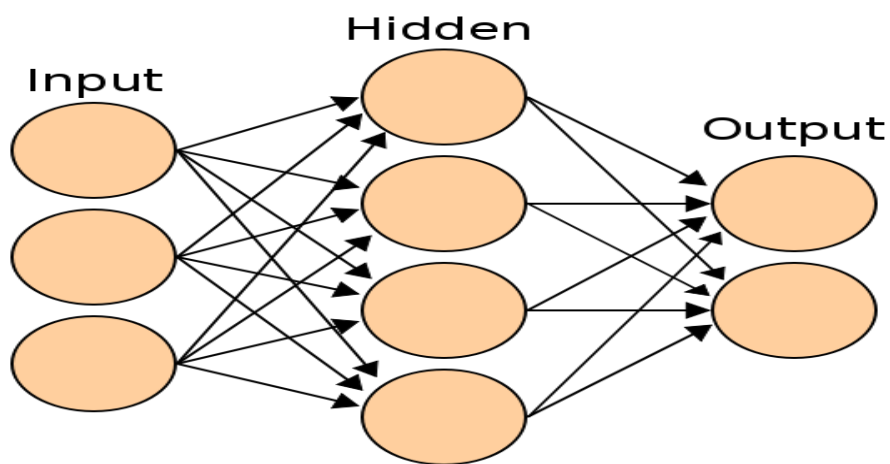




ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΗΠΕΙΡΟΥ
ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε



ΑΝΑΠΤΥΞΗ ΤΕΧΝΗΤΩΝ ΝΕΥΡΟΝΙΚΩΝ ΔΙΚΤΥΩΝ
ΚΑΙ ΕΚΠΑΙΔΕΥΣΗ ΜΕ PSO

Πτυχιακή Εργασία

του

Παργινού Διονύση

ΑΜ: 10055

Επιβλέπων Καθηγητής : Τσούλος Ιωάννης

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Τσούλο Ιωάννη για την δυνατότητα που μου έδωσε να πραγματοποιήσω την παρούσα εργασία όπως σε ένα τέτοιο ενδιαφέρον τομέα όπως είναι τα Νευρωνικά Δίκτυα καθώς και για την σημαντική βοήθειά του.

ΠΕΡΙΛΗΨΗ

Τα τεχνητά νευρωνικά δίκτυα είναι μια επιστημονική περιοχή η οποία έχει αναπτυχθεί κατά τις τελευταίες δεκαετίες και επικαλύπτει όλες σχεδόν τις θετικές επιστήμες και την μηχανολογία. Πρόκειται για ένα αφηρημένο αλγοριθμικό κατασκευάσμα το οποίο εμπίπτει στον τομέα της υπολογιστικής νοημοσύνης. Έχουν χρησιμοποιηθεί σε πολλές εφαρμογές και σε προβλήματα που με τους γνωστούς τρόπους αντιμετώπισης τους παρουσιάζουν δυσκολίες, με αποτέλεσμα να καθίσταται αναγκαίο η χρήση Τεχνητών Νευρικών Δικτύων. Η εφαρμογή τεχνικών βελτιστοποίησης για την επίλυση προβλημάτων είναι συνηθισμένη πρακτική. Εξαιτίας της αυξανόμενης ανάγκης στις μεθόδους βελτιστοποίησης σε όλους τους τομείς της επιστήμης, υπάρχει μεγάλη ζήτηση για αποτελεσματικές και γρήγορες εκτελέσεις σχετικών αλγορίθμων. Η εργασία αυτή χρησιμοποιεί την τεχνική της βελτιστοποίησης ώστε να «εκπαιδεύσει» ένα τεχνητό νευρωνικό δίκτυο.

Στο πρώτο κεφάλαιο αναπτύσσουμε την θεωρία των τεχνητών νευρωνικών δικτύων αναφέροντας τις βασικές αρχές τους αλλά και πώς «εκπαιδεύουμε» ένα τεχνητό νευρωνικό δίκτυο. Στο δεύτερο κεφάλαιο ασχολούμαστε με τις βασικές αρχές της βελτιστοποίησης και την ανάλυση της τεχνικής βελτιστοποίησης σμήνους σωματιδίων. Το κεφάλαιο 3 «εκπαιδεύουμε» το τεχνητό νευρωνικό δίκτυο μας με χρήση βελτιστοποίησης σμήνους σωματιδίων εξάγοντας και τα αποτελέσματα αυτού.

ABSTRACT

The artificial neuronics are a scientific area which has been developed during the latest decades and covers almost all the exact sciences and mechanical engineering. It is about an abstract algorithmic structure which comes under the field of calculating intelligence. They have been used in many applications and problems which face difficulties with the usual ways of confrontation resulting to the use of artificial Neuronics. The application of optimization techniques for the problem solution is a usual practice. Due to the increasing need for optimization methods in every scientific field, there is a great demand for effective and quick implementations of relevant algorithms. This project uses the optimization technique in order to “train” an artificial neuronics.

In the first chapter the theory of artificial neuronics is explained, referring to their basic principles but also how an artificial neuronics is “trained”. The second chapter refers to the basic principles of optimization and the analysis of optimization technique of a swarm of particles. In chapter three our artificial neuronics is “trained” using the optimization of swarm of particles eliciting its results.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΧΑΡΙΣΤΙΕΣ	2
ΠΕΡΙΛΗΨΗ	3
ABSTRACT	4
ΚΕΦΑΛΑΙΟ 1 - Τεχνητά Νευρωνικά Δίκτυα.....	6
1.1 Εισαγωγή στα Νευρωνικά Δίκτυα	6
1.2 Δομή του τεχνητού νευρώνα	7
1.3 Αρχιτεκτονικές τεχνητού νευρωνικού δικτύου	8
1.4 Εκπαίδευση νευρωνικών δικτύων.....	10
1.5 Εφαρμογές νευρωνικών δικτύων.....	13
ΚΕΦΑΛΑΙΟ 2 - Θεωρία της Βελτιστοποίησης.....	15
2.1 Εισαγωγή στη Βελτιστοποίηση.....	15
2.2 Εισαγωγή στη μέθοδο βελτιστοποίηση σμήνους σωματιδίων.....	17
2.2.1 Τροποποιήσεις αλγορίθμων βελτιστοποίησης σμήνους σωματιδίων.....	20
ΚΕΦΑΛΑΙΟ 3 - Εκπαίδευση Τεχνητών Νευρωνικών Δικτύων με Βελτιστοποίηση Σμήνους Σωματιδίων	
3.1 Εισαγωγή	21
3.2 Δομή Κώδικα	23
3.3 Πειραματικά Αποτελέσματα.....	25
Παράρτημα.....	29
Βιβλιογραφία.....	34

ΚΕΦΑΛΑΙΟ 1

Τεχνητά Νευρωνικά Δίκτυα

1.1 ΕΙΣΑΓΩΓΗ ΣΤΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Νευρωνικό δίκτυο ονομάζουμε ένα σύνολο κόμβων ή νευρώνων που ο καθένας συνδέεται με τους υπολοίπους, είναι εμπνευσμένο από το Κεντρικό Νευρικό Σύστημα το οποίο προσπαθεί να προσομοιώσει. Ουσιαστικά είναι ένα μαθηματικό μοντέλο επεξεργασίας και αξιολόγησης πληροφορίας όπου βασικό συστατικό ενός νευρωνικού δικτύου είναι ο νευρώνας. Αποτελείται από ένα μεγάλο αριθμό διασυνδεδεμένων πολλαπλά επεξεργαστών ή αλλιώς νευρώνες βασισμένοι σε κάποιες προδιαγραφές ώστε να δουλεύουν σε πλήρη αρμονία μεταξύ τους για να φέρουν σε πέρας ένα συγκεκριμένο έργο. Στην περίπτωση τεχνητών νευρώνων, πρόκειται για ένα αφηρημένο αλγοριθμικό κατασκευάσμα το οποίο εμπίπτει στον τομέα της υπολογιστικής νοημοσύνης, όταν στόχος του νευρωνικού δικτύου είναι η επίλυση κάποιου υπολογιστικού προβλήματος, ή της υπολογιστικής νευροεπιστήμης, όταν στόχος είναι η υπολογιστική προσομοίωση της λειτουργίας των βιολογικών νευρωνικών δικτύων με βάση κάποιο μαθηματικό μοντέλο τους. Γενικά κάθε νευρώνας δέχεται εισόδους, εκτελεί μία συνάρτηση ή υπολογισμό και στη συνέχεια δίνει το αποτέλεσμα στην έξοδο. Οι διασυνδέσεις μεταξύ των νευρώνων περιλαμβάνονται από ένα βάρος και το σύνολο των τιμών των βαρών που προσδιορίζει την συμπεριφορά του δικτύου η οποία και διαμορφώνεται από μία διαδικασία η οποία ονομάζεται εκπαίδευση.

Τα πρώτα νευρωνικά δίκτυα ξεκίνησαν τις τελευταίες δεκαετίες αλλά η μεγάλη ώθηση δόθηκε μετά το 1980. Το πρώτο μοντέλο νευρωνικού δικτύου δημιουργήθηκε από τους McCulloch και Pitts. Το 1949 ο Hebb εισάγει τον κανόνα μάθησης των νευρωνικών δικτύων στο βιβλίο << The organization of behavior >>. (Wikipedia)[11]

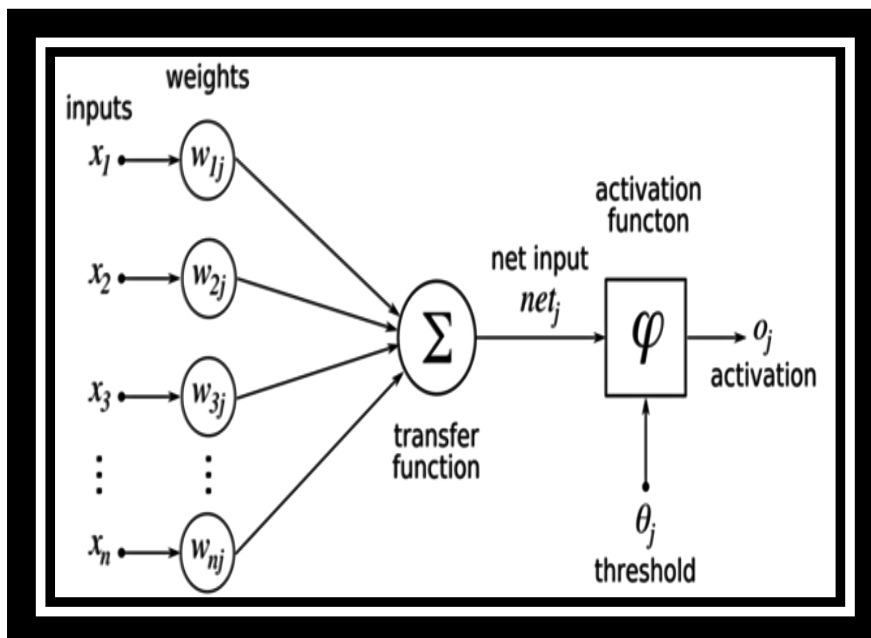
Το 1950 ο J. von Neumann χρησιμοποίησε το μοντέλο ως παράδειγμα για τις υπολογιστικές μηχανές. Μερικά χρόνια αργότερα το 1957 παρουσιάστηκε από τον F. Rosenblatt το μοντέλο του αισθητήρα (perceptron) και 2 χρόνια μετά το 1959 οι Widrow και Hoff ανέπτυξαν δύο νέα μοντέλα το Adaline και το Madaline. Το 1982 ο Hopfield έδειξε με μαθηματική απόδειξη πως ένα νευρωνικό δίκτυο μπορεί να χρησιμοποιηθεί ως αποθηκευτικός χώρος και πως μπορεί να επανακτήσει όλη την πληροφορία ενός συστήματος αν του δοθούν μερικά τμήματα

μόνο και όχι ολόκληρο το σύστημα, το 1986 οι McClelland και Rumelhart, στο << Parallel Distributed Processing>> προτείνουν μία νέα διαδικασία οπισθοδιάδοσης την λεγόμενη (propagation).(ROJA,2010)[8] Τέλος το 1985 έγιναν τα πρώτα συνέδρια και εκδόθηκαν τα πρώτα περιοδικά για νευρωνικά δίκτυα και δημιουργήθηκαν οι πρώτες εταιρίες νευρωνικών δικτύων.

