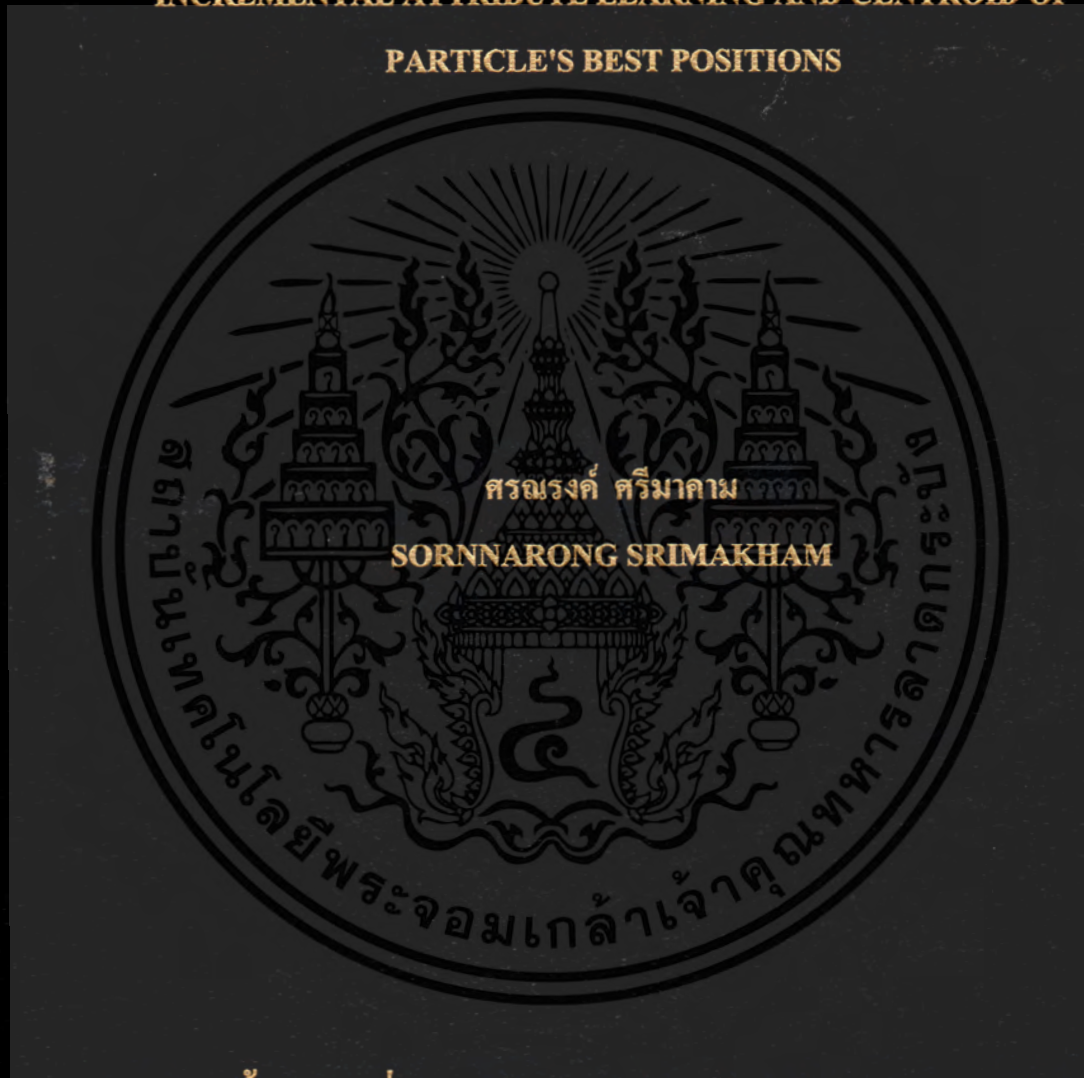


การปรับปรุงการหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคโดยใช้การเรียนรู้แบบ
เพิ่มและจุดเซนทรอยด์ของตำแหน่งที่ดีที่สุด

IMPROVING PARTICLE SWARM OPTIMIZATION BY USING
INCREMENTAL ATTRIBUTE LEARNING AND CENTROID OF
PARTICLE'S BEST POSITIONS



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2561

KMITL-2018-EN-M-070-017

การปรับปรุงการหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคโดยใช้การเรียนรู้แบบ
เพิ่มและจุดเซนทรอยด์ของตำแหน่งที่ดีที่สุด

IMPROVING PARTICLE SWARM OPTIMIZATION BY USING
INCREMENTAL ATTRIBUTE LEARNING AND CENTROID OF
PARTICLE'S BEST POSITIONS



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2561

KMITL-2018-EN-M-070-017

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**IMPROVING PARTICLE SWARM OPTIMIZATION BY USING
INCREMENTAL ATTRIBUTE LEARNING AND CENTROID OF
PARTICLE'S BEST POSITIONS**



**A THESIS SUBMITTED IN PARTIAL 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-070-017**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2018

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การปรับปรุงการหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาคโดยใช้การเรียนรู้แบบเพิ่มและจุด
เซนทรอยด์ของตำแหน่งที่ดีที่สุด

Thesis Title Improving Particle Swarm Optimization by Using Incremental Attribute
Learning and Centroid of Particle's Best Positions

นักศึกษา นายศรณรงค์ ศรีมาคาม

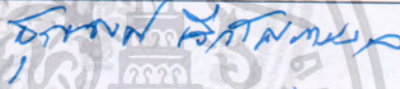

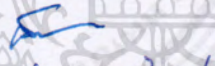
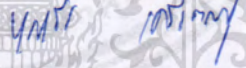

รหัสประจำตัว 59601090

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา วิศวกรรมคอมพิวเตอร์

อาจารย์ที่ปรึกษาวิทยานิพนธ์ รศ.ดร.เกียรติกุล เจียรนัยธนะกิจ

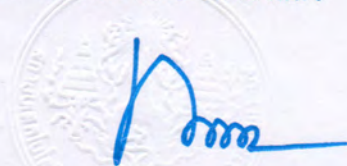
หมายเลขวิทยานิพนธ์ KMITL-2018-EN-M070-017

คณะกรรมการสอบวิทยานิพนธ์	ลายมือชื่อ
รศ.ดร.สุรพงศ์ เอื้อวัฒนามงคล	
ผศ.ดร.ชุตินิเมษฐ์ ศรีนิลทา	
ผศ.ดร.สมศักดิ์ วลัยรัชต์	
รศ.ดร.บุญธีร์ เครือตราชู	
รศ.ดร.เกียรติกุล เจียรนัยธนะกิจ	

วัน / เดือน / ปี ที่สอบ วันศุกร์ที่ 23 กุมภาพันธ์ พ.ศ. 2561 เวลา 08.30-10.30 น.
สถานที่สอบ ณ ห้องประชุม 3 ชั้น 5 อาคาร A

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

คณะวิศวกรรมศาสตร์ รับรองแล้ว



(รองศาสตราจารย์ ดร. คมสัน มาลีสี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า
โดยไม่ได้รับอนุญาตจากคณะวิศวกรรมศาสตร์
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต่อจากนี้จึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
วันที่ 23 กุมภาพันธ์ พ.ศ. 2561

หัวข้อวิทยานิพนธ์	การปรับปรุงการหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคโดยใช้การเรียนรู้แบบเพิ่มและจุดเซนทรอยด์ของตำแหน่งที่ดีที่สุด
นักศึกษา	นายศรณรงค์ ศรีมาคาม
รหัสประจำตัว	59601090
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
พ.ศ.	2561
อาจารย์ที่ปรึกษาวิทยานิพนธ์	รศ. ดร.เกียรติกุล เกียรติยศนักษะกิจ

บทคัดย่อ

การหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization, PSO) คือ อัลกอริทึมที่สามารถค้นหาตำแหน่งที่ดีที่สุดของฟังก์ชันซึ่งประกอบไปด้วยจุดที่ดีที่สุดเฉพาะที่ (Local Optima) อย่างไรก็ตาม PSO ยังคงประสบปัญหา คือ PSO จะทำงานได้ยากขึ้นเมื่อฟังก์ชันมีความซับซ้อนและมีจำนวนมิติเพิ่มมากขึ้น งานวิจัยนี้จึงนำ PSO มาพัฒนาโดยการประยุกต์ใช้กลยุทธ์การเรียนรู้แบบเพิ่ม (Incremental Attribute Learning Strategy) สำหรับครึ่งหนึ่งของกลุ่มอนุภาค และการหาจุดเซนทรอยด์ (Centroid) จากจุดที่ดีที่สุดของกลุ่มอนุภาคทุกตัว (GBest) เพื่อหลีกเลี่ยงจุดที่ดีที่สุดเฉพาะที่ (Local Optima) ซึ่งมักพบได้ในฟังก์ชันที่มีจุดที่ดีที่สุดแท้จริง (Global Optima) มากกว่า 1 จุด (Multimodal Function) จึงได้ทำการทดลองกับฟังก์ชันมาตรฐานทั้งหมด 14 ฟังก์ชันและ 3 ฟังก์ชันเพิ่มเติม เมื่อเปรียบเทียบผลลัพธ์จากวิธีการหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคปกติ (Conventional PSO) และงานวิจัยการหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาคด้วยการเรียนรู้แบบเพิ่ม (Incremental Attribute Based Particle Swarm Optimization, IAPSO) ผลการทดลองแสดงให้เห็นว่าอัลกอริทึมสามารถพัฒนาในส่วนของความแม่นยำ และความเสถียรในฟังก์ชันส่วนใหญ่ เมื่อเปรียบเทียบกับวิธีการหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาคแบบปกติ (Conventional PSO) และ งานวิจัยการหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาคด้วยการเรียนรู้แบบเพิ่ม (Incremental Attribute Based Particle Swarm Optimization, IAPSO)

Thesis	Improving Particle Swarm Optimization by Using Incremental Attribute Learning and Centroid of Particle's Best Positions
Student	Mr.Sornnarong Srimakham
Student ID.	59601090
Degree	Master of Engineering
Program	Computer Engineering
Year	2018
Thesis Advisor	Assoc.Prof.Dr.Kietikul Jearanaitanakij

ABSTRACT

Particle Swarm Optimization (PSO) is a powerful algorithm that can search a solution for a function which contains a large number of peaks and valleys. However, PSO might encounter a difficulty when the function gets more complex or the number of attributes (dimensions) grows larger. This thesis proposes a modification of PSO by using the incremental attribute learning strategy for half of swarm along with the centroid of particle's best positions to avoid the local minima which can easily occur in a multimodal problem. The experimental results from 14 standard benchmark functions and 3 extra functions show that the proposed method can improve PSO in terms of optimality and stability when compared with the conventional PSO and another incremental attribute-based PSO from a most function.

กิตติกรรมประกาศ

วิทยานิพนธ์เล่มนี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา รศ. ดร.เกียรติกุล เกียรตินัยชนะกิจ ที่ให้ความช่วยเหลือ ให้คำชี้แนะแนวทางการแก้ปัญหาตลอดจนให้ความรู้และประการที่ดีแก่ข้าพเจ้า

ขอขอบพระคุณครูอาจารย์ทุกท่านที่ให้ความรู้ ประสบการณ์ คำแนะนำ ตลอดจนจริยธรรมคุณธรรมที่เป็นประโยชน์แก่ผู้วิจัย

ขอขอบคุณสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่อำนวยความสะดวกสถานที่ศึกษาและการทำวิจัยในครั้งนี้

สำหรับคุณงามความดีอันใดที่เกิดจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอบอบให้กับบิดามารดา ซึ่งเป็นที่รักและเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้และถ่ายทอดประสบการณ์ที่ดีให้แก่ข้าพเจ้า

ศรณรงค์ ศรีมาคาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และแจ้งอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญของปัญหา.....	1
1.2 จุดมุ่งหมายและวัตถุประสงค์.....	1
1.3 สมมุติฐานของการศึกษา.....	1
1.4 ขอบเขตของการวิจัย.....	2
1.5 ขั้นตอนของการศึกษา.....	2
1.6 เครื่องมือและอุปกรณ์ที่ใช้ในการวิจัย.....	2
1.7 โครงสร้างของวิทยานิพนธ์.....	3
บทที่ 2 ความรู้พื้นฐานที่เกี่ยวข้องกับงานวิจัย.....	4
2.1 การหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization, PSO)...	4
2.2 กลยุทธ์การเรียนรู้แบบเพิ่ม (Incremental Attribute Learning Strategy).....	7
2.3 กลยุทธ์การเรียนรู้แบบเพิ่มขึ้นทีละรอบ (Iterative Incremental Learning Strategy).....	8
บทที่ 3 งานวิจัยที่เกี่ยวข้อง.....	10
3.1 อัลกอริทึม Incremental attribute based particle swarm optimization (IAPSO) (Bai, Cheng, Tadjouddine, Guan 2012).....	10
3.2 อัลกอริทึม One-dimensional Searching-based Particle Swarm Optimization (OPSO) (Lin, Zhao 2012).....	11
บทที่ 4 งานวิจัยที่นำเสนอ.....	17
4.1 การกำหนดจุดเริ่มต้น (Position Initialization).....	17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และ IV อ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
4.2 การขีดหุ่นจุดเริ่มต้นของอนุภาค (Processed Attribute Boundary).....	17
4.3 การแบ่งกลุ่มอนุภาค (Swarm Classification).....	18
4.4 การคำนวณจุดเซนทรอยด์ (Centroid Calculation).....	19
4.5 การวนรอบการทำงาน (Generation Repetition).....	20
บทที่ 5 การทดลองและการเปรียบเทียบผลการทดลอง.....	22
5.1 ค่าพารามิเตอร์.....	22
5.2 การทดลอง.....	23
5.3 ฟังก์ชันที่ใช้ทดลอง.....	23
5.4 ผลการทดลอง.....	39
5.5 การค้นหาจุดบกพร่องของอัลกอริทึม.....	49
5.6 การทดสอบประสิทธิภาพของอัลกอริทึม.....	54
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ.....	62
6.1 สรุปผลการวิจัย.....	62
6.2 ข้อเสนอแนะ.....	62
เอกสารอ้างอิง.....	64
ภาคผนวก.....	65
ประวัติผู้เขียน.....	70

สารบัญตาราง

ตารางที่	หน้า
4.1 ผลการทดลองการเปรียบเทียบการใช้จุดเซนทรอยด์.....	19
5.1 ฟังก์ชัน Noncontinuous Rastrigin.....	24
5.2 ฟังก์ชัน Ackley.....	25
5.3 ฟังก์ชัน Griewank.....	28
5.4 ฟังก์ชัน Generalized Penalized.....	29
5.5 ฟังก์ชัน Rosenbrock.....	30
5.6 ฟังก์ชัน Dixon-Price.....	31
5.7 ฟังก์ชัน Generalized Modified Rosenbrock's Function No.01.....	32
5.8 ฟังก์ชัน Michalewicz.....	33
5.9 ฟังก์ชัน DE JONG N. 5.....	34
5.10 ฟังก์ชัน EASOM.....	35
5.11 ฟังก์ชัน Trid.....	36
5.12 ฟังก์ชัน SUM OF DIFFERENT POWERS.....	37
5.13 ฟังก์ชัน SUM SQUARES.....	38
5.14 ฟังก์ชัน ZAKHAROV.....	39
5.15 ผลการทดลองของฟังก์ชันประเภท Many Local Minima ใน 50 มิติ.....	40
5.16 ผลการทดลองของฟังก์ชันประเภท Many Local Minima ใน 100 มิติ.....	41
5.17 ผลการทดลองของฟังก์ชันประเภท Valley-Shaped ใน 50 มิติ.....	41
5.18 ผลการทดลองของฟังก์ชันประเภท Valley-Shaped ใน 100 มิติ.....	42
5.19 ผลการทดลองของฟังก์ชันประเภท Steep Ridges/Drops ใน 50 มิติ และ 2 มิติ.....	43
5.20 ผลการทดลองของฟังก์ชันประเภท Steep Ridges/Drops ใน 100 มิติ.....	44
5.21 ผลการทดลองของฟังก์ชันประเภท Bowl-Shaped ใน 50 มิติ.....	44
5.22 ผลการทดลองของฟังก์ชันประเภท Bowl-Shaped ใน 100 มิติ.....	45
5.23 ผลการทดลองของฟังก์ชันประเภท Plate-Shaped.....	45
5.24 สรุปผลการทดลองที่ 2 และ 50 มิติ.....	46
5.25 สรุปผลการทดลองที่ 100 มิติ.....	47
5.26 ฟังก์ชันที่ใช้ในการทดลองเพิ่มเติม.....	50
5.27 ผลการทดลองเพิ่มเติม.....	50

สารบัญตาราง (ต่อ)

ตารางที่	หน้า
5.28 การทดสอบประสิทธิภาพการคำนวณจุดเซนทรอยด์.....	55
5.29 การทดสอบประสิทธิภาพการวนรอบการทำงานที่ 50 มิติ.....	57
5.30 การทดสอบประสิทธิภาพการวนรอบการทำงานที่ 100 มิติ.....	59



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และ VII อ่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

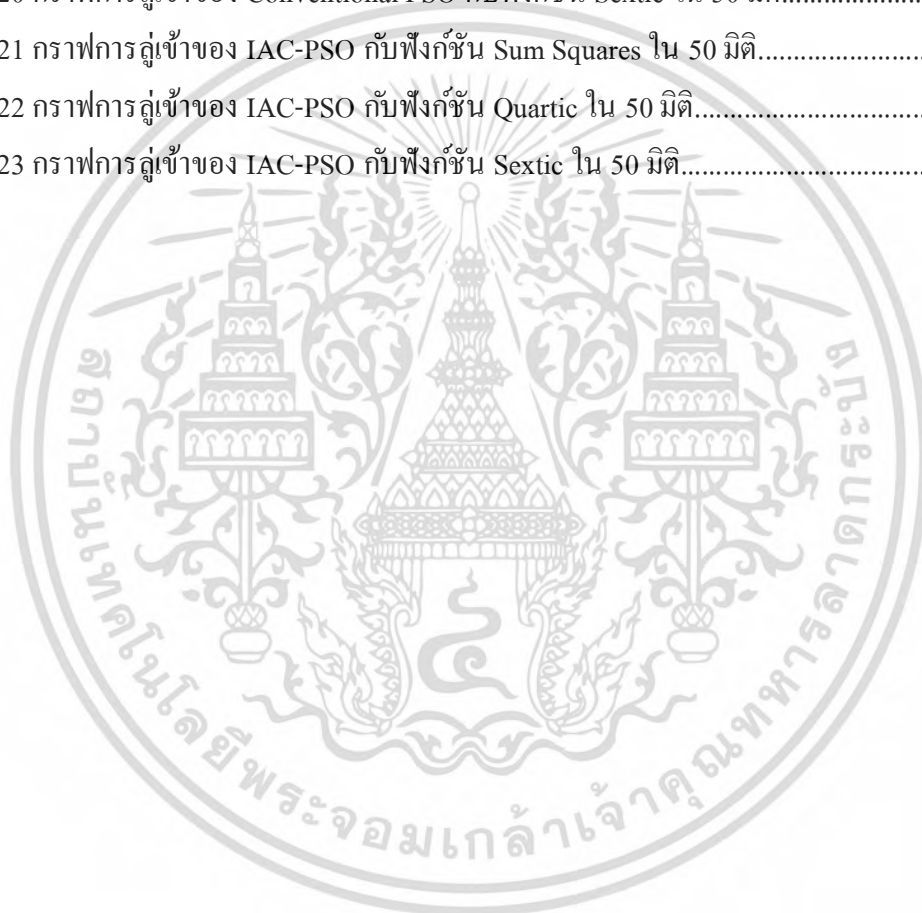
สารบัญรูป

รูปที่	หน้า
2.1 Pseudo Code ของ PSO.....	5
2.2 การกำหนดตำแหน่งของอนุภาค.....	5
2.3 การกำหนดค่าฝูงของอนุภาค.....	6
2.4 การค้นหาบริเวณใกล้เคียงจ่าฝูง.....	6
2.5 ผังการทำงานของกลยุทธ์แบบเพิ่ม.....	7
2.6 ผังการทำงานของกลยุทธ์การเรียนรู้แบบเพิ่มขึ้นที่ละรอบ.....	8
3.1 ผังการทำงานของ IAPSO.....	10
3.2 ตัวอย่าง PCA ในพื้นที่ 2 มิติ.....	14
3.3 ผังการทำงานของ OPSO.....	15
4.1 การค้นหาทีละมิติของ IAC-PSO.....	17
4.2 การแบ่งกลุ่มอนุภาคของ IAC-PSO.....	18
4.3 การหาจุดเซนทรอยด์ในฟังก์ชัน Ackley.....	19
4.4 ผังการทำงานของอัลกอริทึม.....	21
5.1 ฟังก์ชัน Noncontinuous Rastrigin.....	23
5.2 ฟังก์ชัน Rastrigin.....	24
5.3 ฟังก์ชัน Ackley.....	25
5.4 ฟังก์ชัน Griewank.....	26
5.5 ฟังก์ชัน Generalized Penalized แบบ 3 มิติ.....	28
5.6 ฟังก์ชัน Generalized Penalized แบบ 2 มิติ.....	29
5.7 ฟังก์ชัน Rosenbrock.....	30
5.8 ฟังก์ชัน Dixon-Price.....	31
5.9 ฟังก์ชัน Generalized Modified Rosenbrock's Function No.01.....	32
5.10 ฟังก์ชัน Michalewicz.....	33
5.11 ฟังก์ชัน DE JONG N. 5.....	34
5.12 ฟังก์ชัน EASOM.....	35
5.13 ฟังก์ชัน Trid.....	36
5.14 ฟังก์ชัน SUM OF DIFFERENT POWERS.....	37
5.15 ฟังก์ชัน SUM SQUARES.....	38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และ VIII อ่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.16 ฟังก์ชัน ZAKHAROV.....	39
5.17 กราฟการลู่เข้าของ IAC-PSO กับฟังก์ชัน Noncontinuous Rastrigin ใน 100 มิติ.....	49
5.18 กราฟการลู่เข้าของ Conventional PSO กับฟังก์ชัน Sum Squares ใน 50 มิติ.....	51
5.19 กราฟการลู่เข้าของ Conventional PSO กับฟังก์ชัน Quartic ใน 50 มิติ.....	51
5.20 กราฟการลู่เข้าของ Conventional PSO กับฟังก์ชัน Sextic ใน 50 มิติ.....	52
5.21 กราฟการลู่เข้าของ IAC-PSO กับฟังก์ชัน Sum Squares ใน 50 มิติ.....	52
5.22 กราฟการลู่เข้าของ IAC-PSO กับฟังก์ชัน Quartic ใน 50 มิติ.....	53
5.23 กราฟการลู่เข้าของ IAC-PSO กับฟังก์ชัน Sextic ใน 50 มิติ.....	53



บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

การค้นหาค่าที่เหมาะสมสำหรับงานแต่ละชนิดถือเป็นส่วนสำคัญในการดำเนินงาน การหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization, PSO) คืออัลกอริทึมที่สามารถค้นหาค่าเหมาะที่สุดของสมการ หรือฟังก์ชันซึ่งประกอบด้วยจุดดีที่สุดเฉพาะที่ (Local Optima) PSO ได้ถูกนำมาศึกษาวิจัยอย่างกว้างขวางทั้งในการประยุกต์ใช้งานต่างๆ และการปรับเปลี่ยนรูปแบบการค้นหา อย่างไรก็ตาม PSO จะทำงานได้ยากขึ้นเมื่อฟังก์ชันมีความซับซ้อนและมีจำนวนมิติเพิ่มมากขึ้น

งานวิจัยนี้จึงนำ PSO มาพัฒนาโดยการประยุกต์ใช้กลยุทธ์การเรียนรู้แบบเพิ่ม (Incremental Attribute Learning Strategy) และการหาจุดเซนทรอยด์ (Centroid) จากจุดที่ดีที่สุดของอนุภาคทุกตัว (GBest) เพื่อหลีกเลี่ยงจุดดีที่สุดเฉพาะที่ (Local Optima) ซึ่งมักพบได้ในฟังก์ชันที่มีจุดดีที่สุดแท้จริง (Global Optima) มากกว่า 1 จุด (Multimodal Function) ซึ่งจะช่วยให้เพิ่มความแม่นยำ และเสถียรภาพให้กับอัลกอริทึม

1.2 จุดมุ่งหมายและวัตถุประสงค์

จุดมุ่งหมายและวัตถุประสงค์ของวิทยานิพนธ์นี้เพื่อนำเสนออัลกอริทึมที่สามารถค้นหาค่าที่เหมาะสมในฟังก์ชันที่ประกอบไปด้วยจุดดีที่สุดเฉพาะที่ (Local Optima) และมีจำนวนมากได้อย่างแม่นยำ และเสถียร เมื่อเปรียบเทียบผลการทดลองกับการหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาคแบบปกติ (Conventional PSO) และงานวิจัยการค้นหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาคด้วยการเรียนรู้แบบเพิ่ม (Incremental Attribute Based Particle Swarm, IAPSO) โดยใช้ฟังก์ชันมาตรฐานทั้งหมด 14 ฟังก์ชัน ในการเปรียบเทียบ

1.3 สมมุติฐานของการศึกษา

การประยุกต์ใช้กลยุทธ์การเรียนรู้แบบเพิ่ม (Incremental Attribute Learning Strategy) และการหาจุดเซนทรอยด์ (Centroid) จากจุดที่ดีที่สุดของอนุภาคทุกตัว (GBest) จะสามารถหลีกเลี่ยงจุดดีที่สุดเฉพาะที่ (Local Optima) ได้ดีกว่า มีการลู่เข้า (Convergence) ที่เร็วกว่า และเมื่อเพิ่มจำนวนมิติ อัลกอริทึมจะมีความเสถียรมากกว่า เมื่อเปรียบเทียบกับ Conventional PSO, IAPSO

1.4 ขอบเขตของการวิจัย

ขอบเขตของการวิจัยนี้สามารถอธิบายได้ดังต่อไปนี้

1.4.1 อัลกอริทึมจะใช้จำนวนครั้งที่ดำเนินการ (Function Evaluation Times) ได้แก่ 60,000 สำหรับ 50 มิติ และ 120,000 สำหรับ 100 มิติ ในการค้นหา

1.4.2 อัลกอริทึมจะถูกกำหนดมีจำนวนอนุภาค (Particles) และค่าพารามิเตอร์ต่างๆ เท่ากับงานวิจัยการค้นหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาคด้วยการเรียนรู้แบบเพิ่ม (Incremental Attribute Based Particle Swarm, IAPSO) เพื่อการเปรียบเทียบ

1.4.3 อัลกอริทึมสามารถค้นหาคำตอบได้ดีและเสถียรกว่า Conventional PSO, IAPSO ในทุกๆ ฟังก์ชันยกเว้นฟังก์ชันบางส่วน ซึ่งเป็นข้อจำกัดของอัลกอริทึม

1.4.4 การทดลองจะทดลองค้นหาทั้งหมด 50 ครั้งและนำเสนอด้วยค่าทางสถิติทั้ง 4 ค่า ได้แก่ Best, Median, Worst และ Mean

1.5 ขั้นตอนของการศึกษา

ขั้นตอนการศึกษางานวิจัยนี้สามารถอธิบายได้ดังต่อไปนี้

1.5.1 ศึกษาความรู้พื้นฐานการหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization, PSO)

1.5.2 ศึกษางานวิจัยที่เกี่ยวข้องที่มีผู้เสนอมาแล้ว

1.5.3 ออกแบบอัลกอริทึมเพื่อแก้ปัญหา

1.5.4 สร้างโปรแกรมเพื่อทดสอบอัลกอริทึมที่ได้ออกแบบไว้

1.5.5 ทดสอบอัลกอริทึมโดยเปรียบเทียบกับ Conventional PSO, IAPSO

1.5.6 วิเคราะห์และแก้ไขปัญหาที่เกิดขึ้น

1.5.7 ทดสอบอัลกอริทึมอีกครั้ง

1.5.8 สรุปผลการทดลอง

1.5.9 ส่งบทความวิจัยเพื่อตีพิมพ์ในวารสารการประชุมระดับนานาชาติและจัดทำรูปเล่มวิทยานิพนธ์

1.6 เครื่องมือและอุปกรณ์ที่ใช้ในการวิจัย

1.6.1 เครื่องคอมพิวเตอร์ส่วนบุคคลประกอบด้วยหน่วยประมวลผลกลาง Intel Core i5 และหน่วยความจำ (RAM) 20 กิกะไบต์ (GB) จำนวน 1 เครื่อง

1.6.2 ระบบปฏิบัติการวินโดวส์ 10

1.6.3 โปรแกรมเมตแล็บเวอร์ชัน 2010 (Matlab 2010)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.7 โครงสร้างของวิทยานิพนธ์

