ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ

เรื่อง การใช้ขั้นตอนวิธีเชิงพันธุกรรมในการสร้างลำดับการเทียบเรียงกลุ่มลำดับข้อมูลชีวภาพ

โดย นายกล้า วณิชชาโสภณ

ได้รับอนุมัติให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า

_____คณบดีบัณฑิตวิทยาลัย

(อาจารย์ ดร.มงคล หวังสถิตย์วงษ์)

21 พฤษภาคม 2550

คณะกรรมการสอบวิทยานิพนธ์

_____ประธานกรรมการ

(รองศาสตราจารย์ ดร.วรา วราวิทย์)

_____กรรมการ

(ดร.ศิษเฎศ ทองสิมา)

_____กรรมการ

(ดร.สิทธิรักษ์ รอยตระกูล)

_____กรรมการ

(รองศาสตราจารย์ ดร.ณชล ไชยรัตนะ)

การใช้ขั้นตอนวิธีเชิงพันธุกรรมในการสร้างลำดับการเทียบเรียงกลุ่มลำดับข้อมูลชีวภาพ

นายกล้า วณิชชาโสภณ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า
บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ
ปีการศึกษา 2549

ชื่อ : นายกล้า วณิชชาโสภณ
ชื่อวิทยานิพนธ์ : การใช้ขั้นตอนวิธีเชิงพันธุกรรมในการสร้างลำดับการเทียบเรียงกลุ่มลำดับข้อมูลชีวภาพ
สาขาวิชา : วิศวกรรมไฟฟ้า
สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ
ที่ปรึกษาวิทยานิพนธ์ : รองศาสตราจารย์ ดร.วรา วราวิทย์
ดร.ศิษเฎศ ทองสิมา
ปีการศึกษา : 2549

บทคัดย่อ

การเทียบเรียงกลุ่มลำดับข้อมูลชีวภาพ เป็นการเปรียบเทียบหาความคล้ายคลึงของสายลำดับข้อมูลตั้งแต่ 2 สายลำดับขึ้นไป เพื่อนำไปใช้ประโยชน์ทางด้านชีววิทยาโมเลกุลเช่น ทำนายโครงสร้างของโปรตีน การวิเคราะห์วิวัฒนาการของสิ่งมีชีวิตเป็นต้น การเทียบเรียงแบบ Progressive เป็นวิธิการแบบฮิวริสติกที่ใช้สำหรับการแก้ปัญหาการเทียบเรียงกลุ่มลำดับข้อมูล คุณภาพของการการเทียบเรียงแบบ Progressive จะขึ้นอยู่กับลำดับในการเทียบเรียง การหาลำดับในการเทียบเรียงจึงเป็นปัญหาสำคัญของการเทียบเรียงแบบ Progressive ในวิทยานิพนธ์นี้ศึกษาการปรับปรุงการเทียบเรียงแบบ Progressive โดยวิธี Progressive มีวิธีการหาลำดับการเทียบเรียงหลายวิธีเช่น ClustalW, Weiwei Zhong และ Mei-Jie Zhu เป็นต้น แต่ยังไม่มีวิธีที่ทำให้ ได้ลำดับการเทียบเรียงที่ดีที่สุด ในวิทยานิพนธ์นี้เสนอวิธีการหาลำดับการเทียบเรียงโดยใช้ขั้นตอนวิธีเชิงพันธุกรรมในการปรับปรุงประสิทธิภาพของลำดับการเทียบเรียง ที่สร้างขึ้นจากวิธีการต่างๆ ที่มีอยู่ในปัจจุบันให้มีประสิทธิภาพดีขึ้น โดยในงานนี้ได้ทดลองปรับปรุงลำดับการเทียบเรียงที่ได้จากโปรแกรม ClustalW และจากวิธี MSTโดยใช้ค่า Sum of Pairs (SP) เป็นตัววัดคุณภาพของการเทียบเรียง จากผลการทดลองเปรียบเทียบลำดับการเทียบเรียงที่ได้จากขั้นตอนวิธีเชิงพันธุกรรม กับลำดับการเทียบที่ได้จากโปรแกรม ClustalW และ ที่ได้จากวิธี MST ปรากฏว่าลำดับการเทียบเรียง ที่ได้จากขั้นตอนวิธีเชิงพันธุกรรมสามารถให้ค่า SP Score ได้ดีกว่า Guide Tree ของ ClustalW และ MST

(วิทยานิพนธ์มีจำนวนทั้งสิ้น 41 หน้า)

คำสำคัญ : การเทียบเรียงกลุ่มข้อมูลชีวภาพ, ลำดับการเทียบเรียง, ขั้นตอนวิธีเชิงพันธุกรรม

_____อาจารย์ที่ปรึกษาวิทยานิพนธ์

Name : Mr.Kla Wanichasopon

Thesis Title : Using Genetic Algorithm to Improve Guided Tree for Progressive Multiple Sequence Alignment

Major Field : Electrical Engineering

King Mongkut's Institute of Technology North Bangkok

Thesis Advisors : Associate Professor Dr.Vara Varavithya

Dr. Sissades Tongsima

Academic Year : 2006

Abstract

In molecular biology, Multiple Sequence Alignment (MSA) is one of the important tools in analyzing genome. There are several approach in multiple sequence alignment, for examples, structure prediction, Phylogenetic Analysis. MSA search for likelihood of comparing two or more sequences. We focus Progressive MSA (PMSA) which is heuristic algorithm. On of an important step in PMSA is the calculation of alignment order, guided tree, which has direct impact on the quality of alignment results. In this thesis, we propose techniques to improve the quality of guided tree. Presently, there are three efficient methods to determine guided tree which include CLUSTALW, Weiwei Zhong and Mei-Jie Zhu (minimum spanning tree). Further improvement is purposed in this thesis using genetic algorithm. The guided tree results generated from ClustalW and Minimum Spanning Tree methods are used in the initial population. The sum of pairs quality is used in evaluating quality of the alignment. From the simulation results, genetic algorithm with an enhancement of initial population from previously deterministic schemes can improve the quality of guided tree.

(Total 41 pages)

Keywords : Multiple Sequence Alignment, Guided Tree, Genetic Algorithm

_____Advisor

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สมบูรณ์ได้ด้วยความช่วยเหลืออย่างดียิ่งของ รองศาสตราจารย์ ดร.วราวราวิทย์ และ ดร.ศิษเฎศ ทองสิมา อาจารย์ที่ปรึกษาวิทยานิพนธ์ ดร.สิทธิรักษ์ รอยตระกูล และรองศาสตราจารย์ ดร.ณชล ไชยรัตนะ กรรมการสอบวิทยานิพนธ์ ที่ให้คำแนะนำ ข้อคิดเห็น และแนวทางแก้ไขต่างๆ อันเป็นประโยชน์อย่างยิ่งในการทำวิทยานิพนธ์นี้มาโดยตลอด และสุดท้ายขอขอบคุณพี่ๆ เพื่อนๆ ทุกคนที่ ห้องศูนย์วิจัยคอมพิวเตอร์ ที่ช่วยให้วิทยานิพนธ์นี้เสร็จสมบูรณ์ขึ้นมาได้

ท้ายนี้ผู้วิจัยใคร่ขอกราบขอบพระคุณบิดา มารดา และพี่ๆ น้องๆ ทุกคน ที่คอยช่วยเหลือสนับสนุนในด้านต่างๆ และให้กำลังเสมอมาจนสำเร็จการศึกษา

กล้า วณิชชาโสภณ

สารบัญ

สารบัญตาราง

สารบัญภาพ

สารบัญภาพ (ต่อ)

บทที่ 1
บทนำ

การเทียบเรียงกลุ่มลำดับข้อมูลทางชีววิทยาโมเลกุล (Multiple Sequence Alignment) เป็นหนึ่งในเครื่องมือที่มีความสำคัญในงานทางด้านชีววิทยาโมเลกุล การเทียบเรียงกลุ่มลำดับข้อมูลนำไปใช้ประโยชน์ได้หลายทางเช่น นำไปใช้ประโยชน์ในการทำนายโครงสร้างของโปรตีน (Structure Prediction) การวิเคราะห์วิวัฒนาการของสิ่งมีชีวิต (Phylogenetic Analysis) การออกแบบ Primer ในปฏิกิริยาพีซีอาร์ และในกรณีของโปรตีนใช้หาว่าช่วงใดมีความสำคัญควรนำไปศึกษาหรือควรวิเคราะห์ต่อไป เป็นต้น ในส่วนของปัญหาการเทียบเรียงข้องมูล (Alignment) คือการเทียบเรียงลำดับ 2 ลำดับ หรือมากกว่านั้น โดยหาลักษณะเฉพาะของลำดับที่มีลำดับเหมือนกัน เพื่อให้ได้การเทียบเรียงที่เหมาะสม ตัวอักขระที่ไม่เหมือนกัน และช่องว่างจะนำมาเปรียบเทียบโดยใช้ตารางค่าคะแนนในการเปรียบเทียบ การเทียบเรียงข้อมูลแบ่งตามการจัดได้ 2 แบบ คือ การเทียบเรียงแบบโกลบอล (Global Alignment) [1] และการเทียบเรียงแบบโลคอล (Local Alignment) [1] การเทียบเรียงแบบโกลบอล เป็นการเทียบเรียงที่ดีที่สุดของสายลำดับข้อมูลทั้งสาย และการเทียบเรียงแบบโลคอล เป็นการเทียบเรียงที่ดีที่สุดระหว่างส่วนย่อยของสายลำดับข้อมูล ถ้าแบ่งตามจำนวนการเทียบเรียง จะแบ่งออกเป็น การเทียบเรียงคู่ลำดับข้อมูล (Pairwise Alignment) [1] คือ การเทียบเรียงข้อมูลของข้อมูล 2 ชุดของสายอักขระ และการเทียบเรียงกลุ่มลำดับข้อมูล (Multiple Alignment) [1] คือการเทียบเรียงที่มีลักษณะเหมือนกับการเทียบเรียงคู่ลำดับข้อมูลเพียงแต่ จะเป็นการเปรียบเทียบลำดับข้อมูลตั้งแต่ 3 สายลำดับข้อมูลขึ้นไป

ไดนามิคโปรแกรมมิ่ง [2, 3] เป็นวิธีการตรวจสอบความคล้ายคลึงกันของลำดับข้อมูลวิธีหนึ่งที่รับประกันว่าจะได้ค่าที่ดีที่สุดในการเทียบเรียงลำดับข้อมูล โดยนำลำดับที่ต้องการเปรียบเทียบมาเรียงขนานคู่กัน แล้วหาลำดับที่คล้ายกันมากที่สุดเท่าที่จะเป็นไปได้ ในการทดสอบนี้บางครั้งอาจต้องยอมให้ลำดับเบสที่ไม่ตรงกันอยู่คู่กัน หรือยอมให้เกิดช่องว่างขึ้น เมื่อเปรียบเทียบลำดับเบส 2 สาย ในกรณีของเส้นดีเอ็นเอผลลัพธ์ที่เป็นไปได้ในแต่ละตำแหน่งคือ 1 Match คือมีเบสเหมือนกัน 2 Mismatch คือมีเบสต่างกัน 3 Gap คือเกิดช่องว่างในลำดับสายหนึ่ง ณ ตำแหน่งนั้น ส่วนในโปรตีนจะใช้ตารางในการเปรียบเทียบเช่น PAM, BLOSUM, GONNET เป็นต้น การเปรียบเทียบลำดับ 2 สายจะมี Alignment รูปแบบหลายแบบ ทำการเปรียบเทียบใน Alignment ทุกรูปแบบที่เป็นไปได้แล้วคัดเลือก Alignment รูปแบบที่ดีที่สุด โดยใช้ตารางคะแนน โดยวิธีการจะกล่าวในบทที่ 2

วิธีการแบบฮิวริสติค เป็นการประมาณค่าคะแนนความคล้ายกันที่เหมาะสม โดยโปรแกรมส่วนมากที่ใช้ในการเทียบเรียงกลุ่มลำดับข้อมูลจะเป็น (Progressive Alignment)

ปัญหาในการเทียบเรียงกลุ่มลำดับข้อมูลขนาดใหญ่ วิธีการไดนามิคโปรแกรมมิ่ง เมื่อใช้กับการเทียบเรียงกลุ่มลำดับข้อมูลจะมีค่าความซับซ้อนของเวลา คือ $O(n^k)$ เมื่อ n คือ ความยาวของแต่ละสายอักขระ และ k คือจำนวนสายอักขระ วิธีนี้จะใช้ได้กับการเทียบเรียงกลุ่มลำดับที่มีจำนวนน้อยและมีความยาวไม่มากนัก ถ้าต้องการเทียบเรียงข้อมูลที่มีจำนวนมากและขนาดยาว จึงต้องมีเทคนิคที่สามารถทำการเทียบเรียงกลุ่มลำดับที่มีความเร็วกว่าแบบไดนามิคโปรแกรมมิ่ง เทคนิคในการแก้ปัญหาในการเทียบเรียงกลุ่มลำดับข้อมูลแบ่งได้เป็น 2 แบบ คือ การเทียบเรียงแบบโปรเกรสซิฟ (Progressive Alignment) [1] กับ การเทียบเรียงแบบอิทเทอเรทีฟ (Iterative Alignment) [1]

การเทียบเรียงแบบ Progressive Alignment คือการเทียบเรียงสายลำดับแต่ละคู่ เพื่อเปรียบเทียบหาคู่ที่มีความคล้ายคลึงกันมากที่สุดมาทำการเทียบเรียงกันก่อน จากนั้นนำสายที่มีความคล้ายคลึงมากในลำดับถัดมา มาเทียบเรียงกันต่อไปตามลำดับ โดยวิธีการที่ใช้ในการเทียบเรียงที่ได้รับความนิยมมากที่สุดคือ Feng and Doolittle [4] หากลำดับการเทียบเรียงต่างกัน ผลที่ได้ออกมาก็จะมีค่าไม่เหมือนกัน ปัญหาหลักของการเทียบเรียงแบบโปรเกรสซีฟ คือการหาลำดับการเทียบเรียง วิธีการหาลำดับการเทียบเรียงมีหลายวิธีเช่น โปรแกรม ClustalW [5] ใช้วิธีเนเบอร์-จอยนิ่ง (Neighbor-Joining) ในการสร้างลำดับการเทียบเรียง โดยมี ClustalX [6] เป็นรุ่นที่แสดงผลเป็นแบบกราฟฟิค ClustalW-MPI [7] เป็นรุ่นที่ใช้วิธีการคำนวณแบบขนานของ Weiwei Zhong [1] ใช้หลักการ Traveling Salesman Problem (TSP) Algorithm มาใช้ในการหาลำดับการเทียบเรียง Mei-Jie Zhu [8] ใช้วิธีต้นไม้แบบแผ่ที่เล็กที่สุด (Minimum Spanning Tree) มาใช้ในการหาลำดับการเทียบเรียง

การเทียบเรียงแบบอิทเทอเรทีฟ คือการเทียบเรียงกลุ่มลำดับข้อมูล เริ่มจากการเทียบเรียงที่มีคุณภาพต่ำ จากนั้นทำการเทียบเรียงลำดับข้อมูลใหม่ ทำซ้ำไปซ้ำมาเป็นขั้นตอน จนได้การเทียบเรียงที่พอใจ ตัวอย่างโปรแกรมที่ใช้วิธีนี้ได้แก่ โปรแกรม SAGA [9], Anbarasu LA [10], Zhang and Andrew [11, 12] และ Li-Fang [13] ใช้ขั้นตอนวิธีเชิงพันธุกรรม มาใช้ในการเทียบเรียงกลุ่มลำดับข้อมูล Yixin Chen and Yi Pan [14] ใช้ขั้นตอนระบบอนานิคมมด (Ant Algorithm) มาใช้ในการเทียบเรียงกลุ่มลำดับข้อมูล

การเทียบเรียงแบบอิทเทอเรทีฟทำให้ได้ การเทียบเรียงกลุ่มลำดับข้อมูลที่ดีกว่า แต่จะใช้เวลาในการการคำนวณมากกว่าวิธีเทียบเรียงแบบโปรเกรสซีฟ ทำให้วิธีการเทียบเรียงแบบโปรเกรสซีฟ

เป็นที่นิยมมากกว่า โปรแกรมที่ได้รับนิยมมากที่สุดในการเทียบเรียงกลุ่มคือ โปรแกรม ClustalW โดยมี ClustalX เป็นรุ่นกราฟฟิคของ ClustalW

คุณภาพของการเทียบเรียงแบบ Progressive ขึ้นอยู่กับลำดับการเทียบเรียง เนื่องจากลำดับการเทียบเรียงที่ต่างกันจะให้ผลที่ไม่เหมือนกัน ปัญหาของการเทียบเรียงแบบ Progressive คือ การหาลำดับการเทียบเรียงที่ดีที่สุด การหาลำดับการเทียบเรียงในปัจจุบันมีอยู่หลายวิธี โดยส่วนใหญ่จะมีวิธีการทำงานเหมือนกันคือขั้นตอนที่ 1 หา Distance Matrix จากการทำ Pairwise Alignment ทุกคู่ลำดับจำนวน n(n-1)/2 คู่ แล้วเปลี่ยนค่าความคล้ายกัน เป็นDistance เก็บอยู่ในรูป Matrix ขนาด n×n โดย n คือจำนวน Sequence ขั้นตอนที่ 2 นำ Distance Matrix ที่ได้มาทำการหาลำดับการเทียบเรียงด้วยวิธีการต่างๆ ขั้นตอนที่ 3 ทำการเทียบเรียงตามลำดับการเทียบเรียงที่ได้ วิธีการส่วนใหญ่ของขั้นตอนที่ 1 และขั้นตอนที่ 3 จะเหมือนกัน จะต่างกันที่ขั้นตอนที่ 2 แต่เวลาที่ใช้ในการคำนวณส่วนมากจะอยู่ในขั้นตอนที่ 1 ส่วนขั้นตอนที่ 2 และ 3 จะใช้เวลาน้อยมากเมื่อเทียบกับขั้นตอนที่ 1 วิธีการหาลำดับการเทียบเรียงด้วยวิธีการต่างๆ ยังบอกไม่ได้ว่าวิธีไหนจะให้ Alignment ที่ดีกว่าวิธีไหนสำหรับการเทียบเรียงกลุ่มใดๆ