1.2 Η ΔΟΜΗ ΤΟΥ ΤΕΧΝΗΤΟΥ ΝΕΥΡΩΝΑ

Ο Τεχνητός Νευρώνας είναι μια μονάδα δομική του τεχνητού νευρικού δικτύου. Μέσα σε αυτόν γίνεται όλη η επεξεργασία της πληροφορίας. Ο κάθε νευρώνας δέχεται μία πληροφορία, την επεξεργάζεται και δίνει στην έξοδο μία τιμή. Οι εισοδοί του είναι είτε έξοδοι άλλων νευρώνων είτε το πρωταρχικό σήμα εισόδου του δικτύου.

Στην παρακάτω εικόνα παρουσιάζεται το βασικό σχήμα του νευρώνα που συνήθως χρησιμοποιείται σε υλοποιήσεις τεχνητών νευρικών δικτύων.



Σχήμα 1: Σχηματική αναπαράσταση Τεχνητού Νευρώνα

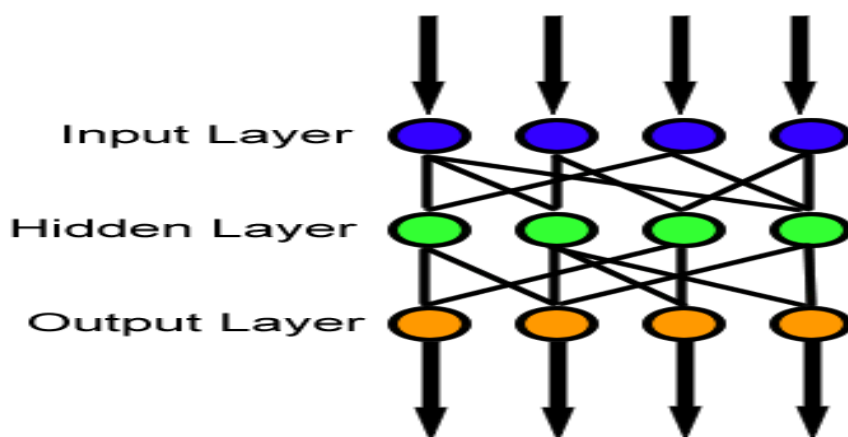
Στο σχήμα αυτό, η πληροφορία που εισάγουμε ρέει πάντα προς μία κατεύθυνση, από τα αριστερά προς τα δεξιά. Με βάση αυτό, διακρίνονται τρεις βασικές φάσεις της λειτουργίας του:

Στην πρώτη φάση η πληροφορία που εισάγεται σαν είσοδος πολλαπλασιάζεται με το βάρος που αντιστοιχείται σε αυτή, κατόπιν σε δεύτερη φάση οι σταθμισμένες εισοδοι και ένας εξωτερικός παράγοντας το κατώφλι (threshold) αθροίζονται και δίνουν το τοπικό πεδίο (net input). (Pauly 2008)[7]. Μέχρι εδώ ο νευρώνας δεν κάνει κάτι άλλο εκτός από το να δίνει ένα γραμμικό συνδυασμό των εισόδων. Τέλος στην τρίτη φάση εφαρμόζεται η συνάρτηση ενεργοποίησης ή συνάρτηση μεταφοράς (activation function) στο τοπικό πεδίο και στην έξοδο μας δίνει το αποτέλεσμα.

1.3 ΑΡΧΙΤΕΚΤΟΝΙΚΕΣ ΤΕΧΝΙΤΩΝ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ

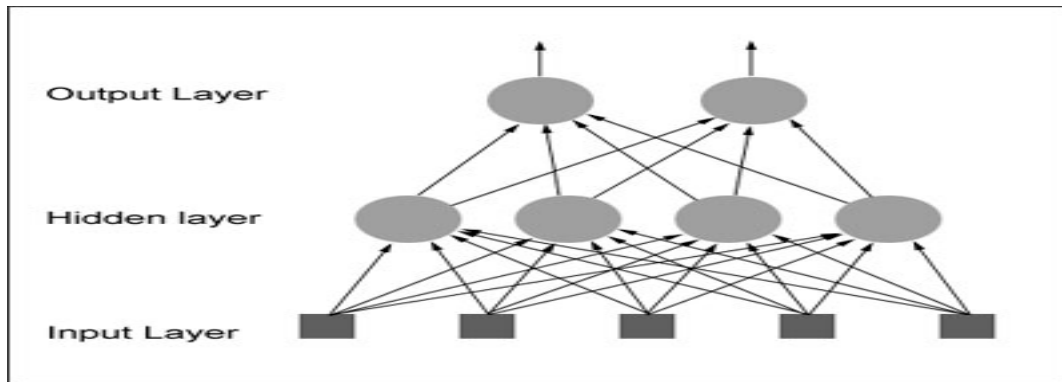
Τα τεχνητά νευρωνικά δίκτυα διακρίνονται από την αρχιτεκτονική τους, τη λειτουργία που πραγματοποιούν καθώς και τον τρόπο που εκπαιδεύονται. Σαν αρχιτεκτονική του τεχνητού νευρωνικού δικτύου ορίζεται η διάταξη των διασυνδέσεων που έχουν οι νευρώνες μεταξύ τους καθώς και το σύνολο των νευρώνων. Οι νευρώνες ενός δικτύου είναι δομημένοι σε μορφή επιπέδων και σχετίζονται με τον αλγόριθμο μάθησης που χρησιμοποιείται για την εκπαίδευση αυτού του δικτύου. Οι αρχιτεκτονικές των δικτύων χωρίζονται σε τρεις κατηγορίες:

- Τα δίκτυα πρόσθιας τροφοδότησης ενός επιπέδου (feed forward) : Είναι η πιο απλή αρχιτεκτονική ενός δικτύου. Περιέχουν ένα επίπεδο εισόδου αποτελούμενο από πηγαίους κόμβους, το οποίο δείχνει στο επόμενο επίπεδο νευρώνων εξόδου και όχι αντίστροφα.[12] Το σήμα κατευθύνεται από το επίπεδο εισόδου προς το επίπεδο εξόδου. Οι νευρώνες εισόδου μεταφέρουν το σήμα στην έξοδο χωρίς να κάνουν καμία επεξεργασία.



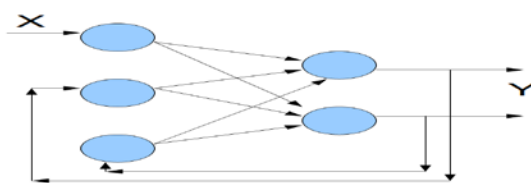
Σχήμα 2 : Δίκτυο (feed forward)

- Πολυεπίπεδα δίκτυα πρόσθιας τροφοδότησης : Υπάρχουν ένα ή περισσότερα κρυφά επίπεδα των οποίων οι νευρώνες ονομάζονται κρυφοί νευρώνες. Σε αυτό το δίκτυο οι νευρώνες έχουν σαν εισόδους τα σήματα εξόδου μόνου του προηγούμενου επιπέδου και το σήμα πηγαίνει από το επίπεδο εισόδου προς το επίπεδο εξόδου με την χρήση των κρυφών νευρώνων.



Σχήμα 3 : Πολυεπίπεδο δίκτυο πρόσθιας τροφοδότησης

- Αναδρομικά δίκτυα (recurrent neural network) : Σε αυτήν την κατηγορία υπάρχει ένας τουλάχιστον βρόγχος ανάδρασης, η έξοδος κάθε νευρώνα ανατροφοδοτεί την είσοδο των άλλων νευρώνων του ίδιου επιπέδου. Οι κλάδοι των βρόχων ανάδρασης περιέχουν στοιχεία μοναδιαίας χρονικής καθυστέρησης που συμβολίζονται με z^{-1} .



Σχήμα 4 : Αναδρομικό δίκτυο

1.4 ΕΚΠΑΙΔΕΥΣΗ ΤΕΧΝΗΤΩΝ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ

Ένα τεχνητό νευρωνικό δίκτυο κατασκευάστηκε για να μπορεί να επιλύει συγκεκριμένα προβλήματα ή να διενεργεί από μόνο του ορισμένες λειτουργίες, ωστόσο για να γίνει αυτό πρέπει να το εκπαιδεύσουμε σωστά. Αυτό γίνεται με την αναζήτηση του τρόπου προσαρμογής των εφαρμοζόμενων στις εισόδους βαρών ώστε να παράγεται το προσδοκώμενο αποτέλεσμα. Αυτό μπορεί να γίνει με δύο μεθόδους. Με την πρώτη από αυτές, την μελέτη με επίβλεψη, στο νευρωνικό δίκτυο εφαρμόζεται ένα διάνυσμα εισόδου και σε αυτό ένα διάνυσμα βαρών. (Κούρους Νικόλαος, 2012)[6] Από τον συνδυασμό αυτό παράγεται ένα αποτέλεσμα. Η απόκλιση του παραγόμενου διανύσματος και του διανύσματος στόχου ανατροφοδοτείται στην είσοδο. Τα βάρη αναπροσαρμόζονται με βάση την ανατροφοδότηση και τελικά λαμβάνεται εκ νέου το διάνυσμα εξόδου. Η διαδικασία επαναλαμβάνεται μέχρι η απόκλιση να ελαχιστοποιηθεί. Με την δεύτερη μέθοδο, την μελέτη χωρίς επίβλεψη, ελέγχονται οι εξόδοι που παράγονται μετά την εφαρμογή συγκεκριμένων εισόδων. Οι πιθανοί είσοδοι ομαδοποιούνται ανάλογα με τις παραγόμενες εξόδους. Με τον τρόπο αυτό μπορεί να προβλεφθεί η συμπεριφορά του δικτύου για δεδομένες εισόδους. Οι πληροφορίες αποθηκεύονται στα βάρη ενός νευρωνικού δικτύου. Η εκπαίδευση είναι ο προσδιορισμός του βάρους. Μετά από τον τρόπο εκπαίδευσης μπορούμε να διακρίνουμε δύο μεγάλες κατηγορίες των νευρωνικών δικτύων :

- Σταθερά δίκτυα : Τα βάρη των οποίων δεν μπορεί να αλλάξει $dW/dt = 0$. Σε αυτά τα δίκτυα τα βάρη είναι σταθερές σύμφωνα με το πρόβλημα το οποίο πρέπει να επιλυθεί
- Προσαρμοστικά δίκτυα : Τα βάρη των δικτύων αυτών είναι σε θέση να αλλάξουν, dW/dt δεν είναι 0.