วิทยานิพนธ์ฉบับนี้แบ่งออกเป็น 6 บท โดยแต่ละบทจะมีรายละเอียดดังต่อไปนี้

- บทที่ 1** จะกล่าวถึงรายละเอียดในเบื้องต้น ได้แก่ ที่มาและความสำคัญของปัญหา จุดมุ่งหมายและวัตถุประสงค์ สมมุติฐานของการศึกษา ขอบเขตของการวิจัย ขั้นตอนของการศึกษา เครื่องมือและอุปกรณ์ที่ใช้การวิจัย และโครงสร้างวิทยานิพนธ์
- บทที่ 2** จะอธิบายความรู้พื้นฐานที่เกี่ยวข้องกับการวิจัย ได้แก่ การหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization, PSO) กลยุทธ์การเรียนรู้แบบเพิ่ม (Incremental Attribute Learning Strategy) และกลยุทธ์การเรียนรู้แบบเพิ่มขึ้นทีละรอบ (Iterative Incremental Learning Strategy)
- บทที่ 3** กล่าวถึงงานวิจัยที่เกี่ยวข้อง
- บทที่ 4** จะอธิบายรายละเอียดของอัลกอริทึม ได้แก่ องค์ประกอบของอัลกอริทึม และขั้นตอนการทำงานของอัลกอริทึม
- บทที่ 5** จะนำเสนอผลการทดลองและการวิเคราะห์ในฟังก์ชันต่างๆทั้งหมด 14 ฟังก์ชัน โดยจะเปรียบเทียบกับ Conventional PSO, IAPSO
- บทที่ 6** สรุปผลการทดลอง ข้อเสนอแนะและแนวทางในการปรับปรุง

บทที่ 2

ความรู้พื้นฐานที่เกี่ยวข้องกับงานวิจัย

2.1 การหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization, PSO)

PSO เป็นอัลกอริทึมที่ใช้ในการหาค่าตอบที่ดีที่สุดของสมการหรือฟังก์ชัน ซึ่งประกอบด้วยตัวแปรหลายตัวเช่นเดียวกับอัลกอริทึมวิวัฒนาการ (Evolutionary algorithms) PSO ถูกคิดค้นโดย J.Kennedy และ R.Eberhart [1] ซึ่งได้รับแรงบันดาลใจมาจากการสำรวจพฤติกรรมการอยู่ร่วมกันเป็นกลุ่มของฝูงสัตว์เช่น ฝูงนก และ ฝูงปลา เป็นต้น ตามธรรมชาติฝูงสัตว์จะช่วยกันหาอาหาร โดยที่สัตว์จะรับข้อมูลจากประสบการณ์การค้นหาจากฝูงและสัตว์ตัวอื่นๆ เมื่อสมาชิกตัวหนึ่งภายในฝูงพบอาหารก็จะดึงดูดให้สมาชิกตัวอื่นๆภายในฝูงเข้ามา

ใน PSO ฝูงสัตว์จะถูกเรียกว่า Swarm ภายในฝูงจะประกอบไปด้วยสมาชิกเรียกว่า อนุภาค และปริมาณอาหารจะเปรียบเสมือนกับค่าจากฟังก์ชัน เมื่ออนุภาคตัวหนึ่งภายใน Swarm พบตำแหน่งที่มีอาหารมากกว่าตัวอื่นๆ (ตำแหน่งที่มีค่ามากที่สุดหรือน้อยสุด) ก็จะกลายเป็นฝูงและดึงดูดให้อนุภาคตัวอื่นๆเข้ามา

PSO จะเริ่มต้นด้วยการสุ่มตำแหน่งอนุภาคทุกตัวใน Swarm ภายในพื้นที่ค้นหา (Search Space) จากนั้นอัลกอริทึมจะคำนวณค่าความเหมาะสม (Fitness) จากอนุภาคและหาค่า GBest โดยจะมีค่าเท่ากับค่าที่ดีที่สุดของอนุภาค (PBest) ที่ดีที่สุดในฝูง ใน 1 กระบวนการทำงาน (Iteration) ตำแหน่งและความเร็วของอนุภาคจะถูกปรับปรุงโดยขึ้นอยู่กับตำแหน่ง PBest และ GBest อัลกอริทึมจะทำงานจนกว่าจะถึงเงื่อนไขจบการทำงาน (Stop Criteria) การทำงานของอนุภาคจะสามารถอธิบายได้ดังนี้

กำหนดให้ w คือค่าถ่วงน้ำหนัก, v_{id} คือ ความเร็วของอนุภาคตัวที่ i^{th} , x_{id} คือตำแหน่งของอนุภาคตัวที่ i^{th} , p_{id} คือตำแหน่งที่ดีที่สุดของอนุภาคตัวที่ i^{th} , p_{gd} คือ ตำแหน่งที่ดีที่สุดของฝูง, c_1 และ c_2 คือ แฟลคเตอร์การเรียนรู้, r คือ ค่าสุ่มตัวเลขในช่วง $[0-1]$, d คือมิติของฟังก์ชัน ตำแหน่งและความเร็วของอนุภาคจะถูกปรับปรุงทุกๆกระบวนการทำงาน (Iteration) ด้วยสมการ 2.1 และ 2.2 นอกจากนี้ Pseudo Code ของ PSO จะแสดงในรูปที่ 2.1

$$v_{id} = wv_{id} + c_1r * (p_{id} - x_{id}) + c_2r * (p_{gd} - x_{id}) \quad (2.1)$$

$$x_{id} = x_{id} + v_{id} \quad (2.2)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

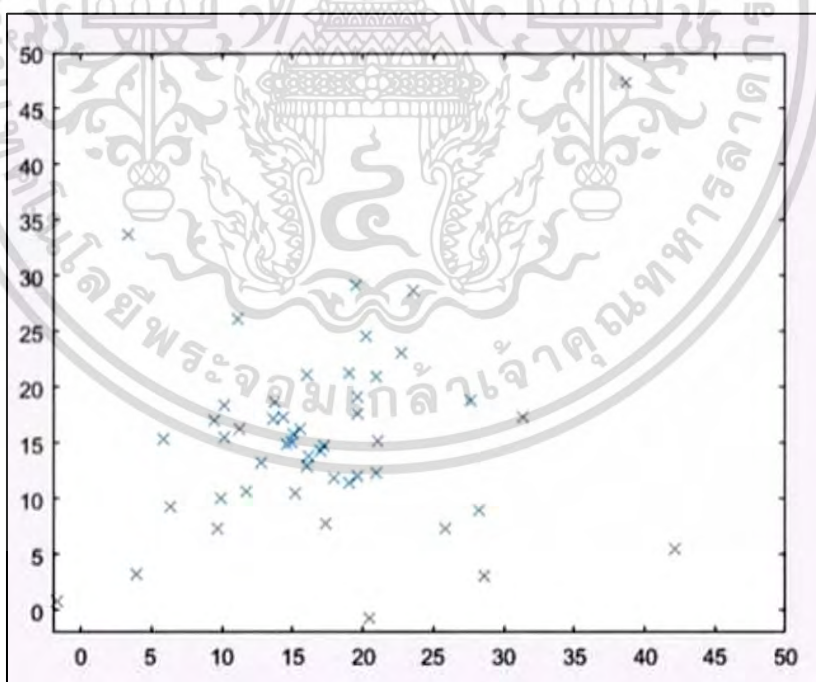
```

1:  $f$ : denotes the function
2:  $P.x_i$ : position of  $i$ th particle
3:  $P.b_i$ : personal best position
4:  $P.\hat{b}$ : previous best position of the whole swarm
5: Initialize a swarm of  $D$ -dimensional particles
6: while not find "good enough" solution or not reach the
   maximum iteration do
7:   for each particle  $i \in [1 \dots s]$ : do
8:     if  $f(P.x_i) < f(P.b_i)$  then
9:        $P.b_i = P.x_i$ 
10:    end if
11:    if  $f(P.b_i) < f(P.\hat{b})$  then
12:       $P.\hat{b} = P.b_i$ 
13:    end if
14:    Update each dimension using equations (1) and (2)
15:  end for
16: end while

```

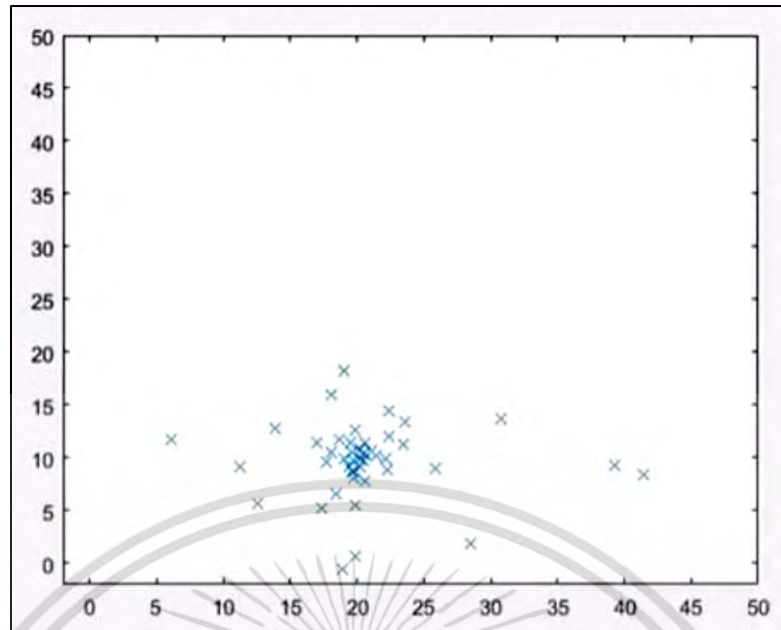
รูปที่ 2.1 Pseudo Code ของ PSO [2]

2.1.1 ตัวอย่างการทำงานของ PSO



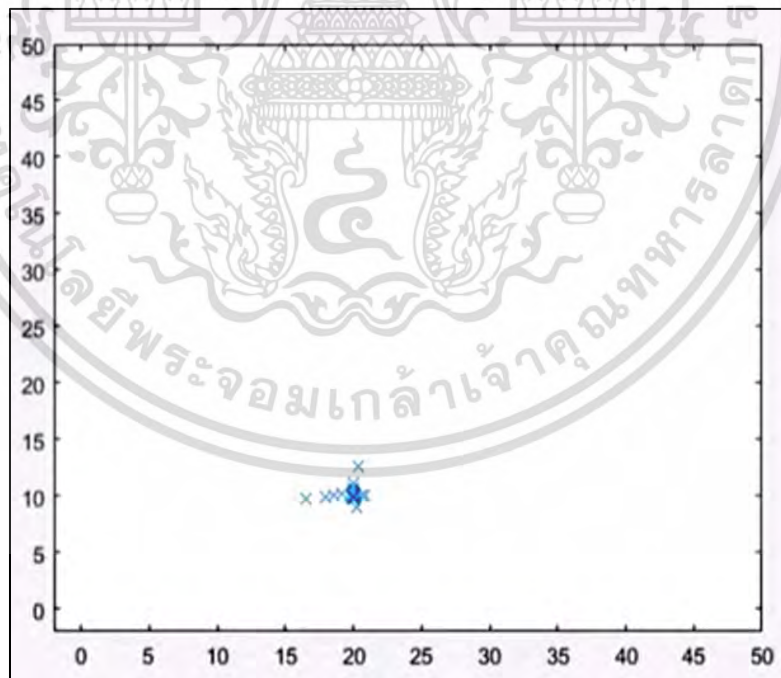
รูปที่ 2.2 การกำหนดตำแหน่งของอนุภาค

จากรูปที่ 2.2 x คืออนุภาคแต่ละตัว ซึ่งกำลังถูกสุ่มตำแหน่งเพื่อค้นหาตำแหน่งที่ดีที่สุดภายใน Search Space 2 มิติ โดยกำหนดขอบเขตการค้นหาตั้งแต่ช่วง $[0,50]$
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 การกำหนดจำฝูงของอนุภาค

ในรูปที่ 2.3 อนุภาคตัวที่อยู่ในตำแหน่งที่ดีที่สุดจะถูกกำหนดให้เป็นจำฝูง จากนั้นอนุภาคตัวอื่นๆจะถูกกำหนดความเร็วให้เข้าหาจำฝูงด้วยสมการที่ 2.1 และ 2.2

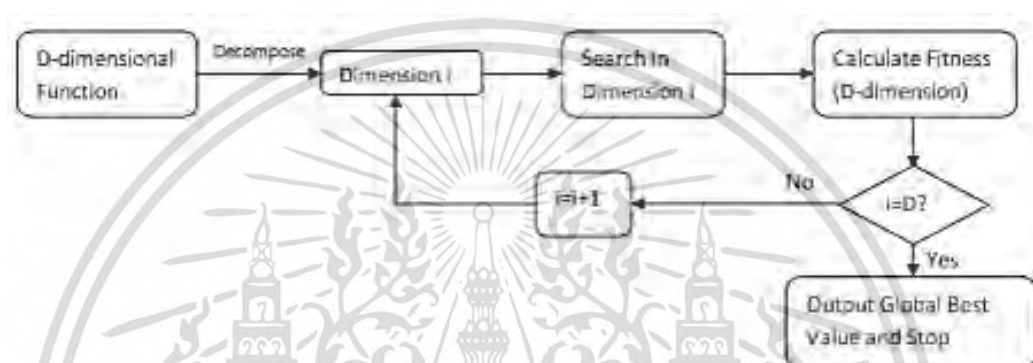


รูปที่ 2.4 การค้นหาบริเวณใกล้เคียงจำฝูง

เมื่อมีอนุภาคตัวอื่นสามารถค้นหาตำแหน่งได้ดีกว่าจำฝูงอนุภาคตัวนั้นจะถูกกำหนดให้เป็นจำฝูงทันที จากรูปที่ 2.4 อนุภาคทุกตัวกำลังค้นหาบริเวณใกล้เคียงจำฝูงในขณะนั้น เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 กลยุทธ์การเรียนรู้แบบเพิ่ม (Incremental Attribute Learning Strategy)

กลยุทธ์การเรียนรู้แบบเพิ่มถูกออกแบบมาเพื่อแก้ปัญหาฟังก์ชันที่มีมิติมาก ซึ่งถือเป็นปัญหาใหญ่ของ PSO เนื่องจากเมื่อเราได้ค่า Fitness ที่ดีจากมิติหนึ่งจะส่งผลให้มิติอื่น ๆ มีค่า Fitness ที่ไม่ดีตามมา ค่า Fitness จะถูกคิดหลังจากการปรับตำแหน่งของอนุภาคในทุก ๆ มิติ Guan [2-5] ได้ประยุกต์กลยุทธ์การเรียนรู้แบบเพิ่มเพื่อพัฒนาประสิทธิภาพของฟังก์ชัน ซึ่งในการใช้กลยุทธ์การเรียนรู้แบบเพิ่มจะสนใจเพียงองค์ประกอบเดียวเท่านั้นหลังจากนั้นจึงจะเริ่มสนใจองค์ประกอบถัดมา



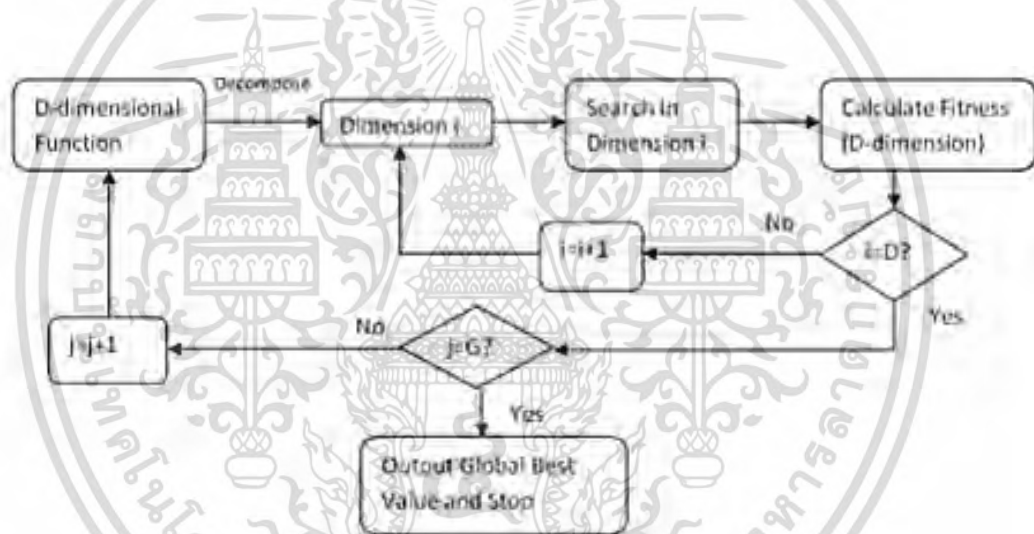
รูปที่ 2.5 ผังการทำงานของกลยุทธ์แบบเพิ่ม [2]

รูปที่ 2.5 คือผังการทำงานของกลยุทธ์การเรียนรู้แบบเพิ่ม ในผังการทำงานนี้มิติของฟังก์ชันจะถูกมองว่าเป็น 1 องค์ประกอบ ดังนั้นเมื่อฟังก์ชันที่ประกอบด้วย D (D เป็นเลขจำนวนเต็มบวก) มิติฟังก์ชันนั้นก็จะประกอบไปด้วยองค์ประกอบทั้งหมด D องค์ประกอบ

กลยุทธ์นี้ใช้การแยกองค์ประกอบในการหาค่าที่ดีที่สุด ซึ่งการวิจัยนี้ได้ใช้กลยุทธ์การแยกองค์ประกอบในการลดความยากของปัญหา กลยุทธ์การเรียนรู้แบบเพิ่มจะเริ่มทำงานจากตรงมุมซ้ายบนของผังการทำงาน ภายในผังการทำงานจะมีรูปการทำงานหลักอยู่ 1 รูปเพื่อใช้ในการค้นหาทีละ 1 มิติ ในขั้นตอนแรกทุกมิติของฟังก์ชันจะถูกแบ่งออกเป็น D องค์ประกอบตามจำนวนมิติของฟังก์ชัน จากนั้นจะเริ่มค้นหามิติที่ i^h ซึ่งในตอนแรก i จะถูกบังคับให้เริ่มต้นด้วย 1 ถัดมาที่ขั้นตอน "Search in dimension i " ที่ขั้นตอนนี้ตำแหน่งของอนุภาคทุกตัวใน i^h มิติจะถูกปรับปรุงจนกว่าจะเป็นที่น่าพอใจ จนกว่าจะถึงขั้นตอนถัดมา "Calculate Fitness (D-dimension)" จะใช้ข้อมูลทุกมิติของอนุภาคในการหาค่า Fitness ในขั้นตอนนี้ จากนั้นก็จะทำเช่นนี้กับมิติถัดมาจนถึงมิติสุดท้าย (i ก็จะมีค่าเท่ากับ D) การทำงานของรูปจะหยุดลง แล้วมาทำงานต่อที่ขั้นตอน "Output Global Best Value and Stop"

กลยุทธ์นี้เป็นการแบ่ง (Divided) เพื่อการพิชิต (Conquer) หลังจากทุกมิติของฟังก์ชันถูกแบ่ง ออกเป็นองค์ประกอบหลายๆองค์ประกอบ อนุภาคก็จะค้นหาที่ละมิติเพื่อหาจุดที่ดีที่สุดของมิติ นั้น และเมื่ออนุภาคต้องการค้นหาในมิติถัดมาก็จะใช้จุดที่ดีที่สุดของมิติก่อนหน้ามาใช้ในการค้นหา ซึ่ง ถือเป็นสิ่งที่ดีสำหรับการหาค่าที่ดีที่สุด หลังจากที่ได้จุดที่ดีที่สุดในทุกมิติแล้วก็จะนำไปสู่คำตอบที่ดีที่สุดของกลยุทธ์นี้ ซึ่งกลยุทธ์นี้สามารถแก้ปัญหาที่เรียกว่า "Two Steps Forward, One Step Back" [2] ซึ่งสามารถพบได้ใน PSO ปัญหานี้คือเมื่อ PSO ทำการค้นหาในฟังก์ชัน 3 มิติ ในการค้นหา 2 มิติ แรกจะได้ค่าที่ดีขึ้นแต่ในมิติที่ 3 จะได้ค่าที่ไม่ดี ซึ่งหากพิจารณาที่ละมิติปัญหานี้ก็จะลดลงเพราะได้ ทำการบันทึกตำแหน่งที่ดีที่สุดของแต่ละมิติ

2.3 กลยุทธ์การเรียนรู้แบบเพิ่มขึ้นทีละรอบ (Iterative Incremental Learning Strategy)



รูปที่ 2.6 ฟังก์ชันการทำงานของกลยุทธ์การเรียนรู้แบบเพิ่มขึ้นทีละรอบ [2]

กลยุทธ์การเรียนรู้แบบเพิ่มสามารถนำมาใช้ทำงานได้หลายๆรอบเพื่อใช้ในการหาค่าที่ดีที่สุด รูปที่ 2.6 จะแสดงฟังก์ชันการทำงานของกลยุทธ์การเรียนรู้แบบเพิ่มขึ้นทีละรอบ ซึ่งได้นำกลยุทธ์การเรียนรู้แบบเพิ่มมาใช้หลายๆรอบ รูปในสุดของฟังก์ชันการทำงานคือกลยุทธ์การเรียนรู้แบบเพิ่ม เหมือนกับรูปที่ 2.5 เมื่อทุกมิติของฟังก์ชันถูกค้นหาเสร็จแล้ว รูปในสุดก็จะหยุดการทำงาน ในรูป ถัดมา j คือจำนวนครั้งที่ทำซ้ำไปแล้วและถูกตั้งค่าไว้ให้เริ่มต้นที่ 1 และจะถูกเพิ่มค่าขึ้นไปเรื่อยๆ จนกว่าจะครบตามเงื่อนไข ถัดมา G คือจำนวนครั้งที่ทำซ้ำ เมื่อรูปนี้เริ่มต้นการทำงาน จะทำการใช้ กลยุทธ์การเรียนรู้แบบเพิ่มพร้อมกับเพิ่มค่า j ขึ้นไปเรื่อยๆจนกระทั่ง j มีค่าเท่ากับ G ก็จะแสดงค่าที่ ดีที่สุดของกลยุทธ์นี้ จำนวนครั้งที่ทำซ้ำ (G) สามารถถูกเรียกอีกชื่อหนึ่งว่า Generation และเมื่อทำ

การค้นหาใน 1 Generation แล้วนั้นที่ Generation ถัดมาก็จะนำคำตอบของ Generation ก่อนหน้ามาใช้ในการหาค่าที่ดีที่สุด

Incremental attribute based particle swarm optimization (IAPSO) ถือเป็นตัวอย่างอัลกอริทึมที่ใช้กลยุทธ์การเรียนรู้แบบเพิ่มขึ้นทีละรอบ ซึ่งจะเป็นอัลกอริทึมที่ถูกนำมาเปรียบเทียบกับงานวิจัยนี้ และจะอธิบายรายละเอียดของอัลกอริทึม ในบทถัดไป

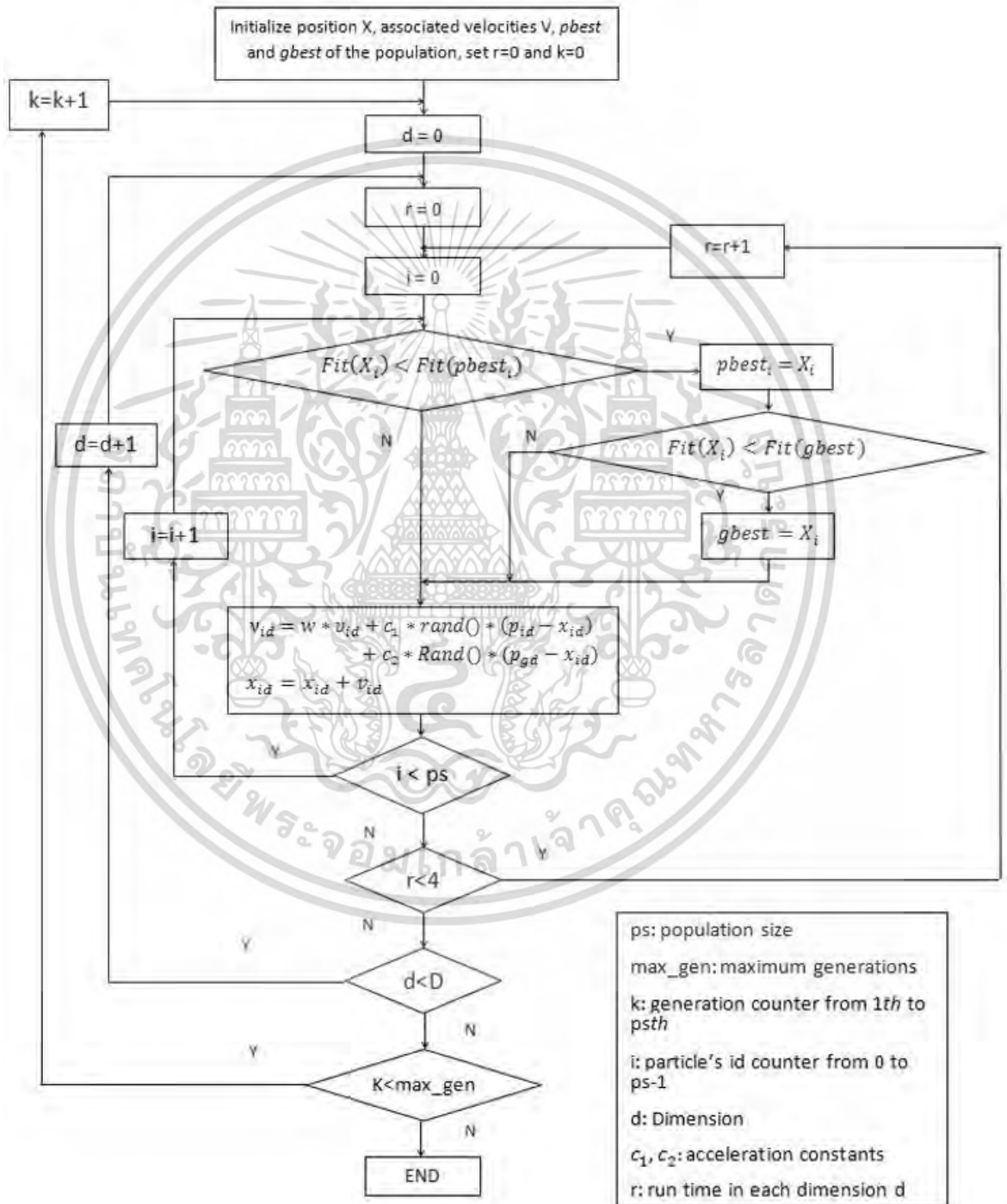


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

งานวิจัยที่เกี่ยวข้อง

3.1 อัลกอริทึม Incremental attribute based particle swarm optimization (IAPSO)



รูปที่ 3.1 ฟังก์ชันการทำงานของ IAPSO [2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IAPSO จะใช้กลยุทธ์การเรียนรู้แบบเพิ่มขึ้นที่ละรอบกับ PSO เพื่อที่จะค้นหาทุกๆมิติโดยการค้นหาทีละมิติ ซึ่งผังการทำงานของ IAPSO จะปรากฏในรูปที่ 3.1 ในขั้นตอนแรกจะกำหนดตำแหน่งเริ่มต้น, ความเร็ว และค่า PBest ของอนุภาคแต่ละตัวภายในพื้นที่การค้นหาและเซตของค่าที่ดีที่สุดของฝูงในแต่ละ Generation ของฝูง ใน 1 มิติจะมีการปรับตำแหน่งและความเร็วของอนุภาคทุกตัวตามสมการที่ 3.1 และ 3.2 ทั้งหมด 4 ครั้ง ซึ่ง 4 ครั้งถือเป็นจำนวนที่เหมาะสมสำหรับงานวิจัยนี้ ค่า Fitness จะถูกคิดหลังจากที่มีการปรับเปลี่ยน จากนั้นก็จะทำวิธีการเช่นนี้เป็น Generation อีกทีเป็นจำนวนเท่ากับ max_gen ครั้ง ซึ่งคำตอบจะแสดงออกมาหลังจากที่ทำ Generation สุดท้ายเสร็จแล้ว

การใช้กลยุทธ์การเรียนรู้แบบเพิ่มขึ้นที่ละรอบทำให้ IAPSO ใช้ข้อมูลของแต่ละมิติ ซึ่งจะช่วยให้หลีกเลี่ยงจุดติดที่สุดเฉพาะที่ได้ในหลายๆมิติของฟังก์ชัน อนุภาคทุกตัวปรับเปลี่ยนตำแหน่ง, ความเร็ว หลายครั้ง เพื่อให้ได้ค่าที่ดีที่สุดของมิตินั้น ซึ่งทำให้ IAPSO ได้ค่าที่ดีจากแต่ละมิติและเป็นผลดีที่จะนำไปสู่คำตอบที่ดีกว่า PSO ปกติที่ใช้ข้อมูลจากทุกมิติในเวลาเดียวกัน

กลยุทธ์นี้เพิ่มประสิทธิภาพการสำรวจของ PSO ซึ่งแสดงให้เห็นว่าอัลกอริทึมนี้มีการค้นหาพื้นที่ใหม่ จากการทดลองพบว่าเมื่อจำนวนมิติเพิ่มขึ้นจาก 50 เป็น 100 ในงานวิจัยนี้ IAPSO สามารถให้ค่าที่เสถียรกว่า PSO ในฟังก์ชันที่มีหลากหลายรูปแบบ หมายความว่าสามารถค้นหาคำตอบได้ดีถึงแม้ว่าฟังก์ชันจะมีจำนวนมิติมากขึ้น

แต่ทั้งนี้ IAPSO ยังคงประสบปัญหาอยู่ 3 ประการคือ

- 1). อัลกอริทึมจำเป็นต้องมีการกำหนดค่าเริ่มต้นก่อนเพื่อใช้สำหรับการค้นหาทีละมิติ
- 2). อัลกอริทึมไม่มีการกำหนดขอบเขตการเคลื่อนที่ของอนุภาคในมิติที่ไม่ได้ค้นหา ส่งผลให้การค้นหาทีละมิติได้ประสิทธิภาพไม่เท่าที่ควร
- 3). หาก Generation ก่อนหน้าค้นหาได้ดี จะส่งผลให้ Generation ถัดไปค้นหาค่าที่ดีกว่าได้ยาก

