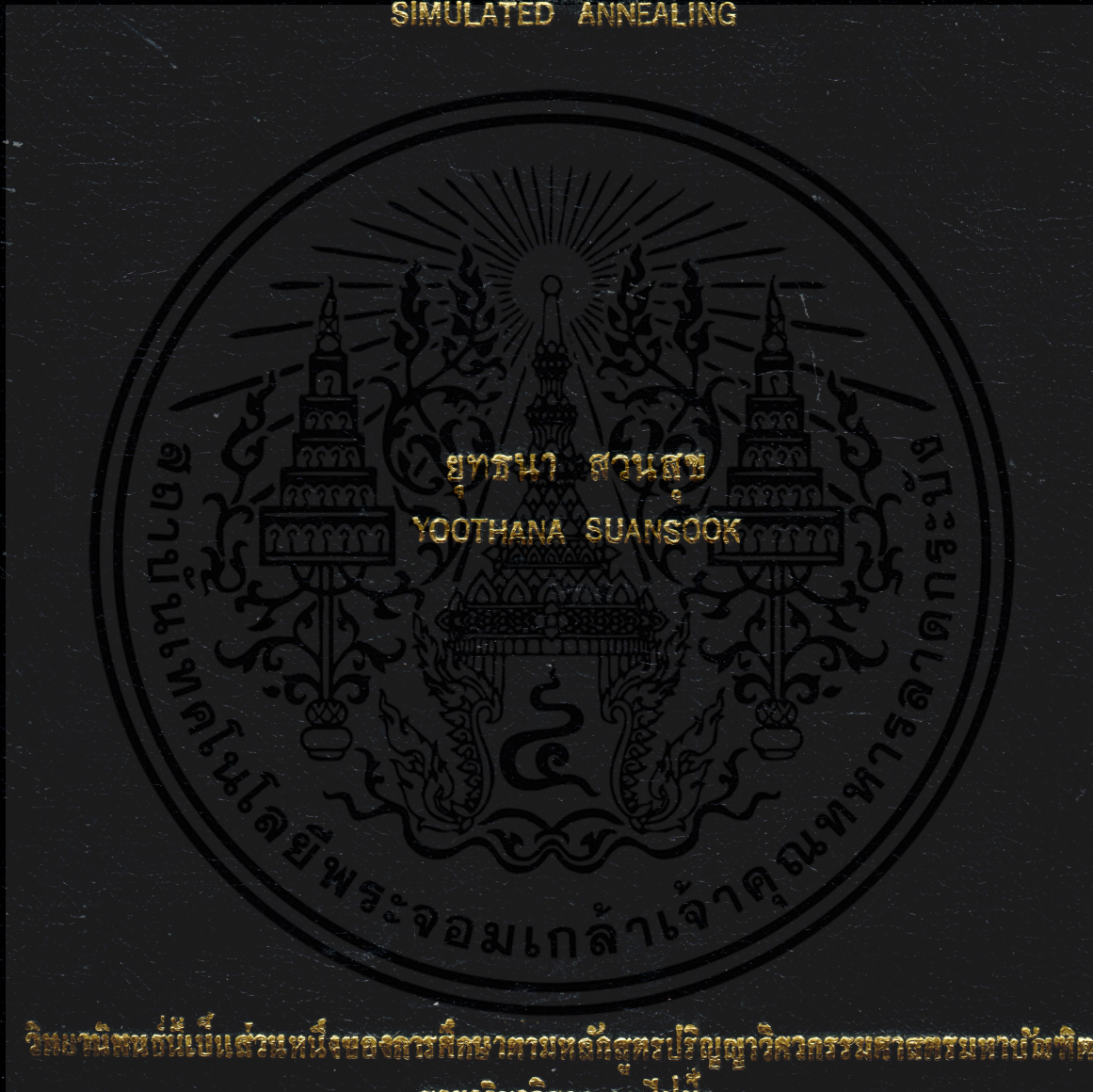


การหาค่าเหมาะสมของการเรียงสับเปลี่ยนโดยวิธีเร่งการจำลองแอนนีลลิ่ง

COMBINATORIAL OPTIMIZATION BY ACCELERATION TECHNIQUES FOR
SIMULATED ANNEALING



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

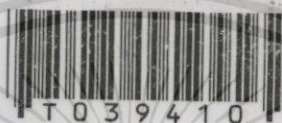
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2544

ISBN 974-648-090-1

การหาค่าเหมาะสมของการเรียงสับเปลี่ยนโดยวิธีเร่งการจำลองแอนนีลลิง

COMBINATORIAL OPTIMIZATION BY ACCELERATION TECHNIQUES FOR
SIMULATED ANNEALING



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2544

ISBN 974-648-090-1

เลขหมู่.....
เลขทะเบียน.....
วัน, เดือน, ปี.....

39410
6
พ.ศ. 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
หากมีการเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**COMBINATORIAL OPTIMIZATION BY ACCELERATION TECHNIQUES FOR
SIMULATED ANNEALING**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE
MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING
SCHOOL OF GRADUDATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2001

ISBN 974-648-090-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การหาค่าเหมาะสม ของการเรียงสับเปลี่ยน โดยวิธีเร่งการจำลองการ
แอนนีลลิ่ง

COMBINATORIAL OPTIMIZATION BY ACCELERATION
TECHNIQUES FOR SIMULATED ANNEALING

ชื่อนักศึกษา นายยุทธนา สนวนสุข

รหัสประจำตัว 37061146

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา วิศวกรรมไฟฟ้า

คณะกรรมการสอบวิทยานิพนธ์	ลายมือชื่อ
ผศ.ดร.ปิติเขต สุรักษา	
ดร.วิศิษฎ์ หิรัญกิตติ	
รศ.ดร.จเร สุรวุฒน์ปัญญา	
รศ.ดร.กิตติ ไพฑูรย์วัฒนกิจ	

วัน/เดือน/ปี ที่สอบ 14 กุมภาพันธ์ 2544 เวลา 12.00-13.00 น.

สถานที่สอบ ณ อาคาร 12 ชั้น ชั้น 4 (ห้อง E12-404)

บัณฑิตวิทยาลัยรับรองแล้ว

(รศ.ดร.บุญวัฒน์ อัครชู)

คณบดีบัณฑิตวิทยาลัย

วันที่ 19 เดือน ธันวาคม พ.ศ. 2544

เอกสารนี้เป็นเอกสารที่ส่งจนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์ การหาค่าเหมาะสมของการเรียงสับเปลี่ยน โดยวิธีเร่งการจำลองแอนนีลลิง
นักศึกษา นายยุทธนา สนวนสุข
รหัสประจำตัว 37061146
ปริญญา วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา วิศวกรรมไฟฟ้า
พ.ศ. 2544
อาจารย์ผู้ควบคุมวิทยานิพนธ์ รศ. ดร. กิตติ ไพฑูรย์วัฒนกิจ

บทคัดย่อ

การประยุกต์ใช้วิธีการทางสถิติและความน่าจะเป็นซึ่งเปรียบเทียบได้กับลักษณะทางกายภาพของ การเย็นตัวลงอย่างช้า ๆ ของสสารเพื่อให้ได้สถานะที่เสถียร ในทางฟิสิกส์ของแข็งซึ่งสามารถนำมาใช้หาค่า ตอบของปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยนหรือคำตอบที่เป็นผลเลิศสำหรับปัญหาที่ยากต่อ การหาค่าตอบหรือปัญหา NP ได้ การเลือกใช้วิธีการคำนวณ โดยการจำลองการแอนนีลลิงจะให้ผลลัพธ์ที่ ต้องพิจารณาถึงคือประสิทธิภาพหรือความเร็วในการคำนวณ เนื่องจากเวลาในการคำนวณด้วยวิธีการนี้ บางครั้งกินเวลานาน ดังนั้นการพิจารณาถึงส่วนประกอบของวิธีการคำนวณด้วยวิธีการแอนนีลลิง เพื่อเร่ง ความเร็วในการหาค่าคำตอบที่เป็นค่าที่เหมาะสมจึงเป็นสิ่งที่ควรศึกษา และมีหลายวิธีที่สามารถใช้เพิ่ม ความเร็วในการคำนวณด้วยวิธีการนี้ได้ ซึ่งแบ่งออกได้เป็นสองลักษณะ คือส่วนที่ง่ายต่อการทำและส่วนที่ ยาก วิธีการที่ง่ายต่อการทำได้แก่ การเลือกการจัดตารางการแอนนีลลิงหรือวิธีการลดอุณหภูมิ ส่วนที่ยาก จะเป็นการปรับปรุงค่าพารามิเตอร์ที่เกี่ยวข้องและการสร้างวิธีการที่จำเพาะสำหรับแก้ปัญหา โดยส่วน ใหญ่จะใช้วิธีดละกันไปสำหรับวิธีการเร่งความเร็วในการคำนวณซึ่งจะให้ผลดีกว่าการเน้นเฉพาะวิธีการ อย่างใดอย่างหนึ่ง ในบางครั้งการนำวิธีการคำนวณแบบนี้ไปเปรียบเทียบกับวิธีการคำนวณแบบทั่วไปจะ พบว่าใช้เวลานาน แต่เนื่องจากวิธีการนี้เหมาะในการใช้หาผลลัพธ์หรือคำตอบของการหาค่าที่เหมาะสม ของการเรียงสับเปลี่ยนได้ วิทยานิพนธ์นี้จะได้เสนอเทคนิคการเพิ่มความเร็วสำหรับวิธีการคำนวณและการ ปรับปรุงวิธีการเพื่อให้ได้วิธีการที่เหมาะสมสำหรับการแก้ปัญหาที่ยากต่อการหาค่าตอบ

Thesis Title Combinatorial Optimization by Acceleration Techniques for Simulated Annealing
Student Mr. Yoothana Suansook
Student ID. 37061146
Degree Master of Engineering
Programme Electrical Engineering
Year 2001
Thesis Advisor Assoc. Prof. Dr. Kitti Paitoonwatanakit

ABSTRACT

Algorithm adapt from physical process of annealing in Condense Matter Physics to obtained stable state capable to find optimal solutions of NP class problem. Decisions made when implementing the simulated annealing algorithm can affect both the algorithm efficiency (speed) and effectiveness. Since the time requirements of the simulated annealing algorithm can often be unacceptable, it is worth investigating various modifications to the basic simulated annealing algorithm in order to accelerate (i.e.'speed up') its convergence to a good quality solution. Several techniques are investigated to accelerate the simulated annealing paradigm. These can be partitioned into two categories those that are simple to implement and those that are more difficult to implement. Simple techniques include the selection of an appropriate annealing schedule, its subsequent parameter fine-tuning, and careful implementation of all problem specific coding requirements. Most of these acceleration techniques are not mutually exclusive and several can often be used in combination to produce time savings of a magnitude greater than those available through the use of individual acceleration techniques. Dramatic time savings are generally unlikely, and although the simulated annealing algorithm can be made to perform more efficiently with a little care, it still remains slow when compared to typical algorithms. Nevertheless, it possesses the ability to regularly produce very good solutions for some difficult combinatorial optimization problems. This thesis refine the acceleration technique for non deterministic algorithm in adaptation to solve a class of difficult problem.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยความช่วยเหลือจากบุคคลหลายฝ่าย ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้ โดยเฉพาะอย่างยิ่ง รองศาสตราจารย์ ดร. กิตติ ไพฑูรย์วัฒนกิจที่ได้กรุณาให้ความอนุเคราะห์ในการใช้อุปกรณ์ระบบไมโครคอมพิวเตอร์ และให้คำแนะนำรวมทั้งตรวจสอบความถูกต้องตลอดการดำเนินการวิจัย

ยุทธนา สวนสุข



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	1
1.3 แนวคิดที่ใช้ในการวิจัย.....	1
1.4 ขอบเขตของการวิจัยและวิธีที่ใช้ในการวิจัย.....	2
บทที่ 2 ปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน.....	3
2.1 วิธีการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน.....	3
2.1.1 แบบจำลองคณิตศาสตร์.....	4
2.2 ความซับซ้อนในการคำนวณ	5
2.3 ปัญหา P และปัญหา NP	7
2.4 การหาค่าคำตอบเฉพาะที่	7
บทที่ 3 แบบจำลองทางคณิตศาสตร์ของการจำลองการแอนนีลลิง.....	9
3.1 หลักการทางกลศาสตร์สถิติและวิธีการจำลองการแอนนีลลิง.....	9
3.2 วิธีการจำลองการแอนนีลลิง.....	10
3.2.1 แบบจำลองคณิตศาสตร์.....	11
3.3 วิธีการเมโทร โพลิส.....	15
3.4 แบบจำลองลูกโซ่มาคอฟ.....	16
3.5 ประเภทของวิธีการจำลองการแอนนีลลิง.....	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
3.5.1 การจำลองการแอนนีลลิงแบบดั้งเดิม.....	19
3.5.2 การจำลองการแอนนีลลิงแบบโบสต์มานน์.....	19
3.5.3 การจำลองการแอนนีลลิงแบบเร็ว.....	19
3.5.4 การจำลองการแอนนีลลิงแบบเร็วมาก.....	20
3.5.5 การจำลองการแอนนีลลิงแบบสองระดับ.....	20
3.7 สมดุลทางสถิติ.....	21
3.8 การพิสูจน์การสุ่มเข้าหาคำตอบ.....	21
บทที่ 4 ค่าพารามิเตอร์ควบคุมของวิธีการจำลองการแอนนีลลิง.....	23
4.1 ประเภทของฟังก์ชันการกระจาย.....	23
4.1.1 คุณสมบัติของฟังก์ชันความหนาแน่นของความน่าจะเป็น.....	23
4.2 เกณฑ์การกำหนดค่าเริ่มต้น.....	27
4.3 วิธีการลดค่าของตัวแปรควบคุม.....	30
4.4 เกณฑ์การหยุดการคำนวณ.....	32
บทที่ 5 การเพิ่มความเร็วยุทธศาสตร์การคำนวณการจำลองการแอนนีลลิง.....	33
5.1 ประเภทของวิธีการเร่งการคำนวณ.....	33
5.1.1 การคำนวณแบบขนาน.....	33
5.1.2 การสร้างขึ้นเป็น Hardware เฉพาะที่ใช้ในการเร่งการคำนวณ.....	35
5.1.3 การออกแบบวิธีการคำนวณในแบบลำดับที่เร็วขึ้น.....	36
5.1.3.1 การเปลี่ยนวิธีการหาค่าฟังก์ชันค่าใช้จ่าย.....	36
5.1.3.2 การสร้างการเปลี่ยนสถานะแบบอื่น.....	37
5.1.3.4 การประมาณค่า Exponential.....	39
5.1.3.5 การประมาณค่าตัวเลขสุ่ม.....	39
5.1.3.6 การเรียกใช้การค้นหาเฉพาะที่ (Local Search Trigger).....	40

สารบัญ (ต่อ)

หน้า

บทที่ 6 การประยุกต์ใช้งานการจำลองการแอนนีลลิ่ง.....	41
6.1 ปัญหาการเดินทางของนักขาย (Traveling Salesman Problem)	41
6.1.1 ฟังก์ชันค่าใช้จ่ายของปัญหาการเดินทางของนักขาย.....	42
6.1.2 วิธีการสร้างการจำลองการแอนนีลลิ่งเพื่อแก้ปัญหาการเดินทางของนักขาย.....	43
6.2 ปัญหาการวางหมากราชินี (N Queen Problem)	49
6.2.1 วิธีการดั้งเดิมที่ใช้ในการแก้ปัญหา	50
6.2.2 วิธีการสร้างขั้นตอนการคำนวณ	51
6.2.3 การคำนวณหาคำตอบโดยการจำลองการแอนนีลลิ่ง.....	52
6.2.4 ฟังก์ชันค่าใช้จ่ายของการวางหมากราชินี.....	52
6.2.5 การกำหนดการเป็นตัวอย่างสำหรับปัญหาการวางหมากราชินี	54
6.3 ปัญหาการจัดวางชิ้นส่วนของอุปกรณ์อิเล็กทรอนิกส์	57
6.3.1 วิธีการที่ใช้ในการจัดวางชิ้นส่วนอุปกรณ์ลงบนแผงวงจรรวม.....	57
6.3.2 การใช้การจำลองการแอนนีลลิ่งสำหรับแก้ปัญหาการจัดวาง.....	59
6.3.3 ฟังก์ชันค่าใช้จ่ายของปัญหาการจัดวาง	60
บทที่ 7 ผลการทดลอง.....	65
7.1 การเปรียบเทียบวิธีการที่ใช้เพิ่มความเร็ว สำหรับแต่ละปัญหา.....	65
7.2 การเปรียบเทียบเวลาที่ใช้ในการคำนวณ	69
บทที่ 8 สรุป วิจัย และข้อเสนอแนะ	73
บรรณานุกรม	74
ภาคผนวก ก. บทความและผลงานการวิจัยที่ได้รับการตีพิมพ์	80
ภาคผนวก ข. การวิเคราะห์ข้อมูลสำหรับการทดลองแบบ RCB (Randomized complete block)	81
ภาคผนวก ค. ตำแหน่งของเมืองสำหรับปัญหาการเดินทางของนักขายขนาด 500 เมือง	85
ภาคผนวก ง. คำตอบของปัญหาการวางหมากราชินีขนาด 200x200	86
ภาคผนวก จ. โปรแกรมต้นฉบับ.....	87
ประวัติผู้เขียน	97

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงเวลาการคำนวณที่เป็นฟังก์ชันของขนาด Input	5
2.2 แสดงความซับซ้อนในการคำนวณของปัญหา	6
4.1 แสดงค่าวิธีการลดอุณหภูมิในแบบต่าง ๆ	31
7.1 แสดงเวลาที่ใช้ในการคำนวณปัญหาการเดินทางของนักศึกษาโดยตัวอย่างปัญหา (Instance) จาก TSPLIB	68
7.2 แสดงเวลาที่ใช้ในการคำนวณสำหรับปัญหาการวางหมากราชินีโดยวิธีการค้นหาเฉพาะที่	69
7.3 เปรียบเทียบเวลาที่ใช้ในการคำนวณปัญหาการเดินทางของนักศึกษา	70
7.4 เปรียบเทียบเวลาที่ใช้ในการคำนวณปัญหาการวางหมากราชินี	70
7.5 ตารางแสดงเวลาที่ใช้ในการคำนวณปัญหาการจัดวางชิ้นส่วนอุปกรณ์อิเล็กทรอนิกส์	71
7.6 ตาราง ANOVA ของปัญหาการเดินทางของนักศึกษา	71
7.7 ตาราง ANOVA ของปัญหาการวางหมากราชินี	72

สารบัญรูป

รูปที่	หน้า
2.1 แสดงอัลกอริทึมการค้นหาเฉพาะที่	8
2.2 แสดงผังวิธีการคำนวณการค้นหาเฉพาะที่.....	8
3.1 แสดงคำตอบของปัญหาการหาค่าที่เหมาะสม ประกอบด้วยค่าต่ำสุดเฉพาะที่และค่าต่ำสุดทั้งหมด...10	
3.2 แสดงผังวิธีการคำนวณการจำลองการแอนนีลลิงแบบทั่วไป	11
3.3 แสดงกราฟฟังก์ชันความน่าจะเป็นของการยอมรับ	12
3.4 แสดง Pseudo Code สำหรับวิธีการจำลองการแอนนีลลิง.....	13
3.5 แสดง Pseudo Code สำหรับวิธีการวิธีการคำนวณแบบเมโทโพรซิส.....	25
4.1 แสดงฟังก์ชันการกระจายแบบ Boltzman Distribution.....	25
4.2 แสดงฟังก์ชันการกระจายแบบ Cauchy Distribution	25
4.3 แสดงฟังก์ชันการกระจายแบบ VFSR Distribution	26
4.4 แสดงการลดค่าอุณหภูมิที่แตกต่างกัน 3 ประเภท ได้แก่ (a) Boltzmann Annealing (b) Cauchy Annealing และ (c) Adaptive Annealing	31
4.5 แสดงการลดค่าอุณหภูมิแบบ Lam	32
5.1 แสดงวิธีการคำนวณการแอนนีลลิงแบบขนานโดยการแบ่งส่วนสำหรับการหาคำตอบ.....	33
5.2 แสดงวิธีการคำนวณแบบขนานของ ProperPLACE-PM	34
5.3 แสดง Schematic Diagram ของ Hardware สำหรับการเร่งความเร็วในการคำนวณการจำลองการ แอนนีลลิง.....	35
5.4 อัลกอริทึมที่ไม่มีการปฏิเสธการยอมรับคำตอบ (Rejectionless) ที่เสนอโดย Greene และ Supowit..	38
6.1 แสดงลักษณะวิธีการสลับสับเปลี่ยนวิธีการเดิน สำหรับปัญหาการเดินทางของนักขาย.....	42
6.2 แสดงกราฟฟังก์ชันค่าใช้จ่ายต่อจำนวนรอบของการคำนวณของปัญหาการเดินทางของนักขาย สำหรับจำนวนเมืองขนาด 123 เมือง.....	43
6.3 แสดงคำตอบของปัญหาการเดินทางของนักขายสำหรับจำนวนเมืองขนาด 123 เมือง.....	44
6.4 แสดงคำตอบของปัญหาการเดินทางของนักขายสำหรับปัญหา drill280 จาก TSPLIB.....	45
6.5 แสดงวิธีการหาคำตอบของปัญหาการเดินทางของนักขาย โดยการจำลองการแอนนีลลิง.....	46
6.6 แสดงคำตอบของปัญหาการเดินทางของนักขายสำหรับจำนวนเมืองขนาด 2000 เมือง โดยเริ่มต้นการคำนวณจาก (a) (b) (c) และ (d) ตามลำดับ	47
6.7 แสดงคำตอบของปัญหาการเดินทางของนักขายสำหรับจำนวนเมืองขนาด 2000 เมือง โดยเริ่มต้นการคำนวณจาก (a) (b) (c) และ (d) ตามลำดับ	48

สารบัญรูป (ต่อ)

รูปที่	หน้า
6.8 แสดงตัวอย่างคำตอบของการวางหมากรากษินีที่จะต้องไม่ให้มีการชนกันของหมากในแนวที่กำหนด.	49.
6.9 แสดงวิธีการหาคำตอบของปัญหาการวางหมากรากษินีโดยการย้อนกลับเส้นทางเดิม.....	50
6.10 แสดงขั้นตอนการคำนวณการจำลองการแอนนิลลิงสำหรับปัญหาการวางหมากรากษินี.....	52
6.11 แสดงฟังก์ชันค่าใช้จ่ายสำหรับปัญหาการวางหมากรากษินี กรณี Random permutation.....	53
6.12 แสดงฟังก์ชันค่าใช้จ่ายสำหรับปัญหาการวางหมากรากษินีเมื่อทำการคำนวณแอนนิลลิง.....	53
6.13 แสดงอัตราส่วนการยอมรับต่อจำนวนรอบของการคำนวณสำหรับปัญหาการวางหมากรากษินี.....	54
6.14 แสดงคำตอบสำหรับการวางหมากรากษินีบนกระดานขนาด 20x20	54
6.15 แสดงคำตอบสำหรับการวางหมากรากษินีบนกระดานขนาด 40x40	55
6.16 แสดงคำตอบสำหรับการวางหมากรากษินีบนกระดานขนาด 100x100	56
6.17 แสดงประเภทของเซลล์ที่จัดวางอยู่ภายในแผงวงจรรวม โดยแยกตามขนาด ได้แก่ (a) Gate array (b) Standard Cell (c) Macro block (d) Sea-of-gate.....	58
6.18 แสดงขั้นตอนการคำนวณการจำลองการแอนนิลลิงสำหรับปัญหาการจัดวาง.....	59
6.19 แสดงค่าความยาวรวมทั้งหมดของจุดที่เชื่อมต่อกันมีค่าลดลง สำหรับ Gate Array cell	60
6.20 แสดงวิธีการพิจารณาหา Bounding Box ระหว่างจุดที่เชื่อมต่อกัน	60
6.21 แสดง Input Netlist ของวงจรตรรกะ.....	61
6.22 แสดง Output ตำแหน่งของแต่ละเซลล์ สำหรับวิธีการจัดวางในแบบ Gate Array	61
6.23 แสดงลักษณะการเชื่อมต่อที่ได้จาก Netlist.....	62
6.24 แสดงผลลัพธ์ของการคำนวณการแอนนิลลิงสำหรับปัญหาการจัดวางชิ้นส่วน โดยฟังก์ชันค่าใช้จ่าย พิจารณาจากผลรวมของระยะทางที่เชื่อมต่อทั้งหมด	63
6.25 แสดงผลลัพธ์ของการคำนวณการแอนนิลลิงสำหรับปัญหาการจัดวางชิ้นส่วน โดยฟังก์ชันค่าใช้จ่าย พิจารณาจากผลรวมของส่วนที่ล้อมรอบ (Bounding Box)	64
7.1 แสดงการคำนวณด้วยวิธีการหาค่าที่เหมาะสมเฉพาะที่ (local optimization)	65
7.2 แสดงเปอร์เซ็นต์ของผลต่างจากค่าที่ทราบค่าที่เหมาะสม (optimum) โดยการเปลี่ยนค่า α สำหรับ ปัญหาการเดินทางของนักขาย โดยใช้ตัวอย่างปัญหา (Instances) จาก TSPLIB	66
7.3 แสดงกราฟเวลาที่ใช้ในการคำนวณปัญหาการเดินทางของนักขายโดยตัวอย่างปัญหา (instance) จาก TSPLIB.....	67
7.4 แสดงกราฟเวลาที่ใช้ในการคำนวณสำหรับปัญหาการวางหมากรากษินีโดยวิธีการค้นหาเฉพาะที่	68

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัญหาการหาค่าตอบที่เป็นผลเลิศหรือการหาค่าที่เหมาะสม เช่น ค่าที่มากที่สุดหรือค่าที่น้อยที่สุด เป็นปัญหาที่พบได้ทั่วไปในทางวิทยาศาสตร์และวิศวกรรมศาสตร์หรือแม้แต่ในทางธุรกิจ การหาผลเลิศสำหรับปัญหาบางประเภทจะยากต่อการหาค่าตอบ การจำลองการแอนนิลลิงเป็นวิธีการหนึ่งที่ประยุกต์หลักการมาจากลักษณะทางกายภาพ ที่ช่วยในการหาค่าตอบของปัญหาที่ยากต่อการหาค่าตอบได้ แต่เนื่องจากวิธีการคำนวณโดยอาศัยการจำลองการแอนนิลลิงเป็นวิธีการคำนวณแบบสุ่มที่อาศัยความน่าจะเป็นในการพิจารณาหาค่าตอบ ดังนั้นข้อจำกัดของวิธีการนี้จะขึ้นกับเวลาที่จะต้องใช้ในการสุ่มตัวอย่างเพื่อที่จะหาค่าตอบที่ดีที่สุดของปัญหา การปรับปรุงวิธีการคำนวณโดยวิเคราะห้พารามิเตอร์ที่มีส่วนเกี่ยวข้อง โดยพิจารณาจากผลลัพธ์ที่ได้จากการคำนวณสำหรับปัญหาแต่ละแบบจะมีประโยชน์อย่างมากเมื่อสามารถนำไปประยุกต์ใช้หาค่าตอบของปัญหาที่ต้องการหาผลเลิศ

1.2 วัตถุประสงค์ของการวิจัย

เพื่อศึกษาหลักการของวิธีการคำนวณแบบสุ่มที่ประยุกต์หลักการมาจากวิธีการทางฟิสิกส์ของของแข็งเพื่อใช้ในการแก้ปัญหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน การศึกษาพารามิเตอร์ที่ใช้ในการควบคุมซึ่งจะเกี่ยวข้องกับประสิทธิภาพในการคำนวณของวิธีการคำนวณด้วยการจำลองการแอนนิลลิง เพื่อที่จะปรับปรุงประสิทธิภาพในการคำนวณ จึงมีการศึกษาลักษณะเฉพาะของพารามิเตอร์แต่ละค่า เช่น ค่าอุณหภูมิเมื่อเริ่มต้น ฟังก์ชันการกระจาย ฟังก์ชันค่าใช้จ่าย บริเวณใกล้เคียง ตัวกำหนดการลดอุณหภูมิ อัตราส่วนของการยอมรับ และลักษณะเฉพาะของแต่ละปัญหา

1.3 แนวคิดที่ใช้ในการวิจัย

ในการคำนวณโดยอาศัยเครื่องคอมพิวเตอร์ หลักเกณฑ์หรือวิธีการที่ใช้ในการแก้ปัญหาค่าที่เหมาะสมของการเรียงสับเปลี่ยนจะขึ้นกับลักษณะของปัญหาว่ามีความซับซ้อนมากน้อยเพียงใด ถ้าปัญหาเป็นปัญหาที่มีขั้นตอนในการคำนวณเพื่อหาค่าตอบไม่ซับซ้อนหรือใช้เวลาการคำนวณไม่มาก ก็สามารถใช้วิธีการคำนวณแบบทั่วไป เช่น การโปรแกรมเชิงเส้นหรือแคลคูลัสในการหาค่าตอบ แต่ถ้าคำตอบของปัญหามีลักษณะที่สามารถจำแนกได้เป็นหลาย ๆ ลักษณะ และมีขนาดหรือจำนวนแบบที่เป็นไปได้เป็นจำนวนมากทำให้ยากต่อการหาค่าตอบด้วยวิธีการทั่วไป การคำนวณโดยการประมาณ หรือการใช้ความน่าจะเป็นและการสุ่มตัวอย่าง ก็สามารถนำมาประยุกต์ใช้กับแก้ปัญหาลักษณะดังกล่าวนี้ได้

1.4 ขอบเขตของการวิจัยและวิธีที่ใช้ในการวิจัย

การวิจัยนี้เป็นการนำวิธีการคำนวณแบบสุ่มประเภทหนึ่งมาใช้สำหรับแก้ปัญหา โดยเฉพาะปัญหาที่จัดเป็นปัญหาที่ยากในการหาคำตอบโดยศึกษาค่าพารามิเตอร์ที่เกี่ยวข้องเพื่อใช้ในการปรับปรุงวิธีการคำนวณให้มีประสิทธิภาพยิ่งขึ้น

ในการแก้ปัญหาแต่ละแบบ ได้ทำการศึกษาความสัมพันธ์ระหว่างค่าพารามิเตอร์ต่าง ๆ ที่เกี่ยวข้อง และเปรียบเทียบผลลัพธ์ที่ได้

วิทยานิพนธ์ฉบับนี้มีวัตถุประสงค์เพื่อที่จะศึกษาวิธีการคำนวณแบบสุ่มที่ใช้หลักการทางกลศาสตร์ และความน่าจะเป็นมาประยุกต์ใช้ในการแก้ปัญหา โดยรายละเอียดของวิทยานิพนธ์นี้สามารถแบ่งออกได้เป็นการศึกษาหลักการพื้นฐานของการหาค่าที่เหมาะสมของวิธีการเรียงสับเปลี่ยนและแบบจำลองทางคณิตศาสตร์ของการหาค่าที่เหมาะสม การจำแนกประเภทของปัญหาที่การหาคำตอบทำได้โดยอาศัยการคำนวณกับเครื่องคอมพิวเตอร์ วิธีการคำนวณแบบสุ่มและความน่าจะเป็น ความน่าจะเป็นของการกระจายที่ใช้ในการคำนวณการจำลองการแอนนิลลิงและพารามิเตอร์ที่เกี่ยวข้อง รวมทั้งปัญหาที่สามารถนำการคำนวณแบบสุ่มไปประยุกต์ใช้ ในการวิจัยจะศึกษาเฉพาะในส่วนการออกแบบวิธีการคำนวณในแบบลำดับ เพื่อที่จะช่วยให้ลดเวลาที่ใช้ในการคำนวณลงและใช้วิธีการวิเคราะห์ทางสถิติ โดยอาศัยการวิเคราะห์ความแปรปรวน (Analysis of Variance) สำหรับทดสอบวิธีการคำนวณ เนื่องจากการคำนวณด้วยวิธีการนี้จะต้องอาศัยเวลาในการพิจารณาหาคำตอบจากเซตคำตอบจำนวนมาก สำหรับบางปัญหาวิธีการคำนวณโดยอาศัยการคำนวณแบบอื่นอาจเหมาะสมกว่า และไม่ใช่ว่าทุกปัญหาจะเหมาะกับการคำนวณด้วยการจำลองการแอนนิลลิงถึงแม้ว่าในการคำนวณจะมีข้อจำกัดในส่วนของเวลา การจำลองการแอนนิลลิงก็สามารถพัฒนาขึ้นใช้ในการคำนวณได้ง่ายซึ่งวิธีการจริง ๆ ของการคำนวณก็คือการเปลี่ยนคำตอบจากคำตอบหนึ่งไปยังคำตอบหนึ่งแล้วเด็อกดูว่าใช่คำตอบที่เหมาะสมแล้วหรือไม่ โดยคำนวณซ้ำไปเรื่อย ๆ จนกว่าจะถึงเกณฑ์การหยุดการคำนวณ ซึ่งกำหนดได้จากค่าพารามิเตอร์ที่เกี่ยวข้อง เช่น อุณหภูมิหรือเกณฑ์ในการหาคำตอบ

บทที่ 2

ปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน

จุดประสงค์ของการหาค่าที่เหมาะสมสามารถพิจารณาได้เป็นการเลือกค่าพารามิเตอร์ (ตัวแปรอิสระ) ที่จะให้ค่าที่เหมาะสม โดยต้องสอดคล้องกับข้อจำกัดที่กำหนดไว้ ปริมาณที่ต้องการหาค่าสูงสุดหรือต่ำสุด จะเรียกว่าฟังก์ชันจุดประสงค์หรือฟังก์ชันค่าใช้จ่าย (objective function) พารามิเตอร์ที่ใช้เปลี่ยนแปลงค่าเพื่อที่จะให้ได้ค่าที่เหมาะสมเรียกว่า ตัวแปรควบคุมหรือตัวแปรตัดสินใจ (decision variables) ข้อจำกัดที่ยอมให้ค่าพารามิเตอร์มีค่าได้ เรียกว่าข้อบังคับ (constraints)

ปัญหาหลายปัญหาในทางปฏิบัติที่ต้องการหาค่าที่ดีที่สุดหรือค่าที่เหมาะสมที่สุดจากเซตของคำตอบที่เป็นไปได้จะมีเงื่อนไขที่ต้องสอดคล้อง ตัวอย่างเช่น การหาค่าน้ำหนักของกระป๋องทรงกระบอกที่มีปริมาตรของของเหลวบรรจุอยู่ในปริมาณที่กำหนดเอาไว้ ให้ถือว่าน้ำหนักของกระป๋องเปล่าเป็นสัดส่วนกับพื้นที่ ปัญหาสามารถจัดรูปได้เป็น การหาขนาดของทรงกระบอกที่มีปริมาตร c ที่มีขนาดของเส้นรอบวงเล็กที่สุด ดังสมการ

minimize
subject to

$$2\pi rh + \pi r^2$$
$$\pi r^2 h = c \text{ และ } r > 0$$

การคำนวณหาค่าตอบจะประกอบด้วยฟังก์ชันที่ต้องการหาค่าต่ำสุดหรือฟังก์ชันจุดประสงค์ และเงื่อนไขที่ต้องสอดคล้องซึ่งกำหนดจากค่าเซตที่เป็นไปได้ ในกรณีนี้คือค่าพารามิเตอร์ (r, h) ที่แทนทรงกระบอกรัศมี r และความสูง h และปริมาตร c เงื่อนไขในกรณีนี้คือข้อกำหนดของปริมาตรซึ่งจะต้องถูกแทนลงในฟังก์ชันค่าใช้จ่าย