Η παραπάνω διαδικασία είναι επανομαζόμενη και ως εκπαίδευση. Όλες οι μέθοδοι μάθησης χρησιμοποιούνται για την προσαρμοστικότητα των νευρωνικών δικτύων και χωρίζονται σε δύο κατηγορίες :

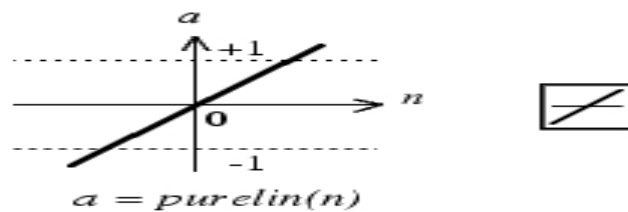
- Υπό Επίβλεψη Τεχνητά Νευρωνικά Δίκτυα:

Στα υπό επίβλεψη νευρωνικά δίκτυα, για κάθε έξοδο προσδιορίζεται μία αναμενόμενη έξοδος. Τα βάρη στα δεδομένα εισόδου αναπροσαρμόζονται ώστε να ελαχιστοποιηθεί η μέση τετραγωνική απόκλιση. Στα συστήματα αυτά μία διάταξη παίζει τον ρόλο του εκπαιδευτή του δικτύου. Το ληφθέν σήμα από το νευρωνικό δίκτυο λαμβάνεται ταυτόχρονα και από τον εκπαιδευτή του δικτύου. Το κύριο χαρακτηριστικό της επιβλεπόμενης μάθησης είναι η ύπαρξη του εξωτερικού δασκάλου ο οποίος με βάση τη γνώση και την εμπειρία του είναι ικανός να διδάξει στο σύστημα μάθησης τις επιθυμητές εξόδους για ένα σύνολο εισόδων εκπαίδευσης. Όταν ο δάσκαλος και το ΤΝΔ λάβουν ένα διάνυσμα εισόδου εκπαίδευσης, ο δάσκαλος δίνει στο ΤΝΔ μια επιθυμητή έξοδο η οποία παριστά τη βέλτιστη ενέργεια που πρέπει να κάνει το ΤΝΔ. Οι παράμετροι του δικτύου ανανεώνονται με βάση τόσο το διάνυσμα εκπαίδευσης όσο και το σήμα σφάλματος (δηλαδή της διαφοράς μεταξύ της πραγματικής απόκρισης $y(t)$ και της επιθυμητής απόκρισης $yd(t)$ του δικτύου). Ορίζοντας μια τετραγωνική συνάρτηση κόστους: όπου w είναι το διάνυσμα των προς επιλογή (ελεύθερων) παραμέτρων του συστήματος μάθησης (δηλαδή του ΤΝΔ), η ανανέωση / προσαρμογή των παραμέτρων παίρνει την μορφή του αλγόριθμου διόρθωσης σφάλματος που είδαμε στην προηγούμενη ενότητα. Η βήμα – προς – βήμα ανανέωση των παραμέτρων κάνει τελικά το ΤΝΔ μάθησης να μιμείται καλά τον δάσκαλο. Δυο περιπτώσεις αλγόριθμων επιβλεπόμενης μάθησης είναι ο αλγόριθμος Ελαχίστου Μέσου Τετραγώνου (Least Mean Square, LMS) και η γενίκευση του που είναι γνωστή ως Αλγόριθμος Ανάστροφης Διάδοσης (Back Propagation, BP).

- Μάθηση χωρίς επίβλεψη: η απόκριση του δικτύου βασίζεται στην ικανότητα του να αυτοοργανώνεται με βάση τα διανύσματα εισόδου καθώς δεν υπάρχουν διανύσματα εξόδου. Σε ένα συγκεκριμένο σύνολο εισόδων αντιδρά ένας συγκεκριμένος νευρώνας. Στην περίπτωση αυτή δίνεται η πληροφορία στο δίκτυο και δεν υπάρχει εξωτερικός εκπαιδευτής που να δίνει την επιθυμητή απόκριση και να επιβλέπει την διαδικασία μάθησης. (Ευστάθιος Ανδριανάκης, 2008)[3] Τα δείγματα εκπαίδευσης αποτελούνται μόνο από δείγματα εισόδου και δεν περιέχουν δείγματα εξόδου. Το δίκτυο χρησιμοποιεί έναν εσωτερικό έλεγχο, ψάχνει να βρει στα σήματα εισόδου κανονικότητα και εκπαιδεύονται ώστε οι έξοδοι να έχουν τα ίδια χαρακτηριστικά με τις εισόδους με αποτέλεσμα το δίκτυο να ελέγχει τον εαυτό του και διορθώνει τα σφάλματα στα δεδομένα με μηχανισμό ανάδρασης. Όταν το δίκτυο σταματήσει να αλλάζει τιμές στα βάρη τότε η εκπαίδευση τελείωσε.

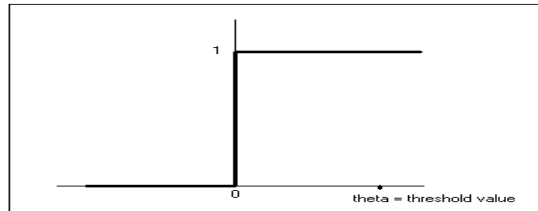
Μετά το στάδιο της εκπαίδευσης ακολουθεί το στάδιο του ελέγχου. Στη φάση αυτή ελέγχεται για το αν παράγονται τα αποτελέσματα που περιμένουμε. Αν τα αποτελέσματα είναι αντίθετα τότε γίνεται νέα εκπαίδευση. Η συμπεριφορά ενός τεχνητού νευρωνικού δικτύου εξαρτάται από τα βάρη και από την συνάρτηση εισόδου – εξόδου (συνάρτηση μεταφοράς) η οποία έχει καθοριστεί για τις μονάδες. Αυτή η συνάρτηση εντάσσεται σε τρεις κατηγορίες :

- Γραμμική : Για την κατηγορία αυτή η δραστηριότητα εξόδου είναι ανάλογη προς τη συνολική σταθμισμένη έξοδο.



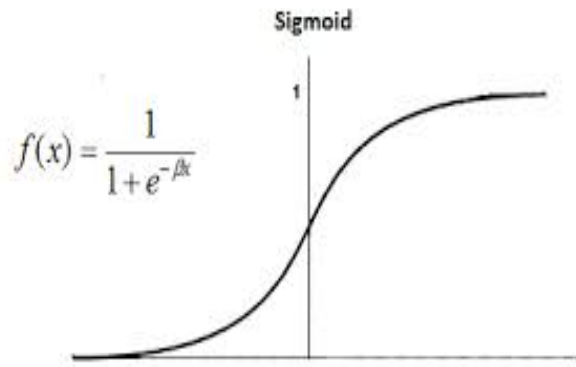
Σχήμα 5 : Γραμμική συνάρτηση

- Κατώτατο όριο (Threshold) : Η έξοδος είναι σε ένα από τα δύο επίπεδα, ανάλογα με το αν η συνολική είσοδος είναι μεγαλύτερη ή μικρότερη από κάποιο όριο.



Σχήμα 6 : Threshold συνάρτηση

- Σιγμοειδής : Η έξοδος μεταβάλλεται συνεχώς αλλά όχι γραμμικά. Οι σιγμοειδής μονάδες φέρουν μεγαλύτερη ομοιότητα ανάμεσα στους πραγματικούς νευρώνες παρά οι άλλες δύο κατηγορίες.



Σχήμα 7 : Σιγμοειδής συνάρτηση

1.5 ΕΦΑΡΜΟΓΕΣ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ

Καταρχάς, αξίζει να αναφέρουμε ότι τα ΤΝΔ χρησιμοποιούνται ήδη σε μεγάλο βαθμό. Κάποιες από τις εφαρμογές του στην καθημερινότητά μας περιλαμβάνουν την εφαρμογή αναγνώρισης τραγουδιών Shazam η οποία χρησιμοποιεί ΤΝΔ όπως και αρκετοί παροχείς email για να ξεχωρίζουν τα ανεπιθύμητα μηνύματα (spam detection). Τα νευρωνικά δίκτυα είναι εφαρμόσιμα σχεδόν σε κάθε κατάσταση στην οποία ισχύει μια σχέση μεταξύ μεταβλητών πρόβλεψης (ανεξάρτητες, εισροές) και προβλεπόμενες μεταβλητές (εξαρτημένες, εκροές), ακόμα και όταν αυτή η σχέση είναι πολύ περίπλοκη για να αποδοθεί με τους συνηθισμένους όρους της «συσχέτισης» ή των «διαφόρων ομάδων». (Χρήστος Στεργίου και Δημήτριος Σιγανός, 2010)[9] Ενδεικτικά αντιπροσωπευτικά παραδείγματα προβλημάτων στα οποία η ανάλυση των νευρωνικών δικτύων έχει εφαρμοστεί με επιτυχία είναι τα εξής:

- Ιατρική διάγνωση: Ένα ευρύ φάσμα ιατρικά συσχετιζόμενων ενδείξεων, όπως ο συνδυασμός της καρδιακής συχνότητας, τα επίπεδα των διαφόρων ουσιών στο αίμα, ο ρυθμός της αναπνοής μπορούν να παρακολουθηθούν. Η εκδήλωση μιας συγκεκριμένης ιατρικής κατάστασης, γίνεται να συσχετιστεί με ένα πολύπλοκο συνδυασμό μεταβολών σε ένα υποσύνολο μεταβλητών που παρακολουθούνται.

Τα νευρωνικά δίκτυα έχουν χρησιμοποιηθεί για την αναγνώριση αυτού του προτύπου πρόβλεψης, ώστε να χορηγηθεί η κατάλληλη θεραπεία.

