Βιοπληροφορική

Διάλεξη 5η :

Στοίχιση πολλών αλληλουχιών :
Αλγόριθμοι, προγράμματα, και εφαρμογές τους.

Εισαγωγή

Η ανάγκη στοίχισης πολλών αλληλουχιών προκύπτει σε μια πληθώρα περιπτώσεων :

- Ταυτοποίηση συγγενών αλληλουχιών (για τη δημιουργία οικογενειών).
- Εύρεση φυλογενετικών σχέσεων μεταξύ των μελών μιας οικογένειας αλληλουχιών (σε συνδυασμό με τους αλγόριθμους φυλογενετικών δένδρων).
- Εύρεση συντηρημένων μοτίβων και αυστηρά συντηρημένων καταλοίπων.
- Βελτίωση της ευαισθησίας της έρευνας των βάσεων δεδομένων μέσω της χρήσης ολόκληρων στοιχίσεων (αντί απλών αλληλουχιών).

Εισαγωγή

Μία στοίχιση πολλών αλληλουχιών έχει τη μορφή δισδιάστατου πίνακα στον οποίο οι γραμμές αντιστοιχούν στις αλληλουχίες και οι στήλες στις αμινοξικές θέσεις. Για παράδειγμα :

```
A  S  P  -  E  R  A
A  -  P  T  E  R  A
A  S  -  T  -  R  A
```

Όπως και για τη στοίχιση δύο αλληλουχιών, στόχος της στοίχισης πολλών αλληλουχιών είναι να αποκαλύψει τις μεταξύ τους εξελικτικές σχέσεις.

Ομοιότητα με τη στοίχιση δύο αλληλουχιών

Το πρόβλημα της στοίχισης πολλών αλληλουχιών φαίνεται να είναι μια άμεση επέκταση του προβλήματος της στοίχισης δύο αλληλουχιών : αντί να βρίσκουμε το βέλτιστο μονοπάτι σε ένα δισδιάστατο πίνακα [όπως κάναμε με τους αλγόριθμους δυναμικού προ-γραμματισμού (N&W, S&W)], τώρα θα αναζητούμε το βέλτιστο μονοπάτι σε ένα πίνακα υψηλότερης διάστασης (σε ένα τρισδιάστατο πίνακα για τρεις αλληλουχίες, σε ένα τετραδιάστατο πίνακα για τέσσερις, κοκ).

Υπολογιστικό πρόβλημα

Το πρόβλημα με την προσέγγιση αυτή είναι το υπολογιστικό κόστος : για δύο αλληλουχίες με μήκη m το κόστος εύρεσης της βέλτιστης στοίχισης είναι (m·m). Για τρεις θα είναι (m·m·m), και για K αλληλουχίες θα είναι (m^K). Έτσι επιστρέψαμε σε ένα υπολογιστικό κόστος εκθετικά ανάλογο του αριθμού των αλληλουχιών, μόνο που σε αυτή την περίπτωση δεν είναι 'φαινομενικό' (όπως είχαμε δείξει για την περίπτωση της στοίχισης δυο αλληλουχιών), αλλά πραγματικό. Έτσι, το υπολογιστικό κόστος της στοίχισης τεσσάρων αλληλουχιών μήκους 100 κατάλοιπων είναι το ίδιο με το κόστος της πραγματοποίησης 10000 στοιχίσεων μεταξύ δύο τέτοιων αλληλουχιών.

Υπολογιστικό πρόβλημα

Για μικρό αριθμό αλληλουχιών (π.χ. τρεις αλληλουχίες) υπάρχουν αλγόριθμοι (και προγράμματα) για την εύρεση της βέλτιστης μεταξύ τους στοίχισης. Αυτός ο περιορισμός σε αριθμό αλληλουχιών μειώνει κατά πολύ τη βιολογική χρησιμότητα αυτών των αλγορίθμων.

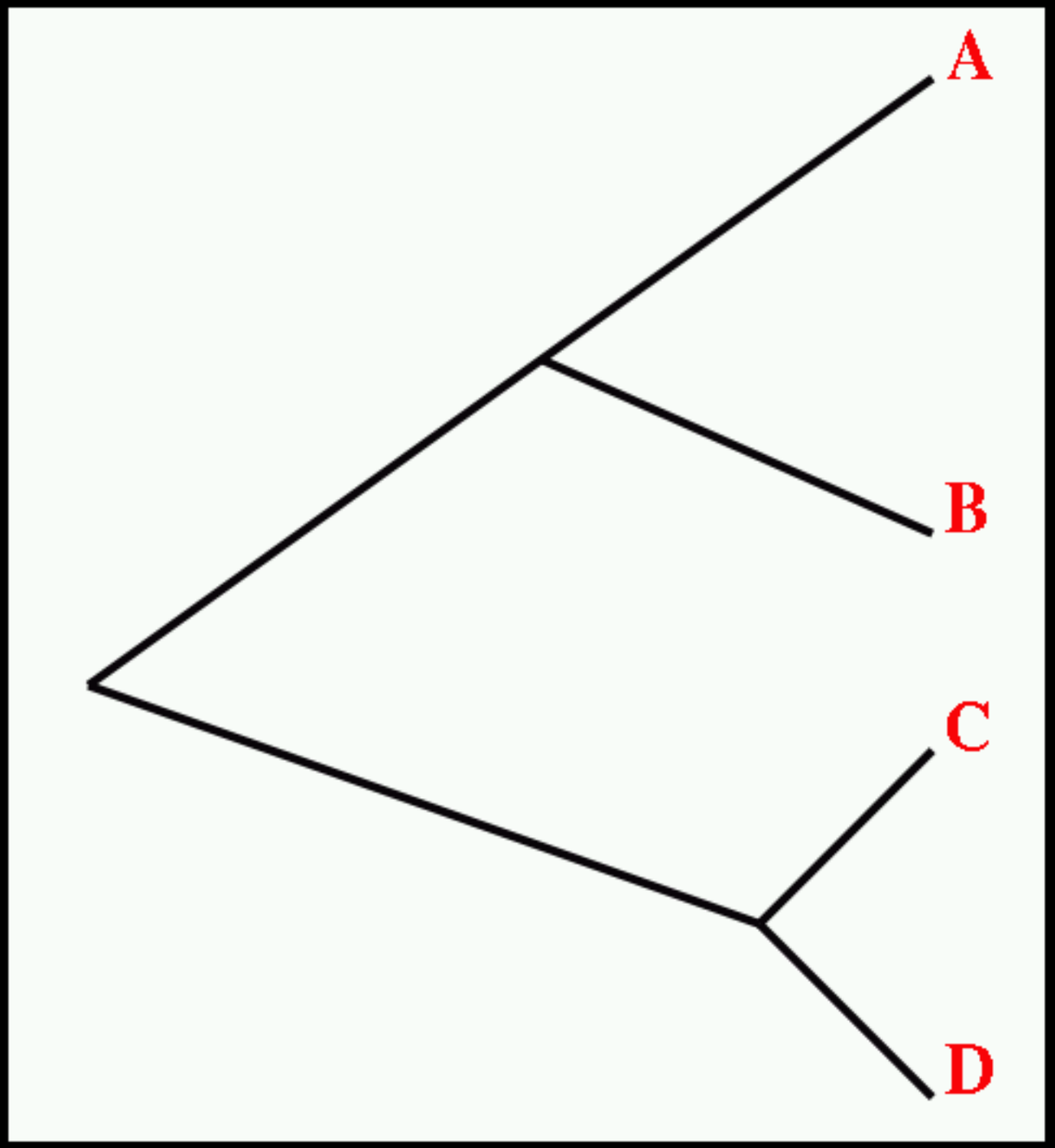
Για αυτό το λόγο, ο χώρος της στοίχισης πολλών αλληλουχιών κυριαρχείται από ευρεστικούς αλγόριθμους οι οποίοι χρησιμοποιούν επιπλέον υποθέσεις και περιορισμούς προκειμένου να μειώσουν το υπολογιστικό κόστος. Οι πλέον γνωστοί αλγόριθμοι για στοίχιση πολλών αλληλουχιών ανήκουν στις λεγόμενες προοδευτικές (progressive) μεθόδους (επίσης γνωστές και ως ιεραρχικές).

