Κεφάλαιο 4
Πολλαπλή Στοίχιση Ακολουθιών

Σύνοψη
Η πολλαπλή στοίχιση είναι μια διαδικασία με κεντρική σημασία στη σύγχρονη βιοπληροφορική. Πολλαπλές στοιχίσεις χρησιμοποιούνται για να εντοπιστούν τα συντηρημένα τμήματα σε μια ομάδα πρωτεϊνικών ακολουθιών και για να χαρακτηριστεί η αντίστοιχη οικογένεια, αλλά και για άλλες αναλύσεις, όπως η εκτίμηση φυλογενετικών σχέσεων και η υποβοήθηση της απόδοσης προγνωστικών αλγορίθμων. Το βασικό πρόβλημα της πολλαπλής στοίχισης είναι ότι δεν υπάρχει εύκολος τρόπος να βρεθεί η μαθηματικά βέλτιστη λύση στο πρόβλημα, όπως έγινε στην περίπτωση της κατά ζεύγη στοίχισης. Στο κεφάλαιο αυτό θα μελετήσουμε τους κύριους αλγόριθμους πολλαπλής στοίχισης και τις αντίστοιχες υλοποιήσεις. Θα δούμε επίσης πώς αξιολογείται μια μέθοδος πολλαπλής στοίχισης, ποια εργαλεία υπάρχουν για την οπτικοποίηση και την επεξεργασία της, και τέλος, θα δούμε πρακτικές συμβουλές για μια καλή πολλαπλή στοίχιση.

Προαπαιτούμενη γνώση
Στο κεφάλαιο αυτό απαραίτητη είναι η γνώση των εννοιών του Κεφαλαίου 3 (στοίχιση ακολουθιών).

4. Εισαγωγή

Αφού μελετήσαμε αναλυτικά την περίπτωση της στοίχισης δύο βιολογικών ακολουθιών, είναι εύλογο, ότι το επόμενο λογικό βήμα θα είναι η προσπάθεια ταυτόχρονης μελέτης περισσότερων από 2 ακολουθιών. Αυτό είναι το αντικείμενο της πολλαπλής στοίχισης το οποίο θα μελετήσουμε σε αυτό το κεφάλαιο. Το θέμα της πολλαπλής στοίχισης, είναι επίσης πολύ σημαντικό στη σύγχρονη υπολογιστική βιολογία και βιοπληροφορική. Οι χρήσεις μιας πολλαπλής στοίχισης, είναι πολλές, διαπερνούν όλο το φάσμα της υπολογιστικής ανάλυσης βιολογικών ακολουθιών, και μπορούμε να τις διακρίνουμε σε τρεις κατηγορίες.

Η προφανής χρήση μιας πολλαπλής στοίχισης αναφέρεται στην ταυτόχρονη μελέτη μιας ομάδας σχετιζόμενων ακολουθιών (συνήθως πρωτεϊνών) και στην προσπάθεια εύρεσης των κοινών χαρακτηριστικών τους. Αυτό, οδηγεί στο χαρακτηρισμό μιας "οικογένειας" πρωτεϊνών και στην αναγνώριση των περιοχών που είναι συντηρημένες. Με τη σειρά του αυτό, μπορεί να οδηγήσει σε χρήσιμες πληροφορίες για διάφορα δομικά ή λειτουργικά χαρακτηριστικά όλων των πρωτεϊνών της οικογένειας (π.χ. συντηρημένα στοιχεία δευτεροταγούς δομής, συντηρημένα κατάλοιπα τα οποία μπορεί να χαρακτηρίζουν το ενεργό κέντρο ενός ενζύμου). Η λογική συνέχεια όλων αυτών των διεργασιών, είναι να κατασκευαστεί με κάποιον μαθηματικό τρόπο, ένα μοντέλο που θα περιγράφει ολόκληρη την πολλαπλή στοίχιση και θα μπορεί να χρησιμοποιηθεί σε μια αναζήτηση σε βάση δεδομένων για ακολουθίες που ταιριάζουν με το μοντέλο πλέον, και όχι με μια συγκεκριμένη ακολουθία. Τέτοια παραδείγματα, είναι η κατασκευή μοντέλων κανονικών προτύπων ή μοτίβων (patterns), προφίλ αλλά και προφίλ Hidden Markov Models τα οποία θα περιγράψουμε σε επόμενα κεφάλαια. Είδαμε ήδη στο κεφάλαιο 2 ότι υπάρχουν ήδη μεγάλες βάσεις δεδομένων οι οποίες περιέχουν κατηγοριοποιήσεις των πρωτεϊνών σε οικογένειες, με τη χρήση τέτοιων μεθόδων (PROSITE, PFAM κ.α.).

Μια δεύτερη, πολύ σημαντική χρήση των πολλαπλών στοιχίσεων προκύπτει στην περίπτωση μελέτης των φυλογενετικών σχέσεων των βιολογικών ακολουθιών, και κατ' επέκταση των οργανισμών προέλευσής τους. Καθώς θεωρούμε ότι οι ακολουθίες έχουν όλες δημιουργηθεί μέσω της διαδικασίας της εξέλιξης από μεταλλάξεις παλαιότερων μορφών, είναι αναμενόμενο ότι οι ομοιότητες και οι διαφορές μιας ομάδας ακολουθιών, μπορούν να ανακατασκευάσουν ένα εξελικτικό δέντρο, το οποίο θα δείχνει τη σειρά με την οποία οι ακολουθίες αυτές εμφάνισαν απόκλιση από τον κοινό πρόγονο. Η διαδικασία αυτή, είναι πολύ σύνθετη και μπορεί να πραγματοποιηθεί με πολλούς διαφορετικούς τρόπους (όπως θα δούμε στο κεφάλαιο 6), αλλά το σημαντικότερο που πρέπει να θυμάται ο αναγνώστης είναι ότι σε κάθε περίπτωση, χρειάζεται μια καλής ποιότητας πολλαπλή στοίχιση σαν σημείο εκκίνησης.

Τέλος, μια πολύ σημαντική χρήση των πολλαπλών στοιχίσεων σχετίζεται με την υποβοήθηση (και μάλιστα σε μεγάλο βαθμό) των αλγορίθμων πρόγνωσης της δομής των πρωτεϊνών. Καθώς είναι γνωστό ότι η δομή συντηρείται περισσότερο από την ακολουθία, μια συντηρημένη περιοχή όπως αποτυπώνεται σε μια πολλαπλή στοίχιση μπορεί να προσφέρει μεγάλη βοήθεια στην προσπάθεια πρόγνωσης. Αυτό διαφέρει από την απλή αναγνώριση μοτίβων και τον εντοπισμό συντηρημένων περιοχών, τα οποία αναφέραμε πριν, αλλά επεκτείνεται και σε αυτοματοποιημένες χρήσεις των πολλαπλών στοιχίσεων, ως δεδομένα εισόδου σε αλγορίθμους πρόγνωσης της δομής, ή άλλων χαρακτηριστικών των πρωτεϊνών. Το θέμα θα εξεταστεί σε

επόμενο κεφάλαιο, καθώς υπάρχουν διαφορετικοί τρόποι με τους οποίους μπορεί να γίνει αυτή η χρήση. Για παράδειγμα, μπορεί η μέθοδος πρόγνωσης να εφαρμόζεται διαδοχικά σε όλες τις πρωτεΐνες της οικογένειας και στο τέλος να γίνεται "προβολή" των αποτελεσμάτων πάνω στην αρχική ακολουθία, ή, εναλλακτικά, οι μέθοδοι πρόγνωσης θα μπορούσαν να τροποποιηθούν έτσι ώστα να χρησιμοποιούν κατευθείαν κάποιο παράγωγο της πολλαπλής στοίχισης, όπως για παράδειγμα ένα profile.

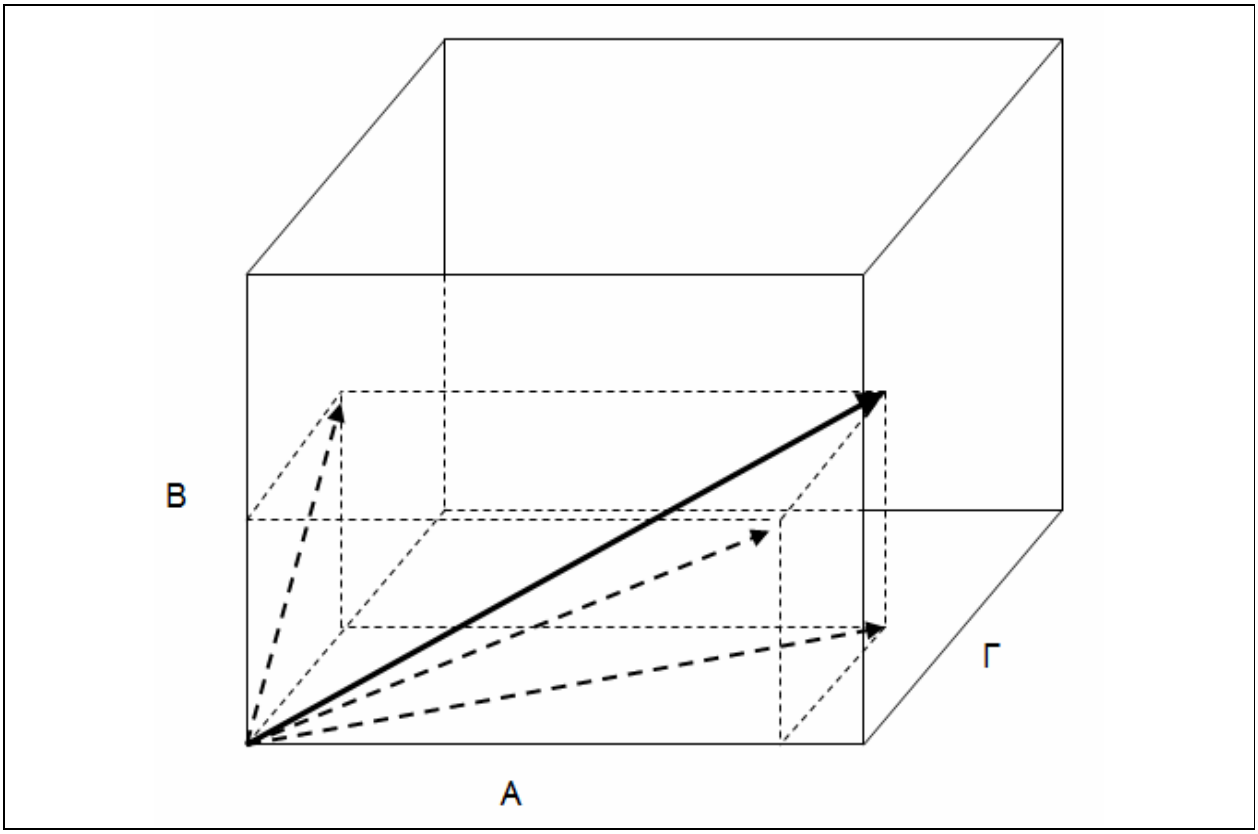
Στις επόμενες παραγράφους, θα παρουσιάσουμε τα βασικά θεωρητικά θέματα που εμπλέκονται στην πολλαπλή στοίχιση και τους βασικούς τύπους αλγορίθμων. Θα παρουσιαστεί επίσης το διαθέσιμο λογισμικό για το σκοπό αυτό, αλλά και οι τρόποι αξιολόγησης και οπτικοποίησης μιας πολλαπλής στοίχισης.

4.1. Πολλαπλή Στοίχιση – Δυναμικός Προγραμματισμός

Για να μελετήσουμε την πολλαπλή στοίχιση, είναι απαραίτητο πλέον να μελετήσουμε ταυτόχρονα περισσότερες από μία ακολουθίες. Έστω ότι έχουμε *r* ακολουθίες:

$$\begin{aligned} \mathrm{x}_1 &= x_{11}x_{12}\ldots x_{1n} \\ \mathrm{x}_2 &= x_{21}x_{22}\ldots x_{2n} \\ &\ldots\ldots\ldots \\ \mathrm{x}_\mathrm{r} &= x_{r1}x_{r2}\ldots x_{rn} \end{aligned} \tag{4.1}$$

ο πιο φυσικός τρόπος που μπορούμε να σκεφτούμε είναι να επεκτείνουμε τους αλγόριθμους δυναμικού προγραμματισμού του κεφαλαίου 3, στις *r* διαστάσεις. Όπως είχαμε δει στο κεφάλαιο 3, το πρόβλημα της στοίχισης δύο ακολουθιών ανάγεται στην εύρεση του βέλτιστου μονοπατιού στον πίνακα που αντιστοιχεί στο διάγραμμα σημείων. Κατ' αναλογία, όταν έχουμε τρεις ακολουθίες, η πολλαπλή στοίχιση αντιστοιχεί στην εύρεση του βέλτιστου μονοπατιού στον τρισδιάστατο πίνακα του οποίου οι έδρες είναι οι πίνακες που αντιστοιχούν στις κατά ζεύγη στοιχίσεις των ακολουθιών.



Εικόνα 4.1 *Σχηματική αναπαράσταση του πίνακα δυναμικού προγραμματισμού, για μια πολλαπλή στοίχιση 3 ακολουθιών Σε περίπτωση περισσότερων ακολουθιών η οπτικοποίηση γίνεται δυσκολότερη καθώς απαιτούνται περισσότερες διαστάσεις.*

Για να ξεκινήσουμε την πολλαπλή στοίχιση είναι αναγκαίο να ορίσουμε μια συνάρτηση για το score:

$$S(m) = G + \sum_i S(m_i) \tag{4.2}$$

όπου m_i είναι η στήλη i της πολλαπλής στοίχισης m, $S(m_i)$ το score της και G είναι μια συνάρτηση (απλή ή σύνθετη) για τα κενά. Για απλότητα, το κενό μπορεί να εισαχθεί και σαν ένα 5ο σύμβολο στις ακολουθίες (-), αν και στην πραγματικότητα αυτό δεν χρησιμοποιείται από τους περισσότερους σύγχρονους αλγόριθμους, γιατί θα ισοδυναμούσε με γραμμική εισαγωγή κενών. Παρόλα αυτά, για λόγους απλότητας, στις επόμενες ενότητες θα χρησιμοποιήσουμε αυτόν τον ορισμό, έτσι ώστε να μπορέσουμε να μελετήσουμε πιο εύκολα τους αλγόριθμους. Έτσι, θα έχουμε:

$$S(m) = \sum_i S(m_i) \tag{4.3}$$

Οι πιθανοί τρόποι να ορίσουμε το πολυδιάστατο score, είναι πολλοί. Ο πρώτος τρόπος τον οποίο θα σκεφτόταν κάποιος, αναλογιζόμενος τους αλγορίθμους του προηγούμενου κεφαλαίου είναι να ορίσει ένα log-odds για τις r διαστάσεις:

$$S\left(m\right) = \sum_i S(m_i) = \sum_i \log\left(\frac{p_{x_{1i}x_{2i}\ldots x_{ri}}}{q_{x_{1i}}q_{x_{2i}}\ldots q_{x_{ri}}}\right) = \sum_i s(x_{1i}, x_{2i}, \ldots, x_{ri}) \tag{4.4}$$

Πρακτικά, αυτό είναι πολύ δύσκολο, γιατί θα σήμαινε ότι για παράδειγμα η δουλειά που έγινε για τους πίνακες ομοιότητας (PAM, BLOSUM κλπ), θα έπρεπε να έχει επαναληφθεί για κάθε πιθανό αριθμό ακολουθιών για τις οποίες θα επιχειρήσουμε μια πολλαπλή στοίχιση. Με άλλα λόγια, θα έπρεπε να υπάρχει προϋπολογισμένος ένας πίνακας για τις στοιχίσεις 3 ακολουθιών, άλλος πίνακας για τις στοιχίσεις 4 ακολουθιών, κ.ο.κ., κάτι που είναι πρακτικά αδύνατο.

