การแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจบ

นายเกียรติโสภณ ไวยพารา

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2555

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย
บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ที่ส่งผ่านทางบัณฑิตวิทยาลัย
The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository(CUIR)
are the thesis authors' files submitted through the Graduate School.

SOLVING SUDOKU PROBLEMS BY COINCIDENCE ALGORITHM

Mr. Kiatsopon Waiyapara

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Engineering Program in Computer Engineering

Department of Computer Engineering

Facalty of Engineering

Chulalongkorn University

Academic Year 2012

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบ
โดย	นายเกียรติโสภณ ไวยพารา
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.สุกรี สินธุภิญโญ)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา)

..... กรรมการภายนอกมหาวิทยาลัย
(รองศาสตราจารย์ ดร.พีรยุทธ์ ชาญเศรษฐิกุล)

เกียรติโสภณ ไวยพารา : การแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบ (SOLVING SUDOKU PROBLEMS BY COINCIDENCE ALGORITHM). อ.ที่ปรึกษาวิทยานิพนธ์หลัก : ศ. ดร.ประภาส จงสถิตย์วัฒนา, 72 หน้า.

วิทยานิพนธ์ฉบับนี้เป็นการนำเสนอวิธีการแก้ปัญหาเชิงการจัดอย่างซูโดะกุโดยขั้นตอนวิธีการบรรจวบ ซึ่งเป็นขั้นตอนวิธีใหม่ในขั้นตอนวิธีประมาณการแจกแจงในกลุ่มขั้นตอนวิธีเชิงวิวัฒน์ ขั้นตอนวิธีดังกล่าวอาศัยการเรียนรู้เชิงลบควบคู่กับการเรียนรู้เชิงบวกในการหาผลเฉลยที่เป็นไปได้ ซึ่งขั้นตอนวิธีดังกล่าวถูกนำมาสำรวจประสิทธิภาพในหลากหลายปัญหาที่อยู่ในกลุ่มปัญหาเชิงการจัด แล้วว่าสามารถแก้ปัญหากลุ่มดังกล่าวได้อย่างมีประสิทธิภาพเมื่อเทียบกับหลายขั้นตอนวิธีในกลุ่มขั้นตอนวิธีเชิงวิวัฒน์ และ จากผลการทดลอง แสดงให้เห็นว่าขั้นตอนวิธีการบรรจวบสามารถแก้ปัญหาซูโดะกุได้แม่นยำและรวดเร็วขึ้น เมื่อเทียบกับขั้นตอนวิธีในกลุ่มขั้นตอนวิธีเดียวกัน ภายในจำนวนประชากร และจำนวนรุ่นค้นหาที่เท่ากัน

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต.....

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก .....

ปีการศึกษา.....2555.....

5470132421 : MAJOR COMPUTER ENGINEERING

KEYWORDS : Coincidence Algorithm / Sudoku Puzzle / Multimodal Optimization / Estimation of Distribution Algorithm / Genetic Algorithm

KIATSOPON WAIYAPARA : SOLVING SUDOKU PROBLEMS BY COINCIDENCE ALGORITHM. ADVISOR : PROF. PRABHAS CHONGSTITVATANA, Ph.D., 72 pp.

In this dissertation, we proposed solving a kind of combinatorial problem, Sudoku, by Coincidence Algorithm, COIN. The algorithm is in a class of estimation of distribution algorithms in evolutionary computation. The algorithm makes use of both better and worse genetic materials for finding satisfactory solutions. COIN can outperform many traditional evolutionary algorithms in part of combinatorial optimization. Especially, in experimental results of Sudoku puzzles, Coin converges to the exact solution in fewer number of evaluation functions with more accuracy comparing to the traditional algorithms in the same experimental setting.

Department: Computer Engineering Student's Signature.....

Field of Study: Computer Engineering Advisor's Signature.....

Academic Year: 2012

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี โดยความกรุณาอย่างสูงจาก ศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา ซึ่งเป็นอาจารย์ที่ปรึกษาวิทยานิพนธ์ ท่านได้กรุณาสละเวลาดูแลและให้คำปรึกษาทางด้านวิชาการงานวิจัย โดยให้คำแนะนำเกี่ยวกับการทำวิจัย และช่วยชี้แนะแนวทางการแก้ไขปัญหาต่างๆ ที่เกิดขึ้นระหว่างการทำวิจัย จึงขอกราบขอบพระคุณเป็นอย่างสูง ณ โอกาสนี้

ขอกราบขอบพระคุณคณะกรรมการสอบวิทยานิพนธ์ทุกท่านเป็นอย่างสูง ได้แก่ ผู้ช่วยศาสตราจารย์ ดร. สุกรี สินธุภิญโญ และ รองศาสตราจารย์ ดร. พีรยุทธ์ ชาญเศรษฐิกุล ในการตรวจแก้ ให้ข้อคิดและคำแนะนำอันเป็นประโยชน์ยิ่งต่องานวิจัย

ขอขอบพระคุณอาจารย์ประจำภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัยทุกท่านเป็นอย่างสูงที่ให้ข้อคิดและแนวทางในการวิจัย ขอขอบคุณเจ้าหน้าที่ประจำภาควิชาวิศวกรรมคอมพิวเตอร์ทุกท่าน รวมถึงเพื่อนๆ และพี่น้องทุกคนในห้องวิจัย ISEL Lab, MIND Lab, SPACE Lab โดยเฉพาะ Intelligence System Lab (ISL) ที่ให้ความช่วยเหลือคำแนะนำ และยังให้กำลังใจตลอดมา ขอบคุณ นายวรินทร์ วัฒนพรพรหมณ์ นายอลงกต บุรุษอาชาไนย และ นายพีระ ตันธีรวงศ์ ที่เสียสละเวลาในการให้คำแนะนำต่างๆที่เป็นประโยชน์ต่องานวิจัย

ขอกราบขอบพระคุณ นาย ประดับ ไวยพารา และ นาง รัตนรร หงษาศิริพรรณ บิดามารดาของผู้วิจัยซึ่งให้การสนับสนุน คอยเป็นห่วงเป็นใย และให้กำลังใจแก่ผู้วิจัยเสมอมา นอกจากนี้ผู้วิจัยขอขอบคุณทุกท่านที่มีส่วนช่วยเหลือทำให้วิทยานิพนธ์ฉบับนี้สำเร็จเรียบร้อยลงได้ด้วยดีทุกประการ

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง
บทคัดย่อภาษาอังกฤษ ..... จ
กิตติกรรมประกาศ ..... ฉ
สารบัญ ..... ช
สารบัญตาราง ..... ฌ
สารบัญภาพ ..... ญ

บทที่

1 บทนำ 1
1.1 ความเป็นมาและความสำคัญของปัญหา 1
1.2 วัตถุประสงค์ของการวิจัย 3
1.3 ขอบเขตของการวิจัย 3
1.4 ขั้นตอนและวิธีดำเนินงานวิจัย 3
1.5 ประโยชน์ที่คาดว่าจะได้รับ 4
2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 5
2.1 การหาค่าเหมาะที่สุดเชิงการจัด 5
2.1.1 ความรู้พื้นฐานเกี่ยวกับการหาค่าเหมาะที่สุดเชิงการจัด 5
2.1.2 เอ็นพีบริบูรณ์ และ เอ็นพีแบบยาก 7
2.1.3 ขั้นตอนวิธีเชิงวัฒน์สำหรับการหาค่าเหมาะที่สุดเชิงการจัด 8
2.1.4 ปัญหาถุงเป้ 10
2.1.5 ปัญหาการเดินทางของพนักงานขาย 16
2.2 การหาค่าเหมาะที่สุดหลากหลายรูปแบบ 22
2.3 ขั้นตอนวิธีการบรรจวบ 24
2.4 ขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบบัพ 27
2.5 ปัญหาซูโดะกุ 29
2.6 งานวิจัยที่เกี่ยวข้อง 32

บทที่ หน้า

2.6.1 ขั้นตอนวิธีการบรรจวบกับปัญหา TSPs 32
2.6.2 ขั้นตอนวิธีการบรรจวบกับปัญหาหลากหลายรูปแบบ 34
2.6.3 ขั้นตอนวิธีเชิงพันธุกรรมแบบที่ 1 กับปัญหาซูโดะกุ 35
2.6.4 ขั้นตอนวิธีเชิงพันธุกรรมแบบที่ 2 กับปัญหาซูโดะกุ 38
2.6.5 ขั้นตอนวิธีเชิงพันธุกรรมแบบที่ 3 กับปัญหาซูโดะกุ 40
2.6.6 ขั้นตอนวิธีประมาณการแจกแจงกับปัญหาซูโดะกุ 43
3 วิธีดำเนินการวิจัย 45
3.1 ภาพรวมทั้งหมดของงานวิจัย 45
3.2 การศึกษาประสิทธิภาพการทำงานของขั้นตอนวิธีการบรรจวบ 47
3.3 การแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบ 47
3.4 การวัดค่าประสิทธิภาพ 53
4 การทดลองและผลการทดลอง 54
4.1 เครื่องมือที่ใช้ในการทดลอง 54
4.2 ผลการทดลอง 59
5 สรุปผลการวิจัย 62
5.1 บทสรุป 62
5.2 ข้อเสนอแนะ 63
รายการอ้างอิง 64
ภาคผนวก 69
ประวัติผู้เขียนวิทยานิพนธ์ 72

สารบัญตาราง

ตารางที่ หน้า

2-1 ตัวอย่างการซ่อมแซมในปัญหาถุงเป้ 13
2-2 ตัวอย่างการเปรียบเทียบขนาดของปริภูมิค้นหาระหว่างการเข้ารหัสฐานสอง และการเรียงสับเปลี่ยน.. 15
2-3 ตัวอย่างการถอดรหัสโครโซม (3,2,4,2,2,1) แบบการเข้ารหัสเชิงประชิตในปัญหา TSP 18
2-4 ผลการทดลองแสดงผลเฉลยที่พบของแต่ละขั้นตอนวิธีในปัญหา Gröstel24 33
2-5 ผลการทดลองแสดงจำนวนของรุ่นที่ใช้ค้นหาในปัญหา Gröstel24 33
2-6 ผลการทดลองโดยเปรียบเทียบซอง COIN ในปัญหาเชิงการจัดต่างๆ 35
2-7 ผลการทดลองแสดงจำนวนครั้งที่พบผลเฉลยที่เหมาะที่สุด และ ค่าทางสถิติที่อ้างอิงตามจำนวนของรุ่นที่พบผลเฉลยที่เหมาะที่สุดของ GA สำหรับปัญหาซูโดะกุ 38
2-8 ผลการทดลองเปรียบเทียบระหว่างวิธีก่อนหน้า วิธีปัจจุบัน และ วิธีปัจจุบันที่ไม่จำกัดจำนวนรุ่นในการค้นหาผลเฉลยที่เหมาะที่สุด 39
2-9 ผลการทดลองในระดับความยากที่แตกต่างกันเทียบกับวิธีการที่เลือกใช้ 43
2-10 ผลการทดลองเมื่อเทียบกับงานวิจัยก่อนหน้า 43
4-1 ผลการทดลองการหาค่าเหมาะที่สุดหลากหลายรูปแบบ 59
4-2 การกำหนดค่าพารามิเตอร์ต่างๆสำหรับการทดลองการแก้ปัญหาซูโดะกุ 60
4-3 ผลการทดลองการแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบ 61

สารบัญภาพ

ภาพที่ หน้า

2-1 ความสัมพันธ์ระหว่างพี, เอ็นพี, เอ็นพีบริบูรณ์, และ เอ็นพีแบบยาก 8
2-2 กราฟแสดงคุณภาพของคำตอบ ของแต่ละกลุ่มขั้นตอนวิธี 9
2-3 กราฟแสดงค่าความซับซ้อนเชิงเวลา ของแต่ละกลุ่มขั้นตอนวิธี 10
2-4 กราฟแสดงจำนวนงานวิจัยทางด้านการหาค่าเหมาะที่สุดเชิงการจัด
โดยขั้นตอนวิธีเชิงวิวัฒน์จาก SCI 10
2-5 ตัวอย่างรูปแบบโครโมโซมที่เข้ารหัสฐานสองสำหรับปัญหาถุงเป้ 11
2-6 กราฟของฟังก์ชันการลงโทษสำหรับปัญหาถุงเป้ 12
2-7 ตัวอย่างแสดงการไขว้เปลี่ยนสำหรับการเข้ารหัสเชิงความยาวตัวแปร 14
2-8 ตัวอย่างการทำงานของการกลายพันธุ์เชิงแลกเปลี่ยน 19
2-9 ตัวอย่างการทำงานของการกลายพันธุ์แบบแทรก 19
2-10 ตัวอย่างการทำงานของการกลายพันธุ์แบบเคลื่อนเป็นกลุ่ม 19
2-11 ตัวอย่างการทำงานของการกลายพันธุ์เชิงผกผันอย่างง่าย 20
2-12 ตัวอย่างการทำงานของการกลายพันธุ์เชิงผกผัน 20
2-13 ตัวอย่างวิธีการไขว้เปลี่ยนแบบจัดคู่เป็นส่วน 21
2-14 ตัวอย่างวิธีการไขว้เปลี่ยนเชิงลำดับ 22
2-15 ตัวอย่างวิธีการไขว้เปลี่ยนเชิงตำแหน่ง 22
2-16 การหาค่าเหมาะที่สุดหลากหลายรูปแบบในกรณีหลายผลเฉลยที่เหมาะที่สุด 23
2-17 การหาค่าเหมาะที่สุดหลากหลายรูปแบบในกรณีหลายผลเฉลยคล้ายที่เหมาะที่สุด 23
2-18 ตัวอย่างเมทริกซ์ความน่าจะเป็นร่วม 24
2-19 ตัวอย่างเมทริกซ์ NHM^t สำหรับ $P(t)$ ที่ $N = 5, L = 5$ และ $B_{ratio} = 0.2$ 28
2-20 ตัวอย่างปัญหาซูโดะกุที่มีขนาด 9×9 29
2-21 ตัวอย่างปัญหาซูโดะกุขนาด 9×9 และ ผลเฉลยที่เหมาะที่สุดสำหรับปัญหาดังกล่าว 30
2-22 ตัวอย่างปัญหาซูโดะกุขนาด 9×9 ในจำนวนตัวเลขเริ่มต้นที่แตกต่างกัน 30
2-23 ตัวอย่างของผลเฉลยเหมาะที่สุดท้องถิ่น (a), (b) และ (c) และผลเฉลยเหมาะที่สุด (d)
ของปัญหาซูโดะกุปัญหาหนึ่ง 31

ภาพที่ หน้า

2-24 กราฟแสดงผลการทดลองขั้นตอนวิธีการบรรจวบเมื่อเกิดการเรียนรู้เชิงบวกหรือเชิงลบ อย่างใดอย่างหนึ่งเทียบกับการเรียนรู้ไปพร้อมๆกัน 32
2-25 ตัวอย่างของแต่ละผลเฉลยที่แทนด้วยสายจำนวนเต็ม ซึ่งแทนโจทย์เริ่มต้น และ จุดที่ สามารถทำการไขว้เปลี่ยนซึ่งตรงกับรอยต่อระหว่างสายย่อยแต่ละสายของเฉลย 36
2-26 วิธีการกลายพันธุ์ทั้งสามรูปแบบ 36
2-27 ตัวอย่างการไขว้เปลี่ยนจากแต่ละแถวและหลักในระดับตารางย่อย 41
2-28 แนวคิดการหรับปรุงการค้นหาเฉพาะที่ 42
3-1 ผังงานแสดงโครงสร้างโดยรวมของงานวิจัย 46
3-2 ตัวอย่างของแต่ละผลเฉลย (b) ที่สร้างโดยแบบจำลองความน่าจะเป็น (a) 47
3-3 แบบจำลองความน่าจะเป็นของขั้นตอนวิธีการบรรจวบแบบบัพที่มีขนาดของปัญหา เท่ากับ 5 และ ผลเฉลยที่เหมาะที่สุดที่สร้างจากแบบจำลองดังกล่าว 48
3-4 ลักษณะของการแทนแบบจำลองความน่าจะเป็นสำหรับปัญหาซูโดะกุ 50
3-5 ผังงานแสดงขั้นตอนการยกเลิกการค้นหาบางสมาชิกของแบบจำลองความน่าจะเป็น 51
3-6 ความสัมพันธ์ระหว่างตารางย่อยของปัญหาซูโดะกุและแบบจำลองความน่าจะเป็นของ ขั้นตอนวิธีการบรรจวบแบบบัพ 52
4-1 ปัญหา TSPs สำหรับการหาค่าเหมาะที่สุดหลากหลายรูปแบบ 55
4-2 ปัญหาซูโดะกุระดับง่ายที่ใช้ในการทดลอง 56
4-3 ปัญหาซูโดะกุระดับปานกลางที่ใช้ในการทดลอง 57
4-4 ปัญหาซูโดะกุระดับยากที่ใช้ในการทดลอง 58
4-5 กราฟแสดงพฤติกรรมการค้นหาผลเฉลยของขั้นตอนวิธีการบรรจวบ 61

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การคำนวณเชิงวิวัฒน์ (Evolutionary Computation) เป็นวิธีการคำนวณประยุกต์ตามความรู้จากธรรมชาติ โดยที่รูปแบบวิวัฒนาการของสิ่งมีชีวิตถูกใช้ออกแบบกระบวนการทำงาน [1] เป็นการปรับตัวเพื่อให้อยู่รอดจากการคัดเลือกโดยธรรมชาติ องค์ประกอบหลักของการทำงานถูกแยกเป็นส่วนสำคัญ [1] คือ ประชากร (Population) ซึ่งประกอบไปด้วยผลเฉลยที่เป็นไปได้จำนวนมากที่มีหน้าตาแตกต่างกันในแต่ละรุ่น (Generation) เพื่อใช้ในการเรียนปัญหาและนำมาปรับปรุงรูปแบบของประชาการในรุ่นถัดไป ต่อมาคือ ค่าความเหมาะสม (Fitness) ใช้สำหรับอธิบายคุณภาพของแต่ละผลเฉลยในประชากร กระบวนการคัดเลือกจะคัดเลือกบางส่วนตามค่าความเหมาะสม การดำเนินการในแต่ละขั้นตอนวิธี ใช้รูปแบบการดำเนินการที่แตกต่างกัน เช่น ขั้นตอนวิธีอาณานิคมมด จะมีการดำเนินการค้นหาผลเฉลยโดยเลียนแบบพฤติกรรมการปล่อยฟีโรโมนของมดเพื่อเลือกเส้นทาง [2] หรือ ขั้นตอนวิธีเชิงพันธุกรรม (Genetic algorithm) ในส่วนของกระบวนการดำเนินการหาผลเฉลย ประยุกต์จากวิธีการไขว้เปลี่ยนโครโมโซมและการกลายพันธุ์ เช่น เลือกผลเฉลยที่มีค่าความเหมาะสมคู่หนึ่ง พร้อมกับทำการไขว้เปลี่ยนค่าบางส่วนระหว่างกันเป็นต้น

กลุ่มปัญหาหนึ่งที่ได้รับความนิยม คือ ปัญหาเชิงการจัด (Combinatorial Problem) เรียกว่า การหาค่าเหมาะที่สุดเชิงการจัด (Combinatorial Optimization) รูปแบบของปัญหาดังกล่าวมีอยู่หลายแบบด้วยกัน เช่น การจัดหมู่ การจัดวาง การเรียงลำดับ การจัดกลุ่ม เป็นต้น [1][3] โดยตัวอย่างของปัญหา คือ การจัดตารางเวลา เช่น ปัญหาการจัดตารางการผลิตสำหรับระบบการผลิตแบบไหล [4] (Flowshop scheduling) ในส่วนของตัวอย่างของปัญหาการจัดกลุ่มได้แก่ ปัญหาการบรรจุผลิตภัณฑ์ (Bin packing) เป็นต้น และ ปัญหาการจัดกลุ่ม (Grouping problem) เป็นต้น ในส่วนของขั้นตอนวิธีต่างๆที่นำมาใช้แก้ปัญหาสามารถแบ่งได้ดังนี้ [1] คือ ขั้นตอนวิธีแม่นตรง (Exact algorithms) ขั้นตอนวิธีฮิวริสติค (Heuristic algorithms) และ ขั้นตอนวิธีเชิงวิวัฒน์ (Evolutionary algorithms) ซึ่งแต่ละกลุ่มขั้นตอนวิธีมีลักษณะเฉพาะแตกต่างกันไป เช่น วิธีแม่นตรงยืนยันได้ว่าผลเฉลยที่ได้คือผลเฉลยที่เหมาะที่สุด เมื่อเทียบกับวิธีเชิงวิวัฒน์ ผลเฉลยที่พบอาจไม่ใช้ผลเฉลยที่เหมาะที่สุด ซึ่งเป็นผลเฉลยที่อยู่ในกลุ่มคำตอบที่ดีเท่านั้น แต่ถ้าปัญหาดังกล่าวเป็นปัญหาประเภทเอ็นพีบริบูรณ์ (NP-Complete) วิธีแม่นตรงจะไม่สามารถค้นหา

ผลเฉลยที่ต้องการภายในเวลาที่เป็นฟังก์ชันพหุนาม (Polynomial time) แต่วิธีเชิงวิวัฒน์ยังคงสามารถค้นหาผลเฉลยที่เหมาะที่สุด หรือใกล้เคียงได้โดยใช้เวลาในการคำนวณน้อยกว่า กล่าวคือ เวลาในการคำนวณไม่ได้สัมพันธ์กับขนาดของปัญหานั่นเอง

งานวิจัยด้านการหาค่าเหมาะที่สุดเชิงการจัด ขั้นตอนวิธีเชิงพันธุกรรมเป็นวิธีหนึ่งที่ได้รับความนิยมเป็นอันดับต้น อย่างไรก็ตามวิธีนี้ยังคงมีความสามารถพื้นฐานที่จำกัด เช่น การไขว้เปลี่ยน (Crossover operator) จำเป็นต้องเพิ่มกระบวนการซ่อมแซมประชากร (Repair function) เนื่องจากการแลกเปลี่ยนข้อมูลระหว่างกันของผลเฉลยพ่อแม่ทำให้เกิดผลเฉลยใหม่ที่ไม่สมเหตุสมผลขึ้นซึ่งจำเป็นต้องอาศัยวิธีที่จำเพาะมากขึ้นสำหรับแก้ปัญหาดังกล่าว ซึ่งมีการเสนอวิธีการแก้ปัญหาอยู่หลายแบบด้วยกัน เช่น การไขว้เปลี่ยนแบบจัดคู่เป็นส่วน (Partially Mapped Crossover หรือ PMX) [5] ในปัญหาการเดินทางของพนักงานขาย (Traveling Salesman Problems หรือ TSPs) เป็นต้น รวมถึง การกลายพันธุ์ (Mutation Operator) ที่จำเพาะมากขึ้นเช่นกัน ในบางงานวิจัยได้มีการกล่าวถึงปัญหาทางด้านปัญหาเส้นเชื่อม (Linkage problem) ซึ่งเกิดขึ้นเมื่อแต่ละองค์ประกอบของผลเฉลยของประชากรในทางสถิติ ไม่ขึ้นต่อกันเลยเมื่ออ้างอิงตามฟังก์ชันความเหมาะสม (Fitness function) [6] มีการเสนอกลุ่มขั้นตอนวิธีเชิงวิวัฒน์รูปแบบใหม่ คือ ขั้นตอนวิธีประมาณการแจกแจง (Estimation of Distribution Aogrithms หรือ EDAs [7][,8]) ซึ่งมีวิธีสร้างประชากรในลักษณะที่แตกต่างกับวิธีแบบเดิม กล่าวคือ มีการใช้แบบจำลองความน่าจะเป็น (Probabilistic model) ในการสร้างประชากร ซึ่งวิธีดังกล่าวสามารถอธิบายได้ว่าเป็นลักษณะการไขว้เปลี่ยนรูปแบบหนึ่งจากหลากหลายพ่อแม่ (Multiparent crossover operator) [6] ทั้งนี้การประยุกต์ใช้ขั้นตอนวิธีในกลุ่ม EDAs สำหรับปัญหาเชิงการจัดสามารถทำได้โดยไม่จำเป็นต้องอาศัยการออกแบบที่ซับซ้อนมากนัก เนื่องจากรูปแบบการทำงานของ EDAs สามารถสร้างประชากรโดยไม่ต้องผ่านกระบวนการซ่อมแซม อย่างไรก็ตาม เพื่อให้ประสิทธิภาพในการหาผลเฉลย ยังต้องออกแบบให้การทำงานมีความจำเพาะเช่น การทำการค้นหาแบบเฉพาะที่ (Local search) เป็นต้น โดยขั้นตอนวิธีในกลุ่มนี้มีอยู่หลายวิธีด้วยกัน [9] เช่น population-based incremental learning (PBIL) [10]. Univariate marginal distribution algorithm (UMDA) [11], compact genetic algorithm (cGA) [12], mutual Information maximizing input clustering (MIMIC) [13] และขั้นตอนวิธีการบรรจบ (Coincidence Algorithm หรือ COIN) [14] เป็นต้น

ขั้นตอนวิธีการบรรจบ ถูกจัดอยู่ในกลุ่มขั้นตอนวิธีการประมาณการแจกแจง ซึ่งอาศัยแบบจำลองความน่าจะเป็นที่เรียกว่าเมทริกซ์ความน่าจะเป็นร่วม (Joint probability matrix หรือ ตาราง H [14]) มีการเรียนรู้จากประชากรที่มีค่าความเหมาะสมสูงกว่าเกณฑ์ และ จากประชากรที่มีค่าความเหมาะสมต่ำกว่าเกณฑ์ เรียกว่า การเรียนรู้สหสัมพันธ์ลูกผสมเชิงลบ (Hybrid Negative

Correlation Learning) [15] หรือ การเรียนรู้สหสัมพันธ์เชิงลบ (Negative Correlation Learning) [14] มีการพิสูจน์ในระดับหนึ่งแล้วว่า ขั้นตอนวิธีดังกล่าวสามารถแก้ปัญหาเชิงการจัดได้ดีทั้งในเชิงปริมาณ และ คุณภาพ เมื่อเทียบกับขั้นตอนวิธีในกลุ่มขั้นตอนวิธีเชิงวิวัฒน์ ซึ่งตัวอย่างปัญหาที่ใช้ทดสอบ ได้แก่ ปัญหาการเดินม้า (Knight's Tour Problems), ปัญหาการเดินทางของพนักงานขาย, ปัญหาการจัดลำดับการผลิตบนสายการประกอบผลิตภัณฑ์ผสมแบบตัวยู (Sequencing Problems on Mixed-Model U-Shaped Assembly Lines) [16] เป็นต้น

ซูโดะกุ (Sudoku) [17] เป็นปัญหาหนึ่งในปัญหาการหาค่าเหมาะที่สุดเชิงการจัด [18] ที่ได้รับความนิยมอย่างแพร่หลายโดยเฉพาะในยุโรปและอเมริกาเหนือ [19] และได้รับความสนใจนำมาแก้ปัญหาโดยขั้นตอนวิธีเชิงวิวัฒน์เช่นกัน ขั้นตอนวิธีเชิงพันธุกรรมได้รับความนิยมในการประยุกต์กับปัญหานี้ [18]-[21] มีการพัฒนาปรับปรุงอย่างต่อเนื่อง ซึ่งส่งผลให้คุณภาพในการหาผลเฉลยเป็นไปในทางที่ดีขึ้น อย่างไรก็ตาม บางงานวิจัยใช้การค้นหาเฉพาะที่เพื่อให้ได้ประสิทธิภาพมากขึ้น [21] ดังนั้น จึงมีความสนใจในการนำขั้นตอนวิธีการบรรจวบ ที่สามารถแก้ปัญหาเชิงการจัดได้อย่างมีประสิทธิภาพ มาประยุกต์ใช้เพื่อแก้ปัญหาดังกล่าว โดยเปรียบเทียบประสิทธิภาพการทำงานตามความแม่นยำที่พบ และ จำนวนครั้งที่เรียกใช้ฟังก์ชันการประเมินค่า (Evaluation Function) ภายใต้จำนวนประชากรที่ใช้ และ จำนวนมากสุดของรุ่นประชากรที่ยอมรับได้ ที่เท่ากัน

1.2 วัตถุประสงค์ของการวิจัย

เพื่อศึกษา ออกแบบและนำเสนอ การทำงานของขั้นตอนวิธีการบรรจวบ สำหรับแก้ปัญหาซูโดะกุที่อยู่ในกลุ่มปัญหาเชิงการจัด ได้อย่างมีประสิทธิภาพ

1.3 ขอบเขตของการวิจัย

1. ออกแบบการทำงานของขั้นตอนวิธีการบรรจวบให้สามารถแก้ปัญหาซูโดะกุได้ โดยอาศัยวิธีในงานวิจัยก่อนหน้าเพื่อปรับใช้ร่วมกับการออกแบบ
2. พัฒนาขั้นตอนวิธีการบรรจวบบนระบบปฏิบัติการวินโดว์ (Windows) ผ่าน Microsoft Visual Studio 2010 โดยใช้ภาษา c++

1.4 ขั้นตอนและวิธีดำเนินการวิจัย

1. ศึกษาค้นคว้าทฤษฎี และ หลักการพื้นฐานของขั้นตอนวิธีเชิงวิวัฒน์ รวมถึงทฤษฎีพื้นฐานอื่นๆ ที่เกี่ยวข้อง

2. ศึกษากระบวนการทำงานของขั้นตอนวิธีการบรรจวบ
3. สำรวจงานวิจัยที่เกี่ยวข้องกับการนำขั้นตอนวิธีเชิงวิวัฒน์มาแก้ปัญหาซูโดะกุ รวมถึงขั้นตอนวิธีอื่นๆอีกด้วย
4. ศึกษาวิธีการแก้ปัญหาซูโดะกุด้วยขั้นตอนวิธีเชิงพันธุกรรม
5. ออกแบบและปรับปรุงขั้นตอนวิธีการบรรจวบสำหรับการแก้ปัญหาซูโดะกุ
6. จัดทำ (implementation) ขั้นตอนวิธีการบรรจวบ
7. วิเคราะห์และสรุปผลการทดลอง

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ทำให้ทราบถึงประสิทธิภาพการทำงานของขั้นตอนวิธีการบรรจวบสำหรับแก้ปัญหาซูโดะกุ
2. ทำให้ทราบถึงความสามารถในการแก้ปัญหาของขั้นตอนวิธีการบรรจวบที่มีต่อกลุ่มปัญหาเชิงการจัดเพิ่มมากขึ้น
3. ได้รับแนวคิดการออกแบบที่แตกต่างจากการแก้ปัญหาในงานวิจัยก่อนหน้าของขั้นตอนวิธีการบรรจวบสำหรับแก้ปัญหาซูโดะกุ

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในส่วนนี้เป็นรายละเอียดต่างๆที่ควรทำความเข้าใจเพื่อใช้เป็นแนวคิดพื้นฐานในการแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบ ซึ่งจะกล่าวถึงแนวความคิดต่างๆในการแก้ปัญหาเชิงการจัด โดยขั้นตอนวิธีเชิงวิวัฒน์ รวมถึงการแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีเชิงวิวัฒน์อย่างขั้นตอนวิธีเชิงพันธุกรรม และสุดท้ายอธิบายขั้นตอนวิธีที่ใช้ในงานวิจัยนี้

2.1 การหาค่าเหมาะที่สุดเชิงการจัด

ในส่วนนี้เป็นรายละเอียดต่างๆที่ควรทำความเข้าใจเกี่ยวกับปัญหาการหาค่าเหมาะที่สุดเชิงการจัด โดยเนื้อหาที่นำเสนอในบทนี้ประกอบด้วย เนื้อหาเบื้องต้นเกี่ยวกับการหาค่าเหมาะที่สุดเชิงการจัด ปัญหาเอ็นพีแบบยาก ปัญหาเอ็นพีบริบูรณ์ ขั้นตอนวิธีเชิงวิวัฒน์สำหรับปัญหาการหาค่าเหมาะที่สุดเชิงการจัด นอกจากนี้ยังมีการเพิ่มรายละเอียดตัวอย่างปัญหาเชิงการจัด ได้แก่ ปัญหาถุงเป้ ปัญหาการเดินทางของพนักงานขาย รวมถึง ปัญหาซูโดะกุที่ถูกใช้เป็นปัญหาหลักในวิทยานิพนธ์นี้อีกด้วย

