



ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ
ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΘΕΜΑ: FUZZY- NEURAL METHOD FOR MODELING AND CONTROL

ΜΕΘΟΔΟΛΟΓΙΑ ΝΕΥΡΟ-ΑΣΑΦΩΝ ΜΟΝΤΕΛΩΝ

ΕΙΣΗΓΗΤΗΣ :

ΑΝΤΩΝΙΟΣ ΚΩΝΣΤΑΝΤΑΡΑΣ

ΦΟΙΤΗΤΗΣ:

ΦΙΛΙΠΠΑΣ ΙΩΑΝΝΗΣ

ΠΕΡΙΛΗΨΗ

Η συγκεκριμένη λογική, τα νευρωνικά δίκτυα και οι γενετικοί αλγόριθμοι έχουν σημειώσει σημαντική ανάπτυξη την τελευταία δεκαετία. Όλα έχουν έναν αριθμό πλεονεκτημάτων και μειονεκτημάτων, συγκρινόμενα το ένα με το άλλο. Η συγκεκριμένη λογική είναι μία συνήθως χρησιμοποιούμενη μέθοδος για αρχική διαμόρφωση, αλλά δεν είναι ικανή για προσαρμογή.

Τα νευρωνικά δίκτυα είναι προσαρμοστικά, αλλά όχι τόσο αποτελεσματικά στη διαμόρφωση, όσο η νευρωνική λογική. Οι γενετικοί αλγόριθμοι μπορούν να εξελίσσονται, αλλά είναι μία πιθανολογική μέθοδος επίλυσης και ανήκει σε μέθοδο αυτοδιδασχής. Η πιο πρόσφατη έρευνα στοχεύει στο να συνδυάσει τις παραπάνω τεχνικές, ώστε να καλύψει τις αδυναμίες του καθενός με τα δυνατά σημεία του άλλου. Αυτό το έγγραφο ερευνά έναν αριθμό συγκεκριμένων-νευρωνικών τεχνικών και των εφαρμογών τους στα πεδία της διαμόρφωσης και ελέγχου.

ΠΕΡΙΕΧΟΜΕΝΑ

1 Τίτλος Έργου.....	6
1.1 Δήλωση του προβλήματος.....	6
1.2 Στόχοι και επιδιώξεις του έργου.....	6
2.1 Διαμόρφωση με χρήση του ANFIS.....	8
2.2 Μοντέλο εκμάθησης και διαδικασία εξαγωγής συμπερασμάτων μέσω ANFIS.....	8
2.3.1 Τι είναι το ANFIS.....	8
2.3.2 Δομή FIS και ρύθμιση παραμέτρων.....	9
2.4 Αρχιτεκτονική του ANFIS.....	11
2.5 Αλγόριθμοι που χρησιμοποιούνται με το ANFIS.....	16

3 Διαμόρφωση του $G(s)$ μέσω συγκεκριμένων-νευρωνικών ρυθμίσεων.....	18
3.1 Διαμόρφωση του $G(s)$ για μία μοναδιαία βηματική είσοδο.....	20
3.2 Διαμόρφωση του $G(s)$ με χρήση μίας ψευδó τυχαίας δυαδικής ακολουθίας σαν είσοδο στο σύστημα.....	37
3.3 Συζήτηση και συμπεράσματα σχετικά με τη διαμόρφωση.....	50
4.1 Συγκεκριμένα-Νευρωνικά Δίκτυα.....	52
4.2 Συγκεκριμένα Τεχνητά Νευρωνικά Δίκτυα (FANN – Fuzzy Artificial Neural Networks).....	54
4.3 Νευρο-Συγκεκριμένος Έλεγχος.....	55
4.3.1 Παραδοσιακός Έλεγχος.....	56
4.3.2 Νευρωνικός Έλεγχος.....	58
4.3.3 Συγκεκριμένος Έλεγχος.....	59
4.3.4 Συγκεκριμένος-Νευρωνικός Έλεγχος.....	60
5 Συγκεκριμένος Ελεγκτής.....	61
5.1 Δυναμική Συστήματος.....	62
5.2 Δημιουργία ενός κατάλληλου σήματος εισόδου.....	62
5.3 Ενίσχυση του σήματος εξόδου.....	75
5.4 Μέθοδοι για περαιτέρω βελτίωση.....	81
5.4.1 Αντίστροφη Εκμάθηση.....	88
5.4.2 Εκπαίδευση των συστημάτων συγκεκριμένης λογικής χρησιμοποιώντας πίσω (ανάστροφη) διάδοση.....	93
5.5 Συζήτηση και συμπεράσματα στο συγκεκριμένο έλεγχο.....	101
Βιβλιογραφία.....	103

1 Τίτλος έργου

Συγκεκριμένες-Νευρωνικές Μέθοδοι για Διαμόρφωση και Έλεγχο

1.1 Δήλωση του προβλήματος

Το $G(s)$ είναι ένα τρίτης τάξης σύστημα, που προέρχεται από ένα στοιχείο (συνιστώσα) δεύτερης τάξης, που έχει υποστεί ελαφρά απόσβεση, και μίας πρώτης τάξης ρύθμιση. Μόνο γνωρίζοντας τις εισόδους και τις εξόδους του $G(s)$, αναπτύσσουμε ένα συγκεκριμένο-νευρωνικό σύστημα για να διαμορφώνει το $G(s)$ για:

- i) μοναδιαία βηματική είσοδο
- ii) ψευδó τυχαίο (απρογραμματίστο) δυαδικό σύστημα

Μιλώντας γενικά, το ψευδό τυχαίο δυαδικό σύστημα (PRBS) θα αποδώσει ένα περισσότερο πολύπλοκο σήμα στην έξοδο του $G(s)$ από ότι η βαθμιαία βηματική είσοδος. Με αυτό τον τρόπο μπορούμε να ερευνήσουμε πόσο κοντά είναι η έξοδος του νευρωνικού-συγκεχυμένου μοντέλου στην πραγματική έξοδο του συστήματος, και συνεπώς να αποσπάσουμε πληροφορίες για τη χρησιμότητα του μοντέλου.

Ερευνάμε μεθόδους στο συγκεχυμένο-νευρωνικό έλεγχο, σχεδιάζουμε ένα συγκεχυμένο ελεγκτή και χρησιμοποιούμε το συγκεχυμένο-νευρωνικό μοντέλο για να οδηγήσουμε τις παραμέτρους του ελεγκτή, με σκοπό να επιτύχουμε ένα σήμα εξόδου όσο δυνατόν πιο κοντά στο σήμα εισόδου. Συζητάμε το αποτέλεσμα και αποσπάμε πληροφορίες για την αποτελεσματικότητα του συγκεχυμένου ελεγκτή σε σύγκριση με τις κλασσικές μεθόδους ελέγχου.

1.2 Στόχοι και επιδιώξεις του έργου

- απόκτηση εξοικείωσης με την εργαλειοθήκη του ANFIS (Τεχνητά Νευρωνικά-Συγκεχυμένα Συμπερασματικά Συστήματα)
- κατανόηση για το πώς το ANFIS μπορεί να χρησιμοποιηθεί για διαμόρφωση
- διαμόρφωση του $G(s)$ μέσα στις συγκεχυμένες-νευρωνικές ρυθμίσεις, μόνο γνωρίζοντας τις εισόδους και τις εξόδους
- διερεύνηση των συγκεχυμένων-νευρωνικών μεθόδων για έλεγχο
- σχεδιασμός του συγκεχυμένου ελεγκτή
- χρησιμοποίηση του συγκεχυμένου-νευρωνικού μοντέλου για να καθοδηγήσει τον ελεγκτή
- διερεύνηση του τρόπου που ο ελεγκτής προσαρμόζει οποιαδήποτε στιγμή υπάρχει αλλαγή στις παραμέτρους του τρόπου πρώτης τάξης

2.1 Διαμόρφωση με χρήση του ANFIS

Ας υποθέσουμε ότι κάποιος θέλει να εφαρμόσει συγκεχυμένη συμπερασματική διαδικασία σε ένα σύστημα, για το οποίο έχει ήδη μία συλλογή δεδομένων εισόδου/εξόδου, τα οποία θα ήθελε να χρησιμοποιήσει για διαμόρφωση. Δεν έχει απαραίτητα ένα προκαθορισμένο μοντέλο δομής βασισμένο στα χαρακτηριστικά των μεταβλητών στο σύστημά του.

Θα υπάρχουν κάποιες καταστάσεις διαμόρφωσης, στις οποίες δεν θα μπορεί απλά να κοιτά τα δεδομένα και να διακρίνει με τι θα έπρεπε να μοιάζουν οι συμμετοχικές συναρτήσεις. Αντί να επιλεγούν οι παράμετροι, που συνδέονται με μία δεδομένη συμμετοχική συνάρτηση αυθαίρετα, αυτές οι παράμετροι θα μπορούσαν να επιλεγούν έτσι ώστε, να κολλήσουν οι συμμετοχικές συναρτήσεις στα δεδομένα

εισόδου/εξόδου, με σκοπό να υπολογιστούν για αυτούς τους τύπους διακύμανσης, στις τιμές των δεδομένων. Σε αυτό στοχεύουν οι λεγόμενες νευρωνικές-προσαρμοστικές εκπαιδευτικές τεχνικές, που είναι ενσωματωμένες στο ANFIS, στην εργαλειοθήκη της συγκεκριμένης λογικής.

2.2 Μοντέλο εκμάθησης και διαδικασία εξαγωγής συμπερασμάτων μέσω του ANFIS

Η βασική ιδέα πίσω από αυτές τις νευρωνικο-προσαρμοστικές εκπαιδευτικές τεχνικές είναι πολύ απλή. Αυτές οι τεχνικές προσφέρουν μία μέθοδο για τη διαδικασία της συγκεκριμένης διαμόρφωσης, για να μαθευτούν πληροφορίες για ένα σύνολο δεδομένων, με σκοπό να υπολογιστούν οι παράμετροι των συμμετοχικών συναρτήσεων, που οι καλύτερες επιτρέπουν το συνεργαζόμενο συγκεκριμένο συμπερασματικό σύστημα να ανιχνεύσει τα δεδομένα στοιχεία εισόδου/εξόδου. Αυτή η εκπαιδευτική μέθοδος δουλεύει παρόμοια με αυτή των νευρωνικών δικτύων. Η λειτουργία της εργαλειοθήκης της συγκεκριμένης λογικής, που πραγματοποιεί τη ρύθμιση των παραμέτρων της συμμετοχικής συνάρτησης, ονομάζεται anfis.

2.3.1 Τι είναι το ANFIS

Το ακρωνύμιο ANFIS προέρχεται από τις λέξεις Adaptive Neuro Fuzzy Inference System (Προσαρμοστικό Νευρωνικό Συγκεκριμένο Συμπερασματικό Σύστημα). Χρησιμοποιώντας ένα δεδομένο σύνολο στοιχείων εισόδου/εξόδου, η λειτουργία anfis της εργαλειοθήκης κατασκευάζει ένα συγκεκριμένο σύστημα συμπερασμού (FIS), του οποίου οι παράμετροι των συμμετοχικών συναρτήσεων συντονίζονται (ρυθμίζονται) χρησιμοποιώντας είτε έναν αλγόριθμο πίσω (ανάστροφης) διάδοσης από μόνο του, ή σε συνδυασμό με τη μέθοδο των ελαχίστων τετραγώνων. Αυτό επιτρέπει τα συγκεκριμένα συστήματα να μάθουν από τα δεδομένα που αυτά διαμορφώνουν.

2.3.2.Δομή FIS και ρύθμιση παραμέτρων

Μία δομή τύπου δικτύου, παρόμοια με αυτή ενός νευρωνικού δικτύου, η οποία σχεδιάζει εισόδους μέσω των συμμετοχικών συναρτήσεων εισόδου και των συνδεδεμένων παραμέτρων, και μετά μέσω των συμμετοχικών συναρτήσεων εξόδου και των συνδεδεμένων παραμέτρων στις εξόδους, μπορεί να χρησιμοποιηθεί για ερμηνεύσει τα σχέδια εισόδου/εξόδου.

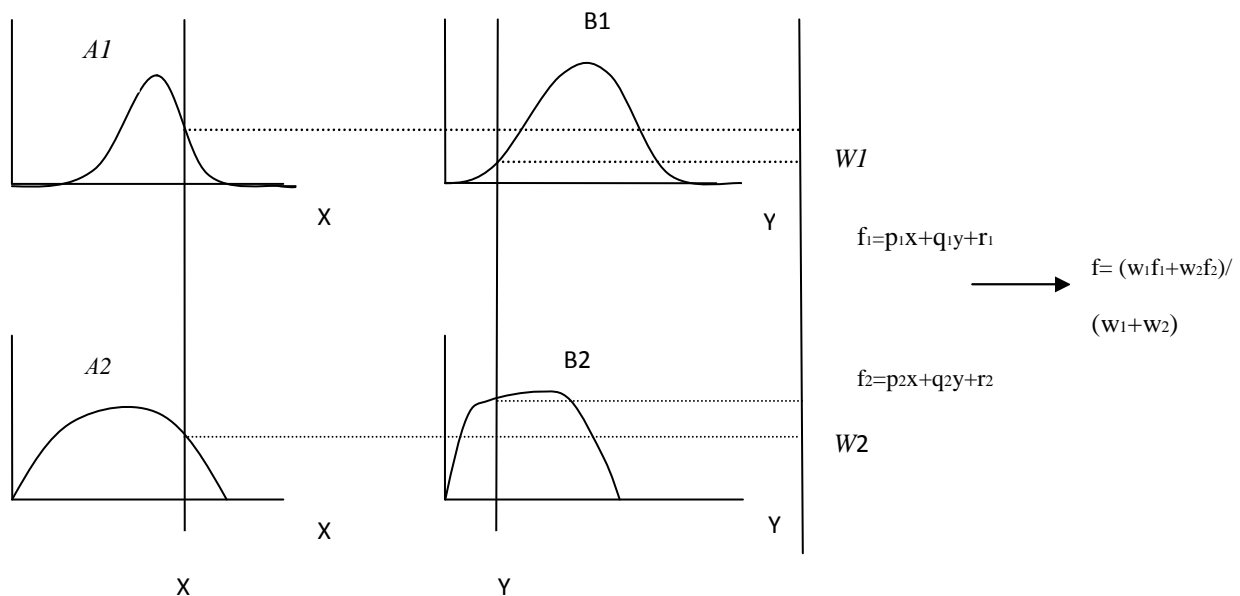
Οι παράμετροι που συνδέονται με τις συμμετοχικές συναρτήσεις θα αλλάξουν μέσω της εκπαιδευτικής διαδικασίας. Ο υπολογισμός αυτών των παραμέτρων (ή η ρύθμισή τους) διευκολύνεται από ένα διάνυσμα (άνυσμα) κλίσης, το οποίο προσφέρει ένα μέτρο του πόσο καλά το συγκεκριμένο συμπερασματικό σύστημα διαμορφώνει τα δεδομένα εισόδου/εξόδου, για ένα δεδομένο σύνολο παραμέτρων. Μόλις το άνυσμα κλίσης εξασφαλιστεί, κάθε ένα από τα διάφορα σύντομα προγράμματα βελτιστοποίησης θα μπορούσε να εφαρμοστεί, με σκοπό να ρυθμίσει τις παραμέτρους, ώστε να μειώσει μερικά σφάλματα μέτρησης (συνήθως ορίζονται από το άθροισμα της διαφοράς των τετραγώνων μεταξύ των πραγματικών και των επιθυμητών εξόδων). Το anfis χρησιμοποιεί είτε πίσω διάδοση ή έναν συνδυασμό υπολογισμού ελαχίστων τετραγώνων και πίσω διάδοσης για την εκτίμηση των παραμέτρων της συμμετοχικής συνάρτησης.

2.4 Αρχιτεκτονική ANFIS

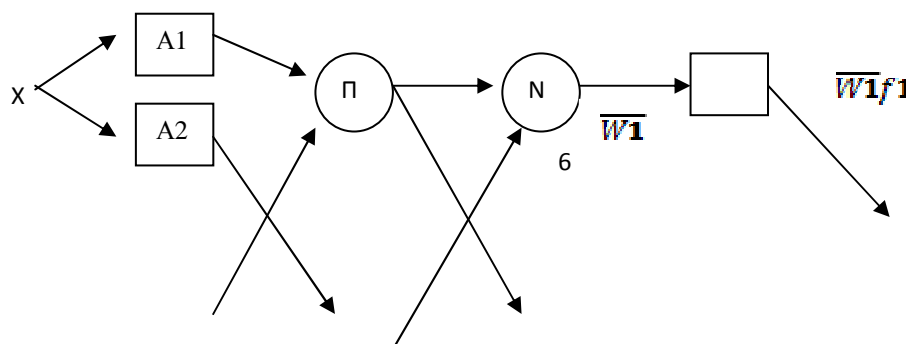
Για απλοικότητα, υποθέτουμε ότι το συγκεκριμένο συμπερασματικό σύστημα, υπό εξέταση έχει δύο εισόδους x και y και μία έξοδο z . Για ένα συγκεκριμένο μοντέλο Sugeno πρώτης τάξης, ένα κοινό πακέτο κανόνων με δύο συγκεκριμένους κανόνες τύπου “αν-τότε” είναι το ακόλουθο:

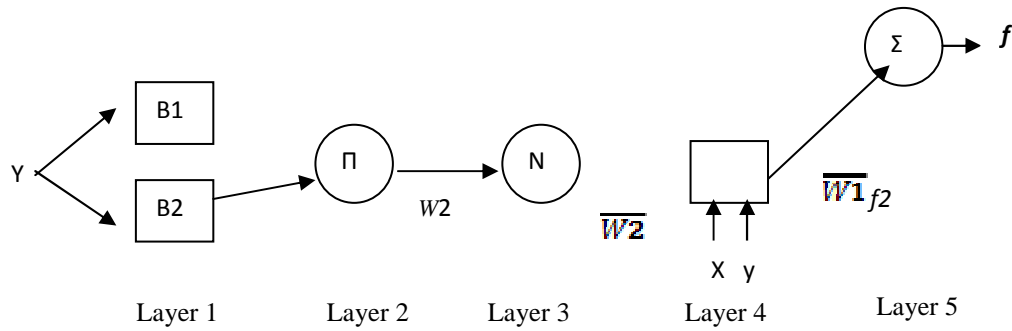
Κανόνας 1: Αν x είναι A_1 και y είναι B_1 , τότε $f_1 = p_1x + q_1y + r_1$,

Κανόνας 2: Αν x είναι A_2 και y είναι B_2 , τότε $f_2 = p_2x + q_2y + r_2$



Σχήμα 1(a): Ένα πρώτης τάξης συγκεκριμένο μοντέλο Sugeno, δύο εισόδων με δύο κανόνες





Σχήμα 1(b): Ισοδύναμη αρχιτεκτονική ANFIS

Το Σχήμα 1(a) απεικονίζει το συλλογιστικό μηχανισμό για αυτό μοντέλο Sugeno. Η αντίστοιχη ισοδύναμη αρχιτεκτονική παρουσιάζεται στο Σχήμα 1(b), όπου οι κόμβοι του ίδιου επιπέδου έχουν παρόμοιες συναρτήσεις, όπως περιγράφεται παρακάτω.

Υποδηλώνουμε την έξοδο του i κόμβου στο επίπεδο / σαν $O_{1,i}$.

Επίπεδο 1: Κάθε κόμβος σε αυτό το επίπεδο είναι ένας προσαρμοστικός κόμβος, με μία συνάρτηση κόμβου:

$$O_{1,i} = \mu_{A_i}(x), \text{ για } i=1, 2, \text{ ή}$$

$$O_{1,i} = \mu_{B_{i-2}}(y), \text{ για } i=3, 4,$$

Όπου x (ή y) είναι η είσοδος στον κόμβο i και A_i (ή B_{i-2}) είναι μία γλωσσική ταμπέλα (σαν να λέμε “μικρός” ή “μεγάλος”), που συνδέεται με αυτό τον κόμβο. Με άλλα λόγια, $O_{1,i}$ είναι ο συμμετοχικός βαθμός ενός συγκεκριμένου συνόλου A ($= A_1, A_2, B_1$ ή B_2) και προσδιορίζει το βαθμό, στον οποίο η δεδομένη είσοδος x (ή y) ικανοποιεί την ποσότητα A . Εδώ, η συμμετοχική συνάρτηση για A μπορεί να είναι κάθε κατάλληλη παραμετρική συμμετοχική συνάρτηση, π.χ. η γενικευμένη συνάρτηση μορφής “καμπάνα”:

$$\mu_A(x) = 1 / (1 + |(x - c_i) / a_i|^{2b}),$$

όπου, $\{a_i, b_i, c_i\}$ είναι ένα σετ παραμέτρων. Όταν οι τιμές αυτών των παραμέτρων αλλάζουν, η συνάρτηση μορφής-καμπάνας κυμαίνεται ανάλογα, συνεπώς παρουσιάζει διάφορες μορφές των συμμετοχικών συναρτήσεων για το συγκεκριμένο σύνολο A . Οι παράμετροι σε αυτό το επίπεδο αναφέρονται ως **συλλογιστικές παράμετροι**.

Επίπεδο 2: Κάθε κόμβος σε αυτό το επίπεδο είναι ένας κόμβος επισημασμένος με Π, του οποίου η έξοδος είναι το προϊόν όλων των σημάτων εισόδου:

$$Q_{2,i} = w_i = \mu_{A_i}(x) \mu_{B_i}(y), \quad i=1, 2.$$

Η έξοδος του κάθε κόμβου αντιπροσωπεύει τη δύναμη τήρησης ενός κανόνα. Γενικά, κάθε άλλος T-nom χειριστής, που εκτελούν συγκεκριμένα AND, μπορούν να χρησιμοποιηθούν σαν την κομβική συνάρτηση σε αυτό το επίπεδο.

