

การค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุด  
และการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย

A MULTI-SWARM OPTIMIZATION WITH CHANGING OF SWARM BEST  
POSITION AND SLIGHTLY MUTATION OF PARTICLE POSITION



วิทยานิพนธ์นี้สำหรับการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต  
สาขาวิชาวิศวกรรมคอมพิวเตอร์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
พ.ศ.2561  
KMITL-2018-EN-M-070-040

การค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุด  
และการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย

A MULTI-SWARM OPTIMIZATION WITH CHANGING OF SWARM BEST  
POSITION AND SLIGHTLY MUTATION OF PARTICLE POSITION



วิทยานิพนธ์นี้สำหรับการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2561

KMITL-2018-EN-M-070-040

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A MULTI-SWARM OPTIMIZATION WITH CHANGING OF SWARM BEST POSITION  
AND SLIGHTLY MUTATION OF PARTICLE POSITION



A THESIS SUBMITTED IN FULFILLMENT  
OF THE REQUIREMENT FOR THE DEGREE OF  
MASTER OF ENGINEERING IN COMPUTER ENGINEERING  
FACULTY OF ENGINEERING  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG  
2018

KMITL-2018-EN-M-070-040

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2018

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุด และการเปลี่ยนตำแหน่งอนุภาคเพียงเล็กน้อย

Thesis Title A Multi-Swarm Optimization with Changing of Swarm Best Position and Slightly Mutation of Particle Position

นักศึกษา นางสาวกนกพร แข่งขุนทด





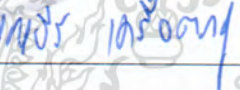
รหัสประจำตัว 58601335

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา วิศวกรรมคอมพิวเตอร์

อาจารย์ที่ปรึกษาวิทยานิพนธ์ รศ.ดร.บุญฉวี เครือตราชู

หมายเลขวิทยานิพนธ์ KMITL-2018-EN-M-070-040

คณะกรรมการสอบวิทยานิพนธ์	ลายมือชื่อ
รศ.ดร.สุรพงศ์ เอื้อวัฒนามงคล	
รศ.ดร.เกียรติคุณ เจริญนัยธนะกิจ	
ผศ.ดร.สมศักดิ์ วลัยรัชต์	
ผศ.ดร.ชุตินิเมษฐ์ ศรีนิลทา	
รศ.ดร.บุญฉวี เครือตราชู	

วัน / เดือน / ปี ที่สอบ วันพฤหัสบดีที่ 26 เมษายน พ.ศ. 2561 เวลา 09.30-11.30 น.  
สถานที่สอบ ณ ห้องประชุม 1 ชั้น 5 อาคาร A

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

คณะวิศวกรรมศาสตร์ รับรองแล้ว



(รองศาสตราจารย์ ดร. คมสัน มาลีสี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
วันที่ 26 เมษายน พ.ศ. 2561

หัวข้อวิทยานิพนธ์	การค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย
นักศึกษา	นางสาวกนกพร เสงข์ขุนทด
รหัสนักศึกษา	58601335
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
พ.ศ.	2561
อาจารย์ที่ปรึกษาวิทยานิพนธ์	รศ.ดร.บุญธีร์ เครือตราชู

## บทคัดย่อ

วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization หรือ PSO) เป็นวิธีที่ใช้ในการหาค่าความเหมาะสมของอนุภาคโดยใช้การเรียนรู้แบบพฤติกรรมการค้นหาอาหารของฝูงนกหรือฝูงปลา ร่วมกับเทคนิคการหาค่าตอบแบบสุ่ม (Stochastic Optimization Technique) วิธีการหาค่าความเหมาะสมของอนุภาคทั้งหมดภายในปริภูมิคำตอบ (Solution Space) โดยที่อนุภาคทั้งหมดจะเคลื่อนที่เข้าหาตำแหน่งของอนุภาคที่ดีที่สุด (GBEST) จนกระทั่งติดอยู่ที่ตำแหน่งเดิม ซึ่งเป็นปัญหาหลักของ PSO เรียกว่าปัญหาการติดอยู่ในจุดสุดสัมพัทธ์ (Local Optimal) ดังนั้นจึงมีนักวิจัยจำนวนมากได้นำเอาเทคนิคการแบ่งกลุ่มอนุภาค (Multi-Swarm Optimization หรือ MPSO) โดยวิทยานิพนธ์ฉบับนี้ได้นำเอา การเปลี่ยนตำแหน่งของอนุภาคทั้งหมดเพียงเล็กน้อยมาประยุกต์ใช้กับ MPSO เพื่อหลีกเลี่ยงหรือลดปัญหาการติดอยู่ในจุดสุดสัมพัทธ์ เรียกว่าการค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย เพื่อแก้ปัญหาการติดอยู่ในจุดสุดสัมพัทธ์ โดยเทคนิคที่นำเสนอนี้ได้ทำการทดสอบกับฟังก์ชันมาตรฐาน 27 ฟังก์ชันโดยผลการทดลอง แสดงให้เห็นว่าวิธีที่นำเสนอมีประสิทธิภาพการค้นหาค่าตอบมากกว่าเมื่อเปรียบเทียบกับวิธีอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<b>Thesis Title</b>	A MULTI-SWARM OPTIMIZATION WITH CHANGING OF SWARM BEST POSITION AND SLIGHTLY MUTATION OF PARTICLE POSITION
<b>Student</b>	Miss. Kanokporn Chengkhuntod
<b>Student ID.</b>	58601335
<b>Degree</b>	Master of Engineering
<b>Program</b>	Computer Engineering
<b>Year</b>	2018
<b>Thesis Advisor</b>	Assoc. Prof. Dr. Boontee Kruatrachue

### Abstract

Particle Swarm Optimization (PSO) algorithm use particles to search for better solution in solution space by changing particles position toward the best particles. One problem of PSO is the trapping in local optima problem. So, many researchers use the division group technique and the distribution group technique in order to avoid the local optima problem. This thesis proposes the novel technique that the multi-swarm optimization with interchange the best position and particle redistribution in order to solve and decrease the trapping in local optima problem. The proposed technique is compared with other techniques on 27 benchmark test functions. The results show that the proposed technique yields better results than other techniques.

## กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี เนื่องจากได้รับความกรุณาและการดูแลที่เต็มเปี่ยมด้วยความกรุณาอย่างยิ่งจากอาจารย์ปรีกษาวิทยานิพนธ์ รศ.ดร. บุญธีร์ เครือตราชู ที่ได้ให้คำปรึกษาแนะนำ ชี้แนะแนวทางในการศึกษาเรียนรู้ ช่วยคิดแก้ไขเพิ่มเติมประเด็นที่มีความสำคัญและให้กำลังใจตั้งแต่เริ่มต้น เวนิการจนกระทั่งแล้วเสร็จจึงขอกราบขอบพระคุณเป็นอย่างสูงมา ณ โอกาสนี้

ขอขอบพระคุณกรรมการสอบวิทยานิพนธ์ทุกท่านที่ได้กรุณาให้คำแนะนำต่างๆ จนทำให้วิทยานิพนธ์ฉบับนี้มีความสมบูรณ์ สามารถครอบคลุมเนื้อหาของงานวิจัยในแง่มุมต่างๆ อย่างครบถ้วน

ขอขอบคุณเจ้าหน้าที่และอาจารย์ทุกท่าน ภาควิชาคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า ลาดกระบัง ที่ได้ให้ความรู้และการช่วยเหลือต่างๆ ทั้งโดยตรงและโดยอ้อม ตลอดจนนักวิจัยทุกท่านที่เอื้อเฟื่องานวิจัยจนทำให้วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ทุกคนที่อยู่ในห้องวิจัยที่คอยให้ความช่วยเหลือสนับสนุนในเรื่องต่างๆ จนทำให้วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี

ขอขอบคุณส่วนสนับสนุนวิชาการและบัณฑิตศึกษา คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้การสนับสนุนในการท้าววิจัยจนทำให้วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี

สุดท้ายขอขอบพระคุณครอบครัวของข้าพเจ้าทุกคนที่ได้ให้การสนับสนุนในเรื่องต่างๆ จนทำให้วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี

หวังว่าผู้อ่านคงได้จะความรู้จากวิทยานิพนธ์ฉบับนี้ไม่มากก็น้อย และหากวิทยานิพนธ์ฉบับนี้มีข้อผิดพลาดประการใด ข้าพเจ้าขอนอมนับไว้แต่เพียงผู้เดียว

อนึ่ง คุณค่าและประโยชน์อันพึงมีซึ่งเป็นผลมาจากงานวิจัยนี้ ขอมอบแด่พระคุณบิดา มารดา ญาติ พี่น้อง ครู อาจารย์ ที่ให้มีโอกาสทางการศึกษาและผู้มีพระคุณทุกท่านมา ณ ที่นี้

กนกพร แซ่ขุนทด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ .....	IV
สารบัญตาราง.....	VIII
สารบัญรูป.....	IX
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	2
1.3 สมมติฐานของการศึกษา.....	3
1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย.....	3
1.5 ขอบเขตการวิจัย.....	3
1.6 ขั้นตอนการศึกษา.....	4
1.7 เครื่องมือและอุปกรณ์ที่ใช้ในงานวิจัย.....	5
1.8 โครงสร้างของวิทยานิพนธ์.....	5
บทที่ 2 ทฤษฎีพื้นฐานที่เกี่ยวข้อง.....	7
2.1 ปัญหาการหาจุดที่ดีที่สุด.....	7
2.1.1 นิยามของปัญหาการหาจุดที่ดีที่สุด.....	7
2.1.1.1 ปัญหาการหาจุดที่ดีที่สุด.....	7
2.1.1.2 ฟังก์ชันเป้าหมาย.....	7
2.1.1.3 ตัวแปรต้น.....	7
2.1.1.4 ข้อจำกัด.....	7
2.2 ประเภทของปัญหาการหาจุดที่ดีที่สุด.....	8
2.2.1 ปัญหาชนิดโหนด.....	8
2.2.2 ปัญหาชนิดโหนด.....	8
2.3 การหาจุดที่ดีที่สุด.....	9
2.3.1 ขั้นตอนวิธีการค้นหาอย่างแน่นอน.....	9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.3.2 ขั้นตอนวิธีการค้นหาอย่างไม่แน่นอน .....	9
2.4 ปัญหาการติดในจุดสุดสัมพัทธ์ .....	9
2.4.1 จุดสุดสัมพัทธ์ .....	9
2.4.2 จุดสุดสัมบูรณ์ .....	9
2.4.3 ปัญหาการติดในจุดสุดสัมพัทธ์ .....	10
2.5 วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค .....	11
2.5.1 อนุภาค .....	12
2.5.2 ประชากร .....	12
2.5.3 รอบการค้นหา คัดบ .....	12
2.5.4 ความเร็วของอนุภาค .....	12
2.5.5 ความเร็วสูงสุด .....	12
2.5.6 ตำแหน่งของอนุภาค .....	12
2.5.7 ตำแหน่งที่ดีที่สุดของอนุภาคแต่ละตัว .....	12
2.5.8 ตำแหน่งที่ดีที่สุดของอนุภาคทั้งหมด .....	12
2.5.9 ฟังก์ชัน คำนวณค่าความเหมาะสมหรือฟังก์ชันเป้าหมาย .....	12
2.6 หลักการท างานของ PSO .....	13
2.6.1 กระบวนการก าหนดค่าเริ่มต้น .....	13
2.6.2 กระบวนการประเมินค่าตำแหน่งของแต่ละอนุภาค .....	13
2.6.3 กระบวนการปรับปรุง PBEST .....	14
2.6.4 กระบวนการเปลี่ยนแปลง GBEST .....	14
2.6.5 กระบวนการค ำนวณความเร็ว .....	14
2.6.6 กระบวนการปรับปรุ้ตำแหน่ง .....	15
2.7 ข้อดีและข้อเสียของ PSO .....	16
2.7.1 ข้อดีของ PSO .....	16
2.7.2 ข้อเสียของ PSO .....	17
2.8 ขั้นตอนวิธีเชิงพันธุกรรม .....	18
2.8.1 นิยามที่ควรทราบเกี่ยวกับขั้นตอนวิธีเชิงพันธุกรรม .....	18
2.8.1.1 โครโมโซม .....	18
2.8.1.2 ประชากร .....	18
2.8.1.3 รอบการค้นหา คัดบ .....	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.8.1.4 ค่าความน่าจะเป็นในการกำหนดค่าประชากรเริ่มต้น .....	19
2.8.1.5 ค่าความน่าจะเป็นในการผสมพันธุ์.....	19
2.8.1.6 ค่าความน่าจะเป็นในการกลายพันธุ์.....	19
2.8.1.7 โครโมโซมที่ดีที่สุด .....	19
2.8.1.8 ฟังก์ชันคำนวณค่าความเหมาะสม .....	19
2.8.2 รายละเอียดขั้นตอนการทำงานของขั้นตอนวิธีเชิงพันธุกรรม.....	19
2.8.2.1 กระบวนการกำหนดค่าเริ่มต้น .....	20
2.8.2.2 กระบวนการผสมพันธุ์.....	20
2.8.2.3 กระบวนการกลายพันธุ์ .....	21
2.8.2.4 กระบวนการคัดสรรประชากรรุ่นถัดไป .....	21
บทที่ 3 งานวิจัยที่เกี่ยวข้อง .....	23
3.1 วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม .....	23
3.1.1 วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม.....	23
3.1.2 งานวิจัยที่นำวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม.....	24
3.1.2.1 วิธีการค้นหาอย่างรวดเร็วแบบหลากหลายกลุ่มโดยการกลายพันธุ์และการผสมพันธุ์.....	24
3.1.2.2 วิธีการคัดเลือกการผสมพันธุ์โดยใช้ความเหมาะสมกับแบบหลากหลายกลุ่ม.....	30
3.2 วิธีหาค่าเหมาะสมที่สุดกับวิธีการกระจายตัวของอนุภาค.....	34
3.2.1 การประยุกต์วิธีการเริ่มต้นการค้นหาใหม่กับวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค.....	34
3.2.2 ขั้นตอนของการเริ่มต้นการค้นหาใหม่.....	35
3.2.3 องค์ประกอบของการเริ่มต้นการค้นหาใหม่.....	36
3.2.4 ข้อดีและข้อเสียของการเริ่มต้นการค้นหาใหม่.....	37
บทที่ 4 การค้นหาค่าเหมาะสมแบบอนุภาคหลากหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย .....	41
4.1 ข้อเสียของขั้นตอนวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม .....	41
4.2 การปรับปรุงขั้นตอนวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่มโดยการใช้ตำแหน่งที่ดีที่สุดของกลุ่มอื่น .....	43
4.3 การกระจายตำแหน่งของอนุภาคเพื่อปรับปรุงการค้นหาแบบหลากหลายกลุ่ม.....	45
4.4 วิธีแก้ปัญหาในกรณีที่อนุภาคของทุกกลุ่มติดอยู่ที่จุดสุดสัมพัทธ์เดียวกัน .....	47
4.5 วิธีการค้นหาค่าเหมาะสมแบบอนุภาคหลากหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย.....	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
บทที่ 5 ผลการทดลอง.....	55
5.1 การทดลองเปรียบเทียบวิธีการเปลี่ยนต าแหน่งที่ดีที่สุดและการเปลี่ยนต าแหน่งของ อนุภาคของวิธีที่ได้นำเสนอ.....	58
5.1 การทดลองเปรียบเทียบประสิทธิภาพของการค้นหาด้วยวิธีที่นำเสนอกับงานวิจัยอื่น... 60	60
บทที่ 6 สรุปผลการทดลองและข้อเสนอแนะ .....	66
6.1 สรุปผลการทดลอง .....	66
6.2 ข้อเสนอแนะ .....	68
เอกสารอ้างอิง .....	70
ภาคผนวก.....	74
ภาคผนวก ก. รายละเอียดของฟังก์ชันมาตรฐาน .....	75
ภาคผนวก ข. งานวิจัยที่ได้รับการตีพิมพ์ .....	102
ภาคผนวก ค. การเปลี่ยนเปอร์เซ็นต์การกระจายต าแหน่งของอนุภาค.....	114
ภาคผนวก ง. การเลื่อนต าแหน่งโดเมนของฟังก์ชันมาตรฐาน.....	118
ประวัติผู้เขียน.....	124

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตาราง

ตารางที่	หน้า
5.1 ค่าพารามิเตอร์ที่ใช้ทดลองกับฟังก์ชันคณิตศาสตร์ .....	56
5.2 ผลการทดลองการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาค .....	58
5.3 ผลการทดลองวิธีที่ 1 เปรียบกับงานวิจัยอื่น .....	60
5.4 ผลสรุปการท างานสุดท้ายของฟังก์ชัน ACKLEY ในหนึ่งรอบการท างาน .....	61
ก.1 สรุปสมบัติของฟังก์ชันคณิตศาสตร์ที่ใช้ในการทดลอง .....	76
ค.1 ผลการทดลอง MPSOSM กับการเปลี่ยนเปอร์เซ็นต์การกระจายต าแหน่ง .....	115
ค.1 สรุปสมบัติของฟังก์ชันคณิตศาสตร์ที่ใช้ในการทดลอง .....	115
ง.1 สรุปสมบัติของฟังก์ชันคณิตศาสตร์ที่ใช้ในการทดลอง .....	118
ง.2 ผลการค้นหาค าดอบการเลื่อนต าแหน่งของอนุภาค .....	120
ง.2 ผลการค้นหาค าดอบการเลื่อนต าแหน่งของอนุภาค (ต่อ).....	122



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

รูปที่	หน้า
2.1 ตัวอย่างปัญหายูนิโมดอล .....	8
2.2 ตัวอย่างปัญหามัลติโมดอล .....	8
2.3 จุดสุดสัมพัทธ์และจุดสุดสัมบูรณ์ .....	10
2.4 ตัวอย่างของปัญหามัลติโมดอลในมุมมองสองมิติ .....	10
2.5 ตำแหน่งอนุภาคในแกนสามมิติ .....	11
2.6 แผนภาพการทำงานของ PSO .....	13
2.7 ตัวอย่างทิศทางการเคลื่อนที่ของอนุภาคในแกนสองมิติ .....	15
2.8 แผนภาพการทำงานของขั้นตอนวิธีเชิงพันธุกรรม .....	20
2.9 ตัวอย่างการสุ่มกำหนดค่าเริ่มต้นให้กับแต่ละมิติของโครโมโซมในประชากร .....	20
2.10 ตัวอย่างการสุ่มกำหนดค่าเริ่มต้นให้กับแต่ละมิติของโครโมโซมในประชากร .....	21
2.11 ตัวอย่างการกระบวนกรกลายพันธุ์ .....	21
2.12 ตัวอย่างการกระบวนกรคัดสรรประชากรรุ่นถัดไป .....	22
3.1 วิธีการหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค (PSO) .....	23
3.2 วิธีการหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม (MPSO) .....	23
3.3 การกระจายตัวแบบ Cauchy .....	24
3.4 แผนภาพการทำงานของวิธี FPSO .....	25
3.5 แผนภาพการทำงานของวิธี FMPSO .....	26
3.6 วิธีการหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม .....	27
3.7 การเคลื่อนที่ของอนุภาคแต่ละกลุ่ม .....	27
3.8 อนุภาคที่ถูกปรับปรุงโดยใช้วิธีการกลายพันธุ์แบบเคซี .....	28
3.9 วิธีการผสมพันธุ์ระหว่างกลุ่มอนุภาคเดียวกัน .....	29
3.10 กระบวนกรผสมพันธุ์ระหว่างกลุ่มอนุภาคต่างกัน .....	30
3.11 แผนภาพการทำงานของวิธี SCMPSO .....	31
3.12 วิธีการเลือกตามธรรมชาติประยุกต์ใช้กับวิธีการคัดเลือกการผสมพันธุ์ .....	32
3.13 กราฟเปรียบเทียบวิธี FMPSO และ SCMPSO .....	34
3.14 แผนภาพการทำงานของรีนหรือรีโพซิชั่น .....	35
3.15 การกระจายตำแหน่งของอนุภาคช่วยปรับปรุงค่าตอบของ PSO .....	38
3.16 การกำหนดค่าเริ่มต้นให้กับ PBEST และ GBEST ใหม่ .....	39

## สารบัญญรูป (ต่อ)

รูปที่	หน้า
3.17	การกำหนดค่าเริ่มต้นที่ไม่มีกำหนดค่าเริ่มต้นให้กับ PBEST และ GBEST..... 40
4.1	ตัวอย่างกราฟวิธืหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค (PSO)..... 41
4.2	ตัวอย่างกราฟวิธืหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม (MPSO)..... 42
4.3	การเปลี่ยนตำแหน่งที่ดีที่สุดของกลุ่มอนุภาคกับกลุ่มอื่น..... 43
4.4	กราฟการใช้ตำแหน่ง GBEST ของกลุ่มข้างเคียง..... 44
4.5	การกระจายตำแหน่งของอนุภาคตามแบบสมการที่ (4.1)..... 46
4.6	การกระจายตำแหน่งของอนุภาคตามแบบสมการที่ (4.2)..... 46
4.7	กราฟการเปลี่ยนตำแหน่งของอนุภาค GBEST..... 48
4.8	อนุภาคทุกกลุ่มติดอยู่ในจุดสุดสัมพัทธ์เดียวกัน..... 48
4.9	การกระจายตำแหน่งของอนุภาคที่ดีที่สุดตามแบบสมการที่ (4.3)..... 50
4.10	การกระจายตำแหน่งของอนุภาคตามสมการที่ (4.2) และ (4.3)..... 50
4.11	รหัสจำลองแสดงการทำงานของ MPSOSM..... 52
4.12	แผนภาพการทำงานของวิธื MPSOSM..... 53
5.1	LBP ของฟังก์ชัน ACKLEY ที่ค้นหาค่าตอบด้วย MPSOSM ในรอบการทำงานทั้งหมด..... 61
5.2	LBP ของฟังก์ชัน ACKLEY ที่ค้นหาค่าตอบด้วย MPSOSM ในรอบที่ 0-5,825..... 62
5.3	LBP ของฟังก์ชัน ACKLEY ที่ค้นหาค่าตอบด้วย MPSOSM ในรอบที่ 5,825- 9,000..... 62
5.4	LBP ของฟังก์ชัน ACKLEY ที่ค้นหาค่าตอบด้วย MPSOSM ในช่วง 9,000 - 21,100..... 63
5.5	LBP ของฟังก์ชัน ACKLEY ที่ค้นหาค่าตอบด้วย MPSOSM ในรอบที่ 21,100 - 29,326..... 64
5.6	LBP ของฟังก์ชัน ACKLEY ที่ค้นหาค่าตอบด้วย MPSOSM ในรอบที่ 28,230 - 28,965..... 64
5.7	LBP ของฟังก์ชัน ACKLEY ที่ค้นหาค่าตอบด้วย MPSOSM ในรอบที่ 29,528 - 29,783..... 65
ก.1	ฟังก์ชัน ACKLEY..... 77
ก.2	ฟังก์ชัน GRIEWANK..... 78
ก.3	ฟังก์ชัน RASTRIGIN..... 79
ก.4	ฟังก์ชัน ROSENBROCK..... 80
ก.5	ฟังก์ชัน SCHWEFEL..... 81
ก.6	ฟังก์ชัน COSINE MIXTURE..... 82
ก.7	ฟังก์ชัน EXPONENTIAL..... 83
ก.8	ฟังก์ชัน LEVY..... 84

## สารบัญรูป (ต่อ)

รูปที่	หน้า
ก.9 ฟังก์ชัน MICHALEWICZ .....	85
ก.10 ฟังก์ชัน STEP .....	86
ก.11 ฟังก์ชัน SCHAFFER.....	87
ก.12 ฟังก์ชัน HOLDER .....	88
ก.13 ฟังก์ชัน BEALE .....	89
ก.14 ฟังก์ชัน SHUBERT .....	90
ก.15 ฟังก์ชัน GOLDSTEIN-PRICE .....	91
ก.16 ฟังก์ชัน TRID.....	92
ก.17 ฟังก์ชัน SPHERE .....	93
ก.18 ฟังก์ชัน PARALLEL ELLIPSOID .....	94
ก.19 ฟังก์ชัน ROTATED ELLIPSOID .....	95
ก.20 ฟังก์ชัน CIGAR .....	96
ก.21 ฟังก์ชัน BROWN .....	97
ก.22 ฟังก์ชัน MULTIMOD.....	98
ก.23 ฟังก์ชัน ZAKHAROV .....	99
ก.24 ฟังก์ชัน EASOM.....	100
ก.25 ฟังก์ชัน EGGHOLDER.....	101
ง.1 กราฟแสดงการเลื่อนตำแหน่งโดเมนของฟังก์ชันตามแนวแกน X .....	120

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

วิธีการหาค่าเหมาะสมแบบอนุภาคกลุ่ม (Particle Swarm Optimization หรือ PSO) [1, 2] ถูกคิดค้นและพัฒนาโดย รัสเซล ซี อีเบอร์ฮาร์ท (Russell C. Eberhart) และ เจมส์ เคเนดี (James Kennedy) ในปี ค.ศ. 1995 โดยเรียนแบบพฤติกรรมเคลื่อนไหวของฝูงสัตว์ เช่น ฝูงนก ฝูงปลา และการสื่อสารแลกเปลี่ยนข้อมูลระหว่างสัตว์ภายในฝูง เป็นต้น โดยที่ PSO เป็นขั้นตอนวิธีที่ใช้แก้ปัญหาการหาจุดที่ดีที่สุด (Optimization Problems) ที่เป็นที่ยอมรับกันอย่างแพร่หลายและถูกนำมาใช้ประยุกต์ใช้ในหลายด้าน เพราะด้วยคุณสมบัติหลายอย่าง เช่น มีจำนวนพารามิเตอร์ที่ต้องปรับน้อย นำเอาไปใช้ต่อได้ง่ายและมีอัตราการลู่เข้าหาค่าตอบที่รวดเร็ว โดย PSO นั้นจะมีประชากรหรือกลุ่มอนุภาค (Particles) ที่จะถูกสุ่มลงบนพื้นที่ที่ต้องการค้นหาคำตอบ (Solution Space) ที่ซึ่งอนุภาคแต่ละตัวจะมีตำแหน่ง (X) และความเร็ว (V) ของตัวเอง ทาหน้าที่เคลื่อนที่ไปมาในปริภูมิของคำตอบเพื่อค้นหาคำตอบที่ดีที่สุด โดยที่อนุภาคทั้งหมดของ PSO จะเคลื่อนที่เข้าหาตำแหน่งที่ดีที่สุดของกลุ่มหรือจำฝูง (GBEST หรือ Global BEST) ที่มีจะอยู่ในบริเวณใกล้เคียงเดิม จนทำให้ PSO เกิดปัญหาการติดอยู่ในจุดสุดสัมพัทธ์ (Local Optimum Problem) และปัญหาการลู่เร็วเกินไป (Premature Convergence Problem)

เพื่อแก้ปัญหาการติดอยู่ในจุดสุดสัมพัทธ์จึงได้มีการนำงานวิจัยนำเอา PSO มาพัฒนา โดยการแบ่งอนุภาคออกเป็นกลุ่มย่อย โดยที่แต่ละกลุ่มนั้นมีอนุภาคและตำแหน่งที่ดีที่สุด (GBEST) ของตัวเอง ถูกเรียกว่า วิธีการหาค่าเหมาะสมแบบหลายกลุ่มอนุภาค (Multi-PSO หรือ MPSO) แต่กระบวนการเหล่านี้กลับทำให้คำตอบในสมการส่วนใหญ่ของ PSO ที่ได้มีค่าที่แย่ลงกว่า PSO ธรรมดา

นักวิจัยหลายคนจึงได้นำเอา MPSO มาปรับปรุงร่วมกับเทคนิคต่างๆ [3-6] เพิ่มประสิทธิภาพให้กับ MPSO ดังเช่นงานวิจัยที่ได้นำมาเปรียบเทียบในบทความในการประชุมวิชาการที่ได้นำเสนอนี้ ได้แก่ ขั้นตอนวิธีการค้นหาอย่างรวดเร็วแบบหลากหลายกลุ่มโดยการกลายพันธุ์และการผสมพันธุ์ (Fast Multi-swarm Optimization with Cauchy Mutation and Crossover operation หรือ FMPSO) และขั้นตอนวิธีคัดเลือกรดาเนินงานตามกระบวนการผสมพันธุ์ในอนุภาคหลายกลุ่ม (Selective Crossover base on Fitness in Multi-Swarm Optimization หรือ SCMPSO) และขั้นตอนวิธีดัดแปลงการค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่ม ด้วยการแลกเปลี่ยนตำแหน่งของอนุภาคที่ดีที่สุดและกระจายอนุภาค (A Modified Multi-Swarm Optimization with Interchange GBEST and Particle Redistribution หรือ MPSOIR) งานวิจัยเหล่านี้ทำให้ได้คำตอบที่ดีขึ้นเมื่อเทียบกับ

PSO ธรรมดาและหลายงานวิจัยที่ได้มาเปรียบเทียบ แต่เทคนิคที่ได้มา ใช้ในงานวิจัยเหล่านี้ กลับเป็นการเพิ่มขึ้นตอนการท างานที่ไม่จำเป็นในแต่ละรอบของการท างานของ PSO

จึงเป็นสาเหตุให้ท าการศึกษา MPSO โดยได้เพิ่มเทคนิค ที่มีแนวคิดมาจากพฤติกรรมกรหาอาหารและการหลบหนีของหมาป่าในงานวิจัยที่ [14] โดยที่เมื่ออนุภาคของแต่ละกลุ่มมีการติดอยู่ในจุดสุดสัมพัทธ์ให้ท าการค้นหาในบริเวณใหม่ที่ก หนดจากจุดเดิมหรือรีโพซิชั่น (REPOSITION) แต่จากการท างานของ PSO อนุภาคของกลุ่มจะลู่เข้าหาจ่าฝูง ดังนั้นเมื่อกระจายอนุภาคออกจากกันด้วยรีโพซิชั่น แทนที่จะกำหนดค่าใหม่ (RESET) ให้กับค่า GBEST เหมือนเทคนิคเดิม [14] ท าให้การเปลี่ยนตาแหน่งของจ่าฝูงหรือ GBEST กับตาแหน่งของจ่าฝูง หรือ GBEST ของกลุ่มข้างเคียงแทน เพื่อให้อนุภาคทั้งหมดของกลุ่มที่ติดในจุดสุดสัมพัทธ์มีโอกาสในการเคลื่อนที่ไปยังตาแหน่งอื่นแล้วมีโอกาสเจอคำตอบที่ดีขึ้น แต่ถ้าตาแหน่งของจ่าฝูงของกลุ่มข้างเคียงติดอยู่ในจุดสุดสัมพัทธ์เดียวกันให้ท าการกำหนดค่า GBEST ใหม่ของกลุ่ม จากเทคนิคที่ได้เพิ่มเข้ามานั้น กลับท าให้คำตอบของ MPSO ดีขึ้นเมื่อเทียบกับงานวิจัยที่ได้มาเปรียบเทียบ และยังได้ลดกระบวนการท างานของ MPSO ให้ท างานเฉพาะเมื่ออนุภาคติดในจุดสุดสัมพัทธ์เดียวกัน

แต่จากการศึกษาทดลองสมการกับรีโพซิชั่น [14] พบว่าอนุภาคจะเคลื่อนที่กระจายออกจากตาแหน่งเดิมมากเกินไป โดยมีโอกาสถึง 70 เปอร์เซ็นต์ในการกระจายออก ในขณะที่คำตอบในบางสมการ อยู่ใกล้เคียงกับตาแหน่งเดิมของอนุภาค ท าให้คำตอบที่ได้ไม่ดีขึ้น ดังนั้นจึงได้ท าการทดลองลดตาแหน่งของนกในการรีโพซิชั่นลง เพื่อให้อนุภาคมีโอกาสค้นหาใกล้เคียงตาแหน่งเดิม ผลลัพธ์ที่ได้กับพบคำตอบที่ดีขึ้น

## 1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

วิทยานิพนธ์ฉบับนี้มุ่งหวังศึกษาแนวทางปรับปรุงประสิทธิภาพการค้นหาของ PSO โดยการแบ่งกลุ่ม PSO ออกเป็นกลุ่มย่อย เรียก MPSO หรือ Multi-PSO และท างานโดยการเปลี่ยนตาแหน่ง GBEST ของกลุ่มที่ติดในจุดสุดสัมพัทธ์แทนการกำหนดค่าใหม่ (RESET) ร่วมกับรีโพซิชั่นมาประยุกต์ใช้ เพื่อลดปัญหาการติดในจุดสุดสัมพัทธ์ ท าให้คำตอบที่ได้จากขั้นตอนเหล่านี้ดีขึ้น โดยท าการทดลองกับปัญหาฟังก์ชันมาตรฐาน (Benchmark Test Functions)

### 1.3 สมมติฐานของการศึกษา

สำหรับงานวิจัยนี้ได้ดำเนินการศึกษาและปรับปรุงพัฒนา PSO จาก Multi-PSO ให้มีประสิทธิภาพการค้นหาและได้รับค าทอบที่ดีขึ้นด้วยการแลกเปลี่ยนค่า GBEST ของกลุ่มและกระจายอนุภาคและการรีโพอิชัน โดยการเปรียบเทียบขั้นตอนวิธีว่า ขั้นตอนวิธีใดมีประสิทธิภาพการค้นหาคดีกว่า เพื่อให้สามารถหาคาตอบได้เร็วขึ้นโดยที่ยังสามารถลดปัญหาการติดอยู่ในจุดสุดสัมพัทธ์ได้ใน PSO

### 1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

จากสมมติฐานที่ว่า PSO มีข้อเสียหลัก คือปัญหาการติดในจุดสุดสัมพัทธ์และการลู่เข้าที่เร็วเกินไป แนวคิดหลักของงานวิจัยนี้จึงได้นำเสนอ เพื่อที่จะแก้ปัญหาคาการติดในจุดสุดสัมพัทธ์ของ PSO และลดกระบวนการทำงานของขั้นตอน (Algorithms) ของงานวิจัยอื่นที่ได้นำมาใช้เปรียบเทียบ โดยที่ได้นำเอาวิธีการรีโพอิชันมาประยุกต์ใช้กับ MPSO เมื่ออนุภาคของกลุ่มใดกลุ่มหนึ่งติดในจุดสุดสัมพัทธ์ให้ทำการกำหนดค่าใหม่ (RESET) ให้กับตำแหน่งที่ดีที่สุดของอนุภาคแต่ละตัว (PBEST) แต่แทนที่จะกำหนดค่าใหม่ให้กับ GBEST ด้วยเมื่อทาการกระจายตำแหน่งของอนุภาค (Reposition) แต่ให้ทาการลู่เปลี่ยนตำแหน่งของ GBEST ที่ติดในจุดสุดสัมพัทธ์กับ GBEST ของกลุ่มข้างเคียงที่ไม่ติดในจุดสุดสัมพัทธ์เดียวกันมาใช้เพื่อให้อนุภาคของกลุ่มที่ติดในจุดสุดสัมพัทธ์มีโอกาสในการเคลื่อนที่ไปยังตำแหน่งอื่นได้อีกแล้วอาจพบคาตอบที่ดีขึ้น แต่ถ้าเกิดกรณีที่อนุภาคทั้งหมดของทุกกลุ่มติดอยู่ในจุดสุดสัมพัทธ์เดียวกัน ให้ทาการกำหนดค่าใหม่ (RESET) ค่าตำแหน่งของ PBEST ของอนุภาคทุกตัว และค่าตำแหน่งของ GBEST ของอนุภาคทุกกลุ่มแล้วทาการรีโพอิชันตำแหน่งที่ดีที่สุดของอนุภาคทุกกลุ่ม (GBEST) ประมาณ 15% ของมิติทั้งหมด และรีโพอิชันตำแหน่งของอนุภาคทั้งหมด (X) ประมาณ 15% ของมิติทั้งหมด เพื่อให้อนุภาคทั้งหมดของทุกกลุ่มกระจายออกไปค้นหาที่ยังตำแหน่งใหม่ ซึ่งทาให้คาคาตอบที่ได้นั้นดีขึ้นมากกว่าการประยุกต์ใช้รีโพอิชันกับ PSO เพียงอย่างเดียว

สำหรับวิธีการรีโพอิชันถ้าหากไม่มีการกำหนดค่าใหม่ (RESET) ตำแหน่งที่ดีที่สุดของอนุภาคแต่ละตัว (PBEST) และตำแหน่งที่ดีที่สุดของอนุภาคทั้งหมด (GBEST) อนุภาคจะเคลื่อนที่กลับมาบริเวณเดิม จึงทาให้คาคาตอบที่ได้จากการค้นหาไม่ดีเท่าที่ควร

### 1.5 ขอบเขตการวิจัย

วิทยานิพนธ์ฉบับนี้ทาการศึกษาและพัฒนาการแลกเปลี่ยนค่า GBEST และวิธีการรีโพอิชันกับ MPSO โดยกำหนดขอบเขตของงานวิจัยดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. งานวิจัยนี้เน้นการปรับปรุง PSO จากการแบ่งกลุ่มอนุภาคออกเป็นกลุ่มย่อย (Multi-PSO หรือ MPSO) โดยกำหนดให้แต่ละกลุ่มมีจำฝูงหรือตำแหน่งที่ดีที่สุดของตัวเอง (GBEST) เพื่อศึกษาการทำงานของงานชด MPSO
2. งานวิจัยนี้ต้องการที่จะค้นหาค่าความเหมาะสม (Fitness Value) หรือค่าคำตอบที่ดีที่สุด โดยพยายามที่จะแก้ไขปัญหาการติดในจุดสุดสัมพัทธ์ ซึ่งเป็นข้อเสียหลักของ PSO โดยขั้นตอนวิธีที่วิธานิพนธ์ฉบับนี้นำเสนอสามารถทำให้หลุดจากจุดสุดสัมพัทธ์เดิมที่ PSO ประสบปัญหาแล้วไปพบจุดสุดสัมพัทธ์ใหม่ที่ดีกว่าเดิมหรืออาจจะพบจุดสุดสัมบูรณ์ (Global Optimum) ดังนั้นขั้นตอนวิธีที่นำเสนอนี้อาจจะไม่สามารถหาค่าตอบที่ดีที่สุดของปัญหาที่นำมาแก้ไขได้หมด แต่ขั้นตอนวิธีที่นำเสนอนี้จะได้รับคำตอบจากค้นหาที่ดีกว่า PSO และขั้นตอนวิธีที่นำมาใช้เปรียบเทียบ
3. งานวิจัยนี้เน้นการปรับปรุง PSO ให้สามารถค้นหาได้รับคำตอบที่ดีกว่า PSO ในปัญหาการหาจุดที่ดีที่สุดเชิงต่อเนื่อง (Continuous Optimization Problem) คือ การหาค่าคำตอบที่ดีที่สุดในกลุ่มอนุภาค
4. เพื่อวัดประสิทธิภาพของวิธีการที่นำเสนอในวิธานิพนธ์ฉบับนี้ ได้เก็บทำการทดลองเพื่อค้นหาค่าตัวแปรที่ดีที่สุดที่สามารถนำมาใช้ให้ได้คำตอบที่ดีที่สุด เช่น จำนวนรอบการติดในจุดสุดสัมพัทธ์ของอนุภาคแต่ละกลุ่ม และ จำนวนอนุภาคที่ถูกใช้ในแต่ละกลุ่มแทนที่เพื่อให้ได้คำตอบที่ดีที่สุด (Fitness Value หรือ Objective Value) เป็นต้น โดยได้ทำการทดลองกับปัญหาฟังก์ชันมาตรฐาน (Benchmark Test Functions)

## 1.6 ขั้นตอนการศึกษา

1. ศึกษาทฤษฎีและความรู้พื้นฐานที่เกี่ยวข้องกับ PSO
2. ศึกษาแนวคิดและทฤษฎีในการแบ่งอนุภาคออกเป็นกลุ่มย่อยของ PSO (MPSO หรือ Multi-PSO) มาเปรียบเทียบกับ PSO ธรรมดา
3. ศึกษาทฤษฎีและความรู้พื้นฐานที่เกี่ยวข้องกับปัญหาฟังก์ชันมาตรฐานที่นำมาใช้ในการเปรียบเทียบ
4. ศึกษาการวิธีการประยุกต์รีรันและรีโพซิชันกับกลุ่มย่อยของ PSO เปรียบเทียบกับ PSO ธรรมดา
5. ศึกษาผลกระทบจากวิธีการประยุกต์รีรันและรีโพซิชันกับ MPSO ร่วมกับการเปลี่ยนตำแหน่งที่ดีที่สุดของอนุภาคทั้งหมดภายในกลุ่ม (GBEST)
6. ศึกษาแนวทางการแก้ปัญหาที่เกิดจากการประยุกต์รีโพซิชันร่วมกับการเปลี่ยนตำแหน่งที่ดีที่สุดของอนุภาคทั้งหมดภายในกลุ่ม (GBEST) กับ MPSO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ศึกษาและทดลองการเปลี่ยนตำแหน่งของอนุภาคที่ดีที่สุด (GBEST) โดยคัดเลือกจากระยะทางของตำแหน่งที่ดีที่สุดและจำนวนรอบในการติดจุดสุดสัมพัทธ์ของกลุ่มที่พบปัญหา
8. ทดลองเพื่อเปรียบเทียบผลลัพธ์จากปัญหาฟังก์ชันมาตรฐาน เพื่อศึกษาข้อดีและข้อเสียของวิธีวิเศษชันร่วมกับการเปลี่ยนตำแหน่งที่ดีที่สุดของอนุภาคทั้งหมดภายในกลุ่ม (GBEST) เพื่อลดปัญหาการติดในการติดจุดสุดสัมพัทธ์
9. ศึกษาและทดลองจำนวนรอบที่ในการติดจุดสุดสัมพัทธ์ก่อนและหลังทาร์วิเศษชันร่วมกับการเปลี่ยนตำแหน่งที่ดีที่สุดของอนุภาคทั้งหมดภายในกลุ่ม (GBEST)
10. สรุปผลการท างานพร้อมวิเคราะห์ผลที่ได้ทั้งหมด
11. จัดท าสารประกอบวิทยานิพนธ์

### 1.7 เครื่องมือและอุปกรณ์ที่ใช้ในงานวิจัย

1. เครื่องคอมพิวเตอร์ส่วนบุคคล
  - หน่วยประมวลผลกลางยี่ห้อ Intel รุ่น Core i7 3770 ความเร็ว 2.4 GHz
  - หน่วยความจำ หลักขนาด 8 GB
2. ระบบปฏิบัติการ Windows Seven Professional
3. โปรแกรม Microsoft Visual C++ 2005

### 1.8 โครงสร้างของวิทยานิพนธ์

วิทยานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 6 บท ดังนี้

บทที่ 1 อธิบายถึงความเป็นมาของงานวิจัย ความมุ่งหมายและวัตถุประสงค์ สมมติฐาน ทฤษฎีที่ใช้ ขอบเขตของการวิจัยและขั้นตอนการศึกษา เครื่องมือและอุปกรณ์ที่ใช้ในการพัฒนางานวิจัย

บทที่ 2 อธิบายถึงนิยามรวมถึงทฤษฎีพื้นฐานที่เกี่ยวข้องคือ ปัญหาการหาจุดที่ดีที่สุด การหาจุดที่ดีที่สุด ปัญหาการติดในจุดสุดสัมพัทธ์ วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค ตัวอย่างการประยุกต์ใช้ PSO ในทางปฏิบัติ ข้อดีและข้อเสียของ PSO

บทที่ 3 อธิบายถึงงานวิจัยที่นำมาใช้เปรียบเทียบและเกี่ยวข้อง ได้แก่ วิธีการค้นหาอย่างรวดเร็วแบบหลากหลายกลุ่มโดยการกลายพันธุ์และการผสมพันธุ์ (Fast Multi-swarm Optimization with Cauchy Mutation and Crossover operation หรือ FMPSO) [9] และ วิธีการคัดเลือกการต นินงานตามกระบวนการผสมพันธุ์ในอนุภาคหลายกลุ่ม (Selective Crossover เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

base on Fitness in Multi-Swarm Optimization หรือ SCMPSO) [10] และวิธีการตัดแปลงการค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่ม ด้วยการแลกเปลี่ยนตำแหน่งของอนุภาคที่ดีที่สุดและกระจายอนุภาค (A Modified Multi-Swarm Optimization with Interchange GBEST and Particle Redistribution หรือ MPSOIR) [15]

บทที่ 4 อธิบายถึงวิธีการรีโพซิชันให้ทำงานร่วมกันกับการสุ่มแลกเปลี่ยน GBEST ของ MPSO ซึ่งได้ปรับปรุงขึ้นแทนการกำหนดค่าใหม่ (RESET) ให้กับค่า GBEST ก่อนการทารีโพซิชันตามสมการของขั้นตอนวิธีที่ [16] เรียก วิธีการค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย (A Multi-Swarm Optimization with Changing of Swarm Best Position and Slightly Mutation of Particle Position หรือ MPSOSM)

บทที่ 5 อธิบายการทดลองเพื่ออ้างอิงแนวคิดและที่มาของวิธีการที่วิทยานิพนธ์ฉบับนี้นำเสนอ MPSOSM และเปรียบเทียบผลลัพธ์กับ PSO วิธีการอื่นที่มีลักษณะใกล้เคียงกับวิธีการที่นำเสนอ โดยดำเนินการทดลองกับปัญหาฟังก์ชันมาตรฐาน

บทที่ 6 อธิบายถึงบทสรุปผลการทดลองและข้อเสนอแนะ รวมทั้งแนวทางในการพัฒนาโปรแกรม

## บทที่ 2

# ทฤษฎีพื้นฐานที่เกี่ยวข้อง

### 2.1 ปัญหาการหาจุดที่ดีที่สุด

ปัญหาการหาจุดที่ดีที่สุด (Optimization Problems) คือ ปัญหาที่เกิดจากการหาค่าที่ดีที่สุดจากค าดมทั้งหมด โดยปัญหานี้สามารถแสดงในรูปของสมการทางคณิตศาสตร์ได้ดังนี้

$$\begin{aligned} \text{Min. or Max. } & f(x) \quad x \in R, I \\ \text{Subject } & g_i(x) \leq a_i, \quad i = 1, \dots, m \\ & h_i(x) = b_i, \quad i = 1, \dots, p \end{aligned} \quad (2.1)$$

โดยที่  $f(x)$  เป็นฟังก์ชันเป้าหมาย (Objective function) คือฟังก์ชันที่นำมาใช้เป็นตัวแทนของปัญหาการหาจุดที่ดีที่สุด เพราะปัญหาการหาจุดที่ดีที่สุดนั้นไม่สามารถนำมาใช้กับคอมพิวเตอร์ได้ ปัญหาการหาจุดที่ดีที่สุดนั้นต้องการแปลงให้อยู่ในรูปฟังก์ชันทางคณิตศาสตร์ก่อน จึงจะสามารถนำมาใช้กับคอมพิวเตอร์เพื่อ ค านวณได้

$x$  เป็นตัวแปรต้น (Control Variables) คือตัวแปรที่อยู่ในฟังก์ชันเป้าหมาย เช่น  $f(x_1, x_2, \dots, x_n)$  โดยตัวแปรต้นจะถูกแทนค่า เพื่อให้ฟังก์ชันเป้าหมายค านวณออกมา เป็นค่าความเหมาะสม (Fitness) โดยปัญหาการหาจุดที่ดีที่สุดต้องการให้ค่าความเหมาะสมนั้น มีค่ามากที่สุดหรือค าน้อยที่สุด

$g_i(x)$  เป็นข้อจ ากัดที่ไม่เท่ากัน (Inequality Constraints)

$h_i(x)$  เป็นข้อจ ากัดที่เท่ากัน (Equality Constraints)

โดยข้อจ ากัด (Constraint) คือขอบเขตสำหรับการหาค่าตัวแปรต้นที่อยู่ในฟังก์ชันเป้าหมาย โดยข้อจ ากัด เกิดมาจากคุณสมบัติของตัวแปรต้น เช่น ตัวแปรต้นคือ น้ำหนัก ระยะทาง เวลา ดังนั้นตัวแปรต้นไม่มีโอกาสติดลบ เป็นต้น

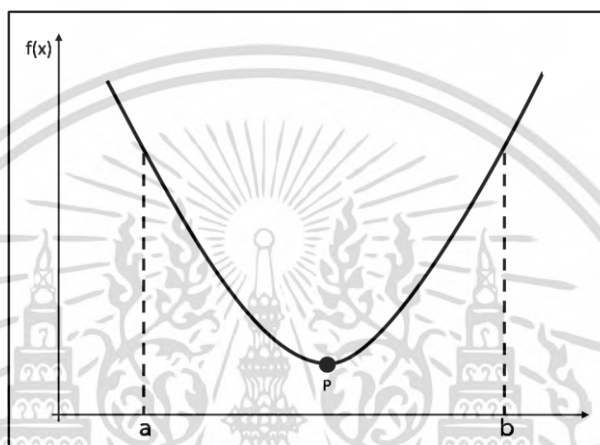
$a_i$  และ  $b_i$  เป็นค่าคงที่ซึ่งเป็นตัวจ ากัดขอบเขตของปัญหาการหาจุดที่ดีที่สุด

$R$  คือ จ านวนจริง

$i$  คือ จ านวนเต็ม

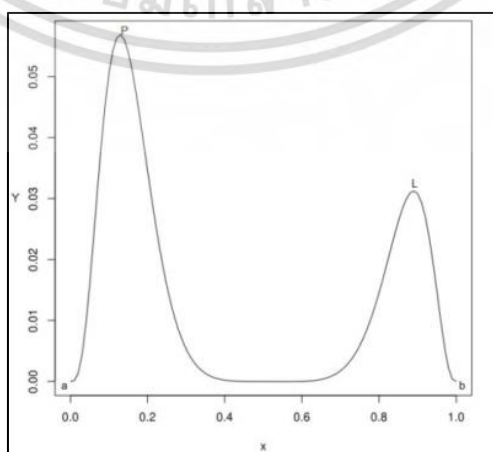
## 2.2 ประเภทของปัญหาการหาจุดที่ดีที่สุด

1) ปัญหาหุณีโมดอล คือ ปัญหาการหาจุดที่ดีที่สุดที่ในขอบเขต โดยมีเพียงจุดสุดสัมบูรณ์ (Global Optimum) เพียงจุดเดียวเท่านั้นและไม่มีจุดสุดสัมพัทธ์ (Local Optimum) ตัวอย่างของปัญหาดังรูปข้างล่างนี้ โดยกำหนดฟังก์ชัน  $f(x)$  ให้มีขอบเขตอยู่ในช่วงระหว่าง  $a$  ถึง  $b$  จากรูปมีจุด  $P$  เพียงจุดเดียวเท่านั้นที่มีค่าน้อยที่สุดในขอบเขตระหว่าง  $a$  ถึง  $b$  ซึ่งเรียกจุด  $P$  ว่าจุดสุดสัมบูรณ์



รูปที่ 2.1 ตัวอย่างปัญหาหุณีโมดอล

2) ปัญหาหัลติโมดอล คือปัญหาการหาจุดที่ดีที่สุดที่ในขอบเขต โดยมีจุดสุดสัมบูรณ์เพียงจุดเดียว แต่มีจุดสุดสัมพัทธ์มากกว่าหนึ่งจุด ตัวอย่างปัญหาดังรูปข้างล่างนี้ กำหนดฟังก์ชัน  $f(x)$  ให้มีขอบเขตอยู่ในช่วงระหว่าง  $a$  ถึง  $b$  จากรูปมีจุด  $P$  เพียงจุดเดียวเท่านั้นที่มีค่ามากที่สุดภายในขอบเขตอยู่ในช่วงระหว่าง  $a$  ถึง  $b$  ซึ่งเรียกจุด  $P$  ว่าจุดสูงสุดสัมบูรณ์และมีจุด  $L$  เป็นจุดที่มีค่ามากกว่าพื้นที่รอบข้างซึ่งเรียกจุด  $L$  ว่าจุดสุดสัมพัทธ์



รูปที่ 2.2 ตัวอย่างปัญหาหัลติโมดอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3 การหาจุดที่ดีที่สุด

การหาจุดที่ดีที่สุดหรือการหาค่าที่ดีที่สุด (Optimization) คือการหาค่าสูงสุดหรือค่าต่ำสุดของปัญหาการหาจุดที่ดีที่สุดหรืออาจกล่าวได้ว่าเป็นการหาค่าตัวแปรต้นที่ทำให้ผลลัพธ์จากฟังก์ชันเป้าหมายมีค่ามากที่สุดหรือค่าน้อยที่สุด โดยแบ่งวิธีการหาเป็นสองวิธีได้แก่

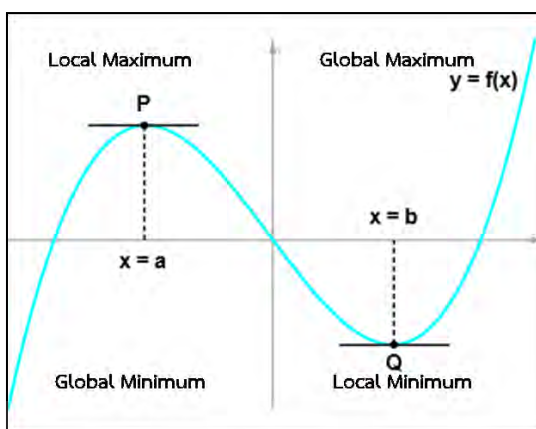
1) ขั้นตอนวิธีการค้นหาอย่างแน่นอน (Deterministic Algorithm) เป็นการค้นหาตอบที่ไม่มีขั้นตอนการสุ่มค่า ดังนั้น ถ้าใช้ขั้นตอนวิธีนี้ ผลลัพธ์การค้นหาเหมือนเดิมทุกครั้ง ตัวอย่างขั้นตอนวิธีนี้เช่น ขั้นตอนวิธีการค้นหาแบบถึก (Brute-force Algorithm), ขั้นตอนวิธีแบบละโมภ (Greedy Algorithm) เป็นต้น

2) ขั้นตอนวิธีการค้นหาอย่างไม่แน่นอน (Stochastic Algorithm) เป็นการค้นหาตอบที่ไม่มีขั้นตอนการสุ่มค่า ดังนั้น ถ้าใช้ขั้นตอนวิธีนี้ ผลลัพธ์การค้นหาในแต่ละครั้งอาจจะไม่เท่ากัน ตัวอย่างขั้นตอนวิธีนี้เช่น ขั้นตอนวิธีการจำลองการอบเหนียว (Simulated Annealing Algorithm, SA), ขั้นตอนวิธีเชิงพันธุกรรม (Genetic Algorithm, GA), ขั้นตอนวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค Particle Swarm Optimization (PSO), ขั้นตอนวิธีหาค่าเหมาะสมที่สุดด้วยระบบอาณาจักรมด (Ant Colony Algorithm, ACO), ขั้นตอนวิธีหาค่าเหมาะสมที่สุดด้วยฝูงหมาป่า (Wolf Search Algorithm, WSA) เป็นต้น

## 2.4 ปัญหาการติดในจุดสุดสัมพัทธ์

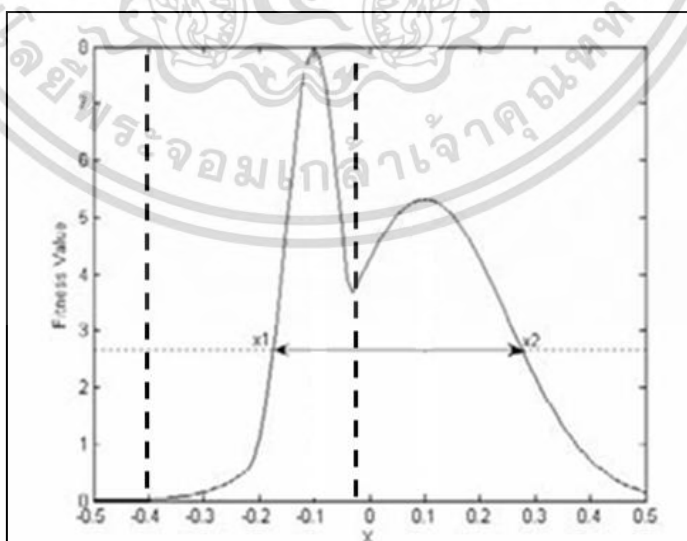
จุดสุดสัมพัทธ์ (Local Optimum) คือจุดที่อยู่บนฟังก์ชัน ซึ่งเป็นจุดที่มีค่าสูงสุดหรือต่ำสุด (จุดสูงสุดสัมพัทธ์หรือจุดต่ำสุดสัมพัทธ์) เมื่อเปรียบเทียบกับบริเวณข้างเคียงดังรูป 2.3 สำหรับจุดนี้อาจจะมีหรือไม่มีก็ได้ และอาจจะมีได้มากกว่าหนึ่งจุดต่อหนึ่งฟังก์ชัน

จุดสุดสัมบูรณ์ (Global Optimum) คือจุดที่อยู่บนฟังก์ชัน ซึ่งเป็นค่าสูงสุดหรือต่ำสุด (จุดสูงสุดสัมบูรณ์หรือจุดต่ำสุดสัมบูรณ์) เมื่อเปรียบเทียบกับฟังก์ชันดังรูป 2.3 สำหรับจุดนี้จะมีได้ไม่เกินหนึ่งจุดต่อหนึ่งฟังก์ชัน



รูปที่ 2.3 จุดสุดสัมพัทธ์และจุดสุดสัมบูรณ์

ปัญหาการติดในจุดสุดสัมพัทธ์ (Local Optimum Problem) คือ ตัวแทน (Agent) ของการค้นหาคาตอบนั้น จะค้นหาบริเวณรอบจุดสุดสัมพัทธ์ซ้ำๆ โดยไม่ออกไปค้นหาคาตอบในบริเวณอื่นๆ อีก ส่งผลให้คาตอบที่ได้นั้นไม่ดีขึ้นเลย ไม่ว่าจะค้นหาคาตอบต่อไปอีกยาวนานแค่ไหนก็ตาม ทำให้ไม่สามารถค้นหาจุดสุดสัมบูรณ์พบ โดยปัญหาการติดในจุดสุดสัมพัทธ์นี้ จะเกิดขึ้นกับปัญหาอัลติโมดอลเท่านั้น จากรูป 2.4 ถ้าต้องการค้นหาจุดสูงสุดสัมบูรณ์โดยขั้นตอนวิธีแบบละโมภ ถ้าตัวแทนเริ่มต้นค้นหาในช่วงระหว่างเส้นประ จะมีโอกาสพบจุดสุดสัมบูรณ์ แต่ถ้าตัวแทนเริ่มต้นค้นหาในช่วงบริเวณอื่นจะไม่มีโอกาสพบจุดสุดสัมบูรณ์ ส หรับสาเหตุของการเกิดปัญหาการติดในจุดสุดสัมพัทธ์ คือความซับซ้อนของปัญหาและขั้นตอนวิธีที่นำมาแก้ปัญหา แต่ในการแก้ปัญหา นั้น สิ่งที่ไม่สามารถทำได้ คือลดความซับซ้อนของปัญหา เพราะนั่นเท่ากับเปลี่ยนไปแก้ปัญหาอื่น ดังนั้นการแก้ปัญหาจึงจำเป็นต้องเพิ่มประสิทธิภาพของขั้นตอนวิธีที่นำมาแก้ปัญหา