ในวิทยานิพนธ์นี้เสนอวิธีการหาลำดับการเทียบเรียง โดยใช้ขั้นตอนวิธีเชิงพันธุกรรมในการปรับปรุงประสิทธิภาพของลำดับการเทียบเรียง ที่สร้างขึ้นจากวิธีการต่างๆ ที่มีอยู่ในปัจจุบันให้มีประสิทธิภาพดีขึ้น โดยในงานนี้ได้ทดลองปรับปรุงลำดับการเทียบเรียงที่ได้จากโปรแกรม ClustalW และจากวิธี Minimum Spaning Tree โดยใช้ค่า Sum of Pairs (SP) เป็นตัววัดคุณภาพของการเทียบเรียงในงาน

วิธีการหาลำดับการเทียบเรียงด้วยวิธีที่มีอยู่ในปัจจุบันมีขั้นตอนทำงานเหมือนกัน จากการที่ยังบอกไม่ได้ว่าวิธีการไหนจะให้ผลลัพธ์ที่ดีที่สุด จากการที่เวลาในการคำนวณในส่วนการสร้างลำดับการเทียบเรียงและ การเทียบเรียงตามลำดับการเทียบเรียง ใช้เวลาในการคำนวณไม่มากเมื่อเทียบกับขั้นตอนการสร้าง Distance Matrix และจากการที่ขั้นตอนวิธีเชิงพันธุกรรมนั้นใช้หลักการเรื่องการอยู่รอดของผู้ที่แข็งแรงที่สุด (Survival of the Fittest) โดยในงานนี้ได้ใช้วิธีขั้นตอนเชิงพันธุกรรมจะทำให้ได้ ลำดับการเทียบเรียงที่ดีขึ้น โดยให้ลำดับการเทียบเรียง เป็นประชากรของขั้นตอนวิธีเชิงพันธุกรรม SP Score เป็น Fitness Function และกำหนดให้ประชากรเริ่มต้นเป็นลำดับการเทียบเรียงที่ได้จากวิธีการต่างๆ โดยในการทดลองนี้คาดหวังว่า การที่ประชากรรุ่นพ่อแม่ดีที่จะทำให้ประชากรรุ่นลูกที่ดี

เนื้อหาในวิทยานิพนธ์เล่มนี้แบ่งออกเป็น 4 ส่วนคือ ทฤษฎีที่เกี่ยวข้อง การประยุกต์ใช้งาน ผลการทดลอง และสรุปผล โดยบทที่ 2-3 เป็นทฤษฎีที่เกี่ยวข้องประกอบด้วย การเทียบเรียงและการเทียบเรียงกลุ่มลำดับข้อมูล และ ขั้นตอนวิธีเชิงพันธุกรรม บทที่ 4 ขั้นตอนและวิธีการทำงาน

ของโปรแกรม บทที่ 5 เป็นผลการทดลองและการอภิปรายผลการทดลอง และบทที่ 6 จะเป็นการสรุปวิทยานิพนธ์และแนวทางการพัฒนางานวิจัยต่อไป

บทที่ 2
การเทียบเรียงคู่และการเทียบเรียงกลุ่ม

การเทียบเรียงคู่ ในปัจจุบันมีหลายวิธี วิธีที่ได้รับความนิยมมากที่สุด คือวิธีไดนามิคโปรแกรมมิ่ง โดยในวิธีนี้จะรับประกันได้ว่าจะได้ Alignment ที่มีความคล้ายคลึงกันมากที่สุด ส่วนการเทียบเรียงกลุ่มนั้น วิธีการที่ได้รับความนิยมมากที่สุดคือวิธีการแบบ Progressive Alignment โดยโปรแกรมที่ได้รับความสนใจมาที่สุดได้แก่โปรแกรม ClustalW ซึ่งมี ClustalX เป็นรุ่นกราฟฟิคของโปรแกรม ClustalW โดยในงานวิจัยนี้เลือกใช้วิธีการเทียบเรียงแบบกลุ่มโดยใช้วิธีการแบบ Progressive Alignment เปรียบเทียบผลกับโปรแกรม ClustalW

2.1 ไดนามิคโปรแกรมมิ่ง (Dynamic Programming)

การเทียบเรียงแบบคู่ (Pairwise Alignment) โดยใช้วิธีไดนามิคโปรแกรมมิ่ง สามารถที่จะแก้ปัญหาได้ในเวลา $O(L^2)$ โดยที่ L คือความยาวของสายลำดับ (Sequence) โดยวิธีไดนามิคโปรแกรมมิ่งจะเป็นวิธีที่หาค่าที่รับประกันได้ว่าจะได้ค่าที่ดีที่สุดในการเทียบเรียงลำดับข้อมูล วิธีไดนามิคโปรแกรมมิ่งถูกคิดค้นโดย Needleman and Wunsch

ตัวอย่างวิธีการไดนามิคโปรแกรมมิ่ง 2 สายลำดับข้อมูล ได้แก่ AGGCCT (Sequence 1) และ AAGCT (Sequence 2) โดยให้ลำดับข้อมูลเหมือนกัน (Match Score) เท่ากับ 2 ต่างกัน (Miss Match Score) เท่ากับ -1 ช่องว่าง (Gap) เท่ากับ -2

2.1.1 การกำหนดค่าเริ่มต้น (Initialization)

ให้ m และ n คือความยาว Sequence1 และ Sequence2 ตามลำดับ ในตัวอย่างนี้ m=6 และ n=5 ในงานนี้จะสร้างตารางแถวในแนวตั้งขนาด m+1 และแถวในแนวนอนขนาด n+1 โดยเริ่มแรกจะใส่ค่าใน 0 ในแถวแรกของแนวตั้งและแนวนอนดังภาพที่ 2-1

2.1.2 การใส่ค่าในตาราง (Table fill)

ให้ $M_{i,j}$ คือค่าคะแนนในตารางคะแนนแถว i คอลัมน์ j โดย $M_{i,j}$ จะเลือกค่าที่ดีสุดจาก $M_{i-1\ j-1} + S(Ci, Cj)$ หรือ $M_{i\ j-1} + g$ หรือ $M_{i-1\ j} + g$ โดยที่ Ci คือลำดับข้อมูลตัวที่ i ของสายลำดับ 2 Cj คือลำดับข้อมูลตัวที่ j ของสายลำดับที่ 1 S(Ci, Cj) คือค่า Match Score ระหว่าง Ci กับ Cj g คือค่า Gap โดยในตัวอย่างนี้ S(Ci, Cj) มีค่าเท่ากับ 2 หรือ -1 ส่วน g มีค่าเท่ากับ -2

		A	G	G	C	C	T
	0	0	0	0	0	0	0
A	0						
A	0						
G	0						
C	0						
T	0						

ภาพที่ 2-1 ตัวอย่างการกำหนดค่าเริ่มต้นของตารางคะแนน

		A	G	G	C	C	T
	0	0	0	0	0	0	0
A	0	2	0	-1	-1	-1	-1
A	0	2	1	-1	-2	-2	-2
G	0	0	4	3	1	-1	-3
C	0	-1	2	3	5	3	1
T	0	-1	0	1	3	4	5

ภาพที่ 2-2 ตัวอย่างตารางคะแนนหลังจากใส่ค่า

2.1.3 การหาเส้นทางกลับ (Traceback)

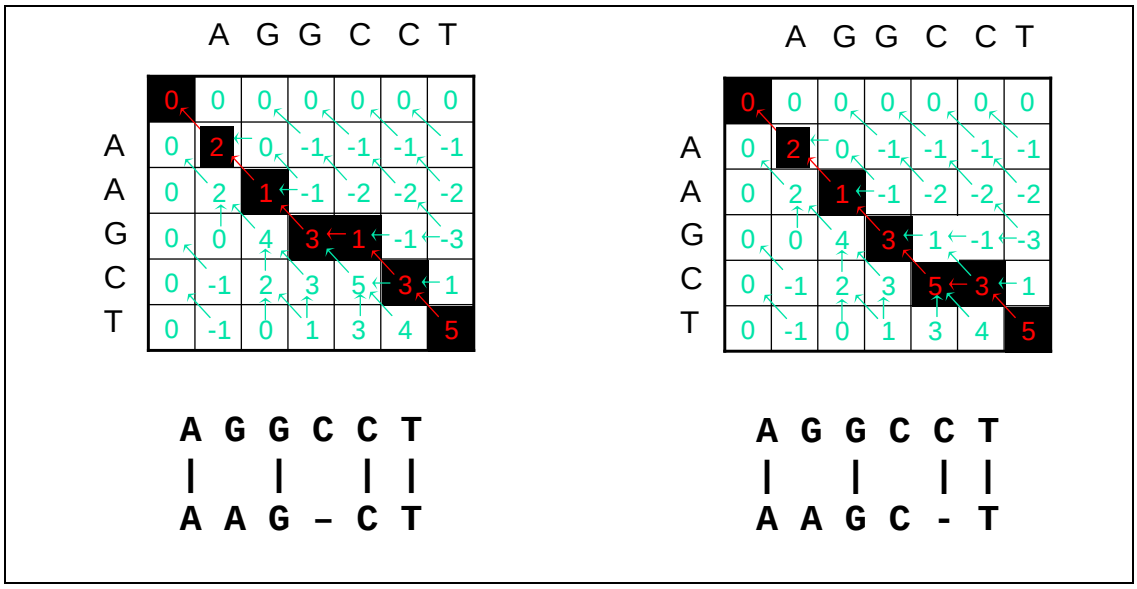
ในขั้นตอนนี้ จะหาเส้นทางกลับ จากค่าที่มีค่า Score ที่มากที่สุด หลังจากใส่ค่าในตารางคะแนนจนครบแล้วในขั้นตอนที่ 2 ค่า Score ที่มากที่สุดที่ได้จากทำ Global Alignment จาก 2 สายลำดับจะอยู่ที่ตำแหน่ง (m, n) ในตาราง ในตัวอย่างนี้ค่า Score ที่ได้คือ 5 ดูได้จากในภาพที่ 2-3 หลังจากนั้นจะตามลูกศรที่ชี้ตำแหน่งที่ให้ค่า Score นี้มา ถ้าลูกศรชี้ไปทาง ทิศตะวันตกเฉียงเหนือ หมายความว่าเป็นสายลำดับทั้งคู่ ถ้าลูกศรชี้ไปทางซ้ายหมายความว่า สายลำดับ 1 เป็นลำดับข้อมูล ส่วนสายลำดับ 2 เป็น Gap และถ้าลูกศรชี้ไปข้างบน หมายความว่าสายลำดับที่ 1 เป็น Gap ส่วน สายลำดับ 2 เป็นลำดับข้อมูล

ในขั้นตอนนี้อาจมีเส้นทางได้มากกว่า 1 เส้นทางที่ทำให้ได้ค่า Score มากที่สุด แต่ทุกเส้นทางจะได้ค่า Score เท่ากัน ดังในภาพที่ 2-3

2.2 Matching Score

ใน DNA ลำดับข้อมูลได้แก่ A ,T ,G,C ค่า Matching Score หาได้จากการกำหนดค่าให้ Matching Score เช่นดังในตัวอย่างที่ 2.1 ส่วนในโปรตีน ลำดับข้อมูลได้แก่ A, R, N, D, C, Q, E, G,

H, I, L, K, M, F, P, S, T, W, Y และ V ค่า Matching Score นั้นหาได้จาก Scoring Matrix ซึ่ง Scoring Matrix นั้นมีหลายตัว แต่ตัวที่ได้รับความนิยมส่วนมากได้แต่ PAM (Percent Accepted Mutation) และ BLOSUM (Blocks Substitution Matrix)



ภาพที่ 2-3 เส้นทางที่ได้จาก Score เท่ากับ 5

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	2																			
R	–2	6																		
N	0	0	2																	
D	0	–1	2	4																
C	–2	–4	–4	–5	4															
Q	0	1	1	2	–5	4														
E	0	–1	1	3	–5	2	4													
G	1	–3	0	1	–3	–1	0	5												
H	–1	2	2	1	–3	3	1	–2	6											
I	–1	–2	–2	–2	–2	–2	–2	–3	–2	5										
L	–2	–3	–3	–4	–6	–2	–3	–4	–2	2	6									
K	–1	3	1	0	–5	1	0	–2	0	–2	–3	5								
M	–1	0	–2	–3	–5	–1	–2	–3	–2	2	4	0	6							
F	–4	–4	–4	–6	–4	–5	–5	–5	–2	1	2	–5	0	9						
P	1	0	–1	–1	–3	0	–1	–1	0	–2	–3	–1	–2	–5	6					
S	1	0	1	0	0	–1	0	1	–1	–1	–3	0	–2	–3	1	3				
T	1	–1	0	0	–2	–1	0	0	–1	0	–2	0	–1	–2	0	1	3			
W	–6	2	–4	–7	–8	–5	–7	–7	–3	–5	–2	–3	–4	0	–6	–2	–5	17		
Y	–3	–4	–2	–4	0	–4	–4	–5	0	–1	–1	–4	–2	7	–5	–3	–3	0	10	
V	0	–2	–2	–2	–2	–2	–2	–1	–2	4	2	–2	2	–1	–1	–1	0	–6	–2	4

ภาพที่ 2-4 PAM 250 scoring matrix [15]

2.3 การประเมินค่าการเทียบเรียงลำดับข้อมูล

ในงานนี้จะวัดคุณภาพของการเทียบเรียงลำดับข้อมูลได้โดยพิจารณาที่ จุดมุ่งหมายของการเทียบเรียงแบบคู่ (Pairwise Alignment) คือ ค่ามากสุดของผลรวม Pairing Score สำหรับการเทียบเรียงกลุ่มลำดับข้อมูล (Multiple Sequence Alignment) จะใช้อยู่ในรูป Opjective Function

2.3.1 Sum-of-pairs (SP)

วิธีการ SP เป็นวิธีการที่ใช้ใน Pairwise Alignment วิธีการนี้เป็นวิธีการที่ได้รับความนิยมมากในงานวิจัยนี้ใช้วิธีการนี้ เป็นวิธีการหาค่า Scoring

SP Score สำหรับแต่ละคอลัมน์ ใน การเทียบเรียงกลุ่มข้อมูล ถูกคำนวณ โดยค่า Matching Score ระหว่างแต่ละคอลัมน์ ให้ $C_{x,I}$ คือค่าลำดับข้อมูลที่ตำแหน่งแถว x คอลัมน์ i ใน การเทียบเรียงกลุ่มข้อมูล และให้ S(C1,C2) คือค่า Matching Score สำหรับ ลำดับข้อมูล C1 และ C2 ใน Scoring Matrix ดังนั้น SP Score สำหรับคอลัมน์ i (SP(i)) คือ

$$SP(i) = \sum_{x<y} S\,(c_{x,i}, c_{y,i}) \qquad (2\text{-}1)$$

ผลรวม ของ SP Score สำหรับการเทียบเรียงกลุ่มลำดับข้อมูลคำนวณได้จาการรวมค่า SP Score ของแต่ละคอลัมน์ ดังนั้น

$$\text{SP score} = \sum_{1\le i\le n} SP(i) \qquad (2\text{-}2)$$

โดยที่ n คือความยาวของสายอักษรในกลุ่มลำดับข้อมูล ที่รวม Gap เมื่อ SP Score ถูกใช้ใน Objective Function การเทียบเรียงกลุ่มที่ดีที่สุดหาได้จาก กลุ่มลำดับข้อมูลที่ให้ค่า SP Score มากที่สุด

2.3.2 Entropy Scoring

วิธีการ Entropy Scoring เป็นวิธีการที่ให้ความสำคัญกับการคำนวณทางสถิติ [16] Entropy Score สำหรับการเทียบเรียงกลุ่ม หาได้จากผลรวมของ Entropy Score ของแต่ละคอลัมน์ ให้ Entropy Score ของคอลัมน์ i คือ Entropy(i) สามารถคำนวณได้จาก

$$\text{Entropy}(i) = -\sum_{a} c_{ia} \log p_{ia} \qquad (2\text{-}3)$$

โดยที่ Cia คือจำนวน ลำดับข้อมูล a ใน คอลัมน์ i และ Pia คือ ความน่าจะเป็นที่ ลำดับข้อมูล a จะอยู่ในคอลัมน์ i หาได้จาก

$$p_{ia} = c_{ia} / \sum_{a'} c_{ia'} \quad (2\text{-}4)$$

คอลัมน์ที่ให้ค่า Entropy Score เป็น 0 คือคอลัมที่มีลำดับข้อมูลทั้งหมดเหมือนกัน คอลัมน์ไหนที่มีลำดับข้อมูลหลายตัวจะให้ค่า Entropy Score ที่สูงกว่า เมื่อ Entropy Score ถูกใช้ใน Objective Function เป้าหมายคือการหาค่า Entropy Score ที่น้อยที่สุด

2.4 วิธีการเทียบเรียงแบบ Progressive Alignment

ตั้งแต่การหาค่าที่ดีที่สุดของการเทียบเรียงกลุ่มต้องการคำนวณอย่างมาก หลายๆ วิธีการได้ถูกคิดค้นพัฒนา เพื่อหาค่าที่ใกล้เคียงค่าที่ดีที่สุดของการเทียบเรียงกลุ่ม ในเวลาที่เหมาะสมในงานนี้สามารถแบ่งวิธีการหาได้ 2 แบบ คือ แบบ Progressive Alignment กับแบบ Iterative Alignment โดยแบบ Progressive Alignment จะได้รับการนิยมมากกว่า

