

วิธีอุปชันจริงในการหาเวลาหยุดเหมาะสมสุดของขั้นตอนวิธีเชิงพันธุกรรม



นางสาวสุนิสา रिมเจริญ

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย
วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรดุษฎีบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2551

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

REAL OPTIONS APPROACH TO FINDING OPTIMAL STOPPING TIME
IN GENETIC ALGORITHMS



Miss Sunisa Rimcharoen

ศูนย์วิทยทรัพยากร

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

Chulalongkorn University

Academic Year 2008

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์

วิธีออกแบบจริงในการหาเวลาหยุดเหมาะสมสุดของชั้นตอนวิธีเชิง
พันธุกรรม

โดย

นางสาวสุนิสา ริมเจริญ

สาขาวิชา

วิศวกรรมคอมพิวเตอร์


อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

ผู้ช่วยศาสตราจารย์ ดร.ดาริชา สุธีวงศ์


อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

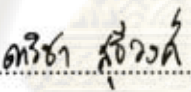
ศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา

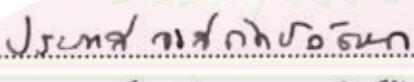
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้หัวข้อวิทยานิพนธ์ฉบับนี้
เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาตรีบัณฑิต


..... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศธีรวงค์)

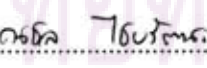
คณะกรรมการสอบวิทยานิพนธ์


..... ประธานกรรมการ
(รองศาสตราจารย์ ดร.สมชาย ประสิทธิ์จตุระกุล)


..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ ดร.ดาริชา สุธีวงศ์)


..... อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม
(ศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา)


..... กรรมการ
(ศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล)


..... กรรมการภายนอกมหาวิทยาลัย
(รองศาสตราจารย์ ดร.ณชล ไชยรัตน์)

ศูนย์วิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สุนิสา ริมเจริญ : วิธีออปชันจริงในการหาเวลาหยุดเหมาะสมสุดของขั้นตอนวิธีเชิงพันธุกรรม (REAL OPTIONS APPROACH TO FINDING OPTIMAL STOPPING TIME IN GENETIC ALGORITHMS). อ.ที่ปรึกษาวิทยานิพนธ์หลัก : ผศ. ดร. คาริชา สุธีวงศ์, อ.ที่ปรึกษาวิทยานิพนธ์ร่วม: ศ. ดร.ประภาส จงสถิตย์วัฒนา, 82 หน้า

วิทยานิพนธ์นี้นำเสนอการใช้วิธีออปชันจริงในการหาเวลาหยุดเหมาะสมสุดของขั้นตอนวิธีเชิงพันธุกรรม วิธีออปชันจริงเป็นเครื่องมือประเมินความเสี่ยงสำหรับการตัดสินใจภายใต้ความไม่แน่นอนซึ่งถูกนำไปใช้วิเคราะห์การลงทุนในหลาย ๆ ด้าน แต่การนำมาใช้ในการตัดสินใจเรื่องการทำงานของอัลกอริทึมเป็นสิ่งที่วิทยานิพนธ์นี้นำเสนอ ผลการวิเคราะห์เวลาหยุดเหมาะสมสุดโดยวิธีออปชันจริงจะให้ข้อมูล 2 อย่าง คือ มูลค่าอัลกอริทึมซึ่งเป็นมูลค่าในการหาคำตอบของอัลกอริทึมภายใต้การตัดสินใจที่ดีที่สุด และเงื่อนไขการหยุดโดยกำหนดขอบเขตของคำตอบในรุ่นต่าง ๆ งานวิจัยนี้นำเสนอการใช้มูลค่าอัลกอริทึมที่ได้เป็นตัววัดเชิงปริมาณสำหรับเปรียบเทียบประสิทธิภาพของอัลกอริทึมในแง่ของความคุ้มค่าเมื่อคำนึงถึงต้นทุนการคำนวณเวลาที่จะได้รับคำตอบ และโอกาสที่จะพบคำตอบ ส่วนค่าขอบเขตการหยุดจะเป็นเงื่อนไขที่บอกว่าอัลกอริทึมควรจะหยุดหรือทำงานต่อ จากการทดลองพบว่าเมื่อขั้นตอนวิธีเชิงพันธุกรรมใช้เงื่อนไขการหยุดตามค่าขอบเขตนี้จะสามารถลดจำนวนครั้งในการประเมินค่าความเหมาะสมลงได้

ศูนย์วิทยทรัพยากร

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต สุนิสา ริมเจริญ
 สาขาวิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก คาริชา สุธีวงศ์
 ปีการศึกษา 2551 ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์ร่วม ประภาส จงสถิตย์วัฒนา

4871829521 : MAJOR COMPUTER ENGINEERING

KEY WORD : REAL OPTIONS / OPTIMAL STOPPING TIME / GENETIC ALGORITHM

SUNISA RIMCHAROEN : REAL OPTIONS APPROACH TO FINDING
OPTIMAL STOPPING TIME IN GENETIC ALGORITHMS. ADVISOR :
ASST. PROF. DARICHA SUTIVONG, Ph.D., CO-ADVISOR : PROF.
PRABHAS CHONGSTITVATANA, Ph.D., 82 pp.

This thesis proposes using a real-options approach to find an optimal stopping time of genetic algorithms. The real options methodology is an evaluation tool for making decision under uncertainty, such as analyzing various investments. Applying this approach to guide algorithms decisions is a novelty of this thesis. The optimal stopping time analysis using the real options technique offers us two pieces of information: an algorithm value, which is the value of algorithms in searching for a solution optimally, and a stopping criterion based on boundary of a fitness value in each generation. This research proposes using an algorithm value as a quantitative measure for comparing the efficiency of algorithms in terms of benefit. It takes a computational cost, time and the possibility of finding a solution into account. The bounds of a fitness value in each generation guide the algorithm on whether to terminate or to keep searching for a solution. The experimental results show that when the genetic algorithms follow the proposed boundary as a stopping criterion, the number of fitness evaluations can be reduced.

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

Department: Computer Engineering Student's Signature: *Sunisa Rimcharoen*

Field of Study: Computer Engineering Advisor's Signature: *Darich Sutivong*

Academic Year: 2008 Co-Advisor's Signature: *P. Chongstitvatana*

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา ผศ. ดร. ดาริชา สุธีวงศ์ และ ศ. ดร.ประภาส จงสถิตย์วัฒนา ที่ได้ให้คำแนะนำ ให้แนวทางการแก้ปัญหา แนะนำวิธีคิด และให้กำลังใจยามที่ผู้วิจัยรู้สึกท้ออย่างยิ่ง ตลอดหลายปีที่ผู้วิจัยได้มีโอกาสทำงาน กับทั้งสองท่าน เป็นประสบการณ์อันมีค่ายิ่ง หลายสิ่งหลายอย่างที่ได้เรียนรู้จากท่านทั้งสองทำ ให้ผู้วิจัยมีวันนี้ได้

ขอขอบคุณกรรมสอบวิทยานิพนธ์ทุกท่าน ได้แก่ รศ. ดร.สมชาย ประสิทธิ์จตุระกุล ศ. ดร.บุญเสริม กิจศิริกุล และ รศ. ดร.ณชล ไชยรัตนะ ในการตรวจแก้และให้คำแนะนำอันเป็น ประโยชน์ยิ่งต่องานวิจัย

ขอขอบคุณ Prof. Xin Yao สำหรับโอกาสอันดีที่ได้ไปศึกษาวิจัยอยู่ที่ The University of Birmingham ขอขอบคุณสำหรับคำแนะนำ และความช่วยเหลือทุก ๆ อย่าง ทั้ง ขณะที่อยู่อังกฤษและในเวลาต่อมา

ตลอดระยะเวลาการศึกษา ผู้วิจัยได้รับทุนสนับสนุนจากสำนักงานคณะกรรมการ การอุดมศึกษา ทุนพัฒนาอาจารย์สาขาขาดแคลน (ม.บูรพา) จึงขอขอบคุณมา ณ ที่นี้

วิทยานิพนธ์นี้สำเร็จได้จากความช่วยเหลือของบุคคลหลายท่านที่คอยให้ความ ช่วยเหลือ ให้คำแนะนำ ข้อคิดเห็นต่าง ๆ และให้กำลังใจด้วยดีตลอดมา ดังนี้ พี่คทา ประดิษฐ์ วงศ์ (สำหรับทุก ๆ ความช่วยเหลือขณะที่ทำวิจัยอยู่ที่อังกฤษ) พี่เฉลิมทรัพย์ สังขวิจิตร (สำหรับคำปรึกษาในทุก ๆ เรื่อง) พี่จระเดช พลสวัสดิ์ (สำหรับความช่วยเหลือในการใช้ server ในการทดลอง) โกเมศ อัมพวัน (สำหรับกำลังใจที่มีให้มาเสมอ ตั้งแต่เรียนปริญญาตรี โท เอก มาด้วยกัน) วีรสิทธิ์ เจริญศิริ (สำหรับกำลังใจและความช่วยเหลือที่คอยผลักดันให้ฝ่าฟัน อุปสรรคมาได้) ขอขอบคุณ พี่ ๆ เพื่อน ๆ น้อง ๆ ทุกคนที่ไม่สามารถเอ่ยชื่อในที่นี้ได้หมด สำหรับน้ำใจที่มีให้ตลอดมา

สุดท้ายนี้ ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และญาติพี่น้องทุกท่านที่คอยเป็น ห่วงเป็นใย ให้กำลังใจ และให้การสนับสนุนในทุกด้านตลอดมา ขอขอบคุณอาประภิน และ Malcolm Holmes สำหรับความช่วยเหลือทั้งคำปรึกษาด้านภาษาอังกฤษและกำลังใจที่มีให้ ตลอดเวลา

สารบัญ

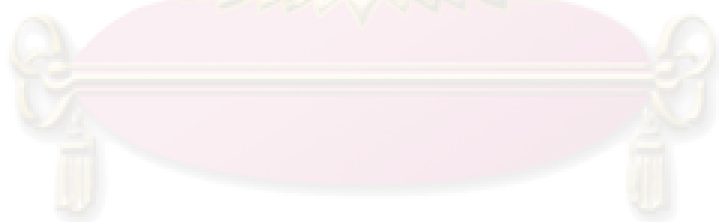
	หน้า
บทคัดย่อภาษาไทย	ง
บทคัดย่อภาษาอังกฤษ	จ
กิตติกรรมประกาศ	ฉ
สารบัญ	ช
สารบัญตาราง	ฌ
สารบัญภาพ	ญ
บทที่	
1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการวิจัย	2
1.3 ขอบเขตของการวิจัย	2
1.4 ขั้นตอนและวิธีดำเนินการวิจัย	3
1.5 ประโยชน์ที่คาดว่าจะได้รับการวิจัย	3
1.6 เนื้อหาในวิทยานิพนธ์นี้	4
1.7 งานตีพิมพ์	4
2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	7
2.1 ขั้นตอนวิธีเชิงพันธุกรรม	7
2.1.1 ขั้นตอนวิธีประมาณการแจกแจง	11
2.1.2 ปัญหาทดสอบ	14
2.2 ออปชันจริง	17
2.3 เงื่อนไขการหยุดของขั้นตอนวิธีเชิงพันธุกรรม	19
3 การใช้ออปชันจริงในการหาเวลาหยุดเหมาะสมสุดของขั้นตอนวิธีเชิงพันธุกรรม	25
3.1 การสร้างตัวแบบความไม่แน่นอนมูลฐาน	25
3.2 การสร้างฟังก์ชันสำหรับคำนวณมูลค่าที่ได้รับจากการทำงานของอัลกอริทึม เมื่อมีการเลือกหยุด	26
3.3 การคำนวณมูลค่าอัลกอริทึม	27
3.4 การสร้างขอบเขตการใช้สิทธิ	28
4 การวิเคราะห์เวลาหยุดเหมาะสมสุดของขั้นตอนวิธีประมาณการแจกแจง	32
4.1 การวิเคราะห์ขั้นตอนวิธีประมาณการแจกแจงโดยใช้จำนวนประชากร 2 ตัว	32
4.2 การวิเคราะห์ขั้นตอนวิธีประมาณการแจกแจงโดยใช้จำนวนประชากรที่มากขึ้น	37

บทที่	หน้า
5 การศึกษาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมแบบง่าย	45
6 การหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมแบบค่อยเป็นค่อยไป	55
6.1 การทดลองการหยุดตามขอบเขตการใช้สิทธิซึ่งมีการปรับปรุงแบบ ค่อยเป็นค่อยไปโดยไม่มีต้นทุนการคำนวณ	57
6.2 การทดลองการหยุดตามขอบเขตการใช้สิทธิซึ่งมีการปรับปรุงแบบ ค่อยเป็นค่อยไปโดยพิจารณาต้นทุนการคำนวณ	59
7 สรุปผลการวิจัย	65
รายการอ้างอิง	67
ภาคผนวก	71
ภาคผนวก ก	72
ประวัติผู้เขียนวิทยานิพนธ์	82

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญตาราง

ตารางที่	หน้า
2.1 การหาค่าความเหมาะสมของประชากร.....	9
4.1 มูลค่าอัลกอริทึมและค่าตรวจเช็คความยากของขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับและวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากร.....	36
4.2 มูลค่าอัลกอริทึมและค่าตรวจเช็คความยากของขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากร และขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปร...	43
4.3 มูลค่าอัลกอริทึมเมื่อพิจารณาอัตราลดค่า	44
5.1 ผลการทดลองเปรียบเทียบระหว่างขั้นตอนวิธีเชิงพันธุกรรมแบบง่าย และขั้นตอนวิธีเชิงพันธุกรรมแบบง่ายที่ใช้เงื่อนไขการหยุดเหมาะสมที่สุด	46
5.2 จำนวนครั้งการหยุดทำงานตามเงื่อนไขเส้นแบ่งด้านบนและด้านล่าง.....	54
6.1 ตารางเปรียบเทียบผลการหยุดโดยใช้ขอบเขตการใช้สิทธิที่มีการปรับปรุงแบบค่อยเป็นค่อยไปและการหยุดแบบปกติของขั้นตอนวิธีเชิงพันธุกรรมแบบง่าย	57
6.2 การทดสอบความแตกต่างของค่าความเหมาะสมเฉลี่ยของทั้ง 2 วิธี	58
6.3 การหยุดโดยใช้ขอบเขตการใช้สิทธิที่มีต้นทุนการคำนวณ 0.01.....	60
6.4 การหยุดโดยใช้ขอบเขตการใช้สิทธิที่มีต้นทุนการคำนวณ 0.05.....	61
6.5 จำนวนครั้งการหยุดในแต่ละรุ่นของการใช้เงื่อนไขการหยุดแบบค่อยเป็นค่อยไป	62
6.6 ค่าเฉลี่ยและค่าความแปรปรวนของจำนวนรุ่นที่อัลกอริทึมตัดสินใจหยุดทำงาน	63



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญญภาพ

รูปที่	หน้า
2.1 รหัสเทียมของขั้นตอนวิธีเชิงพันธุกรรม	8
2.2 ตัวอย่างการไขว้เปลี่ยน	10
2.3 ตัวอย่างการกลายพันธุ์	11
2.4 ขั้นตอนวิธีของการเรียนรู้แบบอาศัยประชากร	12
2.5 ขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปร	13
2.6 ขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับ	14
2.7 ตัวอย่างการคำนวณมูลค่าอุปชันจากต้นไม้ทวินาม	18
3.1 ตัวอย่างเมทริกซ์ค่าความน่าจะเป็นในการเปลี่ยนแปลงค่าความเหมาะสม	25
3.2 ตัวอย่างการคำนวณมูลค่าอัลกอริทึม	28
3.3 ตัวอย่างผลการคำนวณมูลค่าอัลกอริทึมและผลการตัดสินใจ	29
3.4 ตัวอย่างรูปขอบเขตการใช้สิทธิ	29
3.5 ขั้นตอนวิธีในการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรม	30
3.6 ขั้นตอนวิธีในการคำนวณมูลค่าอัลกอริทึม	31
4.1 ขอบเขตการใช้สิทธิจากการวิเคราะห์ด้วยประชากร 2 ตัว	34
4.2 ขอบเขตการใช้สิทธิจากการวิเคราะห์ด้วยประชากรที่มากขึ้น สำหรับปัญหาจำนวนหนึ่งมากที่สุด	38
4.3 ขอบเขตการใช้สิทธิจากการวิเคราะห์ด้วยประชากรที่มากขึ้น สำหรับปัญหากับดักขนาด 3	39
4.4 ขอบเขตการใช้สิทธิจากการวิเคราะห์ด้วยประชากรที่มากขึ้น สำหรับปัญหากับดักขนาด 5	40
4.5 ขอบเขตการใช้สิทธิจากการวิเคราะห์ด้วยประชากรที่มากขึ้น สำหรับปัญหากับดักแบบลำดับชั้น	41
4.6 ขอบเขตการใช้สิทธิจากการวิเคราะห์ด้วยประชากรที่มากขึ้น สำหรับปัญหาลำดับชั้นแบบก็ต่อเมื่อ	42
5.1 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของปัญหาจำนวนหนึ่งมากที่สุด ขนาด 30, 50 และ 100 บิต	47
5.2 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของปัญหากับดักขนาด 3 ต่อกันจำนวน 10, 20 และ 30 ชุด	48
5.3 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของปัญหากับดักขนาด 5 ต่อกันจำนวน 10, 20 และ 30 ชุด	49

5.4 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของปัญหาที่ติดแบบลำดับชั้น ขนาด 27, 81 และ 243 บิต.....	50
5.5 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของปัญหาลำดับชั้นแบบก็ต่อเมื่อ ขนาด 32, 64 และ 128 บิต.....	51
5.6 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของปัญหา 2 ขั้วแบบ 6 บิต ขนาด 30, 60 และ 90 บิต.....	52
6.1 ขั้นตอนวิธีของการหาเวลาหยุดเหมาะสมสุดแบบค่อยเป็นค่อยไป.....	55
6.2 ตัวอย่างขอบเขตการใช้สิทธิจากปัญหาจำนวนหนึ่งมากที่สุดขนาด 30 บิต.....	56
6.3 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาจำนวนหนึ่งมากที่สุดขนาด 30 บิต.....	64
ก.1 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาจำนวนหนึ่งมากที่สุดขนาด 30 บิต.....	72
ก.2 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาจำนวนหนึ่งมากที่สุดขนาด 50 บิต.....	73
ก.3 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาจำนวนหนึ่งมากที่สุดขนาด 100 บิต.....	73
ก.4 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับติดขนาด 3 ต่อกัน 10 ชุด.....	74
ก.5 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับติดขนาด 3 ต่อกัน 20 ชุด.....	74
ก.6 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับติดขนาด 3 ต่อกัน 30 ชุด.....	75
ก.7 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับติดขนาด 5 ต่อกัน 10 ชุด.....	75
ก.8 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับติดขนาด 5 ต่อกัน 20 ชุด.....	76
ก.9 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับติดขนาด 5 ต่อกัน 30 ชุด.....	76
ก.10 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับติดแบบลำดับชั้นขนาด 27 บิต.....	77
ก.11 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับติดแบบลำดับชั้นขนาด 81 บิต.....	77
ก.12 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับติดแบบลำดับชั้นขนาด 243 บิต.....	78
ก.13 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาลำดับชั้นแบบก็ต่อเมื่อขนาด 32 บิต.....	78
ก.14 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาลำดับชั้นแบบก็ต่อเมื่อขนาด 64 บิต.....	79
ก.15 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาลำดับชั้นแบบก็ต่อเมื่อขนาด 128 บิต.....	79
ก.16 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหา 2 ขั้วแบบ 6 บิต ขนาด 30 บิต.....	80
ก.17 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหา 2 ขั้วแบบ 6 บิต ขนาด 60 บิต.....	80
ก.18 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหา 2 ขั้วแบบ 6 บิต ขนาด 90 บิต.....	81

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

คำถามที่ว่า “เวลาหยุดที่เหมาะสมที่สุดคือเมื่อใด” เป็นคำถามทั่วไปที่พบบ่อยในหลายสถานการณ์ เมื่อต้องตัดสินใจว่าสมควรหยุดหรือดำเนินการใด ๆ ต่อ ผลการตัดสินใจดูเหมือนจะง่ายเพราะเพียงแค่อธิบายว่าจะหยุดหรือทำต่อ แต่ในความเป็นจริงการตัดสินใจนี้กระทำได้ยาก เนื่องจากมีหลายปัจจัยเข้ามาเกี่ยวข้องด้วย ปัจจัยสำคัญที่ทำให้การตัดสินใจนี้ยากคือความไม่แน่นอนในอนาคต ซึ่งทำให้เกิดความเสี่ยงในการตัดสินใจตามมา

ปัญหาการตัดสินใจในลักษณะนี้เกี่ยวข้องกับถึงการตัดสินใจลงทุน ผู้ลงทุนต้องพิจารณาความเสี่ยงที่จะเกิดขึ้น แล้วตัดสินใจว่าจะลงทุนเมื่อใด หรือการตัดสินใจปิดกิจการเจ้าของกิจการก็ต้องพิจารณาว่าควรเลิกเมื่อใดจึงจะคุ้มที่สุด เพราะถ้าดำเนินกิจการต่อก็ต้องมีต้นทุนในการดำเนินการ เมื่อเทียบกับกำไรที่อาจจะได้แล้วอะไรคือการตัดสินใจที่ดีกว่า เรายังพบปัญหาการตัดสินใจแบบนี้ได้ในหลาย ๆ สถานการณ์ในชีวิตประจำวันด้วย เช่น ผู้ที่ทำงานอยู่ใจกลางเมือง มักจะเผชิญกับปัญหาการติด โดยเฉพาอย่างยิ่งในช่วงเวลาเลิกงาน การตัดสินใจกลับบ้านในช่วงเวลาใดจึงจะดีที่สุดก็เป็นคำถามของหลาย ๆ คน เพราะถ้ากลับเวลานี้ก็ต้องเผชิญกับปัญหาการติด ในขณะที่ถ้าอยู่ทำงานต่อก็จะได้นานมากขึ้น และอาจใช้เวลาเดินทางน้อยกว่า

ปัญหาการตัดสินใจแบบนี้ไม่ได้มีอยู่แต่ในแง่ของการดำเนินชีวิตจริง แต่เราพบการตัดสินใจลักษณะนี้ในขั้นตอนวิธีทางคอมพิวเตอร์ด้วย ต่างกันตรงที่ว่าคอมพิวเตอร์ไม่สามารถตัดสินใจได้ด้วยวิจรณ์ญาณของตัวเอง มนุษย์ต้องเป็นผู้กำหนดเงื่อนไขในการตัดสินใจให้ปัญหาหนึ่งซึ่งเป็นที่มาของวิทยานิพนธ์นี้ คือ การตัดสินใจหยุดการทำงานของอัลกอริทึม

ในอัลกอริทึมทั่วไป เช่น การหาค่ามากสุดในแถวลำดับ การตัดสินใจหยุดอาจไม่ใช่ปัญหา เนื่องจากเมื่อหาครบทุกค่าในแถวลำดับนั้นแล้ว ก็จะได้คำตอบแน่นอนและการทำงานก็จะสิ้นสุดลง แต่ในอัลกอริทึมบางประเภท การทำงานไม่ได้เป็นเช่นนั้น ตัวอย่างเช่นอัลกอริทึมในการเรียนรู้ เราไม่รู้ว่าเรียนถึงเมื่อใดจึงจะพอ การกำหนดเงื่อนไขการหยุดจึงมักจะกำหนดจำนวนรอบในการเรียนรู้เอาไว้ แต่การกำหนดค่านี้ก็ขึ้นอยู่กับแต่ละปัญหา คุณภาพคำตอบและเวลาที่ใช้ก็แตกต่างกันไป

ขั้นตอนวิธีเชิงพันธุกรรม (Genetic algorithm) เป็นอัลกอริทึมหนึ่งซึ่งมีการทำงานในลักษณะดังกล่าว การปรับปรุงคุณภาพคำตอบจะอาศัยการเลือกสรรคำตอบดี ๆ มาสร้าง

คำตอบใหม่ขึ้นมา แล้วตรวจสอบว่าคำตอบนั้นมีความดีอยู่ในระดับใด เราไม่รู้คำตอบที่ดีที่สุดนั้นหน้าตาเป็นเช่นไร แต่พยายามที่จะปรับปรุงคำตอบไปตามคำตอบดี ๆ ที่เรามีความรู้อยู่ แล้วหวังว่าจะพบคำตอบที่ดีขึ้น กระบวนการทำงานจะอาศัยการลองผิดลองถูก ซึ่งในการทดลองคำตอบแต่ละครั้งก็ต้องใช้ต้นทุนการคำนวณที่มากน้อยต่างกันไปในแต่ละปัญหา การทดลองมากขึ้นอาจพาไปสู่คำตอบที่ดีขึ้น แต่ก็ต้องเสียทรัพยากรที่ใช้ในการทดลองแต่ละครั้งไปเรื่อย ๆ คำถามคือควรจะลองผิดลองถูกไปเท่าใดจึงควรเลิก

การหาเวลาหยุดเหมาะสมที่สุดนั้นได้มีการศึกษาในทางการวิจัยดำเนินการ (Operations research) มานานแล้ว การตัดสินใจใช้สิทธิจากอปชันเป็นตัวอย่างของการประยุกต์ใช้เทคนิคในแนวปัญหานี้ การตัดสินใจใช้สิทธิดังกล่าวเปรียบได้กับการที่อัลกอริทึมตัดสินใจหยุด อาศัยความคล้ายกันของปัญหานี้ วิทยานิพนธ์นี้จึงนำเสนอการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมโดยอาศัยวิธีอปชันจริง และผลลัพธ์ที่ได้ถูกนำมาใช้วิเคราะห์ความเหมาะสมของขั้นตอนวิธีพันธุกรรมแบบต่าง ๆ ในแง่ของความคุ้มค่า ซึ่งพิจารณาถึงต้นทุนการคำนวณ เวลา และความเป็นไปได้ที่จะพบคำตอบที่ดีขึ้น

1.2 วัตถุประสงค์ของการวิจัย

งานวิจัยนี้มีวัตถุประสงค์เพื่อนำเสนอการใช้วิธีอปชันจริงในการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรม

1.3 ขอบเขตของการวิจัย

งานวิจัยนี้นำเสนอการใช้วิธีอปชันจริงในการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมในลักษณะต่าง ๆ ดังนี้

1. การใช้วิธีอปชันจริงในการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรม
2. นำเสนอการตัดสินใจหยุดโดยใช้เงื่อนไขที่ได้จากวิธีอปชันจริง
3. วิเคราะห์เวลาหยุดของขั้นตอนวิธีประมาณการแจกแจงแบบที่ตัวแปรไม่ขึ้นต่อกัน (Univariate estimation of distribution algorithms) ซึ่งเป็นขั้นตอนวิธีเชิงพันธุกรรมแนวใหม่
4. เปรียบเทียบการหยุดโดยใช้วิธีการที่นำเสนอกับเงื่อนไขการหยุดแบบทั่วไป
5. นำเสนอการใช้วิธีอปชันจริงแบบค่อยเป็นค่อยไป (Incremental on-going improvement) ในขั้นตอนวิธีเชิงพันธุกรรมแบบง่าย

6. วิเคราะห์พฤติกรรมการหยุดของขั้นตอนวิธีเชิงพันธุกรรมแบบง่ายในปัญหาทดสอบต่างๆ รวมทั้งแสดงให้เห็นการเปลี่ยนแปลงพฤติกรรมการหยุดเมื่อมีต้นทุนการคำนวณที่สูงขึ้น

1.4 ขั้นตอนและวิธีดำเนินการวิจัย

1. ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์นี้ ทั้งในส่วนของขั้นตอนวิธีเชิงพันธุกรรมและวิธีอุปชันจริง
2. ออกแบบและพัฒนาวิธีการใช้วิธีอุปชันจริงในการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรม
3. ทดลองและปรับปรุงวิธีการที่นำเสนอ
4. วิเคราะห์ผลลัพธ์ที่ได้จากการหยุดโดยใช้วิธีอุปชันจริง
5. เขียนรายงานและจัดทำวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับการวิจัย

ได้วิธีการในการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมโดยใช้วิธีอุปชันจริง ซึ่งมีเงื่อนไขการหยุดเป็นแนวทางให้อัลกอริทึมตัดสินใจ โดยในวิทยานิพนธ์ได้นำเสนอสิ่งใหม่ดังนี้

การนำวิธีอุปชันจริงมาใช้ในการหาเวลาหยุดของอัลกอริทึมเป็นสิ่งใหม่ของวิทยานิพนธ์นี้ นำเสนอ การมองว่าการตัดสินใจลงทุนกับการตัดสินใจหยุดของอัลกอริทึมสามารถนำหลักการเดียวกันมาใช้ตัดสินใจเป็นแนวคิดที่ไม่เคยมีใครทำมาก่อน

เงื่อนไขการหยุดที่ได้จากวิธีอุปชันจริง สามารถบอกขอบเขตของค่าความเหมาะสมในแต่ละรุ่น เพื่อใช้ตัดสินใจว่าควรหยุดหรือทำงานต่อ แนวทางการตัดสินใจในลักษณะนี้เป็นเงื่อนไขการหยุดแบบใหม่ของขั้นตอนวิธีเชิงพันธุกรรม ซึ่งจากผลการทดลองแสดงให้เห็นว่าเมื่ออัลกอริทึมทำงานภายใต้การตัดสินใจหยุดเหมาะสมที่สุดสามารถลดจำนวนครั้งการคำนวณลงได้ โดยที่อัตราส่วนระหว่างคุณภาพคำตอบต่อทรัพยากรที่ใช้ในการคำนวณเพิ่มขึ้น

วิทยานิพนธ์นี้ยังได้เสนอการใช้มูลค่าอัลกอริทึมเป็นตัววัดความเหมาะสมของอัลกอริทึมในการแก้ปัญหาต่าง ๆ ค่านี้จะใช้เปรียบเทียบอัลกอริทึมต่าง ๆ ว่าอัลกอริทึมใดเหมาะสมที่จะนำมาใช้แก้ปัญหานั้น ๆ มากกว่ากัน โดยพิจารณาในแง่ของความคุ้มค่า โอกาส และเวลาที่ใช้ในการหาคำตอบ ดรรชนีความยาก (Difficulty index) ได้ถูกเสนอขึ้นด้วยเพื่อใช้เปรียบเทียบความยากในการแก้แต่ละปัญหาด้วยอัลกอริทึมหนึ่ง ๆ

1.6 เนื้อหาในวิทยานิพนธ์นี้

รายละเอียดต่าง ๆ ในวิทยานิพนธ์จะนำเสนอเป็นลำดับดังนี้ ในบทที่ 2 จะกล่าวถึง ทฤษฎีและงานวิจัยที่เกี่ยวข้อง ได้แก่ ขั้นตอนวิธีเชิงพันธุกรรม วิธีอุปชันจริง และเงื่อนไขการหยุดของขั้นตอนวิธีเชิงพันธุกรรม บทที่ 3 จะเสนอวิธีการนำวิธีอุปชันจริงมาใช้ในการหาเวลาหยุดที่เหมาะสมของขั้นตอนวิธีเชิงพันธุกรรม แนวคิดและวิธีการที่นำเสนอในบทนี้จะถูกนำไปใช้ทดสอบและวิเคราะห์ขั้นตอนวิธีพันธุกรรมแบบต่าง ๆ กับปัญหาทดสอบซึ่งจะนำเสนอในบทต่อไป

วิทยานิพนธ์นี้ได้แสดงการนำวิธีอุปชันจริงมาใช้เวลาหยุดที่เหมาะสมของขั้นตอนวิธีเชิงพันธุกรรมในลักษณะต่าง ๆ ในบทที่ 4 ถึง 6 ดังนี้

บทที่ 4 ได้ทำการวิเคราะห์เวลาหยุดที่เหมาะสมของขั้นตอนวิธีประมาณการแจกแจง ซึ่งเป็นแนวทางวิจัยใหม่ของขั้นตอนวิธีเชิงพันธุกรรม การทดลองจะเปรียบเทียบอัลกอริทึมต่าง ๆ ของขั้นตอนวิธีประมาณการแจกแจงแบบง่ายซึ่งตัวแปรไม่ขึ้นต่อกันกับปัญหาทดสอบต่าง ๆ แล้ววิเคราะห์พฤติกรรมการหยุดรวมทั้งมูลค่าของแต่ละอัลกอริทึม

บทที่ 5 แสดงเวลาหยุดที่เหมาะสมของขั้นตอนวิธีเชิงพันธุกรรมแบบง่ายที่ใช้เงื่อนไขการหยุดตามวิธีอุปชันจริง เปรียบเทียบกับวิธีการหยุดแบบปกติ ในการทดลองจะใช้ข้อมูลเดียวกันมาเปรียบเทียบว่าถ้าใช้เงื่อนไขการหยุดตามวิธีอุปชันจริงจะสามารถลดจำนวนครั้งการคำนวณลงได้จริงหรือไม่

อย่างไรก็ตาม การหาเวลาหยุดที่เหมาะสมด้วยวิธีที่นำเสนออาจจะไม่เหมาะกับการนำไปใช้ในทางปฏิบัติเนื่องจากในความเป็นจริงเราไม่สามารถทำงานจนเสร็จก่อนแล้วค่อยมาหาวิธีการตัดสินใจ ในบทที่ 6 นี้จึงนำเสนอการหาเวลาหยุดแบบค่อยเป็นค่อยไป ซึ่งจะนำความรู้จากการทำงานในรอบก่อนหน้ามาเป็นเงื่อนไขการหยุดในรอบถัดไป โดยเงื่อนไขการหยุดนี้จะถูกปรับปรุงทุกครั้งเมื่อมีการทำงานในรอบใหม่ ในการทดลองยังนำเสนอการหยุดโดยพิจารณาต้นทุนการคำนวณด้วย เมื่ออัลกอริทึมมีต้นทุนการคำนวณที่สูงขึ้น การตัดสินใจหยุดของอัลกอริทึมก็จะมีเปลี่ยนแปลงไปด้วย

บทที่ 7 สรุปผลการวิจัย และจะกล่าวถึงมุมมองและแนวคิดของการนำวิธีอุปชันจริงมาใช้ในการหาเวลาหยุดที่เหมาะสมของขั้นตอนวิธีเชิงพันธุกรรม สรุปผลการทดลองและแนวทางที่น่าจะศึกษาวิจัยต่อไป

1.7 งานตีพิมพ์

ในระหว่างการศึกษาได้มีการตีพิมพ์ผลงานวิจัยดังนี้

Journal

- Sunisa Rimcharoen, Daricha Sutivong, and Prabhas Chongstitvatana, "Real Options Approach to Evaluating Genetic Algorithms". *Applied Soft Computing*, 2009 (doi:10.1016/j.asoc.2008.11.002).

Proceedings

- Sunisa Rimcharoen, Daricha Sutivong, and Prabhas Chongstitvatana, "Optimal Stopping Time of Compact Genetic Algorithm on Deceptive Problem Using Real Options Analysis". In *Proceedings of the 2007 IEEE Congress on Evolutionary Computation (CEC 2007)*, Singapore, September 25-28, 2007.
- Sunisa Rimcharoen, Daricha Sutivong, and Prabhas Chongstitvatana, "A Synthesis of Optimal Stopping Time in Compact Genetic Algorithm Based on Real Options Approach". In *Proceedings of Genetic and Evolutionary Computation Conference (GECCO 2007)*, London, England, July 7-11, 2007.
- Sunisa Rimcharoen, Daricha Sutivong, and Prabhas Chongstitvatana, "Real Options Approach to Finding Optimal Stopping Time in Compact Genetic Algorithm". In *Proceedings of the 2006 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2006)*, Taipei, Taiwan, October 8-11, 2006.
- Sunisa Rimcharoen, Daricha Sutivong, and Prabhas Chongstitvatana, "Updating Strategy in Compact Genetic Algorithm Using Moving Average Approach". In *Proceedings of the 2006 IEEE International Conferences on Cybernetics & Intelligent Systems (CIS) and Robotics, Automation & Mechatronics (RAM) (CIS-RAM 2006)*, Bangkok, Thailand, June 7-9, 2006.
- Sunisa Rimcharoen, Daricha Sutivong, and Prabhas Chongstitvatana, "Prediction of the Stock Exchange of Thailand Using Adaptive Evolution

Strategies". In Proceedings of 17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2005), Hong Kong, China, November 14-16, 2005.