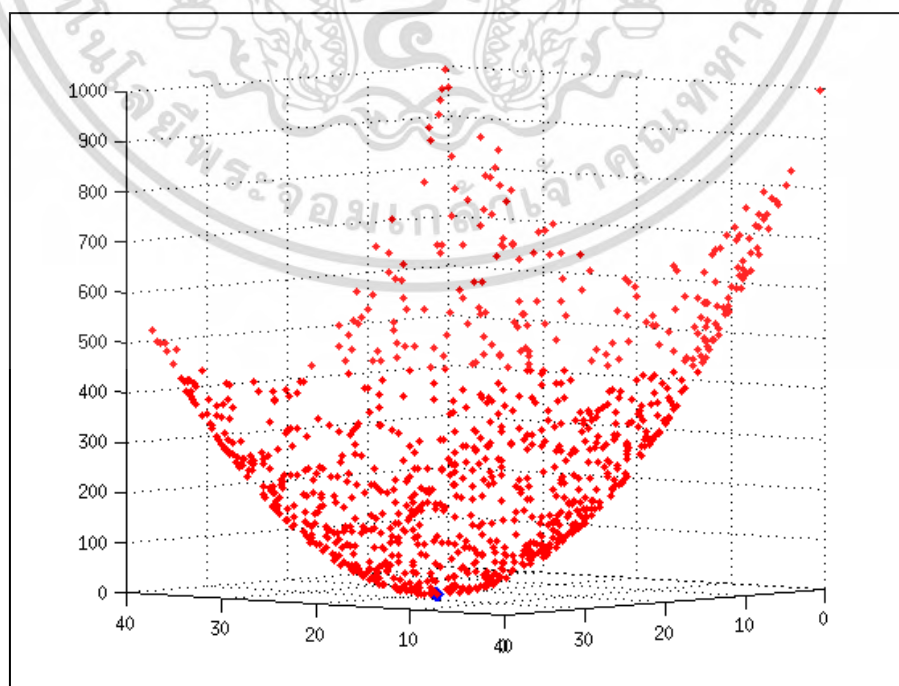


รูปที่ 2.4 ตัวอย่างของปัญหาอัลติโมดอลในมุมมองสองมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5 วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค

วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization หรือ PSO) คือ ขั้นตอนวิธีที่นำมาใช้กับการแก้ปัญหาการหาจุดที่ดีที่สุด ขั้นตอนวิธีนี้มีสมบัติ คือขั้นตอนวิธีพื้นฐาน ประชากร ขั้นตอนวิธีการค้นหาอย่างไม่แน่นอนและเป็นกลุ่มของขั้นตอนวิธีวิวัฒนาการ (Evolutionary Algorithm หรือ EA) โดย เจมส์ เคนเนดี และ รัสเซล อีเบอฮาท ในปี 1995 เป็นผู้คิดค้นขั้นตอนวิธีนี้ โดยพวกเขาได้แรงบันดาลใจมาจากพฤติกรรมการหาอาหารของฝูงสัตว์ เช่น ฝูงนก และฝูงปลา เป็นต้น และการสื่อสารแลกเปลี่ยนข้อมูลระหว่างพวกฝูงสัตว์เหล่านี้ โดยฝูงสัตว์เหล่านี้ จะทำการเคลื่อนที่เข้าหาแหล่งอาหาร แล้วส่งสัญญาณสื่อสารกับเพื่อนๆ ในฝูง เพื่อให้ทราบถึง ตำแหน่งที่มีอาหารอยู่ เพื่อที่จะท ให้เพื่อนๆ ในฝูงเคลื่อนที่ไปหาแหล่งอาหารนั้น แล้วไปค้นหาอาหาร ในบริเวณนั้น ซึ่งแนวคิดนี้ตัวแทนของสัตว์ที่ใช้ในการค้นหาเรียกว่า อนุภาค (Particles หรือ X) โดย อนุภาคนั้น จะมีจำนวนมากและเคลื่อนที่บนพื้นที่ที่ต้องการค้นหาคำตอบ (Solution Space) ซึ่งเป็น ตัวแทนของแหล่งอาหาร โดยอนุภาคจะต้องค้นหาคำตอบที่ดีที่สุด (Optimum) ซึ่งเป็นตัวแทนของ แหล่งอาหารที่มากที่สุด ดังนั้น PSO จึงสามารถประยุกต์ใช้กับปัญหาที่คำตอบสามารถแทนรูปแบบ ของปัญหาให้เป็นจุดบนปริภูมิขนาดหลายมิติได้ ดังรูป 2.5 โดยจุดต่างๆ แทนอนุภาคที่ถูกวางไว้ใน ปริภูมิขนาดหลายมิติหรือพื้นที่ที่ต้องการค้นหาคำตอบ พร้อมกับความเร็วต้นค่าหนึ่ง และสามารถ ติดต่อกับอนุภาคอื่นๆ อนุภาคจะเคลื่อนที่บนปริภูมิขนาดหลายมิติ โดยอนุภาคที่เข้าใกล้คำตอบที่ดี ที่สุดที่เคยค้นหามาทั้งหมด จะดึงดูดให้อนุภาคตัวอื่นเปลี่ยนแปลงเส้นทางการเคลื่อนที่ให้เคลื่อนที่เข้า มาหาอนุภาคตัวที่พบคำตอบที่ดีที่สุด



รูปที่ 2.5 ตำแหน่งอนุภาคในแกนสามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) อนุภาค (Particles) คือ ตัวแทนหรือคำตอบของปัญหาที่ต้องการค้นหา โดย 1 อนุภาค เท่ากับคำตอบของปัญหา 1 คำตอบ ถ้าจำนวนอนุภาคมากก็จะมีคำตอบมากขึ้นกัน โดยการกำหนด ตำแหน่งของอนุภาคนั้น จะถูกกำหนดให้อยู่ภายในขอบเขตของพื้นที่ที่ต้องการค้นหาคำตอบ และแต่ละอนุภาคจะมีองค์ประกอบดังนี้ คือความเร็วของอนุภาค ( $V$ ) ตำแหน่งของอนุภาค ( $X$ ) และตำแหน่ง ที่ดีที่สุดของอนุภาคแต่ละตัว (PBEST)

2) ประชากร (Population) คือ อนุภาคทั้งหมดที่ใช้เคลื่อนที่ เพื่อค้นหาคำตอบในพื้นที่ที่ต้องการค้นหาคำตอบ โดยจะมีการกำหนดจำนวนของอนุภาคหรือจำนวนประชากร ( $N_{pop}$ ) เอาไว้ล่วงหน้า และจะมีจำนวนคงที่ตลอดการค้นหา

3) รอบการค้นหาคำตอบ ( $l$ ) คือ จำนวนรอบที่ใช้ในการค้นหาคำตอบ เพราะ PSO จะค้นหา คำตอบ โดยทำงานแบบวนซ้ำไปเรื่อยๆ โดยรอบที่วนนั้นจะทำให้รอบการค้นหาคำตอบเพิ่มขึ้น จนถึงรอบการค้นหา คำตอบสูงสุด ( $l_{max}$ ) ที่ได้กำหนดไว้ให้หยุดการค้นหา

4) ความเร็วของอนุภาค (Velocity หรือ  $V$ ) คือ ขนาดและทิศทางที่อนุภาคจะเคลื่อนที่ไป จากตำแหน่งเดิมสู่ตำแหน่งใหม่

5) ความเร็วสูงสุด ( $V_{max}$ ) คือ ค่าสูงสุดที่มากที่สุดที่ความเร็วของอนุภาคจะมีค่าได้ เพื่อจำกัด ระยะทางที่อนุภาคจะสามารถเคลื่อนที่ได้มากที่สุดในรอบการค้นหา คำตอบ

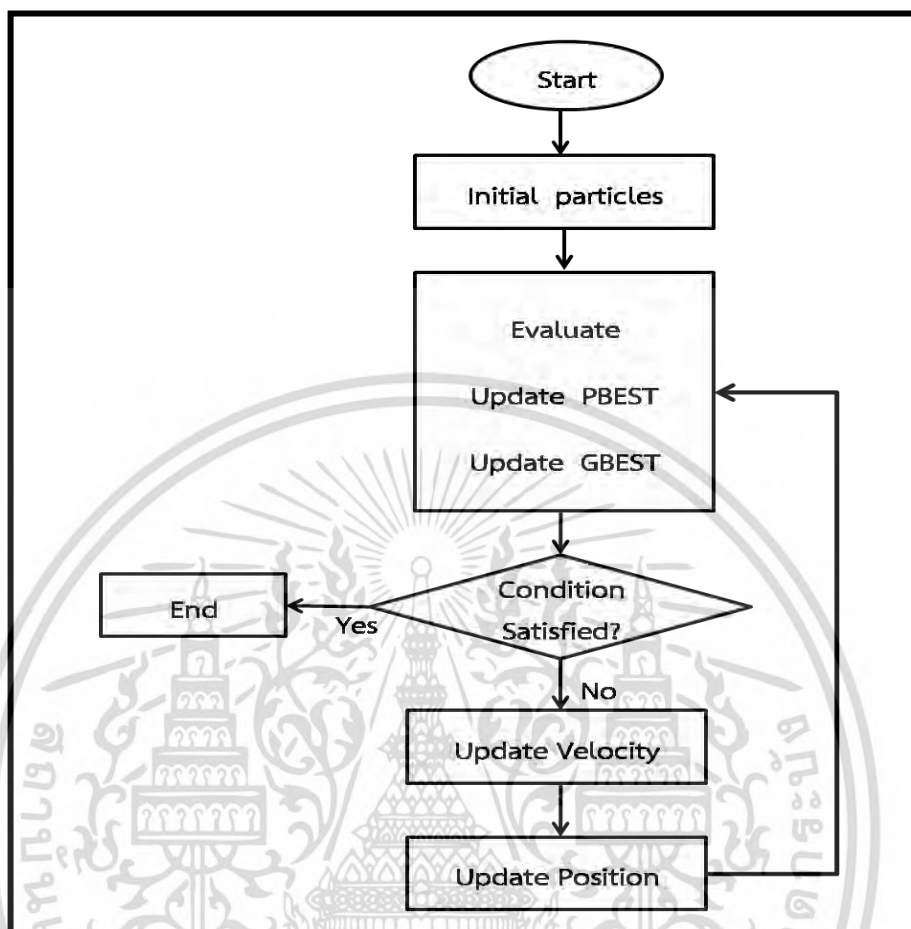
6) ตำแหน่งของอนุภาค (Position หรือ  $X$ ) คือ ตำแหน่งของอนุภาคในปริภูมิคำตอบ ดังรูป 2.9 แสดงตัวอย่างของตำแหน่งอนุภาคบนพื้นที่ที่ต้องการค้นหา คำตอบที่มีลักษณะเป็นสามมิติ

7) ตำแหน่งที่ดีที่สุดของอนุภาคแต่ละตัว (PBEST หรือ Personal Best Position) คือ ค่าความเหมาะสมที่ดีที่สุดของอนุภาคแต่ละตัวหรือตำแหน่งของอนุภาคที่มีค่าความเหมาะสมที่ดีที่สุดเท่าที่อนุภาคตัวนั้นๆ เคยทำการค้นหา คำตอบมาตั้งแต่เริ่มการทำงาน

8) ตำแหน่งที่ดีที่สุดของอนุภาคทั้งหมด (GBEST หรือ Global Best Position) คือ ค่าความเหมาะสมที่ดีที่สุดของอนุภาคทั้งหมดหรือตำแหน่งของอนุภาคที่มีค่าความเหมาะสมดีที่สุดเท่าที่อนุภาคทั้งหมดเคยทำการค้นหา มาตั้งแต่เริ่มการทำงาน

9) ฟังก์ชันคำนวณค่าความเหมาะสม (Fitness Function) คือ ฟังก์ชันเป้าหมาย (Objective Function) โดยเป็นฟังก์ชันที่ใช้คำนวณค่าความเหมาะสม (Fitness) ของแต่ละอนุภาค เพื่อใช้เปรียบเทียบและหาอนุภาคที่ดีที่สุด ซึ่งฟังก์ชันนี้จะแตกต่างกันขึ้นอยู่กับปัญหาที่ต้องการค้นหา คำตอบ

## 2.6 หลักการทำงานของ PSO



รูปที่ 2.6 แผนภาพการทำงานของ PSO

1) กระบวนการกำหนดค่าเริ่มต้น (Initial Particles) เป็นกระบวนการที่ทำงานเพียงครั้งเดียวและทำงานเป็นขั้นตอนแรก โดยสุ่มตำแหน่งและความเร็วเริ่มต้นของอนุภาคทั้งหมดลงบนพื้นที่ที่ต้องการค้นหาคำตอบ ตามจำนวนประชากรที่ได้กำหนดเอาไว้ สำหรับจำนวนประชากรถ้ามีจำนวนมากการค้นหาคำตอบจะทำได้ในบริเวณกว้างทำให้มีโอกาสพบคำตอบได้ง่าย และทำให้โอกาสในการเกิดปัญหาการติดในจุดสุดสัมพัทธ์น้อยลง แต่ต้องใช้เวลาในการค้นหาแต่ละรอบเพิ่มมากขึ้น จำนวนประชากรของนั้นจะไม่มีเพิ่มหรือลดจำนวนตลอดการค้นหาคำตอบ ส่วนความเร็วของอนุภาคนั้นจะเริ่มต้นด้วยการสุ่มความเร็วเริ่มต้นขึ้นมา ความเร็วนี้จะต้องไม่เกินค่าของความเร็วสูงสุดที่กำหนดเอาไว้ แต่ถ้าเกินค่าของความเร็วสูงสุดที่กำหนดเอาไว้ ความเร็วจะถูกแทนที่ด้วยค่าของความเร็วสูงสุด

2) กระบวนการประเมินค่าตำแหน่งของแต่ละอนุภาค (Evaluate) เป็นกระบวนการคำนวณค่าความเหมาะสมของตำแหน่งปัจจุบันของอนุภาคแต่ละตัว โดยค่าความเหมาะสมเหล่านี้จะถูกใช้ในการปรับปรุงค่า PBEST และ GBEST และพิจารณาหยุดการค้นหาคำตอบ โดยถ้าค่าความเหมาะสมของอนุภาคนี้น่าพอใจแล้วก็หยุดการค้นหาคำตอบ นอกจากนั้นก็จะเป็นขั้นตอนต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนักเรียนไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) กระบวนการปรับปรุง PBEST (Update PBEST) เป็นการปรับปรุง PBEST โดยแต่ละอนุภาคจะเปรียบเทียบค่าความเหมาะสมของอนุภาคกับค่าความเหมาะสมของ PBEST ของตัวมันเอง ถ้าค่าความเหมาะสมของอนุภาคดีกว่าค่าความเหมาะสมของ PBEST แล้ว PBEST จะถูกปรับปรุงให้กลายเป็นอนุภาคนั้น สำหรับ PBEST นั้นจะส่งผลต่อการคำนวณความเร็วและทิศทางใหม่ให้กับอนุภาค เพื่อให้อนุภาคเหล่านั้นเคลื่อนที่ไปยังตำแหน่งใหม่

4) กระบวนการเปลี่ยนแปลง GBEST (Update GBEST) เป็นการปรับปรุง GBEST โดยเปรียบเทียบค่าความเหมาะสมของอนุภาคทั้งหมดกับค่าความเหมาะสมของ GBEST ถ้าค่าความเหมาะสมของอนุภาคตัวใด ดีกว่าค่าความเหมาะสมของ GBEST แล้ว GBEST จะถูกปรับปรุงให้กลายเป็นอนุภาคนั้น สำหรับ GBEST นั้นจะส่งผลต่อการคำนวณความเร็วและทิศทางใหม่ให้กับอนุภาค เพื่อให้อนุภาคเหล่านั้นเคลื่อนที่ไปยังตำแหน่งใหม่

5) กระบวนการคำนวณความเร็ว (Calculate Velocity) เป็นกระบวนการที่ทำการคำนวณความเร็วของแต่ละอนุภาค เพื่อใช้ในการเคลื่อนที่อนุภาคแต่ละตัวในแต่ละรอบ ซึ่งความเร็วนี้จะถูกเปลี่ยนแปลง โดยใช้ข้อมูลของความเร็วเดิมของอนุภาคแต่ละตัว PBEST ของแต่ละตัว และ GBEST โดยมีสมการในการคำนวณดังนี้

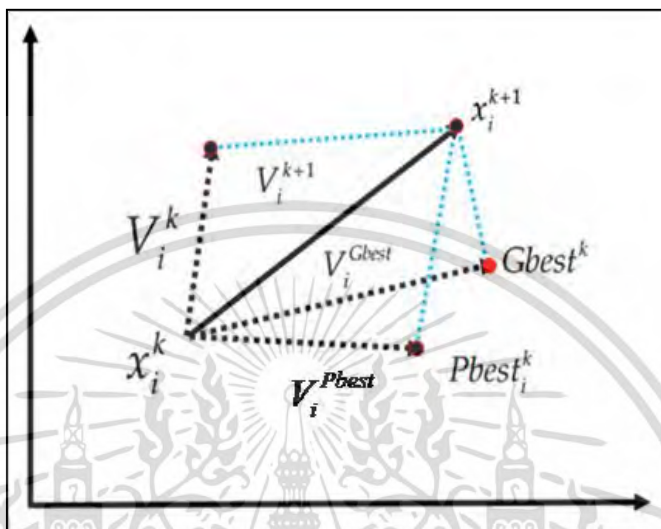
$$\vec{v}_i(t+1) = \omega \vec{v}_i(t) + c_1 \phi_1 (\vec{p}_i(t) - \vec{x}_i(t)) + c_2 \phi_2 (\vec{p}_g(t) - \vec{x}_i(t)) \quad (2.1)$$

โดยกำหนดให้

$\vec{v}_i(t+1)$	คือ ความเร็วในรอบปัจจุบันเป็นเวกเตอร์ที่มีขนาดและทิศทาง
$\vec{v}_i(t)$	คือ ความเร็วในรอบก่อนหน้าเป็นเวกเตอร์ที่มีขนาดและทิศทาง
$c_1, c_2$	คือ ค่าคงที่ที่กำหนดให้เป็นสัมประสิทธิ์
$\phi_1, \phi_2$	คือ ตัวเลขที่ทำการสุ่มอยู่ในช่วง $[0, 1]$
$\omega$	คือ ค่าความเฉื่อยที่กำหนดให้เป็นสัมประสิทธิ์
$\vec{p}_i(t)$	คือ ตำแหน่งที่ดีที่สุดของอนุภาคแต่ละตัว (PBEST)
$\vec{p}_g(t)$	คือ ตำแหน่งที่ดีที่สุดของอนุภาคทั้งหมด (GBEST)
$\vec{x}_i(t)$	คือ ตำแหน่งในปัจจุบันของอนุภาค $i$ (X)

โดยค่าของ  $\vec{v}_i$  จะต้องไม่เกินค่าของความเร็วสูงสุด การเปลี่ยนความเร็วในลักษณะนี้จะทำให้อนุภาค  $i$  สามารถค้นหาคำตอบที่อยู่รอบๆ PBEST และ GBEST ได้ดังรูปที่ 2.11 แสดงตัวอย่างการคำนวณทิศทางความเร็วของอนุภาคในแกนสองมิติ

จากสมการที่ (2.1) พารามิเตอร์  $\omega$  คือ ความเฉื่อยซึ่งเป็นตัวแปรที่มีอิทธิพลควบคุมความเร็วในรอบก่อนหน้าที่ส่งผลต่อการเคลื่อนที่ของอนุภาคในรอบปัจจุบัน  $c_1$  คือ ค่าคงที่ซึ่งเป็นตัวกำหนดแรงดึงดูดจากอิทธิพลของ PBest  $c_2$  คือ ค่าคงที่ซึ่งเป็นตัวกำหนดแรงดึงดูดจากอิทธิพลของ GBest ซึ่งผลจากสมการนี้ ทำให้อนุภาคเคลื่อนที่ไปยังตำแหน่งที่ดีขึ้น



รูปที่ 2.7 ตัวอย่างทิศทางการเคลื่อนที่ของอนุภาคในแกนสองมิติ

รูปที่ 2.7 กำหนดให้ทิศทางของความเร็วในรอบก่อนหน้า  $\vec{v}_i^k$  มีทิศทางเป็นเวกเตอร์สุ่มสุ่มขึ้น สำหรับการคำนวณทิศทางของ PBest ( $Pbest_i^k$ ) ให้นำตำแหน่งของ PBest ลบกับตำแหน่งของอนุภาค ทำให้เกิดเวกเตอร์  $V_i^{Pbest}$  และการคำนวณทิศทางของ GBest ( $Gbest^k$ ) ให้นำตำแหน่งของ GBest ลบกับตำแหน่งของอนุภาค ทำให้เกิดเวกเตอร์  $V_i^{Gbest}$  และเมื่อนำเวกเตอร์ทั้งสามมารวมกันจะได้ทิศทางลัพท์ตั้งเวกเตอร์  $x_i^{k+1}$  คือความเร็วในรอบปัจจุบัน  $V_i^{k+1}$  ซึ่งเป็นทิศทางการเคลื่อนที่ใหม่ของอนุภาค

6) กระบวนการปรับปรุงตำแหน่ง (Update Position) คือการปรับปรุงตำแหน่งของอนุภาคแต่ละอนุภาค โดยอนุภาคแต่ละอนุภาคนั้น จะเคลื่อนที่จากตำแหน่งเดิมไปยังตำแหน่งใหม่ ด้วยระยะที่คำนวณได้จากค่าความเร็วของอนุภาคนั้น โดยมีสมการดังนี้

$$\vec{x}_i(t+1) = \vec{x}_i(t) + \vec{v}_i(t+1) \quad (2.2)$$

โดยกำหนดให้

$\bar{x}_i(t+1)$  คือ ตำแหน่งใหม่ที่อนุภาคจะทำการเคลื่อนที่ไป

$\bar{x}_i(t)$  คือ ตำแหน่งในปัจจุบันของอนุภาค  $i$

$\bar{v}_i(t+1)$  คือ ความเร็วใหม่ที่คำนวณได้

หลังจากกระบวนการปรับปรุงตำแหน่ง ก็จะวนกลับไปทำกระบวนการประเมินค่าตำแหน่งของแต่ละอนุภาคในรอบถัดไป โดยหลักการทางานของ PSO ในช่วงแรกของการค้นหาค่าตอบอนุภาคจะอยู่ห่างกันมาก ทำให้ระยะห่างระหว่าง GBEST กับอนุภาคมีค่ามาก ส่งผลให้อนุภาคแต่ละตัวนั้นเคลื่อนที่เข้าหากันอย่างรวดเร็ว ทำให้ไม่ต้องสูญเสียเวลาค้นหาอย่างละเอียดในบริเวณที่ค่าความเหมาะสมที่มีค่าน้อย เมื่อการค้นหาเกิดการต่อไปเรื่อยๆ อนุภาคจะเคลื่อนที่เข้าใกล้กันมากขึ้น และส่งผลให้ผลต่างระหว่าง GBEST กับอนุภาคมีค่าน้อยลง ทำให้อนุภาคเคลื่อนที่เข้าหากันช้าลง แล้วอนุภาคจะค้นหาบริเวณรอบ GBEST อย่างละเอียด ซึ่งโดยปกติแล้ว ในบริเวณรอบค่าตอบที่ดีนั้นมีโอกาส จะมีโอกาสได้พบค่าตอบที่ดีขึ้นกว่าเดิม

## 2.7 ข้อดีและข้อเสียของ PSO

### 2.7.1 ข้อดีของ PSO

1) PSO สามารถประยุกต์ใช้กับปัญหาการหาจุดที่ดีที่สุดที่มีขนาดใหญ่ โดยใช้เวลาค้นหาไม่นาน โดยปกติปัญหาการหาจุดที่ดีที่สุดที่มีขนาดใหญ่ นั้นจะมีคำตอบจำนวนมาก หากไล่คำตอบทั้งหมดอาจจะเป็นไปไม่ได้ หลักการทางานของ PSO จะค้นหาเฉพาะบริเวณที่คิดว่าน่าจะพบคำตอบที่ดีอย่างละเอียดเท่านั้น จึงเป็นการลดขอบเขตการค้นหา และสามารถค้นหาได้อย่างมีประสิทธิภาพ โดยปกติแล้วในบริเวณที่พบคำตอบที่ดี จะมีโอกาสสูงที่จะได้พบคำตอบที่ดีขึ้นกว่าเดิม ซึ่งคำตอบที่ได้จากการค้นหาของ PSO จะมีค่าที่ดีในระดับหนึ่ง แต่ PSO ไม่สามารถรับประกันว่าคำตอบที่ได้จากการค้นหาจะเป็นค่าตอบที่ดีที่สุด

2) การคำนวณของ PSO มีประสิทธิภาพที่ดี เพราะ PSO จะไม่ค้นหาคำตอบทั้งหมด แต่จะค้นหาคำตอบเฉพาะบริเวณที่พบคำตอบที่ดีเท่านั้น ซึ่งในบริเวณนั้น จะมีโอกาสพบคำตอบที่ดีขึ้นได้มาก

3) พารามิเตอร์ของ PSO มีจำนวนน้อยที่ต้องการการกำหนดค่า โดยจากขั้นตอนการทางานของ PSO ที่ได้กล่าวไปแล้วนั้น จำนวนพารามิเตอร์ของ PSO ที่ต้องมีการกำหนดค่า คือ  $c_1$ ,  $c_2$ ,  $\omega$ ,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนประชากร, จำนวนรอบการค้นหาคาดตอบ เนื่องจาก PSO มีจำนวนพารามิเตอร์น้อย จึงทำให้สามารถนำ PSO ไปประยุกต์ใช้ในทางปฏิบัติได้ง่าย

4) ขั้นตอนการทำงานของ PSO มีรูปแบบไม่ซับซ้อน ทำให้ง่ายต่อการศึกษาและการเขียนพัฒนาโปรแกรม และการประยุกต์ใช้งาน โดย PSO สามารถประยุกต์ใช้กับปัญหาที่มีรูปแบบจำนวนจริง ได้อย่างมีประสิทธิภาพ เนื่องจาก PSO สามารถประยุกต์ใช้กับปัญหาที่มีรูปแบบจำนวนจริง จึงทำให้นักวิจัยนิยมนำมาประยุกต์ใช้กับปัญหาการหาจุดที่ดีที่สุดในงานวิจัยด้านวิทยาศาสตร์ และวิศวกรรม เพราะงานด้านวิทยาศาสตร์และวิศวกรรม เพราะส่วนมากปัญหาเหล่านี้มีรูปแบบเป็นจำนวนจริง ตัวอย่างที่นำ PSO ไปประยุกต์ใช้เช่น ปัญหาฟังก์ชันการหาจุดที่ดีที่สุด (Function Optimization) ปัญหาการฝึกอบรมเครือข่ายประสาทเทียม (Artificial Neural Network Training) ปัญหาการจัดกลุ่มและการควบคุมระบบซับซ้อน (Pattern Classification and Fuzzy System Control)

5) การค้นหาคาดตอบของ PSO มีอัตราการลู่เข้าหาค่า ทอบที่เร็วกว่า และได้รับค่า ทอบที่ดีกว่า ถ้าหากเปรียบเทียบขั้นตอนวิธีอื่นๆ ที่มีการค้นหาข้อมูลแบบกลุ่มที่มีลักษณะพื้นฐานประชากรและมีการค้นหาอย่างไม่แน่นอน (Population-based Stochastic Optimization Algorithm) ตัวอย่างเช่น GA, ACO, WSA เป็นต้น

## 2.7.2 ข้อเสียของ PSO

1) ข้อเสียหลักของ PSO คือ PSO สามารถเกิดปัญหาการติดในจุดสุดสัมพัทธ์ได้ง่าย เพราะผลจากขั้นตอนกระบวนการปรับปรุงตำแหน่งและขั้นตอนกระบวนการคำนวณความเร็วจากสมการ (2.2) ทำให้อนุภาคต่างๆ ลู่เข้าหา GBEST ถ้าหาก GBEST ติดอยู่ในจุดสุดสัมพัทธ์ แล้วไม่มีอนุภาคใดที่สามารถหาตำแหน่งที่ดีกว่า GBEST อนุภาคเหล่านี้ทั้งหมดจะลู่เข้าหา GBEST เมื่ออนุภาคทั้งหมดเคลื่อนที่ไปอยู่บริเวณรอบๆ GBEST ก็เสมือน อนุภาคทั้งหมดเคลื่อนที่ไปติดจุดสุดสัมพัทธ์ เมื่ออนุภาคอยู่ใกล้ GBEST มากขึ้น โดยจากสมการ (2.2) ผลต่างระยะทางระหว่างอนุภาค PBEST และ GBEST มีค่าลดลงเรื่อยๆ เมื่อผลต่างเหล่านี้ มีค่าลดลงส่งผลให้ความเร็วของอนุภาคมีค่าลดลงเรื่อยๆ สามารถสังเกตได้จากกราฟ 2.8 ซึ่งแสดงค่าความเร็วของอนุภาค โดยกราฟอ้างอิงมาจากการทดลอง 5.1 ของ [45] จะเห็นว่าความเร็วของอนุภาคในช่วงแรกมีค่ามากแล้ว เมื่อเวลาผ่านไปเรื่อยๆ ความเร็วของอนุภาคจะค่าลดลงจนเกือบเป็นศูนย์ ผลจากความเร็วจนอนุภาคมีค่าลดลงเรื่อยๆ ทำให้อนุภาคเคลื่อนที่ได้ช้าลงเรื่อยๆ และค้นหาที่จะแคบลงเรื่อยๆ ถ้าหากบริเวณนั้นคือจุดสุดสัมพัทธ์ อนุภาคก็จะค้นหารอบจุดสุดสัมพัทธ์เท่านั้น ไม่ออกไปค้นหาในบริเวณอื่นอีก จึงไม่สามารถหาจุดสุดสัมบูรณ์พบ

เมื่อ PSO ติดในจุดสุดสัมพัทธ์แล้ว GBEST จะไม่มีการเปลี่ยนแปลงสามารถสังเกตได้จาก

กราฟ 2.9 ซึ่งแสดงค่า GBEST ของ PSO โดยจากกราฟ GBEST มีค่าคงที่หลังจากการค้นหาไปสักเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระยะเวลาหนึ่ง แล้วไม่มีแนวโน้มที่จะดีขึ้นไม่ว่าจะค้นหาต่อไปอีกยาวนานแค่ไหนก็ตาม โดยกราฟอ้างอิงมาจากการทดลอง 5.2 ของ [45] โดยจากกราฟไม่ว่าจะดำเนินการค้นหาต่อไปอีกยาวนานแค่ไหน และเมื่อเกิดปัญหาการติดในจุดสุดสัมพัทธ์นี้ขึ้นแล้ว PSO นั้นไม่มีวิธีจัดการกับแก้ปัญหาการติดในจุดสุดสัมพัทธ์ จึงทำให้ไม่สามารถหาจุดสุดสัมบูรณ์ได้อย่างแน่นอน ดังนั้นเมื่อ PSO เกิดปัญหาการติดในจุดสุดสัมพัทธ์ขึ้น ก็เสมือนว่าการค้นหาด้วย PSO นั้นสิ้นสุดลง การค้นหาต่อไปนั้นจะทำให้สิ้นเปลืองทรัพยากรโดยเปล่าประโยชน์

สำหรับองค์ประกอบของการเกิดปัญหาการติดในจุดสุดสัมพัทธ์กับ PSO นั้นมี 2 องค์ประกอบ คือ ขั้นตอนวิธีที่นำมาแก้ปัญหาค้นหาจุดที่ดีที่สุดกับความซับซ้อนของปัญหาการหาจุดที่ดีที่สุด สำหรับความซับซ้อนของปัญหาการหาจุดที่ดีที่สุดนั้น มีผลต่อการติดในจุดสุดสัมพัทธ์ของ PSO โดยปัญหาชนิดใด PSO จะไม่เกิดปัญหาการติดในจุดสุดสัมพัทธ์ และปัญหาลักษณะนี้ PSO สามารถแก้ไขได้ดี เพราะ PSO มีคุณสมบัติการลู่เข้าหาค่าตอบอย่างรวดเร็ว ทำให้พบคำตอบได้อย่างรวดเร็ว และพบคำตอบที่ดีที่สุดได้อย่างแน่นอน แต่ปัญหามัลติโมดอล PSO มีโอกาสมากที่จะติดในจุดสุดสัมพัทธ์ แต่ PSO นั้นก็ไม่ใช่ว่าจะแก้ปัญหาลักษณะนี้ไม่ได้ เพราะถ้าหากความซับซ้อนของปัญหาไม่ได้มากมายหรือจำนวนประชากรของ PSO เพียงพอ PSO นั้นก็จะสามารถหาคำตอบที่ดีที่สุดพบ

## 2.8 ขั้นตอนวิธีเชิงพันธุกรรม

ขั้นตอนวิธีเชิงพันธุกรรม (Genetic Algorithm, GA) ถูกคิดค้นโดย จอห์น ฮอลแลนด์ (John Holland) มีแนวคิดของขั้นตอนวิธีนี้มาจากกระบวนการตามธรรมชาติสำหรับพันธุกรรมของสิ่งมีชีวิต หรือเรียกว่า "ทฤษฎีวิวัฒนาการทางพันธุกรรม" ของ Charles Darwin ซึ่งเป็นทฤษฎีที่อธิบายเกี่ยวกับวัฏจักรของพันธุกรรมว่า เกิดจากความหลากหลายทางพันธุกรรมที่ได้รับการคัดเลือกตามหลักของธรรมชาติ คือสิ่งมีชีวิตที่สามารถปรับตัวให้เข้ากับสภาพแวดล้อมได้ จะมีโอกาสอยู่รอดมากกว่าสิ่งมีชีวิตที่ไม่สามารถปรับตัวได้ เรียกว่า "The Survival of the Fittest" และเมื่อสิ่งมีชีวิตที่ปรับตัวให้เหมาะสมกับสภาพแวดล้อมนั้นผสมพันธุ์จะเกิดลูกหลานออกมา และลูกหลานเหล่านี้จะได้รับสืบทอดวิวัฒนาการให้มีความเหมาะสมกับสภาพแวดล้อมต่อไป จอห์น ฮอลแลนด์ ได้ค้นพบแนวคิดมาปรับใช้จนกลายเป็นวิธีการทางพันธุกรรม เพื่อใช้แก้ปัญหาค้นหาจุดที่ดีที่สุด ซึ่งปัญหานี้ได้รับการยอมรับและถูกใช้อย่างแพร่หลาย

### 2.8.1 นิยามที่ควรทราบเกี่ยวกับขั้นตอนวิธีเชิงพันธุกรรม

1) โครโมโซม (Chromosome) คือ การแทนรูปแบบคำตอบของปัญหาที่ต้องการใช้กับ GA โดยผ่านการเข้ารหัสโครโมโซม (Chromosome Encoding) รูปแบบของโครโมโซมจะแตกต่างกัน ออกไปขึ้นอยู่กับปัญหาที่ต้องการหาคำตอบ

2) ประชากร (Population) คือ กลุ่มของโครโมโซมทั้งหมดที่ใช้ใช้เป็นพ่อแม่พันธุ์ เพื่อ ค้นหาคำตอบในพื้นที่ที่ต้องการค้นหา โดยจะมีการกำหนดจำนวนประชากร ( $N_{pop}$ ) เอาไว้ ล่วงหน้า และจะมีจำนวนคงที่ตลอดการค้นหา และใช้เป็นตัวเริ่มการทำงานของกระบวนการทาง พันธุกรรมรอบต่อไป

3) รอบการค้นหาคำตอบ (I) คือ จำนวนรอบที่ใช้ในการค้นหาคำตอบ เพราะ GA จะมี ขั้นตอนการค้นหาคำตอบเหมือน PSO คือจะทำงานแบบวนซ้ำไปเรื่อยๆ โดยรอบที่วนนั้นจะทำให้ รอบการค้นหาคำตอบเพิ่มขึ้น จนถึงรอบการค้นหาคำตอบสูงสุด ( $I_{max}$ ) ที่ได้กำหนดไว้ให้หยุดการ ค้นหา

4) ค่าความน่าจะเป็นในการกหนดค่าประชากรเริ่มต้น คือ ความน่าจะเป็นในการกหนดค่า แต่ละบิตของโครโมโซมของประชากร โดยถ้าค่านี้มีค่ามากประชากรเริ่มต้น จะมีจำนวนบิตที่มีค่าเป็น 1 จำนวนมาก

5) ค่าความน่าจะเป็นในการผสมพันธุ์ คือ โอกาสที่จะเกิดการผสมพันธุ์ในแต่ละบิตของ โครโมโซม โดยค่าความน่าจะเป็นในการผสมพันธุ์ เป็นค่าที่ต้องกำหนดไว้ล่วงหน้า โดยค่าที่กำหนด ขึ้นอยู่กับความเหมาะสมของปัญหาที่ต้องการค้นหาคำตอบ

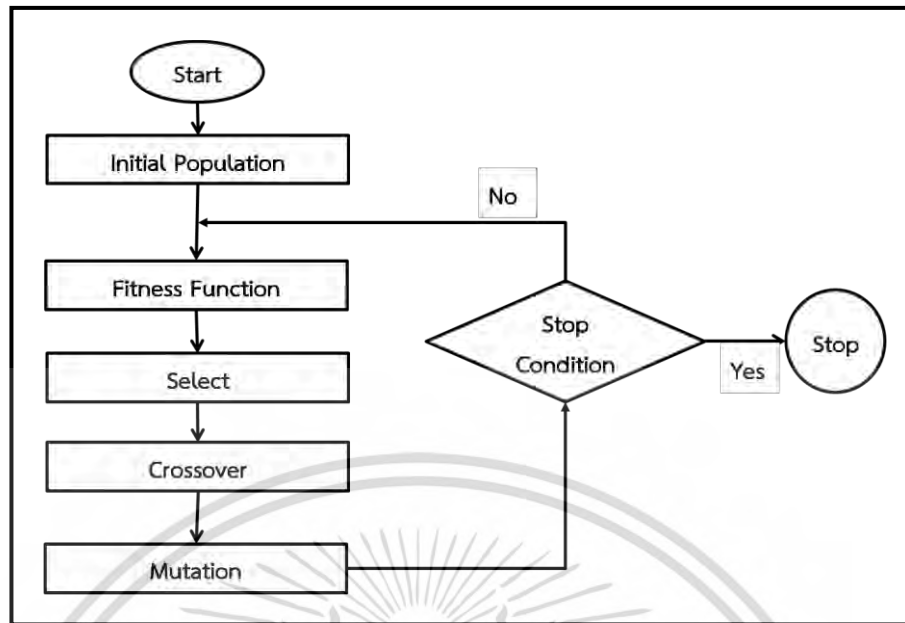
6) ค่าความน่าจะเป็นในการกลายพันธุ์ คือ โอกาสการเกิดการกลายพันธุ์ขึ้นในแต่ละบิตของ โครโมโซม เป็นค่าที่ต้องกหนดไว้ล่วงหน้า โดยส่วนมากแล้วจะกหนดให้มีค่าน้อย

7) โครโมโซมที่ดีที่สุด (The Best Chromosome) คือ โครโมโซมที่มีค่าค่าความเหมาะสมที่ ดีที่สุดในตลอดช่วงเวลาที่ค้นหา คำตอบ

8) ฟังก์ชันคำนวณค่าความเหมาะสม (Fitness Function) คือ ฟังก์ชันเป้าหมาย โดยเป็น ฟังก์ชันที่ใช้คำนวณค่าความเหมาะสม (Fitness) ของแต่ละโครโมโซม เพื่อใช้เปรียบเทียบและหาโครโมโซมที่ดีที่สุด ซึ่งฟังก์ชันนี้จะแตกต่างกันขึ้นอยู่กับปัญหาที่ต้องการค้นหา คำตอบ

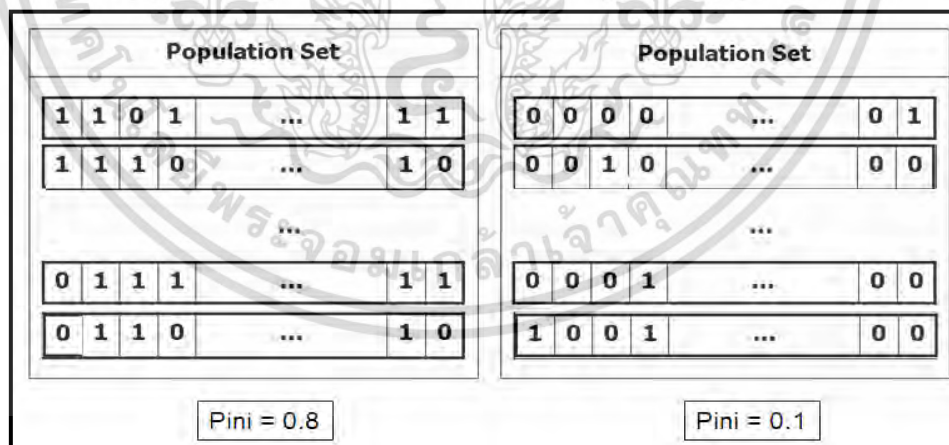
### 2.8.2 รายละเอียดขั้นตอนการทำงานของขั้นตอนวิธีเชิงพันธุกรรม

GA นั้นเป็นการค้นหาที่อาศัยหลักการทางพันธุกรรม ในการค้นหาคำตอบ ซึ่งเป็น กระบวนการที่สามารถนำไปประยุกต์ใช้ได้กับปัญหาหลากหลายแบบขึ้นอยู่กับ การประยุกต์ใช้ GA ใน รูปแบบของโครโมโซมที่เหมาะสมกับปัญหา โดยขั้นตอนการทำงานหลักดังแสดงในรูปที่ 2.8



รูปที่ 2.8 แผนภาพการท งานของขั้นตอนวิธีเชิงพันธุกรรม

1) กระบวนการกำหนดค่าเริ่มต้น (Initial Process) เป็นกระบวนการที่ท งานเป็นขั้นตอนแรกและทงานเพียงครั้งเดียว โดยทการเข้ารหัสโครโมโซมเพื่อกาหนดให้คาคอบของปัญหาอยู่ในรูปแบบโครโมโซม จากนั้นทการสร้างโครโมโซมตามจนวนประชากรที่กาหนด และเรียกกลุ่มประชากรนี้ว่าประชากรเริ่มต้น ดังรูปที่ 2.9

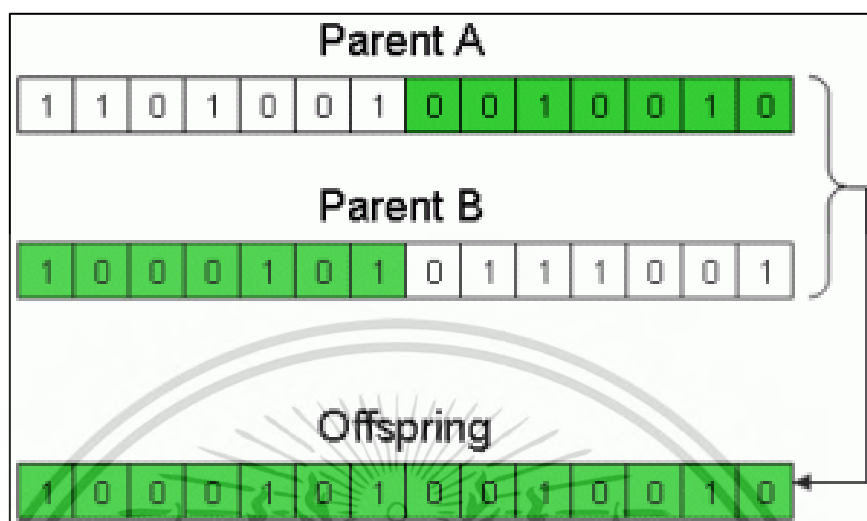


รูปที่ 2.9 การก าหนดค่าเริ่มต้นที่ไม่มีการ ก าหนดค่าเริ่มต้นให้กับ PBEST และ GBEST

2) กระบวนการผสมพันธุ์ (Crossover Process) เป็นกระบวนการที่ทการสุ่มเลือกโครโมโซมจากประชากรมาใช้เป็นพ่อ-แม่พันธุ์เพื่อสร้างโครโมโซมรุ่นลูก โดยโครโมโซมที่มีค่าความเหมาะสมมากจะมีโอกาสถูกสุ่มเลือกเป็นพ่อ-แม่พันธุ์มากกว่าโครโมโซมที่มีค่าความเหมาะสมน้อย

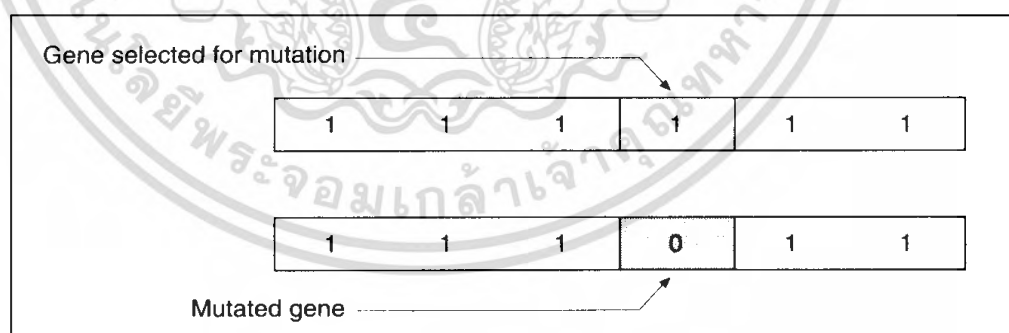
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อได้โครโมโซมพ่อแม่แล้ว จะนำโครโมโซมทั้งสองมาทำการผสมพันธุ์ด้วยค่าความน่าจะเป็นในการผสมพันธุ์ซึ่งกำหนดไว้ก่อนหน้า ดังรูปที่ 2.10



รูปที่ 2.10 ตัวอย่างการสุ่มกหนดค่าเริ่มต้นให้กับแต่ละบิตของโครโมโซมในประชากร

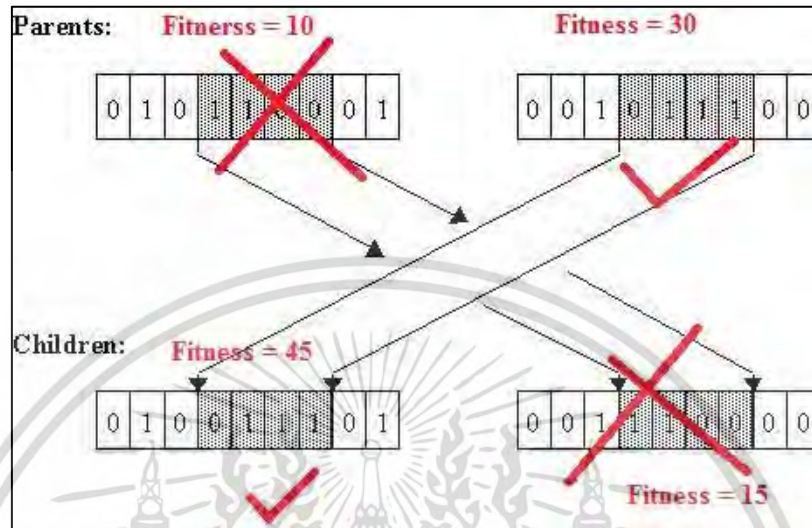
3) กระบวนการกลายพันธุ์ (Mutation Process) เป็นกระบวนการที่ทำให้โครโมโซมของประชากรรุ่นลูกมีโอกาสเปลี่ยนแปลงค่าบิตภายในโครโมโซมด้วยค่าความน่าจะเป็นในการกลายพันธุ์ที่กำหนดเอาไว้ก่อนหน้า โดยมากจะกำหนดให้ค่าความน่าจะเป็นในการกลายพันธุ์ จะมีค่าน้อยมาก การกลายพันธุ์เป็นกระบวนการที่งานโดยอาศัยการสุ่มค่าเพียงอย่างเดียวในการเปลี่ยนแปลงค่าของโครโมโซมรุ่นลูก ดังรูปที่ 2.11



รูปที่ 2.11 ตัวอย่างการกระบวนการกลายพันธุ์

4) กระบวนการคัดสรรประชากรรุ่นถัดไป (Selection Process) เป็นกระบวนการที่เลือกเก็บโครโมโซมบางส่วนเพื่อใช้เป็นพ่อแม่พันธุ์รุ่นถัดไป โดยเมื่อประชากรผ่านขั้นตอนการผสมพันธุ์ จำนวนประชากรจะมีจำนวนเพิ่มขึ้นเท่ากัน ดังนั้นประชากรที่มีคุณภาพต่ำจะถูกคัดออก แล้วประชากรที่มีคุณภาพดี จะถูกเก็บไว้เพื่อเป็นพ่อแม่พันธุ์ในรุ่นถัดไป โดยจะคัดเลือกให้เหลือจำนวนประชากรเท่ากับจำนวนประชากรตอนเริ่มต้น หลังจากนั้นให้การพิจารณาหยุดการค้นหาก่อนหน้านี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำตอบ โดยถ้าค่าความเหมาะสมของประชากรนี้เป็นที่น่าพอใจแล้วก็จะหยุดการค้นหาคำตอบ นอกจากนั้นก็จะดำเนินขั้นตอนต่อไป หลังจากกระบวนการคัดสรรประชากรรุ่นถัดไป แล้วจะวนกลับไปทวนกระบวนการผสมพันธุ์ในรอบถัดไป ดังรูปที่ 2.12



รูปที่ 2.12 ตัวอย่างกระบวนการคัดสรรประชากรรุ่นถัดไป

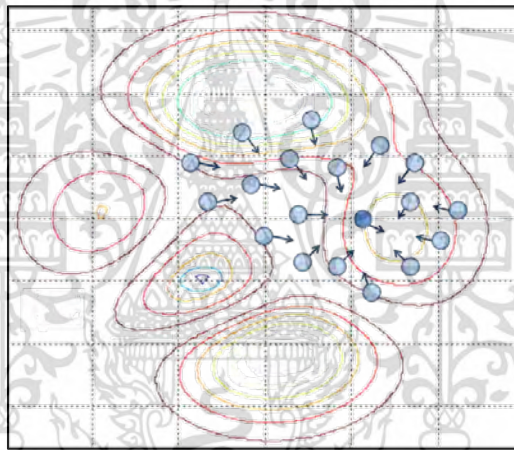
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ 3 งานวิจัยที่เกี่ยวข้อง

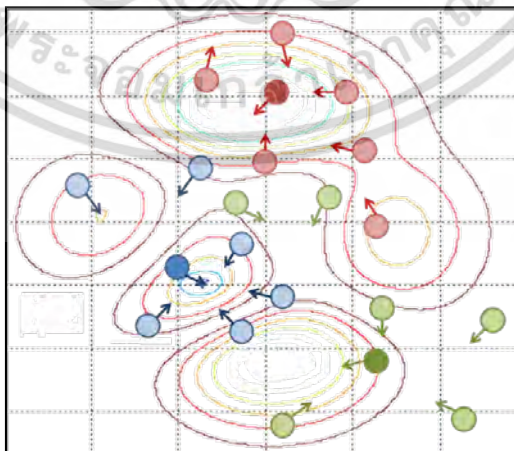
#### 3.1) วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม

##### 3.1.1) วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม

ปัญหาหลักของ PSO ที่เกิดขึ้นเมื่ออนุภาคภายในกลุ่มเคลื่อนที่ค้นหาและลู่เข้าหาอนุภาคตัวที่ดีที่สุดของกลุ่ม (GBEST) ดังรูปที่ 3.1 แล้วไม่สามารถหาคำตอบที่ดีกว่าได้จนติดในจุดสุดสัมพัทธ์ ทำให้มีนักวิจัยหลายคนได้ทำการเพิ่มเทคนิคโดยการแบ่งกลุ่มอนุภาคออกเป็นกลุ่มย่อยที่จำนวนเท่ากัน โดยที่อนุภาคของแต่ละกลุ่มเคลื่อนที่เข้าหา GBEST ของตัวเอง โดยเรียกรวีนี้อาวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม (MPSO) ดังรูปที่ 3.2



รูปที่ 3.1 วิธีการหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค(PSO)



รูปที่ 3.2 วิธีการหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม (MPSO)

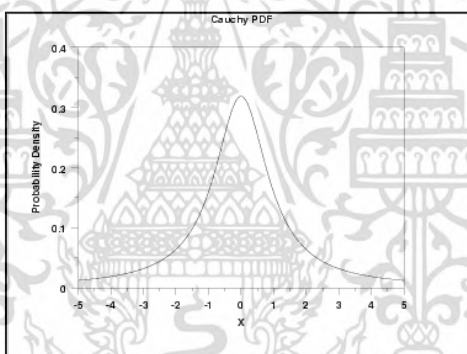
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการแบ่งกลุ่มอนุภาคอนุภาคออกเป็นหลายกลุ่มนั้นและค้นหาในบริเวณที่ต่างกันนั้น เพื่อให้อนุภาคค้นหาในบริเวณที่กว้างขึ้นหรือค้นหาแบบ Local Search ได้ดีขึ้นแทนที่จะลู่ออกหาตำแหน่งของ GBEST ตัวเดียวกันและติดในจุดสัมพัทธ์ที่เดียวกัน

### 3.1.2) งานวิจัยที่นำวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม

#### 1) วิธีการค้นหาอย่างรวดเร็วแบบหลากหลายกลุ่มโดยการกลายพันธุ์และการผสมพันธุ์ (Fast Multi-swarm Optimization with Cauchy Mutation and Crossover operation หรือ FMPSO)

การแจกแจงโคชี หรือ กระจายตัวโคชี (Cauchy Distribution) คือ การกระจายตัวโดยมีพารามิเตอร์บ่งชี้ตำแหน่ง และพารามิเตอร์บ่งชี้ขนาด  $y$  โดยมีฐานจะมีค่าเท่ากับ ดังแสดงในรูปที่ 3.3 และสมการ 3.2 การแจกแจงโคชีมีรูปแบบเหมือนกับการแจกแจงปกติ ต่างตรงที่ขอบทั้งสองด้านมีโอกาสกระจายตัวมากกว่าการแจกแจงปกติ ดังนั้น ค่าที่ได้จากการกระจายตัวแบบโคชีจะมีความกว้างกว่าค่าที่ได้กระจายตัวแบบปกติ



รูปที่ 3.3 การกระจายตัวแบบ Cauchy

$$\delta = \frac{1}{\pi} \frac{t}{t^2 + x^2} \quad (3.1)$$

โดยกำหนดให้

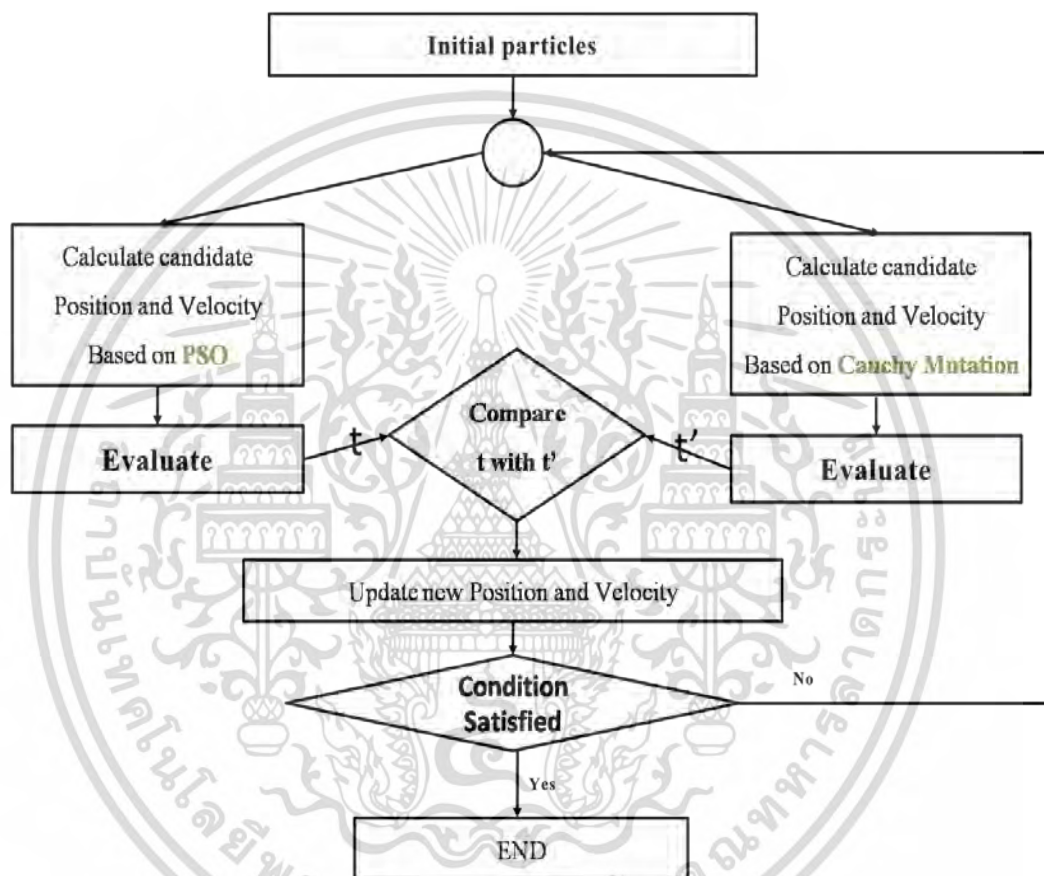
- t คือ ค่ามาตรฐานการกระจายตัวแบบโคชี
- x คือ ตัวเลขที่ทำการสุ่มอยู่ในช่วงขอบเขตที่กำหนด