แนวความคิดพื้นฐานของวิธีการ Progressive Alignment คือการนำเอาวิธีการ Pairwise Alignment นำมาเทียบเรียงซ้ำจนครบทุกสายลำดับข้อมูล ให้ n คือจำนวนสายลำดับข้อมูล L คือความยาวสายอักษร เวลาในการคำนวณสำหรับ Progressive Alignment คือ $O(nL^2)$ ขั้นตอนของ Progressive Alignment ได้แก่

1. เลือกสายลำดับข้อมูลมา 2 สายลำดับ เอามาทำการเทียบเรียงกันโดยใช้ Pairwise Alignment
2. เลือกสายลำดับข้อมูลอื่นมาทำการเทียบเรียงกันกับ กลุ่มสายข้อมูลปัจจุบัน
3. ทำตามขั้นตอนที่ 2 จนครบทุกสายลำดับข้อมูล

ในบางวิธีการสายลำดับข้อมูล ถูกเทียบเรียงในครั้งแรกรวมอยู่ในกลุ่มย่อย หลังจากนั้นค่อยรวมกลุ่มย่อยเป็นกลุ่มเดียวกัน วิธีการ Feng-Doolittle Progressive Alignment ได้ทำการประยุกต์ใช้วิธีการ Dynamic Programming ใน Feng-Doolittle Progressive Alignment กลุ่มของสายลำดับข้อมูลจะถูกรวมมาอยู่ในสายลำดับใน Profile 1 มิติ ก่อนที่จะไปทำการ Pairwise Dynamic Programming

2.4.1 Feng-Doolittle Progressive Alignment

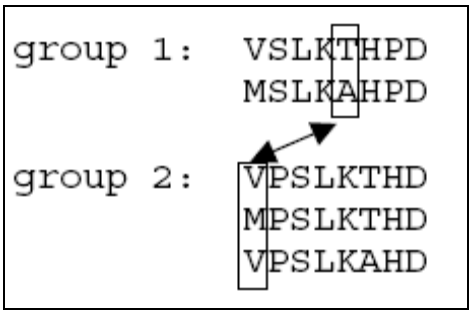
วิธี Feng-Doolittle Progressive Alignment ในส่วนของการหาค่า Matching Score ระหว่าง 2 ตำแหน่ง จาก 2 กลุ่ม หาได้จากการรวมค่า Matching Score ของทุกตัว แล้วนำมาเฉลี่ย ตัวอย่างการเทียบเรียงของ 2 กลุ่มสายลำดับข้อมูลในภาพที่ 2-5 ค่า Matching Score ของตำแหน่งในกลุ่ม

ลำดับข้อมูลแรกที่ลำดับข้อมูล T, A กับ ตำแหน่งข้อมูลในกลุ่มลำดับข้อมูลที่สองที่ตำแหน่ง V, M และ V คือ

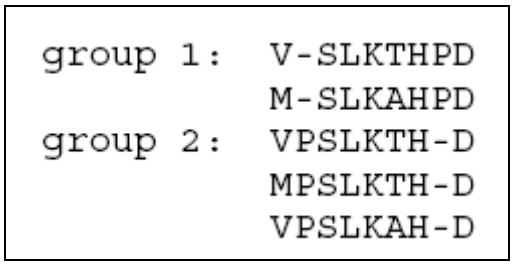
$$(S(T, V) + S(T,M)+S(T, V) + S(A, V) + S(A, M) +S(A,V))/6$$

โดยที่ S (C1,C2) คือค่า Matching Score สำหรับลำดับข้อมูล C1 และ C2 ใน Scoring Matrix ในงานนี้สามารถหา Dynamic Programming Scoring Table สำหรับ 2 กลุ่มลำดับข้อมูลโดยใช้ Pairwise Alignment Algorithm

การประยุกต์ใช้ใน Full Dynamic Programming มีการแก้ไขขั้นตอนหนึ่งในตอน Traceback จะต้องเรียงตามลำดับในการเทียบเรียง 2 กลุ่มลำดับข้อมูล ในขั้นตอนนี้ ถ้าเป็น Gapในกลุ่มลำดับข้อมูลจะต้องถูกเพิ่ม Gap เข้าไปทั้งกลุ่มดังภาพที่ 2-6



ภาพที่ 2-5 ตัวอย่างการเทียบเรียง 2 กลุ่มลำดับข้อมูล



ภาพที่ 2-6 การเพิ่ม Gap ลงในกลุ่มลำดับข้อมูล

กลุ่มลำดับข้อมูลที่ถูกเทียบเรียงโดย Feng-Doolittle Progressive Alignment ผลลัพธ์ของ Gap ที่ได้จากการเทียบเรียงจะถูกกำหนดแน่นอน หลังจากการเทียบเรียง Gap ใน แต่ละสายลำดับข้อมูลเปรียบเสมือนเป็นลำดับข้อมูลตัวหนึ่ง บนกฎ “Once a Gap, Always Ggap” Gap ทุกตัวที่ได้ตั้งแต่ตอนแรกหากผิดตำแหน่งก็จะทำให้ได้ผลผิด เพราะฉะนั้นลำดับในการเทียบเรียงจึงมีความสำคัญมาก

บทที่ 3
ขั้นตอนวิธีเชิงพันธุกรรม

ขั้นตอนวิธีเชิงพันธุกรรม (Genetic Algorithm) ซึ่งถูกพัฒนาขึ้นโดย Holland [17] เป็นเทคนิคหนึ่งซึ่งมีรากฐานมาจากกระบวนการทางชีวะภาพ ซึ่งกระบวนการหาคำนั้นใช้หลักการเรื่องการอยู่รอดของผู้ที่แข็งแรงที่สุด (Survival of the Fittest) ของชาร์ลส์ ดาร์วิน (Charles Darwin) กล่าวโดยสรุปขั้นตอนวิธีเชิงพันธุกรรมเป็นเทคนิคการหาคำตอบแบบเฟ้นสุ่ม (Stochastic Search) การทำงานของขั้นตอนวิธีเชิงพันธุกรรม นั้นก็จะเป็นไปในลักษณะของการหาคำตอบแบบคู่ขนาน (Parallel Search)โดยที่คำตอบที่หาได้ใน 1 รุ่น (Generation) จะผ่านการแปลง (Transformation) เพื่อที่จะได้คำตอบที่ดีขึ้นในรุ่นต่อไป การแปลงที่เกิดขึ้นกับผลคำตอบ (Solution) หรือสมาชิกของประชากร (Individual) ภายในประชากร (Population) 1 รุ่นนั้นเป็นไปเพื่อเป็นการสำรวจพื้นที่ในการค้นหาและส่งเสริมให้มีการถ่ายทอดคุณลักษณะที่ดี (Fit Characteristics) ของคำตอบที่ได้ค้นพบในรุ่นปัจจุบันไปยังรุ่นถัดไป สมาชิกที่แข็งแรงในแต่ละรุ่นจะมีส่วนที่ทำให้เกิดสมาชิกที่แข็งแรงที่สุด (Fittest Individual)

ขั้นตอนวิธีเชิงพันธุกรรมจะทำการหาคำตอบในพื้นที่การค้นหาคำตอบของตัวแปรตัดสินใจ (Decision Variable) ของฟังค์ชันที่ต้องการหาคำตอบ โดยที่ตัวแปรตัดสินใจจะมีการเข้ารหัสให้มีลักษณะคล้ายโครโมโซม (Chromosome) และการหาคำตอบจะเป็นการหาแบบหลายจุดพร้อมกัน (Parallel search) ทำให้โอกาสที่คำตอบที่หาได้จะเป็นค่าเหมาะที่สุดเฉพาะที่ (Local Optimal Value) นั้นลดลง ขั้นตอนวิธีเชิงพันธุกรรมนั้นจะอาศัยข้อมูลตอบแทน (Pay off Information) หรือค่าความแข็งแรงของสมาชิกของกลุ่มประชากรแต่ละตัว ในการกำหนดทิศทางการหาคำตอบในพื้นที่ของการค้นหา แตกต่างจากวิธีการทางแคลคูลัส คือวิธีการทางแคลคูลัสจะใช้ข้อมูลที่เป็นความชันของฟังก์ชันเป็นแนวทางในการหาคำตอบ ดังนั้นในกรณีที่ฟังก์ชันที่ต้องการหาคำตอบไม่สามารถหาความชันได้ก็ไม่สามารถใช้วิธีการทางแคลคูลัสได้ นอกจากนั้นขั้นตอนวิธีเชิงพันธุกรรมเป็นวิธีการหาคำตอบด้วยความน่าจะเป็น (Stochastic Search) แต่วิธีการทางแคลคูลัสจะใช้หลักการกำหนด(Deterministic)ในการหาคำตอบ

ขั้นตอนวิธีเชิงพันธุกรรมนั้นมีรากฐานมาจากทฤษฎีวิวัฒนาการ ดังนั้นคำศัพท์ต่างๆ ที่ใช้ในขั้นตอนวิธีเชิงพันธุกรรมจะเป็นคำศัพท์ทางชีววิทยา โดยจะสามารถแปลความหมายได้ตามที่แสดงไว้ในตารางที่ 3-1

ตารางที่ 3-1 ความหมายของคำศัพท์ในขั้นตอนวิธีเชิงพันธุกรรม

คำศัพท์ในขั้นตอนวิธีเชิงพันธุกรรม	ความหมาย
โครโมโซม (Chromosome)	สายรหัส
ยีน (Gene)	รหัส หรือ อักษร
อัลลีล (Allele)	ค่าของรหัส
โลคัส (Locus)	ตำแหน่งของรหัสบนสายรหัส
จีโนไทป์ (Genotype)	ลักษณะสายรหัส
ฟีโนไทป์ (Phenotype)	สิ่งที่ได้จากการถอดรหัสลักษณะสายรหัส

การทำงานของขั้นตอนวิธีเชิงพันธุกรรมเป็นการหาคำตอบแบบขนาน (Parallel Search) คำตอบที่ได้ในรุ่น (Generation) หนึ่ง จะผ่านการแปลง (Transformation) เพื่อทำให้เกิดคำตอบที่ดีขึ้นในรุ่นถัดไป กระบวนการทำงานของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่ายสามารถแสดงดังภาพที่ 3-1 และ 3-2

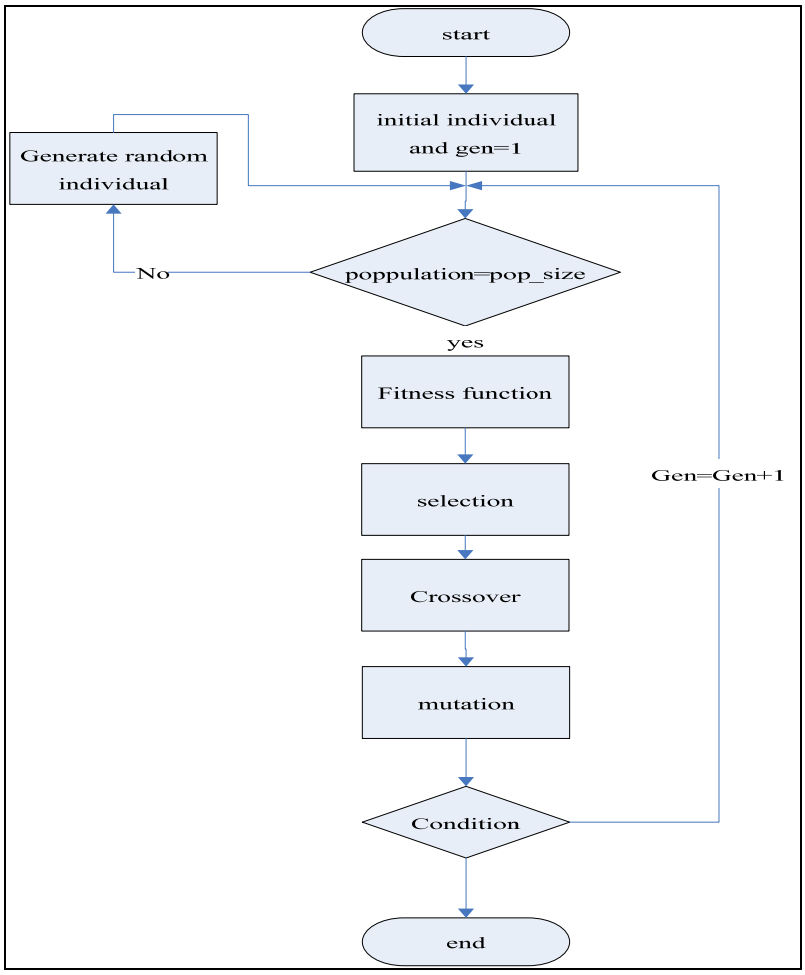
```
/*Genetic Algorithm*/
initialise a set of random individuals;
for (i=0;i< Max generation; i++){
        decision variable = decode(individual);
        fitness = fitness function(decision variable);
        selected individual = selection(individual, fitness);
        crossover individual = crossover(selected individual, p_c);
        mutation individual = mutation(crossover individual, p_m);
        individual = mutate individual;
}
```

ภาพที่ 3-1 กระบวนการทำงานของขั้นตอนวิธีเชิงพันธุกรรมอย่างง่าย

จากภาพที่ 3-2 ขั้นตอนวิธีเริ่มจากการสร้างประชากรเริ่มต้น (Initial Population) นำประชากรเริ่มต้นไปคำนวณค่าความแข็งแรง แล้วคัดเลือกประชากรที่แข็งแรง ไปทำการครอสโอเวอร์และการกลายพันธุ์ จบหนึ่งกระบวนการนี้เรียกว่า 1 รุ่น ในแต่ละลำดับขั้นตอนสามารถอธิบายเป็นหัวข้อได้ดังต่อไปนี้

3.1 การสร้างประชากรเริ่มต้น (Initial Population)

การสร้างประชากรเริ่มต้น เกิดจากสายรหัส (Chromosome) ของอักษรที่มาจากการสุ่ม ซึ่งรูปแบบที่ง่ายสุดในการเข้ารหัสคือเลขฐานสอง ดังนั้นอักษรที่ได้จากการสุ่มคือเลข 0 และ 1 การกำหนดจำนวนประชากรขึ้นอยู่กับการกำหนดของผู้เขียนโปรแกรมเอง การกำหนดจำนวนประชากรเริ่มต้นมากจะทำให้โอกาสในการพบคำตอบมาก แต่ก็ทำให้การคำนวณในแต่ละรุ่นช้าลง เพราะจำนวนประชากรในรุ่นมีมาก



ภาพที่ 3-2 ขั้นตอนการทำงานของวิธีเชิงพันธุกรรมอย่างง่าย

3.2 การเข้ารหัส (Encoding)

การเข้ารหัส เป็นการนำสายรหัสมาแปลงค่าหรือคำนวณเป็นตัวแปรตัดสินใจ รูปแบบของสายรหัสมีหลายแบบเช่น สายรหัสฐานสอง (Binary Chromosome) สายรหัสจัดอันดับ (Sequence Chromosome) และสายรหัสจำนวนเต็ม (Integer Chromosome) เป็นต้น ซึ่งรูปแบบของสายรหัส

ขึ้นอยู่กับตัวปัญหา ในที่นี้จะยกตัวอย่างวิธีการเข้ารหัสที่ง่ายที่สุดคือ รหัสฐานสอง (Binary Coded) ผลที่ได้จากการเข้ารหัสวิธีนี้คือ สายรหัสจะประกอบขึ้นจากกลุ่มของยีน (Gene) หลายกลุ่ม โดยที่แต่ละกลุ่มของยีนจะแทนค่าตัวแปรตัดสินใจ 1 ตัว และแต่ละยีนจะมีอัลลีลเป็น 0 หรือ 1

การเข้ารหัสสำหรับขั้นตอนวิธีเชิงพันธุกรรมเป็นเพียงการเปลี่ยนแปลงรูปแบบของค่าตัวแปรตัดสินใจ ให้มีความเหมาะสมและสะดวกกับการเปลี่ยนแปลงค่าคำตอบโดยตัวดำเนินการต่างๆ ของขั้นตอนวิธีเชิงพันธุกรรม

ค่าของตัวแปรตัดสินใจก่อนเข้ารหัสนั้น สมาชิกของกลุ่มประชากรจะเป็นแบบฟีโนไทป์ (Phenotype) และหลังจากเข้ารหัสนั้นสมาชิกของกลุ่มประชากรจะเป็นแบบจีโนไทป์ (Genotype) ซึ่งทั้งสองแบบนั้นหมายถึงสมาชิกของกลุ่มประชากรตัวเดียวกัน เพียงแต่รูปแบบการเสนอต่างกัน

3.3 การคำนวณค่าความแข็งแรง (Fitness Evaluation)

ค่าความแข็งแรงของสมาชิกแต่ละตัวในประชากร มีความสัมพันธ์โดยตรงกับค่าจุดประสงค์ที่เกิดจากการแทนค่าโดยสมาชิกของประชากรนั้นๆ ค่าความแข็งแรงเกิดจากการคำนวณฟังก์ชันจุดประสงค์ของตัวแปรตัดสินใจที่ได้จากการถอดรหัส โดยฟังก์ชันจุดประสงค์ของปัญหาการหา

ค่าสูงสุด (Maximization Problem) เหมาะที่สุดคือฟังก์ชันกำไร และปัญหาการหาค่าต่ำสุด (Minimization Problem) คือฟังก์ชันค่าใช้จ่าย ดังนั้นค่าความแข็งแรงของปัญหาการหาค่าสูงสุดคือ

$$f_i = Z_i \tag{3-1}$$

และค่าความแข็งแรงของปัญหาการหาค่าต่ำสุดคือ

$$f_i = C_{\max} - J_i \tag{3-2}$$

โดยที่ f_i คือ ค่าความแข็งแรงของประชากรตัวที่ i

Z_i คือ ค่าจุดประสงค์ซึ่งเป็นผลจากฟังก์ชันกำไรของประชากรตัวที่ i

J_i คือ ค่าจุดประสงค์ซึ่งเป็นผลจากฟังก์ชันค่าใช้จ่ายของประชากรตัวที่ i

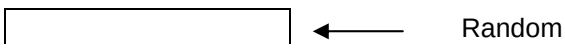
$C_{\max}$ คือ ค่าบวกค่าหนึ่งซึ่งมีค่ามากกว่าค่าจุดประสงค์ที่มากที่สุดที่เป็นไปได้ หาได้โดยอาศัยข้อมูลของปัญหาการหาค่าเหมาะที่สุดที่สนใจเป็นหลัก