2.1.1 ความรู้พื้นฐานเกี่ยวกับการหาค่าเหมาะที่สุดเชิงการจัด

การหาค่าเหมาะที่สุดเชิงการจัด [1] เป็นการหาค่าเหมาะที่สุดที่แยกแขนงมาจากภิณทนคณิตศาสตร์ (Discrete Mathematics) ในด้านการจัดโดยเฉพาะ ประกอบไปด้วยส่วนที่สำคัญต่างๆ คือ การแจงนับ (Enumeration) การจัดหมู่ (Combination) การเรียงสับเปลี่ยน (Permutation) เป็นต้น และมีรายละเอียดดังนี้ โดยสมมติว่ามีเซต S ที่มีขนาดเท่ากับ n

ในส่วนของการแจงนับของเซต S คือ รายการตรงของสมาชิกทั้งหมดภายในเซตดังกล่าว และ ในส่วนของการจัดหมู่ คือ การเลือกสมาชิก k ตัว แบบไม่สนใจลำดับก่อนหลังจากเซต S ซึ่งจำนวนของคำตอบที่เป็นไปได้ทั้งหมดในแต่ละสถานการณ์สามารถแทนได้ดังสมการ

$$C_n^k = \frac{n!}{k!(n-k)!} \tag{2-1}$$

ซึ่ง $n!$ มีค่าเท่ากับ $n \times (n-1) \times (n-2) \times ... \times (1)$ ตัวอย่างเช่น จำนวนทั้งหมดที่เป็นไปได้ของการจัดหมู่ที่มีจำนวนสมาชิกเท่ากับ 2 ของเซต $S = \{w, x, y, z\}$ คือ C_4^2 ซึ่งมีค่าเท่ากับ 6 ประกอบไปด้วย $\{w, x\}, \{w, y\}, \{w, z\}, \{x, y\}, \{x, z\},$ และ $\{y, z\}$ ในส่วนของการเรียง

สับเปลี่ยน เป็นชุดของผลเฉลยที่ประกอบไปด้วยสมาชิก k ตัว ที่มาจากเซต S และ จำนวนของผลเฉลยที่เป็นไปได้ทั้งหมดของแต่ละสถานการณ์สามารถแทนได้ดังสมการ

$$P_n^k = \frac{n!}{(n-k)!} \quad \text{(2-2)}$$

โดยตัวอย่างของการเลือกผลเฉลยที่มีจำนวนสมาชิกเท่ากับ 2 จากเซต $S = \{w, x, y, z\}$ คือ $P_4^2 =$ 12 โดยคำตอบที่เป็นไปได้ทั้งหมดมีดังนี้คือ $\{w,x\}, \{x,w\}, \{w,y\}, \{y,w\}, \{w,z\}, \{z,w\}, \{x,y\}, \{y,x\}, \{x,z\}, \{z,x\}, \{y,z\},$ และ $\{z,y\}$

ปัญหาการหาค่าเหมาะที่สุดปัญหาหนึ่งที่สัมพันธ์กับการจัดหมู่ที่เหมาะที่สุดของกลุ่มสมาชิก เรียกว่าปัญหาการจัดกลุ่ม (Grouping problems) [1] ซึ่งตัวอย่างปัญหาในกลุ่มนี้ คือ ปัญหาการจัดวางบรรจุภัณฑ์ (Bin packing problems) สมมติว่ามีของอยู่ n ชิ้น โดยชิ้นที่ i มีขนาดปริมาณเท่ากับ C_i และ ยังสามารถมีได้หลายบรรจุภัณฑ์โดยมีขนาดปริมาณที่รับได้สูงสุดเท่ากับ B ผลเฉลยเหมาะที่สุดที่ต้องการ คือ จำนวนน้อยสุดของบรรจุภัณฑ์ที่ใช้เพื่อที่จะให้ของทุกชิ้นสามารถบรรจุลงไปในบรรจุภัณฑ์ได้ และ ผลรวมของปริมาณของทุกชิ้นที่บรรจุในแต่ละบรรจุภัณฑ์มีค่าไม่เกิน B โดยในมุมมองของของปัญหาการจัดกลุ่ม ปัญหาดังกล่าวเป็นการจัดกลุ่มจากของ n ชิ้น ที่อยู่ภายใต้ข้อจำกัดของขนาดของกลุ่มและจำนวนน้อยสุดของกลุ่มที่ใช้

การหาค่าเหมาะที่สุดที่สัมพันธ์กับการเรียงสับเปลี่ยนที่เหมาะที่สุดของสมาชิกเรียกว่าปัญหาการจัดกำหนดการ (Scheduling problems) [1] โดยตัวอย่างปัญหาในกลุ่มนี้คือ ปัญหาการจัดตารางการผลิตสำหรับระบบการผลิตแบบไหล (Flow-shop Scheduling problems) สมมติว่ามีอยู่ n งาน แต่ละงานจำเป็นต้องผ่านกระบวนการทำงานทั้งหมด m เครื่องจักร ตั้งแต่เครื่องจักรที่ 1 จนถึงเครื่องจักรที่ m ตามลำดับ โดยเวลาในการดำเนินการของงานที่ i ในเครื่องจักรที่ k แทนด้วย p_{ik} ซึ่งผลเฉลยที่ต้องการคือเวลาที่ใช้ในการดำเนินการโดยรวมน้อยสุดจากการปรับเปลี่ยนลำดับงาน โดยในมุมมองของปัญหาการจัดกำหนดการ ปัญหาที่ยกตัวอย่างข้างต้นคือการจัดกำหนดการก่อนหลังของแต่ละงานเพื่อให้ได้มาซึ่งเวลาดำเนินงานที่น้อยที่สุด

ปัญหาการหาค่าเหมาะที่สุดเชิงการจัด มีหลากหลายแนวทางสำหรับแทนแต่ละผลเฉลย เช่น ปัญหาการจัดวางบรรจุภัณฑ์สามารถแทนด้วยเลขจำนวนเต็มที่มี n จำนวน โดยมีโดเมนของแต่ละเลขจำนวนเต็มในช่วงของ [1, k] โดยที่ k แทนจำนวนบรรจุภัณฑ์ที่มากที่สุดที่ยอมรับ และสามารถแทนแต่ละผลเฉลยที่เป็นไปได้ดังนี้ คือ {3,5,2,1,4} โดยมีความหมายว่า ให้บรรจุของแต่ละชิ้นตามตามหมายเลขที่กำหนดไว้จากซ้ายไปขวาตามลำดับ ลงในบรรจุภัณฑ์แรกจนเต็ม จึงสามารถเริ่มนำของที่เหลือบรรจุลงบรรจุภัณฑ์ถัดไปได้นั่นเอง ซึ่งหากว่าสามารถจำกัดตัวแปรต่างๆ ให้อยู่ในรูปจำนวนเต็มทั้งหมด การแทนปัญหาดังกล่าวสามารถเรียกได้ว่าเป็นปัญหากำหนดการ

จำนวนเต็ม (Integer programming) และ นอกจากนี้ถ้าการแทนปัญหาสามารถจำกัดอยู่ในช่วงจำนวนเต็ม 0 ถึง 1 จะสามารถเรียกได้ว่าเป็นกำหนดการจำนวนเต็มฐานสอง (0/1 programming หรือ Binary integer programming)

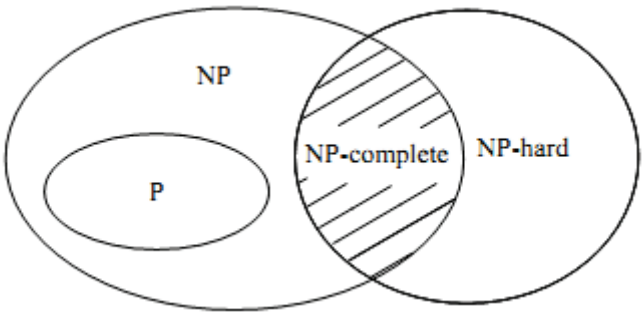
2.1.2 เอ็นพีบริบูรณ์ และ เอ็นพีแบบยาก

ในการหาค่าเหมาะที่สุดเชิงการจัด เวลาที่ใช้แก้ปัญหา สัมพันธ์กับขนาดของปัญหาและขั้นตอนวิธีที่เลือกใช้ และ สำหรับ ปัญหาการจัดตารางการผลิตสำหรับระบบการผลิตแบบไหลที่มี n งาน ถ้าใช้การเรียงสับเปลี่ยนในการแจกแจงทุกๆการเรียงสับเปลี่ยนเพื่อหาผลเฉลยที่เหมาะที่สุด ซึ่งจำเป็นต้องกระจายออกมาทั้งหมดเท่ากับ $n!$ เพื่อหาผลเฉลยดังกล่าว และ เพื่อที่จะเปรียบเทียบค่าความซับซ้อนเชิงเวลา (Time complexity) ในแต่ละขั้นตอนวิธีที่แตกต่างกันภายในปัญหาเดียวกัน จึงมีการใช้ $O(n!)$ หมายถึงค่าความซับซ้อนเชิงเวลาของขั้นตอนวิธีที่จำเป็นต้องใช้ในการหาผลเฉลยที่ต้องการเท่ากับ $n!$ นั่นเอง

ข้อสงสัยอีกอย่างหนึ่งที่สัมพันธ์กับประสิทธิภาพของแต่ละขั้นตอนวิธี คือ ความยากของปัญหาซึ่งนำไปสู่กรอบความคิดของเอ็นพีบริบูรณ์และเอ็นพีแบบยาก [1] โดยแปลงปัญหาการหาค่าเหมาะที่สุดให้อยู่ในรูปของปัญหาการตัดสินใจ (Decision problems) ซึ่งถ้าหากว่าขั้นตอนวิธีหนึ่ง สามารถค้นหาผลเฉลยที่เหมาะที่สุดในปัญหาหนึ่ง โดยใช้ค่าความซับซ้อนเชิงเวลาในระดับพหุนามเท่านั้น ปัญหาดังกล่าวจะถูกจัดอยู่ในกลุ่มปัญหาพี (P) โดยที่ P แทนค่าความซับซ้อนเชิงเวลาเชิงพหุนาม (Polynomial time complexity) และ ยังมีปัญหาการตัดสินใจอีกจำนวนมากที่ไม่สามารถค้นหาผลเฉลยที่เหมาะที่สุดได้ในเวลาเชิงพหุนาม โดยพิสูจน์ไม่ได้ว่าต้องใช้เวลาที่มีอัตราการเติบโตเกินกว่าอัตราเชิงพหุนามหรือไม่ กล่าวคือ ไม่ทราบว่าเป็นปัญหายากหรือง่าย อย่างไรก็ตาม หากเป็นปัญหาที่สามารถทวนสอบได้ในเวลาเชิงพหุนาม ปัญหาดังกล่าวเป็นปัญหาในกลุ่มเอ็นพี (NP) ดังนั้น สามารถสรุปความแตกต่างระหว่าง P และ NP ได้ว่า ปัญหาในกลุ่ม P สามารถใช้ขั้นตอนวิธีหนึ่งในการแก้ปัญหาได้อย่างรวดเร็ว แต่ ปัญหาในกลุ่ม NP ต้องทวนสอบผลเฉลยได้อย่างรวดเร็ว ซึ่งหากค้นหาผลเฉลยได้อย่างรวดเร็ว แปลว่าสามารถทวนสอบได้อย่างรวดเร็วเช่นกัน จึงสรุปได้ว่า P จัดอยู่ใน NP

ถ้าหากว่าสามารถออกแบบขั้นตอนวิธีหนึ่งที่สามารถหาผลเฉลยได้ในค่าความซับซ้อนเชิงเวลาเชิงพหุนาม เพื่อที่จะแปลงคำตอบใดๆของปัญหาตัดสินใจ D_1 ของปัญหาหนึ่งเป็นคำตอบเพียงหนึ่งเดียวของปัญหาตัดสินใจ D_2 ของอีกปัญหาหนึ่ง จะได้ว่าอย่างน้อยที่สุด D_1 มีความยากเท่ากับ D_1 โดยการเปลี่ยนรูปเชิงพหุนาม (Polynomial transformation) บางครั้งสามารถเรียกว่าเป็นการลดรูป (Reduction) และ ถ้าปัญหาการตัดสินใจ C อยู่ในกลุ่ม NP และมีขั้นตอนวิธีเชิง

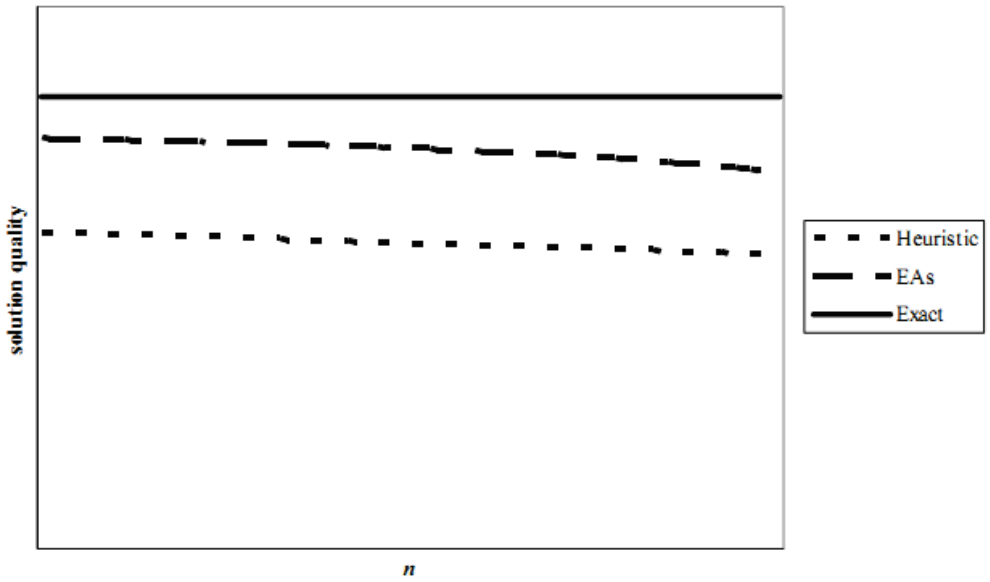
พหุนามที่สามารถแปลงทุกปัญหาในรูป NP ให้อยู่ในรูป C สามารถบอกได้ว่า C เป็น เอ็นพีบริบูรณ์ (NP-complete) และถ้าปัญหา H ไม่ได้อยู่ในกลุ่มเอ็นพีแต่อยู่ในเอ็นพีบริบูรณ์ C ที่ซึ่งสามารถแปลงในรูปของ H จะสามารถอธิบายได้ว่า H อยู่ในกลุ่มของเอ็นพีแบบยาก (NP-hard) ซึ่งมีความหมายว่าอย่างน้อยที่สุด H มีระดับความยากเท่ากับ C ซึ่งความสัมพันธ์ระหว่างสองกลุ่มสามารถแสดงได้ภาพที่ 2-1 [1]



ภาพที่ 2-1 ความสัมพันธ์ระหว่างพี, เอ็นพี, เอ็นพีบริบูรณ์, และ เอ็นพีแบบยาก [1]

2.1.3 ขั้นตอนวิธีเชิงวิวัฒน์สำหรับการหาค่าเหมาะที่สุดเชิงการจัด

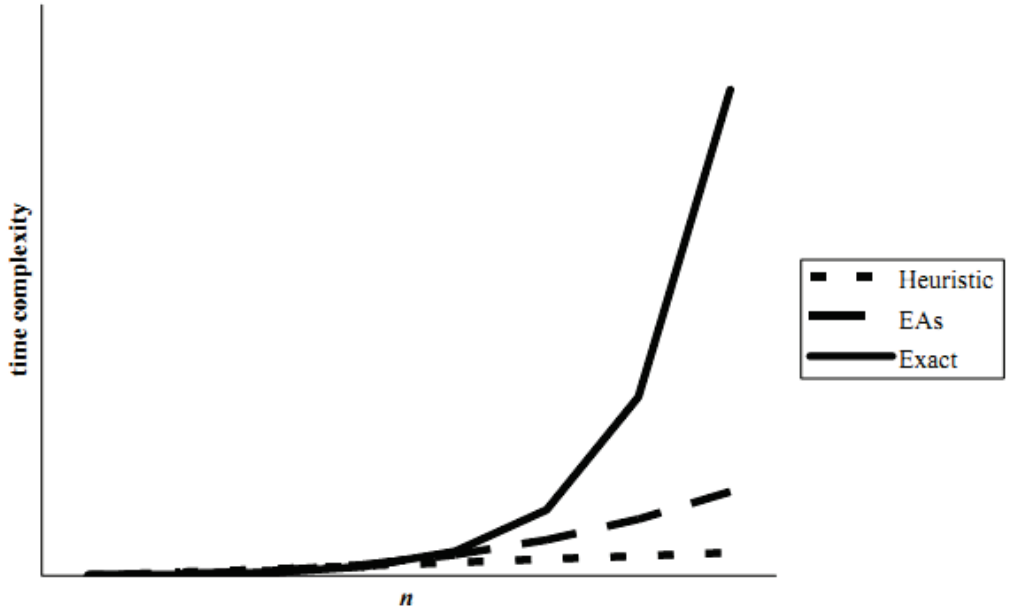
ขั้นตอนวิธีต่างๆสำหรับการหาค่าเหมาะที่สุดเชิงการจัดสามารถจำแนกประเภทออกเป็นสามกลุ่ม [1] ส่วนแรกคือ ขั้นตอนวิธีแม่นตรง (Exact algorithms) ซึ่งขั้นตอนวิธีในกลุ่มนี้รับประกันว่าคำตอบที่ได้เป็นคำตอบที่เหมาะที่สุด แต่เวลาที่ใช้ในการค้นหาไม่จัดอยู่ในความซับซ้อนเชิงเวลาเชิงพหุนามเมื่อนำขั้นตอนวิธีในกลุ่มนี้มาแก้ปัญหาประเภทเอ็นพีบริบูรณ์ ตัวอย่างขั้นตอนวิธีในกลุ่มนี้ ได้แก่ กำหนดการพลวัด (Dynamic programming), การขยายและจำกัดเขต (Branch and bound) เป็นต้น ส่วนถัดมาคือ ขั้นตอนวิธีฮิวริสติค (Heuristic algorithms) ซึ่งขั้นตอนวิธีในกลุ่มนี้สามารถให้ผลเฉลยที่เร็ว และ คุณภาพของคำตอบอยู่ในเกณฑ์ดี ซึ่งสามารถอธิบายได้อีกมุมมองหนึ่งว่าสามารถปรับปรุงผลเฉลยปัจจุบันให้ดีขึ้นได้อย่างรวดเร็ว โดยคุณภาพของผลเฉลยที่ได้จากขั้นตอนวิธีในกลุ่มนี้ในกรณีที่แย่ที่สุดมีคุณภาพที่ต่ำ และ ขั้นตอนวิธีเชิงวิวัฒน์ (Evolutionary algorithms) ซึ่งขั้นตอนวิธีในกลุ่มนี้สามารถค้นหาผลเฉลยที่มีคุณภาพเข้าใกล้คำตอบได้โดยใช้เวลาที่น้อยลงเมื่อเปรียบเทียบกับขั้นตอนวิธีอื่นๆ ซึ่งความแตกต่างที่สำคัญของทั้งสามกลุ่มดังกล่าว สามารถอธิบายได้ดังภาพที่ 2-2 [1] และ ภาพที่ 2-3 [1] โดย *n* คือ ขนาดของปัญหา ดังนี้



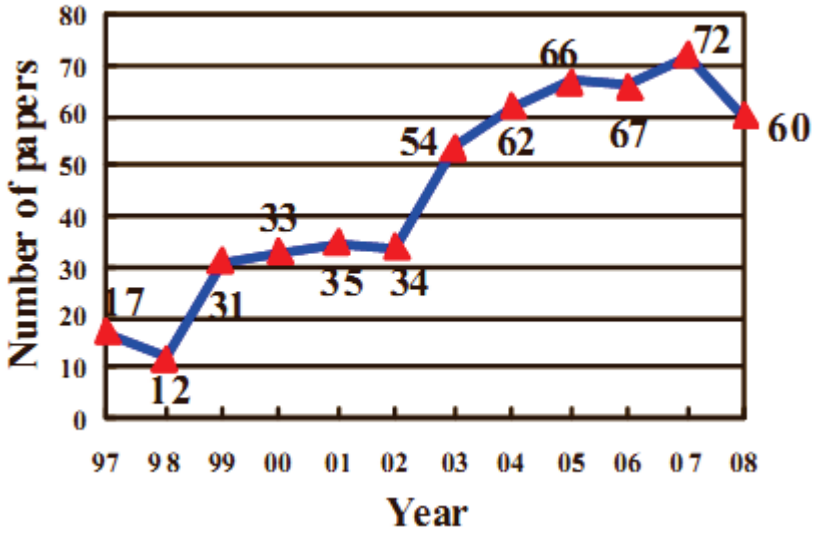
ภาพที่ 2-2 กราฟแสดงคุณภาพของคำตอบ ของแต่ละกลุ่มขั้นตอนวิธี [1]

ตามภาพที่ 2-2 จะได้ว่า คุณภาพของผลเฉลยโดยเฉลี่ย (Solution quality) ที่ได้จากขั้นตอนวิธีแม่นตรงจะมีค่าสูงสุด และ เป็นค่าคงที่ในทุกขนาดของปัญหา และคุณภาพของผลเฉลยที่ดีรองลงมาคือ ขั้นตอนวิธีเชิงวิวัฒน์ และ ขั้นตอนวิธีฮิวริสติค ตามลำดับ โดยคุณภาพของผลเฉลยที่ได้จากขั้นตอนวิธีเชิงวิวัฒน์จะลดลงไปเรื่อยๆแม้ว่าจะใช้ค่าพารามิเตอร์อย่างขนาดประชากร (Population size หรือ popsize) หรือ จำนวนรุ่นค้นหามากที่สุด (Maximum number of generations) โดยไม่เปลี่ยนแปลงตลอดการทดสอบก็ตาม ขั้นตอนวิธีฮิวริสติคเมื่อขนาดของปัญหาเพิ่มขึ้นคุณภาพของผลเฉลยที่ได้ค่อนข้างคงที่ และ ในส่วนของภาพที่ 2-3 ขั้นตอนวิธีแม่นตรงสามารถค้นหาผลเฉลยได้อย่างรวดเร็วเมื่อเทียบกับขั้นตอนวิธีในกลุ่มอื่น อย่างไรก็ตามเมื่อปัญหาการหาค่าเหมาะที่สุดเชิงการจัดมีขนาดใหญ่ จะส่งผลให้ขั้นตอนวิธีดังกล่าวใช้เวลาในการคำนวณที่มากขึ้นจนไม่สามารถนำมาใช้งานได้จริง

ทั้งนี้มีหลายส่วนที่จำเป็นต้องให้ความสนใจในการออกแบบและวิเคราะห์ขั้นตอนเชิงวิวัฒน์สำหรับการหาค่าเหมาะสุดเชิงการจัด ได้แก่ กลไกการถอดรหัสและซ่อมแซม ส่วนถัดมาคือมีการดำเนินการเฉพาะที่หลากหลายในการสร้างแต่ผลเฉลยที่ถอดรหัสได้ ซึ่งถ้าหากใช้วิธีดำเนินการไขว้เปลี่ยนและการกลายพันธุ์แบบเดิมอาจทำให้คำตอบที่ได้ไม่ถูกต้อง และสุดท้ายการพิจารณาเลือกใช้วิธีค้นหาเฉพาะที่ (Local search methods) ปริภูมิการค้นหา (Search space) ของปัญหาการหาค่าเหมาะที่สุดเชิงการจัดมีขนาดค่อนข้างใหญ่ จึงจำเป็นต้องใช้ขั้นตอนวิธีเฉพาะเพื่อให้ผลเฉลยมีคุณภาพดีขึ้นในเวลาที่จำกัด และนอกจากนี้ข้อมูลที่อ้างอิงจาก SCI ระหว่างปี 1997 ถึง 2008 ในส่วนของงานวิจัยทางด้านการแก้ปัญหาการหาค่าเหมาะที่สุดเชิงการจัดโดยขั้นตอนวิธีเชิงวิวัฒน์ ยังได้รับความนิยมเพิ่มขึ้นอย่างต่อเนื่อง ซึ่งแสดงไว้ในภาพที่ 2-4 [1]



ภาพที่ 2-3 กราฟแสดงค่าความซับซ้อนเชิงเวลา ของแต่ละกลุ่มขั้นตอนวิธี [1]



ภาพที่ 2-4 กราฟแสดงจำนวนงานวิจัยทางด้านการหาค่าเหมาะที่สุดเชิงการจัดโดยขั้นตอนวิธีเชิงวิวัฒน์จาก SCI [1]

2.1.4 ปัญหาถุงเป้

ปัญหาถุงเป้ (Knapsack problem) [1] คือ ปัญหาเกี่ยวกับการบรรจุของ n ชิ้นใส่ถุงเป้ภายใต้เงื่อนไขที่ว่าต้องการมูลค่าสูงสุดในขนาดของการบรรจุที่จำกัด โดยสามารถแสดงเงื่อนไขได้ดังสมการ

$$f(x) = \max \sum_{i=1}^{n} p_i x_i \tag{2-3}$$

$$\text{s.t.} \quad \sum_{i=1}^{n} w_i x_i \le W \qquad x_i \in \{0,1\}, i = 1,2,...,n \tag{2-4}$$

ซึ่ง p_i และ w_i แทน มูลค่า (Profit) และน้ำหนัก (Weight) ของสินค้าชิ้นที่ i ตามลำดับ และ W คือขนาดของบรรจุภัณฑ์ และ $x_i = 1$ มีความหมายว่าของชิ้นที่ i จะถูกใส่ลงไปในถุงเป้ และในทางกลับกัน $x_i = 0$ มีความหมายคือของชิ้นที่ i ไม่ได้วางลงไปในถุงเป้นั่นเอง ซึ่งลักษณะปัญหาข้างต้นมีความคล้ายกับหลายปัญหาโลกจริง เช่น ปัญหาการบรรทุกสินค้าในการขนส่งทางอากาศ (Air-cargo loading problems) ปัญหาการเลือกโครงการ (Project selection problems) ปัญหาการจัดสมดุลของสายงานการประกอบ (Assembly line balancing problems) เป็นต้น

การเข้ารหัสฐานสอง (Binary code) [1] ในส่วนของวิธีการเข้ารหัสประชากรสามารถทำได้ 3 แบบดังนี้ โดยอันดับแรก คือ การเข้ารหัสฐานสอง ซึ่งเป็นการออกแบบที่ตรงไปตรงมา โดยอาศัยตำแหน่งในการแทนสินค้าและรูปแบบของยีน (Allele) ถูกแทนในรูปของรหัสฐานสองโดยที่สินค้าในตำแหน่งใดๆหากว่าถูกเลือกจะมีค่าในตำแหน่งนั้นเท่ากับ 1 แต่ถ้าไม่ถูกเลือกให้มีค่าเป็น 0 นั่นเอง โดยตัวอย่างรูปแบบโครโมโซม (Chromosome) หรือ ผลเฉลยที่เป็นไปได้ที่มีขนาดของปัญหาเท่ากับ 7 สามารถแสดงได้ดังภาพที่ 2-5 [1]

1	2	3	4	5	6	7
0	1	0	1	0	1	0

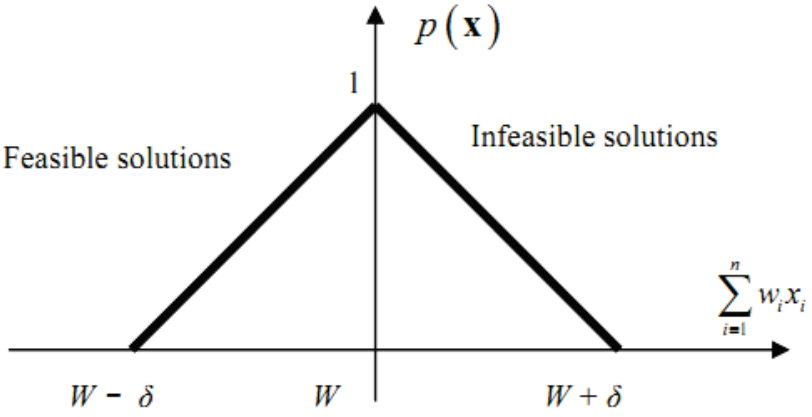
ภาพที่ 2-5 ตัวอย่างรูปแบบโครโมโซมที่เข้ารหัสฐานสองสำหรับปัญหาถุงเป้ [1]

หมายเลขที่กำกับไว้เหนือแต่ละช่องของโครโมโซมแทนจำนวนลำดับของสินค้าและค่าภายในแต่ละช่องของโครโมโซมแทนการเลือกในค่าจำนวนฐานสองเท่ากับหนึ่งหรือไม่เลือกในค่าจำนวนฐานสองเท่ากับศูนย์นั่นเอง อย่างไรก็ตามการเข้ารหัสดังกล่าวอาจทำให้เกิดผลเฉลยที่ไม่ถูกต้อง จึงได้มีการออกแบบวิธีการสำหรับแก้ปัญหาดังกล่าว ซึ่งวิธีแรก ผลเฉลยที่เหมาะที่สุดของปัญหาถุงเป้ต้องมีค่าของน้ำหนักรวมใกล้เคียง หรือ เทียบเท่าขนาดของถุงเป้ ซึ่ง Olsen ได้แนะนำวิธีในการทำโทษผลเฉลยที่มีค่าน้ำหนักรวมเกินหรือต่ำกว่าขนาดของถุงเป้ ซึ่งสามารถแสดงได้ดังสมการ 2-5 และ 2-6 ซึ่งค่าสูงสุดของ δ คือ W และฟังก์ชันการลงโทษ (Penalty function) คือ $p(x)$ นั่นเอง

$$\delta = \min\left\{W, \left|\sum_{i=1}^{n} w_i - W\right|\right\} \tag{2-5}$$

$$p(x) = 1 - \frac{\left|\sum_{i=1}^{n} w_i x_i - W\right|}{\delta} \tag{2-6}$$

นอกจากนี้ฟังก์ชันการลงโทษดังกล่าวสามารถนำเสนอในรูปของกราฟในภาพที่ 2-6 [1] ซึ่งแสดงให้เห็นว่าสามารถจัดการกับผลเฉลยที่ไม่สมเหตุสมผล และ ผลเฉลยที่สมเหตุสมผลที่ไม่ได้อยู่ในขอบเขตของบริเวณที่ต้องการอีกด้วย ในส่วนของฟังก์ชันความเหมาะสม (Fitness function) สามารถแสดงได้ดังสมการ 2-7 ซึ่งเป็นผลคูณระหว่างฟังก์ชันวัตถุประสงค์ (Objective function) กับฟังก์ชันการลงโทษนั่นเอง



ภาพที่ 2-6 กราฟของฟังก์ชันการลงโทษสำหรับปัญหาถุงเป้

$$fitness(x) = f(x)p(x) \tag{2-7}$$

ในส่วนของการซ่อมแซมถือเป็นอีกแนวทางหนึ่งในการจัดการกับทุกผลเฉลยที่ไม่ถูกต้อง ซึ่งใช้แนวคิดทั่วไปโดยใช้อัตราส่วนระหว่างมูลค่าต่อน้ำหนัก นั่นเอง ซึ่งสามารถจำลองเหตุการณ์ตัวอย่างได้ดังนี้ โดยให้ $W = 90$ และ $n = 7$ รวมถึงค่าพารามิเตอร์ที่สำคัญต่างๆดังแสดงในตารางที่ 2-1 [1]

เมื่อใช้ตารางดังกล่าวสำหรับอ้างอิงกับตัวอย่างผลเฉลยในภาพที่ 2-5 จะได้ว่าผลเฉลยดังกล่าวไม่ถูกต้อง เพราะว่า ผลรวมของน้ำหนักของสินค้าที่ 2, 4 และ 6 มีค่าเท่ากับ 100 ซึ่งมีค่ามากกว่าขนาดบรรจุภัณฑ์ที่ยอมรับได้ ซึ่งในที่นี้คือ $W = 90$ และจากอัตราส่วนระหว่างมูลค่าต่อ