จากวิธี FMPSO นั้นถูกพัฒนามาจากวิธีการค้นหาค่าเหมาะสมแบบอนุภาคกลุ่มอย่างรวดเร็ว (Fast Swarm Optimization หรือ FPSO) [9] อธิบายการทำงานได้จากผังงานที่ 3.4 เริ่มจากการกำหนดค่าเริ่มต้นให้กับอนุภาคทั้งหมดเช่นเดียวกับ PSO หลังจากนั้นให้คำนวณหาค่าความเหมาะสม (Fitness) ที่ได้จากวิธีการกลายพันธุ์ของเคชีและวิธีของ PSO ธรรมดา เพื่อนำมาใช้เปรียบเทียบหาค่าความเหมาะสมที่ดีที่สุดในแต่ละรอบของการทำงาน โดยให้นำค่าความเหมาะสมที่ดีที่สุดที่ได้มาใช้

เลือกปรับปรุงตำแหน่ง (Position) และความเร็ว (Velocity) ของอนุภาคโดยวิธีนั้นในรอบถัดไป

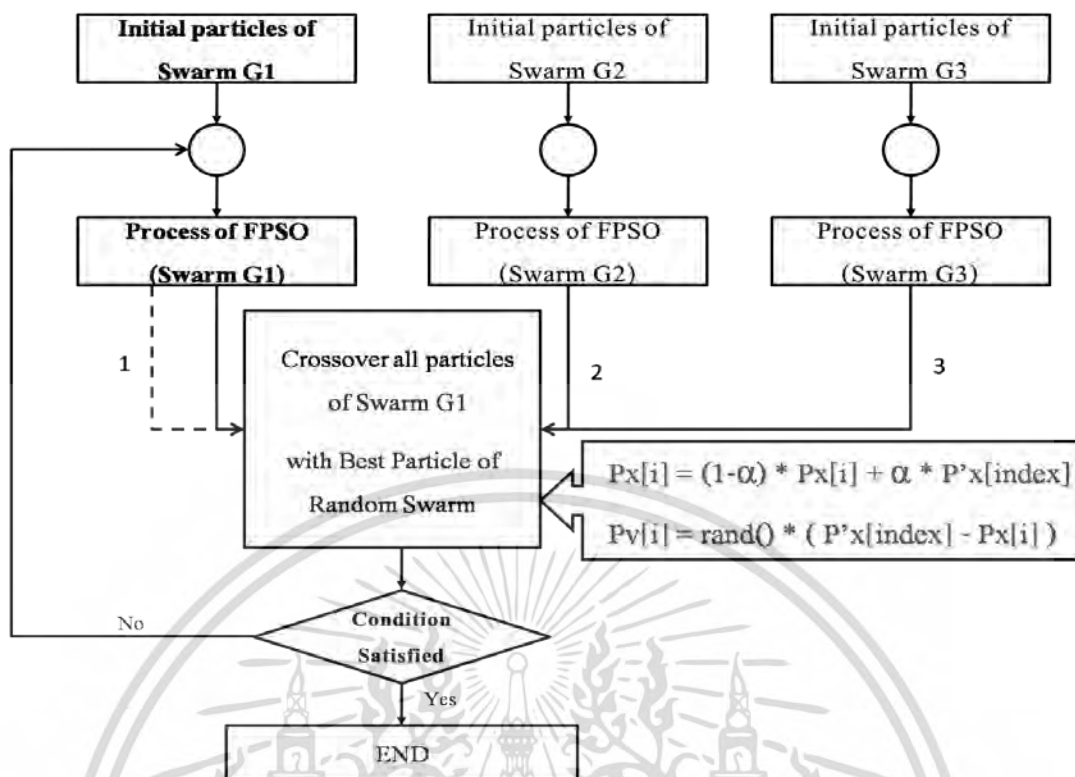
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักจากนั้นนักวิจัยกลุ่มนี้ได้ทำการแบ่งกลุ่มอนุภาคของ PSO ออกเป็นกลุ่มย่อย โดยที่อนุภาคของแต่ละกลุ่มจะทำวิธีตาม FPSO คือมีการเปรียบเทียบหาค่าความเหมาะสมที่ดีที่สุดจากวิธีการกลายพันธุ์ของเคซีและวิธีของ PSO ธรรมชาติ (Natural Selection) ตามผังงานที่ 3.4 เพื่อเพิ่มความหลากหลายให้กับการค้นหาและการติดต่อกันระหว่างกลุ่มอนุภาคจึงได้เพิ่มวิธีการผสมพันธุ์ (Crossover operation) เพื่อลดปัญหาการติดในจุดสุดสัมพัทธ์โดยทำการแลกเปลี่ยนและใช้ข้อมูลรวมกันในกลุ่ม โดยวิธีการผสมพันธุ์มาเป็นกลไกการทำงานของ FMPSO ทำให้อนุภาคมีอัตราการลู่เข้าหา GBEST ได้เร็วขึ้น



รูปที่ 3.4 แผนภาพการทำงานของวิธี FPSO

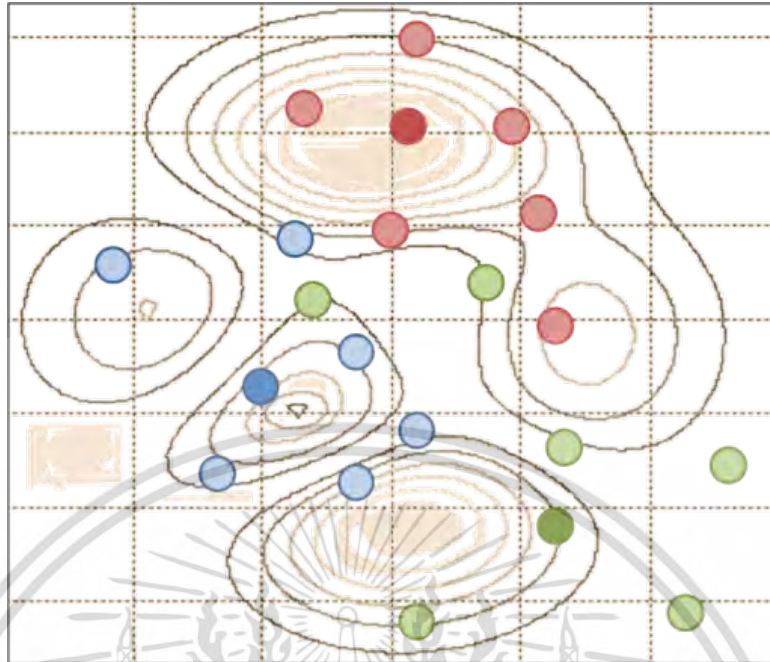
จากผังการทำงานของ FPSO จะเห็นได้ว่าอนุภาคทุกตัวจะถูกกำหนดค่าเริ่มต้นเหมือนกับ PSO ธรรมชาติ หลังจากนั้นจะทำการปรับปรุงตำแหน่ง (Position) และความเร็ว (Velocity) ของอนุภาคในรอบถัดไปโดยวิธีการกลายพันธุ์ของเคซีหรือวิธีของ PSO ธรรมชาติ ที่มีค่าความเหมาะสมที่ดีที่สุดที่ได้จากการคำนวณ



รูปที่ 3.5 แผนภาพการทำงานของวิธีFMPSO

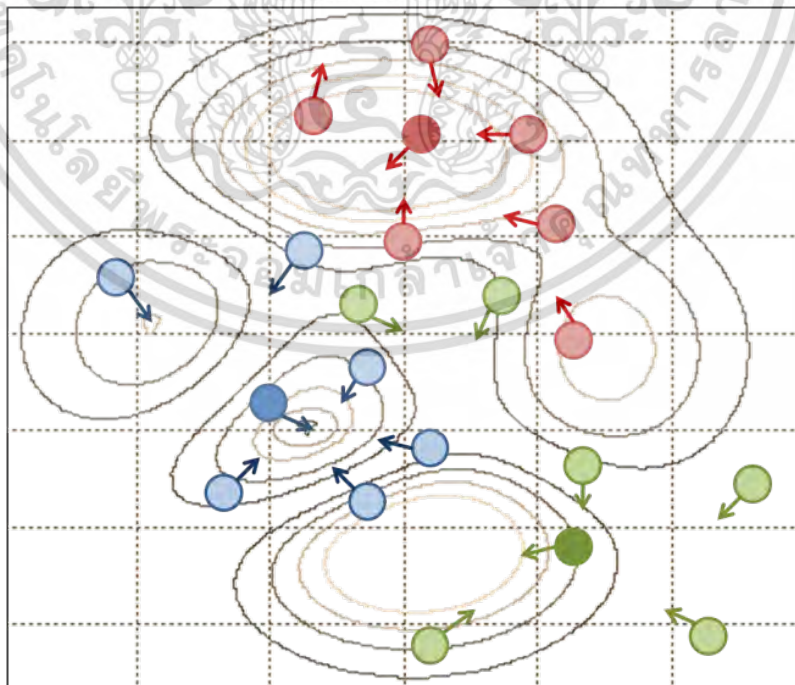
จากรูปผังงานที่ 3.4 อธิบายการทำงานของ FMPSO โดยเริ่มจากการแบ่งอนุภาค (G) ออกเป็นกลุ่มย่อยเท่ากันโดยที่แต่ละกลุ่มได้กำหนดค่าเริ่มต้นทั้งหมดเหมือนกับ PSO ธรรมดา ก่อนที่แต่ละกลุ่มจะเข้าสู่ขั้นตอนของ FPSO หลังจากนั้นจะเข้าสู่วิธีการผสมพันธุ์ (Crossover Operation) โดยทำการสุ่มเลือกอนุภาคที่ดีที่สุดของกลุ่มใดกลุ่มหนึ่งที่ได้จากการสุ่มมาทำการเปลี่ยนข้อมูลกับอนุภาคแต่ละตัวภายในกลุ่ม เพื่อให้อนุภาคแต่ละกลุ่มที่ต่างกันได้ใช้ข้อมูลรวมกัน มาเป็นกลไกการทำงานของ FMPSO ทำให้อัตราการลู่เข้าหาจุดสุดสัมบูรณ์ (Global Optimum) นั้นเร็วขึ้นคำตอบที่ได้ดีขึ้นเมื่อเทียบกับ PSO โดยขั้นตอนการทำงานของ FMPSO มีวิธีการทำงานดังนี้

ขั้นตอนที่1: เริ่มต้นโดยการแบ่งกลุ่มของ PSO ออกเป็นกลุ่มย่อย ตามจำนวนที่กำหนดไว้



รูปที่ 3.6 วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม

ขั้นตอนวิธี 2 คำนวณค่าความเหมาะสม (Fitness) ให้กับแต่ละอนุภาคแต่ละตัวโดยใช้วิธีการค้นหาค่าตอบด้วย PSO ปกติ เมื่อสิ้นสุดการค้นหาค่าตอบด้วย PSO ปกติในแต่ละรอบโดยเมื่อสิ้นสุดขั้นตอนกระบวนการปรับปรุงตำแหน่งให้ทำตามขั้นตอนถัดไปดังรูปที่ 3.7



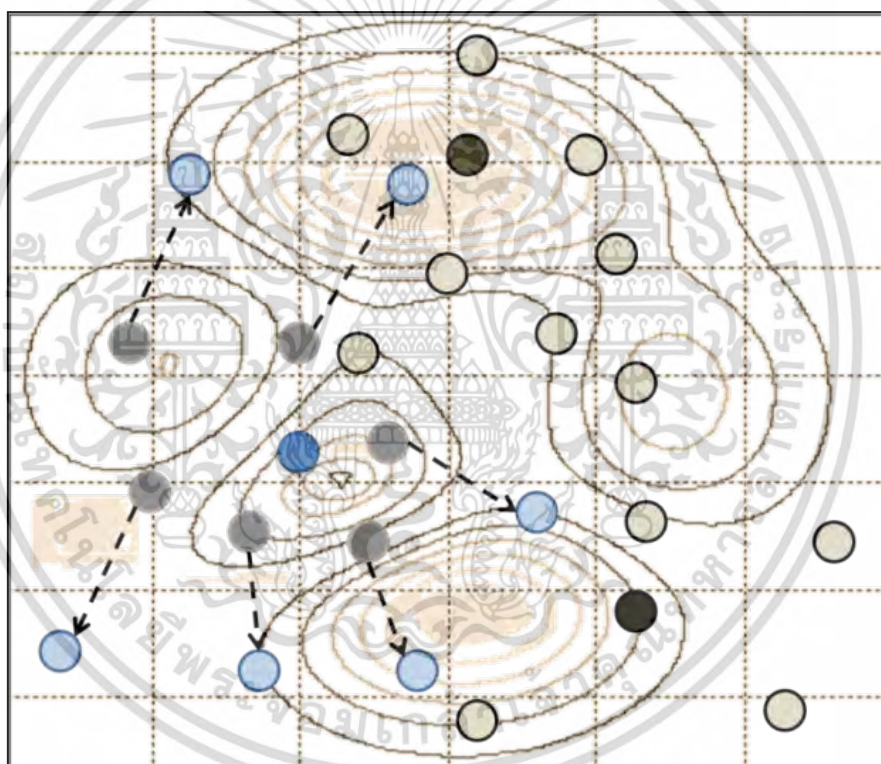
รูปที่ 3.7 การเคลื่อนที่ของอนุภาคแต่ละกลุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 3 ปรับปรุงตำแหน่ง (Position) และความเร็ว (Velocity) ของอนุภาคโดยเลือกจากการคำนวณค่าความเหมาะสม (Fitness) วิธีการกลายพันธุ์แบบเคซี (Cauchy Mutation) และวิธีของ PSO ธรรมดา โดยการกลายพันธุ์แบบเคซีสามารถทำให้อนุภาคกระโดดได้ไกลทำให้ PSO มีโอกาสเพิ่มขึ้นที่จะหนีออกจากจุดสุดสัมพัทธ์โดยอนุภาคจะถูกปรับปรุงความเร็วและตำแหน่งของอนุภาคด้วยสมการกลายพันธุ์ตามสมการที่ (3.2) และ (3.3) โดย  $\delta$  และ  $\delta_{id}$  คือ จำนวนการสุ่มแบบเคซีจากรูป 3.8 มีการกลายพันธุ์เกิดขึ้นทำให้อนุภาคกระจายตัวออก โดยอนุภาคสีน้ำเงินแทนตำแหน่งใหม่ของอนุภาคที่เกิดการกลายพันธุ์จากอนุภาคสีเทา

$$V_{id}' = V_{id} \exp(\delta) \quad (3.2)$$

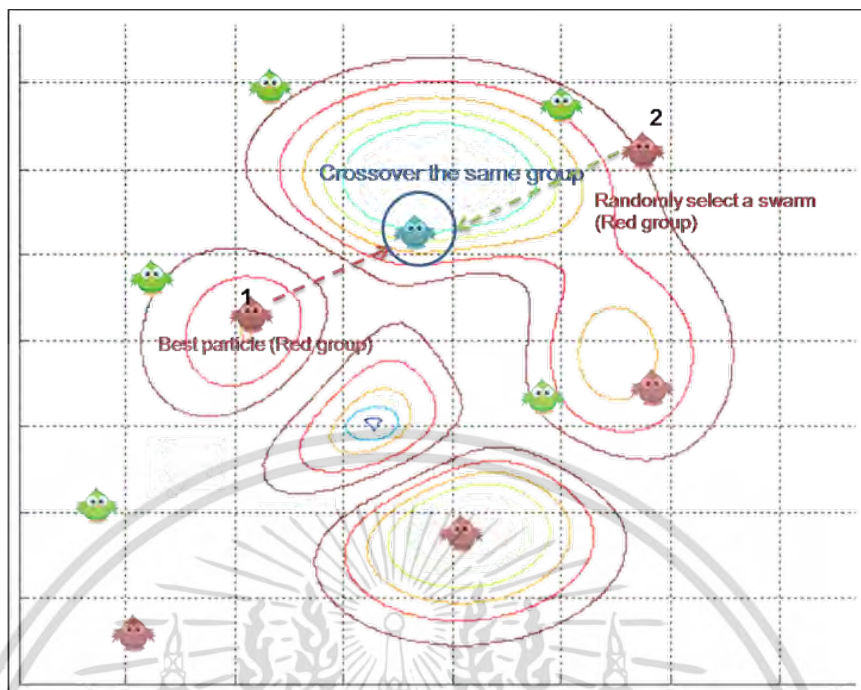
$$X_{id}' = X_{id} + V_{id}' \delta_{id} \quad (3.3)$$



รูปที่ 3.8 อนุภาคที่ถูกปรับปรุงโดยใช้วิธีการกลายพันธุ์แบบเคซี

ขั้นตอนที่ 4 ดำเนินกระบวนการเลือกตามธรรมชาติวิธีนี้คือจะนำผลจากการกลายพันธุ์ของอนุภาคแต่ละอนุภาคนั้นจะถูกนำมาเปรียบเทียบกับอนุภาคที่ถูกกลายพันธุ์ ถ้าผลจากการกลายพันธุ์ดีกว่าก็จะแทนที่ด้วยผลจากการกลายพันธุ์นั้น ดังรูป 3.9 ผลจากการกลายพันธุ์อนุภาคตัวที่ 1 ดีกว่าเดิม ให้เลือกอนุภาคตัวที่หนึ่ง แต่อนุภาคตัวที่ 2 ผลจากการกลายพันธุ์แย่กว่าตำแหน่งเดิม จึงไม่เลือกผลจากการกลายพันธุ์ ผลจากการคัดเลือกอนุภาคที่ดีนี้สามารถช่วยลดโอกาสในการติดในจุดสุดสัมพัทธ์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 วิธีการผสมพันธุ์ระหว่างกลุ่มอนุภาคเดียวกัน

ขั้นตอนที่ 5 ดำเนินกระบวนการกระบวนการผสมพันธุ์ (Crossover Operation) โดยวิธีนี้จะทำการสุ่มอนุภาคแต่ละกลุ่ม สามารถแลกเปลี่ยนข้อมูลกันเพื่อให้กลุ่มของอนุภาคมีการกระจายตัว ค้นหาคำตอบและเพิ่มความเร็วของการลู่เข้าหากัน (Convergence Speed) โดยมีขั้นตอนต่างๆ ดังนี้

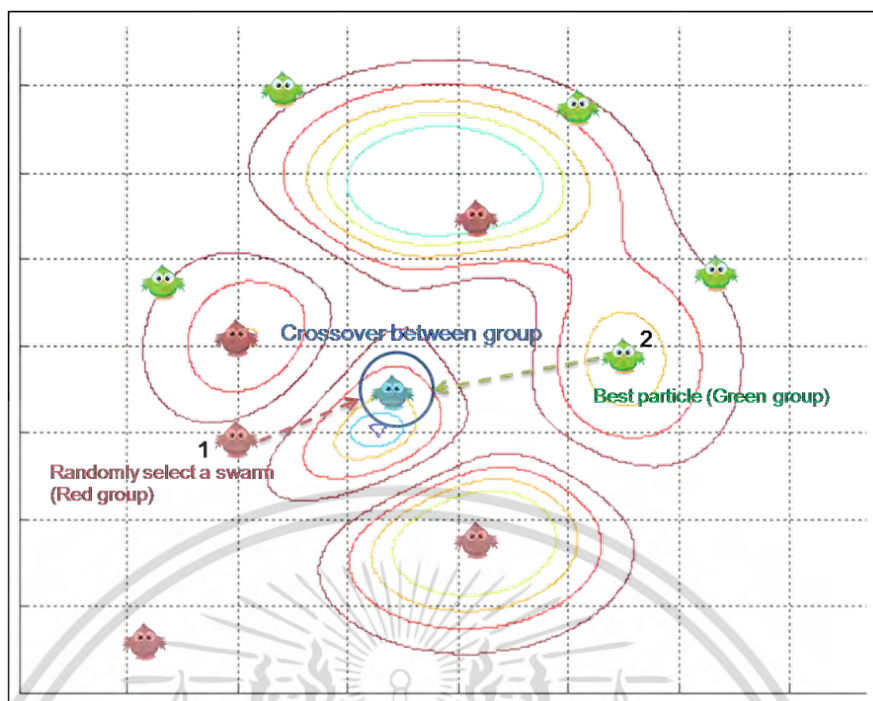
ขั้นตอนที่ 5.1 ทำการสุ่มกลุ่มของอนุภาคมาหนึ่งกลุ่ม ( $k$ ) จากกลุ่มอนุภาคทั้งหมด และให้ทำการสุ่มกลุ่มอนุภาคมาอีกหนึ่งกลุ่มแล้วเลือกอนุภาคตัวที่มีค่าดีที่สุดจากกลุ่มอนุภาคที่สุ่มมานั้น คือ  $p'$

ขั้นตอนที่ 5.2 สำหรับแต่ละมิติ  $i$  ของอนุภาค  $p'$  และกำหนดให้ตำแหน่ง คือ  $p_x[i]$  ของอนุภาค  $p$  และความเร็ว คือ  $p_v[i]$  ของอนุภาค  $p$  หลังจากนั้นทำการสุ่มค่าขึ้นมาหนึ่งค่า ถ้าค่าสุ่มขึ้นมา มีค่าน้อยกว่าอัตราการผสมพันธุ์ (Crossover Rate,  $q_c$ ) ให้ทำการกระบวนการผสมพันธุ์ระหว่าง  $p$  กับ  $p'$  ดังสมการข้างล่างนี้ โดย  $\alpha$  คือ ตัวเลขสุ่มมีค่าระหว่าง 0 กับ 1

$$p_x[i] = (1 - \alpha) \times (p_x[i]) + \alpha \times p_x'[i] \quad (3.4)$$

$$p_v[i] = \text{rand}() \times (p_x'[i] - p_x[i]) \quad (3.5)$$

ขั้นตอนที่ 5.3 คือถ้าอนุภาคแต่ละอนุภาคของกลุ่ม ( $k$ ) ได้ปรับปรุงหมดแล้ว ให้หยุดการทำงาน ถ้ายังไม่หมดให้กลับไปขั้นที่ 1



รูปที่ 3.10 กระบวนการผสมพันธุ์ระหว่างกลุ่มอนุภาคต่างกัน

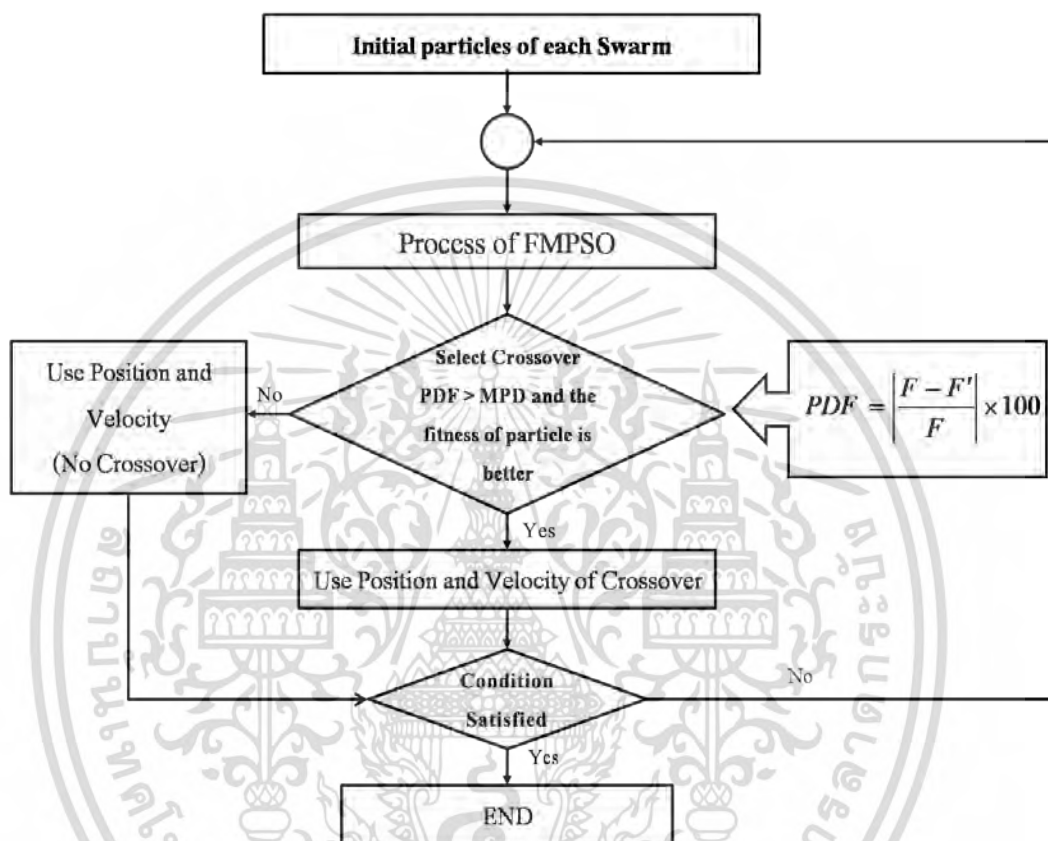
จากรูป 3.10 อธิบายถึงการดำเนินกระบวนการกระบวนการผสมของอนุภาคคนละกลุ่ม สมมุติให้มีการสุ่มเลือกได้นกกลุ่มที่ 1 สีแดงและสุ่มเลือกนกที่ดีที่สุดกลุ่มที่ 2 สีเขียว แล้วดำเนินกระบวนการผสมพันธุ์กันกับนกตัวที่ถูกเลือก ตามสมการ (3.4) และ (3.5) ผลลัพธ์ที่ได้คืออนุภาคในตำแหน่งสีฟ้า หลังจากการกระบวนการผสมพันธุ์ก็จะวนกลับไปทำการกระบวนการประเมินค่าตำแหน่งของ PSO สำหรับแต่ละอนุภาคในรอบถัดไป

## 2) วิธีการคัดเลือกการผสมพันธุ์โดยใช้ความเหมาะสมกับแบบหลากหลายกลุ่ม (Selective Crossover base on Fitness in Multi-Swarm Optimization, SCMPSO)

จากวิธี FMPSO ที่กระบวนการผสมพันธุ์และกระบวนการกลายพันธุ์ ถูกใช้เพื่อป้องกันปัญหาการติดในจุดสุดสัมพัทธ์ของ PSO และเพิ่มความเร็วของการลู่เข้าหากันสำหรับวิธีการคัดเลือกการผสมพันธุ์โดยใช้ความเหมาะสมกับแบบหลากหลายกลุ่ม (Selective Crossover base on Fitness in Multi-Swarm Optimization หรือ SCMPSO) [10] นี้ได้รับการปรับปรุงมาจาก FMPSO โดยที่ผลจากกระบวนการผสมพันธุ์ของ FMPSO ในการแก้ปัญหาบางปัญหาอาจจะก่อให้เกิดการลู่เข้าหากันเร็วเกินไปทำให้ผลลัพธ์ที่ได้จากการค้นหาด้วยกระบวนการผสมพันธุ์ ทำให้ได้ผลลัพธ์ที่แย่กว่าการค้นหาคำตอบด้วยวิธี MPSO ดังนั้น เพื่อแก้ปัญหานี้ SCMPSO ได้ปรับปรุง FMPSO โดยเพิ่มวิธีการเลือกตามธรรมชาติ ตามขั้นตอนที่ 7 หลังจากกระบวนการผสมพันธุ์โดยค่าความเหมาะสมของอนุภาคที่ได้จากกระบวนการผสมพันธุ์เปรียบเทียบกับค่าความเหมาะสมของอนุภาคที่ถูกดำเนินการผสมพันธุ์

ถ้าค่าความเหมาะสมของอนุภาคที่ได้จากกระบวนการผสมพันธุ์ดีกว่าค่าความเหมาะสมของอนุภาคที่  
เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำไปใช้

ถูกดำเนินการผสมพันธุ์ให้นำอนุภาคที่ได้จากกระบวนการผสมพันธุ์แทนที่อนุภาคที่ถูกดำเนินการผสมพันธุ์ โดยจากผลการทดลองแสดงให้เห็นว่า SCMPSO ได้ผลลัพธ์การค้นหาคำตอบที่ดีกว่า FMPSO สำหรับขั้นตอนการทำงานของ SCMPSO มีรายละเอียดดังนี้ และแผนภาพการทำงานของ SCMPSO ดังรูปที่ 3.10

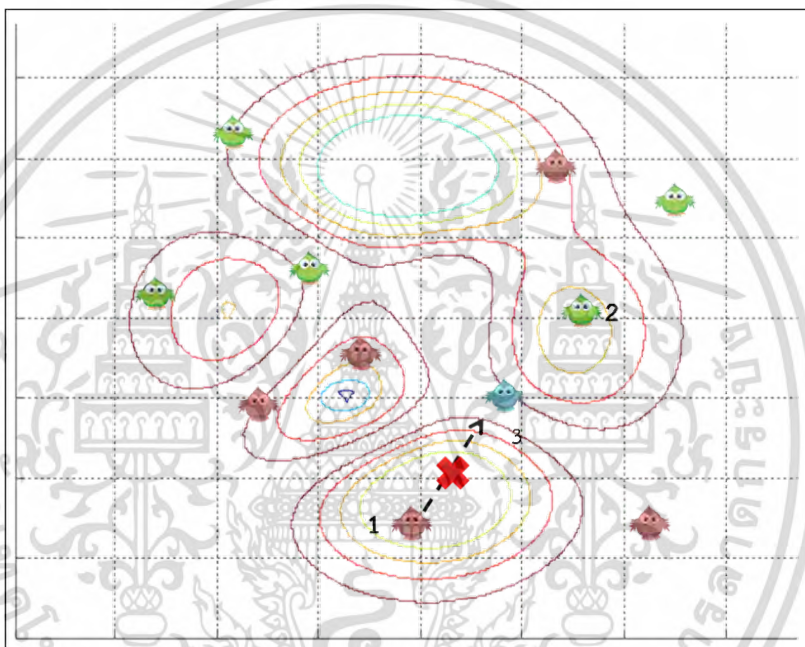


รูปที่ 3.11 แผนภาพการทำงานของวิธี SCMPSO

จากรูปผังงานที่ 3.11 อธิบายการทำงานของ SCMPSO โดยเริ่มจากกำหนดค่าเริ่มต้นให้กับอนุภาคของแต่ละกลุ่มอนุภาคภายในพื้นที่ที่ต้องการค้นหาคำตอบโดยแต่ละกลุ่มอนุภาคจะดำเนินการตามวิธีของ FMPSO ก่อนเพื่อคัดเลือกคำตอบระหว่างวิธีการกลายพันธุ์ของเคซีและวิธีของ PSO หลังจากนั้นก่อนที่จะทำวิธีการผสมพันธุ์นั้นให้ทำการเลือกโดยพิจารณาตำแหน่งใหม่จากเปอร์เซ็นต์ของความเหมาะสมระหว่างตำแหน่งใหม่และตำแหน่งเดิม (PDF)

โดยที่ PDF คือ ค่าเปอร์เซ็นต์ความแตกต่างของค่าความเหมาะสมของแต่ละอนุภาค, MPD คือ ค่าสูงสุดเปอร์เซ็นต์ความแตกต่างของค่าความเหมาะสมถ้าค่าเปอร์เซ็นต์ความแตกต่างของค่าความเหมาะสมของตำแหน่งใหม่มากกว่า MPD ตำแหน่งใหม่จะถูกเลือก,  $F'$  คือ ค่าความเหมาะสมของอนุภาคที่ถูกปรับปรุง,  $F$  คือ ค่าความเหมาะสมของอนุภาคก่อนที่จะถูกปรับปรุง

ขั้นตอนที่ 6 วิธีการค้นหาคำตอบของ SCMPSO เหมือนกับ FMPSO ทั้งหมด แต่หลังจากดำเนินการกระบวนการผสมพันธุ์ แทนที่จะวนกลับไปทำกระบวนการประเมินค่าตำแหน่งของ PSO สำหรับแต่ละอนุภาคในรอบถัดไป ให้ดำเนินการเพิ่มวิธีการเลือกตามธรรมชาติ หลังจากการกระบวนการนี้ แล้วเมื่อผ่านวิธีการเลือกตามธรรมชาติแล้วจึงจะวนกลับไปทำกระบวนการประเมินค่าตำแหน่งของ PSO สำหรับแต่ละอนุภาคในรอบถัดไปจากรูป 3.12 จะเห็นว่าภาพข้างล่างนี้ สมมุติให้มีการสุ่มเลือก โดยได้นกกลุ่มที่ 1 สีน้ำเงิน และเลือกนกที่ดีที่สุด แล้วดำเนินการกระบวนการผสมพันธุ์ กับนกกลุ่มอื่นๆ ตามสมการ (3.1) และ (3.2) ผลลัพธ์ที่ได้คืออนุภาคอยู่ในตำแหน่งสีเขียว ซึ่งสีเขียวอยู่ในตำแหน่งที่แย่กว่าเดิมทำให้ ไม่เลือกสีเขียวมาแทนที่อนุภาคเดิม



รูปที่ 3.12 วิธีการเลือกตามธรรมชาติประยุกต์ใช้กับวิธีการคัดเลือกการผสมพันธุ์

จากวิธีของ FMPSO และ SCMPSO จะเห็นได้ว่าในแต่ละรอบของการค้นหาคำตอบจะต้องทำกระบวนการประเมินค่าตำแหน่งของแต่ละอนุภาค (Evaluate) หลายครั้งทำให้เกิดการเดาคำตอบที่ไม่จำเป็นมากเกินไป โดย FMPSO ในแต่ละรอบของการค้นหาคำตอบ FMPSO ต้องเดาคำตอบ 2 ครั้ง ครั้งแรกเดาจาก การค้นหาด้วย PSO ปกติ ครั้งสองเดาจากการกลายพันธุ์ แล้วนำสองครั้งนี้มาเปรียบเทียบ ดังนั้นการเดาคำตอบในแต่ละรอบจะมากกว่า PSO ปกติ จึงมีโอกาสได้ผลลัพธ์ที่ดีกว่า PSO ปกติ โดยจากการทดลองหัวข้อ [9,10] ได้แสดงให้เห็นว่า FMPSO มีจำนวนเดาคำตอบเท่ากับ PSO ปกติ แล้วได้ผลลัพธ์ที่แย่กว่า PSO ในบางสมการ ซึ่งการพิสูจน์ว่า FMPSO ประสิทธิภาพในการค้นหาคำตอบแย่กว่า PSO เมื่อจำนวนการเดาคำตอบเท่ากัน

สำหรับ SCMPSO ในแต่ละรอบของการค้นหาคำตอบ SCMPSO ต้องเดาคำตอบ (Evaluation Call) 3 ครั้ง โดยที่ครั้งแรกเดาจากการค้นหาด้วย PSO ปกติ ครั้งสองเดาจากการกลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

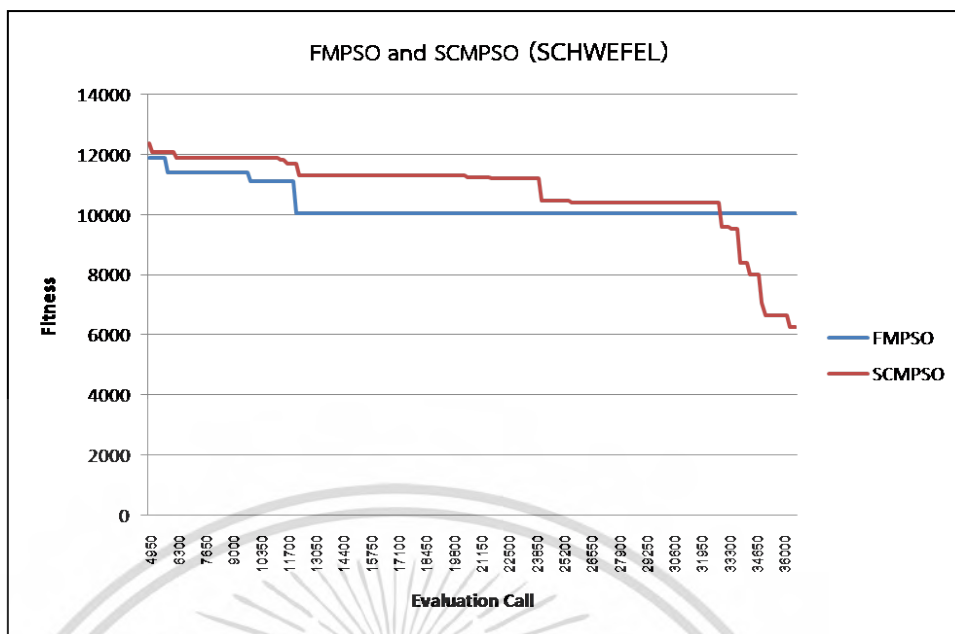
พันธุ์ครั้งที่สามเตาจากการผสมพันธุ์ แล้วนำสามครั้งนี้มาเปรียบเทียบคัดเลือกคำตอบที่ดีที่สุด ดังนั้น การเตาคำตอบในแต่ละรอบจะมากกว่า PSO ปกติ จึงมีโอกาสได้ผลลัพธ์ที่ดีกว่า PSO ปกติ โดยจากการทดลองหัวข้อ [14] ได้แสดงให้เห็นว่า SCMPSO มีจำนวนเตาคำตอบเท่ากับ PSO ปกติ แล้วได้ผลลัพธ์ที่แยกว่า PSO ในบางสมการ ซึ่งการพิสูจน์ว่า SCMPSO ประสิทธิภาพในการค้นหาคำตอบแยกว่า PSO เมื่อจำนวนการเตาคำตอบเท่ากัน

ขั้นตอนวิธีการค้นหาอย่างรวดเร็วแบบหลากหลายกลุ่มโดยการกลายพันธุ์และการผสมพันธุ์ (FMPSO) [5] นี้มีแนวคิด มาจากขั้นตอนวิธีการค้นหาค่าเหมาะสมแบบอนุภาคกลุ่มอย่างรวดเร็ว (Fast Swarm Optimization หรือ FPSO) [9] โดยได้นำเสนอการปรับปรุงอนุภาคกับวิธีการกลายพันธุ์ของเคซี ที่ถูกนำมาใช้บ่อยครั้งในขั้นตอนวิธีของ PSO และขั้นตอนวิธีการคัดเลือกการผสมพันธุ์ โดยใช้ความเหมาะสมกับแบบหลากหลายกลุ่ม (SCMPSO) [10] ที่ได้เพิ่มขั้นตอนการคัดเลือกคำตอบภายใน FMPSO โดยการคัดเลือกคำตอบระหว่างขั้นตอนการผสมพันธุ์ (Crossover Operation) กับขั้นตอนของ PSO ธรรมดา ซึ่งเป็นการนำเอากระบวนการผสมพันธุ์มาประยุกต์ใช้แลกเปลี่ยนข้อมูลระหว่างกลุ่มในทุกรอบของการค้นหา วิธีเหล่านี้ทำให้ได้ผลลัพธ์การค้นหาคำตอบที่ดีขึ้น แต่กลับเพิ่มจำนวนการเตาคำตอบในแต่ละรอบของการทำงานแทน

แต่จากทำงานของ FMPSO นั้นกับพบว่าอนุภาคของแต่ละกลุ่มต้องทำกระบวนการผสมพันธุ์ทุกครั้งของการทำงาน ทำให้เกิดแนวคิด คือ ขั้นตอนการเลือกทำกระบวนการผสมพันธุ์ (Select Crossover) เพื่อลดขั้นตอนกระบวนการผสมพันธุ์

ดังนั้น วิธีของ FMPSO และ SCMPSO จะเห็นได้ว่าในแต่ละรอบของการค้นหาคำตอบ จะต้องทำกระบวนการประเมินค่าตำแหน่งของแต่ละอนุภาค (Evaluate) หลายครั้งทำให้เกิดการเตาคำตอบที่ไม่จำเป็นมากเกินไป โดย FMPSO ในแต่ละรอบของการค้นหาคำตอบ FMPSO ต้องเตาคำตอบ 2 ครั้ง ครั้งแรกเตาจากการค้นหาด้วย PSO ปกติ ครั้งสองเตาจากการกลายพันธุ์ แล้วนำสองครั้งนี้มาเปรียบเทียบ ดังนั้นการเตาคำตอบในแต่ละรอบจะมากกว่า PSO ปกติ จึงมีโอกาสได้ผลลัพธ์ที่ดีกว่า PSO ปกติ โดยจากการทดลองหัวข้อ [14] ได้แสดงให้เห็นว่า FMPSO มีจำนวนเตาคำตอบเท่ากับ PSO ปกติ แล้วได้ผลลัพธ์ที่แยกว่า PSO ในบางสมการ ซึ่งการพิสูจน์ว่า FMPSO ประสิทธิภาพในการค้นหาคำตอบแยกว่า PSO เมื่อจำนวนเตาคำตอบเท่ากัน

สำหรับ SCMPSO ในแต่ละรอบของการค้นหาคำตอบ SCMPSO ต้องเตาคำตอบ 3 ครั้ง ครั้งแรกเตาจากการค้นหาด้วย PSO ปกติ ครั้งสองเตาจากการกลายพันธุ์ครั้งที่สามเตาจากการผสมพันธุ์ แล้วนำสามครั้งนี้มาเปรียบเทียบคัดเลือกคำตอบที่ดีที่สุด ดังนั้นการเตาคำตอบในแต่ละรอบจะมากกว่า PSO ปกติ จึงมีโอกาสได้ผลลัพธ์ที่ดีกว่า PSO ปกติ โดยจากการทดลองหัวข้อ [14] ได้แสดงให้เห็นว่า SCMPSO มีจำนวนเตาคำตอบเท่ากับ PSO ปกติ แล้วได้ผลลัพธ์ที่แยกว่า PSO ในบางสมการ ซึ่งการพิสูจน์ว่า SCMPSO ประสิทธิภาพในการค้นหาคำตอบแยกว่า PSO เมื่อจำนวนเตาคำตอบเท่ากัน



รูปที่ 3.13 กราฟเปรียบเทียบวิธี FMPSO และ SCMPSO

จากรูปที่ 3.13 จะเห็นได้ว่าเมื่อนุภาคของทั้งสองวิธีเคลื่อนที่ในระยะเวลาหนึ่งในขณะที่จำนวนการเดาคำตอบ (Evaluation Call) เท่ากันกับพบว่า SCMPSO ได้คำตอบที่ดีกว่า FMPSO ในรอบการเดาคำตอบที่เท่ากัน

### 3.2) วิธีหาค่าเหมาะสมที่สุดกับวิธีการกระจายตัวของอนุภาค

#### 3.2.1) การประยุกต์วิธีการเริ่มต้นการค้นหาใหม่กับวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค

นอกจากการนำวิธีการกลายพันธุ์ ที่นำมาใช้เพื่อแก้ปัญหาการติดในจุดสุดสัมพัทธ์ของ PSO นักวิจัยจำนวนหนึ่งได้นำเสนอวิธีการเริ่มต้นการค้นหาใหม่เพื่อใช้ในการแก้ปัญหานี้ ซึ่งวิธีนี้สามารถนำมาประยุกต์ใช้ได้ง่ายเช่นเดียวกับการกลายพันธุ์แต่ให้ผลในการแก้ปัญหาการติดในจุดสุดสัมพัทธ์ที่แตกต่างกัน โดยจากงานวิจัย [14] แสดงให้เห็นว่าวิธีนี้สามารถเพิ่มประสิทธิภาพของ PSO โดยวิธีการเริ่มต้นการค้นหาใหม่สามารถแบ่งได้ 2 ประเภท คือ

1) การเริ่มต้นการค้นหาใหม่โดยไม่พิจารณาตำแหน่งก่อนหน้า (RERUN) [10] คือการกระจายอนุภาคโดยอนุภาคนั้นถูกกำหนดค่าเริ่มต้นใหม่ให้ ซึ่งวิธีนี้ตำแหน่งการค้นหาที่ก่อนขบวนการเริ่มต้นจะไม่ถูกนำมาใช้

2) การเริ่มต้นการค้นหาในบริเวณใหม่โดยพิจารณาตำแหน่งก่อนหน้า (REPOSITION) [10, 43] คือการกระจายอนุภาค โดยตำแหน่งของอนุภาคที่ถูกดำเนิน ตำแหน่งนั้นจะดำเนินการกลายพันธุ์

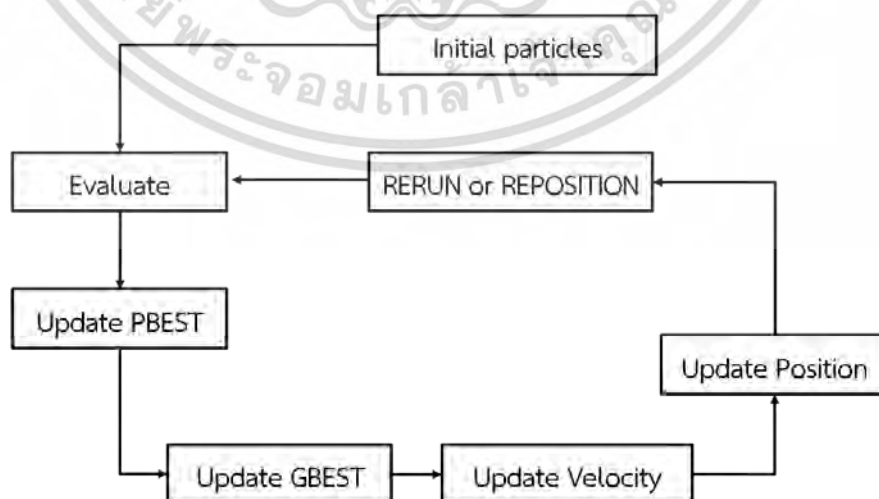
ในจำนวนมาก เช่น 30%, 50%, 70% ของจำนวนมิติทั้งหมดของทุกอนุภาค เป็นต้น โดยจะเรียกค่านี้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการศึกษาไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ว่าความน่าจะเป็นในการรีโพซิชัน (Reposition Probability) ซึ่งค่านี้จะถูกกำหนดให้และเป็นค่าคงที่ เพื่อให้อนุภาคกระจายตัวออกเป็นบริเวณกว้างมากพอที่ให้อนุภาคค้นหาจากบริเวณจุดสุดสัมพัทธ์ที่ติดอยู่ซึ่งวิธีนี้ตำแหน่งการค้นหาก่อนขบวนการการเริ่มต้น จะมีอิทธิพลต่อตำแหน่งหลังจากการดำเนินการ

สรุปได้ว่า การกระจายอนุภาค (Particles Distribution Operation) คือ การกระจายอนุภาคโดยการกลายพันธุ์ในจำนวนมากๆ เช่น 20%, 30%, 50% ของจำนวนมิติทั้งหมดของทุกอนุภาคเป็นต้น โดยจะเรียกค่านี้ว่า ความน่าจะเป็นในการกระจายอนุภาค (Distribution Probability) ซึ่งค่าที่ถูกกำหนดให้และเป็นค่าคงที่ แต่ค่านี้จะไม่มากจนเทียบเท่ากับการสุ่มตำแหน่งอนุภาคใหม่ และค่าที่เปลี่ยนนั้นจะไม่แตกต่างจากค่าเดิมมากนัก ซึ่งต่างจากการเริ่มต้นใหม่ที่ค่านี้เกิดจากการสุ่มค่าขึ้นมาใหม่ทั้งหมด สำหรับค่าความเร็วของอนุภาค, PBEST และ GBEST จะถูกกำหนดให้เป็นค่าเริ่มต้นใหม่เพื่อให้อนุภาคไม่ย้อนกลับมายังตำแหน่งเดิมที่ดีกว่า

สำหรับหลักการการทำงานของวิธีการเริ่มต้นการค้นหาใหม่ทั้ง 2 ประเภท มีหลักการการทำงานที่เหมือนกันดังนี้ ถ้า GBEST ไม่มีค่าดีขึ้นเป็นระยะเวลาใน PSO สามารถสรุปได้ว่า PSO นั้นเกิดปัญหาการติดในจุดสุดสัมพัทธ์ขึ้นและเมื่อปัญหาการติดในจุดสุดสัมพัทธ์เกิดขึ้นแล้ว ให้ดำเนินการวิธีการเริ่มต้นการค้นหาใหม่ หลังจากนั้นก็จะดำเนินการค้นหาแบบ PSO จนพบว่าเกิดปัญหาการติดในจุดสุดสัมพัทธ์อีกครั้ง จึงเริ่มวิธีการเริ่มต้นการค้นหาใหม่อีกครั้งวนเช่นนี้ไปเรื่อยๆ จนกว่าจะครบรอบการค้นหาหรือพบคำตอบ วิธีการเริ่มต้นการค้นหาใหม่ทำให้อนุภาคมีโอกาสกระจายตัวไปค้นหาในบริเวณอื่น ซึ่งอาจจะพบคำตอบที่ดีกว่าเดิม นอกจากนี้การเริ่มต้นการค้นหาใหม่ที่ประยุกต์ใช้กับตำแหน่งของอนุภาคแล้ว ยังสามารถนำมาประยุกต์ส่วนประกอบของ PSO อื่นๆ เช่น PBEST GBEST ความเร็วของอนุภาค ซึ่งส่งผลกระทบต่อการค้นหาที่แตกต่างกัน

### 3.2.2) ขั้นตอนของการเริ่มต้นการค้นหาใหม่



รูปที่ 3.14 แผนภาพการทำงานของรีรันหรือรีโพซิชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยามให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประยุกต์ RERUN หรือ REPOSITION กับ PSO มีขั้นตอนการทำงานดังรูป 3.12 โดยขั้นตอนแรกของการทำงานจะทำงานเหมือนกับ PSO ปกติ คือเริ่มจากการสร้างอนุภาคขึ้นมาโดยสุ่มตำแหน่งและความเร็วเริ่มต้นให้กับอนุภาคแต่ละตัว จากนั้นเป็นขั้นตอนหาค่าความเหมาะสมโดยนำอนุภาคทั้งหมดไปหาค่าความเหมาะสมจากตำแหน่งปัจจุบันของอนุภาค สำหรับค่าความเหมาะสมที่ได้นี้จะถูกนำไปใช้ในการปรับปรุง PBEST และ GBEST ซึ่งค่านี้ใช้ในการคำนวณค่าความเร็วและทิศทางใหม่ของแต่ละอนุภาคเพื่อใช้เคลื่อนที่อนุภาคเหล่านั้นไปยังตำแหน่งใหม่ ในขั้นตอนต่อมาจะเป็นขั้นตอนที่เพิ่มเข้าไป คือถ้าตรวจสอบเงื่อนไขการ RERUN หรือ REPOSITION แล้วตรงตามเงื่อนไขที่กำหนดก็จะดำเนินการ RERUN หรือ REPOSITION ขั้นตอนต่อมาถ้าพบคำตอบหรือครบจำนวนรอบที่กำหนดเอาไว้จบการค้นหา แต่ถ้าไม่พบหรือไม่ครบจำนวนรอบก็กลับไปทำกระบวนการประเมินค่าตำแหน่งของแต่ละอนุภาคในรอบถัดไป

### 3.2.3) องค์ประกอบของการเริ่มต้นการค้นหาใหม่

1) รอบของการเริ่มต้นการค้นหาใหม่ (RESET PERIOD) คือ เงื่อนไขที่ใช้กำหนดว่าเมื่อใดควรดำเนินการเริ่มต้นการค้นหาใหม่ โดยรอบนี้สามารถกำหนดได้หลายวิธี ตัวอย่างงานวิจัยเช่น

งานวิจัย [19, 43] เสนอ RESET PERIOD โดยพิจารณาจากรอบของการค้นหาที่มีรัศมีของกลุ่มอนุภาคมากที่สุด (Maximum Swarm Radius หรือ MSR) โดยสุรมีขั้นตอนดังนี้ การเริ่มต้นการค้นหาใหม่ เมื่อ MSR น้อยกว่าค่าที่กำหนดให้ (Threshold value)

งานวิจัย [13] เสนอ RESET PERIOD โดยพิจารณาจากจำนวนซ้ำของ GBEST ที่ไม่เปลี่ยนแปลงและรัศมีของกลุ่มอนุภาค โดยสุรมีขั้นตอนดังนี้ จะดำเนินการเริ่มต้นการค้นหาใหม่ เมื่อจำนวนซ้ำของ GBEST มีค่าที่ไม่เปลี่ยนแปลงมากกว่าจำนวนที่กำหนดให้ (MAXSTEP) และรัศมีของกลุ่มอนุภาค (SWARMRADIUS) มีค่าน้อยกว่า รัศมีของขอบที่ถูกกำหนดให้ (BORDERRADIUS)

งานวิจัย [48] เสนอ RESET PERIOD โดยพิจารณาจากจำนวนซ้ำของค่าความเหมาะสมของอนุภาคเพื่อนบ้านที่ดีที่สุด (Neighborhood Best Position หรือ NBEST) ที่ไม่เปลี่ยนแปลง โดยสุรมีขั้นตอนดังนี้ ดำเนินการเมื่อจำนวนซ้ำของ NBEST มีค่าไม่เปลี่ยนแปลงมีค่ามากกว่าจำนวนที่กำหนดให้ (MAXSTEP) โดย NBEST นั้นแต่ละอนุภาคจะมีค่าเป็นของตัวเอง นั้นหมายความว่าอนุภาคต่างๆ อาจจะถูกวิธีการเริ่มต้นการค้นหาใหม่ไม่พร้อมกัน โดยที่อนุภาคตัวใดตรงตามเงื่อนไขที่กำหนดนี้ก่อน อนุภาคตัวนั้นก็จะถูกดำเนินการวิธีการเริ่มต้นการค้นหาใหม่ก่อน

งานวิจัย [13] เสนอ RESET PERIOD โดยพิจารณาจากจำนวนซ้ำของค่าความเหมาะสมของอนุภาคเพื่อนบ้านที่ดีที่สุด (Neighborhood Best Position หรือ NBEST) ที่ไม่เปลี่ยนแปลง โดยสุรมีขั้นตอนดังนี้ ดำเนินการเมื่อจำนวนซ้ำของ NBEST มีค่าไม่เปลี่ยนแปลงมีค่ามากกว่าจำนวนที่กำหนดให้ (MAXSTEP) โดย NBEST นั้นแต่ละอนุภาคจะมีค่าเป็นของตัวเอง นั้นหมายความว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อนุภาคต่างๆ อาจจะถูกวิธีการเริ่มต้นการค้นหาใหม่ไม่พร้อมกัน โดยที่อนุภาคตัวใดตรงตามเงื่อนไขที่กำหนดนี้ก่อน อนุภาคตัวนั้นก็จะถูกดำเนินการวิธีการเริ่มต้นการค้นหาใหม่ก่อน

2) จำนวนวิธีการเริ่มต้นการค้นหาใหม่คือ จำนวนอนุภาคที่ถูกดำเนินการเริ่มต้นการค้นหาใหม่ในแต่ละรอบเริ่มต้นการค้นหาใหม่สามารถแบ่งได้สองประเภทคือเริ่มต้นการค้นหาใหม่อนุภาคใหม่ทั้งหมด [13] กับเริ่มต้นการค้นหาใหม่อนุภาคใหม่เพียงบางส่วน [34, 37]

งานวิจัย[13] เสนออนุภาคทั้งหมดเริ่มต้นใหม่ทั้งหมด คือเมื่อถึงรอบที่กำหนด ตำแหน่งของอนุภาคทั้งหมดในประชากรจะถูกกำหนดค่าตำแหน่งเริ่มต้นใหม่ทั้งหมด

งานวิจัย [37] เสนออนุภาคเริ่มต้นใหม่เพียงบางส่วน โดยจำนวนอนุภาคที่ถูกเริ่มต้นการค้นหาใหม่ขึ้นอยู่กับค่าความน่าจะเป็นในการเริ่มต้นการค้นหาใหม่ โดยถ้าค่าที่สุ่มระหว่าง 0 กับ 1 มีค่าน้อยกว่าค่าความน่าจะเป็นนี้ ดำเนินการเริ่มต้นการค้นหาใหม่ โดยจำนวนอนุภาคที่ถูกเริ่มต้นการค้นหาใหม่จะมากหรือน้อยขึ้นอยู่กับผู้ออกแบบขั้นตอน โดยส่วนมากมักจะมีค่ามาก เช่น 30%, 50%, 70% ของจำนวนมิติทั้งหมดของทุกอนุภาค เป็นต้น

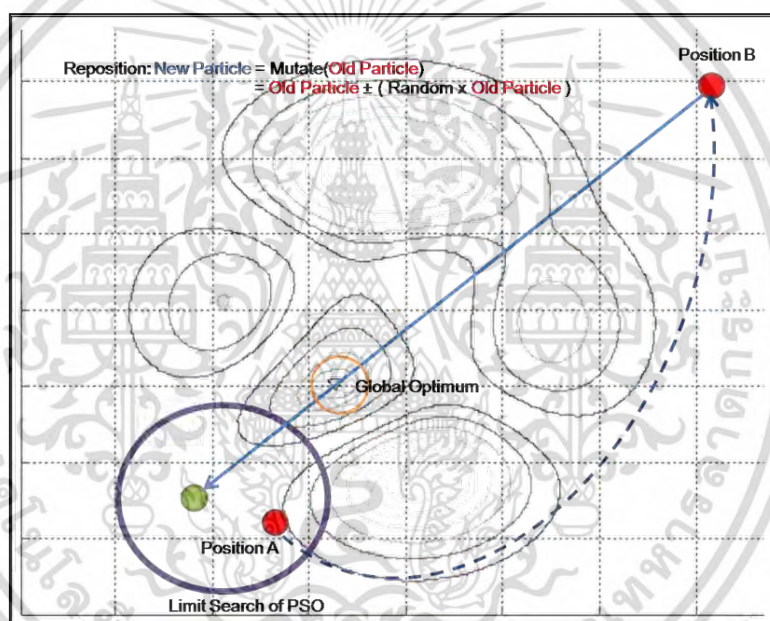
งานวิจัย [34] เสนออนุภาคเริ่มต้นใหม่เพียงบางส่วน โดยจำนวนอนุภาคที่ถูกเริ่มต้นใหม่ขึ้นอยู่กับรอบเริ่มต้นใหม่ของอนุภาคตัวเองในกรณีนี้คืออนุภาคแต่ละตัวจะมีรอบเริ่มต้นใหม่ของตัวเอง ถ้าอนุภาคตัวใดถึงรอบที่กำหนดก็จะดำเนินการเริ่มต้นใหม่ในขณะที่อนุภาคตัวอื่นอาจจะไม่ดำเนินการเริ่มต้นใหม่

3) ส่วนประกอบที่ถูกวิธีการเริ่มต้นการค้นหาใหม่ (Reset Component) คือส่วนประกอบของ PSO เช่น GBEST PBEST ตำแหน่งและความเร็วของอนุภาค ซึ่งส่วนประกอบเหล่านี้จะถูกเริ่มต้นการค้นหาใหม่โดยส่วนประกอบที่ถูกวิธีการเริ่มต้นการค้นหาใหม่เหล่านี้มีผลกระทบต่อการค้นหาที่แตกต่างกัน ตัวอย่างงานวิจัยเช่น กำหนดให้เริ่มต้นการค้นหาใหม่ตำแหน่งและความเร็วของอนุภาค [34, 37] และกำหนดให้เริ่มต้นการค้นหาใหม่ตำแหน่งของอนุภาคเพียงอย่างเดียว [50] เป็นต้น

