



ΔΙΕΘΝΕΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΕΛΛΑΔΟΣ

ΔΙΕΘΝΕΣ ΠΑΝΕΠΙΣΤΗΜΙΟ ΤΗΣ ΕΛΛΑΔΟΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ, ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

**ΓΡΑΦΙΚΗ ΕΠΙΛΥΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ ΤΟΥ
ΠΕΡΙΟΔΕΥΟΝΤΟΣ ΠΩΛΗΤΗ ΜΕ ΤΗ ΧΡΗΣΗ
ΑΛΓΟΡΙΘΜΩΝ ΣΥΝΔΥΑΣΜΟΥ ΝΕΥΡΩΝΙΚΩΝ
ΔΙΚΤΥΩΝ ΚΑΙ ΕΞΕΛΙΚΤΙΚΩΝ ΑΛΓΟΡΙΘΜΩΝ**

Πτυχιακή Εργασία της
Δανάης Ραφτοπούλου (154529)

Επιβλέπων: Κ. Γουλιάνας, Επίκουρος Καθηγητής

ΘΕΣΣΑΛΟΝΙΚΗ, ΙΑΝΟΥΑΡΙΟΣ 2025

Υπεύθυνη Δήλωση: Βεβαιώνω ότι είμαι συγγραφέας αυτής της πτυχιακής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην πτυχιακή εργασία. Επίσης έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης, βεβαιώνω ότι αυτή η πτυχιακή εργασία προετοιμάστηκε από εμένα προσωπικά ειδικά για τις απαιτήσεις του προγράμματος σπουδών του Τμήματος Μηχανικών Πληροφορικής, Υπολογιστών και Τηλεπικοινωνιών του Διεθνούς Πανεπιστημίου της Ελλάδας.

Περίληψη

Η παρούσα εργασία εξετάζει την εφαρμογή της υπολογιστικής νοημοσύνης στο κλασικό πρόβλημα του Πλανόδιου Πωλητή (Travelling Salesman Problem – TSP), αξιοποιώντας συνδυαστικά ένα τεχνητό νευρωνικό δίκτυο και έναν εξελικτικό αλγόριθμο και ολοκληρώνεται με τη δημιουργία γραφικού περιβάλλοντος (GUI), μέσω του οποίου ο χρήστης μπορεί να επιλέξει το επιθυμητό σετ πόλεων και να παρακολουθήσει την εξέλιξη της λύσης σε πραγματικό χρόνο.

Περιεχόμενα

1. Νευρωνικά Δίκτυα στην Τεχνητή Νοημοσύνη

1.1 Εισαγωγή στα Νευρωνικά Δίκτυα

1.2 Τι Είναι τα Νευρωνικά Δίκτυα;

1.3 Τύποι Νευρωνικών Δικτύων – Συνοπτική περιγραφή δομής και εφαρμογών

1.4 Μακροπρόθεσμες Βραχυπρόθεσμες Μνήμες (Long Short-Term Memory Networks - LSTM)

1.5 Συναρτήσεις Ενεργοποίησης

1.6 Ζήτημα Καταστροφικής Παρεμβολής στη Δια Βίου Μάθηση

1.6.1 Καταστροφική Παρεμβολή

1.6.2 Το Δίλημμα Σταθερότητας-Ευελιξίας

1.7 Hebbian Πλαστικότητα και Σταθερότητα

1.7.1 Η Θεωρία του Hebb

1.7.2 Σταθεροποίηση Hebbian Πλαστικότητας

1.8 Ομοιοστατική Πλαστικότητα

1.8.1 Μοντέλα Συναπτικής Πλαστικότητας

1.9 Δια βίου μάθηση και plasticity

2. Εξελικτικοί Αλγόριθμοι

2.1 Ιστορική Αναδρομή των Εξελικτικών Αλγορίθμων

2.2 Κατηγορίες Εξελικτικών Αλγορίθμων

2.3 Τι Είναι οι Εξελικτικοί Αλγόριθμοι;

2.4 Βασικές Αρχές των Εξελικτικών Αλγορίθμων

2.5 Ένας Απλός Γενετικός Αλγόριθμος

2.5.1 Επιλογή

2.5.2 Διασταύρωση (Crossover)

2.5.3 Μετάλλαξη

2.5.4 Διατήρηση του Πληθυσμού

2.6 Χαρακτηριστικά Γενετικών Αλγορίθμων

3 Περιγραφή του TSP

3.1 Δυσκολίες του TSP

3.2 Προσεγγίσεις επίλυσης

3.3 Κατηγοριοποίηση των TSP

4. Συνδυαστικές Προσεγγίσεις Νευρωνικών Δικτύων και Εξελικτικών Αλγορίθμων για το TSP

4.1 Παρουσίαση ερευνών

4.1.1 Γενετικός αλγόριθμος για τη βελτιστοποίηση προβλημάτων πλανόδιου πωλητή με χρονικά παράθυρα (TSP-TW)

4.1.2 Υβρίδιο Hopfield Neural Network (HNN) και Γενετικού Αλγορίθμου (GA)

4.1.3 Εξέλιξη Αυτό - Οργανωνόμενου Χάρτη (SOM) με GA

4.1.4 Βελτιστοποίηση του TSP με χρήση ομαδοποίησης NN και GA

4.1.5 Χαοτικό Νευρωνικό Δίκτυο + GA σε προβλήματα διαδρομής

4.1.6 SOM με Εξελικτική Αναζήτηση για Multi-TSP

4.2 Συμπέρασμα από τη βιβλιογραφία

5. Πρακτική Επίλυση

5.1 Εισαγωγή

5.2 Περιγραφή Μεθοδολογίας και Αλγορίθμου

5.2.1 Δεδομένα

5.2.2 Νευρωνικό Δίκτυο

5.2.3 Εξελικτικός Αλγόριθμος

5.2.4 Ανάλυση Αποτελεσμάτων

5.2.4.1 Σετ Τεσσάρων Πόλεων

5.2.4.2 Σετ Δέκα Πόλεων

5.2.4.3 Σετ Είκοσι Πόλεων

5.2.4.4 Σετ Πενήντα Πόλεων

5.2.6 Συμπεράσματα επίλυσης προβλήματος

5.2.7 Γραφικό περιβάλλον (GUI)

6. Βιβλιογραφία

Εισαγωγή

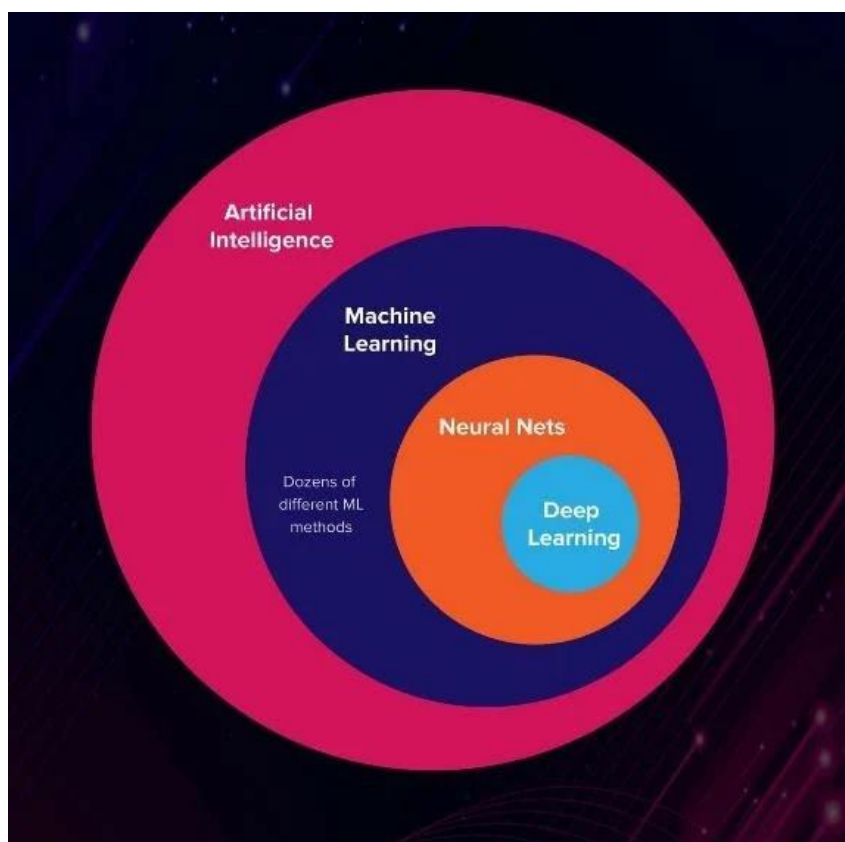
Η παρούσα εργασία εξετάζει την εφαρμογή της υπολογιστικής νοημοσύνης στο κλασικό πρόβλημα του Πλανόδιου Πωλητή (Travelling Salesman Problem – TSP), αξιοποιώντας συνδυαστικά ένα τεχνητό νευρωνικό δίκτυο και έναν εξελικτικό αλγόριθμο. Αρχικά, παρουσιάζεται το θεωρητικό υπόβαθρο των τεχνολογιών που χρησιμοποιούνται. Ακολουθεί σύντομη ανασκόπηση της σχετικής βιβλιογραφίας και προηγούμενων υλοποιήσεων του TSP μέσω τεχνητής νοημοσύνης, όπου καταγράφονται βασικά πλεονεκτήματα και περιορισμοί. Ιδιαίτερη έμφαση δίνεται σε υβριδικές προσεγγίσεις που συνδυάζουν νευρωνικά δίκτυα και γενετικούς αλγόριθμους, οι οποίες έχουν αποδειχθεί αποτελεσματικές σε προβλήματα βελτιστοποίησης. Στο πρακτικό μέρος, σχεδιάστηκε και υλοποιήθηκε ένα σύστημα επίλυσης του TSP με εισαγωγικά δεδομένα από 4 έως 50 ευρωπαϊκές πόλεις. Το νευρωνικό δίκτυο χρησιμοποιείται ως μηχανισμός αρχικοποίησης πληθυσμού, ώστε να βελτιωθεί η ποιότητα των αρχικών λύσεων, ενώ ο εξελικτικός αλγόριθμος αναλαμβάνει την εξερεύνηση του διαστήματος λύσεων. Πραγματοποιούνται συγκρίσεις με τη μέθοδο MST + DFS και Brute Force, ενώ η απόδοση αξιολογείται τόσο ως προς την ποιότητα της διαδρομής όσο και του συνολικού μήκους διαδρομής. Η εργασία ολοκληρώνεται με τη δημιουργία γραφικού περιβάλλοντος (GUI), μέσω του οποίου ο χρήστης μπορεί να επιλέξει το επιθυμητό σετ πόλεων και να παρακολουθήσει την εξέλιξη της λύσης σε πραγματικό χρόνο.

1. Νευρωνικά Δίκτυα στην Τεχνητή Νοημοσύνη

Η Τεχνητή Νοημοσύνη (Artificial Intelligence) αποτελεί τη συστηματική μελέτη του σχεδιασμού και της ανάπτυξης ευφών συστημάτων και προγραμμάτων. Ένα βασικό υποσύνολο της Τεχνητής Νοημοσύνης είναι η Μηχανική Μάθηση (Machine Learning), η οποία επικεντρώνεται στον τρόπο με τον οποίο οι μηχανές μπορούν να μαθαίνουν αυτόνομα από δεδομένα. Στο πλαίσιο της Μηχανικής Μάθησης, η Βαθιά

Μάθηση (Deep Learning) εξετάζει περαιτέρω τη χρήση στρωμάτων νευρωνικών δικτύων για τη δημιουργία σύνθετων εξόδων.

Τα Νευρωνικά Δίκτυα (Neural Networks) αποτελούν ένα σύστημα επεξεργασίας δεδομένων και παραγωγής αποτελεσμάτων, εμπνευσμένο από τη δομή και τη λειτουργία του ανθρώπινου εγκεφάλου. Αυτά τα συστήματα είναι ιδανικά για την αποκάλυψη μη γραμμικών σχέσεων σε μεγάλα σύνολα δεδομένων. Τα δεδομένα μπορεί να προέρχονται από διάφορες αισθητηριακές πηγές, όπως κείμενο, εικόνες ή ήχος. Η παραπάνω ιεραρχία αποτυπώνεται γραφικά στο παρακάτω σχήμα.



Εικόνα 1. Η εικόνα απεικονίζει τη σχέση μεταξύ της Τεχνητής Νοημοσύνης (AI), της Μηχανικής Μάθησης (ML), των Νευρωνικών Δικτύων και της Βαθιάς Μάθησης (Deep Learning). Οι τέσσερις ομόκεντροι κύκλοι δείχνουν την ιεραρχία, με την AI να περιβάλλει τις υπόλοιπες τεχνολογίες, υποδηλώνοντας τη βαθμιαία εξειδίκευση από τη Μηχανική Μάθηση στα Νευρωνικά Δίκτυα και τέλος στη Βαθιά Μάθηση.

1.1 Εισαγωγή στα Νευρωνικά Δίκτυα

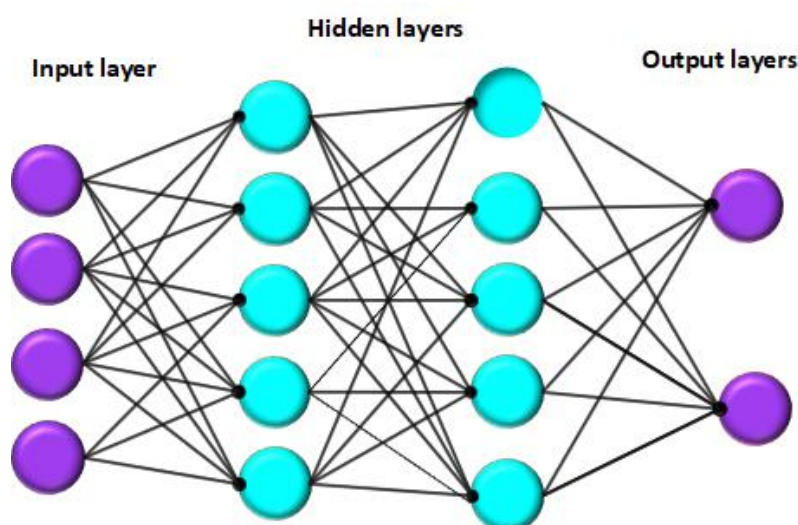
Κλασικά, ο ορισμός της νοημοσύνης βασίζεται σε μεγάλο βαθμό στις ικανότητες των προηγμένων βιολογικών οντοτήτων, με κυρίαρχο παράδειγμα τον άνθρωπο. Ως εκ τούτου, η έρευνα στον τομέα της Τεχνητής Νοημοσύνης (TN) έχει επικεντρωθεί στη δημιουργία μηχανών που μπορούν να αντιλαμβάνονται, να μαθαίνουν και να σκέφτονται. Στόχος τους δηλαδή είναι να μιμηθούν την ανθρώπινη νοημοσύνη και να προσεγγίσουν την ευελιξία της σκέψης και της μάθησης που διαθέτουν οι άνθρωποι.

Στο πλαίσιο αυτό, είναι φυσικό οι επιστήμονες, οι μαθηματικοί και οι φιλόσοφοι που ασχολούνται με την Τεχνητή Νοημοσύνη να αντλούν έμπνευση από τη δομή και τη λειτουργία του ανθρώπινου εγκεφάλου. Οι Νευρωνικές Δομές που υπάρχουν στη βιολογία αποτέλεσαν τη βάση για την ανάπτυξη των Νευρωνικών Δικτύων, τα οποία προσφέρουν έναν αποδοτικό τρόπο αναπαράστασης και επεξεργασίας της πληροφορίας για την επίλυση σύνθετων προβλημάτων.

1.2 Τι Είναι τα Νευρωνικά Δίκτυα;

Τα νευρωνικά δίκτυα, επίσης γνωστά ως τεχνητά νευρωνικά δίκτυα (ANN) ή προσομοιωμένα νευρωνικά δίκτυα (SNN), αποτελούν ένα υποσύνολο της μηχανικής μάθησης και βρίσκονται στο επίκεντρο των αλγορίθμων βαθιάς μάθησης.

Τα τεχνητά νευρωνικά δίκτυα (ANN) αποτελούνται από ένα επίπεδο κόμβων, το οποίο περιλαμβάνει ένα επίπεδο εισόδου, ένα ή περισσότερα κρυφά επίπεδα και ένα επίπεδο εξόδου. Κάθε κόμβος, ή τεχνητός νευρώνας, συνδέεται με έναν άλλο και έχει ένα σχετικό βάρος και κατώφλι. Εάν η έξοδος οποιουδήποτε μεμονωμένου κόμβου είναι πάνω από την καθορισμένη τιμή κατωφλίου, ο εν λόγω κόμβος ενεργοποιείται, στέλνοντας δεδομένα στο επόμενο στρώμα του δικτύου. Διαφορετικά, δεν διαβιβάζονται δεδομένα στο επόμενο επίπεδο του δικτύου.



Εικόνα 2. Διάγραμμα νευρωνικού δικτύου με επίπεδα εισόδου, κρυφά και εξόδου. Τα δεδομένα ρέουν από τα αριστερά προς τα δεξιά μέσω συνδέσεων μεταξύ των κόμβων.

Τα νευρωνικά δίκτυα βασίζονται σε δεδομένα εκπαίδευσης για να μαθαίνουν και να βελτιώνουν την ακρίβειά τους με την πάροδο του χρόνου. Ωστόσο, μόλις αυτοί οι αλγόριθμοι μάθησης ρυθμιστούν με ακρίβεια, αποτελούν ισχυρά εργαλεία στην επιστήμη των υπολογιστών και την τεχνητή νοημοσύνη, επιτρέποντάς μας να ταξινομήσουμε και να ομαδοποιήσουμε δεδομένα με μεγάλη ταχύτητα. Οι εργασίες στην αναγνώριση ομιλίας ή στην αναγνώριση εικόνας μπορούν να διαρκέσουν λίγα λεπτά έναντι ωρών σε σύγκριση με τη χειροκίνητη αναγνώριση από ανθρώπινους εμπειρογνώμονες. Ένα από τα πιο γνωστά νευρωνικά δίκτυα είναι ο αλγόριθμος αναζήτησης της Google.

1.3 Τύποι Νευρωνικών Δικτύων – Συνοπτική περιγραφή δομής και εφαρμογών

1. Πολυστρωματικά Νευρωνικά Δίκτυα (Feedforward Neural Networks - FNN)

Τα FNN είναι τα βασικότερα νευρωνικά δίκτυα και αποτελούνται από διάφορα επίπεδα νευρώνων, τα οποία περιλαμβάνουν το επίπεδο εισόδου, ένα ή περισσότερα κρυφά επίπεδα, και το επίπεδο εξόδου. Οι νευρώνες συνδέονται με κατευθυνόμενες συνδέσεις χωρίς αναδρομικές συνδέσεις, δηλαδή οι πληροφορίες ρέουν μόνο προς τα εμπρός. Χρησιμοποιούνται σε κλασικές εφαρμογές όπως ταξινόμηση, αναγνώριση προτύπων, και πρόβλεψη, κυρίως όταν οι σχέσεις μεταξύ των δεδομένων είναι μη γραμμικές.

2. Αναδρομικά Νευρωνικά Δίκτυα (Recurrent Neural Networks - RNN)

Τα RNN διαθέτουν κυκλικές συνδέσεις που επιτρέπουν στα δίκτυα να διατηρούν πληροφορίες από προηγούμενα χρονικά βήματα. Κάθε νευρώνας τροφοδοτεί την επόμενη χρονική στιγμή τόσο με τις εισόδους του όσο και με τις εξόδους από το προηγούμενο χρονικό βήμα. Έτσι, τα RNN έχουν τη δυνατότητα να αποθηκεύουν και να επεξεργάζονται σειριακά δεδομένα, καθιστώντας τα κατάλληλα για ανάλυση χρονοσειρών και αναγνώριση ομιλίας. Η χρήση τους εντοπίζεται σε εφαρμογές που απαιτούν μνήμη όπως αναγνώριση ομιλίας και μετάφραση κειμένου.

3. Συνελκτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks - CNN)

Τα CNN έχουν σχεδιαστεί ειδικά για την επεξεργασία εικόνων και άλλων

δομημένων δικτύων δεδομένων. Περιλαμβάνουν επίπεδα που εκτελούν συνελίξεις (convolutions) σε συγκεκριμένες περιοχές της εικόνας, και στη συνέχεια ακολουθούν επίπεδα που μειώνουν το μέγεθος των δεδομένων (pooling), διατηρώντας τις πιο σημαντικές πληροφορίες. Τα τελικά επίπεδα συνήθως είναι πλήρως συνδεδεμένα και εκτελούν την τελική ταξινόμηση. Τα CNN εφαρμόζονται για αναγνώριση και ανάλυση εικόνων, ανίχνευση αντικειμένων, επεξεργασία βίντεο, και εφαρμογές όπου η βελτιστοποίηση του χώρου που καταλαμβάνουν τα δεδομένα είναι σημαντική.

4. Νευρωνικά Δίκτυα Γεννήτριας Αντιπαράθεσης (Generative Adversarial Networks - GAN)

Τα GAN αποτελούνται από δύο υποδίκτυα: ένα γεννήτορα (generator) και έναν διακριτή (discriminator). Ο γεννήτορας δημιουργεί δεδομένα που προσπαθεί να είναι παρόμοια με τα πραγματικά δεδομένα, ενώ ο διακριτής προσπαθεί να διακρίνει μεταξύ των πραγματικών και των παραγόμενων δεδομένων. Η αντιπαράθεση μεταξύ των δύο δικτύων οδηγεί στην προοδευτική βελτίωση του γεννήτορα μέχρι τα παραγόμενα δεδομένα να είναι πολύ ρεαλιστικά. Τα GAN χρησιμοποιούνται για τη δημιουργία συνθετικών δεδομένων, ενίσχυση εικόνων, δημιουργία ρεαλιστικών φωτογραφιών, και προσομοιώσεις για εκπαίδευση άλλων μοντέλων.

1.4 Μακροπρόθεσμες Βραχυπρόθεσμες Μνήμες (Long Short-Term Memory Networks - LSTM)

Τα LSTM είναι ειδικές δομές των RNN που μπορούν να αποθηκεύουν πληροφορίες για μεγάλες χρονικές περιόδους. Χρησιμοποιούν μηχανισμούς πύλες (gates) που ρυθμίζουν τη ροή των πληροφοριών μέσω των κυψελών μνήμης (memory cells), επιτρέποντας τους να διατηρούν σημαντικές πληροφορίες και να απορρίπτουν αχρείαστες. Αυτό επιλύει προβλήματα εξαφάνισης ή έκρηξης βαθμίδων (vanishing/exploding gradients). Τα LSTM είναι αποδοτικά για προβλέψεις χρονοσειρών, μοντελοποίηση γλώσσας, αναγνώριση ομιλίας, και εφαρμογές που απαιτούν επεξεργασία μεγάλης μνήμης.

1.5 Συναρτήσεις Ενεργοποίησης

Μια συνάρτηση ενεργοποίησης είναι μια μαθηματική συνάρτηση που εφαρμόζεται στην έξοδο ενός νευρώνα. Εισάγει μη γραμμικότητα στο μοντέλο, επιτρέποντας στο δίκτυο να μαθαίνει και να αναπαριστά σύνθετα μοτίβα στα δεδομένα. Χωρίς αυτό το χαρακτηριστικό της μη γραμμικότητας, ένα νευρωνικό δίκτυο θα συμπεριφερόταν σαν ένα μοντέλο γραμμικής παλινδρόμησης, ανεξάρτητα από το πόσα επίπεδα έχει.

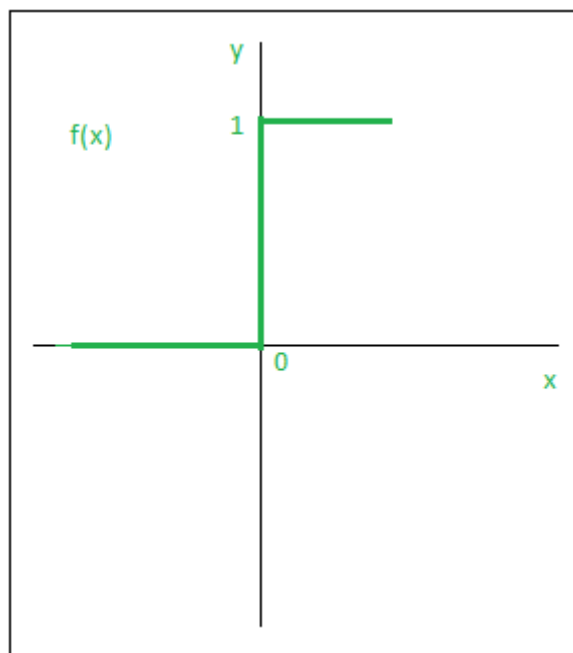
Η συνάρτηση ενεργοποίησης αποφασίζει αν ένας νευρώνας πρέπει να ενεργοποιηθεί υπολογίζοντας το σταθμισμένο άθροισμα των εισόδων και προσθέτοντας έναν όρο απόκλισης. Αυτό βοηθά το μοντέλο να λαμβάνει πολύπλοκες αποφάσεις και προβλέψεις εισάγοντας μη γραμμικές συνθήκες στην έξοδο κάθε νευρώνα.