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎีและงานวิจัยที่เกี่ยวข้อง ได้แก่ ขั้นตอนวิธีเชิงพันธุกรรม (Genetic algorithm) วิธีออปชันจริง (Real options) และรายละเอียดเกี่ยวกับเงื่อนไขการหยุดของขั้นตอนวิธีเชิงพันธุกรรม

2.1 ขั้นตอนวิธีเชิงพันธุกรรม

ในส่วนของขั้นตอนวิธีเชิงพันธุกรรม เนื้อหาในส่วนนี้จะอธิบายตั้งแต่หลักการเบื้องต้นรวมถึงรายละเอียดการทำงานของอัลกอริทึม จากนั้นจะอธิบายขั้นตอนวิธีประมาณการแจกแจง (Estimation of distribution algorithm) ซึ่งเป็นแนวคิดใหม่ของการใช้ตัวแบบของคำตอบ แทนการใช้กลุ่มประชากรของขั้นตอนวิธีเชิงพันธุกรรมแบบเดิม วิธีการนี้เป็นแนวทางใหม่ของการวิจัยในด้านการคำนวณเชิงวิวัฒนาการ (Evolutionary computation) และส่วนสุดท้ายของหัวข้อนี้จะกล่าวถึงปัญหาทดสอบที่ใช้ในการวัดการทำงานของขั้นตอนวิธีเชิงพันธุกรรมที่จะนำมาใช้ในวิทยานิพนธ์

ขั้นตอนวิธีเชิงพันธุกรรมเป็นขั้นตอนวิธีหาคำตอบที่เลียนแบบกระบวนการวิวัฒนาการตามธรรมชาติ โดย Darwin (1859) ได้พูดถึงทฤษฎีวิวัฒนาการไว้ว่า สิ่งมีชีวิตจะมีการถ่ายทอดลักษณะทางพันธุกรรมจากรุ่นสู่รุ่น เพื่อที่จะดำรงเผ่าพันธุ์อยู่ต่อไป สิ่งมีชีวิตที่สามารถปรับตัวได้เหมาะสมกับสภาพแวดล้อมก็จะอยู่รอดและสืบลูกหลาน ส่วนสิ่งมีชีวิตที่อ่อนแอก็จะล้มตายลง วิธีการคัดเลือกตามธรรมชาติที่สิ่งมีชีวิตที่เหมาะสมอยู่รอดและสืบเผ่าพันธุ์ต่อไป ทำให้เกิดกระบวนการวิวัฒนาการขึ้น สิ่งมีชีวิตต่าง ๆ ต้องพยายามปรับตัวเองให้เข้มแข็ง เพื่อที่จะผ่านการคัดสรรตามธรรมชาติ และเมื่อเวลาผ่านไป วิวัฒนาการไประยะหนึ่งก็จะได้สิ่งมีชีวิตที่ดีและเหมาะสม

จากหลักการวิวัฒนาการตามธรรมชาติข้างต้น ทำให้เกิดแรงบันดาลใจในการพัฒนาวิธีการคำนวณเชิงวิวัฒนาการขึ้นมา ในช่วงทศวรรษที่ 1960 ถึง 1970 มีนักวิจัยหลายกลุ่มได้เริ่มพัฒนาเทคนิคการคำนวณโดยอาศัยหลักการนี้ สามวิธีการหลัก ๆ ที่ถูกนำเสนอในยุคนั้นคือ การโปรแกรมเชิงวิวัฒนาการ (Evolutionary programming), กลยุทธ์เชิงวิวัฒนาการ (Evolution strategy) และขั้นตอนวิธีเชิงพันธุกรรม

การโปรแกรมเชิงวิวัฒนาการนำเสนอโดย Fogel et al. (1965, 1966) เพื่อใช้ในการวิวัฒนาการเครื่องจักรสถานะแบบจำกัด (Finite state machine) ในเวลาใกล้เคียงกันนักวิจัยที่

เยอรมันได้นำเสนอกลยุทธเชิงวิวัฒน์ (Rechenberg, 1965, 1971; Schwefel, 1974) เพื่อทำการหาค่าเหมาะสุดเชิงตัวเลข (Numerical optimization) ทั้งสองวิธีนี้มีความใกล้เคียงกันมาก และมีข้อเด่นในการประยุกต์ใช้กับข้อมูลที่เป็นจำนวนจริง แต่ในปัจจุบันขั้นตอนวิธีที่ถูกนำมาใช้อย่างแพร่หลายคือขั้นตอนวิธีเชิงพันธุกรรม ซึ่งนำเสนอโดย Holland (1973, 1975) และกลายเป็นที่รู้จักในวงกว้างจากหนังสือของ Goldberg (1989)

ผลเฉลยหรือคำตอบของปัญหาต่าง ๆ ในขั้นตอนวิธีเชิงพันธุกรรมจะถูกแทนด้วยรูปแบบโครโมโซม (Chromosome) แต่ละโครโมโซมถูกแทนด้วยสายอักขระของเลขฐานสอง ซึ่งในหนึ่งประชากร (Individual) อาจมีมากกว่า 1 โครโมโซมก็ได้ และหลาย ๆ ประชากรรวมกันจะกลายเป็นกลุ่มประชากร (Population) อัลกอริทึมจะอาศัยกลุ่มประชากรเหล่านี้ในการวิวัฒนาการไปสู่คำตอบที่ดีที่สุด

การทำงานของขั้นตอนวิธีเชิงพันธุกรรมจะเริ่มต้นจากการสุ่มประชากรขึ้นมาจำนวนหนึ่ง แล้วประเมินค่าความเหมาะสมของประชากรแต่ละตัว โดยการแปลงสายอักขระที่ถูกจัดเก็บไว้เพื่อหาค่าความเหมาะสม (Fitness value) ผ่านทางฟังก์ชันหาค่าความเหมาะสม (Fitness function) ซึ่งจะแตกต่างกันไปขึ้นกับแต่ละปัญหา ชุดของประชากรที่ทำการประเมินค่าแล้วจะถูกเลือกและสร้างประชากรรุ่นใหม่ขึ้น แต่ละครั้งที่สร้างประชากรรุ่นใหม่ขึ้นมาจะเรียกว่า “รุ่น” (Generation) กระบวนการวิวัฒนาการจะเลือกประชากรที่ดีในแต่ละรุ่นเพื่ออยู่รอดในรุ่นต่อไป และจะวนซ้ำอยู่อย่างนั้นจนกว่าจะถึงเงื่อนไขการหยุดที่กำหนดไว้ ขั้นตอนวิธีเชิงพันธุกรรมเขียนเป็นรหัสเทียมได้ดังรูปที่ 2.1

```

Procedure genetic algorithm
1: Initialize population
2: Evaluate population
3: while termination criterion not fulfilled do
4: begin
5:     Selection
6:     Crossover
7:     Mutation
8:     Evaluate
9: end

```

รูปที่ 2.1 รหัสเทียมของขั้นตอนวิธีเชิงพันธุกรรม

รายละเอียดของขั้นตอนวิธีเชิงพันธุกรรมในรูปที่ 2.1 ประกอบด้วย การสร้างกลุ่มประชากรเริ่มต้น การประเมินค่าคำตอบ การสร้างกลุ่มประชากรรุ่นใหม่ (ขั้นที่ 5 – 7) ซึ่งมีรายละเอียดดังนี้

1. การสร้างกลุ่มประชากรเริ่มต้น

ขั้นตอนวิธีเชิงพันธุกรรมจะกำหนดประชากรเริ่มต้นโดยการสุ่ม ประชากรแต่ละตัวจะถูกแสดงด้วยสายอักขระฐานสองที่มีความยาวคงที่ และจำนวนประชากรก็กำหนดเป็นค่าคงที่เช่นกัน ในที่นี้จำนวนประชากรจะเท่ากันทุกรุ่น

2. การประเมินค่าคำตอบ

ขั้นตอนนี้เป็นการวัดค่าความเหมาะสมของประชากรแต่ละตัว โดยใช้ฟังก์ชันหาค่าความเหมาะสม ค่าที่ได้จากฟังก์ชันนี้เรียกว่าค่าความเหมาะสม ซึ่งเป็นตัวบอกว่าประชากรตัวใดเหมาะสมหรือดีกว่ากัน ค่านี้จะถูกนำไปใช้ในการคัดเลือกประชากรสำหรับรุ่นต่อไป

การประเมินค่าความเหมาะสมจะทำได้โดยการแปลงค่าของประชากรที่จัดเก็บอยู่ในรูปแบบของโครโมโซมซึ่งเป็นเลขฐานสอง ให้เป็นค่าความดีของประชากรตัวนั้น ๆ เช่น ปัญหาการหาค่าสูงสุดของฟังก์ชัน $f(x) = x^2$ เมื่อ x มีค่าตั้งแต่ 0 ถึง 31 สามารถแทนค่า x เป็นสายอักขระฐานสองที่มีความยาวเท่ากับ 5 และสามารถคำนวณค่าความเหมาะสมได้ดังตารางที่ 2.1

ตารางที่ 2.1 การหาค่าความเหมาะสมของประชากร

ประชากร	ค่า x	ค่าความเหมาะสม
0 0 1 0 0	4	16
1 0 0 1 1	19	361
0 1 0 0 1	9	81
1 1 1 0 0	28	784
1 1 1 1 1	31	961

3. การสร้างกลุ่มประชากรรุ่นใหม่

การสร้างกลุ่มประชากรใหม่นั้นเกิดจากการเลือกประชากรรุ่นก่อนหน้ามาสร้างเป็นประชากรใหม่ โดยอาศัยกระบวนการสืบพันธุ์ (Reproduction) การกลายพันธุ์ (Mutation) และการไขว้เปลี่ยน (Crossover) ทำให้ประชากรรุ่นใหม่มีลักษณะบางประการของประชากรรุ่นก่อนติดมาด้วย การคัดเลือกประชากรเพื่อนำไปสร้างกลุ่มประชากรรุ่นถัดไปสามารถทำได้หลายวิธี โดยปกติจะใช้วิธีคัดเลือกตามสัดส่วนของค่าความเหมาะสม (Fitness proportional selection) ซึ่งวิธีการนี้ประชากรที่มีค่าความเหมาะสมสูงกว่า จะมีโอกาสถูกเลือกได้มากกว่าประชากรที่มีค่าความเหมาะสมต่ำ หรือใช้วิธีการคัดเลือกโดยการแข่งขัน (Tournament selection) ที่นำประชากรมาเปรียบเทียบกันทีละคู่ ตัวใดชนะก็จะอยู่แข่งขันต่อ ประชากรตัวสุดท้ายที่อยู่รอดจนครบขนาดของการแข่งขัน (Tournament size) ก็จะถูกเลือกไปสร้างประชากรใหม่ในรุ่นถัดไป ด้วยวิธีการดังต่อไปนี้

3.1 การสืบทอดพันธุ

วิธีการนี้เป็นการสร้างประชากรใหม่ จากประชากรต้นแบบ 1 ตัว โดยประชากรใหม่ที่ได้จะมีลักษณะเหมือนประชากรต้นแบบทุกประการ มักใช้กับประชากรต้นแบบที่มีค่าความเหมาะสมสูงที่สุดในรุ่นนั้น ๆ เพื่อที่จะได้ถ่ายทอดลักษณะที่ดีที่สุดไว้ในรุ่นถัดไป

3.2 การไขว้เปลี่ยน

วิธีการนี้เป็นการสร้างประชากรใหม่จากประชากรต้นแบบ 2 ตัว โดยการแลกเปลี่ยนคุณลักษณะบางประการของประชากรต้นแบบทั้ง 2 ตัว การไขว้เปลี่ยนแบบง่ายจะเริ่มด้วยการสุ่มตำแหน่งในสายอักขระ ที่ตำแหน่งที่สุ่มได้จะตัดแบ่งสายอักขระทั้งคู่ที่ตำแหน่งนั้น และแลกเปลี่ยนชิ้นส่วนที่ตัดออกสลับกัน ทำให้ได้ประชากรใหม่ 2 ตัว ที่มีคุณลักษณะร่วมกันระหว่างประชากรต้นแบบทั้งสองตัว การไขว้เปลี่ยนแสดงในรูปที่ 2.2

ก่อนทำการไขว้เปลี่ยน

1	1	0	0	1
0	1	1	0	1

หลังทำการไขว้เปลี่ยน

1	1	1	0	1
0	1	0	0	1

รูปที่ 2.2 ตัวอย่างการไขว้เปลี่ยน

3.3 การกลายพันธุ

วิธีการนี้เป็นการสร้างประชากรใหม่ขึ้นจากประชากรต้นแบบ 1 ตัว โดยจะทำการเปลี่ยนลักษณะบางประการของประชากรต้นแบบ กล่าวคือ ทำการเปลี่ยนค่าในสายอักขระจากเดิมเป็นค่าใหม่ด้วยการสุ่มตำแหน่งในสายอักขระ แล้วเปลี่ยนค่าตรงตำแหน่งนั้น เช่น ถ้าตำแหน่งเดิมเป็นค่า 0 ก็เปลี่ยนเป็น 1 แต่ถ้าเป็น 1 อยู่ ก็เปลี่ยนเป็น 0 ตัวอย่างการกลายพันธุแสดงในรูปที่ 2.3

ก่อนทำการกลายพันธุ์

0	1	1	0	1
---	---	---	---	---

หลังทำการกลายพันธุ์

0	1	0	0	1
---	---	---	---	---

รูปที่ 2.3 ตัวอย่างการกลายพันธุ์

หลังจากที่สร้างประชากรรุ่นใหม่เสร็จแล้ว ก็จะทำการศึกษาความเหมาะสมของประชากรใหม่ โดยทั่วไปประชากรรุ่นเก่าจะถูกแทนที่ด้วยประชากรใหม่ทั้งหมด แต่ในกรณีที่ต้องการรักษาประชากรส่วนใหญ่เอาไว้ แล้วแทนที่เฉพาะประชากรตัวที่มีคุณภาพต่ำด้วยประชากรใหม่ จะเรียกวิธีการนี้ว่าขั้นตอนวิธีเชิงพันธุกรรมแบบสภาวะมั่นคง (Steady state genetic algorithm)

2.1.1 ขั้นตอนวิธีประมาณการแจกแจง

แนวคิดของขั้นตอนวิธีเชิงพันธุกรรมได้มีการพัฒนาเรื่อยมา จนปัจจุบันนี้ได้มีการเสนอมุมมองใหม่ในการแทนค่าตอบด้วยการประมาณการแจกแจง ซึ่งแทนที่จะแทนค่าตอบด้วยกลุ่มประชากร ก็จะแทนค่าตอบด้วยตัวแบบความน่าจะเป็นแทน ขั้นตอนวิธีประมาณการแจกแจงนี้ จะไม่มีกลุ่มประชากร ไม่มีการดำเนินการทางพันธุกรรม เช่น การไขว้เปลี่ยน การกลายพันธุ์ อีกต่อไป แต่จะใช้วิธีการสร้างและปรับปรุงตัวแบบของคำตอบจากตัวอย่างดี ๆ ที่ได้สุ่มเลือกขึ้นมา

ขั้นตอนวิธีประมาณการแจกแจง (Estimation of distribution algorithms: EDAs) หรือ ขั้นตอนวิธีเชิงพันธุกรรมแบบการสร้างแบบจำลองความน่าจะเป็น (Probabilistic model building genetic algorithms: PMBGAs) นำเสนอโดย Mühlenbein และ Paaß (1996), Pelikan, Goldberg และ Lobo (1999) ในขั้นตอนวิธีนี้แต่ละตัวแปร (อาจมองเทียบได้กับแต่ละบิตในขั้นตอนวิธีเชิงพันธุกรรมแบบเดิม) จะมีค่าความน่าจะเป็นในการเป็นคำตอบ รวมทั้งความสัมพันธ์กับตัวแปรอื่น และจะใช้การปรับค่าความน่าจะเป็นเพื่อปรับปรุงการกระจายตัวของคำตอบให้ไปในทิศทางที่ผลเฉลยจะมีค่าความเหมาะสมสูงขึ้น

ขั้นตอนวิธีประมาณการแจกแจงสามารถแบ่งได้เป็น 3 กลุ่มหลัก ๆ ตามลักษณะการขึ้นต่อกันของตัวแปร คือ ขั้นตอนวิธีประมาณการแจกแจงแบบที่ตัวแปรไม่ขึ้นต่อกัน (Univariate estimation of distribution algorithm) ขั้นตอนวิธีประมาณการแจกแจงแบบที่ตัวแปรขึ้นต่อกันเป็นคู่ (Bivariate estimation of distribution algorithm) และขั้นตอนวิธีประมาณการแจกแจงแบบตัวแปรหลายตัวขึ้นต่อกัน (Multivariate estimation of distribution algorithm) ในที่นี้จะ

พูดถึงเฉพาะขั้นตอนวิธีประมาณการแจกแจงแบบง่ายที่สุดคือไม่มีความขึ้นต่อกันในตัวแปรแต่ละตัว

ขั้นตอนวิธีประมาณการแจกแจงแบบที่ตัวแปรไม่ขึ้นต่อกันมี 3 อัลกอริทึม คือ การเรียนรู้เพิ่มขึ้นแบบอาศัยประชากร (Population based incremental learning: PBIL) ขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปร (Univariate marginal distribution algorithm: UMDA) และขั้นตอนวิธีเชิงพันธุกรรมแบบกะชับ (Compact genetic algorithm: cGA) ซึ่งแต่ละวิธีมีรายละเอียดดังนี้

(1) การเรียนรู้เพิ่มขึ้นแบบอาศัยประชากร

อัลกอริทึมนี้นำเสนอโดย Baluja (1994) ซึ่งมีแนวคิดในการใช้เวกเตอร์ความน่าจะเป็น (Probability vector) แทนที่การใช้กลุ่มประชากรแบบเดิม เวกเตอร์ความน่าจะเป็นนี้จะเป็นตัวแบบในการหาการกระจายตัวของคำตอบดี ๆ โดยแต่ละมิติ (Dimension) ของเวกเตอร์เป็นค่าความน่าจะเป็นที่แต่ละบิตจะเป็น 1 การปรับปรุงค่าความน่าจะเป็นนี้ทำได้โดยการสุ่มสร้างตัวอย่างแล้วปรับค่าความน่าจะเป็นให้เข้าใกล้ตัวอย่างดี ๆ ขั้นตอนการทำงานของ การเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรแสดงในรูปที่ 2.4 เมื่อ l เป็นจำนวนมิติ (จำนวนบิต) p_i คือค่าความน่าจะเป็นในแต่ละมิติของเวกเตอร์ α คือ อัตราการเรียนรู้ซึ่งมีค่าอยู่ระหว่าง $(0, 1]$ และ x_k คือ ค่าในแต่ละบิต (ค่า 1 หรือ 0) ของตัวอย่างตัวที่ k

Procedure population-based incremental learning

- 1: Initialize probability vector (p) with 0.5 at each position.
- 2: Generate M individuals from the vector.
- 3: Select N best individuals, where $N \leq M$.
- 4: Update the probability vector p .

for $i = 1$ to l do

$$p_i = (1 - \alpha)p_i + \alpha \frac{1}{N} \sum_{k=1}^N x_k$$

- 5: Go to step 2 until a termination criterion is met.

รูปที่ 2.4 ขั้นตอนวิธีของการเรียนรู้แบบอาศัยประชากร

การทำงานจะเริ่มต้นด้วยการสร้างเวกเตอร์ความน่าจะเป็นขึ้นมา ในที่นี้กำหนดให้แต่ละมิติของเวกเตอร์มีค่าความน่าจะเป็นเท่ากับ 0.5 นั่นคือให้การกระจายตัวของคำตอบซึ่งเป็นบิต 0 หรือ 1 มีโอกาสเกิดขึ้นได้เท่า ๆ กัน จากนั้นจะสุ่มสร้างตัวอย่างที่เป็นตัวแทนของคำตอบขึ้นมาจากเวกเตอร์ความน่าจะเป็นนี้ M ตัวอย่าง ทำการประเมินค่าความเหมาะสมของตัวอย่างเหล่านี้แล้วเลือกคำตอบดี ๆ มา N ตัว แล้วจะนำตอบตอบที่เลือกมาได้นี้เป็นต้นแบบใน

การปรับปรุงค่าความน่าจะเป็นในแต่ละมิติเพื่อให้การกระจายตัวของคำตอบในรุ่นถัดไปเข้าใกล้กลุ่มตัวอย่างชุดนี้ การปรับปรุงค่าความน่าจะเป็นในเวกเตอร์ทำตามสูตรในขั้นตอนที่ 4

(2) ขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปร

ขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปร นำเสนอโดย Mühlenbein และ Paaß (1996) การประมาณการกระจายตัวของคำตอบดี ๆ โดยวิธีการนี้อาศัยความถี่ของแต่ละบิตปรากฏ แล้วปรับปรุงค่าในเวกเตอร์ความน่าจะเป็นตามความถี่นั้น ซึ่งขั้นตอนการทำงานแสดงในรูปที่ 2.5 เมื่อ l เป็นจำนวนบิตของคำตอบ p_i คือ ค่าความน่าจะเป็นที่แต่ละบิตจะเป็น 1 และ x_k คือ ค่าในแต่ละบิตของตัวอย่างตัวที่ k

Procedure uni vari ate marginal di stri buti on al gori thm

1: Randomly generate M individuals.

2: Select N individuals according to a selection method, where $N \leq M$.

3: Estimate uni vari ate marginal probabilities (p_i) for each x_k .

for $i = 1$ to l do

$$p_i = \frac{1}{N} \sum_{k=1}^N x_k$$

4: Go to step 2 until a termination criterion is met.

รูปที่ 2.5 ขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปร

การทำงานจะเริ่มต้นจากการสุ่มสร้างตัวอย่างคำตอบขึ้นมา M ตัว ทำการประเมินคุณภาพคำตอบนั้นแล้วเลือกตัวอย่างดี ๆ ขึ้นมา N ตัว คำนวณหาค่าความน่าจะเป็นที่แต่ละบิตจะเป็น 1 ตามสูตรในขั้นตอนที่ 3 ในรูปที่ 2.5

(3) ขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับ

ขั้นตอนวิธีนี้นำเสนอโดย Harik et al. (1999) การปรับปรุงเวกเตอร์ความน่าจะเป็นจะปรับตามคำตอบที่ดีกว่า ในการทำงานของอัลกอริทึมนี้จะมีการสุ่มสร้างตัวอย่าง 2 ตัว ขึ้นมา จากเวกเตอร์ความน่าจะเป็น แล้วพิจารณาว่าคำตอบตัวใดดีกว่ากัน เวกเตอร์ความน่าจะเป็นในแต่ละมิติ จะถูกปรับตามบิตของคำตอบตัวที่ดีกว่า ถ้าในบิตนั้นคำตอบที่ดีกว่ามีค่าเป็น 1 เวกเตอร์ความน่าจะเป็นก็จะปรับเข้าใกล้ 1 แต่ถ้าคำตอบที่ดีกว่าเป็น 0 ค่าความน่าจะเป็นในมิตินั้นก็จะลดค่าลง โอกาสที่บิตนั้นจะถูกสร้างมาเป็น 0 ก็จะเพิ่มขึ้นตามด้วย ดังนั้นเมื่อผ่าน

กระบวนการวิวัฒนาการไประยะหนึ่ง เวกเตอร์ความน่าจะเป็นก็จะเป็นโมเดลการกระจายตัวของคำตอบดี ๆ การทำงานของขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับแสดงในรูปที่ 2.6

```

Procedure compact genetic algorithm
1: Initialize probability vector ( $p$ ).
   for  $i := 1$  to  $l$  do  $p_i := 0.5$ ;
2: Generate two individuals from the vector.
    $a := \text{generate}(p)$ ;
    $b := \text{generate}(p)$ ;
3: Let them compete.
    $winner, loser := \text{compete}(a, b)$ ;
4: Update the probability vector towards
   the better one.
   for  $i := 1$  to  $l$  do
     if  $winner_i \neq loser_i$  then
       if  $winner_i = 1$  then
          $p_i := p_i + 1/n$ 
       else
          $p_i := p_i - 1/n$ ;
5: Check if the vector has converged.
   for  $i := 1$  to  $l$  do
     if  $p_i > 0$  and  $p_i < 1$  then
       return to step 2;
  
```

รูปที่ 2.6 ขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับ

2.1.2 ปัญหาทดสอบ

ปัญหาทดสอบที่วิทยานิพนธ์นี้ใช้ในการเปรียบเทียบการทำงานและพฤติกรรมของอัลกอริทึมต่าง ๆ มีดังนี้

(1) ปัญหาจำนวนหนึ่งมากที่สุด (OneMax problem)

ปัญหานี้เป็นปัญหาง่ายสำหรับขั้นตอนวิธีเชิงพันธุกรรม และมักจะถูกใช้เป็นปัญหาทดสอบพฤติกรรมของอัลกอริทึมเพื่อเปรียบเทียบความสามารถในการแก้ปัญหา ค่าความเหมาะสมจะขึ้นอยู่กับจำนวนบิตที่เป็น 1 และค่าสูงสุดที่เป็นไปได้คือทุกบิตเป็น 1 ทั้งหมด ในกรณีนี้ค่าความเหมาะสมที่มากที่สุดจะเท่ากับความยาวของโครโมโซม

กำหนดให้ $\bar{x} = \{x_1, x_2, \dots, x_N\}$ เป็นโครโมโซมความยาว N , โดยที่แต่ละบิต $x_i \in \{0, 1\}$ การหาค่าความเหมาะสมของโครโมโซมนี้สามารถคำนวณได้จากสมการที่ 2.1

$$F(\bar{x}) = \sum_{i=1}^N x_i \quad (2.1)$$

(2) ปัญหาภัยกับดัก (Trap problem)

ปัญหาภัยกับดัก (Goldberg 1987) เป็นปัญหาที่ยากสำหรับขั้นตอนวิธีเชิงพันธุกรรม เนื่องจากฟังก์ชันหาค่าความเหมาะสมจะให้ค่ากับโครโมโซมที่ประกอบด้วยบิต 0 มากกว่าบิต 1 แต่ค่าสูงสุดของฟังก์ชันกลับได้มาจากโครโมโซมที่ประกอบด้วยบิต 1 ทั้งหมด

ปัญหาภัยกับดักนี้จะประกอบด้วยหน่วยสร้าง (Building block) ย่อย ๆ นำมาประกอบต่อกันเป็นภัยกับดักที่ยาวขึ้น หน่วยสร้างเล็ก ๆ เหล่านี้ยาว k บิต และสามารถหาค่าความเหมาะสมของแต่ละหน่วยสร้างได้ตามสมการที่ 2.2

$$F_k(b_0 \dots b_{k-1}) = \begin{cases} f_{\text{high}} & ; \text{ if } u = k \\ f_{\text{low}} - u \frac{f_{\text{low}}}{k-1} & ; \text{ otherwise} \end{cases} \quad (2.2)$$

เมื่อ $b_i \in \{0, 1\}$, $u = \sum_{i=0}^{k-1} b_i$, และ $f_{\text{high}} > f_{\text{low}}$. โดยปกติแล้ว f_{high} จะมีค่าเท่ากับ k และ f_{low} มีค่าเท่ากับ $k-1$

เมื่อนำหน่วยสร้างนี้มาประกอบต่อกันให้ยาวขึ้น ค่าความเหมาะสมสามารถคำนวณได้จากการนำค่าความเหมาะสมของแต่ละหน่วยสร้างย่อย ๆ มาบวกกัน ฟังก์ชันหาค่าความเหมาะสมของการนำหน่วยสร้างขนาด k มาต่อกัน m หน่วย แสดงในสมการที่ 2.3

$$F_{k \times m}(B_0 \dots B_{m-1}) = \sum_{i=0}^{m-1} F_k(B_i), B_i \in \{0, 1\}^k \quad (2.3)$$

(3) ปัญหาภัยกับดักแบบลำดับชั้น (Hierarchical-trap problem: Htrap)

ปัญหาภัยกับดักลำดับชั้น (Pelikan และ Goldberg 2001) เป็นปัญหาที่คำตอบประกอบกันขึ้นตามลำดับชั้น คำตอบในชั้นที่สูงขึ้นจะมาจากคำตอบย่อยในลำดับชั้นก่อนหน้า ซึ่งลำดับชั้นนี้สามารถแสดงได้ในรูปของต้นไม้ ในที่นี้จะพิจารณาคำตอบย่อยที่ละ 3 บิต ถ้าทั้งสามบิตเป็น 0 ทั้งหมด ในลำดับชั้นที่สูงขึ้นจะถูกแสดงด้วยบิต 0 เช่นเดียวกับบิต 1 ถ้าเป็น 1 ทั้งหมดทั้ง 3 บิตในชั้นถัดไปก็จะเป็นบิต 1 ด้วย ในกรณีอื่น ๆ จะถูกแสดงค่าด้วยเครื่องหมาย '-'

ค่าความเหมาะสมจะคำนวณจากค่าที่ได้จากแต่ละโหนดในต้นไม้ โดยเริ่มพิจารณาจากโหนดใบ (Leaf node) ซึ่งค่าของโหนด i คือ c_i คำนวณได้จากสมการที่ 2.4

$$c_i = \begin{cases} 3^h \times F_3(b_0 b_1 b_2) & ; \text{if } b_j \neq "-" \text{ for all } 0 \leq j \leq 2 \\ 0 & ; \text{otherwise} \end{cases} \quad (2.4)$$

เมื่อ h เป็นความสูงของโหนด i , และ b_0, b_1, b_2 เป็นโหนดทางซ้าย ตรงกลาง และทางขวาในลำดับชั้นก่อนหน้าของโหนด i โดยที่ค่า F_3 คำนวณจากสมการที่ 2.2 ข้างต้น โดยกำหนดให้ $f_{\text{high}} = 1$ และ $f_{\text{low}} = 0.9$ ส่วนโหนดอื่น กำหนดให้ $f_{\text{high}} = 1$ และ $f_{\text{low}} = 1$

(4) ปัญหาลำดับชั้นแบบก็ต่อเมื่อ (Hierarchical-if-and-only-if problem: HIFF)

ปัญหาลำดับชั้นแบบก็ต่อเมื่อ (Watson et al., 1998) เป็นอีกหนึ่งปัญหาลำดับชั้นที่สามารถแสดงได้ในรูปของต้นไม้ ปัญหานี้ถูกแสดงในรูปของต้นไม้ทวิภาค (Binary tree) ถ้าโหนดทั้งสองเป็น 1 หรือ 0 ทั้งคู่ ในลำดับชั้นที่อยู่สูงขึ้นไปจะแทนที่ด้วย 1 หรือ 0 ตามโหนดลูก แต่ถ้ามีโหนดใดโหนดหนึ่งเป็นค่าอื่น โหนดในลำดับชั้นถัดไปจะถูกแสดงแทนด้วยเครื่องหมาย '-' ค่าความเหมาะสมของโครโมโซมนี้คือผลรวมของค่าในแต่ละโหนดซึ่งคำนวณได้จากสมการที่ 2.5 เมื่อ h คือความสูงของโหนด i

$$c_i = \begin{cases} 2^h & ; \text{if node } i \text{ is "0" or "1"} \\ 0 & ; \text{if node } i \text{ is "-" } \end{cases} \quad (2.5)$$

ปัญหานี้มีคำตอบที่ดีที่สุดอยู่ 2 คำตอบคือ คำตอบที่ทุกบิตเป็น 1 ทั้งหมด กับคำตอบที่เป็น 0 ทั้งหมด

(5) ปัญหา 2 ขั้วแบบ 6 บิต (6-bipolar problem)

ปัญหา 2 ขั้วแบบ 6 บิต (Pelikan et al., 1999) เป็นปัญหาการลงขนาด 6 บิต การคิดค่าความเหมาะสมจะพิจารณาจากกลุ่มของ 6 บิต ซึ่งคำนวณได้จากสมการที่ 2.6 เมื่อ u คือจำนวนบิตที่เป็น 1 ในกลุ่มบิตนั้น

$$f_{6\text{-bipolar}} = \begin{cases} 1 & \text{if } u = 0 \text{ or } 6 \\ 0.9 & \text{if } u = 3 \\ 0.8 & \text{if } u = 2 \text{ or } 4 \\ 0 & \text{if } u = 1 \text{ or } 5 \end{cases} \quad (2.6)$$

เช่นเดียวกับปัญหาที่ตัดที่ไดกล่าวไว้ข้างต้น ปัญหา 2 ขั้วแบบ 6 บิตนี้ก็สามารถนำกลุ่มของคำตอบ 6 บิตมาต่อกันให้เป็นปัญหาที่ยาวขึ้นได้ ค่าความเหมาะสมก็จะคำนวณจากค่าคำตอบของกลุ่มย่อยมารวมกัน

2.2 ออปชันจริง

ในทางการเงิน ออปชัน (Option) คือ ตราสารอนุพันธ์ (Derivatives) ที่จะให้สิทธิแก่ผู้ถือครอง (โดยไม่ใช้ภาระผูกพัน) ในการที่จะตัดสินใจซื้อขายสินทรัพย์อ้างอิง (Underlying asset) ในราคาและระยะเวลาที่กำหนด โดยออปชันที่ให้สิทธิแก่ผู้ถือครองในการซื้อเรียกว่า คอลออปชัน (Call option) และออปชันที่ให้สิทธิแก่ผู้ถือครองในการขายเรียกว่า พุทออปชัน (Put option) ทั้งนี้ ในการซื้อขายออปชันดังกล่าว ผู้ขายออปชันจะต้องได้รับเงินส่วนหนึ่งจากผู้ซื้อออปชันซึ่งเรียกว่า ออปชันพรีเมียม (Option premium) เพื่อเป็นเงินตอบแทนสำหรับการแบกรับความเสี่ยงที่อาจจะเกิดขึ้นของผู้ขาย

ผู้ถือครองออปชันจะใช้สิทธิในการซื้อหรือขายสินทรัพย์อ้างอิงดังกล่าวเมื่อออปชันนั้นมีมูลค่า กล่าวคือ สถานะในสิทธินั้นก่อให้เกิดประโยชน์กับผู้ถือครองสิทธิ โดยสถานะที่ก่อให้เกิดประโยชน์แก่ผู้ถือคอลออปชัน จะเกิดขึ้นเมื่อราคาสินทรัพย์อ้างอิงภายใต้สัญญา ณ ขณะนั้นสูงกว่าราคาใช้สิทธิ (Exercise price) ในขณะที่สถานะที่ก่อให้เกิดประโยชน์แก่ผู้ถือพุทออปชัน จะเกิดขึ้นเมื่อราคาสินทรัพย์อ้างอิงภายใต้สัญญา ณ ขณะนั้นต่ำกว่าราคาใช้สิทธิ ตัวอย่างเช่น สมมุติว่าเรามีคอลออปชันของหุ้นในบริษัทหนึ่งไว้ในราคาใช้สิทธิ 100 บาท ถ้าหุ้นของบริษัทนี้ขึ้นเป็น 150 บาท เราก็จะสามารถซื้อหุ้นนี้ได้ ในราคา 100 บาทตามสัญญาที่ตกลงกันไว้ ส่วนต่าง 50 บาทที่เกิดขึ้นคือกำไรของผู้ถือคอลออปชัน ในทางกลับกันถ้าถือพุทออปชันของหุ้นบริษัทหนึ่งไว้ในราคาใช้สิทธิที่ 100 บาท ถ้าหุ้นของบริษัทนี้ตกลงเหลือ 50 บาท เราจะสามารถขายหุ้นได้ในราคา 100 บาท ดังนั้นส่วนกำไร 50 บาทที่ได้มาจะเป็นผลประโยชน์ของผู้ถือพุทออปชัน

สิ่งที่ทำให้ออปชันเป็นที่นิยมในการใช้ เป็นเครื่องมือจัดการความเสี่ยงของนักลงทุนก็คือ ความสามารถในการจำกัดขอบเขตของความเสียหายสูงสุด ในขณะที่โอกาสได้รับประโยชน์นั้นไม่จำกัด จากความสามารถนี้เองจึงมีผู้นำออปชันมาใช้ในการวิเคราะห์การลงทุนนอกตลาด โดยการประยุกต์ใช้ออปชันกับการตัดสินใจลงทุนในสินทรัพย์ที่ไม่ได้มีการซื้อขายในตลาดซึ่งเรียกว่า ออปชันจริง (Real options)

การวิเคราะห์มูลค่าของออปชันมีหลายวิธีด้วยกัน แต่วิธีที่นิยมและใช้งานสะดวกคือ การใช้แบบจำลองแบลค-โช (Black-Scholes option pricing model) (Black and Scholes, 1973) โดยมีสมมติฐานว่าการเคลื่อนไหวของราคาของสินทรัพย์อ้างอิงเคลื่อนไหวแบบบราวน์

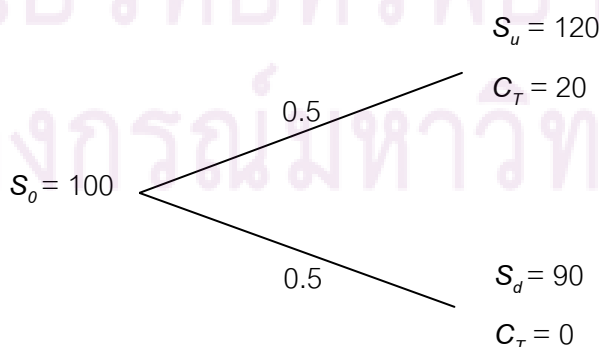
เรขาคณิต (Geometric Brownian motion) ซึ่งเป็นกระบวนการเฟ้นสุ่ม (Stochastic process) รูปแบบหนึ่ง อย่างไรก็ตามวิธีนี้ยังมีข้อจำกัดอื่น ๆ นอกจากสมมติฐาน เช่น ออปชันให้สิทธิในการใช้สิทธิได้ ณ วันหมดอายุของสัญญาเท่านั้น (European option) และอัตราดอกเบี้ยกับค่าความแปรปรวนของอัตราการเปลี่ยนแปลงของสินทรัพย์อ้างอิง ต้องมีค่าคงที่ตลอดอายุของสัญญาออปชัน

นอกจากนี้ ยังมีอีกหลายวิธีที่ใช้วิเคราะห์ราคาสัญญาออปชันเช่น การใช้ต้นไม้ทวินาม (Binomial option pricing model) (Cox และ Ross, 1979) หรือการใช้วิธีเชิงตัวเลข (Numerical approach) เช่น การใช้การจำลองแบบมอนติคาร์โล (Monte Carlo simulation) ซึ่งมีความยืดหยุ่นมากกว่า ในที่นี้จะกล่าวถึงการคำนวณมูลค่าของออปชันโดยใช้ต้นไม้ทวินาม ซึ่งเป็นพื้นฐานของการใช้แลตทิซ (Lattice) ที่วิทยานิพนธ์นี้จะใช้เป็นวิธีในการหาเวลาหยุดเหมาะสมสุดของขั้นตอนวิธีเชิงพันธุกรรมต่อไป

การใช้ต้นไม้ทวินามนั้นมีหลักการดังนี้ สมมติว่าสินทรัพย์อ้างอิงหนึ่งมีราคา S ณ เวลา t ในเวลา $t + \Delta t$ ราคาสินทรัพย์นั้น ๆ จะมีโอกาสสูงขึ้นเป็น S_u หรือลดลงเหลือ S_d เท่านั้น เนื่องจากมีโอกาสเป็นไปได้ 2 แบบคือไม่เพิ่มขึ้นก็ลดลง ดังนั้นราคาของคอลออปชันที่อ้างอิงอยู่กับสินทรัพย์นั้น (C) ก็มีโอกาสเป็นไปได้ 2 อย่างเช่นกัน ตัวอย่างเช่น ในรูปที่ 2.7 ถ้าหากกำหนดให้ $S_0 = 100$, $S_u = 120$ ด้วยความน่าจะเป็น 0.5 และ $S_d = 90$ ด้วยความน่าจะเป็น 0.5 ถ้าราคาของสินทรัพย์เพิ่มขึ้นเป็น 120 คอลออปชันก็มีมูลค่าเท่ากับ 20 ในทางตรงกันข้ามหากราคาสินทรัพย์ลดลงเหลือ 90 คอลออปชันก็จะมีค่าเท่ากับ 0 เนื่องจากการคำนวณมูลค่าของคอลออปชันในวันใช้สิทธิคิดจากสมการที่ 2.7

$$C_T = \max(0, S_T - K) \quad (2.7)$$

เมื่อ C_T คือ ราคาออปชัน ณ วันใช้สิทธิ
 S_T คือ ราคาสินทรัพย์ ณ วันใช้สิทธิ
 K คือ ราคาใช้สิทธิ (Exercise price)



รูปที่ 2.7 ตัวอย่างการคำนวณมูลค่าออปชันจากต้นไม้ทวินาม

ในการหาเวลาหยุดที่เหมาะสมที่สุด จะใช้หลักการเช่นเดียวกับการคำนวณมูลค่าออปชัน กล่าวคือ การตัดสินใจ ณ เวลาใดเวลาหนึ่ง จะเลือกค่าที่มากที่สุดระหว่างการหยุดกับการดำเนินการต่อ สมการสำหรับการหาเวลาหยุดที่เหมาะสมที่สุดได้มาจากสมการเบลแมน (Bellman equation) ตามที่ Dixit และ Pindyck (1994) แสดงไว้ดังสมการที่ 2.8 นี้

$$F(x) = \max \left\{ \Omega(x), \pi(x) + \frac{1}{1 + \rho} \varepsilon[F(x') | x] \right\} \quad (2.8)$$

เมื่อ $F(x)$ คือ มูลค่าของโอกาสในการลงทุน

$\Omega(x)$ คือ มูลค่าที่จะได้รับถ้าตัดสินใจลงทุน

$\pi(x)$ คือ ผลกำไรที่จะได้รับ (Profit flow)

ρ คือ อัตราคิดลด (Discount rate)

$\varepsilon[F(x')|x]$ คือ ค่าคาดหวังของมูลค่าของโอกาสการลงทุนในเวลาถัดไป เมื่อสินทรัพย์อ้างอิงมีมูลค่า x ในปัจจุบัน

ในการคำนวณเวลาหยุดที่เหมาะสมที่สุด จะคำนวณจากเวลาสุดท้ายย้อนกลับมา และผลการตัดสินใจเลือกว่าจะหยุดหรือดำเนินการต่อในแต่ละชั้นเวลา จะเป็นเงื่อนไขที่บอกว่าการตัดสินใจที่เหมาะสมควรเป็นเช่นไร ค่า x ที่เป็นเงื่อนไขในการตัดสินใจหยุดหรือดำเนินการต่อ จะทำให้เราทราบถึงขอบเขตการตัดสินใจ ซึ่งควรจะตัดสินใจหยุดเมื่อค่าของสินทรัพย์อ้างอิงตกอยู่ในด้านหนึ่งของค่า x ในทางกลับกันถ้าค่าของสินทรัพย์อ้างอิงตกอยู่ในอีกด้านหนึ่ง การตัดสินใจดำเนินการต่อจะเป็นการตัดสินใจที่เหมาะสมกว่า ค่า x ที่เป็นเส้นแบ่งการตัดสินใจในชั้นเวลาต่าง ๆ นี้ จะสร้างเป็นขอบเขตการใช้สิทธิ (Exercise region) และจะนำขอบเขตการใช้สิทธิไปเป็นแนวทางในการตัดสินใจที่เหมาะสมต่อไป รายละเอียดของการนำวิธีออปชันจริงไปประยุกต์ใช้กับการหาเวลาหยุดที่เหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมจะนำเสนอในบทที่ 3

2.3 เงื่อนไขการหยุดของขั้นตอนวิธีเชิงพันธุกรรม

เป็นที่ทราบกันดีว่าขั้นตอนวิธีเชิงพันธุกรรมได้มีการนำไปใช้ในงานต่าง ๆ อย่างกว้างขวาง จากขั้นตอนวิธีในรูปที่ 2.1 กระบวนการต่าง ๆ จะถูกดัดแปลงและประยุกต์ใช้ให้เหมาะสมกับปัญหาเฉพาะด้าน แต่กระบวนการหลักที่ทำการปรับปรุงคุณภาพคำตอบและเลือกคำตอบที่ดีที่สุดให้ยู่รอดก็ยังคงเหมือนเดิม ขั้นตอนต่าง ๆ ในกระบวนการวิวัฒนาการมีการนิยามไว้ชัดเจน เช่น ให้สร้างประชากรรุ่นใหม่โดยการไขว้เปลี่ยน การกลายพันธุ์ แล้วทำการเลือกสรร แต่สิ่งที่ขั้นตอนวิธีไม่ได้บอกไว้ก็คือเงื่อนไขการหยุด ในกระบวนการทำงานทั้งหมดบอกเพียงแต่จะให้ทำซ้ำจนกว่าจะตรงกับเงื่อนไขการหยุด โดยมีได้ระบุไว้ชัดเจนว่าเงื่อนไขการหยุดนี้คืออะไร

การกำหนดเงื่อนไขการหยุดทำงานของอัลกอริทึมการเรียนรู้ เป็นหนึ่งในเงื่อนไขที่ต้องระบุเมื่อต้องการจะใช้งาน วิธีทั่วไปที่นิยมใช้กันอย่างแพร่หลายก็คือการกำหนดจำนวนรอบในการเรียนรู้ หรือกำหนดคุณภาพของผลลัพธ์ในระดับที่ผู้ใช้อยอมรับได้ ซึ่งการกำหนดเงื่อนไขเหล่านี้ส่งผลต่อคุณภาพคำตอบและเวลาที่ต้องใช้ในการหาคำตอบเหล่านั้น จึงมีงานวิจัยจำนวนหนึ่งที่ได้ทำการศึกษาเกี่ยวกับเงื่อนไขการหยุดแบบต่าง ๆ ในที่นี้จะแบ่งเงื่อนไขการหยุดออกเป็น 3 ประเภทหลัก ๆ ได้แก่