Ένας άλλος τρόπος, θα ήταν αν κάναμε χρήση της έννοιας της εντροπίας την οποία συναντήσαμε στο προηγούμενο κεφάλαιο. Αν ονομάσουμε m_i^j το σύμβολο στην i στήλη της j ακολουθίας και $n_b(i)$ τον αριθμό των εμφανίσεων του συμβόλου b στη στήλη i, τότε η συνολική πιθανότητα της στήλης αυτής, θα είναι ίση με:

$$P\left(m_i\right) = \prod_{\forall b \in \Omega} p_b\left(i\right)^{n_b(i)} \tag{4.5}$$

όπου $p_s(i)$ θα είναι η πιθανότητα του συμβόλου s στη στήλη i, οποία θα δίνεται από τη σχέση:

$$p_b\left(i\right) = \frac{n_b\left(i\right)}{\sum_{\forall b' \in \Omega} n_{b'}\left(i\right)} \tag{4.6}$$

Τότε, αν πάρουμε το λογάριθμο, θα έχουμε:

$$S\left(m_i\right) = -\sum_{\forall b \in \Omega} n_b\left(i\right) \log p_b\left(i\right) \tag{4.7}$$

Αυτή η σχέση, είναι ξεκάθαρα ένα μέτρο εντροπίας, όπως το ορίσαμε στο προηγούμενο κεφάλαιο, με τη διαφορά ότι τώρα δεν αφορά ένα παράθυρο κατά μήκος της ακολουθίας, αλλά μία στήλη της πολλαπλής στοίχισης. Παρόλα αυτά, η ερμηνεία του είναι απλή και διαισθητική: μία στήλη η οποία είναι 100% συντηρημένη, θα έχει εντροπία ίση με το 0, αντίθετα, μια στήλη με τελείως τυχαία κατανομή συμβόλων, θα έχει μέγιστη εντροπία. Κατά συνέπεια, ένα καλό score, θα ήταν αυτό το οποίο θα ελαχιστοποιούσε την εντροπία της πολλαπλής στοίχισης σε όλο το μήκος της (ή, εναλλακτικά, αυτό το οποίο θα μεγιστοποιούσε την πληροφορία). Στα παραπάνω, κάναμε σιωπηλά, δύο σημαντικές παραδοχές, οι οποίες είναι απαραίτητο να γίνουν, αλλά και απαραίτητο να διευκρινιστούν. Πρώτον, θεωρήσαμε τις r ακολουθίες ανεξάρτητες, -πράγμα το οποίο μπορεί να μην ισχύει ειδικά αν βρίσκονται εξελικτικά πολύ κοντά (σε αυτό θα επανέλθουμε). Δεύτερον, για να αθροίσουμε συνεισφορές του score σε όλο το μήκος της στοίχισης, θεωρούμε και πάλι ότι οι στήλες είναι ανεξάρτητες μεταξύ τους. Σε κάθε περίπτωση, οι περισσότεροι αλγόριθμοι δεν χρησιμοποιούν το σύστημα του score που περιγράψαμε παραπάνω. Παρόλα αυτά, η εντροπία χρησιμοποιείται για να αξιολογήσει το τελικό αποτέλεσμα μιας πολλαπλής στοίχισης ή για να συγκριθούν μεταξύ τους οι διάφοροι αλγόριθμοι όταν εφαρμοστούν στα ίδια δεδομένα.

Στους περισσότερους αλγόριθμους πολλαπλής στοίχισης, χρησιμοποιείται το λεγόμενο SP (Sum of Pairs) score. Το score αυτό ορίζεται για μία στήλη της στοίχισης ως:

$$SP\left(m_i\right)=\sum_{j<j'} s\left(m_i^j, m_i^{j'}\right) \tag{4.8}$$

Όπου οι τιμές της συνάρητησης *s* δίνονται από κάποιον από τους γνωστούς από την κατά ζεύγη στοίχιση ακολουθιών, αλγορίθμους. Για τη συνολική στοίχιση, θα μπορούσε να γραφτεί και ως εξής:

$$SP\left(m\right)=\sum_{i}\sum_{j<j'} s\left(m_i^j, m_i^{j'}\right)=\sum_{i}\left\{\log\left(\frac{p_{x_{1i}x_{2i}}}{q_{x_i}q_{x_{2i}}}\right)+....+\log\left(\frac{p_{x_{(r-1)i}x_{ri}}}{q_{x_{(r-1)i}}q_{x_{ri}}}\right)\right\} \tag{4.9}$$

καθώς είναι το άθροισμα των scores για όλες τις ανά 2 συγκρίσεις των *r* ακολουθιών. Η Μέθοδος αυτή, είναι πολύ βολική, αλλά έχει το μειονέκτημα ότι δεν έχει καλές μαθηματικές ιδιότητες. Το άθροισμα των ανά δύο score, δεν έχει κάποια φυσική ερμηνεία, και οδηγεί σε κάποια παράδοξα. Για παράδειγμα, υπάρχουν περιπτώσεις, στις οποίες μια εισαγωγή ενός διαφορετικού συμβόλου σε μια κατά τα άλλα τέλεια στοίχιση (π.χ. 10 ή 20 όμοιες ακολουθίες), οδηγεί σε μεγαλύτερη μείωση του score στη στοίχιση με τις περισσότερες, σε σχέση με τη στοίχιση με τις λιγότερες ακολουθίες (Durbin, Eddy, Krogh, & Mithison, 1998). Αυτό είναι αντίθετο με τη διαίσθηση, γιατί θα περιμέναμε η μείωση του score να είναι μικρότερη, λ.χ. στην περίπτωση που έχουμε 19/20 ακολουθίες ίδιες, παρά αν είχαμε 9/10, αλλά εξηγείται αν παρατηρήσουμε ότι η μία εισαγωγή του διαφορετικού συμβόλου θα επηρεάσει περισσότερους όρους στο άθροισμα στην πρώτη περίπτωση.

Παρόλα αυτά, αυτή είναι η μέθοδος που χρησιμοποιούν οι περισσότεροι αλγόριθμοι, κυρίως για την υπολογιστική ευκολία που προσφέρει αλλά και λόγω του ότι μπορεί και ενσωματώνει εύκολα την πληροφορία των πινάκων ομοιότητας οι οποίοι είναι ήδη διαθέσιμοι. Δεν πρέπει να ξεχνάμε, ότι στην περίπτωση των πρωτεϊνών, είναι σχεδόν αδύνατο να βρούμε μια στοίχιση πολλών ακολουθιών (π.χ. >50) οι οποίες να έχουν 100% συντηρημένες παρά μόνο λίγες θέσεις. Τούτο συμβαίνει, γιατί πολλές φορές αμινοξέα με παρόμοιες φυσικοχημικές ιδιότητες αντικαθιστούν κάποια άλλα στην εξέλιξη, χωρίς να επηρεάσουν τη δομή και τη λειτουργία της πρωτεΐνης. Ένα μέτρο σαν την εντροπία, θα «έχανε» αυτή την πληροφορία, η οποία όμως εντοπίζεται με χρήση των πινάκων ομοιότητας. Αξίζει να αναφερθεί, ότι το ίδιο ισχύει ακόμα και στην περίπτωση των περιοχών χαμηλής πολυπλοκότητας που είδαμε στο προηγούμενο κεφάλαιο (για παράδειγμα μπορεί να έχουμε επαναλήψεις παρόμοιων αμινοξέων), και έχουν προταθεί και μέτρα πολυπλοκότητας που λαμβάνουν υπόψη τους πίνακες ομοιότητας αμινοξέων.

Αφού έχουμε δει τα βασικά για το score μιας πολλαπλής στοίχισης, ας δούμε πώς διαμορφώνεται ένας αλγόριθμος δυναμικού προγραμματισμού για το σκοπό αυτό. Αν ονομάσουμε $a_{i1,i2,\ ...,iN}$ το μέγιστο score μιας στοίχισης μέχρι και τις υποακολουθίες που τελειώνουν στο $x_{i1}^1, x_{i2}^2, ..., x_{iN}^N$, τότε μια απευθείας επέκταση των αλγορίθμων του κεφαλαίου 3, δίνει:

$$a_{i1}, a_{i2}, ..., a_{in} = \max_{\Delta_1+...+\Delta_n} \begin{cases} a_{i1-1,i2-1,...,in-1} + S\left(x_{i1}^1, x_{i2}^2, ..., x_{in}^n\right) \\ a_{i1,i2-1,...,in-1} + S\left(-, x_{i2}^2, ..., x_{in}^n\right) \\ ... \\ a_{i1-1,i2-1,...,in} + S\left(x_{i1}^1, x_{i2}^2, ..., -\right) \\ a_{i1,i2,i3-1,...,in-1} + S\left(-, -, ..., x_{in}^n\right) \\ ... \end{cases} \tag{4.10}$$

Σε αυτή την περίπτωση το κενό το αντιμετωπίζουμε όπως είπαμε λόγω ευκολίας, σαν ένα πέμπτο σύμβολο (-). Στη σχέση (4.10) στο δεξί σκέλος επιτρέπονται όλοι οι συνδυασμοί των κενών, εκτός από αυτόν στον οποίο όλες οι θέσεις έχουν κενό (2^N-1 συνολικοί συνδυασμοί). Ένας πιο συμπυκνωμένος τρόπος να γραφτεί ο αλγόριθμος, θα είναι (Durbin, et al., 1998; Waterman, 1995):

$$a_{i1}, a_{i2}, ..., a_{in} = \max_{\Delta_1+...+\Delta_n>0} \left\{a_{i1-\Delta_1,i2-\Delta_2,...,in-\Delta_n} + S\left(\Delta_1 x_{i1}^1, \Delta_2 x_{i2}^2, ..., \Delta_n x_{in}^n\right)\right\} \tag{4.11}$$

όπου Δ είναι στοιχεία μιας συνάρτησης για την οποία ισχύει :

$$\Delta_i x = \begin{cases} (x), \text{ αν } \Delta_i = 1 \\ (-), \text{ αν } \Delta_i = 0 \end{cases} \quad (4.12)$$

Όπως είναι φανερό, ο αλγόριθμος αυτός, αν έχουμε *r* ακολουθίες με *n* νουκλεοτίδια η κάθε μια (ή, αμινοξικά κατάλοιπα αν μιλάμε για πρωτεΐνες), απαιτεί χρόνο της τάξης του $O(n^r 2^r)$ και χώρο στη μνήμη $O(n^r)$. Πρακτικά λοιπόν, ένας τέτοιος αλγόριθμος θα είχε μεγάλη πολυπλοκότητα και θα ήταν ιδιαίτερα αργός, δηλαδή θα χρειαζόταν απαγορευτικό χρόνο ακόμα και για λίγες σχετικά ακολουθίες και κατά συνέπεια θα πρέπει να αναζητηθούν τρόποι να περιοριστούν οι απαιτήσεις αυτές, περιορίζοντας το εύρος της αναζήτησης. Έναν τέτοιο αλγόριθμο πρότειναν οι Carrilo και Lipman (Carrillo & Lipman, 1988). Ο αλγόριθμος αυτός, βρίσκει ένα κάτω φράγμα στο score για κάθε ζεύγος στοιχίσεων μεταξύ των ακολουθιών, και στη συνέχεια, ελέγχει στην πολλαπλή στοίχιση μόνο τις περιοχές αυτές που έχουν score μεγαλύτερο από την τιμή αυτή. Οσο πιο υψηλή τιμή έχει αυτό το φράγμα, τόσο πιο γρήγορος θα είναι ο αλγόριθμος. Πρακτικά, για να βρεθεί αυτή η τιμή θα πρέπει να χρησιμοποιηθεί πρώτα ένας γρήγορος ευριστικός αλγόριθμος πολλαπλής στοίχισης (όπως αυτοί της προοδευτικής πολλαπλής στοίχισης που θα περιγραφούν παρακάτω). Ο αλγόριθμος των Carrilo και Lipman έχει υλοποιηθεί στο γνωστό πρόγραμμα **MSA** (Lipman, Altschul, & Kececioglu, 1989), διαθέσιμο στη διεύθυνση http://xylian.igh.cnrs.fr/msa/msa.html το οποίο όμως, ακόμα και έτσι, πρακτικά είναι ικανό να στοιχίσει μόνες μερικές ακολουθίες πρωτεϊνών.

Όπως είναι φανερό, στην περίπτωση της πολλαπλής στοίχισης, η ανάγκη να αναζητήσουμε ευριστικούς αλγόριθμους είναι ακόμα μεγαλύτερη σε σχέση με την περίπτωση της κατά ζεύγη στοίχισης. Στις επόμενες παραγράφους, θα περιγράψουμε τις βασικές κατηγορίες ευριστικών αλγορίθμων για πολλαπλή στοίχιση, και τις παραλλαγές τους όπως αυτές χρησιμοποιούνται στα σύγχρονα εργαλεία λογισμικού.

4.2. Προοδευτική πολλαπλή στοίχιση

Η πιο γνωστή ευριστική (heuristic) μέθοδος που χρησιμοποιείται για πολλαπλή στοίχιση, είναι η λεγόμενη progressive multiple alignment method (προοδευτική πολλαπλή στοίχιση). Κατά τη μέθοδο αυτή η στοίχιση των ακολουθιών γίνεται προοδευτικά ξεκινώντας από δυο ακολουθίες (συνήθως αυτές με την μεγαλύτερη ομοιότητα), και σταδιακά προστίθενται στην στοίχιση μια-μια, οι υπόλοιπες ακολουθίες. Αν και υπάρχουν πολλές παραλλαγές ήδη από τις αρχές της δεκαετίας του 1980, η γενική μέθοδος της προοδευτικής πολλαπλής στοίχισης όπως διατυπώθηκε από τους Feng και Doolittle το 1987 (Feng & Doolittle, 1987) περιλαμβάνει τα παρακάτω κύρια βήματα:

- Αρχικές κατά ζεύγη στοιχίσεις όλων των ακολουθιών
- Με βάση αυτές τις στοιχίσεις, κατάσκευή πίνακα αποστάσεων και ενός δέντρου οδηγού (guide tree)
- Προοδευτική στοίχιση των πιο όμοιων ακολουθιών μεταξύ τους, μέχρι τέλους

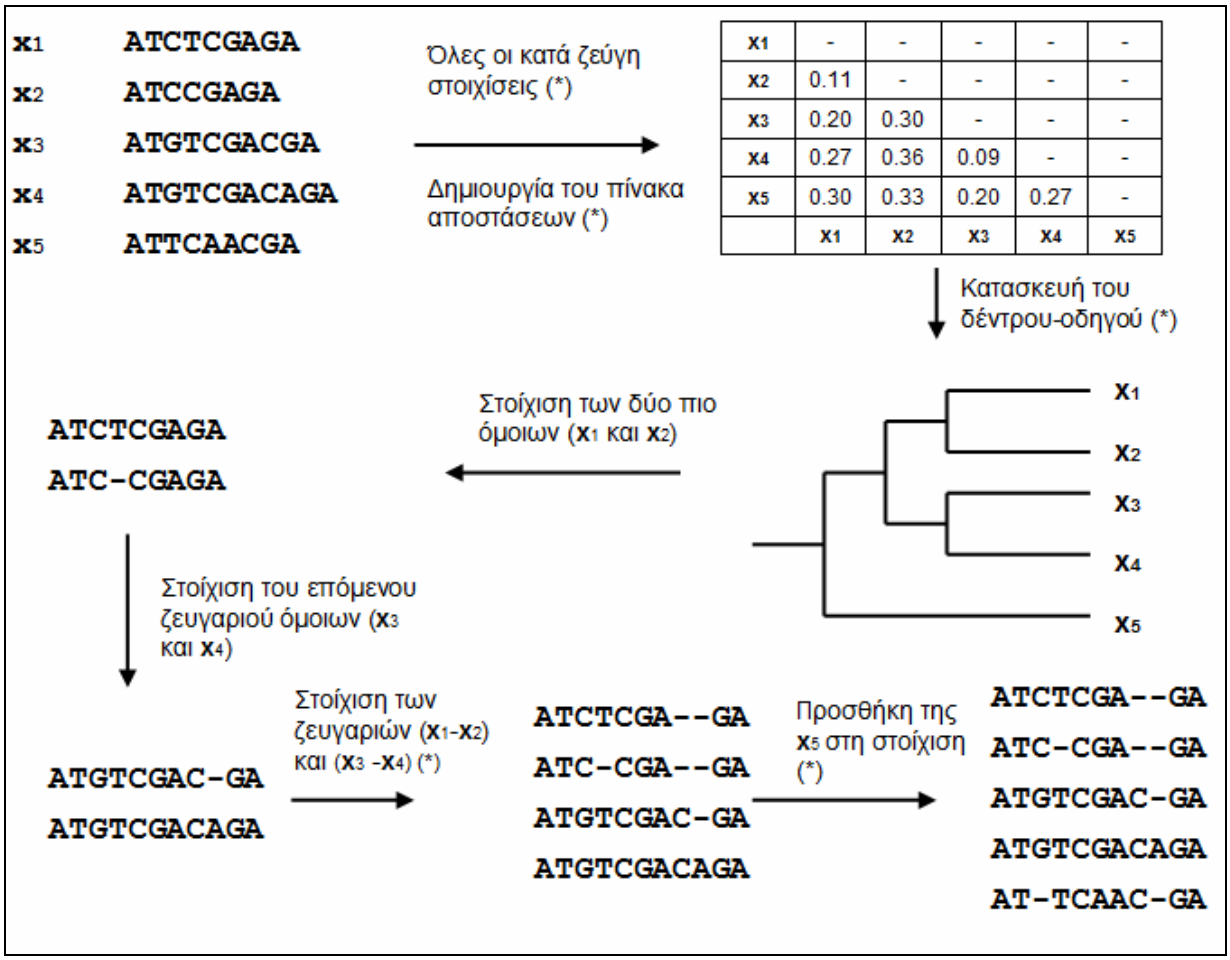
Όπως είναι εμφανές, τα βήματα αυτά, θα μπορούσαν να υλοποιηθούν με διαφορετικούς τρόπους. Για παράδειγμα, οι κατά ζεύγη στοιχίσεις θα μπορούσαν να γίνουν με δυναμικό προγραμματισμό ή με ευριστική μέθοδο (BLAST, FASTA). Ο πίνακας των αποστάσεων θα μπορούσε να οριστεί με τελείως διαφορετικά κριτήρια, ενώ και το δέντρο οδηγός θα μπορούσε να κατασκευαστεί με μια πλειάδα αλγορίθμων ομαδοποίησης (clustering). Τέλος, υπάρχει και το αλγοριθμικό θέμα του πώς θα προχωρήσει η στοίχιση μιας ακολουθίας με μια ήδη υπάρχουσα στοίχιση, ή, μια στοίχισης με μια άλλη στοίχιση. Στις πιο παλιές μεθόδους, υπήρχαν και άλλες πιο βασικές διαφορές, όπως για παράδειγμα η ίδια η ύπαρξη του δέντρου, αλλά εδώ θα μελετήσουμε κυρίως παραλλαγές πάνω σε αυτή τη μέθοδο. Η μέθοδος των Feng και Doolittle (Feng & Doolittle, 1987), ήταν όπως είπαμε μια από τις πρώτες τέτοιες μεθόδους, και έκανε αρχικά τις στοιχίσεις με αλγόριθμο δυναμικού προγραμματισμού (ολικής στοίχισης). Στη συνέχεια, υπολόγιζε τις αποστάσεις, από τα score των στοιχίσεων χρησιμοποιώντας τον τύπο:

$$D = -\log S = \log \frac{S_{obs} - S_{rand}}{S_{\max} - S_{rand}} \quad (4.13)$$

Το S_{obs} είναι το πραγματικό score για τη στοίχιση των δύο ακολουθιών, όπως προέκυψε από τον αλγόριθμο. Το S_{max} είναι το θεωρητικό μέγιστο που θα μπορούσε να προκύψει από τη στοίχιση, αν στοιχίζαμε οποιαδήποτε από τις δύο ακολουθίες με τον εαυτό της, ενώ το S_{rand} είναι το αναμενόμενο score από μια τέτοια στοίχιση ακολουθιών οι οποίες δεν είχαν καμία σχέση μεταξύ τους. Θα μπορούσε να προκύψει με κάποια προσομοίωση όπως περιγράψαμε στο προηγούμενο κεφάλαιο (με shuffling), αλλά οι Feng και Doolittle έδωσαν έναν προσεγγιστικό υπολογισμό. Όπως είναι φανερό, ο τρόπος υπολογισμού του κλάσματος δίνει

περίπου το ποσοστό ομοιότητας των ακολουθιών, οπότε η προσθήκη του –log κάνει το μέτρο περίπου γραμμικό και δίνει μεγαλύτερες τιμές (μεγαλύτερη απόσταση) σε ζευγάρια με μικρή ομοιότητα.

