

การปรับปรุงอัลกอริทึมฝังมดโดยใช้อัลกอริทึมการค้นหาแบบนกคุเหว่า
สำหรับระบบอนุกรม

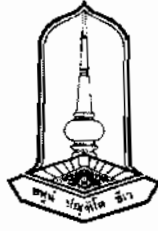
มานะ โสภ

เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

สิงหาคม 2559

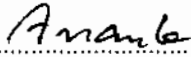
ลิขสิทธิ์เป็นของมหาวิทยาลัยมหาสารคาม





คณะกรรมการสอบวิทยานิพนธ์ ได้พิจารณาวิทยานิพนธ์ของนายมานะ โสภา
แล้วเห็นสมควรรับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาปรัชญาดุษฎีบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ ของมหาวิทยาลัยมหาสารคาม

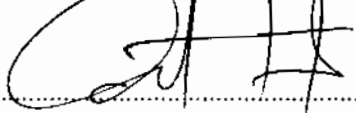
คณะกรรมการสอบวิทยานิพนธ์


.....
(ผศ.ดร.อนันต์ เครือทรัพย์ถาวร)

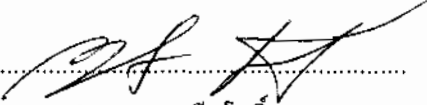
ประธานกรรมการ
(ผู้ทรงคุณวุฒิ)


.....
(ผศ.ดร.นิวัฒน์ อังควิศิษฐพันธ์)

กรรมการ
(อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก)



.....
(รศ.ดร.วรวัฒน์ เสี่ยมวิบูล)


กรรมการ
(อาจารย์บัณฑิตศึกษาประจำคณะ)


.....
(ผศ.ดร.ชลธิ โพธิ์ทอง)

กรรมการ
(อาจารย์บัณฑิตศึกษาประจำคณะ)

มหาวิทยาลัยอนุมัติให้รับวิทยานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ ของมหาวิทยาลัยมหาสารคาม


.....
(รศ.ดร.อนงค์ฤทธิ์ แข็งแรง)
คณบดีคณะวิศวกรรมศาสตร์


.....
(ศ.ดร.ประดิษฐ์ เทอดทูล)
คณบดีบัณฑิตวิทยาลัย
วันที่ 14 เดือน 5.9 พ.ศ. 2559



กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จสมบูรณ์ได้ด้วยความกรุณาและความช่วยเหลืออย่างสูงยิ่งจาก ผู้ช่วยศาสตราจารย์ ดร.นิวัตร อังควิศิษฐพันธ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก ผู้ช่วยศาสตราจารย์ ดร.อนันต์ เครือทรัพย์ถาวร ประธานกรรมการสอบ รองศาสตราจารย์ ดร.วรวิวัฒน์ เสงี่ยมวิบูล และ ผู้ช่วยศาสตราจารย์ ดร.ชลธิ โพร้ทอง กรรมการสอบ

ขอขอบคุณ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยมหาสารคาม ให้ความช่วยเหลือด้านเครื่องมือ และอุปกรณ์ต่าง ๆ ในการทำวิทยานิพนธ์ให้เสร็จสมบูรณ์

ขอขอบคุณ มหาวิทยาลัยราชภัฏอุดรธานี ที่ให้ทุนการศึกษา ตลอดจนให้ความช่วยเหลือ ด้านเครื่องมือและอุปกรณ์ต่าง ๆ ในการทำวิทยานิพนธ์ให้เสร็จสมบูรณ์

ขอขอบคุณพี่ ๆ เพื่อน ๆ ทั้งปริญญาเอกและปริญญาโท สาขาวิชาวิศวกรรมไฟฟ้าและ คอมพิวเตอร์ทุกคนที่ให้ความช่วยเหลือเรื่องการเรียนรู้ การสอบทุก ๆ ครั้ง จนสำเร็จการศึกษา

สุดท้ายขอขอบคุณครอบครัว คุณพ่อ คุณแม่ที่ให้กำเนิด ที่ช่วยอบรมเลี้ยงดู และภรรยา ที่คอยให้กำลังใจ เพื่อให้การเรียนรู้สำเร็จลุล่วงไปได้ด้วยดี

มานะ โสภาก



ชื่อเรื่อง	การปรับปรุงอัลกอริทึมฝูงมดโดยใช้อัลกอริทึมการค้นหาแบบนกดูเหว่า สำหรับระบบอนุกรม
ผู้วิจัย	นายมานะ โสภาก
ปริญญา	ปรัชญาดุษฎีบัณฑิต สาขาวิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.นิวัตร อังควิศิษฐพันธ์
มหาวิทยาลัย	มหาวิทยาลัยมหาสารคาม ปีที่พิมพ์ 2559

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์ เพื่อ 1) เพื่อปรับปรุงอัลกอริทึมฝูงมดโดยใช้อัลกอริทึมการค้นหาแบบนกดูเหว่าสำหรับระบบอนุกรม 2) เพื่อทดสอบประสิทธิภาพของอัลกอริทึมฝูงมดที่ปรับปรุงใหม่โดยใช้ฟังก์ชันทดสอบมาตรฐาน และ 3) เพื่อทดสอบประสิทธิภาพของอัลกอริทึมฝูงมดที่ปรับปรุงใหม่ในการแก้ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดของระบบอนุกรมที่มีข้อจำกัดหลายทางเลือกและงบประมาณ ผลการวิจัยพบว่า อัลกอริทึมฝูงมดร่วมกับอัลกอริทึมการค้นหาแบบดูเหว่า (Hybrid Ant Colony Optimization Cuckoo Search: Hybrid ACO-CS) สามารถแก้ปัญหาออปติไมเซชันเชิงตัวเลขได้ทุกฟังก์ชันที่นำเสนอและมีประสิทธิภาพสูงในการแก้ปัญหาฟังก์ชัน De Jong, ฟังก์ชัน Schwefel (Unimodal), ฟังก์ชัน Ackley, ฟังก์ชัน Griewank และฟังก์ชัน Quartic และการทดสอบประสิทธิภาพของอัลกอริทึมที่นำเสนอสามารถแก้ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดของระบบอนุกรมที่มีข้อจำกัดหลายทางเลือกและงบประมาณ ประกอบด้วยระบบอนุกรมที่มี 15 ระบบย่อย และมีตัวแปรที่ใช้ในการตัดสินใจ 60, 80 และ 100 ตัวแปร อัลกอริทึม Hybrid ACO-CS และ ACO2SWAP ให้ผลลัพธ์การแก้ปัญหาที่ดีและแม่นยำ และใช้เวลาในการค้นหาคำตอบมากขึ้นสำหรับระบบอนุกรมที่มีข้อมูลเมตาฮิวริสติกขนาดใหญ่

คำสำคัญ : อัลกอริทึมฝูงมด ; อัลกอริทึมการค้นหาแบบนกดูเหว่า ; ระบบอนุกรม ; เมตาฮิวริสติก



TITLE An Improved Ant Colony Algorithm using Cuckoo Search Algorithm for a Series System

AUTHOR Mr. Mana Sopa-

DEGREE Doctor of Philosophy **MAJOR** Electrical and Computer Engineering

ADVISORS Asst. Prof. Niwat Angkawisittpan, Ph.D.

UNIVERSITY Maharakham University **YEAR** 2016

ABSTRACT

This research aims to 1) to improve the ant colony algorithm using cuckoo search algorithm for a series system 2) to test the performance of the propose algorithm by using the optimization test functions and 3) to test the effectiveness of the propose algorithm to solve the problem of finding the optimal reliability of a serial systems with budget constraint and multiple-choice. The results showed that the algorithm called Hybrid Ant Colony Optimization Cuckoo Search (Hybrid ACO-CS) can solved all the functions and highly effective solution functions De Jong, Schwefel (Unimodal), Ackley, Griewank and Quartic functions. The Hybrid ACO-CS and ACO2SWAP algorithm can solved the problem of finding the optimal reliability of a serial systems, the system consists of a series of 15 subsystem 60, 80 and 100 decision variable with results in better solutions and precision and take the time to find more answers for the large meta-heuristic.

Key Words : Ant colony algorithm ; Cuckoo Search ; Series system ; Meta-heuristic



สารบัญ

หน้า

กิตติกรรมประกาศ	ก
บทคัดย่อภาษาไทย	ข
บทคัดย่อภาษาอังกฤษ	ค
สารบัญตาราง	ฉ
สารบัญภาพประกอบ	ณ
บทที่ 1 บทนำ	1
1.1 หลักการและเหตุผล	1
1.2 ความมุ่งหมายของการวิจัย	3
1.3 ขอบเขตของการวิจัย	3
1.4 การดำเนินการวิจัย	4
1.5 ประโยชน์ที่คาดว่าจะได้รับ	5
บทที่ 2 ปริทัศน์เอกสารข้อมูล	6
2.1 เมตาฮิวริสติก (Meta-Heuristic)	6
2.2 ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมของระบบอนุกรม (Reliability Optimization Problems for a Series System)	10
2.3 อัลกอริทึมฝูงมด (Ant Colony Algorithm)	12
2.4 อัลกอริทึมการค้นหาแบบนกคuckoo (Cuckoo search algorithm)	17
2.5 การเคลื่อนที่แบบเลวี (Lévy flight)	21
2.6 งานวิจัยที่เกี่ยวข้อง	22
บทที่ 3 วิธีดำเนินการวิจัย	26
3.1 ฟังก์ชันวัตถุประสงค์	26
3.2 การปรับปรุงอัลกอริทึมฝูงมด	27
3.3 การทดสอบประสิทธิภาพของอัลกอริทึม	34
3.4 การวิเคราะห์ข้อมูล	41



สารบัญ (ต่อ)

	หน้า
บทที่ 4 ผลการวิจัย	42
4.1 ผลการทดสอบประสิทธิภาพอัลกอริทึมกับฟังก์ชันทดสอบการหาค่าที่เหมาะสม ...	42
4.2 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เหมาะสมของอัลกอริทึม	52
4.3 ผลการทดสอบประสิทธิภาพของอัลกอริทึมกับระบบอนุกรม	57
บทที่ 5 สรุปผล อภิปรายผล และข้อเสนอแนะ	97
5.1 สรุปผลการวิจัย	97
5.2 อภิปรายผล	97
5.3 ข้อเสนอแนะในการพัฒนาต่อไป	98
เอกสารอ้างอิง	99
ภาคผนวก	105
ประวัติย่อผู้วิจัย	116



สารบัญตาราง

	หน้า
ตาราง 1 ตัวอย่างข้อมูลอิวิริสติกของปัญหา	11
ตาราง 2 สรุปฟังก์ชันทดสอบมาตรฐาน	35
ตาราง 3 ค่าพารามิเตอร์ของอัลกอริทึมอัลกอริทึมที่นำเสนอที่สามารถปรับค่าได้	36
ตาราง 4 สรุปกรณีศึกษาในการทดสอบประสิทธิภาพของอัลกอริทึมกับระบบอนุกรม	40
ตาราง 5 ผลการเปรียบเทียบประสิทธิภาพระหว่าง Hybrid ACO-CS กับ ACS	51
ตาราง 6 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เหมาะสมสำหรับอัลกอริทึม ACO2SWAP	53
ตาราง 7 เวลาเฉลี่ยที่ใช้ในการประมวลผลอัลกอริทึม ACO2SWAP ของค่าพารามิเตอร์ α และ β	53
ตาราง 8 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เกี่ยวข้องกับอัลกอริทึมฝูงมด	54
ตาราง 9 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เกี่ยวข้องกับอัลกอริทึมนกดูเหว่า	55
ตาราง 10 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เกี่ยวข้องกับเลวีไฟลท์	56
ตาราง 11 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 1	59
ตาราง 12 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 1	59
ตาราง 13 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 1	61
ตาราง 14 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 1	61
ตาราง 15 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 1	63
ตาราง 16 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 1	63
ตาราง 17 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 1	65
ตาราง 18 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 2	69
ตาราง 19 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 2	67



สารบัญตาราง (ต่อ)

	หน้า
ตาราง 20 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 2	69
ตาราง 21 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 2	69
ตาราง 22 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 2	71
ตาราง 23 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวม ของอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 2	71
ตาราง 24 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 2	73
ตาราง 25 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 3	74
ตาราง 26 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 3	75
ตาราง 27 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 3 ..	76
ตาราง 28 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 3	77
ตาราง 29 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 3 ..	78
ตาราง 30 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 3	79
ตาราง 31 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 3	80
ตาราง 32 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 4	82
ตาราง 33 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 4	82
ตาราง 34 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 4	84
ตาราง 35 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 4	84



สารบัญตาราง (ต่อ)

	หน้า
ตาราง 36 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 4	86
ตาราง 37 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 4	86
ตาราง 38 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึมทั้ง 3 วิธี สำหรับ กรณีศึกษาที่ 4	88
ตาราง 39 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 5	89
ตาราง 40 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 5	90
ตาราง 41 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 5	91
ตาราง 42 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 5	92
ตาราง 43 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 5	93
ตาราง 44 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 5	94
ตาราง 45 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 5	95
ตาราง 46 ระบบอนุกรมที่มี 15 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 60 ตัวแปร	107
ตาราง 47 ระบบอนุกรมที่มี 15 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 80 ตัวแปร	109
ตาราง 48 ระบบอนุกรมที่มี 15 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 100 ตัวแปร	110
ตาราง 49 ระบบอนุกรมที่มี 25 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 166 ตัวแปร	111
ตาราง 50 ระบบอนุกรมที่มี 40 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 266 ตัวแปร	113



สารบัญภาพประกอบ

	หน้า
ภาพประกอบ 1 ไดอะแกรมระบบอนุกรมที่มีหลายทางเลือก	10
ภาพประกอบ 2 การหาระยะทางที่สั้นที่สุดของฝูงมดจริง	13
ภาพประกอบ 3 รหัสเทียมอัลกอริทึมฝูงมดดั้งเดิม	13
ภาพประกอบ 4 ผังงานอัลกอริทึมฝูงมดดั้งเดิม	15
ภาพประกอบ 5 พฤติกรรมการวางไข่ของนกคุเหว่า	18
ภาพประกอบ 6 รหัสเทียมอัลกอริทึมการค้นหาแบบนกคุเหว่า	18
ภาพประกอบ 7 ผังงานอัลกอริทึมการค้นหาแบบนกคุเหว่า	19
ภาพประกอบ 8 ตัวอย่างการเคลื่อนที่แบบเลวี (Lévy flight)	22
ภาพประกอบ 9 รหัสเทียมการปรับปรุงอัลกอริทึมฝูงมด	23
ภาพประกอบ 10 รหัสเทียมการปรับปรุงอัลกอริทึมฝูงมดฟิชซีเซ็ด	24
ภาพประกอบ 11 แบบจำลองเส้นทางระหว่างรังมดกับแหล่งอาหาร	28
ภาพประกอบ 12 ผังงานอัลกอริทึมฝูงมดที่ปรับปรุงใหม่โดยใช้วิธีการค้นหาพื้นที่ใกล้เคียง ...	30
ภาพประกอบ 13 ผังงานอัลกอริทึมฝูงมดที่ปรับปรุงใหม่โดยใช้หลักการเลวีไฟลท์	33
ภาพประกอบ 14 ผังงานการปรับปรุงอัลกอริทึมฝูงมดร่วมกับอัลกอริทึมการค้นหา แบบนกคุเหว่า	34
ภาพประกอบ 15 แผนภาพสรุปแบบแผนการทดลองและปรับแต่งค่าพารามิเตอร์ สำหรับอัลกอริทึม ACO2SWAP	37
ภาพประกอบ 16 แผนภาพสรุปแบบแผนการทดลองและปรับแต่งค่าพารามิเตอร์ สำหรับอัลกอริทึม ACO-Lévy flight และ Hybrid ACO-CS	39
ภาพประกอบ 17 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Sphere	42
ภาพประกอบ 18 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Schwefel (Unimodal)	43
ภาพประกอบ 19 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Step	44
ภาพประกอบ 20 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Rosenbrock	45
ภาพประกอบ 21 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Ackley	46
ภาพประกอบ 22 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Rastrigin	47
ภาพประกอบ 23 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Griewank	48
ภาพประกอบ 24 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Stochastic ของ Kowalik	49
ภาพประกอบ 25 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Quartic	50



สารบัญภาพ (ต่อ)

	หน้า
ภาพประกอบ 26 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Shekel's Foxholes	51
ภาพประกอบ 27 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 1	58
ภาพประกอบ 28 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 1	60
ภาพประกอบ 29 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 1	62
ภาพประกอบ 30 ผลการเปรียบเทียบระหว่างอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 1	64
ภาพประกอบ 31 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 2	66
ภาพประกอบ 32 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 2	68
ภาพประกอบ 33 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 2	70
ภาพประกอบ 34 ผลการเปรียบเทียบระหว่างอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 2	72
ภาพประกอบ 35 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 3	74
ภาพประกอบ 36 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 3	76
ภาพประกอบ 37 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 3	78
ภาพประกอบ 38 ผลการเปรียบเทียบระหว่างอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 3	80
ภาพประกอบ 39 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 4	81
ภาพประกอบ 40 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 4	83
ภาพประกอบ 41 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 4	85
ภาพประกอบ 42 ผลการเปรียบเทียบระหว่างอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 4	87
ภาพประกอบ 43 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 5	89



สารบัญภาพ (ต่อ)

	หน้า
ภาพประกอบ 44 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 5	91
ภาพประกอบ 45 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 5	93
ภาพประกอบ 46 ผลการเปรียบเทียบระหว่างอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 5	95



1.1 หลักการและเหตุผล

การหาค่าที่เหมาะสมที่สุด (Optimization) ในงานด้านวิศวกรรมมีความจำเป็นอย่างยิ่งสำหรับงานออกแบบอุปกรณ์ทางไฟฟ้าและอิเล็กทรอนิกส์ วงจรควบคุมทางไฟฟ้า การบำรุงรักษาในระบบวิศวกรรม ปัญหาทางด้านวิศวกรรมไฟฟ้า ปัญหาทางด้านวิศวกรรมคอมพิวเตอร์ และปัญหาทางด้านวิศวกรรมอื่น ๆ อีกมากมาย ยกตัวอย่าง เช่น การออกแบบการควบคุมเครื่องกำเนิดไฟฟ้า ให้จ่ายกระแสไฟฟ้าอย่างประหยัด การออกแบบอุปกรณ์ทางไฟฟ้าที่เหมาะสมที่สุด การคัดเลือกชิ้นส่วนอุปกรณ์ที่มีความน่าเชื่อถือสูงสุดสำหรับออกแบบและผลิตสิ่งอำนวยความสะดวกประเภทต่าง ๆ การบีบอัดขนาดความจุของไฟล์รูปภาพให้เล็กลงแต่ยังคงความคมชัดมากที่สุด เป็นต้น ปัญหาการหาค่าที่เหมาะสมที่สุด (Optimization problem) เหล่านี้เป็นปัญหาที่มีความยุ่งยากและซับซ้อนอย่างมาก วิศวกรจึงต้องเลือกเทคนิคและวิธีการที่เหมาะสมสำหรับค้นหาคำตอบเพื่อใช้ในการตัดสินใจหรือแก้ปัญหาให้บรรลุเป้าหมายสูงสุดตามที่ต้องการ

เทคนิคการหาค่าที่เหมาะสมที่สุด (Optimization techniques) จึงเป็นวิธีการที่ถูกนำมาใช้แก้ปัญหาการหาค่าที่เหมาะสมที่สุดทางด้านวิศวกรรม ซึ่งมี 3 ขั้นตอนหลัก คือ ขั้นที่ 1 ทำความเข้าใจกับปัญหาโดยภาพรวมแล้วเปลี่ยนเป็นแบบจำลองทางคณิตศาสตร์ ขั้นที่ 2 กำหนดฟังก์ชันวัตถุประสงค์หรือเป็นการกำหนดเป้าหมายที่ต้องการ เป็นค่าสูงสุด (Maximum) หรือ ค่าต่ำสุด (Minimum) และเงื่อนไขหรือข้อจำกัด (Constraints) และขั้นตอนสุดท้ายหาคำตอบที่เป็นไปได้ตามขั้นตอนวิธีหรืออัลกอริทึม (Algorithm) นั้น ๆ [1]

ในการออกแบบระบบทางด้านวิศวกรรมและอุตสาหกรรม เช่น ระบบการผลิต, ระบบฮาร์ดแวร์ ระบบสื่อสารโทรคมนาคม, ระบบไฟฟ้ากำลัง หรือสิ่งอำนวยความสะดวกต่าง ๆ จำเป็นต้องให้ความสำคัญในการออกแบบและพิจารณาปัจจัยต่าง ๆ หลายด้าน ได้แก่ ความเป็นได้ การจัดลำดับความสำคัญของระบบ ประสิทธิภาพ งบประมาณ และคุณลักษณะอื่น ๆ รวมถึงความน่าเชื่อถือของระบบซึ่งเป็นตัวชี้วัดหนึ่งที่แสดงถึงคุณภาพในการออกแบบและคุณภาพของผลิตภัณฑ์ ดังกรณีศึกษาจากงานวิจัยที่ผ่านมา เช่น การหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดตามข้อจำกัดของประสิทธิภาพและงบประมาณ [2] การหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดตามข้อจำกัดของที่มีส่วนประกอบหลายทางเลือกและงบประมาณ [3] การหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดของปัญหาการจัดสรรความซ้ำซ้อนของระบบอนุกรม-ขนาน [4-12] และปัญหาการออกแบบระบบ k-out-of-n [13-15] เป็นต้น ซึ่งการแก้ปัญหาการหาค่าความน่าเชื่อถือของงานวิจัยที่กล่าวถึงข้างต้นได้นำเสนออัลกอริทึมที่แตกต่าง



กันไปเพื่อนำมาใช้แก้ปัญหา เช่น อัลกอริทึมเชิงพันธุกรรม (Genetic algorithm: GA) อัลกอริทึมฝูงมด (Ant colony optimization: ACO), อัลกอริทึมฝูงนก (Particle Swarm Optimization: PSO), อัลกอริทึมฝูงผึ้ง (Bee Colony Optimization: BCO) รวมถึงการปรับปรุงอัลกอริทึมดั้งเดิมให้สามารถแก้ปัญหาให้ดียิ่งขึ้นกว่าเดิม ซึ่งโดยภาพรวมแล้วการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดนั้นยังคงต้องมีการกำหนดข้อจำกัดบางอย่างเพื่อให้ผลลัพธ์เป็นไปตามเป้าหมายที่กำหนด

การแก้ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดสำหรับระบบอนุกรมตามข้อจำกัดที่มีหลายทางเลือกและงบประมาณเป็นปัญหาที่ยังไม่ได้รับความสนใจมากนัก ปัญหานี้ถูกจัดเป็นปัญหาในกลุ่มเอ็นพีแบบยากและกำหนดปัญหาสำหรับการเขียนโปรแกรมเป็นแบบจำนวนเต็มไบนารี ด้วยฟังก์ชันวัตถุประสงค์ไม่เชิงเส้น [16-17] ซึ่งในช่วงที่ผ่านมาได้มีการนำเลนอเทคนิคและวิธีการแก้ปัญหาด้วยวิธีการที่แตกต่างกันไป ได้แก่ อัลกอริทึมบรันช์และบาวด์ [18], การประยุกต์ใช้นิวรอลเน็ตเวิร์ก [19], อัลกอริทึมฝูงมด [16], อัลกอริทึมฝูงมดร่วมกับฟิชซีลोजิก [17] รวมถึงได้มีการนำตัวแบบปัญหานี้ไปประยุกต์ใช้กับปัญหาการหาค่าที่เหมาะสมที่สุดของผลผลิต (Yield Optimization) โดยใช้อัลกอริทึมฝูงผึ้ง เพื่อแก้ปัญหาการผลิตฮาร์ดดิสก์ [20] ซึ่งวิธีการที่กล่าวถึงในข้างต้นสำหรับปัญหาที่มีตัวแปรในการตัดสินใจขนาดใหญ่ยังคงหาค่าที่เหมาะสมได้ไม่มากนัก

อัลกอริทึมฝูงมด เป็นอัลกอริทึมแรกที่น่าเสนอวิธีการหาค่าที่เหมาะสมที่สุดด้วยอัลกอริทึมฝูงมด (ACO) ที่มีทั้งหมด นำเสนอโดย Dorigo and Gambardella [21] อัลกอริทึมนี้ได้แรงบันดาลใจจากพฤติกรรมการหาอาหารของมดจริง ซึ่งมดจริงสามารถหาเส้นทางที่สั้นที่สุดจากรังมดไปยังแหล่งอาหารได้ทั้ง ๆ ที่มองไม่เห็นเส้นทาง เมื่อมดออกหาอาหาร มดฝูงแรกจะออกสำรวจหาแหล่งอาหารในบริเวณใกล้เคียงของมันแบบสุ่ม ทันทีที่มดพบแหล่งอาหารมันจะประเมินแหล่งอาหารนั้นและนำอาหารกลับรัง ในขณะที่มดเดินทางกลับระหว่างแหล่งอาหารกับรัง มดจะปล่อยสารเคมีชีวภาพที่เรียกว่า ฟีโรโมน (Pheromone) ไว้บนพื้น และมดจะใช้ข้อมูลความหนาแน่นของฟีโรโมนนี้ในการสื่อสารเกี่ยวกับแหล่งอาหารของมัน จำนวนของฟีโรโมนที่ถูกปล่อยออกมานี้ จะสัมพันธ์กับปริมาณและคุณภาพของอาหาร และเป็นตัวนำทางให้มดตัวอื่นใช้เดินทางไปยังแหล่งอาหาร ซึ่งมดตัวอื่น ๆ นั้นจะเดินทางไปตามเส้นทางที่มีร่องรอยของฟีโรโมน เมื่อมดที่เดินผ่านมีจำนวนมากขึ้น ฟีโรโมนก็จะถูกปล่อยบนเส้นทางมากขึ้นด้วย เส้นทางที่มีความหนาแน่นของฟีโรโมนจำนวนมากก็จะมีแนวโน้มที่จะเป็นสูงที่มดเลือกใช้เส้นทางนั้น การวนซ้ำที่เพิ่มมากขึ้นนี้จะเป็นตัวสะท้อนช่วยให้มดหาเส้นทางที่สั้นที่สุดระหว่างรังและแหล่งอาหารของพวกเขา ซึ่งที่ผ่านมาจึงนำวิจัยที่นำเสนอการปรับปรุงอัลกอริทึมฝูงมดเพื่อเป็นการเพิ่มประสิทธิภาพการทำงาน ในบางกรณีมุ่งเน้นปรับปรุงกลไกการคัดเลือกคำตอบ บางกรณีปรับปรุงกระบวนการของฟีโรโมน และในบางกรณีปรับปรุงโครงสร้างการทำงาน [22]

อัลกอริทึมการค้นหาแบบนกคุเหว่า [23] เป็นวิธีการค้นหาที่เหมาะสมที่สุด โดยมีแรงบันดาลใจจากพฤติกรรมการวางไข่ของนกคุเหว่า ตามอุดมคติ 3 ขั้นตอนหลัก คือ ขั้นแรก นกคุเหว่า



แต่ละตัวทำการวางไข่ของมันลงในรังนกชนิดอื่นด้วยวิธีการสุ่มเลือกวาง ชั้นที่สอง รังที่มีคุณภาพสูง กล่าวคือ เป็นไข่นกดูเหว่าที่ไม่ถูกทำลายทิ้ง จะถูกเลือกให้เป็นรุ่นถัดไป และชั้นตอนสุดท้าย นักเจ้าของรัง อาจจะทำลายไข่นกดูเหว่าทิ้งบางส่วนโดยอาศัยความน่าจะเป็น และในชั้นตอนการบินหารังใหม่นั้นจะใช้หลักการของเลวีฟ্লাइट (Lévy flight) ที่มีคุณลักษณะเด่นคือ สามารถเคลื่อนที่ไปได้ไกลกว่าปกติ เพื่อมาวางไข่ใหม่แทนที่รังที่ถูกทำลายไป ซึ่งผู้คิดค้นอัลกอริทึมได้นำเสนอผลการทดสอบประสิทธิภาพที่แสดงให้เห็นว่าอัลกอริทึมนี้มีผลลัพธ์การค้นหาที่ดีกว่าอัลกอริทึมฝูงนกและอัลกอริทึมเชิงพันธุกรรม

ดังนั้น การวิจัยครั้งนี้จึงมีจุดมุ่งหมายเพื่อปรับปรุงอัลกอริทึมฝูงมดบนพื้นฐานของอัลกอริทึมฝูงมดดั้งเดิม สำหรับใช้แก้ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดสำหรับระบบอนุกรมตามข้อจำกัดที่มีหลายทางเลือกและงบประมาณ ซึ่งการปรับปรุงอัลกอริทึมฝูงมดนี้ มุ่งเน้นปรับปรุงกลไกการคัดเลือกชุดคำตอบที่มีค่าที่เหมาะสมที่สุดตามเงื่อนไข และปรับปรุงเพิ่มเติมในส่วนของอัลกอริทึมการค้นหาคำตอบจากบริเวณพื้นที่ใกล้เคียง (Local search) และกฎการปรับปรุงฟีโรโมนแบบกว้าง (Global pheromone update rule) ร่วมเข้าไปกับอัลกอริทึมการค้นหาแบบนกดูเหว่าเข้าไปในแต่ละรอบของการค้นหาคำตอบ

1.2 ความมุ่งหมายของการวิจัย

1. เพื่อปรับปรุงอัลกอริทึมฝูงมดโดยใช้อัลกอริทึมการค้นหาแบบนกดูเหว่าสำหรับระบบอนุกรม
2. เพื่อทดสอบประสิทธิภาพของอัลกอริทึมฝูงมดที่ปรับปรุงใหม่โดยใช้ฟังก์ชันทดสอบมาตรฐาน
3. เพื่อทดสอบประสิทธิภาพของอัลกอริทึมฝูงมดที่ปรับปรุงใหม่ในการแก้ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดของระบบอนุกรมที่มีข้อจำกัดหลายทางเลือกและงบประมาณ

1.3 ขอบเขตของการวิจัย

1. การทดสอบประสิทธิภาพของอัลกอริทึมโดยใช้ฟังก์ชันทดสอบการหาค่าที่เหมาะสม (Optimization Test Functions) ได้แก่
 - 1.1 ฟังก์ชัน De Jong
 - 1.2 ฟังก์ชัน Schwefel (Unimodal)
 - 1.3 ฟังก์ชัน Step
 - 1.4 ฟังก์ชัน Rosenbrock
 - 1.5 ฟังก์ชัน Ackley



- 1.6 ฟังก์ชัน Rastrigin
- 1.7 ฟังก์ชัน Griewank
- 1.8 ฟังก์ชัน Stochastic ของ Kowalik
- 1.9 ฟังก์ชัน Quartic
- 1.10 ฟังก์ชัน Michalewicz

2. การทดสอบประสิทธิภาพของอัลกอริทึมในการแก้ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดของระบบอนุกรมที่มีข้อจำกัดหลายทางเลือกและงบประมาณ โดยใช้ข้อมูลตัวอย่างจากงานวิจัยที่ผ่านมา [16-17] ประกอบด้วย

- 2.1 ระบบอนุกรมที่มี 15 ระบบย่อยและมีตัวแปรฮิวริสติกในการตัดสินใจ 60, 80 และ 100 ตัวแปร กำหนดเงื่อนไข ราคาเริ่มต้น เท่ากับ \$1000, \$900 และ \$1000 ตามลำดับ
- 2.2 ระบบอนุกรมที่มี 25 ระบบย่อยและมีตัวแปรฮิวริสติกในการตัดสินใจ 166 ตัวแปร กำหนดเงื่อนไข ราคาเริ่มต้น เท่ากับ \$1400
- 2.3 ระบบอนุกรมที่มี 40 ระบบย่อยและมีตัวแปรฮิวริสติกในการตัดสินใจ 266 ตัวแปร กำหนดเงื่อนไข ราคาเริ่มต้น เท่ากับ \$2700

1.4 การดำเนินการวิจัย

1. ศึกษาทฤษฎีของระบบอนุกรมที่มีข้อจำกัดหลายทางเลือกและงบประมาณ อัลกอริทึมฝูงมด และอัลกอริทึมการค้นหาแบบนกดูเหว่า
2. สืบค้นงานวรรณกรรมที่เกี่ยวข้อง
3. จัดเตรียมเครื่องคอมพิวเตอร์และโปรแกรมคอมพิวเตอร์สำหรับการวิจัย
4. วิเคราะห์และออกแบบอัลกอริทึมฝูงมดวิธีการใหม่โดยใช้อัลกอริทึมการค้นหาแบบนกดูเหว่า
5. พัฒนาอัลกอริทึมฝูงมดวิธีการใหม่ตามที่ออกแบบไว้
6. วิเคราะห์และประมวลผลประสิทธิภาพของอัลกอริทึมฝูงมดวิธีการใหม่ที่พัฒนาขึ้น
7. สรุปและรายงานผลการวิจัย
8. เขียนวิทยานิพนธ์
9. ส่งวิทยานิพนธ์ฉบับสมบูรณ์



1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้อัลกอริทึมผู้คุมวิธีการใหม่ เพื่อนำมาใช้หาค่าความน่าเชื่อถือที่เหมาะสมที่สุดสำหรับระบบอนุกรมตามข้อจำกัดที่มีหลายทางเลือกและงบประมาณ
2. ได้อัลกอริทึมผู้คุมประมวลผลร่วมกับอัลกอริทึมการค้นหาแบบนกดุเหว่าเพื่อใช้แก้ปัญหาการหาค่าที่เหมาะสมที่สุด
3. ได้แนวทางวิธีการสำหรับการนำอัลกอริทึมผู้คุมที่ประมวลผลร่วมกับอัลกอริทึมการค้นหาแบบนกดุเหว่า วิธีการใหม่นี้ไปประยุกต์ใช้กับปัญหาการหาค่าเหมาะสมที่สุดอื่น ๆ
4. ได้แนวทางวิธีการสำหรับการปรับปรุงอัลกอริทึมเพื่อนำไปใช้ในการออกแบบระบบสนับสนุนเพื่อการตัดสินใจในด้านอื่น ๆ
5. ได้สารสนเทศและแนวทางการประยุกต์ใช้อัลกอริทึมที่ปรับปรุงใหม่นี้ไปใช้ในงานวิศวกรรมแขนงต่าง ๆ



ปริทัศน์เอกสารข้อมูล

งานวิจัยนี้นำเสนอการปรับปรุงอัลกอริทึมผสมโดยใช้อัลกอริทึมการค้นหาแบบนกดูเหว่า สำหรับระบบอนุกรม สำหรับแก้ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดของระบบอนุกรม ตามข้อจำกัดที่มีหลายทางเลือกและงบประมาณ ผู้วิจัยได้ศึกษาปริทัศน์เอกสารข้อมูล ดังนำเสนอตามลำดับหัวข้อต่อไปนี้

2.1 เมตาฮิวริสติก (Meta-Heuristic)

ในปัจจุบันวิธีการหาค่าตอบที่เหมาะสมที่สุด (Optimization) ถูกนำมาใช้ในการแก้ไขปัญหาดังต่าง ๆ อย่างแพร่หลาย ทั้งงานวิจัยทางด้านอุตสาหกรรม วิศวกรรม รวมถึงงานวิจัยทางด้านปัญญาประดิษฐ์ ซึ่งการหาค่าตอบที่เหมาะสมที่สุดของปัญหาหนึ่ง ๆ นั้นมักจะมีข้อจำกัดต่าง ๆ มาประกอบการพิจารณาแก้ปัญหานั้นด้วย โดยเฉพาะปัญหาที่มีข้อมูลตัดสินใจขนาดใหญ่และสลับซับซ้อน มีจำนวนของตัวแปรมากขึ้น อาจจะทำให้ระยะเวลาในการคำนวณตามลำดับการคำนวณนั้นใช้เวลานาน หรืออาจจะไม่สามารถแก้ปัญหานั้นได้ ซึ่งการแก้ปัญหาดัง ๆ ที่มีความซับซ้อนหรือความยากที่เป็นแบบโพลิโนเมียล (Polynomial) กล่าวคือ เมื่อจำนวนหรือขนาดของปัญหาวัดอย่างมากขึ้น ความซับซ้อนทางเวลาในการคำนวณก็จะมากขึ้น เช่น เมื่อขนาดของปัญหาวัดอย่างเป็น n และวิธีการคำนวณใด ๆ ที่มีระยะเวลาในการคำนวณเป็น n^2 จะเรียกว่า ลำดับการคำนวณนั้นเป็นแบบโพลิโนเมียล [24] โดยที่ปัญหาใดที่ยังไม่มีวิธีการแก้ปัญหามีความซับซ้อนทางเวลา จะเรียกปัญหานั้นว่า NP-Problem (Non-Polynomial Problem) หรือปัญหาเอ็นพี สามารถจำแนกออกเป็น 3 ประเภทใหญ่ ๆ คือ NP-Complete, NP-Hard และ Strongly NP-Hard โดยความแตกต่างของปัญหาเอ็นพีทั้ง 3 ประเภทคือ เมื่อมีปัญหาคัดสินใจ (Decision Problem) ใด ๆ ที่มีวิธีการตรวจสอบคำตอบนั้นว่าเป็นคำตอบที่ดีที่สุด โดยวิธีการตรวจสอบนี้มีความซับซ้อนทางเวลาเป็นโพลิโนเมียลและวิธีการนั้นสามารถใช้ตรวจสอบปัญหาอื่น ๆ ได้ในระดับเวลาที่โพลิโนเมียล จะถูกจัดอยู่ในประเภทของปัญหา NP-Complete สำหรับปัญหา NP-Hard จะเป็นปัญหาที่ยังไม่มีวิธีการตรวจสอบคำตอบที่เป็นโพลิโนเมียลได้ แต่สามารถระบุขอบเขตของคำตอบเพื่อใช้ในการค้นหาคำตอบได้ และปัญหาใดที่แม้จะมีขอบเขตของคำตอบให้ แต่ยังไม่มีการตรวจสอบว่าคำตอบที่นั้นถูกต้องหรือไม่ ปัญหานั้นจะถูกจัดเป็นประเภท Strongly NP-Hard



อัลกอริทึมค้นหาคำตอบแบบเมตาฮีริสติก (Meta-Heuristic Algorithms) หรือวิธีการค้นหาคำตอบแบบมีเหตุมีผล จึงได้ถูกนำเสนอขึ้นเพื่อใช้ในการหาคำตอบของปัญหาต่าง ๆ ที่มีความสลับซับซ้อน และมีข้อจำกัด ภายใต้ระยะเวลาในการหาคำตอบของปัญหาต่าง ๆ ที่สมเหตุสมผล โดยนำข้อมูลมาประกอบการพิจารณาตัดสินใจเพื่อช่วยเพิ่มประสิทธิภาพการแก้ปัญหาหรือหาคำตอบที่ดีที่สุด อัลกอริทึมค้นหาคำตอบแบบเมตาฮีริสติกที่นิยมใช้แพร่หลายปัจจุบันประกอบด้วยวิธีซิมเพล็กซ์ (Simplex) วิธีการแก้ไขปัญหาซึ่งประยุกต์มาจากการแก้ไขปัญหาลักษณะอื่น ๆ เช่น วิธีเชิงพันธุกรรม (Genetic Algorithm; GA) วิธีการค้นหาต้องห้าม (Tabu Search Algorithm; TS) วิธีซิมูเลตเต็ดแอนนิลลิ่ง (Simulated Annealing Algorithm; SA) วิธีฝูงมด (Ant Colony Optimization Algorithm; ACO) วิธีฮาร์โมนีเสิร์ช (Harmony Search Algorithm; HAS) และวิธีชัฟเฟิลฟรอก ลีฟปีง (Shuffled Frog Leaping Algorithm; SFLA) เป็นต้น ดังนั้น อัลกอริทึมค้นหาคำตอบแบบเมตาฮีริสติก หมายถึง ลำดับขั้นตอนวิธีการแก้ปัญหา (Algorithm) ชุดหนึ่ง ๆ ที่สามารถนำหลักการเดียวกันไปใช้แก้ปัญหาได้ หลากหลายปัญหา ซึ่งในปัจจุบันได้รับความนิยมเป็นอย่างสูงเนื่องจากคำตอบที่ได้จากวิธีการนี้ให้ผลที่ดี แก้ปัญหาได้รวดเร็ว และใช้งานได้ง่าย โดยหลักการเบื้องต้นของเมตาฮีริสติกมีดังต่อไปนี้ [24]

1. มีระเบียบวิธีในการค้นหาคำตอบที่ดีภายในกรอบพื้นที่ของคำตอบที่อยู่ในเกณฑ์เป็นไปได้ (Feasible Region)
2. มีวัตถุประสงค์เพื่อหาผลลัพธ์ที่ดีที่สุดหรือผลลัพธ์ใกล้เคียงคำตอบที่เหมาะสมที่สุด ภายใต้ใช้เวลาน้อย
3. อัลกอริทึมอาจจะมีทั้งแบบง่ายไม่ซับซ้อน เช่น การปรับปรุงคำตอบเฉพาะที่ (Local Search) หรือแบบที่ยุ่งยากซับซ้อนมากกว่า เช่น อัลกอริทึมฝูงมด (Ant Colony Algorithm) วิธีการเชิงพันธุกรรม (Genetic Algorithm) วิธีการค้นหาต้องห้าม (Tabu Search) วิธีซิมูเลตเต็ดแอนนิลลิ่ง (Simulated Annealing) เป็นต้น
4. เป็นขั้นตอนการประมาณคำตอบ
5. อาจเกิดจากการรวมหลากหลายเทคนิค เพื่อค้นหาคำตอบที่ดีที่สุดภายในพื้นที่คำตอบที่เป็นไปได้
6. มีระเบียบขั้นตอนมาตรฐานที่แน่นอน แม้ว่าเมื่อนำไปประยุกต์ใช้ในปัญหาที่แตกต่างกัน จะมีรายละเอียดของขั้นตอนย่อยที่แตกต่างกัน แต่ฮีริสติกสำหรับปัญหาแต่ละปัญหามักต้องดำเนินการตามขั้นตอนหลักของฮีริสติกดั้งเดิม เช่น อัลกอริทึมฝูงมด (Ant Colony Algorithm) ในปัญหาที่แตกต่างกัน จะมีระเบียบหลัก ๆ ที่เหมือนกัน แต่วิธีการตัดสินใจอาจจะใช้ระดับโอบาสไนแต่ละปัญหาไม่เหมือนกัน และวิธีการเลือกคำตอบมาเพิ่มหรือลดระดับโอบาสไนแต่ละปัญหาอาจจะแตกต่างกันไปด้วย
7. สามารถใช้ได้กับปัญหาที่หลากหลาย
8. อาจมีลักษณะเป็นคำบรรยายโดยย่อก็ได้ หรือไม่จำเป็นต้องมีหลักการทางคณิตศาสตร์



9. ใช้ความจำชั่วคราวมากขึ้นในการจำคำตอบเดิม เพื่อค้นหาคำตอบที่ไม่ซ้ำเดิมหรือแตกต่างไปจากเดิม เช่น วิธีการค้นหาต้องห้าม วิธีระบบมด

จากงานวิจัยและการศึกษาในรอบปีที่ผ่านมาพบว่า อัลกอริทึมค้นหาคำตอบแบบเมตาฮิวริสติก หลายวิธีได้ถูกทำการพัฒนาคุณลักษณะต่าง ๆ เพื่อให้สามารถตอบสนองต่อระบบปัญหาที่มีความหลากหลายได้อย่างมีประสิทธิภาพมากขึ้น โดยสามารถสรุปข้อเด่นสำหรับการประยุกต์ใช้งานได้ดังต่อไปนี้ [25]