น้ำหนักที่กำหนดไว้ในตารางที่ 2-1 จะได้ว่าสินค้าหมายเลข 6 ซึ่งมีค่าอัตราส่วนระหว่างมูลค่ากับน้ำหนักต่ำสุดเมื่อเทียบกับสินค้าหมายเลข 2 และ 4 จะถูกคัดทิ้งจากถุงเป้นั่นเอง ซึ่งจะทำให้ผลเฉลยจากเดิมจะถูกแก้ไขเป็น (0101000) และทั้งนี้การซ่อมแซมที่เกิดขึ้นอาจทำให้พื้นที่ในการค้นหาผลเฉลยน้อยลง ซึ่งเป็นส่วนที่ควรหลีกเลี่ยงในการออกแบบเช่นกัน

ตารางที่ 2-1 ตัวอย่างการซ่อมแซมในปัญหาถุงเป้ [1]

Item	1	2	3	4	5	6	7
p_i	40	60	10	10	3	20	60
w_i	40	50	30	10	10	40	30
p_i / w_i	1	1.2	0.33	1	0.33	0.5	2

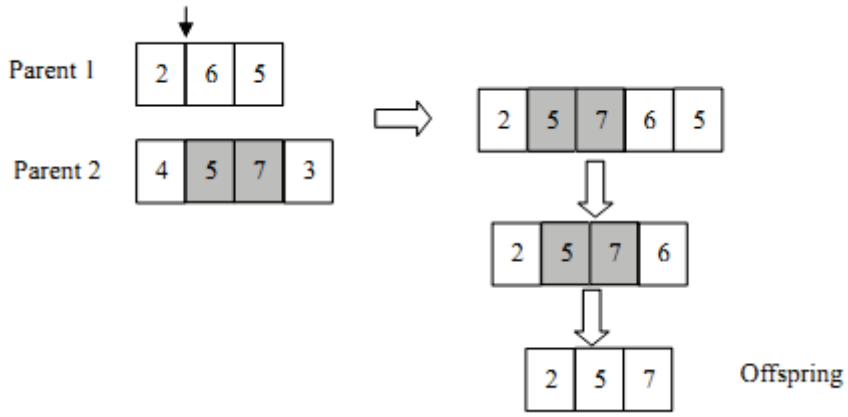
การเข้ารหัสเชิงความยาวตัวแปร (Variable-length code) [22] เสนอโดย Hinterding ซึ่งความยาวของโครโมโซมสามารถเปลี่ยนระหว่างกระบวนการหาผลเฉลยได้ โดยไม่สนใจตำแหน่งก่อนหลังของยีน (gene) และรูปแบบของยีนแสดงจำนวนลำดับของสินค้า ตัวอย่างเช่น (1, 6, 5) มีความหมายว่ามีสินค้าหมายเลข 1, 5, และ 6 อยู่ในถุงเป้นั่นเอง

วิธีการเข้ารหัสดังกล่าวมีลักษณะพิเศษที่ทำให้สามารถคงรูปแบบผลเฉลยที่สมเหตุสมผลได้ โดยในการสร้างแต่ละผลเฉลย อันดับแรกเริ่มจากกำหนดค่าเริ่มต้น โดยมีการอาศัยการสร้างกลุ่มของเลขจำนวนเต็มแบบสุ่มจำนวน n ตัวตามขนาดของปัญหา และเริ่มทำการตรวจสอบจากส่วนแรกสุดที่ถูกเลือกในกลุ่มดังกล่าวไม่ว่าการเลือกดังกล่าวจะขัดแย้งกับข้อจำกัดหรือไม่ก็ตาม และถ้าการเพิ่มส่วนที่พิจารณาล่าสุดลงในผลเฉลยยังคงไม่ขัดแย้งกับขนาดที่ยอมรับได้ของถุงเป้ ส่วนที่ถูกพิจารณาดังกล่าวก็จะถูกเพิ่มลงไปนั่นเอง แต่ถ้าตรวจสอบแล้วขัดแย้งกัน จะถูกยกเลิกพร้อมกับมีการตรวจสอบส่วนถัดไปจนกว่าถุงเป้จะเต็มหรือได้ทำการสำรวจกลุ่มของเลขจำนวนเต็มจนครบทั้งหมด และทั้งนี้แต่ละกลุ่มเลขจำนวนเต็มแบบสุ่มสามารถนำมาใช้สร้างผลเฉลยได้เพียงครั้งเดียวเท่านั้น กระบวนการทำงานดังกล่าวจะวนซ้ำจนกว่าจำนวนของผลเฉลยจะมีค่าเท่ากับจำนวนประชากรที่กำหนดไว้

การไขว้เปลี่ยน ประกอบด้วยหลายขั้นตอนที่สำคัญในการรักษาลักษณะที่ถูกต้องของลูกหลาน (Offspring) ซึ่งสามารถแสดงการทำงานได้ตามภาพที่ 2-7 [1] โดยขนาดที่สามารถบรรจุได้ของถุงเป้มีค่าเท่ากับ $W = 90$ และ $n = 7$ และค่าพารามิเตอร์ในส่วนที่เหลืออ้างอิงตามตารางที่ 2-1 ซึ่งขั้นตอนแรกจะเป็นการเลือกตำแหน่งช่องว่างระหว่างยีนในโครโมโซมของ parent 1 และเลือกกลุ่มย่อยของยีนที่ติดกันในโครโมโซมของ parent 2 ขั้นตอนถัดไปคือ ให้ใส่กลุ่มย่อยของยีนที่

เลือกไว้ใน parent 2 ตรงตำแหน่งช่องว่างที่เลือกไว้ของ parent 1 พร้อมกับลบยีนที่ซ้ำซ้อนใน parent 1 ออกไป และขั้นตอนสุดท้ายให้ใช้อัตราส่วนระหว่างมูลค่าต่อน้ำหนักในการซ่อมแซมหากว่าลูกหลานที่ได้มีคุณสมบัติไม่ถูกต้อง

เมื่อสังเกตตัวอย่างตามภาพที่ 2-7 [1] พบว่าหลังจากการไขว้เปลี่ยนและตัดส่วนที่ซ้ำซ้อนออก ผลเฉลยล่าสุดที่ได้จะมีลักษณะดังนี้ คือ (2, 5, 7, 6) ซึ่งไม่ถูกต้อง และ หากพิจารณาตามอัตราส่วนของมูลค่าต่อน้ำหนักในตารางที่ 2-1 พบว่ายีนหมายเลข 5 คือยีนที่ถูกพิจารณาเป็นอันดับแรกในการคัดออกจากผลเฉลยดังกล่าว อย่างไรก็ตาม แม้ว่าจะคัดยีนหมายเลข 5 ออกไป ผลเฉลยใหม่ที่ประกอบด้วย (2, 7, 6) ยังคงไม่ถูกต้อง ดังนั้น จึงมีการเพิ่มกระบวนการยินยอมให้ยีนหมายเลข 6 ถูกคัดออกบ้าง และได้ผลเฉลยใหม่ที่ประกอบด้วย (2, 5, 7) ซึ่งเป็นผลเฉลยที่ถูกต้อง



ภาพที่ 2-7 ตัวอย่างแสดงการไขว้เปลี่ยนสำหรับการเข้ารหัสเชิงความยาวตัวแปร

ในส่วนของการกลายพันธุ์มีกระบวนการที่ง่าย เริ่มจากการสุ่มเลือกยีนหนึ่งภายในโครโมโซมเพื่อลบออก และสุ่มเพิ่มยีนใหม่ที่ไม่ซ้ำกันลงในโครโมโซม และสุดท้ายให้นำโครโมโซมดังกล่าวมาตรวจสอบว่ามีลักษณะที่ถูกต้องหรือไม่ โดยพิจารณาจากอัตราส่วนระหว่างมูลค่าต่อน้ำหนัก

<u>การเข้ารหัสเชิงการเรียงสับเปลี่ยน (Permutation code)</u> เป็นการเข้ารหัสที่น่าสนใจสำหรับปัญหาถุงเป้อีกรูปแบบหนึ่ง ซึ่งการเรียงสับเปลี่ยนใดๆของแต่ละชุดจำนวน n สามารถอธิบายลำดับของการหยิบสินค้าได้ โดยสมมติให้ $W = 100$, $n = 7$ และ ค่าอื่นๆอ้างอิงตามตารางที่ 2-1 และ พิจารณาผลเฉลยหนึ่ง คือ (1, 6, 4, 7, 3, 2, 5) ซึ่งสามารถนำมาเข้ากระบวนการถอดรหัสตามนี้ เมื่อเลือกยีนแรก หรือ ยีนหมายเลข 1 เพื่อบรรจุลงในถุงเป้ พร้อมกับพิจารณายีนอันดับที่สอง หรือ ยีนหมายเลข 6 พบว่าสามารถบรรจุเพิ่มลงไปถัดจากยีนหมายเลข 1 ได้เพราะเมื่อหา

ผลรวมของขนาดยีนทั้งสองพบว่า ค่าของผลรวมดังกล่าวยังคงมีค่าน้อยกว่าปริมาณสูงสุดที่รองรับได้ของถุงเป้ ซึ่งก็คือ $40+40=80<100=W$ และเมื่อบรรจุยีนหมายเลข 4 เพิ่มเติมลงในถุงเป้ ผลรวมที่ได้ยังคงถูกต้องตามเงื่อนไข $40+40+10=90<100=W$ และเมื่อนำยีนหมายเลข 7 มาบรรจุลงถุงเป้ จะพบว่าไม่สามารถบรรจุลงถุงเป้ได้ เพราะว่าน้ำหนักเกิน อย่างไรก็ตาม เมื่อไม่สามารถบรรจุยีนหมายเลข 7 ลงถุงเป้ได้ ระบบการคัดเลือกจะข้ามไปยีนถัดไปคือยีนหมายเลข 3, 2, และ 5 ตามลำดับจนกว่าจะพิจารณาครบทั้งหมด ซึ่งยีนที่ได้รับการคัดเลือกเป็นส่วนถัดไป คือ ยีนหมายเลข 5 ซึ่งสามารถแสดงค่าน้ำหนักรวมดังนี้คือ $40+40+10+10=100=W$ ดังนั้น ผลลัพธ์จากการถอดรหัสของผลเฉลย (1, 6, 4, 7, 3, 2, 5) คือ 1, 6, 4, และ 5 ซึ่งในกรณีดังกล่าวทุกๆ n การเรียงสับเปลี่ยนสามารถถอดรหัสในรูปของผลเฉลยที่ถูกต้อง

สิ่งที่ควรคำนึงถึงอีกอย่างหนึ่งคือ ปัญหาการไขว้เปลี่ยนและการกลายพันธุ์แบบดั้งเดิม ซึ่งจะทำลายรูปแบบของการเรียงสับเปลี่ยนที่มีอยู่ ดังนั้นการดำเนินการที่มีรูปแบบเฉพาะเจาะจงจึงเป็นอีกทางเลือกหลักในการเข้ารหัสเชิงการเรียงสับเปลี่ยน และอีกอย่างหนึ่งที่ควรพิจารณา คือ บางกลุ่มของผลเฉลยที่เป็นไปได้ภายในประชากร มีผลลัพธ์ที่ได้จากการถอดรหัสที่เท่ากัน ตัวอย่างเช่น เมื่อพิจารณาแต่ละผลเฉลย (1, 6, 4, 7, 3, 2, 5), (1, 4, 6, 7, 3, 2, 5), (6, 4, 1, 7, 3, 2, 5), …, (6, 4, 1, 7, 3, 5, 2) ในส่วนของการถอดรหัสจะพบว่าผลลัพธ์ที่ได้มีค่าตรงกันในทุกๆการถอดรหัสเท่ากับ 1, 6, 4, และ 5 รวมถึงเป็นผลเฉลยที่ถูกต้องอีกด้วย ซึ่งมีความหมายว่าความสัมพันธ์ระหว่างโดเมนการดำเนินการ (operation domain) และ โดเมนการกำหนด (definition domain) เป็นไปในลักษณะหลายต่อหนึ่ง (many-to-one) โดยความสัมพันธ์ดังกล่าวอาจทำให้สูญเสียความสามารถในการค้นหาของขั้นตอนวิธีเชิงวิวัฒน์ไปโดยเปล่าประโยชน์ และเป็นสาเหตุสำคัญที่ทำให้ประสิทธิภาพลดลง

ตารางที่ 2-2 ตัวอย่างการเปรียบเทียบขนาดของปริภูมิค้นหาระหว่างการเข้ารหัสฐานสองและการเรียงสับเปลี่ยน [1]

n	1	2	3	4	5	6	7	8	9	10
2^n	2	4	8	16	32	64	128	256	512	1024
$n!$	1	2	6	24	120	720	5040	40320	362880	3628800

และสุดท้าย เมื่อเปรียบเทียบปริภูมิการค้นหาระหว่างการเข้ารหัสแบบฐานสองและแบบการเรียงสับเปลี่ยน สินค้า n ชิ้น พบว่าขนาดของปริภูมิการค้นหามีค่าเท่ากับ 2^n และในส่วนของการเข้ารหัสเชิงการเรียงสับเปลี่ยนมีค่าเท่ากับ $n!$ และเมื่อเปรียบเทียบในเชิงตัวเลขของขนาด

ปัญหาภายในช่วง 1 ถึง 10 ซึ่งเป็นขนาดปัญหาเล็ก พบว่าขนาดของปริภูมิการค้นหาของการเข้ารหัสเชิงการเรียงสับเปลี่ยนมีขนาดที่ใหญ่มากเมื่อเทียบกับขนาดของการเข้ารหัสฐานสอง ดังตารางที่ 2-2 [1]

2.1.5 ปัญหาการเดินทางของพนักงานขาย

ปัญหาการเดินทางของพนักงานขาย [1] เป็นปัญหาการหาค่าเหมาะที่สุดเชิงการจัดที่มีชื่อเสียง และ ถูกนำมาใช้ในการทดสอบประสิทธิภาพกับวิธีการหาค่าเหมาะที่สุดจำนวนมาก โดยผลเฉลยที่ต้องการคือ เส้นทางสั้นสุดที่พนักงานขายสามารถเดินผ่านทุกเมือง ทั้งนี้ปัญหาดังกล่าวเป็นที่นิยมในการใช้ทดสอบประสิทธิภาพของขั้นตอนวิธีในกลุ่มขั้นตอนวิธีเชิงวิวัฒน์อีกด้วย และเมื่อนำขั้นตอนวิธเชิงพันธุกรรมมาใช้แก้ปัญหาดังกล่าว วิธีการไขว้เปลี่ยนและการกลายพันธุ์ตามปกติที่จะทำให้เกิดประชากรที่ไม่ถูกต้องขึ้น ดังนั้น จึงมีการเสนอวิธีการไขว้เปลี่ยนในหลายรูปแบบด้วยกัน เช่น การไขว้เปลี่ยนแบบจัดคู่เป็นส่วน (Partially Mapped Crossover) เป็นต้น

ในปัจจุบันมีการเสนอแนวคิดที่หลากหลายสำหรับแก้ปัญหาขนาดของปริภูมิการค้นหาของการเดินทางของพนักงานขายแบบสมมาตรของ n เมือง จะมีจำนวนผลเฉลยที่เป็นไปได้ทั้งหมดเท่ากับ $\frac{(n-1)!}{2}$ การเข้ารหัสประชากร มีการออกแบบไว้หลากหลายแนวคิดเช่นกัน [1] ได้แก่ การเข้ารหัสเชิงเส้น (Edge code) การเข้ารหัสฐานสอง (Binary code) การเข้ารหัสเชิงวิถี (Path code) การเข้ารหัสเชิงประชิด (Adjacent code) [23] การเข้ารหัสเชิงอันดับ (Ordinal code) เป็นต้น

<u>การเข้ารหัสเชิงเส้น</u> เป็นการเข้ารหัสโครโมโซมซึ่งแต่ละยีนแทนเส้นเชื่อมที่เชื่อมระหว่างสองเมือง และ เมื่อมี n เมืองที่มีลักษณะของกราฟบริบูรณ์ (Complete graph) จะได้ว่าจำนวนเส้นเชื่อมที่เป็นไปได้ทั้งหมดมีจำนวนเท่ากับ $\frac{n(n-1)}{2}$ เส้น ซึ่งในการสร้างแต่ละโครโมโซมจะเลือกเส้นต่างๆที่เป็นไปได้จากกราฟ โดยเส้นทั้งหมดจะประกอบกันในรูปของกราฟซึ่งลากผ่านแต่ละเมืองเพียงครั้งเดียว และเส้นสุดท้ายจากเมืองสุดท้ายมายังเมืองเริ่มต้นอีกครั้ง เรียกว่า วัฏจักรแฮมิลตัน (Hamiltonian cycle) และตัวอย่างโครโมโซมของ 8 เมืองในปัญหาการเดินทางของพนักงานขายสามารถแสดงได้ดังนี้ (14, 28, 18, 23, 27, 7, 6, 16) อย่างไรก็ตามข้อเสียเปรียบหลักสำหรับวิธีนี้คือต้องแน่ใจว่าเส้นทั้งหมดที่ได้รับเลือกมีลักษณะของวัฏจักรแฮมิลตันจริง และต้องหลีกเลี่ยงวิธีการดำเนินการจำนวนมากที่จะทำให้สูญเสียคุณสมบัติวัฏจักรแฮมิลตัน อีกด้วย ซึ่งเป็นประเด็นที่ค่อนข้างยากจึงทำให้วิธีนี้ไม่ได้รับความนิยมเท่าที่ควร

การเข้ารหัสฐานสอง เป็นวิธีที่แทนโครโมโซมโดยเลขฐานสองทั้งหมด ซึ่งแสดงถึงหมายเลขของแต่ละจุดหรือเมือง โดยสามารถแสดงตัวอย่างการเข้ารหัสสำหรับปัญหาการเดินทางของพนักงานขาย ที่มีขนาด 6 เมือง คือ (000 010 001 100 011 101) ซึ่งมีความหมายเดียวกันกับ 1-3-2-5-4-6-1 นั่นเอง ซึ่งโครโมโซมที่ได้จากการเข้ารหัสดังกล่าวยังคงสามารถถูกทำให้สูญเสียคุณสมบัติได้โดยการดำเนินการแบบเดิม และทั้งนี้โดยทั่วไปแล้ว เมื่ออ้างอิงจากตัวอย่างข้างต้นจะได้ว่าแต่ละยีนความน่าจะเป็นเท่ากับ n ดังนั้น สำหรับปัญหาที่มีขนาดเมืองเท่ากับ n จะได้ว่าโดเมนการดำเนินการของวิธีนี้จะมีขนาดเท่ากับ n^n

การเข้ารหัสเชิงวิถี เป็นวิธีการเข้ารหัสผลเฉลยที่ค่อนข้างเหมาะสมสำหรับปัญหาการเดินทางของพนักงานขาย ซึ่งผลเฉลยที่ได้แทนแต่ละเมืองที่พนักงานเดินทางผ่านเป็นลำดับตามวิถีแฮมิลตัน (Hamiltonian path) โดยตัวอย่างผลเฉลยสำหรับปัญหาดังกล่าวที่มีจำนวนเมืองเท่ากับ n สามารถแสดงในรูปของการเรียงสับเปลี่ยนเลขจำนวนเต็ม เช่น (2-3-6-1-4-5) ซึ่งมีความหมายแบบเดียวกับ 2-3-6-1-4-5-2 ได้มีการเสนอวิธีการเฉพาะสำหรับสร้างลูกหลานที่ถูกต้องรวมถึงมีการปรับปรุงวิธีการกลายพันธุ์เช่นกัน และขนาดของโดเมนการดำเนินการของการเข้ารหัสเชิงวิถีที่มีขนาดของปัญหาเท่ากับ n จะมีค่าเท่ากับ $n!$

ปัญหาอย่างหนึ่งที่ควรทราบสำหรับการเข้ารหัสเชิงวิถี คือ ปัญหาความซ้ำซ้อนของผลเฉลยที่เป็นไปได้ในประชากร ตัวอย่างเช่น (2-3-6-1-4-5), (3-6-1-4-5-2), (6-1-4-5-2-3), (1-4-5-2-3-6), (4-5-2-3-6-1), (5-2-3-6-1-4), (5-4-1-6-3-2), (4-1-6-3-2-5), (1-6-3-2-5-4), (6-3-2-5-4-1), (3-2-5-4-1-6), และ (2-5-4-1-6-3) ล้วนแล้วแต่เป็นผลเฉลยที่มีความหมายเดียวกันทั้งสิ้น ซึ่งความซ้ำซ้อนดังกล่าวส่งผลต่อประสิทธิภาพในการค้นหาที่ลดต่ำลง

การเข้ารหัสเชิงประชิด เป็นการเข้ารหัสที่มีลักษณะคล้ายกับการเข้ารหัสเชิงวิถี ซึ่งมีนิยามของการทำงาน คือ ค่า j ตรงตำแหน่งที่ i มีความหมายว่า เมืองที่ j จะเป็นเมืองถัดไปหากว่าพนักงานขายเดินทางมาถึงเมืองที่ i เรียบร้อยแล้ว ตัวอย่างเช่น ให้ขนาดของปัญหามีค่าเท่ากับ 6 และผลเฉลยหนึ่งที่เข้ารหัสแบบการเข้ารหัสเชิงประชิดมีลักษณะดังนี้ คือ (2-3-6-1-4-5) ซึ่งสามารถถอดรหัสออกมาได้ว่าตำแหน่งที่ 1 เก็บเมืองหมายเลข 2 ไว้ ซึ่งก็คือ เมืองที่ 1 ไปเมืองที่สอง หรือ 1-2 นั่นเอง ต่อมาให้นำหมายเลข 2 ที่ได้ไปค้นหาต่อจะได้ว่าตำแหน่งที่ 2 เก็บเมืองหมายเลข 3 ไว้ และนำมาเชื่อมติดกับข้อมูลก่อนหน้าจะได้เป็น 1-2-3 ถัดมานำหมายเลข 3 ที่ได้รับล่าสุดไปเทียบข้อมูลจากผลเฉลยดังกล่าวจะได้ว่า ตำแหน่งที่ 3 เก็บค่า 6 ไว้ และนำมาปรับปรุงข้อมูลอีกครั้ง ซึ่งมีค่าเท่ากับ 1-2-3-6 และสุดท้ายให้ทำการเทียบค่ากับผลเฉลยตัวอย่าง จนกระทั่งครบทั้งหมด ผลที่ได้จากการถอดรหัสจะมีลักษณะดังนี้ คือ 1-2-3-6-5-4-1 นั่นเอง ทั้งนี้โดเมนการ

ดำเนินการจึงมีขนาดเท่ากับ $n!$ และปัญหาอย่างหนึ่งของการเข้ารหัสแบบนี้ คือยังคงมีบางผลเฉลยที่ไม่สามารถถอดรหัสออกมาเป็นผลลัพธ์ที่ถูกต้องได้ เช่น (2-3-1-6-4-5) เมื่อนำมาถอดรหัสแล้วจะได้ผลลัพธ์อยู่ 2 ส่วนคือ 1-2-3-1 และ 4-6-5-4 นั่นเอง ดังนั้น การเข้ารหัสในลักษณะนี้ยังคงมีความจำเป็นต่อการใช้ฟังก์ชันการซ่อมแซมเช่นเดิม

ตารางที่ 2-3 ตัวอย่างการถอดรหัสโครโมโซม (3, 2, 4, 2, 2, 1) แบบการเข้ารหัสเชิงประชิตในปัญหา TSP [1]

Sequence	Current order list	Current working gene	Current tour
1	1-2-3-4-5-6	(**3**, 2, 4, 2, 2, 1)	3
2	1-2-4-5-6	(3, **2**, 4, 2, 2, 1)	3-2
3	1-4-5-6	(3, 2, **4**, 2, 2, 1)	3-2-6
4	1-4-5	(3, 2, 4, **2**, 2, 1)	3-2-6-4
5	1-5	(3, 2, 4, 2, **2**, 1)	3-2-6-4-5
6	1	(3, 2, 4, 2, 2, **1**)	3-2-6-4-5-1-3

การเข้ารหัสเชิงอันดับ เป็นการเข้ารหัสที่ใช้การเรียงสับเปลี่ยน n ตัวในการแทนแต่ละผลเฉลย และ ใช้รายการลำดับมาตรฐานของเมืองสำหรับ n เมือง เช่น 1-2-3-4-5-6 สำหรับปัญหาขนาด 6 เมือง และ โครโมโซมที่แทนผลเฉลยสำหรับการเข้ารหัสเชิงอันดับ คือ (3, 2, 4, 2, 2, 1) ซึ่งสามารถแสดงกระบวนการถอดรหัสผลเฉลยได้ดังตารางที่ 2-3 [1]

ในส่วนของการออกแบบการดำเนินการสำหรับการเข้ารหัสเชิงการเรียงสับเปลี่ยนของปัญหาการเดินทางของพนักงานขาย ประกอบด้วยส่วนสำคัญ 2 ส่วน คือ การไขว้เปลี่ยน และ การกลายพันธุ์ ซึ่งจนถึงปัจจุบันได้มีการออกแบบวิธีการดำเนินการไว้หลายแบบจากหลากหลายงานวิจัยด้วยกัน เพื่อต้องการให้ขั้นตอนวิธีเชิงพันธุกรรมสามารถแก้ปัญหาดังกล่าวได้อย่างมีประสิทธิภาพ

การกลายพันธุ์เชิงแลกเปลี่ยน (Exchange Mutation หรือ EM) [24] เป็นวิธีการกลายพันธุ์ที่เสนอโดย Banzhaf ซึ่งบางครั้งถูกเรียกว่า การกลายพันธุ์เชิงสับค่า (Swap Mutation) โดยการทำงานของวิธีนี้เริ่มจากสุ่มเลือกตำแหน่งสองตำแหน่งของยีนภายในโครโมโซมขึ้นมา พร้อมกับทำการสลับตำแหน่งของค่าที่อยู่ประจำตำแหน่งนั้น ตัวอย่างเช่น มีผลเฉลยหนึ่ง คือ (1-2-3-4-5-6-7-8-9) สมมติว่าผลจากการสุ่มตำแหน่งคือ ตำแหน่งที่ 3 และ 6 ที่เก็บค่า 3 กับ 6 ตามลำดับ ให้ทำ

การสลับค่าในตำแหน่งทั้งสองทันที และ ผลลัพธ์หลังจากการสลับค่าคือ (1-2-6-4-5-3-7-8-9) ซึ่งสามารถแสดงได้ดังภาพที่ 2-8 [1]

ภาพที่ 2-8 ตัวอย่างการทำงานของการกลายพันธุ์เชิงแลกเปลี่ยน

การกลายพันธุ์แบบแทรก (Insertion Mutation หรือ ISM) [25] เป็นการดำเนินการกลายพันธุ์ที่เสนอโดย Michalewicz ซึ่งกระบวนการทำงานของวิธีนี้เริ่มจากสุ่มเลือกยีนขึ้นมาเพียงยีนเดียว และ สุ่มเลือกตำแหน่งที่ต้องการแทรกยีนที่ถูกเลือกลงไป ตัวอย่างเช่น มีผลเฉลยหนึ่ง คือ (1-2-3-4-5-6-7-8-9) และ ผลของการสุ่มค่า คือ ตำแหน่งยีนที่เก็บเลข 6 และตำแหน่งระหว่าง 2 และ 3 จะได้ว่า ผลลัพธ์หลังจากวิธีดังกล่าวมีลักษณะดังนี้ (1-2-6-3-4-5-7-8-9) และสามารถแสดงได้ดังภาพที่ 2-9 [1]

ภาพที่ 2-9 ตัวอย่างการทำงานของการกลายพันธุ์แบบแทรก

การกลายพันธุ์แบบเคลื่อนเป็นกลุ่ม (Displacement Mutation หรือ DM) [25] เป็นการดำเนินการที่เสนอโดย Michalewicz ซึ่งกระบวนการทำงานมีลักษณะคล้ายกับการกลายพันธุ์แบบแทรก โดยส่วนที่แตกต่างคือจากเดิมเลือกยีนเพียงยีนเดียวเปลี่ยนเป็นสุ่มเลือกยีนเป็นชุดของยีนที่ติดกัน นั่นเอง ตัวอย่างเช่น มีโครโมโซม (1-2-3-4-5-6-7-8-9) และสมมติว่าผลจากการสุ่ม คือ กลุ่มของยีน 4-5-6 และ ตำแหน่งที่แทรกอยู่ระหว่าง 8 และ 9 จะได้ผลลัพธ์ของการทำงาน คือ (1-2-3-7-8-4-5-6-9) ซึ่งสามารถแสดงได้ดังภาพที่ 2-10 [1]

ภาพที่ 2-10 ตัวอย่างการทำงานของการกลายพันธุ์แบบเคลื่อนเป็นกลุ่ม

การกลายพันธุ์เชิงผกผันอย่างง่าย (Simple Inversion Mutation หรือ SIM) [23] เป็นวิธีการกลายพันธุ์ที่เสนอโดย Grefenstette et al. ซึ่งกระบวนการทำงานของวิธีนี้เริ่มจากการสุ่มเลือกตำแหน่งของยีนที่ติดกัน และทำการกลับค่าทั้งหมด ตัวอย่างเช่น มีโครโมโซม (1-2-3-4-5-6-

7-8-9) และ ผลของการสุ่ม คือ กลุ่มของยีน 4-5-6 ดังนั้น ผลลัพธ์ที่ได้จากกระบวนการดังกล่าว คือ (1-2-3-6-5-4-7-8-9) ซึ่งสามารถแสดงได้ดังภาพที่ 2-11 [1]

ภาพที่ 2-11 ตัวอย่างการทำงานของการกลายพันธุ์เชิงผกผันอย่างง่าย

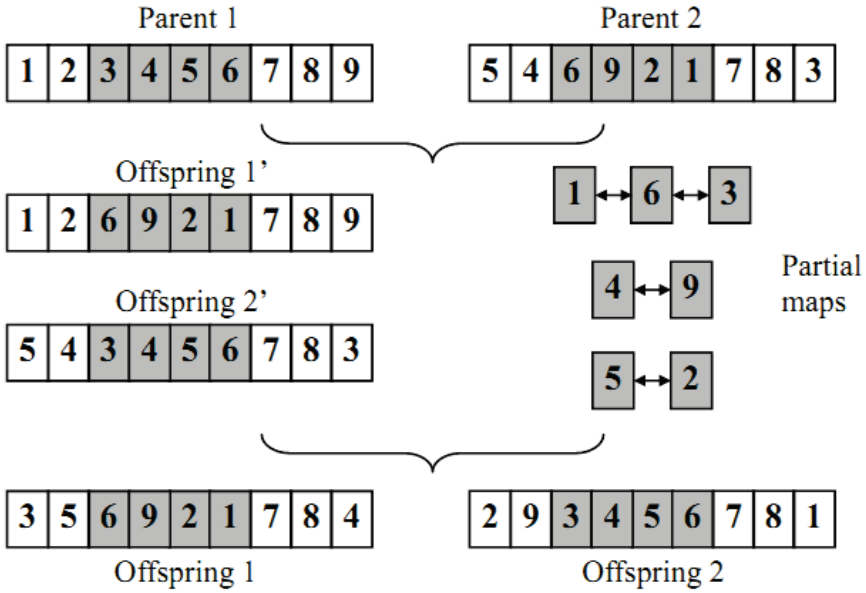
การกลายพันธุ์เชิงผกผัน (Inversion Mutation หรือ IM) [26] เป็นวิธีการกลายพันธุ์ที่นำคุณสมบัติบางอย่างจากทั้งการกลายพันธุ์เชิงผกผันอย่างง่ายและการกลายพันธุ์แบบเคลื่อนเป็นกลุ่ม ซึ่งกระบวนการทำงานของวิธีนี้ เริ่มจากการสุ่มเลือกยีนที่ติดกัน และสุ่มตำแหน่งที่ต้องการแทรก ตัวอย่างเช่น มีโครโมโซม (1-2-3-4-5-6-7-8-9) และ ผลจากการสุ่ม คือ ชุดของยีน 4-5-6 และตำแหน่งแทรกที่อยู่ระหว่าง 8 และ 9 ดังนั้น ผลลัพธ์ที่ได้จากวิธีนี้ คือ (1-2-3-7-8-6-5-4-9) ซึ่งสามารถแสดงได้ดังภาพที่ 2-12 [1]