### 3.2.4) ข้อดีและข้อเสียของการเริ่มต้นการค้นหาใหม่

หลักการทำงานของ PSO อนุภาคทั้งหมดนั้นพยายามเคลื่อนที่เข้าหา GBEST โดยระหว่างทาง ถ้าพบตำแหน่งที่ดีกว่า GBEST นำเอาตำแหน่งนั้นแทน GBEST และเมื่ออนุภาคเคลื่อนที่เข้าใกล้ GBEST อนุภาคค้นหาบริเวณรอบ GBEST เท่านั้น ถ้า GBEST อยู่ในบริเวณจุดสุดสัมพัทธ์ ผลก็คืออนุภาคทั้งหมดนั้นไม่สามารถพบคำตอบที่ดีกว่าจุดสุดสัมพัทธ์เสมือนกับการค้นหานั้นสิ้นสุด การค้นหาต่อไปจะไม่มีประโยชน์เพื่อให้การค้นหาต่อไปมีประโยชน์ หลังจากเกิดปัญหาจุดสุดสัมพัทธ์ จึงได้เพิ่มขั้นตอนวิธีการเริ่มต้นการค้นหาใหม่ โดยการกระจายอนุภาคออกจากบริเวณจุดสุดสัมพัทธ์ ทำเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการศึกษาไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้มีโอกาสในการค้นหาคำตอบใหม่อีกครั้ง จึงเป็นการเพิ่มโอกาสในการลองค้นหามากกว่า PSO ปกติ ดังนั้น วิธีนี้จึงได้คำตอบที่ดีกว่า PSO ปกติ โดยการเริ่มต้นการค้นหาใหม่จะเน้นการแก้ปัญหาการติดในจุดสุดสัมพัทธ์ คือเมื่อเกิดปัญหาการติดในจุดสุดสัมพัทธ์ขึ้นจึงดำเนินการ จากผลการทดลองของวิทยานิพนธ์ฉบับนี้ [14] แสดงให้เห็นว่า ขั้นตอนวิธีการเริ่มต้นการค้นหาใหม่สามารถเพิ่มประสิทธิภาพการค้นหาคำตอบของ PSO โดยผลลัพธ์ของการค้นหาคำตอบของ RERUN และ REPOSITION ดีกว่าของ PSO ปกติ จากรูป 3.15 GBEST (วงกลมสีส้ม) อยู่ที่ตำแหน่งของจุดสุดสัมพัทธ์ (ภายในวงกลมสีม่วง) ทำให้อนุภาคต่างๆ เคลื่อนที่เข้ามารวมกันที่ตำแหน่งจุดสุดสัมพัทธ์และค้นหาคำตอบบริเวณรอบจุดสุดสัมพัทธ์ ทำให้ไม่สามารถพบจุดสุดสัมบูรณ์ เมื่อเกิด RERUN หรือ REPOSITION ทำให้อนุภาค (วงกลมสีม่วง) กระโดดจากตำแหน่ง A ไปยังตำแหน่ง B หลังจากนั้นอนุภาคจะเคลื่อนที่กลับเข้ามายังจุดสุดสัมพัทธ์ ซึ่งมีโอกาสที่จะได้พบกับจุดสุดสัมบูรณ์ในระหว่างทาง

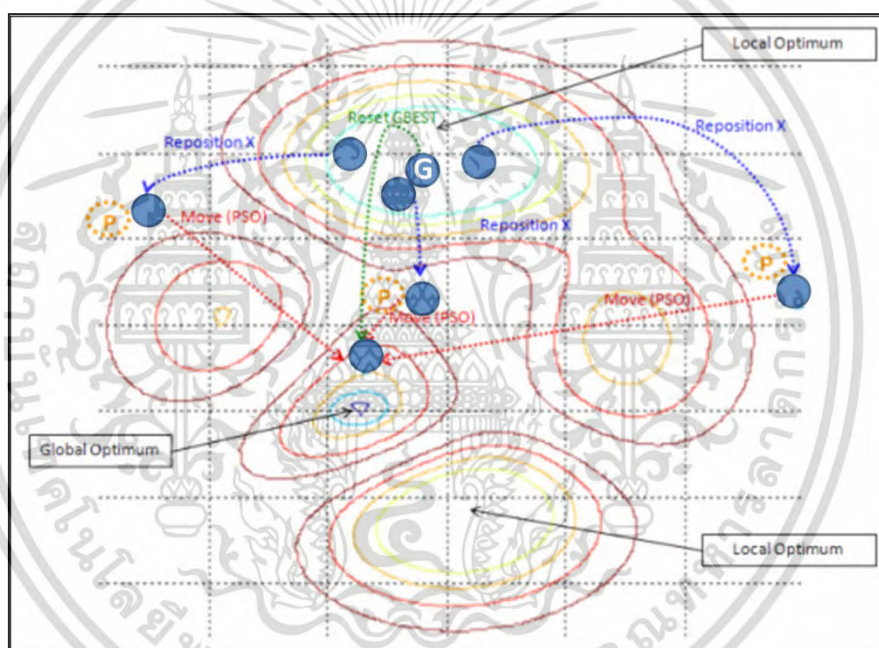


รูปที่ 3.15 การกระจายตำแหน่งของอนุภาคช่วยปรับปรุงคำตอบของ PSO

โดยจุดประสงค์ของ RERUN หรือ REPOSITION คือแก้ปัญหาการติดในจุดสุดสัมพัทธ์โดยให้อนุภาคกระจายตัวออกจากขอบเขตจุดสุดสัมพัทธ์แล้วไปค้นหาบริเวณอื่นนอกจากจุดสุดสัมพัทธ์เดิม การดำเนินการแบบ RERUN ได้ดำเนินการเกินความจำเป็น เพราะ RERUN ทำให้อนุภาคทั้งหมดกระจายตัวเป็นบริเวณกว้างต้องใช้เวลาเพื่ออนุภาคลู่เข้าหากันใหม่ จึงสูญเสียเวลาในการลู่เข้าหากันมาก ทำให้โอกาสการค้นหาในบริเวณอื่นลดลง ดังนั้น REPOSITION จึงมีประสิทธิภาพดีกว่า RERUN เพราะ REPOSITION สามารถกำหนดขอบเขตการกระจายตัวของอนุภาคจึงสามารถทำให้อนุภาคกระจายตัวออกจากกันเท่าที่จำเป็น ส่งผลให้อัตราการลู่เข้าหากันเร็วกว่า RERUN ทำให้ REPOSITION มีโอกาสการค้นหาในบริเวณอื่นมากกว่า RERUN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

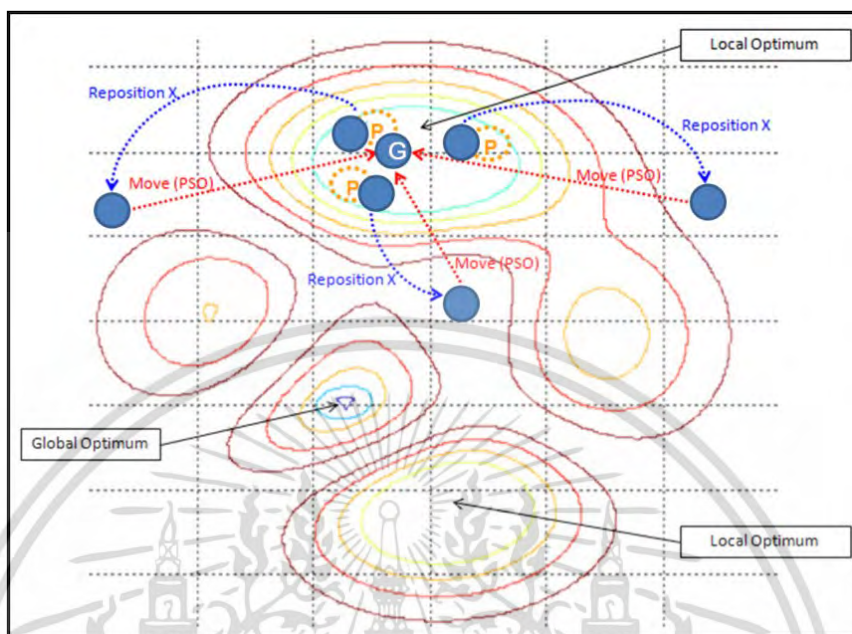
ปัจจัยสำคัญของการเริ่มต้นการค้นหาค่าตอบในบริเวณให้คือ การกำหนดค่าเริ่มต้นให้กับ PBEST และ GBEST โดยจากวิทยานิพนธ์ฉบับนี้ [14] แสดงให้เห็นว่า การกำหนดค่าเริ่มต้นให้กับ PBEST และ GBEST ใหม่ ทำให้อนุภาคเคลื่อนที่ไปค้นหาค่าตอบในบริเวณใหม่ และได้ผลลัพธ์การค้นหาค่าตอบที่ดีกว่า จากรูป 3.16 อนุภาคถูกดำเนินการ REPOSITION ทำให้อนุภาคกระจายตัวออกจากจุดสุดสัมพัทธ์ตามลูกศรเส้นประสีน้ำเงินสำหรับ PBEST คือวงกลมเส้นประสีส้มและ GBEST คือวงกลมตัวอักษร G ซึ่งทั้ง PBEST และ GBEST ถูกกำหนดค่าเริ่มต้นใหม่ ทำให้ GBEST เปลี่ยนตำแหน่งไปตามลูกศรเส้นประสีเขียวและ PBEST กระจายตัวออกจากจุดสุดสัมพัทธ์และอยู่ใกล้กับอนุภาคในตำแหน่งใหม่ เมื่อดำเนินการค้นหาด้วย PSO อนุภาคจะลู่กลับเข้ามา GBEST ตามลูกศรเส้นประสีแดง ซึ่งเป็นการลู่เข้าหาตำแหน่งบริเวณใหม่ทำให้การค้นหาที่นั้นเกิดการค้นหาค่าตอบในบริเวณใหม่จึงมีโอกาสพบคำตอบที่ดีขึ้นกว่าเดิม



รูปที่ 3.16 การกำหนดค่าเริ่มต้นให้กับ PBEST และ GBEST ใหม่

การที่ไม่กำหนดค่าเริ่มต้นให้กับ PBEST และ GBEST ใหม่ เพราะเมื่อดำเนินการ REPOSITION แล้วอนุภาคจะกระจายตัวออกจากกัน หลังจากการนั้นอนุภาคจะดำเนินการค้นหาด้วย PSO อนุภาค แล้วเคลื่อนที่กลับเข้ามารวมกันที่ GBEST เดิม หรือ PBEST เดิมเสมือนว่าการค้นหาไม่ได้ค้นหาในบริเวณใหม่ แต่กลับเคลื่อนที่กลับเข้ามาค้นหาในบริเวณเดิมเพียงแต่ระหว่างทางที่เคลื่อนที่กลับเข้ามาหา GBEST อาจพบตำแหน่งที่ดีกว่า GBEST เดิมจึงสามารถปรับปรุงคำตอบ แต่การค้นหาไม่ได้เกิดจากการค้นหาในบริเวณใหม่ จากรูป 3.17 คือภาพสาธิตของ REPOSITION ที่ไม่มีขั้นตอนการไม่กำหนดค่าเริ่มต้นให้กับ PBEST และ GBEST ใหม่ อนุภาคถูกดำเนินการรีโพซิชั่น ทำให้อนุภาคกระจายตัวออกจากจุดสุดสัมพัทธ์ตามลูกศรเส้นประสีน้ำเงิน แต่เนื่องจาก PBEST คือ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงกลมเส้นประสีส้มและ GBEST คือวงกลมตัวอักษร G ที่ตำแหน่งจุดสุดสัมพัทธ์เดิม เมื่อดำเนินการค้นหาด้วย PSO อนุภาคจะถูกลบเข้ามา GBEST ซึ่งอยู่ในจุดสุดสัมพัทธ์เดิมตามลูกศรเส้นประสีแดง



รูปที่ 3.17 การกำหนดค่าเริ่มต้นที่ไม่มีการกำหนดค่าเริ่มต้นให้กับ PBEST และ GBEST

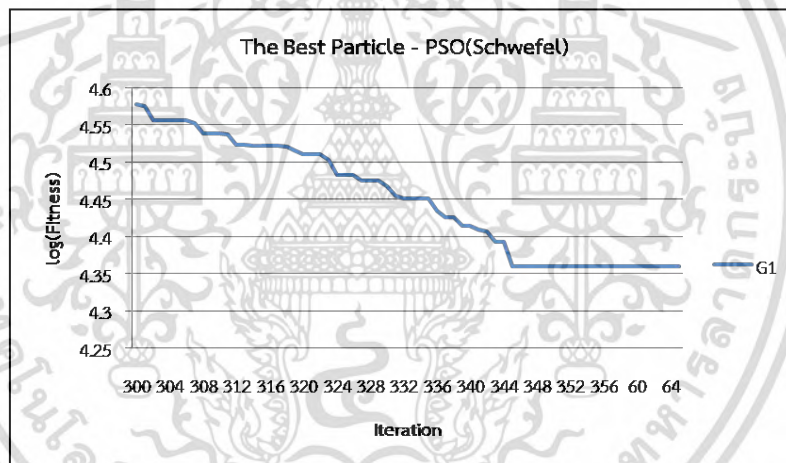
RERUN และ REPOSITION ช่วยลดปัญหาการติดในจุดสุดสัมพัทธ์แต่ผลจาก RERUN และ REPOSITION ทำให้เกิดปัญหาใหม่ คือปัญหาความยุ่งยากในการปรับพารามิเตอร์ให้เหมาะสมกับปัญหาที่แก้ไข โดยถ้าดำเนินการ RERUN และ REPOSITION มากเกินไป RERUN และ REPOSITION จะขัดขวางกระบวนการลู่เข้าหากันของ PSO ในทางกลับกันถ้าดำเนินการ RERUN และ REPOSITION น้อยเกินไป RERUN และ REPOSITION จะไม่สามารถแก้ปัญหาค้นหาจุดสุดสัมพัทธ์ โดยจากวิทยานิพนธ์ฉบับนี้ [14] ได้แสดงให้เห็นผลกระทบจากการปรับค่าพารามิเตอร์ไม่เหมาะสมทำให้ผลลัพธ์จากการค้นหาคำตอบที่ได้นั้นแย่กว่า PSO ปกติ ถ้าดำเนินการ RERUN และ REPOSITION มากเกินไป และในทางกลับกันถ้าดำเนินการ RERUN และ REPOSITION น้อยเกินไป คำตอบที่ได้ให้ผลลัพธ์ที่ไม่ต่างจาก PSO ปกติ

## บทที่ 4

# การค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย

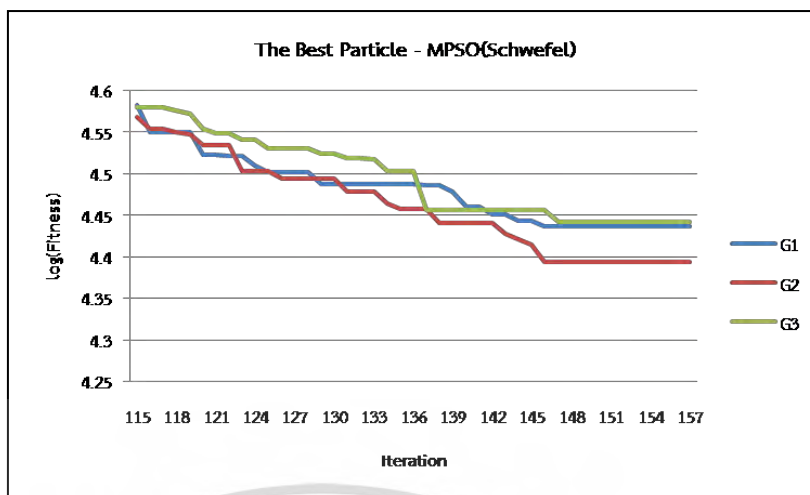
### 4.1 ข้อเสียของวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม

เพื่อแก้ปัญหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม PSO ดังรูปที่ 4.1 นั้นได้มีนักวิจัยหลายคนได้ทำการปรับปรุง PSO โดยการแบ่งกลุ่มอนุภาคออกเป็นกลุ่มย่อยเรียก วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม (MPSO) โดยที่แต่ละกลุ่มจะมีตำแหน่งของอนุภาคที่ดีที่สุด (GBEST) ของตัวเอง ดังรูปที่ 4.2



รูปที่ 4.1 ตัวอย่างกราฟวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค (PSO)

จากกราฟ 4.1 ซึ่งมาจากผลการทดลองในหัวข้อ 5.3 ที่แสดงค่าความเหมาะสมที่ดีที่สุด (GBEST) ในแต่ละรอบการทำงาน (Iteration) ของ PSO เริ่มตั้งแต่ช่วงแรกจนถึงคงที่ที่จะเห็นได้ว่าเส้นกราฟในช่วงแรกจะมีค่าลดลงหรือดีขึ้นเนื่องจากการทำงานของ PSO ทำให้สามารถหาค่าความเหมาะสมของอนุภาคได้ดีขึ้นเรื่อยๆ หลังจากนั้นเส้นกราฟจะคงที่นั่นคือการค้นหาของ PSO ไม่สามารถหาค่าที่ดีกว่านี้ได้เนื่องจากเกิดปัญหาการติดในจุดสุดสัมพัทธ์จนสิ้นสุดการค้นหาตามรอบที่กำหนด

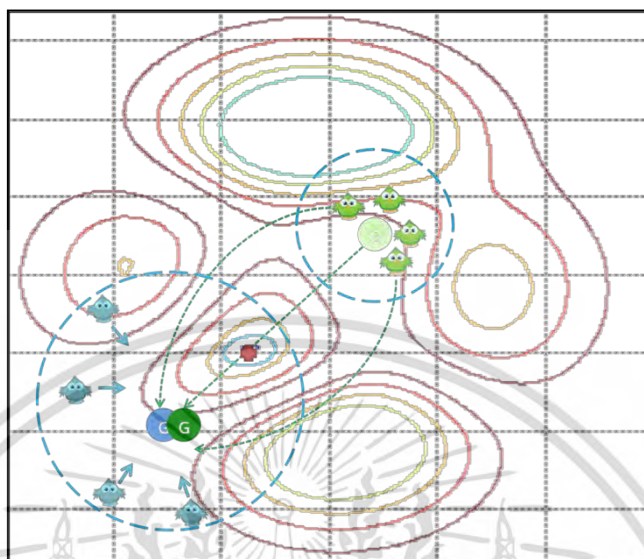


รูปที่ 4.2 ตัวอย่างกราฟวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม (MPSO)

จากรูปกราฟ 4.2 มาจากผลการทดลองของ PSO แบบหลายกลุ่ม (MPSOSM ที่เสนอในงานวิจัยนี้) จะมีเส้นกราฟ 3 เส้นแต่ละเส้นเป็นของแต่ละกลุ่มที่มีการค้นหาแบบอิสระจากกัน จะเห็นว่าเส้นกราฟในช่วงแรกของแต่ละกลุ่มนั้นจะมีค่าลดลงหรือดีขึ้นจากกระบวนการของ PSO ที่ทำให้อนุภาคของแต่ละกลุ่มสามารถค้นหาค่าความเหมาะสมของอนุภาคได้ดีขึ้นเรื่อย ๆ หลังจากนั้นเส้นกราฟจะคงที่ไม่ดีขึ้นในใจ นวนรอบที่เร็วกว่า PSO กลุ่มเดียวเนื่องจาก งานอนุภาคของแต่ละกลุ่มน้อยเป็น 1 ใน 3 ของ PSO กลุ่มเดียวถึงแม้จะมีการกระจายอนุภาคออกเป็น 3 กลุ่มที่ค้นหาต่างที่กัน แต่ความสามารถในการค้นหาแบบเฉพาะจุด (Local Search) ลดลงจึงเกิดปัญหาการติดในจุดสุดสัมพัทธ์ จนสิ้นสุดการค้นหาได้ง่ายกว่า PSO กลุ่มเดียว

เพื่อใช้ประโยชน์จากการค้นหาที่กว้างขึ้นของ PSO แบบหลายกลุ่ม ในกรณีที่แต่ละกลุ่มไม่ติดที่จุดในจุดสัมพัทธ์เดียวกัน สามารถกำหนดให้อนุภาคในกลุ่มที่ติดและไม่สามารถจะหาค่าที่ดีขึ้นได้แล้วเคลื่อนที่เข้าหา GBEST ของกลุ่มอื่น โดยหวังว่าระหว่างทางจะเจอ GBEST ที่ดีขึ้น หรืออาจจะเจอค่าที่ลดลงแต่ยังทำให้อนุภาคภายในกลุ่มสามารถหลุดออกจากการติดในจุดสัมพัทธ์ และเคลื่อนที่ไปค้นหาในบริเวณอื่นได้และอาจจะเจอค่าที่ดีขึ้นในที่สุด ดังนั้น เมื่อกลุ่มใดกลุ่มหนึ่งติดในจุดสุดสัมพัทธ์จะทำการเปลี่ยนตำแหน่ง GBEST เป็นของกลุ่มอื่นที่ไม่ติดในจุดสัมพัทธ์เดียวกัน ร่วมกับทำการเปลี่ยนตำแหน่งของอนุภาคพร้อมกันด้วยเพื่อไม่ให้อนุภาคติดอยู่ในบริเวณเดิม

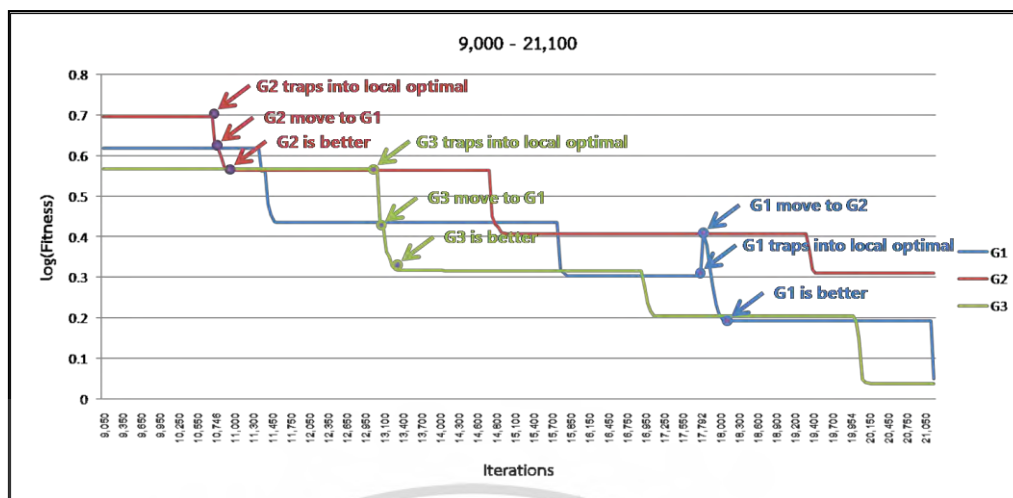
## 4.2 การปรับปรุงขั้นตอนวิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบหลากหลายกลุ่ม โดยการใช้ตำแหน่งที่ดีที่สุดของกลุ่มอื่น



รูปที่ 4.3 การเปลี่ยนตำแหน่งที่ดีที่สุดของกลุ่มอนุภาคกับกลุ่มอื่น

ขั้นตอนวิธีการเปลี่ยนตำแหน่งที่ดีที่สุดของอนุภาคทั้งหมดกับกลุ่มข้างเคียงนี้ช่วยเพิ่มความหลากหลายให้กับอนุภาคภายในกลุ่ม โดยดูได้จากรูปตัวอย่างที่ 4.3 จะเห็นว่าเมื่ออนุภาคที่ดีที่สุดของกลุ่ม G (สีเขียว) เกิดปัญหาการติดในจุดสุดสัมพัทธ์แล้ว GBEST ของกลุ่มจะถูกเปลี่ยนตำแหน่งเป็นตำแหน่ง GBEST ของกลุ่ม G (สีฟ้า) จะเห็นได้ว่าอนุภาคภายในกลุ่ม G (สีเขียว) มีโอกาสเคลื่อนผ่านตำแหน่งที่ดีที่สุดไปยัง GBEST ค่าใหม่ จากรูปคือบริเวณจุดสีแดงทำให้อาจพบคำตอบที่ดีที่สุดขึ้น และลดการติดในจุดสุดสัมพัทธ์ของอนุภาคได้

จากที่เหตุผลที่กล่าวมาทั้งหมดนี้ งานวิจัยนี้จึงได้นำเสนอให้แต่ละกลุ่มท การเปลี่ยนตำแหน่ง GBEST ของกลุ่มอื่น เมื่อกลุ่มนั้นเกิดปัญหาการติดในจุดสุดสัมพัทธ์ ดังรูปกราฟที่ 4.3 โดยทำการเปลี่ยน GBEST ของกลุ่มตัวเองให้กลายเป็น GBEST ของกลุ่มอื่น โดยการสุ่มเลือก เพื่อให้ประชากรกลุ่มนั้น เคลื่อนย้ายไปยัง GBEST ของกลุ่มอื่น ทำให้หลุดออกจากจุดสุดสัมพัทธ์ โดยที่ GBEST ของกลุ่มที่ติดจุดสุดสัมพัทธ์นั้น จะถูกเปลี่ยนค่าเป็น GBEST ของกลุ่มอื่นที่ไม่ซ้ำกับตัวเอง เป็นการแก้ปัญหาการติดในจุดสุดสัมพัทธ์ และไม่เพิ่มจำนวนในการเดาค ทอปในแต่ละรอบการค้นหาคาตอบด้วย



รูปที่ 4.4 กราฟการไล่ตามหาตำแหน่ง GBEST ของกลุ่มข้างเคียง

จากกราฟรูปที่ 4.4 แสดงช่วงในช่วงปลายของรอบการทำงาน PSO ที่มีการแบ่งอนุภาคออกเป็น 3 กลุ่มย่อยโดยอยู่ในรอบ 9,000– 21,000 จะเห็นว่ากลุ่ม G2 ติดอยู่ในจุดสุดสัมพัทธ์และโดนเลือกให้เคลื่อนที่ไปยัง GBEST ของกลุ่ม G1 ที่มีค่าดีกว่าเดิม และในที่สุดอนุภาคในกลุ่ม G2 หลุดออกจากจุดสุดสัมพัทธ์เดิมและเคลื่อนที่ไปยังจุดที่ดีขึ้นทั้งยังได้พบจุดที่ดีขึ้นกว่าจุดเดิมของกลุ่ม G1 อีกด้วย

ในช่วง 12,450 – 13,000 กลุ่ม G3 ได้ติดในจุดสุดสัมพัทธ์ ก่อนที่จะเคลื่อนที่เปลี่ยนตำแหน่งไปยัง GBEST ของ G1 ที่ดีขึ้นทำให้หลุดจากจุดสัมพัทธ์เดิม ก่อนที่อนุภาคของกลุ่ม G3 จะสามารถหาตำแหน่งที่ดีขึ้นกว่า GBEST ของกลุ่ม G1 ที่เปลี่ยนตำแหน่งไปหา ในขณะที่ GBEST ของกลุ่ม G1 ยังมีค่าอยู่ประมาณเดิมที่แย่กว่า GBEST ใหม่ของกลุ่ม G3

นอกจากนี้ในช่วง 17,792-18,300 กลุ่ม G1 ติดอยู่ในจุดสุดสัมพัทธ์และโดนสุ่มเลือกให้เคลื่อนที่ไปที่ GBEST ของ G2 ที่แย่กว่าเดิม แต่ทำให้อนุภาคของกลุ่ม G1 สามารถหาตำแหน่งที่ดีขึ้นกว่าตำแหน่งเดิมของกลุ่ม G1 ได้ ในกรณีนี้ จะเห็นว่าตำแหน่งของ G1 ดีกว่า G2 จึงจำเป็นต้องปรับตำแหน่งของอนุภาคของกลุ่ม G1 มิฉะนั้นอนุภาคจะไม่เคลื่อนที่ไปที่ GBEST ของกลุ่ม G2 ที่มีค่าแย่กว่า แต่ติดอยู่ในจุดสุดสัมพัทธ์เดิมของกลุ่ม G1

วิธีที่นำเสนอในงานวิจัยนี้ [15] มีขั้นตอนการทำงานเช่นเดียวกับ PSO ปกติเพียงแต่มีการแบ่งการทำงานออกเป็นหลายกลุ่ม โดยที่แต่ละกลุ่มจะมีค่า GBEST ของตัวเองและในกรณีที่ค่า GBEST ของกลุ่มใดกลุ่มหนึ่งไม่ดีขึ้นเกินจำนวนรอบที่กำหนด จะทำการปรับตำแหน่งของอนุภาคในกลุ่มนั้นและตั้งค่าตำแหน่งของ GBEST ของกลุ่มนั้นใหม่ให้เท่ากับ GBEST ที่ได้จากการสุ่มของกลุ่มอื่นที่ไม่ติดอยู่ในจุดสุดสัมพัทธ์เดียวกัน จะให้ได้ว่าขั้นตอนเหล่านี้ไม่ได้มีขั้นตอนการทำงานที่เพิ่มขึ้นจาก PSO ปกติ ยกเว้นการปรับตำแหน่งใหม่ของอนุภาคตอนที่ติดจุดสุดสัมพัทธ์เท่านั้น

### 4.3 การกระจายตำแหน่งของอนุภาคเพื่อปรับปรุงการค้นหาแบบหลากหลายกลุ่ม

ขั้นตอนการบังคับให้กลุ่มอนุภาคที่ติดอยู่ในจุดสุดสัมพัทธ์เลื่อนไปยัง GBEST ของกลุ่มอื่นเพื่อทำให้อนุภาคหลุดออกจากตำแหน่งเดิมที่ติดในจุดสุดสัมพัทธ์นั้น จะเปลี่ยนแต่ค่า GBEST อย่างเดียวนั้นไม่เพียงพอ โดยเฉพาะในกรณีที่อนุภาคของกลุ่มที่ติดในจุดสุดสัมพัทธ์นั้นอยู่ในตำแหน่งที่ดีกว่า GBEST ใหม่ กลุ่มอนุภาคนั้นในที่สุดจะเคลื่อนที่กลับมาที่ตำแหน่งเดิมที่มีค่าดีกว่า นอกจากนี้ เพื่อให้กลุ่มอนุภาคมีการค้นหาที่กว้างขึ้น จึงจำเป็นต้องปรับตำแหน่งของอนุภาคให้มีค่าไม่เหมือนเดิม แต่ในกรณีที่ค่าของจุดสุดสัมพัทธ์นั้นมีค่าใกล้เคียงกับจุดสูงสุดสัมบูรณ์ (Global optimum) การปรับค่าตำแหน่งของอนุภาคที่มากเกินไปจะมีค่าเท่ากับการเริ่มค้นหาใหม่

ในงานวิจัยนี้ได้ทำการทดลองปรับค่าตำแหน่งของอนุภาคเมื่อมีการติดอยู่ในจุดสุดสัมพัทธ์ก่อนที่จะทำการกำหนด GBEST ใหม่ ให้อนุภาคในกลุ่มเคลื่อนที่ไปหาโดยใช้สมการในการปรับค่าเช่นเดียวกับที่ใช้ในการกลายพันธุ์ของ GA ดังแสดงในสมการที่ (4.1)

$$x_{id}' = x_{id} \pm (x_{id} \times rand()) \quad (4.1)$$

โดย  $x_{id}'$  แทนตำแหน่งของอนุภาคตัวที่  $i$  ในมิติ  $d$  ค่าสุ่มที่ได้จากฟังก์ชัน  $rand()$  มีค่าอยู่ในช่วง  $(0, 1)$  ดังนั้นค่า  $x_{id}'$  ใหม่ที่ได้จะมีค่าอยู่ในช่วง  $(0, 2x_{id})$  โดยกรณีที่เป็น 0 ได้มาจากค่าสุ่มจาก  $rand()$  เป็น 1 และในการทำสุ่มการทำบวกและลบเป็นการทำการลบ ซึ่งในกรณีนี้การกลายพันธุ์ตามสมการที่ (4.1) จะมีโอกาสปรับค่าลงเป็น 0 หรือเป็นค่าที่ลดลงเรื่อยๆได้ง่าย และสมการในการปรับค่านี้อาจจะไม่เหมาะสมในกรณีที่ค่าตำแหน่งที่ต้องการไม่ได้อยู่ที่ตำแหน่ง 0

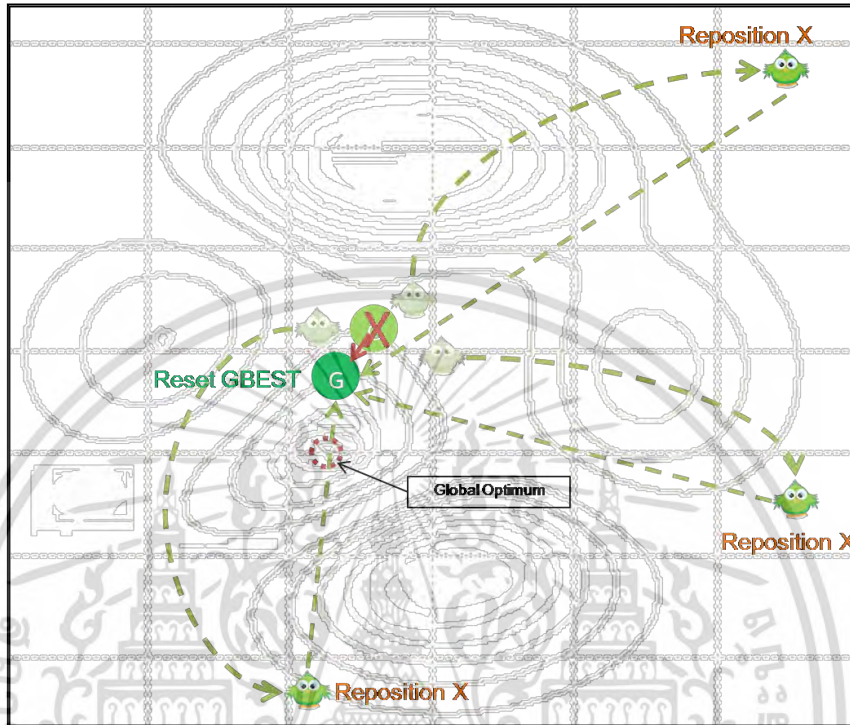
ดังนั้นจึงปรับเปลี่ยนสมการการกลายพันธุ์เป็นสมการที่ (4.2)

$$x_{id}' = \pm(x_{id} \times (0.85 + 0.3 \times rand()))_i \quad (4.2)$$

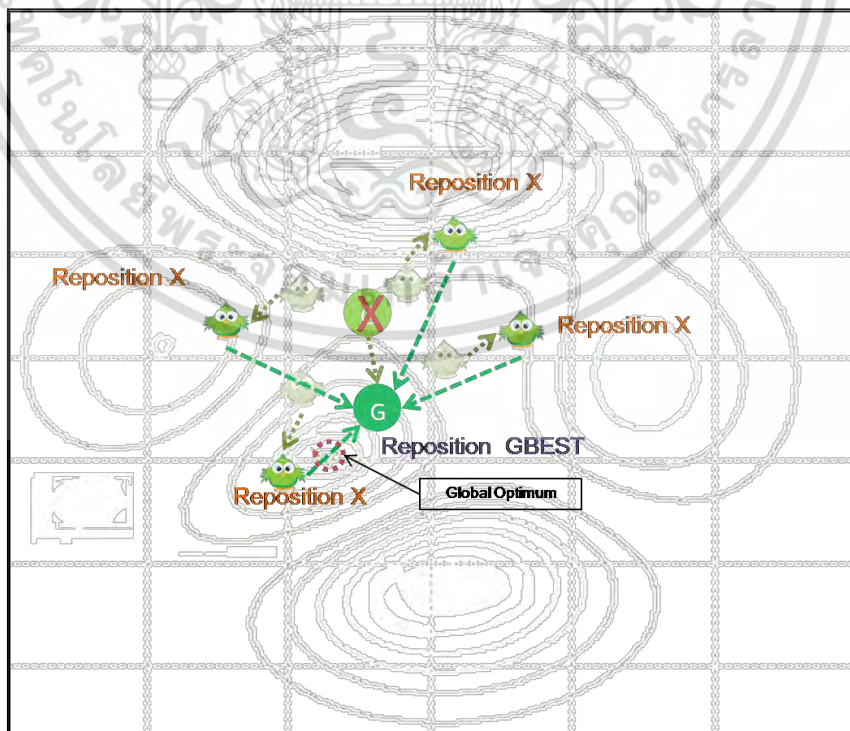
ในสมการนี้ค่าใหม่ที่ได้ จะอยู่ในช่วง  $(0.85)$  ของค่าเดิม ถึง  $(1.15)$  ของค่าเดิมขึ้นกับค่าที่สุ่มได้เป็น 0 และ 1 ตามลำดับ โดยอาจจะมีโอกาสในการกลับเครื่องหมายทั้งบวกและลบได้ 50 เปอร์เซ็นต์ จะเห็นว่าสมการนี้มีโอกาสเกิดค่าใหม่ที่เป็น 0 ได้น้อยลงและมีโอกาสเปลี่ยนเครื่องหมายได้ง่ายขึ้น นอกจากนี้กรณีที่ตำแหน่งของค ตอบไม่ได้อยู่ที่ 0 โอกาสที่การค้นหาจะติดตำแหน่งที่ใกล้เคียงกับคำตอบมีมากขึ้น และปริมาณการปรับจะอยู่ในช่วงที่แคบลง คือ  $(0.85$  ถึง  $1.15)$  ของค่าเดิมเมื่อเทียบกับสมการ (4.1) ที่อยู่ในช่วง  $(0$  ถึง  $2x_{id})$  ของค่าเดิม ดังแสดงใน รูปที่ 4.5 และ 4.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่การปรับปรุงตำแหน่งของทั้งสองวิธีนั้นเป็นการสุ่มเปลี่ยนตำแหน่งในมิติทั้งหมดของอนุภาคแต่ละตัวเพื่อให้อนุภาคมีโอกาสในการค้นหาออกไปยังตำแหน่งอื่นโดยมีเปอร์เซ็นต์ที่เหมาะสมที่แตกต่างไปตามสมการ



รูปที่ 4.5 การกระจายตำแหน่งของอนุภาคตามแบบสมการที่ (4.1)



รูปที่ 4.6 การกระจายตำแหน่งของอนุภาคตามแบบสมการที่ (4.2)

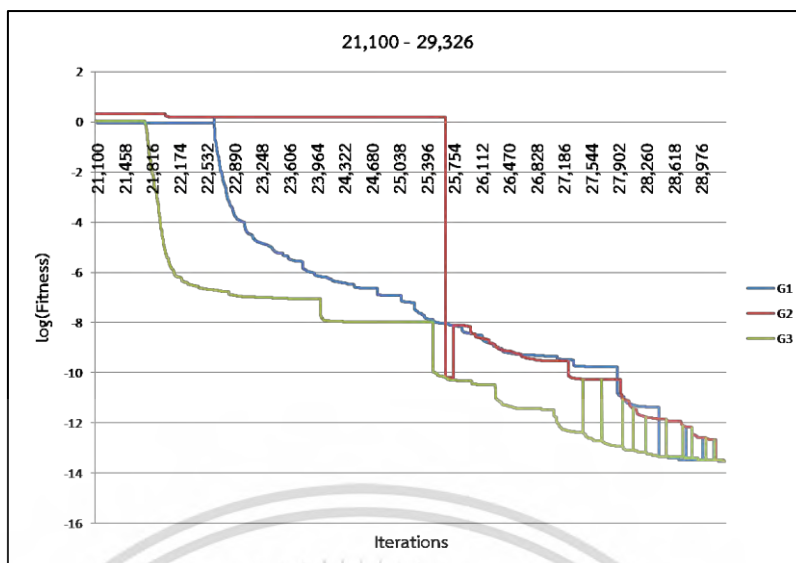
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากการกำหนดขอบเขตให้กับการปรับปรุงตัวแปรหนึ่งโดยเทียบกับค่าของตัวแปรหนึ่งเดิมแล้ว จำนวนมิติของตัวแปรหนึ่งที่ต้องปรับก็เป็นอีกปัจจัยหนึ่งที่จะทำให้ค่าที่ปรับดีขึ้น เช่น ในหลายกรณีจุดสูงสุดสมบูรณ์ (Global optimum) อยู่ใกล้เคียงตำแหน่งเดิมของอนุภาค โดยที่ค่าที่แตกต่างกันมีอยู่ในเพียงบางมิติ ดังนั้นในการเปลี่ยนตำแหน่งของอนุภาคในทุกมิติ นั้นเท่ากับเป็นการเริ่มต้นค้นหาใหม่ ทำให้เสียเวลามาก และอาจจะทำให้อนุภาคไปติดในจุดสุดสัมพัทธ์อื่น ในงานวิจัยนี้ได้ทดลองเปลี่ยนเปอร์เซ็นต์ของมิติทั้งหมดของอนุภาคจาก 10 เปอร์เซ็นต์ของมิติทั้งหมด จนถึง 100 เปอร์เซ็นต์ของมิติทั้งหมด ดังแสดงอยู่ใน ภาคผนวก โดยที่ในงานวิจัยนี้ เลือกปรับตำแหน่งของอนุภาคโดยใช้สมการที่ (4.2) ใน 20 เปอร์เซ็นต์ของจำนวนมิติทั้งหมดของตัวแปรอนุภาคแต่ละตัว

ในงานวิจัยก่อนหน้า [15] ได้ใช้สมการที่ (4.1) และมีจำนวนมิติที่ใช้ในการปรับตำแหน่งของอนุภาคเมื่อติดอยู่ในจุดสุดสัมพัทธ์เป็น 70 เปอร์เซ็นต์ของจำนวนมิติทั้งหมดและในงานวิจัย [16] ใช้สมการที่ (4.2) และปรับตำแหน่งของอนุภาคเมื่อติดอยู่ในจุดสุดสัมพัทธ์เป็น 20 เปอร์เซ็นต์ของจำนวนมิติทั้งหมด ได้ผลที่ดีขึ้น เนื่องจากทั้งสองวิธีในแต่ละงานวิจัยมีข้อแตกต่างอย่างอื่น เพื่อเป็นการทดลองเฉพาะในส่วนของสมการที่ (4.1) และเปอร์เซ็นต์ในการกระจายตำแหน่งของอนุภาค ได้ทดลองทำการเปรียบเทียบ PSO แบบหลายกลุ่มที่เมื่อกลุ่มใดกลุ่มหนึ่งติดในจุดสัมพัทธ์ จะทำการปรับตำแหน่งอนุภาคตามสมการที่ (4.1) และเปลี่ยนมิติของอนุภาคเป็น 70 เปอร์เซ็นต์ตามขั้นตอน [15] โดยวิธีนี้มีชื่อย่อ MPSOR1 และ PSO แบบหลายกลุ่มอีกวิธีหนึ่งที่ใช้สมการที่ (4.2) มาใช้ในการปรับตำแหน่งของอนุภาค และเปลี่ยนมิติของอนุภาคตามสมการเป็น 20 เปอร์เซ็นต์ตามขั้นตอน [15] โดยวิธีนี้มีชื่อย่อ MPSOR2 ทั้งสองวิธีไม่มีการเปลี่ยนค่า GBEST เป็นของของกลุ่มอื่นหลังติดอยู่ในจุดสุดสัมพัทธ์ แต่จะใช้ค่า GBEST ตามตำแหน่งที่ดีที่สุดของอนุภาคหลังจากการปรับตำแหน่ง โดยผลที่ได้แสดงในตารางที่ 5.1 และวิธี MPSOR2 มีค่าที่ดีกว่า

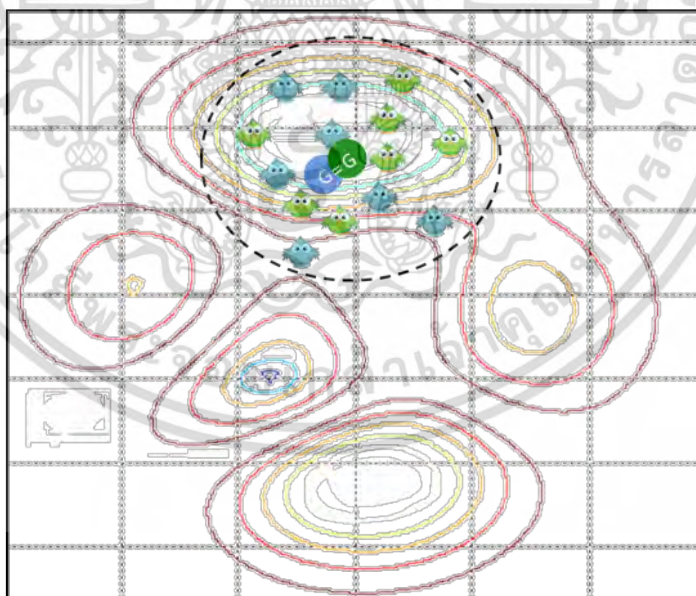
#### 4.4 วิธีแก้ปัญหาในกรณีที่อนุภาคของทุกกลุ่มติดอยู่ที่จุดสุดสัมพัทธ์เดียวกัน

จากวิธีการค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่มด้วยการแลกเปลี่ยนตำแหน่งของอนุภาคที่ดีที่สุด (GBEST) ร่วมกับการปรับเปลี่ยนตำแหน่งของอนุภาคนั้น ถ้าดำเนินการค้นหาในระยะเวลาหนึ่งอนุภาคของแต่ละกลุ่มจะเคลื่อนที่เข้าหากันดังเช่นรูปกราฟที่ 4.7 จะทำให้ GBEST ทุกกลุ่มมีโอกาสติดอยู่ในจุดติดในจุดสุดสัมพัทธ์เดียวกัน ซึ่งวิธีการแลกเปลี่ยนตำแหน่งของ GBEST ระหว่างกลุ่มนั้นไม่สามารถช่วยปรับปรุงค่าที่ดีที่สุดได้ เนื่องจาก GBEST ของทุกกลุ่มอยู่ที่ตำแหน่งเดียวกัน



รูปที่ 4.7 กราฟการเปลี่ยนตำแหน่งของอนุภาค GBEST

จากรูปกราฟ 4.7 อธิบายผลการทดลองของงานวิจัยที่ได้นำเสนอในช่วง 21,100 – 29,326 โดยจะเห็นว่าเมื่ออนุภาคของแต่ละกลุ่มติดอยู่ในจุดสุดสัมพัทธ์ นั้นจะทำการเปลี่ยนตำแหน่งของอนุภาคที่ดีที่สุดของกลุ่มกับกลุ่มอื่นโดยการสุ่ม เป็นผลทำให้อนุภาคของแต่ละกลุ่มเคลื่อนที่เข้าหากัน จนทำให้ทุกกลุ่มติดอยู่ในจุดสุดสัมพัทธ์เดียวกัน



รูปที่ 4.8 อนุภาคทุกกลุ่มติดอยู่ในจุดสุดสัมพัทธ์เดียวกัน

ในกรณีที่อนุภาคของแต่ละกลุ่มติดอยู่ในจุดสุดสัมพัทธ์เดียวกันจะทำการปรับตำแหน่งอนุภาคในหัวข้อที่ 4.2 ไม่สามารถทำได้ ดังนั้น เพื่อให้อนุภาคของแต่ละกลุ่มมีโอกาสหลุดออกจากตำแหน่งที่ติดอยู่ในจุดสุดสัมพัทธ์เดียวกันและมีโอกาสเคลื่อนที่ค้นหาออกไปที่ใหม่ได้ จะต้องหาตำแหน่ง GBEST

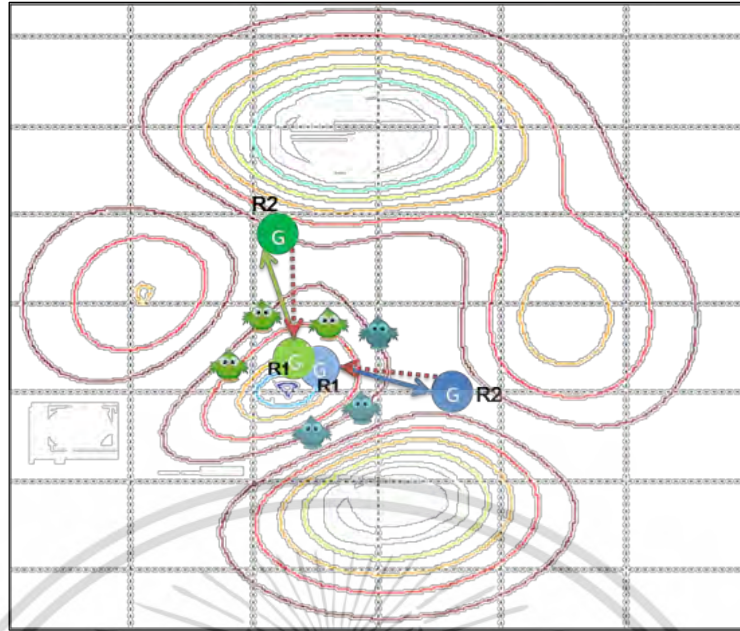
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ใช้ในการกำหนดให้แต่ละกลุ่มเคลื่อนที่ไปหาให้แตกต่างจากตำแหน่งเดิม โดย GBEST นี้อาจได้มาจากการปรับตำแหน่งตามสมการที่ (4.2) ของ GBEST เดิม หรืออาจได้จากการปรับตำแหน่งของ GBEST ที่ได้มาจากตำแหน่งที่ดีที่สุดของอนุภาคทั้งหมดในกลุ่มหลังการปรับตำแหน่งตามสมการที่ (4.2) โดยแสดงในสมการที่ (4.3)

$$GBEST'_i = \pm(GBEST \times (0.85 + 0.3 \times rand))_i \quad (4.3)$$

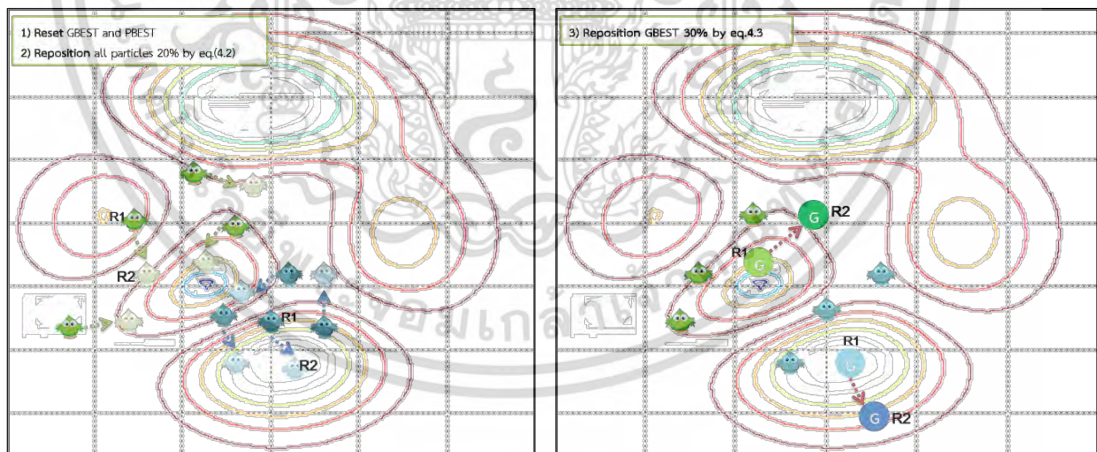
จากการทดลอง ถ้าใช้ค่า GBEST เดิมและใช้สมการที่ (4.3) ปรับค่า GBEST นั้น จะทำให้อนุภาคเคลื่อนที่เข้าสู่จุดสุดสัมพัทธ์เดิม โดยแสดงในรูป 4.9 ดังนั้น จึงเลือกเปลี่ยนตำแหน่งของ GBEST หลังการปรับปรุงตำแหน่งของอนุภาคตามสมการที่ (4.2) ไปแล้ว แต่ผลที่ได้มีการเปลี่ยนแปลงน้อยไป เนื่องจากมีค่าในบางมิติมีค่าที่ไม่ถูกต้องและจำเป็นต้องมีการค้นหาในมิติอื่นอีก แต่อนุภาคมีการสุ่มปรับเพียง 20 เปอร์เซ็นต์ของจำนวนมิติทั้งหมด จึงมีหลายมิติที่มีค่าเหมือนกันทุกอนุภาคและถ้าค่า GBEST ในมิตินั้นมีค่าเท่ากัน กับค่าของอนุภาคในมิตินั้นด้วย ทำให้อนุภาคไม่มีการปรับเปลี่ยนแปลงตามสมการของ PSO ดังนั้นค่าที่ดีที่สุดของอนุภาคหลังการปรับปรุงตำแหน่งยังจะใช้เป็นค่า GBEST ไม่ได้ วิธีนี้แสดงในรูป 4.10 (ซ้าย) จะต้องทำการเปลี่ยนตำแหน่งที่ดีที่สุดที่ได้จากการคัดเลือกจากอนุภาคตามสมการที่ (4.3) ก่อนอีกรอบหนึ่งและในกรณีนี้จะเพิ่มจำนวนมิติที่ต้องเปลี่ยนของอนุภาค GBEST ใหม่เป็น 30 เปอร์เซ็นต์ วิธีนี้แสดงในรูป 4.10 (ขวา)

แต่ทั้งนี้เนื่องจากค่าตำแหน่ง GBEST มีการปรับอีกรอบหนึ่ง โดยมากมักจะทำให้ค่าคำตอบของ GBEST นั้นมีค่าต่ำกว่าคำตอบของ GBEST เดิม ซึ่งตำแหน่ง GBEST นี้มักจะโดนแทนที่ด้วยค่า GBEST ใหม่ที่ดีกว่าเนื่องจากอนุภาคภายในกลุ่มที่เหลือมักจะอยู่ในตำแหน่งที่ดีกว่า ในบางครั้งวิธีการใช้ค่า GBEST ในการดึงกลุ่มอนุภาคออกจากการติดในจุดสุดสัมพัทธ์นี้ จึงทำได้เพียงบางอนุภาคในกลุ่มเท่านั้น



รูปที่ 4.9 การกระจายตำแหน่งของอนุภาคที่ดีที่สุดตามแบบสมการที่ (4.3)

จากรูปที่ 4.9 เมื่อของทุกกลุ่มติดในจุดสัมผัสที่จุดเดียว (R1) แล้วทำการเปลี่ยนตำแหน่ง GBEST ของกลุ่มตามสมการที่ (4.3) ไปยัง (R2) ที่ตำแหน่งใหม่ที่แตกต่างกันตามเส้นทับ โดยเลือกปรับ 30 เปอร์เซ็นต์ของของมิติทั้งหมดของ GBEST ไปยังตำแหน่งที่แย่กว่าสุดท้ายแล้ว GBEST จะเคลื่อนกลับมายังตำแหน่งเดิมแล้วติดในจุดสัมผัส ณ (R1) เหมือนเดิมตามเส้นปะ



รูปที่ 4.10 การกระจายตำแหน่งของอนุภาคตามสมการที่ (4.2) และ (4.3)

จากรูปที่ 4.10 จากปัญหาที่พบในรูปที่ 4.9 หลังจากขั้นตอนกำหนดค่าใหม่ (Reset) ให้กับ PBEST และ GBEST ให้ทำการเปลี่ยนตำแหน่งอนุภาคทุกตัวตามสมการที่ (4.2) ไปยัง (R2) ดังรูป 4.10 (ซ้าย) ที่อนุภาคเปลี่ยนตำแหน่งไปยังตำแหน่งใหม่ที่แตกต่างกันตามเส้นปะ (R2) โดยเลือกปรับ 30 เปอร์เซ็นต์ของมิติทั้งหมด หลังจากนั้นเข้าสู่กระบวนการประเมินค่าตำแหน่งของแต่ละอนุภาค (Evaluate) เพื่อคัดเลือก GBEST ใหม่ แล้วทำการเปลี่ยนตำแหน่ง GBEST ใหม่ของกลุ่มตามสมการที่ (4.3) ไปยัง เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(R2) ดังรูปที่ 4.10 (ขวา) ที่ตำแหน่งใหม่ที่ต่างกันตามเส้นปะ โดยเลือกปรับ 30 เปอร์เซ็นต์ของของมิติทั้งหมดของ GBEST เพื่อดึงให้อนุภาคตัวอื่นในกลุ่มหลุดออกจากการติดในจุดสัมพัทธ์เดิม

#### 4.5 วิธีการค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย

วิธีที่น าสอนจึงใช้การแบ่ง PSO ออกเป็นหลายกลุ่มแต่ละกลุ่มจะมี GBEST ของตัวเองและใช้สมการเคลื่อนที่แบบ PSO ปกติ ในกรณีที่กลุ่มใดกลุ่มหนึ่งติดในจุดสุดสัมพัทธ์จะใช้การปรับตำแหน่งของอนุภาค และใช้ GBEST ของกลุ่มอื่นเพื่อดึงให้อนุภาคของกลุ่มนี้หลุดออกจากการติดในจุดสุดสัมพัทธ์ แต่ในกรณีที่ GBEST ของทุกกลุ่มติดในจุดสุดสัมพัทธ์เดียวกันจะทำการปรับตำแหน่งของอนุภาคแต่ละกลุ่มและเลือกอนุภาคที่ดีที่สุดในแต่ละกลุ่มมาทำการปรับตำแหน่งอีกครั้งหนึ่งและใช้เป็นค่า GBEST ที่ใช้ในการดึงแต่ละกลุ่มให้หลุดจากการติดในจุดสุดสัมพัทธ์ วิธีนี้แสดงในรหัสจำลองแสดงการทำงานของ MPSOSM ในรูปที่ 4.11 และแผนภาพการทำงานของวิธี MPSOSM ในรูปที่ 4.12 อธิบายได้ว่า TR คือ จำนวนรอบของกลุ่มอนุภาคแต่ละติดอยู่ในจุดสัมพัทธ์ TRR คือ จำนวนรอบของอนุภาคทุกกลุ่มติดอยู่ในจุดสัมพัทธ์เดียวกัน และ  $g$  คือ กลุ่มของอนุภาคตั้งแต่ 1 ถึง  $i$  โดยที่  $i = \{1,2,3,\dots,n\}$