Επίπεδο 3: Κάθε κόμβος σε αυτό το επίπεδο είναι ένας σταθερός κόμβος, επισημασμένος με N. Ο κόμβος ith

$$Q_{3,i} = W_i = w_i / (w_1 + w_2), \quad i=1, 2$$

υπολογίζει το ποσοστό της δύναμης εκτέλεσης του ith κανόνα στο σύνολο της δύναμης όλων των κανόνων:

Για άνεση, οι έξοδοι αυτού του επιπέδου ονομάζονται κανονικοποιημένες δυνάμεις εκτέλεσης.

Επίπεδο 4: Κάθε κόμβος i σε αυτό το επίπεδο είναι ένας προσαρμοστικός κόμβος με μία κομβική συνάρτηση

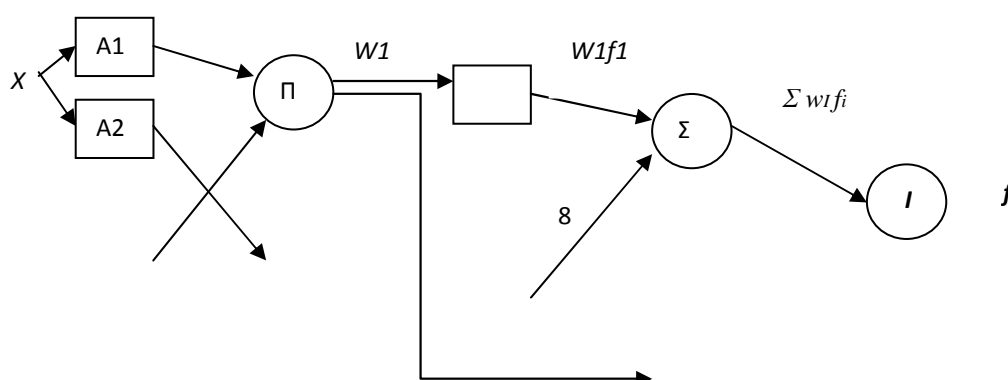
$$O_{4,i} = \bar{W}_i f_i = \bar{W}_i (p_i x + q_i y + r_i), \text{ where } \bar{W}_i$$

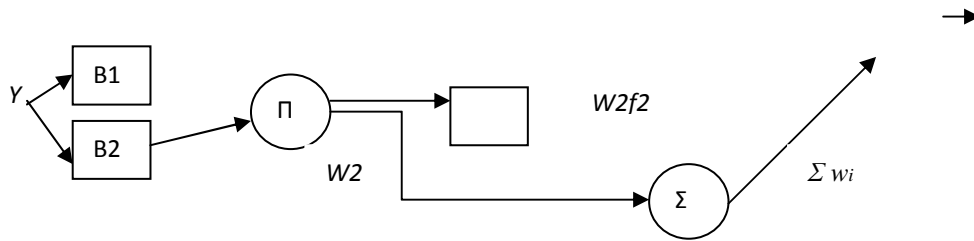
και είναι μία κανονικοποιημένη δύναμη εκτέλεσης από το επίπεδο 3 και $\{p_i, q_i, r_i\}$ είναι το σετ παραμέτρων αυτού του κόμβου. Οι παράμετροι σε αυτό το επίπεδο αναφέρονται ως **επακόλουθοι παράμετροι**.

Επίπεδο 5: Ο απλός κόμβος σε αυτό το επίπεδο είναι ένας σταθερός κόμβος επισημασμένος με Σ, ο οποίος υπολογίζει τη συνολική έξοδο σαν την άθροιση όλων των εισερχόμενων σημάτων:

$$O_{5,1} = \sum_i \bar{W}_i$$

συνολική_έξοδος =
$$if_i = \frac{\sum_i w_i f_i}{\sum_i w_i}$$





Σχήμα 2: Αρχιτεκτονική ANFIS για το συγκεκριμένο μοντέλο Sugeno , όπου η εξομάλυνση του βάρους πραγματοποιείται στο τελευταίο επίπεδο.

Συνεπώς έχουμε κατασκευάσει ένα προσαρμοστικό δίκτυο, το οποίο είναι

	Forward pass	Backward pass
Premise parameters	Fixed	Gradient decent
Consequent parameters	Least – squares estimator	Fixed

συναρτησιακά ισοδύναμο με ένα συγκεκριμένο μοντέλο Sugeno. Παρατηρούμε ότι, η δομή αυτού του προσαρμοστικού δικτύου δεν είναι μοναδική: μπορούμε να συνδυάσουμε τα επίπεδα 3 και 4 για να αποκτήσουμε ένα ισοδύναμο δίκτυο με μόνο 4 επίπεδα. Με το ίδιο τεκμήριο, μπορούμε να πραγματοποιήσουμε την εξομάλυνση του βάρους στο τελευταίο επίπεδο. Το Σχήμα 2 απεικονίζει ένα ANFIS αυτού του τύπου. Στην ακραία περίπτωση, μπορούμε ακόμα να συρρικνώσουμε όλο το δίκτυο σε ένα μοναδικό προσαρμοστικό κόμβο με το ίδιο σετ παραμέτρων. Προφανώς, η ανατεθείσα εργασία των κομβικών συναρτήσεων και η διαμόρφωση του δικτύου είναι αυθαίρετες, όσο ο κάθε κόμβος και το κάθε επίπεδο εκτελεί εποικοδομητικές λειτουργικότητες που αποτελούνται από πρότυπα συναρτησιακά στοιχεία.

Signals	Node outputs	Error signals
---------	--------------	---------------

2.5 Αλγόριθμοι που χρησιμοποιούνται με το ANFIS

Το ANFIS χρησιμοποιεί δύο αλγόριθμους για να πραγματοποιήσει νευρωνική προσαρμογή, τον αλγόριθμο πίσω (ανάστροφης) διάδοσης και τον αλγόριθμο υβριδικής εκμάθησης. Ο αλγόριθμος υβριδικής εκμάθησης είναι ένας συνδυασμός του αλγόριθμου (ανάστροφης) πίσω διάδοσης και της μεθόδου των ελαχίστων τετραγώνων. Πιο συγκεκριμένα, στο εμπρόσθιο πέρασμα του αλγόριθμου υβριδικής εκμάθησης, οι έξοδοι στους κόμβους πάνε μπροστά μέχρι το επίπεδο 4 και οι επακόλουθες παράμετροι προσδιορίζονται με τη μέθοδο των ελαχίστων τετραγώνων. Στο οπισθοδρομικό πέρασμα, τα σήματα σφαλμάτων διαδίδονται προς τα πίσω και οι παράμετροι προϋποθέσεων ενημερώνονται από το διάνυσμα κλίσης. Ο ακόλουθος πίνακας συνοψίζει τις δραστηριότητες σε κάθε πέρασμα.

	Εμπρόσθιο πέρασμα	Οπισθοδρομικό πέρασμα
Παράμετροι Προϋποθέσεων	Σταθερά	Κλίση διανύσματος
Επακόλουθες Παράμετροι	Εκτιμητής Ελαχίστων Τετραγώνων	Σταθερά
Σήματα	Έξοδοι κόμβου	Σήματα Λάθους

Παρόλο που το ANFIS μας επιτρέπει να χρησιμοποιήσουμε οποιονδήποτε από τους δύο αλγόριθμους, υπάρχει μία κλίση προς τον αλγόριθμο υβριδικής εκμάθησης, αφού αυτή η μέθοδος επίλυσης συγκλίνει ταχύτερα, γιατί μειώνει τις διαστάσεις του διαστήματος αναζήτησης της αυθεντικής, θεωρητικής μεθόδου πίσω διάδοσης.

3. Διαμόρφωση $G(s)$ μέσω συγκεκριμένων –νευρωνικών ρυθμίσεων

Σε αυτό το στάδιο του έργου, μας ζητείται να δημιουργήσουμε ένα συγκεκριμένο-νευρωνικό μοντέλο, που να λειτουργεί ακριβώς σαν ένα $G(s)$ plant, μόνο γνωρίζοντας την είσοδο προς και την έξοδο από το plant. Για να εκτιμηθεί η χρησιμότητα των συγκεκριμένων-νευρωνικών μεθόδων στη διαμόρφωση, αποφασίστηκε να δουλέψουμε με δύο διαφορετικά σήματα σαν εισόδους στην κατασκευή μας,

- i) μοναδιαία βηματική είσοδος
- ii) ψευδό τυχαία δυαδική ακολουθία (PRBS)

καθώς κάποιος θα περίμενε ότι η έξοδος του συστήματος μας για μία PRBS θα ήταν πιο περίπλοκη από αυτή για ένα μοναδιαία βηματική είσοδο.

Το ANFIS προσφέρει έναν τρόπο δημιουργίας ενός συγκεκριμένου-νευρωνικού μοντέλου για ένα σύστημα, στην περίπτωση που οι εισοδοί και οι έξοδοί του είναι γνωστές, σύμφωνα με τον τρόπο που συζητήθηκε πριν. Τώρα ο χρήστης χρειάζεται να δημιουργήσει δύο σετ δεδομένων, ένα για εκπαίδευση και ένα για έλεγχο. Το ANFIS χρησιμοποιεί τα δεδομένα εκπαίδευσης για να δημιουργήσει ένα αρχικό συγκεκριμένο-νευρωνικό μοντέλο, και τα δεδομένα ελέγχου για να επικυρώσει την απόδοση του μοντέλου. Μόλις δημιουργηθεί μία συγκεκριμένη συμπερασματική δομή (FIS), ο αλγόριθμος πίσω διάδοσης ή αυτός της υβριδικής εκμάθησης μπορούν να χρησιμοποιηθούν για την εκπαίδευση του δικτύου. Αυτή η διαδικασία (διεργασία) αποδεικνύεται στη συνέχεια, για αμφότερες τη μοναδιαία βηματική είσοδο και την ψευδό τυχαία δυαδική ακολουθία.

3.1 Διαμόρφωση του $G(s)$ για μοναδιαία βηματική είσοδο

Όπως αναφέρθηκε παραπάνω, το πρώτο βήμα στην διαδικασία είναι να δημιουργηθεί ένα σύνολο δεδομένων εκμάθησης και ελέγχου. Για να χρησιμοποιήσουμε με το ANFIS ένα σύνολο δεδομένων, πρέπει αυτό να έχει τη μορφή μίας μήτρας-πίνακα, της οποίας η πρώτη στήλη αντιπροσωπεύει την πραγματική είσοδο στο σύστημα και η τελευταία στήλη την έξοδο. Το $G(s)$ επιλέχθηκε να είναι:

$$(1/(s+2))*(1/(s^2+1.5s+1))$$

αφού αυτό το σύστημα είναι, πράγματι, κατασκευασμένο από ελαφρά αποσβενόμενο στοιχείο δευτέρας τάξεως μαζί με μία ρύθμιση πρώτης τάξης.

Ο ακόλουθος κώδικας του Matlab χρησιμοποιήθηκε για να δημιουργήσει τα εκπαιδευτικά δεδομένα:

- 1) $s = tf('s');$
- 2) $H = (1/(s+2))*(1/(s^2+1.5*s+1));$
- 3) $Ts = 0.6;$
- 4) $T = 0:Ts:15;$

- 5) $Y = \text{step}(H,T);$
- 6) $N = 0.02*(\text{rand}(\text{size}(Y))-0.5);$
- 7) $\text{trd}(:,2) = Y+N;$
- 8) $\text{NI} = 0.02*(\text{rand}(\text{size}(Y))-0.5);$
- 9) $\text{trd}(:,1) = 1+\text{NI};$

Η πρώτη γραμμή ορίζει το 's' ως την ισοδύναμη συνάρτηση μεταφοράς.

Η δεύτερη γραμμή ορίζει το 'H' στη συνάρτηση μεταφοράς $(1/(s+2))*(1/(s^2+1.5*s+1))$

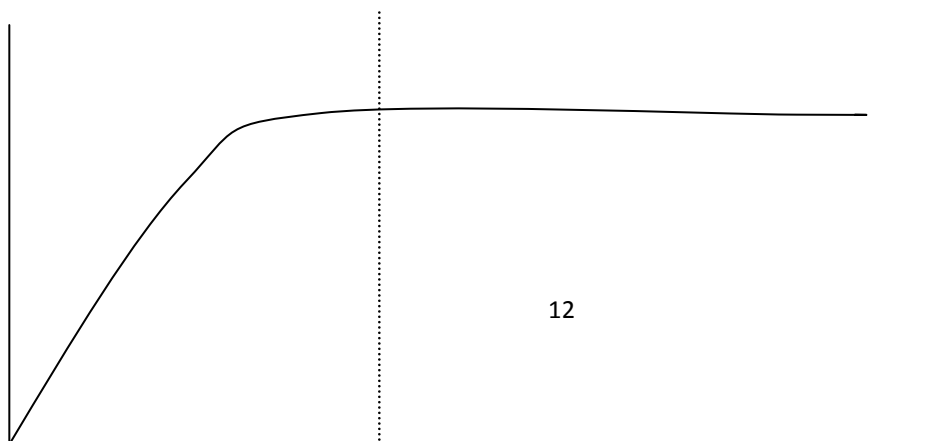
Η τρίτη γραμμή ορίζει το χρόνο δειγματοληψίας.

Η γραμμή τέσσερα ορίζει το διάστημα, όπου τα δείγματα πρέπει να λαμβάνονται. Μεταβάλλοντας το χρόνο δειγματοληψίας μπορούμε να αποφασίσουμε σχετικά με τον αριθμό των δειγμάτων για τη δημιουργία.

Το 'Y' στην πέμπτη γραμμή είναι η έξοδος του plant για μία μοναδιαία βηματική είσοδο. Η 'βηματική' συνάρτηση δημιουργεί την έξοδο ενός plant για ένα προκαθορισμένο χρονικό διάστημα.

Οι γραμμές έξι και οκτώ δημιουργούν σήματα θορύβου, τα οποία εφαρμόζονται στην έξοδο και την είσοδο του plant στις γραμμές επτά και εννιά αντίστοιχα.

Ο κώδικας (το σύνολο των οδηγιών προγραμματισμού) του Matlab δημιουργούν μία μήτρα-πίνακα δύο στηλών. Η μία στήλη αντιπροσωπεύει την είσοδο στο plant και η δεύτερη την έξοδο. Αυτή η μήτρα μπορεί να χωριστεί σε δύο μισά, με το ένα μισό να αποτελεί τα εκπαιδευτικά δεδομένα και το άλλο τα δεδομένα ελέγχου. Αυτή είναι η μέθοδος που ακολουθείται συνήθως, αλλά όχι σε αυτή την περίπτωση! Όπως μπορεί να φανεί στο Σχήμα 3, η έξοδος του $G(s)$ για μία μοναδιαία βηματική είσοδο, μοιάζει με μία ραγδαία (εκθετική) αύξηση. Συνεπώς, αν είχαμε χρησιμοποιήσει το πρώτο μισό για εκπαίδευση και το δεύτερο μισό για έλεγχο, θα υπήρχε μεγάλη διαφορά μεταξύ των δεδομένων και το συμπερασματικό μοντέλο δεν θα ήταν το επιθυμητό.



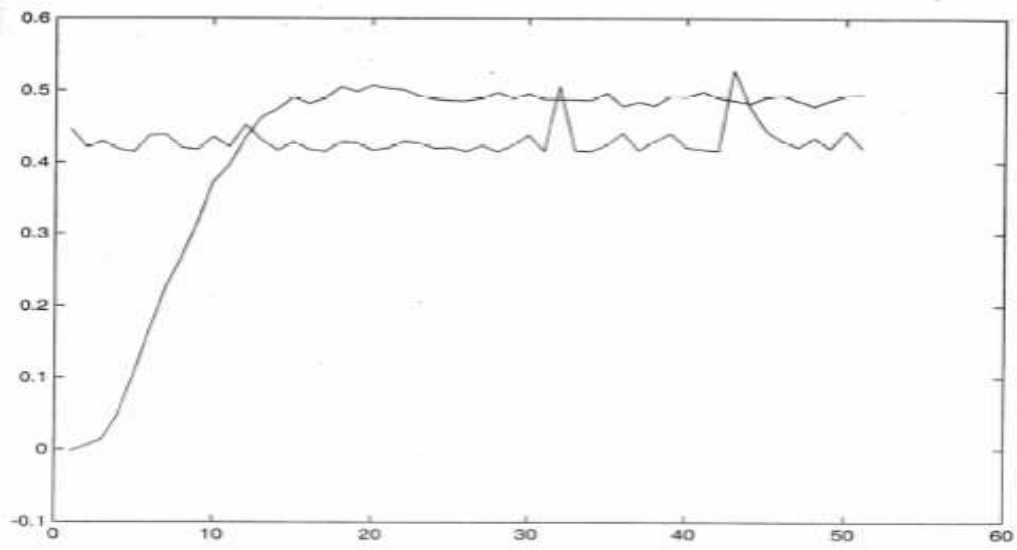
Σχήμα 3: Έξοδος του $G(s)$ για μία μοναδιαία βηματική είσοδο.

Αντί αυτού, αποφασίστηκε να αλλαχθεί ελαφρώς μία από τις παραμέτρους του $G(s)$, η χρονική καθυστέρηση στη ρύθμιση πρώτης τάξης, με ένα τέτοιο τρόπο ώστε, η έξοδος θα παραμείνει σχεδόν η ίδια. Αυτό έγινε, αντικαθιστώντας τη ρύθμιση πρώτης τάξης $1/(s+2)$ με το $1/(s+2.05)$ στον κώδικα του Matlab, για να δημιουργήσουμε ένα δεύτερο σύνολο δεδομένων. Το πρώτο σύνολο χρησιμοποιήθηκε ως τα δεδομένα εκπαίδευσης και το δεύτερο σύνολο ως τα δεδομένα ελέγχου.

Αυτά τα δύο σύνολα δεδομένων χρησιμοποιήθηκαν με το ANFIS για να βελτιώσουν το επιθυμητό συγκεκριμένο-νευρωνικό μοντέλο. Για άλλη μια φορά, η διαδικασία διαμόρφωσης ήταν επιτυχής. Όπως φαίνεται στο Γράφημα 1, η έξοδος του συγκεκριμένου-νευρωνικού μοντέλου δεν είναι κοντά στην έξοδο του plant. Ο λόγος είναι ότι το αρχικό μοντέλο, που δημιουργήθηκε από το ANFIS πριν τη νευρωνική προσαρμογή, δεν ήταν αρκετά καλό.

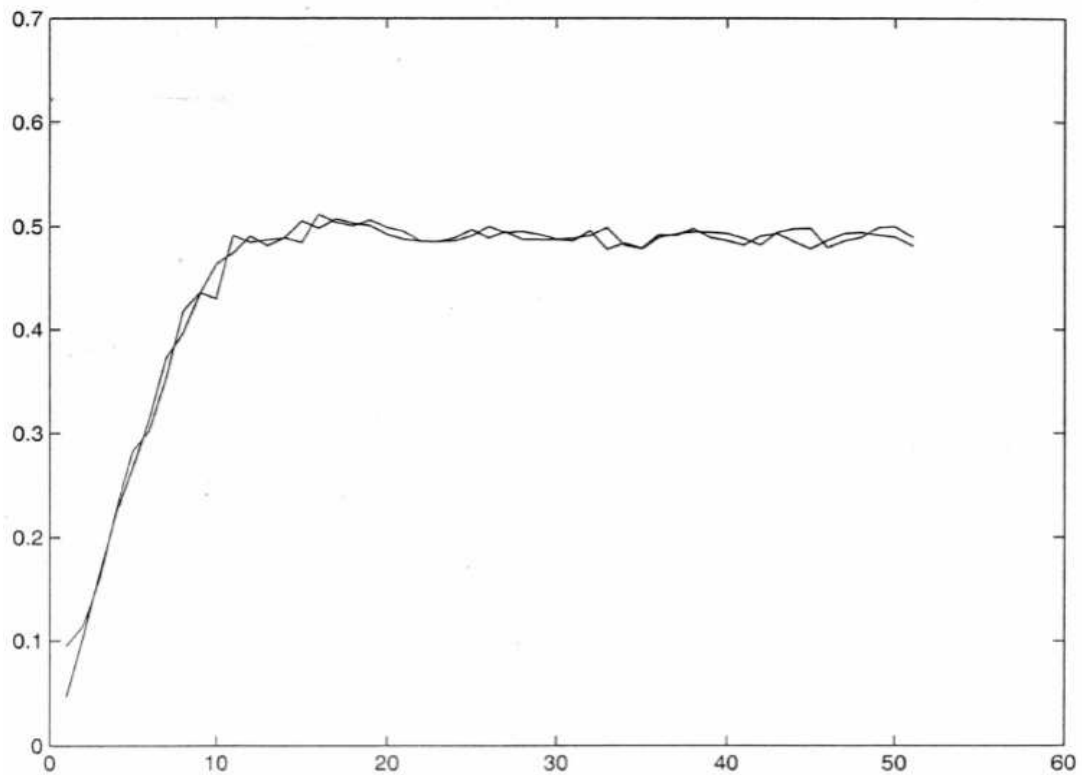
Για να αντιμετωπιστεί αυτό το πρόβλημα ήταν απαραίτητο να βελτιωθούν τα δεδομένα εκπαίδευσης, έτσι ώστε να δημιουργηθεί ένα καλύτερο αρχικό συγκεκριμένο-νευρωνικό μοντέλο, πριν εφαρμοστεί νευρωνική προσαρμογή. Μετά από διάφορα πειράματα έγινε αντιληπτό ότι, η αναδρομή (recursion) ήταν η απάντηση στο πρόβλημά μας. Βελτιώνοντας τους πίνακες των δεδομένων εκπαίδευσης και ελέγχου με τρεις επιπλέον στήλες, αναπαριστώντας $Y(n-3)$, $Y(n-2)$ και $Y(n-1)$ αντίστοιχα, δηλαδή εισάγοντας αναδρομή (recursion) στο σύστημά μας, μπορούσαμε να αναπτύξουμε ένα πολύ καλό αρχικό συγκεκριμένο-νευρωνικό μοντέλο, του οποίου η έξοδος ήταν πολύ κοντά στην έξοδο του plant, ακόμα και πριν από νευρωνική προσαρμογή. Αυτό φαίνεται ξεκάθαρα στο Γράφημα 2.