- Χρηματιστηριακές προβλέψεις: Οι διακυμάνσεις των τιμών των μετοχών και των χρηματιστηριακών δεικτών είναι ακόμα ένα παράδειγμα ενός πολύπλοκου, πολυδιάστατου, αλλά και σε ορισμένες περιπτώσεις εν μέρει ντετερμινιστικού φαινομένου. Τα νευρωνικά δίκτυα χρησιμοποιούνται από πολλούς τεχνικούς αναλυτές, ώστε να κάνουν προβλέψεις σχετικά με τις τιμές των μετοχών, βασιζόμενοι σε ένα μεγάλο αριθμό παραγόντων, όπως δηλαδή, τις προηγούμενες επιδόσεις άλλων αποθεμάτων και διαφόρων οικονομικών δεικτών.
- Πιστωτική ανάθεση: Μια ποικιλία από κομμάτια πληροφοριών, τα οποία είναι συνήθως γνωστά για ένα απαιτούμενο δάνειο. Για παράδειγμα, η ηλικία του αιτούντος, η εκπαίδευση, το επάγγελμα και πολλά άλλα στοιχεία που μπορεί να είναι διαθέσιμα. Μετά την εκπαίδευση ενός νευρωνικού δικτύου σε ιστορικά δεδομένα η ανάλυση μπορεί να εκτοπίσει τα πιο κατάλληλα και σχετικά χαρακτηριστικά και να τα χρησιμοποιήσει για την ταξινόμηση των αιτούντων ως χαμηλού ή υψηλού κινδύνου.
- Παρακολούθηση της κατάστασης των μηχανημάτων: Τα νευρωνικά δίκτυα μπορούν να συμβάλλουν στη μείωση του κόστους με την εξασφάλιση της πρόσθετης εμπειρογνομosύνης για τον προγραμματισμό προληπτικής συντήρησης των μηχανημάτων. Ένα νευρωνικό δίκτυο, λοιπόν, μπορεί να εκπαιδευτεί με τέτοιο τρόπο, ώστε να διακρίνει από τους ήχους τους οποίους παράγει μια μηχανή είτε αν εκτελεί κανονικά τις λειτουργίες της, είτε βρίσκεται στα πρόθυρα εμφάνισης οποιασδήποτε δυσλειτουργίας. Μετά από αυτήν την περίοδο εκπαιδευτικής κατάρτισης, η εμπειρία του ίδιου δικτύου είναι δυνατό να χρησιμοποιηθεί με σκοπό την προειδοποίηση ενός τεχνικού για κάποια επικείμενη βλάβη προτού συμβεί και ενδεχομένως προκαλέσει πολυδάπανες και απρόβλεπτες χρονικές καθυστερήσεις.
- Συστήματα διαχείρισης κινητήρα: Τα νευρωνικά δίκτυα έχουν χρησιμοποιηθεί για την ανάλυση των εισροών που δέχονται οι αισθητήρες ενός κινητήρα. Το νευρωνικό δίκτυο ελέγχει μια ποικιλία παραμέτρων με τις οποίες λειτουργεί ο κινητήρας, προκειμένου να επιτευχθεί ένας συγκεκριμένος στόχος. Για παράδειγμα, το δίκτυο αυτό επιχειρεί την ελαχιστοποίηση της κατανάλωσης των καυσίμων.

Τέλος τα πλεονεκτήματα των νευρωνικών δικτύων πολλά:

- Η μάθησης τους είναι προσαρμοσμένη : Το τεχνητό νευρωνικό δίκτυο εκτελεί εργασίες βασισμένες σε δεδομένα που έχουν παραχθεί κατά την εκπαίδευση του.
- Αυτοοργάνωση : Μία ομάδα εξόδου εκπαιδεύεται για να ανταποκρίνεται σε ομάδες προτύπων που υπάρχουν στην είσοδο. Δεν υπάρχουν εξαρχής σύνολα κατηγοριών καθορισμένα στα οποία θα ταξινομηθούν τα πρότυπα αλλά αναπτύσσει την δική του αναπαράσταση στα ερεθίσματα εισόδου.
- Λειτουργία σε πραγματικό χρόνο
- Δυνατότητα γενίκευσης
- Οι υπολογισμοί ενός Τεχνητού Νευρωνικού Δικτύου μπορούν να εκτελεστούν παράλληλα
- Δυνατότητα διόρθωσης σφαλμάτων : Μερική καταστροφή του δικτύου έχει ως αποτέλεσμα τη μείωση ταχύτητας του επεξεργαστή.(Wikipedia,2014)[11]

ΚΕΦΑΛΑΙΟ 2

Θεωρία της βελτιστοποίησης

2.1 ΕΙΣΑΓΩΓΗ ΣΤΗ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ

Με τον όρο βελτιστοποίηση αναφερόμαστε στην ελαχιστοποίηση ή στη μεγιστοποίηση μιας συνάρτησης. Για μεγαλύτερη ακρίβεια , βελτιστοποίηση είναι η εύρεση τιμών μερικών παραμέτρων για τις οποίες η τιμή μιας συνάρτησης ελαχιστοποιείται ή μεγιστοποιείται. Οι παράμετροι για τις οποίες μιλήσαμε παραπάνω είναι οι γραμμικώς ανεξάρτητες μεταβλητές εισόδου που ορίζουν την τιμή της συνάρτησης. Είναι βέβαια αυτονόητο ότι η ελαχιστοποίηση μιας συνάρτησης f ισοδυναμεί με τη μεγιστοποίηση της συνάρτησης $-f$. Έτσι, η απάντηση στο ερώτημα «μεγιστοποίηση ή ελαχιστοποίηση» εξαρτάται από τη μορφή της συνάρτησης. Η βελτιστοποίηση συναρτήσεων αποτελεί ένα μεγάλο κεφάλαιο των σύγχρονων μαθηματικών και αναλύεται σε πολλές επιμέρους υποκατηγορίες ανάλογα με τις ιδιαιτερότητες της αντικειμενικής συνάρτησης και της ύπαρξης ή μη περιορισμών στις μεταβλητές της. Ειδικότερα, έχει επικρατήσει ο παρακάτω διαχωρισμός:

- Βελτιστοποίηση χωρίς περιορισμούς (unconstrained optimization):

- Η αντικειμενική συνάρτηση είναι δύο φορές συνεχώς διαφορίσιμη αλλά δεν είναι γνωστές οι κλίσεις.
 - Η αντικειμενική συνάρτηση είναι Lipschitz συνεχής.
 - Η κλίση της αντικειμενικής συνάρτησης είναι γνωστή ή μπορεί να υπολογιστεί με πεπερασμένες διαφορές.
 - Η αντικειμενική συνάρτηση είναι κυρτή (convex) και έχουμε μεγάλο πλήθος μεταβλητών.
 - Μεγάλο πλήθος μεταβλητών και ισχυρά μη γραμμική αντικειμενική συνάρτηση.
 - Οι τιμές της αντικειμενικής συνάρτησης δεν είναι γνωστές ακριβώς αλλά περιέχουν θόρυβο.
 - Οι τιμές της αντικειμενικής συνάρτησης έχουν μεγάλο κόστος για τον υπολογισμό τους.
- Βελτιστοποίηση με περιορισμούς (constrained optimization). Η αντικειμενική συνάρτηση είναι η f ενώ έχουμε ορισμένους περιορισμούς ισότητας g και ορισμένους περιορισμούς ανισότητας h . Η μορφή των f, g και h καθορίζει τις υποκατηγορίες. (Ανδρουλάκης, 2007)[1]
 - f, g και h γραμμικές συναρτήσεις (γραμμικός προγραμματισμός).
 - f τετραγωνική κυρτή ή μη κυρτή και g και h γραμμικές.
 - Ημιορισμένες και κωνικές συναρτήσεις
 - Γεωμετρικός προγραμματισμός
 - f, g και h γενικά μη γραμμικές συναρτήσεις.
 - Οι περιορισμοί περιλαμβάνουν μόνο εκφράσεις για τον περιορισμό σε διάστημα στις παραμέτρους.
 - Οι περιορισμοί είναι μόνο γραμμικοί.
 - Γενικές συναρτήσεις αλλά κυρτές.
 - Γενική αντικειμενική συνάρτηση με πολλούς περιορισμούς.
 - Γενική αντικειμενική συνάρτηση αλλά με αραιό πίνακα περιορισμών.
 - Γενική αντικειμενική συνάρτηση αλλά με περιορισμούς ισορροπίας (equilibrium constraints)
 - Η αντικειμενική συνάρτηση είναι μη ομαλή (nonsmooth).

- Ακέραιος προγραμματισμός
- Προγραμματισμός δικτύων
- Η αντικειμενική συνάρτηση είναι ισχυρά μη γραμμική.

Δεν θα μπορούσαμε να παραλείψουμε μια μικρή αναφορά στους γενετικούς. Οι αλγόριθμοι αυτοί στηρίζονται στην αρχής της εξέλιξης των ειδών που αναπτύχθηκε από τον Δαρβίνο. Η θεωρία αυτή προκάλεσε πολλές αντιδράσεις καθώς ερχόταν σε σύγκρουση με τις αντιλήψεις εκείνης της εποχής και ιδιαίτερα τις θρησκευτικές πεποιθήσεις περί προέλευσης των ειδών.

Αν και η θεωρία αυτή χρησιμοποιείται σήμερα για να δώσει απαντήσεις σε πολλά ζητήματα και παρά την τεχνολογική εξέλιξη των τελευταίων δεκαετιών, υπάρχουν ακόμα άνθρωποι που εναντιώνονται σε αυτήν και πιστεύουν πως η ζωή όπως την ξέρουμε σήμερα προήλθε από θεϊκές παρεμβάσεις και όχι σε μια φυσιολογική εξέλιξη.