ถ้าแต่ละสถานะของคำตอบมีสมาชิกเป็นเซตที่นับได้ถ้วน การหาค่าที่เหมาะสมก็จะเป็นการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน ตัวอย่างของปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน เช่น ปัญหาการมอบหมายงาน (Assignment problem) สมมติว่ามีคนงานอยู่จำนวนหนึ่งและมีจำนวนงานเท่ากับจำนวนคนงานที่ต้องทำงานและทราบว่าคนงานแต่ละคนจะต้องใช้เวลาเท่าไรในการทำงานแต่ละงาน ปัญหาคือจะมอบหมายงานให้คนงานแต่ละคนอย่างไรจึงจะทำให้ใช้เวลาในการทำงานทั้งหมดโดยใช้เวลาสั้นที่สุด การคำนวณสามารถจัดให้อยู่ในรูปการคูณของเมตริกซ์โดยที่ผลลัพธ์เป็นค่าที่น้อยที่สุดของแต่ละแถวของเมตริกซ์แทนคนงานและแต่ละคอลัมน์แทนงานที่แต่ละคนได้รับมอบหมาย

2.1 วิธีการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน

การหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน คือการหาค่าคำตอบของปัญหาที่จะให้ผลเป็นค่าสูงสุด (maximization) หรือค่าต่ำสุด (minimization) โดยคำตอบสามารถพิจารณาได้จากเซตของปัญหาสำหรับจุดประสงค์ในการหาค่าคำตอบของปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน (Combinatorial Optimization) สามารถพิจารณาได้ดังนี้ กำหนดให้ (S, S', f) , เมื่อ S แทนเซตที่จำกัดของคำตอบ และ $S' \subseteq S$ แทนเซตของคำตอบที่เป็นไปได้ เช่น เซตของคำตอบที่สอดคล้องกับเงื่อนไขข้อกำหนดของปัญหา และ $f: S \rightarrow \mathbb{R}$ เป็นฟังก์ชันค่าใช้จ่าย (Cost function) ที่มีการกำหนดค่าจริงให้กับแต่ละค่าของคำตอบ สำหรับปัญหาในการหาค่าคำตอบที่เป็นไปได้ก็คือการหาค่าคำตอบที่ทำให้ค่าฟังก์ชันค่าใช้จ่าย f มีค่าต่ำสุดหรือมีค่าที่เหมาะสม ซึ่งเป็นการหาค่า $i \in S'$ ที่ทำให้ $f(i)$ มีค่าเหมาะสม

คำตอบของการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน สามารถจัดให้อยู่ในรูปของเซตของตัวแปรที่ไม่ต่อเนื่องที่มีค่าจำกัด $X = \{x_1, x_2, \dots, x_n\}$ และถ้าโดเมนของแต่ละค่าของตัวแปร $x_i \in X$ กำหนดโดยเซตที่จำกัด X_i ดังนั้นเซตของคำตอบ S ก็จะกำหนดได้โดย $S = x_1 \times x_2 \times \dots \times x_n$

2.1.1 แบบจำลองคณิตศาสตร์

สำหรับปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน ใด ๆ สามารถจัดรูปในอยู่ในลักษณะคู่ลำดับ (S, f) , เมื่อ S แทนเซตที่มีขนาดจำกัดของชุดคำตอบที่เป็นไปได้ทั้งหมด (Solution Space) และค่าฟังก์ชันค่าใช้จ่าย (Cost Function) f แทนได้ด้วยสมการ

$$f: S \rightarrow \mathbb{R}$$

ในกรณีที่เป็นการหาค่าต่ำสุด ปัญหาจะเป็นการหาค่าตอบ $i_{opt} \in S$ ที่สอดคล้องกับ

$$f(i_{opt}) \leq f(i), \text{ สำหรับทุก ๆ ค่า } i \in S$$

ถ้าเป็นกรณีที่เป็นการหาค่าสูงสุด ค่า i_{opt} จะสอดคล้องกับสมการ

$$f(i_{opt}) \geq f(i), \text{ สำหรับทุก ๆ ค่า } i \in S$$

ค่าคำตอบ i_{opt} จะหมายถึงค่าที่เหมาะสมของทั้งหมด (globally-optimal solution) ทั้งที่เป็นค่าต่ำสุด (minimum) หรือค่าสูงสุด (maximum) หรือเป็นเพียงค่าที่เหมาะสม (optimum) $f_{opt} = f(i_{opt})$ จะแทนฟังก์ชันที่ให้ค่าที่ค่าใช้จ่ายเป็นค่าเหมาะสม และ S_{opt} แทนเซตของคำตอบที่เป็นค่าที่เหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ความซับซ้อนในการคำนวณ

ความซับซ้อนในการคำนวณเป็นการจัดลำดับในทางคณิตศาสตร์ว่า อัลกอริทึมหรือวิธีการที่ใช้ในการคำนวณนั้นจะต้องใช้ทรัพยากรในการคำนวณมากน้อยเพียงใด ในการแก้ปัญหาสำหรับการคำนวณด้วยเครื่องคอมพิวเตอร์

ลองพิจารณาถึงจำนวนคำตอบที่เป็นไปได้ของวิธีการจัดเรียงของ n สิ่งจะพบว่าคำตอบที่เป็นไปได้ทั้งหมด $n!$ วิธี ถ้าพิจารณาฟังก์ชันแฟกทอเรียลจะพบว่าเป็นฟังก์ชันที่มีค่าเพิ่มขึ้นอย่างมากตามค่า n ที่เพิ่มขึ้น เช่น ถ้า $n=70$ ค่า $n!$ จะมากกว่า 10^{100} ดังนั้นถึงแม้ว่าจะใช้เครื่องคอมพิวเตอร์ที่มีความสามารถสูงมากก็ยังคงต้องใช้เวลาในการคำนวณทุก ๆ คำตอบเป็นเวลานานมากจนไม่สามารถคำนวณได้ในทางปฏิบัติ ลองพิจารณาปัญหาในทางปฏิบัติเป็นปัญหาการจัดเรียงสิ่งของ เช่น ปัญหาการจัดวางชิ้นส่วนอุปกรณ์อิเล็กทรอนิกส์ในทางปฏิบัติบนแผงวงจรรวมโดยทั่วไปจะพบว่าจำนวนของชิ้นส่วนอิเล็กทรอนิกส์จะมีจำนวนมากกว่า 70 ชิ้น ดังนั้นการคำนวณโดยเลือกพิจารณาทุก ๆ คำตอบ จะไม่สามารถทำได้ พิจารณาได้จากตารางที่ 2.1

ตารางที่ 2.1 แสดง เวลาการคำนวณที่เป็นฟังก์ชันของขนาด Input (เครื่องหมายดอกจันทน์หมายถึงการคำนวณที่เวลาในการคำนวณอยู่ในรูปโพลีโนเมียล)

ความซับซ้อน	N=20	N=50	N=100	N=200	N=500
$100n^*$	0.2 วินาที	0.05 วินาที	0.1 วินาที	0.2 วินาที	0.5 วินาที
$1000 n \log n^*$	0.09 วินาที	0.3 วินาที	0.6 วินาที	1.5 วินาที	4.5 วินาที
$100 n^2$	0.04 วินาที	0.25 วินาที	1 วินาที	4 วินาที	25 วินาที
$10 n^3$	0.02 วินาที	1 วินาที	10 วินาที	1 นาที	21 นาที
$N^{\log n}$	0.04 วินาที	1.1 ชั่วโมง	220 วัน	125 ศตวรรษ	5×10^8 ศตวรรษ
$2^{n/3}$	0.0001 วินาที	0.1 วินาที	2.7 ชั่วโมง	3×10^4 ศตวรรษ	
2^n	1 วินาที	35 ปี	3×10^4 ศตวรรษ		
3^n	58 วินาที	2×10^9 ศตวรรษ			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 แสดง ความซับซ้อนในการคำนวณของปัญหา [83]

(เครื่องหมายดอกจันเครื่องหมายถึงการคำนวณที่เวลาในการคำนวณอยู่ในรูปโพลีโนเมียล)

ความซับซ้อนของ เวลา	1 วินาที	10^2 วินาที (1.7 นาที)	10^4 วินาที (2.7 ชั่วโมง)	10^6 วินาที (12 วัน)	10^8 วินาที (3 ปี)	10^{10} วินาที (3 ศตวรรษ)
$1000 n^*$	10^3	10^5	10^7	10^9	10^{11}	10^{13}
$1000 n \log n^*$	1.4×10^2	7.7×10^3	5.2×10^5	3.9×10^7	3.1×10^9	2.6×10^{11}
$100 n^2^*$	10^2	10^3	10^4	10^5	10^6	10^7
$10 n^3^*$	46	2.1×10^2	10^3	4.6×10^3	2.1×10^4	10^5
$N^{\log n}$	22	36	54	79	112	156
$2^{n/3}$	59	79	99	119	139	159
2^n	19	26	33	39	46	53
3^n	12	16	20	25	29	33

การจัดลำดับจะเขียนในรูปสัญลักษณ์ Big O เช่น การคูณเมตริกซ์ขนาด $n \times n$ จะถูกจัดลำดับเป็น $O(n^3)$ หรือพิจารณาได้จากนิยามต่อไปนี้

นิยาม สัญลักษณ์ลำดับที่

ให้ N และ R แทนเซตของเลขธรรมชาติ (natural number) ที่เป็นบวกหรือศูนย์ และเซตของค่าจำนวนจริง ตามลำดับและสัญลักษณ์ N^* แทนเลขธรรมชาติที่เป็นบวกเพียงอย่างเดียวเท่านั้น และ R^* ก็แทนเซตที่เป็นค่าจริงที่เป็นบวกอย่างเดียวเท่านั้น และเซต R^* แทนเซตของค่าจริงที่ไม่เป็นลบ และเซตของตรรกที่เป็นจริงหรือเท็จแทนด้วย B

ให้ $f: N \rightarrow R^*$ เป็นฟังก์ชันใดๆ เราจะได้ว่า

$$O(f(n)) = \{t: N \rightarrow R^* \mid (\exists c \in R^*) (\exists n_0 \in N) (\forall n \geq n_0) [t(n) \leq cf(n)]\}. \quad (1.1)$$

หรือเรียกอีกอย่างหนึ่งว่า $O(f(n))$ อ่านว่า ลำดับที่ของ $f(n)$ คือเซตของทุก ๆ ฟังก์ชัน $t(n)$ ที่ล้อมไว้ด้วยค่าที่เป็นผลคูณของค่าจริงที่เป็นบวกของ $f(n)$ โดยที่ค่า n มีค่ามากพอ (มากกว่าค่า n_0)

2.3 ปัญหา P และปัญหา NP

ปัญหา NP (Nondeterministic Problem) เป็นปัญหาที่ถูกจำแนกขึ้นมาเนื่องจากว่าไม่มีวิธีการคำนวณในแบบที่แน่นอน (Deterministic) ที่สามารถหาคำตอบของปัญหาได้ในเวลาที่มีลำดับการคำนวณอยู่ในรูปโพลีโนเมียล สำหรับปัญหาที่สามารถคำนวณหาคำตอบได้ด้วยวิธีการที่แน่นอน ในเวลาที่เป็นโพลีโนเมียลจะจัดเป็นปัญหาประเภท P

ปัญหาที่จัดเป็นปัญหา NP จะยากต่อการหาคำตอบให้ได้ในเวลาอันจำกัดเนื่องจากเวลาที่ใช้ในการคำนวณของปัญหาประเภทนี้จะใช้เวลาเป็น $O(n!)$ หรือ $O(2^n)$ การหาคำคำตอบทุกคำตอบ สำหรับจำนวนรายการที่มีขนาดใหญ่ ๆ ไม่สามารถทำได้สะดวก เนื่องจากฟังก์ชันเป็นฟังก์ชันเพิ่มที่เพิ่มขึ้นเร็ว จึงมีวิธีการประมาณหาคำตอบอีกหลายวิธีที่ได้พัฒนาขึ้นมาเพื่อใช้แก้ปัญหาในลักษณะนี้ แต่วิธีการที่ใช้ในการหาคำตอบก็ไม่ได้รับรองผลลัพธ์ของคำตอบที่ได้อาจจะเป็นค่าที่เป็นเหมาะสม (optimal) ตัวอย่าง ปัญหาประเภทนี้ได้แก่ ปัญหาการเดินทางของนักขาย (Traveling Salesman Problem) ปัญหาการวางหมากราชินี (N-Queen Problem) ปัญหาการกำหนด (Assignment Problem) ปัญหาการระบายสีกราฟ (Graph Coloring Problem) ปัญหาการจัดสิ่งของใส่ลงกล่อง (Knapsack Problem) เป็นต้น

2.4 การหาคำคำตอบเฉพาะที่ (Local Search)

การหาคำคำตอบเฉพาะที่ (Local Search) เป็นวิธีการประมาณค่าคำตอบโดยหาจากวิธีการปรับปรุงค่าฟังก์ชันค่าใช้จ่ายโดยพิจารณาบริเวณที่ใกล้เคียงกับคำตอบ (neighbourhoods) กำหนดให้ (S, f) เป็นชุดของปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน บริเวณที่ใกล้เคียงกำหนดได้จาก

$$N: S \rightarrow 2^S,$$

ที่ซึ่งมีค่าสำหรับแต่ละคำตอบ $i \in S$ และ $S_i \subset S$ เป็นคำตอบที่มีค่าใกล้เคียงกับ i ในบางครั้ง เซตของ S_i จะถูกเรียกว่าบริเวณใกล้เคียงคำตอบ (neighbourhood of solution) ของ i และแต่ละ $j \in S_i$ จะเรียกว่าคำตอบบริเวณใกล้เคียง (neighbouring solution) ของ i นอกจากนี้เราจะถือว่า $j \in S_i \iff i \in S_j$,

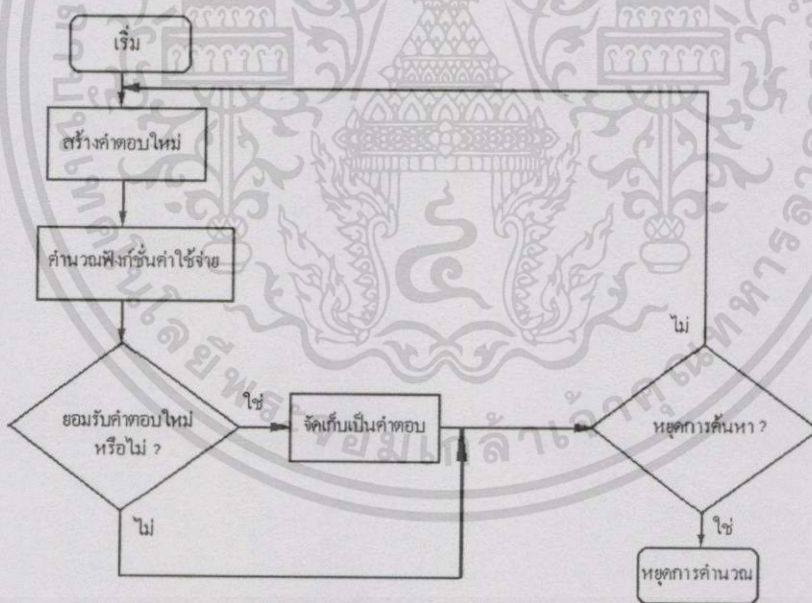
สำหรับปัญหาการหาค่าที่เหมาะสมนั้นเราสามารถหาคำตอบจากบริเวณใกล้เคียง ก่อนได้โดยเริ่มจากการสุ่มค่าเริ่มต้นขึ้นมาค่าหนึ่ง และหาคำคำตอบจากตำแหน่งที่ใกล้ ๆ กัน ถ้าคำตอบที่ได้เป็นคำตอบที่ดีกว่าก็ยอมรับเป็นคำตอบใหม่ วิธีการนี้จะหยุดเมื่อไม่มีการเปลี่ยนแปลงคำตอบ แต่ผลลัพธ์ที่ได้ อาจไม่ใช่คำตอบที่ดีที่สุด วิธีการนี้ เรียกว่า การค้นหาเฉพาะที่ (Local Search Algorithm) ดังแสดงในรูปที่ 2.1

```

procedure LOCAL_SEARCH();
{
    INITIALIZE( $i_{start}$ );
     $i := i_{start}$ ;
    Repeat
        GENERATE ( $j$  from  $S_i$ );
        if  $f(j) < f(i)$  then  $i := j$ ;
    Until  $f(j) \geq f(i)$ , for all  $j \in S$ 
}

```

รูปที่ 2.1 แสดงอัลกอริทึมการค้นหาเฉพาะที่ (Local Search Algorithm)



รูปที่ 2.2 แสดงผังวิธีการคำนวณการค้นหาเฉพาะที่ (Local Search Flow Chart)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

แบบจำลองทางคณิตศาสตร์ของการจำลองการแอนนีลลิ่ง

3.1 หลักการทางกลศาสตร์สถิติของการจำลองการแอนนีลลิ่ง

การจำลองการแอนนีลลิ่งเป็นวิธีการคำนวณแบบสุ่มที่อาศัยหลักการมาจากวิธีการทางกลศาสตร์สถิติ (Statistical Mechanics) ของฟิสิกส์ของของแข็ง (Condense Matter Physics) โดยการคำนวณเป็นการคำนวณที่อาศัยความน่าจะเป็นในการหาคำตอบของปัญหา การคำนวณจะประกอบด้วยฟังก์ชันการยอมรับและฟังก์ชันปฏิเสธ และการคำนวณซ้ำที่อาศัยพารามิเตอร์ที่เป็นตัวกำหนดรอบของการคำนวณที่เปรียบเทียบกับอุณหภูมิในทางฟิสิกส์ ในเรื่องฟิสิกส์ของของแข็งนั้นเมื่อสสารอยู่ในสภาพที่ร้อน การจะทำให้สสารนั้นอยู่ในสภาพที่เป็นผลึกที่มีสถานะเสถียรนั้น สามารถทำได้โดยการค่อย ๆ ทำให้สสารนั้นเย็นตัวลงอย่างช้า ๆ ซึ่งจะให้ความน่าจะเป็นที่ของแข็งที่มีสถานะ i และพลังงาน E อยู่ที่อุณหภูมิ T มีการกระจายแบบโบสต์มันน์ (Boltzmann Distribution) ซึ่งกำหนดได้จากสมการที่ 3.1

$$P\{X = i\} = \frac{1}{Z(T)} \exp\left(-\frac{E_i}{k_b T}\right) \quad (3.1)$$

เมื่อ X เป็นตัวแปรเชิงสุ่มที่แทนสถานะปัจจุบัน และ $Z(T)$ เป็น Partition function ที่มีค่าดังนี้

$$Z(T) = \sum_j \exp\left(-\frac{E_j}{k_b T}\right) \quad (3.2)$$

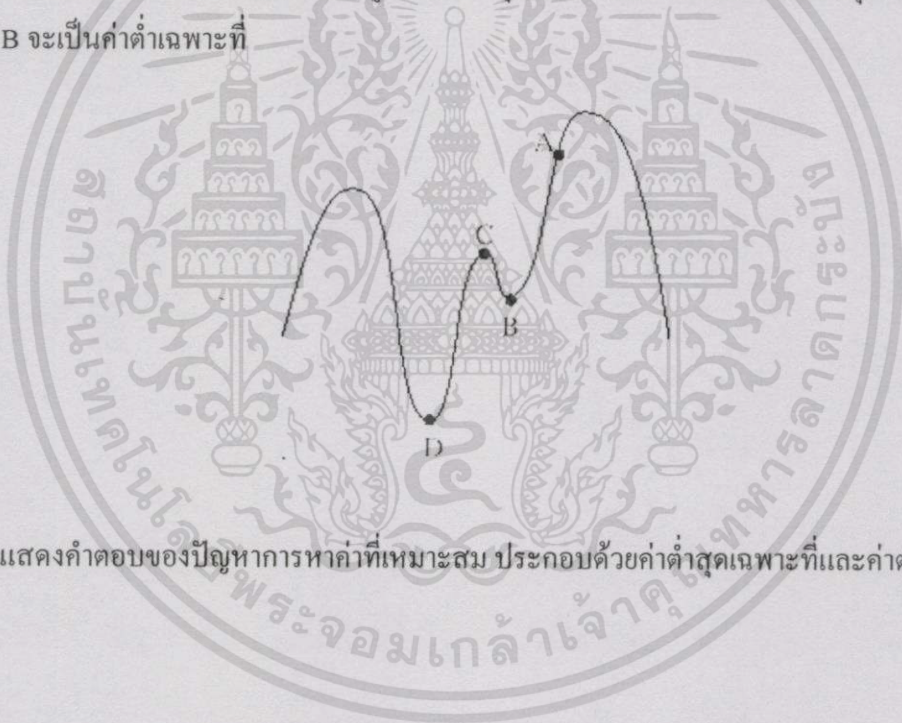
เมื่อเครื่องหมายผลรวมหมายถึงทุก ๆ สถานะที่เป็นไปได้

การเปรียบเทียบลักษณะทางกายภาพกับวิธีการคำนวณสามารถพิจารณาได้ดังนี้ อุณหภูมิที่เริ่มต้นเมื่อร้อนนั้นเป็นตัวกำหนดค่าเริ่มต้นของการคำนวณ การที่สสารค่อย ๆ เย็นตัวลงและอยู่ในสถานะที่เป็นผลึกหรือเป็นสถานะที่เสถียรได้ เปรียบได้กับวิธีการคำนวณที่สามารถหาค่าที่เหมาะสมได้ ซึ่งโดยทั่วไปเป็นค่าที่ต่ำสุดของฟังก์ชัน การค่อย ๆ ลดอุณหภูมิลงทีละน้อย ๆ สามารถเทียบได้กับ วิธีการแอนนีลลิ่งในทางกายภาพ ดังนั้นวิธีการคำนวณนี้จึงเรียกว่า การจำลองการแอนนีลลิ่ง (Simulated Annealing)

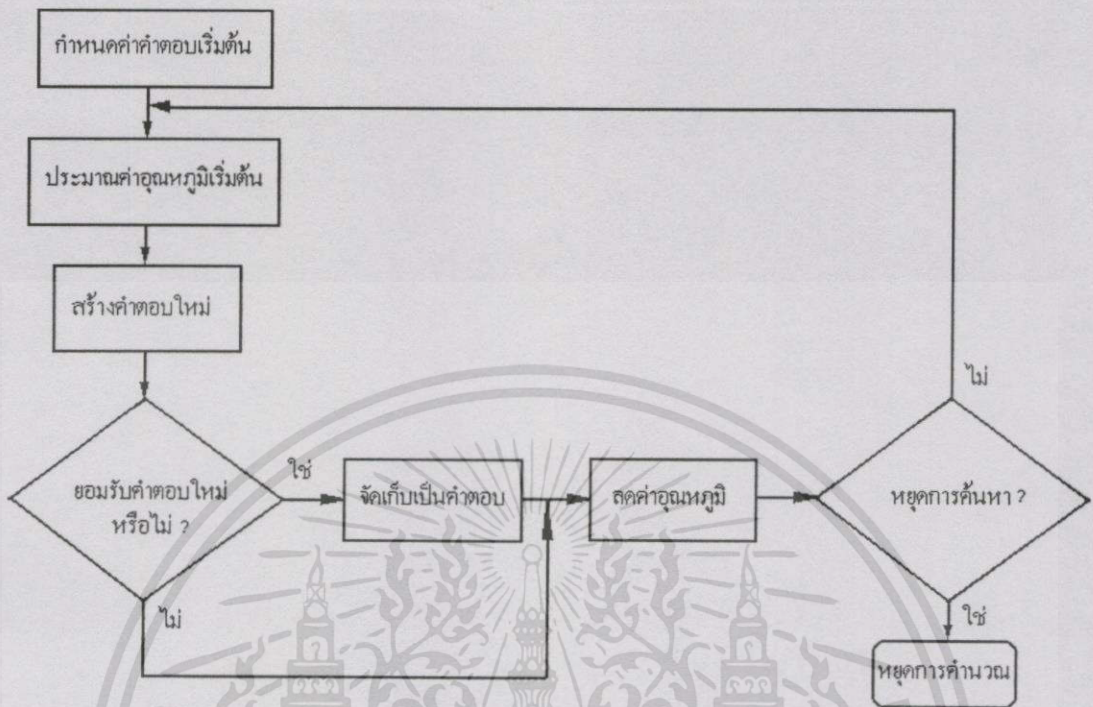
3.2.1 แบบจำลองคณิตศาสตร์

การจำลองการแอนนีลลิ่งเป็นวิธีการหาค่าที่เหมาะสมที่ประยุกต์หลักการมาจากวิธีการทางฟิสิกส์ของการค่อย ๆ เย็นตัวลงของสสารในทางกายภาพเพื่อให้ได้สถานะที่เสถียร เป็นวิธีการที่เหมาะสมสำหรับใช้หาคำตอบของปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยนที่ยากต่อการหาคำตอบหรือปัญหาประเภท NP โดยขั้นตอนวิธีการคำนวณแสดงในรูปที่ 3.2

ส่วนหลักของการคำนวณการจำลองการแอนนีลลิ่งจะประกอบไปด้วยฟังก์ชันการยอมรับซึ่งจะเลือกคำตอบจากสถานะ (state space) ต่าง ๆ ขึ้นมาอย่างสุ่มและจะยอมรับคำตอบจากสถานะที่ถูกเลือกขึ้นมา ถ้าคำตอบที่ได้มีค่าน้อยกว่าคำตอบ ณ ขณะนั้น แต่ถ้าคำตอบที่ได้ไม่ใช่ค่าที่น้อยกว่าก็จะยอมรับเป็นคำตอบด้วยความน่าจะเป็น โดยความน่าจะเป็นที่จะยอมรับคำตอบจากสถานะที่ถูกเลือกขึ้นมาใหม่นั้นจะมีการยอมรับมากหรือน้อยขึ้นกับอุณหภูมิหรือค่าพารามิเตอร์ที่เป็นตัวกำหนดรอบ คำตอบที่ได้ในการคำนวณแต่ละครั้งอาจเป็นค่าที่เหมาะสมเฉพาะที่ ดังในรูปที่ 3.1 จุด D จะเป็นคำตอบที่เป็นค่าต่ำสุดของทั้งหมด ในขณะที่จุด B จะเป็นค่าต่ำเฉพาะที่



รูปที่ 3.1 แสดงคำตอบของปัญหาการหาค่าที่เหมาะสม ประกอบด้วยค่าต่ำสุดเฉพาะที่และค่าต่ำสุดทั้งหมด



รูปที่ 3.2 แสดงผังวิธีการคำนวณการจำลองการแอนนีลลิ่งแบบทั่วไป (Simulated Annealing Flow Chart)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

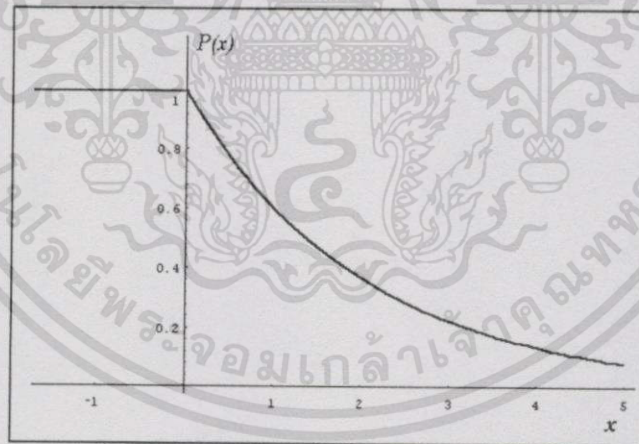
สำหรับคำตอบของปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยนจะเปรียบได้กับสถานะทางกายภาพของสสาร และคำตอบของฟังก์ชันค่าใช้จ่ายจะสอดคล้องกับพลังงานของระบบ

กำหนดให้ (S, f) แทนปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน และค่า i และ j เป็นคำตอบสองคำตอบของฟังก์ชันค่าใช้จ่าย $f(i)$ และ $f(j)$, ตามลำดับ เกณฑ์การยอมรับ (acceptance criterion) จะกำหนดได้จากการที่ค่า j ถูกยอมรับจากค่า i โดยความน่าจะเป็นของการยอมรับ ดังนี้

$$P_c\{\text{accept } j\} = \begin{cases} 1 & \text{if } f(j) \leq f(i) \\ \exp\left(\frac{f(i)-f(j)}{c}\right) & \text{if } f(j) > f(i) \end{cases} \quad (3.3)$$

เมื่อ $c \in \mathbf{R}^+$ แทนค่าพารามิเตอร์ที่เป็นตัวควบคุม ความน่าจะเป็นของการยอมรับแสดงได้ดังในรูปที่ 3.3

ถ้ากำหนดให้ c_k แทนค่าของพารามิเตอร์ควบคุมและ L_k แทนจำนวนที่มีการเปลี่ยนแปลงสถานะจากลำดับที่ k^{th} ในอัลกอริทึมเมโทโพรริส (Metropolis Algorithm) วิธีการจำลองการแอนนิลิ่งสามารถเขียนในรูป pseudocode ได้ดังในรูปที่ 3.4



รูปที่ 3.3 แสดงกราฟฟังก์ชันความน่าจะเป็นของการยอมรับ (Probability of Accept)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure SIMULATED_ANNEALING;
{
  INITIALIZE( $i_{start}, c_0, L_0$ );
   $k := 0$ ;
   $i := i_{start}$ ;
  repeat
    for  $l := 1$  to  $L_k$  do
    {
      GENERATE( $j$  from  $S_i$ );
      if  $f(j) \leq f(i)$  then  $i := j$  else
      if  $\exp(\frac{f(i) - f(j)}{c_k}) > \text{random}[0,1)$  then  $i := j$ 
    }
     $k := k + 1$ ;
    CALCULATE_LENGTH( $L_k$ );
    CALCULATE_CONTROL( $c_k$ );
  until stopcriterion;
}

```

รูปที่ 3.4 แสดง Pseudo Code สำหรับวิธีการจำลองการแอนนีลลิ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 วิธีการคำนวณแบบเมโทโพรสิส

วิธีการคำนวณแบบเมโทโพรสิส ก็คือการคำนวณฟังก์ชันการยอมรับและปฏิเสธด้วยความน่าจะเป็นที่เสนอขึ้นโดย Metropolis ในปี 1953 [61] วิธีการคำนวณแบบนี้มีอีกชื่อหนึ่งว่าวิธีการมอนติคาโล (Monte Carlo) ตั้งขึ้นตามชื่อเมืองที่เป็นแหล่งการพนัน เนื่องจากว่าวิธีการคำนวณใช้ความน่าจะเป็นเป็นหลัก มีการประยุกต์ใช้วิธีการนี้กับปัญหาต่าง ๆ เป็นจำนวนมาก เช่น ปัญหาการไหลของการจราจร (Traffic flow Problem) ปัญหาการออกแบบเครื่องปฏิกรณ์ปรมาณู (Nuclear Reactor design) การฉายรังสีเพื่อการรักษามะเร็ง (Radiation cancer therapy) ตลอดจนถึงปัญหาทางเศรษฐศาสตร์ (Econometrics) ภายหลัง Kirkpatrick ได้เพิ่มตัวแปรควบคุมที่เปรียบได้กับอุณหภูมิเป็นวิธีการจำลองการแอนนีลลิ่ง ในปี 1983 [47]

