7. Δίκτυα Hopfield

Σε μία πολύ γνωστή εργασία το 1982 ο Hopfield παρουσίασε 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 άλυτα προβλήματα, καθ' ότι όλα τα προβλήματα που ασχολείται είχαν λυθεί με άλλες μεθόδους. Μάλιστα οι λύσεις που προτείνονται δεν είναι καν οι καλύτερες, συνήθως είναι αργές, κ.λ.π. Εν τούτοις η χρήση των δικτύων αυτών είναι πολύ χρήσιμη για την διορατικότητα που μας δίνει για την φύση του προβλήματος.

Τα δίκτυα τύπου Hopfield αποτελούνται από ένα μόνο επίπεδο με πολλές όμως μονάδες (νευρώνες). Είναι διαδικά συστήματα, στο ότι κάθε μονάδα χαρακτηρίζεται από μία διαδική κατάσταση, δηλαδή μπορεί να έχει μία από δύο δυνατές τιμές. Συνήθως οι τιμές αυτές είναι 0 και 1, αλλά αυτό δεν είναι απαραίτητο. Μία κατάσταση του συστήματος είναι πλήρης όταν δίδονται οι τιμές των μονάδων από τις οποίες αποτελείται, π.χ. $s=(s_1, s_2, s_3, ...s_n)=(1,0, 1, 1.....0)$ για ένα δίκτυο με n μονάδες. Οι μονάδες του δικτύου έχουν πλήρη συνδεσμολογία, δηλαδή κάθε μονάδα συνδέεται με κάθε άλλη μονάδα στο σύστημα. Είχαμε δει προηγουμένως ότι για ένα δίκτυο με n μονάδες αυτό σημαίνει ότι 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 $w_{ij}=w_{ji}$, διότι τότε εξασφαλίζεται ότι το δίκτυο συγκλίνει και καταλήγει σε μία σταθερή κατάσταση.

Η δομή του δικτύου είναι ίδια με αυτήν του σχήματος 4.1, δηλαδή του προτύπου του στοιχειώδους αισθητήρα. Υπάρχουν πολλές μονάδες (νευρώνες) οι οποίες δίδουν σήμα s_i η κάθε μία, προς την μονάδα j. Γίνεται η γνωστή άθροιση και βρίσκουμε το S. Το νέο κριτήριο είναι τώρα 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 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.

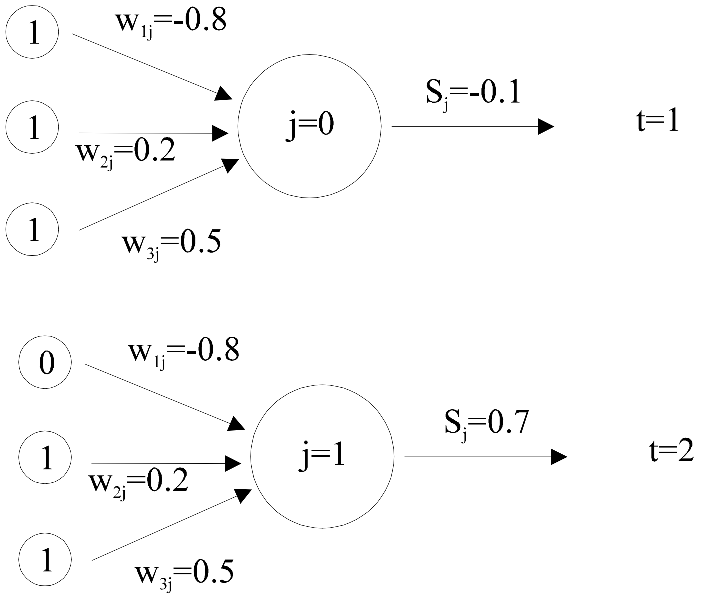
Το άθροισμα των εισερχόμενων σημάτων είναι:

$$S_j = \sum_{i=1}^{n} s_i w_{ji} \quad i \neq j \qquad \textbf{(1)}$$

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. Συνοπτικά

Εάν	Sj>0	τότε	s_j=1
	Sj<0	τότε	s_j=0 **(2)**

Κάθε φορά που θέλουμε να αναπροσαρμόσουμε την τιμή ενός νευρώνα εφαρμόζουμε την εξίσωση 7.2. Ας δούμε ένα παράδειγμα στο σχήμα 7.1. 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=0.7, 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.



