ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ

Μεταπτυχιακό Πρόγραμμα Σπουδών στην Εφαρμοσμένη Πληροφορική

Εφαρμογή Αλγορίθμων ακέραιου και γραμμικού προγραμματισμού στη Βιοπληροφορική.

Διπλωματική Εργασία

Επιβλέπων Καθηγητής: Σαμαράς Νικόλαος.

Επιμέλεια εργασίας: Λένης Βασίλειος- Παναγιώτης.

Θεσσαλονίκη, Οκτώβριος 2012

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Μακεδονίας.

Περίληψη.

Η Βιοπληροφορική αποτελεί ένα σύγχρονο τομέα έρευνας και ανάπτυξης τόσο για τους μοριακούς βιολόγους όσο και για τους επιστήμονες της πληροφορικής. Η συνεργασία των δύο αυτών επιστημών χαρακτηρίζεται αρκετά υποσχόμενη και με ιδιαίτερη σημασία αφού έρχεται να ρίξει φως στην ερμηνεία και το ρόλο της γονιδιακής πληροφορίας και κατ' επέκταση σε αρκετές διαδικασίες της ζωής που ζητούν ερμηνεία.

Στην παρούσα εργασία παρουσιάζονται τα σημαντικότερα προβλήματα που απασχολούν τη συγκεκριμένη επιστήμη με μεγαλύτερη έμφαση στο πρόβλημα της ολικής κατά ζεύγη στοίχισης, το οποίο αποτελεί αναμφίβολα το πιο συνηθισμένο εργαλείο στο χώρο της Βιοπληροφορικής. Είναι ένα απαραίτητο εργαλείο για την ανακάλυψη λειτουργικών, δομικών και εξελικτικών πληροφοριών σε βιολογικές αλληλουχίες. Ακόμη και τα τελευταία χρόνια αποτελεί πρόκληση στην δημιουργία αλγορίθμων και την παραγωγή λογισμικών η εύρεση τρόπων στοίχισης αλληλουχιών υψηλής ποιότητας σε λογικά χρονικά πλαίσια.

Στη συνέχεια παρουσιάζεται ενδελεχώς η μέθοδος του γραμμικού προγραμματισμού και ακεραίου γραμμικού προγραμματισμού με παραδείγματα για την καλύτερη κατανόησή τους, καθώς και ένα μαθηματικό μοντέλο ακεραίου γραμμικού προγραμματισμού για την επίλυση του προβλήματος της στοίχισης κατά ζεύγη.

Σκοπός της παρούσας εργασίας είναι να δειχθεί πως μέθοδοι της επιχειρησιακής έρευνας μπορούν με την κατάλληλη προσαρμογή να επιλύσουν προβλήματα της επιστήμης της βιοπληροφορικής με μεγάλη αποτελεσματικότητα.

Λέξεις κλειδιά: Βιοπληροφορική, στοίχιση κατά ζεύγη, ολική στοίχιση, γραμμικός προγραμματισμός, ακέραιος γραμμικός προγραμματισμός, μαθηματικό μοντέλο, ILOG CPLEX.

Abstract.

Bioinformatics constitutes a modern research and development field both for molecular biologists and scientists of information technology. The collaboration of these two sciences is characterized enough promising and with particular importance as it illuminates the interpretation and the role of gene information and further more decrypts enough processes of life.

This paper presents the most important problems in this field with emphasis on the problem of total pairwise alignment, which undoubtedly is the most common tool in the field of Bioinformatics. It is an essential tool for discovering functional, structural and evolutionary information in biological sequences. Even in recent years, it remains challenging the generation of algorithms and software programs that ensure high quality alignments in reasonable time.

Continuing, we present in details the method of linear programming and integer linear programming with examples for better understanding and a mathematical integer linear programming model for solving the pairwise sequence alignment problem.

The aim of this paper is to show how methods of operations research can solve problems of the field of bioinformatics with great efficiency.

Keywords: Bioinformatics, pairwise sequence alignment, global sequence alignment, linear programming, integer linear programming, mathematical model, ILOG CPLEX.

Περιεχόμενα

ΚΕΦΑΛΑΙΟ 1: Εισαγωγή

1.1. Εισαγωγή – Ορισμοί

Πριν μερικά χρόνια, για έναν βιολόγο η βιοπληροφορική ως μια νέα επιστήμη αποτελούσε κάτι το οξύμωρο. Η επιστημονική κοινότητα, δεν μπορούσε να φανταστεί ότι για την μελέτη και την αξιοποίηση των πληροφοριών που παράγονται και μεταδίδονται σε ένα βιολογικό σύστημα δεν αρκούν οι χειρόγραφες σημειώσεις και ο υπολογισμός με χαρτί και μολύβι των αποτελεσμάτων που προκύπτουν από τη διεξαγωγή των πειραμάτων. Η ανάγκη εύρεσης προηγμένων υπολογιστικών μεθόδων, ο σχεδιασμός και η υλοποίηση πολύπλοκων αλγορίθμων και εν τέλη η επιβολή της χρήσης ηλεκτρονικού υπολογιστή φάνταζε κάτι μακρινό και ίσως απίθανο.

Σαφώς, οι εποχές άλλαξαν. Με την πάροδο των χρόνων, παρατηρείται μια ραγδαία αύξηση των δεδομένων που συλλέγει η μοριακή βιολογία. Για παράδειγμα, τον Απρίλιο του 2012, η GenBank (βάση δεδομένων νουκλεοτιδικών αλληλουχιών) αυξήθηκε κατά 15,4% έχοντας συνολικά 151,8 εκατομμύρια εγγραφές για πάνω από 800 οργανισμούς και συμβάλει με πάνω από 139,3 δισεκατομμύρια νουκλεικές βάσεις.

Η uniProt (βάση δεδομένων πρωτεϊνικών ακολουθιών) περιέχει πάνω από 22,08 εκατομμύρια καταχωρήσεις πρωτεϊνικών ακολουθιών.

Έχει διαπιστωθεί πως κατά μέσο όρο οι συγκεκριμένες βάσεις δεδομένων διπλασιάζουν το μέγεθός τους κάθε 15 μήνες.

Επί πρόσθετα, εάν λάβουμε υπόψη το πλήθος των έργων που σχετίζονται με τη μελέτη της γονιδιακής έκφρασης, τον προσδιορισμό των πρωτεϊνικών δομών που κωδικοποιούνται από τα γονίδια, και την προσπάθεια κατανόησης των αλληλοεπιδράσεων μεταξύ των παραγόμενων

προϊόντων, μπορούμε να φανταστούμε την τεράστια ποσότητα πληροφοριών που παράγεται.

Η χρήση του ηλεκτρονικού υπολογιστή στη σύγχρονη μοριακή βιολογία αποτελεί πλέον επιτακτική ανάγκη.

Η αναγκαστική συνύπαρξη των δύο αυτών επιστημών (πληροφορική-βιολογία) δημιούργησε ένα νέο πεδίο ερευνών, τη βιοπληροφορική.

Υπάρχει μια μεγάλη ποικιλία ορισμών στην βιβλιογραφία και στον παγκόσμιο ιστό για το τι είναι η βιοπληροφορική. Παραθέτουμε ενδεικτικά παρακάτω τους σημαντικότερους.

(Μοριακή) βιο- πληροφορική: είναι η σύλληψη εννοιών της βιολογίας σε μοριακούς όρους (με βάση τη φυσική και τη χημεία) εφαρμόζοντας τεχνικές της πληροφορικής (που προέρχονται από κλάδους όπως τα εφαρμοσμένα μαθηματικά, η στατιστική και η επιστήμη των υπολογιστών) με σκοπό την οργάνωση και την κατανόηση των πληροφοριών των μορίων σε μεγάλη κλίμακα. Εν ολίγοις, η βιοπληροφορική είναι ένα σύστημα διαχείρισης πληροφοριών για τη μοριακή βιολογία που έχει πολλές πρακτικές εφαρμογές (Oxford English Dictionary).

«Βιοπληροφορική: έρευνα, ανάπτυξη, ή εφαρμογή υπολογιστικών εργαλείων για την απόκτηση, αποθήκευση, οργάνωση, αρχειοθέτηση, ανάλυση και απεικόνιση βιολογικών και ιατρικών δεδομένων.» (NIH Biomedical Information Science and Technology)

«Μαθηματικοί, στατιστικοί και πληροφορικοί μέθοδοι που αποσκοπούν στην επίλυση προβλημάτων με τη χρήση βιολογικών και DNA ακολουθιών, αμινοξέων και πληροφορίες που σχετίζονται με αυτά.» (Fredj Tekaia)

«Η βιοπληροφορική περιλαμβάνει την τεχνολογία που χρησιμοποιεί υπολογιστές για αποθήκευση, ανάκτηση, διαχείριση και διανομή των πληροφοριών που σχετίζονται με βιολογικά μακρομόρια, όπως το DNA, RNA και πρωτεΐνες.» (Luscombe, et al.)

«Βιοπληροφορική είναι το επιστημονικό πεδίο, στο οποίο βιολογία, επιστήμη των υπολογιστών και τεχνολογία των πληροφοριών συγχωνεύονται σε ένα ενιαίο πεδίο. υπάρχουν τρείς σημαντικοί υπό-κλάδοι στη

βιοπληροφορική: η ανάπτυξη νέων αλγορίθμων και στατιστικών εφαρμογών με τα οποία μπορούν να αξιολογηθούν οι σχέσεις μεταξύ μεγάλων όγκων δεδομένων. Η ανάλυση και ερμηνεία διαφόρων τύπων δεδομένων, συμπεριλαμβανομένων ακολουθιών νουκλεοτιδίων και αμινοξέων, πρωτεϊνών βάσεων, καθώς και πρωτεϊνικών δομών. Και η ανάπτυξη και εφαρμογή εργαλείων που επιτρέπουν την αποτελεσματική πρόσβαση και διαχείριση των διαφορετικών τύπων των πληροφοριών.» (National Center for Biotechnology Information, 2001)

«Η επιστήμη της Βιοπληροφορικής είναι συνένωση των επιστημών της Μοριακής Βιολογίας και της Πληροφορικής. Έχει ως αντικείμενο της την βέλτιστη συνεργασία τους για το καλό (ελπίζουμε) της ανθρωπότητας, πρακτικά στην αποθήκευση, επεξεργασία και ανάλυση των βιολογικών δεδομένων.» (Dr. Temple Smith)

Στο σημείο αυτό, θα ήταν χρήσιμο να αναφερθούμε στη λεπτή αλλά σημαντική διαφορά μεταξύ της Βιοπληροφορικής (bioinformatics) και της υπολογιστικής βιολογίας (computational biology).

Το αντικείμενο έρευνας της Βιοπληροφορικής περιορίζεται στην ανάλυση ακολουθιών, δομών και λειτουργιών των γονιδίων, γονιδιομάτων και των αντίστοιχων προϊόντων τους. Ποιο αυστηρά, θα μπορούσαμε να την ονομάσουμε υπολογιστική μοριακή βιολογία. Από την άλλη, η υπολογιστική βιολογία περιλαμβάνει όλα τα βιολογικά πεδία που ασχολούνται με τον υπολογισμό. Για παράδειγμα, μαθηματική μοντελοποίηση των οικοσυστημάτων, δυναμική πληθυσμού, εφαρμογή της θεωρίας των παιγνίων σε μελέτες συμπεριφοράς και φυλογενετική κατασκευή χρησιμοποιώντας απολιθώματα. Σε όλες αυτές τις περιπτώσεις χρησιμοποιούνται όλα τα υπολογιστικά εργαλεία, χωρίς όμως αυτό να συνεπάγεται την εμπλοκή βιολογικών μακρομορίων.

Μια άλλη προσέγγιση, εντοπίζει τη διαφορά των δύο κλάδων στο ότι η βιοπληροφορική ασχολείται με την ανάπτυξη και την εφαρμογή των υπολογιστικών εργαλείων για τη διαχείριση των βιολογικών δεδομένων, ενώ η υπολογιστική βιολογία παρέχει περισσότερο το θεωρητικό υπόβαθρο σχεδιάζοντας αλγορίθμους που χρησιμοποιούνται στη βιοπληροφορική.

Παρόλο που τα δύο αυτά επιστημονικά πεδία εννοιολογικά είναι διαχωρισμένα, στην πράξη σε μεγάλο βαθμό παρατηρείται μια αλληλοεπικάλυψη.

Ως αποτέλεσμα βλέπουμε πως στο πεδίο της έρευνας οι δύο αυτοί όροι χρησιμοποιούνται εναλλάξ. Ο λόγος ύπαρξης αυτής της σύγχυσης είναι ακριβώς ο διεπιστημονικός χαρακτήρας της Βιοπληροφορικής. Αποτελεί το σταυροδρόμι της βιολογίας, της επιστήμης των υπολογιστών και της τεχνολογίας πληροφοριών με αποτέλεσμα οι διαφορετικοί αντιπρόσωποι του πεδίου να παραθέτουν διαφορετικές απόψεις για το σκοπό και το ρόλο της.

Ας μην επεκταθούμε όμως περισσότερο στις διαφορές των δύο επιστημονικών πεδίων, αλλά ας εστιάσουμε στην κεντρική ομοιότητά τους. Την αναγκαία είσοδο της πληροφορικής στη βιολογία.

1.2. Ιστορική Αναδρομή

Οι πρώτες προσπάθειες στο χώρο τις Βιοπληροφορικής σημειώνονται τη δεκαετία του ‘60, παρόλο που ο όρος βιοπληροφορική ακόμη δεν υπήρχε.

Η Margaret Dayhoff δημιούργησε την πρώτη βάση πρωτεϊνικών ακολουθιών με το όνομα Atlas of Protein Sequence and Structure (Atlas of Protein Sequence and Structure, 1954–1965). Περιείχε 1660 πρωτεΐνες οι οποίες φυσικά ήταν καταγεγραμμένες στο χαρτί. Χρειάστηκαν να περάσουν περίπου 20 χρόνια, ώστε το έτος 1984 ο Άτλας της Dayhoff να κυκλοφορήσει σε ηλεκτρονική μορφή (Paul G. Higgs,Teresa K, 2005).

Έπειτα, αρχές της δεκαετίας του ’70, το Brookhaven National Laboratory καθιέρωσε την πρώτη πρωτεϊνική τράπεζα δεδομένων αρχειοθετώντας τρισδιάστατες πρωτεϊνικές δομές (Helen M. Berman, et al., 1999).

Στην αρχή, η εξέλιξη της Βιοπληροφορικής προχωρούσε με αργούς αλλά σταθερούς ρυθμούς.

Ο πρώτος αλγόριθμος για τη στοίχιση ακολουθιών, ένας αλγόριθμος δυναμικού προγραμματισμού που δημιουργήθηκε το 1971 από τους Needleman και Wunsch έδωσε μια νέα ώθηση στη δημιουργία

αποτελεσματικότερων τρόπων σύγκρισης αλληλουχιών με βάσεις δεδομένων (Needleman και Wunsch, 1071).

Το 1974 οι Chou και Fasman δημιούργησαν τον πρώτο αλγόριθμο πρόβλεψης πρωτεϊνικής δομής που αν και θεωρείται σήμερα ξεπερασμένος, έπαιξε πρωταγωνιστικό ρόλο σε μια σειρά εξελίξεων στην πρόβλεψη της δομής των πρωτεϊνών (Peter Y**.** Chou, Gerald D. Fasman, 1974).

Τρία χρόνια μετά, δημιουργήθηκε το πρώτο πρόγραμμα ηλεκτρονικού υπολογιστή για την αλληλουχία του DNA, και μπορούσε να χρησιμοποιηθεί αποτελεσματικά για τη συναρμολόγηση ακολουθιών DNA (R.Staden, 1977).

Η πρώτη απόπειρα ορισμού της βιοπληροφορικής έγινε το 1978 από τον Hogeweg, που όρισε τη βιοπληροφορική ως *'τη μελέτη διαδικασιών πληροφόρησης σε βιοτικά συστήματα'* (Hogeweg, 1978)

Το 1981 εδραιώνεται στο χώρο της Βιοπληροφορικής η έννοια του μοτίβου της ακολουθίας η οποία θα δώσει έναυσμα για τη δημιουργία πολλών αλγορίθμων εύρεσης μοτίβου και σύγκρισης ακολουθιών (Doolittle, 1981).

Την ίδια χρονιά υλοποιείται ο αλγόριθμος Smith- Waterman. Ένας αλγόριθμος στοίχισης ακολουθιών, δυναμικού προγραμματισμού όπου βρίσκει τη βέλτιστη στοίχιση σε περιοχές με χαμηλή ομοιότητα μεταξύ μακρινής συγγένεια βιολογικών ακολουθιών (Smith Waterman, 1981).

Αν και ο αλγόριθμος Smith- Waterman είναι ένας αργός αλγόριθμος με μεγάλες απαιτήσεις σε μνήμη, αποτέλεσε το κίνητρο για την εύρεση αποτελεσματικότερων αλγορίθμων για τη στοίχιση μακρομοριακών ακολουθιών.

Η δημοσίευση της διαδικασίας της αλυσιδωτής αντίδρασης της πολυμεράσης (PCR) το 1986 από τον Mullis και τους συνεργάτες του αντιπροσώπευσε ένα ορόσημο στη μοριακή βιολογία και, ταυτόχρονα, τη βιοπληροφορική (Mullis et al., 1986).

Την ίδια χρονιά ιδρύθηκε η βάση δεδομένων SWISS- PROT και ο Thomas Roderick επινόησε τον όρο του γονιδιώματος (genomics) περιγράφοντας την έννοια της αλληλουχίας σε ολόκληρα γονιδιώματα. Δύο χρόνια αργότερα, ιδρύθηκε το Εθνικό κέντρο πληροφοριών βιοτεχνολογίας

(NCBI) στο οποίο σήμερα λειτουργεί μια από τις μεγαλύτερες και σημαντικότερες βάσεις δεδομένων (www.ncbi.nlm.nih.gov).

Η ανάπτυξη γρήγορων αλγορίθμων αναζήτησης στα τέλη της δεκαετίας του ’80 όπως ο FASTA από τον William Pearson (William R. Pearson, et al, 1988) και ο BLAST από τον Stephen Altschul και τους συνεργάτες του (Stephen F. Altschul et al., 1990), καθώς και η έναρξη του έργου του ανθρώπινου γονιδιώματος, έδωσαν μια σημαντική ώθηση στην ανάπτυξη της Βιοπληροφορικής.

Η ραγδαία εξέλιξη της υπολογιστικής ισχύος καθώς και η όλο και ευκολότερη πρόσβαση στο internet που προσέφερε η δεκαετία του ’90 κατέστησε την συγκέντρωση και ανάλυση βιολογικών δεδομένων πιο εφικτή από ποτέ (Christos A. Ouzounis, 2003).

Έχουμε την εμφάνιση των πρώτων εξελιγμένων προγραμμάτων πρόβλεψης γονιδίων το 1991 (Brunak *et al*., 1990), καθώς και την δημιουργία αλγορίθμων για την πρόβλεψη πρωτεϊνικών δομών (Rost and Sander, 1993).

Η παραγωγή μικροσυστοιχιών δεδομένων DNA (DNA micro array data), μικροσυστοιχίες που αποτελούνται από χιλιάδες ολιγονουκλεοτίδια DNA βιολογικών δειγμάτων, βελτίωσε τη μελέτη της γονιδιακής έκφρασης και πυροδότησε την είσοδο της Βιοπληροφορικής στην ιατρική όπου μέχρι και σήμερα παρατηρούμε εξελίξεις σε καθημερινή βάση. (Richard Simon et al., 2002) .

Λαμβάνοντας υπόψη τους παραπάνω σημαντικούς σταθμούς που έπαιξαν καταλυτικό ρόλο στην εξέλιξη της Βιοπληροφορικής, καταλήγουμε σε δύο συμπεράσματα. Πρώτον, ότι είναι σημαντικό τόσο να εκτιμήσουμε όσο και να κατανοήσουμε τα πρώτα βήματα που έγιναν από κάποιους πρωτοπόρους σε έναν άγνωστο μέχρι τότε επιστημονικό πεδίο όπου στη συνέχεια εδραιώνεται και επηρεάζει άμεσα τις εξελίξεις της επιστήμης της βιολογίας, και δεύτερον αυτή η πορεία της ιστορίας μας αποδεικνύει την εξέλιξη καθώς και την εδραίωση της Βιοπληροφορικής ως ένα ανεξάρτητο επιστημονικό πεδίο με τα δικά του προβλήματα, τις δικές του κατευθύνεις και προοπτικές.

Η βιοπληροφορική έγινε ένας ανεξάρτητος επιστημονικός κλάδος τόσο παλιός, όσο και η επιστήμη των υπολογιστών (Christos A. Ouzounis, 2003).

1.3. Στόχοι της Βιοπληροφορικής

Ο απώτερος στόχος της Βιοπληροφορικής είναι η καλύτερη κατανόηση της λειτουργίας ενός ζωντανού κυττάρου σε μοριακό επίπεδο.

Με την ανάλυση μοριακών ακολουθιών και μοριακών δομών δεδομένων, μπορεί να επιτευχθεί μια νέα, πληρέστερη προσέγγιση της εικόνας του κυττάρου και κατ' επέκταση της ίδιας της ζωής.

Η αποκάλυψη και η ανάλυση της πληθώρας βιολογικών πληροφοριών που κρύβονται στη μάζα ακολουθιών, και στις δομές βιολογικών δεδομένων μπορούν να μας ανοίξουν το δρόμο για την απόκτηση μιας σαφέστερης εικόνας για τη θεμελιώδη βιολογία των οργανισμών και να μας καθοδηγήσουν στην δυνατότητα αξιοποίησης αυτών των πληροφοριών για την ενίσχυση του επιπέδου ζωής της ανθρωπότητας (European Bioinformatics Institute (EBI)).

Η συμβολή της Βιοπληροφορικής στην ανάλυση των δεδομένων της γονιδιακής έκφρασης είναι καθοριστική. Το γονίδιο αποτελεί τη βασική μονάδα κληρονομικότητας σε όλους τους ζωντανούς οργανισμούς.

Οι αλυσιδωτές χημικές αντιδράσεις, τα πρωτεϊνικά μονοπάτια ή γενικότερα τα δίκτυα βιομορίων, είναι υπεύθυνα για την ανάπτυξη, την επιβίωση, τη διαίρεση και όλες της ζωτικής σημασίας λειτουργίες του κυττάρου και κατά συνέπεια ολόκληρου του ζωντανού οργανισμού. (Baxevanis et al., 2001) Στην τεκμηρίωση, τον έλεγχο και την τροποποίηση αυτής της διαδικασίας, (η οποία αποτελεί και το κεντρικό δόγμα της μοριακής βιολογίας) επικεντρώνεται η βιοπληροφορική.

Αποτελεί επίσης τη βάση για τη γενετική μηχανική, την χαρτογράφηση του ανθρώπινου γονιδιώματος και τη διάγνωση και θεραπεία των γενετικών παθήσεων.

Για παράδειγμα, με την τροποποίηση της παραπάνω διαδικασίας θα μπορούν να παραχθούν νέες πρωτεΐνες οι οποίες με τη σειρά τους θα αποτελέσουν τη βάση για την παρασκευή νέων φαρμάκων που δεν θα βασίζονται πλέων μόνο στα συμπτώματα των ασθενειών αλλά κατά κύριο λόγο στο γενετικό υπόβαθρο του κάθε ανθρώπου ατομικά (Bryan Bergeron, 2002).

Συνοψίζοντας θα μπορούσαμε να πούμε πως κύριος στόχος της Βιοπληροφορικής αποτελεί η ανάπτυξη εργαλείων που βοηθούν στην οργάνωση, την ανάλυση δεδομένων, και την ερμηνεία αποτελεσμάτων με σκοπό την εκπόνηση σημαντικής βιολογικής γνώσης.

1.4. Μοριακή Βιολογία – Βασικά στοιχεία

Η έννοια της ζωής είναι από τη φύση της πολυσύνθετη. Διάφορες επιστήμες την προσεγγίζουν και ασχολούνται μαζί της από διαφορετικές οπτικές γωνίες. Η κοινωνιολογία με γνώμονα τις διαδικασίες συμβίωσης των ανθρωπίνων οργανισμών, η φιλοσοφία ασχολείται κυρίως με τις πνευματική και άυλη υφή της, η βιολογία με τη μελέτη της δημιουργίας της.

Όλοι οι ζωντανοί οργανισμοί διαφέρουν μεταξύ τους. Ακόμη και οργανισμοί που ανήκουν στο ίδιο είδος έχουν τεράστιες διαφορές. Όσο όμως προχωράμε στο μοριακό τους επίπεδο παρατηρούμε απίστευτες ομοιότητες.

Ένα κοινό χαρακτηριστικό όλων των ζωντανών οργανισμών σε μοριακό επίπεδο, είναι η ύπαρξη μιας οικογένειας μορίων, τις πρωτεΐνες. Στις πρωτεΐνες οφείλονται όλες οι βιοχημικές διεργασίες για τη ζωή ενός οργανισμού. Η κάθε πρωτεΐνη επιτελεί τη δική της ξεχωριστή εργασία. Παρόλα αυτά, παρατηρείται πως πρωτεΐνες που μοιάζουν πολύ μεταξύ τους σε δομή και στον τρόπο λειτουργίας τους εμφανίζονται σε οργανισμούς εντελώς διαφορετικούς μεταξύ τους.

Ακόμη ένα στοιχείο κοινό για όλους τους οργανισμούς είναι η ύπαρξη των νουκλεϊκών οξέων. Μια οικογένεια μορίων που παίζει καθοριστικό ρόλο στη διατήρηση της γενετικής πληροφορίας των οργανισμών.

Τα νουκλεϊκα οξέα και οι πρωτεΐνες είναι μόρια που συγκαταλέγονται στην κατηγορία των μακρομορίων λόγω του μεγάλου ειδικού τους βάρους. Είναι σύνθετα μόρια (προκύπτουν από τη σύνθεση επαναλαμβανόμενων μικρότερων μορίων) και γι αυτό χαρακτηρίζονται και ως μακρομόρια (macromolecules).

Μια από τις σημαντικότερες έννοιες στη βιολογία είναι η έννοια της εξέλιξης. Όλοι οι οργανισμοί ανεξάρτητα από το πόσο πολύπλοκοι ή απλοί είναι έχουν τόσο προγόνους όσο και απογόνους. Η παρουσία κοινών

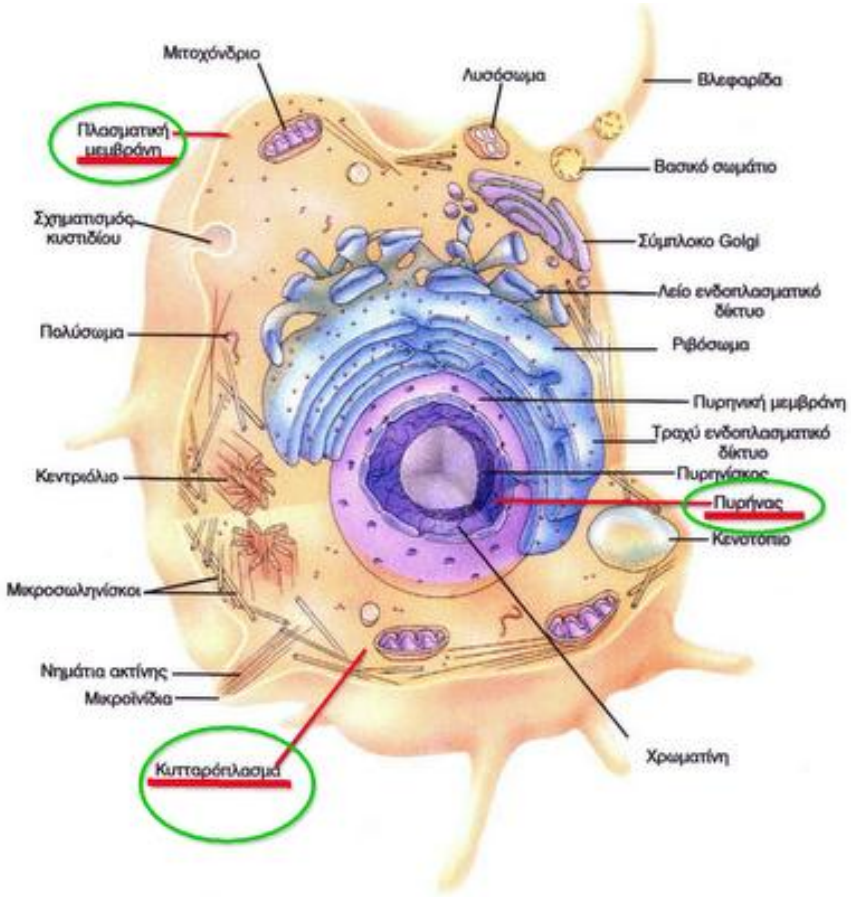
προγόνων αποτελεί τον κύριο λόγο της ομοιότητας που παρατηρείται μεταξύ των μελών κάποιας οικογένειας οργανισμών. Η εξέλιξη αποτελείται από τρείς διαδικασίες (Hunter, 1993; 2004):

Την κληρονομικότητα (inheritance) που είναι η διαδικασία της μεταβίβασης χαρακτηριστικών από τον πρόγονο προς τους απογόνους.

Την μεταβλητότητα (variation) που αποτελείται από τα χαρακτηριστικά που κρατούν τη διαφορετικότητα του προγόνου από τους απογόνους (έτσι τα παιδιά δεν αποτελούν ένα πιστό αντίγραφο των γονέων τους) και τέλος την επιλογή (selection), η διαδικασία κατά την οποία επιλέγονται ποιοι οργανισμοί θα έχουν μεγαλύτερη δυνατότητα αναπαραγωγής σε σχέση με κάποιους άλλους.

Ο θεμελιώδης λίθος στο οικοδόμημα της ζωής είναι το κύτταρο (Κουγιανού, Π. Κ., 1996). Το κύτταρο είναι ένας αυτόνομος ζωντανός οργανισμός αποτελούμενος από ένα σύνολο μορίων που αλληλεπιδρούν μεταξύ τους όπου μπορεί να αναπτυχθεί και να αναπαραχθεί ακόμη και με την απουσία άλλων κυττάρων. Βάση βιοχημικών αντιδράσεων που εκτελούνται στο εσωτερικό του, μπορεί να τρέφεται, να αναπαράγεται και να προσαρμόζεται στις συνθήκες του περιβάλλοντος στο οποίο βρίσκεται. Οι οδηγίες που παρέχονται για όλες τις πολύπλοκες βιοχημικές αντιδράσεις που εκτελούνται βρίσκονται στο γενετικό υλικό στον πυρήνα του κυττάρου.

Ο πυρήνας του κυττάρου αποτελεί το σημαντικότερο αλλά και πολυπλοκότερο όργανο του. Βρίσκεται συνήθως στο κέντρο του κυττάρου και το σχήμα του είναι σφαιρικό. Διαχωρίζεται από το υπόλοιπο κύτταρο με μία διπλή μεμβράνη (κυτταρική μεμβράνη) και περιέχει τις γενετικές πληροφορίες του κυττάρου. Η περιοχή έξω από τον πυρήνα ονομάζεται κυτταρόπλασμα (cytoplasm) στο οποίο υπάρχουν διάφορα οργανίδια του κυττάρου όπως τα λυσοσώματα (lysosomes), η συσκευή Golgi (Golgi apparatus) και τα μιτοχόνδρια (mitochondria). Τα μιτοχόνδρια είναι υπεύθυνα για την παραγωγή ενέργειας στο εσωτερικό του κυττάρου, ενώ τα στοιχεία Golgi αποτελούν τις ενεργειακές αποθήκες. Τα μιτοχόνδρια περιέχουν δικό τους γενετικό υλικό.



Εικόνα 1: Τυπικό Ζωικό Κύτταρο

Με βάση τα ανατομικά τους χαρακτηριστικά τα κύτταρα διαχωρίζονται σε δύο κατηγορίες (Woese et al., 1990).

Τα ευκαριωτικά, στα οποία ανήκουν τα κύτταρα των περισσότερων ζωντανών οργανισμών όπως τα ζώα, τα φυτά και οι μύκητες και τα προκαρυωτικά στα οποία ανήκουν κυρίως οι μικρότεροι οργανισμοί (μονοκύτταροι συνήθως) όπως τα βακτήρια και τα αρχαία. Η κύρια διαφορά των δύο αυτών κατηγοριών είναι ή ύπαρξη του πυρήνα στα ευκαριωτικά κύτταρα και η απουσία του στα προκαρυωτικά.

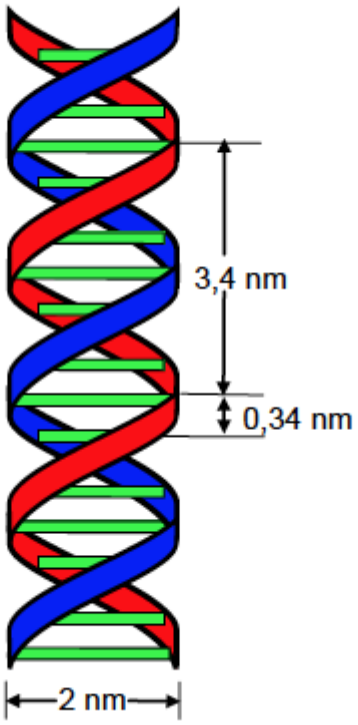
Εκτός από τους πολυκύτταρους οργανισμούς (ευκαρυώτες) και τους μονοκύτταρους οργανισμούς (προκαρυώτες) υπάρχουν και τα ενδοκυτταρικά παράσιτα, τα οποία δεν μοιάζουν με τους ζωντανούς οργανισμούς παρά μόνο όταν βρεθούν μέσα σε αυτούς. Σε αυτή την κατηγορία συγκαταλέγονται οι ιοί (viruses).

Όπως προαναφέραμε, κάθε ζωντανός οργανισμός αποτελείται από μονάδες που ονομάζονται κύτταρα, τα οποία συγκροτούν ομάδες, τους ιστούς, που με τη σειρά τους συγκροτούν μεγαλύτερες ομάδες τα όργανα. Η ζωή και η ανάπτυξη ενός οργανισμού βασίζεται στην ανάπτυξη των κυττάρων του.

Ο κυτταρικός κύκλος (cell cycle) ή αλλιώς ο κύκλος ζωής ενός κυττάρου είναι η διαδικασία που ακολουθεί το κύτταρο κατά την ανάπτυξη του και στη συνέχεια στη διαίρεση του. Αυτή η διαδικασία δεν είναι ίδια για όλα τα κύτταρα. Τα προκαρυωτικά κύτταρα έχουν την ικανότητα να αναπτύσσονται και να διαιρούνται με πολύ γρήγορους ρυθμούς. Αντίθετα, οι πολυκύτταροι οργανισμοί, οι οποίοι αποτελούνται από ευκαριωτικά κύτταρα αρχίζουν τη ζωή τους ως ένα κύτταρο το οποίο προέρχεται από την ένωση ενός αρσενικού και ενός θηλυκού κυττάρου (γαμέτες). Στη συνέχεια αναπτύσσεται, διαιρείται, διαφοροποιείται με σκοπό να αναπτυχθούν οι διάφοροι ιστοί και κατ επέκταση τα όργανα του οργανισμού. Η διαδικασία της αναπαραγωγής και της διαφοροποίησης του κυττάρου ελέγχεται από τις γενετικές πληροφορίες που έχει με σκοπό αποφυγής σοβαρών διαταραχών του οργανισμού όπως τα καρκινώματα που αποτελούν μια ανεξέλεγκτη παραγωγή κυττάρων.

Οι γενετικές πληροφορίες του κυττάρου βρίσκονται κωδικοποιημένες σε ένα μεγάλο και πολύπλοκο μόριο, το DNA (DeoxyriboNucleic Acid).

Το DNA είναι ένα μακρομόριο το οποίο αποτελείται από μικρότερα μόρια, τα νουκλεοτίδια. Βρίσκεται στο εσωτερικό του πυρήνα του κυττάρου μέσα σε ένα ή δύο μικρά σφαιρίδια που ονομάζονται πυρηνίσκοι. Μέσα στον πυρηνίσκο, εκτός από το DNA βρίσκεται και το RNA (RiboNucleic Acid), το οποίο συμβάλει στην πρωτεϊνική σύνθεση (Arthur M. Lesk, 2002).



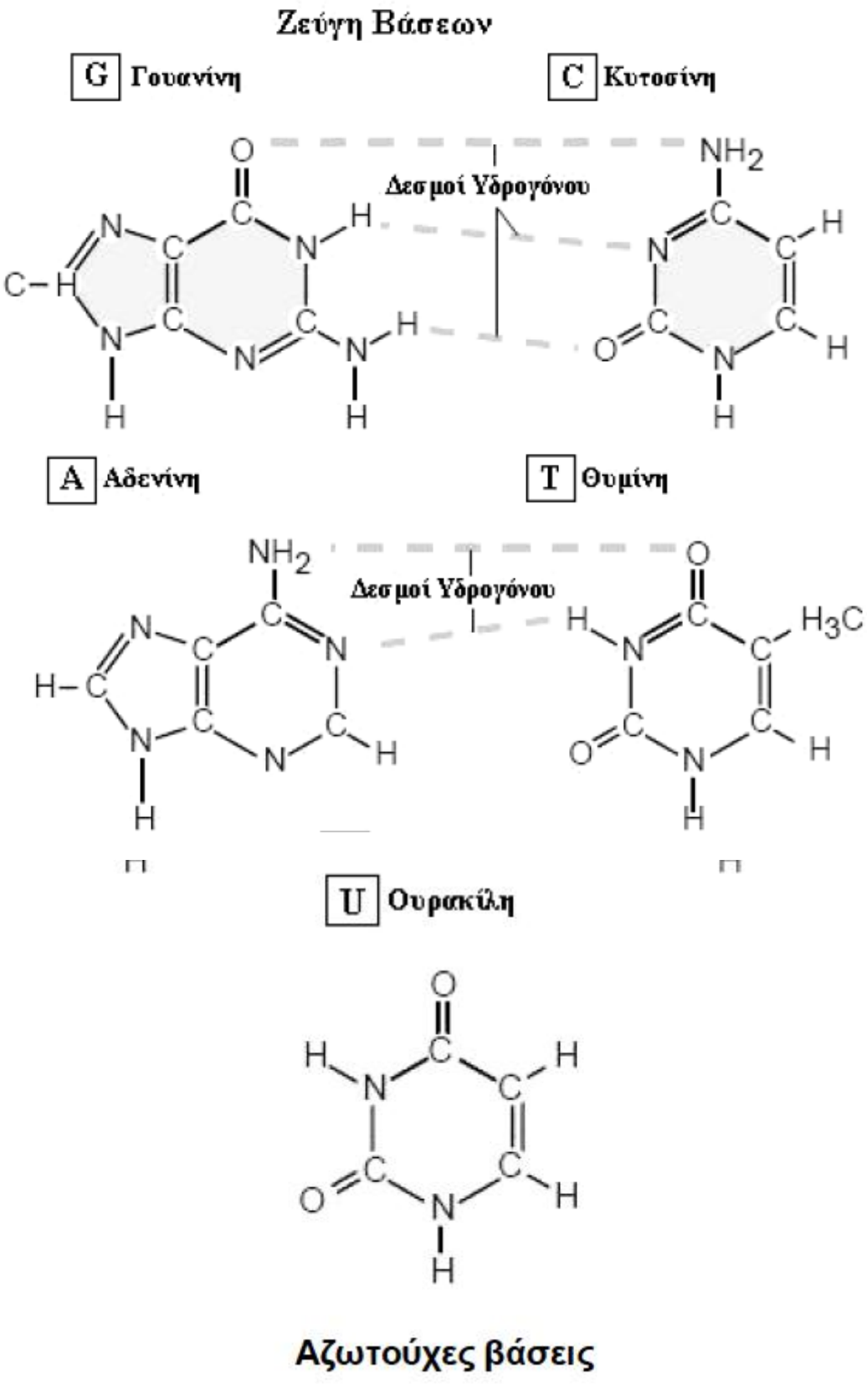
Εικόνα 2: Διπλή Έλικα DNA

Νουκλεοτίδια:

Τα νουκλεοτίδια, η δομική μονάδα των νουκλεικών οξέων (DNA και RNA) είναι σύνθετα μόρια. Αποτελούνται από τρία μόρια συνδεδεμένα μεταξύ τους. Αποτελούνται από μια πεντόζη (ένα σάκχαρο με πέντε άτομα άνθρακα), μια φωσφορική ομάδα και από μια αζωτούχο βάση.

Η πεντόζη των νουκλεοτιδίων του DNA είναι η δεόξυριβόζη (deoxyribose), ενώ η πεντόζη του RNA είναι η ριβόζη (ribose). Όλα τα μόρια του DNA (κατά συνέπεια και του RNA) δεν είναι ίδια. Η διαφορετικότητά τους έγκειται στην αζωτούχο βάση. Οι αζωτούχες βάσεις είναι η αδενίνη A (adenine) και η γουανίνη G (guanine) που ανήκουν στην οικογένεια των πουρινών και η θυμίνη T (thimyne), η κυτοσίνη C (cytosine) και η ουρακίλη U (uracil) που ανήκουν στην οικογένεια των πυριμιδινών.

Η αδενίνη, η γουανίνη και η κυτοσίνη υπάρχουν και στα δύο είδη των νουκλεικών οξέων, ενώ η θυμίνη υπάρχει μόνο στο DNA και η ουρακίλη μόνο στο RNA.



Εικόνα 3: Νουκλεοτίδια

DNA και γονιδίωμα.

Η γενετική πληροφορία κάθε ζωντανού οργανισμού βρίσκεται κωδικοποιημένη σε μόρια DNA τα οποία βρίσκονται σε κάθε κύτταρο του οργανισμού.