```

procedure METROPOLIS(t);
{
  INITIALIZE;
  exit := false;
  WHILE not exit DO
  {
    exit := adjust;
    nstate := displace(cstate);
    nenergy := E(nstate);
    if random < exp(-(nenergy-cenergy)/t) then
    {
      exit := readjust;
      cstate := nstate;
      cenergy := nenergy;
    }
  }
}

```

รูปที่ 3.5 แสดง Pseudo Code สำหรับวิธีการวิธีการคำนวณแบบเมโทโพรสิส (Metropolis Algorithm)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 แบบจำลองลูกโซ่มาคอฟ

แบบจำลองคณิตศาสตร์การคำนวณจำลองการแอนนัลลิ่งสามารถพิจารณาได้เป็นลูกโซ่มาคอฟ โดยนิยามของลูกโซ่มาคอฟ มีดังนี้

นิยาม 3.1 ลูกโซ่มาคอฟ (Markov Chain) จะหมายถึงลำดับของความน่าจะเป็นของการลองผิดลองถูกที่ผลลัพธ์ที่ได้ในการลองผิดลองถูกขึ้นกับผลลัพธ์ที่ได้ก่อนหน้า ซึ่งกำหนดได้จากความน่าจะเป็นแบบมีเงื่อนไข ดังนี้

ให้ $X(k)$ เป็นตัวแปรเชิงสุ่มที่แทนผลลัพธ์ที่ได้จากการลองผิดลองถูกในครั้งที่ k^{th} ดังนั้นความน่าจะเป็นของการเปลี่ยนสถานะหรือ Transition probability ครั้งที่ k^{th} สำหรับแต่ละคู่ i และ j ของผลลัพธ์ที่ได้ จากการลองผิดลองถูกจะกำหนดได้โดย สมการที่ 3.4

$$P(k) = P\{X(k) = j | X(k-1) = i\}. \quad (3.4)$$

เมตริกซ์ $P(k)$ ที่สมาชิกแทนได้ด้วยสมการนี้ จะเรียกว่า transition matrix

ให้ $a_i(k)$ แทนความน่าจะเป็นของผลลัพธ์ครั้งที่ i ของการลองผิดลองถูกในครั้งที่ k^{th} จะได้

$$a_i(k) = P\{X(k) = i\}. \quad (3.5)$$

ดังนั้น $a_i(k)$ จะหาได้จาก สมการ Recursion สมการที่ 3.6

$$a_i(k) = \sum_j a_j(k-1)P_{ji}(k) \quad (3.6)$$

นิยาม 3.2 ลูกโซ่มาคอฟจะมีค่าจำกัดถ้ามีการกำหนดอยู่ในเซตที่จำกัด

นิยาม 3.3 ลูกโซ่มาคอฟเรียกว่า Inhomogeneous ถ้าความสัมพันธ์ระหว่างความน่าจะเป็นของการเปลี่ยนสถานะขึ้นกับการลองผิดลองถูกในครั้งที่ k แต่ถ้าความน่าจะเป็นของการเปลี่ยนสถานะไม่ขึ้นกับการลองผิดลองถูกก็จะเรียกว่า homogeneous

นิยาม 3.4 เวกเตอร์ \mathbf{a} เรียกว่าเป็นแบบสุ่ม (Stochastic) ถ้าประกอบด้วยสมาชิก a_i ที่สอดคล้องกับเงื่อนไขต่อไปนี้

$$a_i \geq 0, \quad \text{สำหรับทุก ๆ ค่า } i \quad \text{และ} \quad \sum_i a_i = 1$$

และเมตริกซ์ P จะเรียกว่าเป็นแบบสุ่ม ถ้าประกอบด้วยสมาชิก P_{ij} ที่สอดคล้องกับเงื่อนไขต่อไปนี้

$$P_{ij} \geq 0, \quad \text{สำหรับทุก ๆ ค่า } i, j \quad \text{และ} \quad \sum_j P_{ij} = 1 \quad \text{สำหรับทุก ๆ ค่า } i$$

นิยาม 3.5 ความน่าจะเป็นของการเปลี่ยนสถานะ (Transition probability) ให้ (S, f) แทนตัวอย่างของปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน จะได้ว่าความน่าจะเป็นของการเปลี่ยนสถานะ (Transition probability) ของวิธีการคำนวณการจำลองการแอนนิลลิงกำหนดได้จากสมการที่ 3.7

$$\forall i, j \in S: P_{ij}(k) = P_{ij}(c_k) = \begin{cases} G_{ij}(c_k)A(c_k) & \text{if } i \neq j \\ 1 - \sum_{l \in S} P_{il}(c_k) & \text{if } i = j \end{cases} \quad (3.7)$$

เมื่อ $G_{ij}(c_k)$ แทนความน่าจะเป็นที่ใช้สร้างคำตอบ (Generation probability) เช่น ความน่าจะเป็นในการสร้างคำตอบ j จากคำตอบ i และ $A_j(c_k)$ แทนความน่าจะเป็นของการยอมรับ เช่น ความน่าจะเป็นของการยอมรับคำตอบที่ j โดยที่คำตอบเป็นค่าที่ได้มาจาก i จะเห็นว่าทั้ง $G_{ij}(c_k)$ และ $A_j(c_k)$ เป็นความน่าจะเป็นแบบมีเงื่อนไข (conditional probabilities) ที่แทนได้ด้วยเมตริกซ์ $G(c_k)$ และ $A(c_k)$ จะเรียกว่า เมตริกซ์การสร้าง (generation matrix) และเมตริกซ์การยอมรับ (acceptance matrix) ตามลำดับ

นิยาม 3.6 (Generation Probability)

$$\forall i, j \in S: G_{ij}(c_k) = G_{ij} = \frac{1}{\Theta} \chi_{(S_i)}(j) \quad (3.8)$$

เมื่อ $\Theta = |S|$ สำหรับทุก ๆ ค่าของ $i \in S$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นิยามที่ 3.7 (Acceptance Probability)

$$\forall i, j \in S : A_{ij}(c_k) = \exp\left(-\frac{(f(j) - f(i))^+}{c_k}\right), \quad (3.9)$$

เมื่อทุก ๆ ค่าของ $a \in \mathbb{R}, a^+ = a$ ถ้า $a > 0$, และ $a^+ = 0$ สำหรับกรณีอื่น ๆ

ดังนั้นความน่าจะเป็นของการสร้างหรือการเลือกคำตอบจะถูกเลือกขึ้นมาอย่างอิสระโดยไม่ขึ้นกับค่าพารามิเตอร์ควบคุม c_k และค่านี้จะถูกเลือกมาอย่างสม่ำเสมอจากบริเวณใกล้เคียง S_i โดยถือว่าบริเวณใกล้เคียงทั้งหมดมีขนาดเท่ากัน เช่น $|S_i| = \Theta$ สำหรับทุก ๆ ค่า $i \in S$

ความน่าจะเป็นของการยอมรับ จะสามารถหาค่าได้จากเกณฑ์การยอมรับ หรือ เกณฑ์เมโทโพรสิส ข้อสังเกตจะเห็นว่าทั้งเมตริกซ์การเปลี่ยนแปลงสถานะและเมตริกซ์การสร้างของวิธีการจำลองการแอนนีลลิงเป็นแบบสุ่ม แต่เมตริกซ์การยอมรับจะไม่เป็นแบบสุ่ม

จากนิยามของฟังก์ชันความน่าจะเป็นในการสร้างคำตอบและฟังก์ชันความน่าจะเป็นของการยอมรับจะสอดคล้องกับนิยามของวิธีการจำลองการแอนนีลลิงแบบดั้งเดิมที่เปรียบเทียบวิธีการคำนวณกับลักษณะทางกายภาพ สำหรับวิธีการจำลองการแอนนีลลิงที่ต่างออกไปจากนี้จะเป็นการเปลี่ยนแปลงค่าพารามิเตอร์และฟังก์ชันที่ใช้กำหนดการกระจายของคำตอบ โดยแกนหลักของการคำนวณก็คือวิธีการยอมรับคำตอบด้วยความน่าจะเป็น ถ้าคำตอบที่ได้จากการเลือกเป็นค่าที่ต่ำกว่าค่าคำตอบปัจจุบันก็จะถูกเลือกเป็นคำตอบ แต่ถ้าคำตอบที่ได้จากการเลือกเป็นค่าที่สูงกว่าค่าคำตอบปัจจุบันก็จะถูกยอมรับด้วยความน่าจะเป็น

3.5 ประเภทของวิธีการจำลองการแอนนีลลิง

ในการหาค่าที่ต่ำสุดของฟังก์ชัน $E(x)$ ด้วยวิธีการจำลองการแอนนีลลิง เมื่อ E เป็นฟังก์ชันที่ต้องการหาค่าที่เหมาะสมและ x เป็นค่าเซตของคำตอบที่มีขนาดจำกัด และให้ $N(i)$ แทนสถานะที่เป็นไปได้ที่จะถูกเลือกขึ้นมา การเลือกบริเวณใกล้เคียงขึ้นเป็นคำตอบ สถานะที่ j จะถูกเลือกจากเซตของ $N(i)$ ด้วยความน่าจะเป็น $g(i, j)$

การคำนวณผลต่างของฟังก์ชันค่าใช้จ่าย ของสถานะที่ $E(j)$ และ $E(i)$ สามารถได้จาก

$$\Delta E = E(j) - E(i) \quad (3.10)$$

การเลือกยอมรับสถานะใหม่เป็นคำตอบ พิจารณาจากกรณีที่ ΔE น้อยกว่าศูนย์ สถานะใหม่จะถูกยอมรับด้วยความน่าจะเป็น $h(k)$ ที่อุณหภูมิ T_k โดย k แทนจำนวนครั้งที่ k ในการคำนวณ โดยความน่าจะเป็น $h(k)$ กำหนดได้จากสมการ 3.11

$$h(k) = \exp\left(-\frac{\Delta E}{T}\right) \quad (3.11)$$

ในการคำนวณจะทำการคำนวณซ้ำที่อุณหภูมิ T_k โดยอุณหภูมิ T_k จะค่อย ๆ ลดลง โดยระบบจะคาดว่าจะสามารถเข้าสู่ค่าที่เป็นค่าต่ำสุดของทั้งหมดได้

วิธีการคำนวณการแอนนิลลิ่งในแบบต่าง ๆ ได้ถูกเสนอขึ้นจะมีข้อแตกต่างกันไปบ้างเล็กน้อยจากวิธีการที่กล่าวมา เช่นการปรับเปลี่ยนค่าของพารามิเตอร์ โดยที่หลักการคำนวณพื้นฐานไม่ค่อยเปลี่ยนแปลง โดยจะประกอบด้วยวิธีการคำนวณแบบเมโทโพรซิสเป็นหลัก

แบบของการจำลองการแอนนิลลิ่งที่เปลี่ยนแปลงฟังก์ชันความน่าจะเป็น $g(x)$ และ $h(x)$ ได้แก่ การคำนวณการจำลองการแอนนิลลิ่งแบบโบลต์ซมันน์ (Boltzmann Annealing), การคำนวณการจำลองการแอนนิลลิ่งแบบโคชี (Cauchy Fast Annealing) และการคำนวณการแอนนิลลิ่งแบบปรับเปลี่ยนได้ (Adaptive Simulated Annealing ASA) ภายหลังใช้ชื่อว่า การจำลองการแอนนิลลิ่งแบบเร็วมาก (Very Fast Simulated Annealing) ซึ่งจะให้ผลลัพธ์ของการเข้าสู่ค่าแตกต่างกันไปในแต่ละวิธี

สรุปได้ว่าการคำนวณจำลองการแอนนิลลิ่งจะประกอบด้วย ฟังก์ชันความน่าจะเป็น 2 ส่วน และฟังก์ชันการลดค่าอุณหภูมิสำหรับการใช้ในการกำหนดจำนวนรอบในการคำนวณ ดังนี้

$g(x)$ เป็นฟังก์ชันความน่าจะเป็นที่จะเลือกคำตอบจากบริเวณใกล้เคียงขึ้นมาเป็นคำตอบใหม่

$h(x)$ เป็นฟังก์ชันความน่าจะเป็นที่จะยอมรับค่าสถานะใหม่เป็นคำตอบ

$T(k)$ เป็นฟังก์ชันที่ขึ้นกับตัวแปรที่เปรียบเทียบกับค่าอุณหภูมิ เรียกว่าฟังก์ชันกำหนดการเย็นตัวลง (cooling schedule) หรือฟังก์ชันการลดอุณหภูมิ ผลของการลดอุณหภูมิจะเนื่องมาจากฟังก์ชันความน่าจะเป็นทั้ง 2 แบบที่กล่าวมาจะให้ค่าผลลัพธ์ในการคำนวณแตกต่างกันไป

นอกจากการคำนวณการแอนนิลลิ่งทั้ง 3 แบบที่กล่าวมาก็ยังมีวิธีการคำนวณการแอนนิลลิ่งวิธีอื่นที่แตกต่างกันไป โดยแทนที่จะเป็นการเปลี่ยนค่าพารามิเตอร์ เช่น การเลือกค่าฟังก์ชันการเย็นตัวลงในแบบที่ต่างออกไป [1] ก็อาจเป็นการเพิ่มวิธีการบางอย่างเข้ามาในการคำนวณหรืออาจใช้การรวมเอาวิธีการคำนวณแบบอื่นเข้าไว้ด้วย เช่น Genetic Annealing [60]

3.5.1 การจำลองการแอนนีลลิ่งแบบดั้งเดิม

การจำลองการแอนนีลลิ่งเสนอขึ้นโดย Kirkpatrick ในปี 1983 [47] โดยที่วิธีการคำนวณวิธีนี้ใช้ในการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยนแบบไม่ต่อเนื่อง ค่าที่ใช้ในการกำหนดการเย็นตัวกำหนดได้จากสมการ

$$T_k = \alpha T_{k-1} = \alpha^k T_0, \quad (3.12)$$

เมื่อ $\alpha \in [0,1)$ เป็นค่าคงที่ โดยทั่วไปค่านี้จะมีค่าอยู่ระหว่าง $0.8 < \alpha < 0.99$ ในการหาค่าที่เหมาะสมโดยวิธีการนี้ สถานะใหม่คำนวณจากการสลับค่าตัวแปรที่แทนคำตอบของแต่ละแบบ สำหรับการคำนวณที่ตัวแปรที่มีลักษณะไม่ต่อเนื่องจะไม่ต้องอาศัยฟังก์ชันการกระจาย

3.5.2 การจำลองการแอนนีลลิ่งแบบโบสค์มานน์

การจำลองการแอนนีลลิ่งวิธีนี้เสนอขึ้นโดย Geman ในปี 1984 [37] โดยเพิ่มเงื่อนไขที่จำเป็นสำหรับการรู้เข้าหาคำตอบเพื่อให้ได้ค่าที่เป็นค่าที่เหมาะสมของทั้งหมด ปกติวิธีการนี้จะเรียกว่าการจำลองการแอนนีลลิ่งแบบดั้งเดิมหรือการจำลองการแอนนีลลิ่งแบบโบสค์มานน์

จากหลักการทางฟิสิกส์ และการกระจายแบบ Gaussian สำหรับหนึ่งตัวแปร จะมีค่าเป็น

$$g(x, s_k) = \frac{1}{\sqrt{2\pi s_k}} e^{-(x-x_0)^2/2s_k} \quad (3.13)$$

เมื่อ x_0 เป็นคำตอบปัจจุบันของตัวแปรที่ใช้หาค่าที่เหมาะสม x และการลดอุณหภูมิคำนวณได้จาก

$$T_k = \frac{T_0}{\ln(k+1)}, k = 1, \dots, \infty \quad (3.14)$$

3.5.3 การจำลองการแอนนีลลิ่งแบบเร็ว

การจำลองการแอนนีลลิ่งแบบเร็วเป็นวิธีการคำนวณที่ถูกเสนอขึ้นโดย Szu ในปี 1987 [84] เป็นวิธีการคำนวณกึ่ง การทำการค้นหาเฉพาะที่เพราะจากลักษณะการกระจายจะมีการกระโดดหรือการสุ่มหาคำตอบใหม่เป็นช่วงกว้าง วิธีการคำนวณปรับปรุงมาจาก การจำลองการแอนนีลลิ่งแบบโบสค์มานน์ โดยอาศัยฟังก์ชันการกระจายแบบ โคชี (Cauchy Distribution) ดังสมการ 3.15

$$g(x, s_k) = \frac{s_k}{\pi[(x - x_0)^2 + s_k]}, \quad (3.15)$$

และใช้การลดอุณหภูมิแบบเร็ว

$$T_k = \frac{T_0}{1+k}, \quad (3.16)$$

3.5.4 การจำลองการแอนนีลิ่งแบบเร็วมาก

ในปี 1989 Ingber ได้เสนอวิธีการจำลองการแอนนีลิ่งแบบเร็วมาก โดยวิธีการนี้จะยอมให้มีการกำหนดการลดอุณหภูมิลงแบบ fast exponential [40] โดยอาศัยฟังก์ชันความหนาแน่น

$$g(x, s_k) = \frac{1}{2\pi(|\Delta x'| + s_k) \ln(1 + \frac{1}{s_k})}, \quad (3.17)$$

เมื่อ $|\Delta x'|$ เป็นค่า normalize step มีค่าเท่ากับ $(x - x_0) / (x_{\max} - x_{\min})$, เมื่อ x_{\min} เป็นขีดจำกัดล่าง และ x_{\max} เป็นขีดจำกัดบนของตัวแปรที่ต้องการหาค่าที่เหมาะสม ค่าขีดจำกัดบนและค่าขีดจำกัดล่างทั้งสองค่านี้จะต้องมีการกำหนดให้กับทุก ๆ ตัวแปรที่ต้องการหาค่าที่เหมาะสม และการลดอุณหภูมิกำหนดได้จาก

$$T_k = T_0 \exp(-ck^{1/n}) \quad (3.18)$$

เมื่อ

$$c = \ln\left(\frac{T_0}{T_f}\right) \exp(-N_{\max}/n) \quad (3.19)$$

และ N_{\max} เป็นจำนวนตัวอย่างที่เลือกที่มากที่สุด (Maximum number of samples)

3.5.5 การจำลองการแอนนีลิ่งแบบสองระดับ

Varanelli และ Cohoon [82][83] ได้เสนอวิธีการจำลองการแอนนีลิ่งที่แบ่งขั้นตอนการคำนวณออกเป็น 2 ชั้นเรียกว่าการจำลองการแอนนีลิ่งแบบสองระดับ (Two-stage Simulated Annealing) โดยวิธีการนี้สามารถนำไปใช้ได้กับปัญหาทั่วไปได้ หลักการคำนวณคือการแบ่งการคำนวณออกเป็น 2 ชั้น โดยชั้นแรกใช้วิธีการคำนวณแบบอื่น เช่น อาจใช้วิธีการค้นหาเฉพาะที่ช่วยในการหาค่าตอบเริ่มต้นก่อน และชั้นที่สองจึงใช้การคำนวณการจำลองการแอนนีลิ่งคำนวณต่อ การกำหนดค่าพารามิเตอร์ที่เกี่ยวข้องสำหรับการคำนวณวิธีนี้จะใช้วิธีการประมาณค่าและหาช่วงความน่าจะเป็น คล้ายกับวิธีการหาค่าคะแนนมาตรฐานสำหรับข้อมูลที่มีการกระจายแบบปกติ [86]

3.6 สมดุลทางสถิติ

สมดุลทางความร้อนสามารถพิจารณาได้จากลักษณะทางกายภาพที่ขึ้นกับสภาพแวดล้อมในขณะนั้น เช่น ถ้าเติมน้ำร้อนที่มีอุณหภูมิ 100 องศาเซลเซียส ปริมาณ 1 ลิตรลงในอ่างน้ำเย็นอุณหภูมิ 20 องศาเซลเซียส ที่มีปริมาณน้ำเย็น 1 ลิตรเท่ากัน สมดุลทางความร้อนของน้ำจะเป็นน้ำอุ่นที่มีอุณหภูมิอยู่ระหว่างค่า 20 -100 องศาเซลเซียส แต่ถ้าเติมน้ำร้อน 1 ลิตรลงในทะเลสาบน้ำแข็ง สมดุลทางความร้อนของน้ำจะเป็นน้ำที่มีอุณหภูมิไม่เปลี่ยนแปลงไปจากเดิมมากนักก็จะเป็นน้ำที่มีอุณหภูมิใกล้เคียงกับน้ำแข็ง จะเห็นว่าลักษณะของสมดุลทางความร้อนจะขึ้นกับสภาพแวดล้อมในขณะนั้น แต่สมดุลทางความร้อนสำหรับการแอนนิลลิงจะหมายถึงการที่คำตอบที่ได้จากการคำนวณในแต่ละรอบไม่เปลี่ยนแปลงค่าไปจากเดิมมากนัก ประเภทของสมดุลทางความร้อนของการจำลองการแอนนิลลิงแบ่งออกได้เป็น homogeneous algorithm คือ การลดอุณหภูมิระหว่างช่วงของการคำนวณลูกโซ่มาคอฟ และ inhomogeneous algorithm คือการลดอุณหภูมิหลังจากทำการคำนวณลูกโซ่มาคอฟแล้ว [4] การคำนวณลูกโซ่มาคอฟจะหมายถึงลำดับของการลดผิดพลาดที่เกิดขึ้นในแต่ละครั้งของการลดผิดพลาดนั้นค่าที่ได้จะขึ้นกับผลการคำนวณก่อนหน้า [51]

3.7 การพิสูจน์การลู่เข้า

การคำนวณด้วยการจำลองการแอนนิลลิงจะมีการลู่เข้าหาคำตอบที่เป็นค่าที่ดีที่สุดของทั้งหมดได้ เนื่องจากมีทฤษฎีที่พิสูจน์การลู่เข้าหาคำตอบสำหรับวิธีคำนวณแบบนี้ไว้ ซึ่งพิจารณาได้ดังนี้

ในการคำนวณโดยการซ้ำ (Iteration) วิธีการแต่ละวิธีจะหาคำตอบได้หรือไม่ขึ้นกับว่า ในการคำนวณนั้น มีการลู่เข้าหาคำตอบที่ต้องการหรือไม่ ในการคำนวณแบบสุ่มก็เช่นกัน ก่อนจะนำวิธีการคำนวณไปใช้ จะต้องแสดงให้เห็นได้ว่ามีการลู่เข้าหาคำตอบได้ก่อน สำหรับอัลกอริทึมการจำลองการแอนนิลลิงจะมีการลู่เข้าหาคำตอบที่เป็นค่าที่ดีที่สุดของทั้งหมดได้จากการเปลี่ยนจากสถานะหนึ่งไปยังอีกสถานะหนึ่ง ต้องสอดคล้องกับเงื่อนไขต่อไปนี้

ให้ x_i เป็นสถานะที่สามารถหาได้จากสถานะที่ x , ถ้ามีเส้นทางที่ $x, x_{i+1}, x_{i+2}, \dots, x_{i+n} = x$ สำหรับค่า n ที่มากกว่าหรือเท่ากับศูนย์ ฟังก์ชันจะไม่สามารถลดรูปได้ก็ต่อเมื่อมีการเปลี่ยนแปลงสถานะจาก x ไปยังสถานะ x , โดยจำนวนครั้งในการเปลี่ยนแปลงมีจำกัด เงื่อนไขการหาค่าที่เหมาะสมของทั้งหมดจะได้ว่า

$$\lim_{x \rightarrow \infty} [x \in R_{opt}] = 1, \tag{3.20}$$

ซึ่งมีค่าเท่ากับ

$$\lim_{x \rightarrow \infty} [x \notin R_{opt}] = 0, \tag{3.21}$$

โดยให้ความน่าจะเป็นที่จะสร้างสถานะที่ x มีค่าเป็น $g(x, s_k)$ ที่อุณหภูมิ T_k และความน่าจะเป็นที่จะไม่สร้างสถานะจึงมีค่าเป็น $1 - g(x, s_k)$ โดยที่ subscript ตัวที่ k แทนรอบที่ k ของการคำนวณ ดังนั้นค่าเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

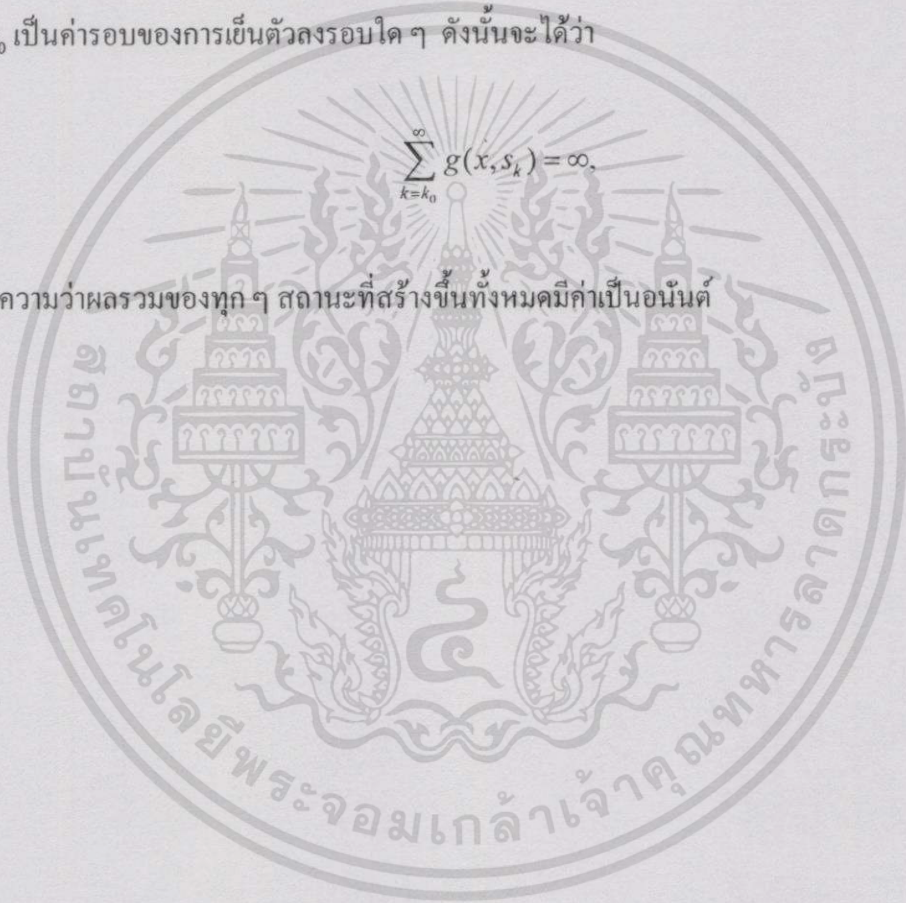
ของคำตอบที่เป็นค่าต่ำสุดของทั้งหมดจะมีการหาได้อย่างแน่นอน ถ้าในทุก ๆ สถานะที่เป็นไปได้มีโอกาสถูกเลือกขึ้นมาในแต่ละรอบของแต่ละค่าของอุณหภูมิ ดังนั้นเพื่อให้แน่ใจได้ว่าทุก ๆ ค่าที่เป็นไปได้มีโอกาสถูกเลือกขึ้นมา จะพิจารณาได้จากเงื่อนไขที่สถานะนั้นจะไม่ถูกสร้าง ซึ่งจะได้ว่าผลคูณของความน่าจะเป็นที่จะไม่ถูกเลือกขึ้นมาในการคำนวณแต่ละรอบเป็นศูนย์ ดังสมการ 3.22

$$\prod_{k=k_0}^{\infty} (1 - g(x, s_k)) = 0, \quad (3.22)$$

เมื่อค่า k_0 เป็นค่ารอบของการเริ่มต้นวงจรรอบใด ๆ ดังนั้นจะได้ว่า

$$\sum_{k=k_0}^{\infty} g(x, s_k) = \infty, \quad (3.23)$$

ซึ่งหมายความว่าผลรวมของทุก ๆ สถานะที่สร้างขึ้นทั้งหมดมีค่าเป็นอนันต์



ค่าพารามิเตอร์ควบคุมของวิธีการคำนวณการจำลองการแอนนีลิ่ง

4.1 ประเภทของฟังก์ชันการกระจาย

ในการคำนวณที่อาศัยความน่าจะเป็นในการสร้างแบบจำลองขบวนการทางกายภาพ เช่น การเคลื่อนที่ของอนุภาคนิวตรอนที่มีพลังงาน 2 MeV ในแท่งน้ำ จะต้องอาศัยการสุ่มค่าจากฟังก์ชันที่อธิบายลักษณะการกระจายอยู่ของอนุภาคดังกล่าว ฟังก์ชันที่ใช้แสดงลักษณะนี้จะเรียกว่า ฟังก์ชันความหนาแน่นของความน่าจะเป็น (Probability Density function) หรือเรียกย่อ ๆ ว่า *pdf* จากตัวอย่าง ฟังก์ชันที่อธิบายลักษณะการกระจายของอนุภาคนิวตรอนคือฟังก์ชันที่มีการกระจายเป็นแบบ Exponential

4.1.1 คุณสมบัติของฟังก์ชันความหนาแน่นของความน่าจะเป็น

สำหรับฟังก์ชัน $f(x)dx$ ที่ความน่าจะเป็นที่ตัวแปรเชิงสุ่มจะมีค่าอยู่ในช่วง $(x, x+dx)$ สามารถเขียนได้เป็น

$$\text{prob}(x \leq x' < x+dx) \equiv P(x \leq x' < x+dx) = f(x)dx \quad (4.1)$$

เนื่องจากความน่าจะเป็นไม่มีหน่วย ดังนั้น $f(x)$ จะมีหน่วยเป็นส่วนกลับกับของตัวแปรเชิงสุ่ม เช่น 1/cm หรือ 1 sec ขึ้นกับหน่วยของ x

สำหรับความน่าจะเป็นที่หาโดยการหาค่าของตัวแปรเชิงสุ่มในช่วง $[a, b]$ จะมีค่าดังนี้

$$\text{prob}(a \leq x < b) \equiv P(a \leq x' \leq b) = \int_a^b f(x')dx \quad (4.2)$$

ซึ่งจะหมายถึงพื้นที่ใต้กราฟ $f(x)$ จาก $x = a$ ไปจนถึง $x = b$

ในการคำนวณการจำลองการแอนนีลิ่งฟังก์ชันการเลือกคำตอบ จะทำการเลือกบริเวณใกล้เคียงขึ้นมาพิจารณาเป็นคำตอบ โดยในการสุ่มตัวอย่างจะเลือกจากฟังก์ชันความหนาแน่นของความน่าจะเป็นซึ่งจะสัมพันธ์กับฟังก์ชันการเปลี่ยนสถานะ ดังนี้

ถ้าสถานะที่ x , ถูกสร้างจากสถานะปัจจุบันที่ x , ที่อุณหภูมิ T ด้วยความน่าจะเป็นในการสร้างคำตอบ (Generation probability) $G_y(T)$ สำหรับปริภูมิขนาด n มิติจะได้ว่าความน่าจะเป็นในการสร้างคำตอบเป็นผลคูณของความน่าจะเป็นย่อย ในแต่ละมิติ $G(x, s)$ ดังสมการที่ 4.3

$$G_y(T) = \prod_{a=1}^n G(x_a, s_a) \quad (4.3)$$

โดย x เป็นคำตอบใดๆ และ s แทนค่าความแปรปรวน

เมื่อ n เป็นจำนวนตัวแปรที่ต้องการหาค่าที่เหมาะสม และ

$$G(x, s) = \int_{-\infty}^x g(\Delta x, s) dx \quad (4.4)$$

เมื่อ $\Delta x = x_j - x$, และ $s = s(T)$ แทนค่าความแปรปรวนของตัวแปรที่หาค่าเหมาะสม ฟังก์ชันการความหนาแน่นของความน่าจะเป็นในการสร้าง $g(\Delta x, s)$ หรือ Generation probability density function จะไม่เกี่ยวข้องกันสำหรับตัวแปรการหาค่าที่เหมาะสมที่ต่างกัน และจากอุณหภูมิที่เป็นพารามิเตอร์ควบคุมในการคำนวณการแอนนีลลิ่ง ที่อุณหภูมิสูงจะมีโอกาสเลือกคำตอบใหม่ได้ไกลกว่าที่อุณหภูมิต่ำ และสำหรับค่าความแปรปรวนจะกำหนดได้จาก

$$s = \int_{-\infty}^{\infty} x^2 g(x, s) dx \quad (4.5)$$

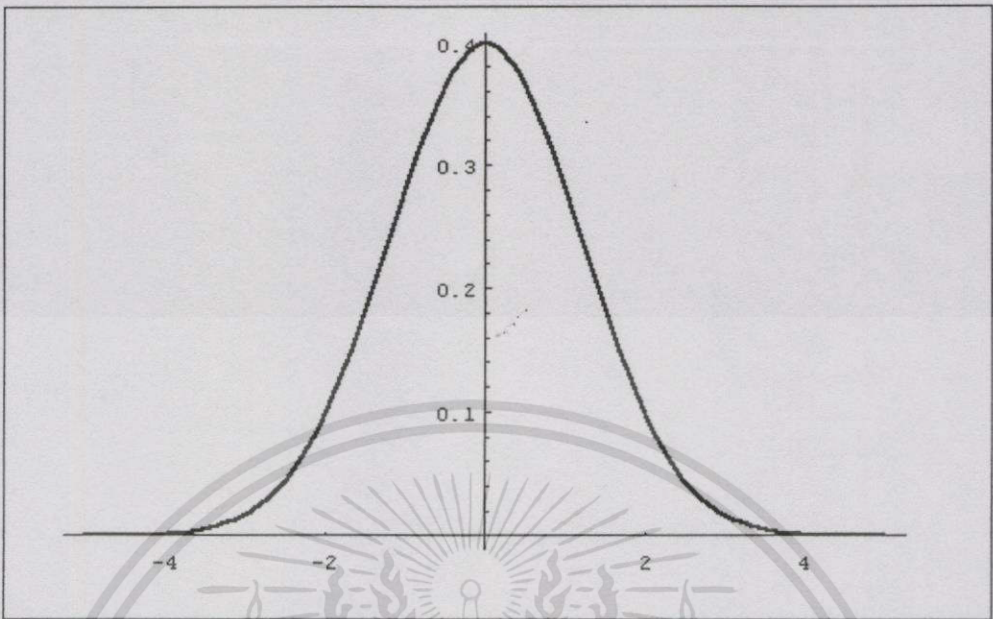
ส่วนเบี่ยงเบนมาตรฐาน (σ) จะหาได้จาก $\sigma = \sqrt{s}$

การคำนวณการจำลองการแอนนีลลิ่งแบบโบสต์มานน์จะอาศัย *pdf* ที่มีการกระจายเป็นแบบ Gaussian ดังสมการที่ 4.13 จะมีลักษณะการกระจายดังแสดงในรูปที่ 4.1

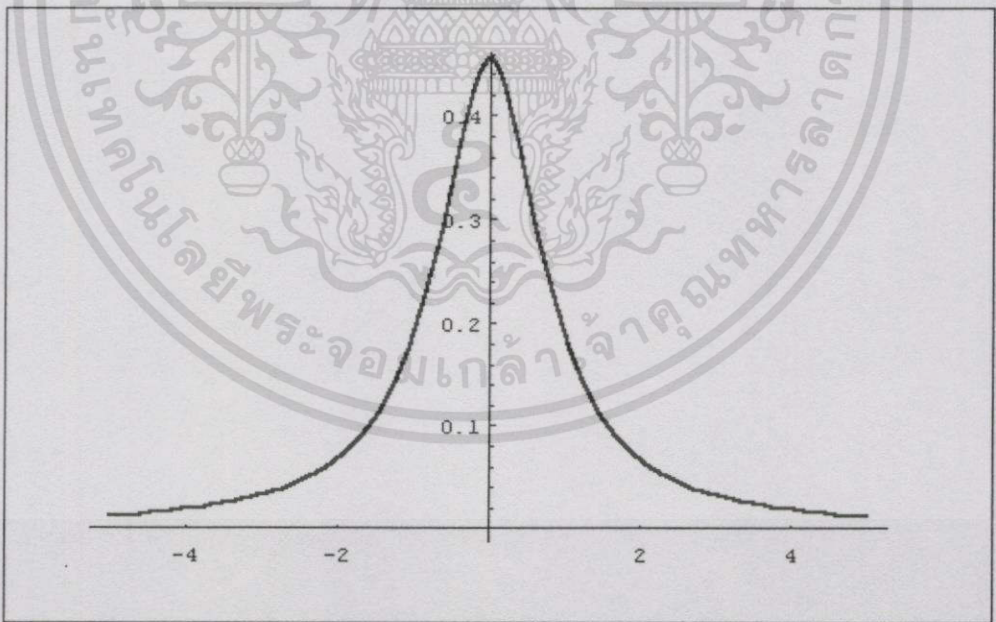
การคำนวณการจำลองการแอนนีลลิ่งแบบเร็วจะอาศัย *pdf* ที่มีการกระจายเป็นแบบ Cauchy ดังสมการที่ 4.15 จะมีลักษณะการกระจายดังแสดงในรูปที่ 4.2

การคำนวณการจำลองการแอนนีลลิ่งแบบเร็วมากจะอาศัย *pdf* ที่มีการกระจายเป็นแบบ VFSSR ดังสมการที่ 4.17 จะมีลักษณะการกระจายดังแสดงในรูปที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

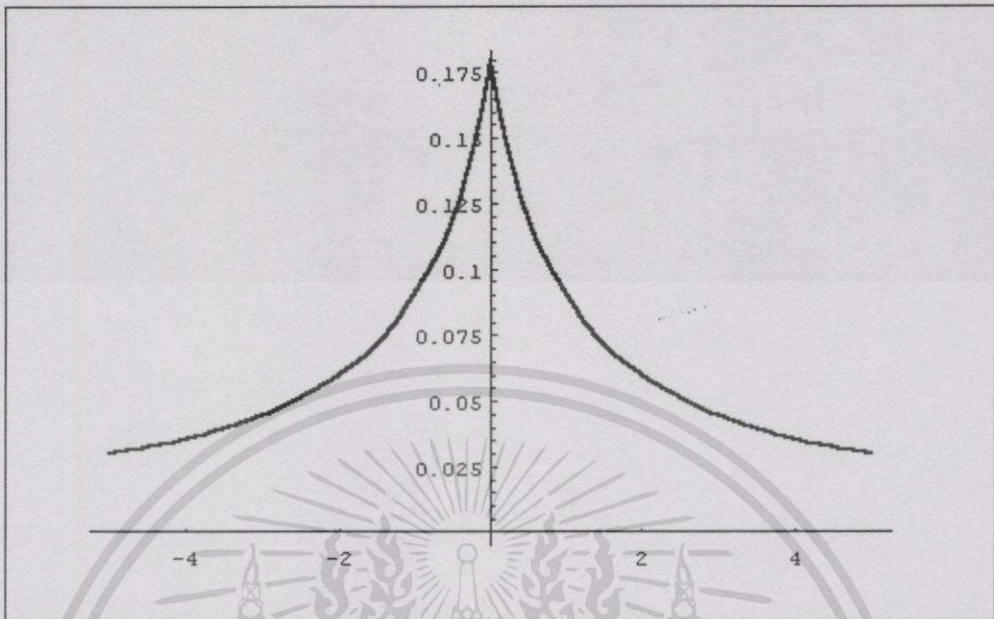


รูปที่ 4.1 แสดงฟังก์ชันการกระจายแบบโบสต์มันน์ (Boltzman Distribution)



รูปที่ 4.2 แสดงฟังก์ชันการกระจายแบบโคชี (Cauchy Distribution)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงฟังก์ชันการกระจายแอนนิลิ่งแบบเร็วมาก VFSR (Very Fast Simulated Annealing)

จากกราฟจะเห็นว่าการกระจายที่มีลักษณะต่อเนื่อง ในการหาค่าตัวแปรที่มีลักษณะไม่ต่อเนื่อง เช่น ปัญหาการจัดวาง จะเป็นการสลับสับเปลี่ยนตำแหน่งของชิ้นส่วนอุปกรณ์อิเล็กทรอนิกส์ ซึ่งมีค่าไม่ต่อเนื่อง จึงไม่ต้องอาศัยฟังก์ชันการกระจาย

4.2 เกณฑ์การกำหนดค่าเริ่มต้น

สำหรับการคำนวณเพื่อให้ได้ผลลัพธ์มีค่าดูเข้าหาค่าที่เป็นค่าที่เหมาะสม จะอาศัยการกำหนดค่าเริ่มต้นให้กับตัวแปรควบคุมซึ่งเปรียบได้กับค่าอุณหภูมิ ในตอนเริ่มต้นการคำนวณการกำหนดค่าเริ่มต้นจะมีความสำคัญต่อผลลัพธ์ที่จะได้เป็นคำตอบ ค่าเริ่มต้นที่จะต้องพิจารณาถึง ประกอบด้วยค่าอุณหภูมิเริ่มต้น T_0 ค่าอุณหภูมิเริ่มต้นจะต้องเป็นค่าที่ถือว่า สำหรับทุก ๆ ลำดับของการเปลี่ยนแปลงสามารถเป็นค่าที่ยอมรับได้ อัตราส่วนการยอมรับ (Acceptance ratio) สำหรับค่าเริ่มต้นจึงควรมีค่าเข้าใกล้ 1 สำหรับค่าเริ่มต้นค่าที่เสนอโดย Kirkpatrick [47] จะมีค่าอัตราส่วนของการยอมรับ เท่ากับ 0.8

ในการหาค่าอุณหภูมิเริ่มต้นจะเปรียบเทียบกับทำให้สสารร้อนขึ้นตามวิธีการแอนนิลลิงในทางฟิสิกส์อนุภาคจะสามารถเคลื่อนที่ไปมาได้ อนุภาคสามารถจะมีค่าอยู่ในสถานะใด ๆ ก็ได้ซึ่งลักษณะนี้จะเปรียบได้กับลักษณะในการคำนวณหาค่าตอบ โดยจะทำการเลือกคำตอบจากสถานะต่าง ๆ ที่เป็นไปได้ขึ้นมาพิจารณา เหมือนกับการเดินสุ่ม (Random Walk) ค่าอุณหภูมิเริ่มต้นควรมีค่าสูงเท่าไรจึงจะพอเหมาะที่จะใช้ในการกำหนดเป็นค่าเริ่มต้นนั้น สามารถหาได้โดยการทำการทดลองหาค่าตัวเลขที่สามารถให้คำตอบที่เป็นค่าที่เหมาะสมได้ (Empirical testing)

เนื่องจากอุณหภูมิหรือค่าเริ่มต้นสำหรับการคำนวณซ้ำเป็นค่าที่เป็นตัวกำหนดจำนวนรอบของการคำนวณ ถ้าค่านี้มีค่าสูงเวลาที่จะใช้ในการจะดูเข้าหาค่าที่เป็นคำตอบก็จะใช้เวลานาน แต่ถ้าให้มีค่าต่ำเกินไปจำนวนรอบในการคำนวณก็จะน้อยทำให้ไม่สามารถหาค่าตอบที่เป็นค่าที่ดีที่สุดของทั้งหมดได้ สำหรับสถานะที่ตัวแปรในการคำนวณหาค่าที่เหมาะสมจะมีค่าเข้าใกล้ค่าที่เป็นค่าต่ำสุดหรือค่าที่เหมาะสมนั้น ค่าส่วนเบี่ยงเบนจะมีค่าลดลงซึ่งแสดงให้เห็นว่าโอกาสที่จะมีการเปลี่ยนจากสถานะหนึ่ง ไปยังอีกสถานะหนึ่ง ทำได้น้อย แต่ถ้าค่าอุณหภูมิหรือตัวกำหนดจำนวนรอบของการคำนวณนี้มีค่าสูง โอกาสที่ค่าตัวแปรที่ทำการหาค่าเหมาะสมจะเปลี่ยนจากสถานะหนึ่ง ไปยังอีกสถานะหนึ่งสามารถทำได้ง่าย จะเห็นว่าถ้าค่าอุณหภูมิเริ่มต้นมีค่าน้อยโอกาสที่จะมีการเปลี่ยนสถานะจะทำได้ยาก

สิ่งที่สำคัญในการคำนวณของการจำลองการแอนนิลลิงก็คือการกำหนดค่าเริ่มต้น ถ้ามีการกำหนดค่าเริ่มต้นไว้สูงเกินไป คำตอบที่ได้ก็จะมีการเปลี่ยนแปลงอย่างมาก พิจารณาได้จากอัตราส่วนของการยอมรับ ถ้าสถานะก่อนหน้าให้คำตอบที่เหมาะสมไว้ คำตอบใหม่ที่ได้ที่อุณหภูมิสูงก็อาจไปแทนที่ผลลัพธ์ที่ได้ก่อนหน้า เพราะที่อุณหภูมิสูงทุกสถานะมีโอกาสถูกเลือกเป็นคำตอบ ซึ่งอาจจะเป็นคำตอบที่ไม่ดีกว่าเดิมก็ได้ แต่ถ้าอุณหภูมิต่ำการยอมรับคำตอบใหม่ก็จะน้อยหรือลดลงจนบางครั้งไม่สามารถกระโดดออกจากบริเวณที่เป็นค่าเฉพาะที่ได้

ค่าอุณหภูมิเมื่อเริ่มต้นการคำนวณสามารถพิจารณาได้จากเงื่อนไขของการยอมรับ ซึ่งสามารถหาความสัมพันธ์ระหว่างอัตราส่วนของความน่าจะเป็นในการยอมรับได้ และอุณหภูมิเริ่มต้นได้ดังนี้

ให้ถือว่าลำดับของการลองผิดลองถูกมีการสร้างค่าพารามิเตอร์ควบคุมที่มีค่าแน่นอนเป็น t และให้ m_1 แทนจำนวนที่เปลี่ยนแปลงในตอนที่ยังค่าใช้จ่ายมีการลดลงในการคำนวณ และ m_2 แทนจำนวนที่เปลี่ยนแปลงไปในตอนที่ฟังก์ชันค่าใช้จ่ายมีค่าเพิ่มขึ้นและถูกยอมรับด้วยความน่าจะเป็น ดังนั้น โดยเงื่อนไขเมโทโพรริส (สมการ 3.3) จะได้ว่าผลรวมของสถานะที่ได้จากการคำนวณจะเท่ากับ

$$m_1 + m_2 \exp\left(-\frac{\Delta f}{t}\right) \quad (4.6)$$

ดังนั้นอัตราส่วนของอัตรากายอมรับเมื่อเริ่มต้น จะมีค่าเท่ากับ

$$\chi = \frac{m_1 + m_2 \exp\left(-\frac{\Delta f}{t}\right)}{m_1 + m_2} \quad (4.7)$$

ซึ่งจะได้ว่า t มีค่าเท่ากับ

$$t = \frac{\overline{\Delta f^+}}{\ln\left(\frac{m_2}{m_2 - m_1(1-\chi)}\right)} \quad (4.8)$$

โดย $\overline{\Delta f^+}$ เป็นค่าเฉลี่ยของทุก ๆ ค่าในขณะที่ฟังก์ชันค่าใช้จ่ายมีค่าเพิ่มขึ้น โดยไม่ต้องพิจารณาถึงตอนที่ฟังก์ชันมีค่าลด จะได้ว่าเราสามารถคำนวณหาค่าอุณหภูมิที่จะใช้เริ่มต้นในการคำนวณการจำลองการแอนนีลลิ่งได้

นอกจากวิธีนี้แล้ว Rose [31] ได้เสนอวิธีการคำนวณค่าอุณหภูมิเริ่มต้นไว้ สำหรับปัญหาวิธีการจัดวาง(Placement Problem) แนวคิดก็คือการพิจารณาถึงสมดุลว่าเป็นสถานะที่ไม่มีการเปลี่ยนแปลงค่าฟังก์ชันค่าใช้จ่าย ดังสมการ

$$E(\Delta C) = 0, \quad (4.9)$$

ให้ $P_{accept}(\Delta C)$ เป็นความน่าจะเป็นที่จะเปลี่ยนแปลงของส่วนต่างของค่าใช้จ่าย ΔC ที่คำนวณได้จากสถานะ ณ ขณะนั้น ถ้าทราบลักษณะการกระจาย และถ้า N เป็นจำนวนที่จะเปลี่ยนจากสถานะปัจจุบัน เราจะได้เงื่อนไขที่สมดุล ดังนี้

$$E(\Delta C) = \int_{-\infty}^{\infty} \Delta C \cdot P(\Delta C) \cdot P_{accept}(\Delta C) d\Delta C = 0, \quad (4.10)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ $P_{accept}(\Delta C)$ เป็นความน่าจะเป็นที่การเปลี่ยนไปด้วยค่าใช้จ่าย ΔC ถูกยอมรับและเกณฑ์การยอมรับจะได้เป็น

$$E(\Delta C) = \int_{-\infty}^{\infty} \Delta C \cdot P(\Delta C) d\Delta C + \int_{-\infty}^{\infty} P_{accept}(\Delta C) d\Delta C = 0 \quad (4.11)$$

$$E(\Delta C) = \int_{-\infty}^{\infty} \Delta C \cdot P(\Delta C) d\Delta C + \int_{-\infty}^{\infty} \Delta C \cdot P(\Delta C) \cdot \exp(-\Delta C/T) d\Delta C = 0 \quad (4.12)$$

ดังนั้น อุณหภูมิที่สมดุลสำหรับการจำลองการแอนนีลลิ่งที่ทราบการกระจาย $P(\Delta C)$ คือ อุณหภูมิที่ $T = T_{equilibrium}$ ที่ทำให้สมการนี้เป็นจริง

ขั้นตอนในการคำนวณอุณหภูมิที่สมดุลมักเรียกส่วนนี้ว่า การจำลองมาตรวัดอุณหภูมิ (Simulated thermometer) เพื่อที่จะทำการสร้างส่วนที่ใช้วัดอุณหภูมินี้เพื่อใช้ในการคำนวณหาสถานะที่สมดุล จะอาศัยเซตของตัวอย่างที่ไม่ต่อเนื่องจำนวน N ตัวอย่าง สำหรับใช้ในการประมาณการกระจายของ ΔC ตลอดการคำนวณ

โดยหลักการแล้วค่า $P(\Delta C)$ จะวัดขณะทำการคำนวณการแอนนีลลิ่ง ในขณะที่ T_{eq} จะวัดเมื่ออุณหภูมิอยู่ที่สมดุล อย่างไรก็ตาม สำหรับการเปลี่ยนแปลงในครั้งที่ i ที่คำนวณได้ จะได้ว่าจำนวนที่เปลี่ยนแปลงที่ได้มาและมีการเปลี่ยนแปลงค่าใช้จ่ายไป ΔC จะเป็น $N \cdot P(\Delta C)$ ซึ่งหมายความว่า การเปลี่ยนแปลงค่าใช้จ่ายที่เฉพาะค่าใดค่าหนึ่งนั้นจะถูกสร้างขึ้นมาด้วยความถี่เดียวกันกับส่วนที่มีอยู่ในการกระจายอย่างต่อเนื่อง ตราบเท่าที่ N มีค่ามากพอ เซตของตัวอย่างที่มีการเปลี่ยนแปลงที่คำนวณได้จะประมาณได้เท่ากับการกระจายของ ΔC ของทุกๆ การเปลี่ยนแปลง จะได้ว่า $P(\Delta C)$ ในสมการคำนวณได้จากการเปลี่ยนแปลงและค่า $E(\Delta C)$

ในการจำลองการวัดอุณหภูมิมเริ่มต้นของปัญหาการจัดวางชิ้นส่วน จะเริ่มด้วยด้วยการจัดวางบล็อกหรือเซลล์ที่มีการเปลี่ยนแปลงอย่างมากของ N ค่าของส่วนที่เปลี่ยนแปลงไปของค่าใช้จ่ายที่สัมพันธ์กับการเปลี่ยนแปลงในแต่ละครั้งจะถูกเก็บไว้เป็นค่า ΔC ไม่ว่าจะถูกยอมรับเป็นคำตอบหรือไม่ก็ตาม ค่าของ $E(\Delta C)$ ก็จะถูกคำนวณ และหาค่าอุณหภูมิที่สมดุลหรือค่าที่ฟังก์ชันค่าใช้จ่ายเป็นศูนย์ เมื่อได้ค่าที่ต้องการแล้ว วิธีการนี้จึงเป็นการหาค่าอุณหภูมิมเริ่มต้นที่เหมาะสม

4.3 วิธีการลดค่าของตัวแปรควบคุม

วิธีการลดค่าตัวแปรควบคุมหรือการลดอุณหภูมิ อาจเรียกอีกอย่างหนึ่งว่า การกำหนดการเย็นตัวลง (Cooling Scheduling) การลดค่าตัวแปรควบคุมเป็นการกำหนดปริมาณความน่าจะเป็นที่จะใช้หาค่าตอบในแต่ละชั้น โดยที่อุณหภูมิสูงจะเป็นส่วนที่มีค่าความน่าจะเป็นในการยอมรับค่าที่ถูกเลือกขึ้นมาเป็นคำตอบสูงกว่าส่วนที่มีอุณหภูมิต่ำ

วิธีการลดค่าของตัวแปรควบคุมของการแอนนีลลิ่งจะไม่ค่อยเปลี่ยนแปลงไปมากนักระหว่างปัญหาหนึ่งกับอีกปัญหาหนึ่ง [34] แต่ก็มีวิธีการที่เสนอวิธีการลดค่าอุณหภูมิแตกต่างกัน นับตั้งแต่มีการเสนอการคำนวณวิธีการจำลองการแอนนีลลิ่ง ดังตารางที่ 4.1

การลดค่าตัวแปรควบคุมหรือการเย็นตัวลงจะเป็นตัวกำหนดให้อุณหภูมิมีค่าต่ำลงไปอย่างไรในขณะทำการคำนวณ จะต้องอาศัยค่าพารามิเตอร์ 4 ค่า ที่จำเป็นในการกำหนดค่าการเย็นตัวลง ประกอบด้วย ค่าอุณหภูมิเริ่มต้น t_0

กฎเกณฑ์ในการลดอุณหภูมิและหาค่าอุณหภูมิที่แต่ละรอบของการคำนวณ t_k

กฎที่ใช้ในการหาความยาว L_k ของลูกโซ่มาคอฟที่แต่ละอุณหภูมิ t_k

เกณฑ์ในการหยุดการคำนวณ ซึ่งจะเป็นตัวกำหนดว่าเมื่อไรจึงจะหยุดการคำนวณ

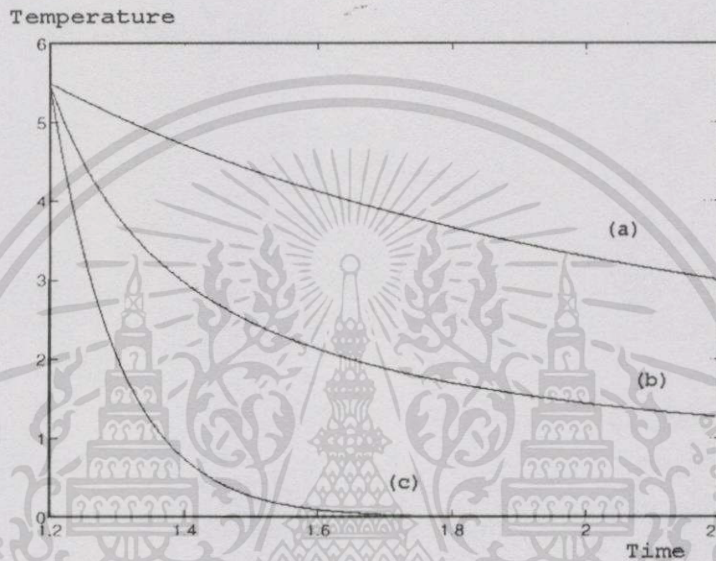
ตารางที่ 4.1 แสดงค่าวิธีการลดค่าอุณหภูมิในแบบต่างๆ

ประเภทของการลดค่าอุณหภูมิ	ค่าของตัวกำหนด	เสนอขึ้นโดย
Logarithm	$T_i = T_0 / \log(t + 2)$	Haario และ Saksman (1991)
Geometric	$T_i = c_i T_0, \quad 0 < c < 1$	Vanderbilt และ Louie (1984)
		Wille และ Vennik (1985)
		Wille (1987)
		Corona (1987)
		Bertocchi และ Sergi (1992)
Linear	$T_i = T_0 / (t + 1)$	Szu และ Hartley (1987)
Function value	$T_i = \alpha f(X_i) + \beta$	Bohachevsky (1986)

ประเภทของการลดค่าตัวแปรควบคุมที่นิยมใช้ในการคำนวณได้แก่การลดค่าแบบเรขาคณิต (Geometric cooling scheduling)

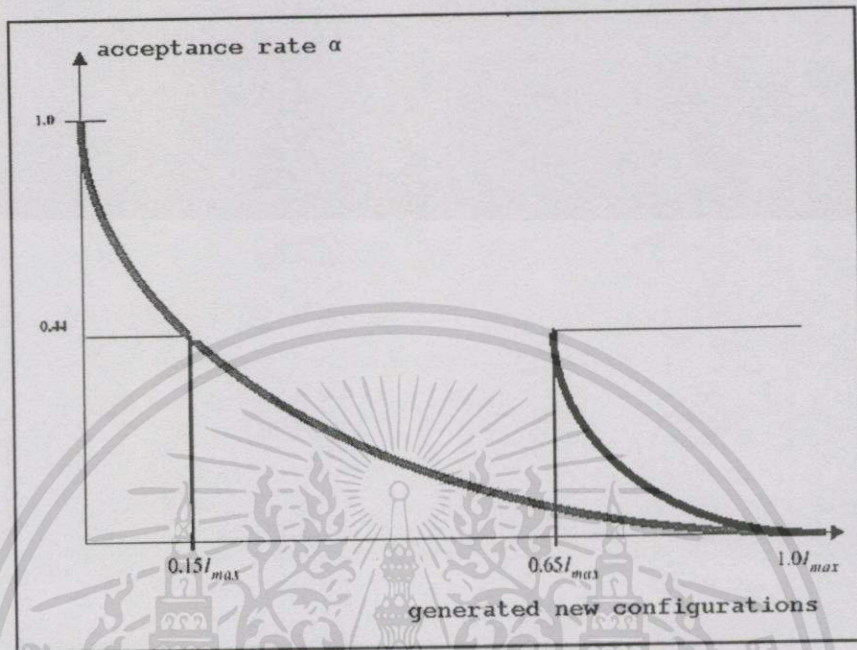
$$T_k = \alpha T_{k-1} = \alpha^k T_0, \quad (38)$$

เมื่อ $\alpha \in [0,1)$ เป็นค่าคงที่ โดยทั่วไปค่านี้จะมีค่าอยู่ระหว่าง $0.8 < \alpha < 0.99$



รูปที่ 4.4 แสดงการลดค่าอุณหภูมิที่แตกต่างกัน 3 ประเภท ได้แก่ (a) Boltzmann Annealing (b) Cauchy Annealing และ (c) Adaptive Annealing

นอกจากการลดอุณหภูมิที่ค่อย ๆ ลดลงจนกระทั่งมีค่าเข้าใกล้ศูนย์แล้ว ก็ยังมีวิธีการลดอุณหภูมิแบบ Cauchy และแบบ Adaptive ดังในรูปที่ 4.5 และการลดอุณหภูมิที่แบ่งเป็นขั้น ได้แก่การลดอุณหภูมิตามแบบของ Lam หรือ Lam Schedule [52] ที่ใช้ในปัญหาการจัดวางชิ้นส่วนอุปกรณ์อิเล็กทรอนิกส์ (Placement Problem)



รูปที่ 4.6 แสดงการลดค่าอุณหภูมิแบบ Lam (Lam Scheduling)

4.4 เกณฑ์การหยุดการคำนวณ

การหยุดการคำนวณสำหรับวิธีการจำลองการแอนนีลิ่งสามารถพิจารณาได้จาก คำตอบหรือผลลัพธ์จากการคำนวณและฟังก์ชันค่าใช้จ่าย เช่น คำตอบที่ได้เป็นค่าที่ทำให้ ค่าของฟังก์ชันค่าใช้จ่ายมีค่าตามที่ต้องการ เช่น กรณีของปัญหาการวางหมากราชินี ถ้าคำตอบที่ได้ทำให้ฟังก์ชันค่าใช้จ่ายมีค่าเป็นศูนย์ แสดงว่าได้คำตอบที่ต้องการแล้วสามารถหยุดทำการคำนวณได้ หรือพิจารณาจากอัตราการเปลี่ยนแปลงของคำตอบว่าไม่มีการเปลี่ยนแปลงอีกแล้ว โดยทั่วไปจะพบว่าในการคำนวณเมื่อใกล้ถึงสมดุล คำตอบจะมีค่าไม่ค่อยเปลี่ยนแปลง นอกจากนี้ก็สามารถใช้การลดค่าของตัวแปรควบคุมเป็นตัวกำหนดว่าจะให้หยุดทำการคำนวณได้ โดยทั่วไปตัวแปรควบคุมหมายถึงค่าที่เปรียบได้กับอุณหภูมิในทางกายภาพ เมื่อทำการลดค่าอุณหภูมิมีค่าเข้าใกล้ศูนย์ก็สามารถหยุดทำการคำนวณได้ โดยสถานะที่หยุดการคำนวณจะถือว่าเป็นสถานะที่ระบบอยู่ในสมดุลในการคำนวณก็จะหมายถึงผลลัพธ์ที่ได้จากการคำนวณซ้ำมีค่าไม่เปลี่ยนแปลง

การเพิ่มความเร็วสำหรับการคำนวณการจำลองการแอนนีลิ่ง

5.1 ประเภทของวิธีการเร่งการคำนวณ

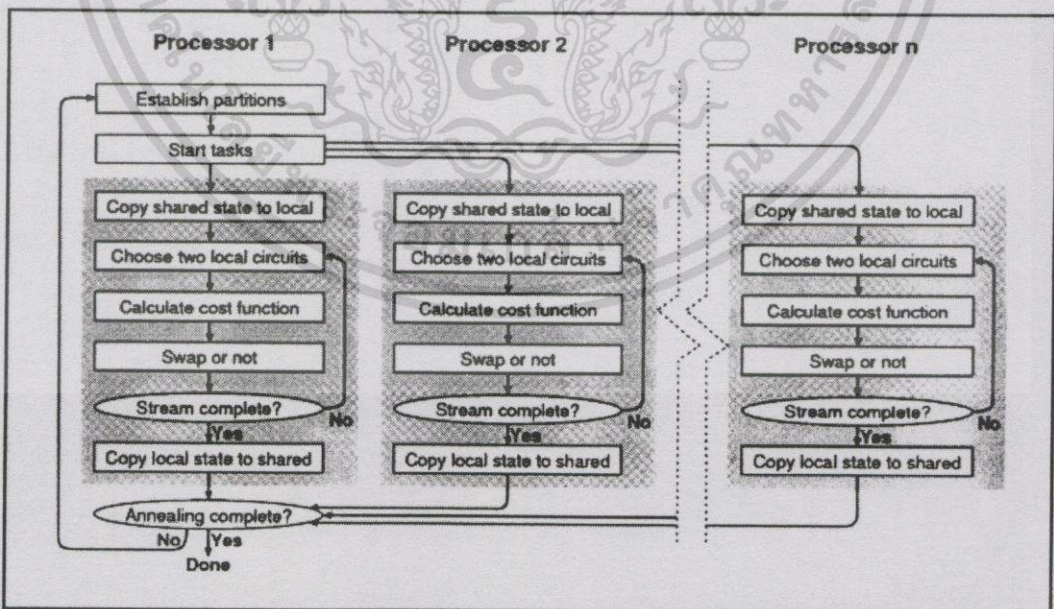
การเร่งการคำนวณการจำลองการแอนนีลิ่งเป็นงานวิจัยที่น่าสนใจตั้งแต่วิธีนี้ถูกเสนอขึ้นมา[4] วิธีการที่ใช้ในการเร่งการคำนวณแบ่งออกเป็น 3 แบบ ด้วยกันคือ

1. การออกแบบวิธีการคำนวณให้เป็นการสามารถทำการคำนวณแบบขนานได้ (Design of parallel algorithm)
2. การสร้างชิ้นเป็น Hardware เฉพาะที่ใช้ในการเร่งการคำนวณ (Hardware acceleration)
3. การออกแบบวิธีการคำนวณในแบบลำดับที่เร็วขึ้น (Design of fast sequential algorithms)

5.1.1 การคำนวณแบบขนาน (Parallel Computing)

หลักการโปรแกรมในลักษณะขนาน (Parallel Programming) จะใช้หน่วยประมวลผลที่มากกว่าหนึ่งตัวในการแบ่งงานกันทำ [4] สำหรับการคำนวณการจำลองการแอนนีลิ่งส่วนที่ใช้เวลานานในการคำนวณคือส่วนใช้ที่พิจารณาหาคำตอบ โดยการเลือกกลุ่มและปฏิเสธคำตอบด้วยความน่าจะเป็น

ในการประมวลผลแบบขนานจะสามารถแบ่งการทำงานได้ โดยการแบ่งส่วนของบริเวณที่ต้องการค้นหาคำตอบให้กับหน่วยประมวลผลแต่ละหน่วย [4] ดังแสดงในรูปที่ 5.1



รูปที่ 5.1 แสดงวิธีการคำนวณการแอนนีลิ่งแบบขนานโดยการแบ่งส่วนสำหรับการหาคำตอบ

ตัวอย่างของการประยุกต์ใช้งานในลักษณะขนานของการจำลองการแอนนิลลิ่ง สำหรับการจัดวางชิ้นส่วนอุปกรณ์อิเล็กทรอนิกส์ ได้แก่ ซอฟต์แวร์ ProperCAD ที่ใช้ในการออกแบบวงจรรวม VLSI เพื่อให้สามารถทำการคำนวณการยอมรับและปฏิเสธตามเงื่อนไขเมโทโพรสได้ในลักษณะขนาน

ProperPLACE-PM()