(1) เงื่อนไขการหยุดโดยทั่วไป

เงื่อนไขการหยุดในกลุ่มนี้คือวิธีที่ใช้กันอยู่โดยทั่วไป ไม่เพียงแต่เฉพาะในขั้นตอนวิธีเชิงพันธุกรรม แต่ยังพบเงื่อนไขการหยุดในลักษณะนี้ในขั้นตอนวิธีการเรียนรู้แบบอื่น ๆ ด้วย เนื่องจากวิธีกำหนดเงื่อนไขการหยุดโดยวิธีนี้ง่ายต่อการใช้งานจึงพบเห็นการนำไปใช้เป็นจำนวนมาก Michalewicz (1996) และ Zielinski et al. (2005) ได้รวบรวมเงื่อนไขการหยุดที่ใช้ในขั้นตอนวิธีเชิงพันธุกรรมไว้ดังนี้

- หยุดเมื่อถึงจำนวนรอบสูงสุดที่กำหนดไว้
- หยุดเมื่อจำนวนครั้งในการคำนวณค่าความเหมาะสมถึงค่าที่กำหนดไว้
- หยุดเมื่อค่าความเหมาะสมในรุ่นถัดไปมีการเปลี่ยนแปลงเพียงเล็กน้อยเมื่อเทียบกับรุ่นปัจจุบัน
- หยุดเมื่อพบคำตอบที่ดีที่สุดแล้ว
- หยุดเมื่อคำตอบมีความคล้ายกันมาก
- หยุดโดยใช้เงื่อนไขหลายข้อรวมกัน

ในการใช้เงื่อนไขการหยุดเหล่านี้ Michalewicz (1996) ยังกล่าวไว้ว่าเราสามารถมองได้ใน 2 ระดับ คือ มองที่โครงสร้างของโครโมโซม (Genotype) หรือมองที่การตีความความหมายของโครโมโซมนั้นออกมาแล้ว (Phenotype) เช่น ในการพิจารณาว่าคำตอบมีความคล้ายคลึงกันมากเพียงใด อาจมองที่โครโมโซมมีความแตกต่างกันเพียงใด หรือมองที่ค่าความเหมาะสมที่ตีความออกมาก็ได้ ทั้งนี้แล้วแต่ผู้ใช้จะเลือกพิจารณา

นอกจากเงื่อนไขการหยุดที่กล่าวมาข้างต้น ยังพบว่าในบางอัลกอริทึมหรือบางปัญหาที่มีการประยุกต์ใช้วิธีการหยุดแบบต่าง ๆ เพื่อให้เหมาะสมกับวิธีการนั้น ๆ มีงานวิจัยจำนวนหนึ่งที่ได้นำเสนอวิธีการหยุดเฉพาะแบบ เช่น Hernandez et al. (2005) ได้เสนอการใช้ความเชื่อมโยงในอดีต (Coupling from the past) ในการหยุดของขั้นตอนวิธีเชิงพันธุกรรมที่เขาได้ออกแบบมา ซึ่งความเชื่อมโยงในอดีตในที่นี้ก็คือการมองกลุ่มประชากรเป็นสถานะต่าง ๆ การเปลี่ยนแปลงประชากรในรุ่นต่าง ๆ ด้วยตัวดำเนินการทางพันธุกรรมก็เปรียบเสมือนการเปลี่ยนจากสถานะหนึ่งเป็นอีกสถานะหนึ่ง ซึ่งสามารถอธิบายได้โดยอาศัยหลักลูกโซ่มาร์คอฟ

(Markov chain) และในที่สุดแล้วอัลกอริทึมจะหยุดเมื่อพบว่าอยู่ในสถานะคงที่ (Stationary) คือ มีประชากรที่เป็นคำตอบที่ดีที่สุดปรากฏอยู่ ข้อจำกัดของเงื่อนไขการหยุดนี้คือใช้ได้กับขั้นตอนวิธีเชิงพันธุกรรมที่สามารถพิสูจน์ได้ว่ามีการกระจายตัวแบบคงที่ของหลักลูกโซ่มาร์คอฟเท่านั้น ส่วนในงานของ Meyer และ Feng (1994) ได้นำเสนอเงื่อนไขการหยุดแบบคลุมเครือ (Fuzzy stop criterion) โดยการทำนายค่าคำตอบที่ดีที่สุดว่าจะเป็นเท่าใด และอัลกอริทึมจะหยุดเมื่อได้คำตอบที่ใกล้เคียงคำตอบที่ดีที่สุดที่ประมาณเอาไว้ ที่จัดกลุ่มให้การใช้เงื่อนไขการหยุดแบบนี้อยู่ในกลุ่มเดียวกับเงื่อนไขการหยุดโดยทั่วไปก็เนื่องจาก ถึงแม้ว่าทั้ง 2 วิธีการหยุดข้างต้นจะถูกออกแบบมาเป็นพิเศษสำหรับอัลกอริทึมเฉพาะแบบ แต่วิธีเหล่านี้ก็ยังอยู่บนหลักพื้นฐานของการพิจารณาการลู่เข้าของคำตอบ ซึ่งต่างจากแนวคิดในการศึกษาเวลาหยุดทางทฤษฎีและแนวคิดในแง่ความคุ้มทุนที่จะกล่าวถึงในหัวข้อถัดไป

(2) เงื่อนไขการหยุดตามทฤษฎี

มีงานวิจัยจำนวนหนึ่งที่ได้ทำการศึกษาถึงจำนวนรอบที่ขั้นตอนวิธีเชิงพันธุกรรม ต้องใช้ในการหาคำตอบ งานวิจัยเหล่านี้ได้ศึกษาในเชิงทฤษฎีเพื่อหาขอบเขต (Bound) ของจำนวนรุ่นที่ใช้ในการลู่เข้าสู่คำตอบ โดยหลักการพื้นฐานที่ใช้ในการวิเคราะห์ตามทฤษฎีเหล่านี้ คือ การอธิบายพฤติกรรมการทำงานของขั้นตอนวิธีเชิงพันธุกรรมด้วยหลักลูกโซ่มาร์คอฟ (Nix และ Vose, 1992) หลักลูกโซ่มาร์คอฟเป็นกระบวนการเฟ้นสุ่มชนิดหนึ่งที่มีคุณสมบัติว่า ณ สถานะในสถานะหนึ่งจะเป็นเช่นใด จะขึ้นกับการเปลี่ยนแปลงจากสถานะก่อนหน้าเท่านั้น ในโมเดลของ Nix และ Vose ได้อธิบายขั้นตอนวิธีเชิงพันธุกรรมด้วยหลักการดังกล่าว โดยการมองว่ากลุ่มประชากรก็คือสถานะหนึ่ง ๆ (State) ในกระบวนการเฟ้นสุ่ม และการเปลี่ยนแปลง (Transition) ก็คือขั้นตอนการปรับปรุงคุณภาพคำตอบของขั้นตอนวิธีเชิงพันธุกรรม ได้แก่ การไขว้เปลี่ยน การกลายพันธุ์ การเลือกสรร เขาได้สร้างสูตรสำหรับการเปลี่ยนแปลงสถานะจากประชากรรุ่นหนึ่งไปสู่อีกรุ่นหนึ่งเป็นค่าความน่าจะเป็นของการเปลี่ยนแปลง (Transition probabilities) ซึ่งเป็นฟังก์ชันของอัตราการไขว้เปลี่ยนและการกลายพันธุ์ และโดยการสร้างเมทริกซ์ของค่าความน่าจะเป็นของการเปลี่ยนแปลงประชากรนี้เอง ทำให้นำไปสู่การวิเคราะห์พฤติกรรมการทำงานและการลู่เข้าสู่คำตอบของขั้นตอนวิธีเชิงพันธุกรรม

Aytug และ Koehler (1996) นำเสนอจำนวนรอบที่ขั้นตอนวิธีเชิงพันธุกรรมจะพบคำตอบที่ดีที่สุดภายใต้ค่าความเชื่อมั่นค่าหนึ่งโดยอาศัยหลักการข้างต้น ขอบเขตนี้หามาจากความน่าจะเป็นที่อัลกอริทึมจะเห็นตัวอย่างประชากรทั้งหมด กำหนดให้ n คือ จำนวนประชากร / คือ ความยาวของโครโมโซม μ คือ อัตราการกลายพันธุ์แบบเอกรูปซึ่งอยู่ในช่วง $(0, 1)$ (Uniform mutation rate) α คือ ค่าความเชื่อมั่นซึ่งอยู่ในช่วง $[0, 1)$ (Confidence probability) ขอบเขตด้านบนของจำนวนรอบที่ต้องใช้ในการหาคำตอบที่ดีที่สุดภายใต้ความเชื่อมั่นที่ค่า α คือ

$$\left[\frac{\ln(1-\alpha)}{\ln(1-\min\{(1-\mu)^n, \mu^n\})} \right] \quad (2.9)$$

และจากขอบเขตดังกล่าว Aytug และ Koehler (2000) ได้ปรับปรุงขอบเขตที่หาได้ให้กระชับขึ้นดังนี้

$$\left[\frac{\ln(1-\alpha)}{n \ln(1-\min[\mu^l, (1-\mu)^l])} \right] \quad (2.10)$$

ต่อมา Pendharkar และ Koehler (2007) ได้ทำการปรับปรุงและนำเสนอขอบเขตที่เป็นพฤติกรรมโดยเฉลี่ย โดยกำหนดให้ $|P|$ คือ จำนวนสถานะในลูกโซ่มาร์คอฟ ขอบเขตดังกล่าวแสดงดังนี้

$$1 - |P| \min(\mu^n, (1-\mu)^n) \quad (2.11)$$

นอกจากการหาขอบเขตจำนวนรุ่นที่ขั้นตอนวิธีเชิงพันธุกรรมต้องใช้ในการหาคำตอบแล้ว ยังมีงานวิจัยอีกกลุ่มหนึ่งที่ได้ศึกษาพฤติกรรมของขั้นตอนวิธีเชิงพันธุกรรมจากเวลาที่จะพบคำตอบแรก (First hitting time) งานวิจัยกลุ่มนี้มุ่งศึกษาไปที่ความซับซ้อนในด้านเวลาการคำนวณของ (1+1)-EA ซึ่งเป็นวิธีการคำนวณเชิงวิวัฒนาการแบบที่ง่ายที่สุด คือ ในแต่ละรุ่นมีประชากรเพียงตัวเดียว และจะสร้างประชากรใหม่ขึ้น 1 ตัวแล้วเลือกประชากรตัวที่ดีกว่าให้อยู่รอด ในอัลกอริทึมนี้พบว่าสำหรับปัญหาจำนวนหนึ่งมากที่สุด (Onemax problem) อัลกอริทึมจะใช้เวลา $O(n \log n)$ สำหรับ (1+1)-EA ที่ใช้การกลายพันธุ์ที่ละบิต (Rudolph, 1997) อย่างไรก็ตามการวิเคราะห์ความซับซ้อนด้านเวลาสำหรับขั้นตอนวิธีเชิงพันธุกรรมแบบอื่น หรือกับปัญหาอื่นยังเป็นเรื่องยาก เนื่องจากความซับซ้อนในการทำงานของอัลกอริทึม

(3) เงื่อนไขการหยุดในแง่ความคุ้มทุน

เป็นที่ยอมรับกันโดยทั่วไปว่า ในการใช้งานจริงเงื่อนไขการหยุดทางทฤษฎีนั้น มีข้อจำกัดในหลาย ๆ ด้าน เช่น ใช้ได้เฉพาะบางสถานการณ์เท่านั้น และส่วนมากทฤษฎีเหล่านี้ก็จะให้ขอบเขตที่กว้างกว่าความเป็นจริง จึงมีนักวิจัยส่วนหนึ่งให้ความสนใจในด้านการใช้งานจริง บางครั้งเราอาจต้องเสียเวลาการคำนวณนานมากเพื่อให้ได้คุณภาพคำตอบที่ดีขึ้นเพียงเล็กน้อย การยุติการทำงานของอัลกอริทึมอย่างคุ้มทุนเป็นอีกทางเลือกหนึ่งที่เป็นไปได้

Sandholm (2003) ได้กล่าวไว้ว่าความยากของการตัดสินใจยุติการทำงานของอัลกอริทึมอย่างคุ่มที่สุดนั้นขึ้นอยู่กับหลายปัจจัย เช่น

- ถ้าปล่อยให้อัลกอริทึมทำงานต่อไป อาจเจอคำตอบที่ดีขึ้น
- ความรู้ที่เพิ่มขึ้นจากการทดลองหาคำตอบที่มากขึ้น อาจช่วยให้การตัดสินใจในอนาคตแม่นยำขึ้น เพราะถ้าหยุดขณะนี้ความรู้ที่มีอยู่อาจไม่เพียงพอต่อการตัดสินใจที่ดีก็เป็นได้
- ถ้าปล่อยให้อัลกอริทึมทำงานต่อไปเรื่อย ๆ ประโยชน์ที่จะได้จากการนำคำตอบไปใช้ได้เร็วก็จะลดลงเรื่อย ๆ
- การคำนวณที่มากขึ้นส่งผลต่อต้นทุนการคำนวณที่มากขึ้นด้วย

จากแนวคิดดังกล่าว Sandholm (2003) จึงได้เสนอวิธีตัดสินใจหยุดสำหรับอัลกอริทึมการตัดสินใจแบบไม่สมบูรณ์ (Incomplete decision algorithm) ซึ่งเป็นกลุ่มของปัญหาการตัดสินใจที่คำตอบมีเพียง 'ใช่' หรือ 'ไม่ใช่' เท่านั้น ในปัญหาการตัดสินใจแบบไม่สมบูรณ์ถ้าอัลกอริทึมตอบว่า 'ไม่ใช่' หมายความว่าอัลกอริทึมจะไม่มีวันหยุด แต่ถ้าคำตอบเป็น 'ใช่' อัลกอริทึมอาจจะหยุดหรือไม่หยุดก็ได้ ดังนั้นถ้าอัลกอริทึมหยุดก็แสดงว่าคำตอบคือ 'ใช่' Sandholm ได้หาเวลาเหมาะสมที่สุดที่จะหยุดการทำงานของอัลกอริทึมนี้โดยคำนวณโอกาสที่คำตอบจะเป็น 'ใช่'

ในการหาเวลาหยุดในแง่ความคุ่มทุนของขั้นตอนวิธีเชิงพันธุกรรมซับซ้อนกว่านั้น เพราะเราไม่รู้ว่าค่าความเหมาะสมที่มีอยู่นี้ใช่คำตอบที่ดีที่สุดหรือไม่ และในหลาย ๆ ปัญหา คำตอบที่เป็นไปได้ก็มีความเป็นไปได้ที่หลากหลาย อย่างไรก็ตามมีงานวิจัยหนึ่ง (Hulin, 1997) ที่พยายามหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมโดยคาดคะเนว่าคำตอบปัจจุบันห่างจากคำตอบที่ดีที่สุดเพียงใด ต้นทุนที่ต้องใช้ในการคำนวณเพื่อที่จะได้คำตอบนั้นถูกนำมาคิดด้วย และอัลกอริทึมจะหยุดเมื่อคำนวณแล้วมีโอกาสในการขาดทุนสูงขึ้น

จากแนวคิดในการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมในแง่ของความคุ่มทุน วิทยานิพนธ์นี้จึงนำแนวคิดทางเศรษฐศาสตร์มาใช้ในการวิเคราะห์เวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรม วิธีที่นำเสนอไม่ได้ขึ้นกับประเภทของขั้นตอนวิธีเชิงพันธุกรรม และสามารถนำมาวิเคราะห์เวลาหยุดในปัญหาต่าง ๆ ได้ไม่จำกัด เพียงขอให้ข้อมูลการเปลี่ยนแปลงของค่าความเหมาะสมในการหาคำตอบของอัลกอริทึมนั้น ๆ วิธีนี้ก็สามารถแสดงให้เห็นได้ว่าเวลาหยุดเหมาะสมควรเป็นเช่นไร

วิธีที่วิทยานิพนธ์นี้นำมาใช้ก็คือ วิธีอุปชันจริง มีงานวิจัยจำนวนมากที่นำเสนอการใช้อุปชันจริงในการวิเคราะห์ความเสี่ยงในการลงทุนหลาย ๆ ด้าน จึงเป็นที่น่าสนใจว่าการตัดสินใจหยุดทำงานของอัลกอริทึม สามารถมองเทียบได้กับการตัดสินใจใช้สิทธิของอุปชันที่

ถือครอง การตัดสินใจใช้สิทธิเทียบได้กับการหยุดทำงานของอัลกอริทึม การถือออพชันเทียบได้กับการปล่อยให้อัลกอริทึมดำเนินการต่อไป เมื่อใดที่ใช้สิทธินั้นคือการยุติการทำงานทั้งหมดและไม่สนใจความเป็นไปได้ในอนาคตอีก ซึ่งเวลาหยุดเหมาะสมที่สุดสามารถคำนวณได้จากมูลค่าอัลกอริทึมที่พิจารณาความคุ้มทุนต่อโอกาสการพบคำตอบที่ดีกว่า

งานวิจัยในอดีตที่เกี่ยวข้องกับวิธีออพชันจริงและขั้นตอนวิธีเชิงพันธุกรรม ล้วนแล้วแต่เป็นการนำการคำนวณเชิงวิวัฒนาการไปใช้ในการสร้างตัวแบบการคำนวณมูลค่าของออพชัน (Chen และ Lee, 1997; Chidambaran, Lee และ Trigueros, 1998; Chen, Yeh และ Lee, 1998; Chen, Lee และ Yeh, 1999; Chidambaran, 2003; Yin, Brabazon และ O'Sullivan, 2007) วิทยานิพนธ์นี้ทำตรงกันข้ามคือนำออพชันเข้ามาใช้ในขั้นตอนวิธีเชิงพันธุกรรมซึ่งเป็นวิธีใหม่ในการประยุกต์ใช้วิธีออพชันจริงในปัญหาการคำนวณ

การใช้วิธีออพชันจริงในการวิเคราะห์เวลาหยุดของขั้นตอนวิธีเชิงพันธุกรรมให้ผลลัพธ์ 2 อย่าง คือ วิธีตัดสินใจหยุดที่เหมาะสมที่สุด และมูลค่าของอัลกอริทึม ผลลัพธ์อย่างแรกเราสามารถนำมาใช้เป็นเงื่อนไขการหยุดและอธิบายพฤติกรรมของขั้นตอนวิธีเชิงพันธุกรรมในการหาคำตอบ ส่วนผลลัพธ์อย่างหลังเป็นตัวเลขที่สามารถนำมาเปรียบเทียบความเหมาะสมของขั้นตอนวิธีพันธุกรรมชนิดต่าง ๆ ในแง่ความคุ้มทุน ซึ่งพิจารณาถึงต้นทุนการคำนวณ เวลา และความเป็นไปได้ที่จะพบคำตอบที่ดีขึ้น มีมาตรวัดจำนวนมากที่ใช้วัดความยากง่ายของปัญหาสำหรับขั้นตอนวิธีเชิงพันธุกรรม (Davidor, 1991; Kauffman, 1993; Jones และ Forrest, 1995; Borenstein และ Poli, 2004; Borenstein และ Poli, 2005) วิธีการวัดเหล่านี้เป็นการวิเคราะห์ความยากในการหาคำตอบของอัลกอริทึม โดยการวิเคราะห์ค่าความเหมาะสมที่เป็นไปได้ในปริภูมิการหาคำตอบ แต่ในหลาย ๆ ปัญหา การวิเคราะห์โดยวิธีนี้ก็ซับซ้อนมากเนื่องจากขนาดของปัญหาและจำนวนคำตอบที่เป็นไปได้มีขนาดใหญ่ ดังนั้นจึงมีผู้ศึกษาความเป็นไปได้ของการออกแบบมาตรวัดที่สามารถวัดความยากง่ายของทุกปัญหาได้อย่างมีประสิทธิภาพ ซึ่งการหามาตรวัดที่ซับซ้อนความยากง่ายได้กับทุกปัญหาถูกพิสูจน์แล้วว่าไม่มีอยู่จริง (He et al., 2007) เนื่องจากไม่สามารถวัดความยากง่ายของกลุ่มปัญหา NP (ทุกกรณีตัวอย่างที่เป็นไปได้) ได้ในเวลาแบบพหุนาม (Polynomial time) มูลค่าอัลกอริทึมที่วิทยานิพนธ์นี้นำเสนออาจดูคล้ายเป็นมาตรวัดความยากง่ายดังกล่าว แต่อันที่จริงมันไม่ได้นำมาใช้วัดความยากง่ายของปัญหา แต่ค่าที่ได้จากวิธีการออพชันจริงนี้ใช้วัดมูลค่าของขั้นตอนวิธีเชิงพันธุกรรมในแง่ความคุ้มทุนในการใช้อัลกอริทึมประเภทหนึ่งกับปัญหาประเภทหนึ่ง เมื่อดำเนินถึงโอกาสที่จะได้คำตอบที่ดีขึ้นภายใต้ต้นทุนและระยะเวลาที่มีอยู่จำกัด

บทที่ 3

การใช้ข้ออุปชันจริงในการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรม

ในบทนี้จะกล่าวถึงการใช้ข้ออุปชันจริงในการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรม ซึ่งมีขั้นตอนหลัก ๆ 4 ขั้นตอน ดังนี้

3.1 การสร้างตัวแบบความไม่แน่นอนมูลฐาน (Modeling underlying uncertainty)

ขั้นตอนแรกของการวิเคราะห์เวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมคือการสร้างตัวแบบความไม่แน่นอนของการค้นหาคำตอบ โดยทั่วไปตัวแบบความไม่แน่นอนนี้สามารถสร้างได้จากกระบวนการเฟ้นสุ่ม (Stochastic process) โดยการกำหนดตัวแปรต่าง ๆ ให้เหมาะสมกับการเคลื่อนไหวของราคาของสินทรัพย์อ้างอิง แต่ในขั้นตอนวิธีเชิงพันธุกรรมเราสามารถทราบการเปลี่ยนแปลงของคุณภาพคำตอบเมื่อผ่านกระบวนการวิวัฒนาการไปในแต่ละระยะได้จากการทดลองจริง ดังนั้นตัวแบบความไม่แน่นอนของขั้นตอนวิธีเชิงพันธุกรรมสามารถสร้างได้โดยการเก็บข้อมูลการเปลี่ยนแปลงค่าความเหมาะสมในแต่ละรุ่น เมื่อทำการทดลองซ้ำ ๆ เราก็จะได้ความน่าจะเป็นที่ค่าความเหมาะสมจะมีค่าเป็นเท่าใดในเวลาต่าง ๆ ค่าความน่าจะเป็นของการเปลี่ยนแปลงค่าความเหมาะสมนี้จะถูกเก็บอยู่ในเมทริกซ์ กำหนดให้เมทริกซ์ M_{ij}^k แทนค่าความน่าจะเป็นที่ค่าความเหมาะสมในรุ่นที่ k จะเปลี่ยนจากค่า i เป็นค่า j ตัวอย่างเมทริกซ์ค่าความน่าจะเป็นในการเปลี่ยนแปลงค่าความเหมาะสมของขั้นตอนวิธีเชิงพันธุกรรมจากรุ่นที่ 1 ถึงรุ่นที่ n แสดงในรูปที่ 3.1

Fitness	0	1	2	3	fitness	0	1	2	3	fitness	0	1	2	3	
0	0.2	0.8	0.0	0.0	0	0.0	0.9	0.1	0.0	0	0.0	0.0	0.0	0.0	
1	0.1	0.4	0.5	0.0	1	0.1	0.2	0.6	0.1	...	1	0.0	0.0	0.0	0.0
2	0.0	0.2	0.6	0.2	2	0.0	0.1	0.3	0.6		2	0.0	0.0	1.0	0.0
3	0.0	0.0	0.0	0.0	3	0.0	0.1	0.2	0.7		3	0.0	0.0	0.0	1.0
generation 1 -> 2					generation 2 -> 3					generation $n-1$ -> n					

รูปที่ 3.1 ตัวอย่างเมทริกซ์ค่าความน่าจะเป็นในการเปลี่ยนแปลงค่าความเหมาะสม

จากตัวอย่างในรูปที่ 3.1 สมมติว่าค่าความเหมาะสมที่จะเป็นไปได้คือ 0 ถึง 3 ค่าที่อยู่ในเมทริกซ์นี้คือค่าความน่าจะเป็นที่ค่าความเหมาะสมใด ๆ จะเปลี่ยนแปลงไปในรุ่นถัดไป เช่น ในรุ่นที่ 1 ค่าความเหมาะสม 0 มีโอกาสไม่เปลี่ยนแปลงไปในรุ่นที่ 2 ด้วยความน่าจะเป็น 0.2 และมีโอกาสเพิ่มเป็น 1 ด้วยความน่าจะเป็น 0.8 เป็นต้น

3.2 การสร้างฟังก์ชันสำหรับคำนวณมูลค่าที่ได้รับจากการทำงานของอัลกอริทึมเมื่อมีการเลือกหยุด

ก่อนที่จะทำการคำนวณมูลค่าอัลกอริทึม เราจะต้องนิยามฟังก์ชันที่เป็นตัวกำหนดมูลค่าของคำตอบรวมทั้งต้นทุนการคำนวณว่าเป็นเท่าใด การให้น้ำหนักกับเวลาที่ใช้ในการได้มาซึ่งคำตอบก็ถูกกำหนดอยู่ในส่วนนี้ด้วยเช่นกัน

ฟังก์ชันการคำนวณมูลค่าอัลกอริทึมจะนำสมการเบลแมนที่ได้กล่าวไว้ในบทที่ 2 (สมการที่ 2.8) มาเป็นต้นแบบ

กำหนดให้ $\Omega(x)$ แทนมูลค่าของคำตอบที่จะได้รับเมื่อหยุดการทำงาน ซึ่งมูลค่านี้จะเท่ากับค่าความเหมาะสมที่ได้จากขั้นตอนวิธีเชิงพันธุกรรม ดังนั้นมูลค่าของคำตอบเมื่อตัดสินใจหยุดการทำงานของอัลกอริทึมสามารถหาได้จากสมการที่ 3.1 ดังนี้

$$\Omega(x) = g(x) \quad (3.1)$$

เมื่อ $g(x)$ เป็นค่าความเหมาะสมของคำตอบ x

นอกจากการสร้างฟังก์ชันในการหามูลค่าของคำตอบถ้าตัดสินใจหยุดแล้ว ก็ต้องกำหนดฟังก์ชันที่ใช้คำนวณมูลค่าเมื่อให้อัลกอริทึมดำเนินการต่อ จากสมการเบลแมนพจน์ที่ใช้กำหนดมูลค่าถ้าดำเนินการต่อก็คือ

$$\pi(x) + \frac{1}{1+\rho} \mathcal{E}[F(x') | x]$$

ในที่นี้เราสามารถตัดเทอม $\pi(x)$ ออกได้ เนื่องจากในการทำงานของขั้นตอนวิธีเชิงพันธุกรรมไม่ได้มีกระแสกำไร (Profit flow) เข้ามาเกี่ยวข้อง พจน์ $1/1+\rho$ เป็นอัตราลดค่า (Discount rate) ที่สามารถกำหนดได้ว่าเราให้ความสำคัญต่อการได้คำตอบเร็วเข้าอย่างไร ถ้าการได้คำตอบเร็ว ๆ มีความสำคัญมาก ก็กำหนดให้อัตราการลดค่าสูง มูลค่าของคำตอบที่ได้ในเวลานานกว่าก็จะถูกลดค่าลงไปมากกว่าคำตอบที่ได้มาเร็ว ซึ่งอัตราลดค่านี้นี้ถูกคูณเข้ากับเทอม $\mathcal{E}[F(x')|x]$ ซึ่งเป็นค่าความคาดหวังของมูลค่าที่ได้ในช่วงเวลาต่อไป มูลค่าที่คำนวณได้นี้เรายังสามารถหักค่าใช้จ่ายที่เกิดขึ้นได้ด้วย โดยเพิ่มพจน์ที่เป็นค่าใช้จ่ายเข้าไป กำหนดให้ค่าใช้จ่ายที่

เกิดจากคำนวณค่าความเหมาะสมแต่ละครั้งคือ c ถ้าในแต่ละรุ่นมีประชากร N ตัว ค่าใช้จ่ายที่เกิดขึ้นในแต่ละรุ่นก็จะเท่ากับ cN ดังนั้น ถ้าจะให้ขั้นตอนวิธีเชิงพันธุกรรมดำเนินการต่อ จะต้องเสียค่าใช้จ่ายนี้ด้วย พจน์ที่ใช้คำนวณมูลค่าของการให้อัลกอริทึมทำงานต่อก็คือ

$$\frac{1}{1+\rho} \varepsilon[F(x') | x] - cN$$

เมื่อกำหนดวิธีคำนวณมูลค่าของการตัดสินใจหยุดการทำงานและมูลค่าของการดำเนินการต่อได้แล้ว การตัดสินใจที่เหมาะสมก็คือ เลือกค่าที่มากที่สุดระหว่าง 2 ค่าข้างต้น และ $F(x)$ ก็คือ มูลค่าอัลกอริทึมที่ได้รับจากการตัดสินใจนั้น ซึ่งคำนวณได้จากสมการที่ 3.2

$$F(x) = \max \left\{ g(x), \frac{1}{1+\rho} \varepsilon[F(x') | x] - cN \right\} \quad (3.2)$$

ในกรณีที่เราไม่พิจารณาเรื่องเวลาในการได้รับคำตอบ พจน์ $1/1+\rho$ ก็สามารถตัดออกได้ เช่นเดียวกับถ้าเราไม่คำนึงถึงต้นทุนในการคำนวณ พจน์ cN ก็จะถูกตัดออกเช่นกัน ในกรณีนี้ฟังก์ชันที่ใช้ในการคำนวณมูลค่าที่ได้จะเหลือเพียง

$$F(x) = \max \{ g(x), \varepsilon[F(x') | x] \} \quad (3.3)$$

จากสมการที่ 3.3 เราสนใจเพียงแค่ว่าความเหมาะสมที่ได้เทียบกับค่าความคาดหวังที่จะพบคำตอบที่ดีขึ้น ถ้ามีโอกาสที่จะพบคำตอบที่ดีขึ้น ขั้นตอนวิธีเชิงพันธุกรรมก็จะเลือกดำเนินการต่อ แต่ถ้าโอกาสที่จะพบคำตอบที่ดีขึ้นน้อยกว่ามูลค่าที่มีในปัจจุบันก็จะตัดสินใจหยุด สมการนี้แสดงให้เห็นวิธีการตัดสินใจหยุดการทำงานของอัลกอริทึมจากพฤติกรรมการหาคำตอบของแต่ละอัลกอริทึมเอง ซึ่งสามารถนำมาใช้วิเคราะห์ค่าคาดหวังจากการทำงานของอัลกอริทึมต่าง ๆ ได้ ดังจะแสดงตัวอย่างการนำมาวิเคราะห์ขั้นตอนวิธีประมาณการแจกแจงในบทถัดไป

3.3 การคำนวณมูลค่าอัลกอริทึม

จากสมการในหัวข้อ 3.2 มูลค่าอัลกอริทึมสามารถคำนวณได้โดยคิดมูลค่าในรุ่นสุดท้าย แล้วคำนวณย้อนกลับมาจนถึงรุ่นแรกสุด และ $F(x)$ ที่คำนวณได้ในรุ่นแรกสุดนี้คือมูลค่าของอัลกอริทึมเมื่อพิจารณาตามโอกาสในการหาคำตอบซึ่งอาจจะคำนึงถึงต้นทุนการคำนวณและมูลค่าของเวลาที่ได้รับคำตอบมาเร็วหรือช้า

ตัวอย่างการคำนวณมูลค่าอัลกอริทึมตามสมการที่ 3.3 แสดงในรูปที่ 3.2 ซึ่งในที่นี้จะแสดงตัวอย่างการคิดมูลค่าอัลกอริทึมจากการทำงาน 2 รุ่น สมมติให้ค่าความเหมาะสมในรุ่นแรกคือ 5 และค่าคำตอบในรุ่นถัดไปจะเป็น 7 และ 4 ด้วยความน่าจะเป็น 0.5

$$\begin{aligned} & \max\{5, (0.5 \cdot 7) + (0.5 \cdot 4)\} \\ & = \max\{5, 5.5\} = 5.5 \end{aligned}$$

รูปที่ 3.2 ตัวอย่างการคำนวณมูลค่าอัลกอริทึม

การคำนวณมูลค่าอัลกอริทึมจะเริ่มคำนวณจากรุ่นสุดท้าย ซึ่งในรุ่นสุดท้ายนี้มูลค่าจะเท่ากับค่าความเหมาะสม ณ เวลานั้นคือ 7 และ 4 เนื่องจากจะไม่มี การดำเนินการใด ๆ ต่อจากนี้อีกแล้ว การคำนวณมูลค่าอัลกอริทึมจะทำย้อนกลับไปในรุ่นก่อนหน้า ตัวอย่างในที่นี้มีเพียง 2 รุ่น ดังนั้นเมื่อคำนวณค่าในรุ่นที่ 2 เสร็จแล้วจึงคำนวณมูลค่าอัลกอริทึมในรุ่นที่ 1 ต่อไปมูลค่าการหยุดในรุ่นที่ 1 คือค่าความเหมาะสมในขณะนั้นซึ่งเท่ากับ 5 ในขณะที่มูลค่าในการดำเนินการต่อคำนวณได้จากผลรวมของค่าความเหมาะสมที่จะได้ในรุ่นถัดไปคูณกับความน่าจะเป็นของค่า นั้น ในตัวอย่างนี้คือ 0.5×7 บวกกับ 0.5×4 ซึ่งก็คือ 5.5 จากนั้นพิจารณาค่า $\max\{5, 5.5\}$ จะได้ว่ามูลค่าอัลกอริทึมคือ 5.5 และการตัดสินใจที่เหมาะสมในรุ่นที่ 1 ก็คือการตัดสินใจดำเนินการต่อ ผลการตัดสินใจรวมทั้งค่าของอัลกอริทึมที่ได้นี้จะถูกนำมาสร้างเป็นขอบเขตการใช้สิทธิ (Exercise region) ซึ่งจะกล่าวรายละเอียดในหัวข้อถัดไป

3.4 การสร้างขอบเขตการใช้สิทธิ

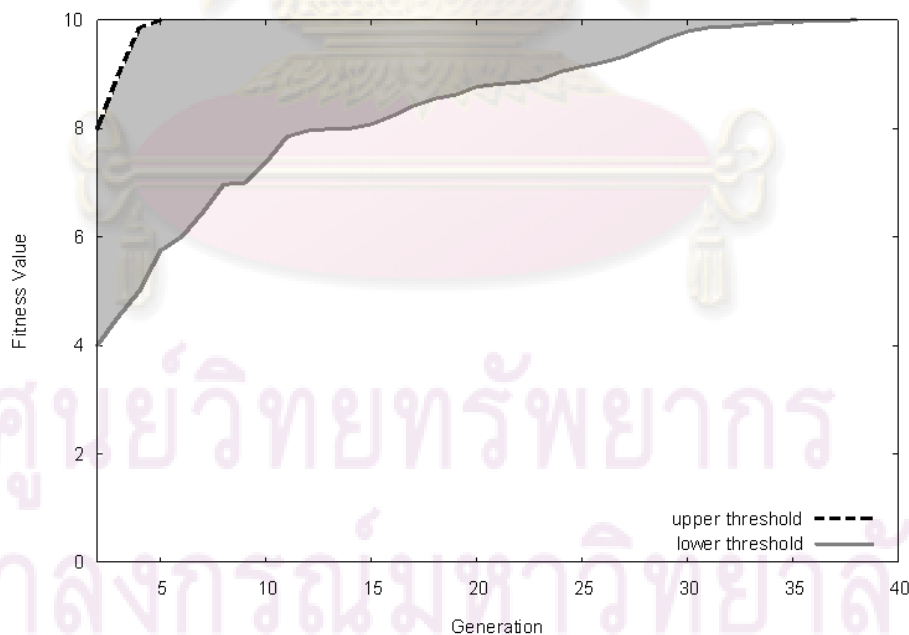
จากการคำนวณมูลค่าอัลกอริทึมตามหัวข้อที่ 3.3 เราจะได้ค่ามากที่สุดระหว่าง การตัดสินใจหยุดหรือดำเนินการต่อ ค่าความเหมาะสมที่เป็นเส้นแบ่งว่าควรตัดสินใจหยุดหรือดำเนินการต่อนี้เองจะสร้างขอบเขตที่บอกว่าเมื่อค่าความเหมาะสมตกอยู่ในด้านหนึ่งของเส้นแบ่งควรตัดสินใจหยุด และในทางกลับกันถ้าค่าความเหมาะสมอยู่ในอีกด้านหนึ่ง การตัดสินใจดำเนินการต่อเป็นการตัดสินใจที่เหมาะสม ขอบเขตการตัดสินใจนี้เรียกว่าขอบเขตการใช้สิทธิ

รูปที่ 3.3 แสดงตัวอย่างของมูลค่าอัลกอริทึมรวมทั้งผลการตัดสินใจที่ได้จากการคำนวณตามสมการที่ 3.3 ค่าในแถวตั้งแสดงมูลค่าอัลกอริทึมที่คำนวณได้ เรียงลำดับตามค่า

ความเหมาะสม 10 ลงมาจนถึง 0 ค่าที่อยู่ในวงเล็บคือ ค่าที่แสดงการตัดสินใจ ในที่นี้ค่า 1 แทนการตัดสินใจหยุด และ 0 แทนการตัดสินใจดำเนินการต่อ พื้นที่ส่วนที่แรเงาคือบริเวณที่เป็นการตัดสินใจดำเนินการต่อ และพื้นที่ที่อยู่นอกเหนือจากนั้นคือการตัดสินใจหยุด เส้นแบ่งระหว่างค่า 1 และ 0 จะสร้างขอบเขตการใช้สิทธิได้ดังรูปที่ 3.4

Bin (fitness value)	10.00(1)	10.00(1)	10.00(1)	10.00(1)	10.00(1)	10.00(1)	10.00(1)
	9.00(1)	9.00(1)	9.43(0)	9.30(0)	9.21(0)	9.03(0)	9.00(1)
	8.00(1)	9.97(0)	8.40(0)	8.40(0)	8.00(1)	8.00(1)	8.00(1)
	9.97(0)	9.97(0)	7.00(1)	7.00(1)	7.00(1)	7.00(1)	7.00(1)
	9.97(0)	9.97(0)	...	6.00(1)	6.00(1)	6.00(1)	6.00(1)
	9.97(0)	9.96(0)	5.00(1)	5.00(1)	5.00(1)	5.00(1)	5.00(1)
	9.96(0)	4.00(1)	4.00(1)	4.00(1)	4.00(1)	4.00(1)	4.00(1)
	3.00(1)	3.00(1)	3.00(1)	3.00(1)	3.00(1)	3.00(1)	3.00(1)
	2.00(1)	2.00(1)	2.00(1)	2.00(1)	2.00(1)	2.00(1)	2.00(1)
	1.00(1)	1.00(1)	1.00(1)	1.00(1)	1.00(1)	1.00(1)	1.00(1)
	0.00(1)	0.00(1)	0.00(1)	0.00(1)	0.00(1)	0.00(1)	0.00(1)

รูปที่ 3.3 ตัวอย่างผลการคำนวณมูลค่าอัลกอริทึมและผลการตัดสินใจ



รูปที่ 3.4 ตัวอย่างรูปขอบเขตการใช้สิทธิ

จากรูปที่ 3.4 เส้นด้านบนเป็นเส้นแบ่งที่การตัดสินใจหยุดอยู่ด้านบน และการตัดสินใจดำเนินการต่ออยู่ด้านล่าง เราจะเรียกเส้นแบ่งนี้ว่า “เส้นแบ่งด้านบน” (Upper threshold) เส้นแบ่งที่เกิดจากการตัดสินใจดำเนินการต่ออยู่ด้านบนและการตัดสินใจหยุดอยู่ด้านล่าง เราจะเรียกว่า “เส้นแบ่งด้านล่าง” (Lower threshold)

เส้นแบ่ง 2 เส้นนี้ ทำให้เกิดพื้นที่ขึ้นมา 3 ส่วน พื้นที่ด้านบนเส้นแบ่งด้านบนและพื้นที่ใต้เส้นแบ่งด้านล่างเรียกว่าอาณาเขตการหยุด (Stopping region) ส่วนพื้นที่ตรงกลางระหว่าง 2 เส้นนี้ที่แรเงาเป็นสีเทาคือ อาณาเขตดำเนินการต่อ (Continuation region) ขอบเขตการใช้สิทธินี้แสดงให้เห็นว่าอัลกอริทึมควรหยุดการทำงานเมื่อค่าความเหมาะสมตกอยู่ในอาณาเขตการหยุด และควรทำงานต่อไปถ้าค่าความเหมาะสมยังอยู่ในอาณาเขตดำเนินการต่อ

ขั้นตอนวิธีของการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมดังที่ได้กล่าวมาข้างต้นแสดงเป็นรหัสเทียมได้ในรูปที่ 3.5 และรหัสเทียมของการคำนวณมูลค่าอัลกอริทึมแสดงในรูปที่ 3.6

