บทที่ 3

งานวิจัยที่เกี่ยวข้อง

บทนี้กล่าวถึงงานวิจัยที่เกี่ยวข้องกับขั้นตอนวิธีเชิงวิวัฒน์แบบหลายวัตถุประสงค์ การจัดเรียงลำดับเบสหลายลำดับ ตัวอย่างขั้นตอนวิธีที่ใช้แก้ปัญหาการจัดเรียงลำดับเบสหลายลำดับ และรายระเอียดของฐานข้อมูล BAliBASE

3.1 งานวิจัยเกี่ยวกับขั้นตอนวิธีเชิงวิวัฒน์แบบหลายวัตถุประสงค์

งานวิจัยเกี่ยวกับขั้นตอนวิธีเชิงวิวัฒน์แบบหลายวัตถุประสงค์สามารถแยกย่อยได้ 2 ส่วน ดังนี้ คือ แบบไม่ใช้พาเรโต (Non-Pareto techniques) และแบบพาเรโต (Pareto techniques)

3.1.1 แบบไม่ใช้พาเรโต

ขั้นตอนวิธีเชิงวิวัฒน์แบบหลายวัตถุประสงค์แบบไม่ใช้พาเรโตส่วนมากจะไม่หาผลเฉลยที่อยู่ในพาเรโตฟรอนต์แต่จะหาค่าจากการใช้ผลรวม หรือการกำหนดขอบเขตของคำตอบเช่น

1 วิธีผลรวมแบบใช้น้ำหนัก (Weighted sum approach) พัฒนาขึ้นโดย Rosenberg ในปี 1967 เป็นวิธีที่ทำการรวมฟังก์ชันวัตถุประสงค์ต่างๆคูณกับน้ำหนักของวัตถุประสงค์นั้นๆเข้าด้วยกันกลายเป็นฟังก์ชันวัตถุประสงค์เดียวเพื่อง่ายต่อการคัดเลือกคำตอบในขั้นตอนวิธีเชิงพันธุกรรม

2 วิธีจำกัดขอบเขตเอปไซรอน (ε-constraint method) [20] พัฒนาขึ้นโดย Szidarovsky และ Duckstein ในปี1982 ใช้วิธีในการเก็บฟังก์ชันวัตถุประสงค์ที่น้อยที่สุด และเก็บฟังก์ชันวัตถุประสงค์ตัวอื่นๆที่มีค่าไม่เกินเอปไซรอนที่ยอมรับได้ เพราะฉะนั้นฟังก์ชันวัตถุประสงค์ที่ได้ในวิธีนี้จะมีค่าเดียว และมีความสัมพันธ์กับฟังก์ชันวัตถุประสงค์อื่นโดยการกำหนดขอบเขตการคัดเลือกผลเฉลย

3 VEGA [21] พัฒนาขึ้นโดย Schaffer ในปี1985 มีขั้นตอนในการปรับปรุงผลเฉลยโดยตัวปฏิบัติการทำการปรับปรุงคำตอบจากการคัดเลือกผลเฉลยจากกลุ่มประชากรย่อยที่ถูกแบ่งออกตามจำนวนฟังก์ชันวัตถุประสงค์ที่นำมาใช้แก้ปัญหา และแต่ละรุ่นจะมีการสลับผลเฉลยระหว่างกลุ่มประชากรย่อยเพื่อช่วยในการปรับปรุงผลเฉลย

3.1.2 แบบใช้พาเรโต

ขั้นตอนวิธีเชิงวิวัฒน์แบบหลายวัตถุประสงค์แบบใช้พาเรโตจะพยายามหาคำตอบจากผลเฉลยในพาเรโตฟรอนต์ โดยแต่ละวิธีจะมีวิธีการคิดแตกต่างกันออกไปเช่น

1 MOGA [22] พัฒนาขึ้นโดย Fonseca และ Fleming ในปี1993 ได้เสนอการแบ่งชั้น (rank) ของผลเฉลย โดยแบ่งชั้นตามความเด่น (Dominate) ในกลุ่มประชากร โดยผลเฉลยที่เด่นจะมีค่าชั้นต่ำ และจะถูกทำโทษให้มีค่าชั้นมากตามความเด่นที่น้อยลงของผลเฉลย จากนั้นทำการคัดเลือกเพื่อปรับปรุงผลเฉลย

2 NSGA [23] พัฒนาขึ้นโดย Srinivas และ Deb ในปี1993 ใช้พื้นฐานการแบ่งชั้น (Layers) ในการคัดแยกผลเฉลย โดยผลเฉลยเด่นทั้งหมดจะถูกแยกออกจากกลุ่มประชากร จากนั้นทำการคัดเลือกผลเฉลยเพื่อปรับปรุงจากกลุ่มผลเฉลยเด่น

3 NPGA [24] พัฒนาขึ้นโดย Horn และ Nafpliotis ในปี1993 เสนอการคัดเลือกแบบแข่งขัน (Tournament selection) บนพื้นฐานพาเรโต (Pareto dominance) โดยค่าความแข็งแรงได้จากการปันค่าความแข็งแรง (Fitness sharing)

3.2 งานวิจัยเกี่ยวกับการจัดเรียงลำดับเบสหลายลำดับ

งานวิจัยทางด้านการจัดเรียงลำดับเบสหลายลำดับมีวิธีการแก้ปัญหาได้หลายวิธี ในที่นี้ทำการนำเสนอการแก้ปัญหาโดยการใช้ขั้นตอนวิธีเชิงวิวัฒน์ และขั้นตอนวิธีต่างๆเช่น ขั้นตอนวิธีเชิงก้าวหน้า(Progressive algorithm) เป็นต้น

3.2.1 การใช้ขั้นตอนวิธีเชิงวิวัฒน์ในการแก้ปัญหา

การใช้ขั้นตอนวิธีเชิงวิวัฒน์แก้ปัญหาการจัดเรียงลำดับเบสหลายลำดับนั้นเป็นที่ยอมรับและมีมาเป็นเวลานาน โดยยกตัวอย่างงานวิจัยที่ผ่านมาเช่น

1 MSA-EA1999 [7] พัฒนาขึ้นโดย Chellapilla และ Fogel ในปี1999 ใช้ขั้นตอนวิธีเชิงพันธุกรรม มีตัวปฏิบัติการ 4 แบบ และใช้สมการผลรวมของจำนวนแถวที่เป็นตัวอักษรเดียวกันต่อจำนวนแถวทั้งหมด และทำโทษด้วยจำนวนแก็ปทั้งหมดในแถวเป็นฟังก์ชันวัตถุประสงค์