Ιεραρχικές μέθοδοι

Η βασική ιδέα πίσω από αυτές τις μεθόδους είναι ότι η ύπαρξη εξελικτικών σχέσεων μεταξύ των αλληλουχιών καθιστά περιττή την αναζήτηση μιας καθολικά βέλτιστης στοίχισης : η στοίχιση που αναζητούμε είναι αυτή που καλύτερα αναπαριστά τις μεταξύ τους εξελικτικές σχέσεις και όχι αυτή που αποδίδει το σύνολο των μεταξύ τους ομοιοτήτων.

Έτσι, εάν για παράδειγμα γνωρίζαμε ότι για τέσσερις αλληλουχίες A, B, C, D ισχύει το δένδρο :

Ιεραρχικές μέθοδοι



τότε, η άμεση αναζήτηση σχέσεων μεταξύ των αλληλουχιών A και D περιττεύει. Καλύτερο είναι το :

Ιεραρχικές μέθοδοι

- Βρες την εξελικτική σχέση ανάμεσα στις αλληλουχίες A & B. Η μεταξύ τους ομολογία αναπαριστά με πληρότητα ό,τι μπορούμε να συνάγουμε για την κοινή τους προγονική αλληλουχία (τον κόμβο του δένδρου από τον οποίο προήλθαν).
- Βρες την εξελικτική σχέση ανάμεσα στις αλληλουχίες C & D. Η μεταξύ τους ομολογία αναπαριστά με πληρότητα ό,τι μπορούμε να συνάγουμε για την κοινή τους προγονική αλληλουχία (τον κόμβο του δένδρου από τον οποίο προήλθαν).
- Χρησιμοποίησε ό,τι γνωρίζεις για τους δύο ενδιάμεσους κόμβους για να συνάγεις την μεταξύ τους εξελικτική σχέση (και με τη ρίζα του δένδρου).

Ιεραρχικές μέθοδοι

Άρα, το "στοίχισε τις αλληλουχίες A,B,C,D" μετασχηματίστηκε στο :

- Στοίχισε τις A & B => AB
- Στοίχισε τις C & D => CD
- Στοίχισε τις AB & CD => ABCD

Η διαφορά από άποψη υπολογιστικού κόστους είναι τεράστια : εάν οι τέσσερις αλληλουχίες είχαν μήκος 200 καταλοίπων, η καινούργια μέθοδος θα ήταν ~13000 φορές πιο γρήγορη από μια σχολαστική μέθοδο στοίχισης.

Ακόμη ένας φαύλος κύκλος ;

Η μέθοδος που παρουσιάστηκε έχει ένα ουσιώδες πρόβλημα : για να βρούμε τις φυλογενετικές σχέσεις μεταξύ των αλληλουχιών χρειαζόμαστε την μεταξύ τους στοίχιση (όλων των αλληλουχιών). Άρα, για να βρούμε τη στοίχιση χρειαζόμαστε το προϊόν της. Το αδιέξοδο αυτό αίρεται μέσω της παραδοχής ότι ένα δενδρόγραμμα-οδηγός για τις εξελικτικές σχέσεις μεταξύ των αλληλουχιών μπορεί να δημιουργηθεί λαμβάνοντας υπόψη μόνο τις ομοιότητες μεταξύ ζευγών αλληλουχιών (χωρίς τη δημιουργία μιας στοίχισης όλων των αλληλουχιών).

Ακόμη ένας φαύλος κύκλος ;

Για παράδειγμα, υποθέστε ότι για τρεις αλληλουχίες Α,Β,Γ πραγματοποιήσαμε όλες τις δυνατές ανά ζεύγη στοιχίσεις, και ελέγξαμε κάθε μια από αυτές με βάση το Z-τεστ που αναφέρθηκε στην προηγούμενη διάλεξη. Τα αποτελέσματα (με τη μορφή πίνακα) ήταν :

	Α	Β	Γ
Α	-	9	4
Β	9	-	5
Γ	4	5	-

Από αυτόν το πίνακα μπορούμε χωρίς καμία επιπλέον ανάλυση να συνάγουμε ότι οι αλληλουχίες Α και Β είναι πιο στενά συνδεδεμένες μεταξύ τους απ' ότι κάθε μία από αυτές με την αλληλουχία Γ.

Ακόμη ένας φαύλος κύκλος ;

Συνεπώς, αυτό που θα κάναμε σε αυτή την περίπτωση είναι να στοιχίσουμε τις A & B (χρησιμοποιώντας για παράδειγμα τον αλγόριθμο των N & W), και στη συνέχεια, να στοιχίσουμε (πάλι με τον N & W) την αλληλουχία Γ με την προϋπάρχουσα στοίχιση των A & B.

Το οποίο μας φέρνει στην ερώτηση πως στοιχίζουμε βέλτιστα (κατά N & W) όχι δυο αλληλουχίες, αλλά μια αλληλουχία και μια ολόκληρη στοίχιση, ή ακόμα και δύο στοιχίσεις μεταξύ τους.

Στοίχιση στοιχίσεων

Η βασική απλοποίηση του προβλήματος προκύπτει από την απαίτηση ότι οι ήδη στοιχισμένες αλληλουχίες θα χρησιμοποιηθούν όχι ως ανεξάρτητες αλληλουχίες, άλλα ως μια (μικτή) αλληλουχία : το ποιο αμινοξύ της μιας είναι στοιχισμένο με ποιο αμινοξύ της άλλης πρόκειται να μείνει αμετάβλητο. Εάν απαιτηθεί η προσθήκη κενών, αυτά (τα κενά) μπαίνουν ταυτόχρονα σε όλες τις ήδη στοιχισμένες αλληλουχίες. Π.χ.

```
ASFKLMEMNERA
     +                          ASFKLMEMTERA
ASPERA              ==>         AS------PERA
APTERA                          AP------TERA
```

Στοίχιση στοιχίσεων

Η ουσιαστική διαφορά από τον απλό αλγόριθμο του Ν & W, έγκειται στον τρόπο βαθμολόγησης τόσο για τις στοιχίσεις μεταξύ αμινοξέων όσο και για την εισαγωγή κενών. Για πληρότητα θα αναφέρουμε έναν από τους πλέον απλοϊκούς αθροιστικούς τρόπους βαθμολόγησης :

Η βαθμολογία της στοίχισης Κ αμινοξέων από μία θέση μίας στοίχισης Α, με Λ αμινοξέα από μία στοίχιση Β, είναι ίση το άθροισμα των βαθμολογιών υποκατάστασης κάθε αμινοξέος (από τα Κ) της στοίχισης Α με κάθε ένα (από τα Λ) της στοίχισης Β. Για παράδειγμα :

