Τ.Ε.Ι. ΣΤΕΡΕΑΣ ΕΛΛΑΔΑΣ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ



**Java Fuzzy Cognitive Maps**

**Design App**

**ΙΑΣΩΝ ΚΟΥΤΟΥΛΙΑΣ**

Επιβλέπων

ΔΡ. ΕΛΠΙΝΙΚΗ ΠΑΠΑΓΕΩΡΓΙΟΥ

Περιεχόμενα

1. Εισαγωγή ................................................................................................................................ 3
2. Σκοπός .................................................................................................................................... 4
3. Εννοιολογική ανάλυση ........................................................................................................... 5
   1. Hebbian Learning Algorithms .......................................................................................... 5
      1. Differential ............................................................................................................ 6
      2. Non Linear ............................................................................................................. 7
      3. Data Driven ........................................................................................................... 8
      4. Active .................................................................................................................... 9
      5. Σχόλιο .................................................................................................................. 10
   2. Επιλογή Κατάλληλου Αλγορίθμου ................................................................................. 12
      1. Παράδειγμα 1 ..................................................................................................... 12
      2. Παράδειγμα 2 ..................................................................................................... 13
      3. Παράδειγμα 3 ..................................................................................................... 14
   3. Συμπεράσματα .............................................................................................................. 16
4. GitHub ................................................................................................................................... 17
5. Βιβλιογραφία ........................................................................................................................ 18

**Εισαγωγή**

Το JFCM Design App είναι μια open source εφαρμογή, γραμμένη σε Java8, που αναπτύχθηκε με σκοπό την επίλυση πραγματικών συστημάτων. Τα συστήματα αντιπροσωπεύονται από Νευρωνικά Δίκτυα και ο τρόπος επίλυσης γίνεται με Hebbian μάθηση.

Η κατασκευή του έγινε με βασικό γνώμονα την ευχρηστία, για να μπορεί να χρησιμοποιηθεί και απο μη γνώστες.

Η χρήση του είναι η εξής :

* Μοντελοποίηση του συστήματος σε νευρώνες
* Προσθήκη συσχετίσεων των νευρώνων υπό τη μορφή βαρών
* Ανάθεση αρχικών τιμών στους νευρώνες και στα βάρη
* Ανάθεση θεμιτών αποτελεσμάτων
* Προσδιορισμός παραμέτρων: Inference Rule, Transfer Function, Hebbian Algorithm.

Καθώς το πρόγραμμα εκπαιδεύει τους νευρώνες και τα βάρη, δημιουργείται μια σύγκλιση στους νευρώνες. Ο αλγόριθμος τρέχει μέχρι να φτάσει σε μία ισορροπία, δηλαδή, η μεταβολή των νευρώνων να είναι αμελητέα, καθώς και όταν οι νευρώνες πάρουν τιμές εντός θεμιτών πλαισίων.

**Σκοπός**

Η μάθηση σε ένα νευρωνικό δίκτυο σημαίνει την προσαρμογή των τιμών Αi των εννοιών Cj και των βαρών wij εφαρμόζοντας κάποιο αλγόριθμο με κάποιες προδιαγραφές που αποσκοπούν στην διαμόρφωση των τελικών τους τιμών στην έξοδο (ολοκλήρωση του αλγόριθμου) μετά από κάποιο αριθμό επαναλήψεων που έχει σκοπό οι τιμές των Αi και wj να διαμορφώνονται σαν διαφορετικές καταστάσεις ενός δυναμικού συστήματος. Η διαδικασία αυτή αποτελεί την υπολογιστική μοντελοποίηση του συστήματος.

Οι αρχικές τιμές των εννοιών  του πίνακα Αi0 (με περιορισμούς μέγιστης-ελάχιστης τιμής) και των βαρών wj0 αποτελούν την είσοδο του αλγορίθμου και κατά συνέπεια είναι οι "ευαίσθητες αρχικές συνθήκες" από τις οποίες εξαρτώνται τα τελικά αποτελέσματα. Για τυπικούς λόγους δεν εξετάζουμε το φαινόμενο του επηρεασμού των αποτελεσμάτων από αυτές τις συνθήκες που θεωρούνται δεδομένες.