1.5.1 Βηματική Συνάρτηση (Step Function)

Είναι η πιο απλή μορφή συνάρτησης ενεργοποίησης. Η έξοδος είναι 0 εάν η είσοδος είναι κάτω από ένα συγκεκριμένο κατώφλι και 1 εάν η είσοδος είναι πάνω από το κατώφλι, κάνοντάς την ιδανική για προβλήματα κατηγοριών με δύο επιλογές. Το μειονέκτημά της είναι ότι δεν είναι ιδανική για περίπλοκες εργασίες, καθώς δεν προσφέρει μια ποικιλία εξόδων και δεν επιτρέπει σταδιακές αλλαγές. Η μαθηματική της έκφραση είναι:

$$f(x) = 1, \text{ if } x \geq 0$$

$$f(x) = 0, \text{ if } x < 0$$



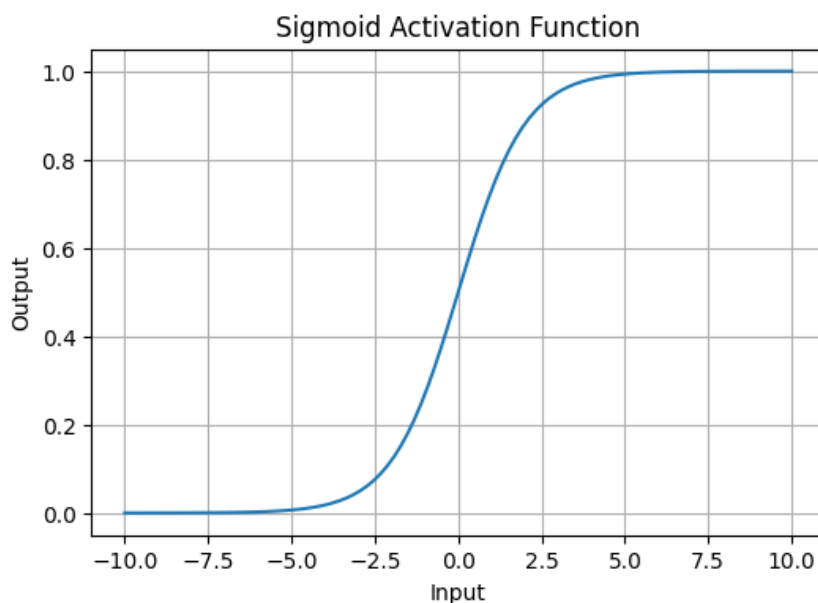
Εικόνα 3. Step Function

1.5.2 Σιγμοειδής Συνάρτηση (Sigmoid Function)

Η συνάρτηση σιγμοειδούς έχει μια καμπύλη σε σχήμα S, με τιμές εξόδου που κυμαίνονται από 0 έως 1. Η μαθηματική της έκφραση είναι:

$$A = \frac{1}{1 + e^x}$$

Συχνά χρησιμοποιείται σε δίκτυα με πολλά επίπεδα καθώς επιτρέπει τη μοντελοποίηση μη γραμμικών σχέσεων. Το μειονέκτημά της είναι ότι μπορεί να προκαλέσει το πρόβλημα της εξαφάνισης βαθμίδων (vanishing gradients), κάνοντας την εκπαίδευση των δικτύων πιο αργή και λιγότερο αποτελεσματική.



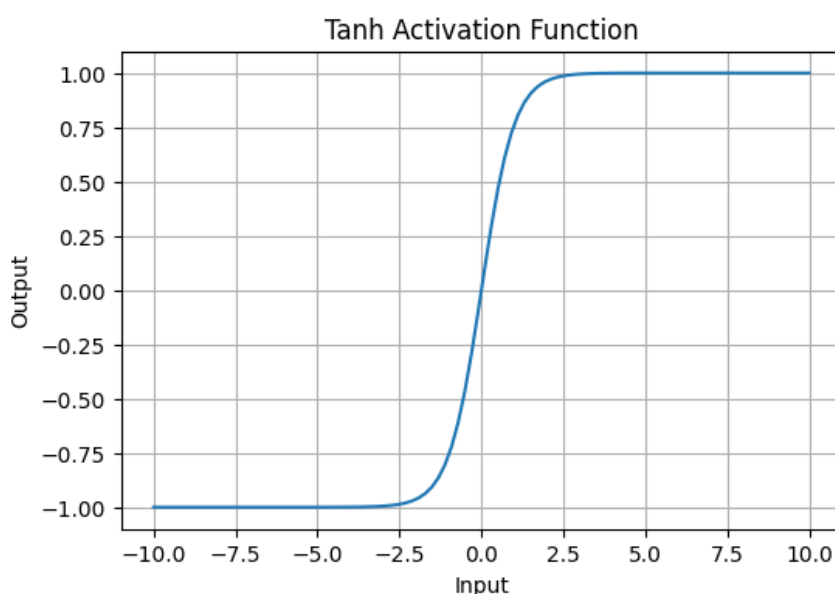
Εικόνα 4. Sigmoid Function

1.5.3 Συνάρτηση Υπερβολικού Εφαπτομένου (Tanh Function)

Είναι παρόμοια με τη σιγμοειδή, αλλά οι τιμές εξόδου κυμαίνονται από -1 έως 1. Η μαθηματική της έκφραση είναι:

$$F(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1$$

Συχνά προτιμάται σε δίκτυα όπου οι τιμές εισόδου μπορεί να είναι θετικές ή αρνητικές, παρέχοντας ένα πιο ισχυρό διαχωρισμό των ενεργοποιήσεων. Αν και μειώνει τα προβλήματα εξαφάνισης βαθμίδων σε σύγκριση με τη σιγμοειδή, εξακολουθεί να μην είναι ιδανική για πολύ βαθιά δίκτυα.



Εικόνα 5. Tanh Function

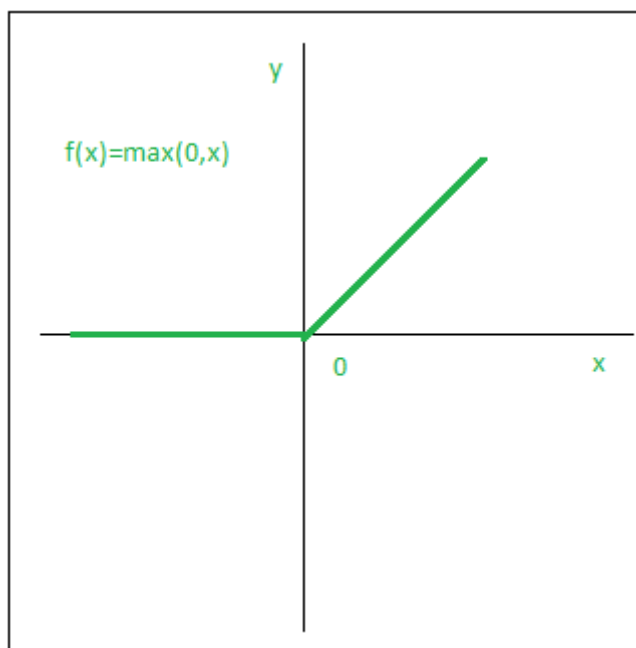
1.5.4 Συνάρτηση ReLU (Rectified Linear Unit)

Η ReLU είναι η πιο δημοφιλής συνάρτηση ενεργοποίησης για βαθιά νευρωνικά δίκτυα. Η μαθηματική της έκφραση είναι:

$$F(x) = \max(0, x)$$

αυτό σημαίνει ότι αν η είσοδος x είναι θετική, η ReLU επιστρέφει x , ενώ αν η είσοδος είναι αρνητική, επιστρέφει 0.

Χρησιμοποιείται ευρέως λόγω της απλότητάς της και της ικανότητάς της να επιταχύνει τη σύγκλιση των δικτύων. Το μειονέκτημά της είναι ότι μπορεί να προκαλέσει το πρόβλημα της “νεκρής ReLU” (dead ReLU), όπου νευρώνες παύουν να ενεργοποιούνται για όλες τις εισόδους τους.



Εικόνα 6. Rectified Linear Unit

1.5.5 Συνάρτηση Softmax

Η Softmax έχει σχεδιαστεί για να χειρίζεται προβλήματα ταξινόμησης πολλαπλών κατηγοριών. Μετατρέπει τις ανεπεξέργαστες βαθμολογίες εξόδου από ένα νευρωνικό δίκτυο σε πιθανότητες. Λειτουργεί συμπιέζοντας τις τιμές εξόδου κάθε κλάσης στο εύρος 0 έως 1, ενώ διασφαλίζει ότι το άθροισμα όλων των πιθανοτήτων ισούται με 1. Η μαθηματική της έκφραση είναι:

$$\sigma(\underline{z})_i = \frac{e^{z^i}}{\sum_{j=1}^K e^{z^j}}$$

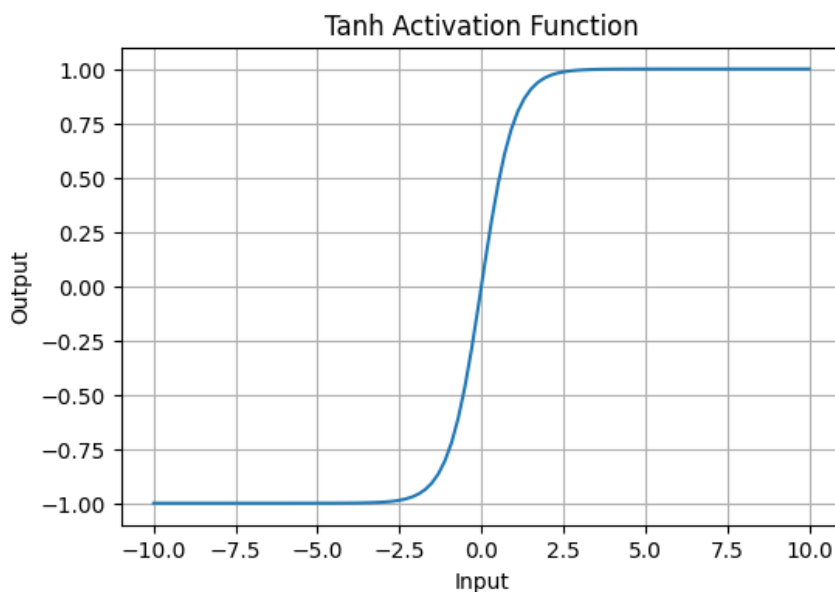
σ = softmax

\underline{z} = διάνυσμα εισόδου

e^{z^i} = τυπική εκθετική συνάρτηση για το διάνυσμα εισόδου

K = αριθμός κλάσεων στον ταξινομητή πολλαπλών κλάσεων

e^{z^j} = τυπική εκθετική συνάρτηση για το διάνυσμα εξόδου



Εικόνα 7. *Softmax*

1.6 Ζήτημα Καταστροφικής Παρεμβολής στη Δια Βίου Μάθηση

1.6.1 Καταστροφική Παρεμβολή

Ένα από τα κύρια ζητήματα που αντιμετωπίζουν τα υπολογιστικά μοντέλα στη δια βίου μάθηση είναι η καταστροφική παρεμβολή. Το φαινόμενο αυτό συμβαίνει όταν η εκπαίδευση ενός μοντέλου με νέες πληροφορίες υποβαθμίζει ή καταργεί τις γνώσεις που είχαν αποκτηθεί προηγουμένως[1]. Αυτό μπορεί να οδηγήσει είτε σε μείωση της απόδοσης είτε, στη χειρότερη περίπτωση, στην πλήρη αντικατάσταση της παλιάς γνώσης από τη νέα.

Τα υπάρχοντα μοντέλα εκμάθησης νευρωνικών δικτύων έχουν αποδειχθεί αποτελεσματικά σε εργασίες ταξινόμησης, αξιοποιώντας μεγάλες συλλογές δειγμάτων[2]. Ωστόσο, αυτή η προσέγγιση προϋποθέτει την πλήρη διαθεσιμότητα όλων των δεδομένων κατά τη φάση της εκπαίδευσης. Σε περιπτώσεις όπου τα δεδομένα καθίστανται σταδιακά διαθέσιμα, η απόδοση των παραδοσιακών νευρωνικών δικτύων μειώνεται δραστικά στις προηγουμένως αποκτηθείσες εργασίες καθώς μαθαίνονται νέες[3].

Η επανεκπαίδευση από την αρχή, παρότι επιλύει το πρόβλημα της καταστροφικής παρεμβολής, είναι εξαιρετικά μη αποτελεσματική. Αυτό καθιστά δύσκολη την εκμάθηση νέων δεδομένων σε πραγματικό χρόνο, ειδικά σε σενάρια αναπτυξιακής

μάθησης καθώς πρέπει να γίνεται άμεση προσαρμογή του μοντέλου στο περιβάλλον τους[4].

1.6.2 Το Δίλημμα Σταθερότητας-Ευελιξίας

Η αποτελεσματική διαχείριση της καταστροφικής παρεμβολής απαιτεί την επίτευξη της ισορροπίας ανάμεσα στη σταθερότητα και την ευελιξία. Τα συστήματα μάθησης πρέπει, αφενός, να αποκτούν νέες γνώσεις και να προσαρμόζονται σε μεταβαλλόμενα δεδομένα και, αφετέρου, να διατηρούν τη συνοχή της υπάρχουσας γνώσης. Αυτό το "δίλημμα σταθερότητας-ευελιξίας" έχει μελετηθεί εκτενώς τόσο σε βιολογικά όσο και σε υπολογιστικά συστήματα[5].

1.7 Hebbian Πλαστικότητα και Σταθερότητα

Πειράματα έχουν δείξει ότι τα νευρωνικά μοτίβα της φλοιώδους οργάνωσης μπορούν να επηρεαστούν από εξωτερικούς περιβαλλοντικούς παράγοντες, τουλάχιστον κατά την αρχική φάση της ανάπτυξης[6].

1.7.1 Η Θεωρία του Hebb

Η πιο γνωστή θεωρία που περιγράφει τους μηχανισμούς της συναπτικής πλαστικότητας προτάθηκε για πρώτη φορά από τον Hebb (1949). Σύμφωνα με αυτήν, όταν ένας νευρώνας ενεργοποιεί τη δραστηριότητα ενός άλλου νευρώνα, η σύνδεση μεταξύ τους ενισχύεται. Κατά την ανάπτυξη, τα νευρωνικά συστήματα σταθεροποιούνται για να διαμορφώσουν βέλτιστα λειτουργικά πρότυπα νευρωνικής συνδεσιμότητας.

1.7.2 Σταθεροποίηση Hebbian Πλαστικότητας

Η σταθερότητα στα Hebbian συστήματα επιτυγχάνεται με την ενίσχυση της Hebbian πλαστικότητας μέσω πρόσθετων περιορισμών, όπως:

- Ανώτατα όρια στα συναπτικά βάρη.
- Περιορισμοί στη μέση νευρωνική δραστηριότητα[7]

Η διαμορφωμένη Hebbian πλαστικότητα περιλαμβάνει την εισαγωγή ενός πρόσθετου διαμορφωτικού σήματος m , οδηγώντας στην τροποποιημένη εξίσωση:

$$\Delta w = m \cdot x \cdot y - \eta \quad (1)$$

όπου η είναι ένας δεδομένος ρυθμός μάθησης. Αυτή η τροποποιημένη έκδοση διασφαλίζει ότι η Hebbian πλαστικότητα λειτουργεί με πιο σταθερό τρόπο, καθιστώντας την κατάλληλη για σύνθετα νευρωνικά συστήματα.

Ωστόσο, η Hebbian πλαστικότητα από μόνη της είναι ασταθής και οδηγεί σε ανεξέλεγκτη νευρική δραστηριότητα, απαιτώντας έτσι αντισταθμιστικούς μηχανισμούς για τη σταθεροποίηση της διαδικασίας μάθησης[8]. Η σταθερότητα στα Hebbian συστήματα επιτυγχάνεται συνήθως με την ενίσχυση της Hebbian πλαστικότητας με πρόσθετους περιορισμούς, όπως ανώτατα όρια στα επιμέρους συναπτικά βάρη ή στη μέση νευρωνική δραστηριότητα[9]. Στην απλούστερη μορφή της, η διαμορφωμένη Hebbian πλαστικότητα μπορεί να μοντελοποιηθεί με την εισαγωγή ενός πρόσθετου διαμορφωτικού σήματος m στην εξίσωση (1), έτσι ώστε η συναπτική ενημέρωση να δίνεται ως εξής:

$$\Delta w = m - x - y - \eta \quad (2)$$

1.8 Ομοιοστατική Πλαστικότητα

Στις νευροεπιστήμες, η ομοιόσταση ή η ομοιοστατική πλαστικότητα αναφέρεται συχνά σε μια αντισταθμιστική διαδικασία που σταθεροποιεί τους ρυθμούς πυροδότησης των νευρώνων. Ένα γνωστό παράδειγμα είναι η ομοιοστατική ρύθμιση της θερμοκρασίας του σώματος στα θηλαστικά, η οποία διατηρείται στους 37°C περίπου ανεξάρτητα από τις καιρικές συνθήκες και τη θερμοκρασία του αέρα.

Έτσι, η καλύτερα μελετημένη πειραματικά μορφή ομοιόστασης δρα σε μια χρονική κλίμακα ωρών έως ημερών. Αυτή η αργή μορφή ομοιοστατικής πλαστικότητας εκδηλώνεται ως η αναπροσαρμογή της αποτελεσματικότητας ή του βάρους όλων των προσαγωγών συνάψεων σε έναν μόνο νευρώνα κατά ένα σταθερό κλάσμα, για παράδειγμα 0,78. Το φαινόμενο αυτό ονομάζεται "συναπτική κλιμάκωση". Τα χαρακτηριστικά της συναπτικής κλιμάκωσης συνοπτικά είναι τα παρακάτω:

- Πρόκειται για αναπροσαρμογή της αποτελεσματικότητας ή του βάρους όλων των προσαγωγών συνάψεων ενός νευρώνα.
- Η αναπροσαρμογή γίνεται κατά ένα σταθερό κλάσμα, όπως π.χ. 0,78.

Αυτή η σταδιακή αλλαγή επιτρέπει στα νευρικά κυκλώματα να παραμένουν λειτουργικά και σταθερά, διατηρώντας τους ρυθμούς πυροδότησης σε φυσιολογικά επίπεδα.

1.8.1 Μοντέλα Συναπτικής Πλαστικότητας

Η συναπτική πλαστικότητα υφίσταται σε διαφορετικές χρονικές κλίμακες. Για παράδειγμα, οι συναπτικές αλλαγές που προκαλούνται από μια ακολουθία τεσσάρων προσυναπτικών αιχμών σε ταχεία αλληλουχία συνήθως αποσυντίθενται μέσα σε μερικές εκατοντάδες χιλιοστά του δευτερολέπτου και ονομάζονται βραχυπρόθεσμη πλαστικότητα. Η ταχεία αποσύνθεση υποδηλώνει ότι οι αλλαγές δεν είναι χρήσιμες για το σχηματισμό μακροπρόθεσμης μνήμης.

Άλλες μορφές πλαστικότητας που προκαλούνται από τα κλασικά πρωτόκολλα επαγωγής μπορούν να έχουν μακροπρόθεσμες επιδράσεις στη χρονική κλίμακα των ωρών ή και περισσότερο. Η μακροπρόθεσμη πλαστικότητα είναι επομένως δυνητικά χρήσιμη για το σχηματισμό μνήμης. Η επαγωγή της μακροχρόνιας πλαστικότητας μπορεί να είναι τόσο γρήγορη όσο δευτερόλεπτα, αλλά οι επαγόμενες αλλαγές παραμένουν για πολύ μεγαλύτερο χρονικό διάστημα. Ανάλογα με την κατεύθυνση της συναπτικής αλλαγής, μιλάμε για μακροχρόνια ενδυνάμωση (LTP) ή μακροχρόνια κατάθλιψη (LTD).

Υπό κατάλληλες συνθήκες οι αλλαγές που επάγονται από ένα πρωτόκολλο LTP ή LTD παγιώνονται περαιτέρω μετά από περίπου μία ώρα. Αυτά τα αποτελέσματα αναφέρονται συχνά ως μακροπρόθεσμη πλαστικότητα όψιμης φάσης.

1.9 Δια βίου μάθηση και placticity

Ο εγκέφαλος των θηλαστικών παραμένει το καλύτερο μοντέλο δια βίου μάθησης, γεγονός που καθιστά τα μοντέλα μάθησης που εμπνέονται από τη βιολογία μια ελκυστική προσέγγιση. Η γενική έννοια της structural plasticity χρησιμοποιείται ευρέως στη βιβλιογραφία της μηχανικής μάθησης και αποτελεί από μόνη της μια πολλά υποσχόμενη λύση για τη δια βίου μάθηση, ακόμη και όταν δεν λαμβάνονται υπόψη τα βιολογικά επιθυμητά στοιχεία. Οι προτεινόμενες υπολογιστικές λύσεις για τον μετριασμό της καταστροφικής λήθης και της παρεμβολής έχουν επικεντρωθεί στη

ρύθμιση των εγγενών επιπέδων πλαστικότητας για την προστασία της αποκτηθείσας γνώσης, στη δυναμική κατανομή νέων νευρώνων ή στρωμάτων δικτύου για την υποδοχή νέων γνώσεων και στη χρήση συμπληρωματικών δικτύων μάθησης με αναπαραγωγή εμπειριών για την ανάπτυξη της μνήμης. Ωστόσο, παρά τις σημαντικές προόδους, τα τρέχοντα μοντέλα δια βίου μάθησης απέχουν ακόμη πολύ από το να παρέχουν την ευελιξία, την ευρωστία και την επεκτασιμότητα που παρουσιάζουν τα βιολογικά συστήματα. Τα πιο δημοφιλή μοντέλα βαθιάς και ρηχής μάθησης της δια βίου μάθησης περιορίζονται στον τομέα της επιβλεπόμενης μάθησης, βασισμένα σε μεγάλες ποσότητες σχολιασμένων δεδομένων που συλλέγονται σε ελεγχόμενα περιβάλλοντα.

2. Εξελικτικοί Αλγόριθμοι

Οι εξελικτικοί αλγόριθμοι (EA) είναι μια ομάδα τεχνικών αναζήτησης και βελτιστοποίησης που βασίζονται στην ιδέα της φυσικής εξέλιξης, όπως την περιέγραψε ο Δαρβίνος στην θεωρία της φυσικής επιλογής. Αυτές οι τεχνικές έχουν αποδειχθεί εξαιρετικά αποτελεσματικές για τη βελτιστοποίηση στόχων σε πολύπλοκα προβλήματα και έχουν ευρύ φάσμα εφαρμογών στους τομείς της βιολογίας, της ρομποτικής, των τηλεπικοινωνιών, της μηχανικής μάθησης, της χημείας και άλλων.

Η διαδικασία της φυσικής επιλογής, που βασίζεται στην επιβίωση των καταλληλότερων οργανισμών, προσομοιώνεται στους εξελικτικούς αλγορίθμους μέσω μηχανισμών όπως η επιλογή, η αναπαραγωγή, η μετάλλαξη και η ανασύνθεση. Αυτοί οι αλγόριθμοι προσφέρουν ένα αποτελεσματικό τρόπο αναζήτησης λύσεων σε προβλήματα με μεγάλο χώρο λύσεων και πολύπλοκες ή μη γραμμικές σχέσεις.

2.1 Ιστορική Αναδρομή των Εξελικτικών Αλγορίθμων

Η έννοια των εξελικτικών αλγορίθμων πρωτοαναπτύχθηκε στις αρχές της δεκαετίας του 1960, όταν ο John Holland εισήγαγε τις πρώτες ιδέες γύρω από τη γενετική αναζήτηση και τη βελτιστοποίηση. Ο Holland μελέτησε τα χαρακτηριστικά των φυσικών εξελικτικών διαδικασιών και διαπίστωσε ότι οι μηχανισμοί όπως η αναπαραγωγή, η μετάλλαξη και η επιβίωση των καλύτερων οργανισμών μπορούν να

χρησιμοποιηθούν για την αναζήτηση βελτιστοποιημένων λύσεων σε υπολογιστικά προβλήματα.

Αργότερα, τη δεκαετία του 1970, οι εξελικτικοί αλγόριθμοι επεκτάθηκαν και εξελίχθηκαν, με τη δημιουργία των γενετικών αλγορίθμων (GAs), οι οποίοι αποτέλεσαν τη βάση για πολλές μετέπειτα εξελικτικές τεχνικές. Οι εξελικτικοί αλγόριθμοι βρήκαν εφαρμογές σε πλήθος τομέων, όπως η βιολογία, η ιατρική, η μηχανική μάθηση, και άλλοι.

2.2 Κατηγορίες Εξελικτικών Αλγορίθμων

Οι εξελικτικοί αλγόριθμοι μπορούν να διαχωριστούν σε διάφορες κατηγορίες ανάλογα με τη φύση και τη διαδικασία που ακολουθούν. Οι κυριότεροι κλάδοι περιλαμβάνουν:

2.2.1 Γενετικοί Αλγόριθμοι (Genetic Algorithms - GAs)

Οι γενετικοί αλγόριθμοι είναι οι πιο γνωστοί και ευρέως χρησιμοποιούμενοι εξελικτικοί αλγόριθμοι. Αποτελούν ένα υποσύνολο της γενετικής αναζήτησης και χρησιμοποιούν τις έννοιες της φυσικής επιλογής, της αναπαραγωγής, και της μετάλλαξης για να εξελίξουν έναν πληθυσμό πιθανών λύσεων σε ένα πρόβλημα. Οι γενετικοί αλγόριθμοι χρησιμοποιούνται για τη βελτιστοποίηση προβλημάτων όπου η λύση δεν είναι άμεσα γνωστή και απαιτεί εξερεύνηση του χώρου λύσεων.

