวลผลบนเครื่องคอมพิวเตอร์ตั้งโต๊ะ โปรแกรมพัฒนาขึ้นด้วย MATLAB™ ต่อไปนี้เป็น การอธิบายถึง การสร้างฟังก์ชันวัตถุประสงค์และการตั้งค่าพารามิเตอร์การค้นหาของ MATS

การสร้างฟังก์ชันวัตถุประสงค์ถูกเขียนด้วย MATLAB™ เช่นเดียวกับ MATS โดยพิจารณาที่ผลรวมค่าความคลาดเคลื่อนกำลังสอง (sum square of error หรือ SSE) ซึ่งได้จากการเปรียบเทียบกัน

ระหว่างมุมแกว่งไกวของลูกตุ้มที่ได้จากการคำนวณ (ϕ_{comp}) กับมุมแกว่งไกวของลูกตุ้มที่ได้จากการวัด (ϕ_{meas}) แสดงดังสมการ (4.18)

$$SSE = \sum_{i=1}^k (\phi_{comp,i} - \phi_{meas,i})^2 \quad (4.18)$$

การคำนวณค่า ϕ_{comp} เริ่มต้นด้วยด้วยการกำหนดค่า f ตามสมการที่ (4.17) แล้วแทนลงในสมการที่ (4.15) และ (4.16) หลังจากนั้นจึงแก้สมการด้วยระเบียบวิธีรุงเงกุตตา (Runge-Kutta method) เพื่อหาคำตอบของแบบจำลอง นั่นคือค่า ϕ_{comp} ส่วนค่า ϕ_{meas} ได้จากการผลการทดสอบจริง ขั้นตอนการคำนวณค่า SSE สามารถเขียนสรุปได้ดังรูปที่ 4.3

The procedures to calculate SSE can be summarized as follows:

STEP 1: Loading of arguments
Load 15 parameters of f as input arguments.

STEP 2: Simulating the CPP system
Substitute the 15 parameter to f and solve dynamic equations via RK4 method.
Return the solution, ϕ_{comp} .

STEP 3: Outputting and exiting
Load measured data, ϕ_{meas} , and then calculate $SSE(\phi_{comp}, \phi_{meas})$ as output argument and then exit.

รูปที่ 4.3 ขั้นตอนการคำนวณค่า SSE

การระบุเอกลักษณ์ตามขั้นตอนดังกล่าวข้างต้น เริ่มต้นด้วย $\phi_{meas,2i}$ (ข้อมูลช่วงที่ 2 เพื่อการระบุเอกลักษณ์) จะได้ $\phi_{comp,2i}$ และ SSE_{2i} แล้วดำเนินการทำนองเดียวกันกับข้อมูลช่วงที่ 3 4 และ 5 ไปตามลำดับจะได้ค่า SSE_{3i} , SSE_{4i} และ SSE_{5i} ส่วนของการตรวจสอบความแม่นยำของแบบจำลองอาศัยวิธีการเดียวกันกับการระบุเอกลักษณ์ แต่แตกต่างกันที่ช่วงข้อมูล โดยจะใช้ข้อมูลช่วงที่ 1 และช่วงที่ 5 สิ่งที่ได้ก็คือ SSE_{1v} และ SSE_{6v} ตามลำดับ กระบวนการดังรูปที่ 4.3 จะถูกเรียกใช้ทั้งหมด 6 ครั้ง จึงจะได้ SSE ครบทั้งหกค่า

เป้าหมายของการแก้ปัญหาการระบุเอกลักษณ์ครั้งนี้ คือ การหาคำตอบที่เป็นชุดพารามิเตอร์ของแรง f ที่เหมาะสมกับแบบจำลองมากที่สุด แต่ตัววัด (measures) ได้เลือกใช้ค่า SSE ซึ่งมีด้วยกันหกค่า จึงจัดเป็นปัญหาการหาค่าเหมาะที่สุดแบบหลายวัตถุประสงค์ และทุกวัตถุประสงค์ก็ต้องการค่า SSE ค่าที่ต่ำที่สุด จึงดำเนินการสร้างฟังก์ชันวัตถุประสงค์เพื่อให้ MATS ดำเนินการค้นหา ด้วยการผนวกค่า SSE ทั้งหมดค่าด้วยสมการผลรวมพีชคณิตเชิงเส้น แล้วถ่วงน้ำหนักด้วยค่า α_i , $i = 1, \dots, 6$ ดังสมการที่ (4.19)

$$J = \{\alpha_2 SSE_{2i} + \alpha_3 SSE_{3i} + \alpha_4 SSE_{4i} + \alpha_5 SSE_{5i}\}_{ident} + \{\alpha_1 SSE_{1v} + \alpha_6 SSE_{6v}\}_{valid} \quad (4.19)$$

กำหนดให้ค่าถ่วงน้ำหนักเท่ากัน คือ $\alpha_i = 1/6$, $i = 1, \dots, 6$ มีความหมายเดียวกันกับว่าค่า J คือ ค่าเฉลี่ยค่าของ SSE ทั้งหมดนั่นเอง การได้ค่า J ยิ่งน้อยย่อมแสดงว่าโดยเฉลี่ยแล้วค่า SSE ทั้งหมดมีค่าน้อย ๆ ด้วย ดังนั้นการมุ่งค้นหาค่า J น้อย ๆ ย่อมทำให้กระบวนการระบุเอกลักษณ์ได้ผลที่ดีตามไปด้วยนั่นเอง เพราะ J เป็นตัวแทนของทั้งส่วนของการระบุเอกลักษณ์แบบจำลอง - *ident* และส่วนของการตรวจสอบความถูกต้องแบบจำลอง - *valid* ด้วยในเวลาเดียวกัน เมื่อได้ฟังก์ชันวัตถุประสงค์ ดังสมการที่ (4.19) แล้ว จึงเป็นการง่ายที่จะให้อยู่ในรูปทั่วไปของปัญหาการหาค่าเหมาะที่สุด แบบหาค่าต่ำสุด - $\min J$