Το DNA είναι μια απέραντη βάση δεδομένων χημικών πληροφοριών που περιέχει όλες τις απαραίτητες οδηγίες για την δημιουργία όλων των πρωτεϊνών που ένα κύτταρο μπορεί να χρειαστεί. Μπορεί να είναι μονόκλωνο ή δίκλωνο. Ένα μονόκλωνο μόριο DNA αποτελεί μια αλυσίδα από νουκλεοτίδια. Η διαμόρφωση του δίκλωνου DNA στο χώρο έχει τη μορφή δύο επιμηκών αλυσίδων, οι οποίες συστρέφονται ελικοειδώς μεταξύ τους με δεξιόστροφη φορά (Β. Μαρμαράς, 2000).

Το κάθε νουκλεοτίδιο συνδέεται με το διπλανό του με φωσφοδιεστερικό δεσμό. Οι δυο κλώνοι συνδέονται μεταξύ τους με δεσμούς υδρογόνου που συνάπτονται μεταξύ των αζωτούχων βάσεών τους. Όπως προαναφέραμε, οι αζωτούχες βάσεις που συμμετέχουν στη σύνθεση του DNA είναι τέσσερεις. Η αδενίνη (A), η κυτοσίνη (C) , η γουανίνη (G) και η Θυμίνη (T).

Δεν μπορούν να σχηματίσουν δεσμούς υδρογόνου όλες οι βάσεις μεταξύ τους. Με άλλα λόγια, υπάρχουν συγκεκριμένες συνδέσεις μεταξύ των βάσεων. Η αδενίνη μπορεί να συνδεθεί με τη θυμίνη με δυο δεσμούς υδρογόνου, ενώ η κυτοσίνη συνδέεται με τη γουανίνη με τρείς δεσμούς υδρογόνου. Παρότι αυτοί οι δεσμοί είναι ασθενείς, λόγω του πλήθους συνδέσεων μεταξύ των δύο κλώνων επιτυγχάνεται μια ισχυρή σύνδεση μεταξύ των δύο συμπληρωματικών αλυσίδων. Οι αζωτούχες βάσεις είναι αυτές που καθορίζουν τη «μοναδικότητα» κάθε μορίου του.

Συνεπώς κάθε μόριο του DNA μπορεί να θεωρηθεί ως μια συμβολοσειρά τεσσάρων χαρακτήρων- γραμμάτων: $\sum_{DNA}=\{A,C,T,G\}$ των τεσσάρων νουκλεοτιδίων (αντίστοιχα το μονόκλωνο μόριο του RNA θεωρείτε ως μια συμβολοσειρά από ένα αλφάβητο $\sum_{RNA}=\{A,C,U,G\}$ όπου τη θέση της γουανίνης τη λαμβάνει η ουρακίλη).

Οι αλυσίδες της έλικας έχουν διαφορετική χημική σύσταση στα άκρα τους. Το ένα άκρο είναι το 5’ και το άλλο το 3’.

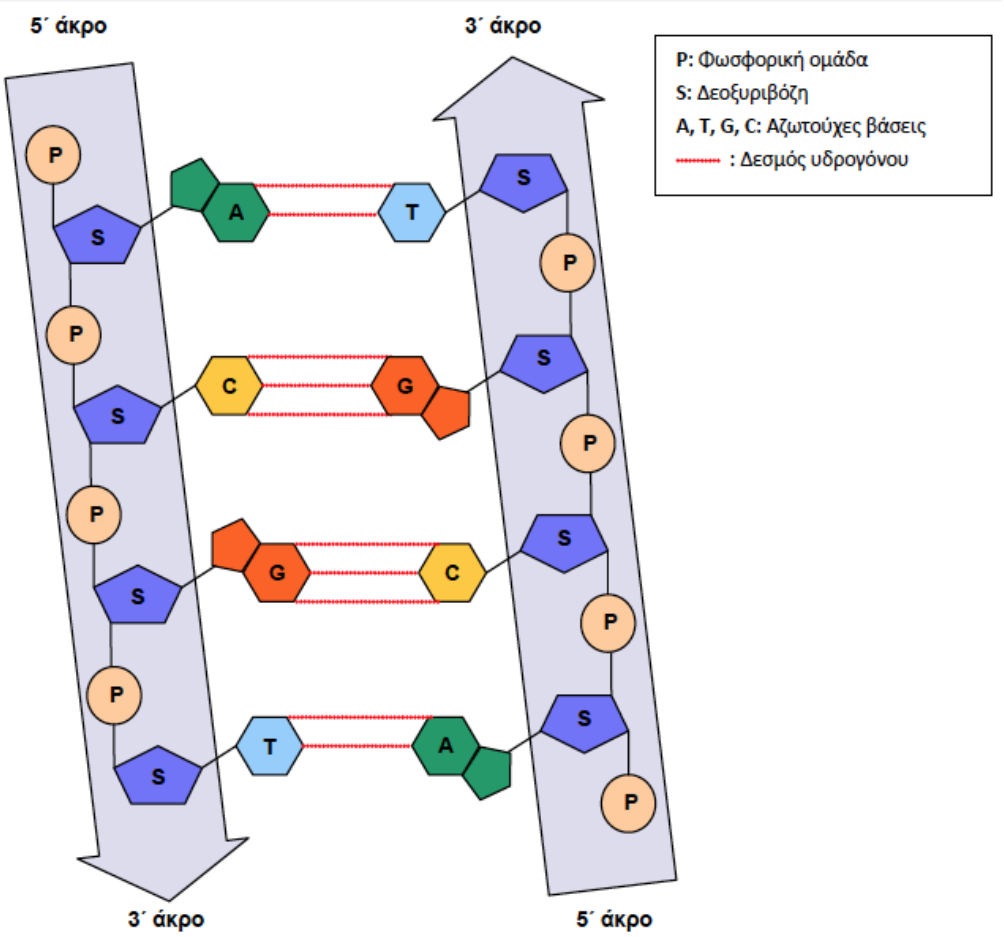
Επομένως η κάθε αλυσίδα είναι κατευθυνόμενη, με προσανατολισμό 5'→ 3'.

Για παράδειγμα αν έχουμε τη συμβολοσειρά AAGTGGA η κατεύθυνσή της είναι 5' A→ A→ G→ T→ G→ G→ A 3', η οποία είναι διαφορετική από τη συμβολοσειρά AGGTGAA με κατεύθυνση 5' A→ G→ G→ T→ G→ A→ A 3'.

Οι δύο αλυσίδες στο δίκλωνο DNA εκτός από συμπληρωματικές είναι και παράλληλες. Για παράδειγμα:

5' A A G T G G A 3'

3' T T C A C C T 5'



Εικόνα 4: Συνδέσεις Νουκλεοτιδίων

Τα ζευγάρια A - T και C - G ονομάζονται ζεύγη βάσεων (base pairs bp) και αποτελούν τη μονάδα μέτρησης του μήκους ενός μορίου DNA. Άλλη μονάδα μέτρησης μήκους αν και δε χρησιμοποιείται τόσο συχνά είναι ο αριθμός των νουκλεοτιδίων που συγκροτούν το μόριο και εκφράζεται σε (nt).

Ο συμπληρωματικός τρόπος διασύνδεσης των νουκλεοτιδίων όπως προαναφέραμε μας επιτρέπει να θεωρήσουμε το μόριο του DNA ως μια μονή έλικα- συμβολοσειρά από την οποία μπορούμε να παράγουμε τη συμπληρωματική της.

Ολόκληρο το DNA ενός ζωντανού οργανισμού λέγεται γονιδίωμα (genome). Το ανθρώπινο γονιδίωμα αποτελείται περίπου από 3 εκατομμύρια βάσεις (ή 10^8 νουκλεοτίδια ανά έλικα) και μπορεί να κατασκευάζει περίπου 100000 πρωτεΐνες μεγέθους μερικών εκατοντάδων αμινοξέων σε σύγκριση με το γονιδίωμα των βακτηρίων που παρασκευάζει γύρω στα 500- 1500 πρωτεΐνες, το κατώτερο όριο παραγωγής πρωτεΐνης για την επιβίωση ενός ζωντανού οργανισμού.

Πιο συγκεκριμένα, υπεύθυνα για την παρασκευή πρωτεϊνών είναι τα γονίδια (genes). Τα γονίδια, είναι οι ενεργές υποομάδες DNA. Το κάθε γονίδιο περιέχει τις οδηγίες εκείνες που αντιστοιχούν στην δημιουργία μιας συγκεκριμένης πρωτεΐνης. Το σύνολο όλων των ζωντανών οργανισμών χωρίζεται σε δύο κατηγορίες. Τους ευκαρυωτικούς οργανισμούς όπου το DNA βρίσκεται στον πυρήνα των κυττάρων και στους προκαρυωτικούς οργανισμούς όπου το DNA βρίσκεται στο κυτταρόπλασμα. Η κύρια διαφορά τους έγκειται στο γεγονός ότι το μεν προκαρυωτικό γονιδίωμα είναι μια συνεχόμενη συμβολοσειρά ενώ το ευκαρυωτικό γονιδίωμα αποτελείται από χρωμοσώματα τα οποία είναι σετ συμβολοσειρών.

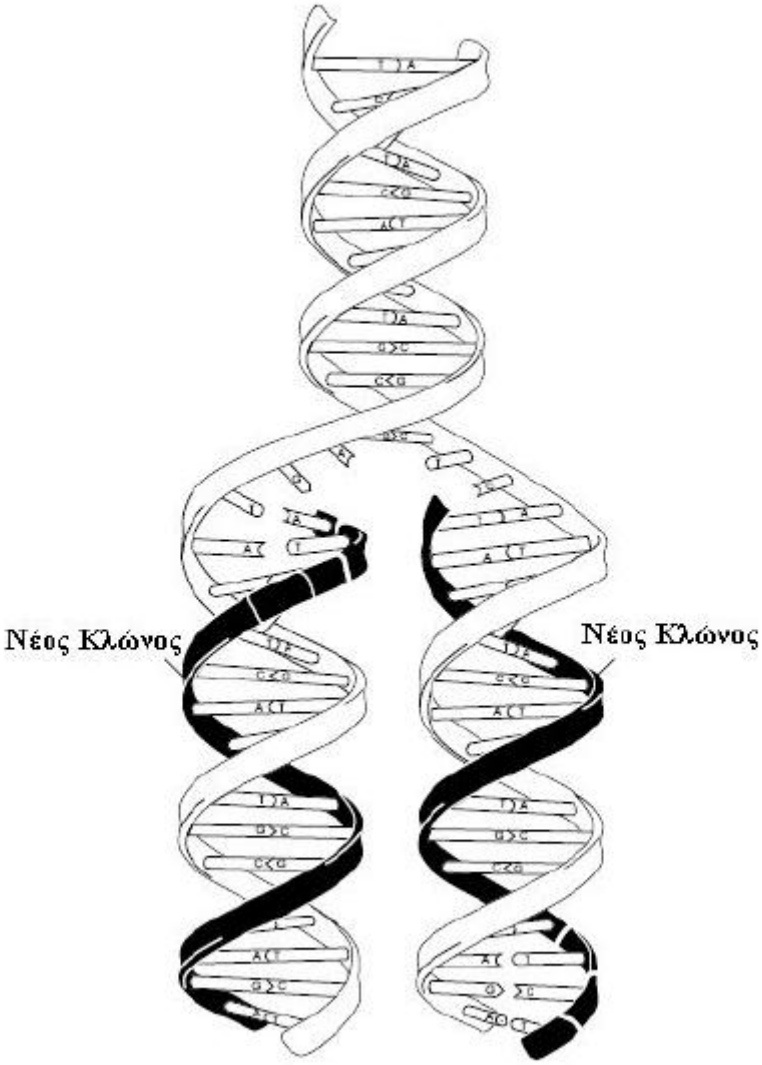
Οι σημαντικότερες διαδικασίες που λαμβάνουν μέρος στο μόριο του DNA είναι τρείς: Η αντιγραφή, η μετάφραση και η μετάλλαξη. Έτσι το χρωμοσωμικό DNA αποτελεί το πρώτο βήμα για τη χημική διεργασία της ζωής.

- Αντιγραφή

  Κατά τη διαδικασία της αντιγραφής παράγονται δυο νέα μόρια δίκλωνου DNA όμοια με το παλιό. Η διαδικασία γίνεται ως εξής.

  Οι δύο αλυσίδες ανοίγουν σπάζοντας η μεταξύ τους δεσμοί υδρογόνου, ελευθερώνοντας τις αζωτούχες βάσεις. Στη συνέχεια κατάλληλα ένζυμα (DNA πολυμεράσες) φέρνουν τις συμπληρωματικές αζωτούχες βάσεις και συνθέτουν μια νέα αλυσίδα. Έτσι το νέο μόριο αποτελείται από έναν κλώνο του παλιού και έναν νέο κλώνο. Ο τρόπος αυτός διπλασιασμού

ονομάζεται ημισυντηρητικός και βασίζεται στην συμπληρωματικότητα των δύο κλώνων του μορίου.



Εικόνα 5: Διαδικασία Αντιγραφής DNA

- Μεταγραφή

  Κατά τη διαδικασία της μεταγραφής το DNA συνθέτει το RNA και συγκεκριμένα το mRNA όπου με τη σειρά του θα κωδικοποιήσει τα αμινοξέα με σκοπό την παραγωγή πρωτεϊνών. Η διαδικασία έχει ως εξής.

  Παρόμοια με την αντιγραφή, οι δύο κλώνοι του DNA ανοίγουν σε συγκεκριμένη όμως περιοχή, απέναντι από τον ένα κλώνο προσκολλούνται αζωτούχες βάσεις (στην θέση της θυμίνης εισέρχεται

η ουρακίλη) ώστε να δημιουργηθεί το RNA. Μόλις δημιουργηθεί, αποκόπτεται από τον συμπληρωματικό κλώνο και επανέρχεται η δίκλωνη αλυσίδα του DNA στην αρχική της μορφή.

Εκτός από το mRNA (message RNA) υπάρχουν άλλα τρία είδη RNA τα οποία δημιουργούνται από το DNA με τον ίδιο τρόπο. Συμμετέχουν στην φάση της μετάφρασης που θα δούμε παρακάτω και είναι το μεταφορικό RNA (tRNA), το ριβοσωμικό RNA (rRNA) και το μικροπυρηνικό (snRNA).

- Μετάφραση.

  Η μετάφραση ίσος αποτελεί και την πιο σύνθετη διαδικασία. Κατά τη διαδικασία της πραγματοποιείται η σύνθεση των πρωτεϊνών. Καταλυτικό ρόλο για αυτή τη φάση παίζει το mRNA το οποίο περνά από ένα επιπρόσθετο στάδιο, το στάδιο της ωρίμανσης. Το στάδιο της ωρίμανσης περιλαμβάνει το μάτισμα του mRNA καθώς και την τροποποίηση των άκρων του.

  Τμήματα του mRNA θα μεταφραστούν σε αμινοξέα τα οποία με τη σειρά τους θα συνθέσουν τις πρωτεΐνες. Κατά τη διαδικασία του ματίσματος γίνεται η διαλογή και η απομάκρυνση ενδιάμεσων τμημάτων που δεν θα κωδικοποιήσουν κανένα αμινοξύ. Αυτά τα τμήματα είναι τα ιντρόνια. Τα εξόνια είναι τα τμήματα που θα παραμείνουν και θα δημιουργήσουν τα κωδικόνια, τριάδες (τριπλέτες) νουκλεωτιδίων που κωδικοποιούν αμινοξέα. Η τροποποίηση των δύο άκρων της αλυσίδας του mRNA επιτυγχάνεται με την προσθήκη μιας καλύπτρας, η οποία για το άκρο 5’ είναι ένα νουκλεοτίδιο με αντίθετο προσανατολισμό (3’ → 5’), ενώ για το άκρο 3’ μια σειρά νουκλεοτιδίων αδενίνης.

  Έτσι τα δύο άκρα σφραγίζονται και προχωρά η διαδικασία της μετάφρασης με τη μεταφορά του mRNA από τον πυρήνα στο κυτταρόπλασμα.

  Οι τριπλέτες νουκλεοτιδίων που θα μεταφραστούν σε αμινοξέα ονομάζονται κωδικόνια (codons). Αν και ο συνδυασμός των νουκλεοτιδίων ανά τρία μας δίνει 3^4= 64 διαφορετικά κωδικόνια, αυτά αντιστοιχούν σε 20 μόνο έγκυρες λέξεις. Όσες είναι τα αμινοξέα. Συνεπώς έχουμε την ύπαρξη ενός εκφυλισμένου κώδικα μιας και

έχουμε την ύπαρξη συνώνυμων κωδικονίων, δηλαδή κωδικόνια που κωδικοποιούν το ίδιο αμινοξύ. Ο κώδικας (γενετικός κώδικας) εμπλουτίζεται με την προσθήκη ακόμη τεσσάρων κωδικονίων, τριών κωδικονίων λήξης (UAA, UAG, UGA) και ενός κωδικονίου έναρξης (AUG).

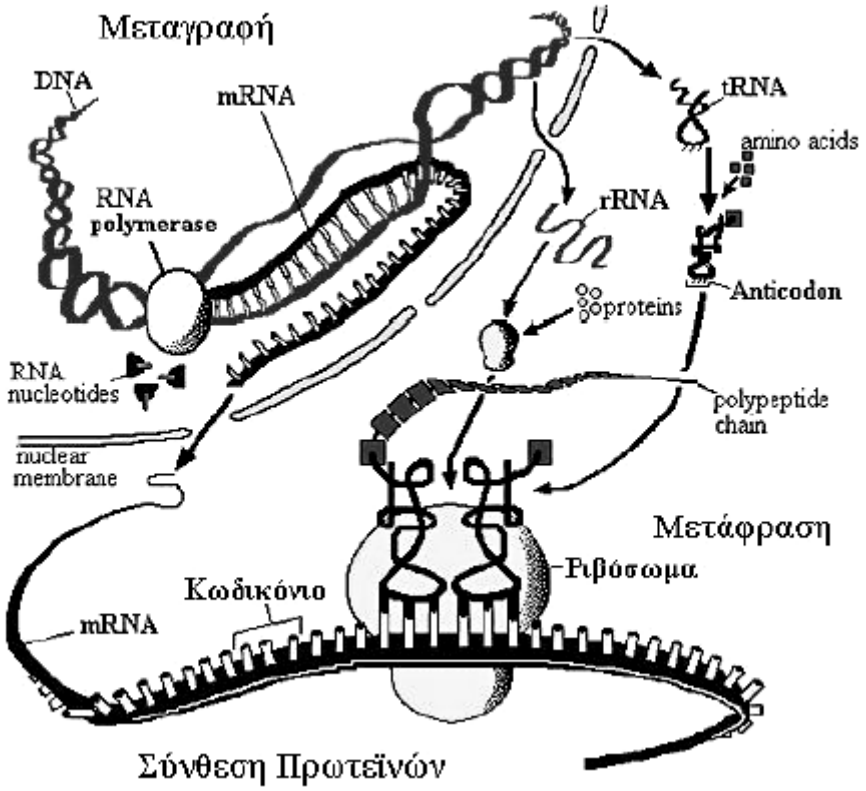
Εδώ θα πρέπει να τονίσουμε και ένα ακόμη χαρακτηριστικό του γενετικού κώδικα το οποίο είναι πολύ βασικό. Η καθολικότητά του. Σχεδόν όλοι οι ζωντανοί οργανισμοί έχουν τον ίδιο γενετικό κώδικα. Έτσι για κάθε οργανισμό έχουμε τα ίδια αποτελέσματα κατά τη διαδικασία της πρωτεϊνοσύνθεσης.

Πίνακας 1: Γενετικός Κώδικας

First Position	Second Position of Codon: T	Second Position of Codon: C	Second Position of Codon: A	Second Position of Codon: G	Third Position
T	TTT Phe [F]	TCT Ser [S]	TAT Tyr [Y]	TGT Cys [C]	T
	TTC Phe [F]	TCC Ser [S]	TAC Tyr [Y]	TGC Cys [C]	C
	TTA Leu [L]	TCA Ser [S]	TAA *Ter* [end]	TGA *Ter* [end]	A
	TTG Leu [L]	TCG Ser [S]	TAG *Ter* [end]	TGG Trp [W]	G
C	CTT Leu [L]	CCT Pro [P]	CAT His [H]	CGT Arg [R]	T
	CTC Leu [L]	CCC Pro [P]	CAC His [H]	CGC Arg [R]	C
	CTA Leu [L]	CCA Pro [P]	CAA Gln [Q]	CGA Arg [R]	A
	CTG Leu [L]	CCG Pro [P]	CAG Gln [Q]	CGG Arg [R]	G
A	ATT Ile [I]	ACT Thr [T]	AAT Asn [N]	AGT Ser [S]	T
	ATC Ile [I]	ACC Thr [T]	AAC Asn [N]	AGC Ser [S]	C
	ATA Ile [I]	ACA Thr [T]	AAA Lys [K]	AGA Arg [R]	A
	ATG Met [M]	ACG Thr [T]	AAG Lys [K]	AGG Arg [R]	G
G	GTT Val [V]	GCT Ala [A]	GAT Asp [D]	GGT Gly [G]	T
	GTC Val [V]	GCC Ala [A]	GAC Asp [D]	GGC Gly [G]	C
	GTA Val [V]	GCA Ala [A]	GAA Glu [E]	GGA Gly [G]	A
	GTG Val [V]	GCG Ala [A]	GAG Glu [E]	GGG Gly [G]	G

Οι πρωτεΐνες συντίθενται στα ριβοσώματα, όπου υπάρχει είδη το rRNA. Με την άφιξη του mRNA και του tRNA το οποίο μεταφέρει το

ανάλογο αμινοξύ, γίνεται η κωδικοποίηση του αμινοξέος με το κωδικόνιο του mRNA. Το ριβόσωμα αλλάζει συνεχώς θέσεις στο mRNA και επομένως και κωδικόνια που αναγνωρίζονται από τα αντίστοιχα tRNA. Με αυτό τον τρόπο, αφού τα αμινοξέα μπουν σε μια σειρά, ενώνονται μεταξύ τους και σχηματίζουν τις πρωτεΐνες. Ο διαχωρισμών των τμημάτων που συνθέτουν την κάθε πρωτεΐνη επιτυγχάνεται με την ύπαρξη του κωδικονίου έναρξης και των κωδικονίων λήξης.



Εικόνα 6: Διαδικασία Μετάφρασης

- Μετάλλαξη

  Μετάλλαξη είναι η διαδικασία που περιγράφει οποιαδήποτε αλλαγή μπορεί να συμβεί στις βάσεις του DNA. Τα είδη των αλλαγών που μπορούν να συμβούν είναι η αντικατάσταση μιας βάσης από μια άλλη, η προσθήκη μιας βάσης ή η αφαίρεση μιας άλλης. Κατά την αντικατάσταση μιας βάσης, η αλλαγή η οποία υφίσταται είναι η αλλαγή

ενός κωδικονίου, κατά συνέπεια η αλλαγή ενός αμινοξέος στη σύνθεση μιας πρωτεΐνης. Μια τέτοια αλλαγή, οδηγεί τον οργανισμό σε παραγωγή τροποποιημένων πρωτεϊνών, κάτι που μπορεί να προκαλέσει ολέθριες ζημιές στην εύρυθμη λειτουργία του. Για παράδειγμα, η αντικατάσταση του αμινοξέος γλουταμίνη (Gln) από το αμινοξύ βαλίνη (Val) η οποία προκαλείται με μια αντικατάσταση μιας βάσης σε μια τριπλέτα του DNA, προκαλεί δρεπανοκυτταρική αναιμία στον οργανισμό.

Κατά την προσθήκη ή την αφαίρεση μιας βάσης, οι αλλαγές που παρατηρούνται είναι πολύ πιο σαρωτικές, μιας και αλλάζουν όλες οι τριπλέτες από εκείνο το σημείο και μετά με συνέπεια την αλλαγή όλων των αμινοξέων.

Οι μεταλλάξεις, οφείλονται κυρίως σε λάθη που παρατηρούνται κατά την αντιγραφή του DNA καθώς και στην έκθεση του οργανισμού σε βλαπτικούς γι αυτόν συνθήκες, όπως η ακτινοβολία ή χημικοί παράγοντες (Θ. Α. Παταργιάς et al., 1996).

Αμινοξέα και Πρωτεΐνες

Ως πρώτη ύλη για την παραγωγή των πρωτεϊνών χρησιμοποιείται το αμινοξύ. Ένα αμινοξύ αποτελείται από ένα μόριο άνθρακα, ένα μόριο υδρογόνου, μια καρβοξυλική ομάδα, μια αμινομάδα και μια ομάδα R. Όπως και στα νουκλεοτίδια η αζωτούχα ομάδα είναι αυτή που τα διαφοροποιεί μεταξύ τους, ανάλογα κι εδώ η ομάδα R είναι αυτή στην οποία οφείλουν τη διαφορετικότητά τους τα αμινοξέα. Τα αμινοξέα που συμμετέχουν στη σύσταση των πρωτεϊνών είναι είκοσι. Παρόλα αυτά, σε κάθε κύτταρο υπάρχουν περίπου εβδομήντα.

Ο παρακάτω πίνακας μας δείχνει τα 20 αυτά αμινοξέα:

Πίνακας 2: Αμινοξέα

Ονομασία	Συμβολισμοί		Ονομασία	Συμβολισμοί	
Αλανίνη (Alanine)	ALA	A	Ιστιδίνη (Histidine)	HIS	H
Αργινίνη (Arginine)	ARG	R	Κυστεΐνη (Cysteine)	CYS	C
Ασπαραγίνη (Asparagine)	ASN	N	Λευκίνη (Leucine)	LEU	L
Ασπαρτικό οξύ (Aspartic acid)	ASP	D	Λυσίνη (Lysine)	LYS	K
Βαλίνη (Valine)	VAL	V	Μεθειονίνη (Methionine)	MET	M
Γλουταμικό οξύ (Glutamic acid)	GLU	E	Προλίνη (Proline)	PRO	P
Γλουταμίνη (Glutamine)	GLN	Q	Σερίνη (Serine)	SER	S
Γλυκίνη (Glycine)	GLY	G	Τρυπτοφάνη (Tryptophan)	TRP	W
Θρεονίνη (Threonine)	THR	T	Τυροσίνη (Tyrosine)	TYR	Y
Ισολευκίνη (Isoleucine)	ILE	I	Φαινυλαλανίνη (Phenylalanine)	PHE	F

Η δημιουργία των πεπτιδικών αλυσίδων από την ένωση των αμινοξέων δημιουργεί πολυδιάστατα μόρια που ονομάζονται πρωτεΐνες.

Οι πρωτεΐνες είναι πολύπλοκα μακρομόρια που αποτελούν περισσότερο από το 50% του ξηρού βάρους των κυττάρων. Αποτελούνται από μια ή περισσότερες πολυπεπτιδικές αλυσίδες και παίζουν καθοριστικό ρόλο στη δομή και τη λειτουργία των κυττάρων. Με βάση τη λειτουργία τους χωρίζονται στις παρακάτω κατηγορίες:

1) Δομικές, όπως το κολλαγόνο και η κερατίνη και συμβάλουν στη δομή του οργανισμού.
2) Ένζυμα, οι βιολογικοί καταλύτες του οργανισμού που λαμβάνουν δράση στο μεταβολισμό.
3) Πρωτεΐνες μεμβράνης, που αναλαμβάνουν τη ρύθμιση και τη συντήρηση του κυτταρικού περιβάλλοντος.

Η δομή των πρωτεϊνών είναι πολυπλοκότερη από τη δομή του DNA και του RNA, λόγω των διαφόρων χημικών αλληλοεπιδράσεων που δημιουργούνται ανάμεσα στα επιμέρους δομικά τους στοιχεία. Όπως προείπαμε, οι πρωτεΐνες αποτελούνται από μια ή και περισσότερες πολυπεπτιδικές αλυσίδες, δηλαδή αλληλουχίες αμινοξέων που μπορούν να αναπαρασταθούν ως συμβολοσειρές του αλφαβήτου:

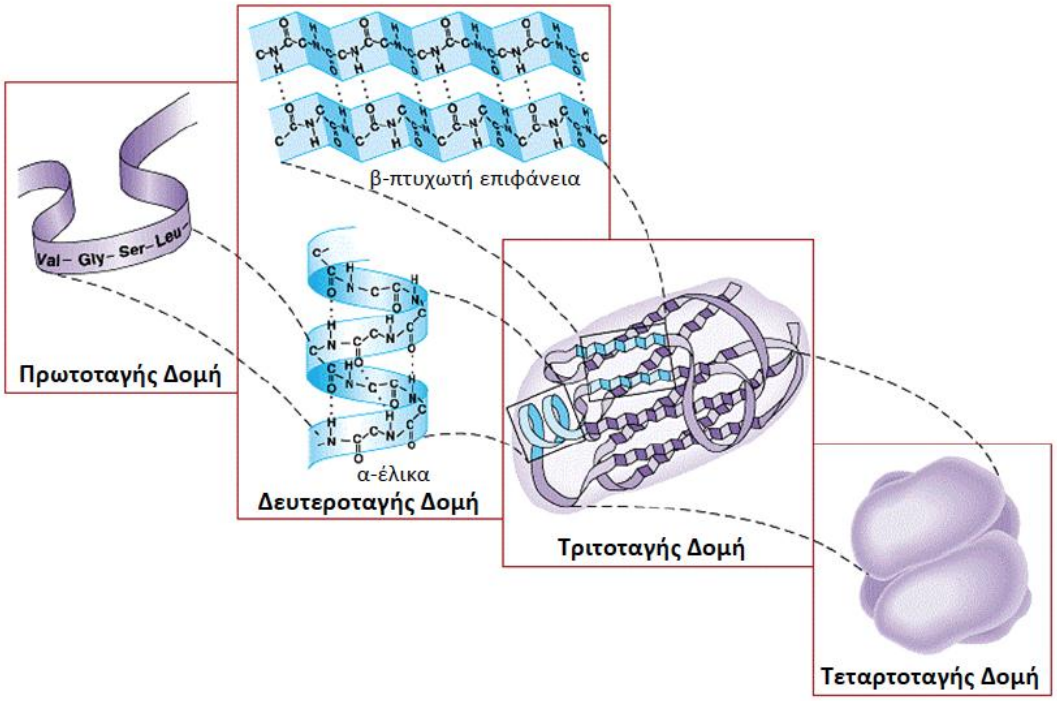
$$\sum_{Protein} = \{Ala, Arg, Asn, Asp, Cys, Gln, Glu, Gly, His, Ile, Len, Lys, Met, Phe, Pro, Thr, Trp, Tyr, Val\}$$

Αυτή είναι η πρωτοταγής δομή (primary structure) των πρωτεϊνών. Η πολυπεπτιδικές αλυσίδες έχουν την ιδιότητα να αναδιπλώνονται στο χώρο με αποτέλεσμα τη δημιουργία δύο δομών μέσα στις αναδιπλωμένες αλυσίδες, την α- έλικα και τη β- πτυχωτή. Οι δύο αυτές δομές ενώνονται μεταξύ τους με βρόγχους και αποτελούν τη δευτεροταγή δομή (secondary structure).

Η τριτοταγής δομή (tertiary structure) της πρωτεΐνης αποτελεί ουσιαστικά την τρισδιάστατη μορφή του μορίου στο χώρο, ως απόρροια της εμφάνισης εσωτερικών ελκτικών ή απωστικών δυνάμεων μεταξύ των μερών του μορίου λόγω του φαινομένου της αναδίπλωσης.

Τέλος, η τεταρτοταγής (quaternary structure) είναι η τρισδιάστατη μορφή ενός μορίου πρωτεΐνης που καταλαμβάνει το χώρο, αλλά πιο σύνθετο, αποτελείται από περισσότερες της μιας πεπτιδικές αλυσίδες.

Τα δύο πρώτα επίπεδα διαμόρφωσης της πρωτεΐνης, η παρουσίασή της στο χώρο ως απλή αλληλουχία αμινοξέων δεν υφίστανται παρά μόνο ως εργαστηριακό αποτέλεσμα με γνώμονα την ευκολότερη διαχείριση των πρωτεϊνών κατά τη μελέτη τους. Η πρόβλεψη της δομής της πρωτεΐνης (protein prediction) από την αλληλουχία των αμινοξέων είναι ένα από τα σημαντικότερα προβλήματα που αντιμετωπίζει η βιοπληροφορική (Β. Προμπόνας, 2006).



Εικόνα 7: Δομή Πρωτεΐνης

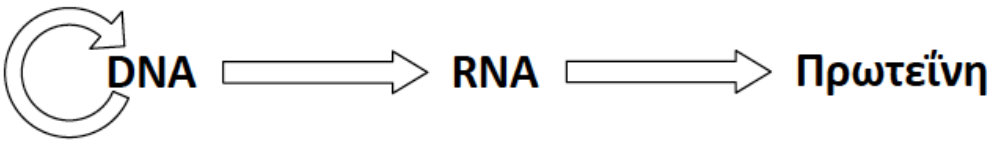
1.5. Το Δόγμα της Μοριακής Βιολογίας

Το 1958, ο Frederic Crick διατύπωσε το κεντρικό δόγμα της μοριακής βιολογίας (central dogma), το οποίο αφορά τη ροή της γενετικής πληροφορίας.

Το DNA μεταγράφει την πληροφορία του στο RNA (διαδικασία της μεταγραφής), το RNA μεταφράζεται σε πρωτεΐνες (διαδικασία της μετάφρασης).

DNA → RNA→ Πρωτεΐνες.

Το DNA επίσης έχει και την ικανότητα του αναδιπλασιασμού (διαδικασία της αντιγραφής).



Εικόνα 8: Βιολογικό Δόγμα

Αν και θεωρείται ως κεντρικό δόγμα, σήμερα είναι γνωστό ότι δεν ισχύει καθολικά για όλους τους οργανισμούς. Για παράδειγμα ο HIV ο ιός που προκαλεί το AIDS ανήκει στην κατηγορία των ρετροϊών έχει την ικανότητα μέσω της διαδικασίας της αντίστροφης μεταγραφής να μετατρέψει το RNA του σε DNA.

Επίσης σε κάποιους άλλους κατώτερους οργανισμούς (ιούς κυρίως) παρατηρείται το φαινόμενο του αυτοδιπλασιασμού του RNA τους (M. Lesk, 2002).

1.6. Βάσεις βιολογικών δεδομένων.

Οι βάσεις βιολογικών δεδομένων περιέχουν δεδομένα από ένα ευρύ φάσμα της γενετικής. Ανάλογα με το είδος των βιολογικών δεδομένων που περιέχουν χωρίζονται στις παρακάτω κατηγορίες.

Γενικευμένες ή αρχειακές βιολογικές βάσεις δεδομένων

Χωρίζονται σε δύο υποκατηγορίες:

- Τις πρωτογενείς βάσεις δεδομένων αλληλουχιών (primary sequence databases) οι οποίες περιέχουν αλληλουχίες DNA και πρωτεϊνών όπου το γονιδίωμα των οργανισμών έχει αποκρυπτογραφηθεί πλήρως ή μερικώς.
- Τις βάσεις δεδομένων των τρισδιάστατων δομών των νουκλεϊκών οξέων και των πρωτεϊνών.

Οι τρείς μεγαλύτερες βάσεις δεδομένων DNA είναι η GenBank με έδρα τις ΗΠΑ, η EMBL με έδρα το Ηνωμένο Βασίλειο και η DDBJ που βρίσκεται στην Ιαπωνία. Πλέων και οι τρείς αυτές βάσεις αλληλεπιδρούν μεταξύ τους με αποτέλεσμα σε περίπτωση εισόδου νέων δεδομένων υπάρχει άμεση ενημέρωσή τους. Η Uniprot, η οποία ξεκίνησε ως Swiss- Prot από το τμήμα ιατρικής βιοχημείας του πανεπιστημίου της Γενεύης, αποτελεί τη μεγαλύτερη βάση δεδομένων πρωτεϊνικών ακολουθιών.

Δευτερογενείς βιολογικές βάσεις δεδομένων

Οι δευτερογενείς βάσεις βιολογικών δεδομένων, είναι οι βάσεις όπου οι εγγραφές τους προέρχονται από την επεξεργασία των εγγραφών των γενικευμένων βιολογικών βάσεων. Έτσι προκύπτουν βάσεις δεδομένων DNA και πρωτεϊνικών ακολουθιών όπου δεν υπάρχουν διπλότυπες καταχωρήσεις, βάσεις δεδομένων όπου έχουν καταγραφεί οι μεταλλάξεις ή οι πολυμορφισμοί σε αλληλουχίες DNA και σε αλληλουχίες πρωτεϊνών, βάσεις δεδομένων που περιέχουν ομαδοποιημένα γονιδιώματα.

Εξειδικευμένες βάσεις δεδομένων

Σε αυτή την κατηγορία ανήκουν οι βάσεις δεδομένων που περιέχουν πληροφορίες για την γονιδιακή έκφραση ή για τις βιοχημικές αντιδράσεις που εκτελούνται μέσα στο κύτταρο, όπως είναι οι βάσεις δεδομένων μικροσυστοιχιών ή οι βάσεις δεδομένων μεταβολικών μονοπατιών, αντίστοιχα.

Διαδικτυακές βιβλιοθήκες

Τέλος, θα πρέπει να αναφέρουμε και την ανάγκη ύπαρξης διαδικτυακών βιβλιοθηκών λόγω της τεράστιας αύξησης του όγκου της διαθέσιμης δημοσιευμένης πληροφορίας σε άρθρα, περιοδικά, βιβλία και πρακτικά συνεδρίων.

Μια τέτοια και ίσως η μεγαλύτερη διαδικτυακή βιβλιοθήκη είναι η PubMed όπου υπάρχουν καταγεγραμμένα περισσότερα από 20 εκατομμύρια άρθρα και μάλιστα κάθε χρόνο προστίθενται περίπου 500.000 νέες εγγραφές. Εξίσου σημαντική είναι και η NCBI BookShelf, η οποία περιέχει πληθώρα ηλεκτρονικών βιβλίων.

1.7. Προβλήματα που απασχολούν τη βιοπληροφορική

Τα βιολογικά δεδομένα που προκύπτουν από πειράματα της μοριακής βιολογίας και αποθηκεύονται στις βάσεις δεδομένων για τις οποίες μιλήσαμε, είναι κυρίως σε μορφή αλληλουχιών.

Η βιοπληροφορική καλείται να δημιουργήσει κατάλληλα εργαλεία με τα οποία θα μπορέσουμε να οργανώσουμε, να αναλύσουμε και να εξάγουμε σημαντικά συμπεράσματα από αυτά τα δεδομένα.

Επομένως η πρόκληση για τους ερευνητές είναι η εύρεση αποδοτικών αλγόριθμων για τη διαχείριση βιολογικών αλληλουχιών.

1.7.1. Αλληλούχιση γονιδιώματος (Sequencing DNA)

Όπως προαναφέραμε, η διαφορετικότητα μεταξύ των μορίων του DNA έγκειται στην ύπαρξη της αζωτούχας βάσης. Έτσι, υπάρχουν τεσσάρων ειδών μόρια DNA, όσες είναι και οι διαφορετικές αζωτούχες βάσεις.

Αν θέλουμε να παρουσιάσουμε το DNA ενός οργανισμού σαν μια ακολουθία, αρκεί να βάλουμε στη σειρά τα ονόματα των βάσεων από τις οποίες αποτελείται. Το μήκος και η πολυπλοκότητας της ακολουθίας διαφέρει από οργανισμό σε οργανισμό. Ένας οργανισμός που αποτελείται από πολλά διαφορετικά όργανα τα οποία επιτελούν πολλές και πολύπλοκες διεργασίες συσσωρεύει περισσότερη πληροφορία στο DNA του, από έναν δομικά πιο απλό οργανισμό. Κάθε μια βάση ξεχωριστά, αλλά και συνδυασμοί αυτών καθοδηγούν κάθε κύτταρο του οργανισμού. Ολόκληρη η αλληλουχία δίνει ρόλους στα κύτταρα. Καθορίζει ποια κύτταρα θα είναι υπεύθυνα για το κάθε όργανο του σώματος και πως θα λειτουργούν.

Ένα σκέλος από το ανθρώπινο DNA περιέχει αρκετά δισεκατομμύρια νουκλεοτίδια, ενώ ενός βακτηρίου περιέχει μερικά εκατομμύρια. Το DNA βρίσκεται στον πυρήνα των κυττάρων σε μορφή «κουβαριού». Αν ξεδιπλώσουμε το DNA ενός ανθρώπινου κυττάρου θα δούμε ότι ξεπερνάει τα δύο μέτρα. Η αλληλούχιση του ανθρώπινου γονιδιώματος ολοκληρώθηκε το 2003.

Η διαδικασία της αλληλούχισης.

Μια απλοϊκή προσέγγιση του προβλήματος, επιτυγχάνεται με το να φανταστούμε ότι έχουμε αρκετά αντίτυπα ενός βιβλίου και τα κόβουμε σε 10 εκατομμύρια μικρά κομμάτια ασύμμετρα μεταξύ τους. Το κάθε κομμάτι μπορεί να συνδεθεί με ένα άλλο κομμάτι άλλου αντιτύπου υπερκαλύπτοντάς το. Ας υποθέσουμε ότι ένα εκατομμύριο από αυτά τα κομμάτια έχει χαθεί, και ένα μεγάλο ποσοστό από τα υπόλοιπα κομμάτια είναι λερωμένο με μελάνι.

Η προσπάθειά μας είναι από αυτά τα κομμάτια, να ανακτήσουμε το αρχικό κείμενο.

Οι βιολόγοι, με την υπάρχουσα τεχνολογία μπορούν να αποσπάσουν από κάθε πείραμα και να διαβάσουν κομμάτια του γονιδιώματος μεγέθους 300 - 500 βάσεων. Το πρόβλημα επομένως έγκειται στη συναρμολόγηση αυτών των κομματιών ώστε να δημιουργηθεί η ακολουθία ολόκληρου του γονιδιώματος, ξεπερνώντας επιμέρους προβλήματα όπως αναπόφευκτα πειραματικά σφάλματα.