2.2.2 Εξελικτικός Προγραμματισμός (Evolutionary Programming - EP)

Ο εξελικτικός προγραμματισμός είναι μια παραλλαγή των γενετικών αλγορίθμων, η οποία επικεντρώνεται στην εξέλιξη λύσεων μέσω παραμέτρων που έχουν μορφή αναπαραστάσεων. Αυτή η μέθοδος χρησιμοποιεί γενικά τη διαδικασία της μετάλλαξης για να εξελίξει τους πληθυσμούς, ενώ η επιλογή και οι άλλοι μηχανισμοί είναι λιγότερο σημαντικοί σε σχέση με τους γενετικούς αλγόριθμους.

2.2.3 Εξελικτικές Στρατηγικές (Evolution Strategies - ES)

Οι εξελικτικές στρατηγικές αναπτύχθηκαν κυρίως για την επίλυση προβλημάτων βελτιστοποίησης πραγματικών αριθμών. Ουσιαστικά, αυτές οι στρατηγικές επικεντρώνονται στη χρήση εξελικτικών διαδικασιών για την αναζήτηση βέλτιστων

λύσεων μέσω παραμέτρων σχεδίασης, που μπορεί να περιλαμβάνουν την ικανότητα εξερεύνησης και εκμετάλλευσης των καλύτερων λύσεων.

2.2.4 Γενετικός Προγραμματισμός (Genetic Programming - GP)

Ο γενετικός προγραμματισμός χρησιμοποιεί τις ιδέες των γενετικών αλγορίθμων για την εξέλιξη προγραμμάτων υπολογιστών. Αντί να εξελίσσει απλώς την αναπαράσταση δεδομένων, όπως οι γενετικοί αλγόριθμοι, ο γενετικός προγραμματισμός εξελίσσει πλήρη προγράμματα ή εκφράσεις για την επίλυση προβλημάτων.

2.3 Τί Είναι οι Εξελικτικοί Αλγόριθμοι;

Οι εξελικτικοί αλγόριθμοι (EA) είναι αποτελεσματικές ευρετικές μέθοδοι αναζήτησης που βασίζονται στην αρχή της δαρβινικής εξέλιξης. Διαθέτουν ισχυρά χαρακτηριστικά στιβαρότητας και ευελιξίας, επιτρέποντας την καταγραφή συνολικών λύσεων σε πολύπλοκα προβλήματα βελτιστοποίησης. Η χρήση των EA προσφέρει υψηλή πιθανότητα εύρεσης σχεδόν βέλτιστης λύσης σε πρώιμα στάδια της διαδικασίας βελτιστοποίησης.

Οι EA δεν απαιτούν πληροφορίες για την κλίση της συνάρτησης καταλληλότητας για να προχωρήσουν, είναι εύκολοι στην παράλληλη επεξεργασία, και έχουν την ικανότητα να ξεφεύγουν από τοπικά ελάχιστα, όπου οι παραδοσιακές προσδιοριστικές μέθοδοι βελτιστοποίησης μπορεί να αποτύχουν ή να μην είναι εφαρμόσιμες. Η σύγκλιση των EA δεν επηρεάζεται από τη συνέχεια ή τη διαφοροποιησιμότητα της προς βελτιστοποίηση συνάρτησης, και μπορούν να χρησιμοποιηθούν όχι μόνο ως αντικατάσταση της παραδοσιακής βελτιστοποίησης, αλλά και για την επέκτασή της και τη βελτίωσή της. Είναι πλέον γνωστό ότι επιτυγχάνονται καλύτερες επιδόσεις με τη χρήση υβριδικών μεθόδων.

Οι EA είναι ιδιαίτερα χρήσιμοι για την επίλυση προβλημάτων βελτιστοποίησης με πολλαπλούς στόχους, ειδικά όταν οι μεταβλητές σχεδιασμού είναι ακέραιες, ασυνεχείς ή διακριτές. Έχουν σημαντικά πλεονεκτήματα έναντι των παραδοσιακών μεθόδων για την επίλυση τέτοιων προβλημάτων, καθώς μπορούν να εφαρμοστούν ταυτόχρονα σε πολλούς επιστημονικούς κλάδους. Επιπλέον, είναι κατάλληλοι για

προβλήματα με στοχαστικά χαρακτηριστικά, αβεβαιότητες ή θορυβώδη καταλληλότητα.

Όσον αφορά το υπολογιστικό κόστος, το πλεονέκτημα των ΕΑ είναι ότι αυξάνονται γραμμικά με το μέγεθος του προβλήματος, αν είναι καλά σχεδιασμένοι, ενώ άλλες μέθοδοι αυξάνονται συνήθως εκθετικά. Οι ΕΑ, ιδιαίτερα οι γενετικοί αλγόριθμοι, είναι εύκολο να παραλληλιστούν, καθώς είναι μέθοδοι αναζήτησης βασισμένες σε πληθυσμό. Παρά την απλότητά τους στη λειτουργία, οι Παράλληλοι Εξελικτικοί Αλγόριθμοι (PEAs) είναι πολύπλοκοι μη γραμμικοί αλγόριθμοι με πολλές παραμέτρους ελέγχου, οι οποίες επηρεάζουν τόσο την ποιότητα της διαδικασίας αναζήτησης όσο και την αποτελεσματικότητά τους.

2.4 Βασικές Αρχές των Εξελικτικών Αλγορίθμων

Οι πρώτες αναφορές στις ιδέες που διέπουν τους γενετικούς αλγορίθμους (GAs) βρίσκονται στις εργασίες του John Holland στις αρχές της δεκαετίας του 1960. Σύμφωνα με τον Holland, το βασικό χαρακτηριστικό των αξιόπιστων φυσικών προσαρμοστικών συστημάτων είναι η επιτυχής χρήση του ανταγωνισμού και της καινοτομίας, προκειμένου να εξασφαλιστεί η ικανότητα δυναμικής αντίδρασης σε απρόβλεπτα γεγονότα και σε μεταβαλλόμενα περιβάλλοντα. Η ιδέα αυτή βασίζεται σε απλά μοντέλα βιολογικής εξέλιξης, τα οποία μπορούν να συλλάβουν τις έννοιες της επιβίωσης των καλύτερων και της συνεχούς παραγωγής νέων απογόνων.

Αυτό το θέμα της χρήσης εξελικτικών μοντέλων για την κατανόηση των φυσικών προσαρμοστικών συστημάτων, καθώς και για το σχεδιασμό αξιόπιστων προσαρμοστικών έργων, έδωσε στο έργο του Holland μια διαφορετική εστίαση από εκείνη άλλων σύγχρονων ερευνητικών ομάδων. Ειδικότερα, οι ομάδες αυτές επικεντρώθηκαν στη χρήση εξελικτικών μοντέλων για τον σχεδιασμό τεχνικών πειραματικής βελτιστοποίησης (Rechenberg, 1965) ή για την εξέλιξη ευφυών πρακτόρων (Fogel et al., 1966).

Μέχρι τα μέσα της δεκαετίας του 1960, οι ιδέες του Holland άρχισαν να παίρνουν διάφορες υπολογιστικές μορφές, όπως αναδείχθηκαν από τους φοιτητές PhD που συνεργάζονταν μαζί του. Από την αρχή, τα συστήματα αυτά είχαν μια σαφή "γενετική" γεύση, με την έννοια ότι τα αντικείμενα που εξελίσσονταν με την πάροδο

του χρόνου αναπαρίστανται εσωτερικά ως "γονιδιώματα". Οι μηχανισμοί αναπαραγωγής και κληρονομικότητας αποτελούν απλές αφαιρέσεις των γνωστών γενετικών μηχανισμών του πληθυσμού, όπως η μετάλλαξη, η διασταύρωση και η αντιστροφή.

Σε αυτούς τους αλγορίθμους, ο χώρος αναζήτησης ενός προβλήματος αναπαρίσταται ως μια συλλογή ατόμων. Τα άτομα αυτά αναπαρίστανται με συμβολοσειρές χαρακτήρων, οι οποίες συχνά αναφέρονται ως χρωμοσώματα. Ο στόχος ενός γενετικού αλγόριθμου είναι να βρει το άτομο από τον χώρο αναζήτησης με το καλύτερο "γενετικό υλικό". Η ποιότητα κάθε ατόμου μετράται με μια συνάρτηση αξιολόγησης. Το τμήμα του χώρου αναζήτησης που εξετάζεται ονομάζεται πληθυσμός. Ο γενετικός αλγόριθμος λειτουργεί με την εξής διαδικασία (βλ. Figure 1).

BEGIN AGA

Make initial population at random.

WHILE NOT stop DO

BEGIN

Select parents from the population.

Produce children from the selected parents.

Mutate the individuals.

Extend the population adding the children to it.

Reduce the extend population.

END

Output the best individual found.

END AGA

Εικόνα 8. Ο ψευδοκώδικας του Αφηρημένου Γενετικού Αλγορίθμου (AGA)

Ένας γενετικός αλγόριθμος λειτουργεί σε έναν πεπερασμένο πληθυσμό χρωμοσωμάτων, τα οποία αναπαριστούν τις λύσεις του προβλήματος σε μορφή συμβολοσειρών bit. Ο μηχανισμός αναζήτησης του γενετικού αλγόριθμου αποτελείται από τρεις βασικές φάσεις:

1. **Αξιολόγηση της καταλληλότητας** κάθε χρωμοσώματος.
2. **Επιλογή των γονικών χρωμοσωμάτων.**
3. **Εφαρμογή των τελεστών μετάλλαξης και ανασυνδυασμού** στα γονικά χρωμοσώματα.

Τα νέα χρωμοσώματα που προκύπτουν από αυτές τις πράξεις αποτελούν την επόμενη γενιά, και η διαδικασία επαναλαμβάνεται μέχρι να μην επιτυγχάνονται πλέον βελτιώσεις στη διαδικασία αναζήτησης.

Για παράδειγμα, ας υποθέσουμε ότι μια συνάρτηση $f(x)$ πρέπει να μεγιστοποιηθεί στο σύνολο των ακεραίων αριθμών που κυμαίνονται από 0 έως 63. Για να εφαρμοστεί ένας γενετικός αλγόριθμος σε αυτό το πρόβλημα, η μεταβλητή x πρέπει να κωδικοποιηθεί ως συμβολοσειρά bit. Εδώ, επιλέγεται μια συμβολοσειρά bit μήκους 6, έτσι ώστε να μπορούν να προκύψουν ακέραιοι αριθμοί μεταξύ 0 (000000) και 63 (111111). Η καταλληλότητα κάθε χρωμοσώματος υπολογίζεται ως $f(x)$, όπου x είναι η ακεραία τιμή που κωδικοποιείται στο χρωμόσωμα.

Υποθέτοντας έναν πληθυσμό οκτώ χρωμοσωμάτων, μπορούμε να δημιουργήσουμε έναν αρχικό πληθυσμό με την τυχαία παραγωγή οκτώ διαφορετικών bit strings και την αξιολόγηση της καταλληλότητάς τους μέσω της συνάρτησης f . Για παράδειγμα, το χρωμόσωμα 1 κωδικοποιεί τον ακεραίο αριθμό 49 και η καταλληλότητά του είναι $f(49)=90$.

Χρωμόσωμα	Καταλληλότητα
1: 110001	90
2: 010101	10
3: 110101	100
4: 100101	5
5: 000011	95
6: 010011	90
7: 001100	5
8: 101010	5

Εικόνα 9: Πληθυσμός χρωμοσωμάτων

Εξετάζοντας τις ομοιότητες και τις διαφορές μεταξύ των χρωμοσωμάτων και συγκρίνοντας τις τιμές καταλληλότητάς τους, μπορούμε να υποθέσουμε ότι τα χρωμοσώματα με υψηλές τιμές καταλληλότητας έχουν δύο "1" στις δύο πρώτες

θέσεις ή δύο "1" στις δύο τελευταίες θέσεις. Αυτές οι ομοιότητες αξιοποιούνται από τη γενετική αναζήτηση μέσω της έννοιας των **σχημάτων** (υπερεπίπεδα, πρότυπα ομοιότητας). Ένα σχήμα αποτελείται από 0 και 1, όπως τα αρχικά χρωμοσώματα, αλλά με το πρόσθετο σύμβολο "μπαλαντέρ" ή "αδιαφορώ" (*), το οποίο αντιπροσωπεύει είτε το 0 είτε το 1. Τα σχήματα αντιπροσωπεύουν υποσύνολα χρωμοσωμάτων στον πληθυσμό.

Για παράδειγμα, το σχήμα 11**** περιλαμβάνει τα χρωμοσώματα 1 και 3 στον πληθυσμό που απεικονίζεται στην εικόνα 2, ενώ το σχήμα ****11 περιλαμβάνει τα χρωμοσώματα 5 και 6.

Δύο θεμελιώδη χαρακτηριστικά των σχημάτων είναι:

1. **Η σειρά:** Ο αριθμός των θέσεων με σταθερές τιμές (π.χ., το σχήμα 11**** έχει σειρά 2, το σχήμα 11000 έχει σειρά 5).
2. **Το καθοριστικό μήκος:** Η απόσταση μεταξύ της πρώτης και της τελευταίας θέσης με σταθερές τιμές (π.χ., το σχήμα 11*** έχει μήκος 1, το σχήμα 1***1 έχει μέγιστο μήκος 5).

Ένα **δομικό στοιχείο** είναι ένα σχήμα χαμηλής σειράς, μικρού μήκους ορισμού και με μέση καταλληλότητα, όπου η καταλληλότητα του σχήματος ορίζεται ως η μέση καταλληλότητα των μελών του στον πληθυσμό. Συνεπώς, ο γενετικός αλγόριθμος κινείται στο χώρο αναζήτησης συνδυάζοντας δομικά στοιχεία από δύο γονικά χρωμοσώματα για τη δημιουργία ενός νέου απογόνου.

Η βασική υπόθεση στον πυρήνα του γενετικού αλγορίθμου είναι ότι ένα καλύτερο χρωμόσωμα μπορεί να βρεθεί συνδυάζοντας τα καλύτερα χαρακτηριστικά δύο καλών χρωμοσωμάτων. Για παράδειγμα, γενετικό υλικό (δηλ. bits) θα ανταλλάσσονταν μεταξύ ενός χρωμοσώματος με δύο 1 στις δύο πρώτες θέσεις και ενός χρωμοσώματος με δύο 1 στις δύο τελευταίες θέσεις, προκειμένου να δημιουργηθεί ένας απόγονος με δύο 1 στις δύο πρώτες θέσεις και δύο 1 στις δύο τελευταίες θέσεις. Αυτός ο απόγονος θα έχει την ελπίδα να παράγει υψηλότερη τιμή $f(x)f(x)f(x)$ από τους δύο γονείς του.

Επομένως, τα δύο δομικά στοιχεία 11** και ***11 που βρίσκονται πάνω από το μέσο όρο συνδυάζονται για να δημιουργήσουν έναν απόγονο με υψηλότερη καταλληλότητα.

2.5 Ένας Απλός Γενετικός Αλγόριθμος

Με βάση τις παραπάνω αρχές, ένας απλός "καθαρός" γενετικός αλγόριθμος μπορεί να οριστεί με τα εξής βήματα:

1. **Δημιουργία του αρχικού πληθυσμού:** Δημιουργούμε έναν πληθυσμό PPP χρωμοσωμάτων (γενιά 0).

2. **Αξιολόγηση της καταλληλότητας:** Αξιολογούμε την καταλληλότητα κάθε χρωμοσώματος στον πληθυσμό.
3. **Επιλογή γονέων:** Επιλέγουμε PPP γονείς από τον τρέχοντα πληθυσμό μέσω αναλογικής επιλογής (δηλαδή, η πιθανότητα επιλογής είναι ανάλογη της καταλληλότητας).
4. **Διασταύρωση:** Επιλέγουμε τυχαία ένα ζεύγος γονέων για ζευγάρωμα και ανταλλάσσουμε συμβολοσειρές bit μέσω της διασταύρωσης ενός σημείου για να δημιουργήσουμε δύο απογόνους.
5. **Μετάλλαξη:** Επεξεργαζόμαστε κάθε απόγονο με τον τελεστή μετάλλαξης και εισάγουμε τους παραγόμενους απογόνους στον νέο πληθυσμό.
6. **Επανάληψη:** Επαναλαμβάνουμε τα βήματα 4 και 5 έως ότου επιλεγούν και ζευγαρώσουν όλοι οι γονείς (δημιουργούνται PPP απόγονοι).
7. **Αντικατάσταση του πληθυσμού:** Αντικαθιστούμε τον παλιό πληθυσμό χρωμοσωμάτων με τον νέο.
8. **Αξιολόγηση του νέου πληθυσμού:** Αξιολογούμε την καταλληλότητα κάθε χρωμοσώματος στον νέο πληθυσμό.
9. **Συνεχίζουμε ή σταματάμε:** Επιστρέφουμε στο βήμα 3 αν ο αριθμός των γενεών είναι μικρότερος από κάποιο ανώτερο όριο. Διαφορετικά, το τελικό αποτέλεσμα είναι το καλύτερο χρωμόσωμα που δημιουργήθηκε κατά τη διάρκεια της αναζήτησης.

Ο παραπάνω αλγόριθμος εισάγει αρκετές σημαντικές έννοιες, όπως η πιθανότητα επιλογής ενός χρωμοσώματος για γονεϊκότητα, ο τελεστής διασταύρωσης ενός σημείου για την ανταλλαγή συμβολοσειρών και ο τελεστής μετάλλαξης για την εισαγωγή τυχαίων διαταραχών στη διαδικασία αναζήτησης. Αυτές οι έννοιες ορίζονται με μεγαλύτερη ακρίβεια στην πράξη του γενετικού αλγορίθμου.

2.5.1 Επιλογή

Η επιλογή είναι η διαδικασία κατά την οποία επιλέγονται τα καλύτερα άτομα από τον πληθυσμό για να αναπαραχθούν και να δημιουργήσουν απογόνους. Η επιλογή πραγματοποιείται συνήθως με βάση την καταλληλότητα κάθε ατόμου. Ο πιο συνηθισμένος μηχανισμός επιλογής είναι η **ρουλέτα επιλογής**, όπου η πιθανότητα να επιλεγεί ένα άτομο είναι ανάλογη της καταλληλότητάς του.

2.5.2 Διασταύρωση (Crossover)

Η διασταύρωση (Crossover) είναι ένα από τα πιο κρίσιμα στάδια στους γενετικούς αλγορίθμους, καθώς η ποιότητα των αποτελεσμάτων του αλγορίθμου εξαρτάται σε μεγάλο βαθμό από την αποτελεσματικότητα αυτής της μεθόδου. Επομένως, η επιλογή της κατάλληλης μεθόδου διασταύρωσης είναι καθοριστική για την επιτυχία του

αλγορίθμοι. Σε αυτό το στάδιο, δύο γονείς συνδυάζονται για να παραχθεί ένας νέος απογόνος, ο οποίος μπορεί να περιέχει ακμές από τους γονείς ή να ενσωματώνει νέες ακμές που δεν υπήρχαν στους γονείς.

Η διασταύρωση μπορεί να είναι **ενός ή δύο σημείων**, με το αποτέλεσμα να είναι ένα νέο χρωμόσωμα που ενσωματώνει τα χαρακτηριστικά και των δύο γονέων. Κάθε τέτοια διασταύρωση έχει ως στόχο να συνδυάσει τα καλύτερα στοιχεία των γονικών λύσεων, με σκοπό την παραγωγή μιας καλύτερης απόφασης.

2.5.2.1 One-point Crossover

Ο τελεστής **διασταύρωσης ενός σημείου** αναφέρεται στην ανταλλαγή τμημάτων των χρωμοσωμάτων μεταξύ δύο γονικών λύσεων. Μια τυχαία θέση επιλέγεται σε όλο το μήκος των γονικών χρωμοσωμάτων, και τα τμήματα μετά την επιλεγμένη θέση ανταλλάσσονται για να δημιουργηθούν δύο απόγονοι.

Για παράδειγμα:

Γονέας 1	1 1 0 1 0 0 1
Γονέας 2	0 1 0 1 1 1 1
Απόγονος 1	1 1 0 1 1 1 1
Απόγονος 2	0 1 0 1 0 0 1

Στη διασταύρωση ενός σημείου, τα γονικά χρωμοσώματα κόβονται σε μια τυχαία θέση, και τα μέρη που βρίσκονται μετά από αυτή την θέση ανταλλάσσονται για να δημιουργηθούν νέοι απόγονοι.

2.5.2.2 Πιθανότητα Εφαρμογής της Διασταύρωσης

Η εφαρμογή του τελεστή διασταύρωσης δεν είναι πάντα υποχρεωτική. Εάν η διασταύρωση δεν εφαρμοστεί στους επιλεγμένους γονείς, αυτοί απλώς αντιγράφονται στον νέο πληθυσμό χωρίς καμία τροποποίηση. Με αυτόν τον τρόπο, τα καλά

χρωμοσώματα μπορούν να διατηρηθούν από τη μια γενιά στην επόμενη, ενισχύοντας τη σταθερότητα του πληθυσμού.

2.5.2.3 Ποσοστό Διασταύρωσης και Επιθετικότητα

Το ποσοστό διασταύρωσης επηρεάζει την "επιθετικότητα" της αναζήτησης του γενετικού αλγορίθμου. Εάν το ποσοστό διασταύρωσης είναι υψηλό, θα δημιουργηθούν περισσότεροι απόγονοι, με τον κίνδυνο να χαθούν πολλά από τα καλά χρωμοσώματα του πληθυσμού. Αντίθετα, όταν το ποσοστό διασταύρωσης είναι χαμηλότερο, τα καλά χρωμοσώματα διατηρούνται για περισσότερες γενιές, και η αναζήτηση του χώρου λύσεων γίνεται πιο συντηρητική.

Μερικές μελέτες προτείνουν ότι η καλή απόδοση ενός αλγορίθμου απαιτεί την επιλογή ενός αρκετά υψηλού ποσοστού διασταύρωσης (π.χ. 0,6), με το 60% των γονέων να υφίστανται διασταύρωση. Αυτό επιτρέπει στην αναζήτηση να προχωρήσει πιο επιθετικά, ενώ παράλληλα διατηρεί την ποιότητα των λύσεων.

2.5.3 Μετάλλαξη

Η μετάλλαξη είναι ένα κρίσιμο βήμα στον εξελικτικό αλγόριθμο, που εφαρμόζεται στα bits των απογόνων που προκύπτουν από τη διασταύρωση ενός σημείου. Ο τελεστής μετάλλαξης επηρεάζει κάθε bit με μικρή πιθανότητα (π.χ. 0.001). Όταν ενεργοποιείται, η τιμή του bit αλλάζει από 0 σε 1 ή από 1 σε 0, προσφέροντας την ευκαιρία για νέες παραλλαγές της λύσης.

Αυτό το βήμα είναι εξίσου σημαντικό, καθώς βοηθά στην αποφυγή τοπικών βέλτιστων, δηλαδή της εύρεσης λύσεων που είναι καλές μόνο για ένα συγκεκριμένο δείγμα του πληθυσμού, χωρίς να είναι απαραίτητα οι καλύτερες λύσεις για το συνολικό πρόβλημα. Χωρίς τη μετάλλαξη, ο αλγόριθμος μπορεί να "κολλήσει" σε περιοχές του χώρου αναζήτησης, μη ανακαλύπτοντας καλύτερες λύσεις.

Επιπλέον, η μετάλλαξη είναι χρήσιμη για την εισαγωγή ποικιλίας σε έναν πληθυσμό που τείνει να ομογενοποιηθεί, καθώς και για την αποκατάσταση των τιμών των bits που δεν μπορούν να ανακτηθούν μέσω της διασταύρωσης (π.χ., όταν όλα τα χρωμοσώματα στον πληθυσμό έχουν την ίδια τιμή για μια δεδομένη θέση bit).