ภาพที่ 2-12 ตัวอย่างการทำงานของการกลายพันธุ์เชิงผกผัน

การดำเนินการไขว้เปลี่ยน (Crossover Operators) มีรูปแบบการดำเนินการพื้นฐานที่สำคัญอยู่สามแบบด้วยกัน คือ การไขว้เปลี่ยนจุดเดียว (Single-point crossover) การไขว้เปลี่ยนหลายจุด (Multiple crossover) และการไขว้เปลี่ยนแบบเอกรูป (Uniform crossover) ซึ่งแต่ละวิธีการไขว้เปลี่ยนดังกล่าวสามารถทำลายรูปแบบการเรียงสับเปลี่ยนได้ทั้งหมด โดยทำให้เกิดผลเฉลยที่ไม่ถูกต้องขึ้น ดังนั้นจึงได้มีการออกแบบวิธีการที่เฉพาะ เพื่อแก้ปัญหาดังกล่าว โดยแต่ละการไขว้เปลี่ยนที่ออกแบบใหม่มีหลายแบบด้วยกันตามแนวคิดของแต่ละงานวิจัยนั่นเอง

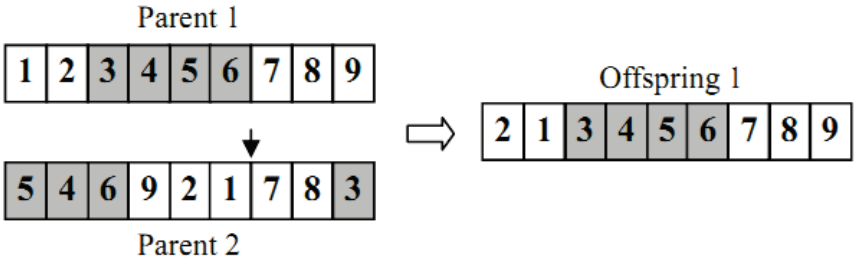
การไขว้เปลี่ยนแบบจัดคู่เป็นส่วน เป็นวิธีการไขว้เปลี่ยนแบบเฉพาะวิธีหนึ่งที่เสนอโดย Goldberg และ Lingle [5] โดยกระบวนการทำงานต่างๆสามารถอธิบายได้โดยเริ่มจากการสุ่มเลือกตำแหน่งภายในโครโมโซมสำหรับแลกเปลี่ยนกลุ่มยีนที่ติดกันและตรงกันระหว่างโครโมโซม โดยในตัวอย่างตามภาพที่ 2-13 [1] โดยสองตำแหน่งที่สุ่มได้อยู่ระหว่าง 2 และ 3 และระหว่าง 6 ถึง 7 เป็นต้น ซึ่งกลุ่มของยีนที่ต้องการสลับจะอยู่ระหว่างสองตำแหน่งดังกล่าวนั่นเอง ขั้นตอนถัดมาให้แลกเปลี่ยนกลุ่มยีนระหว่างกันตามช่วงที่กำหนดไว้ ซึ่งผลลัพธ์ที่ได้คือโครโมโซมลูกหลานจำนวน 2 โครโมโซม แต่ให้บันทึกผลลัพธ์ในขั้นตอนนี้ไว้ในลักษณะผลลัพธ์ชั่วคราวก่อน เรียกว่าโครโมโซมลูกหลานที่ 1’ (Offspring 1’) และ ลูกหลานที่ 2’ (Offspring 2’) เนื่องจากในขั้นตอนนี้มี

ความเป็นไปได้ที่จะทำให้เกิดโครโมโซมที่ไม่ถูกต้อง ซึ่งจำเป็นต้องผ่านฟังก์ชันการซ่อมแซม (Repair function) โดยขั้นตอนของการซ่อมแซม เริ่มจากการกำหนดการจับคู่บางส่วนระหว่างกลุ่มยีนที่ถูกเลือกจากโครโมโซมของพ่อแม่ทั้งสองส่วน ซึ่งเป็นการจับคู่ในตำแหน่งที่ตรงกันของกลุ่มยีนดังกล่าว ซึ่งถ้าทำการจับคู่ตามตัวอย่างที่ให้ไว้จะได้ผลลัพธ์ของการจับคู่บางส่วน ได้แก่ $1 \leftrightarrow 6 \leftrightarrow 3, 4 \leftrightarrow 9$ และ $5 \leftrightarrow 2$ นั่นเอง พร้อมกับนำข้อมูลดังกล่าวมาใช้ในการซ่อมแซมลูกหลานที่สร้างไว้ก่อนหน้า โดยถ้าพิจารณาตามตัวอย่างจะได้ว่า ลูกหลานที่ 1' มียีนที่ซ้ำซ้อนกัน คือ ยีนที่เก็บหมายเลข 1 ไว้ ให้แก้ไขเป็น 6 ตามที่จับคู่ไว้ แต่ เมื่อนำ 6 มาวางแทน ผลลัพธ์ยังคงไม่สมเหตุสมผลอยู่ ให้ 6 จับคู่กับ 3 เป็นต้น



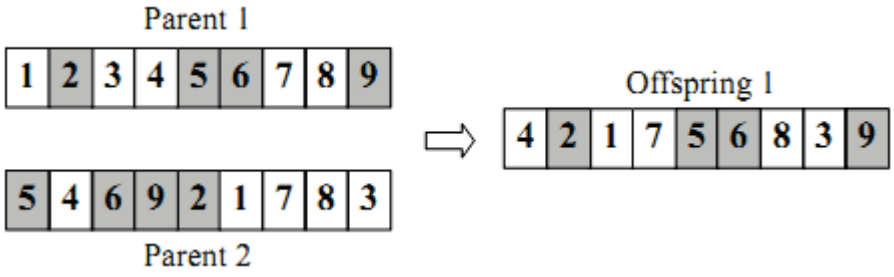
ภาพที่ 2-13 ตัวอย่างวิธีการไขว้เปลี่ยนแบบจัดคู่เป็นส่วน

การไขว้เปลี่ยนเชิงลำดับ (Order Crossover หรือ OX) [27] เป็นวิธีการไขว้เปลี่ยนที่เสนอโดย Davis ซึ่งขั้นตอนการทำงานเริ่มจากการสุ่มเลือกกลุ่มยีนที่ติดกันจากการสุ่มสองตำแหน่งที่ซึ่งอยู่ระหว่างยีนภายในโครโมโซม ตัวอย่างเช่น ในภาพที่ 2-14 [1] ในส่วนของพ่อแม่หมายเลข 1 มีการสุ่มสองตำแหน่งที่อยู่ระหว่างยีน คือ ตำแหน่งระหว่างยีนที่ 2 และ 3 กับตำแหน่งระหว่างยีนที่ 6 และ 7 พร้อมกับสุ่มเลือกหนึ่งตำแหน่งระหว่างยีนของโครโมโซมพ่อแม่หมายเลข 2 และ นำชุดของยีนที่คัดเลือกจากพ่อแม่หมายเลข 1 มาเทียบค่าในโครโมโซมพ่อแม่หมายเลข 2 เพื่อคัดเลือกยีนส่วนที่เหลือจากโครโมโซมพ่อแม่หมายเลข 2 ที่ไม่ซ้ำ มาประกอบรวมกับกลุ่มยีนพ่อแม่หมายเลข 1 กลายเป็นลูกหลานหมายเลข 1 นั่นเอง



ภาพที่ 2-14 ตัวอย่างวิธีการไขว้เปลี่ยนเชิงลำดับ

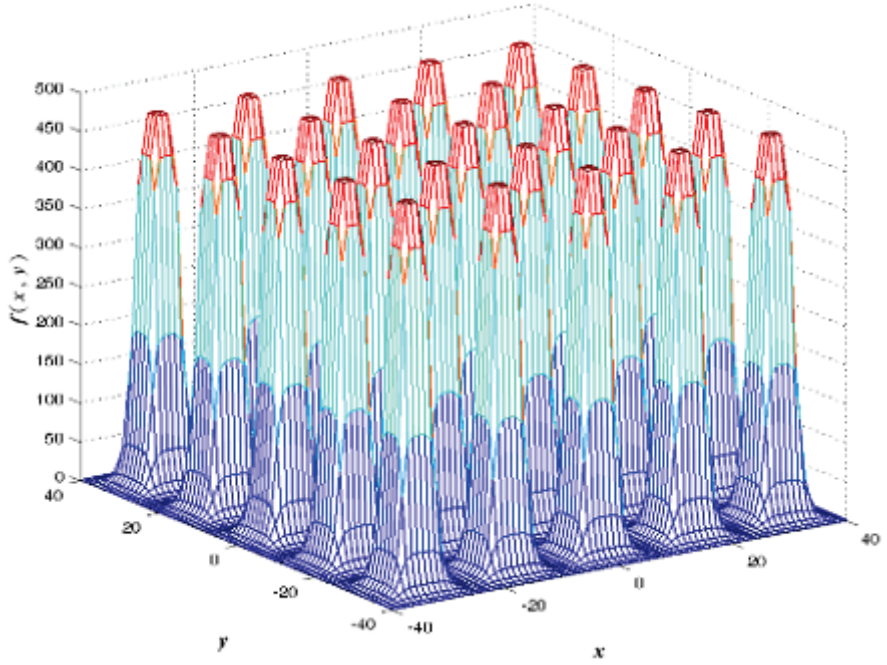
การไขว้เปลี่ยนเชิงตำแหน่ง (Position base crossover หรือ PBX) [28] เป็นวิธีการไขว้เปลี่ยนที่เสนอโดย Syswerda ซึ่งกระบวนการทำงานของวิธีนี้ เริ่มจากเลือกตำแหน่งของยีนแบบสุ่มจากพ่อแม่หมายเลข 1 ตัวอย่างเช่น ในภาพที่ 2-15 [1] ในพ่อแม่หมายเลข 1 มีการสุ่มเลือกตำแหน่งยีนทั้งหมด 4 ตำแหน่ง คือ 2, 5, 6, และ 9 ถัดมาทำการสร้างโครโมโซมลูกหลานหมายเลข 1 โดยการนำแต่ละยีนที่เลือกไว้จากโครโมโซมพ่อแม่หมายเลข 1 มาวางตามตำแหน่งเดิมในโครโมโซมลูกหลาน และสุดท้ายให้ทำสัญลักษณ์ในแต่ละตำแหน่งยีนในโครโมโซมพ่อแม่หมายเลข 2 โดยเลือกทำสัญลักษณ์แต่ยีนที่มีค่าภายในซ้ำซ้อนกับยีนที่ถูกเลือกไว้ก่อนหน้านี้ และคัดแต่ยีนที่ยังไม่ถูกทำสัญลักษณ์มาบรรจุลงช่องที่เหลือในโครโมโซมลูกหลานหมายเลข 1 อย่างเป็นลำดับ



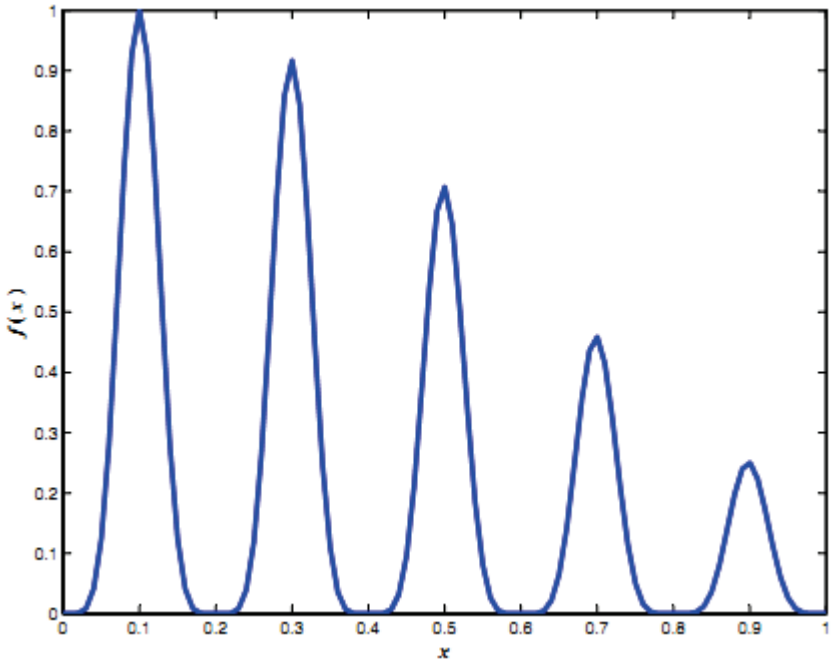
ภาพที่ 2-15 ตัวอย่างวิธีการไขว้เปลี่ยนเชิงตำแหน่ง

2.2 การหาค่าเหมาะที่สุดหลากหลายรูปแบบ

การหาค่าเหมาะที่สุดหลากหลายรูปแบบ (Multimodal Optimization) [1] มีแนวคิดว่าบางครั้งในการใช้ขั้นตอนวิธีเชิงวิวัฒน์ในการแก้ปัญหา ผลลัพธ์ที่ได้ในแต่ละรอบการทำงานแม้ว่าจะไม่ใช่ผลลัพธ์เดียวกันแต่ เมื่อนำกลุ่มผลลัพธ์ดังกล่าวมาเข้าฟังก์ชันประเมินคุณภาพ พบว่า แต่ละผลลัพธ์มีคุณภาพที่เท่ากัน มุ่งประเด็นในการแก้ปัญหาว่าจะทำอย่างไรจึงสามารถค้นหาผลเฉลยทั้งหมดที่มีคุณภาพสูงตามต้องการโดยอาศัยเพียงการค้นหาเพียงครั้งเดียว โดยสามารถแสดงตัวอย่างของผลเฉลยที่เหมาะที่สุดจำนวนมากได้ตามภาพที่ 2-16 [1]



ภาพที่ 2-16 การหาค่าเหมาะที่สุดหลากหลายรูปแบบในกรณีหลายผลเฉลยที่เหมาะที่สุด [1]



ภาพที่ 2-17 การหาค่าเหมาะที่สุดหลากหลายรูปแบบในกรณีหลายผลเฉลยคล้ายที่เหมาะที่สุด

ความหมายของปัญหาหลากหลายรูปแบบยังครอบคลุมถึง ปัญหาที่มีทั้งผลเฉลยที่เหมาะที่สุด และ ผลเฉลยที่เหมาะที่สุดท้องถิ่น (Local optimum solutions) ซึ่งในขั้นตอนวิธีเชิงวิวัฒน์นิยมที่จะออกแบบขั้นตอนวิธีให้สามารถค้นหาผลเฉลยทั้งสองแบบ ซึ่งสามารถแสดงตัวอย่างลักษณะของปัญหาดังกล่าวได้ตามภาพที่ 2-17 [1]

บางครั้งขั้นตอนวิธีดังกล่าวอาจพบได้มากกว่าหนึ่งผลเฉลย อย่างไรก็ตาม ท้ายที่สุดจะไม่สามารถค้นหากลุ่มผลเฉลยที่เหมาะที่สุดต่อไปได้ และเริ่มลู่เข้าสู่ผลเฉลยใดผลเฉลยหนึ่ง

2.3 ขั้นตอนวิธีการบรรจบ

ขั้นตอนวิธีการบรรจบมีการทำงานในแต่ละขั้นตอนดังนี้ [14] อันดับแรกคือการกำหนดค่าเริ่มต้น ให้กับเมทริกซ์ความน่าจะเป็นร่วม (Joint probability matrix) โดยค่าที่กำหนดให้กับแต่ละสมาชิกของตาราง ซึ่งมีแนวคิดที่ว่าทุกเส้นทางในตอนเริ่มต้นจะมีค่าความน่าจะเป็นในการถูกเลือกเท่ากัน ซึ่งสามารถแสดงค่าในรูปของสมการ $\frac{1}{(n-1)}$ โดยที่ n คือ ขนาดของปัญหา และ บางส่วนในตารางที่นำไปสู่เส้นทางที่ซ้ำกันจะมีค่าเป็นศูนย์ตลอดการทำงานของระเบียบวิธีนี้เงื่อนไขดังกล่าว สามารถแทนด้วยสัญลักษณ์ H_{xx} ซึ่งเป็นตัวแปรที่แทนสมาชิกในตำแหน่งแถวที่ x และ หลักที่ x นั่นเอง และ ตัวอย่างของการกำหนดค่าเริ่มต้นให้กับตารางเมื่อมีขนาดของปัญหาเท่ากับ 5 ได้แสดงไว้ดังภาพที่ 2-18 ซึ่งตัวอย่างดังกล่าวสามารถนำมาใช้แก้ปัญหาทางด้านการเดินทางของพนักงานขาย โดยกำหนดให้ x_i แทนเมืองที่ i โดยมีด้วยกันทั้งหมด 5 เมือง และ ความน่าจะเป็นที่พนักงานขายจะเดินทางซ้ำเมืองปัจจุบันย่อมเป็นไปไม่ได้จึงให้มีค่าเท่ากับ 0.0 เป็นต้น

	x_1	x_2	x_3	x_4	x_5
x_1	0.0	0.25	0.25	0.25	0.25
x_2	0.25	0.0	0.25	0.25	0.25
x_3	0.25	0.25	0.0	0.25	0.25
x_4	0.25	0.25	0.25	0.0	0.25
x_5	0.25	0.25	0.25	0.25	0.0

ภาพที่ 2-18 ตัวอย่างของเมทริกซ์ความน่าจะเป็นร่วม

ต่อมาจึงทำการสร้างกลุ่มประชากรตามขนาดประชากรที่กำหนดไว้ โดยการสุ่มเลือกผลเฉลยที่เป็นไปได้ตามความน่าจะเป็นที่กำหนดไว้ในตารางดังกล่าว (เมื่อใช้แก้ปัญหาการเดินทางของพนักงานขายแต่ละผลเฉลยจะแทนการเดินทางครบ 1 รอบที่ผ่านทุกเมืองของพนักงานขายนั่นเอง) โดยแต่ละผลเฉลยจะประกอบไปด้วยตำแหน่งต่างๆจำนวนหนึ่งที่ต่อเนื่องกัน ซึ่งอันดับแรกจะทำการสุ่มเส้นทางเริ่มต้นจากตำแหน่ง x_i ไปยังตำแหน่ง x_j ขึ้นมาก่อน (จะทำส่วนนี้เมื่อปัญหาที่นำมาแก้ไม่ได้บังคับตำแหน่งเริ่มต้น) โดยโอกาสที่แต่ละเส้นทางจะถูกเลือก เป็นไปตามค่า

ความน่าจะเป็นภายในตาราง โดยมีเกณฑ์ในการเลือกคือ ตำแหน่งที่มีค่าความน่าจะเป็นมากจะมีโอกาสถูกเลือกมากและในทางกลับกันตำแหน่งที่มีความน่าจะเป็นน้อยจะมีโอกาสถูกเลือกน้อย ซึ่งในตอนเริ่มต้นยังมีค่าเท่ากันทั้งหมด (ยกเว้นเส้นทางที่ตรงกับเงื่อนไขที่ $H_{xx} = 0.0$) เมื่อเราได้ตำแหน่งแรกและตำแหน่งที่สองมาแล้ว ให้นำตำแหน่งที่สองหรือ x_j มาเป็นจุดเริ่มต้น กล่าวคือนำมาใช้เป็นตำแหน่งที่ x_i ของการสุ่มครั้งถัดไป และทำเช่นนี้ซ้ำอีกครั้งจนกระทั่ง ครบทุกตำแหน่ง ซึ่งถือว่าการสร้างผลเฉลยเสร็จสิ้นหนึ่งผลเฉลยนั่นเอง ทั้งนี้การสุ่มหาผลเฉลยยังคงเป็นไปอย่างต่อเนื่องจนกว่าจะครบตามจำนวนประชากรที่ต้องการ หลังจากนั้นจึงนำประชากรทั้งหมดมาเข้าสู่ขั้นตอนของการประเมินคุณภาพต่อไป

การประเมินคุณภาพของประชากรเป็นการนำแต่ละผลเฉลยที่เป็นไปได้ภายในกลุ่มประชากรมาประเมินค่าผ่านฟังก์ชันการประเมินค่า (Evaluation function) หรือ ฟังก์ชันวัตถุประสงค์ (Objective function) ตัวอย่างของฟังก์ชันดังกล่าวเมื่อเทียบกับปัญหาการเดินทางของพนักงานขายคือ ฟังก์ชันการคำนวณค่าระยะทางนั่นเอง และ ขั้นตอนถัดไปคือการคัดเลือกผลเฉลยตามเกณฑ์ที่ต้องการจากประชากร โดยการคัดเลือกในขั้นตอนวิธีนี้จะคัดเลือกผลเฉลยมาสองกลุ่มคือ กลุ่มที่ดีและกลุ่มที่ไม่ดี ซึ่งอันดับแรกจะทำการจัดลำดับของประชากรทั้งหมด ตามคุณภาพ ถัดมาจึงทำการคัดเลือกโดยมีวิธีการเลือกอยู่สองวิธีคือ Uniform selection ซึ่งเป็นวิธีการคัดเลือกกลุ่มที่ดีและกลุ่มที่ไม่ดีตามเปอร์เซ็นต์ที่กำหนดไว้ และ Adaptive selection จะคัดเลือกกลุ่มที่ดีและกลุ่มที่ไม่ดีที่สูงและต่ำกว่าค่าเฉลี่ย ±2σ ตามลำดับ ซึ่งกลุ่มผลเฉลยที่ถูกเลือกทั้งกลุ่มที่ดีและกลุ่มที่ไม่ดีจะถูกนำมาปรับปรุงเมทริกซ์ความน่าจะเป็นร่วมต่อไป

วิธีการปรับปรุงเมทริกซ์ความน่าจะเป็นร่วมของขั้นตอนวิธีการบรรจวบโดยในขั้นตอนนี้จะเรียกว่าการให้รางวัล และ การทำโทษ (Reward and Punishment) [14] โดยจะนำกลุ่มของผลเฉลยที่ดีและกลุ่มที่ไม่ดีมาทำการให้รางวัลและทำโทษตามลำดับ การให้รางวัลกับเมทริกซ์ความน่าจะเป็นในตำแหน่งแถวที่ x และ หลักที่ y ซึ่งจะถูกปรับเพิ่มด้วยค่า $\frac{k}{(n-1)}$ โดยที่ k คือ step size ซึ่งใช้สำหรับกำหนดน้ำหนักในการให้รางวัลและทำโทษแต่ละครั้ง และ n คือขนาดของปัญหา ทั้งนี้จะมีการปรับค่าที่เหลือภายในแถวเดียวกันอีกด้วยเพื่อที่จะคงผลรวมของความน่าจะเป็นในแต่ละแถวให้มีค่าเท่ากับหนึ่งจึงต้องทำการลดค่าของสมาชิกทั้งหมดภายในแถวด้วยค่า $\frac{k}{(n-1)^2}$ (ยกเว้นเพียงตำแหน่งที่ H_{ii} เท่านั้น) ซึ่งสามารถแทนในรูปของสมการได้ดังนี้

$$H_{x,y}(t+1) = H_{x,y}(t) + \frac{k}{(n-1)}\left(r_{x,y}(t+1)\right) - \frac{k}{(n-1)^2}\left(\sum_{z=1}^{n} r_{x,z}(t+1)\right) \qquad \text{(2-8)}$$

ในทำนองเดียวกัน การทำโทษจะนำเส้นทาง xy ที่ได้จากกลุ่มที่ไม่ดีมาเป็นข้อมูลในการทำโทษ เมทริกซ์ความน่าจะเป็นในตำแหน่งแถวที่ x และ หลักที่ y ซึ่งจะถูกปรับลดค่าด้วย $\frac{k}{(n-1)}$ และรวมถึงการปรับค่าที่เหลือภายในแถวเดียวกัน เพื่อที่จะคงผลรวมของความน่าจะเป็นในแต่ละแถวให้มีค่าเท่ากับหนึ่ง จึงต้องทำการเพิ่มค่าของสมาชิกทั้งหมดภายในแถวด้วยค่า $\frac{k}{(n-1)^2}$ ซึ่งสามารถแทนในรูปของสมการได้ดังนี้

$$H_{x,y}(t+1) = H_{x,y}(t) - \frac{k}{(n-1)}\left(p_{x,y}(t+1)\right) + \frac{k}{(n-1)^2}\left(\sum_{z=1}^{n} p_{x,z}(t+1)\right) \quad \text{(2-9)}$$

และสามารถรวมสมการทั้งสองให้อยู่ในรูปสมการสำหรับการให้รางวัลและทำโทษได้ดังนี้

$$\begin{aligned} H_{x,y}(t+1) &= H_{x,y}(t) - \frac{k}{(n-1)}\left(r_{x,y}(t+1) - p_{x,y}(t+1)\right) \\ &- \frac{k}{(n-1)^2}\left(\sum_{z=1}^{n} r_{x,z}(t+1) - \sum_{z=1}^{n} p_{x,z}(t+1)\right) \end{aligned} \quad \text{(2-10)}$$

สุดท้ายนี้ เมื่อนำกลุ่มผลเฉลยจากกลุ่มที่ดีและกลุ่มที่ไม่ดีมาปรับปรุงตารางเรียบร้อยแล้ว จึงกลับไปเริ่มการทำงานในส่วนของการสร้างประชากรอีกครั้งหนึ่ง รวมถึงทำการประเมินคุณภาพ คัดเลือก และปรับปรุงตารางอีกครั้ง โดยทำซ้ำแบบนี้ไปเรื่อยๆจนกว่าจะตรงกับเงื่อนไขการหยุดตามที่กำหนดเอาไว้จึงจะถือว่าสิ้นสุดการทำงานของขั้นตอนวิธีดังกล่าว

ดังนั้นการทำงานแต่ละขั้นตอนของขั้นตอนวิธีการบรรจบสามารถสรุปได้ดังนี้

1 กำหนดค่าเริ่มต้นให้กับเมทริกซ์ความน่าจะเป็นร่วม

2 สร้างกลุ่มประชากรขึ้นมาจากเมทริกซ์ความน่าจะเป็นร่วม

3 ประเมินคุณภาพของประชากรทั้งหมด

4 เลือกประชากรที่ต้องการ ซึ่งมีระเบียบวิธีในการเลือกอยู่ 2 วิธีด้วยกัน คือ

- Uniform selection: คือ เลือกประชากรที่มีคุณภาพสูงกว่าเกณฑ์และคุณภาพต่ำกว่าเกณฑ์ โดยเกณฑ์ขึ้นอยู่กับการกำหนดค่า C % ว่าต้องการตัดเกณฑ์ที่เปอร์เซ็นต์ที่เท่าไหร่
- Adaptive selection: คือ เลือกประชากรที่มีคุณภาพสูงและต่ำตามค่าเฉลี่ยของ ±2σ

5 สำหรับแต่ละความน่าจะเป็นร่วมในตารางจะถูกปรับค่าตามสมการทั้งการให้รางวัลและการทำโทษ ซึ่งจะได้สมการของการปรับค่า ตามสมการที่ 2-10

6 ย้อนกลับไปทำขั้นตอนที่ 2 ใหม่อีกครั้ง ซึ่งจะทำงานซ้ำจนกว่าจะเป็นไปตามเงื่อนไขที่กำหนดไว้

2.4 ขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบบัพ

ในปัจจุบันขั้นตอนวิธีหนึ่งที่อยู่ในกลุ่มขั้นตอนวิธีเชิงวิวัฒน์ และ ขั้นตอนวิธีประมาณการแจกแจง ที่สามารถแก้ปัญหาการหาค่าเหมาะสุดเชิงการจัดได้อย่างมีประสิทธิภาพ คือ ขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบเส้นเชื่อม (Edge histogram based sampling algorithm หรือ EHBSA) [29] ขั้นตอนวิธีดังกล่าวได้รับการปรับปรุงกระบวนการทำงานเพิ่มเติมเพื่อให้สามารถแก้บางปัญหาได้อย่างมีประสิทธิภาพมากขึ้น โดยขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบบัพ (Node histogram based sampling algorithm หรือ NHBSA) [30] ซึ่งได้มีการเผยแพร่โดย Shigeyoshi Tsutsui, Martin Pelikan และ David E. Goldberg ทั้งนี้กระบวนการค้นหาผลเฉลยที่ออกแบบใหม่ค่อนข้างเป็นกระบวนการที่น่าสนใจสำหรับงานวิจัยนี้อย่างมาก

ในส่วนของแบบจำลองความน่าจะเป็นของขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบบัพ หรือ เมทริกซ์ฮิสโทแกรมแบบบัพ (Node histogram matrix หรือ NHM) เป็นเมทริกซ์ที่แสดงถึงความสัมพันธ์ระหว่างตำแหน่ง (Position) และ บัพ (Node) ของแต่ละผลเฉลยว่าแต่ละบัพมีการกระจายตามตำแหน่งต่างๆภายในผลเฉลยใดๆอย่างไร ซึ่งมีรายละเอียดของการทำงานดังนี้ คือ กำหนดให้ผลเฉลยลำดับที่ k ของประชากรในรุ่นที่ t หรือ $P(t)$ เป็นผลเฉลยหนึ่งซึ่งสามารถแทนด้วย $s_k^t = \{\pi_k^t(0), \pi_k^t(1), ..., \pi_k^t(L-1)\}$ และให้เมทริกซ์ฮิสโทแกรมแบบบัพของประชากร $P(t)$ ซึ่งแทนด้วย $NHM^t(n_{i,j}^t)(i, j = 0,1,...,L-1)$ ประกอบไปด้วยสมาชิกภายในเมทริกซ์ทั้งหมดเท่ากับ L^2 สามารถแสดงในรูปของฟังก์ชันการปรับให้เป็นปัจจุบัน (Update function) ได้ดังนี้

$$n_{i,j}^t = \sum_{k=1}^{N} \delta_{i,j}(s_k^t) + \varepsilon \qquad \text{(2-11)}$$

โดยที่ N คือขนาดของประชากร $\delta_{i,j}(s_k^t)$ คือ ฟังก์ชันเดลตาซึ่งมีนิยามดังนี้

$$\delta_{i,j}(s_k^t) = \begin{cases} 1 & if \quad \pi_k^t(i) = j \\ 0 & otherwise \end{cases} \qquad \text{(2-12)}$$

ซึ่งสามารถกล่าวได้ว่าในส่วนของเลขจำนวนเต็ม $n_{i,j}^t$ ของเมทริกซ์ NHM^t แสดงให้เห็นถึงจำนวนรวมของบัพ j ที่ตำแหน่ง i ของผลเฉลยในประชากร $P(t)$ ซึ่งแทนการกระจายตัวของแต่ละบัพลงในแต่ละตำแหน่งของผลเฉลยใดๆของประชากรนั่นเอง นอกจากนี้ จำนวนโดยเฉลี่ยของแต่ละบัพของ $n_{i,j}^t$ ในเมทริกซ์ NHM^t คือ $LN/L^2 = N/L$ ดังนั้นค่า ε สำหรับเมทริกซ์จะถูกกำหนด

ตามสมการดังสมการที่ 2-13 ซึ่งเป็นองค์ประกอบหนึ่งเพื่อป้องกันการเกิดเหตุการณ์ที่ว่าบางเส้นทางมีโอกาสที่จะไม่ได้รับการปรับปรุงค่า ซึ่งทำให้ความน่าจะเป็นในการถูกเลือกในตำแหน่งนั้นมีค่าเป็นศูนย์ นั่นเอง

$$\varepsilon = \frac{N}{L} B_{ratio} \tag{2-13}$$

ตัวอย่างการปรับปรุงเมทริกซ์ในแต่ละรุ่นโดยที่ $N = 5, L = 5$ และ $B_{ratio} = 0.2$ สามารถแสดงได้ดังภาพที่ 2-19 [30] โดยสมมติว่ามีผลเฉลยที่เป็นไปได้ภายในประชากรทั้งหมด 5 ผลเฉลยด้วยกัน ซึ่งเมื่อพิจารณาการปรับปรุงเมทริกซ์ที่บัพ $j = 0$ และ ตำแหน่ง $i = 4$ พบว่าผลลัพธ์ที่ได้จากการปรับปรุงในตำแหน่งดังกล่าวมีค่าเท่ากับ 3.2 เนื่องมาจากพบความสัมพันธ์ที่ตำแหน่งที่ 1 ถูกวางด้วยบัพที่ 0 ภายในประชากรปัจจุบันด้วยกัน 3 ครั้งนั่นเอง และ มีการบวกเพิ่มค่า ε ที่คำนวณจากสมการที่ 3.2 ซึ่งมีค่าเท่ากับ 0.2 นั่นเอง และ เมื่อพิจารณาการปรับปรุงเมทริกซ์ที่บัพ $j = 2$ และ ตำแหน่ง $i = 0$ พบว่าภายในประชากรปัจจุบันไม่มีความสัมพันธ์ใดๆในแต่ละผลเฉลยที่เป็นไปได้ที่อธิบายว่ามีบัพที่ 2 ถูกวางลงบนตำแหน่งที่ 0 ซึ่งส่งผลให้ความน่าจะเป็นในการเลือกในตำแหน่งดังกล่าวมีค่าเป็นศูนย์ หรือ ไม่มีความเป็นไปได้เลยที่จะถูกเลือกในการสร้างผลเฉลยของประชากรรุ่นถัดไป ดังนั้น การใส่ค่า ε จึงมีความจำเป็นซึ่งจะทำให้ผลลัพธ์ของการคำนวณในตำแหน่งดังกล่าวมีค่าเท่ากับ 0.2 ซึ่งทำให้มีโอกาสถูกเลือกในช่องดังกล่าวนั่นเอง

$$P(t) = \begin{matrix} s^t_0 = (0,\ 1,\ 2,\ 3,\ 4) \\ s^t_1 = (1,\ 3,\ 4,\ 2,\ 0) \\ s^t_2 = (3,\ 4,\ 2,\ 1,\ 0) \\ s^t_3 = (4,\ 0,\ 3,\ 1,\ 2) \\ s^t_4 = (1,\ 4,\ 2,\ 3,\ 0) \end{matrix}$$