Τα πραγματικά δεδομένα εκπαίδευσης και ελέγχου παρουσιάζονται στα Παραρτήματα 1 και 2.



1. Responde after neural adaptation.

-Απόκριση μετά από νευρωνική προσαρμογή



1. Responde before neural adaptation.

-Απόκριση πριν από νευρωνική προσαρμογή

Οι επόμενες δύο γραμμές σχεδιάζουν τις συμμετοχικές συναρτήσεις του συγκεκριμένου νευρωνικού μοντέλου πριν τη νευρωνική προσαρμογή. Αυτές οι συμμετοχικές συναρτήσεις παρουσιάζονται στο Γράφημα 3.

Η τέταρτη γραμμή του προγράμματος εκτιμά την έξοδο πριν τη νευρωνική προσαρμογή, και οι γραμμές έξι και επτά σχεδιάζουν την έξοδο, όπως φαίνεται στο Γράφημα 2.

Η γραμμή πέντε υπολογίζει τη μέση τετραγωνική ρίζα σφάλματος πριν τη νευρωνική προσαρμογή και την βρίσκει ίση με 0.0123.

Η εκπαίδευση του δικτύου εκτελείται χάρη στη συνάρτηση 'anfis' στη γραμμή οκτώ. Το Fismat1 είναι το επακόλουθο συγκεκριμένο-νευρωνικό μοντέλο, που προκύπτει χρησιμοποιώντας τα δεδομένα εκπαίδευσης, και το fismat2, αυτό που εξάγεται χρησιμοποιώντας τα δεδομένα ελέγχου.

Η γραμμή εντολής εννιά δοκιμάζει το συγκεκριμένο-νευρωνικό μοντέλο 'fismat2' με τις εισόδους από τα δεδομένα ελέγχου, με το να παράγει την έξοδο του μοντέλου. Οι γραμμές 10 και 11 μας επιτρέπουν να συγκρίνουμε ορατά, την έξοδο του συγκεκριμένου-νευρωνικού μοντέλου, μετά τη νευρωνική προσαρμογή, με την πραγματική του $G(s)$. Αυτό παρουσιάζεται στο Γράφημα 4.

Οι γραμμές εντολών 12, 14 και 15 κάνουν το ίδιο, μόνο που αυτή τη φορά το συγκεκριμένο νευρωνικό μοντέλο 'fismat1' χρησιμοποιείται με τις εισόδους από τα δεδομένα εκπαίδευσης. Το αποτέλεσμα παρουσιάζεται στο Γράφημα 5.

Η γραμμή 13 υπολογίζει τη μέση τετραγωνική ρίζα μετά από νευρωνική προσαρμογή, η οποία σε αυτή την περίπτωση ισούται με 0,0091, μετά από δύο χρονικές περιόδους (εποχές) στην εκπαίδευση.

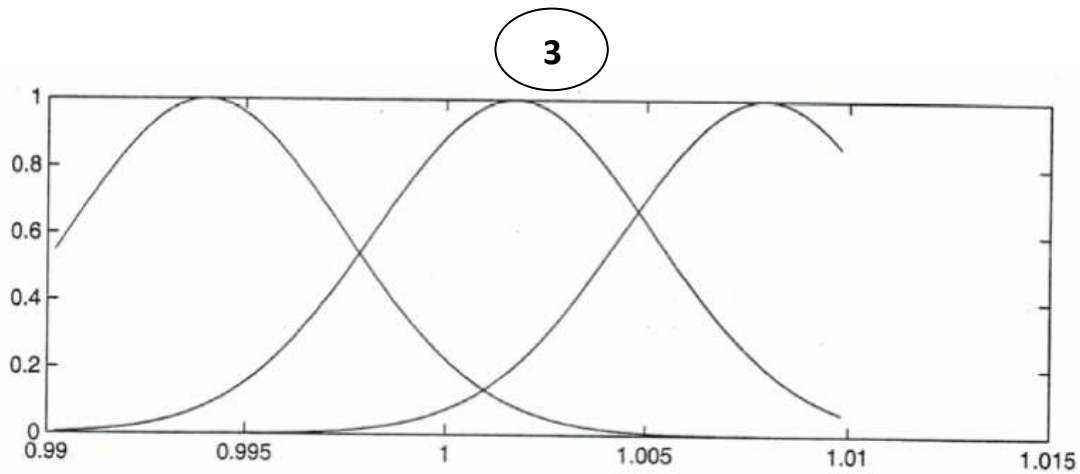
Το επόμενο βήμα είναι να εκτελέσουμε νευρωνική προσαρμογή στο συγκεκριμένο-νευρωνικό μας δίκτυο. Για να γίνει αυτό, χρησιμοποιήσαμε τον υβριδικό αλγόριθμο εκμάθησης, αντί του αλγόριθμου πίσω διάδοσης, αφού αυτός συγκλίνει ταχύτερα, αφού μειώνει τον αριθμό των διαστάσεων του διαστήματος αναζήτησης της μεθόδου πίσω διάδοσης.

Ο κώδικας του Matlab, που ακολουθεί, δημιουργεί ένα αρχικό συγκεκριμένο-νευρωνικό μοντέλο και μετά εκπαιδεύει το δίκτυο για να βελτιώσει την απόδοσή του.

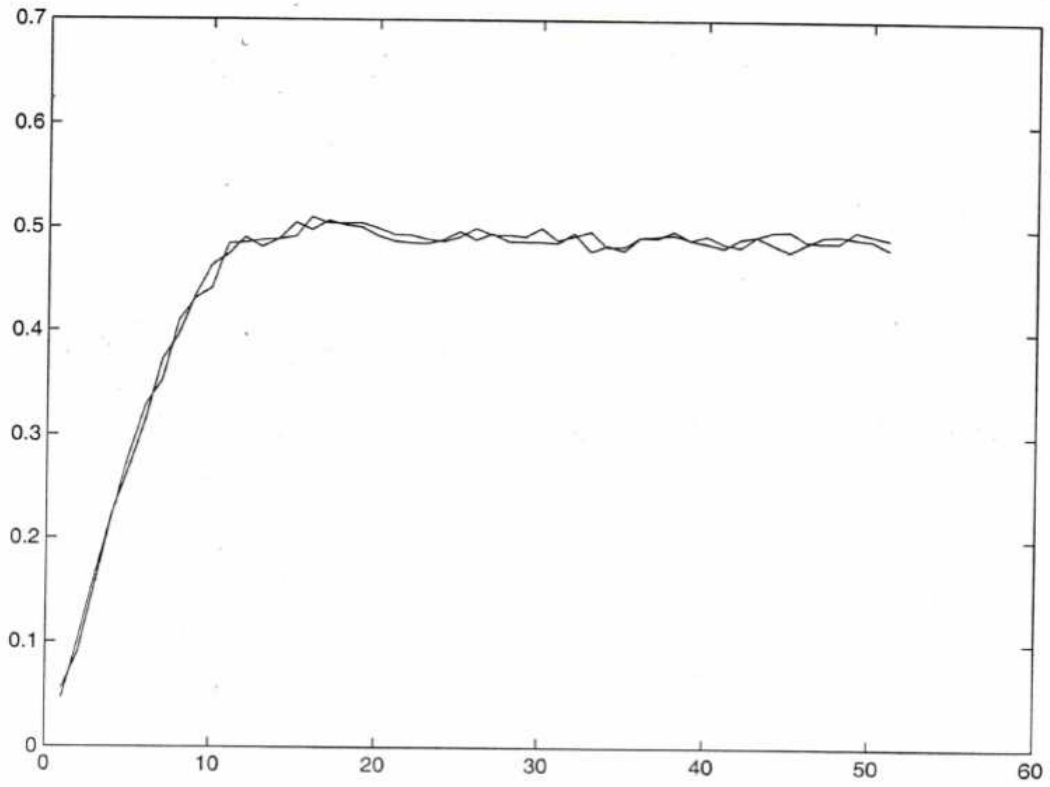
- 1) sysgen=genfis2 (model13trd(:,1:4), model13trd(:, 5), 0.5) ;
- 2) [x,mf]=plotmf(sysgen,'input',1) ;
- 3) subplot (2,1,1), plot(x,mf) ;
- 4) outresp=evalfis(model13chd(:,1:4),sysgen) ;
- 5) rmsebna=norm(outresp-model13chd(:,5))/sqrt(length(outresp)) ;
- 6) figure(2)
- 7) plot([model13chd(:,5) outresp]) ;

- 8) `[fismati,error1,stepsize,fismat2,error2]=anfis(model13trd,sysgen,[200 0 0.01 0.9 1.1] , [],model13chd) ;`
- 9) `outputtrd=evalfis(modelchd(: , 1:4)fismat2) ;`
- 10) `figure(3)`
- 11) `plot ([model13chd(: , 5) outputchd])`
- 12) `outputtrd=evalfis(model13trd (: , 1:4), fismat1) ;`
- 13) `rmseana = norm (outputchd-model13chd(:,5)) / sqrt (length (outputchd)) ;`
- 14) `figure (4)`
- 15) `Plot ([model13trd (: , 5) outputtrd])`

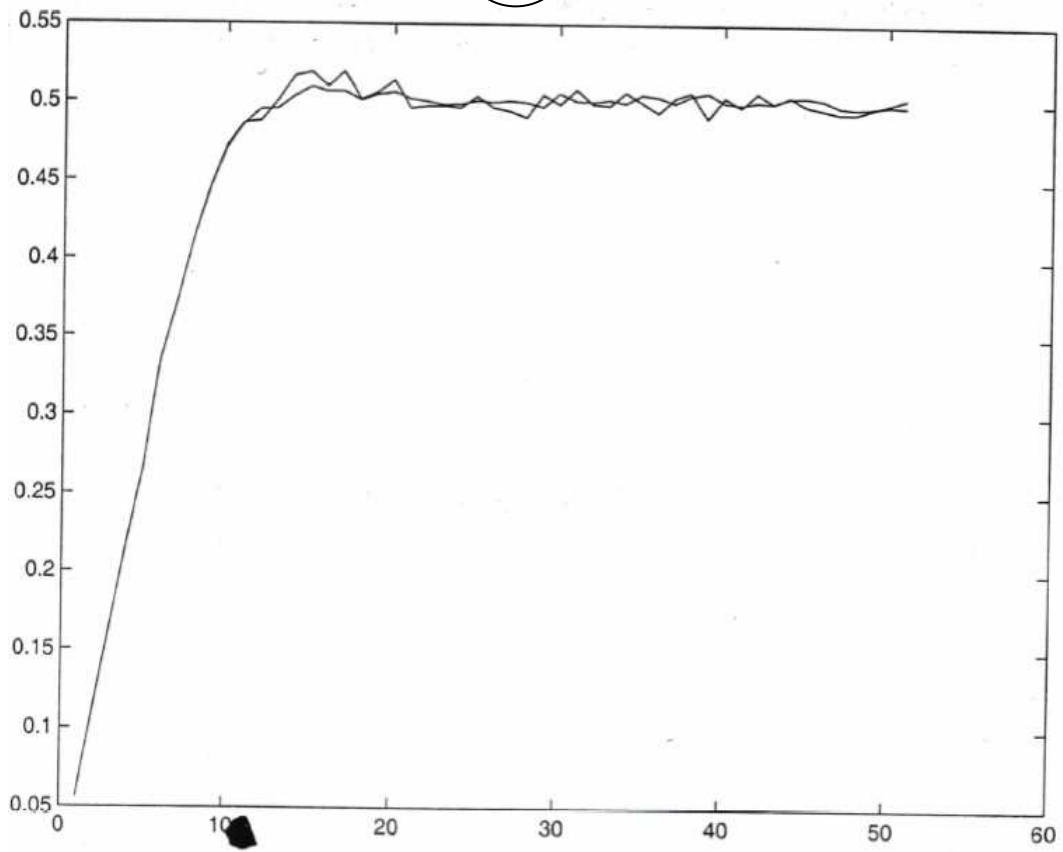
Η πρώτη γραμμή του προγράμματος δημιουργεί το αρχικό συγκεχυμένο-νευρωνικό μοντέλο 'sysgen', χρησιμοποιώντας την συνάρτηση `genfis 2` του Matlab και τα εκπαιδευτικά δεδομένα 'model13trd'.



4



5



Δουλεύοντας από την εργαλειοθήκη του ANFIS, μπορέσαμε να καταστρώσουμε τα δεδομένα εκπαίδευσης, τα δεδομένα ελέγχου και τα δεδομένα εκπαίδευσης και ελέγχου μαζί, όπως φαίνεται στα Γραφήματα 6, 7 και 8 αντίστοιχα.

Μόλις εισαχθούν τα δύο σύνολα δεδομένων, το ANFIS μας επιτρέπει να δημιουργήσουμε ένα αρχικό συγκεχυμένο συμπερασματικό σύστημα, το οποίο στην περίπτωση μας, είναι αυτό που παρουσιάζεται στο Γράφημα 9.

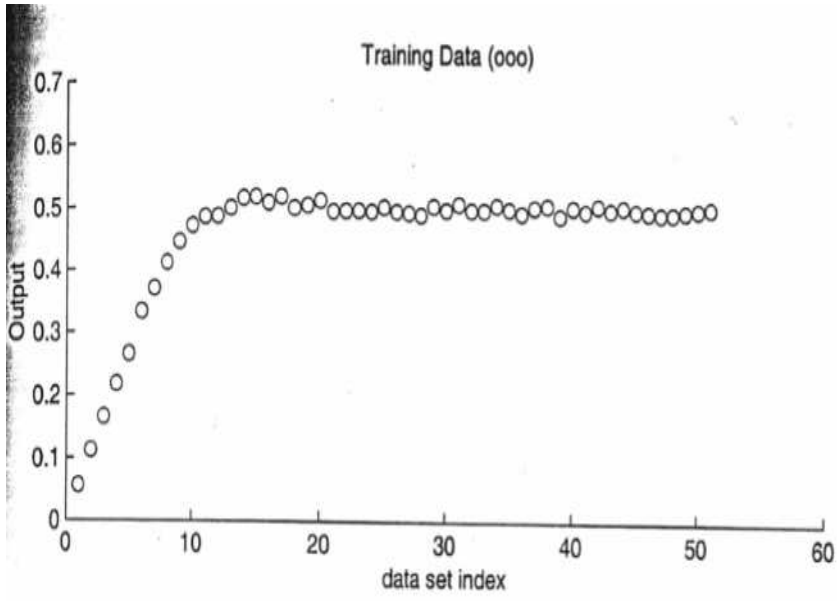
Οι μαύροι κύκλοι στο επίπεδο 1 είναι οι τέσσερις εισοδοί του plant. Η είσοδος 1 είναι η πραγματική είσοδος, ενώ οι εισοδοί δύο, τρία και τέσσερα είναι οι προηγούμενες εξοδοί $Y(n-1)$, $Y(n-2)$ και $Y(n-3)$ του plant, που χρησιμοποιούνται ως αναδρομικές (recursive) εισοδοί.

Στο επίπεδο 2 υπάρχουν τέσσερις ομάδες των τριών νευρώνων η καθεμία, με τον κάθε νευρώνα σε κάθε ομάδα να αντιπροσωπεύει μία από τις τρεις συμμετοχικές συναρτήσεις, ορίζοντας τις εισόδους στο plant.

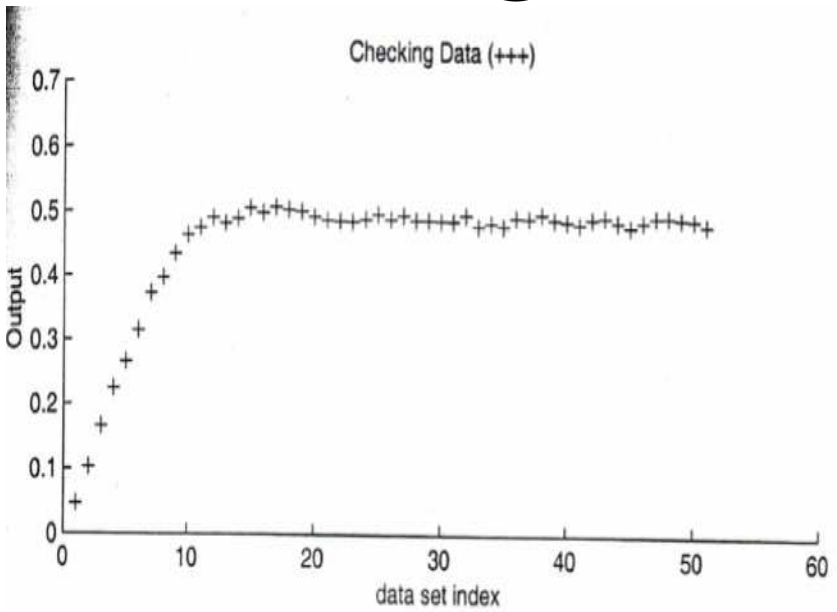
Κάθε ένας από τους 81 νευρώνες στο επίπεδο 3 αντιστοιχεί σε καθένα από τους 81 κανόνες, που ορίστηκαν για το σύστημα μας, αντίστοιχα. Ο κανόνας “νούμερο ένα” αντιπροσωπεύεται από το νευρώνα στο χαμηλότερο άκρο του επιπέδου 3.

Ο κόκκινος κύκλος στο άκρο του επιπέδου 4 είναι ένας παράγοντας εξομάλυνσης, που χρησιμοποιείται για να σταθμίσει κάθε κανόνα στο επίπεδο 3. Στο επίπεδο 4 υπάρχουν 81 νευρώνες επιπλέον, αντιπροσωπεύοντας τις συμμετοχικές συναρτήσεις των εξόδων του συστήματός μας.

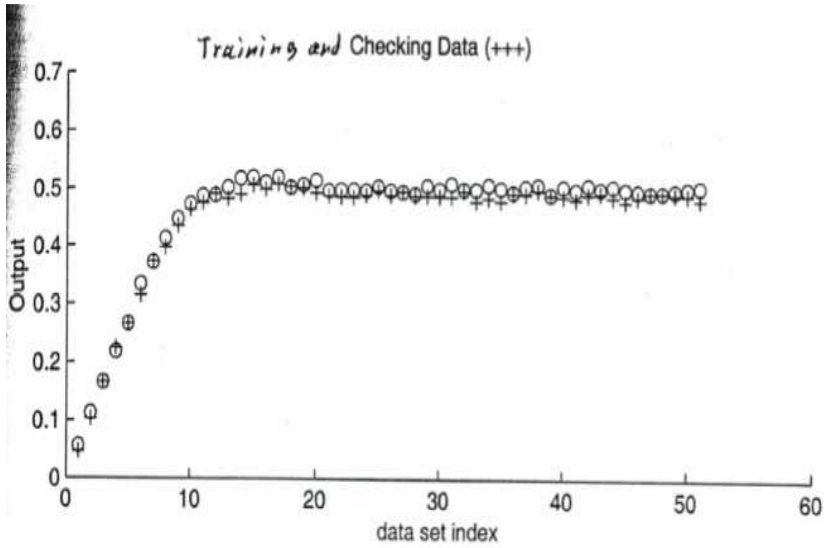
Ένα σταθμισμένο άθροισμα της εξόδου από κάθε μία από τις συμμετοχικές συναρτήσεις υπολογίζεται από το νευρώνα στο επίπεδο 5. Η πραγματική έξοδος προέρχεται από το νευρώνα στο επίπεδο 6, ο οποίος εξομαλύνει την έξοδο από το νευρώνα στο επίπεδο 5.



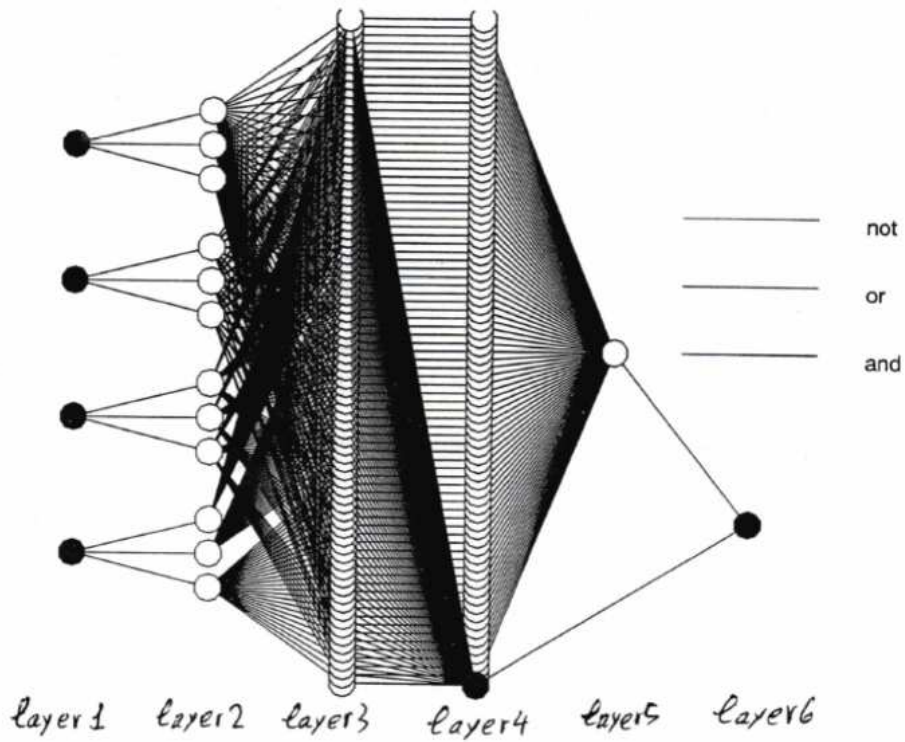
7



8



9



Close

Click on each node to see detailed information

Then we need to choose the training algorithm, the hybrid learning algorithm in our case, to train our network. On graph paper 10, the dots are showing the checking error reducing over the number of training epochs, whilst the stars represent the training error over the same number of epochs. From the graph it appears that training is complete after three epochs, i.e. the error has reached a minimum value.