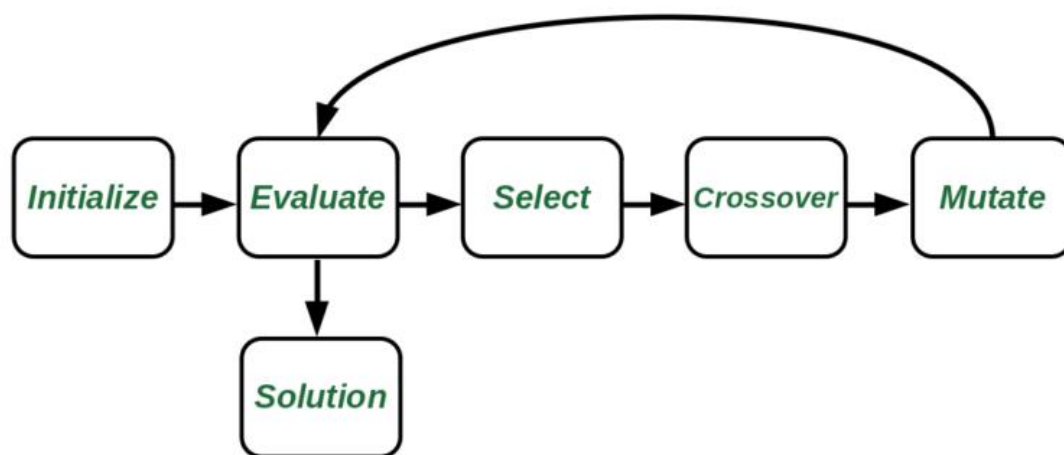
Για να διατηρηθεί ένα αποδεκτό επίπεδο ποικιλίας στον πληθυσμό, είναι καλό να αυξάνεται η πιθανότητα μετάλλαξης καθώς η αναζήτηση προχωρά. Έτσι, ο αλγόριθμος συνεχίζει να εξερευνά νέες περιοχές του χώρου λύσεων, μειώνοντας τον κίνδυνο να εγκλωβιστεί σε τοπικά βέλτιστα.

Ένας εξελικτικός αλγόριθμος θεωρείται πλήρης όταν εφαρμόζονται όλες αυτές οι λειτουργίες, βελτιστοποιώντας την αρχική λύση και παράγοντας καλύτερα αποτελέσματα για το εκάστοτε πρόβλημα.

2.5.4 Διατήρηση του Πληθυσμού

Η διατήρηση του πληθυσμού είναι η διαδικασία κατά την οποία η επόμενη γενιά αντικαθιστά την προηγούμενη ή μέρος αυτής. Κάποιες φορές, η διαδικασία αυτή περιλαμβάνει την ελίτ επιλογής, όπου τα καλύτερα άτομα του πληθυσμού παραμένουν και μεταφέρονται στην επόμενη γενιά χωρίς να υποστούν αλλαγές.

Μπορούμε να δούμε την διαδικασία που μόλις περιγράψαμε στο παρακάτω γράφημα.



Εικόνα 10. Βήματα ενός γενετικού αλγορίθμου (AGA).

2.6 Χαρακτηριστικά Γενετικών Αλγορίθμων

Οι γενετικοί αλγόριθμοι παρουσιάζουν ορισμένα χαρακτηριστικά που καθορίζουν την αναζήτηση και τη διαδικασία βελτιστοποίησης που εκτελούν. Τα βασικά χαρακτηριστικά τους μπορούν να περιγραφούν ως εξής:

1. Χειρισμός Συμβολοσειρών ή Χρωμοσωμάτων

Οι γενετικοί αλγόριθμοι χειρίζονται συμβολοσειρές ή χρωμοσώματα που κωδικοποιούν χρήσιμες πληροφορίες σχετικά με το πρόβλημα που προσπαθούν να επιλύσουν. Τα χρωμοσώματα δεν περιλαμβάνουν άμεσα την ερμηνεία των πληροφοριών αυτών — η κατανόηση του τι αντιπροσωπεύει κάθε συμβολοσειρά γίνεται μέσω της διαδικασίας αξιολόγησης. Τα δεδομένα αυτά δεν χρειάζονται αποκωδικοποίηση ή περαιτέρω ερμηνεία, καθώς η αξιολόγηση της καταλληλότητας γίνεται με βάση τη συνάρτηση καταλληλότητας.

2. Αξιολόγηση με Συναρτήσεις Καταλληλότητας

Οι γενετικοί αλγόριθμοι χρησιμοποιούν τη συνάρτηση καταλληλότητας για να καθοδηγήσουν την αναζήτηση στον χώρο λύσεων. Κάθε χρωμόσωμα αξιολογείται με βάση την ποιότητά του σύμφωνα με τη συνάρτηση καταλληλότητας. Η λειτουργία της συνάρτησης είναι να αποδίδει μια τιμή που αντιπροσωπεύει πόσο κοντά είναι το χρωμόσωμα στην επιθυμητή λύση. Σημειώνεται ότι οι γενετικοί αλγόριθμοι δεν χρησιμοποιούν καμία άλλη πληροφορία σχετικά με τη συνάρτηση καταλληλότητας ή το πεδίο εφαρμογής του προβλήματος — η αξιολόγηση αυτή είναι απολύτως απαραίτητη για να καθοδηγήσουν την αναζήτηση.

3. Πληθυσμιακή Αναζήτηση

Η αναζήτηση στον χώρο των λύσεων εκτελείται παράλληλα από έναν πληθυσμό χρωμοσωμάτων, τα οποία εξετάζουν διαφορετικά μέρη του χώρου αναζήτησης ταυτόχρονα. Αυτό επιτρέπει στον αλγόριθμο να διερευνά πιο αποτελεσματικά το πρόβλημα, αποφεύγοντας τις δυσκολίες που θα μπορούσαν να προκύψουν από μια σειριακή ή μονοδιάστατη αναζήτηση.

4. Στοχαστική Μετάβαση στον Χώρο Λύσεων

Η μετάβαση από ένα χρωμόσωμα σε ένα άλλο στο χώρο λύσεων γίνεται στοχαστικά. Η διαδικασία αυτή περιλαμβάνει τυχαία στοιχεία και αβεβαιότητα, κάτι που επιτρέπει στον αλγόριθμο να εξερευνήσει πολλές διαφορετικές περιοχές του χώρου αναζήτησης. Αυτός ο τυχαίος χαρακτήρας της μετάβασης, σε συνδυασμό με τις διαδικασίες επιλογής, διασταύρωσης και μετάλλαξης, βοηθά τον αλγόριθμο να αποφύγει την παγίδευση σε τοπικά βέλτιστα και να επιτύχει καλύτερες λύσεις συνολικά.

3 Περιγραφή του TSP

Το πρόβλημα του πλανόδιου πωλητή ή TSP, είναι ένα γνωστό αλγοριθμικό πρόβλημα στην επιστήμη των υπολογιστών. Αποτελείται από έναν πωλητή και ένα σύνολο προορισμών και αναφέρεται στην πρόκληση του προσδιορισμού της συντομότερης αλλά και αποδοτικότερης διαδρομής που πρέπει να ακολουθήσει ένας πλανόδιος πωλητής για να επισκεφθεί έναν κατάλογο συγκεκριμένων προορισμών. Ο στόχος είναι η εύρεση της συντομότερης διαδρομής από ένα σύνολο διαφορετικών διαδρομών για την ελαχιστοποίηση της συνολικής διανυόμενης απόστασης και του κόστους ταξιδιού.

Το πρόβλημα του πλανόδιου πωλητή διατυπώθηκε για πρώτη φορά το 1930 από τον Merrill M. Flood, ο οποίος έψαχνε να λύσει ένα πρόβλημα δρομολόγησης σχολικών λεωφορείων. Πρόκειται για ένα από τα πιο εντατικά μελετημένα υπολογιστικά προβλήματα βελτιστοποίησης, το οποίο χρησιμοποιείται ως σημείο αναφοράς για πολλές άλλες μεθόδους βελτιστοποίησης.

3.1 Δυσκολίες του TSP

Το TSP κατατάσσεται στα προβλήματα συνδυαστικής βελτιστοποίησης που είναι γνωστά ως "NP-πλήρη". Ταξινομείται ως "NP-δύσκολο" για δύο λόγους: Δεν υπάρχουν γρήγορες λύσεις. Η πολυπλοκότητα του υπολογισμού της βέλτιστης διαδρομής αυξάνεται όταν προσθέτετε περισσότερους προορισμούς στο TSP. Το TSP μπορεί να επιλυθεί αναλύοντας κάθε διαδρομή μετ' επιστροφής για να προσδιοριστεί η συντομότερη. Καθώς αυξάνεται ο αριθμός των προορισμών, ο αντίστοιχος αριθμός των διαδρομών αυξάνεται εκθετικά.

3.2 Προσεγγίσεις επίλυσης

Λόγω της συνδυαστικής του φύσης, η εύρεση μιας βέλτιστης λύσης στο TSP γίνεται όλο και πιο δύσκολη όσο αυξάνεται ο αριθμός των πόλεων. Έχουν αναπτυχθεί διάφορες προσεγγίσεις για την αντιμετώπιση του προβλήματος.

Οι **ακριβείς αλγόριθμοι** για το TSP εγγυώνται την εύρεση της βέλτιστης λύσης, δηλαδή της συντομότερης δυνατής διαδρομής που πρέπει να ακολουθήσει ο πλανόδιος πωλητής. Ωστόσο, αυτοί οι αλγόριθμοι είναι υπολογιστικά ακριβοί και δεν

είναι πρακτικοί για μεγάλα προβλήματα, καθώς η υπολογιστική πολυπλοκότητα αυξάνεται σημαντικά με τον αριθμό των πόλεων. Παραδείγματα αυτών των αλγορίθμων περιλαμβάνουν:

- **Μέθοδος Branch and Bound:** Αναζητά την καλύτερη λύση διαιρώντας το πρόβλημα σε μικρότερα υποπροβλήματα και εξετάζοντας τις λύσεις τους. Όταν η βελτίωση σε μια λύση δεν είναι δυνατή, το υποπρόβλημα απορρίπτεται.
- **Δυναμικός Προγραμματισμός:** Χρησιμοποιεί την αποθήκευση των λύσεων των υποπροβλημάτων (ή των ενδιάμεσων αποτελεσμάτων) για να αποφευχθεί ο επαναλαμβανόμενος υπολογισμός, μειώνοντας έτσι τον χρόνο υπολογισμού.
- **Ακέραιος Γραμμικός Προγραμματισμός:** Αυτή η μαθηματική προσέγγιση επιλύει το πρόβλημα με τη χρήση γραμμικών εξισώσεων, διασφαλίζοντας τη βέλτιστη λύση για το TSP.

Συμβιβασμός βελτιστότητας: Το TSP απαιτεί την εύρεση του καλύτερου συμβιβασμού μεταξύ υπολογιστικού χρόνου και ποιότητας λύσης. Οι ακριβείς αλγόριθμοι εγγυώνται τη βελτιστότητα, αλλά μπορεί να είναι ανέφικτοι για μεγάλες περιπτώσεις, ενώ οι ευρετικοί αλγόριθμοι θυσιάζουν τη βελτιστότητα για ταχύτερο υπολογισμό. Οι ευρετικοί αλγόριθμοι, αντί να εξετάσουν κάθε πιθανή διαδρομή, χρησιμοποιούν κανόνες ή προσεγγίσεις που επιλέγουν μια "καλή" λύση χωρίς να χρειάζεται να ελέγξουν όλες τις πιθανότητες. Παρόλο που δεν παρέχουν εγγύηση για την καλύτερη λύση, οι ευρετικοί αλγόριθμοι είναι εξαιρετικά χρήσιμοι όταν απαιτείται μια ταχεία προσέγγιση του προβλήματος.

Αυτοί οι αλγόριθμοι είναι συχνά χρήσιμοι όταν ο χρόνος υπολογισμού είναι περιορισμένος και η ακριβής λύση είναι δύσκολο να βρεθεί ή να υπολογιστεί. Οι ευρετικοί αλγόριθμοι θυσιάζουν τη βελτιστότητα υπέρ της ταχύτητας και της ευκολίας υπολογισμού, κάνοντάς τους ιδανικούς για πολύπλοκα προβλήματα όπως το TSP με μεγάλο αριθμό πόλεων.

3.3 Κατηγοριοποίηση των TSP

Τα TSP μπορούν να κατηγοριοποιηθούν σε διάφορους μαθηματικούς τύπους με βάση τα χαρακτηριστικά του προβλήματος:

- **Μετρικό TSP:** Σε αυτόν τον τύπο, οι αποστάσεις μεταξύ των πόλεων ικανοποιούν την τριγωνική ανισότητα. Αυτή είναι η πιο συνηθισμένη παραλλαγή και επιτρέπει πιο αποδοτικούς αλγορίθμους.
- **Ευκλείδειο TSP:** Εδώ, οι πόλεις αναπαρίστανται ως σημεία στον ευκλείδειο χώρο και οι αποστάσεις μεταξύ τους υπολογίζονται με βάση τις συντεταγμένες τους.
- **Ασύμμετρη TSP:** Σε αυτή την παραλλαγή, η απόσταση από μια πόλη σε μια άλλη μπορεί να μην είναι ίδια με την απόσταση προς την αντίθετη κατεύθυνση.
- **Time-Dependent TSP:** Αυτός ο τύπος ενσωματώνει χρονικούς περιορισμούς, όπου οι χρόνοι ταξιδιού μεταξύ των πόλεων εξαρτώνται από την ώρα της ημέρας ή τις συνθήκες κυκλοφορίας.

Συνολικά, το πρόβλημα του πλανόδιου πωλητή είναι ένα δύσκολο πρόβλημα βελτιστοποίησης με διάφορες προσεγγίσεις λύσεων και μαθηματικούς τύπους. Οι ερευνητές συνεχίζουν να διερευνούν νέους αλγορίθμους και τεχνικές για την αποτελεσματική επίλυσή του και την αντιμετώπιση της πολυπλοκότητάς του.

3.4 Διατυπώσεις ακέραιου γραμμικού προγραμματισμού

Μαθηματικά, το TSP μπορεί να περιγραφεί σαν πρόβλημα ακέραιου γραμμικού προγραμματισμού.

$$x_{ij} = \begin{cases} 1 & \text{the path goes from city } i \text{ to city } j \\ 0 & \text{otherwise.} \end{cases}$$

Για $i = 0, \dots, n$, έστω u_i μια τεχνητή μεταβλητή, και τέλος θεωρούμε ότι C_{ij} είναι η απόσταση από την πόλη i στην πόλη j . Τότε το TSP μπορεί να γραφτεί ως το ακόλουθο πρόβλημα ακέραιου γραμμικού προγραμματισμού:

$$\begin{aligned} \min & \sum_{i=0}^n \sum_{j \neq i, j=0}^n C_{ij} x_{ij} \\ & 0 \leq x_{ij} \leq 1 & i, j = 0, \dots, n \\ & u_i \in \mathbf{Z} & i = 0, \dots, n \\ & \sum_{i=0, i \neq j}^n x_{ij} = 1 & j = 0, \dots, n \\ & \sum_{j=0, j \neq i}^n x_{ij} = 1 & i = 0, \dots, n \\ & u_i - u_j + n x_{ij} \leq n - 1 & 1 \leq i \neq j \leq n \end{aligned}$$

4. Συνδυαστικές Προσεγγίσεις Νευρωνικών Δικτύων και Εξελικτικών Αλγορίθμων για το TSP

4.1 Παρουσίαση ερευνών

Το TSP έχει μελετηθεί εκτενώς από ερευνητές και επιστήμονες στον τομέα της υπολογιστικής επιστήμης και της βελτιστοποίησης, λόγω της συνδυαστικής του φύσης και της εφαρμογής του σε ποικίλες πραγματικές καταστάσεις, όπως για παράδειγμα η δρομολόγηση οχημάτων. Πολλές μέθοδοι έχουν προταθεί για την επίλυσή του, από ακριβείς αλγορίθμους μέχρι και ευρετικούς, για την εύρεση βέλτιστων ή σχεδόν βέλτιστων λύσεων σε αποδεκτό χρόνο. Σε αυτή την ενότητα, παρακάτω παρουσιάζονται μερικές από τις έρευνες που συνδυάζουν νευρωνικά δίκτυα και γενετικούς αλγόριθμους για την επίλυση του TSP, διερευνώντας τον τρόπο με τον οποίο οι δύο μέθοδοι συνεργάζονται για την επίτευξη βελτιστοποιημένων λύσεων με ταχύτερη και πιο αποδοτική υπολογιστική απόδοση.

4.1.1 Γενετικός αλγόριθμος για τη βελτιστοποίηση προβλημάτων πλανόδιου πωλητή με χρονικά παράθυρα (TSP-TW)

Αυτή η εργασία προτείνει τη χρήση ενός Γενετικού Αλγορίθμου βασισμένου σε Νευρωνικό Δίκτυο για την επίλυση του TSP με χρονικά παράθυρα (TSP-TW). Ο αλγόριθμος χρησιμοποιεί ένα νευρωνικό δίκτυο για την εκμάθηση της κατανομής των χρόνων ταξιδιού μεταξύ των πόλεων και τον Γενετικό Αλγόριθμο για την εύρεση της βέλτιστης λύσης, ελαχιστοποιώντας τον συνολικό χρόνο ταξιδιού.

Ο αλγόριθμος ξεκινά με τον σχηματισμό ενός αρχικού πληθυσμού που αποτελείται από πιθανούς λύτες, οι οποίοι στη συνέχεια επεξεργάζονται μέσω των τελεστών διασταύρωσης και μετάλλαξης. Τα αποτελέσματα δείχνουν ότι ο αλγόριθμος μπορεί να βρει καλές λύσεις για μεγάλα προβλήματα TSP-TW, ενώ ο αλγόριθμος είναι αποδοτικός και ευέλικτος, επιτρέποντας προσαρμογές στο χρονοδιάγραμμα με βάση τις συνθήκες της διαδρομής.

4.1.2 Υβρίδιο Hopfield Neural Network (HNN) και Γενετικού Αλγορίθμου (GA)

Μια χαρακτηριστική προσέγγιση χρησιμοποίησε ένα δίκτυο Hopfield για να παραγάγει μια αρχική σχεδόν βέλτιστη λύση, την οποία στη συνέχεια τροφοδότησε ως αρχικό πληθυσμό σε έναν GA. Με αυτόν τον τρόπο, ο GA ξεκινά την αναζήτησή του από μια πολύ καλή αφετηρία αντί για τυχαίες διαδρομές, βελτιώνοντας την αποδοτικότητα και την ταχύτητα σύγκλισης. Πειράματα σε μικρής κλίμακας εκδοχές του TSP έδειξαν ότι η υβριδική αυτή μέθοδος βρίσκει τη βέλτιστη διαδρομή ταχύτερα και με λιγότερη προσπάθεια από μεμονωμένους GA ή Hopfield, επιτυγχάνοντας καλύτερη πολυπλοκότητα χρόνου / χώρου από εναλλακτικές μεθόδους.

4.1.3 Εξέλιξη Αυτό - Οργανωμένου Χάρτη (SOM) με GA

Μια άλλη πρωτοποριακή παλαιότερη δουλειά (Jin et al., 2003) συνδύασε γενετικούς αλγορίθμους με αυτο-οργανωμένους χάρτες – ένα τύπο νευρωνικού δικτύου – για την επίλυση Ευκλείδειων TSP μεγάλου μεγέθους. Συγκεκριμένα, χρησιμοποίησαν GA για να εξελίξουν αυτόματα τις παραμέτρους εκπαίδευσης ενός SOM (Self-Organising Map), δημιουργώντας έναν “εξελιγμένο SOM” που παράγει διαδρομές υψηλής ποιότητας. Ο GA εδώ λειτουργεί ως βελτιστοποιητής του νευρωνικού δικτύου, ρυθμίζοντας τις παραμέτρους μάθησης ώστε το SOM να αποδίδει καλύτερες λύσεις. Το εξελιγμένο SOM βρήκε κοντύτερες διαδρομές με λιγότερο χρόνο εκτέλεσης συγκριτικά με συμβατικές μεθόδους, αποδεικνύοντας ότι η ενσωμάτωση GA μπορεί να ενισχύσει τις δυνατότητες των SOM σε μεγάλα Ευκλείδεια TSP.

4.1.4 Βελτιστοποίηση του TSP με χρήση ομαδοποίησης NN και GA

Αυτή η εργασία προτείνει τη χρήση Νευρωνικών Δικτύων για την ομαδοποίηση των πόλεων και των Γενετικών Αλγορίθμων για την επίλυση του TSP. Η βασική προσέγγιση περιλαμβάνει τη διαίρεση των πόλεων σε μικρότερες ομάδες και την εφαρμογή του Γενετικού Αλγορίθμου σε κάθε ομάδα ως υποπρόβλημα.

Η ομαδοποίηση μέσω των Νευρωνικών Δικτύων βοηθά στην ελαχιστοποίηση της απόστασης εντός κάθε ομάδας πόλεων, ενώ ο Γενετικός Αλγόριθμος χρησιμοποιείται για την εύρεση της βέλτιστης διαδρομής για κάθε υποπρόβλημα. Η συνδυασμένη αυτή προσέγγιση μειώνει τον υπολογιστικό χρόνο και βελτιώνει τα αποτελέσματα, προσφέροντας μια αποδοτική μέθοδο για την επίλυση του TSP.

4.1.5 Χαστικό Νευρωνικό Δίκτυο + GA σε προβλήματα διαδρομής

Μια μελέτη του 2022 για προγραμματισμό διαδρομής ρομπότ (ανάλογο ενός Euclidean TSP με εμπόδια) χρησιμοποίησε ένα παροδικό χαστικό νευρωνικό δίκτυο βασισμένο σε Hopfield, σε σύζευξη με GA, για την εύρεση σύντομων διαδρομών. Το χαστικό νευρωνικό δίκτυο (TCNN) λειτουργεί αρχικά υπολογίζοντας μια λύση διαδρομής. Αυτή η λύση κωδικοποιείται σε χρωμοσώματα και εισάγεται στον πληθυσμό του GA, αντικαθιστώντας τα λιγότερο προσαρμοσμένα άτομα. Εν συνεχεία, ο GA εκτελεί τις διαδικασίες επιλογής, διασταύρωσης και μετάλλαξης ώστε να βελτιστοποιήσει περαιτέρω τη διαδρομή. Τα αποτελέσματα έδειξαν ότι η υβριδική αυτή μέθοδος παράγει σύντομότερες διαδρομές σε σύγκριση με άλλους αλγορίθμους, βελτιώνοντας τόσο το μήκος της διαδρομής όσο και τον χρόνο εύρεσης λύσης. Η έμφαση εδώ είναι στην ταχεία εύρεση καλής λύσης από το νευρωνικό στάδιο και στη συνέχεια στη βελτίωσή της από τον εξελικτικό αλγόριθμο.

4.1.6 SOM με Εξελικτική Αναζήτηση για Multi-TSP

Σε εργασία του 2019 εξετάστηκε το πρόβλημα πολλαπλών πλανόδιων πωλητών με κοινή αφετηρία (Single-Depot mTSP) υπό κριτήριο MinMax (ελαχιστοποίηση της μεγαλύτερης διαδρομής). Οι ερευνητές ανέπτυξαν έναν αυτο-οργανωνόμενο χάρτη που αρχικοποιεί μια λύση για το mTSP, και στη συνέχεια εφάρμοσαν έναν εξελικτικό αλγόριθμο καθώς και αποικίες μυρμηγκιών (ACO) για τη βελτιστοποίηση. Οι δοκιμές έδειξαν ότι ο συνδυασμός SOM με έναν εξελικτικό αλγόριθμο (SOM-EA) υπερείχε σημαντικά έναντι του μεμονωμένου EA, βρίσκοντας λύσεις με μικρότερο μέγιστο μήκος διαδρομής σε όλους τους ελέγχους. Επιπλέον, όταν στο υβρίδιο SOM-EA προστέθηκε και μια φάση τοπικής βελτίωσης με τον αλγόριθμο 2-opt (δημιουργώντας έναν μεμετικό αλγόριθμο SOM-EA-2opt), η διαδικασία κατάφερε σε ορισμένες περιπτώσεις να βρει ακόμα και το βέλτιστο δυνατό γύρο (optimum) για τα εξεταζόμενα προβλήματα. Αυτό καταδεικνύει την αποτελεσματικότητα του συνδυασμού: το SOM παρέχει μια καλή αρχική οργάνωση των πόλεων σε διαδρομές, ενώ ο εξελικτικός αλγόριθμος βελτιστοποιεί περαιτέρω τη λύση, πετυχαίνοντας βέλτιστα αποτελέσματα σε πολύπλοκες παραλλαγές του TSP.

4.2 Συμπέρασμα από τη βιβλιογραφία

Συμπερασματικά, η βιβλιογραφία αποτυπώνει μια σαφή τάση όπου οι δύο τεχνολογίες αξιοποιούνται συμπληρωματικά, με το ένα εργαλείο να υποστηρίζει το άλλο προς έναν κοινό στόχο: την ταχύτερη εύρεση καλύτερων λύσεων στο TSP. Στις περισσότερες προσεγγίσεις, το νευρωνικό δίκτυο χρησιμοποιείται για να προσφέρει γνώση ή δομή στην αρχική λύση ή στη διαδικασία βελτίωσης (π.χ. δημιουργώντας ένα καλό αρχικό tour ή αξιολογώντας / διορθώνοντας τις λύσεις) – με αυτόν τον τρόπο “κατευθύνει” τον εξελικτικό αλγόριθμο από μια εκκίνηση κοντά στο βέλτιστο. Από την άλλη πλευρά, οι εξελικτικοί αλγόριθμοι συχνά αναλαμβάνουν τη ρύθμιση των παραμέτρων ή την περαιτέρω βελτιστοποίηση των νευρωνικών μοντέλων, ώστε αυτά να αποδώσουν καλύτερα (όπως στην εξέλιξη βαρών / αρχιτεκτονικής ενός SOM). Ο συνδυασμός αυτός εκμεταλλεύεται την ικανότητα των νευρωνικών δικτύων να εντοπίζουν γρήγορα ικανοποιητικές (αλλά προσεγγιστικές) λύσεις και την ικανότητα των εξελικτικών μεθόδων να κάνουν διεξοδική αναζήτηση σε μεγάλο χώρο λύσεων. Με βάση τη βιβλιογραφία, επικρατεί η πρακτική όπου το νευρωνικό σκέλος παίζει υποστηρικτικό ρόλο ως καθοδηγητής λύσεων, ενώ ο εξελικτικός αλγόριθμος εκτελεί τον κύριο κορμό της βελτιστοποίησης, με στόχο την επίτευξη ταχύτερης σύγκλισης προς πολύ κοντά στο βέλτιστο γύρο και την υπέρβαση των αδυναμιών που θα είχε η κάθε μέθοδος αν εφαρμοζόταν μεμονωμένα.