2 MSA-EA2002 [8] พัฒนาขึ้นโดย Thomsen, Fogel และ Krink ในปี2002 ใช้ขั้นตอนวิธีเชิงพันธุกรรม มีการสร้างผลเฉลยตั้งต้นจากคำตอบของโปรแกรม Clustal V [4] และจากการสุ่ม มีตัวปฏิบัติการ 5 แบบ และใช้สมการผลรวมคู่เบส (Sum-of-pairs) เป็นฟังก์ชันวัตถุประสงค์

3 MSA-EA2003 [9] พัฒนาขึ้นโดย Thomsen, Fogel และ Krink ในปี2003 ใช้ขั้นตอนวิธีเชิงพันธุกรรม มีการสร้างผลเฉลยตั้งต้นจากคำตอบของโปรแกรม Clustal W มีตัวปฏิบัติการ 5 แบบ และใช้สมการผลรวมคู่เบส (Sum-of-pairs) เป็นฟังก์ชันวัตถุประสงค์

4 SAGA [10] พัฒนาขึ้นโดย Notredame และ Higgins ในปี1996 ใช้ขั้นตอนวิธีเชิงพันธุกรรมในการจัดเรียง โดยใช้สมการผลรวมคู่เบสแบบใช้น้ำหนัก (Weight sum-of-pairs) เป็นฟังก์ชันวัตถุประสงค์ และมีตัวปฏิบัติการ 22 แบบ โปรแกรมสามารถจัดเรียงได้ดีแต่ต้องใช้เวลามาก

5 การจัดเรียงลำดับเบสหลายลำดับโดยขั้นตอนวิธีเชิงพันธุกรรมแบบขนาน [11] พัฒนาขึ้นโดย Anbarasu, Narayanasamy และ Sundararajan ในปี2000 ใช้ขั้นตอนวิธีเชิงพันธุกรรมแบบขนาน สร้างผลเฉลยตั้งต้นแบบสุ่ม มีตัวปฏิบัติการ 4 แบบ และใช้สมการผลรวมคู่เบสเป็นฟังก์ชันวัตถุประสงค์

6 การใช้ระบบขั้นตอนวิธีเชิงพันธุกรรมและกำหนดการพลวัต [12] พัฒนาขึ้นโดย Zhang และ Wong ในปี1997 ใช้ขั้นตอนวิธีเชิงพันธุกรรมในการสร้างการจัดเรียงก่อน (Prealignment) จากนั้นใช้กำหนดการพลวัตจัดเรียงลำดับเบสทั้งหมดอีกครั้ง

7 การใช้ระบบขั้นตอนวิธีเชิงพันธุกรรมและวิธีสังเคราะห์ลำดับ [13] พัฒนาขึ้นโดย Zhang และ Wong ในปี1998 ใช้ขั้นตอนวิธีเชิงพันธุกรรมในการสร้างการจัดเรียงก่อน จากนั้นใช้วิธีสังเคราะห์ลำดับ (Sequence synthesis method) จัดเรียงลำดับเบสทั้งหมดอีกครั้ง

8 ขั้นตอนวิธีเชิงพันธุกรรม และปัญหาการจัดลำดับเบสหลายลำดับในชีววิทยา [14] พัฒนาขึ้นโดย Karadimitriou และ Kraft ในปี1996 ใช้ขั้นตอนวิธีเชิงพันธุกรรมในการจัดเรียง โดยมีการจัดเรียง 2 แบบคือ การจัดเรียงแบบไม่มีแก็ป (MSA without gaps) และการจัดเรียงแบบมีแก็ป (MSA with gaps) มีการใช้ค่าแถวที่มีอักษรเหมือนกันเป็นฟังก์ชันวัตถุประสงค์

3.2.2 การใช้ขั้นตอนวิธีต่าง ๆ

การใช้ขั้นตอนวิธีต่างๆในการแก้ปัญหานั้นมีเป็นจำนวนมาก แต่ส่วนใหญ่แล้วขั้นตอนวิธีเชิงก้าวหน้า และการจัดเรียงองค์รวมเป็นที่นิยมมาก ยกตัวอย่างเช่น

1 Clustal W [5] พัฒนาโดย Thompson ในปี1994 ใช้ขั้นตอนวิธีเชิงก้าวหน้า (Progressive algorithm) โดยขั้นแรกทำการคำนวณทุกคู่ลำดับหาค่าเพื่อหาตารางระยะทาง (Distance matrix) และนำไปสร้างต้นไม้วิวัฒนาการ (Phylogenetic tree) ด้วยวิธี Neighbor-joining tree สุดท้ายทำการจัดเรียงคู่ลำดับตามรูปแบบที่ได้จากต้นไม้วิวัฒนาการ

2 DIALIGN [25] พัฒนาขึ้นโดย Morgenstren ในปี1999 ใช้วิธีแบ่งช่วงในการจัดเรียงออกเป็นส่วนๆ (Segment-segment alignment) การจัดลำดับวิธีนี้ใช้ได้ดีในลำดับที่มีความยาวและโครงสร้างใกล้เคียงกัน

3 T-Coffee [26] พัฒนาขึ้นโดย Notredame ในปี2000 ใช้ขั้นตอนวิธีเชิงก้าวหน้า โดยนำการใช้ การจัดเรียงองค์รวม (Global alignment) และ การจัดเรียงเฉพาะที่ (Local alignment) มากหาการสร้างต้นไม้วิวัฒนาการ แล้วทำการจัดเรียงคู่ลำดับตามต้นไม้วิวัฒนาการที่ได้

4 MAFFT [27] พัฒนาขึ้นโดย Kazutaka ในปี2002 ใช้ขั้นตอนวิธีเชิงก้าวหน้าโดยใช้วิธีสร้างต้นไม้วิวัฒนาการแบบ Neighbor-joining tree และใช้ผลการแปลงฟูเรียร์แบบเร็ว (Fast fourier transform) ช่วยในการตรวจหาบริเวณที่ลำดับมีความใกล้เคียงกัน

3.3 ตัวอย่างขั้นตอนวิธีในการแก้ไขปัญหาการจัดเรียงลำดับเบสหลายลำดับ

ในส่วนนี้จะแนะนำเครื่องมือและขั้นตอนวิธีที่ได้เคยมีการทำวิจัยมาแล้วในอดีต ซึ่งได้รับความนิยมและเป็นที่ยอมรับในปัจจุบันเช่น การจัดเรียงองค์รวม (Global alignment), Clustal W, MSA-EA2002 และ MSA-EA2003

3.3.1 การจัดเรียงองค์รวม (Global alignment)