3.2 อัลกอริทึม One-dimensional Searching-based Particle Swarm Optimization (OPSO)

ปัจจุบันได้มีการพัฒนา PSO เพื่อเพิ่มประสิทธิภาพในการค้นหา [6] OPSO [7] ถือเป็นอีกหนึ่งอัลกอริทึมที่ใช้กลยุทธ์การค้นหาทีละมิติร่วมกับกลยุทธ์หลักการวิเคราะห์องค์ประกอบ

3.2.1 ความรู้พื้นฐานที่เกี่ยวข้อง

ก. ลักษณะของปัญหาในเบื้องต้น

เพื่อให้งานวิจัยนี้มีมาตรฐานโดยทั่วไปนั้น ผู้จัดทำจำเป็นต้องอธิบายลักษณะของปัญหา ก่อน

$$y = f(x), x \in S \quad (3.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ $S \in \mathbb{R}^D$ เป็น set ใดๆ, $x = (x_1, x_2, \dots, x_D)^T$ เป็นตัวแปรตัดสินใจ และ D คือจำนวนมิติ โดยปกติ x ใดๆ ในที่นี้ จะถูกกำหนดข้อจำกัดไว้ ในที่นี้จะขอยกตัวอย่างเป็น x_j เช่น

$$L_j \leq x_j \leq U_j, \quad j = 1, 2, \dots, D \quad (3.2)$$

ข. กลยุทธ์การค้นหาทีละมิติ (One-dimensional Search Strategy)

จากสมการที่ 3.1 ภายใต้กระบวนการทำงาน(Iteration) k เส้นตรงซึ่งลากผ่านจุด $x^{(k)}$ ลงสู่จุด $d^{(k)}$ จะกำหนดไว้ดังนี้

$$L = \{x | x = x^{(k)} + \lambda d^{(k)}, -\infty < \lambda < +\infty\} \quad (3.3)$$

เพื่อค้นหาค่าที่ต่ำที่สุดของปัญหา $f(x)$ ตามเส้นตรง L จึงนำมาสู่สมการเอกของงานวิจัยนี้

$$\varphi(\lambda_k) = \min_{\lambda \geq 0} f(x^{(k)} + \lambda d^{(k)}) \quad (3.4)$$

λ_k คือค่าที่ต่ำที่สุดของสมการที่ 3.4 ดังนั้นค่าต่ำสุดของปัญหาดั้งเดิม ($f(x)$) ที่กระบวนการทำงาน k ตามเส้นตรง L จะถูกกำหนดได้ดังนี้

$$x^{(k+1)} = x^{(k)} + \lambda_k d^{(k)} \quad (3.5)$$

ค. กลยุทธ์หลักการวิเคราะห์องค์ประกอบ(Principal Component Analysis Strategy)

ในส่วนนี้เราจะพิจารณาการเปลี่ยนแปลงเชิงเส้นตรงของ s -variable

$$\begin{cases} Z_1 = a'_1 X = a_{11}X_1 + a_{21}X_2 + \dots + a_{s1}X_s, \\ Z_2 = a'_2 X = a_{12}X_1 + a_{22}X_2 + \dots + a_{s2}X_s \\ \dots \dots \dots \dots \dots \dots \\ Z_s = a'_s X = a_{1s}X_1 + a_{2s}X_2 + \dots + a_{ss}X_s \end{cases} \quad (3.6)$$

เมื่อ a_j คือค่าสัมประสิทธิ์ของการเปลี่ยนแปลงเชิงเส้น ดังนั้นผลรวมของเมทริกซ์ความแปรปรวนร่วม(Covariance Matrix)ของ s -variable จะถูกใช้ในกลยุทธ์นี้ จากนั้นจะสามารถสรุปโมเดลการเปลี่ยนแปลงเชิงเส้นข้างต้น ได้ดังนี้

$$\text{Var}(Z_i) = a'_i \Sigma a_i, \quad i = 1, 2 \dots s \quad (3.7)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{Cov}(Z_i, Z_j) = a_i' \Sigma a_j, \quad i, j = 1, 2 \dots s \quad (3.8)$$

เป้าหมายของ PCA คือ เพื่อเปลี่ยน Set ของ X_i ทุกตัวให้กลายเป็น Z_i ($i=1,2,\dots,s$) เพื่อแสดงปริมาณข้อมูลที้องค์ประกอบมีอยู่ จึงได้เลือกใช้ความแปรปรวนสำหรับงานนี้ ($\text{Var}(Z_i)$) ทฤษฎีความน่าจะเป็นบอกเราว่า $\text{Var}(Z_i)$ จะมีค่าเข้าสู่อนันต์หากไม่มีการจำกัดค่าสัมประสิทธิ์ a_i ดังนั้นผู้จัดทำจึงกำหนดการจำกัดเป็นมุมฉากบน a_i ดังนี้ $a_i' a_i = 1$ ($i=1,2,\dots,s$)

เมื่อองค์ประกอบแรกส่งผลต่อองค์ประกอบอื่นๆทั้งหมดอย่างไม่สมมาเสมอ องค์ประกอบอื่น (Z_j) จะถูกใช้ อย่างไรก็ตาม Z_i และ Z_j ยังคงประกอบไปด้วยข้อมูลทั้งหมดของฟังก์ชัน ยิ่งไปกว่านั้นจะมีการเพิ่มข้อจำกัดในสมการที่ 3.8 หากมีการสร้างข้อมูลการส่งผลกระทบต่อองค์ประกอบที่ i^{th} แต่ไม่พบในองค์ประกอบที่ j^{th} ดังนั้นเพื่อแก้ปัญหาดังกล่าวองค์ประกอบที่ i^{th} จะถูกเปลี่ยนเป็นโมเดลทางคณิตศาสตร์ดังนี้

$$\begin{aligned} & \max \text{Var}(a_i' X) \\ & \text{s.t.} \begin{cases} a_i' a_i = 1 \\ a_i' \Sigma a_j = 0 \end{cases} \end{aligned} \quad (3.9)$$

นี่ถือเป็นปัญหาที่สำคัญเนื่องด้วยมีข้อจำกัดถึง 2 ข้อ ซึ่งสามารถแก้ไขได้ด้วยวิธี Lagrange Multiplier สำหรับวิเคราะห์องค์ประกอบแรก Z_1 ดังนั้นข้อจำกัดแรกจึงมีผลเป็นอย่างมากจึงเป็นที่มาของสมการที่ 3.10

$$\phi(a_1) = \text{Var}(a_1' X) - \lambda(a_1' a_1 - 1) = a_1' \Sigma a_1 - \lambda(a_1' a_1 - 1) \quad (3.10)$$

จะเห็นได้ชัดว่า $a_1 \neq 0$ และ $|\Sigma - \lambda I| = 0$ กล่าวคือเพื่อการวิเคราะห์องค์ประกอบ Z_1 จะต้องหาเวกเตอร์เจาะจง (Eigenvector) จากเมทริกซ์ความแปรปรวนร่วมของ X จากนั้นฟังก์ชันที่สนใจจะกลายมาเป็นสมการที่ 3.11

$$\text{Var}(a_1' X) = a_1' \Sigma a_1 = a_1' \lambda_1 a_1 = \lambda_1 \quad (3.11)$$

ซึ่งหมายความว่าค่ามากที่สุดของ $\text{Var}(a_1' X)$ มีค่าพอที่จะหาเวกเตอร์เจาะจงของค่าเจาะจง (Eigenvalue) ที่มากที่สุด และองค์ประกอบที่เหลือสามารถวิเคราะห์ได้โดยการใช้เวกเตอร์เจาะจงกับค่าเจาะจงมากที่สุดที่ j^{th} ยิ่งไปกว่านั้นข้อจำกัดที่ 2 ในสมการที่ 3.9 จะถูกแก้ไขโดยอัตโนมัติ หาก a_i และ a_j มีเวกเตอร์เจาะจงของผลรวมเมทริกซ์ความแปรปรวนร่วมต่างกัน

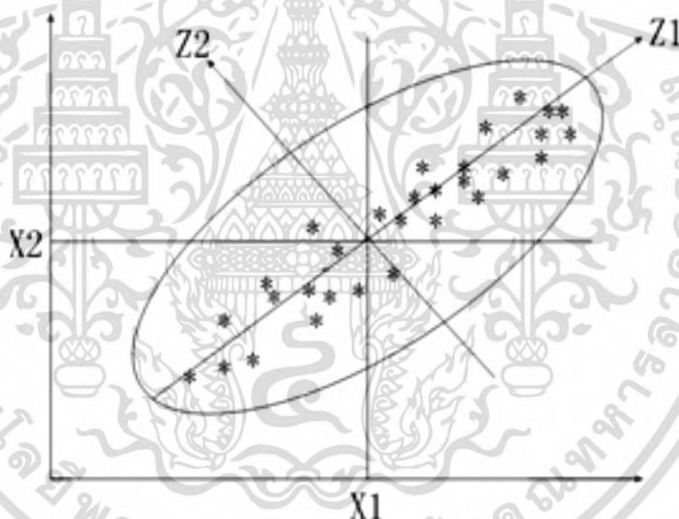
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 การพัฒนา PSO ด้วย PCA และ ODS

OPSO จะใช้ Principal Component Analysis Strategy (PCA) และ One-dimensional Search Strategy (ODS) ใน PSO เพื่อเป็นการเน้นความสามารถในการค้นหาของทิศทางความเร็วตามปกติ ผู้จัดทำได้มีแนวความคิดใหม่สำหรับงานนี้ คืออนุภาคควรมีประสิทธิภาพการค้นหาในทุกๆที่ตลอดตามทิศทางการค้นหาของ PCA

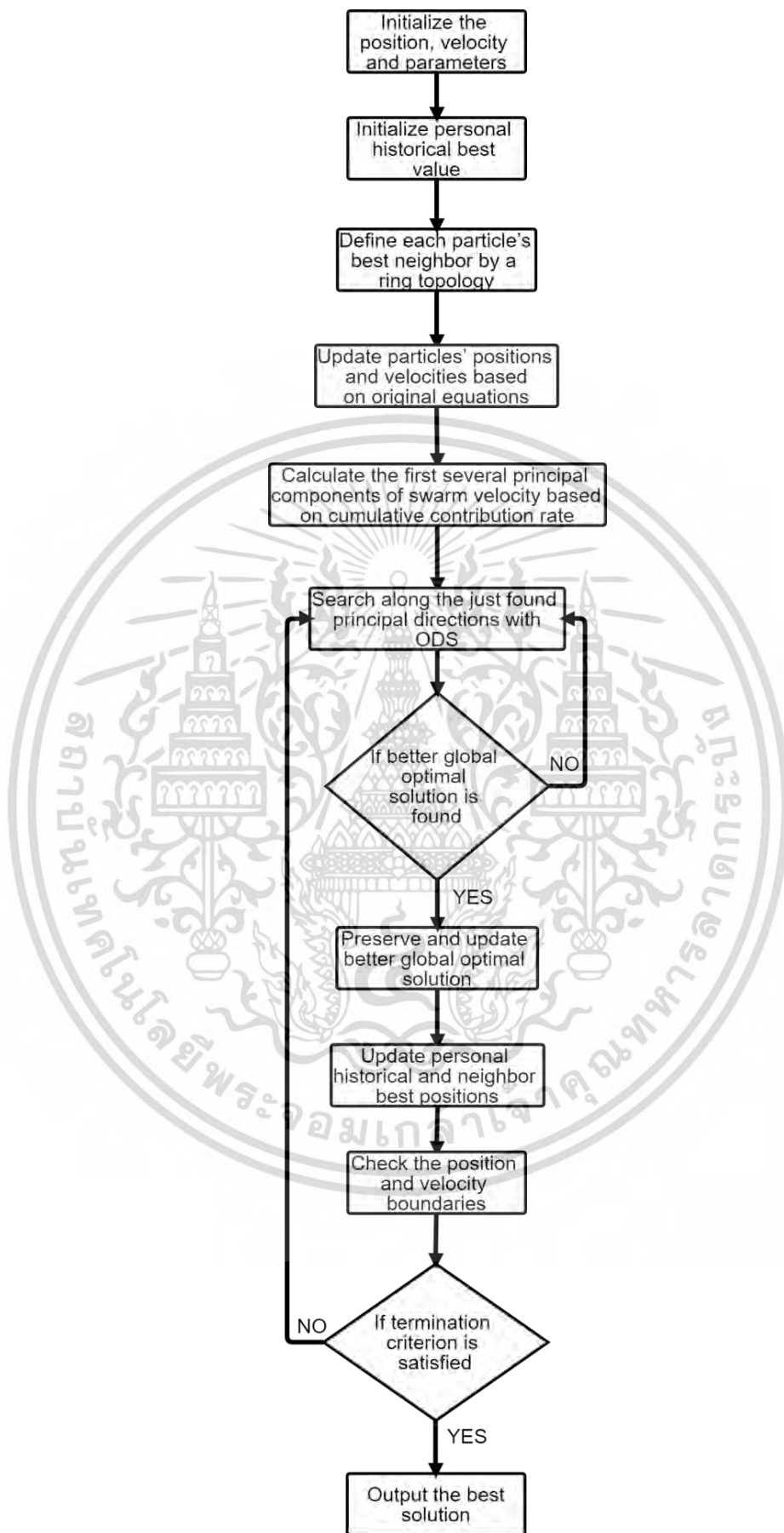
เห็นได้ชัดว่าแนวคิดที่กล่าวมาเมื่อชักรุ่นนี้ค่อนข้างโลกและไม่มีประสิทธิภาพจนเกินไป หากอนุภาคค้นหาทุกที่ตลอดตามทิศทางความเร็วปกติ อย่างไรก็ตาม PCA สามารถทำให้แนวคิดนี้เป็นไปได้ในงานวิจัยนี้ ผลการวิเคราะห์ทางสถิติได้ชี้ให้เห็นว่า PCA สามารถแยกองค์ประกอบข้อมูลหลายตัวแปรออกเป็นเส้นทางต่างๆ โดยจะรักษาเส้นทางแรกที่มีข้อมูลส่วนใหญ่ของต้นฉบับ ซึ่งจะแสดงตัวอย่างของ PCA ในส่วนถัดไป

สมมติว่าความเร็วเริ่มต้นของอนุภาคถูกกำหนดไว้เป็นคู่อันดับ ดังนั้นเราจะสังเกตเห็นองค์ประกอบ Z_1 และ Z_2 ดังรูปภาพที่ 3.2 สำหรับการแก้ปัญหา 2 มิติใน 1 Iteration



รูปที่ 3.2 ตัวอย่าง PCA ในพื้นที่ 2 มิติ [7]

จากรูปที่ 3.2 เราสามารถทราบทิศทางการบินของอนุภาคทุกตัว ซึ่งอาจซ้ำในทิศทาง Z_1 นั้นหมายความว่าเราจำเป็นต้องใช้อนุภาคเพียง 1 ตัวในการสำรวจทิศทางการค้นหาของ PCA อันแรก แทนที่จะใช้อนุภาคค้นหาตามทิศทางความเร็วตามปกติ ในงานวิจัยนี้เราใช้อนุภาคที่ดีที่สุดในการค้นหาตลอดทิศทางแรกของ PCA ถัดมาจะเป็นขั้นตอนการทำงานของ OPSO



รูปที่ 3.3 ฟังก์ชันการทำงานของ OPSO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.3 จะอธิบายการทำงานของ OPSO โดยจะเริ่มต้นด้วยการกำหนด ตำแหน่ง ความเร็ว และ พารามิเตอร์ต่างๆ จากนั้นจึงวิเคราะห์หาความสำคัญของฟังก์ชันในทุกๆมิติและกำหนดบริเวณที่มีความสำคัญที่สุดด้วยกลยุทธ์ PCA โดยบริเวณที่ได้จะมีลักษณะเป็นรูปวงรี เมื่อได้บริเวณที่มีความสำคัญที่สุดแล้วอัลกอริทึมจะค้นหาบริเวณนั้นด้วยกลยุทธ์ ODS จนกว่าจะพบคำตอบที่ดีกว่าในปัจจุบัน(หากยังไม่พบค่าที่ดีกว่าในปัจจุบัน อัลกอริทึมจะยังคงค้นหาต่อไปเรื่อยๆ) เสร็จแล้วอัลกอริทึมจะทำการค้นหาในบริเวณที่มีความสำคัญส่วนถัดไป ซึ่งได้รับมาจากการวิเคราะห์หาความสำคัญของกลยุทธ์ PCA จนครบทุกส่วน จากนั้นก็จะแสดงคำตอบที่ดีที่สุดจากการค้นหาทุกส่วนออกมา

ทั้งนี้การค้นหาของ OPSO ยังคงประสบปัญหาในเรื่องการใช้ Function Evaluation Times มาก เนื่องจาก OPSO จะทำงานจนกว่าจะตรงตามเงื่อนไขเท่านั้นจึงจะหยุดการค้นหา และที่สำคัญ OPSO ไม่มีการค้นหาซ้ำจึงส่งผลให้คำตอบขาดเสถียรภาพ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

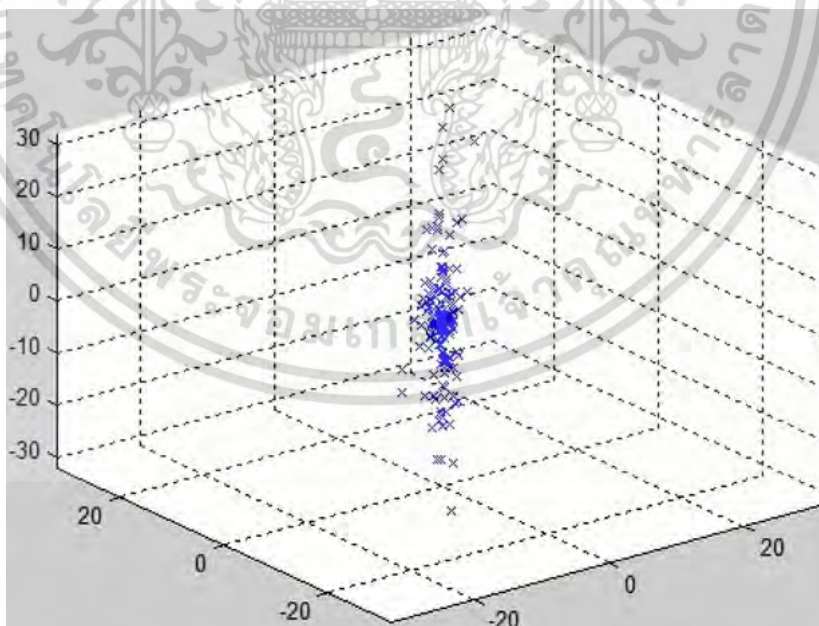
งานวิจัยที่นำเสนอ

งานวิจัยนี้มีชื่อเต็มว่า Improving Particle Swarm Optimization by Using Incremental Attribute Learning and Centroid of Particle's Best Positions (IAC-PSO) ซึ่งจะประกอบด้วยส่วนประกอบหลักทั้งหมด 5 ส่วนหลัก จากนั้นในส่วนท้ายจะแสดงภาพรวมการทำงานของอัลกอริทึมด้วยผังการทำงาน

4.1 การกำหนดจุดเริ่มต้น (Position Initialization)

การค้นหาที่ละมิติจำเป็นจะต้องมีการกำหนดตำแหน่งในมิติอื่นๆที่ยังไม่ได้ค้นหา ดังนั้นการกำหนดจุดเริ่มต้นที่ดีจะช่วยเพิ่มโอกาสที่จะค้นหาในจุด Global Optima ภายในขอบเขตที่กำหนด โดยจะใช้ Conventional PSO ในช่วงแรกของการทำงาน (Preliminary Run) เพื่อหาจุดเริ่มต้นโดยจะค้นหาประมาณ 7,500-15,000 Iteration โดยอัลกอริทึมจะใช้จุดเริ่มต้นเหมือนกันในทุกๆ Generation

4.2 การยืดหยุ่นจุดเริ่มต้นของอนุภาค (Processed Attribute Boundary)

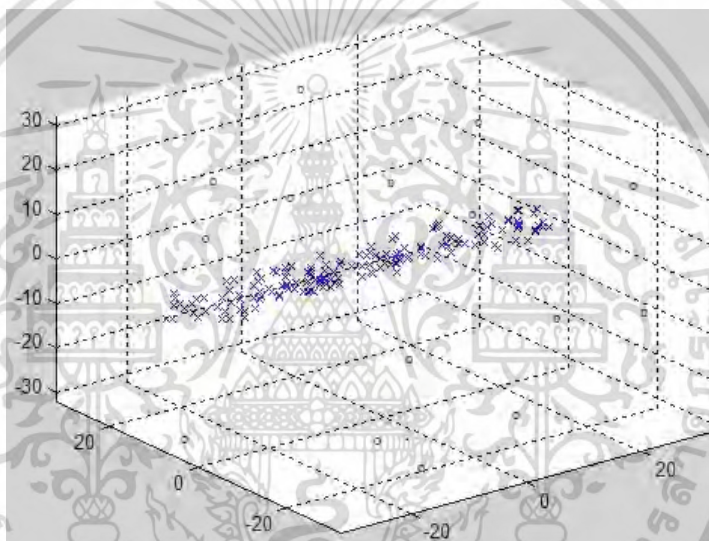


รูปที่ 4.1 การค้นหาที่ละมิติจของ IAC-PSO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลยุทธ์การเรียนรู้แบบเพิ่มจะค้นหามิติถัดไปโดยจะขึ้นอยู่กับตำแหน่งที่ดีที่สุดที่ค้นพบจากมิตีก่อนหน้านั้น แต่ทั้งนี้ IAC-PSO จะยึดหยุ่นให้อนุภาคสามารถเคลื่อนที่ได้ภายในช่วงแคบๆ ใกล้เคียงกับตำแหน่งที่ดีที่สุดที่เคยค้นหาในมิตีก่อนหน้านั้น ซึ่งวิธีนี้จะช่วยลดโอกาสที่ IAC-PSO จะติดอยู่ใน Local Optima ทั้งนี้เนื่องจากตำแหน่งที่ดีที่สุดจากการค้นหาในมิตีก่อนหน้าไม่สามารถระบุได้ว่าเป็นตำแหน่งที่ดีที่สุด รูปที่ 4.1 จะแสดงตัวอย่างการค้นหาที่ละมิติของอัลกอริทึม จากภาพในขณะนี้ อัลกอริทึมกำลังค้นหาในแกน z ซึ่งอัลกอริทึมจะกระจายอนุภาคทั่วทั้งแกน z ส่วนแกน x และ y นั้น อัลกอริทึมจะกำหนดตำแหน่งให้อยู่ภายในอยู่ในช่วงแคบๆ

4.3 การแบ่งกลุ่มอนุภาค (Swarm Classification)



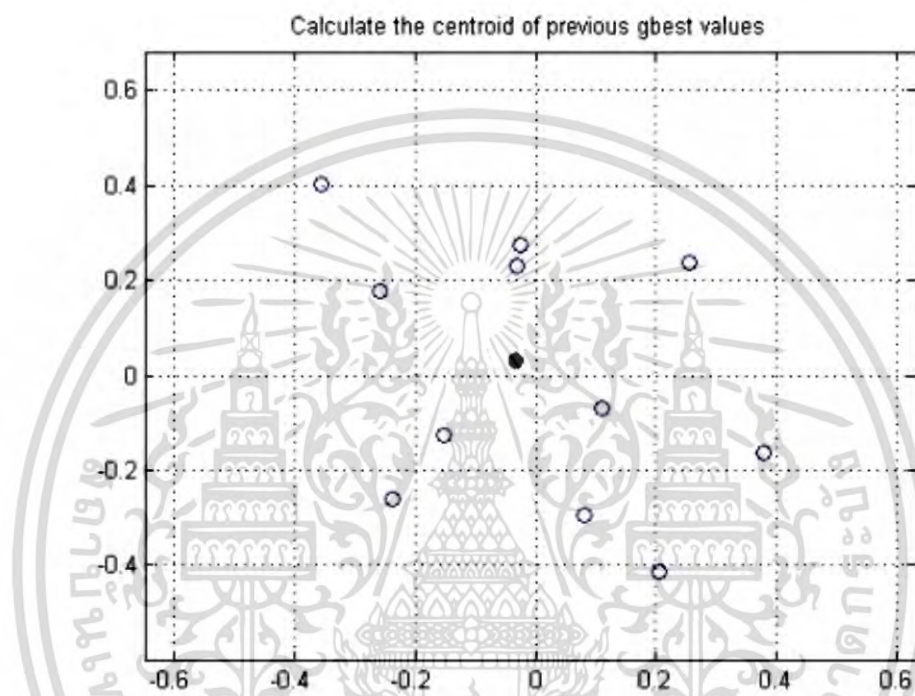
รูปที่ 4.2 การแบ่งกลุ่มอนุภาคของ IAC-PSO

ในการค้นหาที่ละมิตินั้นจะแบ่งอนุภาคออกเป็น 2 กลุ่มหลักโดยใช้ความน่าจํานวนหนึ่งในการจัดแบ่ง โดยเมื่อเริ่มทำการค้นหามิติใหม่กลุ่มแรกจะถูกกำหนดตำแหน่งเริ่มต้นตามหัวข้อ 4.2 ส่วนอีกกลุ่มหนึ่งกำหนดตำแหน่งเริ่มต้นด้วยการสุ่มภายใน Search Space ยกตัวอย่างเช่น กำหนดให้ความน่าจะเป็นที่ใช้ในการจัดแบ่งกลุ่มเท่ากับ 0.6 จากนั้นจะทำการสุ่มค่าในช่วง $[0,1]$ อนุภาคที่สุ่มได้ค่าน้อยกว่า 0.6 จะถูกจัดอยู่ในกลุ่มแรก (ประมาณ 60 เปอร์เซ็นต์) และถูกกำหนดตำแหน่งตามหัวข้อ 4.2 ส่วนอนุภาคที่ได้ค่านามากกว่า 0.6 (ประมาณ 40 เปอร์เซ็นต์) จะถูกจัดอยู่ในกลุ่มที่สองแล้วสุ่มกำหนดตำแหน่งภายใน Search Space เมื่ออนุภาคกลุ่มใดกลุ่มหนึ่งค้นหาจุด GBest ใหม่ได้ อนุภาคทั้ง 2 กลุ่มจะช่วยกันค้นหาบริเวณนั้น และช่วงแคบๆ ที่อนุภาคให้อนุภาคเคลื่อนที่ได้จะถูกย้ายมาที่จุดดังกล่าวเมื่ออัลกอริทึมค้นหาในมิติถัดไป รูปที่ 4.2 จะแสดงตัวอย่างการแบ่งกลุ่มของอนุภาค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยตัวอย่างนี้จะใช้ความน่าจะเป็นที่ 0.9 (อนุภาคประมาณ 90% จะถูกกำหนดตำแหน่งตามหัวข้อ 4.2 และอีก 10% สุ่มกำหนดตำแหน่งภายใน Search Space) ภายในภาพ x คือ อนุภาคกลุ่มแรกที่ค้นหาที่ละมิตี (ตามหัวข้อที่ 4.2) ส่วน o คือ อนุภาคกลุ่มที่สอง ซึ่งจะค้นหาทั่วทั้ง Search Space

4.4 การคำนวณจุดเซนทรอยด์ (Centroid Calculation)



รูปที่ 4.3 การหาจุดเซนทรอยด์ในฟังก์ชัน Ackley

ตารางที่ 4.1 ผลการทดลองการเปรียบเทียบการใช้จุดเซนทรอยด์

Point	No-centroid	Centroided	Best known
Best	1.3944E-13	5.7732E-014	0
Median	4.4560E-12	6.8390E-014	0
Worst	9.9647E-11	7.90480E-14	0
Mean	2.18E-11	6.85319E-14	0

อัลกอริทึมมักจะค้นหาจุด GBest ได้ใกล้เคียงกับจุด Global Optima ดังนั้นการหาจุดเซนทรอยด์จากจุด GBest ล่าสุดจะช่วยให้มีโอกาสค้นหาค่าที่ดีขึ้นกว่า GBest ในปัจจุบันและช่วยเร่งการลู่เข้าของอัลกอริทึม ทั้งนี้หากจุดเซนทรอยด์ไม่สามารถให้ค่าที่ดีกว่าจุด GBest ในปัจจุบัน อัลกอริทึมก็จะยังคงใช้ค่า GBest ในปัจจุบันต่อไป รูปที่ 4.3 แสดงตัวอย่างการหาจุดเซนทรอยด์ (จุดสีดำ) จากเอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปเผยแพร่โดยไม่ได้รับอนุญาตเป็นการฝ่าฝืนกฎหมายลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี หากมีข้อสงสัยหรือต้องการข้อมูลเพิ่มเติม กรุณาติดต่อฝ่ายประชาสัมพันธ์ โทร. 02-214-9999 หรือ e-mail: info@kmutt.ac.th