Τα  δείχνουν χαρακτηριστικά, κύριους παράγοντες ή ιδιότητες του συστήματος που πρόκειται να μοντελοποιηθεί. Oι τιμές των wij δείχνουν την ισχύ(ενέργεια) της τυχαίας Ci που επηρεάζει την Cj και wij[-1, 1]. Η ερμηνεία του προσήμου δείχνει τον τρόπο ή την κατεύθυνση θετική (ομοιόμορφα), αρνητική (ετερόμορφα) του επηρεασμού.

Σε κάθε επανάληψη έχουμε μιά νέα κατάσταση (γνωστική) που όλες μαζί αποτελούν τον "χώρο φάσεων" του συστήματος. Αυτός διαμορφώνεται "τελικά" από την επιλογή του αριθμού επαναλήψεων αποσκοπώντας στην "σύγκλιση" των Αi με στατιστικά κριτήρια εκτίμησης MSE, μέσο τετραγωνικό σφάλμα, και GE, γενική εκτίμηση.   
Τα Ν, MSE και GE αποτελούν τα τεχνικά κριτήρια τερματισμού του αλγορίθμου για τη βέλτιστη μάθηση του νευρωνικού δικτύου.

**Εννοιολογική ανάλυση των γνωστικών αλγορίθμων Hebb**

Η Hebbian μάθηση αποτελεί μια επιβλεπόμενη τεχνική που αρχικά εφαρμόζεται για την κατάρτιση των τεχνητών νευρωνικών δικτύων. Το κύριο χαρακτηριστικό αυτού του κανόνα μάθησης είναι ότι η αλλαγή ενός συναπτικού που υπολογίζεται λαμβάνοντας υπόψη τα προσυναπτικά και μετασυναπτικά σήματα που ρέουν προς την μονάδα επεξεργασίας (νευρώνες) ενός νευρωνικού δικτύου και ορίζεται από τον ακόλουθο τύπο

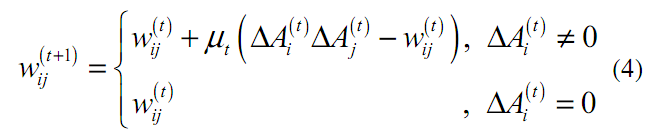


Όπου, xj = προσυναπτικό και yi =μετασυναπτικό σήμα.

Γενικεύοντας ο κανόνας Oja γίνεται :

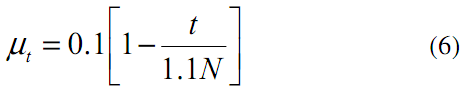
Η παράμετρος η αναφέρεται στον ρυθμό μάθησης και η επιλογή της έχει σκοπό να οδηγήσει στην ισορροπία του συστήματος σε κάθε αλγόριθμο.

**DHL: Differencial Hebbian Learning**



όπου 

Ο αναδρομικός αυτός τύπος έχει τα εξής χαρακτηριστικά :

1. Οι τιμές των βαρών διαμορφώνονται από τις "μεταβολές" των και όχι από τις ίδιες τις τιμές των Αi.(διαφορικός με το χρόνο τυπικά 1).Η μορφή της συνάρτησης είναι γραμμική που χαρακτηρίζεται από το γινόμενο και τη γραμμική παρουσία των βαρών.
2. H συνάρτηση της εξίσωσης (4) δίνεται με βάση το διαχωρισμό των τιμών  σε δύο κατηγορίες, μεταβολών, διαφορετικές από 0 και ίσες με 0. Αυτό δίνει ένα "χαρακτηριστικό" χαρακτήρα στσχέτισης των τιμών του υπολογισμού, ο οποίος δεν είναι πάντα εξαιρετικά ακριβής αφού δεν ξεχωρίζει τις τιμές και σε άλλες κατηγορίες που θα απέδιδε μεγαλύτερη ακρίβεια στους συσχετισμούς των κόμβων μέσω της ποικίλης προσαρμογής των βαρών τους.
3. Είναι μονοπαραμετρικός με τη σταθερά μt να δίνεται από τον τύπο  και να εξαρτάται από το ρυθμό μάθησης (δεύτερη παράγωγος) ο οποίος ελαττώνεται-κυρτώνει τη συνάρτηση με την πρόοδο (αύξηση του Ν) των επαναλήψεων και t την παρούσα επανάληψη (Ν ο μέγιστος αριθμός), εξασφαλίζοντας θετικές τιμές της παραμέτρου. Αυτό σημαίνει από μαθηματική άποψη γι΄αυτό τον αλγόριθμο ότι η σταθερά μt εξασφαλίζει τη σύγκλιση των τιμών του αλγόριθμου μετά από Ν παναλήψεις (αυτό θα φανεί στα διαγράμματα).