After training we test our data to see how close is the output of the fuzzy-neural model to the output of the plant. Graph papers 11 and 12 show the outputs of the fuzzy-neural model along with the output of the plant if the training data or the checking data are used to test the fuzzy-neural model, respectively.

A complete description of the fuzzy inference system is given in appendix 3, where one can study the inputs to and the output from the system, along with all the membership functions and rules.

Graph paper 13a shows what the membership functions of the first input of the fuzzy neural model have become after neural adaptation.

The non-linearity of our system is clearly shown in graph paper 14 where on the surface viewer the output of our fuzzy-neural model is plot against two of the inputs.

Στη συνέχεια πρέπει να επιλεγεί ο αλγόριθμος εκπαίδευσης, στην περίπτωσή μας ο υβριδικός αλγόριθμος εκμάθησης, για να εκπαιδευτεί το δίκτυό μας. Στο Γράφημα 10, οι τελείες παρουσιάζουν το σφάλμα ελέγχου, μειώνοντας επί του αριθμού των εκπαιδευτικών χρονικών περιόδων, ενώ τα αστέρια αντιπροσωπεύουν το εκπαιδευτικό σφάλμα επί του ίδιου αριθμού των χρονικών περιόδων (εποχών). Από το Γράφημα φαίνεται ότι, η εκπαίδευση ολοκληρώνεται μετά από 3 χρονικές περιόδους-εποχές, δηλαδή το σφάλμα έχει φτάσει σε μία κατώτατη τιμή.

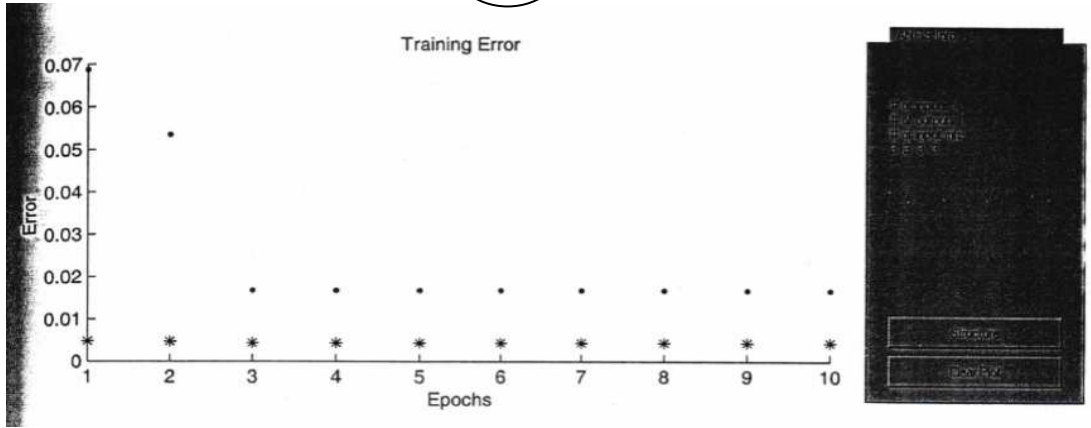
Μετά την εκπαίδευση δοκιμάζουμε τα δεδομένα μας, για να δούμε πόσο κοντά είναι η έξοδος του συγκεκριμένου-νευρωνικού μοντέλου στην έξοδο του plant. Τα Γραφήματα 11-12 παρουσιάζουν τις εξόδους του συγκεκριμένου-νευρωνικού μοντέλου, μαζί με την έξοδο του plant, αν τα εκπαιδευτικά δεδομένα ή τα δεδομένα ελέγχου χρησιμοποιούνται για να ελέγξουν το συγκεκριμένο-νευρωνικό μοντέλο, αντίστοιχα.

Μία πλήρης περιγραφή του συγκεκριμένου συμπερασματικού συστήματος δίνεται στο Παράρτημα 3, όπου κάποιος μπορεί να μελετήσει τις εισόδους προς και την έξοδο από το σύστημα, μαζί με όλες τις συμμετοχικές συναρτήσεις και τους κανόνες.

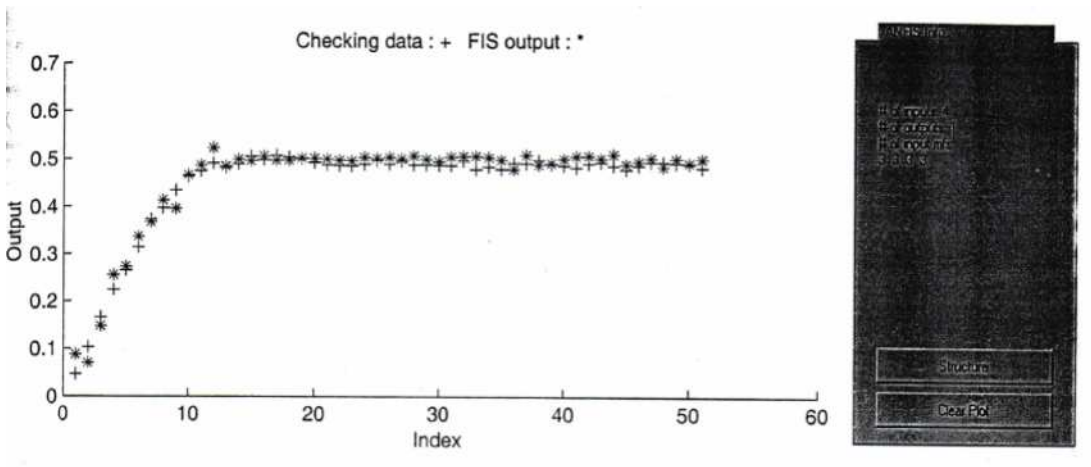
Το Γράφημα 13a παρουσιάζει το πώς έχουν καταλήξει οι συμμετοχικές συναρτήσεις της πρώτης εισόδου του συγκεκριμένου-νευρωνικού μοντέλου μετά τη νευρωνική προσαρμογή.

Η μη γραμμικότητα του συστήματός μας φαίνεται ξεκάθαρα στο Γράφημα 14, όπου στην “επιφάνεια του θεατή” (surface viewer), η έξοδος του συγκεκριμένου-νευρωνικού μοντέλου σχεδιάζεται εναντίον δύο εκ των εισόδων.

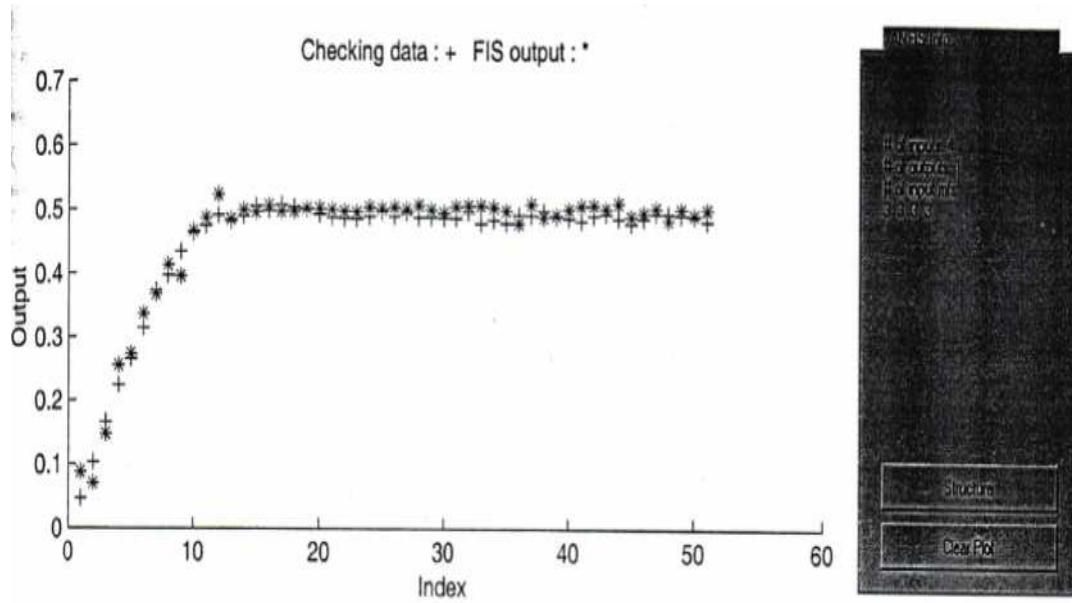
10



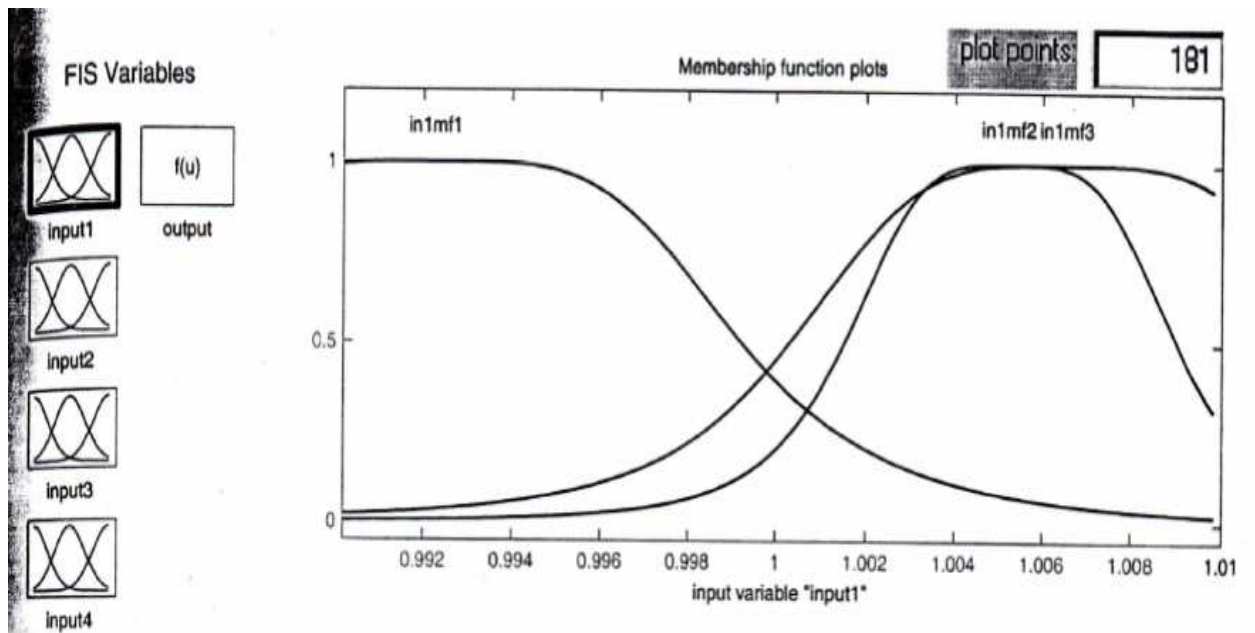
11

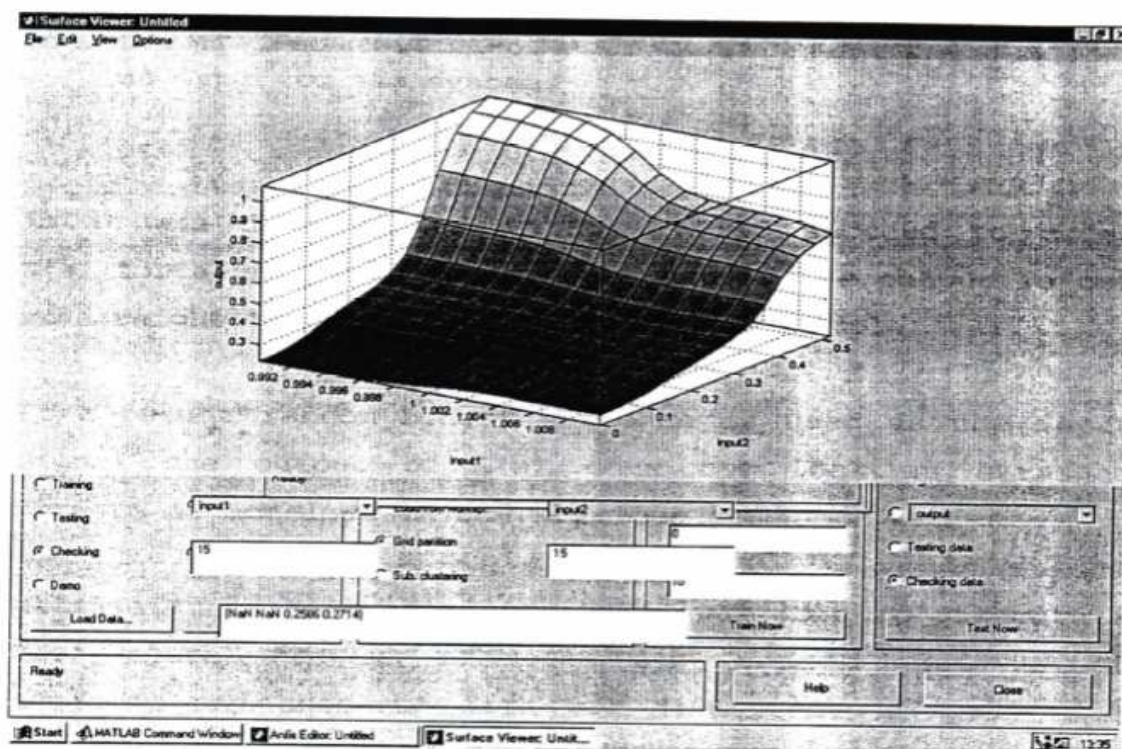


12



13





3.2 Διαμορφώση του $G(s)$ με χρήση μίας ψευδό τυχαίας δυαδικής ακολουθία σαν είσοδο στο σύστημα

Όπως αποδείχθηκε στην παράγραφο 3.1, το ANFIS επέτυχε να διαμορφώσει το $G(s)$ για μία μοναδιαία βηματική είσοδο, αλλά τι γίνεται στην περίπτωση που η έξοδος δεν είναι κατευθείαν εμπρόσθια, όπως στο προηγούμενο παράδειγμα.

Αν μία ψευδό τυχαία δυαδική ακολουθία χρησιμοποιείται σαν σήμα εισόδου, η έξοδος του $G(s)$ δεν φαίνεται τόσο ευθεία στο μοντέλο, όπως στην προηγούμενη περίπτωση.

Άλλη μία φορά, ακόμα και σε αυτή την περίπτωση, το ANFIS μπορούσε να δημιουργήσει ένα συγκεκριμένο-νευρωνικό δίκτυο για να διαμορφώσει το $G(s)$, μόλις αναπτύχθηκε ένα καλό σύνολο δεδομένων εκπαίδευσης και ελέγχου. Ένα “καλό σύνολο δεδομένων” είναι ένα, το οποίο, όταν χρησιμοποιείται με το ANFIS, μπορεί να οδηγήσει στην δημιουργία ενός συγκεκριμένου συμπερασματικού συστήματος, το οποίο μπορεί να παράγει μία έξοδο κοντά στην επιθυμητή, πριν εφαρμοστεί κάποια εκπαιδευτική διαδικασία.

Ο ακόλουθος κώδικας του Matlab δημιουργεί μία ψευδό τυχαία δυαδική ακολουθία:

- 1) $s=tf('s');$
- 2) $H = (1/(s-2)) * (1/(s^2+1.5*s+1));$
- 3) $indata = (1/2) * (mlbs(9) + 1);$
- 4) $t=0 : 1 : \text{length}(indata) - 1;$
- 5) $[YS, TS] = \text{lsim}(H, indata, t);$
- 6) $\text{lsim}(H, indata, t);$
- 7) $data = [indata YS];$

Η πρώτη γραμμή του προγράμματος ορίζει τη συνάρτηση μεταφοράς 's', ενώ στη δεύτερη, η συνάρτηση μεταφοράς $(1/(s+2)) * (1/(s^2+1.5s+1))$ αντιπροσωπεύεται από την 'H'. Η γραμμή τρία δημιουργεί την PBRs, ενώ η γραμμή τέσσερα προσδιορίζει το χρονικό διάστημα και το χρόνο δειγματοληψίας. Η έξοδος του G(s) για μία είσοδο PBRs υπολογίζεται στην πέμπτη γραμμή και σχεδιάζεται από τη γραμμή έξι. Μία μήτρα δύο στηλών δημιουργείται στην έβδομη γραμμή, της οποίας η πρώτη στήλη είναι η είσοδος στο plant, και η δεύτερη στήλη η έξοδος.

Με σκοπό να δημιουργήσουμε τα δεδομένα εκπαίδευσης και τα δεδομένα ελέγχου, η μήτρα χωρίζεται σε δύο μισά, με το πρώτο μισό να μπορεί να χρησιμοποιηθεί για τα δεδομένα εκπαίδευσης και το δεύτερο μισό για τα δεδομένα ελέγχου.

Παρομοίως, αν και στην προηγούμενη περίπτωση, τα δεδομένα μας δεν μπορούν να δημιουργήσουν ένα ικανοποιητικό αρχικό μοντέλο, το οποίο θα παράγει μία έξοδο κοντά στην επιθυμητή, και σαν αποτέλεσμα το ANFIS δεν μπορεί να εκπαιδεύσει το δίκτυο επιτυχώς, ώστε να διαμορφώσει το G(s). Κατά συνέπεια, πρέπει να αναζητήσουμε για ένα κατάλληλο σύνολο δεδομένων εκπαίδευσης και ελέγχου, το οποίο θα βελτιώσει το αρχικό μας μοντέλο. Πάλι παρατηρήθηκε ότι, η αναδρομή (recursion) βελτιώνει σημαντικά τα δεδομένα μας. Όπως φαίνεται στα Παραρτήματα 5 και 6, για τα δεδομένα εκπαίδευσης και τα δεδομένα ελέγχου, αντίστοιχα, οι προηγούμενες τρεις έξοδοι του συγκροτήματος χρησιμοποιήθηκαν σαν είσοδοι αναδρομής (recursion). Όπως θα αποδειχθεί στη συνέχεια, αυτές οι ομάδες δεδομένων χρησιμοποιήθηκαν επιτυχώς με το ANFIS στη διαμόρφωση του G(s).

Ο ακόλουθος κώδικας του Matlab κατάδεικνύει πώς να διαμορφώσετε το G(s) από τη γραμμή εντολών:

- 1) `sysgen=genfis2 (prbs23trd (: , 1 :4) , prbs23trd (: ,5), 0.5) ;`
- 2) `[x,mf]=plotmf (sysgen, 'input' , 1) ;`
- 3) `subplot (2 , 1 , 1) , plot (x,mf) ;`
- 4) `outresp=evalfis (prbs23chd (: , 1 : 4) , sysgen) ;`
- 5) `rmsebna=norm (outresp-prbs23chd (: , 5)) / sqrt (length (outresp)) ;`
- 6) `figure (2)`
- 7) `plot ([prbs23chd (: ,5) outresp])`
- 8) `[fismat1,error1,stepsize,fismat2,error2] = anfis (prbs23trd,sysgen, [200 0 0.01 0.9 1.1] , [] , prbs23chd) ;`
- 9) `outputchd=evalfis(prbs23chd (: , 1 : 4) , fismat2) ;`
- 10) `rmse(outputchd-prbs23chd (: ,5)) /sqrt (length(outputchd)) ;`
- 11) `figure (3)`
- 12) `plot ([prbs23chd(:) ,5) outputchd`
- 13) `outputtrd=evalfis (prbs23trd(: , 1 : 4) , fismat1) ;`
- 14) `figure (4)`
- 15) `plot ([prbs23trd(: , 5) outputtrd])`

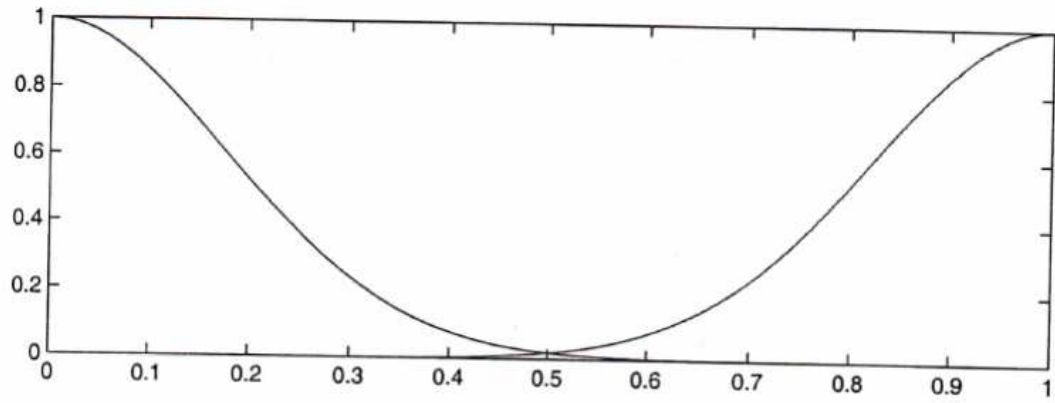
Η γραμμή 1 δημιουργεί ένα συγκεκριμένο-νευρωνικό μοντέλο και το χαρακτηρίζει ως 'sysgen'. Οι επόμενες δύο γραμμές παράγουν και σχεδιάζουν τις συμμετοχικές συναρτήσεις του συστήματος πριν την εκπαίδευση. Αυτές οι συμμετοχικές συναρτήσεις παρουσιάζονται στ Γράφημα 15.

Η γραμμή τέσσερα εκτιμά την έξοδο του συγκεκριμένου-νευρωτικού μοντέλου και τη σχεδιάζει μαζί με την έξοδο του plant.