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

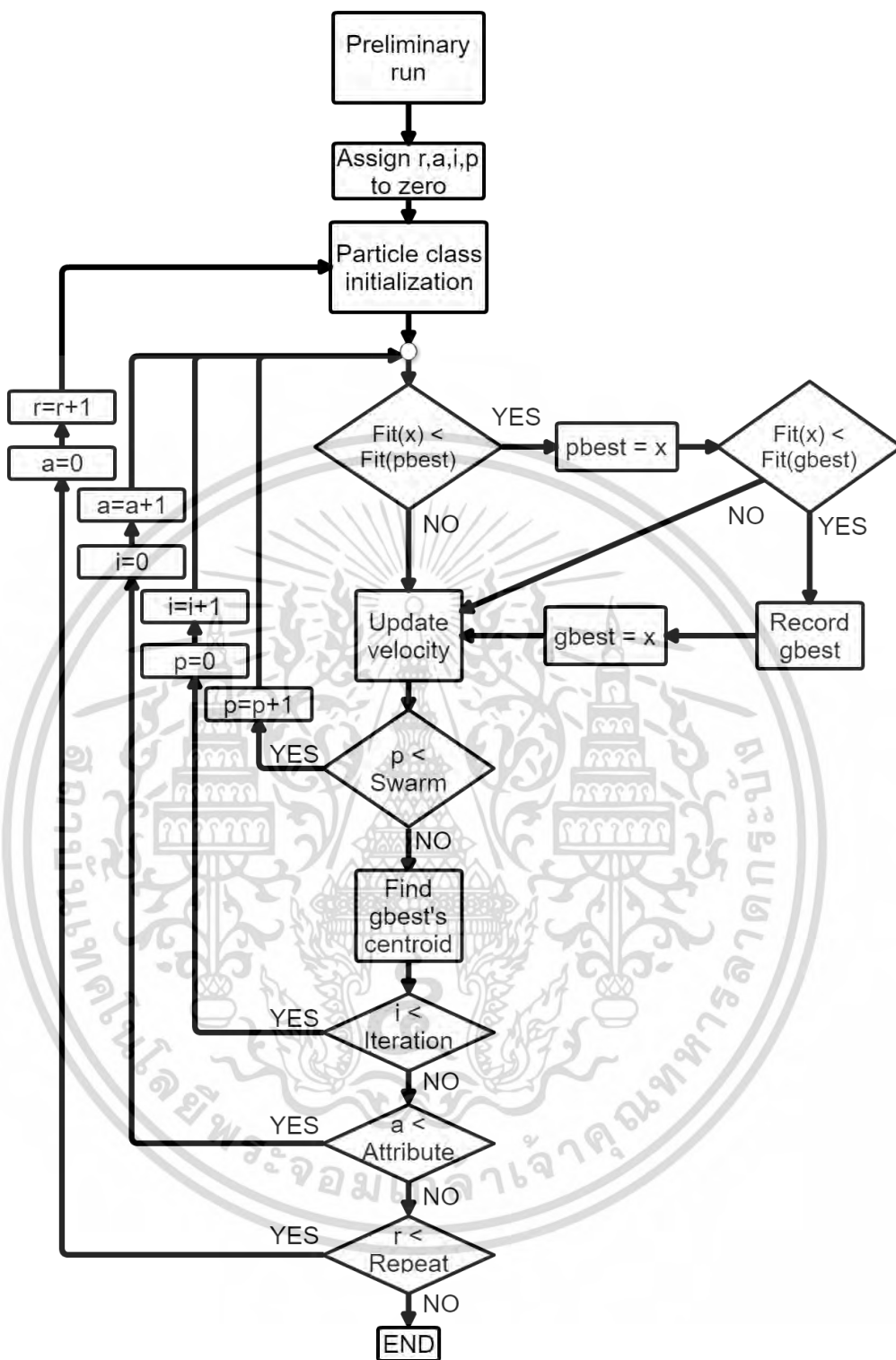
จุด GBest ต่ำสุด (จุดสีขาว) ซึ่งใกล้เคียงกับจุด Global Optima (0,0) ของฟังก์ชัน Ackley ตารางที่ 4.1 จะแสดงผลการเปรียบเทียบการใช้จุดเซนทรอยด์กับฟังก์ชัน Ackley ที่ 100 มิติ จะสังเกตเห็นว่าการใช้จุดเซนทรอยด์จะช่วยให้อัลกอริทึมค้นหาคำตอบได้ดียิ่งขึ้นในทุกๆค่าสถิติ โดยเฉพาะอย่างยิ่งในค่า Worst และ Mean

4.5 การวนรอบการทำงาน (Generation Repetition)

IAPSO [2] จะใช้กลยุทธ์การเรียนรู้แบบเพิ่มหลายๆรอบ ซึ่งแต่ละรอบจะถูกเรียกว่า Generation ดังนั้นวิธีนี้จะช่วยให้สามารถค้นหาค่าที่ดีที่สุดได้เป็นอย่างดี แต่ทั้งนี้ IAC-PSO จะทำการรีเซ็ตค่าที่ได้จาก Generation ก่อนหน้าทั้งหมดเพื่อเพิ่มโอกาสในการค้นหาค่าที่ดีกว่า Generation ก่อนหน้า เนื่องจากหากว่า Generation ก่อนหน้าสามารถค้นหาค่าได้ดีมากๆ จะส่งผลให้ Generation ถัดไป ค้นหาค่าที่ดีกว่าได้ยาก

ภาพรวมการทำงานของ IAC-PSO จะถูกอธิบายด้วยผังงานในรูปที่ 4.4 จากรูป p คือ หมายเลขของอนุภาคที่กำลังประมวลผลในปัจจุบัน, i คือ หมายเลขของ iteration ที่ดำเนินการในปัจจุบัน, a คือหมายเลขของมิติที่ค้นหาอยู่ในปัจจุบัน และ r คือ หมายเลข Generation Repetition ที่ดำเนินการอยู่ในปัจจุบัน

ในช่วงแรกอัลกอริทึมจะเริ่มต้นด้วยการทำ Preliminary Run ด้วย Iteration จำนวนหนึ่งเพื่อกำหนดจุดเริ่มต้น ซึ่งจุดเริ่มต้นนี้จะถูกใช้เป็นจุดเริ่มต้นในทุก Generation ถัดมาที่ขั้นตอน “Particle class initialization” อัลกอริทึมจะแบ่งกลุ่มอนุภาคและกำหนดตำแหน่งให้กับทั้ง 2 กลุ่มดังเช่นที่ได้อธิบายในหัวข้อ 4.3 แล้วจึงค่อยๆค้นหาที่ละมิติพร้อมกับคำนวณจุดเซนทรอยด์เพื่อเปรียบเทียบกับจุด GBest ในปัจจุบัน โดยจะคำนวณทุกๆกระบวนการทำงาน เมื่ออัลกอริทึมทำงานไปได้ระยะหนึ่งจะสามารถระบุจุดที่ดีที่สุด ในมิติได้จากนั้นจึงค้นหามิติถัดไป ในขณะที่อนุภาคกำลังค้นหามิติถัดไป อนุภาคสามารถเคลื่อนที่ได้ในช่วงแคบๆ ใกล้เคียงกับจุดที่ดีที่สุดของมิติที่เคยค้นหาก่อนหน้านี้ เมื่อเสร็จสิ้นกระบวนการจะระบุจุดที่ดีที่สุดของทุกมิติ อัลกอริทึมจะค้นหา Generation ใหม่ทันทีโดยจะเริ่มต้นค้นหาที่จุดเริ่มต้น ซึ่งเป็นจุดที่ได้จากการทำ Preliminary Run โดยจะค้นหาประมาณ 4-7 Generation เพื่อเพิ่มเสถียรภาพให้กับอัลกอริทึม



รูปที่ 4.4 ฟังก์ชันการทำงานของอัลกอริทึม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การทดลองและการเปรียบเทียบผลการทดลอง

5.1 ค่าพารามิเตอร์

เพื่อการเปรียบเทียบค่าพารามิเตอร์จะถูกกำหนดให้มีค่าเท่ากับการทดลองของงานวิจัย IAPSO [2] ใน 5 ตัวแรกของค่าพารามิเตอร์ ซึ่งจะเป็ค่าพารามิเตอร์ทั่วไป และถัดมาจะเป็นค่าพารามิเตอร์เฉพาะของ IAC-PSO

ค่าพารามิเตอร์จะถูกกำหนดไว้ดังนี้

- Population Size: 160 อนุภาค
- Inertia Weight: $w = 0.72984$
- Acceleration Constants: $c1 = c2 = 1.496172$
- Function Evaluation Times: 2,400 สำหรับ 2 มิติ, 60,000 สำหรับ 50 มิติ และ 120,000 สำหรับ 100 มิติ
- Stop Criteria
 - Function Evaluation Times ครบตามกำหนด
 - พบค่า Global Minimum
- Iterations for Preliminary Run: 300 สำหรับ 2 มิติ, 7,500 สำหรับ 50 มิติ และ 15,000 สำหรับ 100 มิติ
- Iterations per Dimension: 150 ต่อมิติ
- Small Slack Region: 5% ของขอบเขต (Search Range)
- Number of Previous GBest for Calculating the Centroid: 10 จุด
- Repetitions: 7 Generations
- Fix Best Position Probability: 0.60

ค่าพารามิเตอร์ที่สำคัญ 2 ค่าจะต้องถูกกำหนดอย่างเหมาะสม อย่างแรกหากช่วงแคบๆสำหรับการค้นหาที่ละมิติ (Small Slack Region) นั้นมีความแคบมากเกินไปจะทำให้อัลกอริทึมพลาดโอกาสที่จะค้นหาจุด Global Optima ที่อยู่ใกล้เคียงกับฝูง อย่างที่สองคือจำนวนจุด GBest ที่ใช้ (Number of Previous GBest for Calculating the Centroid) หากใช้จำนวนมากเกินไปจะทำให้ประสิทธิภาพลดลง

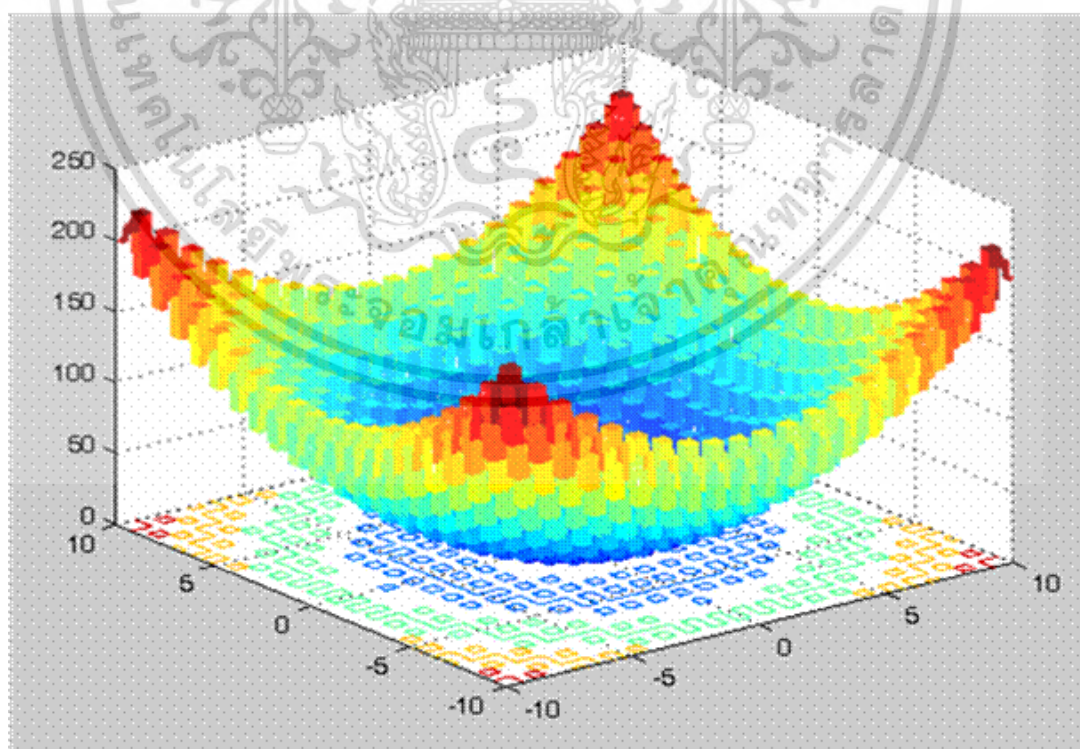
ค่าพารามิเตอร์จำนวนครั้งที่ดำเนินการ (Function Evaluation Times) สามารถคำนวณได้จาก $I_p + R \cdot I \cdot A$ เมื่อ I_p (Iterations for Preliminary Run) คือจำนวน Iteration ที่ใช้ใน Preliminary Run, R (Generation Repetition) คือจำนวน Generation, I (Iterations per dimension) จำนวน Iteration ที่ใช้แต่ละมิติ และ D (Dimension) คือจำนวนมิติ ยกตัวอย่างเช่น เมื่อทำการค้นหาที่ 50 มิติ จำนวนครั้งที่ดำเนินการจะเท่ากับ $7,500 + 7 \cdot 150 \cdot 50 = 60,000$ และเมื่อค้นหาที่ 100 มิติจะเท่ากับ $15,000 + 7 \cdot 150 \cdot 100 = 120,000$

5.2 การทดลอง

การทดลองจะทำการเปรียบเทียบกับ IAPSO และ Conventional PSO ซึ่งในที่นี้จะใช้ Preliminary Run ในการกำหนดจุดเริ่มต้นให้กับ IAPSO เช่นกัน การทดลองจะทดลองทั้งหมด 50 ครั้งและนำเสนอด้วยค่าทางสถิติ ได้แก่ ค่าดีที่สุด (Best), ค่ามัธยฐาน (Median), ค่าแย่ที่สุด (Worst) และ ค่าเฉลี่ย (Mean) โดยจะเปรียบเทียบระหว่าง Conventional PSO, IAPSO และ IAC-PSO ในทุกๆฟังก์ชันที่ใช้ในการทดลอง

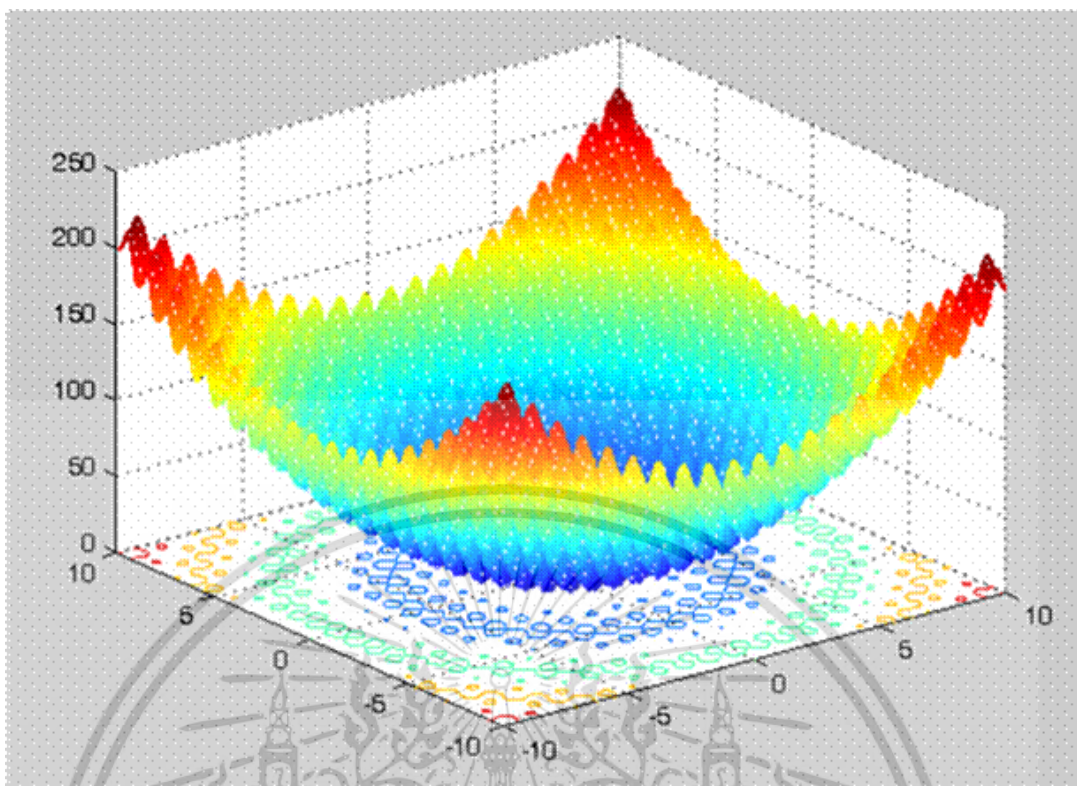
5.3 ฟังก์ชันที่ใช้ทดลอง

5.3.1 ฟังก์ชัน Noncontinuous Rastrigin



รูปที่ 5.1 ฟังก์ชัน Noncontinuous Rastrigin [8]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



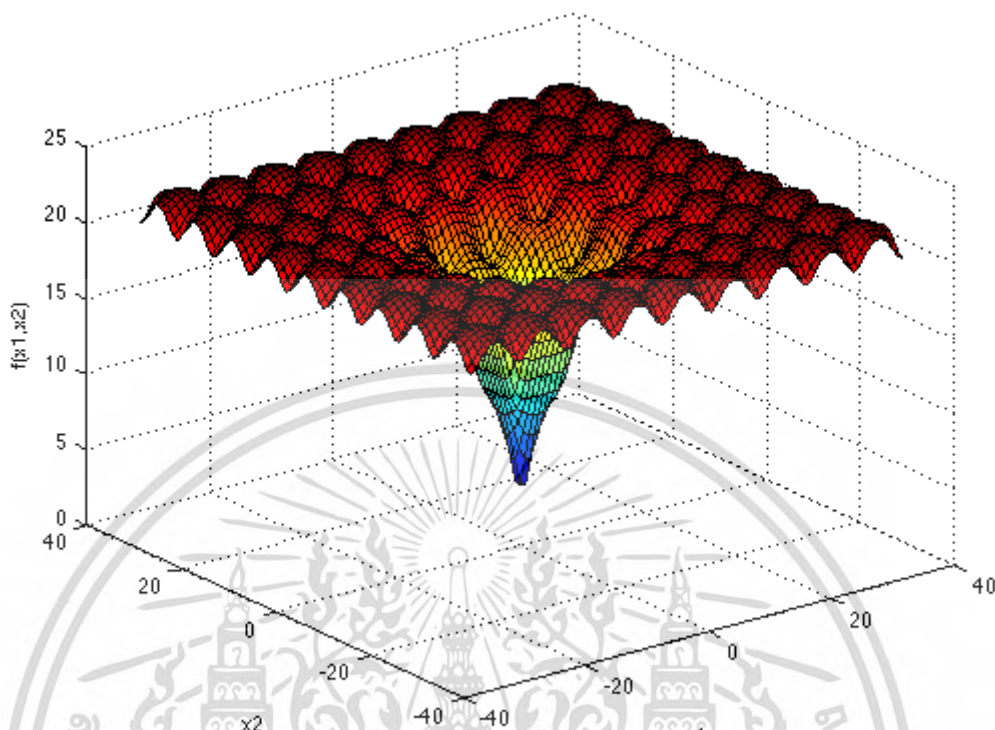
รูปที่ 5.2 ฟังก์ชัน Rastrigin [8]

ตารางที่ 5.1 ฟังก์ชัน Noncontinuous Rastrigin

Function	Equation	Dimension	Range	Global minima	Properties
Noncontinuous Rastrigin	$f(x) = \sum_{i=1}^d [y_i^2 - 10 \cos(2\pi y_i) + 10]$ $y_i = \begin{cases} x_i & x_i < 0.5 \\ \frac{\text{round}(2x_i)}{2} & x_i \geq 0.5 \end{cases}$	50/100	$[-5.12, 5.12]^p$	0	multi mode

ในการทดลองจะใช้ฟังก์ชัน Noncontinuous Rastrigin เท่านั้น ฟังก์ชัน Noncontinuous Rastrigin จัดอยู่ในประเภท Many Local Minima ซึ่งจะประกอบไปด้วยจุด Local Minima จำนวนมหาศาลและมีแนวโน้มต่ำลงเมื่อเข้าใกล้จุด (0,0) รูปที่ 5.1 จะแสดงภาพ 3 มิติของฟังก์ชัน Noncontinuous Rastrigin ซึ่งจะแตกต่างจากฟังก์ชัน Rastrigin ดั้งเดิม (รูปที่ 5.2) ที่มีความต่อเนื่องน้อยกว่าในบริเวณจุด Local Minima รายละเอียดการทดลองฟังก์ชัน Noncontinuous Rastrigin ดังตารางที่ 5.1

5.3.2 ฟังก์ชัน Ackley



รูปที่ 5.3 ฟังก์ชัน Ackley [9]

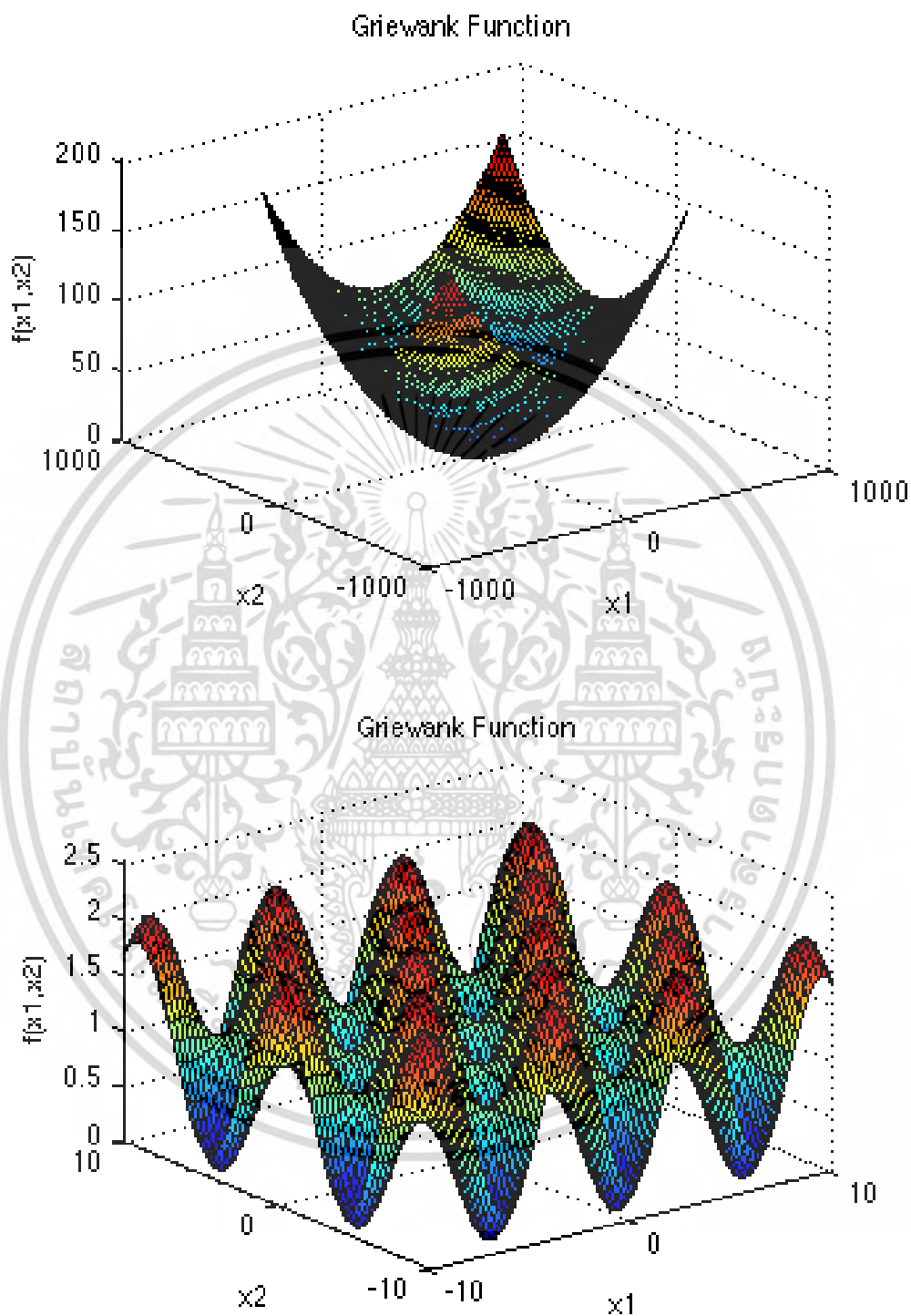
ตารางที่ 5.2 ฟังก์ชัน Ackley

Function	Equation	Dimension	Range	Global minima	Properties
Ackley	$f(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{d} \sum_{i=1}^d x_i^2}\right) - \exp\left(\frac{1}{d} \sum_{i=1}^d \cos(2\pi x_i)\right) + 20 + e$	50/100	$[-32, 32]^D$	0	multi mode

ฟังก์ชัน Ackley จัดอยู่ในประเภท Many Local Minima ซึ่งจะประกอบไปด้วยจุด Local Minima จำนวนมากและที่บริเวณจุด (0,0) จะมีลักษณะเป็นหลุมที่ลึกลงไป รูปที่ 5.3 จะแสดงภาพ 3 มิติของฟังก์ชัน Ackley รายละเอียดการทดลองฟังก์ชัน Ackley ดังตารางที่ 5.2

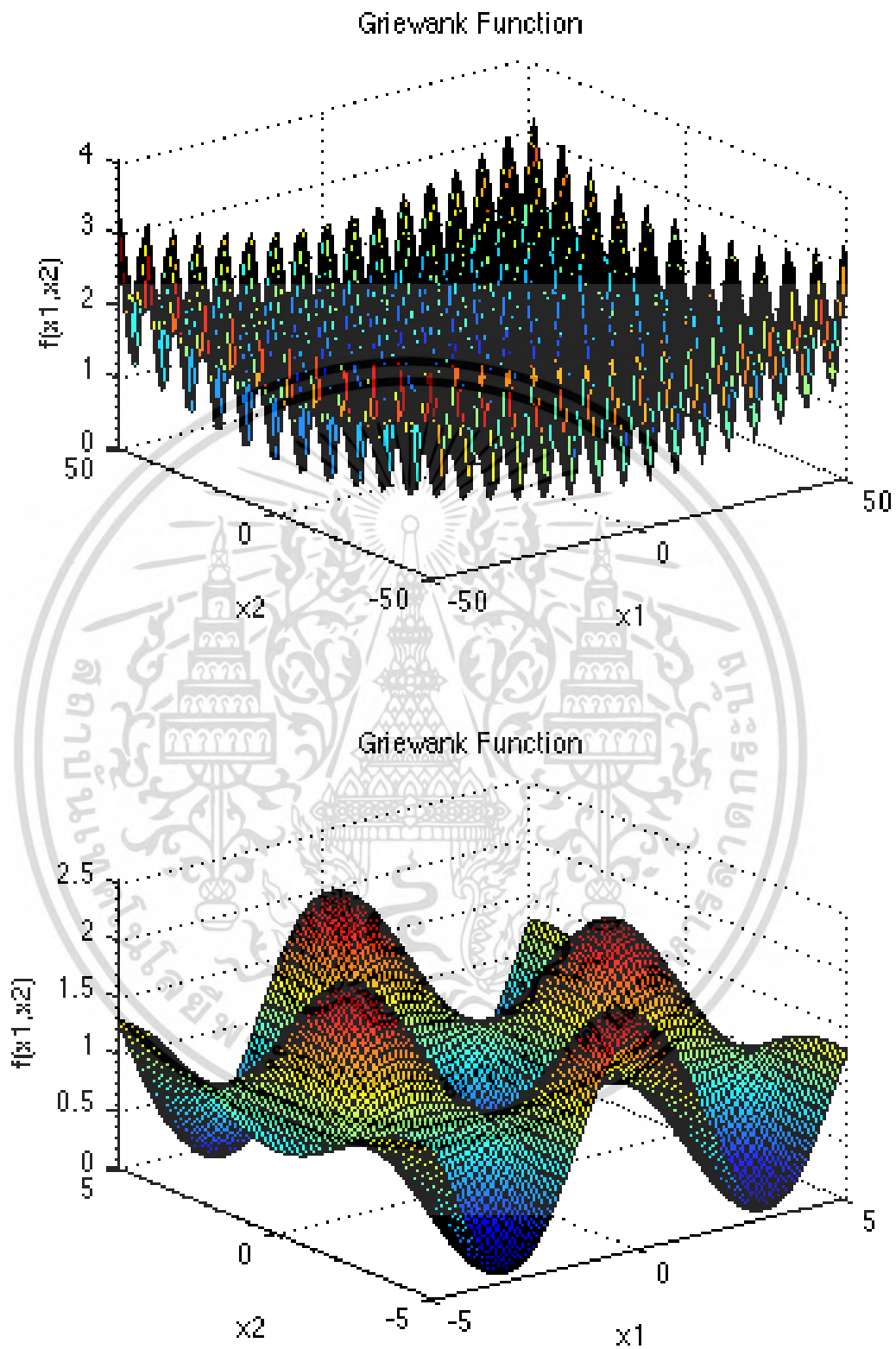
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.3 ฟังก์ชัน Griewank



รูปที่ 5.4 ฟังก์ชัน Griewank [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.4 (ต่อ)