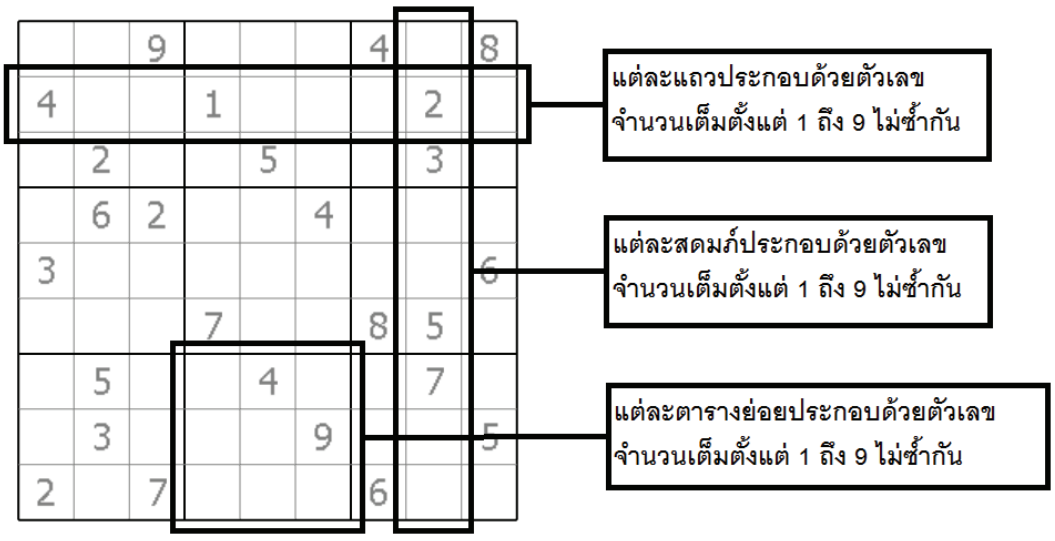
$$\begin{matrix} & \text{node } j \\ \text{position } i & \begin{pmatrix} 1.2 & 2.2 & 0.2 & 1.2 & 1.2 \\ 1.2 & 1.2 & 0.2 & 1.2 & 2.2 \\ 0.2 & 0.2 & 3.2 & 1.2 & 1.2 \\ 0.2 & 2.2 & 1.2 & 2.2 & 0.2 \\ 3.2 & 0.2 & 1.2 & 0.2 & 1.2 \end{pmatrix} \\ & \text{NHM}^t \end{matrix}$$

ภาพที่ 2-19 ตัวอย่างเมทริกซ์ NHM^t สำหรับ $P(t)$ ที่ $N = 5, L = 5$ และ $B_{ratio} = 0.2$ [2-13]

ขั้นตอนวิธีการบรรจวบที่ใช้ในงานวิจัยนี้ มีลักษณะการทำงานของตารางที่คล้ายกับขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบเส้นเชื่อม ดังนั้น จึงมีความเป็นไปได้ว่า เมื่อนำขั้นตอนวิธีการบรรจวบมาออกแบบให้มีความเฉพาะในลักษณะของบัพ เพื่อให้เข้ากับบางปัญหาจะส่งผลให้ขั้นตอนวิธีการบรรจวบสามารถแก้ปัญหาได้อย่างมีประสิทธิภาพมากขึ้น

2.5 ปัญหาซูโดะกุ

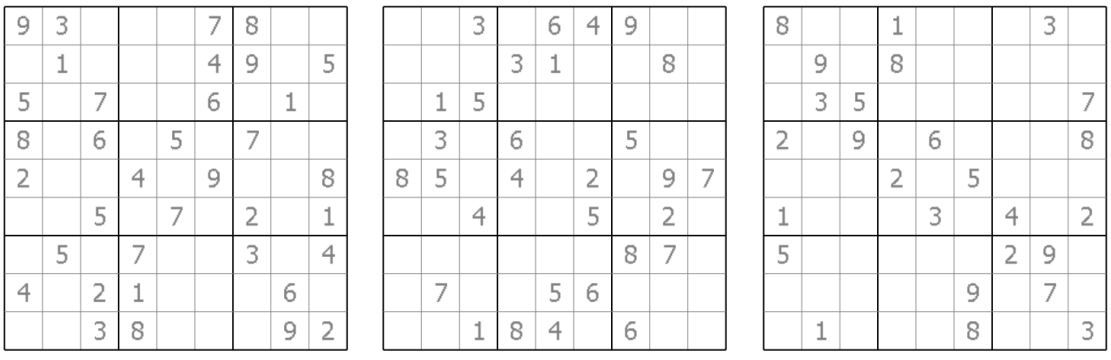
ปัญหาซูโดะกุ (Sudoku Problems หรือ Sudoku Puzzles) เป็นปัญหาเชิงการจัดอีกปัญหาหนึ่งที่ได้รับความนิยมในงานวิจัยทางด้านการคำนวณเชิงวิวัฒน์ [20] โดยปัญหาดังกล่าวมีลักษณะโครงสร้างอยู่ในรูปของตารางที่มีจำนวนแถว (Row) เท่ากับจำนวนสดมภ์ (Column) ที่ซึ่งแต่ละช่องของตารางจะเก็บเลขจำนวนเต็มบวกไว้ และ ปัญหาซูโดะกุที่นิยมโดยทั่วไปมีขนาดเท่ากับ 9×9 ซึ่งจะทำให้ภายในประกอบด้วยตารางย่อย (sub-block) ขนาด 3×3 นั่นเอง โดยตัวอย่างปัญหาซูโดะกำที่มีขนาด 9×9 สามารถแสดงได้ตามภาพที่ 2-20 นั่นเอง โดยมีเงื่อนไขหรือกฎเกณฑ์ในการแก้ปัญหาที่สำคัญดังนี้ คือ แต่ละแถวและสดมภ์รวมถึงตารางย่อยภายในตารางซูโดะกุจะต้องประกอบไปด้วยชุดตัวเลขจำนวนเต็มบวกที่ไม่ซ้ำกัน และ มีเพียงหนึ่งผลเฉลยที่เหมาะที่สุดเท่านั้น กล่าวคือ เมื่ออ้างอิงจากปัญหาซูโดะกุที่มีขนาดเท่ากับ 9×9 จะสามารถอธิบายได้ว่าในแต่ละแถวและสดมภ์รวมถึงภายในตารางย่อยจะประกอบไปด้วยตัวเลขจำนวนเต็มบวกตั้งแต่ 1 ถึง 9 และ ถ้าหากอ้างอิงตามภาพที่ 2-20 ผลเฉลยที่เหมาะที่สุดที่ตรงตามเงื่อนไขที่กล่าวไว้สามารถแสดงได้ตามภาพที่ 2-21 ซึ่งเป็นหนึ่งผลเฉลยที่เหมาะที่สุดสำหรับปัญหาซูโดะกุดังกล่าว



ภาพที่ 2-20 ตัวอย่างปัญหาซูโดะกุที่มีขนาด 9×9

ภาพที่ 2-21 ตัวอย่างปัญหาซูโดะกุขนาด 9×9 และ ผลเฉลยที่เหมาะที่สุดสำหรับปัญหาดังกล่าว

ปัญหาซูโดะกุยังสามารถจัดกลุ่มตามระดับความยาก ซึ่งประกอบด้วย ปัญหาซูโดะกุระดับง่าย ปัญหาซูโดะกุระดับปานกลาง และปัญหาซูโดะกุระดับยาก ซึ่งข้อแตกต่างเบื้องต้นของแต่ละระดับสามารถสังเกตได้จากจำนวนเริ่มต้นของเลขที่ให้มาภายในตาราง กล่าวคือ ยิ่งเลขที่ให้มาในปัญหาซูโดะกุปัญหาหนึ่งมีจำนวนน้อย ก็ยิ่งทำให้ปัญหาดังกล่าวกลายเป็นปัญหาที่ยากขึ้นนั่นเอง นอกจากนี้ระดับความยากของซูโดะกุยังขึ้นกับลักษณะการจัดวางแต่ละเลขจำนวนเต็มภายในตารางซูโดะกุอีกด้วย โดยตัวอย่างของแต่ละปัญหาซูโดะกุที่มีจำนวนเริ่มต้นที่แตกต่างกันสามารถแสดงได้ดังภาพที่ 2-22



ภาพที่ 2-22 ตัวอย่างปัญหาซูโดะกุขนาด 9×9 ในจำนวนตัวเลขเริ่มต้นที่แตกต่างกัน [31]

เมื่อพิจารณาปัญหาซูโดะกุในส่วนของฟังก์ชันความเหมาะสมต่างๆของงานวิจัยก่อนหน้าพบว่าบางปัญหามีผลเฉลยที่เหมาะที่สุดท้องถิ่นอยู่หลายผลเฉลยด้วยกัน ซึ่งเป็นผลเฉลยที่

แตกต่างกัน แต่มีค่าความเหมาะที่เท่ากัน ซึ่งสามารถแสดงได้ดังภาพที่ 2-23 โดยบางผลเฉลยที่มีค่าเหมาะที่สุดท้องถิ่นเมื่อนำมาเปรียบเทียบความแตกต่างกับผลเฉลยที่เหมาะที่สุด พบว่ามีความแตกต่างกันเพียงพอที่จะทำให้การเรียนรู้ของขั้นตอนวิธีเชิงวิวัฒน์ไม่สามารถพบผลเฉลยที่เหมาะที่สุดได้โดยง่าย อย่างไรก็ตามยังคงมีบางผลเฉลยที่เหมาะที่สุดท้องถิ่นที่สามารถนำไปสู่ผลเฉลยที่เหมาะที่สุดได้อีกด้วย ซึ่งจากลักษณะข้างต้นสามารถกล่าวได้ว่า ซูโดะกุจัดอยู่ในกลุ่มปัญหาหลากหลายรูปแบบด้วยเช่นกัน ดังนั้น ขั้นตอนวิธีเชิงวิวัฒน์ที่เหมาะสำหรับการแก้ปัญหาเชิงการจัด และ ปัญหาหลากหลายรูปแบบ จึงมีความสำคัญอย่างมากต่อการแก้ปัญหาดังกล่าว

6	5	3	4	2	7	9	1	6
2	9	1	8	3	6	7	5	4
4	7	8	1	9	5	2	8	3
1	2	7	3	8	9	6	4	5
8	4	5	6	1	2	3	9	7
9	3	6	5	7	4	8	2	1
5	6	2	9	4	3	1	7	8
7	8	4	2	6	1	5	3	9
3	1	9	7	5	8	4	6	2

(a) local optimum soluiton

8	5	3	4	9	7	2	1	6
6	9	1	6	5	3	7	8	4
4	7	2	1	2	8	9	5	3
1	2	5	3	8	9	6	4	7
7	4	8	2	1	6	3	9	5
9	3	6	5	7	4	8	2	1
5	1	7	9	3	2	4	6	8
2	8	4	7	6	1	5	3	9
3	6	9	8	4	5	1	7	2

(b) local optimum solution

6	3	8	4	2	7	9	5	1
2	5	1	6	9	3	7	8	4
4	9	7	1	5	8	2	6	3
7	2	5	3	8	9	1	4	6
8	4	3	2	1	6	3	9	7
9	1	6	5	7	4	8	2	5
5	6	2	9	3	1	4	7	8
1	8	4	7	6	2	5	3	9
3	7	9	8	4	5	6	1	2

(c) local optimum solution

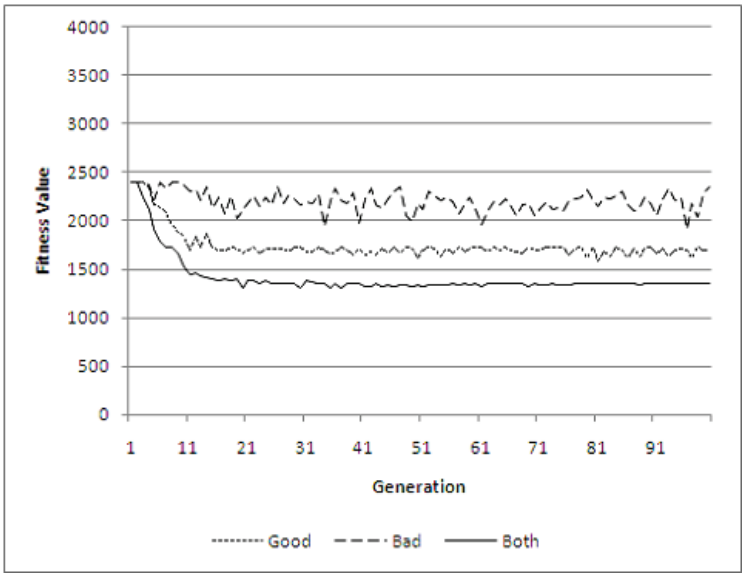
8	5	3	4	9	7	2	1	6
6	9	1	2	3	8	7	5	4
4	7	2	1	5	6	9	8	3
1	2	5	3	8	9	6	4	7
7	4	8	6	1	2	3	9	5
9	3	6	5	7	4	8	2	1
5	1	7	9	2	3	4	6	8
2	8	4	7	6	1	5	3	9
3	6	9	8	4	5	1	7	2

(d) global optimum solution

ภาพที่ 2-23 ตัวอย่างของผลเฉลยเหมาะที่สุดท้องถิ่น (a), (b) และ (c) และผลเฉลยเหมาะที่สุด (d) ของปัญหาซูโดะกุปัญหาหนึ่ง

2.6 งานวิจัยที่เกี่ยวข้อง

2.6.1 ขั้นตอนวิธีการบรรจวบกับปัญหา TSPs จากงานวิจัย Multi-objective Combinatorial Optimization with Coincidence Algorithm [14] เป็นการประยุกต์ขั้นตอนวิธีการบรรจวบสำหรับปัญหา TSPs ถึงแม้ว่าขั้นตอนวิธีการบรรจวบจะเป็นขั้นตอนวิธีใหม่ แต่ขั้นตอนวิธีดังกล่าวได้ถูกสำรวจประสิทธิภาพในหลายปัญหาเชิงการจัด รวมทั้งมีการทดลองของขั้นตอนวิธีการบรรจวบในกรณีที่ว่า ถ้าหากยกเลิกการทำงานในส่วนการเรียนรู้เชิงลบ เพื่อให้เกิดการเรียนรู้เชิงบวกอย่างเดียว ผลลัพธ์ที่ได้จะเป็นอย่างไร รวมถึงมีการทดลองยกเลิกการทำงานเชิงลบด้วย และ ผลการทดลองได้ชี้ให้เห็นว่าการเรียนรู้ของขั้นตอนวิธีนี้สามารถทำงานอย่างมีประสิทธิภาพมากกว่าเมื่อมีการเรียนรู้เชิงบวกและเชิงลบไปพร้อมๆกัน โดยปัญหาที่ใช้ในการทดลองนี้คือ Gröstel24 [14] ซึ่งเป็นหนึ่งในปัญหาการเดินทางของพนักงานขาย ซึ่งคัดเลือกมาจาก TSPLIB [32] และ แสดงผลการทดลองในรูปของกราฟ [14] ดังภาพที่ 2-24 และ จากงานวิจัยที่ผ่านมาพบว่า ขั้นตอนวิธีการบรรจวบสามารถนำมาแก้ไขปัญหาประเภทการหาค่าเหมาะที่สุดเชิงการจัดได้อย่างมีประสิทธิภาพ ตัวอย่างของปัญหาที่ใช้ในการพิสูจน์ คือ ปัญหาการเดินทางของพนักงานขาย โดยเทียบกับขั้นตอนวิธีอื่นๆ ในด้านคุณภาพของคำตอบ และ จำนวนของ function evaluations โดยใช้ Gröstel24 สำหรับปัญหาในการทดสอบนี้



ภาพที่ 2-24 กราฟแสดงผลการทดลองขั้นตอนวิธีการบรรจวบเมื่อเกิดการเรียนรู้เชิงบวกหรือเชิงลบอย่างใดอย่างหนึ่งเทียบกับการเรียนรู้ไปพร้อมๆกัน [14]

มีการเปรียบเทียบขั้นตอนวิธีการบรรจวบกับวิธีการอื่นๆ มากมาย คือ ขั้นตอนวิธีในกลุ่มของขั้นตอนวิธีประมาณการแจกแจง ได้แก่ UMDA, MIMIC, TREE [33], EBNA [34] ทั้งนี้ยังใช้

ขั้นตอนวิธีเชิงพันธุกรรม (เป็นขั้นตอนวิธีเชิงพันธุกรรมในการทบทวนวรรณกรรมของ Larrañaga [35] ปี 1999) ซึ่งใช้อัลกอริทึม GENITOR [36] ได้แก่ GA-ER, และ GA-OX2 ในการการทดลองนี้อีกด้วย และ จากผลการทดลองในตารางที่ 2-4 และ 2-5 [14] สามารถสรุปได้ว่า ขั้นตอนวิธีการบรรจวบทำได้ดีกว่าเทียบกับขั้นตอนวิธีอื่นๆภายในกลุ่มขั้นตอนวิธีประมาณการแจกแจงเดียวกัน และ ผลที่ได้นั้นยังไม่ผ่านการทำการหาค่าเหมาะที่สุดเฉพาะที่ (Local Optimization) อีกด้วย แม้แต่จำนวนรุ่น (Generation) ที่ใช้หาคำตอบ ขั้นตอนวิธีการบรรจวบยังคงทำได้ดีกว่าในกรณีเมื่อใช้ขนาดประชากรที่ใหญ่ขึ้น อย่างไรก็ตาม ขั้นตอนวิธีการบรรจวบไม่เพียงแต่แก้ปัญหาได้ดีในปัญหาการหาค่าเหมาะที่สุดเชิงการจัด แต่ยังสามารถแก้ปัญหาการหาค่าเหมาะที่สุดเชิงการจัดแบบหลายวัตถุประสงค์ได้เช่นกัน [14]

ตารางที่ 2-4 ผลการทดลองแสดงผลเฉลยที่พบของแต่ละขั้นตอนวิธีในปัญหา Gröstel24 [14]

Algorithm	Population & Local Optimization							
	500-without		500-with		1000-without		1000-with	
	Best	Aver	Best	Aver	Best	Aver	Best	Aver
GA-ER*	1272	1272						
GA-OX2*	1300	1367						
UMDA	1339	1495	1272	1272	1329	1496	1272	1272
MIMIC	1391	1486	1272	1272	1328	1451	1272	1272
TREE	1413	1486	1272	1272	1429	1442	1272	1272
EBNA	1431	1528	1272	1272	1329	1439	1272	1272
COIN *unif*	1272	1280			1272	1278		
COIN *adpt*	1272	1272			1272	1272		

* Size of population 200, mutation used SM
unif denotes uniform selection with learning step $k = 0.1$
adpt denotes adaptive selection with learning step $k = 0.1$
Optimum 1272

ตารางที่ 2-5 ผลการทดลองแสดงจำนวนของรุ่นที่ใช้ค้นหาในปัญหา Gröstel24 [14]

Algorithm	Population & Local Optimization			
	500-without	500-with	1000-without	1000-with
	Gen	Gen	Gen	Gen
UMDA	75	19	78	12
MIMIC	47	4	58	4
TREE	37	4	46	2
EBNA	72	16	79	7
COIN	67		48	

ปัญหาการจัดลำดับการผลิตบนสายการประกอบผลิตภัณฑ์ผสมแบบตัวยู คือตัวอย่างหนึ่งที่ถูกนำมาใช้สำหรับพิสูจน์คุณสมบัติการแก้ปัญหาแบบหลายวัตถุประสงค์ของขั้นตอนวิธีเชิงการจัด [16] เนื่องจากปัญหานี้มีคุณสมบัติตามต้องการ คือ เป็นปัญหาที่อยู่ในกลุ่มของการหาค่าเหมาะที่สุดเชิงการจัด และ สามารถออกแบบการแก้ปัญหาให้เป็นแบบหลายวัตถุประสงค์ได้ ทั้งนี้

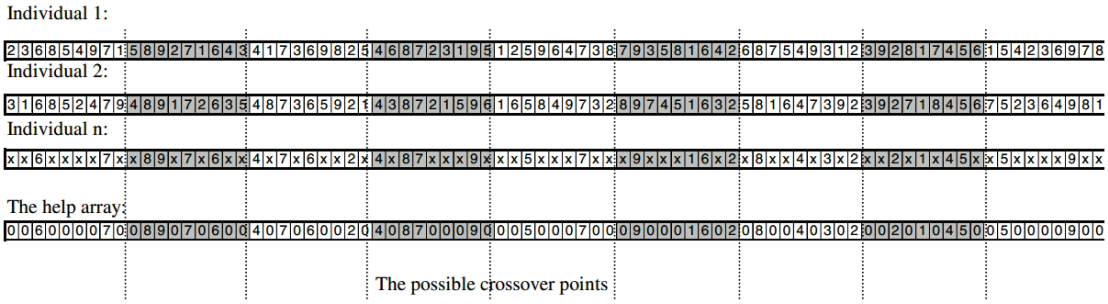
ในการทดลองยังได้นำขั้นตอนวิธีอย่าง Non-dominated Sorting Genetic Algorithm II (NSGA-II) [1][37] ซึ่งเป็นขั้นตอนวิธีที่นิยมใช้แก้ปัญหาแบบหลายวัตถุประสงค์มาเปรียบเทียบด้วย และการทดลองนี้ทำการวัดประสิทธิภาพในสามปัจจัย [16] ด้วยกันคือ การลู่เข้าคำตอบ (Convergence) การกระจายตัวของคำตอบที่พบ (Spread) และ อัตราส่วนของกลุ่มคำตอบที่พบเทียบกับกลุ่มคำตอบที่แท้จริง (Ratio) รวมถึงเวลาที่ใช้ในการคำนวณด้วย ซึ่งผลการทดลองของขั้นตอนวิธีการบรรจวบเมื่อเทียบกับ NSGA-II แสดงให้เห็นว่าขั้นตอนวิธีการบรรจวบโดยรวมทำได้ดีกว่า [14][16] ไม่ว่าจะเป็นด้านการลู่เข้าคำตอบ หรือ อัตราส่วนของกลุ่มคำตอบ รวมถึง เวลาในการคำนวณที่เร็วกว่าด้วย

2.6.2 <u>ขั้นตอนวิธีการบรรจวบกับปัญหาหลากหลายรูปแบบ</u> จากงานวิจัย The Effectiveness of Hybrid Negative Correlation Learning in Evolutionary Algorithm for Combinatorial Optimization Problems [15] ซึ่งนอกจากที่กล่าวไว้แล้วนั้น ขั้นตอนวิธีการบรรจวบยังสามารถแก้ปัญหาการหาค่าเหมาะที่สุดหลากหลายรูปแบบได้อย่างมีประสิทธิภาพด้วยเช่นกัน [15][38] กล่าวคือ มีการนำขั้นตอนวิธีการบรรจวบมาทดสอบประสิทธิภาพกับปัญหาต่างๆ ซึ่งมีคุณสมบัติของเป็นปัญหาเชิงการจัดและปัญหาหลากหลายรูปแบบรวมอยู่ด้วย เช่น ปัญหาควีน 8 ตัว (8-Queens puzzles) ปัญหาการเดินอัศวิน (Knight's tour problems) ปัญหาจตุรัสกล (Magic square problems) เป็นต้น และการทดลองดังกล่าวยังเป็นการเทียบประสิทธิภาพกับขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบเส้นเชื่อม ซึ่งเป็นขั้นตอนวิธีหนึ่งในกลุ่มขั้นตอนวิธีประมาณการแจกแจงเช่นกัน และ ขั้นตอนวิธีดังกล่าวยังถูกออกแบบสำหรับแก้ปัญหาเชิงการจัดโดยเฉพาะอีกด้วย และเมื่อพิจารณาผลลัพธ์ที่ได้จากการทดสอบประสิทธิภาพของทั้งสองขั้นตอนวิธีพบว่า ขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมเส้นเชื่อมสามารถค้นหาคำตอบได้เร็วกว่าขั้นตอนวิธีการบรรจวบ อย่างไรก็ตาม เมื่อพิจารณาในส่วนของจำนวนของผลเฉลยที่เหมาะที่สุดที่แตกต่างกันที่พบโดยเฉลี่ยพบว่าจำนวนของผลเฉลยที่เหมาะที่สุดที่ขั้นตอนวิธีการบรรจวบสามารถค้นหาได้นั้นมีจำนวนมากกว่าของขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมเส้นเชื่อม ดังตารางที่ 2-6 [38] ซึ่งแสดงให้เห็นว่าขั้นตอนวิธีการบรรจวบสามารถแก้ปัญหาหลากหลายรูปแบบได้อย่างมีประสิทธิภาพ โดยขั้นตอนวิธีดังกล่าวสามารถคงสภาพหรือรักษาโครงสร้างที่เป็นไปได้ในการประกอบแต่ละผลเฉลยที่เหมาะที่สุดที่แตกต่างกันนั่นเอง

ตารางที่ 2-6 ผลการทดลองโดยเปรียบเทียบของ COIN ในปัญหาเชิงการจัดต่างๆ [38]

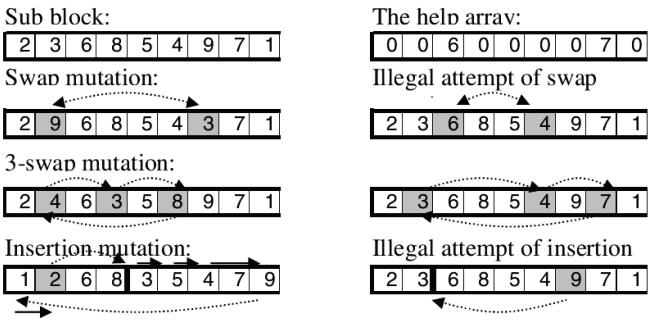
Problem	Algorithm					
	EHBSA			COIN		
	ANE	#SOL	#DSOL	ANE	#SOL	#DSOL
8 Queens-P	8	25	4	8	21	13
8 Queens-C	1821	78	4	3651	10	9
8 Rooks	25	2457	495	454	4	4
14 Bishops	419	408	4	1070	45	8
32 Knights	N/A	0	0	N/A	0	0
Knight's Tour	N/A	0	0	154	2816	2759
3x3 Magic Square	N/A	0	0	35	40	2
4x4 Magic Square	N/A	0	0	N/A*	0	0

2.6.3 ขั้นตอนวิธีเชิงพันธุกรรมแบบที่ 1 กับปัญหาซูโดะกุ จากงานวิจัย Solving and Rating Sudoku Puzzles with Genetic Algorithms เป็นงานวิจัยของ Timo Mantere และ Janne Koljonen [21] ได้เสนอแนวคิดในการออกแบบขั้นตอนวิธีเชิงพันธุกรรมในการแก้ปัญหาซูโดะกุ โดยการออกแบบแต่ละผลเฉลยที่เป็นไปได้จะแสดงในรูปของสายจำนวนเต็มที่มีความยาวเท่ากับ 81 ช่อง (มาจากขนาด 9×9 ของซูโดะกุ) ซึ่งภายในประกอบไปด้วยสายย่อยความยาวเท่ากับ 9 ช่องจำนวน 9 สาย (แทนตารางย่อยขนาด 3×3 ทั้งหมด 9 ตาราง ภายตารางซูโดะกุ) ซึ่งแต่ละสายจะเก็บเลขจำนวนเต็มตั้งแต่ 1 ถึง 9 ที่ไม่ซ้ำกัน และ เนื่องจากแต่ละปัญหาซูโดะกุจะใส่เลขจำนวนเต็มไว้ในตารางบางส่วนในตอนเริ่มต้น จึงมีความจำเป็นในการสร้างสายจำนวนเต็มที่เก็บปัญหาเริ่มต้นไว้ เพื่อที่จะคอยป้องกันการเกิดผลเฉลยที่ไม่สมเหตุสมผลหรือขัดแย้งกับโจทย์เริ่มต้นนั่นเอง และ วิธีดำเนินการที่ใช้นั้นในส่วนของของการดำเนินการไขว้เปลี่ยน ตำแหน่งที่จะทำการไขว้เปลี่ยนได้จะเป็นรอยต่อระหว่างสายย่อยเท่านั้น โดยมีลักษณะการไขว้เปลี่ยนแบบ Uniform Crossover ซึ่งแสดงดังภาพที่ 2-25 [21] และในส่วนของการกลายพันธุ์จะดำเนินการภายในสายย่อยเท่านั้น การกลายพันธุ์ที่ใช้จะมีอยู่สามวิธีที่ใช้ในปัญหาการหาค่าเหมาะที่สุดเชิงการจัด ได้แก่ Swap mutation, 3-swap mutation,และ Insertion mutation และ ในส่วนของการคัดเลือก จะเลือกใช้วิธีการคัดเลือกแบบ Elitism ในการคัดเลือกลูกหลาน เพื่อเป็นพ่อแม่ในรุ่นถัดไป



ภาพที่ 2-25 ตัวอย่างของแต่ละผลเฉลยที่แทนด้วยสายจำนวนเต็ม ซึ่งแทนโจทย์เริ่มต้น และ จุดที่สามารถทำการไขว้เปลี่ยนซึ่งตรงกับรอยต่อระหว่างสายย่อยแต่ละสายของเฉลย [21]

ในแต่ละครั้งของการกลายพันธุ์จะต้องคำนึงถึงโจทย์เริ่มต้นด้วย โดยหากว่าจุดที่เกิดการกลายพันธุ์คาบเกี่ยวกับโจทย์ที่ให้ไว้ กระบวนการดังกล่าวจะถูกยกเลิกพร้อมกับทำการเลือกตำแหน่งใหม่อีกครั้ง การตรวจสอบดังกล่าวจะอ้างอิงกับสายจำนวนเต็มที่เก็บโจทย์เริ่มต้นไว้นั่นเอง และในส่วนวิธีการกลายพันธุ์ทั้งสามแบบตามที่กล่าวไว้จะมีการทำงานดังนี้ คือ Swap mutation จะเป็นแลกเปลี่ยนค่าจำนวนเต็มจากสองตำแหน่งที่ถูกเลือกไว้ ในส่วนของ 3-swap mutation จะเลือกมาสามตำแหน่งแล้วสลับเป็นลำดับตามแต่ละลำดับตำแหน่งเป็นวงวน และสุดท้ายในส่วนของ Insertion จะทำการเลือกตำแหน่งที่เก็บค่าจำนวนเต็มและตำแหน่งที่จะแทรกลงไป แล้วนำค่าจำนวนเต็มดังกล่าวแทรกลงไปในตำแหน่งดังกล่าว ทั้งนี้กระบวนการแทรกต้องคำนึงถึงตำแหน่งของโจทย์ด้วย โดยการทำงานดังกล่าวสามารถแสดงให้เห็นดังภาพที่ 2-26 [21]



ภาพที่ 2-26 วิธีการกลายพันธุ์ทั้งสามรูปแบบ [21]