2.2. ΕΙΣΑΓΩΓΗ ΣΤΗ ΜΕΘΟΔΟ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΣΜΗΝΟΥΣ ΣΩΜΑΤΙΔΙΩΝ

Η μέθοδος αυτή στηρίζεται στην κίνηση ενός σμήνους. Τα έντομα που ζούνε σε αποικίες όπως μυρμήγκια μέλισσες κλπ έκαναν τον άνθρωπο να αναζητήσει τον τρόπο σκέψης τους. Κάθε έντομο σε μία αποικία φαίνεται να έχει τη δική του διάταξη αλλά σαν σύνολο έχουν μια υψηλή οργάνωση. Οι επιστήμονες που μελέτησαν τον τρόπο αυτό ανακάλυψαν ότι η συνεργασία μεταξύ τους είναι αυτοοργανωμένη. Οι αλληλεπιδράσεις που έχουν τα έντομα μεταξύ τους μπορεί να είναι απλές όλες μαζί όμως μπορούνε να δώσουνε λύση σε πολλά και δύσκολα προβλήματα όπως για παράδειγμα να βρεθεί ο συντομότερος δρόμος σε διαδρομές προς την τροφή τους. Η συμπεριφορά αυτή η ομαδική που ενέδειξαν τα έντομα είναι η λεγόμενη <<νοημοσύνη σμήνους>>. (Δρόσος Χρήστος,2010)[2]

Στην επιστήμη των υπολογιστών η μέθοδος βελτιστοποίησης σμήνους είναι μία υπολογιστική μέθοδος που βελτιστοποιεί ένα πρόβλημα μέσα από επαναλήψεις προσπαθώντας να βελτιστοποιήσει μία υποψήφια λύση σε σχέση με ένα συγκεκριμένο μέτρο ποιότητας. Ο αλγόριθμος αυτός βελτιστοποιεί το πρόβλημα από τον πληθυσμό των υποψήφιας λύσεων λεγόμενα και ως σωματίδια. Τα σωματίδια αυτά κινούνται σε ένα διαστατικό ερευνητικό διάστημα. Κάθε σωματίδιο έχει οριστεί

σε μία θέση x_i με ταχύτητα u_i . Επίσης κάθε σωματίδιο επηρεάζεται από την τοπικά καλύτερη γνωστή θέση, αλλά επίσης τα σωματίδια προσανατολίζονται ανάμεσα στις καλύτερες θέσεις στην αναζήτηση του χώρου οι οποίες ενημερώνονται ως οι καλύτερες θέσεις που βρίσκονται από τα άλλα σωματίδια. Η κίνηση αυτή οδηγεί τα σωματίδια στην καλύτερη θέση. (όπως αναφέρεται στο Ζαχαρία Δ. Ζάχαρη, 2011) [10].

Στην ορολογία της ,κάθε μεμονωμένη οντότητα (individual) του σμήνους καλείται σωματίδιο (particle). Ο αριθμός S των σωματιδίων που συνθέτουν το σμήνος ονομάζεται μέγεθος πληθυσμού (population size). Όλα τα σωματίδια ενεργούν κατά τον ίδιο τρόπο, όπως άλλωστε και οι μέλισσες, δηλαδή κινούνται μέσα στο χώρο αναζήτησης και καθορίζουν την ταχύτητά τους σύμφωνα με τις καλύτερες θέσεις που έχουν ήδη βρεθεί είτε από τα ίδια είτε από τους γείτονές τους, προσπαθώντας να βρουν μια ακόμα καλύτερη θέση. (I. Tsoulos, 2008) [4].

Κάθε σωματίδιο αντιμετωπίζεται ως σημείο ενός χώρου N διαστάσεων. Η θέση του i -στου σωματιδίου ($i=1, \dots, S$) συμβολίζεται ως $x_i = [x_{i1} \dots x_{in} \dots x_{iN}]^T$ όπου x_{in} ($n=1, \dots, N$) είναι οι συντεταγμένες της θέσης. Κάθε συντεταγμένη x_{in} μπορεί να περιορίζεται στην

αντίστοιχη (n -στη) διάσταση μεταξύ ενός ανώτερου ορίου U_n και ενός κατώτερου ορίου L_n , έτσι ώστε $n \text{ in } n \text{ } L_n \leq x_{in} \leq U_n$ ($n=1, \dots, N$). Η επίδοση κάθε σωματιδίου μετρείται σύμφωνα με μία προκαθορισμένη μαθηματική συνάρτηση F η οποία ονομάζεται αντικειμενική συνάρτηση και η οποία σχετίζεται με το πρόβλημα του οποίου αναζητείται η λύση. Η τιμή της αντικειμενικής συνάρτησης εξαρτάται από τις συντεταγμένες της θέσης του σωματιδίου. Συγκεκριμένα, η θέση του σωματιδίου θεωρείται ότι βελτιώνεται καθώς ελαττώνεται η τιμή της αντικειμενικής συνάρτησης. Η προηγούμενη καλύτερη θέση (p -best position) του i -στου σωματιδίου είναι η μέχρι στιγμής καλύτερη θέση στην οποία βρέθηκε το σωματίδιο αυτό. Τα σμήνη σωματιδίων έχουν μελετηθεί ως προς δύο πρότυπα γειτονιάς, που ονομάζονται g -best (global best) και l -best (local best). Στο πρότυπο g -best, κάθε σωματίδιο λαμβάνει υπόψη την καλύτερη θέση που έχει βρεθεί απ' όλα τα σωματίδια του σμήνους. Αυτή η θέση ονομάζεται θέση g -best.

Τα βασικά βήματα ενός γενικού pso αλγορίθμου παρουσιάζονται παρακάτω:

Στο βήμα έναρξης ο αλγόριθμος :

- Θέτει τον αριθμό μορίων

- Ξεκινά το μετρητή επανάληψης
- Αναθέτει τις αρχικές θέσεις (x_i) των μορίων και τις ταχύτητες τους (v_i) με ομοιόμορφα διανεμημένους τυχαίους αριθμούς.

Το vector P_i κρατά την καλύτερη επισκεπτόμενη θέση (αυτά με την χαμηλότερη λειτουργική αξία) για το μόριο I και το vector P_{best} είναι το καλύτερο ανάμεσα (P_1, P_2, \dots, P_n). Στο βήμα τερματισμού ο αλγόριθμος ελέγχει κάποια προκαθορισμένα κριτήρια όπως το μέγιστο αριθμό επαναλήψεων ($k \geq k_{max}$) ή πόσο κοντά είναι οι καλύτερες ή χειρότερες λειτουργικές αξίες των μορίων ($|f_{max} - f_{min}| \leq \epsilon, \epsilon > 0$) ή κάποιους άλλους κανόνες παύσης. Στο βήμα 2 η κύρια επανάληψη του αλγορίθμου πραγματοποιείται για κάθε μόριο:

- Η ταχύτητα αναβαθμίζεται
- Η τρέχουσα θέση τροποποιείται ως λειτουργία της σχετικής του ταχύτητας
- Η κατάσταση αξιολογείται
- Η καλύτερη θέση P_i αναβαθμίζεται αν βρεθεί μία καλύτερη θέση.

Μετά τη κύρια επανάληψη η καλύτερη θέση ανάμεσα στο P_i ανατίθεται στο P_{best} . Ο πιο συνηθής μηχανισμός αναβάθμισης για την τρέχουσα θέση ενός μορίου δίνεται από: $x_{i,j} = x_{i,j} + u_{i,j}$ όπου $U_{i,j} = w_{i,j} + r_1 * c_1 * (P_{ij} - x_{ij}) + r_2 * c_2 * (P_{bestj} - x_{ij})$.

Η παράμετρος j δηλώνει το j_{th} στοιχείο του vector όπου $j \in \{1, \dots, n\}$. Οι παράμετροι r_1 και r_2 είναι τυχαίοι αριθμοί στο $[0, 1]$ και οι σταθερές C_1 και C_2 δηλώνουν τις γνωστικές παραμέτρους. Οι αξίες για τις C_1 και C_2 είναι στο διάστημα $[1, 2]$. Η παράμετρος w ονομάζεται αδράνεια με $0 \leq w \leq 1$. Η αδράνεια ακολουθείται από τον τύπο: $w = w_{max} - k/k_{max} * (w_{max} - w_{min})$ όπου k_{max} είναι ο μέγιστος επιτρεπόμενος αριθμός αδρανειών. Τα w_{max} και w_{min} ορίζονται από τον χρήστη. Κοινές αξίες γι' αυτές τις παραμέτρους είναι $w_{min} = 0.4$ και $w_{max} = 0.9$. Παρακάτω δίνεται ο γενικός αλγόριθμος PSO:

1. Initialization.

(a) Set $k = 0$ (iteration counter).

(b) Specify the parameter m (number of particles)

(c) Randomly initialize the positions of the m particles x_1, x_2, \dots, x_m ,

with $x_i \in S \subset \mathbb{R}^n$

(d) Randomly initialize the velocities of the m particles u_1, u_2, \dots, u_m ,

with $u_i \in S \subset \mathbb{R}^n$

- (e) For $i = 1, \dots, m$ do $p_i = x_i$.
- (f) Set $p_{best} = \arg \min_{i \in \{1, \dots, m\}} f(x_i)$
- 2. Termination Check. If the termination criteria hold stop. The final outcome of the algorithm will be p_{best}
- 3. For $i = 1, \dots, m$ Do
 - (a) Update the velocity u_i
 - (b) Update the position x_i as a function of u_i , p_i and p_{best}
 - (c) Evaluate the fitness of the particle i , $f(x_i)$
 - (d) If $f(x_i) < f(p_i)$ then $p_i = x_i$
- 4. End For
- 5. Set $p_{best} = \arg \min_{i \in \{1, \dots, m\}} f(x_i)$
- 6. Set $k = k + 1$.
- 7. Go to Step 2

2.2.1 ΤΡΟΠΟΠΟΙΗΣΕΙΣ ΑΛΓΟΡΙΘΜΩΝ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΣΜΗΝΟΥΣ ΣΩΜΑΤΙΔΙΩΝ

Η πρώτη και πιθανόν η πιο σημαντική τροποποίηση είναι ένας νέος κανόνας παύσης βασισμένος σε ασύμπτωτες θεωρίες. Θεωρήστε την συνάρτηση Rastrigin που δίνεται από τον τύπο : $f(x) = x^2 + x^2 - \cos(18x) - (18x)$.

Η λειτουργία έχει 49 τοπικά ελάχιστα στο $[-1,1]^2$ και το ολικό ελάχιστο εντοπίζεται στο $[0,0]$. Ας αφήσουμε ένα αλγόριθμο PSO για την τοποθεσία του παγκοσμίου ελάχιστου της παραπάνω λειτουργίας με 100 μόρια. Ο αλγόριθμος τερματίζει όταν $k \geq 200$ όπου 200 είναι ο μέγιστος αριθμός επαναλήψεων που επιτρέπεται. Η διαφορετικότητα της λειτουργικής αξίας του P_{best} 'συνωμοτεί' ενάντια στον αριθμό επαναλήψεων k . Όπως μπορούμε να δούμε η διαφορετικότητα μειώνεται προς το μηδέν. Παρ' όλο αυτά ο αλγόριθμος

ανακαλύπτει το ολικό ελάχιστο στην επανάληψη και υπόλοιπες επαναλήψεις ξοδεύονται για να εκπληρώσουν τα κριτήρια τερματισμού. Ο νέος βασικός PSO αλγόριθμος ου χρησιμοποιεί το προτεινόμενο κανόνα παύσης είναι ο εξής. Σε κάθε επανάληψη ο αλγόριθμος υπολογίζει τη διαφορετικότητα (μεταβλητή v) της καλύτερα βρεθείς αξίας (χρησιμοποιώντας της βοηθητικές μεταβλητές v_1 και v_2) και σταματά όταν $v \leq s$ όπου s είναι η διαφορά της καλύτερης αξίας όταν η νέα καλύτερη

αξία ανακαλύφθηκε τελευταία. Ο λόγος για αυτό τον κανόνα παύσης είναι ότι, όταν δεν υπάρχει καθόλου πρόοδος για ένα αλγόριθμο επαναλήψεων, θα μπορούσαμε να πιστέψουμε ότι έχουμε βρει το ολικό ελάχιστο και έτσι πρέπει να σταματήσουμε την επανάληψη.