5. Πρακτική Επίλυση

5.1 Εισαγωγή

Στο παρόν κεφάλαιο παρουσιάζεται η πρακτική εφαρμογή της μεθοδολογίας για την επίλυση του προβλήματος του Πλανόδιου Πωλητή (TSP). Η προσέγγιση που υιοθετήθηκε συνδυάζει εξελικτικούς αλγορίθμους με τη χρήση ενός απλού νευρωνικού δικτύου για την καθοδήγηση της αρχικοποίησης του πληθυσμού. Για τη βελτίωση της αποδοτικότητας, χρησιμοποιούνται τεχνικές όπως το έξυπνο αρχικό φιλτράρισμα πιθανών διαδρομών (smart initialization), ο μηχανισμός διατήρησης της καλύτερης λύσης (elitism) σε κάθε γενιά και ένας προσαρμοστικός ρυθμός μετάλλαξης (adaptive mutation rate) που ενισχύει τη διερεύνηση του χώρου λύσεων όταν δεν παρατηρείται πρόοδος.

Για τη συγκριτική αξιολόγηση των αποτελεσμάτων, ενσωματώθηκαν και μαθηματικές μέθοδοι επίλυσης, όπως Brute-force (για ≤ 10 πόλεις) και MST + DFS (Minimum Spanning Tree με αναζήτηση βάθους) για χρήση ως lower-bound συγκριτικό εργαλείο.

Η υλοποίηση της μεθοδολογίας συνοδεύεται από μια διαδραστική εφαρμογή με γραφικό περιβάλλον (GUI), μέσω της οποίας ο χρήστης μπορεί να επιλέξει το πλήθος των πόλεων, να παρακολουθεί τη βελτίωση των λύσεων σε πραγματικό χρόνο και να βλέπει την τελική βέλτιστη διαδρομή τόσο γραφικά όσο και πάνω σε χάρτη.

5.2 Περιγραφή Μεθοδολογίας και Αλγορίθμου

5.2.1 Δεδομένα

Για την υλοποίηση και αξιολόγηση της προτεινόμενης μεθοδολογίας, χρησιμοποιούνται ως είσοδος σύνολα πόλεων, κάθε μία από τις οποίες αναπαρίσταται μέσω ενός μοναδικού ζεύγους καρτεσιανών συντεταγμένων (x, y). Οι συντεταγμένες αυτές συλλέχθηκαν μέσω online εργαλείων γεωγραφικού εντοπισμού και αντικατοπτρίζουν πραγματικές ευρωπαϊκές πόλεις, με ακρίβεια που βασίζεται σε πραγματικά δεδομένα. Το πρόβλημα λύνεται για τέσσερα set πόλεων (4 πόλεις, 10 πόλεις, 20 πόλεις και 50 πόλεις) προκειμένου να δούμε πως φέρεται σε διαφορετικά μεγέθη του προβλήματος. Η χρήση ρεαλιστικών γεωγραφικών συντεταγμένων προσδίδει πρακτικό χαρακτήρα στο πρόβλημα, καθώς προσομοιώνει σενάρια δρομολόγησης με γεωγραφική σημασία.

Η θεμελιώδης πληροφορία που απαιτείται για την επίλυση του προβλήματος είναι οι μεταξύ των πόλεων αποστάσεις. Οι αποστάσεις μεταξύ των πόλεων υπολογίζονται με τον τύπο Haversine, ο οποίος λαμβάνει υπόψη την καμπυλότητα της Γης και παρέχει ακρίβεια σε χιλιόμετρα.

Ο πίνακας αποστάσεων (distance matrix) που προκύπτει αποτελεί το βασικό δομικό στοιχείο πάνω στο οποίο στηρίζονται οι κρίσιμες λειτουργίες του αλγορίθμου: η αξιολόγηση της ποιότητας κάθε διαδρομής (fitness), η κατασκευή νέων λύσεων μέσω διασταύρωσης, καθώς και η μετάλλαξη και επιδιόρθωση των διαδρομών. Κατά τη διάρκεια της αναπαραγωγής, κάθε νέα διαδρομή αξιολογείται μέσω του συνολικού

μήκους της, το οποίο υπολογίζεται ως το άθροισμα όλων των επιμέρους αποστάσεων ανάμεσα σε διαδοχικές πόλεις του μονοπατιού.

Ομοίως, κατά την οπτικοποίηση των αποτελεσμάτων, οι συντεταγμένες των πόλεων χρησιμοποιούνται για τη γραφική απεικόνιση της βέλτιστης διαδρομής. Η τελική λύση προβάλλεται τόσο σε επίπεδο γράφου (μέσω matplotlib), όσο και πάνω σε δυναμικό χάρτη HTML, διευκολύνοντας την κατανόηση και την παρουσίαση των αποτελεσμάτων.

```
city_coordinates_10 = np.array([
    [48.8566, 2.3522], # Paris
    [52.5200, 13.4050], # Berlin
    [41.9028, 12.4964], # Rome
    [40.4168, -3.7038], # Madrid
    [51.5074, -0.1278], # London
    [37.9838, 23.7275], # Athens
    [48.8566, 2.3522], # Vienna
    [52.3676, 4.9041], # Amsterdam
    [59.3293, 18.0686], # Stockholm
    [38.7223, -9.1393] # Lisbon
])
```

Παράρτημα 1. Παράδειγμα συνόλου πόλεων (10 πόλεις)

Σχετικά με το μέγεθος πληθυσμού, και αριθμών γενεών που ορίστηκε, στις περιπτώσεις με λίγες πόλεις (4, 10 και 20), επιλέχθηκε μεγάλος αριθμός γενεών (π.χ. 500) παρότι το πρόβλημα είναι μικρής κλίμακας, με σκοπό να παρατηρηθεί η σταθερότητα της λύσης και να αποδειχθεί ότι όταν εντοπιστεί το βέλτιστο αποτέλεσμα, αυτό δεν αλλάζει στις επόμενες γενιές. Το μέγεθος πληθυσμού και για τις τρεις περιπτώσεις είναι 1000. Ο αρχικός ρυθμός μετάλλαξης επιλέγεται στο 0.1, προκειμένου να επιτρέψει ικανοποιητική διερεύνηση του χώρου λύσεων στα αρχικά στάδια της εξέλιξης, ενώ σταδιακά μειώνεται μέσω του προσαρμοστικού μηχανισμού (adaptive mutation decay), σύμφωνα με τη στρατηγική που περιγράφεται στην ενότητα του Εξελικτικού Αλγορίθμου.

Στην περίπτωση των 50 πόλεων, παρατηρήθηκε ότι οι αρχικές τιμές παραμέτρων δεν οδηγούσαν σε ικανοποιητικά αποτελέσματα. Για τον λόγο αυτό, χρησιμοποιήθηκαν

πολύ αυξημένες τιμές: πληθυσμός 10.000 ατόμων, 1000 γενιές και αρχικό ποσοστό μετάλλαξης 0.25, ώστε να ενισχυθεί η εξερεύνηση του χώρου λύσεων και να αυξηθεί η πιθανότητα εύρεσης καλής διαδρομής. Η προσαρμογή αυτή ήταν αναγκαία λόγω της εκθετικής αύξησης της πολυπλοκότητας με την προσθήκη περισσότερων πόλεων στο πρόβλημα.

5.2.2 Νευρωνικό Δίκτυο

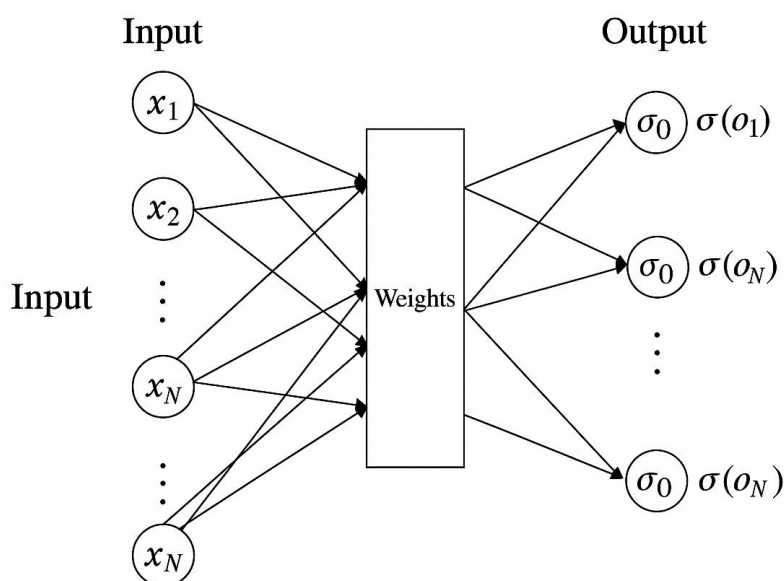
Στο πλαίσιο της προτεινόμενης μεθοδολογίας για την επίλυση του προβλήματος του Πλανόδιου Πωλητή, χρησιμοποιείται ένα απλό νευρωνικό δίκτυο, το οποίο συμβάλλει στην αρχικοποίηση του πληθυσμού πριν από την έναρξη της εξελικτικής διαδικασίας. Αν και το δίκτυο δεν υφίσταται εκπαίδευση, ενσωματώνεται στρατηγικά ώστε να εισάγει μία μορφή «καθοδηγούμενης τυχαιότητας» στην παραγωγή των αρχικών διαδρομών, ενισχύοντας έτσι την ποικιλομορφία και την ποιότητα των λύσεων ήδη από τα πρώτα στάδια της αναζήτησης.

Η αρχιτεκτονική του δικτύου είναι ιδιαίτερα απλοποιημένη και αποτελείται από μία μόνο στρώση, χωρίς ενδιάμεσα κρυφά επίπεδα (hidden layers). Η επιλογή αυτή έγινε με γνώμονα την επίτευξη υψηλής ταχύτητας και λειτουργικής απλότητας, καθώς ο ρόλος του νευρωνικού δεν είναι να πραγματοποιήσει πολύπλοκες προβλέψεις, αλλά να παρέχει αρχικές εκτιμήσεις για τη σχετική σημασία κάθε πόλης ως μέρος μίας διαδρομής.

Το δίκτυο δέχεται ως είσοδο έναν πίνακα διαστάσεων $1 \times N$, όπου N είναι το πλήθος των πόλεων, και κάθε στοιχείο του αντιστοιχεί σε μία μοναδική πόλη του προβλήματος. Η επεξεργασία της εισόδου πραγματοποιείται μέσω ενός γραμμικού μετασχηματισμού, ο οποίος περιλαμβάνει έναν πίνακα βαρών (weights) διαστάσεων $N \times N$ που αρχικοποιείται τυχαία με χρήση κανονικής κατανομής, καθώς και έναν διανυσματικό όρο μεροληψίας (bias) διαστάσεων $1 \times N$ που αρχικοποιείται με μηδενικές τιμές. Η έξοδος του γραμμικού συνδυασμού διέρχεται από τη συνάρτηση ενεργοποίησης softmax, η οποία μετατρέπει τις τιμές σε κανονικοποιημένες πιθανότητες. Με αυτό τον τρόπο, η έξοδος του δικτύου μπορεί να ερμηνευθεί ως πίνακας πιθανοτήτων επιλογής των επόμενων πόλεων σε μία διαδρομή.

Κατά την αρχικοποίηση του πληθυσμού, δημιουργούνται τυχαίες διαδρομές, οι οποίες στη συνέχεια τροποποιούνται με βάση τις προβλέψεις του νευρωνικού δικτύου. Πιο συγκεκριμένα, η έξοδος της softmax χρησιμοποιείται για να αναδιαταχθούν οι πόλεις με βάση την εκτιμώμενη «καταλληλότητα» κάθε θέσης. Το αποτέλεσμα είναι ένα νέο μονοπάτι που ενσωματώνει, με στοχαστικό τρόπο, την επιρροή των βαρών του δικτύου. Σε περιπτώσεις όπου η παραγόμενη διαδρομή περιέχει διπλότυπες ή μη έγκυρες τιμές, εφαρμόζεται μηχανισμός επιδιόρθωσης (repair mechanism) ώστε να διασφαλίζεται η εγκυρότητα κάθε υποψηφίου λύσης.

Αξίζει να σημειωθεί ότι το νευρωνικό αυτό δίκτυο δεν ενημερώνεται κατά τη διάρκεια των γενεών, αλλά λειτουργεί αποκλειστικά στην αρχή της εξελικτικής διαδικασίας. Παρ' όλα αυτά, προσφέρει σημαντικό πλεονέκτημα, καθώς βοηθά στην ταχύτερη σύγκλιση του αλγορίθμου, δίνοντας εξαρχής λύσεις που ενδέχεται να βρίσκονται πιο κοντά στο βέλτιστο.



Εικόνα 11. Μια αρχιτεκτονική νευρωνικού δικτύου ενός στρώματος που χρησιμοποιείται για την αρχικοποίηση του πληθυσμού για το πρόβλημα του TSP. X : Το διάνυσμα εισόδου, που αναπαριστά μία τυχαία διαδρομή πόλεων. Είναι διαστάσεων $1 \times N$, όπου N το πλήθος των πόλεων. Κάθε στοιχείο αντιστοιχεί σε έναν ακέραιο αριθμό που δηλώνει τη θέση μιας πόλης στη διαδρομή. W : Ο πίνακας βαρών του νευρωνικού δικτύου, διαστάσεων $N \times N$. Τα βάρη αυτά καθορίζουν τη σημασία κάθε πόλης ως προς τη θέση της στη διαδρομή. b : Ο όρος μεροληψίας (bias), διαστάσεων $1 \times N$, που προστίθεται γραμμικά στο αποτέλεσμα του πολλαπλασιασμού $X \cdot W$. σ (softmax): Η συνάρτηση ενεργοποίησης που μετατρέπει τα γραμμικά αποτελέσματα σε κανονικοποιημένες πιθανότητες. Μετά το softmax, οι τιμές της εξόδου βρίσκονται στο διάστημα $(0, 1)$ και αθροίζονται σε 1.

Πίνακας 1. Παράδειγμα λειτουργίας του Νευρωνικού Δικτύου

Αρχικό Τυχαίο Tour	Πίνακας πιθανοτήτων του Νευρωνικού Δικτύου	Αναδιαμορφωμένο tour
[2 1 5 8 3 9 0 4 6 7]	[[1.91785962e-09 6.63433036e-20 8.35382622e-02 5.03718743e-02 6.68929676e-15 3.32318066e-03 5.56938281e-11 5.02909767e-12 5.98270922e-14 8.62766681e-01]]	[9 2 3 5 0 6 7 8 4 1]

5.2.3 Εξελικτικός Αλγόριθμος

Η επίλυση του προβλήματος του Πλανόδιου Πωλητή μέσω εξελικτικών τεχνικών εντάσσεται στο ευρύτερο πλαίσιο της Υπολογιστικής Νοημοσύνης, η οποία αντλεί έμπνευση από φυσικά φαινόμενα για τη δημιουργία ευέλικτων αλγορίθμων βελτιστοποίησης. Στην παρούσα εργασία, υλοποιήθηκε ένας ειδικός εξελικτικός αλγόριθμος που συνδυάζει βασικές αρχές φυσικής επιλογής με σύγχρονες τεχνικές καθοδήγησης και ελέγχου τυχαιότητας, επιδιώκοντας την προσέγγιση βέλτιστων ή κοντά στο βέλτιστο λύσεων για το TSP, ακόμη και σε περιπτώσεις αυξημένης πολυπλοκότητας.

Η διαδικασία εκκινεί με την παραγωγή ενός αρχικού πληθυσμού υποψήφιων λύσεων, καθεμία από τις οποίες αντιστοιχεί σε μία πλήρη και έγκυρη διαδρομή που επισκέπτεται όλες τις πόλεις μία και μόνο φορά. Η αρχικοποίηση αυτή δεν βασίζεται αποκλειστικά σε τυχαίες αναδιατάξεις, αλλά ενισχύεται με τη χρήση ενός απλού νευρωνικού δικτύου, το οποίο εισάγει μια μορφή καθοδηγούμενης τυχαιότητας στον τρόπο δημιουργίας των αρχικών διαδρομών. Έτσι, ήδη από την πρώτη γενιά, ο πληθυσμός διαθέτει αυξημένη ποικιλομορφία και υψηλότερο μέσο επίπεδο καταλληλότητας (fitness).

Σε κάθε γενιά, αξιολογείται η ποιότητα όλων των διαδρομών μέσω της συνάρτησης καταλληλότητας, η οποία ορίζεται ως το αντίστροφο του συνολικού μήκους της διαδρομής. Οι συντομότερες διαδρομές, δηλαδή, έχουν μεγαλύτερη τιμή fitness και επομένως υψηλότερη πιθανότητα να επιλεγούν για αναπαραγωγή. Η επιλογή

πραγματοποιείται με τη μέθοδο της ρουλέτας (fitness-proportional selection), κατά την οποία κάθε λύση έχει πιθανότητα επιλογής ανάλογη με τη σχετική της επίδοση. Με αυτόν τον τρόπο, ενθαρρύνεται η εκμετάλλευση των καλών λύσεων χωρίς όμως να αποκλείονται υποσχόμενα άτομα με χαμηλότερο fitness.

Οι επιλεγμένες λύσεις υφίστανται διασταύρωση, χρησιμοποιώντας την τεχνική Ordered Crossover (OX1), η οποία είναι ιδανική για προβλήματα όπου η σειρά των στοιχείων παίζει κρίσιμο ρόλο. Συγκεκριμένα, επιλέγεται ένα συνεχόμενο υποσύνολο πόλεων από τον πρώτο γονέα και τοποθετείται απευθείας στον απόγονο στην ίδια θέση. Οι υπόλοιπες θέσεις συμπληρώνονται με τις πόλεις του δεύτερου γονέα, με τη σειρά εμφάνισής τους και παραλείποντας τις ήδη τοποθετημένες. Η μέθοδος αυτή διασφαλίζει ότι η τελική διαδρομή είναι έγκυρη και επιτρέπει τη μεταβίβαση κρίσιμων τοπικών αλληλουχιών πόλεων (building blocks), ενισχύοντας τη γενετική ποικιλία και τη διατήρηση καλών επιμέρους δομών.

Πίνακας II. Παράδειγμα Ordered Crossover (OX1)

Parent 1	Parent 2	Child 1	Child 2
[2 9 5 3 4 7 6 8 0 1]	[9 2 3 5 7 4 0 8 6 1]	[2, 9, 5, 3, 4, 7, 0, 8, 6, 1]	[9, 2, 3, 5, 7, 4, 6, 8, 0, 1]

Για την ενίσχυση της ποικιλομορφίας και την αποτροπή πρόωρης σύγκλισης, εφαρμόζεται μετάλλαξη με τη μορφή ανταλλαγής δύο τυχαίων πόλεων (swap mutation). Ο ρυθμός μετάλλαξης δεν παραμένει σταθερός, αλλά ακολουθεί μια στρατηγική προσαρμογής με εκθετική φθορά (adaptive mutation with decay). Στις πρώτες γενιές ο ρυθμός είναι υψηλότερος, ενισχύοντας την ευρύτερη διερεύνηση του χώρου λύσεων. Καθώς οι γενιές προχωρούν, ο ρυθμός μειώνεται σταδιακά ώστε να επιτραπεί η εστίαση στην εκμετάλλευση των καλύτερων περιοχών. Εάν για σειρά γενεών δεν παρατηρείται ουσιαστική βελτίωση, ο ρυθμός αυξάνεται προσωρινά, ώστε να ξεπεραστεί η στασιμότητα και να διερευνηθούν νέες περιοχές του χώρου αναζήτησης.

Πίνακας III. Παράδειγμα Swap Mutation

Offspring before Mutation	Mutation Indices	Offspring after Mutation
---------------------------	------------------	--------------------------

[1, 2, 5, 3, 9, 4, 0, 7, 6, 8]	[9 5]	[1, 2, 5, 3, 9, 4, 0, 7, 6, 8]
[3, 7, 4, 8, 2, 0, 6, 5, 1, 9]	[1 8]	[3, 7, 4, 8, 2, 0, 6, 5, 1, 9]
[3, 7, 4, 0, 5, 1, 6, 8, 9, 2]		[3, 7, 4, 0, 5, 2, 6, 8, 9, 1]
[0, 1, 8, 9, 2, 5, 3, 6, 7, 4]		[0, 1, 8, 9, 2, 5, 3, 6, 7, 4]
[2, 5, 3, 7, 0, 1, 6, 8, 9, 4]		[2, 9, 3, 7, 0, 1, 6, 8, 5, 4]
[9, 3, 7, 4, 0, 1, 6, 8, 2, 5]		[9, 3, 7, 4, 0, 1, 6, 8, 2, 5]
[8, 2, 5, 3, 9, 4, 0, 7, 1, 6]		[8, 2, 5, 3, 9, 4, 0, 7, 1, 6]
[8, 5, 7, 4, 1, 2, 3, 9, 0, 6]		[8, 5, 7, 4, 1, 2, 3, 9, 0, 6]
[9, 2, 5, 3, 7, 4, 0, 1, 6, 8]		[9, 2, 5, 3, 7, 4, 0, 1, 6, 8]
[2, 3, 7, 0, 1, 5, 9, 8, 4, 6]		[2, 3, 7, 0, 1, 5, 9, 8, 4, 6]
[2, 0, 3, 7, 8, 4, 6, 1, 5, 9]		[2, 0, 3, 7, 8, 4, 6, 1, 5, 9]

Επεξήγηση Πίνακα – Παράδειγμα Swap Mutation

Ο παραπάνω πίνακας απεικονίζει παραδείγματα εφαρμογής της τεχνικής μετάλλαξης τύπου "swap mutation", η οποία χρησιμοποιείται στο πλαίσιο του εξελικτικού αλγορίθμου για τη διατήρηση της ποικιλομορφίας του πληθυσμού και την αποφυγή πρόωρης σύγκλισης. Στη στήλη Offspring before Mutation εμφανίζονται οι διαδρομές (χρωμοσώματα) πριν την εφαρμογή της μετάλλαξης. Κάθε διαδρομή αποτελεί μία έγκυρη ακολουθία πόλεων προς επίσκεψη. Η στήλη Mutation Indices παρουσιάζει τα δύο σημεία (θέσεις του πίνακα) που επιλέχθηκαν τυχαία για ανταλλαγή. Η επιλογή γίνεται με ομοιόμορφη πιθανότητα και εφαρμόζεται μόνο με πιθανότητα ίση με τον εκάστοτε ρυθμό μετάλλαξης. Η τελευταία στήλη Offspring after Mutation δείχνει το αποτέλεσμα της ανταλλαγής. Τα δύο στοιχεία στις επιλεγμένες θέσεις έχουν αλλάξει θέση, ενώ η υπόλοιπη διαδρομή παραμένει αναλλοίωτη. Το αποτέλεσμα είναι μια νέα, έγκυρη διαδρομή, η οποία διαφέρει ελαφρώς από την αρχική, εισάγοντας στο σύστημα μια μικρή στοχευμένη διαταραχή που μπορεί να οδηγήσει σε καλύτερες λύσεις στις επόμενες γενιές. Η swap mutation είναι ιδανική για προβλήματα όπου η σειρά των στοιχείων έχει σημασία, όπως το TSP, καθώς δεν παραβιάζει τη μοναδικότητα των πόλεων στη διαδρομή και διατηρεί τη φύση του χρωμοσώματος.

Μια ακόμη σημαντική τεχνική που εφαρμόζεται είναι ο ελιτισμός (elitism). Ένα ποσοστό των καλύτερων λύσεων κάθε γενιάς διατηρείται αυτούσιο στον επόμενο πληθυσμό, παρακάμπτοντας τους μηχανισμούς διασταύρωσης και μετάλλαξης. Αυτό διασφαλίζει ότι οι κορυφαίες λύσεις δεν θα χαθούν κατά την εξελικτική διαδικασία και ότι η συνολική απόδοση του πληθυσμού δεν θα υποβαθμιστεί σε επόμενες γενιές.

Ο ελιτισμός είναι ιδιαίτερα σημαντικός για προβλήματα όπως το TSP, όπου η ποιότητα μπορεί να βελτιωθεί με μικρές, στοχευμένες τροποποιήσεις.