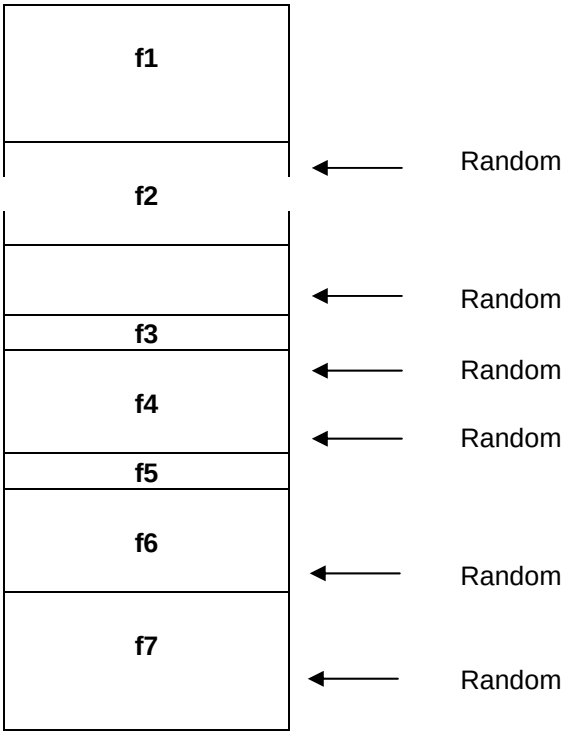
3.4 การคัดเลือกประชากร (Selection)

เป็นตัวดำเนินการที่มีความสำคัญมากในการทำงานของขั้นตอนวิธีเชิงพันธุกรรม ทำหน้าที่คัดเลือกประชากรภายในกลุ่มจากค่าความแข็งแรง ประชากรรุ่นใหม่จะถูกสร้างขึ้นโดยการแพร่พันธุ์ (Reproduction) จากประชากรที่แข็งแรงในรุ่นปัจจุบัน โดยใช้ค่าความแข็งแรงเป็นตัวกำหนดอัตราการแพร่พันธุ์ อัตราส่วนระหว่างค่าความแข็งแรงของประชากรแต่ละตัวกับผลรวมของค่าความแข็งแรงจากประชากรทุกตัวในรุ่นปัจจุบันจะเป็นตัวกำหนดสัดส่วน ของจำนวนประชากรนั้นๆ ในประชากรรุ่นใหม่ กล่าวคือประชากรในรุ่นปัจจุบันที่มีค่าความแข็งแรงมากมีโอกาสแพร่พันธุ์ไปในประชากรรุ่นใหม่สูง ในขณะที่ประชากรที่มีค่าความแข็งแรงน้อยจะมีโอกาสแพร่พันธุ์ไปในประชากรรุ่นใหม่ต่ำ ซึ่งเทคนิคที่ใช้ในการคัดเลือกนี้มีอยู่หลายเทคนิคด้วยกัน ในที่นี้เทคนิคที่จะกล่าวถึงคือ การคัดเลือกแบบวงล้อรูเล็ต (Roulette Wheel Selection) และการคัดเลือกโดยการชักตัวอย่างทุกตัวแบบเฟ้นสุ่ม (Stochastic Universal Sampling Selection)

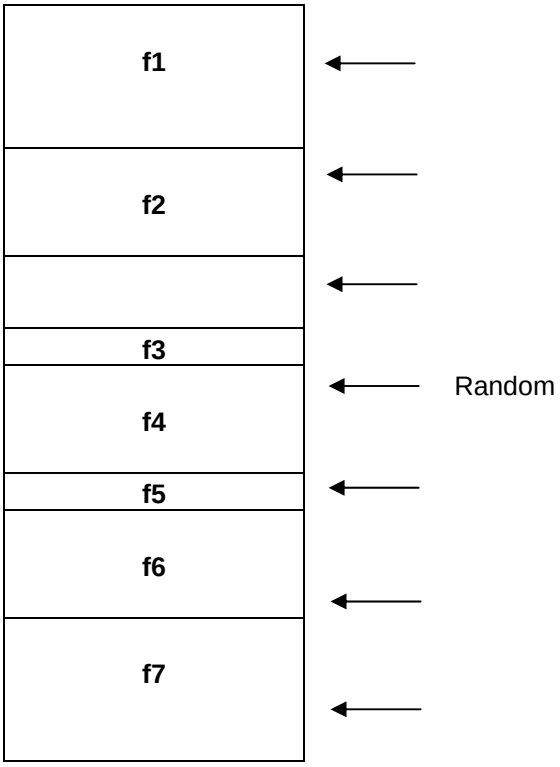
3.4.1 การคัดเลือกแบบวงล้อรูเล็ต (Roulette Wheel Selection) เป็นวิธีการเลือกที่เลียนแบบการเล่นวงล้อรูเล็ต หลักการทำงานคือ กำหนดความกว้างของช่องแต่ละช่อง ของเส้นจากค่าความแข็งแรงของสมาชิกแต่ละตัว ทำให้มีความลำเอียง จากนั้นทำการสุ่มจุดตก จะเลือกสมาชิกของกลุ่มประชากรที่มีตัวชี้ตำแหน่งชี้อยู่ทำเช่นนี้ซ้ำจนได้สมาชิกของกลุ่มประชากรครบตามจำนวนในหนึ่งรุ่น วิธีการเลือกลักษณะนี้จะเห็นได้ว่ามีความลำเอียงในการเลือกค่อนข้างมาก เนื่องจากถ้ามีสมาชิกของกลุ่มประชากรตัดใดที่มีค่าความแข็งแรงสูงจะมีโอกาสในการถูกเลือกซ้ำหลายครั้ง ทำให้สมาชิกของกลุ่มประชากรภายในรุ่นถัดไปของการทำงาน มีลักษณะของสมาชิกของกลุ่มประชากรตัวนั้นหลายตัว

3.4.2 การเลือกสุ่มตัวอย่างแบบเฟ้นสุ่มสากล (Stochastic Universal Sampling Selection) จะมีหลักการเหมือนกับการเลือกตามวงล้อรูเลต ต่างกันที่หลังจากกำหนดจุดชี้ตำแหน่งโดยการสุ่มในครั้งแรกแล้ว ทำการเลือกสมาชิกของกลุ่มประชากรที่มีตัวชี้ตำแหน่งชี้อยู่เป็นตัวแรก ต่อจากนั้นทำการเลื่อนตัวชี้ตำแหน่งจากจุดเดิมทีละขั้น โดยที่แต่ละขั้นนั้นจะเท่ากับผลรวมของค่าความแข็งแรงต่อจำนวนสมาชิกของกลุ่มประชากร แล้วทำการเลือกสมาชิกของกลุ่มประชากรที่มีตัวชี้ตำแหน่งชี้อยู่จนครบตามจำนวนสมาชิกของกลุ่มประชากรในหนึ่งรุ่น ดังนั้นการคัดเลือกพันธุ์แบบการสุ่มตัวอย่างแบบเฟ้นสุ่มสากลนี้สามารถลดความลำเอียงในการคัดเลือกได้ เนื่องจากโอกาสที่สมาชิกของกลุ่มประชากรตัวใดจะถูกเลือกซ้ำหลายๆครั้ง จะเกิดขึ้นต่อเมื่อสมาชิกของกลุ่มประชากรตัวนั้นๆ มีค่าความแข็งแรงสูงมากๆ ตัวอย่างการทำงานของการสุ่มตัวอย่างแบบเฟ้นสุ่มสากลแสดงได้ตามภาพที่ 3-4





ภาพที่ 3-3 การคัดเลือกแบบวงล้อรูเล็ต

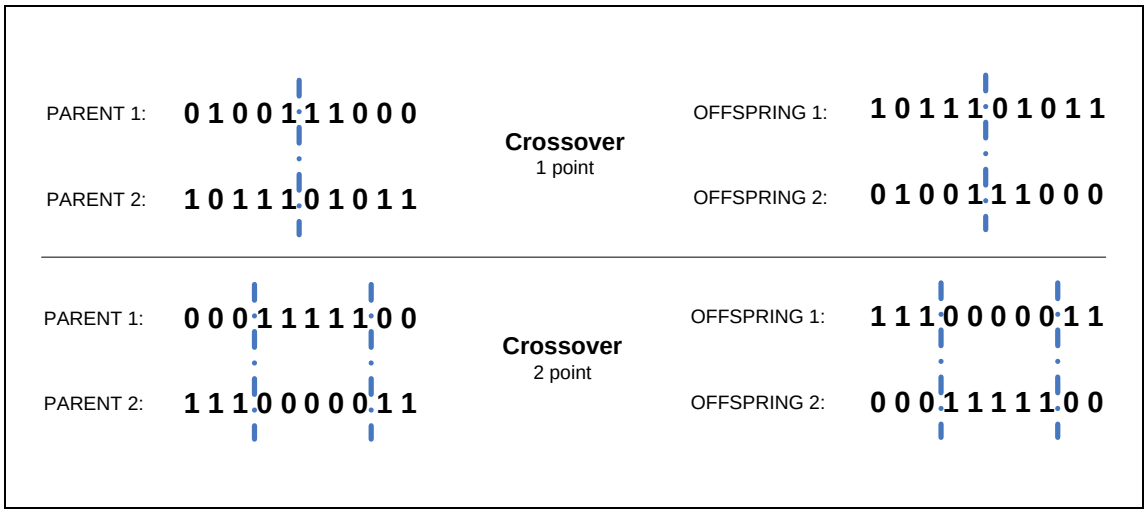


ภาพที่ 3-4 การทำงานของการเลือกสุ่มตัวอย่างแบบเฟ้นสุ่มสากล

3.5 การครอสโอเวอร์ (Crossover)

การครอสโอเวอร์ทำหน้าที่ถ่ายทอดลักษณะของประชากรจากรุ่นหนึ่งไปยังอีกรุ่นหนึ่ง โดยมีการสับเปลี่ยนโครงสร้างทางพันธุกรรมจากประชากรที่ถูกคัดเลือกมา โดยการสุ่มเลือกสมาชิกของประชากรรุ่นพ่อแม่ (Parent Individual) มาจำนวน 2 ตัว ส่งผ่านไปยังสมาชิกของประชากรรุ่นลูก (Offspring Individual) ซึ่งจะมีจำนวน 2 ตัวเช่นกัน โอกาสที่จะมีการครอสโอเวอร์เกิดขึ้นจะถูกกำหนดโดยความน่าจะเป็นในการครอสโอเวอร์ (Crossover Probability) โดยปกติความน่าจะเป็นในการครอสโอเวอร์มีค่าอยู่ระหว่าง 0.7 - 0.9 ซึ่งหมายความว่าถ้าไม่มีการครอสโอเวอร์เกิดขึ้นแล้ว สมาชิกของประชากรรุ่นลูกก็จะเหมือนกับสมาชิกของประชากรรุ่นพ่อแม่ทุกประการ เทคนิคการครอสโอเวอร์ที่จะกล่าวถึงในที่นี้ ได้แก่ การครอสโอเวอร์แบบ N จุด (N-Point Crossover) และการครอสโอเวอร์แบบเอกรูป (Uniform Crossover)

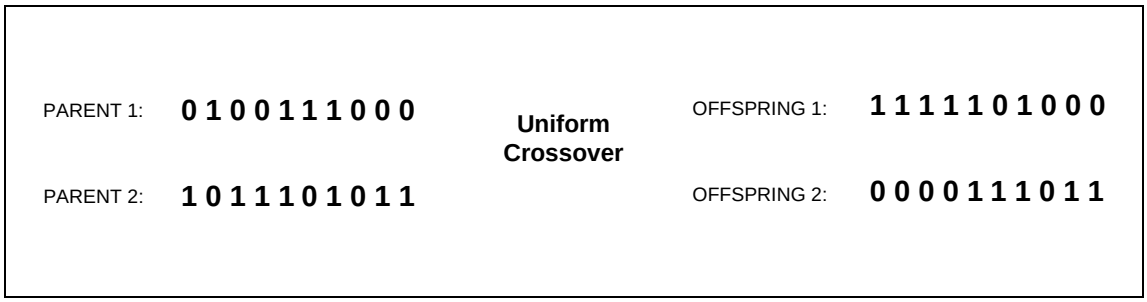
3.5.1 การครอสโอเวอร์แบบ N จุด (N-Point Crossover) เทคนิคนี้การแลกเปลี่ยนยีนระหว่างสมาชิกของประชากรรุ่นพ่อแม่เพื่อให้เกิดการสร้างสมาชิกของประชากรรุ่นลูกนั้น จะเกิดขึ้น ณ ข้างใดข้างหนึ่งของตำแหน่งการครอสโอเวอร์ (Crossover Site) หรือเกิดขึ้นระหว่างตำแหน่งการครอสโอเวอร์ 2 ตำแหน่งบนโครโมโซม ค่า N จะเป็นส่วนที่กำหนดจำนวนของตำแหน่งการครอสโอเวอร์โดยที่ $N \geq 1$ ดังภาพที่ 3-5



ภาพที่ 3-5 การครอสโอเวอร์แบบ 1 จุด และ การครอสโอเวอร์แบบ 2 จุด

3.5.2 การครอสโอเวอร์แบบเอกรูป (Uniform Crossover) เป็นการแลกเปลี่ยนยีนที่แต่ละโลคัสของโครโมโซมในประชากรรุ่นลูก 1 ตัว จะถูกเลือกอย่างสุ่มจากคู่ยีนที่มาจากสมาชิกของประชากรรุ่นพ่อแม่ทั้งสองตัว ดังนั้นโอกาสที่ยีนจะถูกเลือกจากสมาชิกของประชากรรุ่นพ่อแม่ตัวหนึ่ง

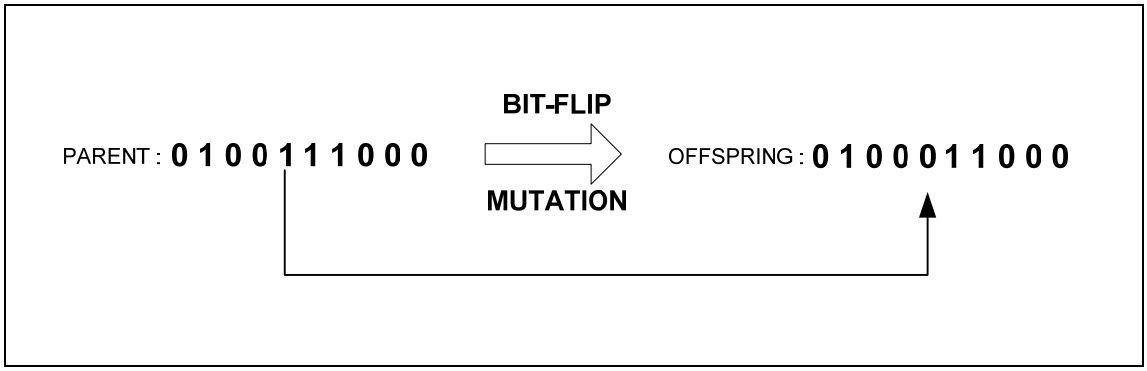
จะเท่ากับโอกาสที่ยีนจะถูกเลือกจากสมาชิกของประชากรรุ่นพ่อแม่อีกตัวหนึ่ง ซึ่งหลังจากที่ยีนสำหรับสมาชิกของประชากรรุ่นลูกที่สนใจได้ถูกกำหนดแล้ว ยีนจากคู่ยีนที่ไม่ได้ถูกเลือกก็จะถูกส่งผ่านไปยังสมาชิกของประชากรรุ่นลูกอีกตัวที่เหลือ ดังภาพที่ 3-6



ภาพที่ 3-6 การครอสโอเวอร์แบบเอกรูป

3.6 การกลายพันธุ์ (Mutation)

การกลายพันธุ์เป็นกระบวนการที่ใช้ในการสร้างประชากรใหม่จากประชากรที่มีอยู่เดิม ซึ่งจะส่งผลให้สมาชิกของประชากรรุ่นใหม่ หรือรุ่นลูกที่เกิดขึ้นมีลักษณะที่ดีกว่าสมาชิกของประชากรรุ่นพ่อแม่ปัจจุบัน โดยค่าความน่าจะเป็นในการกลายพันธุ์จะอยู่ระหว่าง 0 ถึง 0.1 การกลายพันธุ์เป็นวิธีที่ทำให้เกิดการเปลี่ยนแปลงของยีนเพียงเล็กน้อยในโครโมโซมของสมาชิก ของประชากร การเปลี่ยนแปลงดังกล่าวทำให้เกิดการค้นหาคำตอบ ซึ่งอยู่ในตำแหน่งใกล้เคียงกับคำตอบซึ่งแทนโดยสมาชิกของประชากรตัวเดิมในพื้นที่การค้นหา ในกรณีที่โครโมโซมฐานสองถูกใช้ในการแทนค่าผลเฉลย การกลายพันธุ์จะสามารถทำได้โดยการเปลี่ยนค่าบิตจาก 0 เป็น 1 หรือจาก 1 เป็น 0 การกลายพันธุ์ในลักษณะนี้เรียกว่า การกลายพันธุ์โดยกลับค่าบิต (Bit-Flip Mutation) ดังภาพที่ 3-7