Η δεύτερη προτεινόμενη τροποποίηση είναι ο λεγόμενος έλεγχος ομοιότητας. Στην τροποποίηση αυτή σε κάθε επανάληψη η λειτουργία σκοπιμότητας αξιολογείται στο σημείο x_i παρόλο που το σημείο έχει παραμείνει άθικτο κατά τη διάρκεια της τελευταίας επανάληψης. Αυτές οι αχρείαστες αξιολογήσεις λειτουργίας μπορούν να αποφευχθούν ελέγχοντας απλά το σημείο x_i αλλαγές όπως αξιολογήστε το $f(x_i)$ αν και μόνο αν $[x_i^{(k+1)} - x_i^{(k)}] > e$ όπου $x_i^{(k)}$ είναι το σημείο x_i στην επανάληψη k και e ένας μικρός θετικός αριθμός όπως $e=10^{-5}$.

Η τελική τροποποίηση που προτείνουμε είναι ένα νέο χειριστή του πληθυσμού των μορίων : σε κάθε επανάληψη αποφασίζουμε να εφαρμόσουμε μια διαδικασία τοπικής έρευνας στα μέλη του πληθυσμού με πιθανότητα. Για να αποφύγουμε να παγιδευτεί ο αλγόριθμος σε κάποια τοπικά ελάχιστα και να αποφύγουμε αχρείαστες κλήσεις συναρτήσεων και θέτουμε την πιθανότητα στο $P1= 1/m$ όπου m είναι ο αριθμός μορίων. (I.Tsoulos,2008)[4].

Εκτός από αυτές τις τρεις τροποποιήσεις θα αναφερθούνε πολύ σύντομα και κάποιες ακόμα :

- Η Δυναδική βελτιστοποίηση: αναπτύχθηκε από τους Kennedy και Eberhart με σκοπό την αντιμετώπιση προβλημάτων που είναι από την φύση τους δυαδικά.
- Βελτιστοποίηση με ασαφές βάρος αδράνειας: Ο Shi και ο Eberhart πρότειναν μέθοδο μεταβολής του βάρους αδράνειας με την χρήση ενός ασαφούς ελεγκτή.
- Βελτιστοποίηση με αυξημένη ποικιλότητα: Χρησιμοποίηση τελεστών και μεθόδων που αυξάνουν την ποικιλότητα των λύσεων στον πληθυσμό.

ΚΕΦΑΛΑΙΟ 3

ΕΚΠΑΙΔΕΥΣΗ ΤΕΧΝΙΤΩΝ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ ΜΕ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΜΗΝΟΥΣ ΣΩΜΑΤΙΔΙΩΝ

3.1 ΕΙΣΑΓΩΓΗ

Στα κεφάλαια που αναπτύξαμε παραπάνω μιλήσαμε θεωρητικά για το τι είναι νευρωνικά δίκτυα και τη μέθοδος βελτιστοποίησης σμήνους σωματιδίων. Στο κεφάλαιο γίνεται η αναλυτική περιγραφή της μεθόδου που εφαρμόσαμε ώστε να εκπαιδεύσουμε ένα νευρωνικό δίκτυο με την χρήση βελτιστοποίησης σμήνους σωματιδίων . Αρχικά έγινε μια ανάλυση του προβλήματος για να διαπιστώσουμε ποια από τις τρεις μεθόδους θα ήταν η κατάλληλη να χρησιμοποιηθεί ώστε να έρθει η επίλυση του προβλήματος πιο απλά και αποτελεσματικά και αυτή ήταν ο γενικός αλγόριθμος PSO. Κατά την διάρκεια της ανάπτυξης δεν χρησιμοποιήθηκε κάποια βιβλιοθήκη αλλά όλες οι μέθοδοι που χρησιμοποιήθηκαν αναπτύχθηκαν εξ αρχής. Εφόσον έγινε η ανάλυση του προβλήματος καταλήξαμε στο προγραμματιστικό περιβάλλον που θα χρησιμοποιήσουμε. Η προγραμματιστική επίλυση της παραπάνω μεθόδου έγινε με την χρήση προγραμματιστικού περιβάλλοντος MATLAB.

Η mat lab είναι μια υψηλού επιπέδου γλώσσα προγραμματισμού 4ης γενιάς η οποία αποθηκεύει και κάνει πράξεις με άλγεβρα μητρών. Χρησιμοποιείται κυρίως για επίλυση μαθηματικών προβλημάτων αλλά μπορεί να χρησιμοποιηθεί και για προγραμματισμό περιλαμβάνοντας εντολές από την C++ όπως την while την if κλπ. Μπορεί να χρησιμοποιήσει γραφικές παραστάσεις, ιστογράμματα, εμδαδογράμματα, ραβδοδιαγράμματα.

Το περιβάλλον της mat lab περιλαμβάνει τρία κυρίως παράθυρα. Το παράθυρο εντολών (command window) , το παράθυρο επεξεργασίας (editor) και το παράθυρο σχεδίασης (figure). Το παράθυρο σχεδίασης εμφανίζεται κάθε φορά που ο χρήστης εκτελεί κάποια εντολή σχεδίασης. Στο παράθυρο επεξεργασίας μπορούμε να γράψουμε προγράμματα Mat lab (τα οποία καλούνται M-files) και να τα επεξεργαστούμε. Το παράθυρο επεξεργασίας εμφανίζεται όταν γράψουμε την εντολή edit <όνομα_αρχείου>, οπότε και ανοίγει το παράθυρο επεξεργασίας και προβάλλεται το αρχείο <όνομα_αρχείου>. Τέλος το παράθυρο εντολών είναι το κυρίως παράθυρο της Mat lab, όπου μπορούμε να εισάγουμε εντολές προς εκτέλεση. Η γραμμή εντολών της Mat lab εμφανίζει τους χαρακτήρες >> όταν ο χρήστης μπορεί να εισάγει εντολή προς εκτέλεση.(Wikipedia,2014)[11]

3.2 ΔΟΜΗ ΚΩΔΙΚΑ

Στο κεφάλαιο αυτό θα περιγράψουμε τις συναρτήσεις που χρησιμοποιήσαμε, τι σημαίνουν και τι κάνει η κάθε μια.

1) Γενικός αλγόριθμος PSO:

`function` [apotelesma fitness_val] = pso (n,max_iter, dim)

Είναι η συνάρτηση του γενικού αλγορίθμου PSO με όνομα pso και περιλαμβάνει τα εξής:

INPUT:

- n (integer): Το n μας υποδηλώνει τον αριθμό των μορίων του νευρώνα.
- max_iter (integer): Το max_iter μας υποδηλώνει τον μέγιστο αριθμό επαναλήψεων.
- dim (integer): Το dim μας υποδηλώνει την διάσταση του προβλήματος.

OUTPUT:

- apotelesma fitness_val (double) : Στη μεταβλητή αυτή μας δίνεται ως έξοδος το αποτέλεσμα της συνάρτησης fitness της καταλληλότητας δηλαδή της λειτουργίας του αλγορίθμου.

2) Συνάρτηση εξόδου:

`function` [s1] =ann(x, w)

Είναι η συνάρτηση εξόδου του αλγορίθμου με όνομα ann και περιλαμβάνει :

INPUT:

- x (integer): Το x μας υποδηλώνει την θέση σε έναν δισδιάστατο πίνακα.
- w (float) : Το w μας υποδηλώνει το συναπτικό βάρος του νευρώνα το οποίο μεταβάλλεται συνεχώς αποδυναμώνοντας ή ενδυναμώνοντας τους δεσμούς.

OUTPUT :

- s_1 (float) : μεταβλητή στην οποία αποθηκεύεται η τιμή της εξόδου

3) Συνάρτηση Σφάλματος :

`function [s] =ann_error (w)`

Είναι η συνάρτηση σφάλματος του νευρωνικού δικτύου και περιλαμβάνει :

INPUT :

- w (float): Το w μας υποδηλώνει το βάρος που αποδίδεται στα στοιχεία εισόδου.

OUTPUT:

S (float): Είναι η υπολογισμένη τιμή στο επίπεδο εξόδου του νευρώνα σε κάθε κύκλο.

4) Σιγμοειδής συνάρτηση :

`Function [s] = sigmoid(x):`

Είναι μια μαθηματική συνάρτηση η οποία έχει την μορφή S και είναι μη γραμμική συνάρτηση ενεργοποίησης.

INPUT:

- x (real) : Είναι ένας πραγματικός αριθμός που το θέτει ο χρήστης.

OUTPUT:

- s (real) : είναι η μεταβλητή τιμή της σιγμοειδής συνάρτησης.

5) Κώδικας για την τελική εκπαίδευση του νευρώνα test1.m :

Ο κώδικας ο οποίος καλεί όλες τις συναρτήσεις και εκπαιδεύει το νευρωνικό μας δίκτυο, μέσω αυτού γράφουμε κώδικα για τις γραφικές παραστάσεις όπως ακόμα μετράμε και τον χρόνο που διαρκεί η τελική ανάλυση των αποτελεσμάτων.

METABΛΗΤΕΣ:

- Global x (real): Δεδομένα εκπαίδευσης για είσοδο.
- Global y (real): Δεδομένα εκπαίδευσης για έξοδο.

- Global H (real): Δεδομένα για τα επίπεδα του προβλήματος.

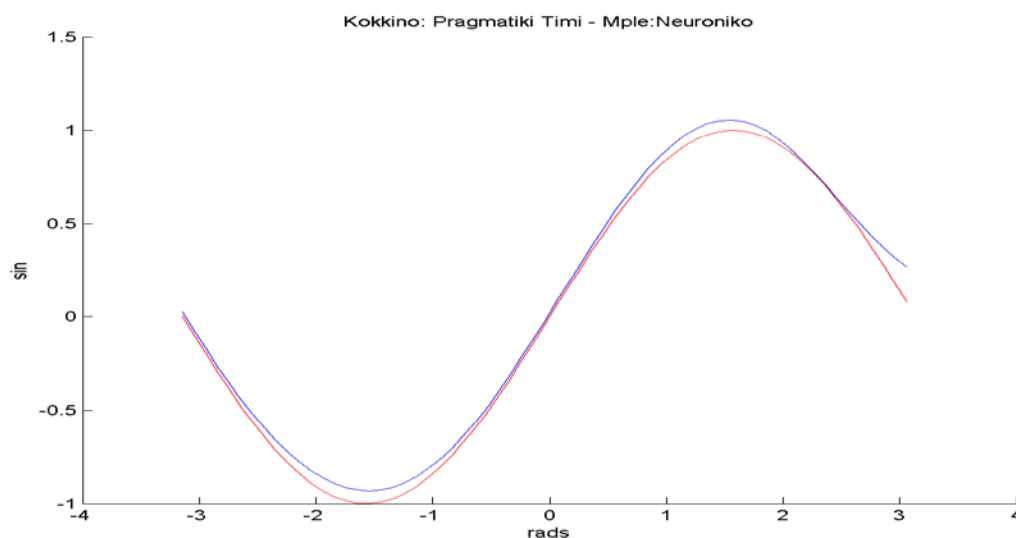
ΠΑΡΑΜΕΤΡΟΙ:

- H (integer): τα επίπεδα του νευρωνικού δικτύου.
- MAX_ITER: οι μέγιστες επαναλήψεις.
- K : ο αριθμός των μορίων.