Στοίχιση στοιχίσεων

Έστω δύο στοιχίσεις τεσσάρων αλληλουχιών,

ASPERA και AMEMPTA
APTERA ASE-PTA

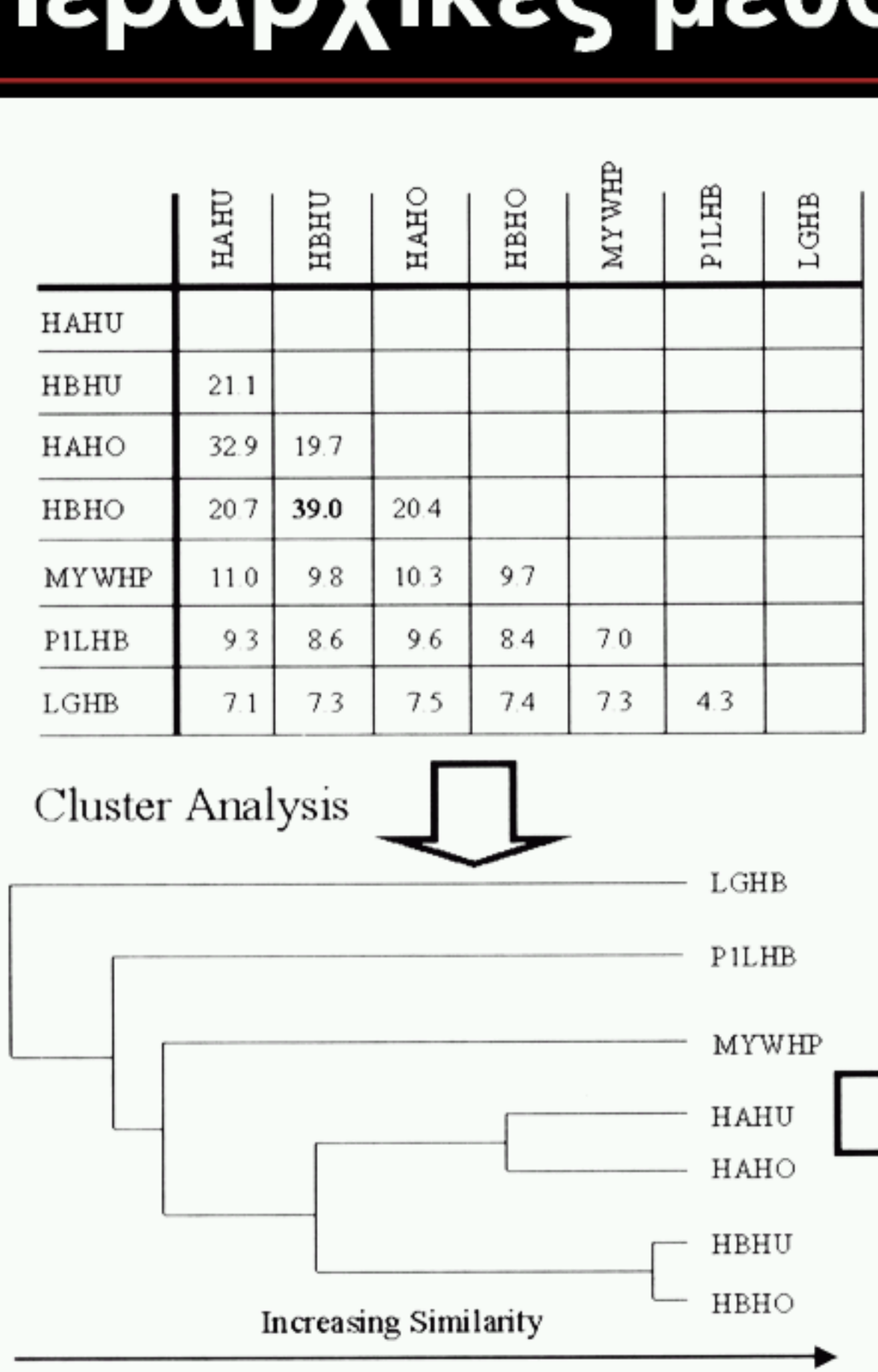
Η βαθμολογία της στοίχισης των P-T (τρίτη θέση πρώτης στοίχισης) με τα M-S (δεύτερη θέση δεύτερης στοίχισης) θα είναι

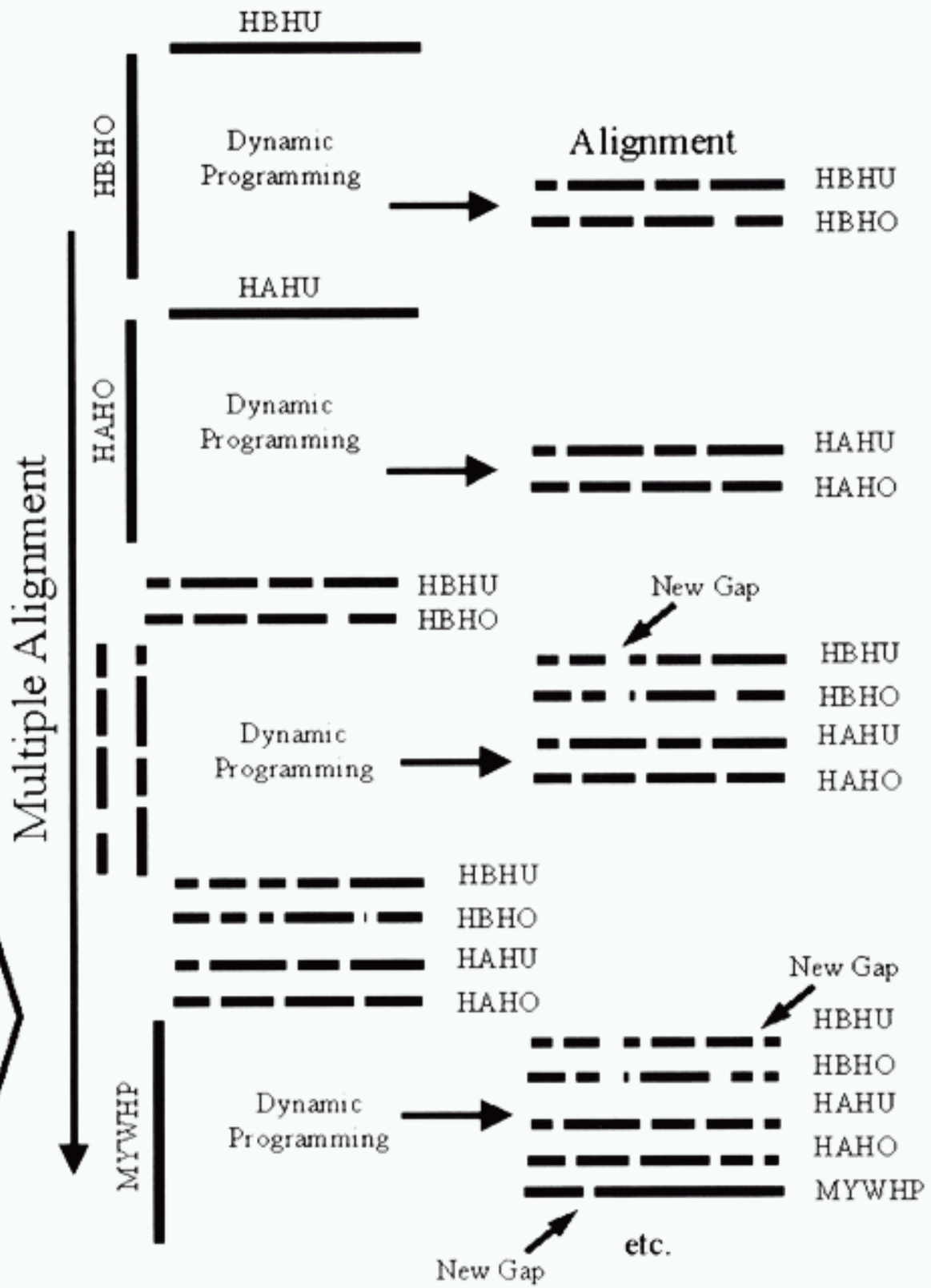
$$\Sigma(P,M)+\Sigma(P,S)+\Sigma(T,M)+\Sigma(T,S)$$

όπου Σ είναι ο πίνακας βαθμολόγησης (π.χ. PAM250).

Ιεραρχικές μέθοδοι

	HAHU	HBHU	HAHO	HBHO	MYWHP	P1LHB	LGHB
HAHU							
HBHU	21.1						
HAHO	32.9	19.7					
HBHO	20.7	**39.0**	20.4				
MYWHP	11.0	9.8	10.3	9.7			
P1LHB	9.3	8.6	9.6	8.4	7.0		
LGHB	7.1	7.3	7.5	7.4	7.3	4.3	





Παράδειγμα

Θέλουμε να στοιχίσουμε τις αλληλουχίες ASPERA ASTRA και APTERA. Ο (μοναδιαίος) πίνακας βαθμολόγησης είναι :

	A	C	M	P	F	...
A	3					
C	0	3				
M	0	0	3			
P	0	0	0	3		
F	0	0	0	0	3	
...						

και το gap penalty έχει τιμή 2 (ανά κενό).

Παράδειγμα

Η μέθοδος επίλυσης είναι :

- Στοιχίζουμε τις αλληλουχίες ASPERA και APTERA
  => Βαθμολογία Σ1
- Στοιχίζουμε τις αλληλουχίες ASPERA και ASTRA
  => Βαθμολογία Σ2
- Στοιχίζουμε τις αλληλουχίες APTERA και ASTRA
  => Βαθμολογία Σ3
- Για το ζεύγος αλληλουχιών με την υψηλότερη βαθμολογία βρίσκουμε τη βέλτιστη στοίχιση.
- Τέλος, στοιχίζουμε το ήδη στοιχισμένο ζεύγος με την τρίτη αλληλουχία.