Η παραπάνω γενίκευση του προβλήματος δεν καλύπτει πλήρως το πρόβλημα της επανασυναρμολόγησης του γονιδιώματος από τη στιγμή που απαιτούνται τέλεια δεδομένα που δεν διαθέτουμε. Η αλυσίδα του DNA περιέχει κομμάτια που επαναλαμβάνονται.

Το ανθρώπινο γονιδίωμα για παράδειγμα περιέχει πολλές επαναλήψεις, όπως μια αλληλουχία 300 bp Alu η οποία επαναλαμβάνεται περίπου ένα εκατομμύριο φορές. Ευτυχώς, διαφορετικά αντίγραφα αυτών των επαναλήψεων μεταλλάσσονται διαφορετικά κατά την διάρκεια της εξέλιξης, με αποτέλεσμα να μη θεωρούνται ακριβώς ως επαναλήψεις. Αυτό δίνει τη δυνατότητα συναρμολόγησης της αλληλουχίας παρόλο την ύπαρξη αυτών των επαναλήψεων. Άλλη μία σημαντική επιπλοκή που παρουσιάζεται, είναι ότι η επιλογή της βασικής αλυσίδας μετά το διαχωρισμό της διπλής έλικας γίνεται με αυθαίρετο τρόπο.

Η προσπάθεια αλληλούχισης του DNA πέρα από το δύσκολο υπολογιστικό πρόβλημα της συναρμολόγησης των κομματιών αναδεικνύει και άλλα εξίσου πολύπλοκα υπολογιστικά προβλήματα όπως την ανεύρεση

μοτίβων (patterns) μέσα σε ένα γονιδίωμα. Τα μοτίβα, αποτελούν περιοδικές επαναλήψεις υποακολουθιών μέσα στο γονιδίωμα που ποικίλουν στη δομή και στο μήκος τους. Με την εύρεσή τέτοιων επαναλαμβανόμενων μοτίβων οι βιολόγοι μπορούν να ορίσουν δείκτες (markers) για την λεπτομερέστερη χαρτογράφηση του γονιδιώματος.

Οι πρώτοι αλγόριθμοι αλληλούχισης ήταν άπληστοι αλγόριθμοι βασισμένοι στην συνένωση των υποσυμβολοσειρών με βάση κάποια κριτήρια έως ότου απομείνει μια συμβολοσειρά.

1.7.2 Χαρτογράφηση Γονιδιώματος (γονιδιακή αναζήτηση) (Genome Mapping)

Η αποκωδικοποίηση των γενετικών πληροφοριών που κρύβονται μέσα στο γονιδίωμα ενός ζωντανού οργανισμού αποτελεί το “Ιερό δισκοπότηρο” στην επιστήμη της μοριακής βιολογίας. Η χαρτογράφηση του γονιδιώματος είναι η διαδικασία του προσδιορισμού της θέσης των γονιδίων μέσα στα χρωμοσώματα.

Αποτελείται από ένα σύνολο σταθμών με βάση τους οποίους ο ερευνητής μπορεί να προσδιορίσει επ’ ακριβώς τη θέση του κάθε γονιδίου, συνδέσεις ομάδων γονιδίων που βρίσκονται στο ίδιο ή ακόμη και σε διαφορετικά χρωμοσώματα και τη μεταξύ τους απόσταση σε αυτά που βρίσκονται στο ίδιο χρωμόσωμα.

Τα ορόσημα- δείκτες (markers) σε ένα γονιδιακό χάρτη μπορεί να είναι ρυθμιστικές περιοχές γονιδίων που ενεργοποιούν ή καθιστούν αδρανή τα γονίδια, σύντομες γονιδιακές ακολουθίες ή ακόμη και τα ίδια τα γονίδια.

Ένας γονιδιακός χάρτης λειτουργεί με τον ίδιο τρόπο που λειτουργεί και ένας οδικός χάρτης. Παρέχει οδηγίες.

Διευκολύνει τον ερευνητή στην εύρεση συγκεκριμένων γονιδίων, κυρίως παθογόνων με σκοπό την περεταίρω μελέτη τους.

Γνωρίζοντας τη θέση συγκεκριμένων δεικτών, η αναζήτηση ενός παθογόνου γονιδίου μέσα σε 3 δισεκατομμύρια βάσεις μπορεί να περιοριστεί σε αναζήτηση μέσα σε μερικών εκατομμυρίων βάσεων.

Η κυστική ίνωση (cystic fibrosis) μια θανατηφόρος ασθένεια που σχετίζεται με αλλεπάλληλες λοιμώξεις του αναπνευστικού, αποτελεί το πλέον χαρακτηριστικό παράδειγμα αναγκαίας ύπαρξης του γονιδιακού χάρτη.

Η νόσος παρατηρείται σε νεαρές ηλικίες με συχνότητα 1 προς 2500. Ένας στους 25 ανθρώπους που ανήκουν στην καυκάσια φυλή φέρει το ελαττωματικό γονίδιο στο οποίο οφείλεται η κυστική ίνωση. Το παιδί που κληρονομεί αυτό το γονίδιο και από τους δύο γονείς θα αρρωστήσει.

Μέχρι τα μέσα της δεκαετίας του ’80, δεν υπήρχε τρόπος εύρεσης του ελαττωματικού γονιδίου που ευθυνόταν. Το 1985 οι βιολόγοι κατόρθωσαν να εντοπίσουν το γονίδιο στο $7^{ο}$ χρωμόσωμα. Έτσι, η έρευνα περιορίστηκε στην περιοχή του $7^{ου}$ χρωμοσώματος και το 1989 εκτός από την ακριβής θέση του έγινε γνωστή και η σύνθεσή του (Neil C. Jones et al., 2004)

Κατέχοντας επομένως τις απαιτούμενες γνώσεις για τη θέση και τη σύσταση ενός παθογόνου γονιδίου, ανοίγεται ο δρόμος προς τη διάγνωση και εντέλη την καταπολέμηση μιας πάθησης.

Υπάρχουν δύο τύποι γονιδιακών χαρτών. Ο γενετικός χάρτης όπου καθορίζεται η σχετική θέση μεταξύ δύο γονιδίων μέσα στο χρωμόσωμα, και ο φυσικός χάρτης όπου καθορίζεται η απόλυτη θέση του γονιδίου μέσα στο χρωμόσωμα.

Η διαδικασία της χαρτογράφηση του DNA ξεκινά με τον κατακερματισμό της αλυσίδας του σε μικρότερα κομμάτια (από μερικές εκατοντάδες kb έως μερικά Mb). Για τη μελέτη αυτών των κομματιών ακολουθεί η διαδικασία της κλωνοποίησής τους όπου ο κάθε κλώνος αποκτά ένα ‘δακτυλικό αποτύπωμα’. Στη συνέχεια επανασυναρμολογείται το DNA με βάση την επικάλυψη των κλώνων. Κατά τη διαδικασία της επανασυναρμολόγησης προκύπτουν πολλά συνδυαστικά και πιθανοτικά προβλήματα και η ανάγκη δημιουργίας αποδοτικών αλγορίθμων είναι επιβεβλημένη.

Η χαρτογράφηση ουσιαστικά περιλαμβάνει τη διαδικασία της διαίρεσης των χρωμοσωμάτων σε μικρότερα κομμάτια για τα οποία γνωρίζοντας την ακριβή τους θέση μέσα στο χρωμόσωμα μπορούμε να μελετήσουμε τη σύσταση και τη λειτουργία τους.

1.7.3. Πρόβλεψη γονιδιώματος (gene prediction)

Η πρόβλεψη γονιδίων αποτελεί προϋπόθεση για την αναλυτική περιγραφή της λειτουργίας των γονιδίων και των γονιδιωμάτων. Ορίζεται ως το πρόβλημα εντοπισμού των γονιδίων σε μια γονιδιωματική αλληλουχία.

Όπως προαναφέραμε, κάθε κωδικόνιο (τριπλέτα νουκλεοτιδίων) κωδικοποιεί ένα αμινοξύ στην αντίστοιχη πρωτεΐνη. Έχει αποδειχθεί ότι ένα γονίδιο και το πρωτεϊνικό προϊόν του είναι συνευθειακά, δηλαδή το πρώτο κωδικόνιο του γονιδίου κωδικοποιεί το πρώτο αμινοξύ της πρωτεΐνης, το δεύτερο κωδικόνιο το δεύτερο αμινοξύ κ.ο.κ.

Αρχικά, μετά από αυτή την ανακάλυψη οι βιολόγοι πίστευαν ότι η κάθε πρωτεΐνη κωδικοποιείται από μια μεγάλη συμβολοσειρά συνεχόμενων τριάδων, πράγμα το οποίο απλοποιεί πολύ τα πράγματα. Το πρόβλημα όμως προέκυψε στα τέλη της δεκαετίας του ‘70, με την ανακάλυψη των διακεκομμένων γονιδίων που παρατηρήθηκαν στον ανθρώπινο οργανισμό. Ένα γονίδιο, αναπαρίσταται συχνά από μία συλλογή συμβολοσειρών και όχι μόνο από μία συμβολοσειρά. Τίθεται πλέον το πρόβλημα του προσδιορισμού της θέσεως των γονιδίων πάνω στη γονιδιωματική αλληλουχία του DNA.

Όπως γνωρίζουμε, το ανθρώπινο γονιδίωμα είναι πολυπλοκότερο από το γονιδίωμα των βακτηρίων. Αυτό είναι μια λογική διαπίστωση, κρίνοντας με βάση του ότι είναι σαφώς μεγαλύτερο και ότι ο ανθρώπινος οργανισμός αποτελείται από περισσότερα όργανα με πολυπλοκότερες λειτουργίες. Σε έναν ευκαρυωτικό οργανισμό όμως, το μέγεθος του γονιδιώματος δεν φαίνεται να σχετίζεται με τη γενετική πολυπλοκότητά του. Σαν παράδειγμα, μπορούμε να φέρουμε τη σύγκριση του ανθρώπου με τη σαλαμάνδρα, όπου το γονιδίωμα της σαλαμάνδρας είναι δέκα φορές μεγαλύτερο από το ανθρώπινο γονιδίωμα. Η αιτία αυτού του φαινομενικά παραδόξου είναι η ύπαρξη

περιοχών DNA που δεν συμβάλουν στη δημιουργία πρωτεϊνών. Αυτά τα κομμάτια του γονιδιώματος χαρακτηρίζονται ως άχρηστο DNA (junk DNA) στην πραγματικότητα όμως δεν γνωρίζουμε επακριβώς τη χρησιμότητά του στις βιολογικές διεργασίες των κυττάρων. Οι περιοχές αυτές του γονιδίου όπως προαναφέραμε ονομάζονται ιντρόνια. Στο παραπάνω παράδειγμα, προφανώς το μέγεθος των γονιδίων της σαλαμάντρας έναντι των ανθρώπινων γονιδίων έγκειται στην ύπαρξη ποσοτήτων άχρηστου DNA.

Ένα διακεκομμένο γονίδιο, δεν αποτελείται επομένως από μια συνεχή αλληλουχία. Είναι κάτι ανάλογο με ένα πολυσέλιδο άρθρο σε ένα περιοδικό. Αρχίζει από τη σελίδα 1 και έχει συνεχόμενες πληροφορίες μέχρι τη σελίδα 4, μεσολαβούν σελίδες διαφημίσεων (άχρηστες πληροφορίες) και αυτή η εναλλαγή συνεχίζεται μέχρι το τέλος του άρθρου και είναι τόσο συχνή ανάλογα με το μέγεθός του. Δεν κατανοούμε ακόμη πλήρως την ύπαρξη των ιντρονίων (άχρηστων τμημάτων) ανάμεσα στα εξόνια. Η κατάσταση γίνεται ακόμη πιο περίπλοκη με τη διαπίστωση ότι υπάρχει διαφορά οργάνωσης αυτών των τμημάτων στα γονίδια διαφορετικών οργανισμών. Ένα γονίδιο στο ανθρώπινο γονιδίωμα είναι οργανωμένο διαφορετικά από το σχετιζόμενο γονίδιο στο γονιδίωμα της σαλαμάνδρας. Τα γονίδια στον ανθρώπινο οργανισμό αποτελούν το 3% του γονιδιώματός του. Η πρόβλεψη γονιδίων, αποτελεί ένα από τα σημαντικότερα προβλήματα στο χώρο της βιοπληροφορικής και μέχρι σήμερα δεν υπάρχει κάποιος αλγόριθμος αναγνώρισης γονιδίων που να παρέχει πλήρως αξιόπιστα αποτελέσματα. Σε αντίθεση με τους ευκαρυωτικούς οργανισμούς, οι προκαρυωτικοί δε διαθέτουν διακεκομμένα γονίδια, με αποτέλεσμα την ύπαρξη απλούστερων και αποδοτικότερων αλγορίθμων για το πρόβλημα της πρόβλεψής τους.

Χωρίς να μπούμε σε περεταίρω λεπτομέρειες, αξίζει να αναφέρουμε ότι οι μέθοδοι που χρησιμοποιούν συνήθως οι ερευνητές για την πρόβλεψη των θέσεων των γονιδίων χωρίζονται σε δύο κατηγορίες.

Στις μεθόδους που βασίζονται στη στατιστική προσέγγιση της πρόβλεψης αναζητώντας χαρακτηριστικά που εμφανίζονται συχνά μόνο στα γονίδια, και στις μεθόδους πρόβλεψης που βασίζονται στην ομοιότητα των γονιδίων που προκύπτει από την εύρεση ομοιοτήτων μεταξύ ενός γονιδίου με πρόσφατα προσδιορισμένη αλληλουχία με ένα ήδη γνωστό γονίδιο.

1.7.4 Στοίχιση αλληλουχιών (sequence alignment)

Η σύγκριση βιομοριακών ακολουθιών για τη βιοπληροφορική αποτελεί ένα μεγάλο κεφάλαιο έρευνας.

Ανάμεσα στα τόσα διαφορετικά είδη ζωντανών οργανισμών που υπάρχουν στον πλανήτη μας, υπάρχουν εξαιρετικές ομοιότητες σε μοριακό επίπεδο.

Όλοι οι οργανισμοί είναι δομημένοι με κύτταρα, τα οποία αποτελούνται από δύο είδη μορίων. Το DNA και τις πρωτεΐνες. Εξαιτίας της γραμμικής δομής των δύο αυτών μακρομορίων η έκφρασή τους ως αλληλουχίες-συμβολοσειρές και κατ' επέκταση η σύγκριση τους αποτελεί μια σύνηθες πρακτική για την μελέτη τους.

Οι λόγοι που μας οδηγούν στη σύγκριση ακολουθιών DNA και πρωτεϊνών είναι πάρα πολλοί. Ενδεικτικά θα μπορούσαμε να αναφέρουμε τον προσδιορισμό των γονιδίων καθώς και την εύρεση κοινών μοτίβων. Αυτό παρουσιάζει μεγάλο βιολογικό ενδιαφέρον.

Η μετάλλαξη του DNA αποτελεί μια φυσιολογική διαδικασία της εξέλιξης. Η ύπαρξη λαθών κατά τη διαδικασία της αντιγραφής του DNA δημιουργεί εισαγωγή νέων νουκλεοτιδίων ή ακόμη και διαγραφή υπαρχόντων στην γονιδιακή ακολουθία με αποτέλεσμα την αλλαγή- αλλοίωση κάποιων τμημάτων της. Τα τμήματα του DNA (ή πρωτεϊνών) που επηρεάζονται από αυτή τη διαδικασία είναι εκείνα τα οποία είναι λιγότερο υπεύθυνα για τη λειτουργία των ζωντανών οργανισμών. Αντίθετα, τα τμήματα που σε μοριακό επίπεδο είναι υπεύθυνα για τις βασικές λειτουργίες εμφανίζουν υψηλή σταθερότητα και ανθεκτικότητα στο φαινόμενο της μετάλλαξης.

Επομένως, η εύρεση κοινών μοτίβων ανάμεσα σε δύο ή και περισσότερες συμβολοσειρές στοχεύει στον προσδιορισμό αυτών των αλληλουχιών που είναι υπεύθυνες για τις βασικές λειτουργίες του οργανισμού.

Στο χώρο των πρωτεϊνών, με τη σύγκριση πρωτεϊνικών αλληλουχιών μπορούμε να καθορίσουμε τη δομή, τη σχέση μεταξύ των πρωτεϊνών και

τέλος να μπορέσουμε να βγάλουμε συμπεράσματα για τις βιολογικές λειτουργίες των νέων γονιδίων.

Με την εύρεση μιας νέα πρωτεΐνης, οι βιολόγοι γνωρίζουν ελάχιστα πράγματα για αυτή. Η άμεση πειραματική προσέγγιση συνήθως είναι πολύ χρονοβόρα. Μια συνηθισμένη τακτική για να βρούμε τις λειτουργίες της νέας πρωτεΐνης είναι να τη συγκρίνουμε και να βρούμε ομοιότητες με ομόλογές της που έχουν μελετηθεί είδη και είναι αποθηκευμένες σε μια βιολογική βάση δεδομένων.

Η σύγκριση αλληλουχιών όμως δεν περιορίζεται μόνο σε αλληλουχίες του ίδιου γονιδιώματος. Μεγάλο ενδιαφέρον παρατηρείτε στη σύγκριση αλληλουχιών διαφορετικών γονιδιωμάτων.

Διαφορετικοί οργανισμοί διαθέτουν καταπληκτικές ομοιότητες στη γονιδιακή τους δομή. Δυο οργανισμοί που ανήκουν στην οικογένεια των θηλαστικών, μπορούν να έχουν μέχρι και 99% ταύτιση στο DNA τους. Το DNA του ανθρώπου με το DNA του χιμπατζή ταυτίζονται κατά 98.4%.

Είναι πραγματικά εκπληκτικό ότι η ύπαρξη ενός Mozart ή ενός Einstein οφείλεται στη διαφοροποίηση ενός 1.6%.

Συγκρίνοντας αλληλουχίες ειδών που σχετίζονται μεταξύ τους ελπίζουμε στην καλύτερη κατανόηση της γλώσσας του DNA δίνοντας φως στην πορεία της εξέλιξης των ειδών. Ο τρόπος σύγκρισης αλληλουχιών γίνεται κυρίως με τη μέθοδο της στοίχισής τους.

Υπάρχουν δύο βασικοί τύποι στοίχισης αλληλουχιών.

- Η στοίχιση κατά ζεύγη (pairwise alignment) και η πολλαπλή στοίχιση (multiple alignment). Στη στοίχιση κατά ζεύγη, επιδιώκεται η εύρεση ομολόγων ενός γονιδίου ή μιας πρωτεΐνης από μια βάση δεδομένων. Η αναζήτηση αυτή επιτυγχάνεται βάση των κριτηρίων της ομοιότητας (similarity) και της ομολογίας (homology).

  Ως κριτήριο ομοιότητας χρησιμοποιείται το ποσοστό ταύτισης βάσεων για τα γονίδια και το ποσοστό ταύτισης αμινοξέων για τις πρωτεΐνες.

Κατά την ομολογία δεν έχουμε κάποια συγκεκριμένα κριτήρια ελέγχου παρά συμπερασματικά ανάλογα με τις ομοιότητες που παρατηρούνται καταλήγουμε στο αν δύο αλληλουχίες είναι ομόλογες, δηλαδή έχουν κοινή εξελικτική προέλευση ή όχι.

Για τη στοίχιση δύο αλληλουχιών υπάρχουν δύο μέθοδοι. Η τοπική στοίχιση (local alignment) και η ολική στοίχιση (global alignment). Κατά την τοπική στοίχιση αναζητούνται ταιριάσματα σε υποσύνολα χαρακτήρων των αλληλουχιών με γνώμονα την εύρεση συγγενικών περιοχών, ενώ κατά την ολική στοίχιση πραγματοποιείται αναζήτηση σε όλο το εύρος τους.

- Ολική στοίχιση μπορεί να πραγματοποιηθεί και με αλλεπάλληλες τοπικές μεθόδους στοίχισης μεταξύ των δύο αλληλουχιών.

  Οι μέθοδοι ολικής στοίχισης (multiple alignment) εφαρμόζονται για την εύρεση κοινών στοιχείων μεταξύ περισσότερων των δύο αλληλουχιών. Δεν στοιχίζεται μια νέα αλληλουχία με μια αλληλουχία είδη γνωστή και καταχωρημένη σε μια βάση δεδομένων, αλλά νέες αλληλουχίες με σκοπό την εύρεση συγγενικών περιοχών και την ανανέωση των βιολογικών βάσεων δεδομένων με νέες εγγραφές.

Δημοφιλείς αλγόριθμοι που χρησιμοποιούνται για την στοίχιση ακολουθιών είναι ο Needleman-Wunsch που χρησιμοποιείται για ολική κατά ζεύγη στοίχιση και ο Smith - Waterman που χρησιμοποιείται για ολική αλλά και τοπική στοίχιση.

1.7.5. Πρόβλεψη δομής πρωτεΐνης (protein structure prediction)

Η δομή μιας διατεταγμένης πρωτεΐνης είναι απαραίτητη για την κατανόηση της λειτουργίας της. Ακόμη και αν ο αριθμός των πρωτεϊνών που έχει καθοριστεί με πειραματικές μελέτες είναι αρκετά αυξημένος στις μέρες μας, υπάρχει ένας εξίσου μεγάλος αριθμός πρωτεϊνών με άγνωστη δομή και

λειτουργίες όπου δεν παρουσιάζει καμία ομολογία με είδη γνωστές πρωτεΐνες που βρίσκονται καταχωρημένες σε βάσεις δεδομένων. Δυστυχώς για την εύρεση της λειτουργίας της δεν είναι αρκετή η ανάλυση μόνο της αμινοξικής αλληλουχίας. Επιβάλλεται ο καθορισμός και η μελέτη της τρισδιάστατης δομής της.

Η πρόβλεψη της δομής μιας πρωτεΐνης από την αλληλουχία των αμινοξέων της αποτελεί το «ιερό δισκοπότηρο» για τη δομική βιολογική κοινότητα. Παρά τις προσπάθειες δεκαετιών παραμένει έως και σήμερα ένα εξαιρετικά δύσκολο πρόβλημα λόγω της αναδιπλούμενης τρισδιάστατης δομής της αλλά και των πολλών βαθμών ελευθερίας που ορίζουν τη δομή της.

Όλες οι θεωρητικές και υπολογιστικές μελέτες που γίνονται για αυτό το σκοπό επιβάλλεται να επικυρωθούν από έλεγχο. Αυτό επιτυγχάνεται συνήθως με την δημιουργία ενός συστήματος προβλέψεων, όπου με την προσπάθεια αναπαραγωγής των πτυχώσεων γνωστών μικρών πρωτεϊνών και τις κατάλληλες δοκιμές, το σύστημα θα «εκπαιδευτεί» να «θυμάται» δομές και να μπορεί να προβλέψει νέες.

ΚΕΦΑΛΑΙΟ 2: Στοίχιση Αλληλουχιών

2.1 Εισαγωγή

Όπως προαναφέραμε, η αναζήτηση ομοιοτήτων μεταξύ δυο ή και περισσότερων βιολογικών αλληλουχιών αποτελεί ένα σημαντικό θέμα για τη σύγχρονη μοριακή βιολογία. Ο κύριος τρόπος προσέγγισης αυτού του σκοπού είναι η μεταξύ τους στοίχιση.

Η στοίχιση ακολουθιών πραγματοποιείται για να μπορούμε να βγάλουμε συμπεράσματα που αφορούν τη δομή, τη λειτουργικότητα και την εξέλιξη βιολογικών ακολουθιών. Η ομοιότητα ακολουθιών νουκλεϊκών οξέων μπορεί να συνεπάγεται την ίδια λειτουργία ή τον ίδιο ρυθμιστικό ρόλο, ενώ στην περίπτωση πρωτεϊνικών ακολουθιών την ίδια βιοχημική λειτουργία ή την ίδια τριτοταγή δομή. Οι συγκρινόμενες ακολουθίες μπορεί να προέρχονται από τον ίδιο ή και από διαφορετικούς οργανισμούς. Όταν προέρχονται από διαφορετικούς οργανισμούς, η πιθανή ομοιότητά τους οδηγεί στο συμπέρασμα της κοινής καταγωγής τους (κοινού προγόνου) και με βάση τον βαθμό ομοιότητάς τους κατατάσσονται και στην ανάλογη θέση στο αντίστοιχο φυλογενετικό δέντρο. Το DNA και οι πρωτεΐνες είναι προϊόντα της εξέλιξης. Τα δομικά στοιχεία που απαρτίζουν αυτά τα μακρομόρια, οι νουκλεοτιδικές βάσεις και τα αμινοξέα, καθορίζουν την κύρια δομή τους και μπορούν να χαρακτηριστούν ως μοριακά απολιθώματα που κωδικοποιούν την ιστορία εκατομμυρίων χρόνων εξέλιξης.

Εάν και οι επιλεγμένες αλληλουχίες με την πάροδο του χρόνου συσσωρεύουν μεταλλάξεις και μπορεί να αποκλίνουν, σε ορισμένα τμήματά τους τα ίχνη της εξέλιξης μπορεί να παραμένουν και να επιτρέπουν την ταυτοποίηση της κοινής τους καταγωγής.

Μια ορθή στοίχιση μπορεί να απεικονίσει την εξελικτική ιστορία δύο ακολουθιών.

“Για πολλές πρωτεϊνικές αλληλουχίες η ιστορία της εξέλιξης μπορεί να μας γυρίσει πίσω ένα έως και δύο εκατομμύρια χρόνια.” (William Pearson).

Υπάρχουν δύο τρόποι στοίχισης βιολογικών αλληλουχιών. Η στοίχιση κατά ζεύγη (pairwise alignment) κατά την οποία στοιχίζονται δύο ακολουθίες και η πολλαπλή στοίχιση (multiple alignment) όπου στοιχίζονται περισσότερες των δύο. Οποιοδήποτε τρόπο και να ακολουθήσουμε μπορούμε να εφαρμόσουμε είτε τοπική στοίχιση (local alignment) κατά την οποία αναζητούνται ομοιότητες σε κομμάτια των δύο ή και περισσότερων ακολουθιών, είτε ολική στοίχιση (global alignment), όπου η στοίχιση γίνεται σε όλο το μήκος των ακολουθιών. Στην παρούσα εργασία θα ασχοληθούμε με την ολική κατά ζεύγη στοίχιση (pairwise alignment).

Η διαδικασία που εφαρμόζεται περιλαμβάνει την ευθυγράμμιση των αλληλουχιών, την εύρεση των ομοιοτήτων τους και την εξαγωγή συμπερασμάτων για το αν οι αλληλουχίες σχετίζονται πραγματικά μεταξύ τους ή η συσχέτιση που έχει προκύψει αποτελεί τυχαίο γεγονός.

Πως μπορούμε όμως να ορίσουμε πότε υπάρχει συσχετισμός μεταξύ αλληλουχιών?

Η διαδικασία της στοίχισης αλληλουχιών βασίζεται στην αναζήτηση κοινών μοτίβων (patterns) με βάση τη σύγκριση στοιχείου με στοιχείο ανάμεσα στις αλληλουχίες που έχουν κάποια σχέση μεταξύ τους.

Για να συγκρίνουμε νουκλεοτίδια ή αμινοξέα που εμφανίζονται σε αντίστοιχες θέσεις αρχικά θα πρέπει να ορίσουμε αυτές τις αντιστοιχίες.

Στη συνέχεια παραθέτουμε τους ορισμούς που έχουν να κάνουν με τη βιολογική ερμηνεία της ομοιότητας που προκύπτει από τη σύγκριση αλληλουχιών.

- **Homologs** (ομόλογα): Ένα γονίδιο Α που σχετίζεται εξελικτικά με ένα δεύτερο γονίδιο Β, καθώς προέρχεται από μια κοινή προγονική αλληλουχία. Ο όρος ομόλογο μπορεί να ισχύει για τη σχέση μεταξύ των γονιδίων που χωρίζονται από την εκδήλωση της ενδογένεσης (ορθόλογα) ή τη σχέση μεταξύ γονίδιων που προήλθαν από γενετική επανάληψη (παράλογα).
- **Orthologs** (ορθόλογα): Ορθόλογα γονίδια είναι ομόλογες ακολουθίες που βρίσκονται σε διαφορετικά είδη και οι οποίες εξελίχθηκαν από έναν

κοινό προγονικό γονίδιο από ενδογένεση. Υπό κανονικές συνθήκες, τα ορθόλογα διατηρούν την ίδια λειτουργία κατά τη διάρκεια της εξέλιξης. Ο προσδιορισμός των ορθολόγων είναι κρίσιμος για την αξιόπιστη πρόβλεψη της λειτουργίας των γονιδίων, ιδιαίτερα σε πρόσφατα γονιδιώματα.

- **Paralogs** (παράλογα): Παράλογα είναι τα ομόλογα γονίδια που σχετίζονται με την επανάληψη μέσα σε ένα γονιδίωμα. Ενώ τα ορθόλογα διατηρούν την ίδια λειτουργία κατά τη διάρκεια της εξέλιξης, τα παράλογα εξελίσσονται σε νέες λειτουργίες, ακόμη και αν αυτά σχετίζονται με τη λειτουργία του αρχικού γονιδίου ((Neil C. Jones et al., 2004).

Όταν στοιχίζουμε δύο αλληλουχίες, υποθέτουμε ότι μοιράζονται κάποιον κοινό πρόγονο και αναζητούμε τις θεμελιώδεις αλλαγές που επήλθαν κατά τη διάρκεια της απόκλισης από το κοινό προγονικό μόριο. Οι τύποι αυτών των αλλαγών είναι τρείς: Η αντικατάσταση (substitution), η προσθήκη-εισαγωγή (insert) και η εξάλειψη- διαγραφή (delete).

Στην περίπτωση της στοίχισης δύο ομόλογων ακολουθιών τα στοιχισμένα κατάλοιπα (residues) που δεν ταυτίζονται αντιπροσωπεύουν τις αντικαταστάσεις που έχουν επέλθει στην μια αλληλουχία λόγω της εξελικτικής της πορείας, ενώ οι περιοχές στις οποίες δεν υπάρχει στοίχιση ερμηνεύονται είτε ως προσθήκη στη μία ακολουθία είτε ως εξάλειψη στην άλλη και εμφανίζονται ως κενά (gaps).

Για την κατανόηση της μεθόδου της στοίχισης παρατίθεται το παρακάτω απλό παράδειγμα.

Έστω ότι έχουμε τις παρακάτω δύο αλληλουχίες DNA:

X= A A T C T G A T A G A A G C C C T A

Y= C C A A T C C A G A A C G C C C A

Μπορούμε να μετασχηματίσουμε την X σε Y (ή αντίστροφα) με μια σειρά απλών αλλαγών βάσεων, μεταλλάξεων ή επεμβατικών λειτουργιών. Οι επιτρεπτές λειτουργίες είναι:

- Ομοιότητα (match): παραμένει η βάση αμετάβλητη
- Μη- ομοιότητα (mismatch): αντικατάσταση μιας βάσης από διαφορετική βάση
- Κενό (gap): εισαγωγή / διαγραφή μιας βάσης

Όπως καταλαβαίνουμε, χρησιμοποιώντας τις παραπάνω κινήσεις μπορούμε να πετύχουμε πολλούς διαφορετικούς συνδυασμούς στην προσπάθεια μετασχηματισμού της μιας αλληλουχίας στην άλλη. Δηλαδή μπορούμε να πετύχουμε πολλές διαφορετικές στοιχίσεις.

Μια τέτοια στοίχιση είναι η παρακάτω:

Θέση: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

X - - A A T C T G A T A G A A G C C C T A

: : : : : : * : : * : : : : :

Y C C A A T C - G A G A - A C G C C C - A

Με **:** συμβολίζεται η ομοιότητα των χαρακτήρων των δύο ακολουθιών, με * η ανομοιότητα και με – το κενό λόγω της εισαγωγής μιας βάσης σε μια αλληλουχία, ή αντίστοιχα η διαγραφή μιας βάσης στην άλλη αλληλουχία.

Στο παραπάνω παράδειγμα, για να μετασχηματίσουμε την αλληλουχία X στην αλληλουχία Y απαιτούνται τα εξής βήματα:

- Αντικατάσταση της βάσης T από τη βάση G στη θέση 10.
- Αντικατάσταση της βάσης A από τη βάση C στη θέση 14.
- Εισαγωγή της βάσης C στις θέσεις 1 και 2.

- Διαγραφή της βάσης T στις θέσεις 7 και 19.
- Διαγραφή της βάσης G στη θέση 12.

Η παραπάνω αντιστοίχιση του παραδείγματος περιέχει 13 ομοιότητες, 2 ανομοιότητες και 5 κενά.

Για να αξιολογήσουμε την συγκεκριμένη αντιστοίχιση χρησιμοποιούμε την έννοια της ομολογίας, η οποία ορίζεται ως το ποσοστό των ομοιοτήτων στο πλήρες μήκος της αντιστοιχίας, το οποίο στη συγκεκριμένη περίπτωση είναι: (13/20)*100=65%.

Εύκολα διαπιστώνει κανείς ότι χρησιμοποιώντας διαφορετική ακολουθία επεμβάσεων στις δύο αλληλουχίες μπορεί να πετύχει καλύτερη ή και χειρότερη αντιστοίχιση τους και κατά συνέπεια καλύτερο ή χειρότερο ποσοστό ομολογίας.

2.2. Σύστημα βαθμολόγησης (Scoring system).

Όπως είδαμε και στο παράδειγμα, για να επιλέξουμε την καλύτερη στοίχιση (με τον όρο καλύτερη στοίχιση εννοούμε τη στοίχιση που θα μας δώσει τις περισσότερες πληροφορίες) χρειαζόμαστε ένα σύστημα βαθμολόγησης. Το εργαλείο με το οποίο επιλέγουμε αν δυο χαρακτήρες είναι ταυτόσημοι ή παρόμοιοι ή αν είναι καλύτερη επιλογή να τοποθετηθεί κενό σε κάποια θέση είναι η αντικειμενική συνάρτηση (objective function). Η αντικειμενική συνάρτηση αποτελεί το μηχανισμό με τον οποίο θα πραγματοποιηθεί η αντιστοίχιση. Ορίζεται έτσι ώστε πέρα από την βέλτιστη στοίχιση από μαθηματικής πλευράς να παρέχει και τη βέλτιστη στοίχιση από βιολογικής πλευράς. Αυτό επιτυγχάνεται αξιοποιώντας κατάλληλα τα βιολογικά δεδομένα των αλληλουχιών τα οποία έχουν να κάνουν με τη δομή τους, τη λειτουργία τους καθώς και την εξελικτική τους πορεία. Στην πράξη, πρόκειται για ένα εξαιρετικά δύσκολο εγχείρημα, οπότε συνήθως περιορίζεται σε πληροφορίες που αφορούν την ομοιότητα των αλληλουχιών στην πρωτοταγή τους δομή. Ο συνηθέστερος τύπος αντικειμενικής συνάρτησης που χρησιμοποιείται βασίζεται στην απόδοση μιας τιμής σε κάθε ζεύγος

κατάλοιπων που στοιχίζεται ανάλογα με την ομοιότητά τους, και στην αφαίρεση μιας τιμής στην στοίχιση καταλοίπου με κενό. Ο υπολογισμός της τιμής που αποδίδεται σε κάθε ζεύγος πέρα από την περίπτωση ταύτισης που είναι η μέγιστη, βασίζεται στην συχνότητα μετάλλαξης των καταλοίπων. Συντηρητικές αντικαταστάσεις (αντικαταστάσεις βάσεων ή αμινοξέων από συγγενικές τους βάσεις/αμινοξέα όπου δεν επηρεάζεται η λειτουργία της αλληλουχίας) το οποίο είναι και το συνηθέστερο στη στοίχιση ακολουθιών, καταγράφουν θετική βαθμολογία. Αντιθέτως οι μη συντηρητικές αλλαγές με τις οποίες παρατηρείται και διαφοροποίηση στη λειτουργία της αλληλουχίας, είναι σπανιότερες και βαθμολογούνται αρνητικά. Κατάλοιπα που μεταλλάσσονται συχνά αντιστοιχούν σε θετικές αποτιμήσεις, ενώ κατάλοιπα που μεταλλάσσονται σπάνια αντιστοιχούν σε αρνητικές αποτιμήσεις.

Η άθροιση των επιμέρους βαθμών μας δίνει και το τελικό score της στοίχισης. Ο συγκεκριμένος τρόπος αποτίμησης της στοίχισης των δύο ή περισσότερων αλληλουχιών προϋποθέτει την μη ύπαρξη αλληλεξαρτήσεων ανάμεσα σε μεταλλάξεις που παρουσιάζονται σε διαφορετικά κατάλοιπα της αλληλουχίας. Η συγκεκριμένη παραδοχή, όσον αφορά πρωτεϊνικές αλληλουχίες και αλληλουχίες βάσεων DNA αποδεικνύεται ευσταθής παρόλο που γνωρίζουμε ότι οι αλληλεπιδράσεις που παρατηρούνται μεταξύ των αμινοξέων σε μια πρωτεΐνη παίζουν σημαντικό ρόλο στη διαμόρφωση της τριτοταγούς της δομής. Στην περίπτωση όμως των RNA ακολουθιών οι αλληλεξαρτήσεις των μεταλλάξεων είναι κάτι το οποίο δεν μπορούμε να το αγνοήσουμε.

Οι τιμές των αντιστοιχίσεων περιέχονται στους πίνακες αντικαταστάσεως (substitution matrices) και έχουν προκύψει με πειραματικές μεθόδους.

2.3. Πίνακες αντικατάστασης (Substitution Matrices)

Οι πίνακες αντικατάστασης ή αλλιώς βαθμολογικοί πίνακες (score matrices) είναι οι πίνακες οι οποίοι παρέχουν πληροφορίες σχετικά με τους ρυθμούς αντικατάστασης των αμινοξέων και των νουκλεϊκών οξέων. Στην περίπτωση των νουκλεϊκών οξέων οι πίνακες αντικατάστασης είναι απλοί. Μπορεί να είναι της μορφής:

$$S = \begin{pmatrix} s_{a,a} & s_{a,c} & s_{a,g} & s_{a,t} \\ s_{c,a} & s_{c,c} & s_{c,g} & s_{a,t} \\ s_{g,a} & s_{g,c} & s_{g,g} & s_{g,t} \\ s_{t,a} & s_{t,c} & s_{t,g} & s_{t,t} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Όπου παρατηρούμε ότι στην περίπτωση ταύτισης των βάσεων έχουμε τη μέγιστη τιμή (κύρια διαγώνιος) που είναι ένα, και σε κάθε άλλη αντικατάσταση έχουμε μηδενικό score.

Μια άλλη μορφή του πίνακα αντικατάστασης που χρησιμοποιείται για τη στοίχιση νουκλεϊκών οξέων είναι:

$$S = \begin{pmatrix} s_{a,a} & s_{a,c} & s_{a,g} & s_{a,t} \\ s_{c,a} & s_{c,c} & s_{c,g} & s_{a,t} \\ s_{g,a} & s_{g,c} & s_{g,g} & s_{g,t} \\ s_{t,a} & s_{t,c} & s_{t,g} & s_{t,t} \end{pmatrix} = \begin{pmatrix} 1 & -1 & -1/2 & -1 \\ -1 & 1 & -1 & -1/2 \\ -1/2 & 1 & 1 & -1 \\ -1 & -1/2 & -1 & 1 \end{pmatrix}$$

Και σε αυτή την περίπτωση παρατηρούμε ότι στην κύρια διαγώνιο έχουμε τη μέγιστη τιμή, ενώ σε περίπτωση μη ταύτισης έχουμε αρνητική βαθμολόγηση ανάλογη με την πιθανότητα αντικατάστασης των συγκεκριμένων βάσεων.

Στην περίπτωση της πρωτεϊνικής στοίχισης οι πίνακες αντικατάστασης παρέχουν πληροφορίες σχετικά με τη συγγένεια, τις ιδιότητες και την οικογένεια των πρωτεϊνών που ανήκουν. με συνέπεια η δημιουργία τους να μην είναι και τόσο απλή υπόθεση. Ως συνέπεια των παραπάνω, παρατηρούμε ότι η στοίχιση πρωτεϊνικών αλληλουχιών αποτελεί πιο πολύπλοκη και επίπονη διαδικασία σε σύγκριση με τη στοίχιση DNA αλληλουχιών. Παρόλα αυτά προτιμάται λόγω των περισσότερων και ουσιαστικότερων συμπερασμάτων που μας παρέχει. Αξίζει επίσης να αναφέρουμε ότι η σύγκριση τριτοταγών δομών των πρωτεϊνών σε σχέση με την ακολουθία των αμινοξέων τους παρέχει πιο λεπτομερείς πληροφορίες για την λειτουργία της πρωτεΐνης και την εξελικτικής της πορεία, λόγω της σταθερότητας που παρουσιάζει η τρισδιάστατη δομή της. Επομένως η σύγκριση των τρισδιάστατων δομών δύο (ή και περισσότερων) πρωτεϊνών σε σχέση με τη

σύγκριση των σειρών τους μοιάζει αποδοτικότερη. Το πρόβλημα όμως ενδείκνυται στο ότι ο αριθμός των γνωστών σήμερα πρωτεϊνικών σειρών είναι κατά τριάντα φορές μεγαλύτερος από τον αριθμό των τριτοταγών δομών που γνωρίζουμε. Επίσης, οι διαθέσιμες μέθοδοι σύγκρισης σειρών είναι πολύ ταχύτερες.

Κατά την εξελικτική πορεία μιας πρωτεΐνης παρατηρούνται συχνότερες αντικαταστάσεις μεταξύ αμινοξέων με κοινές χημικές ιδιότητες σε σχέση με αμινοξέα που ανήκουν σε διαφορετική οικογένεια. Έτσι, οι πίνακες αντικατάστασης δημιουργήθηκαν λαμβάνοντας υπόψη τις ομοιότητες των αμινοξέων, καθώς και την συχνότητα εμφάνισής τους στα πρωτεϊνικά δείγματα που μελετήθηκαν εργαστηριακά κατά την προσπάθεια δημιουργίας τους.