Αφού έχουν υπολογιστεί οι αποστάσεις, ο αλγόριθμος κατασκευάζει ένα δέντρο με τη χρήση του αλγορίθμου των Fitch και Margoliash (Fitch & Margoliash, 1967). Ο αλγόριθμος αυτός είναι από τους πιο γρήγορους αλγόριθμους ομαδοποίησης, και προτάθηκε αρχικά για την κατασκευή φυλογενετικών δέντρων. Γενικά, το δέντρο-οδηγός της προοδευτικής πολλαπλής στοίχισης, έχει πολλά κοινά με τα φυλογενετικά δέντρα τα οποία θα εξετάσουμε στο κεφάλαιο 6, αλλά πρέπει να σημειώσουμε, ότι δεν είναι το ίδιο ένα τέτοιο δέντρο, τουλάχιστον όχι με την αυστηρή έννοια. Ο σκοπός εδώ είναι να παραχθεί γρήγορα μια ομαδοποίηση η οποία θα κατευθύνει τη στοίχιση, και όχι να βρεθεί ο κοινός πρόγονος των ακολουθιών και ο χρόνος απόκλισης της κάθε μίας.

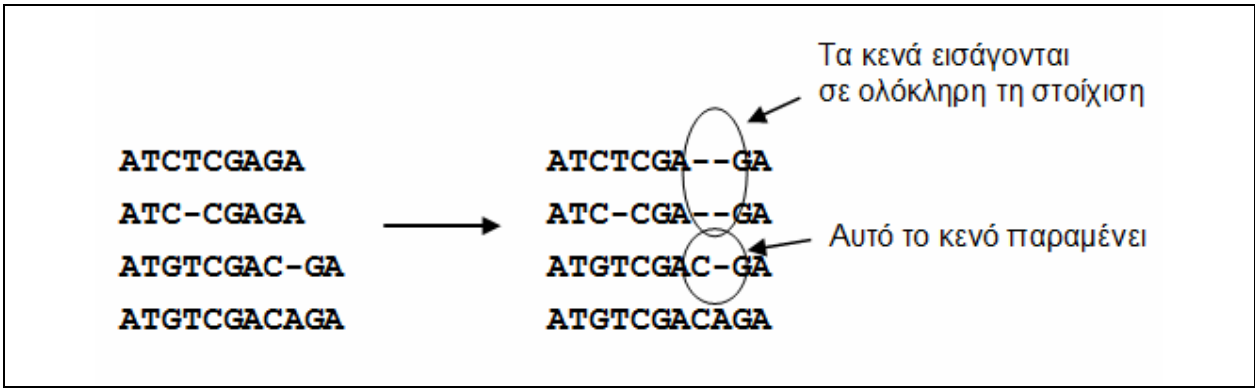


Εικόνα 4.2 *Παράδειγμα προοδευτικής πολλαπλής στοίχισης 5 ακολουθιών (Duret & Abdeddaim, 2000). Με (*) σημειώνονται τα σημεία στα οποία θα μπορούσε να υπάρξει διαφοροποίηση μεταξύ των αλγορίθμων. Όταν σχηματιστεί η στοίχιση των ακολουθιών x_1 και x_2 από τη μια μεριά, και των x_3 και x_4 από την άλλη, στο επόμενο βήμα, το δέντρο υπαγορεύει ότι οι δύο αυτές στοιχίσεις πρέπει να ενωθούν, καθώς οι τέσσερις ακολουθίες που περιέχονται έχουν μεγαλύτερες ομοιότητες μεταξύ τους παρά με την x_5. Στο σημείο αυτό, τη στοίχιση των δύο στοιχίσεων, την κατευθύνει απόλυτα το ζευγάρι με τη μεγαλύτερη ομοιότητα (x_1-x_3). Το ίδιο συμβαίνει και στο επόμενο βήμα, το οποίο καθορίζεται απόλυτα από τη στοίχιση x_3-x_5.*

Τέλος, στο επόμενο βήμα, οι ακολουθίες στοιχίζονται σταδιακά χρησιμοποιώντας την πληροφορία του δέντρου, ξεκινώντας από τις πιο όμοιες. Το βασικό σημείο εδώ, είναι ότι μια ακολουθία (ή μια στοίχιση) προστίθεται σε μια άλλη πολλαπλή στοίχιση, με βάση το ζευγάρι των ακολουθιών που είχε το μεγαλύτερο score (τη μικρότερη απόσταση). Με τον τρόπο αυτό, μια στοίχιση όταν δημιουργηθεί, δεν αλλάζει ξανά, και το μόνο που μπορεί να συμβεί είναι να προστεθούν κενά. Η μέθοδος βέβαια αυτή, έχει και ένα άλλο

μειονέκτημα: τη στοίχιση μιας στοίχισης με μια άλλη στοίχιση, την καθοδηγεί απόλυτα το ζευγάρι το οποίο εμφανίζει τη μέγιστη ομοιότητα. Στο παράδειγμα στην Εικόνα 4.2 βλέπουμε πώς λειτουργεί η μέθοδος για μια στοίχιση 5 ακολουθιών.

Η μέθοδος αυτή, δεν είναι πάντα αποτελεσματική, καθώς το ζευγάρι με τη μεγαλύτερη ομοιότητα μπορεί να προσδώσει συστηματικό σφάλμα (bias) στην πολλαπλή στοίχιση, και όπως είπαμε τα λάθη στην προοδευτική πολλαπλή στοίχιση δεν διορθώνονται σε κάποιο επόμενο βήμα. Για το λόγο αυτό, θα ήταν επιθυμητό, όλες οι ακολουθίες της πολλαπλής στοίχισης να παίζουν κάποιο ρόλο στο πώς μια άλλη ακολουθία θα προστεθεί στη στοίχιση. Ένας τρόπος για να γίνει αυτό, θα ήταν να δημιουργηθεί από κάθε μια πολλαπλή στοίχιση, μια συναινετική ακολουθία (consensus), δηλαδή μια «ψεύτικη» ακολουθία στην οποία κάθε σύμβολο θα ήταν αυτό το οποίο εμφανίζεται με μεγαλύτερο ποσοστό στην πολλαπλή στοίχιση. Αυτή θα ήταν μια εύκολη λύση, αλλά και πάλι αδυνατεί να λάβει υπόψη όλες τις ακολουθίες. Φανταστείτε για παράδειγμα μια στήλη στην οποία υπάρχουν 2 A, 1 T, 1 G, και 1 C. Το A είναι φυσικά, το σύμβολο με τη μεγαλύτερη πιθανότητα, αλλά και πάλι η πληροφορία για το 60% των άλλων συμβόλων της συγκεκριμένης θέσης, δεν χρησιμοποιείται.



Εικόνα 4.3 *Λεπτομέρεια από το προτελευταίο βήμα της πολλαπλής στοίχισης που περιγράφεται στην Εικόνα 4.2. Βλέπουμε τη στοίχιση των ακολουθιών x_1 και x_2 από τη μια μεριά, και των x_3 και x_4 από την άλλη. Στο σημείο αυτό, τη στοίχιση των δύο στοιχίσεων, την κατευθύνει απόλυτα το ζευγάρι με τη μεγαλύτερη ομοιότητα (x_1-x_3). Προσέξτε ότι το κενό που υπήρχε στη στοίχιση των x_3 και x_4 παραμένει, ενώ τα κενά που εισάγονται στις x_1 και x_2, εισάγονται ταυτόχρονα και στις δύο.*

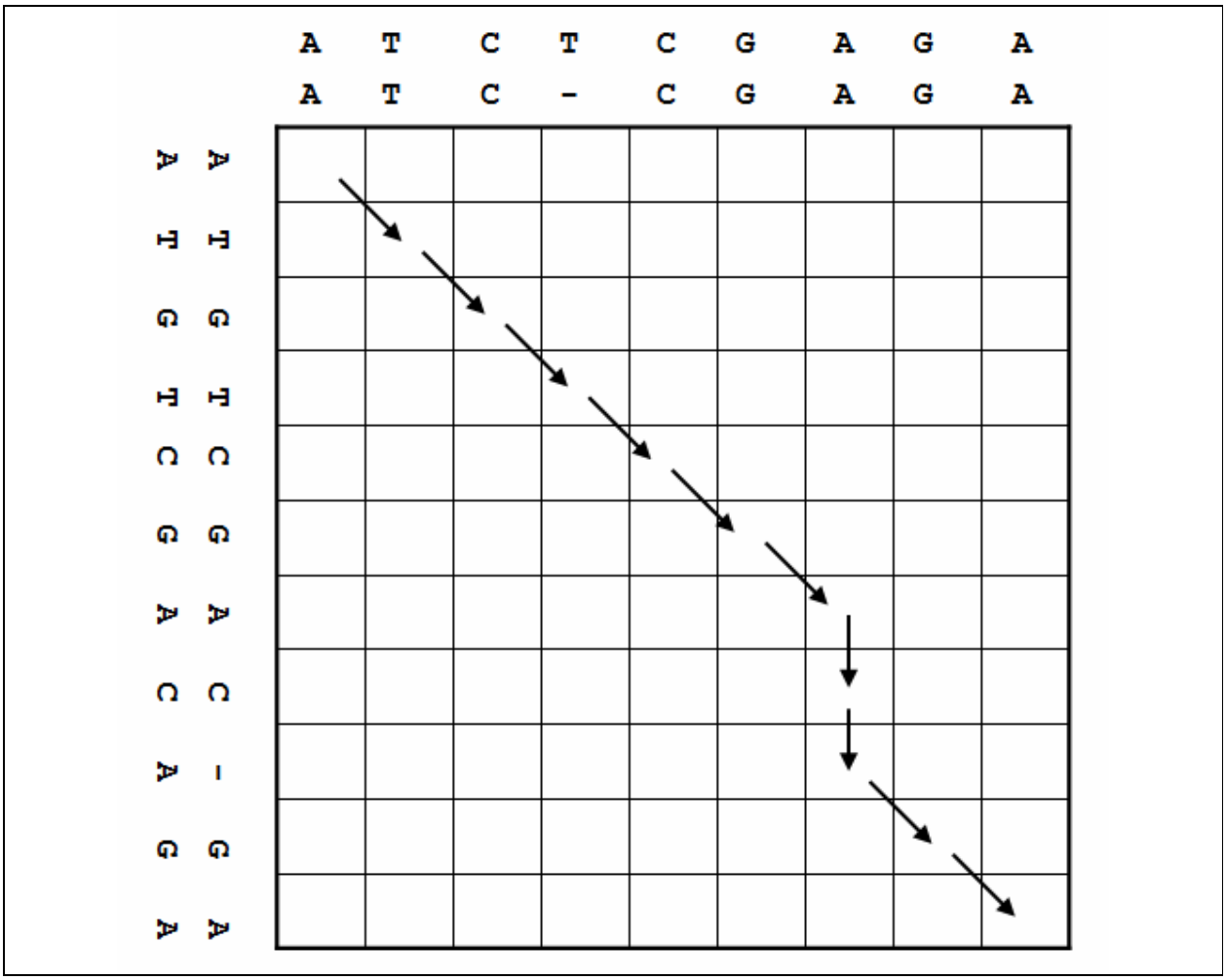
Μια καλύτερη λύση, είναι το λεγόμενο *profile alignment*, το οποίο μετράει τη σχετική συνεισφορά όλων των ακολουθιών της κάθε στοίχισης και τελικά πραγματοποιεί την στοίχιση λαμβάνοντας υπόψη όλες τις ακολουθίες. Τα μαθηματικά της μεθόδου είναι πολύπλοκα, αλλά μπορούν να απλοποιηθούν αν θεωρήσουμε, όπως και παραπάνω, το κενό σαν ένα πέμπτο σύμβολο (-), οπότε θα έχουμε και γραμμική ποινή για τα κενά. Τότε, με τη χρήση του SP score, μπορούμε να σκοράρουμε όλες τις ακολουθίες της μιας στοίχισης με όλες τις ακολουθίες της άλλης. Για απλότητα, θεωρούμε επίσης ότι στη μία στοίχιση περιέχονται οι ακολουθίες από 1 έως n, ενώ στην άλλη, οι ακολουθίες από n+1 έως N. Σε αυτή την περίπτωση η σχέση (4.8) γίνεται:

$$\begin{aligned}\sum_i SP\left(m_i\right) &= \sum_i \sum_{j<j'} s\left(m_i^j, m_i^{j'}\right) \\ &= \sum_i \sum_{j<j'\le n} s\left(m_i^j, m_i^{j'}\right) + \sum_i \sum_{n<j<j'\le N} s\left(m_i^j, m_i^{j'}\right) + \sum_i \sum_{j\le n, n<j'\le N} s\left(m_i^j, m_i^{j'}\right)\end{aligned} \tag{4.14}$$

Τα δύο πρώτα αθροίσματα στο δεξί σκέλος της σχέσης (4.14) δεν αλλάζουν καθώς κάθε μια από τις στοιχίσεις παραμένει σταθερή, οπότε αυτό που μένει να βελτιστοποιηθεί είναι το τελευταίο άθροισμα, το οποίο περιέχει τις συνεισφορές από τις χιαστί συγκρίσεις των ακολουθιών των δύο στοιχίσεων. Η βελτιστοποίηση, γίνεται με τον κλασικό πίνακα του δυναμικού προγραμματισμού που συναντήσαμε στο κεφάλαιο 3 (Εικόνα 4.4). Στην πράξη, η μέθοδος αυτή είναι η πιο αποτελεσματική και χρησιμοποιείται από τους περισσότερους σύγχρονους αλγόριθμους. Παρόλα αυτά, υπάρχουν πάρα πολλές επί μέρους διαφοροποιήσεις, ανάλογα με το σύστημα του score και το πώς ο κάθε αλγόριθμος χειρίζεται τα κενά (Edgar & Sjolander, 2004; Wang & Dunbrack, 2004)

Ίσως το πιο γνωστό και περισσότερο χρησιμοποιημένο, πρόγραμμα πολλαπλής στοίχισης το **CLUSTALW** (Julie D Thompson, Higgins, & Gibson, 1994) ήταν το πρώτο που χρησιμοποίησε profile alignment με την προοδευτική πολλαπλή στοίχιση. Το clustal ξεκίνησε από την έκδοση CLUSTALV (Higgins, Bleasby, & Fuchs, 1992) ενώ στην πορεία αναπτύχθηκε η έκδοση CLUSTALW αλλά και η έκδοση που υποστήριζε γραφικά, η CLUSTALX (J. D. Thompson, Gibson, & Higgins, 2002). Ο βασικός αλγόριθμος βέβαια, είναι ίδιος και εξελίσσεται με τα χρόνια ενσωματώνοντας πολλές ευριστικές τεχνικές οι οποίες έχουν προκύψει από εμπειρική παρατήρηση και οι οποίες προσδίδουν μεγαλύτερη σταθερότητα και αξιοπιστία στη μέθοδο. Η μέθοδος είναι διαθέσιμη στη διεύθυνση www.ebi.ac.uk/clustalw/.Τα βασικά σημεία της, είναι:

α) Στις αρχικές εκδόσεις της μεθόδου, ο αλγόριθμος έκανε τις στοιχίσεις κατά ζεύγη με έναν ευριστικό αλγόριθμο (FASTA), με αποτέλεσμα να είναι ιδιαίτερα γρήγορος. Σε κατοπινές εκδόσεις δίνει τη δυνατότητα, εναλλακτικά, να χρησιμοποιηθεί ένας αλγόριθμος δυναμικού προγραμματισμού, για καλύτερα αποτελέσματα.