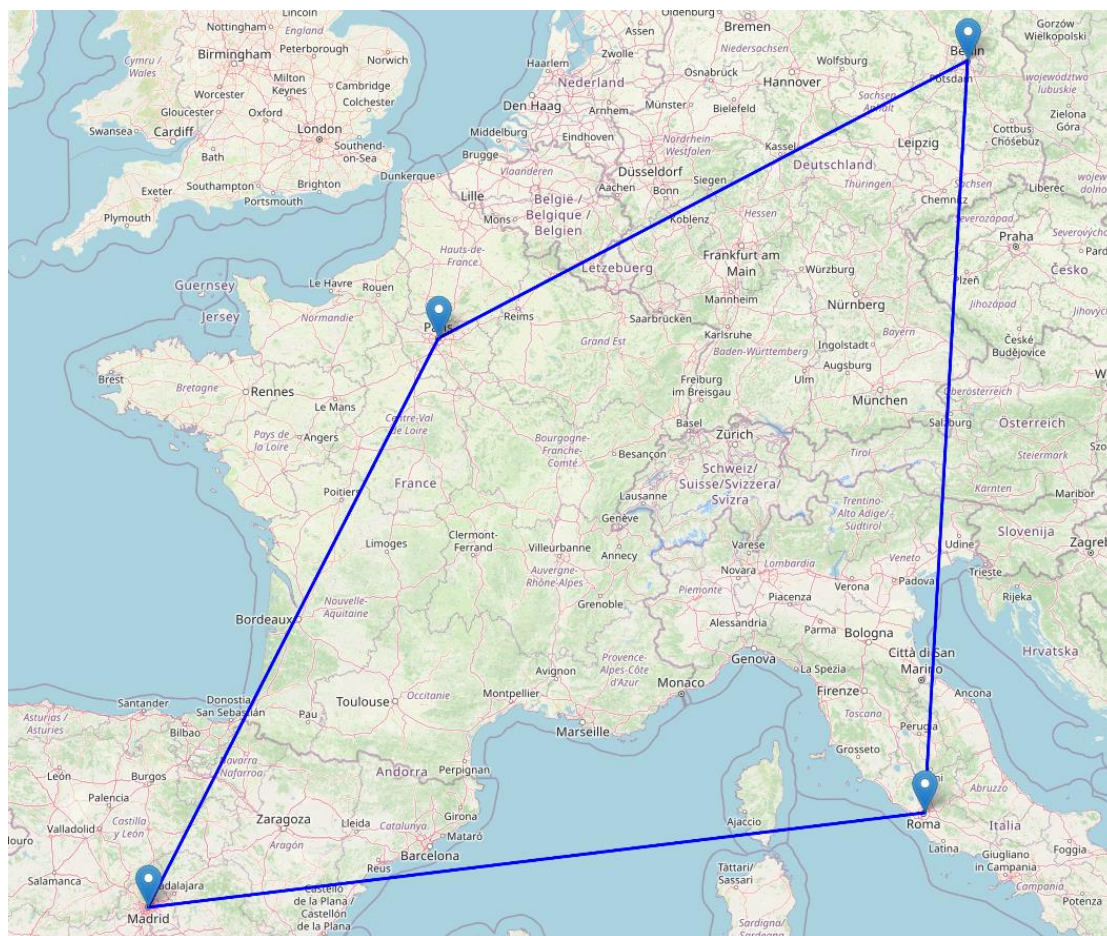
Η υλοποίηση ενσωματώνει επίσης μηχανισμό επιδιόρθωσης (repair mechanism), ο οποίος είναι απαραίτητος για τη διατήρηση της εγκυρότητας των διαδρομών. Κατά τη διασταύρωση ή μετάλλαξη, είναι πιθανό να προκύψουν μη έγκυρες διαδρομές, όπως αυτές που περιλαμβάνουν διπλότυπες ή ελλειπείς πόλεις. Στην περίπτωση αυτή, η διαδρομή είτε αναδομείται με νέα permutation είτε απορρίπτεται και αντικαθίσταται. Αυτός ο έλεγχος είναι κρίσιμος, καθώς επιτρέπει στον εξελικτικό αλγόριθμο να λειτουργεί σταθερά χωρίς αστοχίες.

Η διαδικασία συνεχίζεται για προκαθορισμένο αριθμό γενεών. Σε κάθε γενιά, καταγράφεται η καλύτερη διαδρομή και αποθηκεύεται τόσο σε μορφή οπτικοποίησης (εικόνα ή HTML χάρτης) όσο και σε αναλυτικά αρχεία αποτελεσμάτων. Η τελική λύση που επιστρέφεται από τον αλγόριθμο είναι η βέλτιστη διαδρομή που εντοπίστηκε κατά την εξέλιξη, συνοδευόμενη από τη συνολική απόστασή της, τη σειρά επίσκεψης των πόλεων και την πορεία της βελτιστοποίησης διαχρονικά.

Η συγκεκριμένη προσέγγιση συνδυάζει απλότητα και αποτελεσματικότητα, και αν και δεν εγγυάται την εύρεση της απόλυτα βέλτιστης λύσης, αποδεικνύεται ιδιαίτερα ικανή στην προσέγγιση λύσεων υψηλής ποιότητας σε λογικούς χρόνους, ακόμη και για προβλήματα αυξημένης πολυπλοκότητας.

5.2.4 Ανάλυση Αποτελεσμάτων

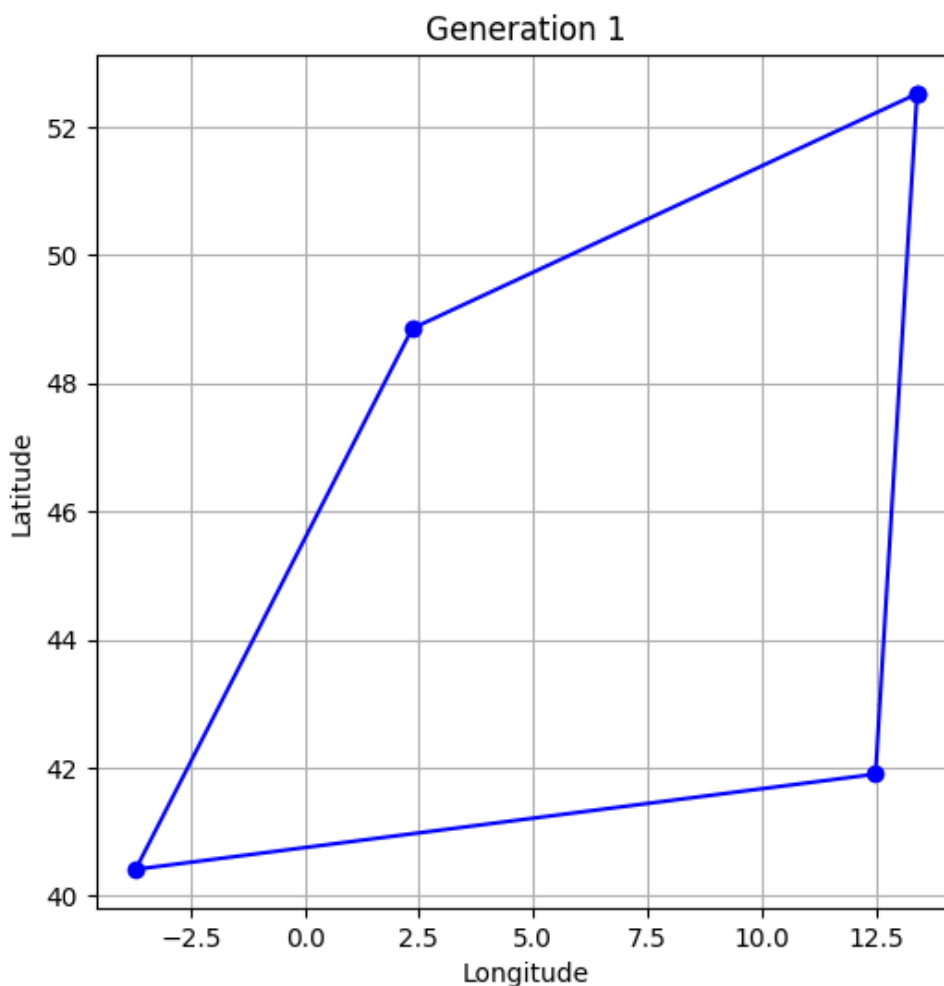
5.2.4.1 Σετ Τεσσάρων Πόλεων



Εικόνα 12. Αποτέλεσμα καλύτερης διαδρομής (best tour) που βρέθηκε από τον αλγόριθμο, πάνω σε διαδραστικό χάρτη.

Η παραπάνω εικόνα παρουσιάζει την τελική βέλτιστη διαδρομή που προέκυψε για το πρόβλημα του TSP με τέσσερις πόλεις. Η διαδρομή που εντοπίστηκε ως βέλτιστη ξεκινάει από Βερολίνο, πάει Ρώμη, στη συνέχεια Μαδρίτη, έπειτα Παρίσι και επιστρέφει στο Βερολίνο, σχηματίζοντας έναν πλήρη κύκλο. Παρατηρείται ότι η διαδρομή είναι οπτικά συμμετρική και «συμπαγής», χωρίς περιττές αποκλίσεις ή περίπλοκες επιστροφές. Το αποτέλεσμα αυτό έχει ιδιαίτερη σημασία, καθώς επιβεβαιώνει τη σωστή λειτουργία του αλγορίθμου σε μικρά σετ πόλεων, όπου το αναμενόμενο αποτέλεσμα μπορεί να υπολογιστεί με ακρίβεια και να επαληθευτεί με βεβαιότητα.

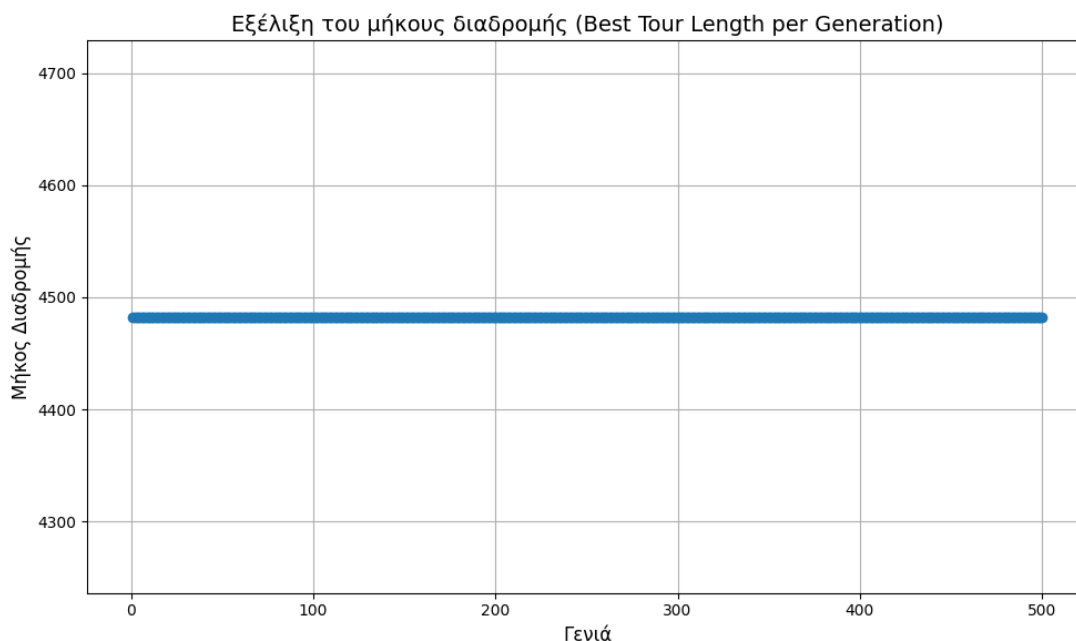
Το συνολικό μήκος της διαδρομής ανέρχεται σε 4482,6 χιλιόμετρα, όπως προέκυψε από την εφαρμογή της ευκλείδειας απόστασης επί των γεωγραφικών συντεταγμένων (latitude/longitude) των πόλεων.



Εικόνα 13. Απεικόνιση εξέλιξης αλγορίθμου ανά γενιά.

Η παραπάνω κινούμενη εικόνα (GIF) απεικονίζει δυναμικά την εξελικτική πορεία του αλγορίθμου. Σε κάθε καρτέ της κινούμενης εικόνας αντιστοιχεί μία γενιά και παρουσιάζεται η καλύτερη διαδρομή που έχει εντοπιστεί μέχρι εκείνο το σημείο της εξέλιξης. Στην συγκεκριμένη περίπτωση βλέπουμε πως η βέλτιστη διαδρομή εντοπίστηκε ήδη από την πρώτη γενιά, και διατηρήθηκε καθ' όλη τη διάρκεια των 500 γενεών χωρίς να υπάρξει βελτίωση. Το γεγονός αυτό δείχνει την ικανότητα του αλγορίθμου να εντοπίζει βέλτιστες λύσεις ήδη από το αρχικό στάδιο, ιδίως σε μικρά σετ πόλεων.

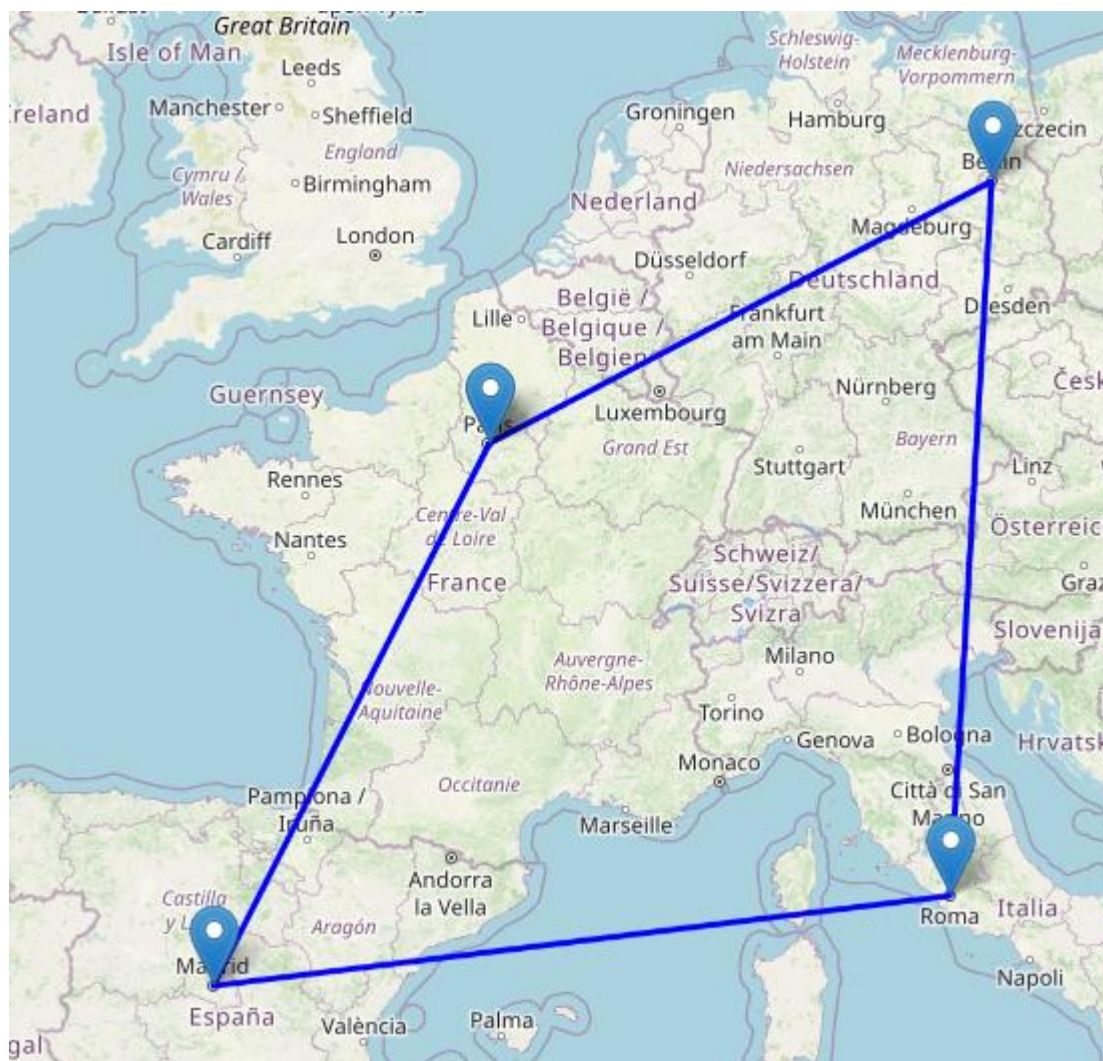
Η σταθερότητα του αποτελέσματος από την αρχή, σε συνδυασμό με την πλήρη απουσία αστοχιών ή λανθασμένων διαδρομών κατά την εξέλιξη, επιβεβαιώνει την ορθή λειτουργία του μοντέλου και τη αποδοτικότητα των τεχνικών που χρησιμοποιούνται — αρχικοποίηση με νευρωνικό δίκτυο, ελιτισμός, προσαρμοστική μετάλλαξη και επιδιόρθωση διαδρομών.



Εικόνα 14. Εξέλιξη μήκους διαδρομής

Το παραπάνω γράφημα απεικονίζει την εξέλιξη του μήκους της βέλτιστης διαδρομής ανά γενιά. Στον οριζόντιο άξονα απεικονίζονται οι γενιές (1 έως 500), ενώ στον κατακόρυφο άξονα εμφανίζεται το συνολικό μήκος της διαδρομής (σε χιλιόμετρα) που θεωρείται ως η καλύτερη μέχρι εκείνη τη στιγμή.

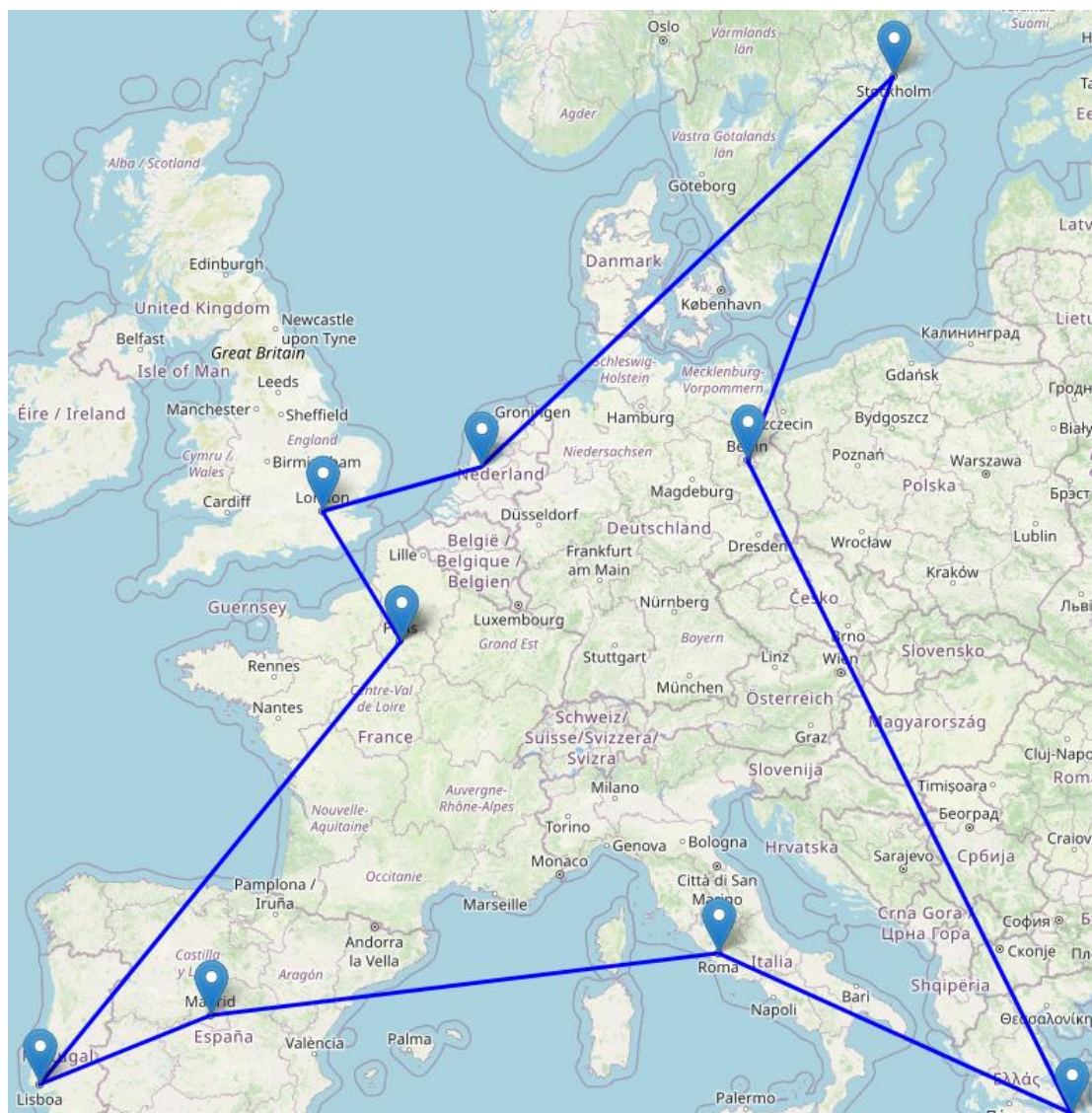
Εδώ η γραμμή είναι απόλυτα οριζόντια και σταθερή όπως είδαμε και στο παραπάνω γράφημα. Η συμπεριφορά αυτή είναι αναμενόμενη για τόσο μικρό πλήθος πόλεων, καθώς ο αριθμός πιθανών διαδρομών είναι εξαιρετικά περιορισμένος και η χρήση του νευρωνικού δικτύου κατά την αρχικοποίηση αυξάνει τις πιθανότητες εύρεσης της βέλτιστης λύσης από νωρίς.



Εικόνα 15. Αποτέλεσμα καλύτερης διαδρομής (best tour) που βρέθηκε με τη μέθοδο brute-force, πάνω σε διαδραστικό χάρτη.

Η εγκυρότητα της παραγόμενης λύσης επαληθεύτηκε μέσω της μεθόδου brute-force, η οποία είναι εφαρμόσιμη σε τόσο μικρά προβλήματα (4 πόλεις). Η εξαντλητική αξιολόγηση όλων των πιθανών διαδρομών ($4! = 24$ διαδρομές) επιβεβαίωσε ότι η διαδρομή που εντοπίστηκε από τον αλγόριθμο ήταν η βέλτιστη, με συνολικό μήκος 4482.6156 km, ίδιο ακριβώς με αυτό που εντοπίστηκε μέσω brute-force.

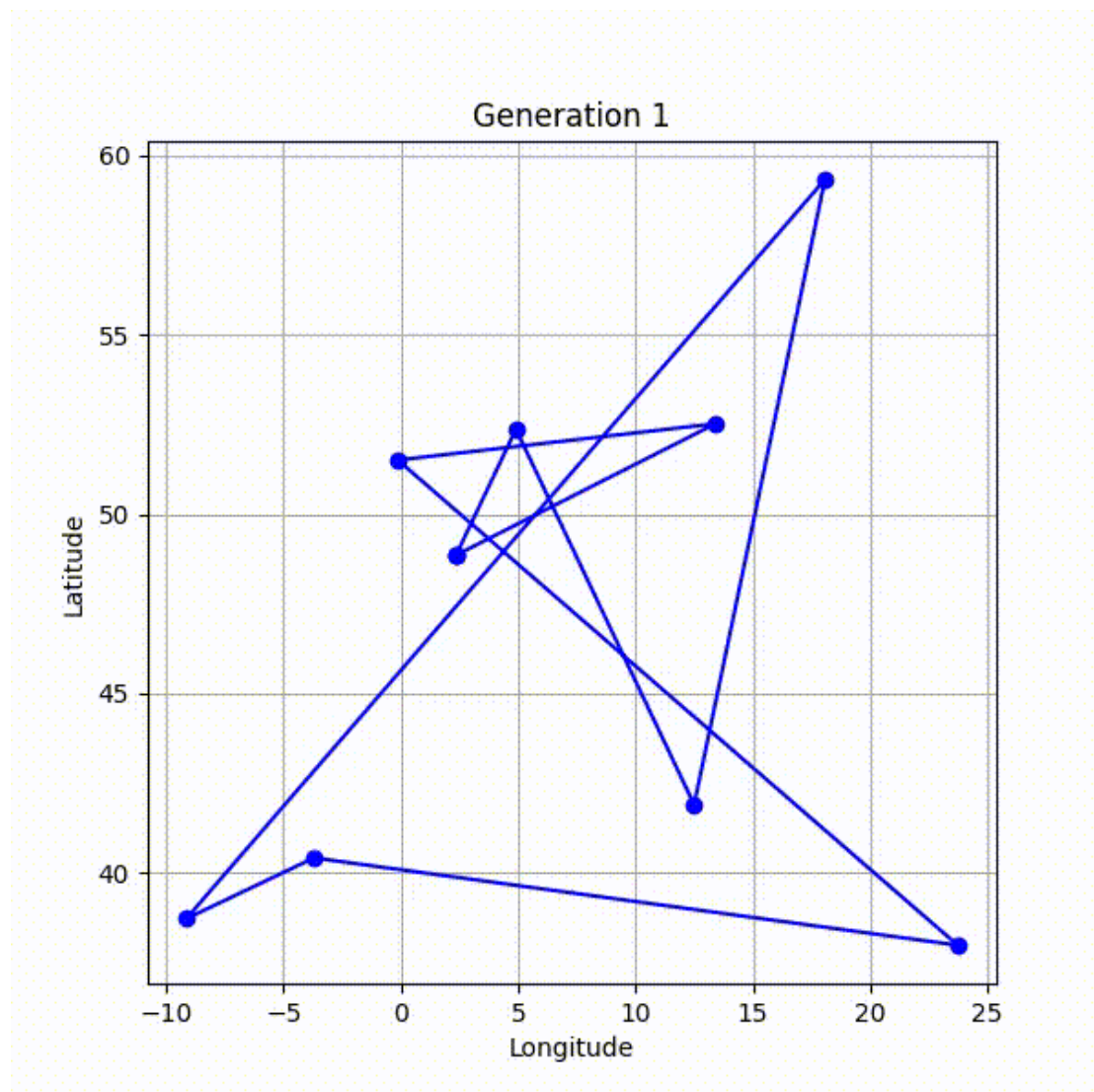
5.2.4.2 Σετ Δέκα Πόλεων



Εικόνα 16. Αποτέλεσμα καλύτερης διαδρομής (best tour) που βρέθηκε από τον αλγόριθμο, πάνω σε διαδραστικό χάρτη.