การจัดเรียงองค์รวมเป็นการจัดเรียงระหว่างลำดับเบส 2 ลำดับ โดยนำกำหนดการพลวัตมาใช้ในการแก้ปัญหา ขั้นตอนวิธีที่มีชื่อเสียงได้แก่ขั้นตอนวิธี Needleman-Wunsch

$$
\begin{aligned}
&S[0,0] = 0 \\
&For\ j = 1\ to\ N\ do \\
&\qquad S[0,j] = S[0,j-1] - d \\
&End \\
&For\ i = 1\ to\ M\ do \\
&\qquad S[i,0] = S[i-1,0] - d \\
&\qquad For\ j = 1\ to\ N\ do \\
&\qquad\qquad Vertical = S[i-1,j] - d \\
&\qquad\qquad Diagonal = S[i-1,j-1] + s(x,y) \\
&\qquad\qquad Horizontal = S[i,j-1] - d \\
&\qquad\qquad S[i,j] = \max\{Vertical, Diagonal, Horizontal\} \\
&\qquad End \\
&End
\end{aligned}
$$

รูปที่3.1 รหัสเทียมของขั้นตอน Needleman-Wunsch

จากขั้นตอน Needleman-Wunsch กำหนดให้ $S[\ ]$ เป็นตาราง 2 มิติ มีความกว้างเท่ากับลำดับเบสที่1 (M) และมีความยาวเท่ากับลำดับเบสที่2 (N) d เป็นค่าการทำโทษเมื่อเกิดแก็ป และ $S[x,y]$ เป็นค่าที่ได้จากตาราง PAM [17] หรือ BLOSUM [18]

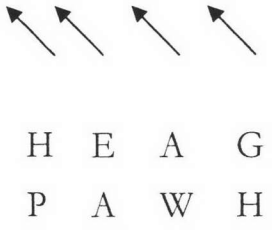
กำหนดการพลวัตมีขั้นตอนการคำนวณคร่าวๆดังนี้

ทำการกำหนดค่าคะแนนลงใน $S[\ ]$ ที่ตำแน่งที่ i และ j ($S[i,j]$) โดยเลือกค่าที่มากที่สุดจาก 3 สมการระหว่าง

$$S[i,j] = \max \begin{cases} S[i-1,j] - d, \\ S[i-1,j-1] + s(x,y), \\ S[i,j-1] - d. \end{cases} \quad (3.1)$$

ทำการจดจำเส้นทางที่ได้คะแนนมากสุดเอาไว้ จากนั้นเก็บคะแนนลงใน $S[i,j]$ จนครบทั้งตาราง ขั้นสุดท้ายให้ทำการเดินกลับไปตามเส้นทางที่ได้คะแนนมากที่สุดที่เคยเก็บไว้ในตอนต้น เราจะได้การจัดเรียงลำดับเบสออกมา

		H	E	A	G
	0	-8	-16	-24	-32
P	-8	-2	-9	-17	-25
A	-16	-10	-3	-4	-12
W	-24	-18	-11	-6	-7
H	-32	-14	-18	-13	-8



รูปที่3.2 การแก้ปัญหาโดยขั้นตอน Needleman-Wunsch โดยใช้ตาราง BLOSUM50 และ d คือ -8

ข้อเสียของวิธีนี้คือเมื่อมีการเพิ่มจำนวนลำดับ และความยาวให้มากขึ้น กำหนดการพลวัตจะยิ่งเสียหน่วยเก็บความจำ และเวลาในการคำนวณมากขึ้นเป็นทวีคูณ ดังนั้นวิธีนี้จึงใช้เปรียบเทียบลำดับเพียง 2-3 ลำดับเท่านั้น

3.3.2 Clustal W

Clustal W [5] เป็นที่ยอมรับ และมีการใช้อย่างแพร่หลายเพื่อใช้แก้ปัญหาการจัดเรียงลำดับเบสหลายลำดับ เนื่องจากใช้เวลาในการคำนวณไม่มากและยังให้คำตอบที่ยอมรับได้ แต่ข้อเสียคือคำตอบที่ได้นั้นอาจเป็นคำตอบที่ไม่ใช่คำตอบที่ดีที่สุด

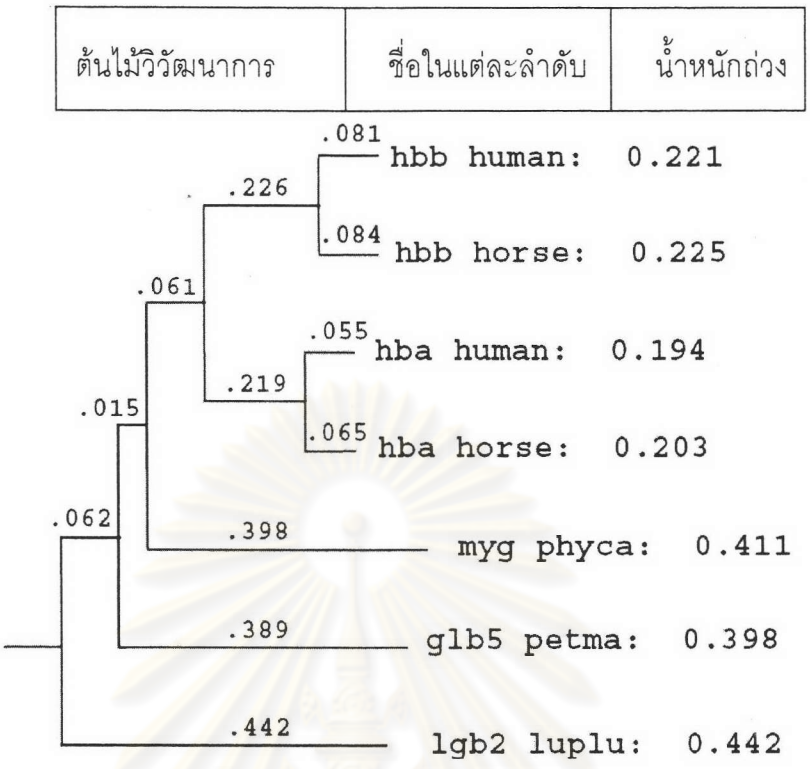
Clustal W นั้นใช้วิธีวิธีการจัดเรียงแบบก้าวหน้าโดยมีขั้นตอนต่างๆ โดยสมมุติให้ทำการจัดเรียงลำดับเบส Hbb_Human, Hbb_Horse, Hba_Human, Hba_Horse, Myg_Phyca, Glb5_Petma และ Lgb2_Luplu การจัดเรียงจะมีขั้นตอนดังนี้