1. มีความเข้าใจได้ง่าย หรือมีวิธีการหาคำตอบแบบมีเหตุมีผลที่เหมาะสม เหมาะต่อการประยุกต์ใช้ในการแก้ปัญหาต่าง ๆ
2. มีความสอดคล้องอย่างมีเหตุมีผล ซึ่งในขั้นตอนต่าง ๆ ของอัลกอริทึมสำหรับการแก้ไขปัญหานั้น ควรยึดถือหลักการจากทฤษฎีพื้นฐานของวิธีการหาคำตอบแบบมีเหตุมีผลในแต่ละชนิดนั้น ๆ ที่สนใจพิจารณา เพื่อก่อให้เกิดประโยชน์สูงสุด
3. มีประสิทธิภาพ กล่าวคือ อัลกอริทึมที่เหมาะสมสำหรับการแก้ปัญหาต่าง ๆ นั้น จะต้องสามารถจัดหาคำตอบให้มีค่าใกล้เคียงค่าที่เหมาะสมที่สุด โดยต้องสามารถแก้ไขปัญหานั้นได้หลากหลาย และครอบคลุมตัวแปรทั้งหมดของปัญหา
4. มีประสิทธิผล โดยที่อัลกอริทึมสำหรับการแก้ปัญหาต้องจัดหาคำตอบที่ใกล้เคียงค่าที่เหมาะสมที่สุดในเวลาที่สมควร
5. มีความทนทาน ในที่นี้หมายถึงอัลกอริทึมจะต้องให้ค่าที่เหมาะสมที่สุดของปัญหาเหล่านั้นในทุก ๆ กรณีที่เกิดขึ้นหรือในสภาพจริง ยกตัวอย่างเช่น การไม่ปรับค่าบางอย่างที่ถูกกำหนดไว้ และให้ค่าคำตอบที่ตื้นน้อยลง เป็นต้น
6. นำไปใช้งานได้ง่าย อัลกอริทึมที่เหมาะสมควรมีนิยามที่ดี ง่ายต่อการเข้าใจ และมีความสำคัญใช้งานได้ง่ายหรือกล่าวอีกนัยหนึ่ง คือ ทฤษฎีไม่ควรมีเนื้อหาที่มากเกินไป แต่ควรอยู่บนพื้นฐานของความเป็นจริงและมีตัวแปรที่แสดงผลลัพธ์
7. เป็นนวัตกรรมใหม่ ทฤษฎีที่เกิดขึ้นใหม่ของอัลกอริทึมนั้น ประสิทธิภาพและประสิทธิผลที่เกิดขึ้นควรได้มาจากการปรับปรุงรูปแบบ หรือการประยุกต์ทฤษฎีอื่น ๆ เพื่อนำมาใช้ในการแก้ไขปัญหานั้น ๆ ได้อย่างเหมาะสม
8. อัลกอริทึมที่ดี ควรเป็นทฤษฎีที่ให้ผลของคำตอบที่เหมาะสมในทุก ๆ ปัญหาที่ทำการวิจัย
9. อัลกอริทึมที่ดีควรให้ผู้ใช้สามารถใช้ความรู้หรือข้อจำกัดในองค์กรมาเป็นข้อมูล หรือความต้องการในการปรับปรุงกระบวนการค้นหาคำตอบ
10. มีความหลากหลาย อัลกอริทึมควรที่จะสามารถแสดงความหลากหลายของคำตอบที่จะเกิดขึ้นใกล้ ๆ จุดที่เป็นค่าที่เหมาะสมที่สุด และให้ผู้ใช้งานเป็นผู้สรุปข้อมูลว่าควรตัดสินใจเลือก

โดยสรุปในปัจจุบันอัลกอริทึมค้นหาคำตอบแบบเมตาฮิวริสติกมีอยู่อย่างหลากหลาย แต่ละวิธีการจะมีจุดดีและจุดด้อยที่แตกต่างกัน บางอัลกอริทึมให้ผลลัพธ์ที่ดีแต่อาจใช้เวลาในการคำนวณนาน บางอัลกอริทึมประมวลผลรวดเร็วแต่ให้ผลลัพธ์ที่ด้อยกว่าวิธีการอื่น ๆ การแบ่งอัลกอริทึมค้นหาคำตอบแบบเมตาฮิวริสติกมีหลายวิธี สรุปได้ดังนี้ [24, 26]

1. แรغبันดาลใจจากธรรมชาติหรือแบบไม่ได้เกิดจากแรงบันดาลใจจากธรรมชาติ วิธีที่เกิดจากแรงบันดาลใจจากธรรมชาติ ได้แก่ วิธีการเชิงพันธุกรรม (Genetic Algorithm) วิธีระบบมด (Ant System) วิธีการลอกแบบ (Memetic Algorithm) วิธีการจำลองการอบอ่อน (Simulated Annealing) ส่วนวิธีการที่ไม่ได้เลียนแบบธรรมชาติ เช่น วิธีการค้นหาต้องห้าม (Tabu Search) การค้นหาคำตอบเฉพาะที่แบบวนรอบซ้ำ (Local Search) วิธีการค้นหาคำตอบใกล้เคียงแบบมีเงื่อนไข (Neighborhood Search) เป็นต้น

2. แบบใช้ประชากรหรือแบบไม่ใช้ประชากร (Population or Non Population Based Heuristic) แบบใช้ประชากรคือในหนึ่งรอบของการคำนวณจะมีคำตอบเพียงคำตอบเดียว วิธีแบบใช้ประชากร เช่น วิธีฝูงมด วิธีการเชิงพันธุกรรม วิธีการลอกแบบ วิธีแบบไม่ใช้ประชากร เช่น วิธีการจำลองการอบอ่อน วิธีการค้นหาต้องห้าม การค้นหาคำตอบเฉพาะที่แบบวนรอบซ้ำ วิธีการค้นหาจากคำตอบใกล้เคียงแบบมีเงื่อนไข บางครั้งเราเรียกกรณีที่ไม่ใช้ประชากรว่า วิธีการแบบทราเจกทอรี (Trajectory Method)

3. แบบสมการเป้าหมายคงที่หรือไม่คงที่ (Dynamic or Static Objective Function) กล่าวคือ ในหนึ่งรอบของการคำนวณอาจจะมีการเปลี่ยนแปลงสมการเป้าหมาย เพื่อให้ได้คำตอบใหม่ ๆ เกิดขึ้นหรือไม่มีการเปลี่ยนแปลงสมการเป้าหมาย หรือในบางกรณีที่มีการเปลี่ยนแปลงสมการเป้าหมาย เช่น การค้นหาคำตอบเฉพาะที่แบบมีการชี้นำ (Guided Local Search) กรณีที่ไม่มีการเปลี่ยนแปลงสมการเป้าหมาย เช่น วิธีฝูงมด วิธีการเชิงพันธุกรรม วิธีการลอกแบบ วิธีการจำลองการอบอ่อน วิธีการค้นหาคำตอบต้องห้าม วิธีการค้นหาคำตอบเฉพาะที่แบบวนรอบซ้ำ วิธีการค้นหาจากคำตอบใกล้เคียงแบบมีเงื่อนไข

4. แบบเปลี่ยนวิธีการหาคำตอบใกล้เคียงคำตอบปัจจุบัน (Neighborhood) คงที่และไม่คงที่ กรณีที่มีวิธีการหาคำตอบใกล้เคียงคงที่ ได้แก่ วิธีฝูงมด วิธีการเชิงพันธุกรรม วิธีการจำลองการอบอ่อน วิธีการค้นหาต้องห้าม วิธีการค้นหาคำตอบเฉพาะที่แบบวนรอบซ้ำ ส่วนกรณีที่มีการเปลี่ยนวิธีการหาคำตอบที่ใกล้เคียง ได้แก่ วิธีการค้นหาจากคำตอบใกล้เคียงแบบมีเงื่อนไข

5. แบบมีและไม่มีหน่วยความจำ ถ้ามีการใช้หน่วยความจำ จะจำว่ามีคำตอบใดบ้างที่ผ่านมาแล้วเพื่อเป็นข้อมูลในการหาคำตอบถัดไป ถ้ามีการใช้หน่วยความจำ จะจำว่ามีคำตอบใดบ้างที่ผ่านมาแล้วเพื่อเป็นข้อมูลในการหาคำตอบถัดไป ซึ่งกรณีที่ใช้หน่วยความจำที่ปรากฏชัด ได้แก่ วิธีระบบมด วิธีการค้นหาต้องห้าม ส่วนวิธีที่ไม่ใช้หน่วยความจำ เช่น วิธีการค้นหาคำตอบเฉพาะที่แบบวนรอบซ้ำ

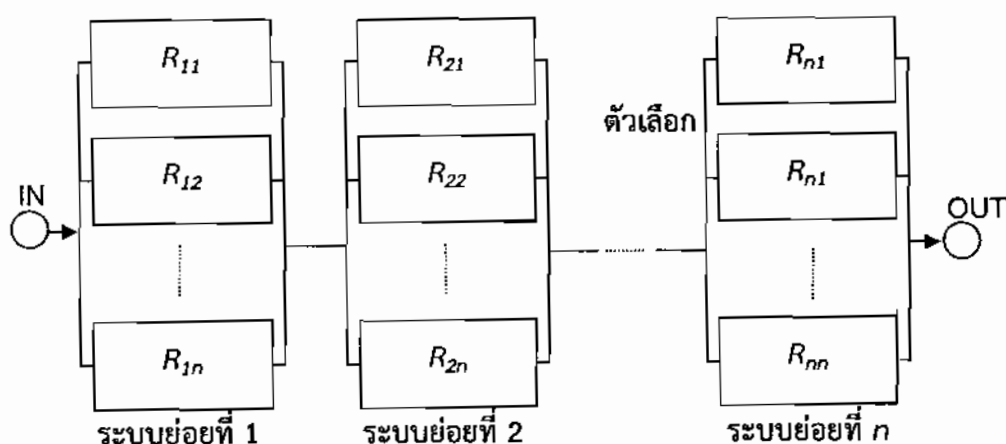
แต่ในปัจจุบันนี้จะมีออกแบบอัลกอริทึมค้นหาคำตอบแบบเมตาฮิวริสติกที่มีการผสมผสานกัน เช่น การใช้งานร่วมกันระหว่างฮิวริสติกที่ใช้หน่วยความจำและไม่ใช้หน่วยความจำ หรือมีการผสมผสาน



กันระหว่างแบบมีประชากรและไม่มีประชากร ดังนั้นการออกแบบอัลกอริทึมค้นหาคำตอบแบบเมตาฮีริสติก จึงขึ้นอยู่กับอคติของผู้ออกแบบอัลกอริทึมและอยู่บนหลักการความเป็นไปได้ ประสิทธิภาพ ประสิทธิภาพในการแก้ปัญหาทั้งในรูปแบบ NP-Complete, NP-Hard และ Strongly NP-Hard

2.2 ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมของระบบอนุกรม (Reliability Optimization Problems for a Series System)

การเพิ่มประสิทธิภาพหรือการหาค่าที่เหมาะสมของปัญหาการหาค่าความน่าเชื่อถือได้ (Reliability Optimization Problems) เทคนิคหนึ่งที่ได้รับค่านิยม คือ ระบบอนุกรม (Series System) ซึ่งมีการนำมาประยุกต์ใช้ในการออกแบบระบบใหญ่หนึ่งระบบที่มีระบบย่อย ๆ ประกอบรวมกัน [3] ยกตัวอย่างเช่น วงจรอิเล็กทรอนิกส์ 1 วงจร จะประกอบด้วยอุปกรณ์ชิ้นส่วนอิเล็กทรอนิกส์หลาย ๆ ชิ้นประกอบรวมกัน ซึ่งชิ้นส่วนอิเล็กทรอนิกส์แต่ละชิ้นจะมีคุณสมบัติที่มีความน่าเชื่อถือต่างกัน แปรผันตามราคา เมื่อคัดเลือกมาประกอบเป็นวงจรรวมจะมีผลทำให้วงจรอิเล็กทรอนิกส์นั้น ๆ มีประสิทธิภาพที่แตกต่างกัน เป็นต้น นอกจากนี้ยังสามารถนำไปประยุกต์ใช้ในการออกแบบระบบทางด้านวิศวกรรมและอุตสาหกรรม ได้แก่ ระบบการผลิต, ระบบฮาร์ดแวร์, ระบบสื่อสารโทรคมนาคม, ระบบไฟฟ้ากำลัง, ระบบสนับสนุนการตัดสินใจ (Decision Support System) ในด้านเทคโนโลยีสารสนเทศต่าง ๆ รวมถึงอุปกรณ์สิ่งอำนวยความสะดวกต่าง ๆ ภาพรวมของไดอะแกรมการกำหนดปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมของระบบอนุกรม แสดงดังภาพประกอบ 1



ภาพประกอบ 1 ไดอะแกรมระบบอนุกรมที่มีหลายทางเลือก [3]



การกำหนดปัญหาในการหาค่าที่เหมาะสม จะพิจารณาจากปัจจัยหลายด้าน เช่น น้ำหนัก ความสำคัญของระบบ ประสิทธิภาพ งบประมาณ ความน่าเชื่อถือได้ และคุณลักษณะอื่น ๆ โดยมักจะกำหนดเป็นฟังก์ชันวัตถุประสงค์ (Objective Function) และกำหนดข้อจำกัด (Constraints) เพื่อให้ได้ค่าที่เหมาะสมที่สุดตามฟังก์ชันวัตถุประสงค์ ซึ่งงานวิจัยนี้จะนำเสนอปัญหานี้เพื่อนำมาประยุกต์ใช้ และทดสอบประสิทธิภาพหลังจากการออกแบบและพัฒนาอัลกอริทึมเสร็จเรียบร้อยแล้ว คือ ต้องการหาค่าสูงสุดของความน่าเชื่อถือของระบบทั้งระบบ ดังสมการที่ 2.1 โดยกำหนดข้อจำกัดเป็นงบประมาณโดยรวมและมีทางเลือกใช้ในการตัดสินใจหลายทางเลือก ดังสมการที่ 2.2-2.4 เป็นต้น [19]

$$\text{maximize } R_{sys} = \prod_{i=1}^n \left(\sum_{j=1}^{N_i} X_{ij} R_{ij} \right) \quad (2.1)$$

$$\text{Subject to } \sum_{i=1}^n \sum_{j=1}^{N_i} X_{ij} C_{ij} \leq B \quad (2.2)$$

$$\sum_{j=1}^{N_i} X_{ij} = 1, \quad \forall i = 1, 2, \dots, n \quad (2.3)$$

$$X_{ij} = \{0,1\}, \quad \forall i = 1, 2, \dots, n \text{ และ } j = 1, 2, \dots, N_i \quad (2.4)$$

เงื่อนไขตามสมการที่ 2.2 คือ งบประมาณที่กำหนด แทนด้วย B กำหนดค่าเป็นจำนวนเต็ม; เงื่อนไขตามสมการที่ 2.3 คือ เทคโนโลยีที่มีหลายทางเลือกสำหรับระบบย่อยหนึ่ง ๆ และสมการที่ 2.4 คือ การกำหนดตัวแปรที่ใช้ตัดสินใจ และเมื่อชุดคำตอบเป็นไปตามข้อจำกัดทั้งหมดจะเรียกว่า ชุดคำตอบที่มีความเป็นไปได้ ถ้าเป็นอย่างอื่น คือ ชุดคำตอบที่เป็นไปไม่ได้

ตาราง 1 ตัวอย่างข้อมูลฮิวริสติกของปัญหา

ระบบย่อย		เทคโนโลยี 1	เทคโนโลยี 2	เทคโนโลยี 3	เทคโนโลยี 4	เทคโนโลยี 5
1	ความน่าเชื่อถือ	0.9	0.99	0.999	0.9999	0.99999
	ราคาต้นทุน (\$)	20	40	60	80	100
2	ความน่าเชื่อถือ	0.85	0.9775	0.9966	0.9995	0.9999
	ราคาต้นทุน (\$)	30	60	90	120	150
3	ความน่าเชื่อถือ	0.8	0.96	0.99	0.998	0.9997
	ราคาต้นทุน (\$)	20	40	60	80	100



ถ้าผลลัพธ์ที่ระบบย่อยเลือกใช้เทคโนโลยีเป็น

ระบบย่อยที่ 1 เลือกเทคโนโลยีที่ 2 จะได้ค่าความน่าเชื่อถือ เท่ากับ 0.99, ต้นทุน \$20

ระบบย่อยที่ 2 เลือกเทคโนโลยีที่ 1 จะได้ค่าความน่าเชื่อถือ เท่ากับ 0.85, ต้นทุน \$30

ระบบย่อยที่ 3 เลือกเทคโนโลยีที่ 3 จะได้ค่าความน่าเชื่อถือ เท่ากับ 0.99, ต้นทุน \$60

$$\text{แทนค่า } R_{\text{sys}} = \prod(0.99, 0.85, 0.99)$$

ดังนั้น จะได้ ความน่าเชื่อถือของระบบทั้งระบบ = 0.833085

จะได้ค่าต้นทุนรวม = \$20+\$30+\$60 = \$110 และเทคโนโลยีที่เลือก คือ 2-1-3

ถ้าตั้งงบประมาณ (B) ไว้ตามข้อจำกัดตามสมการที่ 2.2 ที่ \$120 ชุดคำตอบนี้จะถือว่าเป็นชุดคำตอบที่เป็นไปได้เพราะผลรวมต้นทุนยังต่ำกว่างบประมาณที่ตั้งไว้ แต่ถ้าผลรวมของต้นทุนรวมมากกว่าจะถือว่าเป็นคำตอบที่เป็นไปไม่ได้

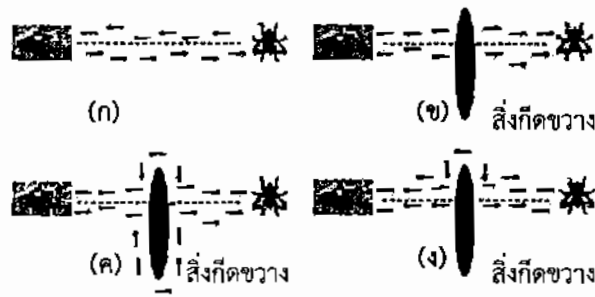
2.3 อัลกอริทึมฝูงมด (Ant Colony Algorithm)

2.3.1 พื้นฐานอัลกอริทึมฝูงมด

อัลกอริทึมฝูงมด เป็นอัลกอริทึมที่ได้แรงบันดาลใจในการออกแบบจากพฤติกรรม

การหาอาหารของมดจริง ซึ่งมดสามารถหาเส้นทางที่สั้นที่สุดจากรังมดไปยังแหล่งอาหาร ระหว่างมดเดินทางไป-กลับแหล่งอาหารกับรัง มดจะปล่อยสารเคมีชีวภาพที่เรียกว่า ฟีโรโมน (Pheromone) ไว้บนพื้น เพื่อใช้สื่อสารกับมดตัวอื่น ๆ โดยใช้ความหนาแน่นของฟีโรโมน ซึ่งหากเส้นทางใดมีฟีโรโมนจำนวนมาก มดก็จะเลือกใช้เส้นทางนั้น [26] พิจารณาตัวอย่างการเดินทางของมดซึ่งเป็นการเดินทางไป-กลับระหว่างรังมดกับแหล่งอาหาร ดังภาพประกอบ 2 (ก) เมื่อมีสิ่งกีดขวาง (Obstacle) มาปิดกั้นเส้นทางเดิม มดที่กำลังเดินทางจากรังจะไปแหล่งอาหาร หรือมดที่กำลังเดินทางจากแหล่งอาหารกลับรัง ต้องตัดสินใจว่าจะเดินไปทางด้านซ้าย หรือขวา ดังภาพประกอบ 2 (ข) เส้นทางที่มดตัดสินใจเลือกเดินทางนั้นจะมีอิทธิพลมาจากความหนาแน่นของจำนวนฟีโรโมนที่มดตัวก่อนหน้าปล่อยไว้เป็นตัวชี้แนะ แต่เนื่องจากไม่มีฟีโรโมนถูกวางไว้ก่อนหน้าทั้งสองทิศทาง มดจึงตัดสินใจเดินทางไปทั้งสองทาง ดังภาพประกอบ 2 (ค) จะเห็นว่าเส้นทางด้านบนจะสั้นกว่าเส้นทางด้านล่าง มดตัวแรกที่เดินทางตามเส้นทางด้านบนจะเดินทางข้ามสิ่งกีดขวางได้ก่อนมดตัวแรกที่เดินทางตามเส้นทางด้านล่างทำให้มดตัวต่อมาจะหาเส้นทางตามเส้นทางที่มีความหนาแน่นของฟีโรโมน เป็นเหตุให้มดส่วนใหญ่ตัดสินใจเลือกเดินทางตามเส้นทางด้านบน เนื่องจากมีความหนาแน่นของปริมาณฟีโรโมนมากกว่าและเป็นเส้นทางที่สั้นกว่า ทำให้มดมีแนวโน้มที่จะเลือกเดินทางตามเส้นทางที่มีระยะทางสั้นกว่าอย่างรวดเร็ว [28] ดังภาพประกอบ 2 (ง)





ภาพประกอบ 2 การหาระยะทางที่สั้นที่สุดของฝูงมดจริง [28]

การคิดค้นอัลกอริทึมฝูงมดในช่วงแรก มีการนำเสนออยู่ 3 เวอร์ชัน คือ Ant-density Ant-quantity และ Ant-cycle [29] แต่ในปัจจุบันเมื่อมีอ้างอิงถึงอัลกอริทึมระบบมด จะหมายถึง Ant-cycle ทั้งนี้เนื่องจากอีก 2 อัลกอริทึมนั้นมีประสิทธิภาพที่ด้อยกว่า [30] รหัสเทียมของอัลกอริทึมระบบมดดั้งเดิม แสดงดังภาพประกอบ 3 ซึ่งมีขั้นตอนการทำงานดังนี้

ขั้นตอนที่ 1 ประกอบด้วยการตั้งค่าเริ่มต้นของฟีโรโมนและพารามิเตอร์ต่าง ๆ

ขั้นตอนที่ 2 ในทุก ๆ รอบของการวนซ้ำ จะเริ่มต้นจาก มดแต่ละตัวนำค่าความหนาแน่นฟีโรโมนจากกฎการปรับปรุงฟีโรโมนแบบกว้าง (Global pheromone update rule) มาหาค่าความน่าจะเป็นโดยใช้กฎการเปลี่ยนสถานะ (State transition rule) ยกเว้นในลูบแรกเท่านั้นที่จะใช้ค่าเริ่มต้นของฟีโรโมนจากขั้นตอนที่ 1 และหลังจากนั้น มดทั้งหมดจะสร้างชุดคำตอบเพื่อใช้สำหรับแก้ปัญหาและต่อมากฎการปรับปรุงฟีโรโมนแบบกว้างจะถูกนำมาใช้ปรับปรุงความหนาแน่นของฟีโรโมนตามเส้นทางที่มดเดินผ่าน พร้อมทั้งปรับปรุงค่าฟีโรโมนของแต่ละเส้นทาง โดยการประมวลผลจะสิ้นสุดเมื่อเป็นไปตามเงื่อนไขที่กำหนด

STEP 1: Initialize the pheromone trails and parameter

STEP 2: Iteration

Repeat for each ant

Solution construction using global pheromone trails

Update the global pheromone trails

Until stopping criteria

ภาพประกอบ 3 รหัสเทียมอัลกอริทึมฝูงมดดั้งเดิม [31]



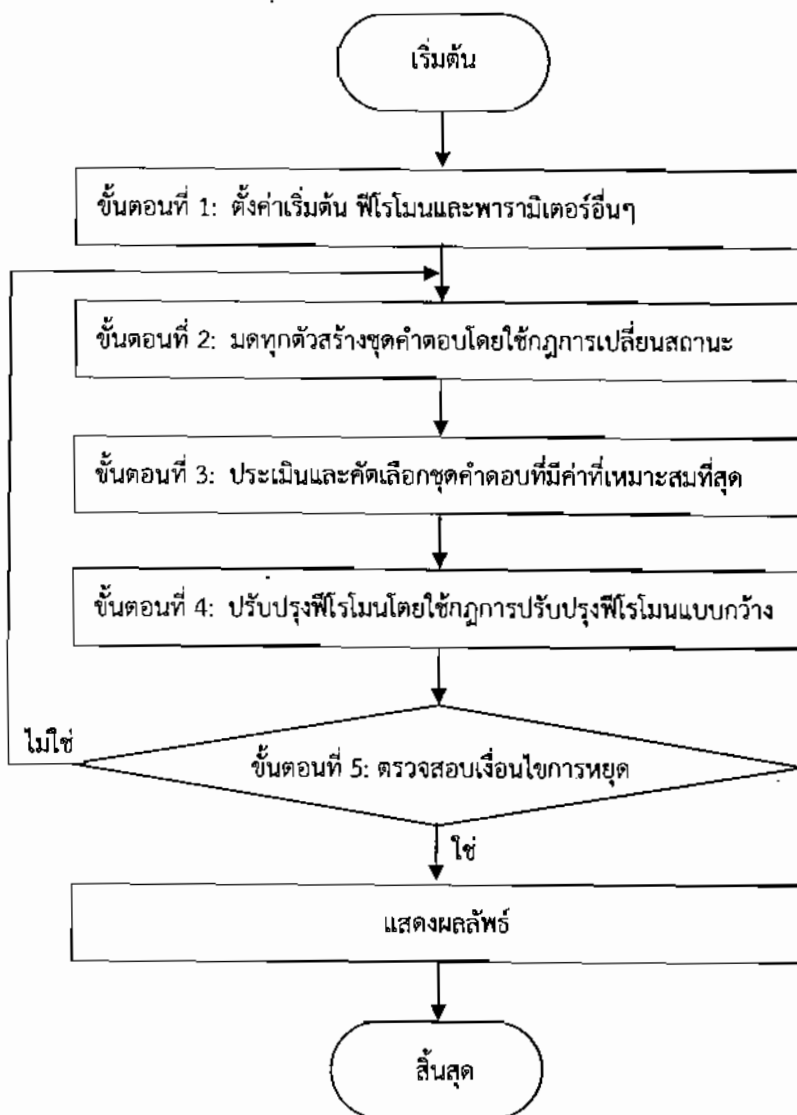
2.3.2 การประยุกต์ใช้อัลกอริทึมฝูงมดในการแก้ปัญหาเมตาฮีริสติก

อัลกอริทึมฝูงมดได้ถูกนำไปใช้แก้ปัญหาเมตาฮีริสติกสำหรับการหาค่าเหมาะที่สุดเชิงการจัด (Combinatorial Optimization Problem) [32] และในเวลาต่อมาก็ประสบความสำเร็จในการนำไปประยุกต์ใช้แก้ปัญหาการหาค่าเหมาะที่สุดเชิงการจัดอื่น ๆ ได้แก่ ปัญหาการเดินทางของพนักงานขาย (Traveling Salesman Problem: TSP) [33] ดิสครีต ออพติไมเซชัน (Discrete Optimization) [34] ปัญหาการกำหนดสมการกำลังสอง (Quadratic Assignment Problem: QAP) [35] ปัญหาการจัดเส้นทางยานพาหนะ (Vehicle Routing Problem: VRP) [36] และ ปัญหาการจัดตารางการผลิตแบบตามสั่ง (job-shop scheduling problem: JSSP) [37] เป็นต้น ต้นแบบของอัลกอริทึมฝูงมดในการหาค่าที่เหมาะสมและเป็นที่ยอมรับกันในเรื่องอัลกอริทึมระบบมด [38] และต่อมาได้มีการปรับปรุงหรือปรับแต่งอัลกอริทึมฝูงมดเพิ่มเติม ได้แก่ Elitist AS [39-40], Ant-Q [40], Ant Colony System (ACS) [41], MMAS [42], AS_rank [43], ANTS [44], BWAS [45], และ Hyper-Cube AS [46] ซึ่งการปรับปรุงอัลกอริทึมฝูงมดดังกล่าวก็เพื่อเป็นการเพิ่มประสิทธิภาพการทำงานในบางกรณีมุ่งเน้นปรับปรุงกลไกการคัดเลือกคำตอบ บางกรณีปรับปรุงกระบวนการของ ฟิวโรโมนและในบางกรณีปรับปรุงโครงสร้างการทำงาน

2.3.3 ระเบียบขั้นตอนของอัลกอริทึมฝูงมด

เพื่ออธิบายเพิ่มเติมเกี่ยวกับระเบียบขั้นตอนของอัลกอริทึมฝูงมด ผู้วิจัยนำเสนอผังงานดังภาพประกอบ 4





ภาพประกอบ 4 ผังงานอัลกอริทึมฝูงมดดั้งเดิม [47]

ขั้นตอนที่ 1 กำหนดค่าเริ่มต้น จำนวนมด ฟิโรโมนเริ่มต้น อัตราการเพิ่มของฟิโรโมน ค่าพารามิเตอร์อื่น ๆ และกำหนดให้มดฝูงแรกเริ่มต้นค้นหาคำตอบที่เป็นไปได้โดยวิธีการสุ่ม พร้อมทั้งคำนวณค่าฟังก์ชันความเหมาะสมของแต่ละคำตอบและเลือกคำตอบที่มีฟังก์ชันความเหมาะสมดีที่สุดเป็นคำตอบที่ดีที่สุดบันทึกเก็บไว้ และหาอัตราการเพิ่มของฟิโรโมนของแต่ละเส้นทาง พร้อมทั้งปรับปรุงค่าฟิโรโมนของแต่ละเส้นทาง

ขั้นตอนที่ 2 มดทุกตัวของฝูงค้นหาคำตอบที่เป็นไปได้โดยใช้ความน่าจะเป็นในการเลือกเส้นทางที่ได้จากสมการที่ 2.5 พร้อมทั้งคำนวณค่าฟังก์ชันความเหมาะสมของแต่ละคำตอบ

ขั้นตอนที่ 3 คัดเลือกคำตอบที่มีค่าฟังก์ชันความเหมาะสมที่สุดบันทึกเป็นคำตอบที่ดีที่สุด



ขั้นตอนที่ 4 หาอัตราการเพิ่มของฟีโรโมนของแต่ละเส้นทาง พร้อมทั้งปรับปรุง ค่าฟีโรโมนแบบกว้างของแต่ละเส้นทางโดยใช้สมการที่ 2.6

ขั้นตอนที่ 5 ตรวจสอบเงื่อนไขการหยุด ถ้าไม่ถึงเงื่อนไขการหยุด ให้ทำซ้ำขั้นตอนที่ 2 แต่ถ้าตรงตามเงื่อนไขการหยุดให้แสดงผลลัพธ์และหยุดทำงาน

องค์ประกอบของอัลกอริทึมฝูงมด

กฎการเปลี่ยนสถานะ (State transition rule) เป็นวิธีพฤติกรรมของมดที่จะใช้เกณฑ์ในการเลือกเส้นทางในรอบถัดไป จากสมการที่ 2.6

$$P^k(t) = \frac{[\tau(t)]^\alpha [\eta(t)]^\beta}{\sum_{k=1}^m [\tau_k(t)]^\alpha [\eta_k(t)]^\beta} \quad (2.5)$$

เมื่อ	τ	คือ ความหนาแน่นของฟีโรโมน
	η	คือ ข้อมูลฮิวริสติก
	m	คือ จำนวนมดที่ใช้ในการค้นหา
	α และ β	คือ ค่าคงที่

กฎการปรับปรุงฟีโรโมนแบบกว้าง (Global pheromone update rule)

กระบวนการค้นหาของอัลกอริทึมฝูงมด ไม่สามารถรับประกันได้ว่าคำตอบที่ได้จะเป็นไปตามเงื่อนไขที่ต้องการหรือไม่ จึงได้มีการปรับปรุงฟีโรโมน คือ ถ้าคำตอบที่ได้เป็นคำตอบที่ดีขึ้น จะให้มามีค่ามาก แต่ถ้าคำตอบที่ได้เป็นคำตอบที่ไม่ดีขึ้นจะมีค่าน้อย ซึ่งถ้าได้คำตอบที่ไม่ดีขึ้นก็มีเงื่อนไขในการปรับปรุงคำตอบให้ดีขึ้น ซึ่งการปรับปรุงความหนาแน่นของฟีโรโมน หาได้จากสมการที่ 2.6

$$\tau(t) = (1 - \rho) \cdot \tau(t-1) + \Delta\tau \quad (2.6)$$

เมื่อ ρ คือ สัมประสิทธิ์ของฟีโรโมน โดยที่การระเหยของฟีโรโมน มีค่าเท่ากับ $(1 - \rho)$ และ $\Delta\tau$ หาได้จาก

$$\Delta\tau = \Delta\tau^k \quad (2.7)$$

เมื่อ m คือ จำนวนมดทั้งหมด และ $\Delta\tau^k$ หาได้จาก



$$\Delta \tau^k = \begin{cases} 1 & \text{if } k^{\text{th}} \text{ ant chooses path} \\ 0 & \text{Otherwise} \end{cases} \quad (2.8)$$

ดังนั้น ขั้นตอนของวิธีฝูงมดสามารถสรุปได้ดังนี้ คือ กำหนดให้มดแต่ละตัวมีความสามารถในการจดจำข้อมูลเกี่ยวกับเส้นทางเพื่อใช้ในการสร้างคำตอบ ใช้ในการประเมินคำตอบไปใช้เส้นทางเดิม ใช้ในการย้อนกลับไปเส้นทางเดิม อัลกอริทึมต้องมีสถานะเริ่มต้นที่สอดคล้องกับขั้นตอนดำเนินการที่ต่อเนื่อง และมีเงื่อนไขในการหยุดที่สัมพันธ์กัน การค้นหาคำตอบจะเริ่มจากสถานะเริ่มต้น และเคลื่อนย้ายไปยังสถานะถัด ๆ ไปที่สัมพันธ์กับคำตอบของปัญหาที่จะค่อย ๆ เพิ่มสูงขึ้น การเคลื่อนย้ายหรือการค้นหาคำตอบจะเกิดขึ้นโดยการส่งผ่านฟังก์ชันของเส้นทางฟีโรโมนและค่าความสามารถ การจดจำของมดตามข้อจำกัดจากปัญหา ขณะที่มดเคลื่อนย้ายตำแหน่งจะเกิดเส้นทาง ฟีโรโมนใหม่ และถ้าหากฟีโรโมนมีความเข้มข้นคำตอบที่เป็นไปได้ก็อาจจะเปลี่ยนไปสัมพันธ์กับเส้นทาง ขั้นตอนการทำงานจะสิ้นสุดเมื่อเงื่อนไขในการหยุดถูกต้องหรือเมื่อได้คำตอบตรงกับที่ต้องการ

2.4 อัลกอริทึมการค้นหาแบบนกคูดุเหว่า (Cuckoo search algorithm)

2.4.1 พื้นฐานอัลกอริทึมการค้นหาแบบนกคูดุเหว่า

อัลกอริทึมการค้นหาแบบนกคูดุเหว่า [23] เป็นวิธีการค้นหาค่าที่เหมาะสมที่สุด โดยมีแรงบันดาลใจจากพฤติกรรมการวางไข่ของนกคูดุเหว่า ซึ่งการวางไข่ลงไปในรังจะถูกสุ่มเลือกรังว่ามีคุณภาพสูงหรือไม่ ถ้าเป็นรังที่มีคุณภาพสูงก็ทำการวางไข่ แล้วเลือกให้เป็นรุ่นถัดไปและในขั้นตอนการบินหารังใหม่นั้นจะใช้หลักการเคลื่อนที่ของเลวี (Lévy flight) ที่มีการเคลื่อนที่ไปได้ไกลกว่าปกติ โดยหลักการทั่วไปของการค้นหาแบบนกคูดุเหว่า มีกฎพื้นฐาน 3 ข้อ คือ

1. นกคูดุเหว่าแต่ละตัวสามารถวางไข่ได้เพียงครั้งละ 1 ฟอง โดยจะทำการวางไข่ของตนเองลงไปนรังที่ได้รับการสุ่มเลือกมา
2. รังที่ดีที่สุด ซึ่งมีไข่ที่มีคุณภาพสูงหรือคำตอบที่ดีกว่าจะได้รับการสืบทอดสู่รุ่นถัดไป
3. จำนวนของรังนกของเจ้าของรังนกตัวจริงจะอยู่คงที่ โดยไข่ที่ถูกวางโดยนกคูดุเหว่า จะมีโอกาสพบโดยนกเจ้าของรังซึ่งอาจจะทำการละทิ้งรังแล้วไปสร้างรังใหม่ที่ดีกว่า





ภาพประกอบ 5 พฤติกรรมการวางไข่ของนกคูดเหว่า [48-50]

รหัสเทียมของอัลกอริทึมการค้นหาแบบนกคูดเหว่า

begin

Objective function $f(x), x = (x_1, \dots, x_d)^T$;

Generate initial population of n host nests $x_i (i = 1, 2, \dots, n)$;

while ($t < \text{MaxGeneration}$) or (stop criterion);

Get a Cuckoo (soy, i) randomly by Lévy flight;

evaluate its quality/fitness F_{\min} ;

Choose a nest among n (soy j) randomly;

if ($F_{\min} > F_j$),

Replce j by the new solution;

end

Abandon a fraction (pa) of worse nests

[ond build new ones at new locations via Lévy flight];

Keep the best solutions (or nests with quality solutions);

Ronk the solutions and find the current best;

end while

Post process results and visuolization;

End

ภาพประกอบ 6 รหัสเทียมอัลกอริทึมการค้นหาแบบนกคูดเหว่า [23]

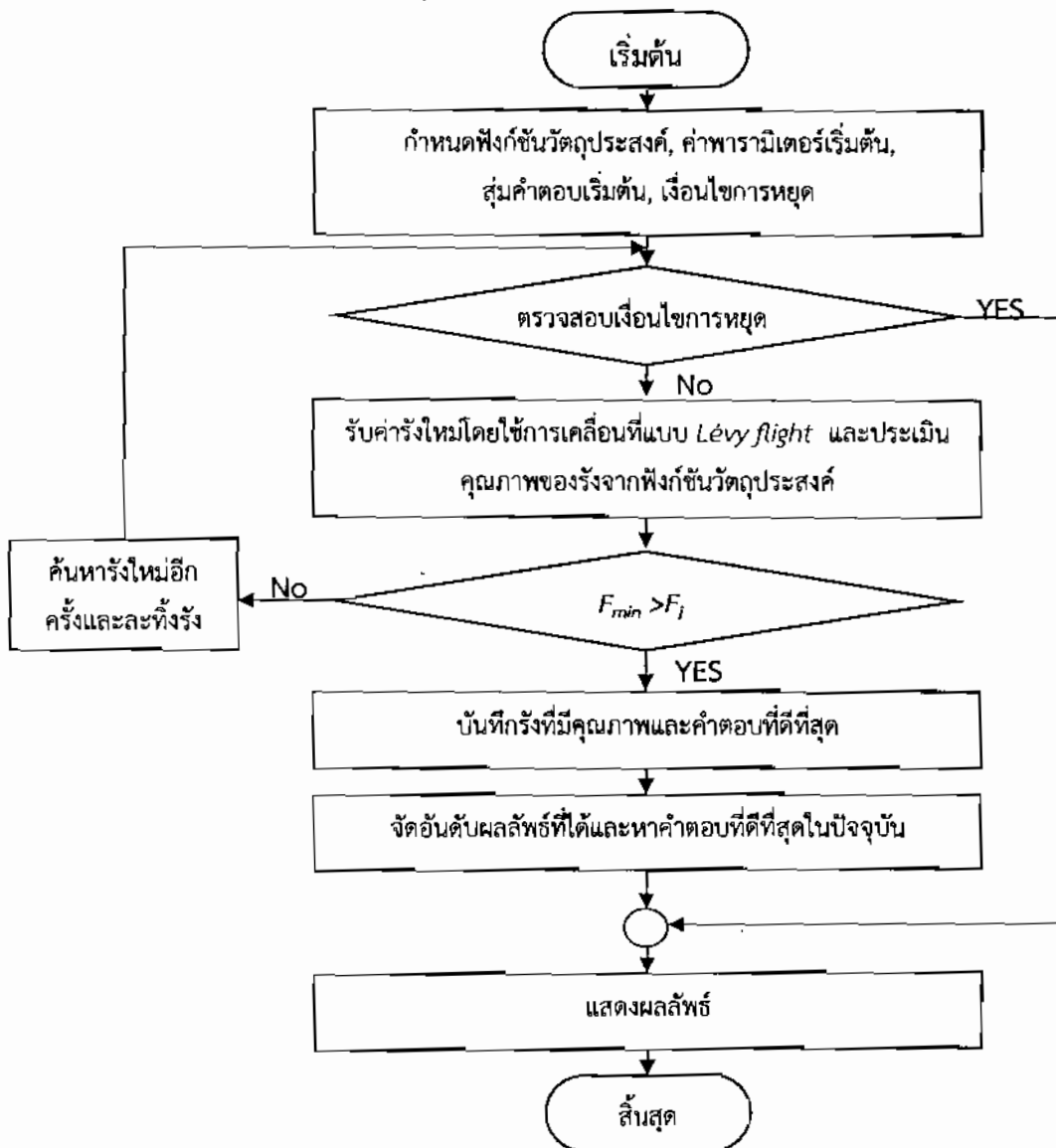


2.4.2 ระเบียบขั้นตอนของอัลกอริทึมการค้นหาแบบนกดูเหว่า

เพื่ออธิบายเพิ่มเติมเกี่ยวกับระเบียบขั้นตอนของอัลกอริทึมการค้นหาแบบนกดูเหว่า ผู้วิจัยนำเสนอผังงานดังภาพประกอบ 7 และมีขั้นตอนการทำงานดังนี้

ขั้นตอนที่ 1 กำหนดฟังก์ชันวัตถุประสงค์, ค่าความน่าจะเป็นในการค้นพบ (pa), จำนวนของรัง (n), เงื่อนไขการหยุดทำงานในที่นี้คือจำนวนรอบการค้นหา, ค่าพารามิเตอร์เริ่มต้นอื่น ๆ และสุ่มค่าตอบเริ่มต้นโดยใช้สมการที่ 2.9

$$nest_{ij}^{(0)} = \left(X_{j,\min} + (X_{j,\max} - X_{j,\min}) \cdot rand \right) \quad (2.9)$$



ภาพประกอบ 7 ผังงานอัลกอริทึมการค้นหาแบบนกดูเหว่า [51]



เมื่อ $nest_{ij}^{(0)}$ คือ รั้งที่เก็บคำตอบเริ่มต้น, ตัวแปร i แทนลำดับจำนวนของรั้ง, ตัวแปร j แทนลำดับของตัวแปรตัดสินใจ; ตัวแปร $X_{j,min}$ and $X_{j,max}$ คือ ค่าสูงสุดและต่ำสุดที่ตัวแปร j ; ฟังก์ชันสุ่ม $rand$ เป็นการสุ่มค่าระหว่าง 0 และ 1

ขั้นตอนที่ 2 ตรวจสอบเงื่อนไขการหยุด ถ้าใช่ให้ทำขั้นตอนที่ 7 แต่ถ้าไม่ใช่รับค่ารั้งใหม่โดยใช้การเคลื่อนที่แบบ *Lévy flight* ดังสมการที่ 2.10

$$nest_i^{(t+1)} = nest_i^{(t)} + \alpha \cdot S \cdot (nest_i^{(t)} - nest_{best}^{(t)}) \cdot r \quad (2.10)$$

เมื่อ $nest_i^{(t)}$ คือลำดับที่ i ของตำแหน่งรั้งในปัจจุบัน, α คือ ค่าพารามิเตอร์สำหรับปรับแต่งการเคลื่อนที่ของ *Lévy flight*; r คือ เลขสุ่มที่มีการแจกแจงปกติแบบมาตรฐาน (Standard normal distribution), $nest_{best}$ คือ รั้งที่มีคุณภาพมากที่สุดในแต่ละรอบการค้นหานี้ และ S คือ การเคลื่อนที่แบบสุ่มของ *Lévy flight* ดังสมการที่ 2.11

$$S = \frac{u}{|v|^{1/\beta}} \quad (2.11)$$

เมื่อ β คือ ค่าพารามิเตอร์ที่ใช้ควบคุมการกระจายตัวแบบ *Lévy flight* มีค่าอยู่ในช่วง $0 < \beta < 2$; u และ v คือ ค่าสัดส่วนเส้นทางการเคลื่อนที่ หาได้จากสมการที่ 2.12 และ 2.13

$$u \sim N(0, \sigma_u^2), \quad v \sim N(0, \sigma_v^2) \quad (2.12)$$

$$\sigma_u = \left\{ \frac{\Gamma(1+\beta) \sin(\pi\beta/2)}{\Gamma[(1+\beta)/2] \beta \cdot 2^{(\beta-1)/2}} \right\}^{1/\beta}, \quad \sigma_v = 1 \quad (2.13)$$