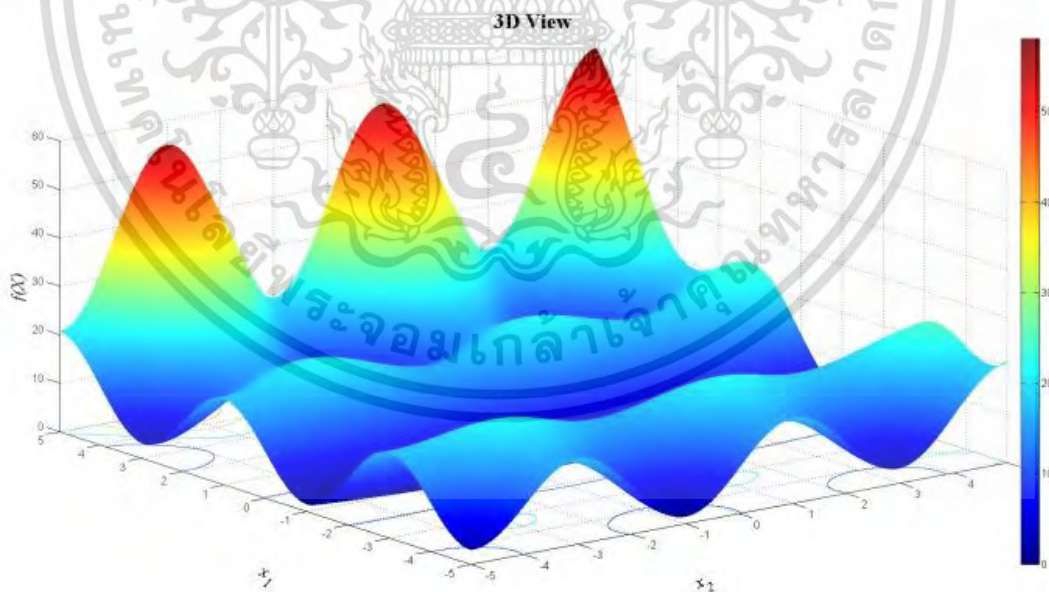
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.3 ฟังก์ชัน Griewank

Function	Equation	Dimension	Range	Global minima	Properties
Griewank	$f(x) = \frac{1}{4000} \sum_{i=1}^A x_i^2 - \prod_{i=1}^A \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	50/100	$[-600,600]^D$	0	multimode

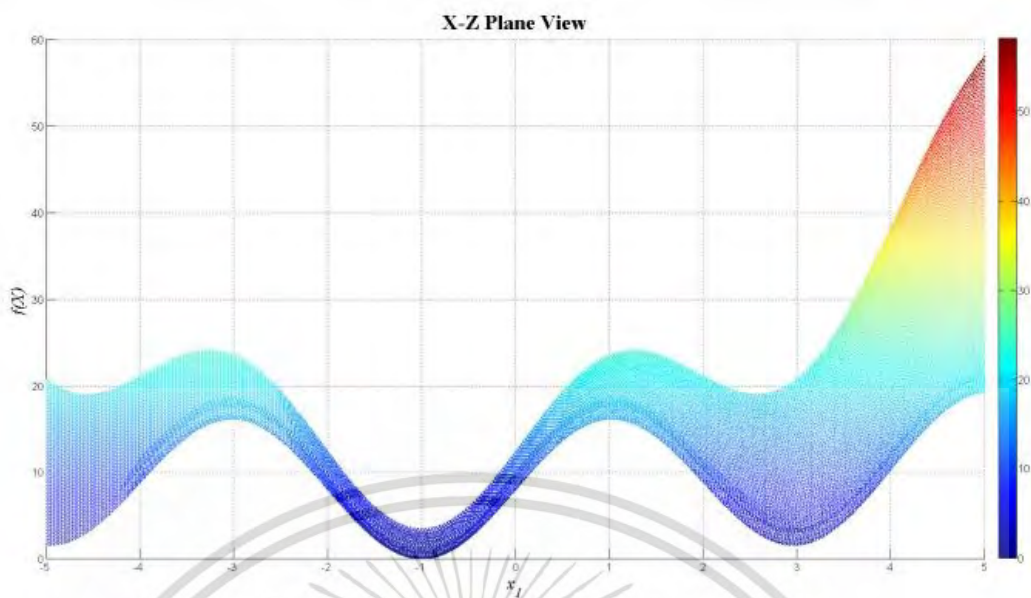
ฟังก์ชัน Griewank จัดอยู่ในประเภท Many Local Minima ซึ่งจะประกอบไปด้วยจุด Local Minima จำนวนหนึ่ง ลักษณะโดยรวมจะมีรูปร่างคล้ายกับซามแต่เมื่อพิจารณาในขอบเขตที่แคบลง จะพบจุด Local Minima ตามผิวของฟังก์ชัน รูปที่ 5.4 จะแสดงภาพ 3 มิติของฟังก์ชัน Griewank ในแต่ละขอบเขต แต่ในงานวิจัยนี้จะค้นหาในขอบเขต $[-600,600]$ เท่านั้น รายละเอียดการทดลองฟังก์ชัน Griewank ดังตารางที่ 5.3

5.3.4 ฟังก์ชัน Generalized Penalized



รูปที่ 5.5 ฟังก์ชัน Generalized Penalized แบบ 3 มิติ [10]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.6 ฟังก์ชัน Generalized Penalized แบบ 2 มิติ [10]

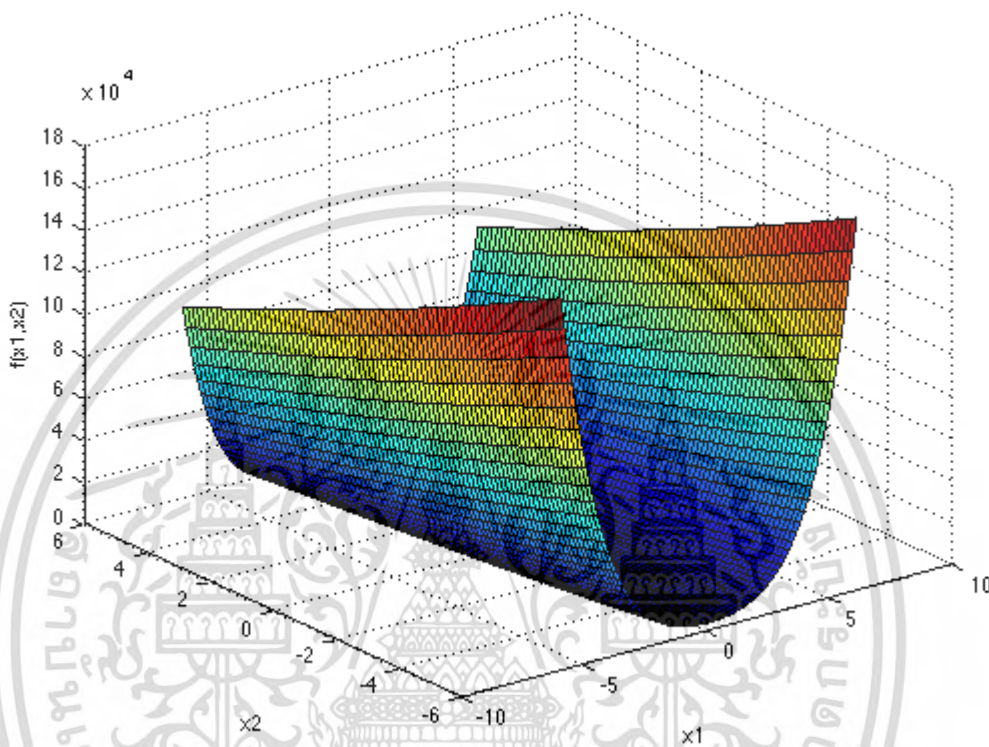
ตารางที่ 5.4 ฟังก์ชัน Generalized Penalized

Function	Equation	Dimension	Range	Global minima	Properties
Generalized Penalized	$f(x) = \frac{\pi}{a} \left\{ 10 \sin(\pi y_1)^2 + \sum_{i=1}^{d-1} (y_i - 1)^2 * [1 + 10 \sin(\pi y_{i+1})^2] + (y_d - 1)^2 \right\} + \sum_{i=1}^d u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{1}{4} (x_i + 1)$ $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	50/100	$[-50, 50]^p$	0	multi mode

ฟังก์ชัน Generalized Penalized จัดอยู่ในประเภท Many Local Minima ซึ่งจะประกอบไปด้วยจุด Local Minima จำนวนหนึ่ง ลักษณะโดยรวมจะคล้ายกับคลื่น รูปที่ 5.5 จะแสดงภาพ 3 มิติของเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน Generalized Penalized และ รูปที่ 5.6 จะแสดงภาพ 2 มิติของฟังก์ชัน Generalized Penalized รายละเอียดการทดลองฟังก์ชัน Generalized Penalized ดังตารางที่ 5.4

5.3.5 ฟังก์ชัน Rosenbrock



รูปที่ 5.7 ฟังก์ชัน Rosenbrock [9]

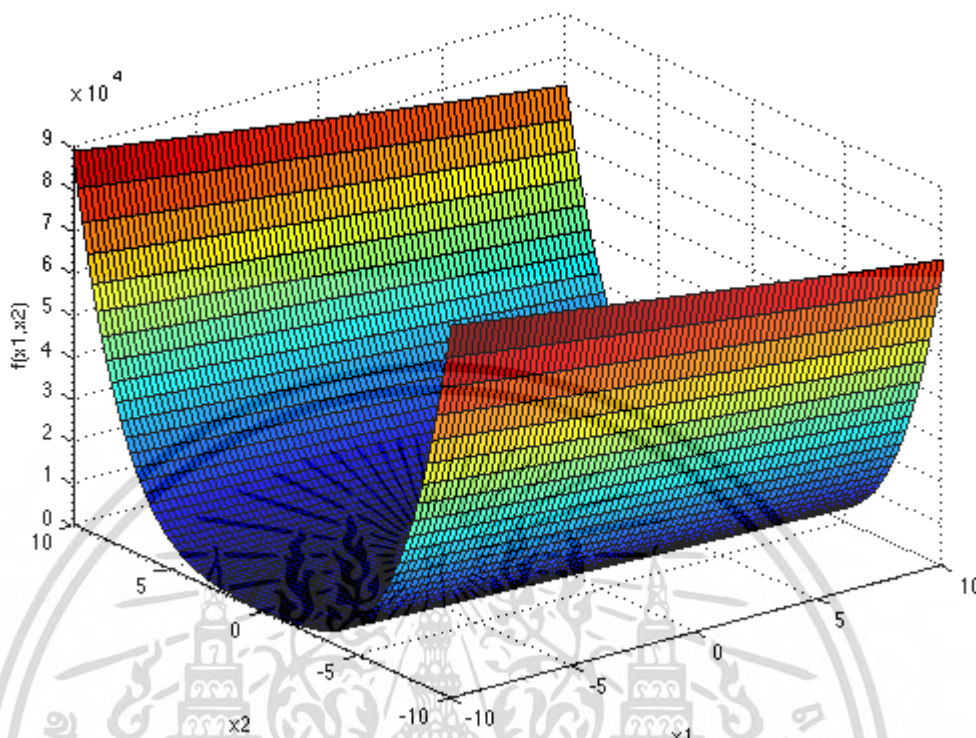
ตารางที่ 5.5 ฟังก์ชัน Rosenbrock

Function	Equation	Dimension	Range	Global minima	Properties
Rosenbrock	$f(x) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	50/100	$[-5, 10]^D$	0	multimode

ฟังก์ชัน Rosenbrock จัดอยู่ในประเภท Valley-Shaped มีลักษณะโดยรวมค่อนข้างเว้าเข้าหาจุด 0 ของแกน x1 รูปที่ 5.7 จะแสดงภาพ 3 มิติของฟังก์ชัน Rosenbrock รายละเอียดการทดลองฟังก์ชัน Rosenbrock ดังตารางที่ 5.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.6 ฟังก์ชัน Dixon-Price



รูปที่ 5.8 ฟังก์ชัน Dixon-Price [9]

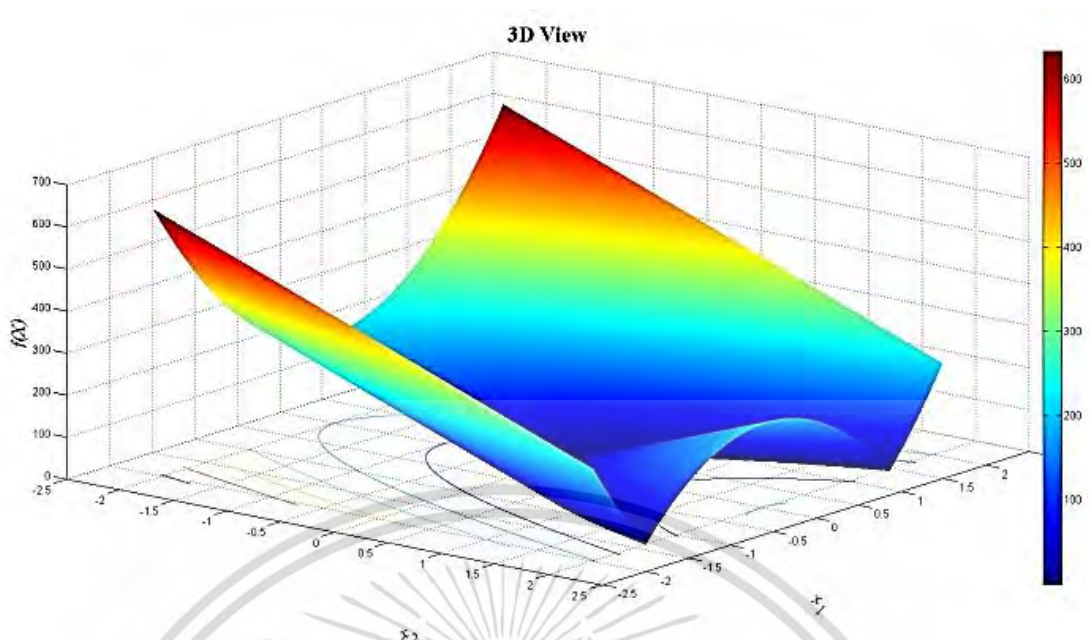
ตารางที่ 5.6 ฟังก์ชัน Dixon-Price

Function	Equation	Dimension	Range	Global minima	Properties
Dixon-Price	$f(x) = (x_1 - 1)^2 + \sum_{i=2}^d i(2x_i^2 - x_{i-1})^2$	50/100	$[-10, 10]^D$	0	multimode

ฟังก์ชัน Dixon-Price จัดอยู่ในประเภท Valley-Shaped มีลักษณะโดยรวมค่อนข้างเว้าเข้าหาจุด 0 ของแกน x_2 รูปที่ 5.8 จะแสดงภาพ 3 มิติของฟังก์ชัน Dixon-Price รายละเอียดการทดลองฟังก์ชัน Dixon-Price ดังตารางที่ 5.6

5.3.7 ฟังก์ชัน Generalized Modified Rosenbrock's Function No.01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



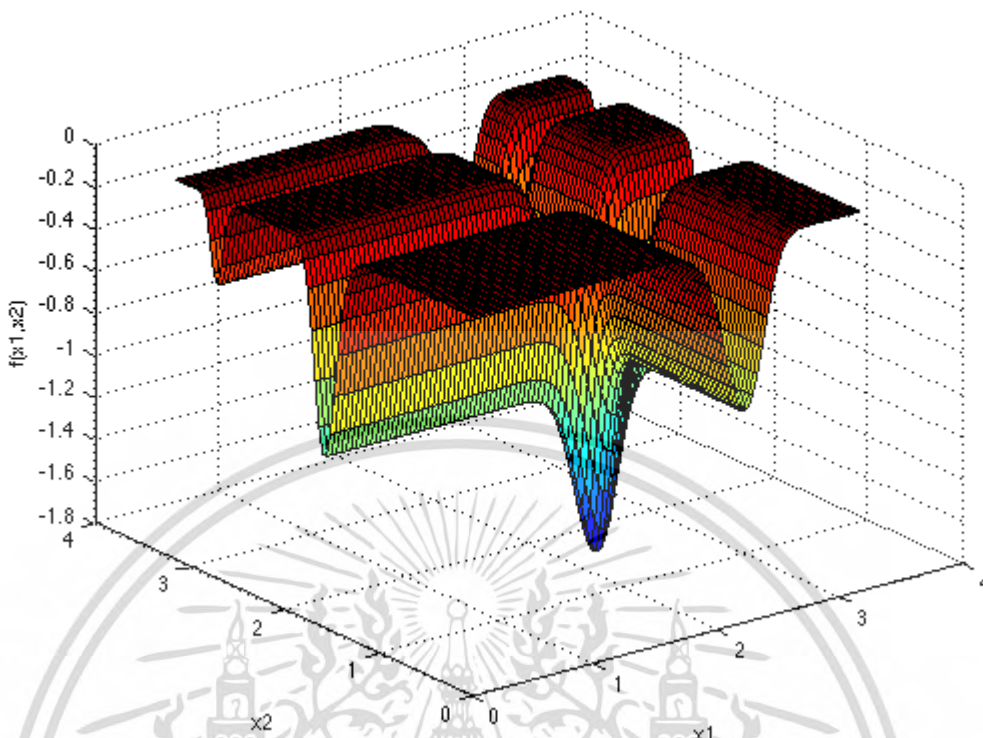
รูปที่ 5.9 ฟังก์ชัน Generalized Modified Rosenbrock's Function No.01 [10]

ตารางที่ 5.7 ฟังก์ชัน Generalized Modified Rosenbrock's Function No.01

Function	Equation	Dimension	Range	Global minima	Properties
Generalized Modified Rosenbrock's Function No.01	$f(x) = \sum_{i=1}^{n-1} [100 x_{i+1} - x_i^2 + (1 - x_i)^2]$	50/100	$[-2000, 2000]^D$	1	multi mode

ฟังก์ชัน Generalized Modified Rosenbrock's Function No.01 ถูกพัฒนามาจากฟังก์ชัน Rosenbrock โดยจะมีรูปร่างแตกต่างกันที่ช่วงล่าง รูปที่ 5.9 จะแสดงภาพ 3 มิติของฟังก์ชัน Generalized Modified Rosenbrock's Function No.01 โดยในฟังก์ชันนี้จะทดลองด้วยขอบเขตที่กว้างกว่าการทดลองของฟังก์ชัน Rosenbrock เพื่อหาจุดบกพร่องของอัลกอริทึม รายละเอียดการทดลองฟังก์ชัน Generalized Modified Rosenbrock's Function No.01 ดังตารางที่ 5.7

5.3.8 ฟังก์ชัน Michalewicz



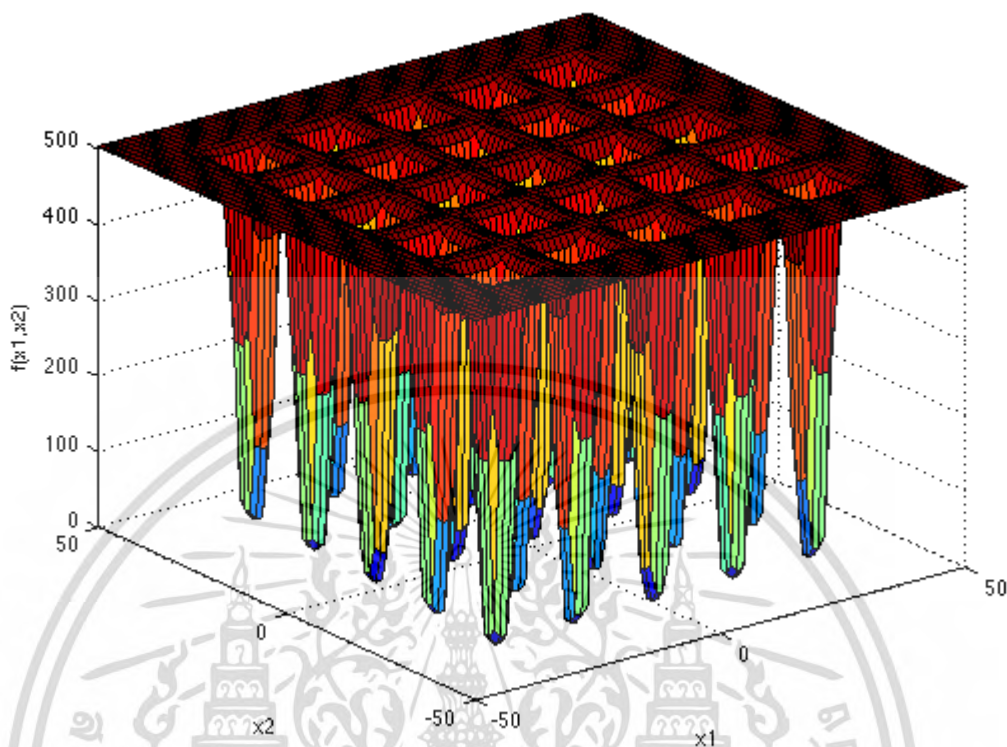
รูปที่ 5.10 ฟังก์ชัน Michalewicz [9]

ตารางที่ 5.8 ฟังก์ชัน Michalewicz

Function	Equation	Dimension	Range	Global minima	Properties
Michalewicz	$f(x) = - \sum_{i=1}^d \sin(x_i) \sin^{2m} \left(\frac{ix_i^2}{\pi} \right)$	50/100	$[0, \pi]^D$	Up on dimensions	multi mode

ฟังก์ชัน Michalewicz จัดอยู่ในประเภท Steep Ridges/Drops โดยมีลักษณะคล้ายกับหน้าผาและประกอบด้วยจุดต่ำสุดที่ระหว่างหน้าผา ฟังก์ชันนี้จะมีค่าต่ำสุดขึ้นอยู่กับจำนวนมิติ รูปที่ 5.10 จะแสดงภาพ 3 มิติของฟังก์ชัน Michalewicz รายละเอียดการทดลองฟังก์ชัน Michalewicz ดังตารางที่ 5.8

5.3.9 ฟังก์ชัน De Jong N. 5



รูปที่ 5.11 ฟังก์ชัน De Jong N. 5 [9]

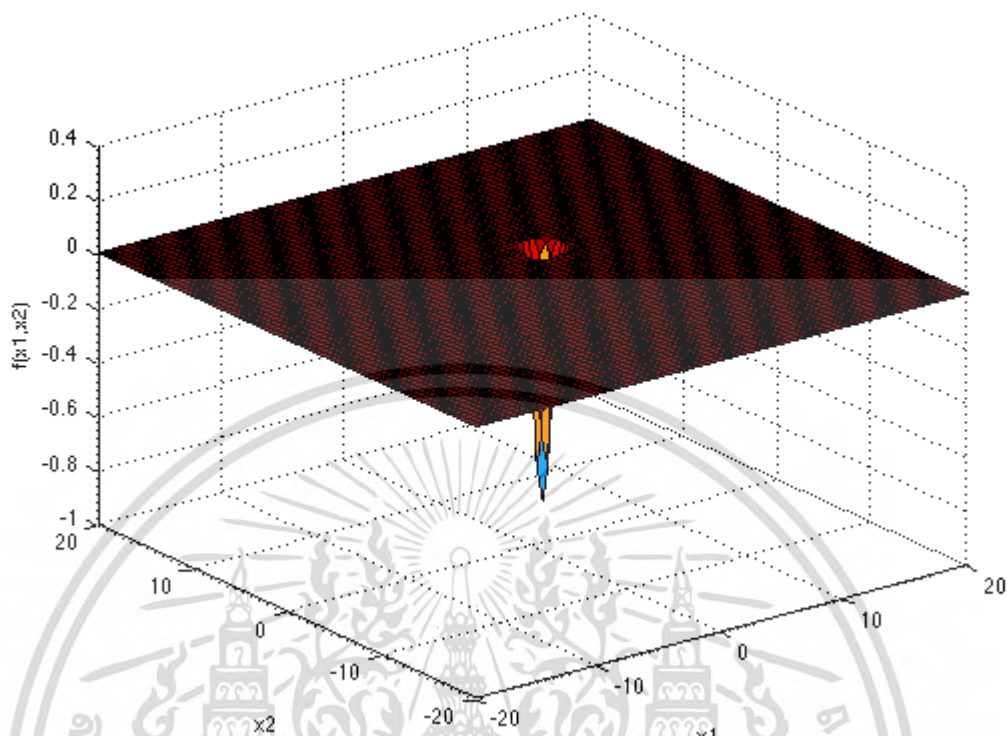
ตารางที่ 5.9 ฟังก์ชัน De Jong N. 5

Function	Equation	Dimension	Range	Global minima	Properties
De Jong N. 5	$f(x) = \left(0.002 + \sum_{i=1}^{25} \frac{1}{i + (x_1 - a_{1i})^6 + (x_2 - a_{2i})^6} \right)^{-1}$, where $a = \begin{pmatrix} -32 & -16 & 0 & 16 & 32 & -32 & \dots & 0 & 16 & 32 \\ -32 & -32 & -32 & -32 & -32 & -16 & \dots & 32 & 32 & 32 \end{pmatrix}$	2	$[-65.536, 65.536]^D$	0.998	multi mode

ฟังก์ชัน De Jong N. 5 จัดอยู่ในประเภท Steep Ridges/Drops โดยจะประกอบด้วยจุด Local Optima หลายจุด นอกจากนี้ฟังก์ชันนี้จะมียอดแค่ประกอบเพียง 2 มิติเท่านั้น รูปที่ 5.11 จะแสดงภาพ 3 มิติของฟังก์ชัน De Jong N. 5 รายละเอียดการทดลองฟังก์ชัน De Jong N. 5 ดังตารางที่ 5.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.10 ฟังก์ชัน Easom



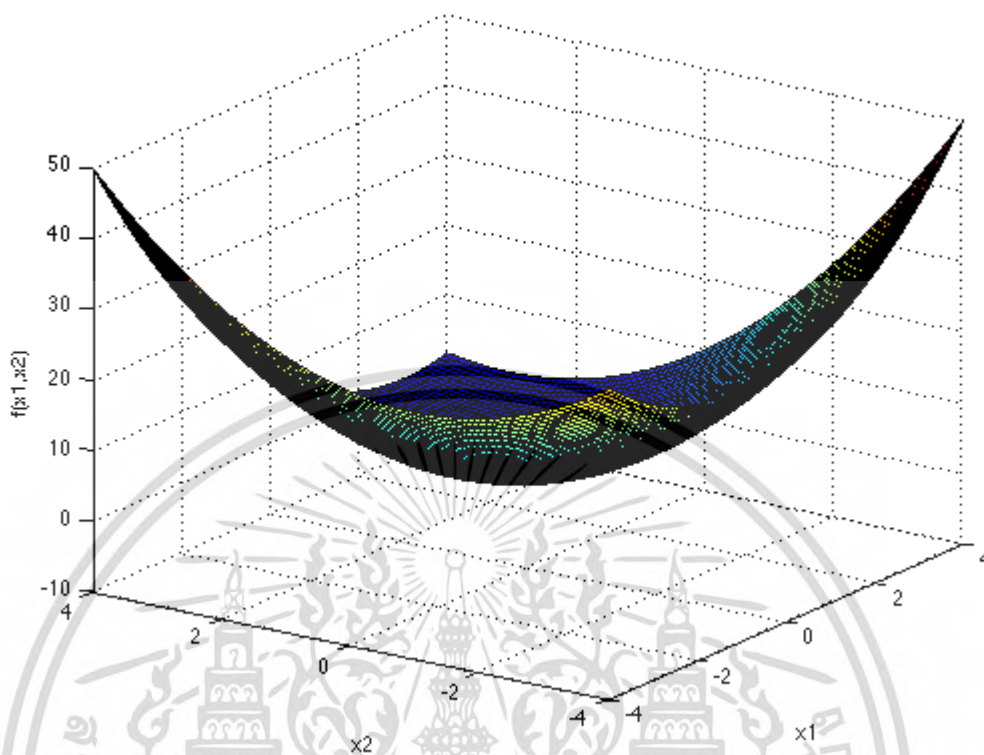
รูปที่ 5.12 ฟังก์ชัน Easom [9]

ตารางที่ 5.10 ฟังก์ชัน Easom

Function	Equation	Dimension	Range	Global minima	Properties
Easom	$f(x) = -\cos(x_1) \cos(x_2) \exp(-(x_1 - \pi)^2 - (x_2 - \pi)^2)$	2	$[-100, 100]^D$	-1	multi mode

ฟังก์ชัน Easom จัดอยู่ในประเภท Steep Ridges/Drops โดยลักษณะจะเป็นพื้นราบเกือบทั้งหมด ยกเว้นจุดต่ำสุดบริเวณตรงกลางจะไม่เป็นพื้นราบ นอกจากนี้ฟังก์ชันนี้จะมีองค์ประกอบเพียง 2 มิติ เท่านั้น รูปที่ 5.12 จะแสดงภาพ 3 มิติของฟังก์ชัน Easom รายละเอียดการทดลองฟังก์ชัน Easom ดังตารางที่ 5.10