Παράδειγμα

ASPERA-APTERA

		A	S	P	E	R	A
	0	-2	-4	-6	-8	-10	-12
A	-2						
P	-4						
T	-6						
E	-8						
R	-10						
A	-12						

Παράδειγμα

ASPERA-APTERA

		A	S	P	E	R	A
	0	-2	-4	-6	-8	-10	-12
A	-2	3	1	-1	-3	-5	-7
P	-4	1	3	4	2	0	-2
T	-6	-1	1	3	4	2	0
E	-8	-3	-1	1	6	4	2
R	-10	-5	-3	-1	4	9	7
A	-12	-7	-5	-3	2	7	12

Παράδειγμα

ASPERA-ASTRA

		A	S	P	E	R	A
	0	-2	-4	-6	-8	-10	-12
A	-2						
S	-4						
T	-6						
R	-8						
A	-10						

Παράδειγμα

ASPERA-ASTRA

		A	S	P	E	R	A
	0	-2	-4	-6	-8	-10	-12
A	-2	3	1	-1	-3	-5	-7
S	-4	1	6	4	2	0	-2
T	-6	-1	4	6	4	2	0
R	-8	-3	2	4	6	7	5
A	-10	-5	0	2	4	6	10

Παράδειγμα

APTERA-ASTRA

		A	P	T	E	R	A
	0	-2	-4	-6	-8	-10	-12
A	-2						
S	-4						
T	-6						
R	-8						
A	-10						

Παράδειγμα

APTERA-ASTRA

		A	P	T	E	R	A
	0	-2	-4	-6	-8	-10	-12
A	-2	3	1	-1	-3	-5	-7
S	-4	1	3	1	-1	-3	-5
T	-6	-1	1	6	4	2	0
R	-8	-3	-1	4	6	7	5
A	-10	-5	-3	2	4	6	10

Παράδειγμα

Βέλτιστη στοίχιση ASPERA-APTERA

		A	S	P	E	R	A
	0	-2	-4	-6	-8	-10	-12
A	-2	3*	1	-1	-3	-5	-7
P	-4	1	3*	4	2	0	-2
T	-6	-1	1	3*	4	2	0
E	-8	-3	-1	1	6*	4	2
R	-10	-5	-3	-1	4	9*	7
A	-12	-7	-5	-3	2	7	12*

A S P E R A

A P T E R A

Παράδειγμα

ASPERA/APTERA vs ASTRA

		A/A	S/P	P/T	E/E	R/R	A/A
	0	-2	-4	-6	-8	-10	-12
A	-2						
S	-4						
T	-6						
R	-8						
A	-10						

Παράδειγμα

ASPERA/APTERA vs ASTRA

		A/A	S/P	P/T	E/E	R/R	A/A
	0	-2	-4	-6	-8	-10	-12
A	-2	6	4	2	0	-2	-4
S	-4	4	9	7	5	3	1
T	-6	2	7	12	10	8	6
R	-8	0	5	10	12	16	14
A	-10	-2	3	8	10	14	22

Παράδειγμα

ASPERA/APTERA vs ASTRA

		A/A	S/P	P/T	E/E	R/R	A/A
	0	-2	-4	-6	-8	-10	-12
A	-2	6*	4	2	0	-2	-4
S	-4	4	9*	7	5	3	1
T	-6	2	7	12*	10*	8	6
R	-8	0	5	10	12	16*	14
A	-10	-2	3	8	10	14	22*

A S P E R A

A P T E R A

A S T - R A

Παράδειγμα 2

Εάν η τιμή του gap penalty ήταν ίση με 1 η βέλτιστη στοίχιση των ASPERA και APTERA θα ήταν :

```
A S P - E R A
A - P T E R A
```

οπότε ο πίνακας στοίχισης του ASTRA με αυτή τη στοίχιση θα ήταν :

Παράδειγμα 2

ASPERA/APTERA vs ASTRA

		A/A	S/-	P/P	-/T	E/E	R/R	A/A
	0	-1	-2	-3	-4	-5	-6	-7
A	-1							
S	-2							
T	-3							
R	-4							
A	-5							

Παράδειγμα 2

ASPERA/APTERA vs ASTRA

		A/A	S/-	P/P	-/T	E/E	R/R	A/A
	0	-1	-2	-3	-4	-5	-6	-7
A	-1	6	5	4	3	2	1	0
S	-2	5	9	8	7	6	5	4
T	-3	4	8	9	11	10	9	8
R	-4	3	7	8	10	11	16	15
A	-5	2	6	7	9	10	15	22

Παράδειγμα 2

ASPERA/APTERA vs ASTRA

		A/A	S/-	P/P	-/T	E/E	R/R	A/A
	0	-1	-2	-3	-4	-5	-6	-7
A	-1	6*	5	4	3	2	1	0
S	-2	5	9*	8*	7	6	5	4
T	-3	4	8	9	11*	10*	9	8
R	-4	3	7	8	10	11	16*	15
A	-5	2	6	7	9	10	15	22*