```

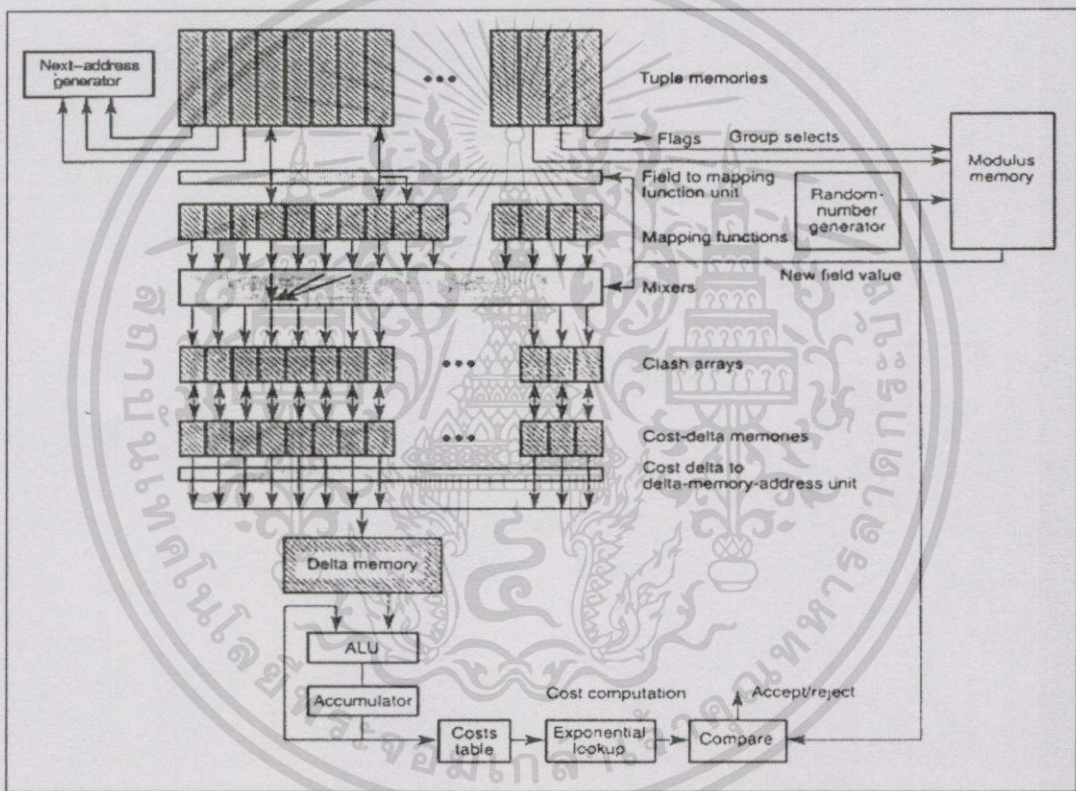
while termination not reached do
  while equilibrium not reached do
    select cell A in this actor's region
    select new location for cell A
    if cell B at new location then
      move is an exchange
      determine move type (M1, M2, M3 or M4)
      evaluate move cost
      if ACCEPT(A,B, move) == yes then
        if move == M4 then
          FREEZE(A,B)
          send AskPermission msg to Circuit
        else
          accept the move
          update local database
          send Update msg to Circuit
    lower T
  
```

รูปที่ 5.2 แสดงวิธีการคำนวณแบบขนานของ ProperPLACE-PM

แบบจำลองการคำนวณแบบขนานยังมีอีกหลายลักษณะ เช่น PVM (Parallel Virtual Machine) และ MPI (Message Passing Interface) ส่วนใหญ่การคำนวณในลักษณะขนานจะไม่ได้เน้นการเปลี่ยนแปลงค่าพารามิเตอร์ของวิธีการคำนวณแต่จะสนใจการทำงานในลักษณะขนานเป็นหลัก โดยใช้อัลกอริทึมเดิม

5.1.2 การสร้างขึ้นเป็น Hardware เฉพาะที่ใช้ในการเร่งการคำนวณ (hardware acceleration)

การสร้างขึ้นเป็น Hardware สำหรับเฉพาะงาน (Special-purpose architecture) สำหรับการคำนวณ การจำลองการแอนนีสลิง Abramson [30] ได้เสนอแบบจำลองของ Hardware ที่ใช้ในการแก้ปัญหาการจัดตาราง (Scheduling Problem) โดยประกอบ ด้วยส่วนต่าง ๆ ดังต่อไปนี้ a next-address generator, tuple memories, a field to mapping-function unit, mapping functions, mixers, clash arrays, cost-delta memories, a cost-delta to delta – memory-address unit, a delta memory, an arithmetic logic unit and accumulator, a random-number generator และ a modulus memory ดังแสดงในรูปที่ 5.3



รูปที่ 5.3 แสดง Schematic Diagram ของ Hardware สำหรับการเร่งความเร็วในการคำนวณการจำลองการแอนนีสลิง

การทำงานยังคงมีลักษณะเป็นแบบลำดับสำหรับการคำนวณการเปลี่ยนแปลงค่าใช้จ่าย แต่อาศัยหน่วยควบคุมโครงสร้างข้อมูล (Data control structures) ที่ต่างไปจากสถาปัตยกรรมของ Von Neumann ทำให้สามารถคำนวณพร้อมกันได้ (concurrent)

5.1.3 การออกแบบวิธีการคำนวณในแบบลำดับที่เร็วขึ้น (Design of fast sequential algorithms)

ในงานวิจัยส่วนใหญ่ที่เลือกปรับปรุงวิธีการคำนวณจะเกี่ยวข้องกับการเลือกการลดอุณหภูมิแบบเร็ว ข้อได้เปรียบของวิธีการนี้ก็คือสามารถทำการคำนวณได้เร็วเนื่องจาก การลดอุณหภูมิเป็นตัวกำหนดระยะเวลาที่ใช้ในการคำนวณ ถ้าใช้การลดอุณหภูมิแบบเร็วระยะเวลาที่ใช้ในการคำนวณก็จะสั้นลง แต่วิธีการในลักษณะนี้ไม่ได้เหมาะสมหรือเฉพาะเจาะจงกับปัญหาใดปัญหาหนึ่งเลย แต่ก็ได้มีการเสนอการลดอุณหภูมิไว้หลายแบบ

สำหรับการเสนอการปรับปรุงวิธีการคำนวณเพื่อลดเวลาที่ใช้ในการคำนวณที่มีการเสนอขึ้นมาได้แก่ การเลือกวิธีการหาค่าฟังก์ชันค่าใช้จ่ายในแบบอื่น (Alternative Cost Evaluation) การสร้างวิธีเปลี่ยนแปลงสถานะแบบอื่น (Alternative Transition Generation) [29] และการเลือกการลดอุณหภูมิที่ใช้เวลาพอเหมาะที่ไม่ใช่แบบทั่วไป (Non-traditional optimal finite-time temperature scheduling) [52] และการรวมกับวิธีการคำนวณอื่น โดยแบ่งหลายขั้นตอนการคำนวณ (Multi-stage)[82]

5.1.3.1 การเปลี่ยนวิธีการหาค่าฟังก์ชันค่าใช้จ่าย

เริ่มศึกษาโดย Grover [31][32] ทำการพิจารณาฟังก์ชันค่าใช้จ่ายที่มีค่าแปรปรวน ที่ต้องอาศัยการประมาณค่าเนื่องจากขาดรายละเอียดของสถานะที่จะนำมาใช้ในการคำนวณ Grover ใช้เทคนิคดังกล่าวกับปัญหาวิธีการจัดวางชิ้นส่วนอุปกรณ์อิเล็กทรอนิกส์สำหรับเซลล์ที่มีขนาดมาตรฐาน (standard cell) ในการคำนวณปัญหาดังกล่าวด้วยวิธีการจำลองการแอนนีลลิ่งจะทำได้โดยการสลับหรือเปลี่ยนตำแหน่งของเซลล์ไปยังจุดอื่น การคำนวณที่เกี่ยวข้องนี้จะใช้เวลาค่อนข้างมาก Grover เสนอว่าไม่จำเป็นต้องยอมรับคำตอบทุก ๆ ค่า ให้เลือกแต่เพียงบางค่าก็ได้ โดยที่ Grover อธิบายว่าไม่จำเป็นต้องปรับปรุงค่าตำแหน่งต่าง ๆ ของทุก ๆ เซลล์เพราะจะมีค่าความคลาดเคลื่อน[31] แต่ใช้หลักการที่วิธีการจำลองการแอนนีลลิ่งสามารถรู้เข้าหาคำตอบได้ช่วย พบว่าผลลัพธ์ที่ได้สามารถเพิ่มความเร็วในการคำนวณได้ 3 ถึง 5 เท่า สำหรับการรู้เข้าของการคำนวณด้วยวิธีการจำลองการแอนนีลลิ่งสำหรับฟังก์ชันค่าใช้จ่ายที่มีค่าผิดพลาด

5.1.3.2 การสร้างการเปลี่ยนสถานะแบบอื่น

การเร่งการคำนวณการจำลองการแอนนีลลิงอีกวิธีหนึ่งคือการสร้างการเปลี่ยนแปลงสถานะคำตอบในแบบที่ต่างไปจากแบบเดิม หรือการเปลี่ยนแปลงฟังก์ชันการยอมรับและฟังก์ชันการยอมรับแบบมีเงื่อนไข Greene และ Supowit [29] ได้กำหนดวิธีการสร้างสถานะคำตอบขึ้นมาใหม่ดังนี้ โดยกำหนดฟังก์ชันเสริมที่ขึ้นกับวิธีการยอมรับแบบเดิมดังนี้

$$\Lambda_{ij}(t) = \min[1, \exp(-\frac{C_{ij}}{t_k})] \quad (5.1)$$

จากนั้นก็ใช้วิธีการเลือกคำตอบในแบบที่ต่างออกไปจากการคำนวณการแอนนีลลิงแบบเดิม โดยให้ขึ้นกับเมตริกซ์การสร้างของเดิม ดังสมการ

$$\forall j \in R: G_{ij}(t) = \frac{\Lambda_{ij}(t)}{\sum_{k \in R} \Lambda_{ij}(t)} \quad (5.2)$$

โดยที่ทุก ๆ การเปลี่ยนแปลงจะมีการยอมรับค่าเมื่อถูกสร้างขึ้นหรือตอนที่เลือกขึ้นเป็นคำตอบ และ $\Lambda_{ij}(t) = 1, \forall i, j \in S, \forall t_k \geq 0$

โดยเกณฑ์ดังกล่าวพิจารณาได้จากข้อเท็จจริงที่ว่าที่อุณหภูมิค่าความน่าจะเป็นของการยอมรับมีค่าน้อยทำให้มีการปฏิเสธสถานะที่สร้างขึ้นมาเป็นส่วนมาก ทำให้ช่วงเวลาคำนวณนานขึ้น เนื่องจากอาจไปเพิ่มความยาวของลูกโซ่มาคอฟ Greene และ Supowit จึงใช้วิธีไม่มีการปฏิเสธ (Rejectionless method) ดังนั้นลูกโซ่มาคอฟที่สร้างขึ้นจากการเปลี่ยนแปลงสถานะของการยอมรับและการเลือกขึ้นเป็นคำตอบ ก็จะจบลงในจำนวนที่เท่ากัน [29] อัลกอริทึมที่แสดงวิธีการคำนวณดังรูปที่ 5.4

วิธีการนี้ได้มีการนำไปใช้งานการคำนวณในแบบขนาน สำหรับปัญหา Ising Sping และปัญหาการเดินทางของนักขาย (Travelling Salesman Problem) โดยผลลัพธ์แสดงให้เห็นว่าสามารถลดเวลาที่ใช้ในการคำนวณลงได้ 3.9 เท่า