Η στατιστική εκτίμηση του αλγόριθμου όπως θα δούμε είναι εξαιρετικά ατελής αφού στα παραδείγματα τα στατιστικά μεγέθη NSE και GE δεν ικανοποιούν τους κατάλληλους περιορισμούς.

**NHL: Non Linear Hebbian Learning**

Προτάθηκε από την Παπαγεωργίου και είναι κοντά στον Οja με το μονοπαραμετρικό τύπο



όπου η είναι η ρυθμός μάθησης έχοντας ένα παρόμοιο ρόλο με τον προηγούμενο στο βαθμό σύγκλισης.

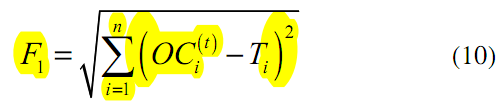
Μιά βελτίωση προτάθηκε με τον τύπο



όπου το sgn(.) είναι η συνάρτηση προσήμου και με τελική μορφή με τη χρήση της παραμέτρου καθυστέρησης γ



Τα χαρακτηριστικά τους είναι τα εξής :

1. Οι εξισώσεις (7) (8), είναι μονοπαραμετρικές αναδρομικές συναρτήσεις με άμεση εξάρτηση των τιμών των βαρών από τα Αi , Aj μεμονωμένα (άρα μη γραμμικά) και τα προηγούμενα βάρη, ενώ η (9) είναι διπαραμετρική δυσκολεύοντας τη σύγκλιση με την αύξηση των ελεύθερων παραμέτρων.
2. Τα κριτήρια σύγκλισης έχουν τη μορφή   Όπου είναι η έξοδος στην t επανάληψη

Ο στόχος του αλγόριθμου είναι να βρεί τη συσχέτιση των βαρών που θα αποδώσει την ελαχιστοποίηση των F με σκοπό την σύγκλιση του συστήματος σε ένα σταθερό σημείο.

Αυτό το φαινόμενο πρέπει να διερευνηθεί, επειδή ο τύπος είναι μη γραμμικός και κατά συνέπεια έχει τάσεις αποσταθεροποίησης, πάνω στη προσπάθεια εξασφάλισης καλύτερης συσχέτισης των τιμών.

Η στατιστική του εκτίμηση ουσιαστικά γίνεται με τη χρήση των κριτηρίων σύγκλισης που είδαμε.

**DD-NHL: Data-Driven Non Linear Hebbian Learning**

Μια τροποποιημένη έκδοση του αλγορίθμου NHL προτάθηκε από τον Stach, η οποία ενσωματώνει ιστορικά στοιχεία της modento η διαδικασία εκμάθησης. Αυτός ο αλγόριθμος ονομάζεται βάσει δεδομένων Nonlinerar Hebbian Μάθηση (DD-NHL) και διαφέρει από τον NHL σε δύο στάδια:

1. Αντί οι τιμές των εννοι'ων να υπολογίζονται από την εξίσωση(1) υπολογίζονται από ιστορικά δεδομένα. Αυτά λαμβάνονται από τον ΝΗL με την εφαρμογή του στο πρώτο βήμα.
2. Το κριτήριο σύγκλισης F1 αντικαθίσταται από τη μέτρηση των εννοιών στο επιθυμητό διάστημα. Περισσότερο ακριβώς το υποψήφιο σύνολο βαρών που χρησιμοποιήθηκε στην προσομοίωση του μοντέλου FCM και αν οι τιμές βρίσκονται σε ένα επιθυμητό επίπεδο η διαδικασία μάθησης ολοκληρώνεται(με εξασφάλιση του κριτηρίου F2), αλλιώς συνεχίζεται.

Ο σκοπός διαμόρφωσης του DD-NHL είναι γιά δύο λόγους.

Ο πρώτος να χρησιμοποιήσει τα ιστορικά βελτιωμένα δεδομένα του NHL, ώστε να εξασφαλιστεί η ελάχιστη εξάρτηση από τις αρχικές "ευαίσθητες" συνθήκες με σκοπό τη καλύτερη σύγκλιση (αποφυγή χαοτικών καταστάσεων), και,

Ο δεύτερος να διατηρήσει το μη γραμμικό χαρακτήρα του NHL ώστε να διαθέτει υψηλό βαθμό συσχέτισης των βαρών, προσπαθώντας να επιφέρει ισορροπία στην τελική κατάσταση του συστήματος.