การทำงานของขั้นตอนวิธีเชิงพันธุกรรมในรูปที่ 3.5 จะเหมือนเดิมทุกประการ เพียงแต่เพิ่มการเก็บการเปลี่ยนแปลงของค่าความเหมาะสมในแต่ละรุ่นเอาไว้ โดยกำหนดจำนวนรุ่นสูงสุดเท่ากับ MAXGEN และจำนวนรอบในการทำงานซ้ำกำหนดด้วยตัวแปร MAXRUN เมื่อขั้นตอนวิธีเชิงพันธุกรรมทำงานเสร็จแล้วจึงทำการคำนวณมูลค่าอัลกอริทึมในรูปที่ 3.6

ในขั้นตอนวิธีคำนวณมูลค่าอัลกอริทึม ได้มีการกำหนดตัวแปรแถวลำดับขนาด 2 มิติ สำหรับการคำนวณมูลค่าอัลกอริทึม (algorithm_value) ผลการตัดสินใจ (decision) เส้นแบ่งด้านบน (upper_threshold) และเส้นแบ่งด้านล่าง (lower_threshold) ของรุ่นต่าง ๆ ที่ค่าความเหมาะสมใด ๆ โดยที่ MAXBIN คือ ขนาดของจำนวนช่องที่ใช้แบ่งค่าความเหมาะสมที่เป็นได้ออกเป็นช่วงต่าง ๆ ซึ่งในที่นี้ใช้ช่วงเท่ากับ 1 นั่นคือค่าความเหมาะสมที่เป็นจำนวนเต็ม

```

Procedure constructing the optimal stopping policy of genetic algorithm
1: Initial a set of matrix for collecting a fitness movement of each generation
2: for  $r = 1$  to MAXRUN do
3:   Initialize population
4:   for  $g = 1$  to MAXGEN do
5:     selection
6:     perform genetic operators
7:     evaluate population
8:     update profile according to a fitness movement from generation  $g-1$  to  $g$ 
9:   end for
10: end for
11: real_optimons_calculation();

```

รูปที่ 3.5 ขั้นตอนวิธีการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรม

```

Procedure real_options_calculation
1: for  $g = \text{MAXGEN}$  downto  $g = 1$  do
2:   for  $bin = \text{MAXBIN}$  downto  $bin = 1$  do
3:     (Note that the algorithm value is calculated using equation (3.3))
     algorithm_value[ $g$ ][ $bin$ ] =  $\max(g(x), \varepsilon[F(x') | x])$ ;
4:     if  $\varepsilon[F(x') | x] > g(x)$  then (decide to continue)
5:       decision[ $g$ ][ $bin$ ] = 0;
6:     else
7:       decision[ $g$ ][ $bin$ ] = 1;
8:     (Form a boundary of preference fitness value in each generation)
     if decision[ $g$ ][ $bin$ ] = 1 and decision[ $g$ ][ $bin-1$ ] = 0 then
9:       upper_threshold[ $g$ ] = fitness_value_at  $bin$ ;
10:    else if decision[ $g$ ][ $bin$ ] = 0 and decision[ $g$ ][ $bin-1$ ] = 1 then
11:      lower_threshold[ $g$ ] = fitness_value_at  $bin$ ;
12:    endfor
13:  endfor

```

รูปที่ 3.6 ขั้นตอนวิธีในการคำนวณมูลค่าอัลกอริทึม

การคำนวณมูลค่าอัลกอริทึมตามขั้นตอนวิธีในบรรทัดที่ 3 ของรูปที่ 3.6 คำนวณได้จากสมการที่ 3.3 ซึ่งเป็นการหาค่ามากที่สุดระหว่างมูลค่าการหยุดและมูลค่าการดำเนินการต่อซึ่งมูลค่าการหยุด $g(x)$ ก็คือค่าความเหมาะสมในขณะนั้น ส่วนมูลค่าการดำเนินการต่อก็คือค่าคาดหวังของมูลค่าอัลกอริทึมในเวลาถัดไป ซึ่งคำนวณจากผลคูณของค่าความน่าจะเป็นที่ค่าความเหมาะสม x จะเปลี่ยนเป็น x' กับมูลค่าอัลกอริทึมในขณะที่ค่าความเหมาะสมมีค่าเป็น x'

จากขั้นตอนวิธีดังกล่าว เราได้ใช้ข้อมูลทั้งหมดจากการทำงานของขั้นตอนวิธีเชิงพันธุกรรมในหลาย ๆ รอบ มาหาเวลาหยุดเหมาะสมที่สุด วิธีนี้อาจไม่เหมาะที่จะนำไปใช้งานโดยตรงเนื่องจากในความเป็นจริงเราไม่สามารถทำงานจนเสร็จก่อนแล้วค่อยมาหาวิธีการตัดสินใจ แต่เราจะนำผลที่ได้ไปใช้วิเคราะห์การทำงานของขั้นตอนวิธีเชิงพันธุกรรม ในการแก้ปัญหาประเภทต่าง ๆ ในบทที่ 4 และวิทยานิพนธ์นี้นำเสนอการดัดแปลงวิธีนี้โดยนำความรู้ของเวลาหยุดเหมาะสมที่สุดในรอบก่อนหน้าไปใช้กับการทำงานรอบต่อไป ซึ่งเงื่อนไขการหยุดนี้ก็เลยค่อย ๆ ถูกปรับปรุงเมื่อมีการทำงานรอบถัดไป ซึ่งจะทำให้อัลกอริทึมไม่ต้องดำเนินงานหลายรอบก่อนที่จะหานโยบายการหยุด โดยรายละเอียดของวิธีการนี้จะนำเสนอในบทที่ 6

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4

การวิเคราะห์เวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีประมาณการแจกแจง

ในบทนี้จะนำวิธีอุปชันจริงมาใช้ในการหาเวลาหยุดเหมาะสมที่สุดของขั้นตอนวิธีประมาณการแจกแจง พร้อมทั้งวิเคราะห์มูลค่าของแต่ละอัลกอริทึมเมื่อรวมการเลือกหยุด โดยในที่นี้จะวิเคราะห์ขั้นตอนวิธีประมาณการแจกแจงแบบที่ง่ายที่สุด คือ ขั้นตอนวิธีประมาณการแจกแจงแบบที่ตัวแปรไม่ขึ้นต่อกัน ซึ่งได้แก่ การเรียนรู้เพิ่มขึ้นแบบอาศัยประชากร (PBIL) ขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปร (UMDA) และขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับ (cGA)

การวิเคราะห์จะแบ่งเป็น 2 ส่วนคือ การใช้จำนวนประชากรแบบน้อยที่สุดเท่าที่จะเป็นไปได้คือ 2 ตัว และการใช้กลุ่มประชากรที่มากขึ้น การที่ต้องแบ่งการทดลองออกเป็น 2 ส่วนเนื่องจากขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับเดิมได้ถูกออกแบบมาเพื่อใช้กับประชากรจำนวน 2 ตัว ซึ่งเป็นจำนวนประชากรน้อยสุดที่เป็นไปได้ การใช้ประชากรที่มากขึ้นถูกพัฒนาขึ้นในภายหลัง ดังนั้น เพื่อศึกษาถึงพฤติกรรมการทำงานของขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับทั้งแบบดั้งเดิมและแบบที่ถูกพัฒนาขึ้นภายหลัง จึงได้ออกแบบการทดลองออกเป็น 2 ส่วนดังที่ได้กล่าวมา

ในการทดลองโดยใช้ประชากร 2 ตัว ซึ่งจะกล่าวในหัวข้อ 4.1 จะทำการเปรียบเทียบวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรกับขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับเท่านั้น ส่วนขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปรจะไม่ถูกนำมากล่าวถึงในส่วนนี้ เนื่องจากขั้นตอนวิธีดังกล่าวจำเป็นต้องใช้ประชากรมากกว่า 2 ตัวในการปรับปรุงเวกเตอร์ความน่าจะเป็นของการกระจายตัวของคำตอบ และเพื่อเปรียบเทียบทั้ง 3 อัลกอริทึมข้างต้น การทดลองโดยใช้ประชากรที่มากขึ้นจะเสนอในหัวข้อ 4.2 ต่อไป

4.1 การวิเคราะห์ขั้นตอนวิธีประมาณการแจกแจงโดยใช้จำนวนประชากร 2 ตัว

ในการทดลองมีการกำหนดค่าพารามิเตอร์ดังนี้ ทั้ง 2 อัลกอริทึมมีจำนวนประชากร 2 ตัว อัตราการเรียนรู้ (α) ในวิธีเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรถูกกำหนดเป็น 0.05 และอัตราการเปลี่ยนแปลงค่าในเวกเตอร์ความน่าจะเป็นของขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับ (Updating step size) คือ 0.02

วิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรและขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับจะทดสอบด้วยปัญหาจำนวนหนึ่งมากที่สุด 30 บิต (Onemax 30) ปัญหาที่บิตขนาด 3 ต่อกัน

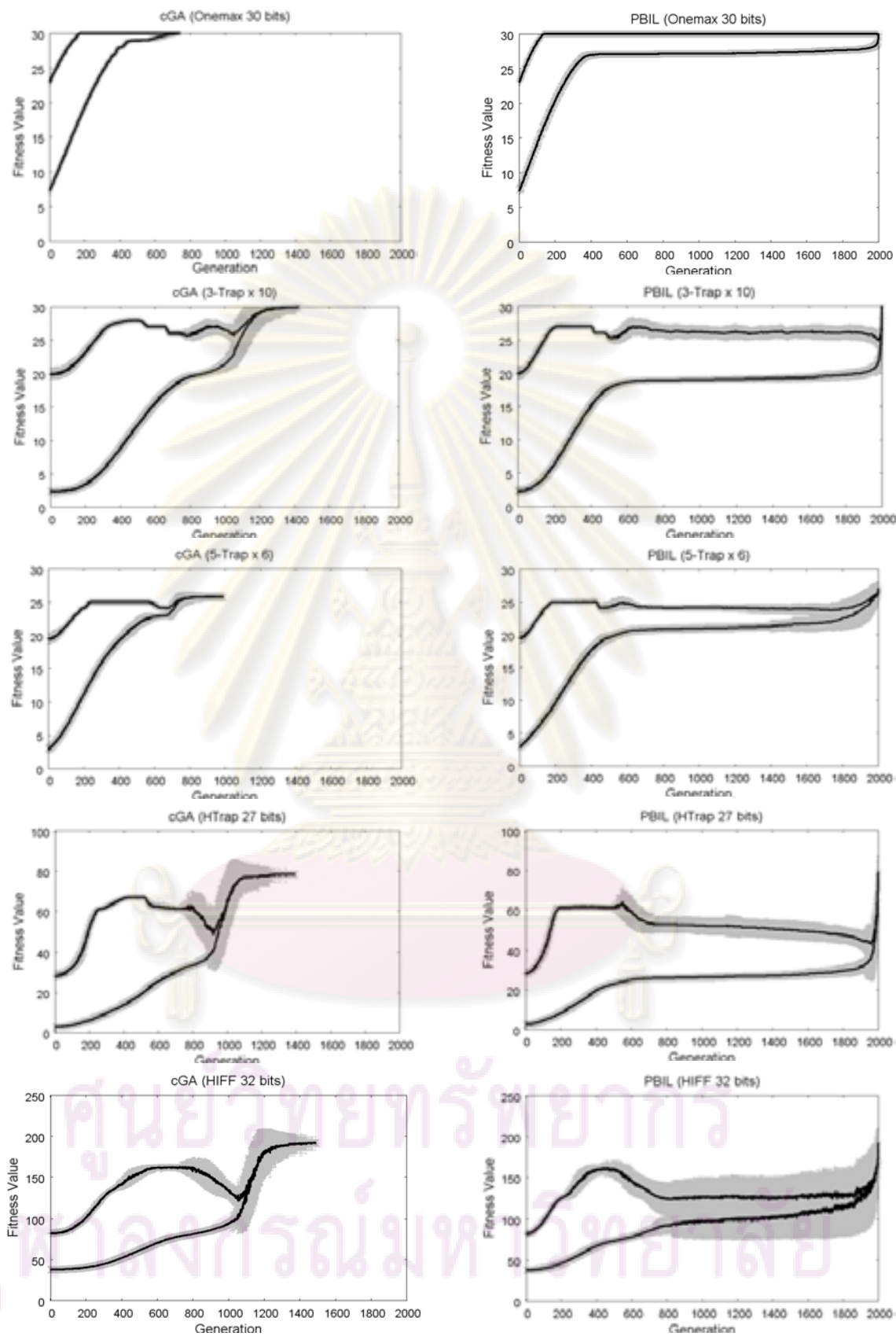
10 ชุด (3-Trap x 10) ปัญหาภัยดับขนาด 5 ต่อกัน 6 ชุด (5-Trap x 6) ปัญหาภัยดับแบบลำดับชั้นขนาด 27 บิต (HTrap 27) และปัญหาลำดับชั้นแบบกึ่งต่อเนื่องขนาด 32 บิต (HIFF 32) โดยรายละเอียดของแต่ละปัญหาทดสอบได้กล่าวไว้ในบทที่ 2

แต่ละอัลกอริทึมจะทำงานทั้งสิ้น 1,000 รอบ และจะเก็บข้อมูลการทำงานนี้ไว้สำหรับคำนวณเวลาหยุดเหมาะสมที่สุดและหามูลค่าอัลกอริทึมจากอัลกอริทึมอื่น ๆ ขอบเขตการตัดสินใจที่ได้ ดังที่จะนำเสนอต่อไป มาจากค่าเฉลี่ยของการคำนวณมูลค่าอัลกอริทึม 100 ครั้ง ในที่นี้หมายความว่า ในแต่ละครั้งของการคำนวณมูลค่าอัลกอริทึม อัลกอริทึมจะเริ่มต้นทำงานใหม่อีก 1,000 รอบ แล้วนำข้อมูลนี้มาคำนวณหาค่าเฉลี่ยของการตัดสินใจ

ขอบเขตการใช้สิทธิที่ได้จากการคำนวณมูลค่าอัลกอริทึมตามสมการที่ 3.3 แสดงในรูปที่ 4.1 ในรูปดังกล่าว กราฟจะถูกนำเสนอเป็น 2 สดมภ์ สดมภ์แรกเป็นขอบเขตการใช้สิทธิของขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับ ส่วนสดมภ์ที่ 2 เป็นขอบเขตการใช้สิทธิของวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากร ในแต่ละแถวของอัลกอริทึมทั้ง 2 แสดงขอบเขตการใช้สิทธิที่แตกต่างกันไปในแต่ละปัญหาทดสอบ ซึ่งกราฟของปัญหาทดสอบต่าง ๆ เรียงจากบนลงล่างคือ ปัญหาจำนวนหนึ่งมากที่สุดขนาด 30 บิต (Onemax 30) ปัญหาภัยดับขนาด 3 ต่อกัน 10 ชุด (3-Trap x 10) ปัญหาภัยดับขนาด 5 ต่อกัน 6 ชุด (5-Trap x 6) ปัญหาภัยดับแบบลำดับชั้นขนาด 27 บิต (HTrap 27) และปัญหาลำดับชั้นแบบกึ่งต่อเนื่องขนาด 32 บิต (HIFF 32)

ดังที่ได้กล่าวไว้ในบทที่ 3 รูปกราฟขอบเขตการใช้สิทธิประกอบด้วยเส้นแบ่ง 2 เส้นที่แบ่งพื้นที่เป็น 3 ส่วนคือ พื้นที่ส่วนบนเส้นแบ่งด้านบนและพื้นที่ใต้เส้นแบ่งด้านล่างเป็นอาณาเขตการหยุด และพื้นที่ตรงกลางระหว่างเส้นแบ่งทั้งสองเป็นอาณาเขตดำเนินการต่อ ขอบเขตนี้ เป็นสิ่งที่ใช้บอกว่าการหยุดตัดสินใจเช่นไรเมื่อค่าความเหมาะสมในรุ่นนั้น ๆ ตกอยู่ในพื้นที่แต่ละบริเวณ อัลกอริทึมควรหยุดถ้าค่าความเหมาะสมตกอยู่ในอาณาเขตการหยุด และควรดำเนินการต่อไปถ้าค่าความเหมาะสมตกอยู่ในอาณาเขตดำเนินการต่อ บริเวณที่เป็นสีเทาในรูปที่ 4.1 คือค่าเบี่ยงเบนมาตรฐานของเส้นแบ่งด้านบนและด้านล่าง ซึ่งคำนวณจากค่าเฉลี่ยของการคำนวณมูลค่าอัลกอริทึม 100 ครั้ง

จากกราฟรูปบนสุดจะสังเกตเห็นได้ว่า สำหรับปัญหาจำนวนหนึ่งมากที่สุดซึ่งเป็นปัญหาง่าย ขอบเขตการใช้สิทธิของขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับกับขอบเขตการใช้สิทธิของวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรมีลักษณะคล้ายกันมาก ต่างกันตรงที่เส้นแบ่งด้านล่างของวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรจะขนานไปกับเส้นแบ่งด้านบน จนกระทั่งถึงรุ่นสุดท้าย ลักษณะเช่นนี้แสดงให้เห็นว่าในช่วงรุ่นท้าย ๆ ของกระบวนการวิวัฒนาการ วิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรเปิดโอกาสให้ประชากรที่มีค่าความเหมาะสมต่ำมีวิวัฒนาการต่อมากกว่าขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับซึ่งเส้นแบ่งด้านล่างแตะขอบบนและตัดสินใจหยุดเร็วกว่า



รูปที่ 4.1 ขอบเขตการใช้สถิติจากการวิเคราะห์ด้วยประชากร 2 ตัว เส้นสีดำในกราฟคือเส้นแบ่งด้านบนและด้านล่าง ส่วนบริเวณที่เป็นสีเทาคือค่าเบี่ยงเบนมาตรฐานของเส้นแบ่งทั้งสอง

สำหรับปัญหากับดักขนาด 3 ต่อกัน 10 ชุด และปัญหากับดักขนาด 5 ต่อกัน 6 ชุด ทั้งสองปัญหานี้มีค่าความเป็นไปได้ของค่าความเหมาะสมเท่ากับปัญหาคำนวณหนึ่งมากที่สุดคือ ช่วง $[0, 30]$ แต่ปัญหากับดักนี้ยากกว่า เนื่องจากฟังก์ชันในการคำนวณค่าความเหมาะสมถูกออกแบบมาเพื่อหลอกให้อัลกอริทึมค้นคำตอบในด้านตรงกันข้าม ลักษณะของขอบเขตการใช้สิทธิในปัญหานี้เมื่อเทียบกับปัญหาคำนวณหนึ่งมากที่สุดจะพบว่า ในช่วงเริ่มต้น ค่าความเหมาะสมเริ่มต้นของปัญหาคำนวณหนึ่งมากที่สุดจะสูงกว่าปัญหากับดัก เนื่องจากตัวอย่างดี ๆ ที่ให้ค่าความเหมาะสมสูง ๆ ในปัญหาคำนวณหนึ่งมากที่สุดจะพบได้ง่ายกว่าปัญหากับดัก เช่นเดียวกันกับปัญหาคำนวณหนึ่งมากที่สุดยังมีพื้นที่การหยุดได้เส้นแบ่งด้านล่างที่ใหญ่กว่าปัญหากับดักด้วย ทั้งนี้เนื่องจากในปัญหาง่าย ถ้าประชากรไม่สามารถเพิ่มค่าความเหมาะสมได้รวดเร็วเพียงพอ อัลกอริทึมก็ไม่ควรที่จะดำเนินการต่อ การหยุดแล้วไปเริ่มต้นใหม่กับประชากรตั้งต้นที่คุณภาพดีกว่าอาจเป็นทางเลือกที่เหมาะสมกว่าก็เป็นได้

จากลักษณะของเส้นแบ่งด้านบน สังเกตได้ว่ามีลักษณะเป็น 2 แบบ คือ ในปัญหาคำนวณหนึ่งมากที่สุด ค่าของเส้นแบ่งด้านบนจะค่อย ๆ เพิ่มขึ้นอย่างต่อเนื่องจนถึงค่าสูงสุด แต่ในปัญหาอื่น ๆ ที่เป็นปัญหายาก เส้นแบ่งด้านบนจะมีลักษณะไม่เรียบ จะมีรอยหยักโค้งขึ้นลงในบางช่วง ซึ่งรอยหยักนี้แสดงให้เห็นถึงความไม่แน่นอนในการหาคำตอบที่ดี ๆ ในปัญหายาก ลักษณะเช่นนี้คล้ายกับพฤติกรรมที่ He et al. (2003) ได้กล่าวไว้ว่า ปัจจัยหนึ่งที่เป็นเงื่อนไขที่ทำให้ปัญหานั้น ๆ ยากสำหรับขั้นตอนวิธีเชิงวิวัฒนาการคือช่องว่างที่กว้าง (Wide gap) ซึ่งช่องว่างที่กว้างในที่นี้หมายถึงเหตุการณ์ที่ความน่าจะเป็นในการได้คำตอบที่ดีขึ้นมีโอกาสน้อยมาก ๆ เหตุการณ์นี้เป็นเหตุผลว่าทำไมเส้นแบ่งด้านบนสำหรับปัญหายากจึงมีรอยหยัก นั่นเป็นเพราะเมื่อผ่านกระบวนการวิวัฒนาการไปแล้วช่วงหนึ่ง ในปัญหายากโอกาสที่เราจะพบคำตอบที่คุณภาพดีกว่าเดิมนั้นมีน้อย บางทีดูเหมือนมีโอกาสเพิ่มขึ้น บางทีก็ได้คำตอบที่ไม่ดี ทำให้ดูเหมือนมีโอกาสลดลง โดยเฉพาะอย่างยิ่งเมื่ออัลกอริทึมถูกหลอกให้หาคำตอบในทางที่เป็นค่าเหมาะสมเฉพาะที่ (Local optimal) อย่างเช่นปัญหากับดัก

จากลักษณะของพื้นที่ดำเนินการต่อของทั้ง 2 อัลกอริทึม ยังให้ข้อสังเกตที่น่าสนใจว่า วิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรนั้นมีพื้นที่ในส่วนของการดำเนินการต่อยาวไปจนถึงรุ่นหลัง ๆ ของกระบวนการวิวัฒนาการ ต่างจากขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับที่เส้นแบ่งด้านบนและด้านล่างบรรจบกันเร็วกว่า ลักษณะเช่นนี้แสดงพฤติกรรมการทำงานที่ต่างกันของทั้ง 2 อัลกอริทึม วิธีการเรียนรู้แบบอาศัยประชากรยอมให้ประชากรมีการปรับปรุงคุณภาพและวิวัฒนาการต่อไปจนหมดเวลา ในขณะที่ขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับเลือกที่จะหยุดทำงานรวดเร็วกว่า

การวิเคราะห์ความสามารถของอัลกอริทึม จะทำการวิเคราะห์ในแง่ของค่าความคาดหวังของค่าความเหมาะสมที่จะได้รับ เมื่อทำการตัดสินใจที่เหมาะสมที่สุด ซึ่งพิจารณาจากสมการคำนวณมูลค่าอัลกอริทึมที่ได้กล่าวไว้ในบทที่ 3 ค่าที่นำมาใช้เปรียบเทียบอัลกอริทึม

ต่าง ๆ ในที่นี้คือมูลค่าอัลกอริทึม และเพื่อให้สามารถเปรียบเทียบความสามารถของอัลกอริทึมหนึ่ง ๆ ในการแก้ปัญหาที่ต่างกัน งานวิจัยนี้ได้เสนอมาตรวัดที่ใช้เปรียบเทียบมูลค่าอัลกอริทึมของปัญหาต่าง ๆ ที่ใช้อัลกอริทึมเดียวกันในการแก้ปัญหา การเปรียบเทียบนี้จะใช้สัดส่วนของมูลค่าอัลกอริทึมที่คำนวณได้เทียบกับค่าเหมาะสมสุด (Global optimal) เพื่อให้รู้ว่ามูลค่าอัลกอริทึมที่ได้มีความใกล้เคียงกับคำตอบที่ดีที่สุดเพียงใด และในกรณีที่เราไม่ทราบว่าคำตอบที่ดีที่สุดเป็นเท่าใด จะใช้คำตอบที่ดีที่สุดที่หาได้เป็นตัวเปรียบเทียบแทน มาตรวัดที่นำเสนอนี้เรียกว่า ดรรชนีความยากของปัญหา (Difficulty index) ซึ่งมีนิยามดังนี้

กำหนดให้ f_i และ f_j เป็นมูลค่าจากอัลกอริทึมเดียวกัน ซึ่งคำนวณได้จากการทำงานกับปัญหา i และ j ให้ค่า F_i และ F_j เป็นค่าคำตอบที่ดีที่สุดของแต่ละปัญหา (หรือเป็นค่าคำตอบดีที่สุดที่เราทราบ) ดรรชนีความยากของปัญหาในการแก้ปัญหา i จะคำนวณได้จาก $1 - (f_i / F_i)$ และจะกล่าวว่าปัญหา i ยากกว่าปัญหา j ถ้า $1 - (f_i / F_i) > 1 - (f_j / F_j)$

มูลค่าอัลกอริทึมของการวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรและขั้นตอนวิธีเชิงพันธุกรรมแบบกระชั้นจากการทำงานกับปัญหาทดสอบต่าง ๆ รวมทั้งค่าดรรชนีความยากในแต่ละปัญหาแสดงในตารางที่ 4.1

ตารางที่ 4.1 มูลค่าอัลกอริทึมและค่าดรรชนีความยากของขั้นตอนวิธีเชิงพันธุกรรมแบบกระชั้นและวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากร

Algorithms Problems	cGA		PBIL	
	Algorithm value (f)	Difficulty Index $1 - (f/F)$	Algorithm value (f)	Difficulty Index $1 - (f/F)$
30 bits OneMax ($F=30$)	30.00	0.00	30.00	0.00
3-Trap x 10 ($F=30$)	27.00	0.10	26.53	0.12
5-Trap x 6 ($F=30$)	24.03	0.20	24.70	0.18
27 bits HTrap ($F=81$)	77.74	0.04	61.53	0.24
32 bits HIFF ($F=192$)	138.34	0.28	126.01	0.34

จากตารางที่ 4.1 ก่อนอื่นเราจะพิจารณาที่มูลค่าอัลกอริทึม ซึ่งมูลค่าอัลกอริทึมยิ่งสูงก็ยิ่งดี เนื่องจากค่าคาดหวังของคำตอบเมื่ออัลกอริทึมทำงานด้วยการตัดสินใจที่ดีที่สุดแล้วมีค่าสูง จากตารางจะเห็นได้ว่าส่วนใหญ่มูลค่าอัลกอริทึมของขั้นตอนวิธีเชิงพันธุกรรมแบบกระชั้นมีค่าสูงกว่ามูลค่าอัลกอริทึมจากวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากร ยกเว้นในกรณีของปัญหากับดักขนาด 5 ซึ่งวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรมีค่าสูงกว่าเล็กน้อย ที่เป็นเช่นนี้อาจจะเป็นเพราะวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรเปิดโอกาสให้มีการพัฒนาคุณภาพคำตอบต่อไปได้ ประชากรที่มีคุณภาพไม่สูงมากนักจึงมีโอกาสอดในการวิวัฒนาการในรุ่นหลัง ๆ ซึ่งอาจพบคำตอบที่มีคุณภาพดีขึ้นได้

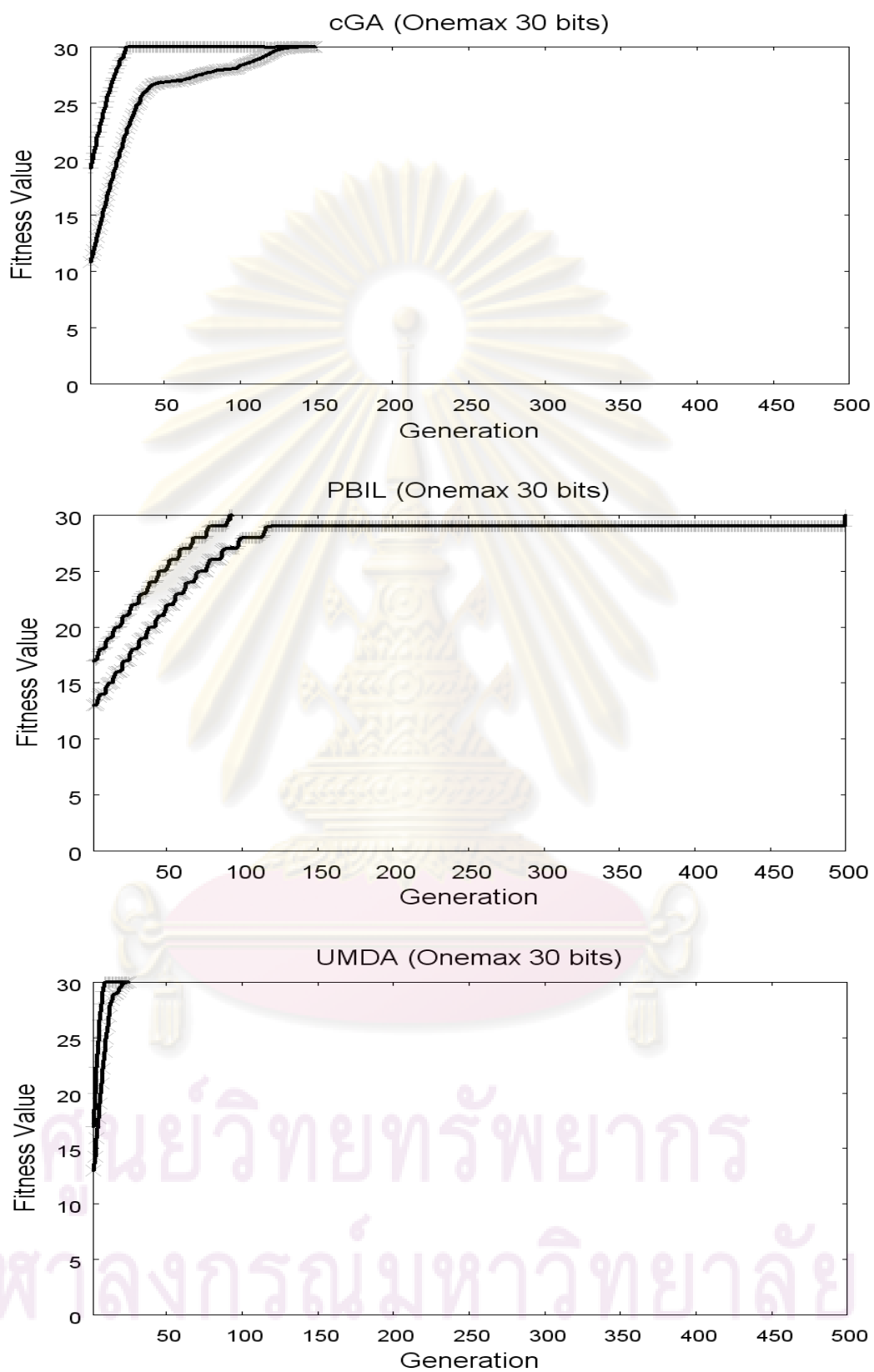
จากการพิจารณาค่าตรรกษณ์ความยาก พบว่าทั้งในขั้นตอนวิธีเชิงพันธุกรรมแบบ กระชับและวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรมีความสามารถในการหาคำตอบสำหรับ ปัญหาจำนวนหนึ่งมากที่สุดได้เท่ากัน นั่นคือทั้งคู่มีตรรกษณ์ความยากเท่ากับ 0 ซึ่งหมายความว่า สามารถหาคำตอบที่ดีที่สุด (หรือดีที่สุดเท่าที่เรารู้) ได้ทั้งคู่ และจากทั้ง 2 อัลกอริทึมจะเห็นได้ว่า ปัญหาลำดับชั้นแบบก็ต่อเมื่อ (HIFF) เป็นปัญหาที่ยากที่สุด เนื่องจากมีค่าสัดส่วนตรรกษณ์ความ ยากของปัญหามากที่สุดเมื่อเทียบกับปัญหาอื่น เป็นที่ทราบกันคืออยู่แล้วว่าปัญหาลำดับชั้นนั้น ยาก จึงไม่แปลกที่ทั้ง 2 อัลกอริทึมนี้แก้ปัญหานี้ได้ไม่คืนัก

4.2 การวิเคราะห์ขั้นตอนวิธีประมาณการแจกแจงโดยใช้จำนวนประชากรที่มากขึ้น

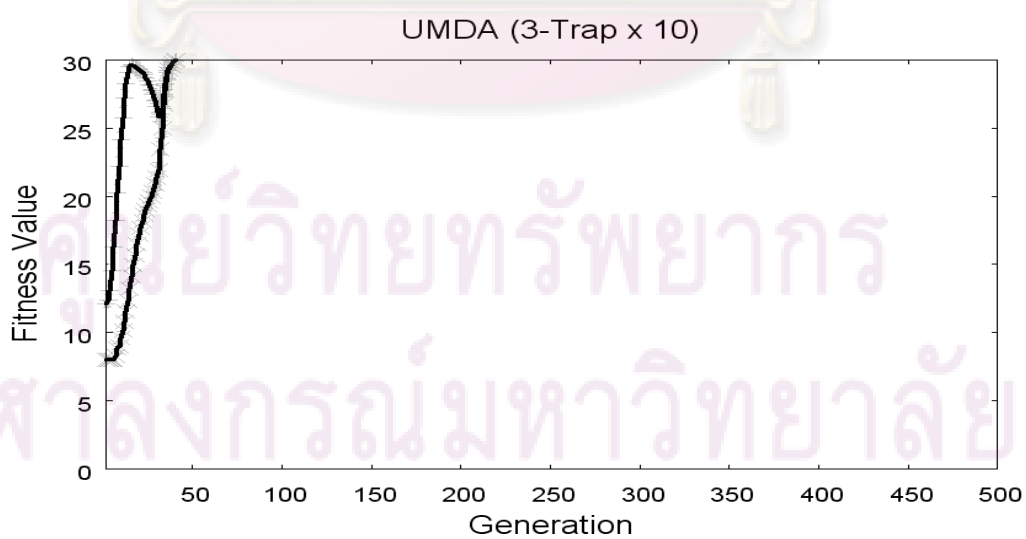
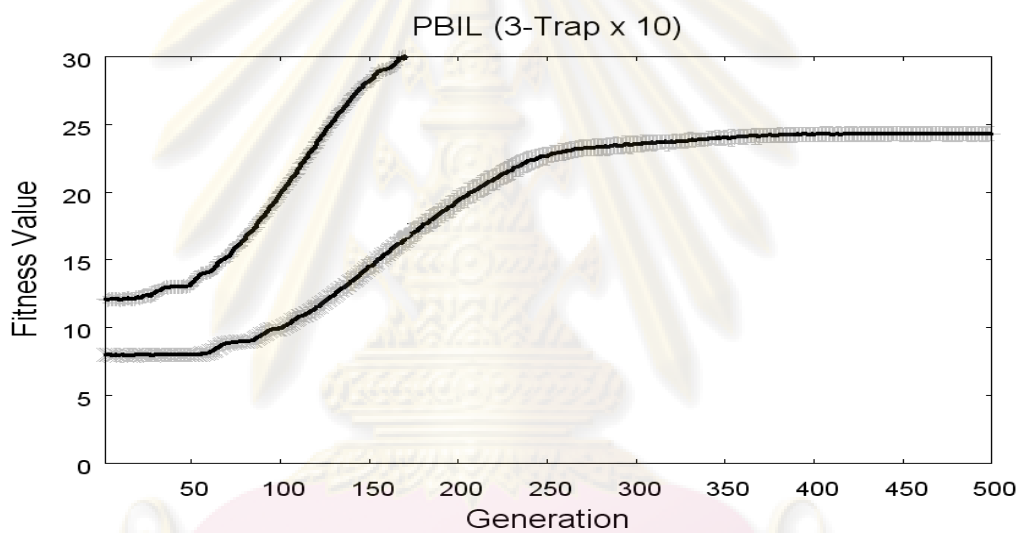
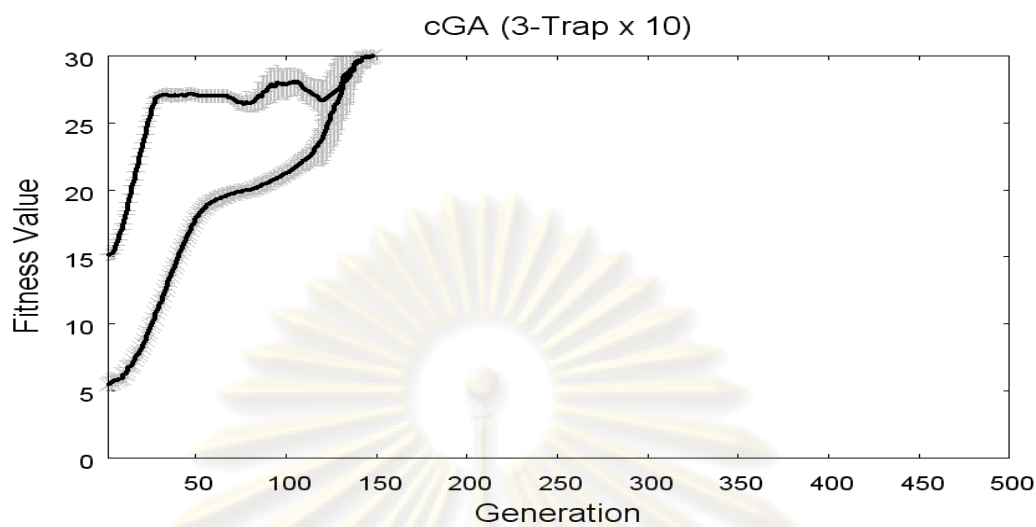
ในหัวข้อนี้จะทำการวิเคราะห์ขั้นตอนวิธีประมาณการแจกแจงแบบที่ตัวแปรไม่ขึ้น ต่อกันทั้ง 3 วิธี ที่ได้กล่าวมาข้างต้น โดยใช้จำนวนประชากรที่มากขึ้นจากหัวข้อที่แล้ว ทั้งนี้ เนื่องจากขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปรเป็นวิธีที่ต้องอาศัยประชากรมากกว่า 2 ตัวใน การประมาณค่าเวกเตอร์ความน่าจะเป็น การทดลองในส่วนนี้จึงใช้ประชากรมากขึ้นและจะทำการ เปรียบเทียบมูลค่าอัลกอริทึมที่ได้จากทั้ง 3 ขั้นตอนวิธี

จำนวนประชากรที่ใช้ในการทดลองในหัวข้อนี้กำหนดไว้เท่ากับ 50 และอัตราการ แข่งขันเพื่อการอยู่รอด (Tournament size) คือ 8 ผลจากการทดลองทำให้ได้ขอบเขตการใช้ สิทธิของทั้ง 3 อัลกอริทึมในปัญหาทดสอบต่าง ๆ ดังรูปที่ 4.2 – 4.6 เส้นสีดำในกราฟดังกล่าวคือ เส้นแบ่งด้านบนและด้านล่าง และบริเวณที่เป็นสีเทาคือค่าเบี่ยงเบนมาตรฐานของเส้นแบ่งทั้งสอง จากการคำนวณมูลค่าอัลกอริทึม 100 รอบ

