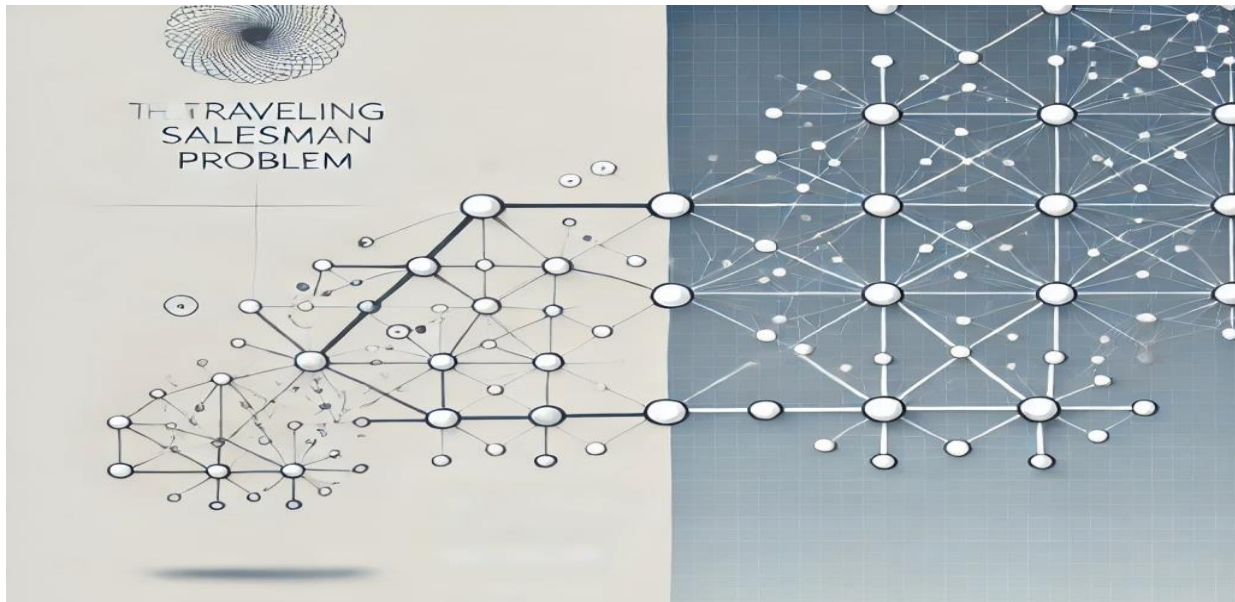


ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

« Γραφική Επίλυση του Προβλήματος του Περιοδεύοντος Πωλητή με τη χρήση Αναδρομικών/τύπου Hopfield Νευρωνικών Δικτύων»



Του φοιτητή
Μήτρου Άγγελου
Αρ. Μητρώου: 144170

Επιβλέπων
Όνοματεπώνυμο: Γουλιάνας Κων/νος
Βαθμίδα: Καθηγητής

29/05/2025

Γραφική επίλυση του Προβλήματος του Περιοδεύοντος Πωλητή με τη χρήση Αναδρομικών/τύπου
Hopfield Νευρωνικών Δικτύων
22166
Μήτρου Άγγελος
Γουλιάνας Κωνσταντίνος
28/09/2022
29/05/2025

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Μήτρου Άγγελου που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

«Αφιέρωση»

Πρόλογος

Σε μια εποχή όπου η τεχνητή νοημοσύνη αναπτύσσεται ραγδαία και έχει ενταχθεί πλέον δυναμικά στην καθημερινότητα μας σε πολλούς τομείς όπως η υγεία, η οικονομία, η αυτοκίνηση αλλά και η διασκέδαση, η επιλογή ενός θέματος που σχετίζεται με αυτή ήταν εξαιρετικά ελκυστική.

Ειδικότερα τα αναδρομικά νευρωνικά δίκτυα τα οποία εμπνέονται από ορισμένες πτυχές της ικανότητας του ανθρώπινου εγκεφάλου να επεξεργάζεται διαδοχικές πληροφορίες και η χρήση τους για την λύση προβλημάτων βελτιστοποίησης όπως αυτό του Περιοδεύοντος Πωλητή κέντρισαν το ενδιαφέρον μου και με οδήγησαν στην συγγραφή της συγκεκριμένης διπλωματικής εργασίας.

Στις ημέρες μας το εμπόριο δεν σταματά να αναπτύσσεται και με τα ενεργειακά κόστη μετακίνησης να αυξάνονται συνεχώς, η επιλογή μιας βέλτιστης διαδρομής μπορεί να έχει μεγάλα οφέλη.

Το διάστημα που εκπόνησα την διπλωματική εργασία μου προσέφερε γνώσεις τόσο σε ακαδημαϊκό όσο και σε προσωπικό επίπεδο τις οποίες έχω ως εφόδιά μου για την μελλοντική μου πορεία.

Περίληψη

Το Πρόβλημα του Ταξιδιώτη Πωλητή (TSP), μια κλασική πρόκληση συνδυαστικής βελτιστοποίησης, περιλαμβάνει την εύρεση της πιο αποτελεσματικής διαδρομής που επισκέπτεται ένα σύνολο πόλεων ακριβώς μία φορά και επιστρέφει στην αρχή, ελαχιστοποιώντας τη συνολική απόσταση που έχει διανυθεί. Η συγκεκριμένη εργασία διερευνά μια προσέγγιση για την επίλυση του TSP αξιοποιώντας τα αναδρομικά νευρωνικά δίκτυα τύπου Hopfield για την εύρεση της βέλτιστης διαδρομής .

Αρχικά, πραγματοποιείται βιβλιογραφική παρουσίαση διάφορων λύσεων που βασίζονται στα αναδρομικά νευρωνικά δίκτυα τύπου Hopfield και έχουν προταθεί για το συγκεκριμένο πρόβλημα.

Γίνεται η επιλογή της εργασίας των García, Talaván και Yáñez (2016)[1] για την υλοποίηση της προσομοίωσης η οποία , προτείνει μια στρατηγική διαίρει και βασίλευε (divide-and-conquer) που βασίζεται σε δύο φάσεις. Η πρώτη φάση περιλαμβάνει τη σύνδεση των πόλεων με τους περισσότερους γείτονες για να οριστεί ένα σύνολο αλυσίδων πόλεων και η δεύτερη φάση, για να ενώσει αυτές με τις απομονωμένες πόλεις έτσι ώστε να ορίσει την τελική περιήγηση

Ακόμη, γίνεται η προσομοίωση με την ταυτόχρονη επίδειξη της διαδικασίας επίλυσης του προβλήματος στο χάρτη της Ελλάδας. Η ενσωμάτωση γραφικών λύσεων υπόσχεται αποτελεσματικές αναλύσεις σε αυτό το περιβόητο πρόβλημα βελτιστοποίησης.

Τέλος, θα δούμε ποιες μεθοδολογίες και ποια εργαλεία χρησιμοποιήθηκαν την ανάλυση στα στάδια της δημιουργίας της προσομοίωσης και τα συμπεράσματα που βγήκαν από το τελικό αποτέλεσμα.

Abstract

The Traveling Salesman Problem (TSP), a classic challenge in combinatorial optimization, involves finding the most efficient route that visits a set of cities exactly once and returns to the starting point, minimizing the total distance traveled. This study explores an approach for solving the TSP using Hopfield-type recurrent neural networks to determine the optimal route.

Initially, a literature review is conducted, presenting various solutions based on Hopfield recurrent neural networks that have been proposed for tackling this specific problem.

A particular study by [Author(s)] is selected for the simulation implementation, which proposes a divide-and-conquer strategy based on two phases. The first phase involves connecting cities with the highest number of neighbors to form a set of city chains. The second phase then connects these chains with the isolated cities to define the final tour.

Additionally, the simulation is presented alongside a visual representation of the problem-solving process on the map of Greece. The integration of graphical solutions promises effective insights into this well-known optimization problem.

Finally, we discuss the methodologies and tools used during the development stages of the simulation, as well as the conclusions drawn from the final results.

Ευχαριστίες

Θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες σε όλους όσοι συνέβαλαν στην περάτωση της διπλωματικής μου εργασίας. Ευχαριστώ το τμήμα της σχολής, για τις γνώσεις και την εμπειρία που μου χάρισε και οι οποίες αποτέλεσαν καθοριστικό παράγοντα για την επιτυχή ολοκλήρωση του έργου και της ακαδημαϊκής μου πορείας. Επίσης, θα ήθελα να ευχαριστήσω την οικογένειά, την σύντροφο καθώς και τους οικείους μου για την αμέριστη στήριξη, την κατανόηση και την ενθάρρυνση που μου παρείχαν σε όλα αυτά τα χρόνια των σπουδών μου. Τέλος, ευχαριστώ τους φίλους και συμφοιτητές μου που με τις συζητήσεις και την ανταλλαγή ιδεών συνέβαλαν στον εμπλουτισμό της σκέψης μου και στη διαμόρφωση των απόψεών σε όλη την πορεία μου καθώς και για το θέμα της παρούσας εργασίας.

Περιεχόμενα

Πρόλογος.....	Error! Bookmark not defined.
Περίληψη	Error! Bookmark not defined.
Abstract.....	Error! Bookmark not defined.
Ευχαριστίες	Error! Bookmark not defined.
Περιεχόμενα	Error! Bookmark not defined.
Κατάλογος Σχημάτων	Error! Bookmark not defined.ii
Κατάλογος Πινάκων	Error! Bookmark not defined.ii
Συντομογραφίες.....	xError! Bookmark not defined.v
Κεφάλαιο 1ο: Το πρόβλημα του περιοδεύοντος πωλητή	Error! Bookmark not defined.6
1.1 Εισαγωγή	Error! Bookmark not defined.6
1.2 Ιστορική αναδρομή.....	Error! Bookmark not defined.6
1.2.1 Μαθηματική απεικόνιση του προβλήματος	Error! Bookmark not defined.9
1.2.2 NP-Πληρότητα	Error! Bookmark not defined.1
1.2.3 Τρόποι επίλυσης	Error! Bookmark not defined.2
1.3 Αλγόριθμος πλησιέστερου γείτονα.....	23
1.3.1 2-opt,3-opt	23
1.3.2 Γενετικοί αλγόριθμοι	Error! Bookmark not defined.3
1.3.3 Αποικία μυρμηγκιών.....	Error! Bookmark not defined.4
1.3.4 Brute force algorithm	Error! Bookmark not defined.4
1.4 Που χρησιμοποιείται η Περιοδεία Περιοδεύοντος Πωλητή	Error! Bookmark not defined.6
1.4.1 Κατασκευή κυκλωμάτων	Error! Bookmark not defined.6
1.4.2 Καλωδίωση υπολογιστή.....	Error! Bookmark not defined.7
1.4.3 Το πρόβλημα επιλογής παραγγελιών σε αποθήκες.....	Error! Bookmark not defined.7
1.4.4 Δρομολόγηση οχημάτων	Error! Bookmark not defined.8
1.4.5 Κρυσταλλογραφιών Ακτίνων X	Error! Bookmark not defined.8
1.4.6 Σχεδιασμός παγκόσμιων τοπογραφικών δικτύων δορυφωρικών συστημάτων πλοήγησης	Error! Bookmark not defined.9
1.5 Σύνδεση με νευρωνικά δίκτυα	Error! Bookmark not defined.9
Κεφάλαιο 2ο: Τα νευρωνικά δίκτυα – Μια εισαγωγή στη τεχνητή νοημοσύνη	Error! Bookmark not defined.1
2.1 Εισαγωγή	Error! Bookmark not defined.1
2.2 Βιολογικό υπόβαθρο και έμπνευση.....	Error! Bookmark not defined.1
2.3 Δομή και λειτουργία των νευρωνικών δικτύων	Error! Bookmark not defined.1
2.4 Εκμάθηση και προσαρμογή	Error! Bookmark not defined.2
2.5 Βαθιά νευρωνικά δίκτυα και επανάσταση στη τεχνητή νοημοσύνη	Error! Bookmark not defined.2

5.2.4	Συμπεράσματα.....	65
5.3	Συνεργατικός νευροδυναμικός αλγόριθμος περιοδούντος πωλητή	66
5.3.1	Μεθοδολογία	66
5.3.2	Πειραματικά αποτελέσματα	67
5.3.3	Συμπεράσματα.....	67
5.4	Συνεργατική νευροδυναμική βελτιστοποίηση με διακριτά δίκτυα Hopfield για την επίλυση του προβλήματος του περιοδούντος πωλητή	67
5.4.1	Μεθοδολογία	68
5.4.2	Συνεργατική νευροδυναμική βελτιστοποίηση.....	68
5.4.3	Πειραματικά αποτελέσματα	69
5.4.4	Συμπεράσματα.....	69
5.5	Επίλυση του προβλήματος του περιοδούντος πωλητή με επιταχυνόμενο δίκτυο Hopfield και πολλαπλασιαστές Lagrange.....	70
5.5.1	Μεθοδολογία	70
5.5.2	Αποτελέσματα	70
5.5.3	Συμπεράσματα.....	71
5.6	Βελτίωση του μοντέλου Hopfield στο πρόβλημα του περιοδούντος πωλητή : Μια στρατηγική διαίρει και βασίλευε.....	71
5.6.1	Θεωρητικό υπόβαθρο.....	72
5.6.2	Η προτεινόμενη στρατηγική διαίρει και βασίλευε.....	72
5.6.3	Παραμετροποίηση των δικτύων Hopfield.....	73
5.6.4	Πειραματικά αποτελέσματα	74
5.6.5	Σύγκριση με την παραδοσιακή προσέγγιση	74
5.6.6	Υπολογιστική απόδοση.....	75
5.6.7	Συμπεράσματα.....	75
Κεφάλαιο 6ο:	Πρακτική υλοποίηση του δικτύου Hopfield για την επίλυση του προβλήματος περιοδούντος πωλητή.....	76
6.1	Εισαγωγή	76
6.2	Υλοποίηση τπρακτικού σκέλους της εργασίας, μεθοδολογία και προετοιμασία δεδομένων	77
6.3	Συλλογή και προετοιμασία δεδομένων.....	77
6.4	Προσαρμογή του κώδικα για επιλογή πόλεων και σεναρίων.....	78
6.5	Τεχνικές λεπτομέρειες της υλοποίησης	79
6.6	Βελτίωση του αλγορίθμου προσομοίωσης για οπτικοποίηση της διαδικασίας.....	80
6.7	Σύγκριση αρχικού και τροποποιημένου αλγόριθμου.....	80
6.8	Ανάπτυξη συστήματος οπτικοποίησης σε πραγματικό χρόνο	80
6.9	Αρχιτεκτονική συστήματος οπτικοποίησης	81
6.10	Κύρια χαρακτηριστικά του συστήματος οπτικοποίησης	81
6.11	Τεχνικές λεπτομέρειες υλοποίησης	82

ειγμα εκτέλεσης	82
6.13 Σύνοψη του συστήματος οπτικοποίησης	8Error! Bookmark not defined.
Κεφάλαιο 7ο: Πειραματικά αποτελέσματα και συγκριτική αξιολόγηση	84
7.1 Εισαγωγή	84
7.2 Πειραματική διαδικασία : Παράμετροι υλοποίησης	84
7.3 Scheme: Divide and Conquer.....	85
7.4 Η σημασία του seed στα δίκτυα Hopfield.....	85
7.5 Αποτελέσματα.....	85
7.6 Λεπτομερή αποτελέσματα ανα σενάριο.....	86
7.6.1 Σενάριο N=5	86
7.6.2 Σενάριο N=10	89
7.6.3 Σενάριο N=20	92
7.6.4 Σενάριο N=50.....	95
7.7 Ανάλυση της συμπεριφοράς των παραμέτρων.....	98
7.8 Σημαντικότητα παραμέτρων	98
7.9 Επίλογος και συμπεράσματα.....	98
7.10 Θεωρητικές επιβεβαιώσεις.....	99
7.11 Πρακτικές εφαρμογές	99
7.12 Μελλοντικές βελτιώσεις	99
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	100

Κατάλογος Σχημάτων

Σχήμα 1.1: Περιοδεία 47 πόλεων από το Commis-Voyageur	17
Σχήμα 1.2: Επίλυση ενός Προβλήματος Περιοδεύοντος Πωλητή.....	19
Σχήμα 1.3: Διάγραμμα Euler για τις κλάσεις P, NP, NP-πλήρη και NP-Δύσκολα	22
Σχήμα 2.1: Νευρωνικό δίκτυο-Δομή και συνδέσεις	32
Σχήμα 3.1: Δομή Νευρωνικού Δικτύου	39
Σχήμα 4.1: Δομή Δικτύου Hopfield με $n=4$	51
Σχήμα 4.2: Άλλη παράσταση της δομής του δικτύου	51
Σχήμα 4.3: Αναπροσαρμογή των νευρώνων.....	53

Κατάλογος εικόνων

Εικόνα 6.1:Γραφικό περιβάλλον για χρήστη στο matlab	79
Εικόνα 7.1: Τελευταία επανάληψη της φάσης 1 για 5 πόλεις	87
Εικόνα 7.2: Τελευταία επανάληψη της φάσης 2 για 5 πόλεις	87
Εικόνα 7.3: Τελική διαδρομή για 5 πόλεις.....	88
Εικόνα 7.4: Τελική διαδρομή με χρήση μεθόδου Simplex για 5 πόλεις.....	88
Εικόνα 7.5: Τελευταία επανάληψη της φάσης 1 για 10 πόλεις	90
Εικόνα 7.6: Τελευταία επανάληψη της φάσης 2 για 10 πόλεις	90
Εικόνα 7.7: Τελική διαδρομή για 10 πόλεις.....	91
Εικόνα 7.8: Τελική διαδρομή με χρήση μεθόδου Simplex για 10 πόλεις	91
Εικόνα 7.9: Τελευταία επανάληψη της φάσης 1 για 20 πόλεις	93
Εικόνα 7.10: Τελευταία επανάληψη της φάσης 2 για 20 πόλεις	93
Εικόνα 7.11: Τελική διαδρομή για 20 πόλεις.....	94
Εικόνα 7.12: Τελική διαδρομή με χρήση μεθόδου Simplex για 20 πόλεις	94
Εικόνα 7.13: Τελευταία επανάληψη της φάσης 1 για 50 πόλεις	96
Εικόνα 7.14: Τελευταία επανάληψη της φάσης 2 για 50 πόλεις	96
Εικόνα 7.15: Τελική διαδρομή για 50 πόλεις.....	97
Εικόνα 7.16: Τελική διαδρομή με χρήση μεθόδου Simplex για 50 πόλεις	97

Κατάλογος Πινάκων

Πίνακας 1.1: Σύγκριση βασικών μεθόδων επίλυσης του προβλήματος του περιοδεύοντος πωλητή	25
Πίνακας 4.2: Αναπαράσταση λύσης με 10 πόλεις.....	58
Πίνακας 7.1:Παράμετροι αλγορίθμου.....	84
Πίνακας 7.2:Παράμετροι δικτύου (Αυτόματα υπολογιζόμενες)	84
Πίνακας 7.3: Παράμετροι Divide and Conquer.....	85
Πίνακας 7.4:Συγκεντρωτικός πίνακας	85

Συντομογραφίες

TSP	Πρόβλημα Περιοδεύοντος Πωλητή
TSPLIB	Βιβλιοθήκες TSP
RNN	Αναδρομικό Νευρωνικό Δίκτυο
LSTM	Δίκτυα Μακράς Βραχυπρόθεσμης Μνήμης
GRU	Πυλωτές Αναδρομικές Μονάδες
ESN	Δίκτυα Κατάστασης Ηχούς
BPTT	Οπισθοδιάδοση Διαμέσου Χρόνου
CHN	Συνεχές Δίκτυο Hopfield
CHNN	Συνεχές Νευρωνικό Δίκτυο Hopfield
DHN	Διακριτά Δίκτυα Hopfield
CNO	Συνεργατική Νευροδυναμική Βελτιστοποίηση
PSO	Βελτιστοποίηση Σμήνους Σωματιδίων
QUBO	Τετραγωνική Δυαδική Βελτιστοποίηση χωρίς Περιορισμούς
AALHNN	Επιταχυνόμενο Επαυξημένο Δίκτυο Hopfield με Πολλαπλασιαστές Lagrange
HNN	Νευρωνικό Δίκτυο Hopfield
NLP	Επεξεργασία Φυσικής Γλώσσας
CNN	Συνελκτικά Νευρωνικά Δίκτυα
GNSS	Παγκόσμια Δορυφορικά Συστήματα Πλοήγησης
VRP	Πρόβλημα Δρομολόγησης Οχημάτων
ACO	Βελτιστοποίηση Αποικίας Μυρμηγκιών
PCB	Πλακέτα Τυπωμένων Κυκλωμάτων
GPU	Μονάδα Επεξεργασίας Γραφικών
API	Διεπαφή Προγραμματισμού Εφαρμογών

CSV	Τιμές Διαχωρισμένες με Κόμμα
NP	Μη Ντετερμινιστικό Πολυωνυμικό
B&B	Διακλάδωση και Οριοθέτηση
ΓΑ	Γενετικοί Αλγόριθμοι
MLP	Πολυστρωματικός Αισθητήρας
AI	Τεχνητή Νοημοσύνη
GPU	Μονάδα Επεξεργασίας Γραφικών
CPU	Κεντρική Μονάδα Επεξεργασίας
BERT	Αναπαραστάσεις αμφίδρομου κωδικοποιητή από μετασχηματιστές
GPT	Γεννήτρια προεκπαιδευμένου μετασχηματιστή
Adam	Adaptive Moment Estimation (αλγόριθμος βελτιστοποίησης)
RMSprop	Root Mean Square Propagation (αλγόριθμος βελτιστοποίησης)
AdaGrad	Adaptive Gradient Algorithm (αλγόριθμος βελτιστοποίησης)
MATLAB	Εργαστήριο Πινάκων
URL	Ενιαίος Εντοπιστής Πόρων
SVG	Κλιμακώσιμα Διανυσματικά Γραφικά
DNA	Deoxyribonucleic Acid (αν αναφέρεται σε βιοπληροφορική)
IEEE	Ινστιτούτο Ηλεκτρολόγων και Ηλεκτρονικών Μηχανικών
ACM	Ένωση Υπολογιστικών Μηχανημάτων
BSD	Berkeley Software Distribution (για άδειες λογισμικού)
HTTP/HTTPS	Πρωτόκολλο μεταφοράς υπερκειμένου
XML	Επεκτατική γλώσσα σήμανσης
JSON	Σημειογραφία αντικειμένου JavaScript

Κεφάλαιο 1ο: Το Πρόβλημα του Περιοδεύοντος Πωλητή

1.1 Εισαγωγή

Το Πρόβλημα του Περιοδεύοντος Πωλητή (Travelling Salesman Problem – TSP) αποτελεί ένα δύσκολο πρόβλημα συνδυαστικής βελτιστοποίησης που μελετάται κυρίως στην επιχειρησιακή έρευνα και τη θετική πληροφορική. Στο Πρόβλημα του Περιοδεύοντος Πωλητή δίνεται ένα σύνολο από κόμβους (συχνά πόλεις) για τους οποίους είναι γνωστές οι μεταξύ τους αποστάσεις. Σε μία πιο γενική περίπτωση, αντί για αποστάσεις χρησιμοποιούνται βάρη που ενώνουν τον κάθε κόμβο με τον επόμενο. Τα βάρη αυτά μπορεί να είναι πέρα από την απόσταση μεταξύ δύο κόμβων μετρημένη σε μονάδες μήκους ή σε μονάδες χρόνου, το κόστος μεταφοράς για να μεταβεί κάποιος από τον έναν κόμβο στον άλλον ή οτιδήποτε άλλο. Έστω ότι μιλάμε για αποστάσεις, τα ίδια ισχύουν σε κάθε περίπτωση.

Στόχος του προβλήματος είναι να βρεθεί η συντομότερη διαδρομή που πρέπει να ακολουθηθεί ώστε ο πωλητής να περάσει μία μόνο φορά από κάθε πόλη μιας δοθείσας λίστας πόλεων και να επιστρέψει πάλι στην πόλη από την οποία ξεκίνησε. Η πρώτη επίσημη διατύπωση του ζητήματος της εύρεσης της βέλτιστης διαδρομής για έναν περιοδεύοντα πωλητή χρονολογείται από τη δεκαετία του 1930, εξελισσόμενο έκτοτε σε ένα από τα θεμελιώδη ζητήματα στην επιστήμη της βελτιστοποίησης.

Το πρόβλημα αυτό λειτουργεί ως πρότυπο για τη μελέτη πολύπλοκων ζητημάτων βελτιστοποίησης και έχει αποτελέσει θεμέλιο για την ανάπτυξη μεθόδων που χρησιμοποιούνται ευρέως και σε άλλους κλάδους της επιστήμης. Επιπλέον, η επίδοση των διαφόρων αλγοριθμικών προσεγγίσεων στην επίλυση του TSP χρησιμοποιείται συχνά ως δείκτης αξιολόγησης νέων μεθόδων βελτιστοποίησης. Παρά την εγγενή υπολογιστική του δυσκολία, η ερευνητική κοινότητα έχει αναπτύξει ένα ευρύ φάσμα μεθοδολογιών που επιτρέπουν την αποτελεσματική προσέγγιση του προβλήματος σε πρακτικές εφαρμογές.

1.2 Ιστορική αναδρομή

Το έργο *Der Handlungsreisende – wie er sein soll und was er zu thun hat, um Aufträge zu erhalten und eines glücklichen Erfolgs in seinen Geschäften gewiß zu sein* το οποίο εκδόθηκε το 1832 από τον Alten Commis-Voyager [2] παρότι δε χρησιμοποιεί ακόμη αλγόριθμους ή βελτιστοποιήσεις προσπαθεί ωστόσο να βρει αποτελεσματικά οργανωμένες διαδρομές ταξιδιών. Επιπλέον, εμφανίζεται σε πρώιμο στάδιο η ανάγκη ο περιπλανώμενος πωλητής μέσω μιας βέλτιστης διαδρομής να επισκέπτεται κάθε τοποθεσία μία μόνο φορά από άποψη οικονομική αλλά και εξοικονόμησης χρόνου.

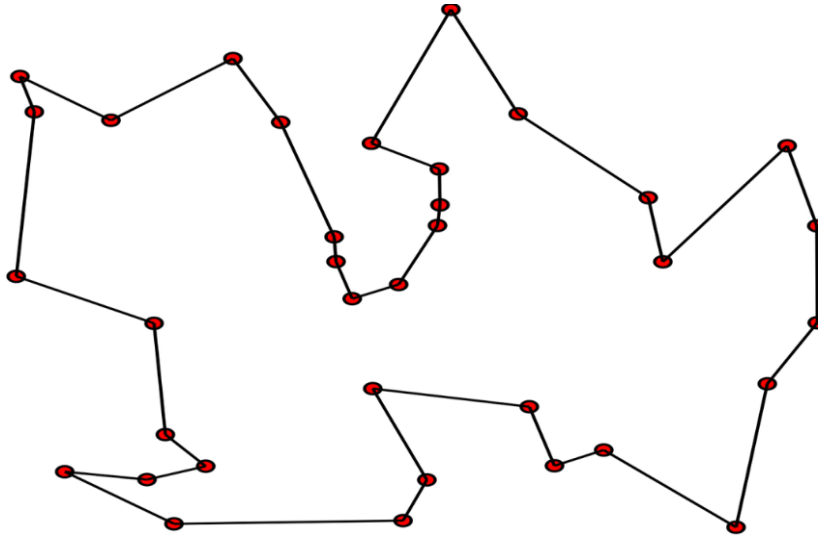
Το βιβλίο περιλαμβάνει πέντε διαδρομές μέσω περιοχών της Γερμανίας και της Ελβετίας. Τέσσερις από αυτές τις διαδρομές περιλαμβάνουν επισκέψεις επιστροφής σε μια προηγούμενη πόλη που χρησιμεύει ως βάση για αυτό το μέρος του ταξιδιού. Η πέμπτη διαδρομή, ωστόσο, είναι πράγματι μια περιοδεία πωλητή, όπως περιγράφεται στο βιβλίο του Alexander Schrijver [3].

της Ευρώπης και της Αμερικής, ειδικότερα αφότου σε ένα συνέδριο στην Santa Monica προσφέρθηκαν βραβεία σε όσους κατάφεραν να συνεισφέρουν στην επίλυση του προβλήματος .

Οι κύριες συνιστώσες για σημερινές πιο επιτυχείς προσεγγίσεις για δύσκολα συνδυαστικά προβλήματα βελτιστοποίησης είναι οι αλγόριθμοι Ευρετικής Αναζήτησης, ο Γραμμικός Προγραμματισμός και οι αλγόριθμοι Branch and Bound (B&B)[11], οι οποίοι, παρότι αναπτύχθηκαν αρχικά για την επίλυση του TSP, εφαρμόζονταν κυρίως σε πιο πρακτικά προβλήματα.. Κατά τη διάρκεια των επόμενων δεκαετιών, το πρόβλημα μελετήθηκε περαιτέρω από πολλούς ερευνητές διαφόρων κλάδων των Μαθηματικών, των Υπολογιστών, της Χημείας, της Φυσικής και της Επιχειρησιακής Έρευνας. Ο Richard Karp[12], έδειξε το 1972, ότι το πρόβλημα με τον κύκλο του Hamilton ήταν ένα NP-πλήρες πρόβλημα (δηλαδή μπορεί να επαληθευτεί σε πολυωνυμικό χρόνο), από το οποίο προκύπτει για το TSP ότι ανήκει στα NP-δύσκολα προβλήματα λόγω της πολυπλοκότητας του. Αυτό παρέχει μια μαθηματική εξήγηση για την προφανή υπολογιστική δυσκολία εύρεσης βέλτιστων περιηγήσεων. Αργότερα, αναπτύχθηκαν νέες αλγοριθμικές τεχνικές και εφαρμόστηκαν στο TSP για να αποδείξουν την αποτελεσματικότητά τους. Παραδείγματα τέτοιων τεχνικών, πέρα από τον αλγόριθμο B&B είναι η μέθοδος πολλαπλασιαστών του Lagrange, ο αλγόριθμος και η ευριστική συνάρτηση των Lin-Kernighan[13], η προσομοιωμένη απόπτωση και το πεδίο των συνδυαστικών πολυέδρων για δύσκολα συνδυαστικά προβλήματα βελτιστοποίησης.

Μεγάλη πρόοδος επιτεύχθηκε στα τέλη των δεκαετιών 1970 και 1980, όταν οι Grotschel, Padberg, Rinaldi[14] και άλλοι, κατάφεραν να επιλύσουν ακριβώς περιπτώσεις με έως και 2.392 πόλεις, χρησιμοποιώντας τις μεθόδους Cutting Plane και B&B. Κατά την δεκαετία του 1990, οι Applegate, Bixby, Chvatal και Cook[15] ανέπτυξαν το πρόγραμμα ‘Concorde TSP Solver’, το οποίο έχει χρησιμοποιηθεί σε πολλές πρόσφατες καταγεγραμμένες λύσεις ενώ είναι διαθέσιμο δωρεάν για ακαδημαϊκούς σκοπούς. Ο Gerhard Reinelt[16] το 1991 δημοσίευσε το TSPLIB, το οποίο είναι μια συλλογή από συγκριτικές αξιολογήσεις περιπτώσεων διαφορετικής δυσκολίας και έχει χρησιμοποιηθεί επίσης από πολλούς ερευνητές για σύγκριση των αποτελεσμάτων. Το 2006, ο Cook[17] και οι συνεργάτες του, υπολόγισαν μία περίπτωση βέλτιστης διαδρομής διαμέσου 85.900 πόλεων, η οποία αποτελεί επί του παρόντος τη μεγαλύτερη λυμένη περίπτωση. Για πολλές άλλες περιπτώσεις, με εκατομμύρια πόλεις, οι λύσεις που μπορούν να βρεθούν εγγυώνται μόνο σε ποσοστό 1% μια βέλτιστη διαδρομή.

Στο Σχήμα 1.2 που ακολουθεί φαίνεται το παράδειγμα επίλυσης ενός Προβλήματος Περιοδεύοντος Πωλητή.



Σχήμα 1.2: Επίλυση ενός Προβλήματος Περιοδεύοντος Πωλητή.

1.2.1 Μαθηματική απεικόνιση του προβλήματος

Όπως προαναφέρθηκε, το Πρόβλημα του Περιοδεύοντος Πωλητή, υπό μία έννοια, αποτελεί ένα εύκολο πρόβλημα καθώς για ένα δοσμένο σύνολο n πόλεων υπάρχει πεπερασμένο πλήθος πιθανών διαδρομών, μία εκ των οποίων είναι και η ολικά βέλτιστη. Η δυσκολία του προβλήματος έγκειται στο γεγονός ότι από ένα πλήθος πόλεων και πάνω, ο αριθμός όλων των πιθανών διαδρομών είναι αρκετά μεγάλος για να τις εξετάσουμε όλες μία προς μία.

Πιο συγκεκριμένα, έστω ότι αναζητάμε τη λύση ενός προβλήματος TSP για n πόλεις. Ξεκινώντας από την πόλη που ορίζεται πρώτη στη διαδρομή (αρχική πόλη) έχουμε $n-1$ επιλογές για το ποια θα είναι η δεύτερη πόλη της διαδρομής μας, $n-2$ επιλογές για την τρίτη πόλη και ούτω κάθε εξής. Κατά συνέπεια, υπάρχουν συνολικά $(n-1) \cdot (n-2) \cdot (n-3) \cdot \dots \cdot 3 \cdot 2 \cdot 1 = (n-1)!$ πιθανές διαδρομές που αποτελούν όλες λύσεις του προβλήματος. Στο TSP αναζητείται η ελάχιστη από όλες αυτές τις λύσεις. Εφόσον το πρόβλημα είναι συμμετρικό, δηλαδή οι αποστάσεις μεταξύ των πόλεων i και j ισούνται με τις αποστάσεις μεταξύ των j και i για κάθε ζεύγος πόλεων i, j του προβλήματος, οι πιθανές λύσεις είναι $(n-1)! / 2$.

Για να κατανοήσουμε το μέγεθος του προβλήματος, εξετάζουμε την παρακάτω περίπτωση. Έστω ότι έχουμε μια λίστα με 30 πόλεις (δηλαδή $n = 30$). Σε αυτή την περίπτωση, το TSP έχει $30! / 2 = 1,3253 \times 10^{32}$ πιθανές λύσεις. Ακόμη και με τους πιο ισχυρούς σύγχρονους υπερυπολογιστές, η εξαντλητική αναζήτηση παραμένει ανέφικτη.

Για παράδειγμα, ο Frontier[18], που βρίσκεται στο Oak Ridge National Laboratory και κατέχει την πρώτη θέση των ταχύτερων υπερυπολογιστών παγκοσμίως από το 2022, προσφέρει υπολογιστική ισχύ που ξεπερνά τα 1,102 exaflops (10^{18} υπολογισμούς ανά δευτερόλεπτο). Ακόμη και με αυτή την εντυπωσιακή υπολογιστική δύναμη, αν υποθέσουμε ότι μπορούμε να αξιολογήσουμε μία πιθανή

Κεφάλαιο 1ο

διαδρομή με έναν υπολογισμό, θα χρειαζόταν περίπου 3,8 δισεκατομμύρια χρόνια για να εξεταστούν όλες οι πιθανές διαδρομές για 30 πόλεις.

Το παραπάνω παράδειγμα, καταδεικνύει ότι η εξαντλητική αναζήτηση δεν αποτελεί βιώσιμη προσέγγιση για το TSP, ακόμη και με την πιο προηγμένη τεχνολογία υπολογιστών, γεγονός που τονίζει την ανάγκη για αποτελεσματικούς ευρετικούς και προσεγγιστικούς αλγορίθμους.

Το Πρόβλημα του Περιοδεύοντος Πωλητή, στη γενική του μορφή μπορεί να περιγράψει με το παρακάτω μαθηματικό μοντέλο[19]:

$$\min \sum_{i=1}^n \sum_{\substack{j=1 \\ j \neq i}}^n d_{ij} x_{ij} \quad (1.1)$$

$$\sum_{\substack{i=1 \\ i \neq j}}^n x_{ij}=1 \text{ για } i=1 \dots n \quad \sum_{\substack{j=1 \\ j \neq i}}^n x_{ij}=1 \text{ για } i=1 \dots n \quad (1.2)$$

$$\sum_{\substack{i=1 \\ i \neq j}}^n x_{ij}=1 \text{ για } i=1 \dots n \quad \sum_{\substack{j=1 \\ j \neq i}}^n x_{ij}=1 \text{ για } i=1 \dots n \quad (1.3)$$

$$y_i - y_j + nx_{ij} \leq n - 1 \text{ για } 2 \leq i \neq j \leq \dots n \quad (1.4)$$

$$x_{ij} \in \{0,1\} \text{ για } i, j = 1 \dots n \quad y_i \in N_0 \text{ για } i = 1 \dots n \quad (1.5)$$

Όπου:

n : ο αριθμος των κόμβων του προβλήματος

d_{ij} : η απόσταση μεταξύ των κομβων i και j

x_{ij} : μεταβλητή που λαμβάνει την τιμή 1 όταν η σύνδεση μεταξύ των κόμβων i και j ανήκει στο Χαμιλτονιανό κύκλο

Οι ισότητες (2) προβλέπουν ότι κάθε κόμβος μπορεί να εισαχθεί και να αφαιρεθεί μόνο μια φορά. Ο περιορισμός (3) προβλέπει ότι ο χαμιλτονιακός κύκλος είναι μόνο μια διαδρομή και όχι ένα άθροισμα μικρότερων κύκλων[19].

1.2.2 NP-Πληρότητα

Η θεωρητική επιστήμη των υπολογιστών ταξινομεί τα υπολογιστικά προβλήματα με βάση την εγγενή τους πολυπλοκότητα. Ένα σημαντικό κριτήριο διάκρισης είναι ο απαιτούμενος χρόνος υπολογισμού για την επίλυσή τους. Ορισμένα προβλήματα επιδέχονται αλγοριθμικές λύσεις που λειτουργούν σε πολυωνυμικό χρόνο, δηλαδή ο χρόνος εκτέλεσης της χειρότερης περίπτωσης μπορεί να εκφραστεί ως $T(n) = O(n^k)$, όπου k είναι μια θετική σταθερά και n το μέγεθος εισόδου. Αυτά τα προβλήματα χαρακτηρίζονται ως "ευεπίλυτα" ή "τρακτέα" (tractable).