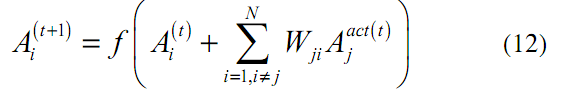
**AHL: Active Hebbian Learning**

Ένα κοινό χαρακτηριστικό των προηγούμενων αλγορίθμων μάθησης είναι ότι σε κάθε επανάληψη των εννοιών του μοντέλου FCM οι κόμβοι ενημερώνονται χρησιμοποιώντας την Εξ. (1), συγχρονισμένα.

Έτσι ο ΑΗL δημιουργήθηκε για να λάβει υπ΄όψη του τη χρονική τυχαιότητα του συστήματος ( όχι τον χρόνο εκτέλεσης του αλγόριθμου) τελειοποιώντας την προσομοίωση με πραγματικά βιολογικά χρονικά χαρακτηριστικά, προσπαθώντας να ξεφύγει από τον χαρακτηρισμό "τεχνητά νευρωνικά δίκτυα".

Η Active Hebbian Μάθηση (AHL) σαν αλγόριθμος, θεωρεί ότι οι κόμβοι των εννοιών ενεργοποιούνται ασύγχρονα από μια συγκεκριμένη ακολουθία. Με αυτό τον τρόπο το σημείο ισορροπίας επιτυγχάνεται με την εξέταση διαφορετικών χρόνων ενεργοποίησης των κόμβων, ένα μηχανισμό που είναι χρήσιμος σε συστήματα όπου οι έννοιες ενεργοποιούνται και βασίζονται σε μια ειδική ακολουθία ενεργοποίησης. Επιπλέον, σε αντίθεση με τους προηγούμενους αλγόριθμους, όπου μόνο τα μη μηδενικά βάρη (βλέπε DHL) ανανεώνονται, στην περίπτωση του αλγορίθμου AHL όλα τα βάρη, εκτός από εκείνα της διαγωνίου ενημερώνονται.

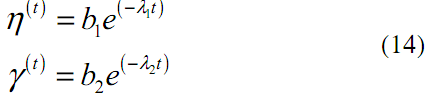
Με βάση τον αλγόριθμο AHL, οι κόμβοι του μοντέλου FCM διακρίνονται σε ενεργοποιημένους και ενεργές έννοιες, από τους προηγούμενους που είναι οι κόμβοι που πρώτα ενεργοποιούνται προκαλώντας την ενεργοποίηση των τελευταίων. Στην περίπτωση αυτή, οι ενεργοποιημένες έννοιες ενημερώνονται μέσα από τον ακόλουθο κανόνα:



Όπου είναι οι τιμές των ενεργοποιημένων εννοιών που επηρεάζουν την έννοια Ci σύμφωνα με τον τύπο



Όπου η σταθερά του ρυθμού μάθησης η και η σταθερά καθυστέρησης γ λαμβάνονται από τις σχέσεις

με 

Έτσι τα κριτήρια του NHL μπορούν να εφαρμόζονται στον AHL προκειμένου να εξασφαλιστεί το σημείο ισορροπίας του συστήματος.

\* Εδώ ένα μικρό σχόλιο για την επιλογή των τύπων των η και γ από τη θεωρία πιθανοτήτων θα ήταν χρήσιμο για την καλύτερη κατανόηση και τη μαθηματική ερμηνεία.

Στη θεωρία πιθανοτήτων και στη στατιστική, η εκθετική κατανομή (π.χ. αρνητική εκθετική κατανομή) είναι η κατανομή πιθανότητας που περιγράφει τον χρόνο μεταξύ των γεγονότων σε μια διαδικασία Poisson, δηλ. Μια διαδικασία στην οποία τα γεγονότα συμβαίνουν συνεχώς και ανεξάρτητα με σταθερό μέσο ρυθμό. Είναι μια ιδιαίτερη περίπτωση της κατανομής γάμμα. Είναι το συνεχές ανάλογο της γεωμετρικής κατανομής και έχει την βασική ιδιότητα να είναι χωρίς μνήμη. Εκτός από το γεγονός ότι χρησιμοποιείται για την ανάλυση των διαδικασιών Poisson, βρίσκεται και σε διάφορα άλλα πλαίσια.