เมื่อ F เป็นฟังก์ชันแกมมาที่เป็นส่วนขยายของฟังก์ชันแฟกทอเรียลบนจำนวนเชิงซ้อน จากนั้นประเมินคุณภาพของรังทุกรังจากฟังก์ชันวัตถุประสงค์ แล้วคัดเลือกรังที่มีค่าตอบที่ดีที่สุดบันทึกค่าไว้เป็น F_{min} เสร็จแล้วเลือกรังอื่นมา 1 รังด้วยการสุ่ม แล้วดำเนินการต่อในขั้นตอนที่ 3

ขั้นตอนที่ 3 นำค่าตอบที่ดีที่สุดในรอบการค้นหานั้นซึ่งแทนด้วย F_{min} เทียบกับค่าตอบที่ดีที่สุดในรอบการค้นหาก่อนหน้านี้ซึ่งแทนด้วย F_j โดยถ้าค่า F_{min} มีค่ามากกว่า F_j ให้ทำการแทนที่รัง j เป็นค่าตอบใหม่ เมื่อเปรียบเทียบเสร็จดำเนินการต่อในขั้นตอนที่ 4

ขั้นตอนที่ 4 ค้นหารังใหม่อีกครั้ง โดยให้การค้นพบรังใหม่ในครั้งนี้มีค่าความน่าจะเป็น $p \in [0,1]$ ซึ่งได้มาจากการสุ่มค่าระหว่าง 0 และ 1 แล้วนำไปเทียบกับค่า pa ที่กำหนดไว้ ดังสมการที่ 2.14 ถ้ามีค่ามากกว่า pa ก็จะปรับให้ตำแหน่งในลำดับ p_{ij} นั้น ๆ มีค่าเป็น 1 ในทางกลับกันก็จะให้ค่าเป็น 0 และให้ละทิ้งรัง แล้วรับค่ารังใหม่แบบสุ่มดังสมการที่ 2.15

$$p_{ij} = \begin{cases} 1 & \text{if } rand < pa \\ 0 & \text{if } rand \geq pa \end{cases} \quad (2.14)$$

$$S = (nests(randp1(n,:),:) - nests(randp2(n,:),:)).rand \quad (2.15)$$

$$nest^{t+1} = nest^t + S.*K$$

เมื่อ $randp1$ และ $randp2$ คือเลขสุ่มในรูปแบบการเรียงสับเปลี่ยนลำดับ (Permutation) ให้มีความแตกต่างในแต่ละแถว แทนที่เข้าไปในรัง และ K คือ ตารางค่าความน่าจะเป็น p_{ij}

ขั้นตอนที่ 5 บันทึกรังที่มีคุณภาพและค่าตอบที่ดีที่สุด

ขั้นตอนที่ 6 จัดอันดับผลลัพธ์ที่ได้และหาค่าตอบที่ดีที่สุดในปัจจุบัน จากนั้นกลับไปทำ

ขั้นตอนที่ 2

ขั้นตอนที่ 7 แสดงผลลัพธ์

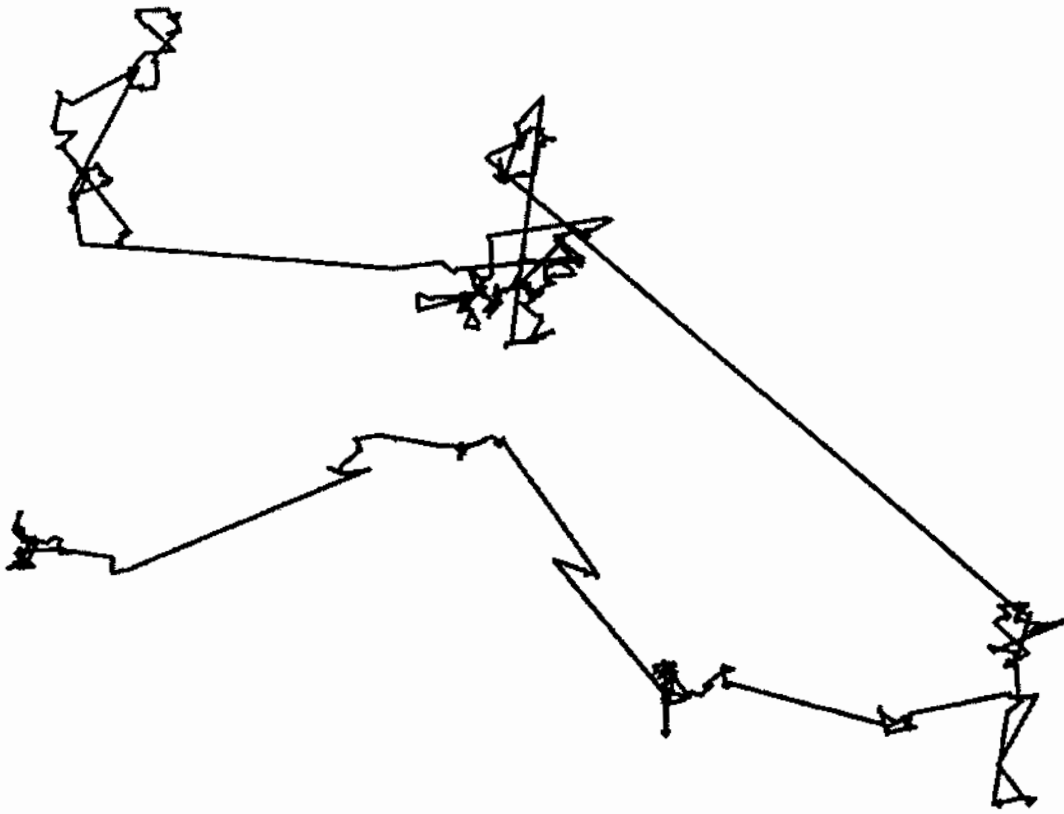
2.5 การเคลื่อนที่แบบเลวี (Lévy flight)

การเคลื่อนที่แบบเลวี (Lévy flight) ถูกนำมาเป็นส่วนหนึ่งของขั้นตอนการค้นหารังใหม่ ของอัลกอริทึมการค้นหาแบบนกคุเหว่า ซึ่งมีลักษณะของการเดินแบบสุ่ม (Random Walk) วนเวียน

ในตำแหน่งเดิม ๆ อย่างต่อเนื่องเป็นระยะ ๆ แล้วเปลี่ยนตำแหน่งในลักษณะทางตรง 90 องศา

โดยประมาณอย่างฉับพลัน ไปในตำแหน่งใหม่ที่ไกลออกไป [23] ขนาดความยาวของการก้าวกระโดดจะอาศัยการแจกแจงความน่าจะเป็นแบบหางหนัก (Heavy-tailed Probability Distribution) [52]

มาใช้ในการค้นหาคำตอบที่สูงสุดหรือต่ำสุดเฉพาะที่ คล้ายวิธีการค้นหาพื้นที่ใกล้เคียง (Local Search) หรืออาจนำมาใช้เป็นกฎการเปลี่ยนสถานะ (State Transition) ของอัลกอริทึม



ภาพประกอบ 8 ตัวอย่างการเคลื่อนที่แบบเลวี (Lévy flight) [53]

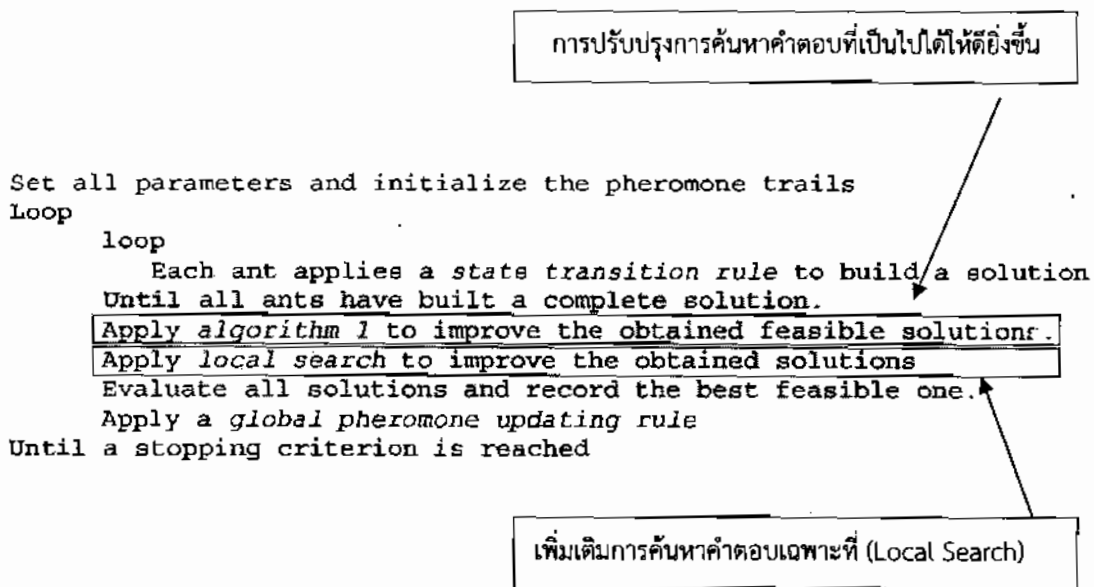
2.6 งานวิจัยที่เกี่ยวข้อง

ผู้วิจัยได้ศึกษางานวิจัยที่เกี่ยวข้องกับการปรับปรุงอัลกอริทึมฝูงมดและรูปแบบปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดของระบบอนุกรมตามข้อจำกัดที่มีหลายทางเลือก โดยสรุปงานวิจัยได้ดังนี้

2.6.1 งานวิจัยเรื่อง Quantized hopfield networks for reliability optimization [19] เป็นการนำโครงข่ายประสาทเทียมมาใช้ในการแก้ปัญหาความน่าเชื่อถือสำหรับระบบอนุกรมตามข้อจำกัดที่มีหลายทางเลือกภายใต้งบประมาณที่ตั้งไว้ ซึ่งการออกแบบโครงข่ายประสาทเทียมในการแก้ปัญหานี้ร่วมกับโครงข่าย Hopfield ผลลัพธ์การค้นหาสามารถค้นหาคำตอบที่เหมาะสมได้และรวดเร็วกว่าโครงข่าย Hopfield แบบอื่น ๆ แต่เนื่องจากเป็นจุดเริ่มต้นงานวิจัยด้านนี้จึงมีจำนวนตัวแปรเมตาฮิวริสติกในการตัดสินใจยังไม่มากนัก



2.6.2 งานวิจัยเรื่อง Ant system for reliability optimization of a series system with multiple-choice and budget constraints [16] เป็นการปรับปรุงอัลกอริทึมระบบมดเพื่อแก้ปัญหาความน่าเชื่อถือสำหรับระบบอนุกรมตามข้อจำกัดที่มีหลายทางเลือกภายใต้งบประมาณที่ตั้งไว้ โดยการปรับปรุงอัลกอริทึมของผู้วิจัยได้มีการปรับปรุงอัลกอริทึมการค้นหาคำตอบที่เป็นไปได้ให้ดียิ่งขึ้น และเพิ่มเติมการค้นหาคำตอบเฉพาะที่ (Local Search) ให้กับอัลกอริทึมการค้นหาระบบมด สรุปได้ดังภาพประกอบ 9



ภาพประกอบ 9 รหัสเทียมการปรับปรุงอัลกอริทึมฝูงมด [16]

โดยงานวิจัยนี้ได้นำเสนอชุดข้อมูลตัวอย่างในการทดสอบประสิทธิภาพของอัลกอริทึมที่มีจำนวนตัวแปรเมตาฮิวริสติกในการตัดสินใจจำนวนที่มากกว่างานวิจัยที่ใช้โครงข่ายประสาทเทียมร่วมกับโครงข่าย Hopfield และผลลัพธ์ของการค้นหาพบว่า สามารถค้นหาคำตอบที่เหมาะสมได้อย่างรวดเร็วและมีประสิทธิภาพสูง

2.6.3 งานวิจัยเรื่อง Reliability optimization of a series system with multiple-choice and budget constraints using an efficient ant colony approach [17] เป็นการปรับปรุงอัลกอริทึมฝูงมดเพื่อแก้ปัญหาความน่าเชื่อถือสำหรับระบบอนุกรมตามข้อจำกัดที่มีหลายทางเลือกภายใต้งบประมาณที่ตั้งไว้ โดยการปรับปรุงอัลกอริทึมฝูงมดของผู้วิจัยได้มีการปรับปรุงอัลกอริทึมการค้นหาคำตอบที่เป็นไปได้ให้ดียิ่งขึ้น (ดังภาพประกอบ 10 ในส่วน Algorithm 2) ได้มีการเพิ่มเติมการค้นหาคำตอบเฉพาะที่ (Local Search) และการปรับปรุงฟีโรโมนเฉพาะที่ (Local pheromone update) ซึ่งปกติอัลกอริทึมฝูงมดจะมีเฉพาะการปรับปรุงฟีโรโมนแบบกว้าง (Global



pheromone update) เท่านั้น ดังภาพประกอบ 10 นอกจากนี้ได้มีการปรับแต่งกฎการเปลี่ยนสถานะ โดยใช้ฟิชซีเซ็ดเพิ่มเติมเข้าไป

- Step 1.** The pheromone trails and the parameters are set.
- Step 2.** The following procedures are iterated Max_iter (an integer parameter) times:
- Step 2.1.** The following actions are iterated Ant_size (an integer parameter) times:
- A. A solution is constructed by repeatedly applying the transition rule.
 - B. If the solution is infeasible, it is replaced by a feasible one using *Algorithm 2*.
 - C. If it is possible, the solution is improved by *Algorithm 3*, i.e., the *local search* procedure.
 - D. The pheromone trails related to the chosen technologies are finally modified according to the *local updating* rule.
- Step 2.2.** The pheromone trails are modified according to the global updating rule.

ภาพประกอบ 10 รหัสเทียมการปรับปรุงอัลกอริทึมฝูงมดร่วมกับฟิชซีเซ็ด [17]

โดยงานวิจัยนี้ได้นำเสนอชุดข้อมูลตัวอย่างที่มีจำนวนตัวแปรเมตาฮิวริสติกในการตัดสินใจเพิ่มเติมจากการทดสอบประสิทธิภาพของการปรับปรุงอัลกอริทึมระบบมด รวมถึงได้นำเสนอแบบแผนการทดลองการปรับแต่งฟิชซีเซ็ดที่แตกต่างกัน 6 กรณีศึกษา และได้มีการเปรียบเทียบผลการทดลอง ซึ่งผลลัพธ์ของการค้นหาส่วนใหญ่พบว่าสามารถค้นหาคำตอบที่เหมาะสมได้อย่างรวดเร็วและมีประสิทธิภาพสูงกว่างานวิจัยการปรับปรุงอัลกอริทึมระบบมดที่ผ่านมา

2.6.4 งานวิจัยเรื่อง Reliability optimization of topology communication network design using an improved ant colony optimization [47] การนำเสนองานวิจัยที่เกี่ยวข้องนี้จะได้นำเสนอในประเด็นด้านการปรับปรุงอัลกอริทึมฝูงมดเพื่อแก้ปัญหาความน่าเชื่อถือ โดยมีจุดเด่นของการปรับปรุงคือการใช้เทคนิคการค้นหาคำตอบจากบริเวณพื้นที่ใกล้เคียง (Neighborhood Search) ซึ่งโดยส่วนใหญ่จะพบว่าจะใช้การค้นหาคำตอบเฉพาะที่ (Local Search) เพื่อให้ปรับปรุงอัลกอริทึมให้มีประสิทธิภาพสูงขึ้น โดยนอกจากนี้ยังพบว่ามีการปรับการตั้งค่ากลับไปเป็นค่าเริ่มต้นของฟีโรโมนใหม่เมื่อพบว่าการค้นหาคำตอบอาจอยู่ในสถานะของคำตอบเฉพาะที่ (Local optimal) นานเกินไป เพื่อให้มีโอกาสค้นพบคำตอบในระดับกว้าง (Global optimal) ขึ้นไป ทั้งนี้การออกแบบ



และปรับปรุงอัลกอริทึมฝูงมดนี้ ได้ผ่านทดสอบกับฟังก์ชันการทดสอบมาตรฐาน ได้แก่ ฟังก์ชัน Schaffer ฟังก์ชัน Sphere ฟังก์ชัน Rosenbrock ฟังก์ชัน Griewank และฟังก์ชัน Rastrigin ซึ่งผลการทดสอบแสดงให้เห็นว่าสามารถหาคำตอบที่เหมาะสมได้

2.6.5 งานวิจัยเรื่อง Energy-aware scheduling using Hybrid Algorithm for cloud computing [54] การนำเสนองานวิจัยที่เกี่ยวข้องนี้จะได้นำเสนอในประเด็นด้านการปรับปรุงอัลกอริทึมฝูงมดร่วมกับอัลกอริทึมการค้นหาแบบนกคุเหว่า ซึ่งมีการปรับปรุงโดยการใช้อัลกอริทึมการค้นหาแบบนกคุเหว่าแทนที่การค้นหาคำตอบเฉพาะที่ (Local Search) ทั้งนี้งานวิจัยดังกล่าวยังไม่ได้แสดงให้เห็นถึงฟังก์ชันการทดสอบมาตรฐานทั่วไป แต่สามารถประยุกต์ใช้แก้ปัญหาที่นำเสนอได้

จากการศึกษาปริทัศน์เอกสารข้อมูลและงานวิจัยที่เกี่ยวข้องเพื่อเป็นแนวทางในการนำเสนอการปรับปรุงอัลกอริทึมฝูงมดโดยใช้อัลกอริทึมการค้นหาแบบนกคุเหว่าสำหรับระบบอนุกรม สำหรับแก้ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดของระบบอนุกรมตามข้อจำกัดที่มีหลายทางเลือกและงบประมาณ ผู้วิจัยจะได้นำเสนอขั้นตอนวิธีต่าง ๆ ในบทที่ 3 ต่อไป



บทที่ 3

วิธีดำเนินการวิจัย

ขั้นตอนการออกแบบและการปรับปรุงอัลกอริทึมผสมเพื่อใช้แก้ปัญหาที่ต้องการค่าความน่าเชื่อถือที่เหมาะสมที่สุดสำหรับระบบอนุกรมตามข้อจำกัดที่มีหลายทางเลือกและงบประมาณ มีดังนี้

3.1 ฟังก์ชันวัตถุประสงค์

พิจารณาระบบอนุกรมหนึ่งระบบที่มี n ระบบย่อย โดยแต่ละระบบย่อยจะต้องคัดเลือกเทคโนโลยีซึ่งมีหลายทางเลือกมาเพียงหนึ่งเทคโนโลยี แล้วนำมาประกอบรวมกันให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุดตามการจำกัดด้วยเงินงบประมาณที่ตั้งไว้ ซึ่งเทคโนโลยีที่ระบบย่อยคัดเลือคนั้นจะมีค่าความน่าเชื่อถือที่แปรตามราคา สามารถกำหนดปัญหาเป็นสมการทางคณิตศาสตร์ และมีสัญลักษณ์บางตัวที่จะนำเสนอให้ทราบก่อนดังนี้

- n แทน จำนวนระบบย่อย
- N_i แทน จำนวนเทคโนโลยีที่มีหลายทางเลือกสำหรับระบบย่อย i
- C_{ij} แทน ราคาต้นทุนของเทคโนโลยี j ที่ระบบย่อย i เลือกใช้
- R_{ij} แทน ค่าความน่าเชื่อถือของเทคโนโลยี j ที่ระบบย่อย i เลือกใช้
- R_{sys} แทน ค่าความน่าเชื่อถือของระบบอนุกรมทั้งระบบ
- B แทน งบประมาณที่ตั้งไว้

กำหนดตัวแปรที่ใช้ตัดสินใจ X_{ij} ($i = 1, 2, \dots, n$ และ $j = 1, 2, \dots, N_i$) ดังนี้:

$$X_{ij} = \begin{cases} 1 & \text{ถ้าระบบย่อย } i \text{ เลือกใช้เทคโนโลยี } j \\ 0 & \text{ถ้าเป็นอย่างอื่น} \end{cases}$$

กำหนดฟังก์ชันวัตถุประสงค์ในแก้ปัญหาการหาค่าเหมาะสมที่สุดของความน่าเชื่อถือของระบบอนุกรมเป็นแบบจำนวนเต็มไบนารีด้วยฟังก์ชันวัตถุประสงค์ไม่เชิงเส้น (Binary integer programming problem with nonlinear objective function) ได้ดังสมการต่อไปนี้



$$\text{maximize } R_{\text{sys}} = \prod_{i=1}^n \left(\sum_{j=1}^{N_i} X_{ij} R_{ij} \right) \quad (3.1)$$

$$\text{Subject to } \sum_{i=1}^n \sum_{j=1}^{N_i} X_{ij} C_{ij} \leq B \quad (3.2)$$

$$\sum_{j=1}^{N_i} X_{ij} = 1, \quad \forall i = 1, 2, \dots, n \quad (3.3)$$

$$X_{ij} = \{0,1\}, \quad \forall i = 1, 2, \dots, n \text{ และ } j = 1, 2, \dots, N_i \quad (3.4)$$

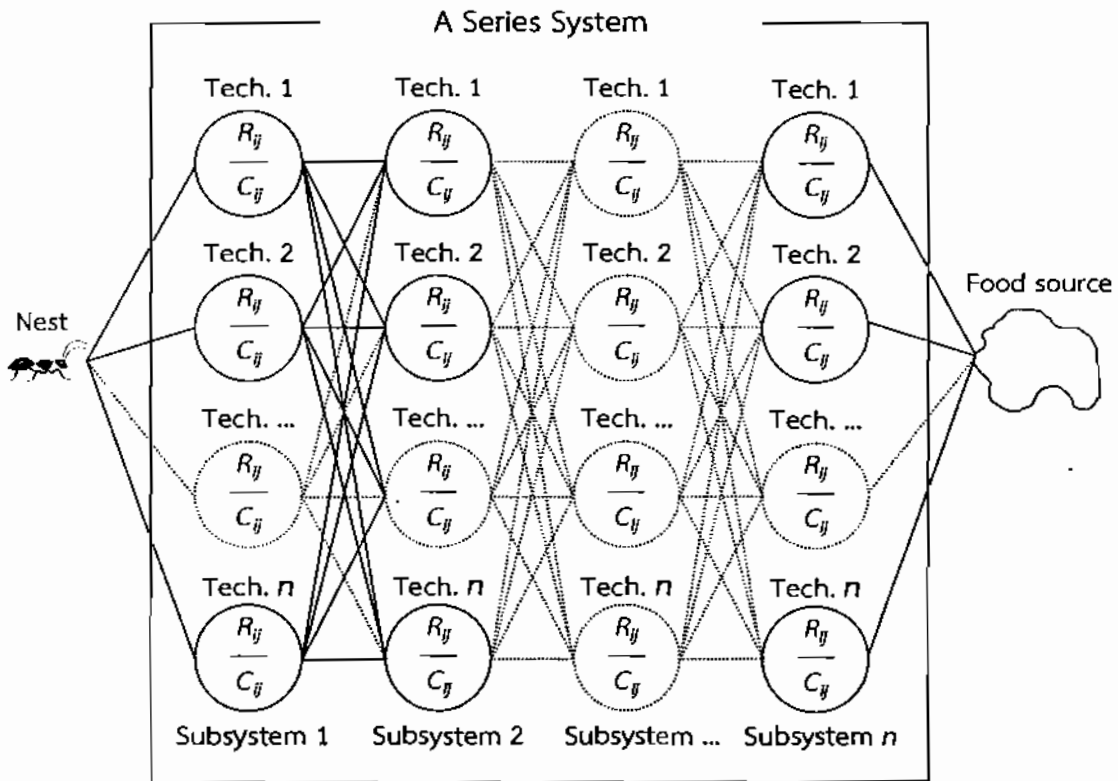
จากสมการที่ 3.2 คือ เงื่อนไขตามงบประมาณที่กำหนด แทนด้วย B กำหนดค่าเป็นจำนวนเต็ม เงื่อนไขตามสมการที่ 3.3 คือ เทคโนโลยีที่มีหลายทางเลือกสำหรับระบบย่อยหนึ่ง ๆ และสมการ 3.4 คือ การกำหนดตัวแปรที่ใช้ตัดสินใจ และเมื่อชุดคำตอบเป็นไปตามข้อจำกัดทั้งหมดจะเรียกว่า ชุดคำตอบที่มีความเป็นไปได้ ถ้าเป็นอย่างอื่น คือ ชุดคำตอบที่เป็นไปไม่ได้

3.2 การปรับปรุงอัลกอริทึมฟุ่มมด

3.2.1 ภาพรวมการออกแบบและปรับปรุงอัลกอริทึมสำหรับระบบอนุกรม

การประยุกต์ใช้อัลกอริทึมกับการแก้ปัญหาการหาค่าเหมาะที่สุดเชิงการจัด (Combinatorial Optimization Problem) ลักษณะของปัญหาจะแทนด้วย เซตของ $G = (N, E)$ เมื่อ N คือ เซตของ โหนดที่มีอยู่ในระบบ และ E คือ กลุ่ม เซตของ node ที่มีด้านที่เชื่อมต่อกัน ซึ่งเมื่อนำมาประยุกต์เข้ากับปัญหานี้ เซตของโหนดจะแทนด้วยระบบย่อย (Subsystem) และเซตของ E จะแทนด้วยกลุ่มเซตของ เทคโนโลยีที่มีระบบย่อยสามารถคัดเลือกนำไปใช้ได้ โดยเขียนเป็นแบบจำลองเส้นทางระหว่างรังมดกับ แหล่งอาหาร แสดงดังภาพประกอบ 11





ภาพประกอบ 11 แบบจำลองเส้นทางระหว่างรังมดกับแหล่งอาหาร

3.2.2 การปรับปรุงอัลกอริทึมฝูงมดโดยใช้วิธีการค้นหาพื้นที่ใกล้เคียง (ACO2SWAP)

ขั้นตอนวิธีการทำงานของอัลกอริทึมฝูงมดที่ปรับปรุงใหม่ ตามภาพประกอบ 12 มีรายละเอียดดังนี้

ขั้นตอนที่ 1 ตั้งค่าเริ่มต้น ได้แก่ พีโรโมนเริ่มต้น, อัตราการระเหยของพีโรโมน, ปริภูมิการค้นหาเริ่มต้น, จำนวนมดที่ใช้ค้นหา, จำนวนรอบที่ใช้ค้นหาสูงสุด และพารามิเตอร์ที่จำเป็นอื่น ๆ

ขั้นตอนที่ 2 มดทุกตัวสร้างชุดคำตอบโดยใช้กฎการเปลี่ยนสถานะ (State transition rule) ซึ่งมดทุกตัวในฝูงจะสร้างชุดคำตอบที่เป็นไปได้โดยอาศัยความน่าจะเป็นในการเลือกเส้นทางตามสมการที่ 3.5 หรือในที่นี้ คือ มดแต่ละตัวจะทำการสุ่มเลือกเทคโนโลยีที่เป็นไปได้ให้กับระบบย่อย

$$P_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{m=1}^{N_j} [\tau_{im}(t)]^\alpha [\eta_{im}]^\beta} \quad (3.5)$$



เมื่อ τ คือ ความหนาแน่นของพีโรโมน

α, β คือ ค่าคงที่ใช้ปรับค่า

m คือ จำนวนของมดที่ใช้ในการค้นหา

η_{ij} คือ ข้อมูลอิวิริสติก ระหว่างระบบย่อย แทนด้วย i และเทคโนโลยี

แทนด้วย j ตามลำดับ ซึ่งการหาค่าข้อมูลอิวิริสติกสำหรับ ปัญหา นี้ จะใช้สมการที่ 3.6

$$\eta_{ij} = \frac{R_{ij}}{C_{ij}} \quad (3.6)$$

เมื่อ R_{ij} และ C_{ij} แทนด้วย ความน่าเชื่อถือและราคาต้นทุน ตามลำดับ

ขั้นตอนที่ 3 อัลกอริทึมคัดเลือกชุดคำตอบที่ดีที่สุดภายใต้ข้อจำกัด หลังจากมดทุกตัวในฝูงสร้างชุดคำตอบเสร็จเรียบร้อยแล้ว อัลกอริทึมนี้จะทำหน้าที่ประเมินคำตอบที่ได้และคัดเลือกเฉพาะชุดคำตอบที่มีผลลัพธ์ของราคาต้นทุนไม่เกินที่ตั้งไว้ โดยสรุปขั้นตอนวิธีย่อยได้ดังนี้

ขั้นตอนย่อยที่ 3.1: ประเมินคำตอบที่ได้ทั้งค่าความน่าเชื่อถือและราคาต้นทุน

ขั้นตอนย่อยที่ 3.2: คัดเลือกเฉพาะชุดคำตอบที่มีความเป็นไปได้

ขั้นตอนย่อยที่ 3.3: จัดเรียงชุดคำตอบที่มีค่าความน่าเชื่อถือมากที่สุดไปหาน้อยสุด

ขั้นตอนย่อยที่ 3.4: คัดเลือกชุดคำตอบที่มีค่าความน่าเชื่อถือสูงสุดจัดเก็บใน

หน่วยความจำ

ขั้นตอนที่ 4 ปรับปรุงผลลัพธ์ด้วยการค้นหาคำตอบจากบริเวณพื้นที่ใกล้เคียง

(Neighborhood Search) เป็นวิธีการเปลี่ยนสลับคู่ (SWAP) ของเทคโนโลยีที่ได้รับการเลือกแล้วโดยใช้คู่อื่น ๆ ซึ่งในแต่ละระบบย่อย เทคโนโลยีจะเป็นดัชนีที่เรียงลำดับตามค่าความน่าเชื่อถือ เช่น ชุดคำตอบ $S = \{a, b, c, \dots\}$ จะหมายถึง ค่าดัชนี a จะแทนด้วยเทคโนโลยีที่ระบบย่อยที่ 1 เลือกใช้, ค่าดัชนี b จะแทนด้วยเทคโนโลยีที่ระบบย่อยที่ 2 เลือกใช้ และค่าดัชนี c จะแทนด้วยเทคโนโลยีที่ระบบย่อยที่ 3 เลือกใช้ เป็นต้น เมื่อพิจารณาจากตัวอย่างระบบอนุกรมที่ประกอบด้วย 3 ระบบย่อย และมี 8 เทคโนโลยีที่สามารถเลือกใช้ได้ โดยสมมติว่าการแก้ปัญหาที่ได้ในรอบนี้ คือ $S = \{4, 1, 6\}$ การค้นหาในบริเวณพื้นที่ใกล้เคียงจะมีการประเมินชุดคำตอบ ดังต่อไปนี้

$$S = \{3, 1, 6\}; S = \{5, 1, 6\}; S = \{4, 1, 6\};$$

$$S = \{4, 2, 6\}; S = \{4, 1, 7\}; S = \{4, 1, 5\};$$

*กำหนดให้มีค่าคงเดิมเนื่องจากอาจเป็นค่าสูงสุดหรือต่ำสุดของเทคโนโลยีที่สามารถเลือกใช้ได้



ในทุกชุดคำตอบจะได้รับการประเมินผลลัพธ์ทั้งค่าความน่าเชื่อถือและราคาต้นทุน โดยเมื่อพบว่ามีชุดคำตอบที่มีความเป็นไปได้ หรือในที่นี้คือ มีชุดคำตอบที่มีค่าความน่าเชื่อถือสูงกว่าชุดคำตอบเดิม ชุดคำตอบใหม่นี้จะถูกแทนที่ชุดคำตอบเดิม แต่ถ้าไม่น้อยกว่าจะยังคงใช้ชุดคำตอบเดิมจากขั้นตอนที่ 3



ภาพประกอบ 12 ผังงานอัลกอริทึมฝูงมดที่ปรับปรุงใหม่โดยใช้วิธีการค้นหาพื้นที่ใกล้เคียง



ขั้นตอนที่ 5 ปรับปรุงฟีโรโมนโดยใช้กฎการปรับปรุงฟีโรโมนแบบกว้าง (Global pheromone update rule) โดยในแต่ละรอบของการค้นหา ชุดคำตอบที่ได้นั้นจะยังไม่สามารถรับประกันได้ว่าผลลัพธ์ที่ได้มีค่าเหมาะสมที่สุดเป็นไปตามเงื่อนไขที่ต้องการหรือยัง จึงได้มีการปรับปรุงฟีโรโมนแบบกว้าง กล่าวคือ ถ้าชุดคำตอบที่ได้มีผลลัพธ์ที่ดีขึ้น ก็จะปรับปรุงความหนาแน่นของฟีโรโมนให้มีความมากขึ้น แต่ถ้าชุดคำตอบที่ได้มีผลลัพธ์ไม่ดีขึ้นก็จะปรับปรุงความหนาแน่นของฟีโรโมนเพียงเล็กน้อย ซึ่งการปรับปรุงความหนาแน่นของฟีโรโมน หาได้จาก

$$\tau_{ij}(t) = (1 - \rho) \cdot \tau_{ij}(t-1) + \Delta\tau_{ij} \quad (3.7)$$

เมื่อ $\Delta\tau_{ij}$ หาได้จาก

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k \quad (3.8)$$

เมื่อ m คือ จำนวนมดทั้งหมด และ $\Delta\tau_{ij}^k$ หาได้จาก

$$\Delta\tau_{ij}^k = \begin{cases} 1 & \text{if } k^{\text{th}} \text{ ant chases path} \\ 0 & \text{Otherwise} \end{cases} \quad (3.9)$$

ขั้นตอนที่ 6 ปรับปรุงผลลัพธ์ด้วยการค้นหาคำตอบจากบริเวณพื้นที่ใกล้เคียงอีกครั้ง โดยเมื่อพบว่า ชุดคำตอบที่มีความเป็นไปได้ดีกว่าคำตอบเดิม ชุดคำตอบใหม่นี้จะถูกแทนที่ชุดคำตอบเดิม แต่ถ้ามันน้อยกว่าจะยังคงใช้ชุดคำตอบเดิมจากขั้นตอนที่ 4

ขั้นตอนที่ 7 ปรับปรุงฟีโรโมนโดยใช้กฎการปรับปรุงฟีโรโมนแบบกว้างอีกครั้ง ทั้งนี้เพื่อให้ความหนาแน่นของฟีโรโมนมีความมากขึ้น ถ้าหากคำตอบในขั้นตอนที่ 6 มีค่าดีขึ้นกว่าเดิม

ขั้นตอนที่ 8 หยุดทำงาน เมื่อเงื่อนไขเป็นจริง ในที่นี้คือ จำนวนรอบในการค้นหาสูงสุดตามที่กำหนดไว้ หากยังไม่เป็นจริงจะย้อนกลับไปยังขั้นตอนที่ 2

ขั้นตอนที่ 9 แสดงผลลัพธ์ ได้แก่ ชุดคำตอบที่มีค่าความน่าเชื่อถือเหมาะสมที่สุด, ค่าความน่าเชื่อถือเหมาะสมที่สุด และราคาต้นทุนรวม

3.2.3 การปรับปรุงอัลกอริทึมฝูงมดโดยใช้เลวีฟ্লাइट (ACO-Lévy flight)

การออกแบบจะใช้หลักการเลวีฟ্লাइट (Lévy flight) แทนที่การค้นหาผลลัพธ์ที่ดีขึ้นจากพื้นที่ใกล้เคียง ตามที่ได้นำเสนอในหัวข้อ 3.2.2 (ขั้นตอนที่ 4-6) ซึ่งการเคลื่อนที่แบบสุ่มของเลวีฟ্লাइट



ตามสมการที่ 3.10 - 3.13 จะเป็นสมูหาค่าชุดคำตอบ (Solution) ชุดใหม่จากคัดเลือกชุดคำตอบที่มีค่าความน่าเชื่อถือที่ดีที่สุดภายใต้ข้อจำกัดก่อนหน้า

$$S_i^{(t+1)} = S_i^{(t)} + \alpha \cdot \text{Step} \cdot (S_i^{(t)} - S_{\text{best}}) \cdot r \quad (3.10)$$

เมื่อ $S_i^{(t)}$ คือลำดับที่ i ของชุดคำตอบในปัจจุบัน, α คือ ค่าพารามิเตอร์สำหรับปรับแต่งการเคลื่อนที่ของเลวีไฟลท์; r คือ เลขสุ่มที่มีการแจกแจงปกติแบบมาตรฐาน (Standard normal distribution), S_{best} คือ ชุดคำตอบที่มีค่าดีที่สุดของรอบการค้นหานั้น ๆ และตัวแปร Step คือ การเคลื่อนที่แบบสุ่มของเลวีไฟลท์ ดังสมการที่ 3.11

$$\text{Step} = \frac{u}{|v|^{1/\beta}} \quad (3.11)$$

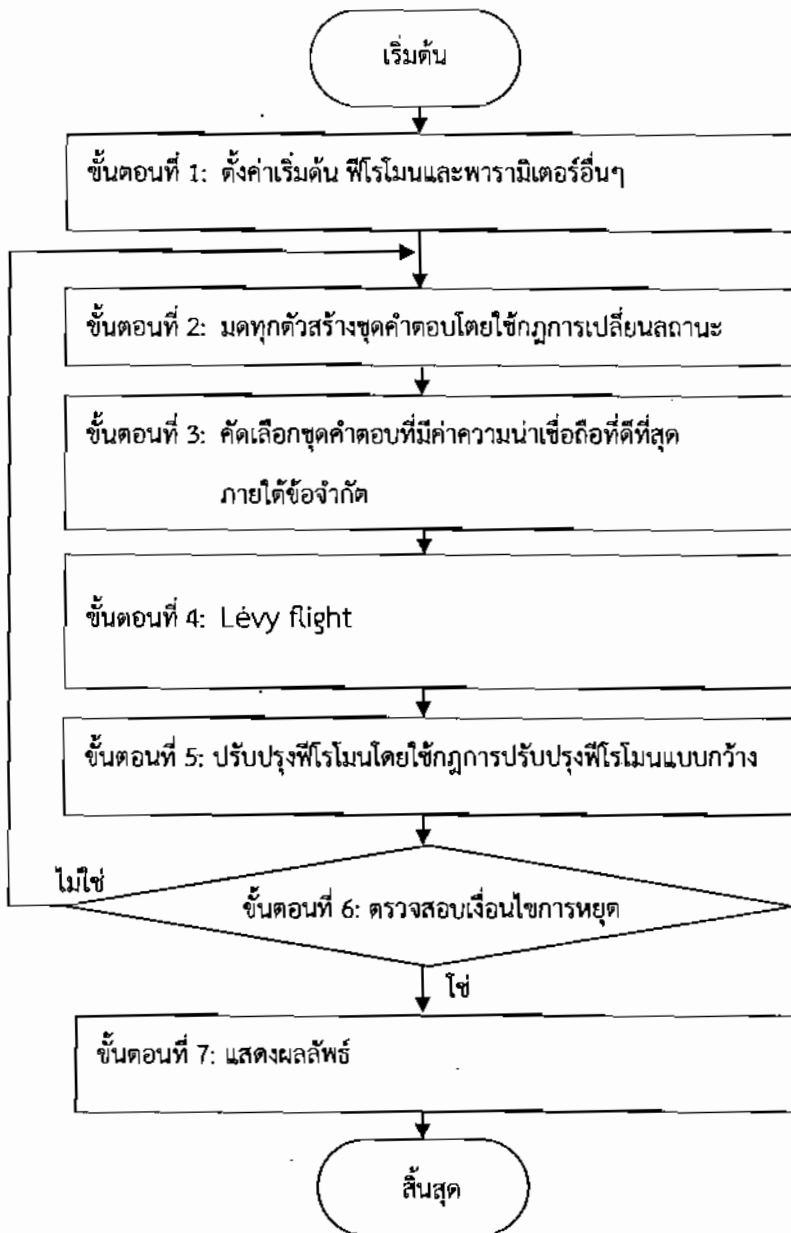
เมื่อ β คือ ค่าพารามิเตอร์ที่ใช้ควบคุมการกระจายตัวแบบเลวีไฟลท์ มีค่าอยู่ในช่วง $0 < \beta < 2$; u และ v คือ ค่าสัดส่วนเส้นทางการเคลื่อนที่ หาได้จากสมการที่ 3.12 และ 3.13

$$u \sim N(0, \sigma_u^2), \quad v \sim N(0, \sigma_v^2) \quad (3.12)$$

$$\sigma_u = \left\{ \frac{\Gamma(1 + \beta) \sin(\pi\beta/2)}{\Gamma[(1 + \beta)/2] \beta \cdot 2^{(\beta-1)/2}} \right\}^{1/\beta}, \quad \sigma_v = 1 \quad (3.13)$$

เมื่อ Γ เป็นฟังก์ชันแกมมาที่เป็นส่วนขยายของฟังก์ชัน แฟกทอเรียลบนจำนวนเชิงซ้อน จากนั้นประเมินคุณภาพของรังทุครั้งจากฟังก์ชันวัตถุประสงค์ แล้วคัดเลือกรังที่มีค่าตอบที่ดีที่สุดบันทึกค่าไว้ ดำเนินการต่อในขั้นต่อไป การออกแบบแสดงในผังงานอัลกอริทึมฝูงมดที่ปรับปรุงใหม่โดยใช้หลักการเลวีไฟลท์ ตามภาพประกอบ 13



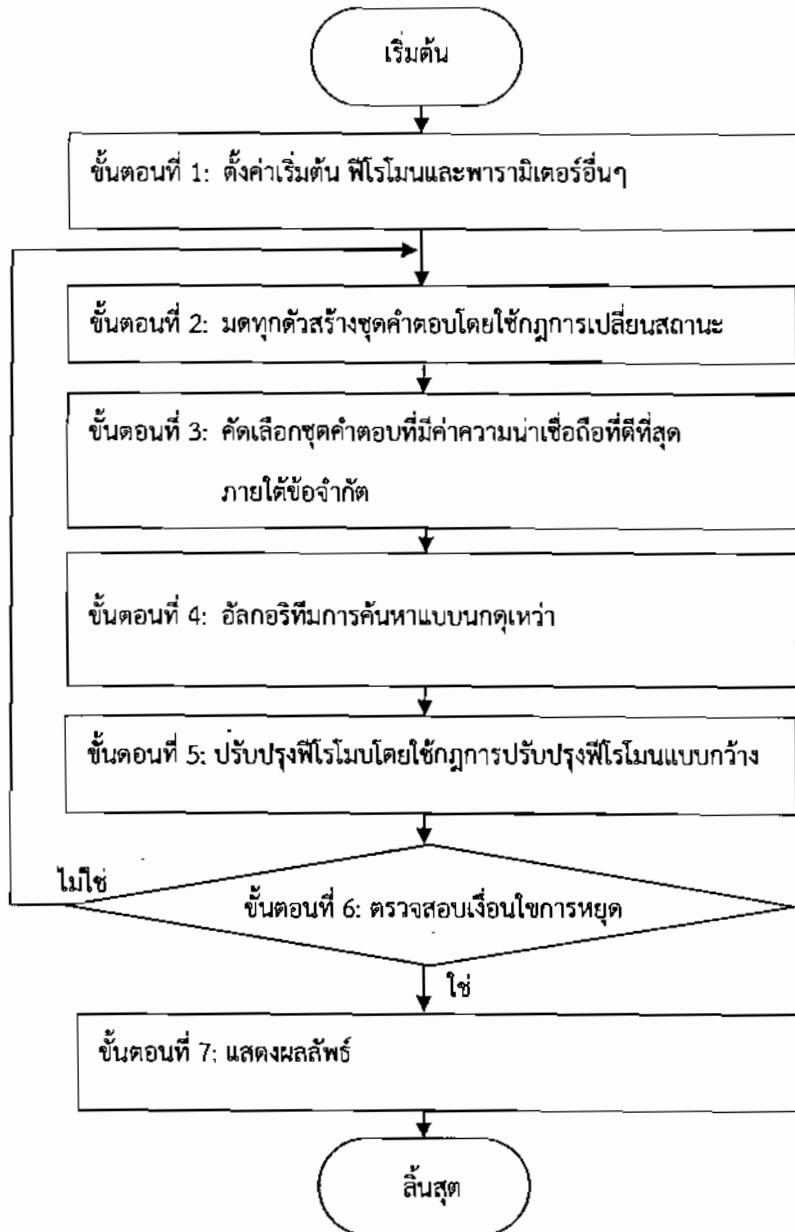


ภาพประกอบ 13 ผังงานอัลกอริทึมฝูงมดที่ปรับปรุงใหม่โดยใช้หลักการเลวีฟลายท์

3.2.4 การปรับปรุงอัลกอริทึมฝูงมดร่วมกับอัลกอริทึมการค้นหาแบบนกดูเหว่า (Hybrid ACO-CS)