Αντιθέτως, υπάρχει μια κατηγορία προβλημάτων για τα οποία δεν έχουν ανακαλυφθεί αλγόριθμοι πολυωνυμικού χρόνου, ή οι γνωστοί αλγόριθμοι απαιτούν υπερπολυωνυμικό χρόνο (όπως εκθετικό ή παραγοντικό). Αυτά τα προβλήματα χαρακτηρίζονται ως "δυσεπίλυτα" ή "μη τρακτέα" (intractable) και αντιπροσωπεύουν ορισμένες από τις μεγαλύτερες προκλήσεις στην επιστήμη των υπολογιστών.

Εντός του φάσματος των υπολογιστικών προβλημάτων, υπάρχει μια ιδιαίτερα σημαντική κατηγορία γνωστή ως NP-πλήρη προβλήματα (NP-complete), τα οποία χαρακτηρίζονται από την εγγενή υπολογιστική τους δυσκολία. Παρά την εκτεταμένη έρευνα δεκαετιών, για κανένα από τα NP-πλήρη προβλήματα δεν έχει ανακαλυφθεί αλγόριθμος που να το επιλύει σε πολυωνυμικό χρόνο. Ωστόσο, είναι κρίσιμο να τονιστεί ότι η απουσία τέτοιων αλγορίθμων δεν αποτελεί μαθηματική απόδειξη της μη ύπαρξής τους—παραμένει ένα ανοιχτό ερευνητικό ζήτημα. Το πρόβλημα του περιοδεύοντος πωλητή ανήκει σε αυτή την κατηγορία, γεγονός που εξηγεί την υπολογιστική του πρόκληση.

Το ερώτημα αν τα προβλήματα της κλάσης P (αυτά που επιλύονται σε πολυωνυμικό χρόνο) ταυτίζονται με τα προβλήματα της κλάσης NP (αυτά των οποίων η λύση μπορεί να επαληθευτεί σε πολυωνυμικό χρόνο) αποτελεί ένα από τα σημαντικότερα ανοιχτά ζητήματα στην επιστήμη των υπολογιστών. Το ερώτημα αυτό, γνωστό ως "P=NP?" [20], διατυπώθηκε επίσημα το 1971 από τον Stephen Cook και παραμένει άλυτο μέχρι σήμερα, παρά τις εκτεταμένες προσπάθειες της επιστημονικής κοινότητας. Η επίλυσή του θα είχε επαναστατικές συνέπειες τόσο στη θεωρητική όσο και στην εφαρμοσμένη πληροφορική, καθώς και σε πολλούς άλλους επιστημονικούς κλάδους.

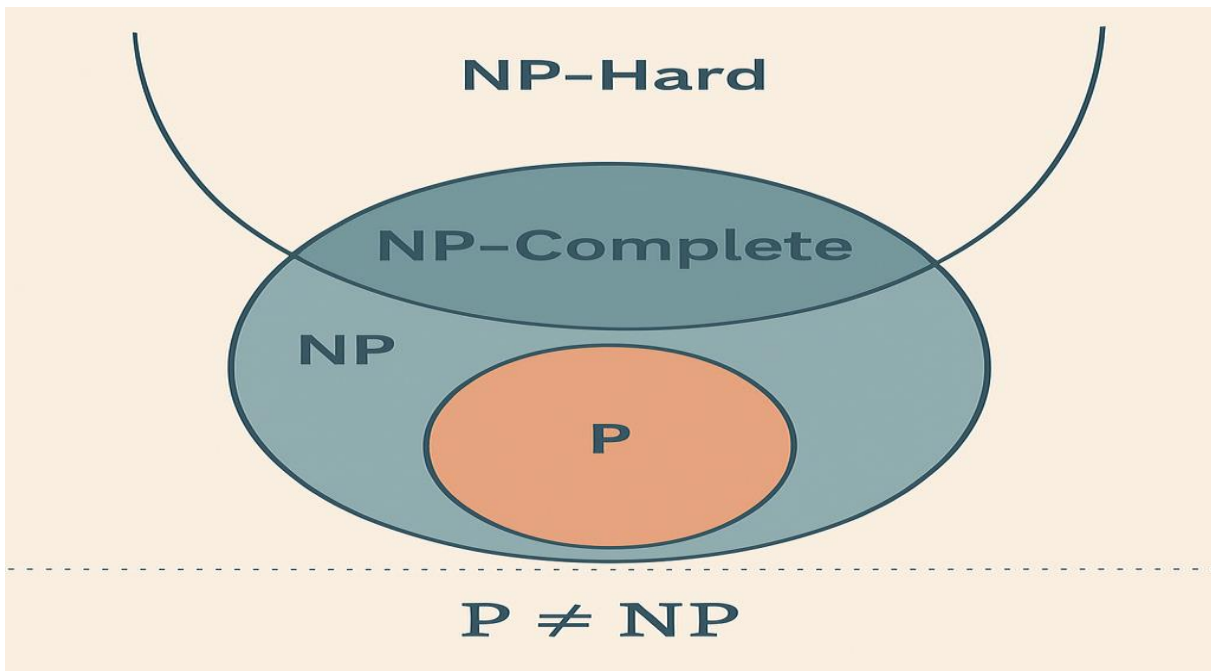
Γενικά, έχουμε διαχωρίσει τα προβλήματα σε κλάσεις, οι οποίες χαρακτηρίζονται για την πολυπλοκότητα τους [21]. Οι κλάσεις πολυπλοκότητας που μας απασχολούν ονομάζονται P, NP και NP-πλήρη. Η κλάση P αναφέρεται στα προβλήματα τα οποία υπάρχει αλγόριθμος πολυωνυμικού χρόνου και δε χρειάζεται να επαληθευτεί από μια μηχανή Turing η ορθότητα του αλγορίθμου. Αυτού του είδους προβλήματα θεωρούνται υποσύνολο της κλάσης NP, με ανοικτό ερώτημα να παραμένει αν η κλάση P αποτελεί γνήσιο υποσύνολο της κλάσης NP ή όχι.

Η κλάση NP περιέχει τα προβλήματα τα οποία μπορούμε να επαληθεύσουμε την επίλυση τους σε πολυωνυμικό χρόνο. Αναγκαίο για την επαλήθευση είναι να δοθεί ένα πιστοποιητικό μιας λύσης ώστε να μπορέσει να επαληθευτεί η ορθότητα της, κάτι που απαιτεί πολυωνυμικό χρόνο σε σχέση με το μέγεθος της εισόδου.

Η κλάση NP (αλλιώς NP-complete) αποτελείται από τα δυσκολότερα προβλήματα της κλάσης NP. Θεωρητικά, για να ανήκει ένα πρόβλημα στην κλάση NP-πλήρη πρέπει να είναι τουλάχιστον το ίδιο δύσκολο με τα υπόλοιπα προβλήματα της κλάσης NP.

Χρησιμοποιώντας παράλληλα με τον ορισμό του ζητήματος P=NP, αν για ένα πρόβλημα της κλάσης NP βρεθεί αλγόριθμος επίλυσης του με πολυωνυμικό χρόνο, τότε όλα τα άλλα προβλήματα της κλάσης

NP λύνονται σε πολυωνυμικό χρόνο, κάτι που έπειτα από πολλή μελέτη δεν έχει βρεθεί, χωρίς φυσικά να σημαίνει ότι δεν υπάρχει.



Σχήμα 1.3: Διάγραμμα Euler για τις κλάσεις P, NP, NP-πλήρη και NP-Δύσκολα

1.2.3 Τρόποι επίλυσης

Η αντίθεση μεταξύ της εννοιολογικής απλότητας του TSP και της αλγοριθμικής του πολυπλοκότητας έχει καταστήσει το πρόβλημα αντικείμενο εκτεταμένης επιστημονικής έρευνας διαχρονικά. Μια κρίσιμη παρατήρηση είναι ότι ο χώρος αναζήτησης λύσεων διογκώνεται με εκθετικό ρυθμό σε σχέση με τον αριθμό των πόλεων, καθιστώντας την εύρεση της απόλυτα βέλτιστης λύσης υπολογιστικά ανέφικτη για προβλήματα μεγάλης κλίμακας.

Αυτός ο εγγενής περιορισμός έχει στρέψει το ερευνητικό ενδιαφέρον προς την ανάπτυξη ευρετικών (heuristic) αλγορίθμων, οι οποίοι θυσιάζουν την εγγύηση βελτιστότητας προς όφελος της υπολογιστικής αποδοτικότητας. Αυτές οι προσεγγίσεις επιδιώκουν να εντοπίσουν υψηλής ποιότητας λύσεις συχνά πολύ κοντά στη βέλτιστη σε πρακτικά εφαρμόσιμο χρονικό διάστημα, καθιστώντας εφικτή την αντιμετώπιση πραγματικών περιπτώσεων του προβλήματος.

Επομένως, μπορούμε να χωρίσουμε τους αλγόριθμους σε κατηγορίες ανάλογα με την επίλυση τους σε κατασκευαστικούς αλγόριθμους, αλγόριθμους βελτίωσης καθώς και υβριδικούς αλγόριθμους.

Σκοπός των κατασκευαστικών αλγορίθμων είναι να δημιουργήσουν τη λύση από την αρχή, συνήθως τοποθετώντας μια πόλη κάθε φορά μέχρι να δημιουργηθεί μια ολοκληρωμένη λύση. Οι αλγόριθμοι βελτιστοποίησης παίρνουν μια αρχική λύση την οποία βελτιώνουν σταδιακά. Τέλος, οι υβριδικοί αλγόριθμοι δέχονται μια αρχικά αποδεκτή λύση από κάποιο κατασκευαστικό αλγόριθμο την οποία βελτιώνουν με βάση κάποιο αλγόριθμο βελτιστοποίησης. Παρακάτω θα αναφερθούμε σε μερικούς από

τους πιο σημαντικούς αλγόριθμους.

1.3 Αλγόριθμος πλησιέστερου γείτονα

Ο αλγόριθμος πλησιέστερου γείτονα (Nearest Neighbor Algorithm) αποτελεί έναν θεμελιώδη κατασκευαστικό ευρετικό αλγόριθμο, ο οποίος ξεχωρίζει για την εννοιολογική του διαύγεια και την απλότητα υλοποίησής του. Η μεθοδολογία του προσομοιώνει μια διαισθητική ανθρώπινη προσέγγιση: ξεκινώντας από μια αυθαίρετη πόλη, επιλέγει επαναληπτικά ως επόμενο προορισμό την πλησιέστερη μη επισκεφθείσα πόλη, μέχρι να δημιουργηθεί μια πλήρης διαδρομή.

Η χρονική πολυπλοκότητα του αλγορίθμου είναι $O(N^2)$, όπου N αντιπροσωπεύει τον αριθμό των πόλεων, γεγονός που τον καθιστά υπολογιστικά αποδοτικό συγκριτικά με άλλες προσεγγίσεις. Ωστόσο, αυτή η αποδοτικότητα αντισταθμίζεται από την περιορισμένη αποτελεσματικότητα του αλγορίθμου όσον αφορά την ποιότητα των παραγόμενων λύσεων. Η άπληστη (greedy) φύση του αλγορίθμου τον καθιστά επιρρεπή στην παγίδευση σε τοπικά βέλτιστα, με αποτέλεσμα να παράγει ικανοποιητικές λύσεις μόνο σε περιπτώσεις με περιορισμένο αριθμό πόλεων.

1.3.1 2-opt,3-opt

Οι αλγόριθμοι αυτοί ανήκουν στην κατηγορία αλγορίθμων βελτιστοποίησης. Επειδή μοιάζουν μεταξύ τους θα τους αναφέρουμε στην ίδια παράγραφο. Οι αλγόριθμοι αυτοί δέχονται σαν είσοδο μια αρχική λύση η οποία μπορεί να έχει δημιουργηθεί με όποιο τρόπο εμείς θέλουμε (για παράδειγμα με του πλησιέστερου γείτονα). Στη συνέχεια αυτό που κάνουν είναι να προσπαθούν σε κάθε βήμα να βελτιώνουν αυτή τη λύση έως ότου φτάσουν σε σημείο να μην μπορεί να βελτιωθεί άλλο, φτάνουμε δηλαδή σε τοπικό ελάχιστο. Ο τρόπος λειτουργίας τους είναι ο εξής: σε κάθε βήμα του ο 2-opt διαγράφει δύο ακμές χωρίζοντας τη διαδρομή σε δύο και στη συνέχεια τις ενώνει με τον αντίθετο τρόπο. Εάν η καινούργια διαδρομή είναι μικρότερη από την παλιά τότε την κρατάει, αλλιώς συνεχίζει με τις επόμενες. Στην 3-opt η διαδικασία είναι παρόμοια με αυτήν του 2-opt εκτός του ότι σε κάθε βήμα χωρίζουμε την διαδρομή με τρεις διαδρομές και τις ενώνουμε πάλι ώστε να μικρύνει η συνολική διαδρομή. Εάν το τοπικό ελάχιστο (τελική λύση) είναι ταυτόχρονα και ολικό ελάχιστο τότε η λύση είναι η βέλτιστη.

Στην κατηγορία των αλγορίθμων βελτίωσης ανήκουν εν μέρη, και οι γενετικοί αλγόριθμοι, οι αποκοίες μυρμηγκιών και τα νευρωνικά δίκτυα. Τα μοντέλα αυτά δείχνουν πως μπορούν διαδικασίες από την πραγματική ζωή με κατάλληλη μετατροπή να χρησιμοποιηθούν ως τρόποι επίλυσης σε προβλήματα υπολογιστών, και θα τα αναφέρουμε ξεχωριστά.

1.3.2 Γενετικοί αλγόριθμοι (ΓΑ)

Οι ΓΑ είναι ευρετικοί αλγόριθμοι βελτιστοποίησης, που βασίζονται σε ένα γενικό πληθυσμό[23]. Ο αλγόριθμος ξεκινάει με ένα πλήθος από λύσεις (χρωμοσώματα) που αποτελούν τον πληθυσμό. Στη συνέχεια σε κάθε γενιά παίρνουμε λύσεις από προηγούμενες γενιές και δημιουργούμε έναν καινούργιο πληθυσμό. Στη διαδικασία αυτή χρησιμοποιούμε μηχανισμούς, εμπνευσμένους από τη βιολογική εξέλιξη, όπως η αναπαραγωγή, η μετάλλαξη, ο ανασυνδυασμός και η επιλογή, με σκοπό η καινούργια γενιά να είναι καλύτερη από την προηγούμενη. Οι λύσεις κάθε γενιάς περνάνε από μια συνάρτηση αξιολόγησης έτσι ώστε αυτές με τις καλύτερες τιμές να έχουν περισσότερες πιθανότητες να επιλεγθούν

για γονείς. Σκοπός είναι στο τέλος να ‘επιβιώσει’ το καλύτερο άτομο, δηλαδή να βρεθεί η καλύτερη λύση. Για το πρόβλημα του περιοδεύοντος πωλητή, το χρωμόσωμα κωδικοποιείται σαν ένας μονοδιάστατος πίνακας ο οποίος περιέχει όλες τις πόλεις που πρέπει να επισκεφτούμε. Η συνάρτηση αξιολόγησης μετράει το μήκος της διαδρομής και δίνει στο κάθε χρωμόσωμα, ανάλογα με το πόσο μικρή ή μεγάλη είναι, την κατάλληλη τιμή. Οι νέες γενιές δημιουργούνται χρησιμοποιώντας ανασυνδυασμό δυο σημείων. Οι γενετικοί αλγόριθμοι δεν εγγυώνται καλά αποτελέσματα αλλά κατά ένα μεγάλο ποσοστό τα αποτελέσματά τους είναι αρκετά καλά. Μειονέκτημα τους είναι ότι δεν μπορούμε να προβλέψουμε το χρόνο που χρειάζονται για να συγκλίνουν σε μία λύση, για αυτό σε μεγάλα προβλήματα χρησιμοποιούνται διάφορες μέθοδοι και από άποψη υλικού (πολυνημάτωση, παράλληλη επεξεργασία), και από άποψη λογισμικού (βελτίωση αρχικού πληθυσμού, ευρετικές μέθοδοι) για να επιταχύνουν τη διαδικασία.

1.3.3 Αποικία μυρμηγκιών

Η βελτιστοποίηση αποικίας μυρμηγκιών (Ant Colony Optimization - ACO) αντλεί έμπνευση από την αξιοσημείωτη ικανότητα των μυρμηγκιών να ανακαλύπτουν βέλτιστες διαδρομές μεταξύ της φωλιάς τους και των πηγών τροφής. Αυτός ο φυσικός μηχανισμός βελτιστοποίησης επιδεικνύει εντυπωσιακή προσαρμοστικότητα, επιτρέποντας στις αποικίες να αναπροσαρμόζουν τις διαδρομές τους ακόμη και όταν προκύπτουν εμπόδια ή αλλαγές στο περιβάλλον[24].

Ο αλγόριθμος ACO μιμείται αυτή τη συμπεριφορά μέσω της δημιουργίας υπολογιστικών μοντέλων τεχνητών μυρμηγκιών που εξερευνούν το χώρο λύσεων του προβλήματος. Η κίνηση των τεχνητών μυρμηγκιών καθορίζεται από δύο κύριους παράγοντες: την ευρετική πληροφορία (συνήθως την απόσταση μεταξύ των πόλεων) και το επίπεδο των φερορμονών που έχουν αποθεθεί στις διαδρομές από προηγούμενες διελεύσεις.

Κάθε τεχνητό μυρμήγκι, κατά την ολοκλήρωση μιας περιήγησης, εναποθέτει φερορμόνες σε αναλογία με την ποιότητα της διαδρομής που ανακάλυψε. Με την πάροδο των επαναλήψεων, οι βέλτιστες διαδρομές συσσωρεύουν υψηλότερες συγκεντρώσεις φερορμονών, καθιστώντας τις ελκυστικότερες για τα επόμενα μυρμήγκια. Παράλληλα, οι λιγότερο αποδοτικές διαδρομές υπόκεινται σε σταδιακή εξάτμιση των φερορμονών, μειώνοντας την πιθανότητα επιλογής τους. Αυτός ο μηχανισμός θετικής ανάδρασης οδηγεί, με την πάροδο του χρόνου, στη σύγκλιση του αλγορίθμου προς μια βέλτιστη ή σχεδόν βέλτιστη λύση του TSP.

1.3.4 Brute Force algorithm

Ο αλγόριθμος εξαντλητικής αναζήτησης (brute force) αποτελεί την απλούστερη εννοιολογικά προσέγγιση για την εξεύρεση της ακριβούς βέλτιστης λύσης του TSP. Η μεθοδολογία του χαρακτηρίζεται από την αδιαμφισβήτητη απλότητα υλοποίησης, ωστόσο συνοδεύεται από απαγορευτική υπολογιστική πολυπλοκότητα για προβλήματα μεγάλου μεγέθους, καθώς απαιτεί την απαρίθμηση και αξιολόγηση κάθε πιθανής διαδρομής.

Η υλοποίηση της εξαντλητικής αναζήτησης για το TSP ακολουθεί το εξής συστηματικό πλαίσιο. Αρχικά, τον καθορισμό ενός αυθαίρετου σημείου εκκίνησης στο γράφημα G που αναπαριστά το πρόβλημα και έπειτα την παραγωγή όλων των δυνατών περιηγήσεων με αφετηρία το επιλεγμένο σημείο, καταγράφοντας κάθε έγκυρη διαδρομή. Στη συνέχεια, ακολουθούν οι υπολογισμοί του

συνολικού κόστους (απόστασης) για κάθε παραγόμενη περιήγηση, και τέλος η επιλογή της περιήγησης με το ελάχιστο συνολικό κόστος ως βέλτιστη λύση.

Η χρονική πολυπλοκότητα του αλγορίθμου είναι $O(n!)$, όπου n ο αριθμός των πόλεων, αντικατοπτρίζοντας την ανάγκη εξέτασης όλων των πιθανών μεταθέσεων των κόμβων του γραφήματος. Παρά την εκθετική του φύση, ο αλγόριθμος έχει επιδειχθεί σημαντικές βελτιώσεις που μειώνουν το πρακτικό υπολογιστικό κόστος, όπως είναι οι τεχνικές διακλάδωσης και οριοθέτησης (branch and bound) οι οποίες επιτρέπουν την απόρριψη μη υποσχόμενων μονοπατιών, η εφαρμογή γραμμικού προγραμματισμού για τη δημιουργία αποτελεσματικών κάτω φραγμάτων καθώς και οι μεθοδολογίες δυναμικού προγραμματισμού που επιτρέπουν την επαναχρησιμοποίηση ενδιάμεσων υπολογισμών.

Αυτές οι βελτιώσεις, αν και διατηρούν τη θεωρητική πολυπλοκότητα χειρότερης περίπτωση, επιτρέπουν την ακριβή επίλυση προβλημάτων μεγαλύτερου μεγέθους από ό,τι θα ήταν εφικτό με την αμιγώς εξαντλητική προσέγγιση.

Το πρόβλημα του Περιοδεύοντος Πωλητή (TSP) έχει αποτελέσει πεδίο εκτεταμένης έρευνας λόγω της πρακτικής του σημασίας και της υπολογιστικής του πολυπλοκότητας. Στον Πίνακα 1.2 παρουσιάζεται μια συγκριτική ανάλυση των κυριότερων προσεγγίσεων που έχουν αναπτυχθεί για την επίλυσή του. Κάθε κατηγορία αλγορίθμων χαρακτηρίζεται από συγκεκριμένα πλεονεκτήματα και περιορισμούς, ενώ η καταλληλότητά τους εξαρτάται από το μέγεθος του προβλήματος και τις απαιτήσεις της εκάστοτε εφαρμογής. Είναι αξιοσημείωτο ότι ενώ οι ακριβείς αλγόριθμοι εγγυώνται την εύρεση της βέλτιστης λύσης, γίνονται πρακτικά ανεφάρμοστοι για μεγάλα προβλήματα, καθιστώντας τις ευρετικές και μετα-ευρετικές προσεγγίσεις πιο ελκυστικές εναλλακτικές λύσεις.

Πίνακας 1.1: Σύγκριση Βασικών Μεθόδων Επίλυσης του Προβλήματος του Περιοδεύοντος Πωλητή

Κατηγορία Αλγορίθμων	Παραδείγματα	Πλεονεκτήματα	Μειονεκτήματα	Πολυπλοκότητα
Ακριβείς Αλγόριθμοι	Branch and Bound, Δυναμικός Προγραμματισμός	Εγγυημένη εύρεση βέλτιστης λύσης	Εκθετικός χρόνος εκτέλεσης για μεγάλα προβλήματα	$O(n!)$ ή $O(2^n)$
Κατασκευαστικοί Ευρετικοί	Πλησιέστερου Γείτονα, Insertion Heuristics	Απλοί στην υλοποίηση, γρήγοροι	Συχνά παγιδεύονται σε τοπικά βέλτιστα	$O(n^2)$
Αλγόριθμοι Βελτίωσης	2-opt, 3-opt, k-opt, Lin-Kernighan	Βελτιώνουν υπάρχουσες λύσεις	Απαιτούν καλή αρχική λύση	$O(n^2)$ έως $O(n^k)$
Μετα-ευρετικοί	Γενετικοί Αλγόριθμοι, Προσομοιωμένη Ανόπτηση, Αποικία Μυρμηγκιών	Καλή εξερεύνηση του χώρου λύσεων	Στοχαστική φύση, απρόβλεπτη απόδοση	Εξαρτάται από παραμέτρους
Νευρωνικά Δίκτυα	Δίκτυα Hopfield, Ανταγωνιστικά δίκτυα	Φυσική παραλληλοποίηση,	Ευαίσθητα στην επιλογή παραμέτρων	Εξαρτάται από την αρχιτεκτονική

		ικανότητα αυτοοργάνωσης		
Υβριδικές Προσεγγίσεις	Συνδυασμοί των παραπάνω	Συνδυάζουν πλεονεκτήματα πολλαπλών μεθόδων	Αυξημένη πολυπλοκότητα υλοποίησης	Ποικίλλει

1.4 Που χρησιμοποιείται η Περιοδεία Περιοδεύοντος Πωλητή

Η δυνατότητα εφαρμογής του Προβλήματος του Περιοδεύοντος Πωλητή εκτείνεται πολύ πέραν της θεωρητικής του αξίας, διεισδύοντας σε ένα ευρύ φάσμα επιστημονικών πεδίων και πρακτικών εφαρμογών. Το πρόβλημα έχει καταστεί θεμελιώδης συνιστώσα σε τομείς που φαινομενικά διαφέρουν μεταξύ τους, όπως η υπολογιστική βιολογία (αλληλουχία γονιδιωμάτων), η μικροηλεκτρονική (σχεδιασμός ολοκληρωμένων κυκλωμάτων), η αστρονομία (σχεδιασμός παρατηρήσεων τηλεσκοπίων), και βεβαίως οι μεταφορές και η εφοδιαστική αλυσίδα (βελτιστοποίηση δρομολογίων οχημάτων, προγραμματισμός δρομολογίων λεωφορείων).

Η ευρύτητα και η κρισιμότητα των εφαρμογών του TSP έχει καταστήσει επιτακτική την ανάπτυξη αλγοριθμικών προσεγγίσεων που εξισορροπούν την ακρίβεια των λύσεων με την υπολογιστική αποδοτικότητα. Ιδιαίτερη έμφαση δίνεται σε μεθοδολογίες που μπορούν να παράγουν υψηλής ποιότητας λύσεις σε εύλογο χρονικό διάστημα, ακόμη και για προβλήματα μεγάλης κλίμακας όπου η εύρεση της απόλυτα βέλτιστης λύσης είναι υπολογιστικά ανέφικτη.

Στις ενότητες που ακολουθούν, εξετάζουμε χαρακτηριστικά παραδείγματα πρακτικών εφαρμογών όπου το TSP αποτελεί το μαθηματικό θεμέλιο για την επίλυση σύνθετων προβλημάτων βελτιστοποίησης σε διάφορους τομείς.

1.4.1 Κατασκευή Κυκλωμάτων

Στον τομέα της κατασκευής τυπωμένων κυκλωμάτων, το TSP βρίσκει μια ιδιαίτερη πρακτική εφαρμογή. Η διαδικασία δημιουργίας οπών στις πλακέτες κυκλωμάτων αποτελεί ένα κρίσιμο στάδιο της παραγωγής, καθώς επιτρέπει τη διασύνδεση αγωγών μεταξύ διαφορετικών επιπέδων της πλακέτας και την τοποθέτηση ολοκληρωμένων κυκλωμάτων.

Η πολυπλοκότητα αυτής της διαδικασίας έγκειται στο γεγονός ότι απαιτούνται οπές διαφορετικών διαμέτρων, οι οποίες δημιουργούνται με εξειδικευμένα τρυπάνια. Η αναγκαιότητα εναλλαγής των τρυπανιών κατά τη διάρκεια της παραγωγής εισάγει σημαντικές καθυστερήσεις, καθώς η κεφαλή του μηχανήματος πρέπει να μετακινηθεί σε συγκεκριμένες θέσεις αλλαγής εργαλείων.

Για τη βελτιστοποίηση αυτής της διαδικασίας, εφαρμόζονται αλγόριθμοι TSP που καθορίζουν τη βέλτιστη σειρά διάτρησης των οπών. Με αυτόν τον τρόπο, ελαχιστοποιείται τόσο ο αριθμός των απαιτούμενων αλλαγών τρυπανιών όσο και η συνολική διαδρομή της κεφαλής διάτρησης. Για κάθε μέγεθος τρυπανιού, αναπτύσσεται μια ξεχωριστή βελτιστοποιημένη διαδρομή, επιτυγχάνοντας έτσι σημαντική αύξηση της παραγωγικότητας και μείωση του χρόνου κατασκευής.

1.4.2 Καλωδίωση υπολογιστή

Μια ενδιαφέρουσα εφαρμογή του TSP στον τομέα της ηλεκτρονικής σχεδίασης, όπως τεκμηριώνεται από τους Lenstra και Rinnooy Kan[25], αφορά τη διασύνδεση ηλεκτρονικών εξαρτημάτων σε πλακέτες τυπωμένων κυκλωμάτων (PCB). Σε αυτό το πλαίσιο, μια σειρά από μονάδες τοποθετημένες σε μια πλακέτα διαθέτουν συγκεκριμένες ακίδες (pins) που πρέπει να διασυνδεθούν μεταξύ τους.

Η ιδιαιτερότητα αυτής της εφαρμογής έγκειται στον περιορισμό που επιβάλλει ότι σε καμία ακίδα δεν επιτρέπεται η σύνδεση περισσότερων από δύο καλωδίων. Αυτή η απαίτηση διαφοροποιεί το πρόβλημα από τα συνήθη προβλήματα δικτύωσης που αναζητούν λύσεις τύπου δέντρου Steiner, μετατρέποντάς το σε αναζήτηση Χαμιλτονιανής διαδρομής (και όχι κύκλου) με απροσδιόριστα σημεία εκκίνησης και τερματισμού.

Παρόμοια μαθηματική διατύπωση εμφανίζεται στην περίπτωση της καλωδίωσης "testbus", μια κρίσιμη διαδικασία στον έλεγχο ποιότητας των ηλεκτρονικών πλακετών. Σε αυτή την περίπτωση, απαιτείται η δημιουργία μιας διαδρομής που ξεκινά από ένα προκαθορισμένο σημείο εισόδου στην πλακέτα, διέρχεται από όλες τις μονάδες μέσω συγκεκριμένων σημείων εισόδου και εξόδου για κάθε μονάδα, και καταλήγει σε ένα καθορισμένο σημείο εξόδου.

1.4.3 Το πρόβλημα επιλογής παραγγελιών σε αποθήκες

Ένα χαρακτηριστικό παράδειγμα εφαρμογής του TSP στην εφοδιαστική αλυσίδα εντοπίζεται στη βελτιστοποίηση της συλλογής παραγγελιών (order picking) εντός αποθηκευτικών εγκαταστάσεων. Στο σενάριο αυτό, μια εισερχόμενη παραγγελία περιλαμβάνει ένα συγκεκριμένο υποσύνολο προϊόντων που βρίσκονται διάσπαρτα σε διάφορες θέσεις της αποθήκης, και ένα όχημα (π.χ. περνοφόρο ή αυτοματοποιημένο ρομποτικό σύστημα) επιφορτίζεται με τη συλλογή τους για αποστολή στον πελάτη.

Η αντιστοίχιση με το μοντέλο του TSP είναι άμεση και εύλογη: οι θέσεις των προϊόντων λειτουργούν ως οι κόμβοι του γράφου, ενώ το βάρος κάθε ακμής αντιπροσωπεύει το χρονικό κόστος μετάβασης του οχήματος μεταξύ των αντίστοιχων θέσεων. Υπό αυτό το πρίσμα, το ζητούμενο μετατρέπεται στην εύρεση της βέλτιστης διαδρομής που ελαχιστοποιεί το συνολικό χρόνο συλλογής, διατηρώντας όλους τους βασικούς περιορισμούς του κλασικού TSP.

Η βελτιστοποίηση αυτής της διαδικασίας επιφέρει σημαντικά οφέλη στην αποδοτικότητα των αποθηκευτικών συστημάτων, μειώνοντας τους χρόνους εκτέλεσης παραγγελιών και αυξάνοντας τη συνολική παραγωγικότητα της εφοδιαστικής αλυσίδας. Σε ειδικές περιπτώσεις αυτό το πρόβλημα μπορεί να λυθεί εύκολα.

1.4.4 Δρομολόγηση οχημάτων

Η δρομολόγηση οχημάτων (Vehicle Routing Problem - VRP) αποτελεί μια κρίσιμη επέκταση του TSP με εκτεταμένες πρακτικές εφαρμογές στην εφοδιαστική αλυσίδα και τη διαχείριση στόλων. Ένα χαρακτηριστικό σενάριο αποτελεί η συλλογή αλληλογραφίας από n ταχυδρομικές θυρίδες σε αστικό περιβάλλον, όπου το σύνολο των συλλογών πρέπει να ολοκληρωθεί εντός προκαθορισμένου χρονικού παραθύρου, π.χ. μιας ώρας. Σε αυτή την περίπτωση, το ζητούμενο είναι διττό: αφενός ο προσδιορισμός του ελάχιστου απαιτούμενου αριθμού οχημάτων, και αφετέρου ο σχεδιασμός βέλτιστων διαδρομών για κάθε όχημα ώστε να ελαχιστοποιηθεί ο συνολικός χρόνος συλλογής.

Σε ένα εναλλακτικό σενάριο διανομής προϊόντων, ένας κεντρικός προμηθευτής καλείται να εξυπηρετήσει n πελάτες, έκαστος με συγκεκριμένες απαιτήσεις όγκου ή βάρους εμπορευμάτων. Ο προμηθευτής διαθέτει στόλο οχημάτων περιορισμένης χωρητικότητας και επιδιώκει να προσδιορίσει: (1) τη βέλτιστη κατανομή πελατών ανά όχημα, λαμβάνοντας υπόψη τους περιορισμούς χωρητικότητας, και (2) τη βέλτιστη ακολουθία επισκέψεων για κάθε όχημα, με στόχο την ελαχιστοποίηση της συνολικής διανυόμενης απόστασης. Αυτά τα προβλήματα δρομολόγησης οχημάτων ενσωματώνουν το TSP ως επιμέρους πρόβλημα, προσθέτοντας παράλληλα επιπλέον περιορισμούς όπως χρονικά όρια εξυπηρέτησης, περιορισμούς χωρητικότητας και τη δυνατότητα χρήσης πολλαπλών οχημάτων, διαμορφώνοντας έτσι μια οικογένεια προβλημάτων ακόμη μεγαλύτερης υπολογιστικής πολυπλοκότητας.

Πολλές παραλλαγές αυτών των δύο προβλημάτων, όπου οι περιορισμοί χρόνου και χωρητικότητας συνδυάζονται, είναι κοινές σε πολλές εφαρμογές του πραγματικού κόσμου. Αυτό το πρόβλημα είναι επιλύσιμο ως TSP εάν δεν υπάρχει χρόνος και χωρητικότητα περιορισμούς και εάν ο αριθμός των φορτηγών είναι σταθερός (ας πούμε m). Σε αυτή την περίπτωση λαμβάνουμε ένα πρόβλημα m - salesmen.

1.4.5 Κρυσταλλογραφία Ακτίνων X

Στο πεδίο της κρυσταλλογραφίας ακτίνων X, η εφαρμογή του TSP εξυπηρετεί τη βελτιστοποίηση της πειραματικής διαδικασίας προσδιορισμού της δομής κρυστάλλων. Η μεθοδολογία αυτή βασίζεται στη χρήση εξειδικευμένων διαθλασιμέτρων που, σε συνδυασμό με ευαίσθητους ανιχνευτές, καταγράφουν την ένταση των ανακλώμενων ακτίνων X από ποικίλα σημεία του κρυσταλλικού πλέγματος.

Αν και η διαδικασία της μέτρησης καθ' εαυτή είναι ταχεία, ο προσδιορισμός της κρυσταλλικής δομής απαιτεί μετρήσεις σε πολυάριθμες διαφορετικές θέσεις συχνά της τάξης των χιλιάδων για σύνθετα κρυσταλλικά συστήματα. Η επανατοποθέτηση του δείγματος και του εξοπλισμού σε κάθε νέα θέση μέτρησης αποτελεί τον κύριο χρονικό περιορισμό του πειράματος.

Η πολυπλοκότητα αυτής της μηχανικής διαδικασίας επιτείνεται από το γεγονός ότι η μετακίνηση μεταξύ διαφορετικών θέσεων μέτρησης απαιτεί τον συντονισμένο χειρισμό έως και τεσσάρων ξεχωριστών κινητήρων ακριβείας, οι οποίοι ρυθμίζουν τη γωνιακή τοποθέτηση του δείγματος και του ανιχνευτή. Δεδομένου ότι η επιστημονική αξία των μετρήσεων δεν εξαρτάται από τη σειρά με την οποία λαμβάνονται, το πρόβλημα μετατρέπεται στην εύρεση της διαδρομής που ελαχιστοποιεί το συνολικό χρόνο μετακίνησης μεταξύ των θέσεων μέτρησης.

Η μοντελοποίηση αυτής της πειραματικής διαδικασίας ως πρόβλημα TSP επιτρέπει τον προσδιορισμό της βέλτιστης ακολουθίας θέσεων μέτρησης, μειώνοντας σημαντικά τη συνολική διάρκεια του πειράματος και αυξάνοντας κατ' επέκταση την παραγωγικότητα των ερευνητικών εγκαταστάσεων.

1.4.6 Σχεδιασμός παγκόσμιων τοπογραφικών δικτύων δορυφορικών συστημάτων πλοήγησης

Μια καινοτόμος εφαρμογή του TSP εντοπίζεται στον τομέα της γεωδαισίας και της δορυφορικής πλοήγησης, όπως τεκμηριώνεται στην ερευνητική εργασία των Saleh και Chelouah[26]. Η μελέτη τους επικεντρώνεται στη βελτιστοποίηση του σχεδιασμού τοπογραφικών δικτύων που αξιοποιούν τα Παγκόσμια Δορυφορικά Συστήματα Πλοήγησης (Global Navigation Satellite Systems - GNSS).

Τα GNSS αποτελούν εξελιγμένα δορυφορικά συστήματα που παρέχουν πλανητική κάλυψη εντοπισμού θέσης και χρονομέτρησης, καθιστώντας τα θεμελιώδη εργαλεία για ποικίλες εφαρμογές κρίσιμης σημασίας. Αυτές περιλαμβάνουν συστήματα έκτακτης προειδοποίησης και αντιμετώπισης φυσικών καταστροφών, παρακολούθηση περιβαλλοντικών παραμέτρων, εφαρμογές γεωργίας ακριβείας, καθώς και πλήθος άλλων εφαρμογών στις οποίες ο ακριβής γεωχωρικός προσδιορισμός είναι μέγιστης σημασίας.