Η παραπάνω εικόνα απεικονίζει την τελική βέλτιστη διαδρομή που εντοπίστηκε από τον εξελικτικό αλγόριθμο με δέκα ευρωπαϊκές πόλεις. Η διαδρομή ξεκινά από την Αθήνα, συνεχίζει προς Ρώμη, Λισαβόνα, Μαδρίτη, Παρίσι, Λονδίνο, Άμστερνταμ, Βερολίνο, Στοκχόλμη, και καταλήγει στη Βιέννη, σχηματίζοντας έναν πλήρη κύκλο επιστρέφοντας στο αρχικό σημείο. Η διαδρομή που προέκυψε καλύπτει τις πόλεις με τρόπο κυκλικό, ξεκινώντας και καταλήγοντας στο ίδιο σημείο και βλέπουμε μια γεωμετρική "ομαλότητα", χωρίς απότομες επιστροφές. Ο αλγόριθμος φαίνεται να έχει διαχειριστεί με επιτυχία την αύξηση των πόλεων, καθώς η λύση παραμένει λογική ως

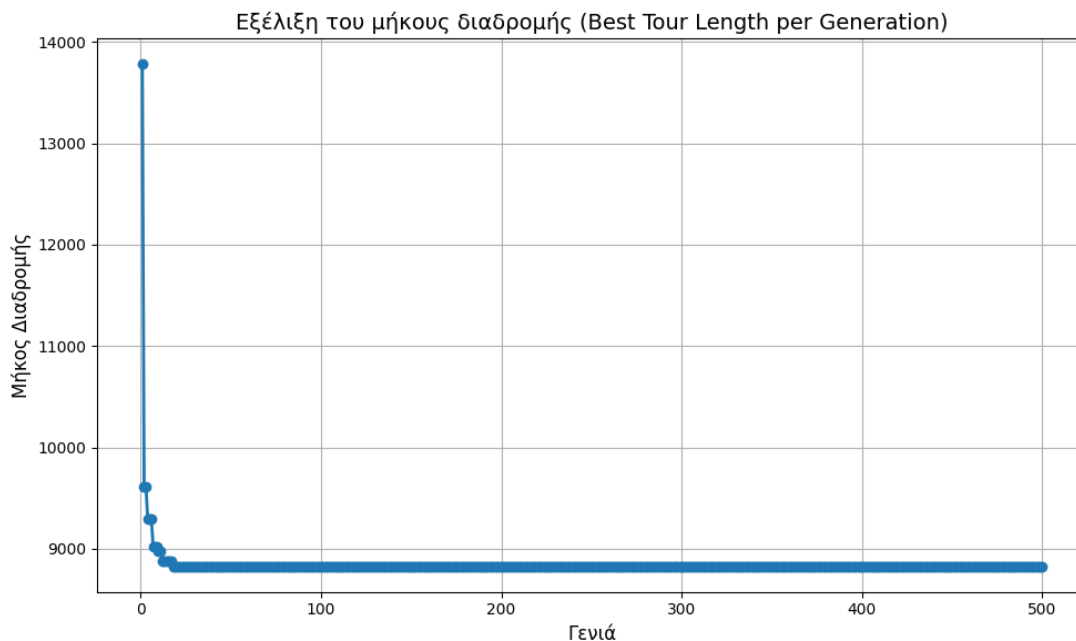
προς την πορεία. Η συνολική απόσταση της βέλτιστης διαδρομής που βρέθηκε είναι 8823.41 km.



Εικόνα 17. Απεικόνιση εξέλιξης αλγορίθμου ανά γενιά.

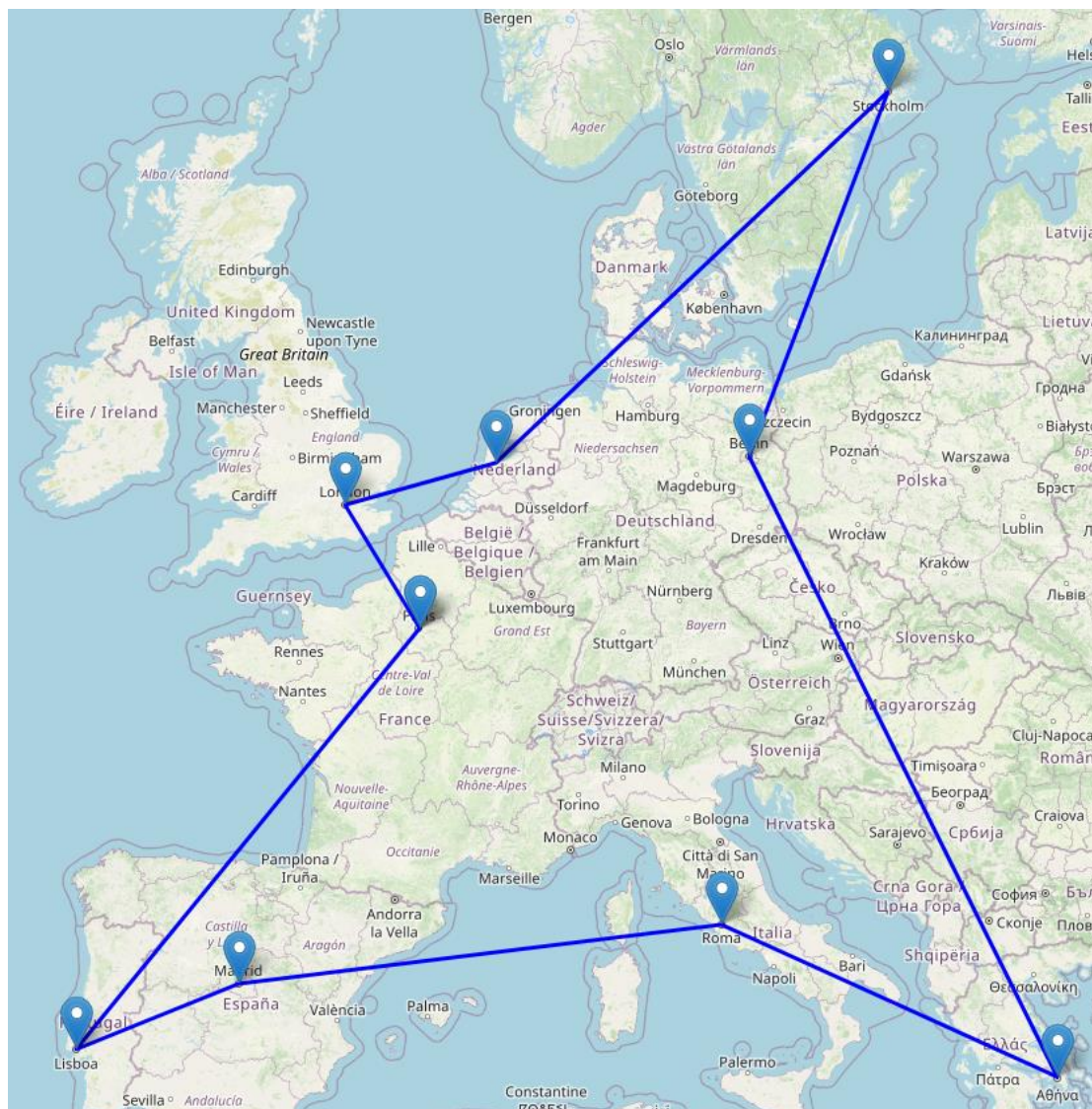
Βλέποντας την εξέλιξη του αλγορίθμου ανά γενιά, παρατηρείται από την πρώτη κιόλας γενιά μια ακανόνιστη διαδρομή, με εμφανείς "τομές" και περιττές μετακινήσεις μεταξύ απομακρυσμένων πόλεων. Η αρχική λύση, αν και έγκυρη, χαρακτηρίζεται από υψηλό συνολικό μήκος διαδρομής και μη αποδοτική σειρά επίσκεψης. Καθώς ο αλγόριθμος εξελίσσεται, η διαδρομή εξομαλύνεται και μειώνεται το συνολικό της μήκος. Οι τεχνικές επιλογής, διασταύρωσης και μετάλλαξης συνδυάζονται αποτελεσματικά ώστε να απορρίπτονται διαδρομές χαμηλής ποιότητας και να διατηρούνται οι καλύτερες. Ειδικά μετά τις πρώτες 20

γενιές, η διαδρομή φαίνεται να σταθεροποιείται σε πολύ καλή μορφή, κάτι που αποδεικνύει την ικανότητα του αλγορίθμου να εντοπίζει αποτελεσματικά τοπικά άριστες λύσεις.



Εικόνα 18. Εξέλιξη μήκους διαδρομής

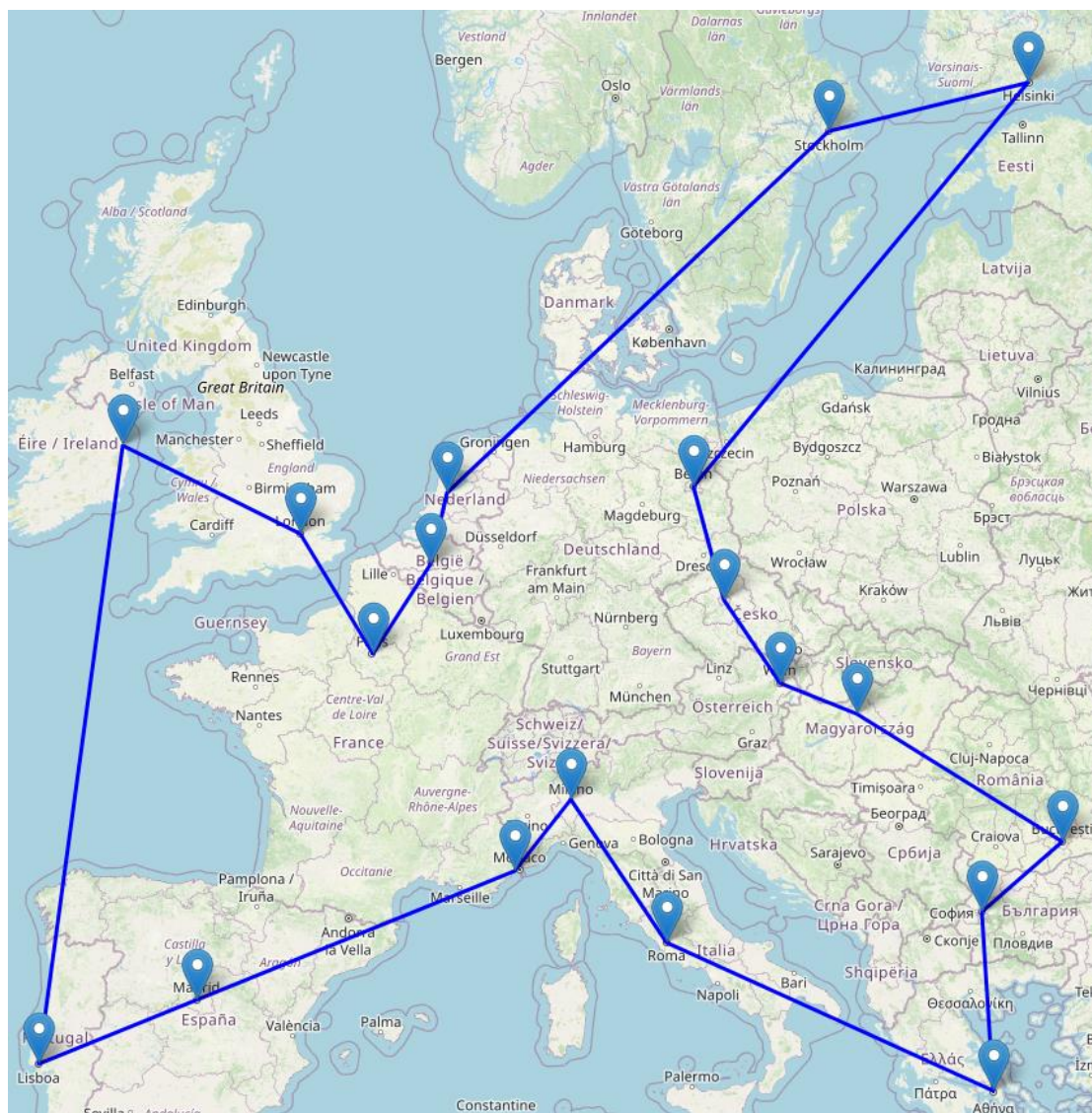
Παρατηρώντας την εξέλιξη του μήκους διαδρομής, μέσα στις πρώτες 20 γενιές επιτυγχάνεται μεγάλη μείωση του συνολικού μήκους διαδρομής. Μετά από αυτό το αρχικό στάδιο ταχείας σύγκλισης, η καμπύλη παρουσιάζει μια εμφανή φάση σταθεροποίησης. Η τιμή του μήκους παραμένει πρακτικά αμετάβλητη για το υπόλοιπο της εξελικτικής διαδικασίας.



Εικόνα 19. Αποτέλεσμα καλύτερης διαδρομής (best tour) που βρέθηκε με τη μέθοδο brute-force, πάνω σε διαδραστικό χάρτη.

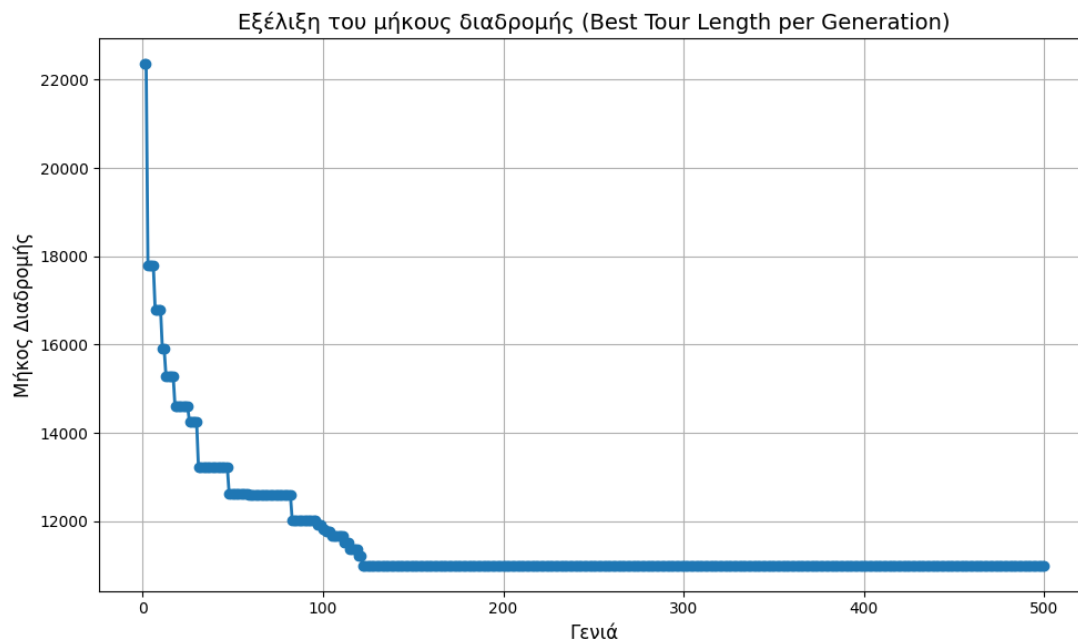
Η εγκυρότητα της παραγόμενης λύσης επαληθεύτηκε μέσω της μεθόδου brute-force, η οποία παραμένει υπολογιστικά εφικτή σε προβλήματα μεσαίας κλίμακας όπως το συγκεκριμένο (10 πόλεις). Η εξαντλητική αξιολόγηση όλων των πιθανών permutations ($10! = 3.628.800$ διαφορετικές διαδρομές) επιβεβαίωσε ότι η διαδρομή που εντοπίστηκε από τον αλγόριθμο ήταν η βέλτιστη, με συνολικό μήκος 8823.4177 km, ίδιο ακριβώς με αυτό που προέκυψε από την πλήρη διερεύνηση του χώρου λύσεων.

5.2.4.3 Σετ Είκοσι Πόλεων



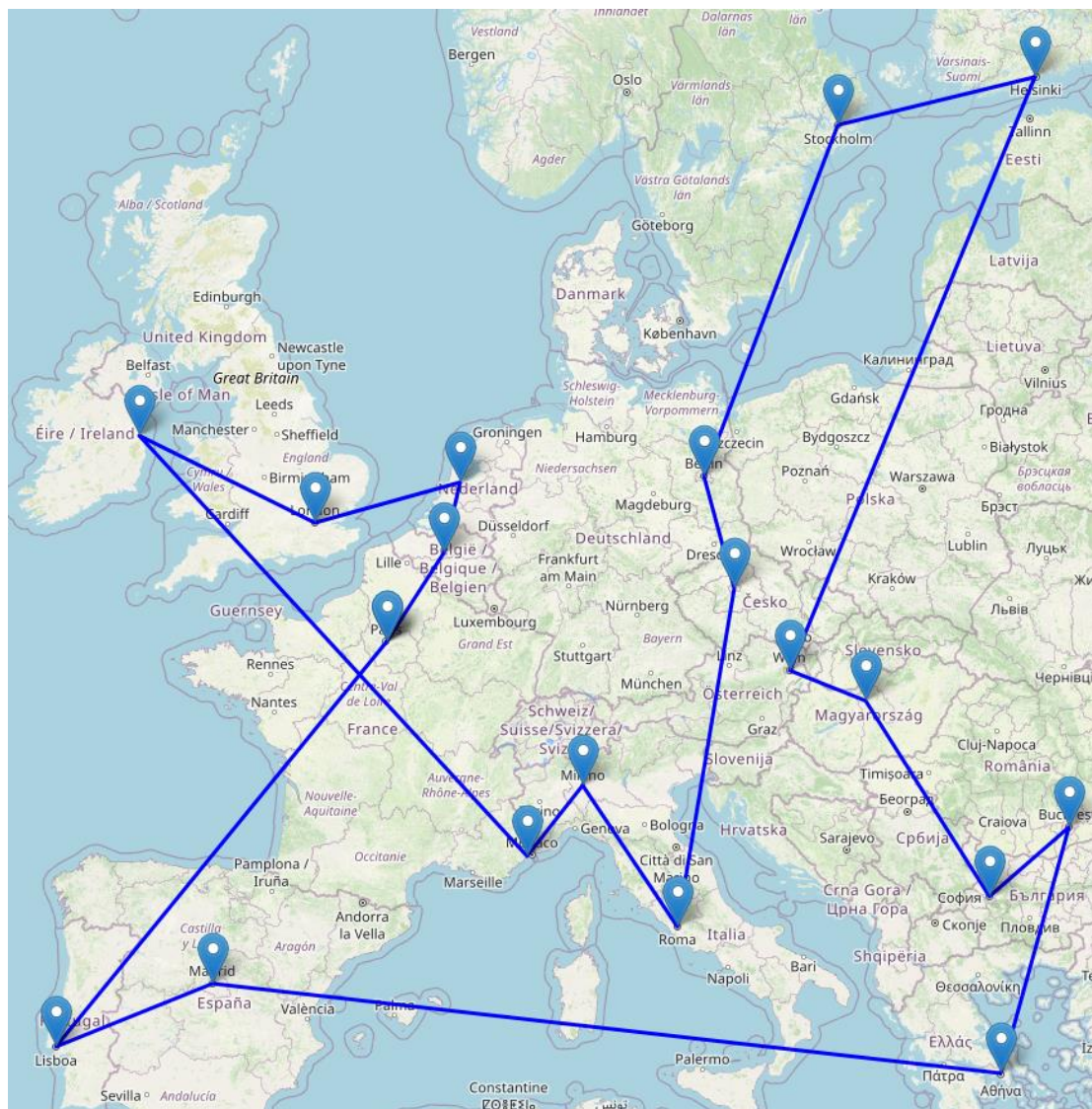
Εικόνα 20. Αποτέλεσμα καλύτερης διαδρομής (best tour) που βρέθηκε από τον αλγόριθμο, πάνω σε διαδραστικό χάρτη.

Η παραπάνω εικόνα απεικονίζει την τελική βέλτιστη διαδρομή που εντοπίστηκε από τον εξελικτικό αλγόριθμο για το πρόβλημα του Πλανόδιου Πωλητή με δέκα ευρωπαϊκές πόλεις. Η διαδρομή ξεκινά από την Αθήνα, συνεχίζει προς Ρώμη, Λισαβόνα, Μαδρίτη, Παρίσι, Λονδίνο, Άμστερνταμ, Βερολίνο, Στοκχόλμη, και καταλήγει στη Βιέννη, σχηματίζοντας έναν πλήρη κύκλο επιστρέφοντας στο αρχικό σημείο. Η διαδρομή που εντοπίστηκε σχηματίζει έναν λογικό κυκλικό βρόχο, αποφεύγοντας μεγάλες αποστάσεις μεταξύ απομακρυσμένων πόλεων και έχοντας σαφή γεωγραφική συνέχεια. Το συνολικό μήκος της βέλτιστης διαδρομής είναι 10.989,83 χιλιόμετρα.



Εικόνα 22. Εξέλιξη μήκους διαδρομής

Η γραφική απεικόνιση της εξέλιξης του μήκους διαδρομής στις 20 πόλεις αποτυπώνει καθαρά την αποτελεσματικότητα του εξελικτικού αλγορίθμου. Κατά τις πρώτες 50–70 γενιές παρατηρείται απότομη μείωση της συνολικής απόστασης, με τον αλγόριθμο να εντοπίζει κρίσιμες βελτιώσεις στη σειρά επίσκεψης των πόλεων. Στη συνέχεια, η καμπύλη παρουσιάζει πιο ήπια πτώση, ενώ από τη 150η γενιά και μετά σταθεροποιείται, υποδεικνύοντας πως ο αλγόριθμος έχει βρει την καλύτερη διαδρομή.

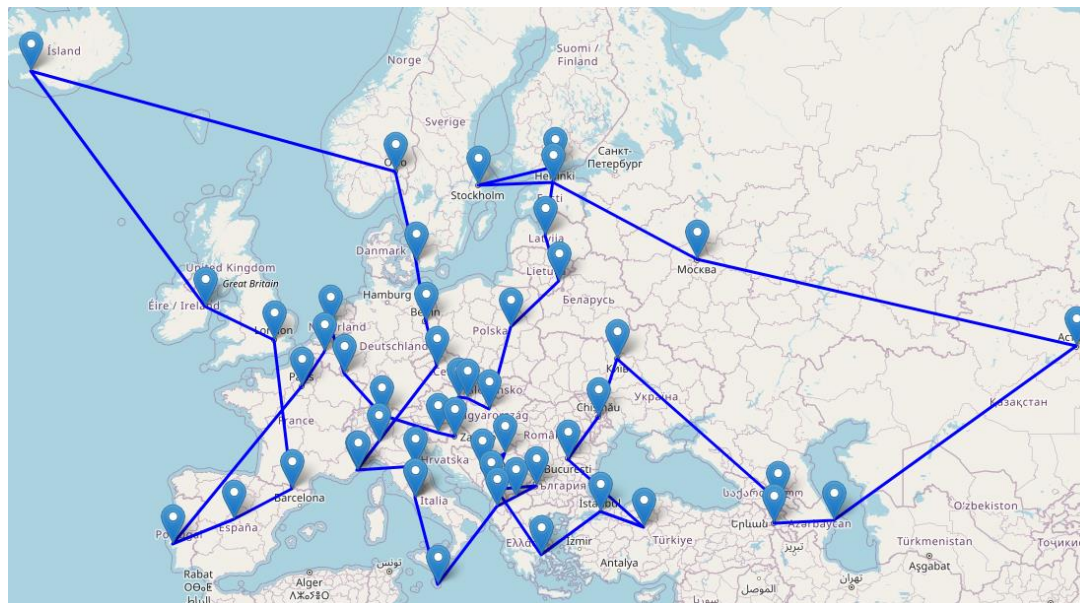


Εικόνα 23. Αποτέλεσμα καλύτερης διαδρομής (best tour) που βρέθηκε με τη μέθοδο MST + DFS, πάνω σε διαδραστικό χάρτη.

Για την περίπτωση των είκοσι πόλεων, πραγματοποιήθηκε σύγκριση της βέλτιστης διαδρομής που προέκυψε από τον εξελικτικό αλγόριθμο με εκείνη που προκύπτει μέσω της κλασικής μεθόδου MST + DFS. Η συνολική απόσταση της διαδρομής του αλγορίθμου ήταν 10.989,83 km, ενώ η αντίστοιχη απόσταση μέσω MST + DFS ανήλθε στα 13.925,26 km.