A S P - E R A

A - P T E R A

A S - T - R A

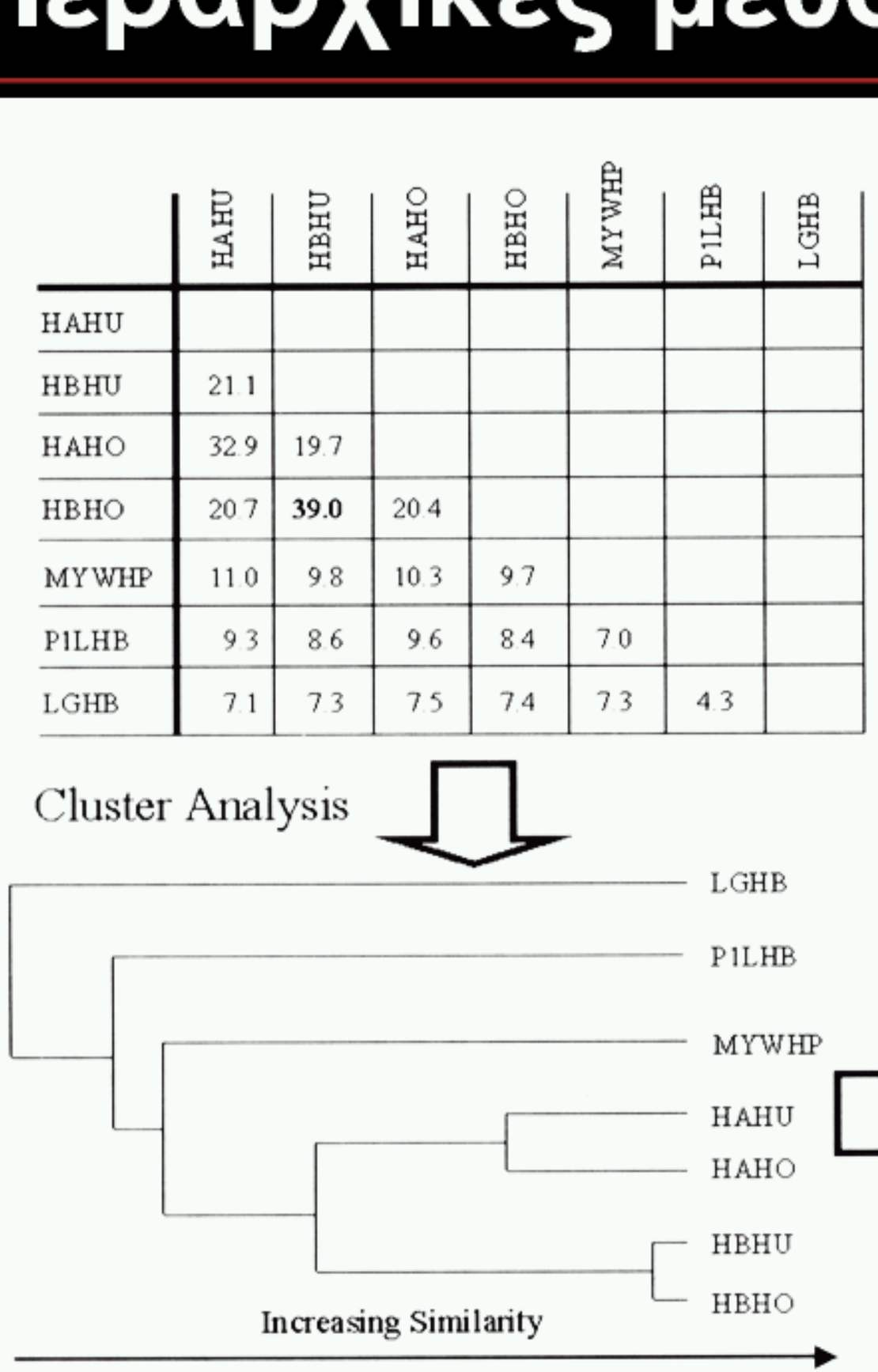
Ιεραρχικές μέθοδοι

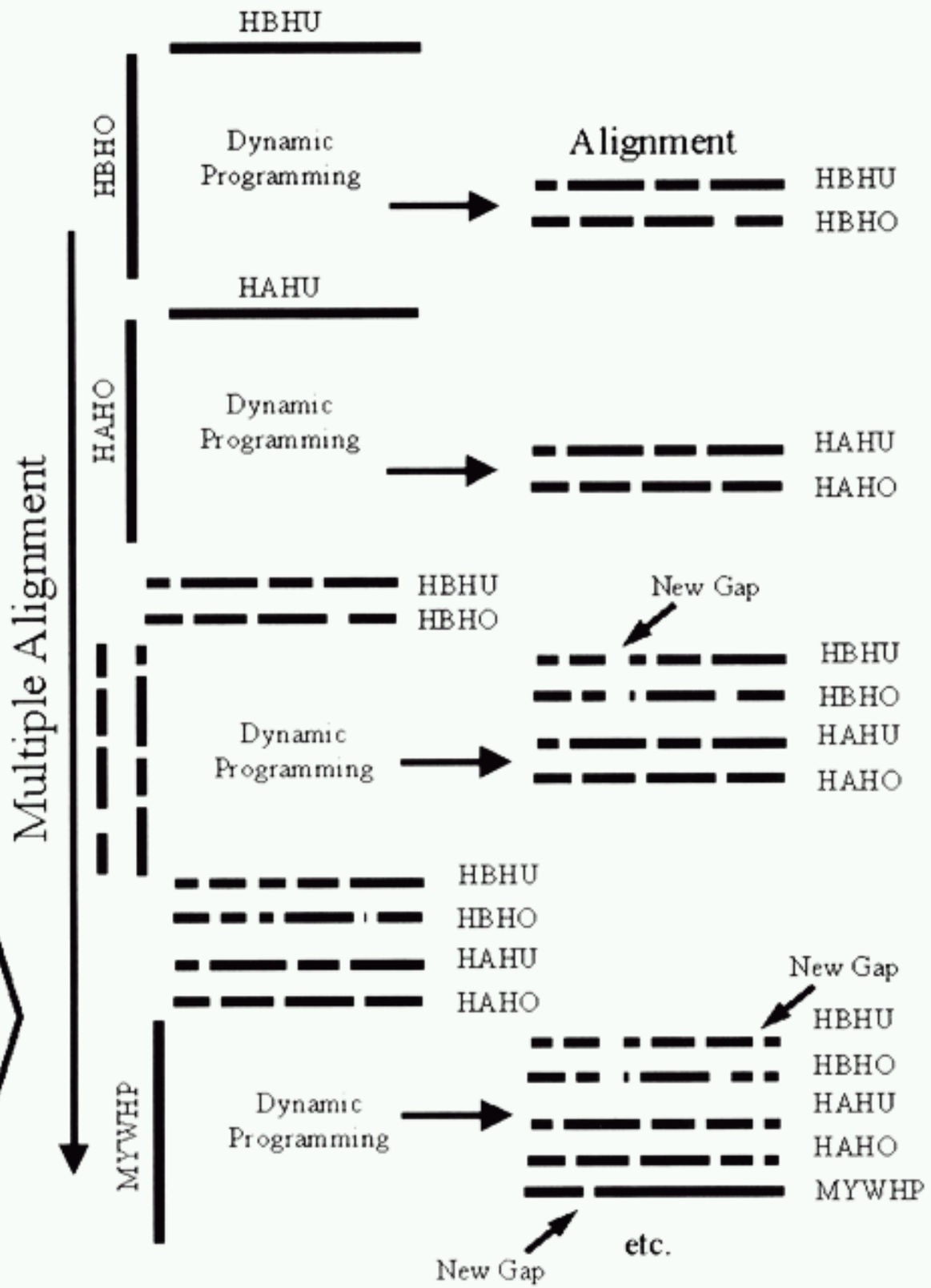
Προφανώς, το απλοϊκό αυτό παράδειγμα δεν μπορεί να καλύψει το εύρος των επιπλέον τεχνικών που χρησιμοποιούνται από τα διάφορα προγράμματα για περαιτέρω βελτίωση των προκυπτόντων στοιχίσεων. Σημειώστε επίσης ότι δεν προσπαθήσαμε καν να συνάγουμε (όπως θα έπρεπε) ένα δενδρόγραμμα από τις ανά ζεύγη στοιχίσεις. Ο λόγος είναι ότι η φυλογενετική ανάλυση αλληλουχιών θα αναλυθεί εκτενέστερα σε μελλοντική διάλεξη.

Παρ' όλα αυτά, ο βασικός αλγόριθμος για την ιεραρχική στοίχιση πολλών αλληλουχιών είναι αυτός που ήδη αναφέρθηκε :

Ιεραρχικές μέθοδοι

	HAHU	HBHU	HAHO	HBHO	MYWHP	P1LHB	LGHB
HAHU							
HBHU	21.1						
HAHO	32.9	19.7					
HBHO	20.7	**39.0**	20.4				
MYWHP	11.0	9.8	10.3	9.7			
P1LHB	9.3	8.6	9.6	8.4	7.0		
LGHB	7.1	7.3	7.5	7.4	7.3	4.3	





Βελτιώσεις και επεκτάσεις

Ο βασικός αλγόριθμος που περιγράψαμε μπορεί να βελτιωθεί. Μερικές από τις πλέον δημοφιλείς βελτιώσεις είναι :

- Επειδή τυχόν λάθη που θα γίνουν στα αρχικά στάδια της στοίχισης, θα συντηρηθούν και στα επόμενα, αρκετά προγράμματα κάνουν μετά το πέρας της αρχικής στοίχισης, επαναστοίχιση των αλληλουχιών. Υποθέστε για παράδειγμα ότι η βέλτιστη στοίχιση του πρώτου ζεύγους αλληλουχιών ήταν :

```
. . . DEFLMPEF . . .
. . . DEEKSTEF . . .
```

άλλα μετά το τέλος της στοίχισης όλες οι υπόλοιπες αλληλουχίες είχαν το μοτίβο :

Βελτιώσεις και επεκτάσεις

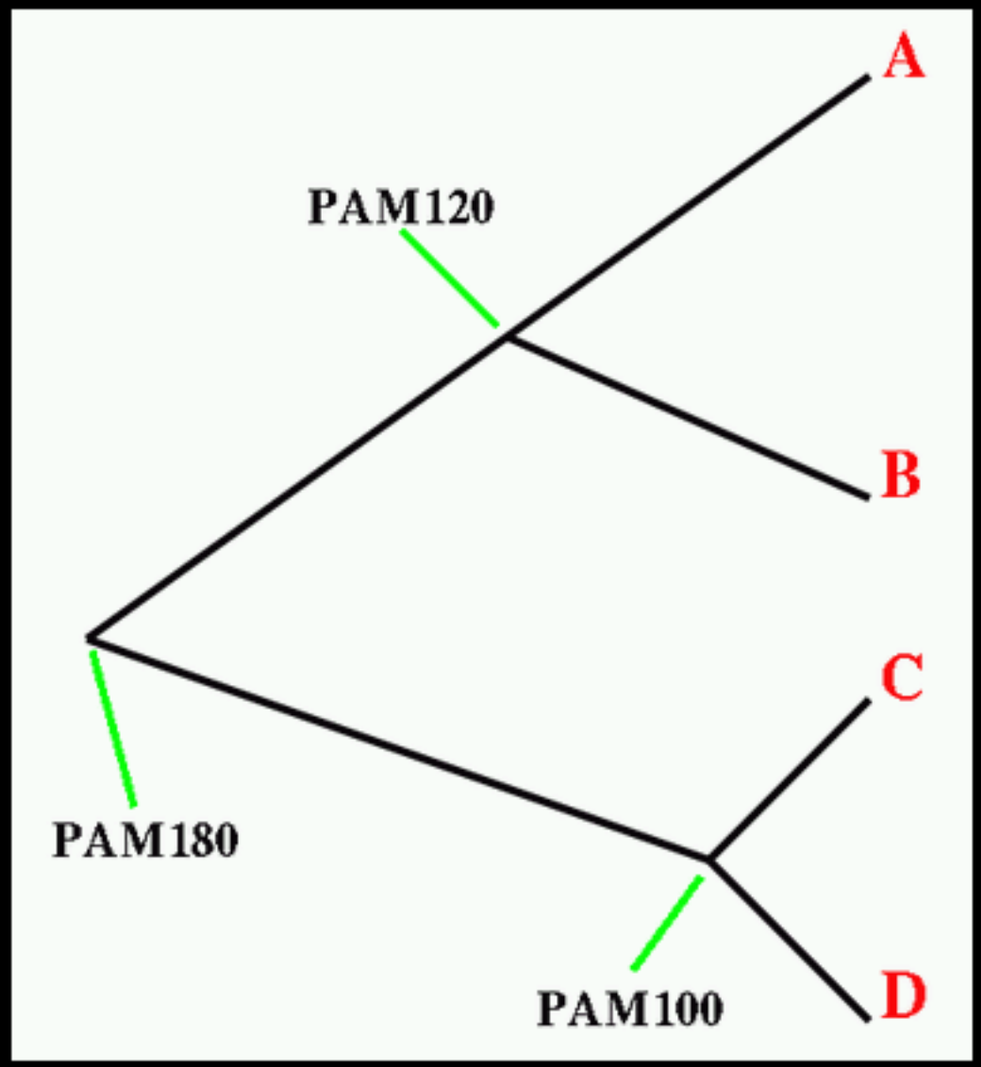
```
. . .  DEEKSTFLMPEF  . . .
. . .  DEEKSTFLMPEF  . . .
. . . . . . . . . . . . . . . . . . . .
```