จากการศึกษาเบื้องต้น เพื่อแก้ปัญหาค่าเหมาะที่สุดชนิดนี้ด้วย ATS (Puangdownreong and Sujitjorn, 2006) พบว่าข้อมูลที่เหมาะสมสำหรับการตั้งค่าพารามิเตอร์ของ ATS ย่อย เป็นดังนี้ search_radius = 0.1, Number_neighb = 5 กลไกย้อนรอยการค้นหาตั้งพารามิเตอร์ดังนี้ kth_backward = 5, N_re_max = 5 กลไกปรับรัศมีการค้นหาตั้งพารามิเตอร์ไว้สามช่วงดังต่อไปนี้ (AR_1) if $J < 20$ search_radius = 0.01, (AR_2) if $J < 8$ search_radius = 0.001 และ (AR_3) if $J < 3.5$ radius = 0.0001 เงื่อนไขยุติการค้นหาที่กำหนดไว้ที่ max_count = 10000 หรือ $J < 3.1895$ การกำหนดจำนวนหน่วยค้นหาให้กับ MATS กรณีนี้พิจารณาจากขนาดของปัญหา ซึ่งพบว่าปัญหาแบบ 15 มิติ จึงเลือกจำนวนหน่วยค้นหาของ MATS เท่ากับจำนวนมิติของปัญหา ทั้งนี้เพื่อความสะดวกและให้ MATS มีโอกาสสำรวจครอบคลุมทุกมิติของปัญหา แต่อย่างไรก็ตามผู้อ่านมีอิสระในการเลือกจำนวนหน่วยค้นหาให้มากหรือน้อยกว่านี้ก็ได้ เพียงแต่ต้องคำนึงถึงผลที่ได้ทั้งแง่ของเวลาที่ใช้ในการค้นหาและคุณภาพของคำตอบที่ได้ด้วย เมื่อตัดสินใจเลือกใช้ MATS#15 เพื่อแก้ปัญหานี้แล้วจึงวางแนวทางการทำงานให้ PM ดังนี้ เนื่องจากมีอินพุต 15 ค่า คือ พารามิเตอร์ A ถึง O จึงกำหนดให้ ATS ย่อย แต่ละเส้นทางเริ่มต้นการค้นหาด้วยคำตอบที่สุ่มและคัดเลือกมาจากแต่ละมิติของปัญหานี้ โดยใช้คำตอบแกน (core solution, S_c) ทำหน้าที่เป็นแกนเพื่อการแบ่งปริภูมิ นอกจากนั้น ในแต่ละเส้นทางการค้นหาของ ATS ATS จะถูกกำหนดให้มีคำตอบเริ่มต้นจากการสุ่มค่าให้กับพารามิเตอร์ที่ต้องการค้นหาเพียงตัวเดียว ค่าพารามิเตอร์อื่น ๆ ที่เหลือใช้ค่า S_c ทั้งสิ้นในการเริ่มต้น เช่น ATS#1 พารามิเตอร์ A มีคำตอบเริ่มต้นจากการสุ่มเท่ากับ -0.3268 พารามิเตอร์ B-O จะมีคำตอบเริ่มต้นเป็น $S_c = 0.1$ ซึ่งจากการทดสอบด้วยการทำ “pre-search” ได้เลือก S_c และกำหนดปริภูมิการค้นหาทั้งหมด ดังปรากฏในตารางที่ 4.1 เมื่อทราบจำนวนหน่วยค้นหา ปริภูมิการค้นหา รวม และคำตอบแกนแล้ว PM ก็สามารถกำหนดปริภูมิการค้นหาย่อยเพื่อจัดหาคำตอบเริ่มต้นให้กับแต่ละหน่วยค้นหาย่อยได้แล้ว ตัวอย่างการแบ่งปริภูมิการค้นหาย่อยสำหรับ MATS#15 พร้อมคำตอบเริ่มต้นที่สุ่มคัดเลือกขึ้นได้จากแต่ละปริภูมิย่อย แสดงอยู่ในตารางที่ 4.2

ตารางที่ 4.1 คำตอบแทนและปฏิภูมิการค้นหาของ MATS#15

A-C	D-F	G-I	J-L	M-O	SSE(J)
cored solution or S_c					
A: 0.1	D: 0.1	G: 0.1	J: 0.1	M: 0.1	41.4846
B: 0.1	E: 0.1	H: 0.1	K: 0.1	N: 0.1	
C: 0.1	F: 0.1	I: 0.1	L: 0.1	O: 0.1	
search space of 15 parameters					
A:[5 -5]	D:[2 -2]	G:[6 -6]	J:[1 -1]	M:[1 -1]	-
B:[5 -5]	E:[2 -2]	H:[5 -5]	K:[1 -1]	N:[1 -1]	
C:[5 -1]	F:[2 -2]	I:[10 -10]	L:[2 -2]	O:[5 -5]	