1 ทุกๆคู่ที่เป็นไปได้ของลำดับจะถูกนำมาจัดเรียงแยกกัน โดยใช้กำหนดการพลวัตในการจัดเรียงคู่ลำดับเบส จากนั้นจึงคำนวณหาตารางค่าระยะทาง

Hbb_Human	-					
Hbb_Horse	0.17	-				
Hba_Human	0.59	0.60	-			
Hba_Horse	0.59	0.59	0.13	-		
Myg_Phyca	0.77	0.77	0.75	0.75	-	
Glb5_Petma	0.81	0.82	0.73	0.74	0.80	-
Lgb2_Luplu	0.87	0.86	0.86	0.88	0.93	0.90

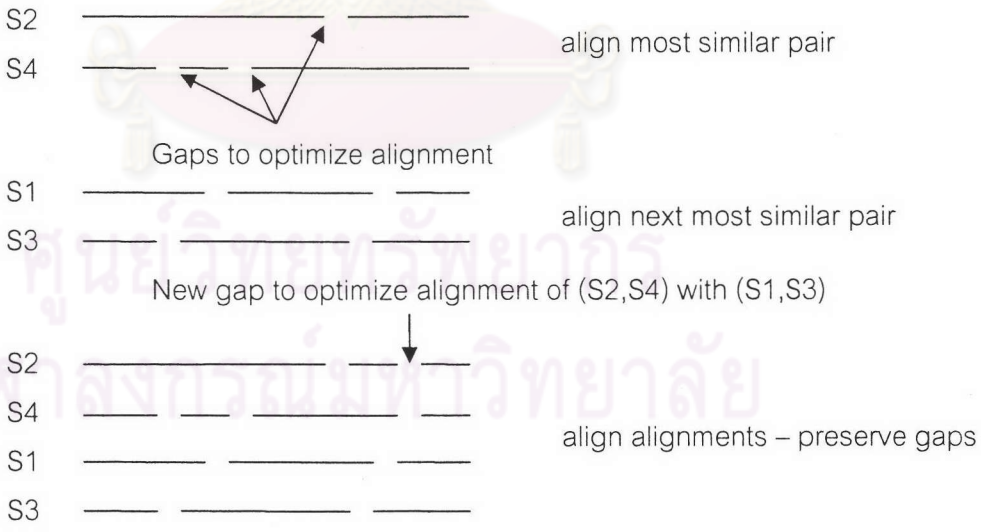
รูปที่3.3 แสดงตารางค่าระยะทาง

2 เมื่อได้ตารางค่าระยะทางมาแล้ว ให้นำค่าที่ได้จากตารางมาสร้างต้นไม้โดยใช้วิธี Neighbor-joining สร้าง Neighbor-joining tree ขึ้นมา



รูปที่3.4 แสดง Neighbor-joining tree

3 เมื่อได้ต้นไม้แล้วเราจะทำการจัดเรียงโดยนำลำดับที่มีความใกล้เคียงกันมาจัดเรียงก่อน จากนั้นจึงค่อยๆจัดเรียงลำดับที่มีความสัมพันธ์ห่างกันออกไปทีหลังตามลำดับ



รูปที่3.5 ยกตัวอย่างสี่ลำดับแรกที่มีการจัดเรียง

4 ลักษณะการให้คะแนน และการทำโทษ

การให้คะแนนมีสองแบบคือ

1 การให้คะแนนโดยไม่มีน้ำหนักถ่วง หรือสมการผลรวมคู่เบส มีสมการคือ

$$SP = \sum_{i,j} A_{i,j} \tag{3.2}$$

2 การให้คะแนนแบบมีน้ำหนักถ่วง หรือสมการผลรวมคู่เบสแบบใช้น้ำหนักมีสมการคือ

$$WSP = \sum_{i,j} w_{i,j} \cdot A_{i,j} \tag{3.3}$$

ซึ่งค่าของ A นั้นจะได้มาจากตาราง PAM [17] หรือตาราง BLOSUM [18] และ w เป็นค่าน้ำหนักที่ได้จาก Neighbor-joining tree

การทำโทษจะมีสมการคือ

$$GapPenaltyScore = GOP + GAPS \times GEP \tag{3.4}$$

เมื่อ GOP คือค่าคงที่เมื่อเจอแก็ปเริ่มต้น $GAPS$ คือจำนวนแก็ปทั้งหมดที่ต่อจากแก็ปเริ่มต้น และ GEP คือค่าคงที่ให้กับแก็ปที่ต่อจากแก็ปเริ่มต้น

3.3.3 MSA-EA2002

เป็นโปรแกรมใช้แก้ปัญหาการจัดเรียงลำดับเบสหลายลำดับโดยใช้ขั้นตอนวิธีพันธุกรรมซึ่ง Thomsen, Fogel และ Krink [8] เป็นผู้พัฒนาขึ้นมา

MSA-EA2002 นั้นมีขั้นตอนในการโปรแกรมดังนี้

การกำหนดค่าอ้างอิง

กำหนดให้ปัญหาการจัดเรียงลำดับเบสหลายลำดับอยู่ในรูปแบบตารางของลำดับ ซึ่งในแต่ละลำดับนั้นจะประกอบไปด้วยตัวอักษร {C,S,T,P,A,G,N,D,E,Q,H,R,K,M,I,L,V,F,Y,W,-,X} อยู่ในตาราง ตัวอักษร "C" ถึง "W" นั้นแสดงถึงกรดอะมิโน (Amino acids) ตัวอักษร "-" นั้นแสดงถึงแก็ป (Gap) ซึ่งในการจัดเรียงนั้นจะบ่งบอกถึงการแทรก และการลบของกรดอะมิโน และตัวอักษร "X" จะแสดงถึงกรดอะมิโนที่ยังไม่เคยพบเห็นมาก่อน

ใน n ลำดับนั้นจะมีความยาวไม่เท่ากันเป็น $l_1, l_2, ..., l_n$ โดยจำนวนหลักที่มีค่ามากที่สุดของตารางคือ $w = (l_{\max} \times 1.5)$ ซึ่ง $l_{\max}$ คือ คามยาวของลำดับเบสที่ยาวที่สุด หรือ $\max(l_1, l_2, ..., l_n)$ เนื่องจากการที่ใช้ 1.5 คูณนั้นเนื่องมาจากต้องการให้มีการจัดเรียงโดยมีพื้นที่เพิ่มขึ้นจากเดิม 50% และหลักที่เป็นแก็ปเหมือนกันจะถูกย้ายไปอยู่ทางด้านขวาสุดเสมอ