Πρόκειται για έναν συμμετρικό πίνακα 40 θέσεων (20x20) όπου καλύπτονται όλοι οι συνδυασμοί και των είκοσι διαφορετικών αμινοξέων. Οι τιμές του πίνακα εκφράζουν την ομοιότητα δύο αμινοξέων ή την απόσταση, το κόστος δηλαδή της αντικατάστασης του ενός αμινοξέος από το άλλο. Είναι προφανές, ότι η μέγιστες τιμές βρίσκονται στην κύρια διαγώνιο του πίνακα όπου υπάρχει η πλήρης ταύτιση των στοιχείων του. Η εύρεση των τιμών ενός πίνακα αντικατάστασης προκύπτει από την εφαρμογή ενός πιθανοθεωρητικού μοντέλου σε συνδυασμό με τν βιολογική ερμηνεία των αντικαταστάσεων.

Ο τύπος υπολογισμού των τιμών είναι:

$S_{i,j}= \log(q_{i,j}/ p_i p_j)$, όπου $S_{i,j}$ είναι το στοιχείο του πίνακα S το οποίο προκύπτει ως ο λογάριθμος του λόγου δύο πιθανοτήτων:

$q_{i,j}$: Η πιθανότητα τα αμινοξέα I και j να έχουν έναν κοινό πρόγονο, να υπάρχει δηλαδή μεταξύ τους συγγενική σχέση λόγω εξέλιξης.

p_i, p_j: Οι πιθανότητες τα αμινοξέα I και j αντίστοιχα, να εμφανίζονται τυχαία, οπότε το γινόμενό τους εκφράζει την πιθανότητα τυχαίας αντικατάστασή τους.

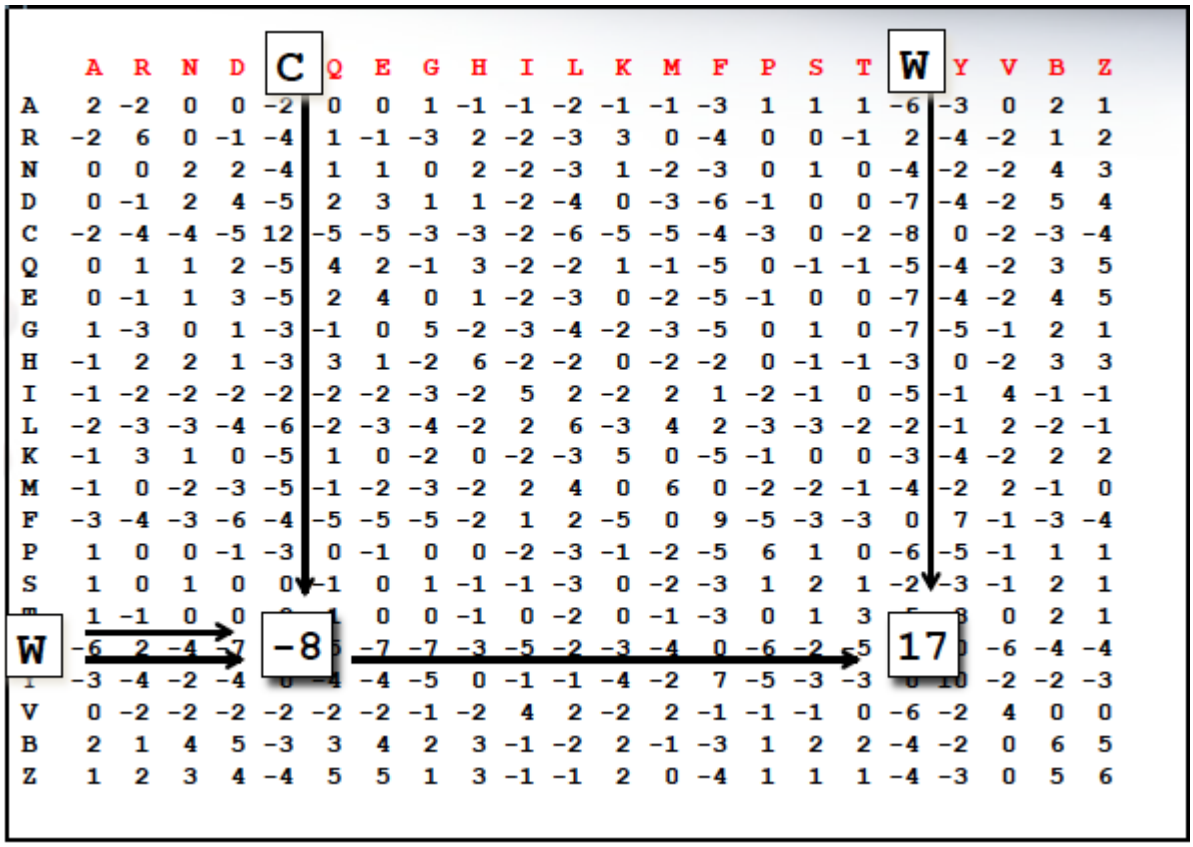
Οι πιο διαδεδομένοι πίνακες αντικατάστασης για την αντιστοίχιση πρωτεϊνών οι οποίοι χρησιμοποιούνται μέχρι και σήμερα είναι οι πίνακες PAM (Dayhoff et al., 1978) και οι πίνακες BLOSUM (Henikoff and Henikoff, 1992).

2.3.1 Πίνακες PAM

Η ιδέα δημιουργίας των πινάκων αντικατάσταση PAM (Point Accepted Mutation) ξεκίνησε από τη Margaret Dayhoff το 1978 η οποία ασχολήθηκε με τις αντικαταστάσεις αμινοξέων σε συγγενικές πρωτεϊνικές ακολουθίες με βάση ένα μακροβιανό μοντέλο εξέλιξης. Σύμφωνα με αυτό το μοντέλο, οι παρούσες πρωτεϊνικές ακολουθίες απέκλιναν από τις προγονικές μέσω αποδεκτών σημειακών μεταλλάξεων. Για παράδειγμα, η μετάλλαξη ενός αμινοξέος Χ σε ένα αμινοξύ Υ μπορεί να γίνει μόνο αν το αμινοξύ Χ έχει τις ίδιες ή παρόμοιες φυσικοχημικές ιδιότητες με το αμινοξύ Υ, έτσι ώστε να μην υπάρχουν δομικές ή λειτουργικές μεταβολές στις πρωτεΐνες. Για την κατασκευή τους χρησιμοποιήθηκαν 71 οικογένειες πρωτεϊνών στις οποίες η ομοιότητες τους είναι στο 85% (τα αμινοξέα των πρωτεϊνικών αλληλουχιών διαφέρουν μεταξύ τους το πολύ κατά 15%). Η Dayhoff, δημιουργώντας ένα θεωρητικό φυλογενετικό δέντρο κατάφερε να προβλέψει ποια αμινοξέα έχουν τη μεγαλύτερη πιθανότητα εμφάνισης στις προγονικές ακολουθίες. Ο πρώτος πίνακας PAM είναι ο PAM-1 ο οποίος ονομάστηκε έτσι γιατί απευθύνεται σε πρωτεϊνικές αλληλουχίες στις οποίες ο επιτρεπτός αριθμός μεταλλάξεων δεν ξεπερνά το 1% του συνολικού μήκους τους. Πολλαπλασιάζοντας τον πίνακα PAM-1 μπορούμε να αυξήσουμε το επιτρεπόμενο όριο μεταλλάξεων. Οι πιο διαδεδομένοι πίνακες PAM που χρησιμοποιούνται είναι οι PAM-80, PAM-120 και PAM-250. Οι τιμές των πινάκων αντιπροσωπεύουν την απόσταση στην εξέλιξη (Evolutionary Distance). Έτσι, μεγαλύτερες τιμές δηλώνουν μεγαλύτερες αποστάσεις.

Μονάδα PAM: Έστω ότι έχουμε τις πρωτεϊνικές ακολουθίες S1 και S2 και έχουν απόσταση εξέλιξης (Evolutionary Distance) μιας μονάδας της κλίμακας PAM. Αυτό σημαίνει ότι εάν η ακολουθία S1 μετατράπηκε στην ακολουθία S2,ο επιτρεπόμενος μέσος όρος μετάλλαξης που ενσωματώθηκε στην πρωτεΐνη για να περάσει στους απογόνους του οργανισμού είναι ένα αμινοξύ ανά 100.

Παρακάτω βλέπουμε τον πιο διαδεδομένο πίνακα PAM, τον PAM-250:



Εικόνα 9: PAM 250

Μεγάλο μειονέκτημα για τη χρήση των συγκεκριμένων πινάκων αποτελεί το γεγονός ότι η δημιουργία τους βασίστηκε σε ένα μόνο πρωτότυπο σύνολο δεδομένων (Data Set) και οι πρωτεΐνες τις οποίες εξετάζει είναι πρωτεΐνες με περιορισμένο εύρος διαφορών (έχουν 85% ομοιότητα).

Επίσης, τα αποτελέσματα του είναι βασισμένα στην εξέταση μικρών σφαιρικών πρωτεϊνών, με αποτέλεσμα τα νούμερα των πινάκων να είναι προκατειλημμένα.

2.3.2. Πίνακες BLOSUM

Μια άλλη κατηγορία πινάκων αντικατάστασης ευρέως χρησιμοποιούμενη είναι οι πίνακες BLOSUM (Block Substitution Matrix) (Henikoff and Henikoff, 1992).

Οι πίνακες BLOSUM δημιουργήθηκαν με βάση τη στοίχιση συντηρημένων αμινοξέων χωρίς κενά (Blocks) που βρίσκονται σε ακολουθίες οι οποίες είναι εξελικτικά απομακρυσμένες μεταξύ τους. Οι πρωτεϊνικές σειρές κατανέμονται σε ομάδες (clusters) ανάλογα με το επίπεδο ομοιότητάς τους. Για να υπολογίσουν την πιθανότητα αντικατάστασης ενός αμινοξέος από ένα άλλο βασίζονται σε πειραματικά αποδεδειγμένες πιθανότητες αντικατάστασης. Ένας πίνακας BLOSUM-X είναι βασισμένος σε ομάδες σειρών χωρίς κενά που μοιράζονται το πολύ X% ποσοστό ομοιότητας. Συνεπώς, ο BLOSUM-62 αντιπροσωπεύει αλληλουχίες που βρίσκονται πιο κοντά μεταξύ τους απ' ότι ο BLOSUM-45. Επομένως, οι πίνακες BLOSUM με μικρότερους αριθμούς αντικατοπτρίζουν μεγαλύτερες εξελικτικές αποκλίσεις και αντίστροφα.

Τα πλεονεκτήματα των πινάκων BLOSUM έναντι των PAM πινάκων είναι ότι από τη στιγμή που χρησιμοποιούν περισσότερα δεδομένα για τη δημιουργία τους παρέχουν και πιο αξιόπιστα στατιστικά στοιχεία. Επίσης, το ότι χρησιμοποιούν για τη δημιουργία τους πολλαπλή τοπική στοίχιση σε πρωτεΐνες της ίδιας οικογένεια σε αντίθεση με την ολική στοίχιση των PAM πινάκων προσδίδει ένα επιπλέον στοιχείο αξιοπιστίας στο ότι συγκεντρώνουν τα στατιστικά στοιχεία των αντικαταστάσεων από τις περιοχές που έχουν σημασία και όχι από ολόκληρες τις ακολουθίες.

Στον παρακάτω πίνακα φαίνονται οι αντιστοιχίσεις των πινάκων PAM και BLOSUM:

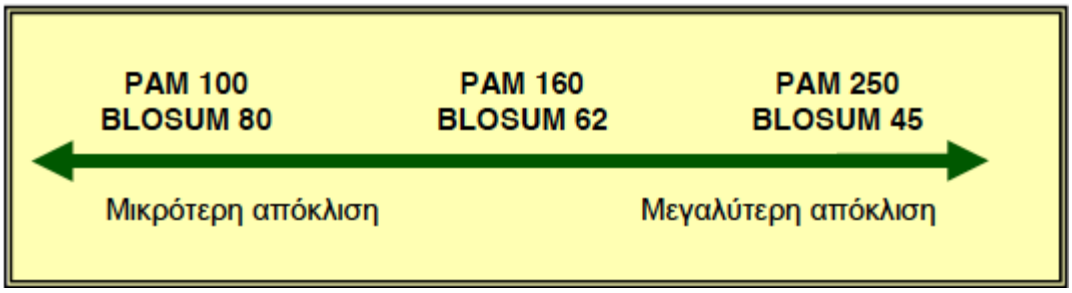
Πίνακας 3: Αντιστοίχιση PAM- BLOSUM

PAM-100	BLOSUM-90
PAM-120	BLOSUM-80
PAM-160	BLOSUM-60
PAM-200	BLOSUM-52
PAM-250	BLOSUM-45

Ανάλογα με το είδος των αλληλουχιών που θέλουμε να στοιχίσουμε χρησιμοποιούμε και τον αντίστοιχο πίνακα. Έτσι, για αλληλουχίες που έχουν

κοντινή εξελικτική συγγένεια χρησιμοποιείται κυρίως ο πίνακας PAM-60 ή ο πίνακας BLOSUM-80. Για αλληλουχίες που απέχουν πολύ μεταξύ τους ενδείκνυται ο πίνακας PAM-250 και οBLOSUM-45, ενώ οι πίνακες PAM-120 και BLOSUM-62 είναι για γενική χρήση.

Με βάση τα παραπάνω, παρατηρούμε ότι για τη σύγκριση στενά συγγενικών πρωτεϊνών θα πρέπει να χρησιμοποιήσουμε χαμηλότερους PAM πίνακες ή υψηλότερους BLOSUM, αντίθετα στη σύγκριση πρωτεϊνικών αλληλουχιών χαμηλού βαθμού συγγένειας οι πιο ενδεδειγμένοι πίνακες είναι οι υψηλοί PAM και χαμηλοί BLOSUM.



Εικόνα 10: PAM- BLOSUM 1

Γενικά, οι πίνακες BLOSUM καταγράφουν καλύτερες επιδόσεις σε προβλήματα τοπικής στοίχισης.

Ο πλέον διαδεδομένος πίνακας που χρησιμοποιείται για την αναζήτηση μιας αλληλουχίας σε πρωτεϊνικές βάσεις δεδομένων είναι οBLOSUM-62.

Παρακάτω βλέπουμε τον πιο διαδεδομένο πίνακα BLOSUM, τον BLOSUM-62:

	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	
C	9																				C
S	-1	4																			S
T	-1	1	5																		T
P	-3	-1	-1	7																	P
A	0	1	0	-1	4																A
G	-3	0	-2	-2	0	6															G
N	-3	1	0	-2	-2	0	6														N
D	-3	0	-1	-1	-2	-1	1	6													D
E	-4	0	-1	-1	-1	-2	0	2	5												E
Q	-3	0	-1	-1	-1	-2	0	0	2	5											Q
H	-3	-1	-2	-2	-2	-2	1	-1	0	0	8										H
R	-3	-1	-1	-2	-1	-2	0	-2	0	1	0	5									R
K	-3	0	-1	-1	-1	-2	0	-1	1	1	-1	2	5								K
M	-1	-1	-1	-2	-1	-3	-2	-3	-2	0	-2	-1	-1	5							M
I	-1	-2	-1	-3	-1	-4	-3	-3	-3	-3	-3	-3	-3	1	4						I
L	-1	-2	-1	-3	-1	-4	-3	-4	-3	-2	-3	-2	-2	2	2	4					L
V	-1	-2	0	-2	0	-3	-3	-3	-2	-2	-3	-3	-2	1	3	1	4				V
F	-2	-2	-2	-4	-2	-3	-3	-3	-3	-3	-1	-3	-3	0	0	0	-1	6			F
Y	-2	-2	-2	-3	-2	-3	-2	-3	-2	-1	2	-2	-2	-1	-1	-1	-1	3	7		Y
W	-2	-3	-2	-4	-3	-2	-4	-4	-3	-2	-2	-3	-3	-1	-3	-2	-3	1	2	11	W

Εικόνα 11: BLOSUM 62

2.4. Ποινές κενών (Gap penalties)

Εκτός από την επιλογή του πίνακα αντικατάστασης ένας άλλος παράγοντας που παίζει σημαντικό ρόλο στη στοίχιση αλληλουχιών είναι η επιλογή του τρόπου βαθμολόγησης των κενών. Η επιβολή ποινής κατά την εισαγωγή ενός κενού στην ακολουθία είναι απαραίτητη γιατί καθορίζει την απόφαση για το αν θα μπει κενό στη συγκεκριμένη θέση ή συμφέρει περισσότερο (στην τελική βαθμολόγηση) η επιλογή της στοίχισης δύο ανόμοιων καταλοίπων. Από βιολογικής άποψης, είναι ευκολότερο για μια αλληλουχία να δεχθεί την αντικατάσταση ενός καταλοίπου σε μια θέση παρά την εισαγωγή ή διαγραφή του.

Συνεπώς, οι εισαγωγές και οι διαγραφές (εισαγωγή κενών) θα πρέπει να είναι σπανιότερες στη στοίχιση αλληλουχιών απ’ ότι οι αντικαταστάσεις.

Εάν η ποινή που θα επιλεγεί για την εισαγωγή κενών δεν είναι η κατάλληλη, καθώς αν επιλεγούν πολύ μικρές τιμές οδηγούμαστε σε παράλογες στοιχίσεις από βιολογικής άποψης (η χρήση πολλών κενών μπορεί να οδηγήσει σε αντιστοιχία εντελώς ανόμοιων ακολουθιών), από την άλλη η μεγάλη ποινικοποίηση των κενών μας οδηγεί σε παράλογες στοιχίσεις χωρίς κενά. Ο τρόπος επιβολής ποινής για τη χρήση των κενών θα πρέπει να οδηγεί σε στοίχιση που θα προσεγγίζει όσο το δυνατόν περισσότερο τη

διαδικασία εξέλιξης των ακολουθιών. Καθοριστικό ρόλο για την επιλογή του κόστους των κενών παίζει η επιλογή του πίνακα αντικατάστασης. Έτσι, για παράδειγμα μια τιμή που θεωρείται μικρή όταν χρησιμοποιείται ο πίνακας PAM-10 μπορεί να είναι μεγάλη στην περίπτωση χρήσης του πίνακα PAM-250.

Οι ποιο γνωστοί μέθοδοι υπολογισμού κόστους των κενών είναι η Affine Gap Penalty και η Concave Gap Penalty.

Κατά την πρώτη μέθοδο, αποδίδεται μια ποινή για το πρώτο κενό που δημιουργείται και μια μικρότερη τιμή για την επέκταση της τιμής των κενών, η οποία υπολογίζεται με τη βοήθεια μιας γραμμικής συνάρτησης.

Η γραμμική συνάρτηση έχει την παρακάτω μορφή:

$$gap(k) = gop + (k-1) * gep,$$

όπου με k συμβολίζουμε τον αριθμό των κενών που θα εισαχθούν, gop είναι η τιμή του αρχικού κενού και gep η τιμή των κενών που ακολουθούν.

Η Concave Gap Penalty μέθοδος χρησιμοποιεί μη γραμμική συνάρτηση για τη "χρέωση" του συνολικού ανοίγματος στη στοίχιση. Η μη γραμμική μέθοδος συνήθως χρεώνει λίγο το άνοιγμα και το κλείσιμο του κενού, ενώ αυξάνει σταδιακά το κόστος διατήρησης του κενού. Αν και είναι γνωστή μέθοδος, συνήθως αποφεύγεται η χρήση της εκτός από εξαιρετικά ειδικές συνθήκες, κυρίως εξαιτίας της έλλειψης Βιολογικής επαλήθευσης της λογικής.

2.5 Μέθοδοι στοίχισης

Πριν την εισαγωγή των υπολογιστών στο χώρο της μοριακής βιολογίας, η στοίχιση αλληλουχιών γινόταν με το χέρι. Ο βιολόγος, κατέγραφε τις αλληλουχίες τη μία κάτω από την άλλη (όπως είδαμε και στο παράδειγμα της προηγούμενης ενότητας) έψαχνε να βρει με το μάτι διάφορα ταιριάσματα και στη συνέχεια μετρώντας το score για κάθε ταίριασμα κρατούσε το μεγαλύτερο. Όπως καταλαβαίνουμε, πρόκειται για μια εξαιρετικά επίπονη διαδικασία όταν αναφερόμαστε σε στοίχιση δύο αλληλουχιών μικρού μήκους,

και αδύνατη όταν αναφερόμαστε σε στοίχιση πολλών αλληλουχιών μεσαίου ή και μεγάλου μήκους.

Με την πάροδο του χρόνου εμφανίστηκαν διάφοροι μέθοδοι για την σύγκριση βιολογικών ακολουθιών.

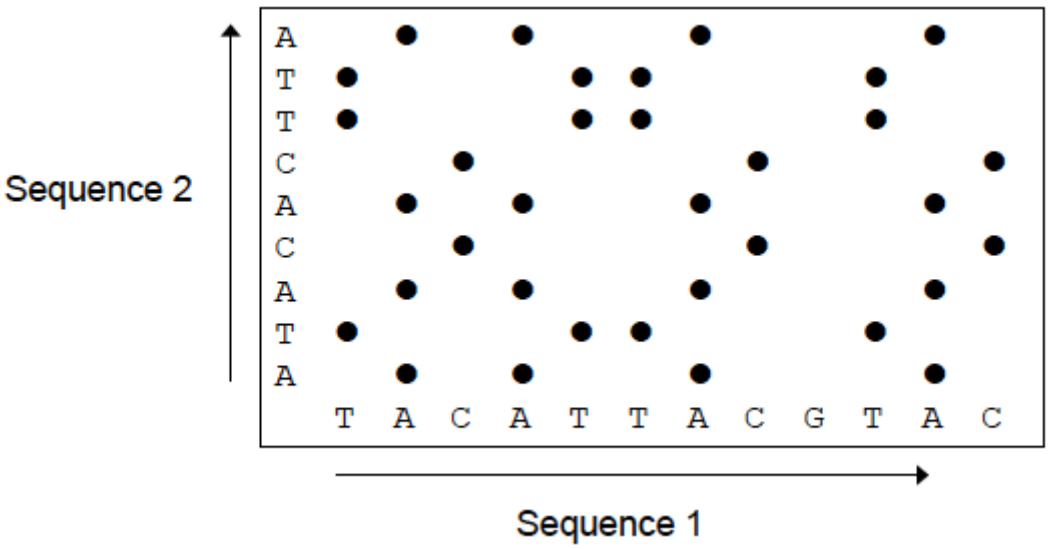
2.5.1. Dot plot

Η πιο απλή μέθοδος εύρεσης ομοιοτήτων μεταξύ δύο αλληλουχιών είναι η μέθοδος dot plot. Στο dot plot, οι ακολουθίες απεικονίζονται με τη μορφή ενός δισδιάστατου πίνακα όπου η μία ακολουθία τοποθετείται στον οριζόντιο άξονα ενώ η άλλη στον κάθετο. Στη συνέχεια αναζητούνται τα όμοια στοιχεία τους και σημαδεύονται τα ανάλογα κελιά. Έτσι, τα κοινά τμήματα των δύο ακολουθιών παρουσιάζονται ως διαγώνιες γραμμές στον πίνακα. Εύκολα αντιλαμβανόμαστε ότι στην περίπτωση ταύτισης των δύο αλληλουχιών ο πίνακας θα διατρέχεται από την κεντρική διαγώνιο στο μέσω του. Στις περιπτώσεις που έχουμε εισαγωγές ή διαγραφές καταλοίπων (εισαγωγή κενών), στον πίνακα εμφανίζονται ως διακοπές της διαγωνίου.

Ας δούμε ένα παράδειγμα. Έστω ότι έχουμε τις πρωτεϊνικές ακολουθίες

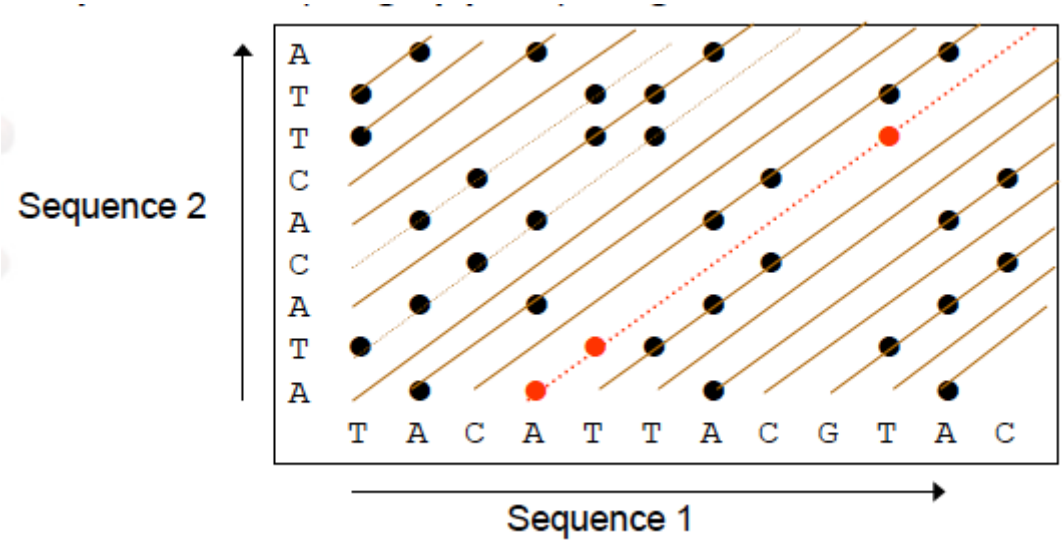
T A C A T T A C G T A C και A T T C A C A T A.

Κατασκευάζοντας τον πίνακα dot plot των δύο αυτών ακολουθιών έχουμε:



Εικόνα 12: Παράδειγμα Dot Plot 1

Οι κουκίδες μας δείχνου τα όμοια στοιχεία των δύο αλληλουχιών. Όπως προαναφέραμε, οι διαγώνιοι του πίνακα αντιστοιχούν σε μια πιθανή στοίχιση χωρίς κενά, όπως φαίνεται παρακάτω:

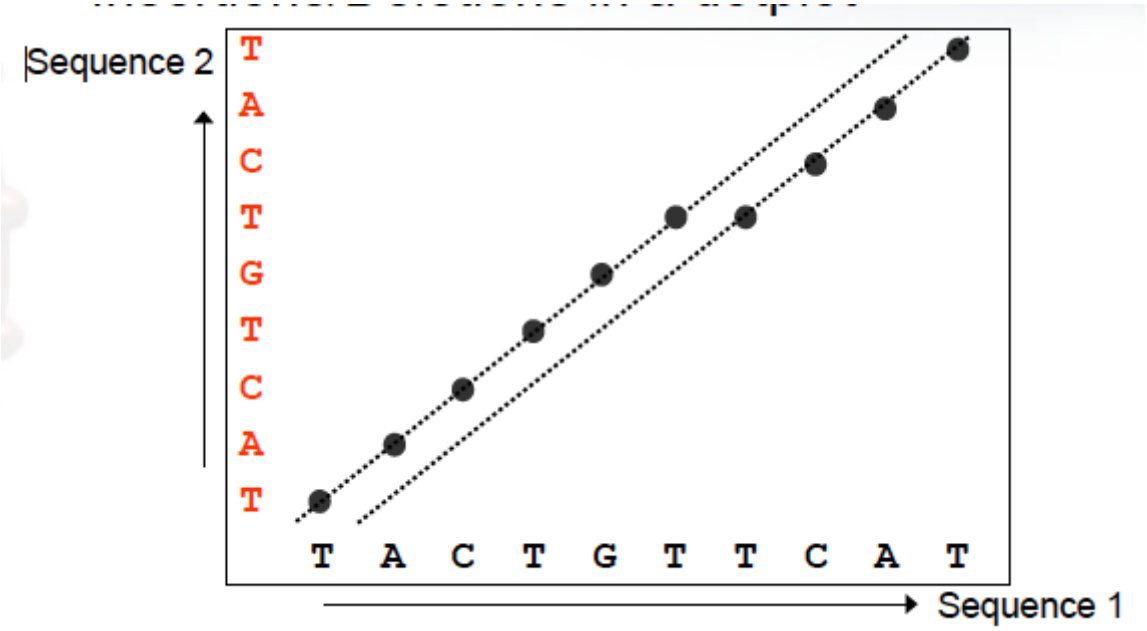


Εικόνα 13: Παράδειγμα Dot Plot 2

Όπου η πιθανή στοίχιση που προκύπτει θα είναι:

```
T A C A T T A C G T A C
    | | |     |
  A T A C A C T T A
```

Μπορούμε να επιχειρήσουμε να βρούμε και καλύτερη στοίχιση αν επιτρέψουμε την εισαγωγή κενών, όπως φαίνεται παρακάτω:



Εικόνα 14: Παράδειγμα Dot Plot 3

Όπου σε αυτή την περίπτωση η στοίχιση που επιτυγχάνεται ακολουθώντας τις τοπικές στοιχίσεις που προσδίδουν οι δύο διαγώνιες είναι:

```
T A C T G - T C A T
| | | | |   | | | |
T A C T G T T C A T
```

Η μέθοδος Dot plot, πέρα από το γεγονός ότι μας περιορίζει στην επιλογή σύγκρισης μόνο δύο αλληλουχιών και μάλιστα μικρού μήκους, δεν παρέχει και αποτελέσματα μεγάλης ακρίβειας καθώς δεν μας δίνει τη δυνατότητα εύρεσης της βέλτιστης στοίχισης. Για να βρεθεί η βέλτιστη στοίχιση θα πρέπει να βρεθεί το βέλτιστο μονοπάτι μέσα στον πίνακα, για το οποίο θα μιλήσουμε στην επόμενη μέθοδο.

2.5.2. Αλγόριθμοι στοίχισης (sequence alignment algorithms)

Προηγουμένως, ορίσαμε την ομοιότητα δύο αλληλουχιών ως την καλύτερη βαθμολόγηση που επιτυγχάνεται από τον συνδυασμό όλων των πιθανών στοιχίσεων μεταξύ τους.

Το να εξετάσουμε κάθε πιθανό συνδυασμό στοίχισης δύο ακολουθιών είναι επίπονο ακόμη και για ακολουθίες που αποτελούνται από ελάχιστα

κατάλοιπα. Για το λόγο αυτό, υπάρχουν ποιο αποτελεσματικές λύσεις που λύνουν το πρόβλημα, όπως η εφαρμογή αλγορίθμων.

2.5.2.1. Η μέθοδος του Δυναμικού Προγραμματισμού.

Ο δυναμικός προγραμματισμός είναι η μέθοδος η οποία επιτυγχάνει να βρει τη βέλτιστη στοίχιση από όλες τις πιθανές στοιχίσεις ανάμεσα σε δύο αλληλουχίες.

Ουσιαστικά, πρόκειται για μια παρόμοια μέθοδο με τη μέθοδο dot plot μιας και δημιουργεί και αυτός ένα παρόμοιο δισδιάστατο πλέγμα ευθυγράμμισης. Ωστόσο, βρίσκει την ευθυγράμμιση με έναν ποιο ποσοτικό τρόπο μετατρέποντας τον πίνακα dot plot σε πίνακα υπολογισμού καλύτερης βαθμολόγησης για να αθροίσει τις επιμέρους βαθμολογήσεις πού προκύπτουν από τα ταιριάσματα και μη-ταιριάσματα των καταλοίπων των αλληλουχιών. Η βέλτιστη στοίχιση επιτυγχάνεται με την αναζήτηση της καλύτερης βαθμολόγησης στον πίνακα από τις βαθμολογήσεις που προκύπτουν από όλες τις πιθανές στοιχίσεις των δύο αλληλουχιών.

Το ότι αναφερόμαστε σε στοίχιση δύο αλληλουχιών (pairwise alignment) για τον δυναμικό προγραμματισμό, δε σημαίνει ότι σε θεωρητικό επίπεδο δεν μπορεί να εφαρμοστεί και σε προβλήματα πολλαπλής στοίχισης (multiple alignment).

Το πρόβλημα όμως παρουσιάζεται στην πρακτική εφαρμογή του, γιατί όπως θα δούμε και στη συνέχεια, οι πίνακες που δημιουργούνται σε ένα τέτοιο πρόβλημα είναι πολλών διαστάσεων και η πολυπλοκότητα του αλγορίθμου μας το απαγορεύει.

Αλγόριθμος Needleman – Wunsch

Ο αλγόριθμος Needleman- Wunsch (Saul B. Needleman et al., 1970) είναι ένας ευρέως διαδεδομένος αλγόριθμος δυναμικού προγραμματισμού που χρησιμοποιείται κυρίως για την εύρεση της βέλτιστης καθολικής στοίχισης ανάμεσα σε δύο αλληλουχίες ίδιου μήκους ή σχεδόν ίδιου μήκους.

Ο τρόπος λειτουργίας του βασίζεται στο κεντρικό δόγμα του δυναμικού προγραμματισμού η εύρεση της βέλτιστης αντιστοίχισης βασίζεται στην χρήση προηγούμενων βέλτιστων αντιστοιχίσεων υποακολουθιών των υπό έρευνα ακολουθιών.

Ο αλγόριθμος βρίσκει τη βέλτιστη αντιστοίχιση των δύο ακολουθιών βασιζόμενος στη βέλτιστη βαθμολογία που προκύπτει από την έρευνα όλων των πιθανών αντιστοιχίσεων των βάσεων των δύο ακολουθιών. Επιτρέπει την προσθήκη κενών στο εσωτερικό αλλά και στα δύο άκρα και των δύο ακολουθιών, και στο τέλος παραθέτει τη βέλτιστη στοίχιση δύο ακολουθιών του ίδιου μήκους.

Ας δούμε τώρα τον τρόπο λειτουργίας του αλγορίθμου. Για την καλύτερη και ευκολότερη κατανόηση του τρόπου λειτουργίας του, θεωρούμε σκόπιμο να απλοποιήσουμε τις καταστάσεις θέτοντας μια σταθερή τιμή στην επιλογή ποινών και μια σταθερή τιμή για την αντικατάσταση βάσεων, χωρίς να λάβουμε υπόψη μας κάποια συγκεκριμένη μέθοδο επιβολής ποινών και κάποιον πίνακα αντικατάστασης.

Έστω ότι έχουμε την ακολουθία S η οποία έχει μήκος n βάσεων, και την ακολουθία T η οποία έχει μήκος m βάσεων, όπου n≠m. Θέτουμε αυθαίρετα την τιμή -1 για κάθε εισαγωγή κενού στις δύο ακολουθίες. Έστω i μια τυχαία βάση της ακολουθίας S, και j μια τυχαία βάση της ακολουθίας T.

Συμβολίζουμε με (S_i,T_j) την αντιστοίχιση του στοιχείου i της ακολουθίας S με το στοιχείο j της ακολουθίας T. Στην περίπτωση ταύτισης, δηλαδή i=j έχουμε $\sigma(S_i,T_j)=2$. Στην περίπτωση i≠ j (μη ταύτιση) έχουμε $\sigma(S_i,T_j)=-1$. Με $(S_i,-)$ συμβολίζουμε την αντιστοίχιση του στοιχείου i της ακολουθίας S με κενό της ακολουθίας T. Συνεπώς $\sigma(S_i,-)=-1$.

Αρχικά δημιουργείται ένας πίνακας (n+1)x(m+1), όπου η γραμμή 0 και η στήλη 0 αναπαριστούν το κόστος που θα είχαμε αν προσθέταμε διαδοχικά κενά και στις δύο ακολουθίες κατά την έναρξη του αλγορίθμου.

Η βαθμολογία που προσδίδεται σε κάθε κελί υπολογίζεται αναδρομικά με βάση τη μέγιστη τιμή ενός γειτονικού του κελιού.

Για την καλύτερη κατανόηση του τρόπου λειτουργίας του αλγορίθμου είναι προτιμότερο να δούμε ένα παράδειγμα.

Έστω S= A C C G G T A T και T= A C C T A T C, οι δύο προς στοίχιση ακολουθίες.

Αρχικά παρατηρούμε ότι n=8 και m=7, τα μήκη των ακολουθιών S και T είναι περίπου ίδια, οπότε μπορούμε να εφαρμόσουμε τον αλγόριθμο καθολικής στοίχισης.

Χρησιμοποιώντας τις αποτιμήσεις που θέσαμε παραπάνω προκύπτει ο παρακάτω στοιχειώδης πίνακας αντικατάστασης:

Πίνακας 4: Πίνακας Αντικατάστασης DNA

	A	**C**	**G**	**T**
A	2	-1	-1	-1
C	-1	2	-1	-1
G	-1	-1	2	-1
T	-1	-1	-1	2

Κατασκευάζουμε πίνακα V με διαστάσεις 9x8 ((8+1)x(7+1)) όπου στις γραμμές τοποθετούμε τις βάσεις της S ακολουθίας και στις στήλες τις βάσεις της ακολουθίας T.

Η πρώτη γραμμή και η πρώτη στήλη όπως προαναφέραμε συμπληρώνεται προσθέτοντας διαδοχικά κενά και στις δύο αλληλουχίες:

Πίνακας 5: Πίνακας Δ. Προγραμματισμού 1

		A	C	C	G	G	T	A	T	(S)
	0 →	-1 →	-2 →	-3 →	-4 →	-5 →	-6 →	-7 →	-8	
A	-1									
C	-2									
C	-3									
T	-4									
A	-5									
T	-6									
C	-7									
(T)										

Για να υπολογίσουμε τις τιμές των υπόλοιπων κελιών κατευθυνόμαστε από αριστερά προς τα δεξιά και υπολογίζουμε γραμμή- γραμμή.

Ο υπολογισμός της τιμής του κελιού V(i,j) βασίζεται στις τιμές των γειτονικών του κελιών που είναι τα V(i-1,j), V(i,j-1), V(i-1,j-1) και δίνεται από τη σχέση:

$$V(i,j)=\max\begin{cases}V(i-1,j-1)+\sigma(Si,Tj)\\V(i-1,j)+\sigma(Si,-)\\V(i,j-1)+\sigma(-,Tj)\end{cases}$$

Για να υπολογίσουμε την τιμή του κελιού V(1,1) χρησιμοποιώντας τον προηγούμενο τύπο έχουμε:

$$V(i,j)=\max\begin{cases}V(i-1,j-1)+\sigma(Si,Tj)\\V(i-1,j)+\sigma(Si,-)\\V(i,j-1)+\sigma(-,Tj)\end{cases}\Rightarrow V(1,1)=\max\begin{cases}V(0,0)+\sigma(S1,T1)\\V(0,1)+\sigma(S1,-)\\V(1,0)+\sigma(-,T1)\end{cases}=\max\begin{cases}0+2\\-1+(-1)=2\\-1+(-1)\end{cases}$$

Με βάση τη μέγιστη τιμή που πήραμε κινούμαστε διαγώνια όπου έχουμε και ταύτιση των δύο ακολουθιών.

Στην περίπτωση που θα έπρεπε να κινηθούμε κάθετα (το μέγιστο προερχόταν από το κελί V(1,0)) θεωρούμε πάντα, ακόμη και σε περίπτωση ταύτισης ότι εισάγουμε κενό στην ακολουθία S η οποία έχει επιλεχθεί ως η οριζόντια ακολουθία στον πίνακα. Το αντίστοιχο συμβαίνει και στην περίπτωση που το μέγιστο προέρχεται από το κελί V(0,1) μόνο που σε αυτή την περίπτωση κινούμαστε οριζόντια και το κενό τοποθετείται στην ακολουθία T.

Κάθε φορά που υπολογίζεται η τιμή ενός κελιού του πίνακα αποθηκεύεται και ο δείκτης ο οποίος μας δείχνει από ποιο στοιχείο μεταβήκαμε στο παρόν στοιχείο. Ο λόγος για τον οποίο γίνεται αυτό, είναι για να μπορέσουμε αναδρομικά μετά το πέρας της συμπλήρωσης του πίνακα να προσδιορίσουμε το βέλτιστο μονοπάτι.

Εάν μία τιμή ενός στοιχείου έχει προκύψει από τα δύο ή και τα τρία προηγούμενα στοιχεία του πίνακα, κρατούνται οι δείκτες και από τα τρία στοιχεία ούτος ώστε να κρατηθούν και τα τρία διαφορετικά μονοπάτια. Κάθε μονοπάτι απεικονίζει και μία διαφορετική αντιστοίχηση και η επιλογή του βέλτιστου μονοπατιού / αντιστοίχησης, γίνεται πλέον με γνώμονα τη μέγιστη βαθμολόγηση που αντιστοιχεί σε κάποιο από αυτά. Σε περίπτωση που τα μονοπάτια είναι ισότιμα, η επιλογή γίνεται αυθαίρετα.