ΣΧΗΜΑ 7.1

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.

Το δίκτυο Hopfield χαρακτηρίζεται από μία παράμετρο Ε, που εξαρτάται από τα σήματα εισόδου και τα βάρη, και η οποία είναι αντίστοιχη της ενέργειας σε φυσικά προβλήματα. Η παράμετρος Ε δίδεται από:

$$\mathrm{E} = -\frac{1}{2}\sum_j \sum_i w_{ji} s_j s_i \qquad i \neq j \qquad \textbf{(3)}$$

Με τις αλλαγές που πραγματοποιούνται στα s, το Ε αλλάζει με τον χρόνο. Η απαίτηση του δικτύου Hopfield είναι ότι η συνάρτηση Ε κατά την διάρκεια της αναπροσαρμογής ελαχιστοποιείται, όπως θα περιμέναμε και σε ένα φυσικό πρόβλημα. Σε κάθε αναπροσαρμογή η ενέργεια Ε ή ελαττώνεται ή παραμένει η ίδια. Ποτέ δεν μεγαλώνει, κατ' αντίθεση με τα δίκτυα Boltzmann που είδαμε στο προηγούμενο κεφάλαιο.

Όταν αναπροσαρμόζεται ο νευρώνας j τότε

$$\mathrm{E}_j = -\frac{1}{2}\sum_i w_{ji} s_j s_i \qquad i \neq j \qquad (4)$$

$$\mathrm{E}_j = -\frac{1}{2} s_j \sum_i w_{ji} s_i \qquad i \neq j \qquad \textbf{(5)}$$

και αυτό συμβαίνει διότι το σήμα s_j είναι σταθερό και μόνον τα s_i αλλάζουν ώστε να φέρουν την αναπροσαρμογή του j. Εάν δεν αλλάξει καθόλου η E_j τότε παλιά και η νέα κατάσταση θα είναι ίδιες. Αν αλλάξει τότε η αλλαγή θα είναι:

$$\Delta\mathrm{E}_j = \mathrm{E}_j(\nu\varepsilon\alpha) - \mathrm{E}_j(\pi\alpha\lambda\iota\alpha) = -\frac{1}{2}\Delta s_j \sum w_{ji} s_i \qquad \textbf{(6)}$$

όπου $\Delta s_j = s_j(\nu\varepsilon o) - s_j(\pi\alpha\lambda\alpha\iota o)$ **(7)**

Εάν το s_j αλλάξει από 0 σε 1 τότε $\Delta s_j = 1$

και $\sum_i w_{ji} s_i \geq 0$ (8)

Αυτό έχει ως αποτέλεσμα να γίνει το ΔΕ αρνητικό, καθ' ότι σύμφωνα με την εξίσωση (7.χ) $\Delta\mathrm{E}j \leq 0$. Αντίθετα, εάν η αλλαγή είναι από 1 σε 0, τότε $\Delta s_j = -1$ το άθροισμα είναι αρνητικό, οπότε και $\Delta\mathrm{E}j$<0. Παρατηρούμε ότι η αλλαγή του Ε είναι αρνητική ή μηδέν, για οποιοδήποτε αλλαγή της κατάστασης του j. Έτσι το δίκτυο πάντοτε θα συγκλίνει. Αυτό σημαίνει ότι το σύστημα θα βαίνει προς ένα ελάχιστο ενέργειας. Όπως και με προηγούμενες μεθόδους το ελάχιστο αυτό μπορεί να είναι τοπικό ή γενικό. Αν τύχει το σύστημα να "πέσει" σε ένα τοπικό ελάχιστο τότε δεν μπορεί να ξεφύγει και έτσι τελικά δεν μπορεί να βρει το γενικό ελάχιστο, όπως γίνεται π.χ. με το δίκτυο Boltzmann του προηγούμενου κεφαλαίου. Τότε απλώς το σύστημα θα δώσει μια "καλή" λύση, αλλά όχι την καλύτερη. Εάν θέλουμε πραγματικά να βρούμε την καλύτερη λύση, τότε πρέπει να ξανα-αρχίσουμε την διαδικασία με διαφορετικές αρχικές συνθήκες, ελπίζοντας ότι δεν θα παγιδευθεί σε τοπικό ελάχιστο όπως πριν.