การสร้างกลุ่มประชากรของผลเฉลยตั้งต้น

ทำการสร้างประชากรเปรียบเทียบกันระหว่างการสุ่ม และการใช้คำตอบจากโปรแกรม Clustal V [4]

การสร้างประชากรแบบสุ่มมีขั้นตอนดังนี้

1 สำหรับในแต่ละลำดับเบส S_i ที่มีความยาว l_i จะถูกนำมาสลับลำดับ (Permutation) โดยการสุ่มจากความยาวทั้งหมด $1,2,...,w$

2 ทำการเก็บค่าที่สลับลำดับแล้วไว้เท่ากับความยาวจริงของแต่ละลำดับ แล้วทำการเรียงค่าที่สลับลำดับแล้วจากจำนวนน้อยไปหาจำนวนมาก

3 ทำการใส่ค่าตัวอักษรลงในตำแหน่งที่ได้ทำการสลับลำดับ นอกจากนั้นให้ใส่เป็นแก็ปทั้งหมด

```
ID Sequence  L   Permutation(1->10)  Positions     Sorted Positions Initial Alignment
S1 ATCAA     (5) 3 5 2 6 9 1 7 4 8   3 5 2 6 9       2 3 5 6 9          -AT-CA--A
S2 TAATCAA   (7) 9 6 7 1 4 8 5 3 2   9 6 7 1 4 8 5   1 4 5 6 7 8 9      T--AATCAA
S3 ATCA      (4) 6 2 5 1 4 8 3 7 9   6 2 5 1         1 2 5 6            AT--CA---
S4 TAATCAT   (7) 7 4 9 1 3 5 8 6 2   7 4 9 1 3 5 8   1 3 4 5 7 8 9      T-AAT-CAT
S5 ATGATT    (6) 5 6 8 4 3 1 9 7 2   5 6 8 4 3 1     1 3 4 5 6 8        A-TGAT-T-
```

รูปที่3.6 แสดงการสร้างประชากรของผลเฉลยตั้งต้น

หลังจากที่ได้มีการสร้างกลุ่มประชากรตั้งต้นแล้วหลังจากนั้นประชากรจะมีการปรับปรุงผลเฉลยที่ต่างกันออกไปโดยมีวิธีการดังนี้

การกลายพันธุ์

โปรแกรม MSA-EA2002 ใช้ตัวปฏิบัติการทั้งหมด 3 ชนิด โดยแต่ละชนิดจะทำการปรับปรุงผลเฉลยในแบบที่แตกต่างกันออกไปดังนี้

1 LocalShuffle

วิธีนี้จะทำการสุ่มเลือกแถว และทำการสุ่มเลือกตัวอักษร โดยที่ตัวอักษรที่สุ่มมานั้นจะต้องมีแก็ปอยู่ติดกับตัวอักษร หลังจากนั้นจะทำการสลับที่ระหว่างตัวอักษรกับแก็ป ถ้าหากมีแก็ปอยู่ติดกับตัวอักษรมากกว่าหนึ่งตัวแล้วจะทำการสุ่มเลือกตำแหน่งเพื่อที่จะสลับตำแหน่งกัน และหากมีแก็ปทั้งสองด้านแล้วจะทำการสุ่มเลือกด้านที่จะทำการสลับตำแหน่งด้วย

LocalShuffleOne

```
123456789      123456789
-AT-CA-AA      -AT-CA-AA
T--AATCAA      T--AATCAA
AT--CA---      AT--C--A-
T-AAT-CAT      T-AAT-CAT
A-TGAT-T-      A-TGAT-T-
```

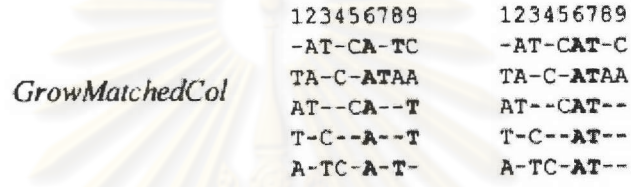
รูปที่3.7 แสดงวิธี LocalShuffle

2 BlockShuffle

วิธีนี้จะมีลักษณะเหมือนกับ LocalShuffle โดยทำการสุ่มเลือกแถว และทำการสุ่มเลือกกลุ่มของตัวอักษร จากนั้นกลุ่มของตัวอักษรจะทำการขยับไปทางซ้ายหรือขวาหนึ่งตำแหน่งถ้าด้านนั้นมีแก็ป ถ้ามีแก็ปทั้งสองด้านแล้วให้ทำการเลือกด้านใดด้านหนึ่งโดยการสุ่ม

3 GrowMatchedColumn

วิธีนี้จะทำการสุ่มหลักที่มีตัวอักษรเหมือนกันทุกตัวยกเว้นแก็ป จากนั้นถ้าตัวอักษรที่อยู่ใกล้กับหลักที่เราสุ่มเลือกไว้เป็นตัวอักษรตัวแบบกันแล้ว เราจะทำการดึงตัวอักษรที่อยู่ใกล้ที่สุดมาชิดกับหลักที่เราสุ่มมาได้ เมื่อทำเสร็จเราจะได้หลักที่มีตัวอักษรเหมือนกันทุกตัวใหม่เพิ่มขึ้น



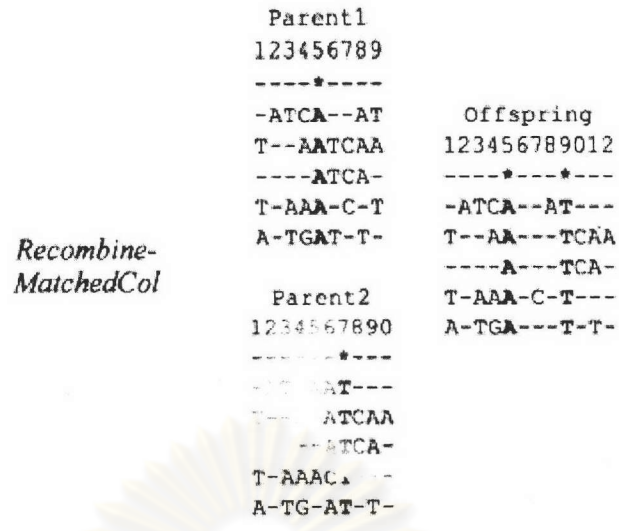
รูปที่3.8 แสดงวิธี GrowMatchedColumn

การไขว้เปลี่ยน

โปรแกรม MSA-EA2002 มีการไขว้เปลี่ยนเพียงชนิดเดียว โดยมีวิธีการปรับปรุงผลเฉลยดังนี้