การออกแบบจะใช้อัลกอริทึมการค้นหาแบบนกดูเหว่าแทนที่การค้นหาผลลัพธ์ที่ดีที่สุดขึ้นจากพื้นที่ใกล้เคียงตามที่ได้นำเสนอในหัวข้อ 3.2.2 (ขั้นตอนที่ 4-6) ซึ่งเป็นการรวมกันระหว่างอัลกอริทึมฝูงมดกับอัลกอริทึมการค้นหาแบบนกดูเหว่าเข้าด้วยกัน โดยภาพรวมการออกแบบแสดงในผังงานตามภาพประกอบ 14





ภาพประกอบ 14 ผังงานการปรับปรุงอัลกอริทึมฝูงมดร่วมกับอัลกอริทึมการค้นหาแบบนกดุเหว่า

3.3 การทดสอบประสิทธิภาพของอัลกอริทึม

3.3.1 การทดสอบประสิทธิภาพของอัลกอริทึมโดยใช้ฟังก์ชันทดสอบการหาค่าที่เหมาะสม

- 3.3.1.1 ฟังก์ชัน De Jong
- 3.3.1.2 ฟังก์ชัน Schwefel (Unimodal)
- 3.3.1.3 ฟังก์ชัน Step



- 3.3.1.4 ฟังก์ชัน Rosenbrock
 3.3.1.5 ฟังก์ชัน Ackley
 3.3.1.6 ฟังก์ชัน Rastrigin
 3.3.1.7 ฟังก์ชัน Griewank
 3.3.1.8 ฟังก์ชัน Stochastic ของ Kowalik
 3.3.1.9 ฟังก์ชัน Quartic
 3.3.1.10 ฟังก์ชัน Shekel's Foxholes

ตาราง 2 สรุปฟังก์ชันทดสอบมาตรฐาน

ฟังก์ชัน	มิติ	ช่วง	ค่าฟังก์ชัน
$f_1(x) = \sum_{i=1}^n x_i^2$	30	[-100,100]	0
$f_2(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	30	[-10,10]	0
$f_3(x) = \sum_{i=1}^n ((x_i + 0.5))^2$	30	[-100,100]	0
$f_4(x) = \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	30	[-30,30]	0
$f_5(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e$	30	[-32,32]	0
$f_6(x) = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i))$	30	[-5.12,5.12]	0
$f_7(x) = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	[-600,600]	0
$f_8(x) = \sum_{i=1}^{11} \left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4} \right]^2$	4	[-5,5]	0.00030



ตาราง 2 (ต่อ)

ฟังก์ชัน	มิติ	ช่วง	ค่าฟังก์ชัน
$f_9(x) = \sum_{i=1}^n ix_i^4 + random[0,1]$	30	[-1.28,1.28]	0
$f_{10}(x) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})^6} \right)$	2	[-65,-65]	1

3.3.2 การวิเคราะห์หาค่าพารามิเตอร์ที่เหมาะสมสำหรับอัลกอริทึม

การวิจัยครั้งนี้ดำเนินการทดลองและเก็บข้อมูลเพื่อทดสอบประสิทธิภาพกับระบบอนุกรมตามลำดับดังนี้

3.3.2.1 ค่าพารามิเตอร์ของอัลกอริทึมอัลกอริทึมฝูงมด หลักการเลวี่ไฟล์ท์ และอัลกอริทึมการค้นหาแบบนกดูเหว่า ที่สามารถปรับค่าได้ สรุปได้ดังนี้

ตาราง 3 ค่าพารามิเตอร์ของอัลกอริทึมอัลกอริทึมที่น่าเสนอที่สามารถปรับค่าได้

อัลกอริทึมฝูงมด	เลวี่ไฟล์ท์	อัลกอริทึมการค้นหาแบบนกดูเหว่า
จำนวนมด	ค่าคงที่ α	จำนวนรังนก
ค่าคงที่ α	ค่าคงที่ β	ค่าความน่าจะเป็นในการค้นพบ (pa)
ค่าคงที่ β	-	-
ค่าอัตราการระเหยของฟีโรโมน (ρ)	-	-

3.3.2.2 การทดลองและปรับแต่งค่าพารามิเตอร์ที่เหมาะสมสำหรับอัลกอริทึม

ACO2SWAP ซึ่งมีลำดับขั้นตอน ดังนี้

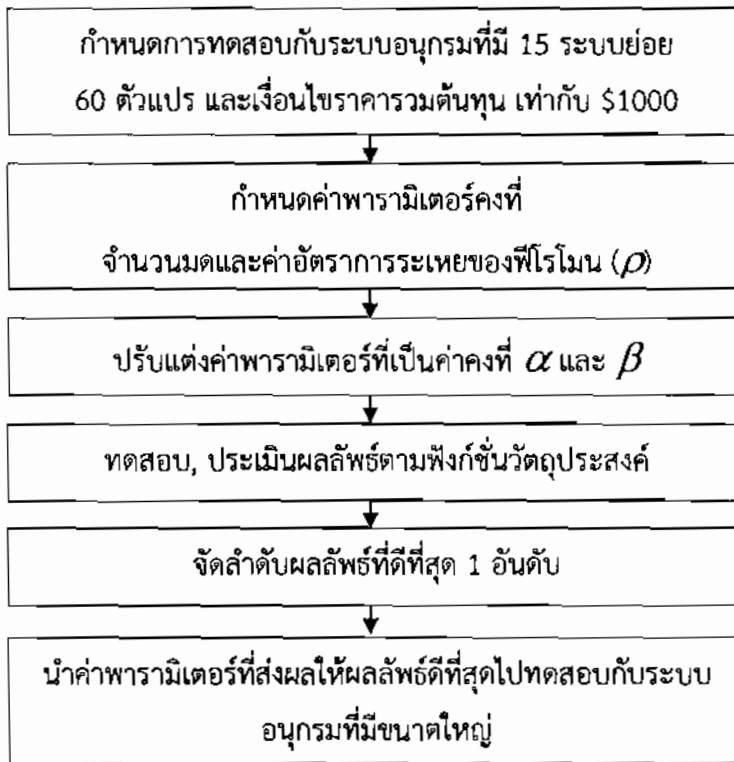
1. ทดสอบกับระบบอนุกรมที่มี 15 ระบบย่อย 60 ตัวแปร เจือปนไขว้การรวมต้นทุนเท่ากับ \$1000
2. กำหนดค่าพารามิเตอร์คงที่ ได้แก่ จำนวนมด 10 ตัว ค่าอัตราการระเหยของฟีโรโมน (ρ) เท่ากับ 0.01
3. ปรับแต่งค่าพารามิเตอร์ที่เป็นค่าคงที่ของอัลกอริทึมฝูงมด ซึ่งมี 2 ค่า คือ α เท่ากับ 0.5,0.8,1,2,3,4,5 และ β เท่ากับ 0.5,0.8,1,2,3,4,5 [16]
4. ทดสอบจำนวน 10 ครั้ง และกำหนดจำนวนรอบค้นหาสูงสุด 2,000 รอบ ต่อ 1 ครั้ง



5. ประเมินผลลัพธ์ตามฟังก์ชันวัตถุประสงค์
6. จัดลำดับผลลัพธ์ที่ดีที่สุด 1 อันดับ
7. นำค่าพารามิเตอร์ที่ส่งผลให้ผลลัพธ์ดีที่สุดไปทดสอบกับระบบอนุกรมที่มี

ขนาดใหญ่และมีตัวแปรตัดสินใจมากขึ้น

สรุปเป็นแผนภาพได้ดังนี้



ภาพประกอบ 15 แผนภาพสรุปแบบแผนการทดลองและปรับแต่งค่าพารามิเตอร์สำหรับอัลกอริทึม ACO2SWAP

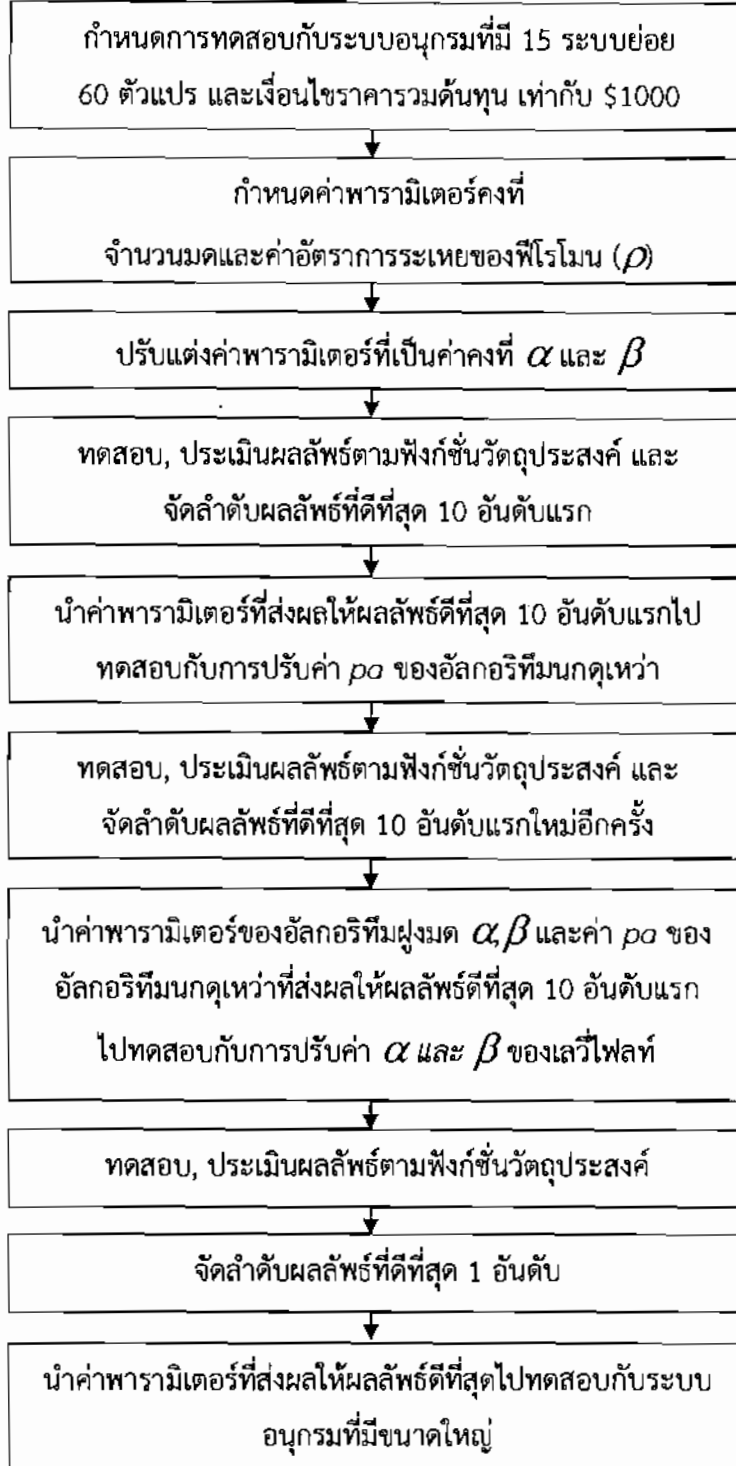
3.3.2.3 การทดลองและปรับแต่งค่าพารามิเตอร์ที่เหมาะสมสำหรับอัลกอริทึม ACO-Lévy flight และ Hybrid ACO-CS ซึ่งมีลำดับขั้นตอน ดังนี้

1. ทดสอบกับระบบอนุกรมที่มี 15 ระบบย่อย 60 ตัวแปร เงื่อนไขราคารวมต้นทุน เท่ากับ \$1000
2. กำหนดค่าพารามิเตอร์คงที่ที่เกี่ยวข้องกับอัลกอริทึมฝูงมด ได้แก่ จำนวนมด 10 ตัว ค่าอัตราการระเหยของฟีโรโมน (ρ) เท่ากับ 0.01



3. ปรับแต่งค่าพารามิเตอร์ที่เป็นค่าคงที่ของอัลกอริทึมฝูงมด ซึ่งมี 2 ค่า คือ α เท่ากับ 0.5,0.8,1,2,3,4,5 และ β เท่ากับ 0.5,0.8,1,2,3,4,5 [16]
 4. ทดสอบ จำนวน 10 ครั้ง และกำหนดจำนวนรอบค้นหาสูงสุด 2,000 รอบ ต่อ 1 ครั้ง
 5. ประเมินผลลัพธ์ตามฟังก์ชันวัตถุประสงค์
 6. จัดลำดับผลลัพธ์ที่ดีที่สุด 10 อันดับแรก
 7. นำค่าพารามิเตอร์ที่ส่งผลให้ผลลัพธ์ที่ดีที่สุด 10 อันดับแรกไปทดสอบกับการปรับค่าความน่าจะเป็นในการค้นพบ (po) เท่ากับ 0.01, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.50, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 1 ของอัลกอริทึมนกคุเหว่า[23],[51]
 8. ทดสอบจำนวน 10 ครั้ง ประเมินผลลัพธ์ตามฟังก์ชันวัตถุประสงค์
 9. จัดลำดับผลลัพธ์ที่ดีที่สุด 10 อันดับแรกใหม่อีกครั้ง
 10. นำค่าพารามิเตอร์ของอัลกอริทึมฝูงมด α, β และค่าความน่าจะเป็นในการค้นพบ (po) ของอัลกอริทึมนกคุเหว่า ที่ส่งผลให้ผลลัพธ์ที่ดีที่สุด 10 อันดับแรกไปทดสอบกับการปรับค่า α เท่ากับ 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1 และ β เท่ากับ 0.5, 1, 1.5, 2 ของเลวีไฟลท์ (Lévy flight) [23],[51]
 11. ทดสอบ จำนวน 10 ครั้ง ประเมินผลลัพธ์ตามฟังก์ชันวัตถุประสงค์
 12. จัดลำดับผลลัพธ์ที่ดีที่สุด 1 อันดับ
 13. นำค่าพารามิเตอร์ที่ส่งผลให้ผลลัพธ์ที่ดีที่สุดไปทดสอบกับระบบอนุกรมที่มีขนาดใหญ่และมีตัวแปรตัดสินใจมากขึ้นต่อไป
- สรุปเป็นแผนภาพได้ดังนี้





ภาพประกอบ 16 แผนภาพสรุปแบบแผนการทดลองและปรับแต่งค่าพารามิเตอร์สำหรับอัลกอริทึม ACO-Lévy flight และ Hybrid ACO-CS



3.3.3 ข้อมูลตัวอย่างระบบอนุกรมที่ใช้ทดสอบประสิทธิภาพของอัลกอริทึม

การทดสอบประสิทธิภาพของอัลกอริทึมที่นำเสนอในครั้งนี ใช้ข้อมูลตัวอย่างจากงานวิจัยที่ผ่านมา [16-17] รวม 5 ชุดตัวอย่าง ดังแสดงในภาคผนวก

3.3.3.1 ระบบอนุกรมที่มี 15 ระบบย่อยและตัวแปรที่ใช้ในการตัดสินใจ 60 ตัวแปร กำหนดเงื่อนไข ราคาเริ่มต้น เท่ากับ \$1000

3.3.3.2 ระบบอนุกรมที่มี 15 ระบบย่อยและตัวแปรที่ใช้ในการตัดสินใจ 80 ตัวแปร กำหนดเงื่อนไข ราคาเริ่มต้น \$900

3.3.3.3 ระบบอนุกรมที่มี 15 ระบบย่อยและตัวแปรที่ใช้ในการตัดสินใจ 100 ตัวแปร กำหนดเงื่อนไข ราคาเริ่มต้น เท่ากับ \$1000

3.3.3.4 ระบบอนุกรมที่มี 25 ระบบย่อยและตัวแปรที่ใช้ในการตัดสินใจ 166 ตัวแปร กำหนดเงื่อนไข ราคาเริ่มต้น เท่ากับ \$1400

3.3.3.5 ระบบอนุกรมที่มี 40 ระบบย่อยและตัวแปรที่ใช้ในการตัดสินใจ 266 ตัวแปร กำหนดเงื่อนไขราคาเริ่มต้น เท่ากับ \$2700

3.3.4 กรณีศึกษาในการทดสอบประสิทธิภาพของอัลกอริทึมกับระบบอนุกรม

เมื่อดำเนินการทดลองเพื่อวิเคราะห์หาค่าพารามิเตอร์ที่เหมาะสมสำหรับอัลกอริทึมแต่ละอัลกอริทึมเสร็จสิ้นแล้ว การทดลองในลำดับต่อไปจะแบ่งเป็นกรณีศึกษา ดังปรากฏในตาราง 4

ตาราง 4 สรุปกรณีศึกษาในการทดสอบประสิทธิภาพของอัลกอริทึมกับระบบอนุกรม

กรณีศึกษา	ระบบอนุกรม	จำนวนตัวแปร	เงื่อนไข ราคาเริ่มต้น (\$)	อัลกอริทึม
1	15 ระบบย่อย	60	1,000	ACO2SWAP
				ACO-Lévy flight
				Hybrid ACO-CS
2	15 ระบบย่อย	80	900	ACO2SWAP
				ACO-Lévy flight
				Hybrid ACO-CS
3	15 ระบบย่อย	100	1,000	ACO2SWAP
				ACO-Lévy flight
				Hybrid ACO-CS



ตาราง 4 (ต่อ)

กรณีศึกษา	ระบบอนุกรม	จำนวนตัวแปร	เงื่อนไข ราคารวมต้นทุน (\$)	อัลกอริทึม
4	25 ระบบย่อย	166	1,400	ACO2SWAP
				ACO-Lévy flight
				Hybrid ACO-CS
5	40 ระบบย่อย	266	2,700	ACO2SWAP
				ACO-Lévy flight
				Hybrid ACO-CS

3.4 การวิเคราะห์ข้อมูล

3.4.1 การวิเคราะห์ข้อมูลเพื่อเปรียบเทียบประสิทธิภาพของอัลกอริทึมใช้สถิติ ค่าเฉลี่ย (Mean) ร้อยละ (Percentage) และส่วนเบี่ยงเบนมาตรฐาน (Standard Deviation)

3.4.2 วิเคราะห์ข้อมูลโดยใช้สถิติค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐาน จากการประมวลผลการทำงานของอัลกอริทึมที่นำเสนอ โดยทดสอบการทำงาน 10 ครั้ง ในแต่ละชุดข้อมูลตัวอย่าง



บทที่ 4

ผลการวิจัย

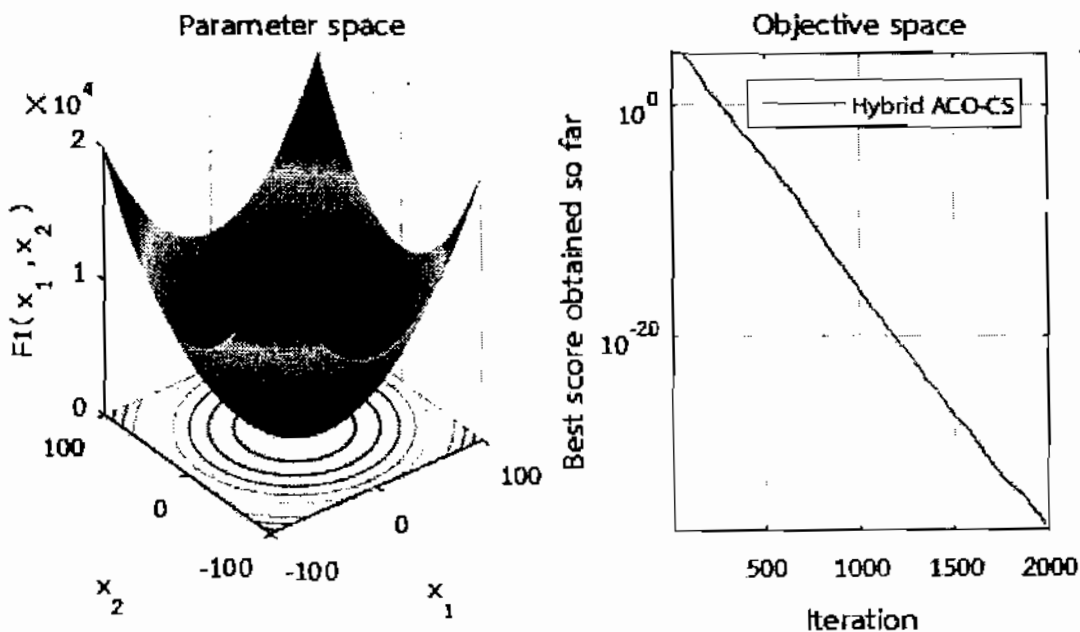
การวิจัยครั้งนี้ได้พัฒนาอัลกอริทึมที่เกิดจากการปรับปรุงขั้นตอนวิธีของอัลกอริทึมฝูงมด ร่วมกับอัลกอริทึมการค้นหาแบบดุกเหว่า (Hybrid Ant Colony Optimization Cuckoo Search: Hybrid ACO-CS) เพื่อแก้ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดสำหรับระบบอนุกรม ซึ่งมีการทดสอบประสิทธิภาพ ดึงนำเสนอตามลำดับดังนี้

4.1 ผลการทดสอบประสิทธิภาพอัลกอริทึมกับฟังก์ชันทดสอบการหาค่าที่เหมาะสม

4.1.1 ฟังก์ชัน De Jong หรือฟังก์ชัน Sphere

$$f_1(X) = \sum_{i=1}^n X_i^2 \quad (4.1)$$

เมื่อ X_i มีค่าอยู่ในช่วง $[-100,100]$ และ n คือ ขนาดของมิติของปัญหา ฟังก์ชันนี้มีค่าฟังก์ชันต่ำสุดเท่ากับ 0 กราฟและผลการค้นหาแสดงดังภาพประกอบ 16



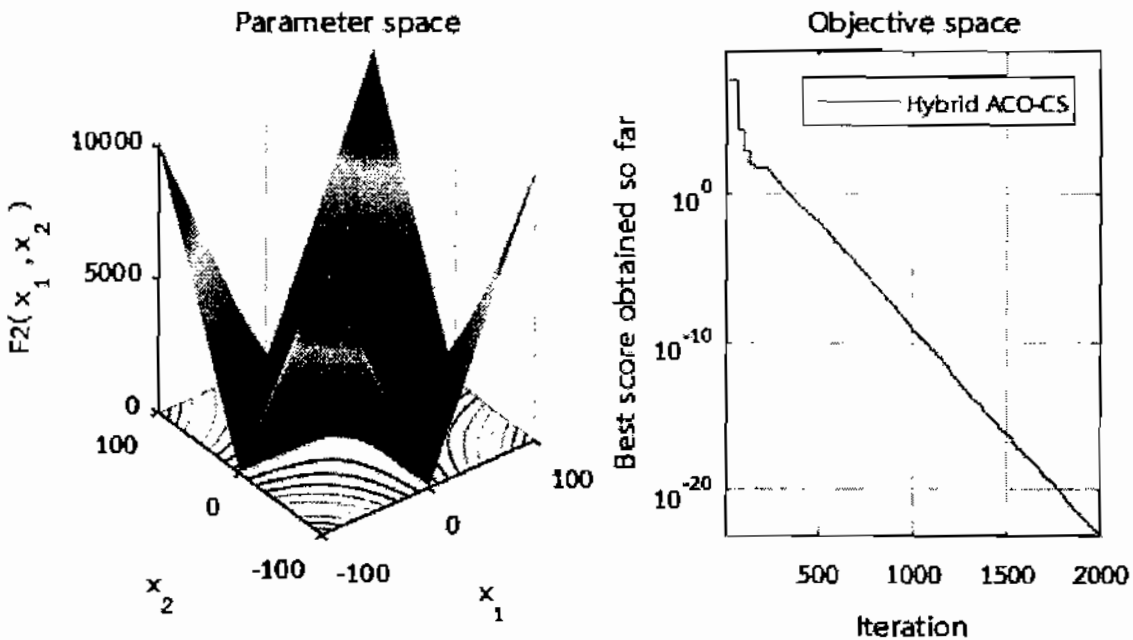
ภาพประกอบ 17 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Sphere



4.1.2 ฟังก์ชัน Schwefel (Unimodal)

$$f_2(X) = \sum_{i=1}^n |x_i| + \prod_{i=1}^n |x_i| \quad (4.2)$$

เมื่อ X มีค่าอยู่ในช่วง $[-10,10]$ และ n คือ ขนาดของมิติของปัญหา ฟังก์ชันนี้มีค่าฟังก์ชันต่ำสุดเท่ากับ 0 กราฟและผลการค้นหาแสดงดังภาพประกอบ 17



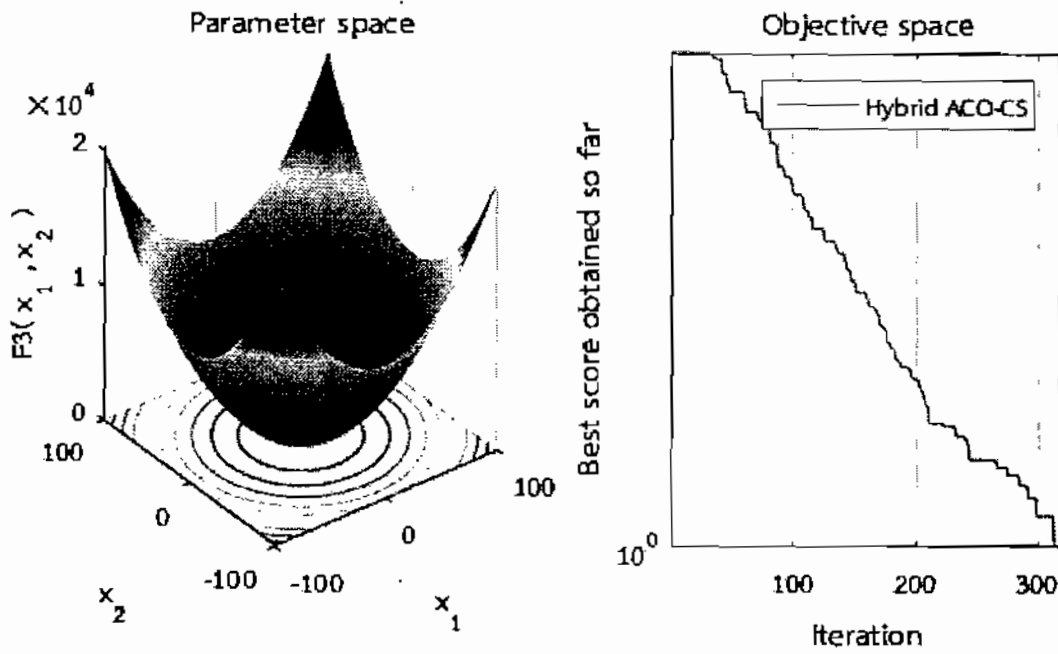
ภาพประกอบ 18 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Schwefel (Unimodal)

4.1.3 ฟังก์ชัน Step

$$f_3(X) = \sum_{i=1}^n ([x_i + 0.5])^2 \quad (4.3)$$

เมื่อ X มีค่าอยู่ในช่วง $[-100,100]$ และ n คือ ขนาดของมิติของปัญหา ฟังก์ชันนี้มีค่าฟังก์ชันต่ำสุดเท่ากับ 0 กราฟและผลการค้นหาแสดงดังภาพประกอบ 18





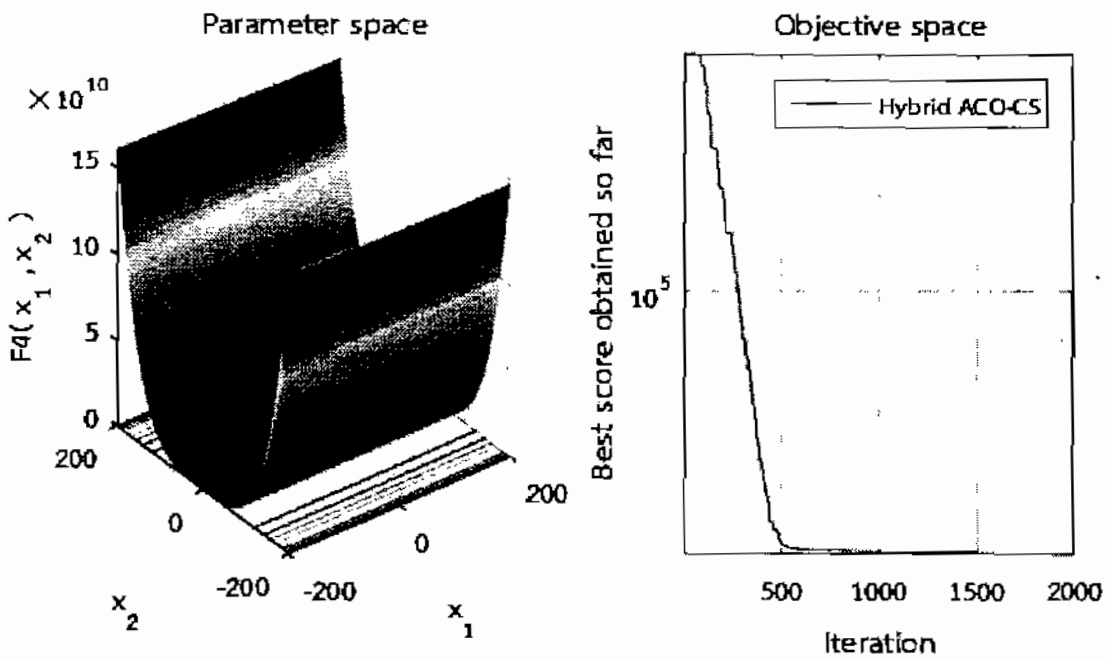
ภาพประกอบ 19 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Step

4.1.4 ฟังก์ชัน Rosenbrock

$$f_4(X) = \sum_{i=1}^{n-1} (100(X_{i+1} - X_i^2)^2 + (X_i - 1)^2) \quad (4.4)$$

เมื่อ X_i มีค่าอยู่ในช่วง $[-30, 30]$ และ n คือ ขนาดของมิติของปัญหา ฟังก์ชันนี้มีค่าฟังก์ชันต่ำสุดเท่ากับ 0 กราฟและผลการค้นหาแสดงดังภาพประกอบ 19





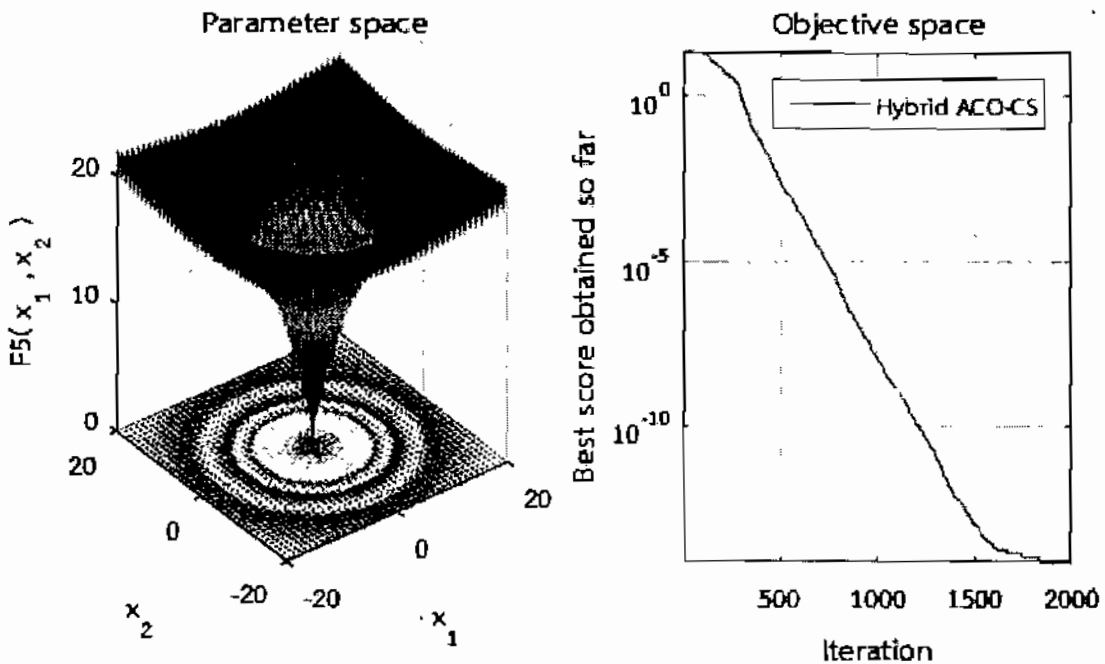
ภาพประกอบ 20 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Rosenbrock

4.1.5 ฟังก์ชัน Ackley

$$f_5(X) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n X_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi X_i)\right) + 20 + e \quad (4.5)$$

เมื่อ X_i มีค่าอยู่ในช่วง $[-32, 32]$ และ n คือ ขนาดของมิติของปัญหา ฟังก์ชันนี้มีค่าฟังก์ชันต่ำสุดเท่ากับ 0 กราฟและผลการค้นหาแสดงดังภาพประกอบ 20





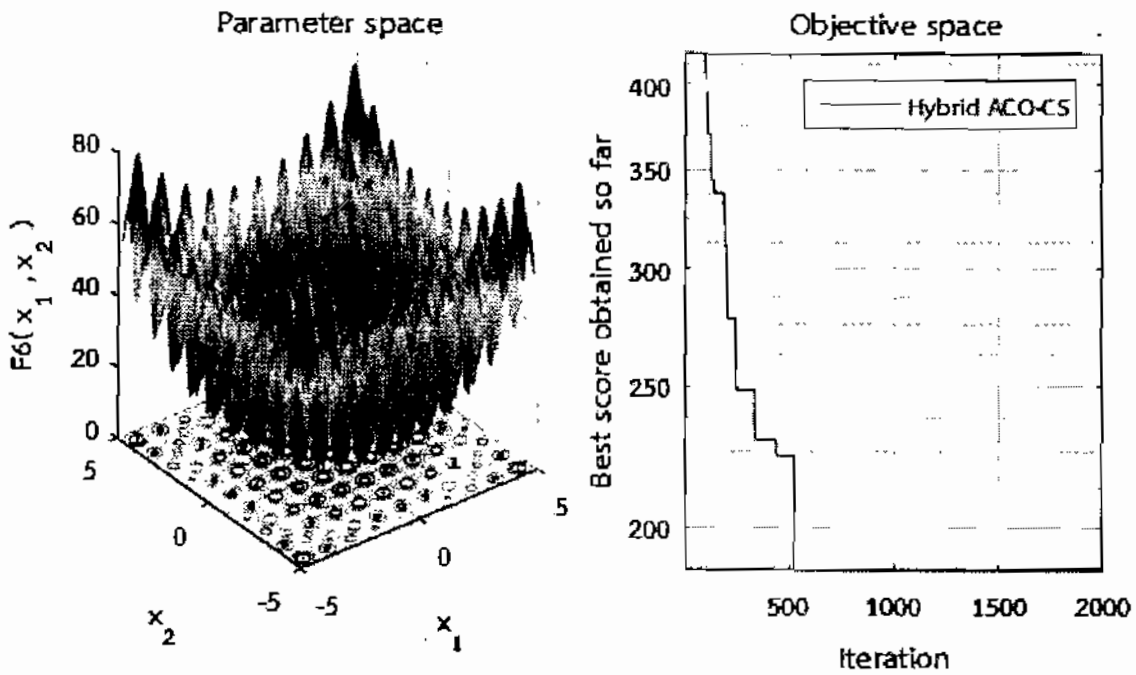
ภาพประกอบ 21 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Ackley

4.1.6 ฟังก์ชัน Rastrigin

$$f_6(X) = 10n + \sum_{i=1}^n (X_i^2 - 10 \cos(2\pi X_i)) \quad (4.6)$$

เมื่อ X_i มีค่าอยู่ในช่วง $[-5.12, 5.12]$ และ n คือ ขนาดของมิติของปัญหา ฟังก์ชันนี้มีค่าฟังก์ชันต่ำสุดเท่ากับ 0 กราฟและผลการค้นหาแสดงดังภาพประกอบ 21





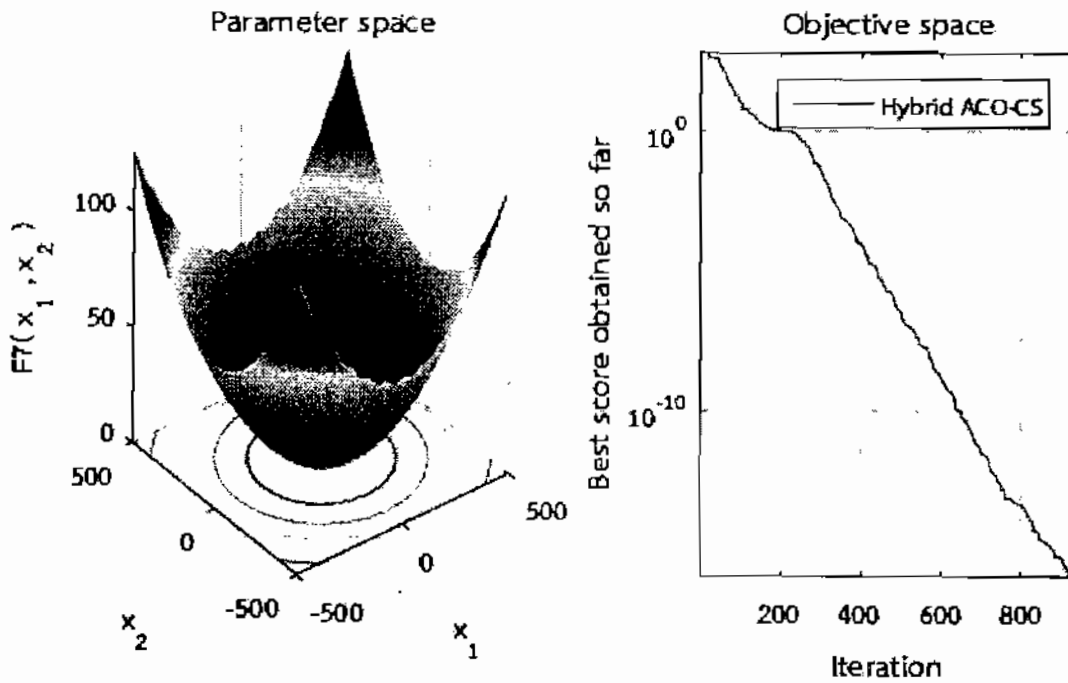
ภาพประกอบ 22 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Rastrigin

4.1.7 ฟังก์ชัน Griewank

$$f_7(x) = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad (4.7)$$

เมื่อ x_i มีค่าอยู่ในช่วง $[-600, 600]$ และ n คือ ขนาดของมิติของปัญหา ฟังก์ชันนี้มีค่าฟังก์ชันต่ำสุดเท่ากับ 0 กราฟและผลการค้นหาแสดงดังภาพประกอบ 22





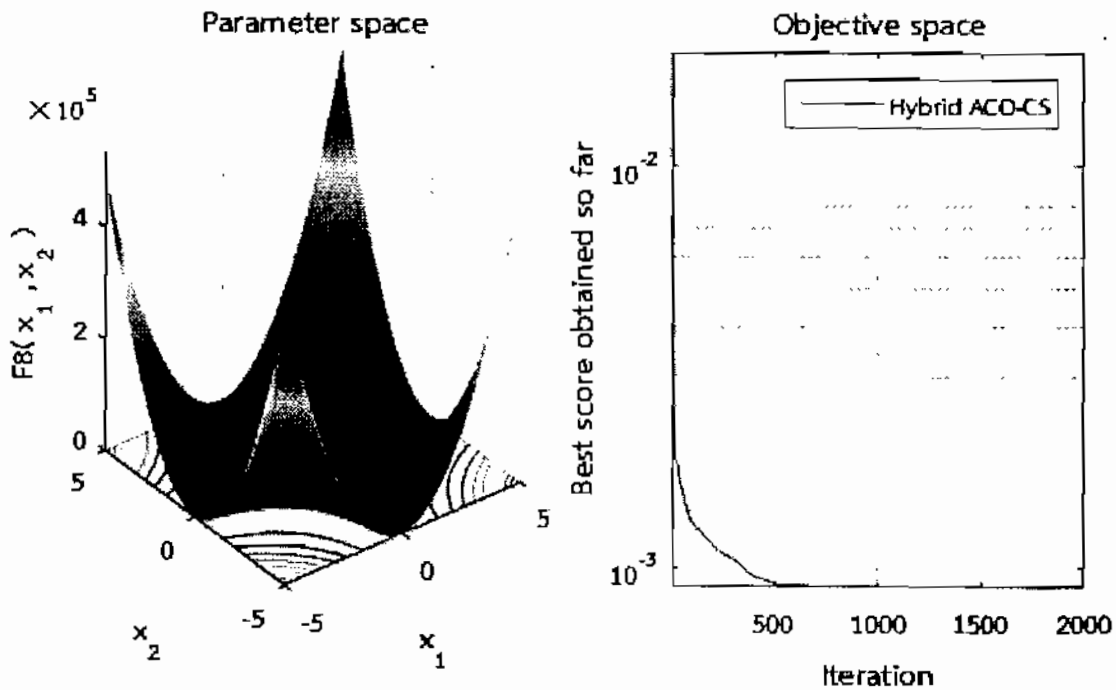
ภาพประกอบ 23 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Griewank

4.1.8 ฟังก์ชัน Stochastic ของ Kowalik

$$f_8(X) = \sum_{i=1}^{11} \left[a_i - \frac{X_1(b_i^2 + b_i X_2)}{b_i^2 + b_i X_3 + X_4} \right]^2 \quad (4.8)$$

เมื่อ X_i มีค่าอยู่ในช่วง $[-5,5]$ และ n คือ ขนาดของมิติของปัญหา ฟังก์ชันนี้มีค่าฟังก์ชันต่ำสุดเท่ากับ 0.00030 กราฟและผลการค้นหาแสดงดังภาพประกอบ 23





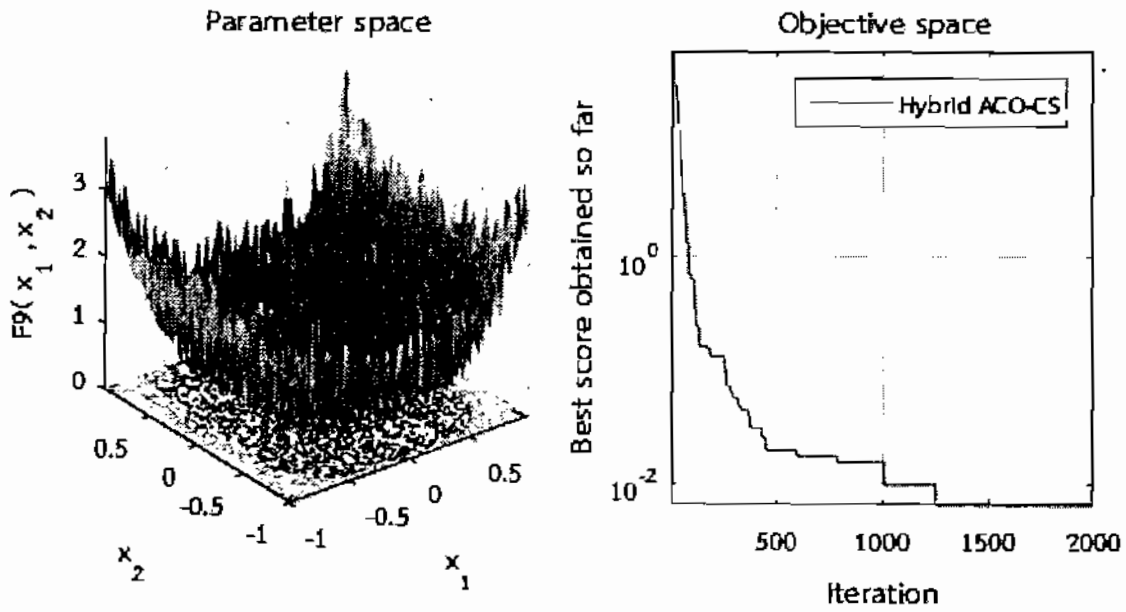
ภาพประกอบ 24 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Stochastic ของ Kowalik