|  |  |
| --- | --- |
| Συνάρτηση πυκνότητας πιθανότητας  [Probability density function](https://en.wikipedia.org/wiki/File:Exponential_pdf.svg) | |
| Σωρευτική συνάρτηση κατανομής  [Cumulative distribution function](https://en.wikipedia.org/wiki/File:Exponential_cdf.svg) | |
| **Παράμετροι** | *λ* > 0 ρυθμός, |
| [**Support**](https://en.wikipedia.org/wiki/Support_%28mathematics%29) | *x* ∈ [0, ∞) |
| [**PDF**](https://en.wikipedia.org/wiki/Probability_density_function) | λ *e*−*λx* |
| [**CDF**](https://en.wikipedia.org/wiki/Cumulative_distribution_function) | 1 − *e*−*λx* |
| [**Mean**](https://en.wikipedia.org/wiki/Expected_value) | *λ*−1 (= *β*) |
| [**Median**](https://en.wikipedia.org/wiki/Median) | *λ*−1 ln(2) |
| [**Mode**](https://en.wikipedia.org/wiki/Mode_%28statistics%29) | 0 |
| [**Variance**](https://en.wikipedia.org/wiki/Variance) | *λ*−2 (= *β*2) |
| [**Skewness**](https://en.wikipedia.org/wiki/Skewness) | 2 |
| [**Ex. kurtosis**](https://en.wikipedia.org/wiki/Excess_kurtosis) | 6 |
| [**Entropy**](https://en.wikipedia.org/wiki/Information_entropy) | 1 − ln(*λ*) |
|  |  |

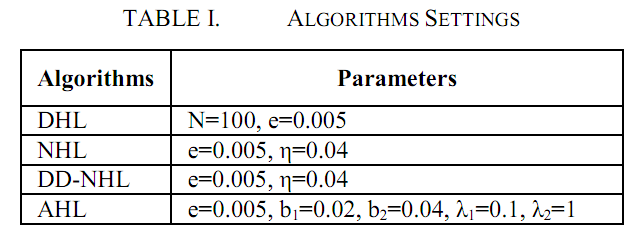
Νομίζω ότι αν "δούμε" τα μαθηματικά χαρακτηριστικά της κατανομής θα φανεί ότι είναι η κατάλληλη μαθηματική επιλογή και για τις σταθερές η και γ αφ΄ενός, αλλά και για την εξασφάλιση της ταχείας σύγκλισης του αλγόριθμου από την άλλη (από τεχνική άποψη). Το διάγραμμα της CDF ταιριάζει απόλυτα στη σύγκλιση του αλγόριθμου που όπως όλοι λειτουργούν αναδρομικά.

Ενώ όλα τη στιγμή αυτή φαίνονται ιδανικά κατασκευασμένα έχοντας λάβει υπόψη όλες τις λεπτομέρειες για τη σύγκλιση του δυναμικού συστήματος μάθησης του δικτύου, οι πρακτικές συνθήκες επιλογής του κατάλληλου αλγόριθμου, αλλά και η συνεχής βελτίωση των υπαρχόντων είναι ένα σκληρό και το πιό σημαντικό ανοικτό θέμα.

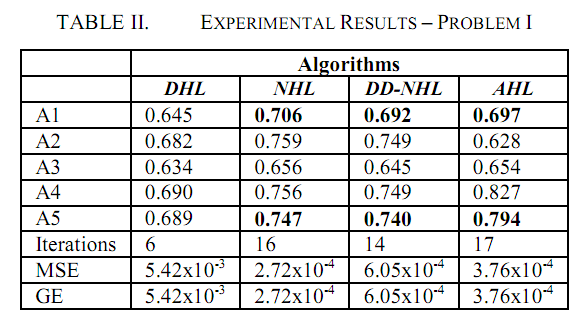
**Επιλογή Κατάλληλου Αλγορίθμου**

Η επιλογή του κατάλληλου αλγόριθμου σύγκλισης γίνεται με παράλληλο τρέξιμο όλων των αλγορίθμων και την κατάλληλη μαθηματική στατιστική εκτίμηση. Αυτή η επιλογή δεν είναι καθολίκη καθώς κάθε σύστημα είναι μοναδικό. Παρακάτω αναλύεται η επιλογή αυτή σε τρία παραδείγματα.