Εικόνα 4.4 *Το προτελευταίο βήμα της πολλαπλής στοίχισης που περιγράφεται στην Εικόνα 4.2.όπως θα είχε πραγματοποιηθεί με χρήση profile alignment. Σε έναν κλασικό πίνακα δυναμικού προγραμματισμού, τοποθετούμε τη στοίχιση των ακολουθιών x_1 και x_2 από τη μια μεριά, και των x_3 και x_4 από την άλλη. Οι δύο στοιχίσεις σκοράρονται με τη σχέση (4.14) και η βέλτιστη διαδρομή εντοπίζεται με τον κλασικό τρόπο. Και σε αυτή την περίπτωση, το κενό που υπήρχε στη στοίχιση των x_3 και x_4 παραμένει, ενώ τα κενά που εισάγονται στις x_1 και x_2, εισάγονται ταυτόχρονα και στις δύο ακολουθίες. Στο παράδειγμα αυτό, η τελική στοίχιση είναι ίδια με όλες τις μεθόδους, αλλά αυτό δεν ισχύει γενικά. Σε άλλες περιπτώσεις, η μέθοδος αυτή θα δώσει διαφορετικά αποτελέσματα, τα οποία σε γενικές γραμμές θα είναι και καλύτερα.*

β) Οι αποστάσεις υπολογίζονται απευθείας από την επί τοις εκατό ομοιότητα των ακολουθιών x (D=1-x/100) ενώ το δέντρο-οδηγός κατασκευάζεται με την ιδιαίτερα αποτελεσματική και σταθερή, μέθοδο

Neighbor-Joining (ένωση γειτόνων) (Saitou & Nei, 1987). Η μέθοδος αυτή είναι μια μέθοδος ομαδοποίησης που προτάθηκε αρχικά για χρήση σε φυλογενετικά δέντρα. Θα την αναλύσουμε στο κεφάλαιο 6.

γ) Για την προσθήκη μιας ακολουθίας (ή μιας πολλαπλής στοίχισης) σε μια υπάρχουσα πολλαπλή στοίχιση, χρησιμοποιεί τη μέθοδο profile alignment.

δ) Τέλος, χρησιμοποιεί μια σειρά από πολύ προσεκτικά επιλεγμένες ευριστικές τεχνικές οι οποίες μεγιστοποιούν το αποτέλεσμα. Για παράδειγμα, οι πολύ όμοιες ακολουθίες λαμβάνουν μικρό σχετικό βάρος (weight) έτσι ώστε να μην επηρεάζουν τόσο πολύ και να μην κατευθύνουν την πολλαπλή στοίχιση. Μια άλλη ιδιαιτερότητα είναι ότι ο πίνακας ομοιότητας δεν είναι σταθερός, αλλά επιλέγεται από τον αλγόριθμο ανάλογα με το ποσοστό ομοιότητας που εντοπίζεται στις υπό μελέτη ακολουθίες. Επιπλέον, οι ποινές για τα κενά, δεν είναι σταθερές, αλλά ειδικές ανά θέση (υδρόφοβες περιοχές λαμβάνουν μεγαλύτερη ποινή για τα κενά, με συνέπεια να καθίσταται πιο δύσκολη η εισαγωγή κενών σε αυτές τις περιοχές, αντίθετα, η ποινή μειώνεται αν βρεθούν πάνω από 5 συνεχόμενα υδρόφιλα κατάλοιπα). Τέλος, οι ποινές για τα κενά αυξάνονται αν στην ίδια στήλη της στοίχισης δεν υπάρχουν κενά, αλλά αντίθετα υπάρχει κάπου δίπλα μια περιοχή με πολλά κενά. Αυτό έχει σαν συνέπεια τα κενά να «συσσωρεύονται» σε συγκεκριμένες θέσεις σε μια στοίχιση. Όλες αυτές οι τεχνικές, έχουν βελτιωθεί με τα χρόνια και έχουν κάνει το CLUSTAL να είναι ένα από τα πιο αξιόπιστα εργαλεία πολλαπλής στοίχισης, παρόλο που κατά βάση στηρίζεται σε μια απλή ευριστική μέθοδο.

Ένας άλλος σύγχρονος αλγόριθμος πολλαπλής στοίχισης, ο οποίος βασίζεται στην προοδευτική πολλαπλή στοίχιση, είναι το **Kalign** (Lassmann & Sonnhammer, 2005), (διαθέσιμο στη διεύθυνση http://msa.sbc.su.se/cgi-bin/msa.cgi). Στο Kalign, όλες οι επιλογές της προοδευτικής πολλαπλής στοίχισης είναι βελτιστοποιημένες με σκοπό την ταχύτητα. Βασισμένοι στην παρατήρηση ότι το μεγαλύτερο ποσοστό του υπολογιστικού χρόνου οι αλγόριθμοι το καταναλώνουν στις κατά ζεύγη στοιχίσεις από τις οποίες θα υπολογιστούν οι αποστάσεις, οι Lassmann και Sonnhammer επέλεξαν αντί για έναν αλγόριθμο στοίχισης δυναμικού προγραμματισμού, τον προσεγγιστικό αλγόριθμο ταύτισης συμβολοσειρών, των Wu και Manber, ο οποίος είναι γραμμικός ως προς το μήκος της ακολουθίας (Wu & Manber, 1992). Με αυτόν τον τρόπο το Kalign εκτιμά τις αποστάσεις το ίδιο γρήγορα με τη μέθοδο των k-tuple, αλλά πολύ πιο αποδοτικά. Επιπλέον, το δέντρο-οδηγός κατασκευάζεται με τη μέθοδο UPGMA η οποία είναι ίσως η πιο γρήγορη (αλλά όχι τόσο ακριβής) μέθοδος ομαδοποίησης. Και αυτή τη μέθοδο θα την αναλύσουμε στο κεφάλαιο των φυλογενετικών σχέσεων. Οι στοιχίσεις πραγματοποιούνται με την κλασική μέθοδο profile alignment, με την επιπλέον επιλογή, οι κοινές ακολουθίες που βρέθηκαν στο πρώτο βήμα, να μπορούν να καθοδηγούν τη στοίχιση (αυτή η επιλογή καθυστερεί κάπως τους υπολογισμούς, αλλά είναι πιο ακριβής). Τέλος, μια άλλη ιδιαιτερότητα της μεθόδου, βασίζεται στην παρατήρηση ότι πολύ όμοιες ακολουθίες στοιχίζονται αρκετά καλά ανεξαρτήτως του πίνακα ομοιότητας, αλλά ακολουθίες οι οποίες βρίσκονται εξελικτικά μακριά, απαιτούν τον κατάλληλο πίνακα (BLOSUM50, PAM250 ή GONNET250). Βασισμένοι σε αυτό, οι συγγραφείς επέλεξαν σε όλες τις περιπτώσεις να χρησιμοποιείται ο πίνακας GONNET250 (Gonnet, Cohen, & Benner, 1992), μια επιλογή που διευκολύνει αρκετά τους υπολογισμούς. Με όλες αυτές τις βελτιστοποιήσεις, το Kalign καταφέρνει να αποδίδει ελάχιστα χειρότερα από το CLUSTAL αλλά να πραγματοποιεί τις στοιχίσεις ως και 10 φορές πιο γρήγορα. Όπως θα δούμε παρακάτω, ανάλογα με την εφαρμογή, υπάρχουν περιπτώσεις στις οποίες ο χρόνος είναι πιο καθοριστικός παράγοντας σε σχέση με την ακρίβεια. Περισσότερα για το πως αξιολογούμε την ακρίβεια μιας μεθόδου πολλαπλής στοίχισης, θα δούμε στο τέλος του κεφαλαίου.

Δεν πρέπει να ξεχνάμε, ότι η προοδευτική πολλαπλή στοίχιση, είναι ευριστική μέθοδος. Δεν βελτιστοποιεί κάποιο ολικό μέτρο «καταλληλότητας» της στοίχισης, και δεν διαχωρίζει τη διαδικασία αξιολόγησης μιας στοίχισης από τον αλγόριθμο βελτιστοποίησης. Το πιο σημαντικό από όλα, είναι το γεγονός ότι με τον τρόπο που δουλεύει η μέθοδος, ένα κενό που εισάγεται νωρίς στη διαδικασία, δεν αναιρείται ποτέ («*once a gap, always a gap*»). Παρόλα αυτά, είναι ιδιαίτερα ενδιαφέρον το γεγονός ότι ευριστικοί αλγόριθμοι με προσεκτικά επιλεγμένες επιλογές, καταφέρνουν να αποδίδουν ιδιαίτερα καλά. Στην επόμενη ενότητα, θα δούμε μια άλλη μεγάλη κατηγορία μεθόδων, οι οποίες αν και είναι υπολογιστικά περισσότερο απαιτητικές επιδιώκουν να διορθώσουν τέτοια αρχικά λάθη της στοίχισης.

4.3. Επαναληπτικές μέθοδοι και μέθοδοι που βασίζονται στη συνέπεια

Η βασική ιδέα των επαναληπτικών μεθόδων, είναι να χρησιμοποιηθεί κάποιου είδους προοδευτική πολλαπλή στοίχιση, αλλά αυτή η διαδικασία να γίνει επαναληπτικά έτσι ώστε λάθη που είναι πιθανό να εισχωρήσουν σε αρχικά στάδια της στοίχισης, να μπορούν να αναιρεθούν σε κάποιο μετέπειτα βήμα. Η επαναληπτική διαδικασία, είναι σε γενικές γραμμές μια εύκολα υλοποιήσιμη ιδέα, και εμπειρικές αναλύσεις έχουν δείξει ότι

μπορεί να χρησιμοποιηθεί ακόμα και σε ήδη υπάρχοντες αλγόριθμους, αυξάνοντας σημαντικά την απόδοσή τους. Για παράδειγμα, η ακρίβεια του CLUSTALW αυξάνει κατά 6% με αυτή τη διαδικασία (Wallace, O'Sullivan, & Higgins, 2005).

Μια από τις πρώτες υλοποιήσεις επαναληπτικού αλγόριθμου, ήταν ο αλγόριθμος των Barton και Sternberg (Barton & Sternberg, 1987). Ο αλγόριθμος σε γενικές γραμμές, έκανε τις στοιχίσεις με κλασικό δυναμικό προγραμματισμό και μετά ξεκινούσε την πολλαπλή στοίχιση από τις ακολουθίες με τη μεγαλύτερη ομοιότητα. Στη συνέχεια, πρόσθετε στη στοίχιση την επόμενη πιο όμοια ακολουθία χρησιμοποιώντας profile alignment. Όταν είχε στοιχίσει όλες τις ακολουθίες, τις αφαιρούσε διαδοχικά μία-μία από τη στοίχιση και τις πρόσθετε εκ νέου, έως ότου βρεθεί μια πολλαπλή στοίχιση με καλύτερο score. Παρόμοια στρατηγική είχε και ο αλγόριθμος του Corpet (Corpet, 1988), στον οποίο βασίζεται το πρόγραμμα πολλαπλής στοίχισης **MULTALIN** (http://prodes.toulouse.inra.fr/multalin/multalin.html). Η βασική διαφορά είναι στο πώς γίνεται το επαναληπτικό βήμα. Στο MULTALIN, όταν ολοκληρωθεί η πρώτη πολλαπλή στοίχιση, το νέο δέντρο, το οποίο προκύπτει με ιεραρχική ομαδοποίηση, περιέχει ένα βραχίονα λιγότερο γιατί οι δύο πιο όμοιες ακολουθίες θεωρούνται μία ομάδα και αυτό συνεχίζεται και στις επόμενες επαναλήψεις.

Το **MUSCLE** είναι ένα σύγχρονο πρόγραμμα προοδευτικής στοίχισης το οποίο εργάζεται επαναληπτικά (Edgar, 2004) (είναι διαθέσιμο στη διεύθυνση http://www.drive5.com/muscle). Στον πρώτο κύκλο, το MUSCLE χρησιμοποιεί μια γρήγορη μέθοδο βασισμένη στα *k*-mers (κοινές υπο-ακολουθίες μήκους *k*), για να υπολογίσει αποστάσεις και να κατασκευάσει γρήγορα ένα δέντρο οδηγό με τη μέθοδο UPGMA, από το οποίο θα κατασκευάσει μια πρόχειρη στοίχιση (την ονομάζει MSA1). Από αυτή τη στοίχιση, θα υπολογιστούν αποστάσεις με τη μέθοδο του Kimura (η οποία απαιτεί την ύπαρξη της πολλαπλής στοίχισης – λεπτομέρειες και για αυτή τη μέθοδο θα δούμε στο κεφάλαιο 6), από τις οποίες με προοδευτική πολλαπλή στοίχιση και profile alignment θα κατασκευαστεί η δεύτερη στοίχιση (την οποία ονομάζει MSA2). Στο τελευταίο βήμα (refinement), η μέθοδος διαγράφει διαδοχικά βραχίονες του δέντρου το οποίο έχει προκύψει, και στοιχίζει ξανά τις ακολουθίες αυτού του βραχίονα με τις υπόλοιπες ακολουθίες του δέντρου. Αυτό το βήμα επαναλαμβάνεται μέχρι η μέθοδος να συγκλίνει ή μέχρι να ολοκληρωθεί ένας προκαθορισμένος από το χρήστη αριθμός επαναλήψεων. Το τελικό αποτέλεσμα είναι αυτό που το πρόγραμμα ονομάζει MSA3, αλλά το λογισμικό δίνει επιλογή να σταματάει και στο MSA2 (αυτή είναι η επιλογή MUSCLE-p) σαν μια γρήγορη λύση, καθώς είναι εμφανές ότι το μεγαλύτερο κομμάτι του χρόνου εκτέλεσης αναλώνεται στο τελευταίο επαναληπτικό βήμα του αλγορίθμου. Το MUSCLE-p έχει πολυπλοκότητα υπολογισμών $O(N^2L+NL^2)$ και μνήμης $O(N^2+NL+L^2)$, ενώ το τελευταίο βήμα προσθέτει ένα επιπλέον $O(N^3L)$ στην πολυπλοκότητα των υπολογισμών. Μια άλλη ιδιαιτερότητα του MUSCLE είναι το γεγονός ότι χρησιμοποιεί μια εντελώς διαφορετική μέθοδο για να σκοράρει το profile alignment, τη μέθοδο «log-expectation score». Το MUSCLE θεωρείται ένα από τα καλύτερα σύγχρονα εργαλεία, ενώ είναι ιδιαίτερα δυνατό στη στοίχιση profiles όχι μόνο σαν ενδιάμεσο βήμα στην κατασκευή της πολλαπλής στοίχισης, αλλά και σαν αυτοδύναμη λειτουργία.

Ίσως ένας από τους πιο ενδιαφέροντες επαναληπτικούς αλγόριθμους, είναι ο αλγόριθος του Gotoh (Gotoh, 1996) ο οποίος υλοποιείται στο λογισμικό **PRRP/PRRN** (http://www.genome.ist.i.kyoto-u.ac.jp/~aln_user/prrn/index.html). Ο αλγόριθμος χρησιμοποιεί μια διπλή επαναληπτική στρατηγική με τυχαιοποίηση, η οποία βελτιστοποιεί ένα σταθμισμένο SP score (weighted sums-of-pairs score) με σύνθετη ποινή για τα κενά. Η πρωτοτυπία του, εντοπίζεται στο γεγονός ότι τόσο τα βάρη όσο και η ίδια η στοίχιση βελτιστοποιούνται ταυτόχρονα. Η εσωτερική επαναληπτική διαδικασία βελτιστοποιεί το σταθμισμένο SP score, ενώ η εξωτερική, βελτιστοποιεί τα βάρη τα οποία υπολογίζονται για το φυλογενετικό δέντρο που εκτιμάται από την παρούσα στοίχιση.