```

procedure REJECTIONLESS_ANNEALING( $s, T, N$ )
{
do
  for all possible changes  $\{s_i\}$  of solution  $s$ 
     $\Delta E = E(s_i) - E(s)$ 
    if  $\Delta E_i \leq 0$  then
       $p_i = 1$ 
    else
       $p_i = \exp(-\Delta E / T)$ 
    replace  $s$  with random choice from  $s_i$ , weighted by their respective  $p_i$ 
  for  $N$  times
  return( $s$ )
}

```

รูปที่ 5.4 อัลกอริทึมที่ไม่มีกระบวนการยอมรับคำตอบ (Rejectionless) ที่เสนอโดย Greene และ Supowit

5.1.3.4 การประมาณค่า Exponential

ในการคำนวณการจำลองการแอนนิลลิงจะมีการคำนวณค่า Exponential ซ้ำ ๆ ในส่วนของฟังก์ชันการเลือกที่จะยอมรับคำตอบ และค่าที่ต้องคำนวณจะเป็นค่าของค่าใช้จ่ยซึ่งเป็นตัวเลขที่มีค่ามาก ในบางครั้งขึ้นกับขนาดของเวลาที่ใช้จัดตารางการลดค่าอุณหภูมิและค่าพารามิเตอร์ที่ใช้ เพื่อให้การคำนวณไม่ยุ่งยากและเสียเวลาในส่วนนี้มากนัก ในการคำนวณสามารถใช้ค่าประมาณ เนื่องจากฟังก์ชัน Exponential เป็นฟังก์ชันที่ต่อเนื่อง การใช้การประมาณสามารถเลือกรูปที่ใช้ประมาณได้โดยให้ค่าผิดพลาดน้อย

การเลือกค่าที่ใช้ในการประมาณ เราสามารถเลือกใช้การกระจายอนุกรมเทเลอร์มาใช้ประมาณค่าฟังก์ชันเลขยกกำลัง $\exp(-\delta/T)$ ได้เป็น

$$\text{เมื่อ } \frac{\delta}{T} \ll 1 \qquad 1 - \frac{\delta}{T} \qquad (5.3)$$

นอกจากการใช้วิธีการประมาณแล้วยังสามารถใช้การอ่านค่าจากตาราง (Lookup table) แทนการคำนวณใหม่ในแต่ละครั้ง [22] ขนาดของเวลาที่ลดลงได้เนื่องจากการคำนวณขึ้นกับประเภทของปัญหาและความต้องการความแม่นยำที่จะใช้ในการคำนวณ

ในงานวิจัยโดย Johnson [22] ได้สรุปไว้ว่าสามารถลดเวลาได้ถึง 33 % สำหรับ การใช้การประมาณโดยอนุกรมเทเลอร์ และลดเวลาลงได้อีก 33 % เช่นกันสำหรับปัญหาการแบ่งกราฟ (Graph Partition)

5.1.3.5 การประมาณค่าตัวเลขสุ่ม

ในการสร้างความน่าจะเป็นให้กับวิธีการคำนวณ การสร้างฟังก์ชันสุ่มสามารถสร้างขึ้นได้จาก linear congruential generators [8] ซึ่งเป็นลำดับของตัวเลขนับ Z_1, Z_2, \dots ที่สร้างขึ้นโดยสมการแบบ Recursive

$$Z_i = (aZ_{i-1} + C) \bmod n \qquad (5.4)$$

เมื่อ m เป็นค่า modulus, a เป็นตัวคูณ, c เป็นค่าที่เพิ่มขึ้น และ Z เป็นตัวสร้างที่เป็นเลขนับที่มากกว่าศูนย์ (seed) ค่าของเลขสุ่มที่นำไปใช้ในการคำนวณการจำลองการแอนนิลลิงจะมี 2 ส่วน คือ ในส่วนของฟังก์ชันการเลือกยอมรับคำตอบ โดยอาศัยเกณฑ์การยอมรับแบบเมโทโพรซิสและการเลือกค่าจากบริเวณใกล้เคียงขึ้นมาพิจารณาเป็นคำตอบ โดยการเปรียบเทียบกับคำตอบปัจจุบัน หน้าทีของตัวเลขสุ่มก็ คือทำให้ทุก ๆ ค่ามีโอกาสถูกเลือกขึ้นมาพิจารณาเป็นคำตอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.3.6 การเรียกใช้การค้นหาเฉพาะที่ (Local Search Trigger)

วิธีการที่จะเพิ่มความเร็วในการคำนวณอีกวิธีหนึ่งได้แก่การใช้วิธีการค้นหาเฉพาะที่เพื่อที่จะลดเวลาของการคำนวณด้วยวิธีการแอนนีลลิ่ง อาจทำได้โดยเริ่มต้นคำนวณที่อุณหภูมิต่ำ แต่เนื่องจากที่อุณหภูมิต่ำคำตอบที่ถูกเลือกขึ้นมาพิจารณามักจะไม่ได้ยอมรับเป็นคำตอบ ค่าที่จะได้รับการยอมรับส่วนใหญ่จะเป็นค่าที่ดีกว่าค่าที่เป็นคำตอบในขณะนั้น จากเหตุผลดังกล่าวจึงสามารถหยุดการคำนวณแอนนีลลิ่งก่อนได้และใช้วิธีการค้นหาเฉพาะที่ทำการคำนวณต่อจากการคำนวณการจำลองการแอนนีลลิ่ง เรียกวิธีการนี้ว่า การเรียกใช้การค้นหาเฉพาะที่ (Local Search Trigger)

ในการคำนวณจริงจะทราบได้อย่างไรว่าเมื่อไรจึงจะสมควรหยุดการคำนวณในส่วนแรกก่อนที่จะเรียกใช้วิธีการค้นหาเฉพาะที่ ให้พิจารณาจากอัตราส่วนการยอมรับ ถ้าอัตราส่วนการยอมรับมีค่าน้อยลงกว่าค่าที่กำหนดไว้ ก็จะทำการคำนวณโดย การเรียกการค้นหาเฉพาะที่ช่วยในการคำนวณ ในทางปฏิบัติ เวลาที่ใช้ในการคำนวณอาจลดลงไม่ได้มากนัก ถ้าเริ่มทำการคำนวณในตอนปลายของการคำนวณ

ในทางปฏิบัติการเลือกสถานะเริ่มต้น (Initial State) สามารถใช้วิธีการค้นหาเฉพาะที่หาคำตอบค่าที่เหมาะสมสำหรับใช้เป็นสถานะเริ่มต้นก่อนทำการคำนวณการแอนนีลลิ่งได้ เนื่องจากคำตอบที่ได้จากการหาโดยวิธีการค้นหาเฉพาะที่สามารถลู่เข้าหาคำตอบที่เป็นค่าที่เหมาะสม (Optimum)

บทที่ 6

การประยุกต์ใช้งานการจำลองการแอนนีลลิง

6.1 ปัญหาการเดินทางของนักขาย

ปัญหาการเดินทางของนักขาย (Travelling Salesman Problem) เป็นปัญหาที่ยากต่อการหาคำตอบ ปัญหาหนึ่งเพราะจัดเป็นปัญหาประเภท NP [23] เพราะยังไม่สามารถหาวิธีการคำนวณที่เวลาในการหาคำตอบสามารถเขียนได้ในรูปโพลีโนเมียลของขนาดของปัญหาได้ สำหรับคำตอบที่ได้ของปัญหานี้สามารถนำไปประยุกต์ใช้งานได้ ในหลายลักษณะ เช่น ปัญหาการหาเส้นทาง (Routing Problem) ปัญหาการออกแบบวงจรรวม (VLSI Placement Problem)

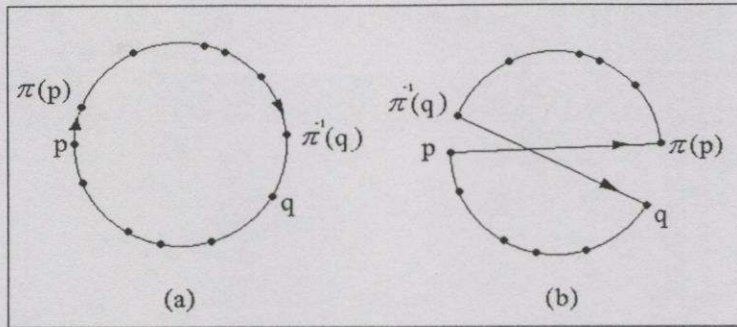
ปัญหามีอยู่ว่า กำหนดเมืองจำนวน n เมือง และให้เมตริกซ์ขนาด $n \times n$ เมตริกซ์ $[d_{pq}]$, แทนระยะทางระหว่างแต่ละคู่ของเมืองสองเมืองจากทั้งหมดจำนวน n เมือง ให้หาเส้นทางที่เดินผ่านไปยังทุก ๆ เมืองเพียงครั้งเดียว โดยใช้ระยะทางสั้นที่สุด [3] สำหรับปัญหานี้คำตอบสามารถหาได้จากการสลับสับเปลี่ยนคำตอบเป็นแบบวนรอบ (cyclic permutation) ดังนี้ $\pi = (\pi(1), \dots, \pi(n))$, เมื่อ $\pi(k)$ แทนเมืองที่อยู่ติดกันกับเมืองที่ k โดยที่ $\pi'(k) = (\pi(1), \dots, \pi(n))$, และ $\pi''(k) = k$ สำหรับทุก ๆ ค่า k แต่ละคำตอบที่ได้จะให้วิธีการเดินที่แตกต่างกัน ดังนั้นเซตของชุดคำตอบจะกำหนดได้โดย

$$S = \{ \text{ทุก ๆ การสลับสับเปลี่ยนค่า } \pi \text{ กับแต่ละเมือง} \}$$

และค่าฟังก์ชันค่าใช้จ่ายจะกำหนดได้โดย

$$f(\pi) = \sum_{i=1}^n d_{i,\pi(i)} \quad (6.1)$$

หมายความว่าฟังก์ชัน $f(\pi)$ จะให้ผลลัพธ์เป็นระยะทางที่สอดคล้องกับการสลับสับเปลี่ยนแบบที่ π ดังนั้นเราจะได้ว่า จำนวนคำตอบหรือวิธีการเดินจะมีได้เท่ากับ $|S| = (n-1)!$ วิธี จะเห็นว่าจำนวนวิธีที่เป็นไปได้มีจำนวนมาก คำตอบของวิธีการเดินทางของนักขายสามารถหาได้จากการสลับสับเปลี่ยนตำแหน่งของวิธีการเดินโดยใช้วิธี 2-change neighborhood operator ดังรูปที่ 6.1



รูปที่ 6.1 แสดงลักษณะวิธีการสลับสับเปลี่ยนวิธีการเดินทาง สำหรับปัญหาการเดินทางของนักขาย

6.1.1 ฟังก์ชันค่าใช้จ่ายของปัญหาการเดินทางของนักขาย

จากสมการ ฟังก์ชันค่าใช้จ่ายของปัญหาการเดินทางของนักขายสามารถพิจารณาได้ดังนี้ คำตอบใหม่จะถูกสร้างจากการเลือกเมืองใด ๆ สองเมือง p และ q และเปลี่ยนลำดับที่อยู่ระหว่างเมืองที่ p และเมืองที่ q จะต้องเดินทาง ค่าผลต่างของฟังก์ชันค่าใช้จ่ายคำนวณได้จากสมการที่ 6.2 จะได้ว่า

$$\Delta f = -d_{p,\pi(p)} - d_{q,\pi^{-1}(q)} + d_{p,\pi^{-1}(q)} + d_{\pi(p),q} \quad (6.2)$$

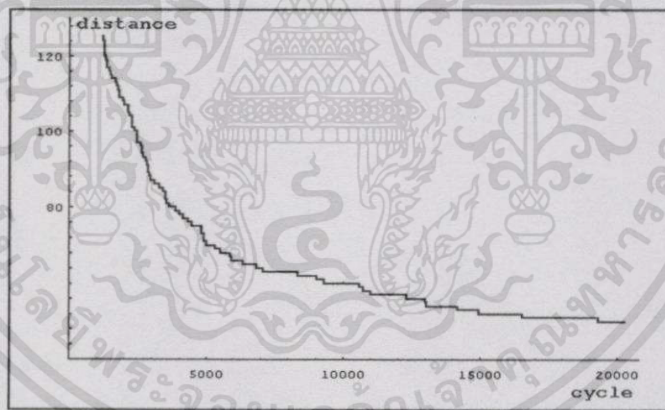
- เมื่อ $\pi(p)$ หมายถึงเมืองที่อยู่ถัดจากเมือง p
 $\pi^{-1}(q)$ หมายถึงเมืองก่อนที่จะไปยังเมือง q
 Δf แทนผลต่างของฟังก์ชันค่าใช้จ่าย f_1 และ f_2 ในการสลับเส้นทางเดิน
 $d_{p,\pi(p)}$ แทนระยะทางจากเมือง p ไปยังเมือง $\pi(p)$
 $d_{q,\pi^{-1}(q)}$ แทนระยะทางจากเมือง q ไปยังเมือง $\pi^{-1}(q)$
 $d_{p,\pi^{-1}(q)}$ แทนระยะทางจากเมือง p ไปยังเมือง $\pi^{-1}(q)$
 $d_{\pi(p),q}$ แทนระยะทางจากเมือง $\pi(p)$ ไปยังเมือง q

ให้ f_1 เป็นฟังก์ชันค่าใช้จ่ายในการเดินทางจากเมือง p ไปยังเมือง $\pi(p)$ และจากเมือง $\pi(p)$ ไปยังเมือง $\pi^{-1}(q)$ และจากเมือง $\pi^{-1}(q)$ ไปยังเมือง q แล้วกลับไปยังเมือง p และ f_2 เป็นฟังก์ชันค่าใช้จ่ายในการเดินทางจากเมือง p ไปยังเมือง $\pi^{-1}(q)$ และจากเมือง $\pi^{-1}(q)$ ไปยังเมือง $\pi(p)$ และจากเมือง $\pi(p)$ ไปยังเมือง q แล้วกลับไปยังเมือง p ซึ่งเป็นการสลับเส้นทางเดินเมื่อเทียบกับ f_1 ผลลัพธ์ที่ได้จากการคำนวณดังแสดงในรูปที่ 6.2 จะเห็นว่าระยะทางรวมมีค่าลดลง

6.1.2 วิธีการสร้างการจำลองการแอนนิลิ่งเพื่อแก้ปัญหาการเดินทางของนักขาย

สำหรับปัญหาการเดินทางของนักขาย ฟังก์ชันค่าใช้จ่ายสามารถคำนวณได้จากระยะทางระหว่างเมืองสองเมืองใด ๆ p และ q โดยเปรียบเทียบค่ากับระยะทางระหว่างเมืองที่อยู่ติดกันระหว่างเมืองทั้งสอง คือเมืองที่ p และ $p+1$ และเมืองที่ q และ $q-1$ โดยเมืองที่ $p+1$ จะแทนได้โดย $\pi(p)$ และเมืองที่ $q-1$ แทนโดย $\pi^{-1}(q)$ ดังแสดงในรูปที่ 6.1 ถ้าฟังก์ชันค่าใช้จ่ายมีค่าลดลง ก็จะทำการเปลี่ยนเส้นทางของเมืองที่จะเดิน แต่ถ้าฟังก์ชันมีค่าเพิ่มขึ้นก็จะยอมรับเป็นเส้นทางใหม่ด้วยความน่าจะเป็น $\exp(-d_{pq})$ เมื่อ d_{pq} แทนระยะทางระหว่างเมือง p และ q ในการคำนวณจะพิจารณาทีละเส้นทางย่อย ๆ ในการเดินจนกระทั่งคำตอบที่ได้ไม่มีการเปลี่ยนแปลงหรือค่าอุณหภูมิที่ใช้กำหนดการเย็นตัวลงมีค่าลดลงและมีค่าน้อยมาก

การจำลองการแอนนิลิ่งเป็นการคำนวณแบบสุ่มที่ใช้หลักการกระจายทางสถิติและความน่าจะเป็น ในการคำนวณการคำนวณจะพิจารณาฟังก์ชันค่าใช้จ่ายเป็นหลัก ฟังก์ชันที่ใช้กำหนดค่าใช้จ่ายสำหรับปัญหานี้คือผลรวมของระยะทางรวมทั้งหมด โดยในการคำนวณผลรวมของระยะทางจะมีค่าลดลงดังรูปที่ 6.2 จนกระทั่งถึงตำแหน่งที่สมดุลของการคำนวณซึ่งจะเป็นตำแหน่งที่คำตอบที่ได้จากการคำนวณมีค่าไม่เปลี่ยนแปลงดังรูปที่ 6.3 โดยมีขั้นตอนการคำนวณดังรูปที่ 6.5

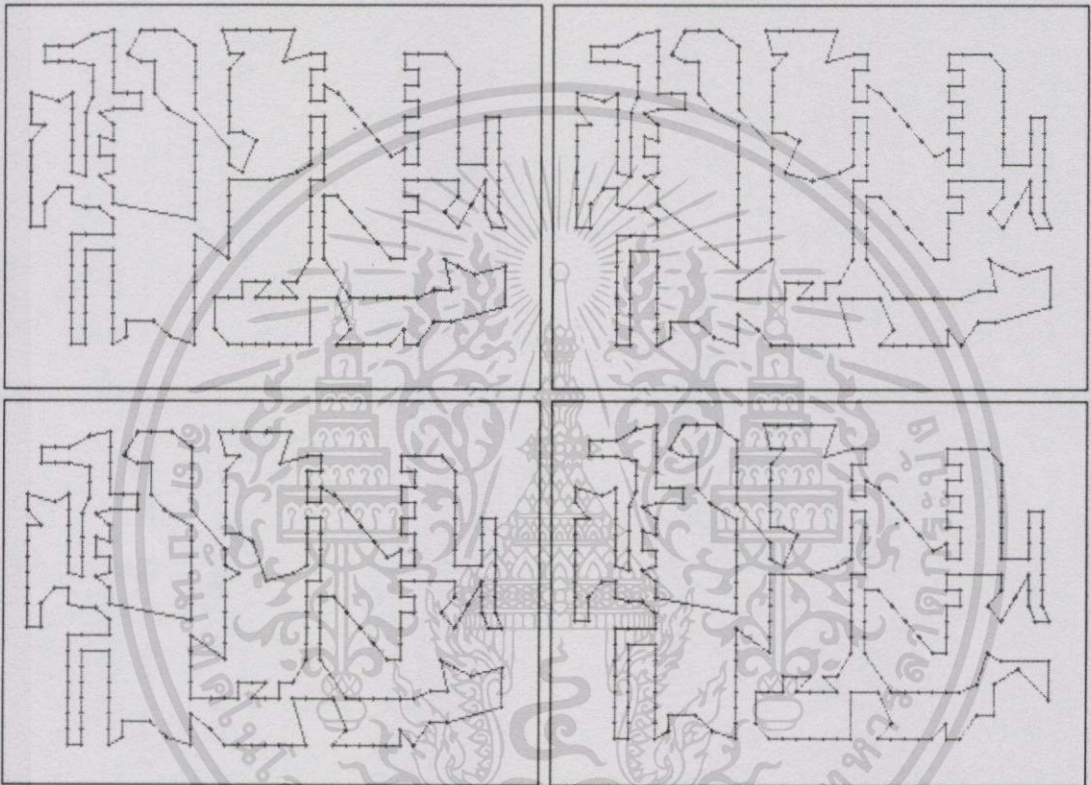


รูปที่ 6.2 แสดงกราฟฟังก์ชันค่าใช้จ่ายต่อจำนวนรอบของการคำนวณของปัญหาการเดินทางของนักขาย สำหรับจำนวนเมืองขนาด 123 เมือง



รูปที่ 6.3 แสดงคำตอบของปัญหาการเดินทางของนักเรียนสำหรับจำนวนเมืองขนาด 123

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.4 แสดงคำตอบของปัญหาการเดินทางของนักขวย สำหรับปัญหา drill280 จาก TSPLIB

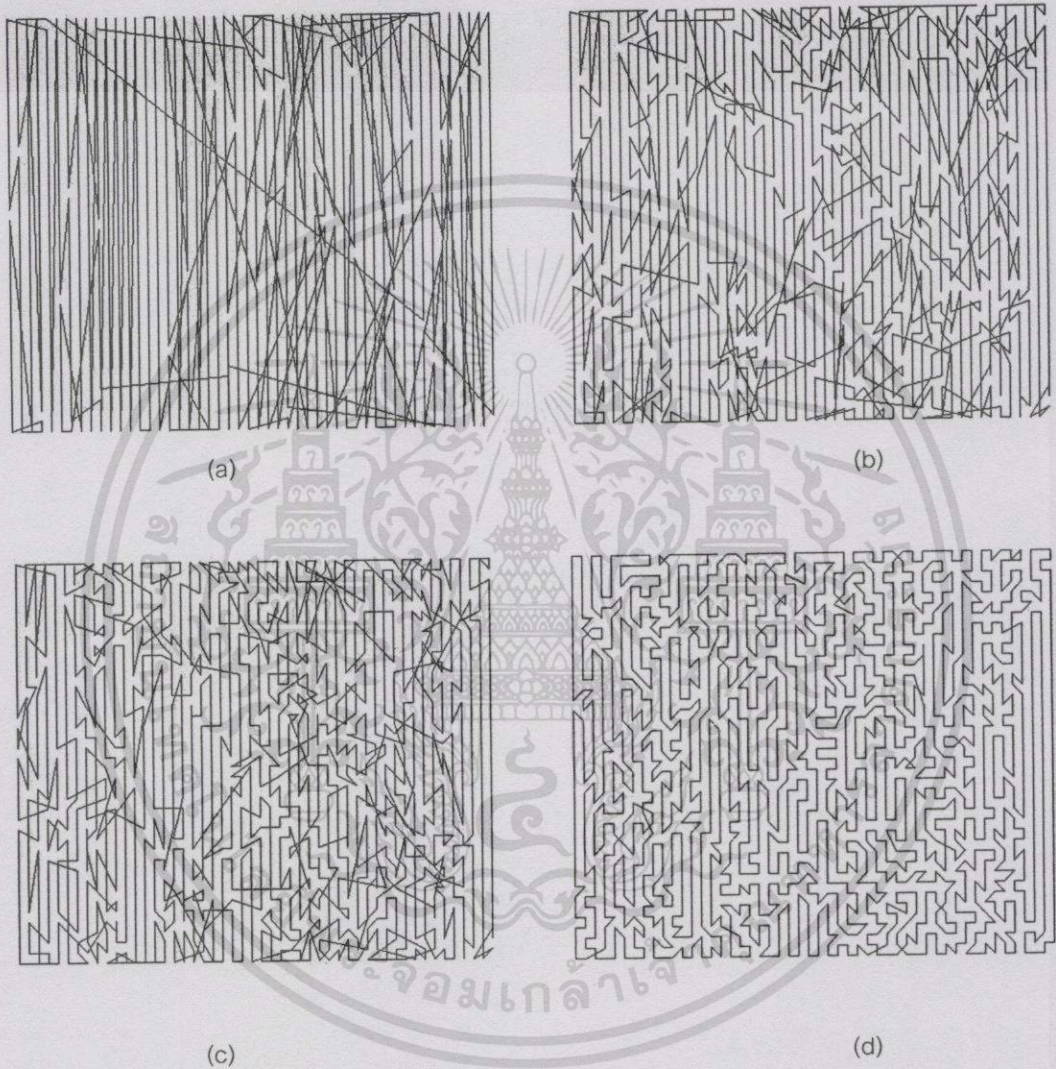
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

T = StartingTemperature()
While not Equilibrium do
{
  swap_count = 0
  Random selected pair of cities
  Calculate distance between cities
  for k := 1 to number of cities do
  Begin
    Swap cities
    Evaluate cost function
    If (distance decrease) then
      Accept swap
    else
      Probabilistic accept
    else
      Reject swap
    swap_count++
  } /* End while */
  T = Adjust(T)
End While

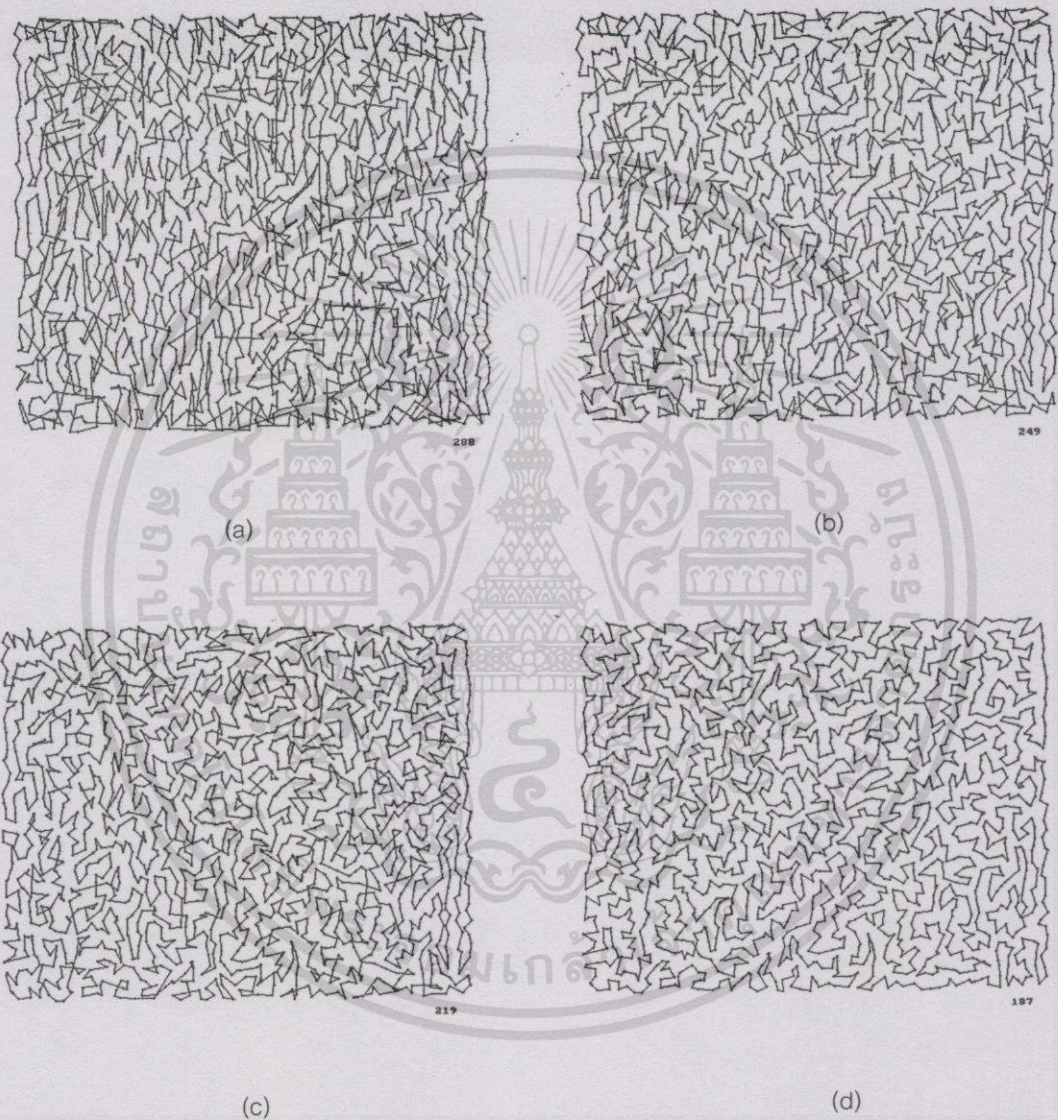
```

รูปที่ 6.5 แสดงวิธีการหาคำตอบของปัญหาการเดินทางของนักขาย โดยการจำลองการแอนนีลลิ่ง



รูปที่ 6.6 แสดงคำตอบของปัญหาการเดินทางของนักขายสำหรับจำนวนเมืองขนาด 2000 เมือง โดยเริ่มตั้งการคำนวณจาก (a) (b) (c) และ (d) ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

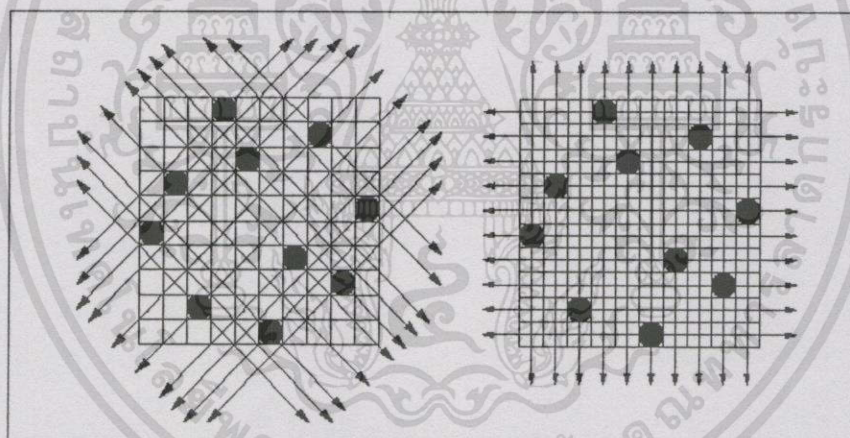


รูปที่ 6.7 แสดงคำตอบของปัญหาการเดินทางของนักเรียนสำหรับจำนวนเมืองขนาด 2000 เมือง โดยตำแหน่งของเมืองกำหนดขึ้นโดยวิธีสุ่ม (Random) เริ่มต้นการคำนวณจาก (a) (b) (c) และ (d) ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 ปัญหาการวางหมากราชินี

ปัญหาทั่วไปที่สามารถจัดให้อยู่รูปของปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน หรือ Combinatorial Optimization นั้นมักจะมีข้อกำหนดที่คำตอบของปัญหาต้องสอดคล้องกับข้อบังคับ (Constraint) ตัวอย่าง ปัญหาประเภทนี้ได้แก่ ปัญหาการวางหมากราชินีจำนวน N ตัว (N-queen Problem) ปัญหานี้เป็นปัญหาดั้งเดิม (Classical Problem) [14] ที่ถูกใช้เป็นปัญหาทดสอบประสิทธิภาพของวิธีการคำนวณหลาย ๆ วิธี ปัญหามีอยู่ว่าต้องการวางหมากราชินีหรือ Queen จำนวน N ตัว ลงบนกระดานขนาด $N \times N$ โดยมีเงื่อนไขว่าหมากแต่ละตัวจะต้องไม่อยู่ในตำแหน่งที่จะกินกัน (attack) คือทั้งในแนวตั้งและแนวนอนและแนวทะแยงที่ตรงกัน ปัญหานี้เป็นปัญหาที่เสนอโดย Gauss การหาคำตอบของปัญหานี้มีวิธีการคำนวณอยู่หลายวิธี เช่น การหาแบบย้อนกลับหรือ (Backtracking) ซึ่งเป็นวิธีการหาคำตอบแบบดั้งเดิม สำหรับการหาคำตอบโดยใช้วิธีการใหม่ที่มีการนำมาใช้ ได้แก่ การหาคำตอบโดยวิธีพันธุศาสตร์ (Genetic Algorithm) การหาคำตอบโดยใช้เครือข่ายประสาท (Neural Network) และการหาคำตอบโดยพิจารณาในรูปการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน โดยใช้การจำลองการแอนนิลลิ่ง (Simulated Annealing) [7]



รูปที่ 6.8 แสดงตัวอย่างคำตอบของการวางหมากราชินีที่จะต้องไม่ให้มีการชนกันของหมากในแนวที่กำหนด

6.2.1 วิธีการดั้งเดิมที่ใช้ในการแก้ปัญหา

วิธีการหาคำตอบของปัญหาการวางหมากราชินีแบบดั้งเดิมคือการคำนวณโดยอาศัยการย้อนกลับเส้นทางเดิม (Backtracking) โดยวิธีการคำนวณแสดง ดังรูปที่ 6.9

```

procedure Try(i :integer);
var k :integer;
{
  for k := 1 to m do
    select kth candidate;
    if acceptable then record it;
    if i < n then
      Try(i + 1)
    else
      Print solution
    End;
    Cancel recording
  }
}

```

รูปที่ 6.9 แสดงวิธีการหาคำตอบของปัญหาการวางหมากราชินีโดยการย้อนกลับเส้นทางเดิม

วิธีการนี้จะมีข้อจำกัดหลายอย่าง และไม่สามารถใช้หาคำตอบสำหรับปัญหาที่มีรายการจำนวนมากได้ เนื่องจากลักษณะการโปรแกรมเป็นแบบเรียกตัวเอง (Recursive) ทำให้ไม่สามารถจัดเก็บรายการจำนวนมากในหน่วยความจำได้ และโครงสร้างข้อมูลสำหรับการเรียกตัวเองมักอยู่ในรูป Stack ทำให้มีขนาดจำกัด

6.2.2 วิธีการสร้างขั้นตอนการคำนวณ

วิธีการหาคำตอบของการวางหมากราชินีบนกระดาน สามารถทำได้โดยการแทนกระดานในรูปของเมตริกซ์หรืออะเรย์ วิธีการนี้จะยุ่งยากตรงที่ต้องคอยตรวจสอบตำแหน่งหรือแนวที่หมากจะกินกันในการคำนวณ จากนิยามของปัญหา เราทราบว่าต้องวางหมากอย่างน้อยหนึ่งตัวในแต่ละคอลัมน์ ซึ่งจะได้โครงสร้างข้อมูลดังนี้

```
VAR x : ARRAY [1..N] OF INTEGER;
    A : ARRAY [1..N] OF BOOLEAN;
    B : ARRAY [b1..b2] OF BOOLEAN;
    C : ARRAY [c1..c2] OF BOOLEAN;
```

เมื่อ x_i แทนตำแหน่งของหมากในคอลัมน์ที่ i

a_j หมายความว่าหมากราชินีอยู่ในแถวที่ j หรือไม่

b_k หมายความว่าหมากราชินีอยู่ในแนวทะแยงที่ k ทางขวาหรือไม่

c_k หมายความว่าหมากราชินีอยู่ในแนวทะแยงที่ k ทางซ้ายหรือไม่

ดังนั้น การวางหมากสามารถแทนได้โดย

$$x[i] := j; a[j] := \text{FALSE}; b[i+j] := \text{FALSE}; c[i-j] := \text{FALSE};$$

และการย้ายหมากสามารถแทนได้โดย

$$a[j] := \text{TRUE}; b[i+j] := \text{TRUE}; c[i-j] := \text{TRUE}$$

เงื่อนไขที่บอกได้ว่าไม่มีหมากตัวอื่นอยู่ในแนวเดียวกันกำหนดได้จาก

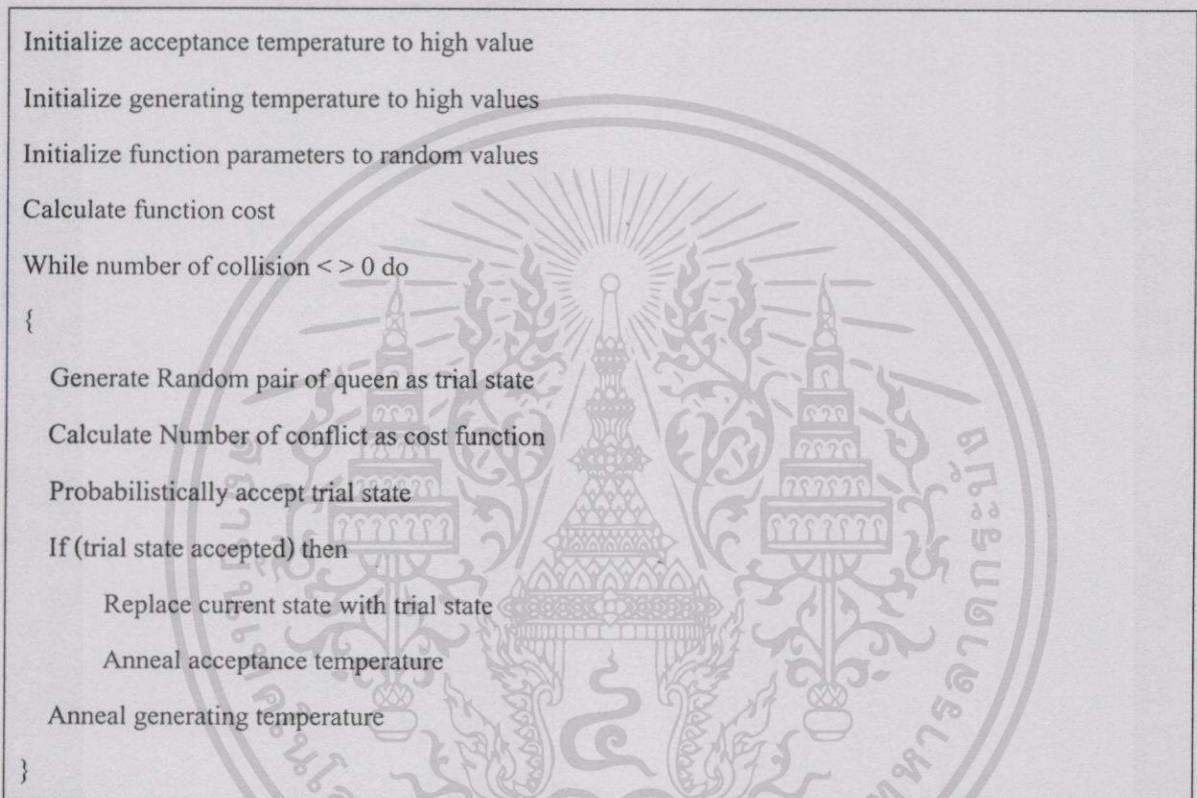
$$a[j] \ \&\& \ b[i+j] \ \&\& \ c[i-j]$$