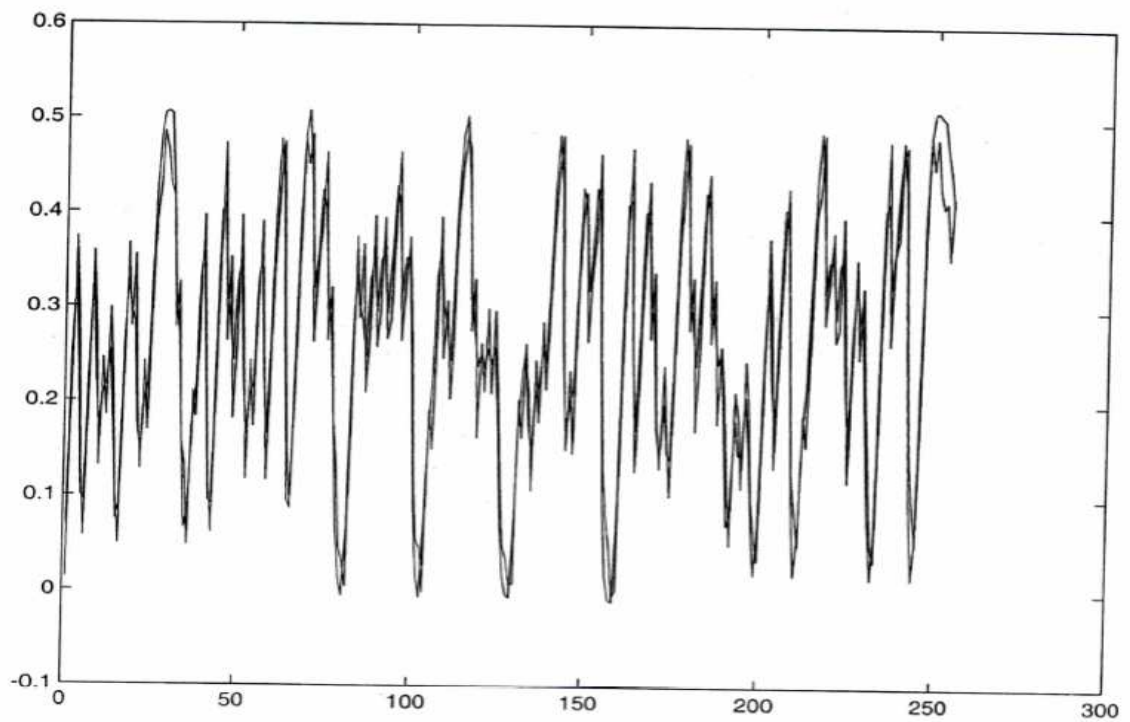
Η μέση τετραγωνική ρίζα σφάλματος υπολογίζεται στην γραμμή 5, και η οποία σε αυτό το παράδειγμα είναι 0,0441.

Οι γραμμές 5 και 6 καταστρώνουν την έξοδο του συγκεκριμένου-νευρωνικού μοντέλου, πριν τη νευρωνική προσαρμογή, μαζί με την πραγματική είσοδο του plant, όπως φαίνεται στο Γράφημα 16. Η εκπαίδευση του συγκεκριμένου-νευρωνικού

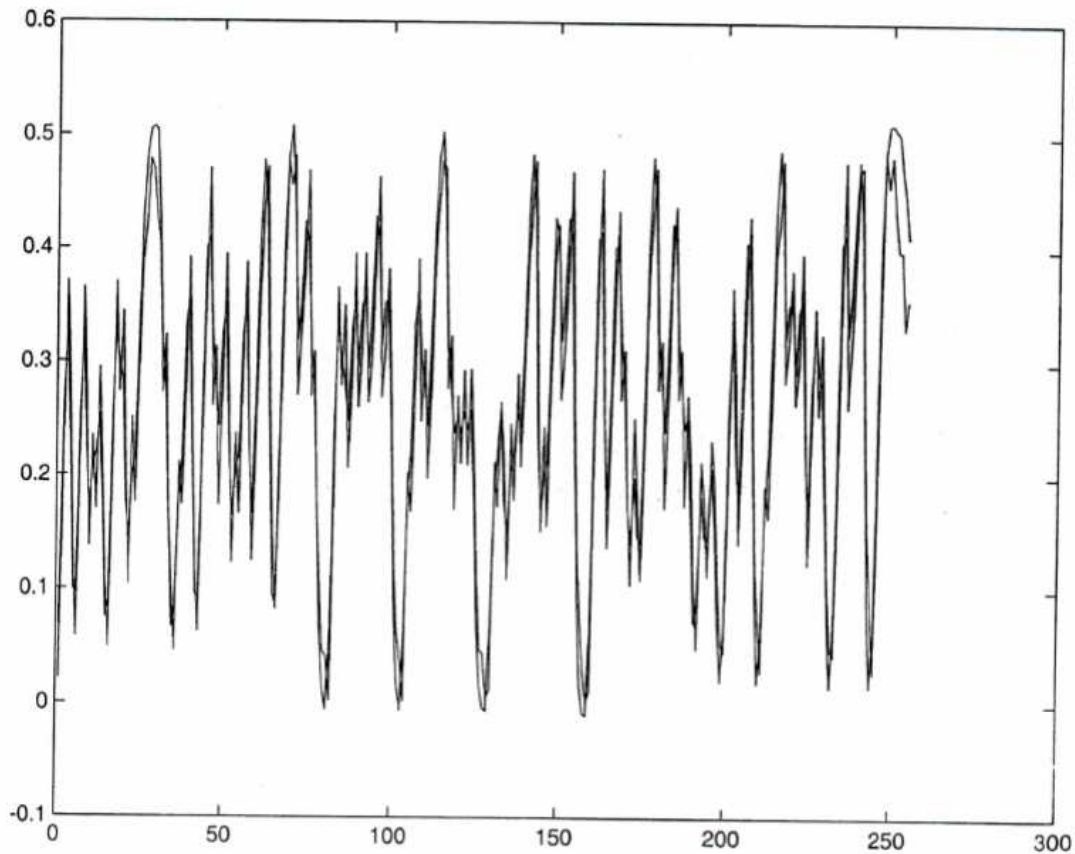
δικτύου πραγματοποιείται στη γραμμή 8, χρησιμοποιώντας το θεμέλιο στην εντολή 'anfis' του Matlab.



16



17



The output of the fuzzy-neural model after training is evaluated in line nine, using the inputs of the checking data.

The root mean square error after neural adaptation is computed in line ten to be 0.0430 after 200 training epochs.

Lines eleven and twelve plot the output of the fuzzyneural model after neural adaptation along with the actual output of the plant, when the inputs of the checking data is used. This is shown in graph paper 17.

Evaluation of the output using the inputs of the trainingdata is performed in line thirteen, whilst lines fourteen and fifteen plot the outcome along with the actual output of the plant. This is shown in graph paper 18.

Working from the ANFIS toolbox the training and checking data must first be entered and then ANFIS will generate an initial fuzzy inference system once we have decided on the number and type of the membership functions. Graph papers 19, 20 and 21 show the training, checking and training and checking data respectively, as plotted by ANFIS.

Η έξοδος του συγκεκριμένου-νευρωτικού μοντέλου, μετά την εκπαίδευση, υπολογίζεται στη γραμμή 9, χρησιμοποιώντας τις εισόδους των δεδομένων ελέγχου.

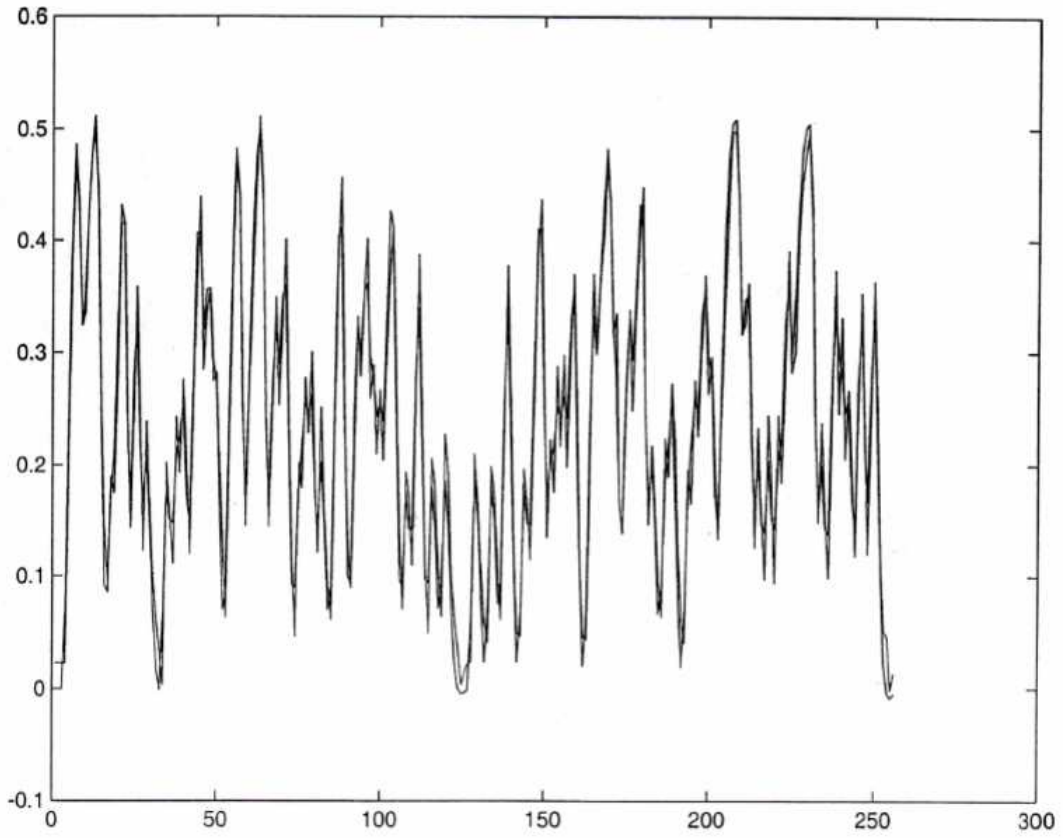
Η μέση τετραγωνική ρίζα σφάλματος μετά τη νευρωνική προσαρμογή υπολογίζεται στη γραμμή 10 και ισούται με 0.0430 μετά από 200 εκπαιδευτικές χρονικές περιόδους (εποχές).

Οι γραμμές 11 και 12 σχεδιάζουν την έξοδο του συγκεκριμένου-νευρωτικού μοντέλου μετά τη νευρωνική προσαρμογή, μαζί με την πραγματική έξοδο του plant, όταν χρησιμοποιούνται οι εισοδοί των δεδομένων ελέγχου. Αυτό παρουσιάζεται στο Γράφημα 17.

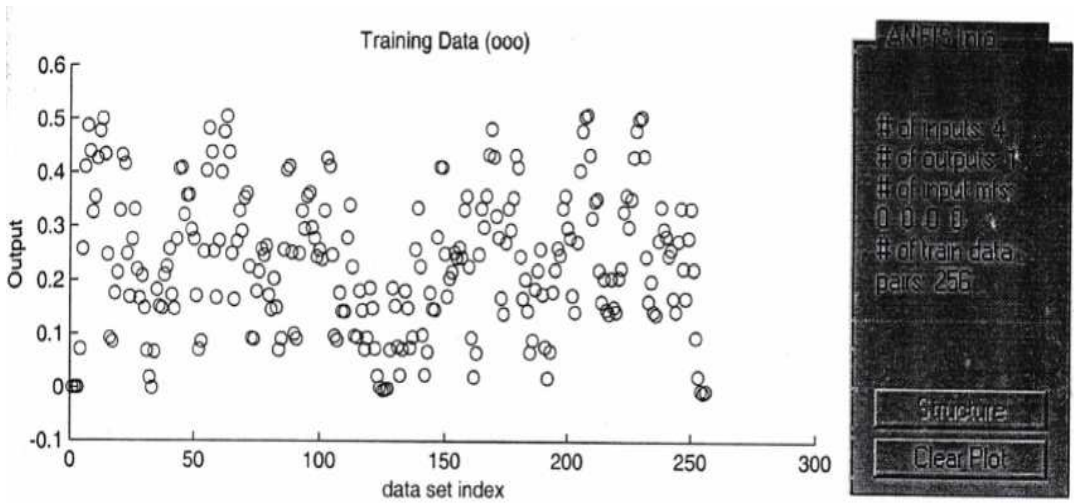
Η εκτίμηση της εξόδου, χρησιμοποιώντας τις εισόδους των δεδομένων εκπαίδευσης, πραγματοποιείται στη γραμμή 13, ενώ οι γραμμές 14 και 15 σχεδιάζουν την έξοδο μαζί με την πραγματική έξοδο του plant. Αυτό παρουσιάζεται στο Γράφημα 18.

Δουλεύοντας από την εργαλειοθήκη του ANFIS, τα δεδομένα εκπαίδευσης και ελέγχου πρέπει πρώτα να εισαχθούν και μετά, μόλις έχουμε αποφασίσει για τον αριθμό και τον τύπο των συμμετοχικών συναρτήσεων, το ANFIS θα δημιουργήσει ένα αρχικό συγκεκριμένο συμπερασματικό σύστημα. Τα Γραφήματα 19, 20 και 21 παρουσιάζουν τα δεδομένα ελέγχου, τα δεδομένα εκπαίδευσης και τα δεδομένα ελέγχου και εκπαίδευσης, αντίστοιχα, όπως σχεδιάστηκαν από το ANFIS.

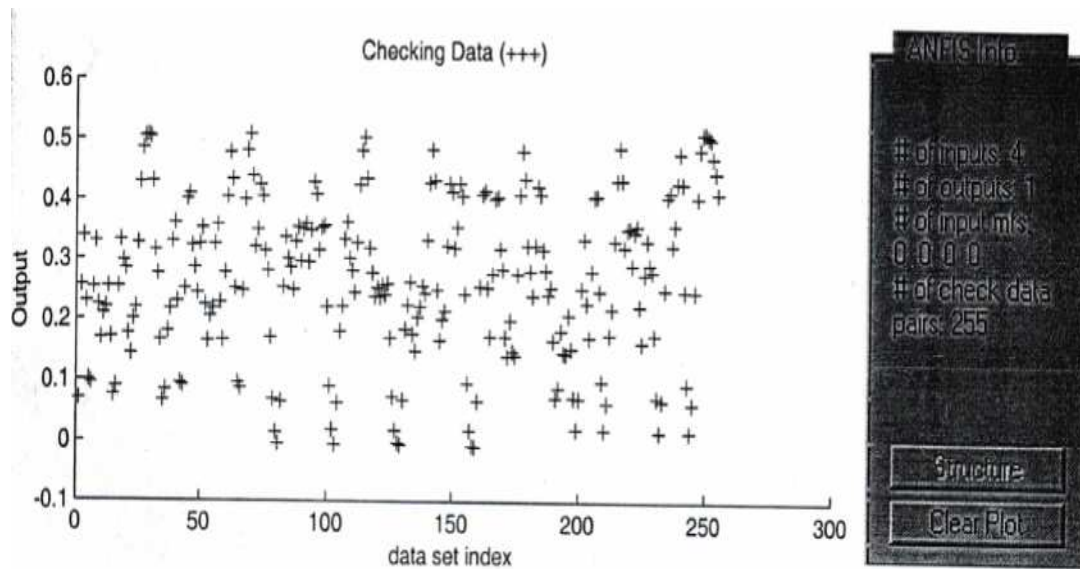
Για το ίδιο λόγο, όπως κι όταν δουλεύαμε με τη μοναδιαία βηματική είσοδο, ο υβριδικός αλγόριθμος εκμάθησης επιλέγεται για την εκπαίδευση του συγκεκριμένου-νευρωτικού δικτύου. Το Γράφημα 22 παρουσιάζει το σφάλμα ελέγχου, σχεδιασμένο με τελείες και το σφάλμα εκπαίδευσης, σχεδιασμένο με αστέρια, συναρτήσεϊ των εκπαιδευτικών χρονικών περιόδων. Όπως παρουσιάζεται στο Γράφημα, η εκπαίδευση ολοκληρώνεται μετά από 34 εκπαιδευτικές περιόδους (εποχές), όταν το σφάλμα ελέγχου φτάνει μία ελάχιστη τιμή.



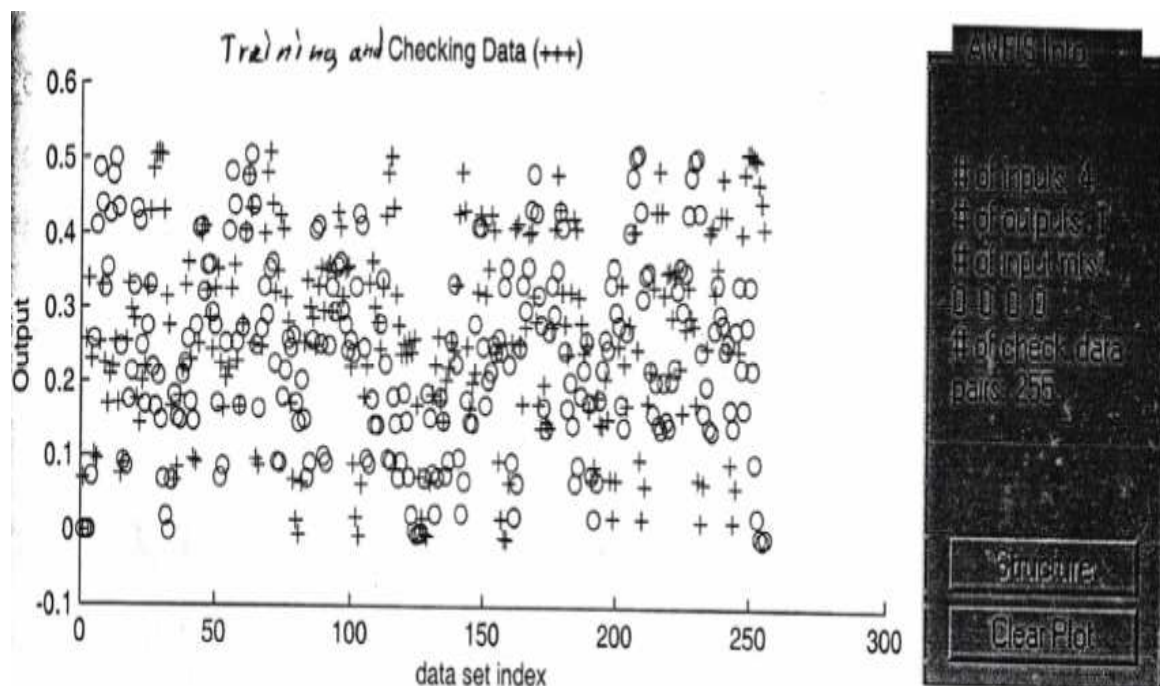
19



20



21



In graph paper 23 one can see the membership functions of input one after neural adaptation.

Στο Γράφημα 23, κάποιος μπορεί να δει τις συμμετοχικές συναρτήσεις της εισόδου, μετά τη νευρωνική προσαρμογή.

Στο Γράφημα 25 παρουσιάζεται η έξοδος του συγκεκριμένου-νευρωνικού μοντέλου μαζί με την έξοδο του $G(s)$, όταν το μοντέλο δοκιμάζεται με τα δεδομένα ελέγχου, ενώ στο Γράφημα 26 παρουσιάζεται το ισοδύναμο που προκύπτει, χρησιμοποιώντας τα εκπαιδευτικά δεδομένα.

Στο Γράφημα 24 παρουσιάζεται η δομή του συγκεκριμένου-νευρωνικού μοντέλου.

Το επίπεδο 1 είναι το επίπεδο εισόδου στο δίκτυο, με τον μαύρο κύκλο του χαμηλότερου άκρου να είναι η είσοδος ένα, δηλαδή η πραγματική είσοδος του συστήματος. Οι άλλες τρεις εισοδοί είναι οι τρεις προηγούμενες εξοδοί $Y(n-1)$, $Y(n-2)$ και $Y(n-3)$ του συστήματος.

Το δεύτερο επίπεδο αποτελείται από 4 ζεύγη νευρώνων, με κάθε νευρώνα σε κάθε ζεύγος να αντιπροσωπεύει μία από τις συμμετοχικές συναρτήσεις.

Οι 16 κανόνες που κουμαντάρουν το μοντέλο, αντιπροσωπεύονται από 16 νευρώνες στο επίπεδο 3.

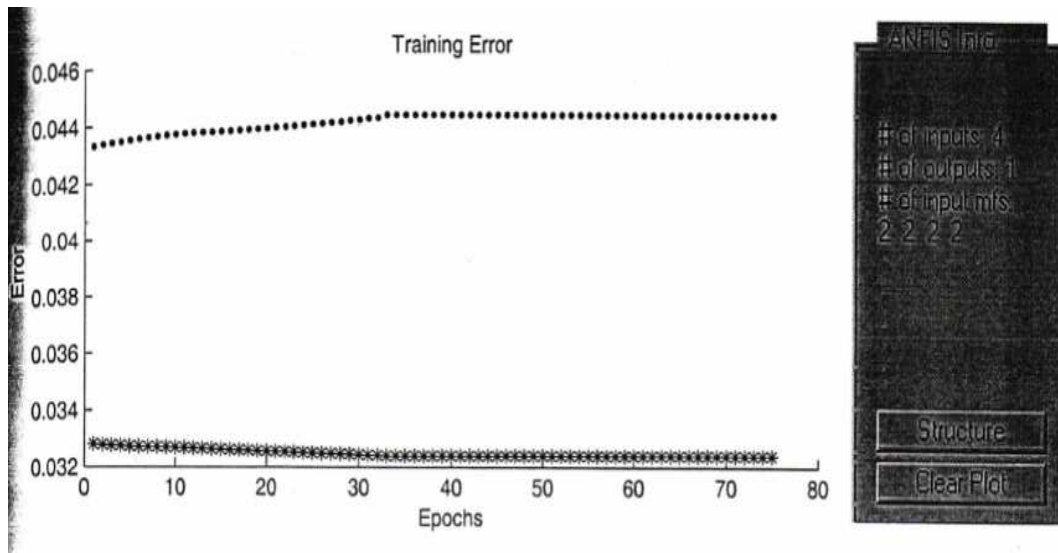
Ο κόκκινος κύκλος στο επίπεδο 4, είναι ένας παράγοντας εξομάλυνσης, που καθορίζει μία κλίση βαρύτητας σε κάθε έναν από τους κανόνες. Οι άλλοι 16 νευρώνες στο επίπεδο 4 μεταφέρουν τις συμμετοχικές συναρτήσεις εξόδου του συστήματος.