หลังจากที่ผ่านกระบวนการสร้างประชากร ทำการไขว้เปลี่ยน และ การกลายพันธุ์แล้ว ต่อมาจะเป็นการประเมินคุณภาพของแต่ละผลเฉลยในประชากรโดยมีแนวคิดที่ว่าหากมีการออกแบบฟังก์ชันความเหมาะสมที่ถูกต้องประชากรที่ดีจะมีโอกาสถูกเลือกมากขึ้นตามค่าความเหมาะที่สูงขึ้น โดยมีแนวคิดการออกแบบที่ว่าแต่ละแถว และ แต่ละหลักต้องเก็บค่าจำนวนเต็มที่อยู่ในช่วง 1 ถึง 9 ซึ่งมีค่าไม่ซ้ำกัน ซึ่งสองสมการแรกแทนผลรวมของค่าจำนวนเต็มทุกค่าของแต่

ละแถวและสดมภ์ ที่จะมีค่าเท่ากับ 45 ดังสมการที่ 2-14 ในส่วนของสองสมการถัดมาแทนผลคูณของค่าจำนวนเต็มทุกค่าของแต่ละแถวและหลักซึ่งควรจะมีค่าเท่ากับ 9! ดังแสดงในสมการที่ 2-15

$$g_{i1}(x) = \left| 45 - \sum_{j=1}^{9} x_{i,j} \right|$$
$$g_{j1}(x) = \left| 45 - \sum_{i=1}^{9} x_{i,j} \right| \quad \text{(2-14)}$$

$$g_{i2}(x) = \left| 9! - \prod_{j=1}^{9} x_{i,j} \right|$$
$$g_{j2}(x) = \left| 9! - \prod_{i=1}^{9} x_{i,j} \right| \quad \text{(2-15)}$$

สุดท้ายนี้สมการที่ต้องการสำหรับการคำนวณค่าความเหมาะอีกสมการหนึ่งอาศัยแนวคิดจากทฤษฎีเซต กล่าวคือ เป็นสมการที่มีความหมายว่าแต่ละแถวที่ $\mathbf{x_i}$ และหลักที่ $\mathbf{x_j}$ ต้องเท่ากับเซตของ A = {1,2, …, 9} โดยสมการดังกล่าวจะคำนวณค่าจำนวนของตัวเลขที่หายไปในเซตของแต่ละแถว (x_i) และหลัก (x_j) ซึ่งแสดงรูปสมการไว้ดังสมการที่ 2-12 และ เพื่อให้ได้ผลเฉลยเหมาะสุดสมการทั้งสามจะต้องมีค่าเท่ากับศูนย์ทั้งหมด และทั้งนี้เมื่อรวมสมการทั้งสามส่วนเข้าด้วยกันจะได้รูปสมการดังสมการที่ 2-13 ซึ่งจะเห็นได้ว่ามีการลงโทษค่อนข้างมากเมื่อมีตัวเลขหายไป และยังเป็นส่วนช่วยในการลงโทษถ้าสมการของผลบวก (สมการที่ 2-11) และผลคูณ (สมการที่ 2-12) ไม่ตรงกับผลเฉลยเหมาะสุดอีกด้วย

$$A = \{1,2,3,4,5,6,7,8,9\}$$
$$g_{i3}(x) = |A - x_i|$$
$$g_{j3}(x) = |A - x_j| \quad \text{(2-16)}$$

$$f(x) = 10 \times \left(\sum_i g_{i1}(x) + \sum_j g_{j1}(x) \right) + \sum_i \sqrt{g_{i2}(x)} + \sum_j \sqrt{g_{j2}(x)}$$
$$+ 50 \times \left(\sum_i g_{i3}(x) + \sum_j g_{j3}(x) \right) \quad \text{(2-17)}$$

ในส่วนของการทดลองมีการกำหนดค่าต่างๆ คือ การกลายพันธุ์จะใช้ทั้ง Swap mutation, 3-swap mutation และ Insertion mutation โดยมีค่าความน่าจะเป็นในการเลือกในอัตราส่วน

50:30:20 ตามลำดับและกระทำภายในสายย่อยของแต่ละผลเฉลยในประชากรในอัตราความน่าจะเป็นเท่ากับ 0.12 และมีการกำหนดขนาดของประชากรไว้ที่ 100 และกำหนดขนาดของการคัดเลือกแบบ elitism ไว้ที่ 40 โดยจะทดสอบกับโจทย์ปัญหาซูโดะกุจำนวน 10 ปัญหาด้วยกัน ในส่วนของ 5 ปัญหาแรกจะมีระดับความยากตั้งแต่ 1 ถึง 5 ดาว ซึ่งจะให้ค่าเริ่มต้นตั้งแต่ 28 ถึง 33 ค่า และ 4 ปัญหาสุดท้ายจะแบ่งความยากออกเป็น Easy, Challenging, Difficult และ Super difficult ซึ่งให้ค่าเริ่มต้นระหว่าง 23 ถึง 36 และสุดท้ายยังมีการทดลองกับซูโดะกุที่ไม่ได้กำหนดค่าเริ่มต้นมาด้วย (New) โดยผลการทดลองสามารถแสดงดังตารางที่ 2-7 [21]

ตารางที่ 2-7 ผลการทดลองแสดงจำนวนครั้งที่พบผลเฉลยที่เหมาะที่สุด และ ค่าทางสถิติที่อ้างอิงตามจำนวนของรุ่นที่พบผลเฉลยที่เหมาะที่สุดของ GA สำหรับปัญหาซูโดะกุ [21]

Difficulty rating	*Givens*	*Count*	*Min*	*Max*	*Average*	*Median*	*Stdev*
New	0	100	206	3824	1390.4	1089	944.67
1 star	33	100	184	23993	2466.6	917	3500.98
2 stars	30	69	733	56484	11226.8	7034	11834.68
3 stars	28	46	678	94792	22346.4	14827	24846.46
4 stars	28	26	381	68253	22611.3	22297	22429.12
5 stars	30	23	756	68991	23288.0	17365	22732.25
Easy	36	100	101	6035	768.6	417	942.23
Challenging	25	30	1771	89070	25333.3	17755	23058.94
Difficult	23	4	18999	46814	20534.3	26162	12506.72
Super difficult	22	6	3022	47352	14392	6722	17053.33

จากผลการทดลองโดยรวมแสดงให้เห็นว่าขั้นตอนวิธีเชิงพันธุกรรมสามารถแก้ปัญหาทางด้านซูโดะกุได้จริง แต่ประสิทธิภาพในการแก้ปัญหายังไม่ดีเท่าที่ควรเมื่อเทียบกับขั้นตอนวิธีอื่นที่ไม่ได้อยู่ในกลุ่มขั้นตอนวิธีเชิงวิวัฒน์ อย่างไรก็ตาม งานวิจัยนี้เป็นงานที่มุ่งเน้นเพื่อการทดสอบประสิทธิภาพในการแก้ปัญหาซูโดะกุของขั้นตอนวิธีเชิงพันธุกรรมที่ปราศจากการใช้กฎเฉพาะเจาะจงในการแก้ปัญหาดังกล่าว ดังนั้น ผู้เขียนงานวิจัยดังกล่าวจึงเชื่อว่าขั้นตอนวิธีเชิงพันธุกรรมยังสามารถเพิ่มประสิทธิภาพได้อีก หากปรับเพิ่มกฎที่สัมพันธ์กับปัญหาเข้าไปในขั้นตอนวิธีดังกล่าว รวมถึงการออกแบบกระบวนการทำงานเชิงลึกเพิ่มเติมที่ควรมีในงานวิจัยถัดไปอีกด้วย

2.6.4 <u>ขั้นตอนวิธีเชิงพันธุกรรมแบบที่ 2 สำหรับปัญหาซูโดะกุ</u> จากงานวิจัน Solving, Rating and Generating Sudoku Puzzles with GA [19] เป็นงานวิจัยของนักวิจัยชื่อ Timo Mantere และ Janne Koljonen ซึ่งมีการปรับปรุงกระบวนการทำงานภายในขั้นตอนวิธีเชิงพันธุกรรม ที่เคยออกแบบไว้ในงานวิจัยก่อนหน้า [21] โดยในส่วนของวิธีการกลายพันธุ์จะเลือกใช้เพียงวิธีเดียวคือ swap mutation จากเดิมที่มีทั้งหมดสามวิธีโดยสองวิธีที่ยกเลิกไปคือ 3-swap

mutation และ insertion mutation โดยมีเงื่อนไขเพิ่มเติมคือจะมีโอกาสเกิดการกลายพันธุ์ที่สลับค่ากันตั้งแต่ สลับกันเพียงสองค่า จนถึงติดต่อกันห้าค่า (2-, 3-, 4-, หรือ 5-swap mutation โดย 2-swap mutation ก็คือ swap mutation แบบเดิมนั่นเอง) โดยการสลับค่าตั้งแต่สามเป็นต้นไปจะมีพฤติกรรมการสลับค่าเช่นเดียวกับ 3-swap mutation โดยสลับตามเข็มนาฬิกาอย่างเป็นลำดับจนครบตามขนาดของการสลับในครั้งนั้น

ในส่วนของฟังก์ชันความเหมาะสม มีการปรับปรุงโดยพิจารณาเพียงค่าที่หายไปในแต่ละแถวและสดมภ์เท่านั้น กล่าวคือ สมการย่อยในงานวิจัยก่อนหน้า [21] จะมีเพียงสมการที่ 6 ที่ถูกเลือกใช้ในการประเมินประชากรในงานวิจัยนี้เท่านั้น และ วิธีที่เพิ่มเข้ามาคือ หากเกิดกรณีที่ผลเฉลยเหมาะสุดของทั้งพ่อแม่และลูกหลาน เป็นผลเฉลยเดียวกัน ค่าความเหมาะของผลเฉลยดังกล่าวจะถูกเพิ่มเข้าทีละ 1 ในทุกๆครั้งที่เกิดเหตุการณ์ดังกล่าว เพื่อให้ผลเฉลยเหมาะสุดที่ยังคงถูกเลือกเป็นลูกหลานซ้ำหลายครั้งไม่สามารถคงอยู่ได้นาน เนื่องจากต้องการให้เกิดความหลากหลายของประชากร และ การปรับปรุงทั้งหมดนี้นำไปสู่ผลการทดลองที่แสดงให้เห็นว่าประสิทธิภาพในการแก้ปัญหาซูโดะกุเพิ่มสูงขึ้น

ในส่วนของการทดลอง ได้ทำการทดลองเทียบกับการทดลองในงานวิจัยก่อนหน้า [21] ประกอบไปด้วยโจทย์ปัญหาซูโดะกุจำนวน 9 ปัญหาด้วยกัน คือ ปัญหา 1-5 stars และ ปัญหาที่ระดับความยาก Easy, Challenging, Difficult, และ Super difficult โดยมีการปรับปรุงค่าเริ่มต้นต่างๆ ได้แก่ จำนวนรุ่นมากสุดที่ขั้นตอนวิธีเชิงพันธุกรรมจะทำงานได้ โดยจากเดิมได้กำหนดไว้สูงถึง 5,000,000 รุ่น ถูกลดจำนวนเหลือเพียง 100,000 รุ่น และการกลายพันธุ์กำหนดไว้ที่ 0.6 ค่าการคัดเลือก (Elitism) เท่ากับ 1 และ จำนวนประชากรกำหนดไว้ที่ 21 ซึ่งแสดงผลการทดลองดังตารางที่ 2-8 [19] ซึ่งในส่วนของวิธีการใหม่จะแสดงให้เห็นทั้งแบบที่ไม่จำกัดจำนวนรุ่นในการหาผลเฉลยเหมาะสุดและที่จำกัดจำนวนไว้ที่ 100,000 รุ่น พบว่า หากไม่มีการจำกัดจำนวนรุ่นขั้นตอนวิธีเชิงพันธุกรรมที่ได้รับการปรับปรุงสามารถแก้โจทย์ซูโดะกุได้ทุกครั้งในทุกระดับความยาก และ ถึงแม้ว่าจะมีการจำกัดจำนวนรุ่นในการหาคำตอบ ผลการทดลองที่ได้ยังคงแสดงให้เห็นถึงประสิทธิภาพที่เพิ่มขึ้นเมื่อเทียบกับวิธีการแบบเดิม

ตารางที่ 2-8 ผลการทดลองเปรียบเทียบระหว่างวิธีก่อนหน้า วิธีปัจจุบัน และ วิธีปัจจุบันที่ไม่จำกัดจำนวนรุ่นในการค้นหาผลเฉลยที่เหมาะที่สุด [19]

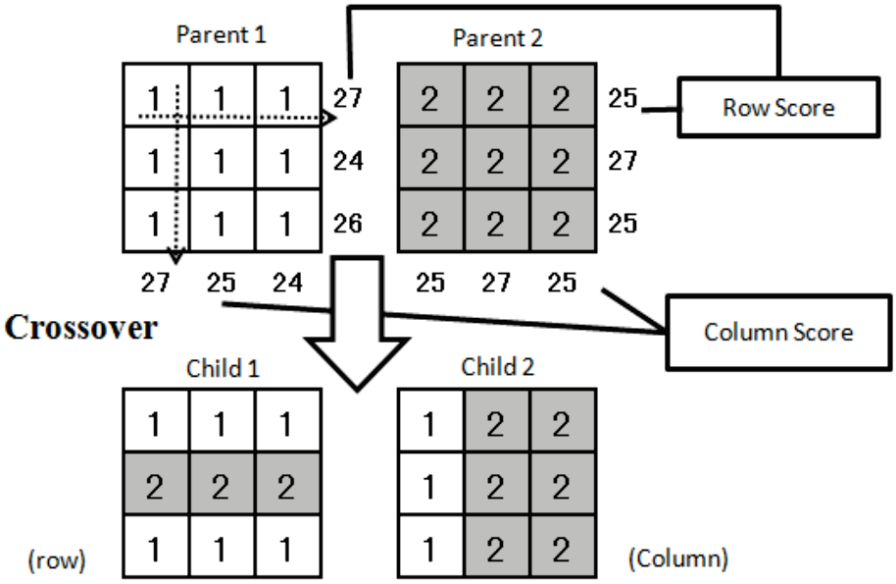
Difficulty rating	1 st	2 st	3 st	4 st	5 st	Ea- sy	Chal leng.	Dif ficu l	Supe r d.
GA, old	100	69	46	26	23	100	30	4	6
GA, new	100	100	100	100	100	100	100	100	100
GA 100 000	100	100	96	63	47	100	60	10	8

2.6.5 ขั้นตอนวิธีเชิงพันธุกรรมแบบที่ 3 กับปัญหาซูโดะกุ จากงานวิจัย Solving Sudoku with Genetic Operations that Preserve Building Blocks [20] เป็นงานวิจัยที่ยังคงเลือกใช้ขั้นตอนวิธีเชิงพันธุกรรมในการแก้ปัญหาซูโดะกุเช่นกัน เสนอโดย Yuji Sato และ Hazuku Inoue ซึ่งครั้งนี้จะให้ความสนใจเกี่ยวกับปัญหาหน่วยโครงสร้าง (Building Blocks) ที่นำไปสู่ผลเฉลยเหมาะสุดถูกทำลาย เนื่องจากการเลือกใช้วิธีการไขว้เปลี่ยนแบบดั้งเดิม และมีการเสนอแนวคิดในการปรับปรุงขั้นตอนวิธีเชิงพันธุกรรมเพิ่มเติม ได้แก่ การปรับปรุงการเข้ารหัสประชากร การปรับปรุงวิธีการไขว้เปลี่ยน รวมถึงการเลือกใช้การค้นเฉพาะที่เพื่อเพิ่มประสิทธิภาพการทำงาน เป็นต้น และในส่วนของโจทย์ปัญหาซูโดะกุทางผู้วิจัยได้กล่าวอีกว่าตัวแปรที่ส่งผลต่อความยากของโจทย์ปัญหาซูโดะกุ ประกอบด้วย จำนวนตัวเลขที่กำหนดไว้ในตารางเริ่มต้น และ วิธีจัดวางตัวเลขดังกล่าว ตัวอย่างเช่น กรณีของปัญหาที่ให้ค่าเริ่มต้นมาน้อยแปลว่ามีจำนวนตัวเลขที่ต้องจัดหรือสลับมากขึ้น ทำให้พื้นที่ในการค้นหาผลเฉลยเหมาะสุดมีมากขึ้นเนื่องจาก ผลเฉลยที่เป็นไปได้มีหลากหลายมากขึ้น เป็นต้น

จากการปรับปรุงกระบวนการทำงานของขั้นตอนวิธีเชิงพันธุกรรม ลักษณะของแต่ละผลเฉลยในประชากรจะแทนด้วยตารางซูโดะกุทั้งตาราง กล่าวคือ จะมีลักษณะเป็นตารางสองมิติขนาด 9×9 โดยกระบวนการทำงานในขั้นแรกจะสร้างผลเฉลยที่เป็นไปได้จากการสร้างตารางใหม่ที่กำหนดค่าเริ่มต้นตามโจทย์ลงไปภายในตาราง พร้อมกับใส่เลขจำนวนเต็มจากการสุ่มภายในช่วงตั้งแต่ 1 ถึง 9 ลงในช่องว่างที่เหลือของตาราง ซึ่งมีเงื่อนไขว่า ภายในแต่ละตารางย่อยของตารางซูโดะกุ (ซึ่งมีขนาด 3×3) จะมีเลขจำนวนเต็มที่ไม่ซ้ำกัน โดยกระบวนการดังกล่าวจะทำซ้ำติดต่อกันจนครบตามจำนวนประชากรที่กำหนด หลังจากนั้นจึงเข้าสู่การดำเนินการเชิงพันธุกรรม (Genetic Operations) สำหรับการไขว้เปลี่ยน และ การกลายพันธุ์

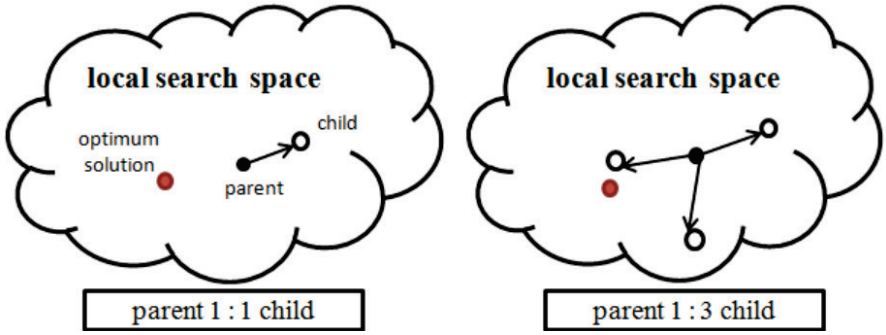
วิธีการไขว้เปลี่ยนจะเริ่มจากการคัดเลือกผลเฉลยพ่อแม่หนึ่งคู่จากประชากร พร้อมกับประเมินคุณภาพ โดยประเมินทีละแถวหรือสดมภ์ ซึ่งค่าความเหมาะสูงสุดในแต่ละแถวหรือสดมภ์ถูกกำหนดให้มีค่าเท่ากับ 9 และ ค่าดังกล่าวจะลดลงตามจำนวนตัวเลขที่หายไป ดังนั้น ผลรวมค่าความเหมาะทีละแถวหรือหลักแต่ละสามตารางย่อยทั้งในแนวตั้งและแนวนอนจะมีค่าสูงสุดเท่ากับ 27 นั่นเอง หลังจากนั้นจึงเริ่มกระบวนการไขว้เปลี่ยนโดยการสร้างผลเฉลยใหม่สองตาราง ซึ่งผลเฉลยแรกสร้างจากการไขว้เปลี่ยนระหว่างพ่อแม่โดยเปรียบเทียบค่าความเหมาะทีละสามตารางย่อยในแนวนอนระหว่างพ่อแม่ทั้งสอง และเลือกสามตารางย่อยดังกล่าวจากพ่อแม่ที่มีค่าความเหมาะโดยรวมมากกว่า ทำเช่นนี้จากบนลงล่างจนครบทั้งตาราง และ ผลเฉลยถัดมาสร้างจากการไขว้เปลี่ยนในลักษณะเดียวกันแต่แตกต่างกันในส่วนของการไขว้เปลี่ยนที่จะกระทำในแนวตั้งและทำจากซ้ายไปขวาจนครบทั้งตาราง ซึ่งสามารถแสดงให้เห็นได้ดังภาพที่ 2-27 [20] ซึ่งประกอบด้วย

ผลเฉลยพ่อแม่ที่ 1 และที่ 2 สำหรับใช้สร้างผลเฉลยลูกหลานที่ 1 (Child 1) และที่ 2 (Child 2) โดยผลเฉลยลูกหลานที่ 1 จะสร้างจากตารางย่อยที่ติดกันสามตารางในแนวนอนของผลเฉลยพ่อแม่ที่ 1 (Parent 1) ผลเฉลยพ่อแม่ที่ 2 (Parent 2) และ ผลเฉลยพ่อแม่ที่ 1 ตามลำดับจากบนลงล่าง ในทำนองเดียวกัน ผลเฉลยลูกหลานที่ 2 จะสร้างจากสามตารางย่อยที่ติดกันในแนวตั้งจากซ้ายไปขวา ของผลเฉลยพ่อแม่ที่ 1 ผลเฉลยพ่อแม่ที่ 2 และ ผลเฉลยพ่อแม่ที่ 2 ตามตามลำดับ



ภาพที่ 2-27 ตัวอย่างการไขว้เปลี่ยนจากแต่ละแถวและหลักในระดับตารางย่อย

ในส่วนของการกลายพันธุ์จะเลือกใช้วิธีสลับค่า โดยสุ่มเลือกตำแหน่งที่ใช้สลับเพียงสองตำแหน่ง โดยตำแหน่งที่เลือกต้องไม่ตรงกับตำแหน่งของค่าเริ่มต้น ทั้งนี้การกลายพันธุ์สามารถเกิดขึ้นได้แค่ภายในแต่ละตารางย่อยเท่านั้น กล่าวคือ ไม่มีการสลับค่าระหว่างสองตารางย่อยนั่นเอง โดยโอกาสการเกิดการกลายพันธุ์จะเป็นไปตามอัตราการกลายพันธุ์ที่กำหนดไว้ และ ในส่วนของวิธีการค้นเฉพาะที่ เริ่มจากการเลือกผลเฉลยที่มีค่าความเหมาะสูงสุดจากประชากร ต่อมาจึงใช้วิธีการกลายพันธุ์เพื่อช่วยในการค้นหาพื้นที่โดยรอบของผลเฉลยดังกล่าว สามารถแสดงรูปแบบการทำงานของการค้นเฉพาะที่ ได้ดังภาพที่ 2-28 [20] และทั้งนี้การเลือกใช้วิธีการคัดเลือกประชากรที่ชื่อว่า tournament selection และ ฟังก์ชันความเหมาะสมได้ออกแบบตามแนวคิดที่ว่าภายในชุดตัวเลขของแต่ละแถวหรือหลักไม่สามารถมีตัวเลขซ้ำกันมากกว่าหนึ่ง โดยคะแนนที่ใช้สำหรับการคำนวณ คือ ค่าความเหมาะที่มาจากจำนวนของตัวเลขต่างๆที่แตกต่างกันภายในแถว (g_i) หรือหลัก (h_j) ซึ่งผลรวมของค่าจากทุกแถวและหลักคือค่าความเหมาะของแต่ละผลเฉลย โดยสมการฟังก์ชันความเหมาะสมสามารถแสดงดังสมการที่ 2-18 ซึ่งสามารถสังเกตจากสมการได้ว่าคะแนนที่คำนวณตลอดทั้งตาราง มีค่าทั้งหมดเท่ากับ 162



ภาพที่ 2-28 แนวคิดการหรับปรุงการค้นหาเฉพาะที่

$$f(x) = \sum_{i=1}^{9} g_i(x) + \sum_{j=1}^{9} h_j(x)$$
$$g_i(x) = |x_i|, h_j(x) = |x_j| \qquad \text{(2-18)}$$

และในส่วนของการทดลองมีการกำหนดค่าดังนี้ คือ ขนาดของประชากรมีค่าเท่ากับ 150 จำนวนพ่อแม่ที่ใช้ให้มีค่าเท่ากับ 2 อัตราการไขว้เปลี่ยนและกลายพันธุ์เท่ากับ 0.3 และขนาดของ Tournament มีค่าเท่ากับ 3 ในส่วนของผลการทดลอง จะแบ่งการทำงานออกเป็นสามส่วนหลัก โดยส่วนแรกจะใช้วิธีการกลายพันธุ์ (Swap mutation) ในการหาคำตอบเท่านั้น ซึ่งมีลักษณะคล้ายกับการค้นหาแบบสุ่ม ในส่วนถัดมาจะเพิ่มการไขว้เปลี่ยนลงในกระบวนการทำงานก่อนหน้า (mut+cross) และสุดท้าย เป็นกระบวนการที่เลือกใช้การค้นเฉพาะที่ การไขว้เปลี่ยน และ การกลายพันธุ์ พร้อมกัน (mut+cross+LS) โดยการทำงานจะเป็นไปอย่างต่อเนื่องจนกว่าจะพบผลเฉลยเหมาะสุด หรือ จำนวนที่ทำซ้ำมีค่าเท่ากับ 100,000 รุ่น ทั้งนี้ผลการทดลองในระดับความยากต่างๆ และ ในแต่ละวิธีที่แตกต่างกัน ได้แสดงให้เห็นว่าการเลือกใช้ทั้งสามวิธีจะทำให้การค้นหาผลเฉลยเหมาะสุดมีประสิทธิภาพมากขึ้น ดังแสดงในตารางที่ 2-9 [20] และ มีผลการทดลองเทียบกับการทดลองก่อนหน้า [19] ดังแสดงในตารางที่ 2-10 [20] ด้วย แม้ว่าขนาดของประชากรจะไม่เท่ากันกับงานวิจัยก่อนหน้า (ขนาดประชากรที่ใช้ในการทดลองก่อนหน้า เท่ากับ 100) แต่ผลที่ได้ค่อนข้างเห็นได้ชัดว่าวิธีการใหม่สามารถค้นหาผลเฉลยเหมาะสุดได้อย่างมีประสิทธิภาพมากขึ้น เมื่อสังเกตจากจำนวนครั้งที่พบผลเฉลยเหมาะสุดในทั้งหมด 100 ครั้ง

ตารางที่ 2-9 ผลการทดลองในระดับความยากที่แตกต่างกันเทียบกับวิธีการที่เลือกใช้ [20]

Difficulty rating	Givens	mut+cross+LS		mut+cross		Swap mutation	
		Count	Average	Count	Average	Count	Average
Easy (No. 1)	38	100	62	100	105	100	223
Easy (No. 11)	34	100	137	100	247	96	6627
Medium (No. 27)	30	100	910	100	2274	86	26961
Medium (No. 29)	29	100	3193	100	6609	66	42141
Difficult (No. 77)	28	100	9482	100	20658	35	77573
Difficult (No. 106)	24	96	26825	74	56428	9	94314

ตารางที่ 2-10 ผลการทดลองเมื่อเทียบกับงานวิจัยก่อนหน้า [20]

Givens	36	33	30	28	25	23	22
Our methods	100	100	100	100	100	100	93
Mantere-2006 [5]	100	100	69	46	30	4	6

2.6.6 ขั้นตอนวิธีประมาณการแจกแจงกับปัญหาซูโดะกุ จากงานวิจัย A Restarted Estimation of Distribution Algorithm for Solving Sudoku Puzzles [39] ซึ่งเมื่อไม่นานนี้ มีการตีพิมพ์งานวิจัยเกี่ยวกับขั้นตอนวิธีเชิงวิวัฒน์บนพื้นฐานของขั้นตอนวิธีประมาณการแจกแจงสำหรับปัญหาซูโดะกุ ชื่อว่า ขั้นตอนวิธีประมาณการแจกแจงแบบเริ่มใหม่ (Restart estimation of distribution algorithm หรือ RESEDA) ซึ่งออกแบบโดย Sylvain Maire และ Cyril Prossette โดยมีแบบจำลองความน่าจะเป็น ซึ่งสามารถแสดงในเทอมของ $p_{i,j,k}^{(n)}$ และ ขั้นตอนวิธีดังกล่าวยังใช้ประโยชน์จากค่าจำนวนเต็มเริ่มต้นที่ให้มาในปัญหาซูโดะกุเพื่อที่จะลดขนาดของปริภูมิค้นหา โดยการปิดการค้นหาในบางช่องของแบบจำลองความน่าจะเป็นเพื่อไม่ให้เลขจำนวนเต็มใดๆที่ให้มาตอนเริ่มต้นมีการเปลี่ยนค่าได้ ซึ่งฟังก์ชันการปรับให้เป็นปัจจุบัน สามารถแสดงได้ดังสมการที่ 2-19 โดยมีเงื่อนไขที่ว่า $0 \le \alpha \le 1$ และ $r_{i,j,k}^{(n+1)}$ แทนการกระจายความน่าจะเป็นเชิงประจักษ์ (Empirical probability distribution) ของ จำนวนที่ k ซึ่งเป็นแต่ละส่วนของผลเฉลยในประชากรนั่นเอง และ α แทนอัตราการเรียนรู้ของฟังก์ชันการปรับให้เป็นปัจจุบันนั่นเอง ซึ่งถ้าหากว่าตัวแปรดังกล่าวมีค่ามากจะทำให้มีความเป็นไปได้ที่จะพบผลเฉลยที่เหมาะที่สุดท้องถิ่น ที่ไม่สามารถไปสู่ผลเฉลยที่เหมาะที่สุดได้ ซึ่งทำให้ขั้นตอนวิธีนี้จำเป็นต้องใช้การเริ่มการทำงานใหม่เพื่อที่จะให้ขั้นตอนวิธีดังกล่าวสามารถค้นหาผลเฉลยในเส้นทางอื่นได้นั่นเอง อย่างไรก็ตามขั้นตอนวิธีนี้มีความจำเป็นที่จะต้องเรียนปัญหาเพื่อเลือกตำแหน่งสมาชิกที่จะทำการเริ่มต้นการค้นหาใหม่ ซึ่งทำให้ไม่สามารถ

เปรียบเทียบกับขั้นตอนวิธีเชิงวิวัฒน์ด้วยกันได้ เพราะเงื่อนไขโดยพื้นฐานของกลุ่มขั้นตอนวิธีเชิงวิวัฒน์ในการทดลอง คือ แต่ละรอบการทำงานจะไม่ขึ้นต่อกัน กล่าวคือ จะไม่มีการเรียนรู้จากการค้นหาในรอบก่อนหน้านั่นเอง

$$p_{i,j,k}^{(n+1)} = \alpha p_{i,j,k}^{(n)} + (1-\alpha) r_{i,j,k}^{(n+1)} \quad \text{(2-19)}$$

บทที่ 3

วิธีดำเนินการวิจัย

ในบทนี้จะกล่าวถึงการดำเนินงานวิจัยซึ่งประกอบด้วย การทดสอบประสิทธิภาพของขั้นตอนวิธีการบรรจวบเพิ่มเติมสำหรับปัญหาเชิงการจัด และ ปัญหาหลากหลายรูปแบบ รวมถึงการประยุกต์แนวคิดต่างๆที่น่าสนใจจากทฤษฎีและงานวิจัยก่อนหน้าเพื่อนำมาปรับใช้ในขั้นตอนวิธีการบรรจวบเพื่อให้สามารถแก้ปัญหาซูโดะกุได้อย่างมีประสิทธิภาพ โดยมีแผนภาพแสดงการดำเนินงานทั้งหมดของงานวิจัย ซึ่งสามารถแสดงได้ดังภาพที่ 3-1 โดยประกอบด้วย 2 ส่วนหลักในการดำเนินการดังนี้