Ο αλγόριθμος παρήγαγε μια σαφώς αποδοτικότερη διαδρομή, με μείωση μήκους κατά περίπου 2.935 km. Η διαδρομή του MST + DFS, αν και γεωμετρικά λογική, περιλαμβάνει πολλές πλάγιες μετακινήσεις που αυξάνουν το συνολικό μήκος. Αντίθετα, ο εξελικτικός αλγόριθμος κατάφερε να συνδυάσει τη γεωμετρική συνέχεια με την ελαχιστοποίηση της απόστασης.

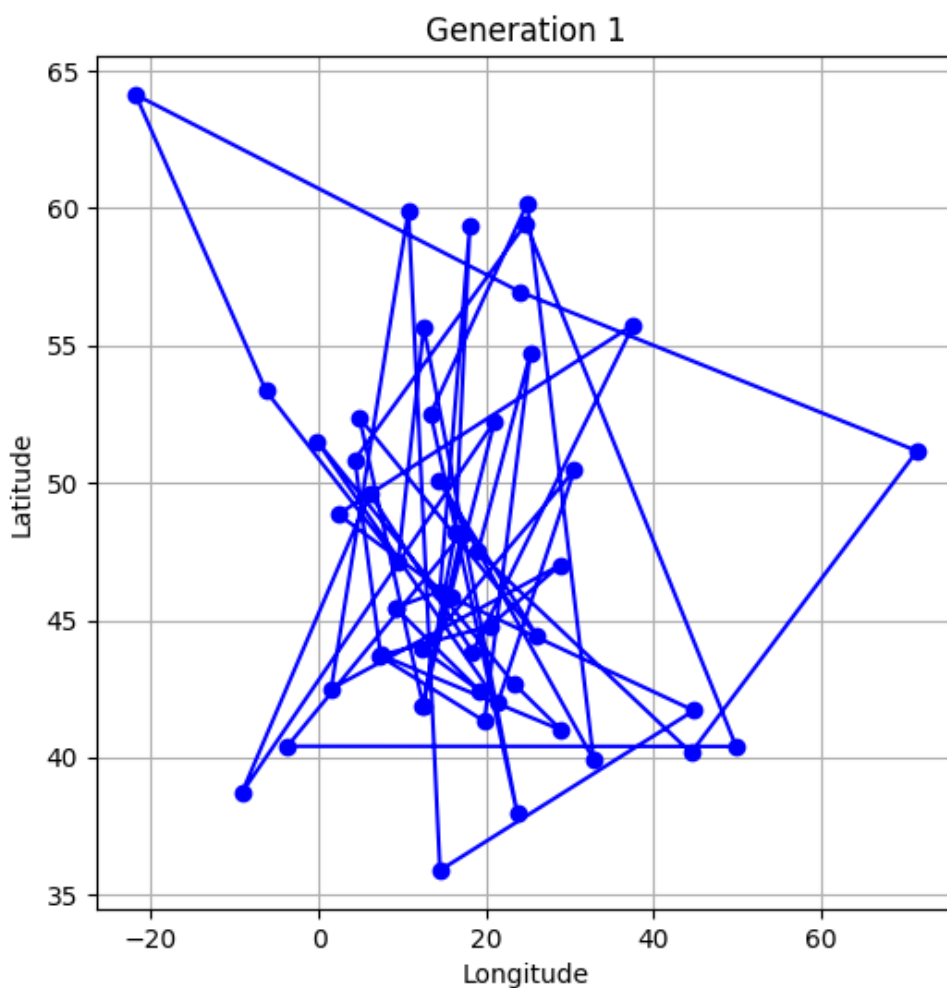
5.2.4.4 Σετ Πενήντα Πόλεων



Εικόνα 24. Αποτέλεσμα καλύτερης διαδρομής (*best tour*) που βρέθηκε από τον αλγόριθμο, πάνω σε διαδραστικό χάρτη.

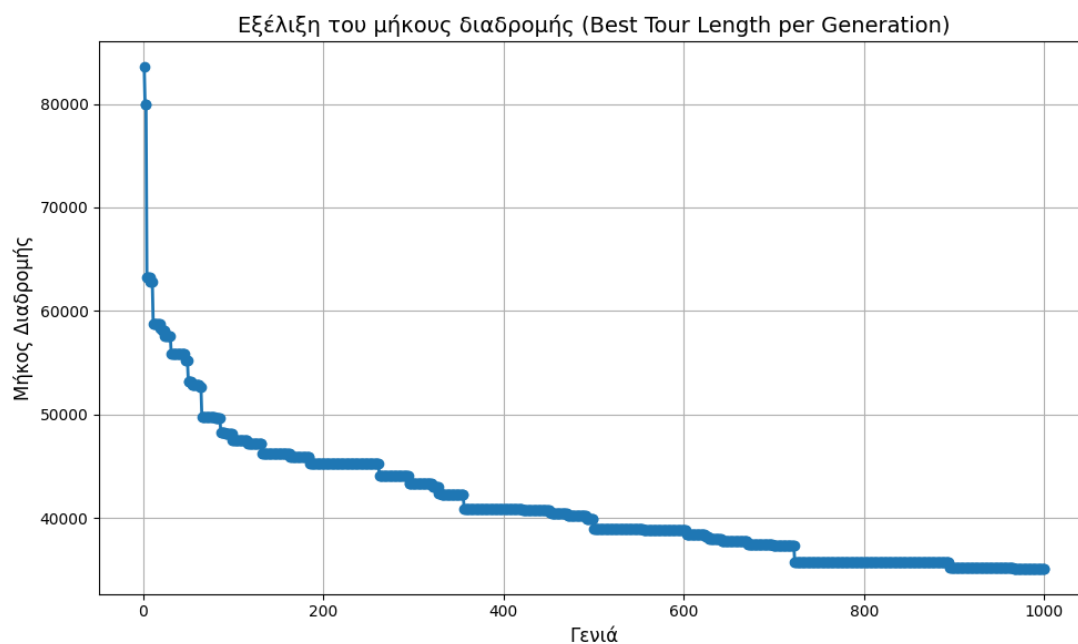
Η σειρά της καλύτερης διαδρομής που βρέθηκε είναι η εξής: Αστάνα → Μόσχα → Τιφλίδα → Γερεβάν → Μπακού → Κίεβο → Κισινάου → Κωνσταντινούπολη → Άγκυρα → Αθήνα → Τίρανα → Ποντοκόριτσα → Σκόπια → Σαράγεβο → Σόφια → Βελιγράδι → Ζάγκρεμπ → Βιέννη → Ανδόρρα → Μονακό → Σαν Μαρίνο → Βατικανό → Ρώμη → Μιλάνο → Νίκαια → Λιουμπλιάνα → Μπρατισλάβα → Βουδαπέστη → Πράγα → Βαλέτα → Ρέικιαβικ → Λουξεμβούργο → Παρίσι → Βρυξέλλες → Βερολίνο → Κοπεγχάγη → Στοκχόλμη → Ελσίνκι → Ταλίν → Ρίγα → Βίλνιους → Άμστερνταμ → Λονδίνο → Δουβλίνο → Μπαρσελόνα → Μαδρίτη → Λισαβόνα → Βαντούζ → Μόναχο → Νυρεμβέργη

Η συνολική απόσταση που διανύθηκε στη συγκεκριμένη διαδρομή ανέρχεται σε 35.108,51 χιλιόμετρα, τιμή η οποία κρίνεται ως ικανοποιητική με βάση την πολυπλοκότητα του προβλήματος. Αν και παρατηρούνται κάποιες εμφανείς διασταυρώσεις, γεγονός που υποδεικνύει ότι πιθανόν να μην πρόκειται για την απόλυτα βέλτιστη λύση, η διαδρομή παραμένει απολύτως αποδεκτή. Η κατανομή των επισκέψεων χαρακτηρίζεται από γεωγραφική συνέπεια και λογική ακολουθία, χωρίς έντονες επιστροφές ή υπερβολικά μεγάλα άλματα. Πρόκειται για μια αξιοπρεπή λύση, όχι όμως η βέλτιστη.



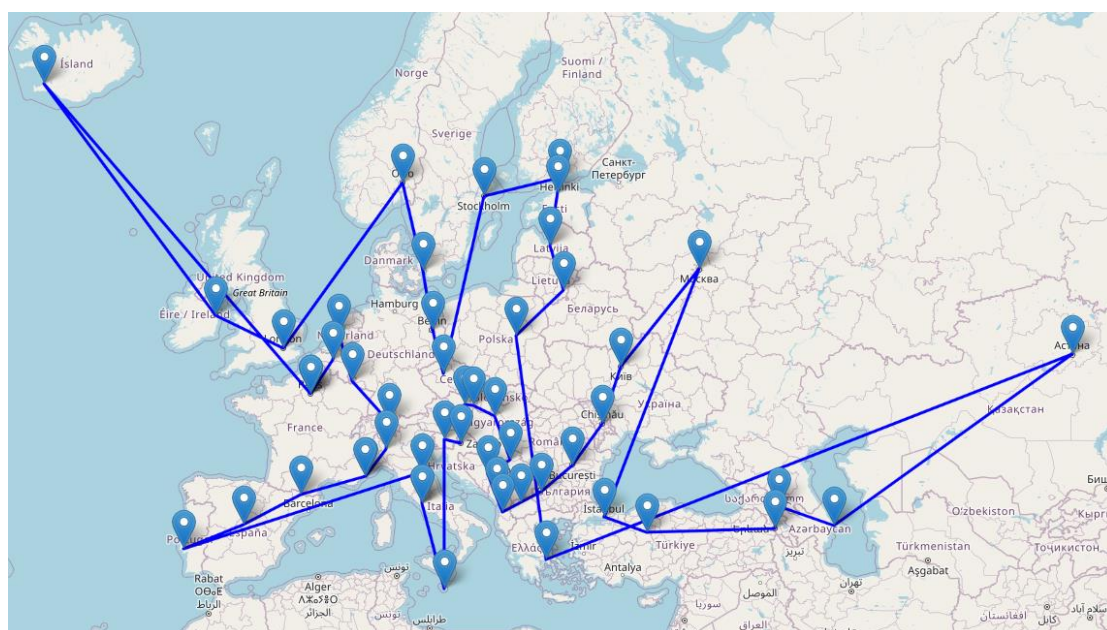
Εικόνα 25. Απεικόνιση εξέλιξης αλγορίθμου ανά γενιά.

Εδώ η αρχική γενιά έχει μια πολύ χαοτική και ακανόνιστη διαδρομή, με πλήθος τομών και μη ρεαλιστικών μεταβάσεων. Καθώς οι γενιές προχωρούν, η διαδρομή εξομαλύνεται, μειώνεται το συνολικό της μήκος και προσεγγίζει πιο «λογική» πορεία. Οι βελτιώσεις είναι ιδιαίτερα έντονες στις πρώτες 100–200 γενιές, ενώ το γράφημα δείχνει μια ξεκάθαρη οπτική απόδειξη της εξελικτικής διαδικασίας βελτιστοποίησης.



Εικόνα 26. Εξέλιξη μήκους διαδρομής

Στην εξέλιξη του μήκους της διαδρομής παρατηρείται μια απότομη πτώση στις πρώτες 100–150 γενιές, κατά την οποία ο αλγόριθμος απορρίπτει τις αρχικά τυχαίες λύσεις. Στη συνέχεια, η μείωση είναι πιο σταδιακή, με ορισμένα «σκαλοπάτια» βελτιώσεων καθώς εμφανίζονται καλύτερες λύσεις και μετά τις ~700 γενιές, η τιμή φαίνεται να σταθεροποιείται γύρω από μια καλή τοπικά άριστη λύση, με ελάχιστες έως καθόλου μεταβολές.



Εικόνα 27. Αποτέλεσμα καλύτερης διαδρομής (best tour) που βρέθηκε με τη μέθοδο MST + DFS, πάνω σε διαδραστικό χάρτη.

Για σύγκριση, εφαρμόστηκε και η μέθοδος MST + DFS και το συνολικό μήκος της διαδρομής που προέκυψε ήταν 345.89 χιλιόμετρα. Η πολύ μικρή τιμή προκύπτει από το γεγονός ότι η μέθοδος MST δεν υπολογίζει το συνολικό κλειστό κύκλο με επιστροφή στην αρχή, αλλά λειτουργεί περισσότερο ως μια προσέγγιση περιήγησης με βάση δέντρο. Έτσι, παρότι εμφανισιακά ο χάρτης του MST+DFS φαίνεται πιο ομαλός, δεν συνιστά απαραίτητως πιο αποδοτική διαδρομή υπό τους όρους του TSP.

5.2.6 Συμπεράσματα επίλυσης προβλήματος

Ο συνδυασμός νευρωνικού δικτύου και εξελικτικού αλγορίθμου παράγαγε ικανοποιητικά αποτελέσματα στην επίλυση του προβλήματος του Πλανόδιου Πωλητή, ιδιαίτερα για περιπτώσεις με μικρό ή μεσαίο πλήθος πόλεων (έως και 20). Οι τελικές διαδρομές χαρακτηρίζονται από λογική και σχετικά μικρό συνολικό μήκος. Η συμβολή του νευρωνικού δικτύου, παρόλο που βασίστηκε σε μια απλή αρχιτεκτονική, αποδείχθηκε σημαντική στην παραγωγή αρχικού πληθυσμού με πιο ευνοϊκή κατανομή, βελτιώνοντας έτσι την ταχύτητα με την οποία ο αλγόριθμος θα φτάσει σε μια βέλτιστη λύση.

5.2.7 Γραφικό περιβάλλον (GUI)

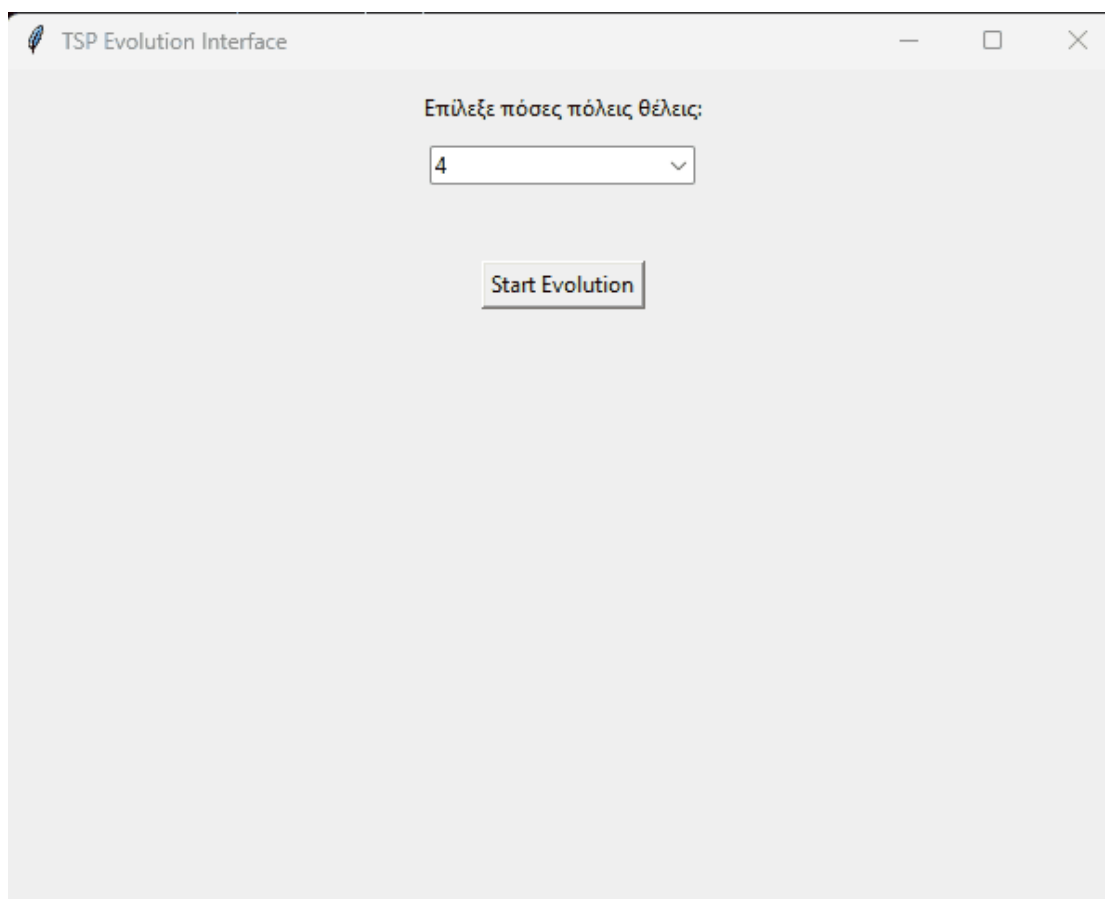
Στο πλαίσιο της παρούσας υλοποίησης, αναπτύχθηκε ένα απλό γραφικό περιβάλλον χρήστη (GUI), το οποίο επιτρέπει στον χρήστη να εκτελέσει το μοντέλο επίλυσης του προβλήματος TSP με εύχρηστο και διαδραστικό τρόπο.

Η εφαρμογή διατίθεται ως εκτελέσιμο αρχείο (.exe) και παρέχει τις εξής δυνατότητες:

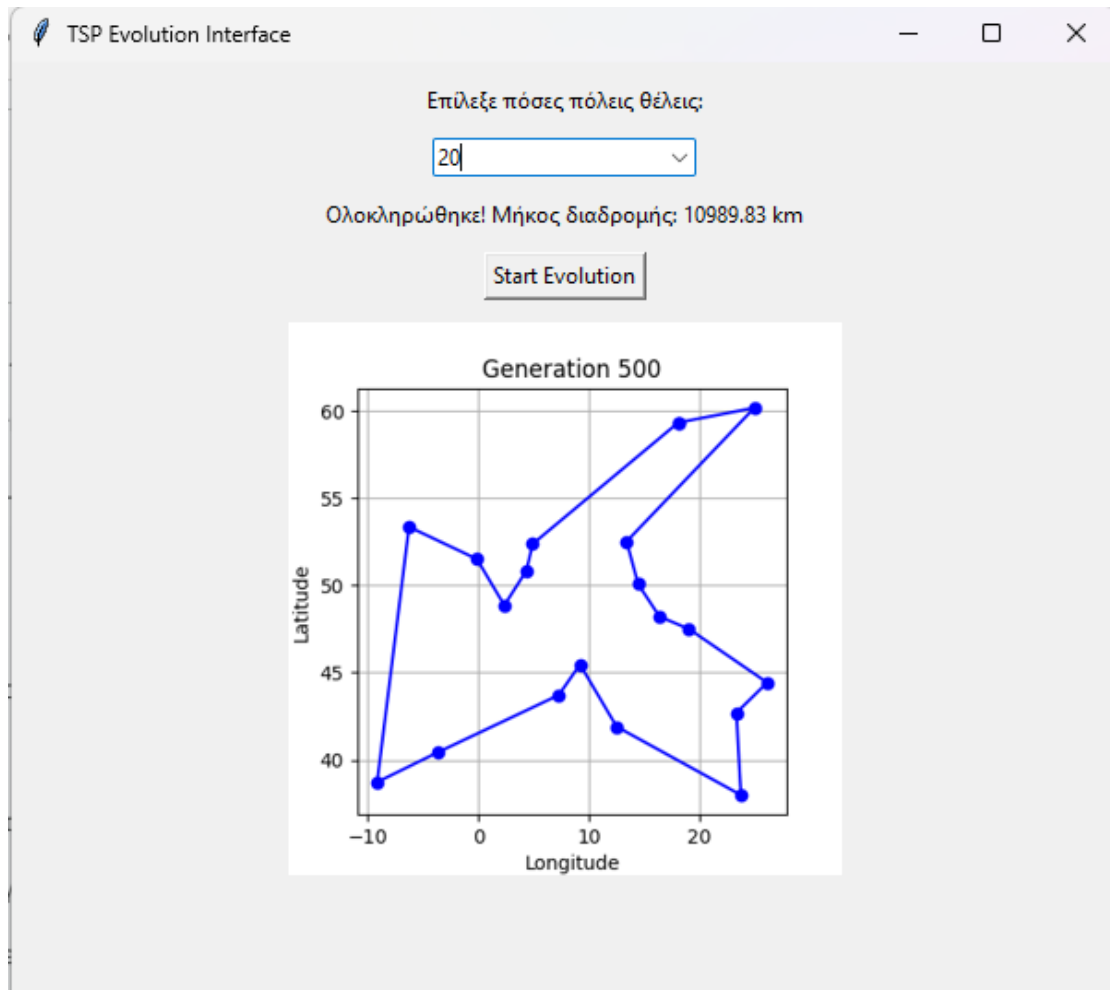
- **Επιλογή Συνόλου Πόλεων:** Ο χρήστης μπορεί να επιλέξει από διαθέσιμα σύνολα πόλεων (π.χ. 4, 10, 20 ή 50 πόλεις), μέσω dropdown list.
- **Live Εμφάνιση Προόδου:** Καθ' όλη τη διάρκεια της εξελικτικής διαδικασίας, εμφανίζεται δυναμικά ένα γράφημα με την τρέχουσα καλύτερη διαδρομή για κάθε γενιά. Ο χρήστης μπορεί να παρακολουθήσει πώς βελτιώνεται σταδιακά η λύση του αλγορίθμου.

- **Αυτόματη Εμφάνιση Χάρτη:** Μετά την ολοκλήρωση της εκτέλεσης, η εφαρμογή ανοίγει αυτόματα στον browser έναν διαδραστικό χάρτη (μέσω folium) με τη βέλτιστη διαδρομή που βρέθηκε.
- **Εμφάνιση Συνολικού Μήκους:** Μαζί με τον χάρτη, προβάλλεται και το συνολικό μήκος της διαδρομής που επετεύχθη, επιτρέποντας την άμεση αξιολόγηση της λύσης.

Η συγκεκριμένη λειτουργικότητα καθιστά το εργαλείο ιδιαίτερα φιλικό προς τον τελικό χρήστη, επιτρέποντάς του όχι μόνο να εκτελέσει το μοντέλο χωρίς χρήση γραμμής εντολών, αλλά και να κατανοήσει οπτικά τη συμπεριφορά και την απόδοση του αλγορίθμου σε διαφορετικά προβλήματα μεγέθους.



Εικόνα 28. Οθόνη GUI



Εικόνα 29. Οθόνη GUI - Τελικό αποτέλεσμα

6. Βιβλιογραφία

- [1] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan και S. Wermter, «Continual lifelong learning with neural networks: A review», *Neural Networks*, vol. 113, pp. 54–71, 2019.
- [2] R. Mahajan και G. Kaur, «Neural Networks using Genetic Algorithms», Guru Nanak Dev University, Amritsar, 2017.
- [3] G. Vahdati, S. Y. Ghouhani και M. Yaghoobi, «A hybrid search algorithm with Hopfield neural network and Genetic algorithm for solving traveling salesman problem», *Islamic Azad University & Ferdowsi University of Mashhad, Iran*.
- [4] A. F. El-Samak και W. Ashour, «Optimization of Traveling Salesman Problem using Affinity Propagation Clustering and Genetic Algorithm», *International Journal of Computer Applications*, vol. 181, no. 34, pp. 25–30, 2019.
- [5] Juwairiah, D. Pratama, H. C. Rustamaji, H. Sofyan και D. B. Prasetyo, «Genetic Algorithm for Optimizing Traveling Salesman Problems with Time Windows (TSP-TW)», *Procedia Computer Science*, vol. 135, pp. 33–40, 2018.
- [6] F. Al-Rubaie και M. Alhanjouri, «Optimization of Travelling Salesman Problem using NN Clustering and Genetic Algorithm», *Islamic University of Gaza, Palestine*, 2020.
- [7] J. Bogerd, «Combining Neural Networks and Linear Programming for the Traveling Salesman Problem», *Bachelor Thesis, Faculty of Science, Radboud University Nijmegen*, 2016.

- [8] D. E. Rumelhart, G. E. Hinton και R. J. Williams, «Learning representations by back-propagating errors», *Nature*, vol. 323, pp. 533–536, 1986.
- [9] Y. LeCun, Y. Bengio και G. Hinton, «Deep learning», *Nature*, vol. 521, pp. 436–444, 2015.
- [10] C. Guo, A. Mousavi, H. Chen και M. B. Goudarzi, «A review of neural networks for the traveling salesman problem», *International Journal of Artificial Intelligence*, vol. 14, no. 1, pp. 1–20, 2016.
- [11] R. Cangelosi και M. Schlesinger, *Developmental Robotics: From Babies to Robots*, MIT Press, 2015.
- [12] J. Tani, *Exploring Robotic Minds: Actions, Symbols, and Consciousness as Self-Organizing Dynamic Phenomena*, Oxford University Press, 2016.
- [13] G. A. Grossberg, «Competitive learning: From interactive activation to adaptive resonance», *Cognitive Science*, vol. 11, no. 1, pp. 23–63, 1987.
- [14] H. D. Hubel και T. N. Wiesel, «Receptive fields of single neurons in the cat's striate cortex», *Journal of Physiology*, vol. 148, pp. 574–591, 1962.