5.3.11 ฟังก์ชัน Trid



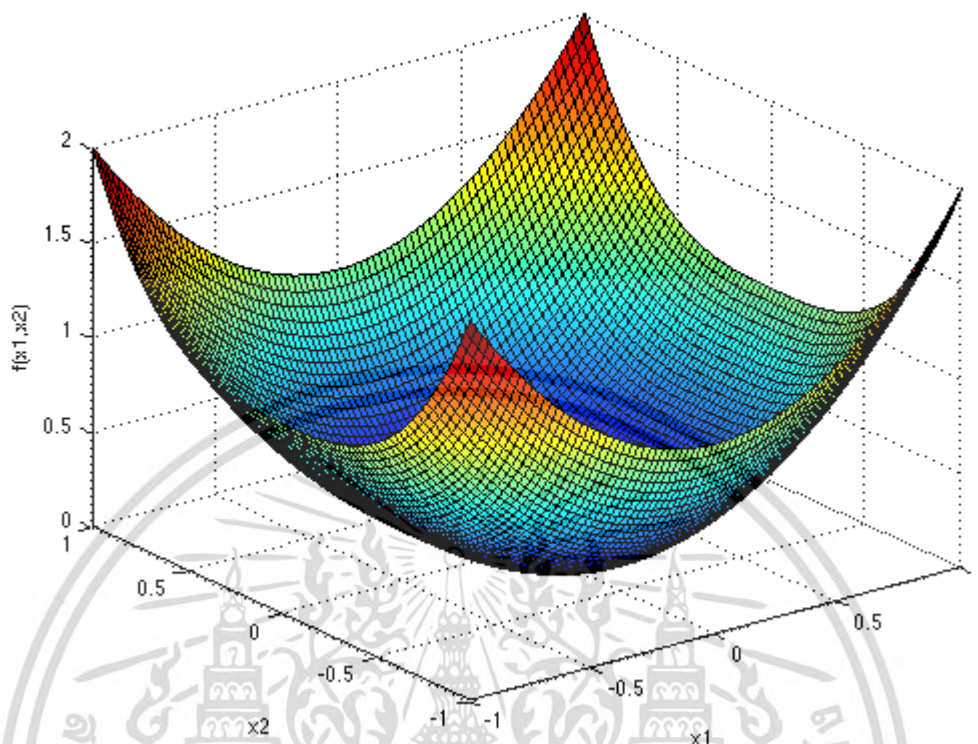
รูปที่ 5.13 ฟังก์ชัน Trid [9]

ตารางที่ 5.11 ฟังก์ชัน Trid

Function	Equation	Dimension	Range	Global minima	Properties
Trid	$f(x) = \sum_{i=1}^d (x_i - 1)^2 - \sum_{i=2}^d x_i x_{i-1}$	50/100	$[-D^2, D^2]^D$	$-D(D+4)(D-1)/6$	multi mode

ฟังก์ชัน Trid จัดอยู่ในประเภท Bowl-Shaped โดยลักษณะจะคล้ายกับถ้วยค่อนข้างเอียงเล็กน้อย ฟังก์ชันนี้จะมีขอบเขตและค่าต่ำสุดขึ้นอยู่กับจำนวนมิติ รูปที่ 5.13 จะแสดงภาพ 3 มิติของฟังก์ชัน Trid รายละเอียดการทดลองฟังก์ชัน Trid ดังตารางที่ 5.11

5.3.12 ฟังก์ชัน Sum of Different Powers



รูปที่ 5.14 ฟังก์ชัน Sum of Different Powers [9]

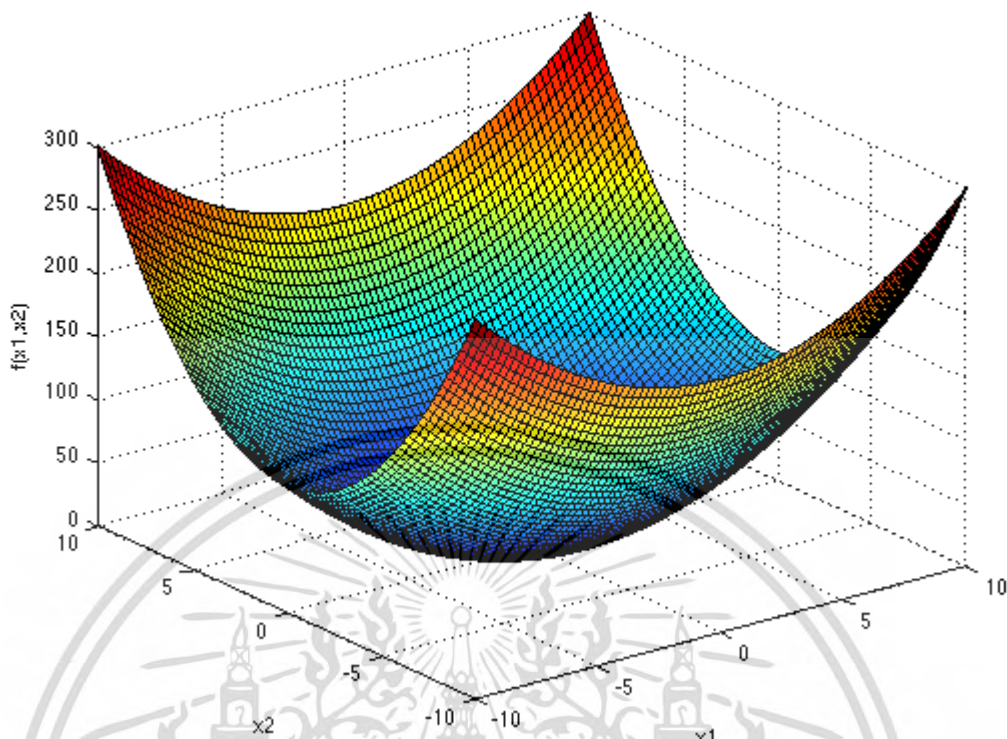
ตารางที่ 5.12 ฟังก์ชัน Sum of Different Powers

Function	Equation	Dimension	Range	Global minima	Properties
Sum of Different Powers	$f(x) = \sum_{i=1}^d x_i ^{i+1}$	50/100	$[-1,1]^D$	0	multi mode

ฟังก์ชัน Sum of Different Powers จัดอยู่ในประเภท Bowl-Shaped โดยลักษณะจะคล้ายกับถ้วยแบบตั้งตรง รูปที่ 5.14 จะแสดงภาพ 3 มิติของฟังก์ชัน Sum of Different Powers รายละเอียดการทดลองฟังก์ชัน Sum of Different Powers ดังตารางที่ 5.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.13 ฟังก์ชัน Sum Squares



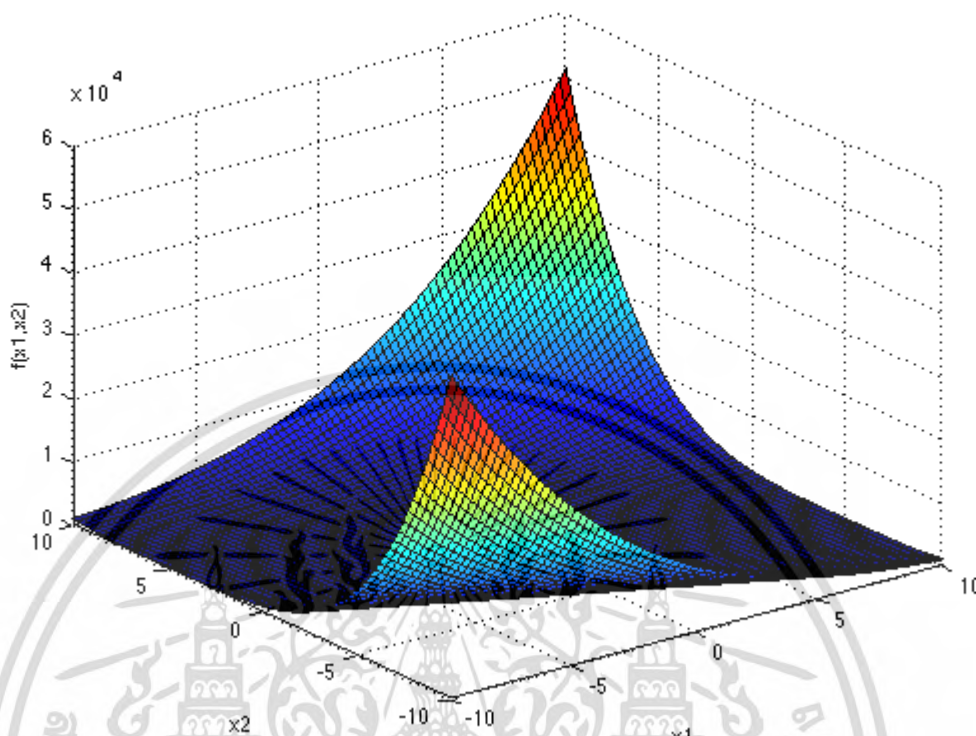
รูปที่ 5.15 ฟังก์ชัน Sum Squares [9]

ตารางที่ 5.13 ฟังก์ชัน Sum Squares

Function	Equation	Dimension	Range	Global minima	Properties
Sum Squares	$f(x) = \sum_{i=1}^d ix_i^2$	50/100	$[-10,10]^D$	0	multi mode

ฟังก์ชัน Sum Squares จัดอยู่ในประเภท Bowl-Shaped โดยลักษณะจะคล้ายกับถ้วยแบบตั้งตรง เลขชี้กำลังของสมการสามารถเป็นตัวบ่งบอกรูปร่างของฟังก์ชันได้ว่ามีความแบนเท่าใด ซึ่งในการทดลองเพิ่มผู้จัดทำได้ทดลองฟังก์ชันนี้ด้วยเลขยกกำลังที่แตกต่างกัน รูปที่ 5.15 จะแสดงภาพ 3 มิติของฟังก์ชัน Sum Squares รายละเอียดการทดลองฟังก์ชัน Sum Squares ดังตารางที่ 5.13

5.3.14 ฟังก์ชัน Zakharov



รูปที่ 5.16 ฟังก์ชัน Zakharov [9]

ตารางที่ 5.14 ฟังก์ชัน Zakharov

Function	Equation	Dimension	Range	Global minima	Properties
Zakharov	$f(x) = \sum_{i=1}^d x_i^2 + \left(\sum_{i=1}^d 0.5ix_i \right)^2 + \left(\sum_{i=1}^d 0.5ix_i \right)^4$	50/100	$[-5,10]^D$	0	multi mode

ฟังก์ชัน Zakharov จัดอยู่ในประเภท Plate-Shaped โดยลักษณะคล้ายกับแผ่นกระดาษ รูปที่ 5.16 จะแสดงภาพ 3 มิติของฟังก์ชัน Zakharov รายละเอียดการทดลองฟังก์ชัน Zakharov ดังตารางที่ 5.14

5.4 ผลการทดลอง

ผลการทดลองจะถูกจัดออกเป็น 5 กลุ่มด้วยกัน ตามประเภทของฟังก์ชัน ทั้งนี้เพื่อทดสอบ

อัลกอริทึมว่ามีผลต่อฟังก์ชันแต่ละประเภทอย่างไรบ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.15 ผลการทดลองของฟังก์ชันประเภท Many Local Minima ใน 50 มิติ

Function	Point	PSO	IAPSO	IAC-PSO
Noncontinuous Rastrigin	Best	32	21	0
	Median	120	64.05	0
	Worst	204	134	12
	Mean	112.64	65.42	0.64
Ackley	Best	1.47E-14	1.47E-14	2.22E-14
	Median	1.16E+00	2.18E-14	2.93E-14
	Worst	1.50E+01	4.31E-14	3.29E-14
	Mean	2.58E+00	2.38E-14	3.04E-14
Griewank	Best	0	0	0
	Median	0.0074	0.007396	0
	Worst	90.747	0.066072	0
	Mean	3.6298	0.008999	0
Generalized Penalized	Best	9.42E-33	9.42E-33	9.42E-33
	Median	6.22E-02	3.89E-32	9.42E-33
	Worst	1.75E+00	6.22E-02	9.42E-33
	Mean	2.12E-01	1.24E-03	9.42E-33

ตารางที่ 5.15 จะแสดงการเปรียบเทียบที่ 50 มิติ IAC-PSO สามารถค้นหาได้ดีกว่า Conventional PSO และ IAPSO เป็นส่วนใหญ่ แต่ยกเว้นในส่วนของ Ackley Function แต่ทั้งนี้ IAC-PSO กับ IAPSO สามารถค้นหาได้ใกล้เคียงกัน และ Generalized Penalized Function นั้น IAC-PSO ค้นหาค่า Best ได้น้อยกว่า Conventional PSO เพียงเล็กน้อยเท่านั้น

ตารางที่ 5.16 ผลการทดลองของฟังก์ชันประเภท Many Local Minima ใน 100 มิติ

Function	Point	PSO	IAPSO	IAC-PSO
Noncontinuous Rastrigin	Best	140	133	11
	Median	288	249.01	32
	Worst	603	455	48
	Mean	298.62	262.55	31.6
Ackley	Best	1.73E+00	5.10E-11	5.77E-14
	Median	1.30E+01	4.30E-10	6.84E-14
	Worst	1.82E+01	1.42E+00	7.90E-14
	Mean	1.12E+01	7.78E-02	6.85E-14
Griewank	Best	0	0	0
	Median	90.9803	0	0
	Worst	360.9793	0.0706	0
	Mean	126.47	0.0091	0
Generalized Penalized	Best	4.71E-33	4.89E-22	4.71E-33
	Median	3.11E-02	1.32E-19	4.71E-33
	Worst	7.89E-01	9.33E-02	4.71E-33
	Mean	1.28E-01	1.06E-02	4.71E-33

ตารางที่ 5.16 จะแสดงการเปรียบเทียบที่ 100 มิติ ซึ่งได้พิสูจน์ประสิทธิภาพของอัลกอริทึมว่าสามารถค้นหาในพื้นที่ (Search Space) ขนาดใหญ่ได้ ผลการทดลองแสดงให้เห็นว่า IAC-PSO ไม่สามารถค้นหาจุด Global Optima ได้ในหลายฟังก์ชัน แต่สามารถได้คำตอบที่ดีที่สุดในทุกค่าสถิติจากทุกฟังก์ชัน

ตารางที่ 5.17 ผลการทดลองของฟังก์ชันประเภท Valley-Shaped ใน 50 มิติ

Function	Point	PSO	IAPSO	IAC-PSO
Rosenbrock	Best	2.45E+01	0.0606	4.91E-04
	Median	1.03E+02	27.3077	7.9362
	Worst	2.25E+02	83.9991	74.3372
	Mean	1.16E+02	32.4621	10.5793

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการวิจัยเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ในการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.17 (ต่อ)

Function	Point	PSO	IAPSO	IAC-PSO
Dixon-Price	Best	6.67E-01	0.6667	0.6667
	Median	3.68E+00	0.667	0.6667
	Worst	1.35E+01	0.6791	0.6667
	Mean	3.85E+00	0.668	0.6667
Generalized Modified Rosenbrock's Function No.01	Best	1.31E+03	290.3119	137.5576
	Median	5.47E+03	2.56E+03	623.3544
	Worst	1.69E+05	1.08E+04	2.12E+03
	Mean	1.51E+04	3.27E+03	810.0134

ตารางที่ 5.17 จะแสดงการเปรียบเทียบที่ฟังก์ชันประเภท Valley-Shaped ที่ 50 มิติ IAC-PSO สามารถค้นหาได้ดีกว่า Conventional PSO และ IAPSO เป็นส่วนใหญ่ แต่ยกเว้นในส่วนของ Dixon-Price Function เนื่องจาก IAC-PSO, IAPSO และ Conventional PSO สามารถค้นหาค่า Best ได้เท่ากับค่า Global Optima

ตารางที่ 5.18 ผลการทดลองของฟังก์ชันประเภท Valley-Shaped ใน 100 มิติ

Function	Point	PSO	IAPSO	IAC-PSO
Rosenbrock	Best	7.00E+02	2.66E+01	2.27E-08
	Median	1.21E+03	151.6514	0.4483
	Worst	2.86E+03	1.01E+03	98.195
	Mean	1.38E+03	190.8011	23.4284
Dixon-Price	Best	1.35E+02	0.6667	0.6667
	Median	4.87E+02	0.6678	0.6667
	Worst	5675.3	34.2394	0.6667
	Mean	7.85E+02	4.83E+00	0.6667
Generalized Modified Rosenbrock's Function No.01	Best	2.43E+06	681.802	465.4799
	Median	2.14E+07	3.54E+03	693.7022
	Worst	1.02E+08	1.74E+04	1.09E+03
	Mean	2.71E+07	4.89E+03	696.6161

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.18 จะแสดงการเปรียบเทียบที่ฟังก์ชันประเภท Valley-Shaped ที่ 100 มิติ เมื่อเพิ่มจำนวนมิติ IAC-PSO สามารถค้นหาได้ดีกว่า Conventional PSO และ IAPSO ในทุกฟังก์ชัน

ตารางที่ 5.19 ผลการทดลองของฟังก์ชันประเภท Steep Ridges/Drops ใน 50 มิติ และ 2 มิติ

Function	Point	PSO	IAPSO	IAC-PSO
Michalewicz (50 dimension)	Best	-44.6969	-41.9221	-44.0777
	Median	-37.7173	-37.7346	-41.6676
	Worst	-32.4989	-22.4904	-38.783
	Mean	-37.9831	-36.1281	-41.5893
De Jong N. 5 (2 dimension)	Best	0.998	0.998	0.998
	Median	0.998	0.998	0.998
	Worst	0.998	0.998	0.998
	Mean	0.998	0.998	0.998
Easom (2 dimension)	Best	-1.00E+00	-1.00E+00	-1.00E+00
	Median	-1.00E+00	-9.95E-01	-1.00E+00
	Worst	-1.00E+00	-8.92E-01	-1.00E+00
	Mean	-1.00E+00	-9.90E-01	-1.00E+00

ตารางที่ 5.19 จะแสดงการเปรียบเทียบที่ Michalewicz Function นั้น IAC-PSO สามารถค้นหาได้ดีกว่า Conventional PSO และ IAPSO เป็นส่วนใหญ่ แต่ยกเว้นในส่วนของค่า Best นั้น IAC-PSO สามารถค้นหาค่าได้แย่กว่า Conventional PSO เล็กน้อย แต่เมื่อจำนวนมิติเพิ่มมากขึ้น IAC-PSO สามารถค้นหาได้ดีกว่าในทุกๆค่า

สำหรับ 2 ฟังก์ชันสุดท้ายทั้ง 3 อัลกอริทึมสามารถค้นหาจุด Global Optima ได้ยกเว้นเพียงแต่อัลกอริทึม IAPSO ที่ Easom Function สามารถค้นหาได้ค่าที่ใกล้เคียงกับ Global Optima เท่านั้น

ตารางที่ 5.20 ผลการทดลองของฟังก์ชันประเภท Steep Ridges/Drops ใน 100 มิติ

Function	Point	PSO	IAPSO	IAC-PSO
Michalewicz	Best	-76.2456	-74.0175	-87.63
	Median	-68.3292	-36.8022	-81.9545
	Worst	-61.0464	-23.1208	-76.4755
	Mean	-68.1044	-38.1932	-82.1025

ตารางที่ 5.20 จะแสดงการเปรียบเทียบที่ Michalewicz Function เมื่อเพิ่มจำนวนมิติ IAC-PSO สามารถค้นหาได้ดีกว่าในทุกๆค่าสถิติ จากเดิมที่ค้นหาค่า Best ได้แยกแยะเพียงเล็กน้อยเท่านั้น

ตารางที่ 5.21 ผลการทดลองของฟังก์ชันประเภท Bowl-Shaped ใน 50 มิติ

Function	Point	PSO	IAPSO	IAC-PSO
Trid	Best	-2.63E+03	-2.18E+04	-2.20E+04
	Median	7.76E+04	-1.59E+04	-1.96E+04
	Worst	2.45E+05	1.34E+04	4.68E+03
	Mean	9.05E+04	-1.25E+04	-1.82E+04
Sum of Different Powers	Best	1.29E-248	4.83E-25	3.36E-40
	Median	1.66E-104	1.39E-23	1.44E-35
	Worst	2.88E-41	4.22E-22	1.80E-33
	Mean	5.77E-43	3.78E-23	1.40E-34
Sum Squares	Best	1.47E-08	5.57E-09	9.90E-44
	Median	1.19E-05	1.39E-04	4.03E-43
	Worst	0.0145	0.0034	1.60E-42
	Mean	0.0011	4.16E-04	4.49E-43

ตารางที่ 5.21 จะแสดงการเปรียบเทียบที่ฟังก์ชันประเภท Bowl-Shaped ที่ 50 มิติ IAC-PSO สามารถค้นหาได้ดีกว่า Conventional PSO และ IAPSO เป็นส่วนใหญ่ในทุกๆค่า แต่ยกเว้นในส่วน ของ Sum of Different Powers Function โดย Conventional PSO สามารถค้นหาค่าได้ดีกว่าในทุกๆค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.22 ผลการทดลองของฟังก์ชันประเภท Bowl-Shaped ใน 100 มิติ

Function	Point	PSO	IAPSO	IAC-PSO
Trid	Best	6.16E+06	-9.31E+04	-1.58E+05
	Median	1.76E+07	2.05E+05	2.62E+03
	Worst	4.01E+07	5.18E+05	5.18E+05
	Mean	1.83E+07	2.52E+06	4.06E+04
Sum of Different Powers	Best	8.56E-195	8.93E-25	2.83E-35
	Median	2.24E-43	1.21E-21	2.84E-33
	Worst	3.45E-26	4.79E-20	3.65E-29
	Mean	6.92E-28	4.03E-21	1.13E-30
Sum Squares	Best	19.3265	2.12E-09	7.44E-41
	Median	89.0816	9.13E-05	1.40E-40
	Worst	408.1147	3.20E-03	3.41E-40
	Mean	125.3078	4.28E-04	1.56E-40

ตารางที่ 5.22 จะแสดงการเปรียบเทียบที่ฟังก์ชันประเภท Bowl-Shaped ที่ 100 มิติ เมื่อจำนวนมิติเพิ่มขึ้นเป็น 100 มิติ IAC-PSO สามารถค้นหาค่าได้ดีกว่าในค่า Worst และ Mean

ตารางที่ 5.23 ผลการทดลองของฟังก์ชันประเภท Plate-Shaped

Function	Point	PSO	IAPSO	IAC-PSO
Zakharov (50 dimension)	Best	4.52E-234	9.83E+00	1.45E-17
	Median	6.74E-222	1.80E+01	6.38E-16
	Worst	1.66E-209	2.75E+01	3.37E-15
	Mean	3.51E-211	1.83E+01	7.52E-16
Zakharov (100 dimension)	Best	1.19E-46	7.09E+01	3.86E-10
	Median	1.02E-37	8.64E+01	2.69E-08
	Worst	3.65E-29	1.08E+02	1.11E-07
	Mean	7.30E-30	8.66E+01	3.03E-08

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.23 จะแสดงการเปรียบเทียบที่ Zakharov Function นั้น Conventional PSO สามารถค้นหาค่าได้ดีกว่าในทุกๆค่า แต่ถึงกระนั้น IAC-PSO ยังคงมีเสถียรภาพมากกว่า Conventional PSO หมายความว่าหากจำนวนมิติเพิ่มมากขึ้นอีกจำนวนหนึ่ง IAC-PSO ก็จะสามารถชนะ Conventional PSO ได้

จากปัญหาข้างต้นที่ Conventional PSO สามารถค้นหาค่าได้ดีกว่าในบางฟังก์ชันนั้น ทางผู้จัดทำจะขออธิบายด้วยการทดลองเพิ่มเติมในบทที่ 5.5

ตารางที่ 5.24 สรุปผลการทดลองที่ 2 และ 50 มิติ

Function	Point	PSO	IAPSO	IAC-PSO
Noncontinuous Rastrigin (50 dimension)	Best	32	21	0
	Median	120	64.05	0
Ackley (50 dimension)	Best	1.47E-14	1.47E-14	2.22E-14
	Median	1.16E+00	2.18E-14	2.93E-14
Griewank (50 dimension)	Best	0	0	0
	Median	0.0074	0.007396	0
Generalized Penalized (50 dimension)	Best	9.42E-33	9.42E-33	9.42E-33
	Median	6.22E-02	3.89E-32	9.42E-33
Rosenbrock (50 dimension)	Best	2.45E+01	0.0606	4.91E-04
	Median	1.03E+02	27.3077	7.9362
Dixon-Price (50 dimension)	Best	6.67E-01	0.6667	0.6667
	Median	3.68E+00	0.667	0.6667
Generalized Modified Rosenbrock's Function No.01 (50 dimension)	Best	1.31E+03	290.3119	137.5576
	Median	5.47E+03	2.56E+03	623.3544
Michalewicz (50 dimension)	Best	-44.6969	-41.9221	-44.0777
	Median	-37.7173	-37.7346	-41.6676

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.24 (ต่อ)

Function	Point	PSO	IAPSO	IAC-PSO
De Jong N. 5 (2 dimension)	Best	0.998	0.998	0.998
	Median	0.998	0.998	0.998
Easom (2 dimension)	Best	-1.00E+00	-1.00E+00	-1.00E+00
	Median	-1.00E+00	-9.95E-01	-1.00E+00
Trid (50 dimension)	Best	-2.63E+03	-2.18E+04	-2.20E+04
	Median	7.76E+04	-1.59E+04	-1.96E+04
Sum of Different Powers (50 dimension)	Best	1.29E-248	4.83E-25	3.36E-40
	Median	1.66E-104	1.39E-23	1.44E-35
Sum Squares (50 dimension)	Best	1.47E-08	5.57E-09	9.90E-44
	Median	1.19E-05	1.39E-04	4.03E-43
Zakharov (50 dimension)	Best	4.52E-234	9.83E+00	1.45E-17
	Median	6.74E-222	1.80E+01	6.38E-16

จากตารางที่ 5.24 จะสรุปผลการทดลองทั้งหมดจากทุกๆ ฟังก์ชันใน 2 และ 50 มิติจากทุกกลุ่ม โดยจะเปรียบเทียบเฉพาะค่า Best และ Median เท่านั้น ผลการเปรียบเทียบแสดงให้เห็นว่า IAC-PSO สามารถค้นหาค่าได้ดีกว่าอัลกอริทึมอื่นๆ ได้ใน 10 ฟังก์ชันจากทั้งหมด 14 ฟังก์ชัน ซึ่งคิดเป็นอัตราส่วนคือ 71.43% โดยฟังก์ชัน Ackley และ Michalewicz นั้น IAC-PSO สามารถค้นหาค่าได้เร็วกว่าเพียงเล็กน้อยเท่านั้น

ตารางที่ 5.25 สรุปผลการทดลองที่ 100 มิติ

Function	Point	PSO	IAPSO	IAC-PSO
Noncontinuous Rastrigin (100 dimension)	Best	140	133	11
	Median	288	249.01	32
Ackley (100 dimension)	Best	1.73E+00	5.10E-11	5.77E-14
	Median	1.30E+01	4.30E-10	6.84E-14

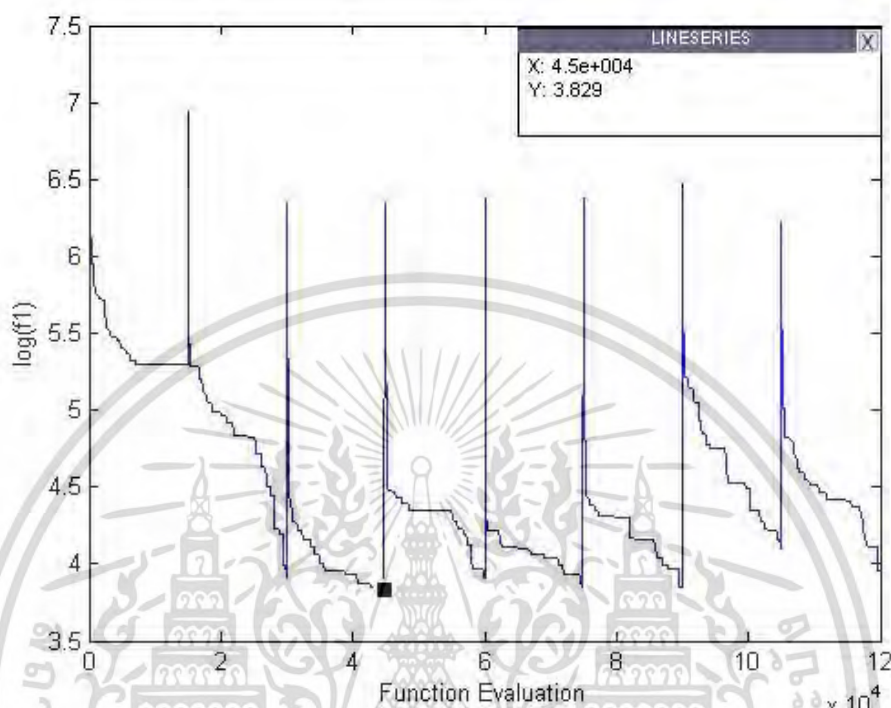
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.25 (ต่อ)

Function	Point	PSO	IAPSO	IAC-PSO
Griewank (100 dimension)	Best	0	0	0
	Median	90.9803	0	0
Generalized Penalized (100 dimension)	Best	4.71E-33	4.89E-22	4.71E-33
	Median	3.11E-02	1.32E-19	4.71E-33
Rosenbrock (100 dimension)	Best	7.00E+02	2.66E+01	2.27E-08
	Median	1.21E+03	151.6514	0.4483
Dixon-Price (100 dimension)	Best	1.35E+02	0.6667	0.6667
	Median	4.87E+02	0.6678	0.6667
Generalized Modified Rosenbrock's Function No.01 (100 dimension)	Best	2.43E+06	681.802	465.4799
	Median	2.14E+07	3.54E+03	693.7022
Michalewicz (100 dimension)	Best	-76.2456	-74.0175	-87.63
	Median	-68.3292	-36.8022	-81.9545
Trid (100 dimension)	Best	6.16E+06	-9.31E+04	-1.58E+05
	Median	1.76E+07	2.05E+05	2.62E+03
Sum of Different Powers (100 dimension)	Best	8.56E-195	8.93E-25	2.83E-35
	Median	2.24E-43	1.21E-21	2.84E-33
Sum Squares (100 dimension)	Best	19.3265	2.12E-09	7.44E-41
	Median	89.0816	9.13E-05	1.40E-40
Zakharov (100 dimension)	Best	1.19E-46	7.09E+01	3.86E-10
	Median	1.02E-37	8.64E+01	2.69E-08