ตารางที่ 4.2 ตัวอย่างปฏิภูมิการค้นหาย่อยของ ATS#1 #2 #4 #14 และ #15

A-C	D-F	G-I	J-L	M-O	SSE(J)
ATS#1					
<i>sub-search-space</i>					
A:[5 -5]	D:[0.1 0.1]	G:[0.1 0.1]	J:[0.1 0.1]	M:[0.1 0.1]	-
B:[0.1 0.1]	E:[0.1 0.1]	H:[0.1 0.1]	K:[0.1 0.1]	N:[0.1 0.1]	
C:[0.1 0.1]	F:[0.1 0.1]	I:[0.1 0.1]	L:[0.1 0.1]	O:[0.1 0.1]	
<i>initial solution randomly obtained</i>					
A: -0.3268	D: 0.1	G: 0.1	J: 0.1	M: 0.1	28.9233
B: 0.1	E: 0.1	H: 0.1	K: 0.1	N: 0.1	
C: 0.1	F: 0.1	I: 0.1	L: 0.1	O: 0.1	
ATS#2					
<i>sub-search-space</i>					
A:[0.1 0.1]	D:[0.1 0.1]	G:[0.1 0.1]	J:[0.1 0.1]	M:[0.1 0.1]	-
B:[5 -5]	E:[0.1 0.1]	H:[0.1 0.1]	K:[0.1 0.1]	N:[0.1 0.1]	
C:[0.1 0.1]	F:[0.1 0.1]	I:[0.1 0.1]	L:[0.1 0.1]	O:[0.1 0.1]	
<i>initial solution randomly obtained</i>					
A: 0.1	D: 0.1	G: 0.1	J: 0.1	M: 0.1	77.5293
B: 0.3011	E: 0.1	H: 0.1	K: 0.1	N: 0.1	
C: 0.1	F: 0.1	I: 0.1	L: 0.1	O: 0.1	
ATS#4					
<i>sub-search-space</i>					
A:[0.1 0.1]	D:[2 -2]	G:[0.1 0.1]	J:[0.1 0.1]	M:[0.1 0.1]	-
B:[0.1 0.1]	E:[0.1 0.1]	H:[0.1 0.1]	K:[0.1 0.1]	N:[0.1 0.1]	
C:[0.1 0.1]	F:[0.1 0.1]	I:[0.1 0.1]	L:[0.1 0.1]	O:[0.1 0.1]	
<i>initial solution randomly obtained</i>					
A: 0.1	D: -1.2395	G: 0.1	J: 0.1	M: 0.1	39.1685
B: 0.1	E: 0.1	H: 0.1	K: 0.1	N: 0.1	
C: 0.1	F: 0.1	I: 0.1	L: 0.1	O: 0.1	
ATS#14					
<i>sub-search-space</i>					
A:[0.1 0.1]	D:[0.1 0.1]	G:[0.1 0.1]	J:[0.1 0.1]	M:[0.1 0.1]	-
B:[0.1 0.1]	E:[0.1 0.1]	H:[0.1 0.1]	K:[0.1 0.1]	N:[1 -1]	
C:[0.1 0.1]	F:[0.1 0.1]	I:[0.1 0.1]	L:[0.1 0.1]	O:[0.1 0.1]	
<i>initial solution randomly obtained</i>					
A: 0.1	D: 0.1	G: 0.1	J: 0.1	M: 0.1	19.8972
B: 0.1	E: 0.1	H: 0.1	K: 0.1	N: 0.0152	
C: 0.1	F: 0.1	I: 0.1	L: 0.1	O: 0.1	
ATS#15					
<i>sub-search-space</i>					
A:[0.1 0.1]	D:[0.1 0.1]	G:[0.1 0.1]	J:[0.1 0.1]	M:[0.1 0.1]	-
B:[0.1 0.1]	E:[0.1 0.1]	H:[0.1 0.1]	K:[0.1 0.1]	N:[0.1 0.1]	
C:[0.1 0.1]	F:[0.1 0.1]	I:[0.1 0.1]	L:[0.1 0.1]	O:[5 -5]	
<i>initial solution randomly obtained</i>					
A: 0.1	D: 0.1	G: 0.1	J: 0.1	M: 0.1	42.7764
B: 0.1	E: 0.1	H: 0.1	K: 0.1	N: 0.1	
C: 0.1	F: 0.1	I: 0.1	L: 0.1	O: -0.0121	

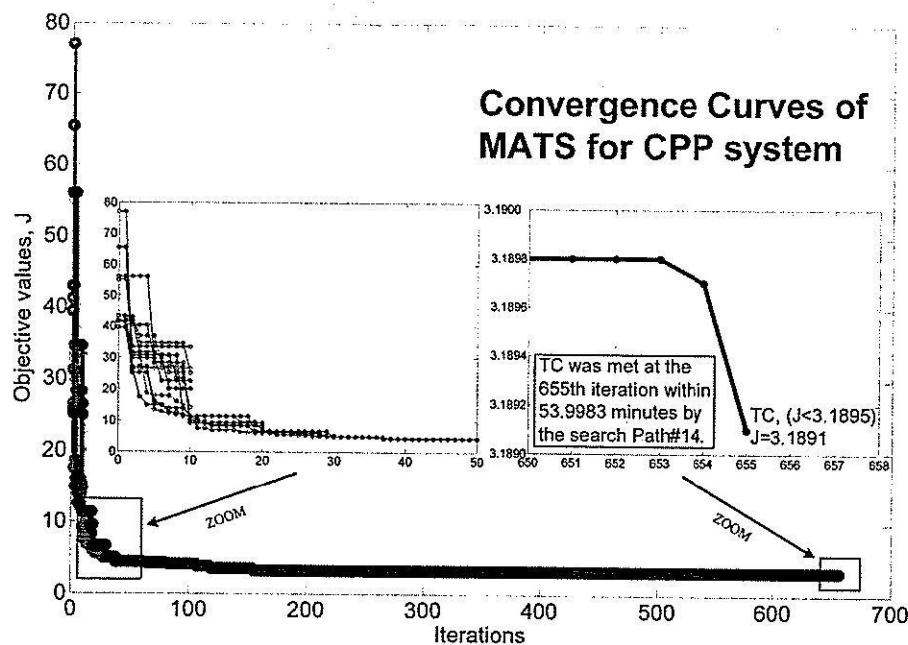
DM ตั้งค่าให้มีการเรียกใช้สี่ครั้งเพื่อลดจำนวนหน่วยค้นหาจาก 15 หน่วยค้นหา จนเหลือหนึ่งหน่วยค้นหา ดังรายละเอียดแสดงในตารางที่ 4.3 ข้อมูลแถวที่ 1 ในตารางแสดงรอบการค้นหาที่มีการเรียกใช้ DM ต่าง ๆ แถวที่ 2 แสดงจำนวนหน่วยค้นหาที่ถูกยกเลิกการค้นหาแต่ละครั้งของ DM และแถวที่ 3 แสดงจำนวนหน่วยค้นหาที่เหลือภายหลังการเรียกใช้ DM ครั้งนั้น ๆ

ตารางที่ 4.3 การตั้งค่าพารามิเตอร์ DM

Items	DM			
	1 st	2 nd	3 rd	4 th
iteration th which DM invoked	10	20	30	40
number of discarded search paths	7	4	2	1
number of remained search paths	8	4	2	1

4.3.3 ผลการระบุเอกลักษณ์ระบบ CPP จากการค้นหาด้วย MATS 15 เส้นทาง

เมื่อกำหนดค่าพารามิเตอร์ของทั้ง ATS และ MATS จนครบ จึงดำเนินการค้นหาคำตอบวงกว้างตามที่ต้องการ ได้ผลการค้นหาของ MATS#15 แสดงอยู่ในรูปกราฟการลู่เข้าหาคำตอบวงกว้าง ดังรูปที่ 4.4 โดยมี ATSย่อยที่ 14 เป็นผู้พบคำตอบภายใต้เงื่อนไขยุติการค้นหาที่มีค่า $J < 3.1895$ ในรอบการค้นหาที่ 655 ด้วยเวลาการค้นหา 53.9983 นาที คำตอบวงกว้างที่ได้นั้น แสดงรายละเอียดอยู่ในตารางที่ 4.4 พร้อมด้วยค่า J และ SSE จากค่าพารามิเตอร์ที่ได้ ทำให้สามารถเขียนแสดงแรง f ได้ดังสมการที่ (4.20)