Δίκτυα με συνεχείς τιμές

Σε όσα είδαμε παραπάνω οι δυνατές τιμές των καταστάσεων των νευρώνων ήταν 0 και 1, δηλαδή το σύστημα είναι διαδικό. Θα μπορούσαν όμως οι τιμές αυτές να είναι σε μία συνεχή περιοχή, και όχι αναγκαστικά διαδικές. Ένα τέτοιο δίκτυο θα μπορούσε να κάνει πιο πολλά πράγματα, γιατί δεν έχει πλέον τον περιορισμό αυτό των τιμών σε 0/1. Έχει την ίδια τοπολογία όπως και το διαδικό δίκτυο. Τα βάρη έχουν την συμμετρική εξίσωση $w_{ij}=w_{ji}$. Η αναπροσαρμογή γίνεται τώρα με την σιγμοειδή συνάρτηση, αντί του κατωφλίου που χρησιμοποιεί το διαδικό δίκτυο. Η σιγμοειδής συνάρτηση είναι η εξίσωση 5.1 (με γραφική παράσταση στο σχήμα 5.2). Αυτή είναι η συνάρτηση:

$$f(x) = \frac{1}{1+e^{-x}} \qquad \textbf{(9)}$$

Μία άλλη σιγμοειδής συνάρτηση είναι η tanhx. Υπάρχουν μόνον λίγες ακόμα τέτοιες συναρτήσεις με τις ιδιότητες που θέλουμε να έχουν. Όπως είχαμε δει και στο κεφάλαιο 5 η σιγμοειδής συνάρτηση λειτουργεί ως εξής: Για τιμές της παραμέτρου x πολύ κάτω από το κατώφλι (που είναι 0), η τιμή της εξόδου είναι σχεδόν 0. Για τιμές πολύ πάνω από το κατώφλι, η έξοδος είναι σχεδόν 1. Για τιμές που είναι στο ενδιάμεσο διάστημα η συνάρτηση αυξάνεται απότομα.

Στην περίπτωση των συνεχών τιμών, οι αναπροσαρμογές στους νευρώνες δεν γίνονται σε τακτά χρονικά διαστήματα (αυθαίρετες μονάδες χρόνου), αλλά στον συνεχή χρόνο.

Ισχύει η εξίσωση:

$$C_j \frac{ds_j}{dt} = \sum_i w_{ji} V_i - \frac{s_j}{R_j} + I_j \qquad \textbf{(10)}$$

όπου C_j >0 είναι μία σταθερά, R_j >0 είναι μία σταθερά που ελέγχει την αντίσταση που παρουσιάζει στο να μηδενίζεται η τιμή του νευρώνα j. I_j είναι εξωτερικό σήμα στον νευρώνα j. V_i είναι η έξοδος του νευρώνα i μετά την εφαρμογή της σιγμοειδούς συνάρτησης. Η εξίσωση ενέργειας τώρα είναι:

$$\mathrm{E} = -\frac{1}{2}\sum_j \sum_i w_{ji} s_j s_i - \sum_j s_j I_j \qquad \textbf{(11)}$$

όπου I_j είναι ο συντελεστής ορμής του σήματος εισόδου.

Συνειρμική μνήμη

Η εφαρμογή αυτή είναι η μία από τις δύο κύριες περιπτώσεις που είναι χρήσιμα τα δίκτυα Hopfield. Συνειρμική μνήμη ονομάζουμε την ιδιότητα που έχει ένα σύστημα που στην περίπτωση που ανακαλεί ένα τμήμα μόνο μιας μνήμης, να μπορεί να αναπαράγει ολόκληρη την μνήμη, ή τουλάχιστον ένα μεγάλο τμήμα της. Ένα καλό παράδειγμα είναι π.χ. όταν ακούσουμε τις πρώτες νότες ενός μουσικού σκοπού, ότι μπορούμε και θυμόμαστε τις υπόλοιπες (τις παρακάτω) νότες χωρίς άλλη εξωτερική βοήθεια. Αυτό είναι αντίθετο με τον τρόπο που δουλεύουν οι υπολογιστές, οι οποίοι έχουν σταθερές και ορισμένες διευθύνσεις για την συνηθισμένη μνήμη. Έτσι στους υπολογιστές δίδεται μια συγκεκριμένη διεύθυνση και μπορεί να εξετασθεί το περιεχόμενο μόνο της διεύθυνσης αυτής.