4.1.9 ฟังก์ชัน Quartic

$$f_9(X) = \sum_{i=1}^n ix_i^4 + \text{random}[0,1] \quad (4.9)$$

เมื่อ X_i มีค่าอยู่ในช่วง $[-1.28, 1.28]$ และ n คือ ขนาดของมิติของปัญหา ฟังก์ชันนี้มีค่าฟังก์ชันต่ำสุดเท่ากับ 0 กราฟและผลการค้นหาแสดงดังภาพประกอบ 24





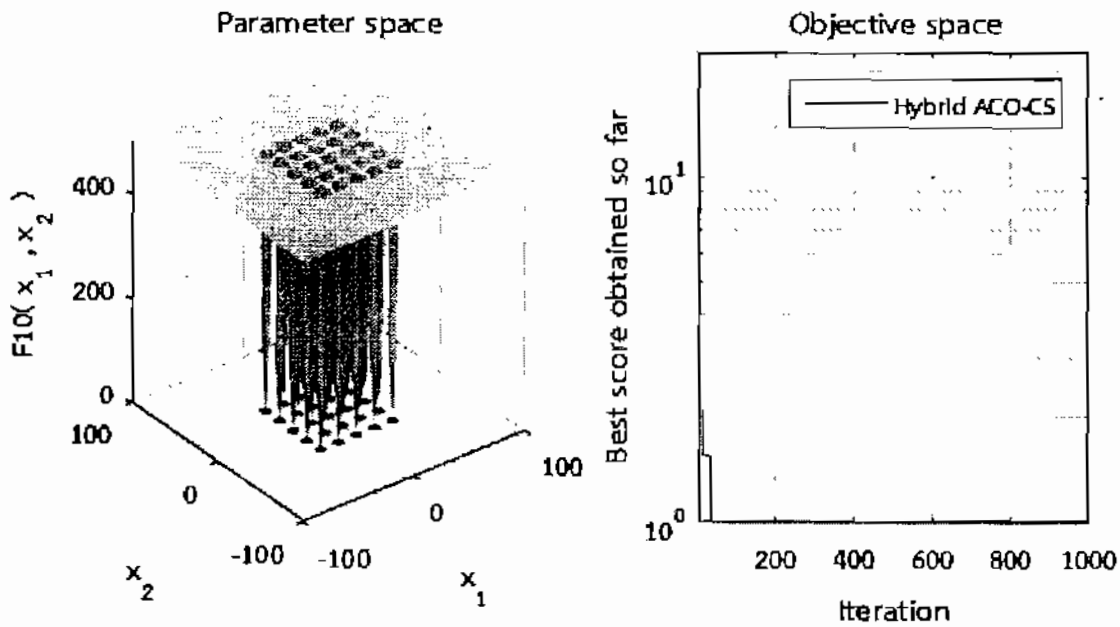
ภาพประกอบ 25 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Quartic

4.1.10 ฟังก์ชัน Shekel's Foxholes

$$f_{10}(X) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (X_i - a_{ij})^6} \right) \quad (4.10)$$

เมื่อ X_i มีค่าอยู่ในช่วง $[-65, 65]$ และ n คือ ขนาดของมิติของปัญหา ฟังก์ชันนี้มีค่าฟังก์ชันต่ำสุดเท่ากับ 1 กราฟและผลการค้นหาแสดงดังภาพประกอบ 25





ภาพประกอบ 26 ผลการทดสอบอัลกอริทึมกับฟังก์ชัน Shekel's Foxholes

การทดสอบประสิทธิภาพครั้งนี้ ได้นำผลลัพธ์ไปเปรียบเทียบกับงานวิจัยที่ปรับปรุงอัลกอริทึมขนาดหว่าจากวิธีการดั้งเดิม [55] โดยใช้การตั้งค่าพารามิเตอร์เดียวกันกับงานวิจัยดังกล่าว ได้แก่ รั้งนคหว่า 25 หน่วย, $\alpha=0.01$, $\beta=1.5$, $pc=0.25$ และตั้งค่าพารามิเตอร์สำหรับอัลกอริทึมฝูงมด จำนวนมด 25 ตัว, $\alpha=1$, $\beta=0.8$, $p=0.01$ [16] และจำนวนรอบการค้นหาสูงสุด 2000 รอบ ทดสอบทั้งสิ้น 10 ครั้ง ผลปรากฏดังตารางที่ 5

ตาราง 5 ผลการเปรียบเทียบประสิทธิภาพระหว่าง Hybrid ACO-CS กับ ACS [55]

ฟังก์ชัน	อัลกอริทึม	ผลลัพธ์ที่ดีที่สุด	\bar{X}	S.D.	เวลาเฉลี่ย (วินาที)
f_1	Hybrid ACO-CS*	2.4813e-011	2.8745e-011	1.9256e-012	13.60
	ACS	3.0916e-011	4.4757e-010	3.9038e-010	4.88
f_2	Hybrid ACO-CS*	6.8166e-005	7.0205e-005	1.9256e-012	11.93
	ACS	7.3860e-005	3.2220e-004	1.6471e-004	5.23
f_3	Hybrid ACO-CS*	0	0	0	5.47
	ACS	0	0	0	6.10
f_4	Hybrid ACO-CS	72.3650	84.2487	15.04 63	35.95
	ACS	19.1351	23.0209	1.5189	5.91



ตาราง 5 (ต่อ)

ฟังก์ชัน	อัลกอริทึม	ผลลัพธ์ที่ดีที่สุด	\bar{X}	S.D.	เวลาเฉลี่ย (วินาที)
f_5	Hybrid ACO-CS*	1.8131e-005	1.9123e-005	6.3675e-007	12.24
	ACS	2.0147e-005	2.1e-003	1.08e-002	5.40
f_6	Hybrid ACO-CS	150.4536	196.5353	25.9053	35.80
	ACS	53.8043	96.0495	18.7348	5.49
f_7	Hybrid ACO-CS*	0	0.0007	0.0023	12.09
	ACS	1.3959e-009	5.8510e-005	2.3119e-004	6.01
f_8	Hybrid ACO-CS	7.6653e-004	9.1676e-004	1.0395e-004	18.58
	ACS	3.0749e-004	3.0749e-004	3.2713e-018	5.25
f_9	Hybrid ACO-CS*	0.0079	0.0098	0.0022	12.28
	ACS	0.0093	0.0184	0.0050	6.14
f_{10}	Hybrid ACO-CS*	0.9980	0.9980	0	1.30
	ACS	0.9980	0.9980	0	16.99

จากตาราง 5 พบว่า อัลกอริทึม Hybrid ACO-CS ที่นำเสนอสามารถแก้ปัญหาการหาค่าที่เหมาะสมเชิงตัวเลขได้ทั้ง 10 ฟังก์ชัน และพบว่าผลลัพธ์ที่มีประสิทธิภาพดีกว่าอัลกอริทึม ACS ได้แก่ ฟังก์ชัน De Jong (f_1), ฟังก์ชัน Schwefel (Unimodal) (f_2), ฟังก์ชัน Ackley (f_3), ฟังก์ชัน Griewank (f_7) และฟังก์ชัน Quartic (f_9) แต่เป็นรองในด้านของเวลาเฉลี่ยของการค้นหาซึ่งไม่มากนัก นอกจากนี้ยังพบว่า มีผลลัพธ์ที่เท่ากันคือ ฟังก์ชัน Step (f_5) และฟังก์ชัน Shekel's Foxholes (f_{10}) แต่ใช้เวลาเฉลี่ยในการค้นหาที่ดีกว่า

4.2 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เหมาะสมของอัลกอริทึม

การวิเคราะห์หาค่าพารามิเตอร์ที่เหมาะสมในแต่ละอัลกอริทึมที่นำเสนอ เพื่อหาค่าพารามิเตอร์ที่ส่งผลให้มีผลลัพธ์ที่ดีที่สุดสำหรับการทดสอบประสิทธิภาพอัลกอริทึมกับระบบอนุกรมตามฟังก์ชันวัตถุประสงค์ โดยดำเนินการทดสอบกับระบบอนุกรมที่มี 15 ระบบย่อย 60 ตัวแปร เงื่อนไขการรวมต้นทุน เท่ากับ \$1000 และกำหนดค่าพารามิเตอร์คงที่ ได้แก่ จำนวนมด 10 ตัว ค่าอัตราการระเหยของฟีโรโมน (ρ) เท่ากับ 0.01 ทดสอบซ้ำ 10 ครั้ง และจำนวนรอบค้นหาสูงสุด 2,000 รอบ ต่อ 1 ครั้ง ผลการวิเคราะห์นำเสนอตามลำดับดังนี้



4.2.1 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เหมาะสมสำหรับอัลกอริทึม ACO2SWAP

การปรับค่าของค่าพารามิเตอร์ที่เป็นค่าคงที่ของอัลกอริทึม ACO2SWAP มี 2 ค่า คือ α และ β ปรากฏผลดังตาราง 6

ตาราง 6 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เหมาะสมสำหรับอัลกอริทึม ACO2SWAP

$\alpha \backslash \beta$	0.5	0.8	1	2	3	4	5
0.5	0.857054*	0.857054*	0.857054*	0.856435	0.854649	0.854687	0.854489
0.8	0.857054*	0.856435	0.857054*	0.855493	0.854866	0.855043	0.852305
1	0.855121	0.856367	0.856865	0.855896	0.855043	0.851583	0.85368
2	0.853111	0.853023	0.854348	0.85375	0.853692	0.849539	0.850715
3	0.854375	0.854023	0.852905	0.850162	0.846774	0.848873	0.849628
4	0.849215	0.849441	0.852813	0.849588	0.847642	0.847973	0.847261
5	0.853184	0.852961	0.855229	0.85255	0.846987	0.843456	0.845072

จากตาราง 6 พบว่า ผลการวิเคราะห์ค่าพารามิเตอร์ที่เหมาะสมสำหรับอัลกอริทึม ACO2SWAP ที่ส่งผลให้มีผลลัพธ์ที่ดีที่สุด มี 5 อันดับสูงสุด คือ

1. ค่า $\alpha = 0.5$ และ $\beta = 0.5$
2. ค่า $\alpha = 0.5$ และ $\beta = 0.8$
3. ค่า $\alpha = 0.5$ และ $\beta = 1$
4. ค่า $\alpha = 0.8$ และ $\beta = 0.5$
5. ค่า $\alpha = 0.8$ และ $\beta = 1$

ดังนั้นเพื่อคัดเลือกให้เหลือเพียง 1 อันดับ เพื่อนำไปใช้ทดสอบประสิทธิภาพกับระบบอนุกรมขนาดใหญ่ขึ้น จึงได้พิจารณาจากเวลาเฉลี่ยที่ใช้ในการประมวลผล ดังแสดงใน ตาราง 7

ตาราง 7 เวลาเฉลี่ยที่ใช้ในการประมวลผลอัลกอริทึม ACO2SWAP ของค่าพารามิเตอร์ α และ β

ค่าพารามิเตอร์	ค่าความน่าเชื่อถือ	เวลาเฉลี่ย (วินาที)
1. ค่า $\alpha = 0.5$ และ $\beta = 0.5$	0.857054	18.77
2. ค่า $\alpha = 0.5$ และ $\beta = 0.8$	0.857054	16.60
3. ค่า $\alpha = 0.5$ และ $\beta = 1$	0.857054	31.72
4. ค่า $\alpha = 0.8$ และ $\beta = 0.5$	0.857054	18.06
5. ค่า $\alpha = 0.8$ และ $\beta = 1$	0.857054	13.63*



จากตาราง 7 พบว่า ผลการวิเคราะห์เวลาเฉลี่ยค่าพารามิเตอร์ที่เหมาะสมสำหรับอัลกอริทึม ACO2SWAP ที่ส่งผลให้มีผลลัพธ์ค่าความน่าเชื่อถือที่ดีที่สุด คือ ค่า $\alpha = 0.8$ และ $\beta = 1$

ดังนั้น ผู้วิจัยจึงนำค่าพารามิเตอร์ที่ได้นี้ไปทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กับระบบอนุกรมขนาดใหญ่

4.2.2 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เหมาะสมสำหรับอัลกอริทึม ACO-Lévy flight และ Hybrid ACO-CS

การปรับค่าของค่าพารามิเตอร์ที่เป็นค่าคงที่ของอัลกอริทึม ACO-Lévy flight และ Hybrid ACO-CS นำเสนอตามลำดับดังนี้

4.2.2.1 การปรับค่าพารามิเตอร์ที่เกี่ยวข้องกับอัลกอริทึมฝูงมด ซึ่งมี 2 ค่า คือ α และ β ผลการวิเคราะห์ข้อมูลปรากฏดังนี้

ตาราง 8 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เกี่ยวข้องกับอัลกอริทึมฝูงมด

$\alpha \backslash \beta$	0.5	0.8	1	2	3	4	5
0.5	0.85447*	0.854002	0.851493	0.796272	0.715903	0.467242	0.363555
0.8	0.856135*	0.852553	0.85446*	0.85197	0.659895	0.472154	0.351566
1	0.855235*	0.855054*	0.854282*	0.821167	0.740338	0.544813	0.411796
2	0.856084	0.853468	0.855295*	0.796615	0.650309	0.486521	0.374901
3	0.848407	0.854888*	0.845758	0.802872	0.649237	0.500313	0.404505
4	0.853516	0.841122	0.854133	0.815914	0.629404	0.496115	0.373155
5	0.855543*	0.844983	0.829814	0.79124	0.641809	0.46382	0.40101

จากตาราง 8 พบว่า ผลการวิเคราะห์ค่าพารามิเตอร์ที่เกี่ยวข้องกับอัลกอริทึมฝูงมด 10 ลำดับแรกที่ให้ผลลัพธ์สูงสุดคือ

ลำดับที่ 1 $\alpha = 0.8$ และ $\beta = 0.5$

ลำดับที่ 2 $\alpha = 2$ และ $\beta = 0.5$

ลำดับที่ 3 $\alpha = 5$ และ $\beta = 0.5$

ลำดับที่ 4 $\alpha = 2$ และ $\beta = 1$

ลำดับที่ 5 $\alpha = 1$ และ $\beta = 0.5$

ลำดับที่ 6 $\alpha = 1$ และ $\beta = 0.8$



ลำดับที่ 7 $\alpha = 3$ และ $\beta = 0.8$

ลำดับที่ 8 $\alpha = 0.5$ และ $\beta = 0.5$

ลำดับที่ 9 $\alpha = 0.8$ และ $\beta = 1$

ลำดับที่ 10 $\alpha = 4$ และ $\beta = 1$

4.2.2.2 การปรับค่าพารามิเตอร์ที่เกี่ยวข้องกับอัลกอริทึมขนาดเหว่า คือ ค่าความน่าจะเป็นในการค้นพบ (pa) โดยการนำค่าพารามิเตอร์ที่เกี่ยวข้องกับอัลกอริทึมผสมคที่ให้ผลลัพธ์ที่ดีที่สุด 10 ลำดับแรกมารวมวิเคราะห์ ซึ่งผลการวิเคราะห์ข้อมูลปรากฏดังนี้

ตาราง 9 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เกี่ยวข้องกับอัลกอริทึมขนาดเหว่า

ค่า pa อัลกอริทึม ขนาดเหว่า	ค่า α, β อัลกอริทึมผสมค									
	0.5, 0.5	0.8, 0.5	0.8, 1	1, 0.5	1, 0.8	1, 1	2, 0.5	2, 1	3, 0.8	5, 0.5
0.01	0.8464	0.8504	0.8437	0.8527	0.8519	0.8478	0.8447	0.8267	0.8039	0.8215
0.05	0.8497	0.8478	0.8464	0.8511	0.8500	0.8470	0.8445	0.8469	0.8307	0.8244
0.10	0.8463	0.8538	0.8516	0.8545	0.8500	0.8502	0.8500	0.8494	0.8427	0.8348
0.15	0.8471	0.8538	0.8535	0.8538	0.8518	0.8550	0.8512	0.8414	0.8360	0.8441
0.20	0.8547	0.8515	0.8529	0.8552	0.8529	0.8538	0.8457	0.8503	0.8458	0.8511
0.25	0.8552	0.8527	0.8505	0.8567	0.8555	0.8537	0.8550	0.8549	0.8545	0.8512
0.30	0.8545	0.8550	0.8543	0.8548	0.8548	0.8535	0.8532	0.8507	0.8491	0.8524
0.35	0.8538	0.8560	0.8551	0.8552	0.8549	0.8544	0.8494	0.8544	0.8446	0.8522
0.40	0.8557	0.8548	0.8521	0.8560	0.8553	0.8532	0.8520	0.8520	0.8523	0.8567
0.45	0.8562	0.8564	0.8544	0.8564	0.8556	0.8560	0.8556	0.8535	0.8534	0.8561
0.50	0.8568	0.8546	0.8559	0.8553	0.8560	0.8556	0.8559	0.8434	0.8567	0.8369
0.55	0.8562	0.8564	0.8548	0.8566	0.8566	0.8551	0.8566	0.8561	0.8567	0.8570*
0.60	0.8568	0.8569*	0.8565	0.8566	0.8565	0.8532	0.8570*	0.8561	0.8545	0.8522
0.65	0.8568	0.8566	0.8570*	0.8569*	0.8567	0.8568	0.8568	0.8568	0.8561	0.8568
0.70	0.8568	0.8568	0.8567	0.8567	0.8569*	0.8566	0.8569	0.8570*	0.8569	0.8440
0.75	0.8568	0.8566	0.8560	0.8569*	0.8568	0.8569*	0.8567	0.8564	0.8569*	0.8569
0.80	0.8562	0.8560	0.8567	0.8566	0.8568	0.8565	0.8566	0.8544	0.8567	0.8566
0.85	0.8560	0.8566	0.8554	0.8566	0.8559	0.8561	0.8563	0.8563	0.8555	0.8563
0.90	0.8544	0.8539	0.8550	0.8547	0.8550	0.8533	0.8539	0.8532	0.8528	0.8538
0.95	0.8412	0.8492	0.8347	0.8503	0.8473	0.8415	0.8249	0.8401	0.8261	0.8262
1	0.8364	0.8466	0.8296	0.8527	0.8457	0.8311	0.8042	0.7654	0.7215	0.7002



จากตาราง 9 พบว่า ผลการวิเคราะห์ค่าพารามิเตอร์ที่เกี่ยวข้องกับอัลกอริทึมฝูงมด 10 ลำดับแรกที่ให้ผลลัพธ์สูงสุดคือ

ลำดับที่ 1 $\alpha = 2, \beta = 1$ และ $pa = 0.7$

ลำดับที่ 2 $\alpha = 5, \beta = 0.5$ และ $pa = 0.55$

ลำดับที่ 3 $\alpha = 0.8, \beta = 1$ และ $pa = 0.65$

ลำดับที่ 4 $\alpha = 2, \beta = 0.5$ และ $pa = 0.6$

ลำดับที่ 5 $\alpha = 1, \beta = 0.5$ และ $pa = 0.65$

ลำดับที่ 6 $\alpha = 1, \beta = 0.8$ และ $pa = 0.7$

ลำดับที่ 7 $\alpha = 3, \beta = 0.8$ และ $pa = 0.75$

ลำดับที่ 8 $\alpha = 0.8, \beta = 0.5$ และ $pa = 0.6$

ลำดับที่ 9 $\alpha = 1, \beta = 0.5$ และ $pa = 0.75$

ลำดับที่ 10 $\alpha = 1, \beta = 1$ และ $pa = 0.75$

4.2.2.3 การปรับค่าพารามิเตอร์ที่เกี่ยวข้องกับเลวีไฟลท์ คือ ค่าคงที่ α และ β

โดยการนำค่าพารามิเตอร์ที่เกี่ยวข้องกับอัลกอริทึมฝูงมด α, β มาร่วมวิเคราะห์กับค่าความน่าจะเป็นในการค้นพบ (pa) ของอัลกอริทึมนกดูเหว่าที่ให้ผลลัพธ์ที่ดีที่สุด 10 ลำดับแรกซึ่งได้จากผลการทดลองที่ผ่านมา ผลการวิเคราะห์ข้อมูลปรากฏดังนี้

ตาราง 10 ผลการวิเคราะห์ค่าพารามิเตอร์ที่เกี่ยวข้องกับเลวีไฟลท์

เลวีไฟลท์ $\beta \alpha$	ค่า α, β อัลกอริทึมฝูงมด, ค่า pa อัลกอริทึมนกดูเหว่า									
	0.8, 0.5, $pa=0.6$	0.8, 1, $pa=0.65$	1, 0.5, $pa=0.65$	1, 0.5, $pa=0.75$	1, 0.8, $pa=0.7$	1, 1, $pa=0.75$	2, 0.5, $pa=0.6$	2, 1, $pa=0.7$	3, 0.8, $pa=0.75$	5, 0.5, $pa=0.55$
0.5, 1	0.85601	0.85410	0.85642	0.85659	0.85183	0.85377	0.85610	0.84253	0.84727	0.85250
1, 1	0.85558	0.85540	0.85661	0.85621	0.85545	0.85463	0.85413	0.84480	0.84073	0.84973
1.5, 1	0.85656	0.85491	0.85592	0.85570	0.85591	0.85445	0.85292	0.84271	0.84574	0.84908
2, 1	0.85675	0.85680	0.85705	0.85685	0.85696	0.85680	0.85682	0.85615	0.85675	0.85362
2, 0.01	0.85680	0.85680	0.85692	0.85677	0.85677	0.85696	0.85596	0.84404	0.85220	0.85365
2, 0.05	0.85689	0.85680	0.85678	0.85689	0.85668	0.85673	0.85646	0.85675	0.85357	0.85699
2, 0.1	0.85687	0.85682	0.85675	0.85689	0.85682	0.85669	0.85680	0.85671	0.85682	0.85662
2, 0.2	0.85689	0.85666	0.85689	0.85666	0.85587	0.85680	0.85680	0.85254	0.85591	0.85689
2, 0.3	0.85680	0.85663	0.85673	0.85682	0.85666	0.85654	0.85672	0.85667	0.85592	0.85255



ตาราง 13 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 1

จำนวนมดและ รังนกคุเหว่า	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	88.24	0.839301	0.855338	0.848277	0.005518
20	167.66	0.844914	0.854652	0.849814	0.002847
30	231.10	0.847373	0.857054	0.852928	0.003319
40	320.54	0.853707	0.857054	0.855716	0.001046
50*	401.90	0.855252	0.857054	0.856161	0.000519

จากตาราง 13 พบว่า จากการทดลองอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 1 จำนวน 10 ครั้ง โดยการตั้งค่าจำนวนมดและรังนก 10, 20, และ 50 สามารถประมวลผลค้นหาคำตอบหาค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงสุด คือ 0.857054 ซึ่งเมื่อพิจารณาแล้วพบว่า การตั้งค่าจำนวนมดและรังนก เท่ากับ 50 ส่งผลให้มีค่าความน่าเชื่อถือที่เหมาะสมโดยเฉลี่ยสูงสุด คือ 0.856161 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.000519 แต่เมื่อพิจารณาในด้านเวลาใช้เวลาเฉลี่ยในการค้นหาข้อมูลสูงที่สุด คือ 401.90 วินาที

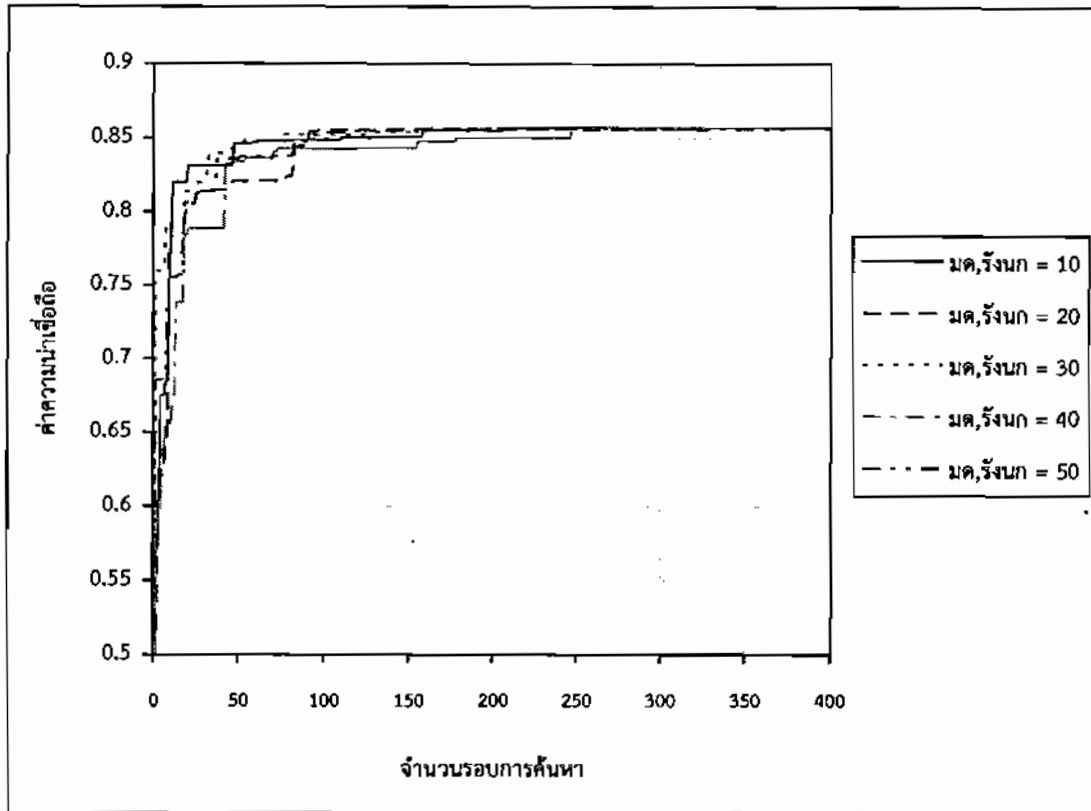
ตาราง 14 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 1

จำนวนมดและ รังนกคุเหว่า	เทคโนโลยีที่เลือก	ค่าความ น่าเชื่อถือสูงสุด	ราคาต้นทุน รวม (\$)
10	4-3-5-2-3-3-2-3-2-2-3-4-3-2	0.855338	980
20	3-3-4-2-3-3-2-4-2-2-3-4-3-3	0.854652	990
30	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990
40	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990
50	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990

จากตาราง 14 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 1 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด โดยการปรับค่าจำนวนมดและรังนก 30, 40 และ 50 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุด เป็นเทคโนโลยีเดียวกันทั้งหมด คือ 3-4-5-2-3-3-2-3-2-2-2-3-4-3-2 และใช้งบประมาณทั้งสิ้น \$990



4.3.1.3 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS



ภาพประกอบ 29 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 1

จากภาพประกอบ 29 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษาที่ 1 โดยการปรับค่าจำนวนมดและรังนก 10, 20, 30, 40 และ 50 ผลลัพธ์ของการค้นหาค่าตอบที่ดีที่สุด พบว่า แนวโน้มการเข้าสู่ค่าตอบของการหาค่าความน่าเชื่อถือที่เหมาะสม ใช้จำนวนรอบในการค้นหาใกล้เคียงกันโดยประมาณ อยู่ในช่วง 200 - 250 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 15



ตาราง 15 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 1

จำนวนมดและ รังนกคุเหว่า	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10*	30.19	0.857054	0.857054	0.857054	0
20	34.63	0.857054	0.857054	0.857054	0
30	43.78	0.857054	0.857054	0.857054	0
40	55.09	0.857054	0.857054	0.857054	0
50	78.92	0.857054	0.857054	0.857054	0

จากตาราง 15 พบว่า จากการทดลองอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษา ที่ 1 จำนวน 10 ครั้ง สามารถประมวลผลค้นหาคำตอบหาค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงสุด เท่ากัน ทั้งหมด ซึ่งมีค่าเฉลี่ยเท่ากับ 0.857054 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.00 และพบว่า การตั้ง ค่าจำนวนมดและรังนก เท่ากับ 10 ใช้เวลาเฉลี่ยในการค้นหาข้อมูลเร็วที่สุด เท่ากับ 30.19 วินาที

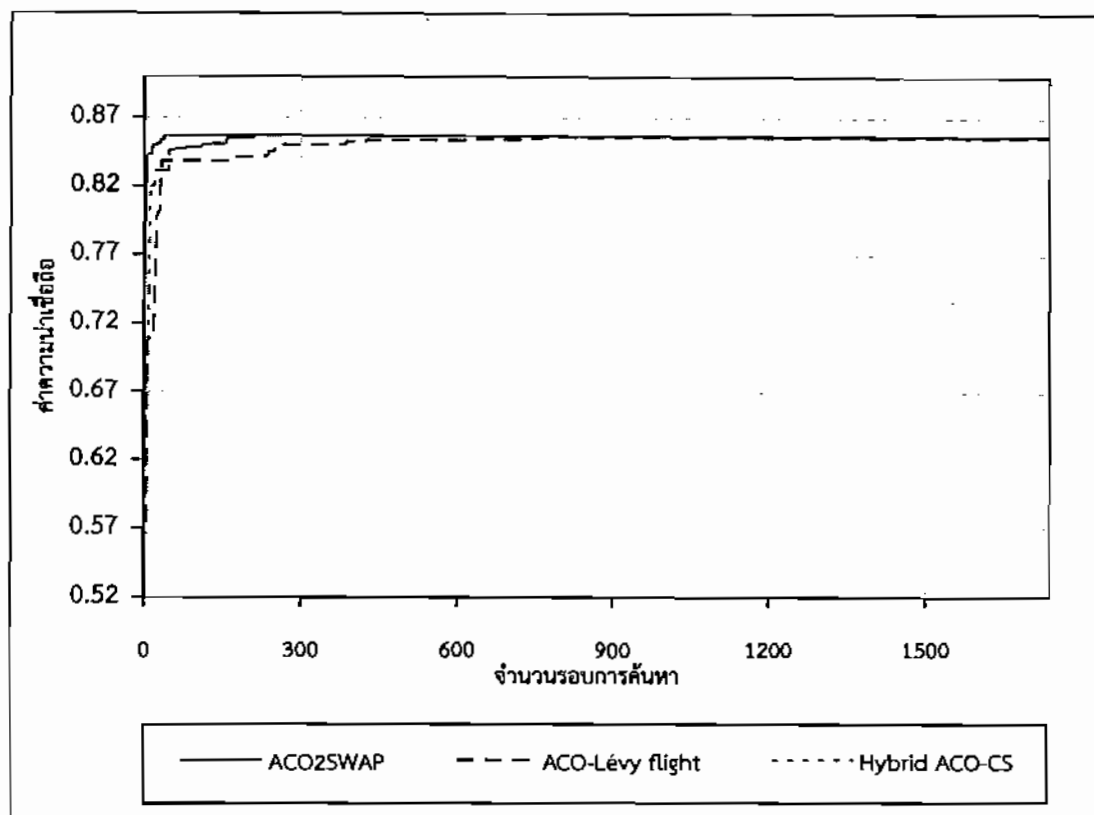
ตาราง 16 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 1

จำนวนมดและ รังนกคุเหว่า	เทคโนโลยีที่เลือก	ค่าความ น่าเชื่อถือสูงสุด	ราคาต้นทุน รวม (\$)
10	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990
20	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990
30	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990
40	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990
50	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990

จากตาราง 16 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม Hybrid ACO-CS สำหรับ กรณีศึกษาที่ 1 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด โดยการปรับค่าจำนวนมดและรังนก 10, 20, 30, 40 และ 50 มีผลการคัดเลือก เทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือ สูงสุด เป็นเทคโนโลยีเดียวกันทั้งหมด คือ 3-4-5-2-3-3-2-3-2-2-2-3-4-3-2 และใช้งบประมาณทั้งสิ้น \$990



4.3.1.4 เปรียบเทียบผลการทดสอบระหว่างอัลกอริทึม ACO2SWAP, ACO-Lévy flight และ Hybrid ACO-CS กรณีศึกษาที่ 1



ภาพประกอบ 30 ผลการเปรียบเทียบระหว่างอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 1

จากภาพประกอบ 30 เป็นการนำเสนอผลการเปรียบเทียบการค้นหาคำตอบที่ดีที่สุดของอัลกอริทึม ACO2SWAP, ACO-Lévy flight และ Hybrid ACO-CS สำหรับกรณีศึกษาที่ 1 ซึ่งมีแนวโน้มการเข้าสู่คำตอบของอัลกอริทึม ACO2SWAP และ Hybrid ACO-CS ใช้จำนวนรอบในการค้นหาคำตอบ โดยประมาณคือ 50-200 รอบ สำหรับ ACO-Lévy flight ใช้จำนวนรอบในการค้นหาคำตอบ โดยประมาณ 600 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 17



ตาราง 17 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 1

อัลกอริทึม	เวลาเฉลี่ย (s)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
ACO2SWAP*	23.34	0.857054	0.857054	0.857054	0
ACO-Lévy flight	401.90	0.855252	0.857054	0.856161	0.000519
Hybrid ACO-CS*	30.19	0.857054	0.857054	0.857054	0

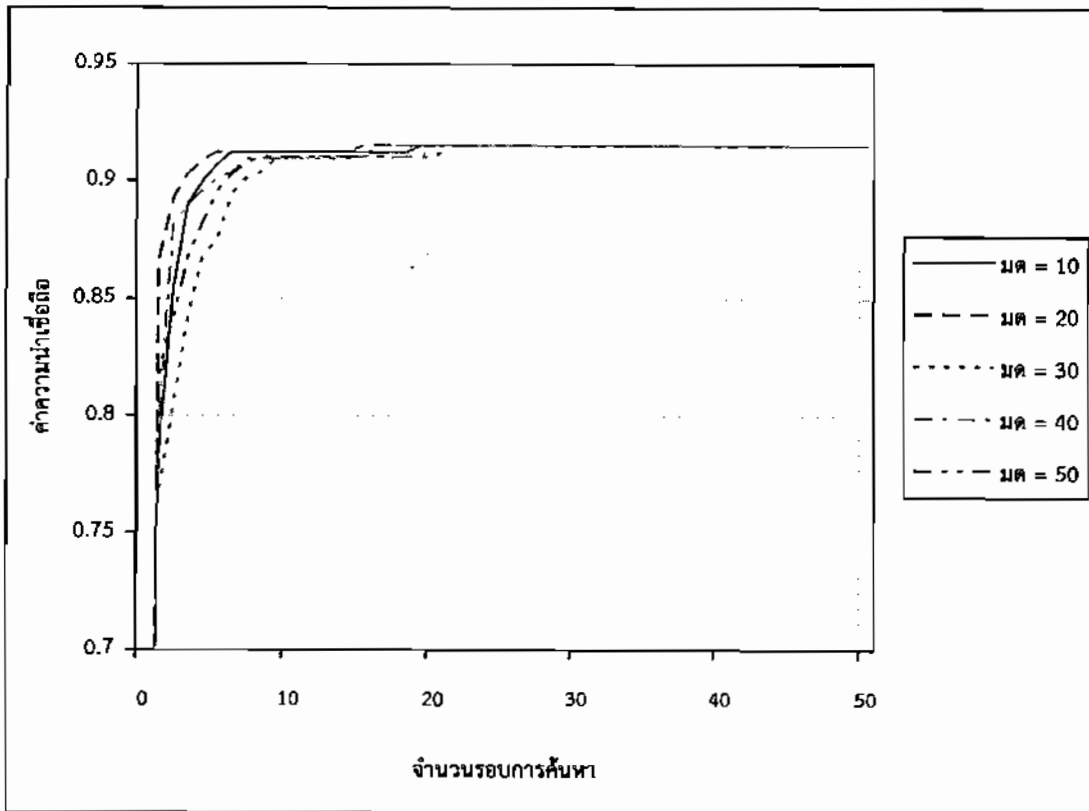
จากตาราง 17 โดยภาพรวมผลลัพธ์การค้นหาคำตอบที่ดีที่สุดสำหรับกรณีศึกษาที่ 1 ของแต่ละอัลกอริทึม พบว่า อัลกอริทึม ACO2SWAP และ Hybrid ACO-CS สามารถหาค่าความน่าเชื่อถือที่เหมาะสมที่สุด ดีกว่า ACO-Lévy flight และใช้เวลาเฉลี่ยในการค้นหาคำตอบใกล้เคียงกัน เมื่อพิจารณาจากค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐานแสดงให้เห็นว่าอัลกอริทึม ACO2SWAP และ Hybrid ACO-CS มีความแม่นยำสูง สำหรับการประมวลผลระบบอนุกรมที่มี 15 ระบบย่อย 60 ตัวแปร

4.3.2 กรณีศึกษาที่ 2

ระบบอนุกรมที่มี 15 ระบบย่อย 80 ตัวแปร กำหนดเงื่อนไขราคารวมต้นทุน เท่ากับ \$900

4.3.2.1 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP





ภาพประกอบ 31 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 2

จากภาพประกอบ 31 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 2 โดยการปรับค่าจำนวนมต 10, 20, 30, 40 และ 50 ผลลัพธ์ของการค้นหาคำตอบที่ดีที่สุด พบว่า แนวโน้มการลู่เข้าสู่คำตอบของการหาค่าความน่าเชื่อถือที่เหมาะสม ใช้จำนวนรอบในการค้นหาใกล้เคียงกันโดยประมาณคือ 30 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหาแสดงในตาราง 18

ตาราง 18 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 2

จำนวนมต	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	6.32	0.915042	0.915042	0.915042	0
20	59.73	0.912294	0.915042	0.91476	0.000869
30	7.37	0.915042	0.915042	0.915042	0
40	7.53	0.915042	0.915042	0.915042	0
50*	5.53	0.915042	0.915042	0.915042	0



จากตาราง 18 พบว่า จากการทดสอบอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 2 จำนวน 10 ครั้ง สามารถประมวลผลค้นหาคำตอบค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงสุด คือ 0.915042 จากการตั้งค่าจำนวนมด 10, 30, 40 และ 50 ซึ่งมีค่าเฉลี่ยเท่ากับ 0.915042 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.00 และพบว่า การตั้งค่าจำนวนมด 50 ตัว ใช้เวลาเฉลี่ยในการค้นหาข้อมูลเร็วที่สุด เท่ากับ 5.53 วินาที

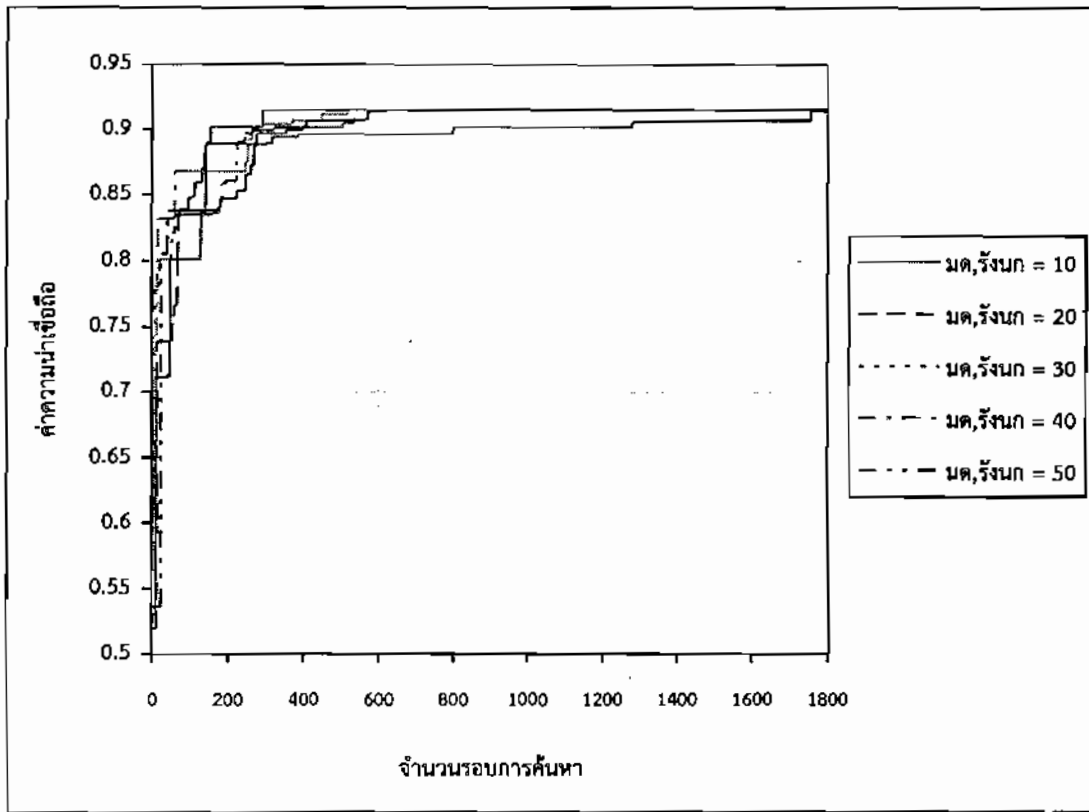
ตาราง 19 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 2

จำนวนมด	เทคโนโลยีที่เลือก	ค่าความน่าเชื่อถือสูงสุด	ราคาต้นทุนรวม (\$)
10	2-3-3-4-2-3-3-2-4-1-2-4-4-3-1	0.915042	895
20	2-3-3-4-2-3-3-2-4-1-2-4-4-3-1	0.915042	895
30	2-3-3-4-2-3-3-2-4-1-2-4-4-3-1	0.915042	895
40	3-3-3-4-2-3-2-2-4-1-2-4-4-3-1	0.915042	895
50	3-3-3-4-2-3-3-2-4-1-2-4-3-3-1	0.915042	895

จากตาราง 19 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 2 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด โดยการปรับค่าจำนวนมด 10, 20, 30, 40 และ 50 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุดที่แตกต่างกัน 3 ค่า แต่มีราคาต้นทุนรวม \$895 ต่ำกว่างบประมาณที่ตั้งไว้ \$900 ปรากฏในการตั้งค่าจำนวนมดเท่ากับ 30, 40 และ 50 คือ 2-3-3-4-2-3-3-2-4-1-2-4-4-3-1, 3-3-3-4-2-3-2-2-4-1-2-4-4-3-1 และ 3-3-3-4-2-3-3-2-4-1-2-4-3-3-1 ตามลำดับ

4.3.2.2 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight





ภาพประกอบ 32 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 2

จากภาพประกอบ 32 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 2 โดยการปรับค่าจำนวนมดและรังนก 10, 20, 30, 40 และ 50 ผลลัพธ์ของการค้นหาคำตอบที่ดีที่สุด พบว่า แนวโน้มการลู่เข้าสู่คำตอบของการหาค่าความน่าเชื่อถือที่เหมาะสมใช้จำนวนรอบในการค้นหาต่ำสุดโดยประมาณจำนวน 600 รอบ และสูงสุดโดยประมาณ 1,700 รอบ และพบว่า การตั้งค่าจำนวนมดและรังนก เท่ากับ 10 ไม่สามารถค้นหาคำตอบที่เหมาะสมที่สุดได้ภายใน 2,000 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 20



ตาราง 20 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 2

จำนวนมดและ รังนกดูเหว่า	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	89.42	0.886055	0.914126	0.902871	0.008284
20	140.13	0.899148	0.915042	0.910658	0.006093
30	174.84	0.901946	0.915042	0.911679	0.005397
40*	248.77	0.901969	0.915042	0.911792	0.004697
50	327.05	0.903167	0.915042	0.911638	0.004479