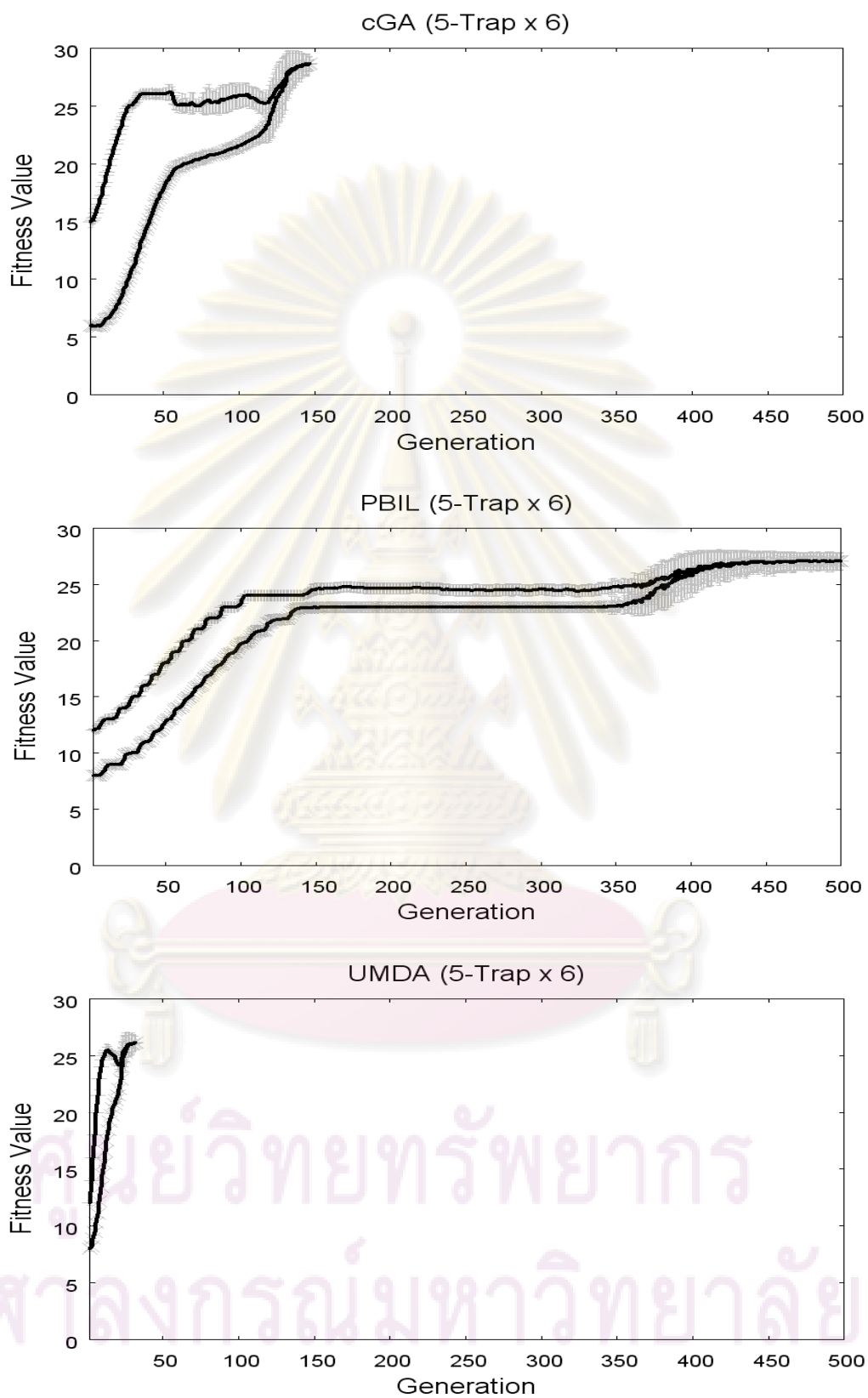
จากการสังเกตขอบเขตการใช้สิทธิของทั้ง 3 อัลกอริทึมนี้ทำให้เห็นลักษณะเฉพาะ ของแต่ละขั้นตอนวิธี สังเกตได้ว่าขอบเขตการใช้สิทธิของขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับ เส้นแบ่งด้านบนและด้านล่างจะบรรจบกันในตอนต้นของกระบวนการวิวัฒนาการและส่งผลให้ อัลกอริทึมหยุดการทำงานเสมอเมื่อเส้นแบ่งทั้งสองมาบรรจบกัน ในวิธีการเรียนรู้เพิ่มขึ้นแบบ อาศัยประชากร เส้นแบ่งด้านบนและด้านล่างมีลักษณะเด่นคือจะขนานกันไปจนถึงช่วงสุดท้าย ของวิวัฒนาการ แสดงว่าอัลกอริทึมมักจะค่อย ๆ ค้นพบคำตอบที่ดีขึ้น และในส่วนของขั้นตอน วิธีแจกแจงตามขอบหนึ่งตัวแปร จะพบว่าขอบเขตการใช้สิทธิมีลักษณะคล้ายกับขั้นตอนวิธีเชิง พันธุกรรมแบบกระชับ แต่เส้นแบ่งทั้ง 2 จะบรรจบกันเร็วกว่า แสดงว่าอัลกอริทึมนี้จะค้นพบ คำตอบได้รวดเร็วกว่าหรือมีฉะนั้นถ้าเวลาผ่านไปช่วงหนึ่งแล้วก็มักจะไม่น่าพบคำตอบที่ดีขึ้นเลย ลักษณะเด่นของขอบเขตการใช้สิทธิของแต่ละอัลกอริทึมแสดงให้เห็นว่าแต่ละขั้นตอนวิธีมีวิธีการ ตัดสินใจหยุดและมีวิธีการค้นหาคำตอบที่ค่อนข้างเป็นเอกลักษณ์ จากลักษณะเด่นดังกล่าวทำให้ เราสามารถสังเกตลักษณะพฤติกรรมของอัลกอริทึมได้จากขอบเขตการใช้สิทธิ



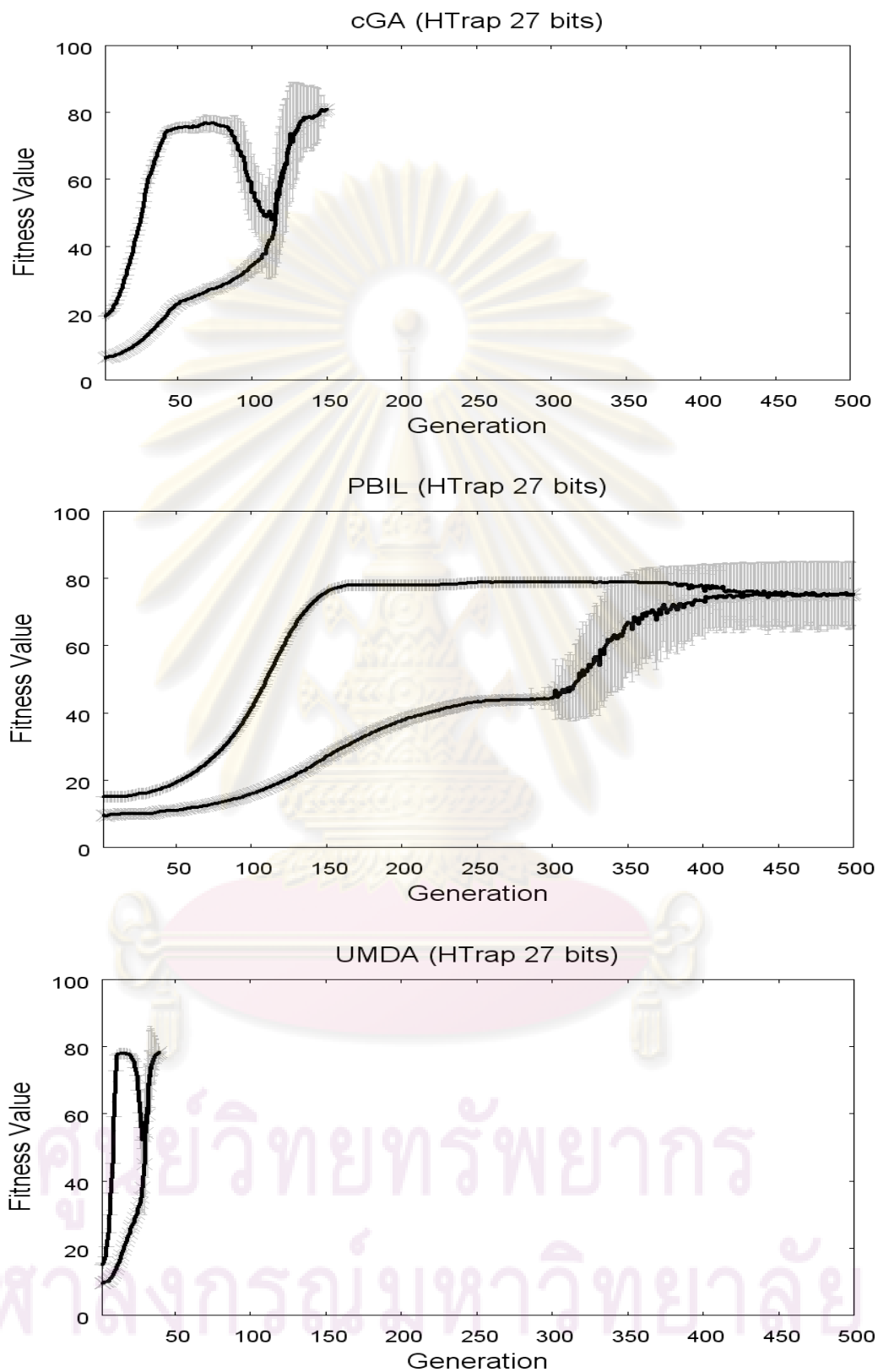
รูปที่ 4.2 ขอบเขตการใช้สิทธิจากการวิเคราะห์ด้วยประชากรที่มากขึ้น สำหรับปัญหาจำนวนหนึ่งมากที่สุด และบริเวณสีเทาในรูปคือค่าเบี่ยงเบนมาตรฐานของเส้นแบ่งทั้งสอง



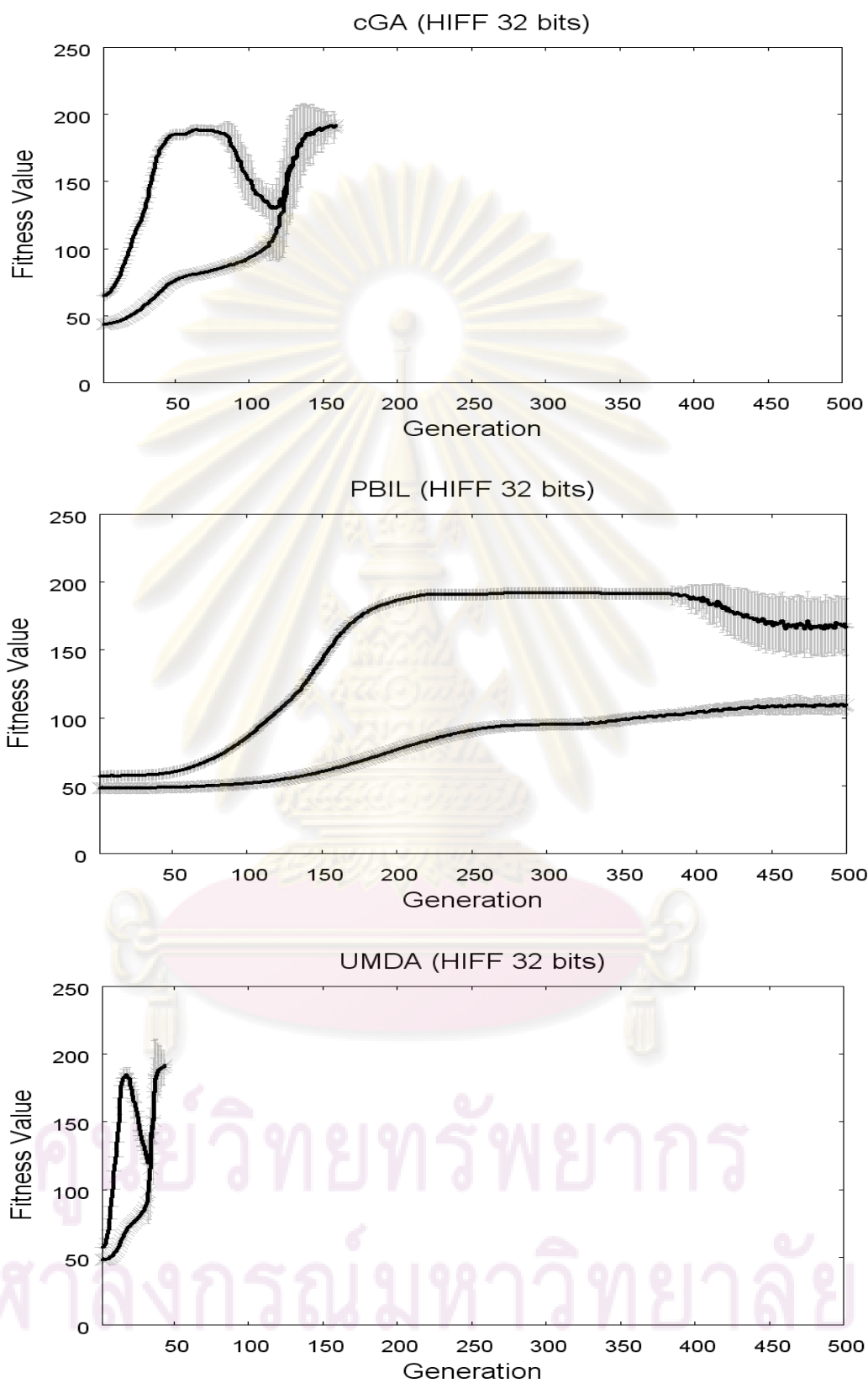
รูปที่ 4.3 ขอบเขตการใช้สิทธิจากการวิเคราะห์ด้วยประชากรที่มากขึ้น สำหรับปัญหาฟังก์ชันขนาด 3 และบริเวณสี่เท่าในรูปคือค่าเบี่ยงเบนมาตรฐานของเส้นแบ่งทั้งสอง



รูปที่ 4.4 ขอบเขตการใช้สิทธิจากการวิเคราะห์ด้วยประชากรที่มากขึ้น สำหรับปัญหาฟังก์ชันขนาด 5 และบริเวณสี่เท่าในรูปคือค่าเบี่ยงเบนมาตรฐานของเส้นแบ่งทั้งสอง



รูปที่ 4.5 ขอบเขตการใช้สิทธิจากการวิเคราะห์ด้วยประชากรที่มากขึ้น สำหรับปัญหาการจับคู่แบบลำดับชั้น และบริเวณสีเทาในรูปคือค่าเบี่ยงเบนมาตรฐานของเส้นแบ่งทั้งสอง



รูปที่ 4.6 ขอบเขตการใช้สิทธิจากการวิเคราะห์ด้วยประชากรที่มากขึ้น สำหรับปัญหาลำดับชั้นแบบกึ่งต่อเนื่อง และบริเวณสีเทาในรูปคือค่าเบี่ยงเบนมาตรฐานของเส้นแบ่งทั้งสอง

จากมูลค่าอัลกอริทึมในตารางที่ 4.2 ส่วนใหญ่แล้ววิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรจะมีมูลค่าอัลกอริทึมสูงที่สุดจากทั้ง 3 อัลกอริทึมที่นำมาเปรียบเทียบ ในขณะที่ขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปรจะเป็นวิธีที่มีมูลค่าอัลกอริทึมน้อยที่สุด ทั้งนี้เนื่องจากขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปรจะใช้ประชากรที่ถูกเลือกทั้งหมดมาเฉลี่ยเพื่อประมาณค่าเวกเตอร์ความน่าจะเป็นตามกลุ่มประชากรที่ถูกเลือกมา ซึ่งส่งผลให้เวกเตอร์ความน่าจะเป็นลู่เข้าอย่างรวดเร็วตามกลุ่มประชากรที่ดี และอาจไม่ใช้วิธีที่ดีสำหรับการแก้ปัญหาที่ดักซึ่งฟังก์ชันคำนวณค่าความเหมาะสมมีการหลอก

ตารางที่ 4.2 มูลค่าอัลกอริทึมและค่าดัชนีความยากของขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากร และขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปร

Algorithms Problems	cGA		PBIL		UMDA	
	Algorithm value (f)	Difficulty Index $1 - (f/F)$	Algorithm value (f)	Difficulty Index $1 - (f/F)$	Algorithm value (f)	Difficulty Index $1 - (f/F)$
30 bits OneMax ($F=30$)	30.00	0.00	30.00	0.00	29.99	0.00
3-Trap x 10 ($F=30$)	26.06	0.13	29.94	0.00	24.41	0.19
5-Trap x 6 ($F=30$)	24.93	0.17	24.00	0.20	23.98	0.20
27 bits HTrap ($F=81$)	45.19	0.44	76.27	0.06	51.44	0.36
32 bits HIFF ($F=192$)	116.58	0.39	132.82	0.31	108.92	0.43

จากค่าดัชนีความยากในตารางที่ 4.2 จะเห็นได้ว่าปัญหาที่ยากที่สุดสำหรับขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับในที่นี้คือปัญหากับดักแบบลำดับชั้น ในขณะที่ปัญหาที่ยากที่สุดสำหรับวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรและขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปรคือปัญหาลำดับชั้นแบบก็ต่อเมื่อ ทั้งนี้ก็เนื่องจากในปัญหาลำดับชั้นแบบก็ต่อเมื่อคำตอบที่ดีที่สุดเป็นไปได้ 2 แบบคือ เป็นบิต 1 ทั้งหมดหรือไม่ก็เป็นบิต 0 ทั้งหมด วิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรและขั้นตอนวิธีแจกแจงตามขอบหนึ่งตัวแปรใช้วิธีประมาณค่าเวกเตอร์ความน่าจะเป็นตามประชากรที่ดี ดังนั้นเมื่อสุ่มตัวอย่างขึ้นมาจากเวกเตอร์ความน่าจะเป็นนี้ การกระจายตัวของคำตอบย่อมมีบิต 1 และ 0 อยู่ปนกัน แต่ในขณะที่ขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับใช้วิธีการปรับปรุงเวกเตอร์ความน่าจะเป็นให้เข้าใกล้คำตอบที่ดี โอกาสที่ตัวอย่างที่ถูกสุ่มขึ้นมาใหม่จะมีคำตอบไปในทางใดทางหนึ่ง คือบิต 1 หรือ 0 มีโอกาสเป็นไปได้สูงกว่า

ผลสรุปที่น่าสนใจสำหรับการใช้วิธีออปชันจริงในการวิเคราะห์ขั้นตอนวิธีประมาณการแจกแจงทั้ง 3 แบบนี้ คือ กรณีแรก ถ้าเรามีทรัพยากรที่จำกัดเช่นมีประชากรน้อย ๆ ดังในการทดลองในหัวข้อ 4.1 ขั้นตอนวิธีเชิงพันธุกรรมแบบกระชับเป็นขั้นตอนวิธีที่เราควรเลือกใช้ในสถานการณ์ดังกล่าว เนื่องจากส่วนใหญ่แล้ววิธีนี้จะให้มูลค่าอัลกอริทึมสูง แต่ในกรณีที่เรามีประชากรจำนวนมาก วิธีที่เราควรเลือกใช้คือขั้นตอนวิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากร เนื่องจากในการทดลองในหัวข้อที่ 4.2 นี้แสดงให้เห็นว่าเมื่อมีจำนวนประชากรจำนวนมาก วิธีการเรียนรู้เพิ่มขึ้นแบบอาศัยประชากรให้มูลค่าอัลกอริทึมที่สูงกว่าเป็นส่วนใหญ่ หรือค่าดรชนี้ความยากเข้าใกล้ 0 แสดงว่าสามารถหาค่าตอบได้ใกล้เคียงค่าเหมาะสมสุด แต่ถ้าในกรณีที่เวลาในการได้รับคำตอบเป็นสิ่งสำคัญที่สุดที่เราจะพิจารณา อัลกอริทึมที่เราควรเลือกใช้คือขั้นตอนวิธีแจกแจงตามของหนึ่งตัวแปร เนื่องจากเมื่อเทียบมูลค่าอัลกอริทึมในกรณีคิดอัตราลดค่าแล้วพบว่าขั้นตอนวิธีแจกแจงตามของหนึ่งตัวแปรเป็นวิธีที่ให้มูลค่าอัลกอริทึมสูงที่สุด ผลการทดลองเมื่อคิดอัตราลดค่า 5% และ 10% แสดงในตารางที่ 4.3

ตารางที่ 4.3 มูลค่าอัลกอริทึมเมื่อพิจารณาอัตราลดค่า

Algorithms Problems	cGA		PBIL		UMDA	
	Discount 5%	Discount 10%	Discount 5%	Discount 10%	Discount 5%	Discount 10%
30 bits OneMax	14.52	13.37	13.86	13.19	18.35	14.18
3-Trap x 10	9.63	8.87	9.13	8.65	10.93	8.65
5-Trap x 6	9.80	9.02	9.32	8.77	13.54	9.45
27 bits HTrap	11.70	10.73	11.00	10.48	23.40	12.09
32 bits HIFF	49.68	47.18	49.31	47.06	49.39	47.07

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5

การศึกษาเวลาหยุดที่เหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่าย

ในบทนี้จะนำเสนอผลการศึกษากการหยุดของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายเมื่อใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิที่ได้มาจากการวิเคราะห์มูลค่าอัลกอริทึมโดยวิธีอุปชั้นจริง การทดลองจะเริ่มจากให้ขั้นตอนวิธีเชิงพันธุกรรมทำงานทั้งสิ้น 1,000 ครั้ง แล้วเก็บข้อมูลการทำงานทั้งหมดไว้ แล้วนำข้อมูลนี้มาวิเคราะห์เวลาหยุดที่เหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมโดยวิธีอุปชั้นจริง จากนั้นนำผลลัพธ์ที่ได้ไปทดสอบกับข้อมูลชุดเดิมที่เราได้มาจากการทำงานในตอนแรก เพื่อศึกษาว่าถ้าอัลกอริทึมตัดสินใจหยุดตามเงื่อนไขการหยุดที่เหมาะสมที่สุดแล้วจะสามารถประหยัดการคำนวณไปได้มากน้อยเพียงใด และค่าคำตอบที่ได้จะเปลี่ยนแปลงไปหรือไม่

ในการทดลองจะทดสอบขั้นตอนวิธีเชิงพันธุกรรมกับปัญหาทดสอบต่าง ๆ ดังนี้ ปัญหาจำนวนหนึ่งมากที่สุดขนาด 30, 50 และ 100 บิต ปัญหาบิตขนาด 3 ต่อกัน 10 ชุด, 20 ชุด และ 30 ชุด ปัญหาบิตขนาด 5 ต่อกัน 10 ชุด, 20 ชุด และ 30 ชุด ปัญหาบิตแบบลำดับชั้นขนาด 27, 81 และ 243 บิต ปัญหาลำดับชั้นแบบก็ต่อเมื่อขนาด 32, 64 และ 128 บิต ปัญหา 2 ชั้นแบบ 6 บิต ขนาด 30, 60 และ 90 บิต

ผลการทดลองจะได้มาจากการข้อมูลการทำงานของขั้นตอนวิธีพันธุกรรม 1,000 ครั้ง โดยมีค่าพารามิเตอร์ในการทดลองดังนี้ ใช้จำนวนประชากรทั้งสิ้น 50 ตัว อัตราการไขว้เปลี่ยน 1.0 อัตราการกลายพันธุ์ 0 จำนวนรุ่นสูงสุด 100 รุ่น และใช้การคัดเลือกแบบการแข่งขันด้วยอัตราการแข่งขันเท่ากับ 2

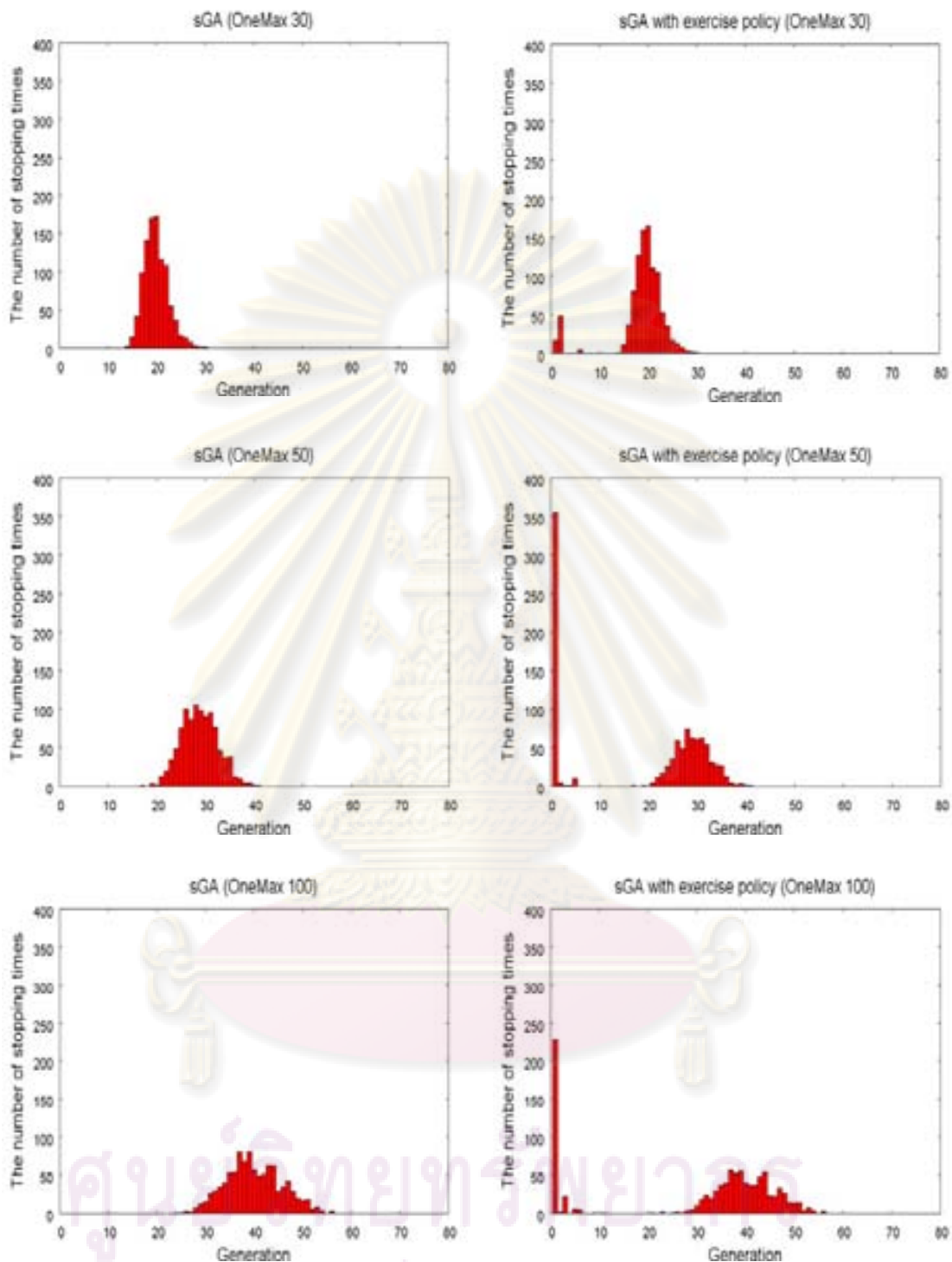
ตารางที่ 5.1 แสดงผลการทดลองเปรียบเทียบระหว่างขั้นตอนวิธีเชิงพันธุกรรมแบบง่าย และขั้นตอนวิธีเชิงพันธุกรรมแบบง่ายที่ใช้เงื่อนไขการหยุดที่เหมาะสมที่สุด เมื่อทดลองกับปัญหาทดสอบต่าง ๆ ค่าที่แสดงในตารางดังกล่าวจะประกอบด้วยค่าความเหมาะสมที่สุดที่ได้รับ (Best) ค่าความเหมาะสมเฉลี่ยจากการทำงานทั้งสิ้น 1,000 รอบ (Avg.) และค่าเบี่ยงเบนมาตรฐาน (S.D.) ส่วนจำนวนครั้งที่ใช้ในการประเมินค่าความเหมาะสมจะถูกแสดงไว้ในสมุดสุดท้ายของทั้ง 2 วิธี จำนวนครั้งที่ต้องใช้ในการคิดค่าความเหมาะสมนี้ถือเป็นค่าใช้จ่ายที่ต้องเสียไปในการหาคำตอบ ดังนั้นยิ่งค่านี้น้อยลงได้มากเพียงใด อัลกอริทึมก็จะสามารถลดการคำนวณลงได้มากขึ้นเท่านั้น ในสมุดสุดท้ายทางด้านขวามือจึงแสดงค่าความแตกต่างของค่าความเหมาะสมที่ได้รับและการลดการคำนวณที่เมื่อขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายใช้เงื่อนไขการหยุดที่เหมาะสมที่สุดแล้วจะสามารถลดการคำนวณลงได้เพียงใด

ตารางที่ 5.1 ผลการทดลองเปรียบเทียบระหว่างขั้นตอนวิธีเชิงพันธุกรรมแบบง่าย (sGA) และขั้นตอนวิธีเชิงพันธุกรรมแบบง่ายที่ใช้เงื่อนไขการหยุดที่เหมาะสมที่สุด โดยที่ $\Delta fitness$ คือผลต่างของความเหมาะสมเฉลี่ยที่ได้รับซึ่งก็คือคุณภาพคำตอบที่เพิ่มขึ้นหรือลดลง และ $\Delta eval$ คือผลต่างของจำนวนครั้งการประเมินคำตอบซึ่งก็คือต้นทุนการคำนวณที่ประหยัดได้

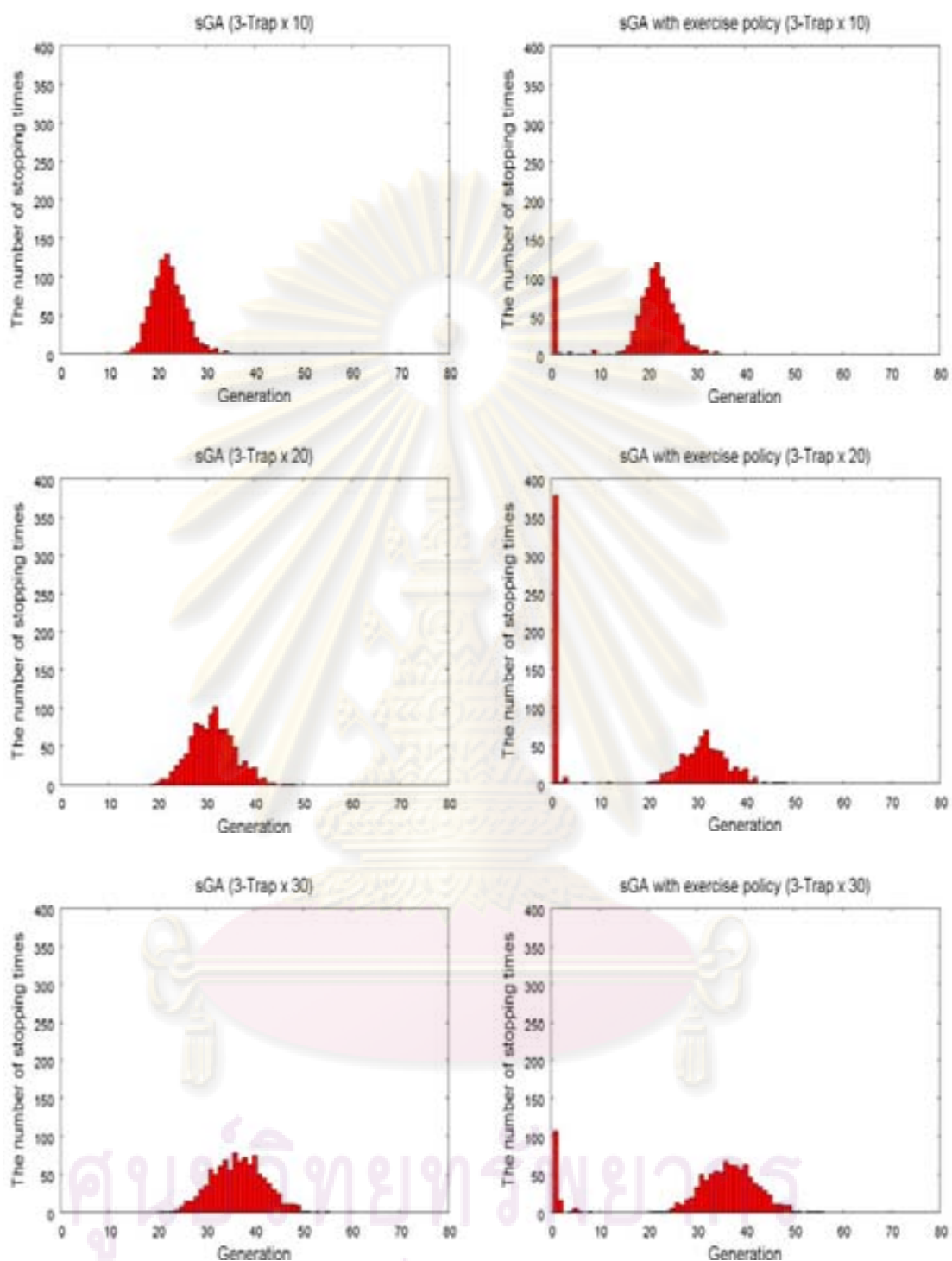
Problem	sGA				sGA with option-based stopping policy				Saving analysis	
	Fitness			# Evaluation	Fitness			# Evaluation	$\Delta fitness$ (of avg.)	saving $\Delta eval$
	Best	Avg.	S.D.		Best	Avg.	S.D.			
onemax30	30.00	29.85	0.37	1,095,550	30.00	29.50	1.34	988,600	-1.19%	9.76%
onemax50	50.00	48.67	1.09	1,538,450	50.00	43.91	6.31	989,500	-10.84%	35.68%
onemax100	97.00	89.65	2.60	2,096,100	97.00	83.38	10.42	1,546,800	-7.52%	26.21%
trap3_10	30.00	28.85	0.95	1,213,450	30.00	28.00	2.57	1,050,650	-3.04%	13.42%
trap3_20	60.00	53.81	2.13	1,668,700	60.00	46.37	9.38	1,043,500	-16.04%	37.47%
trap3_30	84.00	75.71	3.40	1,929,300	84.00	72.16	9.25	1,655,050	-4.92%	14.22%
trap5_10	49.00	43.68	1.85	1,551,700	49.00	38.56	6.95	1,023,500	-13.28%	34.04%
trap5_20	90.00	78.55	3.57	2,020,950	90.00	71.89	11.84	1,527,350	-9.26%	24.42%
trap5_30	124.00	108.57	4.75	2,240,250	121.00	93.01	18.04	1,331,050	-16.73%	40.58%
Htrap27	81.00	59.25	12.13	1,151,650	81.00	54.82	14.23	957,650	-8.08%	16.85%
Htrap81	184.50	121.87	14.31	1,779,950	184.50	100.86	28.64	1,191,000	-20.83%	33.09%
Htrap243	331.50	251.35	18.41	2,385,550	309.00	238.65	34.29	2,057,750	-5.32%	13.74%
HIFF32	192.00	143.63	21.57	1,188,350	192.00	126.39	30.60	854,000	-13.64%	28.14%
HIFF64	352.00	242.32	25.77	1,625,600	352.00	213.35	47.12	1,161,300	-13.58%	28.56%
HIFF128	520.00	402.14	27.72	2,095,050	520.00	374.70	57.26	1,673,450	-7.32%	20.12%
6-bipolar30	5.00	4.78	0.10	1,788,950	5.00	4.73	0.13	1,323,500	-1.06%	26.02%
6-bipolar60	9.90	9.37	0.14	1,948,550	9.90	9.16	0.28	1,180,250	-2.29%	39.43%
6-bipolar90	14.40	13.93	0.16	1,909,250	14.40	13.65	0.40	1,267,000	-2.05%	33.64%

ผลการทดลองจากตารางข้างต้นแสดงให้เห็นว่า เมื่อใช้ขอบเขตการใช้สิทธิเป็นเกณฑ์ในการตัดสินใจหยุดหรือดำเนินการต่อ ขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายใช้ผลรวมของจำนวนครั้งในการคิดค่าความเหมาะสมน้อยกว่าขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้เงื่อนไขการหยุดแบบปกติ ซึ่งในที่นี้คือหยุดเมื่อทำงานถึงจำนวนรุ่นสูงสุดที่กำหนดไว้หรือหยุดเมื่อค่าความเหมาะสมเฉลี่ยในรุ่นถัดไปไม่เปลี่ยนแปลง

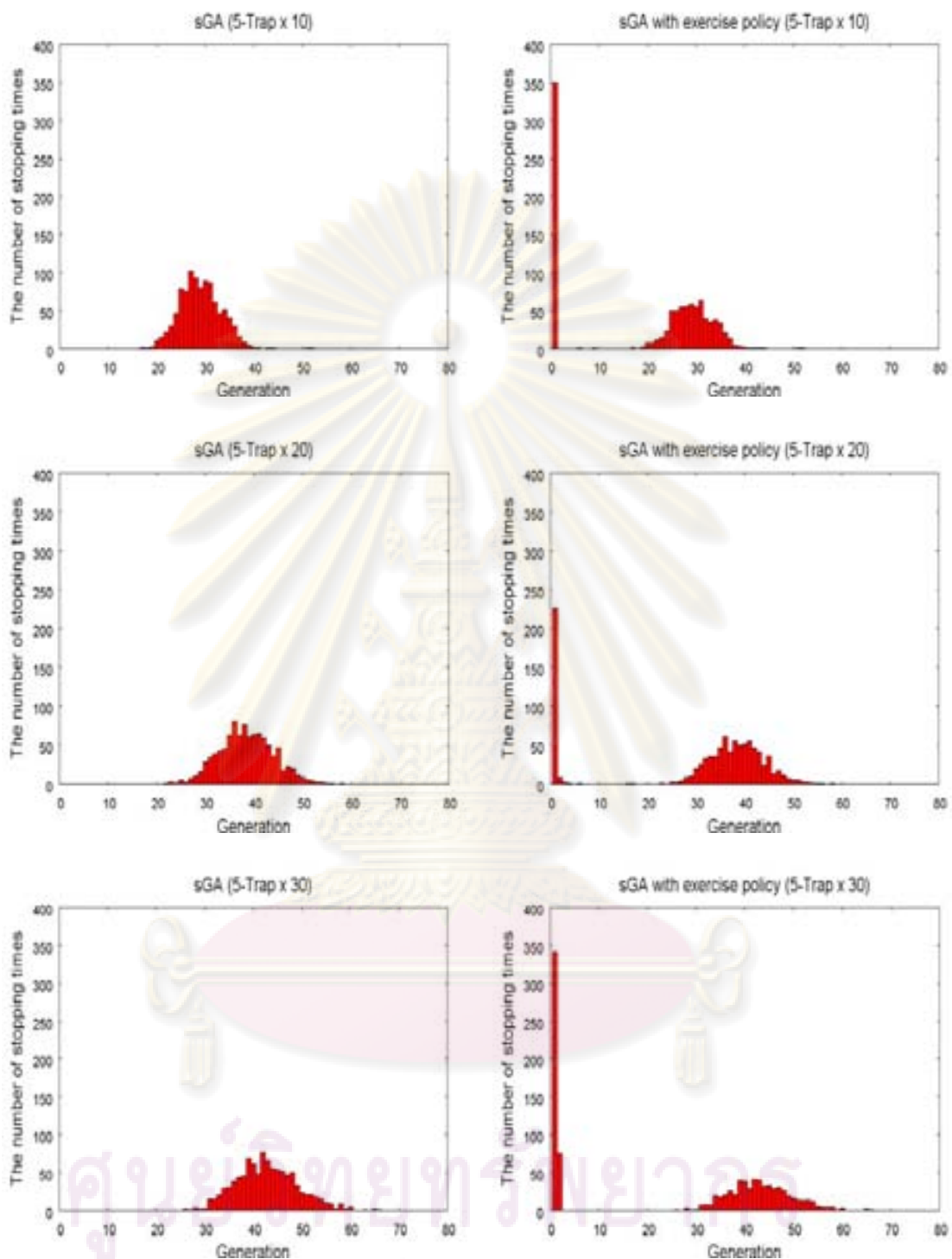
จำนวนครั้งการคิดค่าความเหมาะสมที่ขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้ขอบเขตการใช้สิทธิเป็นเกณฑ์ในการหยุดสามารถลดลงไปได้ มีค่าตั้งแต่ 9.76% ไปจนถึง 40.58% หรือโดยเฉลี่ยประมาณ 26.41% ในขณะที่ค่าความเหมาะสมเฉลี่ยลดลงตั้งแต่ 1.06% จนถึง 20.83% หรือโดยเฉลี่ยประมาณ 8.72% จากตัวเลขดังกล่าวจะเห็นว่าสัดส่วนในการลดการคำนวณมีค่ามากกว่าค่าความเหมาะสมที่ลดลง แต่ค่าความเหมาะสมดีที่สุดที่ได้รับจากทั้งสองวิธีนั้นมีค่าใกล้เคียงกันมาก ทั้งนี้เนื่องจากในบางรอบของการทำงานที่ดูเหมือนจะไม่พบคำตอบที่ดีไปกว่าเดิม ขั้นตอนวิธีเชิงพันธุกรรมที่ใช้ค่าขอบเขตการใช้สิทธิเป็นเกณฑ์ในการหยุดจะตัดสินใจหยุดเร็วกว่าขั้นตอนวิธีเชิงพันธุกรรมที่ใช้เงื่อนไขการหยุดแบบทั่วไป ทั้งนี้สามารถดูได้จากกราฟจำนวนครั้งที่อัลกอริทึมตัดสินใจหยุดในรุ่นต่าง ๆ ที่แสดงในรูปที่ 5.1 ถึง 5.6