ตารางที่ 5.25 จะสรุปผลการทดลองที่ 100 มิติจากทุกกลุ่ม ผลการเปรียบเทียบแสดงให้เห็นว่า IAC-PSO สามารถค้นหาค่าได้ดีกว่าอัลกอริทึมอื่นๆ ได้ใน 10 ฟังก์ชันจากทั้งหมด 12 ฟังก์ชัน ซึ่งคิด เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นอัตราส่วนคือ 83.33% ซึ่งแสดงให้เห็นว่า IAC-PSO จะค้นหาได้ดีกว่าอัลกอริทึมอื่นๆเมื่อจำนวนมิติเพิ่มมากขึ้น



รูปที่ 5.17 กราฟการลู่เข้าของ IAC-PSO ของฟังก์ชัน Noncontinuous Rastrigin ที่ 100 มิติ

นอกจากนี้เราได้นำเสนอกราฟการลู่เข้า (Convergence Graph) ที่แสดงความสัมพันธ์ของค่า Fitness และ Function Evaluation Times ในรูปที่ 5.17 จุดสีดำคือค่าที่ดีที่สุด (ตำแหน่งของจุดสีดำจะแสดงในช่องสี่เหลี่ยมขวามือ) ของ Generation ที่ค้นหาได้ดีที่สุด เส้นตรงตามแนวแกน Y แต่ละแท่งในรูปที่ 5.17 คือการเริ่มต้นใหม่ของ Generation โดย IAC-PSO จะทำทั้งหมด 7 Generation ซึ่งอัลกอริทึมสามารถค้นหาค่าที่เหมาะสมได้ก่อนถึง Generation ในช่วงท้าย ทั้งนี้เราจะลดจำนวน Generation หากไม่มีผลกระทบต่อคำตอบสุดท้ายของอัลกอริทึม

5.5 การค้นหาจุดบกพร่องของอัลกอริทึม

ในส่วนนี้จะเป็นการทดลองเพื่อหารูปแบบฟังก์ชันที่ IAC-PSO ค้นหาได้ไม่ดีเท่ากับ PSO เนื่องจากในการทดลองนั้นมีบางฟังก์ชันที่ IAC-PSO ค้นหาได้ไม่ดีเท่ากับ PSO โดยการทดลองเพิ่มเติมนี้จะทดลองด้วยฟังก์ชันที่มีรูปร่างเหมือนกันแต่แตกต่างกันที่เลขยกกำลังดังตารางที่ 5.26 โดยค่าพารามิเตอร์และอัลกอริทึมที่ใช้ทั้งหมดจะเหมือนกับการทดลองหลัก

ก. ฟังก์ชัน

ตารางที่ 5.26 ฟังก์ชันที่ใช้ในการทดลองเพิ่มเติม

Function	Equation	Dimension	Range	Global minima	Properties
Sum Squares	$f_1(x) = \sum_{i=1}^D [ix_i^2]$	50	$[-1,1]^D$	0	multimode
Quartic	$f_2(x) = \sum_{i=1}^D [ix_i^4]$	50	$[-1,1]^D$	0	multimode
Sextic	$f_3(x) = \sum_{i=1}^D [ix_i^6]$	50	$[-1,1]^D$	0	multimode

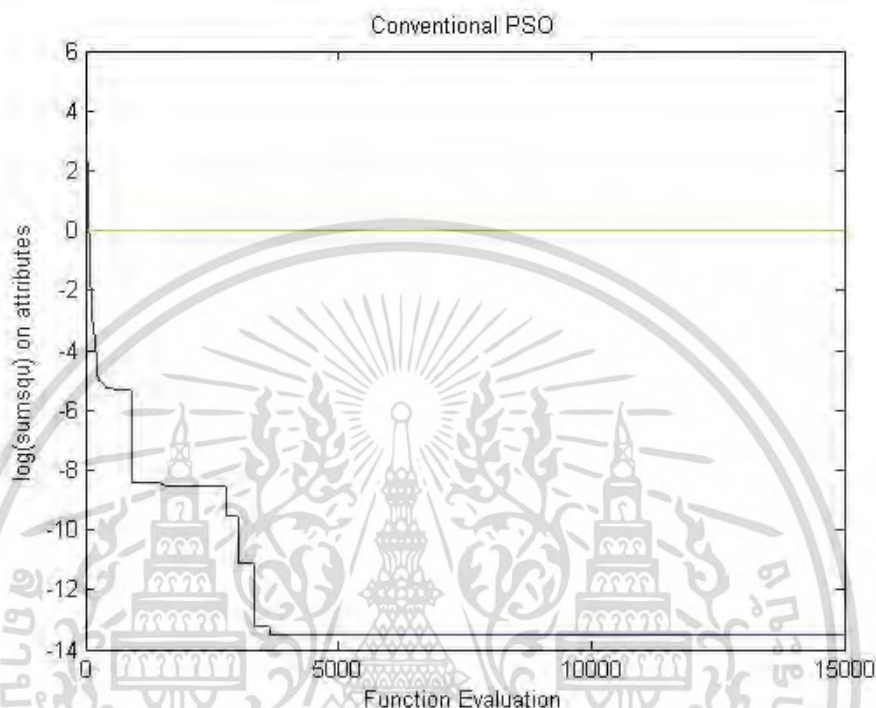
ข. ผลการทดลอง

ตารางที่ 5.27 ผลการทดลองเพิ่มเติม

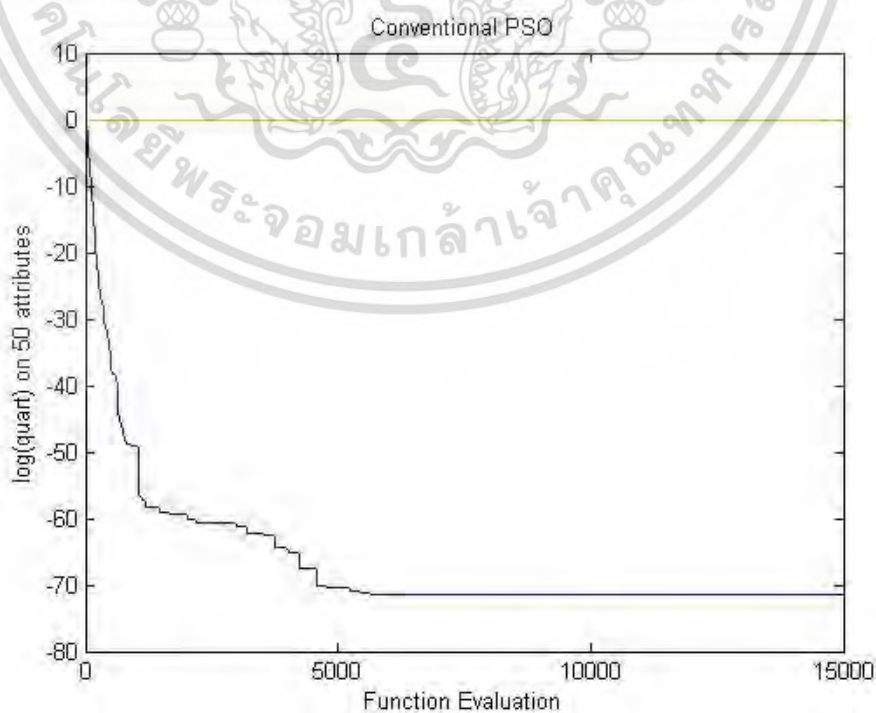
Function	Point	PSO	IAPSO	IAC-PSO
Sum Squares (50 dimension)	Best	3.96E-09	2.48E-08	9.50E-46
	Median	1.50E-07	9.25E-07	1.67E-45
	Worst	1.15E-05	2.15E-05	2.66E-45
	Mean	2.58E-06	5.58E-06	1.61E-45
Quartic (50 dimension)	Best	1.49E-39	3.72E-18	7.90E-77
	Median	6.98E-20	1.52E-13	3.79E-75
	Worst	3.32E-19	3.23E-10	5.70E-75
	Mean	9.61E-20	6.64E-11	3.04E-75
Sextic (50 dimension)	Best	0.00E+00	1.16E-22	8.15E-93
	Median	3.00E-188	7.66E-20	1.65E-91
	Worst	9.78E-86	4.76E-15	4.91E-90
	Mean	1.96E-86	9.51E-16	1.44E-90

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.27 จะแสดงผลการทดลอง ซึ่งแสดงให้เห็นว่า IAC-PSO สามารถเอาชนะ Conventional PSO ได้ใน 2 ฟังก์ชันแรกแต่ในฟังก์ชันที่ 3 นั้น Conventional PSO ค้นหาได้ดีกว่าในค่า Best และ Median ซึ่งสามารถสรุปได้ว่า Conventional PSO จะค้นหาได้ดีขึ้นหากเลขยกกำลังเพิ่มมากขึ้น

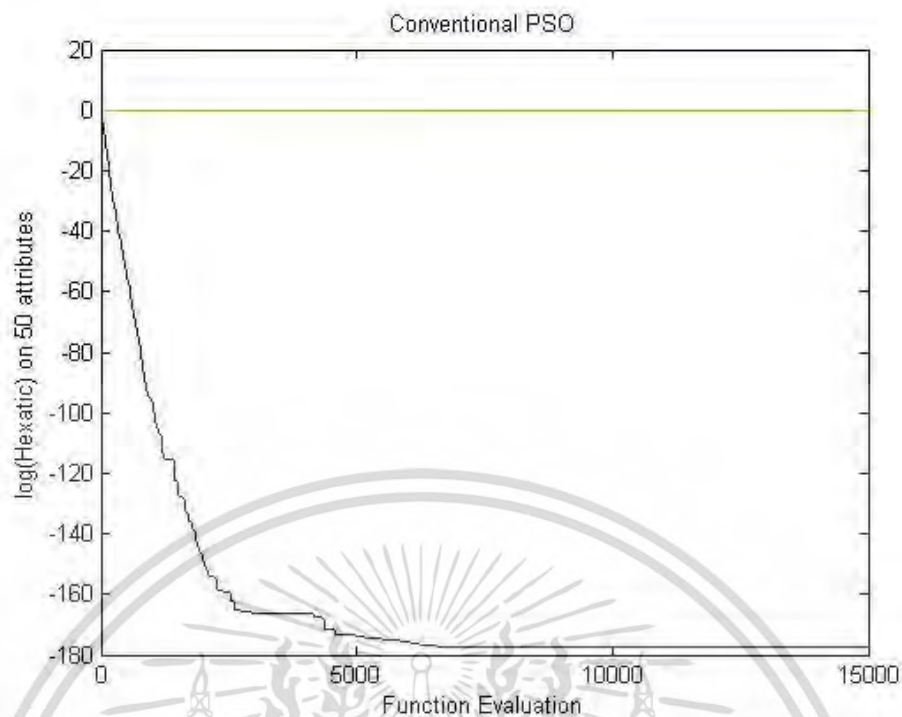


รูปที่ 5.18 กราฟการลู่เข้าของ Conventional PSO กับฟังก์ชัน Sum Squares ใน 50 มิติ

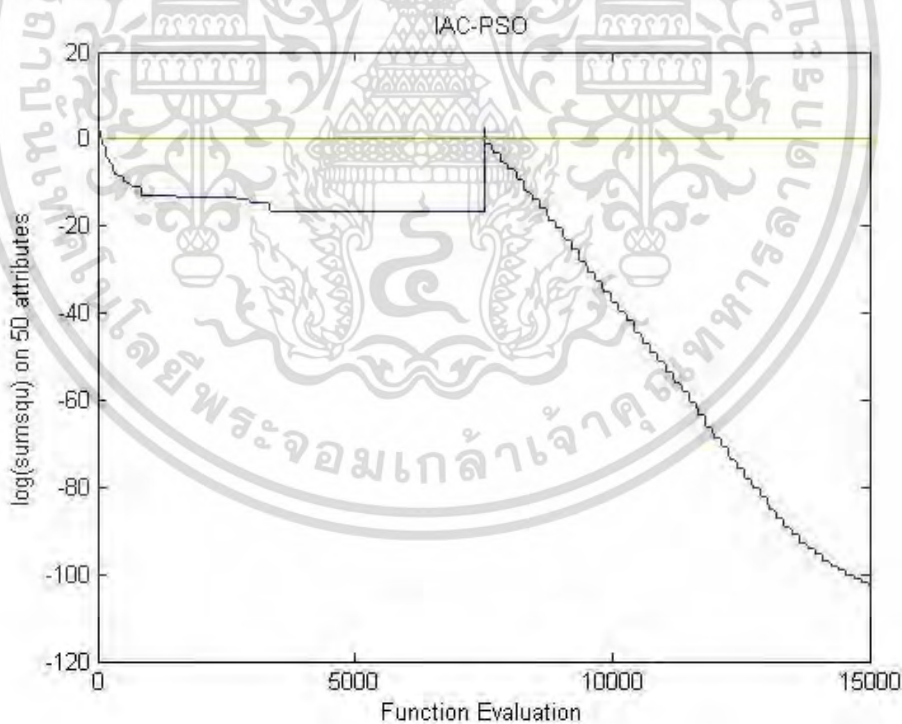


รูปที่ 5.19 กราฟการลู่เข้าของ Conventional PSO กับฟังก์ชัน Quartic ใน 50 มิติ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อเผยแพร่เห็นใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

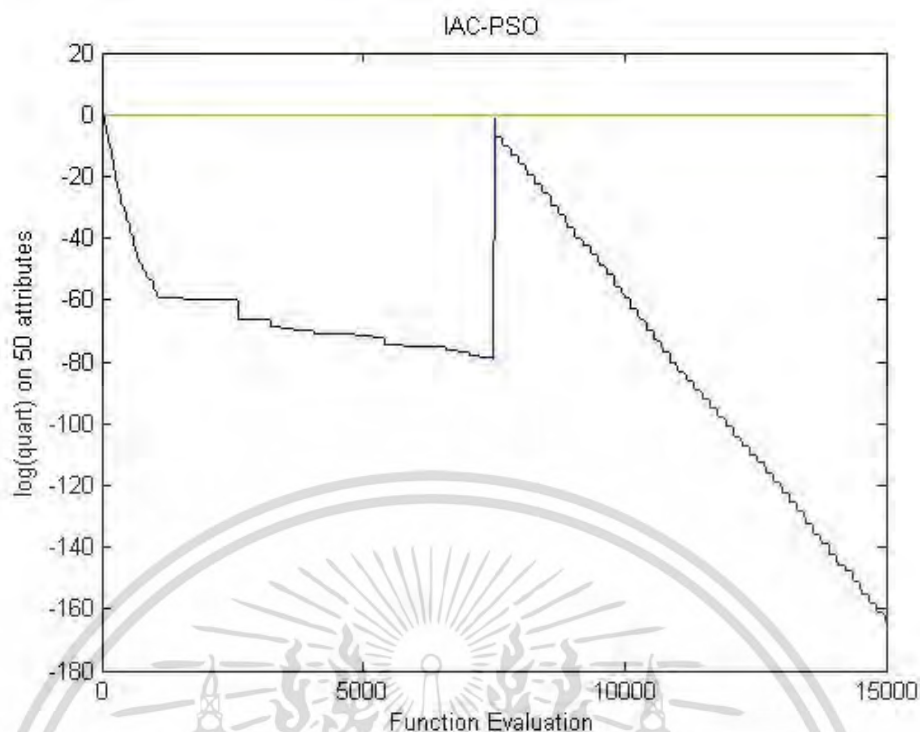


รูปที่ 5.20 กราฟการลู่เข้าของ Conventional PSO กับฟังก์ชัน Sextic ใน 50 มิติ

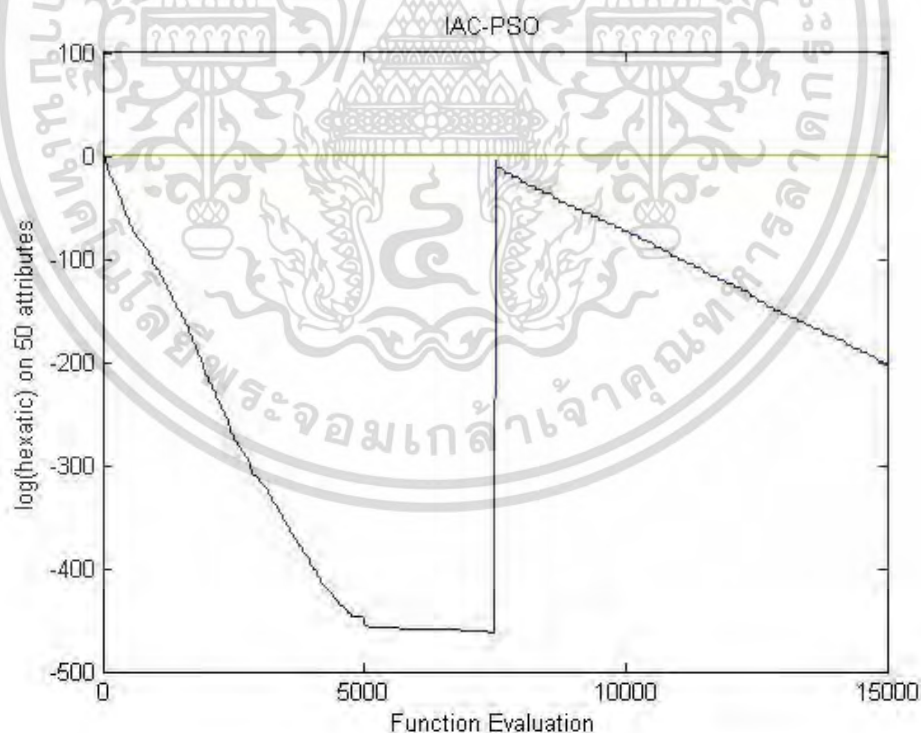


รูปที่ 5.21 กราฟการลู่เข้าของ IAC-PSO กับฟังก์ชัน Sum Squares ใน 50 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.22 กราฟการลู่เข้าของ IAC-PSO กับฟังก์ชัน Quartic ใน 50 มิติ



รูปที่ 5.23 กราฟการลู่เข้าของ IAC-PSO กับฟังก์ชัน Sextic ใน 50 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.18 – 5.20 คือกราฟการลู่เข้าของอัลกอริทึม Conventional PSO และรูปที่ 5.21 – 5.23 คือกราฟการลู่เข้าของอัลกอริทึม IAC-PSO ด้วยฟังก์ชันในส่วนการทดลองเพิ่มเติมทั้ง 3 ฟังก์ชันตามลำดับ

ก. เหตุผล

สาเหตุที่เป็นเช่นนี้เนื่องจากยิ่งเลขยกกำลังเพิ่มมากขึ้นฟังก์ชันจะมีรูปร่างแบน (Flat) ซึ่งส่งผลให้ฟังก์ชันมีจุด Local Optima น้อยลง สามารถสังเกตได้จากกราฟการลู่เข้า ยิ่งเลขยกกำลังมากขึ้น Conventional PSO จะติดอยู่ในจุด Local Optima น้อยลง

อีกสาเหตุหนึ่งที่ทำให้คำตอบของ IAC-PSO แย่กว่า Conventional PSO คือที่ฟังก์ชันที่มีรูปร่างแบนนั้นจะสังเกตได้จากกราฟการลู่เข้าของ IAC-PSO ภายในกราฟจะมีจุดที่การลู่เข้า (ค่า GBest) ถูกรีเซ็ต ซึ่งจะทำให้อนุภาคต้องเริ่มต้นการค้นหาใหม่นั้นเอง ซึ่งจะมีการรีเซ็ตเช่นนี้อีกตามจำนวน Generation นอกจากนี้จะสังเกตเห็นรอยหยักภายในช่วงการทำงานของ 1 Generation ซึ่งรอยหยักนี้คือการชะงักของการลู่เข้าโดยเกิดจากการเริ่มต้นตำแหน่งใหม่ของอนุภาคในทุกๆ การค้นหาที่ละมิติ เนื่องด้วย 2 สาเหตุนี้จึงทำให้การค้นหาของ IAC-PSO ไม่ต่อเนื่องเท่ากับ Conventional PSO ในฟังก์ชันที่มีรูปร่างแบน

5.6 การทดสอบประสิทธิภาพของอัลกอริทึม

ในส่วนนี้จะเป็นการทดลองเพื่อทดสอบประสิทธิภาพขององค์ประกอบจากอัลกอริทึมบางส่วน โดยจะวัดผลประสิทธิภาพในการทำงาน ซึ่งจะทดสอบองค์ประกอบในหัวข้อ 4.4 การคำนวณจุดเซนทรอยด์ (Centroid Calculation) และ 4.5 การวนรอบการทำงาน (Generation Repetition)

ก. การทดสอบประสิทธิภาพการคำนวณจุดเซนทรอยด์

การทดลองนี้จะนับจำนวนครั้งที่จุดเซนทรอยด์สามารถเป็นคำตอบที่ดีที่สุดในปัจจุบัน (จุดเซนทรอยด์มีค่า fitness มากกว่าจุด GBest) โดยจะใช้ค่าพารามิเตอร์เดียวกันกับการทดลองหลัก ดังนั้นการคำนวณจุดเซนทรอยด์ทั้งหมดของแต่ละฟังก์ชันจะเท่ากับ 52,500 สำหรับ 50 มิติ และ 105,000 สำหรับ 100 มิติ (คำนวณได้จาก $R \cdot I \cdot A$ เนื่องจากคำนวณทุกๆ Iteration ยกเว้น Iteration ในช่วง Preliminary run)

วัตถุประสงค์ของการทดลองนี้เพื่อทดสอบประสิทธิภาพการคำนวณจุดเซนทรอยด์ด้วยตัวแทนฟังก์ชันจากแต่ละกลุ่มโดยจะนำเสนอฟังก์ชันที่มีผลบอยที่สุดกับน้อยที่สุด และอีกประการคือ เพื่อ

เปรียบเทียบผลลัพธ์เมื่อจำนวนมิติเพิ่มมากขึ้น ผลการทดลองจะแสดงจำนวนครั้งที่จุดเซนทรอยด์สามารถเป็นคำตอบที่ดีที่สุดในปัจจุบัน และค่าเป็นกึ่งเปอร์เซ็นต์จากจำนวนครั้งที่คำนวณ

ตารางที่ 5.28 การทดสอบประสิทธิภาพการคำนวณจุดเซนทรอยด์

Function	Point	IAC-PSO	%
Noncontinuous Rastrigin (50 dimension)	Best	205.00	0.39
	Median	335.50	0.64
	Worst	434.00	0.83
	Mean	326.00	0.62
Noncontinuous Rastrigin (100 dimension)	Best	566.00	0.53
	Median	787.50	0.75
	Worst	1134.00	1.08
	Mean	798.62	0.76
Rosenbrock (50 dimension)	Best	259.00	0.49
	Median	303.00	0.58
	Worst	384.00	0.73
	Mean	300.68	0.57
Rosenbrock (100 dimension)	Best	540.00	0.51
	Median	640.50	0.61
	Worst	742.00	0.71
	Mean	641.10	0.61
Michalewicz (50 dimension)	Best	146	0.28
	Median	212	0.40
	Worst	279	0.53
	Mean	209.08	0.40
Michalewicz (100 dimension)	Best	436.00	0.42
	Median	547.00	0.52
	Worst	645.00	0.61
	Mean	547.54	0.52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.28 (ต่อ)

Function	Point	IAC-PSO	%
Sum of Different Powers (50 dimension)	Best	27.00	0.05
	Median	51.00	0.097
	Worst	86.00	0.16
	Mean	51.92	0.099
Sum of Different Powers (100 dimension)	Best	50.00	0.095
	Median	90.50	0.086
	Worst	167.00	0.16
	Mean	93.92	0.089
Zakharov (50 dimension)	Best	100.00	0.19
	Median	165.00	0.31
	Worst	258.00	0.49
	Mean	164.66	0.31
Zakharov (100 dimension)	Best	205.00	0.20
	Median	281.00	0.27
	Worst	377.00	0.36
	Mean	285.18	0.27

ผลการทดลองจะแสดงจำนวนครั้งที่จุดเซนทรอยด์สามารถเป็นคำตอบที่ดีที่สุดในปัจจุบัน และคิดเป็นเปอร์เซ็นต์จากจำนวนครั้งที่คำนวณ (คอลัมน์ทางด้านขวามือ) จากการทดลองสามารถสรุปได้ว่า ถึงแม้จุดเซนทรอยด์จะให้คำตอบที่ดีกว่าปัจจุบันได้ไม่ถึง 1 เปอร์เซ็นต์แต่ก็สามารถช่วยให้ค่าออกมาดีขึ้นระดับหนึ่ง นอกจากนี้เมื่อจำนวนมิติเพิ่มมากขึ้นจะส่งผลให้จุดเซนทรอยด์สามารถมีค่าดีกว่าปัจจุบันบ่อยขึ้น และผลการทดลองได้แสดงว่าฟังก์ชัน Rosenbrock จากกลุ่ม Valley-Shaped จุดเซนทรอยด์สามารถมีค่าได้ดีกว่าปัจจุบันบ่อยกว่าฟังก์ชันจากกลุ่มอื่นๆ ส่วนฟังก์ชัน Sum of Different Powers จากกลุ่ม Bowl-Shaped จุดเซนทรอยด์สามารถมีค่าได้ดีกว่าปัจจุบันน้อยกว่าฟังก์ชันจากกลุ่มอื่นๆ

ข. การทดสอบประสิทธิภาพการวนรอบการทำงาน

การทดลองนี้จะทดสอบผลของการ reset ค่า GBest ในทุกๆ Generation โดยจะใช้ conventional PSO ในการค้นหา เมื่ออัลกอริทึมไม่สามารถค้นหาค่าที่ดีกว่าค่า GBest ในปัจจุบันได้ภายในช่วง 10% ของ function evaluation times จะทำการ reset ค่า GBest และสุ่มตำแหน่งอนุภาคใหม่ทั่วทั้ง Search Space และเมื่อค้นหาจนครบ function evaluation times จะใช้ GBest ที่ดีที่สุดจากการค้นหาทั้งหมด (รวมถึงค่า GBest ที่ reset แล้วทั้งหมด) เป็นคำตอบ ทั้งนี้การทดสอบจะใช้ค่าพารามิเตอร์เดียวกันกับการทดลองหลัก ดังนั้นอัลกอริทึมจะ reset เมื่อไม่สามารถค้นหาค่าที่ดีกว่าได้ภายใน 6,000 สำหรับ 50 มิติ และ 12,000 สำหรับ 100 มิติ

โดยวัตถุประสงค์ของการทดลองนี้เพื่อวัดประสิทธิภาพของการวนรอบการทำงานด้วยฟังก์ชันทั้งหมด และทดสอบประสิทธิภาพเมื่อจำนวนมิติเพิ่มมากขึ้น

ตารางที่ 5.29 การทดสอบประสิทธิภาพการวนรอบการทำงานที่ 50 มิติ

Function	Point	Normal	Reset	IAC-PSO
Noncontinuous Rastrigin (50 dimension)	Best	32	54.00	0.00
	Median	120	89.00	0.00
	Worst	204	107.00	12.00
	Mean	112.64	86.74	0.64
Ackley (50 dimension)	Best	1.4655E-14	2.3165E+00	2.2204E-14
	Median	1.1551E+00	3.1068E+00	2.9310E-14
	Worst	1.4990E+01	4.3994E+00	3.2863E-14
	Mean	2.5777E+00	3.1676E+00	3.0376E-14
Griewank (50 dimension)	Best	0	0.00000019148	0
	Median	0.0074	0.00032633000	0
	Worst	90.7470	0.0688	0
	Mean	3.6298	0.0096	0
Generalized Penalized (50 dimension)	Best	9.4226E-33	6.7589E-05	9.4233E-33
	Median	6.2201E-02	5.6250E-01	9.4233E-33
	Worst	1.7501E+00	2.1212E+00	9.4233E-33
	Mean	2.1196E-01	6.8830E-01	9.4233E-33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.29 (ต่อ)

Function	Point	Normal	Reset	IAC-PSO
Rosenbrock (50 dimension)	Best	2.4468E+01	1.2168E+01	4.9146E-04
	Median	1.0267E+02	4.8453E+01	7.9362
	Worst	2.2455E+02	9.6623E+01	74.3372
	Mean	1.1620E+02	5.8930E+01	10.5793
Dixon-Price (50 dimension)	Best	6.6670E-01	6.6680E-01	6.6670E-01
	Median	3.6847E+00	6.7570E-01	6.6670E-01
	Worst	1.3470E+01	7.6817E+00	6.6670E-01
	Mean	3.8479E+00	9.5930E-01	6.6670E-01
Generalized Modified Rosenbrock's Function No.01 (50 dimension)	Best	1.31E+03	7.20E+02	137.5576
	Median	5.47E+03	1.23E+03	623.3544
	Worst	1.69E+05	3.01E+03	2.12E+03
	Mean	1.51E+04	1.38E+03	810.0134
Michalewicz (50 dimension)	Best	-44.6969	-43.4282	-44.0777
	Median	-37.7173	-40.7972	-41.6676
	Worst	-32.4989	-37.9733	-38.783
	Mean	-37.9831	-40.79	-41.5893
De Jong N. 5 (2 dimension)	Best	0.998	0.998	0.998
	Median	0.998	0.998	0.998
	Worst	0.998	0.998	0.998
	Mean	0.998	0.998	0.998
Easom (2 dimension)	Best	-1.00E+00	-1.00E+00	-1.00E+00
	Median	-1.00E+00	-1.00E+00	-1.00E+00
	Worst	-1.00E+00	-1.00E+00	-1.00E+00
	Mean	-1.00E+00	-1.00E+00	-1.00E+00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.29 (ต่อ)