Το **PRALINE**, (Simossis & Heringa, 2005) το οποίο είναι διαθέσιμο στη διεύθυνση http://ibivu.cs.vu.nl/programs/pralinewww/, είναι μια επαναληπτική μέθοδος η οποία βασίζεται σε μια διαφορετική επαναληπτική στρατηγική. Οι ακολουθίες αντικαθίστανται από ένα profile το οποίο κατασκευάζεται με PSI-BLAST από μια αρχική πολλαπλή στοίχιση μόνο των πολύ όμοιων ακολουθιών. Αυτή η διαδικασία επαναλαμβάνεται, έως ότου τα profile συγκλίνουν και η συλλογή παραμείνει σταθερή. Κατόπιν, η πολλαπλή στοίχιση πραγματοποιείται με μια κλασική διαδικασία προοδευτικής στοίχισης στην οποία οι ακολουθίες αντικαθίστανται από τα profiles. Καθώς οι πολλαπλές στοιχίσεις παίζουν ρόλο και στους αλγόριθμους πρόγνωσης της δευτεροταγούς δομής, το PRALINE μπορεί να ενσωματώσει και αυτή την πληροφορία. Για τη χρήση των πολλαπλών στοιχίσεων στην πρόγνωση της δομής, θα μιλήσουμε στο αντίστοιχο κεφάλαιο, ενώ η ιδέα να αντικαθίστανται οι ακολουθίες από profile, θα μας απασχολήσει στο επόμενο κεφάλαιο στο οποίο θα μελετήσουμε αναλυτικά τα profiles. Η ιδέα αυτή είναι πολύ ενδιαφέρουσα, γιατί προτείνει μια εναλλαγή ανάμεσα στη διαδικασία πολλαπλής στοίχισης και τη διαδικασία πρόβλεψης της

δομής, από την οποία επωφελούνται και οι δύο διαδικασίες. Τέλος, η μέθοδος αυτή είναι πολύ ενδιαφέρουσα και για έναν άλλο λόγο. Με τη μέθοδο αυτή, μπορεί να μετρηθεί η συνέπεια (consistency) ανάμεσα στην τελική στοίχιση και στη συλλογή των profiles τα οποία χρησιμοποιήθηκαν. Η έννοια της συνέπειας με κάποιο εξωτερικό κριτήριο είναι βασική στην επόμενη μεγάλη ομάδα αλγορίθμων.

Το **Dialign** (Morgenstern, 2014), (διαθέσιμο στη διεύθυνση http://bibiserv.techfak.uni-bielefeld.de/dialign/), είναι μια ιδιαίτερη περίπτωση, καθώς είναι ένας από τους λίγους αλγόριθμους προοδευτικής πολλαπλής στοίχισης, ο οποίος πραγματοποιεί στοιχίσεις και με χαρακτηριστικά τοπικής στοίχισης (αλλά, φυσικά, διαθέτει και χαρακτηριστικά ολικής στοίχισης, όπως όλοι οι αλγόριθμοι πολλαπλής στοίχισης). Αρχικά πραγματοποιούνται όλες οι ανά δυο στοιχίσεις και στη συνέχεια συλλέγονται οι στοιχισμένες περιοχές στις οποίες δεν υπάρχουν κενά. Το όνομα 'Dialign' βγαίνει από αυτές τις διαγώνιες περιοχές (diagonal alignments in a dot plot). Το πρόγραμμα δεν βάζει αρχικά ποινή για τα κενά, και δεν επιχειρεί να στοιχίσει περιοχές που δεν έχουν μεγάλη ομοιότητα. Κατά συνέπεια, για ακολουθίες με μόνο τοπική ομοιότητα, η πολλαπλή στοίχιση περιορίζεται στις περιοχές με ξεκάθαρη ομολογία, αγνοώντας τις μη όμοιες περιοχές. Όταν όμως πρόκειται για ακολουθίες με την ομοιότητα να εκτείνεται σε όλο το μήκος τους, ο αλγόριθμος εντοπίζει τμήματα από τις ακολουθίες, που εκτείνονται σε όλο το μήκος τους, και έτσι μετατρέπεται σε αλγόριθμο ολικής στοίχισης. Σε ενδιάμεσες περιπτώσεις, ο αλγόριθμος δίνει μια μίξη: στοιχίζει τις κοινές περιοχές που έχουν μεγάλη ομοιότητα (ακόμα και σε όλο το μήκος), ενώ τις ακολουθίες στις οποίες δεν υπάρχει μια κοινή περιοχή, τις αφήνει εκτός στοίχισης. Κατά συνέπεια, το Dialign είναι περισσότερο ευέλικτο από τα υπόλοιπα προγράμματα, και μπορεί να εφαρμοστεί σε περισσότερες περιπτώσεις χωρίς παρέμβαση στις ακολουθίες (π.χ. εντοπισμό και αποκοπή των μη όμοιων περιοχών).

Μια άλλη μέθοδος, η οποία χρησιμοποιεί στοιχεία παρόμοια με αυτά τόσο του PRALINE όσο και του Dialign, είναι το **COBALT** (Papadopoulos & Agarwala, 2007), το οποίο αποτελεί τμήμα της σουίτας εργαλείων του NCBI (ftp://ftp.ncbi.nlm.nih.gov/pub/agarwala/cobalt). Το COBALT χρησιμοποιεί το BLAST και το RPS-BLAST σαν το εργαλείο ομοιότητας, και επιτελεί προοδευτική στοχίση με ένα δέντρο οδηγό το οποίο παράγεται με τη μέθοδο Neighbour-Joining, αλλά χρησιμοποιεί μια ελαφρώς μετασχηματισμένη απόσταση στην οποία συμμετέχουν τα score των ανά δύο στοιχίσεων (d_{ij}=1-(S_{ij}/2)(1/S_{ii}-1/S_{ij}). Στη συνέχεια, κάνει profile alignment με δυναμικό προγραμματισμό στον οποίο όμως υπάρχουν βασικές τροποποιήσεις, τόσο στο σκοράρισμα όσο και στον τρόπο χειρισμού των κενών. Η βασικότερη όμως ιδιαιτερότητά του, είναι ότι με τη χρήση του BLAST κάνει αναζήτηση τοπικής ομοιότητας, και με τη χρήση του RPS-BLAST, πραγματοποιεί αναζητήσεις των ακολουθιών έναντι της βάσης των συντηρημένων περιοχών του NCBI (CDD). Με αυτόν τον τρόπο καταφέρνει να στοιχίζει καλά τις συντηρημένες περιοχές των ακολουθιών, επιτυγχάνοντας κάτι παρόμοιο με το Dialign.

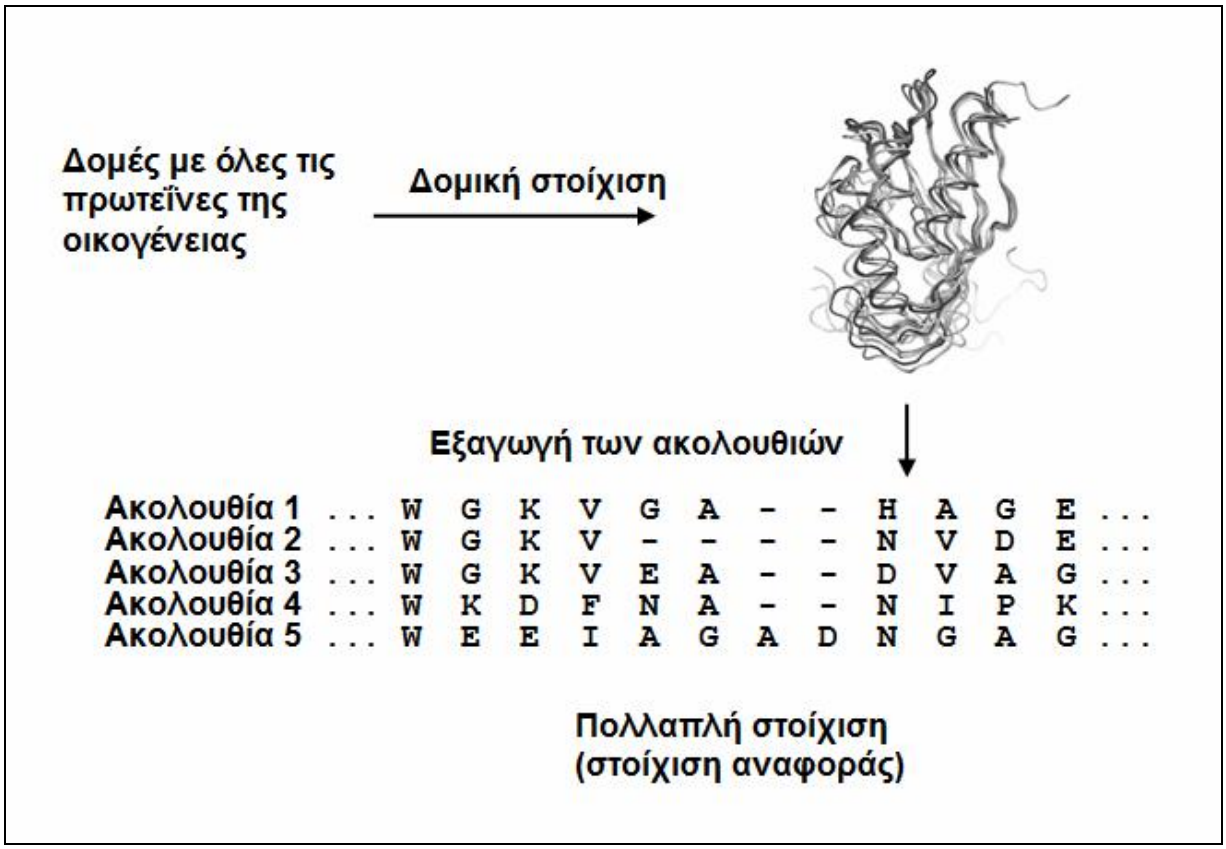
Τέλος, το πιο σημαντικό ίσως από τα εργαλεία που βασίζονται σε προοδευτική πολλαπλή στοίχιση χρησιμοποιώντας την έννοια της συνέπειας, είναι το **T-Coffee** (Magis et al., 2014), (διαθέσιμο στη διεύθυνση http://www.ch.embnet.org/software/TCoffee.html). Το T-Coffee μοιάζει πολύ με το CLUSTALW (τον κώδικα του οποίου μάλιστα, χρησιμοποιεί εσωτερικά): κάνει προοδευτική πολλαπλή στοίχιση, χρησιμοποιεί τον ίδιο αλγόριθμο για την κατασκευή του δέντρου, τον ίδιο τρόπο υπολογισμού των αποστάσεων αλλά και το profile alignment. Η βασική του διαφορά είναι ότι δημιουργεί μια «εκτεταμένη βιβλιοθήκη» όπως την αποκαλεί, στην οποία ένας πίνακας αντικατάστασης ειδικός ανά θέση αντιστοιχίζεται σε κάθε ζεύγος ακολουθιών, και ο οποίος αντικατοπτρίζει τη συμβατότητα της στοίχισης των δύο ακολουθιών με την υπόλοιπη βιβλιοθήκη. Με αυτόν τον τρόπο, οι πίνακες αντικατάστασης, αντικαθίστανται από αυτούς τους ειδικούς ανά θέση πίνακες, οπότε, κάθε πιθανή επέκταση μιας στοίχισης δύο ακολουθιών, ελέγχεται όχι με ένα γενικό πίνακα, αλλά με βάση το αν αυτές ταιριάζουν καλά στις υπόλοιπες ακολουθίες της βιβλιοθήκης. Μια άλλη διαφορά σε σχέση με το CLUSTALW είναι το γεγονός ότι στην αρχική βιβλιοθήκη, αυτή την οποία δημιουργεί από τις κατά ζεύγη στοιχίσεις, περιέχονται τόσο αποτελέσματα ολικής ομοιότητας (το βήμα αυτό επιτελείται με χρήση του CLUSTALW), όσο και αποτελέσματα τοπικής ομοιότητας (με χρήση του LALIGN από το πακέτο FASTA). Με αυτές τις ιδιαιτερότητες, το T-Coffee συνδυάζει τα χαρακτηριστικά από πολλούς από τους αλγορίθμους που αναφέραμε προηγουμένως. Κάνει προοδευτική στοίχιση που είναι γρήγορη, αλλά ελέγχει και τα διάφορα βήματα για τη συνέπειά τους έτσι ώστε να διορθώνονται τα λάθη. Επιπλέον, χρησιμοποιεί και τοπική αλλά και ολική πληροφορία. Με όλα αυτά, ο αλγόριθμος καταφέρνει να πραγματοποιεί πολύ καλές στοιχίσεις και να θεωρείται ίσως ο πιο πετυχημένος αλγόριθμος γενικής χρήσης αυτή τη στιγμή.

Τέλος, αξίζει να αναφερθεί και μια άλλη κατηγορία αλγορίθμων πολλαπλής στοίχισης, οι οποίοι δεν βασίζονται στην προοδευτική πολλαπλή στοίχιση, αλλά βελτιστοποιούν ένα συνολικό κριτήριο πάνω στην πολλαπλή στοίχιση και στηρίζονται σε τεχνικές γνωστές από το χώρο της τεχνητής νοημοσύνης και των

στοχαστικών μοντέλων. Μια τέτοια κατηγορία μεθόδων, είναι αυτές που βασίζονται στο λεγόμενο simulated annealing (Kim, Pramanik, & Chung, 1994), αλλά δεν υπάρχουν αυτή τη στιγμή αξιόπιστες σύγχρονες υλοποιήσεις του. Μια άλλη κατηγορία αποτελούν οι μέθοδοι που βασίζονται στους γενετικούς αλγόριθμους, όπως για παράδειγμα το παλιότερο πρόγραμμα **SAGA** (Notredame & Higgins, 1996), ενώ η πιο μεγάλη κατηγορία είναι οι μέθοδοι που βασίζονται σε πιθανοθεωρητικά μαρκοβιανά μοντέλα (Hidden Markov Models), όπως η μέθοδος **ProbCons** και **ProbAlign** (Roshan, 2014) (διαθέσιμα στο http://probalign.njit.edu/standalone.html). Η τελευταία κατηγορία μεθόδων είναι πολύ σημαντική, γιατί τα μοντέλα αυτά βρίσκουν πολλές εφαρμογές στη βιοπληροφορική, παρέχουν μια πιθανοθεωρητική ερμηνεία των αποτελεσμάτων αποφεύγοντας τις ευριστικές λύσεις, αλλά κυρίως, γιατί καταφέρνουν να επιλύουν παρόμοια προβλήματα με πολύ ικανοποιητικό τρόπο. Σε επόμενο κεφάλαιο, θα αναπτύξουμε κάποια βασικά θέματα που αφορούν τα μοντέλα αυτά.

4.4. Αξιολόγηση των εργαλείων πολλαπλής στοίχισης

Αφού παρουσιάσαμε τους κύριους αλγόριθμους πολλαπλής στοίχισης και τις διάφορες υλοποιήσεις τους, πρέπει να επιστρέψουμε τώρα στο πρόβλημα της αξιολόγησης. Πώς μπορούμε να αξιολογήσουμε αν ένα δεδομένο πρόγραμμα πολλαπλής στοίχισης δουλεύει καλά; Πώς μπορούμε να συγκρίνουμε δύο ή περισσότερα προγράμματα; Όπως είδαμε, υπάρχει ένας τρόπος να αξιολογηθεί μια δεδομένη πολλαπλή στοίχιση, και αυτό μπορεί να γίνει με χρήση κάποιου γενικού στατιστικού μέτρου όπως για παράδειγμα της εντροπίας. Παρόλα αυτά, χρειαζόμαστε και κάποιο εξωτερικό κριτήριο αντικειμενικότητας. Σε αυτό, θα πρέπει με κάποιον τρόπο να ενσωματωθεί και η βιολογική όψη του προβλήματος, καθώς για τις πολλαπλές στοιχίσεις, δεν υπάρχει γενικώς αποδεκτός ή εύκολος τρόπος υπολογισμού της στατιστικής σημαντικότητας.



Εικόνα 4.5 *Σχηματική αναπαράσταση του τρόπου δημιουργίας μιας δομικής στοίχισης.*

Γενικά, έχει γίνει αποδεκτό, ότι κριτήριο αναφοράς για μια πολλαπλή στοίχιση, είναι η λεγόμενη «δομική στοίχιση» (structural alignment). Μια δομική στοίχιση, προκύπτει από την υπέρθεση των τρισδιάστατων δομών μια πρωτεϊνικής οικογένειας, για την οποία γνωρίζουμε ότι τα μέλη της έχουν

ξεκάθαρη εξελικτική και δομική ομοιότητα. Βασική προϋπόθεση φυσικά, είναι να υπάρχει κάποια ή κάποιες οικογένειες πρωτεϊνών, για τις οποίες υπάρχουν τρισδιάστατες δομές για μεγάλο αριθμό από τα μέλη τους. Η υπέρθεση των δομών, στην πιο απλή της μορφή, ελαχιστοποιεί τις αποστάσεις των αντίστοιχων ατόμων από τις διαφορετικές πρωτεΐνες και τις φέρνει όσο το δυνατό πιο κοντά στο χώρο. Στην πιο περίπλοκη κατάσταση, κατά την οποία οι πρωτεΐνες δεν έχουν το ίδιο μήκος, επειδή για παράδειγμα σε κάποιες λείπουν κάποιες περιοχές, απαιτούνται ειδικοί αλγόριθμοι στοίχισης των τρισδιάστατων δομών. Σε κάθε περίπτωση, από μία καλά κατασκευασμένη δομική στοίχιση, μπορεί να προκύψει μια πολλαπλή στοίχιση ακολουθιών, απλά αγνοώντας τη δομή και κρατώντας την πληροφορία μόνο της ακολουθίας. Με αυτόν τον τρόπο, κάθε στήλη της πολλαπλής στοίχισης αντιστοιχεί στα αντίστοιχα αμινοξέα των περιλαμβανόμενων στη στοίχιση πρωτεϊνών, τα οποία βρίσκονται πιο κοντά στο χώρο. Όπως γίνεται φανερό, μια τέτοια στοίχιση θεωρείται σημείο αναφοράς («gold standard») για τις πρωτεΐνες της οικογένειας, και μια μέθοδος πολλαπλής στοίχισης θα θεωρείται καλή αν καταφέρνει να ανακατασκευάζει αυτή τη στοίχιση ή να την προσεγγίζει. Για το σκοπό αυτό, έχουν αναπτυχθεί μια σειρά από βάσεις δεδομένων οι οποίες περιέχουν τέτοιες δομικές πολλαπλές στοιχίσεις πρωτεϊνικών οικογενειών, με διαφορετικά χαρακτηριστικά. Η πρώτη τέτοια βάση δεδομένων ήταν η **BAliBASE** (J. D. Thompson, Plewniak, & Poch, 1999) και για πολλά χρόνια οι περισσότερες αξιολογήσεις γίνονταν πάνω σε αυτήν. Την τελευταία δεκαετία όμως έχουν αναπτυχθεί και άλλες τέτοιες βάσεις δεδομένων/συλλογές πρωτεϊνικών ακολουθιών οι οποίες δίνονται στον Πίνακα 4.1.