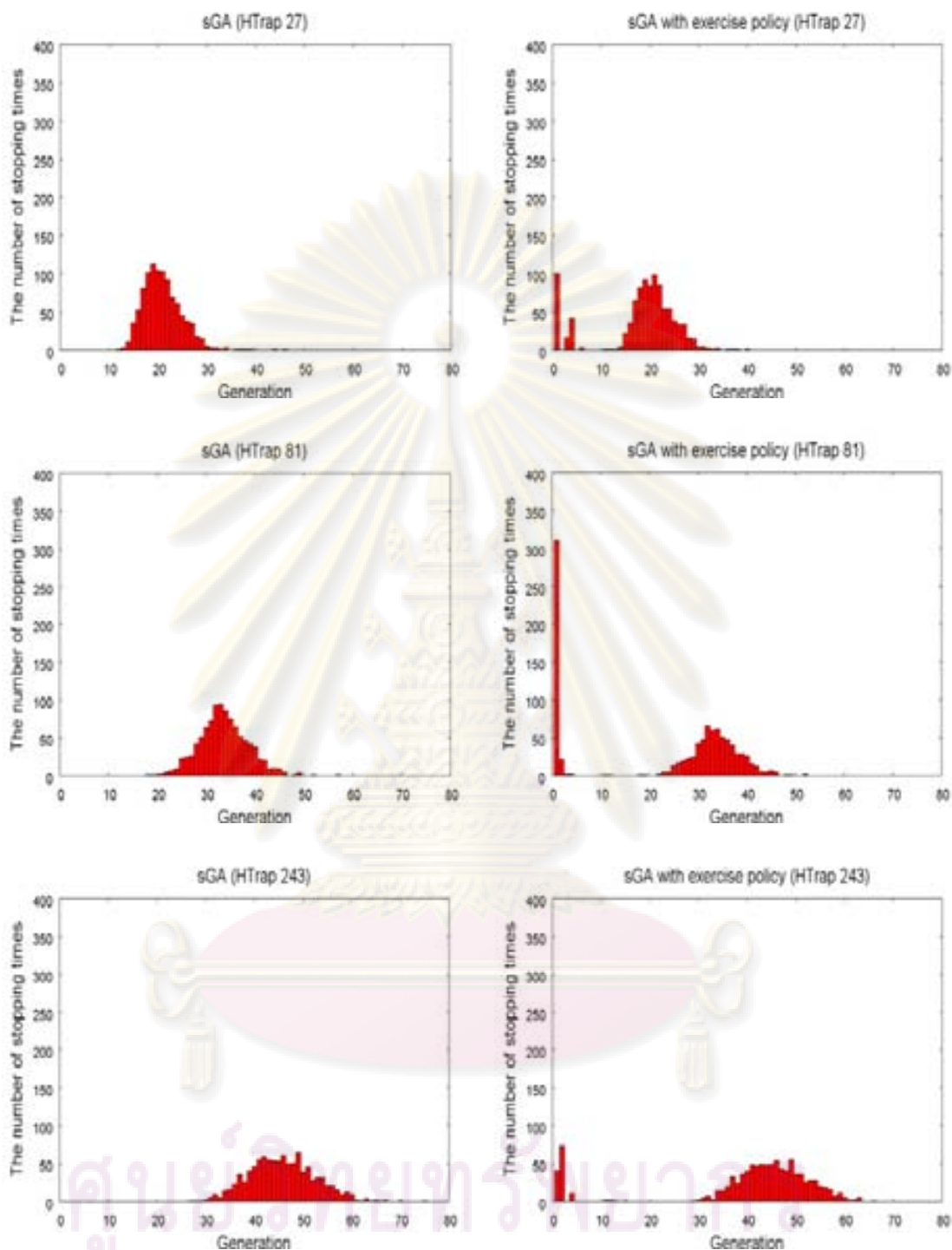
รูปที่ 5.1 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่าย (กราฟทางซ้าย) และขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิ (กราฟทางขวา) โดยเรียงลำดับปัญหาทดสอบจากบนลงล่างคือ ปัญหาจำนวนหนึ่งมากที่สุดขนาด 30, 50 และ 100 บิต



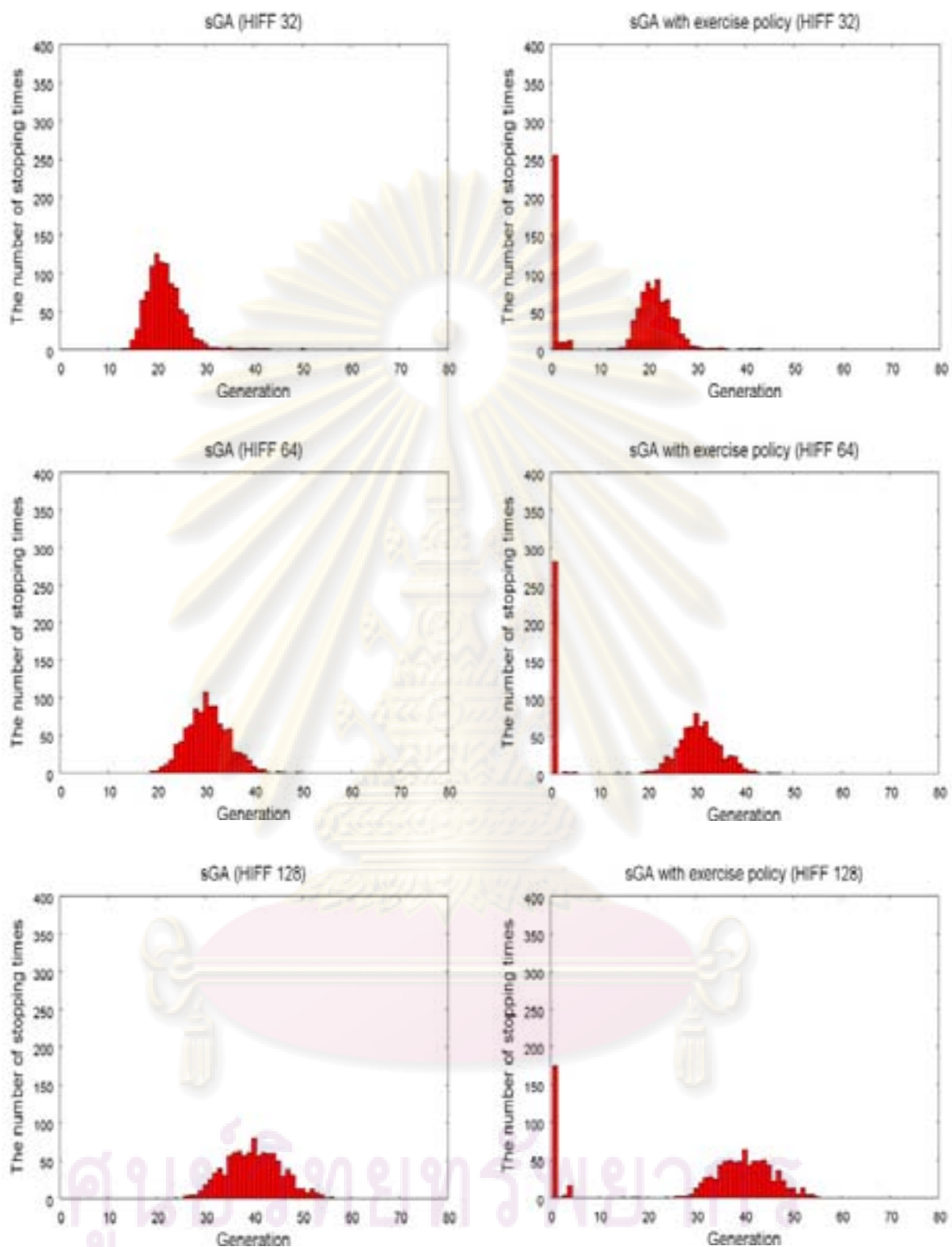
รูปที่ 5.2 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่าย (กราฟทางซ้าย) และขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิ (กราฟทางขวา) โดยเรียงลำดับปัญหาทดสอบจากบนลงล่างคือ ปัญหาที่ดักขนาด 3 ต่อกันจำนวน 10, 20 และ 30 ชุด



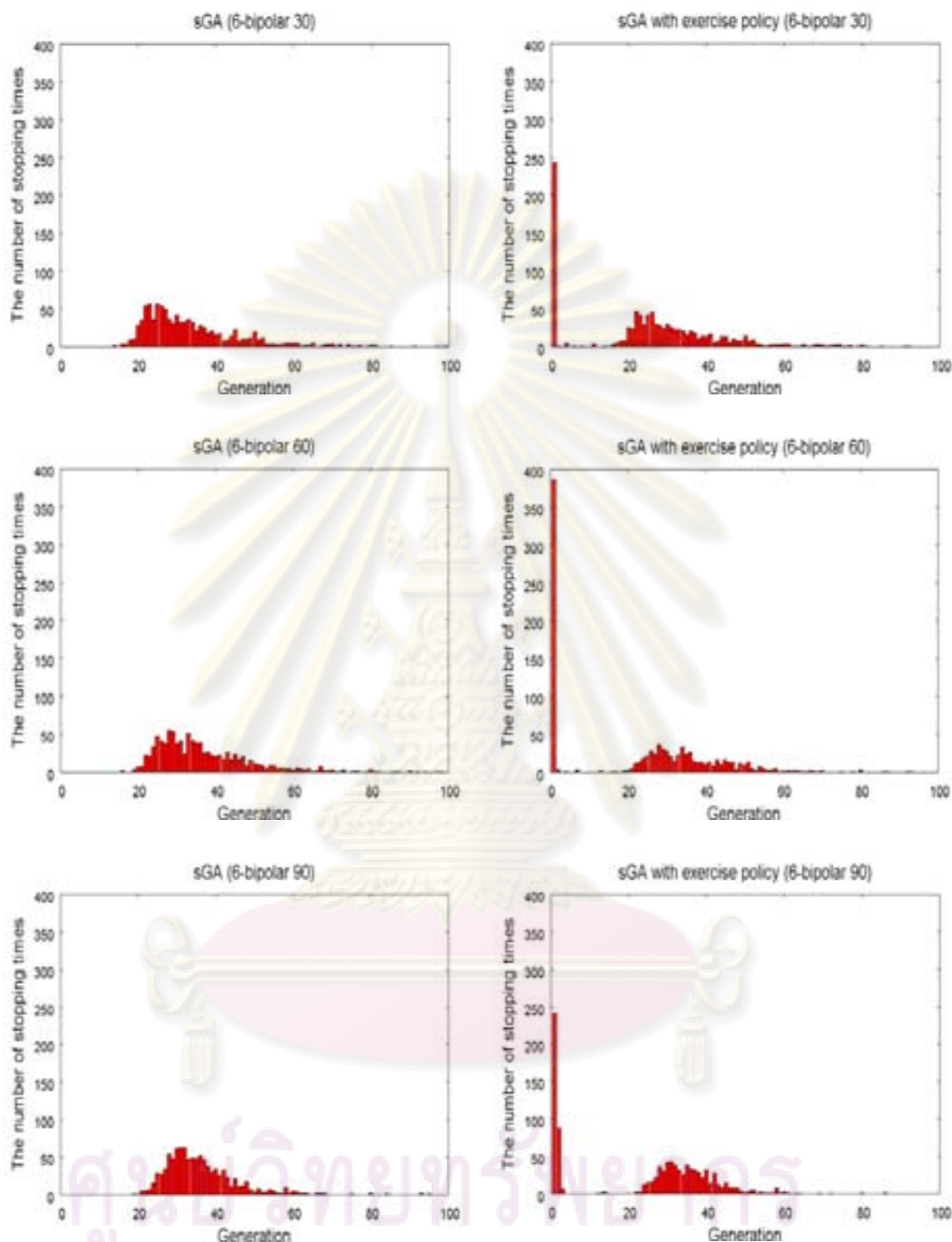
รูปที่ 5.3 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่าย (กราฟทางซ้าย) และขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิ (กราฟทางขวา) โดยเรียงลำดับปัญหาทดสอบจากบนลงล่างคือ ปัญหากับดักขนาด 5 ต่อกันจำนวน 10, 20 และ 30 ชุด



รูปที่ 5.4 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่าย (กราฟทางซ้าย) และขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิ (กราฟทางขวา) โดยเรียงลำดับปัญหาทดสอบจากบนลงล่างคือ ปัญหาที่ 27, 81 และ 243 บิต



รูปที่ 5.5 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่าย (กราฟทางซ้าย) และขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิ (กราฟทางขวา) โดยเรียงลำดับปัญหาทดสอบจากบนลงล่างคือ ปัญหาลำดับชั้นแบบกั ต่อเมื่อขนาด 32, 64 และ 128 บิต



รูปที่ 5.6 จำนวนครั้งการหยุดทำงานในรุ่นต่าง ๆ ของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่าย (กราฟทางซ้าย) และขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิ (กราฟทางขวา) โดยเรียงลำดับปัญหาทดสอบจากบนลงล่างคือ ปัญหา 2 ขั้วแบบ 6 บิต ขนาด 30, 60 และ 90 บิต

จากกราฟดังกล่าว ในแต่ละปัญหาทดสอบจะแสดงกราฟเป็น 2 กราฟคู่กัน กราฟทางซ้ายเป็นจำนวนครั้งการหยุดในรุ่นต่าง ๆ ของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้เงื่อนไขการหยุดแบบทั่วไป และกราฟทางขวาคือจำนวนครั้งการตัดสินใจหยุดการทำงานในรุ่นต่าง ๆ จากการใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิ

จากกราฟจะเห็นได้ว่าเมื่อใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิ จะมีการทำงานบางรอบที่อัลกอริทึมตัดสินใจหยุดเร็วกว่าปกติ ผลการลดจำนวนครั้งในการคำนวณก็มาจากการเลิกการทำงานในบางรอบที่ค่าความเหมาะสมต่ำเกินกว่าที่ควรจะทำงานต่อ หรือไม่ก็หยุดในกรณีที่ค่าความเหมาะสมสูงเพียงพอแล้ว ในกรณีแรกคือกรณีที่อัลกอริทึมหยุดทำงานเนื่องจากค่าความเหมาะสมเฉลี่ยในรุ่นนั้น ๆ ต่ำกว่าค่าเส้นแบ่งด้านล่างในขอบเขตการใช้สิทธิที่ตัดสินใจหยุดก็เนื่องจากเหตุการณ์นี้หมายความว่าประชากรกลุ่มนั้นไม่สามารถเพิ่มค่าความเหมาะสมได้รวดเร็วเพียงพอจากกระบวนการวิวัฒนาการที่ผ่านมา ดังนั้นจึงควรหยุดการทำงานกับกลุ่มประชากรนี้ซึ่งวิวัฒนาการช้ากว่าที่จะยอมรับได้ ในกรณีที่สอง อัลกอริทึมจะหยุดทำงานเมื่อค่าความเหมาะสมเฉลี่ยสูงกว่าค่าเส้นแบ่งด้านบน เหตุการณ์นี้แสดงว่าขณะนี้เราได้รับคำตอบที่ดีกว่าค่าคาดหวัง ซึ่งคำตอบเหล่านี้ก็ได้มาในเวลาที่รวดเร็วด้วย การตัดสินใจหยุดการทำงานแล้วยอมรับคำตอบดี ๆ ที่ได้มาในเวลาอันสั้น จึงเป็นทางเลือกที่คุ้มที่สุด

ดังที่ได้กล่าวไปแล้วว่าการตัดสินใจหยุดเกิดจาก 2 กรณีข้างต้น ดังนั้นเพื่อให้ทราบว่าในแต่ละปัญหาทดสอบ อัลกอริทึมหยุดเนื่องจากเหตุการณ์ใดเป็นจำนวนครั้งเท่าใดในการทดลองจึงได้นับจำนวนครั้งที่หยุดการทำงานตามเงื่อนไขดังกล่าวแสดงในตารางที่ 5.2

ตารางที่ 5.2 แสดงจำนวนครั้งการหยุดโดยใช้เงื่อนไขการหยุดจากขอบเขตการใช้สิทธิในแต่ละปัญหาทดสอบ สถิติทางซ้ายเป็นจำนวนครั้งการหยุดเมื่อค่าความเหมาะสมเฉลี่ยสูงกว่าเส้นแบ่งด้านบน และสถิติทางขวาแสดงจำนวนครั้งการหยุดเมื่อค่าความเหมาะสมเฉลี่ยต่ำกว่าเส้นแบ่งด้านล่าง ตัวเลขที่แสดงในทั้ง 2 สถิตินี้เป็นจำนวนครั้งการหยุดตามเงื่อนไขดังกล่าวภายใต้การทำงานทั้งสิ้น 1,000 รอบ ตัวอย่างเช่น ในปัญหาจำนวนหนึ่งมากที่สุดขนาด 30 บิต มีจำนวนครั้งการหยุดจากขอบเขตด้านบนทั้งสิ้น 70 ครั้ง จาก 1,000 ครั้ง ในขณะที่ไม่มีการหยุดจากขอบเขตด้านล่างเลย

ตารางที่ 5.2 จำนวนครั้งการหยุดทำงานตามเงื่อนไขเส้นแบ่งด้านบนและด้านล่าง

Problem	Stop over upper threshold	Stop under lower threshold
onemax30	70	0
onemax50	371	0
onemax100	268	0
trap3_10	114	0
trap3_20	390	0
trap3_30	135	0
trap5_10	354	0
trap5_20	244	0
trap5_30	420	0
Htrap27	192	0
Htrap81	344	0
Htrap243	134	11
HIFF32	324	1
HIFF64	296	0
HIFF128	204	6
6-bipolar30	445	1
6-bipolar60	414	1
6-bipolar90	359	2

จากตารางดังกล่าวจะเห็นได้ว่าส่วนใหญ่แล้วอัลกอริทึมจะตัดสินใจหยุดการทำงานจากเงื่อนไขเส้นแบ่งด้านบน จะปรากฏว่ามีการหยุดเมื่อค่าความเหมาะสมต่ำกว่าเส้นแบ่งด้านล่างก็ในปัญหาที่ดักแบบลำดับชั้นขนาด 243 บิต ปัญหาลำดับชั้นแบบก็ต่อเมื่อขนาด 32 และ 128 บิต และกลุ่มของปัญหา 2 ชั้นแบบ 6 บิต ทั้งหมด สังเกตได้ว่าปัญหาเหล่านี้ที่มีการตัดสินใจหยุดจากขอบเขตด้านล่าง ล้วนแล้วแต่เป็นปัญหาที่ยากทั้งสิ้น จากผลการทดลองนี้แสดงให้เห็นว่าในปัญหาที่ยากมาก ๆ การตัดสินใจหยุดเมื่อกลุ่มประชากรมีแนวโน้มว่าไม่สามารถจะเจอคำตอบที่ดีขึ้นได้ เป็นการตัดสินใจที่เหมาะสมและช่วยลดภาระการคำนวณลงได้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 6

การหาเวลาหยุดที่เหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมแบบค่อยเป็นค่อยไป

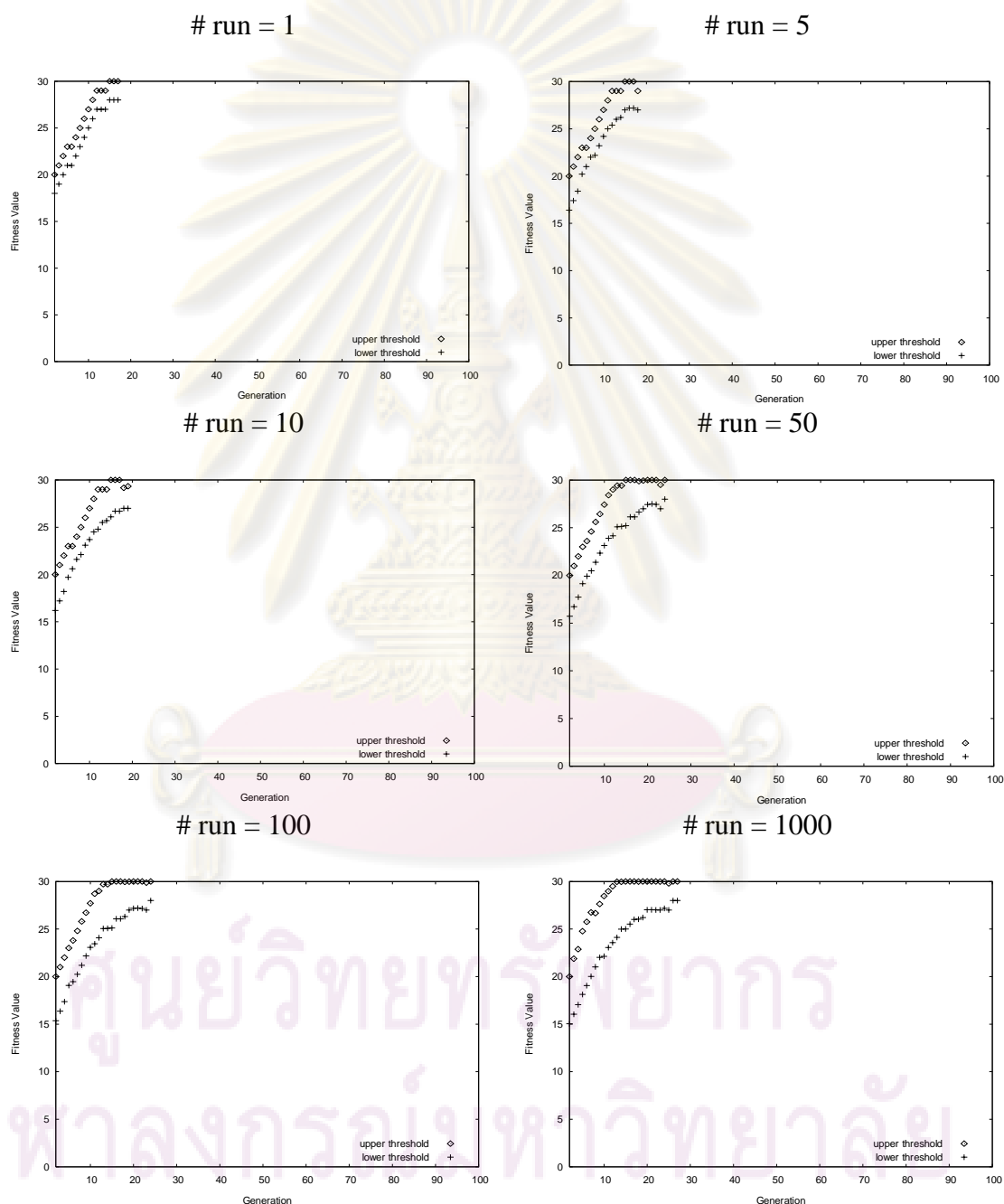
ในบทที่ผ่านมา ได้ทำการหาเวลาหยุดที่เหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่าย และใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิเพื่อศึกษาถึงการลดการคำนวณของอัลกอริทึมเมื่อตัดสินใจหยุดแบบคุ่มที่สุด ในการทดลองดังกล่าวเราได้ใช้ข้อมูลจากการทำงานทั้งหมดของอัลกอริทึมมาหาวิธีตัดสินใจที่เหมาะสมที่สุด การทดลองดังกล่าวถูกสร้างขึ้นเพื่อทดสอบว่าเมื่ออัลกอริทึมตัดสินใจตามเงื่อนไขการหยุดของขอบเขตการใช้สิทธิจะสามารถลดจำนวนครั้งการคำนวณลงได้จริง วิธีนี้อาจไม่เหมาะไปใช้ในทางปฏิบัติเพราะในความเป็นจริงเราไม่สามารถทำงานจนเสร็จแล้วค่อยมาหาวิธีการตัดสินใจ ดังนั้นเพื่อให้สามารถนำวิธีการหาเวลาหยุดเหมาะสมที่สุดโดยวิธีอุปชันจริงนี้ไปใช้กับขั้นตอนวิธีเชิงพันธุกรรมได้ในทางปฏิบัติ วิทยานิพนธ์นี้จึงได้นำเสนอการหาเวลาหยุดเหมาะสมที่สุดแบบค่อยเป็นค่อยไป กล่าวคือ ในการทำงานแต่ละรอบเราจะเก็บข้อมูลของการทำงานนี้ไว้ แล้วนำมาคำนวณมูลค่าอัลกอริทึมและวิธีการตัดสินใจที่เหมาะสม แล้วนำวิธีตัดสินใจนี้ไปเป็นเงื่อนไขการหยุดของการทำงานรอบถัดไป ข้อมูลการทำงานจะถูกปรับปรุงขึ้นเรื่อย ๆ และขอบเขตการใช้สิทธิก็จะแม่นยำขึ้น การทำงานโดยวิธีนี้ขั้นตอนวิธีเชิงพันธุกรรมจะต้องทำงานไปอย่างน้อย 1 รอบก่อนเสมอ ดังนั้นวิธีนี้จะมีประโยชน์ก็ต่อเมื่อนำไปใช้กับงานที่มีการทดลองมากกว่า 1 รอบ ซึ่งก็เป็นวิธีปฏิบัติที่ใช้อยู่กับขั้นตอนวิธีเชิงพันธุกรรมโดยทั่วไป เนื่องจากการทดลองครั้งหนึ่ง ๆ ขั้นตอนวิธีเชิงพันธุกรรมอาจให้คำตอบที่ต่างกัน การทดลองซ้ำหลาย ๆ รอบจึงเป็นวิธีที่ใช้กันโดยปกติ

หลักการงานของการหาเวลาหยุดเหมาะสมที่สุดแบบค่อยเป็นค่อยไปคือ จะมีการปรับปรุงค่าขอบเขตการใช้สิทธิทุก ๆ ครั้งที่มีการทำงานครั้งใหม่ การทำงานของขั้นตอนวิธีเชิงพันธุกรรม 1 รอบ จะมีการคำนวณมูลค่าอัลกอริทึมใหม่ทุกครั้ง และค่าที่ได้พร้อมกับวิธีการตัดสินใจที่เหมาะสมที่สุดจะถูกใช้เป็นเงื่อนไขการหยุดสำหรับการทำงานของขั้นตอนวิธีเชิงพันธุกรรมในรอบถัดไป ขั้นตอนวิธีของการหาเวลาหยุดแบบค่อยเป็นค่อยไปนี้แสดงในรูปที่ 6.1

```
Procedure Option-based GA with an incremental on-going improvement
1: Initial a set of matrix for collecting a fitness movement of each generation
2: for  $r = 1$  to MAXRUN do
3:   Initialize population
4:   for  $g = 1$  to MAXGEN do
5:     selection
6:     perform genetic operators
7:     evaluate population
8:     update profile according to a fitness movement from generation  $g-1$  to  $g$ 
9:     if fitness value falls into stopping regions then
10:      stop the algorithm (exit this loop and perform real_options_calculati on module)
11:   end for
12:   real_options_calculati on();
13:   update stoppi ng poli cy accordi ng to the calcul ated al gori thm val ue
14: end for
```

รูปที่ 6.1 ขั้นตอนวิธีของการหาเวลาหยุดเหมาะสมที่สุดแบบค่อยเป็นค่อยไป

จากการทำงานตามอัลกอริทึมดังกล่าว ขอบเขตการใช้สิทธิจะมีการเปลี่ยนแปลงค่าใหม่ในทุก ๆ รอบของการทำงาน ตัวอย่างขอบเขตการใช้สิทธิในการทำงานรอบต่าง ๆ แสดงในรูปที่ 6.2 จากรูปจะสังเกตเห็นได้ว่าขอบเขตการใช้สิทธิจะมีการเปลี่ยนแปลงเมื่อจำนวนรอบของการทำงานมากขึ้น เราจะเริ่มใช้ประโยชน์จากวิธีการตัดสินใจได้ในระยะเวลาอันสั้น การทำงาน 5 ถึง 10 รอบก็ให้ขอบเขตการใช้สิทธิที่ค่อนข้างใกล้เคียงกับความเป็นจริง



รูปที่ 6.2 ตัวอย่างขอบเขตการใช้สิทธิจากปัญหาจำนวนหนึ่งมากที่สุดขนาด 30 บิต

6.1 การทดลองการหยุดตามขอบเขตการใช้สิทธิซึ่งมีการปรับปรุงแบบค่อยเป็นค่อยไปโดยไม่มีต้นทุนการคำนวณ

ในหัวข้อนี้จะทำการทดลองใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิซึ่งมีการปรับปรุงแบบค่อยเป็นค่อยไปในกรณีที่ไม่มีต้นทุนการคำนวณเข้ามาเกี่ยวข้อง ค่าพารามิเตอร์ต่าง ๆ ที่ใช้ในการทดลองได้ถูกกำหนดให้เหมือนกับบทที่ผ่านมา แต่เพิ่มเงื่อนไขการหยุดในแต่รอบการทำงานของขั้นตอนวิธีเชิงพันธุกรรม โดยถ้าค่าความเหมาะสมที่สุดที่ได้รับในขณะนั้นมากกว่าเส้นแบ่งด้านบน หรือค่าความเหมาะสมต่ำสุดต่ำกว่าเส้นแบ่งด้านล่างก็จะหยุดการทำงานแล้วเริ่มการทำงานด้วยประชากรชุดใหม่ การที่เราพิจารณาการหยุดจากค่าความเหมาะสมสูงสุดและต่ำสุดนี้ก็เพื่อให้แน่ใจว่าค่าความเหมาะสมที่ได้รับในขณะนั้นสูงหรือต่ำพอที่สมควรจะหยุดการทำงาน ผลการทดลองด้วยวิธีการหยุดแบบนี้จะนำมาเปรียบเทียบกับวิธีหยุดแบบปกติของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่าย ผลการทดลองแสดงในตารางที่ 6.1

ตารางที่ 6.1 ตารางเปรียบเทียบผลการหยุดโดยใช้ขอบเขตการใช้สิทธิซึ่งมีการปรับปรุงแบบค่อยเป็นค่อยไปและการหยุดแบบปกติของขั้นตอนวิธีเชิงพันธุกรรมแบบง่าย โดยที่ Δ fitness คือผลต่างของค่าความเหมาะสมเฉลี่ยที่ได้รับซึ่งก็คือคุณภาพคำตอบที่เพิ่มขึ้นหรือลดลง และ Δ eval คือผลต่างของจำนวนครั้งการประเมินคำตอบซึ่งก็คือต้นทุนการคำนวณที่ประหยัดได้

Problem	sGA				sGA with option-based stopping policy				Saving analysis	
	Fitness			# Evaluation	Fitness			# Evaluation	Δ fitness (of avg.)	saving Δ eval
	Best	Avg.	S.D.		Best	Avg.	S.D.			
onemax30	30.00	29.85	0.37	1,095,550	30.00	29.85	0.38	1,087,400	0.00%	0.74%
onemax50	50.00	48.67	1.09	1,538,450	50.00	48.63	1.03	1,529,300	-0.08%	0.59%
onemax100	97.00	89.65	2.60	2,096,100	96.00	89.55	2.60	2,053,500	-0.11%	2.03%
trap3_10	30.00	28.85	0.95	1,213,450	30.00	28.80	1.02	1,205,450	-0.17%	0.66%
trap3_20	60.00	53.81	2.13	1,668,700	60.00	53.82	2.22	1,649,050	0.02%	1.18%
trap3_30	84.00	75.71	3.40	1,929,300	84.00	75.68	3.37	1,872,000	-0.04%	2.97%
trap5_10	49.00	43.68	1.85	1,551,700	49.00	43.61	1.81	1,531,850	-0.16%	1.28%
trap5_20	90.00	78.55	3.57	2,020,950	88.00	78.30	3.36	1,906,450	-0.32%	5.67%
trap5_30	124.00	108.57	4.75	2,240,250	122.00	107.96	4.61	2,019,700	-0.57%	9.84%
Htrap27	81.00	59.25	12.13	1,151,650	81.00	59.75	12.53	1,129,350	0.84%	1.94%
Htrap81	184.50	121.87	14.31	1,779,950	184.50	122.27	13.95	1,693,500	0.33%	4.86%
Htrap243	331.50	251.35	18.41	2,385,550	316.50	250.14	16.75	2,155,250	-0.48%	9.65%
HIFF32	192.00	143.63	21.57	1,188,350	192.00	144.00	21.65	1,146,700	0.26%	3.50%
HIFF64	352.00	242.32	25.77	1,625,600	384.00	241.07	25.10	1,517,350	-0.52%	6.66%
HIFF128	520.00	402.14	27.72	2,095,050	492.00	399.38	25.76	1,886,250	-0.69%	9.97%
6-bipolar30	5.00	4.78	0.10	1,788,950	5.00	4.78	0.10	1,670,050	0.00%	6.65%
6-bipolar60	9.90	9.37	0.14	1,948,550	9.80	9.38	0.14	1,746,800	0.11%	10.35%
6-bipolar90	14.40	13.93	0.16	1,909,250	14.50	13.94	0.16	1,788,000	0.07%	6.35%

จากผลการทดลองในตารางที่ 6.1 ค่าความเหมาะสมเฉลี่ยที่ได้รับจากวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้วิธีหยุดแบบปกติกับใช้วิธีหยุดตามเงื่อนไขเส้นแบ่งด้านบนและเส้นแบ่งด้านล่างของขอบเขตการใช้สิทธิมีค่าเฉลี่ยใกล้เคียงกันมาก การทดสอบความแตกต่างในทางสถิติจึงถูกนำมาใช้ทดสอบว่าค่าความเหมาะสมเฉลี่ยที่ได้รับจากทั้ง 2 วิธีดังกล่าวมีความแตกต่างกันหรือไม่ วิธีที่นำมาใช้ทดสอบคือการวัดความแตกต่างแบบจับคู่ (Paired-sample t-test) ซึ่งมีสมมติฐานในการทดสอบดังนี้

$$H_0: \mu = \mu_0$$

$$H_A: \mu \neq \mu_0$$

สมมติฐาน H_0 หมายความว่าค่าความเหมาะสมเฉลี่ยของทั้ง 2 วิธีไม่มีความแตกต่างกัน ในขณะที่สมมติฐาน H_A หมายความว่าค่าเฉลี่ยจากทั้ง 2 วิธีแตกต่างกัน และเราจะปฏิเสธสมมติฐาน H_0 เมื่อค่าความน่าจะเป็น (p-value) น้อยกว่า 0.05 เมื่อกำหนดค่าความเชื่อมั่นที่ 95% ตารางที่ 6.2 แสดงผลการทดสอบความแตกต่างของค่าความเหมาะสมเฉลี่ยของทั้ง 2 วิธี

ตารางที่ 6.2 การทดสอบความแตกต่างของค่าความเหมาะสมเฉลี่ยของทั้ง 2 วิธี

Problem	t	p-value (Sig. 2-tailed)
onemax30	.237	.813
onemax50	.883	.377
onemax100	.852	.394
trap3_10	1.093	.275
trap3_20	-.062	.950
trap3_30	.213	.832
trap5_10	.840	.401
trap5_20	1.646	.100
trap5_30	2.863	.004
Htrap27	-.889	.374
Htrap81	-.643	.520
Htrap243	1.559	.119
HIFF32	-.377	.706
HIFF64	1.062	.288
HIFF128	2.333	.020
6-bipolar30	.548	.584
6-bipolar60	-.564	.573
6-bipolar90	-2.732	.006

ค่าสถิติในตารางที่ 6.2 แสดงว่าค่าความเหมาะสมเฉลี่ยของการหยุดโดยขอบเขตการใช้สิทธิที่มีการปรับปรุงแบบค่อยเป็นค่อยไปกับวิธีการหยุดแบบปกติไม่แตกต่างกันที่ความเชื่อมั่น 95% ยกเว้นในปัญหาเกี่ยวกับดัชนีขนาด 5 ต่อกัน 30 ชุด ปัญหาลำดับชั้นแบบก็ต่อเมื่อขนาด 128 บิต และปัญหา 2 ชั้นแบบ 6 บิตขนาด 90 บิต ซึ่งค่าความเหมาะสมเฉลี่ยต่างกันอย่างมีนัยสำคัญเล็กน้อย

จากผลการทดลองในตารางที่ 6.1 และ 6.2 สามารถสรุปได้ว่าที่ค่าความเชื่อมั่น 95% ที่ค่าความเหมาะสมเฉลี่ยของทั้ง 2 วิธีไม่แตกต่างกัน วิธีการหยุดโดยใช้ขอบเขตการใช้สิทธิสามารถลดการคำนวณลงได้ตั้งแต่ 0.59% ถึง 10.35%

การทดลองในส่วนถัดไป จะนำต้นทุนในการคำนวณมาพิจารณาร่วมด้วย เพื่อศึกษาถึงพฤติกรรมของการหยุดของอัลกอริทึมเมื่อต้นทุนการคำนวณสูงขึ้น

6.2 การทดลองการหยุดตามขอบเขตการใช้สิทธิซึ่งมีการปรับปรุงแบบค่อยเป็นค่อยไปโดยพิจารณาต้นทุนการคำนวณ

การทดลองในส่วนนี้ได้นำต้นทุนในการคำนวณมาพิจารณาร่วมด้วย สมการในการคำนวณมูลค่าอัลกอริทึมจะเป็นไปตามสมการที่ 3.2 แต่ไม่มีการคิดอัตราลดค่า (Discount rate) กำหนดให้ค่าใช้จ่ายที่เกิดจากคำนวณค่าความเหมาะสมแต่ละครั้งคือ c ถ้าในแต่ละรุ่นมีประชากร N ตัว ค่าใช้จ่ายที่เกิดขึ้นในแต่ละรุ่นก็จะเท่ากับ cN ในการคำนวณก็จะหักต้นทุนนี้ตั้งสมการที่ 6.1

$$F(x) = \max \{ g(x), \mathcal{E}[F(x') | x] - cN \} \quad (6.1)$$

ในความเป็นจริง ต้นทุนการคำนวณนี้สามารถคิดได้จากต้นทุนจริงของการทดสอบค่าความเหมาะสมของคำตอบใด ๆ ตัวอย่างเช่นในงานที่ต้องนำคำตอบไปทดสอบกับอุปกรณ์จริงไม่ว่าจะเป็นอุปกรณ์ฮาร์ดแวร์หรือการทดสอบสารเคมี ต้นทุนในการทดสอบสามารถคำนวณได้จากค่าใช้จ่ายที่เกิดขึ้นจริง แต่การทดลองในส่วนนี้ทำขึ้นเพื่อศึกษาผลที่ได้จากการคิดต้นทุนการคำนวณ ดังนั้นจะกำหนดค่าต้นทุนการคำนวณ (c) เป็น 0.01 และ 0.05 ตามลำดับ ผลการทดลองแสดงในตารางที่ 6.3 และ 6.4

เมื่อกำหนดต้นทุนการคำนวณเป็น 0.01 พบว่าจำนวนครั้งในการประเมินค่าความเหมาะสมลดลงตั้งแต่ประมาณ 2% ถึง 90% โดยที่กลุ่มปัญหา 2 ชั้นแบบ 6 บิตเป็นกลุ่มปัญหาที่มีการลดการคำนวณลงมากที่สุด ทั้งนี้ก็เป็นเพราะว่ากลุ่มปัญหานี้มีค่าความเหมาะสมสูงสุดที่เป็นไปได้ต่ำกว่าปัญหาอื่น ดังนั้นในกรณีที่กำหนดให้ต้นทุนการคำนวณเท่ากัน คือ 0.01 หรือ

0.05 ต้นทุนนี้จึงนับเป็นค่ามากเมื่อสัมพันธ์กับค่าความเหมาะสมของปัญหาอื่น การทำงานต่อจึงไม่คุ้มกับค่าใช้จ่ายที่เกิดขึ้น จึงตัดสินใจหยุดในที่นี้

ในการเปรียบเทียบการคำนวณที่สามารถลดลงได้ ปัญหาจำนวนหนึ่งมากที่สุดขนาด 30 บิตเป็นปัญหาที่น่าสนใจสังเกต เนื่องจากในปัญหานี้ค่าความเหมาะสมที่ได้ทั้งจากการใช้วิธีหยุดแบบปกติ วิธีการหยุดโดยใช้ขอบเขตการใช้สิทธิทั้งในแบบที่ไม่พิจารณาต้นทุนการคำนวณและแบบที่มีต้นทุนการคำนวณเข้ามาเกี่ยวข้อง ค่าความเหมาะสมจากทั้ง 3 วิธีดังกล่าวเท่ากันทั้งหมด สิ่งที่น่าสนใจคือ เมื่อใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิโดยไม่มีต้นทุนการคำนวณ ขั้นตอนวิธีเชิงพันธุกรรมสามารถลดการคำนวณลงได้ 0.74% (จากตารางที่ 6.1) และเมื่อใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิที่มีต้นทุนการคำนวณ 0.01 ขั้นตอนวิธีเชิงพันธุกรรมสามารถลดการคำนวณลงได้ 11.35% (จากตารางที่ 6.3)

ตารางที่ 6.3 การหยุดโดยใช้ขอบเขตการใช้สิทธิที่มีต้นทุนการคำนวณ 0.01 โดยที่ Δ fitness คือผลต่างของค่าความเหมาะสมเฉลี่ยที่ได้รับซึ่งก็คือคุณภาพคำตอบที่เพิ่มขึ้นหรือลดลง และ Δ eval คือผลต่างของจำนวนครั้งการประเมินคำตอบซึ่งก็คือต้นทุนการคำนวณที่ประหยัดได้

Problem	sGA				sGA with option-based stopping policy				Saving analysis	
	Fitness			# Evaluation	Fitness			# Evaluation	Δ fitness (of avg.)	saving Δ eval
	Best	Avg.	S.D.		Best	Avg.	S.D.			
onemax30	30.00	29.85	0.37	1,095,550	30.00	29.85	0.38	971,200	0.00%	11.35%
onemax50	50.00	48.67	1.09	1,538,450	50.00	48.11	0.97	1,193,700	-1.16%	22.41%
onemax100	97.00	89.65	2.60	2,096,100	93.00	88.38	2.13	1,616,550	-1.44%	22.88%
trap3_10	30.00	28.85	0.95	1,213,450	30.00	28.78	1.00	1,069,200	-0.24%	11.89%
trap3_20	60.00	53.81	2.13	1,668,700	59.00	53.58	2.16	1,450,200	-0.43%	13.09%
trap3_30	84.00	75.71	3.40	1,929,300	85.00	75.46	3.13	1,723,200	-0.33%	10.68%
trap5_10	49.00	43.68	1.85	1,551,700	49.00	43.42	1.72	1,335,500	-0.60%	13.93%
trap5_20	90.00	78.55	3.57	2,020,950	90.00	78.20	3.56	1,798,700	-0.45%	11.00%
trap5_30	124.00	108.57	4.75	2,240,250	120.00	107.92	4.51	1,971,800	-0.60%	11.98%
Htrap27	81.00	59.25	12.13	1,151,650	81.00	58.57	12.38	1,123,800	-1.16%	2.42%
Htrap81	184.50	121.87	14.31	1,779,950	199.50	120.31	13.79	1,512,950	-1.30%	15.00%
Htrap243	331.50	251.35	18.41	2,385,550	298.50	249.36	17.03	2,179,150	-0.80%	8.65%
HIFF32	192.00	143.63	21.57	1,188,350	192.00	143.14	21.58	1,157,300	-0.34%	2.61%
HIFF64	352.00	242.32	25.77	1,625,600	336.00	241.79	24.98	1,558,450	-0.22%	4.13%
HIFF128	520.00	402.14	27.72	2,095,050	496.00	399.50	25.18	1,869,300	-0.66%	10.78%
6-bipolar30	5.00	4.78	0.10	1,788,950	4.80	4.56	0.08	151,050	-4.82%	91.56%
6-bipolar60	9.90	9.37	0.14	1,948,550	9.40	8.94	0.11	242,550	-4.81%	87.55%
6-bipolar90	14.40	13.93	0.16	1,909,250	14.00	13.26	0.14	292,200	-5.05%	84.70%