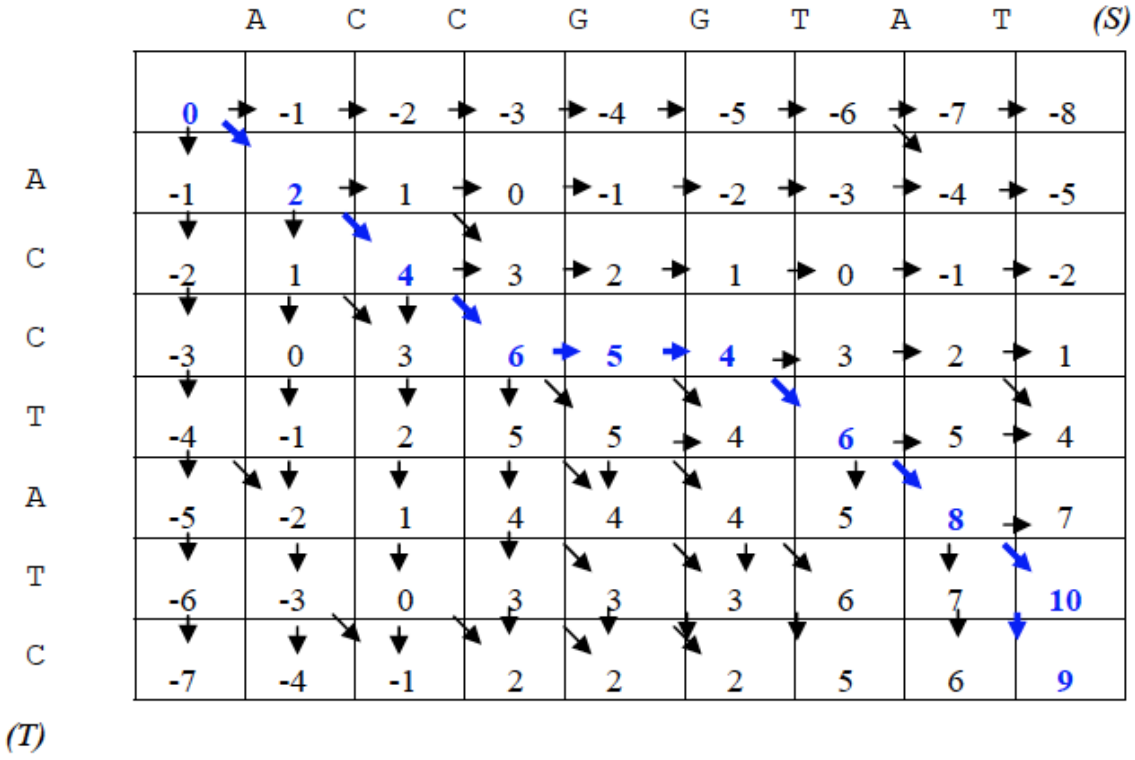
Ακολουθώντας όλα τα βήματα του αλγορίθμου που περιγράψαμε καταλήγουμε στον παρακάτω συμπληρωμένο πίνακα. Τα βελάκια μας δείχνουν το προηγούμενο κελί από το οποίο μεταβήκαμε.

Πίνακας 6: Πίνακας Δ. Προγραμματισμού 2

		A	C	C	G	G	T	A	T	(S)
	0	-1	-2	-3	-4	-5	-6	-7	-8	
A	-1	2	1	0	-1	-2	-3	-4	-5	
C	-2	1	4	3	2	1	0	-1	-2	
C	-3	0	3	6	5	4	3	2	1	
T	-4	-1	2	5	5	4	6	5	4	
A	-5	-2	1	4	4	4	5	8	7	
T	-6	-3	0	3	3	3	6	7	10	
C	-7	-4	-1	2	2	2	5	6	9	
(T)										

Στη συνέχεια, περνάμε στο δεύτερο μέρος του αλγορίθμου, όπου ξεκινώντας από το τελευταίο κελί κάτω- δεξιά ακολουθούμε αναδρομική πορεία υπολογίζουμε τη βέλτιστη διαδρομή όπως φαίνεται στον παρακάτω πίνακα:

Πίνακας 7: Πίνακας Δ. Προγραμματισμός- Μονοπάτι 1



Η διαδρομή που ακολουθήσαμε μας δείχνει και την τελική στοίχιση των ακολουθιών, η οποία είναι:

S: A C C G G T A T –
| | | | | |
T: A C C - - T A T C

Η παραπάνω στοίχιση η οποία είναι και η βέλτιστη μας δίνει συνολικό score που προκύπτει από τα επί μέρους αθροίσματα των έξι ταυτίσεων (2x6=12) και των τριών κενών (3x(-1)=-3).

Ακολουθεί η ολοκληρωμένη περιγραφή του αλγορίθμου.

Αλγόριθμος Needleman Wunsch.

Αλγόριθμος 1.

(1) S[0, 0] ← 0

(2) **for** j ← 1 **to** N

(3) S[0, j] ← S[0, j − 1] + s(−, bj)

(4) **for** i ← 1 **to** M

(5) S[i, 0] ← S[i − 1, 0] + s(ai ,−)

(6) **for** j ← 1 **to** N

(7) **for** j ← 1 **to** M

(8) Horizontal ← S[i, j − 1] + s(−, bj)

(9) Vertical ← S[i − 1, j] + s(ai ,−)

(10) Diagonal ← S[i − 1, j − 1] + s(ai , bj)

(11) S[i, j] ← max{Horizontal, Vertical, Diagonal}

Οι ποσότητες s(ai, −) και s(−, bj) εκφράζουν τις ποινές που αποδίδονται σε κάθε κενό. Οι ποινές των κενών υπολογίζονται με βάση τη μέθοδο διαχείρισης κενών που θα επιλέξουμε, με ποιο συνηθισμένη την affine. Συγκεκριμένα, αν στην προηγούμενη θέση υπάρχει κενό, τα s(ai, −) και s(−, bj) παίρνουν την τιμή gep (από τη γραμμική συνάρτηση που αναφερθήκαμε παραπάνω) ενώ σε αντίθετη περίπτωση παίρνουν την τιμή gop (βλέπε εξίσωση χ). Η ποσότητα s(ai, bj) λαμβάνεται από τον πίνακα αντικατάστασης PAM ή BLOSUM. Για την εύρεση της βέλτιστης στοίχισης ακολουθείται η αντίστοιχη αναδρομική διαδρομή που περιγράφεται στον παρακάτω αλγόριθμο.

Αλγόριθμος 2: Traceback.

(1) i ← M , j ← N , r ← 1

(2) while i != 0 and j != 0

(3) Horizontal ← S[i, j − 1] + s(−, bj)

(4) Vertical ← S[i − 1, j] + s(ai ,−)

(5) Diagonal ← S[i − 1, j − 1] + s(ai , bj)

(6) k ← max{Horizontal, Vertical, Diagonal}

(7) if k==Horizontal

(8) i ← i , j ← j − 1

(9) else if k==Vertical

(10) i ← i − 1 , j ← j

(11) else if k==Diagonal

(12) i ← i − 1 , j ← j − 1

(13) Traceback[r] ← [i, j]

(14) r ← r + 1

Ο αλγόριθμος Needleman Wunsch μας δίνει πάντα τη βέλτιστη στοίχιση αλλά καταναλώνει πολύ χρόνο επεξεργασίας και μνήμη. Αν λάβουμε υπόψη μας ότι χρειάζεται να αποθηκεύσει τα στοιχεία ενός πίνακα (n+1)x(m+1) και για κάθε στοιχείο να εκτελέσει τέσσερεις στοιχειώδεις πράξεις, τρία αθροίσματα και την εύρεση ενός μεγίστου παρατηρούμε ότι η πολυπλοκότητα του αλγορίθμου η οποία είναι O(nm) τόσο για τον απαιτούμενο χρόνο όσο και για την απαιτούμενη μνήμη, τον καθιστά εφαρμόσιμο μόνο σε αλληλουχίες μικρού μήκους. Δυστυχώς οι βιολογικές αλληλουχίες είναι συνήθως μεγάλου μήκους, με αποτέλεσμα αν και ο συγκεκριμένος αλγόριθμος αποδεικνύεται εφικτός και αποδίδει πάντα σωστά αποτελέσματα, εντούτοις είναι εξαιρετικά αργός.

Αλγόριθμος Smith- Waterman

Με τον αλγόριθμο Needleman – Wunsch καταφέραμε την ολική στοίχιση δύο αλληλουχιών. Πολλές φορές όμως, υπάρχουν βιολογικές ακολουθίες οι οποίες αν και είναι ομόλογες μεταξύ τους με το πέρασμα των χρόνων έχουν υποστεί μεγάλο αριθμό μεταλλάξεων, πράγμα το οποίο τις καθιστά εντελώς διαφορετικές μεταξύ τους στο σύνολο. Σε τέτοιες ακολουθίες

είναι χρήσιμο να προσπαθούμε να βρούμε ομοιότητες σε υποσύνολά τους και όχι σε όλο το μήκος τους. Ουσιαστικά αναφερόμαστε σε μια τοπική στοίχιση (local alignment) και όχι σε μια ολική (global alignment) όπως έχουμε δει μέχρι τώρα.

Δεν υπάρχουν ιδιαίτερες διαφοροποιήσεις μεταξύ τοπικής και ολικής στοίχισης. Κι εδώ γίνεται η χρήση των ίδιων πινάκων αντικατάσταση καθώς και των ίδιων μεθόδων υπολογισμού των κενών. Και στην τοπική στοίχιση ο δυναμικός προγραμματισμός μας εγγυάται την βέλτιστη λύση, χρησιμοποιείται όμως με ορισμένες παραλλαγές.

Ο αντιπροσωπευτικότερος αλγόριθμος δυναμικού προγραμματισμού για τοπική στοίχιση είναι ο αλγόριθμος Smith Waterman (Smith, 1981). Ο πίνακας αντιστοίχισης και σε αυτόν τον αλγόριθμο κατασκευάζεται με τον ίδιο τρόπο, μόνο που στην κλαδική συνάρτηση εύρεση του μεγίστου όρου εισάγεται και μια νέα επιλογή, το μηδέν όπως φαίνεται παρακάτω:

$$V(i,\ j) = \max \begin{cases} 0 \\ V(i-1,\ j-1) + \sigma(Si, Tj) \\ V(i-1,\ j) + \sigma(Si, -) \\ V(i,\ j-1) + \sigma(-, Tj) \end{cases}$$

Έτσι, παρατηρούμε ότι η κατώτερη τιμή που μπορεί να μπει σε ένα κελί είναι η τιμή μηδέν (κάτω φράγμα). Η εισαγωγή της κατώτερης τιμής έχει την έννοια της έναρξης μιας νέας στοίχισης. Η λογική του βασίζεται στο ότι σε περίπτωση που προκύψει μια αρνητική τιμή είναι προτιμότερο να ξεκινήσει μια νέα αντιστοίχιση.

Μετά το πέρας της συμπλήρωσης του πίνακα αντιστοίχισης, η διαδικασία εύρεσης της βέλτιστης διαδρομής λειτουργεί ως εξής:

Αρχικά βρίσκουμε το στοιχείο του πίνακα με τη μεγαλύτερη τιμή, το οποίο μπορεί να βρίσκεται σε οποιαδήποτε θέση του πίνακα. Στη συνέχεια ξεκινάμε την αναδρομική διαδικασία μέχρι να καταλήξουμε σε ένα μηδενικό στοιχείο. Η βαθμολογία της υποακολουθίας που βρέθηκε είναι τη τιμή του πρώτου στοιχείου.

Εφαρμόζοντας τον αλγόριθμο Smith Waterman στο παράδειγμα που χρησιμοποιήσαμε στον αλγόριθμο στον αλγόριθμο Needleman – Wunsch καταλήγουμε στον παρακάτω πίνακα αντιστοίχισης:

Πίνακας 8: Πίνακας Δ. Προγραμματισμός με S-W

		A	C	C	G	G	T	A	T	*(S)*
	0	0	0	0	0	0	0	0	0	
T	0	0	0	0	0	0	2	1	2	
T	0	0	0	0	**0**	0	2	1	3	
G	0	0	0	0	2	**2**	1	1	2	
T	0	0	0	0	1	1	**4**	3	3	
A	0	2	1	0	0	0	3	**6**	5	
T	0	1	1	0	0	0	2	5	**8**	
C	0	0	3	3	2	1	1	4	7	
(T)										

Ξεκινώντας από το στοιχείο του πίνακα με τη μεγαλύτερη τιμή το οποίο είναι το V(9,7) με τιμή 8 και ακολουθώντας τους δείκτες αναδρομικά καταλήγουμε στο στοιχείο V(5,3) με τιμή 0.

Η μέγιστη τοπική στοίχιση των αλληλουχιών S και T έχει score 8 και είναι:

S: G T A T

| | | |

T: G T A T

Σε πολλές περιπτώσεις, όπως και στο συγκεκριμένο παράδειγμα η υποακολουθία που δημιουργείται κατά την τοπική στοίχιση μπορεί να αποτελεί υποσύνολο του αποτελέσματος της εφαρμογής ολικής στοίχισης των δύο ακολουθιών. Θα πρέπει όμως να γνωρίζουμε ότι αυτό δεν συμβαίνει πάντα.

Ο τρόπος λειτουργίας του αλγορίθμου Smith Waterman είναι παραπλήσιος με τον τρόπο λειτουργίας του αλγορίθμου Needleman – Wunsch και κατά συνέπεια έχει την ίδια ακριβώς πολυπλοκότητα (O(nm)) και σε χρόνο αλλά και σε μνήμη.

2.5.2.2. Ευρεστικές Μέθοδοι (Heuristic Methods)

Στην προηγούμενη παράγραφο παρουσιάσαμε τους αλγορίθμους στοίχισης οι οποίοι βασίζονται στο Δυναμικό Προγραμματισμό. Είδαμε τα πλεονεκτήματα τους σε σχέση με την ποιότητα λύσης που μας παρέχουν, καθώς και τα μειονεκτήματα τους σε σχέση με το χρόνο και την κατανάλωση μνήμης που απαιτούν.

Πολλές φορές, το πρόβλημα στοίχισης αλληλουχιών (δύο ή περισσότερων) επεκτείνεται σε πρόβλημα αναζήτησης μιας ομόλογης αλληλουχίας σε μια βάση δεδομένων.

Αναζητώντας ομοιότητες και κοινά μοτίβα μιας υπό εξέτασης ακολουθίας σε μια μεγάλη βάση δεδομένων με μεθόδους δυναμικού προγραμματισμού όπως για παράδειγμα ο αλγόριθμος Smith- Waterman, παρόλο που είναι ακριβείς και αξιόπιστοι, αποδεικνύεται ότι είναι εξαιρετικά αργοί και μη πρακτικό όταν οι υπολογιστικοί πόροι είναι περιορισμένοι. Μια υπολογιστική εκτίμηση που πραγματοποιήθηκε σχεδόν μια δεκαετία πριν, έδειξε ότι η αναζήτηση σε μια βάση δεδομένων 300000 πρωτεϊνικών ακολουθιών χρησιμοποιώντας ένα ερώτημα μιας ακολουθίας 100 καταλοίπων χρειάστηκε 2 με 3 ώρες για να ολοκληρωθεί σε ένα υπολογιστικό σύστημα εκείνης της εποχής. Για την επιτάχυνση της όλης διαδικασίας δοκιμάστηκαν διάφορες προσεγγίσεις όπως η χρήση παράλληλων υπολογιστών στους οποίους κατανέμονται οι διάφορες διαδικασίες και εκτελούνται στον ίδιο χρόνο σε διαφορετικούς υπολογιστές. Η παραλληλοποίηση λύνει το πρόβλημα ως ένα βαθμό, αποτελεί όμως λύση με εξαιρετικό κόστος. Η εύρεση ταχύτερων διαδικασιών μέσα σε λογικά οικονομικά πλαίσια οδήγησε στη δημιουργία των ευρεστικών διαδικασιών. Με τον όρο ευρεστικές διαδικασίες ονομάζονται οι μέθοδοι

επίλυσης προβλημάτων οι οποίες βασίζονται στην εμπειρία και κινούνται προς μία λύση οδηγούμενες από δοκιμές και τυχόν λάθη τους. Με τη χρήση ευρεστικών διαδικασιών κερδίζουμε μεν χρόνο και καλύτερη διαχείριση μνήμης, θυσιάζουμε όμως την απόλυτη αξιοπιστία που μας παρέχουν οι αλγόριθμοι Δυναμικού Προγραμματισμού και αυτό γιατί δεν μπορούν να εγγυηθούν ότι θα καταλήξουν στην καλύτερη δυνατή αντιστοίχηση, καθώς τα αποτελέσματα τους είναι προσεγγιστικά. Η απόδοσή τους είναι καλύτερη γιατί εξετάζουν μόνο κομμάτια πιθανών αντιστοιχίσεων και όχι σε ολόκληρο το μήκος των ακολουθιών όπως οι αλγόριθμοι Δυναμικού Προγραμματισμού. Παρόλα αυτά, παρέχουν μια καλή εικόνα της ομοιότητας των δύο προς σύγκριση ακολουθιών.

Επί του παρόντος, υπάρχουν δύο ευρέος χρησιμοποιούμενες ευρεστικές μέθοδοι για την αναζήτηση αλληλουχιών σε βάσεις δεδομένων. Η μέθοδος BLAST και η FASTA.

(Εδώ θα πρέπει να σημειώσουμε ότι αν και οι ευρεστικές μέθοδοι δεν πληρούν τις απαιτούμενες προδιαγραφές για να χαρακτηριστούν αλγόριθμοι υπό την στενή έννοια των αλγορίθμων, εντούτοις καταχρηστικά όπως και πολλοί συγγραφείς, αναφερόμαστε σε αυτές τις μεθόδους ως αλγορίθμους).

Ο αλγόριθμος FASTA.

Ο αλγόριθμος FASTA (Lipman & Pearson, 1985) αποτέλεσε την πρώτη ευρεστική μέθοδο για την αναζήτηση ομοιοτήτων σε μια βιολογική βάση δεδομένων. Το όνομά του προέρχεται από τις λέξεις FAST ALL και ουσιαστικά αναζητά μέσα σε μια βάση δεδομένων παρόμοιες ακολουθίες με μια ζητούμενη ακολουθία. Η διαδικασία επιτυγχάνεται με την σύγκριση της ακολουθίας με κάθε μια ακολουθία που είναι αποθηκευμένη στη βάση δεδομένων ξεχωριστά και τέλος με την ανάκτηση των ακολουθιών που βρέθηκαν οι περισσότερες ομοιότητες. Η όλη διαδικασία αποτελείται από τέσσερα βήματα.

Το πρώτο βήμα για την ευθυγράμμιση FASTA είναι να εντοπιστούν τα ktups μεταξύ δύο ακολουθιών χρησιμοποιώντας τη στρατηγική του

κατακερματισμού. Τα ktups είναι λέξεις μεγέθους k, όπου για πρωτεϊνικές αλληλουχίες αποτελούνται συνήθως από δύο κατάλοιπα ενώ για αλληλουχίες DNA από έξι βάσεις. Η στρατηγική του κατακερματισμού (hashing strategy) λειτουργεί με την κατασκευή ενός πίνακα αναζήτησης που δείχνει τη θέση κάθε ktup για τις δύο ακολουθίες υπό εξέταση. Η διαφορά θέσης για κάθε λέξη μεταξύ των δύο ακολουθιών ή αλλιώς μετατόπιση, υπολογίζεται αφαιρώντας τη θέση της πρώτης σειράς από εκείνη της δεύτερης σειράς.

Στη συνέχεια, οι ktups που έχουν την ίδια μετατόπιση συνδέονται με στόχο να αποκαλύψουν μια συνεχόμενη πανομοιότυπη περιοχή που αντιστοιχεί σε ένα τμήμα της διαγωνίου ενός δισδιάστατου πίνακα, όπως στη μέθοδο dot plot.

Κατά το δεύτερο βήμα, περιορίζονται οι περιοχές ομοιότητας ανάμεσα στις δύο ακολουθίες. Οι περιοχές ομοιότητας που εντοπίστηκαν στο προηγούμενο βήμα (οι διαγώνιες του δισδιάστατου πίνακα) συνήθως είναι πολλές. Κρατούνται οι δέκα περιοχές με την υψηλότερη ομοιότητα (μεγαλύτερη πυκνότητα των διαγωνίων). Στη συνέχεια ακολουθεί η βαθμολόγηση αυτών των περιοχών με βάση τον πίνακα αντικατάστασης που έχουμε επιλέξει. Τα γειτονικά τμήματα με το μεγαλύτερο score που βρίσκονται κατά μήκος της ίδιας διαγωνίου επιλέγονται και δημιουργούν τη στοίχιση. Σε αυτό το βήμα επιτρέπεται η εισαγωγή κενών μεταξύ των διαγωνίων και η κατάλληλη επιλογή ποινών για τα κενά που θα εισαχθούν. Τέλος, υπολογίζεται εκ νέου η βαθμολογία της στοίχισης.

Στο βήμα 3 η στοίχιση που έχει βρεθεί θα βελτιστοποιηθεί περεταίρω με την εφαρμογή του αλγορίθμου Δυναμικού Προγραμματισμού Smith-Waterman ώστε να πάρουμε και την τελική στοίχιση.

Κατά το τελικό βήμα (βήμα 4) έχουμε την εκτέλεση μιας στατιστικής αξιολόγησης της τελικής στοίχισης με την εύρεση της τιμής E- value.

Για να αποτυπωθεί το μέγεθος αξιοπιστίας μιας αντιστοίχισης αλληλουχιών, χρησιμοποιούνται δύο στατιστικά μεγέθη, το p- value και το E-value.

Το p- value αντιπροσωπεύει το συσχετισμό του αποτελέσματος μιας αντιστοίχισης με την πιθανότητα να είναι τυχαίο. Ως στατιστικό μέγεθος, οι

τιμές του διακυμαίνονται μεταξύ 0 και 1. Όσο ποιο κοντά η τιμή του βρίσκεται στο 0, τόσο μεγαλύτερη αξιοπιστία υπάρχει στο αποτέλεσμα.

Το E- value περιγράφει τον αριθμό επιτυχιών (ομοιοτήτων) που αναμένεται να είναι τυχαία στην αναζήτηση μιας βάσης δεδομένων συγκεκριμένου μεγέθους (όταν το E-value πάρει την τιμή 1 για ένα ταίριασμα, αυτό μπορεί να ερμηνευτεί ότι στην τρέχουσα έρευνα, αναμένεται μόνο από τύχη να βρεθεί μια ομοιότητα με ίδιο αποτέλεσμα. Μια τιμή 0 δηλώνει ότι κανένα δεν αναμένεται να είναι τυχαίο.

Στο παράδειγμα που ακολουθεί μπορούμε να δούμε την λειτουργία του αλγορίθμου FASTA.

Οι δύο προς σύγκριση ακολουθίες αμινοξέων είναι:

1. Sequence 1: A M P S D G L
   Sequence 2: G P S D N A T
2. Με τη διαδικασία του κατακερματισμού καταλήγουμε στον παρακάτω πίνακα:

Πίνακας 9: Πίνακας κατακερματισμού

amino acid	sequence position		offset
	seq 1	seq 2	
A	1	6	-5
D	5	4	1
G	6	1	5
L	7	-	-
M	2	-	-
N	-	5	-
P	3	2	1
S	4	3	1
T	-	7	-

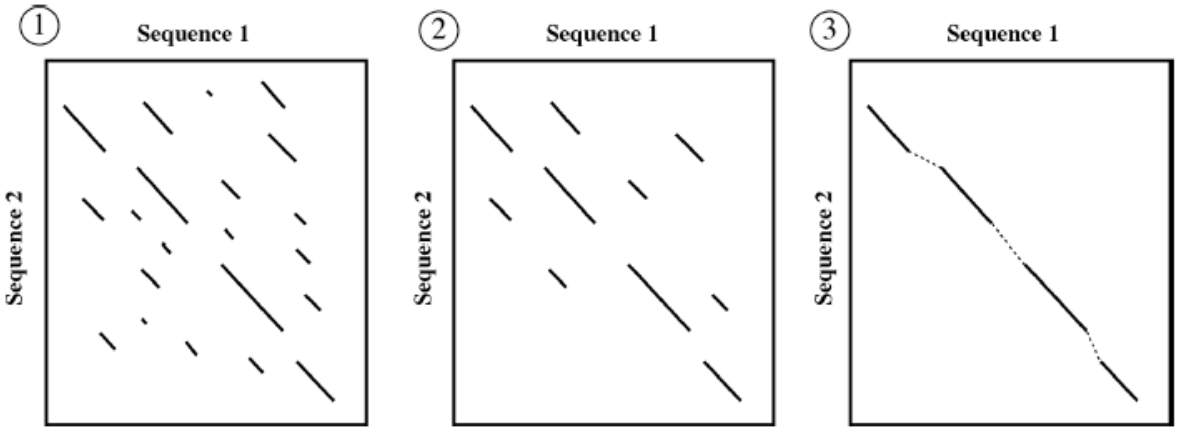
3. Υπογραμμισμένα με γκρι πλαίσιο είναι τα κατάλοιπα με την ίδια μετατόπιση (offset).
4. Εντοπίζουμε το ταίριασμα των λέξεων που αποτελούνται από τα τρία κατάλοιπα τα οποία βρίσκονται στις θέσεις 3, 4 και 5 στην πρώτη ακολουθία και 2, 3 και 4 στη δεύτερη.
5. Με βάση τα παραπάνω, καταλήγουμε στην εξής στοίχιση:

Sequence 1: A M P S D G L -

| | |

Sequence 2: - G P S D N A T

Με βάση τον δισδιάστατο πίνακα τύπου dot-plot μπορούμε να δούμε ξεκάθαρα τα βήματα που ακολουθεί ο FASTA για τη στοίχιση δύο ακολουθιών.



Εικόνα 15: Βήματα FASTA

Στο πρώτο βήμα παρατηρούμε τον εντοπισμό όλων των πιθανών στοιχίσεων ανάμεσα στις δύο ακολουθίες χωρίς την χρήση κενών.

Στο δεύτερο βήμα, βλέπουμε την επιλογή των δέκα καλύτερων στοιχίσεων με βάση την βαθμολόγηση που έχει υπολογιστεί με τη χρήση ενός πίνακα αντικατάστασης.

Τέλος, στο τρίτο βήμα καταλήγουμε στη βέλτιστη στοίχιση μετά την εισαγωγή κενών για τη συνένωση των διαγωνίων, και την εκ νέου βαθμολόγηση μέσω του αλγορίθμου Smith- Waterman.

Ο αλγόριθμος FASTA είναι υλοποιημένος σε ένα web-based πρόγραμμα με την ίδια ονομασία το οποίο προσφέρεται από το Ευρωπαϊκό Ινστιτούτο Βιοπληροφορικής (European Bioinformatics Institute www.ebi.ac.uk/) το οποίο μας επιτρέπει τη χρήση πρωτεϊνικών ή DNA ακολουθιών ως ερωτήματα για αναζήτηση σε πρωτεϊνικές βάσεις και βάσεις νουκλεοτιδίων. Το πρόγραμμα FASTA παρέχει και άλλες δυνατότητες μέσω υποπρογραμμάτων του όπως το FASTX το οποίο δημιουργεί ερωτήματα

μιας μεταφρασμένης DNA ακολουθίας σε πρωτεϊνική βάση και το TFASTX με το οποίο επιτυγχάνεται η αντίστροφη διαδικασία (δημιουργία ερωτήματος μεταφρασμένης πρωτεϊνικής αλληλουχίας σε νουκλεοτιδική βάση).

Ο αλγόριθμος BLAST (BASIC LOCAL ALIGNMENT SEARCH TOOL)

Ο αλγόριθμος BLAST(Altschul, Gish, Miller, Myers, Lipman - 1990) αποτελεί την ποιο διαδεδομένη τεχνική σύγκρισης βιολογικών ακολουθιών ακόμη και σήμερα. Η δημιουργία του βασίστηκε εν πολύς στον τρόπο λειτουργίας του FASTA. Παρόλα αυτά είναι ταχύτερος πράγμα το οποίο αποτελεί μεγάλο πλεονέκτημα αν σκεφτούμε την τεράστια συνεχή αύξηση των βιολογικών βάσεων δεδομένων. Παρά το γεγονός ότι αποτελεί μια ευριστική μέθοδο, καταφέρνει να συνδυάσει την ταχύτητα με την εξαγωγή αποτελεσμάτων σε ικανοποιητικά πλαίσια ακρίβειας. Ο σκοπός του αλγορίθμου είναι η εύρεση τμημάτων υψηλής βαθμολόγησης μεταξύ των σχετικών ακολουθιών. Η ύπαρξη μιας υψηλής βαθμολογίας που ξεπερνάει κάποιο όριο (κατώφλι) που θέτεται, μας δείχνει ομοιότητα η οποία δεν προέρχεται από τυχαίο γεγονός και μας βοηθάει στη διάκριση ακολουθιών της βάσης δεδομένων που σχετίζονται εξελικτικά μεταξύ τους από ακολουθίες που η ομοιότητά τους είναι τυχαία.

Ο αλγόριθμος BLAST ακολουθεί τα εξής βήματα:

- Κατά το πρώτο βήμα, δημιουργεί μια λίστα λέξεων από την ακολουθία που έχει τεθεί ως ερώτημα. Η κάθε λέξη συνήθως αποτελείτε από τρία κατάλοιπα στην περίπτωση πρωτεϊνικών ακολουθιών ή από έντεκα στην περίπτωση ακολουθιών DNA. Η λίστα περιέχει όλους τους πιθανούς συνδυασμούς λέξεων που μπορούν να προέρθουν από την ακολουθία του ερωτήματος. Το δεύτερο βήμα αφορά την αναζήτηση των συγκεκριμένων λέξεων μέσα στη βάση δεδομένων. γίνεται βαθμολόγηση των ακολουθιών που βρέθηκαν να ταιριάζουν με τις λέξεις με βάση του πίνακα αντικατάστασης που χρησιμοποιείται. Καταλογίζεται ως ταίριασμα της λέξης με λέξη μιας ακολουθίας της βάσεως δεδομένων, μόνο

στην περίπτωση που το score είναι πάνω από την τιμή ενός προκαθορισμένου κατωφλίου.

- Με την ολοκλήρωση της αντιστοίχισης στο σημείο των λέξεων ξεκινά η επέκτασή της και προς τις δύο κατευθύνσεις υπολογίζοντας το score στοίχισης με τον ίδιο πίνακα αντικατάστασης. Η επέκταση συνεχίζεται μέχρι το score της αντιστοίχισης λόγω των αναντιστοιχιών μεταξύ καταλοίπων πέσει κάτω από μια προκαθορισμένη τιμή (το όριο πτώσης είναι είκοσι δύο για τις πρωτεΐνες και είκοσι για το DNA). Τα τμήματα των δύο ακολουθιών που έχουν στοιχηθεί χωρίς κενά ονομάζονται τμήματα υψηλής βαθμολόγησης (HSP- High Scoring Segment Pair) και παρουσιάζονται ως τα τελικά αποτελέσματα.
- Μια πρόσφατη βελτίωση στην εφαρμογή του αλγορίθμου μας παρέχει τη δυνατότητα εισαγωγής κενών (επιτρέπει τις εισαγωγές και διαγραφές καταλοίπων στις ακολουθίες) χρησιμοποιώντας δυναμικό προγραμματισμό. Σε αυτή την περίπτωση η επέκταση συνεχίζεται εκατέρωθεν των δύο άκρων των ακολουθιών όσο το συνολικό score βρίσκεται πάνω από την τιμή του κατωφλίου. Παρόλα αυτά, επιτρέπεται μια πτώση του στην περίπτωση που στην περίπτωση που αυτή θα είναι προσωρινή.
- Τέλος, ο αλγόριθμος θα ξανατρέξει τις ολικές στοιχίσεις που ικανοποίησαν τα προηγούμενα κριτήρια ώστε εκτός από τις τελικές βαθμολογίες να αποδώσει και ακριβή στατιστικά αποτελέσματα.

Παράδειγμα:

Έστω ότι έχουμε την ακολουθία- ερώτημα: **M R D PYN K L I S**

Ο αλγόριθμος δημιουργεί λέξεις τριών καταλοίπων από το ερώτημα και τις αναζητά στη βάση δεδομένων.

Ας υποθέσουμε ότι για τη λέξη **P Y N**, η βάση μας έδωσε τα παρακάτω αποτελέσματα:

Ερώτημα	P Y N	P Y N	P Y N	P Y N	…

Βάση δεδομένων	P Y N	P F N	P F Q	P F E	…

Χρησιμοποιώντας έναν πίνακα αντικατάστασης (για παράδειγμα τον BLOSUM 62) παίρνουμε τις παρακάτω τιμές:

Ερώτημα	P Y N	P Y N	P Y N	P Y N	…
Βάση δεδομένων	P Y N	P F N	P F Q	P F E	…
Βαθμολόγηση	20	16	10	10	

Βρίσκουμε την ακολουθία της βάσης που μας παρέχει την καλύτερη στοίχιση της λέξης και επεκτείνουμε τη στοίχιση και από τις δύο πλευρές.

Ερώτημα: **M R D PYN K L I S**
Βάση: **M H E PYN D V P W**

Η επέκταση συνεχίζεται μέχρι να πέσει η βαθμολόγηση κάτω από την τιμή του κατωφλίου (το κατώφλι για τις πρωτεϊνικές αλληλουχίες είναι 22).

Ερώτημα	M	R	D	PYN	K	L	I	S
Βάση	M	H	E	PYN	D	V	P	W
βαθμός	5	0	2	20	-1	1	-3	-3

Τέλος, θα πρέπει να αναφέρουμε ότι όπως ο FASTA έτσι ο BLAST είναι υλοποιημένος σε ένα web-based πρόγραμμα με την ίδια ονομασία το οποίο βρίσκεται υλοποιημένο στο Εθνικό κέντρο Βιολογικών Πληροφοριών (NCBI- National Centre for Biological Information www.ncbi.nlm.nih.gov/BLAST) το οποίο μας επιτρέπει τη χρήση πρωτεϊνικών ή DNA ακολουθιών ως ερωτήματα για αναζήτηση σε πρωτεϊνικές βάσεις και βάσεις νουκλεοτιδίων. Υπάρχει μια ολόκληρη οικογένεια υποπρογραμμάτων όπως τα BLASTN, BLASTP, BLASTX TBLASTN, TBLASTX που μας παρέχουν πολλές δυνατότητες. Ενδεικτικά αναφέρουμε ότι με το BLASTN

μπορούμε να κάνουμε αναζήτηση σειράς αποτελούμενης από νουκλεοτίδια σε μια νουκλεοτιδική βάση, ενώ με το BLASTP δημιουργούμε πρωτεϊνικής δομής ερωτήματα για αναζήτηση στοίχισης σε πρωτεϊνικές βάσεις δεδομένων.

Σύγκριση FASTA- BLAST:

Παρόλο που και οι δύο ευρετικές μέθοδοι που παρουσιάσαμε συμπεριφέρονται εξίσου καλά στην αναζήτηση ακολουθίας σε βάσεις δεδομένων, ωστόσο παρουσιάζουν κάποιες διαφορές που αξίζει να σημειώσουμε. Η κύρια διαφορά τους εντοπίζεται στο στάδιο στοίχιση των λέξεων. Ο BLAST χρησιμοποιεί πίνακες αντικατάστασης για να βρει τις λέξεις που ταιριάζουν, ενώ ο FASTA χρησιμοποιεί τη διαδικασία του κατακερματισμού για την ταυτοποίηση τους. Χρησιμοποιεί μικρότερες λέξεις (στοιχίζει μικρότερα τμήματα) με αποτέλεσμα να μας παρέχει πιο ευαίσθητα αποτελέσματα σε σύγκριση με τον BLAST, με ένα καλύτερο ποσοστό κάλυψης για τις ομόλογες ακολουθίες. Ωστόσο, ο BLAST είναι αρκετά ταχύτερος και πέρα από την εύρεση της ακολουθίας με την καλύτερη βαθμολογία που επιστρέφει ο FASTA μας παρέχει τη δυνατότητα επιστροφής μιας λίστας ακολουθιών με τα υψηλότερα score στοίχισης με βάση το ερώτημα- ακολουθία που θέσαμε.

Είναι ξεκάθαρο ότι ο BLAST και ο FASTA κάνουν κάποιες θυσίες όσον αφορά την ευαισθησία (ακρίβεια) των αποτελεσμάτων σε σχέση με την ταχύτητα. Παρόλο αυτά, η βέλτιστη στοίχιση δεν είναι απαραίτητα και το βέλτιστο αποτέλεσμα από βιολογικής πλευράς, έτσι η θυσία της ακρίβειας με αντάλλαγμα την ταχύτητα μπορεί να μην είναι και τόσο επιβλαβής όσο φαίνεται.

ΚΕΦΑΛΑΙΟ 3: Γραμμικός και Ακέραιος Προγραμματισμός

3.1 Επιχειρησιακή Έρευνα

Με τον όρο επιχειρησιακή έρευνα (Operations Research ή αλλιώς Operational Research) αναφερόμαστε στον επιστημονικό κλάδο που ασχολείται με τη δημιουργία και την προσαρμογή μαθηματικών μοντέλων πάνω σε πολύπλοκα προβλήματα που ανακύπτουν στη διεύθυνση και διοίκηση μεγάλων συστημάτων που αποτελούνται από ανθρώπους, μηχανές ή υλικά. Κύριος σκοπός της είναι η βελτιστοποίηση των καταστάσεων κατά τη λήψη αποφάσεων.

Η επιχειρησιακή έρευνα εμφανίστηκε στις αρχές του δευτέρου παγκοσμίου πολέμου στη Μ. Βρετανία, όπου προσπάθησε να δώσει λύση σε καθαρά λειτουργικά προβλήματα όπως την βέλτιστη τοποθέτηση ραντάρ για την αποτελεσματικότερη αναχαίτιση των δυνάμεων του άξονα, την τοποθέτηση αποθηκών για τον καλύτερο ανεφοδιασμό των στρατευμάτων, αλλά και σε ποιο μακάβρια προβλήματα όπως την εύρεση αποδοτικότερων τρόπων εξολόθρευσης ανθρώπων. Αργότερα, εν καιρώ ειρήνης τα αποτελέσματα του νέου αυτού κλάδου προσέλκυσαν το ενδιαφέρον της βιομηχανίας.

Η εξέλιξη της επιχειρησιακής έρευνας είναι αλληλένδετη με την ανάπτυξη των Η/Υ. η είσοδος των Η/Υ έδωσε μια νέα ώθηση στην προσπάθεια εύρεσης τρόπων για την επίλυση προβλημάτων μεγάλων διαστάσεων με πληθώρα δεδομένων όπου είναι αδύνατη η εκτέλεση των απαιτούμενων υπολογισμών με το χέρι.

Λόγω της ευρείας γκάμας προβλημάτων που αντιμετωπίζει αποτελεί κοινό πεδίο ερευνών για πολλούς διαφορετικούς επιστημονικούς κλάδους όπως τη φυσική, το μαθηματικό, την επιστήμη των υπολογιστών καθώς και τη βιολογία και τη χημεία.

Οι κατηγορίες μεθόδων της Επιχειρησιακής Έρευνας είναι οι εξής:

- Μαθηματικός Προγραμματισμός
- Δέντρα αποφάσεων (Decision trees)
- Πολυκριτηριακή Ανάλυση (Multiple Criteria Decision Analysis)
- Ανάλυση δικτύων (Network flows, PERT, CPM)
- Διαχείριση αποθεμάτων (Inventory control, EOQ)
- Γραμμές αναμονής (Queuing theory)
- Στοχαστικές Διεργασίες (Stochastic Processes)
- Θεωρία Παιγνίων (Game theory)
- Προσομοίωση (simulation)

Τα βήματα αντιμετώπισης ενός προβλήματος που ακολουθεί η Επιχειρησιακή Έρευνα είναι:

- Αναγνώριση του προβλήματος
- Κατασκευή του μαθηματικού μοντέλου
- Επίλυση του μοντέλου
- Έλεγχος του μοντέλου και της λύσης του
- Εφαρμογή της τελικής λύσης

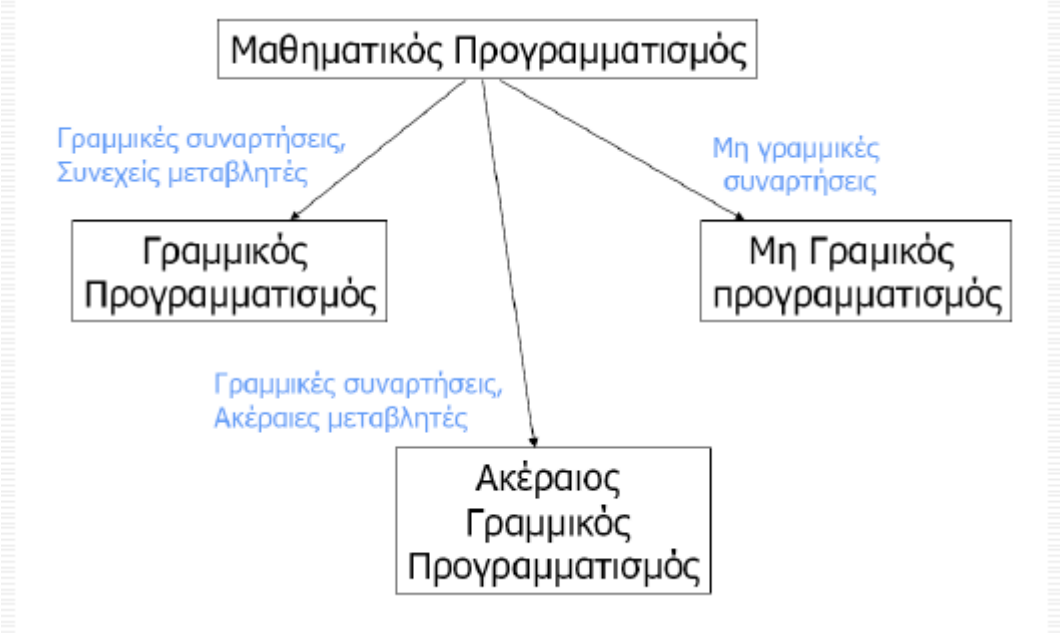
Η δημιουργία ενός αξιόπιστου μαθηματικού μοντέλου για την επίλυση του προβλήματος αποτελεί το βασικότερο κομμάτι της Επιχειρησιακής Έρευνας και είναι μια εξαιρετικά δύσκολη και επίπονη διαδικασία.

Το μαθηματικό μοντέλο που καλείται να αντιμετωπίσει το πρόβλημα περιλαμβάνει τις μεταβλητές του προβλήματος οι οποίες θα μας δώσουν και τη λύση του, τις παραμέτρους που πρέπει να λάβουμε υπόψη μας, τους περιορισμούς που τίθενται από το ίδιο το πρόβλημα και την αντικειμενική συνάρτηση η οποία αποτελεί και τον στόχο του μοντέλου (Χ. Ι. Σχοινάς, 2007).