ภาพที่ 3-7 การกลายพันธุ์โดยกลับค่าบิต

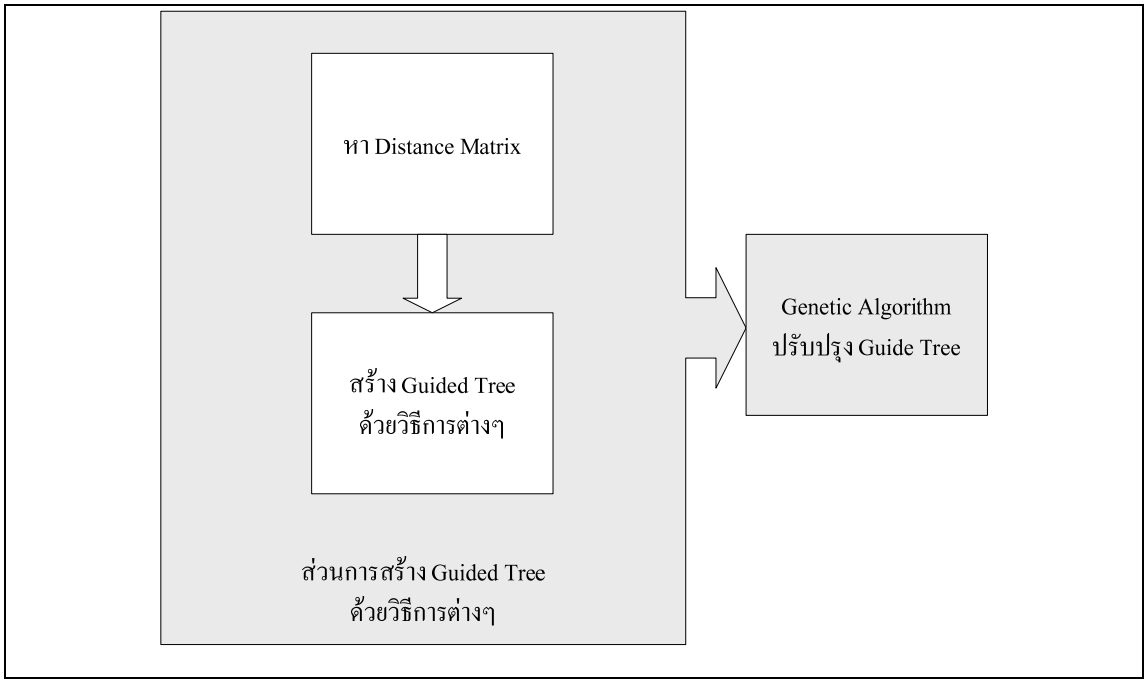
บทที่ 4
ขั้นตอนวิธีเชิงพันธุกรรมในการสร้างลำดับการเทียบเรียงกลุ่มลำดับ

การเทียบเรียงกลุ่มลำดับข้อมูลทางชีววิทยาโมเลกุล (Multiple Sequence Alignment) เป็นหนึ่งในเครื่องมือที่มีความสำคัญในงานทางด้านชีววิทยาโมเลกุล การเทียบเรียงกลุ่มลำดับข้อมูลนำไปใช้ประโยชน์ได้หลายทางเช่น เอาไปใช้ประโยชน์ในการทำนายโครงสร้างของโปรตีน (Structure Predition) การวิเคราะห์วิวัฒนาการของสิ่งมีชีวิต (Phylogenetic Analysis) การออกแบบ Primer ในปฏิกิริยาพีซีอาร์ และในกรณีของโปรตีนใช้หาว่าช่วงใดมีความสำคัญควรนำไปศึกษาหรือวิเคราะห์ต่อไป เป็นต้น การเทียบเรียงกลุ่มลำดับข้อมูลคือการเปรียบเทียบความเหมือนกันของสายลำดับตั้งแต่3 สายลำดับเป็นต้นไป เป็นแบบ Global Alignment การเทียบเรียงกลุ่มมีวิธีการทำอยู่ 2 วิธีคือ 1 แบบ Iterative กับแบบ Progressive แบบ Progressive จะเป็นที่ได้รับความนิยมมากกว่า โดยจะมีโปรแกรม ClustalW เป็นตัวที่ได้รับความนิยมมากที่สุด คุณภาพของการเทียบเรียงแบบ Progressive ขึ้นอยู่กับลำดับการเทียบเรียง เนื่องจากลำดับการเทียบเรียงที่ต่างกันจะให้ผลที่ไม่เหมือนกัน ปัญหาของการเทียบเรียงแบบ Progressive คือการหาลำดับการเทียบเรียงที่ดีที่สุด การหาลำดับการเทียบเรียงในปัจจุบันมีอยู่หลายวิธี โดยส่วนใหญ่จะมีวิธีการทำงานเหมือนกันคือ ขั้นตอนที่ 1 หา Distance Matrix จาก การทำ Pairwise Alignment ทุกคู่ลำดับจำนวน n(n-1)/2 คู่ แล้วเปลี่ยนค่าความคล้ายกัน เป็นDistance เก็บอยู่ในรูป Matrix ขนาด n×n โดย n คือจำนวน Sequence ขั้นตอนที่ 2 นำ Distance Matrix ที่ได้มาทำการหา ลำดับการเทียบเรียงด้วยวิธีการต่างๆ ขั้นตอนที่สาม ทำการเทียบเรียงตามลำดับการเทียบเรียงที่ได้ วิธีการส่วนใหญ่ ขั้นตอนที่ 1 และขั้นตอนที่ 3 จะเหมือนกัน จะต่างกันที่ขั้นตอนที่ 2 แต่เวลาที่ใช้ในการคำนวณส่วนมากจะอยู่ในขั้นตอนที่ 1 ส่วน ขั้นตอนที่ 2 และ 3 จะใช้เวลาน้อยมากเมื่อเทียบกับขั้นตอนที่ 1 วิธีการหาลำดับการเทียบเรียงด้วยวิธีการต่างๆ ยังบอกไม่ได้ว่าวิธีไหนจะให้ Alignment ที่ดีกว่าวิธีไหนสำหรับการเทียบเรียงกลุ่มใดๆ

ในวิทยานิพนธ์นี้เสนอวิธีการหาลำดับการเทียบเรียง โดยใช้ขั้นตอนวิธีเชิงพันธุกรรมในการปรับปรุงประสิทธิภาพของลำดับการเทียบเรียง ที่สร้างขึ้นจากวิธีการต่างๆที่มีอยู่ในปัจจุบันให้มีประสิทธิภาพดีขึ้น โดยในงานนี้ได้ทดลองปรับปรุงลำดับการเทียบเรียงที่ได้จากโปรแกรม ClustalW และจากวิธี Minimum Spaning Tree โดยใช้ค่า Sum of Pairs(SP) เป็นตัววัดคุณภาพของการเทียบเรียงในงาน

จากการที่วิธีการหาลำดับการเทียบเรียงด้วยวิธีที่มีอยู่ในปัจจุบัน มีขั้นตอนทำงานเหมือนกัน จากการที่ยังบอกไม่ได้ว่าวิธีการไหนจะดีกว่าวิธีการไหน จากการที่เวลาในการคำนวณในส่วนการสร้างลำดับการเทียบเรียงและ การเทียบเรียงตามลำดับการเทียบเรียง ใช้เวลาในการคำนวณไม่มาก เมื่อเทียบกับขั้นตอนการสร้าง Distance Matrix และจากการที่ขั้นตอนวิธีเชิงพันธุกรรม นั้นใช้หลักการเรื่องการอยู่รอดของผู้ที่แข็งแรงที่สุด (Survival of the Fittest) โดยในงานนี้หวังว่า การที่ใช้ขั้นตอนวิธีเชิงพันธุกรรมจะทำให้ได้ลำดับการเทียบเรียงที่ดีขึ้น โดยให้ลำดับการเทียบเรียง เป็นประชากรของขั้นตอนวิธีเชิงพันธุกรรม SP Score เป็นFitness Function และกำหนดให้ประชากรเริ่มต้นเป็น ลำดับการเทียบเรียงที่ได้จากวิธีการต่างๆ โดยในงานนี้หวังว่าการที่ประชากรรุ่นพ่อแม่ดีที่จะทำให้ประชากรรุ่นลูกที่ดี

โดยในงานวิจัยนี้แบ่งการทำงานเป็น 2 ส่วนคือส่วนการสร้างลำดับการเทียบเรียงด้วยวิธีที่มีอยู่กับส่วนของการใช้ขั้นตอนวิธีเชิงพันธุกรรมในการปรับปรุงประสิทธิภาพของลำดับการเทียบเรียงที่ได้มาจากส่วนแรก ตามภาพที่ 4-1 ขั้นตอนหา Distance Matrix ในส่วนแรก และขั้นตอนการหาค่า Fitness Function ในส่วนขั้นตอนเชิงพันธุกรรม ได้นำโปรแกรม ClustalW มาเขียนโปรแกรมเพิ่มเติมให้สามารถหาค่าในสองขั้นตอนนี้ได้ เหตุผลที่เลือกโปรแกรม ClustalW มาใช้เนื่องจากโปรแกรม ClustalW เป็นโปรแกรมที่ได้รับความนิยมมากที่สุด มีความน่าเชื่อถือสูง มีการพัฒนาโปรแกรมต่อเนื่อง และ สามารถนำไปประยุกต์ใช้กับงานต่างๆ ทางด้านชีววิทยาโมเลกุลได้



ภาพที่ 4-1 ขั้นตอนการทำงาน แบ่งเป็น 2 ส่วนคือ ส่วนการสร้าง ลำดับการเทียบเรียง และส่วนการ ใช้ขั้นตอนวิธีเชิงพันธุกรรมในการปรับปรุงลำดับการเทียบเรียง

4.1 การสร้าง Guided Tree ด้วยวิธีการต่างๆ

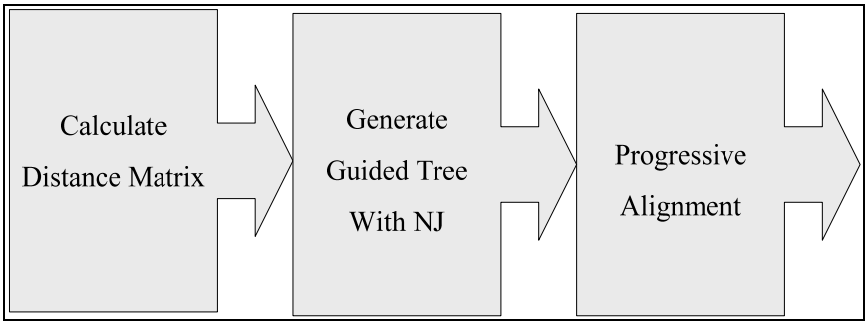
มีงานวิจัยเกี่ยวกับการเทียบเรียงแบบ Progressive อยู่หลายงาน ในวิทยานิพนธ์ได้ทดลองนำวิธีการสร้าง ลำดับการเทียบเรียงมาจางานวิจัยต่างๆ 2 งานได้แก่ 1 Thomson (ClustalW) 2 Mei-Jie Zhu(MST) โดยในงานวิจัยของ Thomson และ Mei-Jie มีขั้นตอนการทำงานดังนี้

4.1.1 Thomson (ClustalW) แบ่งการทำงานหลักๆได้ดังภาพที่ 4-2

4.1.1.1 วิธีการสร้าง Distance Matrix โปรแกรม ClustalW ใช้วิธีการ Pairwise Alignment โดยใช้วิธี Dynamic Programming เทียบเรียงทุกคู่ลำดับจำนวน n(n-1)/2 เสร็จแล้วนำค่าร้อยละความเหมือนที่ได้มาเปลี่ยนเป็นเป็นระยะทางโดยหารด้วยร้อย แล้วลบออกจาก 1 โดยที่ n คือจำนวนสายข้อมูล

4.1.1.2 วิธีการสร้าง Guided Tree โปรแกรม ClustalW ใช้วิธีการ Neighbor-joining [11] ในการสร้าง Guided Tree

4.1.1.3 ทำการเทียบเรียงลำดับข้อมูลตาม Guide Tree โดยโปรแกรม ClustalW ใช้วิธีการเทียบเรียงแบบ Feng-Doolittle Progreesive Alignment ในการเทียบเรียง



ภาพที่ 4-2 ขั้นตอนการทำงานของโปรแกรม ClustalW มี3 ขั้นตอน 1 หา Distance Matrix 2 หาลำดับการเทียบเรียง และ 3 เทียบเรียงตามลำดับการเทียบเรียง

4.1.2 งานวิจัยของ Mei-Jie Zhu แบ่งการทำงานได้ดังภาพที่ 4-3

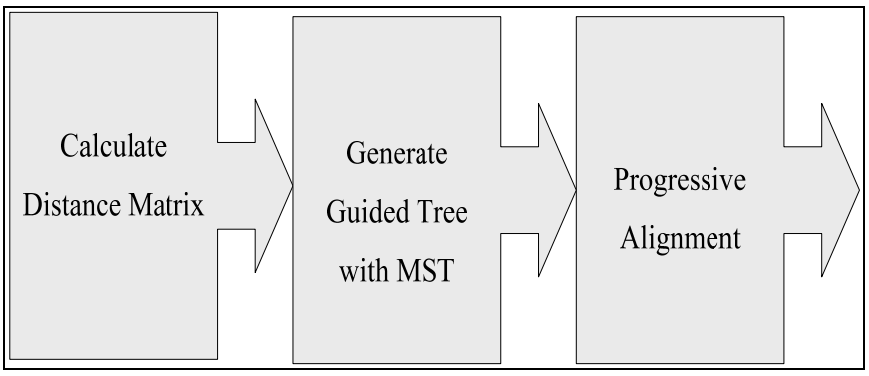
4.1.2.1 หา Distance Matrix ในงานวิจัยของ Mei-Jieใช้วิธีการ Pairwise Alignment โดยใช้วิธี Full Dynamic Programming ทุกคู่ลำดับจำนวน n(n-1)/2 เสร็จแล้วนำค่า SP Score มาแปลงเป็น Distance Matrix โดย

$$Distance_{ij} = SPmax\text{-}Sp_{ij} + 1$$

โดยที่ i ,j คือ ลำดับข้อมูลตัวที่ i และ j SPmax คือ ค่า SP Score ที่มากที่สุดในทุกคู่ลำดับ Sp_{ij} คือค่า SP Score ของลำดับข้อมูลตัวที่ i กับ ลำดับข้อมูลตัวที่ j แต่ในงานวิจัยนี้จะใช้ Distance Matrix ที่ได้จากโปรแกรม ClustalW

4.1.2.2 หา Guided Tree ในงานวิจัยนี้หา Guide Tree โดยหาจากวิธี Minimun Spanning Tree (MST) โดยใช้วิธีของ Kruskal Algorithm

4.1.2.3 ทำการเทียบเรียงลำดับข้อมูลตาม Guided Tree ในงานวิจัยของ Mei-Jie Zhu ใช้วิธีการเทียบเรียงแบบ Feng-Doolittle Progressive Alignment ในการเทียบเรียงในงานวิจัยนี้ จะทำการเทียบเรียงโดยใช้ โปรแกรม ClustalW ในการเทียบเรียง



ภาพที่ 4-3 ขั้นตอนการทำงานของ Mei-Jie Zhu แบ่งขั้นตอนเป็น 3 ขั้นตอน 1 หาDistance Matrix 2 หาลำดับการเทียบเรียง และ 3 เทียบเรียงตามลำดับการเทียบเรียง

การนำวิธีการหาลำดับการเทียบเรียงมาใช้ในวิทยานิพนธ์นี้ทำโดย เริ่มแรกจะทำการใช้โปรแกรม Clustalw ที่ได้ทำการดัดแปลง มาหา Distance Matrix และหาลำดับการเทียบเรียงด้วยวิธี Neighbor-Joining หลังจากนั้นจะทำการนำ Distance Matrix ที่ได้มาทำการหาลำดับการเทียบเรียงด้วยวิธี Minimum Spaning Tree ด้วย Kruskal Algorithm โดย Kruskal Algorithm มีขั้นตอนการทำงานดังนี้

วิธี Kruskal Algorithm

กำหนดให้ M เป็นที่เก็บกลุ่มของจุดโดยที่แต่ละกลุ่มไม่มีจุดร่วม และให้ Q เป็นที่เก็บเส้นโยงทั้งหมดของกราฟ G=(V,E,W) โดยที่เส้นโยงที่เก็บอยู่ใน Q ถูกเรียงตามลำดับน้ำหนักจากน้อยไปมาก ต้นไม้แบบแผ่ที่เล็กที่สุดจะถูกเก็บอยู่ใน T ขั้นตอนต่างๆ มีดังภาพที่ 4-4

หลังจากที่ได้ลำดับการเทียบเรียงและ Distance Matrix จากโปรแกรม Clustalw หลังจากนั้นนำ Distance Matrixที่ได้ไปหาลำดับการเทียบเรียงด้วยวิธี Kruskal ถ้าต้องการเพิ่มลำดับการเทียบเรียงด้วยวิธีอื่นก็สามารถเพิ่มวิธีการได้ในขั้นตอนนี้ หลังจากเราจะนำลำดับการเทียบเรียงที่ได้นำไปให้ขั้นตอนวิธีเชิงพันธุกรรมทำการปรับปรุงประสิทธิภาพต่อไป

```
1.  T = เซตว่าง ;
2.  M = เซตว่าง ;
3.  เก็บเส้นโยงต่างๆของ G ใน Q โดยเรียงตามน้ำหนักจากน้อยไปมากแต่ละเส้นโยง
    จะถูกแทนด้วยจุด 2 จุดที่เส้นโยงนั้นเกิดกบมัน;
4.  for all v เป็นสมาชิกของ V do M=M ยูเนียน{{v}} ;
5.  while จำนวนกลุ่มใน M > 1 do
6.  begin
7.      เลือกเส้นโยง {v,w} ที่อยู่ใน Q ที่มีน้ำหนักน้อยที่สุด ;
8.      ลบเส้นโยง {v,w}ออกจาก Q ;
9.      if v และ w อยู่คนละกลุ่มใน M นั้นคือ v อยู่ในกลุ่ม A และ w อยู่ในกลุ่ม B then
10.         begin
11.           แทนกลุ่ม A และกลุ่ม B.ใน M ด้วยกลุ่ม (A ยูเนียน B);
              ให้ T = T ยูเนียน {{v,w}};
12.         end
13. end
```