Στο πλαίσιο της τοπογραφίας, η οποία επιδιώκει τον ακριβή προσδιορισμό των συντεταγμένων άγνωστων σημείων στην επιφάνεια της Γης ή στο εγγύς διαστημικό περιβάλλον, το TSP προσφέρει ένα μαθηματικό μοντέλο για τη βελτιστοποίηση της διαδρομής των γεωδαιτικών μετρήσεων. Η εφαρμογή του TSP σε αυτόν τον τομέα συμβάλλει στην ελαχιστοποίηση του συνολικού χρόνου μέτρησης και στη μεγιστοποίηση της ακρίβειας των τοπογραφικών δικτύων, παράγοντες που είναι καθοριστικοί για την αποτελεσματικότητα των σύγχρονων γεωχωρικών εφαρμογών.

1.5 Σύνδεση με Νευρωνικά Δίκτυα

Από τις πολυάριθμες προσεγγίσεις για την επίλυση του TSP, τα νευρωνικά δίκτυα έχουν αναδειχθεί ως μια ιδιαίτερα ενδιαφέρουσα και πολλά υποσχόμενη κατηγορία. Η σύνδεση μεταξύ του TSP και των νευρωνικών δικτύων δεν είναι τυχαία. Τα νευρωνικά δίκτυα, και ιδιαίτερα τα αναδρομικά και τα δίκτυα τύπου Hopfield, προσφέρουν μια φυσική προσέγγιση για την αντιμετώπιση προβλημάτων βελτιστοποίησης.

Η καινοτομία στη χρήση νευρωνικών δικτύων για το TSP έγκειται στη δυνατότητά τους να μοντελοποιούν σύνθετους περιορισμούς και να βρίσκουν λύσεις μέσω της ελαχιστοποίησης μιας συνάρτησης ενέργειας. Το 1985, οι Hopfield και Tank[27] πρότειναν την πρώτη εφαρμογή νευρωνικών δικτύων για την επίλυση του TSP, δημιουργώντας ένα μοντέλο όπου η κατάσταση ελάχιστης ενέργειας του δικτύου αντιστοιχεί σε μια έγκυρη και (ιδανικά) βέλτιστη διαδρομή.

Έκτοτε, έχουν αναπτυχθεί πολυάριθμες βελτιώσεις και παραλλαγές της αρχικής προσέγγισης Hopfield, οι οποίες στοχεύουν στην αντιμετώπιση των προβλημάτων σύγκλισης και της τάσης να παγιδεύονται σε τοπικά βέλτιστα. Σύγχρονες προσεγγίσεις περιλαμβάνουν υβριδικά μοντέλα που συνδυάζουν νευρωνικά δίκτυα με άλλες τεχνικές βελτιστοποίησης, καθώς και εξειδικευμένες αρχιτεκτονικές που είναι προσαρμοσμένες στη δομή του TSP.

Κεφάλαιο 1ο

Στα επόμενα κεφάλαια, θα εμβαθύνουμε στη λειτουργία των αναδρομικών νευρωνικών δικτύων και των δικτύων Hopfield, και θα εξετάσουμε λεπτομερώς πώς αυτές οι αρχιτεκτονικές μπορούν να εφαρμοστούν για την αποτελεσματική επίλυση του TSP, εστιάζοντας τόσο στις θεωρητικές βάσεις όσο και στις πρακτικές εφαρμογές.

Κεφάλαιο 2: Τα Νευρωνικά Δίκτυα - Μια Εισαγωγή στην Τεχνητή Νοημοσύνη

2.1 Εισαγωγή

Τα νευρωνικά δίκτυα αποτελούν έναν από τους πιο επαναστατικούς τομείς της σύγχρονης επιστήμης της πληροφορικής και της τεχνητής νοημοσύνης. Εμπνευσμένα από τον τρόπο λειτουργίας του ανθρώπινου εγκεφάλου, αυτά τα υπολογιστικά μοντέλα έχουν μεταμορφώσει πολλούς τομείς, από την αναγνώριση εικόνων και την επεξεργασία φυσικής γλώσσας έως την ιατρική διάγνωση και την αυτόνομη οδήγηση.

Η ιστορία των νευρωνικών δικτύων ξεκινά από τη δεκαετία του 1940, όταν οι McCulloch και Pitts (1943) [28] πρότειναν το πρώτο μαθηματικό μοντέλο τεχνητού νευρώνα. Ωστόσο, η πραγματική επανάσταση άρχισε στις αρχές του 21ου αιώνα με την ανάπτυξη των βαθιών νευρωνικών δικτύων (deep neural networks), που οδήγησε στη γέννηση αυτού που σήμερα ονομάζουμε "deep learning".

2.2 Βιολογικό Υπόβαθρο και Έμπνευση

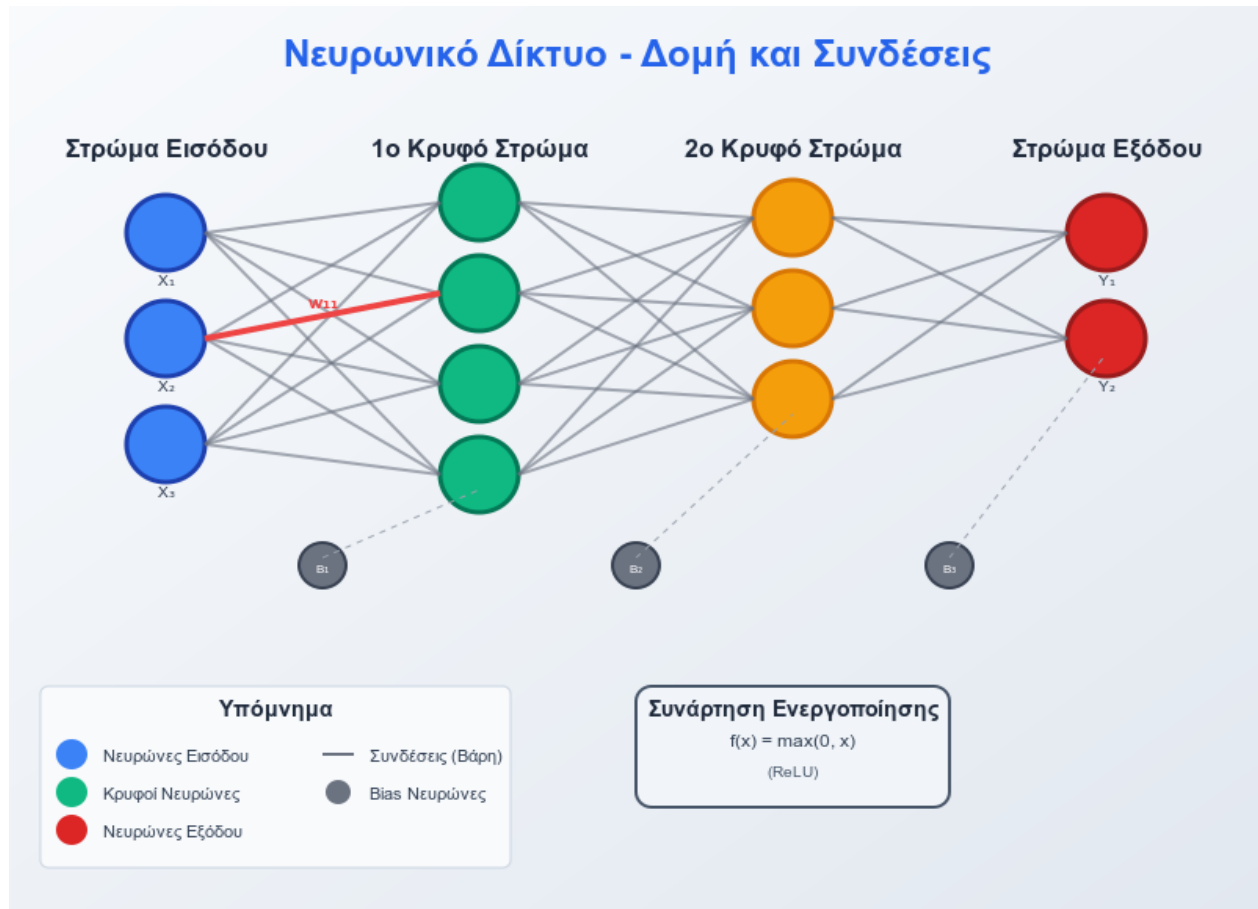
Για να κατανοήσουμε τα τεχνητά νευρωνικά δίκτυα, είναι χρήσιμο να εξετάσουμε πρώτα τη βιολογική τους έμπνευση. Ο ανθρώπινος εγκέφαλος περιέχει περίπου 86 δισεκατομμύρια νευρώνες, που συνδέονται μεταξύ τους μέσω τρισεκατομμυρίων συνάψεων. Κάθε νευρώνας δέχεται σήματα από άλλους νευρώνες μέσω των δενδριτών του, επεξεργάζεται αυτές τις πληροφορίες στο κυτταρικό του σώμα και στη συνέχεια μεταδίδει το αποτέλεσμα μέσω του άξονά του σε άλλους νευρώνες.

Αυτή η απλή αλλά ισχυρή αρχή λειτουργίας ενέπνευσε τους επιστήμονες να δημιουργήσουν μαθηματικά μοντέλα που μιμούνται αυτή τη συμπεριφορά. Ο Rosenblatt (1958) [29] ανέπτυξε το perceptron, το πρώτο αλγοριθμικό μοντέλο που μπορούσε να μάθει από δεδομένα, εμπνευσμένο από τη λειτουργία των πραγματικών νευρώνων.

2.3 Δομή και Λειτουργία των Νευρωνικών Δικτύων

Ένα τεχνητό νευρωνικό δίκτυο αποτελείται από διασυνδεδεμένους κόμβους (νευρώνες) που οργανώνονται σε στρώματα. Το πιο απλό νευρωνικό δίκτυο περιλαμβάνει ένα στρώμα εισόδου (input layer), όπου εισάγονται τα δεδομένα, ένα ή περισσότερα κρυφά στρώματα (hidden layers) που επεξεργάζονται τις πληροφορίες, και ένα στρώμα εξόδου (output layer) που παράγει το τελικό αποτέλεσμα.

Κάθε σύνδεση μεταξύ των νευρώνων έχει ένα βάρος (weight), το οποίο καθορίζει τη δύναμη της σύνδεσης. Κατά τη διαδικασία της εκμάθησης, αυτά τα βάρη προσαρμόζονται ώστε το δίκτυο να μπορεί να κάνει ακριβείς προβλέψεις ή κατηγοριοποιήσεις. Αυτή η διαδικασία γίνεται συνήθως μέσω ενός αλγορίθμου που ονομάζεται backpropagation (Rumelhart et al., 1986)[30], ο οποίος χρησιμοποιεί τη μέθοδο της κλιμακωτής καθόδου για να ελαχιστοποιήσει το σφάλμα μεταξύ των προβλέψεων του δικτύου και των πραγματικών αποτελεσμάτων.



Σχήμα 2.1 : Νευρωνικό δίκτυο-Δομή και συνδέσεις

2.4 Εκμάθηση και Προσαρμογή

Η διαδικασία εκμάθησης σε ένα νευρωνικό δίκτυο είναι μια επαναληπτική διαδικασία που μιμείται τον τρόπο που μαθαίνει ο ανθρώπινος εγκέφαλος μέσω εμπειρίας. Το δίκτυο τροφοδοτείται με πολλά παραδείγματα δεδομένων εκπαίδευσης, και για κάθε παράδειγμα, προσαρμόζει τα βάρη του για να βελτιώσει την απόδοσή του.

Υπάρχουν τρεις κύριοι τύποι εκμάθησης στα νευρωνικά δίκτυα: η επιβλεπόμενη εκμάθηση (supervised learning), όπου το δίκτυο εκπαιδεύεται με παραδείγματα που έχουν τις σωστές απαντήσεις, η μη επιβλεπόμενη εκμάθηση (unsupervised learning), όπου το δίκτυο πρέπει να ανακαλύψει μόνο του μοτίβα στα δεδομένα, και η ενισχυτική εκμάθηση (reinforcement learning), όπου το δίκτυο μαθαίνει μέσω δοκιμής και σφάλματος, λαμβάνοντας επιβραβεύσεις ή ποινές για τις ενέργειές του.

2.5 Βαθιά Νευρωνικά Δίκτυα και Επανάσταση στην Τεχνητή Νοημοσύνη

Η εισαγωγή των βαθιών νευρωνικών δικτύων, τα οποία έχουν πολλά κρυφά στρώματα (συνήθως περισσότερα από τρία), σηματοδότησε μια νέα εποχή στην τεχνητή νοημοσύνη. Αυτά τα δίκτυα μπορούν

να μάθουν σύνθετες αναπαραστάσεις των δεδομένων, επιτρέποντάς τους να αντιμετωπίσουν προβλήματα που παραδοσιακά θεωρούνταν δύσκολα ή αδύνατα για τους υπολογιστές.

Οι Hinton et al. (2006)[31] επανέφεραν το ενδιαφέρον για τα βαθιά νευρωνικά δίκτυα με την εργασία τους στους deep belief networks, ενώ ο AlexNet του Krizhevsky et al. (2012)[32] απέδειξε τη δύναμη των βαθιών συνελκτικών νευρωνικών δικτύων στην αναγνώριση εικόνων, επιτυγχάνοντας σημαντική βελτίωση στη διεθνή ανταγωνιστική αξιολόγηση ImageNet.

2.6 Τύποι Νευρωνικών Δικτύων

Τα νευρωνικά δίκτυα έχουν εξελιχθεί σε πολλές διαφορετικές αρχιτεκτονικές, καθεμία κατάλληλη για συγκεκριμένους τύπους προβλημάτων. Τα συνελκτικά νευρωνικά δίκτυα (Convolutional Neural Networks, CNNs) είναι ιδανικά για την επεξεργασία εικόνων και έχουν επαναστατήσει τομείς όπως η ιατρική απεικόνιση και η αυτόνομη οδήγηση. Τα δίκτυα με επαναληπτικές συνδέσεις (Recurrent Neural Networks, RNNs) και οι μετεξελιξίσεις τους, όπως τα Long Short-Term Memory (LSTM) networks, είναι κατάλληλα για την επεξεργασία ακολουθιακών δεδομένων όπως η φυσική γλώσσα και οι χρονοσειρές.

Μια από τις σημαντικότερες πρόσφατες εξελίξεις είναι ο μηχανισμός προσοχής (attention mechanism) και οι Transformers (Vaswani et al., 2017)[33], οι οποίοι έχουν οδηγήσει στην ανάπτυξη ισχυρών μοντέλων φυσικής γλώσσας όπως το GPT και το BERT.

2.7 Εφαρμογές και Επίδραση στην Κοινωνία

Τα νευρωνικά δίκτυα έχουν βρει εφαρμογή σε πληθώρα τομέων. Στην ιατρική, χρησιμοποιούνται για την ανάλυση ιατρικών εικόνων, την ανακάλυψη φαρμάκων και την πρόβλεψη πορείας ασθενειών. Στον τομέα των μεταφορών, αποτελούν το θεμέλιο των συστημάτων στην τέχνη και τη δημιουργικότητα, έχουν ανοίξει νέους ορίζοντες με τη δημιουργία τέχνης που παράγεται από τεχνητή νοημοσύνη.

Η επιτυχία των νευρωνικών δικτύων έχει επίσης εγείρει σημαντικές ηθικές και κοινωνικές ανησυχίες. Ζητήματα όπως η αμεροληψία των αλγορίθμων, η ιδιωτικότητα των δεδομένων και ο αντίκτυπος στην απασχόληση απαιτούν προσεκτική εξέταση και ρύθμιση. Επιπλέον, η ερμηνευσιμότητα των νευρωνικών δικτύων παραμένει ένα σημαντικό πρόβλημα, καθώς οι αποφάσεις τους συχνά λειτουργούν ως "μαύρο κουτί", καθιστώντας δύσκολη την κατανόηση του τρόπου με τον οποίο φτάνουν σε συγκεκριμένα συμπεράσματα.

2.8 Συμπεράσματα

Τα νευρωνικά δίκτυα έχουν μεταμορφώσει τον τρόπο που βλέπουμε και προσεγγίζουμε την τεχνητή νοημοσύνη. Από τα απλά perceptrons των πρώτων δεκαετιών έως τα σύνθετα deep learning μοντέλα σήμερα, αυτή η τεχνολογία έχει ανοίξει νέους ορίζοντες και έχει αναδείξει δυνατότητες που κάποτε θεωρούνταν επιστημονική φαντασία. Καθώς συνεχίζουμε να εξελίσσουμε και να βελτιώνουμε αυτές τις τεχνολογίες, είναι κρίσιμο να διατηρήσουμε μια ισορροπημένη προσέγγιση που αγκαλιάζει τις δυνατότητες τους ενώ παραμένει ευαίσθητη στις ηθικές και κοινωνικές επιπτώσεις τους. Το μέλλον των νευρωνικών

Κεφάλαιο 2ο

δικτύων υπόσχεται να είναι εξίσου συναρπαστικό με το παρελθόν τους, και η συνεχής έρευνα και ανάπτυξη σε αυτόν τον τομέα θα συνεχίσει να μας εκπλήσσει με νέες δυνατότητες και εφαρμογές.

Κεφάλαιο 3^ο: Recurrent Neural Networks (Αναδρομικά Νευρωνικά Δίκτυα)

3.1 Εισαγωγή

Τα αναδρομικά νευρωνικά δίκτυα είναι αυτά που η δομή τους μοιάζει περισσότερο με τη λειτουργία του ανθρώπινου εγκεφάλου, όπου οι διεργασίες της αντίληψης, της λήψης αποφάσεων και της εκμάθησης είναι αμιγώς μη γραμμικές. Το κύριο χαρακτηριστικό τους που τα διαφοροποιεί από τα feedforward δίκτυα είναι ότι περιέχουν τουλάχιστον μια ανάδραση ανάμεσα στους κόμβους του ίδιου επιπέδου ή κόμβους διαφορετικών επιπέδων, όπως αναφέρθηκε στην προηγούμενη ενότητα. Η ιδιότητά αυτή τους προσδίδει δυο πολύ σημαντικά χαρακτηριστικά έναντι των feedforward δικτύων. Το πρώτο είναι αυτό της αντίληψης του χρόνου, δηλαδή σε εφαρμογές με δυναμικά συστήματα μπορούν να χρησιμοποιηθούν για να προβλέψουν την έξοδο της στιγμής $t + 1$ έχοντας δεδομένα της στιγμής t ή και προηγούμενων. Το επόμενο χαρακτηριστικό είναι η εισαγωγή της μνήμης στα νευρωνικά δίκτυα. Τέτοιου είδους δίκτυα μπορούν να χρησιμοποιηθούν εγγενώς σε προβλήματα στα οποία οι είσοδοι δεν είναι ανεξάρτητες μεταξύ τους, αλλά υπάρχει κάποια συσχέτιση με προηγούμενα δείγματα.

Τα αναδρομικά νευρωνικά δίκτυα είναι δυναμικά συστήματα που μπορούν να μοντελοποιήσουν και να αναπαραστήσουν χρονικά μεταβαλλόμενα μοντέλα. Το γεγονός ότι χρησιμοποιούνται για να προσομοιώσουν συστήματα με τη χρήση κοινών διαφορικών εξισώσεων, τα κάνει κατάλληλα για ψηφιακή υλοποίηση και έχουν επιπλέον ισχυρές υπολογιστικές δυνατότητες όταν χρησιμοποιούνται σε αυτά τα μοντέλα. Τέτοιου είδους μοντέλα είναι και όλα τα βιολογικά νευρωνικά δίκτυα. Μπορούν, επίσης, να χρησιμοποιηθούν αποδοτικά σε θέματα που περιλαμβάνουν την αναγνώριση και αντίστροφη αναγνώριση συστημάτων, το φιλτράρισμα και την πρόβλεψη πληροφοριών, την κατάταξη δεδομένων σε κλάσεις, την μοντελοποίηση στοχαστικών ακολουθιών, την συμπίεση δεδομένων κ.α.. Κάποιοι από τους τομείς στους οποίους χρησιμοποιούνται αποδοτικά ή έχουν καταδείξει πολλά υποσχόμενα αποτελέσματα είναι οι τηλεπικοινωνίες, ο έλεγχος χημικών εγκαταστάσεων, ο έλεγχος μηχανών και γεννητριών, η ρομποτική, η αναγνώριση ομιλίας, η παραγωγή meta πληροφοριών από εικόνες, η ανάλυση δεδομένων σε βίντεο, η μετάφραση και η αυτόματη συγγραφή κειμένου ή κώδικα.

Τα αναδρομικά νευρωνικά δίκτυα (Recurrent Neural Networks - RNNs) αποτελούν μια κατηγορία τεχνητών νευρωνικών δικτύων που έχουν σχεδιαστεί ειδικά για την επεξεργασία ακολουθιακών δεδομένων και χρονοσειρών. Η αρχιτεκτονική τους μιμείται περισσότερο τη λειτουργία του ανθρώπινου εγκεφάλου, όπου οι διεργασίες της αντίληψης, της λήψης αποφάσεων και της εκμάθησης είναι αμιγώς μη γραμμικές και διαθέτουν την ιδιότητα της μνήμης.

Το κύριο χαρακτηριστικό που διαφοροποιεί τα RNNs από τα συμβατικά feed-forward δίκτυα είναι η παρουσία τουλάχιστον μίας ανάδρασης (feedback) μεταξύ των νευρώνων, είτε στο ίδιο επίπεδο είτε μεταξύ διαφορετικών επιπέδων. Αυτή η ιδιότητα προσδίδει στα RNNs κάποια σημαντικά πλεονεκτήματα, με τα κυριότερα να είναι η αντίληψη του χρόνου και η ενσωματωμένη μνήμη. Στην πρώτη περίπτωση, τα RNNs μπορούν να επεξεργαστούν δεδομένα με χρονική διάσταση, επιτρέποντάς τους να προβλέψουν μελλοντικές καταστάσεις με βάση προηγούμενες παρατηρήσεις ενώ στη δεύτερη περίπτωση μέσω των αναδρομικών

συνδέσεων, τα RNNs διατηρούν πληροφορίες από προηγούμενα βήματα, λειτουργώντας ως ένα είδος "εσωτερικής μνήμης" του δικτύου.

Λόγω αυτών των χαρακτηριστικών, τα RNNs έχουν καταστεί ιδιαίτερα αποτελεσματικά σε προβλήματα όπου τα δεδομένα εισόδου δεν είναι ανεξάρτητα μεταξύ τους, αλλά υπάρχει χρονική ή ακολουθιακή συσχέτιση, όπως στην αναγνώριση ομιλίας, τη μετάφραση κειμένου, την πρόβλεψη χρονοσειρών και τη γενικότερη μοντελοποίηση ακολουθιακών δεδομένων.

3.2 Ιστορική Αναδρομή και Εξέλιξη των RNNs

Η ιδέα των αναδρομικών συνδέσεων σε νευρωνικά δίκτυα πρωτοεμφανίστηκε τη δεκαετία του 1980, όταν οι ερευνητές άρχισαν να διερευνούν μοντέλα που θα μπορούσαν να διαχειριστούν χρονικά μεταβαλλόμενα δεδομένα. Αναλύοντας την εξέλιξη των RNNs προκύπτουν κάποιες χρονικές στιγμές ορόσημα, οι οποίες θα αναφερθούν αναλυτικότερα παρακάτω.

Αρχικά, το 1982 προκύπτει η εισαγωγή του δικτύου Hopfield από τον John Hopfield[27], το οποίο είναι και από τα πρώτα RNNs που χρησιμοποιήθηκε για προβλήματα βελτιστοποίησης αλλά και συνειρμικής μνήμης. Έπειτα, κατά τη διάρκεια του 1986 έχουμε τη παρουσίαση του αλγορίθμου Backpropagation Through Time (BPTT) από τους David Rumelhart, Geoffrey Hinton και Ronald Williams[30], οι οποίοι επέκτειναν τον κλασικό αλγόριθμο backpropagation για τη χρήση του σε αναδρομικά δίκτυα.

Κατά τη δεκαετία του 1990 ξεκίνησαν να κάνουν την εμφάνιση τους προβλήματα "εκπαίδευσης" των RNNs, όπως για παράδειγμα το "vanishing gradient problem" ή αλλιώς πρόβλημα εξαφανιζόμενης κλίσης, όπου η κλίσεις κατά το backpropagation τείνουν να μειώνονται εκθετικά με το βάθος του δικτύου. Η λύση θα προταθεί το 1997 από τους Hochreiter και Schmidhuber[34], μια ειδική αρχιτεκτονική RNN σχεδιασμένη για την αντιμετώπιση του προβλήματος εξαφανιζόμενης κλίσης με την ονομασία Long Short-Term Memory (LSTM) networks. Η συγκριμένη λύση LSTM πρόκειται να απλοποιηθεί με λιγότερες παραμέτρους το 2014 από τον Cho[35] και τους συνεργάτες του, οι οποίοι εισήγαγαν τα Gated Recurrent Units (GRUs).

Σήμερα, τα RNNs και οι παραλλαγές τους αποτελούν θεμελιώδη συστατικά πολλών συστημάτων τεχνητής νοημοσύνης, ιδιαίτερα στους τομείς της επεξεργασίας φυσικής γλώσσας και της ανάλυσης χρονοσειρών.

3.3 Τύποι Αναδρομικών Νευρωνικών Δικτύων

Τα αναδρομικά νευρωνικά δίκτυα μπορούν να κατηγοριοποιηθούν με βάση διάφορα κριτήρια, όπως η δομή των συνδέσεων και ο τρόπος λειτουργίας τους.

Ο διαχωρισμός τους με βάση τη δομή συνδέσεων είναι σε Πλήρως Αναδρομικά Δίκτυα (Fully Recurrent Networks) και σε Τοπικά Αναδρομικά Δίκτυα (Locally Recurrent Networks).

Στη πρώτη κατηγορία και στα Πλήρως Αναδρομικά Δίκτυα, κάθε νευρώνας μπορεί να συνδεθεί με οποιονδήποτε άλλο νευρώνα, συμπεριλαμβανομένου και του εαυτού του. Αυτή η γενική αρχιτεκτονική προσφέρει μεγάλη ευελιξία, αλλά μπορεί να παρουσιάσει προβλήματα ευστάθειας κατά την εκπαίδευση και να απαιτεί περίπλοκους αλγορίθμους μάθησης.

Στη δεύτερη κατηγορία, τα Τοπικά αναδρομικά Δίκτυα περιλαμβάνουν αναδρομικές συνδέσεις μόνο μεταξύ νευρώνων που βρίσκονται στο ίδιο επίπεδο. Η διαδικασία διάδοσης προς τα εμπρός μοιάζει περισσότερο με αυτήν των feed-forward δικτύων, αλλά με το πρόσθετο στοιχείο της μνήμης εντός κάθε επιπέδου.

Στη συνέχεια, ακολουθεί ο διαχωρισμός με βάση τον τρόπο λειτουργίας τους, και έχουμε τα Αναδρομικά Δίκτυα με Χρονοκαθυστέρηση (Time-delay Neural Networks) αλλά και τα Σύγχρονα Αναδρομικά Δίκτυα (Synchronous Recurrent Networks).

Στην περίπτωση των Αναδρομικών Δικτύων με Χρονοκαθυστέρηση, τα δίκτυα εκπαιδεύονται με στόχο τη μείωση του σφάλματος πρόβλεψης, χρησιμοποιώντας καθυστερημένες εισόδους προκειμένου να αποκτήσουν χρονικό πλαίσιο αναφοράς.

Στη δεύτερη περίπτωση των Σύγχρονων Αναδρομικών Δικτύων, δεν επικεντρώνονται στη μνήμη ή την πρόβλεψη, αλλά αξιοποιούν τις αναδράσεις για να βελτιώσουν την ικανότητα προσέγγισης συναρτήσεων, σύμφωνα με τις αρχές της θεωρίας του Turing και της υπολογιστικής πολυπλοκότητας.

3.3.1 Ειδικές Αρχιτεκτονικές RNN

Αρχικά, είναι τα δίκτυα LSTM (Long Short-Term Networks), τα οποία είναι μια ειδική μορφή RNN σχεδιασμένη για να αντιμετωπίσει το πρόβλημα της εξαφανιζόμενης κλίσης. Διαθέτουν μια πιο σύνθετη δομή νευρώνων με "πύλες" (gates) που ελέγχουν τη ροή της πληροφορίας, επιτρέποντας στο δίκτυο να διατηρεί πληροφορίες για μεγάλες χρονικές περιόδους.

Στη συνέχεια, ακολουθούν τα αμφίδρομα RNNs (Bidirectional RNNs), τα οποία επεξεργάζονται τα δεδομένα εισόδου και προς τις δύο κατευθύνσεις (από αριστερά προς τα δεξιά και αντίστροφα), επιτρέποντας στο δίκτυο να λαμβάνει υπόψη τόσο προηγούμενες όσο και μελλοντικές πληροφορίες για κάθε χρονικό σημείο.

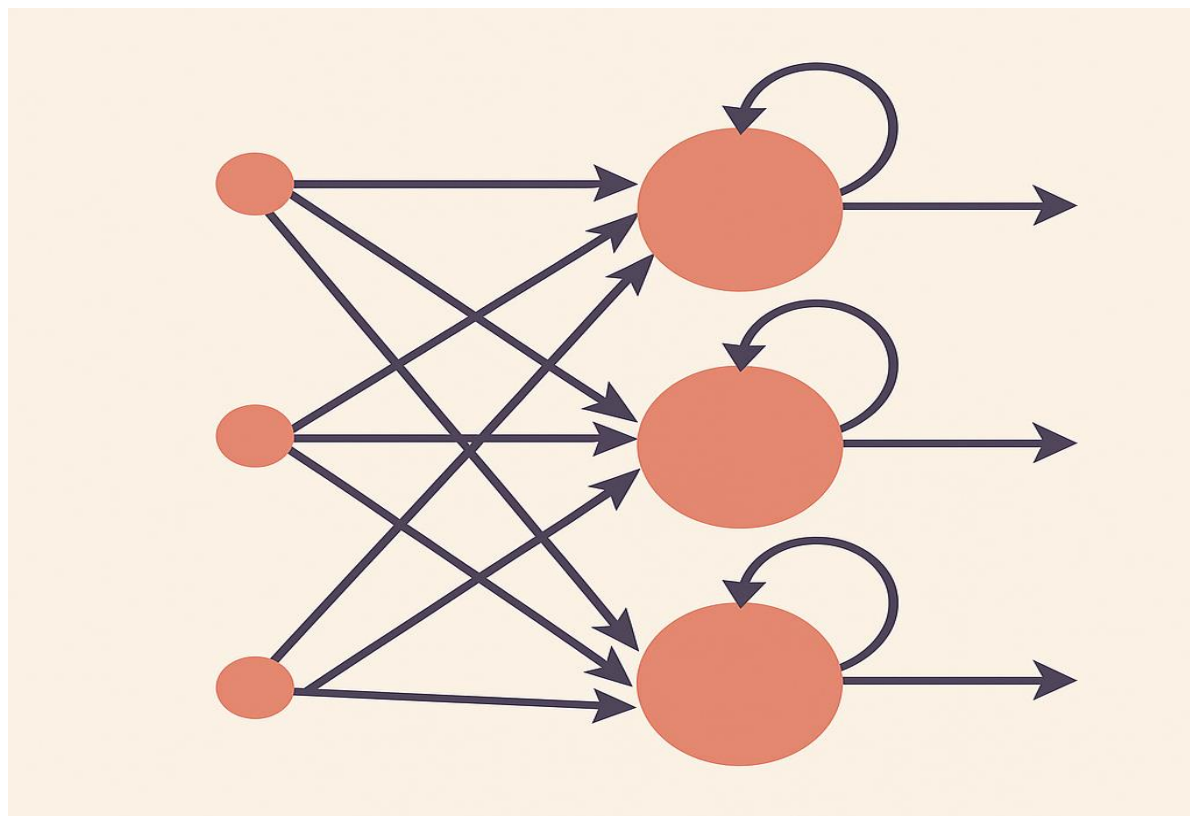
Τελικά, είναι τα ESNs (Echo State Networks) τα οποία υπόκεινται στην κατηγορία των "reservoir computing" και χαρακτηρίζονται από μια μεγάλη, τυχαία συνδεδεμένη, αραιή δεξαμενή νευρώνων. Μόνο τα βάρη των συνδέσεων εξόδου εκπαιδεύονται, ενώ τα υπόλοιπα παραμένουν σταθερά.

3.4 Περιγραφή της Αρχιτεκτονικής των RNN

Το βασικό δομικό στοιχείο ενός αναδρομικού νευρωνικού δικτύου είναι οι νευρώνες (μονάδες ή κόμβοι του συστήματος). Οι νευρώνες συνδέονται μεταξύ τους με συναπτικούς δεσμούς οι οποίοι περιγράφονται από το βάρος τους. Διαχωρίζονται συνήθως σε κόμβους εισόδου, εσωτερικούς κόμβους και εξόδου. Κάθε χρονική στιγμή οι κόμβοι έχουν μια τιμή ενεργοποίησης, η οποία αναπαρίσταται με $u(n)$ για τους κόμβους εισόδου, $x(n)$ για τους εσωτερικούς και $y(n)$ για τους κόμβους εξόδου. Πολλές φορές αναφερόμαστε σε όλους με την ονομασία $x(n)$.

Τα αναδρομικά νευρωνικά δίκτυα κατηγοριοποιούνται σε πλήρως αναδρομικά δίκτυα, στα οποία επιτρέπεται η σύνδεση όλων των νευρώνων μεταξύ τους και στα τοπικά αναδρομικά δίκτυα. Στα πλήρως αναδρομικά δίκτυα δεν υπάρχει διάκριση στους κόμβους εισόδου και κάθε κόμβος μπορεί να αποτελέσει είσοδο για οποιονδήποτε άλλο κόμβο, συμπεριλαμβανομένου και του εαυτού του. Τα δεύτερα περιέχουν, συνήθως, μόνο αναδρομικές συνδέσεις μεταξύ των νευρώνων που βρίσκονται ίδιο επίπεδο και η διαδικασία της διάδοσης προς τα εμπρός κατά την εκμάθηση μοιάζει πολύ με αυτή των feedforward δικτύων. Συγκριτικά με τα τοπικά αναδρομικά δίκτυα, τα πλήρως αναδρομικά πάσχουν από θέματα ευστάθειας κατά το training και απαιτούν τη χρήση περίπλοκων και χρονοβόρων αλγορίθμων εκμάθησης. Τα τοπικά αναδρομικά δίκτυα έχουν, αντιθέτως, πιο απλή δομή που τα κάνει πιο αποδοτικά κατά τη διαδικασία εκμάθησης και παρέχουν τη δυνατότητα για έλεγχο της ευστάθειας στους εσωτερικούς τους κόμβους.

Μια ακόμα διάκριση των RNN που χρησιμοποιούνται σε εφαρμογές διακριτού χρόνου είναι σε αναδρομικά δίκτυα με χρονοκαθυστέρηση και σύγχρονα αναδρομικά δίκτυα. Τα δίκτυα που λειτουργούν με χρονοκαθυστέρηση εκπαιδεύονται με στόχο τη μείωση του σφάλματος πρόβλεψης, ενώ τα σύγχρονα δίκτυα δε στοχεύουν στην ιδιότητα του να έχουν μνήμη ή καλύτερη πρόβλεψη όσο εκπαιδεύεται το δίκτυο, αλλά κάνουν χρήση των αναδράσεων με στόχο να προσφέρουν καλύτερη δυνατότητα προσέγγισης συναρτήσεων σύμφωνα με τις αρχές της θεωρίας του Turing και της περιπλοκότητας. Σε αυτό τον τομέα έχει αποδειχθεί ότι έχουν πολύ ισχυρές ικανότητες και έχει φανεί πειραματικά ότι μπορούν να “μάθουν” οποιαδήποτε συνάρτηση πηγάει από κάποιο MLP (Multi Layer Perceptron), χωρίς όμως να ισχύει το αντίστροφο.



Σχήμα 3.1 Δομή Νευρωνικού δικτύου

Παρακάτω ακολουθεί η περιγραφή και ο ορισμός των παραμέτρων ενός απλού αναδρομικού δικτύου. Το συγκεκριμένο μοντέλο αποτελείται από K μονάδες εισόδου που περιγράφονται από το διάνυσμα ενεργοποίησης εισόδου [3]:

$$u(n) = (u_1(n), \dots, u_k(n))^t, \quad (3.1)$$

υπάρχουν N εσωτερικοί κόμβοι με διάνυσμα ενεργοποίησης:

$$x(n) = (x_1(n), \dots, x_k(n))^t, \quad (3.2)$$

και L κόμβοι εξόδου με διάνυσμα ενεργοποίησης:

$$y(n) = (y_1(n), \dots, y_k(n))^t, \quad (3.3)$$

όπου με t ορίζεται ο ανάστροφος πίνακας.

Κεφάλαιο 3ο

Τα βάρη των συνδέσεων των κόμβων εισόδου, εσωτερικών και εξόδου συλλέγονται σε τρεις διαφορετικούς πίνακες βαρών, διαστάσεων $N_x K$, $N_x N$ και $L_x(K + N)$ αντίστοιχα. Οι πίνακες αυτοί ορίζονται ως:

$$W^{in} = [W^{in}(i, j)], W = [w(i, j)], W^{out} = [W^{out}(i, j)] \quad (3.4)$$

Οι κόμβοι εξόδου μπορεί να συνδέονται ξανά με τους εσωτερικούς κόμβους με βάρη που αποθηκεύονται σε έναν ακόμη πίνακα

$$W^{back} = [W^{back}(i, j)] \quad (3.5)$$

Μια αναπαράσταση ενός τέτοιου αναδρομικού δικτύου φαίνεται στο Σχήμα 3.1.

Οι συνδέσεις που έχουν βάρος 0 θεωρούνται ως μη υπαρκτές και δεν επηρεάζουν το σύστημα, ενώ μπορούν να υπάρχουν συνδέσεις με τους κόμβους εξόδου όχι μόνο από τους εσωτερικούς κόμβους αλλά και κατευθείαν από τους κόμβους εισόδου ή αναδρομικές από αυτές που βρίσκονται στο ίδιο το επίπεδο της εξόδου.

Η ενεργοποίηση των εσωτερικών κόμβων γίνεται σύμφωνα με τον τύπο:

$$x(n + 1) = f(W^{in}u(n + 1) + Wx(n) + W^{back}y(n)) \quad (3.6)$$

όπου $u(n+1)$ είναι η είσοδος του παρόντος δείγματος και f είναι η συνάρτηση μεταφοράς του συγκεκριμένου κόμβου ή αλλιώς συνάρτηση ενεργοποίησης. Τις περισσότερες φορές αυτή η συνάρτηση έχει τη μορφή κάποιας σιγμοειδούς συναρτήσεως, όπως αυτή της υπερβολικής εφαπτομένης ($f = \tanh$). Μπορεί όμως να είναι μια οποιαδήποτε συνάρτηση.

