**BIOINFORMATIC DATABASES AND DATA MINING**

**ΣΧΟΛΗ: ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ**

**ΟΝΟΜΑΤΕΠΩΝΥΜΑ (ΑΜ): ΑΝΤΩΝΙΟΣ ΠΑΠΑΓΕΩΡΓΙΟΥ (1066591), ΑΝΑΣΤΑΣΙΟΣ ΚΟΝΤΟΥΛΗΣ(1066489)**

**ΕΤΟΣ: 4Ο**

Περιεχόμενα

[1. Εισαγωγή 4](#_Toc107171894)

[**1.1. Γενικά** 4](#_Toc107171895)

[**1.2. Προβλήματα** 4](#_Toc107171896)

[**1.2.1. Πρόβλημα αντιστοίχισης προτύπων** 4](#_Toc107171897)

[**1.2.2. Πρόβλημα χώρου** 5](#_Toc107171898)

[2. Θεωρία 6](#_Toc107171899)

[**2.1. Βιολογία** 6](#_Toc107171900)

[**2.1.1. DNA** 6](#_Toc107171901)

[**2.1.2. Κεντρικό δόγμα μοριακής βιολογίας** 6](#_Toc107171902)

[**2.1.3. Αντιγραφή** 6](#_Toc107171903)

[**2.2. Βιοπληροφορική** 7](#_Toc107171904)

[**2.2.1. Εφαρμογές** 7](#_Toc107171905)

[**2.2.2. Τομείς έρευνας** 7](#_Toc107171906)

[**2.3. Εξόρυξη δεδομένων** 10](#_Toc107171907)

[**2.3.1. Εξόρυξη Δεδομένων στη Βιοπληροφορική** 10](#_Toc107171908)

[**2.3.2. Προσεγγίσεις** 10](#_Toc107171909)

[3. Multiple Longest Common Subsequence (MLCS) 12](#_Toc107171910)

[**3.1. Εισαγωγή** 12](#_Toc107171911)

[**3.2. Βασικές αρχές μεθόδου κυρίαρχου σημείου** 12](#_Toc107171912)

[4. Τεχνολογίες αλληλουχίας 14](#_Toc107171913)

[**4.1. Εισαγωγή** 14](#_Toc107171914)

[**4.2. High-throughput sequencing (HTS)** 15](#_Toc107171915)

[5. Approximate string matching 16](#_Toc107171916)

[**5.1. Στοίχιση ακολουθίας** 16](#_Toc107171917)

[**5.2. Εισαγωγή στο ASP** 16](#_Toc107171918)

[**5.3. Κλασικές δομές ευρετηρίου** 16](#_Toc107171919)

[**5.3.1. Hashes based indexes** 16](#_Toc107171920)

[**5.3.2. Suffix tree** 16](#_Toc107171921)

[**5.3.3. Suffix array** 17](#_Toc107171922)

[6. Exact Pattern Matching 18](#_Toc107171923)

[**6.1. First-Last Pattern Matching Algorithm** 18](#_Toc107171924)

[**6.1.1. Φάση προεπεξεργασίας** 18](#_Toc107171925)

[**6.1.2. Φάση αντιστοιχίας** 19](#_Toc107171926)

[**6.1.3. Παράδειγμα** 20](#_Toc107171927)

[**6.2. Processor-Aware Pattern Matching Algorithm** 20](#_Toc107171928)

[**6.3. Least Frequency Pattern Matching Algorithm** 21](#_Toc107171929)

[7. Συμπεράσματα 27](#_Toc107171930)

[**7.1. Σύγκριση** 27](#_Toc107171931)

[**7.1.1. Χρόνος φάσης προεπεξεργασίας** 27](#_Toc107171932)

[**7.1.2. Χρόνος φάσης αντιστοιχίας** 28](#_Toc107171933)

[**7.1.3. Συνολικός χρόνος** 29](#_Toc107171934)

[**7.2. Συμπεράσματα** 29](#_Toc107171935)

[8. Βιβλιογραφία 30](#_Toc107171936)

# **1. Εισαγωγή**

## **1.1. Γενικά**

Τα τελευταία χρόνια, οι ραγδαίες εξελίξεις στη γονιδιωματική και πρωτεομική έχουν δημιουργήσει μεγάλο αριθμό βιολογικών δεδομένων, κάτι που αποτελεί κίνητρο για να επιταχυνθεί η ανακάλυψη λύσεων σε πολλούς τομείς της υπολογιστικής βιοπληροφορικής. Η εξαγωγή συμπερασμάτων από αυτά τα δεδομένα απαιτεί πολύπλοκες υπολογιστικές αναλύσεις. Στη συνέχεια όμως, η αντιστοίχιση προτύπων είναι μια πολύ πρακτική λειτουργία. Για παράδειγμα, επιτρέπει στους χρήστες να βρίσκουν τις θέσεις συγκεκριμένων υποακολουθιών DNA σε μια βάση δεδομένων ή μια ακολουθία DNA. Επιπλέον, σε αυτές τις βιολογικές βάσεις δεδομένων, ορισμένα πρότυπα ενημερώνονται με την πάροδο του χρόνου. Έτσι, για να πραγματοποιηθούν ταχύτερες αναζητήσεις, απαιτούνται αλγόριθμοι αντιστοίχισης μοτίβων υψηλής ταχύτητας, οι οποίοι αυξάνουν την απόδοση χρησιμοποιώντας επεξεργασία κειμένου και ακόμα αναζητώντας τη λιγότερο συχνή λέξη του προτύπου στην ακολουθία.

Έτσι η βιοπληροφορική, ή αλλιώς η υπολογιστική βιολογία, είναι η επιστήμη της ερμηνείας βιολογικών δεδομένων χρησιμοποιώντας την τεχνολογία της πληροφορίας και την επιστήμη των υπολογιστών. Όσο αυξάνονται οι ποσότητες γονιδιωματικών, πρωτεομικών και άλλων δεδομένων που παράγουμε και ενσωματώνουμε, τόσο αυξάνεται και η ανάγκη μελέτης αυτού του πεδίου έρευνας [[4]](#four).

Ένας ιδιαίτερος ενεργός τομέας έρευνας στη βιοπληροφορική είναι η εφαρμογή και ανάπτυξη τεχνικών εξόρυξης δεδομένων για την επίλυση βιολογικών προβλημάτων. Η ανάλυση μεγάλων συνόλων βιολογικών δεδομένων απαιτεί την κατανόηση των δεδομένων. Παραδείγματα αυτού του τύπου ανάλυσης αποτελούν η πρόβλεψη δομής πρωτεϊνών, η ταξινόμηση γονιδίου, η ταξινόμηση καρκίνου με βάση δεδομένα μικροσυστοιχιών, η ομαδοποίηση δεδομένων γονιδιακής έκφρασης, η στατιστική μοντελοποίηση της αλληλεπίδρασης πρωτεΐνης με πρωτεΐνη και πολλά άλλα [[4]](#four).

## **1.2. Προβλήματα**

### **1.2.1. Πρόβλημα αντιστοίχισης προτύπων**

Το πρόβλημα αντιστοίχισης προτύπων προκύπτει στα διαφορετικά πεδία της υπολογιστικής βιοπληροφορικής, τα οποία περιλαμβάνουν τη βασική αναζήτηση τοπικής ευθυγράμμισης, την ανακάλυψη βιοδεικτών, την ευθυγράμμιση αλληλουχίας, την πρωτεογενωμική χαρτογράφηση και την ανίχνευση ομόλογων σειρών. Σε αυτούς τους κλάδους, υπάρχει ανάγκη να αναγνωριστούν οι θέσεις πολλαπλών προτύπων, συμπεριλαμβανομένων εκείνων των αμινοξέων και των νουκλεοτιδίων σε βάσεις δεδομένων. Στη βιοτεχνολογία, την εγκληματολογία, την ιατρική και τη γεωργία, η γνώση της ανάλυσης γονιδίων και των αλληλουχιών DNA μπορεί να εφαρμοστεί κατά τη διερεύνηση πιθανών διαγνώσεων ασθενειών ή ανωμαλιών. Η σύγκριση ενός συγκεκριμένου γονιδίου με παρόμοια γονίδια του ίδιου ή διαφορετικών οργανισμών και η πρόβλεψη της λειτουργίας του μπορεί επίσης να χρησιμοποιήσει ανάλυση αλληλουχίας DNA. Η λειτουργικότητα μιας πρόσφατα ανακαλυφθείσας αλληλουχίας DNA μπορεί ακόμα να προκαθοριστεί διερευνώντας την ομοιότητά της με γνωστές αλληλουχίες DNA, μία προσέγγιση που έχει χρησιμοποιηθεί σε διάφορες ερευνητικές μελέτες και ιατρικές εφαρμογές [[2]](#two).

### **1.2.2. Πρόβλημα χώρου**

Οι προκλήσεις της ανάλυσης δεδομένων που προκαλούνται από τεχνολογίες υψηλής απόδοσης έχουν επίσης επηρεάσει τον σχεδιασμό νέων δομών δεδομένων για πρωτογενή και δευτερογενή αποθήκευση. Πολλές κλασικές δομές ευρετηρίου για “Approximate String Matching” έχουν αποδειχθεί ότι δεν είναι κατάλληλες για την κλίμακα των διαχειριζόμενων δεδομένων και έχουν γίνει αρκετές προτάσεις για την υπέρβαση του τείχους μνήμης και την εξοικονόμηση αποθήκευσης δίσκου. Μεταξύ άλλων συνοπτικών ευρετηρίων, το “FM-Index” , που είναι βασισμένο στον ίδιο μετασχηματισμό “Burrows Wheeler” που χρησιμοποιείται για τη συμπίεση δεδομένων, είναι ένας από τους γνωστότερους ευθυγραμμιστές ακολουθίας. Παρέχοντας ένα ευρετήριο πλήρους κειμένου σε έναν πολύ μικρό χώρο, αυτή η δομή ευρετηρίου επέτρεψε στους αλγόριθμους να λειτουργούν στη μνήμη RAM με γονιδιώματα ανθρώπινου μεγέθους [[1]](#one).

Τα τελευταία χρόνια όμως, το μέγεθος των βιολογικών δεδομένων έχει αυξηθεί σημαντικά και ωστόσο αυτοί οι μεγάλοι όγκοι δεδομένων πρέπει να αναλυθούν σε εύλογο χρονικό διάστημα. Αυτό το ζήτημα συναντάται στη μοριακή βιολογία επειδή αλληλουχίες αμινοξέων ή νουκλεοτιδίων συχνά εφαρμόζονται σε κατά προσέγγιση βιολογικά μόρια. Ένα άλλο παράδειγμα είναι οι βασικές πληροφορίες των ειδών που διατηρούνται από αλληλουχίες DNA και η πρόσβαση σε αυτές μέσω αντιστοίχισης προτύπων. Ακόμα, σε μια αλληλουχία DNA, ο καθορισμός πιθανών ανωμαλιών ή σφαλμάτων συχνά απαιτεί ανάλυση. Επιπλέον, η αντιστοίχιση προτύπων είναι κατάλληλη σε πεδία όπως η φυλογενετική και η εξελικτική βιολογία, όπου συγκεκριμένες υποακολουθίες DNA εξάγονται από τα γονιδιωματικά δεδομένα των διαφορετικών ειδών των οργανισμών για σκοπούς κατανόησης της συγγένειας, της καταγωγής και της προέλευσής τους. Επομένως, σε αυτό το πλαίσιο, ένας αλγόριθμος αντιστοίχισης προτύπων πρέπει να μπορεί να αναζητά σε βάσεις δεδομένων που εκτείνονται από “gigabyte” έως “terabyte” ή περισσότερα και σε ολόκληρα γονιδιώματα με 3 δισεκατομμύρια ζεύγη βάσεων. Από την άλλη πλευρά, οι αλληλουχίες του DNA είναι πολύ μεγάλες, άρα ο χρόνος που καταναλώνεται για την αντιστοίχιση με το πρότυπο θεωρείται ως η πιο κρίσιμη μέτρηση [[2]](#two).