Ένα σταθμικό άθροισμα της εξόδου από κάθε μία από τις συμμετοχικές συναρτήσεις, υπολογίζεται από το νευρώνα στο επίπεδο 5.

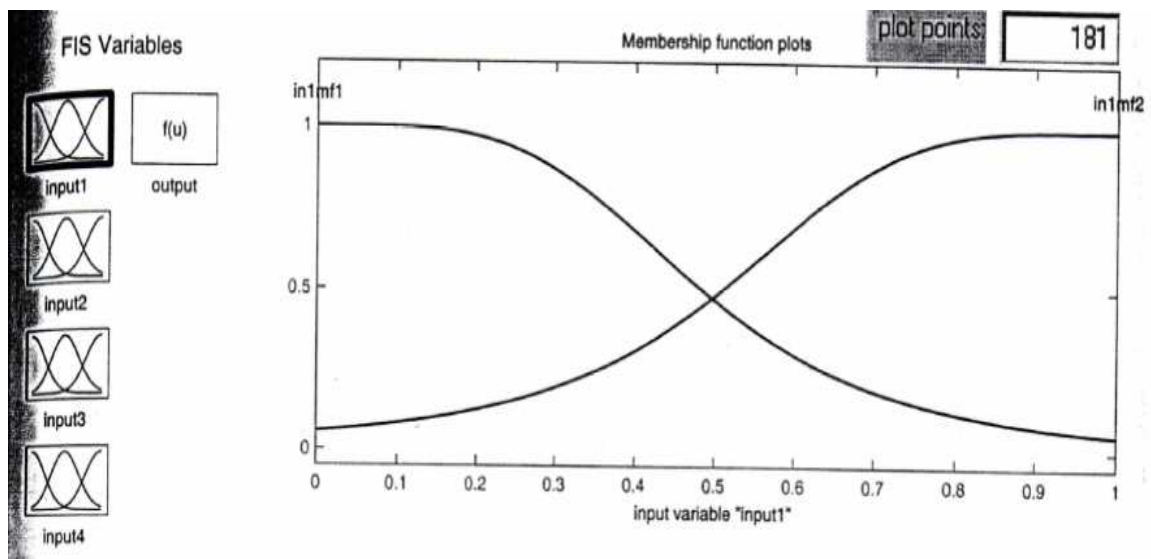
Ο νευρώνας στο επίπεδο 6 εξομαλύνει την έξοδο από το νευρώνα στο επίπεδο 5, για να δώσει την πραγματική έξοδο του συστήματος.

Το Παράρτημα 6 δίνει όλες τις πληροφορίες για το συγκεκριμένο συμπερασματικό σύστημα όπως εισόδους, έξοδο, συμμετοχικές συναρτήσεις και κανόνες.

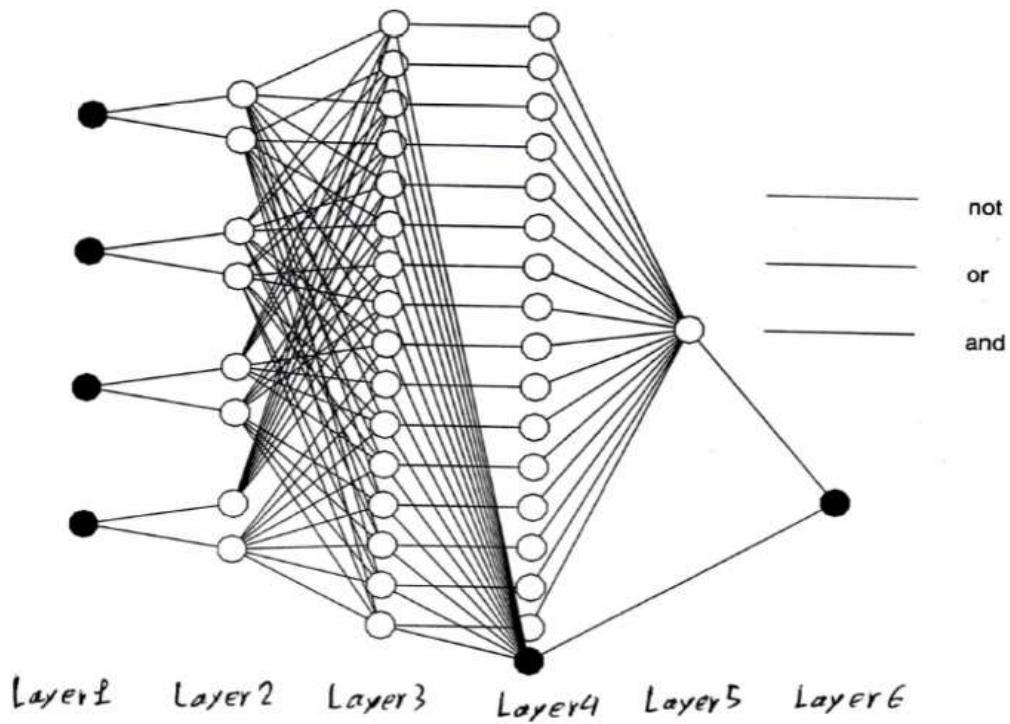
Οι κανόνες που κουμαντάρουν το σύστημα παρουσιάζονται στο Γράφημα 25, ενώ η επιφάνεια στο Γράφημα 26 παρουσιάζει την υψηλή μη γραμμικότητα του συγκεκριμένου-νευρωνικού μοντέλου.



23



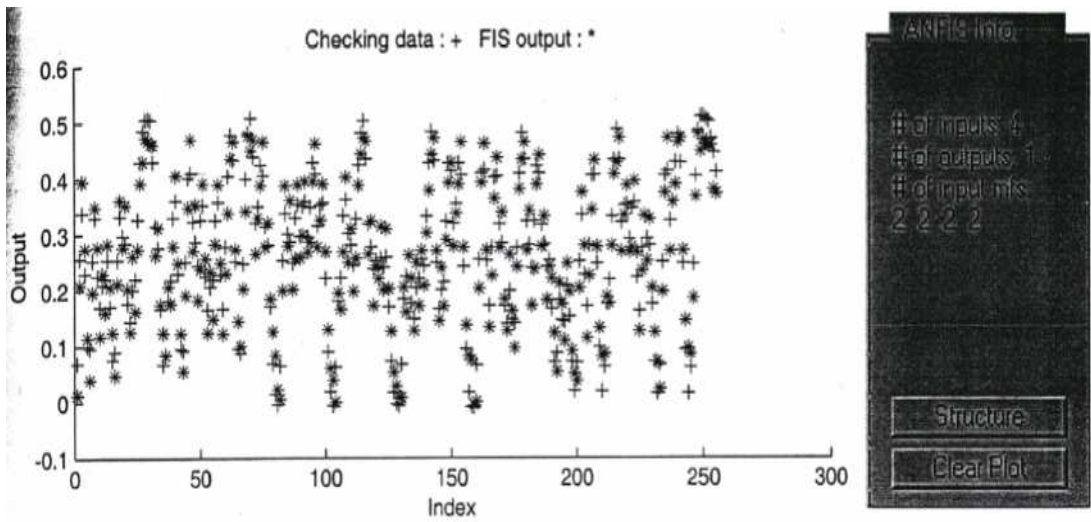
24



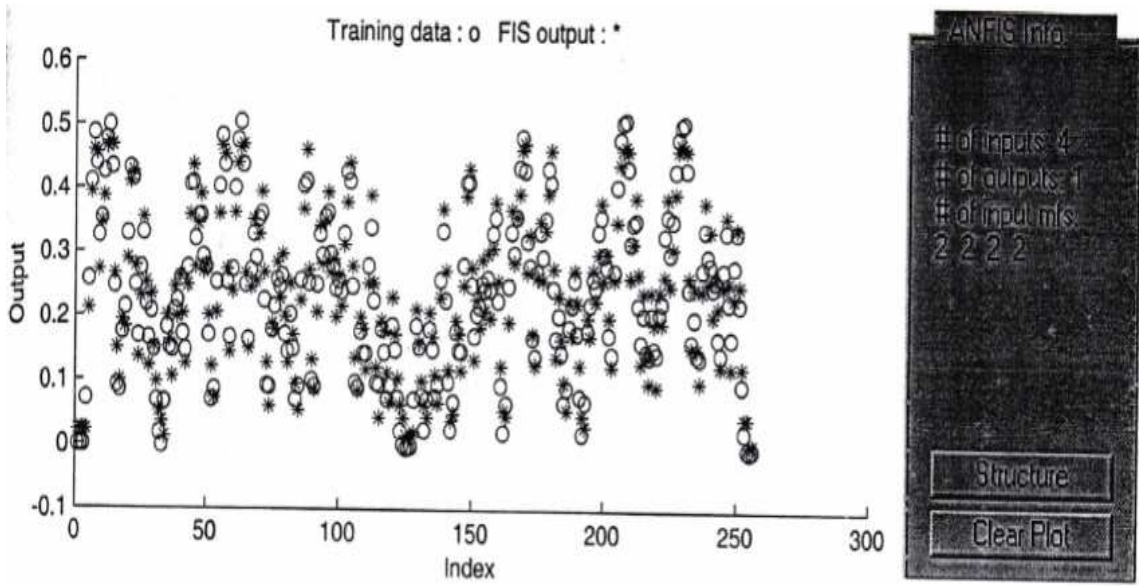
Close

Click on each node to see detailed information

25



26



27

Rule Editor: pbs

File Edit View Options

```

1. If input1 is in1m1 and input2 is in2m1 and input3 is in3m1 and input4 is in4m1 then output is out1m1 (1)
2. If input1 is in1m1 and input2 is in2m1 and input3 is in3m1 and input4 is in4m2 then output is out1m2 (1)
3. If input1 is in1m1 and input2 is in2m1 and input3 is in3m2 and input4 is in4m1 then output is out1m4 (1)
4. If input1 is in1m1 and input2 is in2m1 and input3 is in3m2 and input4 is in4m2 then output is out1m4 (1)
5. If input1 is in1m1 and input2 is in2m2 and input3 is in3m1 and input4 is in4m1 then output is out1m5 (1)
6. If input1 is in1m1 and input2 is in2m2 and input3 is in3m1 and input4 is in4m2 then output is out1m5 (1)
7. If input1 is in1m1 and input2 is in2m2 and input3 is in3m2 and input4 is in4m1 then output is out1m7 (1)
8. If input1 is in1m1 and input2 is in2m2 and input3 is in3m2 and input4 is in4m2 then output is out1m6 (1)
9. If input1 is in1m2 and input2 is in2m1 and input3 is in3m1 and input4 is in4m1 then output is out1m3 (1)
10. If input1 is in1m2 and input2 is in2m1 and input3 is in3m1 and input4 is in4m2 then output is out1m1 (1)
11. If input1 is in1m2 and input2 is in2m1 and input3 is in3m2 and input4 is in4m2 then output is out1m1 (1)
12. If input1 is in1m2 and input2 is in2m1 and input3 is in3m2 and input4 is in4m1 then output is out1m1 (1)
13. If input1 is in1m2 and input2 is in2m1 and input3 is in3m1 and input4 is in4m1 then output is out1m3 (1)
14. If input1 is in1m2 and input2 is in2m2 and input3 is in3m1 and input4 is in4m2 then output is out1m4 (1)
15. If input1 is in1m2 and input2 is in2m2 and input3 is in3m2 and input4 is in4m1 then output is out1m5 (1)
16. If input1 is in1m2 and input2 is in2m2 and input3 is in3m2 and input4 is in4m2 then output is out1m6 (1)

```

Input dropdowns: in1m1, in1m2, none

Output dropdowns: out1m1, out1m2, out1m3, out1m4, out1m5, out1m6, out1m7, out1m8, out1m9, out1m10

Connection: or and

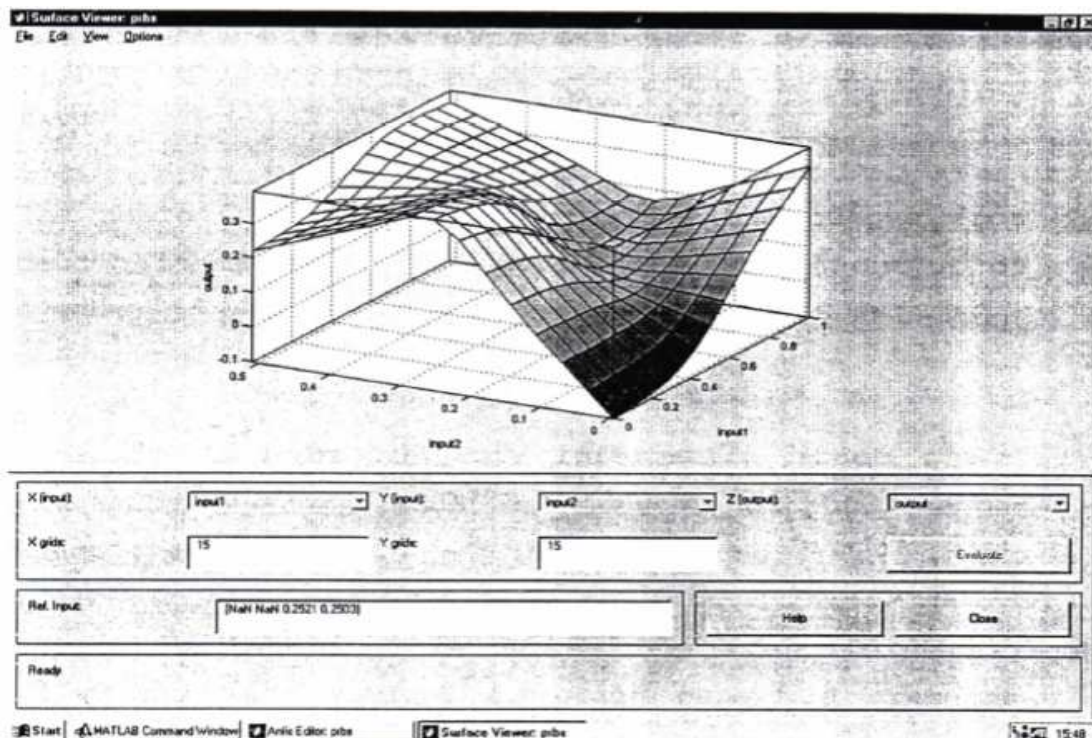
Weight: 1

Buttons: Delete rule, Add rule, Change rule, Help, Close

FIS Name: pbs

Taskbar: Start, MATLAB Command Window, ANFIS Editor: pbs, Rule Editor: pbs, 15:30

28



3.3 Συζήτηση και συμπεράσματα για τη διαμόρφωση

Ένας αριθμός διαφωνιών για τη χρηστότητα του ANFIS, αναφορικά με την διαμόρφωση, έχουν αναπτυχθεί, τη στιγμή που εκτελούνταν τα πειράματα.

Πρώτον, είναι η αδυναμία του ANFIS να διαμορφώσει επιτυχώς ένα σύστημα εάν τα δεδομένα εκπαίδευσης, που χρησιμοποιούνται με το ANFIS, δεν μπορούν να δημιουργήσουν ένα αρχικό μοντέλο, πριν εφαρμοστεί οποιαδήποτε εκπαίδευση, του οποίου η έξοδος τουλάχιστον να μοιάζει με την απαιτούμενη έξοδο. Αυτό σημαίνει ότι, το ANFIS δεν μπορεί να δημιουργήσει ένα συγκεκριμένο-νευρωνικό μοντέλο, εκτός αν το αρχικό μοντέλο προσπαθεί να αντιγράψει το πραγματικό σύστημα.

Η χρηστότητα ενός αρχικού μοντέλου εξαρτάται από τα δεδομένα εκπαίδευσης. Ένας πίνακας δύο στηλών, με τη μία στήλη να είναι η είσοδος του συστήματος και η δεύτερη η έξοδος, δεν είναι αρκετή ώστε να δημιουργήσει ένα καλό αρχικό μοντέλο. Η recursion (αναδρομή) είναι μία μέθοδος που μπορεί να βελτιώσει τα δεδομένα εκπαίδευσης. Αποδείχθηκε στα παραδείγματα παραπάνω ότι, οι τελευταίες τρεις ή τέσσερις προηγούμενες έξοδοι του συστήματος χρησιμοποιήθηκαν σαν αναδρομικές (recursive) είσοδοι για να βελτιώσουν το σύνολο των δεδομένων εκπαίδευσης. Βρέθηκε επίσης, ότι ο αριθμός των δεδομένων επηρεάζει τη χρηστότητα του

μοντέλου. Όσο περισσότερα δεδομένα εκπαίδευσης χρησιμοποιούνται, τόσο καλύτερο είναι το σύστημα.

Επιπλέον, έχει αποδειχθεί ότι, το σφάλμα εκπαίδευσης σταματάει να αυξάνεται μετά από έναν αριθμό εποχών. Αυτό δεν θα ήταν πρόβλημα σε εκείνο το σημείο που το σφάλμα έχει φτάσει μία κατώτατη, ελάχιστη τιμή. Αλλά, είναι συνήθης η περίπτωση που το σφάλμα μπορεί να πέσει σε ένα τοπικό ελάχιστο, από το οποίο είναι δύσκολο να ξεφύγει.

Ένα άλλο θέμα είναι ότι, παρόλο που ο υβριδικός αλγόριθμος εκμάθησης μειώνει τον αριθμό των υπολογισμών, μειώνοντας τις διαστάσεις του διαστήματος αναζήτησης του αυθεντικού αλγόριθμου πίσω διάδοσης, αυτή η μέθοδος ακόμη παραμένει μία έντονη υπολογιστική διαδικασία.

Έχουν αναπτυχθεί κι άλλες μέθοδοι για διαμόρφωση, απαιτώντας λιγότερους υπολογισμούς, χωρίς παραγώγους, χωρίς απαιτήσεις για ιδιαίτερη γνώση και δεξιότητες του συστήματος, μέθοδοι που συγκλίνουν ταχύτερα, παράγουν καλύτερα αποτελέσματα και μειώνουν την πιθανότητα να πέσουν σε τοπικά ελάχιστα. Αυτές οι μέθοδοι συνήθως, είναι ένας συνδυασμός συγκεκριμένης λογικής και γενετικών αλγόριθμων.

4.1 ΣΥΓΚΕΧΥΜΕΝΟ-ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Τα νευρωνικά συστήματα ασχολούνται με τρέχουσα-ενεργή συγκεκριμένη πληροφόρηση και είναι ικανά να παρέχουν καθαρές εξόδους. Εντούτοις, στα συγκεκριμένα συστήματα δεν υπάρχει εκμάθηση και κατά κάποιον ασαφή τρόπο, οι σχέσεις εισόδου-εξόδου και οι συγκεκριμένοι κανόνες πρέπει να είναι γνωστοί από πριν, σαν προγενέστεροι (priori). Τα συγκεκριμένα συστήματα απαιτούν μία πλήρη κατανόηση των συγκεκριμένων μεταβλητών και των συμμετοχικών συναρτήσεων, των σχέσεων εισόδου-εξόδου, όπως και την καλή κρίση να επιλέξεις τους συγκεκριμένους κανόνες που συνεισφέρουν περισσότερο στην επίλυση της εφαρμογής. Για παράδειγμα, αν κάποιος θεωρήσει ένα συγκεκριμένο σύστημα με τρεις εισόδους και μία έξοδο και με 5 συμμετοχικές συναρτήσεις σε κάθε είσοδο, τότε ο συνολικός αριθμός κανόνων είναι 125 και για τέσσερις εισόδους γίνεται 3125. Αυτός είναι ένας μεγάλος αριθμός κανόνων και πολλοί μπορεί να μην συνεισφέρουν σημαντικά στο πρόβλημα. Συνεπώς, η καλή κρίση είναι απαραίτητη για να εξαλειφθούν μη αναγκαίοι και άχρηστοι κανόνες.

Με νευρωνικά δίκτυα μόνα τους, το δίκτυο είναι ένα μαύρο κουτί που χρειάζεται να οριστεί. Αυτό απαιτεί μία πολύ εντατική υπολογιστική διαδικασία. Κάποιος πρέπει να αναπτύξει μία καλή αντίληψη, μετά από εκτενή πειραματισμό και πρακτική, της πολυπλοκότητας του δικτύου και του αλγόριθμου εκμάθησης για να χρησιμοποιηθεί και του βαθμού ακρίβειας αποδεκτού από την εφαρμογή.

Τα νευρωνικά δίκτυα και τα συγκεκριμένα συστήματα, παρόλο που είναι διαφορετικά, έχουν μία στενή σχέση: αμφότερα μπορούν να δουλέψουν με έλλειψη ακρίβειας σε ένα διάστημα που δεν έχει οριστεί με ξεκάθαρα, αιτιοκρατικά όρια. Τα μειονεκτήματα και οι ελλείψεις των νευρωνικών δικτύων μπορούν να ξεπεραστούν, αν ενσωματώσουμε τις λειτουργίες συγκεκριμένης λογικής μέσα στα νευρωνικά δίκτυα και την εκμάθηση και την κωδικοποίηση των νευρωνικών δικτύων μέσα στα συγκεκριμένα συστήματα. Το αποτέλεσμα ονομάζεται συγκεκριμένο τεχνητό νευρωνικό δίκτυο (FANN).