Ας εξετάσουμε ένα παράδειγμα, στο οποίο η μνήμη είναι ένα διάνυσμα από 0/1. Κάθε διάνυσμα είναι το όνομα ενός προσώπου και συνδέεται με ένα συγκεκριμένο χρώμα. Κάθε διάνυσμα είναι μία κατάσταση του συστήματος που συνεπάγεται ένα ελάχιστο της ενέργειας. Εάν το σύστημα βρεθεί σε μία αρχικά τυχαία κατάσταση, τότε θα αρχίσει να αναπροσαρμόζεται μέχρις ότου φθάσει στο ελάχιστο ενέργειας, που θα είναι η σωστή απάντηση για το δίκτυο ως προς την αντιστοιχία ονόματος-χρώματος. Στο σχήμα 7.2 βλέπουμε τρία ονόματα: Πέτρος, Χάρης, Ελένη και τα τρία αντίστοιχα χρώματα, κόκκινο, πράσινο και μπλε.

Στο παράδειγμα αυτή η πλήρης μνήμη χρειάζεται 24 νευρώνες, 12 για το όνομα και 12 για το χρώμα. Στην παράγραφο (1) βλέπουμε ότι κάθε διάνυσμα αντιστοιχεί σε ένα όνομα και χρώμα, Πέτρος-κόκκινο, Χάρης-πράσινο, Ελένη-μπλε, και λέμε ότι το δίκτυο έχει κάνει τον συνειρμό του Πέτρου με το κόκκινο, κ.λ.π. Στην παράγραφο (2) βλέπουμε ότι όταν παρουσιάζουμε μόνον το όνομα χωρίς κανένα χρώμα απαιτούμε το σύστημα να βρει το αντίστοιχο χρώμα. Δίνουμε το όνομα Πέτρος και θέλουμε το σύστημα να βρει την σχέση Πέτρος-κόκκινο. Τέλος στην παράγραφο (3) βλέπουμε την περίπτωση που έχουμε θόρυβο όταν δίνουμε ένα σήμα με θόρυβο π.χ. το Χάρης-πράσινο, το δίκτυο μετά από την αναπροσαρμογή θα πρέπει να μας δώσει την ίδια σωστή απάντηση αλλά χωρίς θόρυβο.

(1) Ζεύγη συνειρμικής μνήμης

Πέτρος	001100001101	|	001101111000	κόκκινο
Χάρης	010100111110	|	010111100111	πράσινο
Ελένη	100011000111	|	011100011100	μπλε

(2) Παρουσιάζεται τμήμα μόνο της μνήμης

Πέτρος 001100001101 | - - - - - - - - - - - κενό

Ανακαλείται μετά από προσαρμογή

Πέτρος 001100001101 | 001101111000 κόκκινο

(3) Παρουσιάζεται ένα θορυβώδες σχήμα

Χάρης 001100111111 | 011011100111 πράσινο

Ανακαλείται μετά από προσαρμογή

Χάρης 001100001101 | 010111100111 πράσινο

Σχήμα 7.2: Παράδειγμα συνειρμικής μνήμης

Ένα από τα πιο χαρακτηριστικά σημεία των νευρωνικών δικτύων είναι αυτό το τελευταίο, δηλαδή το ότι μπορεί ένα νευρωνικό δίκτυο να διορθώνει ένα σήμα με θόρυβο και να δίνει την σωστή απάντηση. Πάντα υπάρχει το καίριο ερώτημα ως προς το ποιο μπορεί να είναι το ανώτερο ποσοστό που μπορεί να συμβαίνει αυτό. Και βέβαια δεν υπάρχει μία γενική απάντηση για όλα τα προβλήματα, αλλά εκτιμάται εμπειρικά ότι κατά μέσο όρο το ποσοστό αυτό είναι γύρω στο 15%. Ο αριθμός αυτός λοιπόν πρέπει να χρησιμοποιείται προσεκτικά.