รูปที่ 4.4 กราฟการลู่เข้าหาผลเฉลยของ MATS กับปัญหาการระบุเอกลักษณ์ระบบพลวัต CPP

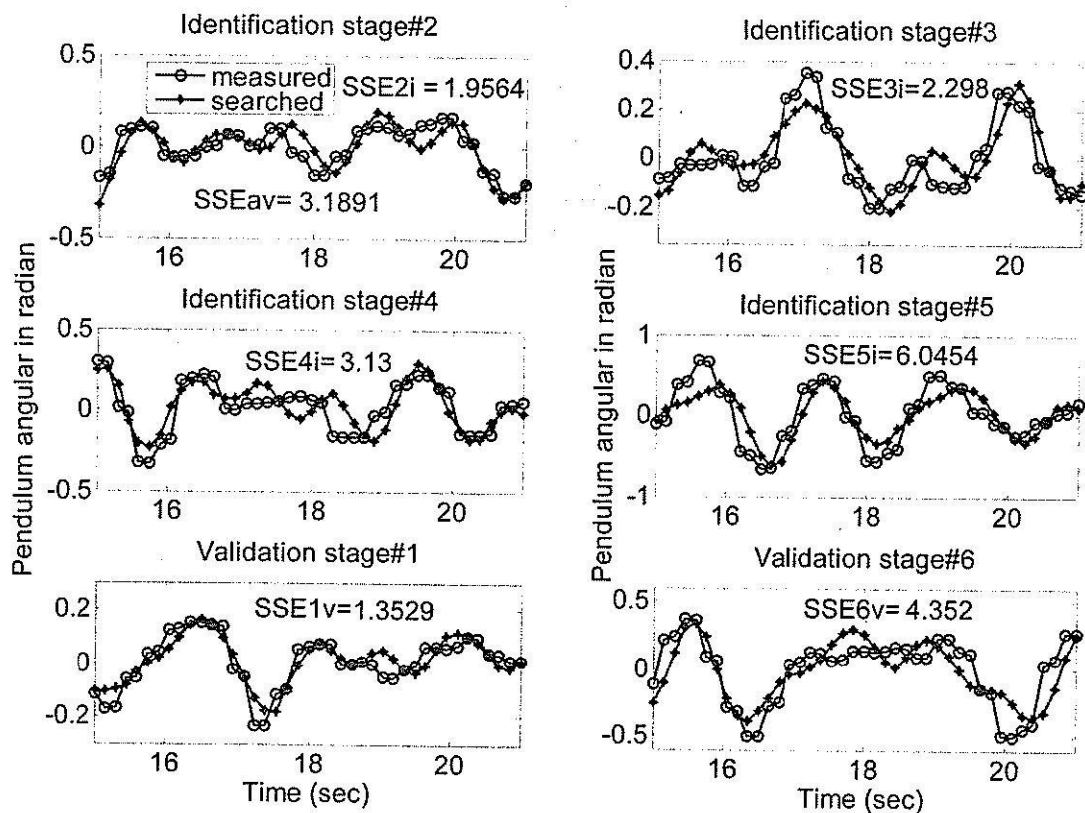
ตารางที่ 4.4 ค่าตอบที่ได้จากการค้นหาด้วย MATS#15

A-C	D-F	G-I	J-L	M-O	SSE		
					<i>J</i>	<i>ident</i>	<i>valid</i>
A: -0.4425	D: -1.3226	G: -3.7936	J: 0.3684	M: -0.0996	3.1891	1.9564 ^a	1.3529 ^c
B: 0.5408	E: 0.6382	H: -0.6304	K: -0.2778	N: -0.0069		2.2980 ^b	4.3520 ^f
C: 0.0466	F: 0.2141	I: -7.6637	L: 0.1446	O: -0.3913		3.1300 ^e	
						6.0454 ^d	

These SSEs with superscripts, a to f, are from data lot#2 #3 #4 #5 #1 and #6 respectively.

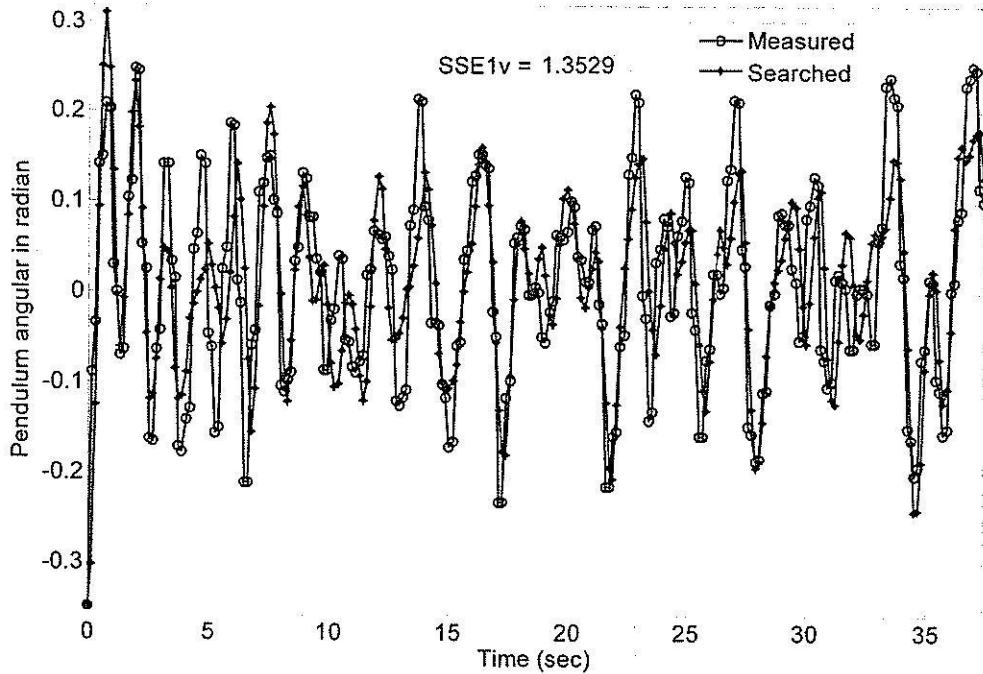
เมื่อนำสมการที่ (4.20) แทนลงในสมการที่ (4.15) และ (4.16) แล้วคำนวณหามุมแกว่งไกวของลูกตุ้มได้ และเมื่อนำผลที่ได้จากการคำนวณนี้วาดกราฟเปรียบเทียบกับผลที่ได้จากการทดสอบสามารถแสดงได้ดังรูปที่ 4.5 เฉพาะบางช่วงเวลาที่ตรงกัน โดยเส้น -o- ได้จากการวัด และเส้น -* ได้จากการคำนวณตามแบบจำลองทางคณิตศาสตร์ที่มีพารามิเตอร์ของแรง f ได้มาจากการค้นหา นั่นเอง

$$f(\theta, y, \phi, u) = -0.4425\ddot{\theta} + 0.5408\dot{\theta} + 0.0466\theta - 1.3226\ddot{y} + 0.6382\dot{y} + 0.2141y - 3.7936\ddot{\phi} - 0.6304\dot{\phi} - 7.6637\phi + 0.3684e^{-0.2778u} + 0.1446u^2 - 0.0996u - 0.0069u^{-1} - 0.3913 \quad (4.20)$$