# **2. Θεωρία**

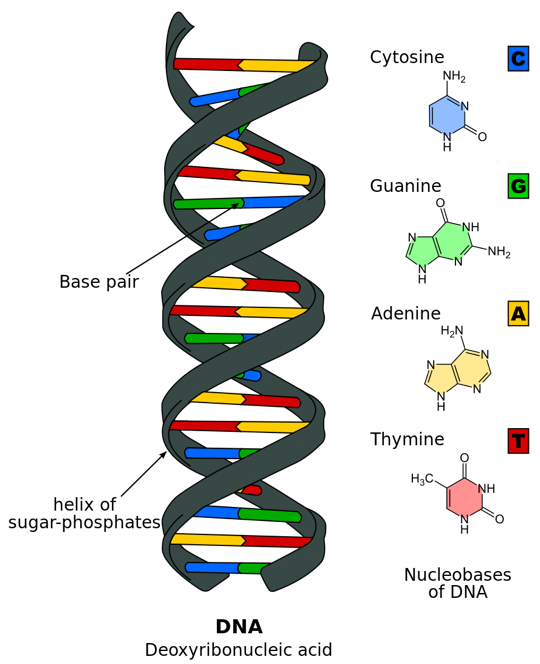
## **2.1. Βιολογία**

### **2.1.1. DNA**

Τα μόρια DNA αποτελούνται από δύο κλώνους νουκλεοτιδίων που συνδέονται μεταξύ τους για να σχηματίσουν μια διπλή έλικα, τη δευτερογενή δομή του DNA. Οι αλληλουχίες νουκλεοτιδίων μεταξύ των δύο κλώνων είναι συμπληρωματικές, με κάθε Α σε έναν κλώνο να συνδέεται με ένα Τ στον άλλο και αντίστροφα, και κάθε C σε έναν κλώνο να συνδέεται με ένα G στον άλλο και αντίστροφα. Αυτό αποτελεί έναν σημαντικό μηχανισμό πλεονασμού του DNA [[1]](#one).

### **2.1.2. Κεντρικό δόγμα μοριακής βιολογίας**

Ορισμένες υποακολουθίες μέσα σε ένα χρωμόσωμα, γνωστές ως γονίδια, κωδικοποιούν τις πρωτεΐνες. Σε γενικές γραμμές, υπάρχουν μοριακοί μηχανισμοί που διαβάζουν αυτά τα γονίδια και τα ερμηνεύουν ως οδηγίες για τη δημιουργία πρωτεϊνών από αμινοξέα. Τελικά, οι πρωτεΐνες αποτελούν τα βασικά δομικά στοιχεία της ζωής, επιτελώντας έναν τεράστιο αριθμό λειτουργιών στους ζωντανούς οργανισμούς. Για παράδειγμα, απαιτούνται για τη δομή, τη λειτουργία και τη ρύθμιση όλων των ιστών και οργάνων του σώματος [[1]](#one).



Εικόνα 2.1: Δεοξυριβονουκλεϊκό οξύ.

### **2.1.3. Αντιγραφή**

Για να καταστεί δυνατή η βιολογική κληρονομικότητα, το μόριο DNA πρέπει να αντιγραφεί. Κατά τη διάρκεια της αντιγραφής του DNA, οι δεσμοί μεταξύ των νουκλεοτιδίων με αντίθετους κλώνους της έλικας σπάνε και έτσι διαχωρίζονται και οι δύο κλώνοι του DNA. Με αυτόν τον τρόπο, κάθε σκέλος μπορεί να χρησιμεύσει ως πρότυπο για τη δημιουργία του αντίστοιχου του. Κάθε νέο αντίστοιχο συντίθεται από το ένζυμο DNA του πολυμεραζίου, διαβάζοντας προς την κατεύθυνση 3' έως 5' και προσθέτοντας τα συμπληρωματικά νουκλεοτίδια σε κάθε πρότυπο. Επιπλέον, πραγματοποιούνται μηχανισμοί κυτταρικής διόρθωσης και ελέγχου σφαλμάτων για την εξασφάλιση σχεδόν τέλειας αναπαραγωγής, χωρίς όμως να αποκλείεται η πιθανότητα να προκύψουν σφάλματα και να μην εντοπιστούν [[1]](#one).

## **2.2. Βιοπληροφορική**

### **2.2.1. Εφαρμογές**

Σε γενικές γραμμές, η βιοπληροφορική είναι ένα διεπιστημονικό πεδίο που χρησιμοποιεί υπολογιστές για την επίλυση προβλημάτων της μοριακής βιολογίας. Περιλαμβάνει κυρίως την ανάπτυξη αλγορίθμων και εργαλείων που μπορούν να αναλύσουν και να ερμηνεύσουν τεράστιες ποσότητες βιολογικών δεδομένων, όπως αλληλουχίες νουκλεοτιδίων, τομείς πρωτεϊνών, δομές πρωτεϊνών και άλλα, που θα ήταν ανέφικτα χωρίς αυτοματοποιημένο υπολογισμό. Αλλά η βιοπληροφορική συνεπάγεται επίσης τη δημιουργία και την επιμέλεια βάσεων δεδομένων, την ανάπτυξη στατιστικών τεχνικών και την πρόοδο της θεωρίας για την επίλυση τυπικών προβλημάτων που προκύπτουν από την ανάλυση βιολογικών δεδομένων κλίμακας γονιδιώματος. Από την ανάπτυξη του έργου του ανθρώπινου γονιδιώματος, ο τομέας της βιοπληροφορικής έχει αποκτήσει μεγάλο ενδιαφέρον. Σήμερα, η βιοπληροφορική εξελίσσεται ραγδαία παράλληλα με τις πιο πρόσφατες τεχνολογικές βελτιώσεις στις πλατφόρμες αλληλούχισης. Πολλοί ερευνητές έχουν ενώσει τις δυνάμεις τους σε διεπιστημονικές ομάδες προκειμένου να διερευνήσουν τις γνώσεις της βιολογίας του γονιδιώματος. Μερικοί επιστήμονες κάνουν μια διάκριση μεταξύ του όρου βιοπληροφορική και της υπολογιστικής βιολογίας. Ενώ αυτοί οι τομείς είναι πράγματι ευρείες και ποικίλες, αυτές οι διακρίσεις από άποψη δεν είναι συνεπείς ή σαφώς καθορισμένες [[1]](#one).

Καλά παραδείγματα βιοπληροφορικών εφαρμογών είναι η αναδίπλωση πρωτεϊνών, η κατασκευή εξελικτικών δέντρων, η χαρτογράφηση γονιδιώματος και η ανίχνευση παραλλαγών. Μεταξύ πολλών άλλων, μια από τις θεμελιώδεις εφαρμογές της βιοπληροφορικής είναι η χαρτογράφηση του γονιδιώματος. Όπως εξηγήθηκε παραπάνω, λόγω τεχνικών περιορισμών ένα μεμονωμένο χρωμόσωμα δεν μπορεί να διαβαστεί ταυτόχρονα. Ωστόσο, όπως γνωρίζουμε, σχετικά μικρές αλληλουχίες DNA μπορούν να ληφθούν με προσδιορισμό αλληλουχίας. Είναι σημαντικό να σημειωθεί ότι μια ανάγνωση μπορεί να προέρχεται από οποιοδήποτε από τα δύο σκέλη ενός χρωμοσώματος. Αυτό μπορεί να γίνει ακόμη πιο περίπλοκο στην περίπτωση του πολυπλοειδούς οργανισμού, όπου η ανάγνωση μπορεί να προέρχεται από οποιονδήποτε από τους απλότυπους. Για την πλειονότητα των εφαρμογών, όλα αυτά τα μικρά κομμάτια πρέπει να τακτοποιηθούν μαζί σαν παζλ για να ληφθεί ένα “de-novo” γονιδίωμα ή να ευθυγραμμιστούν με ένα ήδη υπάρχον [[1]](#one).

### **2.2.2. Τομείς έρευνας**

Ο όρος βιοπληροφορική επινοήθηκε από τον Paulien Hogeweg το 1979 για τη μελέτη των διεργασιών πληροφορικής σε βιοτικά συστήματα. Χρησιμοποιήθηκε αρχικά από τα τέλη της δεκαετίας του 1980 στη γονιδιωματική και τη γενετική, ιδιαίτερα σε εκείνους τους τομείς της γονιδιωματικής που περιλαμβάνουν μεγάλης κλίμακας προσδιορισμό αλληλουχίας DNA [[4]](#four).

Η βιοπληροφορική είναι η επιστήμη της αποθήκευσης, εξαγωγής, οργάνωσης, ανάλυσης, ερμηνείας και χρήσης πληροφοριών από βιολογικές αλληλουχίες και μόρια. Τροφοδοτήθηκε κυρίως από τις προόδους στις τεχνικές προσδιορισμού αλληλουχίας και χαρτογράφησης DNA. Τις τελευταίες δεκαετίες, οι ραγδαίες εξελίξεις στις τεχνολογίες του γονιδιώματος και άλλων μοριακών ερευνών και οι εξελίξεις στις τεχνολογίες της πληροφορίας έχουν συνδυαστεί για να παράγουν έναν τεράστιο όγκο πληροφοριών που σχετίζονται με τη μοριακή βιολογία. Ο πρωταρχικός στόχος της βιοπληροφορικής είναι να αυξήσει την κατανόηση των βιολογικών διεργασιών [[4]](#four).