4.2 Συγκεκριμένο τεχνητό νευρωνικό δίκτυο (FANN)

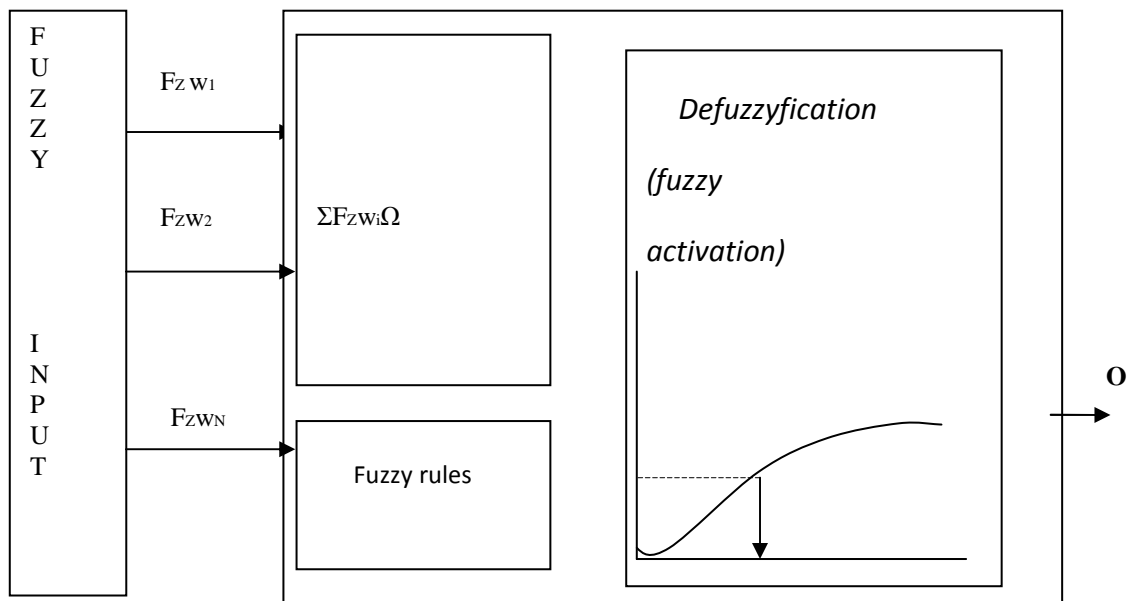
Στο FANN, το τμήμα του νευρωνικού δικτύου χρησιμοποιείται πρωτίστως για τις δυνατότητες εκμάθησης και κωδικοποίησης (ταξινόμησης), που διαθέτει, και ως πρότυπο (υπόδειγμα) συνεργασίας και αποκατάστασης. Το τμήμα του νευρωνικού δικτύου δημιουργεί αυτομάτως κανόνες συγκεκριμένης λογικής και συμμετοχικές συναρτήσεις κατά τη διάρκεια της εκπαιδευτικής περιόδου. Επιπλέον, ακόμα και μετά την εκπαίδευση, το νευρωνικό δίκτυο συνεχίζει την ενημέρωση των συμμετοχικών συναρτήσεων και των κανόνων συγκεκριμένης λογικής, καθώς μαθαίνει όλο και περισσότερο από τα σήματα εισόδου. Από την άλλη μεριά, η συγκεκριμένη λογική χρησιμοποιείται για να συμπεραίνει και να παρέχει μία καθαρή ή μία αποσυγκεκριμένη έξοδο, όταν υπάρχουν συγκεκριμένες παράμετροι.

Ο συγκεκριμένος νευρώνας προσεγγίζει το νευρώνα McCulloch-Pitts. Εντούτοις, υπάρχουν κάποιες βασικές διαφορές: στη θέση μίας μονοδιάστατης στάθμης, ο συγκεκριμένος νευρώνας χρησιμοποιεί συγκεκριμένα σύνολα και επίσης, συγκεκριμένο σύνολο μπορεί να χρησιμοποιηθεί στη θέση της συνάρτησης ενεργοποίησης.

Ένας συγκεκριμένος νευρώνας λειτουργεί ως εξής:

Όταν λαμβάνεται το σήμα S_i , ο συγκεκριμένος κανόνας αθροίζει το elicited μέτρο των συγκεκριμένων βαρών W_i και οι είσοδοι $F_z W_i$ και διαιτητεύονται από ένα παράγοντα εξασθένησης Ω . Το άθροισμα $\sum_{i=1:N} (F_z W_i \Omega)$ αντιπροσωπεύει την αθροιστική ισχύ εισόδου. Αυτό το σήμα στη συνέχεια αποτυπώνεται σε μία ζώνη συγκεκριμένου σετ, αντιστοιχώντας στα επίπεδα πιθανής ενεργοποίησης, που συνδέονται με τον αριθμό ενεργών σημάτων ή την εξασθενημένη ισχύ του αθροιστικού σήματος. Στη συνέχεια, χρησιμοποιώντας μέθοδο αποσυγκεκριμενοποίησης (τεκμηρίωση κέντρου βάρους), η συγκεκριμένη περιοχή υποβαθμίζεται σε μία μονοδιάστατη (μονόμετρη) τιμή, αντιπροσωπεύοντας την αναμενόμενη τιμή της συγκεκριμένης περιοχής υπό τις συνθήκες του σήματος. Αν η ισχύς του σήματος βρίσκεται πάνω από ένα ελάχιστο όριο, τότε ο νευρώνας ξεκινά τη λειτουργία του, αλλιώς όχι. Εντούτοις, αν η συνάρτηση ενεργοποίησης είναι επίσης, ένα συγκεκριμένο σύνολο, τότε η έξοδος που

λαμβάνεται δεν είναι ένα κλασσικό σήμα ενεργοποίησης, αλλά ένα σήμα ποικίλου βαθμού ισχύος. Σε αυτή την περίπτωση το όριο είναι επίσης, συγκεκριμένο.



Σχήμα 4: Γενικό συγκεκριμένο νευρωνικό μοντέλο

4.3 Νευρο-Συγκεκριμένος Έλεγχος

Αυτό που ακολουθεί είναι μία περιγραφή του ελέγχου των συγκεκριμένων, νευρωνικών και συγκεκριμένων-νευρωνικών δικτύων.

4.3.1 Παραδοσιακός Έλεγχος

Εδώ, ο όρος παραδοσιακός έλεγχος σημαίνει τον συνεχή ή ξεχωριστό έλεγχο του συστήματος που δεν έχει ικανότητες εκμάθησης ελέγχου. Γενικά, οι παραδοσιακοί

ελεγκτές κατηγοριοποιούνται σε ανοικτού κυκλώματος και κλειστού κυκλώματος συστήματα. Ο ρόλος των συστημάτων ελέγχου είναι να κάνουν ρυθμίσεις και προσαρμογές, που υποδηλώνονται σαν βέλη στο Σχήμα 5, στην έξοδο ενός plant, όπως αυτό, που η ενεργή έξοδος είναι πολύ κοντά στην επιθυμητή, βασισζόμενα στην ανάγνωση των αισθητήρων, που συγκροτούν την είσοδο του ελεγκτή. Το σύστημα ανοικτού κυκλώματος ελέγχει την είσοδο του plant (Σχήμα 6). Ωστόσο, η έξοδος αναμένεται να είναι ντετερμινιστική (αιτιοκρατική) και σταθερή. Αν η έξοδος του plant παρεκκλίνει από την προκαθορισμένη οδό σε μία απρόβλεπτη, τότε το σύστημα χάνει την ελεγκτική του ικανότητα και γίνεται ασταθές. Η αστάθεια του συστήματος αναγνωρίστηκε γρήγορα και το σύστημα κλειστού κυκλώματος ή το σύστημα ανατροφοδότησης επιλήφθηκε του προβλήματος. Το σύστημα ελέγχου κλειστού κυκλώματος ανατροφοδοτεί την έξοδο του, η οποία επανακαθίσταται από μία συνάρτηση, και η τιθέμενη έξοδος μαζί με την είσοδο χρησιμοποιούνται για να ελέγχουν την πραγματική είσοδο. Ως εκ τούτου, αν η πραγματική έξοδος τείνει να αποκλίνει από την επιθυμητή, ανιχνεύεται και πραγματοποιείται μία διόρθωση στην πραγματική έξοδο.

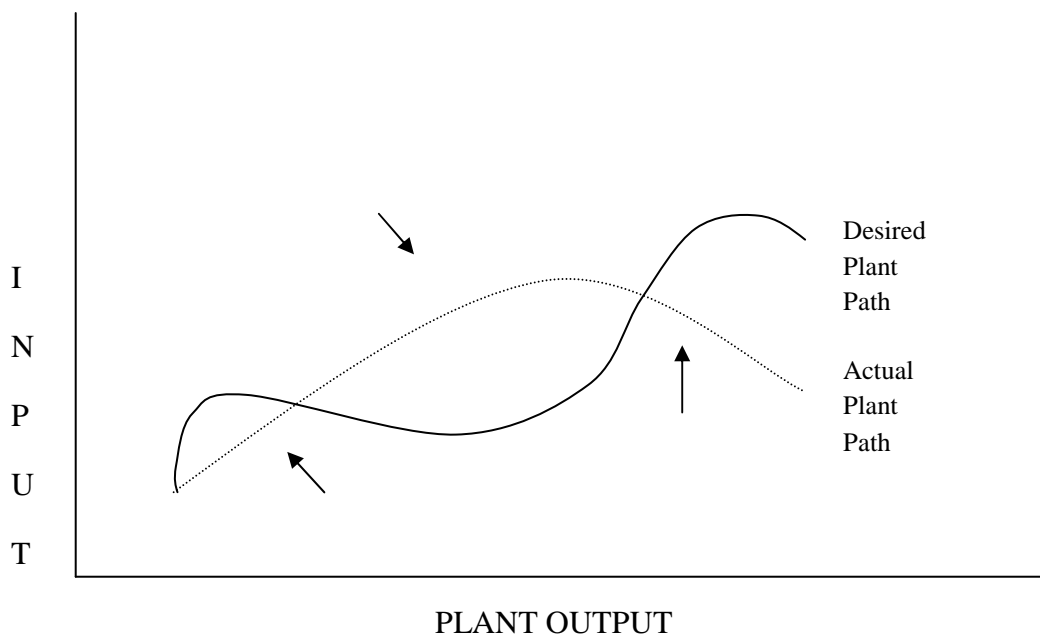
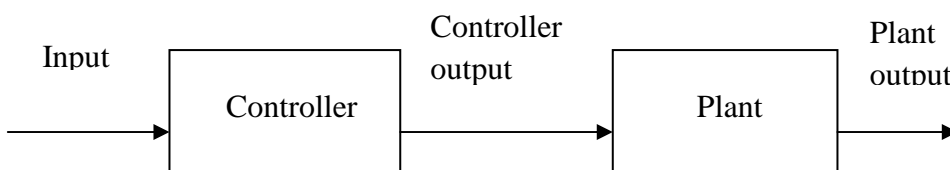
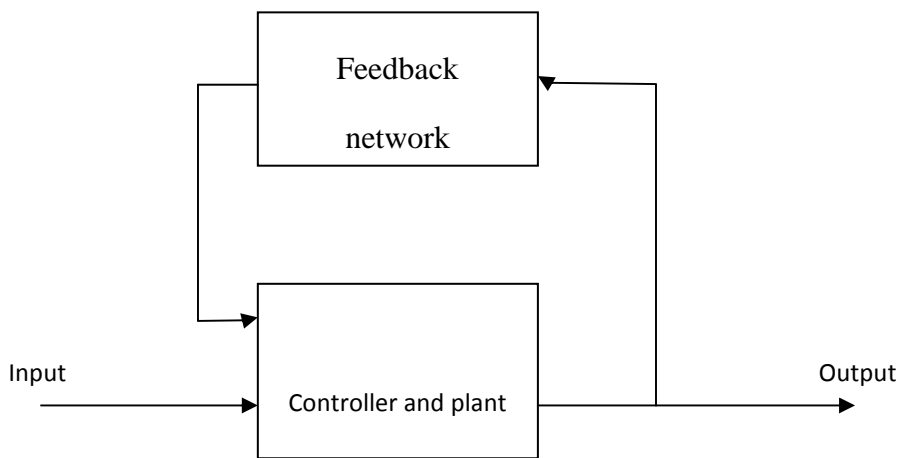


Figure 5 Desired and actual plant path.

Σχήμα 5: Επιθυμητή και πραγματική οδός συγκροτήματος (plant)



Σχήμα 6: Ελεγκτής ανοικτού κυκλώματος

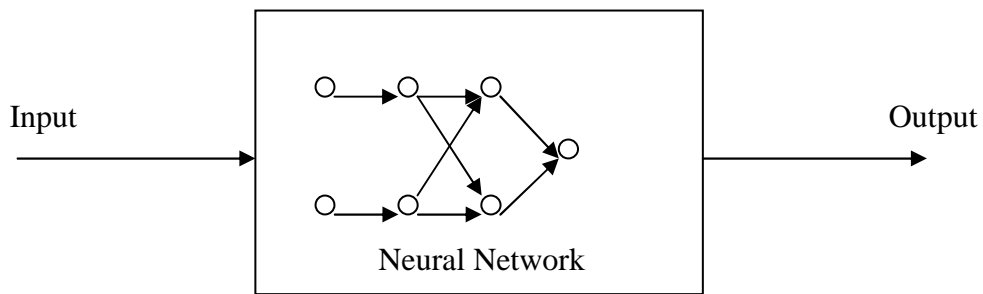


Σχήμα 7: Ελεγκτής κλειστού κυκλώματος

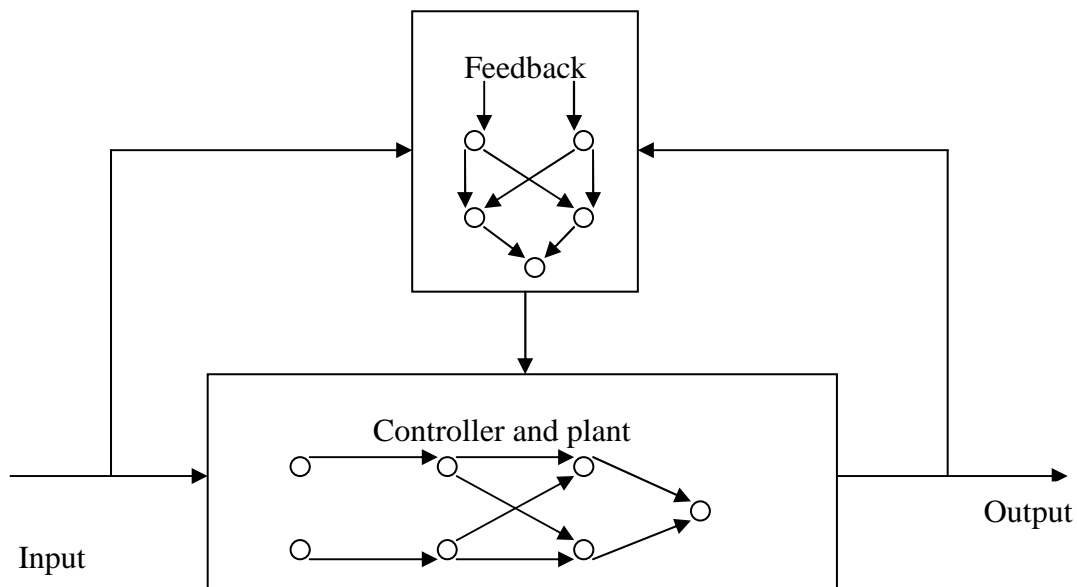
4.3.2 Νευρωνικός Έλεγχος

Τα νευρωνικά δίκτυα έχουν βρει πολλές εφαρμογές στον έλεγχο. Γενικά, ένα νευρωνικό δίκτυο μπορεί να εκπαιδευτεί για να ανταποκρίνεται σε διακυμάνσεις της εισόδου, έτσι ώστε η είσοδος να διατηρείται όσο το δυνατόν πιο κοντά στην επιθυμητή. Όπως και στον παραδοσιακό έλεγχο, μπορούμε να θεωρήσουμε ένα ανοικτού κυκλώματος και ένα κλειστού κυκλώματος σύστημα ελέγχου. Στην πρώτη περίπτωση, θεωρούμε ένα νευρωνικό δίκτυο, εκπαιδευμένο στο να ανταποκρίνεται στις τιμές εισόδου με μία επιθυμητή έξοδο (Σχήμα 8). Αν μία τιμή εισόδου δεν περιλαμβάνεται στο σετ των εκπαιδευτικών δεδομένων, τότε δεν είναι πάντα ξεκάθαρο τι θα είναι η έξοδος του νευρωνικού δικτύου και αν η έξοδος θα προκαλέσει αστάθεια στο plant. Όπως και στον παραδοσιακό έλεγχο, έτσι κι εδώ μπορεί να συμπεριλαμβάνεται ένα νευρωνικό δίκτυο ανατροφοδότησης (Σχήμα 9). Σε

αυτό το δίκτυο ελέγχου, το εμπρόσθιο δίκτυο εκπαιδεύεται στις αρχικές σχέσεις εισόδου-εξόδου. Αυτό εκπαιδεύεται για να ανταποκρίνεται στις εισόδους, γνωστές ως a priori (προγενέστερες), με εξόδους γνωστές ως a priori (προγενέστερες). Το δίκτυο ανατροφοδότησης συνεχώς προσαρμόζεται και μαθαίνει, ταξινομεί και δημιουργεί νέες σχέσεις εισόδου-εξόδου και ανανεώνει το εμπρόσθιο δίκτυο. Συνεπώς, αυτό το δίκτυο ανατροφοδότησης, μέσα σε βέβαια όρια γνώσης, έχει τη νοημοσύνη να μαθαίνει και να προσαρμόζεται συνεχώς.



Σχήμα 8: Ελεγκτής ANN ανοικτού κυκλώματος



Σχήμα 9: Ελεγκτής ANN κλειστού κυκλώματος

4.3.3 Συγκεκριμένος Έλεγχος

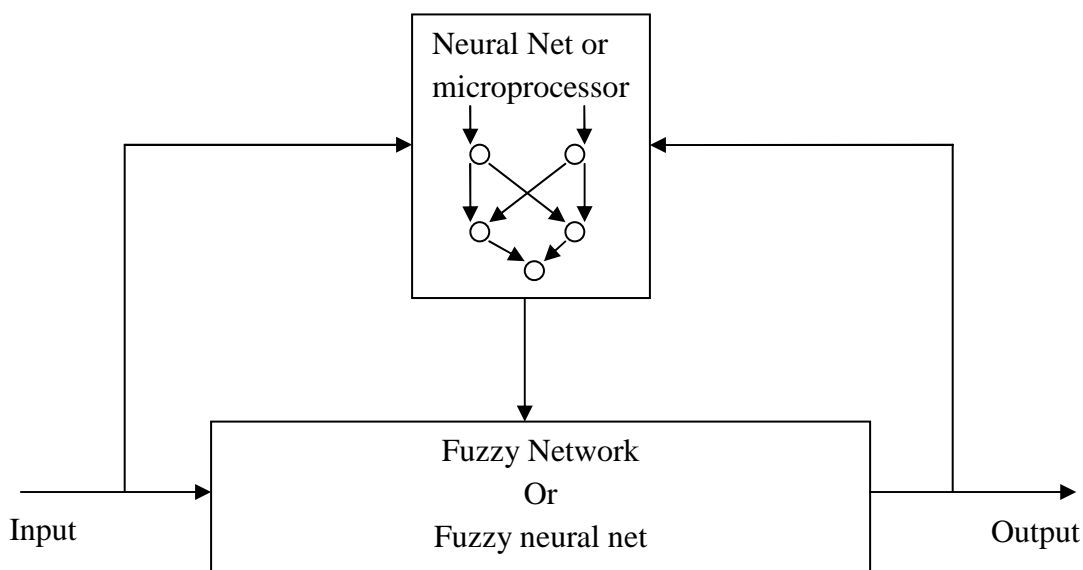
Τα συγκεκριμένα συστήματα χρησιμοποιούνται επίσης στον έλεγχο. Θεωρείστε ένα κύκλωμα κλειστού κυκλώματος, όπου το δίκτυο αποτελείται από εισόδους και εξόδους, όπως φαίνεται στο Σχήμα 10. Οι σχέσεις εισόδου-εξόδου εδώ είναι γνωστές με έναν “συγκεκριμένο τρόπο”. Οι κανόνες του plant ελέγχου, οι μεταβλητές και οι συμμετοχικές συναρτήσεις είναι γνωστές. Εδώ, δεν λαμβάνει χώρα εκμάθηση, όπως στα νευρωνικά δίκτυα. Με βάση αυτό, οι αισθητήρες του plant παρέχουν εισόδους στο συγκεκριμένο σύστημα και με βάση τους κανόνες, το σύστημα παράγει μία έξοδο που ελέγχει το plant. Σε αυτή την περίπτωση, η συμπεριφορά του plant πρέπει να είναι γνωστή πολύ καλά, ούτως ώστε να χρησιμοποιείται ένα ολοκληρωμένο σύνολο κανόνων. Αν το σετ δεν είναι ολοκληρωμένο, τότε το plant μπορεί να γίνει ασταθής.



Σχήμα 10: Συγκεκριμένος ελεγκτής ανοικτού κυκλώματος

4.3.4 Συγκεκριμένος-Νευρωνικός Έλεγχος

Τα συγκεκριμένα-νευρωνικά δίκτυα συνδυάζουν τις καλύτερες ιδιότητες των δύο κόσμων. Θεωρούν συγκεκριμένες εισόδους και, την ίδια στιγμή, έχουν τη δυνατότητα εκμάθησης, Σχήμα 11. Εδώ, το προς τα εμπρός δίκτυο είναι συγκεκριμένο και η ανατροφοδότηση είναι ένα νευρωνικό δίκτυο. Το νευρωνικό δίκτυο λαμβάνει τις εισόδους και τις πραγματικές εξόδους, δημιουργεί νέες κωδικοποιήσεις και σχέσεις εισόδου-εξόδου και παράγει νέους κανόνες. Επίσης, ενημερώνει το εμπρόσθιο δίκτυο με τους νέους κανόνες. Μία παραλλαγή του δικτύου είναι να έχει ένα δίκτυο βασισμένο σε μικροεπεξεργαστή στη θέση του νευρωνικού δικτύου, όπως φαίνεται στο Σχήμα 11. Ο μικροεπεξεργαστής, στη συνέχεια, συμπεριφέρεται αλγοριθμικά, σαν ένα νευρωνικό δίκτυο.