3.2. Μαθηματικός Προγραμματισμός

Ο μαθηματικός Προγραμματισμός αποτελεί μια κατηγορία της Επιχειρησιακής Έρευνας με την οποία θα ασχοληθούμε παρακάτω.

Χωρίζεται σε δύο κύριες κατηγορίες. Τον Γραμμικό Προγραμματισμό και τον Μη- Γραμμικό Προγραμματισμό. Όπως και στο Δυναμικό Προγραμματισμό, η έννοια του Προγραμματισμού έχει να κάνει με την οργάνωση- σχεδίαση και όχι με τον προγραμματισμό σε Η/Υ (Β. Κώστογλου, 2003).



Εικόνα 16: Μαθηματικός Προγγραμματισμός

Στην εικόνα βλέπουμε τους επιμέρους κλάδους του Μαθηματικού προγραμματισμού.

Στην παρούσα εργασία θα ασχοληθούμε με το Γραμμικό Προγραμματισμό και τον Ακέραιο Γραμμικό Προγραμματισμό.

3.3. Γραμμικός Προγραμματισμός (Linear Programming)

Ο γραμμικός προγραμματισμός αποτελεί ένα μοντέλο κατά το οποίο επιτυγχάνεται η μεγιστοποίηση ή η ελαχιστοποίηση μιας γραμμικής συνάρτησης κάτω από γραμμικούς περιορισμούς. Θεωρείται μια από τις μεγαλύτερες μαθηματικές ανακαλύψεις του εικοστού αιώνα και στις μέρες μας αποτελεί ένα μοντέλο ευρείας χρήσης για καθημερινά ζητήματα καθώς και

στην επίλυση πολύπλοκων μαθηματικών προβλημάτων. Αλγόριθμοι γραμμικού προγραμματισμού όπως ο ελλειψοειδής αλγόριθμος (ο πρώτος αλγόριθμος γραμμικού προγραμματισμού με πολυωνυμικό χρόνο) ή ο αλγόριθμος εσωτερικών σημείων χρησιμοποιούνται με επιτυχία για την αποδοτική επίλυση πολλών συνδυαστικών προβλημάτων όπως ο υπολογισμός βέλτιστων ροών σε ένα δίκτυο, ο χρωματισμός ενός τέλειου γραφήματος ή η εύρεση του μέγιστου ταιριάσματος σε ένα γράφημα.

Εκτός από τα μαθηματικά προβλήματα που επιλύει, τα άμεσα καθημερινά προβλήματα στα οποία εφαρμόζεται έχουν να κάνουν με την επίλυση προβλημάτων κατανομής περιορισμένων μέσων ή πόρων σε εναλλακτικές και ανταγωνιστικές μεταξύ τους δραστηριότητες κατά τον καλύτερο δυνατό τρόπο. Κλασικά παραδείγματα αποτελούν τα προβλήματα προγραμματισμού των πληρωμάτων μιας αεροπορικής εταιρίας, ο υπολογισμός του συνδυασμού πρώτων υλών σε ένα εργοστάσιο που μεγιστοποιεί το κέρδος του τελικού προϊόντος, ή ο υπολογισμός των ροών αυτοκινήτων σε ένα οδικό δίκτυο, ή του φόρτου πληροφοριών σε ένα δίκτυο επικοινωνίας.

Ο αλγόριθμος Simplex (Simplex Algorithm, G. B. Dantzig 1947), είναι ο πρώτος αλγόριθμος Γραμμικού Προγραμματισμού και προτάθηκε τη δεκαετία του ’40 από τον G. B. Dantzig. Παρόλο που διαθέτει εκθετικό χρόνο εκτέλεσης στη χειρότερη περίπτωσή του, στην πράξη λειτουργεί πολύ αποδοτικά. Η καλή του απόδοση του είναι αυτό που ακόμη και σήμερα, μετά από μισό αιώνα και την δημιουργία νέων αλγορίθμων Γραμμικού Προγραμματισμού τον καθιστά εξαιρετικά χρήσιμο (Παπαρρίζος Κ, 2009).

3.3.1. Προσομοίωση προβλημάτων σε προβλήματα Γραμμικού Προγραμματισμού.

Όπως προαναφέραμε, το μεγαλύτερο και δυσκολότερο μέρος ενός προβλήματος που εφαρμόζεται η μέθοδος της Επιχειρησιακής Έρευνας είναι η δημιουργία ενός μαθηματικού μοντέλου που να ικανοποιεί πλήρως όλες τις απαιτήσεις του. Με τη δημιουργία ενός κατάλληλου μοντέλου, ο αναλυτής έχει στη διάθεσή του ένα σύνολο εργαλείων όχι μόνο για να βρει την καλύτερη λύση του προβλήματος αλλά και για να προχωρήσει σε μια ανάλυση

υποθέσεων για τις διάφορες παραμέτρους του. Η λύση, ανεξάρτητα από το πόσο λεπτομερής ή εξεζητημένη είναι έχει απλά έναν υποστηρικτικό ρόλο.

Η μαθηματική απεικόνιση των στοιχείων ενός προβλήματος περιλαμβάνει:

- Μεταβλητές

Αποτελούν τα δομικά στοιχεία ενός προβλήματος, τα οποία μπορεί να επηρεάσει ο αναλυτής. Αναφέρονται αλλιώς και ως μεταβλητές ελέγχου ή και μεταβλητές απόφασης.

Συμβολίζονται συνήθως με το γράμμα “x”. Ένα πρόβλημα μπορεί να διαθέτει για την επίλυσή του πολύ μεγάλο πλήθος μεταβλητών.

x_1, x_1, x_3,…x_n , όπου n το πλήθος των δραστηριοτήτων που λαμβάνουν χώρα στο πρόβλημα.

- Περιορισμοί

Οι περιορισμοί εκφράζουν τους περιορισμούς του περιβάλλοντος στο οποίο αναπτύσσεται η δραστηριότητα. Οι μεταβλητές απόφασης μπορούν να λάβουν οποιοδήποτε συνδυασμό τιμών, για να αποτελέσει όμως ένας εξ’ αυτών εφικτή λύση του προβλήματος θα πρέπει να ικανοποιεί όλους τους περιορισμού που έχουν τεθεί. Εκφράζονται συνήθως ως σύστημα ανισοτήτων και το δεξιό σκέλος τους συμβολίζεται συνήθως με το γράμμα b.

b_1, b_2, b_3, …, b_m , όπου m το πλήθος των ανισοτήτων- ισοτήτων των περιορισμών του προβλήματος.

- Τεχνολογικοί Συντελεστές

Είναι παράμετροι του τύπου a_{ij} όπου καθορίζουν τη σχέση κάθε μεταβλητής απόφασης i με τον περιορισμό j.

- Αντικειμενική Συνάρτηση $f(x_i)$

Η αντικειμενική συνάρτηση είναι η γραμμική συνάρτηση που θα μας οδηγήσει στο στόχο του προβλήματός μας. Με βάση την αντικειμενική συνάρτηση θα βρούμε εκείνες τις τιμές των μεταβλητών απόφασης οι οποίες θα βελτιστοποιήσουν τα κριτήρια απόδοσης που ορίζουμε στο μαθηματικό μας μοντέλο.

Ο στόχος μπορεί να είναι η μεγιστοποίηση (max(f)) ή η ελαχιστοποίησή της (min(f)).

Λόγω της ταυτότητας min(f(x))= -max(-f(x)) κάθε πρόβλημα ελαχιστοποίησης μπορεί να μετατραπεί σε πρόβλημα μεγιστοποίησης και αντίστροφα.

Έτσι το μαθηματικό μοντέλο κατασκευάζεται με αντικειμενική συνάρτηση:

$$f(x) = \sum_{j=1}^{n} c_j x_j = c_1 x_1 + c_2 x_2 + ... + c_n x_n ,$$

όπου c οι συντελεστές της γραμμικής συνάρτησης ή αλλιώς αντικειμενικοί συντελεστές. Σε προβλήματα μεγιστοποίησης χαρακτηρίζονται ως συντελεστές κέρδους, ενώ σε προβλήματα ελαχιστοποίησης ως συντελεστές κόστους.

Ψάχνουμε το min(f(x)) ή το max(f(x)) με περιορισμούς:

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i$$, όπου i= 1,2,3…m.

Συνεπώς η μορφή του γενικού προβλήματος του Γραμμικού Προγραμματισμού είναι:

$$z = \{\min, \max\}(c_1 x_1 + c_2 x_2 + ... + c_n x_n)$$

$$a_{11} x_1 + a_{12} x_2 + ... + a_{1n} x_n \le, =, \ge b_1$$
$$a_{21} x_1 + a_{22} x_2 + ... + a_{2n} x_n \le, =, \ge b_2$$
$$.....$$
$$a_{m1} x_1 + a_{m2} x_2 + ... + a_{mn} x_n \le, =, \ge b_m$$
$$x_1, x_2, ..., x_n \ge 0$$

Παράδειγμα 1.1

Το πρόβλημα της δίαιτας αποτελεί ένα από τα πρώτα προβλήματα στα οποία εφαρμόστηκε η μέθοδος του Γραμμικού Προγραμματισμού. Μέχρι την ανακάλυψη του αλγόριθμου Simplex εφαρμόστηκαν ευρετικές διαδικασίες για την επίλυσή του. Οι ευρετικές διαδικασίες εφαρμόστηκαν σε μια μορφή με 77 μεταβλητές και έδωσαν μια προσεγγιστική λύση, στην οποία η τιμή της αντικειμενική συνάρτησης ήταν 39,39. Ο αλγόριθμος Simplex ήρθε να επιλύσει το πρόβλημα βελτιστοποιώντας την αντικειμενική τιμή σε 39,67. Στη συνέχεια θα δούμε μια απλοϊκή μορφή του προβλήματος για να καταλάβουμε

τον τρόπο με τον οποίο μπορεί ένα πρόβλημα εκφρασμένο σε φυσική γλώσσα να παρασταθεί σε μαθηματικό μοντέλο.

Ας υποθέσουμε ότι μπορούμε να λαμβάνουμε καθημερινά έναν ορισμένο αριθμό θερμίδων. Επίσης, καθημερινά απαιτείται να λαμβάνουμε μια ελάχιστη ποσότητα πρωτεϊνών και ασβεστίου καθώς και να δαπανούμε το ελάχιστο ποσό. Ο παρακάτω πίνακας απεικονίζει τις πρωτεΐνες και το ασβέστιο που μας παρέχει μια τυπική δοσολογία κάθε είδους τροφής, τις θερμίδες που περιέχουν καθώς και το οικονομικό κόστος τους. Η τελευταία γραμμή του πίνακα μας δίνει τις καθημερινές μας απαιτήσεις σε πρωτεΐνες, ασβέστιο και θερμίδες.

Σκοπός μας είναι, με βάση τις καθημερινές μας απαιτήσεις να βρούμε τις δόσεις που μας εξασφαλίζουν το ελάχιστο κόστος.

Είδος τροφής	Δοσολ.	θερμ.(Kcal)	Πρωτ.(gr)	Ασβ.(mg)	Τιμή(ευρώ)
(1) Δημητριακά	28 γρ.	110	4	2	0.3
(2) Κοτόπουλο	100 γρ.	205	32	12	2.4
(3) Αβγά	2	160	13	54	1.3
(4) Γάλα	237κ.ε.	160	8	285	0.9
(5) Γλυκό	170 γρ.	420	4	22	2.0
(6) Χοιρινό	260 γρ.	260	14	80	1.9
Απαιτήσεις		2000	55	800	

Έστω x_i με $i \in \{1,2,3,4,5,6\}$, οι δόσεις από κάθε τροφική ομάδα που ψάχνουμε.

Με βάση τις καθημερινές απαιτήσεις σε θερμίδες που πρέπει κατά ελάχιστο να είναι 2000 Kcal, εύκολα καταλήγουμε στον παρακάτω περιορισμό:

$$110x_1 + 205x_2 + 160x_3 + 160x_4 + 420x_5 + 260x_6 \geq 2000$$

Με τον ίδιο ακριβώς τρόπο μπορούμε να δημιουργήσουμε και τις ανισότητες των άλλων δύο περιορισμών καθημερινής λήψης πρωτεϊνών και ασβεστίου αντίστοιχα:

$$4x_1 + 32x_2 + 13x_3 + 8x_4 + 4x_5 + 14x_6 \geq 55$$

$$2x_1 + 12x_2 + 54x_3 + 285x_4 + 22x_5 + 80x_6 \geq 800$$

Η αντικειμενική συνάρτηση που προσδιορίζεται από την ελαχιστοποίηση του κόστους θα είναι:

$$\min(0,3x_1 + 2,4x_2 + 1,3x_3 + 0,9x_4 + 2x_5 + 1,9x_6)$$

Το πρόβλημά μας παρέχει ακόμη έναν περιορισμό, ο οποίος δεν είναι άμεσα εμφανής.

Λόγω της φύσης των δεδομένων, δεν μπορούμε να έχουμε αρνητικές τιμές (δεν είναι δυνατόν να καταναλώνουμε αρνητικές ποσότητες τροφών).

Επομένως το συνολικό μοντέλο του προβλήματος είναι:

$$Αντικειμενική Συνάρτηση:$$
$$\min(0,3x_1 + 2,4x_2 + 1,3x_3 + 0,9x_4 + 2x_5 + 1,9x_6)$$

$$Περιορισμοί:$$
$$110x_1 + 205x_2 + 160x_3 + 160x_4 + 420x_5 + 260x_6 \geq 2000$$

$$4x_1 + 32x_2 + 13x_3 + 8x_4 + 4x_5 + 14x_6 \geq 55$$

$$2x_1 + 12x_2 + 54x_3 + 285x_4 + 22x_5 + 80x_6 \geq 800$$

$$x1, x2, x3, x4, x5, x6 \geq 0$$

Η λύση ενός προβλήματος Γραμμικού Προγραμματισμού ονομάζεται κάθε σύνολο x_j , j=1,2,3,…n το οποίο ικανοποιεί τους περιορισμούς τους προβλήματος.

Εφικτή ή δυνατή λύση είναι κάθε λύση που ικανοποιεί τους περιορισμούς μη αρνητικότητας ($x_j \geq 0$).

Βέλτιστη λύση είναι κάθε μια εφικτή λύση η οποία βελτιστοποιεί την αντικειμενική συνάρτηση. Σε ένα πρόβλημα Γραμμικού Προγραμματισμού συνήθως υπάρχουν άπειρες λύσεις και αυτό που αναζητούμε είναι η βέλτιστη δυνατή.

3.3.2. Γραφική επίλυση προβλημάτων Γραμμικού Προγραμματισμού.

Προβλήματα σχετικά απλά με δύο ή και τρείς μεταβλητές μπορούν να λυθούν γραφικά.

Τα προβλήματα με δύο μεταβλητές μπορούν να αναπαρασταθούν στο επίπεδο, ενώ τα προβλήματα με τρείς μπορούν να αναπαρασταθούν στο χώρο.

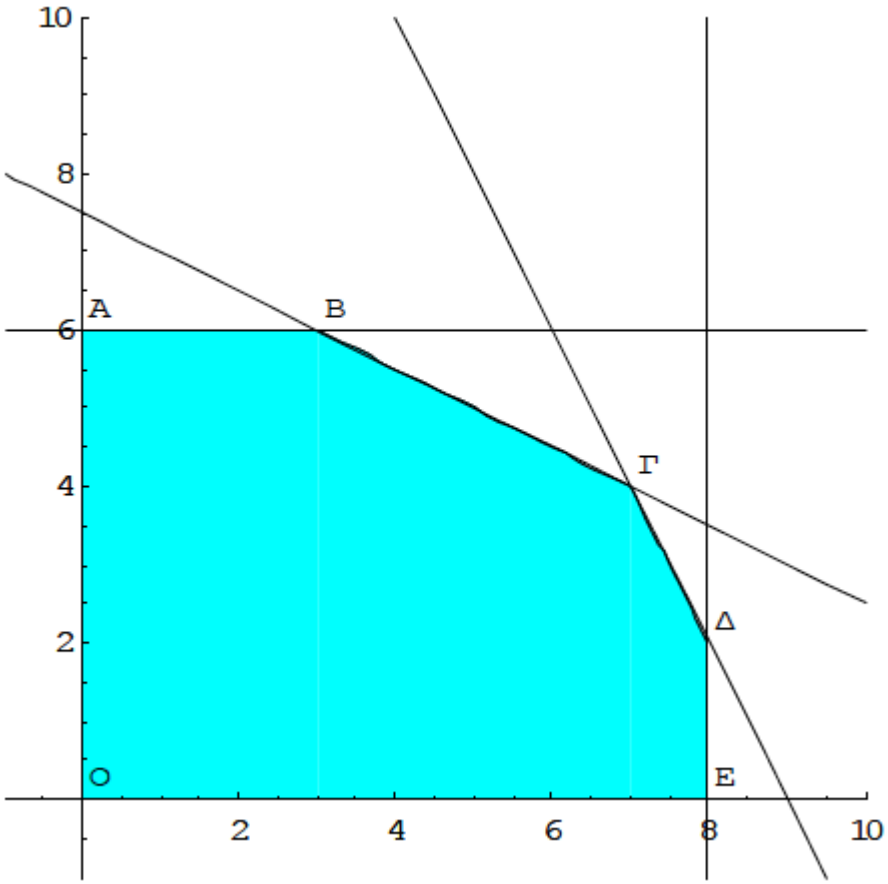
Παράδειγμα 1.2

Θέλουμε να λύσουμε το παρακάτω πρόβλημα Γραμμικού Προγραμματισμού:

$$\max z = 4x_1 + 3x_2$$

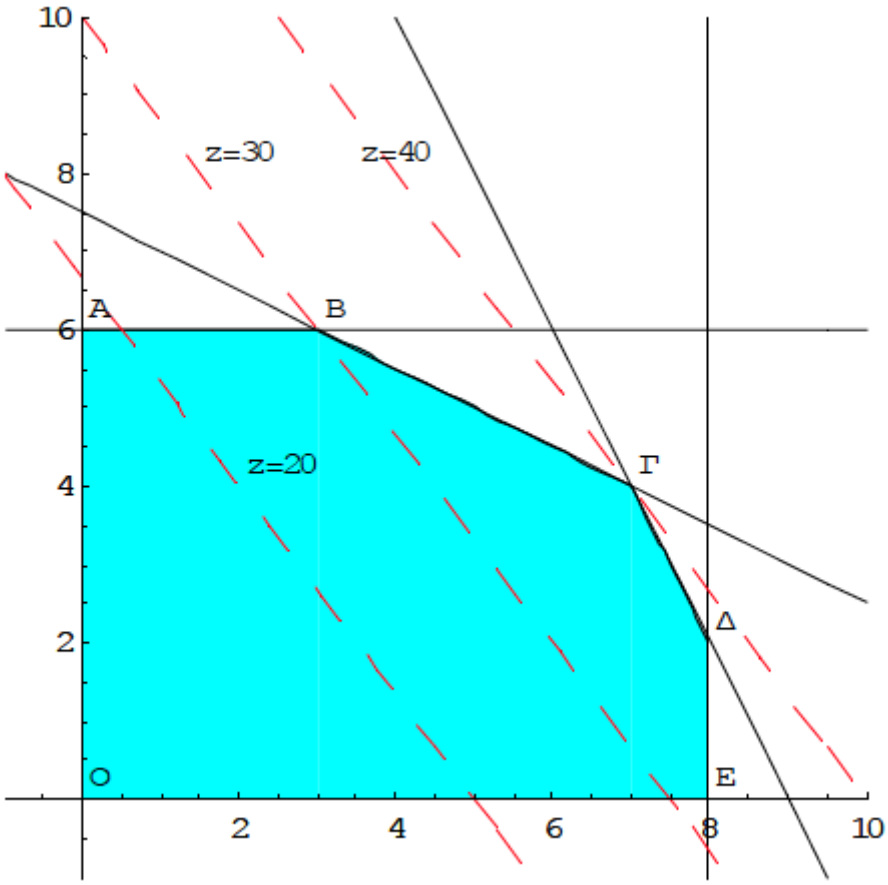
$$\Pi\varepsilon\rho\iota o\rho\iota\sigma\mu o\acute{\iota}:$$
$$x_1 \leq 8$$
$$x_2 \leq 6$$
$$x_1 + 2x_2 \leq 15$$
$$2x_1 + x_2 \leq 18$$
$$x_1, x_2 \geq 0$$

Απεικονίζοντας τις ευθείες που αναπαριστούν τις ανισότητες των περιορισμών στο επίπεδο, αποτυπώνουμε την εφικτή περιοχή η οποία δημιουργείται από τα σημεία τομής των ευθειών (Α, Β, Γ, Δ, Ε και Ο(0,0)), όπως φαίνεται με το γαλάζιο χρώμα.



Εικόνα 17: Διάγραμμα εφικτής περιοχής 1

Στη συνέχεια, σχεδιάζουμε την ευθεία της αντικειμενικής συνάρτησης και μετατοπίζοντάς τη παράλληλα μέχρι να βρούμε το σημείο στο οποίο μεγιστοποιείται.



Εικόνα 18: Διάγραμμα εφικτής περιοχής 2

Παρατηρούμε ότι η ευθεία η οποία εφάπτεται με την εφικτή περιοχή στο σημείο Γ μας δίνει τη μέγιστη τιμή z= 40 η οποία είναι και η λύση μας.

Εδώ αξίζει να σημειωθεί πως στην τελική λύση μπορούμε να φτάσουμε και βασιζόμενοι στο γνωστό θεώρημα σύμφωνα με το οποίο η βέλτιστη ή οι βέλτιστες λύσεις ενός προβλήματος Γραμμικού Προγραμματισμού (αν υπάρχουν) βρίσκονται σε κάποιο (ή κάποια) από τα ακραία σημεία- κορυφές της κλειστής κυρτής περιοχής εφικτών λύσεων του προβλήματος.

Συνεπώς, υπολογίζοντας τα x_1, x_2 των κορυφών του πολυγώνου που σχηματίζει η εφικτή περιοχή (επιλύοντας τα ανάλογα συστήματα εξισώσεων) και δοκιμάζοντας τις λύσεις στην αντικειμενική συνάρτηση θα καταλήξουμε στην ίδια λύση.

Το παραπάνω πρόβλημα με τη χρήση δύο μεταβλητών παρατηρούμε ότι λύνεται εύκολα με γραφικό τρόπο. Βέβαια στην περίπτωση που το πρόβλημα περιείχε περισσότερους περιορισμούς η επίλυσή του θα ήταν

αρκετά πιο πολύπλοκη λόγω έλλειψης ευκρίνειας της εφικτής περιοχής. Όπως προαναφέραμε, στην περίπτωση που το πρόβλημα περιέχει τρείς μεταβλητές η γραφική του αναπαράσταση γίνεται στο χώρο (τρεις διαστάσεις) όπου είναι και εξαιρετικά δύσκολη η επίλυσή του. Από τέσσερεις μεταβλητές και πάνω, η επίλυση με τη χρήση αλγορίθμων Γραμμικού Προγραμματισμού είναι μονόδρομος.

3.4. Μέθοδος SIMPLEX

Εξετάζοντας τη γεωμετρία του γραμμικού προβλήματος στο χώρο των μεταβλητών διαπιστώνουμε ότι αν ένα γραμμικό πρόβλημα είναι βέλτιστο (υπάρχει βέλτιστη λύση), τότε υπάρχει τουλάχιστο μια βέλτιστη κορυφή του εφικτού πολυέδρου. οι αλγόριθμοι τύπου Simplex στηρίζονται σε αυτό το γεγονός. Αναζητούν την βέλτιστη λύση μεταπηδώντας από βασική εφικτή λύση σε βασική εφικτή λύση (μέσα στην εφικτή περιοχή) έτσι ώστε σε κάθε βήμα να βελτιώνεται η τιμή της αντικειμενικής συνάρτησης. Ο αλγόριθμος τερματίζει όταν καταλήξει σε κορυφή από την οποία δεν μπορεί να μεταβεί σε κάποια γειτονική με καλύτερη τιμή της αντικειμενικής συνάρτησης. Αυτή είναι και η βέλτιστη λύση.

Για να τρέξουμε τον αλγόριθμο Simplex αρχικά θα πρέπει να φέρουμε το πρόβλημά μας σε τυποποιημένη μορφή.

Η γενική μορφή του προβλήματος που περιγράψαμε προηγουμένως αποτελεί και την κανονική του μορφή. Στην κανονική μορφή όλες οι μεταβλητές υπακούουν στους περιορισμούς μη αρνητικότητας και όλοι οι τεχνολογικοί περιορισμοί είναι ανισοτικοί.

Το πρώτο βήμα που θα πρέπει να κάνουμε είναι να μετασχηματίσουμε το πρόβλημά μας στην τυποποιημένη του μορφή, κατά την οποία όλες οι μεταβλητές υπακούουν στους περιορισμούς μη αρνητικότητας ενώ όλοι οι τεχνολογικοί περιορισμοί είναι ισοτικοί.

Οι δύο αυτές μορφές του προβλήματος είναι ισοδύναμες, για το λόγω αυτό και ο μετασχηματισμός ονομάζεται μετασχηματισμός ισοδυναμίας.

Ισοδύναμα ορίζουμε τα προβλήματα στα οποία υπάρχει μια ένα προς ένα αντιστοιχία μεταξύ των εφικτών σημείων τους και των αντίστοιχων αντικειμενικών τιμών τους. Ο μετασχηματισμός του προβλήματος από την κανονική στην τυποποιημένη του τιμή επιτυγχάνεται με την εισαγωγή χαλαρών μεταβλητών (slack variables).

Μια ανισότητα της μορφής $a_1x_1+ a_2x_2+\ldots+ a_nx_n\leq b$, μετατρέπεται σε ισότητα με πρόσθεση μιας μεταβλητής x_{n+1} στο αριστερό της μέλος. Τότε η προηγούμενη ανισότητα είναι ισοδύναμη με το σύστημα των δύο περιορισμών $a_1x_1+ a_2x_2+\ldots+ a_nx_n+ x_{n+1}= b$ και $x_{n+1}\geq 0$. Η νέα μη αρνητική μεταβλητή x_{n+1} ονομάζεται ελλειμματική (deficit).

Ένας ανισοτικός περιορισμός της μορφής $a_1x_1+a_2x_2+\ldots+a_nx_n\geq b$, μετατρέπεται σε ισοτικό με αφαίρεση μιας μεταβλητής x_{n+1} στο αριστερό της μέλος. Τότε η προηγούμενη ανισότητα είναι ισοδύναμη με το σύστημα των δύο περιορισμών $a_1x_1+ a_2x_2+\ldots+ a_nx_n- x_{n+1}= b$ και $x_{n+1}\geq 0$. Η νέα μη αρνητική μεταβλητή x_{n+1} ονομάζεται πλεονασματική (surplus).

Παράδειγμα 1.3

Το παρακάτω γραμμικό πρόβλημα μεγιστοποίησης δίνεται στην κανονική του μορφή.

$$\max(x_1 + x_2 - 4x_3 - 15)$$

$$\begin{aligned} -3x_1 + 2x_2 - x_3 &\geq 5 \\ 2x_1 - 3x_2 + 2x_3 &\leq 9 \\ x_1 - x_2 + 3x_3 &\leq 5 \\ x_j &\geq 0, (j = 1,2,3) \end{aligned}$$

Καλούμαστε να το μετατρέψουμε στην τυποποιημένη του μορφή.

Ο πρώτος περιορισμός είναι της μορφής ≥. Με την αφαίρεση της πλεονασματικής μεταβλητής x_4 από το δεξιό του μέλος μετατρέπεται στο ισοδύναμο σύστημα:

$$-3x_1+ 2x_2- x_3- x_4= 5,\ \ x_4\geq 0.$$

Επειδή ο δεύτερος και ο τρίτος περιορισμός είναι της μορφής ≤ προσθέτουμε στα δεξιά τους μέρη τις ελλειμματικές μεταβλητές x5, x6 αντίστοιχα οπότε μετασχηματίζονται στα ισοδύναμα συστήματα:

$$2x_1 - 3x_2 + 2x_3 + x_5 = 9,\ x_5 \geq 0.$$

$$x_1 - x_2 + 3x_3 + x_6 = 5,\ x_6 \geq 0.$$

Επομένως το τυποποιημένο γραμμικό πρόβλημα έχει τη μορφή:

$$\max(x_1 + x_2 - 4x_3 - 15)$$

$$\begin{aligned} -3x_1 + 2x_2 - x_3 - x_4 &\geq 5 \\ 2x_1 - 3x_2 + 2x_3 + x_5 &\leq 9 \\ x_1 - x_2 + 3x_3 + x_6 &\leq 5 \\ x_j &\geq 0, (j = 1,2,3,4,5,6) \end{aligned}$$

Η γενική τυποποιημένη μορφή του γραμμικού προβλήματος είναι:

$$z = \{\min, \max\}(c_1x_1 + c_2x_2 + ... + c_nx_n)$$

$$\begin{aligned} &a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n = b_1 \\ &a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n = b_2 \\ &..... \\ &a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n = b_m \\ &x_1, x_2, ..., x_n \geq 0 \end{aligned}$$

Ισοδύναμα με μορφή πινάκων το πρόβλημα γράφεται ως εξής:

$$z = \{\min, \max\} f(x)c^T x$$

$$\begin{aligned} Ax &= b \\ x \geq 0, b &\geq 0 \end{aligned}$$

Όπου:

$$x = \begin{pmatrix} x_1 \\ x_2 \\ ... \\ x_n \end{pmatrix} \in M_{nx1}, c = \begin{pmatrix} c_1 \\ c_2 \\ ... \\ c_n \end{pmatrix} \in M_{nx1}, b = \begin{pmatrix} b_1 \\ b_2 \\ ... \\ b_n \end{pmatrix} \in M_{mx1}$$

$$A = \begin{pmatrix} a_{11} & ... & a_{1n} \\ ... & ... & ... \\ a_{m1} & ... & a_{mm} \end{pmatrix} \in M_{mxn}$$

Υποθέτουμε ότι m<n και ότι οι γραμμές του πίνακα Α είναι ανεξάρτητες.

Βασικές Λύσεις.

Όπως είδαμε στην τυπική μορφή του προβλήματος το σύστημα Ax= b έχει n μεταβλητές και m γραμμικές εξισώσεις με m≤ n.

Βασική εφικτή λύση του προβλήματος είναι μια λύση που προκύπτει αν οι μεταβλητές m ενός υποπίνακα Β (βασικός πίνακας) του πίνακα Α με διαστάσεις mxm είναι θετικές και διάφορες του μηδενός και οι υπόλοιπες n- m μεταβλητές (μη βασικές μεταβλητές) είναι ίσες με το μηδέν. Απαραίτητη προϋπόθεση, τα m διανύσματα του πίνακα Β (ο οποίος λέγεται και βασικός) να είναι γραμμικώς ανεξάρτητα, ή αλλιώς η ορίζουσά του να είναι διάφορη του μηδενός (det(B)≠ 0). Τότε βασική λύση x_B μπορεί να υπολογιστεί από τη σχέση $x_B= B^{-1}b$.

Παραθέτουμε κάποιες ιδιότητες των λύσεων του γραμμικού συστήματος:

- Ο αριθμός των εφικτών λύσεων του γραμμικού συστήματος είναι πεπερασμένος και το πολύ $\frac{n!}{m!(n-m)!}$
- Μη εκφυλισμένη βασική εφικτή λύση είναι η βασική εφικτή λύση η οποία έχει τουλάχιστο μια από τις βασικές λύσεις ίση με μηδέν.
- Σε ένα πρόβλημα Γραμμικού Προγραμματισμού το σύνολο των εφικτών λύσεων είναι κυρτό κλειστό σύνολο.
- Κάθε βασική εφικτή λύση ενός προβλήματος Γραμμικού Προγραμματισμού είναι μια κορυφή του πολυγώνου της εφικτής περιοχής και αντίστροφα.
- Αν υπάρχει βέλτιστη εφικτή λύση σε ένα πρόβλημα Γραμμικού Προγραμματισμού, τότε αυτή η βέλτιστη λύση βρίσκεται σε κάποια κορυφή του πολυέδρου της εφικτής περιοχής.
- Αν υπάρχει τουλάχιστο μια βέλτιστη εφικτή λύση που δεν είναι βασική, τότε υπάρχουν άπειρες βέλτιστες δυνατές λύσεις.

Πριν ξεκινήσουμε να περιγράφουμε τον αλγόριθμο θα πρέπει να αναφερθούμε στις δυικές μεταβλητές, οι οποίες αποτελούν τα κριτήρια βελτιστότητας τα οποία είναι απαραίτητα για το σταμάτημα των αλγορίθμων. Έχοντας μια οποιαδήποτε βασική διαμέριση (B, N) όπου B ο βασικός πίνακας και N ο εναπομείναν πίνακας με διαστάσεις ((n-m)x m) και B^{-1} ο αντίστροφος βασικός πίνακας, μπορούμε να υπολογίσουμε τις δυικές μεταβλητές w και s με τους τύπους:

$$w^T = (c_B)^T (A_B)^{-1}$$

$$s^T = c^T - w^T A = c^T - (c_B)^T (A_B)^{-1} A$$

Έστω (B, N) μια βασική διαμέριση ενός γραμμικού προβλήματος. Η αντίστοιχη βασική λύση (x_B, x_N) είναι βέλτιστη, αν είναι $x_B \geq 0$ και $s_N \geq 0$.

3.4.1. Μεθοδολογία αλγορίθμων Simplex.

Κάθε αλγόριθμος τύπου Simplex κατασκευάζει μια πεπερασμένη ακολουθία βασικών λύσεων. Αν B η τρέχων βασικός πίνακας και N ο μη τρέχων, τότε αν οι αντίστοιχη βασική λύση (x_B, x_N) είναι βέλτιστη, οι αλγόριθμοι σταματούν. Στην περίπτωση μη βέλτιστης λύσης γίνεται προσπάθεια εύρεσης δεικτών $k \in B$ και $l \in N$. Αν η προσπάθεια αυτή δεν είναι επιτυχής οι αλγόριθμοι σταματούν. Διαφορετικά έχουμε το φαινόμενο της περιστροφής (pivoting) κατά το οποίο οι k και l εναλλάσσονται. Ο δείκτης k γίνεται μη βασικός (εξέρχεται από τον τρέχων βασικό πίνακα), ενώ ο δείκτης l γίνεται βασικός (εισέρχεται στον στο νέο βασικό πίνακα), κατασκευάζεται ο νέος βασικός πίνακας και η διαδικασία επαναλαμβάνεται. Η ονοματολογία των εισερχόμενων και εξερχόμενων μεταβλητών καθώς και των εισερχόμενων και εξερχόμενων στηλών του πίνακα A προσδιορίζεται με βάση τους δείκτες k και l.

Έτσι, με x_k και a_k συμβολίζουμε την εξερχόμενη μεταβλητή και την εξερχόμενη στήλη του πίνακα A αντίστοιχα, και με x_l και a_l την εισερχόμενη μεταβλητή και στήλη. Η θέση στην οποία βρίσκεται η εισερχόμενη μεταβλητή στο σύνολο N συμβολίζεται με t, ενώ η θέση στην οποία βρίσκεται η εξερχόμενη μεταβλητή στο σύνολο B συμβολίζεται με r.

Συνεπώς, η ανανέωση των συνόλων B και N γίνεται απλώς θέτοντας N(t)= k και B(r)= l.

Κάθε αλγόριθμος τύπου Simplex αποτελείτε από τρία βήματα. Το πρώτο βήμα, το ονομάζουμε και Βήμα 0 αποτελεί το ξεκίνημα του αλγορίθμου κατά το οποίο υπολογίζονται όλα τα απαραίτητα στοιχεία για να ξεκινήσει ο αλγόριθμος. Το Βήμα 0 εκτελείται μία μόνο φορά κατά την εκκίνηση και δεν επαναλαμβάνεται. Κατά το επόμενο βήμα (Βήμα 1), εκτελείται ο έλεγχος βελτιστότητας του παρόντος σημείου. Αν το σημείο είναι βέλτιστο, ο αλγόριθμος σταματά, στην αντίθετη περίπτωση γίνεται η επιλογή της εξερχόμενης και της εισερχόμενης μεταβλητής. Αν αποτύχει στην εύρεση της εισερχόμενης και εξερχόμενης μεταβλητής ο αλγόριθμος σταματά. Σε αυτή την περίπτωση, θα πρέπει να αποδειχθεί ότι το γραμμικό πρόβλημα δεν είναι βέλτιστο. Στην περίπτωση που η εύρεση των δύο μεταβλητών είναι επιτυχής ο αλγόριθμος περνάει στο Βήμα 2 κατά το οποίο πραγματοποιείται η ανανέωση των δεδομένων και η διαδικασία των βημάτων 1 και 2 επαναλαμβάνεται.

Στη συνέχεια θα παρακολουθήσουμε τα βήματα του πρωτεύοντος αλγορίθμου Simplex αναλυτικότερα.

Περιγραφή αλγορίθμου.

Υποθέτουμε ότι B μια εφικτή βάση του γραμμικού προβλήματος της μορφής

$\min\{c^T x : Ax = b, x \geq 0\}$, όπου $c, x \in R^n, b \in R^m, A \in R^{mxn}$.

Οι τιμές των βασικών μεταβλητών δίνονται από τη σχέση $x^B = (B)^{-1} b, x^B \geq 0$

Επίσης x_N= 0, όπου N= {1,2,3,…,n} ~ B το σύνολο των μη βασικών δεικτών.

Για τον έλεγχο βελτιστότητας, αφού έχουμε την παρούσα βασική λύση $x_B \geq 0$ (εφικτή βασική λύση) αρκεί να εξετάσουμε αν είναι και $s_N \geq 0$. Αν δηλαδή, $J = \varnothing$, όπου

$$J = \{ j : j \in N, s_j = c_j - w^T a_j < 0 \}.$$

Σε αυτή την περίπτωση, το σημείο που εξετάζουμε είναι βέλτιστο και ο αλγόριθμος τερματίζει.

Αν το σημείο δεν είναι βέλτιστο, επιλέγεται η εισερχόμενη μεταβλητή x_l έτσι ώστε $l \in J$.

Σε αυτό το σημείο αξίζει να πούμε πως ανάλογα με τον αλγόριθμο τύπου Simplex που χρησιμοποιούμε μπορεί να τεθεί και διαφορετικό κριτήριο επιλογής του δείκτη l από το σύνολο J. Οι επιπλέον κανόνες που χρησιμοποιούν οι αλγόριθμοι για την επιλογή της εισερχόμενης μεταβλητής ονομάζονται κανόνες περιστροφής (pivoting rules). Ένας ευρέος διαδεδομένος κανόνας περιστροφής είναι ο κανόνας του Dantzig ή αλλιώς κανόνας του ελαχίστου στοιχείου (least element rule) όπου ο δείκτης l επιλέγεται από τη σχέση: $s_l = \min\{s_j : j \in J\}$.

Στη συνέχεια θα πρέπει να επιλέξουμε την εξερχόμενη μεταβλητή x_k έτσι ώστε $x_k = x_{B(r)}$ με $x_k = 0$ στην επόμενη βάση. Επειδή κατά τη μετάβαση από την τρέχουσα βάση στην επόμενη οι μοναδικές μεταβλητές που αλλάζουν τιμή είναι οι βασικές x_B και η εισερχόμενη x_l, ισχύει η σχέση: $x_B = (B)^{-1}b - h_l x_l$, όπου $h_l = (B)^{-1}a_l$.

Αν $h_l \leq 0$, τότε συμπεραίνουμε ότι $x_B \geq 0 \forall x_l \leq 0$, δηλαδή $I_+ = \{i : h_{il} > 0\} = \varnothing$.

Ελέγχοντας το s_l, στην περίπτωση που το βρούμε αρνητικό τότε το πρόβλημά μας είναι απεριόριστο, και ο αλγόριθμος σταματά. Αλλιώς, η τιμή xl επιλέγεται από τη σχέση:

$x_l = \min\{\frac{(B^{-1}b)_i}{h_{il}} : i \in I_+\}$ με τον κανόνα του ελαχίστου όρου.

Οπότε η μεταβλητή x_l γίνεται πλέον βασική, η x_B μη βασική, κατασκευάζεται μια νέα εφικτή βάση και η διαδικασία επαναλαμβάνεται.

Συνοπτικά τα βήματα του αλγορίθμου:

ΒΗΜΑ 0:

a. Δημιουργία εφικτής βάσης (B, N).
b. Υπολογισμός των στοιχείων: B^{-1}, c_B, c_N, x_B, w^T, s_N^T.

ΒΗΜΑ 1:

a. Έλεγχος Βελτιστότητας: Αν είναι $J = \varnothing$, Σταμάτημα αλγορίθμου, το βέλτιστο σημείο βρέθηκε και είναι το τρέχων. Αλλιώς επιλογή δείκτη l με κάποιον κανόνα περιστροφής. Η μεταβλητή xl είναι εισερχόμενη.
b. Έλεγχος ελαχίστου λόγου: Υπολογισμός h_l και συνόλου I_+. αν $I_+ = \varnothing$, Σταμάτημα αλγορίθμου, το πρόβλημα είναι απεριόριστο. Αλλιώς, επιλογή εξερχόμενης μεταβλητής $x_{B(r)} = x_k$.

ΒΗΜΑ 2:

Θέτουμε N(t)= k και B(r)= l για τον υπολογισμό της νέας βάσης και επανερχόμαστε στο δεύτερο μέρος του βήματος 0.