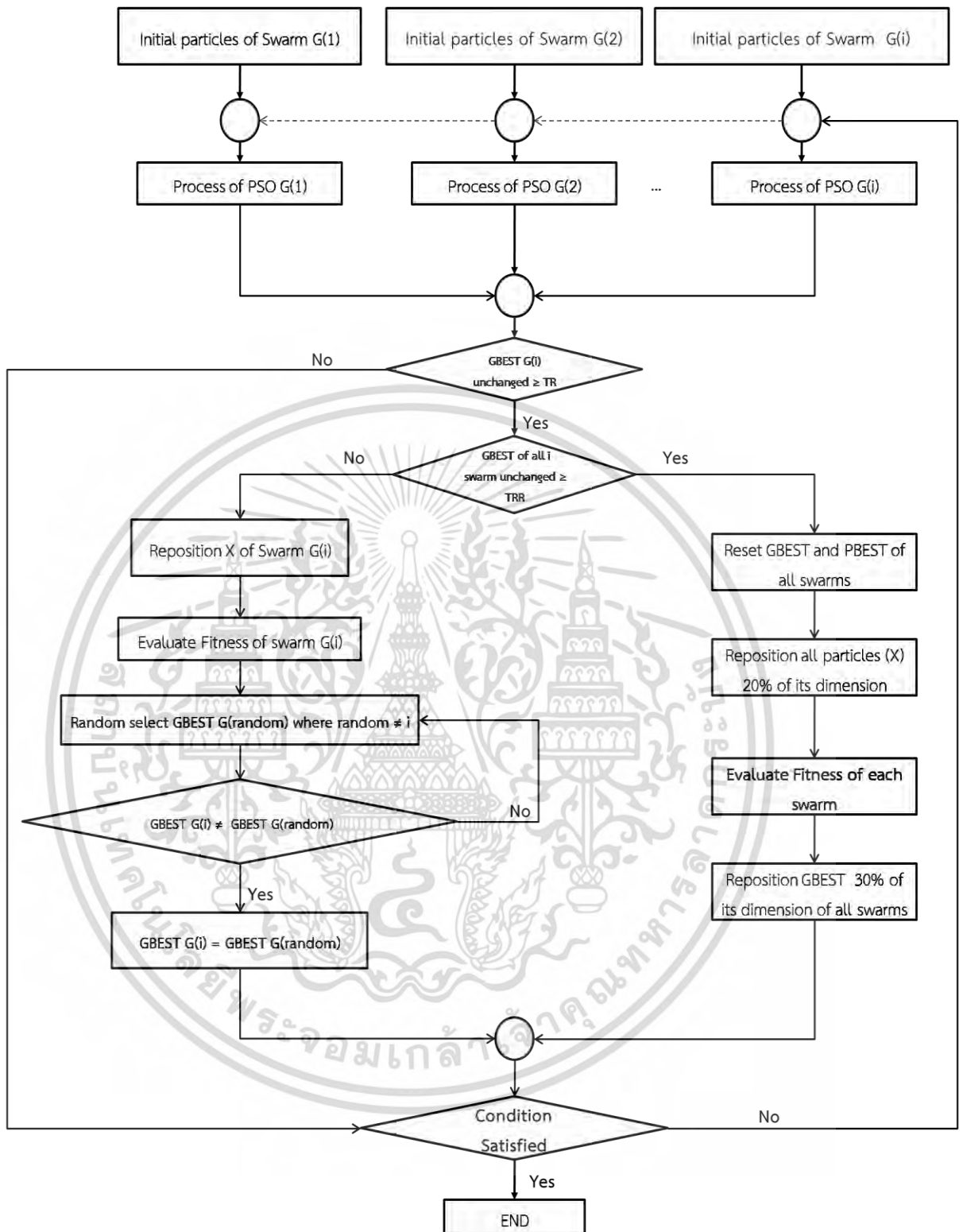
```

1. Initial particles position; evaluate fitness and Update PBEST, GBEST of all particles
   of each swarm
2. While (termination condition ≠ true) do
3.     Update particle position, PBEST, GBEST by eq. (2.1) and (2.2), of all particles
4.     If GBEST of swarm g unchanged for consecutive amount of time (TR)
5.         If GBEST of all swarm unchanged for consecutive amount of time (TRR)
6.             Reset GBEST and PBEST of all particles of all swarms
7.             For to dimensions
8.                 Randomly all X in 20% of all dimensions by eq. (4.2) for all particles
9.                 Randomly all GBEST in 30% of all dimensions by eq. (4.3)
10.            End For
11.        End If
12.    Else
13.        Reset PBEST of swarm g
14.        Randomly X of swarm g in 15% of all dimensions by eq. (4.2)
15.        If Fitness of swarm g is equal Fitness random from other swarm
16.            Random other swarm, Fitness value is not equal
17.            Change GBEST of swarm g with those other swarm
18.            Change GBEST position of swarm g with other swarm
19.        End If
20.    Else If Fitness of swarm g is not equal Fitness random of other swarm
21.        Change GBEST of swarm g with other swarm
22.        Change GBEST position of swarm g with other swarm
23.    End Else If
24. End Else
25. End If
26. End While

```

รูปที่ 4.11 รหัสงานแสดงการทำงานของ MPSOSM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 แผนภาพการทางานของวิธี MPSOSM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปผังงานที่ 4.12 อธิบายการทำงานของ MPSOSM โดยเริ่มจากการแบ่งกลุ่มอนุภาคของ PSO ออกเป็นกลุ่มย่อยตามจำนวน  $i$  กลุ่ม มีจำนวนอนุภาคภายในกลุ่มเท่ากัน และอนุภาคภายในกลุ่มจะเคลื่อนที่เข้าหาอนุภาคที่ดีที่สุด ( $GBEST_i$ ) ของกลุ่มตัวเอง ตามวิธีของ PSO จนกระทั่งติดในจุดสุดสัมพัทธ์ หลังจากนั้นให้ตรวจสอบว่ากลุ่มที่ติดในจุดสุดสัมพัทธ์ ติดในจุดสุดสัมพัทธ์เดียวกันทั้งหมดทุกกลุ่มหรือไม่ ถ้าไม่ให้ทำการปรับปรุงตำแหน่งของอนุภาค ( $x_{id}$ ) ของกลุ่ม  $G(i)$  ตามสมการที่ (4.2) เป็น 20 เปอร์เซ็นต์ของมิติทั้งหมด เข้าสู่กระบวนการประเมินค่าตำแหน่งของแต่ละอนุภาค (Evaluate) อนุภาคของกลุ่ม  $G(i)$  ที่ติดในจุดสัมพัทธ์ ทาการเปลี่ยนตำแหน่ง GBEST กับ GBEST ของกลุ่มอื่นที่ได้จากการสุ่ม ถ้าใช้ให้ทำการกำหนดค่าใหม่ (Reset) ให้กับ PBEST และ GBEST ของทุกกลุ่ม หลังจากนั้นให้ทำการปรับปรุงตำแหน่งของอนุภาค ( $x$ ) ทั้งหมดด้วยสมการ (4.2) เข้าสู่กระบวนการประเมินค่าตำแหน่งของแต่ละอนุภาค (Evaluate) เป็นกระบวนการประเมินค่าความเหมาะสมของแต่ละอนุภาค ณ ตำแหน่งปัจจุบัน เพื่อใช้ในการปรับค่า PBEST และ GBEST ของอนุภาคทั้งหมด ท การปรับปรุง GBEST ของแต่ละกลุ่มที่ได้จากอนุภาคที่โดยปรับปรุงแล้วด้วยสมการ (4.3) ทั้งหมด 30 เปอร์เซ็นต์ของมิติทั้งหมดของ GBESTกระบวนการเหล่านี้ทำให้คำตอบที่ได้ดีขึ้นและช่วยเพิ่มประสิทธิภาพของ PSO เพราะถึงแม้ว่าการกระจายตำแหน่งของอนุภาคที่ดีที่สุด (GBEST) ตามสมการที่ (4.3) อาจทำให้ค่าความเหมาะสมของ GBEST แ่ลงแต่กระบวนการเหล่านี้กับเพิ่มความหลากหลายและลดปัญหาการติดในจุดสุดสัมพัทธ์ของอนุภาคได้ ทำให้อนุภาคสามารถเคลื่อนต่อไปจนพบคำตอบที่ดีขึ้น

## บทที่ 5

### ผลการทดลอง

บทนี้กล่าวถึงการเปรียบเทียบผลลัพธ์ของการค้นหาจุดต่ำสุดของฟังก์ชันมาตรฐาน จากวิธีต่างๆ ด้านล่าง

1. วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization, PSO) [1, 2]
2. วิธีการค้นหาอย่างรวดเร็วแบบหลากหลายกลุ่มโดยการกลายพันธุ์และการผสมพันธุ์ (Fast Multi-swarm Optimization with Cauchy Mutation and Crossover operation, FMPSO) [9]
3. วิธีการคัดเลือกการผสมพันธุ์โดยใช้ความเหมาะสมกับแบบหลากหลายกลุ่ม (Selective Crossover base on Fitness in Multi-Swarm Optimization, SCMPSO)[10]
4. วิธีการดัดแปลงการค้นหาค่าเหมาะสมแบบอนุภาคหลากหลายกลุ่ม ด้วยการแลกเปลี่ยนตำแหน่งของอนุภาคที่ดีที่สุดและกระจายอนุภาค (A Modified Multi-Swarm Optimization with Interchange GBEST and Particle Redistribution, MPSOIR) [15] โดยเมื่อกลุ่มใดกลุ่มหนึ่งติดในจุดสัมพัทธ์จะทำการปรับตำแหน่งของอนุภาคในกลุ่มนั้นทุกอนุภาคด้วยสมการที่ (4.1) และสุ่มปรับตำแหน่งใน 70 เปอร์เซ็นต์ของมิติทั้งหมดของแต่ละอนุภาคและค่า PBEST ของแต่ละอนุภาคก็คือตำแหน่งใหม่หลังการปรับนอกจากนั้นยังทำการเปลี่ยนค่า GBEST ของกลุ่มที่ ติดในจุดสัมพัทธ์กับกลุ่มอื่นที่ กำหนดไว้ก่อนหน้า
5. วิธีนี้คือวิธีในข้อที่ 4. แต่ตัดส่วนที่เป็นการกระจายอนุภาคออกเมื่ออนุภาคกลุ่มใดกลุ่มหนึ่งติดในจุดสุดสัมพัทธ์จะทำการเปลี่ยนค่า GBEST ของกลุ่มนั้นให้เท่ากับกลุ่มอื่นโดยไม่มีการปรับตำแหน่งของอนุภาคของกลุ่มนั้นแต่มีการกำหนดค่า PBEST ใหม่ให้มีค่ามากมากเพื่ ื่อให้ เมื่อมีการเปลี่ยนตำแหน่งตามสมการ PSO ตำแหน่งใหม่ที่ได้จะเป็นตำแหน่ง PBESTโดยใช้ชื่อย่อเรียกวิธีนี้ว่า MPSOIR เพื่อเปรียบเทียบประสิทธิภาพของการปรับตำแหน่งอนุภาคร่วมกับการเปลี่ยนค่า GBEST
6. วิธีนี้คือวิธีที่ 4. แต่ไม่มีการเปลี่ยนค่า GBEST เมื่อติดในจุดสุดสัมพัทธ์ เมื่ออนุภาคของกลุ่มใดติดในจุดสัมพัทธ์จะทำการกระจายตำแหน่งอนุภาคด้วยสมการที่ (4.1) และสุ่มปรับ 70 เปอร์เซ็นต์ของจำนวนมิติทั้งหมดของแต่ละอนุภาค ใช้ชื่อย่อว่า MPSOR1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. วิธีนี้คือวิธีที่ 4. แต่ไม่มีการเปลี่ยนค่าGBEST เมื่อติดในจุดสุดสัมพัทธ์ เมื่ออนุภาคของกลุ่มใดติดในจุดสัมพัทธ์จะทำการกระจายตำแหน่งอนุภาคด้วยสมการที่ (4.2) และสุ่มปรับ 20 เปอร์เซ็นต์ของจำนวนมิติทั้งหมดของแต่ละอนุภาค ใช้ชื่อว่า MPSOR2
8. วิธีการค้นหาค่าเหมาะสมแบบอนุภาคหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุด และการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย (A Multi-Swarm Optimization with Changing of Swarm Best Position and Slightly Mutation of Particle Position, MPSOSM) [16](ซึ่งเป็นวิธีที่วิทยานิพนธ์ฉบับนี้ได้นำเสนอ) วิธีนี้ดัดแปลงมาจาก วิธีที่ 4. โดยมีความเชื่อว่าตำแหน่งที่อนุภาคติดจุดสัมพัทธ์นั้นมีค่าใกล้เคียงกับจุดสุดสัมพัทธ์แล้ว คือมีเพียงบางมิติของอนุภาคที่ มีค่าไม่ถูกต้อง ดังนั้นในการปรับตำแหน่งของอนุภาคตามสมการที่ (4.2) และจะสุ่มปรับเพียง 20 เปอร์เซ็นต์จากตำแหน่งเดิม และทำการตั้งค่า GBEST ใหม่ให้เท่ากับค่า GBEST ของกลุ่มอื่นที่สุ่มมา ในกรณีที่ทุกกลุ่มมีการติดในจุดสุดสัมพัทธ์ที่เดียวกันจะทำการกระจายอนุภาคของแต่ละกลุ่มจากตำแหน่งเดิม จากนั้นจะใช้ค่า GBEST ที่ได้มาจากการกระจายตำแหน่งของแต่ละกลุ่มและทำการสุ่มปรับค่า GBEST ของกลุ่มนั้นๆไป 30 เปอร์เซ็นต์ของจำนวนมิติทั้งหมดตามสมการที่ (4.3)

การทดลองนี้ได้ทดลองกับปัญหาฟังก์ชันมาตรฐานทั้งหมด 27ฟังก์ชัน [46 - 56] จากตาราง ก.1โดยที่17ฟังก์ชันแรกเป็นปัญหาแบบมัลติโมดอลและสำหรับ 10ฟังก์ชันหลังเป็นปัญหาแบบยูนิโมดอลจุดประสงค์ของการทดลองชุดนี้ เพื่อวัดประสิทธิภาพการค้นหาของวิธีที่นำเสนอMPSOSM เปรียบเทียบกับ PSO, FMPSO, SCMPSO, MPSOIR, MPSOI, MPSOR1 และ MPSOR2

การทดลองสามารถแบ่งออกเป็น 2มีรายละเอียดดังนี้

- การทดลองเปรียบเทียบวิธีการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาคของวิธีที่ ได้นำเสนอ
- การทดลองเปรียบเทียบประสิทธิภาพของการค้นหาด้วยวิธีที่ นำเสนอฮับงานวิจัยอื่น

### ตารางที่ 5.1 ค่าพารามิเตอร์ที่ใช้ทดลองกับฟังก์ชันคณิตศาสตร์

สำหรับค่าพารามิเตอร์ต่างๆ ในงานวิจัยนี้แบ่งได้เป็น 2 ส่วนคือค่าพารามิเตอร์ที่ใช้ในส่วน ของงานวิจัยที่ ได้นำเสนอ MPSOSM และค่าพารามิเตอร์ของงานวิจัยที่ได้นำมาเปรียบเทียบ

พารามิเตอร์	ค่า
จำนวนอนุภาคที่ใช้ในการทดลองของขั้นตอนPSO	90อนุภาค

เอกสารนี้เป็นเอกสารทบทวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์ การนำเอกสารไปใช้ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พารามิเตอร์	ค่า
จำนวนอนุภาคที่ใช้ในการทดลองของขั้นตอนMPSO ต่อกลุ่ม	30อนุภาค
จำนวนกลุ่มอนุภาคที่ใช้ของขั้นตอน MSPO	3 กลุ่ม
จำนวนครั้งมากที่สุดของการคำนวณหาค่าความเหมาะสม (Evaluation Call)	10,000,000 ครั้ง
ขอบเขตการค้นหาของแต่ละฟังก์ชัน	ตามตารางในภาคผนวก ก.1
จำนวนมิติของแต่ละฟังก์ชัน	ตามตารางในภาคผนวก ก.1
ความเร็วมากที่สุด (Max velocity)	ตามตารางในภาคผนวก ก.1
ความเร็วน้อยสุด (Min velocity)	ตามตารางในภาคผนวก ก.1
จำนวนครั้งในการทดลอง	50 ครั้งต่อฟังก์ชัน
ค่าความเฉื่อย (Weight, $\omega$ )	0.72984 [4]
ค่าคงที่ ขส PBEST (Constant number PBEST, $C_1$ )	1.49618 [4]
ค่าคงที่ ขส GBEST (Constant number GBEST, $C_2$ )	1.49618 [4]
ความน่าจะเป็นในการกระจายตำแหน่งของอนุภาค (X) ของขั้นตอน MPSOR1 (Probability of Reposition Particle)	0.70
ความน่าจะเป็นในการกระจายตำแหน่งของอนุภาค (X) ของขั้นตอน MPSOR2 (Probability of Reposition Particle)	0.20
รอบของการกระจายอนุภาค(Threshold of Reposition)	100
ค่าคงที่ $\alpha$ เป็นค่าสุ่มระหว่าง (0,1) ของขั้นตอน FMPSO	0.60[33]

โปรแกรมที่ใช้ในการทดลอง คือ Microsoft Visual C++ 2005 พัฒนาโปรแกรมโดยใช้ภาษา C++ ทำการทดลองบนเครื่องคอมพิวเตอร์ส่วนบุคคล มีหน่วยประมวลผลกลางยี่ห้อ Intel รุ่น Core i7 3770 ความเร็ว 2.4 GHz หน่วยความจำหลักขนาด 8 GB ระบบปฏิบัติการ Windows Seven Professional

ค่าที่ใช้เปรียบเทียบประสิทธิภาพของวิธีสำหรับการทดลอง คือค่าเฉลี่ย ของค่าความเหมาะสมที่ดีที่สุด (The Mean Best Fitness Value หรือMBF) คือ ตัวชี้วัดประสิทธิภาพในการค้นหาค่าตอบของวิธีที่ได้นำเสนอและวิธีที่นำมาเปรียบเทียบในกรณีการทดลองกับฟังก์ชันมาตรฐานจากตาราง ก.1จะมีจุดที่ดีที่สุดหรือ  $f(x) = 0$  ทุกฟังก์ชันถ้าหากวิธีใดที่สามารถหาค่าตอบใกล้เคียงกับศูนย์มากที่สุดวิธีนั้นก็จะมีประสิทธิภาพในการแก้ปัญหาฟังก์ชันมาตรฐานมากเท่านั้นและค่าเฉลี่ยของค่าการเดาค่าตอบที่ดีที่สุด (The Mean Best Evaluation Call หรือ MEC) คือ ค่าเฉลี่ยของค่าการเดาค่าตอบที่ดีที่สุดในรอบสุดท้ายของการค้นหาจากการทดลองทั้งหมด

## 5.1 การทดลองเปรียบเทียบวิธีการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาคของวิธีที่ได้นำเสนอ

ในหัวข้อนี้จะพิจารณาถึงความสามารถในการค้นหาค่าตอบของฟังก์ชันมาตรฐานด้วย MPSOR1, MPSOR2, MPSOI, MPSOIR และ MPSOSM เพื่อวิเคราะห์ขั้นตอนของวิธีการค้นหาที่เหมาะสมแบบอนุภาคหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย ได้ทดสอบกับฟังก์ชันมาตรฐานในตารางก.1 โดยใช้พารามิเตอร์ตามตาราง 5.1

โดยตารางที่ 5.2 แสดงผลของค่าเฉลี่ยค่าความเหมาะสมที่ดีที่สุด (MBF) โดยกำหนดจำนวนการเดาค่าตอบมากที่สุดเท่ากันทุกวิธีและทุกฟังก์ชันการทดลอง

จากตารางที่ 5.2 ได้ทำการแบ่งการทดลองออกเป็นส่วนของวิธีการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อยด้วยสมการที่ (4.1) กับ 70 เปอร์เซ็นต์ของมิติทั้งหมดของอนุภาคใช้ชื่อว่า MPSOR1 และ (4.2) กับ 20 เปอร์เซ็นต์ของมิติทั้งหมดของอนุภาคใช้ชื่อว่า MPSOR2 รวมถึงได้ทดลองวิธีการเปลี่ยนตำแหน่งของอนุภาคที่ดีที่สุดใช้ชื่อว่า MPSOI ร่วมกับการกระจายตำแหน่งของอนุภาคด้วยสมการที่ (4.1) กับ 70 เปอร์เซ็นต์ของมิติทั้งหมดของอนุภาคใช้ชื่อว่า MPSOIR แต่จากกระบวนการทั้งหมดเหล่านี้รวมกันกลับค้นพบว่าเมื่ออนุภาคมีการเปลี่ยนตำแหน่งในระยะเวลาหนึ่งอนุภาคของแต่ละกลุ่มจะเคลื่อนที่เข้าหากันจนติดในจุดสุดสัมพัทธ์ ทำให้มีการเพิ่มวิธีการเปลี่ยนตำแหน่งที่ดีที่สุดของอนุภาคทุกกลุ่มรวมกับการเปลี่ยนตำแหน่งของอนุภาคทุกตัวเพียงเล็กน้อยสมการที่ (4.2) กับ 20 เปอร์เซ็นต์ของมิติทั้งหมดของอนุภาค ใช้ชื่อว่า MPSOSM

จากตารางที่ 5.2 MPSOR2 ซึ่งเปลี่ยนตำแหน่งอนุภาคหลังจากการติดในจุดสัมพัทธ์ในระยะเวลาใกล้เคียงตามสมการที่ (4.2) มีผลการทดลองที่ถึงจุดสูงสุดสมบูรณ์มากกว่า MPSOR1 มีเพียงฟังก์ชัน DIXON-PRICE เท่านั้นที่มีค่าแย่กว่า ส่วน MPSOI มีผลการทดลองโดยรวมแย่กว่าทั้ง MPSOR1 และ MPSOR2 ยกเว้นฟังก์ชัน DIXON-PRICE ที่มีผลเท่ากับ MPSOR1 แต่เมื่อนำเทคนิคการปรับตำแหน่งของอนุภาค MPSOR1 มารวมกับการปรับตำแหน่ง GBEST ของ MPSOI ทำให้ MPSOIR ได้ผลลัพธ์ที่ดีกว่าวิธีการที่ปรับค่าอนุภาคหรือปรับค่า GBEST อย่างเดียว

แต่จากกระบวนการ MPSOIR พบว่าเมื่ออนุภาคที่ดีที่สุดเปลี่ยนตำแหน่งกับกลุ่มข้างเคียงโดยการสุ่มตาม MPSOI และทำการเปลี่ยนตำแหน่งของอนุภาคภายในกลุ่มเพื่อให้อนุภาคมีโอกาสในการค้นหาแบบระยะใกล้ (Local Search) ตาม MPSOIR กลับพบว่าเมื่ออนุภาคมีการค้นหาในระยะเวลาหนึ่งอนุภาคของทุกกลุ่มจะเคลื่อนที่เข้าหากันจนติดในจุดสัมพัทธ์ที่จุดเดียวกัน ดังนั้นจึงได้ทำการเพิ่มวิธีการเปลี่ยนตำแหน่งของอนุภาคตาม MPSOR2 และการเปลี่ยนตำแหน่งของอนุภาค GBEST จึงทำให้ MPSOSM ได้ผลลัพธ์ที่ดีที่สุด

ตารางที่ 5.2 ผลการทดลองการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาค

ALGORITHM	MPSOR1	MPSOR2	MPSOI	MPSOIR	MPSOSM
FUNCTION	MBF	MBF	MBF	MBF	MBF
ACKLEY	6.89E-15	6.47E-15	0.88707	3.55E-15	0
GRIEWANK	0.000739358	0	0.00175646	0	0
RASTRIGIN	14.4468	0.23879	9.95223	0	0
ROSENBROCK	6.29E-30	1.48E-31	4.17E-29	2.86E-28	3.94E-29
SCHWEFEL	1397.89	790.69	2294.811	2224.07	3789.74
COSINE MIXTURE	0.00591137	0	0	0	0
EXPONENTIAL	3.11E-17	6.66E-17	5.34E-17	0	0
LEVY	0.0272594	0.0053717	0.0175438	1.14E-06	0
MICHALEWICZ	0	0	0	0	0
DIXON-PRICE	0	3.77E-17	0	0	0
STEP	0	0	750	0	0
SCHAFFER	0.000582955	0	0.00136023	0	0
HOLDER	0	0	0	0	0
BEALE	3.48E-29	0	1.31E-30	0	0
SHUBERT	0	0	0	0	0
GOLDSTEIN-PRICE	0	0	0	0	0
SIX-HUMP CAMEL	0	0	0	0	0
SPHERE	8.76E-205	1.65E-205	1.91E-40	1.53E-260	0
PARALLEL	1.38E-201	3.27E-202	6.84E-39	4.64E-246	2.9643e-323
ROTATED	1.65E-208	2.83E-211	1.62E-38	1.43E-252	1.4822e-323
CIGAR	1.65E-199	4.39E-200	2.79E-34	3.42E-244	3.4584e-323
BROWN	1.71E-203	7.89E-206	5.34E-40	2.82E-253	6.9169e-323
MULTIMOD	2.84E-76	1.77E-76	4.30E-15	1.39E-99	6.93E-272
ZAKHAROV	1.83E-40	1.32E-40	1.31E-05	1.29E-17	1.39E-99
TRID	8.11E-12	5.64E-13	3.62E-12	4.55E-12	0
EASOM	0	0	1.21E-06	1.69E-06	0
EGGHOLDER	24.9856	16.0088	27.0648	0	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5.2 การทดลองเปรียบเทียบประสิทธิภาพของการค้นหาด้วยวิธีที่นำเสนอกับงานวิจัยอื่น

จากตารางที่ 5.3 เมื่อนำ MPSOSM มาเปรียบเทียบกับวิธีอื่นๆ ที่มีลักษณะการทำงานที่ใกล้เคียงกันพบว่า วิธี MPSOSM มีประสิทธิภาพการค้นหาค่าตอบที่ดีกว่าวิธีอื่นๆ คือได้จุดต่ำสุดที่ต่ำกว่าและพบจำนวนจุดสูงสุดสมบูรณ์ในหลายฟังก์ชันมากกว่า โดยจากผลการทดลองเปรียบเทียบกับ FMP SO และ SCMP SO พบว่า MPSOSM ได้ผลลัพธ์การค้นหาค่าตอบที่ดีที่สุด และพบค่าตอบทุกฟังก์ชัน นอกจากนี้การทดลองนี้แสดงให้เห็นค่าเฉลี่ยของการเดาค่าตอบที่ดีที่สุด (MEC) จะเห็นว่า มีค่าที่ต่ำสุดเมื่อเทียบกับวิธีอื่นๆ (ในบางวิธีที่เช่น PSO มีค่าที่ต่ำกว่าแต่ค่าตอบไม่ถึงจุดสูงสุดสมบูรณ์) เป็นการยืนยันว่าวิธีที่ นำเสนอนั้นสามารถปรับปรุงประสิทธิภาพและการค้นหาค่าตอบของ PSO ได้ดี

ตารางที่ 5.3 ผลการทดลองวิธีที่นำเสนอกับงานวิจัยอื่น

ALGORITHM	PSO		FMP SO		SCMP SO		MPSOSM	
	MEC	MBF	MEC	MBF	MEC	MBF	MEC	MBF
ACKLEY	715,981	4.09188	5,564,625	0.88707	4,605,609	1.42E-14	5,272,422	0
GRIEWANK	318,707	0.0972755	3,401,397	1.55E-15	6,531,819	0	686,922	0
RASTRIGIN	352,752	140.431	9,504,741	960.912	9,292,905	450.764	4,222,824	0
ROSENBROCK	720,415	1.25E-26	9,965,025	1.51E-26	780,714	7.32E-28	9,999,966	3.94E-29
SCHWEFEL	336,475	8718.08	9,492,141	27688.5	9,891,366	6162.96	9,999,281	3789.74
COSINE MIXTURE	337,798	3.30923	1,915,473	2.95845	8,700,762	0	1,295,051	0
EXPONENTIAL	317,797	2.04E-15	4,468,797	2.22E-16	478,197	2.22E-16	988,408	0
LEVY	397,759	16.8214	9,725,337	1.14E-06	4,068,258	89.4557	8,917,597	0
MICHALEWICZ	627,646	0.285482	2,910	0	4,956	0	172,429	0
DIXON-PRICE	8,074	0	32,307	0	10,791	0	11,587	0
STEP	754,208	187	947,847	634	8,357,346	0	175,541	0
SCHAFFER	18,075	0	33,651	0	78,792	0	66,789	0
HOLDER	9,663	0	366	0	267	0	6,634	0
BEALE	31,980	0	26,436	0	28,554	0	32,416	0
SHUBERT	5,239	0	24,279	0	2,228,700	0	4,689	0
GOLDSTEIN-PRICE	10,570	0	7,230	0	8,460	0	12,008	0
SIX-HUMP CAMEL	2,224	0	750	0	1,068	0	2,497	0
SPHERE	1,000,013	4.79244e-322	9,999,819	1.94E-109	7,863,309	1.57E-11	10,000,002	0
PARALLEL	1,000,058	5.68175e-322	10,000,227	1.67E-105	6,371,319	1.38E-09	10,000,076	2.9643e-323
ROTATED	999,877	1.77864e-322	9,999,945	8.52E-109	8,418,546	8.54E-07	10,000,002	1.4822e-323
CIGAR	1,000,067	7.48776e-319	9,999,594	5.11E-103	5,993,316	6.90E-05	10,000,077	3.4584e-323
BROWN	999,985	1.90709e-321	9,999,732	9.89E-108	5,102,970	1.56E-11	10,000,073	6.9169e-323
MULTIMOD	142,156	100.097	9,641,556	1.00E+06	10,000,209	4.00E+09	10,000,061	6.93E-272

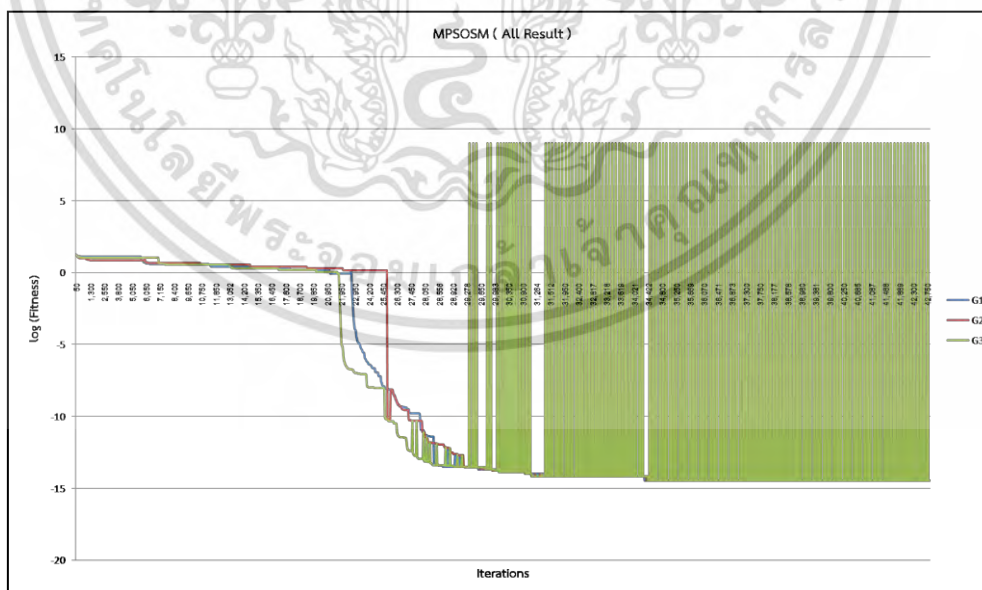
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ALGORITHM	PSO		FMPSO		SCMPSO		MPSOSM	
FUNCTION	MEC	MBF	MEC	MBF	MEC	MBF	MEC	MBF
ZAKHAROV	999,696	13.1127	9,950,103	185.315	9,913,647	500.464	9,999,811	1.39E-99
TRID	86,891	0	887,118	4.82E-12	8,920,044	1.82E-12	163,072	0
EASOM	18,094	0	20,208	1.69E-06	26,646	1.69E-06	17,965	0
EGGHOLDER	28,388	0	25,928	43.4722	20,280	2.72178	10,000,079	0

ตารางที่ 5.4 ผลสรุปการทำงานสุดท้ายของฟังก์ชัน ACKLEY ในหนึ่งรอบการทำงาน

ฟังก์ชัน	มิติ	รอบสุดท้าย ในการทำงาน	รอบการเดา คำตอบสุดท้าย	รอบการเดา คำตอบที่ดีที่สุด	จำนวนทุก กลุ่มติดในจุด สัมพันธ์เดียว กัน	จำนวนรอบแต่ ละกลุ่มติดใน จุดสัมพันธ์	คำตอบ
ACKLEY	100	42,879	3,896,340	3,896,261	131	59	0

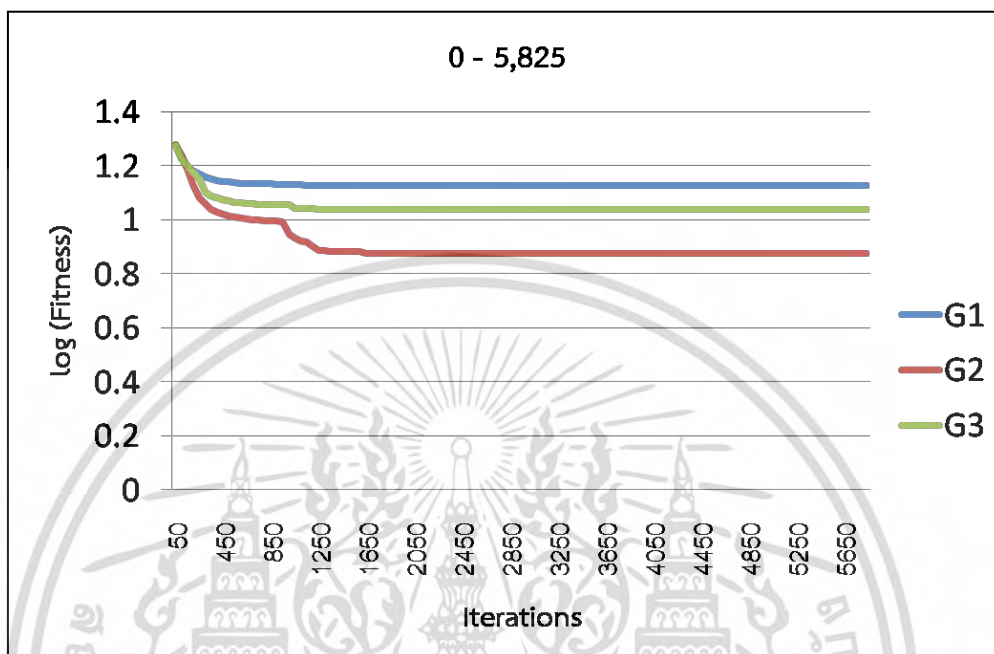
จากตารางที่ 5.4 แสดงผลสรุปการทำงานจนจบการทำงานของฟังก์ชัน ACKLEY โดยมีการเดาคำตอบทั้งหมด 3,896,340 ครั้ง และเจอคำตอบที่ดีที่สุดเมื่อมีการเดาคำตอบไป 3,896,261 ครั้ง โดยใช้วิธี MPSOSM โดยผลการทำงานทั้งหมดตั้งแต่รอบแรกจนถึงรอบสุดท้ายถูกนำมาแสดงในกราฟที่ 5.1 ถึง 5.8 เส้นกราฟแสดงค่าความเหมาะสมของอนุภาคที่ดีที่สุดใน Log Scale(log10) ใช้จำนวนรอบการเดาคำตอบสูงสุด 10,000,000 รอบโดยกำหนดให้หยุดเมื่อ พบคำตอบที่ดีที่สุด



รูปที่ 5.1 LBP ของฟังก์ชัน ACKLEY ที่ ค้นหาคำตอบด้วย MPSOSM ในรอบการทำงานทั้งหมด

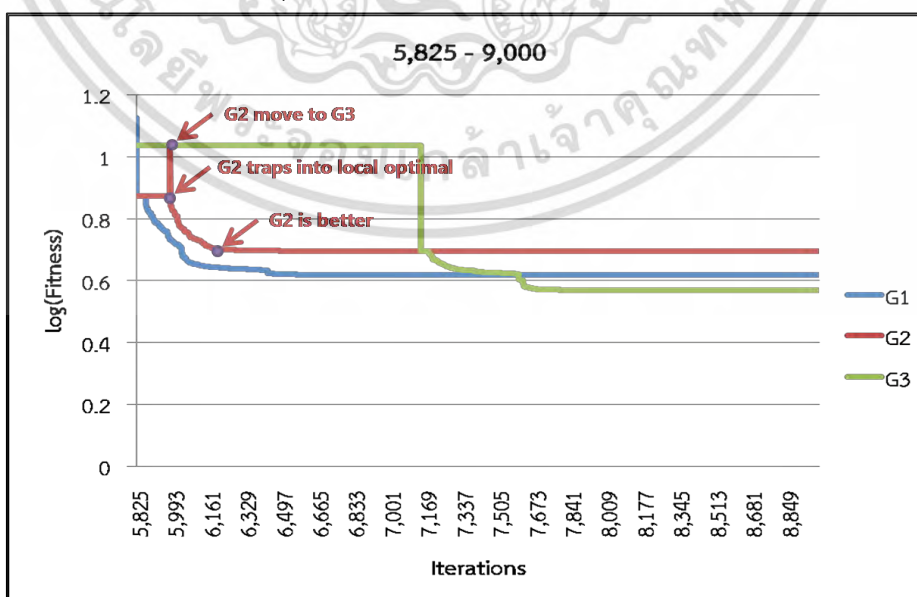
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปภาพที่ 5.1 เป็นกราฟแสดงการทำงานของวิธี MPSOSM ทั้งหมดตั้งแต่ต้นจนจบการทำงานของฟังก์ชัน ACKLEY จนสิ้นสุดการทำงานและได้แบ่งการทำงานออกเป็นช่วงตามรูปภาพที่ 5.2 ถึง 5.8 ตามลำดับ



รูปที่ 5.2 LBP ของฟังก์ชัน ACKLEY ที่ ค้นหาคำตอบด้วย MPSOSM ในช่วง 0-5,825

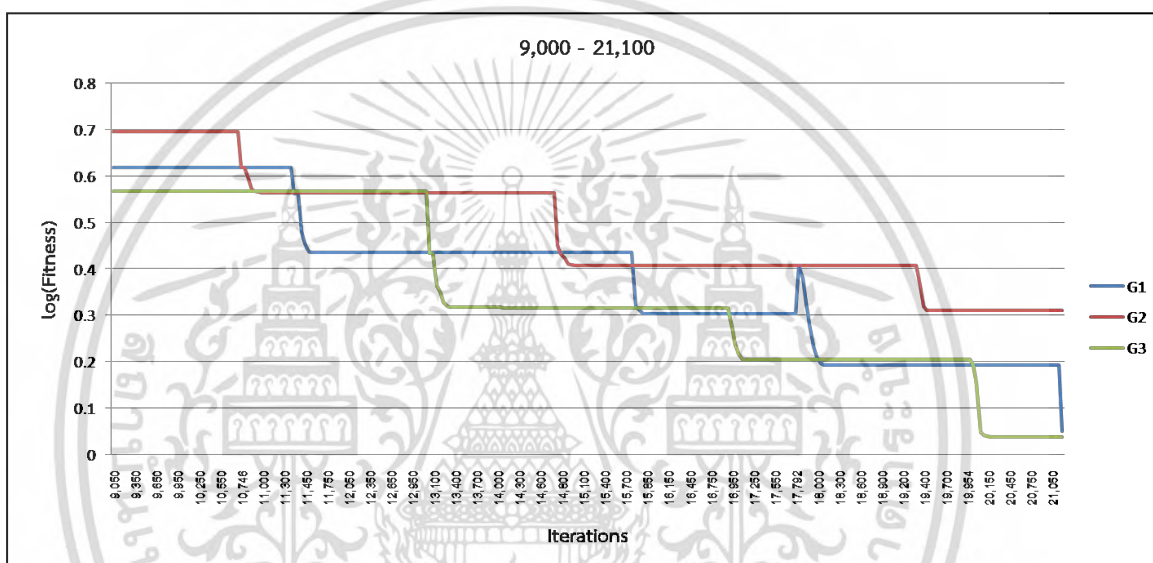
โดยช่วงแรกของกราฟ 5.2 แสดงผลในช่วง 0 – 5,825 เส้นกราฟแต่ละเส้น แทนด้วยคำตอบหรือค่าความเหมาะสมที่ดีที่สุด (Best Fitness) ของแต่ละกลุ่มอนุภาคจะเห็นว่าคำตอบที่ได้ของแต่ละกลุ่มจะลดลงแล้วคงที่ หรือติดในจุดสัมพัทธ์เหมือนกับ PSO



รูปที่ 5.3 LBP ของฟังก์ชัน ACKLEY ที่ ค้นหาคำตอบด้วย MPSOSM ในรอบที่ 5,825– 9,000

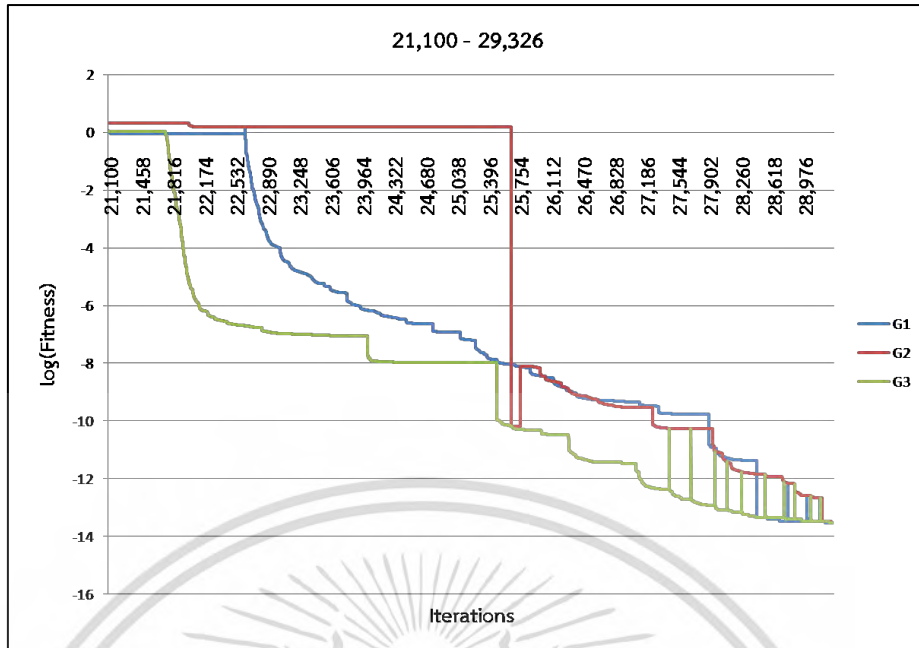
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปกราฟที่ 5.2 แสดงผลในช่วง 5,825 – 9,000 อยู่ในช่วงต่อจากรูปที่ 5.2 จะเห็นว่าเมื่ออนุภาคของกลุ่ม G2 คงที่หรือติดในจุดสัมพันธ์ตามรอบที่กำหนดของวิธี MPSOSM ตามตารางที่ 5.2 (100 รอบ) จะดำเนินการสุมเปลี่ยนตำแหน่งที่ดีที่สุดของกลุ่ม (GBEST) สุมได้กลุ่ม G3 ที่ไม่ได้ติดในจุดสัมพันธ์เดียวกัน จะเห็นได้จากกราฟเคลื่อนที่ไปยังกลุ่ม G3 ที่มีค่าตอบที่แย่กว่า ก่อนที่จะหลุดออกจากการติดในจุดสัมพันธ์หลังจากนั้นจะดำเนินการค้นหาด้วย PSO จึงทำให้เส้นกราฟลดลง ซึ่งระหว่างนี้อาจจะพบค่าที่ดีกว่า GBEST เดิมทำให้ค่าตอบนั้นดีขึ้น สังเกตได้จากกราฟบางช่วงที่หลังจากเปลี่ ษตำแหน่งไปยังจุดที่แย่ลงหลังจากนั้นอนุภาคจะลู่ลงไปมากกว่าเดิม แสดงในช่วง 5,990 - 6,000



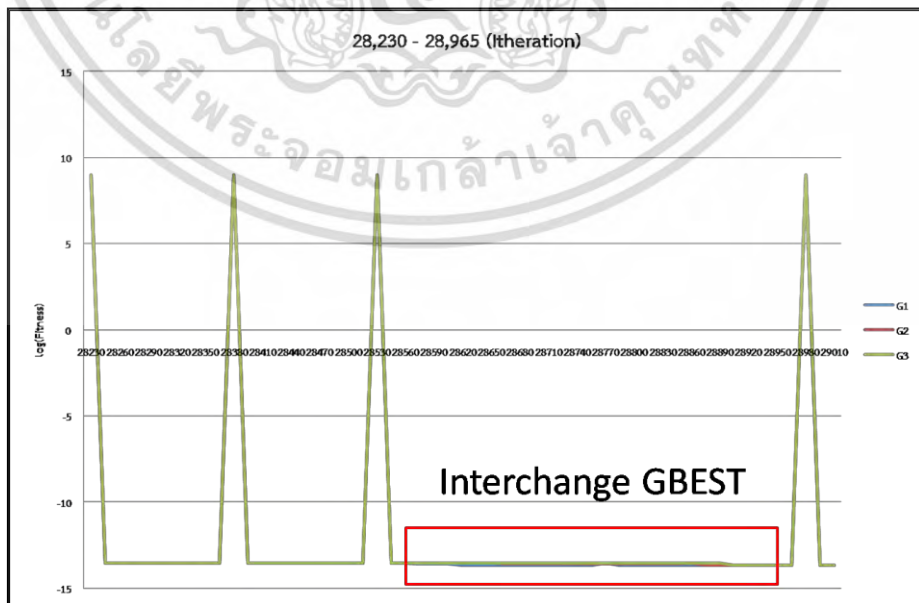
รูปที่ 5.4 LBP ของฟังก์ชัน ACKLEY ที่ ค้นหาตอบด้วย MPSOSM ในช่วง 9,000 - 21,100

จากรูปกราฟที่ 5.3 แสดงค่าในช่วง 9,000 – 21,100 อนุภาคของแต่ละกลุ่มยังคงทำการเปลี่ยนตำแหน่ง GBEST กับกลุ่มข้างเคียงที่ได้จากการสุมจนหลุดออกจากจุดสัมพันธ์และเคลื่อนที่ค้นหาไปยังตำแหน่งที่ดีขึ้น



รูปที่ 5.5 LBP ของฟังก์ชัน ACKLEY ที่ ค้นหาด้วย MPSOSM ในช่วงที่ 21,100 - 29,326

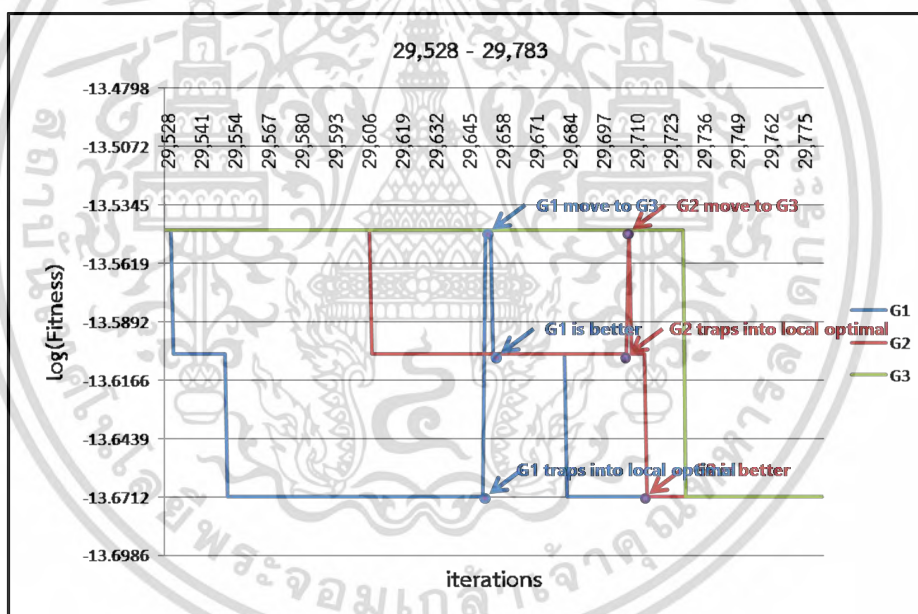
จากรูปภาพที่ 5.4 แสดงการทำงานของ MPSOSM ในช่วง 21,100 - 29,326 อธิบายได้ว่า เมื่ออนุภาคของกลุ่มที่ติดในจุดสุดสัมพัทธ์มีการสุ่มเปลี่ยนตำแหน่งที่ดีที่สุดกับกลุ่มข้างเคียงในระยะเวลาหนึ่งอนุภาคของแต่ละกลุ่ม จะเคลื่อนที่เข้ามาอยู่ใกล้กัน โดยดูได้จากกราฟที่ 5.4 ที่เริ่มต้นนั้นระยะห่างของค่าตอบของแต่ละกลุ่มจะกว้าง แล้วเริ่มแคบลงเมื่อมีการเปลี่ยนตำแหน่ง GBEST ของแต่ละกลุ่มเป็นระยะเวลาหนึ่ง อนุภาคของแต่ละกลุ่มจะเคลื่อนที่เข้าหากันจนทำให้อนุภาคของทุกกลุ่มติดในจุดสุดสัมพัทธ์เดียวกัน



รูปที่ 5.6 LBP ของฟังก์ชัน ACKLEY ที่ ค้นหาด้วย MPSOSM ในรอบที่ 28,230 - 28,965

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปกราฟที่ 5.6 แสดงการทำงานของ MPSOSM ในช่วง 28,230 - 28,965 ดังนั้นจึงได้เพิ่มขึ้นตอนการกระจายตำแหน่งของอนุภาคแต่ละตัว (PBEST) เพื่อให้อนุภาคมีโอกาสเคลื่อนที่ค้นหาในบริเวณเดิมซึ่งอาจมีโอกาสเจอค่าตอบที่ดีขึ้น ได้เร็วกว่าการกระจายตำแหน่งไปบริเวณกว้างแล้วรอให้อนุภาคกลับเข้ามาพร้อมกับการกระจายตำแหน่งของอนุภาคที่ดี (GBEST) เพียงบางมิติ เพื่อให้ค่าตอบแย่งลงกว่าเดิมในรอบถัดไป ให้อนุภาคมีโอกาสหลุดออกจากจุดสุดสัมพัทธ์ ซึ่งระหว่างนี้อาจจะพบค่าที่ดีกว่า GBEST เดิมทำให้ค่าตอบนั้นดีขึ้น ในกรณีที่มีการกระจายตำแหน่งของอนุภาคนั้นควรที่จะมีการกำหนดค่าใหม่ (Reset) ให้กับ PBEST และ GBEST เพื่อให้อนุภาคไม่กลับไปยัง GBEST เดิม ดังกราฟที่ 5.6 จะเห็นได้ว่ากราฟสูงขึ้นแล้วลดลงมาอยู่ที่ตำแหน่งเดิมหรือตำแหน่งที่ดีกว่าทำให้อนุภาคของแต่ละกลุ่มหลุดจากการติดในจุดสัมพัทธ์แล้วเคลื่อนที่แบบ PSO ปกติจนแต่ละกลุ่มที่ติดในจุดสัมพัทธ์ในบริเวณที่ต่างกันทำให้เกิดการเปลี่ยนตำแหน่ง GBEST กับกลุ่มข้างเคียง



รูปที่ 5.7 LBP ของฟังก์ชัน ACKLEY ที่ ค้นหาตอบด้วย MPSOSM ในรอบที่ 29,528 - 29,783

จากรูปกราฟที่ 5.7 แสดงการทำงานของ MPSOSM ในช่วง 29,528 - 29,783 เมื่อทำการเปลี่ยนตำแหน่งให้กับอนุภาค (x) ใน 20 เปอร์เซ็นต์ของมิติทั้งหมดพร้อมกับเปลี่ยนตำแหน่งที่ดีที่สุด (GBEST) ใน 30 เปอร์เซ็นต์ของมิติทั้งหมดทำให้อนุภาคของแต่ละกลุ่มหลุดจากการติดในจุดสัมพัทธ์เดียวกันกลับมาเคลื่อนที่ได้ปกติ แต่ในระยะเวลาหนึ่งเหมือนอนุภาคของแต่ละกลุ่มติดในจุดสัมพัทธ์อีกครั้ง ให้ทำการเปลี่ยนตำแหน่ง GBEST กับกลุ่มข้างเคียงเพื่อให้อนุภาคในกลุ่มนั้นหลุดออกจากจุดสัมพัทธ์ ดังรูป

## สรุปผลการทดลองและข้อเสนอแนะ

### 6.1 สรุปผลการทดลอง

วิธีหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค (PSO) นิยม มาประยุกต์ใช้แก้ปัญหาค่าเหมาะที่สุดที่ ดีที่สุดทั้งในงานด้านการขนส่งคณิตศาสตร์ วิทยาศาสตร์ วิศวกรรมศาสตร์เชิงธุรกิจ และด้าน อุตสาหกรรม เป็นต้น โดยมีงานวิจัยจำนวนหนึ่ง เช่น [3-7] ได้นำเอา PSO มาแบ่งกลุ่มออกเป็นกลุ่ม ย่อยถูกเรียกว่าวิธีการหาค่าเหมาะสมที่สุดแบบหลายกลุ่มอนุภาค (MPSO)

จากข้อดีของ PSO เมื่อเปรียบเทียบกับขั้นตอนวิธีเชิงวิวัฒนาการพบว่า PSO สามารถ แก้ปัญหาค่าเหมาะที่สุดที่ ดีที่สุดได้ดีกว่าหลายปัญหาเพราะวิธีแบบ PSO นั้นมีพฤติกรรมการลู่เข้าหา ค่าตอบที่ดีที่สุดของกลุ่มตัวเอง ( $GBEST_i$ ) ดังนั้น PSO จึงมีความเร็วในการลู่เข้าหาค่าตอบมากกว่า และได้รับคำตอบจากการค้นหาที่ดีกว่า จึงมีความเร็วในการลู่เข้าหาค่าตอบได้ดีกว่าขั้นตอนวิธีเชิง วิวัฒนาการแบบอื่น ดังนั้นจึงได้รับคำตอบจากการค้นหาที่ดีกว่า นอกจากนี้ PSO มีข้อดีอื่นอีก มากมายเช่น สามารถนำไปประยุกต์ใช้กับปริภูมิของคำตอบขนาดใหญ่ สามารถเขียนโปรแกรมหรือ น ไปประยุกต์ใช้แก้ปัญหาค่าเหมาะที่สุดที่ ดีที่สุดได้ง่าย สามารถนำไปประยุกต์ใช้ปัญหาที่มีรูปแบบปริภูมิ ค่าตอบถูกแทนด้วยจำนวนจริง สามารถนำไปประยุกต์ใช้ปัญหาที่มีรูปแบบปริภูมิค่าตอบถูกแทน ด้วยไบนารี สามารถนำไปประยุกต์ใช้ปัญหาที่มีรูปแบบปริภูมิค ตอบถูกแทนด้วยการสลับล ดับกันใน พืชคณิต การคำนวณของ PSO มีประสิทธิภาพที่ดีและค้นหาได้เร็ว สำหรับการพัฒนานั้นจะพบว่า พารามิเตอร์ของ PSO ที่ต้อง กหนดค่ามีจ นวนน้อย ดังนั้น จึงเป็นสาเหตุให้นักวิจัย ำหนดมากน ำเอา PSO มาประยุกต์ใช้เพื่อแก้ปัญหาค่าเหมาะที่สุดที่ ดีที่สุด (Optimization Problems)

แต่อย่างไรก็ตาม PSO ก็มีข้อเสียหลักด้วยกันหลายข้อ เช่น ปัญหาค่าลู่เร็วเกินไปของ อนุภาค และปัญหาการติดในจุดสุดสัมพัทธ์ของอนุภาค PSO เมื่อมีการค้นหาในระยะเวลาหนึ่ง ทำให้ เมื่อนำ PSO ไปใช้ลดหรือแก้ปัญหาค่าเหมาะที่สุดที่ ดีที่สุดจะได้รับผลลัพธ์จากการค้นหาไม่ดีเท่าที่ควร ดังนั้น เพื่อลดหรือแก้ปัญหาค่าเหมาะที่สุดที่ ดีที่สุดของ PSO จึงมีหลายงานวิจัยที่ได้นำเอา PSO มา ประยุกต์กับเทคนิคหรือวิธีต่างๆ เพื่อเพิ่มความหลากหลายหรือลดปัญหาของ PSO

ดังเช่น วิทยานิพนธ์ฉบับนี้มีแนวคิดในการเพิ่มความหลากหลายให้กับอนุภาคภายในกลุ่มของ PSO ต้องการปรับปรุง PSO ให้มีประสิทธิภาพการค้นหาที่ดีขึ้น โดยพิจารณาจากลักษณะการค้นหา ค่าตอบของ PSO ที่เหมาะสมกับการค้นหาแบบระยะใกล้ (Local Search) ที่อนุภาคลู่เข้าหาอนุภาค ที่ ดีที่สุด (GBEST) จนติดในจุดสุดสัมพัทธ์ (Local Optimum) โดยวิธีที่ได้นำเสนอนี้ได้ทำการแบ่งกลุ่ม อนุภาคออกเป็นกลุ่มย่อย (MPSO) แต่ละกลุ่มมีตำแหน่งที่ ดีที่สุด ( $GBEST_i$ ) ของตัวเอง เคลื่อนที่ใน รูปแบบของ PSO ปกติ เพื่อเพิ่มการค้นหาแบบระยะไกล (Global Search) ให้กับอนุภาคแต่ละกลุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้มีโอกาสในการค้นหาบริเวณที่ต่างกัน แต่เนื่องจากจำนวนอนุภาคลดลงทำให้แต่ละกลุ่มติดในจุดสัมพัทธ์ได้ง่ายขึ้น ดังนั้น วิธีที่ที่ได้นำเสนอนี้ จึงได้เพิ่มขึ้นตอนการเปลี่ยนข้อมูลกันระหว่างกลุ่มอนุภาค โดยใช้ วิธีการค้นหาที่เหมาะสมแบบอนุภาคหลายกลุ่มด้วยการเปลี่ยนตำแหน่งที่ดีที่สุดและการเปลี่ยนตำแหน่งของอนุภาคเพียงเล็กน้อย (MPSOSM) ซึ่งเป็นวิธีที่เกิดจากการประยุกต์นำเอาปัญหาการติดในจุดสุดสัมพัทธ์ของอนุภาคแต่ละกลุ่มของ MPSO มาพิจารณาการเปลี่ยนตำแหน่งที่ดีที่สุดของแต่ละกลุ่มรวมกับการเปลี่ยนตำแหน่งของอนุภาค (X) เพียงเล็กน้อย เพื่อให้อนุภาคหลุดออกจากจุดติดในจุดสัมพัทธ์ แต่เมื่อเปลี่ยนตำแหน่งของ (*GBEST<sub>i</sub>*) ในระยะเวลาหนึ่งอนุภาคทั้งหมดของทุกกลุ่มจะเคลื่อนที่เข้าหากันจนติดในจุดสัมพัทธ์เดียวกัน เพื่อลดปัญหานี้ จึงได้เพิ่มวิธีการเปลี่ยนตำแหน่งของอนุภาคทั้งหมด (X) รวมกับการกระจายหรือเปลี่ยนตำแหน่งของอนุภาคที่ดีที่สุด (*GBEST<sub>i</sub>*) เมื่ออนุภาคของทุกกลุ่มเคลื่อนที่จนติดในจุดสุดสัมพัทธ์เดียวกัน โดยทั้งสองเทคนิคนี้เมื่อประยุกต์ใช้ร่วมกันสามารถแก้หรือลดปัญหาการติดในจุดสุดสัมพัทธ์และเพิ่มประสิทธิภาพการค้นหาของ PSO ให้ได้รับคำตอบจากการค้นหาที่ดีขึ้น