Η έξοδος του δικτύου υπολογίζεται από τον τύπο:

$$y(n + 1) = f^{out}(W^{out}(u(n + 1) + x(n + 1) + y(n))) \quad (3.7)$$

Όπου:

$$(u(n + 1), x(n + 1), y(n)) \quad (3.8)$$

είναι το συγχωνευμένο διάνυσμα των διανυσμάτων εισόδου, εσωτερικών και εξόδου και η συνάρτηση f^{out} είναι συνήθως η ίδια συνάρτηση με την f , αλλά δεν απαγορεύεται και να διαφέρουν μεταξύ τους.

3.5 Εκπαίδευση και Βελτιστοποίηση των RNNs

Η εκπαίδευση των αναδρομικών νευρωνικών δικτύων παρουσιάζει σημαντικές προκλήσεις που οφείλονται στην ιδιαίτερη αρχιτεκτονική και τη χρονική διάσταση των δεδομένων που επεξεργάζονται. Οι συμβατικοί αλγόριθμοι εκπαίδευσης που χρησιμοποιούνται στα feed-forward νευρωνικά δίκτυα πρέπει να τροποποιηθούν κατάλληλα ώστε να λαμβάνουν υπόψη τις αναδρομικές συνδέσεις και τις χρονικές εξαρτήσεις που χαρακτηρίζουν τα RNNs.

Η κύρια μέθοδος εκπαίδευσης των RNNs είναι η τεχνική Backpropagation Through Time (BPTT), η οποία αποτελεί επέκταση του συμβατικού αλγορίθμου οπισθοδιάδοσης. Η βασική ιδέα της οπισθοδιάδοσης είναι η μετατροπή του αναδρομικού δικτύου σε ένα ισοδύναμο feed-forward δίκτυο μέσω του "ξεδιπλώματος" (unfolding) της χρονικής διάστασης. Συγκεκριμένα, το αναδρομικό δίκτυο αναπαρίσταται ως ένα βαθύ feed-forward δίκτυο, όπου κάθε επίπεδο αντιστοιχεί σε ένα χρονικό βήμα και οι συνάψεις μεταξύ των επιπέδων αντιστοιχούν στις αναδρομικές συνδέσεις του αρχικού δικτύου.

Ωστόσο, η BPTT παρουσιάζει δύο σημαντικά προβλήματα που περιορίζουν την αποτελεσματικότητά της στην εκπαίδευση των RNNs: το πρόβλημα της εξαφανιζόμενης κλίσης (vanishing gradient problem) και το πρόβλημα της εκρηκτικής κλίσης (exploding gradient problem).

Το πρόβλημα της εξαφανιζόμενης κλίσης εμφανίζεται όταν οι κλίσεις του σφάλματος ως προς τα βάρη του δικτύου τείνουν να μειώνονται εκθετικά καθώς διαδίδονται προς τα πίσω στο χρόνο. Αυτό συμβαίνει επειδή κατά την οπισθοδιάδοση, οι κλίσεις πολλαπλασιάζονται με τα βάρη του δικτύου σε κάθε χρονικό βήμα. Εάν τα βάρη έχουν τιμές μικρότερες της μονάδας, το γινόμενο τους για πολλά χρονικά βήματα μπορεί να γίνει πολύ μικρό, οδηγώντας σε μηδενικές κλίσεις. Σε αυτή την περίπτωση, το δίκτυο αδυνατεί να μάθει μακροπρόθεσμες εξαρτήσεις, καθώς οι αλλαγές στα βάρη που αντιστοιχούν σε προηγούμενα χρονικά βήματα είναι αμελητέες.

Αντίθετα, το πρόβλημα της εκρηκτικής κλίσης εμφανίζεται όταν οι κλίσεις αυξάνονται εκθετικά καθώς διαδίδονται προς τα πίσω στο χρόνο. Αυτό συμβαίνει όταν τα βάρη του δικτύου έχουν τιμές μεγαλύτερες της μονάδας, οδηγώντας σε πολύ μεγάλες αλλαγές στα βάρη κατά την εκπαίδευση. Αυτό μπορεί να προκαλέσει αστάθεια στην εκπαίδευση του δικτύου και να οδηγήσει σε αποτυχία σύγκλισης ή σε υπερχειλίση αριθμητικών τιμών.

Για την αντιμετώπιση αυτών των προβλημάτων, έχουν αναπτυχθεί διάφορες τεχνικές. Μία από τις πιο απλές αλλά αποτελεσματικές προσεγγίσεις για το πρόβλημα της εκρηκτικής κλίσης είναι η τεχνική "gradient clipping", όπου οι κλίσεις περιορίζονται σε ένα προκαθορισμένο εύρος τιμών όταν υπερβαίνουν ένα συγκεκριμένο κατώφλι. Αυτό αποτρέπει τις μεγάλες αλλαγές στα βάρη του δικτύου και προάγει τη σταθερότητα κατά την εκπαίδευση.

Επιπλέον, έχουν αναπτυχθεί ειδικές αρχιτεκτονικές RNNs που σχεδιάστηκαν ειδικά για να αντιμετωπίσουν το πρόβλημα της εξαφανιζόμενης κλίσης. Οι πιο γνωστές από αυτές είναι τα Long Short-Term Memory (LSTM) δίκτυα και τα Gated Recurrent Units (GRU). Και οι δύο αυτές αρχιτεκτονικές χρησιμοποιούν

μηχανισμούς πυλών (gates) που ελέγχουν τη ροή της πληροφορίας μέσα στο δίκτυο, επιτρέποντας την αποτελεσματικότερη μάθηση μακροπρόθεσμων εξαρτήσεων.

Μια άλλη προσέγγιση για τη βελτίωση της αποδοτικότητας της εκπαίδευσης των RNNs είναι η τεχνική "Truncated BPTT". Σε αυτή την παραλλαγή της BPTT, η οπισθοδιάδοση των κλίσεων σταματά μετά από έναν συγκεκριμένο αριθμό χρονικών βημάτων, αντί να συνεχίζεται μέχρι την αρχή της ακολουθίας. Αυτό μειώνει την υπολογιστική πολυπλοκότητα της εκπαίδευσης αλλά μπορεί να περιορίσει την ικανότητα του δικτύου να μάθει πολύ μακροπρόθεσμες εξαρτήσεις.

Επιπροσθέτως, σύγχρονες προσεγγίσεις στην εκπαίδευση των RNNs περιλαμβάνουν τη χρήση προσαρμοστικών αλγορίθμων βελτιστοποίησης όπως ο Adam, ο RMSprop και ο AdaGrad, οι οποίοι προσαρμόζουν τον ρυθμό μάθησης ξεχωριστά για κάθε παράμετρο του δικτύου, βελτιώνοντας έτσι τη σύγκλιση και την ευστάθεια της εκπαίδευσης.

Τέλος, η χρήση τεχνικών κανονικοποίησης όπως για παράδειγμα η κανονικοποίηση παρτίδας (batch normalization) και η κανονικοποίηση στρώματος (layer normalization) έχει αποδειχθεί αποτελεσματική στη βελτίωση της εκπαίδευσης των RNNs, καθώς συμβάλλουν στην σταθεροποίηση της κατανομής των ενεργοποιήσεων των νευρώνων και μειώνουν την ευαισθησία του δικτύου στην αρχικοποίηση των βαρών.

3.6 Εφαρμογές των Αναδρομικών Νευρωνικών Δικτύων

Τα αναδρομικά νευρωνικά δίκτυα, χάρη στις μοναδικές τους ιδιότητες και την ικανότητά τους να μοντελοποιούν χρονικές εξαρτήσεις, έχουν βρει εφαρμογή σε πληθώρα επιστημονικών πεδίων και τεχνολογικών εφαρμογών. Η ευρεία χρήση τους σε διαφορετικούς τομείς αποδεικνύει την ευελιξία και την αποτελεσματικότητά τους στην αντιμετώπιση πολύπλοκων προβλημάτων που εμπεριέχουν ακολουθιακά δεδομένα.

Στον τομέα της επεξεργασίας φυσικής γλώσσας (Natural Language Processing - NLP), τα RNNs αποτελούν θεμελιώδη συστατικά πολλών συστημάτων. Χρησιμοποιούνται εκτενώς για την ανάλυση συναισθήματος σε κείμενα, όπου το δίκτυο μπορεί να αξιολογήσει τη συναισθηματική χροιά ενός κειμένου λαμβάνοντας υπόψη όχι μόνο τις λέξεις που περιέχει αλλά και τη σειρά με την οποία εμφανίζονται. Επιπλέον, τα RNNs χρησιμοποιούνται ευρέως σε συστήματα μηχανικής μετάφρασης, όπως το Google Translate, όπου το δίκτυο πρέπει να κατανοήσει τη σημασιολογική δομή μιας πρότασης στη γλώσσα πηγή και να παράγει την αντίστοιχη δομή στη γλώσσα στόχο. Ακόμη, τα RNNs έχουν χρησιμοποιηθεί με επιτυχία στη δημιουργία κειμένου, όπου το δίκτυο εκπαιδεύεται να προβλέπει την επόμενη λέξη σε μια ακολουθία, οδηγώντας σε μοντέλα που μπορούν να παράγουν συνεκτικό κείμενο σε διάφορα στυλ και θέματα. Τέλος, τα συνομιλιακά συστήματα και οι εικονικοί βοηθοί, όπως η Siri και η Alexa, αξιοποιούν RNNs για την κατανόηση και την παραγωγή φυσικής γλώσσας σε πραγματικό χρόνο.

Στον τομέα της αναγνώρισης ομιλίας, τα RNNs έχουν συμβάλει σημαντικά στην ανάπτυξη συστημάτων μετατροπής ομιλίας σε κείμενο (speech-to-text). Αυτά τα συστήματα χρησιμοποιούν RNNs για να μοντελοποιήσουν τις χρονικές εξαρτήσεις στα ηχητικά σήματα και να αντιστοιχίσουν ακολουθίες ηχητικών χαρακτηριστικών σε ακολουθίες λέξεων. Επιπλέον, τα RNNs χρησιμοποιούνται στην αναγνώριση ομιλητή,

όπου το δίκτυο εκπαιδεύεται να αναγνωρίζει τα μοναδικά χαρακτηριστικά της φωνής κάθε ατόμου, επιτρέποντας την ταυτοποίηση του ομιλητή με βάση τη φωνή του.

Στην ανάλυση χρονοσειρών, τα RNNs έχουν αποδειχθεί εξαιρετικά αποτελεσματικά στην πρόβλεψη μελλοντικών τιμών με βάση προηγούμενες παρατηρήσεις. Στον τομέα των χρηματοοικονομικών, χρησιμοποιούνται για την πρόβλεψη τιμών μετοχών, συναλλαγματικών ισοτιμιών και άλλων οικονομικών δεικτών, λαμβάνοντας υπόψη τόσο τις πρόσφατες τάσεις όσο και μακροπρόθεσμα πρότυπα. Στη μετεωρολογία, τα RNNs χρησιμοποιούνται για την πρόγνωση καιρικών συνθηκών, αναλύοντας τις χρονικές εξαρτήσεις σε δεδομένα θερμοκρασίας, υγρασίας, ατμοσφαιρικής πίεσης και άλλων μετεωρολογικών παραμέτρων. Επιπλέον, στον τομέα της ενέργειας, τα RNNs χρησιμοποιούνται για την πρόβλεψη της κατανάλωσης ενέργειας, επιτρέποντας την αποτελεσματικότερη διαχείριση των ενεργειακών πόρων και τη βελτιστοποίηση της παραγωγής ενέργειας.

Στον τομέα της βιοπληροφορικής, τα RNNs έχουν εφαρμοστεί στην ανάλυση αλληλουχιών DNA και πρωτεϊνών. Χρησιμοποιούνται για την αναγνώριση προτύπων σε γονιδιακές αλληλουχίες, την πρόβλεψη λειτουργικών περιοχών σε πρωτεΐνες και την ταξινόμηση γονιδίων με βάση τις λειτουργικές τους ιδιότητες. Επιπλέον, τα RNNs έχουν χρησιμοποιηθεί στην πρόβλεψη της τρισδιάστατης δομής πρωτεϊνών από την αλληλουχία αμινοξέων, ένα από τα πιο δύσκολα προβλήματα στη βιολογία.

Στον τομέα της ρομποτικής και του ελέγχου, τα RNNs χρησιμοποιούνται για τον έλεγχο ρομποτικών συστημάτων σε πραγματικό χρόνο. Μπορούν να μοντελοποιήσουν τη δυναμική συμπεριφορά του ρομπότ και του περιβάλλοντός του, επιτρέποντας την προσαρμοστική λήψη αποφάσεων σε μεταβαλλόμενες συνθήκες. Επιπλέον, στα αυτόνομα οχήματα, τα RNNs χρησιμοποιούνται για την επεξεργασία δεδομένων από αισθητήρες και την πρόβλεψη της κίνησης άλλων οχημάτων και πεζών, συμβάλλοντας στην ασφαλή πλοήγηση.

Αξίζει να σημειωθεί ότι οι εφαρμογές των RNNs συνεχώς διευρύνονται, καθώς οι ερευνητές ανακαλύπτουν νέους τρόπους αξιοποίησης των μοναδικών χαρακτηριστικών τους σε διάφορους τομείς. Η ικανότητά τους να μοντελοποιούν χρονικές εξαρτήσεις και να διατηρούν πληροφορίες από προηγούμενα βήματα τα καθιστά ιδανικά για την αντιμετώπιση πολύπλοκων προβλημάτων που εμπεριέχουν ακολουθιακά δεδομένα.

3.7 Περιορισμοί και Μελλοντικές Κατευθύνσεις

Παρά την αξιοσημείωτη επιτυχία των αναδρομικών νευρωνικών δικτύων σε πολλές εφαρμογές, η αρχιτεκτονική τους και ο τρόπος λειτουργίας τους εμφανίζουν ορισμένους εγγενείς περιορισμούς που επηρεάζουν την απόδοση και την εφαρμοσιμότητά τους σε συγκεκριμένα προβλήματα. Η κατανόηση αυτών των περιορισμών είναι κρίσιμη για την ανάπτυξη βελτιωμένων αρχιτεκτονικών και τεχνικών εκπαίδευσης στο μέλλον.

Ένας από τους κύριους περιορισμούς των RNNs είναι η υπολογιστική τους πολυπλοκότητα, ιδιαίτερα κατά την εκπαίδευση με τον αλγόριθμο Backpropagation Through Time (BPTT). Η πολυπλοκότητα αυτή

αυξάνεται γραμμικά με το μήκος της ακολουθίας εισόδου, καθιστώντας την εκπαίδευση των RNNs σε μεγάλες ακολουθίες υπολογιστικά απαιτητική και χρονοβόρα. Επιπλέον, ο σειριακός χαρακτήρας της επεξεργασίας στα RNNs περιορίζει τις δυνατότητες παραλληλισμού κατά την εκπαίδευση, σε αντίθεση με άλλες αρχιτεκτονικές όπως τα Συνελκτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks - CNNs) που επιτρέπουν πιο αποτελεσματική παράλληλη επεξεργασία.

Ένας άλλος σημαντικός περιορισμός των RNNs είναι η δυσκολία τους στην εκμάθηση μακροπρόθεσμων εξαρτήσεων, παρά τις βελτιώσεις που έχουν επιτευχθεί με αρχιτεκτονικές όπως τα LSTM και τα GRU. Αν και αυτές οι αρχιτεκτονικές έχουν σχεδιαστεί ειδικά για να αντιμετωπίσουν το πρόβλημα της εξαφανιζόμενης κλίσης, εξακολουθούν να αντιμετωπίζουν δυσκολίες στην αποτελεσματική διατήρηση πληροφοριών για πολύ μεγάλα χρονικά διαστήματα. Αυτό μπορεί να είναι ιδιαίτερα προβληματικό σε εφαρμογές όπως η επεξεργασία μεγάλων κειμένων ή η ανάλυση χρονοσειρών με μακροπρόθεσμες τάσεις.

Επιπρόσθετα, τα RNNs είναι επιρρεπή σε υπερπροσαρμογή (overfitting), ιδιαίτερα όταν τα διαθέσιμα δεδομένα εκπαίδευσης είναι περιορισμένα. Αυτό οφείλεται στον μεγάλο αριθμό παραμέτρων που πρέπει να εκτιμηθούν, ιδιαίτερα σε πιο σύνθετες αρχιτεκτονικές όπως τα LSTM. Τεχνικές κανονικοποίησης όπως το dropout και η κανονικοποίηση L2 μπορούν να βοηθήσουν στην αντιμετώπιση αυτού του προβλήματος, αλλά συχνά απαιτείται προσεκτική ρύθμιση των υπερπαραμέτρων.

Τέλος, τα RNNs αντιμετωπίζουν δυσκολίες στην επεξεργασία παράλληλων ακολουθιών ή στην ενσωμάτωση πολλαπλών πηγών πληροφοριών. Για παράδειγμα, σε εφαρμογές όπως η πολυτροπική μάθηση (multimodal learning), όπου απαιτείται η συγχώνευση πληροφοριών από διαφορετικές πηγές (π.χ., κείμενο, εικόνα, ήχο), τα απλά RNNs μπορεί να μην είναι η βέλτιστη επιλογή, απαιτώντας την ανάπτυξη πιο σύνθετων αρχιτεκτονικών.

Αναγνωρίζοντας αυτούς τους περιορισμούς, η έρευνα στον τομέα των RNNs προσανατολίζεται προς διάφορες κατευθύνσεις για μελλοντική ανάπτυξη. Μία από τις πιο υποσχόμενες κατευθύνσεις είναι η ανάπτυξη υβριδικών αρχιτεκτονικών που συνδυάζουν τα πλεονεκτήματα των RNNs με άλλες αρχιτεκτονικές. Για παράδειγμα, ο συνδυασμός RNNs με CNNs μπορεί να επιτρέψει την αποτελεσματικότερη επεξεργασία χρονικά μεταβαλλόμενων χωρικών δεδομένων, όπως σε εφαρμογές αναγνώρισης δραστηριοτήτων από βίντεο. Επιπλέον, η ενσωμάτωση μηχανισμών προσοχής (attention mechanisms) σε RNNs έχει αποδειχθεί ιδιαίτερα αποτελεσματική, επιτρέποντας στο δίκτυο να εστιάζει επιλεκτικά σε συγκεκριμένα τμήματα της ακολουθίας εισόδου, βελτιώνοντας έτσι την απόδοση σε εφαρμογές όπως η μηχανική μετάφραση και η αναγνώριση ομιλίας.

Η ανάπτυξη βελτιωμένων αλγορίθμων εκπαίδευσης για RNNs αποτελεί επίσης μια σημαντική κατεύθυνση για μελλοντική έρευνα. Αυτό περιλαμβάνει την ανάπτυξη πιο αποδοτικών και σταθερών αλγορίθμων βελτιστοποίησης, καθώς και τεχνικών για την αντιμετώπιση των προβλημάτων εξαφανιζόμενης και εκρηκτικής κλίσης. Επιπλέον, η εφαρμογή τεχνικών μεταφοράς μάθησης (transfer learning) και προ-εκπαίδευσης (pre-training) σε RNNs μπορεί να βελτιώσει την απόδοσή τους σε εφαρμογές με περιορισμένα δεδομένα εκπαίδευσης.

Τέλος, η ανάπτυξη νευρομορφικών υλοποιήσεων των RNNs, δηλαδή υλοποιήσεων σε ειδικό υλικό που μιμείται πιο πιστά τη βιολογική λειτουργία του εγκεφάλου, αποτελεί μια συναρπαστική προοπτική για το μέλλον. Αυτές οι υλοποιήσεις θα μπορούσαν να επιτρέψουν την αποδοτικότερη επεξεργασία ακολουθιακών δεδομένων σε πραγματικό χρόνο, με σημαντικά μειωμένες απαιτήσεις ενέργειας σε σύγκριση με τις συμβατικές υλοποιήσεις σε γενικού σκοπού επεξεργαστές.

Συνολικά, παρά τους περιορισμούς τους, τα RNNs εξακολουθούν να αποτελούν ένα από τα πιο ισχυρά εργαλεία στο οπλοστάσιο της μηχανικής μάθησης για την επεξεργασία ακολουθιακών δεδομένων. Οι τρέχουσες και μελλοντικές εξελίξεις στην αρχιτεκτονική και την εκπαίδευση των RNNs υπόσχονται να διευρύνουν περαιτέρω το φάσμα των εφαρμογών τους και να βελτιώσουν την απόδοσή τους σε υπάρχουσες εφαρμογές.

3.8 Αναδρομικά Νευρωνικά Δίκτυα και Προβλήματα Βελτιστοποίησης

Πέρα από την επεξεργασία ακολουθιακών δεδομένων και χρονοσειρών, τα αναδρομικά νευρωνικά δίκτυα έχουν αποδειχθεί εξαιρετικά αποτελεσματικά στην επίλυση προβλημάτων βελτιστοποίησης. Η ικανότητά τους να λειτουργούν ως συστήματα δυναμικής μνήμης, σε συνδυασμό με τις μη γραμμικές ιδιότητές τους, τα καθιστά κατάλληλα για την αντιμετώπιση σύνθετων προβλημάτων βελτιστοποίησης, με χαρακτηριστικό παράδειγμα το πρόβλημα του Περιοδούντος Πωλητή (Travelling Salesman Problem - TSP).

Η σύνδεση μεταξύ των αναδρομικών νευρωνικών δικτύων και των προβλημάτων βελτιστοποίησης έγκειται στη δυνατότητα αντιστοίχισης της συνάρτησης ενέργειας του δικτύου στη συνάρτηση κόστους του προβλήματος βελτιστοποίησης. Ιδιαίτερα τα δίκτυα Hopfield, ως μια ειδική κατηγορία αναδρομικών νευρωνικών δικτύων, έχουν χρησιμοποιηθεί εκτενώς για την επίλυση προβλημάτων συνδυαστικής βελτιστοποίησης λόγω της εγγενούς τους τάσης να συγκλίνουν σε καταστάσεις ελάχιστης ενέργειας.

Η προσέγγιση αυτή επιτρέπει την αντιμετώπιση προβλημάτων NP-δύσκολων (NP-hard), τα οποία χαρακτηρίζονται από την εκθετική αύξηση του υπολογιστικού κόστους με την αύξηση του μεγέθους του προβλήματος. Αντί να αναζητούμε εξαντλητικά όλες τις πιθανές λύσεις, τα αναδρομικά νευρωνικά δίκτυα προσφέρουν ένα πλαίσιο για την αποτελεσματική εξερεύνηση του χώρου λύσεων και την εύρεση προσεγγιστικών λύσεων υψηλής ποιότητας σε εύλογο χρονικό διάστημα.

3.9 Ιδιότητες των RNNs στη Βελτιστοποίηση

Τα αναδρομικά νευρωνικά δίκτυα, και ιδιαίτερα τα δίκτυα Hopfield, διαθέτουν ορισμένες θεμελιώδεις ιδιότητες που τα καθιστούν εξαιρετικά κατάλληλα για την αντιμετώπιση προβλημάτων βελτιστοποίησης. Αυτές οι ιδιότητες πηγάζουν από τη δομή και τη λειτουργία τους και επιτρέπουν την αποτελεσματική αναπαράσταση και επίλυση πολύπλοκων προβλημάτων.

Μια από τις πιο σημαντικές ιδιότητες είναι η ύπαρξη μιας συνάρτησης ενέργειας που σχετίζεται άμεσα με την κατάσταση του δικτύου. Η συνάρτηση αυτή, γνωστή ως συνάρτηση Lyapunov, έχει τη θεμελιώδη ιδιότητα να μειώνεται μονοτονικά καθώς το δίκτυο εξελίσσεται σύμφωνα με τους κανόνες ενημέρωσής του. Αυτό εγγυάται ότι το δίκτυο θα συγκλίνει τελικά σε μια σταθερή κατάσταση που αντιστοιχεί σε ένα τοπικό

ελάχιστο της συνάρτησης ενέργειας. Η συνάρτηση ενέργειας μπορεί να σχεδιαστεί ώστε να αντιστοιχεί στη συνάρτηση κόστους του προβλήματος βελτιστοποίησης, με αποτέλεσμα οι σταθερές καταστάσεις του δικτύου να αντιστοιχούν σε λύσεις του προβλήματος.

Η δεύτερη σημαντική ιδιότητα είναι η ικανότητα των αναδρομικών νευρωνικών δικτύων να ενσωματώνουν περιορισμούς και προτιμήσεις στη διαδικασία βελτιστοποίησης. Αυτό επιτυγχάνεται μέσω του κατάλληλου σχεδιασμού των βαρών των συνδέσεων μεταξύ των νευρώνων. Οι περιορισμοί του προβλήματος μπορούν να ενσωματωθούν στη συνάρτηση ενέργειας με τέτοιο τρόπο ώστε οι καταστάσεις που παραβιάζουν τους περιορισμούς να έχουν υψηλή ενέργεια και επομένως να αποφεύγονται από το δίκτυο κατά τη διαδικασία σύγκλισης.

Επιπλέον, τα αναδρομικά νευρωνικά δίκτυα προσφέρουν εγγενή παραλληλισμό στην επεξεργασία πληροφοριών. Σε αντίθεση με τους συμβατικούς αλγόριθμους βελτιστοποίησης που αξιολογούν τις λύσεις σειριακά, τα αναδρομικά νευρωνικά δίκτυα μπορούν να επεξεργάζονται ταυτόχρονα πολλαπλές πτυχές του προβλήματος. Αυτό μπορεί να οδηγήσει σε σημαντική επιτάχυνση της διαδικασίας βελτιστοποίησης, ιδιαίτερα όταν υλοποιείται σε παράλληλο υλικό όπως οι επεξεργαστές γραφικών (GPUs) ή ειδικά σχεδιασμένα νευρομορφικά συστήματα.

Τέλος, τα αναδρομικά νευρωνικά δίκτυα διαθέτουν αξιοσημείωτη ευρωστία στην παρουσία θορύβου και ατελών δεδομένων. Η κατανεμημένη φύση της αναπαράστασης πληροφοριών στα νευρωνικά δίκτυα τα καθιστά ανθεκτικά σε τοπικές διαταραχές, επιτρέποντάς τους να διατηρούν καλή απόδοση ακόμη και όταν μέρος των δεδομένων εισόδου είναι ανακριβές ή ελλιπές. Αυτή η ιδιότητα είναι ιδιαίτερα σημαντική σε πρακτικές εφαρμογές, όπου τα δεδομένα συχνά περιέχουν θόρυβο ή ασάφειες.

Συνολικά, οι ιδιότητες αυτές καθιστούν τα αναδρομικά νευρωνικά δίκτυα ισχυρά εργαλεία για την αντιμετώπιση σύνθετων προβλημάτων βελτιστοποίησης, προσφέροντας μια εναλλακτική προσέγγιση στους παραδοσιακούς αλγόριθμους που συχνά αντιμετωπίζουν δυσκολίες με την κλιμάκωση σε μεγάλα προβλήματα ή την ενσωμάτωση πολύπλοκων περιορισμών.

3.10 Εφαρμογή RNNs στο Πρόβλημα του Περιοδεύοντος Πωλητή

Το πρόβλημα του Περιοδεύοντος Πωλητή (Traveling Salesman Problem - TSP) αποτελεί ένα από τα πιο γνωστά προβλήματα βελτιστοποίησης στην επιστήμη των υπολογιστών. Σε απλά λόγια, αναζητούμε την καλύτερη διαδρομή για να επισκεφθούμε πολλές τοποθεσίες μία φορά και να επιστρέψουμε στην αφετηρία με το μικρότερο δυνατό κόστος (απόσταση, χρόνο, κλπ.). Παραδοσιακά, το TSP αντιμετωπίζεται με διάφορες μεθόδους, αλλά τα τελευταία χρόνια έχουν αναπτυχθεί προσεγγίσεις που χρησιμοποιούν Αναδρομικά Νευρωνικά Δίκτυα (RNNs), συνδυάζοντας τη βαθιά μάθηση με τη συνδυαστική βελτιστοποίηση.

Τα Αναδρομικά Νευρωνικά Δίκτυα έχουν σχεδιαστεί για να επεξεργάζονται ακολουθίες δεδομένων και μπορούν να "θυμούνται" πληροφορίες από προηγούμενα βήματα. Αυτό τα καθιστά ιδανικά για το TSP, καθώς η επίλυσή του απαιτεί τον προσδιορισμό μιας ακολουθίας πόλεων[36]. Τα RNNs μπορούν να λαμβάνουν αποφάσεις για την επόμενη πόλη που θα επισκεφθούν βασιζόμενα σε όλες τις προηγούμενες επιλογές.

Η κεντρική ιδέα είναι να εκπαιδευσουμε ένα RNN ώστε να "μάθει" να προτείνει καλές διαδρομές για το TSP. Αντί να χρησιμοποιούμε εξαντλητικούς αλγορίθμους που δοκιμάζουν όλες τις πιθανές διαδρομές (που γίνεται πρακτικά αδύνατο για πολλές πόλεις), το νευρωνικό δίκτυο μαθαίνει μοτίβα και κανόνες που οδηγούν σε καλές λύσεις.

Μερικά πλεονεκτήματα της Προσέγγισης με RNNs είναι:

- **Η μάθηση από παραδείγματα:** Το σύστημα μπορεί να εκπαιδευτεί σε πολλά παραδείγματα TSP και να βελτιωθεί με τον καιρό
- **Η γρήγορη εξαγωγή λύσεων:** Μετά την εκπαίδευση, η παραγωγή διαδρομών είναι πολύ γρήγορη σε σύγκριση με παραδοσιακούς αλγορίθμους
- **Η Προσαρμοστικότητα:** Μπορεί να προσαρμοστεί σε διαφορετικές παραλλαγές του TSP (π.χ. με χρονικά παράθυρα ή περιορισμούς χωρητικότητας)
- **Και τέλος, αποτελεσματικότητα για μεγάλα προβλήματα:** Η προσέγγιση κλιμακώνεται καλά σε μεγάλο αριθμό πόλεων.

Ενώ μερικές προκλήσεις και Περιορισμοί είναι:

- **Η ποιότητα λύσης:** Τα RNNs συνήθως παράγουν καλές αλλά όχι πάντα βέλτιστες λύσεις
- **Καθώς και η ανάγκη για δεδομένα εκπαίδευσης:** Απαιτείται μεγάλος αριθμός παραδειγμάτων για την εκπαίδευση

Η προσέγγιση με RNNs για το TSP έχει εφαρμογές σε πολλούς τομείς:

- Δρομολόγηση οχημάτων και logistics
- Σχεδιασμός περιηγήσεων για τουρίστες
- Προγραμματισμός παραγωγής σε βιομηχανίες
- Σχεδιασμός κυκλωμάτων στην ηλεκτρονική

3.11 Επίλογος

Σε αυτό το κεφάλαιο παρουσιάστηκαν τα Αναδρομικά Νευρωνικά Δίκτυα (RNNs), τα οποία αποτελούν μια ισχυρή κατηγορία νευρωνικών δικτύων με δυνατότητα μοντελοποίησης χρονικά μεταβαλλόμενων δεδομένων. Η εισαγωγή της έννοιας της ανάδρασης και της μνήμης στα νευρωνικά δίκτυα επιτρέπει στα RNNs να επεξεργάζονται ακολουθιακά δεδομένα, αναγνωρίζοντας μοτίβα και συσχετίσεις που εξαρτώνται από προηγούμενες καταστάσεις. Επιπλέον, αναλύθηκε η αρχιτεκτονική των RNNs, εξηγώντας τη δομή τους, τον τρόπο λειτουργίας τους και τις διαφορετικές κατηγορίες τους.

Εξετάσαμε πώς τα αναδρομικά νευρωνικά δίκτυα διαφέρουν από τα κλασικά feedforward δίκτυα ως προς τη δυνατότητά τους να διατηρούν πληροφορίες μέσω του χρόνου, καθιστώντας τα κατάλληλα για εφαρμογές όπως η επεξεργασία φυσικής γλώσσας, η αναγνώριση ομιλίας, η πρόβλεψη χρονοσειρών και άλλα πεδία όπου η χρονική διάσταση των δεδομένων είναι σημαντική.

Η κατανόηση των αναδρομικών νευρωνικών δικτύων αποτελεί το θεμέλιο για την περαιτέρω μελέτη των δικτύων Hopfield, τα οποία θα αναλυθούν στο επόμενο κεφάλαιο. Τα δίκτυα Hopfield αποτελούν μια ειδική κατηγορία αναδρομικών νευρωνικών δικτύων με πλήρη συνδεσμολογία, που έχουν εφαρμογή σε προβλήματα βελτιστοποίησης όπως το Πρόβλημα του Περιοδευόντος Πωλητή. Η μετάβαση από τα γενικά

Κεφάλαιο 3ο

RNNs στα εξειδικευμένα δίκτυα Hopfield θα μας επιτρέψει να εμβαθύνουμε στους μηχανισμούς που καθιστούν δυνατή την επίλυση σύνθετων συνδυαστικών προβλημάτων με τη χρήση νευρωνικών δικτύων.

Κεφάλαιο 4^ο: Δίκτυα Hopfield

4.1 Εισαγωγή στα Δίκτυα Hopfield

Τα δίκτυα Hopfield αποτελούν μια ειδική κατηγορία αναδρομικών νευρωνικών δικτύων, τα οποία προτάθηκαν από τον John Hopfield το 1982[27]. Η καινοτομία του έγκειται στο συνδυασμό της νευροβιολογικής προσέγγισης με έννοιες παρμένες από τη φυσική και ειδικότερα από τα μοντέλα spin glass [37] της στατιστικής μηχανικής, δημιουργώντας ένα νευρωνικό δίκτυο με μοναδικές ιδιότητες[28].

Σε αντίθεση με τα συμβατικά αναδρομικά νευρωνικά δίκτυα που εξετάσαμε στο προηγούμενο κεφάλαιο, τα δίκτυα Hopfield διαθέτουν πλήρη συνδεσμολογία, όπου κάθε νευρώνας συνδέεται με όλους τους άλλους νευρώνες του δικτύου αλλά όχι με τον εαυτό του. Επιπλέον, οι συνδέσεις είναι συμμετρικές, δηλαδή το βάρος της σύνδεσης από το νευρώνα i στο νευρώνα j είναι ίδιο με το βάρος της σύνδεσης από το νευρώνα j στο νευρώνα i . Αυτή η συμμετρία είναι κρίσιμη για τη σταθερότητα του δικτύου και εξασφαλίζει τη σύγκλιση του σε σταθερές καταστάσεις, που αντιστοιχούν σε τοπικά ελάχιστα μιας συνάρτησης ενέργειας.

Η ιστορική σημασία των δικτύων Hopfield είναι σημαντική, καθώς η εργασία του Hopfield το 1982 αναζωογόνησε το ενδιαφέρον για τα νευρωνικά δίκτυα μετά από μια περίοδο στασιμότητας στο πεδίο. Αυτό συνέβη γιατί τα δίκτυα Hopfield έδειξαν ότι τα νευρωνικά δίκτυα μπορούν να χρησιμοποιηθούν για την επίλυση πρακτικών προβλημάτων βελτιστοποίησης και όχι μόνο για την αναγνώριση προτύπων. Δύο βασικές εφαρμογές των δικτύων Hopfield που αναδείχθηκαν είναι η συνειρμική μνήμη (associative memory) και η επίλυση προβλημάτων συνδυαστικής βελτιστοποίησης, όπως το Πρόβλημα του Περιοδευόντος Πωλητή (TSP).

Το μαθηματικό υπόβαθρο των δικτύων Hopfield τα καθιστά ιδιαίτερα χρήσιμα για προβλήματα βελτιστοποίησης, καθώς η δυναμική τους συμπεριφορά κατευθύνεται προς την ελαχιστοποίηση μιας συνάρτησης ενέργειας. Αυτή η ιδιότητα επιτρέπει την αντιστοίχιση προβλημάτων βελτιστοποίησης σε κατάλληλα σχεδιασμένες συναρτήσεις ενέργειας, η ελαχιστοποίηση των οποίων οδηγεί στη λύση του αρχικού προβλήματος.

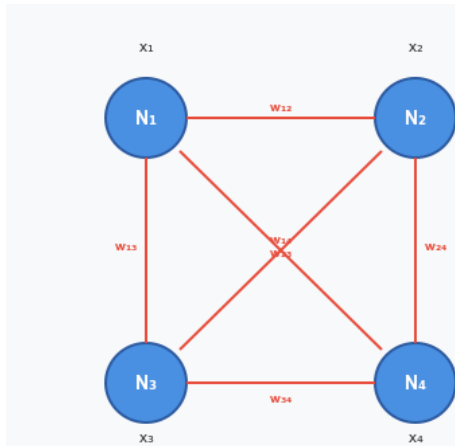
Στο παρόν κεφάλαιο, θα εξετάσουμε λεπτομερώς τη δομή και τη λειτουργία των δικτύων Hopfield, την έννοια της συνάρτησης ενέργειας, τις διαφορετικές εκδοχές τους (διακριτές και συνεχείς), καθώς και τις εφαρμογές τους στη συνειρμική μνήμη και στο Πρόβλημα του Περιοδευόντος Πωλητή. Τέλος, θα συζητήσουμε τους περιορισμούς και τις προκλήσεις που αντιμετωπίζουν αυτά τα δίκτυα στην πράξη.