Προφανώς αυτό που απαιτείται είναι μια διόρθωση της αρχικής στοίχισης ώστε να γίνει :

```
. . .  DE----FLMPEF  . . .
. . .  DEEKST----EF  . . .
. . .  DEEKSTFLMPEF  . . .
. . .  DEEKSTFLMPEF  . . .
. . . . . . . . . . . . . . . . . . . .
```

Βελτιώσεις και επεκτάσεις

Μια άλλη συνηθισμένη βελτίωση είναι η χρήση διαφορετικών πινάκων βαθμολόγησης σε διαφορετικά στάδια ανάπτυξης της στοίχισης. Αυτή η βελτίωση έχει προφανές βιολογικό περιεχόμενο :



Βελτιώσεις και επεκτάσεις

Εάν η δευτεροταγής δομή μίας εκ των αλληλουχιών είναι γνωστή, τότε μπορεί να υλοποιηθεί μία από τις βασικότερες βελτιώσεις (η οποία μάλιστα μπορεί να μεταφερθεί αυτούσια και στους αλγόριθμους στοίχισης δύο αλληλουχιών). Η βελτίωση αφορά την χρήση διαφορετικών τιμών για το gap penalty ανάλογα με το εάν το κενό πρόκειται να εισαχθεί σε μια περιοχή της αλληλουχίας που αντιστοιχεί σε ένα στοιχείο δευτεροταγούς δομής (α,β) ή όχι (στροφές και random coil). Η διόρθωση αυτή έχει προφανές βιολογικό περιεχόμενο (indels στο μέσο μιας στροφής απορροφούνται πιο εύκολα απ' ότι indels στο μέσο ενός στοιχείου δευτεροταγούς δομής).

Βελτιώσεις και επεκτάσεις

Μια άλλη τεχνική που αφορά διαφορική εφαρμογή των gap penalties, στηρίζεται στις παρακάτω υποθέσεις :

- Οι περιοχές των πλέον όμοιων αλληλουχιών οι οποίες συσσωρεύουν κενά, είναι και οι περιοχές στις οποίες το gap penalty για τις υπόλοιπες (λιγότερο συγγενείς αλληλουχίες) θα πρέπει να έχει μικρότερες τιμές.
- Οι περιοχές των πλέον όμοιων αλληλουχιών οι οποίες δείχνουν τη μεγαλύτερη ποικιλότητα, είναι επίσης περιοχές για τις οποίες το κόστος εισαγωγής κενών (για τις υπόλοιπες, λιγότερο συγγενείς αλληλουχίες) θα πρέπει να μειωθεί.

Παράδειγμα προγράμματος : ClustalW

```
***********************************************************
******** CLUSTAL W (1.82) Multiple Sequence Alignments  ******
***********************************************************

     1. Sequence Input From Disc
     2. Multiple Alignments
     3. Profile / Structure Alignments
     4. Phylogenetic trees

     S. Execute a system command
     H. HELP
     X. EXIT (leave program)


Your choice:
```

Βάσεις δεδομένων

Υπάρχει μια πληθώρα βάσεων δεδομένων που περιέχουν στοιχίσεις πολλών αλληλουχιών. Ο στόχος αυτών των βάσεων είναι να συγκεντρώσουν και να ομαδοποιήσουν τις υπάρχουσες πρωτοταγείς αλληλουχίες σε οικογένειες. Παρουσίαση της δομής των καταχωρήσεων αυτών των βάσεων μία-προς-μία είναι άσκοπη. Αυτό που αξίζει να αναφερθεί είναι η διάκριση ανάμεσα σε βάσεις που βασίζονται σε αυτόματες στοιχίσεις και σε βάσεις στις οποίες οι στοιχίσεις ελέγχονται από τους φροντιστές της βάσης.

Βάσεις δεδομένων : Pfam

```
A1AA_HUMAN  SLKYP....AI..MTER.KAA..AILALL.WVV.AL.VVSVGP.LLG...WKEPV..PPDE....RF
A1AA_RAT    SLKYP....AI..MTER.KAA..AILALL.WAV.AL.VVSVGP.LLG...WKEPV..PPDE....RF
A1AB_CANFA  SLQYP....TL..VTRR.KAI..LALLGV.WVL.ST.VISIGP.LLG...WKEPA..PNDD....KE
A1AB_HUMAN  SLQYP....TL..VTRR.KAI..LALLSV.WVL.ST.VISIGP.LLG...WKEPA..PNDD....KE
A1AB_MESAU  SLQYP....TL..VTRR.KAI..LALLSV.WVL.ST.VISIGP.LLG...WKEPA..PNDD....KE
A1AB_RAT    SLQYP....TL..VTRR.KAI..LALLSV.WVL.ST.VISIGP.LLG...WKEPA..PNDD....KE
A1AC_BOVIN  PLRYP....TI..VTQK.RGL..MALLCV.WAL.SL.VISIGP.LFG...WRQPA..PEDE.....T
A1AC_HUMAN  PLRYP....TI..VTQR.RGL..MALLCV.WAL.SL.VISIGP.LFG...WRQPA..PEDE.....T
A1AC_RAT    PLRYP....TI..VTQR.RGV..RALLCV.WVL.SL.VISIGP.LFG...WRQPA..PEDE.....T
OAR_DROME   PINYA....QK..RTVG.RVL..LLISGV.WLL.SL.LISSPP.LIG...W.NDW..PDEFT..SAT
D1DR_CARAU  PFRYE....RK..MTPR.VAF..VMISGA.WTL.SV.LISFIPVQLK...WHKAQ..PIGFL..EVN
D1DR_FUGRU  PFRYE....RK..MTPK.VAC..LMISVA.WTL.SV.LISFIPVQLN...WHKAQ..TASYVELNGT
DADR_DIDMA  PFRYE....RK..MTPK.AAF..ILISVA.WTL.SV.LISFIPVQLN...WHKARPLSSPDG..NVS
....        ....

A1AA_HUMAN  ..............CGI..TE.....................EAG...YA.....VF......SS
A1AA_RAT    ..............CGI..TE.....................EVG...YA.....IF......SS
A1AB_CANFA  ..............CGV..TE.....................EPF...YA.....LF......SS
A1AB_HUMAN  ..............CGV..TE.....................EPF...YA.....LF......SS
A1AB_MESAU  ..............CGV..TE.....................EPF...YA.....LF......SS
A1AB_RAT    ..............CGV..TE.....................EPF...YA.....LF......SS
A1AC_BOVIN  ..............ICQI..NE.....................EPG...YV.....LF......SA
A1AC_HUMAN  ..............ICQI..NE.....................EPG...YV.....LF......SA
A1AC_RAT    ..............ICQI..NE.....................EPG...YV.....LF......SA
OAR_DROME   ..............PCEL..TS.....................QRG...YV.....IY......SS
D1DR_CARAU  ..............ASRR...DLPTDNC........DSSL....NRT...YA.....IS......SS
D1DR_FUGRU  ..............YAGD..LPPDNCD..........SSL....NRT...YA.....IS......SS
DADR_DIDMA  ..............SQDE...TMDNCD..........SSL....SRT...YA.....IS......SS
....        ....

A1AA_HUMAN  V.CSFY...LPMAVIV.VMY.CRV.YVV....ARS..TTRSLEA.GVKR
A1AA_RAT    V.CSFY...LPMAVIV.VMY.CRV.YVV....ARS..TTRSL.....EA
A1AB_CANFA  L.GSFY...IPLAVIL.VMY.CRV.YIV....AKR..TTKNL.....EA
A1AB_HUMAN  L.GSFY...IPLAVIL.VMY.CRV.YIV....AKR..TTKNL.....EA
A1AB_MESAU  L.GSFY...IPLAVIL.VMY.CRV.YIV....AKR..TTKNL.....EA
A1AB_RAT    L.GSFY...IPLAVIL.VMY.CRV.YIV....AKR..TTKNL.....EA
A1AC_BOVIN  L.GSFY...VPLTIIL.VMY.CRV.YVV....AKR..ESRGLKS.GLKT
A1AC_HUMAN  L.GSFY...LPLAIIL.VMY.CRV.YVV....AKR..ESRGLKS.GLKT
A1AC_RAT    L.GSFY...VPLAIIL.VMY.CRV.YVV....AKR..ESRGLKS.GLKT
OAR_DROME   L.GSFF...IPLAIMT.IVY.IEI.FVA....TRR..RLRERA..RANK
D1DR_CARAU  L.ISFY...IPVAIMI.VTY.TQI.YRI....AQK..QIRRIS..ALER
D1DR_FUGRU  L.ISFY...IPVAIMI.VTY.TRI.YRI....AQK..QIRRIS..ALER
DADR_DIDMA  L.ISFY...IPVAIMI.VTY.TRI.YRI....AQK..QIRRIS..ALER
....        ....
```