สำหรับการนำการเปลี่ยนตำแหน่งของอนุภาคที่ดีที่สุดและการกระจายตำแหน่งของอนุภาคไปประยุกต์ใช้กับ PSO โดยปกติแล้วสามารถเพิ่มความหลากหลายให้กับประชากรจึงทำให้อนุภาคสามารถหลีกเลี่ยงหรือหลุดจากจุดสุดสัมพัทธ์ได้ดีกว่า PSO ปกติ เพราะขั้นตอนการทำงานเหล่านี้ได้เพิ่มความหลากหลายและการค้นหาค่าตอบในเชิงลึกให้กับอนุภาคที่มีโอกาสในการพบคำตอบที่ดีขึ้นจากบริเวณใกล้เคียงเดิมแทนการค้นหาค่า ตอบให้กว้างขึ้นในขณะที่ติดอยู่ในจุดสุดสัมพัทธ์โดยวิธีการเลือกกระจายตำแหน่งของอนุภาคทั้งหมดและตำแหน่งของอนุภาคที่ดีที่สุดเพียงเล็กน้อย ทำให้อนุภาคมีโอกาสในการค้นหาค่าตอบได้มากขึ้นและเพียงพอที่จะทำให้อนุภาคหลุดออกจากจุดสุดสัมพัทธ์จะทำให้ได้รับคำตอบจากการค้นหาที่ดีกว่า PSO ปกติ

จากผลการทดลองในบทที่ 5 ในหัวข้อที่ 5.3, 5.4 จะเห็นว่าวิธีที่วิทยานิพนธ์ฉบับนี้นำเสนอ (MPSOSM) ได้ผลลัพธ์ในการค้นหาค่าตอบดีกว่าขั้นตอนวิธีในเอกสารอื่นที่นำมาเปรียบเทียบโดยขั้นตอนวิธีเหล่านี้มีลักษณะใกล้เคียงกันขั้นตอนวิธีที่นำเสนอ ทั้งในปัญหาฟังก์ชันมาตรฐาน (PSO, FMPSO, SCMSO, MPSOI, MPSOR1, MPSOR2, MPSOSM) จากผลการทดลองแสดงให้เห็นว่า MPSOSM มีประสิทธิภาพในการค้นหาดีกว่า PSO, FMPSO, SCMSO, MPSOI, MPSOR1, MPSOR2 และ MPSOSM ทั้งในด้านคุณภาพของค่าตอบที่ได้รับและความน่าเชื่อถือของขั้นตอนวิธี โดย MPSOSM สามารถหาค่าตอบพบในทุกปัญหาฟังก์ชันมาตรฐานที่นำมาทดลองและหาค่าตอบได้พบทุกครั้ง เนื่องจาก MPSOSM สามารถลดหรือแก้ปัญหาการติดในจุดสุดสัมพัทธ์ของ PSO ได้ดีที่สุดในเมื่อเปรียบเทียบกับ PSO ที่ถูกนำมาใช้ในการทดสอบเปรียบเทียบทั้งหมด จึงทำให้ MPSOSM ได้รับผลลัพธ์การค้นหาค่าตอบที่ดีที่สุด

## 6.2 ข้อเสนอแนะ

วิธีที่นำเสนอ (MPSOSM) ได้ใช้หลักการทำงานของ PSO เป็นหลัก ดังนั้นข้อดีของ MPSOSM จึงได้รับการสืบทอดมาจาก PSO เช่น พารามิเตอร์ของ MPSOSM มีจำนวนน้อยที่ต้องการการกำหนดค่าขั้นตอนการทำงานของ MPSOSM มีรูปแบบที่ไม่ซับซ้อน และการทำงานของวิธี MPSOSM มีประสิทธิภาพที่ดีกว่าวิธีที่ได้นำมาเปรียบเทียบ นอกจากนี้ MPSOSM ยังสามารถลดหรือแก้ปัญหาคาการติดในจุดสุดสัมพัทธ์ได้ดีกว่า PSO ซึ่งทำให้ MPSOSM ได้รับผลลัพธ์การค้นหาคาตอบที่ดีกว่า PSO ปกติ

จากผลการทดลองในบทที่ 5 MPSOSM สามารถหาคาตอบได้พบทุกปัญหาที่นำมาทดลองของปัญหาฟังก์ชันมาตรฐานแต่การที่ MPSOSM สามารถหาคาตอบพบได้นั้นก็เพราะการกำหนดจำนวนของการกระจายตาแห่งของอนุภาค (PBEST) และการกระจายตาแห่งที่ดีที่สุดของอนุภาคทั้งหมด (GBEST) เพื่อให้อนุภาคไม่ได้ออกไปไกลเกินไป และเพราะการกำหนดจำนวนของอนุภาคและรอบของการค้นหา รวมทั้งพารามิเตอร์ของขั้นตอนการกระจายอนุภาคนั้นเหมาะสมกับปัญหาที่แก้ไขจึงสามารถหาคาตอบพบแต่ในการใช้งานจริงเราไม่ทราบคาตอบของปัญหาที่นำขั้นตอนวิธี MPSOSM ไปแก้ไขจึงไม่ทราบว่า พารามิเตอร์ที่ใช้นี้สามารถแก้ปัญหาได้เหมาะสมหรือไม่ โดยปกติแล้ว PSO เองก็ต้องการกำหนดพารามิเตอร์อยู่บางส่วนแต่สำหรับวิธีที่นำเสนอต้องมีพารามิเตอร์เพิ่มขึ้นจาก PSO ปกติ จึงทำให้ขั้นตอนวิธีมีความยุ่งยากในการประยุกต์ใช้งานดังนั้นน่าจะมีขั้นตอนการปรับพารามิเตอร์เหล่านี้ให้เหมาะสมกับปัญหาที่นำไปประยุกต์ใช้ ซึ่งอาจจะทำให้ประสิทธิภาพในการค้นหาคาตอบดีขึ้นและประยุกต์ใช้งานได้ง่ายขึ้น นอกจากนี้วิธีที่นำเสนอนี้

ถึงแม้ผลการทดลองในบทที่ 5 วิธีที่นำเสนอ (MPSOSM) สามารถหาคาตอบที่ดีที่สุดของปัญหาที่นำมาทดลองพบทุกปัญหา แต่ผลการทดลองในบทที่ 5 ก็ไม่ใช่ตัวแทนของปัญหาการหาจุดที่ดีที่สุดที่มีความซับซ้อนมากหรือตัวแทนของปัญหาการหาจุดที่ดีที่สุดทั้งหมดในโลกเป็นแค่ตัวแทนของปัญหาการหาจุดที่ดีที่สุดที่มีความซับซ้อนในระดับหนึ่งเท่านั้น ดังนั้น MPSOSM ไม่ได้รับประกันว่าจะสามารถแก้ปัญหาคาการหาจุดที่ดีที่สุดได้ทุกกรณี เพราะถ้าหากนำ MPSOSM ไปประยุกต์ใช้กับปัญหาการหาจุดที่ดีที่สุดที่มีความซับซ้อนมาก MPSOSM ก็อาจจะเกิดปัญหาคาการติดในจุดสุดสัมพัทธ์และไม่สามารถหาคาตอบที่ดีที่สุดพบตัวอย่างเช่น ถ้าหากเพิ่มจำนวนมิติของฟังก์ชันมาตรฐานเป็น 200 มิติทุกฟังก์ชัน แล้วกำหนดพารามิเตอร์เท่าเดิม คาตอบที่ได้จากการค้นหาด้วย MPSOSM อาจจะไม่พบคาตอบที่ดีที่สุดของฟังก์ชันมาตรฐานที่นำมาทดลองนี้ เป็นต้น ดังนั้น MPSOSM ไม่สามารถแก้ปัญหาคาการติดในจุดสุดสัมพัทธ์ได้อย่างสมบูรณ์ เพียงแต่ลดปัญหาคาการติดในจุดสุดสัมพัทธ์หรือแก้ปัญหาคาการติดในจุดสุดสัมพัทธ์ได้ถ้าหากปัญหาการหาจุดที่ดีที่สุดนั้นมีไม่ซับซ้อนมากเกินไปดังนั้น MPSOSM ควรจะต้องมีการศึกษาพัฒนาให้มีประสิทธิภาพในการแก้ปัญหาคาการหาจุดที่ดีที่สุดที่ซับซ้อนมากให้สามารถแก้หรือลดแก้ปัญหาคาการติดในจุดสุดสัมพัทธ์ให้ได้ดีขึ้นเพื่อให้ได้คาตอบที่ดีขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้ MPSOSM ใช้การทำงานของ PSO เป็นส่วนใหญ่ ซึ่งการทำงานของ PSO เป็นการค้นหาแบบสุ่ม ดังนั้นการค้นหาคำตอบในแต่ละรอบอาจจะได้คำตอบไม่เท่ากัน ในบางครั้งอาจจะพบคำตอบที่ดีที่สุดและในบางครั้งอาจจะไม่พบคำตอบที่ดีที่สุดถึงแม้ผลการทดลองจะพบว่า MPSOSM สามารถหาคำตอบที่ดีที่สุดในทุกชุดทดลองพบทุกครั้ง แต่นั่นก็ไม่ได้เป็นการรับประกันว่า MPSOSM จะสามารถหาคำตอบที่ดีที่สุดพบทุกครั้งในทุกปัญหา เพียงแต่ผลการทดลองแสดงให้เห็นว่า MPSOSM เป็นขั้นตอนวิธีที่มีความน่าเชื่อถือมากกว่า PSO ดังนั้น MPSOSM ควรจะต้องมีการศึกษาพัฒนาให้สามารถรับประกันคำตอบที่ได้จากการค้นหาให้ได้รับคำตอบที่ดีที่สุดเท่าเดิมทุกครั้ง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เอกสารอ้างอิง

- [1] R. C. Eberhart, and J. Kennedy, "Particle Swarm Optimization," IEEE International Conference on Neural Networks, pp. 1942-1948, 1995.
- [2] R. C. Eberhart, and J. Kennedy, "A New Optimizer Using Particle Swarm Theory," Proceedings of the 6th International Symposium on Micro Machine and Human Science, pp. 39-43, 1995.
- [3] B. Al-Kazemi, and C. K. Mohan, "Multi-phase discrete particle swarm optimization," In Proc. of 4th Int. Workshop on Frontiers on Evolut.Alg., Research Triangle Park, NC, 2002.
- [4] F.van Den Bergh, "An analysis of particle swarm optimizers," Ph.D. Dissertation, University of Pretoria, 2001
- [5] T. Peram, K. Veeramachaneni, and C. K. Mohan, "Fitness-distance ratio based particle swarm optimization," In Proc. of IEEE Swarm Intell. Symp., Indianapolis, IN, pp. 88-94, 2003.
- [6] S. Baskar, and P. N. Suganthan, "A novel concurrent particle swarm optimization," In Proc. of Cong. on Evolut. Comput., Portland, OR, pp. 792-796, 2004.
- [7] B. Al-Kazemi, and C. K. Mohan, "Multi-phase discrete particle swarm optimization," In Proc. of 4th Int. Workshop on Frontiers on Evolut.Alg., Research Triangle Park, NC, 2002.
- [8] Liang, Jane-Jing, and Ponnuthurai Nagarathnam Suganthan. "Dynamic multi-swarm particle swarm optimizer with local search," Evolutionary Computation, 2005. The 2005 IEEE Congress on. Vol. 1. ieee, 2005.
- [9] Changhe Li, Qing Zhang, Yong Liu, and Lishan Kang, "A Fast Particle Optimization Algorithm with Cauchy Mutation and Natural Selection Strategy," Proceedings of the 2nd international conference Advances in computation and Intelligence 2007, Wuhan, China, pp. 334-343.
- [10] C. Ratanavilisagul, and B. Kruatrachue, "Selective Crossover base on Fitness in Multi Swarm Optimization," International Conference on Emerging Trends in Computer and Image Processing (ICETCIP'2011) Bangkok Dec., 2011.
- [11] เฉียบวุฒิ รัตนวิไลสกุล, "การหลีกเลี่ยง โลกคอล ออปติ้มัม สำหรับ พาร์ติเคิล สวอร์ม ออปติ้มัม เซชัน โดยการปรับรอบการกลายพันธุ์และการเปลี่ยนค่า โกลบอล เบสท์", วิทยานิพนธ์ วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2556.
- [12] C. Ratanavilisagul, and B. Kruatrachue, "A Modified Particle Swarm Optimization with Mutation and Reposition," ICIC international, 2014.
- [13] Zhao, S-Z., et al. "Dynamic multi-swarm particle swarm optimizer with harmony search," Expert Systems with Applications 38.4 (2015): 3735-3742.
- [14] เฉียบวุฒิ รัตนวิไลสกุล, "การปรับปรุงการหาค่าเหมาะสมแบบอนุภาคกลุ่มด้วยการสุ่มรอบอนุภาคและกระจายอนุภาค", วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา วิศวกรรมศาสตรดุษฎีบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2558.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [15] K. Chengkhuntod, B. Kruatrachue, and K. Siriboon, "A Modified Multi-Swarm Optimization with Interchange GBEST and Particle Redistribution," *ieecon*, 2017.
- [16] K. Chengkhuntod, B. Kruatrachue, and K. Siriboon, "Improving Multi-Swarm by Slightly Mutation Particle and GBEST of Stuck Swarm Along with Randomly Selecting GBEST of other Swarm," *The 21st International Computer Science and Engineering Conference 2017*, November 15-18, 2017 Bangkok, Thailand.
- [17] W. Phuchan, B. Kruatrachue, and K. Siriboon. "Hybrid Multi-swarm with Harmony Search algorithm," *ECTI-CON 2017*, 27-30 June 2017, Phuket, Thailand.
- [18] Rui Tang, Simon Fong, Xin-She Yang and Suash Deb, "Wolf Search Algorithm with Ephemeral Memory," *2012 Seventh International Conference on Digital Information*, August 22-24, 2012.
- [19] C. Ratanavilisakul, "Improvements of Particle Swarm Optimization with Samplings Particle Proximity and Particle Redistribution", Ph.D Dissertation, King Mongkut Institute of Technology Ladkrabang, Thailand, 2015.
- [20] H. Gao and W. Xu, "Particle swarm algorithm with hybrid mutation strategy *Applied Soft Computing*", pp. 5129-5142, 2011.
- [21] L. Fung-Bao, "Inverse estimation of wall heat flux by using particle swarm optimization algorithm with Gaussian mutation", *International Journal of Thermal Sciences*, vol. 54, pp. 62-69, 2012.
- [22] J. Tang and X. Zhao, "Particle Swarm Optimization with Adaptive Mutation WASE", *International Conference on Information Engineering*, pp. 234-237, 2009.
- [23] P. Millie, T. Radha, V. P. Singh and A. Ajith, "Particle Swarm Optimization Using Sobol Mutation", *Int. Journal of Simulation Systems, Science and Technology*, Vol. 10, pp. 91-96, 2009.
- [24] X. Wu and M. Zhong, "Particle Swarm Optimization Based on Power Mutation", *ISECS International Colloquium on Computing, Communication, Control, and Management*, pp. 464-467, 2009.
- [25] Z. Xiaoxia, "A modified particle swarm optimization with differential evolution mutation", *In Sixth International Conference on Natural Computation*, pp. 2530-2534, 2010.
- [26] C. Li, S. Yang and I.A. Korejo, "An Adaptive Mutation operator for Particle Swarm", *Proceedings of the 2008 UK Workshop on Computational Intelligence*, pp. 165-170, 2008.
- [27] J. Tang and X. Zhao, "A hybrid particle swarm optimization with adaptive local search", *Journal of Networks*, pp. 411-418, 2010.
- [28] J. Tang and X. Zhao, "Particle swarm optimization using adaptive local search", *International Conference on Future BioMedical Information Engineering*, art., pp. 300-303, 2009.
- [29] C. C. A. Omar, R. Andrew, and L. F. Rafael, "A PSO-based algorithm with local search for multimodal optimization without constraints", *Informatica (CLEI), Conferencia Latinoamericana En*, pp. 1-7, 2012.
- [30] C. Lei, "Particle Swarm Optimization with a Novel Mutation Operator", *International Conference on Mechatronic Science, Electric Engineering and Computer*, pp. 970-973, 2011.

- [31] M. Pant, R. Thangaraj and A. Abraham, "Particle swarm optimization using adaptive mutation", Proc. 19th International Conference on Database and Expert Systems Application, pp. 519-523, 2008.
- [32] R. Hsin and C. Kuo-Hua, "Particle swarm optimization with mutation for the inspection allocation in reentrant production systems", Proceedings of the International Conference on Machine Learning and Cybernetics, pp. 1025-1030, 2011.
- [33] M. S. Mohamed, G. E. R. Mona and K. A. Hanan, "A New Fixed Channel Assignment Algorithm Using Adaptive Mutation Particle Swarm Optimization", Innovations on Communication Theory, pp. 1-5, 2012.
- [34] L. Xuedan, W. Qiang, L. Haiyan and L. Lili, "Particle Swarm Optimization with Dynamic Inertia Weight and Mutation", Third International Conference on Genetic and Evolutionary Computing, pp. 620-623, 2009.
- [35] Y. Gao and Y. Duan, "A New Particle Swarm Optimization Algorithm with Adaptive Mutation Operator", Second International Conference on Information and Computing Science, pp. 58-61, 2009.
- [36] F. van Den Bergh, "An analysis of particle swarm optimizers", Ph.D. Dissertation, University of Pretoria, 2001.
- [37] L. Ning, S. Debao, C. Yigang, and Z. Tong, "Particle swarm optimization with mutation operator", Computer Engineering and Applications, pp.12-14, 2004.
- [38] Wikipedia, the free encyclopedia, "Local optimum". [[http://en.wikipedia.org/wiki/Local\\_optimum](http://en.wikipedia.org/wiki/Local_optimum)]
- [39] Wikipedia, the free encyclopedia, "Global optimum". [[http://en.wikipedia.org/wiki/Global\\_optimum](http://en.wikipedia.org/wiki/Global_optimum)]
- [40] Wikipedia, the free encyclopedia, "Knapsack problem". [[http://en.wikipedia.org/wiki/Knapsack\\_problem](http://en.wikipedia.org/wiki/Knapsack_problem)]
- [41] C. Chiemchaisri, J. P. Juanga, and C. Visvanathan, "Municipal solid waste management in Thailand and disposal emission inventory", Springer Science and Business Media, pp. 34-41, 2006.
- [42] R. A. Sarker and C. S. Newton, "Optimization Modelling A Practical Approach", CRC Press, Taylor & Francis Group, 2008.+
- [43] C. Maurice, "Particle Swarm Optimization", John Wiley & Sons, 2010.
- [44] Information on <http://www.sfu.ca/~ssurjano/index.html/>
- [45] Information on <http://www-optima.amp.i.kyoto-u.ac.jp/>
- [46] M. Marcin and S. Czesław, Test functions for optimization needs, 2005.
- [47] A. P. Ernesto, "Multivariate Test Functions Library in C for Unconstrained Global Optimization", Department of Mathematics U.P. Diliman, 2005.
- [48] H. Abdel-Rahman, "Global Optimization Test Problems". [[http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar\\_files/TestGO.htm](http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar_files/TestGO.htm)]
- [49] S. Sonja, B. Derek, and U. F. Simon, "Virtual Library of Simulation Experiments: Test Functions and Datasets". [<http://www.sfu.ca/~ssurjano>]
- [50] DEAP Project, "Benchmarks". [<http://deap.gel.ulaval.ca/doc/0.9/api/benchmarks.html>]
- [51] Computational Intelligence Laboratory, Peking University, "Benchmark functions for PSO". [[http://www.cil.pku.edu.cn/resources/benchmark\\_pso](http://www.cil.pku.edu.cn/resources/benchmark_pso)]
- [52] G. Andrea, "Global Optimization Benchmarks and AMPGO". [[http://infinity77.net/global\\_optimization](http://infinity77.net/global_optimization)]

- [53] C. A. Floudas, P. M. Pardalos, C. Adjiman, W. R. Esposito, Z. H. Gümüs, S. T. Harding, J. L. Klepeis, C. A. Meyer and C. A. Schweiger, "Handbook of Test Problems in Local and Global Optimization", Springer, The Netherlands., pp. 484, 1999.
- [54] S. Nootyaskool and B. Kruatrachue, "Hybrid Genetic Algorithm with Baum-Welch Algorithm by using Diversity Population Technique", International Symposium on Communications and Information Technology (ISCIT), pp. 15-20, 2006.
- [55] S. Lee, S. Soak, S. Oh, W. Pedrycz and M. Jeon, "Modified binary particle swarm optimization", Progress in Natural Science, pp. 1161-1166, 2008.
- [56] R. C. Eberhart and J. Kennedy, "A discrete binary version of the particle swarm algorithm", Proceedings of the 1997 Conference on Systems, Man, and Cybernetics, pp. 4104-4109, 1997.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ก

### รายละเอียดของฟังก์ชันมาตรฐาน

การค้นหาค่าตอบของฟังก์ชันมาตรฐาน คือการค้นหาค่าที่ดีที่สุดของฟังก์ชันมาตรฐาน สำหรับฟังก์ชันมาตรฐานที่รู้จักที่ดีที่สุด คือฟังก์ชันมาตรฐานชนิดที่ไม่มีเงื่อนไขบังคับ (Unconstrained Optimization) โดยฟังก์ชันจะเขียนในรูป  $f(x)$  ประกอบไปด้วย  $x$  แทนด้วยมิติ ซึ่งภายในฟังก์ชัน มีหลายมิติจึงเขียนในรูป  $x = \{x_1, x_2, \dots, x_n\}$  สำหรับขอบเขตของฟังก์ชันแสดงในช่วง  $x_{lower} < x_i < x_{upper}$  ความซับซ้อนของฟังก์ชันมาตรฐานกำหนดได้จากจำนวนมิติ สำหรับการทดลองนี้ ใช้จำนวนมิติสูงสุดอยู่ที่ 100 มิติ ซึ่งมีความซับซ้อนสูงเพื่อใช้ในการเปรียบเทียบประสิทธิภาพของ ขั้นตอนวิธีที่ใช้ค้นหาการหาจุดที่ดีที่สุด

ฟังก์ชันมาตรฐานที่นำมาทดลองได้อ้างอิงจาก [17] โดยมีทั้งหมด 24 ฟังก์ชัน สำหรับ รายละเอียดของฟังก์ชันนั้นแสดงไว้ในตาราง ก.1 และรูปฟังก์ชันได้แสดงไว้ในรูป ก.1 ถึง ก.24 เป็น ลักษณะ 2 มิติเพื่อให้ง่ายต่อการมองเห็นลักษณะของฟังก์ชันและสมการของฟังก์ชันได้แสดงไว้ในสมการ ก.1 ถึง ก.24 โดยฟังก์ชันมาตรฐานสามารถแบ่งตามลักษณะใด 2 แบบดังนี้

1. ฟังก์ชันมัลติโมดอล (Multimodal Function) คือฟังก์ชันที่มีจุดสุดสัมพัทธ์ โดย ลักษณะจะมีความลาดเอียงหลายรูปแบบผสมผสานกัน เนื่องจากสร้างมาจากการ รวมฟังก์ชันยูนิโมดอลหลายรูปแบบ ทำให้ฟังก์ชันมัลติโมดอลนี้มีความซับซ้อน มากกว่าฟังก์ชันยูนิโมดอล
2. ฟังก์ชันยูนิโมดอล (Single-modal Function) คือฟังก์ชันที่ไม่มีจุดสุดสัมพัทธ์ โดย ลักษณะจะมีความลาดเอียงเพียงรูปแบบเดียว เช่นฟังก์ชันมาตรฐานที่สร้างมาจาก สัญญาณ SINE เพียงอย่างเดียว ฟังก์ชันยูนิโมดอลนี้ดูเหมือนง่ายต่อการค้นหา ค่าตอบแต่ในความจริงฟังก์ชันยูนิโมดอลมีส่วนประกอบของพื้นที่ราบเรียบจึงทำให้ การค้นหานั้นจะเคลื่อนที่เข้าหาค่า ท่อกระทบ ได้ยาก

ตารางที่ ก.1 สรุปสมบัติของฟังก์ชันคณิตศาสตร์ที่ใช้ในการทดลอง

ชื่อฟังก์ชัน	ขอบเขต $[X_{\max}, X_{\min}]$	จุดที่ตี ที่สุด	จำนวน มิติ	คุณสมบัติ	$f(X^*) = 0$ $X^*$	$V_{\max} \cdot V_{\min}$	$Eva_{\max}$
ACKLEY	$x \in [-32.768, 32.768]^n$	$f(x) = 0$	100	Multimodal	$(0, \dots, 0)$	$\pm 32.768$	10,000,000
GRIEWANK	$x \in [-300, 300]^n$	$f(x) = 0$	100	Multimodal	$(0, \dots, 0)$	$\pm 300$	10,000,000
RASTRIGIN	$x \in [-5.12, 5.12]^n$	$f(x) = 0$	100	Multimodal	$(0, \dots, 0)$	$\pm 5.12$	10,000,000
ROSENBROCK	$x \in [-2.048, 2.048]^n$	$f(x) = 0$	100	Multimodal	$(1, \dots, 1)$	$\pm 2.048$	10,000,000
SCHWEFEL	$x \in [-500, 500]^n$	$f(x) = 0$	100	Multimodal	$(420.9687, \dots, 420.9687)$	$\pm 500$	10,000,000
COSINE MIXTURE	$x \in [-1, 1]^n$	$f(x) = 0$	100	Multimodal	$(0, \dots, 0)$	$\pm 1$	10,000,000
EXPONENTIAL	$x \in [-5.12, 5.12]^n$	$f(x) = 0$	100	Multimodal	$(0, \dots, 0)$	$\pm 1$	10,000,000
LEVY	$x \in [-5.12, 5.12]^n$	$f(x) = 0$	100	Multimodal	$(1, \dots, 1)$	$\pm 10$	10,000,000
MICHALEWICZ	$x \in [-65.536, 65.536]^n$	$f(x) = 0$	100	Multimodal	$(2.20, 1.57)$	$\pm 3.14$	10,000,000
DIXON-PRICE	$x \in [0, PI]^n$	$f(x) = 0$	10	Multimodal	$(0, \dots, 0)$	$\pm 100$	10,000,000
STEP	$x \in [-100, 100]^n$	$f(x) = 0$	100	Multimodal	$(0.5, \dots, 0.5)$	$\pm 100$	10,000,000
SCHAFFER	$x \in [-100, 100]^n$	$f(x) = 0$	2	Multimodal	$(0, \dots, 0)$	$\pm 10$	10,000,000
HOLDER	$x \in [-10, 10]^n$	$f(x) = 0$	2	Multimodal	$(-8.05502, 9.66459)$	$\pm 4.5$	10,000,000
BEALE	$x \in [-4.5, 4.5]^n$	$f(x) = 0$	2	Multimodal	$(3, 0.5)$	$\pm 10$	10,000,000
SHUBERT	$x \in [-10, 10]^n$	$f(x) = 0$	2	Multimodal	$(0, \dots, 0)$	$\pm 2$	10,000,000
GOLDSTEIN- PRICE	$x \in [-2, 2]^n \cdot f(X^*) = 3$	$f(x) = 0$	2	Multimodal	$(0, -1)$	$\pm 100$	10,000,000
TRID	$x \in [-100, 100]^n$	$f(x) = 0$	10	Multimodal	$(-d^2, d^2)$	$\pm 100$	10,000,000
SIX-HUMP CAMEL	$x \in [-2, 2]^n$	$f(x) = 0$	2	Multimodal	$(0.0898, -0.7126)$	$\pm 5.12$	10,000,000
SPHERE	$x \in [-5.12, 5.12]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 5.12$	200,000
PARALLEL	$x \in [-5.12, 5.12]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 65.536$	200,000
ROTATED	$x \in [-65.536, 65.536]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 10$	200,000
CIGAR	$x \in [-10, 10]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 1$	200,000
BROWN	$x \in [-1, 4]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 10$	500,000
MULTIMOD	$x \in [-10, 10]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 5.12$	500,000
ZAKHAROV	$x \in [-5, 10]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 100$	200,000
EASOM	$x \in [-100, 100]^n$	$f(x) = 0$	2	Unimodal	$(0, \dots, 0)$	$\pm 100$	200,000
EGGHOLDER	$x \in [-512, 512]^n$	$f(x) = 0$	2	Unimodal	$(512, 404.2319)$	$\pm 100$	200,000

หมายเหตุ สำหรับฟังก์ชันที่มีเครื่องหมาย \* อยู่ข้างหน้าจะมีการกำหนดพารามิเตอร์หรือคุณสมบัติพิเศษเพิ่มเติม โดยรายละเอียดจะถูกเขียนไว้ในหัวข้อของฟังก์ชันนั้นๆ ที่จะถูกกล่าวดังต่อไปนี้

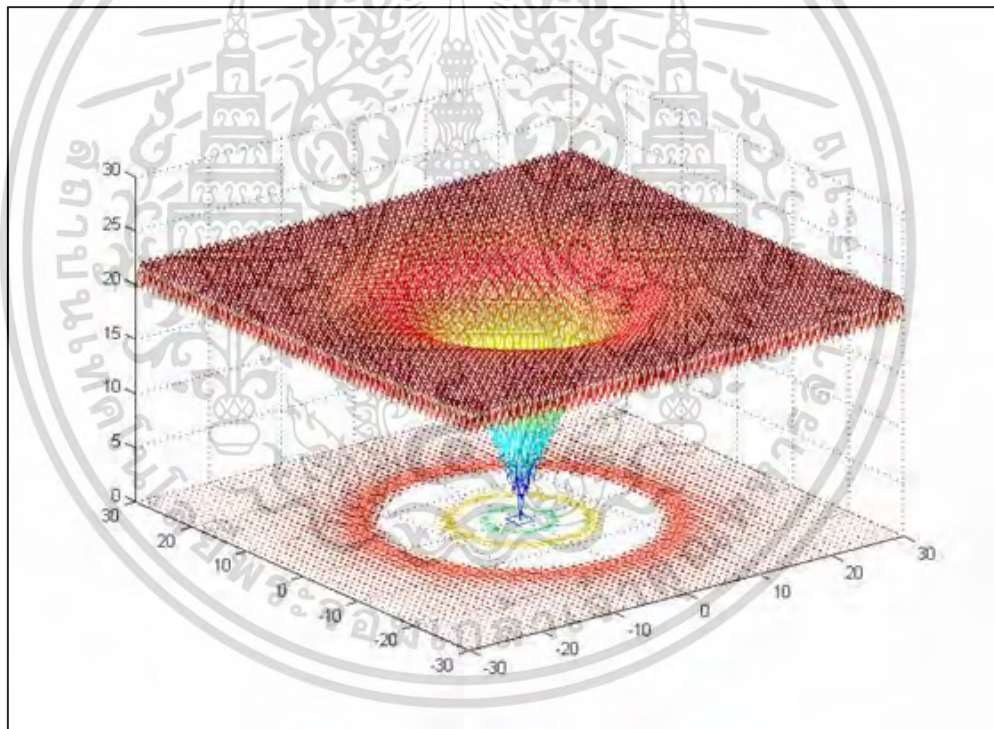
- ฟังก์ชัน ACKLEY

ฟังก์ชัน ACKLEY คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยฟังก์ชันนี้มีพื้นผิวที่ครอบคลุมด้วยการกระจายตัวแบบ EXPONENTIAL ซึ่งการกระจายตัวแบบนี้จะทำให้เกิดจุดสุดสัมพัทธ์กระจายตัวในรูปแบบปกติอย่างสม่ำเสมอบนพื้นที่ผิวของปริภูมิของค ทอพ โดยรูปของฟังก์ชันคือรูป ก.1 และสมการของฟังก์ชันคือสมการ ก.1

$$f(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e \quad (\text{ก.1})$$

มีขอบเขตอยู่ในช่วง  $x \in [-30, 30]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.1 ฟังก์ชัน ACKLEY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

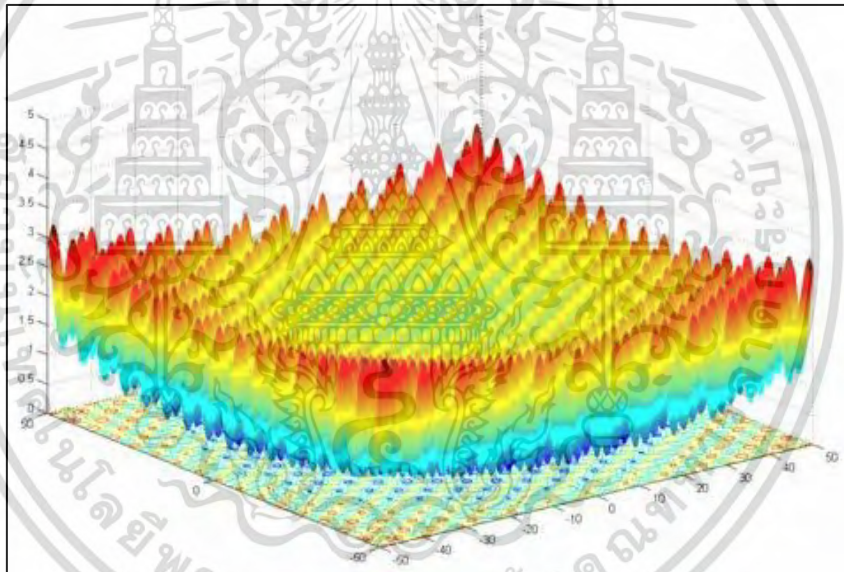
- ฟังก์ชัน GRIEWANK

ฟังก์ชัน GRIEWANK คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยมีลักษณะเหมือนกับฟังก์ชัน RASTRIGIN แต่มีการกระจายตัวของจุดสุดสัมพัทธ์เป็นบริเวณกว้างกว่าฟังก์ชัน RASTRIGIN อย่างไรก็ตามจุดสุดสัมพัทธ์ก็มีการกระจายตัวอย่างปกติ และสมมาตรรอบพื้นที่ผิวของฟังก์ชัน โดยรูปของฟังก์ชันคือรูป ก.2 และสมการของฟังก์ชันคือสมการ ก.2

$$f(x) = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos(x_i / \sqrt{i}) + 1 \quad (\text{ก.2})$$

มีขอบเขตอยู่ในช่วง  $x \in [-300, 300]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.2 ฟังก์ชัน GRIEWANK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

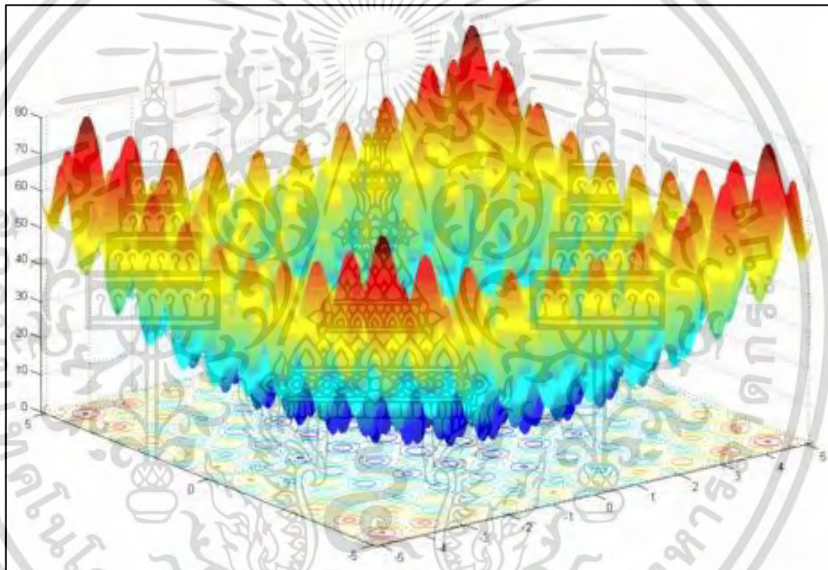
- ฟังก์ชัน RASTRIGIN

ฟังก์ชัน RASTRIGIN คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยจุดสุดสัมพัทธ์นั้นได้มีการกระจายตัวแบบ COSINE บนพื้นผิวของฟังก์ชัน แต่จุดสุดสัมพัทธ์มีการกระจายตัวอย่างปกติและสม่ำเสมอบนพื้นผิวของฟังก์ชัน โดยรูปของฟังก์ชันคือรูป ก.3 และสมการของฟังก์ชันคือสมการ ก.3

$$f(x) = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i)) \quad (\text{ก.3})$$

มีขอบเขตอยู่ในช่วง  $x \in [-5.12, 5.12]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.3 ฟังก์ชัน RASTRIGIN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

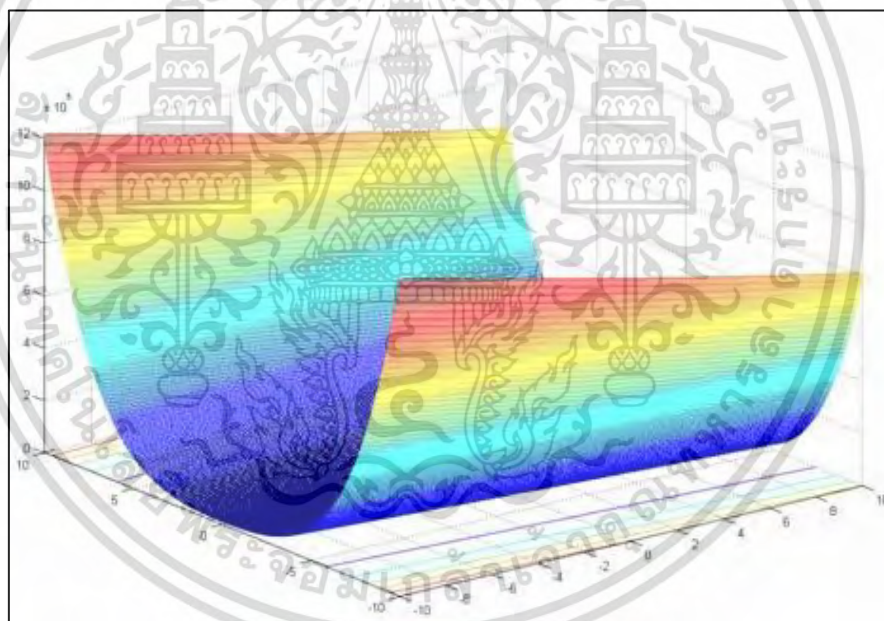
- ฟังก์ชัน ROSENBROCK

ฟังก์ชัน ROSENBROCK คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันยูนิโมดอล โดยภายในฟังก์ชันมีช่วงที่ยาวและลึกเป็นรูปแบบ HYPERBOLIC ซึ่งทำให้เกิดรูปแบบเป็นร่องหุบเขาที่กว้างยาวและราบเรียบ ดังนั้นจึงเป็นเรื่องยากที่จะลู่เข้าหาคาตอบภายในร่องหุบเขานี้ สำหรับฟังก์ชันนี้ถ้ามีจำนวนมิติมากกว่า 3 มิติจะมีสมบัติเป็นฟังก์ชันมัลติโมดอล โดยรูปของฟังก์ชันคือรูป ก.4 และสมการของฟังก์ชันคือสมการ ก.4

$$f(x) = \sum_{i=1}^n [100(x_{i+1} - x_i)^2 + (x_i - 1)^2] \quad (\text{ก.4})$$

มีขอบเขตอยู่ในช่วง  $x \in [-2.048, 2.048]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (1, \dots, 1), f(x^*) = 0$



รูปที่ ก.4 ฟังก์ชัน ROSENBROCK

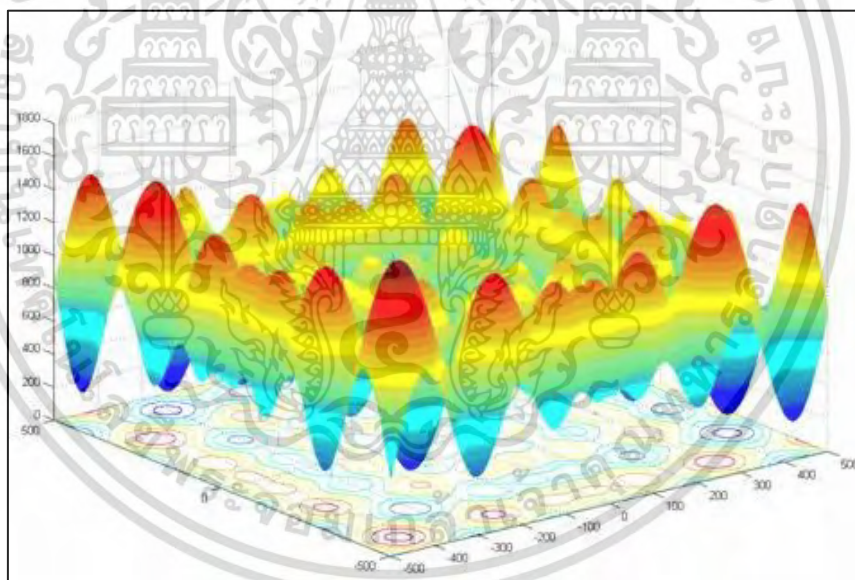
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน SCHWEFEL

ฟังก์ชัน SCHWEFEL คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยฟังก์ชันจะยกกำลัง  $1/2$  ซึ่งทำให้ยากที่จะแบ่งพื้นผิวออกเป็นส่วนๆ นอกจากนี้ในแต่ละแกนจะมีการกระจายตัวของจุดสุดสัมพัทธ์เป็นจำนวนมาก ดังนั้นการค้นหาค่าที่ดีที่สุดจึงเป็นเรื่องยากและทำได้ช้า เพราะว่ายากที่จะแบ่งพื้นผิวออกเป็นส่วนๆ โดยรูปของฟังก์ชันคือรูป ก.5 และสมการของฟังก์ชัน คือสมการ ก.5 สำหรับฟังก์ชันนี้ในการทดลองในบทที่ 5 สมการการกลายพันธุ์ที่ 3.15 และ 3.16 ได้มีการนำค่าคงที่ คือ 2.4 คูณกับ  $\text{rand}()$

$$f(x) = 418.9829 \times n + \sum_{i=1}^n (x_i \times \sin(\sqrt{|x_i|})) \quad (\text{ก.5})$$

มีขอบเขตอยู่ในช่วง  $x \in [-500, 500]^n$   
 ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (-420.96, \dots, -420.96), f(x^*) = 0$



รูปที่ ก.5 ฟังก์ชัน SCHWEFEL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

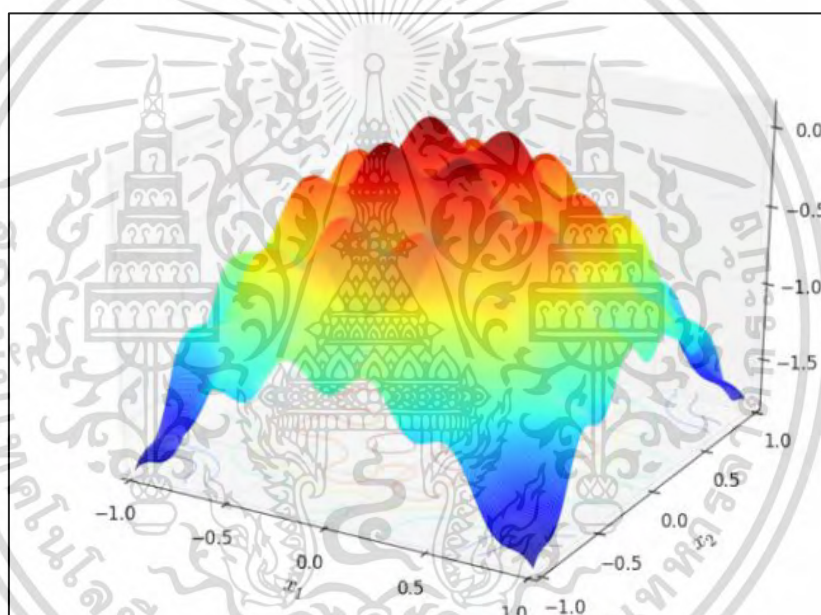
- ฟังก์ชัน COSINE MIXTURE

ฟังก์ชัน COSINE MIXTURE คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยฟังก์ชันมีจุดสุดสัมพัทธ์อยู่ในขอบเขตการค้นหา โดยรูปของฟังก์ชันคือรูป ก.6 และสมการของฟังก์ชันคือสมการ ก.6

$$f(x) = -0.1 \times \sum_{i=0}^n \cos(5\pi x_i) + \sum_{i=0}^n x^2 + 0.1n \quad (\text{ก.6})$$

มีขอบเขตอยู่ในช่วง  $x \in [-1,1]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.6 ฟังก์ชัน COSINE MIXTURE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

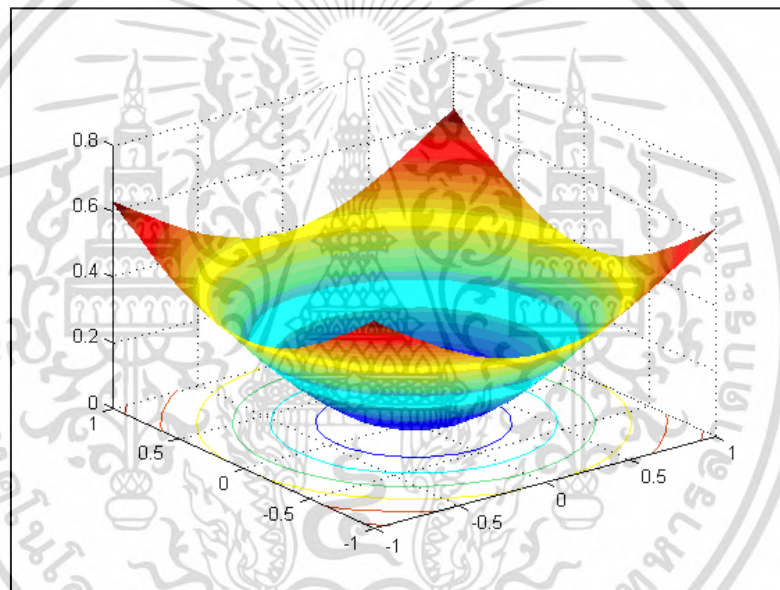
- ฟังก์ชัน EXPONENTIAL

ฟังก์ชัน EXPONENTIAL คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยฟังก์ชันมีจุดสุดสัมพัทธ์อยู่ภายในขอบเขตการค้นหา โดยรูปของฟังก์ชันคือรูป ก.7 และสมการของฟังก์ชันคือสมการ ก.7

$$f(x) = -\exp(-0.5 \sum_{i=1}^n x_i^2) + 1 \quad (\text{ก.7})$$

มีขอบเขตอยู่ในช่วง  $x \in [-1, 1]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.7 ฟังก์ชัน EXPONENTIAL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน LEVY

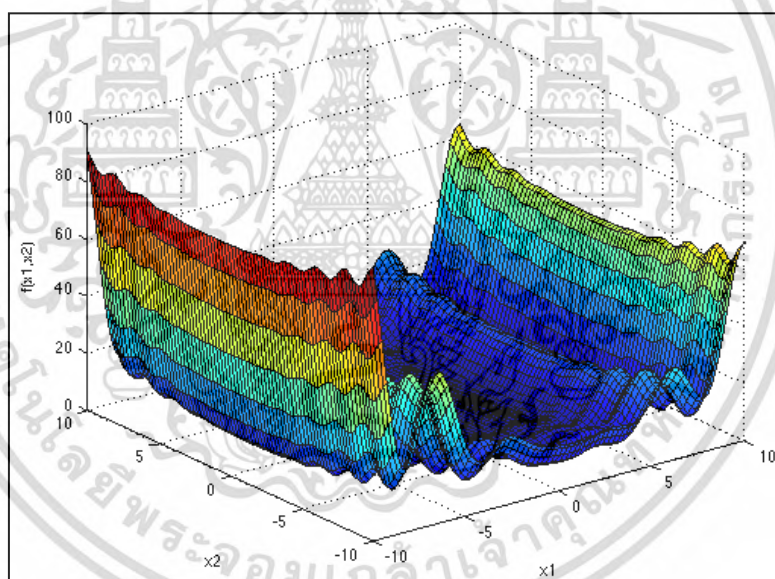
ฟังก์ชัน LEVY คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยฟังก์ชันมีจุดสุดสัมพัทธ์อยู่ในขอบเขตการค้นหา โดยรูปของฟังก์ชันคือรูป ก.8 และสมการของฟังก์ชันคือสมการ ก.8 สำหรับฟังก์ชันนี้ได้ลบค่าด้วย  $10^{-30}$  เพราะถ้าแทนค่า 1 ลงไปในสมการนี้แล้ว ค่าตอบที่ได้ประมาณ  $10^{-30}$  ดังนั้นถ้าหากลบด้วย จะได้ค่าเท่ากับ 0

$$f(x) = \sin^2(\pi w_1) + \sum_{i=1}^{d-1} (w_i - 1)^2 [1 + 10 \sin^2(\pi w_i + 1)] + (w_d - 1)^2 [1 + \sin^2(2\pi w_d)] \quad (ก.8)$$

$$w_i = 1 + \frac{x_i - 1}{4}, \text{ for all } i = 1, \dots, d$$

มีขอบเขตอยู่ในช่วง  $x \in [-10, 10]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (1, \dots, 1), f(x^*) = 0$



รูปที่ ก.8 ฟังก์ชัน LEVY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน MICHALEWICZ

ฟังก์ชัน MICHALEWICZ คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยฟังก์ชันมีจุดสุดสัมพัทธ์อยู่ในขอบเขตการค้นหาค้นหาสำหรับพารามิเตอร์  $m$  เป็นตัวกำหนดความสูงชันของหุบเขาและสันเขา โดย  $m$  ยิ่งมีค่ามากก็จะยิ่งทำให้การค้นหาที่มีความยุ่งยากมากขึ้นซึ่งโดยปกติแล้วค่า  $m$  ถูกกำหนดให้มีความเท่ากับ 10 โดยรูปของฟังก์ชันคือรูป ก.9 และสมการของฟังก์ชันคือสมการ ก.9 สำหรับฟังก์ชันนี้ในการทดลองในบทที่ 5 ได้มีการกำหนดจำนวนมิติเท่ากับ 10 มิติ และมีการบวกค่า 9.66015 เข้าไปเพื่อให้ค่า  $f(x)$  มีค่าเท่ากับศูนย์

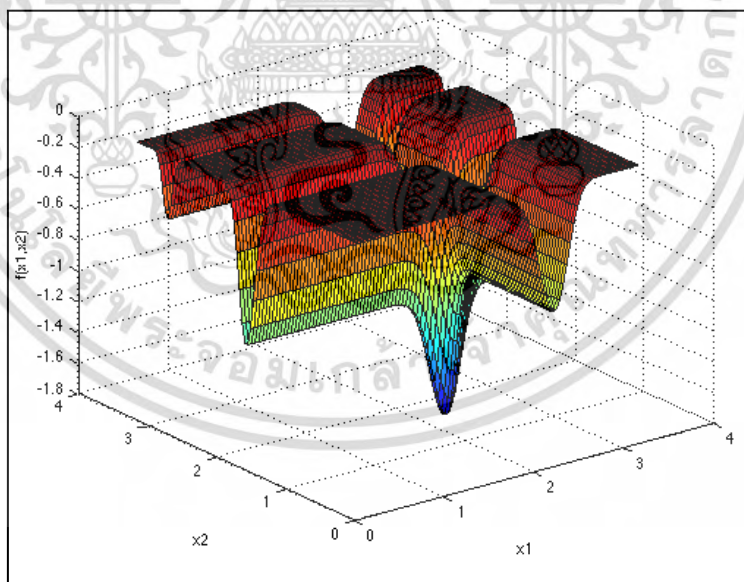
$$f(x) = -\sum_{i=1}^d \sin(x_i) \sin^{2m}\left(\frac{ix_i^2}{\pi}\right) \quad (\text{ก.9})$$

มีขอบเขตอยู่ในช่วง  $x \in [0, \pi]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $d = 2 : f(x^*) = -1.8013, x^* = (2.20, 1.57)$

$d = 5 : f(x^*) = -4.687658$

$d = 10 : f(x^*) = -9.66015$



รูปที่ ก.9 ฟังก์ชัน MICHALEWICZ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

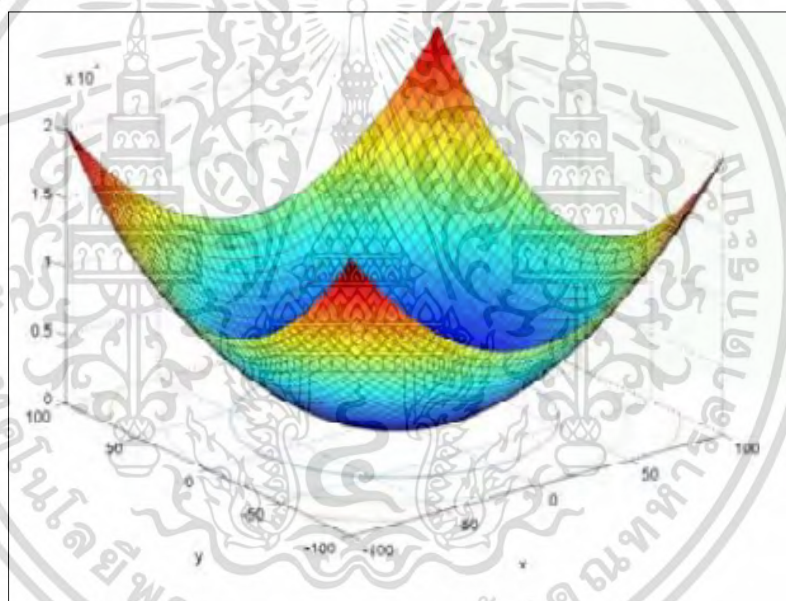
- ฟังก์ชัน STEP

ฟังก์ชัน STEP คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยลักษณะของฟังก์ชันมีทั้งค่าที่มีความชันสูงและพื้นราบลาดถึงจุดสุดสัมบูรณ์ นอกจากนี้มีการปรับค่าทศนิยมขึ้นลงทำให้คำตอบแบ่งออกเป็นชั้นๆ ส่งผลให้สมบัติเป็นฟังก์ชันมัลติโมดอล โดยรูปของฟังก์ชันคือรูป ก.10 และสมการของฟังก์ชันคือสมการ ก.10

$$f(x) = \sum_{i=0}^{n-1} (\lfloor x_i \rfloor + 0.5)^2 \quad (\text{ก.10})$$

มีขอบเขตอยู่ในช่วง  $x \in [-100, 100]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0.5, \dots, 0.5), f(x^*) = 0$



รูปที่ ก.10 ฟังก์ชัน STEP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

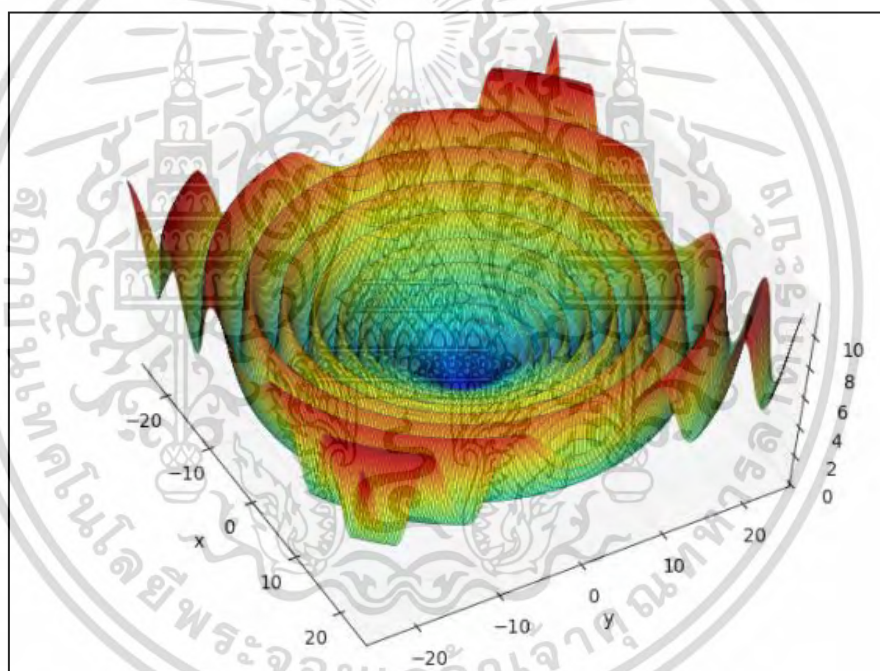
- ฟังก์ชัน SCHAFFER

ฟังก์ชัน SCHAFFER คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยมีจำนวนมิติ 2 มิติ และฟังก์ชันมีจุดสุดสัมพัทธ์อยู่ภายในขอบเขตการค้นหา โดยรูปของฟังก์ชันคือรูป ก.11 และสมการของฟังก์ชันคือสมการ ก.11

$$f(x) = 0.5 + \frac{\sin^2 \sqrt{x^2 + y^2} - 0.5}{(1.0 + 0.001 \times (x^2 + y^2))^2} \quad (\text{ก.11})$$

มีขอบเขตอยู่ในช่วง  $x \in [-100, 100]$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.11 ฟังก์ชัน SCHAFFER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน HOLDER

ฟังก์ชัน HOLDER คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล ฟังก์ชันมีจุดสุดสัมพัทธ์อยู่ภายในขอบเขตการค้นหาและมีจำนวนมิติ 2 มิติ มีจำนวนจุดสุดสัมบูรณ์ 4 จุด โดยรูปของฟังก์ชันคือรูป ก.12 และสมการของฟังก์ชันคือสมการ ก.12

$$f(x) = -\left| \sin(x_1) \cos(x_2) e^{\left(1 - \sqrt{\frac{x_1^2 + x_2^2}{\pi}}\right)} \right| \quad (\text{ก.12})$$