**Παράδειγμα Ι**

****

Οι τιμές των σταθερών που αναφέρονται στον πίνακα δεν είναι ικανές να μας προσανατολίσουν για την επιλογή και χρειάζονται περισσότερο αναλυτικά στοιχεία.

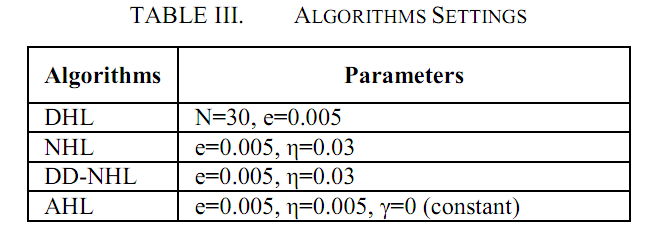
 

Εδώ μπορούμε να παρατηρήσουμε τα εξής :

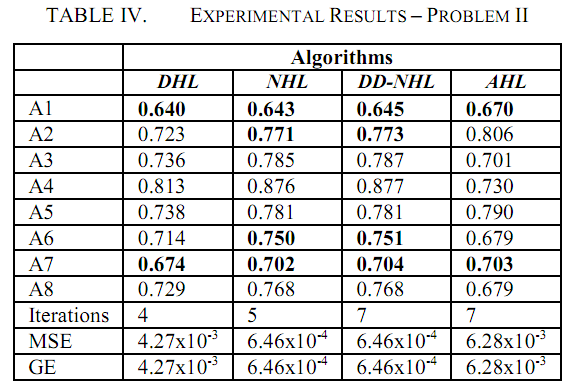
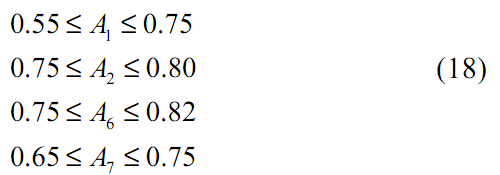
1. Οι τιμές του DHL αν και γενικά είναι στο επιθυμητό εύρος (17) έχουν σημαντικές παραλλαγές ή αποκλίσεις, σχετικά με τα μέγιστα-ελάχιστα. Αν και ο αριθμός των επαναλήψεων είναι μικρός (πράγμα γενικά επιθυμητό) τα στατιστικά μεγέθη εκτίμησης MSE, GE έχουν τις μεγαλύτερες τιμές συγκριτικά με τους άλλους, ώστε αυτά τα στοιχεία είναι ικανά για την απόρρψή του.

2. Από τα στοιχεία των υπόλοιπων αλγόριθμων παρατηρούμε ότι η προσέγγιση στα μέγιστα-ελάχιστα είναι καλύτερη για όλους και ο αριθμός επαναλήψεων περίπου ο ίδιος, ωστόσο δεν είναι ισοδύναμοι αφού ο DD-NHL είναι αυτός με την καλύτερη στατιστική εκτίμηση των MSE, GE. Αυτό θα είναι τελικά και το κριτήριο της επιλογής του.

**Παράδειγμα 2**



Παρατηρείστε ότι αν και το e έχει την ίδια τιμή με το προηγούμενο παράδειγμα, η σταθερά η έχει λίγο μικρότερη τιμή (ρυθμός), ενώ η σταθερά γ είναι 0 δηλαδή δεν υπάρχει καθυστέρηση. Αυτά τα στοιχεία σε πρώτη φάση θα επέλεγαν τον ΑΗL, ωστόσο ο αναλυτικός πίνακας που ακολουθεί βοηθά παραιτέρω.

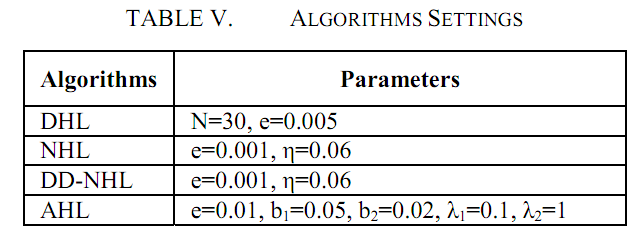
Εδώ έχουμε τα εξής :

1. Τα παραγόμενα πειραματικά αποτελέσματα για την περίπτωση του δεύτερου  
προβλήματος είναι σχεδόν το ίδιο με εκείνα του πρώτου. Ο αλγόριθμος NHL και η τροποποιημένη έκδοση του DD-NHL, δείχνουν την καλύτερη απόδοση διατηρώντας τις τέσσερις έννοιες εξόδου εντός των προκαθορισμένων περιοχών.