การทดลองโดยเพิ่มต้นทุนการคำนวณเป็น 0.05 ผลลัพธ์แสดงในตารางที่ 6.4 เมื่อต้นทุนในการคำนวณสูงขึ้น การใช้เงื่อนไขการหยุดจากขอบเขตการใช้สิทธิสามารถลดการคำนวณลงได้มากขึ้นด้วย เพราะเมื่อต้นทุนการคำนวณแพงมากจนไม่คุ้มที่จะดำเนินการต่อ ขั้นตอนวิธีเชิงพันธุกรรมก็จะยิ่งตัดสินใจหยุดเร็วขึ้น ส่วนใหญ่แล้วอัลกอริทึมสามารถลดการคำนวณลงได้มากกว่า 60% ในปัญหาทดสอบชุดนี้ ยกเว้นในกลุ่มปัญหากับดักแบบลำดับชั้น และปัญหาลำดับชั้นแบบกึ่งต่อเนื่อง ทั้ง 2 กลุ่มปัญหานี้เป็นปัญหายากและค่าความเหมาะสมก็สูงกว่าปัญหาอื่นๆ ดังนั้นถึงแม้ว่าต้นทุนการคำนวณจะสูงขึ้นแล้วก็ตาม ปัญหานี้ก็ยังต้องการการคำนวณที่มากขึ้นเพื่อที่จะหาคำตอบ

ตารางที่ 6.4 การหยุดโดยใช้ขอบเขตการใช้สิทธิที่มีต้นทุนการคำนวณ 0.05 โดยที่ Δ fitness คือ ผลต่างของค่าความเหมาะสมเฉลี่ยที่ได้รับซึ่งก็คือคุณภาพคำตอบที่เพิ่มขึ้นหรือลดลง และ Δ eval คือผลต่างของจำนวนครั้งการประเมินคำตอบซึ่งก็คือต้นทุนการคำนวณที่ประหยัดได้

Problem	sGA				sGA with option-based stopping policy				Saving analysis	
	Fitness			# Evaluation	Fitness			# Evaluation	Δ fitness saving (of avg.)	Δ eval
	Best	Avg.	S.D.		Best	Avg.	S.D.			
onemax30	30.00	29.85	0.37	1,095,550	30.00	22.89	1.14	151,200	-30.41%	86.20%
onemax50	50.00	48.67	1.09	1,538,450	49.00	35.28	1.64	151,450	-37.95%	90.16%
onemax100	97.00	89.65	2.60	2,096,100	89.00	73.09	2.39	438,700	-22.66%	79.07%
trap3_10	30.00	28.85	0.95	1,213,450	28.00	19.89	1.58	151,100	-45.05%	87.55%
trap3_20	60.00	53.81	2.13	1,668,700	51.00	35.48	2.28	201,350	-51.66%	87.93%
trap3_30	84.00	75.71	3.40	1,929,300	80.00	60.21	3.11	542,700	-25.74%	71.87%
trap5_10	49.00	43.68	1.85	1,551,700	44.00	28.68	1.97	151,400	-52.30%	90.24%
trap5_20	90.00	78.55	3.57	2,020,950	84.00	65.50	3.58	637,850	-19.92%	68.44%
trap5_30	124.00	108.57	4.75	2,240,250	113.00	94.81	5.53	838,200	-14.51%	62.58%
Htrap27	81.00	59.25	12.13	1,151,650	81.00	59.36	12.38	1,100,350	0.19%	4.45%
Htrap81	184.50	121.87	14.31	1,779,950	192.00	119.82	15.06	1,418,850	-1.71%	20.29%
Htrap243	331.50	251.35	18.41	2,385,550	307.50	246.48	18.36	1,832,400	-1.98%	23.19%
HIFF32	192.00	143.63	21.57	1,188,350	192.00	143.66	22.22	1,152,250	0.02%	3.04%
HIFF64	352.00	242.32	25.77	1,625,600	336.00	242.86	25.71	1,503,700	0.22%	7.50%
HIFF128	520.00	402.14	27.72	2,095,050	494.00	399.50	28.21	1,880,700	-0.66%	10.23%
6-bipolar30	5.00	4.78	0.10	1,788,950	4.80	4.56	0.07	151,450	-4.82%	91.53%
6-bipolar60	9.90	9.37	0.14	1,948,550	9.30	8.86	0.12	151,800	-5.76%	92.21%
6-bipolar90	14.40	13.93	0.16	1,909,250	14.10	13.06	0.15	151,700	-6.66%	92.05%

ผลการนับจำนวนครั้งที่ขั้นตอนวิธีเชิงพันธุกรรมหยุดการทำงานจากเงื่อนไขเส้นแบ่งด้านบนและด้านล่างในตารางที่ 6.5 สนับสนุนเหตุผลว่าทำไมเมื่อเพิ่มต้นทุนการคำนวณแล้วจึงสามารถลดจำนวนครั้งการประเมินค่าความเหมาะสมลงได้ ตารางที่ 6.5 แสดงจำนวนครั้งที่อัลกอริทึมหยุดจากเงื่อนไขดังกล่าวจากจำนวนครั้งการทำงานทั้งสิ้น 1,000 ครั้ง เมื่อเพิ่มต้นทุนการคำนวณให้สูงขึ้น อัลกอริทึมก็จะหยุดการทำงานมากขึ้นด้วย สังเกตว่าในบางปัญหา

อัลกอริทึมจะไม่ทำงานรอบที่ 2 ต่อเลย การทำงาน 999 ครั้งจะตัดสินใจหยุดเพราะเส้นแบ่งด้านบนเป็นเงื่อนไขที่บอกว่าควรหยุดการทำงานเนื่องจากต้นทุนการคำนวณแพงมากไม่คุ้มที่จะดำเนินการต่อ

ตารางที่ 6.6 แสดงค่าเฉลี่ยและค่าความแปรปรวนของจำนวนรุ่นที่อัลกอริทึมตัดสินใจหยุดการทำงาน โดยค่าในตารางจะประกอบด้วยผลการหยุดจากขั้นตอนวิธีเชิงพันธุกรรมที่ใช้เงื่อนไขการหยุดแบบปกติ ขั้นตอนวิธีเชิงพันธุกรรมที่ใช้วิธีการหยุดแบบค่อยเป็นค่อยไปที่ไม่คิดต้นทุนการคำนวณ และแบบที่มีต้นทุนการคำนวณ 0.01 และ 0.05 ตามลำดับ จากผลการทดลองที่แสดงในตารางดังกล่าวจะเห็นได้ว่าขั้นตอนวิธีเชิงพันธุกรรมจะหยุดการทำงานเร็วขึ้นเมื่อใช้วิธีการหยุดแบบค่อยเป็นค่อยไป ซึ่งดูได้จากค่าเฉลี่ยที่ลดลงของจำนวนรุ่นที่อัลกอริทึมหยุดทำงาน การที่อัลกอริทึมหยุดทำงานเร็วขึ้นนั้นก็หมายถึงจำนวนครั้งในการคำนวณที่ลดลงด้วย

ตารางที่ 6.5 จำนวนครั้งการหยุดในแต่ละรุ่นของการใช้เงื่อนไขการหยุดแบบค่อยเป็นค่อยไป

Problem	No cost		Cost = 0.01		Cost = 0.05	
	Stop over upper threshold	Stop under lower threshold	Stop over upper threshold	Stop under lower threshold	Stop over upper threshold	Stop under lower threshold
onemax30	159	0	974	0	999	0
onemax50	178	0	972	0	999	0
onemax100	183	28	896	46	999	0
trap3_10	267	3	963	1	999	0
trap3_20	170	14	680	173	999	0
trap3_30	300	20	692	83	994	5
trap5_10	221	1	881	34	999	0
trap5_20	461	21	533	216	964	35
trap5_30	560	44	663	90	670	329
Htrap27	183	72	133	111	218	139
Htrap81	133	249	46	799	226	719
Htrap243	474	66	391	136	292	656
HIFF32	129	165	147	58	140	202
HIFF64	505	14	299	70	250	264
HIFF128	509	33	584	32	423	145
6-bipolar30	464	1	999	0	999	0
6-bipolar60	494	7	999	0	999	0
6-bipolar90	378	9	999	0	999	0

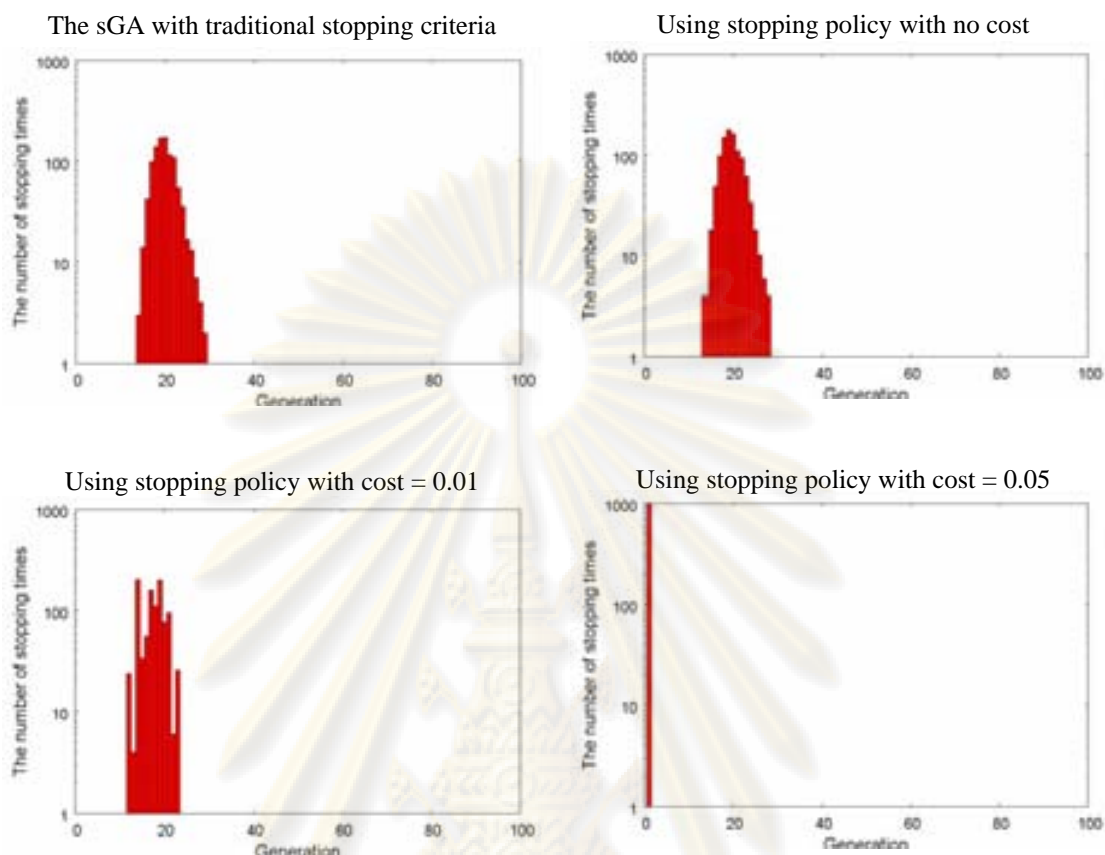
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 6.6 ค่าเฉลี่ยและค่าความแปรปรวนของจำนวนรุ่นที่อัลกอริทึมตัดสินใจหยุดทำงาน

Problem	sGA		No cost		Cost = 0.01		Cost = 0.05	
	Mean	Variance	Mean	Variance	Mean	Variance	Mean	Variance
onemax30	19.91	6.10	19.75	5.97	17.42	6.76	1.02	0.57
onemax50	28.77	13.75	28.59	13.32	21.87	10.40	1.03	0.84
onemax100	39.92	32.61	39.07	32.75	30.33	8.88	6.77	1.46
trap3_10	22.27	11.52	22.11	11.39	19.38	6.03	1.02	0.48
trap3_20	31.37	20.54	30.98	17.14	27.00	9.15	2.03	0.73
trap3_30	36.59	28.30	35.44	21.22	32.46	11.98	8.85	1.39
trap5_10	29.03	18.29	28.64	16.86	24.71	11.94	1.03	0.78
trap5_20	38.42	32.17	36.13	26.34	33.97	14.60	10.76	2.49
trap5_30	42.81	40.22	38.39	30.52	37.44	19.33	14.76	3.89
Htrap27	21.03	16.59	20.59	12.95	20.48	14.56	20.01	13.05
Htrap81	33.60	24.74	31.87	21.11	28.26	8.19	26.38	10.95
Htrap243	45.71	46.16	41.10	54.85	41.58	38.40	34.65	19.12
HIFF32	21.77	16.30	20.93	10.54	21.15	12.46	21.05	8.74
HIFF64	30.51	19.16	28.35	19.18	29.17	15.19	28.07	17.71
HIFF128	39.90	32.29	35.72	33.01	35.39	30.82	35.61	29.95
6-bipolar30	33.78	164.04	31.40	70.58	1.02	0.44	1.03	0.84
6-bipolar60	36.97	161.97	32.94	53.58	2.85	1.00	1.04	1.29
6-bipolar90	36.19	90.41	33.76	40.44	3.84	0.69	1.03	1.15

กราฟในรูปที่ 6.3 แสดงให้เห็นการลดการคำนวณในปัญหาจำนวนหนึ่งมากที่สุดขนาด 30 บิต ว่าเกิดขึ้นได้เพราะเหตุใด ในรูปจะประกอบด้วยกราฟ 4 กราฟที่แสดงจำนวนครั้งการหยุดในแต่ละรุ่นของการทำงานทั้งสิ้น 1,000 ครั้ง โดยกราฟในแกนตั้งแสดงเป็นมาตราส่วนลอการิทึม (Log scale) กราฟด้านบนซ้ายแสดงการกระจายตัวของการหยุดในรุ่นต่าง ๆ ของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้เงื่อนไขการหยุดแบบทั่วไป กราฟทางด้านบนขวาแสดงกราฟจากการใช้เงื่อนไขการหยุดแบบค่อยเป็นค่อยไปที่ไม่มีต้นทุนการคำนวณ กราฟด้านล่างทางซ้ายและขวาแสดงจำนวนครั้งการหยุดจากการใช้เงื่อนไขการหยุดแบบค่อยเป็นค่อยไปโดยที่มีต้นทุนการคำนวณ 0.01 และ 0.05 ตามลำดับ ส่วนกราฟการหยุดของปัญหาทดสอบทั้งหมดซึ่งมีลักษณะคล้ายกันจะแสดงไว้ในภาคผนวก ก

จากกราฟในรูปที่ 6.3 จะเห็นได้ว่าเมื่อมีการใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิ กราฟจะมีลักษณะเคลื่อนไปทางซ้าย และเมื่อมีการพิจารณาต้นทุนการทำงานเข้ามาเกี่ยวข้อง การกระจายตัวของการหยุดก็จะยิ่งเลื่อนมาด้านซ้ายมากขึ้น จากกราฟการกระจายตัวของการหยุดการทำงานในรุ่นต่าง ๆ ทำให้เห็นว่าอัลกอริทึมจะหยุดการทำงานเร็วขึ้น จำนวนครั้งการหยุดในรุ่นแรก ๆ จะสูงขึ้น และเป็นเหตุผลว่าทำไมขั้นตอนวิธีเชิงพันธุกรรมที่ใช้เงื่อนไขการหยุดตามขอบเขตการใช้สิทธิจึงสามารถลดการคำนวณลงได้ และเมื่อต้นทุนการคำนวณสูงมาก ๆ อัลกอริทึมก็จะหยุดการทำงานในรุ่นแรกทั้งหมด เนื่องจากการหาค่าตอบไม่คุ้มกับต้นทุนการคำนวณที่ต้องเสียไป



รูปที่ 6.3 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาจำนวนหนึ่งมากที่สุดขนาด 30 บิต

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 7

สรุปผลการวิจัย

วิทยานิพนธ์นี้นำเสนอการหาเวลาหยุดที่เหมาะสมที่สุดของขั้นตอนวิธีเชิงพันธุกรรมโดยใช้วิธีอุปชันจริง การใช้วิธีอุปชันจริงในการตัดสินใจลงทุนได้มีการนำไปใช้กันอย่างแพร่หลายและประสบผลสำเร็จในแง่การบริหารสินทรัพย์ภายใต้ความไม่แน่นอน การนำมาประยุกต์ใช้ในการตัดสินใจทำงานของอัลกอริทึมเป็นสิ่งใหม่ที่น่าสนใจในวิทยานิพนธ์นี้ เราสามารถมองการตัดสินใจหยุดหรือทำงานต่อของขั้นตอนวิธีเชิงพันธุกรรมเทียบได้กับการตัดสินใจลงทุน การดำเนินการต่อของอัลกอริทึมเปรียบเสมือนการรอโอกาสที่เหมาะสมที่จะลงทุน และการตัดสินใจหยุดคือการยอมรับคำตอบที่หามาได้เช่นเดียวกับการตัดสินใจลงทุนที่ยุติการดูแลโน้มของตลาดและตัดสินใจรับผลประโยชน์ในขณะนี้ วิธีอุปชันจริงจะให้เกณฑ์ที่เหมาะสมในการตัดสินใจดังกล่าว ซึ่งในที่นี้ผลจากการวิเคราะห์เวลาหยุดของขั้นตอนวิธีเชิงพันธุกรรมโดยใช้วิธีอุปชันจริงจะให้ผลลัพธ์ 2 อย่าง คือ มูลค่าอัลกอริทึมและขอบเขตการใช้สิทธิ

มูลค่าอัลกอริทึมคือค่าที่จะได้รับการหาคำตอบโดยใช้ขั้นตอนวิธีเชิงพันธุกรรมภายใต้การตัดสินใจที่ดีที่สุด ซึ่งการตัดสินใจนี้หมายถึงการตัดสินใจภายใต้ความไม่แน่นอนของการได้มาซึ่งคำตอบ การตัดสินใจที่เหมาะสมนี้ยังคำนึงถึงต้นทุนในการคำนวณและความสำคัญว่าเวลาที่จะได้รับคำตอบเร็วช้าอย่างไร มูลค่าอัลกอริทึมที่คำนวณได้นี้จะถูกนำมาใช้เปรียบเทียบอัลกอริทึมต่าง ๆ ว่าวิธีใดมีค่าคาดหวังที่จะได้รับคำตอบที่มีมูลค่าสูงกว่ากันเมื่อคำนึงถึงปัจจัยความไม่แน่นอนและต้นทุนการคำนวณต่าง ๆ การวิเคราะห์มูลค่าอัลกอริทึมของขั้นตอนวิธีประมาณการแจกแจงแบบที่ตัวแปรไม่ขึ้นต่อกัน ได้นำเสนอไปแล้วในบทที่ 4 ซึ่งเป็นตัวอย่างของการนำวิธีอุปชันจริงไปใช้ในการศึกษาการทำงานของอัลกอริทึมต่าง ๆ เนื้อหาในส่วนนี้ยังได้ตีพิมพ์ในวารสารวิชาการซึ่งสามารถอ่านเพิ่มเติมได้จาก (Rimcharoen et al., 2009)

ขอบเขตการใช้สิทธิคือแผนภาพแสดงเส้นแบ่งการตัดสินใจที่เหมาะสมที่สุด ค่าที่อยู่ในแต่ละด้านของเส้นแบ่งนี้หมายถึงการตัดสินใจที่ต่างกัน ในที่นี้เส้นแบ่งจะบอกวิธีการตัดสินใจที่เหมาะสมที่สุดของอัลกอริทึมว่าควรดำเนินการต่อหรือหยุดการทำงาน โดยพิจารณาจากค่าความเหมาะสมที่ได้รับในขณะนั้นว่าสูงหรือต่ำกว่าเส้นแบ่งนี้ ผลการทดลองใช้เงื่อนไขการหยุดตามค่าขอบเขตการใช้สิทธิได้นำเสนอไปในบทที่ 5 และ 6 เมื่อขั้นตอนวิธีเชิงพันธุกรรมใช้เงื่อนไขการหยุดดังกล่าวพบว่าสามารถลดจำนวนครั้งในการประเมินคำตอบลงได้ นั่นหมายถึงการตัดสินใจหยุดอย่างเหมาะสมทำให้สามารถลดการคำนวณลงได้

การนำวิธีอุปชันจริงไปใช้ในขั้นตอนวิธีเชิงวิวัฒน์แบบอื่น ๆ สามารถทำได้โดยการ
 เพิ่มขึ้นตอนการคำนวณมูลค่าอัลกอริทึมเข้าไปในอัลกอริทึมโดยไม่ต้องเปลี่ยนแปลง
 กระบวนการหลักของวิธีนั้น ๆ เพียงแต่เพิ่มขึ้นตอนการเก็บความเปลี่ยนแปลงค่าความ
 เหมาะสมในแต่ละรุ่นไว้แล้วนำมาคำนวณมูลค่าอัลกอริทึมและสร้างขอบเขตการใช้สิทธิเมื่อ
 อัลกอริทึมทำงานครบรอบ วิธีนี้สามารถประยุกต์ใช้ได้กับทุกอัลกอริทึมที่มีค่าความเหมาะสมใน
 แต่ละรุ่น โดยไม่สนใจวิธีการหาคำตอบของขั้นตอนวิธีนั้น ๆ เลย เราเพียงแต่นำค่าความ
 เหมาะสมที่ได้จากการประเมินคำตอบที่ปรับปรุงมาได้มาใช้ ดังนั้นวิธีนี้จึงเป็นอีกทางเลือก
 หนึ่งที่จะนำมาใช้วิเคราะห์พฤติกรรมการทำงานของอัลกอริทึมที่ซับซ้อน ที่อาจจะยากในการ
 วิเคราะห์แบบปกติ

ข้อเสนอแนะสำหรับงานวิจัยที่น่าจะศึกษาต่อ ได้แก่ การวิเคราะห์เวลาหยุดของ
 ขั้นตอนวิธีเชิงพันธุกรรมกับปัญหาจริง การศึกษาความแตกต่างระหว่างเวลาหยุดเหมาะสมสุดของ
 ขั้นตอนวิธีเชิงวิวัฒนาการแบบต่าง ๆ เป็นต้น



ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

- Aytug, H. and Koehler, G. J. New stopping criterion for genetic algorithm. European Journal of Operational Research, 126 (2000): 662-674.
- Aytug, H. and Koehler, G. J. Stopping criterion for finite length genetic algorithms. INFORMS Journal on Computing, 8 (1996): 183-191.
- Baluja, S. Population-based incremental learning: A method for integrating genetic search based function optimization and competitive learning. Technical Report CMU-CS-95-163, Carnegie Mellon University, 1994.
- Black, F. and Scholes, M. The pricing of options and corporate liabilities. Journal of Political Economy, 81 (1973): 637-654.
- Borenstein, Y. and Poli, R. Fitness distributions and GA hardness. In Proceedings of the 8th International Conference on Parallel Problem Solving from Nature, 2004.
- Borenstein, Y. and Poli, R. Information landscapes and problem hardness. In Proceedings of the 2005 Genetic and Evolutionary Computation Conference, 2005.
- Chen, S. H. and Lee, W. C. Option pricing with genetic algorithms: separating out-of-the-money from in-the-money, in Proceeding of the IEEE International Conference on Intelligent Processing Systems, 1997.
- Chen, S. H., Lee, W. C. and Yeh, C. H. Hedging derivative securities with genetic programming, International Journal of Intelligent Systems in Accounting, Finance and Management, 8 (1999): 237-251.
- Chen, S. H., Yeh, A. H. and Lee, W. C. Option pricing with genetic programming, in Proceeding of the Third Annual Genetic Programming Conference, 1998.
- Chidambaran, N. K. Genetic programming with monte carlo simulation for option pricing, in Proceeding of the 2003 Winter Simulation Conference, 2003.
- Chidambaran, N. K., Lee, C. H. J. and Trigueros, J. R. An adaptive evolutionary approach to option pricing via genetic programming, in Proceeding of the Third Annual Genetic Programming Conference, 1998.
- Cox, J. C., Ross, S. A. and Rubinstein, M. Option pricing: a simplified approach. Journal of Financial Economics, 7 (1979): 229-263.

- Darwin, C. R. On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life. London: John Murray, 1859.
- Davidor, Y. Epistasis variance: A viewpoint on GA-hardness. in Foundations of Genetic Algorithms, 1991.
- Fogel, L. J., Owens, A. J., and Walsh, M. J. Artificial intelligence through a simulation of evolution. In Biophysics and Cybernetic Systems: Proceedings of the 2nd Cybernetic Sciences Symposium, 1965.
- Fogel, L. J., Owens, A. J., and Walsh, M. J. Artificial Intelligence through Simulated Evolution. New York: John Wiley & Sons, 1966.
- Goldberg, D. E. Genetic Algorithms in Search, Optimization, and Machine Learning. Reading: Addison-Wesley, 1989.
- Goldberg, D. E. Simple genetic algorithms and the minimal deceptive problem, Genetic Algorithms and Simulated Annealing, Morgan Kaufmann Publisher, 1987.
- Harik, G. R., Lobo, F. G., and Goldberg, D. E. The compact genetic algorithm. IEEE Transaction on Evolutionary Computation, 3 (1999): 287-297.
- He, J. and Yao, X. Towards an analytic framework for analyzing the computation time of evolutionary algorithms, Artificial Intelligence, 145 (2003): 59-97.
- He, J., Reeves, C., Witt, C. and Yao, X. A note on problem difficulty measures in black-box optimization: Classification, realizations and predictability, Evolutionary Computation, 15 (2007): 435-443.
- Holland, J. H. Adaptation in Natural and Artificial Systems. Ann Arbor: The University of Michigan Press, 1975.
- Holland, J. H. Genetic algorithms and the optimal allocation of trials. SIAM Journal on Computing, 2 (1973): 88-105.
- Hulin, M. An optimal stop criterion for genetic algorithms: A bayesian approach, in Proceeding of the International conference on Genetic Algorithms, 1997.
- Jones, T. and Forrest, S. Fitness distance correlation as a measure of problem difficulty for genetic algorithms. in Proceedings 6th International Conference on Genetic Algorithms, 1995.
- Kauffman, S. The Origins of Order: Self-Organization and Selection in Evolution, Oxford University Press, Oxford, 1993.

- Meyer L. and Feng, X. A fuzzy stop criterion for genetic algorithms using performance estimation, In Proceedings of the Third IEEE Conference on Fuzzy Systems, 1994.
- Michalewicz, Z. Genetics Algorithms + Data Structures = Evolution Programs, New York: Springer-Verlag, 1996.
- Mühlenbein, H., and Paaß, G. From recombination of genes to the estimation of distributions I. Binary parameters. Parallel Problem Solving from Nature, LNCS 1141, Berlin: Springer, 1996.
- Nix, A. E. and Vose, M. D. Modeling genetic algorithms with Markov chains. Annals of Mathematics and Artificial Intelligence, 5 (1992): 79-88.
- Pelikan, M., and Goldberg, D. E. Escaping hierarchical traps with competent genetic algorithm, In Proceedings of the Genetic and Evolutionary Computation Conference, 2001.
- Pelikan, M., Goldberg, D. E., and Cantú-Paz, E. BOA: the bayesian optimization algorithm, In Proceedings of the Genetic and Evolutionary Computation Conference, 1999.
- Pelikan, M., Goldberg, D. E., and Lobo, F. G. A survey of optimization by building and using probabilistic models. Urbana, IL: University of Illinois Genetic Algorithms Laboratory (IlliGAL Report No. 99018), 1999.
- Pendharkar, P. C. and Koehler, G. J. A general steady state distribution based stopping criteria for finite length genetic algorithms. European Journal of Operational Research, 176 (2007): 1436-1451.
- Rechenberg, I. Cybernetic solution path of an experimental problem. Ministry of Aviation, Royal Aircraft Establishment, U.K., 1965.
- Rechenberg, I. Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. Frommann-Holzboog, 1971.
- Rimcharoen, S., Sutivong, D. and Chongstitvatana, P. Real options approach to evaluating genetic algorithms, Applied Soft Computing, 2009.
- Rudolph, G. Convergence Properties of Evolutionary Algorithms. Ph.D. Thesis (Kovač, Hamburg), 1997.
- Sandholm, T. Terminating decision algorithm optimally, in Proceeding of the International Conference on Principles and Practice of Constraint Programming, 2003.
- Schwefel, H. P. Numerische Optimierung von Computer-Modellen. Birkh"user, 1974.

- Watson, R. A., Hornby, G. S., and Pollack, J. B. Modeling building-block interdependency, Parallel Problem Solving from Nature – PPSN V, 1998.
- Yin, Z., Brabazon, A. and O'Sullivan, C. Adaptive genetic programming for option pricing. in Proceedings of the 2007 GECCO Conference Companion on Genetic and Evolutionary Computation, 2007.
- Zielinski, K., Peters D. and Laur, R. Stopping criteria for single-objective optimization, In Proceedings of the Third International Conference on Computational Intelligence, Robotics and Autonomous Systems, 2005.



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

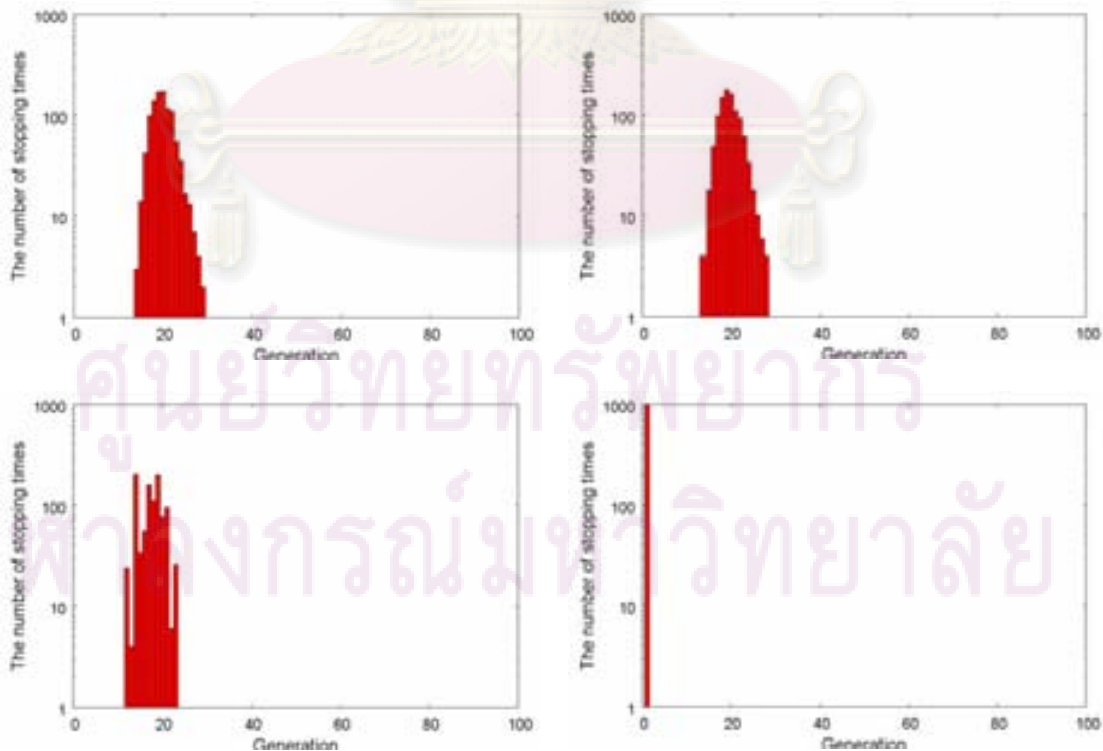


ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

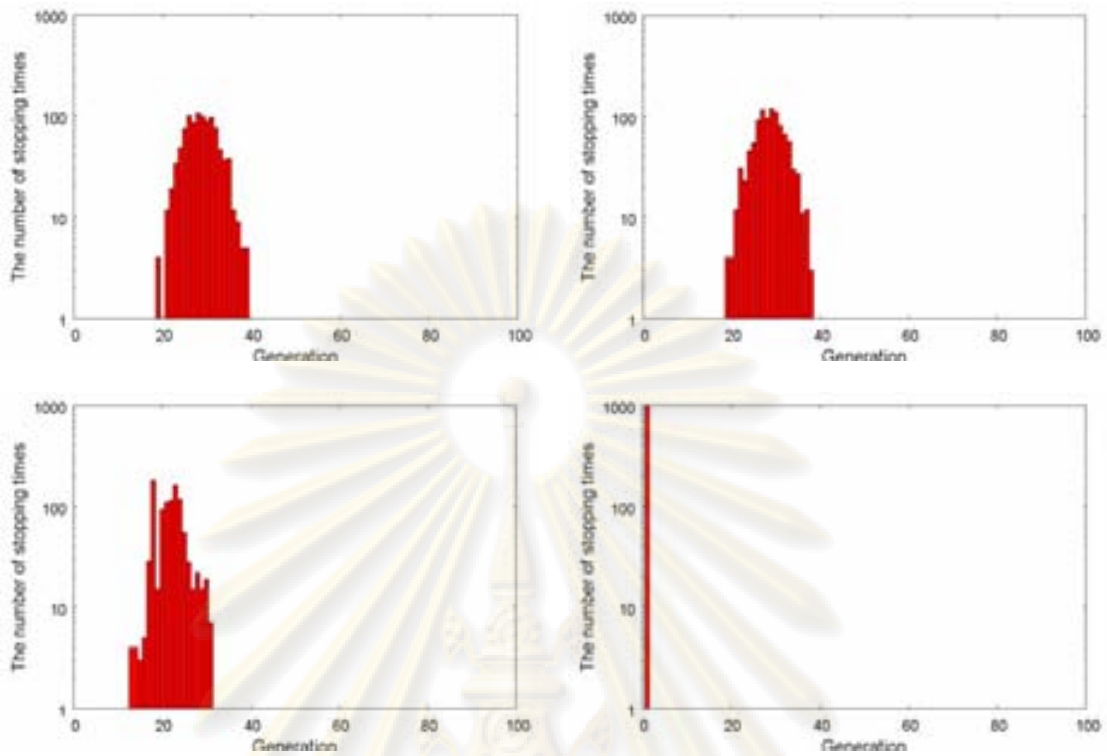
ภาคผนวก ก.