Ο αριθμός των νευρώνων που πρέπει να περιέχει το νευρωνικό δίκτυο πρέπει να είναι ίσος με τον αριθμό των στοιχείων των προτύπων τα οποία θέλουμε να κρατήσει στην συνειρμική μνήμη. Κάθε πρότυπο, όπως είδαμε στο παράδειγμα του σχήματος 7.2, είναι ένα διάνυσμα από 0/1, και συμβολίζεται:

$$\mathrm{A}_p = (a_{p1}, a_{p2}, a_{p3}, ...a_{pn}) \qquad (12)$$

όπου α_{pi} = το στοιχείο i του προτύπου p. Αν υπάρχουν m πρότυπα τότε:

$$w_{ji} = \sum_{p=1}^{m} (2a_{pi} - 1)(2a_{pj} - 1) \qquad \mathbf{(13)}$$

Οι όροι ($2\alpha_{pi}$-1) και ($2\alpha_{pj}$-1) μπορεί να έχουν τιμή -1 ή 1 μόνον. Όταν συμβεί να έχουμε $\alpha_{pj}=\alpha_{pi}$, τότε το w_{ji} αυξάνεται κατά 1, καθ' ότι δύο στοιχεία του προτύπου p είναι ολόιδια. Όταν συμβεί $\alpha_{pj} \neq \alpha_{pi}$ τότε το w_{ji} ελαττώνεται κατά 1, καθ' ότι δύο στοιχεία του προτύπου p είναι διαφορετικά. Αυτό ισχύει και γίνεται για όλα τα στοιχεία και για όλα τα πρότυπα. Με τον τρόπο αυτό βρίσκουμε τα βάρη w_{ji}, τα οποία είναι πλέον καθορισμένα, και δεν αλλαζουν κατά την διάρκεια της εκπαίδευσης. Παρατηρούμε λοιπόν ότι η τελική αυτή επιλογή των βαρών που βασίζεται στην εξίσωση (7.χ) εξαρτάται από την επιλογή των προτύπων τα οποία παρουσιάζονται στο δίκτυο.

Υπάρχουν όμως και αρκετοί περιορισμοί και δυσκολίες που παρουσιάζουν τα δίκτυα Hopfield όταν χρησιμοποιούν συνειρμική μνήμη. Δεν είναι ικανά να βρίσκουν πάντοτε την σωστή απάντηση, αλλά μερικές φορές δίνουν λανθασμένους συνδιασμούς (spurious states). Μερικές φορές δίνουν ιδιαίτερη έμφαση σε μία λύση, χωρίς λόγο, την οποία ανακαλούν πιο συχνά από ότι πρέπει. Ο λόγος που συμβαίνει αυτό είναι σχετικός με την πολυπλοκότητα της γνωστής επιφάνειας που ορίζει το σύστημα. Αν η αρχική κατάσταση είναι κοντά σε ένα τοπικό ελάχιστο, τότε το σύστημα εύκολα ελαχιστοποιεί την ενέργειά του εισερχόμενο σε αυτό. Αλλά μπορεί να μην αντιστοιχεί το ελάχιστο αυτό στην πιο σωστή λύση. Ο Hopfield προσπαθώντας να διορθώσει τους περιορισμούς αυτούς εισάγει έναν νέο όρο στην εξίσωση (7.χ), που γίνεται:

$$\Delta w_{ji} = -\varepsilon(2a_{pj} - 1)(2a_{pi} - 1) \quad \textbf{(14)}$$

όπου το ε είναι ένας μικρός παράγων, 0<ε<<1. Η εξίσωση αυτή δεν λύνει όλα τα προβλήματα, αλλά είναι οπωσδήποτε μία καλυτέρευση.