Βάση δεδομένων	***Ηλεκτρονική Διεύθυνση***
BAliBASE (J. D. Thompson, et al., 1999)	http://www-igbmc.u-strasbg.fr/BioInfo/BAliBASE/index.html
OxBench (Raghava, Searle, Audley, Barber, & Barton, 2003)	http://www.compbio.dundee.ac.uk/
SABmark (Van Walle, Lasters, & Wyns, 2005)	http://bioinformatics.vub.ac.be/databases/databases.html
PREFAB (Edgar, 2004)	http://drive5.com/muscle/prefab.htm.

Πίνακας 4.1 *Οι βάσεις δεδομένων με δομικές στοιχίσεις πρωτεϊνικών οικογενειών που χρησιμοποιούνται για την αξιολόγηση των μεθόδων πολλαπλής στοίχισης*

Σε μια αξιολόγηση, συνήθως επιλέγεται ένα μεγάλο και ετερογενές σύνολο από οικογένειες από κάποια βάση, και οι ακολουθίες υποβάλλονται στα προγράμματα για πολλαπλή στοίχιση. Ιδανικά, μια μέθοδος αποδίδει «τέλεια» όταν ανακατασκευάζει στο 100% την αρχική δομική στοίχιση. Προφανώς, όταν οι οικογένειες έχουν πολλά μέλη, αυτό είναι σχετικά δύσκολο να συμβεί και για αυτό το λόγο επιλέγονται διάφορα μέτρα που αξιολογούν, λ.χ. πόσες στήλες της πολλαπλής στοίχισης έχουν ανακατασκευαστεί σωστά από το κάθε πρόγραμμα. Υπάρχουν διάφορα τέτοια μέτρα, με άλλα να υπολογίζουν το ποσοστό των σωστών στηλών επί του συνόλου της στοίχισης, και άλλα να κάνουν τον υπολογισμό με βάση τη στοίχιση αναφοράς (J. D. Thompson, Linard, Lecompte, & Poch, 2011), ενώ υπάρχουν ακόμα και μέτρα που αξιολογούν τη σωστή στοίχιση των διαφόρων περιοχών (blocks) (Raghava, et al., 2003). Γενικά στην αξιολόγηση, πρέπει να επιλέγεται ένα μεγάλο δείγμα οικογενειών, αντιπροσωπευτικό του τι αναμένεται να συναντήσει ο ερευνητής σε μια πραγματική κατάσταση. Οι παράγοντες στους οποίους πρέπει να δοθεί βαρύτητα είναι: το μέγεθος της οικογένειας και το μέσο μήκος των πρωτεϊνών της οικογένειας (παράγοντες που αναμένεται να επηρεάζουν τόσο τη συνολική απόδοση των μεθόδων σε απόλυτες τιμές, όσο και τον χρόνο εκτέλεσης της στοίχισης), η ομοιότητα των μελών της οικογένειας (είναι πολύ σημαντικό να ξέρουμε αν ένα πρόγραμμα δουλεύει καλά τόσο σε κοντινές εξελικτικά πρωτεΐνες, όσο και σε μακρινές), αλλά και το αν στην οικογένεια υπάρχουν πρωτεΐνες-μέλη οι οποίες αποτελούν θραύσματα (fragments), καθώς αυτό θα ελέγξει την ικανότητα του προγράμματος στην «τοπική» πολλαπλή στοίχιση και στην αναγνώριση των διακριτών περιοχών.

Υπάρχουν ακόμα και μέτρα που δεν συγκρίνουν τη στοίχιση με κάποια στοίχιση αναφοράς, αλλά κάνουν απευθείας αναγωγή στις τρισδιάστατες δομές για να δώσουν περισσότερο βάρος σε περιοχές με κανονική δευτεροταγή δομή, σε αντίθεση με τις βρόχους στις οποίες τα κενά και τα λάθη είναι περισσότερο κοινά αλλά και λιγότερο σημαντικά (Raghava, et al., 2003). Ένα τέτοιο μέτρο είναι το APDB (O'Sullivan et al., 2003). Το APDB, αξιολογεί μια πολλαπλή στοχίση, με κριτήριο αναφοράς δύο ή περισσότερες δομές από

την PDB, χωρίς όμως να απαιτεί μια στοίχιση αναφοράς ή υπέρθεση των δομών. Στη σχετική δημοσίευση, οι συγγραφείς έδειξαν ότι το μέτρο παράγει αξιολογήσεις που σε γενικές γραμμές συμφωνούν με αυτές που προκύπτουν από μια στοίχιση αναφοράς, και κατά συνέπεια το APDB θα μπορούσε να χρησιμοποιηθεί γενικά, ακόμα και σε οικογένειες για τις οποίες δεν υπάρχει μια αξιόπιστη πολλαπλή στοίχιση.

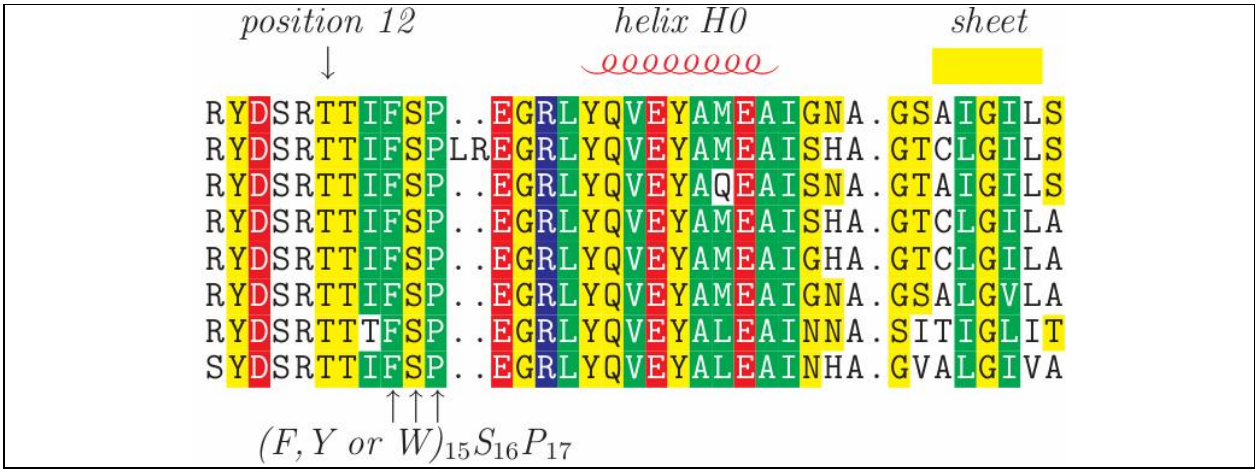
Τα τελευταία χρόνια, έχουν γίνει αρκετές διαφορετικές μελέτες αξιολόγησης των προγραμμάτων πολλαπλής στοίχισης, κάθε φορά με διαφορετικές οικογένειες πρωτεϊνών και πολλές φορές και με διαφορετικά μέτρα αξιολόγησης (Pais, Ruy Pde, Oliveira, & Coimbra, 2014; J. D. Thompson, et al., 2011; J. D. Thompson, et al., 1999). Επίσης, κάθε νέος αλγόριθμος πολλαπλής στοίχισης, πρέπει πλέον να αξιολογείται με παρόμοια κριτήρια, αν πρόκειται να δημοσιευθεί. Παρόλο που οι επιμέρους μελέτες διαφέρουν πολλές φορές ως προς τη μεθοδολογία, μπορούμε να εξάγουμε κάποια γενικά συμπεράσματα. Για παράδειγμα, τα περισσότερα από τα σύγχρονα εργαλεία που αναφέραμε παραπάνω, σε ένα ευρύ φάσμα συνθηκών αποδίδουν πολύ καλά, πετυχαίνοντας πάνω από 50% επιτυχία στην ανακατασκευή των στοιχίσεων αναφοράς, ακόμα και σε οικογένειες με μέσο ποσοστό ομοιότητας γύρω στο 20%. Το T-Coffee, το ProbCons και το ProbAlign είναι σε γενικές γραμμές οι πιο αποδοτικοί αλγόριθμοι, αλλά είναι και πιο χρονοβόροι και με μεγάλες απαιτήσεις σε μνήμη (ιδιαίτερα τα δύο τελευταία). Το ClustalW και το MUSCLE, ακολουθούν με μικρή διαφορά στην απόδοση, αλλά υπερτερούν σε ταχύτητα εκτέλεσης και σε απαιτήσεις σε μνήμη. Το Prrp/Prrn είναι επίσης καλό, αλλά πιο αργό. Το Kalign, είναι σε γενικές γραμμές ελαφρώς χειρότερο, αλλά είναι έως και 10 φορές γρηγορότερο από το CLUSTALW (πολύ δε περισσότερο από τα υπόλοιπα), και κατά συνέπεια καλύτερο για αναλύσεις μεγάλου όγκου δεδομένων σε καθημερινή βάση. Τέλος, οι αλγόριθμοι που κάνουν ολική στοίχιση, αποδίδουν σε γενικές γραμμές καλύτερα, εκτός αν στις πολλαπλές στοιχίσεις υπάρχουν μεγάλες περιοχές στο αμινοτελικό ή στο καρβοξυτελικό άκρο, οι οποίες δεν ταυτίζονται σε όλα τα μέλη της οικογένειας (δηλαδή, αν υπάρχουν οικογένειες με μέλη τα οποία εμφανίζουν τοπική ομοιότητα). Το T-Coffee γενικά, είναι ένας καλός συμβιβασμός, καθώς τα καταφέρνει σχετικά καλά σε όλες τις περιπτώσεις, ενώ το Dialign αποδεικνύεται καλύτερο μόνο σε κάποια από τα σετ με τέτοιες ακολουθίες (στις πιο ακραίες περιπτώσεις).

Τα δύο τελευταία χαρακτηριστικά, δηλαδή η ταχύτητα και η ικανότητα σωστής στοίχισης σε περιπτώσεις τοπικής ομοιότητας πρέπει να ελέγχονται προσεκτικά και να λαμβάνονται σοβαρά υπόψη στην επιλογή προγράμματος. Η ταχύτητα για παράδειγμα, δεν είναι σημαντική όταν κάνουμε μια μελέτη μιας συγκεκριμένης οικογένειας (θέλουμε να πάρουμε την καλύτερη δυνατή στοίχιση και δεν μας πειράζει να περιμένουμε λίγο). Από την άλλη όμως, είναι ένας σημαντικός παράγοντας αν πρόκειται τις πολλαπλές στοιχίσεις να τις χρησιμοποιούμε λ.χ. για την υποβοήθηση μιας μεθόδου πρόγνωσης, γιατί σε αυτή την περίπτωση θα χρειάζεται να επαναλαμβάνουμε τις στοιχίσεις καθημερινά (για παράδειγμα, αν φτιάχνουμε μια διαδικτυακή εφαρμογή). Κάτι αντίστοιχο ισχύει και για τις τοπικές ομοιότητες των πρωτεϊνών. Αν μελετάμε μια συγκεκριμένη οικογένεια πρωτεϊνών, κατά πάσα πιθανότητα θα ξέρουμε τι είδους στοίχιση να περιμένουμε. Αν όμως πρόκειται η πολλαπλή στοίχιση να χρησιμοποιείται σε μια αυτοματοποιημένη διαδικασία, τότε δεν έχουμε αυτή την πολυτέλεια. Τέλος, ένας άλλος παράγοντας που πρέπει να λαμβάνεται υπόψη είναι και η ευκολία προς τον απλό χρήστη. Τα περισσότερα από τα προγράμματα που αναφέραμε (CLUSTALW, T-Cofffee, Dialign, Kalign, MUSCLE, ProbAlign, Prrp/Prrn), προσφέρονται σαν διαδικτυακές εφαρμογές αλλά και σαν τοπικές εφαρμογές τις οποίες ο χρήστης μπορεί να εγκαταστήσει στον υπολογιστή του. Τα περισσότερα από αυτά, είναι ιδιαίτερα εύκολα στην εγκατάσταση σε όλα τα συστήματα (Windows, Linux, Mac), αλλά το COBALT και το PRALINE, τα οποία απαιτούν χρήση και άλλων προγραμμάτων (PSI-BLAST κλπ), είναι πιο δύσκολα στη ρύθμιση (και για την ακρίβεια, για το PRALINE δεν είμαστε σίγουροι αν υπάρχει και διαθέσιμη εφαρμογή πέραν της διαδικτυακής). Όλα τα παραπάνω είναι παράγοντες που πρέπει να λαμβάνονται σοβαρά υπόψη από τον χρήστη πριν επιλέξει με ποιο πρόγραμμα θα πραγματοποιήσει την ανάλυση του, και σε κάθε περίπτωση, είναι χρήσιμο πάντα κάποιος να δοκιμάζει αρκετές εναλλακτικές προτάσεις.

4.5. Οπτικοποίηση και Επεξεργασία μιας Πολλαπλής Στοίχισης

Το τελευταίο μέρος του κεφαλαίου, είναι αφιερωμένο στο λογισμικό οπτικοποίησης και επεξεργασίας των πολλαπλών στοιχίσεων, όπως και στους τύπους αρχείων πολλαπλής στοίχισης. Τα περισσότερα από τα προγράμματα που αναφέραμε, διαβάζουν ακολουθίες σε απλή μορφή (απλό κείμενο ή FASTA) και παράγουν τις πολλαπλές στοιχίσεις σε κάποια από τις γνωστές μορφές. Υπάρχουν αρκετοί τύποι αρχείων πολλαπλής στοίχισης, αλλά οι βασικοί είναι το Multi-FASTA, το MSF και το CLUSTAL. Το Multi-FASTA, είναι η πιο

απλή μορφή και είναι μια γενίκευση του FASTA. Κάθε ακολουθία δίνεται ξεχωριστά, με την πρώτη γραμμή να αποτελεί το όνομα της ή την περιγραφή (με ένα «>» στην αρχή), ενώ οι επόμενες γραμμές περιέχουν την ακολουθία. Για να αναπαρασταθεί η έννοια της στοίχισης, στις ακολουθίες υπάρχουν κενά (-) με συνέπεια το μήκος των ακολουθιών να είναι ίδιο σε κάθε αρχείο. Η μορφή αυτή είναι πολύ απλή, αλλά δεν είναι εύκολα κατανοητή από το ανθρώπινο μάτι. Η μορφή MSF, λύνει αυτό το πρόβλημα καθώς σε αυτήν, οι ακολουθίες δίνονται σε κομμάτια, στοιχισμένα το ένα κάτω από το άλλο. Αν η στοίχιση έχει για παράδειγμα 3 ακολουθίες, θα υπάρχουν 3 γραμμές με τα πρώτα 50 αμινοξικά κατάλοιπα της κάθε ακολουθίας σε δεκάδες (μαζί με τα κενά της στοίχισης), μετά θα ακολουθούν άλλες 3 γραμμές με τα 50 επόμενα, κ.ο.κ. Προφανώς, σε κάθε γραμμή αναφέρεται το όνομα της πρωτεΐνης για να μπορούμε να την ξεχωρίζουμε. Το αρχείο, στην αρχή, περιέχει σε ξεχωριστή ενότητα (που ξεχωρίζει από τους χαρακτήρες «//») τα ονόματα όλων των ακολουθιών που υπάρχουν στη στοίχιση. Η μορφή CLUSTAL (η οποία προήλθε από το ομώνυμο πρόγραμμα), είναι πιο απλή στην αρχή (δεν περιέχει σε ξεχωριστή ενότητα τα ονόματα των ακολουθιών, και οι ακολουθίες δίνονται συνεχόμενα, χωρίς κενά), αλλά περιέχει μια επιπλέον γραμμή σε κάθε τμήμα το οποίο διαχωρίζει 60 αμινοξικά κατάλοιπα της στοίχισης. Η γραμμή αυτή συμβολίζει τη συνολική «ποιότητα» της στοίχισης και είναι εύκολα κατανοητή από το ανθρώπινο μάτι. Αν σε μια στήλη της πολλαπλής στοίχισης υπάρχει απόλυτη συντήρηση, στη γραμμή αυτή υπάρχει «*». Το «:» και το «.» συμβολίζουν μεγάλη και μικρότερη συντήρηση αντίστοιχα (εξαρτώνται από τον πίνακα ομοιότητας, όχι μόνο από το ποσοστό), ενώ το κενό (« ») συμβολίζει τη διαφορά (μη ταύτιση). Υπάρχουν και άλλες μορφές αρχείων πολλαπλής στοίχισης, όπως η PHYLIP ή η STOCKHOLM, αλλά αυτές που αναφέρθηκαν παραπάνω είναι οι πιο κοινές και διαβάζονται από όλα τα προγράμματα. Εργαλεία, όπως το READSEQ (http://www.ebi.ac.uk/Tools/sfc/readseq/), αλλά και αντίστοιχα modules στην BioPerl, BioPython ή BioJava, επιτρέπουν την εύκολη μετατροπή των αρχείων από τη μια μορφή στην άλλη.