1 RecombineMatchedColumn

วิธีนี้จะทำการสุ่มเลือกหลักที่มีตัวอักษรเหมือนกันที่ไม่รวมแก็ปจากประชากรสองตัวมาอย่างละตัว ซึ่งไม่รวมตัวอักษรที่มีลำดับเดียวกันจากทั้งประชากรทั้งสอง จากนั้นจะทำการไขว้เปลี่ยนกันจนทำให้เกิดประชากรใหม่ขึ้น โดยประชากรใหม่ที่ได้นั้นจะมีหลักที่มีตัวอักษรเหมือนกันของประชากรก่อนหน้า



รูปที่3.9 แสดงวิธี RecombineMatchedColumn

หลังจากที่มีการกลายพันธุ์ และการไขว้เปลี่ยนเกิดขึ้นแล้วนั้นอาจจะทำให้เกิดหลักที่เป็นแก็ปทั้งหลักขึ้น ดังนั้นจึงต้องใช้ตัวปฏิบัติการณ์ CleanUpGapColumn เสมอ

CleanUpGapColumn

วิธีนี้จะทำการนำเอาหลักที่เป็นแก็ปทั้งหมดนั้นย้ายไปยังตำแหน่งด้านขวาสุดของตาราง หลังจากเกิดการกลายพันธุ์ หรือการไขว้เกี่ยวเกิดขึ้นเสมอ

การวัดค่าความแข็งแรงของผลเฉลย

ค่าความแข็งแรงของผลเฉลยนั้นจะทำโดยเมื่อเจอหลักที่มีตัวอักษรเหมือนกันแล้วจะมีการให้รางวัล และเมื่อเจอแก็ปแล้วจะทำการลงโทษ โดยมีสมการดังนี้

$$fitness = SymbolScore - GapPenaltyScore \tag{3.5}$$

เมื่อ $SymbolScore$ เป็นสมการผลรวมของคู่ลำดับ (Sum-of-pairs) ซึ่งค่านี้จะขึ้นอยู่กับตารางแต่ละชนิดกันเช่นตาราง PAM หรือตาราง BLOSUM เป็นต้น ตารางแต่ละชนิดนั้นจะบอกความเป็นไปได้ในการที่จะเกิดคู่เบสที่เหมือนกัน หรือคู่เบสที่ไม่เหมือนกันได้ ในกรณีของ MSA-EA2002 นั้นใช้ตาราง PAM100

$$SymbolScore = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} PAM(l_i, l_j) \tag{3.6}$$

และเมื่อ $GapPenaltyScore$ ใช้เจาะจงในการทำโทษสำหรับแก็ปที่ถูกจัดเรียง โดยความสัมพันธ์ของค่าในการลงโทษแก็ปคือ

$$GapPenaltyScore = GOP + GAPS \times GEP \tag{3.7}$$

เมื่อ GOP คือค่าคงที่เมื่อเจอแก็ปเริ่มต้น $GAPS$ คือจำนวนแก็ปทั้งหมดที่ต่อจากแก็ปเริ่มต้น และ GEP คือค่าคงที่ให้กับแก็ปที่ต่อจากแก็ปเริ่มต้น โดยโปรแกรม MSA-EA2002 กำหนดค่า GOP เท่ากับ 13 และค่า GEP เท่ากับ 1.3

การค้นหาคำตอบ

ขั้นแรกทำการสร้างกลุ่มประชากรของผลเฉลยตั้งต้น โดยให้มีประชากรทั้งหมด 150 ตัว ต่อมาให้ทำขบวนการวิวัฒนาการจนกว่าจะเจอเงื่อนไขในการออกจากขบวนการ เช่น จำนวนรุ่นเท่ากับ 100000 รุ่น โดยขบวนการวิวัฒนาการนั้นจะกำหนดความน่าจะเป็นที่จะเกิดการกลายพันธุ์เท่ากับ 0.7 ซึ่งวิธีต่างๆนั้นจะมีโอกาสถูกเลือกเท่ากัน และความน่าจะเป็นที่จะเกิดการไขว้เปลี่ยนเท่ากับ 0.8 ส่วนการคัดเลือกนั้นจะใช้วิธีการคัดเลือกแบบแข่งขันโดยมีประชากรที่ใช้ในการแข่งขันสองตัว และประชากรตัวที่มีค่าความแข็งแรงมากจะมีโอกาสถูกเลือกเพื่อนำมาใช้ในประชากรรุ่นต่อไป

โปรแกรม MSA-EA2002 มีขั้นตอนวิธีดังรูปที่3.10 เมื่อ t คือจำนวนรุ่น และ T คือจำนวนรุ่นที่มากที่สุด

ขั้นที่1: กำหนดค่าเริ่มต้นโดยการสุ่ม และจากคำตอบของโปรแกรม Clustal V เป็นประชากรเริ่มต้น

$t = 0$

ขั้นที่2: คำนวนหาค่าความแข็งแรงให้กลุ่มประชากร

ขั้นที่3: ตรวจเงื่อนไขการหยุดโปรแกรมเมื่อ $t \geq T$

ขั้นที่4: ทำการกลายพันธุ์ หรือการไขว้เปลี่ยนเพื่อปรับปรุงผลเฉลย

ขั้นที่5: คัดเลือกประชากรรุ่นถัดไป

ขั้นที่6: คำนวณฟังก์ชันวัตถุประสงค์ให้กับกลุ่มประชากรรุ่นถัดไป

$t = t + 1$

ทำซ้ำขั้นที่3

รูปที่3.10 รหัสเทียมขั้นตอนวิธีของ MSA-EA2002

3.3.4 MSA-EA2003

เป็นโปรแกรมใช้แก้ปัญหาการจัดเรียงลำดับเบสหลายลำดับโดยใช้ขั้นตอนวิธีเชิงวิวัฒน์ซึ่ง Thomsen, Fogel และ Krink [9] เป็นผู้พัฒนาต่อจาก MSA-EA2002

MSA-EA2003 นั้นมีขั้นตอนในการโปรแกรมดังนี้

การกำหนดค่าอ้างอิง

ใช้การกำหนดค่าอ้างอิงแบบเดียวกันกับ MSA-EA2002 ยกเว้นการเพิ่มความยาวของผลเฉลยจากเดิม 50 % เปลี่ยนเป็น 20 % ($w = (l_{\max} \times 1.2)$)