Το πρόβλημα του πλανώδιου πωλητή

Το πρόβλημα του πλανώδιου 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), τα οποία δεν έχουν ακριβείς λύσεις, και έτσι οι λύσεις που προτείνονται είναι "ευριστικές 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 πλανώδιος πωλητής ακολουθώντας μια συγκεκριμένη διαδρομή. Οι περιορισμοί που υπάρχουν στην διαδρομή είναι οι εξής: Ο πλανώδιος πωλητής πρέπει να επισκεφθεί κάθε πόλη μόνον μία φορά και στο τέλος πρέπει να επιστρέψει στην αρχική πόλη. Ζητείται να βρούμε την διαδρομή εκείνη η οποία είναι η μικρότερη. Ο Hopfield επρότεινε 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 μία σειρά n νευρώνων. Μόνο ένας νευρώνας σε κάθε σειρά μπορεί να είναι 1, ενώ όλοι οι άλλοι πρέπει να είναι 0. Ο νευρωνας που είναι ίσος με 1 δείκνει την σειρά με την οποία θα επισκεφθεί την συγκεκριμένη πόλη ο πλανώδιος πωλητής. Το σχήμα 7.1 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 7.1

Πόλη	1	2	3	4	5	6	7	8	9	10
A	0	0	0	1	0	0	0	0	0	0
B	0	1	0	0	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 πλανώδιος 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 7.1. Το πρόβλημα με n πόλεις έχει n!/(2n) διαφορετικές διαδρομές, επομένως έχει αυτό τον αριθμό των λύσεων. Αν n=10, τότε υπάρχουν 181440 διαφορετικές διαδρομές, ενώ αν n=60, τότε υπάρχουν $6.9x10^{79}$ διαφορετικές διαδρομές. Βλέπουμε λοιπόν το μέγεθος του προβλήματος.

Στην λύση που προτείνεται κάθε νευρώνας έχει δύο δείκτες, που δείχνουν την πόλη και την σειρά με την οποία ο πλανώδιος πωλητής θα την επισκεφθεί. Η συνάρτηση της ενέργειας μόνον για τις λύσεις αυτές που δίνουν μόνο ένα 1 σε κάθε στήλη και κάθε σειρά. Επίσης πρέπει να προτιμώνται οι λύσεις που έχουν την μικρότερη διαδρομή. Έτσι η ενέγεια θα δίνεται από:

$$E = (A/2)\sum_{x}\sum_{i}\sum_{j} V_{xi}V_{xj} \qquad j \neq i \qquad \text{1ος όρος}$$
$$+(B/2)\sum_{i}\sum_{x}\sum_{y} V_{xi}V_{yi} \qquad y \neq x \qquad \text{2ος όρος}$$

$$+(C/2)\sum_{x}\sum_{i}\left(V_{xi} - n^2\right) \qquad \text{3ος όρος}$$

$$+(D/2)\sum_{x}\sum_{y}\sum_{i} d_{xy}s_{xi}(V_{x,i+1} + V_{y,i-1}) \qquad \text{4ος όρος} \qquad \textbf{(15)}$$

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

Κάθε όρος στην εξίσωση της ενέγειας 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) μόνον όταν κάθε σειρά έχει μόνο ένα 1. Ο δεύτερος όρος δεν επιτρέπει στον πλανώδιο 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.