จากกระดานขนาด $n \times n$ จะสังเกตได้ว่าผลต่างของตำแหน่งของคอลัมน์และผลต่างของตำแหน่งแถวในแนวทะแยงในด้านที่ความชันมีค่าเป็นลบมีค่าคงที่ ส่วนผลบวกของตำแหน่งคอลัมน์และผลบวกของตำแหน่งแถวในแนวแนวทะแยงในด้านที่ความชันเป็นบวกก็มีค่าคงที่ สามารถนำมาใช้ตรวจสอบการชนกันในแนวทะแยงทั้งสองแนวได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2.3 การคำนวณหาค่าตอบโดยการจำลองการแอนนีลลิ่ง

การจำลองการแอนนีลลิ่งเป็นการคำนวณแบบสุ่มที่ใช้หลักการกระจายทางสถิติและความน่าจะเป็น ในการคำนวณการคำนวณจะพิจารณาฟังก์ชันค่าใช้จ่ายเป็นหลัก ฟังก์ชันที่ใช้กำหนดค่าใช้จ่ายสำหรับปัญหานี้คือจำนวนการชนกันของหมากราชินี โดยตำแหน่งที่สมดุลของการคำนวณจะเป็นตำแหน่งที่จำนวนการชนกันของหมากเป็นศูนย์ โดยมีขั้นตอนการคำนวณดังรูปที่ 6.10



รูปที่ 6.10 แสดงขั้นตอนการคำนวณการจำลองการแอนนีลลิ่งสำหรับปัญหาการวางหมากราชินี

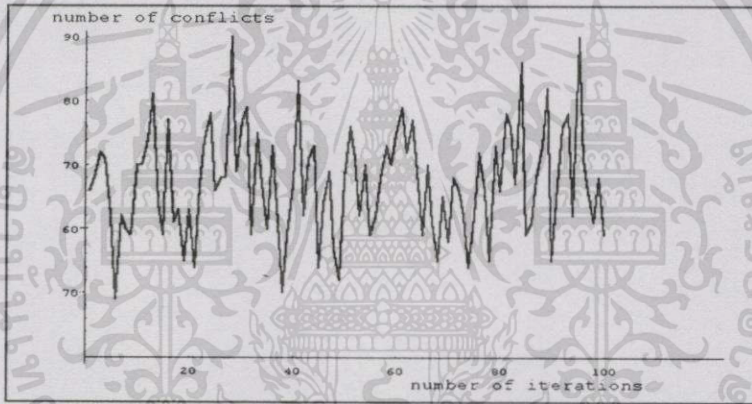
6.2.4 ฟังก์ชันค่าใช้จ่ายของการวางหมากราชินี

ในการคำนวณการจำลองการแอนนีลลิ่งเพื่อแก้ปัญหาการวางหมากราชินี ฟังก์ชันค่าใช้จ่ายจะเป็นจำนวนหมากที่มีการชนกันในการวางแต่ละครั้ง ในการคำนวณการจัดวางตำแหน่งที่สมดุลของปัญหานี้คือจุดที่ทำให้ฟังก์ชันค่าใช้จ่ายมีค่าเป็นศูนย์คือไม่มีการกินกันของหมากตัวใดตัวหนึ่งเลย วิธีการคำนวณทำได้โดยกำหนดให้ชุดของคำตอบให้อยู่ในรูปของการเรียงของค่าอะเรย์เชิงเส้นของจำนวนหมากทั้งหมด n ตัว โดยจะไม่วางจำนวนหมากมากกว่าหนึ่งตัวในแต่ละแถว ในการกำหนดวิธีการสำหรับจัดวางแบบนี้สามารถตัดปัญหาการกินกันในแนวตั้งและแนวนอนไปได้ ปัญหาจะเหลือเพียงการหาวิธีการวางในแนวทะแยง

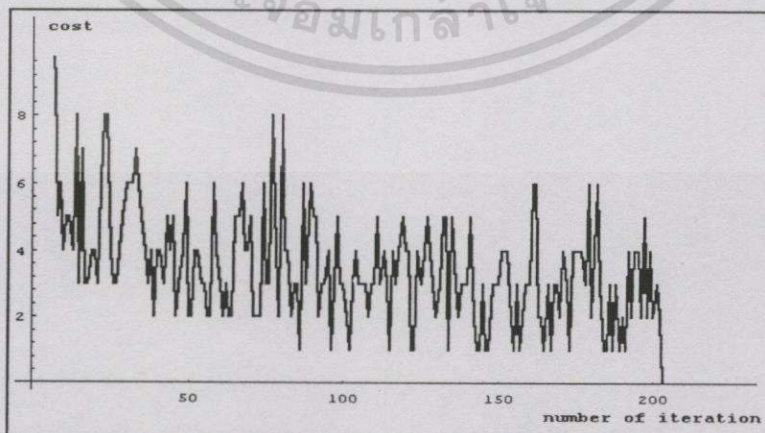
ในการหาการกั้นกันระหว่างหมากราชินีตัวที่ i และตัวที่ j ใน tuple หมากตัวที่ i จะอยู่ในแถวที่ q_i และหมากตัวที่ j จะอยู่ในแถวที่ q_j หมากสองตัวนี้จะอยู่ในแนวทะแยงเดียวกันก็ต่อเมื่อ

$$i - q_i = j - q_j \text{ หรือ } i + q_i = j + q_j \quad (6.1)$$

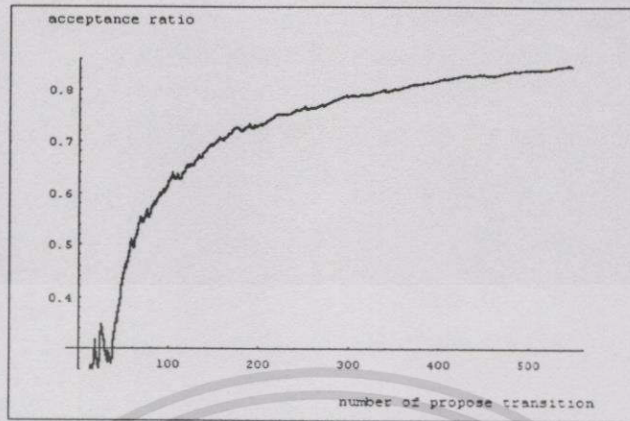
ในการคำนวณฟังก์ชันค่าใช้จ่ายสามารถคำนวณจำนวนได้จากการชนกันของหมากแต่ละตัวซึ่งคำนวณได้จากสมการที่ 6.1 ถ้าทำการสลับตำแหน่งเพียงอย่างเดียว (Random permutation) จะได้ค่าฟังก์ชันค่าใช้จ่าย ดังรูปที่ 6.11 ซึ่งจะมีจำนวนการชนกันของหมากอยู่เท่ากับค่าเฉลี่ยค่าหนึ่ง ซึ่งใช้เป็นสถานะเริ่ม (Initial state) ในการคำนวณการแอนนีลลิ่งได้ หลังจากนั้นจึงทำการคำนวณด้วยการจำลองแอนนีลลิ่ง จะได้ผลลัพธ์ ดังแสดงในรูปที่ 6.12



รูปที่ 6.11 แสดงฟังก์ชันค่าใช้จ่ายสำหรับปัญหาการวางหมากราชินี กรณี Random permutation



รูปที่ 6.12 แสดงฟังก์ชันค่าใช้จ่ายสำหรับปัญหาการวางหมากราชินีเมื่อทำการคำนวณการแอนนีลลิ่ง เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

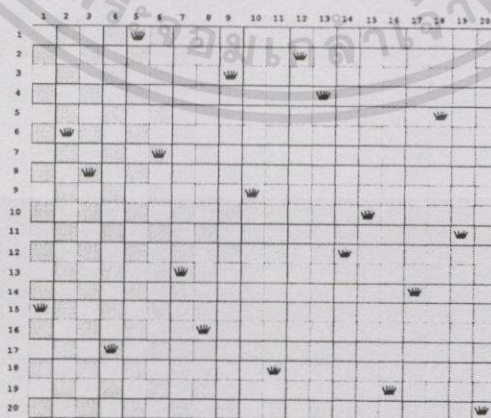


รูปที่ 6.13 แสดงอัตราส่วนการยอมรับต่อจำนวนรอบของการคำนวณสำหรับปัญหาการวางหมากราชินี

จากกราฟแสดงอัตราส่วนการยอมรับต่อจำนวนรอบในการคำนวณของปัญหาการวางหมากราชินี อัตราส่วนการยอมรับคำตอบหมายถึงอัตราส่วนระหว่างจำนวนคำตอบที่ได้รับการยอมรับต่อจำนวนที่เลือกขึ้นมาเป็นคำตอบทั้งหมด จะพบว่าอัตราส่วนในการยอมรับคำตอบจากการเลือกในแต่ละครั้งค่อย ๆ เพิ่มขึ้นจนมีค่าคงที่

6.2.5 การกำหนดการเย็นตัวสำหรับปัญหาการวางหมากราชินี

สำหรับการกำหนดการลดอุณหภูมิของวิธีการคำนวณในปัญหานี้ ใช้การลดอุณหภูมิแบบเรขาคณิต (Geometric cooling schedules) โดยใช้ค่า $\alpha = 0.9$ ในการคำนวณ



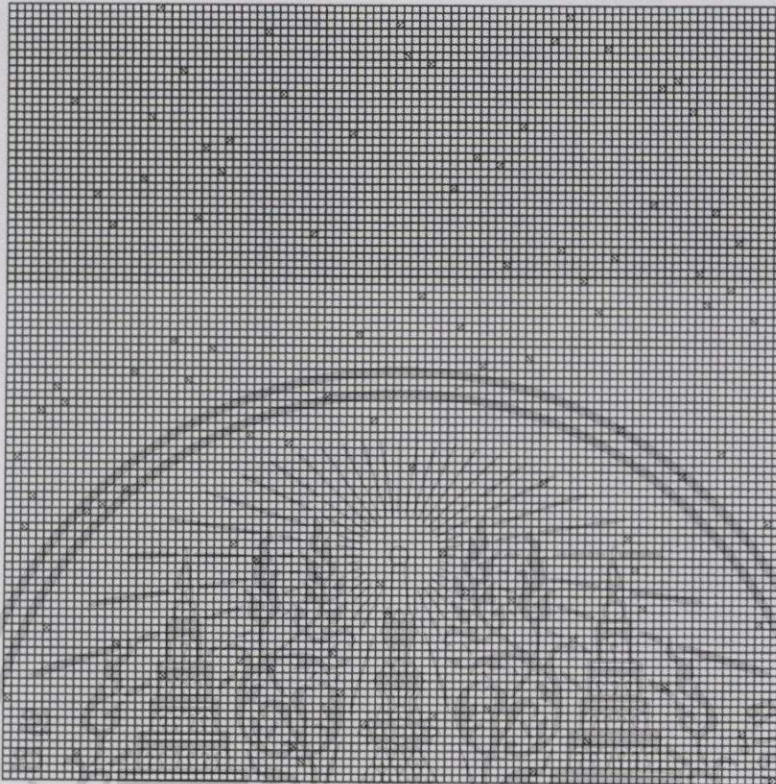
รูปที่ 6.14 แสดงคำตอบสำหรับการวางหมากราชินีบนกระดานขนาด 20x20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.15 แสดงคำตอบของการวางหมากรากินีสำหรับกระดานขนาด 40 x 40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.16 แสดงคำตอบของการวางหมากราชินีสำหรับกระดานขนาด 100 x 100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

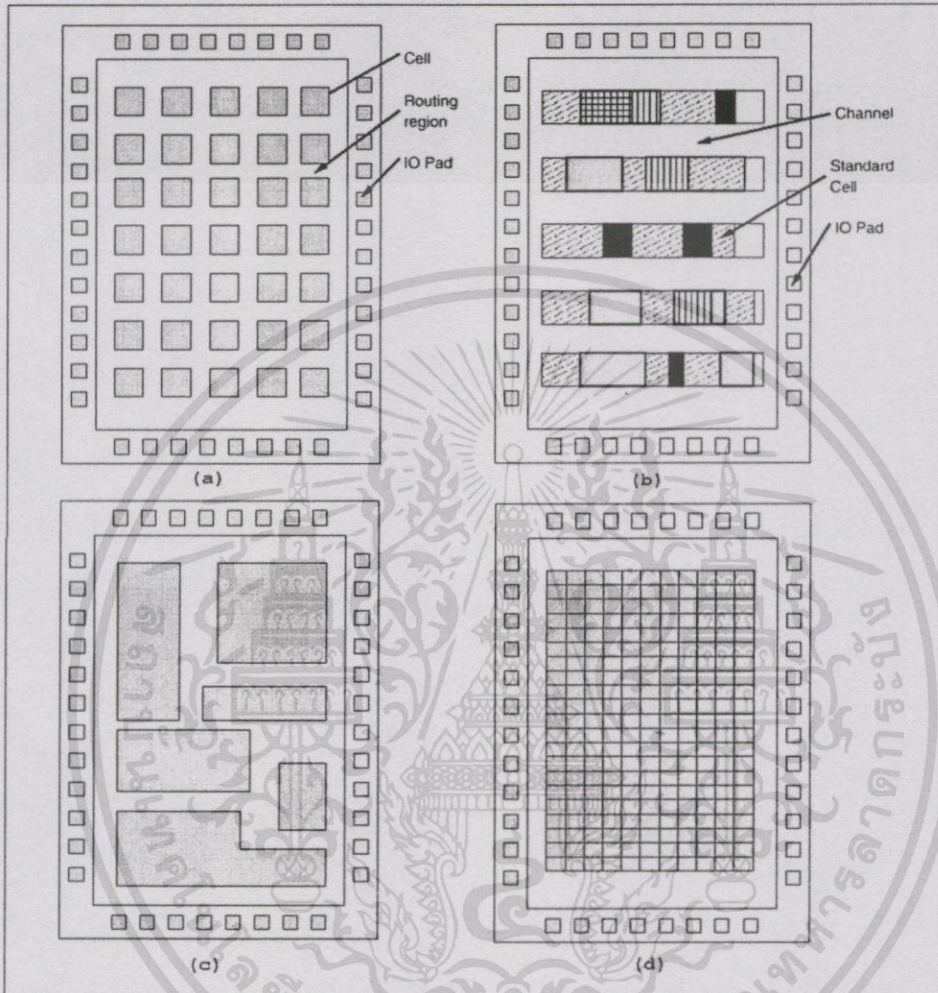
6.3 ปัญหาการจัดวางชิ้นส่วนของอุปกรณ์อิเล็กทรอนิกส์

ปัญหาการจัดวางชิ้นส่วนของอุปกรณ์อิเล็กทรอนิกส์เป็นปัญหา NP ดังนั้นวิธีการหาคำตอบของปัญหานี้ไม่สามารถคำนวณในเวลาที่เป็นโพลิโนเมียลของขนาดของปัญหาได้ [59] ในการหาคำตอบที่เหมาะสมที่สุดในการจัดวางชิ้นส่วน โดยการหาจากวิธีการที่เป็นไปได้ทั้งหมดจะพบว่าจะมีจำนวนวิธีการวางเท่ากับการหาค่า factorial ของจำนวนของชิ้นส่วนของอุปกรณ์ ซึ่งในทางปฏิบัติไม่สามารถคำนวณได้ในเวลาที่เหมาะสม การหาวิธีการวางโดยวิธีการลองสุ่มจึงเป็นทางเลือกหนึ่งสำหรับการหาคำตอบ คุณภาพของคำตอบที่ได้จากวิธีการวางโดยการลองสุ่มจึงขึ้นกับวิธีการที่ทำมาใช้ อย่างแรกที่ต้องการคือหาวิธีการจัดวางเพื่อให้ใช้ปริมาณการเชื่อมต่อที่น้อยที่สุด ถึงแม้ว่าวิธีการแต่ละวิธีที่นำมาใช้ไม่ได้รับรองว่าคำตอบหรือผลลัพธ์ที่ได้จะเป็นค่าที่เหมาะสมที่สุดก็ตาม

ในการจัดวางชิ้นส่วนของอุปกรณ์อิเล็กทรอนิกส์บนแผงวงจรไฟฟ้า ชิ้นส่วนต่าง ๆ ของอุปกรณ์อิเล็กทรอนิกส์จะมีการเชื่อมต่อกันเป็นวงจร การจัดวางชิ้นส่วนของอุปกรณ์ลงบนแผงวงจรสามารถพิจารณาได้เป็นปัญหาการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน โดยจุดประสงค์ในการหาค่าที่เหมาะสมของปัญหานี้คือการจัดเรียงชิ้นส่วนของอุปกรณ์ต่าง ๆ โดยใช้พื้นที่ให้น้อยที่สุดและการเชื่อมต่อกันระหว่างอุปกรณ์ใช้ระยะทางสั้นที่สุด ปัญหานี้คล้ายกับปัญหาการเดินทางของนกขมิ้น ต่างกันตรงที่ชิ้นส่วนของอุปกรณ์อิเล็กทรอนิกส์แต่ละชิ้นจะมีการเชื่อมโยงกับอุปกรณ์อื่นมากกว่าหนึ่งจุด ปัญหาการจัดวางชิ้นส่วนนี้มีการคำนวณหาคำตอบของวิธีการจัดวางอยู่หลายวิธี เช่น การคำนวณโดยวิธีการตัดแบ่งส่วนที่น้อย (Min-Cut Placement) การคำนวณโดยการแบ่งส่วน (Partitioning Placement) การคำนวณโดยวิธีการพันธุศาสตร์ (Placement by Genetic Algorithm) [59] แต่วิธีการส่วนใหญ่ที่นำไปใช้ในการจัดวางชิ้นส่วนและการหาเส้นทางเชื่อมโยงระหว่างอุปกรณ์จะนิยมใช้วิธีการจำลองการแอนนิลลิ่ง เนื่องจากวิธีการนี้เป็นวิธีการที่เหมาะสมกับปัญหาที่มีจำนวนรายการที่ต้องพิจารณาจำนวนมากและเป็นปัญหาของการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน

6.3.1 วิธีการที่ใช้ในการจัดวางชิ้นส่วนของอุปกรณ์ลงบนแผงวงจรรวม

วิธีการจัดวางอุปกรณ์ลงบนแผงวงจรรวมจัดเป็นปัญหาที่ยากต่อการหาคำตอบหรือ NP Complete ปัญหาหนึ่ง วิธีการหรือเทคนิคที่นำมาใช้ในการจัดวางอุปกรณ์ได้แก่ Simulated Annealing, force-directed placement, min-cut placement, placement by numerical optimization และวิธี evolution-based placement [59] แต่ละวิธีจะให้ผลลัพธ์ที่แตกต่างกันเนื่องจากวิธีการที่นำมาใช้ การจำลองการแอนนิลลิ่งจะให้ผลที่ไม่ซ้ำกันในแต่ละครั้งที่ทำการคำนวณ ในขณะที่วิธีการแบบ Force-directed จะจัดวางโดยพิจารณาว่าเซลล์จะติดกันในลักษณะสปริง



รูปที่ 6.17 แสดงประเภทของเซลล์ที่จัดวางอยู่ภายในแผงวงจรรวม โดยแยกตามขนาด ได้แก่

(a) Gate array (b) Standard Cell (c) Macro block (d) Sea-of-gate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.2 การใช้การจำลองการแอนนีลิ่งสำหรับแก้ปัญหาการจัดวางชิ้นส่วนอุปกรณ์อิเล็กทรอนิกส์

สำหรับปัญหาการจัดวางชิ้นส่วนอิเล็กทรอนิกส์ ได้กำหนดให้แต่ละเซลล์มีขนาดเท่ากันและมีการจัดวางในลักษณะ Gate Array ดังรูปที่ 6.17 (a) ค่าฟังก์ชันค่าใช้จ่ายที่ต้องพิจารณาถึงจะมีเพียงแต่ ค่าใช้จ่ายเนื่องจากการเชื่อมโยงแต่ละจุดบนแผงวงจรรวม

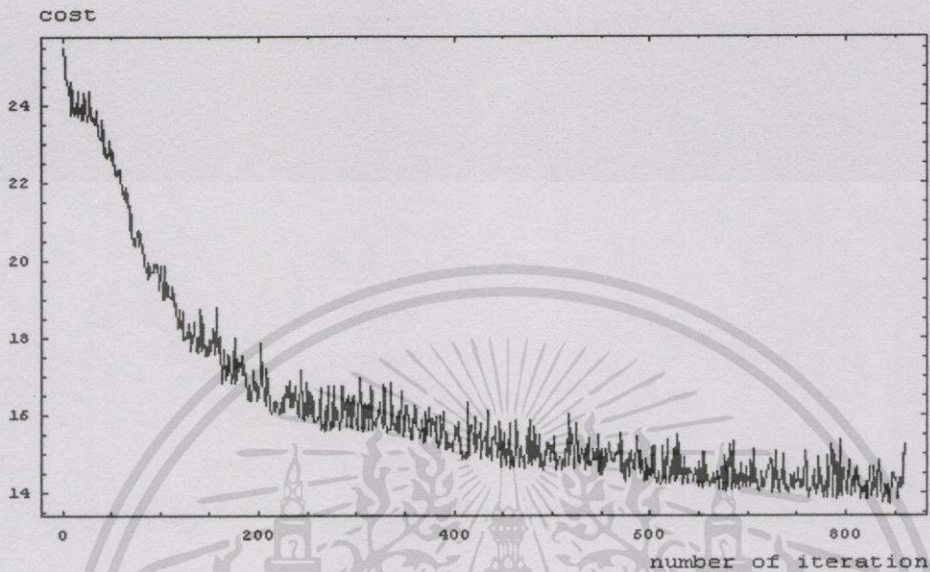
```

T = StartingTemperature()
Moves_per_iteration = MovesPerIter()
While StoppingCriterion(T) = False do
{
  Move_count = 0
  While (Moves_count < Moves_per_iter) do
  {
    Swap blocks
    Evaluate bounding box as cost function
    If (cost function decrease) then
      Accept swap
    else
      Probabilistic accept
    else
      Reject swap
    Move_count++
  } End while
  T = Adjust(T)
} End While

```

รูปที่ 6.18 แสดงขั้นตอนการคำนวณการจำลองการแอนนีลิ่งสำหรับปัญหาการจัดวาง

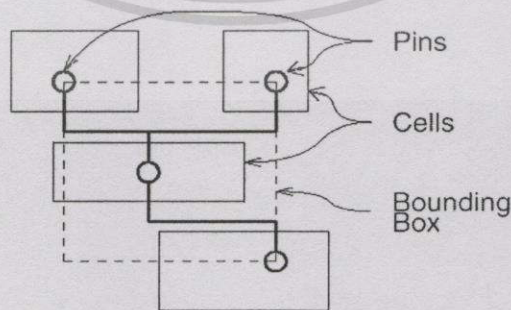
สำหรับการกำหนดวิธีการลดอุณหภูมิลง ใช้การลดอุณหภูมิแบบเรขาคณิต (Geometric cooling schedules) โดยใช้ค่า $\alpha = 0.97$ ในการคำนวณ จะได้ผลรวมของความยาวรวมดังแสดงในรูปที่ 6.19



รูปที่ 6.19 แสดงค่าความยาวรวมทั้งหมดของจุดที่เชื่อมต่อกันมีค่าลดลง สำหรับ Gate Array cell

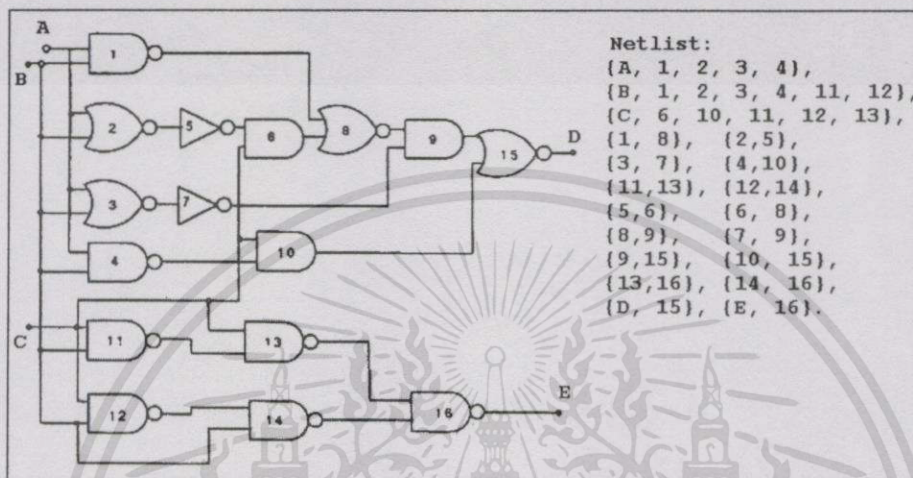
6.3.3 ฟังก์ชันค่าใช้จ่ายของปัญหาการจัดวาง

ในการคำนวณการจำลองการแอนนิลิ่งเพื่อแก้ปัญหาการจัดวาง ฟังก์ชันค่าใช้จ่ายจะเป็นผลรวมของระยะทางรวมที่ใช้ในการเชื่อมต่อระหว่างจุดต่าง ๆ ทั้งหมด ซึ่งประมาณค่าได้จากบริเวณที่ล้อมรอบ (Bounding Box) ดังรูปที่ 6.20 โดยในการคำนวณจะเริ่มจากการอ่านตำแหน่งของเซลล์ที่จัดอยู่ในรูปของ Netlist ซึ่งเป็นส่วนที่กำหนดว่าจะมีส่วนใดบ้างเชื่อมต่อกันจากนั้นก็ใช้วิธีการคำนวณตามรูปที่ 6.21 ซึ่งจะได้ผลลัพธ์ดังรูปที่ 6.23

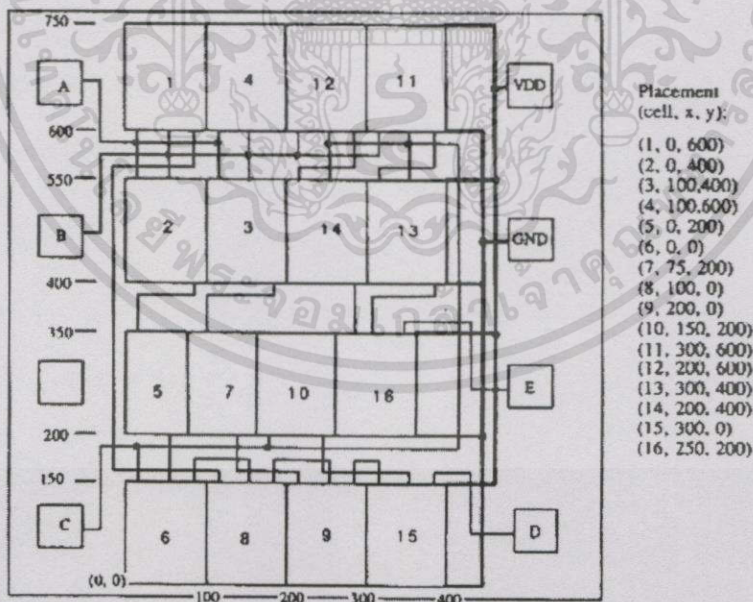


รูปที่ 6.20 แสดงวิธีการพิจารณาหา Bounding Box ระหว่างจุดที่เชื่อมต่อกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

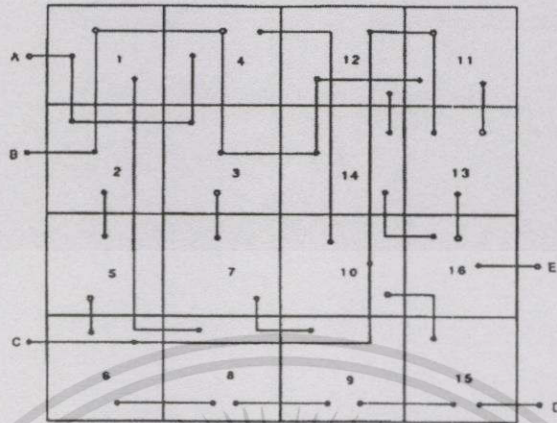


รูปที่ 6.21 แสดง Input Netlist ของวงจรรหัส



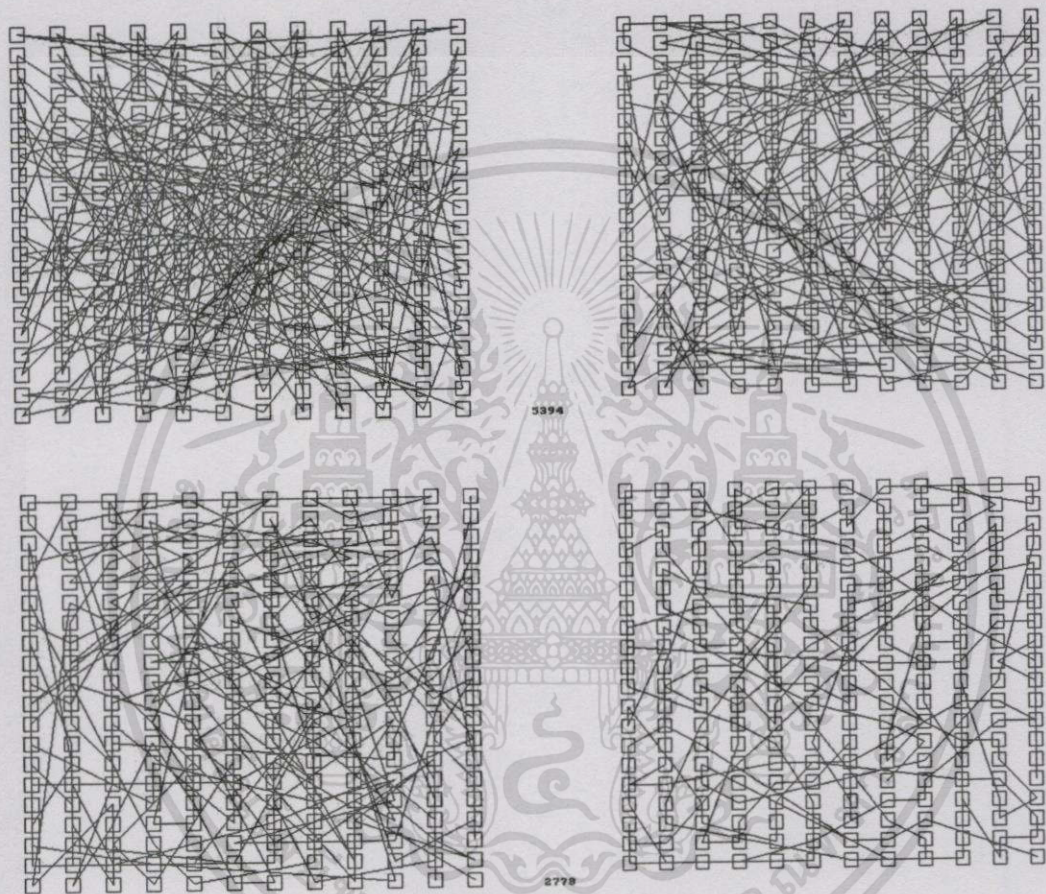
รูปที่ 6.22 แสดง Output ตำแหน่งของแต่ละเซลล์ สำหรับวิธีการจัดวางในแบบ Gate Array

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



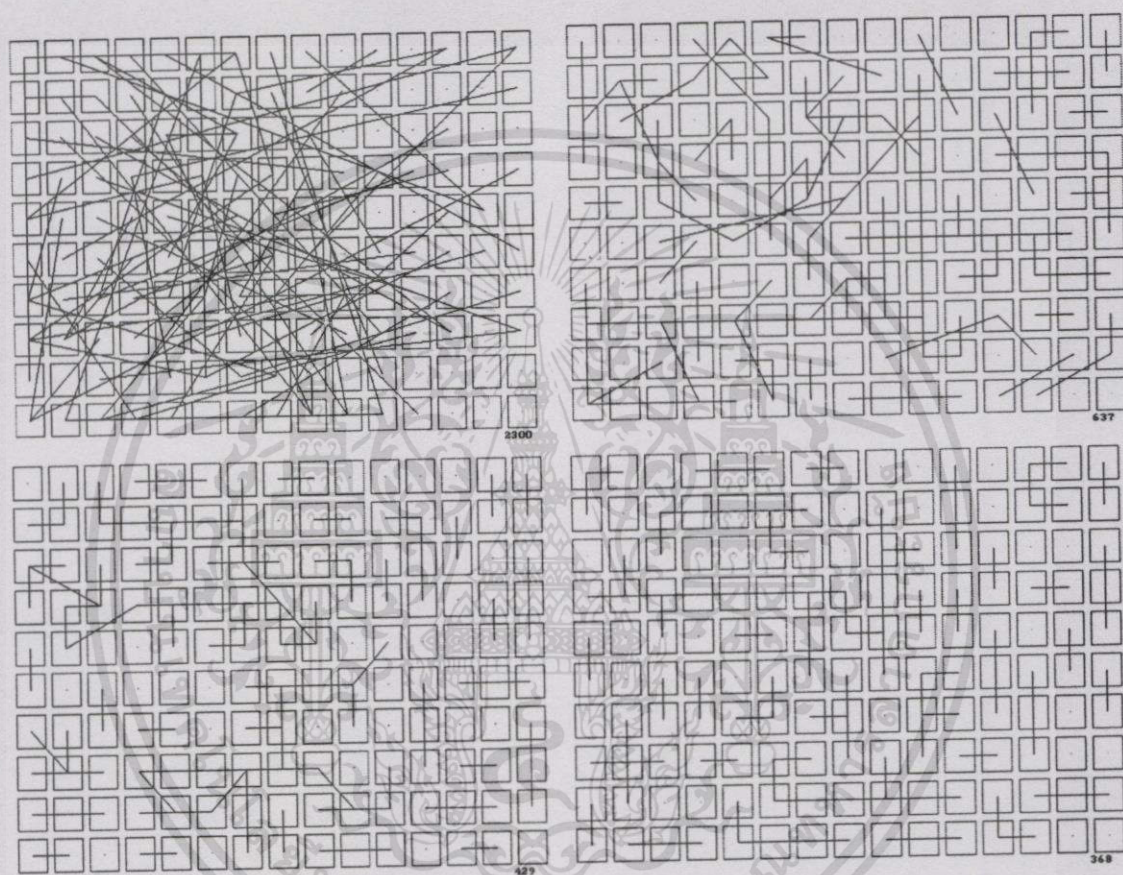
รูปที่ 6.23 แสดงลักษณะการเชื่อมต่อที่ได้จาก Netlist