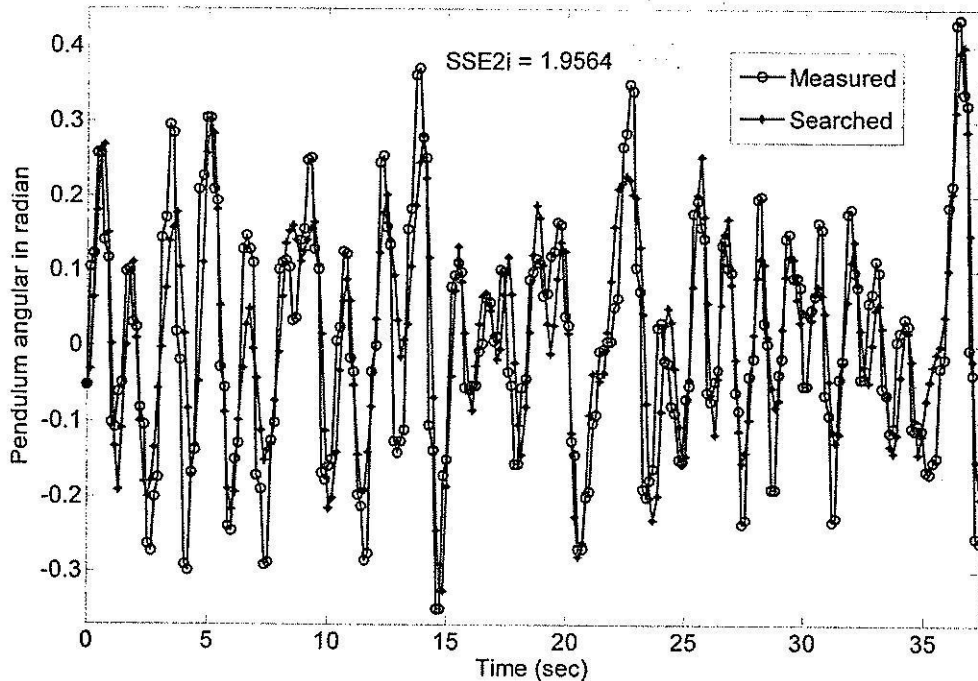


รูปที่ 4.5 ผลการระบุเอกลักษณ์ ที่ค่า SSE = 3.1891

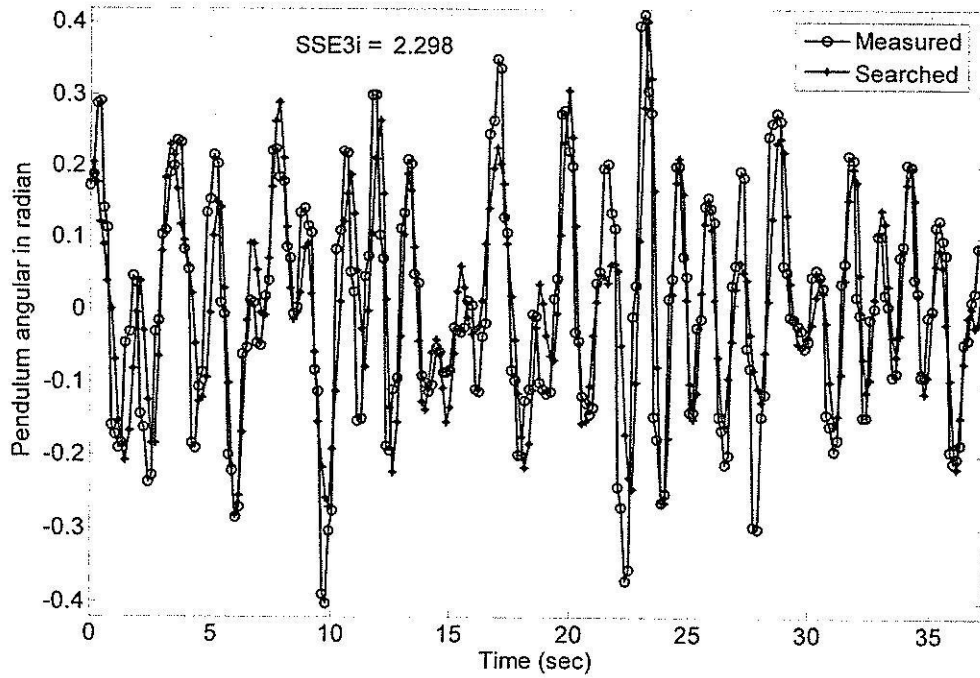
ภาพสัญญาณทั้งที่ได้จากการวัดและจากการระบุเอกลักษณ์ทั้งหกช่วงถูกวาดรวมกันเพื่อความสะดวกสำหรับการเปรียบเทียบ ได้รับการแสดงไว้ในรูปที่ 4.6 ซึ่งเห็นได้ว่าการระบุเอกลักษณ์ให้ระบบ CCP ด้วยการประยุกต์อัลกอริทึม MATS ให้ผลดีมาก และดีกว่าผลงานวิจัยเดิมที่เคยปรากฏมาก่อนแล้ว (Puangdownreong and Sujitjorn, 2006)