* Διαδοχική ανάλυση: Η ανάλυση ακολουθίας είναι η πιο πρωτόγονη λειτουργία στην υπολογιστική βιολογία. Αυτή η λειτουργία συνίσταται στην εύρεση ποιο μέρος των βιολογικών αλληλουχιών είναι όμοιο και ποιο μέρος διαφέρει κατά τη διάρκεια της ιατρικής ανάλυσης και των διαδικασιών χαρτογράφησης γονιδιώματος. Η ανάλυση αλληλουχίας συνεπάγεται την υποβολή μιας αλληλουχίας DNA ή πεπτιδίου σε ευθυγράμμιση αλληλουχιών, βάσεις δεδομένων αλληλουχιών, επαναλαμβανόμενες αναζητήσεις αλληλουχιών ή άλλες μεθόδους βιοπληροφορικής σε έναν υπολογιστή.
* Γονιδιωματική σημείωση: Στο πλαίσιο της γονιδιωματικής, ο σχολιασμός είναι η διαδικασία σήμανσης των γονιδίων και άλλων βιολογικών χαρακτηριστικών σε μία αλληλουχία DNA. Το πρώτο σύστημα λογισμικού σχολιασμού γονιδιώματος σχεδιάστηκε το 1995 από τον Δρ. Owen White.
* Ανάλυση γονιδιακής έκφρασης: Η έκφραση πολλών γονιδίων μπορεί να προσδιοριστεί με τη μέτρηση των επιπέδων mRNA με διάφορες τεχνικές όπως μικροσυστοιχίες, προσδιορισμός αλληλουχίας εκφραζόμενης αλληλουχίας cDNA (EST), σειριακή ανάλυση αλληλουχίας ετικετών έκφρασης γονιδίου (SAGE), αλληλουχία μαζικής παράλληλης υπογραφής (MPSS) ή διάφορες εφαρμογές πολυπλεξίας υβριδισμού “in-situ” και άλλα. Όλες αυτές οι τεχνικές είναι εξαιρετικά επιρρεπείς στο θόρυβο και υπόκεινται σε μεροληψία στη βιολογική μέτρηση. Εδώ ο κύριος ερευνητικός τομέας περιλαμβάνει την ανάπτυξη στατιστικών εργαλείων για τον διαχωρισμό του σήματος από τον θόρυβο σε μελέτες γονιδιακής έκφρασης υψηλής απόδοσης.
* Ανάλυση έκφρασης πρωτεΐνης: Η γονιδιακή έκφραση μετριέται με πολλούς τρόπους, συμπεριλαμβανομένης της έκφρασης mRNA και πρωτεΐνης, ωστόσο η έκφραση πρωτεΐνης είναι μία από τις καλύτερες ενδείξεις πραγματικής γονιδιακής δραστηριότητας, καθώς οι πρωτεΐνες είναι συνήθως οι τελικοί καταλύτες της κυτταρικής δραστηριότητας. Οι μικροσυστοιχίες πρωτεϊνών και η φασματομετρία μάζας υψηλής απόδοσης μπορούν να παρέχουν ένα στιγμιότυπο των πρωτεϊνών που υπάρχουν σε ένα βιολογικό δείγμα.
* Ανάλυση μεταλλάξεων στον καρκίνο: Στον καρκίνο, τα γονιδιώματα των προσβεβλημένων κυττάρων αναδιατάσσονται με πολύπλοκους ή ακόμα και απρόβλεπτους τρόπους. Μαζικές προσπάθειες προσδιορισμού αλληλουχίας χρησιμοποιούνται για τον εντοπισμό προηγουμένως άγνωστων σημειακών μεταλλάξεων σε μια ποικιλία γονιδίων στον καρκίνο. Οι βιοπληροφορικοί συνεχίζουν να παράγουν εξειδικευμένα αυτοματοποιημένα συστήματα για τη διαχείριση του τεράστιου όγκου των δεδομένων αλληλουχίας που παράγονται και δημιουργούν νέους αλγόριθμους και λογισμικό για να συγκρίνουν τα αποτελέσματα της αλληλουχίας με την αυξανόμενη συλλογή αλληλουχιών ανθρώπινου γονιδιώματος και πολυμορφισμών βλαστικής σειράς. Χρησιμοποιούνται νέες τεχνολογίες φυσικής ανίχνευσης, όπως μικροσυστοιχίες ολιγονουκλεοτιδίων για τον εντοπισμό χρωμοσωμικών κερδών και απωλειών και συστοιχίες πολυμορφισμού ενός νουκλεοτιδίου για την ανίχνευση γνωστών σημειακών μεταλλάξεων. Ένας άλλος τύπος δεδομένων που απαιτεί ανάπτυξη νέας πληροφορικής είναι η ανάλυση βλαβών που βρέθηκε να είναι υποτροπιάζουσες μεταξύ πολλών όγκων.
* Πρόβλεψη δομής πρωτεΐνης: Η αλληλουχία αμινοξέων μιας πρωτεΐνης μπορεί εύκολα να προσδιοριστεί από την αλληλουχία στο γονίδιο που την κωδικοποιεί. Στις περισσότερες περιπτώσεις, αυτή η κύρια δομή καθορίζει μοναδικά μια δομή στο φυσικό της περιβάλλον. Η γνώση αυτής της δομής είναι ζωτικής σημασίας για την κατανόηση της λειτουργίας της πρωτεΐνης. Λόγω έλλειψης καλύτερων όρων, οι δομικές πληροφορίες ταξινομούνται συνήθως ως δευτερογενείς, τριτογενείς και τεταρτοταγείς. Η πρόβλεψη της πρωτεϊνικής δομής είναι μία από τις πιο σημαντικές για το σχεδιασμό φαρμάκων και το σχεδιασμό νέων ενζύμων. Μια γενική λύση σε τέτοιες προβλέψεις παραμένει ένα ανοιχτό πρόβλημα για τους ερευνητές.
* Συγκριτική γονιδιωματική: Η συγκριτική γονιδιωματική είναι η μελέτη της σχέσης δομής και λειτουργίας του γονιδιώματος μεταξύ διαφορετικών βιολογικών ειδών. Η εύρεση γονιδίων είναι μια σημαντική εφαρμογή της συγκριτικής γονιδιωματικής, όπως και η ανακάλυψη νέων, μη κωδικοποιητικών λειτουργικών στοιχείων του γονιδιώματος. Η συγκριτική γονιδιωματική εκμεταλλεύεται τόσο τις ομοιότητες όσο και τις διαφορές στις πρωτεΐνες, το RNA και τις ρυθμιστικές περιοχές διαφορετικών οργανισμών.
* Μοντελοποίηση βιολογικών συστημάτων: Η μοντελοποίηση βιολογικών συστημάτων είναι ένα σημαντικό έργο της βιολογίας συστημάτων και της μαθηματικής βιολογίας. Η βιολογία υπολογιστικών συστημάτων στοχεύει στην ανάπτυξη και χρήση αποτελεσματικών αλγορίθμων, δομών δεδομένων, εργαλείων οπτικοποίησης και επικοινωνίας για την ενοποίηση μεγάλων ποσοτήτων βιολογικών δεδομένων με στόχο τη μοντελοποίηση υπολογιστή. Περιλαμβάνει τη χρήση προσομοιώσεων βιολογικών συστημάτων σε υπολογιστή, όπως κυτταρικά υποσυστήματα όπως τα δίκτυα μεταβολιτών και ενζύμων, μονοπάτια μεταγωγής σήματος και ρυθμιστικά δίκτυα γονιδίων για την ανάλυση και την οπτικοποίηση των πολύπλοκων συνδέσεων αυτών των κυτταρικών διεργασιών. Η τεχνητή ζωή είναι μια προσπάθεια κατανόησης των εξελικτικών διαδικασιών μέσω της προσομοίωσης σε υπολογιστή απλών μορφών ζωής.
* Ανάλυση εικόνας υψηλής απόδοσης: Οι υπολογιστικές τεχνολογίες χρησιμοποιούνται για την επιτάχυνση ή την πλήρη αυτοματοποίηση της επεξεργασίας, της ποσοτικοποίησης και της ανάλυσης μεγάλων ποσοτήτων βιοϊατρικών εικόνων υψηλού περιεχομένου. Τα σύγχρονα συστήματα ανάλυσης εικόνας αυξάνουν την ικανότητα ενός παρατηρητή να κάνει μετρήσεις από ένα μεγάλο ή πολύπλοκο σύνολο εικόνων. Ένα πλήρως ανεπτυγμένο σύστημα ανάλυσης μπορεί να αντικαταστήσει πλήρως τον παρατηρητή. Η βιοϊατρική απεικόνιση γίνεται πιο σημαντική τόσο για τη διάγνωση όσο και για την έρευνα.
* Σύνδεση πρωτεΐνης με πρωτεΐνη: Τις τελευταίες δύο δεκαετίες, έχουν προσδιοριστεί δεκάδες χιλιάδες τρισδιάστατες δομές πρωτεΐνης με κρυσταλλογραφία ακτίνων Χ και φασματοσκοπία πυρηνικού μαγνητικού συντονισμού πρωτεΐνης. Ένα κεντρικό ερώτημα για τον βιολόγο επιστήμονα είναι αν είναι πρακτικό να προβλέψει κανείς πιθανές αλληλεπιδράσεις πρωτεΐνης με πρωτεΐνη μόνο με βάση αυτά τα τρισδιάστατα σχήματα, χωρίς να κάνει πειράματα αλληλεπίδρασης πρωτεΐνης με πρωτεΐνη. Μια ποικιλία μεθόδων έχουν αναπτυχθεί για την αντιμετώπιση του προβλήματος σύνδεσης πρωτεΐνης με πρωτεΐνη, αν και φαίνεται ότι υπάρχει ακόμη πολλή δουλειά που πρέπει να γίνει σε αυτόν τον τομέα.

## **2.3. Εξόρυξη δεδομένων**

### **2.3.1. Εξόρυξη Δεδομένων στη Βιοπληροφορική**

Η εξόρυξη δεδομένων αναφέρεται στην εξαγωγή ή την «εξόρυξη» γνώσης από μεγάλες ποσότητες δεδομένων. Η Εξόρυξη Δεδομένων (DM) είναι η επιστήμη της εύρεσης νέων προτύπων και σχέσεων σε τεράστιο όγκο δεδομένων. Ορίζεται ως «η διαδικασία ανακάλυψης σημαντικών νέων συσχετισμών, μοτίβων και τάσεων με την αναζήτηση μεγάλων ποσοτήτων δεδομένων που είναι αποθηκευμένα σε αποθήκες». Η εξόρυξη δεδομένων ονομάζεται επίσης μερικές φορές ανακάλυψη γνώσης σε βάσεις δεδομένων (KDD). Η εξόρυξη δεδομένων δεν είναι συγκεκριμένη για κανέναν κλάδο. Απαιτεί έξυπνες τεχνολογίες και την προθυμία να διερευνηθεί η δυνατότητα της κρυφής γνώσης που βρίσκεται στα δεδομένα [[2]](#two).

Οι προσεγγίσεις της Εξόρυξης Δεδομένων φαίνονται ιδανικά κατάλληλες για τη Βιοπληροφορική, καθώς είναι πλούσιες σε δεδομένα, αλλά στερείται μιας ολοκληρωμένης θεωρίας για την οργάνωση της ζωής σε μοριακό επίπεδο. Οι εκτεταμένες βάσεις δεδομένων βιολογικών πληροφοριών δημιουργούν προκλήσεις και ευκαιρίες για την ανάπτυξη νέων μεθόδων KDD. Η εξόρυξη βιολογικών δεδομένων βοηθά στην εξαγωγή χρήσιμης γνώσης από τεράστια σύνολα δεδομένων που συγκεντρώνονται στη βιολογία και σε άλλους σχετικούς τομείς των βιοεπιστημών, όπως η ιατρική και η νευροεπιστήμη [[2]](#two).

Οι εφαρμογές της εξόρυξης δεδομένων στη βιοπληροφορική περιλαμβάνουν εύρεση γονιδίων, ανίχνευση τομέα πρωτεϊνικής λειτουργίας, ανίχνευση μοτίβου λειτουργίας, συμπέρασμα πρωτεϊνικής λειτουργίας, διάγνωση ασθενειών, πρόγνωση ασθενειών, βελτιστοποίηση θεραπείας ασθενειών, ανακατασκευή δικτύου αλληλεπίδρασης πρωτεϊνών και γονιδίων, καθαρισμό δεδομένων και πρόβλεψη υποκυτταρικής θέσης πρωτεΐνης [[2]](#two).

Για παράδειγμα, οι τεχνολογίες μικροσυστοιχιών χρησιμοποιούνται για την πρόβλεψη της έκβασης ενός ασθενούς. Με βάση τα δεδομένα γονοτυπικής μικροσυστοιχίας των ασθενών, μπορεί να εκτιμηθεί ο χρόνος επιβίωσής τους και ο κίνδυνος μετάστασης ή υποτροπής του όγκου. Η μηχανική μάθηση μπορεί να χρησιμοποιηθεί για την ταυτοποίηση πεπτιδίων μέσω φασματοσκοπίας μάζας. Η συσχέτιση μεταξύ ιόντων θραυσμάτων σε ένα φάσμα μάζας σε σειρά είναι ζωτικής σημασίας για τη μείωση των στοχαστικών αναντιστοιχιών για την ταυτοποίηση πεπτιδίων μέσω αναζήτησης βάσης δεδομένων. Ένας αποτελεσματικός αλγόριθμος βαθμολόγησης που εξετάζει τις σχετικές πληροφορίες με συντονισμένο και ολοκληρωμένο τρόπο είναι ιδιαίτερα επιθυμητός [[2]](#two).

### **2.3.2. Προσεγγίσεις**

Πολλές μέθοδοι εξόρυξης δεδομένων έχουν σχεδιαστεί για συλλογές αντικειμένων που αντιπροσωπεύονται καλά σε άκαμπτες μορφές πίνακα. Ωστόσο, εκτός από μαζικά σύνολα αδόμητων πληροφοριών και μη τυποποιημένων πληροφοριών, βρισκόμαστε όλο και περισσότερο αντιμέτωποι με μεγάλες συλλογές αλληλένδετων αντικειμένων των οποίων η φυσική αναπαράσταση είναι σε δεδομένα νέφους σημείου ή σε γραφήματα δακτυλογράφησης [[5]](#five).

Οι προηγμένες προσεγγίσεις εξόρυξης δεδομένων περιλαμβάνουν:

* Εξόρυξη δεδομένων βάσει γραφημάτων
* Εξόρυξη δεδομένων με βάση την εντροπία
* Εξόρυξη τοπολογικών δεδομένων

# **3. Multiple Longest Common Subsequence (MLCS)**

## **3.1. Εισαγωγή**

Το πρόβλημα Multiple Longest Common Subsequence (MLCS) είναι να βρεθεί η μεγαλύτερη υποακολουθία που μοιράζεται μεταξύ δύο ή περισσότερων αλληλουχιών. Είναι ένα κλασικό πρόβλημα της επιστήμης των υπολογιστών με σημαντικές εφαρμογές σε πολλούς τομείς όπως η ανάκτηση πληροφοριών και η υπολογιστική βιολογία. Για περισσότερα από 30 χρόνια, έχουν γίνει σημαντικές προσπάθειες για να βρεθεί ένας αποτελεσματικός αλγόριθμος για το πρόβλημα MLCS. Η πιο σημαντική συνεισφορά έχει γίνει στη μελέτη της απλούστερης περίπτωσης MLCS δύο ή τριών ακολουθιών. Ωστόσο, ενώ αρκετές προσπάθειες για την εύρεση ενός αποτελεσματικού αλγορίθμου για μια γενική περίπτωση με περισσότερες από 3 ακολουθίες, δεν έχει ακόμη αναπτυχθεί. Μια γενική περίπτωση MLCS έχει τεράστια αξία για την υπολογιστική βιολογία και την υπολογιστική γονιδιωματική που ασχολείται με τις βιολογικές αλληλουχίες. Με τον αυξανόμενο όγκο βιολογικών δεδομένων και την επικρατούσα χρήση εργαλείων υπολογιστικής ανάλυσης ακολουθιών, ένας αποτελεσματικός αλγόριθμος MLCS που εφαρμόζεται σε πολλές ακολουθίες θα έχει σημαντικό αντίκτυπο στην υπολογιστική βιολογία και τις εφαρμογές [[3]](#three).

## **3.2. Βασικές αρχές μεθόδου κυρίαρχου σημείου**

Υποθέτοντας ότι οι ακολουθίες είναι σειρές χαρακτήρων που ορίζονται σε ένα πεπερασμένο αλφάβητο Σ. Έστω a και b δύο ακολουθίες μηκών n και k αντίστοιχα. Για την ακολουθία a = s1s2 . . . sn, ακολουθία b = si1si2 ...sik ονομάζεται υποακολουθία του a εάν 1 ≤ ij ≤ n, για 1≤ j ≤ k, και is < it, για 1 ≤s < t ≤ k. Έστω S = {a1,a2,...,ad} ένα σύνολο ακολουθιών πάνω από το αλφάβητο Σ. Μια πολλαπλή μακρύτερη κοινή υποακολουθία (MLCS) για το σύνολο S είναι μια ακολουθία b τέτοια ώστε (i) b είναι μια υποακολουθία του ai, 1 ≤ i ≤ d και (ii) b είναι η μεγαλύτερη που ικανοποιεί το (i) [[3]](#three).

Έστω L ο πίνακας βαθμολογίας για ένα σύνολο d ακολουθιών, a1, a2, ..., ad. Ένα σημείο p στον πίνακα L συμβολίζεται ως p = [p1,p2,...,pd], όπου κάθε pi είναι μια συντεταγμένη του p για την αντίστοιχη συμβολοσειρά ai. Η τιμή στη θέση p του πίνακα L συμβολίζεται ως L[p]. Ένα σημείο p = [p1,p2,...,pd] στο L ονομάζεται ταίριασμα εάν a1[p1] = a2[p2] = . . . = ad[pd]. Αν ένα ταίριασμα p αντιστοιχεί στον χαρακτήρα s ∈ Σ, δηλ. ai[pi] = s, συμβολίζεται ως p(s) [[3]](#three).

Για δύο σημεία p = [p1,p2,...,pd] και q = [q1,q2,...,qd], κυριαρχεί το p στο q αν pi ≤ qi, για 1 ≤ i ≤ d.Αν το p κυριαρχεί του q, αυτή η σχέση δηλώνεται σαν p ≤ q. Ομοίως, το p κυριαρχεί έντονα στο q εάν pi < qi, 1 ≤ i ≤ d. Η σχέση αυτή συμβολίζεται ως p < q. Το p δεν κυριαρχεί στο q, αν υπάρχει ένα i, 1≤ i ≤d, έτσι ώστε pi < qi. Για κάποια σημεία p και q, γίνεται να θσχύουν ταυτόχρονα p̸≤q και q̸≤p. Ένα Ταίριασμα p(s) ονομάζεται k-κυρίαρχο σημείο, ή k-κυρίαρχο, ή κυρίαρχο στο επίπεδο k, αν (i) L[p] = k και (ii) δεν υπάρχει άλλο ταίριασμα q(s) ικανοποιώντας την (i) και q ≤ p, δηλαδή, το p δεν κυριαρχείται από ένα άλλο σημείο q με την ίδια τιμή k και για το ίδιο k χαρακτήρας s. Το σύνολο όλων των k-κυρίαρχων συμβολίζεται ως D [[3]](#three).

Ένα παράδειγμα του πίνακα L, το σύνολο των κυρίαρχων σημείων και οι υπόλοιπες αντιστοιχίσεις για δύο ακολουθίες φαίνονται στην εικόνα παρακάτω [[3]](#three).

Εικόνα που περιέχει βέλος

Περιγραφή που δημιουργήθηκε αυτόματα

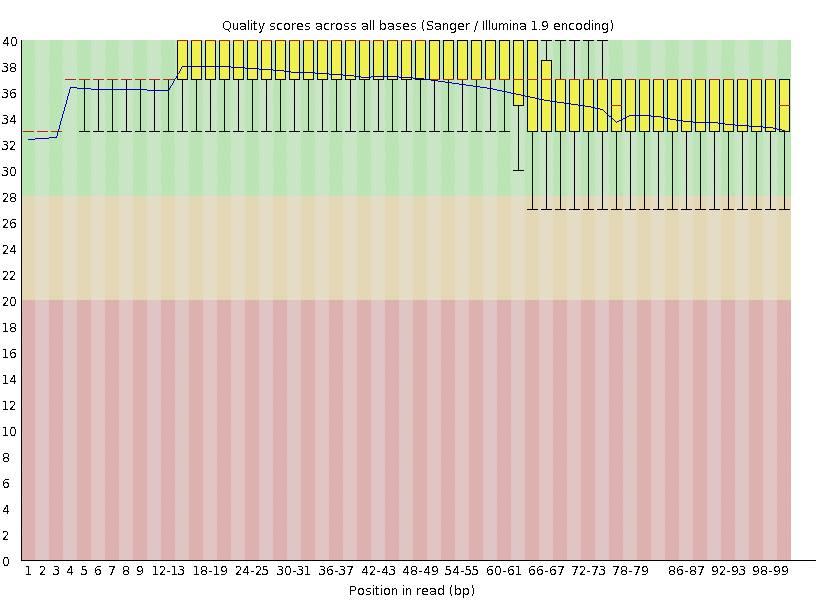
Εικόνα 3.1: Τα κυρίαρχα σημεία και οι αντιστοιχίσεις για δύο ακολουθίες, a1 = GTAATCTAAC και a2 = GATTACA. Οι κυρίαρχες θέσεις κυκλώνονται, ενώ οι υπόλοιπες αντιστοιχίες, που δεν είναι κυρίαρχες, τετραγωνίζονται.

# **4. Τεχνολογίες αλληλουχίας**

## **4.1. Εισαγωγή**

Η αλληλούχιση είναι η διαδικασία λήψης της αλληλουχίας των βάσεων των μορίων DNA. Η ικανότητα προσδιορισμού αλληλουχίας γονιδιωμάτων είναι μία από τις κύριες πηγές των σημερινών βιολογικών πληροφοριών, που επιτρέπει στους επιστήμονες να διερευνήσουν τις κυτταρικές διεργασίες και οδηγεί σε ανεκτίμητες βιολογικές γνώσεις. Από την εμφάνιση των πρώτων πειραματικών τεχνικών που το επιτρέπουν, η αλληλουχία DNA έχει γίνει βασική τεχνολογία όχι μόνο για τη γενική βιολογική έρευνα αλλά και για πολλούς εφαρμοσμένους τομείς όπως η βιοτεχνολογία, οι εξελικτικές μελέτες, η μεταγονιδιωματική, η ιατρική διαγνωστική και η εξατομικευμένη ιατρική [[1]](#one).

* High-throughput sequencing (HTS): Η ακολουθία υψηλής απόδοσης (HTS) υποδηλώνει κυρίως εκείνες τις τεχνολογίες που είναι ικανές να ακολουθούν την τάξη των εκατομμυρίων αναγνώσεων ανά εκτέλεση, σχετικά γρήγορα σε πολύ χαμηλή τιμή. Χρησιμοποιώντας την, γίνεται να ακολουθηθεί σε βάθος ένα γονιδίωμα παράγοντας τεράστιες ποσότητες γονιδιωματικών δεδομένων. Αυτές οι τεχνικές ονομάζονται συχνά μαζική παράλληλη αλληλουχία DNA καθώς βασίζονται σε εκατομμύρια αντιδράσεις που εκτελούνται ταυτόχρονα για να επιτευχθούν πολύ υψηλές αποδόσεις παραγωγής.
* Synthetic long-read sequencing (SLSR): Προκειμένου να αυξηθεί το αποτελεσματικό μήκος ανάγνωσης που παράγουν οι πλατφόρμες “Illumina”, αναπτύχθηκε η συνθετική αλληλουχία μακράς ανάγνωσης της “Moleculo TruSeq”. Αυτή η μέθοδος βασίζεται σε ένα τροποποιημένο πρωτόκολλο προετοιμασίας βιβλιοθήκης στο οποίο μακρά θραύσματα DNA - μήκους 10 Kbp - εξάγονται, ενισχύονται και επανασυναρμολογούνται από τακτικές σύντομες αναγνώσεις. Αυτά τα μεγάλα θραύσματα τοποθετούνται σε “wells” και στη συνέχεια εκχωρούνται μια μοναδική ακολουθία γραμμωτού κώδικα. Αυτό ακολουθείται από μια τακτική διαδικασία αλληλουχίας “Illumina”. Αργότερα, όλες οι διαδοχικές αναγνώσεις απόκωδικοποιούνται σε “pools” και “de-novo” συναρμολογούμενες ανεξάρτητα.



Εικόνα 4.1: Προφίλ ποιότητας Illumina.

* Single molecule sequencing (SMS): Η αλληλουχία ενός μορίου (SMS) φαίνεται να δίνει λύση σε μερικά από τα πιο ενοχλητικά προβλήματα που προέρχονται από τεχνολογίες HTS. Στοχεύει στην απλούστευση της προετοιμασίας του δείγματος, στη μείωση της ποσότητας του δείγματος που απαιτείται για τη λειτουργία και στην άρση της ανάγκης για ενίσχυση προτύπου,αφαιρώντας προκαταλήψεις και τεχνουργήματα.

## **4.2. High-throughput sequencing (HTS)**

Ένα από τα εξαιρετικά ορόσημα που επιτεύχθηκε τα τελευταία χρόνια στον τομέα της έρευνας στη βιοτεχνολογία ήταν η ανάπτυξη της αλληλουχίας υψηλής απόδοσης (HTS). Λόγω του γεγονότος ότι αυτή τη στιγμή είναι τεχνικά αδύνατο να αποκωδικοποιηθεί το γονιδίωμα στο σύνολό του, οι τεχνολογίες HTS διαβάζουν δισεκατομμύρια σχετικά μικρά κομμάτια ενός γονιδιώματος σε τυχαίες τοποθεσίες. Οι τεχνολογίες HTS είναι σε θέση να επαναπροσδιορίσουν την αλληλουχία ενός ανθρώπου, δηλαδή να καθορίσουν τις διαφορές μεταξύ του ατομικού γονιδιώματος του/της και του γονιδιώματος αναφοράς για το ανθρώπινο είδος, σε πολύ σύντομο χρονικό διάστημα. Έχουν επίσης ανοίξει το δρόμο για την ανάπτυξη μιας σειράς νέων πρωτοκόλλων και μεθόδων, που οδηγούν σε νέες γνώσεις στη γονιδιωματική και τη βιολογία γενικότερα [[1]](#one).

Ωστόσο, οι τεχνολογίες HTS αποτελούν επίσης πρόκληση για τις παραδοσιακές μεθόδους ανάλυσης δεδομένων. Αυτό οφείλεται στον τεράστιο όγκο δεδομένων προς επεξεργασία και στην ανάγκη για βελτιωμένους αλγόριθμους ευθυγράμμισης που μπορούν να παράγουν γρήγορα ακριβή αποτελέσματα [[1]](#one).

# **5. Approximate string matching**

## **5.1. Στοίχιση ακολουθίας**

Η στοίχιση αλληλουχίας υποδηλώνει μια σειρά βιοπληροφορικών αλγορίθμων των οποίων ο σκοπός είναι να καθιερώσουν ομολογία μεταξύ γονιδιωματικών αλληλουχιών. Για να γίνει αυτό, οι ακολουθίες ευθυγραμμίζονται –ευθυγραμμισμένες μεταξύ τους– έτσι ώστε ο βαθμός ομοιότητας των κατάλληλα ταιριασμένων τμημάτων των διαφορετικών ακολουθιών, στην περίπτωση τοπικής ευθυγράμμισης, ή ο βαθμός ομοιότητας των πλήρων ακολουθιών, σε περίπτωση της καθολικής ευθυγράμμισης, μεγιστοποιείται σύμφωνα με μια δεδομένη συνάρτηση απόστασης συμβολοσειράς ή βαθμολογία. Η στοίχιση ακολουθιών είναι ένας γενικός όρος και έχει πολλαπλές υλοποιήσεις, όπως στις συγκρίσεις αλληλουχιών ανά ζεύγη, στοίχιση πολλαπλών ακολουθιών, κατασκευή εξελικτικών δέντρων και άλλα [[1]](#one).

## **5.2. Εισαγωγή στο ASP**

Για πολλές δεκαετίες στο πεδίο της επιστήμης των υπολογιστών, το πρόβλημα της ευθυγράμμισης ακολουθιών έχει επίσημα μελετηθεί και είναι γενικά γνωστό ως κατά προσέγγιση ταίριασμα συμβολοσειρών. Η κατά προσέγγιση αντιστοίχιση συμβολοσειρών (ASM) αναφέρεται στην εύρεση συμβολοσειρών που ταιριάζουν περίπου με ένα μοτίβο. Εκτεταμένη έρευνα έχει διεξαχθεί σε αυτό το πεδίο από τη δεκαετία του 1960, κυρίως στο πλαίσιο της ανάκτησης. Ως αποτέλεσμα, πολλοί αλγόριθμοι και τεχνικές έχουν προταθεί για την αντιμετώπιση του προβλήματος [[1]](#one).

Το ASM είναι ένα πολύ γενικό πεδίο στην επιστήμη των υπολογιστών και έχει πολλές διατυπώσεις. Ανάλογα με το πλαίσιο, τα προβλήματα ASM μπορούν εύκολα να αντιμετωπιστούν με ελαφρούς αλγόριθμους ή να αποκαλύψουν την πολύ δυσεπίλυτη φύση του ASM ως NP-πλήρη προβλήματα [[1]](#one).

## **5.3. Κλασικές δομές ευρετηρίου**

### **5.3.1. Hashes based indexes**

Πιθανώς η πιο απλή ιδέα ευρετηρίασης είναι αυτή της καταγραφής κάθε “k-mer” στο αρχικό κείμενο. Με αυτόν τον τρόπο, ένας πίνακας που μοιάζει με κατακερματισμό επιτρέπει την άμεση πρόσβαση σε κάθε “k-mer” και σε όλες τις θέσεις του κειμένου όπου εμφανίζεται. Παρά την απλότητα αυτής της ιδέας, οι κατακερματισμοί είναι το επίκεντρο των ερευνητών για πολλά χρόνια και η κεντρική δομή ευρετηρίου πολλών σημαντικών εργαλείων χαρτογράφησης, μερικά από αυτά εξακολουθούν να χρησιμοποιούνται ευρέως [[1]](#one).

### **5.3.2. Suffix tree**

Το “Suffix Tree” είναι μια από τις πιο θεμελιώδεις θεωρητικές δομές δεδομένων στην ανάκτηση πληροφοριών και στην επιστήμη των υπολογιστών γενικότερα. Το “Suffix Tree” μιας συμβολοσειράς T μήκους n είναι μια δοκιμασία που δημιουργήθηκε από όλα τα επιθήματα του T όπου συμπιέζονται ενιαίες διαδρομές. Καταλαμβάνει χώρο O(n) [[1]](#one).

Το κύριο πλεονέκτημα του “Suffix Tree” είναι ότι μπορεί να μετρήσει τον αριθμό των εμφανίσεων ενός μοτίβου P στο κείμενο σε χρόνο O(m). Μπορεί κάποιος να αναζητήσει το μοτίβο P στο δέντρο του επιθήματος κατεβαίνοντας από τη ρίζα του δέντρου σύμφωνα με τα διαδοχικά σύμβολα του P. Μόλις εντοπιστεί ο κόμβος που αντιστοιχεί στο μοτίβο, όλες οι εμφανίσεις μπορούν να καταγραφούν στον βέλτιστο χρόνο O(occ(P,T)) διασχίζοντας το υποδέντρο του κόμβου [[1]](#one).

### **5.3.3. Suffix array**

Ο “Suffix array” (SA) είναι μια πρακτική δομή ευρετηρίου που προτείνεται με σκοπό τη μείωση του αποτυπώματος μνήμης του “Suffix Tree”. Κατά κάποιο τρόπο, το SA είναι μια συμπαγής αναπαράσταση του “Suffix Tree”. Ωστόσο, εξακολουθεί να απαιτεί bits Θ(nlog(n)), αν και στις περισσότερες υλοποιήσεις, αν υποθέσουμε ότι οι συμβολοσειρές έχουν μέγιστο μήκος 4 GB, το μέγεθος είναι 4 φορές μεγαλύτερο από αυτό του κειμένου αναφοράς [[1]](#one).

# **6. Exact Pattern Matching**

## **6.1. First-Last Pattern Matching Algorithm**

### **6.1.1. Φάση προεπεξεργασίας**

Εφόσον το FLPM δρα με βάση συγκρίσεις, η φάση προεπεξεργασίας FLPM σαρώνει το κείμενο t για να διακρίνει τα παράθυρα του κειμένου που χρησιμοποιούνται αργότερα από τη φάση αντιστοίχισης. Στην αρχή της φάσης προεπεξεργασίας, ο πρώτος χαρακτήρας του μοτίβου p, δηλ., p[0], αναζητείται στο κείμενο t. Καθώς το μήκος του p είναι m, η αναζήτηση εκτελείται στο διάστημα t[0..n − m]. Σε αντίθεση με τον αλγόριθμο DCPM, κάθε φορά που ο χαρακτήρας p[0] ευθυγραμμίζεται με τον χαρακτήρα στη θέση s (στην οποία 0 ≤ s ≤ n−m) του κειμένου t, δηλ. t[s] = p[0], ο FLPM αλγόριθμος ελέγχει αμέσως την αντιστοιχία του p[m−1] στο t [s + m− 1] προκειμένου να προσδιορίσει το όριο του άκρου του παραθύρου. Εάν αυτοί οι δύο χαρακτήρες είναι επίσης ίδιοι, το παράθυρο t [s..s + m − 1] επιλέγεται ως υποψήφιο διάστημα που θα ελεγχθεί με μεγαλύτερη ακρίβεια στην επόμενη φάση. Στη συνέχεια, η φάση προεπεξεργασίας συνεχίζεται για τη διάγνωση περισσότερων παραθύρων. Κατά την αναγνώριση των παραθύρων, αν και αυτός ο αλγόριθμος εστιάζει στον πρώτο και τον τελευταίο χαρακτήρα του μοτίβου, μπορεί να επεκταθεί λαμβάνοντας υπόψη τους χαρακτήρες σε άλλες θέσεις, όπως m/4, m/2 και 3m/4 [[2]](#two).

Ο αλγόριθμος 1 παρουσιάζει τον ψευδοκώδικα αυτής της φάσης. Σε αυτόν τον αλγόριθμο, η αρχική ποσότητα του μετρητή βρόχου while, που υποδεικνύεται από τη μεταβλητή count, είναι μηδέν (Γραμμή 1). Ο βρόχος while αυτού του αλγορίθμου ξεκινά με μέτρηση = 0 και συνεχίζει μέχρι τη στιγμή που αυτός ο μετρητής φτάσει στο n − m (Γραμμή 3). Κατά τη διάρκεια του βρόχου (Γραμμές 3-11), ο πρώτος και ο τελευταίος χαρακτήρας του μοτίβου, δηλ., p[0] και p[m − 1], συγκρίνονται με t [count ] και t[count + m − 1], αντίστοιχα. Εάν το αποτέλεσμα και των δύο συγκρίσεων είναι σωστό, ο δείκτης έναρξης αυτού του παραθύρου, δηλαδή η καταμέτρηση, αποθηκεύεται στον πίνακα window\_index και ο αριθμός των παραθύρων, που αποθηκεύονται στη μεταβλητή num\_window, αυξάνεται κατά ένα. Επομένως, ο αριθμός των παραθύρων και οι δείκτες έναρξης τους επιτυγχάνονται στη φάση προεπεξεργασίας [[2]](#two).

~~Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα~~

Αλγόριθμος 6.1: Φάση προεπεξεργασίας αλγορίθμου FLPM.

### **6.1.2. Φάση αντιστοιχίας**

Αφού η φάση προεπεξεργασίας προσδιορίσει τα παράθυρα, η φάση αντιστοίχισης ερευνά τα παράθυρα για να βρει όλες τις εμφανίσεις του μοτίβου στο κείμενο. Επομένως, για τον δείκτη έναρξης κάθε παραθύρου, ας πούμε si, που βρέθηκε στην προηγούμενη φάση, οι χαρακτήρες p[1..m − 2] πρέπει να συγκριθούν με τους χαρακτήρες t [si + 1..si + m − 2]. Ο πρώτος και ο τελευταίος χαρακτήρας του μοτίβου και του παραθύρου, αντίστοιχα, δεν πρέπει να συγκριθούν ξανά επειδή η ομοιότητά τους είχε ήδη ελεγχθεί στην προηγούμενη φάση, δηλαδή, p[0] = t[si] και p[m−1] = t[ si +m−1]. Εάν υπάρχει μια ολόκληρη ομοιότητα του μοτίβου με το παράθυρο κειμένου, τότε αυτό το παράθυρο είναι μια απάντηση. Διαφορετικά, υπάρχει αναντιστοιχία. Μετά από έναν αγώνα ή αναντιστοιχία, αυτή η φάση συνεχίζεται με τη διερεύνηση των επόμενων παραθύρων [[2]](#two).

Ο αλγόριθμος 2 παρέχει τον ψευδοκώδικα αυτής της φάσης και θα περιγραφεί λεπτομερώς εδώ. Ο αριθμός των παραθύρων που προσδιορίστηκαν στην προηγούμενη φάση και οι δείκτες έναρξης τους θεωρούνται οι είσοδοι του Αλγόριθμου 2. Στις Γραμμές 3-17, ο βρόχος outer while του αλγορίθμου επαναλαμβάνει τις οδηγίες για όλα τα παράθυρα, δηλαδή, count = 0, 1, . . . , num\_window − 1. Η γραμμή 4 εκχωρεί τον δείκτη έναρξης του τρέχοντος παραθύρου στο s. Ο εσωτερικός βρόχος while (Γραμμές 6-11) ελέγχει την αντιστοιχία των μη οριακών χαρακτήρων του μοτίβου και εκείνων του παραθύρου κειμένου με τον δείκτη έναρξης του s. Εάν η στοίχιση πετύχει, τότε το s αποθηκεύεται στον πίνακα match\_index (Γραμμές 12-15) έτσι ώστε να προσδιοριστεί ο δείκτης έναρξης μιας εμφάνισης του μοτίβου p στο κείμενο t. Στη συνέχεια, ο αλγόριθμος συνεχίζει να μελετά το επόμενο παράθυρο [[2]](#two).

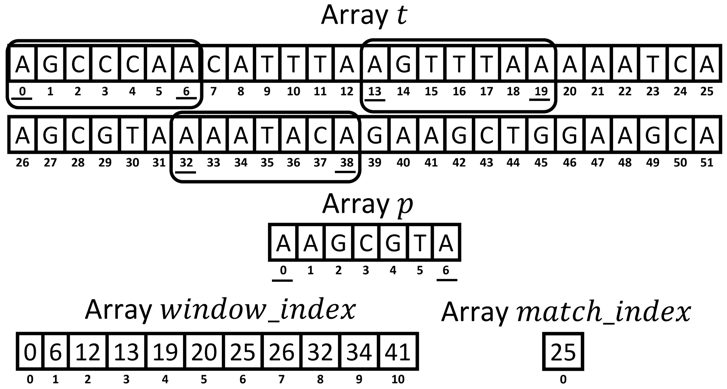
Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Αλγόριθμος 6.2: Φάση αντιστοίχισης του αλγορίθμου FLPM.

### **6.1.3. Παράδειγμα**

Για παράδειγμα, το παρακάτω σχήμα δείχνει το κείμενο t στον πίνακα t[0..51] και το μοτίβο p, το οποίο είναι AAGCGTA στον πίνακα p[0..6]. Η φάση προεπεξεργασίας αναζητά τον πρώτο και τον τελευταίο χαρακτήρα του μοτίβου, δηλαδή, p[0] και p[6], στο κείμενο. Στην αρχή του Αλγορίθμου 1, τα t[0] και t[6] ευθυγραμμίζονται με τα p[0] και p[6], αντίστοιχα. Έτσι, ο δείκτης έναρξης του πρώτου παραθύρου, δηλ. 0, αποθηκεύεται στον πίνακα window\_index. Ακολουθώντας αυτό το παράδειγμα, η φάση προεπεξεργασίας προσδιορίζει δέκα άλλα παράθυρα. Μετά τον τερματισμό αυτής της φάσης, η φάση αντιστοίχισης χρησιμοποιεί τον Αλγόριθμο 2 για να ελέγξει όλα τα παράθυρα προκειμένου να βρει το μοτίβο p στο κείμενο t. Κατά συνέπεια, το παράθυρο του t [25..31] και το σχέδιο είναι το ίδιο [[2]](#two).



Εικόνα 6.1: .Παράδειγμα για τη λειτουργία του αλγορίθμου FLPM.

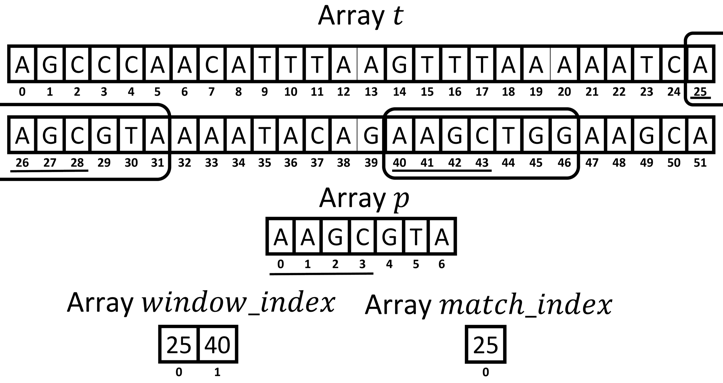
## **6.2. Processor-Aware Pattern Matching Algorithm**

Ο PAPM εκτελεί συγκρίσεις με βάση μια λέξη που αποτελείται από πολλούς χαρακτήρες ενώ το FLPM χρησιμοποιεί έναν αλγόριθμο αντιστοίχισης προτύπων που βασίζεται σε χαρακτήρες [[2]](#two).

Με την επεξεργαστική ισχύ ενός επεξεργαστή, ο PAPM συγκρίνει δύο λέξεις ταυτόχρονα. Στην περίπτωση ενός επεξεργαστή b−bit, οι καταχωρητές έχουν επίσης μήκος b−bit και, σε κάθε κύκλο εκτέλεσης, ο επεξεργαστής μπορεί να συγκρίνει τα δεδομένα δύο καταχωρητών. Δεδομένου ότι κάθε byte (ή χαρακτήρας) αποτελείται από οκτώ bit, ο αριθμός των επεξεργάσιμων byte (ή το μήκος λέξης) για αυτόν τον επεξεργαστή, που υποδεικνύεται με το word\_len, υπολογίζεται ως word\_len = b/8. Με άλλα λόγια, χρησιμοποιώντας τους καταχωρητές του κάθε φορά, ο επεξεργαστής μπορεί να συγκρίνει μια λέξη (συμπεριλαμβανομένων των χαρακτήρων word\_len) με μια άλλη. Για παράδειγμα, ένας επεξεργαστής 32-bit μπορεί να συγκρίνει ταυτόχρονα μια λέξη τεσσάρων χαρακτήρων με μια άλλη [[2]](#two).

Για να εφαρμοστεί η ικανότητα αυτού του επεξεργαστή στο πρόβλημα αντιστοίχισης προτύπων κατά την πρώτη φάση του PAPM, η πρώτη λέξη του μοτίβου αναζητείται στο κείμενο. Όπως φαίνεται στις Γραμμές 1-10 του Αλγορίθμου 3, η φάση προεπεξεργασίας του PAPM σαρώνει το διάστημα t [0..n−m] για να βρει τη λέξη p[0..word \_len−1]. Το αρχικό ευρετήριο των παραθύρων που βρέθηκαν αποθηκεύεται στον πίνακα window\_index. Σε αντίθεση με το FLPM, όταν τα παράθυρα αναζητούνται στο PAPM, ο μεγαλύτερος αριθμός χαρακτήρων (μια λέξη) που αναζητούνται στο κείμενο αποδίδει μικρότερο αριθμό παραθύρων, γεγονός που μπορεί να μειώσει τη χρονική διάρκεια που απαιτείται για την επόμενη φάση [[2]](#two).

Στη φάση αντιστοίχισης του Αλγορίθμου 3, που εμφανίζεται στις Γραμμές 11-34, οι λέξεις του μοτίβου p συγκρίνονται με τις λέξεις των παραθύρων που βρέθηκαν, αντίστοιχα. Στην αρχή αυτής της φάσης (Γραμμές 11-16), υπολογίζεται ο δείκτης έναρξης για τη σύγκριση λέξεων. Με τη ρύθμιση αυτού του δείκτη έναρξης, ο αλγόριθμος εκτελείται σωστά, ακόμα κι αν το μήκος του μοτίβου και των παραθύρων δεν είναι ακέραιο πολλαπλάσιο του μήκους της λέξης. Οι γραμμές 19-34 του Αλγόριθμου 3 είναι παρόμοιες με τις Γραμμές 3-17 του Αλγόριθμου 2. Ωστόσο, ο Αλγόριθμος 3 εκτελεί επεξεργασία βασισμένη σε λέξεις σε αντίθεση με την επεξεργασία βάσει χαρακτήρων στον Αλγόριθμο 2 [[2]](#two).



Εικόνα 6.2: Ένα παράδειγμα της λειτουργίας του αλγορίθμου PAPM.

Το Σχήμα παρέχει ένα παράδειγμα που χρησιμοποιεί τον Αλγόριθμο 3 που εκτελείται σε έναν επεξεργαστή 32 bit. Στη φάση προεπεξεργασίας, η πρώτη λέξη (αποτελούμενη από τους πρώτους τέσσερις χαρακτήρες) του μοτίβου p αναζητείται στο κείμενο t. Μετά τη φάση προεπεξεργασίας του αλγόριθμου PAPM, ο πίνακας window\_index αποτελείται από δύο δείκτες έναρξης των παραθύρων που βρέθηκαν, δηλ. 25 και 4. Για αυτό το παράδειγμα, θα πρέπει να σημειωθεί ότι ο αλγόριθμος FLPM προσδιορίζει 11 δείκτες έναρξης ως υποψήφια διαστήματα ή παράθυρα . Επομένως, το PAPM μειώνει τον αριθμό των παραθύρων που προσδιορίζονται. Στη φάση αντιστοίχισης, δεδομένου ότι το υπόλοιπο του μήκους του μοτίβου στο μήκος της λέξης είναι 3, ο δείκτης έναρξης για την αντιστοίχιση είναι επίσης 3 (Γραμμές 11-16). Επομένως, η δεύτερη λέξη του μοτίβου για ευθυγράμμιση με τη δεύτερη λέξη των windows είναι η CGTA. Μετά τον τερματισμό αυτής της φάσης, μόνο ένα από τα δύο παράθυρα ταιριάζει με το μοτίβο [[2]](#two).

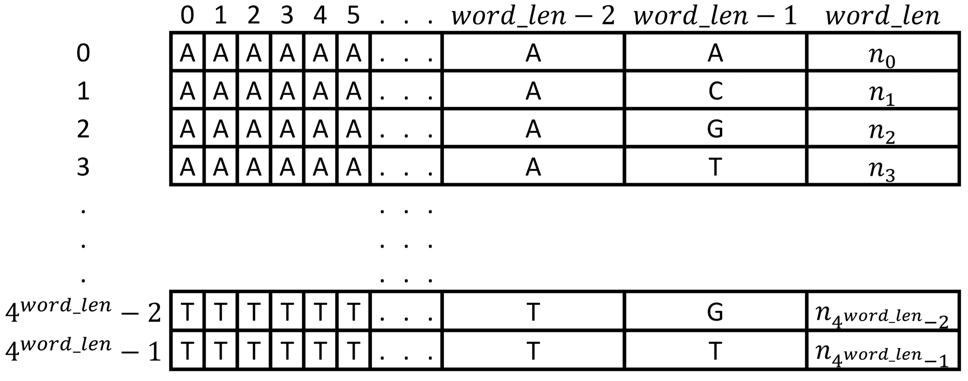
## **6.3. Least Frequency Pattern Matching Algorithm**

Ο αλγόριθμος αντιστοίχισης μοτίβων ελάχιστης συχνότητας (LFPM) είναι μια βελτίωση του PAPM που είναι εξειδικευμένη για εφαρμογές DNA. Ωστόσο, ο αλγόριθμος LFPM μπορεί να επεκταθεί και σε άλλες εφαρμογές αντιστοίχισης προτύπων. Το LFPM είναι μια κατάλληλη επιλογή όταν πρέπει να αναζητηθούν πολλά μοτίβα στο κείμενο, αλλά όχι μια αποτελεσματική επιλογή για λίγα μοτίβα λόγω του υπερβολικού χρόνου. Για να μειώσει τον αριθμό των αναγνωρισμένων παραθύρων, το LFPM αναζητά μια λέξη χαμηλής συχνότητας του μοτίβου στο κείμενο. Με άλλα λόγια, το LFPM εστιάζει σε μια λέξη που αναμένεται να εμφανίζεται λιγότερο από άλλες λέξεις του μοτίβου που εμφανίζονται στο κείμενο [[2]](#two).

Στο πρώτο βήμα πριν από τη φάση προεπεξεργασίας, το LFPMυπολογίζει τη συχνότητα όλων των πιθανών λέξεων για τις σχετικές αλφάβητο. Αν και οι υπολογισμοί αυτού του βήματος δημιουργούν ένα γενικό χρόνο, μειώνει δραματικά το κόστος χρόνου των

επερχόμενων φάσεων. Για να υπολογιστεί η συχνότητα των διαφορετικών λέξεων, το τρέχον κείμενο ή ένα σχετικό σύνολο δεδομένων μπορεί να επιλεγεί ως αναφορά. Εδώ, το ανθρώπινο γονιδίωμα αναφοράς (HRG) εφαρμόζεται ως αναφορά για την καταμέτρηση όλων των πιθανών λέξεων που έχουν το μήκος του word\_len πάνω από το πεπερασμένο αλφάβητο Σ= {A, C , G, T }. Οι τέσσερις νουκλεοτιδικές βάσεις σε ένα μόριο DNA είναι η Αδενίνη,η Γουανίνη, η Κυτοσίνη και η Θυμίνη. Επειδή το word\_len είναι το μήκος μιας λέξης σε έναν συγκεκριμένο υπολογιστή, ο αριθμός όλων των πιθανών λέξεων πάνω από το αλφάβητο Σ είναι 4world\_len [[2]](#two).

Αυτός ο αριθμός είναι σταθερός για όλους τους υπολογιστές b−bit. Για παράδειγμα, σε έναν υπολογιστή 32-bit, μπορούν να δημιουργηθούν 256 διαφορετικές λέξεις που έχουν μήκος τεσσάρων χαρακτήρων από το σχετικό αλφάβητο Σ. Η συχνότητα όλων των λέξεων αποθηκεύεται σε έναν πίνακα που ορίζεται από το freq\_table και έχει 4 σειρές word\_len και word\_len + 1 στήλες, όπως υποδεικνύεται στον Πίνακα 1. Σε κάθε σειρά, υπάρχει μια λέξη με το μήκος των χαρακτήρων word\_len στις στήλες από μηδέν έως τη λέξη \_len − 1. Επιπλέον, η συχνότητα κάθε λέξης εμφανίζεται στη στήλη word\_len αυτής της λέξης. Δεδομένου ότι το word\_len είναι είτε τέσσερα είτε οκτώ σε έναν υπολογιστή 32 ή 64 bit, αντίστοιχα, η ποσότητα μνήμης που απαιτείται για τη διατήρηση του πίνακα συχνοτήτων είναι λογική. Επομένως, αυτός ο πίνακας μπορεί να τοποθετηθεί στην κύρια μνήμη [[2]](#two).



Πίνακας 6.1: Ο freq\_table.

Εικόνα που περιέχει πίνακας

Περιγραφή που δημιουργήθηκε αυτόματα

Αλγόριθμος 6.4: Συμπλήρωση του πίνακα συχνοτήτων.

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει πίνακας

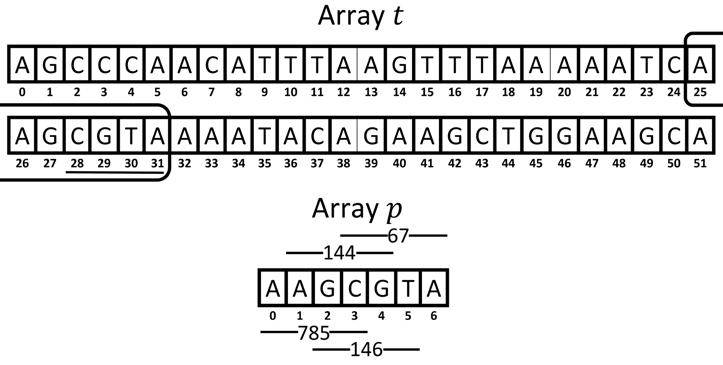
Περιγραφή που δημιουργήθηκε αυτόματα

Αλγόριθμος 6.5: Αλγόριθμος LFPM.

Για λεπτομέρειες, ο Αλγόριθμος 4 εξηγεί την εργασία πλήρωσης του freq\_table. Στις γραμμές 1 και 2, όλες οι πιθανές λέξεις με χαρακτήρες λέξης \_len πάνω από το αλφάβητο Σ αντιστοιχίζονται στις στήλες 0..word\_len − 1 όλων των σειρών του πίνακα. Η γραμμή 3 αρχικοποιεί τα κελιά της στήλης της word \_len κατά μηδέν. Στις γραμμές 5-19, ο εξωτερικός βρόχος επαναλαμβάνει τις οδηγίες για όλες τις λέξεις της αναφοράς. Από την άλλη πλευρά, ο εσωτερικός βρόχος while (Γραμμές 8-16) ταιριάζει με την τρέχουσα λέξη αναφοράς σε μία από τις λέξεις freq\_table [[2]](#two).

Όταν συμβεί αντιστοίχιση, η συχνότητα αυτής της λέξης, που είναι αποθηκευμένη στη στήλη word \_len, αυξάνεται κατά ένα (Γραμμή 17). Θα πρέπει να αναφερθεί ότι η συνολική αναζήτηση του πίνακα θα πρέπει να αποφευχθεί καθώς είναι μια χρονοβόρα εργασία. Αντίθετα, μπορεί να εφαρμοστεί μια ευρετική μέθοδος, που υλοποιείται από τις Γραμμές 8-16. Σε κάθε επανάληψη του εσωτερικού βρόχου while, ένας χαρακτήρας της λέξης συγκρίνεται με τα στοιχεία του Σ. Σύμφωνα με το αποτέλεσμα αυτής της σύγκρισης, το row\_count αυξάνεται. Αυτή η ενέργεια συνεχίζεται μέχρι τη στιγμή που θα επιτευχθεί η θέση της τρέχουσας λέξης στον πίνακα. Καθώς το μήκος του πίνακα αναφοράς παρουσιάζεται από το REF\_len και η σύγκριση βάσει λέξεων εκτελείται στο αλφάβητο Σ, η χρονική πολυπλοκότητα για τη συμπλήρωση του freq\_table είναι θ (word \_len × (REF \_len − word \_len + 1)). Αν και η συμπλήρωση του freq\_table είναι μια χρονοβόρα εργασία, αυτό εκτελείται μόνο μία φορά στα δεδομένα αναφοράς. Μετά την ολοκλήρωση, ο πίνακας μπορεί να χρησιμοποιηθεί για την αναζήτηση κάθε μοτίβου πάνω από το σχετικό αλφάβητο Σ. Θα πρέπει να σημειωθεί ότι ο πίνακας μπορεί να κατασκευαστεί για οποιοδήποτε αλφάβητο με βάση άλλα σύνολα δεδομένων ή κείμενο για άλλες εφαρμογές. Ωστόσο, σε εφαρμογές με μεγάλο αλφάβητο, το μέγεθος του πίνακα μπορεί να μεγεθύνεται. Σημειώστε ότι, εάν η αναφορά τροποποιηθεί με την πάροδο του χρόνου, ο πίνακας μπορεί να ενημερώνεται περιοδικά [[2]](#two).

Όποτε το freq\_table είναι έτοιμο, μπορεί να ξεκινήσει η φάση προεπεξεργασίας. Στην αρχή αυτής της φάσης (Γραμμές 1-21 του Αλγορίθμου 5), η λιγότερο συχνή λέξη στο μοτίβο πρέπει να καθοριστεί χρησιμοποιώντας τον γεμάτο freq\_table. Επομένως, σε κάθε επανάληψη, η τρέχουσα συχνότητα λέξης του μοτίβου λαμβάνεται από τον πίνακα. Στη συνέχεια, εάν αυτή η συχνότητα είναι αυτήν τη στιγμή η χαμηλότερη, ο δείκτης έναρξης αυτής της λέξης και η τιμή της αποθηκεύονται για συγκρίσεις με αυτές άλλων λέξεων στο μοτίβο (Γραμμές 16-19). Η χρονική πολυπλοκότητα για τον προσδιορισμό της λιγότερο συχνής λέξης είναι θ(word\_len × (m − word\_len)). Ωστόσο, καθώς το word\_len είναι σταθερό για κάθε υπολογιστή, αυτή η χρονική πολυπλοκότητα μπορεί να δηλωθεί ως θ (m). Στο δεύτερο βήμα της φάσης προεπεξεργασίας, παρόμοια με το PAPM, το κείμενο σαρώνεται για να ανακαλύψει τα παράθυρα βρίσκοντας τη λιγότερο συχνή λέξη (Βλ. Γραμμές 22-31). Στη συνέχεια προτείνεται η φάση αντιστοίχισης (Γραμμές 32-54) του Αλγορίθμου 5, η οποία είναι παρόμοια με αυτή του Αλγορίθμου 3 [[2]](#two).



Εικόνα 6.3: ΣΧΗΜΑ 3. Ένα παράδειγμα για τη λειτουργία του αλγορίθμου LFPM.

Για παράδειγμα, το Σχ. 3 απεικονίζει τη λειτουργία του LFPM σε έναν υπολογιστή 32 bit. Υποτίθεται ότι το freq\_table είναι έτοιμο από πριν. Επομένως, η συχνότητα όλων των λέξεων στο μοτίβο μπορεί να εξαχθεί από τον πίνακα ως AAGC: 785, AGCG: 144, GCGT: 146 και CGTA: 67. Ως αποτέλεσμα, η λέξη CGTA επιλέγεται ως λέξη που έχει το λιγότερο συχνότητα. Στη συνέχεια, ξεκινά η αναζήτηση για να συναντήσει το CGTA στο κείμενο. Ο δείκτης έναρξης του σχετικού παραθύρου είναι μόνο 25. Καθώς η λιγότερο συχνή λέξη αναζητείται στο κείμενο, ο αριθμός των παραθύρων που χρησιμοποιούν LFPM είναι συχνά μικρότερος από ό,τι στους αλγόριθμους που παρουσιάστηκαν προηγουμένως. Αφού διακρίνει τα παράθυρα, η φάση αντιστοίχισης ελέγχει την ομοιότητα των λέξεων μοτίβου με εκείνες των παραθύρων. Σε αυτήν την περίπτωση, οι λέξεις AAGC και GTA, αντίστοιχα, συγκρίνονται με την πρώτη και τη δεύτερη λέξη του παραθύρου. Αυτή η σύγκριση αποκαλύπτει ότι το αναγνωρισμένο παράθυρο στον δείκτη 25 είναι η απάντηση. Κατά συνέπεια, η υπεροχή του LFPM είναι η προσέγγιση λέξεων χαμηλότερης συχνότητας που μειώνει τρομερά τον αριθμό των παραθύρων που πρέπει να ελεγχθούν στην επόμενη φάση [[2]](#two).

# **7. Συμπεράσματα**

## **7.1. Σύγκριση**

Εδώ συγκρίνεται η απόδοση των παρουσιαζόμενων αλγορίθμων (FLPM, PAPM και LFPM) με αυτή των αλγορίθμων Brute Force (BF), Boyer-Moore (BM) και Divide and Conquer Pattern Matching (DCPM). Οι προδιαγραφές του υπολογιστικού περιβάλλοντος για την εκτέλεση διαφορετικών προσομοιωμένων αλγορίθμων ήταν οι εξής [[2]](#two):

• Intell®core™2 Duo CPU T6600 (a 2.2 GHz clock)

• 2GB Memory

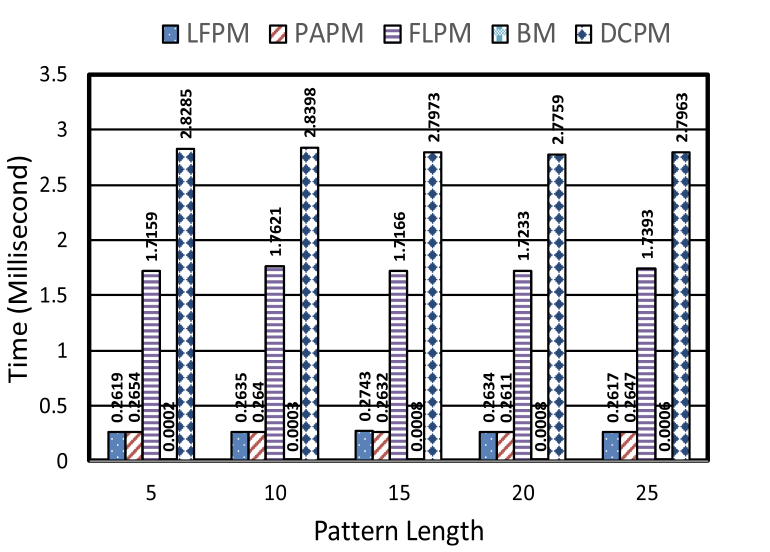
• Acer (Aspire 5738)