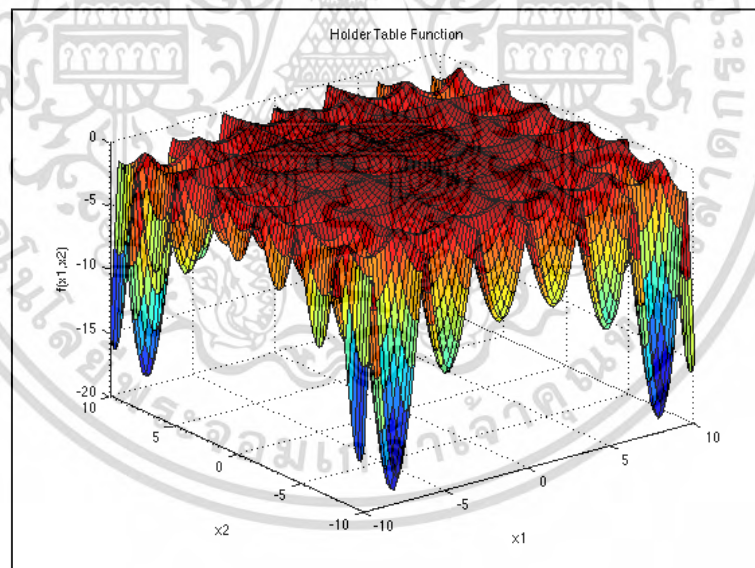
มีขอบเขตอยู่ในช่วง  $x \in [-10, 10]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (8.05502, 9.66459), f(x^*) = -19.2085$

$x^* = (8.05502, -9.66459), f(x) = -19.2085$

$x^* = (-8.05502, 9.66459), f(x) = -19.2085$

$x^* = (-8.05502, -9.66459), f(x) = -19.2085$



รูปที่ ก.12 ฟังก์ชัน HOLDER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

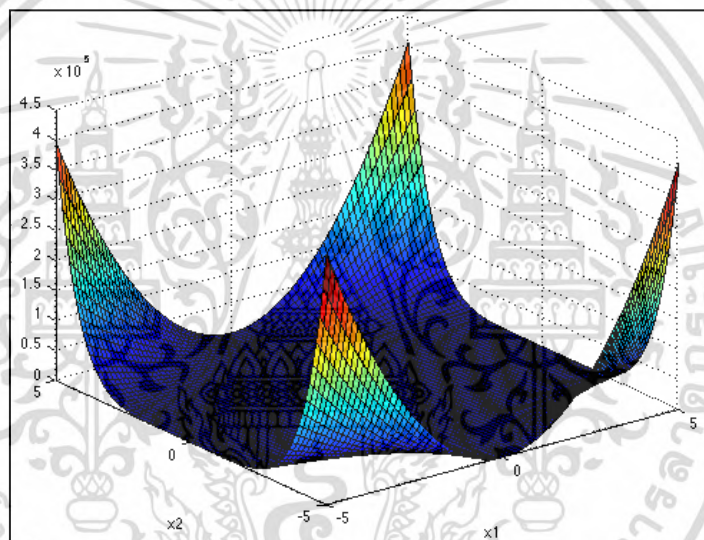
- ฟังก์ชัน BEALE

ฟังก์ชัน BEALE คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล มีจำนวนมิติ 2 มิติ สำหรับฟังก์ชันนี้มียอดแหลมอยู่ที่มุมของขอบเขตปริภูมิของคำตอบ โดยรูปของฟังก์ชันคือรูป ก.13 และสมการของฟังก์ชันคือสมการ ก.13

$$f(x) = (1.5 - x_1 + x_1x_2)^2 + (2.25 - x_1 + x_1x_2^2)^2 + (2.625 - x_1 + x_1x_2^3)^2 \quad (\text{ก.13})$$

มีขอบเขตอยู่ในช่วง  $x \in [-4.5, 4.5]$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (3, 0.5), f(x^*) = 0$



รูปที่ ก.13 ฟังก์ชัน BEALE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

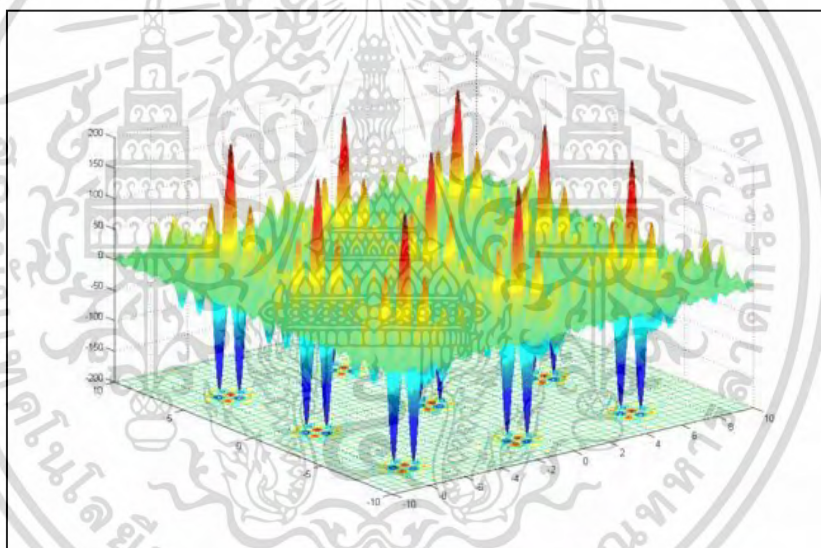
- ฟังก์ชัน SHUBERT

ฟังก์ชัน SHUBERT คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล มีจำนวนมิติ 2 มิติ และมีจุดสุดสัมบูรณ์อยู่ 18 จุดกระจายอยู่บนปริภูมิของคำตอบ โดยรูปของฟังก์ชันคือรูป ก.14 และสมการของฟังก์ชันคือสมการ ก.14 ส สำหรับฟังก์ชันนี้ในการทดลองในบทที่ 5 มีการบวกค่า 186.7309 เข้าไปเพื่อให้ค่า  $f(x)$  มีค่าเท่ากับศูนย์

$$f(x) = \sum_{i=1}^5 i \cos((i+1)x_1 + i) \left( \sum_{i=1}^5 i \cos((i+1)x_2 + i) \right) \quad (\text{ก.14})$$

มีขอบเขตอยู่ในช่วง  $x \in [-10, 10]^2$

ค่าสูงสุดสัมบูรณ์ คือ  $f(x^*) = -186.7309$



รูปที่ ก.14 ฟังก์ชัน SHUBERT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน GOLDSTEIN-PRICE

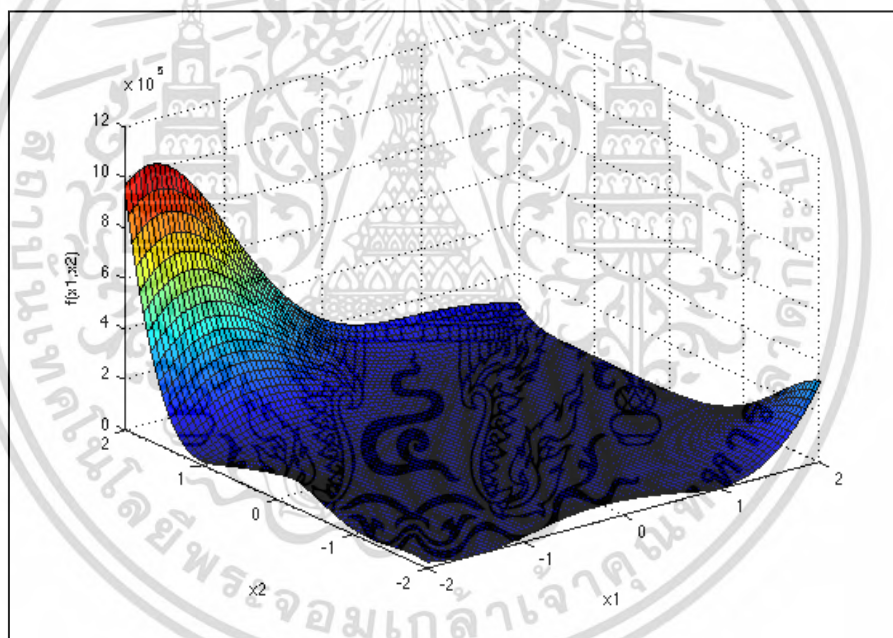
ฟังก์ชัน GOLDSTEIN-PRICE คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยฟังก์ชันมีจุดสุดสัมพัทธ์อยู่ภายในขอบเขตการค้นหาและมีจำนวนมิติ 2 มิติ โดยรูปของฟังก์ชันคือรูป ก.15 และสมการของฟังก์ชันคือสมการ ก.15 สำหรับฟังก์ชันนี้ในการทดลองในบทที่ 5 มีการลบค่า 3 เข้าไปเพื่อให้ค่า  $f(x)$  มีค่าเท่ากับศูนย์

$$f(x^*) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \quad (ก.15)$$

$$\times [30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$$

มีขอบเขตอยู่ในช่วง  $x \in [-2, 2]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, 1), f(x^*) = 3$



รูปที่ ก.15 ฟังก์ชัน GOLDSTEIN-PRICE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน TRID

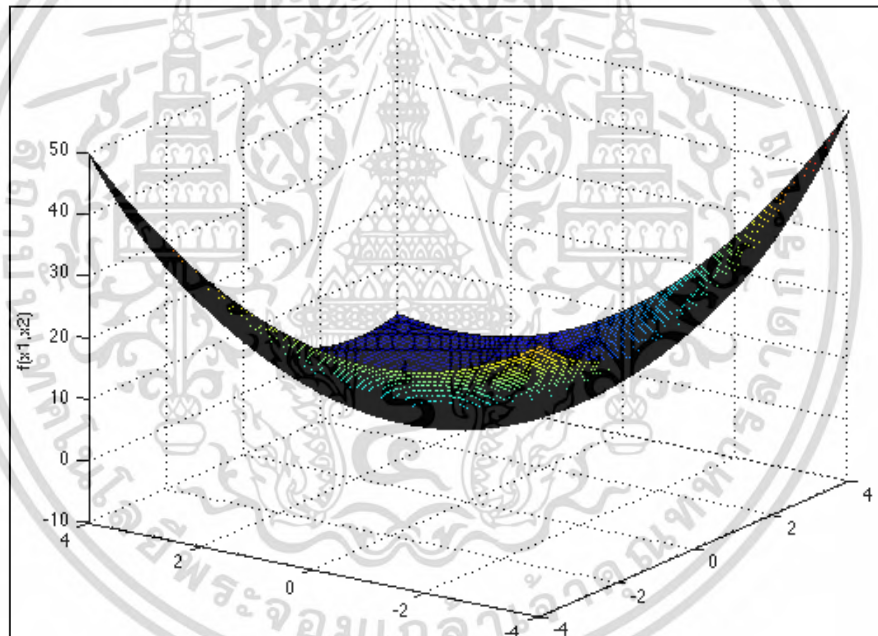
ฟังก์ชัน TRID คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันมัลติโมดอล โดยฟังก์ชันมีจุดสุดสัมพัทธ์อยู่ภายในขอบเขตการค้นหา โดยรูปของฟังก์ชันคือรูป ก.16 และสมการของฟังก์ชันคือสมการ ก.16 สำหรับฟังก์ชันนี้ในการทดลองในบทที่ 5 มีการบวกค่า 200 เข้าไปเพื่อให้ค่า  $f(x)$  มีค่าเท่ากับศูนย์

$$f(x) = \sum_{i=1}^d (x_i - 1)^2 - \sum_{i=2}^d x_i x_{i-1} \quad (\text{ก.16})$$

มีขอบเขตอยู่ในช่วง  $x \in [-d^2, d^2]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $d=6: f(x^*)=3$

$d=10: f(x^*)=-200$



รูปที่ ก.16 ฟังก์ชัน TRID

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน SPHERE

ฟังก์ชัน SPHERE คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันยูนิโมดอล โดยลักษณะเป็นฟังก์ชันต่อเนื่องที่มีโครงสร้างเป็นรูปทรงกรวยคว่ำและไม่มีจุดสุดสัมพัทธ์ มีจุดสุดสัมบูรณ์เพียงจุดเดียว โดยรูปของฟังก์ชันคือรูป ก.17 และสมการของฟังก์ชันคือสมการ ก.17

$$f(x) = \sum_{i=1}^n x_i^2 \quad (\text{ก.17})$$

มีขอบเขตอยู่ในช่วง  $x \in [-5.12, 5.12]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.17 ฟังก์ชัน SPHERE

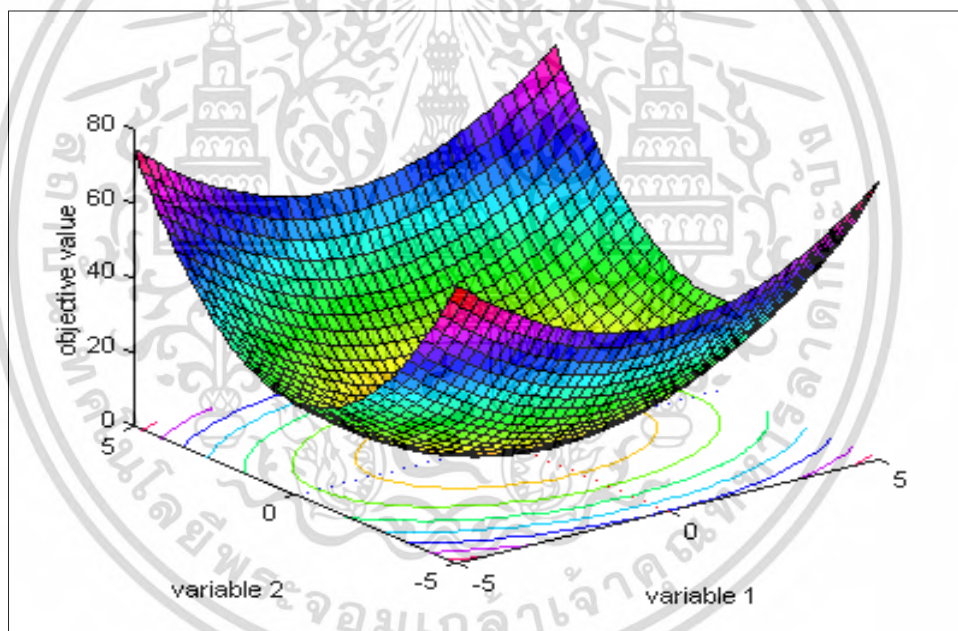
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน PARALLEL ELLIPSOID

ฟังก์ชัน PARALLEL ELLIPSOID คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันยูนิโมดอล โดยลักษณะฟังก์ชันจะมีอัตราการลาดชันเป็นลักษณะทวิคูณ เนื่องจากผลรวมของตัวแปรต้นยกกำลังสองและคูณด้วยจำนวนของมิติรวมกัน ส่งผลให้การลู่เข้าไปหาจุดสุดสัมบูรณ์ต้องใช้เวลาานาน โดยรูปของฟังก์ชันคือรูป ก.18 และสมการของฟังก์ชันคือสมการ ก.18

$$f(x) = \sum_{i=1}^n (i \times x_i^2) \quad (\text{ก.18})$$

มีขอบเขตอยู่ในช่วง  $x \in [-5.12, 5.12]^n$   
ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.18 ฟังก์ชัน PARALLEL ELLIPSOID

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

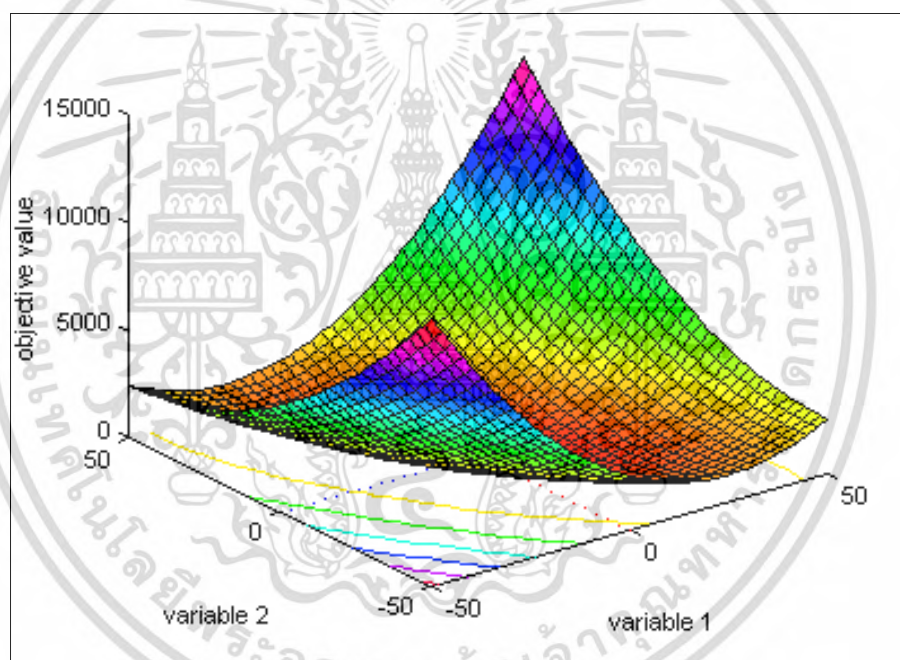
- ฟังก์ชัน ROTATED ELLIPSOID

ฟังก์ชัน ROTATED ELLIPSOID คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันยูนิโมดอล โดยลักษณะฟังก์ชันจะมีอัตราการลาดชันเป็นทวีคูณเป็นผลมาจากผลรวมตัวแปรต้นสองชุดทำให้การลู่อู่หาจุดสุดสัมบูรณ์ต้องใช้เวลาานาน โดยรูปของฟังก์ชันคือรูป ก.19 และสมการของฟังก์ชันคือสมการ ก.19

$$f(x) = \sum_{i=1}^m (\sum_{j=1}^i x_j^2) \quad (\text{ก.19})$$

มีขอบเขตอยู่ในช่วง  $x \in [-65.536, 65.536]^m$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.19 ฟังก์ชัน ROTATED ELLIPSOID

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

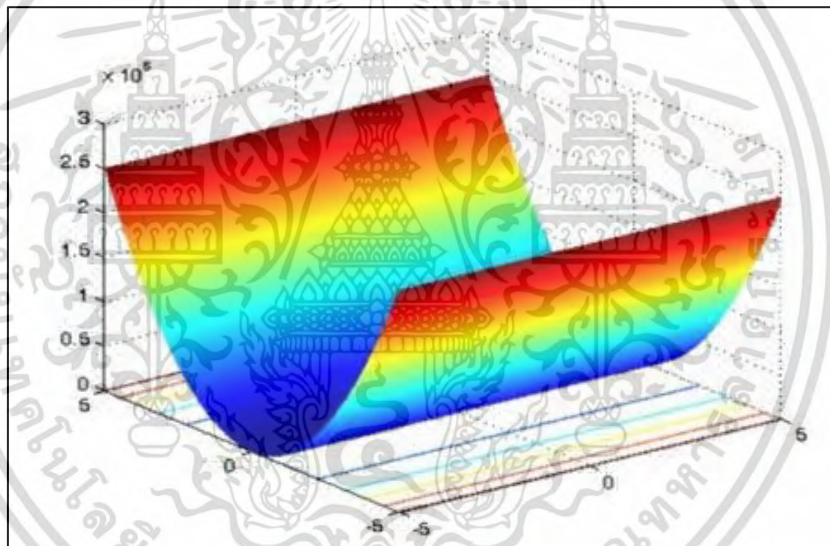
- ฟังก์ชัน CIGAR

ฟังก์ชัน CIGAR คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันยูนิโมดอล โดยลักษณะของฟังก์ชันเป็นฟังก์ชันไม่มีจุดสุดสัมพัทธ์และมีจุดสุดสัมบูรณ์เพียงจุดเดียว โดยรูปของฟังก์ชันคือรูป ก.20 และสมการของฟังก์ชันคือสมการ ก.20

$$f(x) = x_0^2 + 10^6 \sum_{i=1}^N x_i^2 \quad (\text{ก.20})$$

มีขอบเขตอยู่ในช่วง  $x \in [-10, 10]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.20 ฟังก์ชัน CIGAR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

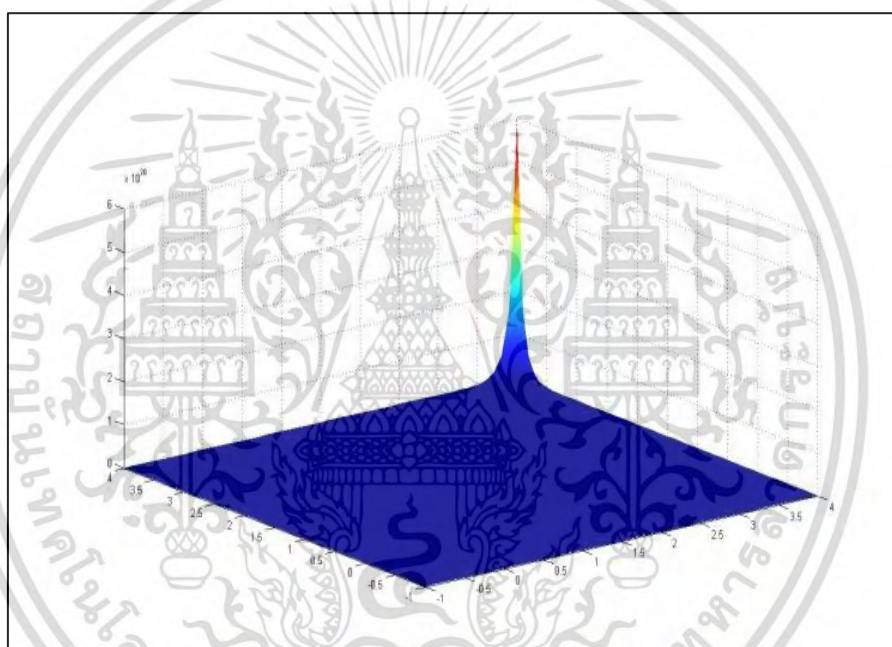
- ฟังก์ชัน BROWN

ฟังก์ชัน BROWN คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันยูนิโมดอล โดยลักษณะของฟังก์ชันเป็นฟังก์ชันไม่มีจุดสุดสัมพัทธ์และมีจุดสุดสัมบูรณ์เพียงจุดเดียว โดยรูปของฟังก์ชันคือรูป ก.21 และสมการของฟังก์ชันคือสมการ ก.21

$$f(x) = \sum_{i=0}^{n-1} [(x_i^2)^{(x_i^2+1)} + (x_{i+1}^2)^{(x_i^2+1)}] \quad (\text{ก.21})$$

มีขอบเขตอยู่ในช่วง  $x \in [-1,4]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.21 ฟังก์ชัน BROWN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน MULTIMOD

ฟังก์ชัน MULTIMOD คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันยูนิโมดอล โดยลักษณะของฟังก์ชันมีทั้งค่าที่มีความชันสูงและพื้นราบลาดถึงจุดสุดสัมบูรณ์ ทำให้การลู่เข้าหาจุดสุดสัมบูรณ์ต้องใช้เวลาาน โดยรูปของฟังก์ชันคือรูป ก.22 และสมการของฟังก์ชันคือสมการ ก.22

$$f(x) = \sum_{i=1}^n |x_i| \times \prod_{i=1}^n |x_i| \quad (\text{ก.22})$$

มีขอบเขตอยู่ในช่วง  $x \in [-10, 10]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.22 ฟังก์ชัน MULTIMOD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

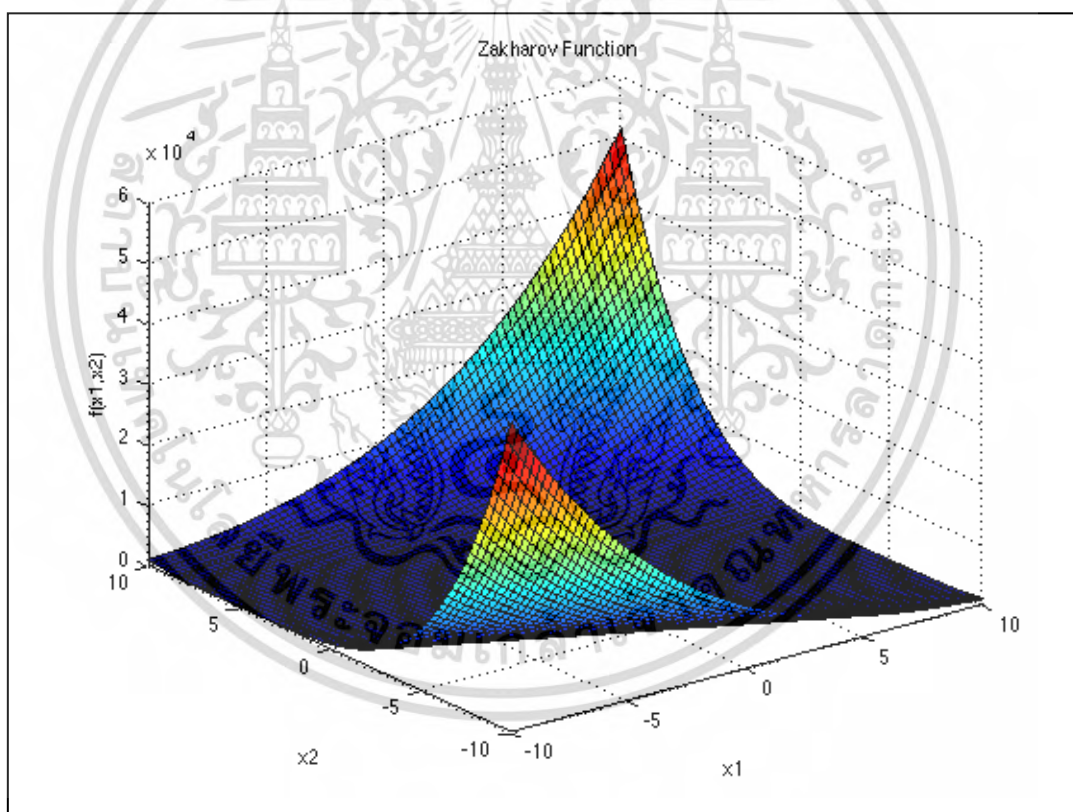
- ฟังก์ชัน ZAKHAROV

ฟังก์ชัน ZAKHAROV คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันยูนิโมดอล โดยลักษณะของฟังก์ชันเป็นฟังก์ชันไม่มีจุดสุดสัมพัทธ์และมีจุดสุดสัมบูรณ์เพียงจุดเดียว โดยรูปของฟังก์ชันคือรูป ก.23 และสมการของฟังก์ชันคือสมการ ก.23

$$f(x) = \sum_{i=1}^n x_i^2 + \left[ \sum_{i=1}^n \frac{i}{2} x_i \right]^2 + \left[ \sum_{i=1}^n \frac{i}{2} x_i \right]^4 \quad (\text{ก.23})$$

มีขอบเขตอยู่ในช่วง  $x \in [-5, 10]^n$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (0, \dots, 0), f(x^*) = 0$



รูปที่ ก.23 ฟังก์ชัน ZAKHAROV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

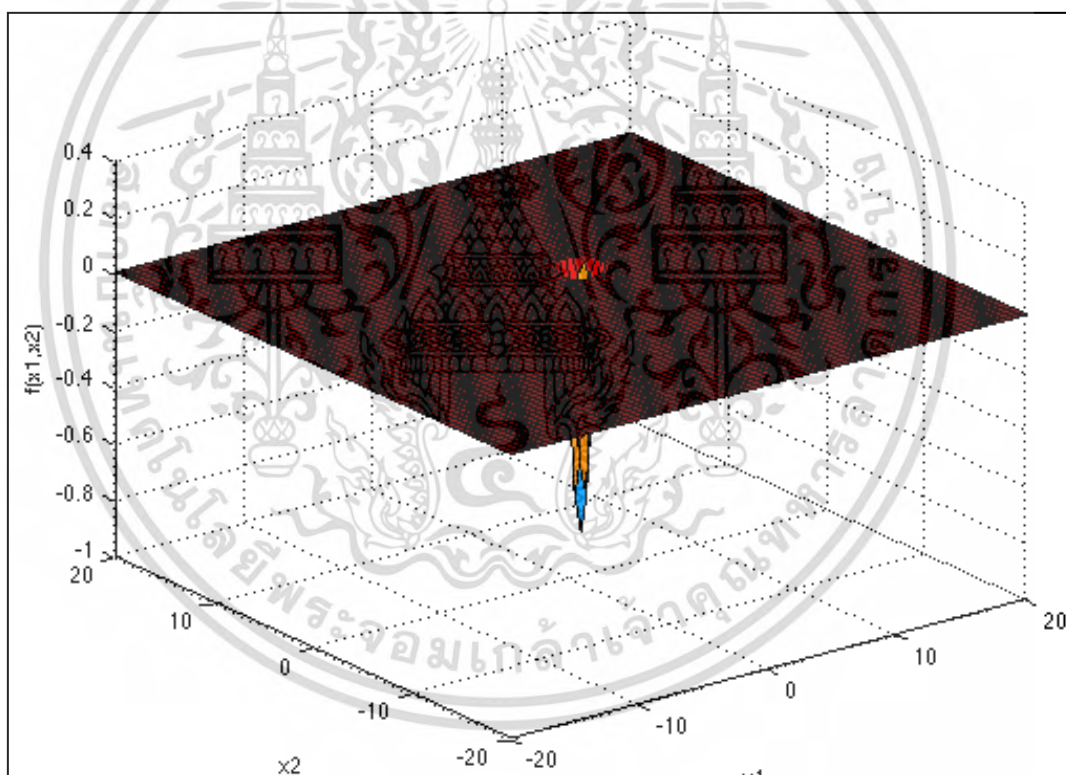
- ฟังก์ชัน EASOM

ฟังก์ชัน EASOM คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันยูนิโมดอล โดยลักษณะของฟังก์ชันคือจุดสุดสัมบูรณ์มีพื้นที่ขนาดเล็กมากเมื่อเทียบกับพื้นที่ค้นหาทั้งหมดโดยรูปของฟังก์ชันคือรูป ก.24 และสมการของฟังก์ชันคือสมการ ก.24 สำหรับฟังก์ชันนี้ในการทดลองในบทที่ 5 มีการบวกค่า 1 เข้าไปเพื่อให้ค่า  $f(x)$  มีค่าเท่ากับศูนย์

$$f(x) = -\cos(x_1) \cos(x_2) \exp(-(x_1 - \pi)^2 - (x_2 - \pi)^2) \quad (\text{ก.24})$$

มีขอบเขตอยู่ในช่วง  $x \in [-100, 100]$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (\pi, \pi), f(x^*) = -1$



รูปที่ ก.24 ฟังก์ชัน EASOM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

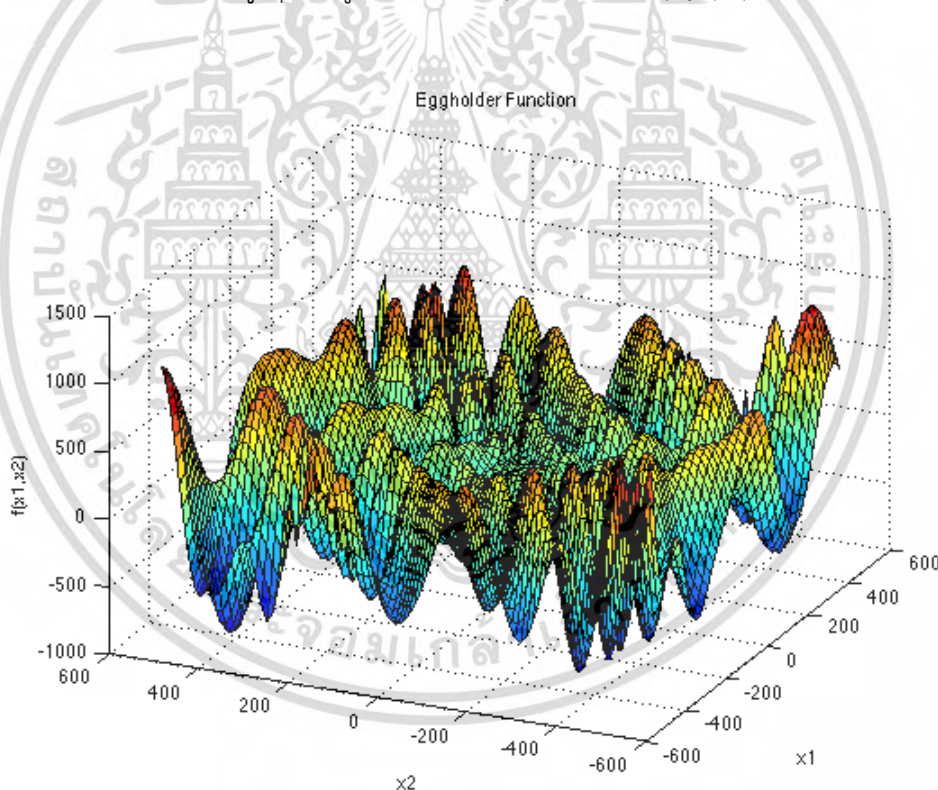
- ฟังก์ชัน EGGHOLDER

ฟังก์ชัน EGGHOLDER คือฟังก์ชันที่จัดอยู่ในกลุ่มฟังก์ชันยูนิโมดอล โดยลักษณะของฟังก์ชันคือฟังก์ชันที่ยากในการเพิ่มประสิทธิภาพเนื่องจากมีจำนวนมินิที่น้อยมากโดยรูปของฟังก์ชันคือรูป ก. 25 และสมการของฟังก์ชันคือสมการ ก.25 สำหรับฟังก์ชันนี้ในการทดลองในบทที่ 5 มีการบวกค่า 1 เข้าไปเพื่อให้ค่า  $f(x)$  มีค่าเท่ากับศูนย์

$$f(x) = -(x_2 + 47) \sin\left(\sqrt{\left|x_2 + \frac{x_1}{2} + 47\right|}\right) - x_1 \sin\left(\sqrt{\left|x_1 - (x_2 + 47)\right|}\right) \quad (\text{ก.24})$$

มีขอบเขตอยู่ในช่วง  $x \in [-512, 512]$

ค่าสูงสุดสัมบูรณ์ คือ  $x^* = (512, 404.2319)$ ,  $f(x^*) = -959.6407$



รูปที่ ก.25 ฟังก์ชัน EGGHOLDER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ข.

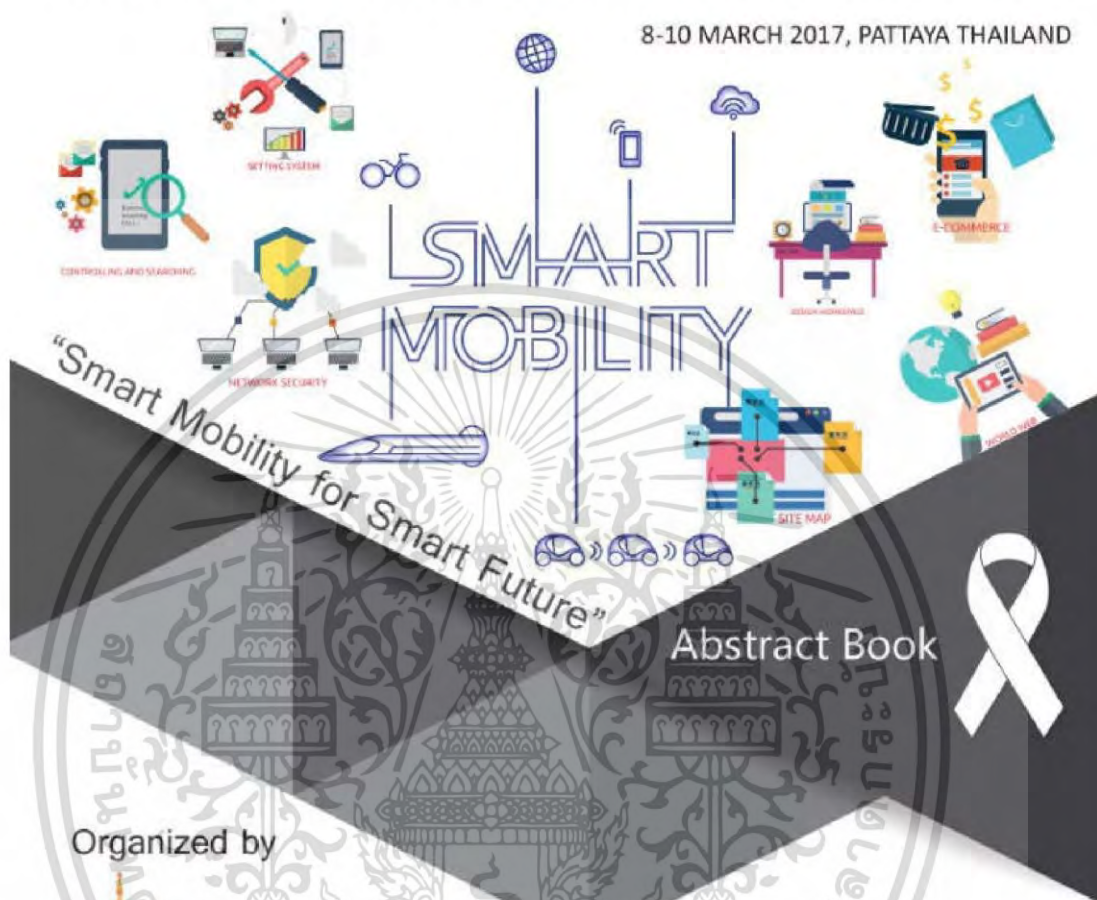
## งานวิจัยที่ได้รับการตีพิมพ์

1. Kanokporn Chengkhuntod, Boontee Kruatrachue, Kritawan Siriboon, “A Modified Multi-Swarm Optimization with Interchange GBEST and Particle Redistribution”, 5<sup>th</sup> International Electrical Engineering Congress, 8-10 March 2017, Pattaya, Thailand, pp 371-374.

# iEECON 2017

The 2017 International Electrical Engineering Congress

8-10 MARCH 2017, PATTAYA THAILAND



Abstract Book

Organized by



Technical Co-sponsored by



POWER & ENERGY

COMMUNICATIONS

ELECTRONICS & CONTROL

DIGITAL SIGNAL PROCESSING

COMPUTER & IT

Sponsored by



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# A Modified Multi-Swarm Optimization with Interchange GBEST and Particle Redistribution

Kanokporn Chengkhuntod, Boontee Kruatrachue, and Kritawan Siriboon

Department of Computer Engineering, Faculty of Engineering

King Mongkut's Institute of Technology Ladkrabang

Bangkok, Thailand

E-mail: k.chengkhuntod@gmail.com, boontec@yahoo.com, kritawans@yahoo.com

**Abstract**—The Particle Swarm Optimization (PSO) is an optimization algorithm using multiples particle to search solution space for an optimize solution. Each particle of PSO moves toward the best solution within its group. For this behavior, PSO often traps in local optimum. Many researchers proposed splitting a swarm into multiple swarms so that they may move to different local optimum. Besides, the mutation operation technique, the natural selection technique and the crossover operation technique are added to normal PSO process. These proposed techniques are called Selective Crossover base on Fitness in Multi-Swarm Optimization (SFMPPO) and Fast Multi-swarm Optimization (FMPSO). However, both techniques used too many evaluation calls dues to crossover and the mutation operation. This paper proposes setting the best position (GBEST) of a trapped swarm to GBEST of the other swarm. Then, the swarm's particle is redistributed in solution space before restart the trapped swarm. This proposed technique is tested on a set of twenty-six benchmark test functions. The experimental results show that the results are better than those of PSO, FMPSO and SFMPPO.

**Keywords**—Swarm intelligence; Multiple Swarm; Particle Swarm Optimization; Cauchy mutation;

## I. INTRODUCTION

Kennedy and Eberhart [1], [2] proposed Particle Swarm Optimization (PSO) in 1995. PSO is inspired by the behavior of bird's flocking and their communication mechanism in solving optimization problems. The PSO has many advantages such as rapid convergence, easy implementation with little parameters to set. The main disadvantages are trapping in local optimum and premature convergence. To overcome these problems, many researchers increase PSO population.

Ratnaweera, Halgamuge and Watson [3] proposed mutation operator with PSO to lessen the local optimum problem and to improve its performance. Sun, Feng and Xu [4] used the quantum model to improve standard PSO, which was called Quantum Particle Swarm Optimization (QPSO). Liu, Xu and Sun [5] proposed improved QPSO by applying Cauchy mutation operator. The results showed that QPSO with mutation were better than solutions of the standard PSO. Krohling and Dos Santos Coelho [6], [7] proposed applying

Cauchy mutation and Gaussian mutation with PSO to the local minima problem.

However, the results from mutation diminish convergence rate [12]. So, some researchers proposed multiple swarms with information exchange between them. Al-Kazemi and Mohan [8] separated population into two groups. The first group moved toward GBEST while the second group moved to the opposite direction. Peram, Veeramachaneni and Mohan [10] also separated population into two groups as in [8] with different update equation. The first group used the same updated equation as standard PSO while the other one used the Fitness-to-Distance ratio updated.

Li, Zhang, Liu and Kang [12] proposed Fast Particle Swarm Optimization (FPSO) where particle position was selected from a better position among Cauchy mutation position and normal update position of PSO. Moreover, they proposed Fast Multiple Swarm (FMPSO) where FPSO population was separated into multiple groups, FPSO were performed in all group and all population of a single random selected group were crossover with the best particle of another random selected group. The experiment results showed improvements over FPSO and the standard PSO.

Ratanavilisagul and Kruatrachue [15] proposed Selective Crossover base on Fitness in Multi-Swarm Optimization (SFMPPO). SFMPPO improved FMPSO by selectively apply crossover operation only when the population had better fitness value. However, the disadvantages of SFMPPO and FMPSO were the extra evaluation calls for each candidate position from mutation operation, crossover operation.

Apart from increasing diversity by mutation operation, reposition technique [16] was used to mute particle position from their position when swarm trap in local optimum.

The diversity of population is very small when PSO is trapped in local optimum [9]. The exchange GBEST between its swarm and other swarms can increase diversity of population [12]. But, the exchanged of GBEST should only be performed when population lacks diversity or trapping in local optimum to avoid slow converging rate.

This paper tackled trapping in local optimum problem using multiple swarm, mutation and reposition technique. When a swarm trapped in local optima, its population was reposition (slightly mutation from its current position) and reset GBEST to that of its neighbor swarm. This method is called "A Multi-Swarm Optimization with Interchange GBEST and Particle Redistribution (MPSOIR)". MPSOIR results are compared with PSO, FMPSO and SFMPSO using twenty-six benchmark test functions. All techniques are compared with equal number of benchmark function calls. The experiment results showed that MPSOIR gets the best results for all test functions.

The rest of this paper is organized as follows. Section 2 explains the standard PSO [1], [2], FPSO [12], FMPSO [12] and SFMPSO [15]. Section 3 explains MPSOIR. Section 4 explains twenty-six benchmark test functions, describes the experiment setup and experiment results. Finally, Section 5 concludes the paper with a brief summary.

## II. RELATED WORK

### A. Particle Swarm Optimization

The member of each population is called particle. The particle shares the information with its neighbors. Each particle's element is velocity ( $V_i$ ), position ( $X_i$ ). The movement of each particle in search space depends on its velocity, the best position of swarm (GBEST) and the best position of each particle (PBEST). Each particle starts with random position and velocity. The fitness value of each particle is calculated from the objective function at its current position  $X_i$ . Each particle is updated its position and velocity as follows:

$$\tilde{V}_i^{(t+1)} = \omega \tilde{V}_i^{(t)} + c_1 \text{rand}() (pbest_i^{(t)} - X_i^{(t)}) + c_2 \text{rand}() (gbest^{(t)} - X_i^{(t)}) \quad (1)$$

$$X_i^{(t+1)} = X_i^{(t)} + \tilde{V}_i^{(t+1)} \quad (2)$$

Where  $X_i^{(t)}$  represents the current position of particle  $i$ ,  $X_i^{(t-1)}$  represents the previous position of particle  $i$ ,  $V_i^{(t)}$  represents the previous velocity of particle  $i$ ,  $\tilde{V}_i^{(t+1)}$  represents the current velocity of particle  $i$ ,  $p_i^{(t)}$  represents the best position of  $i$  particle (PBEST),  $p_g^{(t)}$  represents the best position of swarm (GBEST),  $0 \leq \omega \leq 1$  is an inertia weight,  $c_1$  and  $c_2$  are acceleration constants,  $\text{rand}()$  is uniform random number from interval  $[0, 1]$ . The maximum of velocity is called  $V_{\max}$ . If the maximum of particle's velocity is more than  $V_{\max}$ . The velocity is set to  $V_{\max}$ . The performance of each particle is assessed according to a predefined fitness function, which is related to the problem to be solved.

### B. Fast Swarm Optimization

Each PSO particle converts to GBEST, PBEST and finally trapping in local optimum. FPSO [12] proposed position generated from Cauchy mutation as an alternate to normal PSO position. The new particle position is the one with the

better fitness value. The following equations used Cauchy mutation to calculate velocity and position.

$$V_i^{(t+1)} = V_i^{(t)} \exp(\delta) \quad (3)$$

$$X_i^{(t+1)} = X_i^{(t)} + V_i^{(t+1)} \delta_1 \quad (4)$$

where  $\delta$  and  $\delta_1$  denote Cauchy random numbers.

### C. Fast Multiple Swarms Algorithm

FMPSO [12] was a Multi-Swarm that used Cauchy mutation as in FPSO. Moreover, FMPSO introduce the crossover operation in order to increase convergence rate. The crossover operation is applied to the best particle of a randomly selected swarm to each particle of another randomly selected swarm. The crossover was performed last in each PSO and Cauchy iteration of all swarm using formula (5) and (6). Amount of particles dimension that are crossover depend on crossover rate.

$$P_c = (1 - \alpha) * P_c + \alpha * Xbest_i \quad (5)$$

$$V_c = \text{rand}() * (Xbest_i - \alpha * P_c) \quad (6)$$

Where  $q_c$  is crossover rate,  $\alpha$  is a random number from interval  $[0, 1]$ ,  $P_c$  is position dimension  $d$  of particle  $x$  of random swarm and  $V_c$  is velocity of particle  $x$  of random swarm.  $Xbest_i$  is the best particle of random swarm, index is index of the best particle.

### D. Selective Crossover base on Fitness in Multi-Swarm Optimization

SFMPPO [15] suggested the selective use of crossover operation only when it improves fitness. Hence, only particles that have position generated from crossover operation with better fitness value are updated. Other particles in the swarm remain unchanged. The crossover is used when fitness improve more than some PDF percent relative to the current fitness as in formula (7).

$$PDF = \left| \frac{F - F'}{F} \right| \times 100 \quad (7)$$

Where  $F$  and  $F'$  represents fitness of previous position and new position generated from crossover operation respectively.

## III. A MODIFIED MULTI-SWARM OPTIMIZATION WITH INTERCHANGE GBEST AND PARTICLE REDISTRIBUTION

In this paper, the trapping in local optimum problem is handling by using multiple swarms. Each swarm use normal PSO update position and each one has their own GBEST. In each generation, each swarm move one by one. If a swarm trap in local optima, its particles are slightly mutated (reposition), its GBEST is copied from the next swarm if they are different otherwise GBEST is reset. After handling trapping of each swarm one by one, all swarm continue in the next generation until they reach maximum number of

generation. To check trapping in local optimum, each swarm GBEST must remain unimproved for some number of generations (TR). Pseudo code of this algorithm is shown below.

Multi-Swarm or PSO encounter trapping in local optimum. PSO traps only one point while Multi-Swarm can trap many points. In Multi-Swarm, the trapped point or GBEST can exchange GBEST of other swarms in order to distribute and search for new area. This technique can improve solution.

However, the trapped point of swarm can be same the trapped point of exchanged swarm. In this case, the trapped point of swarm should reset in order that particles in swarm search for new area. Moreover, the trapped point of each swarm usually lives near area. Searching remains in old area, so obtained solution rarely improves. In order that particles distribute wide area then they search for new areas. This paper proposed reposition technique [15] apply with Multi-Swarm when a swarm trapped in local optimum. Then GBEST of this swarm replaces GBEST of other swarms. If GBEST of this swarm equal GBEST of other swarms, GBEST of this swarm is reset.

```

Initial particles of each swarm (1)
While (termination condition ≠ true) do (2)
    Evaluate the fitness of each particle of each swarm (3)
    If fitness of each particle is better than PBEST of each swarm, (4)
        Update PBEST of each swarm (5)
    End If (6)
    If fitness of each particle is better than GBEST of each swarm, (7)
        Update GBEST of each swarm (8)
    End If (9)
    Update each particle position according to (1) and (2) of each swarm (10)
    If (times of GBEST consecutive unchanged) ≥ TR (11)
        If GBEST equals NGBEST (12)
            Reset GBEST (13)
        Else (14)
            BEST = NGBEST (15)
        End if (16)
    For j = 1 to N //The number of particles in each swarm (17)
        Reset PBEST (18)
        For d = 1 to D //Dimension (19)
            If PR > rand () then (20)
                 $particle_{i,d} = particle_{i,d} \pm (particle_{i,d} \times rand())$  (8) (21)
            End If (22)
        Next d (23)
    Next j (24)
End If (25)
End while (26)

```

Where PR is the probability of reposition, TR is the threshold of number of generation that GBEST is unchanged, and NGBEST is GBEST of neighbor.

The reposition of particle in line (8) is randomly changing some dimension of particle position to increment or decrement some random percent of the current position in the dimension.

#### IV. EXPERIMENTS AND RESULTS

##### A. Benchmark Test Functions

The proposed algorithm is tested on twenty-six well-known benchmark functions [13], [14]. These functions consist of seventeen multimodal functions from function one to seventeen. The remaining nine functions are unimodal functions. The results of MPSOIR on the benchmark functions are compared with PSO [1], [2], FMPSO [12] and SFMPSO [15].

##### B. Parameters Setting

Parameters are as follows for all experiments:  $\eta_1$  and  $\eta_2$  are both set to be 1.496180 and  $\omega = 0.729844$  as suggested by den Bergh [11]. The number of experiments of each function is 50 runs. The non-PSO parameters are as follows: parameters of FMPSO and SFMPSO are set according to suggested by the original papers [15]. Except population size set to be 90 for PSO and 30 for FMPSO, SFMPSO, MPSOIR. The number of swarms set to be 3 for FMPSO, SFMPSO, and MPSOIR. Parameters of MPSOIR, PR = 0.70, TR = 100. To guarantee fairness, the number of evaluations is set to 200000 for dimensions are more than or equal 10 dimensions. The remaining is set to 10000.

The measures of algorithm performance in the experiments are as follows: The mean best fitness value (MBF) is the mean of best fitness in the final iteration from all running (50 runs). MBF indicates the solution searching efficiency of an algorithm. All experiments benchmark functions have zero results as a minimum value. The closer the MBF to the zero point of a method, the better the method. SD is the standard deviation. SD shows the reliability of an algorithm. The less the SD is, the better the algorithm is.

##### C. The Experiment Results

From the experimental results in Table 1, MPSOIR outperforms PSO, FMPSO, and SFMPSO with regard to the reliability and quality of solutions in all test functions because its lower MBF and SD than them. MPSOIR produces the best quality solution in multimodal function. Since, MPSOIR has a technique that solves trapping problem while PSO has not any technique to solve trapping problem. MPSOIR solve trapping problem when it occur. Hence, it seldom increase evaluation calls while both FMPSO and SFMPSO prevent trapping problem by adding mutation and crossover into process of PSO. Amount of evaluation calls enhance to twice. Moreover, the evaluation calls originates mutation and crossover. It may be used to searching solution. If amount of evaluation calls equal, solving trapping of MPSOIR is better than solving trapping of FMPSO and SFMPSO.

For multimodal function, generally, the quality solution both FMPSO and SFMPSO are better than the quality solution PSO. Since, both FMPSO and SFMPSO have technique for solving trapping problem. However, both FMPSO and SFMPSO get solutions worse than PSO such as ROSENROCK, DIXON-PRICE, and SHUBERT. Since, PSO can solve these problems better than FMPSO and SFMPSO. Hence, both mutation technique and crossover technique are added into process of PSO. These techniques may be useless.

FMPSO and SFMPSO get solutions worse than PSO for all unimodal functions. In this case, PSO is not encounter trapping in local optimum. Both mutation technique and crossover technique are useless so, these techniques cause to lose evaluation calls uselessly. While PSO and MPSOIR have similar overall results, MPSOIR is not an encounter trapping. Its searching is similar to searching of PSO.

TABLE I. TABLE STYLES

Function	PSO		FMPSO		SFMPSO		MPSOIR	
	MBF	SD	MBF	SD	MBF	SD	MBF	SD
Ackley	3.7E-01	7.3E-01	1.4E-01	3.9E-01	5.0E-12	3.1E-12	1.2E-14	8.5E-15
Griewank	1.1E-02	1.0E-02	4.1E-02	7.2E-02	5.6E-03	8.7E-07	2.2E-03	4.1E-03
Rastrigin	7.7E-01	1.7E-01	2.0E-01	1.4E-01	8.3E-07	5.5E-01	2.2E-01	1.1E-01
Rosenbrock	5.2E-12	5.6E-12	1.1E-10	2.0E-10	1.0E-11	1.5E-13	6.7E-13	2.0E-12
Schwefel	2.9E-03	1.6E-03	5.2E-05	1.1E-03	1.9E-03	6.7E-05	1.5E-03	3.8E-03
Cosmic	3.1E-01	2.2E-01	2.7E-01	1.5E-01	4.0E-16	1.2E-18	0	0
Exponential	1.1E-16	3.4E-17	1.0E-16	1.5E-17	6.9E-17	4.0E-17	0	0
Levy	5.02	3.94	9.9E-01	3.19	9.1E-01	6.7E-02	1.1E-04	3.7E-13
Dixon-price	0	0	2.40	3.7E-01	2.34	5.8E-01	0	0
Michalewicz	0	0	0	0	0	0	0	0
Step	2.0E-02	1.4E-01	2.0E-01	7.2E-01	0	0	0	0
Schaffter	2.5E-03	4.2E-03	5.3E-03	3.6E-03	3.0E-03	4.2E-03	3.8E-03	4.0E-04
Holder	0	0	0	0	0	0	0	0
Beale	1.0E-12	5.0E-12	1.0E-07	5.5E-07	4.8E-12	7.3E-12	1.3E-12	1.3E-12
Shubert	0	0	2.9E-01	3.9E-01	1.4E-01	6.6E-01	0	0
Goldstein	0	0	0	0	0	0	0	0
Six-bump	0	0	0	0	0	0	0	0
Sphere	0	0	7.3E-25	0	0	0	0	0
Parallel	0	0	6.2E-24	0	0	0	0	0
Rotated	0	0	5.9E-23	1.3E-22	1.6E-15	0	0	0
Cigar	0	0	1.8E-19	1.0E-19	5.8E-42	0	0	0
Brown	0	0	3.7E-25	0	0	0	0	0
Multimod	2.8E-21	1.9E-20	3.0E-01	1.03	3.1E-15	7.8E-19	1.2E-27	0
Zakharov	1.5E-05	2.0E-05	2.4E-01	5.46	3.3E-02	4.4E-02	2.4E-09	4.0E-09
Trid	2.3E-03	1.5E-03	6.7E-02	1.7E-01	2.5E-03	1.0E-03	1.0E-03	1.2E-03
Easom	2.9E-02	1.4E-01	2.7E-01	3.5E-01	4.0E-01	3.6E-01	1.6E-02	1.1E-01

## V. CONCLUSION

This research proposes the particles redistribution and interchanging GBEST of multiple swarms when trapping in local optimum. The proposed technique can lessen the local optimum problem and find better optima than PSO, FMPSO, and SFMPSO in many of the test functions.

## REFERENCES

- [1] J. Kennedy, and R. C. Eberhart, "Particle Swarm Optimization," IEEE International Conference on Neural Networks, pp. 1942-1948, 1995.
- [2] R. C. Eberhart, and J. Kennedy, "A New Optimizer Using Particle Swarm Theory," Proceedings of the 6th International Symposium on Micro Machine and Human Science, pp. 39-43, 1995.
- [3] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," IEEE Transactions on Evolutionary Computation, vol. 8, no. 3, pp. 240-255, 2004.
- [4] J. Sun, B. Feng, and W. Xu, "Particle swarm optimization with particles having quantum behavior," in Proceedings of the IEEE Congress on Evolutionary Computation, Portland, Oregon USA, pp. 325-331, 2004.
- [5] Jing Liu, Wenbo Xu, and Jun Sun, "Quantum-behaved particle swarm optimization with mutation operator," Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence Pages: 237 - 240, 2005.
- [6] R. A. Krohling, "Gaussian particle swarm with jumps," in Proceedings of the IEEE Congress on Evolutionary Computation, Edinburgh, UK, pp. 1226-1231, 2005.
- [7] R. A. Krohling, and L. dos Santos Coelho, "PSO-E: Particle Swarm with Exponential Distribution," in Proceedings of the IEEE Congress on Evolutionary Computation, pp. 1428-1433, July 2006.
- [8] B. Al-Kazemi, and C. K. Mohan, "Multi-phase discrete particle swarm optimization," In Proc. of 4th Int. Workshop on Frontiers on Evolut. Alg., Research Triangle Park, NC, 2002.
- [9] S. Baskar, and P. N. Suganthan, "A novel concurrent particle swarm optimization," In Proc. of Cong. on Evolut. Comput., Portland, OR, pp. 792-796, 2004.
- [10] T. Peram, K. Veeramachaneni, and C. K. Mohan, "Fitness-distance ratio based particle swarm optimization," In Proc. of IEEE Swarm Intell. Symp., Indianapolis, IN, pp. 88-94, 2003.
- [11] F. van Den Bergh, "An analysis of particle swarm optimizers," Ph.D. Dissertation, University of Pretoria, 2001.
- [12] Changhe Li, Qing Zhang, Yong Liu, and Lishan Kang, "A Fast Particle Optimization Algorithm with Cauchy Mutation and Natural Selection Strategy," Proceedings of the 2nd international conference Advances in computation and intelligence 2007, Wuhan, China, pp. 334-343.
- [13] Information on <http://www.siu.ca/~ssurjano/index.html>
- [14] Information on <http://www.optima.amp.i.kyoto-u.ac.jp>
- [15] R. Chhabwot, and K. Boontee, "Selective Crossover base on Fitness in Multi Swarm Optimization," International Conference on Emerging Trends in Computer and Image Processing (ICETCIP2011) Bangkok Dec., 2011.
- [16] R. Chhabwot and K. Boontee, "A Modified Particle Swarm Optimization with Mutation and Reposition," ICIC international, 2014.