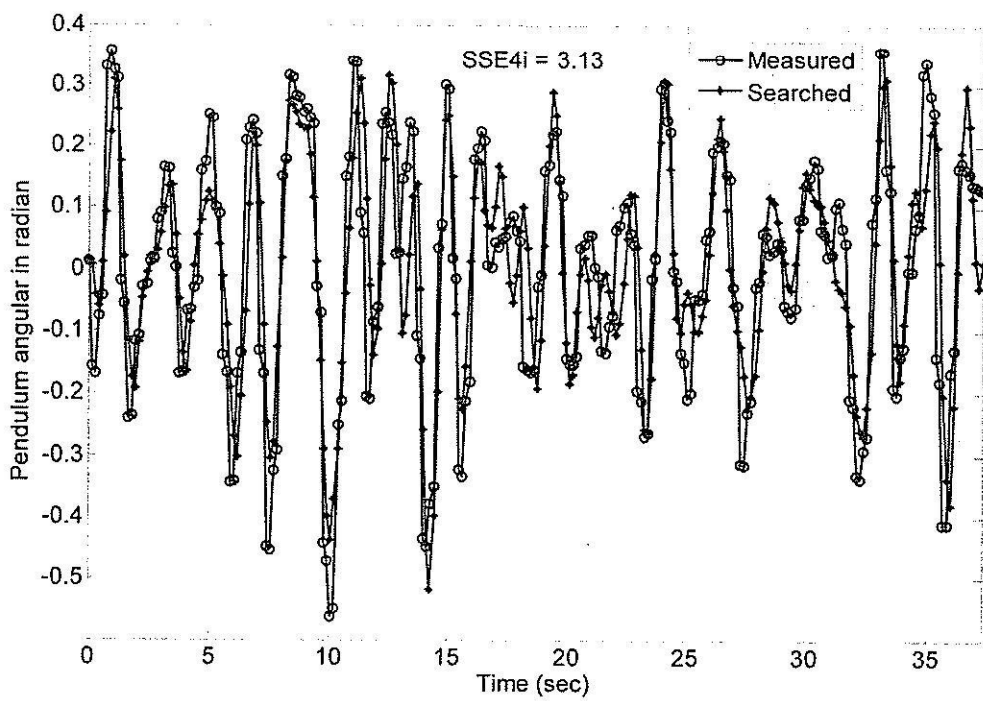
(ก)



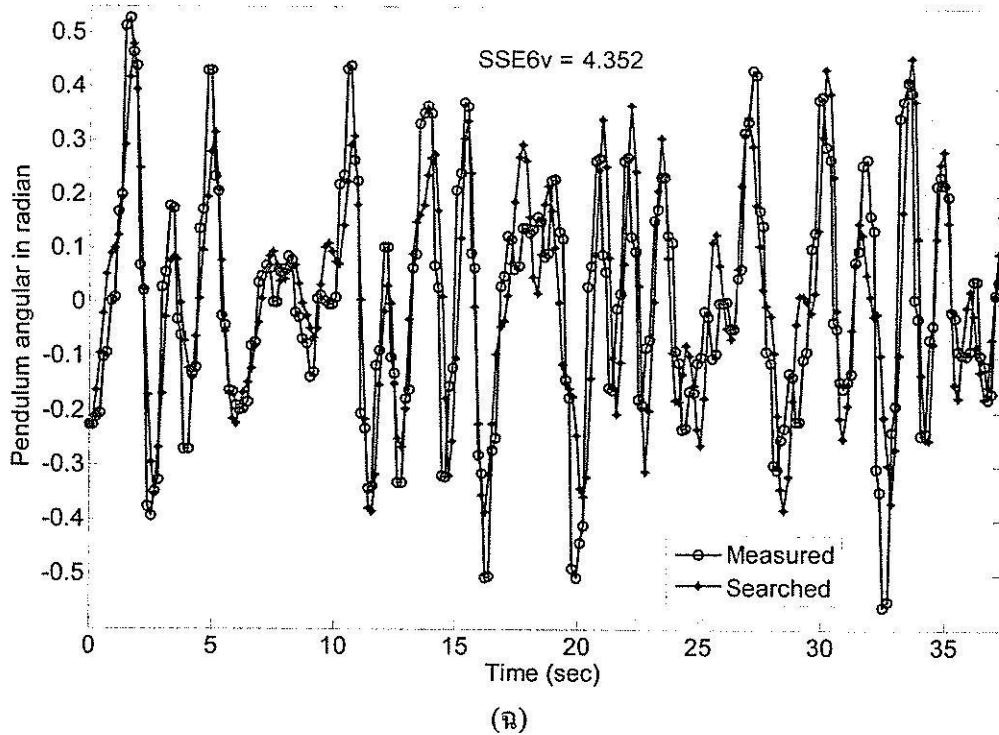
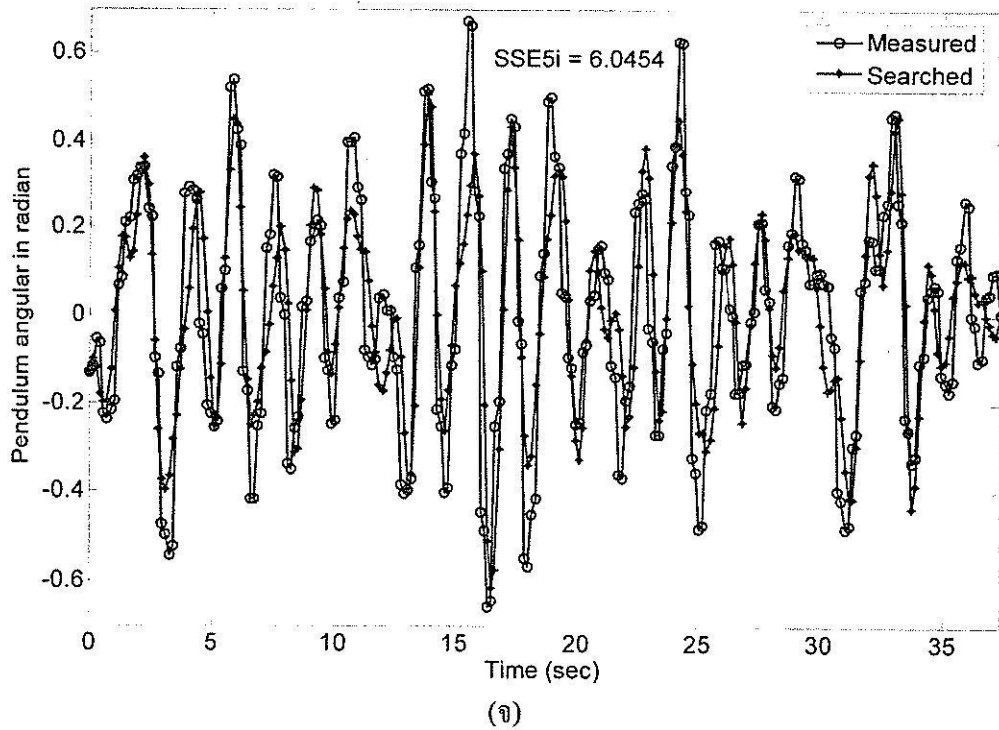
(ข)



(f)



(g)