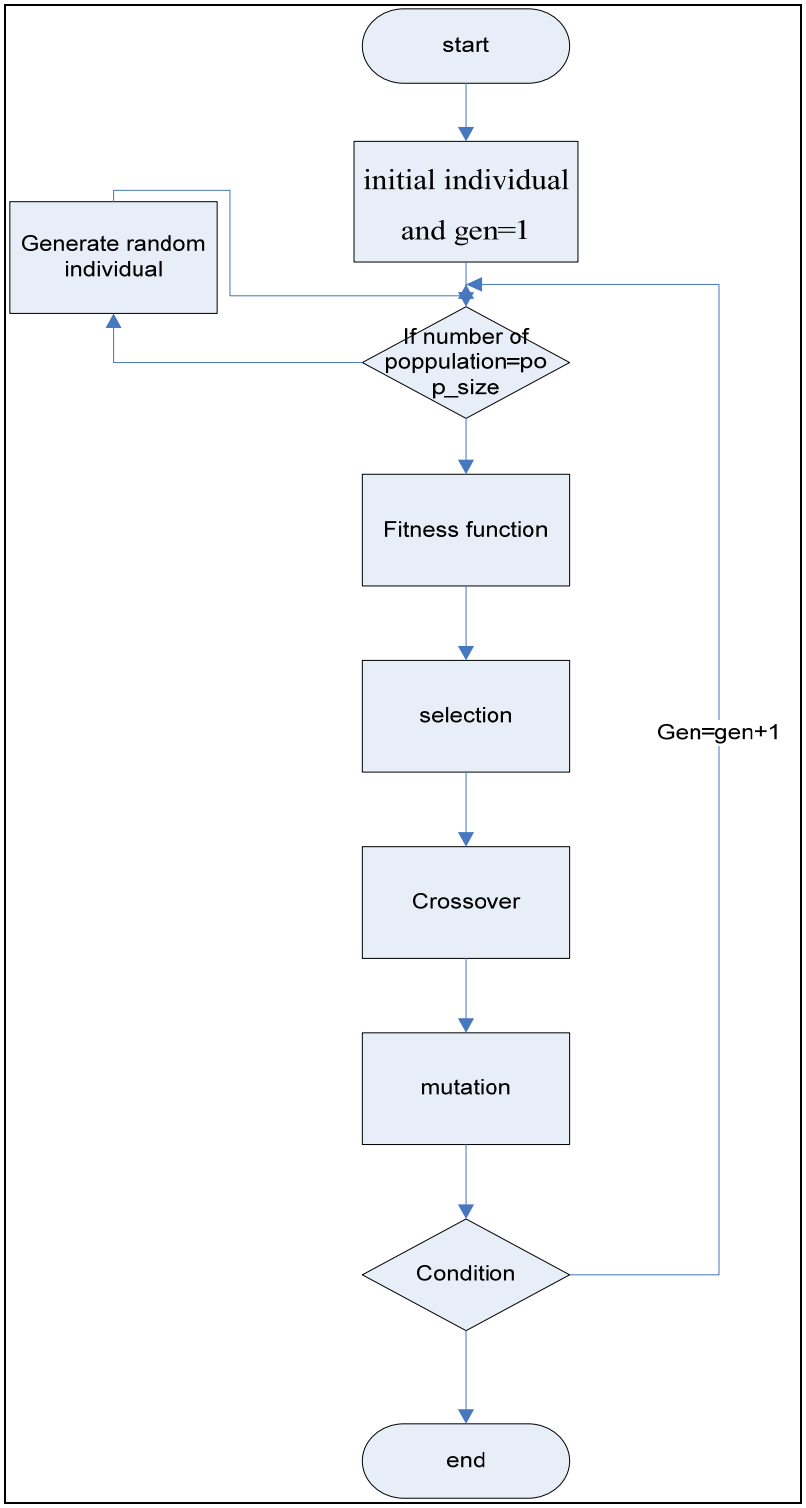
ภาพที่ 4-4 ขั้นตอนการทำงานของวิธี Kruskal Algorithm

4.2 การใช้ขั้นตอนวิธีเชิงพันธุกรรมในการปรับปรุง Guided Tree

ขั้นตอนวิธีเชิงพันธุกรรมดูได้จากในภาพที่ 4-5 ประชากรของขั้นตอนวิธีเชิงพันธุกรรมนี้คือ Guided Tree

ตารางที่ 4-1 ตัวดำเนินการและค่าต่างๆ ของขั้นตอนวิธีเชิงพันธุกรรม

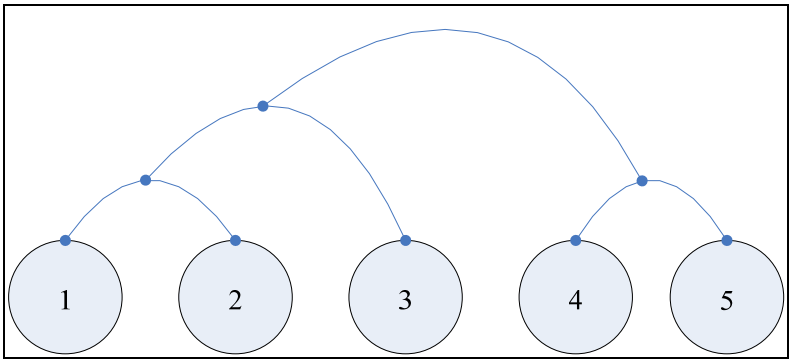
พารามิเตอร์	ค่าที่กำหนดใช้
ประชากร	Guided tree
ความยาวสายรหัส	2(n-1) โดยที่ n คือจำนวนลำดับข้อมูล
จำนวนประชากร	50
จำนวนเจนเนอเรชัน	50
จำนวนครั้งที่ทำการจำลอง	10
วิธีที่เลือกใช้ในฟังก์ชัน - Selection - Crossover - Mutation	 - Stochastic Universal Sampling - 1 Cut Point Crossover ($p_c = 0.9$) - Bit-flip Mutation ($p_m = 0.3$)



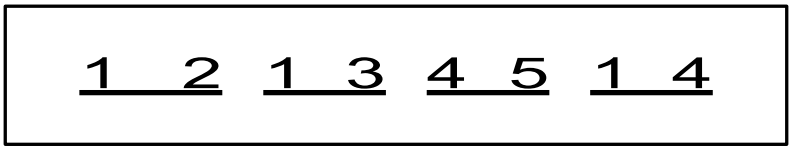
ภาพที่ 4-5 ขั้นตอนการทำงานของขั้นตอนวิธีเชิงพันธุกรรม

4.2.1 การสร้างประชากรเริ่มต้น (Intitial Population) ในการสร้างประชากรเริ่มต้นจะนำ Guided Tree ที่ได้จากวิธีการ Neighbor-Joining และจากวิธีการ Minimum Spanning Tree with Kruskal Algorithm เข้ามาเป็นประชากรเริ่มต้น แล้วทำการสร้างประชากรที่เหลือให้ครบตาม Population Size โดยการสร้าง Guided Tree แบบสุ่มโดยใช้หลักความน่าจะเป็น

การแปลงจาก Guided Tree เป็นประชากร ตัวอย่าง Guided Tree ในภาพที่ 4-6 ในการแปลง Guided Tree เป็นประชากร จะแปลงจากใบทางซ้ายมือสุดมาทีละคู่ โดยจะแทนกลุ่มของลำดับข้อมูลที่ถูกเทียบเรียงแล้วด้วยลำดับที่น้อยที่สุดของกลุ่มในตัวอย่างนี้คือ Sequence1 และ Sequence2 หลังจากนั้นจะทำการเลือกคู่ต่อไปโดยจะพิจารจากคู่ที่แล้วว่าถึงรากของต้นไม้หรือยัง ถ้ายังไม่ถึงรากของต้นไม้ จะทำการเลือกในลำดับถัดมาจนถึงราก โดยจะแทนคู่ที่ถูกเลือกแล้วด้วย Sequence ลำดับน้อยที่สุดในกลุ่มนั้น ในตัวอย่างนี้คู่ที่ถูกเลือกมาคือ Sequence1 และ Sequence3 ทำตามวิธีนี้ต่อไปจนครบจะได้ประชากร ดังภาพที่ 4-7



ภาพที่ 4-6 ตัวอย่าง Guide Tree



ภาพที่ 4-7 ประชากรที่สร้างได้จาก Guide Tree ในภาพที่ 4-6

4.2.2 การคำนวณค่าความแข็งแรง (Fitness Evaluation) ในการคำนวณค่าความแข็งแรงในงานวิจัยนี้ค่าความแข็งแรงคิดจาก SP Score ที่ได้จากการทำ Progreesive Alignment ของโปรแกรม ClustalW ตามประชากร (Guided Tree)

วิธีการหาค่า SP Score สำหรับแต่ละคอลัมน์ ใน การเทียบเรียงกลุ่มข้อมูล ถูกคำนวณ โดยค่า Matching Score ระหว่างแต่ละคอลัมน์ ให้ $C_{x,I}$ คือค่าลำดับข้อมูลที่ตำแหน่งแถว x คอลัมน์ i ในการเทียบเรียงกลุ่มข้อมูล และให้ S(C1,C2) คือค่า Matching Score สำหรับ ลำดับข้อมูล C1 และ C2 ใน Scoring Matrix ดังนั้น SP Score สำหรับคอลัมน์ i (SP(i)) คือ

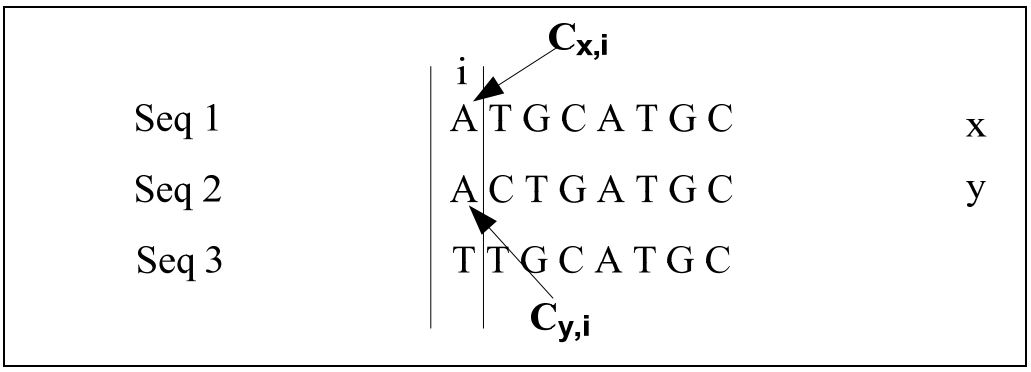
$$SP(i) = \sum_{x<y} S(c_{x,i}, c_{y,i}) \quad (4\text{-}1)$$

ผลรวมของ SP Score สำหรับการเทียบเรียงกลุ่มลำดับข้อมูลคำนวณได้จากการรวมค่า SP Score ของแต่ละคอลัมน์

ดังนั้น

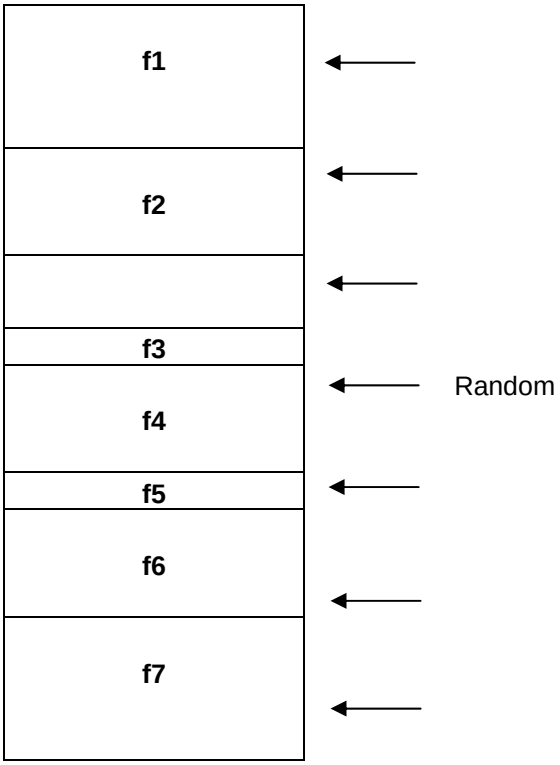
$$\text{SP score} = \sum_{1\le i\le n} SP(i) \quad (4\text{-}2)$$

โดยที่ n คือความยาวของสายอักษรในกลุ่มลำดับข้อมูล ที่รวม Gap เมื่อ SP Score ถูกใช้ใน Objective Function การเทียบเรียงกลุ่มที่ดีที่สุดหาได้จากกลุ่มลำดับข้อมูลที่ให้ค่า SP Score มากที่สุด



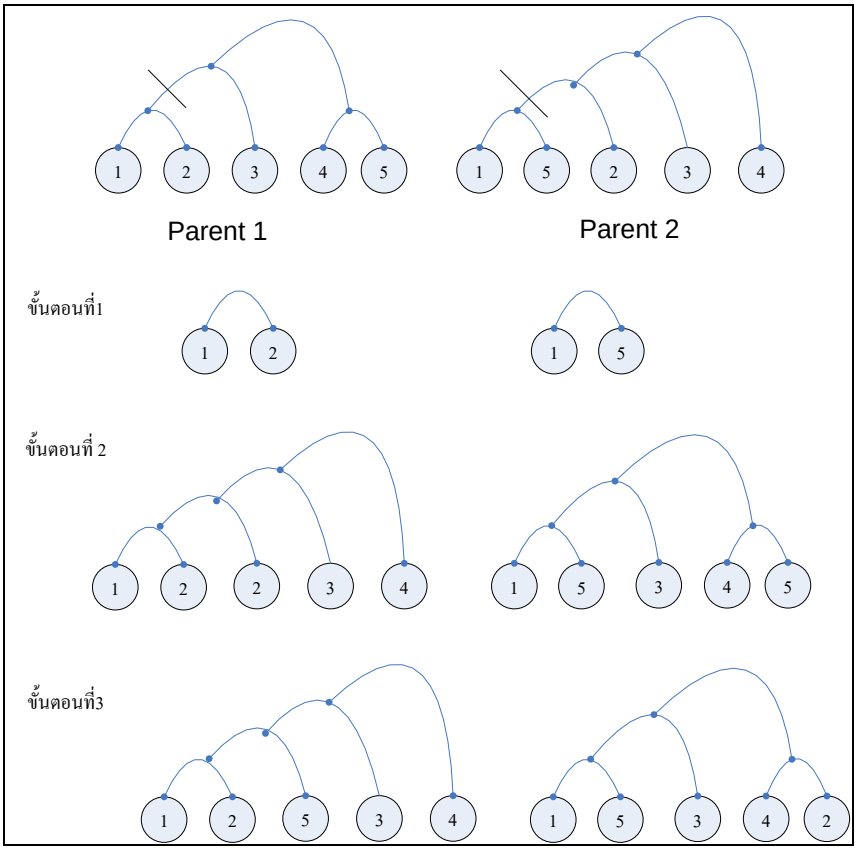
ภาพที่ 4-8 ตัวอย่างการคิด SP Score

4.2.3 การคัดเลือก (Selection) ในงานวิจัยนี้ใช้การ Selection แบบการเลือกสุ่มตัวอย่างแบบเฟ้นสุ่มสากล (Stochastic Universal Sampling Selection) จะการทำงานดังภาพ ที่ 4-9 โดยจะมีจำการนำค่า Fitness ของประชากรนำมาแปลงเป็นเส้น นำ Fitness มาต่อกันเป็นเส้น จะได้เส้นที่มีความยาวเท่ากับ ผลรวมของ Fitness หลังจากนั้นจะทำการ Random จุดตกบนเส้น Fitness ถ้าตกลงใน Fitness ตัวไหนก็จะเลือกประชากรตัวนั้นมา หลังจากนั้นจะทำการเลื่อนตำแหน่งจากจุดที่สุ่มได้เป็นระยะทางเท่ากับผลรวมของ Fitness หารด้วยจำนวนประชากร และพิจารณาว่าตำแหน่งที่เลื่อนไปอยู่ที่ Fitness ตัวไหน ทำการเลือกประชากรตัวนั้นมา ทำอย่างนี้ n ครั้ง เท่ากับจำนวนประชากร สุดท้ายตำแหน่งที่เลื่อนจะตกลงตรงที่เดิมที่สุ่มได้ หลังจากการทำ Selection จะได้ประชากรจำนวนเท่ากับ Pop Size โดยตัวที่ค่า Fitness ดีก็จะมีโอกาสที่จะถูกเลือกมามากกว่า 1 ครั้ง