จากตาราง 20 พบว่า จากการทดลองอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 2 จำนวน 10 ครั้ง โดยการตั้งค่าจำนวนมดและรังนก 20, 30, 40 และ 50 สามารถประมวลผลค้นหาคำตอบค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงสุด คือ 0.915042 ซึ่งเมื่อพิจารณาแล้วพบว่า การตั้งค่าจำนวนมดและรังนก เท่ากับ 40 ส่งผลให้มีค่าความน่าเชื่อถือที่เหมาะสมโดยเฉลี่ยสูงสุด คือ 0.911792 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.004697 ใช้เวลาเฉลี่ยในการค้นหาข้อมูล คือ 248.77 วินาที

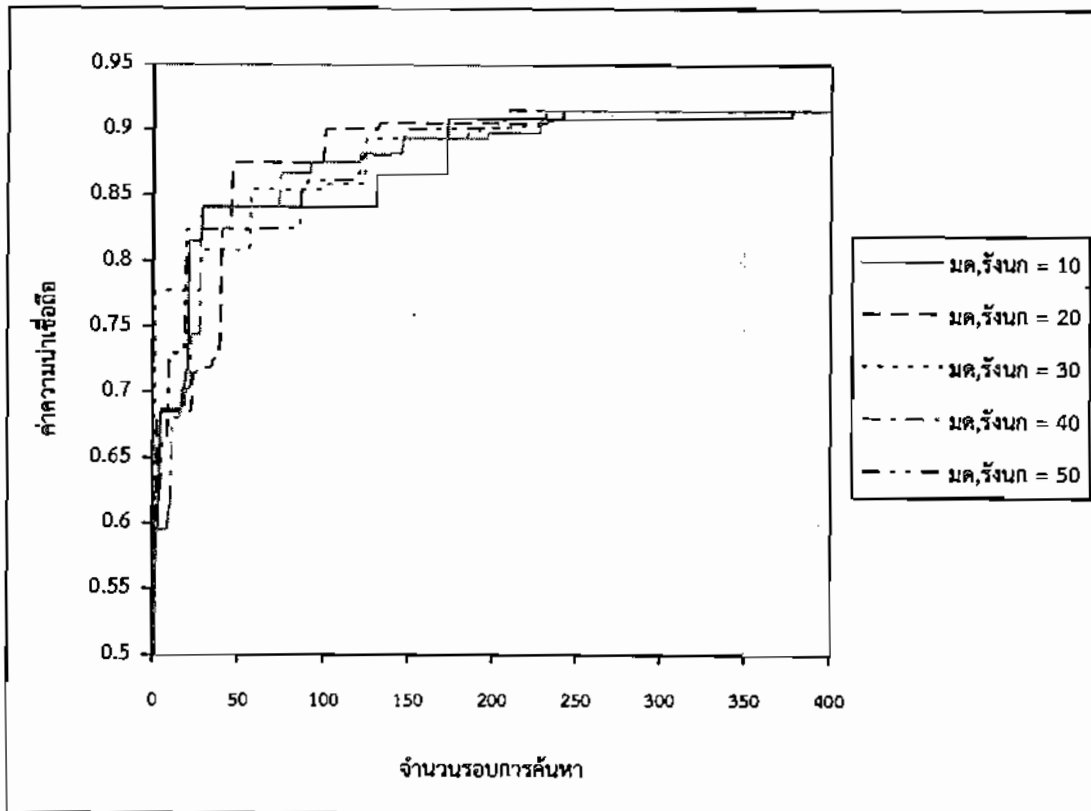
ตาราง 21 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 2

จำนวนมดและ รังนกดูเหว่า	เทคโนโลยีที่เลือก	ค่าความ น่าเชื่อถือสูงสุด	ราคาต้นทุน รวม (\$)
10	2-3-4-4-2-3-2-2-4-1-2-4-4-3-1	0.914126	895
20	3-3-3-4-2-3-2-2-4-1-2-4-4-3-1	0.915042	895
30	2-3-3-4-2-3-3-2-4-1-2-4-4-3-1	0.915042	895
40	2-3-3-4-3-3-2-2-4-1-2-4-4-3-1	0.915042	900
50	3-3-3-4-2-3-3-2-4-1-2-4-3-3-1	0.915042	895

จากตาราง 21 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 2 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด โดยการปรับค่าจำนวนมดและรังนก 10, 20, 30, 40 และ 50 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุดที่แตกต่างกัน 3 ค่า แต่มีราคาต้นทุนรวม \$895 ต่ำกว่างบประมาณที่ตั้งไว้ \$900 ปรากฏในการตั้งค่าจำนวนมดและรังนก เท่ากับ 20, 30 และ 50



4.3.2.3 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS



ภาพประกอบ 33 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 2

จากภาพประกอบ 33 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษาที่ 2 โดยการปรับค่าจำนวนมดและรังนก 10,20,30,40 และ 50 ผลลัพธ์ของการค้นหาค่าตอบที่ดีที่สุด พบว่า แนวโน้มการลู่เข้าสู่ค่าตอบของการหาค่าความน่าเชื่อถือที่เหมาะสม ใช้จำนวนรอบในการค้นหาใกล้เคียงกันโดยประมาณ อยู่ในช่วง 250 - 350 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 22



ตาราง 22 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 2

จำนวนมดและ รังนกคุเหว่า	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	41.12	0.912294	0.915042	0.914767	0.000869
20*	48.68	0.915042	0.915042	0.915042	0
30	86.36	0.915042	0.915042	0.915042	0
40	119.54	0.915042	0.915042	0.915042	0
50	116.55	0.915042	0.915042	0.915042	0

จากตาราง 22 พบว่า จากการทดลองอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษาที่ 2 จำนวน 10 ครั้ง โดยการตั้งค่าจำนวนมดและรังนก 20, 30, 40 และ 50 สามารถประมวลผลค้นหาคำตอบหาค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงสุด มีค่าเฉลี่ย เท่ากับ 0.915042 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0 ซึ่งเมื่อพิจารณาแล้วพบว่า การตั้งค่าจำนวนมดและรังนก เท่ากับ 20 ใช้เวลาเฉลี่ยในการค้นหาข้อมูลเร็วที่สุด คือ 48.68 วินาที

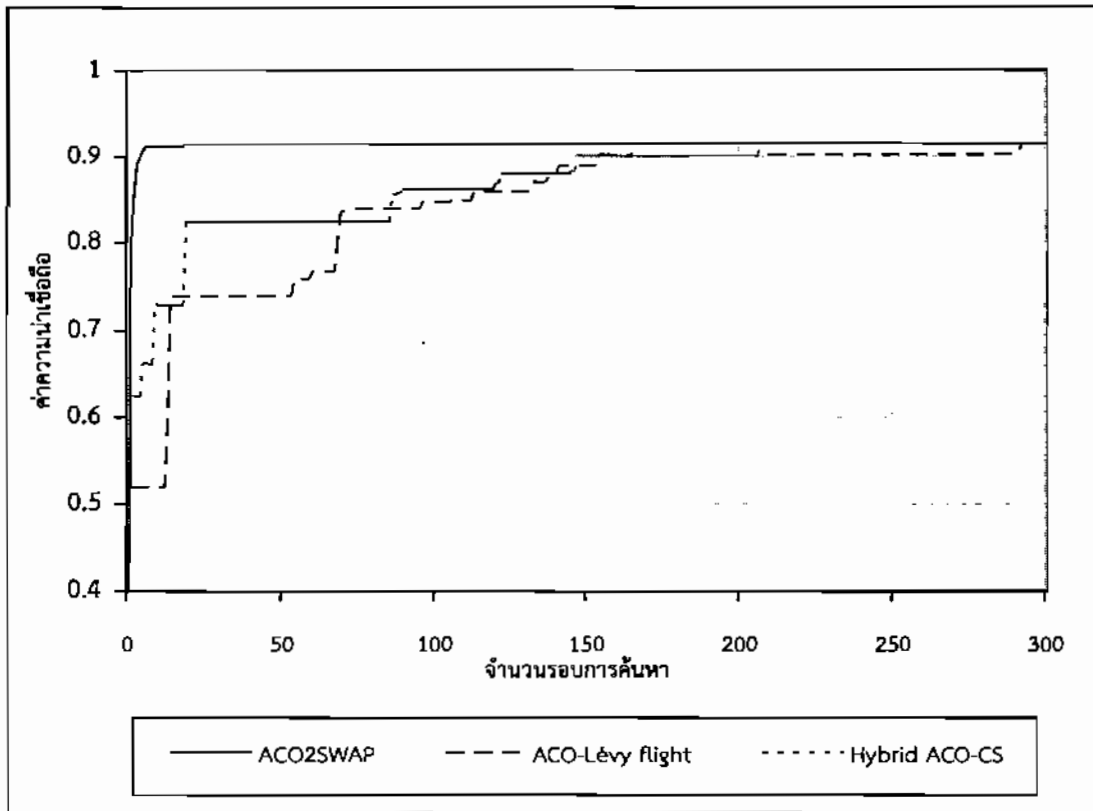
ตาราง 23 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 2

จำนวนมดและ รังนกคุเหว่า	เทคโนโลยีที่เลือก	ค่าความ น่าเชื่อถือสูงสุด	ราคาต้นทุน รวม (\$)
10	3-3-3-4-2-3-3-2-4-1-2-4-3-3-1	0.915042	895
20	2-3-3-4-2-3-3-2-4-1-2-4-4-3-1	0.915042	895
30	3-3-3-4-2-3-3-2-4-1-2-4-3-3-1	0.915042	895
40	2-3-3-4-2-3-3-2-4-1-2-4-4-3-1	0.915042	895
50	3-3-3-4-2-3-2-2-4-1-2-4-4-3-1	0.915042	895

จากตาราง 23 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษาที่ 2 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด โดยการปรับค่าจำนวนมดและรังนก 10, 20, 30, 40 และ 50 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุดที่แตกต่างกัน 3 ค่า แต่มีราคาต้นทุนรวม \$895 ต่ำกว่างบประมาณที่ตั้งไว้ \$900 ปรากฏในการตั้งค่าจำนวนมดและรังนก ชุดแรก คือ 10, 30 ชุดที่ 2 คือ 20, 40 และสุดท้าย คือ 50 ตามลำดับ



4.3.2.4 เปรียบเทียบผลการทดสอบระหว่างอัลกอริทึม ACO2SWAP, ACO-Lévy flight และ Hybrid ACO-CS กรณีศึกษาที่ 2



ภาพประกอบ 34 ผลการเปรียบเทียบระหว่างอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 2

จากภาพประกอบ 34 เป็นการนำเสนอผลการเปรียบเทียบการค้นหาคำตอบที่ดีที่สุดของอัลกอริทึม ACO2SWAP, ACO-Lévy flight และ Hybrid ACO-CS สำหรับกรณีศึกษาที่ 2 ซึ่งมีแนวโน้มการเข้าสู่คำตอบของอัลกอริทึม ACO2SWAP ใช้จำนวนรอบในการค้นหาคำตอบ น้อยที่สุดโดยประมาณคือ 30 รอบ รองลงมาคือ อัลกอริทึม Hybrid ACO-CS ใช้จำนวนรอบในการค้นหาคำตอบโดยประมาณคือ 200 รอบ และลำดับสุดท้าย คือ อัลกอริทึม ACO-Lévy flight ใช้จำนวนรอบในการค้นหาคำตอบ โดยประมาณ 280 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 24



ตาราง 24 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 2

อัลกอริทึม	เวลาเฉลี่ย (s)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
ACO2SWAP*	5.53	0.915042	0.915042	0.915042	0
ACO-Lévy flight	248.77	0.901969	0.915042	0.911792	0.004697
Hybrid ACO-CS*	48.68	0.915042	0.915042	0.915042	0

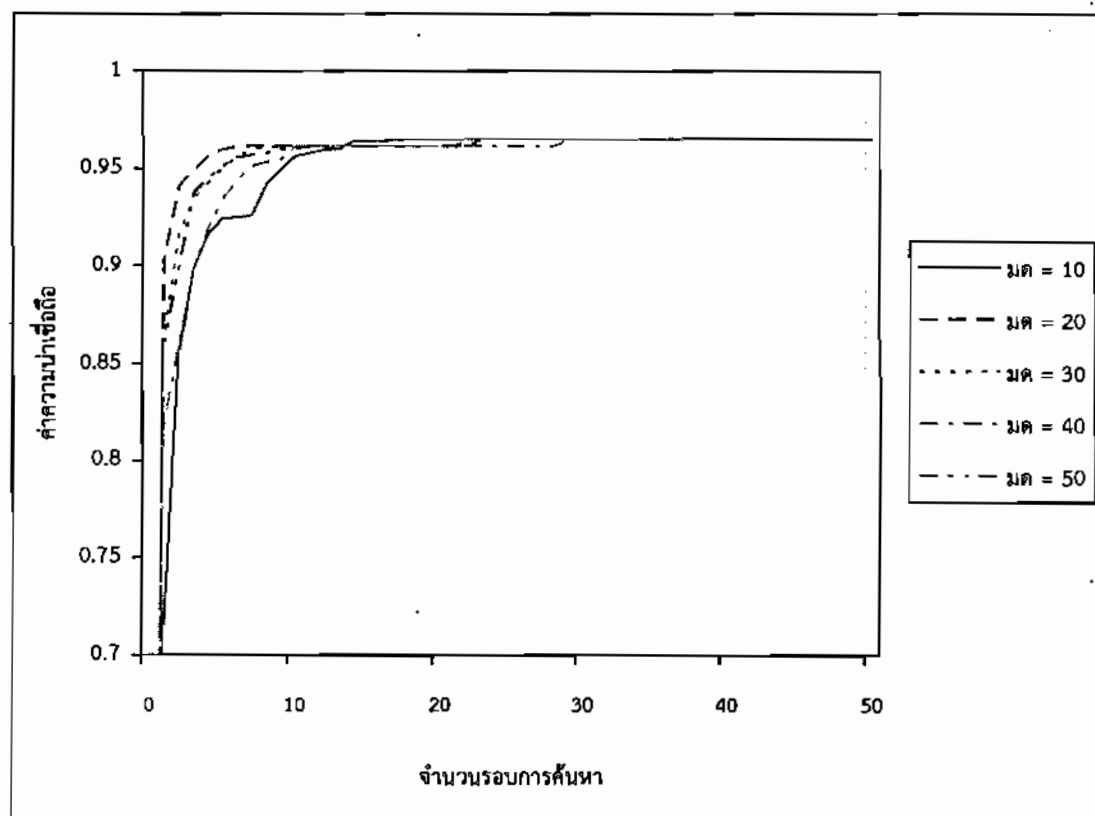
จากตาราง 24 โดยภาพรวมผลลัพธ์การค้นหาคำตอบที่ดีที่สุดสำหรับกรณีศึกษาที่ 2 ของแต่ละอัลกอริทึม พบว่า อัลกอริทึม ACO2SWAP และ Hybrid ACO-CS สามารถหาค่าความน่าเชื่อถือที่เหมาะสมที่สุด ดีกว่าอัลกอริทึม ACO-Lévy flight แต่และอัลกอริทึม ACO2SWAP ใช้เวลาเฉลี่ยในการค้นพบคำตอบที่ดีกว่า Hybrid ACO-CS ซึ่งเมื่อพิจารณาจากค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐาน แสดงให้เห็นว่าอัลกอริทึม ACO2SWAP และ Hybrid ACO-CS มีความแม่นยำสูงสำหรับการประมวลผลระบบอนุกรมที่มี 15 ระบบย่อย 80 ตัวแปร

4.3.3 กรณีศึกษาที่ 3

ระบบอนุกรมที่มี 15 ระบบย่อย 100 ตัวแปร กำหนดเงื่อนไขราคารวมต้นทุน เท่ากับ \$1000

4.3.3.1 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP





ภาพประกอบ 35 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 3

จากภาพประกอบ 35 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 3 โดยการปรับค่าจำนวนมด 10, 20, 30, 40 และ 50 ผลลัพธ์ของการค้นหาคำตอบที่ดีที่สุด พบว่า แนวโน้มการลู่เข้าสู่คำตอบของการหาค่าความน่าเชื่อถือที่เหมาะสม ใช้จำนวนรอบในการค้นหาใกล้เคียงกันโดยประมาณคือ 30 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหาแสดงในตารางที่ 25

ตาราง 25 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 3

จำนวนมด	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	8.03	0.965134	0.965134	0.965134	0
20	9.64	0.965134	0.965134	0.965134	0
30*	6.80	0.965134	0.965134	0.965134	0
40	8.07	0.965134	0.965134	0.965134	0
50	7.24	0.965134	0.965134	0.965134	0



จากตาราง 25 พบว่า จากการทดลองอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 3 จำนวน 10 ครั้ง การตั้งค่าจำนวนมด 10, 20, 30, 40 และ 50 สามารถประมวลผลค้นหาค่าตอบหาค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงสุดเท่ากันทั้งหมด คือ 0.965134 และมีค่าส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.00 เท่ากันทั้งหมดเช่นกัน โดยภาพรวมใช้เวลาเฉลี่ยในการค้นหาข้อมูลอยู่ระหว่าง 6-10 วินาที

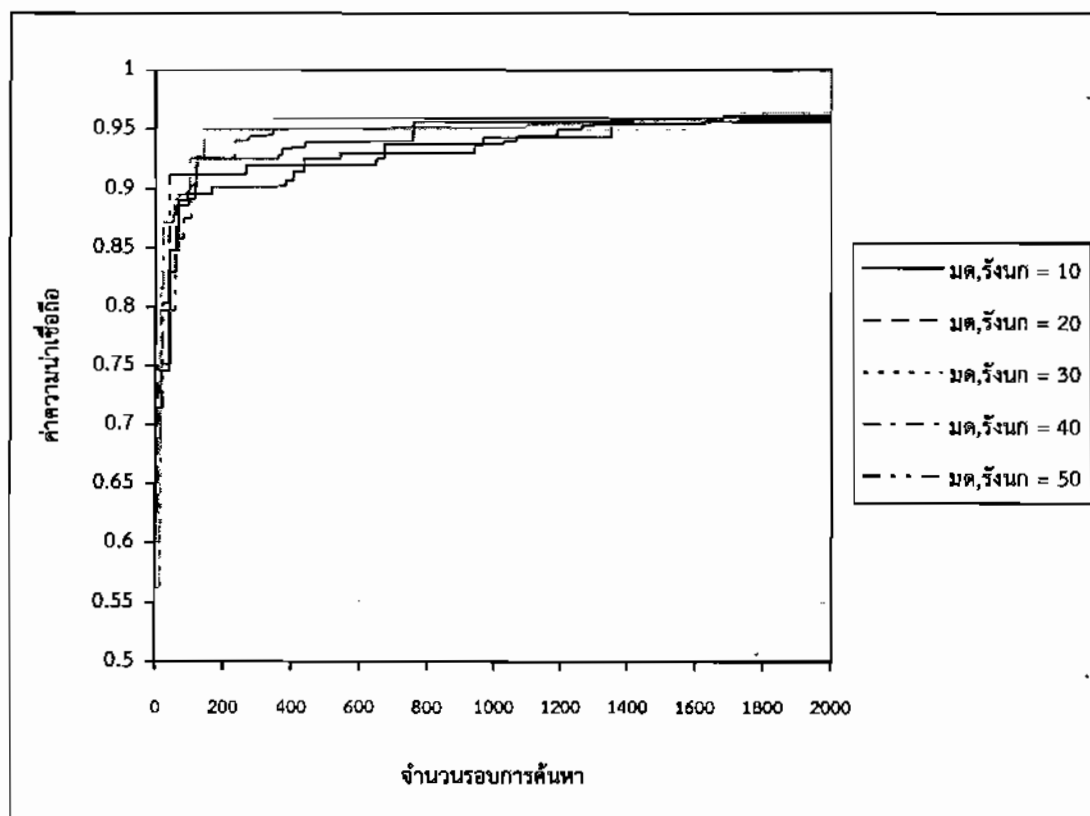
ตาราง 26 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 3

จำนวนมด	เทคโนโลยีที่เลือก	ค่าความน่าเชื่อถือสูงสุด	ราคาต้นทุนรวม (\$)
10	3-3-4-4-3-3-2-2-3-2-2-4-4-4-2	0.965134	995
20	3-3-4-4-3-3-2-2-3-2-2-4-4-4-2	0.965134	995
30	3-3-4-4-3-3-2-2-3-2-2-4-4-4-2	0.965134	995
40	3-3-4-4-3-3-2-2-3-2-2-4-4-4-2	0.965134	995
50	3-3-4-4-3-3-2-2-3-2-2-4-4-4-2	0.965134	995

จากตาราง 26 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 3 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด โดยการปรับค่าจำนวนมด 10, 20, 30, 40 และ 50 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุด เป็นชุดเดียวกันทั้งหมด มีราคาต้นทุนรวม \$995 ต่ำกว่างบประมาณที่ตั้งไว้ \$1000 เทคโนโลยีที่เลือก คือ 3-3-4-4-3-3-2-2-3-2-2-4-4-4-2

4.3.3.2 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight





ภาพประกอบ 36 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 3

จากภาพประกอบ 36 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 3 โดยการปรับค่าจำนวนมดและรังนก 10, 20, 30, 40 และ 50 ผลลัพธ์ของการค้นหาคำตอบที่ดีที่สุด พบว่า ไม่สามารถค้นหาคำตอบที่เหมาะสมที่สุดได้ภายใน 2,000 รอบ เมื่อเทียบกับอัลกอริทึม ACO2SWAP ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตารางที่ 27

ตาราง 27 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 3

จำนวนมดและ รังนกคู่เหว่า	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	89.97	0.935545	0.958067	0.946887	0.007194
20	170.05	0.937275	0.960667	0.951029	0.007252
30	250.06	0.943860	0.963202	0.953443	0.005847
40*	329.33	0.948178	0.964070	0.956515	0.004523
50	409.60	0.948105	0.964070	0.955490	0.004838



จากตาราง 27 พบว่า จากการทดลองอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 3 จำนวน 10 ครั้ง โดยการตั้งค่าจำนวนมดและรังนก เท่ากับ 40 สามารถประมวลผลค้นหาคำตอบ ค่าความน่าเชื่อถือที่เหมาะสมได้เฉลี่ยสูงสุด คือ 0.956515 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.004523 ใช้เวลาเฉลี่ยในการค้นหาข้อมูล คือ 329.33 วินาที

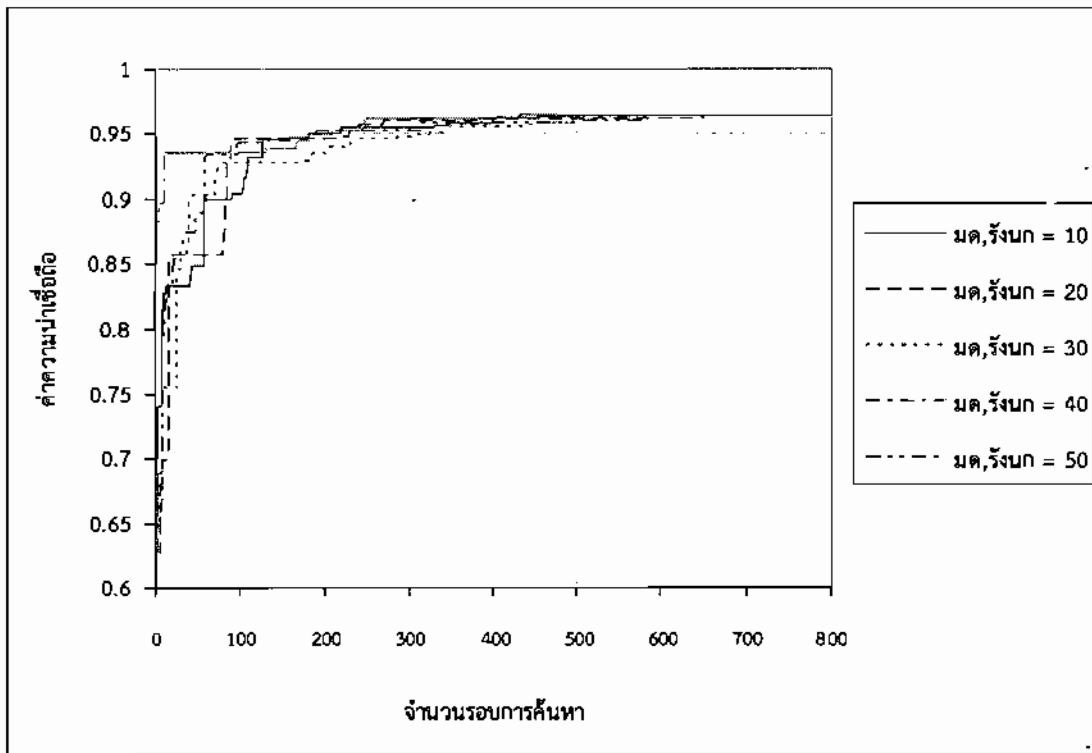
ตาราง 28 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 3

จำนวนมดและ รังนกดูเหว่า	เทคโนโลยีที่เลือก	ค่าความ น่าเชื่อถือสูงสุด	ราคาต้นทุน รวม (\$)
10	3-3-4-4-4-3-2-2-3-2-3-4-4-1	0.958067	1000
20	3-3-3-5-3-3-3-2-4-1-2-4-4-4-2	0.960667	1000
30	3-3-4-4-3-3-3-2-3-1-2-4-4-4-2	0.963202	990
40*	3-3-4-5-3-3-3-2-3-2-2-4-4-4-1	0.964070	995
50	3-3-4-5-3-3-3-2-3-1-2-4-4-4-2	0.964070	1000

จากตาราง 28 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 3 ซึ่งจากการปรับค่าจำนวนมดและรังนก เท่ากับ 40 ทำให้ได้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด เท่ากับ 0.964070 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุด คือ 3-3-4-5-3-3-3-2-3-2-2-4-4-4-1 มีราคาต้นทุนรวม \$995 ต่ำกว่างบประมาณที่ตั้งไว้ \$1000 และเมื่อเทียบกับค่าจำนวนมดและรังนก เท่ากับ 50

4.3.3.3 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS





ภาพประกอบ 37 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 3

จากภาพประกอบ 37 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษาที่ 3 โดยการปรับค่าจำนวนมดและรังนก 10,20,30,40 และ 50 ผลลัพธ์ของการค้นหาคำตอบที่ดีที่สุด พบว่า แนวโน้มการลู่เข้าสู่คำตอบของการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุด คือ การตั้งค่าจำนวนมดและรังนก เท่ากับ 40 โดยใช้จำนวนรอบในการค้นหาโดยประมาณ 450 รอบ และการตั้งค่าจำนวนมดและรังนก เท่ากับ 10, 20, 30 และ 50 ใช้จำนวนรอบในการค้นหาใกล้เคียงกัน โดยประมาณ คือ 800 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตารางที่ 29

ตาราง 29 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 3

จำนวนมดและ รังนกดูหว่า	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	87.44	0.964070	0.965134	0.964815	0.000514
20	180.77	0.964070	0.965134	0.964834	0.000484
30*	183.66	0.965134	0.965134	0.965134	0
40	286.40	0.964070	0.965134	0.965028	0.000337
50	315.25	0.965134	0.965134	0.965134	0



จำนวน 10 ครั้ง การหาค่าความน่าเชื่อถือและวงเงินเท่ากับ 50 และ 50 ได้มีการประเมินผลเกณฑ์ที่หาค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงที่สุด เท่ากับ 0.965134 โดยมีค่าเฉลี่ย เท่ากับ 0.965134 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0 เมื่อพิจารณาในด้านเวลา พบว่า การตั้งค่าจำนวนมดและรังนก เท่ากับ 30 ใช้เวลาเฉลี่ยในการค้นหาข้อมูล 180.77 วินาที น้อยกว่า พบว่า การตั้งค่าจำนวนมดและรังนก เท่ากับ 50

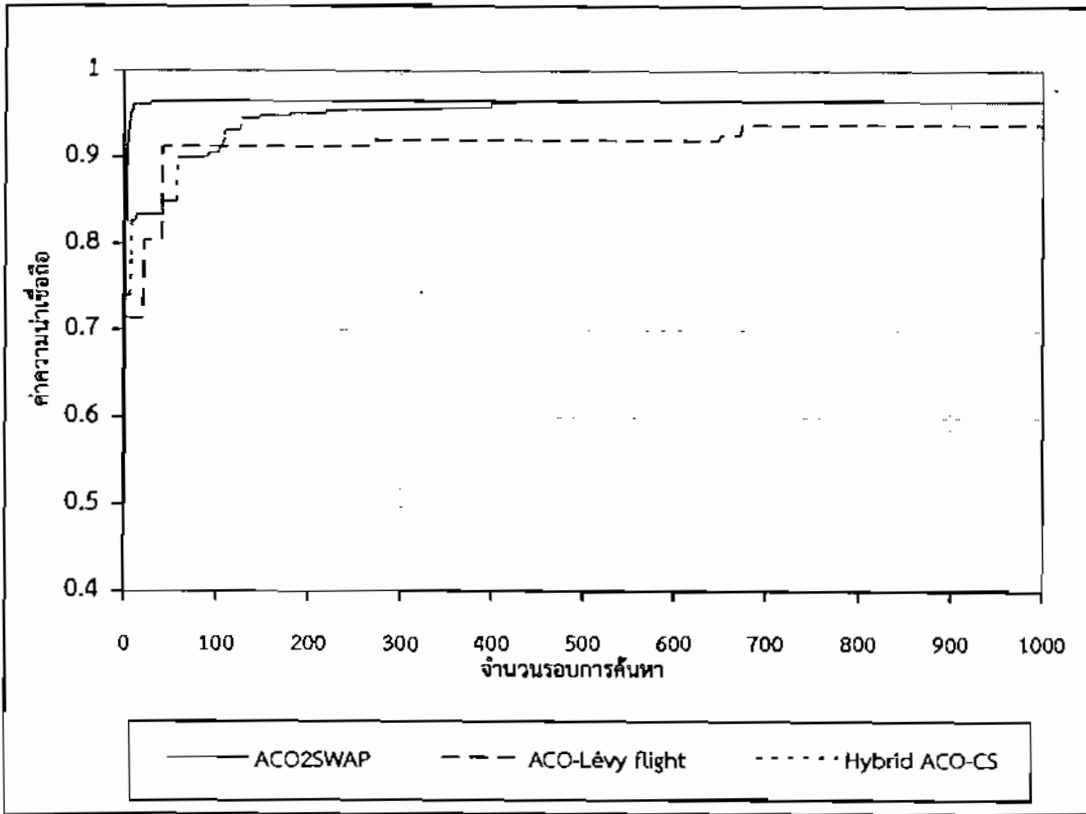
ตาราง 30 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 3

จำนวนมดและรังนกคู่เหว่า	เทคโนโลยีที่เลือก	ค่าความน่าเชื่อถือสูงสุด	ราคาต้นทุนรวม (\$)
10	3-3-4-4-3-3-2-2-3-2-2-4-4-4-2	0.965134	995
20	3-3-4-4-3-3-2-2-3-2-2-4-4-4-2	0.965134	995
30	3-3-4-4-3-3-2-2-3-2-2-4-4-4-2	0.965134	995
40	3-3-4-4-3-3-2-2-3-2-2-4-4-4-2	0.965134	995
50	3-3-4-4-3-3-2-2-3-2-2-4-4-4-2	0.965134	995

จากตาราง 30 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษาที่ 3 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด โดยการปรับค่าจำนวนมดและรังนก 10, 20, 30, 40 และ 50 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุดเป็นชุดเดียวกันทั้ง 5 ชุด แต่มีราคาต้นทุนรวม \$995 ต่ำกว่างบประมาณที่ตั้งไว้ \$1000

4.3.3.4 เปรียบเทียบผลการทดสอบระหว่างอัลกอริทึม ACO2SWAP, ACO-Lévy flight และ Hybrid ACO-CS กรณีศึกษาที่ 3





ภาพประกอบ 38 ผลการเปรียบเทียบระหว่างอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 3

จากภาพประกอบ 38 เป็นการนำเสนอผลการเปรียบเทียบการค้นหาคำตอบที่ดีที่สุดของอัลกอริทึม ACO2SWAP, ACO-Lévy flight และ Hybrid ACO-CS สำหรับกรณีศึกษาที่ 3 ซึ่งมีแนวโน้มการลู่เข้าสู่คำตอบของอัลกอริทึม ACO2SWAP ใช้จำนวนรอบในการค้นหาคำตอบ น้อยที่สุดโดยประมาณคือ 30 รอบ รองลงมาคือ อัลกอริทึม Hybrid ACO-CS ใช้จำนวนรอบในการค้นหาคำตอบโดยประมาณคือ 450 รอบ และอัลกอริทึม ACO-Lévy flight พบว่า ยังไม่สามารถค้นหาคำตอบที่เหมาะสมที่สุดได้ภายใน 2,000 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 31

ตาราง 31 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 3

อัลกอริทึม	เวลาเฉลี่ย (s)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
ACO2SWAP*	6.80	0.965134	0.965134	0.965134	0
ACO-Lévy flight	329.33	0.948178	0.964070	0.956515	0.004523
Hybrid ACO-CS	183.66	0.965134	0.965134	0.965134	0

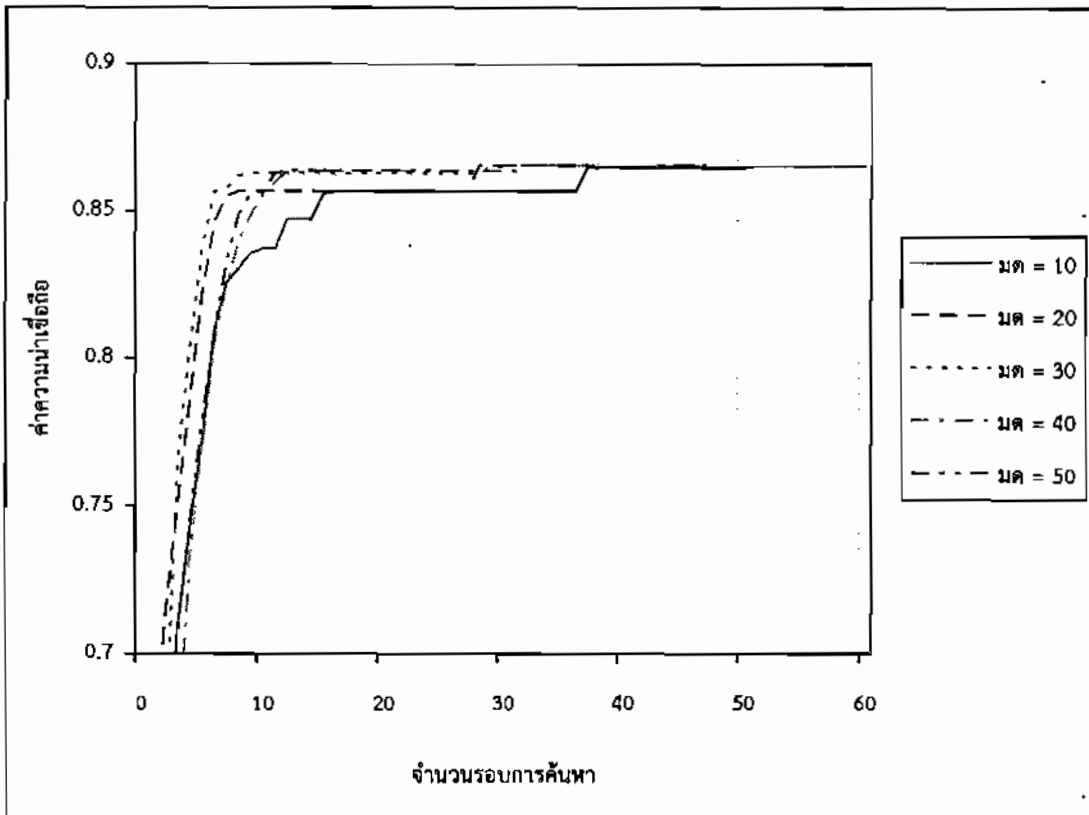


จากตาราง 31 โดยภาพรวมผลลัพธ์การค้นหาคำตอบที่ดีที่สุดสำหรับกรณีศึกษาที่ 3 ของแต่ละอัลกอริทึม พบว่า อัลกอริทึม ACO2SWAP และ Hybrid ACO-CS สามารถหาค่าความน่าเชื่อถือที่เหมาะสมที่สุด ดีกว่าอัลกอริทึม ACO-Lévy flight แต่และอัลกอริทึม ACO2SWAP ใช้เวลาเฉลี่ยในการค้นพบคำตอบที่ดีกว่า Hybrid ACO-CS ซึ่งเมื่อพิจารณาจากค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐาน แสดงให้เห็นว่าอัลกอริทึม ACO2SWAP และ Hybrid ACO-CS มีความแม่นยำสูงสำหรับการประมวลผลระบบอนุกรมที่มี 15 ระบบย่อย 100 ตัวแปร

4.3.4 กรณีศึกษาที่ 4

ระบบอนุกรมที่มี 25 ระบบย่อย 166 ตัวแปร กำหนดเงื่อนไขไรคารวมต้นทุน เท่ากับ \$1400

4.3.4.1 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP



ภาพประกอบ 39 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 4

คำตอบที่ดีที่สุด พบว่า แนวโน้มการลู่เข้าสู่คำตอบของการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุด ใช้จำนวนรอบในการค้นหาใกล้เคียงกันโดยประมาณ ระหว่าง 30 - 60 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 32

ตาราง 32 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 4

จำนวนมด	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	46.05	0.865439	0.865439	0.865439	0
20	57.30	0.865439	0.865439	0.865439	0
30	49.25	0.865439	0.865439	0.865439	0
40*	39.72	0.865439	0.865439	0.865439	0
50	194.04	0.864660	0.865439	0.865361	0.000246

จากตาราง 32 พบว่า จากการทดลองอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 4 จำนวน 10 ครั้ง การตั้งค่าจำนวนมด 10, 20, 30 และ 40 สามารถประมวลผลค้นหาคำตอบหาค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงสุดเท่ากัน คือ 0.865439 และมีค่าส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.00 เมื่อพิจารณาโดยภาพรวม พบว่า การตั้งค่าจำนวนมด เท่ากับ 40 ใช้เวลาเฉลี่ยในการค้นหาข้อมูลเร็วที่สุดคือ 39.72 วินาที

ตาราง 33 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 4

จำนวนมด	เทคโนโลยีที่เลือก	ค่าความน่าเชื่อถือสูงสุด	ราคาต้นทุนรวม (\$)
10	2-3-3-4-2-3-2-2-3-2-2-3-4-4-1-2-3-3-5-2-3-2-2-3-1	0.865439	1400
20	2-3-3-5-2-3-2-2-3-2-2-3-4-3-1-3-3-3-4-2-3-2-2-3-1	0.865439	1400
30	2-3-3-4-2-3-2-2-3-1-2-3-4-4-1-3-3-3-5-2-3-2-2-3-1	0.865439	1395
40*	2-3-3-4-2-3-2-2-3-1-2-3-4-4-1-3-3-3-5-2-3-2-2-3-1	0.865439	1395
50	2-3-3-4-2-3-2-2-3-2-2-3-4-3-1-3-3-3-5-2-3-2-2-3-1	0.865439	1400



ตาราง 10 (ต่อ)

เลขี่ ไฟล์ท์ $\beta \alpha$	ค่า α, β อัลกอริทึมฝูงมด, ค่า pa อัลกอริทึมนกดูเหว่า									
	0.8, 0.5, $pa=0.6$	0.8, 1, $pa=0.65$	1, 0.5, $pa=0.65$	1, 0.5, $pa=0.75$	1, 0.8, $pa=0.7$	1, 1, $pa=0.75$	2, 0.5, $pa=0.6$	2, 1, $pa=0.7$	3, 0.8, $pa=0.75$	5, 0.5, $pa=0.55$
2, 0.4	0.85584	0.85611	0.85651	0.85662	0.85659	0.85644	0.85241	0.85647	0.85601	0.85217
2, 0.5	0.85665	0.85560	0.85685	0.85644	0.85586	0.85402	0.85659	0.85511	0.85586	0.85206
2, 0.6	0.85588	0.85433	0.85682	0.85667	0.85507	0.85524	0.85653	0.84749	0.85091	0.85613
2, 0.7	0.85598	0.85574	0.85584	0.85516	0.85610	0.85351	0.85236	0.85306	0.85057	0.85512
2, 0.8	0.85571	0.85501	0.85639	0.85621	0.85556	0.85327	0.85466	0.83741	0.85223	0.84532
2, 0.9	0.85639	0.85526	0.85573	0.85564	0.85565	0.85296	0.85670	0.84932	0.84657	0.85458

จากตาราง 10 พบว่า ผลการวิเคราะห์ค่าพารามิเตอร์ที่เกี่ยวข้องกับอัลกอริทึมฝูงมด คือ $\alpha = 1$, $\beta = 0.5$ อัลกอริทึมนกดูเหว่า คือ $pa = 0.65$ และเลขี่ไฟล์ท์ คือ $\beta = 2$ $\alpha = 1$, ที่ให้ผลลัพธ์ดีที่สุด

โดยสรุปจากผลการทดลองข้างต้น ค่าพารามิเตอร์ที่จะนำไปใช้ในการทดสอบประสิทธิภาพอัลกอริทึมกับระบบอนุกรม มีดังนี้

1. อัลกอริทึม ACO2SWAP ตั้งค่าพารามิเตอร์คือ ค่า $\alpha = 0.8$ และ $\beta = 1$
2. อัลกอริทึม ACO-Lévy flight และ Hybrid ACO-CS ตั้งค่าพารามิเตอร์ดังนี้
 - 2.1 อัลกอริทึมฝูงมด คือ $\alpha = 1$, $\beta = 0.5$
 - 2.2 อัลกอริทึมนกดูเหว่า คือ $pa = 0.65$
 - 2.3 เลขี่ไฟล์ท์ คือ $\beta = 2$ $\alpha = 1$

4.3 ผลการทดสอบประสิทธิภาพอัลกอริทึมกับระบบอนุกรม

การทดสอบประสิทธิภาพอัลกอริทึมกับระบบอนุกรมโดยใช้ข้อมูลตัวอย่างจากงานวิจัยที่ผ่านมา [16-17] รวม 5 ชุดตัวอย่าง ดังแสดงในภาคผนวก และใช้การตั้งค่าพารามิเตอร์จากผลการทดลองในหัวข้อที่ผ่านมา โดยการทดสอบครั้งนี้จะมีการปรับค่าเฉพาะจำนวนมดและจำนวนรังนก คือ 10, 20, 30, 40, 50 และกำหนดเงื่อนไขการหยุด คือ จำนวนรอบการค้นหาสูงสุด เท่ากับ 2000 รอบ หรือเมื่ออัลกอริทึมพบคำตอบที่เหมาะสมที่สุด ซึ่งการทดสอบอัลกอริทึม ACO2SWAP, ACO-Lévy flight และ Hybrid ACO-CS จะแบ่งเป็น 5 กรณีศึกษา ดังแสดงในตารางที่ 4 สรุปได้ดังนี้

1. กรณีศึกษาที่ 1 ระบบอนุกรมที่มี 15 ระบบย่อย 60 ตัวแปร
2. กรณีศึกษาที่ 2 ระบบอนุกรมที่มี 15 ระบบย่อย 80 ตัวแปร