รูปที่ 4.6 เปรียบเทียบสัญญาณจากการวัดและจากการระบุเอกลักษณ์ ตลอดทั้งช่วงข้อมูล 37.5 วินาที จำนวน 250 จุด แต่ละจุดห่างกัน 0.15 วินาที (ก) ช่วงข้อมูลที่ 1 (ข) ช่วงข้อมูลที่ 2 (ค) ช่วงข้อมูลที่ 3 (ง) ช่วงข้อมูลที่ 4 (จ) ช่วงข้อมูลที่ 5 และ (ฉ) ช่วงข้อมูลที่ 6 โดยช่วงข้อมูลที่ 2 ถึง 5 สำหรับการระบุเอกลักษณ์ ส่วนช่วงข้อมูลที่ 1 และ 6 สำหรับการประเมินความถูกต้อง

บทที่ 5

สรุป

5.1 สรุปผลการวิจัย

อัลกอริธึมการค้นหาแบบตามูเชิงปรับตัวหลายวิถี หรือ MATS ได้รับการพัฒนาขึ้นภายใต้แนวคิดใหม่เพื่อปรับปรุงอัลกอริธึมการค้นหาแบบตามูเชิงปรับตัว (ATS) ในระดับกระบวนการ (procedural level) ซึ่งกลไกหลักของอัลกอริธึมการค้นหาแบบตามูเชิงปรับตัว ทั้งกลไกย้อนรอยและกลไกปรับรัศมี ยังคงทำหน้าที่ได้ตามปกติและที่สำคัญคุณสมบัติการลู่เข้าหาผลเฉลยก็ยังคงอยู่เช่นเดิม แต่สิ่งที่เพิ่มเติมขึ้นมาได้แก่ กลไกแบ่งปริภูมิ (PM) กลไกลำดับการค้นหา (SM) และกลไกยกเลิกการค้นหา (DM) เพื่อมาทำหน้าที่บริหารและจัดการหน่วยค้นหาให้ทำการค้นหาคำตอบทั่วไปให้ใช้เวลาการค้นหาน้อยที่สุด MATS ที่พัฒนาขึ้นสามารถปรับใช้ได้ทั้งกับคอมพิวเตอร์ที่มีตัวประมวลผลตัวเดียวหรือหลายตัว การใช้งาน MATS มีความซับซ้อนขึ้นมาก จำเป็นต้องตั้งค่าพารามิเตอร์ให้แก่กลไก PM SM และ DM ดังรายละเอียดที่อธิบายไว้ในบทที่ 3 ทั้งนี้ เพื่อให้ได้สมรรถนะการค้นหาที่ดียิ่งกว่า ATS ตามปกติ จากการพัฒนา MATS บนโปรแกรม MATLAB และทดสอบกับปัญหาการหาค่าเหมาะที่สุดเชิงพื้นที่จำนวนสามฟังก์ชัน ได้แก่ ฟังก์ชัน โบฮาชอฟสกี ฟังก์ชันราสทริกิน และฟังก์ชันซีเคลฟอกซ์โฮล ผลการประเมินสมรรถนะเทียบกับอัลกอริธึมแบบตามูเชิงปรับตัวแบบเดี่ยว (ATS) แสดงให้เห็นชัดเจนว่าอัลกอริธึมใหม่สามารถทำการค้นหาให้รวดเร็วขึ้นได้ตามเป้าหมายของผู้วิจัยอย่างน่าพอใจ นั่นคือ สามารถลดเวลาการค้นหาลงได้ 6.35-56.79% อาจกล่าวได้ว่า ในภาพรวมเฉลี่ย MATS สามารถลดเวลาการค้นหาลงได้ประมาณ 26.26% เมื่อเทียบกับ ATS นอกจากนี้ยังได้ประยุกต์ใช้ MATS กับปัญหาจริงเพื่อการระบุเอกลักษณ์ระบบพลวัต ซึ่งก็คือระบบเพนดูลัมผนวงกรหรือ CPP ให้ผลเป็นที่น่าพึงพอใจมากดังการนำเสนอในบทที่ 4

5.2 ข้อเสนอแนะ

1. การพัฒนาเงื่อนไขการทำงานของบางกลไกหรือทั้งหมดของ MATS ซึ่งก็คือ PM SM และ DM สามารถทำให้ประสิทธิภาพการทำงานของ MATS สูงขึ้นได้ จึงถือว่าเป็นแนวทางที่จะพัฒนาต่อยอดงานวิจัยนี้ได้ในอนาคต เช่นการปรับปรุง PM ให้มีความสามารถในการปรับตัว (adaptive) เพื่อให้การสำรวจปริภูมิการค้นหาทำได้กว้างขวาง ท้วถึง และรวดเร็ว การปรับปรุง DM ให้เป็น “การค้นหาเฉพาะถิ่นที่มีการแนะแนวทาง (Guided Local Search)” เพื่อที่จะให้มีประสิทธิภาพยิ่งขึ้นในการบ่งชี้เส้นทางการค้นหาที่มีคุณภาพสูง เป็นต้น

2. เพื่อผลักดันให้มีการใช้งานอัลกอริธึมนี้ในภาคอุตสาหกรรมหรือเชิงพาณิชย์มากยิ่งขึ้น
ควรที่จะต้องพัฒนาตัวโปรแกรมอัลกอริธึมการค้นหาแบบตามเชิงปรับตัวหลายวิธีให้มีความ
กระชับและง่ายต่อการใช้งานมากกว่านี้ และอาจเผยแพร่เป็นสาธารณะประโยชน์