4.2 Δομή δικτύου

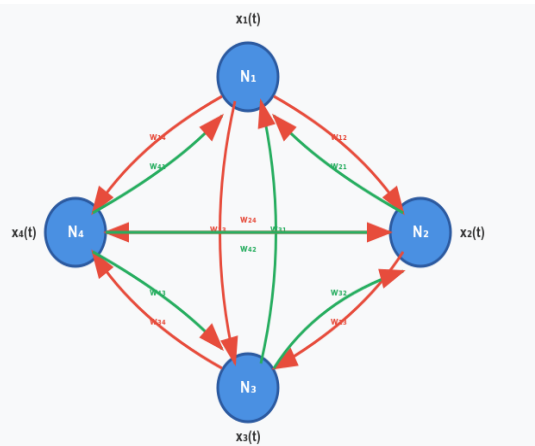
Σε μία πολύ γνωστή εργασία το 1982 ο Hopfield παρουσίασε [39] μια νέα κατηγορία δικτύων, τα οποία έχουν μεγάλες υπολογιστικές ικανότητες και είναι χρήσιμα σε δύο κατηγορίες προβλημάτων. Πρώτα, σε προβλήματα μνήμης συνειρμού (associative memory) και δεύτερον σε προβλήματα βελτιστοποίησης

(optimization). Η νέα αυτή μέθοδος πρέπει να πούμε ότι δεν έλυσε για πρώτη φορά γνωστά άλλα προβλήματα, καθότι όλα τα προβλήματα που ασχολείται είχαν λυθεί στο παρελθόν με άλλες μεθόδους. Μάλιστα οι λύσεις που προτείνονται δεν είναι καν οι καλύτερες, μερικές φορές είναι πιο αργές από άλλες παλαιότερες λύσεις. Εντούτοις η χρήση των δικτύων στα προβλήματα αυτά είναι πολύ χρήσιμη για την διορατικότητα που μας δίνει ως προς την φύση του προβλήματος που εξετάζουν. Η χρονική στιγμή που εμφανίστηκε η εργασία αυτή ήταν επίσης σημαντική, γιατί ήταν σε μία εποχή που η κατάσταση στα νευρωνικά δίκτυα ήταν στάσιμη και έτσι η εργασία αυτή ανακίνησε μεγάλο ενδιαφέρον και ενθουσιασμό σε αυτό το πεδίο, κάτι που ήταν εκ των πραγμάτων απαραίτητο [40]. Το μοντέλο του Hopfield είναι εμπνευσμένο από τη στατιστική φυσική και συγκεκριμένα από μοντέλα μαγνητικών υλικών όπου μικροί μαγνήτες υπό μορφή ατόμων είναι διατεταγμένοι σε κανονική δομή πλέγματος που αντιπροσωπεύει την κρυσταλλική δομή του υλικού. Στα μοντέλα αυτά βασικό χαρακτηριστικό των ατόμων είναι το 'spin' τους το οποίο μπορεί να πάρει δύο διακριτές κατευθύνσεις «πάνω» ή «κάτω». Αυτό αναπαρίσταται από μια δυαδική μεταβλητή που παίρνει τιμή 1 (αν το spin κοιτάει προς τα πάνω) ή -1 (αν το spin κοιτάει προς τα κάτω).

Τα δίκτυα τύπου Hopfield αποτελούνται από ένα μόνο επίπεδο με πολλές όμως μονάδες (νευρώνες). Στα δίκτυα αυτά όμως η έννοια των επιπέδων νευρώνων είναι διαφορετική από τα περισσότερα αλλά νευρωνικά δίκτυα της εποχής. Δεν υπάρχει χωριστό επίπεδο εισόδου ή εξόδου. Κάθε νευρώνας δέχεται σήματα από το περιβάλλον και έχει εξόδους προς αυτό. Κάθε νευρώνας είναι μία μονάδα όπως ο στοιχειώδης αισθητήρας, με τις ίδιες ιδιότητες. Στα δίκτυα Hopfield κάθε νευρώνας δίνει στην έξοδό του σήμα s_i προς όλες τις άλλες μονάδες j . Είναι δυαδικά συστήματα, ως προς το ότι κάθε μονάδα χαρακτηρίζεται από μία δυαδική κατάσταση, δηλαδή μπορεί να έχει μία από δύο δυνατές τιμές. Συνήθως οι τιμές αυτές είναι 0 και 1, αλλά αυτό δεν είναι απαραίτητο (θα μπορούσε, για παράδειγμα να είναι 1 και -1). Μία κατάσταση του συστήματος είναι πλήρης όταν δίδονται οι τιμές των μονάδων από τις οποίες αποτελείται, λ.χ. $s = (s_1, s_2, s_3 \dots s_n) = (1, 0, 1, \dots, 0)$ για ένα δίκτυο με n μονάδες. Οι μονάδες του δικτύου έχουν πλήρη συνδεσμολογία, δηλαδή κάθε μονάδα συνδέεται με κάθε άλλη μονάδα στο σύστημα. Στο δίκτυο Hopfield ισχύει ότι για ένα δίκτυο με n μονάδες θα έχουμε $n(n-1)$ συνδέσεις. Αυτό ισχύει διότι κάθε μονάδα συνδέεται με κάθε άλλη μονάδα, αλλά όχι με τον εαυτό της. Στην γενική περίπτωση οι συνδέσεις έχουν και συγκεκριμένη κατεύθυνση. Έτσι σε κάθε ζευγάρι μονάδων που συνδέονται υπάρχει σύνδεση και προς τις δύο κατευθύνσεις, δηλαδή μεταξύ των μονάδων i και j υπάρχει η σύνδεση w_{ij} και η σύνδεση w_{ji} . Γενικά, στα δίκτυα Hopfield θα έχουμε πάντοτε ότι $w_{ij} = w_{ji}$, πράγμα που σημαίνει ότι τα βάρη θα είναι ίσα για σήματα και προς τις δύο κατευθύνσεις. Τότε εξασφαλίζεται ότι το δίκτυο συγκλίνει και καταλήγει σε μία σταθερή κατάσταση. Αυτό το απέδειξαν οι Cohen-Grossberg[38], με ένα θεώρημα που αποδεικνύει ότι δίκτυα με μηχανισμό ανάδρασης είναι ευσταθή εάν ο πίνακας των βαρών W_{ij} είναι συμμετρική με 0 στην κύρια διαγώνιο της, δηλ $w_{ij} \neq w_{ji}$ για όλα τα $i = j$, και $w_{ii} = 0$, για όλα τα i



Σχήμα 4.1 Δομή δικτύου Hopfield με $n = 4$



Σχήμα 4.2 Άλλη παράσταση της δομής του δικτύου

Μία δομή δικτύου Hopfield φαίνεται στα Σχήματα 3.1 και 3.2. Το δίκτυο στα Σχήματα αυτά έχει 4 νευρώνες. Άρα, έχουμε ένα σύνολο από $n(n-1) = 4 \times 3 = 12$ συνάψεις. Στο Σχήμα 3.1 φαίνονται αυτές οι συνάψεις ως γραμμές που συνδέουν ζεύγη νευρώνων και έχουν κατευθυντικότητα. Στο Σχήμα 3.2 φαίνεται το ίδιο δίκτυο, αλλά με άλλη παράσταση, όπου και εδώ έχουμε 12 συνάψεις. Από το Σχήμα αυτό βλέπουμε ότι το δίκτυο Hopfield χρησιμοποιεί τον μηχανισμό ανάδρασης (feedback mechanism), καθόσον η έξοδος από κάθε νευρώνα ταυτόχρονα γίνεται είσοδος για τους άλλους νευρώνες του δικτύου. Η έξοδος υπολογίζεται πάλι και η διαδικασία αυτή επαναλαμβάνεται ξανά και ξανά. Εάν το δίκτυο είναι ευσταθές οι αλλαγές που επιτελούνται σε κάθε κύκλο γίνονται συνεχώς μικρότερες και μικρότερες, μέχρις ότου τελικά οι εξοδοί δεν αλλάζουν καθόλου και έτσι φτάσει σε μια σταθερή κατάσταση. Αν όμως το δίκτυο είναι ασταθές τότε, δεν συγκλίνει ποτέ και παρουσιάζει χαρακτηριστικά χαοτικού συστήματος.

4.3 Ανάκληση

Η ανάκληση στα δίκτυα αυτά γίνεται με τον παρακάτω τρόπο. Οι βασικές διεργασίες γίνονται σε κάθε νευρώνα χωριστά, οι παράμετροι που εισέρχονται είναι οι τιμές των εισερχομένων σημάτων και τα βάρη των συνάψεων. Γίνεται η γνωστή άθροιση των σημάτων s_i και βρίσκουμε το S . Το νέο κριτήριο είναι τώρα το εξής ότι εάν το S είναι θετικό, τότε η έξοδος είναι 1, ενώ εάν είναι αρνητικό τότε η έξοδος είναι 0. Αυτό σημαίνει ότι έχουμε και εδώ μια κρίσιμη τιμή, ένα κατώφλι με τιμή ίση με 0.

Το δίκτυο Hopfield έχει ως χαρακτηριστικό του ότι οι νευρώνες του συνεχώς αναπροσαρμόζονται. Αυτό αποτελεί την εκπαίδευση του δικτύου. Αρχικά μπορεί να βρίσκεται σε μία τυχαία κατάσταση, και ακολούθως κάθε νευρώνας αναπροσαρμόζει την κατάσταση του, ακολουθώντας κάποιους κανόνες, ως συνάρτηση του χρόνου (αυθαίρετες μονάδες χρόνου). Κατά την αναπροσαρμογή αυτή μερικές φορές η κατάσταση των νευρώνων αλλάζει, ενώ μερικές φορές παραμένει η ίδια. Η εκπαίδευση ολοκληρώνεται όταν δεν πραγματοποιούνται πλέον άλλες αλλαγές.

Κεφάλαιο 4ο

Το άθροισμα των εισερχόμενων σημάτων είναι:

$$S_j = \sum_{i=1}^n S_i W_{ij} \quad i \neq j \quad (4.1)$$

που είναι ίδιο με αυτό της εξίσωσης (4.1). Η σύγκριση τώρα γίνεται διαφορετικά. Εξετάζουμε εάν το S_j είναι θετικό ή αρνητικό. Εάν είναι θετικό, τότε η έξοδος $s_j = 1$, εάν είναι αρνητικό, τότε $s_j = 0$.

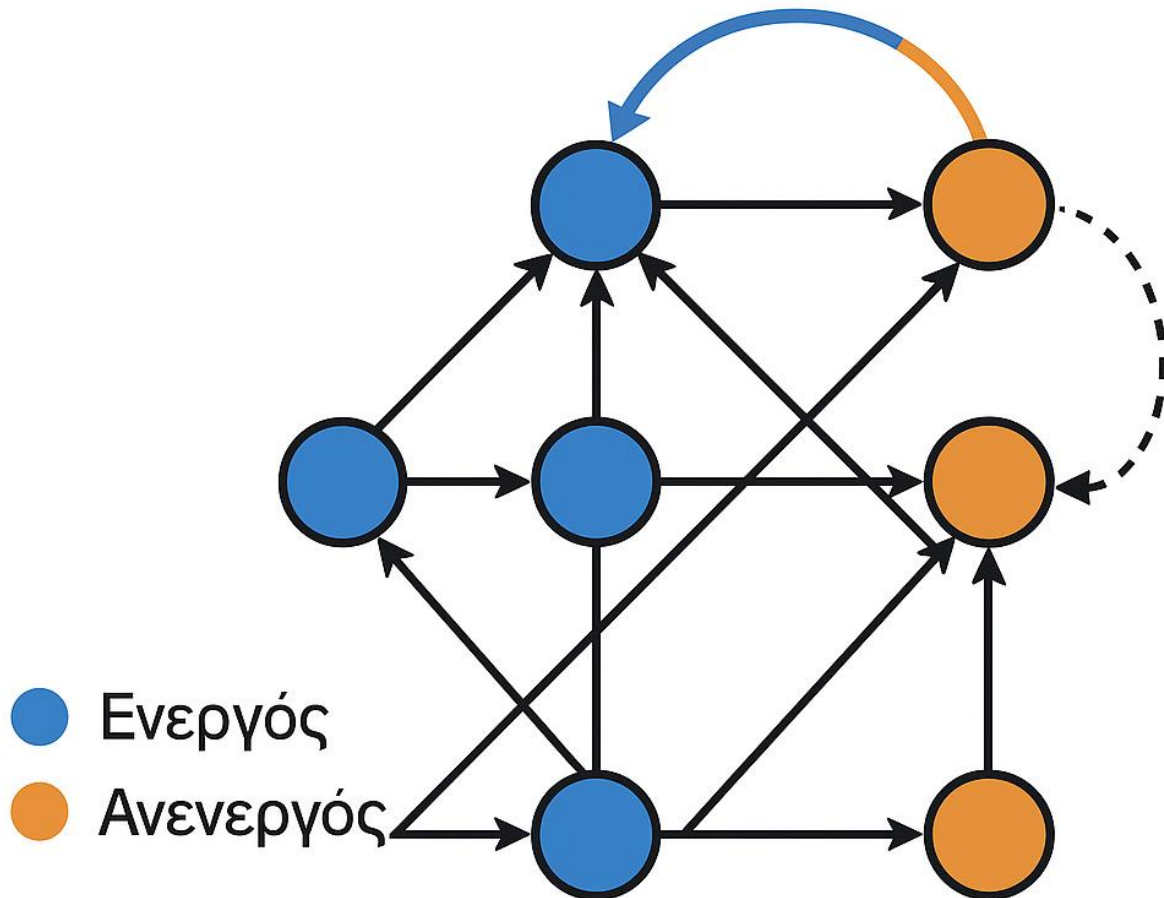
Εάν $S_j > 0$ τότε $s_j = 1$

Εάν $S_j < 0$ τότε $s_j = 0$ (4.2)

Εάν $S_j = 0$ τότε η μονάδα παραμένει στην προηγούμενη κατάσταση πριν την αναπροσαρμογή. Κάθε φορά που θέλουμε να αναπροσαρμόσουμε την τιμή ενός νευρώνα εφαρμόζουμε την εξίσωση 4.2.

Ας δούμε ένα παράδειγμα στο σχήμα 3.1. Στην χρονική στιγμή $t=1$ ο νευρώνας j έχει τιμή 0, καθ' ότι το άθροισμα $S_j = -0.1$, σύμφωνα με την εξίσωση 7.2. Στην χρονική στιγμή $t=2$ αναπροσαρμόζεται το δίκτυο με τέτοιο τρόπο ώστε η από τις τρεις εισόδους η πρώτη δίνει 0 αντί για 1. Αυτό έχει ως αποτέλεσμα να βγει άθροισμα $S_j = 0.9$, και έτσι ο νευρώνας j τώρα είναι σε κατάσταση με τιμή 1. Για κάθε αναπροσαρμογή πρέπει να βρούμε κάθε φορά το νέο άθροισμα και να κάνουμε την σύγκριση εκ νέου. Ένας τρόπος να γίνεται η αναπροσαρμογή είναι να ξεκινήσουμε από την πρώτη είσοδο και να αλλάζουμε τις τιμές μία-μία μέχρις ότου το σύστημα φτάσει σε μία σταθερή κατάσταση.

Ο Hopfield πρότεινε όμως οι αναπροσαρμογές να γίνονται με τυχαίο τρόπο. Δηλαδή δεν υπάρχει συγκεκριμένη σειρά με την οποία επιλέγονται οι νευρώνες, αλλά κάθε επιλογή γίνεται τυχαία.



Σχήμα 4.3 Αναπροσαρμογή των νευρώνων

Ο τρόπος αυτός έχει χαρακτηριστικά ασύγχρονης λειτουργίας, και είναι πιο κοντά στην πραγματικότητα για τα βιολογικά δίκτυα. Στα βιολογικά δίκτυα οι νευρώνες αναπροσαρμόζονται με τυχαία αλληλουχία, χωρίς να υπάρχει συσχετισμός από νευρώνα σε νευρώνα. Πάντως με την μέθοδο αυτή το δίκτυο πάντοτε καταλήγει σε σταθερή κατάσταση.

4.4 Συνάρτηση Ενέργειας

Το δίκτυο Hopfield ανήκει στην κατηγορία των αναδρομικών νευρωνικών δικτύων, όπου οι νευρώνες συνδέονται μεταξύ τους και επικοινωνούν μεταξύ τους μέσω των συνδέσεων. Αυτό το μοντέλο νευρωνικού δικτύου αναπτύχθηκε από τον John Hopfield το 1982. Το βασικό χαρακτηριστικό του είναι η ικανότητά του να μπορεί να μνημονεύει και να ανακαλεί πρότυπα.

Σε ένα δίκτυο Hopfield, κάθε νευρώνας είναι συνδεδεμένος με κάθε άλλο νευρώνα, συμπεριλαμβανομένου του εαυτού του, και κάθε σύνδεση έχει ένα βάρος. Τα πρότυπα αποθηκεύονται στα βάρη αυτών των

συνδέσεων. Η ανάκληση ενός προτύπου ξεκινά από μια αρχική κατάσταση και οδηγείται από την ενέργεια προς μια ευσταθή κατάσταση.

4.5 Θεωρητικό υπόβαθρο

Το δίκτυο Hopfield αποτελεί ένα αναδρομικό νευρωνικό δίκτυο, όπου κάθε νευρώνας μπορεί να βρίσκεται σε μία από δύο καταστάσεις: ενεργοποιημένος ή απενεργοποιημένος. Η κατάσταση κάθε νευρώνα αναπαρίσταται από ένα διάνυσμα, όπου κάθε στοιχείο είναι είτε 1 εάν ο νευρώνας είναι ενεργοποιημένος είτε -1 εάν είναι απενεργοποιημένος.

Η συνάρτηση ενέργειας E του δικτύου Hopfield ορίζεται ως εξής:

$$E = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N w_{ij} s_i s_j + \sum_{i=1}^N \theta_i s_i \quad (4.3)$$

όπου,

- N είναι ο αριθμός των νευρώνων,
- w_{ij} τα βάρη των συνδέσεων μεταξύ των νευρώνων i και j
- s_i η κατάσταση του νευρώνα i (είτε +1 εάν είναι ενεργοποιημένος είτε -1 εάν είναι απενεργοποιημένος),
- θ_i η επίδραση του εξωτερικού πεδίου στο νευρώνα i .

Η συνάρτηση ενέργειας αντιπροσωπεύει το συνολικό κόστος του δικτύου σε μια συγκεκριμένη κατάσταση. Στόχος του δικτύου είναι να μειώσει την ενέργεια προς εύρεση μιας ευσταθούς κατάστασης.

Η διαδικασία εκμάθησης του δικτύου Hopfield συνήθως βασίζεται στην αρχή του Hebbian[41]. Σύμφωνα με αυτήν την αρχή, η αλλαγή στα βάρη συνδέσεων εξαρτάται από τις ταυτόχρονες ενεργοποιήσεις των νευρώνων.

Με βάση αυτές τις αρχές, το δίκτυο Hopfield μπορεί να επιτύχει την αποθήκευση και ανάκληση πολλών προτύπων με χρήση της συνάρτησης ενέργειας για την αξιολόγηση της κατάστασης του δικτύου.

Η συνάρτηση ενέργειας του δικτύου Hopfield έχει ευρεία εφαρμογή σε πολλούς τομείς. Χρησιμοποιείται στην ανάκληση προτύπων, την ανίχνευση ανωμαλιών σε δεδομένα, καθώς και σε προβλήματα βελτιστοποίησης ενέργειας και στη μοντελοποίηση δυναμικών συστημάτων.

4.6 Συμπεράσματα

Η συνάρτηση ενέργειας του δικτύου Hopfield αποτελεί ένα βασικό εργαλείο στο πεδίο των νευρωνικών δικτύων. Η μαθηματική της ανάλυση μας επιτρέπει να κατανοήσουμε τη λειτουργία του δικτύου Hopfield και να εφαρμόσουμε αποτελεσματικά το μοντέλο σε ποικίλες εφαρμογές. Με περαιτέρω έρευνα και ανάπτυξη, η συνάρτηση ενέργειας μπορεί να βελτιωθεί και να επεκταθεί για να αντιμετωπίσει πιο σύνθετα προβλήματα και εφαρμογές.

4.7 Διακριτό δίκτυο Hopfield

Το Διακριτό Δίκτυο Hopfield αποτελεί ένα από τα πιο κλασικά μοντέλα νευρωνικών δικτύων, το οποίο χρησιμοποιείται για την ανάκληση προτύπων και την επίλυση προβλημάτων βελτιστοποίησης.

Μαθηματική Περιγραφή: Το Διακριτό Δίκτυο Hopfield αποτελείται από ένα σύνολο νευρώνων, όπου η κατάσταση κάθε νευρώνα μπορεί να είναι είτε ενεργοποιημένη (+1) είτε απενεργοποιημένη (-1). Αναπαρίσταται με ένα διάνυσμα S με στοιχεία στον χώρο $\{-1, +1\}$.

Αρχικά, ας εξετάσουμε την ενέργεια του δικτύου Hopfield πιο λεπτομερώς. Η ενέργεια E ορίζεται ως εξής:

$$E = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N w_{ij} s_i s_j + \sum_{i=1}^N \theta_i s_i \quad (4.4)$$

Ο πρώτος όρος της ενέργειας αντιπροσωπεύει τη συνολική αλληλεπίδραση μεταξύ των νευρώνων, όπου w_{ij} είναι τα βάρη των συνδέσεων μεταξύ των νευρώνων i και j . Ο δεύτερος όρος αντιπροσωπεύει την επίδραση του εξωτερικού πεδίου θ_i στον νευρώνα i .

Η ενέργεια χρησιμεύει ως κριτήριο για την αξιολόγηση της κατάστασης του δικτύου. Κατά τη διαδικασία ανανέωσης της κατάστασης των νευρώνων, προσπαθούμε να ελαχιστοποιήσουμε την ενέργεια του δικτύου.

Ο τρόπος με τον οποίο οι καταστάσεις των νευρώνων ενημερώνονται δίνεται από την εξίσωση:

$$s_i(t+1) = \text{sign}\left(\sum_{j=1}^N w_{ij} s_j(t) - \theta_i\right) \quad (4.5)$$

Αυτή η εξίσωση υπολογίζει τη νέα κατάσταση ενός νευρώνα i στο επόμενο χρονικό βήμα, λαμβάνοντας υπόψη τις καταστάσεις των γειτονικών νευρώνων και την επίδραση του εξωτερικού πεδίου.

Τέλος, για να αποθηκεύσουμε πρότυπα μνήμης στο δίκτυο, χρησιμοποιούμε την εξίσωση:

$$w_{ij} = \frac{1}{N} \sum_{\mu=1}^P P_i^{\mu} P_j^{\mu} \quad (4.6)$$

Αυτή η εξίσωση υπολογίζει τα βάρη w_{ij} με βάση τα πρότυπα μνήμης P_i^{μ} . Οι πρόσθετες λεπτομέρειες αυτές ενισχύουν την κατανόηση της μαθηματικής περιγραφής του δικτύου Hopfield.

4.8 Συνεχές Δίκτυο Hopfield

Το συνεχές δίκτυο Hopfield (CHN) είναι ένα αρχετυπικό δίκτυο ανατροφοδότησης, όπου όλοι οι νευρώνες είναι τόσο είσοδοι όσο και έξοδοι στο CHN. Ας υποθέσουμε ότι υπάρχουν n νευρώνες στο CHN και κάθε νευρώνας συνδέεται μεταξύ τους. Οι καταστάσεις όλων των νευρώνων μπορούν να συμβολίζονται με $u = [u_1, \dots, u_n]$. Το CHN μπορεί να ενημερώσει όλους τους νευρώνες ταυτόχρονα με την ακόλουθη φόρμα:

$$u(t+1) = u(t) + \frac{du}{dt} \Delta t,$$

$$u = g(u), \quad (4.7)$$

Όπου, Δt και υποδηλώνουν μια σταθερά και ένα διάνυσμα

κατάστασης, αντίστοιχα. Το $\frac{du}{dt}$ αποφασίζεται από την ακόλουθη εξίσωση:

$$\frac{du}{dt} = -\frac{u}{t} + Tu + 1, \quad (4.8)$$

Όπου, $T \in R^{n \times n}$ και $I \in R^{n \times 1}$ δηλώνουν έναν συμμετρικό πίνακα και έναν πίνακα πόλωσης, αντίστοιχα. Το $g(u_i)$ της Εξ. (4.10) είναι εκφράζεται ως εξής:

$$u_i = g(u_i) = \frac{1}{2} \left(1 + \tanh \left(\frac{u_i}{u_0} \right) \right), u_0 > 0, i = 1, 2 \dots n, \quad (4.9)$$

Όπου, το u_0 είναι θετική σταθερά. Για να ικανοποιηθεί η ιδιότητα σύγκλισης του CHN synchronous, θα πρέπει να πληρούνται δύο προϋποθέσεις. Πρώτον, κανένας νευρώνας δεν θα πρέπει να υπάρχει

αυτοανατροφοδότηση. Δεύτερον, το βάρος σύνδεσης μεταξύ των νευρώνων T_{ij} και T_{ji} και θα πρέπει να είναι το ίδιο. Γενικά, η ενεργειακή συνάρτηση [42] του CHN περιγράφεται από :

$$E = \frac{1}{2} u^t T u - (i^b)^t u. \quad (4.10)$$

Για την εξ. (3), υπάρχουν δύο τρόποι ενημέρωσης (δηλαδή, ασύγχρονος ή σύγχρονος). Η ασύγχρονη λειτουργία σημαίνει ότι κάθε νευρώνας v_i μπορεί να ενημερώνεται διαδοχικά. Η σύγχρονη λειτουργία μπορεί να ενημερώσει όλους τους νευρώνες ταυτόχρονα. Οι δύο τρόποι ενημέρωσης έχουν μελετηθεί εκτενώς στο [39][43][44]. Σε αυτό το άρθρο, χρησιμοποιούμε τη σύγχρονη λειτουργία. Το T της Εξ. Το (4) πρέπει να πληροί τις ακόλουθες δύο προϋποθέσεις: (1) οι τιμές των διαγώνιων στοιχείων πρέπει να είναι μηδενικές. (2) Το T πρέπει να είναι συμμετρικό. Η αρχική τιμή του v αρχικοποιείται τυχαία. Επομένως, το CHN μπορεί να επιτύχει διαφορετικές τοπικές βέλτιστες λύσεις με διαφορετικές αρχικές τιμές. Με άλλα λόγια, η CHN δεν μπορεί να αναζητήσει μια συνολική βέλτιστη λύση. Στην επόμενη υποενοότητα, παρουσιάζουμε το Particle Swarm Optimization για την αναζήτηση μιας συνολικής βέλτιστης λύσης.

4.9 Το δίκτυο Hopfield και το πρόβλημα του περιοδεύοντος πωλητή

Η δεύτερη κατηγορία προβλημάτων στα οποία χρησιμοποιούνται τα δίκτυα Hopfield είναι τα προβλήματα βελτιστοποίησης. Η βελτιστοποίηση είναι μία ευρεία περιοχή των τεχνικών επιστημών. Είναι ο κλάδος που, όταν έχουμε ένα πρόβλημα με πολλές συναφείς λύσεις, προσπαθεί να βρει την καλύτερη (τη βέλτιστη) δυνατή. Μερικά τέτοια προβλήματα είναι η βέλτιστη διαδρομή σήματος σε ένα ηλεκτρονικό κύκλωμα, η διαδρομή που πρέπει να ακολουθεί ο βραχίονας ενός ρομπότ στην γραμμή παραγωγής, κτλ. Το να βρεθεί η καλύτερη λύση δεν είναι πάντα εύκολο, αλλά σήμερα υπάρχουν πολλοί τρόποι που αντιμετωπίζουν παρόμοια προβλήματα. Ο Hopfield έδειξε πως μπορεί ένα νευρωνικό δίκτυο να χρησιμοποιηθεί για να δώσει άλλη μία λύση σε προβλήματα της κατηγορίας αυτής. Θα εξετάσουμε την τεχνική αυτή με ένα παράδειγμα, αυτό του πλανόδιου πωλητή [4, 6, 15].

Το πρόβλημα του πλανόδιου πωλητή (traveling salesman problem, TSP) είναι από τα πιο γνωστά προβλήματα στην περιοχή της βελτιστοποίησης και από τα πιο δύσκολα να λυθούν. Έχει λυθεί με διάφορες μεθόδους, αλλά καμία δεν είναι εύκολη και άμεση. Ανήκει στην κατηγορία των προβλημάτων τύπου «NP complete» (non-deterministic polynomial)[21], τα οποία δεν έχουν ακριβείς λύσεις, και έτσι οι λύσεις που προτείνονται είναι «ευρετικές λύσεις». Πολλές φορές μάλιστα σε τέτοια προβλήματα, παρ' όλο που μία συγκεκριμένη λύση είναι η πλέον σωστή, επειδή είναι πολύ δύσκολο ή αδύνατο να την βρούμε, είμαστε ικανοποιημένοι αν βρούμε μία άλλη λύση που είναι αρκετά καλή, έστω και αν δεν είναι η καλύτερη.

Το πρόβλημα TSP ορίζεται ως εξής:

Έχουμε μία ομάδα από πόλεις σε έναν χάρτη με συγκεκριμένες συντεταγμένες που πρέπει να επισκεφθεί ένας πλανόδιος πωλητής ακολουθώντας μια συγκεκριμένη διαδρομή. Οι περιορισμοί που υπάρχουν στη διαδρομή είναι οι εξής: Ο πλανόδιος πωλητής πρέπει να επισκεφθεί κάθε πόλη μόνον μία φορά και στο

Κεφάλαιο 4ο

τέλος πρέπει να επιστρέψει στην αρχική πόλη. Ζητείται να βρούμε τη διαδρομή εκείνη η οποία είναι η μικρότερη.

Ο Hopfield πρότεινε μία λύση χρησιμοποιώντας ένα δίκτυο του τύπου που περιγράφεται παραπάνω, και ακολουθεί την παρακάτω διαδικασία: Η λύση περιλαμβάνει την επίσκεψη σε μία ομάδα από n πόλεις. Θέλουμε να αναπαραστήσουμε την ομάδα αυτή σε ένα νευρωνικό δίκτυο. Κάθε πόλη είναι ένας νευρώνας σε μία σειρά n νευρώνων. Μόνο ένας νευρώνας σε κάθε σειρά μπορεί να είναι 1, ενώ όλοι οι άλλοι πρέπει να είναι 0. Ο νευρώνας που είναι ίσος με 1 δείχνει την σειρά με την οποία θα επισκεφθεί την συγκεκριμένη πόλη ο πλανόδιος πωλητής. Ο πίνακας 4.2 δείχνει μια τέτοια παράσταση για το πρόβλημα με 10 πόλεις. Οι πόλεις είναι A, B, C, D, ..., J. Οι στήλες του πίνακα δείχνουν πότε (δηλ. με ποια σειρά) γίνεται η επίσκεψη στην αντίστοιχη συγκεκριμένη πόλη. Έτσι η διαδρομή αρχίζει από την πόλη G, ακολούθως, B, J, A, D, H, C, F, I, E.

Πίνακας 4.2: Αναπαράσταση λύσης με 10 πόλεις.

Πόλη	1	2	3	4	5	6	7	8	9	10
A	0	0	0	1	0	0	0	0	0	0
C	0	0	0	0	0	0	1	0	0	0
D	0	0	0	0	1	0	0	0	0	0
E	0	0	0	0	0	0	0	0	0	1
F	0	0	0	0	0	0	0	1	0	0
G	1	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	1	0	0	0	0
I	0	0	0	0	0	0	0	0	1	0
J	0	0	1	0	0	0	0	0	0	0

Το σύστημα αυτό απαιτεί n^2 νευρώνες, και βλέπουμε ότι ο αριθμός αυτός αυξάνεται πολύ γρήγορα με το n . Καθ' ότι ο πλανόδιος πωλητής επισκέπτεται κάθε πόλη μία μόνο φορά, και μπορεί να επισκεφθεί μία μόνο πόλη την φορά (δεν μπορεί να βρίσκεται ταυτόχρονα σε δύο πόλεις) υπάρχει μόνον ένα 1 σε κάθε σειρά και σε κάθε στήλη του Πίνακα 5.1. Το πρόβλημα με n πόλεις έχει $n!/(2n)$ διαφορετικές διαδρομές[3], επομένως έχει αυτό τον αριθμό των λύσεων. Αν $n = 10$, τότε υπάρχουν 181440 διαφορετικές διαδρομές, ενώ αν $n = 60$, τότε υπάρχουν $6,9 \times 10^{79}$ διαφορετικές διαδρομές. Βλέπουμε λοιπόν ότι το μέγεθος του προβλήματος είναι εξαιρετικά μεγάλο, λόγω του παραγοντικού που έχει ο αριθμός των διαδρομών.

Στην λύση που προτείνεται κάθε νευρώνας έχει δύο δείκτες. Ο πρώτος δείκτης δείχνει την πόλη και ο δεύτερος δείκτης δείχνει την σειρά με την οποία ο πλανόδιος πωλητής θα την επισκεφθεί. Έτσι, το στοιχείο V_{xi} λέει ότι η πόλη x είναι η $i^{\text{στη}}$ πόλη στην σειρά επίσκεψης. Η συνάρτηση της παραμέτρου E ορίζεται μόνον για τις λύσεις αυτές που δίνουν μόνο ένα 1 σε κάθε στήλη και κάθε σειρά[27]. Επίσης πρέπει να προτιμώνται οι λύσεις που έχουν την μικρότερη διαδρομή. Έτσι η παράμετρος E θα δίνεται από:

$$E = \binom{A}{2} \sum_x \sum_i \sum_j V_{xi} V_{xj} \quad j \neq i \quad \text{1ος όρος}$$

$$\begin{aligned}
 & + \left(\frac{B}{2}\right) \sum_i \sum_x \sum_y V_{xi} V_{yi} \quad y \neq x && \text{2ος όρος} \\
 & + \left(\frac{C}{2}\right) \sum_x \sum_i (V_{xi} - n^2) && \text{3ος όρος} \\
 & + \left(\frac{D}{2}\right) \sum_x \sum_y d_{xy} s_{xi} (V_{x,i+1} + V_{y,i-1}) && \text{4ος όρος}
 \end{aligned}
 \tag{4.11}$$

όπου A, B, C και D είναι παράμετροι (σταθερές), $V_{x,i}$ είναι ένα στοιχείο του Πίνακα, στην σειρά x, και στην στήλη i.

Παρατηρούμε ότι η εξίσωση **4.11** έχει συνολικά 4 όρους. Κάθε όρος στην εξίσωση αυτή εισάγει κάποιους περιορισμούς τους οποίους έχει εξ ορισμού το πρόβλημα TSP. Κάθε περιορισμός επιβάλλεται με την ελαχιστοποίηση του αντίστοιχου όρου. Ο πρώτος όρος επιτρέπει μία μόνο επίσκεψη σε κάθε πόλη. Ο όρος αυτός έχει μικρή τιμή (είναι 0) μόνο όταν κάθε σειρά έχει μόνο ένα 1. Ο δεύτερος όρος δεν επιτρέπει στον πλανόδιο πωλητή να βρίσκεται σε δύο διαφορετικές πόλεις την ίδια στιγμή. Ο όρος αυτός είναι μικρός (είναι 0) μόνον όταν κάθε στήλη έχει μόνο ένα 1. Ο τρίτος όρος επιτρέπει μόνον n πόλεις να υπάρχουν στην διαδρομή. Ο όρος αυτός είναι μικρός (είναι 0) όταν υπάρχουν ακριβώς n αριθμοί από 1 στον Πίνακα. Ο τέταρτος όρος παριστάνει το συνολικό μήκος της διαδρομής που πρέπει να ελαχιστοποιηθεί. Οι τιμές των σταθερών A, B και C πρέπει να είναι μεγάλες, έτσι ώστε οι καταστάσεις που έχουν χαμηλή ενέργεια να παριστάνουν «καλές» διαδρομές. Η μεγάλη τιμή στο D εξασφαλίζει ότι θα βρεθεί μία σύντομη διαδρομή. Το επόμενο βήμα είναι να βρεθούν τα βάρη w.

Έτσι έχουμε:

$$W_{xi,yi} = A\delta_{xy}(1 - \delta) - B\delta_{ij}(1 - \delta_{xy}) - C - Dd_{xy}(\delta_{j,i+1} + \delta_{j,i-1})
 \tag{4.12}$$

όπου δ_{ij} είναι το Kronecker, δ_{ij} εάν $i = j$ και $\delta_{ij} = 0$ εάν $i \neq j$. Ο όρος A δεν επιτρέπει πλέον του ενός 1 σε μία σειρά, ο όρος B δεν επιτρέπει πλέον του ενός 1 σε μία στήλη, ο όρος C δίνει έναν συνολικό περιορισμό, ενώ τέλος ο όρος D είναι όρος απόστασης. Επιπλέον, κάθε νευρώνας έχει μία εξωτερική ορμή (bias):

$$I_{xj} = C_n \quad (4.13)$$

όπου C είναι η παραπάνω σταθερά και n είναι ο αριθμός των πόλεων. Η αρχική κατάσταση του νευρωνικού δικτύου είναι τυχαία και το δίκτυο υφίσταται συνεχείς αναπροσαρμογές μέχρις ότου συγκλίνει σε μία σταθερή κατάσταση. Όταν επιτευχθεί η σύγκλιση τότε η διαδρομή του πλανόδιου πωλητή δίδεται από την τελική κατάσταση του δικτύου.

Το δίκτυο Hopfield λύνει το πρόβλημα TSP ικανοποιητικά για τιμές έως $n = 30$, αλλά έχει προβλήματα για μεγαλύτερες τιμές. Μας δίνει όμως ένα πολύ καλό παράδειγμα προβλημάτων που λύνουν τα νευρωνικά δίκτυα. Οι Hopfield–Tank έλυσαν το 1985 το πρόβλημα TSP για $n = 10$, χρησιμοποιώντας την εξής σιμοειδή συνάρτηση:

$$S = \frac{1}{2} \left[1 + \tanh \left(\frac{S}{u_0} \right) \right] \quad (4.14)$$

όπου s είναι η έξοδος του δικτύου, S είναι η γνωστή μας ποσότητα το άθροισμα των γινομένων (εξίσωση 3.1), και u_0 είναι μία σταθερά. Το αποτέλεσμα ήταν ότι 16 από τις 20 πραγματοποιήσεις συνέκλιναν σε σωστές διαδρομές, ενώ το 50% των λύσεων ήταν «καλές», δηλ. οι προτεινόμενες διαδρομές ήταν από τις πλέον σύντομες. Από τον τρόπο λειτουργίας του δικτύου στο πρόβλημα TSP, το δίκτυο Hopfield έχει χαρακτηριστικά παράλληλης αρχιτεκτονικής, πράγμα που θα μπορούσε να είναι χρήσιμο σε αντίστοιχες πειραματικές διατάξεις. Ο ασύγχρονος τρόπος αναπροσαρμογής των στοιχείων του δικτύου είναι μία ιδιότητα που μόνο στα δίκτυα αυτά εμφανίζεται και μπορεί να είναι ιδιαίτερα χρήσιμο στοιχείο όταν αναζητούμε γρηγορότερες ταχύτητες λειτουργίας των επεξεργαστών.