Βάσεις δεδομένων : PRINTS

```
OAR_DROME   PINYAQ...KRTVGRVLLLISGVWLLSLLISSP.PLIG.WND.....WPDEFTSATP...
D2DR_RAT    PMLYNTR..YSSKRRVTVMIAIVWVLSFTISCP.LLFG.LNN...T.DQNE.........
D3DR_RAT    PVHYQHGTGQSSCRRVALMITAVWVLAFAVSCP.LLFG.FNT...TGDPSI.........
DADR_RAT    PFQYER...KMTPKAAFILISVAWTLSVLISFI.PVQLSWHKAK.PTWPLDGNFTSLEDT
DBDR_RAT    PFRYER...KMTQRVALVMVGLAWTLSILISFI.PVQLNWHRDKAGSQGQEGLLSNGTPW
A1AB_RAT    SLQYPT...LVTRRKAILALLSVWVLSTVISIG.PLLG.WKE......PAPNDDKE....
B1AR_RAT    PFRYQS...LLTRARARALVCTVWAISALVSFL.PILMHWW.....RAESD.EARRCYND
B2AR_HUMAN  PFKYQS...LLTKNKARVIILMVWIVSGLTSFL.PIQMHWY.....RATHQ.EAINCYAN
B2AR_RAT    PFKYQS...LLTKNKARVVILMVWIVSGLTSFL.PIQMHWY.....RATHK.QAIDCYAK
B3AR_RAT    PLRYGT...LVTKRRARAAVVLVWIVSATVSFA.PIMSQWW.....RVGADAEAQECHSN
5HTA_RAT    PIDYVN...KRTPRRAAALISLTWLIGFLISIP.PMLG.WRTPEDRSDPDA.........
5HTD_RAT    ALEYSK...RRTAGHAAAMIAAVWAISICISIP.PLF..WRQ..ATAHEEMSD.......
5HT2_RAT    PIHHSR...FNSRTKAFLKIIAVWTISVGISMPIPVFG.LQDDSKVFKEGS.........
…..         …..

OAR_DROME   ............CELTSQRG.................YVIYSSLGSFFIPLAIMTIVYIEIFVATRRRLRERARANK
D2DR_RAT    ............CIIANPA.................FVVYSSIVSFYVPFIVTLLVYIKIYIVLRKRRKR......
D3DR_RAT    ............CSISNPD.................FVIYSSVVSFYVPFGVTVLVYARIYIVLRQRQRK......
DADR_RAT    ED........DNCDTRLSRT................YAISSSLISFYIPVAIMIVTYTSIYRIAQKQIRR......
DBDR_RAT    EEGWELEGRTENCDSSLNRT................YAISSSLISFYIPVAIMIVTYTRIYRIAQVQIRR......
A1AB_RAT    ............CGVTEEPF.................CALFCSLGSFYIPLAVILVMYCRVYIVAKRTTKN......
B1AR_RAT    PK..........CCDFVTNRA................YAIASSVVSFYVPLCIMAFVYLRVFREAQKQVKK......
B2AR_HUMAN  ET..........CCDFFTNQA................YAIASSIVSFYVPLVIMVFVYSRVFQEAKRQLQK......
B2AR_RAT    ET..........CCDFFTNQA................YAIASSIVSFYVPLVVMVFVYSRVFQVAKRQLQK......
B3AR_RAT    PR..........CCSFASNMP................YALLSSSVSFYLPLLVMLFVYARVFVVAKRQRRF......
5HTA_RAT    ............CTISKDHG.................YTIYSTFGAFYIPLLLMLVLYGRIFRAARFRIRK......
5HTD_RAT    ............CLVNTSQIS................YTIYSTCGAFYIPSILLIILYGRIYVAARSRILN......
5HT2_RAT    ............CLLADDN.................FVLIGSFVAFFIPLTIMVITYFLTIKSLQKEATL......
…..         …..
```

Βάσεις δεδομένων

Η σύγκριση αυτή δείχνει ότι καθώς η ομοιότητα των αλληλουχιών μειώνεται, η βιολογική σημασία των στοιχίσεων που προκύπτουν από τις αυτόματες μεθόδους φαίνεται επίσης να μειώνεται (συνήθως λόγω υπερβολών στη χρήση των κενών).

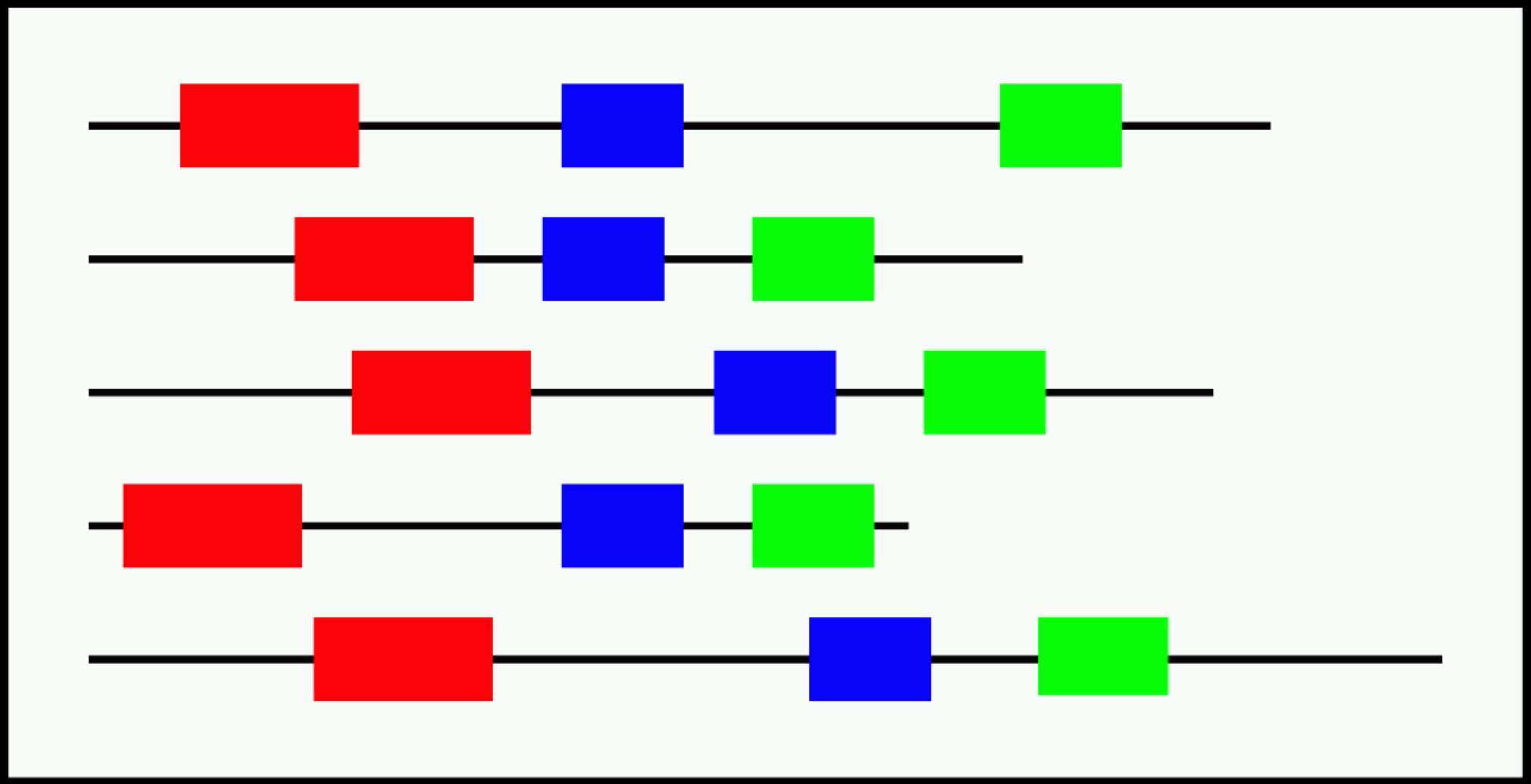
Έτσι, καθώς η ομοιότητα των αλληλουχιών μειώνεται, αυξάνει η ανάγκη ανθρώπινης παρέμβασης (με χρήση κάποιου multiple sequence alignment editor).