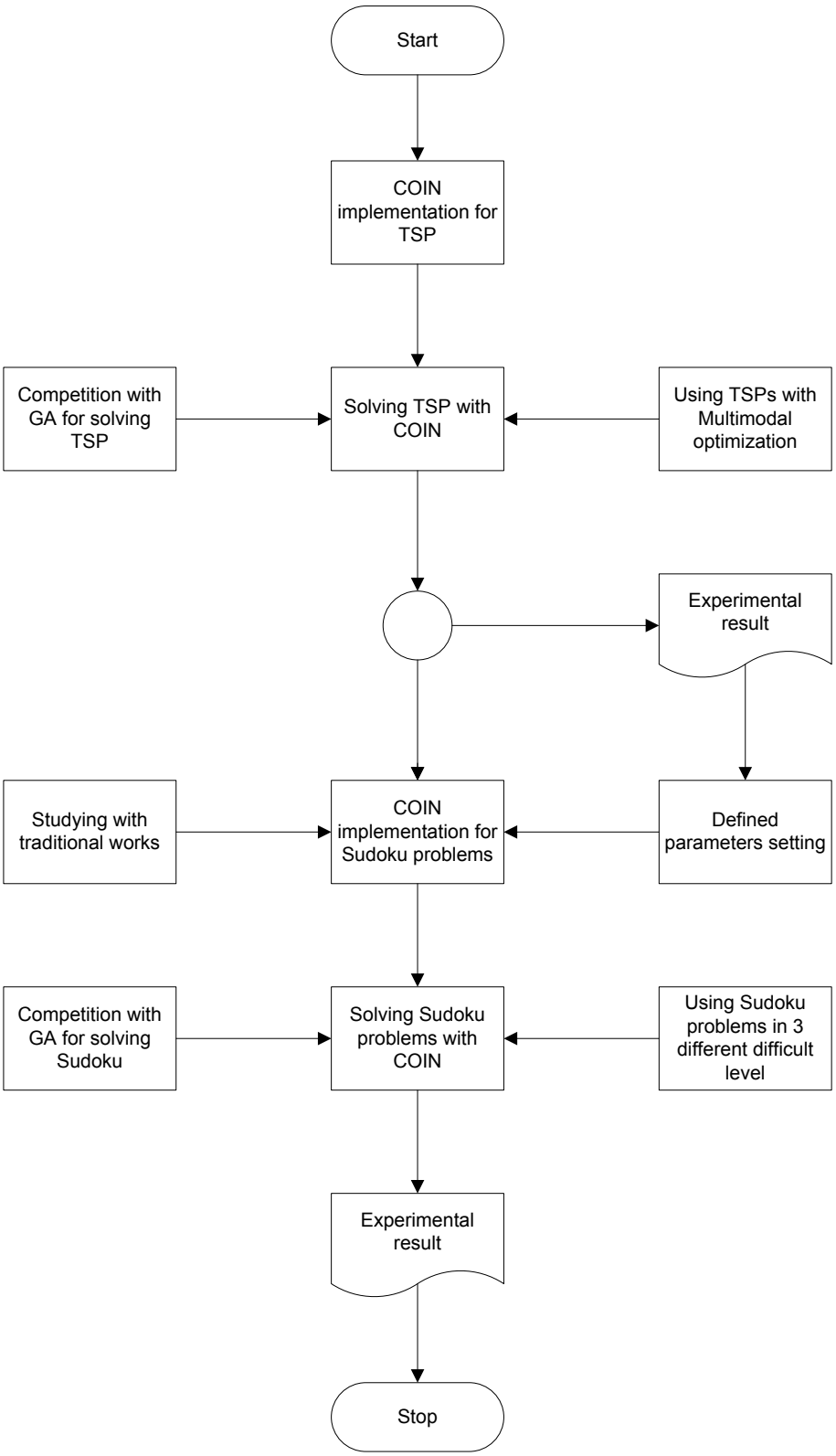
3.1 ภาพรวมทั้งหมดของงานวิจัย

การศึกษาประสิทธิภาพการทำงานของขั้นตอนวิธีการบรรจวบ [39] สามารถจำแนกออกเป็นแต่ละส่วนได้ดังนี้

1. ปัญหาที่ใช้ทดสอบประสิทธิภาพ คือ ปัญหาการเดินทางของพนักงานขาย
2. ขั้นตอนวิธีที่ใช้ในการเปรียบเทียบประสิทธิภาพ คือ ขั้นตอนวิธีเชิงพันธุกรรม
3. ปัญหาที่ใช้ทดสอบเป็นปัญหาที่อยู่ทั้งในกลุ่มปัญหาเชิงการจัด และ กลุ่มปัญหาหลากหลายรูปแบบ

การแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบสามารถจำแนกออกเป็นแต่ละส่วนได้ดังนี้

1. ออกแบบกระบวนการทำงานของขั้นตอนวิธีการบรรจวบให้สามารถแก้ปัญหาซูโดะกุได้อย่างถูกต้อง และ มีประสิทธิภาพ
2. มีการทดสอบประสิทธิภาพของผลลัพธ์โดยอ้างอิงจากขั้นตอนวิธีเชิงพันธุกรรม
3. ตัวอย่างปัญหาที่ใช้เป็นปัญหาชุดเดียวกับที่ใช้ทดสอบประสิทธิภาพขั้นตอนวิธีเชิงพันธุกรรม ซึ่งมีหลายระดับความยาก



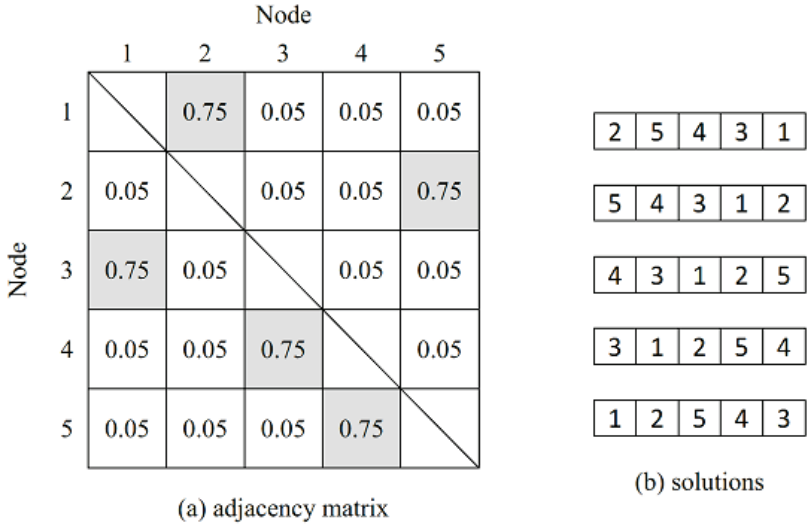
ภาพที่ 3-1 ผังงานแสดงโครงสร้างโดยรวมของงานวิจัย

3.2 การศึกษาประสิทธิภาพการทำงานของขั้นตอนวิธีการบรรจวบ

ในงานวิจัยนี้ มีจุดประสงค์ในการแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบเป็นหลัก ซึ่งการดำเนินการเบื้องต้น เริ่มจากการพัฒนาขั้นตอนวิธีการบรรจวบสำหรับแก้ปัญหาการเดินทางของพนักงานขาย โดยมุ่งเน้นศึกษาคุณสมบัติที่น่าสนใจของขั้นตอนวิธีการบรรจวบอย่างการหาค่าเหมาะที่สุดหลากหลายรูปแบบ ซึ่งปัญหาที่ใช้ทดสอบ คือ ปัญหาการเดินทางของพนักงานขายซึ่งถูกออกแบบให้มีคุณสมบัติของปัญหาหลากหลายรูปแบบ และในส่วนของกระบวนการหาผลเฉลยยังคงเป็นไปตามรูปแบบดั้งเดิมของงานวิจัยก่อนหน้า โดยมีการเพิ่มเติมกระบวนการบางส่วนเพื่อใช้นับจำนวนของผลเฉลยที่เหมาะที่สุดที่มีลักษณะแตกต่างกัน นั่นเอง

3.3 การแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบ

ในงานวิจัยนี้ ได้ออกแบบกระบวนการค้นหาผลเฉลย โดยอาศัยแนวคิดต่างๆจากงานวิจัยก่อนหน้า โดยส่วนแรกที่ได้รับการปรับปรุง คือ แบบจำลองความน่าจะเป็น ซึ่งจากเดิมอยู่ในรูปของเมทริกซ์ความน่าจะเป็นร่วม ซึ่งแต่ละสมาชิกภายในเมทริกซ์แทนความสัมพันธ์เชิงเส้นระหว่างบัพ ซึ่งเมื่อนำมาพิจารณากับปัญหาซูโดะกุพบว่ากระบวนการเรียนรู้ของเมทริกซ์ดังกล่าวมีข้อจำกัดอยู่อย่างหนึ่ง ซึ่งสามารถแสดงได้ดังภาพที่ 3-2



ภาพที่ 3-2 ตัวอย่างของแต่ละผลเฉลย (b) ที่สร้างโดยแบบจำลองความน่าจะเป็น (a)

สมมติว่ามีการนำขั้นตอนวิธีการบรรจวบแบบดั้งเดิมที่ใช้เมทริกซ์ความน่าจะเป็นร่วมมาแก้ปัญหาการจัดตารางการผลิตสำหรับระบบการผลิตแบบไหลปัญหาหนึ่ง โดยอ้างอิงตาม ภาพที่ 3-2 ซึ่งขนาดของปัญหาเท่ากับ 5 และ สมมติให้ผลเฉลยที่เหมาะที่สุดของปัญหา คือ {1, 2, 5, 4,

3} รวมถึง สมมติให้เมทริกซ์ดังกล่าวได้มีการเรียนรู้ในระดับหนึ่งแล้วว่าลำดับงานที่เหมาะที่สุดมีลักษณะตามที่กล่าวไว้ เมื่อนำเมทริกซ์ดังกล่าวมาสร้างแต่ละผลเฉลยที่เป็นไปได้พบว่า ผลเฉลยที่เหมาะที่สุดตามที่เมทริกซ์ดังกล่าวเข้าใจนั้นมีผลเฉลยอื่นๆอีก คือ {2, 5, 4, 3, 1}, {5, 4, 3, 1, 2}, {4, 3, 1, 2, 5}, และ {3, 1, 2, 5, 4} ซึ่งเป็นความเข้าใจที่ผิดของเมทริกซ์ที่สร้างผลเฉลยที่ไม่ตรงกับผลเฉลยที่เหมาะที่สุดตามที่กล่าวไว้ โดยมีลักษณะเลื่อนเป็นวงวน อย่างไรก็ตาม ปัญหาดังกล่าวยังเกิดกับขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบเส้นเชื่อมด้วยเช่นกัน โดยขั้นตอนวิธีดังกล่าวได้รับการปรับปรุงแก้ไข และ กลายเป็นขั้นตอนวิธีใหม่ ชื่อว่าขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบบัพ ดังนั้น จึงมีความเป็นไปได้ที่จะปรับปรุงเมทริกซ์ความน่าจะเป็นร่วมเพื่อให้สามารถเรียนรู้และ สร้างผลเฉลยที่เหมาะที่สุดได้อย่างถูกต้องมากขึ้น

Node

Position	1	2	3	4	5
1	0.15	0.4	0.15	0.15	0.15
2	0.4	0.15	0.15	0.15	0.15
3	0.15	0.15	0.15	0.4	0.15
4	0.15	0.15	0.15	0.15	0.4
5	0.15	0.15	0.4	0.15	0.15

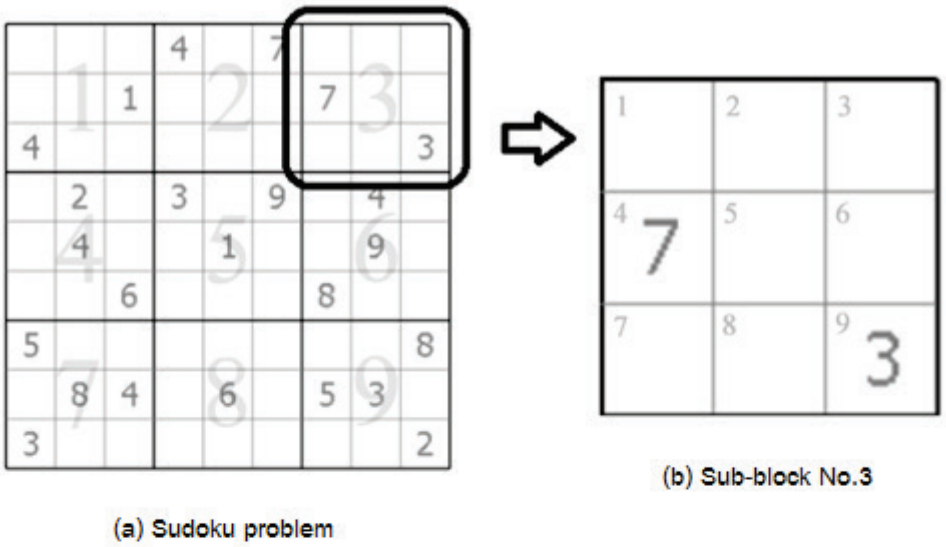
Position	1	2	3	4	5
Node	2	1	4	5	3

ภาพที่ 3-3 แบบจำลองความน่าจะเป็นของขั้นตอนวิธีการบรรจวบแบบบัพที่มีขนาดของปัญหาเท่ากับ 5 และ ผลเฉลยที่เหมาะที่สุดที่สร้างจากแบบจำลองดังกล่าว

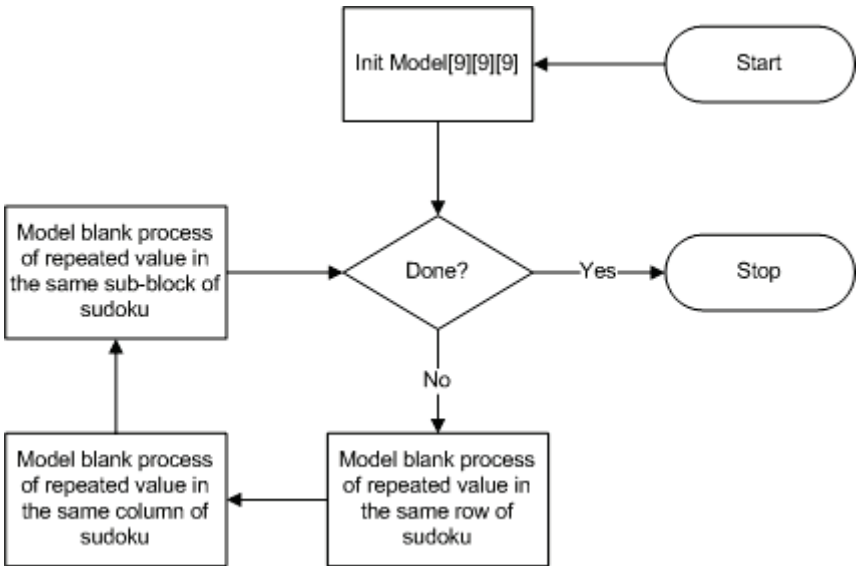
จากแนวคิดข้างต้น จึงมีการปรับปรุงขั้นตอนวิธีการบรรจวบในส่วนเมทริกซ์ความน่าจะเป็นร่วม เพื่อให้กลายเป็นแบบจำลองที่อธิบายความสัมพันธ์ระหว่างบัพและตำแหน่ง เรียกว่าขั้นตอนวิธีการบรรจวบแบบบัพ (Node based coincidence algorithm หรือ NB-COIN) ซึ่งเป็นแบบจำลองที่อธิบายการกระจายตัวของแต่ละบัพตามตำแหน่งต่างๆของผลเฉลยใดๆ เมื่อแบบจำลองความน่าจะเป็นมีการเรียนรู้ในระดับที่เพียงพอ และ นำแบบจำลองดังกล่าวมาสร้างแต่

ละผลเฉลยที่เป็นไปได้ จะพบว่าผลเฉลยที่เหมาะที่สุดตามที่แบบจำลองความน่าจะเป็นเข้าใจยังคงมีเพียงผลเฉลยเดียวเท่านั้น

ในส่วนของการประยุกต์ใช้ขั้นตอนวิธีการบรรจวบแบบบัพสำหรับแก้ปัญหาซูโดะกุนั้น เนื่องจากซูโดะกุขนาด 9×9 เป็นที่นิยมอย่างแพร่หลาย จึงใช้ขนาดดังกล่าวในการอ้างอิงสำหรับการออกแบบ โดยเริ่มจากการพิจารณาแต่ละตารางย่อยที่มีจำนวนสมาชิกทั้งหมดเท่ากับ 9 จัดเรียงอยู่ในลักษณะของตารางขนาด 3×3 อยู่ภายในตารางซูโดะกุ พบว่า สามารถใช้แบบจำลองความน่าจะเป็นสำหรับขนาดของปัญหาเท่ากับ 9 เพื่ออธิบายผลเฉลยในแต่ละตารางย่อยได้ ดังนั้น ปัญหาซูโดะกุสำหรับขนาด 9×9 จึงมีทั้งหมด 9 แบบจำลองความน่าจะเป็นด้วยกัน เพื่อใช้เรียนรู้แต่ละตารางย่อยที่มีจำนวนทั้งหมดเท่ากับ 9 นั่นเอง และ นอกจากนี้ เมื่ออ้างอิงจากการแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการประมาณการแจกแจงแบบเริ่มใหม่ของงานวิจัยที่ผ่านมา พบว่า แบบจำลองความน่าจะเป็นสำหรับปัญหาซูโดะกุสามารถที่จะยกเลิกการค้นหาในบางสมาชิกที่ไม่จำเป็นเพื่อที่จะลดขนาดของปริภูมิการค้นหาได้ โดยเหตุผลที่ว่าสมาชิกดังกล่าวอธิบายผลเฉลยที่ให้มาในตารางซูโดะกุตอนเริ่มต้นอยู่แล้ว หรือ สมาชิกดังกล่าวอธิบายผลเฉลยที่ห้ามวางลงภายในตารางซูโดะกุเนื่องจากผิดกฎ เป็นต้น ผลลัพธ์ของการประยุกต์การทำงานในส่วนของแบบจำลองสามารถแสดงตัวอย่างบางส่วนได้ดังภาพที่ 3-4 ซึ่งเป็นตัวอย่างของแบบจำลองความน่าจะเป็นหมายเลข 3 ที่มีหน้าที่เรียนรู้ผลเฉลยของตารางย่อย โดยค่าเริ่มต้นภายในตารางย่อยมีค่าเท่ากับ 7 และ 3 ในตำแหน่งที่ 4 และ 9 ตามลำดับ รวมถึงมีการลดขนาดของปริภูมิค้นหาในส่วนของแบบจำลองความน่าจะเป็นหมายเลข 3 โดยการพิจารณาตามกฎของปัญหาที่ว่าห้ามมีเลขที่ซ้ำกันภายในแถว และ ภายในหลักเดียวกัน รวมถึง ภายในตารางย่อยเดียวกันอีกด้วยตามภาพที่ 3-5 ซึ่งเป็นผังงานที่แสดงถึงกระบวนการทำงานโดยย่อในส่วนของการลดขนาดของปริภูมิค้นหา



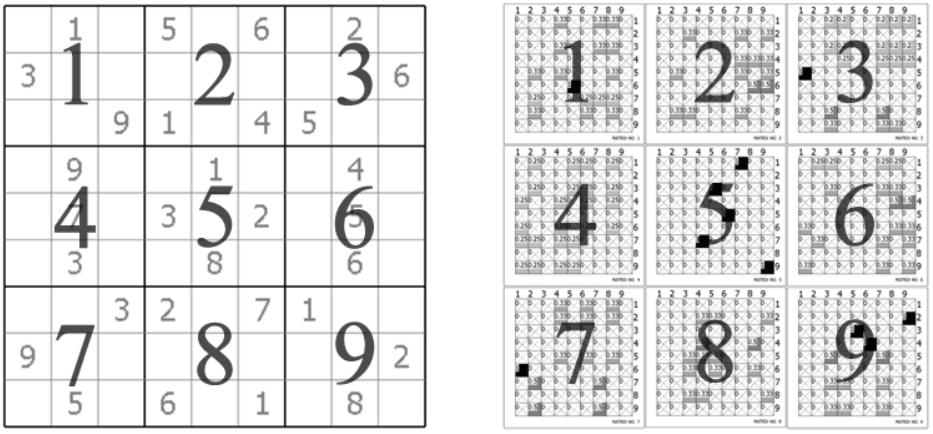
ภาพที่ 3-4 ลักษณะของการแทนแบบจำลองความน่าจะเป็นสำหรับปัญหาซูโดะกุ



ภาพที่ 3-5 ผังงานแสดงขั้นตอนการยกเลิกการค้นหาบางสมาชิกของแบบจำลองความน่าจะเป็น

กระบวนการสร้างผลเฉลยจากแบบจำลองเริ่มจากการสร้างทีละตารางย่อยจนครบ และนำมาประกอบกันเป็นหนึ่งตารางซูโดะกุ ตัวอย่างเช่น เมื่อปัญหาซูโดะกุมีขนาดเท่ากับ 9×9 ประกอบด้วยตารางย่อยทั้งหมด 9 ตาราง ซึ่งภายในมีจำนวนสมาชิกหรือจำนวนช่องว่างสำหรับใส่เลขจำนวนเต็มทั้งหมด 9 ช่อง แบบจำลองความน่าจะเป็นที่ใช้สำหรับปัญหานี้มีขนาดเท่ากับ 9×9 พร้อมกับมีลำดับที่ชัดเจนกำกับไว้ทั้งส่วนของแบบจำลองความน่าจะเป็นและตารางย่อย เพื่อให้กระบวนการสร้างผลเฉลย รวมถึงกระบวนการปรับปรุงดำเนินการภายในลำดับหรือคู่ที่ตรงกันตลอดกระบวนการทำงาน ซึ่งความสัมพันธ์ระหว่างตารางย่อยและแบบจำลองความน่าจะเป็นสามารถแสดงดังภาพที่ 3-6

เมื่อนำแบบจำลองดังกล่าวมาใช้สร้างประชากรจำนวนหนึ่ง กระบวนการถัดไปของขั้นตอนวิธี คือ การประเมินคุณภาพของแต่ละผลเฉลยภายในประชากรนั่นเอง โดยการนำแต่ละผลเฉลยที่เป็นไปได้มาคำนวณผ่านฟังก์ชันความเหมาะสม ซึ่งวิธีการคำนวณอ้างอิงตามงานวิจัยก่อนหน้า แสดงได้ดังสมการที่ 2-18 โดยมีแนวคิดการคำนวณมาจากการพิจารณาในแต่ละแถวและสดมภ์ว่ามีเลขไหนหายไปหรือไม่ ซึ่งตามขนาดของปัญหาซูโดะกุที่มีขนาดเท่ากับ 9×9 จะได้ว่า ในแต่ละแถวหรือสดมภ์จะมีค่าความเหมาะเท่ากับจำนวนสมาชิกทั้งหมดของเซต {1, 2, 3, 4, 5, 6, 7, 8, 9} ที่ตรงกับสมาชิกของเลขจำนวนเต็มในแต่ละแถวหรือหลักดังกล่าวพร้อมกับ นำค่าความเหมาะในทุกๆแถวและหลักมามาหาผลรวม โดยผลรวมสำหรับผลเฉลยที่เหมาะที่สุดจะมีค่าความเหมาะเท่ากับ 162



(a) 9 sub-blocks of Sudoku (b) 9 probabilistic models

ภาพที่ 3-6 ความสัมพันธ์ระหว่างตารางย่อยของปัญหาซูโดะกุและแบบจำลองความน่าจะเป็นของขั้นตอนวิธีการบรรจวบแบบบัพ

กระบวนการคัดเลือกผลเฉลยของขั้นตอนวิธีการบรรจวบแบบบัพ ยังคงใช้วิธีการแบบเดียวกับขั้นตอนวิธีดั้งเดิม และเมื่อทำการคัดเลือกแต่ละผลเฉลยที่ต้องการทั้งที่อยู่ในกลุ่มดีและกลุ่มที่ดีน้อยกว่าเรียบร้อยแล้ว ให้นำทั้งสองกลุ่มมาเข้ากระบวนการปรับปรุงแบบจำลองความน่าจะเป็น ซึ่งส่วนนี้เป็นส่วนที่แตกต่างจากกระบวนการแบบเดิม เนื่องจาก แต่ละผลเฉลยที่ได้รับการคัดเลือกจะต้องนำมากระจายออกเป็นแต่ละตารางย่อย พร้อมกับนำไปปรับปรุง แต่ละแบบจำลองความน่าจะเป็นตามลำดับที่ตรงกัน ตัวอย่างเช่น ให้การหาค่าเหมาะที่สุดดังกล่าวของปัญหาซูโดะกุมีขนาดเท่ากับ 9×9 พบว่า ในขั้นตอนของการปรับปรุงตาราง แต่ละผลเฉลยที่ได้รับการคัดเลือกจะต้องกระจายออกเป็นตารางย่อย 9 ตาราง โดยแต่ละตารางจะถูกเลือกเพื่อนำไปปรับปรุง โดยจับคู่แต่ละตารางย่อยกับแบบจำลองความน่าจะเป็นอย่างเป็นลำดับตามหมายเลขที่ตรงกัน โดยมีอัตราการปรับปรุงค่าของแบบจำลองความน่าจะเป็นที่ขึ้นกับค่า k เช่นเดิม และ เมื่อกระบวนการปรับปรุงแบบจำลองความน่าจะเป็นมีการดำเนินการเสร็จสมบูรณ์ กระบวนการถัดไปจะย้อนกลับมาที่ส่วนของการสร้างประชากรอีกครั้ง โดยจำนวนครั้งในการวนซ้ำการทำงานเดิมขึ้นอยู่กับจำนวนรุ่นค้นหามากที่สุดที่กำหนดไว้ตอนเริ่มต้นการทำงาน กล่าวคือ กระบวนการค้นหาผลเฉลยจะสิ้นสุดเมื่อจำนวนรุ่นค้นหาปัจจุบันมีค่ามากกว่าจำนวนรุ่นค้นหามากที่สุดที่กำหนดไว้

3.4 การวัดประสิทธิภาพ

งานวิจัยที่ทดสอบประสิทธิภาพของขั้นตอนวิธีการบรรจวบ ขั้นตอนวิธีเชิงพันธุกรรมถูกใช้ในการเปรียบเทียบ โดยมีการวัดผลในสองด้านด้วยกัน คือ จำนวนรุ่นที่ใช้ค้นหาผลเฉลยที่เหมาะที่สุด และ จำนวนที่พบผลเฉลยที่เหมาะที่สุดที่แตกต่างกัน โดยเฉลี่ย ซึ่งมีการกำหนดค่าเริ่มต้นสำหรับทั้งสองขั้นตอนวิธีที่ตรงกันเท่าที่จะเป็นไปได้เพื่อความเท่าเทียมกัน และ ในการวัดประสิทธิภาพการแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบแบบบัพ มีการเทียบกับขั้นตอนวิธีเชิงวิวัฒน์ โดยเทียบกับผลการทดลองของขั้นตอนวิธีเชิงพันธุกรรมที่มีการทดลองไว้แล้วในงานวิจัยก่อนหน้า ซึ่งมีการเทียบประสิทธิภาพในสองส่วน คือ จำนวนครั้งที่พบผลเฉลยที่เหมาะที่สุดในการทำงานร้อยครั้งที่ไม่ขึ้นต่อกันในแต่ละรอบการค้นหา และ จำนวนรุ่นที่ใช้ค้นหาผลเฉลยที่เหมาะที่สุดโดยเฉลี่ย ภายใต้การกำหนดค่าเริ่มต้นที่ตรงกันเท่าที่จะเป็นไปได้ เนื่องจากเป็นการเทียบผลการทดลองกับการทดลองในงานวิจัยก่อนหน้า ปัญหาที่ใช้ในการทดสอบจึงเป็นปัญหาเดียวกัน โดยมีทั้งหมด 3 ระดับความยาก คือ ง่าย ปานกลาง และ ยาก นั่นเอง

บทที่ 4

การทดลองและผลการทดลอง

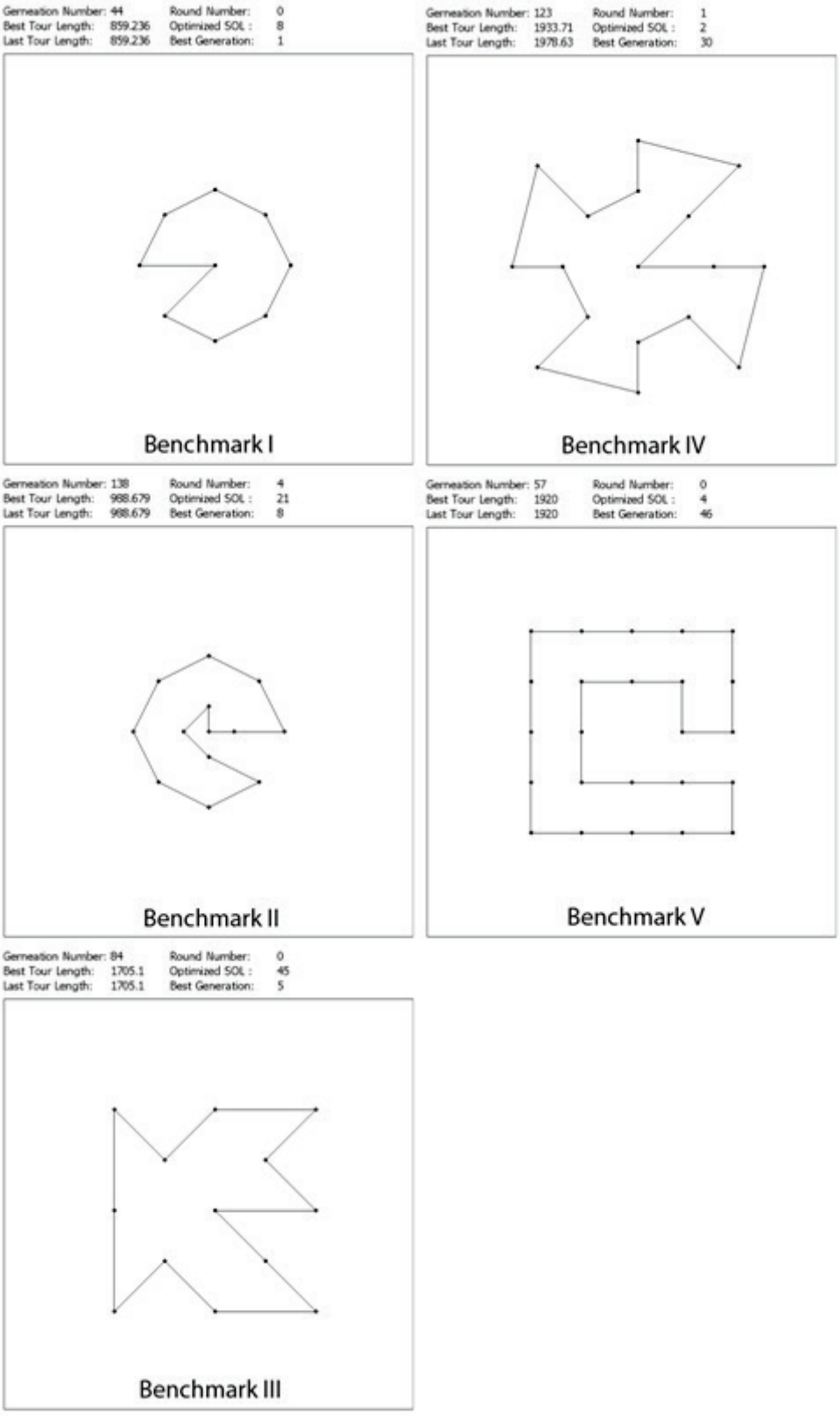
บทนี้กล่าวถึงผลการทดลองต่างๆทั้งในส่วนของการทดสอบประสิทธิภาพของขั้นตอนวิธีการบรรจวบทางด้านการหาค่าเหมาะที่สุดหลากหลายรูปแบบ และ การหาค่าเหมาะที่สุดเชิงการจัดอย่างปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบแบบบัพ รวมถึงองค์ประกอบต่างๆที่ใช้ในการทดลอง เช่น เครื่องมือที่ใช้สร้างการทดลอง เป็นต้น การเปรียบเทียบประสิทธิภาพอาศัยการเปรียบเทียบในแต่ละผลการทดลองโดยขั้นตอนวิธีเชิงพันธุกรรม ซึ่งเป็นขั้นตอนวิธีหนึ่งในขั้นตอนวิธีเชิงวิวัฒน์ที่ได้รับความนิยมในการแก้ปัญหาที่หลากหลายจนถึงปัจจุบัน

4.1 เครื่องมือที่ใช้ในการทดลอง

ซอฟต์แวร์ที่ใช้ในการสร้างการทดลองในงานวิจัยนี้ คือ Visual Studio เวอร์ชัน 2010 และพัฒนาขั้นตอนวิธีการบรรจวบทั้งแบบปกติและแบบบัพโดยภาษา C++

ฮาร์ดแวร์ที่ใช้ในการทดสอบคือ Lenovo Thinkpad รุ่น T410 รหัสผลิตภัณฑ์ 25379FT ซึ่งประกอบด้วย ซีพียู Intel® Core™ i5-540M Processor (3M Cache, 2.53 GHz) หน่วยความจำหลัก 4096 MB ฮาร์ดดิสก์ 500 GB บนระบบปฏิบัติการ 32-bit Microsoft Windows 7