จากรูปที่ 6.21-6.23 จะเป็นส่วนที่แสดงให้เห็นว่าในการหาค่าที่เหมาะสมของปัญหาการจัดวางชิ้นส่วนอุปกรณ์ คือการหาค่าตำแหน่งของอุปกรณ์โดยมีข้อกำหนดหรือ Constraint เป็น Netlist ซึ่งเป็นส่วนที่ใช้ในการคำนวณฟังก์ชันค่าใช้จ่าย สำหรับฟังก์ชันค่าใช้จ่ายที่พิจารณาจะมี 2 ลักษณะคือการคำนวณผลรวมของส่วนที่เชื่อมต่อทั้งหมดและการประมาณค่า ในการคำนวณผลรวมทั้งหมดค่าที่ได้จะเป็นค่าที่มีความผิดพลาดเนื่องจากตำแหน่งที่จัดวางในแต่ละครั้งจะมีการเปลี่ยนแปลงผลลัพธ์ที่ได้ ดังแสดงในรูปที่ 6.24 สำหรับการประมาณค่าโดยใช้ส่วนที่ล้อมรอบระหว่างจุดที่เชื่อมต่อกัน (Bounding Box) จะให้ผลลัพธ์ที่ดีกว่าการคำนวณผลรวม ดังในรูปที่ 6.25 นอกจากนี้ฟังก์ชันค่าใช้จ่ายสำหรับปัญหาการจัดวางอาจพิจารณาได้จาก Rectilinear Steiner Tree [36]



รูปที่ 6.24 แสดงผลลัพธ์ของการคำนวณการแอนนิลลิ่งสำหรับปัญหาการจัดวางชิ้นส่วน โดยฟังก์ชันค่าใช้
 จ่ายพิจารณาจากผลรวมของระยะทางที่เชื่อมต่อทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.25 แสดงผลลัพธ์ของการคำนวณการแอนนิลลิ่งสำหรับปัญหาการจัดวางชิ้นส่วนโดยฟังก์ชันค่า
ใช้จ่ายพิจารณาจากผลรวมของส่วนที่ล้อมรอบ (Bounding Box)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

ผลการทดลอง

7.1 การเปรียบเทียบวิธีการที่ใช้เพิ่มความเร็วสำหรับแต่ละปัญหา

ในการเปรียบเทียบระหว่างวิธีการคำนวณ 2 วิธี ถ้าวิธีการหนึ่งสามารถคำนวณได้เร็วกว่าอีกวิธีการหนึ่งแต่ให้ผลลัพธ์ที่ถูกต้องน้อยกว่าอีกวิธีการหนึ่งที่คำนวณได้ช้ากว่า วิธีการที่คำนวณได้เร็วจะไม่สามารถถือได้ว่ามีประสิทธิภาพสูงกว่า ดังนั้นในการเปรียบเทียบวิธีการคำนวณในเรื่องของเวลาจึงต้องใช้คุณภาพของคำตอบที่ได้เป็นตัววัดด้วยส่วนหนึ่ง วิธีการในการเปรียบเทียบสามารถทำได้โดยใช้ปัญหาที่ทราบคำตอบเป็นเกณฑ์ในการวัด แล้วใช้ค่าทางสถิติเป็นตัวเปรียบเทียบ

สำหรับวิธีการคำนวณการแอนนิลลิ่ง สำหรับปัญหาทั้ง 3 ที่เสนอขึ้นมา นั้น วิธีการที่สามารถนำมาใช้ในการลดเวลาในการคำนวณจะใช้วิธีการแบ่งการทำงานออกเป็น 2 ชั้น โดยใช้วิธีการอื่นในการคำนวณก่อน จากนั้นจึงหาอุณหภูมิเริ่มต้นสำหรับการใช้ในการคำนวณการจำลองแอนนิลลิ่งต่อ วิธีการที่นำมาใช้ในการคำนวณก่อนจะใช้การคำนวณที่มีลักษณะคล้ายกับวิธีการจำลองการแอนนิลลิ่ง คือวิธีการหาค่าเฉพาะที่ เพราะวิธีการนี้สามารถคำนวณคำตอบได้รวดเร็วแต่จะติดปัญหาที่ค่าเฉพาะที่ วิธีการคำนวณดังแสดงในรูปที่ 7.1

```
procedure ITERATIVE_IMPROVEMENT()  
{  
  minS = Sinf;  
  WHILE not (stopping_condition) DO {  
    S = random state;  
    WHILE not (local_minimum(S)) DO  
      {  
        S' = random state in neighbors (S);  
        If cost(S') < cost(S) then S = S'  
      }  
    If cost(S) < cost(minS) then minS = S;  
  }  
  Return(minS);  
}
```

รูปที่ 7.1 แสดงการคำนวณด้วยวิธีการหาค่าที่เหมาะสมเฉพาะที่ (local optimization)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับปัญหาการเดินทางของนักขายใช้วิธีการสุ่มตำแหน่งเมืองขึ้นมาและวัดระยะทางบน Euclidean plane และเปรียบเทียบผลลัพธ์ที่ได้กับคำตอบที่ทราบค่าจาก TSPLIB ดังตารางที่ 7.1 ในการเปรียบเทียบกับคำตอบโดยการเปลี่ยนค่า α จะพบว่าเปอร์เซ็นต์ของผลต่างจากค่าที่เหมาะสมจะน้อยลงเมื่อค่า α มีค่าเข้าใกล้ 0.99 ดังกราฟที่ 7.2 สำหรับปัญหาการวางแผนการขุดเจาะใช้วิธีการวัดเวลาที่ใช้ในการหาคำตอบที่จำนวนการชนกันทั้งหมดของหมากเป็นศูนย์ ดังรูปที่ 7.4 และสำหรับปัญหาการจัดวางจะกำหนดขนาดของเซลล์ให้มีขนาดเท่ากัน และแต่ละเซลล์จะมี Pin ที่เชื่อมต่อกับเซลล์อื่นอีก ก่อนทำการคำนวณการแอนนีลลิ่งสถานะเริ่มต้นของแต่ละปัญหาจะเป็นสถานะที่ถูกเลือกขึ้นมาอย่างสุ่มซึ่งมีค่าใช้จ่ายสูง จากนั้นทำการคำนวณหาคำตอบที่เป็นค่าที่เหมาะสมสำหรับใช้เป็นคำตอบตั้งต้นของการคำนวณการแอนนีลลิ่ง โดยวิธีการคำนวณนี้จะเรียกว่าการคำนวณแอนนีลลิ่งแบบสองขั้น (Two Stage Simulated Annealing)

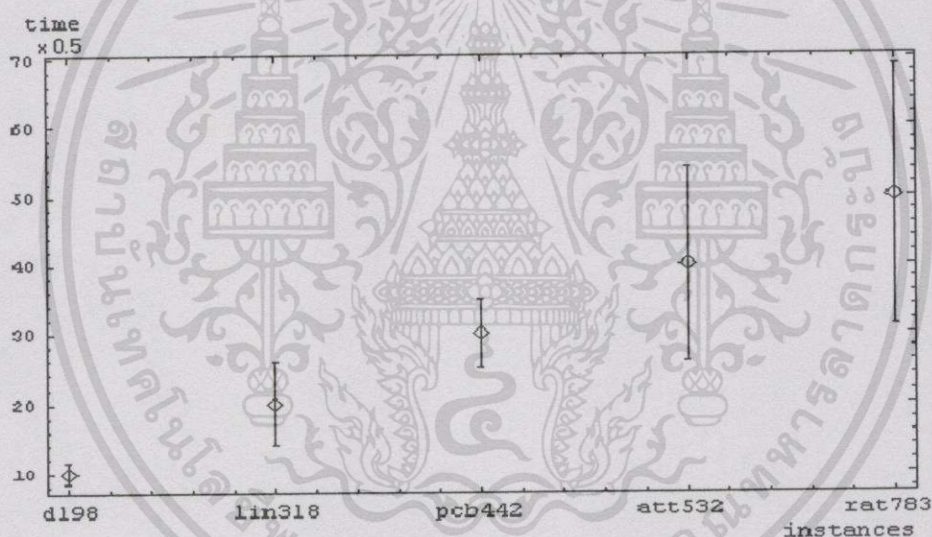


รูปที่ 7.2 แสดงเปอร์เซ็นต์ของผลต่างจากค่าที่ทราบค่าที่เหมาะสม(optimum) โดยการเปลี่ยนค่า α สำหรับปัญหาการเดินทางของนักขายโดยใช้ตัวอย่างปัญหา (Instances) จาก TSPLIB

ตารางที่ 7.1 แสดงเวลาที่ใช้ในการคำนวณปัญหาการเดินทางของนักขายโดยตัวอย่างปัญหา (instance)

จาก TSPLIB

ตัวอย่างปัญหา (Instances)	ค่าที่น้อย (Best)	ค่าเฉลี่ย (Average)	ค่าที่มาก (Worst)	เวลาโดยเฉลี่ย (Average time)	เวลามากที่สุด (max time)
d198	15780	15780.4	15784	1:01	2:50
lin318	42029	42029.0	42029	1:34	7:30
pcb442	50778	50911.2	51047	5:08	10:00
att532	27686	27707.9	27756	6:27	20:50
rat783	8806	8814.4	8837	16:05	35:00



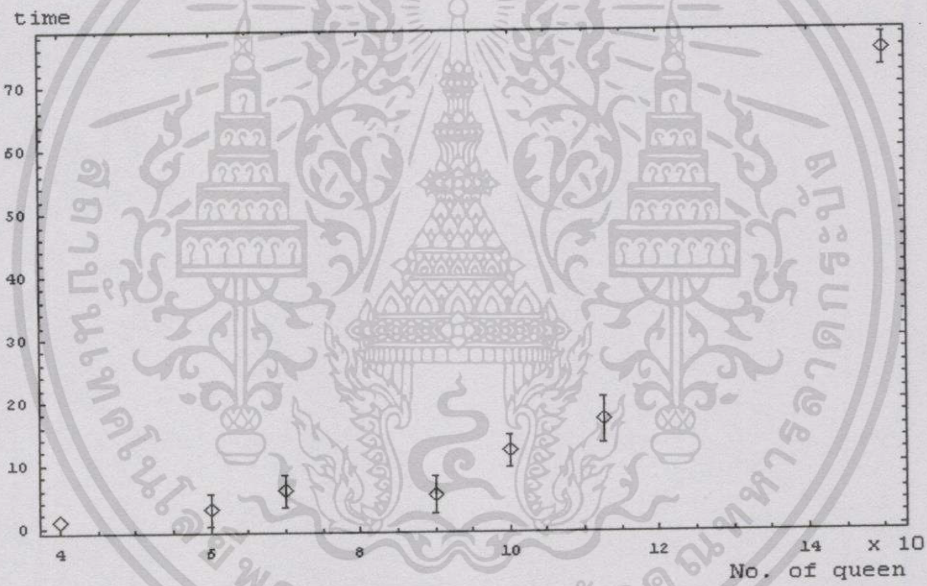
รูปที่ 7.3 แสดงกราฟเวลาที่ใช้ในการคำนวณปัญหาการเดินทางของนักขายโดยตัวอย่างปัญหา (instance)

จาก TSPLIB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 7.2 แสดงเวลาที่ใช้ในการคำนวณสำหรับปัญหาการวางหมากราชินีโดยวิธีการค้นหาเฉพาะที่

จำนวน N	จำนวนรอบ (No. of iteration)	เวลาโดยเฉลี่ย (Average time)	เวลามากที่สุด (max time)
40	2630	1.1	1
60	2901	3.1	3
70	4517	6.2	6
90	2222	5.7	5
100	6085	12.9	12
150	7950	77.5	77



รูปที่ 7.4 แสดงกราฟเวลาที่ใช้ในการคำนวณสำหรับปัญหาการวางหมากราชินีโดยวิธีการค้นหาเฉพาะที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2 การเปรียบเทียบเวลาที่ใช้ในการคำนวณ

ในการเปรียบเทียบเวลาที่ใช้ในการคำนวณระหว่างวิธีการใช้การเลือกคำตอบที่เป็นค่าที่เหมาะสมเป็นค่าเริ่มต้นสำหรับการคำนวณการแอนนิลลิงกับการคำนวณการแอนนิลลิงแบบโดยใช้สถานะเริ่มต้นเป็นค่าสุ่ม ความเร็วที่เพิ่มขึ้นสำหรับปัญหาการเดินทางของนักขายประมาณ 1.4 เท่าและ 1.1 เท่า สำหรับปัญหาการวางหมากราชินี

ตารางที่ 7.3 เปรียบเทียบเวลาที่ใช้ในการคำนวณปัญหาการเดินทางของนักขาย

จำนวนเมือง	SA	TSSA	จำนวนเท่า
42	0.9 วินาที	0.639 วินาที	1.41
50	1.23 วินาที	0.85 วินาที	1.45
57	1.92 วินาที	1.29 วินาที	1.49
123	6.71 วินาที	4.23 วินาที	1.59
318	125.98 วินาที	87.63 วินาที	1.44
532	651.70 วินาที	345.49 วินาที	1.89

ตารางที่ 7.4 เปรียบเทียบเวลาที่ใช้ในการคำนวณปัญหาการวางหมากราชินี

N	SA		TSSA		จำนวนเท่า
	เวลา	จำนวนรอบ	เวลา	จำนวนรอบ	
10	8 วินาที	900	7 วินาที	700	1.14
20	31 วินาที	17000	30 วินาที	20500	1.03
50	2 นาที	60000	2 นาที	58000	1.00
100	9 นาที	236000	8 นาที	252000	1.13
150	22 นาที	382000	20 นาที	289000	1.10
200	50 นาที	510000	47 นาที	322000	1.06

ตารางที่ 7.5 ตารางแสดงเวลาที่ใช้ในการคำนวณปัญหาการจัดวางชิ้นส่วนอุปกรณ์อิเล็กทรอนิกส์

จำนวน cell	SA	ความยาวรวม	TSSA	ความยาวรวม
140	45 นาที	179	40 นาที	155
240	230 นาที	1449	210 นาที	1436

การวิเคราะห์ที่ใช้การวิเคราะห์ความแปรปรวน (Analysis of Variance) หรือ ANOVA ในการทดสอบวิธีการจากตารางที่ 7.3 และ 7.4 สามารถคำนวณหาค่าสำหรับทดสอบได้ดังนี้

ในการทดสอบสมมติฐาน $H_0: \mu_1 = \mu_2$ จะได้ว่า $H_1: \mu_1 \neq \mu_2$ สำหรับปัญหาการเดินทางของนักขายตาราง ANOVA จะมีค่าดังนี้

ตารางที่ 7.6 ตาราง ANOVA ของปัญหาการเดินทางของนักขาย

ANOVA				
S.V.	df.	S.S.	M.S.	f
Treatment	1	10113.47	10113.47	$f_c=0.28$
Error	12	431794.21	35982.85	
Total	13			

$$n_1 = n_2 = 6, N = 12, t = 2$$

$$CF = \frac{G^2}{N} = \frac{(1228.57)^2}{12} = \frac{1509384.24}{12} = 125782.02$$

$$\sum_{i=1}^2 \sum_{j=1}^6 x_{ij}^2 = 567697.89$$

$$SST = \sum_{i=1}^2 \sum_{j=1}^6 x_{ij}^2 - CF = 567697.89 - 125790.21 = 441907.68$$

$$SS_{Str} = \frac{1}{6}(788.49^2 + 440.12^2) - CF$$

$$SS_{Str} = \frac{1}{6}(621716.48 + 193705.61) - 125790.21 = 10113.47$$

$$SSE = SST - SS_{Str} = 441907.68 - 10113.47 = 431794.21$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้ μ_i เป็นค่าเฉลี่ยที่เพิ่มขึ้นของวิธีการที่ $i, i=1,2$

$$H_0 : \mu_1 = \mu_2$$

$$H_1 : \mu_1 \neq \mu_2$$

$$\alpha = 0.05$$

$$f_c = 0.28$$

$$CR : F > f_{(1,12),0.05} = 4.75$$

เนื่องจาก $f_c < f_{(1,12),0.05}$ แสดงว่า f_c ตกอยู่นอก CR เราไม่ปฏิเสธ H_0 ที่ $\alpha = 0.05$ สรุปได้ว่าการคำนวณมีความเร็วพอ ๆ กัน

สำหรับปัญหาการวางหมากราชินีจากตารางที่ 7.4 นำมาสร้างเป็นตาราง ANOVA ได้ดังนี้

ตารางที่ 7.7 ตาราง ANOVA ของปัญหาการวางหมากราชินี

ANOVA				
S.V.	df.	S.S.	M.S.	f
Treatment	1	5.336	5.336	$f_c = 0.009$
Error	12	7167.334	597.27	
Total	13			

$$n_1 = n_2 = 6, N = 12, t = 2$$

$$\sum_{i=1}^2 \sum_{j=1}^6 x_{ij}^2 = 11814$$

$$CF = \frac{G^2}{N} = \frac{55696}{12} = 4641.33$$

$$SST = 11814 - 4641.33 = 7172.67$$

$$SSTr = \frac{1}{6}(122^2 + 114^2) - CF$$

$$SSTr = \frac{1}{6}(14884 + 12996) - 4641.33 = 5.336$$

$$SSE = SST - SSTr = 7172.67 - 5.336 = 7167.334$$

$$f_c = \frac{MStr}{MSE} = \frac{SSTr / 1}{SSE / 12} = \frac{5.336}{597.27} = 0.009$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้ μ_i เป็นค่าเฉลี่ยที่เพิ่มขึ้นของวิธีการที่ $i, i=1,2$

$$H_0 : \mu_1 = \mu_2$$

$$H_1 : \mu_1 \neq \mu_2$$

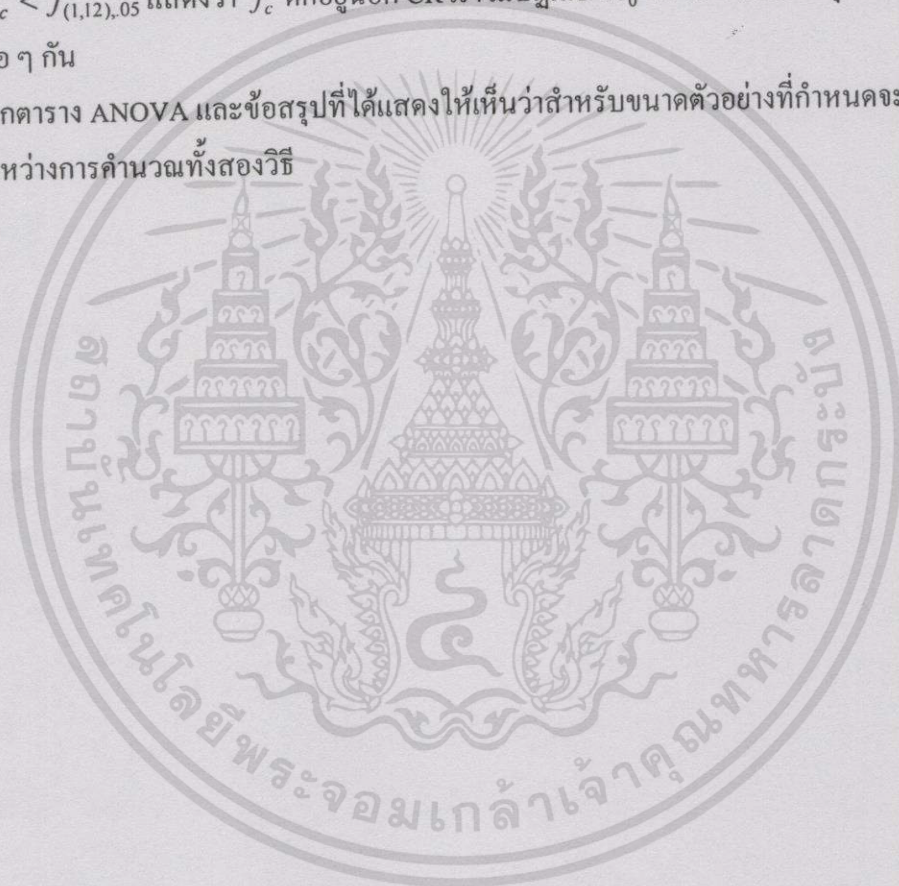
$$\alpha = 0.05$$

$$f_c = 0.009$$

$$CR : F > f_{(1,12),0.05} = 4.75$$

เนื่องจาก $f_c < f_{(1,12),0.05}$ แสดงว่า f_c ตกอยู่นอก CR เราไม่ปฏิเสธ H_0 ที่ $\alpha = 0.05$ สรุปได้ว่าการคำนวณมีความเร็วพอ ๆ กัน

จากตาราง ANOVA และข้อสรุปที่ได้แสดงให้เห็นว่าสำหรับขนาดตัวอย่างที่กำหนดจะไม่มี ความแตกต่างระหว่างการคำนวณทั้งสองวิธี



บทที่ 8

สรุป วิจารณ์และข้อเสนอแนะ

วิทยานิพนธ์ฉบับนี้ได้เสนอการศึกษาวิธีการคำนวณการหาค่าที่เหมาะสมของการเรียงสับเปลี่ยน โดยอาศัยการจำลองการแอนนิลลิง โดยศึกษาค่าพารามิเตอร์ที่มีผลต่อคำตอบของปัญหา เช่น ปัญหาการเดินทางของนกฮูก ปัญหาการวางหมากราชินีและปัญหาการจัดวางชิ้นส่วนอุปกรณ์อิเล็กทรอนิกส์บนแผงวงจร รวม ซึ่งทั้งหมดเป็นปัญหาที่จัดเป็นปัญหาที่ยากต่อการหาคำตอบ เพื่อหาวิธีการเพิ่มความเร็วในการคำนวณ เนื่องจากปัญหาในการคำนวณด้วยวิธีการนี้ใช้เวลาในการคำนวณมาก เมื่อเปรียบเทียบกับวิธีการอื่น แต่ผลลัพธ์ที่ได้จะเป็นคำตอบที่เป็นค่าที่เหมาะสมของทั้งหมดและสามารถนำไปใช้ได้กับปัญหาที่จัดเป็นปัญหาที่ยากต่อการคำนวณ

จากผลการทดลอง จะได้การกำหนดคชดุมหภูมิเริ่มต้นสำหรับการคำนวณปัญหาทั้งสามแบบ และเกณฑ์ในการกำหนดการเย็นตัว เพื่อลดเวลาที่ใช้ในการคำนวณ

การปรับปรุงวิธีการคำนวณการจำลองการแอนนิลลิงนั้นยังสามารถปรับเปลี่ยนพารามิเตอร์ได้อีกหลายวิธี ตัวอย่างเช่น ฟังก์ชันการกระจายที่นำมากล่าวถึงจะมีเพียง 3 แบบ เท่านั้น ในขณะที่การกระจายในรูปแบบอื่นยังมีอีกเป็นจำนวนมาก เช่น Laplace distribution, Levy distribution, Rayleigh distribution, Fermi Dirac distribution แต่การจะนำมาใช้งานนั้นจะต้องพิจารณาถึงส่วนที่เกี่ยวข้องอื่น ๆ ก่อน อาจเป็นชนิดของปัญหา ตัวแปรและพารามิเตอร์ในการคำนวณรวมทั้งฟังก์ชันค่าใช้จ่ายด้วย

การเลือกฟังก์ชันค่าใช้จ่ายสำหรับนำมาใช้ในการคำนวณก็มีผลต่อคำตอบที่ได้เช่นกัน จากในตัวอย่าง ปัญหาการจัดวางชิ้นส่วนอุปกรณ์อิเล็กทรอนิกส์ ถ้าชนิดของเซลล์เป็นแบบ Macro block layout ที่ขนาดแตกต่างกัน ฟังก์ชันค่าใช้จ่ายจะต้องพิจารณาถึงอีกส่วนหนึ่งคือ วิธีการวางในลักษณะที่จะไม่ให้ซ้อนทับกันและจัดให้อยู่ในบริเวณที่ใช้เนื้อที่น้อยที่สุด เวลาในการคำนวณก็จะแตกต่างกันไปตามลักษณะของปัญหา

ในงานวิจัยเรื่องปัญหาการจัดวางที่พิจารณาปัญหาให้อยู่ในรูปที่ง่ายขึ้น อาจกำหนดลักษณะบางอย่างขึ้นมาเป็นเงื่อนไขบังคับ เช่น การจัดเฉพาะแถว หรือคอลัมน์ หรือกำหนดให้มีขนาดเท่ากัน เช่น Chip FPGA (Field Programmable Gate Array) จากนั้นจึงใช้วิธีการคำนวณแบบแอนนิลลิง ตัวอย่างซอฟต์แวร์ที่ใช้วิธีการจำลองการแอนนิลลิงกับปัญหาการจัดวางในการคำนวณได้แก่ TimberWolf, PUPPY, KOAN และ Octools การคำนวณการแอนนิลลิงของแต่ละซอฟต์แวร์จะใช้ฟังก์ชันค่าใช้จ่ายและวิธีการลดคชดุมหภูมิที่ต่างกันออกไป โดยอาจพิจารณาเรื่องของเวลาเพิ่มเข้ามาเป็นเงื่อนไขบังคับสำหรับฟังก์ชันค่าใช้จ่ายด้วย ดังนั้นการจะใช้วิธีการคำนวณแบบใดเพื่อแก้ปัญหาอะไรจะต้องพิจารณารายละเอียดและส่วนต่าง ๆ ก่อน

บรรณานุกรม

- [1] E.H.L Aarts and P.J.M. van Laarhoven. 1985. *A New Polynomial-Time Cooling Schedule*, Proc.IEEE ICCAD-85, Santa Clara, CA : 206-208
- [2] E.H.L. Aarts, F.M.J. de Bont, E.H.A. Habers, P.J.M. van Laarhoven. 1986. *Parallel Implementations of the Statistical Cooling Algorithm*,” Integration,vol. 4: 209-238
- [3] E.H.L. Aarts, J.H.M. Korst,P.J.M. van Laarhoven. 1988. “*Solving Traveling Salesman Problems by Simulated Annealing*,” J. Stat. Phys., vol. 50, 187-206.
- [4] E. Aarts, J. Korst. 1990. *Simulated Annealing and Boltzmann Machines*. John Wiley & Sons, Inc
- [5] R. Azencott, Ed., 1992. *Simulated Annealing: Parallelization Techniques*, John Wiley and Sons, New York, NY
- [6] D. R. Greening. 1995. “*Simulated Annealing with Errors*,” Ph.D. dissertation, University of California at Los Angeles
- [7] K. Paithonwattanakit, Y. Suansook, “*Simulated Annealing for N-Queen Problem*” in Proceeding APSBC2000 , Dec 22, 2000
- [8] A.M. Law, W. D. Kelton. 1982. *Simulation Modeling and Analysis*. McGraw-Hill Book Company
- [9] K.D. Boese, A.B. Kahng. 1994. “*Best-So-far vs. Where-You-Are: Implications for Optimal Finite-Time Annealing*”, Sys. and Control Letters, vol. 22: 71-78.
- [10] T. Boseniuk, W. Ebeling, Boltzmann-, Darwin-, and Haeckel 1991. ”*Strategies in Optimization Problems*,” Lecture Notes in Comp. Sci.: Parallel Prob. Solving from Nature, vol. 496, 430-444
- [11] A. Casotto, F. Romeo,A. Sangiovanni-Vincentelli. 1987. ” *A Parallel Simulated Annealing Algorithm for the Placement of Macro-Cells*”, IEEE Trans. CADICS, vol. 6: 838-847
- [12] A. Casotto, A. Sangiovanni-Vincentelli . 1987. “*Placement of Standard Cells Using Simulated Annealing on the Connection Machine*”, Proc. IEEE ICCD: 350-353
- [13] V. Cerny. 1985. “*Thermodynamical Approach to the Traveling Salesman Problem: An Efficient Simulation Algorithm*”, J. Optimization Th. and Appl., vol. 45: 41-51
- [14] K. D. Crawford. 1992. “Solving the n-Queens Problems Using Genetic Algorithms”, ACM :1039-1047
- [15] T.H. Cormen, C.E. Leiserson, and R.L. Rivest. 1990, *Introduction to Algorithms*, MIT Press, Cambridge, MA
- [16] G.A. Croes. 1958. “*A Method for Solving Traveling-Salesman Problems*”, Operations Research, vol. 5: 791-812

- [17] G.B. Dantzig, D.R. Fulkerson, and S.M. Johnson. 1954. "Solution of a Large Scale Traveling-Salesman Problem", *Operations Research*, vol. 2 : 393-410
- [18] F. Darema, S. Kirkpatrick, and V.A. Norton. 1987. "Parallel Techniques for Chip Placement by Simulated Annealing", *Proc. IEEE ICCD* : 87-90
- [19] L. Davis, Ed. 1987, *Genetic Algorithms and Simulated Annealing*, Pitman Publishers, London, England
- [20] T.E. Davis , J.C. Principe. 1991. "A Simulated Annealing Like Convergence Theory for the Simple Genetic Algorithm", *Proc. Fourth Int. Conf. Genetic Algs. and Their Apps.*, San Diego, CA: 174-181
- [21] A.E. Dunlop ,B.W. Kernighan. 1985. "A Procedure for Placement of Standard-Cell VLSI Circuits", *IEEE Trans. CADICS*, vol. 4, 92-98.
- [22] D.S. Johnson, C. R. Argon, L.A. McGeoch, C. Schevon. 1986."Optimization by Simulated Annealing:An Experimental Evaluation; Part I, Graph Partitioning. *Operations Research*, 865-892,1986
- [23] M.R. Garey ,D.S. Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Co., San Francisco, CA.
- [24] S.B. Gelfand , S.K. Mitter. 1985. "Analysis of Simulated Annealing for Optimization", *Proc. 24th Conf. Dec. and Control*, Ft. Lauderdale, FL, 779-886
- [25] S. Geman ,D. Geman. 1984. "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images", *IEEE Trans. PAMI*, vol. 6: 721-741
- [26] B. Gidas. 1985. "Non-Stationary Markov Chains and Convergence of the Annealing Algorithm", *J. Stat. Phys.*, vol. 39: 73-131
- [27] D.E. Goldberg. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley, Reading, MA.
- [28] D.E Goldberg.1990. "A Note On Boltzmann Tournament Selection for Genetic Algorithms and Population-Oriented Simulated Annealing", *Complex Systems*, vol. 4: 445-460
- [29] J.W. Greene,K.J. Supowit. 1986. "Simulated Annealing Without Rejected Moves", *IEEE Trans. CADICS*, vol. 5: 221-228.
- [30] D. Abramson. 1992. "A Very High Speed Architecture for Simulated Annealing", *Computer*: 27-36
- [31] J.S. Rose, W. Klebsch, J. Wolf, "Temperature Measurement and Equilibrium Dynamics of Simulated Annealing Placements", *IEEE Trans. CADICS*, vol. 9, 253-259,1990

- [32] L.K. Grover. 1987. "Standard Cell Placement Using Simulated Sintering", Proc. 24th ACM/IEEE DAC, Miami Beach, FL, 56-59.
- [33] B. Hajek, "A Tutorial Survey of Theory and Applications of Simulated Annealing", Proc. 24th IEEE Conf. Decision and Control, Ft. Lauderdale, FL, 755-760, 1985.
- [34] B. Hajek, "Cooling Schedules for Optimal Annealing", Math. Op. Res., vol. 13, 311-329, 1988.
- [35] B. Hajek and G. Sasaki, "Simulated Annealing - To Cool or Not ", Sys. and Control Letters, vol. 12, 443-447, 1989.
- [36] M. Hanan. 1966. "On Steiner's Problem With Rectilinear Distance", SIAM J. Appl. Math., vol. 14, : 255-265
- [37] S. Geman, D. Geman. 1984. "Stochastic Relaxation, Gibbs Distribution and the Bayesian Restoration in Images". IEEE Transactions on Pattern Analysis and Machine Intelligence: 721-741
- [38] M.D. Huang, F. Romeo, and A. Sangiovanni-Vincentelli. 1986. "An Efficient General Cooling Schedule for Simulated Annealing", Proc. IEEE ICCAD-86, Santa Clara, CA, 381-384,.
- [39] F.K. Hwang, D.S. Richards, P. Winter. 1992. *The Steiner Tree Problem*, North-Holland, Amsterdam, Netherlands,
- [40] L. Ingber and B. Rosen. 1992. "Genetic Algorithms and Very Fast Reannealing: A Comparison", Math. Comput. Modelling, vol. 16: 87-100
- [41] R. Jayaraman and R. Rutenbar. 1987. "Floorplanning by Annealing on a Hypercube Multiprocessor", IEEE ICCAD-87, Santa Clara, CA : 346-349
- [42] R. Jayaraman and F. Damera. 1988. "Error Tolerance in Parallel Simulated Annealing Techniques", Proc. IEEE ICCD, Rye Brook, NY: 545-548
- [43] D.S. Johnson. 1990. "Local Optimization and the Traveling Salesman Problem", Proc. 17th ICALP, 446-461,
- [44] M. Jones and P. Banerjee. 1987. "Performance of a Parallel Algorithm for Standard Cell Placement on the Intel Hypercube", 24th IEEE/ACM DAC, Miami Beach, FL :807-813
- [45] A.B. Kahng and G. Robins. 1992., "A New Class of Iterative Steiner Tree Heuristics with Good Performance", IEEE Trans. CADICS, vol. 11: 893-902
- [46] R.L Karg and G.L. Thompson. 1964. "A Heuristic Approach to Solving Traveling-Salesman Problems", Management Science, vol. 10 :225-247
- [47] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi. 1983. "Optimization by Simulated Annealing", Science, vol. 220 :45-54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [48] S.A. Kravitz and R.A. Rutenbar. 1987. "Placement by Simulated Annealing on a Multiprocessor", IEEE Trans. CADICS, vol. 6: 534-549
- [49] J.B. Kruskal 1956. "On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem", Proc. American Math. Soc., vol. 7: 48-50
- [50] P.J.M. van Laarhoven. 1988. "Theoretical and Computational Aspects of Simulated Annealing", Center for Mathematics and Computer Science, Amsterdam, Netherlands
- [51] P.J.M. van Laarhoven , E.H.L. Aarts. 1987. "Simulated Annealing: Theory and Applications", Reidel Publishing, Dordrecht, Netherlands
- [52] J. Lam and J.-M. Delosme. 1988. "Performance of a New Annealing Schedule", Proc. 25th ACM/IEEE DAC, Anaheim, CA: 306-311
- [53] J. Lam and J.-M. Delosme. 1988. "Simulated Annealing: A Fast Heuristic for Some Generic Layout Problems", Proc. IEEE ICCAD-88, Santa Clara, CA : 510-513.
- [54] E.L Lawler, J.K. Lenstra, A.H.G Rinnooy Kan, D.B. Shmoys, Ed. 1985. *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, John Wiley and Sons, New York, NY
- [55] F.-T. Lin, C.-Y. Kao, C.-C. Hsu. 1991. "Incorporating Genetic Algorithms into Simulated Annealing", Proc. Fourth Int. Symp. on AI : 290-297
- [56] S. Lin. 1965. "Computer Solutions of the Traveling Salesman Problem", Bell System Tech. J., vol. 44: 2245-2269
- [57] S. Lin and B.W. Kernighan. 1973. "An Effective Heuristic Algorithm for the Traveling Salesman Problem", Operations Research, vol. 21 :498-516
- [58] M. Lundy ,A. Mees. 1986. "Convergence of an Annealing Algorithm", Math. Programming, vol. 34 :111-124
- [59] K. Shahookar, P. Mazumder. 1991. "VLSI Cell Placement Techniques", ACM Computing Surveys, Vol. 23, No. 2, :145-220
- [60] S.W. Mahfoud ,D.E. Goldberg. 1995. "Parallel Recombinative Simulated Annealing: A Genetic Algorithm", Parallel Computing, vol. 21 :1-28.
- [61] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. 1953. "Equation of State Calculations by Fast Computing Machines", J. Chem. Phys.,vol. 21 :1087-1092
- [62] D. Mitra, F. Romeo, and A. Sangiovanni-Vincentelli. 1986. "Convergence and Finite-Time Behavior of Simulated Annealing", Adv. Appl. Prob., vol. 18: 747-771
- [63] R.H.J.M Otten and L.P.P.P. van Ginneken. 1984. "Floorplan Design Using Simulated Annealing", Proc. IEEE ICCAD-84, Santa Clara, CA :96-98

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [64] R.H.J.M. Otten and L.P.P.P. van Ginneken. 1986. "Annealing Applied to Floorplan Design in a Layout Compiler", Proc. Automation 86, Houston, TX: 185-228
- [65] R.H.J.M. Otten, L.P.P.P. van Ginneken. 1989. "Stop Criteria in Simulated Annealing", Proc. IEEE ICCD, Rye Brook, NY : 549-552
- [66] R.H.J.M. Otten, L.P.P.P. Ginneken.. *The Annealing Algorithm*, Kluwer Academic Publishers, Boston, MA.
- [67] W. Padberg, G. Rinaldi. 1987. "Optimization of a 532-City Symmetric TSP", Op. Res. Lett., vol. 6 : 1-7.
- [68] C.H. Papadimitriou, K. Steiglitz. 1982. *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, New York, NY.
- [69] B. Preas. 1987. "Benchmarks for Cell-Based Layout Systems", Proc. 24th ACM/IEEE DAC, Miami Beach, FL : 319-320
- [70] M. Rabaudengo and M. Sonza Reorda. 1994. "Floorplan Area Optimization Using Genetic Algorithms", Proc. Fourth GLSVLSI, South Bend :22-25.
- [71] Reinelt, G. 1994. *The Traveling Salesman: Computational Solutions for TSP Applications*, Springer-Verlag, New York, NY
- [72] F. Romeo and A. Sangiovanni-Vincentelli. 1991. "A Theoretical Framework for Simulated Annealing", *Algorithmica*, vol. 6: 302-345
- [73] J.S. Rose, W.M. Snelgrove, Z.G. Vranesic. 1988. "Parallel Standard Cell Placement with Quality Equivalent to Simulated Annealing", IEEE Trans. CADICS, vol. 7: 387-396.
- [74] J.S. Rose, W. Klebsch, J. Wolf 1990. "Temperature Measurement and Equilibrium Dynamics of Simulated Annealing Placements", IEEE Trans. CADICS, vol. 9 : 253-259.
- [75] S.M. Ross. 1989. *Introduction to Probability Models*, Academic Press, Boston, MA.
- [76] P. Roussel-Ragot, G. Dreyfus. 1990. "A Problem Independent Parallel Implementation of Simulated Annealing: Models and Experiments", IEEE Trans. CADICS, vol.9 :827-835.
- [77] S. Sahni, A. Bhatt, and R. Raghavan. 1986. "The Complexity of Design Automation Problems", Proc. Automation '86, Houston, TX :82-98.
- [78] J.S. Sargent and P. Banerjee. 1989. "A Parallel Row-Based Algorithm for Standard Cell Placement With Integrated Error Control", Proc. 26th ACM/IEEE DAC : 590-593.
- [79] C. Sechen and A. Sangiovanni-Vincentelli. 1985. "The TimberWolf Placement and Routing Package", IEEE J. Solid-State Circuits, vol. 20 : 510-522.