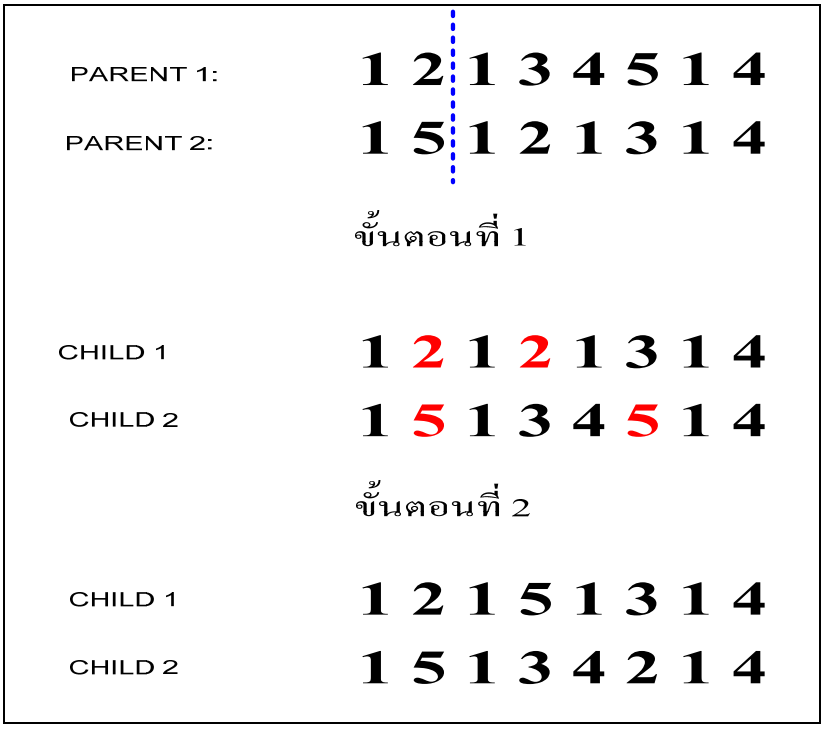


ภาพที่ 4-9 การทำงานของขั้นตอนการเลือกประชากร

4.2.4 การข้ามพันธุ์ (Crossover) ใช้ วิธีการ One-Point Crossover เทคนิคนี้เป็นการแลกเปลี่ยนยีนระหว่างสมาชิกของประชากรรุ่นพ่อแม่เพื่อให้เกิดการสร้างสมาชิกของประชากรรุ่นลูกนั้น จะเกิดขึ้น ณ ข้างใดข้างหนึ่งของตำแหน่งการครอสโอเวอร์ (Crossover Site) โดยขั้นแรกจะทำการสุ่มตำแหน่งที่ต้องการจะตัด หลังจากได้ตำแหน่งการตัดแล้วจะทำการแลกเปลี่ยนยีนของประชากรรุ่นพ่อรุ่นแม่ ดังภาพที่ 4-10 หลังจากทำการแลกเปลี่ยนยีนแล้วจะทำการเช็คเงื่อนไขของการแลกเปลี่ยนว่าเป็นไปได้หรือไม่ จากตัวอย่างนี้ที่ 4-9 จะเห็นได้ว่าจะ Sequence2 ของ ลูกตัวที่ 1 กับ Sequence 5 ลูก ตัวที่ 2 นั้นจะมีค่าซ้ำกัน จะทำการตัด Sequence ที่ซ้ำของลูกฝั่งขวาทิ้งและจะทำการนำ Sequence ที่ยังไม่ถูกเลือกมาใส่แทน หลังจากทำการ Crossover จะได้ประชากรรุ่นลูกมา 2 ตัวในงานวิทยานิพนธ์นี้จะทำการเลือกลูกที่ดีที่สุดมาเพียงตัวเดียวโดยจะดูจากค่า Fitnees

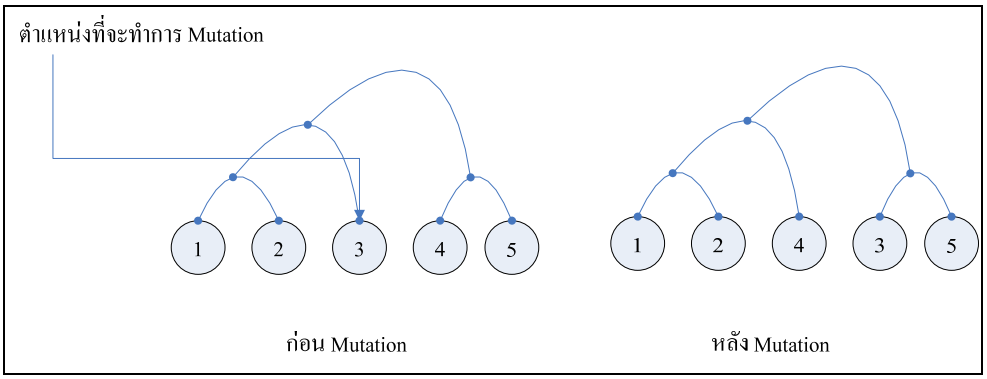


ภาพที่ 4-10 ตัวอย่างการทำ Crossover ในรูปแบบ Guide Tree



ภาพที่ 4-11 ตัวอย่างการ Crossover ในรูปแบบประชากรของขั้นตอนวิธีเชิงพันธุกรรม

4.2.5 การกลายพันธุ์ (Mutation) เป็นกระบวนการที่ใช้ในการสร้างประชากรใหม่จากประชากรที่มีอยู่เดิม ซึ่งจะส่งผลให้สมาชิกของประชากรรุ่นใหม่หรือรุ่นลูกที่เกิดขึ้นมีลักษณะที่ดีกว่าสมาชิกของประชากรรุ่นพ่อแม่ปัจจุบัน โดยค่าความน่าจะเป็นในการกลายพันธุ์ในงานวิจัยนี้ใช้ค่า 0.3 การกลายพันธุ์เป็นวิธีที่ทำให้เกิดการเปลี่ยนแปลงของยีนเพียงเล็กน้อย ในโครโมโซมของสมาชิกของประชากร การเปลี่ยนแปลงดังกล่าวทำให้เกิดการค้นหาคำตอบ ซึ่งอยู่ในตำแหน่งใกล้เคียงกับคำตอบซึ่งแทนโดยสมาชิกของประชากรตัวเดิมในพื้นที่การค้นหา โดยการเปลี่ยนสลับที่ใบกับใบทางขวา ดังภาพที่ 4-12



ภาพที่ 4-12 ขั้นตอนการทำ Mutation

4.2.6 เงื่อนไขในการออกจากขั้นตอนเชิงพันธุกรรม (Condition) ในงานวิจัยนี้ตั้งค่า Condition ไว้คือถ้า Fitness ที่มากที่สุดของ Generation มีค่าเหมือนเดิม 20 Generation หรือ ค่า Generation มีค่าเท่ากับ Max Generation ในที่นี้คือ 100

บทที่ 5
การประเมินประสิทธิภาพของการเทียบเรียง

การเทียบเรียงกลุ่มลำดับข้อมูลทางชีววิทยาโมเลกุล (Multiple Sequence Alignment) เป็นหนึ่งในเครื่องมือที่มีความสำคัญในงานทางด้านชีววิทยาโมเลกุล การเทียบเรียงกลุ่มลำดับข้อมูลนำไปใช้ประโยชน์ได้หลายทางเช่น เอาไปใช้ประโยชน์ในการทำนายโครงสร้างของโปรตีน (Structure Predition) การวิเคราะห์วิวัฒนาการของสิ่งมีชีวิต (Phylogenetic Analysis) การออกแบบ Primer ในปฏิกิริยาพีซีอาร์ และในกรณีของโปรตีนใช้หาว่าช่วงใดมีความสำคัญควรนำไปศึกษาหรือวิเคราะห์ต่อไป เป็นต้น การเทียบเรียงกลุ่มลำดับข้อมูลคือการเปรียบเทียบความเหมือนกันของสายลำดับตั้งแต่ 3 สายลำดับเป็นต้นไป

ในการวัดประสิทธิภาพของการเทียบเรียงนั้นมีวิธีวัดได้หลายวิธี ส่วนมากจะใช้ค่า Score เป็นตัววัดประสิทธิภาพ โดย Score นั้น มีอยู่หลาย Score เช่น SP Score, Entropy Score, Quality Score Colum Score ตัวที่ได้รับความนิยมมากที่สุดคือ SP Score โดยในวิทยานิพนธ์นี้ใช้ SP Score เป็นตัววัดประสิทธิภาพ

วิธีการหาค่า SP Score นั้นหาได้จากการรวมค่า Sp Score ของแต่ละคอลัมน์มารวมกันสำหรับแต่ละคอลัมน์ ในการเทียบเรียงกลุ่มข้อมูลถูกคำนวณโดยค่า Matching Score ระหว่างแต่ละคอลัมน์ ให้ $C_{x,I}$ คือค่าลำดับข้อมูลที่ตำแหน่งแถว x คอลัมน์ i ใน การเทียบเรียงกลุ่มข้อมูล และให้ S(C1,C2) คือค่า Matching Score สำหรับ ลำดับข้อมูล C1 และ C2 ใน Scoring Matrix ดังนั้น SP Score สำหรับคอลัมน์ i (SP(i)) คือ

$$SP(i) = \sum_{x<y} S\,(c_{x,i}, c_{y,i}) \qquad (5\text{-}1)$$

ผลรวม ของ SP Score สำหรับการเทียบเรียงกลุ่มลำดับข้อมูลคำนวณได้จากการรวมค่า SP Score ของแต่ละคอลัมน์

ดังนั้น

$$SP\ score = \sum_{1\le i\le n} SP(i) \qquad (5\text{-}2)$$

โดยที่ n คือความยาวของสายอักษรในกลุ่มลำดับข้อมูล ที่รวม Gap เมื่อ SP Score ถูกใช้ใน Objective Function การเทียบเรียงกลุ่มที่ดีที่สุดหาได้จาก กลุ่มลำดับข้อมูลที่ให้ค่า SP Score มากที่สุด ในงานวิทยานิพนธ์นี้ได้ทำการจำลองระบบตามที่ได้กล่าวมาแล้วในบทที่แล้ว โดยมีขั้นตอนหลักๆ ดังภาพที่ 5-1 ในวิทยานิพนธ์นี้ได้ทำการ จำลองระบบด้วยภาษา Python สาเหตุที่เลือกภาษา Python มาใช้เนื่องจากเป็นภาษาที่เขียนง่าย มีเครื่องมือที่ใช้รองรับกับงานทางด้าน ชีววิทยาโมเลกุลจำนวนมาก เช่น BioPython เป็นต้น มีเครื่องมือที่ใช้รองรับกันงานทางด้านการคำนวณแบบขนานสำหรับงานวิจัยต่อไปในอนาคตได้ เช่น MPI python โมดูลที่ใช้รวมกับ ภาษา C ได้ซึ่งเป็นภาษาที่ใช้เขียนโปรแกรม ClustalW และภาษา Python ยังสามารถทำงานได้ทุกระบบปฏิบัติการ ในงานวิจัยนี้ทดสอบบนระบบปฏิบัติการ Linux โดยทดสอบบนเครื่อง IBM sattelise 2 cpu โดยในงานวิจัยนี้ได้ทดสอบกับชุดข้อมูล 2 ชุด ประกอบด้วย ชุดข้อมูลRef1 แล ชุดข้อมูล Ref2 โดยแต่ละชุดจะมี Multiple Sequence File จำนวน 10 File เป็นโปรตีนFile อยู่ในรูป Fasta Format โดยในชุด Ref1 จะเป็นชุดที่จำนวนสายลำดับจำนวนไม่มาก ประมาณ 20 สายลำดับ ส่วนในข้อมูล Ref2 จะเป็นชุดข้อมูลที่มีจำนวนสายลำดับประมาณ 100 สายลำดับ

ในวิทยานิพนธ์นี้จะทำการวัดประสิทธิภาพของการเทียบเรียงโดยดูจาก SP Score ที่ได้จากการทำ Progressive Alignment ตามลำดับการเทียบเรียงที่ได้จากขั้นตอนวิธีทางพันธุกรรม ด้วยโปรแกรม ClustalW เปรียบเทียบกับ SP Score ที่ได้ที่ได้จากการทำ Progressive Alignment ตามลำดับการเทียบเรียงที่ได้จากโปรแกรม ClustalW และวิธี Minimum Spanning tree โดยใช้โปรแกรม ClustalW เป็นตัวทำ Progressive Alignment

ผลการทดลองได้ดังตารางที่ 5-1 และ5-2 โดยที่ N คือจำนวนสายลำดับในกลุ่มลำดับข้อมูล Len คือความยาวที่มากที่สุดในกลุ่มลำดับข้อมูล ค่า Score ที่ได้จะเป็นค่า SP Score ต่อ 1 ลำดับข้อมูลคำนวณได้จากสมการที่ 5-3 GA โดยที่ N คือจำนวนสายลำดับข้อมูล Len คือความยาวสายลำดับข้อมูลหลังทำการเทียบเรียงแล้ว Score คือค่าคะแนนที่ได้จากการใช้ลำดับการเทียบเรียงที่ได้จากขั้นวิธีเชิงพันธุกรรมที่ทำการทดขั้นตอนวิธีเชิงพันธุกรรมจำนวน 10 ครั้งมาทำการเฉลี่ยกัน MST Score คือค่าคะแนนที่ได้จากการเทียบเรียงด้วยลำดับการเทียบเรียงที่ได้จากวิธี Minimum Spanning Tree ClustalW Score คือค่าคะแนนที่ได้จากการเทียบเรียงจากโปรแกรม ClustalW

$$\text{Score} = \text{SP Score} / ((N(N-1)/2) \times \text{Len}) \qquad (5\text{-}3)$$

โปรแกรม ClustalW ที่นำมาใช้เป็นเวอร์ชัน 1.82 ทำงานบนระบบปฏิบัติการ Linux ค่า Parameter ที่ใช้ในโปรแกรม เป็นค่า Default ของโปรแกรม

จากตารางที่ 5-1 และ 5-2 เมื่อนำค่า Score ที่ได้ไปเปรียบเทียบกับ ClustalW Score จะได้ดังตารางที่ 5-3 และ 5-4 นำไปวาดกราฟได้ดังภาพที่ 5-1 และ 5-2

จากผลการทดลองข้อมูลใน Ref1 ข้อมูล R1_2, R1_6, R1_7, R1_8 และ R1_10 ขั้นตอนวิธีเชิงพันธุกรรมสามารถหา Guided Tree ที่ดีกว่า MST และ ClustalW ส่วนในข้อมูล R1_1, R1_3, R1_4, R1_5 ได้ Guided Tree เหมือนกับ MST และ ClustalW นั้นอาจจะเป็นเพราะ Guide Tree ที่ได้อาจจะเป็น Guide Tree ที่ดีที่สุดอยู่แล้ว และถ้าต้องการรู้ว่าค่า Score ที่ได้มากน้อยเพียงไรก็ทำได้โดยการนำค่า Score ที่ได้ไปคูณด้วย ((N(N-1)/2) × Len) ก็จะได้เป็นค่า Score

ตารางที่ 5-1 ผลการทดลองชุดข้อมูล Ref1

ข้อมูล	N	Len	GA	MST	Clustalw
ทดลอง			score	score	score
Ref1_1	21	469	6.0628	6.0628	6.0628
Ref1_2	22	469	2.3008	4.5996	4.596
Ref1_3	20	469	5.736	5.736	5.736
Ref1_4	22	469	5.122	5.122	5.122
Ref1_5	24	469	6.022	6.022	6.022
Ref1_6	19	1979	1.6972	1.6943	1.6698
Ref1_7	18	1634	1.3484	1.3003	1.315
Ref1_8	21	1534	4.3036	4.3028	4.3014
Ref1_9	19	1511	4.0292	4.0276	4.0276
Ref1_10	25	2055	0.4938	0.4904	0.4774

ตารางที่ 5-2 ผลการทดลองชุดข้อมูล Ref2

ข้อมูล	N	Len	GA	MST	Clustalw
ทดลอง			score	score	score
Ref2_1	127	471	2.2954	2.296	2.2692
Ref2_2	100	469	4.465	4.4608	4.444
Ref2_3	116	469	3.5138	3.5138	3.5094
Ref2_4	94	469	5.3146	5.3146	5.3136
Ref2_5	87	459	6.2242	6.2242	6.2242

ตารางที่ 5-2 (ต่อ)

ข้อมูล	N	Len	GA	MST	Clustalw
ทดลอง			score	score	score
Ref2_6	90	469	5.51	5.4104	5.41
Ref2_7	102	469	4.38	4.38	4.38
Ref2_8	84	469	4.462	4.4038	4.399
Ref2_9	98	469	5.5562	5.5542	5.5542
Ref2_10	105	469	5.0436	5.0352	5.0372

ตารางที่ 5-3 ผลการทดลองชุดข้อมูล Ref1 เทียบกับ ClustalW Score

ข้อมูล	N	Len	GA	MST	Clustalw
ทดลอง			Score(10^{-3})	Score(10^{-3})	Score(10^{-3})
Ref1_1	21	469	0	0	0
Ref1_2	22	469	5.6177	3.5848	0
Ref1_3	20	469	0	0	0
Ref1_4	22	469	0	0	0
Ref1_5	24	469	0	0	0
Ref1_6	19	1979	27.4018	24.5027	0
Ref1_7	18	1634	33.3792	-14.7247	0
Ref1_8	21	1534	2.2421	1.4273	0
Ref1_9	19	1511	1.6011	0	0
Ref1_10	25	2055	16.423	13.003	0

ตารางที่ 5-4 ผลการทดลองชุดข้อมูล Ref2 เทียบกับ ClustalW Score

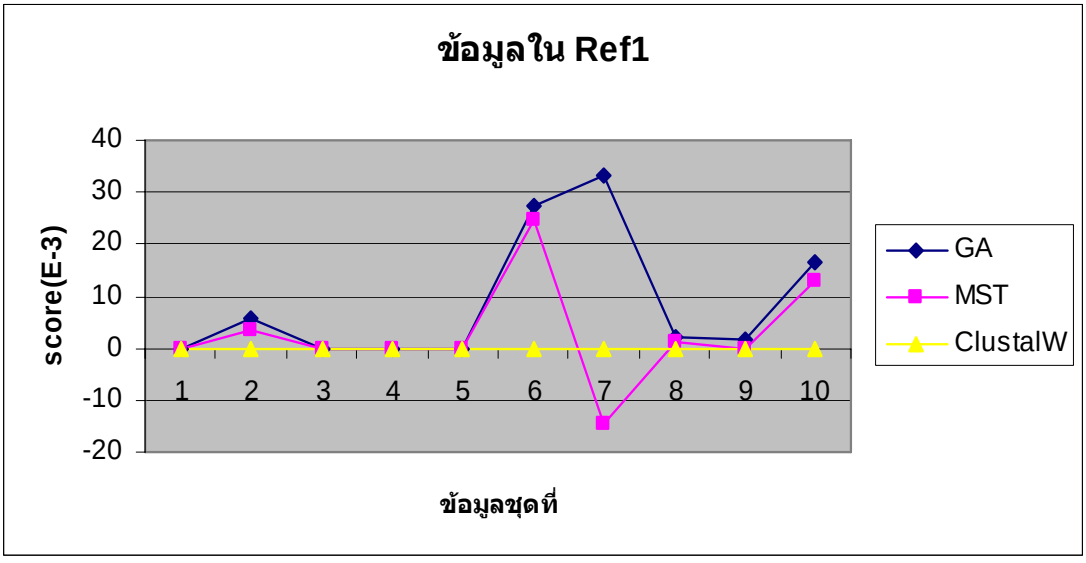
ข้อมูล	N	Len	GA	MST	Clustalw
ทดลอง			Score(10-3)	Score(10-3)	Score(10-3)
Ref2_1	127	471	26.2621	16.5355	0
Ref2_2	100	469	21.1246	16.8009	0
Ref2_3	116	469	4.4658	4.3711	0
Ref2_4	94	469	0.8771	0.0384	0