Αργότερα ο Hopfield χρησιμοποίησε παρόμοιες δομές και για μια άλλη εφαρμογή: κατασκεύασε ένα δίκτυο που επιτελεί μία αναλογική σε ψηφιακή μετατροπή τεσσάρων δυαδικών ψηφίων (4 bits). Οι αντιστάσεις είναι τα βάρη w . Οι νευρώνες είναι ενισχυτές και δίνουν και κανονική και ανεστραμμένη (inverted) έξοδο, ώστε τα w να μπορεί να είναι και αρνητικά. Το δίκτυο αυτό χρησιμοποιεί 4 νευρώνες και λειτουργεί με επιτυχία.

4.10 Επίλογος

Στο παρόν κεφάλαιο μελετήσαμε τα δίκτυα Hopfield, μια ειδική κατηγορία αναδρομικών νευρωνικών δικτύων με εφαρμογές στη συνειρμική μνήμη και στην επίλυση προβλημάτων βελτιστοποίησης. Παρουσιάσαμε τη δομή και τη λειτουργία των δικτύων αυτών, εξετάζοντας τόσο τα διακριτά όσο και τα συνεχή μοντέλα, και αναλύσαμε τον τρόπο με τον οποίο η συνάρτηση ενέργειας καθοδηγεί τη δυναμική συμπεριφορά τους προς σταθερές καταστάσεις που αντιστοιχούν σε τοπικά ελάχιστα.

Ιδιαίτερη έμφαση δόθηκε στην εφαρμογή των δικτύων Hopfield στο Πρόβλημα του Περιοδούοντος Πωλητή, όπου δείξαμε πώς μπορεί να μετατραπεί ένα συνδυαστικό πρόβλημα βελτιστοποίησης σε μια κατάλληλη συνάρτηση ενέργειας, η ελαχιστοποίηση της οποίας οδηγεί σε μια προσέγγιση της βέλτιστης λύσης. Παράλληλα, συζητήσαμε τους περιορισμούς των δικτύων Hopfield, όπως τα ζητήματα με τα τοπικά

ελάχιστα, τους περιορισμούς στη χωρητικότητα μνήμης και τις προκλήσεις κλιμάκωσης για μεγάλα προβλήματα.

Τα δίκτυα Hopfield αποτελούν ένα σημαντικό ορόσημο στην ιστορία των νευρωνικών δικτύων, καθώς συνέδεσαν έννοιες από τη φυσική και τη νευροβιολογία, προσφέροντας νέες προοπτικές για την αντιμετώπιση πολύπλοκων προβλημάτων βελτιστοποίησης. Παρά τους περιορισμούς τους, εξακολουθούν να αποτελούν αντικείμενο έρευνας και ανάπτυξης, με νεότερες προσεγγίσεις που επιχειρούν να ξεπεράσουν τις εγγενείς αδυναμίες τους.

Στο επόμενο κεφάλαιο, θα παρουσιάσουμε μια σειρά από προτεινόμενες λύσεις που βασίζονται σε δίκτυα Hopfield για την αντιμετώπιση του Προβλήματος του Περιοδευόντος Πωλητή. Θα εξετάσουμε διάφορες προσεγγίσεις που έχουν αναπτυχθεί για τη βελτίωση της απόδοσης των δικτύων Hopfield σε αυτό το πρόβλημα, συμπεριλαμβανομένων των τεχνικών βελτιστοποίησης, των υβριδικών μοντέλων και των στρατηγικών διαίρει και βασίλευε. Μέσα από αυτή την ανάλυση, θα αναδείξουμε τις δυνατότητες και τις προοπτικές των δικτύων Hopfield στην επίλυση πρακτικών προβλημάτων βελτιστοποίησης.

Κεφαλαίο 5^ο: Παρουσίαση προτεινόμενων λύσεων

5.1 Εισαγωγή: Συνεχές Νευρωνικό Δίκτυο Hopfield Βασισμένο σε Δυναμικό Βήμα για το Πρόβλημα του Περιοδεύοντος Πωλητή

Στην εργασία των Zhong, Chu, Luo και Gan (2017)[45], προτείνεται μια μέθοδος βελτιστοποίησης του συνεχούς νευρωνικού δικτύου Hopfield (CHNN) για την επίλυση του προβλήματος του περιοδεύοντος πωλητή (TSP). Το πρόβλημα του περιοδεύοντος πωλητή αποτελεί ένα κλασικό πρόβλημα συνδυαστικής βελτιστοποίησης με ευρεία εφαρμογή σε διάφορους τομείς, συμπεριλαμβανομένου του προγραμματισμού διαδρομών για ρομπότ. Λόγω της υπολογιστικής του πολυπλοκότητας, η χρήση νευρωνικών δικτύων έχει προταθεί ως αποτελεσματική προσέγγιση για την εύρεση προσεγγιστικών λύσεων.

5.1.1 Μεθοδολογία

Οι ερευνητές επικεντρώνονται στην επίλυση ενός θεμελιώδους προβλήματος των παραδοσιακών υλοποιήσεων CHNN: της αμοιβαίας περιοριστικής σχέσης μεταξύ της ακρίβειας σύγκλισης και της ταχύτητας σύγκλισης του δικτύου. Στο κλασικό CHNN, το μέγεθος βήματος παραμένει σταθερό κατά την επαναληπτική διαδικασία, δημιουργώντας έναν συμβιβασμό μεταξύ ταχύτητας και ακρίβειας:

$$u_i(t + 1) = u_i(t) + step(t) \times \frac{dE}{dt} \quad (5.1)$$

Όπου:

- u_i είναι η κατάσταση του νευρώνα i
- $step$ είναι το σταθερό μέγεθος βήματος
- $\frac{dE}{dt}$ είναι η παράγωγος της ενεργειακής συνάρτησης

Η καινοτομία της προτεινόμενης μεθόδου έγκειται στην εισαγωγή ενός δυναμικού μεγέθους βήματος που μεταβάλλεται ανάλογα με την πρόοδο της επαναληπτικής διαδικασίας:

$$step(t) = step_0 \left(1 - r \times \tan sig \left(t \times \frac{a}{L} \right) \right) \quad (5.2)$$

Όπου:

- $step_0$ είναι η αρχική τιμή του βήματος
- t είναι ο τρέχων αριθμός επανάληψης
- L είναι ο μέγιστος αριθμός επαναλήψεων
- r και a είναι παράμετροι ελέγχου της συνάρτησης αλλαγής

Η λογική της προσέγγισης είναι η εξής: στα αρχικά στάδια της διαδικασίας, χρησιμοποιείται μεγαλύτερο βήμα για ταχύτερη σύγκλιση, ενώ καθώς το δίκτυο πλησιάζει στο σημείο σύγκλισης, το βήμα μειώνεται σταδιακά, επιτρέποντας μεγαλύτερη ακρίβεια στην τελική λύση και αποφεύγοντας την υπερπήδηση της βέλτιστης λύσης.

5.1.2 Πειραματική Αξιολόγηση

Η αξιολόγηση της προτεινόμενης μεθόδου πραγματοποιήθηκε σε δύο πειραματικές ρυθμίσεις:

α) Πρόβλημα 10 πόλεων

Τα πειραματικά αποτελέσματα έδειξαν για τυχαία αρχική διαδρομή, συνολική απόσταση 3.8987, για CHNN με σταθερό βήμα, συνολική απόσταση 2.371 και για CHNN με δυναμικό βήμα, συνολική απόσταση 2.2316. Η βελτίωση του CHNN με δυναμικό βήμα έναντι του CHNN με σταθερό βήμα είναι 0.1394 μονάδες (περίπου 5.9%). Σε όρους ενεργειακής συνάρτησης, το CHNN με δυναμικό βήμα πέτυχε τιμή 199.2551, ενώ το CHNN με σταθερό βήμα πέτυχε 210.7254.

β) Πρόβλημα 20 πόλεων

Για το μεγαλύτερο πρόβλημα των 20 πόλεων, τα αποτελέσματα έδειξαν για CHNN με σταθερό βήμα, συνολική απόσταση 4.123 και για CHNN με δυναμικό βήμα, συνολική απόσταση 3.967. Η βελτίωση είναι 0.156 μονάδες (περίπου 3.8%). Σε όρους ενεργειακής συνάρτησης, το CHNN με δυναμικό βήμα πέτυχε τιμή 376.6066, ενώ το CHNN με σταθερό βήμα πέτυχε 381.9886.

5.1.3 Συμπεράσματα

Τα αποτελέσματα της έρευνας καταδεικνύουν ότι το συνεχές νευρωνικό δίκτυο Hopfield με δυναμικό βήμα αποτελεί μια αποτελεσματική προσέγγιση για την επίλυση του προβλήματος του περιοδεύοντος πωλητή. Η προτεινόμενη μέθοδος επιτυγχάνει τη ταχύτερη σύγκλιση κατά τα αρχικά στάδια της διαδικασίας, τη μεγαλύτερη ακρίβεια στη τελική λύση καθώς και αποτελεσματικότερη αποφυγή παγίδευσης σε τοπικά ελάχιστα.

Η βελτίωση της απόδοσης είναι εμφανής τόσο στο πρόβλημα των 10 πόλεων όσο και στο πρόβλημα των 20 πόλεων, υποδεικνύοντας την καταλληλότητα της μεθόδου για διάφορα μεγέθη προβλημάτων TSP. Ωστόσο, η συγκεκριμένη εργασία περιορίζεται στον αριθμό των πόλεων και οι προσομοιώσεις που γίνονται είναι έως 20 πόλεις, καθώς η μόνη αλλαγή από το κλασικό CHNN είναι η μετατροπή του βήματος από σταθερό σε δυναμικό, πράγμα το οποίο δεν επαρκεί για ικανοποιητικά αποτελέσματα σε προβλήματα μεγαλύτερης κλίμακας.

5.2 Σχεδιασμός και Βελτίωση του Δικτύου Hopfield για το Πρόβλημα του Περιοδεύοντος Πωλητή

Στην εργασία του Yanfen Luo (2019)[46], παρουσιάζεται μια μέθοδος βελτίωσης της απόδοσης του νευρωνικού δικτύου Hopfield για την επίλυση του προβλήματος του περιοδεύοντος πωλητή (TSP). Το TSP αποτελεί ένα κλασικό πρόβλημα συνδυαστικής βελτιστοποίησης με πολλές πρακτικές εφαρμογές, όπως ο σχεδιασμός ασφαλών και αποτελεσματικών δικτύων μεταφοράς και ο προγραμματισμός διαδρομών εφοδιαστικής αλυσίδας.

Ο αριθμός των πιθανών διάδρομων στο TSP αυξάνεται εκθετικά όσο μεγαλώνει ο αριθμός των πόλεων N και η βέλτιστη λύση είναι δύσκολο να υπολογιστεί. Λόγω της κακής σύγκλισης και των μη έγκυρων λύσεων του δικτύου Hopfield στο TSP, προτείνεται ως λύση η βελτίωση της ενεργειακής συνάρτησης του δικτύου Hopfield και η εφαρμογή του γενετικού αλγορίθμου για τη βελτιστοποίηση της αρχικής κατάστασης του δικτύου.

5.2.1 Μεθοδολογία: Βελτίωση της Ενεργειακής Συνάρτησης E του Δικτύου Hopfield

Η ενεργειακή συνάρτηση που φαίνεται στην εξίσωση 4.11 στο δίκτυο Hopfield δεν είναι μόνο αργή στον υπολογισμό, αλλά είναι και εύκολο να πέσει σε τοπική βέλτιστη λύση. Όλες οι σειρές "0" ή όλες οι στήλες "0" στον πίνακα μεταφοράς μπορούν να εμφανιστούν στη σταθερή κατάσταση του δικτύου. Ο κύριος λόγος είναι ο τρίτος όρος της. Η ενεργειακή συνάρτηση εγγυάται μόνο καθολικούς περιορισμούς. Δεν είναι εγγυημένο ότι ο πίνακας μεταγωγής της σταθερής εξόδου του νευρωνικού δικτύου έχει μόνο ένα "1" ανά γραμμή και στήλη.

Για να αντιμετωπιστεί αυτό το πρόβλημα, προτείνεται μια βελτιωμένη ενεργειακή συνάρτηση:

$$\begin{aligned}
 E = & \frac{A}{2} \sum_{x=1}^N \sum_{i=1}^N \sum_{j=1, j \neq i}^N V_{xi} V_{xi} + \frac{B}{2} \sum_{i=1}^N \sum_{x=1}^N \sum_{y=1, y \neq x}^N V_{xi} V_{yj} \\
 & + \frac{C}{2} \left[\sum_{x=1}^N \left(\sum_{i=1}^N V_{xi} - 1 \right)^2 + \sum_{i=1}^N \left(\sum_{x=1}^N V_{xi} - 1 \right)^2 \right] \\
 & + \frac{D}{2} \sum_{x=1}^N \sum_{y=1, y \neq x}^N \sum_{i=1}^N d_{xy} V_{xi} (V_{y, i+1} + V_{y, i-1})
 \end{aligned} \tag{5.3}$$

Ο τρίτος όρος στην εξίσωση (5.3) είναι η βελτιστοποίηση των περιορισμών γραμμής και στήλης, που εγγυάται ότι μόνο όταν ο αριθμός "1" σε κάθε γραμμή ή στήλη ενός πίνακα μετάθεσης είναι ένα, τότε μπορεί να βρεθεί η βέλτιστη λύση. Έτσι, η άκυρη λύση μπορεί να αποφευχθεί αποτελεσματικά.

5.2.2 Βελτίωση του Δικτύου Hopfield με Βάση τον Γενετικό Αλγόριθμο

Η δομή του δικτύου Hopfield προκαλεί ορισμένα προβλήματα στη λύση TSP, όπως τοπικό ελάχιστο, αστάθεια και υψηλή ευαισθησία σε παραμέτρους. Για την αντιμετώπιση αυτών των προβλημάτων, ο γενετικός αλγόριθμος εισάγεται στο δίκτυο Hopfield για τη βελτιστοποίηση της αρχικής κατάστασης του νευρωνικού δικτύου.

Η λύση του TSP μπορεί να θεωρηθεί ως αναζήτηση ενός συνόλου μονοπάτιων που ικανοποιούν περιορισμούς για τη βελτιστοποίηση της συνάρτησης ελαχιστοποίησης. Για την κωδικοποίηση του προβλήματος, υιοθετείται η κωδικοποίηση ακεραίου αριθμού: ένας διατεταγμένος πίνακας ακεραίων που υποδηλώνει τη σειρά με την οποία γίνεται πρόσβαση σε N κόμβους πόλεων.

Η λειτουργία του αλγορίθμου ακολουθεί κάποια συγκεκριμένα βήματα. Για αρχή, τη δημιουργία του αρχικού πληθυσμού για τη παραγωγή κωδικοποίησης ολικών αριθμών και στη συνέχεια την αποκωδικοποίηση κάθε χρωμοσώματος στον πίνακα μετάθεσης του δικτύου Hopfield. Ακολουθεί η εφαρμογή επιλογής, διασταύρωσης και μετάλλαξης για τη δημιουργία νέων λύσεων καθώς και η μετάφραση του βέλτιστου χρωμοσώματος στον πίνακα μετάθεσης του δικτύου Hopfield. Τελικώς, ακολουθεί η εκτέλεση του δικτύου Hopfield μέχρι τη σταθερή σύγκλιση.

5.2.3 Πειραματική Αξιολόγηση

Η αξιολόγηση της προτεινόμενης μεθόδου πραγματοποιήθηκε σε τρία πειραματικά σενάρια: με 8, 10 και 30 πόλεις.

Τα αποτελέσματά της για το πρόβλημα των 8 πόλεων έδειξαν για τη χρήση κλασικού δικτύου Hopfield το συνολικό μήκος διαδρομής είναι 3.4468 ενώ με το βελτιωμένο δίκτυο Hopfield με γενετικό αλγόριθμο το συνολικό μήκος διαδρομής είναι 3.1172.

Για το πρόβλημα των 10 πόλεων με τη χρήση κλασικού δικτύου Hopfield έχει συνολικό μήκος διαδρομής 3.2425 ενώ με το βελτιωμένο δίκτυο Hopfield με γενετικό αλγόριθμο το συνολικό μήκος της διαδρομής είναι 2.9137.

Τέλος, για το πρόβλημα των 30 πόλεων με τη χρήση κλασικού δικτύου Hopfield το αποτέλεσμα είναι "άκυρο μονοπάτι αναζήτησης"(αποτυχία εύρεσης έγκυρης λύσης), ενώ με τη χρήση βελτιωμένου δικτύου Hopfield με γενετικό αλγόριθμο το συνολικό μήκος της διαδρομής είναι 456.9790.

5.2.4 Συμπεράσματα

Τα αποτελέσματα της προσομοίωσης δείχνουν ότι το βελτιωμένο δίκτυο Hopfield με γενετικό αλγόριθμο μπορεί να πετύχει καλύτερη βελτιστοποίηση από το κλασικό δίκτυο Hopfield για την επίλυση του TSP. Τα πειράματα αποδεικνύουν ότι μεταξύ άλλων η βελτιωμένη ενεργειακή συνάρτηση αποφεύγει αποτελεσματικά την αναζήτηση "άκυρων" λύσεων. Έπειτα, με τη βελτιστοποίηση της αρχικής κατάστασης του νευρωνικού δικτύου με γενετικό αλγόριθμο μειώνεται η πιθανότητα παγίδευσης σε τοπικά ελάχιστα.

Τέλος, η προτεινόμενη μέθοδος μπορεί να αντιμετωπίσει αποτελεσματικά, προβλήματα μεγαλύτερης κλίμακας, όπου το κλασικό δίκτυο Hopfield αποτυγχάνει.

Η σύγκλιση του βελτιωμένου δικτύου δεν εξαρτάται από την τοπολογική κατανομή των πόλεων για την επίλυση του TSP, ενώ παρέχει καλύτερη συνολική αποτελεσματικότητα έναντι του κλασικού αλγορίθμου του νευρωνικού δικτύου Hopfield.

5.3 Συνεργατικός Νευροδυναμικός Αλγόριθμος Βελτιστοποίησης για το Πρόβλημα του Περιοδεύοντος Πωλητή

Οι Zhong, Feng, Tang, Xiong, Dai και Zhang (2022)[47] προτείνουν μια μέθοδο συνεργατικής νευροδυναμικής βελτιστοποίησης (Collaborative Neurodynamic Optimization - CNO) για την επίλυση του προβλήματος του περιοδεύοντος πωλητή (Traveling Salesman Problem - TSP). Το πρόβλημα του περιοδεύοντος πωλητή, ως γνωστόν, στοχεύει στην εύρεση μιας διαδρομής που επισκέπτεται κάθε πόλη ακριβώς μία φορά και επιστρέφει στο σημείο εκκίνησης, ελαχιστοποιώντας τη συνολική απόσταση. Η προτεινόμενη προσέγγιση αποτελεί μια καινοτόμο συνεργασία μεταξύ Συνεχών Νευρωνικών Δικτύων Hopfield (Continuous Hopfield Neural Network - CHNN) και του αλγορίθμου Βελτιστοποίησης Σμήνους Σωματιδίων (Particle Swarm Optimization - PSO).

5.3.1 Μεθοδολογία

Η προτεινόμενη μεθοδολογία συνεργατικού νευροδυναμικού αλγορίθμου βελτιστοποίησης για το Πρόβλημα του Περιοδεύοντος Πωλητή ακολουθεί τρία βασικά βήματα όπως αναλύονται παρακάτω.

Πρώτο βήμα είναι η αναδιατύπωση του προβλήματος. Οι συγγραφείς αναδιατυπώνουν το TSP ως ένα τετραγωνικό πρόβλημα δυαδικής[48] βελτιστοποίησης χωρίς περιορισμούς (Quadratic Unconstrained Binary Optimization - QUBO), μετατρέποντας τις συναρτήσεις ποινής σε περιορισμούς ισότητας. Συγκεκριμένα, δημιουργείται ένα Συνεχές Νευρωνικό Δίκτυο Hopfield με $n \times n$ νευρώνες για n πόλεις.

Έπειτα ακολουθεί η μοντελοποίηση με συνεχές δίκτυο Hopfield. Για να διασφαλιστεί η σύγκλιση του CHNN, το πρόβλημα TSP αναδιατυπώνεται ώστε να ικανοποιεί τις συνθήκες σύγκλισης του CHNN. Η ενεργειακή συνάρτηση του δικτύου ορίζεται ως: $E = -\frac{1}{2} u^t T_u - (i^b)^t u$ όπου T είναι ένας συμμετρικός πίνακας βαρών και i^b ένας πίνακας πόλωσης.

Τελευταίο βήμα είναι η συνεργατική βελτιστοποίηση. Προτείνεται η χρήση ενός πληθυσμού CHNN για την αναζήτηση τοπικών βέλτιστων λύσεων, ενώ η αρχικοποίηση των καταστάσεων κάθε δικτύου πραγματοποιείται με τη χρήση του αλγορίθμου PSO. Αυτή η διαδικασία επαναλαμβάνεται μέχρι να επιτευχθεί η καθολική βέλτιστη λύση ή να φτάσουμε στο μέγιστο αριθμό επαναλήψεων.

Ο αλγόριθμος CHN_PSO, όπως παρουσιάζεται στην εργασία, ενσωματώνει τις παραπάνω αρχές και περιλαμβάνει μεταξύ άλλων την αρχικοποίηση ενός πληθυσμού νευρωνικών δικτύων Hopfield, τη χρήση του CHNN για τη βελτιστοποίηση της συνάρτησης ενέργειας, την ενημέρωση των αρχικών καταστάσεων

με τον αλγόριθμο PSO καθώς και τη συνεχή επανάληψη της διαδικασίας μέχρι την επίτευξη της καθολικής βέλτιστης λύσης.

5.3.2 Πειραματικά Αποτελέσματα

Για την αξιολόγηση της προτεινόμενης μεθόδου, οι συγγραφείς πραγματοποίησαν πειράματα σε τέσσερα σημεία αναφοράς του TSP: att48, burma14, ulysse16 και ulysse22. Οι παράμετροι του αλγορίθμου προσαρμόστηκαν ανάλογα με το εκάστοτε πρόβλημα, με τις τιμές των A, D, u0, t, Iter, N, M, c1 και c2 να επιλέγονται κατάλληλα για κάθε σύνολο δεδομένων.

Τα πειραματικά αποτελέσματα συγκρίθηκαν με δύο άλλους αλγορίθμους: το Διακριτό Νευρωνικό Δίκτυο Hopfield (Discrete Hopfield Network - DHN) και το απλό Συνεχές Νευρωνικό Δίκτυο Hopfield (CHN). Τα αποτελέσματα που προέκυψαν έδειξαν ότι ο προτεινόμενος αλγόριθμος CHN_PSO παρουσίασε στατιστικά καλύτερη απόδοση από τους αλγορίθμους CHN και DHN σε όλα τα εξεταζόμενα σύνολα δεδομένων. Επιπλέον, η τελική διαδρομή βελτιστοποίησης της μεθόδου CHN_PSO ξεπέρασε σε απόδοση τους αλγορίθμους CHN και DHN στα σύνολα δεδομένων att48, burma14 και ulysse16. Επιπροσθέτως, στην περίπτωση του ulysse22, αν και οι αλγόριθμοι CHN και DHN ήταν κοντά στην απόδοση του CHN_PSO, εξακολουθούσαν να μην παρέχουν ικανοποιητικά αποτελέσματα.

Επιπλέον, τα πειραματικά αποτελέσματα έδειξαν ότι οι επαναλήψεις εξωτερικού βρόχου (που σχετίζονται με τον αλγόριθμο PSO) ήταν λιγότερες από τις επαναλήψεις εσωτερικού βρόχου (που σχετίζονται με το CHNN) για την επίτευξη σύγκλισης της συνάρτησης.

5.3.3 Συμπεράσματα

Συμπερασματικά, η προτεινόμενη συνεργατική νευροδυναμική προσέγγιση (CNO) αποδεικνύεται αποτελεσματική για την επίλυση του προβλήματος του περιοδεύοντος πωλητή. Ο συνδυασμός του Συνεχούς Νευρωνικού Δικτύου Hopfield (CHNN) με τον αλγόριθμο Βελτιστοποίησης Σμήνους Σωματιδίων (PSO) επιτρέπει την αποτελεσματική αναζήτηση τόσο τοπικών όσο και καθολικών βέλτιστων λύσεων.

Οι συγγραφείς σημειώνουν ότι μελλοντικά σχεδιάζουν να διερευνήσουν τη χρήση του Διακριτού Νευρωνικού Δικτύου Hopfield για την επίλυση του προβλήματος, καθώς και τον συνδυασμό του με άλλους αλγορίθμους νοημοσύνης σμήνους. Η έρευνα αυτή αποτελεί βάση για περαιτέρω μελέτη σχετικά με την αποτελεσματική και αποδοτική συνεργασία μεταξύ διαφορετικών προσεγγίσεων βελτιστοποίησης.

5.4 Συνεργατική Νευροδυναμική Βελτιστοποίηση με Διακριτά Δίκτυα Hopfield για την Επίλυση του Προβλήματος του Περιοδεύοντος Πωλητή

Οι Li, Wang και Wang (2021) παρουσίασαν μια καινοτόμο προσέγγιση για την επίλυση του Προβλήματος του Περιοδεύοντος Πωλητή (Travelling Salesman Problem - TSP) βασισμένη στη Συνεργατική Νευροδυναμική Βελτιστοποίηση (Collaborative Neurodynamic Optimization - CNO)[49] με τη χρήση διακριτών δικτύων Hopfield (Discrete Hopfield Networks - DHN). Το TSP αποτελεί ένα κλασικό πρόβλημα

συνδυαστικής βελτιστοποίησης NP-hard, το οποίο έχει εφαρμογές σε πολλούς τομείς, συμπεριλαμβανομένων του προβλήματος τετραγωνικής ανάθεσης, των προβλημάτων χρονοπρογραμματισμού και πολλών άλλων δυσεπίλυτων προβλημάτων συνδυαστικής βελτιστοποίησης.

5.4.1 Μεθοδολογία: Διατύπωση του Προβλήματος

Οι ερευνητές διατύπωσαν το TSP ως πρόβλημα τετραγωνικής δυαδικής βελτιστοποίησης με περιορισμούς και στη συνέχεια το αναδιατύπωσαν ως πρόβλημα τετραγωνικής δυαδικής βελτιστοποίησης χωρίς περιορισμούς (Quadratic Unconstrained Binary Optimization - QUBO) χρησιμοποιώντας μια συνάρτηση ποινής που προέκυψε από τους περιορισμούς ισότητας. Συγκεκριμένα, διατύπωσαν το TSP ως εξής:

$$\min \sum_{i=1}^n \sum_{j=1}^n \sum_{l=1, l \neq j}^n d_{jl} x_{ij} x_{l+1, i} \quad (5.4)$$

με τους περιορισμούς:

- $\sum_{i=1}^n x_{ij} = 1, j = 1, \dots, n$ (κάθε πόλη επισκέπτεται μόνο μία φορά)
- $\sum_{j=1}^n x_{ij} = 1, i = 1, \dots, n$ (ο πωλητής αναχωρεί από κάθε πόλη μόνο μία φορά)
- $x_{ij} \in \{0, 1\}, i, j = 1, \dots, n$

όπου n είναι ο αριθμός των πόλεων, d_{jl} είναι η απόσταση μεταξύ των πόλεων j και l , και η μεταβλητή απόφασης $x_{ij} = 1$ σημαίνει ότι η i -οστή στάση είναι η πόλη j .

5.4.2 Συνεργατική Νευροδυναμική Βελτιστοποίηση

Η προτεινόμενη προσέγγιση CNO βασίζεται σε έναν πληθυσμό διακριτών δικτύων Hopfield που χρησιμοποιούνται για την αναζήτηση τοπικών βέλτιστων λύσεων. Τα δίκτυα επανεκκινούνται επανειλημμένα χρησιμοποιώντας τον κανόνα βελτιστοποίησης σμήνους σωματιδίων (Particle Swarm Optimization - PSO) προς μια συνολική βέλτιστη λύση[50].

Ένα σημαντικό στοιχείο της μεθόδου είναι η ενεργοποίηση των νευρώνων του διακριτού δικτύου Hopfield σε ομάδες. Συγκεκριμένα, οι νευρώνες μπορούν να χωριστούν σε $2n+1$ ομάδες όπου οι νευρώνες στην ίδια ομάδα δεν συνδέονται μεταξύ τους. Ως αποτέλεσμα, οι νευρώνες στην ίδια ομάδα μπορούν να ενεργοποιηθούν συγχρονισμένα, γεγονός που επιταχύνει τη διαδικασία σύγκλισης.

Οι συγγραφείς ανέπτυξαν τρεις αλγόριθμους, έναν αλγόριθμο για τον καθορισμό των ομάδων νευρώνων, έναν αλγόριθμο για την υλοποίηση του DHN ο οποίος ενεργοποιείται σε ομάδες και τέλος τον αλγόριθμο TSP βασιζόμενο στην προσέγγιση CNO.

5.4.3 Πειραματικά Αποτελέσματα

Οι συγγραφείς αξιολόγησαν την προτεινόμενη προσέγγιση CNO σε τέσσερα σύνολα δεδομένων από την TSPLIB: burma14, ulysses16, ulysses22 και bays29, όπου ο αριθμός στο όνομα υποδεικνύει τον αριθμό των πόλεων.

Για την παραμετροποίηση του αλγορίθμου, διεξήγαγαν δοκιμές Monte Carlo για τον καθορισμό των τιμών των δύο υπερπαραμέτρων N (μέγεθος πληθυσμού) και M (κριτήριο τερματισμού). Για το σύνολο δεδομένων burma14, οι βέλτιστες τιμές ήταν $M=20$ και $N=200$, ενώ για το ulysses16 ήταν $M=20$ και $N=300$.

Τα αποτελέσματα έδειξαν ότι:

- Για το burma14, η μέθοδος πέτυχε βέλτιστη απόσταση 3323 (ελάχιστη) με μέση τιμή 3674 ± 190
- Για το ulysses16, επιτεύχθηκε βέλτιστη απόσταση 6859 με μέση τιμή 7365 ± 295
- Για το ulysses22, επιτεύχθηκε βέλτιστη απόσταση 7013 με μέση τιμή 7695 ± 354
- Για το bays29, επιτεύχθηκε απόσταση 2254 (ελάχιστη) με μέση τιμή 2555 ± 144 , ενώ η βέλτιστη γνωστή λύση είναι 2020

Οι ερευνητές παρουσίασαν επίσης γραφικές απεικονίσεις των ομάδων νευρώνων, της δυναμικής συμπεριφοράς ενός μεμονωμένου DHN και των σχετικών τιμών των συναρτήσεων, της συγκλίνουσας συμπεριφοράς της προσέγγισης CNO και των βέλτιστων διαδρομών που επιτεύχθηκαν.

5.4.4 Συμπεράσματα

Η προτεινόμενη συνεργατική νευροδυναμική προσέγγιση με τη χρήση ενός πληθυσμού διακριτών δικτύων Hopfield που ενεργοποιούνται σε ομάδες αποδείχθηκε αποτελεσματική για την επίλυση του TSP. Τα πειραματικά αποτελέσματα επαλήθευσαν την αποτελεσματικότητα της προσέγγισης σε τέσσερα σύνολα δεδομένων αναφοράς TSP.

Οι συγγραφείς σημειώνουν ότι για τη βελτίωση της κλιμάκωσης της συνεργατικής νευροδυναμικής προσέγγισης στο TSP, θα πρέπει να ενσωματωθούν αλγόριθμοι μάθησης χωρίς επίβλεψη, με επίβλεψη και ημιοποπτευόμενης μάθησης. Για παράδειγμα, η ομαδοποίηση, η βαθιά μάθηση και η ενισχυτική μάθηση θα μπορούσαν να χρησιμοποιηθούν για τη δημιουργία πρότερης γνώσης ως περιορισμοί TSP για τη μείωση της χρονικής και χωρικής πολυπλοκότητας της προτεινόμενης νευροδυναμικής προσέγγισης.

5.5 Επίλυση του Προβλήματος του Περιοδεύοντος Πωλητή με Επιταχυνόμενο Δίκτυο Hopfield και Πολλαπλασιαστές Lagrange

Στην εργασία των Hu και Duan (2022)[51] παρουσιάζεται μια σημαντική βελτίωση στην επίλυση του Προβλήματος του Περιοδεύοντος Πωλητή (Traveling Salesman Problem - TSP) μέσω νευρωνικών δικτύων τύπου Hopfield. Οι συγγραφείς αναγνωρίζουν ένα θεμελιώδες πρόβλημα στην κλασική προσέγγιση του δικτύου Hopfield (Hopfield Neural Network - HNN): την τάση του να παγιδεύεται σε τοπικά ελάχιστα, οδηγώντας σε μη αποδοτικές λύσεις. Για την αντιμετώπιση αυτής της αδυναμίας, προτείνουν έναν νέο αλγόριθμο, το Επιταχυνόμενο Επαυξημένο Δίκτυο Hopfield με Πολλαπλασιαστές Lagrange (Accelerated Augmented Lagrangian Hopfield Neural Network - AALHNN).

5.5.1 Μεθοδολογία

Το παραδοσιακό νευρωνικό δίκτυο Hopfield χρησιμοποιεί τη μέθοδο ποινής (penalty method) για την επίλυση του TSP, η οποία παρουσιάζει εγγενείς αδυναμίες. Αντί αυτής της προσέγγισης, οι ερευνητές εισάγουν μια νέα μεθοδολογία η οποία βασίζεται σε κάποια βασικά αλγοριθμικά εργαλεία βελτιστοποίησης. Την μέθοδο πολλαπλασιαστών Lagrange[52], η οποία απελευθερώνει τον αλγόριθμο από το δίλλημα της μεθόδου ποινής, εξασφαλίζοντας ότι η λύση είναι πάντα αποδεκτή. Έπειτα, την επαυξημένη συνάρτηση Lagrange, με την οποία προστίθεται ένας δεύτερης τάξης παράγοντας, ο οποίος σταθεροποιεί το δυναμικό μοντέλο του νευρωνικού δικτύου, βελτιώνοντας την αποτελεσματικότητα της λύσης. Τέλος, τη τεχνική επιτάχυνσης Nesterov , η οποία χρησιμοποιείται για την ενημέρωση των πολλαπλασιαστών Lagrange, επιταχύνοντας τη σύγκλιση του αλγορίθμου.

Το TSP μοντελοποιείται ως πρόβλημα βελτιστοποίησης υπό περιορισμούς, όπου η συνάρτηση στόχου αντιπροσωπεύει το συνολικό μήκος της διαδρομής και οι περιορισμοί εξασφαλίζουν ότι κάθε πόλη επισκέπτεται ακριβώς μία φορά. Αυτό μετατρέπεται σε μια συνάρτηση ενέργειας του δικτύου Hopfield που πρέπει να ελαχιστοποιηθεί.

Οι ερευνητές αποδεικνύουν θεωρητικά ότι το ακρότατο που επιτυγχάνεται με τον βελτιωμένο αλγόριθμο είναι η βέλτιστη λύση του αρχικού προβλήματος. Η ροή του αλγορίθμου περιλαμβάνει τα εξής σημαντικά στάδια. Αρχίζοντας, τη μετατροπή του αρχικού προβλήματος TSP καθώς και την αρχικοποίηση των πολλαπλασιαστών Lagrange. Στη συνέχεια, την επίλυση του προβλήματος με τη χρήση του AALHNN και τη λήψη εφικτής λύσης. Τελικά, ακολουθεί ο έλεγχος κριτηρίων τερματισμού και η ενημέρωση των πολλαπλασιαστών.

5.5.2 Αποτελέσματα

Οι ερευνητές πραγματοποίησαν εκτενείς προσομοιώσεις για την αξιολόγηση του αλγορίθμου AALHNN συγκριτικά με το παραδοσιακό HNN και με τυχαίες διαδρομές, χρησιμοποιώντας προβλήματα TSP με 8, 10, 12 και 15 πόλεις.

Τα αποτελέσματα ήταν αξιοσημείωτα:

- **Ποσοστό επιτυχίας λύσεων:** Ενώ το HNN παρουσίασε ποσοστό επιτυχίας περίπου 80% για το TSP με 10 πόλεις, το οποίο μειώθηκε στο 71% για 15 πόλεις, το AALHNN πέτυχε σταθερά 100% επιτυχία ανεξάρτητα από τον αριθμό των πόλεων.
- **Ποιότητα λύσεων:** Το AALHNN παρήγαγε συστηματικά καλύτερες λύσεις (μικρότερο μήκος διαδρομής) σε σύγκριση με το HNN. Για παράδειγμα, για 15 πόλεις, το AALHNN πέτυχε μήκος διαδρομής 3,4019, ενώ το HNN πέτυχε 5,4106.
- **Σταθερότητα απόδοσης:** Η απόδοση του AALHNN παρέμεινε σταθερή καθώς αυξανόταν ο αριθμός των πόλεων, ενώ η απόδοση του HNN μειωνόταν σημαντικά.
- **Ταχύτητα σύγκλισης:** Το AALHNN πέτυχε ταχύτερη σύγκλιση σε σύγκριση με το ALHNN (χωρίς την τεχνική επιτάχυνσης Nesterov)[53], με ρυθμό σύγκλισης $O\left(\frac{1}{k^2}\right)$ έναντι $O\left(\frac{1}{k}\right)$
- του παραδοσιακού ALHNN.

5.5.3 Συμπεράσματα

Η έρευνα των Hu και Duan αποδεικνύει ότι ο προτεινόμενος αλγόριθμος AALHNN ξεπερνά σημαντικά τους περιορισμούς του παραδοσιακού HNN για την επίλυση του TSP. Η καινοτομία της προσέγγισης έγκειται στην αντικατάσταση της μεθόδου ποινής με τη μέθοδο πολλαπλασιαστών Lagrange, προσθέτοντας παράλληλα τετραγωνικούς όρους για σταθεροποίηση και εφαρμόζοντας την τεχνική επιτάχυνσης Nesterov.