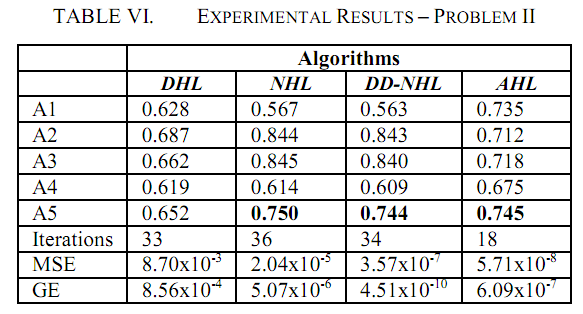
2. Οι αλγόριθμοι DHL και AHL έχουν ακρίβεια 50%, αφού ελέγχουν ικανοποιητικά τις δύο από τις τέσσερις εξόδους του μοντέλου FCM. Και πάλι, όλα τα σετ βαρών εξασφαλίζουν την ευρωστία του μοντέλου FCM με τις αλλαγές στην κατάσταση των αρχικών εννοιών, όπως δείχνει η ισοδυναμία των στατιστικών δεικτών MSE, GE.

3. Τελικά έχουμε ένα είδος ισοδυναμίας των αλγορίθμων NHL , DD-NHL που μας δείχνει και πάλι ότι η καταλληλότερη επιλογή ταλαντεύεται μεταξύ αυτών των δύο που μη ξεχνάμε είναι απόλυτα αλληλοεξαρτώμενοι.

**Παράδειγμα 3**



Εδώ έχουμε κάποιες μεταβολές στις παραμέτρους με μικρότερο e και μεγαλύτερο η. Αυτό το γεγονός όπως θα δούμε επηρεάζει την απόδοση των αλγορίθμων αφού επηρεάζει τις "ευαίσθητες" αρχικές συνθήκες του συστήματος.

Παρόμοια έχουμε :

1. Με βάση τον παραπάνω πίνακα συνάγεται ότι ο DHL αλγόριθμος αποτυγχάνει να εξισορροπήσει σε ένα επιθυμητό σημείο αφού η τελική τιμή της έννοιας εξόδου βρίσκεται εκτός της επιθυμητής περιοχής (19).

2. Από την άλλη πλευρά, οι υπόλοιποι αλγόριθμοι δίνουν μια ικανοποιητική κατάσταση ισορροπίας του μοντέλου FCM, ιδιαίτερα του αλγόριθμου AHL, ο οποίος δίνει την καλύτερη τιμή εξόδου σε μικρότερο χρόνο (18 επαναλήψεις).

3. Όσον αφορά τον δείκτη στατιστικής απόδοσης GE, ο αλγόριθμος DD-NHL βελτιώνει τις δυνατότητες γενίκευσης, καθώς εξισορροπεί την ίδια κατάσταση ανεξάρτητα από την αρχική κατάσταση του μοντέλου. Επιπλέον, αν και η απόδοση του αλγορίθμου NHL είναι αποδεκτή, είναι αρκετά περιθωριακή.

4. Γενικά, ο αλγόριθμος NHL παρουσιάζει ικανοποιητική συμπεριφορά, καθώς ελέγχει πλήρως τις παραλλαγές των εννοιών εξόδου διατηρώντας τες μεταξύ των προδιαγραφών του συστήματος. Παρόλο που το DD-NHL παρουσιάζει παρόμοιες επιδόσεις, η εξάρτησή του από τα ιστορικά δεδομένα, τους δίνει υψηλό χρόνο εκτέλεσης χωρίς σημαντική βελτίωση της απόδοσής του.

5. Οι εναπομείναντες αλγόριθμοι DHL και AHL φαίνεται να επιτελούν το χειρότερο για τα τρία και τα δύο προβλήματα υπό μελέτη αντίστοιχα, αλλά το χαρακτηριστικό του αλγόριθμου AHL για αναζήτηση νέων διασυνδέσεων μεταξύ των εννοιών μπορεί να συμβάλει στον σχεδιασμό πιο αξιόπιστων μοντέλων FCM.