Ο πρωτεύων αλγόριθμος Simplex που περιγράψαμε χρειάζεται μια εφικτή βάση για να ξεκινήσει. Σε μερικά προβλήματα η βασική λύση, στην οποία όλες οι χαλαρές μεταβλητές είναι βασικές και όλες οι μεταβλητές απόφασης μη βασικές, είναι εφικτή. Σε αυτά τα προβλήματα ο πρωτεύων αλγόριθμος μπορεί να εφαρμοστεί ως έχει. Στις περιπτώσεις που η βασική διαμέριση δεν είναι εφικτή, το πρόβλημα λύνεται χρησιμοποιώντας διάφορες μεθόδους τις οποίες απλά θα αρκεστούμε να τις αναφέρουμε στα πλαίσια αυτής της διπλωματικής εργασίας.

Μια τέτοια μέθοδος είναι η μέθοδος των δύο φάσεων στην οποία ο πρωτεύων αλγόριθμος εφαρμόζεται δύο φορές. Στην πρώτη φάση επιλύεται ένα τροποποιημένο γραμμικό πρόβλημα ώστε να δημιουργηθεί μια βασική εφικτή λύση και να επιλυθεί το πρόβλημά μας στη δεύτερη φάση.

Μια εξίσου σημαντική μέθοδος με τη μέθοδο των δύο φάσεων είναι η μέθοδος του μεγάλου Μ. Στη μέθοδο του μεγάλου Μ δεν λύνονται δύο διαφορετικά προβλήματα πράγμα το οποίο δεν είναι καλό από υπολογιστικής απόψεως, αλλά ένα τροποποιημένο πρόβλημα από το οποίο εξάγονται συμπεράσματα για τη λύση του αρχικού μας προβλήματος. Το τροποποιημένο πρόβλημα το οποίο ονομάζεται πρόβλημα του μεγάλου Μ (big M problem) έχει τις ίδιες μεταβλητές και τους ίδιους περιορισμούς με τη φάση ένα της μεθόδου των δύο φάσεων. Η μόνη διαφορά βρίσκεται στην αντικειμενική συνάρτηση, η οποία αποτελείται από δύο όρους. Ο ένας όρος είναι η αντικειμενική συνάρτηση του αρχικού προβλήματος. Ο άλλος όρος είναι η αντικειμενική συνάρτηση του προβλήματος της φάσης ένα πολλαπλασιασμένη με έναν πάρα πολύ μεγάλο αριθμό Μ.

Παράδειγμα 1.3

Να βρεθεί η βέλτιστη λύση του προβλήματος:

$$\min z = -x_3 - 2x_4$$

$$x_1 + x_3 + x_4 = 5$$
$$-x_2 - 2x_3 - 3x_4 = -3$$
$$x_j \geq 0, (j = 1, 2, 3, 4)$$

Εκφράζουμε τα δεδομένα σε μορφή πινάκων και διανυσμάτων.

$$A = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & -1 & -2 & -3 \end{bmatrix}, c^T = \begin{pmatrix} 0 & 0 & -1 & -2 \end{pmatrix}, b = \begin{bmatrix} 5 \\ -3 \end{bmatrix}$$

Επανάληψη 1

ΒΗΜΑ 0

a. Παρατηρούμε ότι οι μεταβλητές x_1 και x_2 είναι χαλαρές. Η βάση B= [1, 2] είναι εφικτή. Επομένως ο αλγόριθμος μπορεί να ξεκινήσει με την εφικτή διαμέριση:
   B= [1, 2] και N= [3, 4].
b. Υπολογίζονται τα αρχικά δεδομένα:

$$(c_B)^T = (0 \quad 0), (c_N)^T = (-1 \quad -2)$$

$$B = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, N = \begin{bmatrix} 1 & 1 \\ -2 & -3 \end{bmatrix}$$

$$B^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$x_B = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = B^{-1}b = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} 5 \\ -3 \end{bmatrix} = \begin{bmatrix} 5 \\ 3 \end{bmatrix} \geq 0$$

$$w^T = (c_B)^T B^{-1} = (0 \quad 0)$$

$$(s_N)^T = (c_N)^T - w^T N = (c_N)^T = (-1 \quad -2)$$

ΒΗΜΑ 1

a. Παρατηρούμε ότι υπάρχουν αρνητικές τιμές s_j, οπότε ο αλγόριθμος δε σταματά. Σαν εισερχόμενος δείκτης μπορεί να επιλεγεί ο 3 ή ο 4. Επιλέγουμε τυχαία τον 3 οπότε: l= 3, και εισερχόμενη μεταβλητή η x_3. (t=1, N(1)= l= 3).

b. $h_3 = B^{-1}a_3 = \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ -2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$

Επειδή $h_3 \geq 0$ ο αλγόριθμος δε σταματά. Συνεχίζουμε με τον έλεγχο ελαχίστου λόγου.

$$\min\left\{\frac{x_{B[1]}}{h_{13}}, \frac{x_{B[2]}}{h_{23}}\right\} = \min\left\{\frac{5}{1}, \frac{3}{2}\right\} = \frac{3}{2} = \frac{x_{B[2]}}{h_{23}}$$

Άρα r= 2 και k= B(2)= 2. Η εξερχόμενη μεταβλητή είναι η x_2.

ΒΗΜΑ 2:

Θέτουμε B(r)= B(2)= l= 3 και N(t)= N(1)= k= 2.

Τα νέα σύνολα είναι:

B= [1, 3], N= [2, 4]

Επανάληψη 2

ΒΗΜΑ 0

b. Τα νέα δεδομένα είναι:

$$(c_B)^T = (0 \quad -1), (c_N)^T = (0 \quad -2)$$

$$B = \begin{bmatrix} 1 & 1 \\ 0 & -2 \end{bmatrix}, N = \begin{bmatrix} 0 & 1 \\ -1 & -3 \end{bmatrix}$$

$$B^{-1} = -\frac{1}{2}\begin{bmatrix} -2 & -1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1/2 \\ 0 & 1/2 \end{bmatrix}$$

$$x_B = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = B^{-1}b = \begin{bmatrix} 1 & 1/2 \\ 0 & 1/2 \end{bmatrix}\begin{bmatrix} 5 \\ -3 \end{bmatrix} = \begin{bmatrix} 7/2 \\ 3/2 \end{bmatrix} \geq 0$$

$$w^T = (c_B)^T B^{-1} = (0 \quad -1)\begin{bmatrix} 1 & 1/2 \\ 0 & 1/2 \end{bmatrix} = \begin{pmatrix} 0 & \frac{1}{2} \end{pmatrix}$$

$$(s_N)^T = (c_N)^T - w^T N = (0 \quad -2) - \begin{pmatrix} 0 & \frac{1}{2} \end{pmatrix}\begin{bmatrix} 0 & 1 \\ -1/2 & -3/2 \end{bmatrix} = (\frac{1}{2} \quad -\frac{1}{2})$$

ΒΗΜΑ 1

a. Παρατηρούμε ότι υπάρχουν αρνητικές τιμές s_j, οπότε ο αλγόριθμος δε σταματά. Εισερχόμενος είναι ο 4. Οπότε: l= 4, και εισερχόμενη μεταβλητή η x_4.

b. $h_4 = B^{-1}a_4 = \begin{bmatrix} 1 & 1/2 \\ 0 & -1/2 \end{bmatrix}\begin{bmatrix} 1 \\ -3 \end{bmatrix} = \begin{bmatrix} -1/2 \\ 3/2 \end{bmatrix}$

Επειδή $h_4 \geq 0$ ο αλγόριθμος δε σταματά. Εξερχόμενη μεταβλητή είναι η x_3, r= 2 και k= B(2)= 3.

ΒΗΜΑ 2:

Θέτουμε B(r)= B(2)= l= 4 και N(t)= N(1)= k= 3.

Τα νέα σύνολα είναι:

B= [1, 4], N= [2, 3]

Επανάληψη 3

ΒΗΜΑ 0

b. Τα νέα δεδομένα είναι:

$$(c_B)^T = (0 \quad -2), (c_N)^T = (0 \quad -1)$$

$$B = \begin{bmatrix} 1 & 1 \\ 0 & -3 \end{bmatrix}, N = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix}$$

$$B^{-1} = -\frac{1}{3}\begin{bmatrix} -3 & -1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1/3 \\ 0 & -1/3 \end{bmatrix}$$

$$x_B = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = B^{-1}b = \begin{bmatrix} 1 & 1/3 \\ 0 & -1/3 \end{bmatrix}\begin{bmatrix} 5 \\ -3 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \end{bmatrix} \geq 0$$

$$w^T = (c_B)^T B^{-1} = (0 \quad -1)\begin{bmatrix} 1 & 1/3 \\ 0 & -1/3 \end{bmatrix} = \begin{pmatrix} 0 & \frac{2}{3} \end{pmatrix}$$

$$(s_N)^T = (c_N)^T - w^T N = (0 \quad -1) - \begin{pmatrix} 0 & \frac{2}{3} \end{pmatrix}\begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix} = (\frac{2}{3} \quad \frac{1}{3})$$

ΒΗΜΑ 1

a. έχουμε $s_j \geq 0$ οπότε ο αλγόριθμος σταματά. Η τρέχουσα βάση είναι βέλτιστη. Η βασική λύση είναι:

$$(x_B)^T = \begin{pmatrix} x_1 & x_2 \end{pmatrix} = \begin{pmatrix} 4 & 1 \end{pmatrix} \text{ και } (x_N)^T = \begin{pmatrix} x_2 & x_3 \end{pmatrix} = \begin{pmatrix} 0 & 0 \end{pmatrix}$$

Και η βέλτιστη αντικειμενική τιμή:

$$z = (c_B)^T x_B = \begin{pmatrix} 0 & -2 \end{pmatrix} \begin{pmatrix} 4 & 1 \end{pmatrix}^T = -2$$

Στο παρόν κεφάλαιο προσπαθήσαμε να παρουσιάσουμε τα βασικά σημεία του πρωτεύοντος αλγόριθμου Simplex. Παρουσιάσαμε μια απλή μορφή του με τη χρήση πινάκων. Υπάρχουν διάφορες μορφές υλοποίησης του συγκεκριμένου αλγορίθμου, όπως η μορφή tableau ή η μορφή λεξικών.

Στην οικογένεια αλγορίθμων Simplex εκτός του πρωτεύοντος αλγορίθμου υπάρχουν πολλοί ακόμη αλγόριθμοι με εξαιρετικό ενδιαφέρον όπως ο δυικός αλγόριθμος Simplex ο οποίος επιλύει ουσιαστικά το συμπληρωματικό πρόβλημα του primal Simplex ή ο αλγόριθμος εξωτερικών σημείων, ο οποίος υπερτερεί σημαντικά του κλασικού αλγορίθμου ερευνώντας κορυφές οι οποίες βρίσκονται έξω από την εφικτή περιοχή με στόχο την εύρεση συντομότερου δρόμου για τη βέλτιστη λύση.

Γεωμετρικά, ο κλασσικός αλγόριθμος Simplex όπως είδαμε κινείται κατά μήκος των ακμών του πολυέδρου που δημιουργεί η εφικτή περιοχή. Αυτή η διαδρομή είναι γνωστή και ως διαδρομή Simplex.

Ο αλγόριθμος εξωτερικών σημείων τύπου Simplex δημιουργεί δυο δρόμος για τη βέλτιστη λύση. Ο ένας δρόμος ακολουθεί εφικτά σημεία, ενώ ο άλλος μη εφικτά. Με αυτό τον τρόπο ο αλγόριθμος μπορεί να ακολουθήσει συντομότερη διαδρομή παρακάμπτοντας τα σύνορα της εφικτής περιοχής. Είναι πολύ σημαντικό όμως να μη χαθεί η επαφή με την εφικτή περιοχή διαφορετικά η επάνοδός του στην εφικτή περιοχή και κατά συνέπεια η εύρεση της βέλτιστης λύσης είναι εξαιρετικά δύσκολη αν όχι ανέφικτη. Αυτή την επαφή την εξασφαλίζει ο δρόμος εφικτών σημείων, ο οποίος όμως δεν είναι δρόμος Simplex. Έτσι, καταφέρνουμε να αποφύγουμε το σημαντικότερο υπολογιστικό μειονέκτημα των εφικτών αλγορίθμων Simplex το οποίο είναι ότι κινούνται από μια κορυφή στην γειτονική της.

Παρόλο που ανακαλύφθηκαν νέου τύπου αλγόριθμοι γραμμικού προγραμματισμού όπως οι αλγόριθμοι εξωτερικών σημείων οι οποίοι είναι εξαιρετικά αποτελεσματικοί σε προβλήματα μεγάλης κλίμακας ή ο

ελλειψοειδής η ρώσικος αλγόριθμος ο οποίος παρουσιάζει μια ιδιομορφία, ενώ η μέση πολυπλοκότητά του είναι πολυωνυμική εντούτοις δουλεύει πάντα στην χειρότερη περίπτωση, πράγμα που τον καθιστά αργότερο από τον Simplex, το ενδιαφέρον για τους αλγορίθμους Simplex έχει κρατηθεί αμείωτο από την ερευνητική κοινότητα με συνεχείς προσπάθειες για την βελτίωση της επιστημονικής του συμπεριφοράς.

3.5. Ακέραιος Γραμμικός Προγραμματισμός

Πολλές φορές η βέλτιστη λύση που μας εξασφαλίζει ο αλγόριθμος Simplex (ή οποιοσδήποτε άλλος αλγόριθμος Γραμμικού Προγραμματισμού) δεν μας καλύπτει όσο αφορά τη φυσική υπόσταση του προβλήματος. Για παράδειγμα αν το πρόβλημά μας έχει να κάνει με εργατικό δυναμικό ή με την απόφαση αν μια επιχείρηση θα πρέπει ή όχι να χρηματοδοτηθεί, μια λύση η οποία μας δίνει 5,476 εργάτες ή 1,0973 να χρηματοδοτηθεί η επιχείρηση δεν μας ικανοποιεί. Σε αυτές τις περιπτώσεις δημιουργείτε από τη φύση του προβλήματος ακόμη ένας περιορισμός, ότι οι λύσεις εκτός από μη αρνητικές θα πρέπει να είναι και ακέραιες. Τα προβλήματα του Γραμμικού Προγραμματισμού στα οποία όλες οι μεταβλητές απόφασης υποχρεούνται να πάρουν ακέραιες τιμές εμπίπτουν στο πεδίο του Ακέραιου Προγραμματισμού (Integer Programming). Όταν μας ενδιαφέρει κάποιες και όχι όλες από τις μεταβλητές απόφασης να είναι ακέραιες, τότε αναφερόμαστε σε προβλήματα Μικτού Προγραμματισμού (Mixed Integer Programming). Στον ακέραιο Προγραμματισμό οι αντικειμενικές συναρτήσεις καθώς και οι περιορισμοί εκφράζονται είτε γραμμικά είτε μη γραμμικά. Επειδή αναφερόμαστε σε διακεκριμένες τιμές και όχι σε συνεχείς, εκ των πραγμάτων ο Ακέραιος Προγραμματισμός είναι μη γραμμικός. Παρόλο αυτά, μπορούμε να μετατρέψουμε τα προβλήματα του σε προβλήματα Γραμμικού Προγραμματισμού αν χαλαρώνοντας τους ακέραιους περιορισμούς των μεταβλητών προκύπτουν γραμμικές συναρτήσεις. Επίσης, αξίζει να αναφέρουμε και την ιδιαίτερη περίπτωση προβλημάτων Ακεραίου Προγραμματισμού όπου όλες οι κάποιες μεταβλητές τους καλούνται να είναι δυαδικές. Αν είναι όλες δυαδικές, τότε αναφερόμαστε σε προβλήματα

Δυαδικού Γραμμικού Προγραμματισμού (binary integer programming) ή αλλιώς σε προβλήματα μηδέν- ένα (zero- one problems).

Η χρήση του Ακεραίου Προγραμματισμού αύξησε σημαντικά το εύρος εφαρμογών του Γραμμικού Προγραμματισμού παρόλο το ότι η δυσκολία επίλυσης τέτοιου είδους προβλημάτων εξακολουθεί μέχρι και σήμερα να μας προβληματίζει. Ενδεικτικά αξίζει να αναφέρουμε ότι ένα πρόβλημα Γραμμικού Προγραμματισμού με εκατοντάδες χιλιάδες συνεχείς μεταβλητές μπορεί να επιλυθεί εύκολα με αρκετά λογισμικά του εμπορίου. Αντίθετα, είναι πολύ δύσκολο να επιλυθεί ένα πρόβλημα Ακεραίου Προγραμματισμού με μόλις 100 μεταβλητές (Papadimitriou Christos et al., 1982).

Μαθηματικό μοντέλο.

Η μαθηματική διατύπωση του γενικού προβλήματος του Ακεραίου Προγραμματισμού είναι:

$$z = \{\min, \max\} g_0(x_1, x_2, ..., x_n)$$

$$g_i(x_1, x_2, ..., x_n) \begin{Bmatrix} \leq \\ \geq \\ = \end{Bmatrix} b_i, i \in M \equiv \{1, 2, ..., m\}$$

$$x_j \geq 0,\ j \in N \equiv \{1, 2, ..., n\}$$

$$x_j = \text{int}\, eger,\ j \in I \subseteq N$$

Αν I = N τότε όλες οι μεταβλητές απόφασης παίρνουν ακέραιες τιμές, οπότε το πρόβλημά μας είναι πρόβλημα ακεραίου προγραμματισμού, αν $I \subset N$ κάποιες από τις μεταβλητές απόφασης μπορούν να παίρνουν συνεχείς τιμές οπότε αναφερόμαστε σε προβλήματα μικτού προγραμματισμού.

Όταν οι συναρτήσεις (αντικειμενική συνάρτηση και συναρτήσεις των περιορισμών) είναι γραμμικές, τότε το μαθηματικό μοντέλο παίρνει τη μορφή:

$$z = \{\min, \max\}(c_1 x_1 + c_2 x_2 + ... + c_n x_n)$$

$$a_{11} x_1 + a_{12} x_2 + ... + a_{1n} x_n \leq, =, \geq b_1$$
$$a_{21} x_1 + a_{22} x_2 + ... + a_{2n} x_n \leq, =, \geq b_2$$
$$.....$$
$$a_{m1} x_1 + a_{m2} x_2 + ... + a_{mn} x_n \leq, =, \geq b_m$$
$$x_1, x_2, ..., x_n \geq 0$$
$$x_1, x_2, ..., x_n, \text{int}\, eger$$

3.5.1. Μέθοδοι επίλυσης

Η συνθήκη ακεραίου (Integrality Condition) που εντάσσεται στον ακέραιο γραμμικό προγραμματισμό, διαφοροποιεί κατά πολύ τον τρόπο προσέγγισης της λύσης σε σχέση με τον συνεχή γραμμικό προγραμματισμό και αυτό γιατί αναζητούνται διακριτές λύσεις στην εφικτή περιοχή. Κατά βάση, η συνθήκη ακεραίου αλλοιώνει τη βασική ιδιότητα της εφικτής περιοχής που είναι η κυρτότητα. Συνεπώς, ο αλγόριθμος Simplex δεν μπορεί να εφαρμοστεί σε τέτοιου είδους προβλήματα μιας και η αναζήτηση των εφικτών λύσεων γίνεται στις κορυφές του πολυτρόπου. Παρόλα αυτά, λόγω της ευχρηστίας και της ακρίβειας που μας παρέχουν τέτοιου είδους αλγόριθμοι Γραμμικού Προγραμματισμού συνηθίζεται να χρησιμοποιούνται στον Ακέραιο Προγραμματισμό αφού πρώτα η εφικτή περιοχή μετατραπεί σε ισοδύναμο συνεχή και κυρτό χώρο.

Οι μέθοδοι επίλυσης που χρησιμοποιούνται κατατάσσονται σε τρείς κατηγορίες:

- Μέθοδοι Στρογγυλοποίησης (Rounding Methods)
- Μέθοδοι Αναζήτησης (Searching Methods)
- Μέθοδοι Τομών (Cutting Methods)

Με τη στρογγυλοποίηση η δυνατότητα απόκτησης αξιόπιστων αποτελεσμάτων είναι αρκετά περιορισμένη, ιδιαίτερα σε περιπτώσεις που το πεδίο ορισμού των μεταβλητών είναι περιορισμένο. Ο τρόπος λειτουργίας τους βασίζεται στην στρογγυλοποίηση των αποτελεσμάτων που παίρνουμε επιλύοντας το πρόβλημα ως γραμμικό συνεχές. Κατά πάσα πιθανότητα η λύση που θα προκύψει θα είναι δυνατή, αλλά όχι απαραίτητα βέλτιστη.

Οι μέθοδοι αναζήτησης βασίζονται στην ύπαρξη πεπερασμένου αριθμού ακέραιων λύσεων. Μια τεχνική η οποία εφαρμόζεται κυρίως σε προβλήματα με δυαδικές μεταβλητές είναι η τεχνική της απαρίθμησης (enumeration), κατά την οποία στην απλούστερη μορφή της γίνεται απαρίθμηση όλων των λύσεων και στη συνέχεια επιλέγεται η βέλτιστη. Σε αυτή την κατηγορία μεθόδων υπάγεται και η οικογένεια αλγορίθμων κλάδου-ορίου (Branch and Bound) όπου χρησιμοποιείται ευρέως. Η τεχνική κλάδου-ορίου βασίζεται στη μέθοδο Διαίρει και Βασίλευε κατά την οποία διαιρείται τμηματικά η εφικτή περιοχή μέχρι την απομόνωση του τμήματος που περιέχει την ακέραιη βέλτιστη λύση. Θα αναφερθούμε παρακάτω ενδελεχώς στη συγκεκριμένη τεχνική.

Τέλος, οι μέθοδοι τομών εισάγουν σταδιακά περιορισμούς ώστε να μικραίνει η περιοχή των εφικτών λύσεων μέχρι να αποκτηθεί η βέλτιστη ακέραιη λύση (Μ. Μποναζούντας, 2001) .

3.5.2. Αλγόριθμοι Branch and Bound

Η μέθοδος Branch and Bound δεν είναι μια τεχνική επίλυσης προβλημάτων η οποία περιορίζεται στο να εφαρμόζεται μόνο σε προβλήματα ακεραίου προγραμματισμού. Είναι μια μέθοδος ευρέσεως λύσης που μπορεί να εφαρμοστεί σε πολλούς τύπους προβλημάτων. Τεχνική της βασίζεται στην αρχή ότι ένα σύνολο εφικτών λύσεων μπορούμε να το διαιρέσουμε σε μικρότερα υποσύνολα των λύσεων. Αυτά τα υποσύνολα μπορούν να εκτιμηθούν συστηματικά μέχρι να βρεθεί η βέλτιστη λύση. Σε ένα πρόβλημα ακεραίου προγραμματισμού η Branch and Bound τεχνική σε χρησιμοποιώντας την μη ακέραιη λύση του προβλήματος θα μας δώσει τη βέλτιστη ακέραια λύση.

Έστω η γενική μορφή του προβλήματος ακεραίου προγραμματισμού όπως την περιγράψαμε και προηγουμένως:

$$\max z = c^T x$$

$$Ax \le b$$
$$x \ge 0, x \,\text{int}\, eger$$

Τα βήματα των αλγορίθμων Branch and Bound είναι τα παρακάτω τέσσερα:

- **ΒΗΜΑ 1:**

  Λύνεται το πρόβλημα του ακεραίου γραμμικού προγραμματισμού ως πρόβλημα γραμμικού προγραμματισμού, χωρίς δηλαδή να ληφθούν υπόψη οι περιορισμοί ακεραιότητας (relaxed solution).

- **ΒΗΜΑ 2:**

  Εάν η λύση που βρέθηκε στο ΒΗΜΑ 1 ικανοποιεί τους περιορισμούς ακεραιότητας ο αλγόριθμος σταματάει. Η βέλτιστη λύση βρέθηκε.

  Εάν η λύση δεν ικανοποιεί τους περιορισμούς ακεραιότητας καθορίζεται μια πρώτη ακέραια λύση μετά από στρογγυλοποιήσεις των μεταβλητών με μη ακέραιες τιμές και υπολογίζεται η αντικειμενική συνάρτηση z. Η τιμή της αποτελεί το αρχικό κάτω φράγμα.

- **ΒΗΜΑ 3:**

  Το σύνολο των μη ακέραιων λύσεων διακλαδίζεται σε δύο υποσύνολα εισάγοντας νέους περιορισμούς για να ικανοποιηθεί ο περιορισμός ακεραιότητας για μια βασική μεταβλητή με ρητή τιμή.

  Για παράδειγμα, έστω ότι η βέλτιστη λύση του γραμμικού προβλήματος είναι η $x^*= (x_1^*, x_2^*, x_3^*,\ldots x_n^*)^T$ και η τιμή x_i^* της μεταβλητής x_i δεν είναι ακέραιη, τότε δημιουργούνται τα εξής δύο υποπροβλήματα:

$"A"$	$"B"$
$\max z = c^T x$	$\max z = c^T x$
$Ax \le b$	$Ax \le b$
$x \ge 0$	$x \ge 0$
$x_i \le \lfloor x_i^* \rfloor$	$x_i \ge \lfloor x_i^* \rfloor + 1$

  Όπου με $[x_i^*]$ αναφερόμαστε στο ακέραιο μέρος του αριθμού x_i^*.

- **ΒΗΜΑ 4:**

  Σε κάθε υποσύνολο λύσεων επιλύεται εκ νέου το γραμμικό πρόβλημα και η τιμή της αντικειμενικής συνάρτησης της βέλτιστης μη ακέραιης λύσης αποτελεί το άνω φράγμα. Η τιμή της αντικειμενικής συνάρτησης της καλύτερης μέχρι τώρα ακέραιης λύσης αποτελεί το κάτω φράγμα. Τα υποσύνολα λύσεων για τα οποία τα άνω φράγματα έχουν μικρότερη τιμή από το κάτω φράγμα δεν εξετάζονται περεταίρω. Εάν βρεθεί εφικτή

ακέραιη λύση με τιμή αντικειμενικής συνάρτησης μεγαλύτερη ή ίση του άνω φράγματος κάθε υποσυνόλου τότε αυτή η λύση είναι και η βέλτιστη λύση του προβλήματος. Εάν όχι, επιλέγεται το υποσύνολο με το καλύτερο άνω φράγμα και επαναλαμβάνεται το ΒΗΜΑ 3.

Παράδειγμα 1.4

Έχουμε το παρακάτω πρόβλημα ακέραιου γραμμικού προγραμματισμού:

$$\max z = 40x_1 + 30x_2$$

$$2x_1 + 2x_2 \le 59$$
$$3x_1 + 2x_2 \le 75$$
$$x_1 + 2x_2 \le 50$$
$$x_1, x_2 \ge 0 \,\text{int}\, eger$$

- ΒΗΜΑ 1:

Επιλύοντας το πρόβλημα με τη μέθοδο Simplex χωρίς να λάβουμε υπόψη μας τους περιορισμούς ακεραιότητας παίρνουμε τη βέλτιστη λύση η οποία είναι:

x_1= 22,75, x_2=6,75 και z= 1112,5.

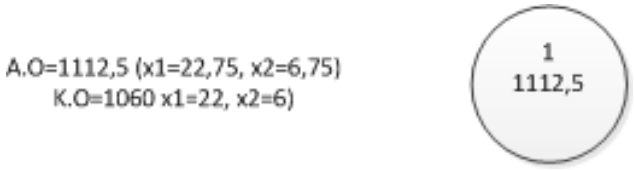
Η τιμή της αντικειμενικής συνάρτησης αποτελεί το άνω φράγμα.

Το κάτω φράγμα υπολογίζεται στρογγυλοποιώντας προς τα κάτω τις μεταβλητές απόφασης. Επομένως x_1= 22, x_2= 6 και z= 1060.

- ΒΗΜΑ 2:

Η βέλτιστη λύση του γραμμικού προβλήματος δεν ικανοποιεί τους περιορισμούς ακεραιότητας.

Ο πρώτος κόμβος σχηματίζεται:



- ΒΗΜΑ 3:

Το αρχικό πρόβλημα διακλαδίζεται σε δύο υποπροβλήματα.

Επιλέγεται η βασική μεταβλητή η οποία δεν ικανοποιεί το κριτήριο της ακεραιότητας για να γίνει η διακλάδωση. Στο συγκεκριμένο παράδειγμα

καμία από τις βασικές μεταβλητές δεν ικανοποιούν το συγκεκριμένο κριτήριο και επιλέγουμε αυθαίρετα την μεταβλητή x_1.

Υποπρόβλημα Α:

$$\max z = 40x_1 + 30x_2$$

$$2x_1 + 2x_2 \le 59$$
$$3x_1 + 2x_2 \le 75$$
$$x_1 + 2x_2 \le 50$$
$$x_1 \le 22$$
$$x_1, x_2 \ge 0$$

Υποπρόβλημα Β:

$$\max z = 40x_1 + 30x_2$$

$$2x_1 + 2x_2 \le 59$$
$$3x_1 + 2x_2 \le 75$$
$$x_1 + 2x_2 \le 50$$
$$x_1 \ge 23$$
$$x_2 \ge 0$$

- ΒΗΜΑ 1:

Επιλύοντας τα δύο υποπροβλήματα με τη μέθοδο Simplex χωρίς να λάβουμε υπόψη μας τους περιορισμούς ακεραιότητας παίρνουμε τη βέλτιστη λύση η οποία είναι:

Για το υποπρόβλημα Α: x_1= 22, x_2=7,5 και z= 1105.

Για το υποπρόβλημα Β: x_1= 23, x_2=6 και z= 1100.

Η τιμή της αντικειμενικής συνάρτησης αποτελεί το άνω φράγμα. Έτσι το άνω φράγμα για το υποπρόβλημα Α είναι 1105, ενώ για το υποπρόβλημα Β είναι 1100.

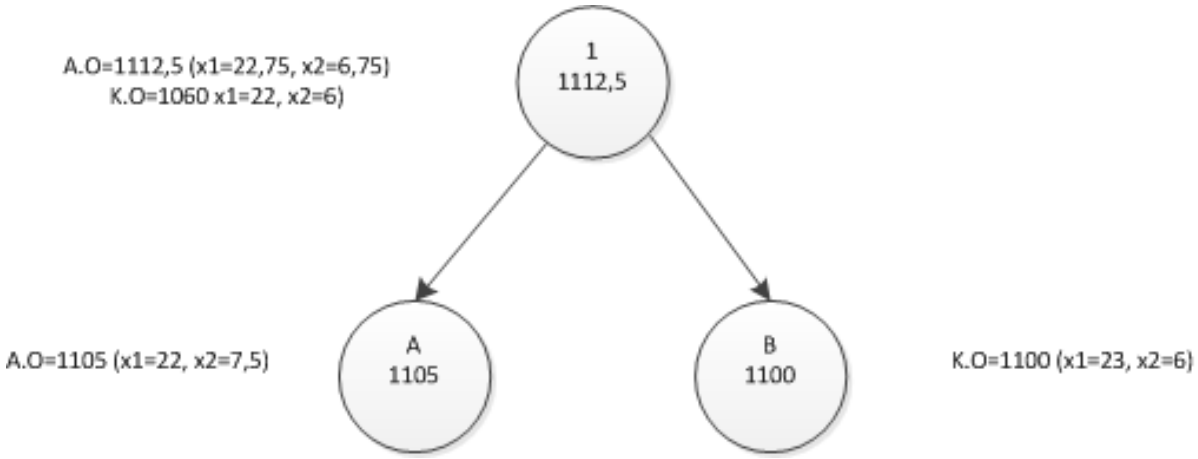
Το κάτω φράγμα υπολογίζεται στρογγυλοποιώντας προς τα κάτω τις μεταβλητές απόφασης. Επομένως για το υποπρόβλημα Α έχουμε: x_1= 22, x_2= 7 και z= 1105, ενώ για το υποπρόβλημα Β έχουμε: x_1= 23, x_2=6 και z= 1100.

- ΒΗΜΑ 2:

Η βέλτιστη λύση του γραμμικού υποπροβλήματος Β ικανοποιεί τους περιορισμούς ακεραιότητας και δίνει ένα νέο κάτω φράγμα (1100). Από τον κόμβο Β δεν συνεχίζεται η διακλάδωση.

Η βέλτιστη λύση του γραμμικού υποπροβλήματος Α δεν ικανοποιεί τους περιορισμούς ακεραιότητας, μας δίνει όμως άνω φράγμα 1105, το οποίο είναι μεγαλύτερο από το νέο κάτω φράγμα οπότε από αυτόν τον κόμβο θα έχουμε περαιτέρω διακλάδωση.

Μέχρι στιγμής το σχήμα μας είναι:



- ΒΗΜΑ 3:

Το υποπρόβλημα Α διακλαδίζεται σε δύο νέα υποπροβλήματα.

Επιλέγεται η βασική μεταβλητή η οποία δεν ικανοποιεί το κριτήριο της ακεραιότητας για να γίνει η διακλάδωση που είναι η x_2.

Υποπρόβλημα Α1:

$$\max z = 40x_1 + 30x_2$$

$$2x_1 + 2x_2 \le 59$$
$$3x_1 + 2x_2 \le 75$$
$$x_1 + 2x_2 \le 50$$
$$x_1 \le 22$$
$$x_2 \le 7$$
$$x_1, x_2 \ge 0$$

Υποπρόβλημα Α2:

$$\max z = 40x_1 + 30x_2$$

$$2x_1 + 2x_2 \le 59$$
$$3x_1 + 2x_2 \le 75$$
$$x_1 + 2x_2 \le 50$$
$$x_1 \le 22$$
$$x_2 \ge 8$$
$$x_1 \ge 0$$

- ΒΗΜΑ 1:

Επιλύοντας τα δύο υποπροβλήματα Α1 και Α2 με τη μέθοδο Simplex χωρίς να λάβουμε υπόψη μας τους περιορισμούς ακεραιότητας παίρνουμε τη βέλτιστη λύση η οποία είναι:

Για το υποπρόβλημα Α1: x_1= 22, x_2=7 και z= 1090.

Για το υποπρόβλημα Α2: x_1= 21,5, x_2=8 και z= 1100.

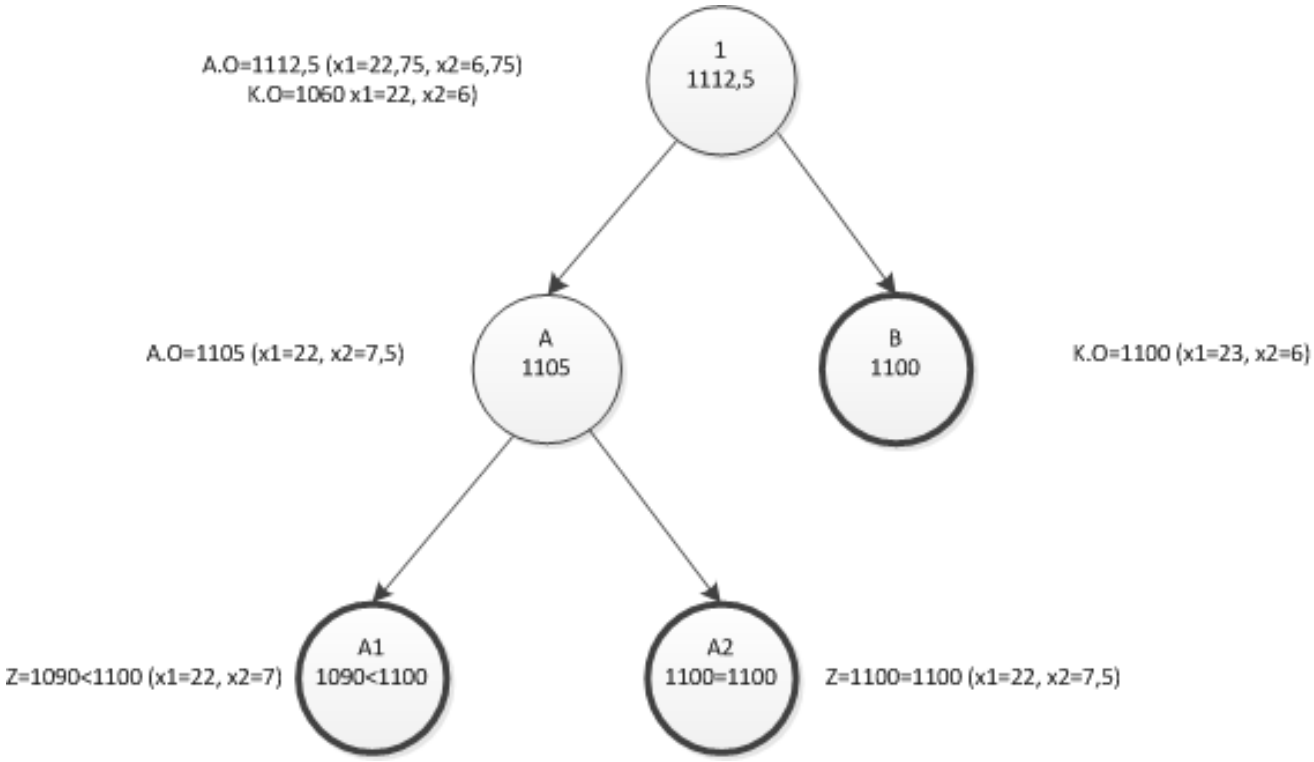
- ΒΗΜΑ 2:

Παρατηρούμε ότι οι τιμές των αντικειμενικών συναρτήσεων και των δύο υποπροβλημάτων δεν ξεπερνούν το κάτω φράγμα που έχει τεθεί με το

υποπρόβλημα Β. οπότε ο αλγόριθμος τερματίζει εδώ. Η βέλτιστη ακέραιη λύση του προβλήματος είναι:

x_1= 23, x_2= 6 και z= 1100.

Το ολοκληρωμένο δέντρο του αλγορίθμου είναι:



Παρατηρήσεις:

- Η διαδικασία έκβασης του αλγορίθμου θα μπορούσε να ήταν διαφορετική αν στην αρχική διακλάδωση επιλέγαμε τη βασική μεταβλητή x_2 αντί της x_1 που τελικά επιλέξαμε, μιας και η επιλογή ήταν αυθαίρετη λόγω του ότι και η δυο μεταβλητές είχαν ρητή τιμή. Υπάρχουν διάφορες τεχνικές για τέτοιες περιπτώσεις πέρα από την τυχαία επιλογή, όπως η επιλογή μεταβλητής με το μεγαλύτερο δεκαδικό μέρος στην τιμή της.
- Η επιλογή της αρχικής ακεραίας τιμής ως αρχικό κάτω φράγμα δεν είναι υποχρεωτική, ωστόσο είναι πολύ σημαντική για τη διεξαγωγή της μεθόδου για το λόγο ότι όσο ποιο υψηλό είναι το αρχικό κάτω φράγμα τόσο ποιο γρήγορα θα συγκλίνει ο αλγόριθμος. Παίζει επομένως

ιδιαίτερο ρόλο η επιλογή μιας καλής αρχικής λύσης, πράγμα το οποίο σε μερικές περιπτώσεις αποτελεί αρκετά δύσκολο εγχείρημα.

Η μέθοδος βελτιστοποίησης ακέραιων γραμμικών προβλημάτων Branch and Bound είναι ευρέως διαδεδομένη. Έχουνε γίνει πολλές τροποποιήσεις και έχουν δημιουργηθεί πολλοί αλγόριθμοι βασιζόμενοι σε αυτή τη μέθοδο. Ενδεικτικά αναφέρουμε τον αλγόριθμο Branch and Cut, όπου τροποποιεί τη βασική μέθοδο με μια μέθοδο cutting ώστε να ενισχύσει τη χαλάρωση του γραμμικού προβλήματος με την εισαγωγή νέων ανισοτήτων πριν τη διακλάδωση του προβλήματος σε υποπροβλήματα, και ο αλγόριθμος Branch and price, ο οποίος είναι παραπλήσιος σε λειτουργία με τον Branch and Cut μόνο που εστιάζει στην εισαγωγή νέων μεταβλητών και όχι νέων περιορισμών όπως ο Branch and Cut.

Στην πραγματικότητα, και δυο διαδικασίες (pricing και cutting) είναι συμπληρωματικές τεχνικές που ενισχύουν την χαλάρωση του γραμμικού προβλήματος που καλείται η μέθοδος να λύσει στο πρώτο της βήμα.

Όπως είναι φυσικό, η μέθοδος Branch and Bound (με όλες τις παραλλαγές της) χρησιμοποιείται ευρέως και για την επίλυση μεικτών γραμμικών προβλημάτων. Σε τέτοιες περιπτώσεις οι μεταβλητές απόφασης που δεν υπόκεινται σε περιορισμούς ακεραιότητας μένουν ως έχουν και η παραπάνω μεθοδολογία εφαρμόζεται στις υπόλοιπες.

3.5.3. Επίλυση προβλημάτων 0- 1.

Όπως αναφέραμε και στην αρχή του κεφαλαίου, τα δυαδικά προβλήματα αποτελούν ένα πολύ σημαντικό κομμάτι του ακεραίου προγραμματισμού. Η μεθοδολογία Branch and Bound αποτελεί ένα πολύ χρήσιμο εργαλείο για την επίλυση τέτοιου τύπου προβλημάτων. Οι περιορισμοί των μεταβλητών απόφασης στα προβλήματα 0- 1 είναι του τύπου $x_j \leq 0$. Η φυσική σημασία αυτού του περιορισμού είναι ότι η μεταβλητή μπορεί να πάρει είτε την τιμή 0, είτε την τιμή 1. Δηλαδή, το μέγεθος το οποίο εκφράζει μπορεί να επηρεάζει τη βέλτιστη τιμή ή όχι. Στο παρακάτω παράδειγμα, θα δούμε ένα πρόβλημα εκφρασμένο σε φυσική γλώσσα για την καλύτερη

κατανόηση της σημασίας των δυαδικών μεταβλητών (Carlos E. Ferreira, 1997).