การสร้างกลุ่มประชากรของผลเฉลยตั้งต้น

เนื่องจากการสร้างกลุ่มประชากรของผลเฉลยตั้งต้นของ MSA-EA2002 นั้นมีการใช้การสร้างกลุ่มประชากรแบบสุ่มทำให้ใช้เวลานานในการหาผลเฉลยที่ถูกต้อง ดังนั้นใน MSA-EA2003 จึงใช้กลุ่มประชากรของผลเฉลยต้นจากผลที่ได้จากโปรแกรม Clustal X เพียงอย่างเดียว ซึ่งผลที่ได้มานั้นเข้าใกล้ผลเฉลยที่ดีที่สุดโดยไม่เสียเวลาในหาคำตอบนานจนเกินไป

จากการทดลองใน MSA-EA2002 นั้นได้แสดงว่าถึงมีการไขว้เปลี่ยนแต่ก็ไม่ทำให้ผลเฉลยดีขึ้น ดังนั้นในการทดลอง MSA-EA2003 จึงได้ตัดการไขว้เปลี่ยนออก และใช้เพียงแต่การกลายพันธุ์เท่านั้น

การกลายพันธุ์

โปรแกรม MSA-EA2003 ใช้การกลายพันธุ์ทั้งหมด 5 ชนิด โดยใช้การกลายพันธุ์ LocalShuffle และ BlockShuffle จาก MSA-EA2002 และได้ทำการสร้างการกลายพันธุ์แบบใหม่อีก 3 ชนิด ดังนี้

1 DirectedLocalShuffle

วิธีนี้จะมีลักษณะคล้าย LocalShuffle โดยทำการสุ่มเลือกแถว และทำการสุ่มเลือกตัวอักษร จากนั้นจะทำการสลับที่กันระหว่างตัวอักษรกับแก็ป โดยการสลับที่กันนั้นจะสลับตัวอักษรไปยังที่ที่มีค่าความแข็งแรงมากที่สุด

2 PassGaps

วิธีนี้ในขั้นแรกจะทำการสุ่มเลือกแถว แล้วจะทำการหาบริเวณที่เป็นแก็ปทั้งหมดในแถว จากนั้นทำการสุ่มเลือกบริเวณที่เป็นแก็ปมา ตัวอักษรทางด้านซ้าย หรือขวาของบริเวณแก็ปที่ถูกเลือกจะเคลื่อนที่ไปยังอีกฟากหนึ่งของบริเวณแก็ปนั้น

3 RandomMoveGap

วิธีนี้ตอนแรกจะทำการสุ่มเลือกแถว และสุ่มเลือกแก็ปในแถวนั้น จากนั้นจะทำการเคลื่อนที่แก็ปที่ถูกเลือกโดยการสุ่มตำแหน่ง โดยตำแหน่งที่ทำการสุ่มนั้นจะอยู่ระหว่างตัวอักษรตัวแรกไปจนถึงตัวอักษรตัวสุดท้าย

หลังจากที่ได้ทำการกลายพันธุ์แล้วจะต้องทำการลบแถวที่เป็นแก็ปโดยใช้ตัวปฏิบัติการณ์ CleanUpGapColumn เสมอ

การวัดค่าความแข็งแรงของผลเฉลย

ใช้การวัดค่าความแข็งแรงแบบเดียวกับ MSA-EA2002 แต่แตกต่างกันโดย

1. เปลี่ยนตารางในการให้รางวัลจาก $PAM100$ เป็น $BLOSUM62$
2. เปลี่ยนค่า GOP จาก 13 เป็น 10
3. เปลี่ยนค่า GEP จาก 1.3 เป็น 1

การค้นหาคำตอบ

ขั้นแรกทำการสร้างกลุ่มประชากรของผลเฉลยตั้งต้น และทำการวิวัฒนาการจนกว่าจะเจอเงื่อนไขในการออกจากขบวนการ ซึ่งกำหนดไว้ที่ 5000 รุ่น โดยขบวนการในแต่ละรุ่นจะมีผลเฉลยในประชากรลูก (Offspring population) 30 ตัว และจะมีผลเฉลยในประชากรพ่อแม่ (Parent population) 15 ตัว โดยประชากรพ่อแม่ในรุ่นใหม่นั้นจะคัดเลือกจากประชากรที่มีค่าความแข็งแรงสูงสุดในประชากรลูก และพ่อแม่จากรุ่นก่อนหน้า ส่วนการปรับปรุงผลเฉลยนั้นจะกำหนดความน่าจะเป็นที่จะเกิดการกลายพันธุ์เท่าๆกัน ส่วนการสร้างประชากรพ่อแม่ในรุ่นใหม่นั้นจะทำการเลือกผลเฉลยที่มีค่าความแข็งแรงมากที่สุดจากประชากรลูก และประชากรพ่อแม่ในรุ่นก่อน

โปรแกรม MSA-EA2003 มีขั้นตอนวิธีดังรูปที่3.11 เมื่อ t คือจำนวนรุ่น และ T คือจำนวนรุ่นที่มากที่สุด