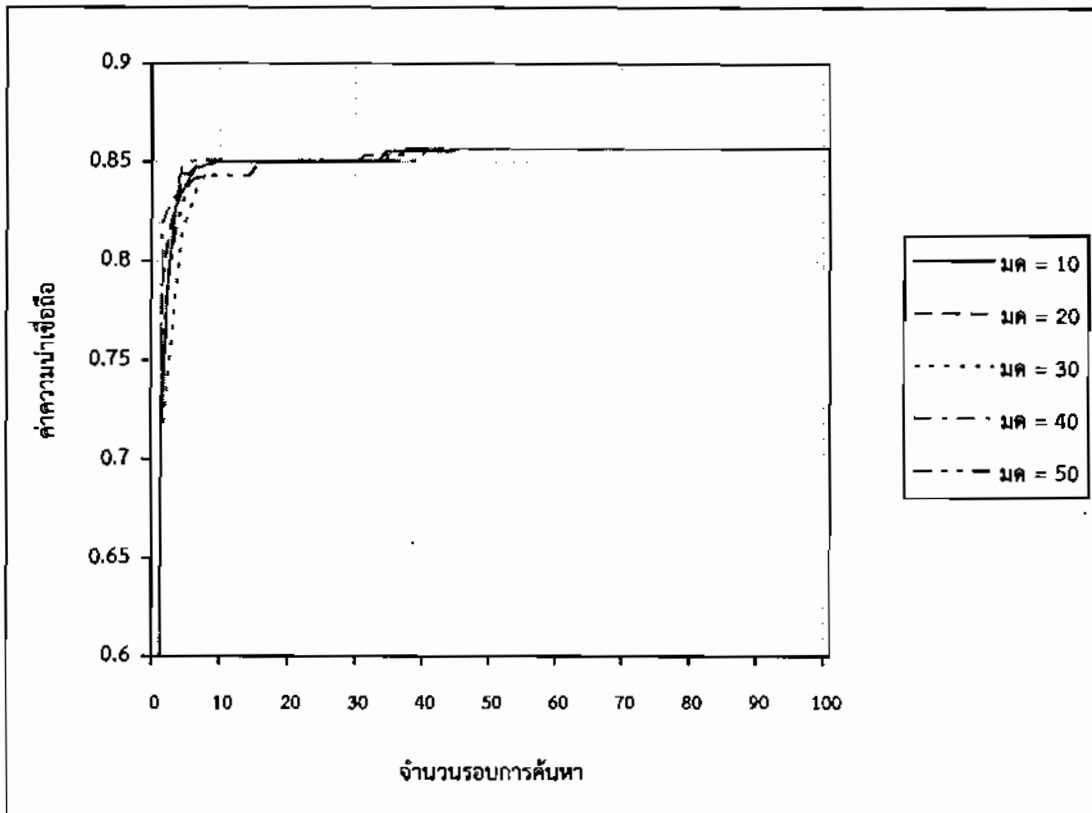


3. กรณีศึกษาที่ 3 ระบบอนุกรมที่มี 15 ระบบย่อย 100 ตัวแปร
 4. กรณีศึกษาที่ 4 ระบบอนุกรมที่มี 25 ระบบย่อย 166 ตัวแปร
 5. กรณีศึกษาที่ 5 ระบบอนุกรมที่มี 40 ระบบย่อย 266 ตัวแปร
- สามารถนำเสนอผลการทดสอบตามลำดับดังนี้

4.3.1 กรณีศึกษาที่ 1

ระบบอนุกรมที่มี 15 ระบบย่อย 60 ตัวแปร กำหนดเงื่อนไขราคารวมต้นทุน เท่ากับ \$1000

4.3.1.1 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP



ภาพประกอบ 27 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 1

จากภาพประกอบ 27 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 1 โดยการปรับค่าจำนวนมค 10,20,30,40 และ 50 ผลลัพธ์ของการค้นหาค่าตอบที่ดีที่สุด พบว่า แนวโน้มการเข้าสู่ค่าตอบของการหาค่าความน่าเชื่อถือที่เหมาะสม ใช้จำนวนรอบในการค้นหาใกล้เคียงกันโดยประมาณคือ 50 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหาแสดงในตาราง 11



ตาราง 11 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 1

จำนวนมด	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10*	23.34	0.857054	0.857054	0.857054	0
20	34.39	0.857054	0.857054	0.857054	0
30	43.22	0.857054	0.857054	0.857054	0
40	129.80	0.850098	0.857054	0.856264	0.002187
50	58.92	0.857054	0.857054	0.857054	0

จากตาราง 11 พบว่า จากการทดลองอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 1 จำนวน 10 ครั้ง สามารถประมวลผลค้นหาคำตอบหาค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงสุด คือ 0.857054 โดยการตั้งค่าจำนวนมด 10, 20, 30 และ 50 มีค่าเฉลี่ยเท่ากับ 0.857054 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.00 และพบว่า การตั้งค่าจำนวนมด 10 ตัว ใช้เวลาเฉลี่ยในการค้นหาข้อมูลเร็วที่สุด เท่ากับ 23.34 วินาที

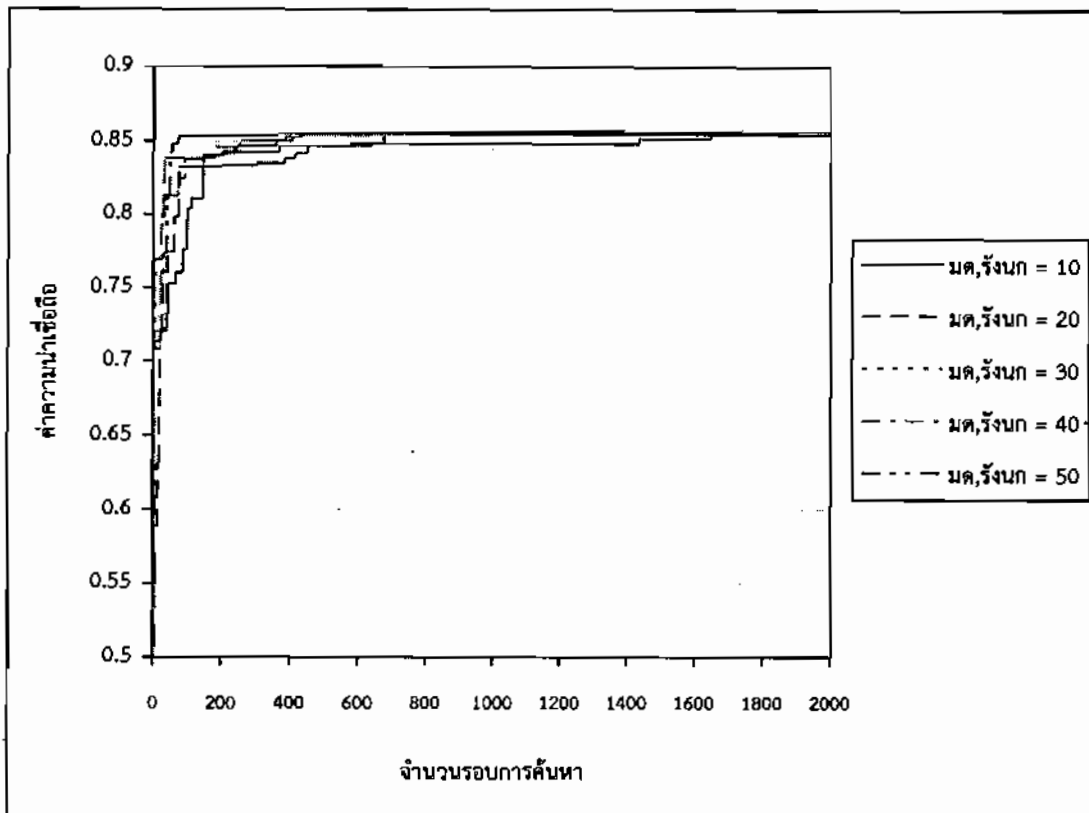
ตาราง 12 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 1

จำนวนมด	เทคโนโลยีที่เลือก	ค่าความ น่าเชื่อถือสูงสุด	ราคาต้นทุน รวม (\$)
10	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990
20	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990
30	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990
40	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990
50	3-4-5-2-3-3-2-3-2-2-2-3-4-3-2	0.857054	990

จากตาราง 12 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 1 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด โดยการปรับค่าจำนวนมด 10, 20, 30, 40 และ 50 มีผลการคัดเลือก เทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุด เป็นเทคโนโลยีเดียวกันทั้งหมด คือ 3-4-5-2-3-3-2-3-2-2-2-3-4-3-2 และใช้งบประมาณทั้งสิ้น \$990



4.3.1.2 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight



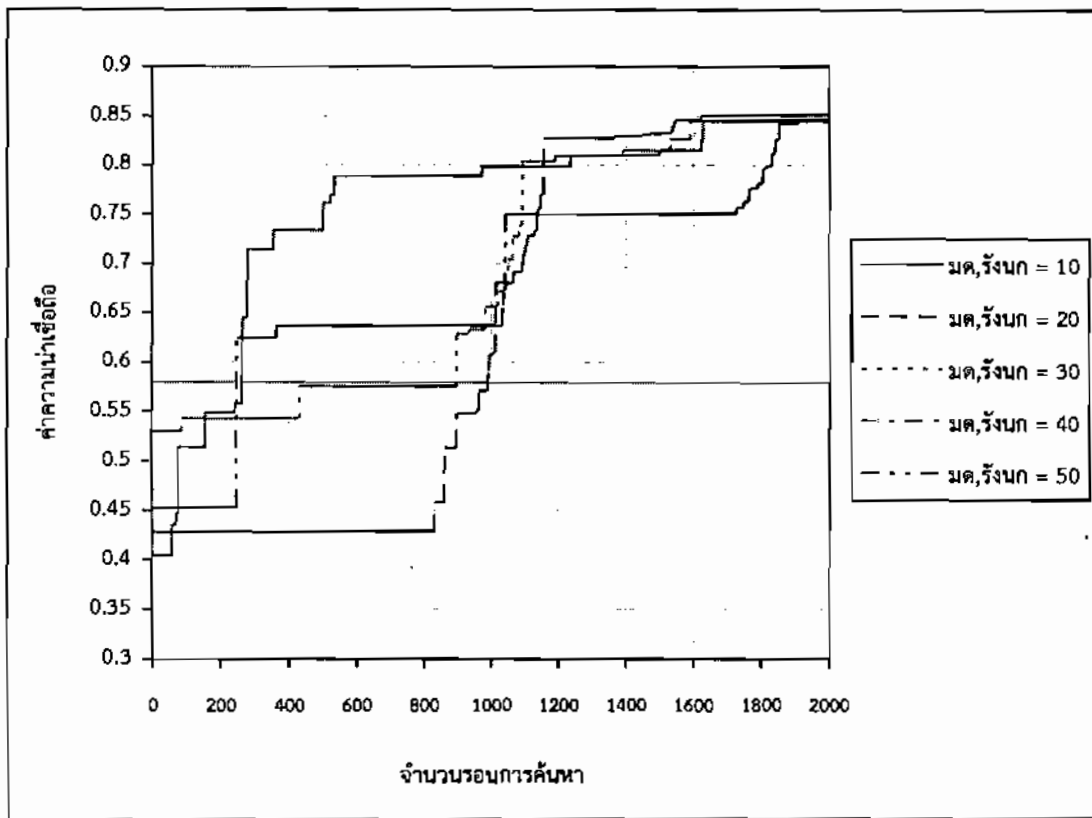
ภาพประกอบ 28 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 1

จากภาพประกอบ 28 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 1 โดยการปรับค่าจำนวนมดและรังนก 10,20,30,40 และ 50 ผลลัพธ์ของการค้นหาคำตอบที่ดีที่สุด พบว่า แนวโน้มการลู่เข้าสู่คำตอบของการหาค่าความน่าเชื่อถือที่เหมาะสม ใช้จำนวนรอบในการค้นหาต่ำสุดโดยประมาณจำนวน 1,000 รอบ และสูงสุดโดยประมาณ 1,800 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 13



จากตาราง 33 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 4 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด โดยการปรับค่าจำนวนมด 30 และ 40 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุดเป็นชุดเดียวกันทั้งหมด มีราคาต้นทุนรวม \$1395 ต่ำกว่างบประมาณที่ตั้งไว้ \$1400 เทคโนโลยีที่เลือก คือ 2-3-3-4-2-3-2-2-3-1-2-3-4-4-1-3-3-3-5-2-3-2-2-3-1

4.3.4.2 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight



ภาพประกอบ 40 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 4

จากภาพประกอบ 40 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 4 โดยการปรับค่าจำนวนมดและรังนก 10, 20, 30, 40 และ 50 ผลลัพธ์ของการค้นหาคำตอบที่ดีที่สุด พบว่า ไม่สามารถค้นหาคำตอบที่เหมาะสมที่สุดได้ภายใน 2,000 รอบ เมื่อเทียบกับอัลกอริทึม ACO2SWAP ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 34



ตาราง 34 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 4

จำนวนมดและ รังนกคู่เหว่า	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	105.19	0.457247	0.844611	0.66549	0.147837
20	199.13	0.498226	0.846008	0.648748	0.129655
30*	292.96	0.546349	0.851192	0.707444	0.116789
40	386.71	0.66338	0.862714	0.66338	0.138439
50	482.12	0.516299	0.847068	0.618078	0.091077

จากตาราง 34 พบว่า จากการทดลองอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 4 จำนวน 10 ครั้ง โดยการตั้งค่าจำนวนมดและรังนก เท่ากับ 30 สามารถประมวลผลค้นหาคำตอบค่าความน่าเชื่อถือที่เหมาะสมได้เฉลี่ยสูงสุด คือ 0.707444 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.116789 ใช้เวลาเฉลี่ยในการค้นหาข้อมูล คือ 292.96 วินาที

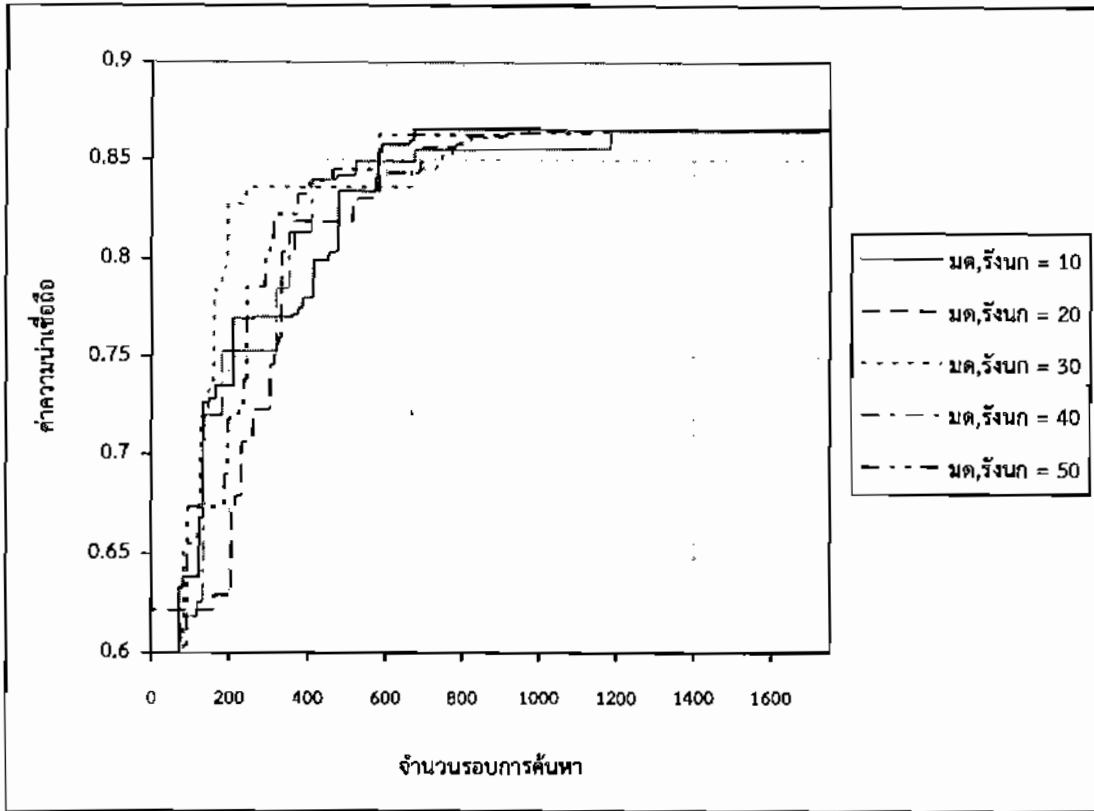
ตาราง 35 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 4

จำนวนมด และรังนก คู่เหว่า	เทคโนโลยีที่เลือก	ค่าความ น่าเชื่อถือ สูงสุด	ราคา ต้นทุน รวม (\$)
10	3-2-3-3-2-3-2-2-3-2-2-3-4-3-1-2-3-4-4-2-3-3-2-3-1	0.844611	1390
20	4-2-3-5-3-3-3-2-3-1-2-3-3-3-1-3-2-4-4-2-3-2-2-3-1	0.846008	1400
30	2-3-5-5-2-3-2-2-3-1-3-3-3-3-1-2-3-3-4-2-3-2-2-3-1	0.851192	1395
40*	3-2-3-4-2-3-2-2-3-1-2-4-4-3-1-3-3-4-4-2-3-2-2-3-1	0.862714	1400
50	2-3-3-4-2-3-3-2-3-1-3-4-3-3-1-2-2-3-4-3-3-2-2-3-1	0.847068	1385

จากตาราง 35 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 4 ซึ่งจากการปรับค่าจำนวนมดและรังนก เท่ากับ 40 ทำให้ได้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด เท่ากับ 0.862714 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุด คือ 3-2-3-4-2-3-2-2-3-1-2-4-4-3-1-3-3-4-4-2-3-2-2-3-1 มีราคาต้นทุนรวม \$1400 เท่ากับงบประมาณที่ตั้งไว้



4.3.4.3 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS



ภาพประกอบ 41 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 4

จากภาพประกอบ 41 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษาที่ 4 โดยการปรับค่าจำนวนมดและรังนก 10, 20, 30, 40 และ 50 ผลลัพธ์ของการค้นหาคำตอบที่ดีที่สุด พบว่า แนวโน้มการลู่เข้าสู่คำตอบของการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุด คือ การตั้งค่าจำนวนมดและรังนก เท่ากับ 10 โดยใช้จำนวนรอบในการค้นหาโดยประมาณ 650 รอบ และการตั้งค่าจำนวนมดและรังนก เท่ากับ 20, 30, 40 และ 50 ใช้จำนวนรอบในการค้นหาใกล้เคียงกัน โดยประมาณ คือ 1200 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 36



ตาราง 36 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 4

จำนวนมัดและ รังนกคุเหว่า	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	113.74	0.856569	0.865439	0.863505	0.002866
20	265.79	0.856569	0.865439	0.862365	0.003407
30	359.05	0.86466	0.865439	0.865049	0.000411
40	520.24	0.862714	0.865439	0.864495	0.001007
50*	540.44	0.863578	0.865439	0.865175	0.000612

จากตาราง 36 พบว่า จากการทดลองอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษาที่ 4. จำนวน 10 ครั้ง โดยการตั้งค่าจำนวนมัดและรังนก เท่ากับ 50 สามารถประมวลผลค้นหาคำตอบค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงสุด เท่ากับ 0.865439 โดยมีค่าเฉลี่ย เท่ากับ 0.865175 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.000612 ใช้เวลาเฉลี่ยในการค้นหาข้อมูล คือ 540.44 วินาที

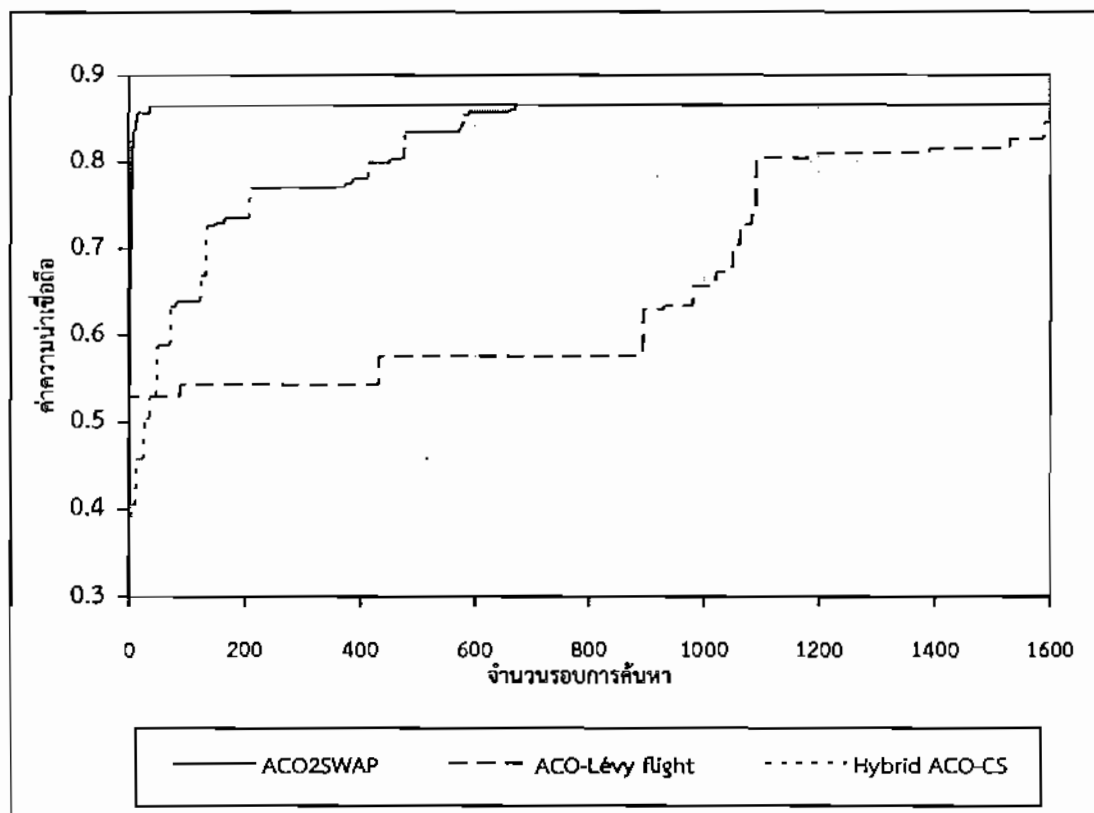
ตาราง 37 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 4

จำนวน มัดและรัง นกคุเหว่า	เทคโนโลยีที่เลือก	ค่าความ น่าเชื่อถือ สูงสุด	ราคา ต้นทุน รวม (\$)
10	3-3-3-5-2-3-2-2-3-2-2-3-4-3-1-2-3-3-4-2-3-2-2-3-1	0.865439	1400
20	2-3-3-5-2-3-2-2-3-1-2-3-4-4-1-2-3-3-4-2-3-2-2-3-2	0.865439	1400
30	3-3-3-5-2-3-2-2-3-1-2-3-4-3-1-2-3-3-4-2-3-2-2-3-2	0.865439	1400
40	3-3-3-5-2-3-2-2-3-2-2-3-3-3-1-3-3-3-4-2-3-2-2-3-1	0.865439	1400
50	3-3-3-5-2-3-2-2-3-1-2-3-3-3-1-3-3-3-4-3-3-2-2-3-1	0.865439	1400

จากตาราง 37 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษาที่ 4 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด โดยการปรับค่าจำนวนมัดและรังนก 10, 20, 30, 40 และ 50 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุดที่แตกต่างกัน ทั้ง 5 ชุด แต่มีราคาต้นทุนรวม \$1400 เท่ากับงบประมาณที่ตั้งไว้ \$1400 แสดงให้เห็นว่าสามารถคัดเลือกเทคโนโลยีได้หลายทางเลือกแต่ยังคงให้ค่าความน่าเชื่อถือสูงสุดที่เท่ากัน



4.3.4.4 เปรียบเทียบผลการทดสอบระหว่างอัลกอริทึม ACO2SWAP, ACO-Lévy flight และ Hybrid ACO-CS กรณีศึกษาที่ 4



ภาพประกอบ 42 ผลการเปรียบเทียบระหว่างอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 4

จากภาพประกอบ 42 เป็นการนำเสนอผลการเปรียบเทียบการค้นหาคำตอบที่ดีที่สุดของอัลกอริทึม ACO2SWAP, ACO-Lévy flight และ Hybrid ACO-CS สำหรับกรณีศึกษาที่ 4 ซึ่งมีแนวโน้มการเข้าสู่คำตอบของอัลกอริทึม ACO2SWAP ใช้จำนวนรอบในการค้นหาคำตอบน้อยที่สุดโดยประมาณคือ 50 รอบ รองลงมาคือ อัลกอริทึม Hybrid ACO-CS ใช้จำนวนรอบในการค้นหาคำตอบโดยประมาณคือ 650 รอบ และลำดับสุดท้าย คือ อัลกอริทึม ACO-Lévy flight ใช้จำนวนรอบในการค้นหาคำตอบ โดยประมาณ 1600 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 38

ตาราง 38 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 4

อัลกอริทึม	เวลาเฉลี่ย (s)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
ACO2SWAP*	39.72	0.865439	0.865439	0.865439	0
ACO-Lévy flight	292.96	0.546349	0.851192	0.707444	0.116789
Hybrid ACO-CS	540.44	0.863578	0.865439	0.865175	0.000612

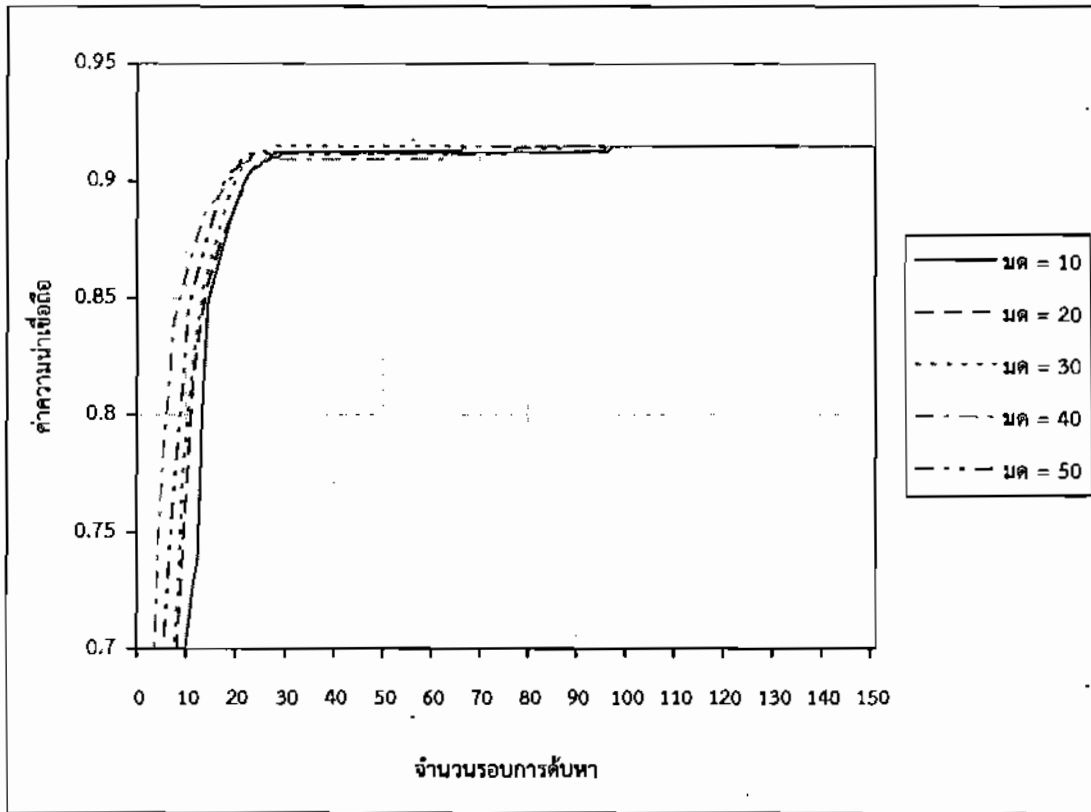
จากตาราง 38 โดยภาพรวมผลลัพธ์การค้นหาคำตอบที่ดีที่สุดสำหรับกรณีศึกษาที่ 4 ของแต่ละอัลกอริทึม พบว่า อัลกอริทึม ACO2SWAP และ Hybrid ACO-CS สามารถหาค่าความน่าเชื่อถือที่เหมาะสมที่สุด ดีกว่าอัลกอริทึม ACO-Lévy flight แต่และอัลกอริทึม ACO2SWAP ใช้เวลาเฉลี่ยในการค้นหาคำตอบที่ดีกว่า Hybrid ACO-CS ซึ่งเมื่อพิจารณาจากค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐาน แสดงให้เห็นว่าอัลกอริทึม ACO2SWAP มีความแม่นยำสูงสุด รองลงมา คือ อัลกอริทึม Hybrid ACO-CS และอัลกอริทึม ACO-Lévy flight ตามลำดับ สำหรับการประมวลผลระบบอนุกรมที่มี 25 ระบบย่อย 166 ตัวแปร

4.3.5 กรณีศึกษาที่ 5

ระบบอนุกรมที่มี 40 ระบบย่อย 266 ตัวแปร กำหนดเงื่อนไขราคารวมต้นทุน เท่ากับ \$2700

4.3.5.1 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP





ภาพประกอบ 43 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 5

จากภาพประกอบ 43 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 5 โดยการปรับค่าจำนวนมด 10, 20, 30, 40 และ 50 ผลลัพธ์ของการค้นหาคำตอบที่ดีที่สุด พบว่า แนวโน้มการเข้าสู่คำตอบของการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุด ใช้จำนวนรอบในการค้นหาใกล้เคียงกันโดยประมาณ 100 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 39

ตาราง 39 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 5

จำนวนมด	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	529.16	0.914895	0.914895	0.914895	0
20	415.82	0.914895	0.914895	0.914895	0
30*	280.92	0.914895	0.914895	0.914895	0
40	399.15	0.914895	0.914895	0.914895	0
50	679.14	0.910819	0.914895	0.914487	0.001289



จากตาราง 39 พบว่า จากการทดลองอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 5 จำนวน 10 ครั้ง การตั้งค่าจำนวนมด 10, 20, 30 และ 40 สามารถประมวลผลค้นหาคำตอบค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงสุดเท่ากัน คือ 0.914895 และมีค่าส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.00 เมื่อพิจารณาโดยภาพรวม พบว่า การตั้งค่าจำนวนมด เท่ากับ 30 ใช้เวลาเฉลี่ยในการค้นหาข้อมูลเร็วที่สุดคือ 280.92 วินาที

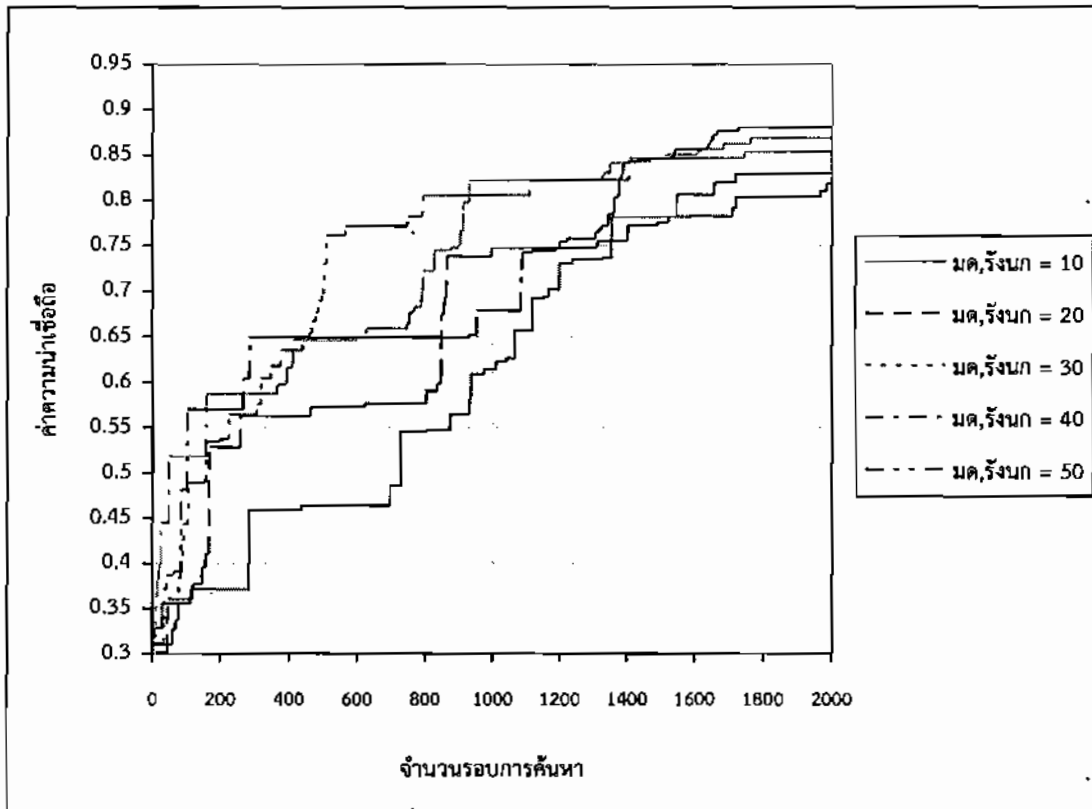
ตาราง 40 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO2SWAP กรณีศึกษาที่ 5

จำนวนมด	เทคโนโลยีที่เลือก	ค่าความน่าเชื่อถือสูงสุด	ราคาต้นทุนรวม (\$)
10	3-3-4-4-3-3-3-2-3-2-2-4-4-4-2-3-3-4-4-3-3-2-3-2-3-3-4-4-3-4-3-3-3-2-2-4-4-4-2	0.914895	2695
20	3-3-4-4-3-3-3-2-3-2-2-4-4-4-2-3-3-4-4-3-3-3-2-3-2-3-3-4-4-3-4-3-3-3-2-2-4-4-4-2	0.914895	2695
30	3-3-4-4-3-3-3-2-3-2-2-4-4-4-2-3-3-4-4-3-3-3-2-3-2-3-3-4-4-3-4-3-3-3-2-2-4-4-4-2	0.914895	2695
40	3-3-4-4-3-3-3-2-3-2-2-4-4-4-2-3-3-4-4-3-3-3-2-3-2-3-3-4-4-3-4-3-3-3-2-2-4-4-4-2	0.914895	2695
50	3-3-4-4-3-3-3-2-3-2-2-4-4-4-2-3-3-4-4-3-3-3-2-3-2-3-3-4-4-3-4-3-3-3-2-2-4-4-4-2	0.914895	2695

จากตาราง 40 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม ACO2SWAP สำหรับกรณีศึกษาที่ 5 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด โดยการปรับค่าจำนวนมด 10, 20, 30, 40 และ 50 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุดเป็นชุดเดียวกันทั้งหมด มีราคาต้นทุนรวม \$2695 ต่ำกว่างบประมาณที่ตั้งไว้ \$2700 เทคโนโลยีที่เลือก คือ 3-3-4-4-3-3-3-2-3-2-2-4-4-4-2-3-3-4-4-3-3-3-2-3-2-3-3-4-4-3-4-3-3-3-2-2-4-4-4-2

4.3.5.2 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight





ภาพประกอบ 44 ผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 5

จากภาพประกอบ 44 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 5 โดยการปรับค่าจำนวนมดและรังนก 10, 20, 30, 40 และ 50 ผลลัพธ์ของการค้นหาคำตอบที่ดีที่สุด พบว่า ไม่สามารถค้นหาคำตอบที่เหมาะสมที่สุดได้ภายใน 2,000 รอบ เมื่อเทียบกับอัลกอริทึม ACO2SWAP ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 41

ตาราง 41 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 5

จำนวนมดและ รังนกดูเหว่า	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	162.61	0.520571	0.817993	0.700549	0.103337
20	314.81	0.643255	0.828607	0.762263	0.056979
30	466.61	0.751944	0.853868	0.790978	0.029717
40	617.22	0.717897	0.868660	0.810328	0.037266
50*	769.51	0.766308	0.880625	0.844521	0.035569



จากตาราง 41 พบว่า จากการทดลองอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 5 จำนวน 10 ครั้ง โดยการตั้งค่าจำนวนมดและรังนก เท่ากับ 50 สามารถประมวลผลค้นหาคำตอบ ค่าความน่าเชื่อถือที่เหมาะสมได้เฉลี่ยสูงสุด คือ 0.844521 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.035569 ใช้เวลาเฉลี่ยในการค้นหาข้อมูล คือ 769.51 วินาที

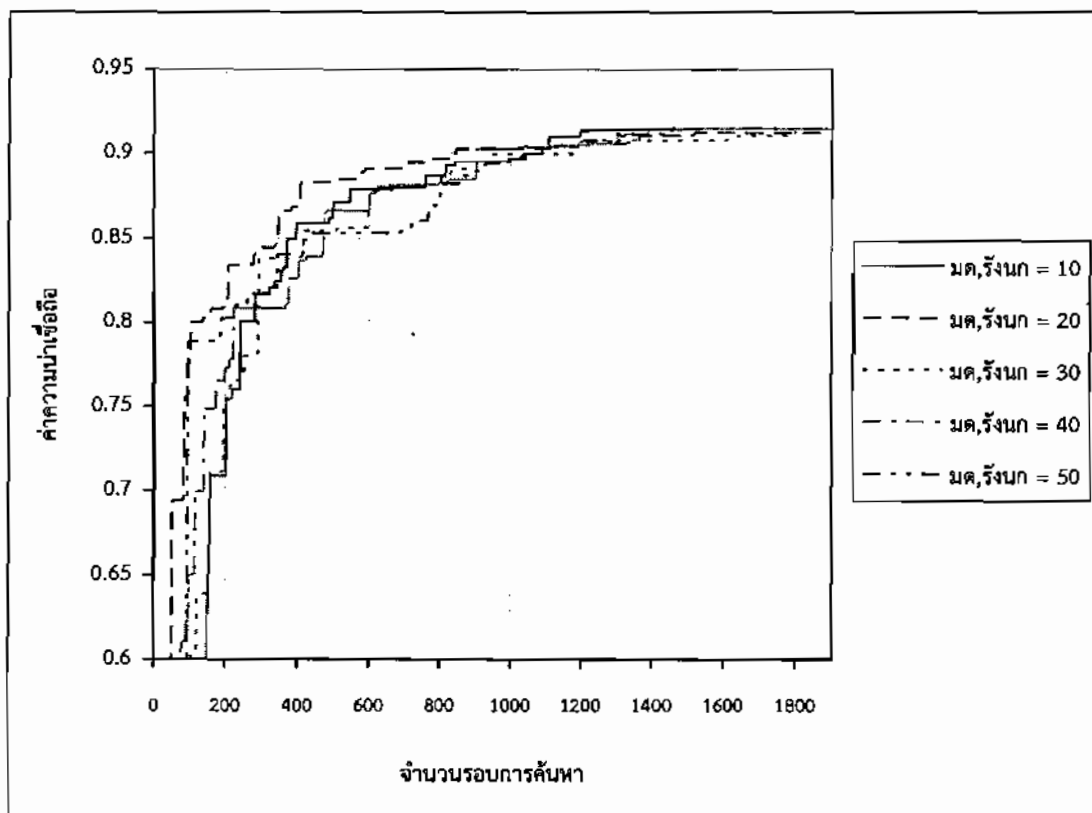
ตาราง 42 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม ACO-Lévy flight กรณีศึกษาที่ 5

จำนวนมด และรังนก ดูเหว่า	เทคโนโลยีที่เลือก	ค่าความ น่าเชื่อถือ สูงสุด	ราคา ต้นทุน รวม (\$)
10	2-2-4-5-2-3-4-2-5-1-2-3-3-4-3-4-3-5-4-2-4-4-2-3-1- 3-3-4-3-4-3-5-3-3-2-4-3-4-3-1	0.817993	2665
20	3-3-4-4-2-3-3-2-5-1-2-3-5-4-1-2-4-4-3-3-4-2-2-3-1- 2-4-5-4-3-3-3-3-2-2-4-8-3-1	0.828607	2655
30	3-3-3-5-2-4-4-2-4-1-2-5-4-3-2-2-3-4-4-2-4-4-3-4-1- 3-3-4-5-5-3-3-3-2-1-4-4-3-1	0.853868	2690
40	3-3-4-4-3-4-4-2-3-1-2-4-4-3-2-3-3-3-4-3-3-3-2-4-1- 3-4-5-3-3-3-4-3-4-2-2-4-4-4-1	0.868660	2690
50	4-3-5-5-3-3-3-2-3-1-3-4-4-3-2-3-3-3-5-3-3-3-2-3-1- 3-3-5-4-3-4-4-3-3-2-3-3-5-3-2	0.880625	2700

จากตาราง 42 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม ACO-Lévy flight สำหรับกรณีศึกษาที่ 5 ซึ่งจากการปรับค่าจำนวนมดและรังนก เท่ากับ 50 ทำให้ได้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด เท่ากับ 0.880625 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุด คือ 4-3-5-5-3-3-3-2-3-1-3-4-4-3-2-3-3-3-5-3-3-3-2-3-1-3-3-5-4-3-4-4-3-3-2-3-3-5-3-2 มีราคาต้นทุนรวม \$2700 เท่ากับงบประมาณที่ตั้งไว้

4.3.5.3 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS





ภาพประกอบ 45 ผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 5

จากภาพประกอบ 45 แผนภูมิแสดงผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษาที่ 5 พบว่า ผลลัพธ์ของการค้นหาค่าตอบที่ดีที่สุดมีแนวโน้มการเข้าสู่ค่าตอบจากการตั้งค่าจำนวนมดและรังนก เท่ากับ 40 และ 50 โดยใช้จำนวนรอบในการค้นหาโดยประมาณ 1500 รอบ และการตั้งค่าจำนวนมดและรังนก 10, 20 และ 30 ใช้จำนวนรอบในการค้นหาโดยประมาณ คือ 1800 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 43

ตาราง 43 สรุปผลการทดสอบประสิทธิภาพอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 5

จำนวนมดและ รังนกดูเหว่า	เวลาเฉลี่ย (วินาที)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
10	160.62	0.899130	0.914895	0.906920	0.004714
20*	298.16	0.902209	0.914895	0.912812	0.003883
30	470.28	0.909015	0.914872	0.911444	0.001801
40	621.80	0.909085	0.914872	0.912224	0.00198
50	734.29	0.905873	0.914895	0.912430	0.002807



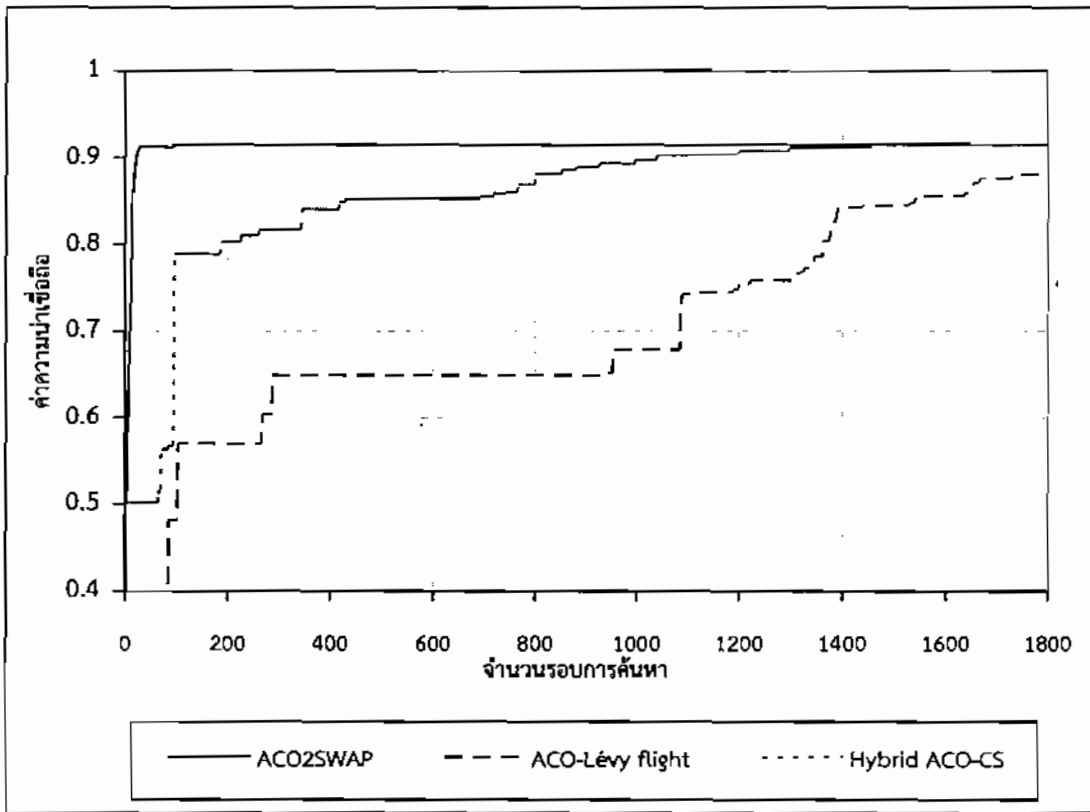
จากตาราง 43 พบว่า จากการทดลองอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษา ที่ 5 จำนวน 10 ครั้ง โดยการตั้งค่าจำนวนมดและรังนก เท่ากับ 20 สามารถประมวลผลค้นหาคำตอบ ค่าความน่าเชื่อถือที่เหมาะสมได้ค่าสูงสุด เท่ากับ 0.914895 โดยมีค่าเฉลี่ย เท่ากับ 0.912812 ส่วนเบี่ยงเบนมาตรฐาน (S.D.) เท่ากับ 0.003883 ใช้เวลาเฉลี่ยในการค้นหาข้อมูล คือ 298.16 วินาที