Για αλληλουχίες σχετικά υψηλής ομοιότητας (π.χ. Z>6), αλγόριθμοι και άνθρωποι δίνουν συγκρίσιμα αποτελέσματα (παρόμοιες στοιχίσεις).

Άλλες μέθοδοι

Είναι τέτοια η σημασία της στοίχισης αλληλουχιών ώστε να μην είναι λίγοι οι δημοσιευμένοι αλγόριθμοι οι οποίοι ξεφεύγουν από το μοτίβο των ιεραρχικών μεθόδων. Για παράδειγμα, υπάρχουν αλγόριθμοι οι οποίοι για να μειώσουν το μέγεθος του προβλήματος καταφεύγουν στην ιδέα των λεγόμενων αγκυροβολίων (anchors). Τα αγκυροβόλια είναι υπακολουθίες οι οποίες είναι απόλυτα συντηρημένες σε όλες τις αλληλουχίες. Με το να απαιτείται (από τον αλγόριθμο) η στοίχιση των αγκυροβολίων, ουσιαστικά μειώνεται το μέγεθος των αλληλουχιών που πρέπει να στοιχιθούν.

Άλλες μέθοδοι



Το προφανές πρόβλημα με τη μέθοδο είναι ότι καθώς η εξελικτική απόσταση μεταξύ των αλληλουχιών αυξάνει, η πιθανότητα εξεύρεσης αγκυροβολίων μειώνεται.

Εφαρμογές στην έρευνα βάσεων δεδομένων

Μια εμφανής χρήση της ύπαρξης μίας στοίχισης πολλών αλληλουχιών είναι η δυνατότητα να ερευνηθούν οι βάσεις δεδομένων για συγγενείς αλληλουχίες χρησιμοποιώντας όχι μια αλληλουχία, αλλά μία ολόκληρη στοίχιση. Η χρήση της στοίχισης αναμένεται να βελτιώσει το λόγο σήματος προς θόρυβο για την έρευνα λόγω του υψηλότερου πληροφοριακού περιεχομένου της στοίχισης (π.χ. πληροφορία για το ποια αμινοξέα είναι αποδεκτά σε κάποια θέση της αλληλουχίας). Μια τέτοιου τύπου έρευνα είναι ανάλογη με την έρευνα των βάσεων με ένα μοτίβο.

PSI-BLAST

Ο αλγόριθμος που χρησιμοποιείται από το PSI-BLAST για την έρευνα των βάσεων δεδομένων είναι άξιος ξεχωριστής μνείας. Το Position-Specific Iterated BLAST αντιπροσωπεύει ένα υβρίδιο ανάμεσα στις ανά ζεύγη μεθόδους στοίχισης και τις στοιχίσεις πολλών αλληλουχιών. Η κεντρική ιδέα είναι η εξής : μετά από μία αρχική έρευνα των βάσεων δεδομένων (με μία αλληλουχία-στόχο), όσες νέες αλληλουχίες δείχνουν αρκετή ομοιότητα προς την αλληλουχία-στόχο χρησιμοποιούνται για τη δημιουργία ενός μοτίβου (μέσω της στοίχισης με την αρχική αλληλουχία). Το μοτίβο αυτό χρησιμοποιείται εκ νέου για την επανεξέ-ταση των βάσεων δεδομένων μέχρις συγκλίσεως.

PSI-BLAST

Το αποτέλεσμα (από τη μεριά του τελικού χρήστη) είναι ότι μπορεί να πραγματοποιηθεί μια έρευνα των βάσεων δεδομένων με την ευαισθησία που περιμένουμε από τη χρήση μοτίβων, αλλά χωρίς τη χρονοβόρα και απαιτητική διαδικασία της δημιουργίας τους. Το πρόβλημα είναι ότι το όριο για την εισαγωγή μίας νέας αλληλουχίας στο 'μοτίβο' θα πρέπει να είναι αρκετά αυστηρό ώστε να εξασφαλίζει ότι δεν θα υπεισέλθουν ψευδή θετικά στην διαδικασία αναζήτησης. Αυτό είναι ιδιαίτερα σημαντικό για την περίπτωση που η αλληλουχία-στόχος περιέχει LCR περιοχές.

Σύνοψη

Τα βασικά στάδια για τη δημιουργία μίας στοίχισης πολλών αλληλουχιών είναι :

- Εύρεση των αλληλουχιών προς στοίχιση μέσω έρευνας των βάσεων δεδομένων (προγράμματα : FASTA, BLAST, PSI-BLAST).
- Εύρεση των περιοχών των αλληλουχιών που θα χρησιμοποιηθούν στη στοίχιση : τα περισσότερα προγράμματα για στοίχιση πολλών αλληλουχιών δεν αποδίδουν ιδιαίτερα καλά εάν οι αλληλουχίες διαφέρουν σημαντικά στο μήκος τους. Σε τέτοιες περιπτώσεις είναι απαραίτητο να γίνει προ-επεξεργασία των αλληλουχιών για να απομονωθούν οι προς στοίχιση περιοχές (προγράμματα : editor).

Σύνοψη

- Πριν τη στοίχιση όλων των αλληλουχιών, είναι χρήσιμο να υπολογιστεί μια στοίχιση των στενά μόνο συσχετισμένων αλληλουχιών (π.χ. με τιμή E < 0.1). Αυτό θα δώσει μία μάλλον ακριβή στοίχιση η οποία μπορεί να χρησιμοποιηθεί στο επόμενο βήμα ως μέτρο σύγκρισης (για το κατά πόσο η προσθήκη των υπολοίπων αλληλουχιών επαναλαμβάνει και συμφωνεί με τις παρατηρήσεις από τις στενά συσχετιζόμενες αλληλουχίες).
- Πολλαπλή στοίχιση αλληλουχιών. Εάν υπάρχει υπολογιστικός χρόνος διαθέσιμος, αξίζει να γίνει εξέταση της επίδρασης των gap penalties στη στοίχιση.

Σύνοψη

- Εξέταση της στοίχισης για τυχόν προβλήματα με έμφαση σε περιοχές κατακερματισμένες από κενά. Η εξέταση απλοποιείται με τη χρήση προγραμμάτων τα οποία χρωματίζουν τα κατάλοιπα ανάλογα με τις φυσικοχημικές τους ιδιότητες (ALSCRIPT, AMAS, JalView).
- Αφαίρεση των αλληλουχιών που φαίνεται να θίγουν την ποιότητα της στοίχισης και επαναστοίχιση των υπολοίπων αλληλουχιών.
- Μετά την ταυτοποίηση των σημαντικών καταλοίπων και μοτίβων, προσπάθεια επαναστοίχισης των αλληλουχιών που είχαν αφαιρεθεί.