6. Τέλος, πρέπει να αναφερθεί ότι όλοι οι αλγόριθμοι μάθησης που μοιάζουν με Hebbian που μελετήθηκαν σε αυτή την εργασία, εξαρτώνται από κάποιες ελεύθερες παραμέτρους (η, γ, e, Ν), οι οποίες επηρεάζουν σημαντικά τη σύγκλιση και τον τερματισμό των αλγορίθμων στο επιθυμητό μοντέλο FCM. Επομένως, πρέπει να δοθεί ιδιαίτερη προσοχή στην επιλογή αυτών των παραμέτρων εξ΄αρχής.

**Συμπεράσματα**

Μια λεπτομερής περιγραφή της μάθησης που μοιάζει με τους Hebbian αλγόριθμους για την εκπαίδευση μοντέλων FCM παρουσιάστηκε στις προηγούμενες ενότητες. Έχουν υπάρξει τα κατάλληλα πειράματα με σκοπό την ανάλυση της απόδοσης κάθε αλγορίθμου υπό τις ίδιες συνθήκες. Τα αποτελέσματα δείχνουν ότι ο αλγόριθμος NHL παρουσιάζει ικανοποιητική συμπεριφορά με χαμηλή αλγοριθμική πολυπλοκότητα\* σε σύγκριση με τους άλλους παρόμοιους αλγορίθμους. Ενώ η σύγκριση των αλγορίθμων πραγματοποιήθηκε με τη χρήση ορισμένων δεικτών απόδοσης, υπάρχει ανάγκη να αναπτυχθεί μια μεθοδολογία που να επιτρέπει τη σύγκριση των διαφορετικών λύσεων στον εργασιακό χώρο του πραγματικού συστήματος.

Η εξάρτηση των αλγορίθμων που μοιάζουν με Hebbian με ορισμένες ελεύθερες παραμέτρους που επηρεάζουν σημαντικά την απόδοση τους είναι ένα μειονέκτημα αυτών των αλγορίθμων, καθώς ελέγχουν όχι μόνο το τελικό σημείο ισορροπίας του μοντέλου αλλά και τον επιτυχή τερματισμό του αλγορίθμου. Ένας νέος αλγόριθμος με λιγότερη ευαισθησία σε τέτοιες παραμέτρους και με υψηλό ρυθμό σύγκλισης στις επιθυμητές καταστάσεις παραμένει ένας σημαντικός στόχος των επιστημόνων στον τομέα αυτό.

\* Το θέμα της "υπολογιστικής πολυπλοκότητας" είναι πολύ σημαντικό αφού δεν επιζητείται μόνο η βέλτιστη αριθμητική προσέγγιση σαν "βέλτιστη λύση" αλλά και η πιό "απλή". Κάποιοι παράγοντες που επηρεάζουν την υπολογιστική πολυπλοκότητα γενικά είναι οι εξής :

1. Ο χρόνος εκτέλεσης και τερματισμού του αλγόριθμου (αριθμός επαναλήψεων και κριτήρια τερματισμού).

2. Η ελαχιστοποίηση του πλήθους των ελεύθερων παραμέτρων (e, η, γ...) που επηρεάζουν τις ¨ευαίσθητες" αρχικές συνθήκες αρα και τη σύγκλιση του συστήματος.

3. Η επιλογή κατάλληλων μεθόδων στατιστικής εκτίμησης (MSE, GE) με ελαχιστοποίηση των σφαλμάτων.

4. Τα μεταφυσικά-φιλοσοφικά ερωτήματα :

"αν υπάρχει και μπορεί να εφευρεθεί ο μοναδικός ικανοποιητικός μαθησιακός αλγόριθμος"

"αν υπάρχει μιά απειρία "λύσεων" στο πρόβλημα, με ποιά κριτήρια θα επιλέξουμε την "κατάλληλη" λύση"

"αν τέλος η υπολογιστική μάθηση είναι μια "τελειωτική" διακασία τουλάχιστον όσον αφορά τα νευρωνικά δίκτυα"

**GitHub**

Το project είναι αναρτημένο στο [GitHub](https://github.com/ziqushru/JFCM-Design-App), υπό το [Apache License 2.0](https://www.apache.org/licenses/LICENSE-2.0), με σκοπό την ελεύθερη χρήση του κώδικα και του εκτελέσιμου, εφόσον καλύπτονται οι απαιτούμενοι όροι. Το versioning έχει γίνει βάση του [Sematic Versioning 2.0.0](http://semver.org/).

**Βιβλιογραφία**