ตาราง 44 เปรียบเทียบผลการคัดเลือกเทคโนโลยีและราคาต้นทุนรวมของอัลกอริทึม Hybrid ACO-CS กรณีศึกษาที่ 5

จำนวนมดและรังนกดูเหว่า	เทคโนโลยีที่เลือก	ค่าความน่าเชื่อถือสูงสุด	ราคาต้นทุนรวม (\$)
10	3-3-4-4-3-3-3-2-3-2-2-4-4-4-2-3-3-4-4-3-3-3-2-3-2-3-3-4-4-3-3-3-2-3-2-3-3-4-4-3-3-3-2-2-4-4-4-2	0.914895	2695
20	3-3-4-4-3-3-3-2-3-2-2-4-4-4-2-3-3-4-4-3-3-3-2-3-2-3-3-4-4-3-3-3-2-2-4-4-4-2	0.914895	2695
30	3-3-4-4-3-3-3-2-4-2-2-4-4-4-2-3-3-4-4-3-3-3-2-3-2-3-3-4-4-3-3-3-2-2-3-4-4-2	0.914872	2700
40	3-3-4-4-3-3-3-2-4-2-2-4-4-4-2-3-3-4-4-3-3-3-2-3-2-3-3-4-4-3-3-3-2-2-3-4-4-2	0.914872	2700
50	3-3-4-4-3-3-3-2-3-2-2-4-4-4-2-3-3-4-4-3-3-3-2-3-2-3-3-4-4-3-3-3-2-2-4-4-4-2	0.914895	2695

จากตาราง 44 พบว่า การทดสอบประสิทธิภาพของอัลกอริทึม Hybrid ACO-CS สำหรับกรณีศึกษาที่ 5 ที่ให้คำตอบค่าความน่าเชื่อถือที่เหมาะสมที่สุด คือ การปรับค่าจำนวนมดและรังนก 10, 20 และ 50 มีผลการคัดเลือกเทคโนโลยีที่ทำให้ระบบอนุกรมทั้งระบบมีค่าความน่าเชื่อถือสูงสุด มีราคาต้นทุนรวม \$2695 ต่ำกว่างบประมาณที่ตั้งไว้ \$2700

4.3.5.4 เปรียบเทียบผลการทดสอบระหว่างอัลกอริทึม ACO2SWAP, ACO-Lévy flight และ Hybrid ACO-CS กรณีศึกษาที่ 3



ภาพประกอบ 46 ผลการเปรียบเทียบระหว่างอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 5

จากภาพประกอบ 46 เป็นการนำเสนอผลการเปรียบเทียบการค้นหาคำตอบที่ดีที่สุดของอัลกอริทึม ACO2SWAP, ACO-Lévy flight และ Hybrid ACO-CS สำหรับกรณีศึกษาที่ 5 ซึ่งแนวโน้มการเข้าสู่คำตอบของอัลกอริทึม ACO2SWAP ใช้จำนวนรอบในการค้นหาคำตอบน้อยที่สุดโดยประมาณคือ 100 รอบ รองลงมาคือ อัลกอริทึม Hybrid ACO-CS ใช้จำนวนรอบในการค้นหาคำตอบโดยประมาณคือ 1500 รอบ และอัลกอริทึม ACO-Lévy flight ใช้จำนวนรอบในการค้นหาคำตอบ มากกว่า 1800 รอบ ซึ่งค่าความน่าเชื่อถือและเวลาเฉลี่ยในการค้นหา แสดงในตาราง 45

ตาราง 45 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึมทั้ง 3 วิธี สำหรับกรณีศึกษาที่ 5

อัลกอริทึม	เวลาเฉลี่ย (s)	ค่าความน่าเชื่อถือ			
		ต่ำสุด	สูงสุด	เฉลี่ย	S.D.
ACO2SWAP*	280.92	0.914895	0.914895	0.914895	0
ACO-Lévy flight	769.51	0.766308	0.880625	0.844521	0.035569
Hybrid ACO-CS	298.16	0.902209	0.914895	0.912812	0.003883



จากตาราง 45 โดยภาพรวมผลลัพธ์การค้นหาคำตอบที่ดีที่สุดสำหรับกรณีศึกษาที่ 5 ของแต่ละอัลกอริทึม พบว่า อัลกอริทึม ACO2SWAP และ Hybrid ACO-CS สามารถหาค่าความน่าเชื่อถือที่เหมาะสมที่สุด ดีกว่าอัลกอริทึม ACO-Lévy flight แต่และอัลกอริทึม ACO2SWAP ใช้เวลาเฉลี่ยในการค้นหาคำตอบที่ดีกว่า Hybrid ACO-CS ซึ่งเมื่อพิจารณาจากค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐานแสดงให้เห็นว่าอัลกอริทึม ACO2SWAP มีความแม่นยำสูงสุด รองลงมา คือ อัลกอริทึม Hybrid ACO-CS และอัลกอริทึม ACO-Lévy flight ตามลำดับ สำหรับการประมวลผลระบบอนุกรมที่มี 40 ระบบย่อย 166 ตัวแปร



สรุปผลวิจัย อภิปรายผล และข้อเสนอแนะ

การวิจัยครั้งนี้เป็นการปรับปรุงขั้นตอนวิธีของอัลกอริทึมฝูงมดร่วมกับอัลกอริทึมการค้นหาแบบดุกเหว่า (Hybrid Ant Colony Optimization Cuckoo Search: Hybrid ACO-CS) เพื่อแก้ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดสำหรับระบบอนุกรม ซึ่งได้มีการทดสอบประสิทธิภาพ ดึงนำเสนอตามลำดับดังนี้

5.1 สรุปผลการวิจัย

5.1.1 การทดสอบประสิทธิภาพของอัลกอริทึมฝูงมดที่ปรับปรุงใหม่ร่วมกับอัลกอริทึมการค้นหาแบบดุกเหว่า (Hybrid ACOCS) โดยใช้ฟังก์ชันทดสอบการหาค่าที่เหมาะสม (Optimization Test Functions) จำนวน 10 ฟังก์ชัน สามารถแก้ปัญหาออปติไมเซชันเชิงตัวเลขได้ทุกฟังก์ชันที่นำเสนอ และมีประสิทธิภาพที่ดีกว่าอัลกอริทึมที่นำเสนอในงานวิจัยที่ผ่านมา ได้แก่ ฟังก์ชัน De Jong (f_1), ฟังก์ชัน Schwefel (Unimodal) (f_2), ฟังก์ชัน Ackley (f_3), ฟังก์ชัน Griewank (f_7) และฟังก์ชัน Quartic (f_9) รวมถึงมีผลลัพธ์ที่เท่ากัน คือ ฟังก์ชัน Step (f_3) และฟังก์ชัน Shekel's Foxholes (f_{10})

5.1.2 การทดสอบประสิทธิภาพของอัลกอริทึมฝูงมดที่ปรับปรุงใหม่ในการแก้ปัญหาการหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดของระบบอนุกรมที่มีข้อจำกัดหลายทางเลือกและงบประมาณ ประกอบด้วยระบบอนุกรมที่มี 15 ระบบย่อยและมีตัวแปรฮิวริสติกใช้ตัดสินใจ 60, 80 และ 100 ตัวแปร อัลกอริทึม Hybrid ACO-CS และ ACO2SWAP ให้ผลลัพธ์การค้นหาที่ดีและแม่นยำมากกว่าเมื่อเทียบกับอัลกอริทึม ACO-Lévy flight และใช้เวลาในการค้นหาคำตอบมากขึ้นสำหรับระบบอนุกรมที่มีข้อมูลเมตาฮิวริสติกขนาดใหญ่ คือ 25 ระบบย่อย 166 ตัวแปร และ 40 ระบบย่อย 266 ตัวแปร

5.2 อภิปรายผล

ผลการทดสอบประสิทธิภาพของอัลกอริทึมที่นำเสนอสามารถแก้ปัญหาออปติไมเซชันเชิงตัวเลขได้ทุกฟังก์ชันที่นำเสนอและมีประสิทธิภาพดีกว่าอัลกอริทึมที่นำเสนอในงานวิจัยที่ผ่านมา ได้แก่ ฟังก์ชัน De Jong (f_1), ฟังก์ชัน Schwefel (Unimodal) (f_2), ฟังก์ชัน Ackley (f_3), ฟังก์ชัน Griewank (f_7) และฟังก์ชัน Quartic (f_9) และมีผลลัพธ์ที่เท่ากัน คือ ฟังก์ชัน Step (f_3) และฟังก์ชัน Shekel's Foxholes (f_{10}) ทั้งนี้อาจเนื่องมาจากการร่วมกันทำงานของ 2 อัลกอริทึมในลักษณะทำงาน

ร่วมกันส่งผลให้ประสิทธิภาพสูงขึ้น แต่เมื่อพิจารณาในด้านเวลาที่ใช้ในการค้นหาคำตอบยังแสดงให้เห็นว่าใช้เวลาเฉลี่ยสูงกว่า และผลจากการประเมินผลประสิทธิภาพของอัลกอริทึมฝูงมดที่ปรับปรุงใหม่ในการแก้ปัญหาค่าความน่าเชื่อถือที่เหมาะสมที่สุดของระบบอนุกรมที่มีข้อจำกัดหลายทางเลือกและงบประมาณ ต้องอาศัยการปรับแต่งค่าพารามิเตอร์ เพื่อให้ได้ค่าผลลัพธ์ที่ดีขึ้น การปรับหรือตั้งค่าพารามิเตอร์ต่าง ๆ ของอัลกอริทึมจึงเป็นปัจจัยหนึ่งที่ทำให้การทำงานของอัลกอริทึมหาค่าที่เหมาะสมที่สุดได้ รวมถึงบางพารามิเตอร์ยังช่วยให้การค้นหานั้นพบคำตอบที่เร็วขึ้นด้วย

5.3 ข้อเสนอแนะในการพัฒนาต่อไป

อัลกอริทึมที่นำเสนอสามารถนำไปประยุกต์ใช้ในงานอื่น ๆ ได้อย่างหลากหลาย ทั้งในระบบสนับสนุนการตัดสินใจ ระบบปัญญาประดิษฐ์ งานประยุกต์ด้านวิศวกรรม ทั้งนี้ต้องพิจารณาเตรียมข้อมูลป้อนเข้า (Input) และข้อมูลอิวิริสติกให้สอดคล้องกับการทำงานของอัลกอริทึมที่นำเสนอ



เอกสารอ้างอิง



เอกสารอ้างอิง

- [1] U. Diwekar. "Introduction to Applied Optimization". 2nd ed, New York : Springer; 2008.
- [2] R. Meziane, Y. Massim, A. Zebalah, A. Ghoraf, R. Rahli. "Reliability optimization using ant colony algorithm under performance and cost constraints". *Electric Power Systems Research* 2005; 76: 1–8.
- [3] Ning Ruan, Xiaoling Sun. "An exact algorithm for cost minimization in series reliability systems with multiple component choices". *Applied Mathematics and Computation* 2006; 181: 732–741.
- [4] Won Young Yuna, Young Man Songa, Ho-Gyun Kim. "Multiple multi-level redundancy allocation in series systems". *Reliability Engineering and System Safety* 2007; 92: 308–313.
- [5] R. Tavakkoli-Moghaddam, J. Safari, F. Sassanic. "Reliability optimization of series-parallel systems with a choice of redundancy strategies using a genetic algorithm". *Reliability Engineering and System Safety* 2008; 93: 550–556.
- [6] R.K. Gupta, A.K. Bhunia, D. Roy. "A GA based penalty function technique for solving constrained redundancy allocation problem of series system with interval valued reliability of components". *Journal of Computational and Applied Mathematics* 2009; 232: 275-284.
- [7] Manju Agarwal, Vikas K. Sharma. "Ant colony approach to constrained redundancy optimization in binary systems". *Applied Mathematical Modelling* 2010; 34: 992–1003.
- [8] Noura Beji, Bassem Jarboui, Mansour Eddaly, Habib Chabchoub. "A Hybrid Particle Swarm Optimization Algorithm for the Redundancy Allocation Problem". *Journal of Computational Science* 2010; 1: 159–167.
- [9] Vikas K. Sharma, Manju Agarwal, Kanwar Sen. "Reliability evaluation and optimal design in heterogeneous multi-state series-parallel systems". *Information Sciences* 2011; 181: 362–378.
- [10] Wei-Chang Yeh, Tsung-JungHsieh. "Solving reliability redundancy allocation problems using an artificial bee colony algorithm". *Computers & Operations Research* 2011; 38: 1465–1473.
- [11] Hatice Tekiner-Mogulkoc, David W. Coit. "System Reliability Optimization Considering Uncertainty: Minimization of the Coefficient of Variation for Series-Parallel Systems". *IEEE Transactions on Reliability* 2011; 60: 667-674.



- [12] Amirhossain Chambari, Seyed Habib A. Rahmati, Amir Abbas Najafi, Aida karimi. "bi-objective model to optimize reliability and cost of system with a choice of redundancy strategies". *Computers & Industrial Engineering* 2012; 63: 109–119.
- [13] Wei-Chang Yeh. "A new algorithm for generating minimal cut sets in k-out-of-n networks". *Reliability Engineering and System Safety* 2006; 91: 36–43.
- [14] Eduardo Sáenz-de-Cabezón, Henry P. Wynn. "Computational algebraic algorithms for the reliability of generalized k-out-of-n and related systems". *Mathematics and Computers in Simulation* 2011; 82: 68 –78.
- [15] Yong Wang, Lin Li, Shuhong Huang, Qing Chang. "Reliability and covariance estimation of weighted k-out-of-n multi-state systems". *European Journal of Operational Research* 2012; 221: 138 –147.
- [16] N. Nahas, M. Nourelfath. "Ant system for reliability optimization of a series system with multiple-choice and budget constraints". *Reliability Engineering and System Safety* 2005; 87: 1–12.
- [17] F. Ahmadizar, H. Soltanpanah. "Reliability optimization of a series system with multiple-choice and budget constraints using an efficient ant colony approach". *Expert Systems with Applications* 2011; 38: 3640–3646.
- [18] C.S. Sung, Y.K. Cho. "Reliability optimization of a series system with multiple-choice and budget constraints". *European Journal of Operational Research* 2000; 127: 159-171.
- [19] M. Nourelfath, N. Nahas. "Quantized hopfield networks for reliability optimization". *Reliability Engineering and System Safety* 2003; 81: 191-196.
- [20] Wuthichai Wongthatsanekorn, Nuntana Matheekrieangkrai. "Bee Algorithm for Solving Yield Optimization Problem for Hard Disk Drive Component under Budget and Supplier's Rating Constraints and Heuristic Performance Comparison". In: *Intelligent Automation and Systems Engineering of the series Lecture Notes in Electrical Engineering*. New York: Springer; 2011. pp. 203-216.
- [21] M. Dorigo, L.M. Gambardella. ".Ant colonies for the traveling salesman problem". *BioSystems* 1997; 43: 73–81.
- [22] M. Dorigo, L.M. Gambardella. "Ant Colony System: A cooperative learning approach to the traveling salesman problem". *IEEE Transactions on Evolutionary Computation* 1997; 1: 53–66.
- [23] X.-S. Yang, S. Deb. "Engineering Optimisation by Cuckoo Search". *Int. J. Mathematical Modelling and Numerical Optimisation* 2010; 1: 330–343.
- [24] ระพีพันธ์ ปิตาคะโส. "วิธีการเมตาฮิวริสเพื่อแก้ไขปัญหาการวางแผนการผลิตและการจัดการโลจิสติกส์". กรุงเทพฯ: สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น); 2554.
- [25] พงศ์ชนัน เหลืองไพฑูริย์. "การวิจัยดำเนินงาน". กรุงเทพฯ: ท้อป; 2553.



- [26] C. Blum, A. Roli. "Metaheuristics in combinatorial optimization: overview and conceptual comparison". *ACM Computing Surveys* 2003; 35: 268–308.
- [27] M. Duran Toksari. "Minimizing the multimodal functions with Ant Colony Optimization approach". *Expert Systems with Applications* 2009; 36: 6030-6035.
- [28] S. Pothiya , I. Ngamroo & W. Kongprawechnon, "Ant colony optimisation for economic dispatch problem with non-smooth cost functions". *Electrical Power and Energy Systems* 2010; 32: 478–487.
- [29] M. Dorigo. "Optimization, learning and natural algorithms" [Ph.D. thesis]. Italy: Politecnico di Milano University; 1992.
- [30] C. Blum, M. Dorigo, "The hyper-cube framework for ant colony optimization". *IEEE Transactions on Systems, Man, and Cybernetics—Part B* 2004; 34: 1161–1172.
- [31] O. Baskan, S. Haldenbilen, H. Ceylan, H. Ceylan. "A new solution algorithm for improving performance of ant colony optimization". *Applied Mathematics and Computation* 2009; 211: 75-84
- [32] M. Dorigo, G. Di Caro. "The Ant Colony Optimization meta-heuristic in *New Ideas in Optimization*". London: McGraw Hill; 1999.
- [33] M. Dorigo, G. Di Caro. 1999b; "Ant Colony Optimization: A New Meta-Heuristic". *Proceedings of the 1999 Congress on Evolutionary Computation (CEC 99)*; 6-9 July 1999; Washington DC, USA. 1999. pp. 1477.
- [34] M. Dorigo, G. Di Caro, L.M. Gambardella. "Ant algorithms for discrete optimization", *Artificial Life* 1999; 5: 137–172.
- [35] M.L. Ibáñez, L. Paquete, T. Stützle. "On the design of ACO for the biobjective quadratic assignment problem". *Proceedings of 4th International Workshop (ANTS 2004)*; 5-8 September 2004; Brussels, Belgium. 2004. pp. 214–225.
- [36] R. Montemanni, L.M. Gambardella, A.E. Rizzoli, A.V. Donati. "Ant colony system for a dynamic vehicle routing problem". *Journal of Combinatorial Optimization* 2005; 10: 327–343.
- [37] C. Blum. "Beam-ACO—Hybridizing ant colony optimization with beam search: An application to open shop scheduling". *Computers & Operations Research* 2005; 32: 1565–1591.
- [38] M. Dorigo, V. Maniezzo, A. Coloni. "Ant System: Optimization by a colony of cooperating agents". *IEEE Transactions on Systems, Man and Cybernetics—Part B* 1996; 26: 29–41.
- [39] M. Dorigo, V. Maniezzo, A. Coloni. "Distributed optimization by ant colonies". *Proceedings of ECAL91 - European Conference on Artificial Life*; Paris, France. 1991. pp. 134–142.



- [40] L.M. Gambardella, M. Dorigo. "Ant-Q: A reinforcement learning approach to the traveling salesman problem". Proceedings of 12th International Conference on Machine Learning (ML- 95), Lake Tahoe, California, USA. 1995. pp. 252–260.
- [41] L.M. Gambardella, M. Dorigo. 1996; "Solving symmetric and asymmetric TSPs by ant colonies". Proceedings of IEEE International Conference on Evolutionary Computation (ICEC'96); Piscataway, NJ. 1996. pp. 622–627.
- [42] T. Stützle, H.H. Hoos. "MAX-MIN Ant System". *Future Generation Computer Systems* 2000; 16: 889–914.
- [43] B. Bullnheimer, R.F. Hartl, C. Strauss. "A new rank-based version of the Ant System: A computational study". *Central European Journal for Operations Research and Economics* 1999; 7: 25–38.
- [44] O. Cordon, I.F. de Viana, F. Herrera, L. Moreno. "A new ACO model integrating evolutionary computation concepts: The best-worst Ant System". Proceedings of ANTS 2000, Université libre de Bruxelles, Belgium, 2000. pp. 22–29.
- [45] V. Maniezzo. "Exact and approximate nondeterministic tree-search procedures for the quadratic assignment problem". *INFORMS Journal on Computing* 1999; 11: 358–369.
- [46] C. Blum, A. Roli, M. Dorigo. "HC-ACO: The hyper-cube framework for Ant Colony Optimization". Proceedings of MIC'2001 - Metaheuristics International Conference, Porto, Portugal, 2001. pp. 399–403.
- [47] K. Watcharasitthiwat, P. Wardkein. "Reliability optimization of topology communication network design using an improved ant colony optimization". *Computers & Electrical Engineering* 2009; 35: 730–747.
- [48] Lanz von Horsten, 2015, "A Cape Batis male feeds a Klaass Cuckoo chick while the female looks on". [Online]. 2015 [cited 20 May 2015]; Available from: <http://www.gettyimages.com/license/624714145>
- [49] Wikipedia. "Brown-headed cowbird". [Online]. 2015 [cited 20 May 2015]; Available from: https://en.wikipedia.org/wiki/Brown-headed_cowbird
- [50] Vittorio Baglione. "Cuckoos and crows teach us how parasites can be good". [Online]. 2014 [cited 20 May 2015]; Available from: <http://www.treehugger.com/natural-sciences/cuckoos-and-crows-teach-us-how-parasites-can-be-good.html>
- [51] Rutuparna Panda, Sanjay Agrawal, Sudipta Bhuyan, "Edge magnitude based multilevel thresholding using Cuckoo search technique". *Expert Systems with Applications* 2013; 40: 7617-7628.



- [52] Ehsan Valian, Shahram Mohanna, Saeed Tavakoli, "Improved Cuckoo Search Algorithm for Global Optimization". *International Journal of Communications and Information Technology* 2011; 1: 31-44.
- [53] Xin-She Yang, Suash Deb, "Multiobjective cuckoo search for design optimization". *Computers & Operations Research* 2013; 40: 1616-1624.
- [54] R.G. Babukarthik, R. Raju, P. Dhavachelvan. "Energy-aware scheduling using Hybrid Algorithm for cloud computing". *Proceedings of Computing Communication & Networking Technologies (ICCCNT)*, 26-28 July 2012, Coimbatore, 2012. pp. 1 – 6.
- [55] M. Naik, M. R. Nath, A. Wunnava, S. Sahany, R. Panda. 2015; "A new adaptive Cuckoo search algorithm". *Proceedings of 2015 IEEE 2nd International Conference on Recent Trends in Information Systems (ReTIS)*, Kolkata, 2015. pp. 1-5.



ภาคผนวก



ข้อมูลตัวอย่างที่ใช้ทดสอบประสิทธิภาพของอัลกอริทึม



ข้อมูลตัวอย่างที่ใช้ทดสอบประสิทธิภาพของอัลกอริทึม

การทดสอบประสิทธิภาพของอัลกอริทึมที่นำเสนอในครั้งนี ใช้ข้อมูลตัวอย่างจากงานวิจัยที่ผ่านมา [16-17] รวม 5 ชุดตัวอย่าง ดังแสดงในภาคผนวก ตารางที่ 37-41

1. ระบบอนุกรมที่มี 15 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 60 ตัวแปร กำหนดราคารวมต้นทุน เท่ากับ \$1000

2. ระบบอนุกรมที่มี 15 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 80 ตัวแปร กำหนดราคารวมต้นทุน \$900

3. ระบบอนุกรมที่มี 15 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 100 ตัวแปร กำหนดราคารวมต้นทุน เท่ากับ \$1000

4. ระบบอนุกรมที่มี 25 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 166 ตัวแปร กำหนดราคารวมต้นทุน เท่ากับ \$1400

5. ระบบอนุกรมที่มี 40 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 266 ตัวแปร กำหนดราคารวมต้นทุน เท่ากับ \$2700

ตาราง 46 ระบบอนุกรมที่มี 15 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 60 ตัวแปร

Subsystems	Tech 1	Tech 2	Tech 3	Tech 4	Tech 5	Tech 6	Tech 7	Tech 8
1 Reliability	0.9	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0.99999999
Cost (\$)	20	40	60	80	100	120	140	180
2 Reliability	0.85	0.9775	0.9966	0.9995	0.9999	0	0	0
Cost (\$)	30	60	90	120	150	0	0	0
3 Reliability	0.8	0.96	0.99	0.998	0.9997	0	0	0
Cost (\$)	20	40	60	80	100	0	0	0
4 Reliability	0.75	0.938	0	0	0	0	0	0
Cost (\$)	30	40	0	0	0	0	0	0
5 Reliability	0.85	0.99	0.999	0	0	0	0	0
Cost (\$)	20	40	65	0	0	0	0	0
6 Reliability	0.9	0.95	0.999	0.9999	0	0	0	0
Cost (\$)	25	30	50	70	0	0	0	0
7 Reliability	0.95	0.99	0	0	0	0	0	0
Cost (\$)	40	60	0	0	0	0	0	0
8 Reliability	0.85	0.995	0.999	0.9999	0.99999	0	0	0
Cost (\$)	10	30	60	80	120	0	0	0



ตาราง 46 (ต่อ)

	Subsystems	Tech 1	Tech 2	Tech 3	Tech 4	Tech 5	Tech 6	Tech 7	Tech 8
9	Reliability	0.9	0.95	0	0	0	0	0	0
	Cost (\$)	30	50	0	0	0	0	0	0
10	Reliability	0.99	0.999	0.9999	0.99999	0.999999	0	0	0
	Cost (\$)	15	40	70	100	130	0	0	0
11	Reliability	0.95	0.999	0.9998	0.99999	0.999998	0.9999999	0	0
	Cost (\$)	20	40	60	80	100	120	0	0
12	Reliability	0.8	0.9	0.99	0	0	0	0	0
	Cost (\$)	40	60	85	0	0	0	0	0
13	Reliability	0.75	0.85	0.99	0.999	0	0	0	0
	Cost (\$)	30	50	80	100	0	0	0	0
14	Reliability	0.8	0.95	0.99	0	0	0	0	0
	Cost (\$)	10	30	40	0	0	0	0	0
15	Reliability	0.99	0.999	0.9999	0.99999	0	0	0	0
	Cost (\$)	50	80	110	140	0	0	0	0



ตาราง 47 ระบบอนุกรมที่มี 15 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 80 ตัวแปร

Subsystems	Tech 1	Tech 2	Tech 3	Tech 4	Tech 5	Tech 6	Tech 7	Tech 8
1 Reliability	0.9	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0.99999999
Cost (\$)	20	40	60	80	100	120	140	180
2 Reliability	0.85	0.9775	0.9966	0.9995	0.9999			
Cost (\$)	30	60	90	120	150			
3 Reliability	0.8	0.96	0.99	0.998	0.9997			
Cost (\$)	20	40	60	80	100			
4 Reliability	0.75	0.938	0.97	0.99	0.995			
Cost (\$)	30	40	60	70	80			
5 Reliability	0.85	0.99	0.999	0.9999				
Cost (\$)	20	40	65	80				
6 Reliability	0.9	0.95	0.999	0.9999				
Cost (\$)	25	30	50	70				
7 Reliability	0.95	0.99	0.999	0.9999				
Cost (\$)	40	60	80	100				
8 Reliability	0.85	0.995	0.999	0.9999	0.99999			
Cost (\$)	10	30	60	80	120			
9 Reliability	0.9	0.95	0.98	0.995	0.9999			
Cost (\$)	30	50	70	90	120			
10 Reliability	0.99	0.999	0.9999	0.99999	0.999999			
Cost (\$)	15	40	70	100	130			
11 Reliability	0.95	0.999	0.9998	0.99999	0.999998	0.9999999		
Cost (\$)	20	40	60	80	100	120		
12 Reliability	0.8	0.9	0.99	0.999	0.9999	0.99999	0.999997	0.9999995
Cost (\$)	40	60	85	100	120	140	155	170
13 Reliability	0.75	0.85	0.99	0.999				
Cost (\$)	30	50	80	100				
14 Reliability	0.8	0.95	0.99	0.996	0.9993	0.9999	0.99996	0.999998
Cost (\$)	10	30	40	60	80	95	120	140
15 Reliability	0.99	0.999	0.9999	0.99999				
Cost (\$)	50	80	110	140	50	80	110	140



ตาราง 48 ระบบอนุกรมที่มี 15 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 100 ตัวแปร

Subsystems	Tech 1	Tech 2	Tech 3	Tech 4	Tech 5	Tech 6	Tech 7	Tech 8
1 Reliability	0.9	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0.99999999
Cost (\$)	20	40	60	80	100	120	140	180
2 Reliability	0.85	0.9775	0.9966	0.9995	0.9999			
Cost (\$)	30	60	90	120	150			
3 Reliability	0.8	0.96	0.99	0.998	0.9997	0.9999	0.99999	0.999999
Cost (\$)	20	40	60	80	100	120	140	160
4 Reliability	0.75	0.938	0.98	0.999	0.9999			
Cost (\$)	30	40	50	60	70			
5 Reliability	0.85	0.99	0.999	0.9999	0.99998	0.999998	0.9999998	0.99999998
Cost (\$)	20	40	65	80	100	120	140	155
6 Reliability	0.9	0.95	0.999	0.9999	0.99999			
Cost (\$)	25	30	50	70	90			
7 Reliability	0.95	0.99	0.997	0.9997	0.99997	0.999997	0.9999997	0.99999997
Cost (\$)	40	60	80	100	120	140	160	180
8 Reliability	0.85	0.995	0.999	0.9999	0.99999			
Cost (\$)	10	30	60	80	120			
9 Reliability	0.9	0.95	0.995	0.9995	0.99995	0.999995	0.9999995	0.99999995
Cost (\$)	30	50	70	90	110	130	150	170
10 Reliability	0.99	0.999	0.9999	0.99999	0.999999	0.9999999		
Cost (\$)	15	40	70	100	130	160		
11 Reliability	0.95	0.999	0.9998	0.99999	0.999998	0.9999999	0.99999997	0.999999999
Cost (\$)	20	40	60	80	100	120	140	160
12 Reliability	0.8	0.9	0.99	0.999	0.9999			
Cost (\$)	40	60	85	110	130			
13 Reliability	0.75	0.85	0.99	0.999	0.9996	0.99996	0.999996	0.9999996
Cost (\$)	30	50	80	100	120	140	160	180
14 Reliability	0.8	0.95	0.99	0.999	0.9999			
Cost (\$)	10	30	40	60	80			
15 Reliability	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0.99999998	0.999999999
Cost (\$)	50	80	110	140	160	180	200	220



ตาราง 49 ระบบอนุกรมที่มี 25 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 166 ตัวแปร

Subsystems	Tech 1	Tech 2	Tech 3	Tech 4	Tech 5	Tech 6	Tech 7	Tech 8
1 Reliability	0.9	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0.99999999
Cost (\$)	20	40	60	80	100	120	140	180
2 Reliability	0.85	0.9775	0.9966	0.9995	0.9999	0	0	0
Cost (\$)	30	60	90	120	150	0	0	0
3 Reliability	0.8	0.96	0.99	0.998	0.9997	0.9999	0.99999	0.999999
Cost (\$)	20	40	60	80	100	120	140	160
4 Reliability	0.75	0.938	0.98	0.999	0.9999	0	0	0
Cost (\$)	30	40	50	60	70	0	0	0
5 Reliability	0.85	0.99	0.999	0.9999	0.99998	0.999998	0.9999998	0.99999998
Cost (\$)	20	40	65	80	100	120	140	155
6 Reliability	0.9	0.95	0.999	0.9999	0.99999	0	0	0
Cost (\$)	25	30	50	70	90	0	0	0
7 Reliability	0.95	0.99	0.997	0.9997	0.99997	0.999997	0.9999997	0.99999997
Cost (\$)	40	60	80	100	120	140	160	180
8 Reliability	0.85	0.995	0.999	0.9999	0.99999	0	0	0
Cost (\$)	10	30	60	80	120	0	0	0
9 Reliability	0.9	0.95	0.995	0.9995	0.99995	0.999995	0.9999995	0.99999995
Cost (\$)	30	50	70	90	110	130	150	170
10 Reliability	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0	0
Cost (\$)	15	40	70	100	130	160	0	0
11 Reliability	0.95	0.999	0.9998	0.99999	0.999998	0.9999999	0.99999997	0.999999999
Cost (\$)	20	40	60	80	100	120	140	160
12 Reliability	0.8	0.9	0.99	0.999	0.9999	0	0	0
Cost (\$)	40	60	85	110	130	0	0	0
13 Reliability	0.75	0.85	0.99	0.999	0.9996	0.99996	0.999996	0.9999996
Cost (\$)	30	50	80	100	120	140	160	180
14 Reliability	0.8	0.95	0.99	0.999	0.9999	0	0	0
Cost (\$)	10	30	40	60	80	0	0	0



ตาราง 49 (ต่อ)

Subsystems	Tech 1	Tech 2	Tech 3	Tech 4	Tech 5	Tech 6	Tech 7	Tech 8
15 Reliability	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0.99999998	0.999999995
Cost (\$)	50	80	110	140	160	180	200	220
16 Reliability	0.9	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0.99999999
Cost (\$)	20	40	60	80	100	120	140	180
17 Reliability	0.85	0.9775	0.9966	0.9995	0.9999	0	0	0
Cost (\$)	30	60	90	120	150	0	0	0
18 Reliability	0.8	0.96	0.99	0.998	0.9997	0.9999	0.99999	0.999999
Cost (\$)	20	40	60	80	100	120	140	160
19 Reliability	0.75	0.938	0.98	0.999	0.9999	0	0	0
Cost (\$)	30	40	50	60	70	0	0	0
20 Reliability	0.85	0.99	0.999	0.9999	0.99998	0.999998	0.9999998	0.99999998
Cost (\$)	20	40	65	80	100	120	140	155
21 Reliability	0.9	0.95	0.999	0.9999	0.99999	0	0	0
Cost (\$)	25	30	50	70	90	0	0	0
22 Reliability	0.95	0.99	0.997	0.9997	0.99997	0.999997	0.9999997	0.99999997
Cost (\$)	40	60	80	100	120	140	160	180
23 Reliability	0.85	0.995	0.999	0.9999	0.99999	0	0	0
Cost (\$)	10	30	60	80	120	0	0	0
24 Reliability	0.9	0.95	0.995	0.9995	0.99995	0.999995	0.9999995	0.99999995
Cost (\$)	30	50	70	90	110	130	150	170
25 Reliability	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0	0
Cost (\$)	15	40	70	100	130	160	0	0



ตาราง 50 ระบบอนุกรมที่มี 40 ระบบย่อยและมีตัวแปรที่ใช้ในการตัดสินใจ 266 ตัวแปร

Subsystems	Tech 1	Tech 2	Tech 3	Tech 4	Tech 5	Tech 6	Tech 7	Tech 8
1 Reliability	0.9	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0.99999999
Cost (\$)	20	40	60	80	100	120	140	180
2 Reliability	0.85	0.9775	0.9966	0.9995	0.9999	0	0	0
Cost (\$)	30	60	90	120	150	0	0	0
3 Reliability	0.8	0.96	0.99	0.998	0.9997	0.9999	0.99999	0.999999
Cost (\$)	20	40	60	80	100	120	140	160
4 Reliability	0.75	0.938	0.98	0.999	0.9999	0	0	0
Cost (\$)	30	40	50	60	70	0	0	0
5 Reliability	0.85	0.99	0.999	0.9999	0.99998	0.999998	0.9999998	0.99999998
Cost (\$)	20	40	65	80	100	120	140	155
6 Reliability	0.9	0.95	0.999	0.9999	0.99999	0	0	0
Cost (\$)	25	30	50	70	90	0	0	0
7 Reliability	0.95	0.99	0.997	0.9997	0.99997	0.999997	0.9999997	0.99999997
Cost (\$)	40	60	80	100	120	140	160	180
8 Reliability	0.85	0.995	0.999	0.9999	0.99999	0	0	0
Cost (\$)	10	30	60	80	120	0	0	0
9 Reliability	0.9	0.95	0.995	0.9995	0.99995	0.999995	0.9999995	0.99999995
Cost (\$)	30	50	70	90	110	130	150	170
10 Reliability	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0	0
Cost (\$)	15	40	70	100	130	160	0	0
11 Reliability	0.95	0.999	0.9998	0.99999	0.999998	0.9999999	0.99999997	0.999999999
Cost (\$)	20	40	60	80	100	120	140	160
12 Reliability	0.8	0.9	0.99	0.999	0.9999	0	0	0
Cost (\$)	40	60	85	110	130	0	0	0
13 Reliability	0.75	0.85	0.99	0.999	0.9996	0.99996	0.999996	0.9999996
Cost (\$)	30	50	80	100	120	140	160	180
14 Reliability	0.8	0.95	0.99	0.999	0.9999	0	0	0
Cost (\$)	10	30	40	60	80	0	0	0



ตาราง 50 (ต่อ)

Subsystems	Tech 1	Tech 2	Tech 3	Tech 4	Tech 5	Tech 6	Tech 7	Tech 8
15 Reliability	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0.99999998	0.999999995
Cost (\$)	50	80	110	140	160	180	200	220
16 Reliability	0.9	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0.99999999
Cost (\$)	20	40	60	80	100	120	140	180
17 Reliability	0.85	0.9775	0.9966	0.9995	0.9999	0	0	0
Cost (\$)	30	60	90	120	150	0	0	0
18 Reliability	0.8	0.96	0.99	0.998	0.9997	0.9999	0.99999	0.999999
Cost (\$)	20	40	60	80	100	120	140	160
19 Reliability	0.75	0.938	0.98	0.999	0.9999	0	0	0
Cost (\$)	30	40	50	60	70	0	0	0
20 Reliability	0.85	0.99	0.999	0.9999	0.99998	0.999998	0.9999998	0.99999998
Cost (\$)	20	40	65	80	100	120	140	155
21 Reliability	0.9	0.95	0.999	0.9999	0.99999	0	0	0
Cost (\$)	25	30	50	70	90	0	0	0
22 Reliability	0.95	0.99	0.997	0.9997	0.99997	0.999997	0.9999997	0.99999997
Cost (\$)	40	60	80	100	120	140	160	180
23 Reliability	0.85	0.995	0.999	0.9999	0.99999	0	0	0
Cost (\$)	10	30	60	80	120	0	0	0
24 Reliability	0.9	0.95	0.995	0.9995	0.99995	0.999995	0.9999995	0.99999995
Cost (\$)	30	50	70	90	110	130	150	170
25 Reliability	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0	0
Cost (\$)	15	40	70	100	130	160	0	0
26 Reliability	0.95	0.99	0.999	0.9995	0.99999	0.999995	0.9999999	0.99999998
Cost (\$)	25	35	55	70	95	115	140	160
27 Reliability	0.85	0.97	0.997	0.9995	0.9999	0	0	0
Cost (\$)	40	60	90	120	145	0	0	0



ตาราง 50 (ต่อ)

Subsystems	Tech 1	Tech 2	Tech 3	Tech 4	Tech 5	Tech 6	Tech 7	Tech 8
28 Reliability	0.85	0.96	0.99	0.998	0.9998	0.99995	0.99999	0.999999
Cost (\$)	25	45	60	85	100	125	150	170
29 Reliability	0.8	0.9	0.98	0.998	0.9995	0	0	0
Cost (\$)	30	45	60	70	85	0	0	0
30 Reliability	0.8	0.98	0.995	0.9995	0.99995	0.999995	0.9999995	0.99999995
Cost (\$)	20	40	60	80	100	120	140	160
31 Reliability	0.85	0.9	0.99	0.999	0.9999	0	0	0
Cost (\$)	20	30	50	70	90	0	0	0
32 Reliability	0.9	0.97	0.997	0.9997	0.99997	0.999997	0.9999997	0.99999997
Cost (\$)	30	50	70	90	110	130	150	170
33 Reliability	0.85	0.95	0.995	0.9995	0.99995	0	0	0
Cost (\$)	15	30	60	85	110	0	0	0
34 Reliability	0.9	0.95	0.995	0.9995	0.99995	0.999995	0.9999995	0.99999995
Cost (\$)	25	45	65	85	105	125	145	165
35 Reliability	0.95	0.999	0.9995	0.99999	0.999995	0.9999999	0	0
Cost (\$)	20	45	70	100	140	170	0	0
36 Reliability	0.99	0.998	0.9998	0.99998	0.999998	0.9999998	0.99999998	0.999999998
Cost (\$)	30	40	60	80	100	120	140	160
37 Reliability	0.8	0.9	0.99	0.999	0.9999	0	0	0
Cost (\$)	30	50	80	115	130	0	0	0
38 Reliability	0.75	0.85	0.95	0.996	0.9996	0.99996	0.999996	0.9999996
Cost (\$)	20	40	75	100	115	140	155	175
39 Reliability	0.75	0.9	0.99	0.999	0.9995	0	0	0
Cost (\$)	15	30	40	60	80	0	0	0
40 Reliability	0.99	0.999	0.9999	0.99999	0.999999	0.9999999	0.99999998	0.999999995
Cost (\$)	40	70	100	130	160	185	210	225



ประวัติย่อผู้วิจัย



ประวัติย่อผู้วิจัย

ชื่อ นามสกุล	นายมานะ โสภ
วัน เดือน ปีเกิด	วันที่ 16 เมษายน พ.ศ. 2521
จังหวัด และประเทศที่เกิด	อำเภอเมือง จังหวัดสุรินทร์ ประเทศไทย
ประวัติการศึกษา	พ.ศ. 2536 มัธยมศึกษาที่ 3 โรงเรียนสุรวิทยาคาร พ.ศ. 2539 ประกาศนียบัตรวิชาชีพ (ปวช.) ช่างไฟฟ้าอิเล็กทรอนิกส์ วิทยาลัยเทคนิคสุรินทร์ พ.ศ. 2541 ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.) ช่างอิเล็กทรอนิกส์ วิทยาลัยเทคนิคสุรินทร์ พ.ศ. 2543 ประกาศนียบัตรครุเทคนิคชั้นสูง (ปทส.) ครุเทคนิคไฟฟ้าสื่อสารวิทยาลัยเทคนิคอุดรธานี พ.ศ. 2546 ปริญญาครุศาสตร์อุตสาหกรรมบัณฑิต (ค.อ.บ.) สาขาวิศวกรรมอิเล็กทรอนิกส์และโทรคมนาคม สถาบันเทคโนโลยีปทุมวัน พ.ศ. 2549 ปริญญาครุศาสตร์อุตสาหกรรมมหาบัณฑิต (ค.อ.ม.) สาขาวิชาเทคโนโลยีคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ พ.ศ. 2559 ปริญญาปรัชญาดุษฎีบัณฑิต (ปร.ด.) สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ มหาวิทยาลัยมหาสารคาม
ตำแหน่ง สถานที่ทำงาน	อาจารย์ สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ คณะวิทยาศาสตร์ มหาวิทยาลัยราชภัฏอุดรธานี อำเภอเมือง จังหวัดอุดรธานี 41000
ที่อยู่ที่สามารถติดต่อได้	บ้านเลขที่ 99/57 หมู่ 1 ถนนนิตโย ตำบลหนองนาคำ อำเภอเมือง จังหวัดอุดรธานี 41000

ผลงานวิจัย

Mana Sopa, Niwat Angkawisittpa. "An Application of Cuckoo Search Algorithm for Series System with Cost and Multiple Choices ConstraintsOriginal Research Article".
Procedia Computer Science 2016, 86: 453-456.