Παράδειγμα 1.5

Το δημοτικό συμβούλιο μιας πόλης θα πρέπει να αποφασίσει ποιες εγκαταστάσεις θα πρέπει να κατασκευάσει στα πλαίσια ενός προγράμματος ανάπλασης του δήμου. Έχουν προταθεί οι τέσσερεις παρακάτω εγκαταστάσεις: Κολυμβητήριο, γήπεδο τένις, γήπεδο στίβου και γυμναστήριο. Το συμβούλιο θέλει να κατασκευάσει τις εγκαταστάσεις με γνώμονα τη μεγιστοποίηση της καθημερινής χρήσης τους από τους πολίτες, σεβόμενη το κόστος και τις απαιτήσεις χώρου για κάθε εγκατάσταση. Η αναμενόμενη καθημερινή χρήση, το κόστος κατασκευής καθώς και οι απαιτήσεις χώρου για την κατασκευή κάθε εγκατάστασης φαίνονται στον παρακάτω πίνακα:

Πίνακας 10: Πίνακας Παραδείγματος

Εγκατάσταση	Προβλεπόμενη Χρήση (άνθρωποι/ ημέρα.	Κόστος (Σε Ευρώ)	Απαιτήσεις Γής (Σε στρέμματα)
Κολυμβητήριο	300	35000	4
Γήπεδο Τένις	90	10000	2
Γήπεδο Στίβου	400	25000	7
Γυμναστήριο	150	90000	3

Ο προϋπολογισμός που διαθέτει το συμβούλιο είναι120000 Ευρώ και η έκταση που μπορεί να χρησιμοποιήσει είναι 12 στρέμματα. Λόγω του ότι το γήπεδο τένις και το κολυμβητήριο μπορούν να κατασκευαστούν στην ίδια περιοχή, μπορεί να κατασκευαστεί μόνο μία από τις δύο εγκαταστάσεις. Το συμβούλιο θα πρέπει να αποφασίσει για ποιες από τις παραπάνω εγκαταστάσεις θα πρέπει να δώσει άδεια για την κατασκευή τους με σκοπό τη μέγιστη καθημερινή τους χρήση από τους πολίτες.

Το μαθηματικό μοντέλο του προβλήματός μας είναι:

$$(\max) Z = 300x_1 + 90x_2 + 400x_3 + 150x_4$$

$$35000x_1 + 10000x_2 + 25000x_3 + 90000x_4 \leq 120000$$
$$4x_1 + 2x_2 + 7x_3 + 3x_4 \leq 12$$
$$x_1 + x_2 \leq 1$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

Όπου:

Z: Η αναμενόμενη καθημερινή χρήση των εγκαταστάσεων (άνθρωποι/ ημέρα)

x_1: Η κατασκευή κολυμβητηρίου

x_2: Η κατασκευή γηπέδου τένις

x_3: Η κατασκευή γηπέδου στίβου

x_4: Η κατασκευή γυμναστηρίου

Ο πρώτος περιορισμός του μοντέλου μας αναφέρεται στις δαπάνες που είναι διατεθειμένο το συμβούλιο να καταβάλλει ενώ ο δεύτερος στον περιορισμό χώρου. Ο τελευταίος περιορισμός δηλώνει πως τα αποτελέσματα του γραμμικού συστήματος θα πρέπει να είναι αυστηρά 0 και 1.

Με τον τρίτο περιορισμό, βλέπουμε ότι οι μεταβλητές x_1 και x_2 δεν μπορούν ταυτόχρονα να πάρουν την τιμή 1, δηλαδή δεν μπορούν να κατασκευαστούν κολυμβητήριο και γήπεδο τένις. Θα κατασκευαστεί ένα από τα δύο ή κανένα. Αυτό το είδος περιορισμού αναφέρεται και ως περιορισμός αποκλειστικότητας (mutually exclusive constrain).

Για την εφαρμογή της μεθόδου Branch and Bound για την επίλυση του προβλήματος θα χρειαστεί να εισάγουμε τους περιορισμούς:

$$x_1 \leq 1$$
$$x_2 \leq 1$$
$$x_3 \leq 1$$
$$x_4 \leq 1$$

Η μόνη διαφορά στην εφαρμογή της μεθόδου στο μοντέλο μας παρουσιάζεται στο βήμα 3 κατά τη διαδικασία της διακλάδωσης. Όταν επιλεχθεί η μη ακέραιη μεταβλητή x_j οι δύο νέοι περιορισμοί που θα εισαχθούν και θα αποτελούν και τους νέους κόμβους θα είναι οι: $x_j = 0$ και $x_j = 1$.

Μια άλλη μέθοδος επίλυσης των προβλημάτων 0-1 είναι η μέθοδος της έμμεσης απαρίθμησης (implicit enumeration). Κατά τη μέθοδο της έμμεσης

απαρίθμησης απορρίπτουμε εξαρχής τις μη εφικτές λύσεις και αξιολογούμε τις εναπομείναντες έως ότου βρούμε τη βέλτιστη.

Ολόκληρη η απαρίθμηση λύσεων του μοντέλου μας φαίνεται στον παρακάτω πίνακα:

Πίνακας 11: Πίνακας Λύσης

Λύση	x1	x2	x3	x4	Εφικτότητα	Z
1	0	0	0	0	Εφικτό	0
2	1	0	0	0	Εφικτό	300
3	0	1	0	0	Εφικτό	90
4	0	0	1	0	Εφικτό	400
5	0	0	0	1	Εφικτό	150
6	1	1	0	0	Μη Εφικτό	∞
7	1	0	1	0	Εφικτό	700
8	1	0	0	1	Μη Εφικτό	∞
9	0	1	1	0	Εφικτό	490
10	0	1	0	1	Εφικτό	240
11	0	0	1	1	Εφικτό	550
12	1	1	1	0	Μη Εφικτό	∞
13	1	0	1	1	Μη Εφικτό	∞
14	1	1	0	1	Μη Εφικτό	∞
15	0	1	1	1	Μη Εφικτό	∞
16	1	1	1	1	Μη Εφικτό	∞

Οι λύσεις 6, 12, 4 και 16 μπορούν να αποκλειστούν αμέσως επειδή παραβιάζουν τον περιορισμό $x_1+x_2 \leq 1$. Οι λύσεις 8, 13, 15 μπορούν να αποκλειστούν λόγω παραβίασης των δύο πρώτων περιορισμών. Αποκλείοντας και τη λύση 1 για ευνόητους λόγους απομένουν 8 πιθανές λύσεις. Εκτιμώντας την αντικειμενική συνάρτηση Z και για τις 8 καταλήγουμε στο ότι η βέλτιστη λύση του προβλήματός μας είναι η λύση 7 με τιμές: $x_1= 1$, $x_2= 0$, $x_3= 1$ και $x_4= 0$. Με βάση την διατύπωση του προβλήματος θα πρέπει

να κατασκευαστεί ένα κολυμβητήριο και ένα γήπεδο στίβου για να επιτύχουμε τη μεγαλύτερη καθημερινή χρήση η οποία θα είναι 700 άτομα ανά ημέρα.
Η διαδικασία του αποκλεισμού των μη εφικτών λύσεων και της εκτίμησης των εφικτών για την εύρεση της βέλτιστης λύσης αποτελεί τη βασική αρχή της μεθόδου της έμμεσης απαρίθμησης. Παρόλα αυτά η έμμεση απαρίθμηση χρησιμοποιείται ποιο συστηματικά αξιολογώντας τις λύσεις με τη βοήθεια δενδροειδών διαγραμμάτων παρόμοιων με αυτά που χρησιμοποιούνται στη μέθοδο Branch and Bound, παρά με τη σάρωση των εφικτών λύσεων από τον πίνακα όπως είδαμε στο παράδειγμά μας.

ΚΕΦΑΛΑΙΟ 4: Ακέραιος Προγραμματισμός και Βιοπληροφορική

4.1 Εισαγωγή

Στην παρούσα ενότητα παρουσιάζεται ένα μοντέλο βελτιστοποίησης ακεραίου γραμμικού προγραμματισμού με σκοπό την αντιμετώπιση του προβλήματος της ολικής στοίχισης κατά ζεύγη ακολουθιών. Το μοντέλο δουλεύει είτε τόσο για πρωτεϊνικές ακολουθίες όσο και για ακολουθίες νουκλεοτιδίων.

Είναι ανεξάρτητο από τη μέθοδο που θα χρησιμοποιηθεί για την επιβολή ποινών για τα κενά, καθώς και από την επιλογή του πίνακα αντικατάστασης.

Το συγκεκριμένο μοντέλο βελτιστοποίησης, όπως κάθε μοντέλο ακέραιου γραμμικού προγραμματισμού παρέχει μια ντετερμινιστική εγγύηση εύρεσης της βέλτιστης ολικής στοίχισης.

Η ελευθερία κινήσεων που μας παρέχει ο ακέραιος γραμμικός προγραμματισμός αποτελεί και το μεγάλο πλεονέκτημα της εφαρμογής της συγκεκριμένης μεθόδου για την επίλυση του παραπάνω προβλήματος. Για παράδειγμα, προσθέτοντας στο μοντέλο κατάλληλους περιορισμούς ακεραιότητας μπορούμε να παράγουμε μια λίστα με τις στοιχίσεις που βγάζουν το μεγαλύτερο score, πέρα από τη βέλτιστη στοίχιση. Μια τέτοια λίστα μπορεί να έχει εξαιρετικό βιολογικό ενδιαφέρον στο πρόβλημα εύρεσης κοινών προγόνων (S.R. McAllister, 2009).

4.2 Μαθηματικό Μοντέλο.

Έστω ότι έχουμε δύο πρωτεϊνικές ακολουθίες, την S_1 μήκους M και την S_2 μήκους N. Χωρίς βλάβη της γενικότητας υποθέτουμε ότι ισχύει M> N.
Χρησιμοποιούμε τους δείκτες i και j για να δηλώσουμε τις θέσεις των καταλοίπων στην αλληλουχία S_1 και S_2 αντίστοιχα.

$$i \in 1,2,\dots,M$$

$$j \in 1,2,\ldots,N$$

Εισάγουμε μια νέα δυαδική μεταβλητή N_{ij} η οποία μας δείχνει τη στοίχιση του καταλοίπου που βρίσκεται στη θέση i της ακολουθίας S_1 και του καταλοίπου με θέση j της ακολουθίας S_2. Η στοίχιση των καταλοίπων των δύο αλληλουχιών μας δίνει το βάρος Sij το οποίο προκύπτει από τον πίνακα αντικατάστασης που θα χρησιμοποιήσουμε.

Δημιουργώντας πλέγμα με κόμβους τις μεταβλητές N_{ij} όπου οι κόμβοι που θα είναι ενεργοί αποτελούν και τα σημεία στοίχισης των ακολουθιών, εισάγουμε μια νέα μεταβλητή τη $y_{ii',jj'}$ με την οποία συμβολίζουμε την ύπαρξη ακμής μεταξύ των γειτονικών κόμβων N_{ij} και $N_{i'j'}$. Για τον υπολογισμό των βαρών των ακμών χρησιμοποιείται η μεταβλητή $C_{ii',jj'}$ η οποία υπολογίζεται επίσης από τον πίνακα αντικατάστασης σε συνδυασμό με τη μέθοδο υπολογισμού των κενών που χρησιμοποιούμε. Για την καλύτερη κατανόηση των μεγεθών που προαναφέραμε, ας δούμε το παρακάτω παράδειγμα.

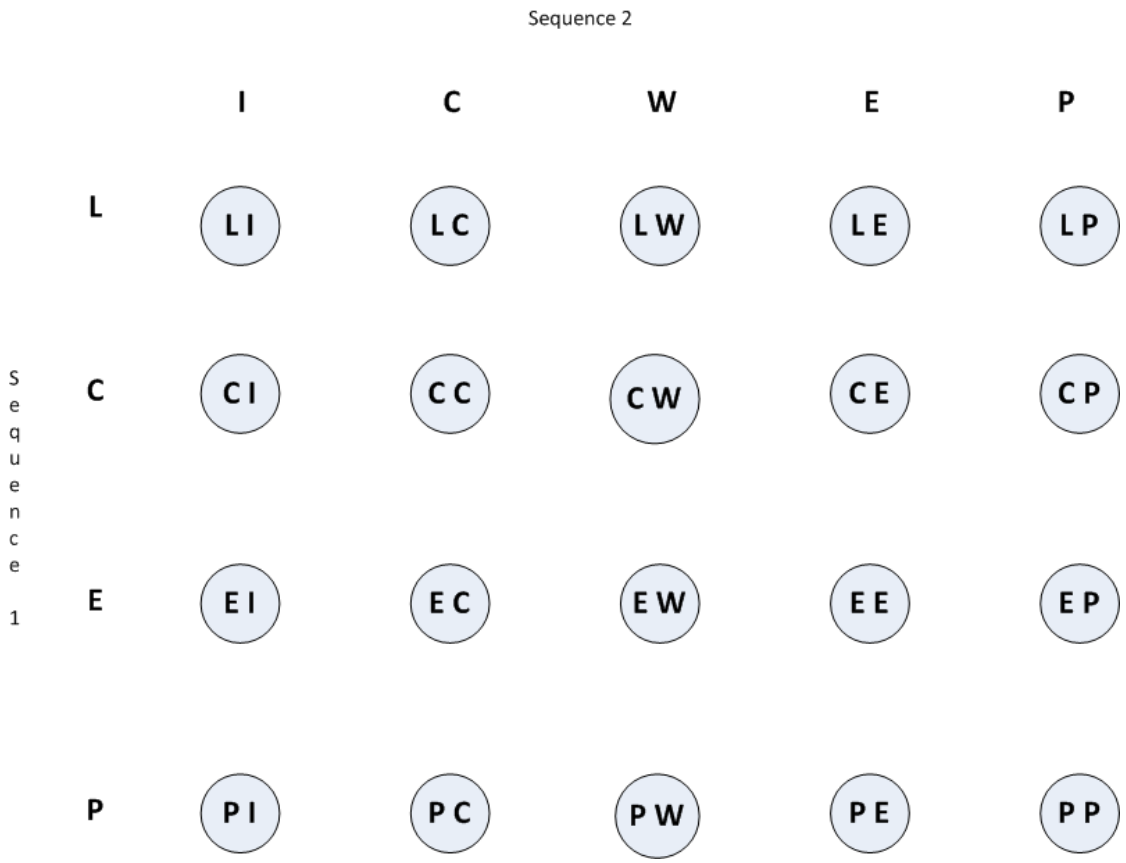
Παράδειγμα 1.6

Έστω ότι οι πρωτεϊνικές ακολουθίες που θα συγκρίνουμε είναι οι:

S_1: L C E P

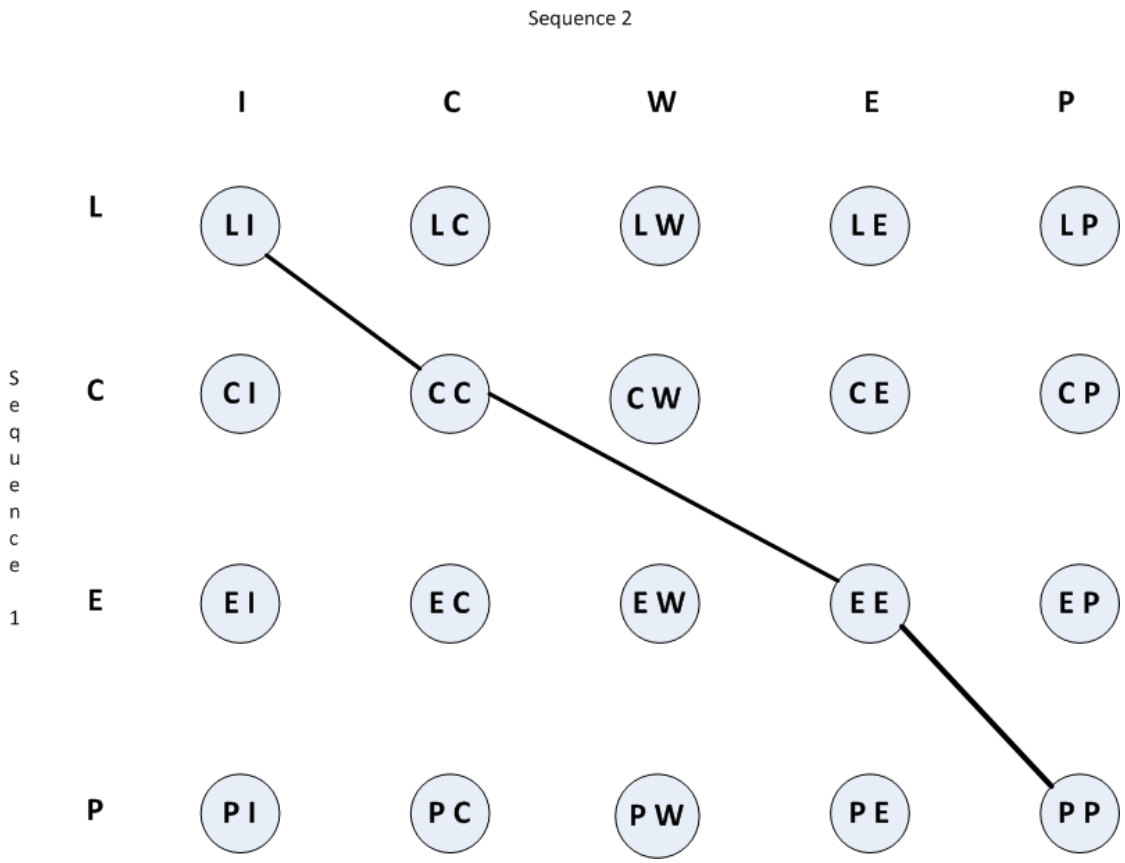
S_2: I C W E P

Δημιουργώντας το ανάλογο πλέγμα έχουμε:



Εικόνα 19: Αρχικό πλέγμα

Η βέλτιστη στοίχιση για τις δύο αλληλουχίες που θα προκύψει με βάση το μαθηματικό μας μοντέλο θα είναι το βέλτιστο μονοπάτι του πλέγματος.



Εικόνα 20: Μονοπάτι

Όπου οι κόμβοι $N_{1,1}$, $N_{2,2}$, $N_{3,4}$, $N_{4,5}$ είναι ενεργοί και έχουν την τιμή 1., καθώς και οι ακμές που σχηματίζουν το μονοπάτι $y_{1.1,2.2}$, $y_{2.2,3.4}$, $y_{3.4,4.5}$ αποτελούν τις ενεργές ακμές με τιμή 1.

Η εύρεση του βέλτιστου μονοπατιού θα πραγματοποιείται με τη μεγιστοποίηση της αντικειμενικής συνάρτησης που προκύπτει από το άθροισμα των γινομένων των ενεργών ακμών $y_{ii',jj'}$ και των αντίστοιχων βαρών τους $C_{ii',jj'}$.

$$\max \sum_{i} \sum_{i'>i} \sum_{j} \sum_{j'>j} y_{ii',jj'} C_{ii',jj'} \quad (1)$$

Στο σημείο αυτό, θα πρέπει να τονίσουμε δύο βασικές παραμέτρους για την αντικειμενική συνάρτηση.

Πρώτον, ο πίνακας αντικατάστασης που θα χρησιμοποιηθεί αποτελεί σημαντικό σημείο για την εύρεση της βέλτιστης στοίχισης. Ό πίνακας μας δίνει τιμές βασιζόμενος στη στοίχιση ενός ζεύγους αμινοξέων (ή αντίστοιχα ενός ζεύγους νουκλεοτιδίων). Εάν τα δύο αυτά κατάλοιπα ανήκουν στην ίδια οικογένεια (παρουσιάζουν ελάχιστες διαφορές σε επίπεδο μικρομορίων),

πρόκειται για μια συντηρητική στοίχιση όπου ο πίνακας μας δίνει υψηλή τιμή. Συνεπώς το ταίριασμα αμινοξέων με παρόμοιες ιδιότητες αυξάνει το score στην αντικειμενική συνάρτηση σε σχέση με το ταίριασμα λιγότερο όμοιων καταλοίπων.

Η δεύτερη παράμετρος που πρέπει να λάβουμε υπόψη μας είναι η μέθοδος υπολογισμού των ποινών που θα επιβληθούν στη χρήση κενών. Ανάλογα με το μοντέλο επηρεάζονται και οι τιμές της αντικειμενικής συνάρτησης. Το ποιο σύνηθες μοντέλο υπολογισμού ποινών είναι το affine gap penalty. Όπως προαναφέραμε, το συγκεκριμένο μοντέλο λειτουργεί με την επιβολή μιας μεγάλης σταθερής τιμής για το άνοιγμα του κενού και στη συνέχεια με την προσθήκη μιας μικρότερης τιμής για κάθε νέα κενή θέση. Για παράδειγμα για ένα κενό τριών χαρακτήρων θα επιβληθεί ποινή ανοίγματος για τον πρώτο χαρακτήρα και δύο μικρότερες τιμές για την προέκταση των κενών στις άλλες δύο θέσεις.

4.2.1. Περιορισμοί.

Απαραίτητη προϋπόθεση για την ορθή λειτουργία του μοντέλου είναι η εξασφάλιση της συνδεσιμότητας των κόμβων του πλέγματος. Σκοπός του μοντέλου όπως προαναφέραμε είναι η εύρεση του βέλτιστου μονοπατιού.

Οι μεταβλητές $y_{ii',jj'}$ έχουν οριστεί για την επικοινωνία των γειτονικών κόμβων. Έτσι ο κόμβος $N_{i'j'}$ όταν έχει ενεργοποιημένη μία εισερχόμενη ακμή, θα πρέπει να έχει ενεργοποιημένη και μία εξερχόμενη για να μπορεί να αποτελέσει κομμάτι του τελικού μονοπατιού.

$$\sum_{i<i'}\sum_{j<j'} y_{ii',jj'} - \sum_{i'<i''}\sum_{j'<j''} y_{i'i'',j'j''} = 0, \forall 1 < i' < M, 1 < j' < N \quad (2)$$

Με τον παραπάνω περιορισμό εξασφαλίζεται μια ισορροπία για κάθε εσωτερικό κόμβο του πλέγματος.

Κατά τη στοίχιση των δύο αλληλουχιών θα πρέπει να αποφευχθεί η στοίχιση του πρώτου καταλοίπου μιας εκ των δύο αλληλουχιών με ένα κενό, μιας και αυτό αποτελεί μια στοίχιση χωρίς φυσική σημασία. Ο περιορισμός που ακολουθεί φροντίζει να αποτραπεί ένα τέτοιο γεγονός.

$$\sum_{i'>i}\sum_{j}\sum_{j'>j} y_{i=1i',jj'} + \sum_{i}\sum_{i'>i}\sum_{j'>j} y_{ii',j=1j'} - \sum_{i'>1}\sum_{j'>1} y_{i=1i',j=1j'} = 1 \quad (3)$$

Εάν μια από τις δύο ακολουθίες τελειώνει με κενό, τότε θα πρέπει να αποφευχθεί η στοίχιση του τελευταίου καταλοίπου της άλλης ακολουθίας να στοιχηθεί με προηγούμενο κατάλοιπο μιας και αυτό δεν θα έχει κάποια φυσική σημασία. Το γεγονός αυτό επιτυγχάνεται δημιουργώντας έναν περιορισμό στον οποίο θα εμπλέκονται οι τελευταίοι όροι και των δύο ακολουθιών.

$$\sum_{i<M}\sum_{j}\sum_{j'>j} y_{ii'=M,jj'} + \sum_{i}\sum_{i'>i}\sum_{j<N} y_{ii',jj'=N} - \sum_{i<M}\sum_{j<N} y_{ii'=M,jj'=N} = 1 \quad (4)$$

Στην περίπτωση στοίχισης μεγάλων αλληλουχιών, για παράδειγμα ολόκληρων γονιδιομάτων ή μεγάλων πρωτεϊνικών αλυσίδων θα ήταν πιο αποτελεσματικό να οριστεί ένα μέγιστο μέγεθος στοίχισης K. Αυτό το μέγιστο μέγεθος στοίχισης που θα οριστεί, διαφέρει από περίπτωση σε περίπτωση. Εξαρτάται καθαρά από τη φύση του προβλήματος. Μια καλή διαδεδομένη πρακτική για τον υπολογισμό του K είναι να αποτελεί το 25-50% του μήκους της μικρότερης αλληλουχίας. Αυτή η επιλογή θα μπορούσε να λειτουργήσει ως ένα χαλαρό άνω όριο, επιτρέποντας εισαγωγή ενός αρκετά υψηλού ποσοστού κενών κατά τη στοίχιση.

Η βιολογική υφή των αλληλουχιών μας επιβάλει, κάποια κατάλοιπα για τα οποία φαινομενικά η στοίχισή τους δεν έχει κάποιο νόημα, λόγω της ιδιαίτερης λειτουργίας που μπορεί να παίζουν στην δομή μιας πρωτεΐνης να στοιχίζονται υποχρεωτικά. Αυτά τα κατάλοιπα μπορεί να αποτελούν μια δισουλφιδική γέφυρα για τη σταθερότητα της δομής μιας πρωτεΐνης, ή ακόμα κα ένα χαρακτηριστικό της διαφύλαξης μιας λειτουργίας της.

Αυτό μπορεί να αποτυπωθεί στο μαθηματικό μοντέλο συνδέοντας με τις παρακάτω εξισώσεις τις μεταβλητές N_{ij} και $y_{ii',jj'}$.

$$\sum_{i<i',j<j'} y_{ii',jj'} = N_{i'j'}, \forall i'>1, j'>1 \quad (5)$$

$$\sum_{i'>i,j'>j} y_{ii',jj'} = N_{i'j'}, \forall i=1, j=1 \quad (6)$$

Ο παραπάνω λειτουργικός περιορισμός που αναφέραμε χρησιμοποιώντας μόνο τη μεταβλητή N_{ij} μπορεί να γραφεί:

$$\sum N_{i^* j} = 1 \qquad (7)$$

Όπου η θέση i* είναι η θέση του καταλοίπου στην αλληλουχία 1 που αποτελεί στοιχείο «κλειδί» για την λειτουργία της πρωτεΐνης.

Η λειτουργικότητα του παρόντος μαθηματικού μοντέλου μπορεί να επεκταθεί εισάγοντας περιορισμούς αποκοπής. Μετά την επιτυχή επίλυση του μοντέλου η βέλτιστη λύση μπορεί να εξαιρεθεί από την εφικτή περιοχή με σκοπό να μπορούμε να πάρουμε την αμέσως επόμενη βέλτιστη λύση, και με αυτό τον τρόπο να καταφέρουμε να φτιάξουμε έναν κατάλογο των καλύτερων στοιχίσεων των δύο αλληλουχιών.

$$\sum_{(ii'\,jj')\in A} y_{ii',jj'} - \sum_{(ii'\,jj')\in I} y_{ii',jj'} \le card(A) - 1 \qquad (8)$$

Στην παραπάνω εξίσωση το A αποτελεί το σύνολο των ενεργών μεταβλητών, όπου όλες οι μεταβλητές υποθέτουμε ότι έχουν την τιμή 1. Με I συμβολίζουμε το σύνολο των ανενεργών μεταβλητών και το card(A) εκφράζει το πλήθος των μελών του συνόλου A

4.2.2. Επιλογή Solver.

Το μοντέλο ακεραίου γραμμικού προγραμματισμού που περιγράψαμε μπορεί να εφαρμοστή για κάθε πρόβλημα στοίχισης βιολογικών αλληλουχιών με ένα λογικό μήκος. Αρχικά θα πρέπει να επιλέξουμε τον πίνακα αντικατάστασης που θα χρησιμοποιήσουμε, καθώς και τη μέθοδο υπολογισμού των κενών. Στη συνέχεια, η επόμενη επιλογή έχει να κάνει με τον solver που θα χρησιμοποιήσουμε για να τρέξουμε το μοντέλο. Μια αξιόπιστη λύση solver αποτελεί ο CPLEX, ένας εμπορικός solver της εταιρίας ILOG.

Η ILOG CPLEX Optimization Suite παρέχει απεικόνιση δεδομένων υψηλής απόδοσης για τις διεπιφάνειες επικοινωνίας με το χρήστη (user interfaces) – ακέραιες, γραμμικές και με περιορισμούς επιλύσεις για τη βελτιστοποίηση πόρων και τις εφαρμογές οργάνωσης, λογιστικών και σχεδιασμού, δυναμικά συστήματα κανόνων για ευφυείς πράκτορες (intelligent

agents) και έλεγχο πραγματικού χρόνου στη ροή δεδομένων, καθώς και στοιχεία για modules ενοποίησης με πραγματικού χρόνου και σχετικούς πόρους δεδομένων.

Το ILOG CPLEX χρησιμοποιεί τη μέθοδο Branch and Bound για την επίλυση προβλημάτων μικτού ακέραιου προγραμματισμού. Στη μέθοδο Branch and Bound, λύνεται μια σειρά από γραμμικά προγράμματα, δημιουργώντας ένα δέντρο Branch and Bound. Το μονοπάτι που ακολουθεί το CPLEX σ' αυτό το δέντρο μπορεί να προσδιοριστεί από έναν αριθμό εισόδων από τον χρήστη.

Για παράδειγμα, σε κάθε κόμβο στο δέντρο της Branch and Bound, το CPLEX μπορεί είτε να διερευνήσει βαθύτερα είτε να επιστρέψει προς τα πίσω. Ο χρήστης μπορεί να καθορίσει τα χαρακτηριστικά της παραμέτρου της αναδρομής ή να την αφήσει ως έχει (default τιμή). Και όταν το CPLEX κάνει ένα βήμα πίσω, υπάρχει τυπικά ένας μεγάλος αριθμός ανεξερεύνητων κόμβων, από τους οποίους μπορεί να επιλέξει. Η παράμετρος επιλογής κόμβου χρησιμοποιείται για να θέσει τον κανόνα επιλογής του επόμενου κόμβου της διαδικασίας κατά την αναδρομή.

Ο χρήσης έχει επίσης την ευκαιρία να επιλέξει το είδος του αλγορίθμου που θα χρησιμοποιηθεί για την επίλυση των γραμμικών προβλημάτων που δημιουργούνται στο δέντρο αναζήτησης της Branch and Bound. Η τυπική στρατηγική είναι να χρησιμοποιηθεί μέθοδος dual-simplex, αλλά κάποια προβλήματα λύνονται πιο αποτελεσματικά με χρήση primal simplex ή φραγής.

Η κοπή επιπέδων (cutting planes) μπορεί να είναι πολύ αποτελεσματικό για τη βελτίωση της απόδοσης του CPLEX κατά την επίλυση προβλημάτων μεικτού ακέραιου προγραμματισμού. Ωστόσο, ορισμένα μόνο προβλήματα έχουν τη θεμελιώδη δομή για να χρησιμοποιήσουν κοπή επιπέδων, και το CPLEX ερευνά αυτόματα το κάθε πρόβλημα για να δει αν υπάρχει αυτή η θεμελιώδης δομή. Αν ο χρήστης έπειτα καθορίσει ότι το πρόβλημα δεν θα βοηθηθεί από τη χρήση αυτής της δομής, αυτή η επιλογή μπορεί να απενεργοποιηθεί. Από την άλλη πλευρά, αν αυτή η εκτίμηση είναι λανθασμένη, ο χρήστης μπορεί να θέσει τις παραμέτρους που επιτρέπουν στο CPLEX να ενεργοποιήσει την κοπή.

Η πιο πρόσφατη καινοτομία του προγραμματισμού περιορισμών είναι μια πολύ αποτελεσματική τεχνολογία που χρησιμοποιεί μείωση πεδίου και διάδοση περιορισμών για την ικανοποιητική επίλυση προβλημάτων που είναι πολύ συνδυαστικά και με πολύ λογικό περιεχόμενο. Η μεθοδολογία προγραμματισμού περιορισμών επιτρέπει τη φυσική έκφραση πολύπλοκων σχέσεων, περιλαμβάνοντας και λογικές εκφράσεις.

Ο προγραμματισμός περιορισμών χρησιμοποιεί την πληροφορία που εμπεριέχεται στο πρόβλημα για να ‘κλαδέψει’ το εύρος αναζήτησης, ώστε να αναγνωρίσει ταχύτερα τις εφικτές λύσεις. Τα πεδία των παραμέτρων ανανεώνονται συνεχώς κατά τη διάρκεια της αναζήτησης της λύσης, παρέχοντας πολύ χρήσιμες πληροφορίες για τη βελτίωση της στρατηγικής αναζήτησης. Βεβαίως, ο χρήστης μπορεί να κατευθύνει τη διαδικασία αναζήτησης με βάση τη δική του γνώση επί του προβλήματος.

ΚΕΦΑΛΑΙΟ 5: Συμπεράσματα - Μελλοντική Εργασία.

Στην παρούσα εργασία περιγράψαμε το πρόβλημα της ολικής στοίχισης αλληλουχιών κατά ζεύγη, ως ένα από τα θεμελιώδη προβλήματα της βιοπληροφορικής. Παρουσιάσαμε όλες τις προαπαιτούμενες γνώσεις βιολογικού καθώς και υπολογιστικού χαρακτήρα που χρειάζεται ο αναγνώστης για να κατανοήσει το συγκεκριμένο πρόβλημα. Δώσαμε μια αναλυτική παρουσίαση των τρόπων επίλυσής του, καθώς και ένα προτεινόμενο μοντέλο γραμμικού ακεραίου προγραμματισμού που αποτελεί μια διαφορετική προσέγγιση του θέματος. Η μοντελοποίηση αυτή προσομοιώνει την τεχνική του δυναμικού προγραμματισμού με μια διαφορετική προσέγγιση στον τρόπο βαθμολόγησης των στοιχίσεων. Το πλεονέκτημα της χρήσης ακέραιου γραμμικού προγραμματισμού για την επίλυση του συγκεκριμένου προβλήματος βρίσκεται στην μεγάλη ευελιξία που παρέχει η συγκεκριμένη μαθηματική μέθοδος με τη χρήση των κατάλληλων περιορισμών. Μας δίνεται η δυνατότητα πέρα από τη βέλτιστη στοίχιση που εγγυάται το μοντέλο να πάρουμε και έναν πίνακα με τις εναλλακτικές στοιχίσεις που αποδίδουν το μεγαλύτερο score, καθώς και να διατηρήσουμε στοιχήσεις μεταξύ καταλοίπων με μεγάλη βιολογική σημασία.

Μια πιθανή βελτίωση του μοντέλου μπορεί να συνδυάσει μια πιο αποτελεσματική προσέγγιση δυναμικού προγραμματισμού με μαθηματική διατύπωση που θα είναι αποτελεσματικότερη στην διαχείριση του πίνακα των διαφορετικών στοιχίσεων με την προσθήκη περισσότερων περιορισμών αποκοπής και χωρίς να περιλαμβάνει την υποχρεωτική κράτηση καταλοίπων μεγάλης βιολογικής σημασίας. Επίσης, με την κατάλληλη επέκτασή του θα μπορούσε να διαχειριστεί και άλλα παρόμοια προβλήματα της βιοπληροφορικής όπως η πολλαπλή στοίχιση αλληλουχιών ή η αναγνώριση μοτίβων.

ΚΕΦΑΛΑΙΟ 6: Βιβλιογραφία

1) Arthur M. Lesk Introduction to Bioinformatics, University of Cambridge, 2002.

2) Bryan Bergeron, Bioinformatics Computing, Prentice Hall PTR, November 19, 2002.

3) Baxevanis, A.D., Francis Ouellette, B.F., Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins, Second Edition, Wiley, 2001.

4) Carlos E. Ferreira, On Combinatorial Optimization Problems Arising in Computer Systems Design, November 1997.

5) Christos Papadimitriou, Ken Steiglitz, Combinatorial optimization: algorithms and complexity Prentice-Hall, 1982.

6) Christos A. Ouzounis1, Alfonso Valencia, Early bioinformatics: the birth of a discipline- a personal view, 2003.

7) Collecting, Comparing, and Computing Sequences: The Making of Margaret O. Dayhoff's Atlas of Protein Sequence and Structure, 1954–1965.

8) Dan Gusfield, Algorithms on Strings, Threes and Sequences- computer science and computational biology, Cambridge University Press, 1997

9) Dayhoff et al., A new method for modeling protein evolution, 1978.

10) Doolittle, Similar amino acid sequences: chance or common ancestry, Science, 9 October 1981: 149-159.

11) European Bioinformatics Institute (EBI) www.ebi.ac.uk

12) Glasgow, J., Jurisica, I., and Rost, BAI and Bioinformatics, AI Magazine, 25(1): 7-8., 2004.

13) H. Pearson Genetics: What is gene? Nature 441, 398-401, 24 May 2006.

14) Helen M. Berman1,2,*, John Westbrook1,2, Zukang Feng1,2, Gary Gilliland1,3, T. N. Bhat1,3, Helge Weissig1,4, Ilya N. Shindyalov4 and Philip E. Bourne1,4,5,6 The Protein Data Bank, September 20, 1999.

15) Henikoff and Henikoff, Amino acid substitution matrices from protein blocks,1992.

16) Hieter, et al., Functional Genomics: It's All How You Read It, Science, 24 October 1997.

17) Hunter L., Molecular Biology for Computer Scientists, Artificial Intelligence and Molecular Biology, AAAI Press, 1993.

18) Hunter, L., Life and Its Molecules: A Brief Introduction, AI Magazine, 25(1), 9-22, 2004.

19) ILOG CPLEX 10 User's Manual pdf, ILOG, 2006.

20) Lipman, D. and Pearson, W., Improved Tools for Biological Sequence Comparison In Proceedings of the National Academy of Sciences, 85: 2444-2448, 1988.

21) Mark B. Gerstein, Can Bruce, Joel S. Rozowsky, et al., What is a gene, post-encode? History and updated definition, Genome Res. 2007 17: 669-681.

22) N.M. Luscombe, D. Greenbaum and M. Gernstein, What is Bioinformatics? A proposed Definition and Overview of the field, 2001.

23) Neil. C. Jones and Pavel. A. Pevzner, An Introduction to Bioinformatics Algorithms, MIT press, 2004.

24) Paul G. Higgs,Teresa K, Bioinformatics and molecular evolution, 2005.

25) Pennisi, DNA Study Forces Rethink of What It Means to Be a Gene, Science, 15 June 2007: 1556-1557.

26) Peter Y. Chou and Gerald D. Fasman, Prediction of Protein Conformation Biochemistry, Vol. 13, 1974.

27) R.Staden, Sequence data handling by computer Nucl. Acids Res. 4(11): 4037-4052.,1977

28) R.M. Steinman and C.L. Moberg, A triple tribute to the experiment that transformed biology. J. Exp. Med. 179:379–384.

29) Rost and Sander, structures.pdf, 1993.

30) Robert J. Vanderbei, KAP, 2001.

31) Richard Simon, Michael D. Radmacher, Kevin Dobbin and Lisa M. McShane, Pitfalls in the Use of DNA Microarray Data for Diagnostic and Prognostic Classification, October 11 2002.

32) S.R. McAllister a , R. Rajgaria a & C.A. Floudas, A path selection approach to global pairwise sequence alignment using integer linear optimization, Princeton University, 27 Oct 2009.

33) Saul B. Needleman, Christian D. Wunsch, A general method applicable to the search for similarities in the amino acid sequence of two proteins, 1970.

34) Schuler, G.D., Sequence Alignment and Database Searching, In A.D. Baxevanis and B.F.F. Ouellette (Eds.), Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins, 145-171, Wiley-Liss, Inc, 1998.

35) Smith Waterman Identification of Common Molecular Subsequences Reprinted from J. Mol. Biol. (1981) 147, 195-197, 1981.

36) Stephen F. Altschu, P Warren Gish , Webb Miller, Eugene W. Myers, David J. Lipman, Basic Local Alignment Search Tool, 1990.

37) T.L., A. S. BLAST and PSI-BLAST: a new generation of protein database search programs-Nucleic Acids Research . Oxford University Press, 1997.

38) Thomas Lenganer, bioinformatics- from Genomes to Drugs volume 1 & 2, willey-VCH, Weinheiw, 2002.

39) Velculescu, V.E., Zhang, L. Vogelstein, B., and Kinzler, K.W., Serial Analysis of Gene Expression, Science, 270 (5235), 484-487, 1995.

40) Wiliam R. Pearson, David J. Lipman, Improved tools for biological sequence comparison, 1988.

41) Woese, C.R., Kandler, O., and Wheelis, M.L., Towards a Natural System of Organisms: Proposal for the Domains Archaea, Bacteria, and Eucarya. Proceedings of the National Academy of Sciences, 87, 4576-4579, 1990.

42) Α. Περδικούρη,Α. Τσακαλίδης, Εισαγωγή στη Βιοπληροφορική, πανεπιστημιακές σημειώσεις, Μάρτιος 2004.

43) Β. Μαρμαράς, Μ. Λαμπροπούλου Μαρμαρά. Βιολογία κυττάρου, Μοριακή προσέγγιση, εκδόσεις Typorama, Σεπτέμβριος 2000.

44) Β. Κώστογλου – Επιχειρησιακή Έρευνα, Εκδόσεις Τζιόλα, Θεσσαλονίκη 2003.

45) Β. Προμπόνας, Στοιχεία Αρχιτεκτονικής της τρισδιάστατης δομής πρωτεϊνών. Σύγκριση και κατηγοριοποίηση πρωτεϊνικών δομών, Πανεπιστημιακές σημειώσεις, 2006.

46) Θ.Α. Παταργιάς, Κ. Κομητοπούλου, Σ. Κουγιανού, Εισαγωγή στη Βιολογία, Πανεπιστήμιο Αθηνών, 1996.

47) Κουγιανού, Π. Κ., Εισαγωγή στη Βιολογία , Αθήνα Φαρμακευτικό τμήμα 1996.

48) Μ. Μποναζουντας (ΕΜΠ), Πανεπιστημιακές σημειώσεις, 2001.

49) Παπαρρίζος Κωνσταντίνος- Γραμμικός Προγραμματισμός, Εκδόσεις Ζυγός, 2009.

50) Χ. Ι. Σχοινάς - Επιχειρησιακή Έρευνα, Πανεπιστημιακές Παραδόσεις, Ξάνθη, 2007.