ปัญหาที่ใช้ในส่วนของการทดสอบประสิทธิภาพ เป็นปัญหาการเดินทางของพนักงานขาย ซึ่งมีจำนวนปัญหาทั้งหมดเท่ากับ 5 และแต่ละปัญหาสามารถพบผลเฉลยที่เหมาะที่สุดได้มากกว่าหนึ่ง แสดงได้ตามภาพที่ 4-1 โดยปัญหาที่หนึ่ง (Benchmark I) มีขนาดเท่ากับ 9 และมีจำนวนผลเฉลยที่เหมาะที่สุดเท่ากับ 8 ปัญหาที่สอง (Benchmark II) และ ปัญหาที่สาม (Benchmark III) มีขนาดเท่ากับ 13 และ มีจำนวนผลเฉลยที่เหมาะที่สุดเท่ากับ 24 และ 96 ตามลำดับ ปัญหาที่สี่ (Benchmark IV) มีขนาดเท่ากับ 17 และ มีจำนวนผลเฉลยที่เหมาะที่สุดเท่ากับ 8 และ ปัญหาที่ห้า (Benchmark V) มีขนาดเท่ากับ 24 และ มีจำนวนผลเฉลยที่เหมาะที่สุดเท่ากับ 10



ภาพที่ 4-1 ปัญหา TSPs สำหรับการหาค่าเหมาะที่สุดหลากหลายรูปแบบ

ปัญหาซูโดะกุที่ใช้ในการทดลองมีจำนวนทั้งหมด 6 ปัญหา ซึ่งแต่ละปัญหาเป็นปัญหาที่นำมาจากงานวิจัยก่อนหน้า โดยสามารถแบ่งตามระดับความยากได้สามระดับ ซึ่งแต่ละระดับจะมีจำนวนปัญหาเท่ากับ 2 และ สามารถแสดงปัญหาระดับง่ายดังภาพที่ 4-2 โดยกลุ่มปัญหาระดับดังกล่าวประกอบด้วยปัญหาหมายเลข 1 และ 11 มีจำนวนของเลขจำนวนเต็มเริ่มต้นเท่ากับ 38 และ 34 ตามลำดับ

No. 1 (Easy level)

		9				1		
2	1	7				3	6	8
			2		7			
	6	4	1		3	5	8	
	7						3	
1	5		4	2	8		7	9
			5	8	9			
4	8	5				2	9	3
		6	3		2	8		

(a) The initial Sudoku problem (38 Givens)

No. 1 (Easy level)

5	4	9	8	3	6	1	2	7
2	1	7	9	5	4	3	6	8
6	3	8	2	1	7	9	5	4
9	6	4	1	7	3	5	8	2
8	7	2	6	9	5	4	3	1
1	5	3	4	2	8	6	7	9
3	2	1	5	8	9	7	4	6
4	8	5	7	6	1	2	9	3
7	9	6	3	4	2	8	1	5

(b) The result of the problem (38 Givens)

No. 11 (Easy level)

2	9		7		1			
5	3			6		1		
		6	3				4	
			5	9				4
	1	5			4	6	8	9
			1	8				3
		2	6				9	
3	6			4		7		
9	4		8		5			

(c) The initial Sudoku problem (34 Givens)

No. 11 (Easy level)

2	9	4	7	5	1	8	3	6
5	3	8	4	6	9	1	2	7
1	7	6	3	2	8	9	4	5
6	8	3	5	9	7	2	1	4
7	1	5	2	3	4	6	8	9
4	2	9	1	8	6	5	7	3
8	5	2	6	7	3	4	9	1
3	6	1	9	4	2	7	5	8
9	4	7	8	1	5	3	6	2

(d) The result of the problem (34 Givens)

ภาพที่ 4-2 ปัญหาซูโดะกุระดับง่ายที่ใช้ในการทดลอง

ปัญหาระดับปานกลางสามารถแสดงได้ดังภาพที่ 4-3 โดยกลุ่มปัญหาดังกล่าวประกอบด้วยปัญหาหมายเลข 27 และ 29 ที่มีจำนวนของเลขจำนวนเต็มเริ่มต้นเท่ากับ 30 และ 29 ตามลำดับ

No. 27 (Medium level)

	1		5		6		2	
3								6
		9	1		4	5		
	9			1			4	
	7		3		2		5	
	3			8			6	
		3	2		7	1		
9								2
	5		6		1		8	

(e) The initial Sudoku problem (30 Givens)

No. 27 (Medium level)

4	1	8	5	3	6	9	2	7
3	2	5	9	7	8	4	1	6
7	6	9	1	2	4	5	3	8
8	9	6	7	1	5	2	4	3
1	7	4	3	6	2	8	5	9
5	3	2	4	8	9	7	6	1
6	8	3	2	4	7	1	9	5
9	4	1	8	5	3	6	7	2
2	5	7	6	9	1	3	8	4

(f) The result of the problem (30 Givens)

No. 29 (Medium level)

		1		8				
			3		4	7	5	
	6			5				
8		6			2	3	4	9
		9						
3		4			8	1	7	2
	3			7				
			8		1	5	6	
		2		3				

(g) The initial Sudoku problem (29 Givens)

No. 29 (Medium level)

5	4	1	2	8	7	9	3	6
9	2	8	3	6	4	7	5	1
7	6	3	1	5	9	8	2	4
8	7	6	5	1	2	3	4	9
2	1	9	7	4	3	6	8	5
3	5	4	6	9	8	1	7	2
1	3	5	4	7	6	2	9	8
4	9	7	8	2	1	5	6	3
6	8	2	9	3	5	4	1	7

(h) The result of the problem (29 Givens)

ภาพที่ 4-3 ปัญหาซูโดะกุระดับปานกลางที่ใช้ในการทดลอง

ปัญหาในกลุ่มสุดท้ายนี้เป็นปัญหาระดับยาก ซึ่งสามารถแสดงได้ดังภาพที่ 4-4 ประกอบด้วย ปัญหาหมายเลข 77 และ 106 ซึ่งมีจำนวนของเลขจำนวนเต็มเริ่มต้นเท่ากับ 28 และ 24 ตามลำดับ

No. 77 (Difficult level)

5								9
9			8		5			6
3			9		7			5
				9				
	9			1			2	
	3	8				9	4	
4								2
		3	5		9	6		
		2	4		1	3		

(i) The initial Sudoku problem (28 Givens)

No. 77 (Difficult level)

5	8	7	1	6	2	4	3	9
9	2	4	8	3	5	1	7	6
3	6	1	9	4	7	2	8	5
1	4	5	2	9	8	7	6	3
7	9	6	3	1	4	5	2	8
2	3	8	7	5	6	9	4	1
4	1	9	6	7	3	8	5	2
8	7	3	5	2	9	6	1	4
6	5	2	4	8	1	3	9	7

(j) The result of the problem (28 Givens)

No. 106 (Difficult level)

			4		7			
		1				7		
4								3
	2		3		9		4	
	4			1			9	
		6				8		
5								8
	8	4		6		5	3	
3								2

(k) The initial Sudoku problem (24 Givens)

No. 106 (Difficult level)

8	5	3	4	9	7	2	1	6
6	9	1	2	3	8	7	5	4
4	7	2	1	5	6	9	8	3
1	2	5	3	8	9	6	4	7
7	4	8	6	1	2	3	9	5
9	3	6	5	7	4	8	2	1
5	1	7	9	2	3	4	6	8
2	8	4	7	6	1	5	3	9
3	6	9	8	4	5	1	7	2

(l) The result of the problem (24 Givens)

ภาพที่ 4-4 ปัญหาซูโดะกุระดับยากที่ใช้ในการทดลอง

4.2 ผลการทดลอง

งานวิจัยในส่วนแรก เป็นการทดลองที่เกี่ยวกับการนำขั้นตอนวิธีการบรรจวบแบบปกติมาแก้ปัญหาการเดินทางของพนักงานขายที่มีจำนวนทั้งหมด 5 ปัญหา ซึ่งจัดอยู่ในกลุ่มปัญหาการหาค่าเหมาะที่สุดหลากหลายรูปแบบ และ มีการเลือกใช้ขั้นตอนวิธีเชิงพันธุกรรมที่มีการพัฒนาสำหรับแก้ปัญหาดังกล่าวไว้เรียบร้อยแล้วซึ่งมีการพัฒนาภายใต้ภาษา C# โดย Michael LaLena [40] และในส่วนของการกำหนดค่าเริ่มต้นแต่ละค่าของการทดลอง จำนวนประชากร และ จำนวนรุ่นค้นหามากที่สุด ของทั้งสองขั้นตอนวิธีจะมีค่าเท่ากับ 400 และ 10,000,000 ตามลำดับ และในส่วนอัตราการกลายพันธุ์ (Mutation rate) ของขั้นตอนวิธีเชิงพันธุกรรมมีค่าเท่ากับ 10% และในส่วนของขั้นตอนวิธีการบรรจวบ ขนาดของการคัดเลือกผลเฉลย (Selection pressure) และ ขนาดการปรับค่า (k) มีค่าเท่ากับ 5% และ 0.1 ตามลำดับ ดังแสดงในตารางที่ 4-1 โดย ตารางดังกล่าวเป็นผลการทดลองโดยเฉลี่ยจากการค้นหาผลเฉลยซ้ำ 10 ครั้ง และ PROBLEM แทนลำดับของปัญหา PSIZE แทนขนาดของปัญหา OPSOL แทนจำนวนผลเฉลยที่เหมาะที่สุดทั้งหมดที่มีลักษณะแตกต่างกัน AVG GEN แทนจำนวนของรุ่นโดยเฉลี่ยที่พบผลเฉลยที่เหมาะที่สุด และ AVG SOL แทนจำนวนของผลเฉลยที่เหมาะที่สุดที่แตกต่างกันของแต่ละขั้นตอนวิธี และจากผลการทดลองพบว่าจำนวนของผลเฉลยที่เหมาะที่สุดที่ต่างกันที่พบของขั้นตอนวิธีการบรรจวบมีจำนวนมากกว่าในขั้นตอนวิธีเชิงพันธุกรรมในปัญหาที่ 2 ถึง 5 และ จำนวนของรุ่นสำหรับค้นหาผลเฉลยของขั้นตอนวิธีการบรรจวบยังคงมีจำนวนที่ใช้น้อยกว่าของขั้นตอนวิธีเชิงพันธุกรรม เช่นกัน โดยเฉพาะในปัญหาที่ 4 และ 5 ที่มีความแตกต่างกันของจำนวนรุ่นที่ใช้อย่างชัดเจน

ตารางที่ 4-1 ผลการทดลองการหาค่าเหมาะที่สุดหลากหลายรูปแบบ

PROBLEM	PSIZE	OPSOL	AVG GEN		AVG SOL	
			GA	COIN	GA	COIN
I	9	8	55	2	8	8
II	13	24	440	3	5	24
III	13	96	430	5	76	96
IV	17	8	496642	39	4	8
V	24	10	968397	46	1	8

การทดลองในส่วนถัดมาเป็นการทดลองแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบ ซึ่งปัญหาที่ใช้ในการทดลองมีจำนวนทั้งหมด 6 ปัญหามีการเปรียบเทียบประสิทธิภาพของขั้นตอนวิธีดังกล่าวกับขั้นตอนวิธีเชิงพันธุกรรม กำหนดค่าบางค่าสำหรับทั้งสองขั้นตอนวิธีให้มีค่าตรงกันเพื่อ

ความเท่าเทียมในการเปรียบเทียบ ซึ่งการกำหนดค่าต่างๆสามารถแสดงแสดงดังตารางที่ 4-2ค่าตรงกันของทั้งสองขั้นตอนวิธี คือ จำนวนขนาดของประชากร และ จำนวนรุ่นค้นหาที่มากที่สุด โดยมีค่าเท่ากับ 150 และ 100,000 ตามลำดับ โดย N/A มีความหมายว่าไม่มีข้อมูลของพารามิเตอร์นี้ในขั้นตอนวิธีนั่นเอง

ตารางที่ 4-2 การกำหนดค่าพารามิเตอร์ต่างๆสำหรับการทดลองการแก้ปัญหาซูโดะกุ

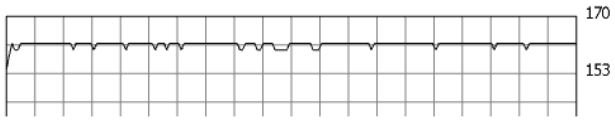
Parameter	COIN	GA
Population size	150	
Selection size	25 (C %)	2
Step size (k)	0.4	N/A
Upper bound	0.99	N/A
Maximum generation	100000	
Independent run	100	
Crossover rate	N/A	0.3
Mutation rate	N/A	0.3

ผลการทดลองการแก้ปัญหาซูโดะกุสามารถแสดงได้ดังตารางที่ 4-3 โดยหลักที่อยู่อันดับแรก คือ Difficulty rating ซึ่งบ่งบอกถึงระดับความยากของปัญหา และ หมายเลขของปัญหา ส่วนถัดมาคือ Givens ซึ่งบ่งบอกว่าปัญหาดังกล่าวมีจำนวนของเลขจำนวนเต็มเริ่มต้นเป็นจำนวนเท่าไหร่ และ Count คือ จำนวนครั้งในทั้งหมดร้อยครั้งของการค้นหาผลเฉลยที่พบผลเฉลยที่เหมาะที่สุด และ AVG GEN คือ จำนวนรุ่นที่ใช้ในการหาผลเฉลยที่เหมาะที่สุดโดยเฉลี่ย นั่นเอง ซึ่งจากการทดลองพบว่าขั้นตอนวิธีการบรรจวบพบผลเฉลยที่เหมาะที่สุดทุกครั้งในทุกๆปัญหาที่กำหนดไว้และใช้จำนวนรุ่นในการค้นหาผลเฉลยที่เหมาะที่สุดน้อยกว่าขั้นตอนวิธีเชิงพันธุกรรมอีกด้วย

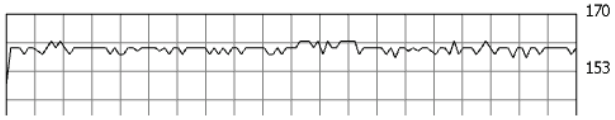
เมื่อพิจารณาการทดลองในทุกระดับความยากในมุมมองของกราฟ พบว่าขั้นตอนวิธีการบรรจวบมีความสอดคล้องกับคุณสมบัติในการหาค่าเหมาะที่สุดหลากหลายรูปแบบ กล่าวคือถึงแม้ว่าการค้นหาจะผ่านมาหลายรุ่น ขั้นตอนวิธีดังกล่าวยังคงไม่ลู่เข้าค่าใดค่าหนึ่ง ซึ่งแปลว่ามีโอกาสที่จะพบผลเฉลยที่มีรูปแบบที่หลากหลายมากขึ้น ซึ่งสามารถแสดงได้ดังภาพที่ 4-5

ตารางที่ 4-3 ผลการทดลองการแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบ

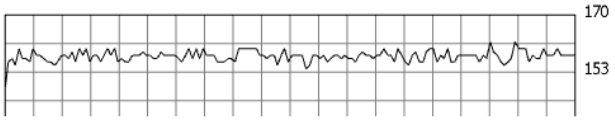
Difficulty rating	Givens	COIN		GA (mut+cross+LS)	
		Count	AVG GEN	Count	AVG GEN
Easy (No. 1)	38	100	2	100	62
Easy (No. 11)	34	100	4	100	137
Medium (No.27)	30	100	130	100	910
Medium (No. 29)	29	100	1196	100	3193
Difficult (No. 77)	28	100	2710	100	9482
Difficult (No. 106)	24	100	2341	96	26825



(a) The graph of solving easy level Sudoku by COIN



(b) The graph of solving medium level Sudoku by COIN



(c) The graph of solving difficult level Sudoku by COIN

ภาพที่ 4-5 กราฟแสดงพฤติกรรมการค้นหาผลเฉลยของขั้นตอนวิธีการบรรจวบ

บทที่ 5

สรุปผลการวิจัย

ในบทนี้จะกล่าวถึงผลสรุปงานวิจัย ปัญหาที่พบ และข้อเสนอแนะอันจะเป็นแนวทางการประยุกต์ใช้ขั้นตอนวิธีการบรรจวบ

5.1 บทสรุป

วิทยานิพนธ์ฉบับนี้ได้นำเสนอการประยุกต์ใช้ขั้นตอนวิธีการบรรจวบสำหรับแก้ปัญหาซูโดะกุในแต่ละระดับความยากได้อย่างมีประสิทธิภาพ โดยเริ่มจากการศึกษาพฤติกรรมการทำงานของขั้นตอนวิธีการบรรจวบ โดยการนำมาแก้ปัญหาการเดินทางของพนักงานขายที่มีลักษณะของปัญหาหลากหลายรูปแบบ และ สุดท้ายจึงนำขั้นตอนวิธีการบรรจวบมาปรับปรุงการทำงานบางส่วนเพื่อให้สามารถแก้ปัญหาซูโดะกุได้อย่างถูกต้องและมีประสิทธิภาพ โดยอ้างอิงการออกแบบจากแนวคิดต่างๆจากงานวิจัยที่ผ่านมา

จากการทดลองในส่วนแรกพบว่า ขั้นตอนวิธีการบรรจวบสามารถแก้ปัญหาหลากหลายรูปแบบได้จริง กล่าวคือ จำนวนของผลเฉลยที่เหมาะที่สุดที่ขั้นตอนวิธีการบรรจวบพบในแต่ละปัญหานั้น มีค่าเท่ากับหรือมากกว่าจำนวนผลเฉลยที่พบของขั้นตอนวิธีเชิงพันธุกรรม ซึ่งแสดงให้เห็นว่าขั้นตอนวิธีดังกล่าวสามารถเก็บรักษาหน่วยโครงสร้างที่นำไปสู่ผลเฉลยที่เหมาะที่สุดจำนวนมากได้อย่างมีประสิทธิภาพ อย่างไรก็ตาม ในบางปัญหาอาจมีผลเฉลยที่เหมาะที่สุดที่มีความแตกต่างกันมากเมื่อเทียบกับผลเฉลยที่เหมาะที่สุดอื่นๆ ซึ่งทำให้ขั้นตอนวิธีดังกล่าวไม่สามารถค้นหาผลเฉลยดังกล่าวได้โดยง่าย

จากการทดลองแก้ปัญหาซูโดะกุโดยขั้นตอนวิธีการบรรจวบ ขั้นตอนวิธีดังกล่าวสามารถแก้ปัญหาดังกล่าว รวมถึงสามารถแก้ปัญหาซูโดะกุที่มีหลายผลเฉลยที่เหมาะที่สุดท้องถิ่นได้อย่างมีประสิทธิภาพ นอกจากนี้เมื่อเปรียบเทียบผลการทดลองกับงานวิจัยก่อนหน้าซึ่งใช้ขั้นตอนวิธีเชิงพันธุกรรมที่ได้รับการปรับปรุงประสิทธิภาพสำหรับการแก้ปัญหาซูโดะกุอย่างต่อเนื่อง และ มีผลการทดลองที่ดีขึ้นเรื่อยๆในการปรับปรุงแต่ละครั้ง พบว่า ขั้นตอนวิธีการบรรจวบสามารถแก้ปัญหาซูโดะกุได้อย่างแม่นยำมากขึ้น และ สามารถค้นหาผลเฉลยที่เหมาะที่สุดโดยใช้จำนวนรุ่นการค้นหาที่น้อยลง โดยไม่ได้อาศัยการทำงานของการค้นหาเฉพาะที่ และ วิธีการกลายพันธุ์เหมือนกับที่ใช้ในขั้นตอนวิธีเชิงพันธุกรรมที่นำมาเปรียบเทียบอีกด้วย

5.2 ข้อเสนอแนะ

จากที่กล่าวไว้ว่าขั้นตอนวิธีการบรรจวบมีลักษณะบางอย่างที่คล้ายกับขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบเส้นเชื่อมโดยเฉพาะในส่วนของแบบจำลองความน่าจะเป็น โดยในงานวิจัยนี้ได้นำลักษณะบางอย่างของตัวอย่างฮิสโทแกรมแบบบัพมาใช้ปรับปรุงขั้นตอนวิธีการบรรจวบ ซึ่งเป็นหนึ่งในการปรับปรุงที่ทำให้ขั้นตอนวิธีการบรรจวบสามารถแก้ปัญหาซูโดะกุได้อย่างมีประสิทธิภาพมากขึ้น อย่างไรก็ตาม จากงานวิจัยที่ผ่านมา ขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบบัพยังมีการปรับปรุงประสิทธิภาพเพิ่มเติม ซึ่งทำให้สามารถแก้ปัญหาได้อย่างมีประสิทธิภาพ โดยมีคุณภาพของผลเฉลยที่ดีขึ้น ชื่อว่า ขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบบัพโดยแผ่นแบบ (Node histogram based sampling algorithm with template) ดังนั้น จึงมีความเป็นไปได้ว่าขั้นตอนวิธีการบรรจวบยังคงสามารถปรับปรุงกระบวนการค้นหาผลเฉลยได้ ตัวอย่างเช่น การประยุกต์การทำงานของแผ่นแบบเพิ่มเติมลงในระบบการค้นหาของขั้นตอนวิธีการบรรจวบแบบบัพ เป็นต้น

รายการอ้างอิง

[1] Xinjie, Y., and Mitsuo, G. Introduction to Evolutionary Algorithm. United Kingdom: Springer-Verlag London Limited, 2010.

[2] Sodsoon, S., Sindhuchao, S., Pitakaso, R., and Pathumnakul, S. An Ant Colony Optimization (ACO) and Solution Improvement Procedures for Multi-Depot Vehicle Routing Problem. IE Network 25 (October 2007): 273-280

[3] Lawler, E.L., Combinatorial Optimization: Networks and Matroids. Mineola, N.Y. : United States of America Dover Publications, 1976.

[4] Intawong, S., and Shevasuthisilp, S. Improvement of Scheduling Using Heuristic Method and Simulation Technique in Electronic Assemblies Industry Production. In The international MultiConference of Engineers and Computer Scientists, 1731-1735. Hong Kong: 2009.

[5] Goldberg, D. E., Lingle, R. Alleles, loci and the TSP. In The first international conference on genetic algorithms and their applications, 154-159. 1985.

[6] Pošík, P. Estimation of Distribution Algorithms. In Pedro Quaresma, Soft Computing and Complex Systems, 119-122. Coimbra, Portugal: Centro Internacional de Matematica, 2003.

[7] Bi, L., and Zhang, S. Analysis and Research Models of the Estimation of Distribution Algorithms. In International Conference on Computer Science and Network Technology (ICCSNT), 2014-2018. China: 2011.

[8] Larrañaga, P. Estimation of Distribution Algorithms. New York: Kluwer Academic Publishers, 2002.

[9] Shin, S., Cho, D., and Zhang, B. Function Optimization with Latent Variable Models. In The Third International Symposium on Adaptive Systems (ISAS2001), 2001.

[10] Baluja, S., and Caruana, R. Removing the Genetics from the Standard Genetic Algorithm. In The 12th International Conference on Machine Learning, 38-46. Morgan Kaufmann Publishers, 1995.

[11] Muhlenbein, H., and Paaß, G. From recombination of genes to the estimation of distributions I: Binary parameters. In Parallel Problem Solving from Nature IV, 178-187. Springer-Verlag, 1996.

[12] Harik, G., Lobo, F., and Glodberg, D. The Compact Genetic Algorithm. IEEE Transactions on Evolutionary Computation 3 (November 1999): 287-297.

[13] Bonet, J., Isbell, C., and Viola, P. MIMIC: Finding Optima by Estimating Probability Densities. Advances in Neural Information Processing Systems 9 (1996): 424-430.

[14] Wattanapornprom, W., and Chongstitvatana, P. Multi-objective Combinatorial Optimization with Coincidence Algorithm. In IEEE Congress on Evolutionary Computation, 1675-1682. Norway: 2009.

[15] Sirovetnukul, R., Chutima, P., Wattanapornprom, W., and Chongstitvatana, P. The Effectiveness of Hybrid Negative Correlation Learning in Evolutionary Algorithm for Combinatorial Optimization Problems. In IEEE Int. Conf. on Industrial Engineering and Engineering Management, 476-481. Singapore: 2011.

[16] Chutima, P., Kampirom, N., Wattanapornprom, W., and Chongstitvattana, P. Application of Combinatorial optimization with coincidence for Multi-Objective Sequencing Problems on Mixed-Model U-Shaped Assembly Lines in JIT Production Systems. In Annual Conference of Kasetsart University, Bangkok: 2008.

[17] Silurian Software. Sudoku Dragon [Online]. 2005. Available from : http://www.sudokudragon.com/ [2013, May 7]

[18] Almog, J. Evolutionary computing methodologies for constrained parameter, combinatorial optimization: Solving the Sudoku puzzle. In AFRICON, 1-6. 2009.

[19] Mantere, T., and Koljonen, J. Solving, rating and generating Sudoku puzzles with GA. In IEEE Congress on Evolutionary Computation (CEC), 1382-1389. 2007.

[20] Sato, Y., and Inoue, H. Solving Sudoku with Genetic Operations that Preserve Building Blocks. In The IEEE Conference on Computational Intelligence and Games (CIG), 23-29. 2010.

[21] Mantere, T., and Koljonen, J. Solving and Rating Sudoku Puzzles with Genetic Algorithms. In The 12th Finnish Artificial Intelligence Conference STeP, 1382-1389. 2007.

[22] Hinterding, R. Mapping, order-independent genes and the knapsack problem. In The IEEE genetic and evolutionary computation conference, 13-17. 1994.

[23] Grefenstette, JJ., Gopal R., .and Rosmaita BJ. et al. Genetic algorithm for the traveling salesman problem. In The 1st international conference on genetic algorithms, 160-168. 1985.

[24] Banzhaf, W. The molecular traveling salesman. In Biol Cybern, 7-14. 1990.

[25] Michalewicz, Z. Genetic algorithms + data structures = evolution programs. Berlin Heidelberg, New York : Springer, 1998.

[26] Fogel, D. A parallel processing approach to a multiple .traveling salesman problem using evolutionary programming. In The fourth annual parallel processing symposium, 318-326. 1990.

[27] Davis, L. Handbook of genetic algorithms. New York: Van Nostrand Reinhold Company, 1991.

[28] Syswerda, G. Schedule optimization using genetic algorithms. In Handbook of genetic algorithms, pp.332-349. New York: 1991.

[29] Tsutsui, S., Pelikan, M., and Goldberg, D.E. Using Edge Histogram Models to Solve Permutation Problems with Probabilistic Model Building Genetic Algorithms. In Illinois Genetic Algorithms Lab (IlliGAL), 2003.

[30] Tsutsui, S., Pelikan, M., and Goldberg, D.E. Node Histogram vs. Edge Histogram: A Comparison of PMBGAs in Permutation Domains. In Missouri Estimation of Distribution Algorithms Laboratory (MEDAL), 2006.

[31] Genina. Sudoku Free [Online]. 2009. Available from: http://www.genina.com/apps/sudoku [2013, May 7]

[32] Reinelt, G. TSPLIB [Online]. 2006. Available from: http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/ [2013, May 7]

[33] Chow, C., and Liu, C. Approximating Discrete Probability Distributions with Dependency trees. In IEEE Transactions on Information Theory, 462-467. 1967.

[34] Etxeberria, R., and Larrañaga, P. Global Optimization using Bayesian Networks. In Special Session on Distributions and Evolutionary Optimization, 332-339. 1999.

[35] Larrañaga, P., Kujipers, C.M. H., Murga, R.H., Inza, I., and Dizdarevic, S. Genetic Algorithms for the Traveling Salesman Problem: A Review of Representation and Operators. Artificial Intelligence Review 13 (1999): 129-170.

[36] Whitley, D., Starkweather, D., and Fuquay, D. Scheduling Problems and Traveling Salesman: The Genetic Edge Recombination Operator. In The International Joint Conference on Artificial Intelligence, 133-140. 1989.

[37] Deb, K., Pratap, A., Agrawal, S., and Meyarivan, T. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA II. IEEE Transactions on Evolutionary Computation 6 (April 2002): 182-197.

[38] Wattanapornprom, W., and Chongstitvatana, P. Solving Multimodal Combinatorial Puzzles with Edge-Based Estimation of Distribution Algorithm. In The Genetic and Evolutionary Computation Conference (GECCO), 67-68. 2011.

[39] Maire, S., and Prissette, C. A restarted estimation of distribution algorithm for solving Sudoku puzzles. Statistics and Computing (2012): 147-160.

[40] Waiyapara, K., and Chongstitvatana, P. Solving Multimodal Problems by Coincidence Algorithm. In Adisorn Leelasantitham, Int. Joint Conf. on Computer Science and Software Engineering (JCSSE), 45-48. Bangkok: 2012.

[41] Lalena, M. TSP solver [Online]. 1996. Available from: http://www.lalena.com/ai/tsp [2013, May 7]

ภาคผนวก

ศัพท์บัญญัติ

ภาษาไทย	ภาษาอังกฤษ
การไขว้เปลี่ยน	crossover
การไขว้เปลี่ยนแบบจุดเดียว	one point crossover
การไขว้เปลี่ยนแบบหลายจุดตัด	multiple point crossover, multi-point crossover
การไขว้เปลี่ยนเอกรูป	uniform crossover
การเชื่อมโยง	linkage
การผกผัน	inversion
การกลายพันธุ์	mutation
การรวมเข้าด้วยกัน, การประกอบเข้าด้วยกันใหม่	recombination
การเลือก, วิธีการเลือก , การเลือกสรร	selection
การลู่เข้า	convergence
การสำรวจ	exploration
การแสวงหาประโยชน์	exploitation
ขั้นตอนวิธี, วิธี	algorithm
ขั้นตอนวิธีการบรรจวบ	Coincidence Algorithm (COIN)
ขั้นตอนวิธีการบรรจวบแบบบัพ	Node based Coincidence Algorithm (NB-COIN)
ขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบเส้นเชื่อม	Edge histogram based sampling algorithm (EHBSA, EHBSA/WO)
ขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบบัพ	Node histogram based sampling algorithm (NHBSA, NHBSA/WO)
ขั้นตอนวิธีการสุ่มตัวอย่างฮิสโทแกรมแบบบัพ-โดยแผ่นแบบ	Node histogram based sampling algorithm (NHBSA/WT)
ขั้นตอนวิธีเชิงพันธุกรรม	Genetic Algorithms (GAs)
ขั้นตอนวิธีประมาณการแจกแจง	Estimation of Distribution Algorithms (EDAs)
ขั้นตอนวิธีเชิงวิวัฒน์	Evolutionary Algorithms (EAs)
ความกดดันในการเลือก	selection pressure

ความซับซ้อนเชิงเวลา	time complexity
ความหลากหลาย	diversity
ค่าความเหมาะสม, ค่าความเหมาะ	fitness value
ค่าเหมาะสมที่สุดครอบคลุม	global optimum
ค่าเหมาะสมที่สุดท้องถิ่น	local optimum
คำตอบ, ผลเฉลย	desired solution
คู่ยีน	allele
จีโนไทป์, รูปแบบพันธุกรรม	genotype
ปริภูมิค้นหา	search space
ปัญหาค่าเหมาะสมที่สุด	optimization problems
ผลเฉลย, คำตอบ	solutions
ผลเฉลยที่เป็นไปได้	candidate solutions
พ่อพันธุ์แม่พันธุ์	parent
ฟังก์ชันประเมินค่าความเหมาะสม	fitness evaluations, function evaluation (FEs)
ฟีโนไทป์, รูปแบบปรากฏ	phenotype
รุ่นประชากร	generation
ลำดับย่อย	sub-string
วิธีการเลือกแบบเอกรูป	uniform selection
หน่วยการสร้าง	building blocks
ปัญหาถุงเป้	Knapsack problem
ปัญหาการเดินทางของพนักงานขาย	Travelling Salesman Problems (TSPs)
ปัญหาการจัดตารางการผลิตสำหรับ-ระบบการผลิตแบบไหล	Flow-shop Scheduling problems (FSSPs)

ประวัติผู้เขียนวิทยานิพนธ์

นายเกียรติโสภณ ไวยพารา เกิดเมื่อวันที่ 14 ตุลาคม พ.ศ. 2530 ที่กรุงเทพมหานคร สำเร็จการศึกษาปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ จากมหาวิทยาลัยมหิดล ในปีการศึกษา 2553 หลังจากนั้นได้เข้าศึกษาในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2554