ในส่วนของภาคผนวกนี้ได้แสดงกราฟจำนวนครั้งการตัดสินใจหยุดของอัลกอริทึมที่หยุดในรุ่นต่าง ๆ เมื่อใช้เงื่อนไขการหยุดแบบค่อยเป็นค่อยไป กราฟการหยุดนี้จะเปลี่ยนแปลงไปเมื่อมีการคิดต้นทุนการคำนวณเข้ามาเกี่ยวข้อง ในการทดลองจะทดสอบขั้นตอนวิธีเชิงพันธุกรรมกับปัญหาทดสอบต่าง ๆ ดังนี้ ปัญหาจำนวนหนึ่งมากที่สุดขนาด 30, 50 และ 100 บิต ปัญหาภัยดับกขนาด 3 ต่อกัน 10 ชุด, 20 ชุด และ 30 ชุด ปัญหาภัยดับกขนาด 5 ต่อกัน 10 ชุด, 20 ชุด และ 30 ชุด ปัญหาภัยดับกแบบลำดับชั้นขนาด 27, 81 และ 243 บิต ปัญหาลำดับชั้นแบบกต่อเมื่อขนาด 32, 64 และ 128 บิต ปัญหา 2 ชั้นแบบ 6 บิต ขนาด 30, 60 และ 90 บิต ซึ่งจะแสดงในรูปที่ ก.1 – ก.18 ตามลำดับ

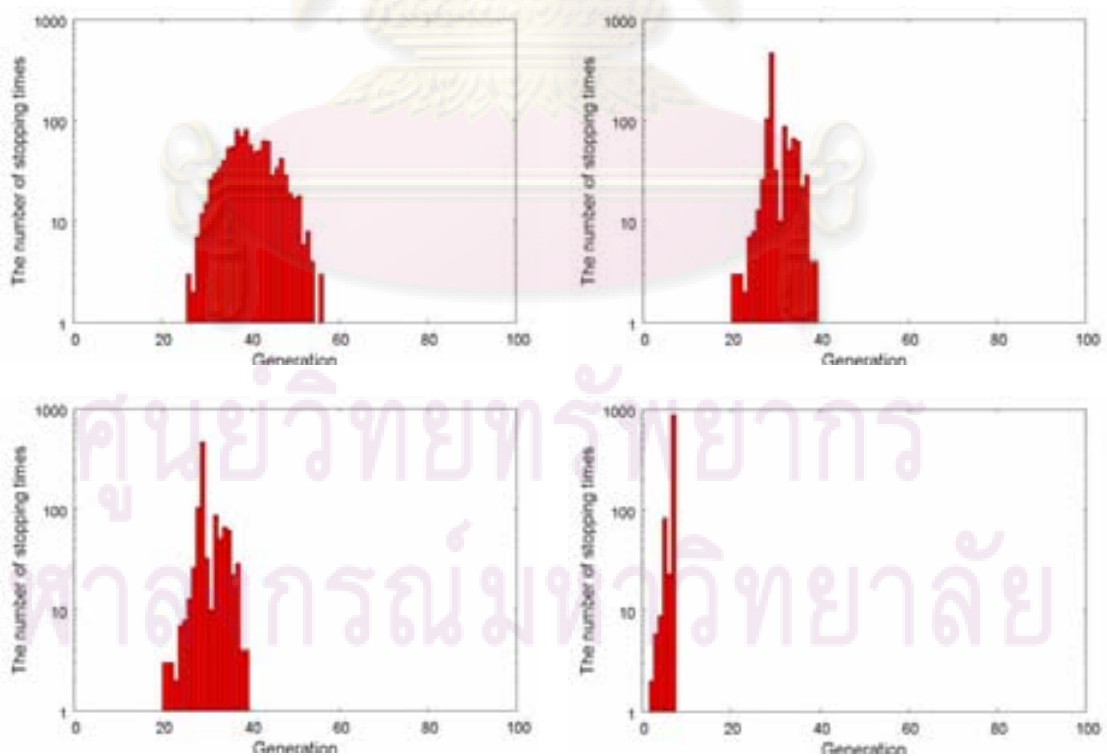
กราฟในแกนตั้งแสดงเป็นมาตราส่วนลอการิทึม (Log scale) โดยในแต่ละรูปจะประกอบด้วยกราฟ 4 กราฟที่แสดงจำนวนครั้งการหยุดในแต่ละรุ่นของการทำงานทั้งสิ้น 1,000 ครั้ง กราฟด้านบนซ้ายแสดงการกระจายตัวของการหยุดในรุ่นต่าง ๆ ของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายที่ใช้เงื่อนไขการหยุดแบบทั่วไป กราฟทางด้านบนขวาแสดงกราฟจากการใช้เงื่อนไขการหยุดแบบค่อยเป็นค่อยไปที่ไม่มีต้นทุนการคำนวณ กราฟด้านล่างทางซ้ายและขวาแสดงจำนวนครั้งการหยุดจากการใช้เงื่อนไขการหยุดแบบค่อยเป็นค่อยไปโดยที่มีต้นทุนการคำนวณ 0.01 และ 0.05 ตามลำดับ



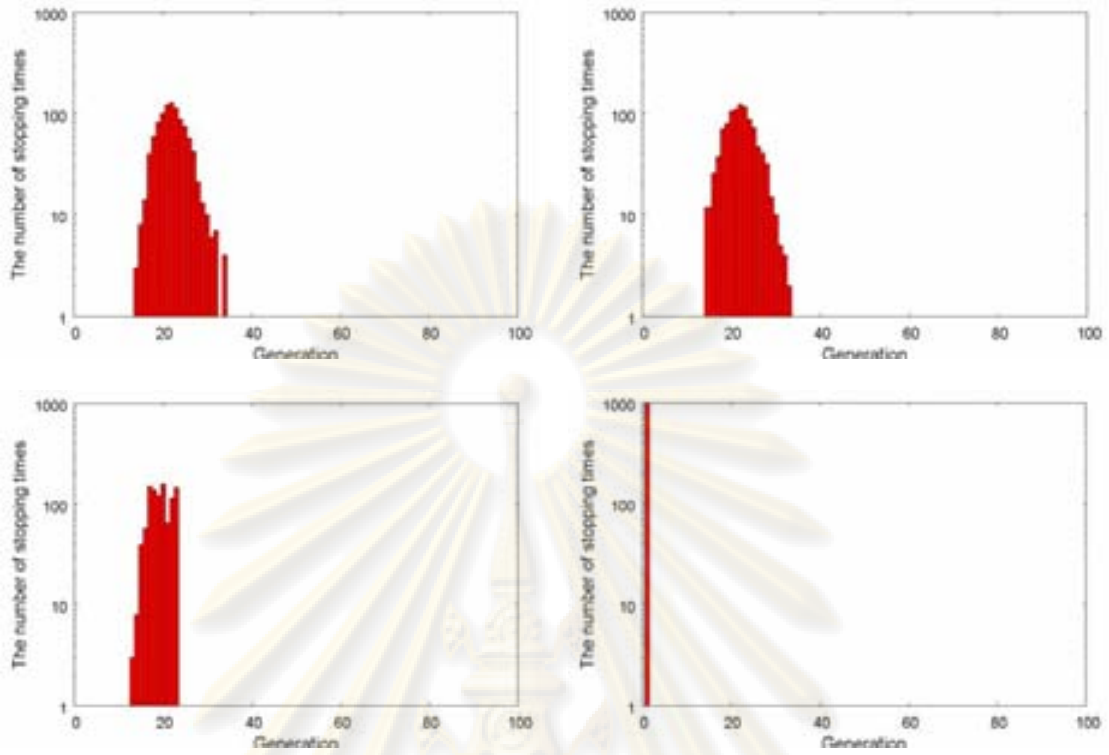
รูปที่ ก.1 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาจำนวนหนึ่งมากที่สุดขนาด 30 บิต



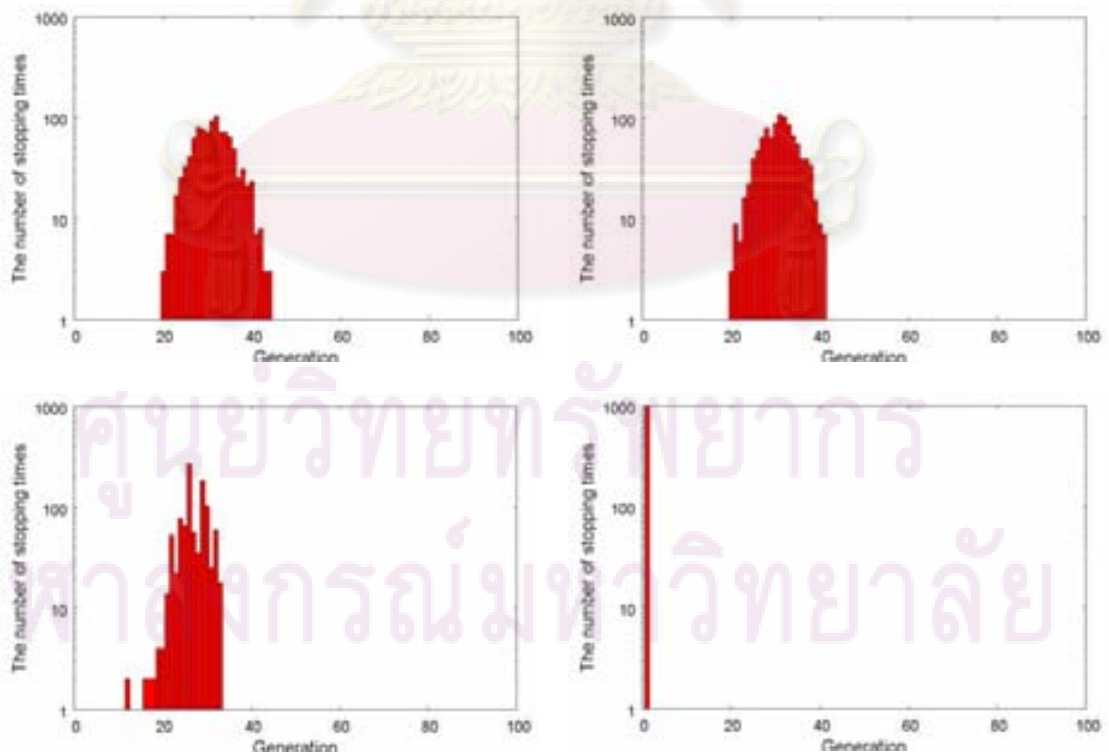
รูปที่ ก.2 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาจำนวนหนึ่งมากที่สุดขนาด 50 บิต



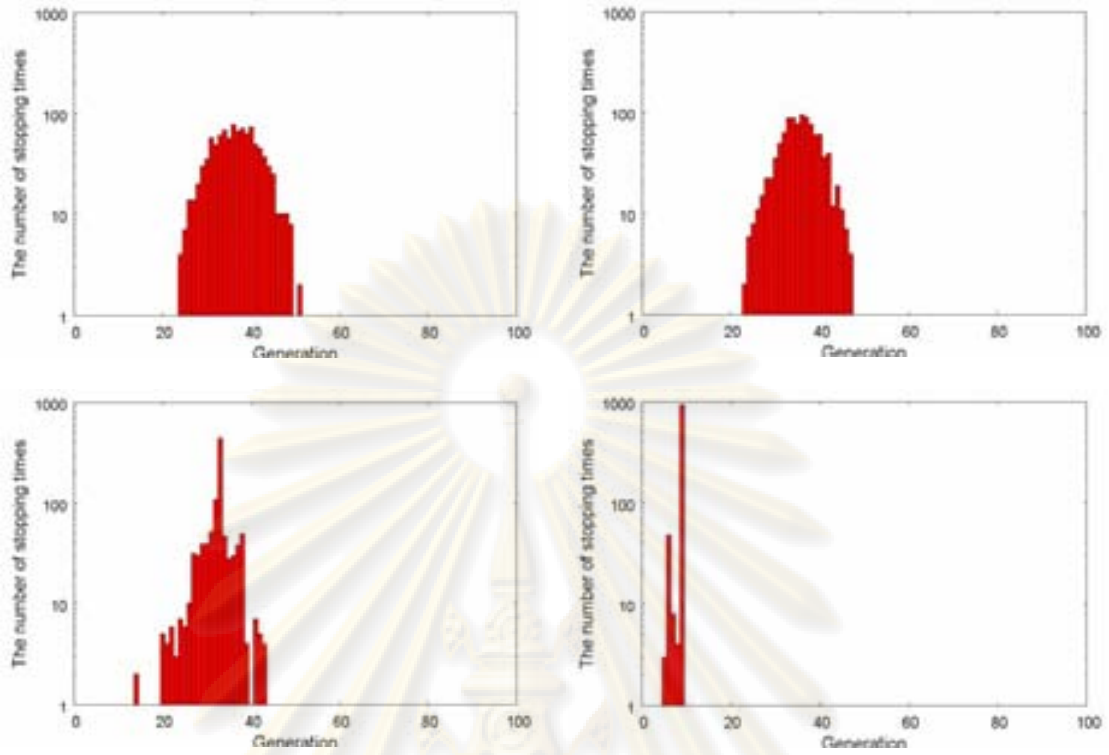
รูปที่ ก.3 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาจำนวนหนึ่งมากที่สุดขนาด 100 บิต



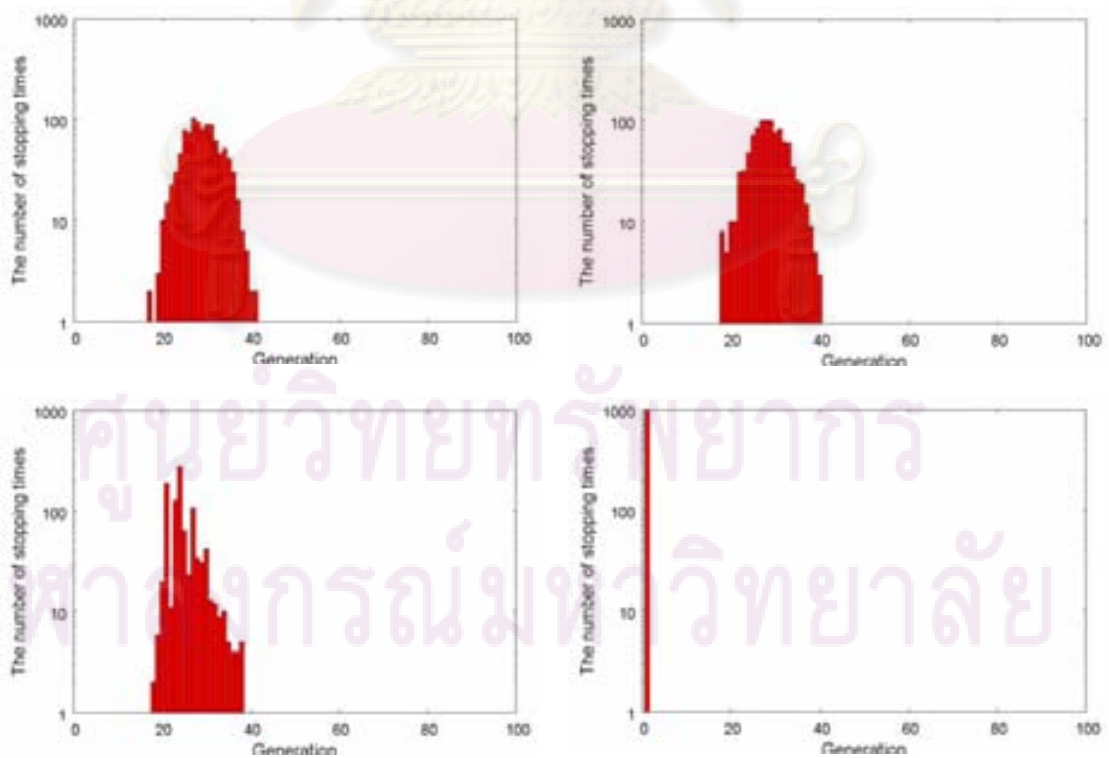
รูปที่ ก.4 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับดัชนีขนาด 3 ต่อกัน 10 ชุด



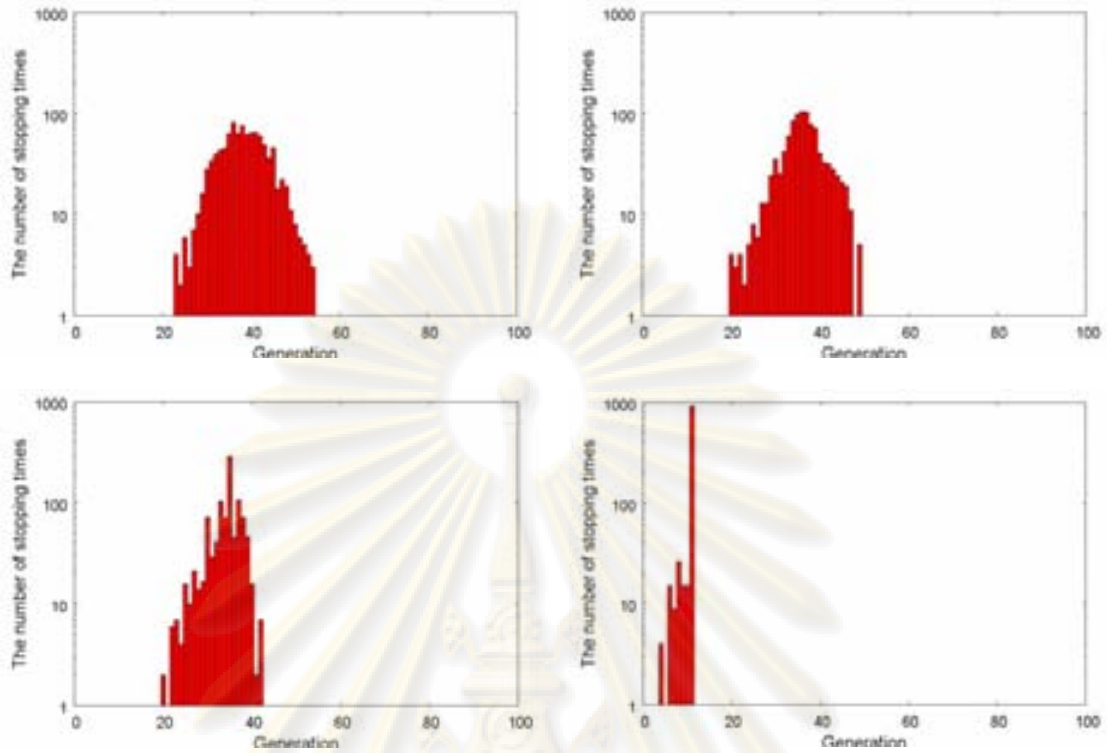
รูปที่ ก.5 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับดัชนีขนาด 3 ต่อกัน 20 ชุด



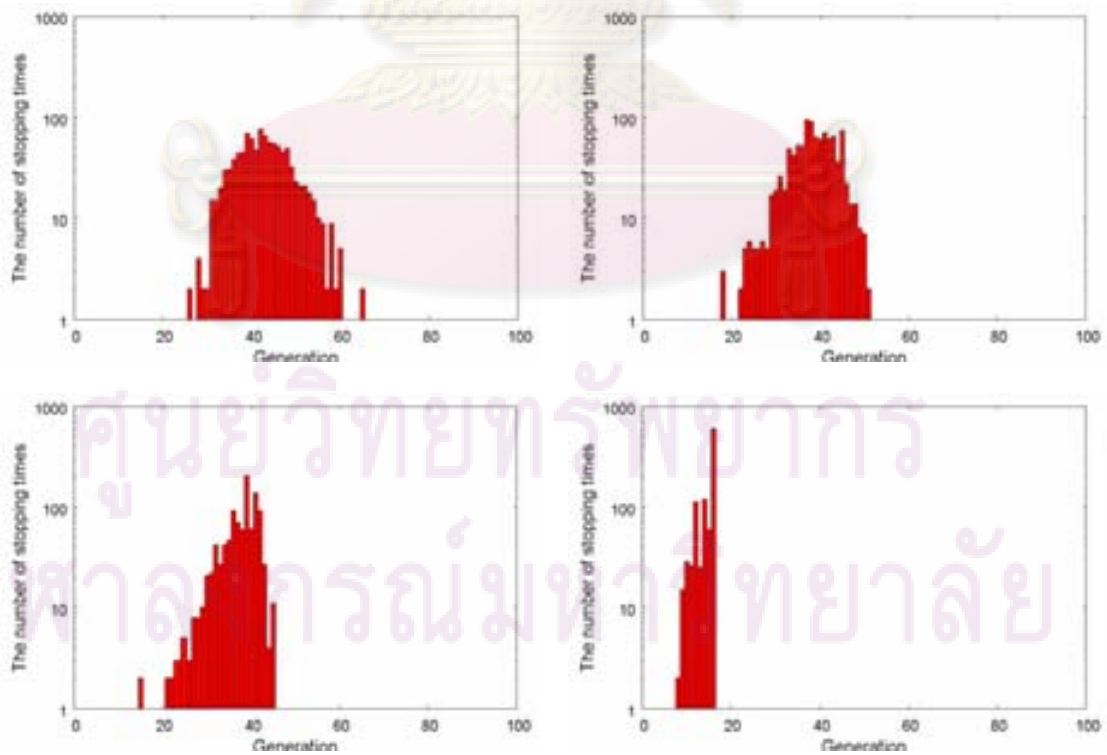
รูปที่ ก.6 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับดัชนีขนาด 3 ต่อกัน 30 ชุด



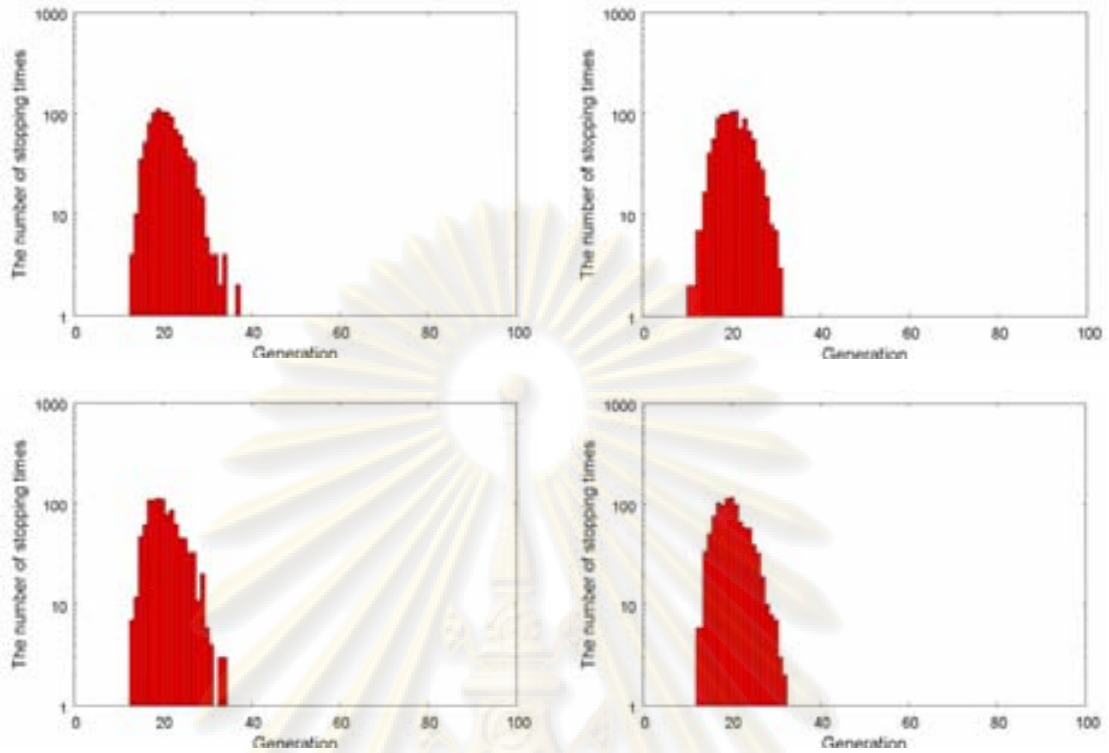
รูปที่ ก.7 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับดัชนีขนาด 5 ต่อกัน 10 ชุด



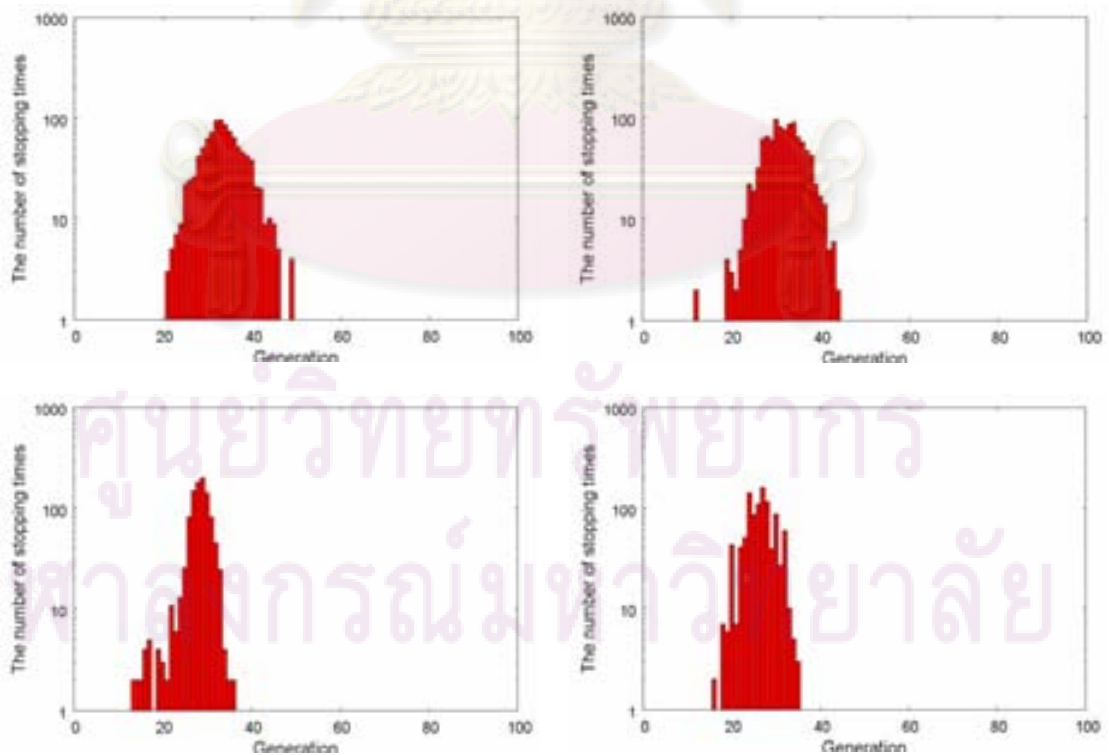
รูปที่ ก.8 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับดัชนีขนาด 5 ต่อกัน 20 ชุด



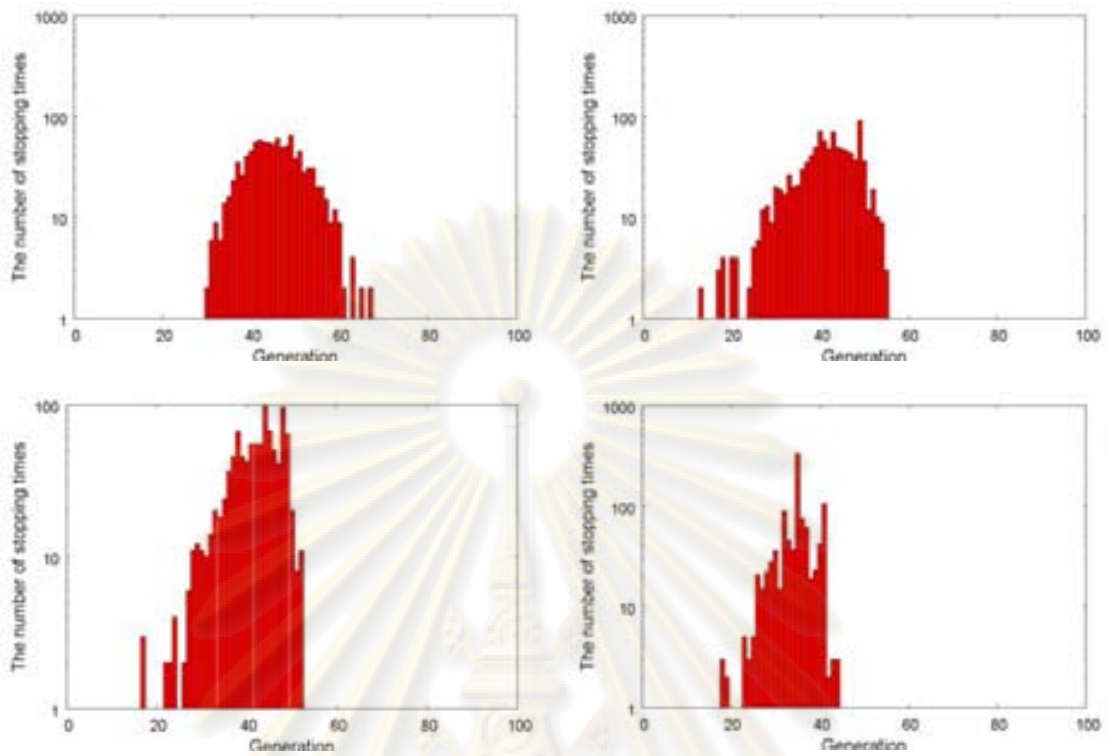
รูปที่ ก.9 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับดัชนีขนาด 5 ต่อกัน 30 ชุด



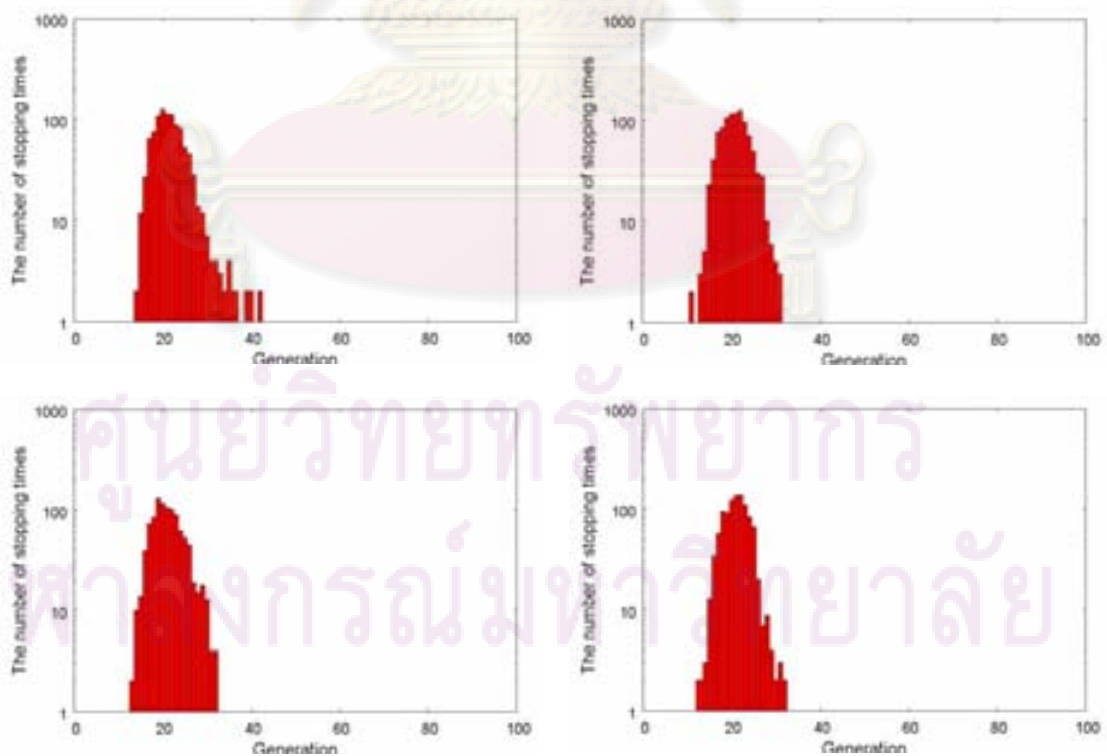
รูปที่ ก.10 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับดักแบบลำดับชั้นขนาด 27 บิต



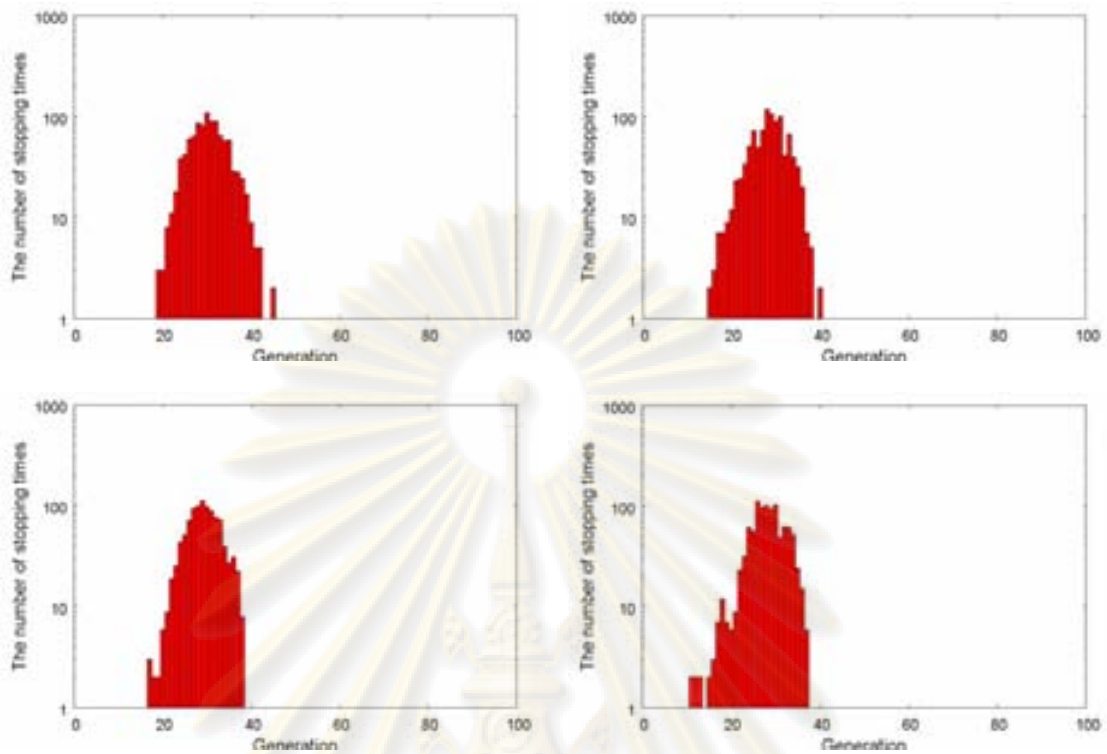
รูปที่ ก.11 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับดักแบบลำดับชั้นขนาด 81 บิต



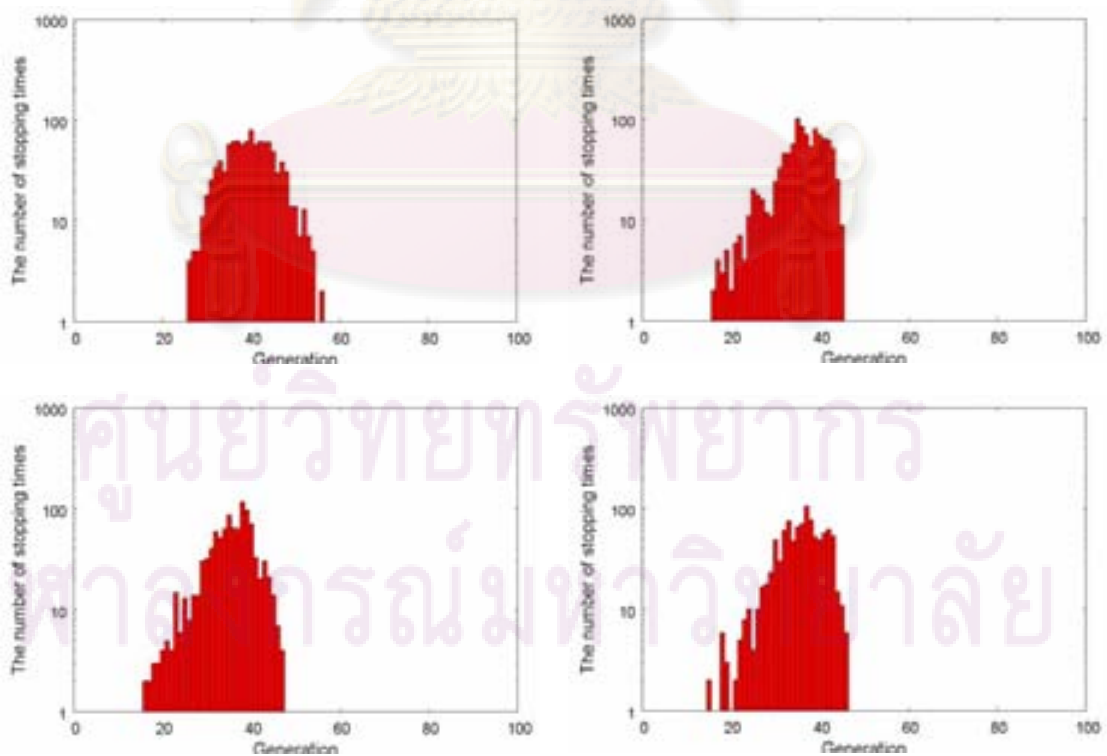
รูปที่ ก.12 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหากับดักแบบลำดับชั้นขนาด 243 บิต



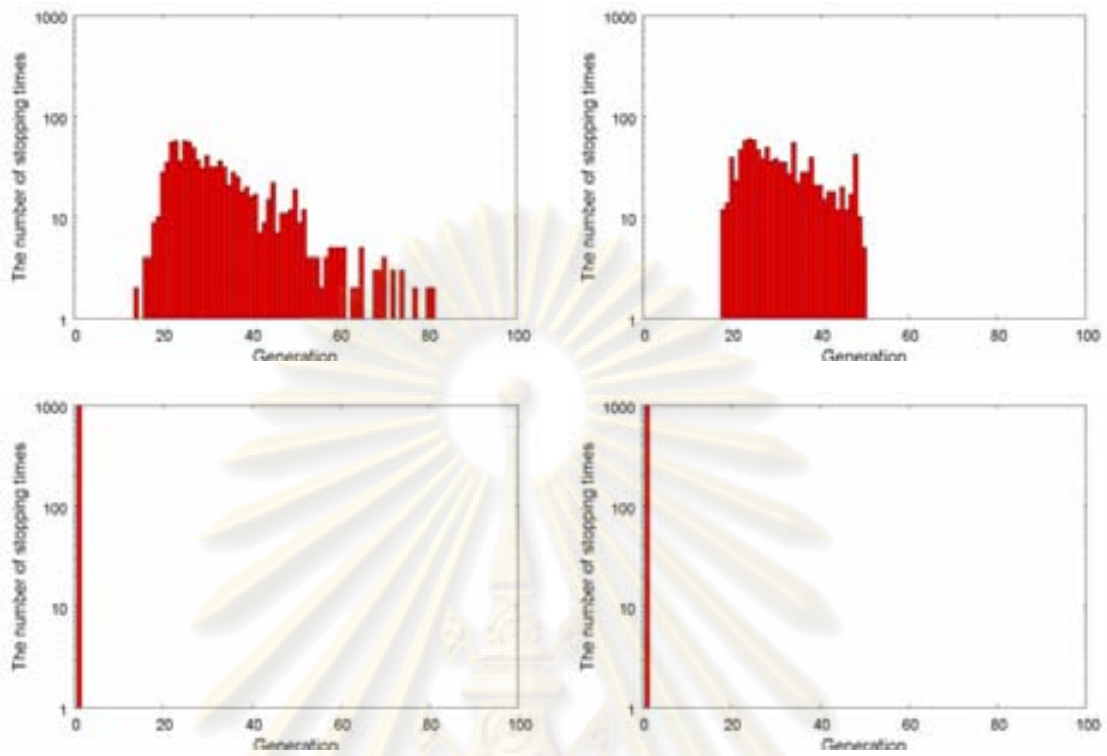
รูปที่ ก.13 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาลำดับชั้นแบบกึ่งต่อเนื่องขนาด 32 บิต



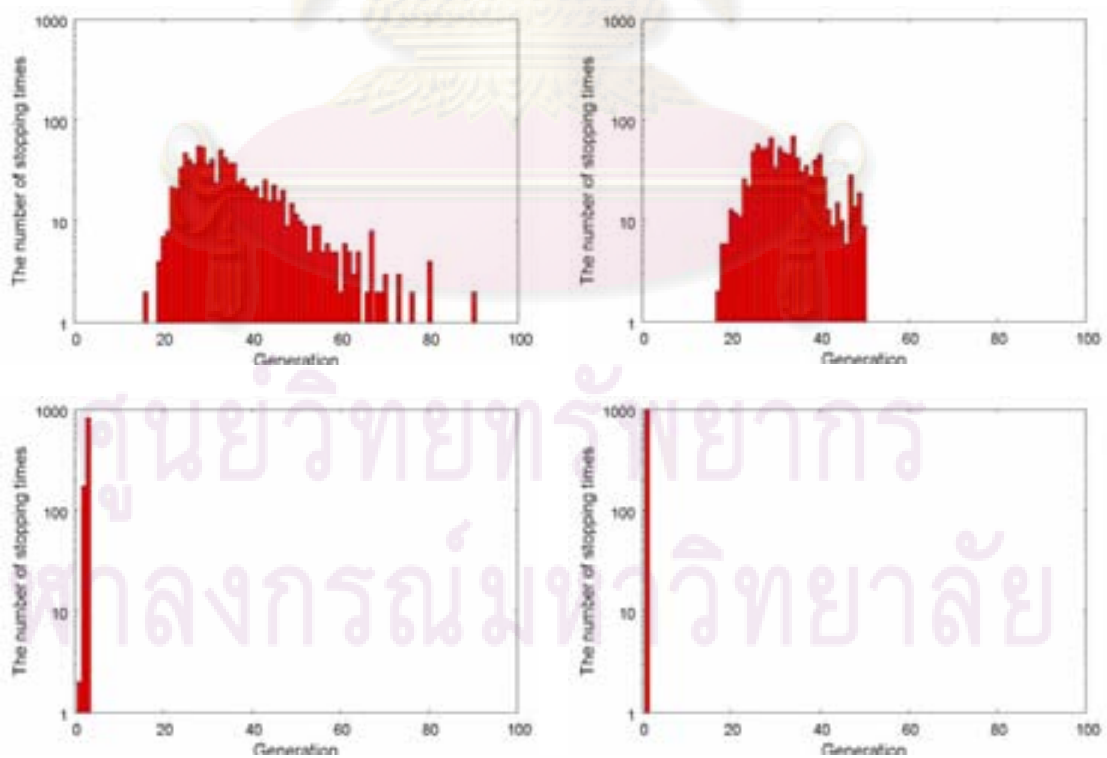
รูปที่ ก.14 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาลำดับชั้นแบบกิตต่อเมื่อขนาด 64 บิต



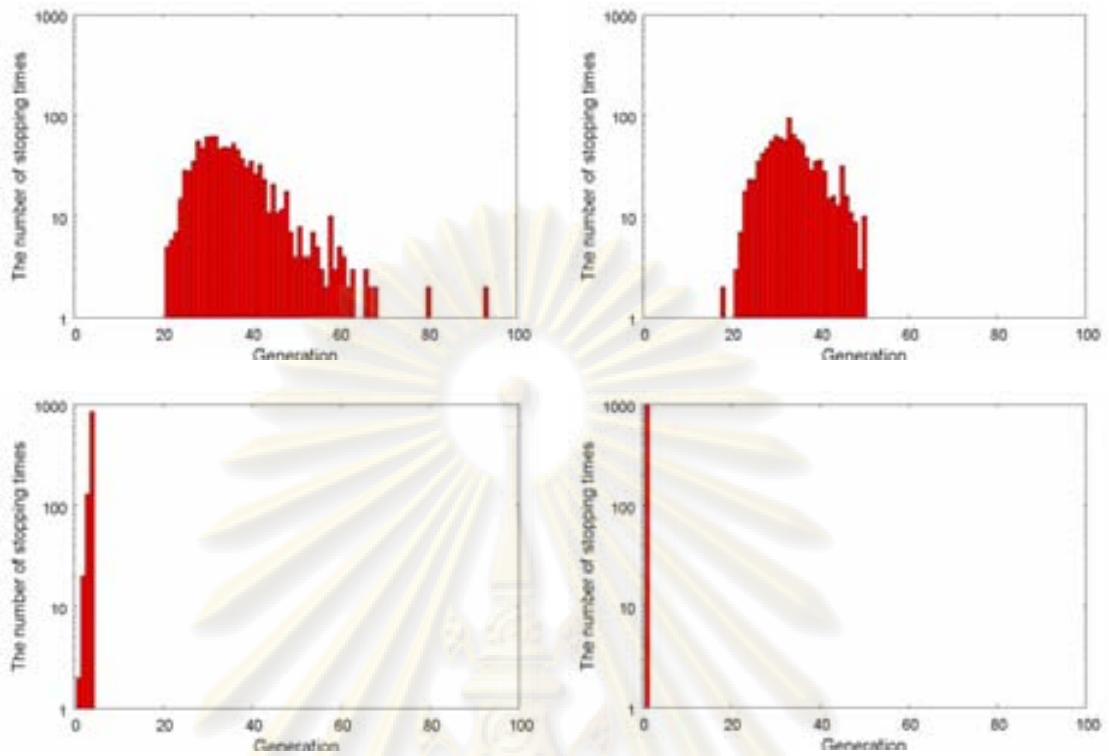
รูปที่ ก.15 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหาลำดับชั้นแบบกิตต่อเมื่อขนาด 128 บิต



รูปที่ ก.16 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหา 2 ชั้นแบบ 6 บิต ขนาด 30 บิต



รูปที่ ก.17 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหา 2 ชั้นแบบ 6 บิต ขนาด 60 บิต



รูปที่ ก.18 จำนวนครั้งการหยุดในแต่ละรุ่นของปัญหา 2 ขั้วแบบ 6 บิต ขนาด 90 บิต

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียนวิทยานิพนธ์

นางสาวสุนิสา ริมเจริญ เกิดเมื่อวันที่ 16 เมษายน พ.ศ. 2525 ที่กรุงเทพมหานคร สำเร็จการศึกษาปริญญาวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์ จากมหาวิทยาลัยบูรพา ในปีการศึกษา 2545 และสำเร็จศึกษาในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิทยาศาสตร์คอมพิวเตอร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ปีการศึกษา 2547 หลังจากนั้นได้เข้าศึกษาในหลักสูตรวิศวกรรมศาสตรดุษฎีบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2548 โดยได้รับทุนอุดหนุนการศึกษา โครงการพัฒนาอาจารย์สาขาขาดแคลนเพื่อศึกษาในประเทศ ตามความต้องการของมหาวิทยาลัยบูรพา



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย