ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
"ΤΕΧΝΟΛΟΓΙΕΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΣΤΗΝ ΙΑΤΡΙΚΗ ΚΑΙ ΤΗ ΒΙΟΛΟΓΙΑ"

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Υπολογιστική μέθοδος για την κατασκευή και αξιολόγηση του UNIQUOME ενός οργανισμού.

Βασίλειος Α. Πιέρρος

Επιβλέπων: **Γεώργιος Τσάγκαρης,** Ειδικός Λειτουργικός Επιστήμονας (βαθμίδα Α΄)

ΑΘΗΝΑ

ΙΟΥΝΙΟΣ 2018

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Υπολογιστική μέθοδος για την κατασκευή και αξιολόγηση του UNIQUOME ενός οργανισμού.

Βασίλειος Α. Πιέρρος

Α.Μ.: ΠΙΒ0146

ΕΠΙΒΛΕΠΩΝ: **Δρ. Γεώργιος Τσάγκαρης,** Ειδικός Λειτουργικός Επιστήμονας (βαθμίδα Α΄)

ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ **Τσάγκαρης Γεώργιος,** Ειδικός Λειτουργικός Επιστήμονας (βαθμίδα Α΄)
Αναστασιάδου Έμα, Ερευνήτρια (βαθμίδα Δ')
Αναγνωστόπουλος Αθανάσιος, Συνεργαζόμενος Ερευνητής

ΙΟΥΝΙΟΣ 2018

ΠΕΡΙΛΗΨΗ

Παλαιότερη μελέτη είχε προσδιορίσει τις μικρότερες αμινοξικές ακολουθίες που μπορούσαν να βρεθούν μόνο σε μια πρωτεΐνη, ορίζοντας έτσι το Uniquome ενός συνόλου πρωτεϊνών. Για να είναι ένα πεπτίδιο μοναδικό δεν πρέπει να εμφανίζεται σε καμία άλλη πρωτεΐνη εντός της ομάδας στην οποία γίνεται η αναζήτηση. Ως uniquome λοιπόν ορίζεται ως το σύνολο των μικρότερων αμινοξικών ακολουθιών που μπορούν να βρεθούν σε μία μόνο πρωτεΐνη και να την χαρακτηρίσουν μοναδικά εντός του συνόλου πρωτεϊνών (πρωτεώματος) το οποίο μελετούμε.

Το ανθρώπινο πρωτέωμα αποτελείται από περισσότερες από 20.000 πρωτεΐνες. Ο διαχωρισμός του αυτών σε πεπτίδια μεγέθους 4 έως 100 αμινοξέων δημιουργεί ένα πεδίο αναζήτησης περίπου 500 $x10^6$ πεπτιδίων. Από τα παραπάνω γίνεται εύκολα αντιληπτό ότι η δημιουργία του uniquome είναι μια αρκετά απαιτητική διαδικασία τόσο σε υπολογιστικό χρόνο όσο σε μνήμη και σε αποθηκευτικό χώρο

Στα πλαίσια της εργασίας αυτής αναπτύχθηκε εργαλείο σε C# το οποία μπορεί να δέχεται ως είσοδο μια ομάδα πρωτεϊνών (πχ το πρωτέωμα ενός οργανισμού) και να κατασκευάζει το Uniquome της συγκεκριμένης ομάδας. Το παραπάνω εργαλείο κάνει χρήση παράλληλων και κατανεμημένων μεθόδων ώστε να πετύχει καλύτερη διαχείριση των διαθέσιμων πόρων. Ο χρήστης μπορεί να θέσει τις επιθυμητές παραμέτρους ή να αφήσει το εργαλείο να προτείνει κατάλληλο τρόπο διαχωρισμού του πρωτεώματος, ανάλογα με το διαθέσιμο υλικό. Ο χρήστης μπορεί επιπλέον να αν επιθυμεί τον τρόπο με τον οποίο θέλει να ομαδοποιήσει τα δεδομένα (π.χ. ανά οικογένεια πρωτεϊνών). Τέλος το εργαλείο μπορεί να εκτελέσει διάφορες χρήσιμες μετά-αναλύσεις στα παραγόμενα αποτελέσματα καθώς και να εκτελέσει PERL κώδικα για περαιτέρω ανάλυση.

Το εργαλείο κατασκεύασε και αποθήκευσε το ανθρώπινο uniquome σε ένα φορητό υπολογιστή i7 με 12 πυρήνες και 32GB μνήμης σε λιγότερο από το 30 % του χρόνου που χρειαζόταν η προηγούμενη εφαρμογή του αλγορίθμου σε PERL. Η υλοποίηση έχει γίνει σε .NET Core με σκοπό να μπορεί να εκτελεστεί στα κυριότερα λειτουργικά συστήματα (Windows, Linux, MacOS) και σε μια πληθώρα διαφορετικού υλικού από SOC (System on a Chip) όπως το Raspberry Pi μέχρι μεγάλους διακομιστές (servers) με δεκάδες επεξεργαστές.

Με την χρήση αυτού του εργαλείου, πιστεύουμε πως οι ερευνητές μπορούν σχετικά γρήγορα να κατασκευάσουν και να αξιολογήσουν το uniquome κάθε οργανισμού. Το εργαλείο μπορεί να εξάγει τα δεδομένα σε αρκετές διαφορετικές μορφές όπως αρχεία κειμένου, FASTA, CSV, JSON.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Βιοπληροφορική

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: πρωτεωμική, παράλληλη επεξεργασία, κατανεμημένη επεξεργασία, πεπτίδιο μοναδικής ελάχιστης ακολουθίας, πρωτεΐνες .

ABSTRACT

In a previous study we had identified the shortest aminoacids appearing solely in one protein, thus defining the uniquome of a set of proteins or proteome. For a peptide to be unique, it should not appear again in the set of proteins (except those coming from the same protein).

Human proteome consists of more than 20,000 reviewed proteins. The separation of human proteome in distinct peptides range from 4 to 100 aminoacids creates a search space of around 500 x 10^6 peptides. That renders the construction of the uniquome a quite expensive, in resource usage problem, in terms of CPU, RAM and storage. In this study we present the tools we developed for the construction and evaluation of the uniquome.

A tool suite was developed in C# that implemented the uniquome construction algorithm. It is highly scalable since it can work in parallel, distributed or hybrid mode, based on the underlying hardware. User can either set the parameters, or let the tool propose a data separation strategy based on the hardware. User can also decide (if any) the method in which the data should be grouped together (i.e. protein family). Finally, the tool can perform various meta data analysis in the created data and can be configured to run scripts in PERL for further knowledge extraction.

Our tool has constructed and stored the human uniquome in an i7 laptop with 8 cores and 32Gb of RAM in less than 30 % of the time needed by the previous PERL implementation. The developed tools were implemented in .NET Core so that they can be executed in all mainstream Operating Systems (Windows, Linux, MacOS) and run in a wide variety of hardware ranging from a SOC (like Raspberry) to high end servers with tens of CPUs.

With the use of this suite researchers can relatively quickly construct and evaluate the uniquome of any organization. Tool offers a variety of export formats such as Text, FASTA, CSV, JSON.

SUBJECT AREA: Bioinformatics

KEYWORDS: proteomics, parallel computation, distributed computation, core unique peptide, proteins.

Είμαι ένα πλάσμα εφήμερο, αδύναμο, καμωμένο από λάσπη κι ονείρατα.
Μα μέσα μου νογώ να στροβιλίζουνται όλες οι δυνάμες του Σύμπαντου.
Θέλω μια στιγμή, προτού με συντρίψουν, ν΄ ανοίξω τα μάτια μου και να τις δω.
Άλλο σκοπό δε δίνω στη ζωή μου.

(Νίκος Καζαντζάκης – Ασκητική)

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω ιδιαίτερα τον Δρ. Τσάγκαρη Γιώργο και την Δρ. Αναστασιάδου Έμα για την στενή συνεργασία, την μεγάλη τους υπομονή καθώς και την στήριξή τους σε όσα ακαδημαϊκά θέματα προέκυψαν μέχρι την ολοκλήρωση της εκπόνηση της συγκεκριμένης εργασίας και του συνόλου της φοίτησής μου στο μεταπτυχιακό πρόγραμμα σπουδών.

Τον Δρ. Αναγνωστόπουλο Αθανάσιο για τις σημαντικές παρατηρήσεις του ειδικά σε ότι αφορά τα επόμενα βήματα για την εξέλιξη της παρούσας εργασίας.

Τον Δρ. Σπύρου Γιώργο για τις πολύτιμες συμβουλές και κατευθύνσεις του στα πρώτα στάδια της ανάπτυξης του αλγορίθμου και την συμπαράστασή του μέχρι την ολοκλήρωση αυτού.

Τον υποψήφιο διδάκτορα Κοντοπόδη Ευάγγελο για την εξαιρετική μας συνεργασία τα τελευταία χρόνια.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΑΤΑΛΟΓΟΣ ΓΡΑΦΗΜΑΤΩΝ....12

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ....13

ΠΡΟΛΟΓΟΣ....14

1. ΖΩΗ....15

2. ΒΙΟΛΟΓΙΚΟ ΥΠΟΒΑΘΡΟ....18

2.1 Αμινοξέα....18

2.2 Πεπτίδια....23

2.3 Πρωτεΐνες....24

2.4 Πρωτεωμική....26

3. ΤΑΥΤΟΠΟΙΗΣΗ ΠΡΩΤΕΪΝΩΝ....27

3.1 MS/MS Spectrometry....27

3.2 Βιολογικά Δεδομένα και Δημόσιες Βάσεις Δεδομένων....30

3.3 Uniquome....32

4. ΥΠΟΛΟΓΙΣΤΙΚΟ ΥΠΟΒΑΘΡΟ....34

4.1 Επίλυση προβλημάτων - Αλγόριθμος....34

4.2 Βασικές αρχές επεξεργασίας....34

4.3 Παράλληλη Επεξεργασία....36

4.4 Κατανεμημένα Συστήματα....40

5. ΣΤΟΧΟΣ ΤΗΣ ΕΡΓΑΣΙΑΣ....43

5.1 Προηγούμενες εργασίες....43

5.2 Προβλήματα και προκλήσεις....44

5.3 Νέοι στόχοι....45

6. ΑΚΟΛΟΥΘΙΑΚΟΣ ΑΛΓΟΡΙΘΜΟΣ....47

6.1 Περιγραφή Ακολουθιακού Αλγορίθμου (Sequential Algorithm)....47

6.2 Έκταση Προβλήματος....48

6.3 Μειονεκτήματα Ακολουθιακού Αλγορίθμου....48

7. ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΛΕΙΤΟΥΡΓΙΩΝ ΕΝΤΟΣ ΤΟΥ ΙΔΙΟΥ ΝΗΜΑΤΟΣ ΕΚΤΕΛΕΣΗΣ (SINGLE THREAD OPTIMIZATIONS) 49
7.1 Βελτιστοποίηση αναζήτησης Πεπτιδίου σε Πρωτεΐνη 49
7.2 Αποκλεισμός Πεπτιδίων από την αναζήτηση αν είναι ή όχι CUP 50
8. ΠΑΡΑΛΛΗΛΗ ΕΚΤΕΛΕΣΗ ΛΕΙΤΟΥΡΓΙΩΝ 52
8.1 Βήμα 1 – Χωρισμός του πρωτεώματος 52
8.2 Βήμα 2 – Εκτέλεση 53
8.3 Λειτουργία Master 54
8.4 Πλεονεκτήματα παράλληλης προσέγγισης 55
8.5 Μειονεκτήματα παράλληλης προσέγγισης 55
8.6 Απόδοση Παράλληλου αλγορίθμου 56
9. ΚΑΤΑΜΕΡΙΣΜΕΝΗ ΕΠΕΞΕΡΓΑΣΙΑ ΣΕ ΠΟΛΛΑΠΛΑ ΣΥΣΤΗΜΑΤΑ 58
9.1 Πλεονεκτήματα - Μειονεκτήματα 59
10. ΜΕΤΑΝΑΛΥΣΗ 60
10.1 Κατασκευή Composite Unique Peptides 60
10.2 Εύρεση σχετικής θέσης εμφάνισης Core Unique Peptide 60
10.3 Στατιστικά πρωτεϊνών 61
10.4 Κοινά CUP μεταξύ οργανισμών 61
11. ΣΤΟΙΧΕΙΑ ΥΛΟΠΟΙΗΣΗΣ 62
11.1 Περιβάλλον ανάπτυξης 62
11.2 Αποθήκευση Δεδομένων 63
11.3 Τεχνικά Χαρακτηριστικά Υλοποίησης 64
12. ΑΠΟΤΕΛΕΣΜΑΤΑ 69
12.1 Αναλύσεις Uniquome 69
12.2 Ανάλυση Human – Rat – Mouse 69
12.3 Σύγκριση κοινών CUP Human – Rat – Mouse 71
13. ΕΠΟΜΕΝΑ ΒΗΜΑΤΑ – ΕΠΕΚΤΑΣΕΙΣ 73
13.1 Επιτάχυνση διαδικασίας μέσω χρήσης GPU (CUDA – OpenGL) 73
13.2 Χρήση εργαλείων αναζήτησης κειμένων 73

13.3 Cloud – Microservices 73

13.4 Δημιουργία μικρής συστοιχίας με SoC 74

13.5 Δημιουργία Web Server 74

14. ΣΥΜΠΕΡΑΣΜΑΤΑ 75

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ 76

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ 77

ΑΝΑΦΟΡΕΣ 78

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1: Καλλιτεχνική απεικόνιση της νεαρής γης 15
Εικόνα 2: Υδροθερμικές πηγές στον πάτο του ωκεανού.....16
Εικόνα 3: Η εξέλιξη των οργανισμών.....17
Εικόνα 4: Γενικός συντακτικός τύπος α-αμινοξέων.....18
Εικόνα 5: Μετάφραση κωδικωνίων σε Αμινοξέα.....19
Εικόνα 6: Ο γενετικός κώδικας.....19
Εικόνα 7: Κατοπτρική μορφή D- / L- αμινοξέων.....19
Εικόνα 8: Ομαδοποίηση α-αμινοξέων βάση της πολικότητάς τους.....21
Εικόνα 9: Σχηματισμός διπεπτιδίου με πεπτιδικό δεσμό και αποβολή ενός μορίου νερού 23
Εικόνα 10: Παραγωγή πρωτεΐνης από DNA.....24
Εικόνα 11: Πίνακας αντιστοίχισης 3αδων βάσεων με αμινοξέα 24
Εικόνα 12: Πρωτεϊνικές δομές.....25
Εικόνα 13: Ηλεκτροφόρηση 2 διαστάσεων 27
Εικόνα 14: Ανάλυση πρωτεΐνης με χρήση MS/MS φασματογραφίας.....28
Εικόνα 15: m/z προς Ένταση γράφημα 28
Εικόνα 16: Ανάλυση Πεπτιδίων (Μέρος Β προηγούμενης εικόνας) 29
Εικόνα 17: Ρυθμός αύξησης δεδομένων αλληλούχισης DNA 30
Εικόνα 18: Σύγκριση δεδομένων αλληλούχισης DNA με άλλες πλατφόρμες περιεχομένου.....30
Εικόνα 19: Δημιουργία Σύνθετου Μοναδικού Πεπτιδίου από τα αντίστοιχα Ελάχιστου Μήκους Μοναδικά.....32
Εικόνα 20: Παράδειγμα εκτέλεσης πολλών προγραμμάτων σε μονοπύρηνο επεξεργαστή με ένα νήμα μέσω διαμοιρασμού χρόνου. 35
Εικόνα 21: Intel Knights Landing Many core CPU (72 cores x 4 threads) 36
Εικόνα 22: Παραδείγματα 2-πύρηνων επεξεργαστών με 1 (N=2) και 2 (N=4) νήματα ανά πυρήνα αντίστοιχα.....37
Εικόνα 23: Νόμος του Amdahl.....38
Εικόνα 24: Ανταλλαγή μηνυμάτων σε παράλληλο σύστημα 40
Εικόνα 25: Ανταλλαγή μηνυμάτων σε κατανεμημένο σύστημα.....40
Εικόνα 26: Παράδειγμα αρχιτεκτονικής Client – Server 41
Εικόνα 27: Παράδειγμα αρχιτεκτονικής Πολλαπλών Επιπέδων (N-Tier).....41
Εικόνα 28: Παράδειγμα αρχιτεκτονικής ομότιμων λ κόμβων (Peer to Peer) 42
Εικόνα 29: Web Interface του BioServer 43
Εικόνα 30: Windows Based Interface για την εφαρμογή Protein Analysis.....44
Εικόνα 31: Ανάλυση CUP ανά Αμινοξύ στην εφαρμογή Protein Analysis.....44
Εικόνα 32: Προσπέλαση πρωτεΐνης για ανεύρεση Core Unique Peptide 47

Εικόνα 33: Τμήμα χαρτογράφησης αμινοξέων πρωτεΐνης P0AAW9 49
Εικόνα 34: Παράδειγμα αποκλεισμού πεπτιδίων από την αναζήτηση 51
Εικόνα 35: Χωρισμός του πρωτεώματος σε N υποσύνολα.....52
Εικόνα 36: Αποστολή Περιεχομένου Ερώτησης από τον Master στους Search Buckets 54
Εικόνα 37: Αναζήτηση πεπτιδίου εντός του Search Bucket.....55
Εικόνα 38: Χρόνος κατασκευής Uniquome (sec).....56
Εικόνα 39: Ποσοστιαία βελτίωση απόδοσης της κατασκευής του Uniquome 56
Εικόνα 40: Καταμερισμός αναζήτησης σε περισσότερα συστήματα.....58
Εικόνα 41: Κατασκευή Composite Unique Peptide από τα αντίστοιχα CUP. 60
Εικόνα 42: Οικοσύστημα .NET Core Framework.....63
Εικόνα 43: Αρχιτεκτονική Εφαρμογής.....65
Εικόνα 44: Παραμετροποίηση εφαρμογής.....66
Εικόνα 45: Αρχείο Παραμετροποίησης εφαρμογής σε JSON. 66
Εικόνα 46: Παράδειγμα εκτέλεσης κατασκευής Uniquome για τον οργανισμό E-coli.....67
Εικόνα 47: Παράδειγμα εκτέλεσης κατασκευής Composite Unique Peptides για το ποντίκι 67
Εικόνα 48: Σχήμα Εφαρμογής 68

ΚΑΤΑΛΟΓΟΣ ΓΡΑΦΗΜΑΤΩΝ

Γράφημα 1: Εξέλιξη της αύξησης απόδοσης βάσει του νόμου του Amdahl 38
Γράφημα 2: Απόδοση Αλγορίθμου σε σχέση με την εκτέλεση σε 1 thread. 57
Γράφημα 3: Κατανομή CUP ανά μέγεθος πεπτιδίου και στους 3 οργανισμούς. 70

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1: α-αμινοξέα.....20
Πίνακας 2: Ομαδοποίηση α-αμινοξέων με βάση την υδροφοβικότητά τους.....22
Πίνακας 3 : Πρωτεϊνικές Βάσεις Δεδομένων 31
Πίνακας 4: Ορολογίες που θα χρησιμοποιηθούν στο υπόλοιπο κείμενο 33
Πίνακας 5: Άδειες χρήσης λογισμικού.....62
Πίνακας 6:Core Unique Peptides ανά Οργανισμό 69
Πίνακας 7:Ανάλυση Uniquome των 3 οργανισμών 70
Πίνακας 8:Πίνακας κατανομής ανά μήκος πεπτιδίου και για τους 3 οργανισμούς.....70
Πίνακας 9:Κοινά CUP ανάμεσα στους οργανισμούς.....71
Πίνακας 10: Ανάλυση σχετικής θέσης των κοινών CUP στους 3 οργανισμούς.....72
Πίνακας 11:Διαφορές στην σχετική θέση εμφάνισης ων CUP μεταξύ των 3 οργανισμών 72

ΠΡΟΛΟΓΟΣ

Η παρούσα διπλωματική εργασία εκπονήθηκε μεταξύ του διαστήματος Ιούλιος 2016 – Ιούλιος 2018 με την συνεργασία του Ινστιτούτου Ιατρικών και Βιολογικών Ερευνών της Ακαδημίας Αθηνών (ΙΙΒΕΑΑ) και του τμήματος Πληροφορικής και Τηλεπικοινωνιών του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών. Εκπονήθηκε στα πλαίσια της ολοκλήρωσης των σπουδών μου στο Πρόγραμμα Μεταπτυχιακών Σπουδών στην Ιατρική και την Βιολογία (κατεύθυνση Βιοπληροφορικής).

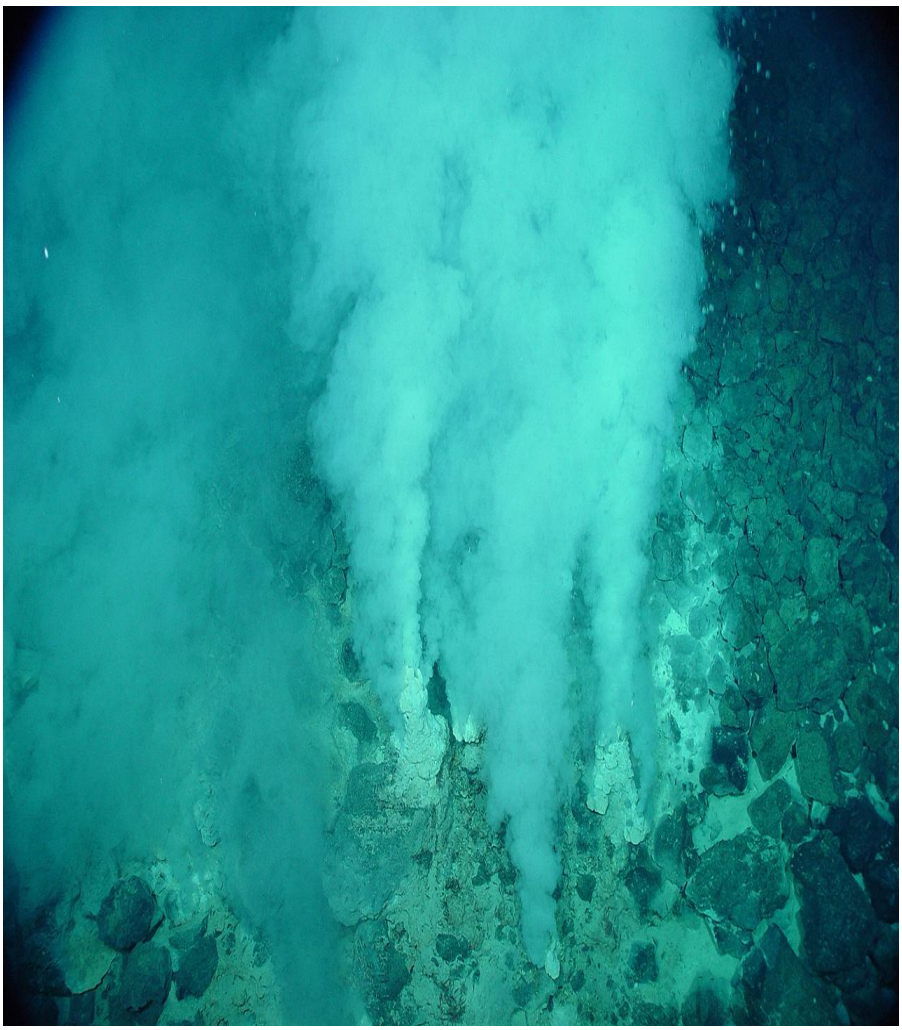
1. ΖΩΗ

Σύμφωνα με την επικρατούσα επιστημονική άποψη, ένα σημαντικό γεγονός συνέβη πριν από 13.8 δισεκατομμύρια χρόνια. Το πρώτο γεγονός. Η μεγάλη έκρηξη (Big Bang) η οποία έδωσε υπόσταση στον χώρο και τον χρόνο ενώ δημιούργησε το σύμπαν. Στις πρώτες του βρεφικές στιγμές το σύμπαν ήταν μια καυτή βροχή χαλασμάτων που ταξίδευαν με τεράστια ταχύτητα διαστέλλοντάς το. Σιγά σιγά η βαρύτητα άρχισε να φέρνει κοντά τα σωματίδια δημιουργώντας τα πρώτα ουράνια σώματα, τους αστεροειδείς τους πλανήτες, τους ήλιους. Το σύμπαν άρχισε να μοιάζει περισσότερο με αυτό που σήμερα παρατηρούμε, ενώ νέα βαρύτερα στοιχεία δημιουργήθηκαν.

Πριν από 4,5 δισεκατομμύρια χρόνια δημιουργήθηκε η γη. Στην αρχή ήταν ένα αφιλόξενο περιβάλλον. Ένας καυτός βράχος μετάλλων που στροβιλιζόταν με ταχύτητα γύρω από ένα μεσαίου μεγέθους άστρο. Η μέρα διαρκούσε μόλις 6 ώρες. Τρισεκατομμύρια χαλάσματα, κομμάτια της μεγάλης έκρηξης συγκρούονταν με τον καυτό αυτό βράχο. Κουβαλούσαν μαζί τους κρυστάλλους που μέσα τους έκρυβαν το μεγάλο συστατικό της ζωής, το νερό.

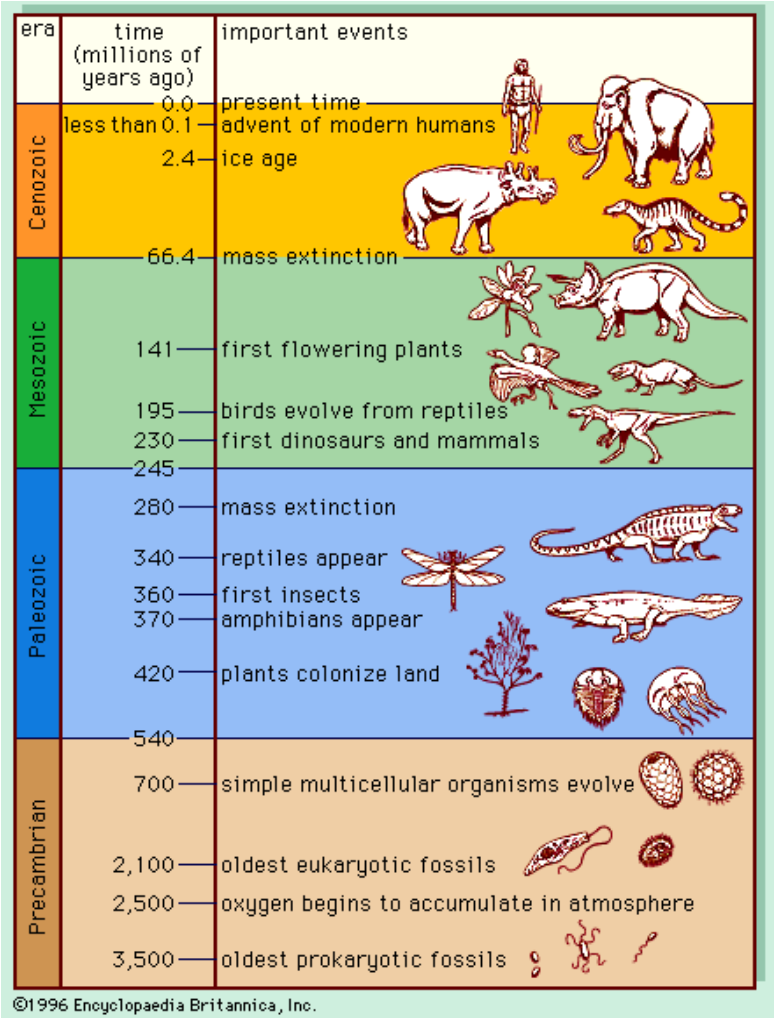
Εικόνα 1: Καλλιτεχνική απεικόνιση της νεαρής γης

Δεκάδες εκατομμύρια χρόνια μετά την δημιουργία της, η επιφάνεια της γης άρχιζε να παγώνει οδηγώντας στον σχηματισμό των πρώτων ωκεανών. Στο επόμενο διάστημα, νέοι μετεωρίτες θα εναποθέσουν τα μεταλλικά στοιχεία στους νέο σχηματισμένους ωκεανούς. Μέσα σε αυτή την πρωταρχική σούπα στοιχείων [primordial soup] (Oparin 1924) και κάτω από εξαιρετικά αντίξοες και μοναδικές συνθήκες τα πρώτα αμινοξέα σχηματίστηκαν στον πάτο των ωκεανών, γύρω από υδροθερμικές πηγές (εικόνα 2). Εκεί θα γεννηθούν και οι πρώτοι ζωντανοί οργανισμοί, τα προκαρυωτικά κύτταρα που με την σειρά τους θα δημιουργήσουν τα βακτήρια, απολιθώματα των οποίων χρονολογούνται πριν από 4,3 με 3,7 εκατομμύρια χρόνια. Η ζωή έχει ξεκινήσει. Σχεδόν 4 δισεκατομμύρια χρόνια μετά την δημιουργία της γης, το πείραμα των Μίλερ – Ουρεη (Miller – Urey) [1] κατόρθωσε από μια αντίστοιχη σούπα στοιχείων να αναδείξει μερικά βασικά αμινοξέα, που είναι τα δομικά συστατικά των πρωτεϊνών, της ίδιας της ζωής !



Εικόνα 2: Υδροθερμικές πηγές στον πάτο του ωκεανού

Στον σημερινό κόσμο υπάρχουν περίπου 9 εκατομμύρια διαφορετικοί οργανισμοί [2] που έχουν εξελιχθεί από εκείνα τα πρώιμα βακτήρια, σε μια διαδικασία εξέλιξης που συνεχίζεται ακόμη και στην πορεία της έχει αναδείξει και έχει εξαλείψει εκατομμύρια άλλους οργανισμούς. Αυτή η εξέλιξη από κοινούς προγόνους έχει ως αποτέλεσμα όλοι οι οργανισμοί να έχουν μεγάλα κομμάτια κοινά στο DNA και όμως να είναι τόσο διαφορετικοί μεταξύ τους. Το γονιδίωμα του ανθρώπου (Homo Sapiens) έχει μόλις 5% διαφορά από αυτό του ποντικού (Mus musculus) [3]. Τι είναι όμως αυτό που προκαλεί τόσο μεγάλη διαφοροποίηση μεταξύ οργανισμών με συντριπτικά μεγάλη ομοιότητα γονιδιώματος; Η απάντηση έρχεται από την έκφραση του γονιδιώματος, από τις πρωτεΐνες που παράγει ο οργανισμός. Μπορούμε έτσι να πούμε πως ο οργανισμός είναι το πρωτέωμά του ! Όπως μπορεί κάποιος εύκολα να φανταστεί, αυτή η ομοιότητα στο DNA των οργανισμών, εμφανίζεται κατ' επέκτασιν και στο πρωτέωμα αυτού. Και όμως οι σχετικά μικρές διαφορές σε αυτό έχουν δημιουργήσει το μεγάλο δέντρο της ζωής που βλέπουμε γύρω μας.



Εικόνα 3: Η εξέλιξη των οργανισμών

Οι σύγχρονες τεχνικές που έχουν αναπτυχθεί όπως η φασματογραφία μάζας (MS/MS Spectrometry), μας έχουν δώσει μια πληθώρα δεδομένων για τα πρωτεώματα πολλών οργανισμών. Οι πληροφορίες αυτές τροποποιούνται και επεκτείνονται συνεχώς, καθιστώντας την γνώση μας πάνω στους οργανισμούς ολοένα και πλουσιότερη.

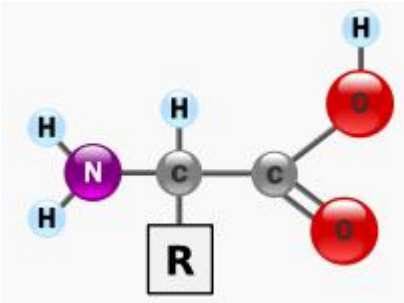
Σκοπός της εργασίας αυτής είναι να αναδείξει, με υπολογιστικές μεθόδους, μέσα από τα μοναδικά αυτά σύνολα πρωτεωμάτων οργανισμών (ή αυθαίρετα ορισμένων ομάδων πρωτεϊνών) , τα μικρότερα δομικά στοιχεία που μπορούν να ταυτοποιήσουν μοναδικά μια πρωτεΐνη μέσα σε αυτό το σύνολο.

2. ΒΙΟΛΟΓΙΚΟ ΥΠΟΒΑΘΡΟ

2.1 Αμινοξέα

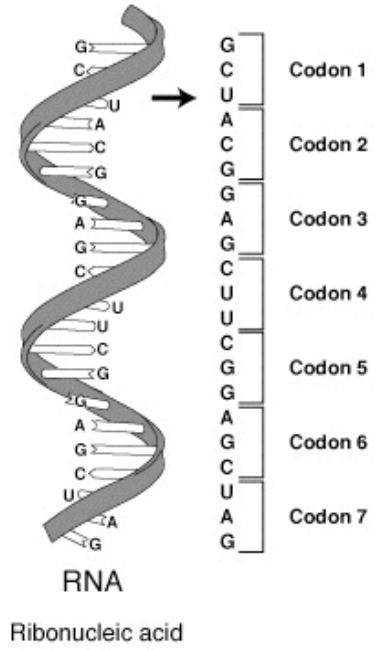
Τα αμινοξέα είναι μικρά οργανικά μόρια τα οποία περιέχουν μια αμινομάδα (-NH2), μια καρβοξυλομάδα (-COOH) καθώς και μια πλευρική αλυσίδα [R] που διαφοροποιεί κάθε αμινοξύ από τα υπόλοιπα. Τα βασικά στοιχεία των αμινοξέων είναι ο Άνθρακας (C), το υδρογόνο (H) το οξυγόνο (O) και το άζωτο (N) αν και σε κάποια αμινοξέα μπορούν να βρεθούν και άλλα στοιχεία στην πλευρική τους αλυσίδα.

Σήμερα γνωρίζουμε περίπου 500 αμινοξέα τα οποία μπορούν να σχηματιστούν με φυσικό τρόπο. Από αυτά μόνο μια μικρή μειοψηφία έχει ιδιαίτερο βάρος για την βιοχημεία. Τα συγκεκριμένα αμινοξέα έχουν ένα κεντρικό άτομο άνθρακα ενωμένο με μια αμινομάδα, μια καρβοξυλομάδα, ένα άτομο υδρογόνου και μια πλευρική αλυσίδα (-R) (Εικόνα 4). Ονομάζονται α-αμινοξέα και αποτελούν τις δομικές μονάδες των πρωτεϊνών, γι' αυτό και καλούνται πρωτεινογενετικά (proteinogenic). Το όνομά τους (άλφα) προέρχεται από ένα παλαιότερο σύστημα ονοματολογίας της οργανικής χημείας όπου τα άτομα μιας υδρογονανθρακικής αλυσίδας που είναι συνδεδεμένη ομοιοπολικά με μια καρβοξυλομάδα, συμβολίζονται με γράμματα του ελληνικού αλφαβήτου (α, β, ...) [4]

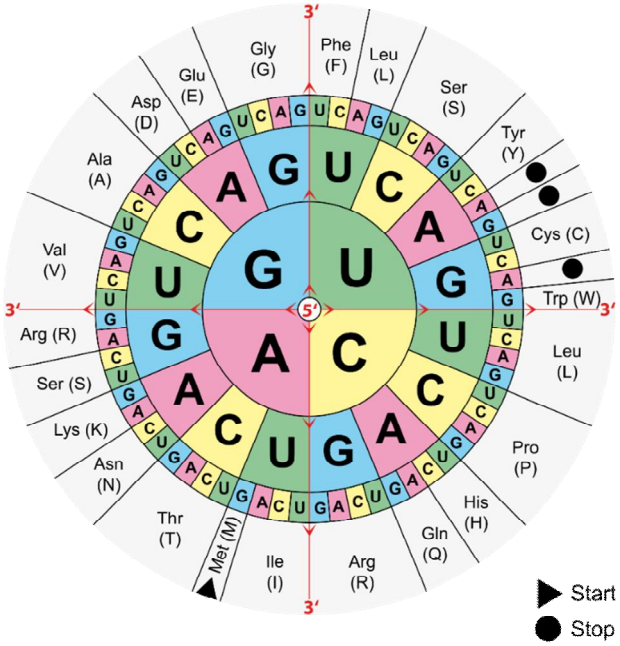


Εικόνα 4: Γενικός συντακτικός τύπος α-αμινοξέων

Υπάρχουν 22 α-αμινοξέα από τα οποία μόνο τα 20 μπορούν εξαχθούν κατευθείαν από 3-άδες κωδικονίων του γενετικού κώδικα (Εικόνες 5 κ 6). Τα 20 αυτά αμινοξέα συμβολίζονται με 1 ή 3 γράμματα (πίνακας 1).

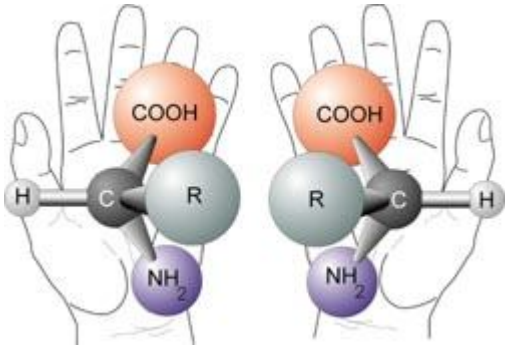


Εικόνα 5: Μετάφραση κωδικωνίων σε Αμινοξέα



Εικόνα 6: Ο γενετικός κώδικας

Με εξαίρεση την γλυκίνη, όλα τα α-αμινοξέα έχουν τέσσερα διαφορετικά είδη λειτουργικών ομάδων συνδεδεμένα στο α-άτομο άνθρακα. Αυτές μπορούν να διαταχθούν στον χώρο με δύο διαφορετικούς τρόπους σχηματίζοντας δυο διαφορετικά στερεοισομερή τα D-(Dextrorotatory) και L-(levorotatory) που είναι μεταξύ τους κατοπτρικά είδωλα. Από τις δύο αυτές μορφές μόνο η L- χρησιμοποιείται από τα κύτταρα για την παραγωγή των πρωτεϊνών.



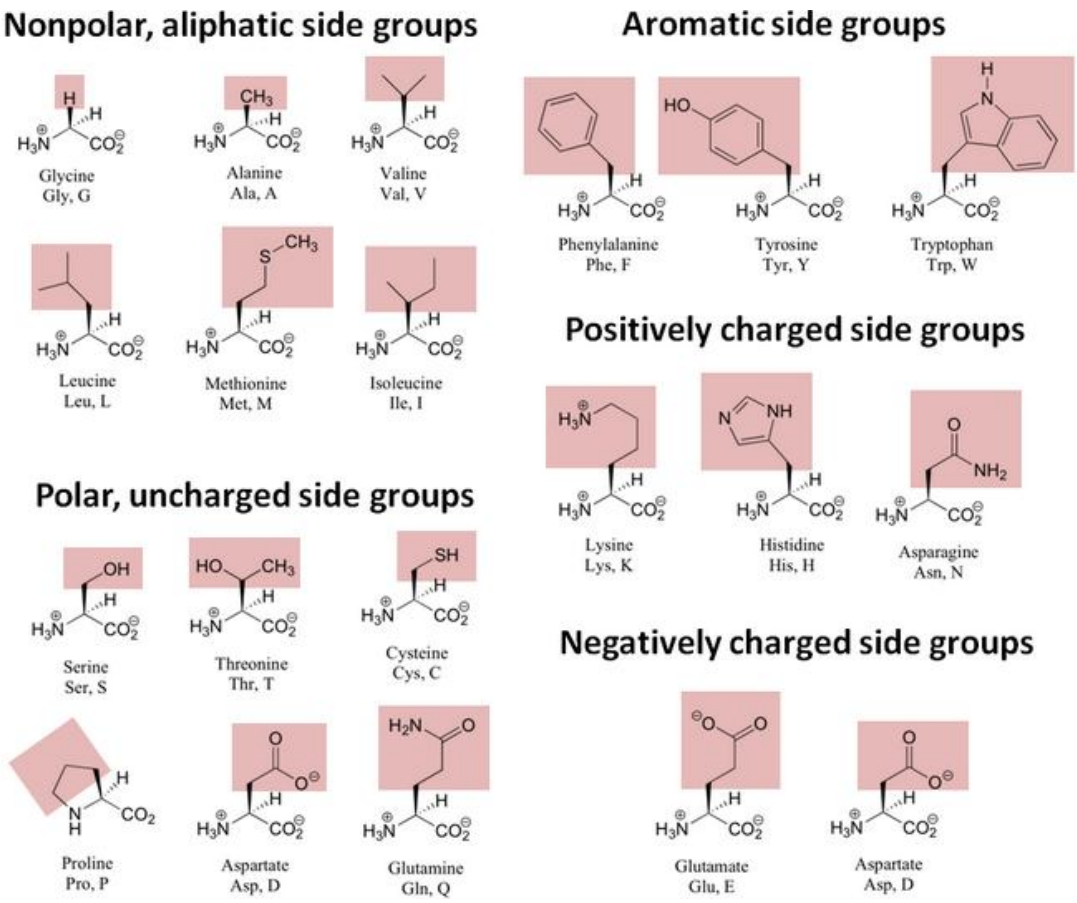
Εικόνα 7: Κατοπτρική μορφή D- / L- αμινοξέων

Πίνακας 1: α-αμινοξέα

Amino acid	3-letter code	1-letter code	MW (Da)	Structure
Alanine	Ala	A	89.1	CH3-CH(NH2)-COOH
Arginine	Arg	R	174.2	HN=C(NH2)-NH-(CH2)3-CH(NH2)-COOH
Asparagine	Asn	N	132.1	H2N-CO-CH2-CH(NH2)-COOH
Aspartic Acid	Asp	D	133.1	HOOC-CH2-CH(NH2)-COOH
Cysteine	Cys	C	121.2	HS-CH2-CH(NH2)-COOH
Glutamic Acid	Glu	E	147.1	HOOC-(CH2)2-CH(NH2)-COOH
Glutamine	Gln	Q	146.1	H2N-CO-(CH2)2-CH(NH2)-COOH
Glycine	Gly	G	75.1	NH2-CH2-COOH
Histidine	His	H	155.2	NH-CH=N-CH=C-CH2-CH(NH2)-COOH |_____|
Isoleucine	Ile	I	131.2	CH3-CH2-CH(CH3)-CH(NH2)-COOH
Leucine	Leu	L	131.2	(CH3)2-CH-CH2-CH(NH2)-COOH
Lysine	Lys	K	146.2	H2N-(CH2)4-CH(NH2)-COOH
Methionine	Met	M	149.2	CH3-S-(CH2)2-CH(NH2)-COOH
Phenylalanine	Phe	F	165.2	Ph-CH2-CH(NH2)-COOH
Proline	Pro	P	115.1	NH-(CH2)3-CH-COOH |_____|
Serine	Ser	S	105.1	HO-CH2-CH(NH2)-COOH
Threonine	Thr	T	119.1	CH3-CH(OH)-CH(NH2)-COOH
Tryptophan	Trp	W	204.2	Ph-NH-CH=C-CH2-CH(NH2)-COOH |_____|
Tyrosine	Tyr	Y	181.2	HO-p-Ph-CH2-CH(NH2)-COOH
Valine	Val	V	117.1	(CH3)2-CH-CH(NH2)-COOH

Οι φυσικές και χημικές ιδιότητες κάθε αμινοξέος καθορίζονται από την χαρακτηριστική πλευρική του αλυσίδα (-R). Αμινοξέα με παρόμοιες πλευρικές αλυσίδες έχουν και παρόμοιες ιδιότητες λόγω της πολικότητας που αυτή τους προσδίδει. Ανάλογα με αυτές τις ιδιότητες τα 20 α-αμινοξέα διακρίνονται σε 4 ομάδες:

1. **Βασικό χαρακτήρα:** Οι πλευρικές ομάδες είναι δέκτες πρωτονίων. Σε αυτή την ομάδα ανήκουν τα αμινοξέα **αργεντινή** (arginine), **λυσίνη** (lysine), **ιστιδίνη** (histidine).
2. **Όξινο χαρακτήρα:** Έχουν στην πλευρική ομάδα μια ομάδα καρβοξυλίου. Αυτή είναι μη πρωτονιωμένη και έχει αρνητικό φορτίο σε pH 7. Σε αυτή την ομάδα ανήκουν το **ασπαρτικό οξύ** (aspartic acid) και το **γλουταμικό οξύ** (glutamic acid).
3. **Πολικό χαρακτήρα αλλά χωρίς φορτίο σε pH 7:** Σε αυτή την ομάδα ανήκουν τα αμινοξέα **ασπαριγίνη** (asparagine), **γλουταμίνη** (glutamine), **σερίφη** (serine), **θρεονίνη** (threonine) και **τυροσίνη** (tyrosine).
4. **Μη πολικό χαρακτήρα:** Σε αυτή την ομάδα ανήκουν τα αμινοξέα **γλυκίνη** (glycine), **αλανίνη** (alanine), **ισολευκίνη** (isoleucine), **λευκίνη** (leucine), **βαλίνη** (valine), **φαινυλαλανίνη** (phenylalanine), **τρυπτοφάνη** (tryptophan), **μεθειονίνη** (methionine) και η **προλίνη** (proline).



Εικόνα 8: Ομαδοποίηση α-αμινοξέων βάση της πολικότητάς τους

Ένα σημαντικό χαρακτηριστικό με ιδιαίτερη σημασία για την κατανομή στον χώρο των α-αμινοξέων σε μια πεπτιδική αλυσίδα είναι η υδροφοβικότητά τους. Με τον όρο υδροφοβικότητά αναφερόμαστε στην προδιάθεση ενός μορίου να απωθηθεί από τα περιβάλλονται μόρια του νερού (πρακτικά αναφερόμαστε στην ανυπαρξία ελκτικών δυνάμεων και όχι στην ύπαρξη απωθητικών). Τα υδροφοβικά μόρια τείνουν να μην παρουσιάζουν πολικότητα και γι' αυτό έχουν προτίμηση στα μη πολικά διαλύματα καθώς και στα ουδέτερα μόρια. Παραδείγματα υδροφοβικών μορίων είναι τα λίπη και τα έλαια. Με βάση αυτό το χαρακτηριστικό τα α-αμινοξέα μπορούν να χωριστούν όπως φαίνεται στον πίνακα 2. Η υδροφοβικότητα των αμινοξέων επηρεάζει την αναδίπλωση των πρωτεϊνών στον τρισδιάστατο χώρο, αφού κάποια αμινοξέα έλκονται από τα μόρια του νερού που βρίσκονται στον περιβάλλοντα χώρο αυτής οδηγώντας στην τελική της διάταξη.

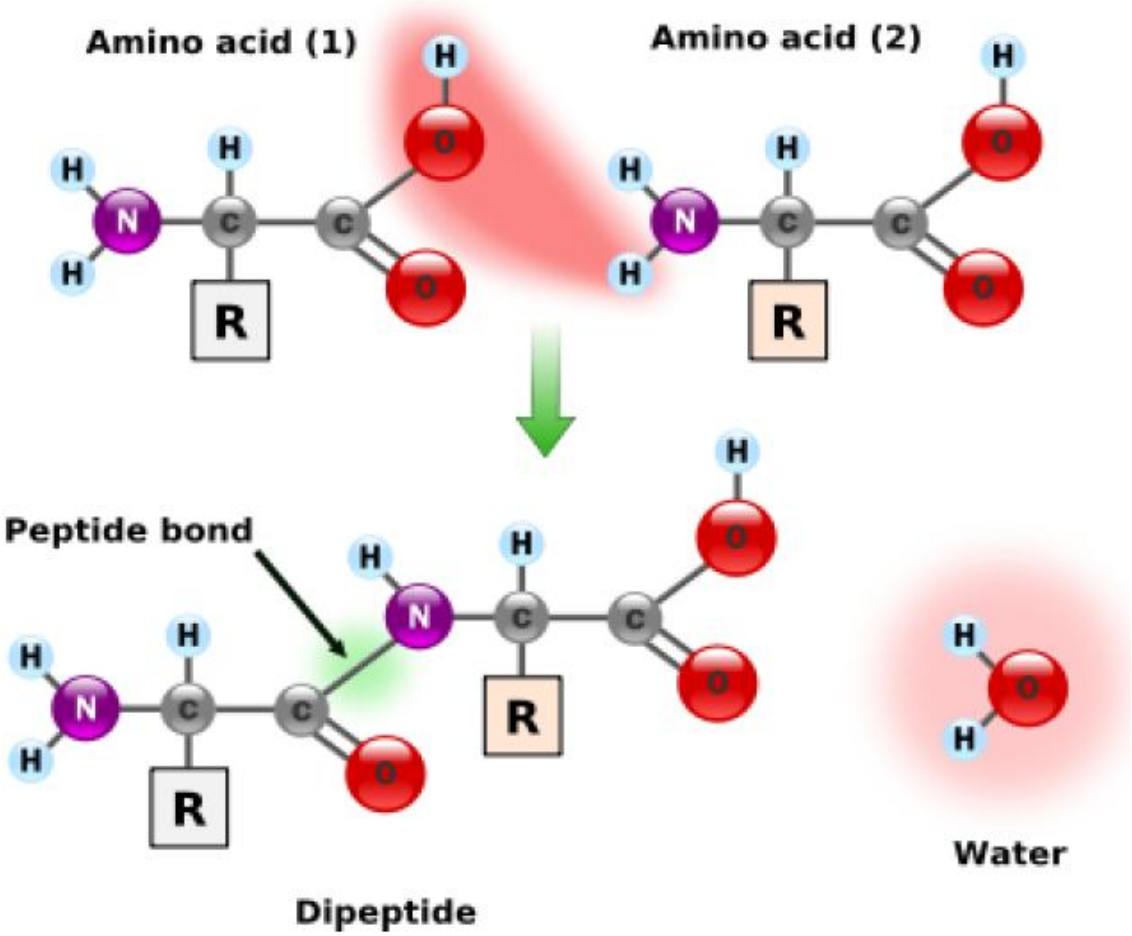
Πίνακας 2: Ομαδοποίηση α-αμινοξέων με βάση την υδροφοβικότητά τους

At pH 2[A]		At pH 7[B]	
Very Hydrophobic			
Leu	100	Phe	100
Ile	100	Ile	99
Phe	92	Trp	97
Trp	84	Leu	97
Val	79	Val	76
Met	74	Met	74
Hydrophobic			
Cys	52	Tyr	63
Tyr	49	Cys	49
Ala	47	Ala	41
Neutral			
Thr	13	Thr	13
Glu	8	His	8
Gly	0	Gly	0
Ser	-7	Ser	-5
Gln	-18	Gln	-10
Asp	-18		
Hydrophilic			
Arg	-26	Arg	-14
Lys	-37	Lys	-23
Asn	-41	Asn	-28
His	-42	Glu	-31
Pro	-46	Pro	-46 (used pH 2)
		Asp	-55

[A] pH 2 values: Normalized from Sereda et al., J. Chrom. 676: 139-153 (1994).
[B] pH 7 values: Monera et al., J. Protein Sci. 1: 319-329 (1995).

2.2 Πεπτίδια

Τα α-αμινοξέα συνδέονται μεταξύ τους μέσω του πεπτιδικού δεσμού σχηματίζοντας μεγαλύτερες αλυσίδες τα πεπτίδια. Συγκεκριμένα η σύνδεση αυτή επιτυγχάνεται μέσω της αντίδρασης της α-καρβοξυλομάδας ενός αμινοξέος με την α-αμινομάδα ενός άλλου. Αυτός ο δεσμός είναι ομοιοπολικός ο οποίος κατά τον σχηματισμό του οδηγεί στην αποβολή ενός μορίου νερού.

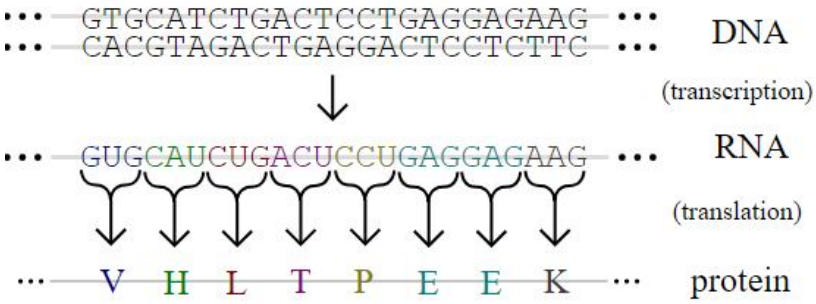


Εικόνα 9: Σχηματισμός διπεπτιδίου με πεπτιδικό δεσμό και αποβολή ενός μορίου νερού

Σε αντιστοιχία με τις πολυνουκλεοτιδικές αλυσίδες, οι πεπτιδικές αλυσίδες εμφανίζουν επίσης πολικότητα, αφού στο ένα άκρο της πεπτιδικής αλυσίδας υπάρχει μια ελεύθερη αμινομάδα (θετικά φορτισμένη) ενώ στο άλλο μια ελεύθερη καρβοξυλομάδα (αρνητικά φορτισμένη). Τα πεπτίδια χωρίζονται κατά σύμβαση ως προς το μέγεθος τους σε 2 ομάδες, τα ολιγοπεπτίδια που έχουν 50 ή λιγότερα αμινοξέα και τα πολυπεπτίδια με περισσότερα από 50 αμινοξέα. Το πλήθος των πιθανών πεπτιδίων όπως εύκολα γίνεται κατανοητό είναι απεριόριστο αφού μόνο για τα ολιγοπεπτίδια μήκους 50 έχουμε 20^{50} διαφορετικά πεπτίδια που μπορούν να σχηματιστούν.

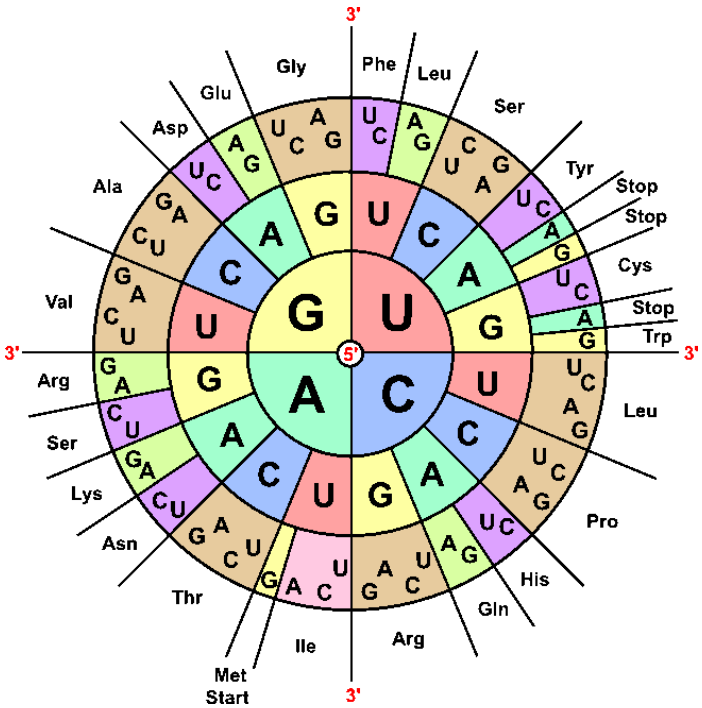
2.3 Πρωτεΐνες

Οι πρωτεΐνες είναι μεγάλα και πολύπλοκα μόρια τα οποία παίζουν πολλούς και σημαντικούς ρόλους στην λειτουργία των έμβιων οργανισμών. Αποτελούνται από λίγα έως χιλιάδες αμινοξέα και είναι απαραίτητες για την δόμηση, την λειτουργία και τον έλεγχο των ιστών και των οργάνων. Συνθέτονται με βάση τις οδηγίες που είναι κωδικοποιημένες στο DNA κάθε οργανισμού μέσω της διαδικασίας της μεταγραφής / μετάφρασης.



Εικόνα 10: Παραγωγή πρωτεΐνης από DNA

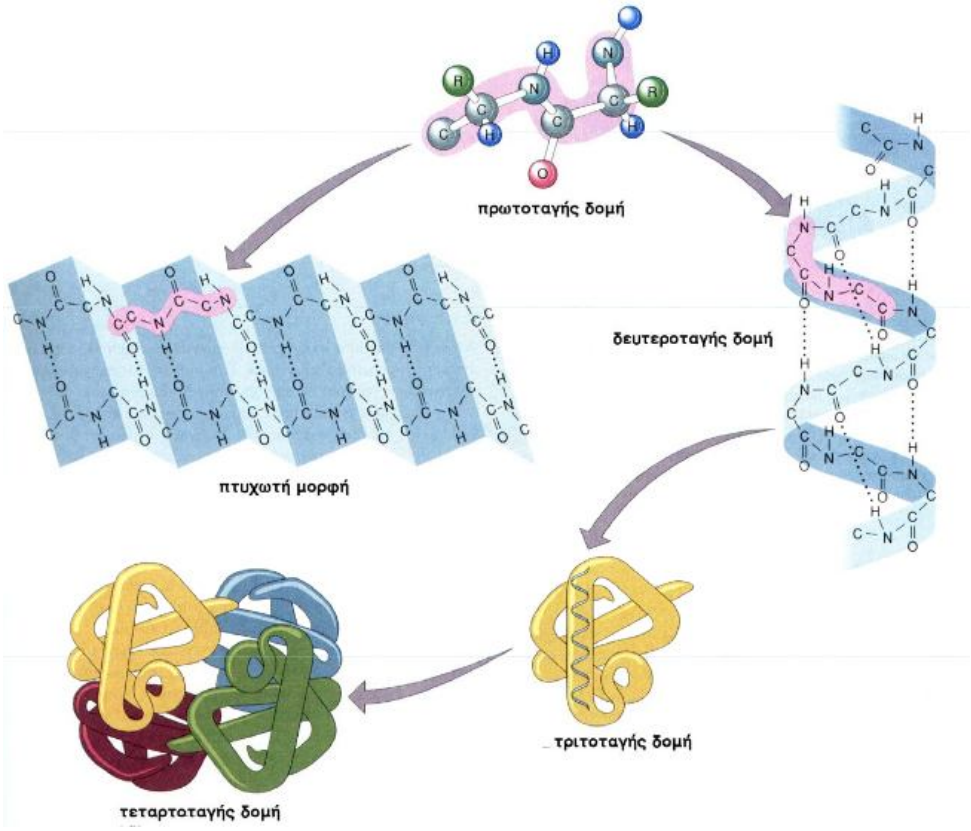
Όπως αναφέρθηκε παραπάνω, μια 3άδα βάσεων DNA μπορεί να κωδικοποιήσει ένα α-αμινοξύ, ενώ ένα α-αμινοξύ μπορεί να προέλθει από περισσότερους του ενός συνδυασμούς 3άδων βάσεων όπως φαίνεται και στην εικόνα 11 .



Εικόνα 11: Πίνακας αντιστοίχισης 3αδων βάσεων με αμινοξέα

Αρχικά οι πρωτεΐνες μπορούν να θεωρηθούν ως απλές αλληλουχίες αμινοξέων. Αυτή η θεώρηση μας είναι η **πρωτοταγής δομή** της πρωτεΐνης. Όμως οι πρωτεΐνες «διπλώνουν» στον 2διάστατο χώρο, δημιουργώντας αρκετά πολύπλοκες δομές που. Η δίπλωση αυτή προκύπτει αρχικά από την φυσική σύνδεση με δεσμούς υδρογόνου μεταξύ των καρβοξυλομάδων και αμινομάδων των πρωτεϊνών. Αυτή είναι η **δευτεροταγής δομή** της πρωτεΐνης. Στην συνέχεια τα πολυπεπτίδια υφίστανται μια αρκετά πιο πολύπλοκη δίπλωση στον 3διαστατο χώρο δίνοντας μας την **τριτοταγή δομή** της πρωτεΐνης. Τέλος διάφορες πρωτεΐνες μπορεί να αποτελούνται από

επιμέρους πολυπεπτίδια που ενώνονται μεταξύ τους δημιουργώντας την **<u>τεταρτοταγή δομή</u>** της πρωτεΐνης.



Εικόνα 12: Πρωτεϊνικές δομές

2.4 Πρωτεωμική

Το σύνολο όλων των πρωτεϊνών που παράγονται σε έναν οργανισμό ή σύστημα ονομάζεται πρωτέωμα αυτού [Marc Wilkins 1994]. Η πρωτεωμική (proteomics) είναι η ευρείας κλίμακας μελέτη των πρωτεωμάτων. Ορίστηκε το 1997 σε αναλογία με τον όρο γενωμική (genomics) που αναφέρεται στην μελέτη του γονιδιώματος. Η πρωτεωμική χρησιμοποιείται για την διερεύνηση των πρωτεωμάτων σε πολλά διαφορετικά επίπεδα όπως φαίνεται στην παρακάτω λίστα:

- Που και πότε εκφράζονται οι πρωτεΐνες.
- Τον ρυθμό παραγωγής, αποδόμησης καθώς και το πλήθος των πρωτεϊνών σε σταθερή κατάσταση.
- Πως τροποποιούνται οι πρωτεΐνες μετά την μετάφρασή τους.
- Την μετακίνησή τους ανάμεσα στα τμήματα των κυττάρων.
- Την εμπλοκή των πρωτεϊνών στα μεταβολικά μονοπάτια.
- Πως αλληλοεπιδρούν οι πρωτεΐνες μεταξύ τους.

Επίσης η μελέτη της πρωτεωμικής μπορεί να αναδείξει σημαντικές βιολογικές πληροφορίες όπως:

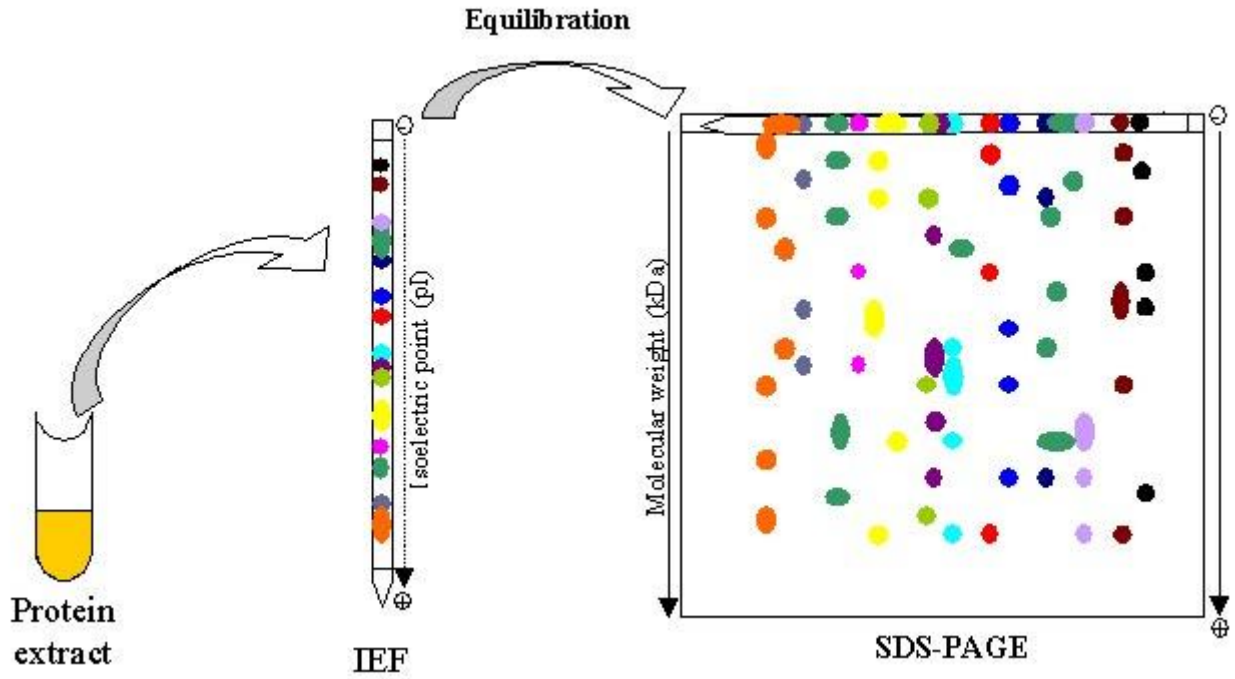
- Ποιες πρωτεΐνες αλληλοεπιδρούν με μια συγκεκριμένη πρωτεΐνη που μας ενδιαφέρει.
- Ποιες πρωτεΐνες δρουν τοπικά σε κάποιον υπό-κυτταρικό τομέα (subcellular location)
- Ποιες πρωτεΐνες εμπλέκονται σε μια βιολογική διαδικασία

Μέχρι σήμερα έχουν αναπτυχθεί αρκετές διαφορετικές τεχνολογίες για την διερεύνηση των πρωτεωμάτων. Η σημαντικότερες από αυτές βασίζονται στην φασματογραφία μάζας (Mass Spectrometry) όπως η υποβοηθούμενη MS (Tandem MS) ή σε τεχνολογίες χρήσης γέλης (gel) όπως η διαφορική ηλεκτροφόρηση γέλης (differential ingel electrophoresis - DIGE)[5].

3. ΤΑΥΤΟΠΟΙΗΣΗ ΠΡΩΤΕΪΝΩΝ

3.1 MS/MS Spectrometry

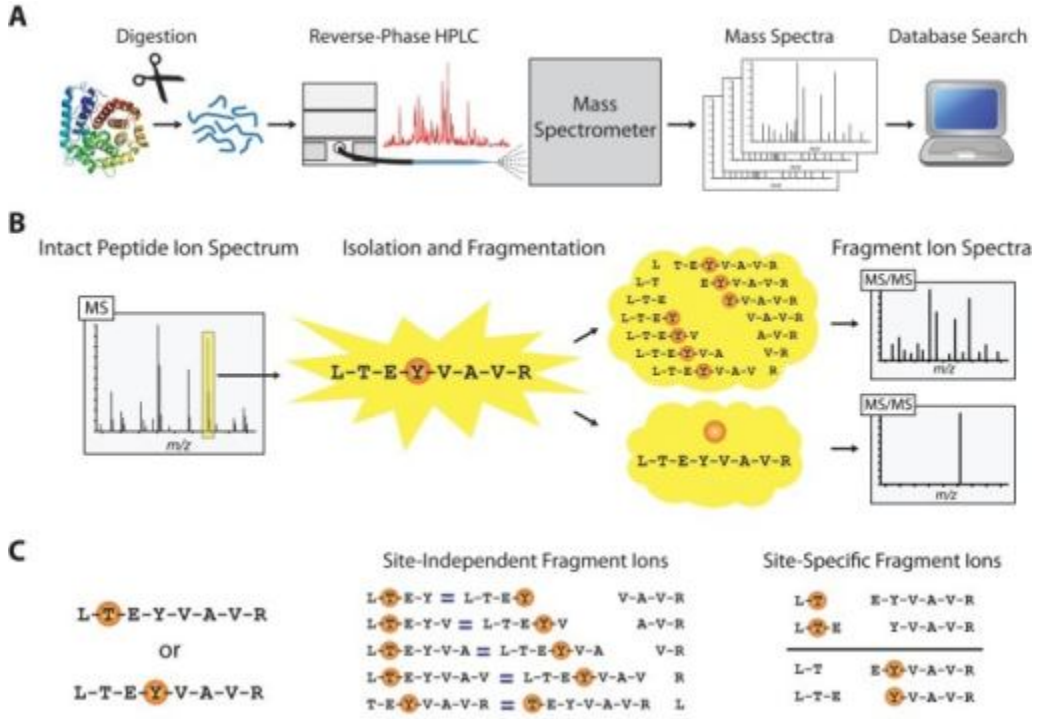
Οι πρωτείνες εξάγονται από τα κύτταρα, σωματικά υγρά ή τους ιστούς και διαχωρίζονται μεταξύ τους, συνήθως με χρήση ηλεκτροφόρησης δύο διαστάσεων. Στην συγκεκριμένη τεχνική το δείγμα που περιέχει τις πρωτείνες αναμειγνύεται με κάποιον διαλύτη και τοποθετείται σε μια βαθμονομημένη ταινία από πολυμερές υλικό η μία διάσταση της οποία δείχνει το pH και η άλλη την μοριακή μάζα σε kDa (kiloDalton). Στην συνέχεια εφαρμόζεται ηλεκτρικό πεδίο στην ταινία τέτοιο ώστε κάθε πρωτείνη να γίνει ηλεκτρικά ουδέτερη. Το pH σε αυτή την φάση ονομάζεται ισοηλεκτρικό pH (ηλεκτροφόρηση στην πρώτη διάσταση και συμβολίζεται με pi). Αμέσως μετά η ταινία ηλεκτροφορείται πάλι και (ηλεκτροφόρηση στην δεύτερη διάσταση). Σε αυτό το σημείο οι πρωτείνες μετακινούνται κατά μήκος της ταινίας. Η κίνησή τους επηρεάζεται από την μοριακή τους μάζα και συγκεκριμένα όσο μικρότερη είναι αυτή τόσο πιο γρήγορα κινείται η πρωτείνη ανάμεσα στους δύο πόλους του ηλεκτρικού πεδίου. Μετά το πέρας της διαδικασίας οι πρωτείνες του αρχικού μείγματος έχουν διαχωριστεί μεταξύ τους και μπορούν να χρωματιστούν κατάλληλα με χρωστικές ουσίες όπως φαίνεται στην εικόνα 13.



Εικόνα 13: Ηλεκτροφόρηση 2 διαστάσεων

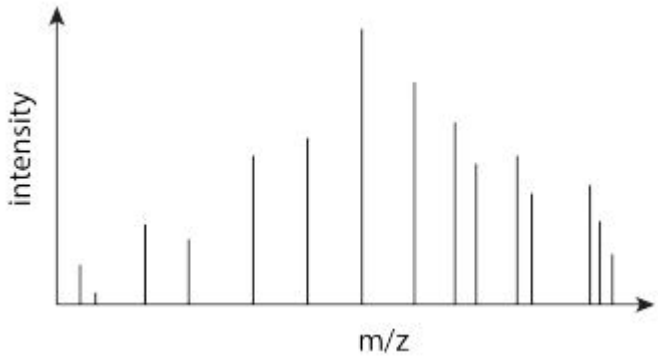
Σε αυτό το σημείο αν και η μοριακή μάζα της πρωτεΐνης είναι γνωστή, η αλληλουχία των αμινοξέων της δεν είναι. Μια από τις σημαντικότερες τεχνολογίες ταυτοποίησης πρωτεϊνών είναι η υποβοηθούμενη φασματογραφία μάζας (Tandem MS ή MS/MS Spectrometry). Η φασματογραφία Μάζας είναι μια αναλυτική τεχνική για την μέτρηση της αναλογίας μάζας προς φορτίο και την ένταση των φορτισμένων μορίων ή σωματιδίων. Τα αποτελέσματα της οπτικοποιούνται όπως στην εικόνα όπου στον άξονα χ εμφανίζονται οι αναλογίες Μάζας/Φορτίου (m/z) και στον άξονα y η ένταση (intensity) [11] .

Η τεχνική της MS φασματογραφίας μάζας για την ταυτοποίηση των πρωτεϊνών (shotgun proteomics) μοιάζει πολύ με την αντίστοιχη τεχνική που χρησιμοποιείται για την εύρεση του DNA όπου μεγάλα κομμάτια DNA κόβονται σε μικρότερα μέρη πριν γίνει η αλληλούχισή τους (shotgun genomics).



Εικόνα 14: Ανάλυση πρωτεΐνης με χρήση MS/MS φασματογραφίας

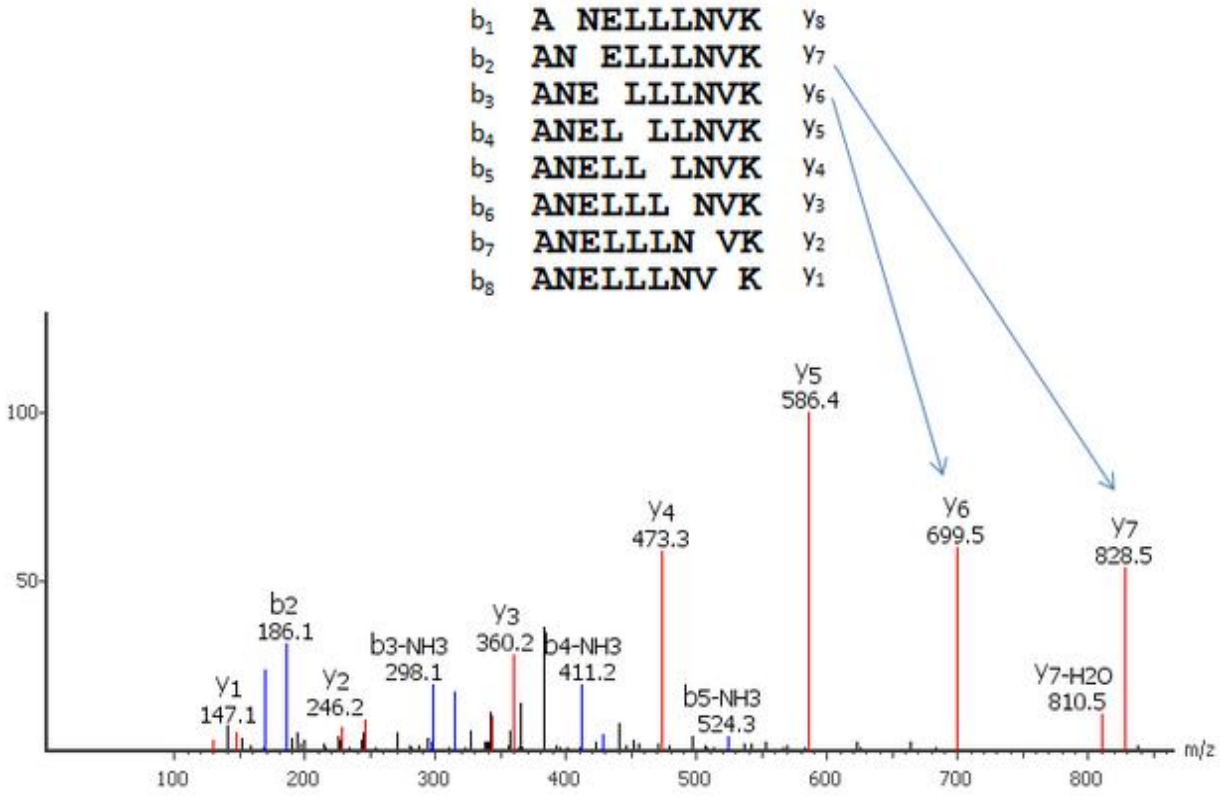
Κάθε μία από τις πρωτείνες που αναγνωρίστικαν στο πρώτο στάδιο της ηλεκτροφόρησης δύο διαστάσεων χωρίζονται σε πεπτίδια με την χρήση πρωτεολυτικών ενζύμων όπως η τρυψίνη. Για τα κομμάτια αυτά μετράται η μάζα τους με χρήση της φασματομετρίας μάζας μέσω της διαδικασίας peptide mass finger printings (εικόνα 14 A). Στο επόμενο στάδιο τα πεπτίδια τοποθετούνται στο φασματόμετρο μαζών όπου αφού μετατραπούν σε μια λεπτή δέσμη ιόντων σε αέρια φάση διέρχονται μέσω μαγνητικού ή ηλεκτρικού πεδίο όπου κάθε ιόν αποκλίνει διαφορετικά ανάλογα με τον λόγο m/z (μάζας/ φορτίου) που έχει (εικόνες 14 B και 15).



Εικόνα 15: m/z προς Ένταση γράφημα

Τα αποτελέσματα (μοριακά βάρη) της διαδικασίας αυτής συγκρίνονται με γνωστές βάσεις δεδομένων μοριακών βαρών αμινοξέων, από όπου και προκύπτει με χρήση

υπολογιστικών μεθόδων η σύνθεση της αλληλουχίας της αμινοξικής ακολουθίας τους πεπτιδίου (εικόνες 14 C και 16).

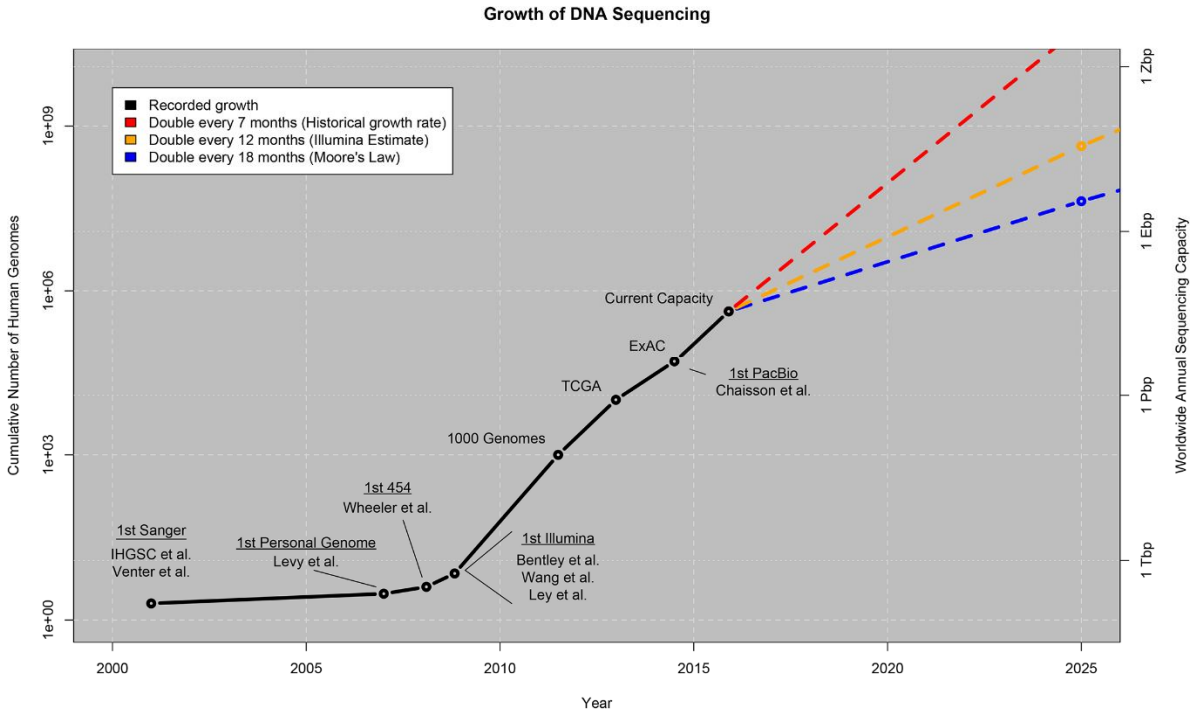


Εικόνα 16: Ανάλυση Πεπτιδίων (Μέρος Β προηγούμενης εικόνας)

Ο ιονισμός των πρωτεϊνικών δειγμάτων γίνεται με διάφορες μεθόδους. Μερικές από τις σημαντικότερες είναι ο ιονισμός με ηλεκτροψεκασμό (electro spray ionization), ο χημικός ιονισμός (chemical ionization), ο ιονισμός εκρόφησης με λέιζερ παρουσία βοηθητικής ουσίας (MALDI), ο βομβαρδισμός με ταχέα άτομα (fast atom bombardment – FAB) καθώς και ο θερμικός ιονισμός (thermal ionization)

3.2 Βιολογικά Δεδομένα και Δημόσιες Βάσεις Δεδομένων

Οι τεχνολογίες που αναπτύχθηκαν τις τελευταίες δεκαετίες για την διερεύνηση του γονιδιώματος και του πρωτεώματος των οργανισμών, καθώς και τις αλληλεπιδράσεις και συσχετίσεις τους έχουν δημιουργήσει και συνεχίζουν να δημιουργούν τεράστια ποσά δεδομένων τα οποία είναι αδύνατο να αναλυθούν και εκτιμηθούν χωρίς την βοήθεια της πληροφορικής [6][7].



Εικόνα 17: Ρυθμός αύξησης δεδομένων αλληλούχισης DNA

Data Phase	Astronomy	Twitter	YouTube	Genomics
Acquisition	25 zetta-bytes/year	0.5–15 billion tweets/year	500–900 million hours/year	1 zetta-bases/year
Storage	1 EB/year	1–17 PB/year	1–2 EB/year	2–40 EB/year
Analysis	In situ data reduction	Topic and sentiment mining	Limited requirements	Heterogeneous data and analysis
	Real-time processing	Metadata analysis		Variant calling, ~2 trillion central processing unit (CPU) hours
	Massive volumes			All-pairs genome alignments, ~10,000 trillion CPU hours
Distribution	Dedicated lines from antennae to server (600 TB/s)	Small units of distribution	Major component of modern user's bandwidth (10 MB/s)	Many small (10 MB/s) and fewer massive (10 TB/s) data movement

doi:10.1371/journal.pbio.1002195.t001

Εικόνα 18: Σύγκριση δεδομένων αλληλούχισης DNA με άλλες πλατφόρμες περιεχομένου.

Αποτέλεσμα του συγκερασμού αυτών είναι η δημιουργία διαφόρων ανοιχτών βάσεων δεδομένων για μια πληθώρα διαφορετικών τύπων όπως γενωμικής, πρωτεωμικής, ασθενειών, μεταβολικής και άλλων. Στην παρούσα εργασία μας το ενδιαφέρον μας είναι κυρίως στα πρωτεϊνικά δεδομένα. Μερικές από τις σημαντικότερες βάσεις εμφανίζονται στον παρακάτω πίνακα[8] . Οι βάσεις που παρουσιάζονται συνοπτικά στον πίνακα, επιλέχθηκαν είτε γιατί έγινε χρήση των δεδομένων τους, είτε γιατί η πληροφορία που περιέχουν θεωρείτε πως μπορεί να ενταχθεί στην παρούσα μελέτη.

Πίνακας 3 : Πρωτεϊνικές Βάσεις Δεδομένων

Ονομασία	Δεδομένα που περιέχονται – Παρατηρήσεις
UnitProtKB/ Swiss-Prot	Περιέχει δεδομένα πρωτεϊνών που έχουν επισημανθεί και αξιολογηθεί χειροκίνητα από ερευνητές και αξιολογητές. Στην βάση αποθηκεύονται μόνο αναθεωρημένα (reviewed) δεδομένα.
UnitProtKB/TrEMBL	Περιέχει δεδομένα που έχουν αναλυθεί και εμπλουτιστεί με επισημάνσεις μέσω υπολογιστικών μεθόδων.
Protein Information Resource	Περιέχει πλήθος διαφορετικών δεδομένων που αφορούν τις πρωτεΐνες όπως πληροφορίες για την λειτουργικότητα ή την τοπικότητα (localization) των πρωτεϊνών, βιβλιογραφίας και άλλες.
Uniprot	Αποτελείται από την σύνθεση των τριών παραπάνω βάσεων καθώς και από άλλες πηγές που ομαδοποιούνται κάτω από μια κοινή βάση. Παράλληλα προσφέρει πληθώρα εργαλείων, όπως εργαλεία για την στοίχιση ακολουθιών (alignment), αναζήτησης εντός των βάσεων και άλλα.
Human Protein Atlas	Αντιστοίχιση πρωτεϊνών του ανθρώπινου οργανισμού με τα όργανα χρησιμοποιώντας διάφορα δεδομένα (-omics data)
Protein Data Bank	Περιέχει τις τριδιάστατες δομές πρωτεϊνών που έχουν προκύψει από διάφορες τεχνολογίες όπως Κρυσταλλογραφία ακτινών X (X-ray crystallography και η NMR φασματοσκοπία (NMR spectroscopy). Αποτελείται από σύμπραξη των παρακάτω: Protein Data Bank in Europe Protein Data Bank in Japan Research Collaboratory for Structural Bioinformatics (RCSB)
Pfam	Περιέχει πληροφορίες για τον διαχωρισμό των πρωτεϊνών σε οικογένειες με βάση τις επισημάνσεις, την στοίχιση των ακολουθιών τους καθώς και την χρήση Κρυφών Μαρκοβιανών Μοντέλων (Hidden Markov Models)
Reactome	Περιέχει πληροφορίες για βιολογικά μονοπάτια και διαδικασίες και την εμπλοκή των πρωτεϊνών σε αυτά.

Τα δεδομένα που θα αναλύσουμε παρακάτω έχουν ληφθεί από την UniprotKB/Swiss-Prot. Όπως αναφέρθηκε παραπάνω η συγκεκριμένη βάση περιέχει μόνο τις πρωτεΐνες αυτές για τις οποίες η ύπαρξη καθώς και η λειτουργία έχει επιβεβαιωθεί χειροκίνητα από αξιολογητές (Reviewed). Η βάση παρέχει την δυνατότητα στον χρήστη να επιλέξει τα δεδομένα που επιθυμεί ανά οργανισμό και να τα μεταφορτώσει στην μορφή που επιθυμεί. Οι παρεχόμενες επιλογές είναι:

- FASTA (Canonical)
- FASTA (Canonical & Isoform)
- Tab-Separated
- Excel
- XML
- RDF/XML
- Text
- GFF
- List

Για κάθε οργανισμό μεταφορτώθηκε το αρχείο των Reviewed πρωτεϊνών του σε FASTA (canonical) μορφή και χρησιμοποιήθηκε ως είσοδος στην εφαρμογή.

3.3 Uniquome

Η ακολουθία κάθε πρωτεΐνης είναι προφανώς μοναδική εντός του πρωτεώματος του συγκεκριμένου οργανισμού. Όμως υπάρχουν μικρά τμήματα αυτών των ακολουθιών τα οποία επίσης συναντώνται μόνο σε μια πρωτεΐνη. Αυτή η μοναδική ακολουθία είναι ένα **πεπτίδιο μοναδικής αλληλουχίας αμινοξέων (Unique Peptide).** Ιδιαίτερο ενδιαφέρον έχει ο εντοπισμός των μικρότερων τέτοιων δυνατών ακολουθιών εντός μιας πρωτεΐνης, το οποία είναι δυνατόν να χαρακτηρίσουν την πρωτεΐνη. Τα ελάχιστα αυτά τμήματα ονομάζονται **Core Unique Peptides** (πεπτίδια ελάχιστης μοναδικής ακολουθίας) [9][10]. Κατά αντιστοιχία με το πρωτέωμα που είναι το σύνολο των πρωτεϊνών ενός οργανισμού ή συστήματος, το **Uniquome** είναι το σύνολο των πεπτιδίων ελάχιστης μοναδικής ακολουθίας ενός πρωτεώματος [10].

Η ανάλυση του uniquome έδειξε ότι υπάρχουν πεπτίδια μοναδικής ακολουθίας τα οποία αποτελούνται από 2 ή περισσότερα πεπτίδια ελάχιστης μοναδικής ακολουθίας που αλληλεπικαλύπτονται ή γειτνιάζουν. Τα πεπτίδια αυτά τα ονομάζουμε σύνθετα μοναδικά πεπτίδια **(Composite Unique Peptide)** [10].

Για να γίνουν καλύτερα αντιληπτές οι παραπάνω έννοιες παρατίθεται ένα παράδειγμα που αφορά την πρωτεΐνη P61513 του ανθρώπινου πρωτεώματος. Η συγκεκριμένη πρωτεΐνη έχει μήκος 92 αμινοξέα και περιέχει 40 πεπτίδια ελάχιστης μοναδικής ακολουθίας. Αυτά συνθέτουν 3 σύνθετα μοναδικά πεπτίδια. Ένα από αυτά είναι το **TTSAVTVKSAI** που βρίσκεται μεταξύ των θέσεων 73 και 83 της πρωτεΐνης. Αυτό συντίθεται από τα παρακάτω 5 πεπτίδια ελάχιστης μοναδικής ακολουθίας που αλληλεπικαλύπτονται:

- TTSAVT θέση 73
- SAVTVK θέση 75
- AVTVKS θέση 76
- VTVKSA θέση 77
- TVKSAI θέση 78

...	T	T	S	A	V	T	V	K	S	A	I	...
	T	T	S	A	V	T						
			S	A	V	T	V	K				
				A	V	T	V	K	S			
					V	T	V	K	S	A		
						T	V	K	S	A	I	

Εικόνα 19: Δημιουργία Σύνθετου Μοναδικού Πεπτιδίου από τα αντίστοιχα Ελάχιστου Μήκους Μοναδικά

Η καταγραφή του Uniquome είναι ιδιαίτερα σημαντική αφού επιτρέπει την ταυτοποίηση πρωτεϊνών με μεγάλη βεβαιότητα από μικρά πεπτίδια τα οποία είναι μοναδικά εντός αυτής. Συγκριμένα μετά από την ανάλυση ενός μίγματος πρωτεϊνών με φασματομετρία μάζας και την ταυτοποίηση των πεπτιδίων τους είναι δυνατή η ταυτοποίηση μίας πρωτεΐνης από ένα μόνο πεπτίδιο εφόσον αυτό είναι μοναδικό για την εν λόγω πρωτεΐνη. Επιπλέον μπορεί να αποκαλύψει σημαντικά στοιχεία σε ότι αφορά την βιολογική λειτουργία αυτών αναγνωρίζοντας μοναδικότητάς εντός οικογενειών πρωτεϊνών ή υποσυστημάτων.

Για την ευκολότερη ανάγνωση, στο υπόλοιπο της παρούσας εργασίας αντί των εξελληνισμένων όρων θα χρησιμοποιείται η αγγλική ορολογία, το αρκτικόλεξο ή η αγγλική ορολογία έχοντας παραλείψει την λέξη peptide:

Πίνακας 4: Ορολογίες που θα χρησιμοποιηθούν στο υπόλοιπο κείμενο

Ελληνική Ορολογία	Αγγλική Ορολογία	Αρκτικόλεξο στα Αγγλικά
Πεπτίδιο Ελάχιστης Μοναδικής Ακολουθίας	Core Unique Peptide	CUP
Σύνθετο Πεπτίδιο Μοναδικής Ακολουθίας	Composite Unique Peptide	ComUP

4. ΥΠΟΛΟΓΙΣΤΙΚΟ ΥΠΟΒΑΘΡΟ

4.1 Επίλυση προβλημάτων - Αλγόριθμος

Ο άνθρωπος από σχετικά νωρίς στην ιστορία του, έμαθε να χρησιμοποιεί επαναλαμβανόμενες διαδικασίες για να επιλύσει τα καθημερινά του προβλήματα. Με το πέρασμα των αιώνων και την έλευση των μαθηματικών κωδικοποίησε αυτές τις διαδικασίες με καλώς ορισμένους τρόπους.

Η κωδικοποίηση αυτή ονομάστηκε αλγόριθμος και ορίστηκε ως: «μια πεπερασμένη σειρά ενεργειών αυστηρά καθορισμένων και εκτελέσιμων σε πεπερασμένο χρονικό όριο που στοχεύει στην επίλυση ενός προβλήματος». Πήρε το όνομά της προς τιμήν του Πέρση μαθηματικού Μοχάμεντ ιμπν Μουσά αλ-Χουαρίζμι που ήταν ο πρώτος που συστηματικά συγκέντρωσε και τυποποιημένες λύσεις για την επίλυση αλγεβρικών προβλημάτων.

Όπως γίνεται εύκολα κατανοητό ο όρος αλγόριθμος δεν περιορίζεται μόνο στην επίλυση μαθηματικών προβλημάτων αλλά περιλαμβάνει κάθε πρόβλημα το οποίο μπορεί να λυθεί σε εύλογο χρονικό διάστημα ακολουθώντας τα ίδια βήματα και προσδοκώντας πάντα το ίδιο αποτέλεσμα (ντετερμινιστική έκβαση της επίλυσης).

Ένας αλγόριθμος πρέπει να ικανοποιεί τα παρακάτω κριτήρια:

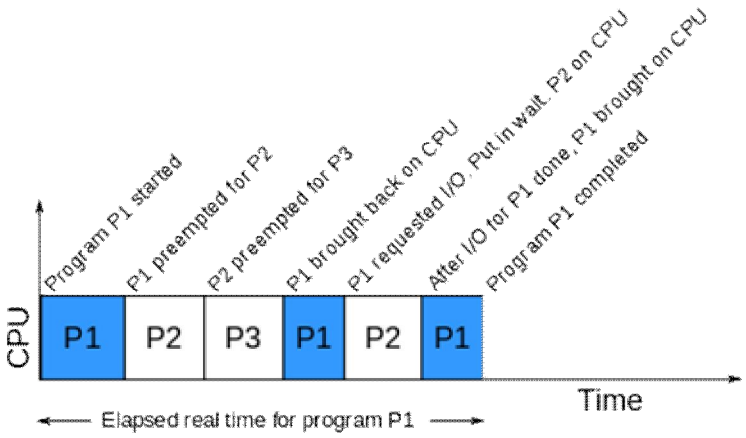
- **Περατότητα – Finiteness:** Κάθε εκτέλεση του αλγορίθμου πρέπει να ολοκληρώνεται σε πεπερασμένο χρονικό διάστημα.
- **Καθοριστικότητα – Definiteness:** Κάθε εντολή πρέπει να ορίζεται ακριβώς χωρίς να μπορεί να παρερμηνευτεί ο τρόπος με τον οποίο θα εκτελεστεί και να μπορεί να αντιμετωπίσει τυχόν ανωμαλίες που μπορεί να προκύψουν κατά την εκτέλεση αυτών.
- **Αποτελεσματικότητα – Effectiveness:** Κάθε βήμα / εντολή του αλγορίθμου πρέπει να σαφώς οριστεί και να είναι εκτελέσιμη.
- **Είσοδος Δεδομένων - Input Data:** Ένας αλγόριθμος μπορεί λαμβάνει κάποια δεδομένα ως είσοδο χωρίς αυτό όμως να είναι υποχρεωτικό (π.χ. μπορεί απλά να παράγει τυχαίους αριθμούς).
- **Έξοδος Δεδομένων – Output Data:** Όλοι οι αλγόριθμοι πρέπει να έχουν τουλάχιστον μία έξοδο με κάποια μορφή (π.χ. αριθμητικά δεδομένα, ενεργοποίηση μιας λειτουργίας κ.α.).

4.2 Βασικές αρχές επεξεργασίας

Οι ηλεκτρονικοί υπολογιστές έχουν μπει στην ζωή μας από τα μέσα του 20^{ου} αιώνα. Η τεράστια εξέλιξή τους έχει βοηθήσει στην επίλυση προβλημάτων για τα οποία οι άνθρωποι θα ήθελαν πολλά χρόνια να εκτελέσουν χωρίς την ταχύτητα επίλυσης που τους πρόσφεραν. «Bicycle for the mind» («Ποδήλατα για το μυαλό») όπως περιέγραφε ο Steve Jobs. Για να μπορέσει ο υπολογιστής να επιλύσει αυτά τα προβλήματα είναι απαραίτητο να γνωρίζει τα βήματα που πρέπει να ακολουθήσει για την επίλυσή τους, δηλαδή τον αλγόριθμο.

Οι σημερινοί ηλεκτρονικοί υπολογιστές μπορούν να καταλάβουν μόνο το δυαδικό αριθμητικό σύστημα (0 / 1). Ο τρόπος με τον οποίο η περιγραφή ενός αλγορίθμου από την φυσική γλώσσα που καταλαβαίνει ο άνθρωπος μεταφέρεται στην δυαδική που καταλαβαίνει ο υπολογιστής είναι μέσα από τις διάφορες γλώσσες προγραμματισμού που αναλαμβάνουν με την βοήθεια και του λειτουργικού συστήματος να γεφυρώσουν το παραπάνω χάσμα. Ο κώδικας μεταφράζεται από την γλώσσα προγραμματισμού που έχει γραφτεί (π.χ. C, C++, Java) μέσω κατάλληλου λογισμικού σε εντολές γλώσσας μηχανής και εκτελείται από την Κεντρική Μονάδα Επεξεργασίας (CPU). Για να δρομολογήσει την εκτέλεση των εντολών στην CPU το λειτουργικό σύστημα χρησιμοποιεί την έννοια του νήματος (thread). Ως νήμα ορίζεται η μικρότερη ανεξάρτητη ακολουθία εντολών που μπορεί να δρομολογήσει το λειτουργικό σύστημα. Το νήμα είναι αυτό το οποίο αναλαμβάνει την πραγματική εκτέλεση των εντολών στον πυρήνα της CPU.

Οι πρώιμες CPU (μέχρι τα τέλη της δεκαετίας του 1990) αποτελούνταν από ένα πυρήνα οπότε σε κάθε χρονική στιγμή μπορούσαν να εξυπηρετήσουν ένα μόνο νήμα. Όμως οποιοσδήποτε έχει δουλέψει σε παλιότερα μηχανήματα γνωρίζει πως υπήρχε η δυνατότητα εκτέλεσης πολλών προγραμμάτων «παράλληλα». Ο τρόπος με τον οποίο μονοπύρηνα συστήματα φαίνεται να εκτελούν παράλληλα περισσότερες διεργασίες, είναι μέσω της κατάτμησης του χρόνου σε μικρές υπομονάδες και της εκτέλεσης κάθε νήματος για ένα μικρό χρονικό διάστημα μετά από το οποίο το λειτουργικό σύστημα αλλάζει σε άλλο νήμα εκτέλεσης για άλλο ένα χρονικό διάστημα (εικόνα 20). Μέσω αυτής της διαδικασίας ο χρήστης έχει την ψευδαίσθηση πως τα προγράμματά του εκτελούνται παράλληλα. Όπως μπορεί εύκολα να γίνει κατανοητό, η εναλλαγή μεταξύ των διαφορετικών προγραμμάτων ώστε να φαίνεται πως εκτελούνται παράλληλα, εισάγει επιπλέον καθυστερήσεις στο σύστημα. Σε παλιότερα μονοπύρηνα συστήματα συχνές ήταν οι περιπτώσεις όπου οι εντολές κάποιου νήματος οδηγούσαν σε κάποια κατάσταση ανταγωνισμού με άλλο (race condition) με αποτέλεσμα ακόμη και την πλήρη παύση λειτουργίας του λειτουργικού συστήματος και την ανάγκη επανεκκίνησης του υπολογιστή.



Εικόνα 20: Παράδειγμα εκτέλεσης πολλών προγραμμάτων σε μονοπύρηνο επεξεργαστή με ένα νήμα μέσω διαμοιρασμού χρόνου.

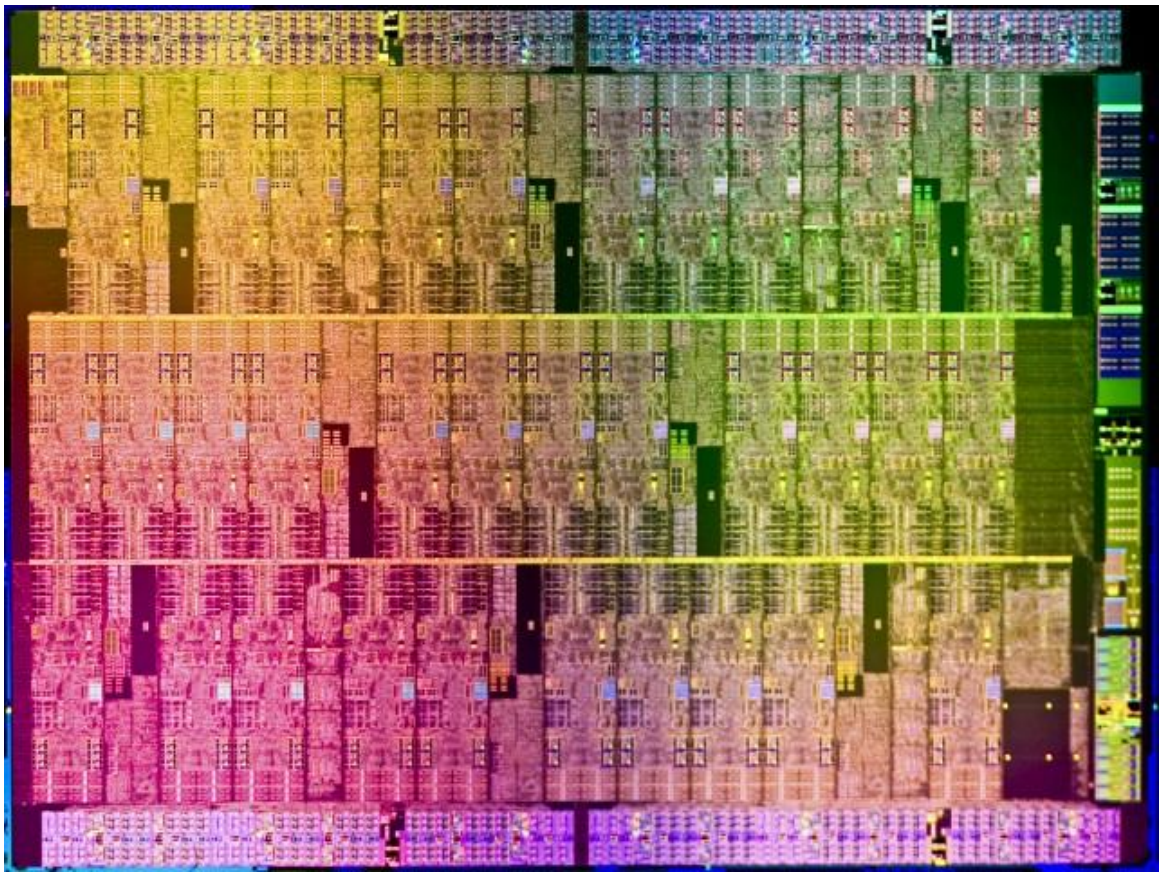
4.3 Παράλληλη Επεξεργασία

Ο παραλληλισμός των υπολογιστικών διεργασιών, παρότι έχει γίνει ευρέως διαδεδομένη τα τελευταία χρόνια, δεν είναι μια καινούρια ιδέα. Υπάρχει τουλάχιστον από τα μέσα του 1950.

Τις τελευταίες δεκαετίες οι επεξεργαστές γινόντουσαν όλο και γρηγορότεροι και πιο δυνατοί ακολουθώντας τον νόμο του Moore που όριζε πως κάθε 18 μήνες περίπου το ρολόι τους θα διπλασιαζόταν σε απόδοση. Η συρρίκνωση των τρανζίστορ εντός των chip των επεξεργαστών καθώς και οι πολύ υψηλές ταχύτητες αυτών, οδήγησαν στα τέλη της δεκαετίας του 1990 τους σχεδιαστές να αγγίξουν τα φυσικά όρια στα οποία μπορούσε να φτάσει ένας επεξεργαστής πριν την καταστροφή του ή την αλλοίωση των αποτελεσμάτων, μέσω της εμφάνισης κβαντικών φαινομένων ανάμεσα στα τρανζίστορ του.

Μπροστά στο αδιέξοδο οι κατασκευαστές των επεξεργαστών υιοθέτησαν μια διαφορετική προσέγγιση, ενσωματώνοντας περισσότερους από έναν πυρήνες εντός του chip της CPU. Σήμερα ο νόμος του Moore συνεχίζει να ισχύει, μόνο που πλέον αυτό που διπλασιάζεται ανά 18 μήνες περίπου είναι το πλήθος των επεξεργαστών σε μια μονάδα CPU.

Είναι πολύ σύνηθες πλέον σε οικονομικά συστήματα μερικών εκατοντάδων ευρώ να συναντώνται επεξεργαστές με πολλούς πυρήνες / νήματα επεξεργασίας (4, 8 ή και παραπάνω). Αυτή η αύξηση στην επεξεργαστική ισχύ μέσω της αύξησης των πυρήνων, ανάγκασε και τους δημιουργούς λογισμικού να προσεγγίζουν διαφορετικά τις υλοποιήσεις τους.

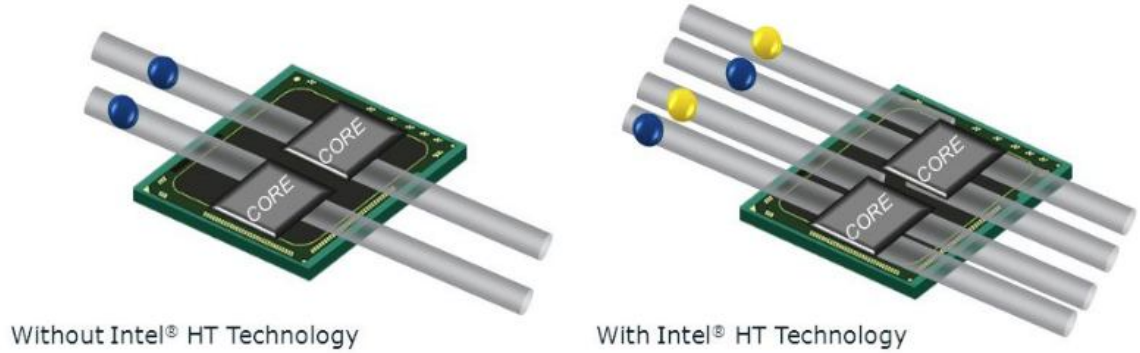


Εικόνα 21: Intel Knights Landing Many core CPU (72 cores x 4 threads)

Ο παραλληλισμός μπορεί να επιτευχθεί με αρκετούς τρόπους. Μερικοί από αυτούς είναι [13]:

- **Κατανεμημένος Παραλληλισμός:** Ολόκληρες εφαρμογές ή κομμάτια υπολογισμών μεταφέρονται για εκτέλεση σε άλλα συστήματα, ανταλλάσσοντας μεταξύ τους στοιχεία με πρωτόκολλα όπως η Διεπαφή Μετάδοσης Μηνύματος (MPI: Message Passage Interface)
- **Εικονικοποίηση (Virtualization):** Εκτέλεση πολλών λειτουργικών συστημάτων στον ίδιο υπολογιστή. Είναι απαραίτητη η ύπαρξη κατάλληλου λογισμικού (hypervisor) που αναλαμβάνει την διαχείριση των πόρων.
- **Παραλληλισμός σε επίπεδο διεργασίας:** Εστιάζει στην παραλληλοποίηση ολόκληρων διεργασιών. Συνήθως υπάρχουν περισσότερες διεργασίες απ' ότι νήματα εκτέλεσης (threads) οπότε υπάρχουν κατάλληλοι μηχανισμοί χρονοπρογραμματισμού ώστε κάθε διεργασία να έχει δίκαιη χρονική πρόσβαση σε κάποιο νήμα εκτέλεσης.
- **Παραλληλισμός σε επίπεδο νήματος:** Μια εφαρμογή μπορεί να επιλέξει να χρησιμοποιήσει πολλαπλά νήματα για την εκτέλεση των υπολογιστικών τις μερών.
- **Παραλληλισμός σε επίπεδο εντολής:** Για να μπορέσει να υλοποιηθεί θα πρέπει να υποστηρίζεται από τον επεξεργαστή και συνήθως γίνεται αυτόματα από αυτόν. Ο επεξεργαστής έχει την ικανότητα να τρέξει παράλληλα διαφορετικές εντολές εντός του.
- **Παραλληλισμός σε επίπεδο δεδομένων:** Για να μπορέσει να υλοποιηθεί ο συγκεκριμένος τύπος παραλληλοποίησης θα πρέπει ο επεξεργαστής να υποστηρίζει λειτουργίες Single Instruction Multiple Dataset. Βασίζεται στην ύπαρξη μεγάλων καταχωρητών πχ 128 bit οι οποίοι μπορούν να διατηρήσουν 4 αριθμούς 32 bit και να εκτελέσουν μια κοινή λειτουργία στον ίδιο κύκλο του ρολογιού και στους 4. Παραδείγματα τέτοιων τεχνολογιών είναι το AVX της Intel, το CUDA της NVIDIA και άλλα.

Όπως περιγράφηκε νωρίτερα για την επίλυση ενός προβλήματος ο υπολογιστής ακολουθεί καλά καθορισμένα βήματα που περιγράφουν τον τρόπο επίλυσης της συγκεκριμένης κλάσης προβλημάτων (σειριακή επίλυση). Αν το σύστημα έχει περισσότερες από μια υπολογιστικές μονάδες (VPU) τότε αυτές μπορούν να χρησιμοποιηθούν ώστε να εκτελέσουν τμήματα των υπολογισμών παράλληλα ώστε να ολοκληρωθεί ταχύτερα η επίλυση. Θεωρητικά αν είχαμε N υπολογιστικές μονάδες θα μπορούσαμε να μειώσουμε τον χρόνο εκτέλεσης στο 1/N.



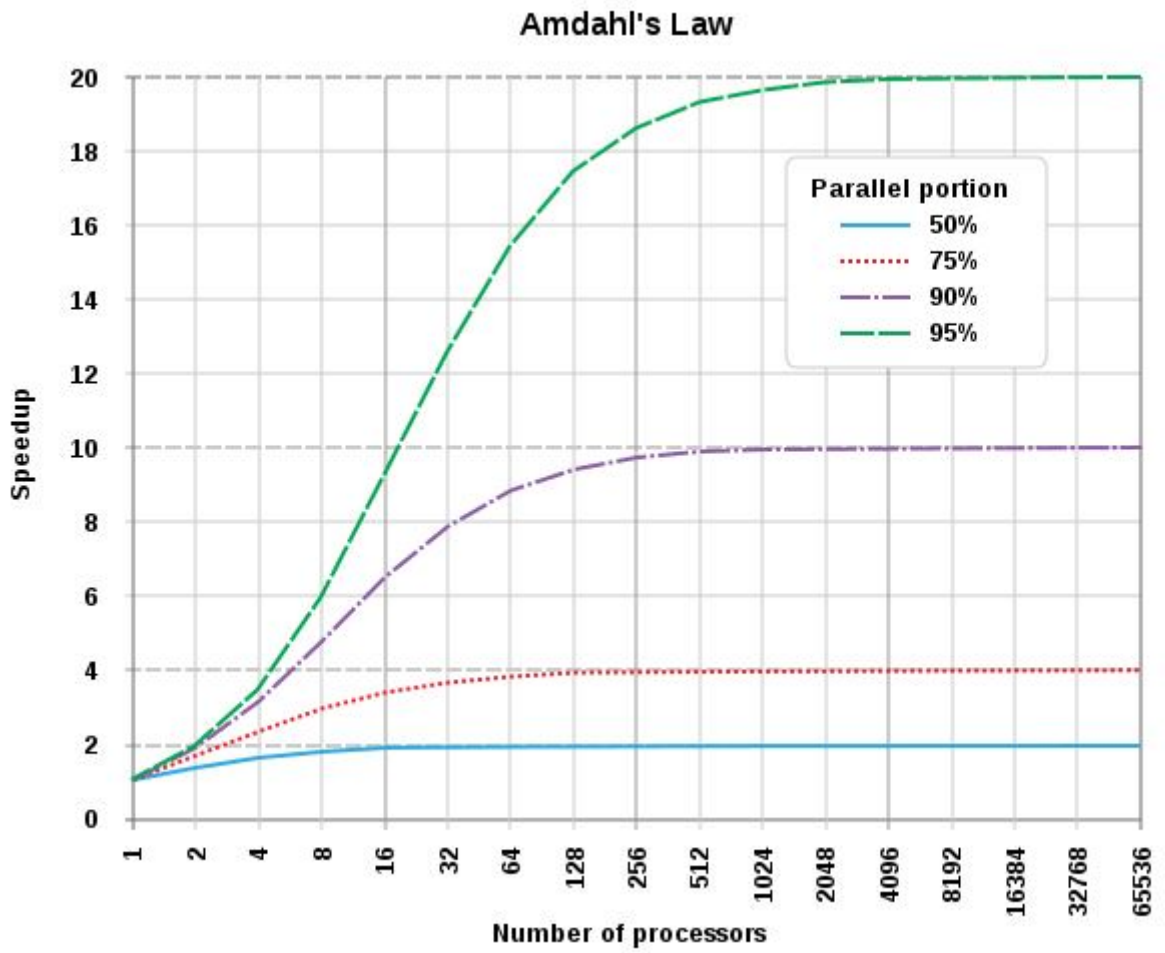
Εικόνα 22: Παραδείγματα 2-πύρηνων επεξεργαστών με 1 (N=2) και 2 (N=4) νήματα ανά πυρήνα αντίστοιχα

Στην πράξη όμως η πραγματική απόδοση που θα πάρουμε είναι πολύ κατώτερη όπως είχε από νωρίς αποδείξει ο Amdahl στον ομώνυμο νόμο του (εικόνα 23) και θα εξαρτάται πλήρως από την αναλογία χρόνου του μέρους του υπολογισμού που δεν μπορούμε να παραλληλοποιήσουμε (εικόνες 23 και γράφημα 1) .

$$S_{\text{latency}}(s) = \frac{1}{(1-p) + \frac{p}{s}}$$

$$\begin{cases} S_{\text{latency}}(s) \leq \dfrac{1}{1-p} \\ \lim\limits_{s \to \infty} S_{\text{latency}}(s) = \dfrac{1}{1-p}. \end{cases}$$

Εικόνα 23: Νόμος του Amdahl



Γράφημα 1: Εξέλιξη της αύξησης απόδοσης βάσει του νόμου του Amdahl

Επιπλέον της προηγούμενης παρατήρησης, ο συγχρονισμός των νημάτων, ειδικά όταν αυτά θα πρέπει να ανταλλάξουν μεταξύ τους πληροφορία μπορεί να υποβαθμίσει κατά πολύ την απόδοση ενός παράλληλου αλγορίθμου. Η μετατροπή λοιπόν των ακολουθιακών αλγορίθμων σε παράλληλους θα πρέπει να γίνεται με γνώμονα την μεγαλύτερη δυνατή ανεξαρτησία μεταξύ της εκτέλεσης σε κάθε νήμα, καθώς και της ελάχιστης ανάγκης ανταλλαγής δεδομένων μεταξύ των νημάτων. Η πιο αργή διαδικασία είναι αυτή που πάντα θα προσδιορίζει τον απαιτούμενο χρόνο.

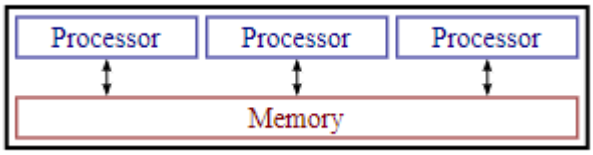
Στην παρούσα εργασία η παραλληλοποίηση που σχεδιάστηκε είναι σε επίπεδο νήματος. Μερικές από τις σημαντικότερες προκλήσεις όσων αφορά τον καλό σχεδιασμό ενός παράλληλου συστήματος σε επίπεδο νήματος είναι:

- Καλός διαχωρισμός των τμημάτων εκτέλεσης ώστε το τμήμα N να μην εξαρτάται από το N-1 και αντίστροφα
- Ελαχιστοποίηση των αλληλεπιδράσεων μεταξύ των νημάτων.
- Ελαχιστοποίηση της πρόσβασης των νημάτων σε κοινούς πόρους, ειδικά αν πρόκειται για αργά υποσυστήματα όπως της αποθήκευσης σε αρχεία στον σκληρό δίσκο.
- Καλή κατανομή των δεδομένων μεταξύ των νημάτων ώστε να υπάρχει αναλογικότητα στον χρόνο εκτέλεσης κάθε νήματος.

4.4 Κατανεμημένα Συστήματα

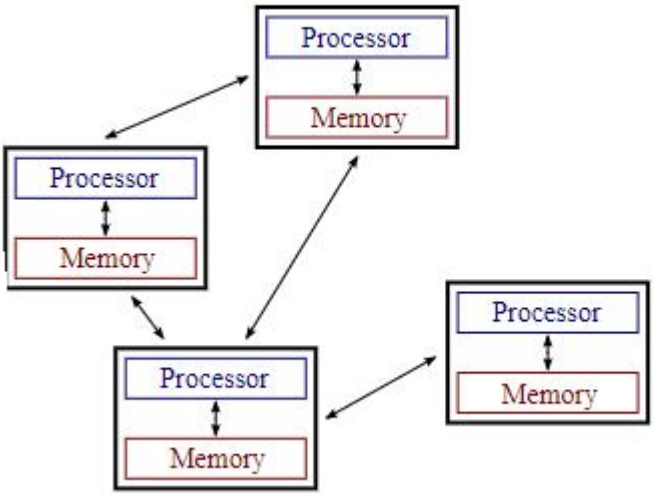
Στην προηγούμενη ενότητα αναφέρθηκε ως τύπος παραλληλισμού ο κατανεμημένος παραλληλισμός (distributed parallelism). Ως κατανεμημένο ορίζεται ένα σύστημα τα συστατικά του οποίου βρίσκονται σε διαφορετικές διαδικτυακές τοποθεσίες και επικοινωνούν συγχρονίζοντας την λειτουργίας τους μέσα από την ανταλλαγή μηνυμάτων μεταξύ τους αποβλέποντας στην ολοκλήρωση ενός κοινού στόχου.

Μια σημαντική διαφορά των κατανεμημένων και των παράλληλων συστημάτων είναι η πρόσβαση που έχουν οι μονάδες επεξεργασίας κάθε συστήματος στην μνήμη RAM. Στα παράλληλα συστήματα που αποτελούνται από διαφορετικές CPU εντός του ίδιου φυσικού μηχανήματος, οι πυρήνες έχουν πρόσβαση σε μια κοινή μνήμη μέσω της οποίας μπορούν να ανταλλάξουν γρήγορα μηνύματα.



Εικόνα 24: Ανταλλαγή μηνυμάτων σε παράλληλο σύστημα

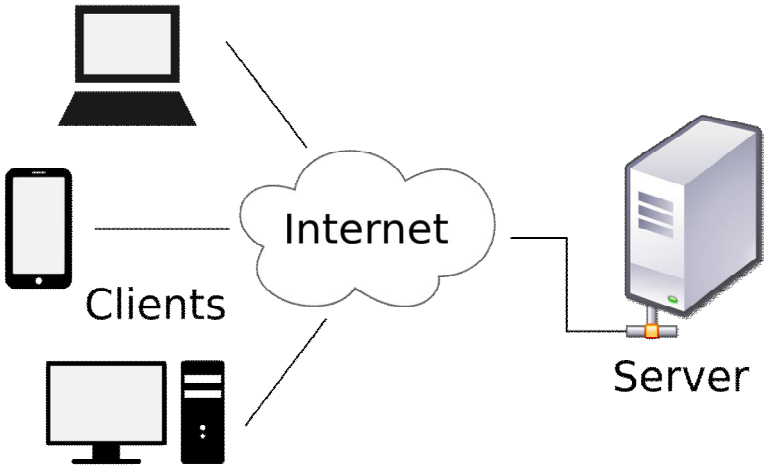
Αντίθετα στα κατανεμημένα συστήματα κάθε επεξεργαστής έχει πρόσβαση μόνο στην δική του μνήμη RAM και μπορεί να ανταλλάξει μηνύματα μόνο μέσω του πολύ πιο αργού δικτύου που συνδέει τα συστατικά του συστήματος μεταξύ τους. Φυσικά σε ένα κατανεμημένο σύστημα, τα επιμέρους συστατικά μπορούν να υλοποιούν και αυτά κάποια παράλληλη επεξεργασία εσωτερικά για επιτάχυνση των διεργασιών τους .



Εικόνα 25: Ανταλλαγή μηνυμάτων σε κατανεμημένο σύστημα

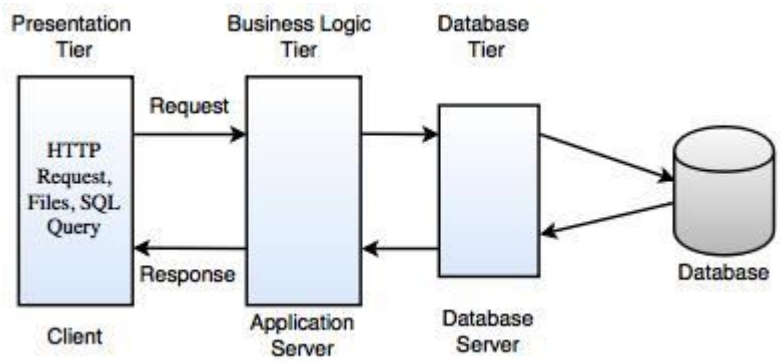
Υπάρχουν αρκετές διαφορετικές αρχιτεκτονικές για τον σχεδιασμό ενός κατανεμημένου συστήματος. Μερικές από τις πιο διαδεδομένες είναι οι:

- **Διακομιστής – Πελάτης (Client – Server):** Στην συγκεκριμένη αρχιτεκτονική γίνεται διαχωρισμός μεταξύ των διεργασιών που παρέχει ο προμηθευτής (server) μιας υπηρεσίας και αυτού ο οποίος την ζητάει (client)



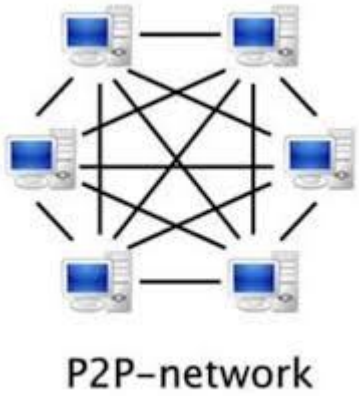
Εικόνα 26: Παράδειγμα αρχιτεκτονικής Client – Server

- **Πολυεπίπεδη αρχιτεκτονική (N-Tier):** Βασίζεται στην προαναφερθείσα αρχιτεκτονική πελάτη – διακομιστή όπου κάθε επίπεδο ορίζεται να επιτελεί ένα συγκεκριμένο μέρος των εντολών που μπορεί να έχει ένας πελάτης. Τα επίπεδα επικοινωνούν μεταξύ τους με σαφώς καθορισμένες διεπαφές. Κάθε επίπεδο μπορεί να επικοινωνήσει μόνο με τους άμεσους γείτονές του. Συνήθως χρησιμοποιείται για να απομονώσει μεταξύ τους λειτουργίες όπως η Διεπαφή χρήστη (Presentation Tier), η πρόσβαση στην Βάση Δεδομένων (Database Tier), η πρόσβαση στους επιχειρησιακούς κανόνες (Business Tier)



Εικόνα 27: Παράδειγμα αρχιτεκτονικής Πολλαπλών Επιπέδων (N-Tier)

- **Δίκτυο ομότιμων κόμβων (Peer to Peer):** Κάθε κόμβος σε ένα τέτοιο κατανεμημένο σύστημα μπορεί να λειτουργήσει και να διαμοιραστεί αρχεία ή άλλους πόρους χωρίς την ανάγκη ύπαρξης ενός κεντρικού διακομιστή. Ένας κόμβος εντός του συστήματος μπορεί να συνδεθεί με περισσότερους από έναν κόμβους, δημιουργώντας έτσι ένα πολύπλοκο δίκτυο κόμβων που εξασφαλίζει την βιωσιμότητα του δικτύου απέναντι στην κατάρρευση αρκετών από αυτούς.



Εικόνα 28: Παράδειγμα αρχιτεκτονικής ομότιμων λ κόμβων (Peer to Peer)

Ανεξάρτητα με την επιλεχθείσα αρχιτεκτονική ο σχεδιασμός ενός κατανεμημένου συστήματος πρέπει να γίνεται με γνώμονα την μεγαλύτερη δυνατή ανεξαρτησία κάθε συστατικού μέσα στο σύστημα, καθώς και την ελάχιστη δυνατή επικοινωνία μεταξύ τους. Ένα κατανεμημένο σύστημα θα πρέπει να έχει δικλείδες ώστε να μπορεί να συνεχίσει την επίλυση του προβλήματος που του έχει ανατεθεί ακόμη και αν κάποιο από τα συστατικά του βγει εκτός λειτουργίας

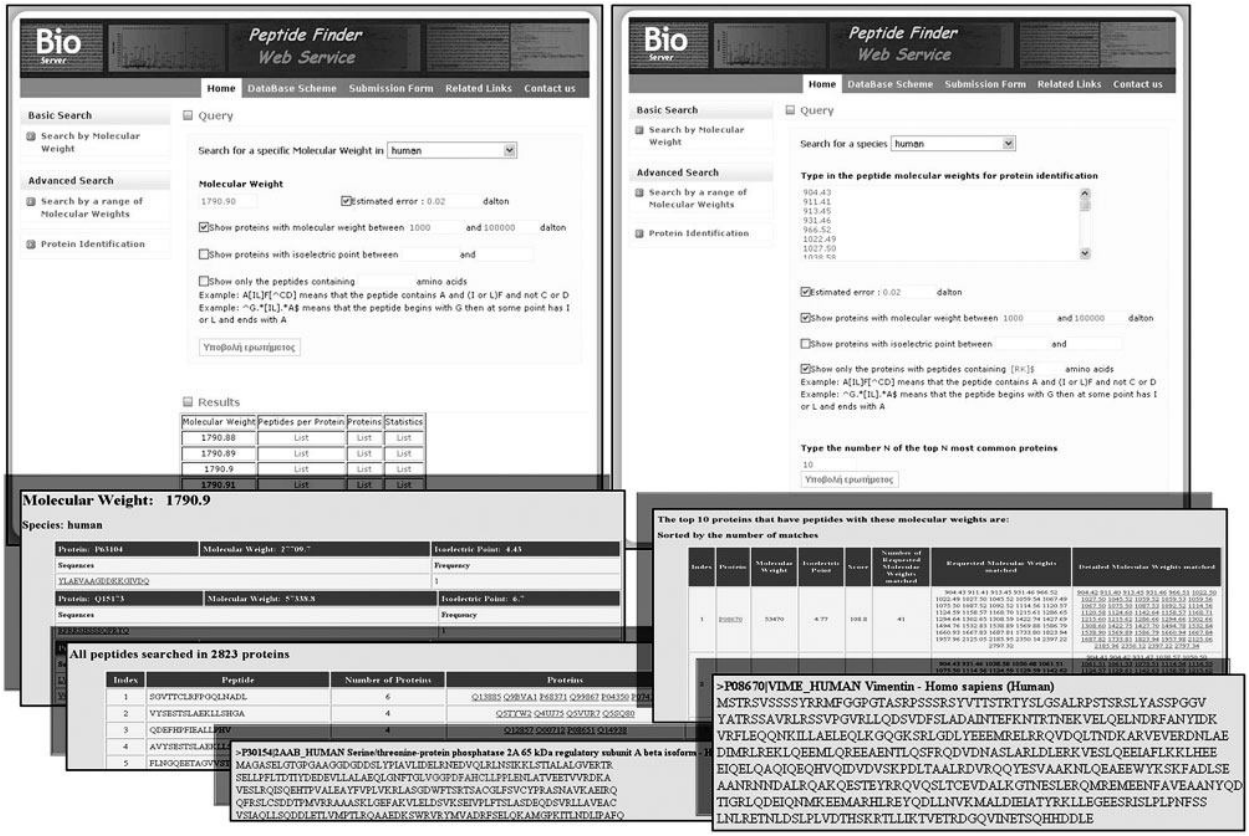
Τα κατανεμημένα συστήματα σήμερα είναι ευρέως διαδεδομένα λόγω της πληθώρας δεδομένων που συγκεντρώνονται στους σύγχρονους διακομιστές και πρέπει να αναλυθούν. Η ημερήσια παραγωγή δεδομένων το 2017 υπολογίζεται σε περίπου 2.5 exabytes[15]. Η μεταφορά δεδομένων πολλών Terrabyte (ίσως και Petabyte) σε κοινά μηχανήματα για την επεξεργασία τους είναι απαγορευτική και αυτό οδήγησε στην μεγάλη αποδοχή των κατανεμημένων συστημάτων από τον κόσμο της πληροφορικής. Η μεταφορά του αλγορίθμου υπολογισμού μεταξύ των συστατικών είναι πολύ φτηνότερη από την μεταφορά των δεδομένων. Ο πολύ πετυχημένος αλγόριθμος Map / Reduce είναι ένα εξαιρετικό παράδειγμα.

Μεγάλο προτέρημά των κατανεμημένων συστημάτων είναι επίσης η σχετικά εύκολη επεκτασιμότητα που παρέχουν. Αρκεί η προσθήκη νέων μηχανημάτων στο κατανεμημένο σύστημα για να επεκταθεί η συνολική επεξεργαστική και αποθηκευτική του ισχύς . Τέλος σημαντικό πλεονέκτημά τους είναι και η δυνατότητα να επεκταθούν από ετερόκλητα μεταξύ τους συστατικά μηχανήματα.

5. ΣΤΟΧΟΣ ΤΗΣ ΕΡΓΑΣΙΑΣ

5.1 Προηγούμενες εργασίες

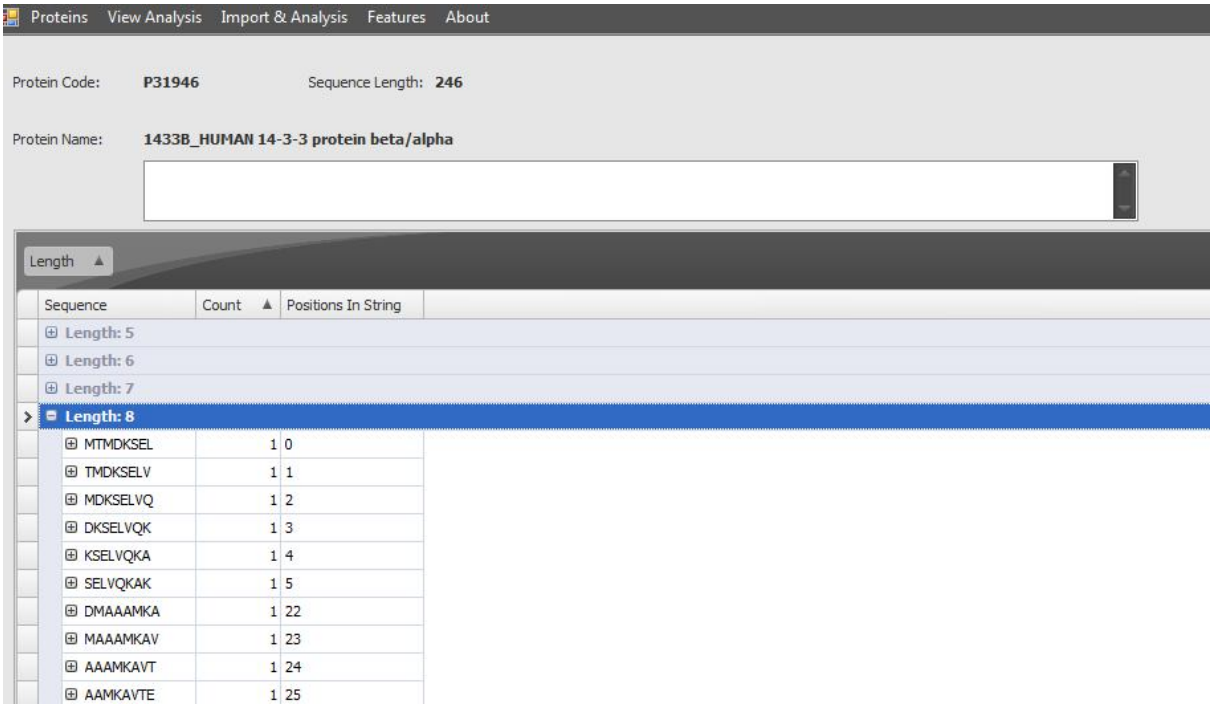
Όπως αναφέρθηκε παραπάνω, προηγούμενες έρευνες είχαν ορίσει τις έννοιες του Core Unique Peptide και του Composite Unique Peptide [9] [10]. Η πρώτη προσπάθεια ανάπτυξης μιας εφαρμογής που θα μπορούσε να εντοπίσει τα Core Unique πεπτίδια ενός πρωτεώματος ήταν το **UniMap** το οποίο υλοποιήθηκε σε CGI Perl και MySQL και ολοκληρώθηκε το 2009 [10]. Το UniMap αποτελεί μέρος μια ομάδας εργαλείων και βάσεων δεδομένων που αναπτύχθηκαν στο ΙΙΒΕΑΑ και λειτουργούν σε ένα Linux διακομιστή με προσβάσιμες από το διαδίκτυο υπηρεσίες που ονομάζεται BioServer [9]. Το UniMap έχει στηριχθεί στην ανάλυση του ανθρώπινου πρωτεώματος και μπορεί να ενημερώνει την βάση του κάθε φορά που ανανεώνεται η Swiss-Prot από την οποία αντλεί τα δεδομένα του. Επίσης παρέχει ένα Web Interface το οποίο μπορούν να προσπελάσουν οι χρήστες και να αναζητήσουν πεπτίδια για την μοναδικότητά τους εντός του ανθρώπινου πρωτεώματος. Τέλος παράλληλα με την αναζήτηση με χρήση της πεπτιδικής ακολουθίας, μπορούσε να εντοπίσει μοναδικά πεπτίδια χρησιμοποιώντας το μοριακή τους μάζα [9].



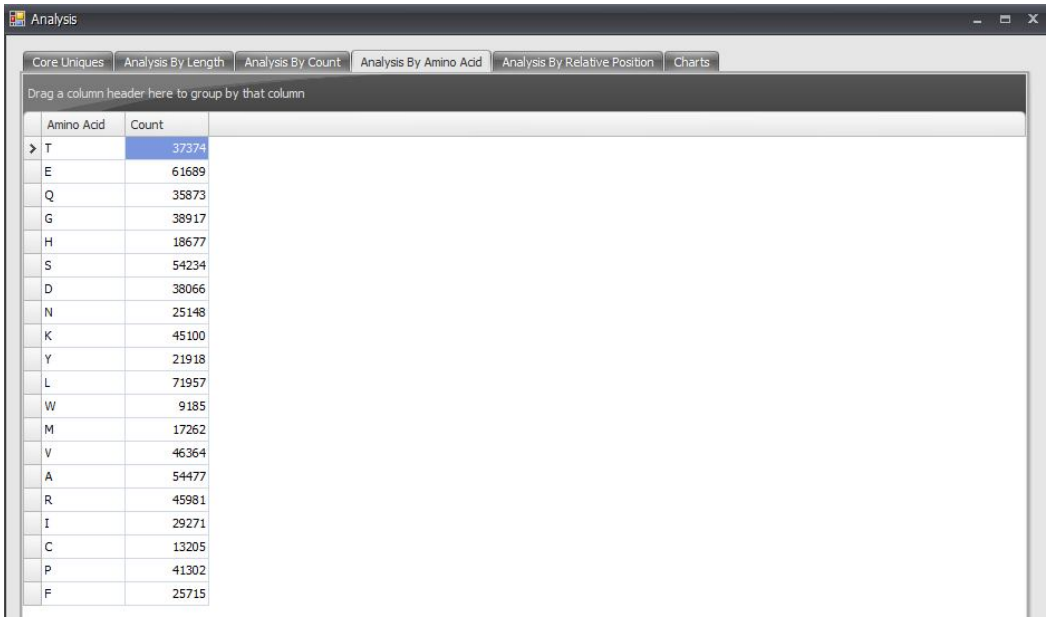
Εικόνα 29: Web Interface του BioServer

Το 2015 ο αλγόριθμος του UniMap υλοποιήθηκε εκ νέου στην προσπάθεια της ομάδας να εντοπίσει τα Composite Unique peptides αλλά και να αναλύσει τα ποιοτικά χαρακτηριστικά τόσο των Core όσο και τον Composite Unique peptides εντός του ανθρώπινου πρωτεώματος [11]. Η νέα εφαρμογή ονομάστηκε **Protein Analysis (vpierros)** και υλοποιήθηκε σε Microsoft .NET 4 με γλώσσα C# ενώ τα αποτελέσματα αποθηκεύτηκαν σε Microsoft SQL Server. Ο χρήστης μπορούσε να αποθηκεύσει

διαφορετικές εκδόσεις του πρωτεώματος και κατ' επέκτασιν και του Uniquome. Επιπλέον η νέα εφαρμογή επέτρεπε την ελεύθερη επιλογή σχεσιακού συστήματος βάσεων δεδομένων (RDBMS). Τέλος μπορούσε να εκτελεστεί σε Microsoft Windows διευκολύνοντας έτσι την χρήση της από χρήστες μη εξοικειωμένους με το λειτουργικό σύστημα Linux.



Εικόνα 30: Windows Based Interface για την εφαρμογή Protein Analysis



Εικόνα 31: Ανάλυση CUP ανά Αμινοξύ στην εφαρμογή Protein Analysis

5.2 Προβλήματα και προκλήσεις

Η εμπειρία της ομάδας από τις δυο παραπάνω έρευνες ανέδειξε μερικά προβλήματα καθώς και ενδιαφέρουσες προκλήσεις για τον επαναπροσδιορισμό της κατασκευής του

Uniquome του ανθρώπινου πρωτεώματος και την επέκταση αυτού σε άλλους οργανισμούς

Ένα από τα βασικά μειονεκτήματα των παραπάνω υλοποιήσεων ήταν ο πολύ μεγάλος χρόνος που απαιτείται για την κατασκευή του Uniquome του ανθρώπινου πρωτεώματος. Ενδεικτικά στην πρώτη υλοποίηση (UniMap) ο χρόνος εκτέλεσης υπερέβαινε τις 3 μέρες στον πολυπύρηνο διακομιστή του IIBEAA. Η δεύτερη υλοποίηση αν και σχετικά γρηγορότερη, απαιτούσε λίγο παραπάνω από 48 ώρες για την κατασκευή του uniquome, παρέμενε αρκετά αργή καταναλώνοντας μεγάλα ποσά μνήμης. Το πλεονέκτημα της όμως ήταν πως ο καλύτερος αυτός χρόνος επιτεύχθηκε σε μηχανήματα υποδεέστερης υπολογιστικής ισχύος, συγκεκριμένα σε ένα οικιακό μηχάνημα με 8 πυρήνες και 8 GB RAM. Ο αλγόριθμος που υλοποιήθηκε και στις δύο εφαρμογές δεν εμπεριείχε στον σχεδιασμό του καμία πρόβλεψη για παράλληλη επεξεργασία, με αποτέλεσμα η μόνη αύξηση της απόδοσης σε συστήματα πολλών επεξεργαστών/ πυρήνων να προέρχεται από το λειτουργικό σύστημα η οποία ήταν ελάχιστη.

Το UniMap αποθήκευε την ανάλυση της τελευταίας έκδοσης του ανθρώπινου πρωτεώματος της Swiss-Prot και μπορούσε να κατασκευάσει το uniquome και άλλων οργανισμών, ενώ το Protein Analysis μπορούσε μεν να αποθηκεύσει διαφορετικές εκδόσεις του πρωτεώματος καθώς και του uniquome, αλλά όλα θεωρούνταν πως ανήκουν στο συνολικό πρωτέωμα του Άνθρωπου.

Ο αλγόριθμος που υλοποιούσαν οι προηγούμενες εφαρμογές δεν παρείχε ευελιξία ως προς τον ορισμό της μοναδικότητας. Η ελάχιστη «μονάδα» που έπρεπε να εξεταστεί για το αν περιέχει ή όχι ένα πεπτίδιο ήταν η πρωτεΐνη. Όμως η περαιτέρω ανάλυση της βιολογικής σημασίας του Uniquome ανέδειξε και άλλες επιθυμητές αναζητήσεις όπως η μοναδικότητα ενός πεπτιδίου εντός μιας ομάδας πρωτεϊνών.

Παρότι το Protein Analysis διέθετε την δυνατότητα εξαγωγής μερικών βασικών στατιστικών επί του πρωτεώματος και του παραγόμενου από αυτό Uniquome, αυτά ήταν ελάχιστα και σχετικά δύσκολα επεκτάσιμα (απαιτούταν η εκ νέου εγκατάσταση νέας έκδοσης της εφαρμογής με τις όποιες βελτιώσεις). Η εξαγωγή ποιοτικών χαρακτηριστικών απαιτούσε πάλι την συγγραφή script σε perl / python για την μετανάλυση του uniquome.

Το UniMap μπορούσε να εκτελεστεί και σε οικιακούς υπολογιστές με λειτουργικό σύστημα Windows ή MacOs αρκεί να υπήρχε εγκατεστημένο κατάλληλο περιβάλλον για την Perl. Όμως η ολοκλήρωση της κατασκευής του σε αυτά τα μηχανήματα θα ήταν πολύ πιο χρονοβόρα, λόγω της συνήθους υποδεέστερης υπολογιστικής ισχύος αυτών των μηχανημάτων σε σύγκριση με τον διακομιστή όπου είχε σχεδιαστεί να εκτελείται.

Τέλος και οι δύο εφαρμογές μπορούσαν να εκτελεστούν μόνο σε ένα φυσικό (ή λογικό) μηχάνημα, αδυνατώντας να χρησιμοποιήσουν υπολογιστικούς πόρους που θα μπορούσαν να είναι διαθέσιμοι σε ένα εργαστήριο, επιταχύνοντας τόσο την διαδικασία της κατασκευής όσο και της ανάλυσης ενός uniquome.

Όλα τα παραπάνω οδήγησαν στην ανάγκη επαναπροσδιορισμού της προσέγγισης για την κατασκευή του uniquome ενός οργανισμού.

5.3 Νέοι στόχοι

Ο κύριος στόχος της παρούσας διπλωματικής είναι η δημιουργία ενός νέου και επεκτάσιμου αλγορίθμου ο οποίος θα μπορεί να εκτελείται παράλληλα αλλά και κατανεμημένα, για την κατασκευή του Uniquome ενός οργανισμού ή συστήματος. Επιπλέον ο αλγόριθμος θέλουμε να έχει την δυνατότητα προαιρετικού καθορισμού

υπό-ομάδων του δοθέντος πρωτεώματος ώστε να μπορεί να βρίσκει μοναδικότητες εντός αυτών (π.χ. οικογένειες πρωτεϊνών).

Επόμενος στόχος είναι η υλοποίηση του παραπάνω αλγορίθμου σε ένα σύγχρονο υπολογιστικό σύστημα που θα μπορεί να εκμεταλλεύεται με τον καλύτερο δυνατό τρόπο το υποκείμενο υλικολογισμικό (software / hardware) του υπολογιστή στον οποίο θα εκτελείται. Το νέο αυτό σύστημα το ονομάσαμε **Uniquome Analysis.**

Για το Uniquome Analysis ορίσαμε τις παρακάτω τεχνικές προδιαγραφές:

- Δυνατότητα να εκτελείται σε όλα τα σύγχρονα υπολογιστικά συστήματα από SoC όπως το RaspberryPI , μέχρι μεγάλα συστήματα διακομιστών με πολλαπλούς επεξεργαστών.
- Δυνατότητα να εκτελείται και στα τρία κύρια σύγχρονα λειτουργικά συστήματα Windows, Linux και MacOS με τον ίδιο τρόπο.
- Δυνατότητα επέκτασης της εκτέλεσης σε περισσότερα από ένα φυσικά ή λογικά μηχανήματα (scalability).
- Εύκολη εγκατάσταση και παραμετροποίηση και χρήση χωρίς την ανάγκη ύπαρξης τρίτων συστημάτων (π.χ. RDBMS).
- Δυνατότητα επέκτασης της λειτουργικότητας της εφαρμογής με χρήση πρόσθετων αρχείων χωρίς να είναι απαραίτητη η επανεγκατάσταση της εφαρμογής (add-ons).
- Δυνατότητα χρήσης της εφαρμογής από την γραμμή τερματικού (terminal) αλλά και ανάπτυξη γραφικής διεπαφής (User Interface).
- Αποθήκευση των αποτελεσμάτων σε αρχεία κειμένου για την εύκολη μεταφορά αλλά και χρήση τους από οποιοδήποτε υπολογιστή που μπορεί να επεξεργαστεί κείμενο.

Επιπλέον ορίσαμε τις παρακάτω λειτουργίες:

- Δυνατότητα στον ερευνητή να ορίζει το σύστημα ή τον οργανισμό που θέλει να μελετήσει κατασκευάζοντας το uniquome του.
- Δυνατότητα ορισμού υπό-ομάδων εντός ενός πρωτεώματος για την ανεύρεση του Uniquome με ελάχιστη μονάδα αναζήτησης την υπό-ομάδα και όχι την πρωτεΐνη
- Δυνατότητα σύγκρισης μεταξύ δύο ή περισσοτέρων Uniquome για την ανεύρεση κοινών πεπτιδίων.

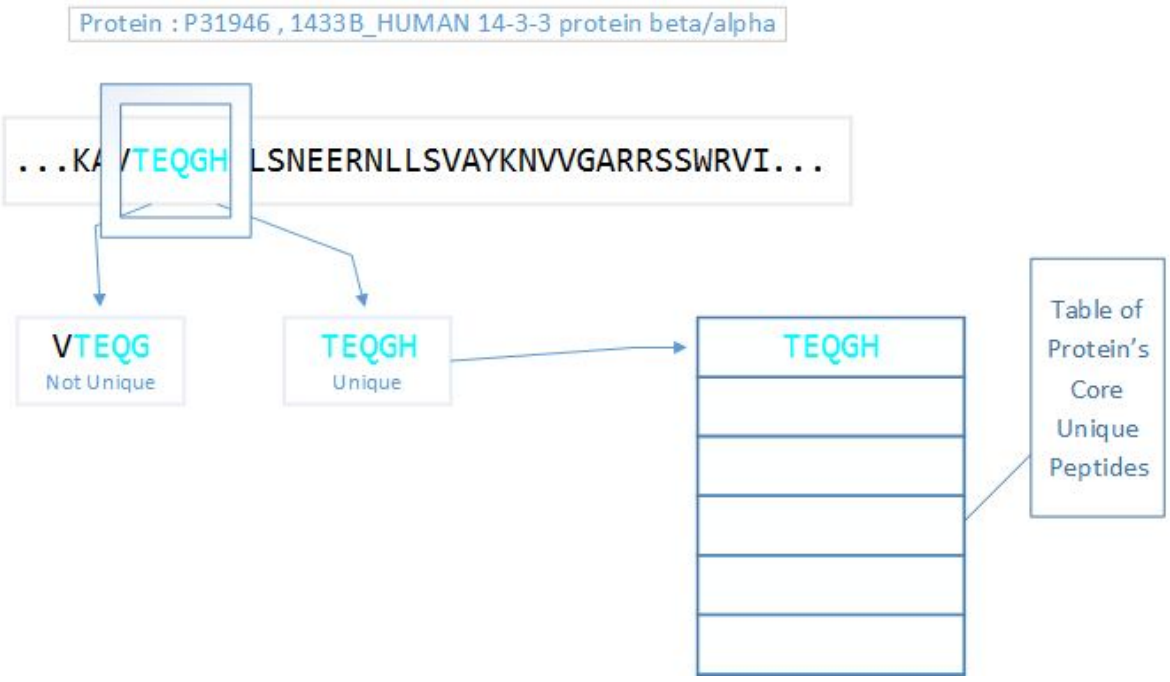
6. ΑΚΟΛΟΥΘΙΑΚΟΣ ΑΛΓΟΡΙΘΜΟΣ

6.1 Περιγραφή Ακολουθιακού Αλγορίθμου (Sequential Algorithm)

Ο πρώτος αλγόριθμος που σχεδιάστηκε και υλοποιήθηκε στο UniMap και το Protein Analysis ήταν σειριακός και μπορεί να περιγράφει από το παρακάτω κομμάτι ψευδοκώδικα:

- Για κάθε πρωτεΐνη P σε μία σύνολο πρωτεϊνών
  - Για K από X μέχρι Y
    - Δημιούργησε ένα κυλιόμενο παράθυρο μεγέθους K και ολίσθησέ το στην ακολουθία της πρωτεΐνης P
      - Πάρε το πεπτίδιο C κάτω από το παράθυρο. Αν το πεπτίδιο C περιέχει ήδη ένα Core Unique peptide τότε σταμάτα την αναζήτηση.
      - Για κάθε πρωτεΐνη M στο ίδιο σύνολο πρωτεϊνών εκτός της P
        - Αν το πεπτίδιο C υπάρχει στην πρωτεΐνη Mf τότε σταμάτα την αναζήτηση.
      - Αν το πεπτίδιο C δεν υπάρχει σε καμία άλλη πρωτείνη τότε είναι Core Unique peptide και αποθήκευσέ το.

Στην παρακάτω εικόνα μια οπτική εκτέλεση του αλγορίθμου όπως περιγράφηκε.



Εικόνα 32: Προσπέλαση πρωτεΐνης για ανεύρεση Core Unique Peptide

Ο αλγόριθμος δημιουργεί παράθυρα μεγέθους **M** μεταξύ δυο προκαθορισμένων μηκών X και Y όπου σύνηθες τιμές αυτών είναι το X= 4, Y=100. Η επιλογή αυτών των τιμών

έχει γίνει από την βιβλιογραφία. Στην συνέχεια για κάθε παράθυρο εξάγει ένα κατάλληλο πεπτίδιο και ελέγχει αν αυτό περιέχει μέσα του κάποιο άλλο Core Unique Peptide. Αν δεν περιέχει τότε αναζητεί στις υπόλοιπες πρωτεΐνες αν υπάρχει ή όχι το πεπτίδιο. Σε περίπτωση που το βρει σταματάει την αναζήτηση για το συγκεκριμένο πεπτίδιο. Αν αυτό δεν βρεθεί θεωρείται Core Unique και αποθηκεύεται.

6.2 Έκταση Προβλήματος

Ο παραπάνω αλγόριθμος για κάθε παράθυρο μεγέθους W σε μια πρωτεΐνη μεγέθους L δημιουργεί ένα σύνολο **C = L – W + 1,** διαφορετικών πεπτιδίων που πρέπει να αναζητηθούν εντός του συνόλου των υπολοίπων πρωτεϊνών ως προς την μοναδικότητα.

Το ανθρώπινο πρωτέωμα αποτελείται από 20,192 πρωτεΐνες (UniPort 8 /2013) με μεγέθη από 4 έως 34.350 αμινοξέα. Με βάση τον παραπάνω συλλογισμό για την μεγαλύτερη πρωτεΐνη (**TITIN**) πρέπει να πραγματοποιηθούν 3.326.809 αναζητήσεις. Στο σύνολο του ανθρώπινου πρωτεώματος τα διαφορετικά πεπτίδια που θα πρέπει να ερευνηθούν ως προς την μοναδικότητα πλησιάζουν τα 500.000.000. Αυτό που εύκολα γίνεται αντιληπτό είναι πως ενώ για ένα μη μοναδικό πεπτίδιο η αναζήτηση μπορεί να ολοκληρωθεί σχετικά σύντομα, για να αποδειχτεί η μοναδικότητα ενός πεπτιδίου είναι υποχρεωτικό να εξετασθεί πλήρως όλο το σύνολο των πρωτεϊνών. Η τελευταία παρατήρηση καθώς και το μέγεθος του πρωτεώματος είναι αυτές που καθιστά τον σειριακό αλγόριθμο μη αποδοτικό χρονικά.

6.3 Μειονεκτήματα Ακολουθιακού Αλγορίθμου

- Αργός
- Απαιτεί μεγάλα ποσά μνήμης
- Δεν μπορεί να κάνει χρήση περισσοτέρων πόρων (scaling)
- Δεν μπορεί να εκμεταλλευτεί τις νέες τάσεις του hardware (multi-cores)
- Ίδιος χρόνος εκτέλεσης σε multi/many core συστήματα
- Μνήμη / Ταχύτητα: Βελτίωση ταχύτητας απαιτεί δέσμευση μεγαλύτερων ποσών μνήμης
- Η χρήση Αρχείων / ΒΔ ως ενδιάμεση αποθήκευση είναι ιδιαίτερα αργή λόγω του αλληλεπίδρασης με ένα αργό μέσω όπως ο σκληρός δίσκος (η χρήση SSD μπορεί να επιταχύνει την διαδικασία).

7. ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΛΕΙΤΟΥΡΓΙΩΝ ΕΝΤΟΣ ΤΟΥ ΙΔΙΟΥ ΝΗΜΑΤΟΣ ΕΚΤΕΛΕΣΗΣ (SINGLE THREAD OPTIMIZATIONS)

7.1 Βελτιστοποίηση αναζήτησης Πεπτιδίου σε Πρωτεΐνη

Όπως προκύπτει από τον ακολουθιακό αλγόριθμο που περιγράφηκε πριν, μια από τις βασικότερες λειτουργίες που πρέπει να εκτελεστεί είναι η αναζήτηση ενός πεπτιδίου μέσα σε μία πρωτεΐνη. Ασφαλής απάντηση για την ύπαρξη του ή όχι μπορεί να δοθεί μόνο όταν αναζητηθεί κατά μήκος όλης της ακολουθίας της πρωτεΐνης. Για να γίνει καλύτερα κατανοητό το μέγεθος του προβλήματος ας υποθέσουμε ότι αναζητούμε το 5-πεπτίδιο **ELLLK** στην TITIN που έχει μέγεθος 34.350 αμινοξέων και η οποία δεν το περιέχει. Για να μπορέσει ο αλγόριθμος αναζήτησης να μας επιστρέψει αυτή την απάντηση, θα πρέπει να συγκρίνει το ELLK κατά μήκος της ακολουθίας ξεκινώντας από την θέση 1 μέχρι την θέση 34.347. Θα εκτελέσει λοιπόν τουλάχιστον 34.347 συγκρίσεις. Φυσικά το πλήθος των συγκρίσεων θα είναι πολύ μεγαλύτερο αφού για κάθε γράμμα του πεπτιδίου που ταιριάζει με το σημείο της ακολουθίας που ελέγχει θα εκτελεί μία ακόμη σύγκριση όπως φαίνεται στο παρακάτω παράδειγμα από την αναζήτηση σε τμήμα της συγκεκριμένης πρωτεΐνης:

VAADKAKEQELKSRTKEVITTKQEQMHVTHEQIRKETEKTFVPKVVISAAKAKEQETR

ELLK

Σε αυτό το παράδειγμα για να αποκλειστούν τα αμινοξέα με την πορτοκαλί επισήμανση (**VAADKAK**) ως σημεία πιθανής έναρξης του πεπτιδίου ELLK απαιτείται μία σύγκριση για το καθένα, άρα 7 συγκρίσεις. Στην συνέχεια το επόμενο τμήμα EQ απαιτούνται 3 συγκρίσεις μια για το E που ταιριάζει με την πρώτη θέση του μια για να αποκλειστεί το Q που δεν ταιριάζει με το δεύτερο αμινοξύ L και μία για να αποκλειστεί το Q μετά την ολίσθηση του ELLK κατά μία θέση δεξιά στην ακολουθία ώστε να συγκριθεί το E με το Q. Κατ' αντιστοιχία για να αποκλειστεί το ELK απαιτούνται 4. Γίνεται εύκολα αντιληπτό το πολύ μεγάλο υπολογιστικό κόστος που πληρώνουμε για συγκρίσεις που ξέρουμε ότι δεν μπορούν να καταλήξουν σε ταίριασμα του πεπτιδίου με την ακολουθία αφού δεν ξεκινούν με το ίδιο γράμμα με το προς αναζήτηση πεπτίδιο.

Αντιμετωπίσαμε το παραπάνω πρόβλημα εν μέρει σαν πρόβλημα αναζήτησης λέξης σε λεξικό. Όταν ψάχνουμε για μια λέξη η πρώτη κίνηση είναι να πάμε στο σημείο όπου ξεκινούν οι λέξεις με ίδιο πρώτο γράμμα όπως η προς αναζήτηση λέξη. Με βάση αυτή την παρατήρηση το πρώτο βήμα του νέου αλγορίθμου είναι να προ επεξεργαστεί όλες τις πρωτεΐνες χαρτογραφώντας τις θέσεις που περιέχεται κάθε αμινοξύ. Δημιουργούμε έτσι ένα πίνακα όπως ο παρακάτω που αφορά την πρωτεΐνη P0AAW9 του οργανισμού e-coli με ακολουθία:

MLELLKSLVFAVIMVPVVMAIILGLIYGLGEVFNIFSGVGKKDQPGQNH

M	1	14	19				
L	1	3	4	7	8	23	29
E	3	31					
K	6	41	42				
S	7	37					
V	9	17	18	32	39		

Εικόνα 33: Τμήμα χαρτογράφησης αμινοξέων πρωτεΐνης P0AAW9

Έχοντας στην διάθεσή μας τον παραπάνω χάρτη μπορούμε να τροποποιήσουμε τον μέρος του αλγορίθμου που αναζητά το πεπτίδιο στην πρωτεΐνη ως εξής:

- Εντόπισε την λίστα των θέσεων που αντιστοιχούν στο πρώτο αμινοξύ του προς αναζήτηση πεπτιδίου.
- Για κάθε θέση από τη λίστα των θέσεων του αμινοξέος έλεγξε τα αμινοξέα που ξεκινούν από την θέση + 1 σε σχέση με τα αμινοξέα του πεπτιδίου μετά το πρώτο. Το πρώτο αμινοξύ δεν χρειάζεται να ελεγχθεί αφού η ταυτοποίησή του έχει επιτευχθεί μέσω του χάρτη. Επομένως για ένα πεπτίδιο μεγέθους L και για κάθε θέση του χάρτη θα γίνουν το πολύ L-1 συγκρίσεις.

Στο προηγούμενο παράδειγμα που παραθέσαμε για να αποκλειστεί η ύπαρξη του **ELLK** εντός του τμήματος της πρωτεΐνης αρκούν 10 συγκρίσεις. Μία για κάθε αμινοξύ **E** εντός της ακολουθίας, εκτός του τμήματος EL όπου χρειάζονται 2 αφού το L είναι κοινό μεταξύ του τμήματος και του πεπτιδίου. Για το ίδιο τμήμα ο προηγούμενος αλγόριθμος χρειαζόταν να εκτελέσει 77 συγκρίσεις.

Για κάθε πρωτεΐνη ο εντοπισμός της θέσης του προς αναζήτηση αμινοξέος στον χάρτη έχει πολυπλοκότητα O(1). Αυτό το πετυχαίνουμε καταναλώνοντας ένα μικρό ποσό μνήμης χρησιμοποιώντας όλα τα γράμματα της αγγλικής αλφαβήτου, ακόμη και αυτά που δεν αντιστοιχούν σε αμινοξέα, ενώ παράλληλα θεωρούμε ως συνάρτηση εύρεσης της θέσης στον πίνακα την f(γράμμα) = γράμμα – 64. Ο αριθμός 64 προκύπτει από την δεκαδική αναπαράσταση του χαρακτήρα πριν το γράμμα A στον ASCII πίνακα . Έτσι όταν αναζητήσουμε τον χάρτη για το αμινοξύ Q αντί να εκτελέσουμε μια επανάληψη εντός του χάρτη μέχρι να βρούμε το σημείο που περιέχει τις θέσεις του στην ακολουθία εκτελούμε την συνάρτηση f(Q) = 81 – 64 = 17

7.2 Αποκλεισμός Πεπτιδίων από την αναζήτηση αν είναι ή όχι CUP

Όπως περιγράφηκε στον ακολουθιακό αλγόριθμο για να μπορέσουμε να εντοπίσουμε τα CUP ξεκινούμε από ένα παράθυρο μεγέθους X (συνήθως 4) και συνεχίζουμε την αναζήτησή μας αυξάνοντας κατά 1 αμινοξύ το μέγεθος του παραθύρου. Ο λόγος για τον οποίο η αναζήτηση γίνεται με το μικρότερο δυνατό πεπτίδιο είναι για να αποκλειστούν από το uniquome πεπτίδια τα οποία περιέχουν μικρότερα Core Unique Peptides εντός τους. Αυτό επιτυγχάνεται ελέγχοντας όλα τα Core Unique Peptides της πρωτεΐνης που έχουν μέγεθος μικρότερο ή ίσο με το πεπτίδιο που αναζητούμε για το αν περιέχονται μέσα του. Για την αναζήτηση κάθε CUP εντός του νέου πεπτιδίου ισχύουν τα προβλήματα που αναφέρθηκαν στην προηγούμενη ενότητα, χωρίς όμως να είναι εφικτή η δημιουργία ενός μίνι χάρτη αμινοξέων για κάθε CUP που εντοπίζεται αφού αυτό κρίνεται ασύμφορο τόσο σε υπολογιστικούς κύκλους όσο και στην μνήμη που απαιτείται.

Η παραπάνω παρατήρηση μπορεί να λειτουργήσει θετικά αποκλείοντας μεγάλο μέρος των πεπτιδίων που θα πρέπει να ελεγχθούν για μοναδικότητα. Αν το παράθυρο που ελέγχουμε περιέχει ένα ή περισσότερα CUP τότε ο αλγόριθμος μπορεί να εκτελέσει ένα «άλμα» ξεκινώντας την επόμενή του αναζήτηση από το δεύτερο αμινοξύ του τελευταίου CUP που περιέχεται εντός του παραθύρου.

Ένα παράδειγμα αυτού φαίνεται στην παρακάτω εικόνα 34. Το προς έλεγχο παράθυρο είναι μεγέθους 14 αμινοξέων. Τα πεπτίδια **ELLK** και **FAVIM** είναι CUP, οπότε

οποιοδήποτε πεπτίδιο τα περιέχει δεν μπορεί να είναι επίσης CUP. Ο αλγόριθμος μπορεί να παραβλέψει το κομμάτι της ακολουθίας με το πράσινο πλαίσιο **MLELLKSLVF** και να ξεκινήσει την αναζήτηση αμέσως μετά από αυτό με πρώτο πεπτίδιο το **AVIMVPVVMAIILG**.

Εικόνα 34: Παράδειγμα αποκλεισμού πεπτιδίων από την αναζήτηση

Η τελευταία βελτιστοποίηση δίνει μεγάλη επιτάχυνση σε πρωτεΐνες οι οποίες έχουν σχετικά πολλά CUP ειδικά αν αυτά εμφανίζονται με μια μικρή διασπορά. Στην περίπτωση που η πρωτεΐνη δεν περιέχει κανένα CUP η επιτάχυνση είναι μηδενική ενώ αν περιέχει λίγα ή πολύ κοντά μεταξύ τους, το όφελος επίσης μειώνεται σημαντικά. Σε κάθε περίπτωση δεν προσθέτει παραπάνω υπολογιστικό κόστος, αφού η συγκεκριμένη αναζήτηση των μικρότερων CUP της πρωτεΐνης εντός του νέου προς αναζήτηση πεπτιδίου είναι υποχρεωτικό να γίνει, άρα μεγιστοποιούμε το όφελος του υπολογιστικού κόστους που ήδη πληρώνουμε γι' αυτή την αναζήτηση.

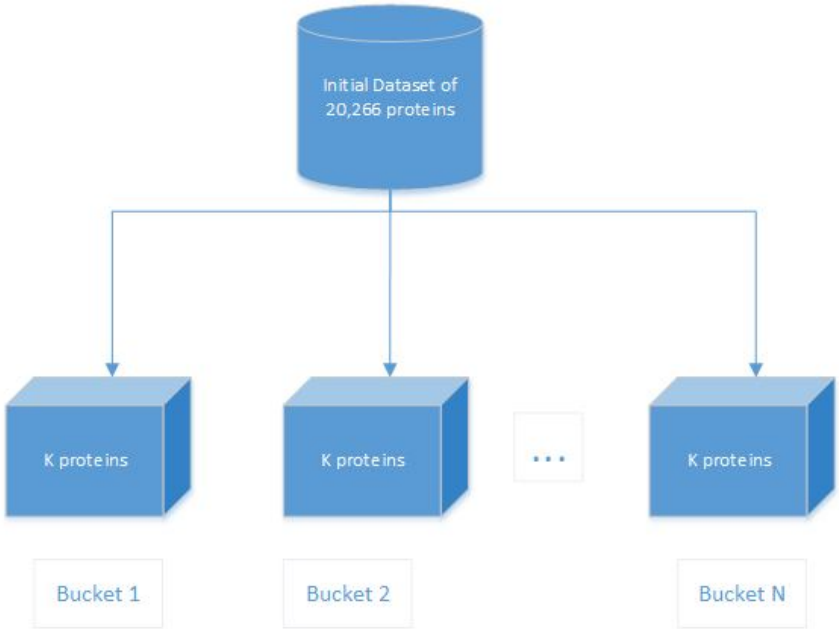
8. ΠΑΡΑΛΛΗΛΗ ΕΚΤΕΛΕΣΗ ΛΕΙΤΟΥΡΓΙΩΝ

Στην προηγούμενη ενότητα αναπτύχθηκαν οι βελτιστοποιήσεις του αλγορίθμου εντός του ίδιου νήματος εκτέλεσης που αφορούν την αναζήτηση εντός μιας πρωτεΐνης καθώς και τον χωρισμό των πεπτιδίων που τελικά θα πρέπει να αναζητηθούν για ύπαρξη ή όχι στις άλλες πρωτεΐνες του συνόλου. Σε αυτή την ενότητα θα αναλύσουμε τις βελτιστοποιήσεις που έγιναν στον αλγόριθμο ώστε να μπορεί να εκτελεστεί παράλληλα και να εκμεταλλευτεί τα σύγχρονα πολυπύρηνα συστήματα.

8.1 Βήμα 1 – Χωρισμός του πρωτεώματος

Όπως μπορούμε εύκολα να παρατηρήσουμε η αναζήτηση ενός πεπτιδίου σε κάθε πρωτεΐνη είναι μια εντελώς ανεξάρτητη διαδικασία. Αν θέλουμε να αναζητήσουμε ένα πεπτίδιο σε Χ πρωτεΐνες, η αναζήτηση στην πρωτεΐνη Κ -1 δεν επηρεάζεται από την αναζήτηση στην πρωτεΐνη Κ. Αυτό που έχει σημασία είναι πως αν το πεπτίδιο εντοπιστεί σε μία πρωτεΐνη θα πρέπει να σταματήσουμε την περαιτέρω αναζήτηση γι' αυτό.

Παρατηρήσαμε πως μπορούμε να χωρίσουμε το πρωτέωμα σε ομάδες και να αναθέσουμε σε κάθε επεξεργαστή την αναζήτηση εντός μιας ομάδας για ένα πεπτίδιο. Ονομάζουμε αυτά τα υποσύνολα κουβάδες (bucket). Αν θεωρήσουμε πως τα διαφορετικά σύνολα έχουν περίπου το ίδιο μέγεθος αμινοξέων τότε έχουμε πετύχει μια καλή κατάτμηση του προβλήματος ευελπιστώντας σε βελτίωση ταχύτητας που θεωρητικά μπορεί να προσεγγίσει το Ν, όπου Ν είναι το πλήθος των δημιουργημένων υποομάδων. Η εικόνα δείχνει τον χωρισμό των δεδομένων.



Εικόνα 35: Χωρισμός του πρωτεώματος σε Ν υποσύνολα.

Η επιλογή του Ν μπορεί να επηρεάσει αρκετά την απόδοση του αλγορίθμου και θα πρέπει να επιλέγεται σε συνάρτηση με τους διαθέσιμους λογικούς πυρήνες του επεξεργαστή στον οποίο θα γίνει η εκτέλεση. Αν το Ν είναι αρκετά μεγάλο τότε το εκάστοτε υποσύνολο θα περιέχει ένα μικρό σχετικά πλήθος πρωτεϊνών που αυξάνει τις

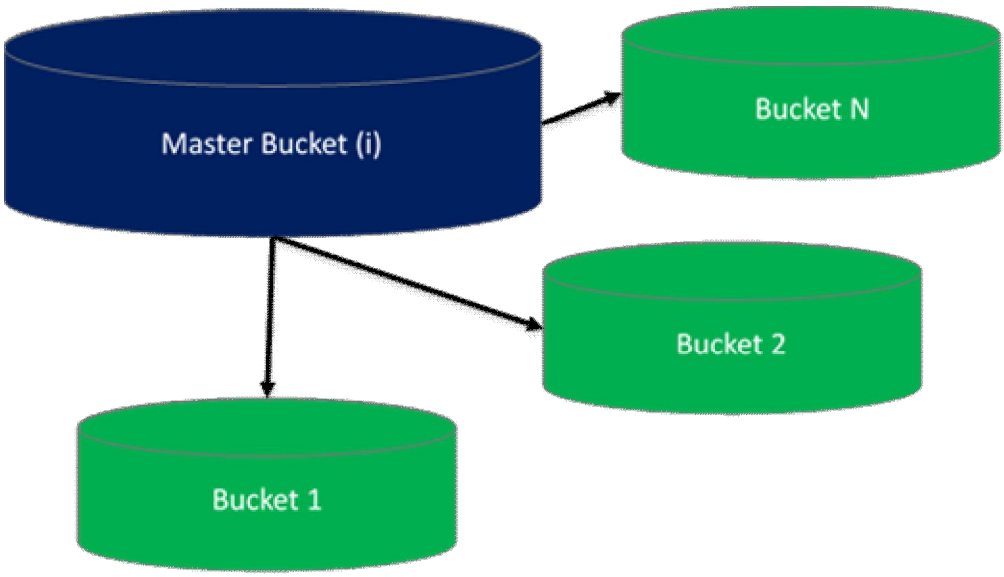
πιθανότητες να πετύχουμε νωρίς ένα πεπτίδιο. Για να γίνει καλύτερα κατανοητό τι εννοούμε με το τελευταίο, αν το προς αναζήτηση πεπτίδιο βρίσκεται στην θέση 15.000 του πρωτεώματος που αναλύουμε, τότε ο σειριακός αλγόριθμος θα πρέπει να ελέγξει 14.999 πρωτεΐνες μέχρι να βρει ότι υπάρχει πάλι το πεπτίδιο. Αν θέσουμε το N = 100 τότε κάθε κουβάς θα έχει να αναλύσει περίπου 200 πρωτεΐνες ανεβάζοντας κατά πολύ την πιθανότητα να βρεθεί πολύ σύντομα το πεπτίδιο, αφού θα είναι το πρώτο πεπτίδιο του κουβά 75. Όπως αναφέραμε όμως και νωρίτερα στην ενότητα 4 η εναλλαγή του λειτουργικού μεταξύ των νημάτων έχει κόστος. Όσο μεγαλύτερο είναι το N από το πλήθος των λογικών πυρήνων του επεξεργαστή τόσο θα ανεβάζει το κόστος εναλλαγής των νημάτων για αναζήτηση στους υπόλοιπους κουβάδες (buckets).

8.2 Βήμα 2 – Εκτέλεση

Ένα από τα βασικότερα προβλήματα των παράλληλων εφαρμογών είναι η ανάγκη για προσπέλαση κοινών πόρων ανά επεξεργαστική μονάδα. Όποτε χρειάζεται να γίνει αυτό, η μονάδα που θέλει να προσπελάσει τον πόρο πρέπει να κλειδώσει την πρόσβαση σε αυτόν ώστε να αποτρέψει την μη ορθή χρήση και την ενδεχόμενη φθορά των δεδομένων του. Το κλείδωμα αυτό όμως είναι ιδιαίτερα ακριβό σε πόρους συστήματος, ενώ, για την ώρα που το κλείδωμα είναι σε λειτουργία, ακυρώνεται η όποια παραλληλία της υλοποίησης αφού τα υπόλοιπα νήματα περιμένουν την ολοκλήρωση του κλειδώματος για να συνεχίσουν την λειτουργία τους.

Στην περίπτωση της δικής μας εφαρμογής ο κοινός πόρος είναι η βάση των Core Unique Peptides. Θέλουμε να βρούμε ένα τρόπο ώστε να μηδενίσουμε την ανάγκη για κλείδωμα των όποιων πόρων, άρα πρέπει να μειώσουμε την ανάγκη για ύπαρξη κοινών πόρων. Για να πετύχουμε τον στόχο μας «δανειστήκαμε» μέρος της λογικής του αλγορίθμου MAP-REDUCE.

Όπως αναφέρθηκε στο πρώτο βήμα, έχουμε χωρίσει το σύνολο των πρωτεϊνών σε μικρότερα υποσύνολα που ονομάσαμε «κουβάδες δεδομένων» (Buckets). Κάθε ένας από αυτούς εκτός από το να αποτελεί αποθετήριο (repository) ακολουθιών πρωτεϊνών, αποτελεί και μια ξεχωριστή υπολογιστική μονάδα. Η προσέγγισή μας είναι να μην εκτελεί κάθε κουβάς τον ίδιο ρόλο στην διάρκεια του υπολογισμού του Uniquome. Προσπελαύνουμε σειριακά το σετ των διαφορετικών buckets, και σε κάθε επανάληψη ο τρέχων θεωρείται ως Master. Μόνο αυτός έχει πρόσβαση στην βάση δεδομένων και σε όσους πόρους απαιτούν να γίνει κάποιου είδος κλείδωμα. Οι υπόλοιποι (Search Buckets) εκτελούν την ίδια λειτουργία μεταξύ τους που για το uniquome είναι η εύρεση ή όχι ενός πεπτιδίου στις πρωτεΐνες του αποθετηρίου τους, άρα δεν έχουν καμία αλληλεπίδραση μεταξύ τους. Μπορούμε εύκολα να αντιπαραβάλουμε την συγκεκριμένη αρχιτεκτονική με την Client- Server όπως αυτή περιγράφηκε στην ενότητα των κατανεμημένων συστημάτων, όπου ο Master Bucket έχει τον ρόλο του Client και οι Search Buckets τον ρόλο του Server.



Εικόνα 36: Αποστολή Περιεχομένου Ερώτησης από τον Master στους Search Buckets

8.3 Λειτουργία Master

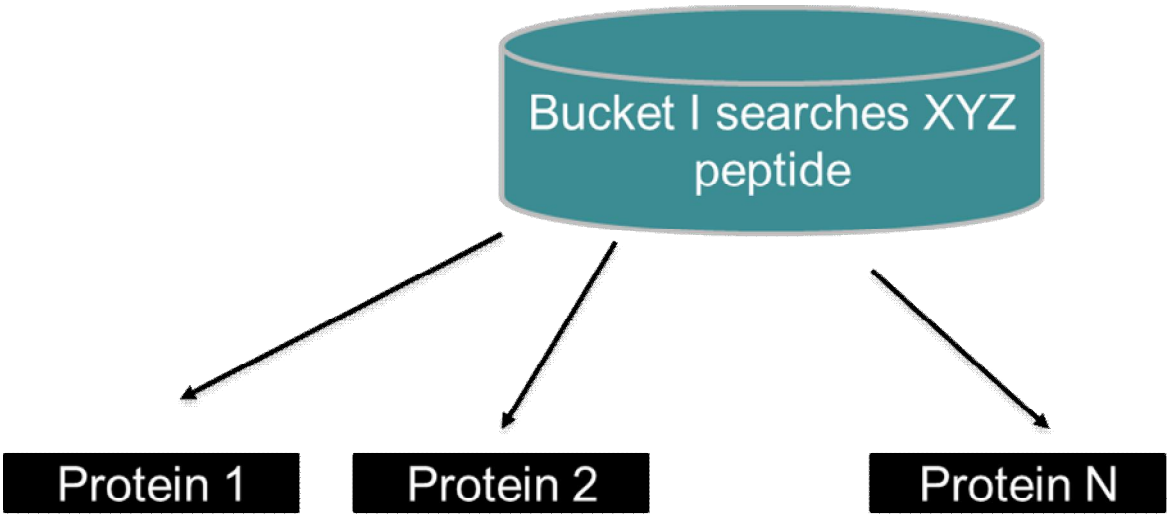
Ο ρόλος του Master Bucket έχει να επιτελέσει πολύ περισσότερα από τον ρόλο του Search Bucket. Εντός της υπολογιστικής μονάδας Master ακολουθείται ο σειριακός αλγόριθμος που περιγράφηκε νωρίτερα. Συγκεκριμένα για κάθε πρωτεΐνη εντός του Master εκτελούνται τα παρακάτω βήματα:

- Επιλογή πεπτιδίου
- Αναζήτηση αν το πεπτίδιο περιέχει κάποιο Core Unique που έχει ήδη εντοπιστεί για την πρωτεΐνη (σε αυτή την περίπτωση είναι unique αλλά όχι Core Unique). (εικόνα 8-3)
- Αποστολή πεπτιδίου προς αναζήτηση στους Search Buckets και αναμονή αποτελεσμάτων.
- Συγχρονισμός αναζήτησης των Search Buckets και παύση της αναζήτησής εντός αυτών όταν πρέπει.

Ο master παρακολουθεί την αναζήτηση των Search. Αν οποιοσδήποτε από αυτούς εντοπίσει το πεπτίδιο που αναζητά τότε στέλνει μήνυμα στον master ότι βρέθηκε, ο οποίος με την σειρά του ενημερώνει τους υπόλοιπους Search Buckets να σταματήσουν την αναζήτηση του τρέχοντος πεπτιδίου και να περιμένει για το επόμενο προς αναζήτηση. Η προσέγγιση αυτή επιτρέπει τον σύντομο τερματισμό της αναζήτησης για ένα πεπτίδιο, από την στιγμή που βρεθεί πως δεν είναι Core Unique. Φυσικά ο μόνος τρόπος για να γίνει αποδεκτό ένα πεπτίδιο ως Core Unique Peptide είναι η εξαντλητική αναζήτηση στο σύνολο των πρωτεϊνών. Μπορεί εύκολα κάποιος να υποστηρίξει πως ο χρόνος εύρεσης ενός CUP μειώνεται μόνο από τις βελτιστοποιήσεις που κάναμε εντός του single thread.

Ο σχεδιασμός του αλγορίθμου επιτρέπει την δημιουργία διαφορετικών ερωτήσεων στις οποίες ο Search Bucket μπορεί να απαντάει με ΝΑΙ / ΟΧΙ ή ακόμη και με πιο πολύπλοκες απαντήσεις όπως η αποστολή κατηγορηματικών δεδομένων ή ακόμη και αριθμητικών. Για να μπορέσουμε να απαντήσουμε στο ερώτημα αν ένα πεπτίδιο είναι Core Unique εντός μιας οικογένειας πρωτεϊνών, τότε αρκεί σταματήσουμε την αναζήτηση για το πεπτίδιο εντός του Master Bucket. Άρα αν δεν βρεθεί σε άλλο κουβά είναι μοναδικό σε αυτόν (τον Master Bucket) άρα και στην οικογένεια. Φυσικά

απαραίτητη προϋπόθεση γι' αυτό είναι ο χωρισμός του πρωτεώματος σε κουβάδες ανάλογα με την οικογένεια κάθε πρωτεΐνης.



Εικόνα 37: Αναζήτηση πεπτιδίου εντός του Search Bucket

8.4 Πλεονεκτήματα παράλληλης προσέγγισης

Η παραπάνω παράλληλη σχεδίαση μας έδωσε αρκετά πλεονεκτήματα:

- Κάθε κουβάς εκτελεί την αναζήτησή του εντός του δικού του νήματος εκτέλεσης απομονωμένος από τις υπόλοιπες διεργασίες, οπότε το σύστημα μπορεί να εκμεταλλευτεί πλήρως την προσθήκη περισσότερων λογικών πυρήνων.
- Μειώνεται κατά πολύ το πλήθος των πρωτεϊνών που πρέπει να εξεταστούν όταν το πεπτίδιο υπάρχει σε περισσότερες από μια.
- Μπορεί να επεκταθεί και να εκτελεστεί σε δυνατότερους επεξεργαστές χωρίς καμία αλλαγή.
- Το σχετικά μικρό υπολογιστικό κόστος που έχει η ερώτηση που υλοποιείται μέσα στους κουβάδες αναζήτησης δίνει συγκριτικό πλεονέκτημα σε αρχιτεκτονικές με απλούστερα σετ εντολών (instruction set) και χαμηλότερων συχνοτήτων αλλά με περισσότερους λογικούς επεξεργαστές (GPU's , KNL επεξεργαστές της intel).

8.5 Μειονεκτήματα παράλληλης προσέγγισης

Δυστυχώς η εισαγωγή παραλληλίας στο σύστημα, έχει και μειονεκτήματα:

- Η δημιουργία των νημάτων μπορεί να εισάγει μεγάλες καθυστερήσεις στο σύστημα, ειδικά όταν το επιλεγμένο πλήθος υποομάδων είναι κατά πολύ μεγαλύτερο από τους διαθέσιμους λογικούς πυρήνες.
- Για μικρά πρωτεώματα ένας κακός χωρισμός μπορεί να οδηγήσει σε πολύ χειρότερους χρόνους εκτέλεσης από τον σειριακό αλγόριθμο.
- Τα μεγέθη των κουβάδων πρέπει να είναι αναλογικά, μια καλή εμπειρική παρατήρηση είναι κάθε κουβάς να μην έχει λιγότερο από το 1/10 των προς αναζήτηση πρωτεϊνών.

- Για ερωτήσεις όπως η εύρεση CUP ανά οικογένεια, χάνεται κάποιο μέρος των πλεονεκτημάτων, αφού οι κουβάδες δεν μπορούν εγγυημένα να έχουν αναλογική διανομή.

8.6 Απόδοση Παράλληλου αλγορίθμου

Για την εκτίμηση της αθροιστικής απόδοσης των βελτιστοποιήσεων τόσο στο Single Thread καθώς και με την χρήση του παράλληλου αλγορίθμου, χρησιμοποιήσαμε 3 πρωτεώματα διαφορετικού μεγέθους:

- Μικρό (Wheat Uniprot 4/2017) με 371 πρωτεΐνες.
- Μεσαίο (E-Coli Uniprot 4/2017) με 4.436 πρωτεΐνες.
- Μεγάλο (Human Uniprot 4/2017) με 20.394 πρωτεΐνες.

Όλα τα παραδείγματα εκτελέστηκαν στον ίδιο υπολογιστή με την παρακάτω σύνθεση ο οποίος είναι και το μηχάνημα αναφοράς για την υπόλοιπη εργασία:

- CPU intel i7 7700 4 Cores / 8 Threads
- 32 GB RAM DDR4
- SSD για την αποθήκευση των δεδομένων.

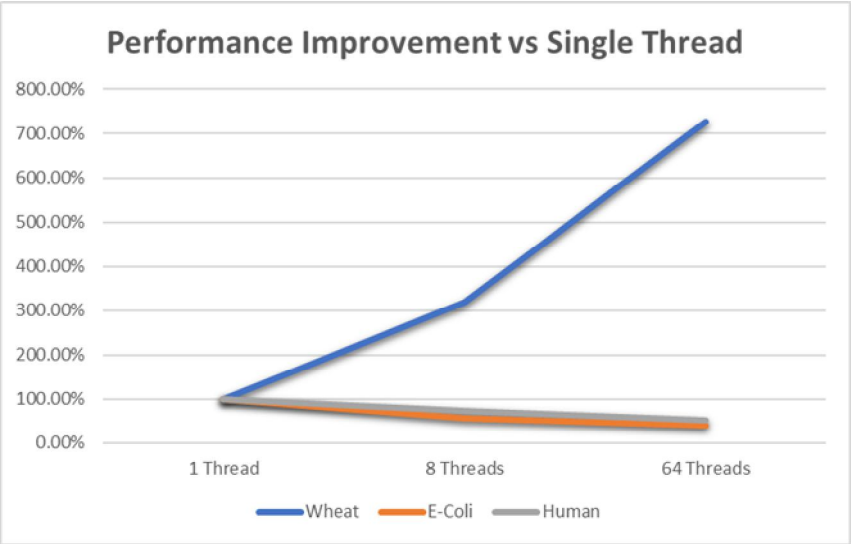
Όπως ήταν αναμενόμενο για το μικρό πρωτέωμα είχαμε μεγάλη πτώση στην απόδοση όσο ανέβαινε το πλήθος των threads που κάναμε διαθέσιμα στο σύστημα. Επίσης φαίνεται πως υπάρχει περιθώριο βελτίωσης του αλγορίθμου αφού στο μεσαίο πρωτέωμα το ποσοστό αύξησης μεταξύ των 8 και 64 threads είναι πολύ μικρότερο από το 50 % αυτού από το 1 Thread στα 8.

Execution Times (in Secs)	1 Thread	8 Threads	64 Threads
Wheat 4/2017	55.00	175.00	400.00
E-coli 4/2017	4,850.00	2,800.00	1,900.00
Human 4/2017	172,800.00	126,000.00	88,000.00

Εικόνα 38: Χρόνος κατασκευής Uniquome (sec)

Execution Times (in Secs)	1 Thread	8 Threads	64 Threads
Wheat 4/2017	100.00%	318.18%	727.27%
E-coli 4/2017	100.00%	57.73%	39.18%
Human 4/2017	100.00%	72.92%	50.93%

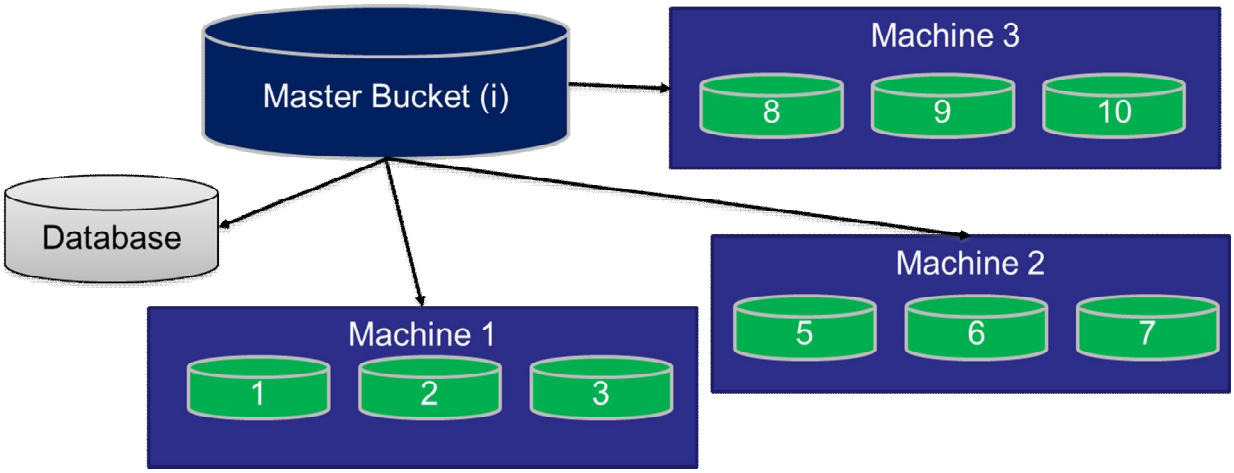
Εικόνα 39: Ποσοστιαία βελτίωση απόδοσης της κατασκευής του Uniquome



Γράφημα 2: Απόδοση Αλγορίθμου σε σχέση με την εκτέλεση σε 1 thread.

9. ΚΑΤΑΜΕΡΙΣΜΕΝΗ ΕΠΕΞΕΡΓΑΣΙΑ ΣΕ ΠΟΛΛΑΠΛΑ ΣΥΣΤΗΜΑΤΑ

Η παράλληλη υλοποίηση του αλγορίθμου επιτρέπει την περαιτέρω βελτίωση της απόδοσης μέσω καταμερισμού της αναζήτησης σε περισσότερα τους ενός αυτόνομων συστημάτων. Όπως φαίνεται στην εικόνα μπορούμε να χωρίσουμε τους κουβάδες του πρωτεώματος από το πρώτο βήμα του αλγορίθμου σε περισσότερα συστήματα. Η λογική παραμένει η ίδια όπως και στην παράλληλη έκδοση του αλγορίθμου. Μόνο ένας κουβάς μπορεί να έχει τον ρόλο του master στην εκτέλεση. Το πρώτο στάδιο της αναζήτησης αν το πεπτίδιο περιέχει ήδη κάποιο Core Unique παραμένει επίσης το ίδιο. Όμως τώρα ο master στέλνει το προς αναζήτηση πεπτίδιο στους υπόλοιπους υπολογιστές που έχουν οριστεί. Εντός αυτών εκτελείται πάλι η ίδια παράλληλη διαδικασία με την αποστολή της απάντησης στον master. Αυτή η επικοινωνία μπορεί να επιτευχθεί μέσω TCP/IP.



Εικόνα 40: Καταμερισμός αναζήτησης σε περισσότερα συστήματα

Το μηχάνημα που εκκινεί την κατασκευή είναι υπεύθυνο για τον συντονισμό της διαδικασίας. Κατά την προεργασία και αφού χωριστούν οι πρωτεΐνες σε κουβάδες, στέλνει σε κάθε δευτερεύων μηχάνημα ένα υποσύνολο των κουβάδων. Το σύστημα γνωρίζει την δυναμική κάθε επιμέρους υποσυστήματος οπότε ο χωρισμός του συνόλου των κουβάδων μπορεί να γίνει με βάση αυτή την γνώση, οπότε να στείλει στα λιγότερο δυνατά μηχανήματα (ή σε μηχανήματα που δεν πρέπει να κάνει χρήση του 100 % των πόρων) ένα μικρότερο υποσύνολο. Παράλληλα το ίδιο μηχάνημα έχει ένα πλήρες αντίγραφο όλων των κουβάδων του συστήματος ώστε να μπορεί να φέρνει τον καθένα στον ρόλο master. Το ίδιο δεν είναι απαραίτητο να κρατήσει κάποιο υποσύνολο από search buckets. Όμως πρέπει να διατηρεί μια σωστή εικόνα της λειτουργίας της διαδικασίας. Αν κάποιος κόμβος βγει εκτός συστήματος τότε μπορεί να υποκαταστήσει την λειτουργία του, χρησιμοποιώντας τον εαυτό του και ως ρόλο Search για τους κουβάδες που περιείχε ο κόμβος που βγήκε εκτός λειτουργίας.

Όπως γίνεται εύκολα κατανοητό η επικοινωνία συστημάτων εισάγει μια σημαντική καθυστέρηση στην διαδικασία. Επομένως για να είναι εποικοδομητική αυτή η διαχείριση θα πρέπει να υπόλοιπα συστήματα να έχουν αρκετούς πόρους (threads) διαθέσιμους για την λειτουργία. Αν δεν ισχύει αυτό η κατάτμηση της αναζήτησης σε περισσότερα μηχανήματα μπορεί να προκαλέσει μεγαλύτερη καθυστέρηση .

9.1 Πλεονεκτήματα - Μειονεκτήματα

- Μπορεί να ενσωματώσει συστήματα εντός του εργαστηρίου που δεν εκτελούν βαριές υπολογιστικές διεργασίες.
- Μειώνει κατά πολύ τον χρόνο κατασκευής ενός uniquome, ειδικά για μεγάλα πρωτεώματα όπως αυτό του ανθρώπου.
- Αν ο χρήστης επιλέξει να κρατήσει χαμηλά των αριθμό των διαθέσιμων threads ανά μηχάνημα, μπορεί να μειώσει τον φόρτο, επιτρέποντας την χρήση του μηχανήματος για άλλες εργασίες.
- Αν για κάποιο λόγο η κατασκευή του uniquome σταματήσει, μπορεί να επανεκκινήσει σε επόμενο χρόνο αφού ο master bucket, που είναι υπεύθυνος για τον σωστό συγχρονισμό όπως και στην αμιγώς παράλληλη έκδοση, διατηρεί την πληροφορία για το ποια buckets έχουν περάσει από τον ρόλο Master και ποιες πρωτεΐνες εντός του τελευταίου bucket είχαν αναλυθεί προτού διακοπεί η διαδικασία.

Το σημαντικότερο μειονέκτημα της κατανεμημένης προσέγγισης είναι η σχετικά δυσκολότερη παραμετροποίηση και εγκατάσταση ενός τέτοιου συστήματος από μη εξοικειωμένους χρήστες.

10. ΜΕΤΑΝΑΛΥΣΗ

Το παραγόμενο uniquome ενός οργανισμού είναι το πιο βαρύ υπολογιστικά κομμάτι της τρέχουσας εργασίας. Από την στιγμή της δημιουργίας του μπορεί να χρησιμοποιηθεί σε συνδυασμό με το αρχείο πρωτεώματος από το οποίο παρήχθη για να μας δώσει αρκετές χρήσιμες πληροφορίες. Όλες αυτές τις θεωρούμε μέρος της μετανάλυσης του uniquome. Παρακάτω αναλύονται μερικές μόνο από τις μεταναλύσεις που εκτελέσαμε ενώ η εφαρμογή μπορεί εύκολα να επεκταθεί και να αναλύσει ότι επιθυμεί ο κάθε ερευνητής σχετικά με το Uniquome.

10.1 Κατασκευή Composite Unique Peptides

Για την κατασκευή των CoUP η εφαρμογή ταξινομεί το uniquome κάθε πρωτεΐνης. Με βάση τον αλγόριθμο που παρουσιάστηκε στην ενότητα 3 ενώνει τα υπερκαλύπτομενα ή εφάπτοντα CUP ώστε να συνθέσει το αντίστοιχο Composite. Ο αλγόριθμος παραγωγής μπορεί να δεχτεί ως είσοδο την μέγιστη επιθυμητή απόσταση που μπορούν να έχουν δυο CUP ώστε να θεωρηθούν μέρος του ίδιου Composite. Η προεπιλογή είναι το 0 άρα θα πρέπει είτε να αλληλοεπικαλύπτονται είτε να εφάπτονται. Μετά την κατασκευή του Composite Unique Peptide, ο αλγόριθμος αποθηκεύει την θέση του εντός της πρωτεΐνης καθώς και τα CUP από τα οποία αποτελείται και τις θέσεις αυτών.

...	T	T	S	A	V	T	V	K	S	A	I	...
	T	T	S	A	V	T						
			S	A	V	T	V	K				
				A	V	T	V	K	S			
					V	T	V	K	S	A		
						T	V	K	S	A	I	

Εικόνα 41: Κατασκευή Composite Unique Peptide από τα αντίστοιχα CUP.

10.2 Εύρεση σχετικής θέσης εμφάνισης Core Unique Peptide

Μια από τις σημαντικές πληροφορίες που εξάγουμε από το uniquome είναι η σχετική θέση εμφάνισης όλων των Core Unique peptides εντός της πρωτεΐνης και ξεχωριστά την θέση εμφάνισης του πρώτου. Για τον υπολογισμό αυτού ο αλγόριθμος παίρνει ως είσοδο το uniquome και το αντίστοιχο πρωτέωμα. Στην συνέχεια ταξινομεί τα CUP κάθε πρωτεΐνης με βάση την θέση εμφάνισής τους. Υπολογίζει την σχετική θέση εμφάνισης για το πρώτο Core Unique αποθηκεύοντας αυτή την πληροφορία ξεχωριστά, ενώ στην συνέχεια εκτελεί τον ίδιο υπολογισμό για κάθε CUP. Η σχετική θέση δίνεται από τον παρακάτω τύπο W = P / L όπου W είναι η σχετική θέση, P η θέση του CUP και L το μέγεθος της ακολουθίας της πρωτεΐνης.

10.3 Στατιστικά πρωτεϊνών

Για κάθε πρωτεΐνη υπολογίζονται τα παρακάτω:

- Πλήθος CUP.
- Ποσοστό κάλυψης της Πρωτεΐνης από CUP. Αυτό υπολογίζεται ως το πλήθος των μη αλληλοεπικαλυπτόμενων αμινοξέων όλων των CUP προς το συνολικό μήκος της πρωτεΐνης.
- Πλήθος Composite Unique Peptide.
- Ποσοστό κάλυψης της Πρωτείνης από Composite Unique Peptides. Αυτό υπολογίζεται ως το πλήθος των μη αλληλοεπικαλυπτόμενων αμινοξέων όλων των Composite Unique peptides προς το συνολικό μήκος της πρωτεΐνης.

10.4 Κοινά CUP μεταξύ οργανισμών

Θεωρούμε αρκετά ενδιαφέρον το να μελετήσουμε τα κοινά CUP μεταξύ οργανισμών ως προς την αλληλουχία τους. Για να γίνει αυτό δίνουμε ως είσοδο τα uniquomes των οργανισμών καθώς και τα αντίστοιχα πρωτεώματα. Ο αλγόριθμος εντοπίζει το μικρότερο uniquome και αναζητά κάθε CUP αυτού μέσα στα Uniquomes των υπόλοιπων οργανισμών. Αν βρεθεί σε όλους τότε θεωρείτε κοινό.

Είναι προφανές πως όλες οι παραπάνω αναλύσεις μπορούν να κάνουν πολύ καλή χρήση της παράλληλης και κατανεμημένης δομής του αλγορίθμου. Για παράδειγμα η εύρεση των κοινών CUP μεταξύ οργανισμών μπορεί να μελετάει κάθε πρωτεΐνη ή ομάδα πρωτεϊνών σε διαφορετικό νήμα ανεβάζοντας κατά πολύ την απόδοση της ανάλυσης.

Οι αναλύσεις που περιγράφησαν παραπάνω δεν είναι οι μόνες που μπορούν να γίνουν στο Uniquome. Η εφαρμογή έχει σχεδιαστεί με τέτοιο τρόπο ώστε να μπορεί κάποιος προγραμματιστής να επεκτείνει τις αναλύσεις της, ενσωματώνοντας νέες, αρκεί να βασίζεται στα αρχεία του uniquome και το Fasta του πρωτεώματος και να συσχετίζει σωστά το ένα με το άλλο.

11. ΣΤΟΙΧΕΙΑ ΥΛΟΠΟΙΗΣΗΣ

Όπως αναφέρθηκε νωρίτερα σκοπός της εργασίας είναι η παραγωγή ενός λογισμικού με τα παρακάτω χαρακτηριστικά:

- Ορθή υλοποίηση του παραπάνω παράλληλου αλγορίθμου.
- Βέλτιστη χρήση του υποκείμενου υλικού/λογισμικού.
- Εκτέλεση σε πολλούς επεξεργαστές.
- Μόνιμη αποθήκευση των παραγόμενων Uniquomes.
- Δυνατότητα μετα-ανάλυσης των uniquomes – Δημιουργία Composite Unique Peptides
- Δυνατότητα εκτέλεσης σε Windows, Linux και ιδανικά MacOS.
- Δυνατότητα εκτέλεσης σε «Οικιακά» συστήματα.

11.1 Περιβάλλον ανάπτυξης

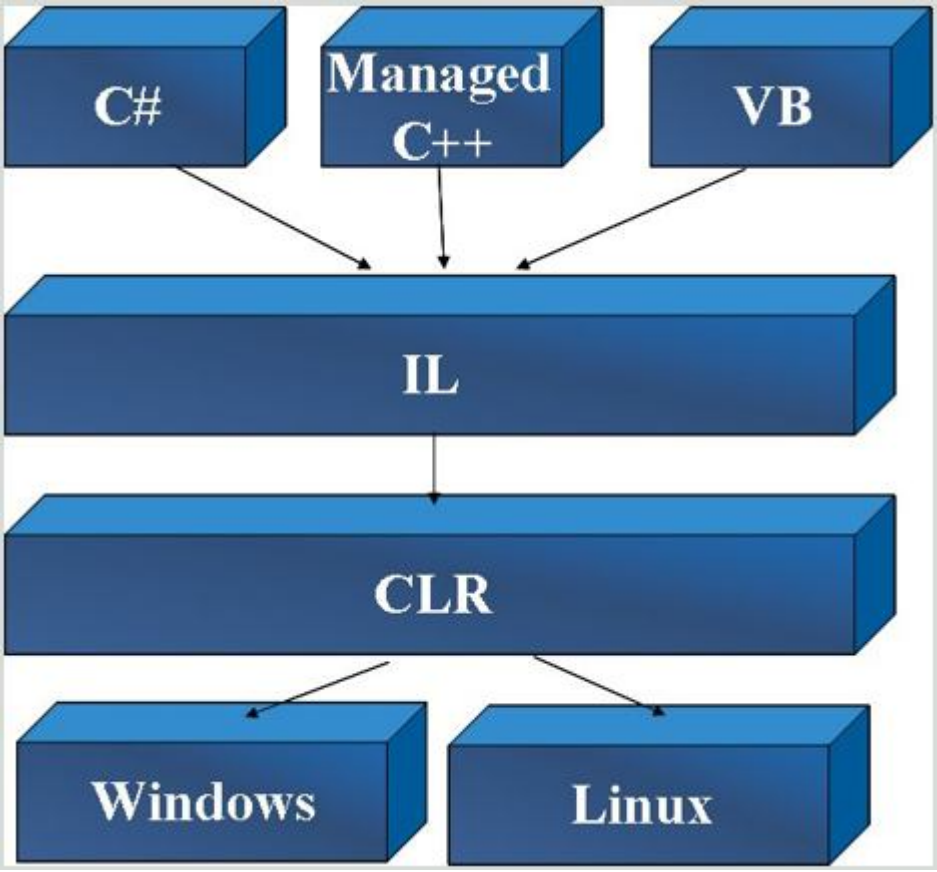
Για την υλοποίηση επιλέχθηκε το περιβάλλον .NET Core της Microsoft με χρήση της γλώσσα C#. Το .NET Core είναι η νεότερη προσθήκη εργαλείων του οικοσυστήματος του .NET που εμφανίστηκε στα μέσα του 2016 ως έκδοση 1.0. Την ώρα της συγγραφής της εργασίας βρισκόταν ήδη στην έκδοση 2.1 και όλο το λογισμικό έχει μεταφερθεί σε αυτή.

Το .NET Core σχεδιάστηκε από την Microsoft με σκοπό να παράξει λογισμικό που μπορεί να εκτελεστεί από τα 3 μεγάλα υπάρχοντα λογισμικά, χωρίς να είναι απαραίτητο να ξαναγραφτεί κάποιο κομμάτι του κώδικα. Αρκεί η μεταγλώττιση του προγράμματος στο επιθυμητό λειτουργικό, και φυσικά η εγκατάσταση του .NET στο μηχάνημα όπου θα εκτελεστεί η εφαρμογή. Η χρήση του εμπίπτει κάτω από διαφορετικές άδειες χρήσης, με το σύνολο αυτών να επιτρέπουν την δωρεάν χρήση του λογισμικού. Στον παρακάτω πίνακα εμφανίζονται οι άδειες για τα επιμέρους τμήματα του οικοσυστήματος του .NET.

Πίνακας 5: Άδειες χρήσης λογισμικού

Component	License
NET Framework (redistributable package)	Proprietary software[80]
Reference source code of .NET Framework 4.5 and earlier	Microsoft Reference License (Ms-RSL[a])[7][81]
Reference source code of .NET Framework 4.6	MIT License[82]
Mono	MIT License[83]
.NET Core, CoreFX, CoreCLR and CLI	MIT License[84]
.NET Micro Framework	Apache License 2.0[85]
.NET Compiler Platform (codename "Roslyn")	Apache License 2.0[86]
ASP.NET MVC, Web API and Web Pages (Razor)	Apache License 2.0[87]
ASP.NET Core	Apache License 2.0[88]
ASP.NET Ajax Control Toolkit	BSD License[89]
ASP.NET SignalR	Apache License 2.0[90]
Entity Framework	Apache License 2.0[91]
NuGet	Apache License 2.0[92]

Η ανάπτυξη της εφαρμογής έγινε εξ' ολοκλήρου στην ακαδημαϊκή έκδοση του Visual Studio 2017.



Εικόνα 42: Οικοσύστημα .NET Core Framework

Η εφαρμογή έχει δοκιμαστεί με επιτυχία σε Windows, MacOs, Ubuntu Linux και Raspian Linux το οποίο εκτελείται σε SoC Raspberry Pi.

11.2 Αποθήκευση Δεδομένων

Μια σημαντική δυνατότητα που θέλαμε να προσφέρει η εφαρμογή είναι η μόνιμη αποθήκευση των Uniquomes. Όπως φάνηκε και από τα παραπάνω η δημιουργία ενός uniquome είναι μια ακριβή υπολογιστικά διεργασία, η οποία δεν έχει κανένα λόγο να επαναλαμβάνεται κάθε φορά που ο/η ερευνητής/τρια επιθυμεί να αναζητήσει στοιχεία εντός αυτού. Απαιτείται ο επανυπολογισμός αυτού (Uniquome) μόνο όταν αλλάξει το πρωτέωμα. Επομένως είναι ιδιαίτερα χρήσιμη η μόνιμη αποθήκευση των uniquomes. Για την επίτευξη αυτού επιλέχθηκε η χρήση αρχείων κειμένου σε μορφή CSV. Οι δυο προηγούμενες εφαρμογές χρησιμοποιούσαν συστήματα σχεσιακών βάσεων δεδομένων (RDBMS). Κρίθηκε πως τουλάχιστον για την κατασκευή του uniquome η χρήση ενός τέτοιου συστήματος δεν συμφέρει. Ο λόγος είναι πως τα συστήματα αυτά κατά την εισαγωγή δεδομένων εκτελούν πολλούς ελέγχους ακεραιότητας των δεδομένων κάτι το οποίο εισάγει καθυστέρηση στην κατασκευή του uniquome χωρίς να δίνει κανένα όφελος σε αυτό το σημείο. Επιπρόσθετα η σύνδεση της κατασκευής με ένα σχεσιακό σύστημα προϋποθέτει πως αυτό θα είναι εγκατεστημένο σε στο μηχάνημα που θα εκτελεστεί η εφαρμογή. Αντιθέτως η αποθήκευση των δεδομένων σε αρχείο κειμένου (στο οποίο γίνεται πραγματική εγγραφή ανά μερικές χιλιάδες CUP ανεβάζοντας ακόμη περισσότερο την απόδοση) είχε πολλά συγκριτικά πλεονεκτήματα.

- Είναι αρκετά πιο γρήγορη από την αποθήκευση σε βάση δεδομένων.
- Δεν χρειάζεται κάποιο ιδιαίτερο εργαλείο για να μπορέσει κάποιος να διαβάσει τα δεδομένα.
- Δεν χρειάζεται ιδιαίτερες γνώσεις από τους ερευνητές για την αναζήτηση εντός αυτού (π.χ. γλώσσα ερωτημάτων SQL) .
- Μπορεί εύκολα να μετατραπεί σε άλλες μορφές αρχείων π.χ. FASTA / JSON.
- Μπορεί εύκολα να εισαχθεί σε άλλα συστήματα όπως μια βάση δεδομένων (σχεσιακή ή μη) για περαιτέρω ανύλυση.

Η αρχική κατασκευή κάθε uniquome οδηγεί τουλάχιστον σε ένα αρχείο με την δομή:

- Κωδικός Πρωτείνης
- Ακολουθία Πεπτίδιου

Αν ο χρήστης το επιλέξει η εφαρμογή μπορεί να αποθηκεύσει και μια επεκταμένη μορφή του αρχείου με τα παρακάτω πεδία

- Κωδικός Πρωτείνης
- Μέγεθος Πρωτείνης
- Πρώτη θέση εμφάνισης του Πεπττιδίου
- Ακολουθία Πεπτιδίου
- Όλες οι θέσεις εμφάνισης του πεπτιδίου (συμπεριλαμβανομένης και της πρώτης)

Με βάση τα 2 παραπάνω αρχεία και το αρχείο Fasta του πρωτεώματος ο ερευνητής μπορεί να βρει πλήθος αναλύσεων ή ακόμη και να γράψει scripts σε μια γλώσσα προτίμησης αναζητώντας βαθύτερη πληροφορία.

11.3 Τεχνικά Χαρακτηριστικά Υλοποίησης

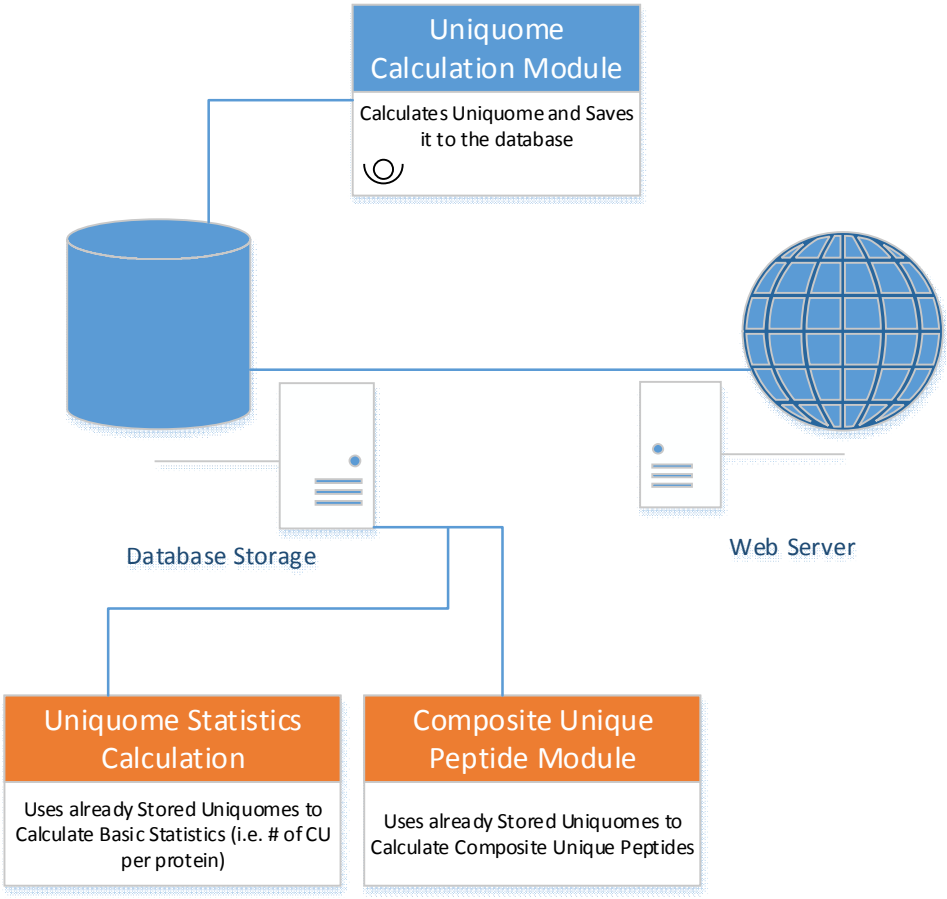
Στις βασικές προδιαγραφές του συστήματος είναι η ικανότητα επέκτασης τόσο της απόδοσης του υπολογισμού του uniquome (scalability) , αλλά και η προσθήκης νέων χαρακτηριστικών όπως για παράδειγμα η εύρεση των Composite Unique Peptides, η εκτέλεση προκαθορισμένα ερωτήματα και άλλα. Αποφασίσαμε η εφαρμογή να είναι αρθρωτή (modular), ώστε κάθε μέρος της να μπορεί να αντικατασταθεί ή να επεκταθεί μελλοντικά. Η σύνδεση των επιμέρους τμημάτων γίνεται με την χρήση διεπαφών (interfaces) για τις in-process διαδικασίες, ενώ εξωτερικά διασυνδεδεμένα υποσυστήματα αυτής επικοινωνούν με χρήση JSON αρχείων.

Η εφαρμογή μπορεί να δεχτεί επεκτάσεις οι οποίες πρέπει να υλοποιούν συγκεκριμένα interfaces. Κατά την έναρξή της ανιχνεύει τα αρχεία του φακέλου που είναι εγκατεστημένη (καθώς και αρχεία σε άλλους φακέλους που ορίζονται στην παραμετροποίησή της) και αν εντοπίσει αρχεία δυναμικών βιβλιοθηκών (dll) που υλοποιούν τα επιθυμητά interfaces φορτώνει επιπρόσθετες λειτουργίες που αυτά υλοποιούν.

Η προηγούμενη δυνατότητα γίνεται εφικτή μέσω της τεχνολογίας Reflection η οποία δίνει ε ένα πρόγραμμα την δυνατότητα να εξετάσει και να μεταβάλει την δομή καθώς και

την συμπεριφορά του κατά την διάρκεια της εκτέλεσής του. Επιτρέπει την δυναμική ανίχνευση και καταχώρηση νέων τύπων δεδομένων καθώς και λειτουργιών. Το .NET framework υλοποιεί πλήρως την παραπάνω τεχνολογία. Ο χρήστης λοιπόν σε κάθε νέα εκκίνηση της εφαρμογής μπορεί χωρίς να προβεί σε καμία ενέργεια πέραν της αντιγραφής των αρχείων στο φάκελο να έχει στην διάθεσή του τις νέες λειτουργικότητες.

Αυτή την στιγμή η εφαρμογή διαθέτει διεπαφή μόνο μέσω τερματικής κονσόλας, αφού προς το παρών το .NET Core 2 δεν υποστηρίζει την δημιουργία διεπαφών για διαφορετικά λειτουργικά κάτι που ίσως τελικά γίνει διαθέσιμο στην έκδοση 3. Στον σχεδιασμό της ομάδας είναι η μελλοντική ανάπτυξη Web εφαρμογής για την διαχείριση της εφαρμογής η αρχιτεκτονική της οποίας φαίνεται στην εικόνα 43.



Εικόνα 43: Αρχιτεκτονική Εφαρμογής

Κατά την πρώτη εκτέλεση της εφαρμογής ο χρήστης υποχρεώνεται να συμπληρώσει τα στοιχεία της παραμετροποίησης του υπολογιστή στον οποίο εκτελεί την εφαρμογή, όπως φαίνεται στην παρακάτω εικόνα 44.

```
λ dotnet ProteinAnalysis.ConsoleClient.dll
Please Provide Necessary Initialization Information
Client Name (Default [Machine Name]):
TITAN
Data Location (Default [Data]):

Data Catalog (Default [Catalog]):

Select one of the available IPs or 0 for custom.
1: 192.168.1.102
2: 172.27.220.33
192.168.1.102
Select one of the available IPs or 0 for custom.
1: 192.168.1.102
2: 172.27.220.33
0
Fill in the IP
192.168.1.115
Please specify Port or 0 to use default  [1980] (Firewall must allow inbound / outbound):
1980
Select Available System Memory:
32
Select Max Memory to Use:
16
Saving file
Configuration Loaded For Client: TITAN
Finished !
```

Εικόνα 44: Παραμετροποίηση εφαρμογής

Όπως βλέπουμε ο χρήστης καλείται να δώσει το όνομα του υπολογιστή (έχει πρακτική σημασία μόνο στην κατανεμημένη επεξεργασία), τους φακέλους στους οποίους θα αποθηκεύει τους Καταλόγους (το σχήμα που διατηρεί την πληροφορία οργανισμών/ πρωτεωμάτων) υποκαθιστώντας εν μέρει την βάση δεδομένων (δεν είναι υποχρεωτική η χρήση του). Επίσης συμπληρώνει το μέγιστο ποσοστό μνήμης που επιτρέπεται να δεσμεύσει καθώς και την IP και την πόρτα που ακούει (επίσης απαραίτητα μόνο στην κατανεμημένη λειτουργία). Η παραμετροποίηση αυτή αποθηκεύεται με μορφή JSON όπως φαίνεται στην παρακάτω εικόνα

```
{
  "ClientName": "TITAN",
  "DataLocation": "Data",
  "Catalog": "Catalog.json",
  "OperatingSystem": 100,
  "SystemMemory": 32,
  "MemoryToUse": 16,
  "NoOfVirtualProcessors": 12,
  "MaxVirtualProcessorsToUse": 12,
  "IpAddress": "192.168.1.115",
  "Port": 1980
}
```

Εικόνα 45: Αρχείο Παραμετροποίησης εφαρμογής σε JSON.

Όπως φαίνεται πέραν των στοιχείων που ζήτησε η εφαρμογή έχει βρει και αποθηκεύσει το πλήθος των λογικών πυρήνων (threads) που έχει στην διάθεσή της. Κάθε μηχάνημα που εκτελεί την εφαρμογή έχει ένα αντίστοιχο αρχείο. Το μόνο που απαιτείται από τον χρήστη για να συνδέσει όλα τα μηχανήματα σε ένα μια κοινή συστοιχία και να μπορεί να εκμεταλλευτεί την κατανεμημένη μορφή του αλγορίθμου είναι να αντιγράψει όλα αυτά τα αρχεία στο φάκελο της εφαρμογής κάθε μηχανήματος. Η εφαρμογή μόλις ξεκινήσει

θα διαβάζει όλα τα αρχεία παραμετροποίησης και θα γνωρίζει το οικοσύστημα στο οποίο μπορεί να εκτελέσει κατανεμημένα τις λειτουργίες της. Φυσικά για να μπορούν τα μηχανήματα να είναι μέλη στην συστοιχία θα πρέπει να εκτελούν την εφαρμογή σε λειτουργία server.

Ο τρόπος με τον οποίο κατορθώνει να συγχρονίσει τους υπολογιστές μέλη της συστοιχίας είναι με την υλοποίηση ενός μικρού Web Server ο οποίος μπορεί να στείλει κλήσεις καθώς και να δεχτεί μέσω REST API μέσω του HTTP πρωτοκόλλου. Για να μπορέσουν να επικοινωνήσουν μεταξύ τους τα μηχανήματα, θα πρέπει το firewall καθενός να επιτρέπει Inbound / Outbound στην δοθείσα πόρτα.

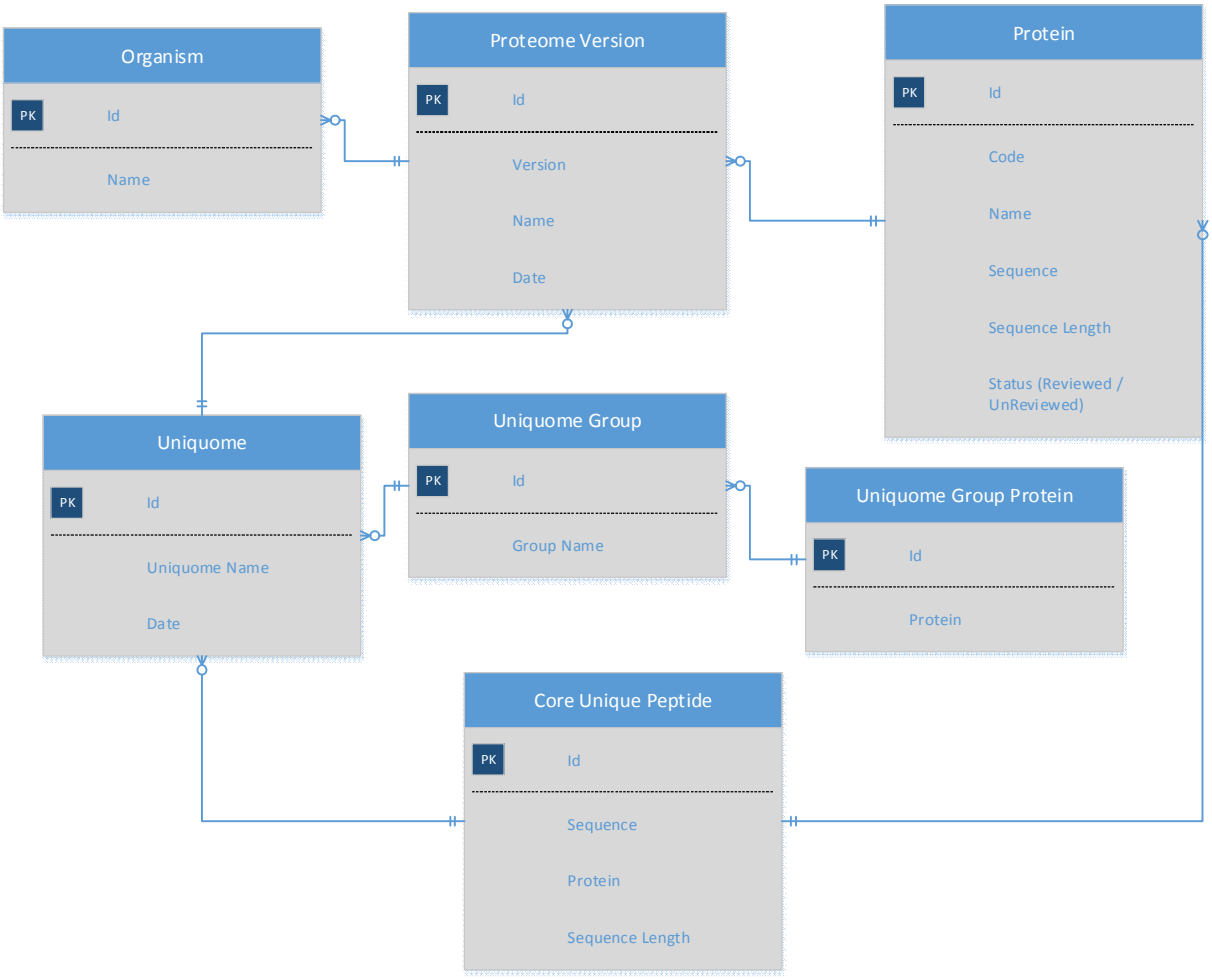
```
λ dotnet ProteinAnalysis.ConsoleClient.dll create -p e:\proteomes\uniprot-ecoli.fasta -u ecoli.uniquome -b 48 -i 4 -a 100
Configuration Loaded For Client: TITAN
4  100
Separating Proteome
Creating Uniquome...
Bucket 2beb4295-6290-4615-8565-57c70707771f
Q46901  502     385     2/11/2018 5:27:27 μμ    2/11/2018 5:27:27 μμ    22.837.576      1/4.436 0,02 %
P76491  199     148     2/11/2018 5:27:27 μμ    2/11/2018 5:27:27 μμ    23.272.248      2/4.436 0,05 %
P06134  354     246     2/11/2018 5:27:27 μμ    2/11/2018 5:27:28 μμ    24.207.712      3/4.436 0,07 %
P00350  468     352     2/11/2018 5:27:28 μμ    2/11/2018 5:27:28 μμ    23.892.488      4/4.436 0,09 %
P37685  512     352     2/11/2018 5:27:28 μμ    2/11/2018 5:27:28 μμ    25.053.752      5/4.436 0,11 %
P00963  330     239     2/11/2018 5:27:28 μμ    2/11/2018 5:27:28 μμ    24.025.752      6/4.436 0,14 %
P24242  336     241     2/11/2018 5:27:28 μμ    2/11/2018 5:27:28 μμ    24.023.464      7/4.436 0,16 %
P0ABB8  898     657     2/11/2018 5:27:28 μμ    2/11/2018 5:27:29 μμ    25.874.120      8/4.436 0,18 %
P12998  384     275     2/11/2018 5:27:29 μμ    2/11/2018 5:27:29 μμ    25.734.504      9/4.436 0,20 %
P06282  251     198     2/11/2018 5:27:29 μμ    2/11/2018 5:27:29 μμ    21.493.096      10/4.436    0,23 %
P0A6F9  97      68      2/11/2018 5:27:29 μμ    2/11/2018 5:27:29 μμ    24.901.704      11/4.436    0,25 %
P0AE95  129     90      2/11/2018 5:27:29 μμ    2/11/2018 5:27:29 μμ    22.147.960      12/4.436    0,27 %
P08401  474     337     2/11/2018 5:27:29 μμ    2/11/2018 5:27:30 μμ    23.677.552      13/4.436    0,29 %
P31459  292     213     2/11/2018 5:27:30 μμ    2/11/2018 5:27:30 μμ    24.341.720      14/4.436    0,32 %
P77381  475     331     2/11/2018 5:27:30 μμ    2/11/2018 5:27:30 μμ    22.028.768      15/4.436    0,34 %
P0A8J2  179     129     2/11/2018 5:27:30 μμ    2/11/2018 5:27:30 μμ    24.883.600      16/4.436    0,36 %
P75858  356     262     2/11/2018 5:27:30 μμ    2/11/2018 5:27:30 μμ    21.675.496      17/4.436    0,38 %
P0CB39  577     423     2/11/2018 5:27:30 μμ    2/11/2018 5:27:31 μμ    25.997.576      18/4.436    0,41 %
P25798  552     397     2/11/2018 5:27:31 μμ    2/11/2018 5:27:31 μμ    27.126.176      19/4.436    0,43 %
P39405  262     198     2/11/2018 5:27:31 μμ    2/11/2018 5:27:31 μμ    25.670.416      20/4.436    0,45 %
P09148  348     265     2/11/2018 5:27:31 μμ    2/11/2018 5:27:31 μμ    26.421.304      21/4.436    0,47 %
```

Εικόνα 46: Παράδειγμα εκτέλεσης κατασκευής Uniquome για τον οργανισμό E-coli

```
λ dotnet ProteinAnalysis.ConsoleClient.dll compositeUnique -p "e:\proteomes\Vaggelis Oct 2018\uniprot-rat-
06-10-2018.fasta" -u "e:\proteomes\Vaggelis Oct 2018\rat.uniquome" -f "e:\proteomes\Vaggelis Oct 2018\rat.
composite" -m 0 -t 24 -o -e0
Configuration Loaded For Client: TITAN
Calculating Composite Core Uniques
Will Create Extended Stats on Composite Core Uniques
Composite Unique Peptides were analyzed
Analysed Protein 334/334        Analysed Bucket:24 / 24Saved Stats 1 at e:\proteomes\Vaggelis Oct 2018\rat
.composite.Stats.Protein
Saved Stats 2 at e:\proteomes\Vaggelis Oct 2018\rat.composite.Stats.PeptidesByLength
Saved Stats 3 at e:\proteomes\Vaggelis Oct 2018\rat.composite.Stats.PeptidesByCupCount
Saved Stats 4 at e:\proteomes\Vaggelis Oct 2018\rat.composite.Stats.PeptidesByRelativePosition
Total CU Proteins 24. Started: 2/11/2018 5:29:35 μμ: Ended : 2/11/2018 5:29:50 μμ
Memory Stats:
Memory At Start:        644.120
Memory At End:  731.249.496
Finished !
```

Εικόνα 47: Παράδειγμα εκτέλεσης κατασκευής Composite Unique Peptides για το ποντίκι

Το σύνολο των διαθέσιμων πρωτεωμάτων ενημερώνεται διαρκώς είτε με την προσθήκη των πρωτεωμάτων νέων οργανισμών, είτε με την ενημέρωση των υπαρχόντων. Θέλουμε τα παραγόμενα Uniquomes να είναι όσο το δυνατόν πιο ευέλικτα γι' αυτό καταλήξαμε στην δημιουργία του παρακάτω σχήματος της εικόνας 11-2. Αυτό δεν αναφέρεται σε κάποιο σχεσιακό σχήμα, αλλά σε μια «χαλαρή» περιγραφή με χρήση JSON που επιτρέπει στο σύστημα να δέχεται καλύτερα δομημένη πληροφορία και η οποία μελλοντικά θα μπορούσε να χρησιμοποιηθεί για την εισαγωγή των στοιχείων σε μια NoSQL βάση δεδομένων.



Εικόνα 48: Σχήμα Εφαρμογής

Η εφαρμογή επιτρέπει τον ορισμό νέων οργανισμών. Ένα πρωτέωμα ανήκει σε κάποιον οργανισμό. Ως πρωτέωμα οι χρήστες μπορούν να ορίσουν οποιοδήποτε σύνολο πρωτεϊνών. Για παράδειγμα για τον οργανισμό «Homo Sapiens» το σύστημα μπορεί να έχει πολλαπλές εκδόσεις των πρωτεωμάτων όπως αυτά εκδίδονται από την Uniprot. Ο χρήστης όμως μπορεί επίσης να δημιουργήσει ένα custom πρωτέωμα, πχ από τις πρωτεΐνες που εκφράζονται σε συγκεκριμένο όργανο ή ασθένεια. Για κάθε πρωτέωμα που δημιουργείται μπορεί να οριστεί το σύνολο των πρωτεϊνών που περιέχονται είτε με την φόρτωση ενός FASTA αρχείου είτε με αντιγραφή και τροποποίηση ενός υπάρχοντος. Για την δημιουργία του Uniquome ο χρήστης πρέπει να επιλέξει το πρωτέωμα που επιθυμεί και να ορίσει τις επιμέρους ομάδες (Uniquome Group). Η προεπιλεγμένη λειτουργία της εφαρμογής είναι μια ομάδα ανά uniquome που μεταφράζεται ως αναζήτηση των Core Unique Peptides στο σύνολο του πρωτεώματος. Εναλλακτικά ο χρήστης μπορεί να ορίσει ομάδες πρωτεϊνών (πχ χωρισμός των πρωτεϊνών με βάση την λειτουργία τους) και να δημιουργήσει το uniquome ανά ομάδα. Σε αυτή την περίπτωση η μοναδικότητα αναζητείται μόνο διαμέσου των ομάδων. Αυτό σημαίνει ότι ένα Core Peptide για να είναι unique θα πρέπει να υπάρχει μόνο σε μία ή περισσότερες πρωτεΐνες της ίδιας ομάδας.

12. ΑΠΟΤΕΛΕΣΜΑΤΑ

12.1 Αναλύσεις Uniquome

Αναλύσαμε τα πρωτεώματα όλων των παρακάτω οργανισμών με την νέα εφαρμογή και επιβεβαιώσαμε τα αποτελέσματα με την προηγούμενη έκδοση της που αναφέρθηκε στην ενότητα 5.

Πίνακας 6:Core Unique Peptides ανά Οργανισμό

Organism	Unique Peptides
Homo Sapiens (Model Data)	8,479,357
Homo Sapiens	7,244,892
Mouse	6,385,830
Rat	2,717,612
Bovine	1,629,165
Gallus Gallus (Chicken)	758,266
Horse	62,036
E-coli	965,470
Maize Zea Mays	138,489
Rabbit	256,111
Ceanorabditis Elegants	1,443,318
Drosophila Melanogaster	1,482,643
Baker's Yeast	2,002,342
Garden Pea	74,420
Soybean	71,054
Pig	367,308
Common Tobacco	76,736
Wheat	49,427
Sheep	91,068
Zebra Fish	963,101
Combined Rat-Mouse	2,600,000

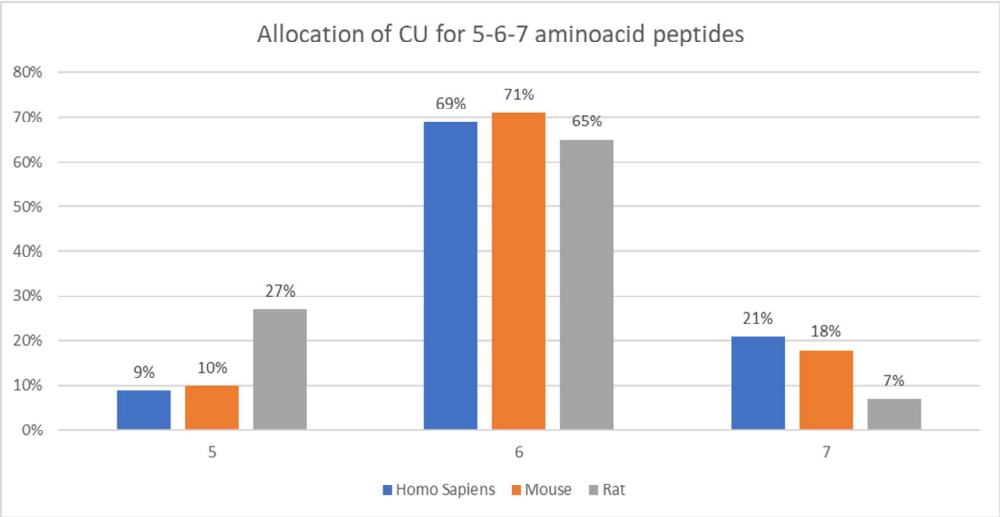
12.2 Ανάλυση Human – Rat – Mouse

Εκτιμούμε πως η ανάλυση των 3 οργανισμών Human / Rat και mouse και η αντιπαραβολή των uniquomes τους έχει ιδιαίτερη σημασία λόγω της καλής ανάλυσης που έχουν τα συγκεκριμένα πρωτεώματα, αλλά και γιατί χρησιμοποιούνται συχνά στα πειράματα που αφορούν φάρμακα ή άλλες εφαρμογές στον άνθρωπο.

Στον πίνακα 8 εμφανίζονται τα αποτελέσματα της ανάλυσης του uniquome για τους 3 αυτούς οργανισμούς, ενώ στο γράφημα 3 και τον πίνακα 7 εμφανίζεται η κατανομή των CUP ανά μέγεθος και στους 3 οργανισμούς. Παρατηρούμε σημαντική ομοιότητα στον άνθρωπο και το ποντίκι και μια σημαντική απόκλιση στα 5πεπτίδια και τα 7πεπτίδια για τον αρουραίο. Εκτιμούμε πως αυτό χρήζει περαιτέρω διερεύνησης .

Πίνακας 7:Ανάλυση Uniquome των 3 οργανισμών

	Human	Rat	Mouse
# of Proteins	20.198 (11.325.091 aminoacids)	7.985 (4.042.857 aminoacids)	16.859 (9.509.641 aminoacids)
Proteins with CU	20.083	7.978	16.836
Proteins without CU	115	7	23
Core Unique Peptides	7.244.892	2.717.612	6.385.830
Total Density	64%	67%	67%
Size of Core Unique Peptides	6-peptide (69%) 7-peptide (21%)	6-peptide (65%) 5-peptide (27%)	6-peptide (71%) 7-peptide (18%)



Γράφημα 3: Κατανομή CUP ανά μέγεθος πεπτιδίου και στους 3 οργανισμούς.

Πίνακας 8:Πίνακας κατανομής ανά μήκος πεπτιδίου και για τους 3 οργανισμούς.

	Human		Mouse		Rat	
Length	Count	%	Count	%	Count	%
4	792.00	< 1 %	1,188.00	< 1 %	4,174.00	< 1 %
5	628,442.00	9%	666,018.00	10%	743,662.00	27%
6	5,005,123.00	69%	4,504,826.00	71%	1,755,425.00	65%
7	1,485,276.00	21%	1,139,200.00	18%	179,889.00	7%
8	85,020.00	1%	54,393.00	1%	8,570.00	< 1 %
9	14,803.00	< 1 %	8,420.00	< 1 %	2,200.00	< 1 %
10	6,927.00	< 1 %	3,802.00	< 1 %	1,047.00	< 1 %

12.3 Σύγκριση κοινών CUP Human – Rat – Mouse

Στην συνέχεια θελήσαμε να μελετήσουμε τα κοινά σημεία μεταξύ των uniquomes των 3 οργανισμών. Από την ανάλυσή μας προέκυψε πως υπάρχουν 699.582 κοινά CUP μεταξύ των 3 αυτών οργανισμών. Στον πίνακα 9 μπορούμε να δούμε το πλήθος των κοινών CUP για τον συνδυασμό ανά 2 των οργανισμών καθώς και το ποσοστό ομοιότητας μεταξύ αυτών. Βλέπουμε μεγάλη ομοιότητα μεταξύ ανθρώπου και ποντικού κάτι που ίσως αναμέναμε με βάση την παρατήρηση που αναφέρθηκε στην πρώτη ενότητα όπου η διαφορά του γονιδιώματος των 2 είναι περίπου 5%.

Πίνακας 9:Κοινά CUP ανάμεσα στους οργανισμούς

Common Core Unique Peptides	RAT 4/2017	Mouse 4/2017	Human 4/2017
RAT 4/2017	2,717,612.00	1,321,301.00	1,020,208.00
Mouse 4/2017	-	6,385,830.00	2,722,506.00
Human 4/2017	-	-	7,244,892.00

Common Core Unique Peptides	RAT 4/2017	Mouse 4/2017	Human 4/2017
RAT 4/2017	100.00%	48.62%	37.54%
Mouse 4/2017	20.69%	100.00%	42.63%
Human 4/2017	14.08%	37.58%	100.00%

Στην συνέχεια συγκρίνοντας την σχετική θέση εμφάνισης των CUP στους οργανισμούς παρατηρούμε πως 502.994 ή το 71 % αυτών εμφανίζονται στην ίδια σχετική θέση (διαφορά σχετικής θέσης μικρότερη του 1%). Εκτιμούμε πως οφείλεται στην εξέλιξη των 3 οργανισμών από ένα κοινό πρόγονο.

Πίνακας 10: Ανάλυση σχετικής θέσης των κοινών CUP στους 3 οργανισμούς

Common Relative Position of Common Core Unique Peptides	RAT 4/2017	Mouse 4/2017	Human 4/2017
RAT 4/2017	2,717,612.00	1,084,215.00	668,726.00
Mouse 4/2017	-	6,385,830.00	2,058,034.00
Human 4/2017	-	-	7,244,892.00

Common Relative Position of Common Core Unique Peptides	RAT 4/2017	Mouse 4/2017	Human 4/2017
RAT 4/2017	100.00%	39.90%	24.61%
Mouse 4/2017	16.98%	100.00%	32.23%
Human 4/2017	9.23%	28.41%	100.00%

Τέλος μελετήσαμε τις διαφορές στην σχετική θέση εμφάνισης των CUP οργανισμών και βρήκαμε πως 53.776 ή το 7% αυτών έχουν μακρινή σχετική θέση εμφάνισης (διαφορά > 5%) Πίνακας 11.

Πίνακας 11:Διαφορές στην σχετική θέση εμφάνισης ων CUP μεταξύ των 3 οργανισμών

Different Relative Position of Common Core Unique Peptides	RAT 4/2017	Mouse 4/2017	Human 4/2017
RAT 4/2017	2,717,612.00	159,355.00	261,000.00
Mouse 4/2017	-	6,385,830.00	435,659.00
Human 4/2017	-	-	7,244,892.00

Different Relative Position of Common Core Unique Peptides	RAT 4/2017	Mouse 4/2017	Human 4/2017
RAT 4/2017	100.00%	5.86%	9.60%
Mouse 4/2017	2.50%	100.00%	6.82%
Human 4/2017	3.60%	6.01%	100.00%

13. ΕΠΟΜΕΝΑ ΒΗΜΑΤΑ – ΕΠΕΚΤΑΣΕΙΣ

13.1 Επιτάχυνση διαδικασίας μέσω χρήσης GPU (CUDA – OpenGL)

Οι σύγχρονες κάρτες γραφικών προσφέρουν την δυνατότητα επεξεργασίας σε πολλούς πυρήνες. Μια σχετικά απλή κάρτα μπορεί να έχει 300 επεξεργαστές (π.χ. σειρά 1050 της Nvidia με κόστος < 200 € έχει 684 επεξεργαστές με 4 GB DDR5 RAM). Οι επεξεργαστές αυτοί δεν έχουν την δυνατότητα να εκτελέσουν όλες τις λειτουργίες ενός επεξεργαστή γενικής χρήσης (CPU). Η αναζήτηση κειμένου όμως είναι μια σχετικά απλή διαδικασία και πιστεύουμε πως μπορεί να υλοποιηθεί στο προγραμματιστικό περιβάλλον των καρτών γραφικών. Η εφαρμογή θα μπορούσε να επεκταθεί ώστε να κάνει χρήση της αρχιτεκτονικής των καρτών γραφικών και της ταχύτητας αναζήτησης που προσφέρουν.

13.2 Χρήση εργαλείων αναζήτησης κειμένων

Η αναζήτηση κειμένου είναι μια από τις πιο διαδεδομένες διαδικασίες που καλούνται να επιτελέσουν οι σύγχρονοι υπολογιστές. Η ανάγκη για γρήγορη αναζήτηση στο διαδίκτυο έχει οδηγήσει στην δημιουργία ιδιαίτερα ικανών συστημάτων ευρετηρίασης δεδομένων κειμένου με ικανότητα ανάκτησης των αποτελεσμάτων της αναζήτησης εντός λίγων millisecond. Ένα από τα πιο διαδεδομένα συστήματα είναι το ElasticSearch της εταιρείας elastic Co. Επόμενες εκδόσεις της εφαρμογής θα μπορούσαν να χρησιμοποιήσουν αυτό ή κάποιο αντίστοιχο εργαλείο ώστε να επιταχύνουν την αναζήτηση εντός τόσο του Master όσο και των search buckets. Η συγκεκριμένη προσέγγιση θα μπορούσε να αποδειχτεί ιδιαίτερα αποδοτική κυρίως στην κατανεμημένη μορφή του αλγορίθμου, ενώ για την αμιγώς παράλληλη πιστεύουμε πως θα εισάγει μεγαλύτερες καθυστερήσεις. Επιπρόσθετα θα μπορούσαμε να εισάγουμε τα uniquomes με μορφή JSON στον ElasticSearch πετυχαίνοντας πολύ μεγαλύτερες ταχύτητες αναζήτησης εντός αυτού μετά την κατασκευή του.

13.3 Cloud – Microservices

Το cloud είναι πλέον ιδιαίτερα διαδεδομένο και καινούριες υπηρεσίες προστίθενται σε αυτό. Η δομή της εφαρμογής επιτρέπει την σχετικά εύκολη τροποποίησή της σε μια δομή microservices που με την σειρά τους θα μπορούσαν να μπουν σε κάποιο container (πχ Docker). Με αυτό τον τρόπο ο χρήστης της εφαρμογής θα μπορούσε να επεκτείνει την διαθέσιμη υπολογιστική ισχύ εκτελώντας μερικές απλές εντολές στο τερματικό του, με το οποίο θα μπορούσε να στήσει αυτόματα την απαιτούμενη υποδομή. Θεωρούμε πως η δομή του αλγορίθμου καθώς και ή υλοποίησή του μπορεί να εκμεταλλευτεί πλήρως τις συγκεκριμένες τεχνολογίες.

13.4 Δημιουργία μικρής συστοιχίας με SoC

Όταν ξεκίνησε η υλοποίηση με στόχο να καλύψουμε ακόμη και ιδιαίτερα χαμηλών επιδόσεων μηχανήματα όπως το SoC Raspberry Pi, δεν περιμέναμε ότι θα μπορέσουμε πραγματικά να ολοκληρώσουμε την κατασκευή ενός έστω και μεσαίου μεγέθους uniquome σε αυτό. Αν και πήρε πολλές ώρες η κατασκευή του uniquome του E-coli (τουλάχιστον 5 φορές περισσότερο απ' ότι στο μηχάνημα αναφοράς), εντούτοις η διαδικασία ολοκληρώθηκε με επιτυχία. Το Raspberry είναι ένα πολύ οικονομικό μηχάνημα (περίπου 40 € στα μέσα του 2018) το οποίο παρέχει gigabit ethernet / wifi / ARM επεξεργαστή με 4 πυρήνες, 1GB RAM και μέγιστη κατανάλωση 5 Watt, ενώ υποστηρίζεται πλήρως από τις υπάρχουσες και τις μελλοντικές εκδόσεις του .NET. Μία από τις επιδιώξεις μας είναι η δημιουργία μιας συστοιχίας (Cluster) αρχικά 5 ή 10 μονάδες. Αθροιστικά αυτές μας παρέχουν 20 εώς 40 πυρήνες αρκετή μνήμη για την ανάλυση ακόμη και του ανθρώπινου πρωτεώματος και μέγιστη συνολική κατανάλωση που δεν θα ξεπεράσει τα 50 Watt. Εκτιμούμε πως αν η απόδοση είναι αποδεκτή (τουλάχιστον κοντά σε αυτή του μηχανήματος αναφοράς), τότε μπορούμε εύκολα και οικονομικά να παράσχουμε μια καλή υποδομή κατασκευής και ανάλυσης του Uniquome σε κάθε ερευνητή.

13.5 Δημιουργία Web Server

Θεωρούμε σημαντικό να δημιουργήσουμε ένα Web Service αντίστοιχο του BioServer όπου οι ερευνητές θα μπορούν να αναζητούν πληροφορίες για το uniquome των οργανισμών καθώς και τις μεταναλύσεις που παρέχει η υπάρχουσας εφαρμογή.

14. ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα εργασία επανασχεδιάστηκε ο αλγόριθμος κατασκευής του uniquome ενός οργανισμού ενσωματώνοντας σύγχρονες προσεγγίσεις όπως η παράλληλη επεξεργασία σε πολυπύρηνα συστήματα καθώς η επέκτασή του με χρήση κατανεμημένης επεξεργασίας σε ετερογενή συστήματα.

Η επιλογή μας να χρησιμοποιήσουμε το .NET Core αποδείχτηκε σωστή, αφού η ίδια εφαρμογή εκτελέστηκε με επιτυχία σε όλα τα λειτουργικά συστήματα τα οποία είχαμε θέσει ως στόχο. Παράλληλα η δυνατότητα του να εκτελεστεί (έστω και με χαμηλή ταχύτητα) σε συστήματα SoC όπως το Raspberry Pi μας έδωσε νέες ιδέες χρήσης των συστημάτων αυτών που αργότερα μπορούν να χρησιμοποιηθούν για να απαντηθούν άλλα βιολογικά ερωτήματα.

Κρίνουμε ιδιαίτερα ικανοποιητική την δυνατότητα μέσω μικρών τροποποιήσεων και κυρίως χωρίς να πειράξουμε την παράλληλη και κατανεμημένη υποδομή να ενσωματώσουμε στο σύστημα άλλα βιολογικά ερωτήματα, ακόμη και μη πρωτεωμικά, απλά αλλάζοντας την ερώτηση στην οποία αναζητά απάντηση ο κουβάς εντός του συνόλου των πρωτεϊνών του.

Θεωρούμε πως τα Core και τα Composite Unique Peptides μπορούν να αποκαλύψουν ιδιαίτερης αξία βιολογική πληροφορία. Επίσης λόγω της μοναδικότητάς τους μπορούν να αποτελέσουν ασφαλείς στόχους για τον σχεδιασμό φαρμάκων (αν βρίσκονται εντός κάποιου σημείου πρόσδεσης της πρωτεΐνης) καθώς και να λειτουργήσουν ως βιοδείκτες.

Συγκρίνοντας ομοιότητες ανάμεσα στα uniquomes καλά μελετημένων οργανισμών όπως είναι ο άνθρωπος το ποντίκι και ο αρουραίος, μπορούμε να αποκαλύψουμε συνδέσεις μεταξύ τους που έχουν διατηρηθεί στην της εξέλιξη αυτών.

Καλά διατηρημένα Core Unique Peptides ανάμεσα στους οργανισμούς μπορούν να αποκαλύψουν κοινούς προγόνους των οργανισμών διαμέσου του φυλογενετικού δέντρου, δίνοντας μια νέα οπτική στην μελέτη της εξέλιξης τους.

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ

Ξενόγλωσσος όρος	**Ελληνικός Όρος**
Core Unique Peptide	Πεπτίδιο Μοναδικής Αμινοξικής Ακολουθίας
Composite Unique Peptide	Σύνθετο Πεπτίδιο Μοναδικής Αμινοξικής Ακολουθίας
Central Processing Unit	Κεντρική Μονάδα Επεξεργασίας
Thread	Νήμα εκτέλεσης υπολογισμών

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ

GB	Gigabyte
RAM	Random Access Memory
TCP/IP	Transmission Control Protocol/ Internet Protocol
RDBMS	Relational Database Management System
SoC	System on a Chip
SSD	Solid State Drive
ASCII	American Standard Code for Information Interchange
kDa	Kilo Dalton
VPU	Virtual Processing Unit
CPU	Central Processing Unit
SIMD	Single Instruction Multiple Dataset
AVX	Advanced Vector eXtensions
CUDA	Compute Unified Device Architecture
VPU	Virtual Processing Unit
API	Application Programmer Interface
REST	Representational State Transfer
HTTP	Hyper Text Transfer Protocol

ΑΝΑΦΟΡΕΣ

[1] Miller- Urey Experiment
[2] How Many Species Are There on Earth and in the Ocean?
[3] A preliminary comparison of the mouse and human genomes
[4] Βασικές Αρχές Μοριακής Βιολογίας (Burton E. Tropp) Α Ελληνική Έκδοση.
[5] Proteomics – An Introduction to EMBL – EBI resources
[6] Genome researchers raise alarm over big data.
[7] Big Data: Astronomical or Genomical.
[8] Wikipedia: Biological Databases.
[9] Peptide Finder: mapping measured molecular masses to peptides and proteins
[10] Anastasia Alexandridou, George Th. Tsangaris, Konstantinos Vougas, Konstantina Nikita and George Spyrou : *UniMap: finding unique mass and peptide signatures in the human proteome 2009. Vol. 25 no. 22 2009, pages 3035–3037 doi:10.1093/bioinformatics/btp516*
[11] Κοντοπόδης Ευάγγελος *Ανάλυση των πεπτιδίων μοναδικής αλληλουχίας αμινοξέων (Core Unique Peptides) στο ανθρώπινο πρωτέωμα. Διπλωματική Εργασία*: ΠΙΒ0102 Μάιος 2015
[12] Yafeng Zhu *Mass Spectrometry based proteomics: Data Analysis and applications*. 2018
[13] Stephen Blair Chappel *Parallel Programming with Intel Parallel Studio X,* 2016
[14] Αλγόριθμός και θεωρία Υπολογισμού
[15] https://www.northeastern.edu/levelblog/2016/05/13/how-much-data-produced-every-day/