Ωστόσο, οι ερευνητές επισημαίνουν ότι η αποτελεσματικότητα του AALHNN μειώνεται καθώς αυξάνεται ο αριθμός των πόλεων, λόγω της αυξημένης πολυπλοκότητας των εξισώσεων και του υπολογιστικού φόρτου. Προτείνουν την απλοποίηση των εξισώσεων ως μελλοντική κατεύθυνση έρευνας για τη διεύρυνση του πεδίου εφαρμογής του αλγορίθμου.

Συνολικά, η εργασία αυτή παρουσιάζει μια σημαντική συμβολή στην επίλυση του TSP με νευρωνικά δίκτυα τύπου Hopfield, προσφέροντας καλύτερες λύσεις και υψηλότερο ποσοστό επιτυχίας, ενώ παράλληλα διατηρεί γρήγορη σύγκλιση.

5.6 Βελτίωση του Μοντέλου Hopfield στο Πρόβλημα του Περιοδευόντος Πωλητή: Μία Στρατηγική Διαίρει και Βασίλευε

Το πρόβλημα του περιοδευόντος πωλητή (Traveling Salesman Problem - TSP) αποτελεί ένα από τα πιο διάσημα και πολυμελετημένα προβλήματα συνδυαστικής βελτιστοποίησης. Το δίκτυο Hopfield, ιδιαίτερα στη συνεχή του μορφή (Continuous Hopfield Network - CHN), έχει χρησιμοποιηθεί εκτενώς για την επίλυση του TSP, αλλά συχνά αντιμετωπίζει προβλήματα όταν εφαρμόζεται σε περιπτώσεις μεγάλου μεγέθους.

Οι García, Talaván και Yáñez (2016)[1] στην εργασία τους "Improving the Hopfield model performance when applied to the traveling salesman problem: A divide-and-conquer scheme" προτείνουν μια καινοτόμο προσέγγιση διαίρει και βασίλευε (divide-and-conquer) για τη βελτίωση της απόδοσης του μοντέλου

Hopfield στην επίλυση του TSP. Η προτεινόμενη μέθοδος αντιμετωπίζει τα κύρια μειονεκτήματα των παραδοσιακών υλοποιήσεων του δικτύου Hopfield και επιτυγχάνει καλύτερα αποτελέσματα, ιδιαίτερα σε προβλήματα μεγάλης κλίμακας.

5.6.1 Θεωρητικό Υπόβαθρο

Το συνεχές δίκτυο Hopfield (Continuous Hopfield Network - CHN) είναι ένα πλήρως συνδεδεμένο νευρωνικό δίκτυο με N συνεχείς νευρώνες. Η δυναμική του περιγράφεται από μια διαφορική εξίσωση:

$$\frac{du}{dt} = -\frac{u}{\Lambda} + Tv + i^b \quad (5.5)$$

όπου το $v = g(u)$ είναι η συνάρτηση εξόδου, συνήθως μια υπερβολική εφαπτομένη:

$$g(u_i) = \frac{1}{2} \left(1 + \tanh \left(\frac{u_i}{u_0} \right) \right), u_0 > 0 \quad (5.6)$$

Η ύπαρξη σημείου ισορροπίας εξασφαλίζεται αν υπάρχει μια συνάρτηση Lyapunov ή ενέργειας. Ο Hopfield (1984) απέδειξε ότι αν ο πίνακας T είναι συμμετρικός, τότε η ακόλουθη συνάρτηση Lyapunov υπάρχει και μπορεί να ελαχιστοποιηθεί:

$$E(u) = -\frac{1}{2} u^t T u - (i^b)^u u + \frac{1}{\Lambda} \sum_{i=1}^N \int_0^{u_i} g^{-1}(x) dx \quad (5.7)$$

Κατά την εφαρμογή του CHN στο TSP, η ιδέα είναι να αντιστοιχιστεί το πρόβλημα βελτιστοποίησης στην ελαχιστοποίηση της συνάρτησης ενέργειας του δικτύου.

5.6.2 Η Προτεινόμενη Στρατηγική Διαίρει και Βασίλευε

Η βασική ιδέα της προσέγγισης διαίρει και βασίλευε που προτείνουν οι García et al. είναι να χωρίσουν το πρόβλημα TSP σε δύο φάσεις:

Φάση 1: Δημιουργία Αλυσίδων Γειτονικών Πόλεων (TSP_1^t)

Στην πρώτη φάση, για κάθε πόλη x λαμβάνονται υπόψη μόνο οι t πλησιέστερες γειτονικές της πόλεις, διατηρώντας τις αρχικές αποστάσεις μεταξύ τους. Οι υπόλοιπες πόλεις "απομακρύνονται" ορίζοντας τις αποστάσεις τους σε μια πολύ μεγάλη τιμή d_U (άνω όριο). Αυτό δημιουργεί το τροποποιημένο πρόβλημα TSP_1^t .

Ο πίνακας αποστάσεων τροποποιείται ως εξής:

$$d_{xy}^t = \{d_{xy} \text{ αν η } y \text{ είναι μία από τις } t\text{-γειτονικές πόλεις της } x$$

ή η x είναι μία από τις t -γειτονικές πόλεις της y

$$d_u - \left(\frac{d_L}{d_{x,y}} - \frac{d_L}{d_U} \right) \text{ διαφορετικά}$$

Το πρόβλημα TSP_1^t επιλύεται με ένα CHN, παράγοντας ως αποτέλεσμα K αλυσίδες γειτονικών πόλεων και πιθανώς κάποιες απομονωμένες πόλεις.

Φάση 2: Σύνδεση Αλυσίδων και Απομονωμένων Πόλεων (TSP_2^*)

Στη δεύτερη φάση, οι K αλυσίδες που δημιουργήθηκαν στην πρώτη φάση θεωρούνται ως οντότητες που πρέπει να συνδεθούν μεταξύ τους, μαζί με τις απομονωμένες πόλεις, για να δημιουργηθεί η τελική διαδρομή. Αυτό αποτελεί ένα νέο πρόβλημα TSP_2^* με $n = 2K + (\text{αριθμός απομονωμένων πόλεων})$ κόμβους.

Η ιδιαιτερότητα αυτής της φάσης είναι ότι κάθε αλυσίδα αναπαρίσταται από τα δύο ακραία σημεία της, με τον περιορισμό ότι αν η περιήγηση συνδέεται μέσω ενός από τα ακραία σημεία, θα εξέλθει από το άλλο. Οι απομονωμένες πόλεις θεωρούνται ως ζευγαρωμένες με τον εαυτό τους.

Αυτή η προσέγγιση επίσης επιλύεται με ένα CHN, αλλά με τροποποιημένη παραμετροποίηση για να εξασφαλιστεί η εγκυρότητα των λύσεων.

5.6.3 Παραμετροποίηση των Δικτύων Hopfield

Για την επίλυση του TSP_1^t , χρησιμοποιείται η παραμετροποίηση που προτάθηκε από τους Talaván και Yáñez (2002)[54]:

1. $D = \frac{1}{d_U}$
2. $N' = N + \frac{3}{C}$
3. $B = C + 3$
4. $A = B - \frac{d_L}{d_U}$

Για την επίλυση του TSP_2^* , προτείνεται μια νέα παραμετροποίηση:

1. $D = \frac{1}{d_U^*}$
2. $N' = n + \frac{3}{C} + K$
3. $A = 3 + C$
4. $B = A + \frac{d_L^*}{d_U^*}$

όπου d_L^* και d_U^* είναι το κάτω και άνω όριο των αποστάσεων στο τροποποιημένο πρόβλημα, και n είναι ο αριθμός των κόμβων στο TSP_2^* .

5.6.4 Πειραματικά Αποτελέσματα

Οι συγγραφείς πραγματοποίησαν εκτεταμένες προσομοιώσεις για να ελέγξουν την αποτελεσματικότητα της προτεινόμενης μεθόδου. Χρησιμοποιήθηκαν πολλές περιπτώσεις από τη βιβλιοθήκη προβλημάτων TSPLIB[16], από 24 έως 13.509 πόλεις.

Μία σημαντική παρατήρηση είναι ότι η τιμή της παραμέτρου t (ο αριθμός των γειτονικών πόλεων που λαμβάνονται υπόψη στην πρώτη φάση) έχει σημαντική επίδραση στην ποιότητα της λύσης. Οι συγγραφείς διαπίστωσαν ότι για κάθε πρόβλημα υπάρχει μια βέλτιστη τιμή t^* που δίνει τα καλύτερα αποτελέσματα.

Για $t = t^*$, η ποιότητα των λύσεων βελτιώνεται σημαντικά σε σχέση με το κλασικό CHN, αλλά με το κόστος αύξησης του χρόνου υπολογισμού. Αυτό οφείλεται στο γεγονός ότι η πολυπλοκότητα του δικτύου αυξάνεται όσον αφορά τον αριθμό των συμμετριών του προβλήματος, αναγκάζοντας το δίκτυο να πραγματοποιεί περισσότερες επαναλήψεις για να καταλήξει σε λύση.

5.6.5 Σύγκριση με την Παραδοσιακή Προσέγγιση

Σε σύγκριση με τον αλγόριθμο που προτάθηκε από τους Talaván και Yáñez (2002), η προτεινόμενη μέθοδος διαίρει και βασίλευε δίνει σημαντικά καλύτερα αποτελέσματα. Για παράδειγμα, για το πρόβλημα PR1002 (1002 πόλεις), ο αλγόριθμος διαίρει και βασίλευε πέτυχε λόγο απόδοσης (performance ratio) 2.22, ενώ η μέθοδος Talaván-Yáñez πέτυχε 4.35.

Επιπλέον, η προτεινόμενη μέθοδος κατάφερε να επιλύσει προβλήματα πολύ μεγάλου μεγέθους, όπως το USA13509 (13.509 πόλεις), με λόγο απόδοσης 9.22, σε σύγκριση με τον λόγο απόδοσης 16.59 της μεθόδου Talaván-Yáñez.

5.6.6 Υπολογιστική Απόδοση

Οι συγγραφείς σημειώνουν ότι για την επίλυση προβλημάτων μεγάλου μεγέθους απαιτείται αποτελεσματικός προγραμματισμός και χειρισμός πινάκων, καθώς το αντίστοιχο δίκτυο Hopfield θα είχε τεράστιο αριθμό νευρώνων. Για παράδειγμα, το πρόβλημα USA13509 θα απαιτούσε περισσότερους από 180 εκατομμύρια νευρώνες (13.509^2) και ο πίνακας βαρών, που είναι πυκνός, θα απαιτούσε περισσότερα από 260 petabytes μνήμης.

5.6.7 Συμπεράσματα

Η στρατηγική διαίρει και βασίλευε που προτείνεται από τους García et al. παρέχει μια σημαντική βελτίωση στην απόδοση του μοντέλου Hopfield όταν εφαρμόζεται στο TSP. Οι κυριότερες συνεισφορές της εργασίας θα αναφερθούν επιγραμματικά καλύτερα παρακάτω.

Αρχίζοντας, η ιδέα σύνδεσης των γειτονικών πόλεων σε αλυσίδες στην πρώτη φάση και στη συνέχεια η σύνδεση αυτών των αλυσίδων στη δεύτερη φάση καθώς και η νέα παραμετροποίηση για το δίκτυο Hopfield στη δεύτερη φάση που εξασφαλίζει την εγκυρότητα των λύσεων καθώς και η δυνατότητα σταδιακής βελτίωσης της ποιότητας της λύσης μέσω της τροποποίησης της παραμέτρου τ είναι μερικές από τις σημαντικότερες συνεισφορές της στρατηγικής διαίρει και βασίλευε.

Τα πειραματικά αποτελέσματα δείχνουν ότι η προτεινόμενη μέθοδος είναι ιδιαίτερα αποτελεσματική για προβλήματα μεγάλου μεγέθους, όπου η κλασική προσέγγιση Hopfield αντιμετωπίζει σημαντικά προβλήματα.

Η εργασία αυτή ανοίγει επίσης ορισμένα ανοιχτά ερωτήματα σχετικά με τη σύγκλιση του δικτύου Hopfield σε σχέση με τις ελεύθερες παραμέτρους του, C και τ , και αν υπάρχει κάποια μετρική σχετική με την αταξία ή την εντροπία της κατανομής των πόλεων που να σχετίζεται άμεσα με τον συνολικό αριθμό των πόλεων.

Κεφαλαίο 6ο Πρακτική Υλοποίηση του Δικτύου Hopfield για την Επίλυση του Προβλήματος Περιοδευόντος Πωλητή

6.1 Εισαγωγή

Το πρακτικό κομμάτι της παρούσας εργασίας έχει ως επίκεντρο την υλοποίηση και την οπτικοποίηση ενός νευρωνικού δικτύου Hopfield για την επίλυση του Προβλήματος του Περιοδευόντος Πωλητή (Traveling Salesman Problem - TSP) στο χάρτη της Ελλάδας. Αποτελεί τη μετάβαση από τη θεωρητική ανάλυση που παρουσιάστηκε στο βιβλιογραφικό τμήμα, στην πρακτική εφαρμογή και απεικόνιση της λειτουργίας του αλγορίθμου σε πραγματικά δεδομένα.

Ο στόχος της υλοποίησης είναι διττός: αφενός, η πειραματική επαλήθευση της αποτελεσματικότητας του δικτύου Hopfield με τη μέθοδο "διαίρει και βασίλευε" όπως προτείνεται από τους García, Talaván και Yáñez, και αφετέρου, η δημιουργία ενός διαδραστικού εργαλείου που επιτρέπει την οπτική παρακολούθηση της εξελικτικής διαδικασίας επίλυσης του προβλήματος.

Η υλοποίηση εστιάζει σε έναν πρακτικό και ρεαλιστικό σχεδιασμό διαδρομής μεταξύ ελληνικών πόλεων, χρησιμοποιώντας πραγματικά δεδομένα αποστάσεων που συλλέχθηκαν μέσω του Google Maps API. Αυτή η προσέγγιση επιτρέπει την αξιολόγηση του αλγορίθμου σε ένα αυθεντικό πλαίσιο, προσομοιώνοντας τις απαιτήσεις που θα αντιμετώπιζε ένας πραγματικός περιοδευών πωλητής που επισκέπτεται διάφορες πόλεις στην Ελλάδα.

Σε αντίθεση με τις περισσότερες υλοποιήσεις που παρουσιάζουν μόνο το τελικό αποτέλεσμα, το σύστημα που αναπτύχθηκε στο πλαίσιο της εργασίας αυτής προσφέρει μια λεπτομερή, βήμα προς βήμα οπτικοποίηση της προόδου του αλγορίθμου, επιτρέποντας στο χρήστη να κατανοήσει:

1. Τον τρόπο με τον οποίο το δίκτυο Hopfield σχηματίζει αρχικά τοπικές αλυσίδες πόλεων (Φάση 1)
2. Τη διαδικασία σύνδεσης αυτών των αλυσίδων και των απομονωμένων πόλεων σε μια ενιαία διαδρομή (Φάση 2)
3. Τις ενδιάμεσες καταστάσεις του δικτύου κατά τις επαναλήψεις του αλγορίθμου
4. Την επίδραση του μεγέθους του προβλήματος (αριθμός πόλεων) στην αποδοτικότητα και την ποιότητα της λύσης

Στις ενότητες που ακολουθούν περιγράφονται αναλυτικά τα στάδια ανάπτυξης της υλοποίησης, από τη συλλογή και προετοιμασία των γεωγραφικών δεδομένων, την προσαρμογή του αλγορίθμου Hopfield, έως τη δημιουργία ενός ολοκληρωμένου συστήματος οπτικοποίησης σε πραγματικό χρόνο. Επιπλέον, παρουσιάζονται παραδείγματα εκτέλεσης για διαφορετικά σύνολα πόλεων, καθώς και μια αξιολόγηση της ποιότητας των λύσεων που παράγονται από το σύστημα.

Η συγκεκριμένη υλοποίηση αποτελεί όχι μόνο ένα εργαλείο επίδειξης της λειτουργίας του δικτύου Hopfield, αλλά και μια αναλυτική προσέγγιση για την κατανόηση των πολύπλοκων διεργασιών και αλληλεπιδράσεων που συμβαίνουν κατά την επίλυση προβλημάτων βελτιστοποίησης με νευρωνικά δίκτυα.

6.2 Υλοποίηση του πρακτικού σκέλους της εργασίας, μεθοδολογία και προετοιμασία δεδομένων

Για την υλοποίηση του πρακτικού σκέλους της εργασίας, επέλεξα να βασιστώ στο μοντέλο Hopfield για την επίλυση του Προβλήματος του Περιοδεύοντος Πωλητή (TSP), όπως αναλύεται στο επιστημονικό άρθρο "Improving the Hopfield model performance when applied to the traveling salesman problem" των García, Talaván και Yáñez. Ως βάση για την υλοποίηση χρησιμοποίησα το αποθετήριο GitHub "HopfieldNetworkToolbox" (<https://github.com/mathinking/HopfieldNetworkToolbox>), το οποίο είναι αναπτυγμένο σε MATLAB και διανέμεται υπό άδεια BSD-2-Clause από τον Lucas García Rodríguez. Η συγκεκριμένη άδεια επιτρέπει την αναδιανομή και τροποποίηση του κώδικα, με την προϋπόθεση διατήρησης των πληροφοριών πνευματικών δικαιωμάτων και των όρων χρήσης. Προσαρμόζοντάς το στις ανάγκες της εργασίας μου και κάνοντας τις απαραίτητες τροποποιήσεις.

6.3 Συλλογή και προετοιμασία δεδομένων

Το πρώτο βήμα ήταν η συλλογή των απαραίτητων γεωγραφικών δεδομένων για πόλεις τις Ελλάδος. Συγκεκριμένα

Επιλογή πόλεων: Επέλεξα ένα σύνολο 70 πόλεων της Ελλάδας, καλύπτοντας όλες τις γεωγραφικές περιοχές της χώρας νησιωτική και ηπειρωτική χωρά καθώς και η διασπορά τους να είναι ομοιογενής στην επικράτεια της χώρας. Επομενο βημα ήταν η συλλογή συντεταγμένων συγκέντρωσα για κάθε πόλη, συγκέντρωσα τις γεωγραφικές συντεταγμένες (γεωγραφικό πλάτος και μήκος) και τις αποθήκευσα σε ένα αρχείο CSV.

Δημιουργία πίνακα αποστάσεων: Ανέπτυξα ένα script χρησιμοποιώντας το Google Maps API[55] (το οποίο φαίνεται στο αρχείο "google_distance_matrix.py"), για να υπολογίσω τις πραγματικές οδικές αποστάσεις μεταξύ όλων των πόλεων. Το script αυτό πραγματοποιεί κλήσεις στο API της Google και συγκεντρώνει τις οδηγικές αποστάσεις μεταξύ των πόλεων, δημιουργώντας έναν πλήρη πίνακα αποστάσεων διαστάσεων 70×70 .

Μορφοποίηση των δεδομένων: Τα δεδομένα μετατράπηκαν στο πρότυπο αρχείο .tsp του TSPLIB, το οποίο είναι ευρέως χρησιμοποιούμενο για προβλήματα TSP. Το αρχείο greece70.tsp περιλαμβάνει:

- Πληροφορίες για την ονομασία του προβλήματος ("greece70")
- Τον αριθμό των πόλεων (70)
- Τον τύπο του προβλήματος (συμμετρικό TSP)
- Τη μορφή των δεδομένων απόστασης (FULL_MATRIX)
- Τον πίνακα αποστάσεων μεταξύ των πόλεων
- Τις συντεταγμένες των πόλεων για οπτικοποίηση (DISPLAY_DATA_SECTION)

Το τελικό αρχείο greece70.tsp περιέχει όλες τις απαραίτητες πληροφορίες για την εκτέλεση των αλγορίθμων επίλυσης του TSP με βάση το νευρωνικό δίκτυο Hopfield.

Η προετοιμασία των δεδομένων αποτέλεσε κρίσιμο στάδιο για την επιτυχή εφαρμογή του αλγορίθμου, καθώς η ακρίβεια των αποστάσεων και η σωστή μορφοποίηση των δεδομένων επηρεάζουν άμεσα την ποιότητα των αποτελεσμάτων.

6.4 Προσαρμογή του κώδικα για επιλογή πόλεων και σενάριων

Μετά την προετοιμασία των δεδομένων, το επόμενο βήμα ήταν η προσαρμογή του υπάρχοντος κώδικα MATLAB για να επιτρέπει την επιλογή συγκεκριμένων πόλεων από το σύνολο των 70 πόλεων της Ελλάδας. Αυτή η τροποποίηση ήταν απαραίτητη για να ικανοποιηθεί η απαίτηση της εργασίας να μπορεί ο χρήστης να επιλέγει 5, 10, 20 ή 50 πόλεις για την επίλυση του TSP.

Τροποποίηση του Συστήματος Επιλογής Πόλεων

Ο αρχικός κώδικας MATLAB (`cb_tabTSPLIBmenu.m`) που περιλαμβανόταν στο repository `HorfieldNetworkToolbox` είχε σχεδιαστεί για να φορτώνει και να επεξεργάζεται ολόκληρα αρχεία TSP χωρίς τη δυνατότητα επιλογής συγκεκριμένων πόλεων. Δημιούργησα μια τροποποιημένη έκδοση του κώδικα που παρέχει αυτή τη δυνατότητα.

Οι κύριες τροποποιήσεις περιλαμβάνουν:

Προσθήκη διεπαφής επιλογής πόλεων: Ενσωμάτωσα ένα νέο στοιχείο ελέγχου (`selectCities`) που επιτρέπει στο χρήστη να επιλέξει συγκεκριμένες πόλεις από το σύνολο των 70.

Επανυπολογισμός του πίνακα αποστάσεων: Ο τροποποιημένος κώδικας επανυπολογίζει τον πίνακα αποστάσεων με βάση τις επιλεγμένες πόλεις, δημιουργώντας έναν υποπίνακα που περιέχει μόνο τις αποστάσεις μεταξύ των επιλεγμένων πόλεων.

Δυναμική δημιουργία αρχείου TSP: Ο κώδικας δημιουργεί ένα νέο αρχείο TSP που περιέχει μόνο τις επιλεγμένες πόλεις, διατηρώντας όλες τις απαραίτητες πληροφορίες όπως οι συντεταγμένες και οι αποστάσεις.

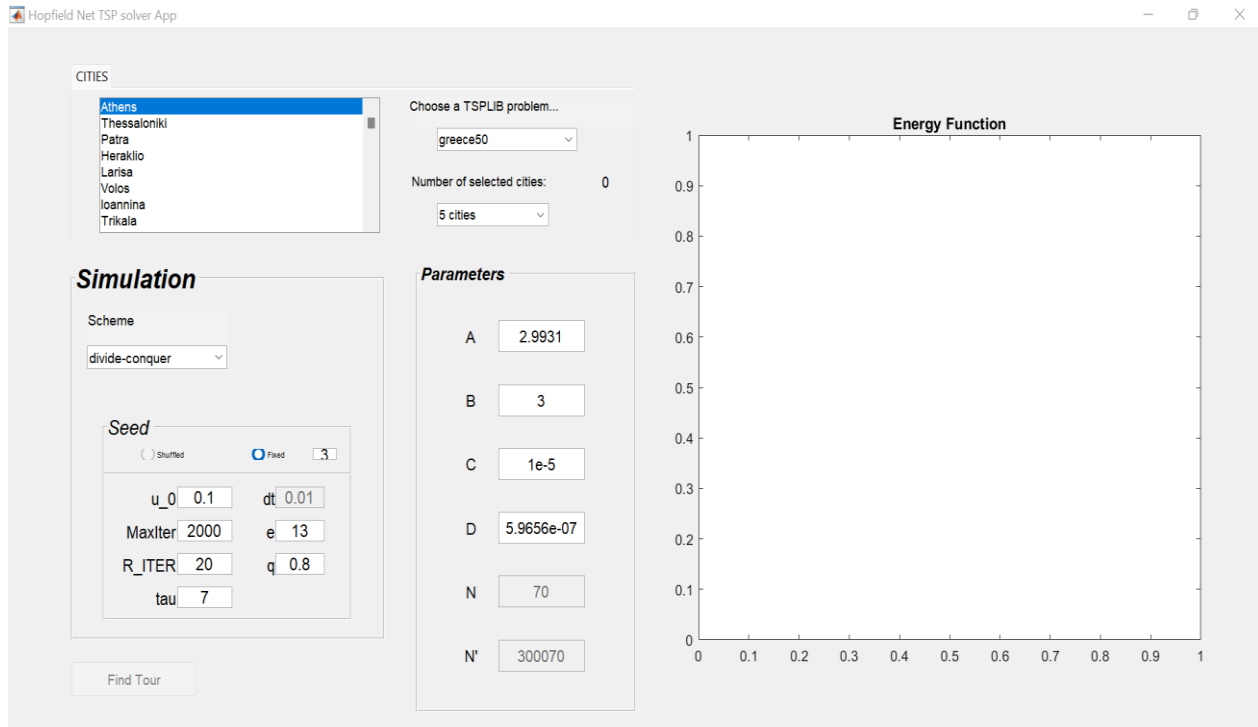
Προσθήκη προκαθορισμένων σεναρίων: Υλοποίησα επιπλέον ένα μενού επιλογής που επιτρέπει την ταχεία επιλογή προκαθορισμένων σεναρίων με 5, 10, 20, 45 ή 50 πόλεις, διευκολύνοντας τη σύγκριση της απόδοσης του αλγορίθμου σε διαφορετικά μεγέθη προβλημάτων.

6.5 Τεχνικές Λεπτομέρειες της Υλοποίησης

Ο κώδικας `cb_tabTSPLIBmenu.m` περιλαμβάνει τα ακόλουθα βασικά βήματα:

1. **Εξαγωγή των επιλεγμένων πόλεων** από το στοιχείο ελέγχου `selectCities`.
2. **Φόρτωση του αρχικού αρχείου TSP** (`greece70n.tsp`) που περιέχει τα πλήρη δεδομένα για τις 70 πόλεις.
3. **Δημιουργία νέου πίνακα αποστάσεων και συντεταγμένων** που περιλαμβάνει μόνο τις επιλεγμένες πόλεις, προσαρμόζοντας κατάλληλα τις διαστάσεις τους.
4. **Αποθήκευση του νέου αρχείου TSP** στη διαδρομή `data/TSPFiles/greece70.tsp` και δημιουργία αντιγράφου στον φάκελο `hopfield_vis/hopfield/` για την οπτικοποίηση.
5. **Φόρτωση του προβλήματος με τις επιλεγμένες πόλεις** χρησιμοποιώντας τη συνάρτηση `tsplib` και προετοιμασία του νευρωνικού δικτύου Hopfield με τη συνάρτηση `createTspHopfieldNet`.

Με αυτές τις τροποποιήσεις, η εφαρμογή μπορεί πλέον να χειριστεί διαφορετικά μεγέθη προβλημάτων και επιτρέπει στο χρήστη να επιλέξει συγκεκριμένες πόλεις της Ελλάδας για την επίλυση του TSP, ικανοποιώντας έτσι τις απαιτήσεις της εργασίας για ευελιξία στην επιλογή των δεδομένων.



Εικόνα 6.1:Γραφικό περιβάλλον χρήστη στο matlab.

6.6 Βελτίωση του αλγορίθμου προσομοίωσης για οπτικοποίηση της διαδικασίας

Ένα σημαντικό μέρος της εργασίας ήταν η οπτικοποίηση της διαδικασίας επίλυσης του προβλήματος TSP, δείχνοντας όχι μόνο την τελική διαδρομή αλλά και τα ενδιάμεσα βήματα. Ο αρχικός αλγόριθμος από το HopfieldNetworkToolbox εμφάνιζε τα αποτελέσματα μόνο μετά την ολοκλήρωση της προσομοίωσης, χωρίς να παρέχει λεπτομέρειες για τις ενδιάμεσες καταστάσεις του δικτύου. Τροποποίησα τον αλγόριθμο για να παρέχει μια λεπτομερή αναπαράσταση της εξέλιξης του δικτύου Hopfield σε κάθε φάση καθώς και σε κάθε επανάληψη της διαδικασίας επίλυσης.

6.7 Σύγκριση αρχικού και τροποποιημένου αλγορίθμου

Ο τροποποιημένος αλγόριθμος διαφέρει από τον αρχικό στα εξής καίρια σημεία:

Εμφάνιση δεδομένων σε κάθε επανάληψη: Στον αρχικό αλγόριθμο, η συνάρτηση ComputeTour() και η εμφάνιση των αλυσίδων (chains) γινόταν μόνο στο τέλος της κάθε φάσης, ενώ στην τροποποιημένη έκδοση εκτελούνται μέσα στο βρόχο επανάληψης, επιτρέποντας την προβολή των αποτελεσμάτων σε κάθε βήμα.

Συνεχής ενημέρωση της οπτικοποίησης: Πρόσθεσα εντολές εμφάνισης εντός της επανάληψης για να αναδείξω πώς εξελίσσονται οι αλυσίδες των πόλεων σε κάθε επανάληψη (Iter), βοηθώντας στην κατανόηση της σταδιακής βελτιστοποίησης της διαδρομής.

Συνέχεια του μετρητή επαναλήψεων: Στον αρχικό αλγόριθμο, ο μετρητής επαναλήψεων (Iter) επαναρχικοποιείτο για τη Φάση 2, ενώ στην τροποποιημένη έκδοση, ο μετρητής συνεχίζει από τη Φάση 1, παρέχοντας μια συνεχή αναπαράσταση της προόδου του αλγορίθμου.

Προσθήκη παραμέτρου επανάληψης στη συνάρτηση ComputeTour(): Τροποποίησα τη συνάρτηση ComputeTour() ώστε να δέχεται ως παράμετρο τον αριθμό της τρέχουσας επανάληψης, επιτρέποντας τη συσχέτιση των αποτελεσμάτων με συγκεκριμένες φάσεις της προσομοίωσης.

Βελτιστοποίηση της ροής πληροφοριών: Αναδιοργάνωσα τη δομή του αλγορίθμου ώστε να εμφανίζει συγκεκριμένες πληροφορίες σε κάθε βήμα, όπως τις τρέχουσες αλυσίδες, την ενέργεια του συστήματος και τη διαδρομή που έχει σχηματιστεί μέχρι εκείνο το σημείο.

6.8 Ανάπτυξη Συστήματος Οπτικοποίησης σε Πραγματικό Χρόνο

Το τελευταίο και ιδιαίτερα σημαντικό κομμάτι της υλοποίησης ήταν η δημιουργία ενός συστήματος οπτικοποίησης που επιτρέπει την παρακολούθηση της διαδικασίας επίλυσης του TSP σε πραγματικό χρόνο. Το σύστημα αυτό απεικονίζει γραφικά τη διαδικασία στο χάρτη της Ελλάδας, επιτρέποντας στο χρήστη να κατανοήσει πλήρως τη λειτουργία του αλγορίθμου διαίρει και βασίλευε με τη χρήση του δικτύου Hopfield.

6.9 Αρχιτεκτονική Συστήματος Οπτικοποίησης

Η αρχιτεκτονική του συστήματος βασίζεται σε δύο κύρια συστατικά:

1. **Καταγραφή Δεδομένων Προσομοίωσης (MATLAB):** Κατά τη διάρκεια της εκτέλεσης του αλγορίθμου Hopfield, τα ενδιάμεσα και τελικά αποτελέσματα καταγράφονται σε ένα αρχείο (output.txt), το οποίο περιλαμβάνει:
 - Τις επιλεγμένες πόλεις
 - Τις αλυσίδες πόλεων σε κάθε επανάληψη της Φάσης 1
 - Τις διαδρομές που δημιουργούνται στη Φάση 2
 - Την τελική βέλτιστη διαδρομή
2. **Ανάγνωση και Οπτικοποίηση Δεδομένων:** Ένα ξεχωριστό script οπτικοποίησης (main.py) διαβάζει το αρχείο εξόδου σε πραγματικό χρόνο και απεικονίζει τα αποτελέσματα στο χάρτη της Ελλάδας χρησιμοποιώντας τη βιβλιοθήκη Matplotlib και Basemap[57].

6.10 Κύρια Χαρακτηριστικά του Συστήματος Οπτικοποίησης

Το σύστημα οπτικοποίησης παρέχει τα ακόλουθα χαρακτηριστικά:

1. **Διαδραστικός Χάρτης της Ελλάδας:** Χρησιμοποιεί τη βιβλιοθήκη Basemap για την απεικόνιση ενός λεπτομερούς χάρτη της Ελλάδας με ακριβείς γεωγραφικές συντεταγμένες για κάθε πόλη.
2. **Χρωματική Κωδικοποίηση:**
 - Διαφορετικά χρώματα για κάθε αλυσίδα πόλεων στη Φάση 1
 - Διακριτά χρώματα για τα νέα τμήματα που προστίθενται στη Φάση 2
 - Ειδική χρωματική σήμανση για απομονωμένες πόλεις και την αρχική πόλη της διαδρομής
3. **Προοδευτική Απεικόνιση:** Δείχνει σταδιακά πώς εξελίσσεται η λύση σε κάθε επανάληψη του αλγορίθμου, συμπεριλαμβανομένων:
 - Της δημιουργίας και εξέλιξης των αλυσίδων πόλεων στη Φάση 1
 - Της σύνδεσης των αλυσίδων και των απομονωμένων πόλεων στη Φάση 2
 - Της τελικής βέλτιστης διαδρομής
4. **Πληροφορίες Διαδρομής:** Εμφανίζει:
 - Τα ονόματα των επιλεγμένων πόλεων
 - Την τρέχουσα φάση και αριθμό επανάληψης
 - Μια λίστα με τις απομονωμένες πόλεις (αν υπάρχουν)
 - Το συνολικό μήκος της τελικής διαδρομής σε χιλιόμετρα
5. **Διαδραστικά Στοιχεία Ελέγχου:**
 - Slider για μετακίνηση μεταξύ διαφορετικών επαναλήψεων
 - Κουμπιά για έλεγχο της προσομοίωσης (αναπαραγωγή, παύση, επαναφορά)
 - Δυνατότητα προσαρμογής της ταχύτητας αναπαράστασης (αργή, κανονική, γρήγορη)

6.11 Τεχνικές Λεπτομέρειες Υλοποίησης

Η υλοποίηση του συστήματος οπτικοποίησης περιλαμβάνει τα ακόλουθα βασικά τεχνικά στοιχεία:

1. **Ανάγνωση και Επεξεργασία Δεδομένων:**
 - ο Ανάλυση του αρχείου output.txt με χρήση κανονικών εκφράσεων (regular expressions)
 - ο Μετατροπή των δεικτών πόλεων στις πραγματικές συντεταγμένες και ονόματα πόλεων
2. **Οπτικοποίηση σε Πραγματικό Χρόνο:**
 - ο Παρακολούθηση του αρχείου εξόδου για αλλαγές με συγκεκριμένο χρονικό διάστημα ελέγχου
 - ο Δυναμική ενημέρωση του χάρτη σε κάθε νέα επανάληψη του αλγορίθμου
3. **Υπολογισμός Μήκους Διαδρομής:**
 - ο Ανάκτηση του πίνακα αποστάσεων από το αρχείο .tsp
 - ο Άθροιση των αποστάσεων μεταξύ διαδοχικών πόλεων στην τελική διαδρομή
4. **Χειρισμός Ειδικών Περιπτώσεων:**
 - ο Οπτικοποίηση πόλεων που είναι ταυτόχρονα αρχικές και απομονωμένες
 - ο Διατήρηση των χρωμάτων τμημάτων της Φάσης 1 στην τελική διαδρομή

6.12 Παράδειγμα Εκτέλεσης

Το παρακάτω παράδειγμα αναπαριστά την εξέλιξη της λύσης για ένα σενάριο με 10 επιλεγμένες πόλεις (με ID: 6, 7, 8, 9, 10, 11, 12, 13, 14, 15):

1. **Φάση 1:**
 - ο Αρχικά, ο αλγόριθμος δημιουργεί μικρές αλυσίδες πόλεων (4-3, 7-6)
 - ο Σταδιακά, οι αλυσίδες αναπτύσσονται και συγχωνεύονται
 - ο Στην τελική επανάληψη της Φάσης 1, έχουμε δύο κύριες αλυσίδες (2-1-3-9) και (6-7-5)
2. **Φάση 2:**
 - ο Ο αλγόριθμος συνδέει τις αλυσίδες και τις απομονωμένες πόλεις
 - ο Δοκιμάζει διάφορες διατάξεις: (6-5-2-9), (6-2-5-4), κ.λπ.
 - ο Σταδιακά προσθέτει και τις υπόλοιπες πόλεις: 10, 8
3. **Τελική Διαδρομή:**
 - ο Η βέλτιστη διαδρομή που βρίσκεται είναι: 5-7-6-4-2-1-3-9-10-8-5
 - ο Το σύστημα υπολογίζει και εμφανίζει το συνολικό μήκος της διαδρομής

Μέσω αυτής της οπτικοποίησης, ο χρήστης μπορεί να κατανοήσει πλήρως πώς λειτουργεί ο αλγόριθμος διαίρει και βασίλευε με το δίκτυο Hopfield, παρακολουθώντας την προοδευτική δημιουργία των αλυσίδων πόλεων και την τελική σύνδεσή τους σε μια βέλτιστη διαδρομή.

6.13 Σύνοψη του Συστήματος Οπτικοποίησης

Το αναπτυχθέν σύστημα οπτικοποίησης προσφέρει ένα ισχυρό εργαλείο για την κατανόηση και αξιολόγηση της λειτουργίας του δικτύου Hopfield που υλοποιήθηκε σε MATLAB στην επίλυση του προβλήματος του Περιοδούντος Πωλητή. Επιτρέπει την παρακολούθηση της σταδιακής εξέλιξης της λύσης, παρέχοντας πολύτιμες πληροφορίες για τον τρόπο με τον οποίο το δίκτυο Hopfield αναπτύσσει τοπικές αλυσίδες και τις συνδυάζει για να σχηματίσει τη βέλτιστη συνολική διαδρομή. Αυτό το χαρακτηριστικό επιτρέπει τη σύγκριση της απόδοσης του αλγορίθμου με διαφορετικά μεγέθη προβλημάτων, παρέχοντας πολύτιμα συμπεράσματα για την κλιμάκωση και αποτελεσματικότητα της μεθόδου.