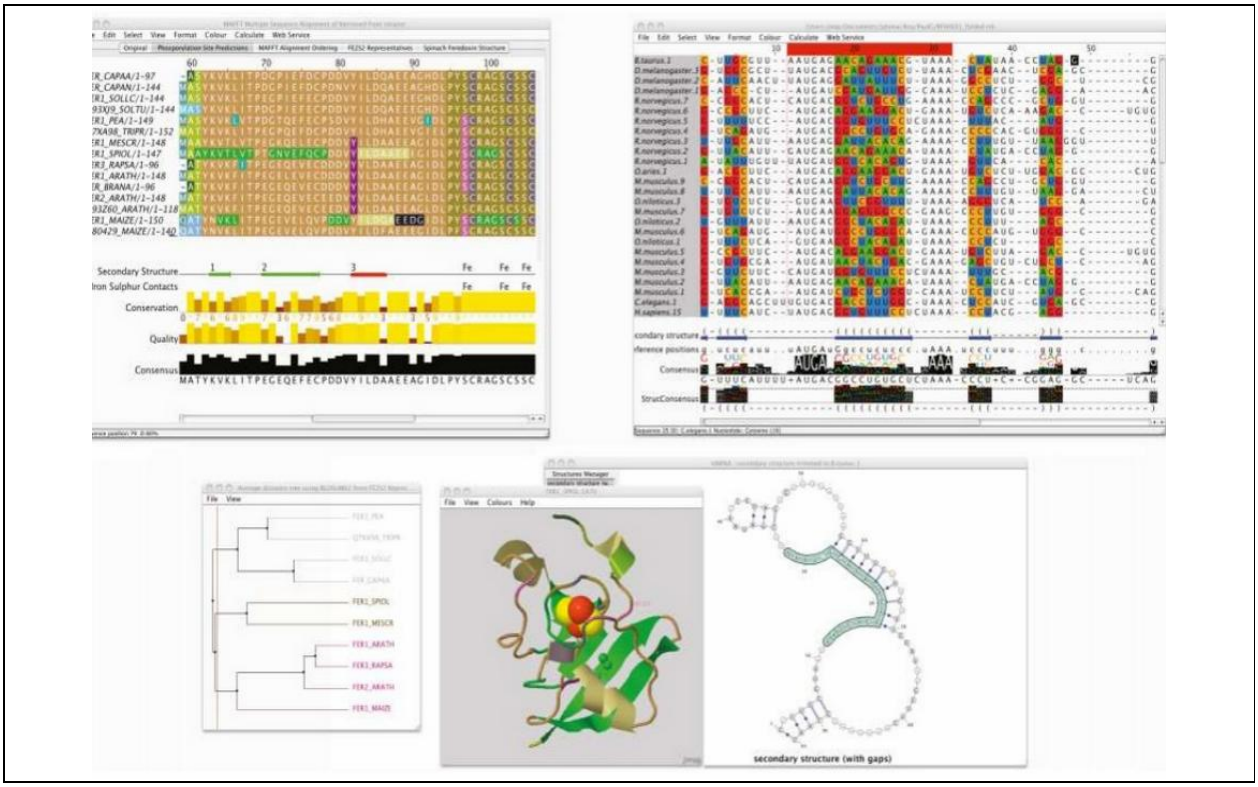


Εικόνα 4.6 *Πολλαπλή στοίχιση, όπως αναπαρίσταται από το Strap. Φαίνονται χρωματισμένα τα συντηρημένα κατάλοιπα ανάλογα με τις φυσικοχημικές τους ιδιότητες, και 3 θέσεις οι οποίες έχουν ενδιαφέρον έχουν επισημανθεί. Από πάνω, φαίνονται και οι προγνώσεις δευτεροταγούς δομής με το JNET.*

Φυσικά, ένα μεγάλο μέρος της εργασίας που απαιτείται σε μια πολλαπλή στοίχιση, δεν είναι μόνο η ίδια η στοίχιση και η εκτέλεσή της, αλλά και η οπτικοποίηση, η επεξεργασία και η ερμηνεία της. Καταλαβαίνουμε, όσο περιεκτικό και αν είναι από πλευράς πληροφορίας το αρχείο με τα αποτελέσματα μια πολλαπλής στοίχισης, ότι αυτό είναι δύσκολο να μελετηθεί και να αναλυθεί σωστά με το ανθρώπινο μάτι, ειδικά αν μιλάμε για στοιχίσεις μεγάλων πρωτεϊνικών οικογενειών. Για το σκοπό αυτό, έχουν αναπτυχθεί εδώ και χρόνια ειδικά εργαλεία τα οποία οπτικοποιούν τις στοιχίσεις ή τμήματα αυτών, και τις μορφοποιούν σε μορφή κατανοητή και κατάλληλη για παρουσίαση ή δημοσίευση. Τα παλιότερα από αυτά τα εργαλεία, έφτιαχναν απλά στατικές εικόνες βασισμένες σε ένα σύνολο οδηγιών (γραμματοσειρά, χρώμα καταλοίπων κ.ο.κ.). Τα σύγχρονα όμως εργαλεία, προσφέρουν πολλά περισσότερα. Δεν είναι μόνο διαδραστικά εργαλεία (editors), αλλά προσφέρουν και ένα ολοκληρωμένο περιβάλλον εργασίας, με διασυνδέσεις με άλλα εργαλεία (προγράμματα στοίχισης, προγνωστικούς αλγορίθμους κλπ) τόσο τοπικά όσο και στο διαδίκτυο, αλλά και διασυνδέσεις με τις βάσεις δεδομένων (ακολουθιών και δομών). Τα κυριότερα εργαλεία που χρησιμοποιούνται για το σκοπό αυτό είναι τα παρακάτω:

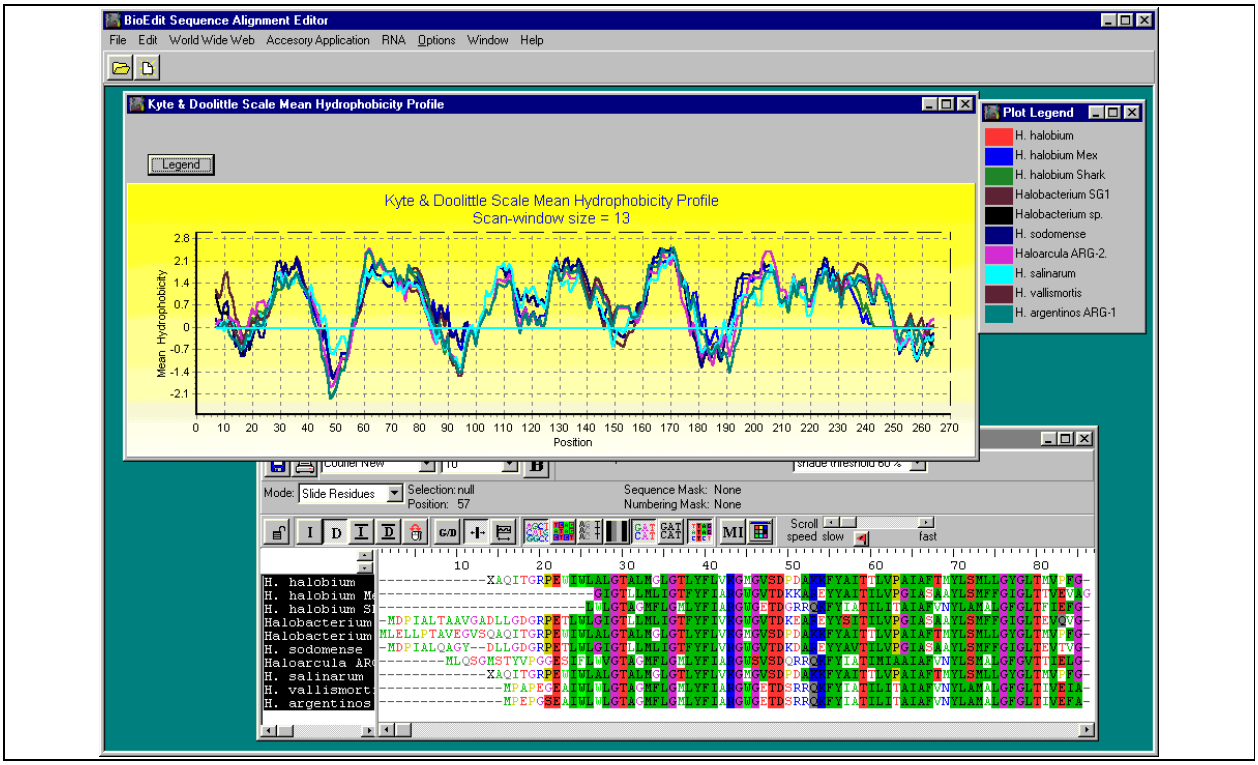
- Jalview (http://www.jalview.org/)
- Strap (http://www.bioinformatics.org/strap/)
- Seqpup (http://iubio.bio.indiana.edu/soft/molbio/seqpup/java/seqpup-doc.html)
- Seaview (http://pbil.univ-lyon1.fr/software/seaview.html)
- Cinema (http://aig.cs.man.ac.uk/research/utopia/cinema/cinema.php)
- Boxshade (http://www.ch.embnet.org/software/BOX_form.html)
- Bioedit (http://www.mbio.ncsu.edu/BioEdit/bioedit.html)

Τα εργαλεία αυτά, προσφέρουν μια σειρά από μεγάλες ευκολίες στο χρήστη. Τα περισσότερα είναι εφαρμογές Desktop με ολοκληρωμένο περιβάλλον διαχείρισης αρχείων πολλαπλών στοιχίσεων (κάποια βέβαια, λειτουργούν και διαδικτυακά ως applets). Ο χρήστης μπορεί να φορτώσει μια πολλαπλή στοίχιση και να την επεξεργαστεί. Κάποια μάλιστα, επικοινωνούν και με προγράμματα πολλαπλής στοίχισης, έτσι ώστε η ίδια η πολλαπλή στοίχιση να γίνει μέσω του περιβάλλοντος αυτού. Η βασική εργασία, την οποία επιτελούν όλα τα προγράμματα, είναι να μορφοποιούν την πολλαπλή στοίχιση σε μορφή κατανοητή από το ανθρώπινο μάτι. Για το σκοπό αυτό, χρωματίζουν διαφορετικά τα διάφορα αμινοξικά κατάλοιπα, συνήθως με το όμοιο χρώμα να δίνεται σε αμινοξέα με παρόμοιες φυσικοχημικές ιδιότητες, ενώ τα περισσότερα επιτρέπουν τον καθορισμό του χρωματισμού. Ο χρωματισμός, είναι πολύ βολικός για το χρήστη, γιατί επιτρέπει να εντοπιστούν συντηρημένες περιοχές με μια γρήγορη επισκόπιση της στοίχισης. Το ίδιο έργο επιτελούν και διάφορα στατιστικά, ανά στήλη της πολλαπλής στοίχισης, τα οποία δίνουν το ποσοστό συντήρησης. Τα περισσότερα από τα προγράμματα αυτά, επιτρέπουν την ταυτόχρονη παράθεση της γνωστής ή προβλεφθείσας δευτεροταγούς δομής, παράλληλα με τη στοίχιση. Η δευτεροταγής δομή, παίζει παρόμοιο ρόλο, καθώς είναι γνωστό ότι οι συντηρημένες περιοχές είναι πιθανότερο να έχουν κάποια συγκεκριμένη δομή (α-έλικα ή β-πτυχωτή επιφάνεια), ενώ οι περιοχές με πολλά κενά πιθανότερο είναι να αντιστοιχούν σε βρόχους. Πολλά άλλα παρόμοια στοιχεία είναι επίσης πιθανό να ενσωματώνονται, όπως οι προγνώσεις διαμεμβρανικών τμημάτων, οι θέσεις γλυκοζυλίωσης ή οι δισουλφιδικοί δεσμοί, τα οποία βοηθούν τον ερευνητή να αποκτήσει μια λεπτομερέστερη εικόνα της βιολογίας των υπό μελέτη των πρωτεϊνών. Δεδομένα από δημόσιες βάσεις, ειδικά από βάσεις δομών όπως η PDB, είναι επίσης πιθανό να ανασύρονται και να αναπαρίστανται παράλληλα με τη στοίχιση, ενώ πολλά εργαλεία δείχνουν επιπλέον και το φυλογενετικό δέντρο των ακολουθιών της στοίχισης (για ακρίβεια, το δέντρο οδηγό). Τέλος, πρέπει να τονιστεί, ότι κάποια από τα εργαλεία αυτά, είναι παραμετροποιήσιμα, δηλαδή επιτρέπουν στον έμπειρο χρήστη να προσθέσει λειτουργικότητες στο πρόγραμμα, διασυνδέοντάς το με επιπλέον προγράμματα πρόγνωσης, εργαλεία στοίχισης ή τοπικές βάσεις δεδομένων, κάνοντας τα με αυτόν τον τρόπο αναπόσπαστο τμήμα της καθημερινής εργασίας που αφορά την ανάλυση πρωτεϊνικών ακολουθιών.



Εικόνα 4.7 *Παραδείγματα λειτουργίας του Jalview. Πάνω, μια πολλαπλή στοίχιση πρωτεϊνικών ακολουθιών και μια πολλαπλή στοίχιση RNA. Κάτω, ένα φυλογενετικό δέντρο, οπτικοποίηση δομών με το Jmol και αναπαράσταση δευτεροταγούς δομής RNA.*

Έχοντας όλα τα παραπάνω υπόψη μας, μπορούμε τέλος, να δούμε εν συντομία τα βασικά βήματα που πρέπει να κάνει κανείς για να προχωρήσει σε μια σωστή πολλαπλή στοίχιση. Συνήθως, οι ακολουθίες προέρχονται από αναζήτηση ομοιότητας σε κάποια βάση δεδομένων, με βάση 1-2 ακολουθίες αναφοράς που έχει ο ερευνητής και πιθανώς τις έχει μελετήσει πειραματικά. Σε μια τέτοια περίπτωση τα ευρήματα πρέπει να ελέγχονται. Για παράδειγμα, μπορεί οι οργανισμοί από τους οποίους προέρχονται να μην έχουν θεωρητικά τέτοιες πρωτεΐνες ή μπορεί η ομοιότητα που εντοπίσαμε να είναι σε μια μόνο μικρή περιοχή. Η γνώση των περιοχών των πρωτεϊνών είναι πολύ σημαντική. Δεν επιχειρούμε, παρά μόνο σε ειδικές περιπτώσεις και με κατάλληλο λογισμικό, να στοιχίσουμε πρωτεΐνες με μεγάλες αποκλίσεις στο μήκος (και άρα, με μεγάλες αποκλίσεις στη σύσταση των περιοχών τους). Αν έχουμε εντοπίσει την περιοχή που χρειαζόμαστε, καλό είναι να πραγματοποιούμε και τις αναζητήσεις μόνο με αυτήν. Σε κάθε περίπτωση, πληροφορία από αντίστοιχες βάσεις πρωτεϊνικών περιοχών (PFAM, PROSITE κλπ), θα είναι πολύ χρήσιμη, και συνίσταται να γίνεται έλεγχος σε αυτές τις βάσεις (το BLAST παρέχει μια τέτοια επιλογή παράλληλα με την αναζήτηση ομοιότητας).