ตารางที่ 5-4 (ต่อ)

ข้อมูล	N	Len	GA	MST	Clustalw
ทดลอง			Score(10-3)	Score(10-3)	Score(10-3)
Ref2_5	87	459	0	0	0
Ref2_6	90	469	0.9931	0.3664	0
Ref2_7	102	469	0	0	0
Ref2_8	84	469	7.09	4.6689	0
Ref2_9	98	469	1.8799	-0.0638	0
Ref2_10	105	469	5.3554	-1.9188	0

จากผลการทดลองข้อมูลใน Ref2 ข้อมูล R2_1, R2_ 2, R2_3, R2_4, R2_6, R2_8, R2_9 และ R2_10 ขั้นตอนวิธีเชิงพันธุกรรมสามารถหา Guided Tree ที่ดีกว่า MST และ ClustalW ส่วนในข้อมูล R2_5 และ R2_7 ได้ Guided Tree เหมือนกับ MST และ ClustalW นั้นอาจจะเป็นเพราะ Guide Tree ที่ได้อาจจะเป็น Guide Tree ที่ดีที่สุดอยู่แล้ว

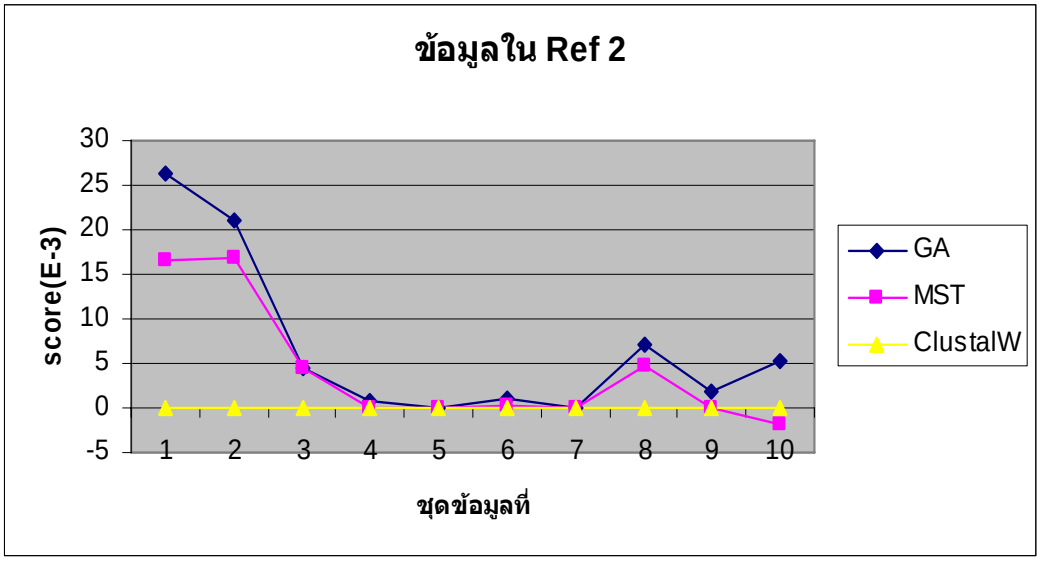
จากผลการทดลองของชุดข้อมูล Ref1 และ Ref2 ดังในตารางที่ 5-1 และ 5-2 นำไปหาค่าเฉลี่ยได้ดังในตารางที่ 5-5 นำไปวาดกราฟได้ดังภาพที่ 5-3



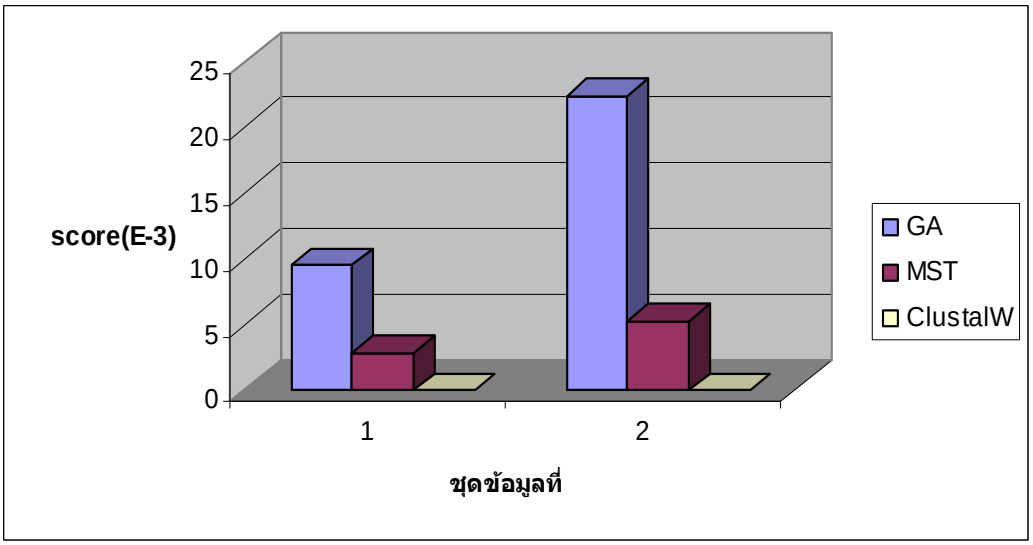
ภาพที่ 5-1 กราฟแสดงการเปรียบเทียบกับระหว่าง Score ของลำดับเทียบเรียงอื่นๆ กับ ClustalW Score ของชุดข้อมูล Ref1

ตารางที่ 5-5 ค่า Score เฉลี่ย ที่ได้จากชุดข้อมูล Ref1 และ Ref2

ข้อมูลชุดที่	GA	MST	Clustalw
	Score(10^{-3})	Score(10^{-3})	Score(10^{-3})
Ref 1	9.47	2.78	0
Ref 2	22.4	5.22	0



ภาพที่ 5-2 กราฟแสดงการเปรียบเทียบกับระหว่าง Score ของลำดับเทียบเรียงอื่นๆ กับ ClustalW Score ของชุดข้อมูล Ref2



ภาพที่ 5-3 กราฟแสดงการเปรียบเทียบค่า Score เฉลี่ย ของลำดับเทียบเรียงอื่นๆ กับค่าเฉลี่ยของ ClustalW Score ในชุดข้อมูล Ref1 และ Ref2

ในงานนี้ใช้ขั้นตอนเชิงพันธุกรรมในการปรับปรุงประสิทธิภาพลำดับการเทียบเรียงที่ได้จากวิธีการต่างๆ โดยในงานนี้ได้ทดลองปรับปรุงประสิทธิภาพลำดับการเทียบเรียงที่ได้จากโปรแกรม ClustalW และจากวิธี Minimum Spanning Tree โดยขั้นตอนวิธีเชิงพันธุกรรมสามารถหาลำดับการเทียบเรียงที่ให้ค่า SP-Score ได้มากกว่าหรือเทียบเท่าตัวที่ดีที่สุดของลำดับเทียบเรียงที่นำมาปรับปรุงได้

บทที่ 6
สรุปผลการวิจัย และข้อเสนอแนะ

คุณภาพของการเทียบเรียงแบบ Progressive ขึ้นอยู่กับลำดับการเทียบเรียง เนื่องจากลำดับการเทียบเรียงที่ต่างกันจะให้ผลที่ไม่เหมือนกัน ปัญหาของการเทียบเรียงแบบ Progressive คือ การหาลำดับการเทียบเรียงที่ดีที่สุดการหาลำดับการเทียบเรียงในปัจจุบันมีอยู่หลายวิธี โดยส่วนใหญ่จะมีวิธีการทำงานเหมือนกันคือขั้นตอนที่ 1 หา Distance Matrix จาก การทำ Pairwise Alignment ทุกคู่ลำดับจำนวน n(n-1)/2 คู่ แล้วเปลี่ยนค่าความคล้ายกัน เป็นDistance เก็บอยู่ในรูป Matrix ขนาด n×n โดย n คือจำนวน Sequence ขั้นตอนที่ 2 นำ Distance Matrix ที่ได้มาทำการหาลำดับการเทียบเรียงด้วยวิธีการต่างๆ ขั้นตอนที่สาม ทำการเทียบเรียงตามลำดับการเทียบเรียงที่ได้ วิธีการส่วนใหญ่ ขั้นตอนที่ 1 และ ขั้นตอนที่ 3 จะเหมือนกัน จะต่างกันที่ขั้นตอนที่ 2 แต่เวลาที่ใช้ในการคำนวณส่วนมากจะอยู่ในขั้นตอนที่ 1 ส่วน ขั้นตอนที่ 2 และ 3 จะใช้เวลาน้อยมากเมื่อเทียบกับขั้นตอนที่ 1 วิธีการหาลำดับการเทียบเรียงด้วยวิธีการต่างๆ ยังบอกไม่ได้ว่าวิธีไหนจะให้ Alignment ที่ดีกว่าวิธีไหนสำหรับการเทียบเรียงกลุ่มใดๆ

ในวิทยานิพนธ์นี้เสนอวิธีการหาลำดับการเทียบเรียงโดยใช้ขั้นตอนวิธีเชิงพันธุกรรม ซึ่งใช้หลักการการอยู่รอดของผู้แข็งแรงที่สุด มาทำการปรับปรุงประสิทธิภาพของลำดับการเทียบเรียงที่สร้างขึ้นจากวิธีการต่างๆที่มีอยู่ในปัจจุบันให้มีประสิทธิภาพดีขึ้น

โดยได้ทดลองใช้ขั้นตอนวิธีเชิงพันธุกรรม มาทำการปรับปรุงลำดับการเทียบเรียงที่ได้จากโปรแกรม ClustalW และลำดับการเทียบเรียงที่ได้จากวิธี Minimum Spanning Tree โดยใช้ลำดับการเทียบเรียงเป็นประชากรของขั้นตอนวิธีเชิงพันธุกรรม ใช้ SP Score เป็นตัววัดประสิทธิภาพของการเทียบเรียง โดยหวังว่าการที่มีประชากรรุ่นพ่อแม่ ที่ดีจะทำให้ได้ประชากรรุ่นลูกที่ดี ทำการเปรียบเทียบลำดับการเทียบเรียงที่ได้จากขั้นตอนวิธีเชิงพันธุกรรม มาเปรียบเทียบกับลำดับการเทียบเรียงที่ได้จากโปรแกรม ClustalW โดยใช้ค่า SP Score เป็นตัวเปรียบเทียบ ทำการทดลองกับข้อมูล 2 ชุดข้อมูล โดยชุดข้อมูลที่ 1 จะเป็นชุดข้อมูลที่มีจำนวนสายลำดับข้อมูลไม่มากประมาณ 20 สายลำดับข้อมูล กับชุดข้อมูลที่ 2 ที่จำนวนสายลำดับข้อมูล ประมาณ 100 สายลำดับข้อมูล ผลการทดลองปรากฏว่าลำดับการเทียบเรียงที่ได้จากขั้นตอนวิธีเชิงพันธุกรรมสามารถให้ค่า SP Score ได้มากกว่า SP Score ที่ได้จากการเทียบเรียงด้วยโปรแกรม ClustalW ทั้ง 2 ชุดข้อมูล

เนื่องจากในงานวิจัยนี้ใช้โปรแกรม ClustalW เป็นตัวเทียบเรียง ซึ่งโปรแกรม ClustalW เป็นโปรแกรมที่นักวิจัยทางด้านชีววิทยาโมเลกุล นำไปใช้ประโยชน์อื่นได้หลายทาง และลำดับการเทียบเรียงที่ได้จากงานวิจัยนี้อาจจะนำไปใช้ในงานทางด้านชีววิทยาโมเลกุลอื่นๆ ที่ต้องใช้ลำดับการเทียบเรียงได้โดย ไม่เจาะจงเฉพาะการทำ Multiple Sequence Alingment เพียงเท่านั้น

งานวิจัยต่อไป ในงานวิจัยนี้สามารถเพิ่มประสิทธิภาพให้ดีขึ้นได้ ในเรื่องของเวลาคำนวณ เนื่องจากขั้นตอนวิธีเชิงพันธุกรรม เป็นการหาคำตอบแบบคู่ขนาน (Parallel Search) ดังนั้นเราจึงสามารถปรับปรุงประสิทธิภาพในการคำนวณของขั้นตอนวิธีเชิงพันธุกรรม โดยใช้การคำนวณแบบขนาน (Parallel Computing) มาใช้ได้ ในเรื่องของคุณภาพของลำดับการเทียบเรียง ในงานวิจัยนี้สามารถเพิ่มวิธีการในการหา ลำดับการเทียบเรียงในส่วนของการสร้างลำดับการเทียบเรียงด้วยวิธีการต่างๆ และสุดท้ายในงานวิจัยนี้สามารถเปลี่ยนตัววัดคุณภาพของการเทียบเรียง ไปเป็นตัววัดคุณภาพที่สนใจได้ ไม่เฉพาะเจาะจงต้องใช้ SP Score เท่านั้น

เอกสารอ้างอิง

1. Weiwei Zhong. **Using Traveling Salesman Problem Algorithms to Determine Multiple Sequence Alignment Orders**. Ph.D. Thesis, Faculty of Science, University of Georgia, 2002.
2. Needlelman ,S. B. and Wunsch C. D. "A general method applicable to the search for similarities in the amino acid sequence of two proteins." **J. Mol. Bio**. 48 (1970) : 443-453.
3. Smith, T. F. and Waterman, M. S. "Identification of common molecular subsequences." **Journal or Molecular Biology**. 147 (1981) :195-197.
4. Feng D. and Doolittle R. "Progressive sequence alignment as a prerequisite to correct phylogenetic trees." **J. Mol .Evol**. 25 (1987) : 351-360.
5. Thompson., J. D. and Gibsion T. J. "CLUSTALW: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties, and matrix choice." **Nucleic Acids Res**. 22 (1994) : 4673-4680.
6. Thompson J. D., Gibson T. J. "The clustal_x windows interface: flexible strategies for multiple sequence alignment aided by quality analysis tools." **Nucleic Acids Research**. 25 (1997) : 4876-4882.
7. Li K. B. "Clustalw-mpi: Clustalw analysis using distributed and parallel computing, Bioinformatics." **Bioinformatics**. 19 (2003) : 1585-1586.
8. Mei-Jie Zhu. "Multiple sequence alignment using Minimum spanning tree." Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005.
9. Notredame and Higgins. "SAGA:sequence alignment by genetic algorithm." **Nucleic Acids Research**. 24 (1996) : 1515-1524.
10. Anbarasu, L. A., Narayanasamy, P. and Sundararajan, V. "Multiple molecular sequence alignment by island parallel genetic algorithm." **Current Science**. 78 (2000) : 858-863.
11. Zhang, C. and Andrew, K. C. "A genetic algorithm for multiple molecular sequence alignment." **Cabios**. 13 (1997) : 565-581.

12 Zhang, C. and Andrew, K. C. "Toward efficient multiple molecular sequence alignment: A system of genetic algorithm and dynamic programming." **IEEE Transactions on Systems**. 27 (1997) : 918-932.

13. Li-Fang LIU., Hong-wei HUO and Bao-shu WANG. "Aligning multiple sequences by genetic algorithm." **IEEE**. (2004) : 994-998.

14. Yixin Chen and Yi Pan. **Partitioned optimization algorithms for multiple sequence alignment**. Washington : Department of Computer Science Washington University, 2006.

15. Dayhoff, M. O., Schwartz , R. and Orcutt, B. C. "A model of evolutionary change in proteins." **Atlas of Protein Sequence and Structure National Biomedical Research Foundation**. 5 (1978) : 345-352.

16. Nicholas, H. B., Ropelewski, A.J. and Deerfield, D.W. "Strategies for multiple sequence alignment." **Biotechniques**. 32 (2002) : 572-578.

17. Holland J. **Adaptation in Natural and Artificial system**. AM Arbour MI : University of Michigan Press, 1975.

18. Saitou, N. and Nei, M. "The neighbor-joining method: a new method for reconstructing phylogenetic trees." **Mol. Biol. Evol**. 4 (1987) : 406-425.

19. Kenneth and Rosen, H. **Discreate mathematics and its application**. New York : MC Graw-Hill, 1999.

ประวัติผู้วิจัย

ชื่อ : นายกล้า วณิชชาโสภณ

ชื่อวิทยานิพนธ์ : การใช้ขั้นตอนวิธีเชิงพันธุกรรมในการสร้างลำดับการเทียบเรียงกลุ่มข้อมูลชีวภาพ

สาขาวิชา : วิศวกรรมไฟฟ้า

ประวัติ

ประวัติส่วนตัว สถานที่ติดต่อ 352/28 ซอยเขมาเนรมิตร ถนนประชาราษฎร์สาย 1 แขวงบางซื่อ เขตบางซื่อ กรุงเทพมหานคร 10800

ประวัติการศึกษา สำเร็จการศึกษาปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ ปีการศึกษา 2545