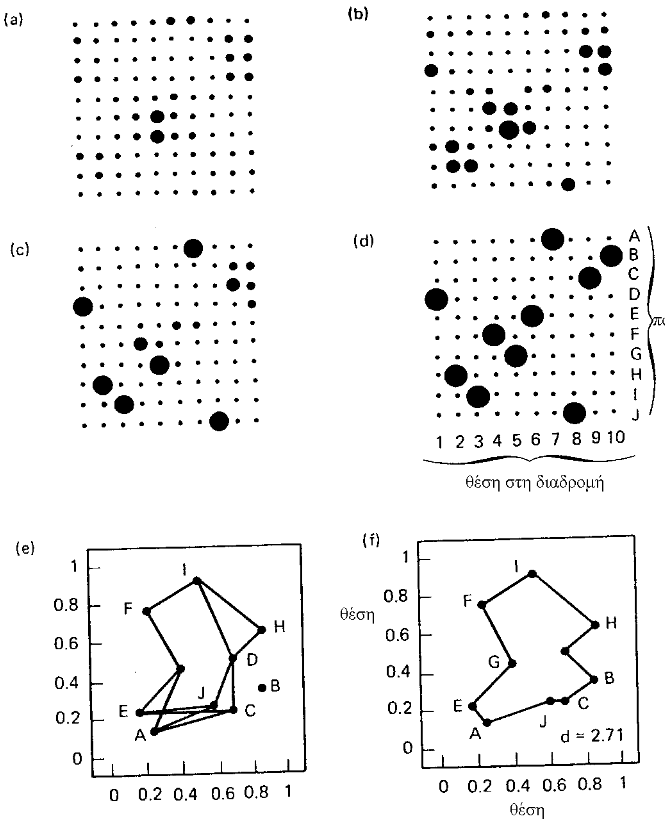
Οι τιμές των σταθερών A, B και C πρέπει να είναι μεγάλες, έτσι ώστε οι καταστάσεις που έχουν χαμηλή ενέγεια να παριστάνουν "καλές" διαδρομές. Η μεγάλη τιμή στο D εξασφαλίζει ότι θα βρεθεί μία σύντομη διαδρομή. Το επόμενο βήμα είναι να βρεθούν τα βάρη w. Έτσι έχουμε:

$$\begin{aligned} w_{xi,yi} = &-A\delta_{xy}(1-\delta_{ij}) \\ &-B\delta_{ij}(1-\delta_{xy}) \\ &-C \\ &-Dd_{xy}(\delta_{j,i+1} + \delta_{j,i-1}) \end{aligned} \qquad \textbf{(16)}$$

όπου δ_{ij} είναι το Kronecker, $\delta_{ij}=1$ 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. Επί πλέον κάθε νευρώνας έχει μία εξωτερική ορμή (bias):

$$\mathbf{I_{xj}=C_n} \qquad \textbf{(17)}$$

Η αρχική κατάσταση του νευρωνικού δικτύου είναι τυχαία, και το δίκτυο υφίσταται συνεχείς αναπροσαρμογές μεχρις ότου συγκλίνει σε μία σταθερή κατάσταση. Όταν επιτευχθεί η σύγκλιση τότε η διαδρομή του πλανώδιου πωλητή δίδεται από την τελική κατάσταση του δικτύου.



ΣΧΗΜΑ 7.3

Στο σχήμα 7.3 βλέπουμε την εξέλιξη του δικτύου ενώ αναπροσαρμόζει τις τιμές των μονάδων του. Κάθε πίνακας έχει διάσταση 10x10 για το πρόβλημα n=10. Η τιμή του στοιχείου του πίνακα είναι ανάλογη με την διάμετρο του κύκλου. Στα (α), (b), και (c) έχουμε φωτογραφίες από ενδιάμεσες καταστάσεις, ενώ στο (d) έχουμε την τελική λύση, αφού έχουν τελειώσει όλες οι αναπροσαρμογές. Στο (e) έχουμε την διαδρομή του πωλητή για μία ενδιάμεση τέτοια κατάσταση, που δείχνει ότι μία ενδιάμεση κατάσταση μπορεί να έχει διαδρομή με δύο πόλεις την ίδια χρονική στιγμή (δύο στοιχεία στην ίδια στήλη), ή να μην επισκέπτεται καθόλου κάποιες πόλεις (κανένα στοιχείο σε μία σειρά). Στο (f) έχουμε την τελική διαδρομή, η οποία και υπακούει όλους τους κανόνες του προβλήματος. Επί πλέον η συνολική απόσταση είναι η μικρότερη από άλλες πιθανές διαδρομές.

Το δίκτυο Hopfield λύνει το πρόβλημα TSP ικανοποιητικά για τιμές n=10-30, αλλά έχει προβλήματα για μεγαλύτερα n. Μας δίνει όμως ένα πολύ καλό παράδειγμα προβλημάτων που λύνουν τα νευρωνικά δίκτυα. Επί πλέον, το δίκτυο Hopfield έχει χαρακτηριστικά παράλληλης αρχιτεκτονικής, πράγμα που θα μπορούσε να είναι χρήσιμο σε αντίστοιχες πειραματικές διατάξεις.