Function	Point	Normal	Reset	IAC-PSO
Trid (50 dimension)	Best	-2.63E+03	-1.44E+04	-2.20E+04
	Median	7.76E+04	1.02E+04	-1.96E+04
	Worst	2.45E+05	6.29E+04	4.68E+03
	Mean	9.05E+04	1.34E+04	-1.82E+04
Sum of Different Powers (50 dimension)	Best	1.2913E-248	2.1859E-319	3.3612E-40
	Median	1.6573E-104	2.3974E-184	1.4374E-35
	Worst	2.8845E-41	2.8301E-78	1.7962E-33
	Mean	5.7690E-43	5.6603E-80	1.3987E-34
Sum Squares (50 dimension)	Best	1.4651E-08	1.4449E-10	9.9038E-44
	Median	1.1907E-05	5.9002E-08	4.0329E-43
	Worst	0.0145	2.67E-05	1.5986E-42
	Mean	0.0011	1.27E-06	4.4916E-43
Zakharov (50 dimension)	Best	4.5178E-234	3.7675E-37	1.4540E-17
	Median	6.7427E-222	3.7990E-35	6.3781E-16
	Worst	1.6644E-209	5.2057E-34	3.3713E-15
	Mean	3.5116E-211	8.4609E-35	7.5245E-16

ตารางที่ 5.29 ผลการเปรียบเทียบแสดงให้เห็นว่าการ reset ค่า GBest สามารถค้นหาค่าที่ดีที่สุดว่า อัลกอริทึมอื่นๆ ได้ใน 11 ฟังก์ชันจากทั้งหมด 14 ฟังก์ชัน ซึ่งคิดเป็นอัตราส่วนคือ 78.57% ซึ่งแสดงให้เห็นว่าการ reset ค่า GBest จะช่วยเพิ่มโอกาสในการค้นหาค่าที่ดีที่สุดปัจจุบัน

ตารางที่ 5.30 การทดสอบประสิทธิภาพการวนรอบการทำงานที่ 100 มิติ

Function	Point	Normal	Reset	IAC-PSO
Noncontinuous Rastrigin (100 dimension)	Best	140.00	208.000	11.00
	Median	288.00	262.125	32.00
	Worst	603.00	308.000	48.00
	Mean	298.62	261.275	31.60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.30 (ต่อ)

Function	Point	Normal	Reset	IAC-PSO
Ackley (100 dimension)	Best	1.72662E+00	8.08150E+00	5.77320E-14
	Median	1.29718E+01	9.45280E+00	6.83900E-14
	Worst	1.81503E+01	1.08396E+01	7.90480E-14
	Mean	1.12111E+01	9.43760E+00	6.85319E-14
Griewank (100 dimension)	Best	0	1.1585	0
	Median	90.9803	1.5970	0
	Worst	360.9793	2.4372	0
	Mean	126.4700	1.6115	0
Generalized Penalized (100 dimension)	Best	4.7116E-33	5.2712E+00	4.7116E-33
	Median	3.1101E-02	9.5427E+00	4.7116E-33
	Worst	7.8900E-01	1.1923E+01	4.7116E-33
	Mean	1.2770E-01	9.3360E+00	4.7116E-33
Rosenbrock (100 dimension)	Best	6.9956E+02	6.7589E-05	2.2717E-08
	Median	1.2088E+03	5.6250E-01	0.4483
	Worst	2.8613E+03	2.1212E+00	98.195
	Mean	1.3817E+03	6.8830E-01	23.4284
Dixon-Price (100 dimension)	Best	1.3510E+02	7.6937E+01	0.6667
	Median	4.8720E+02	1.9437E+02	0.6667
	Worst	5675.30	600.79	0.6667
	Mean	7.8467E+02	2.1468E+02	0.6667
Generalized Modified Rosenbrock's Function No.01 (100 dimension)	Best	2.43E+06	1.96E+06	465.4799
	Median	2.14E+07	6.07E+06	693.7022
	Worst	1.02E+08	1.73E+07	1.09E+03
	Mean	2.71E+07	7.18E+06	696.6161

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.30 (ต่อ)

Function	Point	Normal	Reset	IAC-PSO
Michalewicz (100 dimension)	Best	-76.2456	-77.3886	-87.63
	Median	-68.3292	-71.8888	-81.9545
	Worst	-61.0464	-68.3717	-76.4755
	Mean	-68.1044	-72.3933	-82.1025
Trid (100 dimension)	Best	6.16E+06	7.12E+06	-1.58E+05
	Median	1.76E+07	1.10E+07	2.62E+03
	Worst	4.01E+07	1.88E+07	5.18E+05
	Mean	1.83E+07	1.15E+07	4.06E+04
Sum of Different Powers (100 dimension)	Best	8.5556E-195	6.1615E-271	2.8320E-35
	Median	2.2444E-43	3.7551E-117	2.8350E-33
	Worst	3.4485E-26	3.4024E-59	3.6489E-29
	Mean	6.9171E-28	6.8048E-61	1.1331E-30
Sum Squares (100 dimension)	Best	19.3265	8.6013	7.44E-41
	Median	89.0816	26.5425	1.40E-40
	Worst	408.1147	60.9775	3.41E-40
	Mean	125.3078	28.6849	1.56E-40
Zakharov (100 dimension)	Best	1.1949E-46	1.2895E-07	3.8603E-10
	Median	1.0206E-37	5.6216E-06	2.6902E-08
	Worst	3.6519E-29	1.6389E-04	1.1061E-07
	Mean	7.3039E-30	1.9333E-05	3.0312E-08

ตารางที่ 5.30 แสดงให้เห็นว่าเมื่อเพิ่มจำนวนมิติเข้าไปการ reset ค่า GBest จะยิ่งส่งผลดีมากขึ้น สังเกตได้จาก 3 ฟังก์ชันจากเดิมที่ไม่สามารถค้นหาค่าได้ดีกว่าการค้นหาแบบปกติ สามารถค้นหาได้ดีขึ้นในการทดลองที่ 100 มิติเมื่อเปรียบเทียบกับผลทดลองใน 50 มิติ ดังนั้นจึงสามารถสรุปได้ว่า การ reset ค่า GBest จะช่วยให้สามารถค้นหาได้ดีในกรณีที่ค้นหาในฟังก์ชันที่มีมิติจำนวนมาก

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

6.1 สรุปผลการวิจัย

งานวิจัยนี้ได้นำเสนอการประยุกต์ PSO โดยใช้กลยุทธ์การเรียนรู้แบบเพิ่มพร้อมกับการใช้จุดเซนทรอยด์จากจุด GBest ล่าสุด โดยแตกต่างจากอัลกอริทึมที่ใช้กลยุทธ์การเรียนรู้แบบเพิ่มอื่นๆ เนื่องจากอนุภาคสามารถเคลื่อนที่ได้ในช่วงแคบๆ ใกล้กับตำแหน่งที่ดีที่สุดของมิติก่อนหน้าและมีภารกิจกำหนดตำแหน่งเริ่มต้นของอนุภาคเป็น 2 แบบ ซึ่งจะช่วยให้อัลกอริทึมหลีกเลี่ยงจุด Local Optima ได้ นอกจากนี้จุดเซนทรอยด์จะช่วยเพิ่มโอกาสค้นหาตำแหน่งที่ดีกว่าจุด GBest ในปัจจุบัน

การทดลองได้ทดสอบอัลกอริทึมนี้กับฟังก์ชันหลากหลายรูปแบบได้แก่ Many Local Minima, Valley-Shaped, Steep Ridges/Drops, Bowl-Shaped และ Plate-Shaped ซึ่งเดิมทีอัลกอริทึมนี้ถูกออกแบบมาเพื่อแก้ปัญหาฟังก์ชันแบบ Many Local Minima ที่ประกอบด้วยมิติจำนวนมาก ซึ่งสามารถแก้ปัญหาฟังก์ชันแบบ Many Local Minima ได้ดี อีกทั้งยังสามารถแก้ปัญหาฟังก์ชันรูปแบบอื่นๆ ได้ดีอีกด้วย

จากการทดลองและเปรียบเทียบในฟังก์ชันรูปแบบต่างๆ แสดงให้เห็นว่า IAC-PSO เหนือกว่าอัลกอริทึมอื่นในเรื่องของความแม่นยำ และเสถียรภาพเมื่อจำนวนมิติเพิ่มมากขึ้น แต่ทั้งนี้ IAC-PSO ยังคงค้นหาได้เร็วกว่า Conventional PSO เมื่อประสบกับปัญหา Flat Function ที่ประกอบด้วยมิติจำนวนน้อย ซึ่งสาเหตุเกิดจาก IAC-PSO มีการรีเซ็ตค่าที่ดีที่สุด (GBest) หลายครั้งและ IAC-PSO มีการกำหนดตำแหน่งอนุภาคใหม่ในทุกๆ การค้นหาทีละมิติ ส่งผลให้การค้นหาไม่ต่อเนื่อง

6.2 ข้อเสนอแนะ

IAC-PSO มีการรีเซ็ตค่าที่ดีที่สุดถึง 7 ครั้งทำให้การค้นหาไม่ต่อเนื่องและสิ้นเปลือง Function Evaluation Times แต่เพราะเหตุนี้ทำให้อัลกอริทึม IAC-PSO สามารถหลีกเลี่ยงการค้นหาที่ติดอยู่ในจุด Local Optima ได้ การหาตำแหน่งเริ่มต้นด้วย Conventional PSO อาจเกิดปัญหาในบางครั้ง เพราะหากจุดเริ่มต้นอาจอยู่ในตำแหน่งที่ไม่ดีก็จะส่งผลให้ผลลัพธ์ออกมาไม่ดีด้วยเช่นกัน

นอกจากนี้การค้นหาทีละมิติจะมีการกำหนดตำแหน่งอนุภาคทุกมิติ ซึ่งในบางฟังก์ชันอนุภาคกลับมาค้นหาในตำแหน่งเดิมซ้ำๆ หลังจากการกำหนดตำแหน่งใหม่ ซึ่งในที่นี้ผู้วิจัยสนใจใช้การบังคับให้ Particle แยกตัวออกจากกันเมื่อค้นหาที่ตำแหน่งเดิมเป็นเวลานานมาใช้ในการแก้ปัญหา

หากงานวิจัยนี้ไปพัฒนาต่อควรระวังการเพิ่มจำนวนจุด GBest ล่าสุดที่ใช้ในการหาจุดเซนทรอยด์ เพราะหากใช้จำนวนมากเกินไปจะทำให้ Particle หนีออกจากจุด Local Optima ได้ยาก

เพราะการเพิ่มจำนวนจุด GBest ล่าสุดจะเร่งการลู่เข้าของอนุภาคเข้าไปอยู่ในจุดต่ำสุดของจุด Local Optima

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการแข่งขันเพื่อการศึกษาเท่านั้น เมื่อนุญาดเห็นเข้าใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Optima ซึ่งโอกาสในการค้นหาที่ดีกว่าจุดต่ำสุดดังกล่าวมีค่อนข้างน้อยมาก ทางที่ดีควรใช้จำนวนน้อยๆหรือปรับค่าให้เหมาะสมสำหรับงานที่ค้นหา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] J. Kennedy and R. Eberhart, "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks. IV, pp. 1942–1948, 1995.
- [2] W. Bai, S. Cheng, E.M. Tadjouddine and S. Guan, "Incremental Attribute Based Particle Swarm Optimization". Proceedings of 2012 8th International Conference on Natural Computation (ICNC 2012). pp. 669-674, 2012.
- [3] S.-U. Guan and S.Li. Incremental learning with respect to new in coming input attributes. Neural Processing Letters, 14(3):241–260, December 2001.
- [4] S.-U. Guan and F. Zhu. An incremental approach to genetic algorithms based classification. IEEE Transactions on Systems, Man and Cybernetics Part B, 35(2):227–239, April 2005.
- [5] S.-U. Guan and W. Mo. Incremental evolution strategy for function optimization. International Journal of Hybrid Intelligence Systems, 3:187–203, 2006.
- [6] Y. Shi and R.C. Eberhart, "A modified particle swarm optimizer". Proceedings of IEEE International Conference on Evolutionary Computation. pp. 69–73, 1998.
- [7] W. Lin, Y. He and X. Zhao, "One-dimensional Searching-based Particle Swarm Optimization". Proceedings of 2012 2nd International Conference on Computer and Information Application (ICCA 2012). Pp. 214-217, 2012.
- [8] Computational Intelligence Laboratory, "Benchmark functions for PSO." [Online]. Available : http://www.cil.pku.edu.cn/resources/benchmark_pso/, 2018.
- [9] S. Sonja and B. Derek, "Test Functions and Datasets." [Online]. Available : <https://www.sfu.ca/~ssurjano/optimization.html>, 2018.
- [10] Anonymous, "Electric Power Systems Analysis & Nature-Inspired Optimization Algorithms." [Online]. Available : <http://al-roomi.org/benchmarks/unconstrained/n-dimensions#gkPageTop>, 2018.

ภาคผนวก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Improving Particle Swarm Optimization by Using Incremental Attribute Learning and Centroid of Particle's Best Positions

Sornmarong Srimakham^{#1} and Kietikul Jearanaitanakij^{#2}

Department of Computer Engineering

Faculty of Engineering

King Mongkut's Institute of Technology Ladkrabang

Bangkok, Thailand

E-mail: ^{#1}sornmarong.sri@gmail.com ^{#2}kietikul.je@kmitl.ac.th

Abstract—Particle Swarm Optimization (PSO) is a powerful algorithm that can search a solution for a function which contains a large number of peaks and valleys. However, PSO might encounter a difficulty when the function gets more complex or the number of attributes (dimensions) grows larger. This paper proposes a modification of PSO by using the incremental attribute strategy along with the centroid of particle's best positions to avoid the local minima which can easily occur in a multimodal problem. The experimental results from four standard benchmarks show that the proposed method can improve PSO in terms of optimality and stability when compared with the conventional PSO and another incremental attribute-based PSO.

Keywords: Particle Swarm Optimization, Incremental attribute learning strategy, Multimodal function

I. INTRODUCTION

Function optimization is the process of selecting the value to each variable to optimize (either maximizing or minimizing) the value of the function. However, there are a plenty of functions that cannot be optimized efficiently by mathematics, especially, when the number of attributes grows larger. One of the popular nature-inspired optimizing algorithms that can search for the near optimal solution is Particle Swarm Optimization (PSO). PSO [1, 2] is originally developed in order to simulate to movement of creatures in a bird flock or fish school as a social behavior. Although PSO contains stochastic in the algorithm, it cannot guarantee that the optimal solution is always found. PSO may be stuck in local optima when it operates under the complex multimodal function. Moreover, a large number of attributes expands the search space and obstructs PSO from the optimal position.

In order to alleviate the local minima problem, incremental attribute learning strategy was developed by Guan et al. [3-5] by focusing on one attribute at a time and other attributes are processed incrementally. All attributes, including those are not yet processed, are used to calculate the fitness of each individual particle. The incremental attribute learning strategy can generate more stable results than the conventional PSO. This strategy may be repeated for a certain number of iterations in order to ensure the global optima. Bai et al. developed the Incremental Attribute based PSO (IAPSO) [6] to solve the local optima problem and the rise of attributes. IAPSO searches each

attribute by four iterations to guarantee that the best position in an attribute is found. The algorithm then incrementally processes the next attribute with the same number of iterations. During the searching on the next attribute, IAPSO assigns all completed attributes with the best positions that have been found in previous steps. The algorithm repeats the whole process as a new generation when all attributes are searched.

We propose a modification of PSO by using the Incremental Attribute strategy and the Centroid of particle's global best positions (IAC-PSO.) The proposed algorithm also allows particles to narrowly move within the slack region around the best positions found in attributes which have been searched. This slack region helps the algorithm to reach a location which is better than the current best position. It is helpful when we optimize the complex function which contains many steep peaks. In addition, the global optima usually locate near the global best position. Therefore, calculating the centroid from the group of previous global best positions increases the chance of exploring the global optima.

This paper is categorized into sections as follows. Section II provides a short review of the conventional PSO algorithm. Section III explains the proposed algorithm, IAC-PSO. Section IV describes four standard functions and experimental results along with comparisons. Lastly, conclusions and guideline for the future work are given in Section V.

II. PARTICLE SWARM OPTIMIZATION

The conventional PSO algorithm begins initializing the positions of all particles in the swarm with random values in a certain range. The algorithm then calculates the fitness value for each particle and finds the global best position, $gbest$, from a set of local best positions, $pbest(s)$. Before ending the iteration, the position and velocity of every particle are updated with respect to its $pbest$ and the swarm's $gbest$. The algorithm iterates the process until the stopping criterion is satisfied. In order to represent the update process of each particle, let us define w as an inertia weight, v_{id} as the velocity of the i^{th} particle, x_{id} as the position of the i^{th} particle, p_{id} as the position of the i^{th} particle's $pbest$, p_{gd} as the position of the swarm ($gbest$), c_1 and c_2 as acceleration constants, and r as a random number. The subscript d in variable stands for the dimension that the algorithm is updating. Velocity and position of a particle are

updated at the end of iteration as described by Eq. (1) and (2), respectively.

$$v_{id} = wv_{id} + c_1r * (p_{id} - x_{id}) + c_2r * (p_{gd} - x_{id}) \quad (1)$$

$$x_{id} = x_{id} + v_{id} \quad (2)$$

III. PROPOSED ALGORITHM

The following subsections explain four main components of IAC-PSO. The complete algorithm is proposed at the end of this section along with its flowchart.

A. Position Initialization

Proper initial positions in all attributes of particles increase the chance of obtaining the global optimal solution in the searching space. These initial positions can be found by a preliminary run of the conventional PSO algorithm for a certain some number of iterations, e.g., 7,500 – 15,000 iterations. The algorithm uses the same set of initial positions every time that the new generation begins.

B. Processed Attribute Boundary

According to the incremental attribute learning strategy, the algorithm searches the next attribute based on the best positions found in previous attributes. IAC-PSO relaxes particles to move within a small slack region around the best positions found in previous attributes. This slack region reduces the chance that IAC-PSO will be trapped in the local optima since the best positions found in previous attributes may not guarantee the optimal location.

C. Centroid Calculation



Figure 1. Finding gbest's centroid in Ackley function

Since the position of gbest usually locates near the global optimal position, it is wise to calculate the centroid of previous gbest values, denoted by white dots. If the centroid gives better fitness value than the current gbest, the position of centroid will be assigned as a new gbest. Otherwise, we continue the next iteration with the current gbest. Fig. 1 illustrates a sample scenario in Ackley function when the centroid of previous gbests, represented as a black dot, is located near the global optimal position (0, 0).

D. Generation Repetition

According to IAPSO [6], the incremental attribute learning strategy can be applied repeatedly during the optimization. The

total number of times that the algorithm repeats can be treated as the number of generations in PSO. Repeating process as a generation ensures the algorithm having a high potential to find the global optimal solution.



Figure 2. Flowchart of the IAC-PSO

IAC-PSO algorithm can be described along with its flowchart shown in Fig. 2 as follows. At the beginning, the algorithm performs preliminary run with some number of iterations to determine proper initial positions of all particles. These initial positions are fixed to be the same for every generation. The algorithm then incrementally searches another attribute along with the comparison between gbest and its centroid at the end of iteration. After a certain number of iterations, the algorithm returns the best position of the attribute and continues searching on the next attribute. During the next attribute searching, particles are allowed to move within a small region around the best positions of previous attributes that have been searched. Once a complete solution (position which composes of final values of all attributes) is found, the algorithm repeats as a new generation and by relocating particles back to their initial positions. The number of repeated generations can be a small number, e.g., 4 – 7, in order to stabilize the best solution of the algorithm.

TABLE I THE BENCHMARK FUNCTIONS

Function	Equation	Range	Global minima
Noncontinuous Rastrigin	$f_1(x) = \sum_{i=1}^A [y_i^2 - 10 \cos(2\pi y_i) + 10]$ $y_i = \begin{cases} x_i & x_i < 0.5 \\ \text{round}(2x_i)/2 & x_i \geq 0.5 \end{cases}$	$[-5.12, 5.12]^A$	0
Ackley	$f_2(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{A} \sum_{i=1}^A x_i^2}\right) - \exp\left(\frac{1}{A} \sum_{i=1}^A \cos(2\pi x_i)\right) + 20 + e$	$[-32, 32]^A$	0
Griewank	$f_3(x) = \frac{1}{4000} \sum_{i=1}^A x_i^2 - \prod_{i=1}^A \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]^A$	0
Generalized Penalized	$f_4(x) = \frac{\pi}{A} \left[10 \sin(\pi y_1)^2 + \sum_{i=1}^{A-1} (y_i - 1)^2 + [1 + 10 \sin(\pi y_{A+1})^2] + (y_A - 1)^2 \right] + \sum_{i=1}^A u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{1}{4} (x_i + 1)$ $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	$[-50, 50]^A$	0

IV. EXPERIMENTS AND COMPARISONS

A. Benchmark functions

IAC-PSO's performance is measured by 4 standard multimodal functions as listed in Table I.

B. Experimental Study

In order to be a fair comparison, all common parameters are assigned to be the same values as those appeared in IAPSO [6]. The first four lines are common parameters and the rest items are custom settings for IAC-PSO.

- Population size: 160, Inertia weight: $w = 0.72984$
- Acceleration constants: $c_1 = c_2 = 1.496172$
- Stop criteria: Function evaluation times reached or minimum value found.
- Function evaluation times: 60,000 for 50 attributes and 120,000 for 100 attributes. These numbers includes those used in the preliminary run.
- Iterations for preliminary run: 7,500 for 50 attributes and 15,000 for 100 attributes.
- Iterations per attribute: 150
- Small slack region: 5% of search range
- Number of previous gbests for calculating the centroid: 10
- Repetitions: 7 generations

Two important parameters are configured by some preliminary runs based on the following concerns. If the slack region is too narrow, we might miss the global optima located around the current position. In addition, if the number of previous gbests for calculating the centroid is too large, some poor previous gbests are included in the centroid calculation and the quality of the centroid will deteriorate.

The number of function evaluations of this research can be calculated as follows. Let I_p be the number of iterations for preliminary run, R be the number of repetitions (generations), I be the number of iterations per attribute, and A be the total number of attributes. The number of function evaluations is $I_p + R * I * A$. For example, if there are 50 attributes to be optimized, the number of function evaluation times is $7,500 + 7 * 150 * 50 = 60,000$. Similarly, for 100 attributes, the number of function evaluation is $15,000 + 7 * 150 * 100 = 120,000$.

C. Experimental Results and Comparisons

This section reports the experimental results from 50 runs in form of statistical values, e.g., best, median, worst, and mean. The results are shown in Table II for 50 attributes and Table III for 100 attributes. These Tables also show comparisons between conventional PSO, IAPSO, and IAC-PSO for each benchmark function.

TABLE II RESULTS OF 50 ATTRIBUTES: (A) NONCONTINUOUS RASTRIGIN, (B) ACKLEY, (C) GRIEWANK, AND (D) GENERALIZED PENALIZED

Result	PSO	IAPSO	IAC-PSO
Best	32.000000000002	21.0000003030006	3.000000000001
Median	120.000000000010	64.0491506226967	10.000000000030
Worst	204.000000000005	134.000020310886	18.000000000001
Mean	112.640000000002	65.4223177420262	10.280000000000

(a)

Result	PSO	IAPSO	IAC-PSO
Best	1.4654943925E-14	1.46549439250E-14	2.2204E-14
Median	1.15514850270983	2.17603712826E-14	3.2863E-14
Worst	14.9902685922491	4.30766533554E-14	3.9968E-14
Mean	2.57774800078812	2.38209452163E-14	3.0589E-14

(b)

Result	PSO	IAPSO	IAC-PSO
Best	0	0	0
Median	0.00739604033411	0.007396040334115	0
Worst	90.7470094566445	0.066071506229681	0
Mean	3.62979602487837	0.008999462284951	0

(c)

Result	PSO	IAPSO	IAC-PSO
Best	9.4226462131E-33	9.42264621310E-33	9.4233E-33
Median	0.06220141215029	3.88522167545E-32	9.4233E-33
Worst	1.75006860865319	0.062201412150297	9.4233E-33
Mean	0.21195792378163	0.001244028243005	9.4233E-33

(d)

For 50-attribute comparisons, IAC-PSO significantly outperforms both conventional and IAPSO algorithms in most benchmarks, except for Ackley function where IAPSO and IAC-PSO are competitive.

TABLE III. RESULTS OF 100 ATTRIBUTES: (A) NONCONTINUOUS RASTRIGIN, (B) ACKLEY, (C) GRIEWANK, AND (D) GENERALIZED PENALIZED

Result	PSO	IAPSO	IAC-PSO
Best	140.0000000000005	133.004257747039	11.0000000000001
Median	288.0000000000097	249.008493314756	31.5000000000120
Worst	603.000000001311	455.000264519153	67.0000000000005
Mean	298.620000011377	262.54740515019	32.1000000000070

(a)

Result	PSO	IAPSO	IAC-PSO
Best	1.72661744742322	5.101385980E-11	5.7732E-014
Median	12.9718350132823	4.304863132E-10	6.8390E-014
Worst	18.1503080702469	1.4235364879729	9.3259E-014
Mean	11.2111404237847	0.0778188378343	6.8887E-014

(b)

Result	PSO	IAPSO	IAC-PSO
Best	0	0	0
Median	90.9803161636051	0	0
Worst	360.979343546945	0.070599383003331	0
Mean	126.470042547612	0.008068312510022	0

(c)

Result	PSO	IAPSO	IAC-PSO
Best	4.71132310655E-33	4.889018957578E-22	4.7116E-33
Median	0.031100706075148	1.316591049873E-19	4.7116E-33
Worst	0.788995697611049	0.0932891936157191	4.7116E-33
Mean	0.127699232048925	0.0105739815733561	4.7116E-33

(d)

Comparing the experimental results in higher attributes can prove the ability of the algorithm to optimize the problem with larger search space. The results in Table III show the performances of three algorithms in 100 attributes. Although IAC-PSO cannot explore the global minima in three benchmarks, it achieves the best results comparing among other algorithms.

Besides the experimental results given above, we also provide the convergence graphs showing the relation between

fitness value and the number of function evaluations in Fig. 3. The global best point (represented as the black dot with its position shown at the upper right corner) among all generations is used as the best solution of an algorithm.

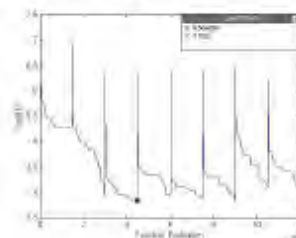


Figure 3. Convergence of 100 attributes for Noncontinuous Rastrigin

Each spike in Fig. 3 represents the reset for each swarm generation in IAC-PSO. Although IAC-PSO repeats the optimizing process for 7 generations, the convergence graphs indicates that the algorithm can find the optimal positions during the early generations. We may reduce the number of repeating generations without affecting the final solution.

V. CONCLUSIONS

We propose a modification of PSO by using the incremental attribute strategy along with the centroid of particle's best positions. Unlike other incremental attribute strategy algorithms, the particles are allowed to move within a small slack region around the best position found in previous attributes. In addition, the centroid of global best positions increases the probability to discover better position comparing to the current gbest. The experimental results and comparisons show that IAC-PSO is superior to other algorithms in terms of optimality and stability. One of the future works of IAC-PSO is to force particles in the swarm to spread out when a group of particles are clustered at the same position for a long period of time. Allowing particles to reposition in other locations can further reduce the chance that the swarm will be stuck at the local optima.

REFERENCES

- [1] J. Kennedy and R. Eberhart, "Particle Swarm Optimization", Proceedings of IEEE International Conference on Neural Networks, IV, pp. 1942-1948, 1995.
- [2] Y. Shi and R.C. Eberhart, "A modified particle swarm optimizer", Proceedings of IEEE International Conference on Evolutionary Computation, pp. 69-73, 1998.
- [3] S.-U. Guan and S. Li, "Incremental learning with respect to new in coming input attributes", Neural Processing Letters, 14(3):241-260, December 2001.
- [4] S.-U. Guan and F. Zhu, "An incremental approach to genetic algorithms based classification", IEEE Transactions on Systems, Man and Cybernetics Part B, 35(2):227-239, April 2005.
- [5] S.-U. Guan and W. Mo, "Incremental evolution strategy for function optimization", International Journal of Hybrid Intelligence Systems, 3:187-203, 2006.
- [6] Bai, W., Cheng, S., Tadjouddine E.M. and Guan, S. (2012). "Incremental Attribute Based Particle Swarm Optimization", 2012 8th International Conference on Natural Computation (ICNC 2012), Page 669-674.

ประวัติผู้เขียน

ชื่อ-นามสกุล นายสรณรงค์ ศรีมาคาม
 วัน เดือน ปีเกิด 27 ธันวาคม 2536 ที่นนทบุรี
 ที่อยู่ 83/441 หมู่บ้านเสนาแกรนด์โฮม ถ.ซ่อมสร้าง
 ต.บางขุน ปทุมฯ 12000
 ประวัติการศึกษา 2558 วิศวกรรมศาสตรบัณฑิต ภาควิชาคอมพิวเตอร์
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ความชำนาญเฉพาะด้าน 1.) การเขียน โปรแกรมในภาษาต่างๆ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้