ขั้นที่1: กำหนดค่าเริ่มต้นจากคำตอบของโปรแกรม Clustal W เป็นกับประชากรเริ่มต้น

$t = 0$

ขั้นที่2: คำนวนหาค่าประภาพให้กลุ่มประชากร

ขั้นที่3: ตรวจเงื่อนไขการหยุดโปรแกรมเมื่อ $t \geq T$

ขั้นที่4: ทำการกลายพันธุ์ หรือการไขว้เปลี่ยนเพื่อปรับปรุงผลเฉลย

ขั้นที่5: คัดเลือกประชากรรุ่นถัดไปด้วยวิธี μ+λ selection

ขั้นที่6: คำนวณฟังก์ชันวัตถุประสงค์ให้กับกลุ่มประชากรรุ่นถัดไป

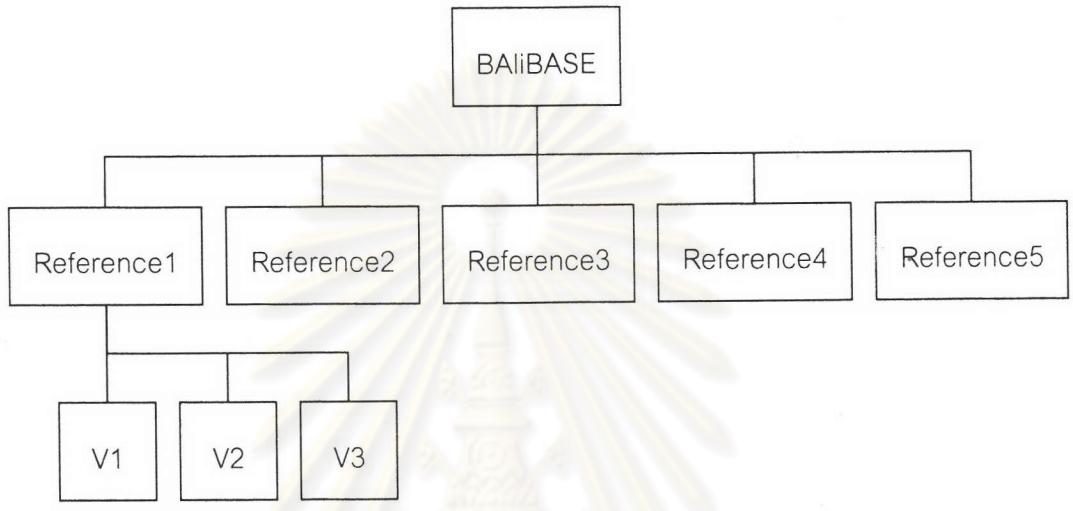
$t = t + 1$

ทำซ้ำขั้นที่3

รูปที่3.11 รหัสเทียมขั้นตอนวิธีของ MSA-EA2003

3.4 ฐานข้อมูล BAliBASE

ฐานข้อมูล BAliBASE [16] เป็นฐานข้อมูลที่ได้จากโครงสร้างโปรตีนสามมิติ ซึ่งรู้ลักษณะของลำดับโปรตีนที่แน่นอน ฐานข้อมูลนี้ประกอบไปด้วยกลุ่มลำดับอ้างอิง (Reference set of sequence alignments) จำนวน 142 กลุ่ม จากกลุ่มลำดับโปรตีนอ้างอิงทั้งหมดสามารถจำแนกออกได้เป็น 5 สารบบอ้างอิง โดยสารบบแรกจะมี 3 สารบบอ้างอิงย่อย ดังรูปที่3.12



รูปที่3.12 สารบบในฐานข้อมูล BAliBASE

แต่ละสารบบอ้างอิงสามารถแบ่งตามลักษณะความยาวของกลุ่มลำดับอ้างอิงได้ คือกลุ่มที่มีความยาวสั้น กลุ่มที่มีความยาวปานกลาง และกลุ่มที่มีความยาวมาก ซึ่งกลุ่มลำดับอ้างอิงในแต่ละสารบบอ้างอิงแบ่งได้ดังตารางที่3.1

Reference	Short (< 100 residues)	Medium (200-300 residues)	Long (>140 residues)
Reference1			
V1	7	8	8
V2	10	9	10
V3	10	10	8
Reference2	9	8	7
Reference3	5	3	5
Reference4	12		
Reference5	12		

ตารางที่3.1 แสดงลักษณะความยาวของกลุ่มลำดับอ้างอิงในแต่ละสารบบ

สารบบอ้างอิงที่1 ประกอบด้วยลำดับที่มีความยาวใกล้เคียงกัน ในสารบบอ้างอิงย่อยที่1 (V1) มีกลุ่มลำดับอ้างอิงที่มีร้อยละของเอกลักษณ์ (Percent identity) น้อยกว่าร้อยละ25 สารบบอ้างอิงย่อยที่2 มีกลุ่มลำดับอ้างอิงที่มีร้อยละของเอกลักษณ์ระหว่าง ร้อยละ20 ถึงร้อยละ40 และสารบบย่อยอ้างอิงที่3 มีกลุ่มลำดับอ้างอิงที่มีร้อยละของเอกลักษณ์มากกว่า ร้อยละ30

สารบบอ้างอิงที่2 ประกอบด้วยลำดับที่สายพันธุ์ (Families) มีความสัมพันธ์ใกล้เคียงกัน โดยมีร้อยละเอกลักษณ์ มากกว่าร้อยละ25 และประกอบด้วยกลุ่มลำดับออร์แฟน (Orphan)

สารบบอ้างอิงที่3 ประกอบด้วยลำดับที่มีความหลากหลายทางสายพันธุ์ โดยแต่ละลำดับที่ต่างสายพันธุ์จะมีร้อยละของเอกลักษณ์ น้อยกว่า ร้อยละ25

สารบบอ้างอิงที่4 ประกอบด้วยลำดับที่มีส่วนการขยายส่วนเอ็นซี (N/C-terminal extension)

สารบบอ้างอิงที่5 ประกอบด้วยลำดับที่มีส่วนการแทรกปลายเอ็นซี (N/C-terminal insertion)

การคำนวณความถูกต้อง

ฐานข้อมูล BAliBASE มีการคำนวณอยู่ 2 สมการเพื่อบ่งบอกความถูกต้องในการจัดลำดับเบสหลายลำดับเมื่อเทียบกับฐานข้อมูล สมการแรกเป็นสมการค่าผลรวมคู่เบส (Sum-of-pairs

score, SPS) ซึ่งจะเป็นค่าบอกถึงลำดับที่ถูกจัดเรียงได้ถูกต้อง สมการที่สองเป็นสมการค่าแถว (Column score, CS) ซึ่งจะบ่งบอกถึงความสามารถของโปรแกรมในการจัดเรียงลำดับทุกแถวได้ถูกต้อง

ในการคำนวณสมการค่าผลรวมคู่เบส กำหนดให้การจัดเรียงทดสอบ (Test alignment) มี N ลำดับ M แถว และตัวอักษรในลำดับเบสเป็น $A_{i1}, A_{i2}, ..., A_{iN}$ ในแต่ละคู่ของตัวอักษร A_{ij} และ A_{ik} จะกำหนดให้ $p_{ijk} = 1$ ถ้าคู่ของตัวอักษร A_{ij} และ A_{ik} มีการจัดเรียงเหมือนในการจัดเรียงอ้างอิง (Reference alignment) นอกจากนั้น $p_{ijk} = 0$ ค่า S_i สำหรับแถวที่ i จะมีค่าดังสมการ

$$S_i = \sum_{j=1, j\neq k}^{N} \sum_{k=1}^{N} p_{ijk} \tag{3.8}$$

สมการค่าผลรวมคู่เบสคือ

$$SPS = \sum_{i=1}^{M} \frac{S_i}{\sum_{i=1}^{M_r} S_{ri}} \tag{3.9}$$

เมื่อ M_r คือจำนวนแถวทั้งหมดในการจัดเรียงอ้างอิง และ S_{ri} คือค่า S_i สำหรับแถวที่ i ของการจัดเรียงอ้างอิง

ในการคำนวณสมการค่าแถว จากการกำหนดเหมือนสมการค่าผลรวมคู่เบส ค่า $C_i = 1$ ถ้าตัวอักษรในแถวเหมือนการจัดเรียงอ้างอิง นอกจากนั้น $C_i = 0$

สมการค่าแถวคือ

$$CS = \sum_{i=1}^{M} \frac{C_i}{M} \tag{3.10}$$