Εικόνα 4.8 *Παραδείγματα λειτουργίας του BioEdit. Το BioEdit εκτός από το ευέυλικτο περιβάλλον που δίνει ο Editor, παρέχει και τα περισσότερα εργαλεία ανάλυσης όπως: αναλύσεις υδοφοβικότητας (στο σχήμα), προγνωστικούς αλγόριθμους, οπτικοποίηση περιοριστικών χαρτών, διασυνδέσεις με πολλές βάσεις δεδομένων, εργαλεία στοίχισης και οπτικοποίησης στοιχίσεων (BLAST, dot plot κλπ), αναζητήσεις προτύπων, αμοιβαία πληροφορία, εργαλεία φυλογενετικής ανάλυσης, αλλά και δυνατότητα προσθήκης και άλλων εργαλείων από το χρήστη.*

Είτε εντοπίσουμε περιοχές ενδιαφέροντος με αυτούς τους τρόπους, είτε όχι, το επόμενο βήμα είναι επίσης σημαντικό. Θα πρέπει να κάνουμε οπωσδήποτε προγνώσεις δευτεροταγούς δομής ή/και άλλων δομικών και λειτουργικών χαρακτηριστικών που ενδέχεται να σχετίζονται με τη συγκεκριμένη πρωτεϊνική οικογένεια. Όπως είδαμε, η δευτεροταγής δομή είναι σημαντική γιατί μπορεί να μας δώσει εικόνα της καταλληλότητας της πολλαπλής στοίχισης, να εντοπίσει ασάφειες κλπ. Τα υπόλοιπα χαρακτηριστικά που πρέπει να προβλέψουμε, μπορεί να ποικίλουν ανάλογα με την περίπτωση. Για παράδειγμα, σε στοιχίσεις εξωκυττάριων πρωτεϊνών, θα ήταν καλό να αφαιρεθεί το πεπτίδιο οδηγητής (signal peptide), και για την ακρίβεια, θα ήταν σημαντικό να ξέρουμε αν έχουν όλες οι πρωτεΐνες μας ένα τέτοιο πεπτίδιο (το ίδιο ισχύει και για όλα τα άλλα προπεπτίδια ή πεπτίδια στόχευσης). Για κάποιες περιπτώσεις, χρήσιμες πληροφορίες οι οποίες μπορεί να είναι χρήσιμες στην πολλαπλή στοίχιση μπορεί να μας δώσουν οι δισουλφιδικοί δεσμοί, αλλά και άλλες θέσεις μετα-μεταφραστικής τροποποίησης (γλυκοζυλίωση, φοσφωρυλίωση κ.ο.κ.). Τούτο συμβαίνει, γιατί οι περιοχές αυτές είναι πιθανό να συντηρούνται στην πολλαπλή στοίχιση, αλλά επιπλέον, με αυτόν τον τρόπο μπορούμε να «διορθώσουμε» τη στοίχιση, αν μια τέτοια θέση έχει τοποθετηθεί λάθος. Το τελευταίο βέβαια, είναι αρκετά επικίνδυνο, και πρέπει να γίνεται με μεγάλη προσοχή γιατί απαιτεί εμπειρία (ενώ πρέπει να θυμόμαστε ότι και οι αλγόριθμοι δεν είναι αλάνθαστοι!). Μια εναλλακτική και ίσως πιο συνετή στρατηγική, είναι, σε περίπτωση εντοπισμού σφάλματος στη στοίχιση να αφαιρούμε την ακολουθία που εμφανίζει το πρόβλημα και να επαναλαμβάνουμε τη στοίχιση χωρίς αυτήν. Κατόπιν, μπορούμε να δοκιμάσουμε να την προσθέσουμε εκ των υστέρων στη στοίχιση με χρήση profile alignment, λειτουργία που υποστηρίζουν τα περισσότερα σύγχρονα εργαλεία.

Βιβλιογραφία

Barton, G. J., & Sternberg, M. J. (1987). A strategy for the rapid multiple alignment of protein sequences. Confidence levels from tertiary structure comparisons. *J Mol Biol, 198*(2), 327-337.

Carrillo, H., & Lipman, D. (1988). The multiple sequence alignment problem in biology. *SIAM Journal on Applied Mathematics 48*(5), 1073-1082.

Corpet, F. (1988). Multiple sequence alignment with hierarchical clustering. *Nucleic Acids Res, 16*(22), 10881-10890.

Durbin, R., Eddy, S., Krogh, A., & Mithison, G. (1998). *Biological sequence analysis, probabilistic models of proteins and nucleic acids.*: Cambridge University Press.

Duret, L., & Abdeddaim, S. (2000). Multiple alignment for structural, functional, or phylogenetic analyses of homologous sequences. *Bioinformatics: Sequence, Structure, and Databanks*, 51-76.

Edgar, R. C. (2004). MUSCLE: a multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics, 5*, 113.

Edgar, R. C., & Sjolander, K. (2004). A comparison of scoring functions for protein sequence profile alignment. *Bioinformatics, 20*(8), 1301-1308.

Feng, D.-F., & Doolittle, R. F. (1987). Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *Journal of molecular evolution, 25*(4), 351-360.

Fitch, W. M., & Margoliash, E. (1967). Construction of phylogenetic trees. *science, 155*(3760), 279-284.

Gonnet, G. H., Cohen, M. A., & Benner, S. A. (1992). Exhaustive matching of the entire protein sequence database. *Science, 256*(5062), 1443-1445.

Gotoh, O. (1996). Significant improvement in accuracy of multiple protein sequence alignments by iterative refinement as assessed by reference to structural alignments. *J Mol Biol, 264*(4), 823-838.

Higgins, D. G., Bleasby, A. J., & Fuchs, R. (1992). CLUSTAL V: improved software for multiple sequence alignment. *Computer applications in the biosciences: CABIOS, 8*(2), 189-191.

Kim, J., Pramanik, S., & Chung, M. J. (1994). Multiple sequence alignment using simulated annealing. *Comput Appl Biosci, 10*(4), 419-426.

Lassmann, T., & Sonnhammer, E. L. (2005). Kalign--an accurate and fast multiple sequence alignment algorithm. *BMC Bioinformatics, 6*, 298.

Lipman, D. J., Altschul, S. F., & Kececioglu, J. D. (1989). A tool for multiple sequence alignment. *Proceedings of the National Academy of Sciences, 86*(12), 4412-4415.

Magis, C., Taly, J. F., Bussotti, G., Chang, J. M., Di Tommaso, P., Erb, I., . . . Notredame, C. (2014). T-Coffee: Tree-based consistency objective function for alignment evaluation. *Methods Mol Biol, 1079*, 117-129.

Morgenstern, B. (2014). Multiple sequence alignment with DIALIGN. *Methods Mol Biol, 1079*, 191-202.

Notredame, C., & Higgins, D. G. (1996). SAGA: sequence alignment by genetic algorithm. *Nucleic Acids Res, 24*(8), 1515-1524.

O'Sullivan, O., Zehnder, M., Higgins, D., Bucher, P., Grosdidier, A., & Notredame, C. (2003). APDB: a novel measure for benchmarking sequence alignment methods without reference alignments. *Bioinformatics, 19 Suppl 1*, i215-221.

Pais, F. S., Ruy Pde, C., Oliveira, G., & Coimbra, R. S. (2014). Assessing the efficiency of multiple sequence alignment programs. *Algorithms Mol Biol, 9*(1), 4.

Papadopoulos, J. S., & Agarwala, R. (2007). COBALT: constraint-based alignment tool for multiple protein sequences. *Bioinformatics, 23*(9), 1073-1079.

Raghava, G. P., Searle, S. M., Audley, P. C., Barber, J. D., & Barton, G. J. (2003). OXBench: a benchmark for evaluation of protein multiple sequence alignment accuracy. *BMC Bioinformatics, 4*, 47.

Roshan, U. (2014). Multiple sequence alignment using Probcons and Probalign. *Methods Mol Biol, 1079*, 147-153.

Saitou, N., & Nei, M. (1987). The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular biology and evolution, 4*(4), 406-425.

Simossis, V. A., & Heringa, J. (2005). PRALINE: a multiple sequence alignment toolbox that integrates homology-extended and secondary structure information. *Nucleic Acids Res, 33*(Web Server issue), W289-294.

Thompson, J. D., Gibson, T. J., & Higgins, D. G. (2002). Multiple sequence alignment using ClustalW and ClustalX. *Curr Protoc Bioinformatics, Chapter 2*, Unit 2 3.

Thompson, J. D., Higgins, D. G., & Gibson, T. J. (1994). CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic acids research, 22*(22), 4673-4680.

Thompson, J. D., Linard, B., Lecompte, O., & Poch, O. (2011). A comprehensive benchmark study of multiple sequence alignment methods: current challenges and future perspectives. *PLoS One, 6*(3), e18093.

Thompson, J. D., Plewniak, F., & Poch, O. (1999). A comprehensive comparison of multiple sequence alignment programs. *Nucleic Acids Res, 27*(13), 2682-2690.

Van Walle, I., Lasters, I., & Wyns, L. (2005). SABmark--a benchmark for sequence alignment that covers the entire known fold space. *Bioinformatics, 21*(7), 1267-1268.

Wallace, I. M., O'Sullivan, O., & Higgins, D. G. (2005). Evaluation of iterative alignment algorithms for multiple alignment. *Bioinformatics, 21*(8), 1408-1414.

Wang, G., & Dunbrack, R. L., Jr. (2004). Scoring profile-to-profile sequence alignments. *Protein Sci, 13*(6), 1612-1626.

Waterman, M. S. (1995). *Introduction to Computational Biology*: Chapman and Hall, London.

Wu, S., & Manber, U. (1992). Fast text searching allowing errors. *Communications of the ACM 35*(10), 83-91.

Παράρτημα

Οι πιο γνωστές μορφές αρχείων πολλαπλής στοίχισης

Multi-FASTA

```
>sw:CD5R_BOVIN Q28199 Cyclin-dependent kinase 5 activator 1 precursor
MGTVLSLSPSYRKATLFEDGAATVGHYTAVQNSKNAKDKNLKRHSIISVLPWKRIVAVSA
KKKNSKKVQPNSSYQNNITHLNNENLKKSLSCANLSTFAQPPPAQPPAPPASQLSGSQTG
VSSSVKKAPHPAVSSAGTPKRVIVQASTSELLRCLGEFLCRRCYRLKHLSPTDPVLWLRS
VDRSLLLQGWQDQGFITPANVVFLYMLCRDVISSEVGSDHELQAVLLTCLYLSYSYMGNE
ISYPLKPFLVESCKEAFWDRCLSVINLMSSKMLQINADPHYFTQVFSDLKNESGQEDKKR
LLLGLDR
>sw:CD5R_HUMAN Q15078 Cyclin-dependent kinase 5 activator 1 precursor
MGTVLSLSPSYRKATLFEDGAATVGHYTAVQNSKNAKDKNLKRHSIISVLPWKRIVAVSA
KKKNSKKVQPNSSYQNNITHLNNENLKKSLSCANLSTFAQPPPAQPPAPPASQLSGSQTG
GSSSVKKAPHPAVTSAGTPKRVIVQASTSELLRCLGEFLCRRCYRLKHLSPTDPVLWLRS
VDRSLLLQGWQDQGFITPANVVFLYMLCRDVISSEVGSDHELQAVLLTCLYLSYSYMGNE
ISYPLKPFLVESCKEAFWDRCLSVINLMSSKMLQINADPHYFTQVFSDLKNESGQEDKKR
LLLGLDR
>sw:CD5R_MOUSE Q62938 Cyclin-dependent kinase 5 activator 1 precursor
MGTVLSLSPSYRKATLFEDGAATVGHYTAVQNSKNAKDKNLKRHSIISVLPWKRIVAVSA
KKKNSKKAQPNSSYQSNIAHLNNENLKKSLSCANLSTFAQPPPAQPPAPPASQLSGSQTG
VSSSVKKAPHPAITSAGTPKRVIVQASTSELLRCLGEFLCRRCYRLKHLSPTDPVLWLRS
VDRSLLLQGWQDQGFITPANVVFLYMLCRDVISSEVGSDHELQAVLLTCLYLSYSYMGNE
ISYPLKPFLVESCKEAFWDRCLSVINLMSSKMLQINADPHYFTQVFSDLKNESGQEDKKR
LLLGLDR
```

MSF

```
MSF: 307 Type: P Check: 4977 ..
Name: CD5R_BOVIN oo Len: 307 Check: 5281 Weight: 33.3
Name: CD5R_HUMAN oo Len: 307 Check: 5196 Weight: 33.3
Name: CD5R_MOUSE oo Len: 307 Check: 4500 Weight: 33.3
//
CD5R_BOVIN MGTVLSLSPS YRKATLFEDG AATVGHYTAV QNSKNAKDKN LKRHSIISVL
CD5R_HUMAN MGTVLSLSPS YRKATLFEDG AATVGHYTAV QNSKNAKDKN LKRHSIISVL
CD5R_MOUSE MGTVLSLSPS YRKATLFEDG AATVGHYTAV QNSKNAKDKN LKRHSIISVL
CD5R_BOVIN PWKRIVAVSA KKKNSKKVQP NSSYQNNITH LNNENLKKSL SCANLSTFAQ
CD5R_HUMAN PWKRIVAVSA KKKNSKKVQP NSSYQNNITH LNNENLKKSL SCANLSTFAQ
CD5R_MOUSE PWKRIVAVSA KKKNSKKAQP NSSYQSNIAH LNNENLKKSL SCANLSTFAQ
CD5R_BOVIN PPPAQPPAPP ASQLSGSQTG VSSSVKKAPH PAVSSAGTPK RVIVQASTSE
CD5R_HUMAN PPPAQPPAPP ASQLSGSQTG GSSSVKKAPH PAVTSAGTPK RVIVQASTSE
CD5R_MOUSE PPPAQPPAPP ASQLSGSQTG VSSSVKKAPH PAITSAGTPK RVIVQASTSE
```

```
CD5R_BOVIN LLRCLGEFLC RRCYRLKHLS PTDPVLWLRS VDRSLLLQGW QDQGFITPAN
CD5R_HUMAN LLRCLGEFLC RRCYRLKHLS PTDPVLWLRS VDRSLLLQGW QDQGFITPAN
CD5R_MOUSE LLRCLGEFLC RRCYRLKHLS PTDPVLWLRS VDRSLLLQGW QDQGFITPAN
CD5R_BOVIN VVFLYMLCRD VISSEVGSDH ELQAVLLTCL YLSYSYMGNE ISYPLKPFLV
CD5R_HUMAN VVFLYMLCRD VISSEVGSDH ELQAVLLTCL YLSYSYMGNE ISYPLKPFLV
CD5R_MOUSE VVFLYMLCRD VISSEVGSDH ELQAVLLTCL YLSYSYMGNE ISYPLKPFLV
CD5R_BOVIN ESCKEAFWDR CLSVINLMSS KMLQINADPH YFTQVFSDLK NESGQEDKKR
CD5R_HUMAN ESCKEAFWDR CLSVINLMSS KMLQINADPH YFTQVFSDLK NESGQEDKKR
CD5R_MOUSE ESCKEAFWDR CLSVINLMSS KMLQINADPH YFTQVFSDLK NESGQEDKKR
CD5R_BOVIN LLLGLDR
CD5R_HUMAN LLLGLDR
CD5R_MOUSE LLLGLDR
```

CLUSTAL

```
CLUSTAL W (1.82) multiple sequence alignment
CD5R_BOVIN MGTVLSLSPSYRKATLFEDGAATVGHYTAVQNSKNAKDKNLKRHSIISVLPWKRIVAVSA
CD5R_HUMAN MGTVLSLSPSYRKATLFEDGAATVGHYTAVQNSKNAKDKNLKRHSIISVLPWKRIVAVSA
CD5R_MOUSE MGTVLSLSPSYRKATLFEDGAATVGHYTAVQNSKNAKDKNLKRHSIISVLPWKRIVAVSA
           ************************************************************
CD5R_BOVIN KKKNSKKVQPNSSYQNNITHLNNENLKKSLSCANLSTFAQPPPAQPPAPPASQLSGSQTG
CD5R_HUMAN KKKNSKKVQPNSSYQNNITHLNNENLKKSLSCANLSTFAQPPPAQPPAPPASQLSGSQTG
CD5R_MOUSE KKKNSKKAQPNSSYQSNIAHLNNENLKKSLSCANLSTFAQPPPAQPPAPPASQLSGSQTG
           *******.*******.**:*****************************************
CD5R_BOVIN VSSSVKKAPHPAVSSAGTPKRVIVQASTSELLRCLGEFLCRRCYRLKHLSPTDPVLWLRS
CD5R_HUMAN GSSSVKKAPHPAVTSAGTPKRVIVQASTSELLRCLGEFLCRRCYRLKHLSPTDPVLWLRS
CD5R_MOUSE VSSSVKKAPHPAITSAGTPKRVIVQASTSELLRCLGEFLCRRCYRLKHLSPTDPVLWLRS
           ***********::***************************************************
CD5R_BOVIN VDRSLLLQGWQDQGFITPANVVFLYMLCRDVISSEVGSDHELQAVLLTCLYLSYSYMGNE
CD5R_HUMAN VDRSLLLQGWQDQGFITPANVVFLYMLCRDVISSEVGSDHELQAVLLTCLYLSYSYMGNE
CD5R_MOUSE VDRSLLLQGWQDQGFITPANVVFLYMLCRDVISSEVGSDHELQAVLLTCLYLSYSYMGNE
           ************************************************************
CD5R_BOVIN ISYPLKPFLVESCKEAFWDRCLSVINLMSSKMLQINADPHYFTQVFSDLKNESGQEDKKR
CD5R_HUMAN ISYPLKPFLVESCKEAFWDRCLSVINLMSSKMLQINADPHYFTQVFSDLKNESGQEDKKR
CD5R_MOUSE ISYPLKPFLVESCKEAFWDRCLSVINLMSSKMLQINADPHYFTQVFSDLKNESGQEDKKR
           ************************************************************
CD5R_BOVIN LLLGLDR
CD5R_HUMAN LLLGLDR
CD5R_MOUSE LLLGLDR
           *******
```