Κεφάλαιο 7: Πειραματικά Αποτελέσματα και Συγκριτική Αξιολόγηση

7.1 Εισαγωγή

Στο παρόν κεφάλαιο παρουσιάζονται τα αποτελέσματα από την εφαρμογή του Hopfield Network με τον αλγόριθμο Divide and Conquer στο πρόβλημα του Περιοδευόντος Πωλητή. Η αξιολόγηση έγινε με βάση πραγματικά δεδομένα από 70 ελληνικές πόλεις, για τέσσερα διαφορετικά σενάρια μεγεθών ($N=5, 10, 20, 50$). Ως benchmark χρησιμοποιήθηκε η μέθοδος Simplex.

7.2 Πειραματική Διαδικασία: Παράμετροι Υλοποίησης

Για όλες τις προσομοιώσεις χρησιμοποιήθηκαν οι ακόλουθες βασικές παράμετροι του Hopfield Network:

Πίνακας 7.1: Παράμετροι Αλγορίθμου

Παράμετρος	Τιμή	Περιγραφή	Τεχνική Σημασία
U0	0.1	Κλίση συνάρτησης μεταφοράς	Προσδιορίζει το πόσο απότομη θα είναι η συνάρτηση tanh. Μικρότερη τιμή=ομαλότερη μετάβαση
E	20	Εκθέτης κριτηρίου τερματισμού	Ανοχή σύγκλισης: $C_1 = 10^{-20}$, $C_2 = 10^{-30}$
MaxIter	2000	Μέγιστος αριθμός επαναλήψεων	Αποτρέπει την άπειρη εκτέλεση
R_Iter	20	Επαναλήψεις μείωσης βήματος	Πρώτες 20 επαναλήψεις με μειωμένο βήμα
Q	0.8	Συντελεστής μείωσης βήματος	Πολλαπλασιαστής: $dt_new = 0.8 \times dt_old$

Πίνακας 7.2: Παράμετροι Δικτύου (Αυτόματα Υπολογιζόμενες)

Συντελεστής	Περιγραφή	Υπολογισμός
A	Συντελεστής Row Constraints	Phase 1: $A = B - \rho$, Phase 2: $A = 3 + C$
B	Συντελεστής Column Constraints	Phase 1: $B = 3 + C$, Phase 2: $B = A + \rho$
C	Ελεύθερη παράμετρος	1×10^{-5} (σταθερή για όλα τα σενάρια)
D	Συντελεστής κανονικοποίησης	$D = 1/dU$

7.3 Scheme: Divide and Conquer

Το Divide and Conquer scheme εφαρμόστηκε με τις παρακάτω παραμέτρους όπως παρουσιάζεται στον πίνακα 7.3:

Πίνακας 7.3: Παράμετροι Divide and Conquer.

N	Tau (τ)	Seed	Φάση 1	Φάση 2
5	2	3	τ -neighbour	Σύνδεση αλυσίδων
10	3	61	τ -neighbour	Σύνδεση αλυσίδων
20	2	6	τ -neighbour	Σύνδεση αλυσίδων
50	5	2979	τ -neighbour	Σύνδεση αλυσίδων

7.4 Η Σημασία του Seed στα Hopfield Networks

Το **seed** είναι καθοριστικός παράγοντας για την επαναληψιμότητα και την ποιότητα των αποτελεσμάτων:

Ανάλυση της Επίδρασης του Seed

1. Επαναληψιμότητα: Σταθερό seed εξασφαλίζει ίδια αποτελέσματα σε επαναλαμβανόμενες εκτελέσεις
2. Ποιότητα λύσης: Διαφορετικά seeds οδηγούν σε διαφορετικές τοπικές λύσεις
3. Αρχική κατάσταση: Επηρεάζει την αρχική κατάσταση V και U του δικτύου
4. Στρατηγική επιλογής:
 - N=5 ,20: Μικρά seeds (3,4,6) για σταθερότητα
 - N = 10 Μέτριο seeds (61)
 - N=50: Μεγάλο seed (2979) μετά από εκτεταμένο tuning

7.5 Αποτελέσματα

Πίνακας 7.4: Συγκεντρωτικός Πίνακας

N	Divide& Conquer (χλμ)	Simplex (χλμ)	Απόλυτη Διαφορά (χλμ)	Ποσοστιαία Διαφορά(%)
5	1.886,16	1.879,25	6,91	0,37

10	3.075,34	3.028,517	46,823	1,52
20	3.462,135	3.406,163	55,972	1,62
50	6.645,17	4.705,35	1.939,82	41,23

7.6 Λεπτομερή Αποτελέσματα ανά Σενάριο

7.6.1 Σενάριο N=5

Για N=5 οι επιλεγμένες πόλεις είναι Αθήνα, Θεσσαλονίκη, Πάτρα, Ηράκλειο και Ιωάννινα.

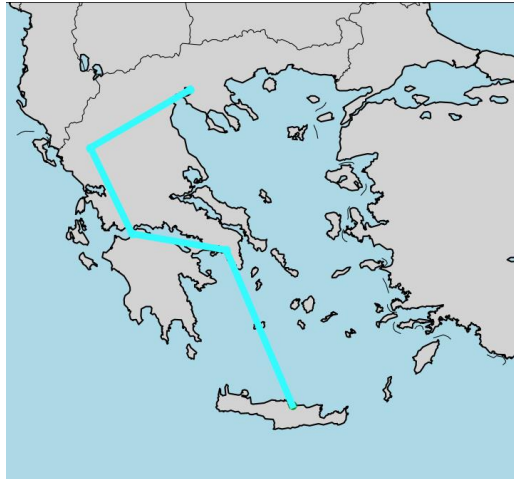
Παράμετροι Δικτύου:

- $A = 2.96, B = 3, C = 1e-5, D = 0.00036$
- $\text{Tau} = 2, \text{Seed} = 3$

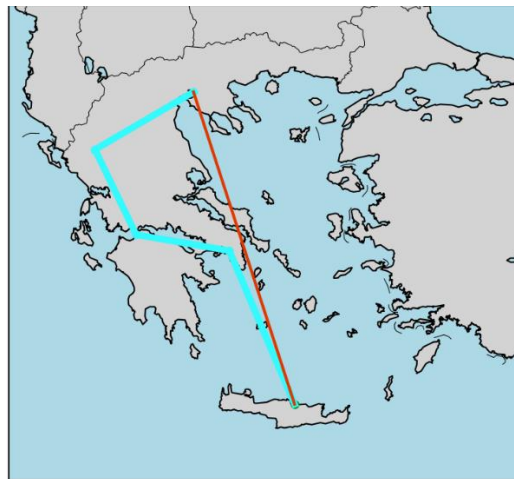
Αποτελέσματα:

- Φάση 1 (Εικόνα 7.1): Η πρώτη φάση του αλγορίθμου δημιούργησε αρχικές υπο-αλυσίδες με την τεχνική τ -neighbour για $\tau=2$. Έχει δημιουργηθεί μια και μοναδική αλυσίδα για την φάση 1 καθώς λόγω των λιγοστών πόλεων υπάρχει μικρή πολυπλοκότητα. Παρατηρούμε συνδέσεις μεταξύ Κρήτης και ηπειρωτικής Ελλάδας, καθώς και μεταξύ Αθήνας και βόρειων πόλεων.
- Φάση 2 (Εικόνα 7.2): Στη δεύτερη φάση, η υποαλυσίδα συνδέθηκε σε έναν πλήρη κύκλο Hamilton[58]. Η κόκκινη γραμμή δείχνει την ένωση των αλυσίδων που προτείνει ο αλγόριθμος στην συγκεκριμένη προσομοίωση υπάρχει μόνο μια, με μια λογική πορεία από την Κρήτη προς την Αθήνα, τη Δυτική Ελλάδα και τη Βόρεια Ελλάδα.
- Τελική Διαδρομή (Εικόνα 7.3): Ηράκλειο \rightarrow Αθήνα \rightarrow Πάτρα \rightarrow Ιωάννινα \rightarrow Θεσσαλονίκη \rightarrow Ηράκλειο με συνολικό μήκος 1.886,16 χλμ.
- Σύγκριση με Simplex (Εικόνα 7.4): Η μέθοδος Simplex έδωσε διαδρομή με μήκος 1.879,25 χλμ. Η διαφορά των 6,91 χλμ αντιστοιχεί σε απόκλιση μόλις 0,37%. Παρατηρούμε πολύ μικρές διαφορές μεταξύ των δύο λύσεων, που εξηγεί την ελάχιστη απόκλιση.
- Επιτυχία: Εξαιρετική απόδοση με διαφορά μόλις 0.37%
- Ανάλυση: Για μικρά προβλήματα, το Hopfield Network παρουσιάζει σχεδόν ιδανική απόδοση. Η διαδρομή είναι γεωγραφικά λογική, η επιλογή $\tau=2$ είναι επαρκής, και το μικρό seed (3) παρέχει σταθερά αποτελέσματα.

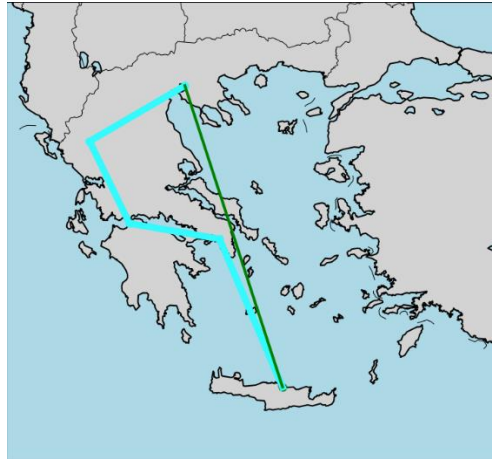
Οπτική απεικόνιση σενάριου N=5:



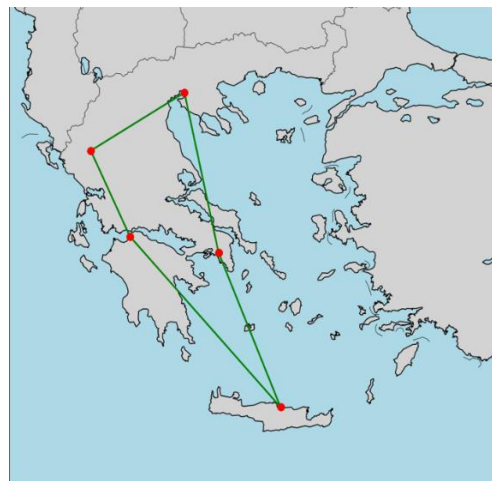
7.1: Τελευταία επανάληψη της φάσης 1 για 5 πόλεις.



7.2: Τελευταία επανάληψη της φάσης 2 για 5 πόλεις.



7.3: Τελική διαδρομή για 5 πόλεις.



7.4: Τελική διαδρομή με χρήση μεθόδου Simplex για 5 πόλεις.

Τελική σειρά επίσκεψης

Ηράκλειο -> Αθήνα -> Πάτρα -> Ιωάννινα -> Θεσσαλονίκη -> Ηράκλειο.

7.6.2 Σενάριο N=10

Για N=10 οι επιλεγμένες πόλεις είναι Αθήνα, Θεσσαλονίκη, Ηράκλειο, Ιωάννινα , Αλεξανδρούπολη , Κομοτηνή , Ρόδος, Κοζάνη, Μυτιλήνη και Κέρκυρα.

Παράμετροι Δικτύου:

- $A = 2.96, B = 3, C = 1e-5, D = 0.00036$
- $\text{Tau} = 3, \text{Seed} = 61$

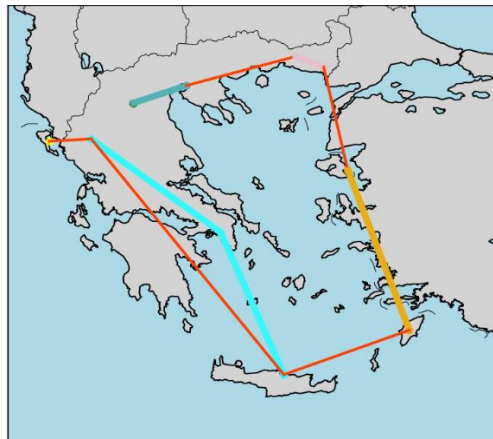
Αποτελέσματα:

- Φάση 1 (Εικόνα 7.5): Στην πρώτη φάση, με $\tau=3$, βλέπουμε συνολικά τέσσερις αλυσίδες οι οποίες αποτελούνται από τρεις αλυσίδες με δύο πόλεις και μια με τρεις. Παρατηρούμε μια κίτρινη κουκκίδα(απομονωμένη πόλη) και πιο πυκνή διασύνδεση λόγω του αυξημένου τ .
- Φάση 2 (Εικόνα 7.6): Η δεύτερη φάση δείχνει πώς τα άκρα των υποαλυσίδων ενωθήκαν μεταξύ τους καθώς και με την απομονωμένη πόλη, καλύπτοντας πλέον μεγαλύτερο τμήμα της ελληνικής επικράτειας.
- Τελική Διαδρομή (Εικόνα 7.7): Η τελική διαδρομή με πράσινο χρώμα έχει συνολικό μήκος 3.075,34 χλμ. Ακολουθεί μια λογική γεωγραφική σειρά, περιλαμβάνοντας γειτονικές πόλεις σε διαδοχικές στάσεις.
- Σύγκριση με Simplex (Εικόνα 7.8): Η μέθοδος Simplex έδωσε λύση με μήκος 3.028,517 χλμ. Η διαφορά των 46,823 χλμ αντιστοιχεί σε απόκλιση 1,52%. Παρατηρώντας προσεκτικά, μπορούμε να διακρίνουμε τις μικρές διαφορές που οδηγούν στην απόκλιση.
- Επιτυχία: Πολύ καλή απόδοση με διαφορά 1.52%
- Ανάλυση: Η αύξηση του τ (από 2 σε 3) βοήθησε στη διατήρηση καλής απόδοσης παρά την αύξηση του προβλήματος. Η ελαφρά αύξηση της απόκλισης (από 0,37% σε 1,52%) είναι αναμενόμενη λόγω αυξημένης πολυπλοκότητας. Η στρατηγική Divide and Conquer λειτουργεί αποτελεσματικά, δημιουργώντας καλές υπο-αλυσίδες που οδηγούν σε σχεδόν βέλτιστη τελική λύση.

Οπτική απεικόνιση σενάριου N=10:



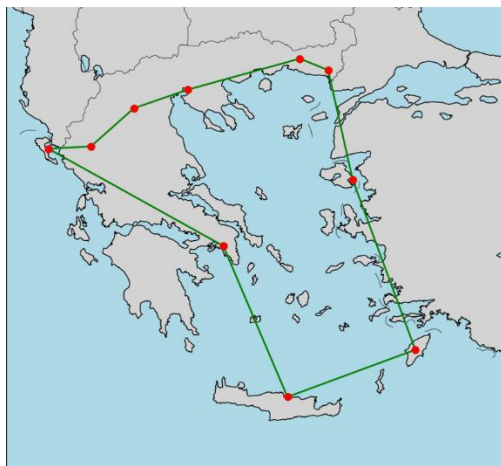
7.5: Τελευταία επανάληψη της φάσης 1 για 10 πόλεις.



7.6: Τελευταία επανάληψη της φάσης 2 για 10 πόλεις.



7.7: Τελική διαδρομή για 10 πόλεις.



7.8: Τελική διαδρομή με χρήση μεθόδου Simplex για 10 πόλεις.

Τελική σειρά επίσκεψης

Κοζάνη -> Θεσσαλονίκη -> Κομοτηνή -> Αλεξανδρούπολη -> Μυτιλήνη -> Αθήνα -> Ρόδος-> Ηράκλειο
-> Αθηνά -> Ιωάννινα -> Κέρκυρα -> Κοζάνη .

7.6.3 Σενάριο $N=20$

Για $N=20$ οι επιλεγμένες πόλεις είναι Αθήνα, Θεσσαλονίκη, Πάτρα, Ηράκλειο, Ιωάννινα, Χαλκίδα, Σέρρες, Αλεξανδρούπολη, Ξάνθη, Αγρίνιο, Καβάλα, Χανιά, Κομοτηνή, Ρόδος, Βέροια, Κοζάνη, Μυτιλήνη, Κέρκυρα, Άρτα και Πρέβεζα.

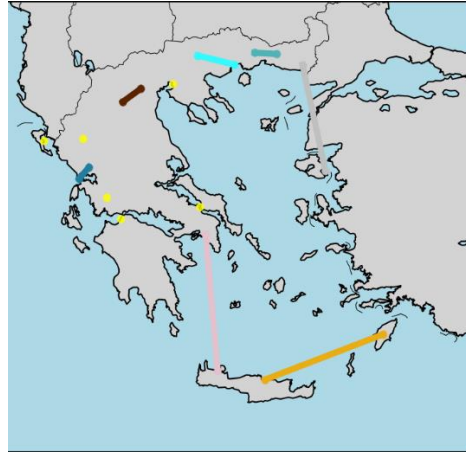
Παράμετροι Δικτύου:

- $A = 2.96, B = 3, C = 1e-5, D = 0.00036$
- $\text{Tau} = 2, \text{Seed} = 6$

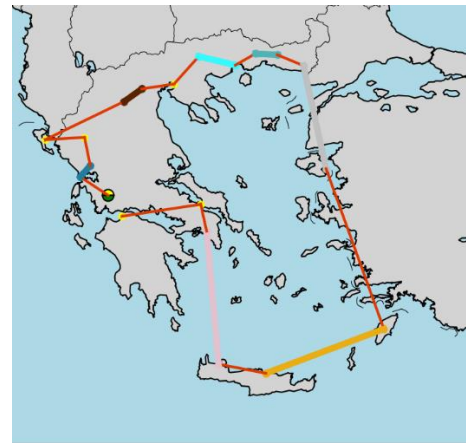
Αποτελέσματα:

- **Φάση 1 (Εικόνα 7.9):** Στην πρώτη φάση, παρά την αύξηση του αριθμού των πόλεων, επιλέχθηκε $\tau=2$. Παρατηρείται καλή συνδεσμολογία δημιουργώντας 7 αλυσίδες, με κάθε πόλη να συνδέεται μόνο με τις 2 γειτονικές της.
- **Φάση 2 (Εικόνα 7.10):** Η δεύτερη φάση παρουσιάζει τη διαδικασία σύνδεσης των υπο-αλυσίδων για τη δημιουργία του τελικού κύκλου Hamilton. Οι συνδέσεις είναι περισσότερες και πιο πολύπλοκες λόγω του αυξημένου αριθμού πόλεων.
- **Τελική Διαδρομή (Εικόνα 7.11):** Η τελική διαδρομή έχει μήκος 3.462,135 χλμ. Διατηρείται μια λογική γεωγραφική συνέχεια, αν και με περισσότερες διακλαδώσεις και τμήματα που διασχίζουν μεγαλύτερες αποστάσεις.
- **Σύγκριση με Simplex (Εικόνα 7.12):** Η μέθοδος Simplex έδωσε λύση με μήκος 3.406,163 χλμ. Η διαφορά των 55,972 χλμ αντιστοιχεί σε απόκλιση 1,62%. Η σύγκριση των εικόνων βοηθά στην κατανόηση των αιτιών για την απόκλιση.
- **Επιτυχία:** Στατιστικά σταθερή απόδοση (1.62%)
- **Ανάλυση:** Παρά τη μείωση του τ από 3 (στο $N=10$) σε 2, η απόδοση παραμένει σχεδόν αμετάβλητη (1,55% vs 1,62%). Το δίκτυο Hopfield διατηρεί την ικανότητά του να βρίσκει λύσεις πολύ κοντά στη βέλτιστη, ακόμα και για 20 πόλεις. Η επιλογή μικρότερου τ είναι ενδιαφέρουσα και δείχνει ότι η αύξησή του δεν είναι πάντα απαραίτητη καθώς αυξάνεται το N .

Οπτική απεικόνιση σενάριου N=20:



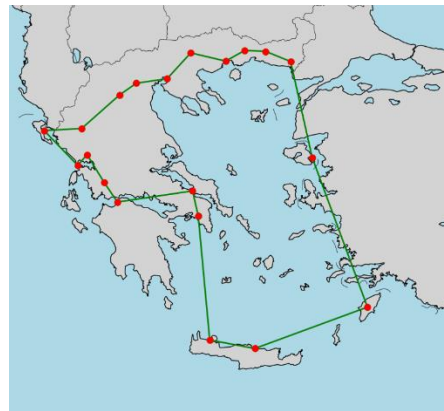
7.9: Τελευταία επανάληψη της φάσης 1 για 20 πόλεις.



7.10: Τελευταία επανάληψη της φάσης 2 για 20 πόλεις.



7.11: Τελική διαδρομή για 20 πόλεις.



7.12: Τελική διαδρομή με χρήση μεθόδου Simplex για 20 πόλεις.

Τελική σειρά επίσκεψης

Αγρίνιο -> Πρέβεζα -> Άρτα -> Ιωάννινα -> Κέρκυρα -> Κοζάνη -> Βέροια -> Θεσσαλονίκη -> Σέρρες -> Καβάλα -> Ξάνθη -> Κομοτηνή -> Αλεξανδρούπολη -> Μυτιλήνη -> Ρόδος -> Ηράκλειο -> Χανιά -> Αθήνα -> Χαλκίδα -> Πάτρα -> Αγρίνιο.

7.6.4 Σενάριο N=50

Για N=50 οι επιλεγμένες πόλεις είναι Αθήνα, Θεσσαλονίκη, Πάτρα, Ηράκλειο, Ιωάννινα, Χαλκίδα, Σέρρες, Αλεξανδρούπολη, Ξάνθη, Αγρίνιο, Καλαμάτα, Καβάλα, Χανιά, Κομοτηνή, Ρόδος, Δράμα, Βέροια, Κοζάνη, Ρέθυμνο, Τρίπολη, Κόρινθος, Γιαννιτσά, Μυτιλήνη, Πύργος, Κιλκίς, Θήβα, Άργος, Άρτα, Κως, Πρέβεζα., Σπάρτη, Νάουσα, Έδεσσα, Γρεβενά, Μέτσοβο, Ηγουμενίτσα, Πάργα, Αμφιλοχία, Μεσολόγγι, Ναύπακτος, Αμαλιάδα, Ναύπλιο, Κέα, Μήλος, Άγιος Νικόλαος, Κάρπαθος, Σάμος, Λήμνος, Σαμοθράκη και Πάτμος.

Παράμετροι Δικτύου:

- $A = 2.96, B = 3, C = 1e-5, D = 0.00036$
- $\text{Tau} = 5, \text{Seed} = 2979$

Αποτελέσματα:

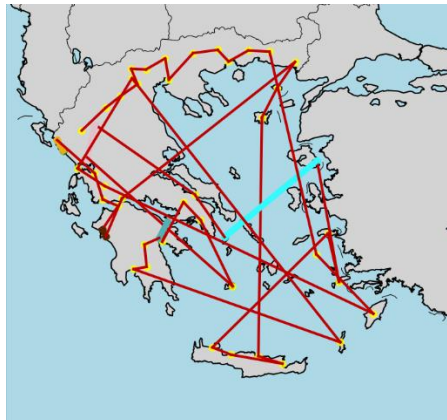
- Φάση 1 (Εικόνα 7.13): Στην πρώτη φάση για το μεγαλύτερο πρόβλημα των 50 πόλεων, επιλέχθηκε $\tau=5$. Παρατηρείται πολύ μικρο δίκτυο αλυσιων συνδέσεων μεταξύ των πόλεων σε όλη την ελληνική επικράτεια.
- Φάση 2 (Εικόνα 7.14): Η δεύτερη φάση δείχνει τις κόκκινες γραμμές που απεικονίζουν τις επιλεγμένες συνδέσεις. Η πολυπλοκότητα είναι εξαιρετικά αυξημένη, με πολλές διασταυρώσεις και μη βέλτιστες συνδέσεις.
- Τελική Διαδρομή (Εικόνα 7.15): Η τελική διαδρομή έχει μήκος 6.645,17 χλμ. Είναι εμφανείς οι ανεπάρκειες της λύσης, με πολλές διασταυρώσεις και συνδέσεις μεταξύ απομακρυσμένων πόλεων.
- Σύγκριση με Simplex (Εικόνες 7.16): Η μέθοδος Simplex έδωσε λύση με μήκος 4.705,35 χλμ. Η διαφορά των 1.939,82 χλμ αντιστοιχεί σε απόκλιση 41,23%. Η σύγκριση των εικόνων καταδεικνύει τη μεγάλη διαφορά στην ποιότητα των λύσεων.
- Πρόκληση: Σημαντική μείωση απόδοσης (41.23%)
- Αιτίες:
 1. Εκθετική αύξηση της πολυπλοκότητας του προβλήματος
 2. Ανεπαρκής τιμή του τ παρά την αύξησή του σε 5
 3. Ευαισθησία στο seed παρά την προσεκτική επιλογή του (2979)
 4. Περιορισμοί του βασικού αλγορίθμου για μεγάλης κλίμακας προβλήματα

Κεφάλαιο 7ο

Οπτική απεικόνιση σενάριου N=50:



7.13: Τελευταία επανάληψη της φάσης 1 για 50 πόλεις.



7.14: Τελευταία επανάληψη της φάσης 2 για 50 πόλεις.



7.15: Τελική διαδρομή για 50 πόλεις.



7.16: Τελική διαδρομή με χρήση μεθόδου Simplex για 50 πόλεις.

Τελική σειρά επίσκεψης

Πάργα -> Ηγουμενίτσα -> Αμφιλοχία -> Μεσολόγγι -> Πάτρα -> Αμαλιάδα -> Πύργος -> Ναύπακτος -> Αλεξανδρούπολη -> Σαμοθράκη -> Λίμνος -> Ηράκλειο -> Άγιος Νικόλαος -> Ρέθυμνο -> Χανιά -> Σάμος -> Κως -> Μυτιλήνη -> Κέα -> Χαλκίδα -> Μέτσοβο -> Άρτα -> Μήλος -> Αθήνα -> Θήβα -> Ναύπλιο -> Κόρινθος -> Άργος -> Τρίπολη -> Σπάρτη -> Καλαμάτα -> Κάρπαθος -> Νάουσα -> Πρέβεζα -> Αγρίνιο -> Καβάλα -> Δράμα -> Σέρρες -> Θεσσαλονίκη -> Κιλκίς -> Γιαννιτσά -> Έδεσσα -> Βέροια -> Κοζάνη -> Γρεβενά -> Ιωάννινα -> Πάργα.

7.7 Ανάλυση της Συμπεριφοράς των Παραμέτρων

Παρατηρήσεις:

- Για $N \leq 20$: Τα $\tau=2-3$ είναι επαρκή
- Για $N=50$: Το $\tau=5$ δεν αρκεί για τη διατήρηση της απόδοσης
- Προτεινόμενη βελτίωση: Δυναμικό $\tau = \max(3, \text{round}(\sqrt{N}))$

Επίδραση του E (Εκθέτης Ανοχής)

Η χρήση $E=20$ (ανοχή 10^{-20}) εξασφαλίζει:

- Υψηλή ακρίβεια σύγκλισης
- Κατάλληλη για μικρά προβλήματα
- Μπορεί να προκαλεί υπερ-fitting για μεγάλα προβλήματα

Προτεινόμενη βελτίωση: Για μεγαλύτερα N, ένας μικρότερος εκθέτης ($E=13-15$) μπορεί να προσφέρει καλύτερη ισορροπία μεταξύ ακρίβειας και ευελιξίας.

7.8 Σημαντικότητα Παραμέτρων

1. Seed Selection:

- Κρίσιμος παράγοντας για τη σταθερότητα
- Απαιτεί εκτεταμένο tuning για μεγάλα N
- Προτείνεται εκτέλεση πολλαπλών runs με διαφορετικά seeds

2. Tau (τ) Selection:

- Βασική παράμετρος του Divide and Conquer
- Πρέπει να κλιμακώνεται με το N
- Προτεινόμενη φόρμουλα: $\tau = \max(3, \text{round}(\sqrt{N}))$

7.9 Επίλογος και Συμπεράσματα

Η παρούσα μελέτη αποδεικνύει ότι τα Hopfield Networks με τον αλγόριθμο Divide and Conquer μπορούν να αποτελέσουν αποτελεσματική προσέγγιση για την επίλυση του TSP, ιδιαίτερα για προβλήματα μικρής και μεσαίας κλίμακας. Τα βασικά συμπεράσματα της έρευνας είναι:

Ισχυρά Σημεία:

- Εξαιρετική απόδοση για $N \leq 20$: Ποσοστιαία διαφορά κάτω του 2% από τη βέλτιστη λύση Simplex
- Σχετική απλότητα υλοποίησης: Η δομή του Divide and Conquer είναι εννοιολογικά κατανοητή

- Παραμετρική ευελιξία: Δυνατότητα fine-tuning μέσω seed, τ , και άλλων παραμέτρων
- Φυσική ερμηνεία: Το δίκτυο προσομοιώνει φυσικές διαδικασίες ελαχιστοποίησης ενέργειας

Αδυναμίες:

- Προβλήματα κλιμάκωσης: Δραματική μείωση απόδοσης για $N \geq 50$
- Παραμετρική ευαισθησία: Η απόδοση εξαρτάται κρίσιμα από την επιλογή seed και τ
- Τοπικά ελάχιστα: Τάση εγκλωβισμού σε υποβέλτιστες λύσεις για μεγάλα προβλήματα

7.10 Θεωρητικές Επιβεβαιώσεις

Τα αποτελέσματα επιβεβαιώνουν την θεωρητική ανάλυση των Hopfield Networks:

1. Ενεργειακό τοπίο: Για μικρά N το ενεργειακό τοπίο είναι σχετικά απλό, επιτρέποντας αποτελεσματική εύρεση του global minimum
2. Κλιμάκωση πολυπλοκότητας: Η εκθετική αύξηση του αριθμού τοπικών ελαχίστων με το N δικαιολογεί τη μειωμένη απόδοση

7.11 Πρακτικές Εφαρμογές

Η μέθοδος είναι κατάλληλη για:

- Logistics: Εσωτερική διανομή σε περιφερειακό επίπεδο (10-20 στάσεις)
- Τουρισμός: Προγραμματισμός ημερήσιων διαδρομών επίσκεψης αξιοθέατων
- Real-time applications: Όπου απαιτείται γρήγορη εύρεση καλής (όχι απαραίτητα βέλτιστης) λύσης

7.12 Μελλοντικές Βελτιώσεις:

- Adaptive τ based on problem size: Υιοθέτηση δυναμικής φόρμουλας για το τ ανάλογα με το N
- Multi-restart με διαφορετικά seeds: Εκτέλεση πολλαπλών προσομοιώσεων με διαφορετικά seeds
- Βελτιωμένες στρατηγικές για τη Phase 2: Πιο εξελιγμένοι αλγόριθμοι για τη σύνδεση των αλυσίδων

Το Hopfield Network με Divide and Conquer αποδεικνύεται αποτελεσματική μέθοδος για μικρής έως μεσαίας κλίμακας TSP προβλήματα, με την αναγκαιότητα βελτίωσης της κλιμάκωσης για μεγαλύτερα προβλήματα.

Βιβλιογραφία

[1]García, L., Talaván, P. M., & Yáñez, J. (2016). Improving the Hopfield model performance when applied to the traveling salesman problem: A divide-and-conquer scheme. *Neural Processing Letters*, 44, 363-385. <https://doi.org/10.1007/s11063-015-9460-y>

[2]Ein alter Commis-Voyageur. (1832). *Der Handlungsreisende—wie er sein soll und was er zu thun hat, um Aufträge zu erhalten und eines glücklichen Erfolgs in seinen Geschäften gewiss zu sein—Von einem alten Commis-Voyageur*. B. Fr. Voigt.

[3]_Schrijver, A. (2003). *Combinatorial optimization*. Springer.

[4]Spears, T. B. (1994). *100 years on the road: The traveling salesman in American culture*. Yale University Press.

[5]Friedman, W. A. (2004). *Birth of a salesman: The transformation of selling in America*. Harvard University Press.

[6]Brockett, L. P. (1871). *The commercial traveller's guide book: Being a complete manual for the use of commercial travellers and sellers of goods by sample*. H. Dayton & Company.

[7]Hamilton, W. R. (1857). Memorandum respecting a new system of roots of unity. *Philosophical Magazine*, 12, 446.

[8] Menger, K. (1930). Das Botenproblem. *Ergebnisse eines Mathematischen Kolloquiums*, 2, 11-12.

[9]Flood, M. M. (1956). The traveling-salesman problem. *Operations Research*, 4(1), 61-75.

[10] Whitney, H. (1934). Planar graphs. *Fundamenta Mathematicae*, 21(1), 73-84.

[11]Land, A. H., & Doig, A. G. (1960). An automatic method of solving discrete programming problems. *Econometrica*, 28(3), 497-520

[12] Karp, R. M. (1972). Reducibility among combinatorial problems. In R. E. Miller & J. W. Thatcher (Eds.), *Complexity of computer computations* (pp. 85-103). Plenum Press.

[13] Lin, S., & Kernighan, B. W. (1973). An effective heuristic algorithm for the traveling-salesman problem. *Operations Research*, 21(2), 498-516.

[14] Grötschel, M., Padberg, M. W., & Rinaldi, G. (1991). Optimization of a 532-city symmetric traveling salesman problem by branch and cut. *Operations Research*, 39(4), 508-516.

[15] Applegate, D. L., Bixby, R. E., Chvátal, V., & Cook, W. J. (2007). *The traveling salesman problem: A computational study*. Princeton University Press.

- [16] Reinelt, G. (1991). TSPLIB—A traveling salesman problem library. *ORSA Journal on Computing*, 3(4), 376-384.
- [17] Cook, S. A. (1971). The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing* (pp. 151-158). ACM.
- [18] **Frontier Supercomputer**: Oak Ridge National Laboratory. (2022). Frontier supercomputer. <https://www.olcf.ornl.gov/frontier/>
- [19] Miller, C. E., Tucker, A. W., & Zemlin, R. A. (1960). Integer programming formulation of traveling salesman problems. *Journal of the ACM*, 7(4), 326-329.
- [20] Cook, S. A. (1971). The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing* (pp. 151-158). ACM.
- [21] Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability: A guide to the theory of NP-completeness*. W. H. Freeman.
- [22] Croes, G. A. (1958). A method for solving traveling-salesman problems. *Operations Research*, 6(6), 791-812.
- [23] Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley.
- [24] Dorigo, M., & Stützle, T. (2004). *Ant colony optimization*. MIT Press.
- [25] Lenstra, J. K., & Rinnooy Kan, A. H. G. (1974). Some simple applications of the travelling salesman problem (BW 38/74). Stichting Mathematisch Centrum.
- [26] Saleh, H. A., & Chelouah, R. (2004). The design of the global navigation satellite system surveying networks using genetic algorithms. *Engineering Applications of Artificial Intelligence*, 17, 111-122.
- [27] Hopfield, J. J., & Tank, D. W. (1985). "Neural" computation of decisions in optimization problems. *Biological Cybernetics*, 52(3), 141-152. <https://doi.org/10.1007/BF00339943>
- [28] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115-133. <https://doi.org/10.1007/BF02478259>
- [29] Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386-408. <https://doi.org/10.1037/h0042519>
- [30] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533-536. <https://doi.org/10.1038/323533a0>
- [31] Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7), 1527-1554. <https://doi.org/10.1162/neco.2006.18.7.1527>

- [32] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25, 1097–1105.
- [33] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 5998–6008. <https://arxiv.org/abs/1706.03762>
- [34] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.
- [35] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.
- [36] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press
- [37] Edwards, S. F., & Anderson, P. W. (1975). Theory of spin glasses. *Journal of Physics F: Metal Physics*, 5(5), 965-974.
- [38] Cohen, M. A., & Grossberg, S. (1983). Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(5), 815-826.
- [39] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8), 2554–2558
- [40] Hopfield, J. J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences*, 81(10), 3088–3092.
- [41] Hebb, D. O. (1949). *The Organization of Behavior: A Neuropsychological Theory*. Wiley.
- [42] Talaván, P. M., & Yáñez, J. (2005). A continuous Hopfield network equilibrium points algorithm. *Computers & Operations Research*, 32(8), 2179–2196.
- [43] Bruck, J., & Goodman, J. W. (1988). A generalized convergence theorem for neural networks. *IEEE Transactions on Information Theory*, 34(5), 1089–1092.
- [44] Cottrell, M. (1988). Stability and attractivity in associative memory networks. *Biological Cybernetics*, 58(2), 129–139.
- [45] Zhong, N., Chu, H., Luo, Y., & Gan, W. (2017). Dynamic step continuous Hopfield neural network for solving traveling salesman problem. *Neural Processing Letters*, 45(2), 575-588.
- [46] Luo, Y. (2019). Design and improvement of Hopfield network for traveling salesman problem. *Journal of Algorithms & Computational Technology*, 13, 1-15.

- [47] Zhong, H., Feng, Y., Tang, C., Xiong, N., Dai, X., & Zhang, W. (2022). Collaborative neurodynamic optimization for solving traveling salesman problem. *Neurocomputing*, 483, 263-275.
- [48] Glover, F., Kochenberger, G., & Du, Y. (2019). Quantum bridge analytics I: A tutorial on formulating and using QUBO models. *4OR*, 17(4), 335–371.
- [49] Li, X., Wang, J., & Wang, L. (2021). Collaborative neurodynamic optimization with discrete Hopfield networks for traveling salesman problem. *IEEE Transactions on Neural Networks and Learning Systems*, 32(4), 1715-1726.
- [50] Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *IEEE International Conference on Neural Networks*, 4, 1942–1948.
- [51] Hu, X., & Duan, H. (2022). Accelerated augmented Lagrangian Hopfield neural network for solving traveling salesman problem. *Neural Computing and Applications*, 34(12), 9635-9648.
- [52] Bertsekas, D. P. (2016). *Nonlinear Programming* (3rd ed.). Athena Scientific.
- [53] Ahn, K., & Sra, S. (2020). Understanding Nesterov's acceleration via proximal point method (arXiv preprint arXiv:2005.08304).
- [54] Talaván, P. M., & Yáñez, J. (2002). Parameter setting of the Hopfield network applied to TSP. *Neural Networks*, 15(3), 363–373
- [55] Google Developers. (2023). *Distance Matrix API Documentation*. Google Cloud Platform.
- [56] García Rodríguez, L. (2018). *HopfieldNetworkToolbox* [Computer software]. GitHub. <https://github.com/mathinking/HopfieldNetworkToolbox>
- [57] Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3), 90-95.
- [58] Diestel, R. (2017). *Graph Theory* (5th ed.). Springer.