3.3 Πειραματικά Αποτελέσματα:

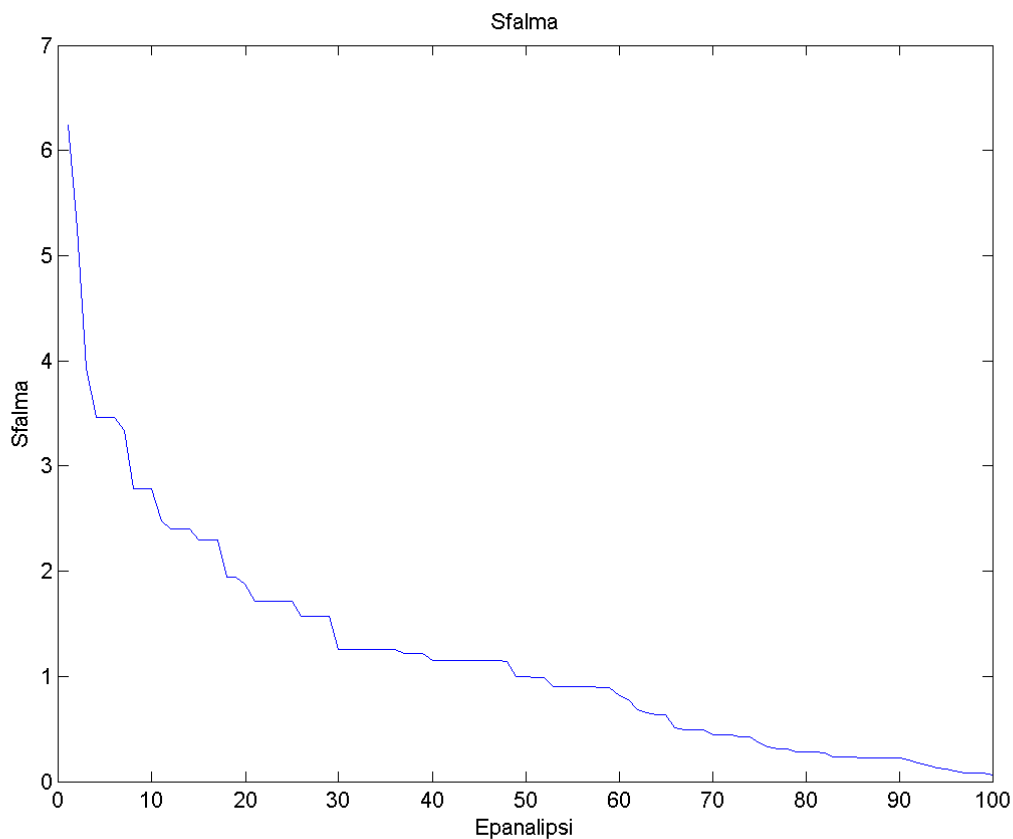
Ο αλγόριθμος PSO εφαρμόστηκε σε ένα υπολογιστή με τα παρακάτω στοιχεία : Intel core I5 στα 3.00 GHZ με μνήμη RAM 4GB λειτουργικού συστήματος Microsoft Windows 7. Όπως προαναφέρθηκε και παραπάνω ο σκοπός δημιουργίας του αλγορίθμου είναι για να «εκπαιδεύσουμε» ένα τεχνητό νευρωνικό δίκτυο. Γι' αυτό δημιουργήσαμε τον γενικό αλγόριθμο PSO, το σφάλμα , την έξοδο του νευρωνικού, τη σιγμοειδή συνάρτηση, το αρχείο test1 και τα αρχεία BK1 και BK2. Σκοπός του προβλήματος μας είναι μέσα από τον υπολογισμό των βαρών να μειώνεται το σφάλμα.

Οι παράμετροι που χρησιμοποιήθηκαν για τον αλγόριθμο ήταν: $c1=0.12$, $c2=1.2$, $w=0.9$, $max_iter=100$, $H=3$, $K=1000$, $d=1$, $n=63$. Εξετάσαμε δύο περιπτώσεις με διαφορετικές τιμές στις παραμέτρους και ο χρόνος εκτέλεσης ήταν της τάξης των 146 δευτερολέπτων. Στην πρώτη περίπτωση θέσαμε σαν τιμές τις παραπάνω και πήραμε το εξής αποτέλεσμα :



Σχήμα 8 : Γραφική παράσταση ημιτόνου για $d=1$ και $n=63$

Όπως μπορούμε να διαπιστώσουμε και από τη γραφική παράσταση οι παράμετροι που θέσαμε συμπίπτουν αρκετά με το νευρωνικό μας δίκτυο όχι τέλεια κάτι που είναι δύσκολο να επιτευχθεί αλλά αρκετά καλά. Επίσης για τις παραπάνω τιμές δημιουργήθηκε και η γραφική παράσταση του σφάλματος όπως φαίνεται και παρακάτω :

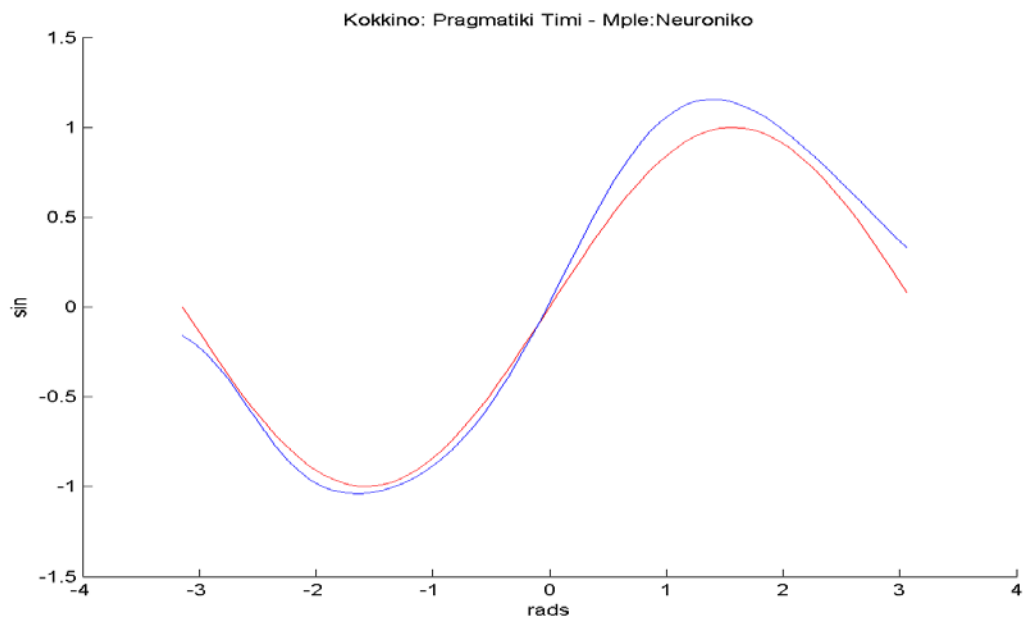


Σχήμα 9: Γραφική παράσταση σφάλματος για d=1 και n=63

Διαπιστώνουμε με τις παραπάνω παραμέτρους ότι το σφάλμα μας με το πέρασμα των επαναλήψεων μειώνεται διαρκώς, αυτό που θέλαμε και εμείς σύμφωνα με την έρευνα που πραγματοποιήσαμε. Σε μία δεύτερη μέτρηση που κάναμε με τις εξής παραμέτρους και διάρκεια 508 δευτερολέπτων βγάλαμε τις παρακάτω γραφικές παραστάσεις.

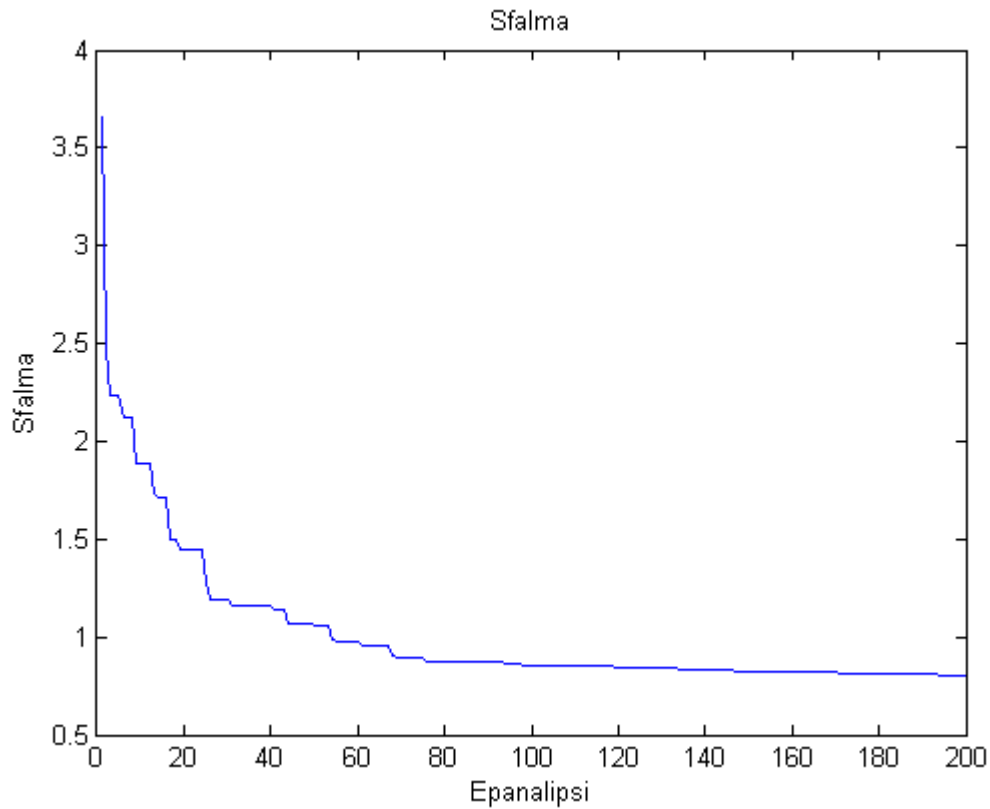
c2	1.2
c1	2
w	0.9

H	2
MAX_ITER	200
K	2000
d	1
n	63



Σχήμα 10: Γραφική παράσταση ημιτόνου

Όπως βλέπουμε από το γράφημα η πραγματική τιμή με την τιμή του νευρωνικού τείνουν να είναι πολύ κοντά οι δύο τους.