## งานวิจัยที่ได้รับการตีพิมพ์

2. Kanokporn Chengkhuntod, Boontee Kruatrachue, Kritawan Siriboon, “Improving Multi-Swarm by Slightly Mutation Particle and GBEST of Stuck Swarm Along with Randomly Selecting GBEST of other Swarm”, 21st International Computer Science and Engineering Conference 2017 (ICSEC'2017), 15-18 November 2017, Bangkok, Thailand, pp 213-216.



# The 21st International Computer Science and Engineering Conference 2017

November 15-18, 2017

Bangkok, THAILAND



CONFERENCE

# PROCEEDING

ISBN : 978-1-5386-0787-9

IEEE Catalog Number: CFP17IBE-ART

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Improving Multi-Swarm by Slightly Mutation Particle and GBEST of Stuck Swarm Along with Randomly Selecting GBEST of other Swarm

Kanokporn Chengkhuntod, \*Boontee Kruatrachue, and Kritawan Siriboon

Department of Computer Engineering, Faculty of Engineering  
King Mongkut's Institute of Technology Ladkrabang  
Bangkok, Thailand

E-mail: k.chengkhuntod@gmail.com, \*Corresponding Author: boontee.kr@kmitl.ac.th, kritawans.si@kmitl.ac.th

**Abstract**— This paper proposed another approach in handling trapping in local optimum problem of Particle Swarm Optimization (PSO) using multi-swarm. Since each swarm might trap in different local optimum, the trapped swarm restart with slightly mutation (15% of each particle attributes) along with swaying swarm by randomly use of other swarm GBEST position. In the case of all swarm trapping in the same location, the trap GBEST is also slightly mutate in the same way as particle position. This proposed technique is tested on a set of twenty-four benchmark test functions. The experimental results show that the proposed method is better than other comparing methods.

**Keywords**— Swarm intelligence; Multiple Swarm; PSO; Particle Swarm Optimization.

## 1. INTRODUCTION

Eberhart and Kennedy [1], [2] proposed Particle Swarm Optimization (PSO) in 1995 inspiration by food-search behavior and birds communication mechanism and schooling fish. PSO is used to solve the optimization problems. Advantages of PSO are faster in convergence, easy to implement and a few parameter to adept. The main of disadvantages of PSO are trapping into local optima problem and premature convergence.

To solve trapping into local optima problem and premature convergence, many researchers increased a wide variety of search. Some researchers proposed multi-swarm and exchanges information between swarm as follows.

Al-Kazemi and Mohan [3] improved PSO by grouping particles into two swarms to search independently to increase the chance of exploring in different area. There is no exchange of information between both swarms. Each swarm performs normal PSO movement. Finally, both swarm stuck in local optimal.

Baskar and Suganthan [4] also split particles into two swarms. Both swarms perform normal PSO except both use GBEST of the other swarm.

Peram, Veeramachaneni and Mohan [10] used optimization techniques the same as paper [3], but the equation of moving particles is different. The first group uses

the same updated equation of standard PSO and the second group use the Fitness-to-Distance ratio.

Changhe, Qing, Yong and Lishan [7] proposed Fast Particle Swarm Optimization (FPSO) to decrease the trapping into local optimal by using Cauchy mutation. They also extend FPSO using multi-swarm and crossover operation (FMPSO). Particle position is calculated from normal PSO movement and applied Cauchy mutation to the position the better one is select as a new particle position. So evaluation call to objective function is double comparing to normal PSO. Also, crossover operation is performing between each particle of all swarm and the GBEST of a random group in multi-swarm.

Ratanavilisagul and Kruatrachue [10] improved FMPSO [7] algorithm by adding the selection between position generate from crossover operation and current position based on their fitness.

Ratanavilisagul and Kruatrachue [11] add 5 mutations for each particle along with normal particle movement in PSO. The mutation is used to update PBEST and GBEST but not particle position. This modification improve local optimal point in comparing to PSO. The results of [11] show that these techniques can get better solution comparing to other PSO modified method in most of their benchmark functions.

Chengkhuntod, Kruatrachue and Siriboon [12] used the reposition technique to improve multi-swarm. This technique is called A Modified Multi-Swarm Optimization with Interchange GBEST and Particle Redistribution (MPSOIR). They proposed the use of neighbor GBEST and particle position mutation to sway a trapped swarm from its local optimum. The purpose of the mutation is to reposition each particle in the trapped swarm near its current position.

Zhao, S-Z., et al. [13] proposed the Dynamic Multi-Swarm Particle Swarm Optimizer with Harmony Search (DMS-PSO-HS). This algorithm added the search mechanism of HS to the Multi-Swarm PSO (DMS-PSO) which proposed by J. J. Liang and P. N. Suganthan [14]. Phuchan, Kruatrachue and Siriboon [15] also proposed Hybrid Multi-swarm with Harmony Search algorithm (HMSHS) where HS on PBEST is used to replace the worst particle in the swarm.

The previously mentioned methods still have the local optimum problem with different local optimum points. Some has better solution in some benchmark function and worse in the others. Sometimes, the optimum point is reach in some function depending on the number of particle and search space dimension (dimension of benchmark function). This paper tries to solve this trap swarm problem in the same way as in [12], where particle position of the trap swarm mutes with much less amount also with the random group of other swarm GBEST to sway the trap swarm. In case all swarm trap in the same local optima, GBEST of the trap swarm are also slightly mute in the same way as particle position. The proposed method is called MPSSOM for its use of multi-swarm and slightly mutation technique.

This proposed method is tested on a set of twenty-four benchmark test functions and compared with PSO, HMSHS, FMPSO, MPSSOM. The experimental results show that the proposed method has the best results for all the test functions.

The rest of this paper is organized as follows. Section 2 explains the standard PSO [1], [2], FPSO and FMPSO [10], HMSHS [13] and MPSSOM [12] Section 3 explains MPSSOM. Section 4 explains twenty-four benchmark test functions, describes the experiment setup and experiment results. Finally, Section 5 concludes the paper with a brief summary.

## II. RELATED WORK

### A. Particle Swarm Optimization

Particle Swarm Optimization (PSO) is inspired by the behavior of bird's flocking, flash schooling and their communication mechanism in solving optimization problems. It performs the search for optimal function in the search space by randomly distributes solutions called particle on a search space. Each particle searches to the new position by moving randomly toward its best position in its history (PBEST) and the best seen position among all other particles (GBEST) as in eq. (1) and eq. (2).  $V_i^{(t)}$  is a particle's velocity of particle  $i$  at time  $t$  and  $X_i^{(t)}$  is a particle's position of particle  $i$  at time  $t$ . Each particle starts with random position  $X_i^{(0)}$  and velocity  $V_i^{(0)}$ . The fitness value of each particle is calculated from the objective function ( $f(X^{(t)})$ ) at its current position and use to update  $pbest_i^{(t)}$  and  $gbest^{(t)}$ .

$$V_i^{(t+1)} = \omega V_i^{(t)} + c_1 \text{rand}() (pbest_i^{(t)} - X_i^{(t)}) + c_2 \text{rand}() (gbest^{(t)} - X_i^{(t)}) \quad (1)$$

$$X_i^{(t+1)} = X_i^{(t)} + V_i^{(t+1)} \quad (2)$$

$\omega$  is an inertia weight in the range  $0 \leq \omega \leq 1$ ,  $c_1$  and  $c_2$  are acceleration constants,  $\text{rand}()$  is uniform random number from interval 0 to 1. The maximum of velocity is called  $V_{\max}$ . The maximum of particle's velocity is more than  $V_{\max}$ . The velocity is set to  $V_{\max}$ . PSO is widely used due to its simplicity and fast convergence rate. Nevertheless, the main disadvantage of PSO is easy to trap into local optimum in multidimensional space.

### B. Fast Swarm Optimization

FPSO [7] is proposed to lessen the trapping in local optimum. The Cauchy muted position (3) and (4) is generated from current particle position and compared to normal PSO position update. The best fitness position is selected for the next particle position.

$$V_i' = V_i \exp(\delta) \quad (3)$$

$$X_i' = X_i + V_i' \delta_i \quad (4)$$

where  $X_i$  is represent the current positions of  $i$  particle,  $X_i$  is represent the previous positions of  $i$  particle,  $V_i$  is represent the previous velocity of  $i$  particle,  $V_i$  is represent the current velocity of  $i$  particle,  $\delta$  and  $\delta_i$  denote Cauchy random numbers.

### C. Fast Multiple Swarms Algorithm

FMPSO [7] is a multi-swarm that modified from FPSO. FMPSO add the crossover operation in order to increase convergence rate, where the best particle from a swarm is randomly selected, Next, select a swarm which is not selected previously. Then, all particles of the selected swarm in the second step executes crossover operation with the best particle in the first step according to the formula (5) and (6). Amount of dimension on particles that are executed crossover operation depend on crossover rate.

$$P_c[i] = (1 - \alpha) * P_c[i] + \alpha * P_c[i] \quad (5)$$

$$P_c[i] = \text{rand}() * (P_c[i] - \alpha * P_c[i]) \quad (6)$$

Where  $q_c$  is crossover rate,  $\alpha$  is a random number from interval [0, 1],  $P_c$  is position of particle  $P$  and  $P_c$  is velocity of particle  $P$ .

### D. A Modified Multi-Swarm Optimization with Interchange GBEST and Particle Redistribution

The research [12] split a swarm into multi-swarm (MPSSOM). Each swarm updates particle position and velocity identical to normal PSO until trapping in local optimum. Thus, if one swarm traps in local optimal, its 70 % of each particles attributes are mutate from its current position by random amount of its current value, as in formula (7). GBEST of the trap swarm is replaced by GBEST of neighbor swarms. But, If both GBEST are the same, GBEST is set from the best mutation particle in the trap swarm. And all swarm continue in the next generation until they reach maximum number of generation or trap into local optimal again. Hence, it seldom increase evaluation calls of objective function in comparing to FMPSO and SFMPSO, since the call only increase when trapping occur.

$$particle_{id} = particle_{id} \pm (particle_{id} * \text{rand}()) \quad (7)$$

### E. Hybrid Multi-swarm with Harmony Search algorithm (HMSHS)

This research proposed a new meta-heuristic algorithm, using both multi-swarm and harmony search algorithm to improve from [13] and [14]. To reduce trapping in local optimal of normal PSO, HMSHS used HS algorithm with

standard PSO to add diversity. Each swarm perform normal PSO for two consecutive iterations, then, HS create 10 new solutions from harmony memory of current PBEST of all particles and select the best one to replace the worst particle in each swarm if the new solution is better. The harmony memory is reset to PBEST each time HS performed. The experiment results show that solutions of HMSHS are better than solutions of PSO and solutions of comparative algorithms. Moreover, HMSHS gets good solutions in problem with high dimension especially in unimodal functions. However, HMSHS did not perform well in smaller dimension problem when it trap in local optimum.

### III. IMPROVING MULTI-SWARM BY SLIGHTLY MUTATION PARTICLE AND GBEST OF STUCK SWARM WITH RANDOMLY SELECTING GBEST OF OTHER SWARM

Research [12] proposed the reposition technique to a multi-swarm, when a swarm stick to a local optimum, each particle reposition from its stuck position by mutation 70% of its attributes (dimensions) using eq. (7). The mutation attribute value,  $x$ , is set to its current value decrement or increment randomly with random part of itself (new  $x$  is in the range from 0 to  $2x$ ). Moreover, the stuck swarm is sway from its stuck position by using GBEST of the neighbor swarm. In the case that, neighbor swarm also stuck in the same local optimum, the best particle from the stuck swarm after reposition process is set to GBEST.

The new proposed algorithm improves the above method in three ways. Firstly, the smaller range of mutation along with the chance of inverse sign. Secondly, the random use of neighbor GBEST of a stuck swarm instead of fixed neighbor and lastly the use of GBEST reposition when all swarm stuck in the same local optimum.

```

Initial particles position; evaluate fitness and (1)
Update PBEST, GBEST of all particles of each swarm (2)
While (termination condition ≠ true) do (3)
  Update particle position, PBEST, GBEST by eq (1) and (2) of all particles (4)
  If GBEST of swarm g unchanged for consecutive amount of time (TR) (5)
    If GBEST of all swarm unchanged for consecutive amount of time (TR) (6)
      Reset GBEST and PBEST of all particles of all swarms (7)
      For I to dimensions (8)
        Randomly all X in 15% of all dimensions by eq (8) for all particles (9)
        Randomly all GBEST in 15% of all dimensions by eq (9) (10)
      End For (11)
    End If (12)
  Else (13)
    Reset PBEST of swarm g (14)
    Randomly X of swarm g in 15% of all dimensions by eq (8) (15)
    If Fitness of swarm g is equal Fitness random from other swarm (16)
      Random other swarm, Fitness value is not equal (17)
      Change GBEST of swarm g with those other swarm (18)
      Change GBEST position of swarm g with other swarm (19)
    End If (20)
  Else If Fitness of swarm g is not equal Fitness random of other swarm (21)
    Change GBEST of swarm g with other swarm (22)
    Change GBEST position of swarm g with other swarm (23)
  End Else If (24)
End While (25)
End If (26)
End While (27)

```

When a swarm is trapped in local optimum, it's time to restart. Instead of random particle position uniformly across

the search space, this research chooses to reposition particle near its previous position using eq. (8) in only 15% of all of its attributes (the remaining attribute are unchanged). Also, the range of mutation attribute value,  $x$ , is within  $\pm 15\%$  of  $x$  (new  $x$  is in the range from 0.85 to 1.15 $x$ ). In addition,  $x$  can flip sign with the probability of 0.05, so that each particle has chance to reverse its sign in only 5% percent of previously random 15% particle's attributes.

To sway the new reposition swarm from its previous local optimum, other group GBEST that is not locate in the same local optimum is select randomly and set to GBEST of the new reposition swarm. This extends the search from the vicinity of the stuck swarm to the randomly chosen GBEST position of the other group instead of fixed to the same other group as in previous algorithm. This change improves the searching performance in previous method. There are many cases that, all swarm stuck in the same local optimum, the same reposition technique is used with the GBEST. That is, only 15% of GBEST attributes of each stuck swarm are mutated using eq. (9) and 5% of those reverse their signs.

$$X_{id} = \pm (X_{id} \times (0.85 + 0.3 \times \text{rand} ())) \quad (8)$$

$$GBEST_{id} = \pm (GBEST_{id} \times (0.85 + 0.3 \times \text{rand} ())) \quad (9)$$

TABLE I. DETAILS OF BENCHMARK TEST FUNCTIONS

Name Function	Search Space $[X_{min}, X_{max}]$	Objective Function Value	Dim	$f(X^*)=0$ $X^*$
ACKLEY	$x \in [-32, 768, 32, 768]^D$	0	100	(0,...,0)
GRUFWANK	$x \in [-300, 300]^D$	0	100	(0,...,0)
RASTRIGIN	$x \in [-5, 12.5, 5, 12]^D$	0	100	(0,...,0)
ROSENBROCK	$x \in [-2, 048, 2, 048]^D$	0	100	(1,...,1)
SCHWEFEL	$x \in [-500, 500]^D$	0	100	(420.9687, ..., 420.9687)
FOSSIM	$x \in [-1, 1]^D$	0	100	(0,...,0)
EXPONENTIAL	$x \in [-5, 12.5, 12]^D$	0	100	(0,...,0)
LEVY	$x \in [-5, 12.5, 5, 12]^D$	0	100	(1,...,1)
MICHAELWICZ	$x \in [-65, 536, 65, 536]^D$	0	100	(2.20, 1.57)
DIDSON-PRICE	$x \in [0, P]^D$	0	10	(0,...,0)
STEP	$x \in [-100, 100]^D$	0	100	(0.5, ..., 0.5)
SCHAFER	$x \in [-100, 100]^D$	0	2	(0,...,0)
BOEDER	$x \in [-10, 10]^D$	0	2	(-8.05502, 9.66459)
BEALE	$x \in [-4, 5, 4, 5]^D$	0	2	(3, 0.5)
SOUTHBT	$x \in [-10, 10]^D$	0	2	(0,...,0)
GOLDSTEIN	$x \in [-2, 2]^D, f(X^*) = 3$	0	2	(0,-1)
SIX-HUMP	$x \in [-2, 2]^D$	0	2	(0.0898, -0.7126)
SPHERE	$x \in [-5, 12.5, 12]^D$	0	100	(0,...,0)
PARALLEL	$x \in [-5, 12.5, 12]^D$	0	100	(0,...,0)
ROTATED	$x \in [-65, 536, 65, 536]^D$	0	100	(0,...,0)
CIGAR	$x \in [-10, 10]^D$	0	100	(0,...,0)
BROWN	$x \in [-1, 4]^D$	0	100	(0,...,0)
MULTIMOD	$x \in [-10, 10]^D$	0	100	(0,...,0)
ZAKHAROV	$x \in [-5, 10]^D$	0	100	(0,...,0)

### IV. EXPERIMENTS AND RESULTS

This technique is tested a set of twenty-four benchmark test functions Table 1 [8], [9]. These functions consist of seventeen multimodal functions from function one to

seventeen. The remaining nine functions are unimodal functions. The results of MPSOSM on the benchmark functions are compared with PSO [1], [2], HMSHS [15], FMPSO [10], and MPSOIR [12].

TABLE II. COMPARATIVE RESULTS OF PSO, HMSHS, FMPSO, MPSOIR, AND MPSOSM ON BENCHMARK TEST FUNCTIONS

Function	PSO		HMSHS		FMPSO		MPSOIR		MPSOSM	
	MBF	MEC	MBF	MEC	MBF	MEC	MBF	MEC	MBF	MEC
ACKLEY	0.207	1345874	4.29 E-14	587270	1.68 E-14	7568312	1.40 E-10	3612074	0	6995672
BISEWALK	0.012	2327160	0	238474	5.48 E-6	3232207	0	751510	0	141892
RASTRIN	70.60	8707330	30.84	8763883	93.16	9219015	0	3887838	0	5817055
ROSENROCK	3.90 E-30	708648	4.189E+5	440762	4.51 E-28	9996642	3.80 E-24	1660671	2.50 E-37	4443511
SCHWEFEL	6655	2092964	18714	6554243	16883	5983017	6413	9990979	5009	8879902
CORNE	0.289	7167705	3.55 E-15	499631	0.0229585	2223066	0	1233139	0	1043016
EXPONENTIAL	1.13 E-16	1009027	1	499631	3.18 E-10	6374172	0	1069235	0	1776347
LEWY	4.706	3708846	2.50 E-16	1287432	0.786387	9931556	1.00 E-06	5163376	1.62 E-29	7833766
MICHAELWIG	0	1325	0	9696271	0	1023	0	245	0	221
TRIGON-POLY	2.24 E-06	11693	0	568617	0	3060	0	8076	0	11671
STEP	0.08	8994157	0	32334	0	438669	0	153362	0	154788
SCHAFER	0.000972	22307	0	13586	9.67 E-06	32103	3.88 E-10	34625	0	22187
HOLODER	0	31101	0	68304	0	213	0	808	1.80 E-07	5999
BEALE	0	31592	-4.368	51486	3.62 E-22	26685	0	32788	0	33972
SHEPHERD	0	5838	0	110194	0.00904	43788	0	9618	0	3226
GOLDSTEIN	0	12609	0	62774	0	17912	0	19512	0	12909
SIX-HUMP	0	2205	0	14721	0	1158	0	2565	0	2098
SPHERE	2.60 E-06	199737	4.97 E-24	561392	1.1261	194995	6.60 E-06	230939	5.05 E-19	199440
PARALLEL	3.66 E-54	199943	1.97 E-33	5026474	31.357	199630	2.06 E-05	200024	1.98 E-08	199916
ROTATED	2.62 E-35	199930	2.45 E-33	7284829	4742.8	199749	5.04402	199971	2.56488	20006
CLEAR	3.29 E-91	199884	1.81 E-33	698641	37415	0.0432	138.9	200031	131	199844
BROWN	4.62 E-57	199912	0	520381	0.410	199704	3.80 E-06	200067	2.57 E-07	199778
MULTIMOD	2.92 E-13	180334	2.53 E-15	9400008	9.0	200142	1.20 E-08	186056	2.00 E-09	151000
ZAKHAROV	3.46 E-06	199344	2.77 E-37	9769604	1064.3	165346	6.07523	199907	614.503	199355

Parameters, for all experiments, are as follows:  $\eta_1$  and  $\eta_2$  are both 1.496180 and  $\omega = 0.729844$  as suggested by Den Bergh [11]. The number of particles is 90 particles for standard PSO, HMSHS, MPSOIR and MPSOSM algorithm. MPSOSM has 3 sub-swarms with 30 particles each. The number of experiments of each function is 50 runs. To guarantee fairness among all algorithms, the maximum number of evaluations is set to 10,000,000 for dimensions larger or equal to 10 dimensions and 10,000 for dimension less than 10. PR is set 0.15, TR and TRR are set 100. EC is Evaluation Call.

The performance measurement of each algorithm in the experiments is as follows: The mean best fitness value (MBF) is the mean value of the best fitness function from all running (50 runs). MBF indicates the solution searching efficiency of an algorithm and can be used to compare which algorithm trap in better local optimal. All experiments benchmark test functions have zero results as a minimum value. The closer the MBF to the zero point of a method, the better the method. The mean evaluation call (MEC) is the mean number of evaluation function call that achieves the best fitness function value in MBF. This number indicates the convergence rate.

From Table 2, MPSOSM are better than PSO, HMSHS, FMPSO, and MPSOIR in almost all multimodal test functions

except Holder and Rosebrock function. This mainly due to MPSOSM handling of trapping in local optima situation where each particle positions of the trapped swarm is slightly change and also its GBEST position is set to other swarms. In case of all swarm trapped, both particle position and GBEST are slightly mutation. Even though reposition improves optimal, it also increases calculation time in most cases. The performance of unimodal function is worse than HMSHS due to too early reposition. Therefore, particle has many jumping out of local optimal before improving optimal and also use many more evaluation calls.

## V. CONCLUSION

The use of less amount of mutation in muted particle position and muted GBEST for multi-swarm improve both the fitness of particle and its calculation time (number of evaluation function call) in comparing to previous works (MPSOIR and FMPSO). However, even though its performance in multimodal functions is better, its unimodal performance is lower than those of normal PSO and PSO hybrid with harmony search.

## REFERENCES

- [1] R. C. Eberhart, and J. Kennedy, "Particle Swarm Optimization," IEEE International Conference on Neural Networks, pp. 1942-1948, 1995.
- [2] R. C. Eberhart, and J. Kennedy, "A New Optimizer Using Particle Swarm Theory," Proceedings of the 6th International Symposium on Micro Machine and Human Science, pp. 39-43, 1995.
- [3] B. Al-Kazemi, and C. K. Mohan, "Multi-phase discrete particle swarm optimization," In Proc. of 4th Int. Workshop on Frontiers on Evolut. Alg., Research Triangle Park, NC, 2002.
- [4] S. Baskar, and P. N. Suganthan, "A novel concurrent particle swarm optimization," In Proc. of Cong. on Evolut. Comput., Portland, OR, pp. 792-796, 2004.
- [5] T. Peram, K. Veeramachaneni, and C. K. Mohan, "Fitness-distance ratio based particle swarm optimization," In Proc. of IEEE Swarm Intell. Symp., Indianapolis, IN, pp. 88-94, 2003.
- [6] F. van Den Bergh, "An analysis of particle swarm optimizers," Ph.D. Dissertation, University of Pretoria, 2001.
- [7] Changhe Li, Qing Zhang, Yong Liu, and Lishan Kang, "A Fast Particle Optimization Algorithm with Cauchy Mutation and Natural Selection Strategy," Proceedings of the 2nd international conference Advances in computation and Intelligence 2007, Wuhan, China, pp. 334-343.
- [8] Information on <http://www.sfu.ca/~ssurjano/index.html/>
- [9] Information on <http://www-optima.amp.i.kyoto-u.ac.jp/>
- [10] C. Ratanavilaisagul, and B. Krautrachue, "Selective Crossover base on Fitness in Multi Swarm Optimization," International Conference on Emerging Trends in Computer and Image Processing (ICETCIP2011) Bangkok Dec., 2011.
- [11] C. Ratanavilaisagul, and B. Krautrachue, "A Modified Particle Swarm Optimization with Mutation and Reposition," ICIC international, 2014.
- [12] K. Chengkhuntod, B. Krautrachue, and K. Siriboon, "A Modified Multi-Swarm Optimization with Interchange GBEST and Particle Redistribution," icocon, 2017.
- [13] Zhao, S-Z., et al. "Dynamic multi-swarm particle swarm optimizer with harmony search," *Expert Systems with Applications* 38.4 (2015): 3735-3742.
- [14] Liang, Jane-Jing, and Ponnuthurai Nagarathnam Suganthan, "Dynamic multi-swarm particle swarm optimizer with local search," *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, Vol. 1, IEEE, 2005.
- [15] W. Phuchan, B. Krautrachue, and K. Siriboon, "Hybrid Multi-swarm with Harmony Search algorithm," ECTI-CON, 2017.

## ภาคผนวก ค

### การเปลี่ยนเปอร์เซ็นต์การกระจายตำแหน่งของอนุภาค

ในภาคผนวกนี้จะดำเนินการทดลองเพื่อทดสอบจำนวนมิติที่เปลี่ยนเมื่อมีการกระจายตำแหน่งของอนุภาคว่าจะเปลี่ยนเป็นกี่เปอร์เซ็นต์ของจำนวนมิติทั้งหมด การเลือกใช้เปอร์เซ็นต์การกระจายตำแหน่งของอนุภาคของวิธีที่นำเสนอ (MPSOSM) มีเปอร์เซ็นต์ของ  $x$  เป็น 20 เปอร์เซ็นต์ และ GBEST เป็น 30 เปอร์เซ็นต์ นั้น สามารถเพิ่มประสิทธิภาพการค้นหาของ PSO ได้อย่างมีประสิทธิภาพโดยเปอร์เซ็นต์การกระจายตำแหน่งของอนุภาคเหล่านี้ถูกใช้ในการทดลองในบทที่ 5 และภาคผนวก ง

ปัญหาของการกระจายตำแหน่งของอนุภาคหรือรีโพซิชัน (Reposition) เกิดจากเมื่ออนุภาคของ MPSO มีการเคลื่อนที่ในระยะเวลาหนึ่ง อนุภาคทั้งหมดจะเคลื่อนเข้าหาตำแหน่งที่ดีที่สุดจนติดในจุดสุดสัมพัทธ์ ดังนั้นจึงต้องมีการเพิ่มกระบวนการต่างเข้าไปเพื่อให้อนุภาคมีการกระจายตัวออก โดยที่การรีโพซิชันนี้จะถูกกำหนดตามกำหนดของรีโพซิชัน (Threshold of Reposition หรือ TR)

สำหรับการทดลองนั้นได้ทำการเปลี่ยนเปอร์เซ็นต์การกระจายตำแหน่งของอนุภาคทั้งหมดหรือเปอร์เซ็นต์ในการรีโพซิชัน  $x$  ของสมการ (4.2) และตำแหน่งของอนุภาคที่ดีที่สุด (GBEST) ของสมการ (4.3) โดยวิธีที่นำเสนอนี้จะทำเมื่ออนุภาคแต่ละกลุ่มติดในจุดสุดสัมพัทธ์ที่ต่างกัน จะทำการกระจายตำแหน่งของอนุภาค  $x$  ร่วมกับการเปลี่ยนตำแหน่งของ GBEST ให้เท่ากับ GBEST ของกลุ่มข้างเคียงที่ได้จากการสุ่มที่ไม่ได้ติดในจุดสุดสัมพัทธ์เดียวกันนอกจากนี้จะทำเมื่อทุกกลุ่มติดในจุดสุดสัมพัทธ์เดียวกัน จึงเปลี่ยนเปอร์เซ็นต์การกระจาย  $x$  และ GBEST อีกครั้ง

สำหรับรายละเอียดขั้นตอนวิธีที่นำมาทดลองและรายละเอียดของฟังก์ชันมาตรฐานถูกกล่าวไว้ในบทที่ 5 โดยทดลองกับฟังก์ชันมาตรฐานทั้งหมด

#### ค.1 การเปลี่ยนเปอร์เซ็นต์การกระจายตำแหน่งของอนุภาค

โดยตาราง ค.1 ได้ใช้พารามิเตอร์ตามตาราง 5.1 และรอบของการค้นหาคาดตามตาราง 5.2 ได้ใช้จำนวนรอบการค้นหา 50 รอบและได้ทำการเปลี่ยนเปอร์เซ็นต์การกระจายตำแหน่งของอนุภาค  $x$  และ GBEST เป็น 5%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% และ 100% เท่ากัน โดยได้เปรียบเทียบกับเปอร์เซ็นต์ที่ได้ตามวิธี MPSOSM ที่ได้นำเสนอ ตามตาราง ค.1 แสดงค่าเฉลี่ยของการเดาคา ตอย (MEC) และ ค่าเฉลี่ยของค ทอบที่ที่ดีที่สุด (MBF)

ตารางที่ ค.1 ผลการทดลอง MPSOSM กับการเปลี่ยนเปอร์เซ็นต์การกระจายตำแหน่ง

SHIFT	5%		10%		20%		MPSOSM	
	MEC	MBF	MEC	MBF	MEC	MBF	MEC	MBF
ACKLEY	8,458,584	0	7,161,593	0	3,698,141	0	5,272,422	0
GRIEWANK	763,164	0	814,260	0	800,831	0	686,922	0
RASTRIGIN	4,671,041	0	4,885,961	0	5,842,541	0	4,222,824	0
ROSENBROCK	10,000,050	4.18E-26	9,999,972	3.83E-26	10,000,013	5.41E-26	9,999,966	3.94E-29
SCHWEFEL	10,000,040	1776.29	9,999,943	3837.24	10,000,042	3715.98	9,999,281	3789.74
COSINE MIXTURE	1,224,546	0	1,171,671	0	1,000,379	0	1,295,051	0
EXPONENTIAL	1,953,340	0	1,502,104	0	921,381	0	988,408	0
LEVY	10,000,067	5.29E-30	10,000,017	9.57E-30	9,999,934	1.95E-29	8,917,597	0
MICHALEWICZ	97,093	0	144,753	0	150,393	0	172,429	0
DIXON-PRICE	8,886	0	9,966	0	8,090	0	11,587	0
STEP	194,228	0	145,139	0	146,630	0	175,541	0
SCHAFFER	22,060	0	214,955	0	20,228	0	66,789	0
HOLDER	13,752	0	9,030	0	7,490	0	6,634	0
BEALE	34,363	0	32,386	0	31,112	0	32,416	0
SHUBERT	7,246	0	6,827	0	5,114	0	4,689	0
GOLDSTEIN-PRICE	10,775	0	11,640	0	11,014	0	12,008	0
SIX-HUMP CAMEL	2,397	0	2,671	0	2,073	0	2,497	0
SPHERE	10,000,066	1.10E-251	10,000,078	1.91E-253	10,000,021	1.36E-252	10,000,002	0
PARALLEL	10,000,070	2.32E-255	10,000,074	6.53E-248	10,000,056	9.52E-251	10,000,076	2.9643e-323
ROTATED	10,000,042	1.23E-258	10,000,075	2.95E-259	10,000,068	1.94E-262	10,000,002	1.4822e-323
CIGAR	10,000,042	1.41E-243	10,000,071	1.19E-245	10,000,026	2.33E-244	10,000,077	3.4584e-323
BROWN	10,000,055	3.50E-260	10,000,074	4.05E-257	9,999,997	2.96E-250	10,000,073	6.9169e-323
MULTIMOD	9,999,969	6.34E-95	10,000,080	1.44E-96	10,000,025	2.66E-93	10,000,061	6.93E-272
ZAKHAROV	9,999,152	1.53E-20	10,000,057	1.09E-18	9,999,830	1.72E-17	9,999,811	1.39E-99
TRID	161,340	0	140,530	0	236,069	0	163,072	0
EASOM	19,828	0	17,106	0	21,137	0	17,965	0
EGGHOLDER	9,991,873	65.0611	9,994,866	0.078674	9,979,307	0.0146668	10,000,079	0

ตารางที่ ค.1 ผลการทดลอง MPSOSM กับการเปลี่ยนเปอร์เซ็นต์การกระจายตำแหน่ง (ต่อ)

SHIFT	30%		40%		50%		60%	
	MEC	MBF	MEC	MBF	MEC	MBF	MEC	MBF
ACKLEY	3,008,954	0	2,765,127	0	2,927,695	0	2,526,824	0
GRIEWANK	811,591	0	694,958	0	802,940	0	881,464	0
RASTRIGIN	8,148,623	0	7,985,174	0	9,129,860	0	9,153,346	0
ROSENBROCK	10,000,038	1.35E-26	10,000,028	1.22E-26	9,999,863	2.92E-26	9,999,971	2.96E-26
SCHWEFEL	9,999,991	4737.36	9,999,907	4026.61	9,999,979	5329.44	10,000,068	5823.04
COSINE MIXTURE	1,110,587	0	863,894	0	809,679	0	890,930	0
EXPONENTIAL	759,619	0	757,269	0	725,480	0	732,435	0
LEVY	10,000,009	3.22E-29	10,000,012	1.63E-29	9,999,913	1.63E-29	10,000,047	0.0895283
MICHALEWICZ	192,415	0	196,489	0	201,615	0	388,098	0
DIXON-PRICE	9,810	0	10,603	0	9,540	0	8,480	0
STEP	201,615	0	166,379	0	148,339	0	154,324	0
SCHAFFER	52,542	0	18,984	0	20,635	0	63,157	0
HOLDER	47,777	0	9,580	0	22,162	0	7,352	0
BEALE	34,052	0	33,219	0	35,142	0	32,249	0
SHUBERT	7,056	0	5,171	0	7,244	0	6,200	0

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SHIFT	30%		40%		50%		60%	
	MEC	MBF	MEC	MBF	MEC	MBF	MEC	MBF
GOLDSTEIN-PRICE	11,241	0	13,583	0	13,338	0	14,130	0
SIX-HUMP CAMEL	2,110	0	2,842	0	1,848	0	2,236	0
SPHERE	10,000,072	5.41E-260	10,000,051	6.36E-252	9,999,947	1.46E-253	10,000,067	1.96E-249
PARALLEL	9,999,948	1.10E-251	10,000,049	5.29E-251	10,000,080	4.39E-251	10,000,052	2.33E-254
ROTATED	10,000,039	5.03E-251	10,000,024	5.14E-257	9,999,998	7.18E-258	10,000,035	1.49E-258
CIGAR	10,000,069	2.26E-249	9,999,943	2.46E-245	10,000,038	7.15E-249	10,000,073	5.13E-241
BROWN	10,000,079	2.30E-254	10,000,070	2.64E-252	10,000,057	1.46E-252	10,000,054	1.04E-251
MULTIMOD	10,000,009	1.16E-94	10,000,000	3.11E-102	10,000,009	4.49E-97	10,000,047	1.86E-96
ZAKHAROV	9,999,205	8.61E-19	9,999,939	5.88E-18	9,999,696	8.20E-18	10,000,056	5.28E-16
TRID	984,872	0	3,907,436	0	9,999,980	1.82E-12	924,961	1.82E-12
EASOM	20,239	0	18,853	0	19,821	0	18,957	0
EGGHOLDER	9,999,852	0.0117833	9,999,692	0.00248663	9,998,571	0.0149627	9,999,817	0.0113796

ตารางที่ ค.1 ผลการทดลองMPSOSM กับการเปลี่ยนเปอร์เซ็นต์การกระจายตำแหน่ง (ต่อ)

SHIFT	70%		80%		90%		100%	
	MEC	MBF	MEC	MBF	MEC	MBF	MEC	MBF
ACKLEY	2,293,647	0	2,952,911	0	1,872,961	0	2,112,690	0
GRIEWANK	729,833	0	735,020	0	734,483	0	684,701	0
RASTRIGIN	5,856,807	0	8,767,723	0	10,000,068	53.7278	8,873,637	0
ROSENBROCK	10,000,015	1.54E-26	9,999,940	1.92E-26	10,000,065	8.68E-27	10,000,014	9.38E-27
SCHWEFEL	9,999,959	5428.14	9,999,992	4710.88	10,000,031	5329.44	10,000,055	5092.56
COSINE MIXTURE	867,375	0	819,946	0	1,008,128	0	783,873	0
EXPONENTIAL	680,531	0	646,564	0	695,053	0	699,130	0
LEVY	10,000,014	5.91E-29	10,000,020	2.97E-29	9,999,942	2.36E-17	9,999,980	1.15E-27
MICHALEWICZ	164,785	0	32,567	0	30,326	0	42,615	0
DIXON-PRICE	10,262	0	14,573	0	9,090	0	9,084	0
STEP	146,936	0	146,939	0	129,479	0	141,782	0
SCHAFFER	20,289	0	21,055	0	43,375	0	17,172	0
HOLDER	18,878	0	6,681	0	7,988	0	47,910	0
BEALE	34,270	0	32,818	0	35,742	0	32,205	0
SHUBERT	5,852	0	7,008	0	5,850	0	5,700	0
GOLDSTEIN-PRICE	12,017	0	12,464	0	10,026	0	11,683	0
SIX-HUMP CAMEL	2,404	0	2,167	0	2,375	0	2,250	0
SPHERE	10,000,076	1.14E-249	10,000,003	7.36E-246	9,999,982	6.04E-258	10,000,011	7.99E-248
PARALLEL	10,000,071	1.42E-252	10,000,079	2.26E-250	10,000,039	3.45E-247	10,000,030	2.12E-247
ROTATED	10,000,009	4.69E-256	9,999,994	1.94E-251	10,000,044	4.64E-255	10,000,021	2.86E-263
CIGAR	10,000,062	2.04E-248	9,999,931	4.03E-252	10,000,068	5.05E-250	10,000,077	5.53E-242
BROWN	10,000,075	3.76E-261	10,000,009	5.68E-252	10,000,072	4.90E-258	10,000,058	7.32E-254
MULTIMOD	10,000,032	1.05E-98	10,000,058	2.79E-96	10,000,040	4.58E-98	9,999,969	3.61E-94
ZAKHAROV	9,999,514	5.30E-19	9,999,570	1.80E-19	9,999,593	9.01E-21	9,999,425	1.87E-17
TRID	10,000,079	1.82E-12	2,076,681	9.09E-13	10,000,056	9.09E-13	9,996,417	1.82E-12
EASOM	16,800	0	17,992	0	15,982	0	19,520	0
EGGHOLDER	9,999,704	0.0425524	9,999,166	0.0080736	9,996,353	0.0101539	10,000,007	0.00158414

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตาราง ค.1 จะเห็นว่าเปอร์เซ็นต์การกระจายนั้นมีผลต่อค่าเฉลี่ยของการเดาคาตอบ (MEC) และ ค่าเฉลี่ยของคาตอบที่ดีที่สุด (MBF) โดยจะเห็นได้จากค่า MEC ที่ได้แตกต่างกันเมื่อมีเปอร์เซ็นต์การกระจายที่ต่างกัน แต่ค่า MBF ของ วิธี MPSOSM จะมีค่าคาตอบเป็น 0 มากที่สุด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ง

### การเลื่อนตำแหน่งโดเมนของฟังก์ชันมาตรฐาน

ในภาคผนวกนี้ได้ดำเนินการทดลองโดยการเปลี่ยนตำแหน่งโดเมน (Domain) ของฟังก์ชันมาตรฐานเพื่อเปลี่ยนค่าคำตอบของฟังก์ชันไปยังตำแหน่งอื่นที่ไม่ใช่  $(0, \dots, 0)$  โดยทดสอบกับวิธีที่ได้นำเสนอ (MPSOSM) กลับพบว่าวิธีที่ได้นำเสนอนี้ไม่เหมาะกับการหาคำตอบของฟังก์ชันที่มีคำตอบที่อยู่ตำแหน่ง  $(0, \dots, 0)$  รวมถึงวิธีอื่นที่นำมาใช้เปรียบเทียบกับยังไม่เหมาะกับการเปลี่ยนโดเมนของฟังก์ชันไปยังจุดที่ไม่ใช่  $(0, \dots, 0)$

ตารางที่ ง.1 สรุปสมบัติของฟังก์ชันคณิตศาสตร์ที่ใช้ในการทดลอง

ชื่อฟังก์ชัน	ขอบเขต $[X_{\max}, X_{\min}]$	จุดที่ต่ำ ที่สุด	จำนวน มิติ	คุณสมบัติ	$f(X^*) = 0$ $X^*$	$V_{\max} \times V_{\min}$	$Eva_{\max}$
ACKLEY	$x \in [-32.768, 32.768]^n$	$f(x) = 0$	100	Multimodal	$(0, \dots, 0)$	$\pm 32.768$	10,000,000
GRIEWANK	$x \in [-300, 300]^n$	$f(x) = 0$	100	Multimodal	$(0, \dots, 0)$	$\pm 300$	10,000,000
RASTRIGIN	$x \in [-5.12, 5.12]^n$	$f(x) = 0$	100	Multimodal	$(0, \dots, 0)$	$\pm 5.12$	10,000,000
ROSENBROCK	$x \in [-2.048, 2.048]^n$	$f(x) = 0$	100	Multimodal	$(1, \dots, 1)$	$\pm 2.048$	10,000,000
SCHWEFEL	$x \in [-500, 500]^n$	$f(x) = 0$	100	Multimodal	$(420.9687, \dots, 420.9687)$	$\pm 500$	10,000,000
COSINE MIXTURE	$x \in [-1, 1]^n$	$f(x) = 0$	100	Multimodal	$(0, \dots, 0)$	$\pm 1$	10,000,000
EXPONENTIAL	$x \in [-5.12, 5.12]^n$	$f(x) = 0$	100	Multimodal	$(0, \dots, 0)$	$\pm 1$	10,000,000
LEVY	$x \in [-5.12, 5.12]^n$	$f(x) = 0$	100	Multimodal	$(1, \dots, 1)$	$\pm 10$	10,000,000
MICHALEWICZ	$x \in [-65.536, 65.536]^n$	$f(x) = 0$	100	Multimodal	$(2.20, 1.57)$	$\pm 3.14$	10,000,000
DIXON-PRICE	$x \in [0, PI]^n$	$f(x) = 0$	10	Multimodal	$(0, \dots, 0)$	$\pm 100$	10,000,000
STEP	$x \in [-100, 100]^n$	$f(x) = 0$	100	Multimodal	$(0.5, \dots, 0.5)$	$\pm 100$	10,000,000
SCHAFFER	$x \in [-100, 100]^n$	$f(x) = 0$	2	Multimodal	$(0, \dots, 0)$	$\pm 10$	10,000,000
HOLDER	$x \in [-10, 10]^n$	$f(x) = 0$	2	Multimodal	$(-8.05502, 9.66459)$	$\pm 4.5$	10,000,000
BEALE	$x \in [-4.5, 4.5]^n$	$f(x) = 0$	2	Multimodal	$(3, 0.5)$	$\pm 10$	10,000,000
SHUBERT	$x \in [-10, 10]^n$	$f(x) = 0$	2	Multimodal	$(0, \dots, 0)$	$\pm 2$	10,000,000
GOLDSTEIN- PRICE	$x \in [-2, 2]^n$ * $f(X^*) = 3$	$f(x) = 0$	2	Multimodal	$(0, -1)$	$\pm 100$	10,000,000
TRID	$x \in [-100, 100]^n$	$f(x) = 0$	10	Multimodal	$(-d^2, d^2)$	$\pm 100$	10,000,000
SIX-HUMP CAMEL	$x \in [-2, 2]^n$	$f(x) = 0$	2	Multimodal	$(0.0898, -0.7126)$	$\pm 5.12$	10,000,000
SPHERE	$x \in [-5.12, 5.12]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 5.12$	200,000
PARALLEL	$x \in [-5.12, 5.12]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 65.536$	200,000
ROTATED	$x \in [-65.536, 65.536]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 10$	200,000
CIGAR	$x \in [-10, 10]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 1$	200,000
BROWN	$x \in [-1, 4]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 10$	500,000
MULTIMOD	$x \in [-10, 10]^n$	$f(x) = 0$	100	Unimodal	$(0, \dots, 0)$	$\pm 5.12$	500,000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

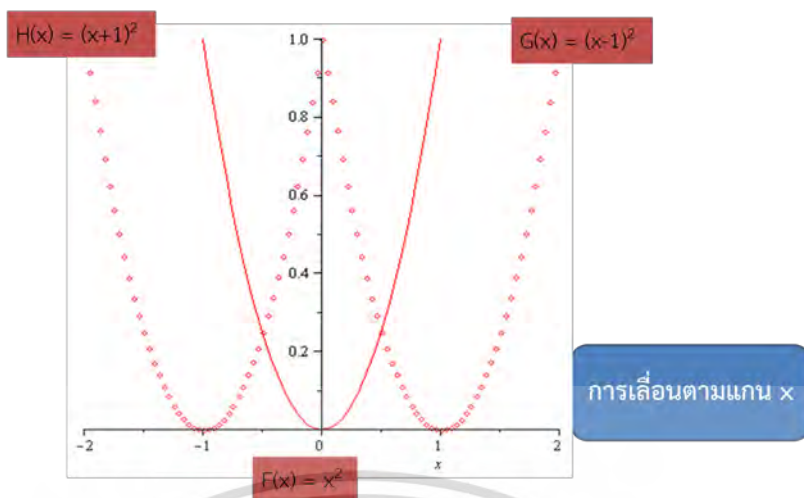
ชื่อฟังก์ชัน	ขอบเขต [ $X_{\max}, X_{\min}$ ]	จุดที่ดีที่สุด	จำนวนมิติ	คุณสมบัติ	$f(X^*)=0$ $X^*$	$V_{\max}, V_{\min}$	$Eva_{\max}$
ZAKHAROV	$x \in [-5, 10]^n$	$f(x) = 0$	100	Unimodal	(0, ..., 0)	$\pm 100$	200,000
EASOM	$x \in [-100, 100]^n$	$f(x) = 0$	2	Unimodal	(0, ..., 0)	$\pm 100$	200,000
EGGHOLDER	$x \in [-512, 512]^n$	$f(x) = 0$	2	Unimodal	(512, 404.2319)	$\pm 100$	200,000

ตาราง ง.1 อธิบายได้ว่า สำหรับการทดลองนี้ได้มีการเปลี่ยนตำแหน่งโดเมนของอนุภาคโดยการเพิ่มค่าให้กับตำแหน่งอนุภาคในแต่ละมิติ  $x^*$  ที่ทำให้  $f(X^*) = 0$  โดยจะเขียนในรูป  $X^* = \{x_1, x_2, \dots, x_n\}$  สำหรับขอบเขตของการค้นหาแสดงในช่วง  $x_{lower} < x_i < x_{upper}$  ความซับซ้อนของฟังก์ชันจะถูกกำหนดได้จากจำนวนมิติ สำหรับการทดลองนี้ใช้จำนวนมิติสูงสุดที่ 100 มิติตามตารางซึ่งมีความซับซ้อนสูงเพื่อใช้ในการเปรียบเทียบประสิทธิภาพของขั้นตอนวิธีที่ใช้ค้นหาค่าการหาจุดที่ดีที่สุด

### ง.1 การเลื่อนตำแหน่งโดเมนของฟังก์ชันมาตรฐาน

สำหรับการทดลองนี้ได้ใช้ฟังก์ชันมาตรฐานมาทดสอบประสิทธิภาพของ PSO และนำมาใช้เปรียบเทียบกับกอริทึมที่มีความแตกต่างกัน โดยการหาค่าหรือคำตอบที่ดีที่สุด  $f(x)$  ที่ได้จากการแทนค่า  $x^*$  หรือตำแหน่งอนุภาคในแต่ละมิติภายในฟังก์ชันโดยจะเขียนในรูป  $X^* = \{x_1, x_2, \dots, x_n\}$  โดยที่ความซับซ้อนของฟังก์ชันจะถูกกำหนดด้วยจำนวนมิติ ถูกจำกัดในขอบเขตการค้นหาแสดงในช่วง  $x_{lower} < x_i < x_{upper}$  ความซับซ้อนของฟังก์ชันจะถูกกำหนดได้จากจำนวนมิติของอนุภาค

สำหรับการเลื่อนโดเมนของฟังก์ชันมาตรฐานนั้น อธิบายได้ว่าเมื่อกำหนดให้  $y = f(x)$  มีจุดคำตอบอยู่ที่ 0 ดังนั้น เมื่อ  $y = f(x-k)$  เป็นการเลื่อนกราฟของ  $y = f(x)$  ไปทางขวาตามแนวแกน X เป็นจำนวน k หน่วย และ  $y = f(x+k)$  เป็นการเลื่อนกราฟของ  $y = f(x)$  ไปทางซ้ายตามแนวแกน X เป็นจำนวน k หน่วยโดยดูได้จากรูป ง.1



รูปที่ ง.1 กราฟแสดงการเลื่อนตำแหน่งโดเมนของฟังก์ชันตามแนวแกน X

จะกล่าวได้ว่าการคำนวณหาค่าความเหมาะสม (Fitness) ของอนุภาคแต่ละตัวของวิธี PSO จะถูกคำนวณด้วยตำแหน่งของอนุภาคในแต่ละมิติโดยจะเขียนในรูป  $X^* = \{x_1, x_2, \dots, x_n\}$  ก่อนจะนำไปเปรียบเทียบเพื่อหาตำแหน่งของอนุภาคที่ดีที่สุดจากค่าความเหมาะสมที่ใกล้เคียงหรือตรงกับคำตอบของฟังก์ชันมาตรฐาน  $f(x)$  ในการทดลองนี้พวกเราจึงได้ทำการเพิ่มค่า  $k$  เข้าไปในตำแหน่งของอนุภาคในแต่ละมิติเพื่อเลื่อนตำแหน่งโดเมนของฟังก์ชันจากเดิมออกไป  $k$  หน่วย โดยจะเขียนในรูปของ  $X^* = \{x_1 + k, x_2 + k, \dots, x_n + k\}$  ดังนั้น ถ้าคำตอบเดิมเคยอยู่ที่ตำแหน่ง  $(0, \dots, 0)$  คำตอบจะย้ายไปอยู่ที่  $(-k, \dots, -k)$

## ง.2 การเลื่อนตำแหน่งโดเมนของฟังก์ชันมาตรฐานเพื่อทดสอบประสิทธิภาพของวิธีMPSOSM

โดยตาราง ง.2 ได้ใช้พารามิเตอร์ตามตาราง 5.1 และรอบของการค้นหาคำตอบตามตาราง 5.2 ได้ใช้จำนวนรอบการค้นหา 50 รอบและได้เพิ่มค่าคงที่เพื่อเลื่อนตำแหน่งโดเมนของฟังก์ชันโดยแทน  $x$  ด้วย  $(x+k)$  โดยเปลี่ยนค่า  $k$  ไปตั้งแต่ 0.00001, 0.0001, 1, 10, และ 100 เข้าไปยังตำแหน่งอนุภาคในแต่ละมิติเพื่อให้คำตอบของฟังก์ชันที่กำหนดไว้เปลี่ยนไป ตามตาราง ง.1 แสดงค่าเฉลี่ยของการเดาคำตอบ (MEC) และ ค่าเฉลี่ยของคำตอบที่ดีที่สุด (MBF)

ตารางที่ ง.2 ผลการค้นหาคำตอบการเลื่อนตำแหน่งของอนุภาค

ค่าคงที่	MPSOSM(k=0)		k=0.00001		k=0.0001	
	MEC	MBF	MEC	MBF	MEC	MBF
ACKLEY	5,272,422	0	9,999,810	1.99E-13	10,000,061	9.95E-14
GRIEWANK	686,922	0	945,731	0	826,760	0
RASTRIGIN	4,222,824	0	9,999,955	2.27E-13	6,743,278	0

ค่าคงที่ ชื่อฟังก์ชัน	MPSOSM(k=0)		k=0.00001		k=0.0001	
	MEC	MBF	MEC	MBF	MEC	MBF
ROSENBROCK	9,999,966	3.94E-29	9,999,913	3.23E-27	10,000,050	1.28E-13
SCHWEFEL	9,999,281	3789.74	9,984,321	4480.63	1,213,788	0
COSINE MIXTURE	1,295,051	0	3,559,055	0	9,999,832	2.27E-13
EXPONENTIAL	988,408	0	10,000,046	5.55E-16	9,999,985	2.06E-26
LEVY	8,917,597	0	10,000,039	1.20E-28	10,000,046	4954.38
MICHALEWICZ	172,429	0	227,017	0	10,000,047	3.55E-15
DIXON-PRICE	11,587	0	1,495	0	10,000,078	5.55E-16
STEP	175,541	0	138,144	0	10,000,047	3.10E-30
SCHAFFER	66,789	0	23,874	0	309,558	0
HOLDER	6,634	0	5,185	0	640	0
BEALE	32,416	0	33,025	0	159,946	0
SHUBERT	4,689	0	5,980	0	23,853	0
GOLDSTEIN- PRICE	12,008	0	11,786	0	6,340	0
SIX-HUMP CAMEL	2,497	0	2,519	0	33,875	0
SPHERE	10,000,002	0	10,000,002	6.06E-40	6,342	0
PARALLEL	10,000,076	2.9643e- 323	10,000,015	6.95E-38	12,863	0
ROTATED	10,000,002	1.4822e- 323	10,000,070	9.56E-249	1,757	0
CIGAR	10,000,077	3.4584e- 323	10,000,078	5.25E-35	10,000,020	1.42E-37
BROWN	10,000,073	6.9169e- 323	9,984,733	4.90E-09	10,000,038	1.19E-36
MULTIMOD	10,000,061	6.93E-272	9,999,935	8.17E-18	10,000,067	6.20E-265
ZAKHAROV	9,999,811	1.39E-99	9,999,918	1.15E-16	9,999,953	6.54E-33
TRID	163,072	0	9,997,661	0.0021	9,999,962	4.90E-07
EASOM	17,965	0	19,100	0	10,000,036	1.79E-17
EGGHOLDER	10,000,079	0	9,995,733	0.01082	9,999,896	1.39E-16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ง.2 ผลการค้นหาคำตอบการเลื่อนตำแหน่งของอนุภาค (ต่อ)

ค่าคงที่	k=1		k=10		k=100	
	MEC	MBF	MEC	MBF	MEC	MBF
ACKLEY	10,000,024	1.32313	9,994,016	1.03E-13	9,999,990	20
GRIEWANK	10,000,007	1.11E-16	5,880,046	5.55E-16	10,000,003	2.44E-15
RASTRIGIN	9,999,967	37.8084	10,000,080	3529.17	9,999,792	915396
ROSENBROCK	1,936,097	0	9,986,927	3.59E+07	9,999,657	1.00E+10
SCHWEFEL	9,999,992	5092.56	9,999,999	4974.12	10,000,043	4737.24
COSINE MIXTURE	9,999,422	13.8351	9,999,628	8355.22	9,998,898	982864
EXPONENTIAL	9,997,293	1	9,997,293	1	9,997,293	1
LEVY	2,125,709	0	9,999,992	1.36297	10,000,028	59451.4
MICHALEWICZ	97,389	0	225,170	0	2,595,426	0
DIXON-PRICE	9,998,407	177.5	9,993,305	1.80E+06	9,993,626	1.00E+10
STEP	218,849	0	207,646	0	9,999,123	15022
SCHAFFER	63,167	0	17,602	0	10,000,044	0.000547
HOLDER	89	0	28	0	33	0
BEALE	34,256	0	9,977,821	859822	9,992,616	1.00E+10
SHUBERT	5,469	0	6,372	0	7,280	0
GOLDSTEIN-PRICE	9,998,195	1.05E-07	9,999,512	1.24E+07	9,994,347	1.00E+10
SIX-HUMP CAMEL	1,215	0	9,999,707	95243.3	9,998,004	1.00E+10
SPHERE	9,999,981	8.11E-30	9,996,723	3125	10,000,007	913076
PARALLEL	10,000,054	1.40E-28	9,999,785	157529	9,991,335	4.61E+07
ROTATED	10,000,006	3.02E-253	10,000,064	1.21E-258	9,999,958	1.55E-256
CIGAR	10,000,030	4.60E-25	9,998,322	1.56E+07	9,988,545	1.00E+10
BROWN	10,000,030	34.765	9,999,941	9874.15	9,999,978	989801
MULTIMOD	10,000,020	7.84E-14	9,998,665	1.00E+10	9,999,626	1.00E+10
ZAKHAROV	9,999,963	6.33E-05	9,998,344	1.00E+10	9,996,891	1.00E+10
TRID	9,993,355	159.545	604	0	9,998,545	3170.32
EASOM	21,047	0	16,083	0	159,286	0
EGGHOLDER	81,179	0	1,873	0	440	0
ZAKHAROV	10,000,045	6.29E-05	9,998,344	264369	9,990,480	4.28E+09
TRID	10,000,024	1.32313	604	1.03E-13	9,999,990	20
EASOM	10,000,007	1.11E-16	16,083	5.55E-16	10,000,003	2.44E-15
EGGHOLDER	9,999,967	37.8084	1,873	3529.17	9,999,792	915396

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตาราง ง.2 จะเห็นว่าเมื่อมีการเลื่อนตำแหน่งโดเมนของฟังก์ชันออกไป  $k$  หน่วย ทำให้คำตอบของ  $f(x^*)$  ไม่อยู่ที่ตำแหน่ง 0 จะไม่เจอคำตอบ โดยที่ถ้าเลื่อนยิ่งเลื่อนตำแหน่งโดเมนของฟังก์ชันออกไปกว้างขึ้น คำตอบที่ทำให้  $f(x^*) = 0$  ที่ได้ก็จะน้อยลงไปด้วย ยกเว้นในบางฟังก์ชัน เช่น ฟังก์ชัน ROSENBROCK ที่มีคำตอบในแต่ละมิติอยู่ที่  $(1, \dots, 1)$  ทำให้เมื่อเลื่อนตำแหน่งโดเมนของฟังก์ชันออกไป 1 คำตอบของ  $f(x^*)$  ที่ได้นั้นก็จะเข้าใกล้หรือมีโอกาสที่  $f(x^*) = 0$  มากขึ้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้เขียน

ชื่อ - นามสกุล                      นางสาวกนกพร เสงข์นท  
 วัน เดือน ปีเกิด                    2 ตุลาคม 2531  
 การศึกษา                            ปีการศึกษา 2544 - 2547 ระดับมัธยมศึกษาตอนต้น  
     โรงเรียนสตรีราชินูทิศ อุตรธานี  
     ปีการศึกษา 2547 - 2550 ระดับมัธยมศึกษาตอนปลาย  
     การศึกษานอกโรงเรียน อุตรธานี  
     ปีการศึกษา 2550 - 2554 ระดับอุดมศึกษา  
     คณะวิทยาศาสตร์ สาขาวิชาวิทยาการคอมพิวเตอร์  
     มหาวิทยาลัยขอนแก่น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้