• Windows 7 ultimate 32 bit

Λόγω της χρήσης ενός υπολογιστή 32-bit, το μήκος λέξης για τους αλγόριθμους PAPM και LFPM θεωρήθηκε ως τέσσερα byte. Στο LFPM, το HRG χρησιμοποιήθηκε ως αναφορά για την κατασκευή του πίνακα συχνοτήτων. Η προσομοίωση έγινε με τη γλώσσα προγραμματισμού C. Σε κάθε πείραμα, δέκα μοτίβα αναζητήθηκαν στην αναφορά και στη συνέχεια αναφέρθηκε ο μέσος όρος των αποτελεσμάτων. Θα πρέπει να σημειωθεί ότι το LFPM έχει επιβάρυνση χρόνου για την κατασκευή του πίνακα συχνοτήτων. Καθώς το σύνολο δεδομένων HRG χρησιμοποιήθηκε για όλα τα πειράματα, αυτή η επιβάρυνση του χρόνου καθορίστηκε κατά τη διάρκεια της προσομοίωσης. Το ποσό των γενικών εξόδων ήταν 12 χιλιοστά του δευτερολέπτου. Ωστόσο, αυτή η επιβάρυνση χρόνου αγνοήθηκε στους υπολογισμούς επειδή ο πίνακας συχνότητας δημιουργείται μόνο μία φορά για οποιοδήποτε κείμενο ή βάση δεδομένων που έχει επεξεργαστεί το LFPM. Τα αποτελέσματα της αξιολόγησης της απόδοσης των προσομοιωμένων αλγορίθμων στη φάση προεπεξεργασίας, τη φάση αντιστοίχισης και τις συνολικές φάσεις ως προς το κόστος χρόνου παρουσιάζονται στη συνέχεια [[2]](#two).

### **7.1.1. Χρόνος φάσης προεπεξεργασίας**

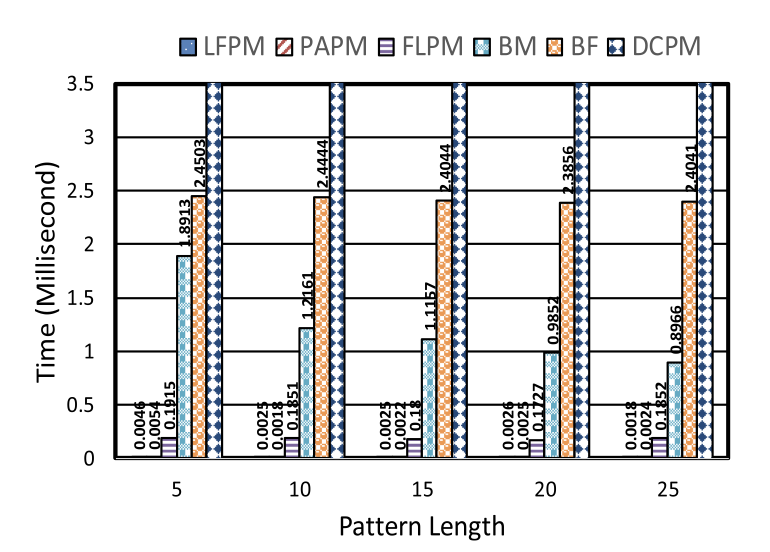
Η παρακάτω εικόνα απεικονίζει το χρόνο της φάσης προεπεξεργασίας για διαφορετικούς προσομοιωμένους αλγόριθμους στο μήκος του σχεδίου. Πρέπει να σημειωθεί ότι ο χρόνος του BM είναι αμελητέος καθώς το μοτίβο αναλύεται μόνο κατά τη φάση προεπεξεργασίας αυτού του αλγορίθμου. Αυτό το σχήμα αποκαλύπτει την κυριαρχία των αλγορίθμων της παρούσας μελέτης έναντι των άλλων αλγορίθμων. Στο FLPM, ένα πέρασμα στο κείμενο είναι αρκετό για να ανακαλύψετε ταυτόχρονα τον πρώτο και τον τελευταίο χαρακτήρα του μοτίβου. Αντίθετα, το DCPM απαιτεί δύο περάσματα: ένα για την αναζήτηση του πιο αριστερού χαρακτήρα του μοτίβου και ένα άλλο για το δεξιότερο. Το PAPM αναζητά την πρώτη λέξη του μοτίβου στο κείμενο, για παράδειγμα, τους τέσσερις πρώτους χαρακτήρες σε έναν υπολογιστή 32 bit. Επομένως, η φάση προεπεξεργασίας PAPM χρειάζεται επίσης ένα πέρασμα. Όπως φαίνεται στο Σχ. 4, αναμένεται ότι το PAPM βρίσκει λιγότερα παράθυρα από το FLPM, επειδή εξετάζει μια λέξη με μερικούς χαρακτήρες για να αναγνωρίσει τα παράθυρα, ενώ το FLPM εστιάζει μόνο σε δύο χαρακτήρες (τον πρώτο και τον τελευταίο). Επειδή το LFPM αναζητά τη λιγότερο συχνή λέξη του μοτίβου, βρίσκει λιγότερα παράθυρα. Ωστόσο, η εύρεση της λιγότερο συχνής λέξης από τον πίνακα συχνοτήτων στο LFPM είναι χρονοβόρα. Ως αποτέλεσμα, το PAPM και το LFPM καταναλώνουν τον λιγότερο χρόνο μεταξύ των προσομοιωμένων αλγορίθμων στο Σχ. 4. Είναι αξιοσημείωτο ότι το BF έχει αφαιρεθεί από αυτό το σχήμα καθώς δεν έχει καμία φάση προεπεξεργασίας [[2]](#two).



Εικόνα 7.1: Χρονικές συγκρίσεις στη φάση προεπεξεργασίας.

### **7.1.2. Χρόνος φάσης αντιστοιχίας**

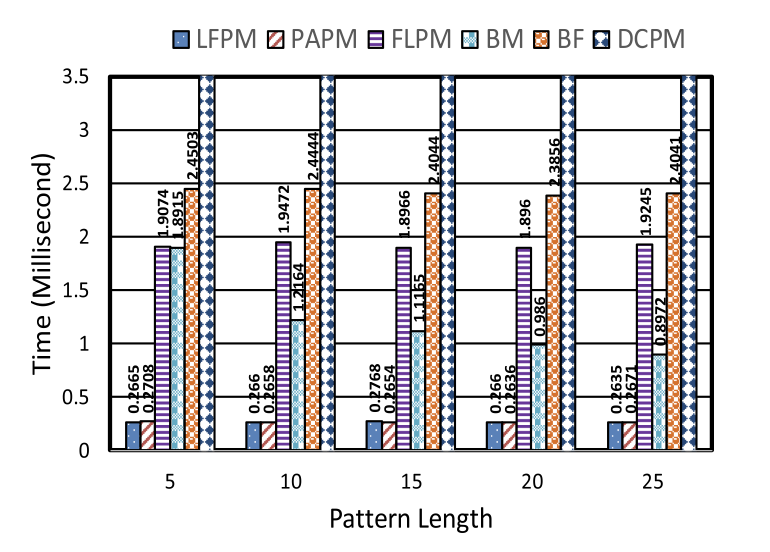
Η παρακάτω εικόινα παρουσιάζει το κόστος χρόνου της φάσης αντιστοίχισης για τους αλγόριθμους. Επειδή το DCPM συγκρίνει κάθε στοιχείο στον πιο δεξιό πίνακα με όλα τα στοιχεία του πιο αριστερού πίνακα, η φάση αντιστοίχισής του είναι πολύ χρονοβόρα. Έτσι, η γραφική παράσταση DCPM έχει κοπεί στα 3,5 χιλιοστά του δευτερολέπτου. Όπως φαίνεται, τα FLPM, PAPM και LFPM έχουν καλύτερη απόδοση από τους άλλους αλγόριθμους. Ο κύριος λόγος της υπεροχής τους είναι γιατί προσδιορίζουν χαμηλό αριθμό παραθύρων στην προηγούμενη φάση. Επιπλέον, με τη χρήση επεξεργασίας κειμένου, η έρευνα των PAPM και LFPM για την αντιστοίχιση των παραθύρων και του μοτίβου είναι πολύ πιο γρήγορη από αυτή των προσομοιωμένων αλγορίθμων που βασίζονται σε χαρακτήρες, π.χ., BF, DCPM, BM και FLPM [[2]](#two).



Εικόνα 7.2: Συγκρίσεις χρόνου σε φάση αντιστοίχισης.

### **7.1.3. Συνολικός χρόνος**

Η παρακάτω εικόνα παρέχει το άθροισμα του χρόνου κόστους για τις δύο φάσεις, όπως φαίνεται στα δύο προηγούμενα σχήματα. Αυτό το σχήμα αποκαλύπτει ότι η χρήση επεξεργασίας κειμένου από τα PAPM και LFPM μειώνει σημαντικά τον χρόνο που απαιτείται για την εκτέλεση της αντιστοίχισης προτύπων. Στο LFPM, υπάρχει διαφορά μεταξύ του αριθμού επανάληψης της λιγότερο συχνής λέξης και αυτών των άλλων λέξεων του μοτίβου. Η υψηλότερη τιμή αυτής της διαφοράς οδηγεί σε μεγαλύτερη βελτίωση στην απόδοση του LFPM [[2]](#two).



Εικόνα 7.3: Συνολικές συγκρίσεις χρόνου.

## **7.2. Συμπεράσματα**

Η τρέχουσα εργασία εισάγει τρεις νέους αλγόριθμους: FLPM, PAPM και LFPM. Παρόμοια με προηγούμενα έργα, το LFPM είναι ένας αλγόριθμος αντιστοίχισης μοτίβων που βασίζεται σε χαρακτήρες, ενώ ο PAPM και ο LFPM λειτουργούν με βάση την προσέγγιση επεξεργασίας κειμένου. Επιπλέον, το LFPM αναζητά τη λέξη χαμηλότερης συχνότητας του μοτίβου προκειμένου να ελαχιστοποιήσει τον χρόνο εκτέλεσης του αλγορίθμου. Τα πειραματικά αποτελέσματα της παρούσας εργασίας αποκαλύπτουν ότι οι προτεινόμενοι αλγόριθμοι, ιδιαίτερα ο LFPM, ξεπερνούν τους άλλους προσομοιωμένους αλγόριθμους ως προς το κόστος χρόνου. Αυτή η βελτίωση οφείλεται κυρίως στη μείωση του αριθμού των παραθύρων που βρέθηκαν [[2]](#two).

# **8. Βιβλιογραφία**

[1] Santiago Marco-Sola, (2016). *Efficient Approximate String-Matching Techniques for Sequence Alignment* (Διπλωματική Εργασία). Polytechnic University of Catalonia.

[2] Neamatollahi, P., Hadi, M., & Naghibzadeh, M. (2020). Simple and Efficient Pattern Matching Algorithms for Biological Sequences. *IEEE Access 8*, p.p. 23838-23846. Ανακτήθηκε από: <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8967097>

[3] Wang, Q., Korkin, D., & Shang, Y. (2009). Efficient Dominant Point Algorithms for the Multiple Longest Common Subsequence (MLCS) Problem. California, USA, p.p. 1494-1499. Ανακτήθηκε από: <https://www.ijcai.org/Proceedings/09/Papers/250.pdf>

[4] RAZA, K. (2012). Application of data mining in Bioinformatics. *Indian Journal of Computer Science and Engineering 1(*2), p.p. 114-118. Ανακτήθηκε από: <https://arxiv.org/ftp/arxiv/papers/1205/1205.1125.pdf>

[5] Holzinger, A., Dehmer, M., & Jurisica, I. (2014). Knowledge Discovery and interactive Data Mining in Bioinformatics - State-of-the-Art, future challenges and research directions. *BMC Bioinformatics 15*(6). Ανακτήθηκε από: <https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-15-S6-I1>