เอกสารอ้างอิง

- Battiti, R. and Tecchiolli, G., (1994). The reactive tabu search. **ORSA Journal on Computing**. 6(2): 126-140.
- Bloch, A.M., Leonard, N.E., Marsden, J.E., (2000). Controlled lagrangians and the stabilization of mechanical system I: the first matching theorem. **IEEE Transaction on Automatic Control**. 45: 2253-2270.
- Chen, J-S., Pan, J. C-H. and Wu, C-K., (2006). Minimizing makespan in reentrant flow-shops using hybrid tabu search. **Int. J Adv Manuf Technol**. Springer-Verlag London. (online).
- Crainic, T.G., Toulouse, M., and Gendreau, M., (1996). Parallel asynchronous tabu search for multicommodity location-allocation with balancing requirements. **Annals of Operations Research**. 63: 277-299.
- Franca, P. M., Sosa, N. M., and Pureza, V. (1999). An adaptive tabu search algorithm for the capacitated clustering problem. **International Transaction in Operation Research**. 6: 665-678.
- Glover, F. (1978). Parametric branch and bound. **The International Journal of Management Science**. 6: 145-152.
- Glover, F. (1986). Future paths for integer programming and link to artificial intelligence. **Computers and Operation Research**. 13: 533-549.
- Glover, F. (1989). Tabu search – Part I. **ORSA Journal on Computing**. 1: 190-206.
- Glover, F. (1990). Tabu search – Part II. **ORSA Journal on Computing**. 2: 4-32.
- Glover, F. and Laguna, M. (1997). **Tabu Search**. Kluwer Academic Publishers. Norwell.
- Gong, D., Zhou, Y., Guo, X., Xiaoping M, and Ming Li, (2002). Study on an adaptive tabu search genetic algorithm. In **Proceedings of the 4th World Congress on Intelligent Control and Automation**. (pp.3063-3065).
- Kalinli, A., (2004). Component value selection for active filters using parallel tabu search algorithm. **Int. J. Electron. Commun. (AEU)**. 60: 85-92.
- Kochetov, Y. and Goncharov, E., (2000). Behavior of a probabilistic tabu search algorithm for the multi stage uncapacitated facility location problem. In **Proceeding of Operation Research** . (pp. 65-70)
- Kolahan, F. and Liang, M., 1998. (1998). An adaptive tabu search approach to JIT sequencing with variable processing times and sequence-dependent setups. **European Journal of Operational Research**. 109: 142-159.

- Liu, Yo., Liu, Ya., Wang, L. and Chen, K., (2005). A hybrid tabu search based clustering algorithm. **Lecture Notes in Artificial Intelligence**. Springer-Verlag Berlin Heidelberg. 3682: 66-77.
- Mathew, J.H. and Fink, K.D., (2004). Numerical methods using MATLAB. PRT Prentice-Hall.
- Mori, H. and Tani, H., (2003). Fast hybrid method of parallel tabu search and ordinal optimization for on-line capacitor control in distribution systems. **IEEE General Meeting of Power Engineering Society**. 3: 1553-1558.
- Nowicki, E. and Smutnicki, C., (2005). An advanced tabu search algorithm for the job shop problem. **Journal of Scheduling**. 8: 145-159.
- Ongsakul, W. and Bhasaputra, P., (2002). Optimal power flow with FACTS devices by hybrids TS/SA approach. **Journal of Electrical & Energy Systems**. 24: 851-857.
- Pedroso, J. P. and Kubo, M., (2005). Hybrid tabu search for lot sizing problem. **Lecture Notes in Computer Science**. Springer-Verlag Berlin Heidelberg. 3636: 66-77.
- Puangdownreong, D., Kulworawanichpong, T., Sujitjorn, S., (2004). Finite convergence and performance evaluation of adaptive tabu search. **Lecture Notes in Computer Science**. Springer-Verlag Heidelberg. (online)
- Puangdownreong, D. and Sujitjorn, S., (2006). Image approach to system identification. **WSEAS Transaction on Systems**. 5: 930-938.
- Sait, S. M., Youssef, H., Barada, H. R. and Ahmad A.-Y., (2000). Parallel tabu search algorithm for VLSI standard-cell placement. **IEEE International Symposium on Circuits and Systems ISCAS 2000**. (pp.581-584).
- Swarnkar, R. and Tiwari, M.K., (2004). Modeling machine loading problem of FMSs and its solution methodology using a hybrid tabu search and simulated annealing-based heuristic approach. **Journal of Robotics and Computer-Integrated Manufacturing**. 20: 199-209.
- Tahir, M. A., Bouridane A. and Kurugollu, F., (2007). Simultaneous feature selection and feature weighting using hybrid tabu search/K-nearest neighbor classifier. **Journal of Pattern Recognition Letters**. 28: 438-446.
- Talbi, E.G., Hafidi, Z. and Geib, J-M. (1998). A parallel adaptive tabu search approach. **Parallel Computing**. 24: 2003-2019.
- Victoire, T. A. A. and Jeyakumar, A. E., (2006). A tabu search based hybrid optimization approach for a fuzzy modeled unit commitment problem. **Journal of Electric Power Systems Research**. 76: 413-425.
- Wassan, N., (2006). A reactive tabu search for the vehicle routing problem. **Journal of the ORSA Journal on Society**. 57: 111-116.

- Xu, J., Chiu, S.Y. and Glover, F., (1997). Probabilistic tabu search for telecommunications network design. **Journal of Combinatorial Optimization Theory and Practice.** 1: 69-94.
- Zhang, D., Fu, Z. and Zhang, L., (2006). An improved TS algorithm for loss-minimum reconfiguration in large-scale distribution systems. **Journal of Electric Power Systems Research.** 77: 685-694.

ภาคผนวก ก.
บทความที่ได้รับการตีพิมพ์

รายชื่อบทความที่ได้รับการตีพิมพ์เผยแพร่ในระดับนานาชาติ

1. J. Kluabwang and S. Sujitjorn, "*Tabu Search: Its modified forms and management agent.*", *Recent Patents on Electrical Engineering*, vol. 2, no. 2, pp. 135-145, June 2009.
2. J. Kluabwang, D. Puangdownreong and S. Sujitjorn, "*Management Agent for Search Algorithms with Surface Optimization Applications.*", *WSEAS Trans. on computers*, vol. 7, no. 6, pp. 791-803, June 2008.
3. J. Kluabwang, D. Puangdownreong and S. Sujitjorn, "*Management Agent for Search Algorithms.*", In Proc. of 12th WSEAS International Conference on COMPUTERS (ICCOMP'08), Included in ISI/SCI Web of Science and Web of Knowledge, Heraklion, Crete Island, Greece, July 22-25, Part II, pages 375-382, 2008.

ประวัติผู้วิจัย

นาวาอากาศโท ดร.สราวุฒิ สุจิตจร สำเร็จการศึกษาวิศวกรรมไฟฟ้า (เกียรตินิยมอันดับ 1) จากโรงเรียนนายเรืออากาศ เมื่อ พ.ศ.2527 และ PhD (Electronic and Electrical Engineering) จาก University of Birmingham, UK เมื่อ พ.ศ. 2530 ปัจจุบันดำรงตำแหน่ง ศาสตราจารย์ ประจำสาขา วิชาวิศวกรรมไฟฟ้า สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี มีความชำนาญด้าน control system, applied signal processing, AI มีความชำนาญพิเศษ ด้านเครื่องสายไทย การเลี้ยง และฝึกสุนัข มีผลงานหนังสือและตำรา 4 รายการ บทความวิจัยกว่า 120 รายการ และได้จดสิทธิบัตรการประดิษฐ์ไว้ 15 ผลงาน