- [80] P. Sibani, J.M. Pedersen, K.H. Hoffmann, P. Salamon. 1990. "Monte Carlo Dynamics of Optimization Problems: A Scaling Description", Phys. Rev. A, vol. 42 : 7080-7086.
- [81] P.N. Strenski, S. Kirkpatrick. 1991. "Analysis of Finite Length Annealing Schedules", Algorithmica, vol. 6 : 346-366
- [82] J.M. Varanelli, J.P. Cohoon. 1993, "Two-Stage Simulated Annealing", Proc. 4th ACM/SIGDA Phys. Des. Wksp., Lake Arrowhead, CA : 1-10.
- [83] W. P. Swartz. 1993. "Automatic Layout of Analog and Digital Mixed Macro/Standard Cell Integrated Circuits". Ph.D. dissertation, Yale University.
- [84] H. Szu, R. Hartley, "Fast Simulated Annealing". Physics Letters A, 157-162, 1987
- [85] J.M. Varanelli, J.P. Cohoon. 1995. "Population-Oriented Simulated Annealing: A Genetic/Thermodynamic Approach to Optimization", Proceedings ICGA '95, 174-181, Pittsburgh, PA, 1995.
- [86] J.M. Varanelli, J.P. Cohoon. 1995. "A Fast Method for Generalized Starting Temperature Determination in Monotonically Cooling Two-Stage Simulated Annealing Systems," Technical 133 Report CS-95-08, Department of Computer Science, University of Virginia
- [87] J.M. Varanelli, J.P. Cohoon. 1995. "A Two-Stage Simulated Annealing Methodology", Proc. 5th Great Lakes Symp. on VLSI, Buffalo, NY : 50-53
- [88] S.R. White. 1984. "Concepts of Scale in Simulated Annealing", Proc. IEEE ICCD, Port Chester, NY: 646-651
- [89] E.E. Witte, R.D. Chamberlain, M.A. Franklin. 1991. "Parallel Simulated Annealing Using Speculative Computation," IEEE Trans. Parallel and Dist. Sys., vol. 2: 483-494
- [90] D.F. Wong, H.W. Leong, C.L. Liu. 1988. *Simulated Annealing for VLSI Design*, Kluwer Academic Publishers, Boston, MA

ภาคผนวก ก. บทความและงานวิจัยที่ได้รับการตีพิมพ์

1. กิตติ ไพฑูรย์วัฒนกิจ, ยูธนา สวนสุข, "การจำลองการแอนนีลลิ่ง" วารสารวิศวกรรมสาร
ในพระบรมราชูปถัมภ์ ปีที่ 47, เล่มที่ 8 สิงหาคม 2537 หน้า 81-87
2. Kitti Paithonwattanakit, Yoothana Suansook, "Simulated Annealing for N-Queen Problem" in
Proceeding APSBC2000 , Dec 22, 2000



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข. การวิเคราะห์ข้อมูลสำหรับการทดลองแบบ RCB (Randomized complete block)

วิธีการทางสถิติที่ใช้สำหรับการวิเคราะห์ข้อมูลซึ่งจัดหรือลดความคลาดเคลื่อนจากการทดลองสามารถทำได้โดยวิธีการแยกกลุ่ม (blocking) ของหน่วยทดลอง ในการแยกกลุ่มเราแยกหน่วยทดลองออกเป็นกลุ่ม ๆ โดยที่หน่วยที่อยู่กลุ่มเดียวกันจะเหมือน ๆ กัน แต่หน่วยที่อยู่ต่างกลุ่มกันจะต่างกัน ตัวอย่างเช่น ในการทดสอบประสิทธิภาพของวิธีการคำนวณ 2 แบบต่อปัญหาที่กำหนดให้ ประสิทธิภาพจะวัดจากคุณภาพของคำตอบและเวลาที่ใช้ในการคำนวณ โดยวิธีการจะถูกแบ่งออกตามจำนวนกลุ่มที่ต้องการเปรียบเทียบ ดังตารางโครงสร้างข้อมูลสำหรับการทดลองแบบ RCB

ตารางโครงสร้างข้อมูลสำหรับการทดลองแบบ RCB เพื่อเปรียบเทียบ k วิธีการและมี b กลุ่ม

กลุ่ม	วิธีการ				Totals $x_{i.}$	Means $\bar{x}_{i.}$
	1	2	K		
1	x_{11}	x_{12}	x_{1k}	$x_{1.}$	$\bar{x}_{1.}$
2	x_{21}	x_{22}	x_{2k}	$x_{2.}$	$\bar{x}_{2.}$
...
...
.....
b	x_{b1}	x_{b2}	x_{bk}	$x_{b.}$	$\bar{x}_{b.}$
Totals: $x_{.j}$	$x_{.1}$	$x_{.2}$	$x_{.k}$	$x_{..}$
Means: $\bar{x}_{.j}$	$\bar{x}_{.1}$	$\bar{x}_{.2}$	$\bar{x}_{.k}$		$\bar{x}_{..}$

x_{ij} แทนข้อมูลในกลุ่มที่ i ที่ได้รับวิธีการที่ j (ข้อมูลใน cell (i, j)) $i = 1, \dots, b$ และ $j = 1, \dots, k$ หมายเหตุ ในที่นี้มีทั้งหมด bk Cells และจำนวนที่ทำซ้ำ (replications) ในแต่ละ cell เป็น 1 จำนวนที่ทำซ้ำในแต่ละ cell อาจมีมากกว่า 1 ได้ แต่วิธีการวิเคราะห์จะเปลี่ยนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ

$$x_{.i} = \sum_{j=1}^k x_{ij} \text{ ผลรวมของกลุ่มที่ } i$$

$$x_{.j} = \sum_{i=1}^b x_{ij} \text{ ผลรวมของวิธีการที่ } j$$

$$x_{..} = \sum_{i=1}^b \sum_{j=1}^k x_{ij} = \text{ผลรวมของทั้งหมด (Grand total)}$$

$$\bar{x}_{.i} = \frac{x_{.i}}{k} = \text{mean ของกลุ่มที่ } i$$

$$\bar{x}_{.j} = \frac{x_{.j}}{b} = \text{mean ของวิธีการที่ } j$$

$$\bar{x}_{..} = \frac{x_{..}}{bk} = \text{Grand mean}$$

ANOVA table สำหรับการทดลองแบบ RCB ดังนี้

ที่มา (S.V)	df.	SS	$MS = \frac{SS}{df}$	fc
วิธีการ (Treatment)	$k-1$	$SS_{Tr} = b \sum_{j=1}^k (\bar{x}_{.j} - \bar{x}_{..})^2$	MS_{blk}	$f_1 = \frac{MS_{Tr}}{MSE}$
กลุ่ม (Blocks)	$b-1$	$SS_{blk} = k \sum_{i=1}^b (\bar{x}_{.i} - \bar{x}_{..})^2$	MSE	$f_2 = \frac{MS_{blk}}{MSE}$
Error	$(k-1)(b-1)$	$SSE = \sum_{i=1}^b \sum_{j=1}^k (x_{ij} - \bar{x}_{.i} - \bar{x}_{.j} + \bar{x}_{..})^2$		
Total	$bk-1$	$SST = \sum_{i=1}^b \sum_{j=1}^k (x_{ij} - \bar{x}_{..})^2$		

การทดสอบสมมติฐาน

จุดประสงค์ของการทดสอบแบบ RCB คือการเปรียบเทียบวิธีการไม่ใช่เพื่อเปรียบเทียบกลุ่ม การแยกกลุ่มเพื่อแยกส่วนของความแปรปรวนที่เนื่องมาจากความแตกต่างของกลุ่ม เพื่อให้ความแปรปรวนที่เหลืออยู่เป็นส่วนที่เนื่องมาจากความแตกต่างของวิธีการ ขั้นตอนในการทดสอบสมมติฐานเพื่อเปรียบเทียบ k วิธีการ (treatment)

1. $H_0 : \mu_1 = \mu_2 = \dots = \mu_k$ หรือ $H_0 : \beta_1 = \beta_2 = \dots = \beta_k = 0$
2. H_1 : มี mean อย่างน้อย 1 คู่ที่ไม่เท่ากัน หรือ H_1 : มี β อย่างน้อย 1 ตัวไม่เท่ากับ 0
3. กำหนดค่าระดับนัยสำคัญ α

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ใช้การทดสอบ F โดยพิจารณาจาก CR (Critical Region) : $F > f_{((k-1),(b-1)(k-1)),\alpha}$
5. คำนวณค่า $f_c = f_1 = \frac{MStr}{MSE}$
6. สรุป

ในการทดสอบสมมติฐานเพื่อเปรียบเทียบ b กลุ่ม (block)

1. $H_0 : \mu_1 = \mu_2 = \dots = \mu_k$ หรือ $H_0 : \alpha_1 = \alpha_2 = \dots = \alpha_b = 0$
2. H_1 : มี mean อย่างน้อย 1 คู่ที่ไม่เท่ากัน หรือ H_1 : มี α อย่างน้อย 1 ตัวไม่เท่ากับ 0
3. กำหนดค่าระดับนัยสำคัญ α
4. ใช้การทดสอบ F โดยพิจารณาจาก CR (Critical Region) : $F > f_{((k-1),(b-1)(k-1)),\alpha}$
5. คำนวณค่า $f_c = f_2 = \frac{MStr}{MSE}$
6. สรุป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ภาคผนวก ก. ตำแหน่งของเมืองสำหรับปัญหาการเดินทางของ
นักขายขนาด 500 เมือง**

แสดงตำแหน่งของคำตอบของปัญหาการเดินทางของนักขายโดยผู้ลำดับแต่ละค่าจะแทนจุดใน
ระนาบในระบบพิกัดฉาก

(20,20)	(567,12)	(370,267)	(280,300)	(498,24)	(560,391)	(426,164)	(280,320)	(443,447)	(98,111)
(278,68)	(280,340)	(78,117)	(243,108)	(289,382)	(280,360)	(46,39)	(100,18)	(259,392)	(280,380)
(162,241)	(466,448)	(10,94)	(280,400)	(451,326)	(125,377)	(509,280)	(300,20)	(289,86)	(236,297)
(29,349)	(300,40)	(322,298)	(310,185)	(52,228)	(300,60)	(489,353)	(377,172)	(35,148)	(300,80)
(175,446)	(574,212)	(328,190)	(300,100)	(93,246)	(443,314)	(300,160)	(514,187)	(514,106)	(404,25)
(300,180)	(292,84)	(126,408)	(560,352)	(300,200)	(466,44)	(113,415)	(596,384)	(300,220)	(299,337)
(104,307)	(257,337)	(300,240)	(270,10)	(190,407)	(172,394)	(300,260)	(443,368)	(291,331)	(297,143)
(300,280)	(534,221)	(328,67)	(102,197)	(300,300)	(391,89)	(81,364)	(149,50)	(300,320)	(484,298)
(79,93)	(334,426)	(300,340)	(571,89)	(480,321)	(133,146)	(300,360)	(411,108)	(272,75)	(376,88)
(300,380)	(376,406)	(318,78)	(286,292)	(300,400)	(471,284)	(243,6)	(557,400)	(320,20)	(273,63)
(484,376)	(214,282)	(320,40)	(287,181)	(435,365)	(528,125)	(320,60)	(269,311)	(114,303)	(210,315)
(320,80)	(353,420)	(222,318)	(61,59)	(320,100)	(590,42)	(155,16)	(525,414)	(320,120)	(309,140)
(334,129)	(28,261)	(320,140)	(497,167)	(419,220)	(570,362)	(320,160)	(5,48)	(214,379)	(474,68)
(320,180)	(79,167)	(142,185)	(179,130)	(320,200)	(467,14)	(361,270)	(169,356)	(320,220)	(20,242)
(74,269)	(442,447)	(320,240)	(275,436)	(119,129)	(57,169)	(320,260)	(446,47)	(192,289)	(417,330)
(320,280)	(530,7)	(105,8)	(246,389)	(320,300)	(456,311)	(264,327)	(61,38)	(320,320)	(160,42)
(8,233)	(101,449)	(320,340)	(159,306)	(187,141)	(345,191)	(320,360)	(482,188)	(497,195)	(594,2)
(320,380)	(29,71)	(36,71)	(288,45)	(320,400)	(307,179)	(4,317)	(515,238)	(340,20)	(380,239)
(401,245)	(146,199)	(340,40)	(476,99)	(293,419)	(178,147)	(340,60)	(309,102)	(318,315)	(370,59)
(340,80)	(353,330)	(506,301)	(54,332)	(340,100)	(482,106)	(219,153)	(175,311)	(340,120)	(365,52)
(294,408)	(124,317)	(340,140)	(457,249)	(335,318)	(115,139)	(340,160)	(593,350)	(349,31)	(333,16)
(340,180)	(594,315)	(165,268)	(73,51)	(340,200)	(426,90)	(401,296)	(79,74)	(340,220)	(522,60)
(175,155)	(394,316)	(340,240)	(464,368)	(133,220)	(128,362)	(340,260)	(9,407)	(176,338)	(200,323)
(340,280)	(166,192)	(528,320)	(560,100)	(340,300)	(320,124)	(502,443)	(404,386)	(340,320)	(375,423)
(540,281)	(375,364)	(340,340)	(160,176)	(375,122)	(70,39)	(340,360)	(277,65)	(351,312)	(463,295)
(340,380)	(279,66)	(483,147)	(276,36)	(340,400)	(424,52)	(154,148)	(346,161)	(360,20)	(244,381)
(322,226)	(53,151)	(360,40)	(77,130)	(360,231)	(111,305)	(360,60)	(469,260)	(120,404)	(94,245)
(360,80)	(381,181)								

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ภาคผนวก ค. ตำแหน่งของเมืองสำหรับปัญหาการเดินทางของ
นักขายขนาด 500 เมือง (ต่อ)**

(593,199)	(546,89)	(360,100)	(167,72)	(143,410)	(100,170)	(360,120)	(272,304)	(320,74)	(402,295)
(360,140)	(494,205)	(75,87)	(515,22)	(360,160)	(553,315)	(233,223)	(303,272)	(360,180)	(525,424)
(123,182)	(358,243)	(360,200)	(236,306)	(344,168)	(259,380)	(360,220)	(536,86)	(294,340)	(424,26)
(360,240)	(13,151)	(430,338)	(439,384)	(360,260)	(96,20)	(477,64)	(299,298)	(360,280)	(386,161)
(36,116)	(546,138)	(360,300)	(263,383)	(377,396)	(111,95)	(360,320)	(412,293)	(133,378)	(214,228)
(360,340)	(396,380)	(364,277)	(365,374)	(360,360)	(442,270)	(599,402)	(19,302)	(360,380)	(442,133)
(43,398)	(148,340)	(360,400)	(496,401)	(395,122)	(8,445)	(380,20)	(116,394)	(540,418)	(467,349)
(380,40)	(406,132)	(397,83)	(111,241)	(380,60)	(339,24)	(52,140)	(65,68)	(380,80)	(211,330)
(478,83)	(585,85)	(380,100)	(300,263)	(413,78)	(378,363)	(380,120)	(43,421)	(32,1)	(2,27)
(380,140)	(19,300)	(295,96)	(578,285)	(380,160)	(318,284)	(529,117)	(565,103)	(380,180)	(165,157)
(444,82)	(183,350)	(380,200)	(107,84)	(116,68)	(339,299)	(380,220)	(108,305)	(72,208)	(490,255)
(380,240)	(389,47)	(308,130)	(208,124)	(380,260)	(540,117)	(96,96)	(182,388)	(380,280)	(101,27)
(179,206)	(79,48)	(380,300)	(382,247)	(282,359)	(277,172)	(380,320)	(240,216)	(218,98)	(168,186)
(380,340)	(330,320)	(454,78)	(551,286)	(380,360)	(395,16)	(338,343)	(94,413)	(380,380)	(347,134)
(315,433)	(483,396)	(380,400)	(492,164)	(522,122)	(358,384)	(400,20)	(310,250)	(205,61)	(275,271)
(400,40)	(475,202)	(553,251)	(309,150)	(400,60)	(138,322)	(206,24)	(76,447)	(400,80)	(425,355)
(543,258)	(481,85)	(400,100)	(534,450)	(299,228)	(35,9)	(400,120)	(431,329)	(199,169)	(331,52)
(400,140)	(448,56)	(224,268)	(310,298)	(400,160)	(498,29)	(495,92)	(114,238)	(400,180)	(450,354)
(118,374)	(562,81)	(400,200)	(349,203)	(148,164)	(268,56)	(400,220)	(442,235)	(572,366)	(520,76)
(400,240)	(559,15)	(247,131)	(521,316)	(400,260)	(505,292)	(388,404)	(86,35)	(400,280)	(159,380)
(77,63)	(344,370)	(400,300)	(524,38)	(208,249)	(347,101)	(400,320)	(70,348)	(246,81)	(503,219)
(400,340)	(290,109)	(108,334)	(114,443)	(400,360)	(412,34)	(331,98)	(380,305)	(400,380)	(141,273)
(437,175)	(20,178)	(400,400)	(590,348)	(166,4)	(33,177)	(33,196)	(20,20)		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง. คำตอบของปัญหาการวางหมากรากินีขนาด 200x200

แสดงตำแหน่งของคำตอบของปัญหาการวางหมากรากินีโดยคู่ลำดับแต่ละค่าจะแทนแถวและคอลัมน์ที่สามารถวางได้

(189,1),(192,2),(55,3),(162,4),(138,5),(134,6),(127,7),(64,8),(146,9),(126,10),(135,11),(69,12),(117,13),
 (73,14),(136,15),(66,16),(13,17),(120,18),(122,19),(133,20),(174,21),(147,22),(88,23),(1,24),(6,25),
 (23,26),(53,27),(33,28),(56,29),(41,30),(145,31),(54,32),(8,33),(101,34),(197,35),(191,36),(185,37),
 (150,38),(58,39),(79,40),(47,41),(87,42),(14,43),(77,44),(154,45),(74,46),(137,47),(16,48),(67,49),
 (132,50),(42,51),(24,52),(176,53),(142,54),(148,55),(76,56),(12,57),(43,58),(102,59),(52,60),(98,61),
 (15,62),(50,63),(93,64),(95,65),(39,66),(172,67),(81,68),(29,69),(63,70),(143,71),(30,72),(72,73),(17,74),
 (151,75),(171,76),(164,77),(140,78),(121,79),(10,80),(139,81),(49,82),(181,83),(128,84),(22,85),(36,86),
 (178,87),(187,88),(20,89),(196,90),(92,91),(78,92),(75,93),(9,94),(4,95),(91,96),(163,97),(158,98),
 (26,99),(89,100),(108,101),(40,102),(166,103),(198,104),(103,105),(38,106),(184,107),(193,108),
 (19,109),(71,110),(105,111),(35,112),(123,113),(194,114),(37,115),(199,116),(186,117),(165,118),
 (160,119),(110,120),(70,121),(84,122),(3,123),(173,124),(90,125),(109,126),(51,127),(149,128),
 (141,129),(32,130),(167,131),(18,132),(188,133),(104,134),(183,135),(115,136),(2,137),(34,138),
 (170,139),(156,140),(175,141),(21,142),(61,143),(27,144),(65,145),(100,146),(179,147),(83,148),
 (157,149),(57,150),(28,151),(190,152),(129,153),(44,154),(118,155),(180,156),(60,157),(114,158),
 (85,159),(169,160),(25,161),(106,162),(48,163),(168,164),(182,165),(11,166),(155,167),(80,168),
 (97,169),(144,170),(116,171),(195,172),(7,173),(125,174),(177,175),(112,176),(82,177),(111,178),
 (113,179),(94,180),(159,181),(130,182),(45,183),(153,184),(96,185),(152,186),(5,187),(107,188),
 (131,189),(31,190),(99,191),(62,192),(68,193),(119,194),(124,195),(59,196),(161,197),(86,198),(46,199),
 (200,200)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ. โปรแกรมต้นฉบับ

```

program N_QueenProblem;

uses
  Crt, Dos, Graph;

const N=1000;
      PI=3.142857;

type
  queen_position = record
    row,col :longint;
  end;

  queen_type = array[1..N] of queen_position;

var
  GraphDriver : integer; { The Graphics device driver }
  GraphMode   : integer; { The Graphics mode value }
  MaxX, MaxY  : word;    { The maximum resolution of the screen }
  ErrorCode   : integer; { Reports any graphics errors }
  MaxColor    : word;    { The maximum color value available }
  OldExitProc : Pointer; { Saves exit procedure address }

  count,num_queen :integer;
  test_queen,queen,r : queen_type;
  q,i             : longint;
  filevar        : text;
  qr             : real;
  u,v,w          :integer;
  no_of_swap     :integer;
  no_of_collision_before :integer;
  no_of_collision_after  :integer;
  row,col        :longint;
  num_row,num_col :integer;

procedure swap(var i,j :longint);
var temp :longint;
begin
  temp := j;
  j := i;
  i := temp;
end;

function rand(var i :longint):longint;
var rnd:longint;
begin
  randomize;
  rnd := random(N);
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
function tan(x:real):real;
begin
  tan := sin(x)/cos(x);
end;
```

```
function cauchy_rand(r:longint;p:real):real;
var u : real;
begin
  randomize;
  while (u = 0.5) do
  begin
    u := random(r)-0.5;
  end;
  cauchy_rand := p * tan(Pi*u);
end;
```

```
function expon(x:real):real;
begin
  if x < -40 then expon := 0 else
  if x > 40 then expon := exp(40) else expon := exp(x);
end;
```

```
procedure initialize(var q:queen_type);
var i :longint;
begin
  (* initial place queen on diagonal *)
  for i := 1 to num_queen do
  begin
    queen[i].row := i;
    queen[i].col := i;
  end;
end;
```

```
procedure show_queen(var q:queen_type);
var i :longint;
total :longint;
begin
  total := 0;
  (* Show position of each queen *)
  for i := 1 to num_queen do
  begin
    write('(', queen[i].row:1, ', ', queen[i].col:1, '), ');
  end;
  writeln;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(* Show diff of each row and each col  *)

for i := 1 to num_queen do
  begin
    total := total + (queen[i].row-queen[i].col);
    write(queen[i].row-queen[i].col:5);
  end;
  writeln;

(* Show sum of each row and each col  *)

for i := 1 to num_queen do
  begin
    write(queen[i].row+queen[i].col:5);
  end;
end;

function E(var q:queen_type):longint; (* number of collision *)
var
  i,j,k          :longint;
  collision       :longint; (* count number of collision *)
  number_of_collision :longint; (* count number of collision *)
begin
  collision := 0;
  number_of_collision := 0;
  for i := 1 to num_queen do
    begin
      for j := i+1 to num_queen do
        begin
          if ((q[i].row-q[i].col)=(q[j].row-q[j].col)) or
              ((q[i].row+q[i].col)=(q[j].row+q[j].col)) then
            begin
              collision := collision + 1;
            end;
          end;
        end;
      number_of_collision := collision;
      E := number_of_collision;
    end;
  end;

function attack(var q:queen_type;i,j :longint):boolean;
begin
  if ((q[i].row-q[i].col)=(q[j].row-q[j].col)) or
      ((q[i].row+q[i].col)=(q[j].row+q[j].col)) then
    attack := true else attack := false;
end;

```

```

procedure permutation(var q:queen_type);
var i,j :longint;
    seed :longint;

begin
    seed := num_queen;
    for i := 1 to num_queen do
        begin
            j := Random(seed-1)+1;
            swap(q[i].row,q[j].row);
        end;
    end;
end;

```

```

procedure swap_queen(var q:queen_type);
var i,j :longint;
    seed :longint;

```

```

begin

```

```

    seed := num_queen;
    for i := 1 to num_queen do
        begin
            i := Random(seed-1)+1;
            j := Random(seed-1)+1;
            swap(q[i].row,q[j].row);
        end;
    end;
end;

```

```

procedure simanneal(var q:queen_type);
var finish,conflict :boolean;
    i,j,k :longint;
    count :real; (* count number of iteration *)
    collision :real; (* count number of collision *)
    number_of_collision :real; (* count number of collision *)
    delta,t :real;
    E1,E2 :real;
    aa,bb,cc :real;

```

```

begin

```

```

    j := 0; aa := 0; bb := 0; cc := 0; count := 0;
    finish := false; t := 1E+30;
    E1 := E(q);
    randomize;
    While (E(q) <> 0) do
        begin
            E2 := E(q);
            count := count+1;

            k := random(num_queen-1)+1;
            delta := E2-E1;

            for i := 1 to num_queen do
                Begin
                    if (E2-E1 < 0) then
                        begin
                            swap(q[i].row,q[k].row);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        end else
        if (expon(-delta/t) > Random(10)) then
        begin
            swap(q[i].row,q[k].row);
        end;
    End; (* for *)

    t := 0.9*t;

End; (* while *)

end;

procedure draw_board(var q:queen_type);
var i,j,k :longint;

begin
    for i := 1 to num_queen do
    begin
        for j := 1 to num_queen do
        begin
            if (q[j].row=i) and (q[j].col=j) then
            begin
                write('|x')
            end else
            begin
                write('| ');
            end;
            end;
            writeln;
        end;
    end;

procedure check_conflict(q:queen_type);
var
    i,j,k          :longint;
    collision       :real;    (* count number of collision *)
    number_of_collision:real; (* count number of collision *)

begin
    collision := 0;
    number_of_collision := 0;
    for i := 1 to num_queen do
    begin
        for j := i+1 to num_queen do
        begin
            if ((q[i].row-q[i].col)=(q[j].row-q[j].col)) or
                ((q[i].row+q[i].col)=(q[j].row+q[j].col)) then
            begin
                write('(',queen[i].row:1,',',queen[i].col:1,')');
                writeln('(',queen[j].row:1,',',queen
[j].col:1,')');
                collision := collision + 1;
            end;
        end;
    end;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        end;
        number_of_collision := collision;
        writeln('number of collision : ', number_of_collision:5:0);
    end;

procedure swap_for_conflict(q:queen_type);
var
    i,j,k          :longint;
    u,v,w          :longint;
    collision       :real;      (* count number of collision *)
    no_of_collision_before :real;
    no_of_collision_after  :real;

begin
    collision := 0;
    no_of_collision_before := E(queen);
    for u := 1 to num_queen do
        begin
            for v := u+1 to num_queen do
                begin
                    if ((q[u].row-q[u].col)=(q[v].row-q[v].col)) or
                        ((q[u].row+q[u].col)=(q[v].row+q[v].col)) then
                        begin
                            swap(q[u].row,q[v].row);
                            no_of_collision_after := E(queen);
                            if no_of_collision_after > no_of_collision_before
                                then
                                    begin
                                        (* restore to before swap *)
                                        swap(q[v].row,q[u].row);
                                    end;
                                end;
                            end;
                        end;
                    end;
                end;
            end;
        end;
    end;

function Int2Str(L : LongInt) : string;
{ Converts an integer to a string for use with OutText, OutTextXY }
var
    S : string;
begin
    Str(L, S);
    Int2Str := S;
end; { Int2Str }

{$F+}
procedure MyExitProc;
begin
    ExitProc := OldExitProc; { Restore exit procedure address }
    CloseGraph;             { Shut down the graphics system }
end; { MyExitProc }
{$F-}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure Init;
{ Initialize graphics and report any errors that may occur }
var
  InGraphicsMode : boolean; { Flags initialization of graphics mode }
  PathToDriver   : string;  { Stores the DOS path to *.BGI & *.CHR }
begin
  { when using Crt and graphics, turn off Crt's memory-mapped writes }
  DirectVideo := False;
  OldExitProc := ExitProc;           { save previous exit proc }
  ExitProc := @MyExitProc;          { insert our exit proc in chain }
}
  PathToDriver := '';
  repeat

{$IFDEF Use8514}                               { check for Use8514 $DEFINE }
  GraphDriver := IBM8514;
  GraphMode := IBM8514Hi;
{$ELSE}
  GraphDriver := Detect;                     { use autodetection }
{$ENDIF}

  InitGraph(GraphDriver, GraphMode, PathToDriver);
  ErrorCode := GraphResult;                 { preserve error return }
  if ErrorCode <> grOK then                  { error? }
  begin
    Writeln('Graphics error: ', GraphErrorMsg(ErrorCode));
    if ErrorCode = grFileNotFound then { Can't find driver file }
    begin
      Writeln('Enter full path to BGI driver or type <Ctrl-Break> to
quit:');
      Readln(PathToDriver);
      Writeln;
    end
    else
      Halt(1);                               { Some other error: terminate }
    end;
  until ErrorCode = grOK;
  Randomize;                                 { init random number generator }
  MaxColor := GetMaxColor;                   { Get the maximum allowable drawing color }
  MaxX := GetMaxX;                           { Get screen resolution values }
  MaxY := GetMaxY;
end; { Initialize }

function RandColor : word;
{ Returns a Random non-zero color value that is within the legal
color range for the selected device driver and graphics mode.
MaxColor is set to GetMaxColor by Initialize }
begin
  RandColor := Random(MaxColor)+1;
end; { RandColor }

procedure DefaultColors;
{ Select the maximum color in the Palette for the drawing color }
begin
  SetColor(MaxColor);
end; { DefaultColors }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure DrawBorder;
{ Draw a border around the current view port }
var
  ViewPort : ViewPortType;
begin
  DefaultColors;
  SetLineStyle(SolidLn, 0, NormWidth);
  GetViewSettings(ViewPort);
  with ViewPort do
    Rectangle(0, 0, x2-x1, y2-y1);
end; { DrawBorder }

```

```

procedure FullPort;
{ Set the view port to the entire screen }
begin
  SetViewPort(0, 0, MaxX, MaxY, ClipOn);
end; { FullPort }

```

```

procedure box(x,y,w,h:integer);
begin
  moveto(x-w,y+h);
  lineto(x+w,y+h);

  moveto(x+w,y+h);
  lineto(x+w,y-h);

  moveto(x+w,y-h);
  lineto(x-w,y-h);

  moveto(x-w,y-h);
  lineto(x-w,y+h);
end;

```

```

procedure box_cross(x,y,w,h:integer);
begin
  moveto(x-w,y+h);
  lineto(x+w,y+h);

  moveto(x+w,y+h);
  lineto(x+w,y-h);

  moveto(x+w,y-h);
  lineto(x-w,y-h);

  moveto(x-w,y-h);
  lineto(x-w,y+h);

  (* cross *)

  moveto(x-w,y+h);
  lineto(x+w,y-h);

  moveto(x-w,y-h);
  lineto(x+w,y+h);
end;

```

end;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure draw_board_graphics(num_row,num_col:integer;var q:queen_type);
var row,col :longint;
    i,j,k :longint;
    width :integer;

begin
  init;
  setBkColor(15);
  if num_queen > 100 then
    width := 2
  else
    width := 4;

  row := 1;
  i := 1;
  while i <= num_row do
    begin
      row := row + 2*width;
      col := 1;
      j := 1;
      while j <= num_col do
        begin
          col := col + 2*width;
          if (q[j].row=i) and (q[j].col=j) then
            begin
              setcolor(9);
              (* circle(row,col,1); *)
              box_cross(row,col,width,width);
            end else
              begin
                setcolor(4);
                (* circle(row,col,1); *)
                box(row,col,width,width);
              end;
          j := j+1;
        end;
      i := i + 1;
    end;
  readln;

  closegraph;
end;

begin
  repeat
    Write('N :? ');
    readln(num_queen);
  until (num_queen >= 4) or (num_queen <=500);

  randomize;

  initialize(queen);

  (* permutation(queen); *)

  (* for i := 1 to 1 do *)
  count := 0;
  no_of_swap := 0;

  repeat
    begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

repeat
  u := random(num_queen);
  v := random(num_queen);
until (u <>0) and (v <> 0);

no_of_collision_before := E(queen);

swap(queen[u].row, queen[v].row);
(* writeln(' swap ',u,' ',v); *)

no_of_collision_after := E(queen);

if no_of_collision_after > no_of_collision_before then
begin
  (* swap back = no swap *)
  swap(queen[v].row, queen[u].row);

  no_of_swap := no_of_swap + 1;

end;
if no_of_collision_after < no_of_collision_before then
begin
  (* writeln(' accept swap '); *)
end;
end;

count := count + 1;

until (no_of_collision_before=0) or (count >50000);

simanneal(queen);

writeln(' number of conflict ',E(queen));
writeln(' number of swap ',no_of_swap);

(* show_queen(queen); *)

(* Show position of each queen *)

for i := 1 to num_queen do
begin
  write('(', queen[i].row:1, ', ', queen[i].col:1, ')', ', ');
end;
writeln;

(* draw_board(queen); *)
readln;
draw_board_graphics(num_queen, num_queen, queen);

end.

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อผู้เขียน	นายยุทธนา สนวนสุข
วัน เดือน ปีเกิด	วันที่ 18 สิงหาคม พ.ศ. 2509
วุฒิการศึกษาระดับปริญญาตรี	วิทยาศาสตรบัณฑิต สาขาฟิสิกส์
สถานที่สำเร็จการศึกษา	มหาวิทยาลัยมหิดล
ปีที่สำเร็จการศึกษา	2533
ประสบการณ์การทำงาน	นักเขียนโปรแกรม
อาชีพปัจจุบัน	นักเขียนโปรแกรม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้