Σχήμα 11: Γραφική παράσταση ημιτόνου

Στο παραπάνω γράφημα μπορούμε να διαπιστώσουμε ότι το σφάλμα σταδιακά μειώνεται και με τις νέες παραμέτρους που θέσαμε. Άρα καταλήγουμε στο συμπέρασμα ότι το νευρωνικό μας δίκτυο εκπαιδεύεται σωστά σύμφωνα με όσα μελετήσαμε παραπάνω.

Στη συνέχεια κάναμε πειράματα σε ένα σύνολο δεδομένων και τα αποτελέσματα παραθέτονται στον παρακάτω πίνακα:

ΣΩΜΑΤΙΔΙΑ	ΣΦΑΛΜΑ
100	1.31
150	1.01
200	0.96
300	0.93
500	0.90
1000	0.87

Όπως γίνεται κατανοητό στο παραπάνω πείραμα διαπιστώνεται ότι όσο αυξάνονται τα σωματίδια τόσο μειώνεται το σφάλμα κάτι το οποίο περιμέναμε. Επίσης να αναφερθεί ότι στα 100 σωματίδια το πείραμα διήρκησε 20 sec, τα 150 - 30 sec, τα 200 - 40 sec, τα 300 - 60 sec, τα 500 - 100 sec, τα 1000 - 200 sec.

ΠΑΡΑΡΤΗΜΑ

1) Συνάρτηση σφάλματος:

```
function [s]=ann_error(w)

%GLOBAL VARS
global x;
global y;

M=size(x,1);

s=0;
for i=1:M
    s=s+(ann(x(i,:),w)-y(i))^2;
end
```

2) Συνάρτηση εξόδου:

```
function [s1]=ann(x,w)

%GLOBAL VARS
global H;

s1=0; d =size(x,2);

for i=1:H
    s2=0;
    for j=1:d
        s2=s2+w((d+2)*i-(d+1)+j)*x(j)+w((d+2)*i);
    end
    s2=sigmoid(s2);
    s1=s1+w((d+2)*i-(d+1))*s2;
end
```

3) Συγμοειδής συνάρτηση:

```
function [s] = sigmoid(x)
s=1/(1+exp(-x));
```

4) Αλγόριθμος PSO:

```
function [apotelesma fitness_val] = pso(n,max_iter,dim)
```

```
%n      = number of swarms
%max_iter = maximum iterations
%dim     = problem dimensionality

c2 =1.2;      % PSO parameter C1
c1 =0.12;     % PSO parameter C2
w = 0.9;      % pso momentum or inertia
```

```
%-----%
%  ARXIKOPOIHSH  %
%-----%
```

```
R1 = rand (dim, n);
R2 = rand (dim, n);
current_fitness =0*ones (n,1);
```

```
trexousa_thesi = randn(dim,n);
velocity = .3*randn(dim,n) ;
topika_kaliterh_thesi = trexousa_thesi ;
```

```
for i = 1:n
    current_fitness(i) = ann_error(trexousa_thesi(:,i));
end
```

```
local_best_fitness = current_fitness ;
[global_best_fitness,g] = min(local_best_fitness) ;
```

```
for i=1:n
    kaliterh_thesi(:,i) = topika_kaliterh_thesi(:,g) ;
end
```

```
%-----%
%  VELOCITY UPDATE  %
%-----%
```

```
velocity = w *velocity + c1*(R1.*(topika_kaliterh_thesi-trexousa_thesi)) +
c2*(R2.*(kaliterh_thesi-trexousa_thesi));
```

```
%-----%
%  SWARM UPDATE  %
%-----%
```

```

trexousa_thesi = trexousa_thesi + velocity ;

%-----%
%  ALGORITHMOS  %
%-----%
iter = 0 ; fitness_val = zeros(max_iter,1);
while ( iter < max_iter )
    iter = iter + 1;

    for i = 1:n,
        current_fitness(i) = ann_error(trexousa_thesi(:,i)) ;
    end

    for i = 1 : n
        if current_fitness(i) < local_best_fitness(i)
            local_best_fitness(i) = current_fitness(i);
            topika_kaliterh_thesi(:,i) = trexousa_thesi(:,i) ;
        end
    end

    [trexousa_kaliterh_fitness,g] = min(local_best_fitness);

    fitness_val(iter) = trexousa_kaliterh_fitness;

    if trexousa_kaliterh_fitness < global_best_fitness
        global_best_fitness = trexousa_kaliterh_fitness;

        for i=1:n
            kaliterh_thesi(:,i) = topika_kaliterh_thesi(:,g);
        end

    end

    velocity = w *velocity + c1*(R1.*(topika_kaliterh_thesi-trexousa_thesi)) +
c2*(R2.*(kaliterh_thesi-trexousa_thesi));
    trexousa_thesi = trexousa_thesi + velocity;
end

[Jbest_min,I] = min(current_fitness); % minimum fitness
apotelesma = trexousa_thesi(:,I); % best solution

```

5) Παράδειγμα εκτέλεσης κώδικα:

```

clear all; close all; fclose all; clc;

%GLOBAL VARS

```

```

global x;
global y;
global H;
%READ DATA
%fid = fopen('BK1.train','r');
fid = fopen('BK2.train','r');
d = fscanf(fid, '%d',1); %PROBLEM DIMENSION
M = fscanf(fid, '%d',1); %HOW MANY DATA
%PARAMETRS
NODES = 10*(d+2);

H = NODES/(d+2); %nodes in the hidden layers
MAX_ITER = 100; %max number of iterations
K = [100 150 200 300 500 1000]; %number of swarms

x = zeros(M,d); y = zeros(M,1);

for i=1:M
    line = fscanf(fid,'%f',d+1);
    x(i,:) = line(1:d);
    y(i) = line(d+1);
end
fclose(fid);
Q = (d+2)*H; %DIMENSION OF WEIGHT MATRIX
%
% Call PSO => TRAIN artificial neural network
%
sfalma_belt = zeros(length(K),1);
for k=1:length(K)
    fprintf('PSO...');
    tic;

```



```

[w xxx] = pso(K(k),MAX_ITER,Q); sfalma_belt(k) = xxx(length(xxx));
t = toc;
fprintf('(%0.0f sec) TELOS.\n',t);
end
fid = fopen('sfalma_bel.html','w');
fprintf(fid,'<html<head></head><body><table
border="1"><tr><td>Somatidia</td><td>Sfalma</td></tr>');
for i=1:length(K)
    fprintf(fid,'<tr><td>%d</td><td>%0.2f</td></tr>',K(i),sfalma_belt(i));
end
fprintf(fid,'</table></body></html>');
fclose(fid);
%test ann plot
if d == 1
    h = figure;
    z = zeros(M,1);
    for i=1:M
        z(i) = ann(x(i),w);
    end
    e = abs(z-y);
    Hold on;
    plot(x,y,'r-');
    plot(x,z,'b-');
    title('Kokkino: Pragmatiki Timi - Mple: Neuroniko');
    xlabel('rads');
    ylabel('sin');
    saveas(h,'paradeigma_sin.tiff');
    clf;
    plot(xxx,'-');
    title('Sfalma');
    xlabel('Epanalipsi');
    ylabel('Sfalma');
    saveas(h,'error.tiff');
end

```

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Ανδρουλάκης, Βελτιστοποίηση συναρτήσεων. 2007. http://androulakis.bma.upatras.gr/media/wiki/index.php?title=%CE%92%CE%B5%CE%BB%CF%84%CE%B9%CF%83%CF%84%CE%BF%CF%80%CE%BF%CE%AF%CE%B7%CF%83%CE%B7_%CF%83%CF%85%CE%BD%CE%B1%CF%81%CF%84%CE%AE%CF%83%CE%B5%CF%89%CE%BD&oldid=1774, Αθήνα 2007.

[2] Δρόσος Χρήστος, Η Νοημοσύνη του Σμήνους. 2010. digilib.lib.unipi.gr/dspace/bitstream/unipi/4826/1/Drosos.pdf

[3] Ευστάθιος Ανδριανάκης, Εντοπισμός ατελειών σε επίπεδη πλάκα ομογενούς και σύνθετου υλικού με χρήση νευρωνικών δικτύων και μεθόδων πεπερασμένων στοιχείων, Διπλωματική εργασία, Αθήνα Ιούλιος 2008.

[4] I.Tsoulos, Neural network construction and training using grammatical evolution.2008. http://users.cs.uoi.gr/~itsoulos/publications/nnc_neurocomputing.pdf

[4] I. Tsoulos, Neural network construction and training using grammatical evolution.2008. http://users.cs.uoi.gr/~itsoulos/publications/nnc_neurocomputing.pdf

[6] Κούρους Νικόλαος, Δυναμικός αρθρωτός νευροασαφής ταξινομητής, Διπλωματική Εργασία, Θεσσαλονίκη 2012.

[7] Pauly, Nico. «ΤΟ ΑΝΘΡΩΠΙΝΟ ΝΕΥΡΙΚΟ ΣΥΣΤΗΜΑ ΜΙΑ ΦΙΛΟΣΟΦΙΚΗ ΚΑΙ ΦΥΣΙΟΛΟΓΙΚΗ.» Προ- σεμιναριακό υλικό για την εκπαίδευση στην Νεύρο-Ρεφλεξολογία (NR) και στην Νευροθεραπεία δια χειρισμών(MNT). 2008. <http://www.mnt-nr.com/downloads/Nio%20progreek.pdf>

- [8] Roja, Raul.<< Weighted Network – The Perceptron. >>2010.
<http://page.mi.fuberlin.de/rojas/neural/neural/chapter/k3.pdf>
- [9] Χρήστος Στεργίου και Δημήτριος Σιγανός, Neural Networks.2010.
[http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html#Why use neural networks](http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html#Why_use_neural_networks)
- [10] Ζαχαρία Δ. Ζάχαρη ,Σχεδίαση Βέλτιστων Στοιχειοκεραιών και Μορφοποίηση Δέσμης Ακτινοβολίας με τη Χρήση Τεχνικών Βελτιστοποίησης Σμήνους Σωματιδίων,Διπλωματική Εργασία, Θεσσαλονίκη 2011.
- [11] Wikipedia, Artificial neural network.Διαθέσιμο στο δικτυακό τόπο:
http://en.wikipedia.org/wiki/Artificial_neural_network
- [11] Wikipedia, Artificial neural network.Διαθέσιμο στο δικτυακό τόπο:
http://en.wikipedia.org/wiki/Artificial_neural_network
- [12] <http://jmlr.csail.mit.edu/proceedings/papers/v9/glorot10a/glorot10a.pdf>