Σχήμα 11: FANN Ελεγκτής κλειστού κυκλώματος

5. ΣΥΓΚΕΧΥΜΕΝΟΣ ΕΛΕΓΚΤΗΣ

Στη δεύτερη φάση της μελέτης το $G(s)$ αντικαθίσταται από το συγκεκριμένο-νευρωνικό μοντέλο, που αναπτύχθηκε στην Παράγραφο 3. Στη συνέχεια, απαιτείται ένας συγκεκριμένος ελεγκτής για να ελέγξει την έξοδο του συστήματος, ώστε το σήμα εξόδου να είναι όσο το δυνατόν πιο κοντά στο σήμα εισόδου.

5.1 Δυναμική του συστήματος

Για να αποκτήσουμε μία ιδέα του αριθμού των εισόδων, εξόδων, συμμετοχικών συναρτήσεων και κανόνων που απαιτούνται για να κουμαντάρουν τον ελεγκτή, είναι απαραίτητο να ερευνήσουμε τη δυναμική του plant.

Για να το πράξουμε, μεταβάλλαμε το πλάτος του σήματος εισόδου και παρακολουθήσαμε την έξοδο. Παρατηρήθηκε ότι, για ένα σήμα εισόδου με μέγεθος 1, το μέγεθος του σήματος εξόδου φτάνει μία μέγιστη τιμή στο 0.5. Αυτό παρουσιάζεται στο Γράφημα 29.

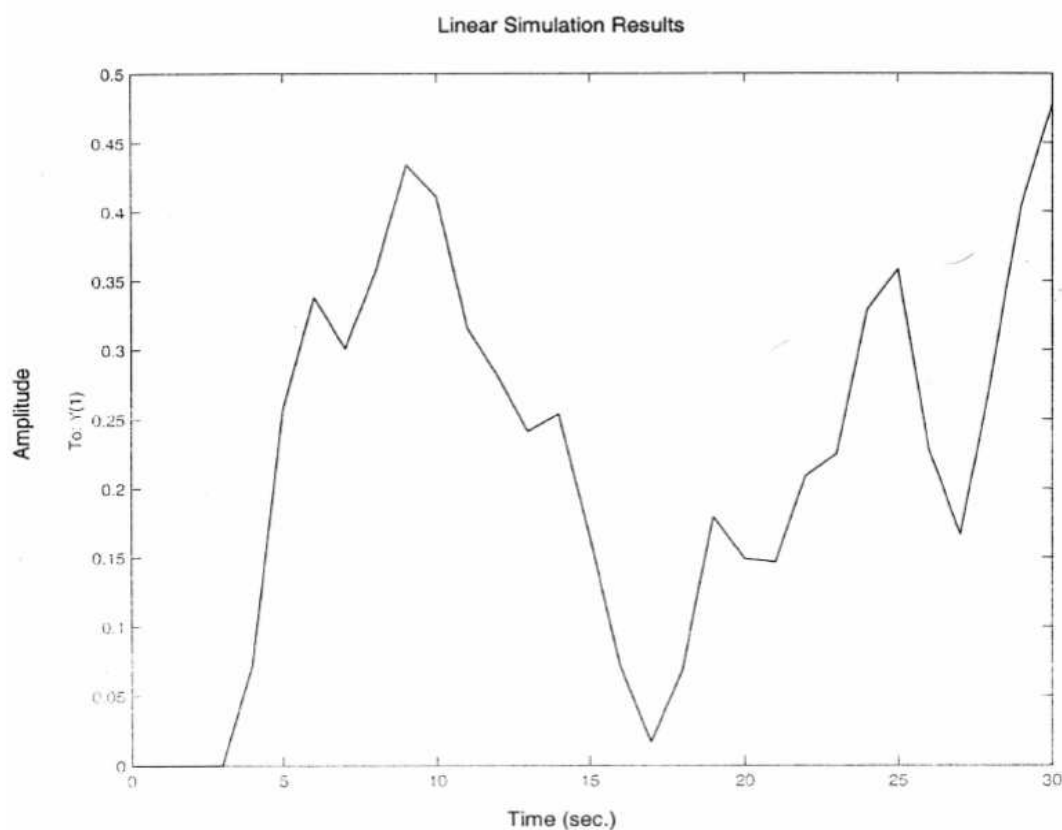
Αν διπλασιάσουμε το μέγεθος του σήματος εισόδου, η έξοδος διπλασιάζεται επίσης, καθώς φτάνει το ένα. Αν η είσοδος τριπλασιαστεί, το ίδιο γίνεται και για την έξοδο. Το ίδιο συμβαίνει αν το μέγεθος της εισόδου τετραπλασιαστεί, εξαπλασιαστεί ή δεκαπλασιαστεί, όπως φαίνεται και στο Γράφημα 30.

Αυτή η παρατήρηση μας επιτρέπει να καθιερώσουμε έναν αριθμό κανόνων, αν γνωρίζουμε την πραγματική έξοδο του συστήματος, την είσοδο και την επιθυμητή έξοδο. Ο στόχος είναι να παράγουμε ένα σήμα στην έξοδο του συστήματος, όσο το δυνατόν πιο κοντά στο σήμα εισόδου. Οι ακόλουθες παράγραφοι παρουσιάζουν μερικούς τρόπους για να γίνει αυτό.

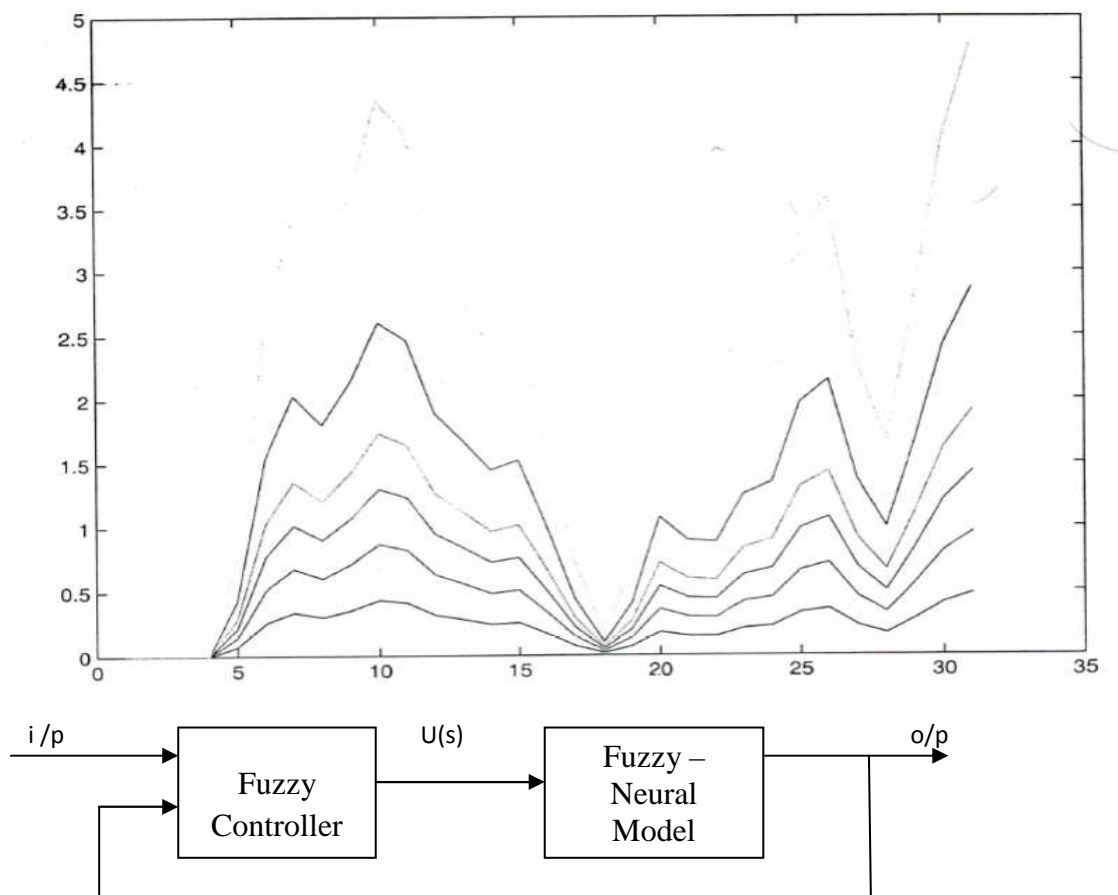
5.2 Δημιουργία ενός κατάλληλου σήματος εισόδου

Η πρώτη προσέγγιση στοχεύει στο να δημιουργήσουμε ένα σήμα εισόδου για το συγκεκριμένο-νευρωνικό μοντέλο, ικανό να δημιουργήσει ένα σήμα εξόδου παρόμοιο με αυτό της PRBS εισόδου του ελεγκτή.

29



30



Σχήμα 12: Συγκεκριμένος ελεγκτής που ελέγχει την είσοδο στο μοντέλο F-N

Το Γράφημα 31 παρουσιάζει τις εισόδους και την έξοδο του ελεγκτή, παρόμοια με αυτές που παρουσιάζονται στο Σχήμα 12. Ο συγκεκριμένος ελεγκτής έχει δύο εισόδους. Η πρώτη είναι η πραγματική είσοδος στο σύστημα, δηλαδή για αυτό το παράδειγμα μία ψευδό τυχαία δυαδική αλληλουχία. Η δεύτερη είσοδος είναι το σήμα εξόδου του συγκεκριμένου-νευρωνικού μοντέλου, που τροφοδοτείται πίσω στον ελεγκτή.

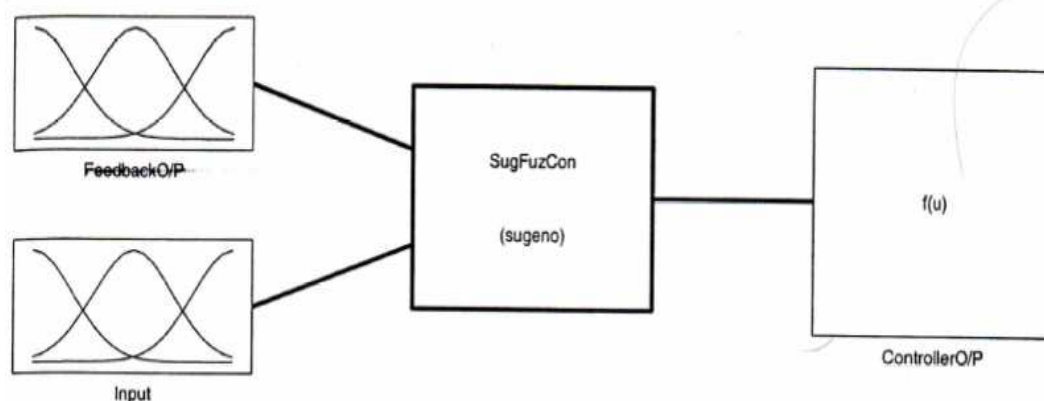
Η βασική ιδέα είναι, παρατηρώντας την είσοδο και την έξοδο του συστήματος, να αποδώσουμε κατάλληλους κανόνες και ταιριαστές συμμετοχικές συναρτήσεις για να παράγουν ένα ελεγκτικό σήμα εξόδου $U(s)$, ικανό να δημιουργεί την απαιτούμενη απόκριση στην έξοδο του συγκεκριμένου-νευρωνικού μοντέλου.

Ο ελεγκτής λειτουργεί σε δύο στάδια:

Πρώτον, στρέφεται στην έξοδο του συστήματος και την ταξινομεί σαν N (Αρνητική), VS (Πολύ Μικρή), S (Μικρή), MA (Μεσαίο Μέσης Τιμής), A (Μέσης Τιμής), B (Μεγάλο), VB (Πολύ Μεγάλο), αναλόγως σε ποια συμμετοχική συνάρτηση συντάσσεται.

Στη συνέχεια, λαμβάνει υπόψη την πραγματική είσοδο του συστήματος, μηδέν ή ένα, και αποφασίζει σχετικά με την έξοδό του, σύμφωνα με ένα σύνολο προκαθορισμένων κανόνων.

31



Όπως παρατηρήθηκε, κατά τη διάρκεια της διερεύνησης της δυναμικής του συγκεκριμένου-νευρωνικού μοντέλου, το σήμα εξόδου αυξάνει ή μειώνει το ποσοστό του μεγέθους του, με τον ίδιο ακριβώς τρόπο, όπως αυτό που εφαρμόζεται στο σήμα εισόδου. Συνεπώς, αν η απαιτούμενη έξοδος είναι ΕΝΑ και η πραγματική έξοδος είναι 0.5, και η πραγματική είσοδος είναι ΕΝΑ, αυτό που πρέπει να γίνει για τον ελεγκτή είναι να παράγει ΔΥΟ στην έξοδό του. Παρομοίως, αν η πραγματική έξοδος είναι 0,33, ο ελεγκτής πρέπει να παράγει ΤΡΙΑ στην έξοδό του, δηλαδή το σύστημα είναι γραμμικό.

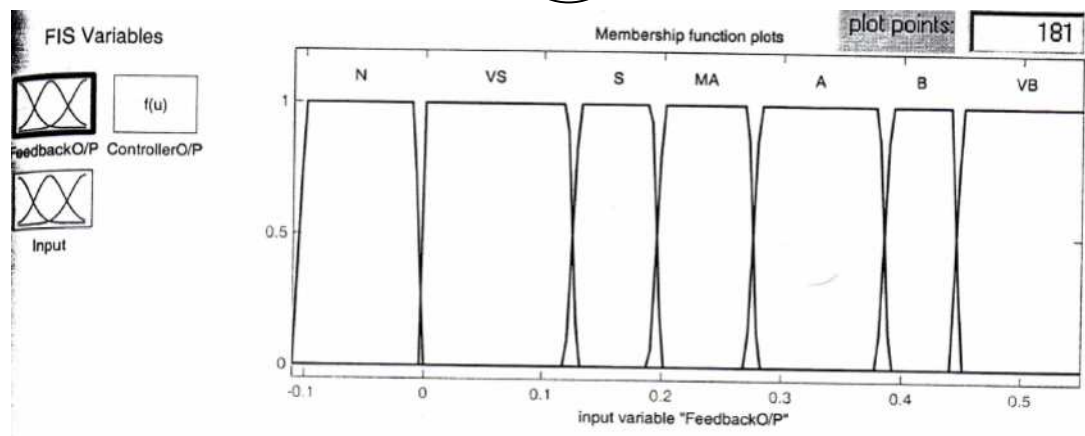
Από την άλλη, αν η απαιτούμενη έξοδος είναι ΜΗΔΕΝ και η πραγματική είσοδος στο σύστημα είναι ΜΗΔΕΝ, και η πραγματική έξοδος είναι ΔΙΑΦΟΡΗ ΤΟΥ ΜΗΔΕΝΟΣ, η έξοδος του ελεγκτή πρέπει να είναι ΜΗΔΕΝ ή ένα πολύ μικρό νούμερο (στην περίπτωση που μία μηδενική τιμή προκαλεί κάποια προβλήματα, όπως διαίρεση με το μηδέν, όταν χρησιμοποιείται σε εφαρμογές λογισμικού).

Τα ολοκληρώματα των συμμετοχικών συναρτήσεων της εισόδου ανατροφοδότησης παρουσιάζονται στο Γράφημα 32. Δουλεύοντας πάνω σε αυτή την εφαρμογή, ένας αριθμός διαφορετικών συμμετοχικών συναρτήσεων δοκιμάστηκαν, για να βρεθεί ότι

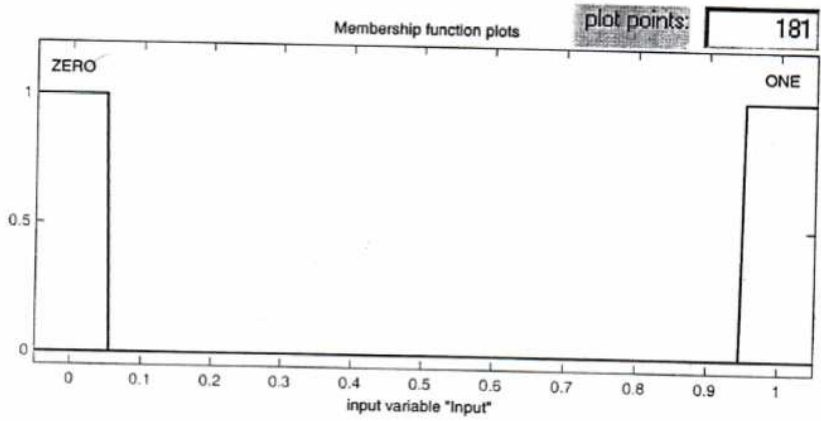
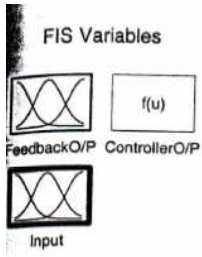
η καλύτερη απόκριση δινόταν από επίπεδες συμμετοχικές συναρτήσεις, όπως η συνάρτηση μορφής “καμπάνα” και η τραπεζοειδής συνάρτηση και όχι από σκληρές (δύσκαμπτες) συναρτήσεις, όπως οι τριγωνικές. Παρατηρήθηκε ότι, οι τριγωνικές συναρτήσεις, παρόλο που μειώνουν ελαφρώς το σκοπούμενο οριακό ποσό, όταν η είσοδος συμπέσει ακριβώς στην κορυφή της συνάρτησης, τείνουν να δημιουργήσουν περισσότερες κορυφές υπέρβασης, αν η είσοδος δεν ήταν κοντά στην κορυφή της συμμετοχικής συνάρτησης. Επίσης, η καμπανοειδής συνάρτηση παράγει καλύτερα αποτελέσματα από την τραπεζοειδή συνάρτηση, αφού είναι πιο ομαλή στα άκρα. Επιπροσθέτως, σε αυτό το παράδειγμα, η τραπεζοειδής συνάρτηση χρησιμοποιήθηκε όσο το δυνατόν ευκολότερα, για να ορίσει τις παραμετρικές ρυθμίσεις.

Τα Γραφήματα 33 και 34 παρουσιάζουν τις συμμετοχικές συναρτήσεις της πραγματικής εισόδου και της εξόδου του ελεγκτή, όπως παρουσιάστηκαν από τον συντάκτη της συγκεκριμένης εργαλειοθήκης του Matlab. Στη συνέχεια, σκιαγραφούνται οι δεκατέσσερις κανόνες που διέπουν τον ελεγκτή, ενώ το Γράφημα 35 παρουσιάζει τους κανόνες, όπως εκτίθενται από τον συντάκτη των κανόνων και τον θεατή των κανόνων της συγκεκριμένης εργαλειοθήκης

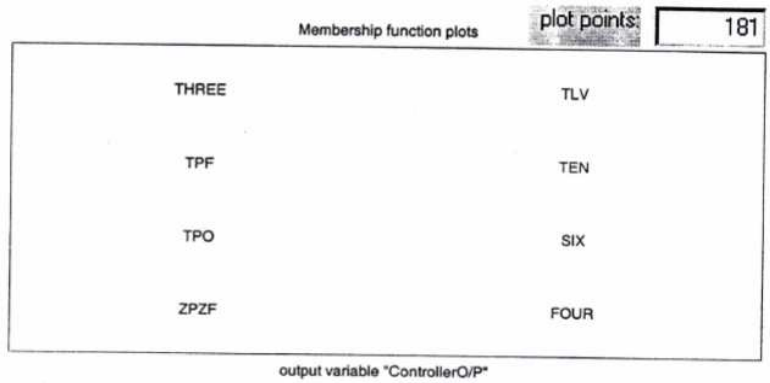
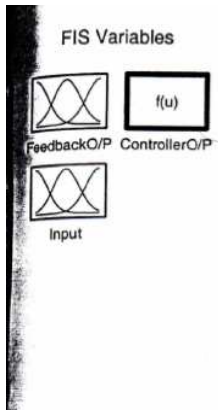
32



33



34



35

Rule Editor: SugFuzCon

File Edit View Options

```

1. If (FeedbackO/P is N) and (Input is ZERO) then (ControlO/P is ZPZF) (1)
2. If (FeedbackO/P is VS) and (Input is ZERO) then (ControlO/P is ZPZF) (1)
3. If (FeedbackO/P is S) and (Input is ZERO) then (ControlO/P is ZPZF) (1)
4. If (FeedbackO/P is MA) and (Input is ZERO) then (ControlO/P is ZPZF) (1)
5. If (FeedbackO/P is A) and (Input is ZERO) then (ControlO/P is ZPZF) (1)
6. If (FeedbackO/P is B) and (Input is ZERO) then (ControlO/P is ZPZF) (1)
7. If (FeedbackO/P is VB) and (Input is ZERO) then (ControlO/P is ZPZF) (1)
8. If (FeedbackO/P is H) and (Input is ONE) then (ControlO/P is TLV) (1)
9. If (FeedbackO/P is VS) and (Input is ONE) then (ControlO/P is TEN) (1)
10. If (FeedbackO/P is S) and (Input is ONE) then (ControlO/P is SQ) (1)
11. If (FeedbackO/P is MA) and (Input is ONE) then (ControlO/P is FOUR) (1)
12. If (FeedbackO/P is A) and (Input is ONE) then (ControlO/P is THREE) (1)
13. If (FeedbackO/P is B) and (Input is ONE) then (ControlO/P is TP) (1)
14. If (FeedbackO/P is VB) and (Input is ONE) then (ControlO/P is TPO) (1)

```

If FeedbackO/P is and Input is Then ControlO/P is

VS S MA A B VB none ONE none TP TEN THREE FOUR SQ TEN TLV none

not not not

Connection Weight

is and

Delete rule Add rule Change rule

FIS Name: SugFuzCon

Help Close

Start MATLAB Command Window FIS Editor: Untitled FIS Editor: SugFuzCon Membership Function Editor Rule Editor: SugFuzCon 12:08

Rule Viewer: SugFuzCon

File Edit View Options

FeedbackO/P = 0.22 Input = 1 ControlO/P = 4

1 2 3 4 5 6 7 8 9 10 11 12 13 14

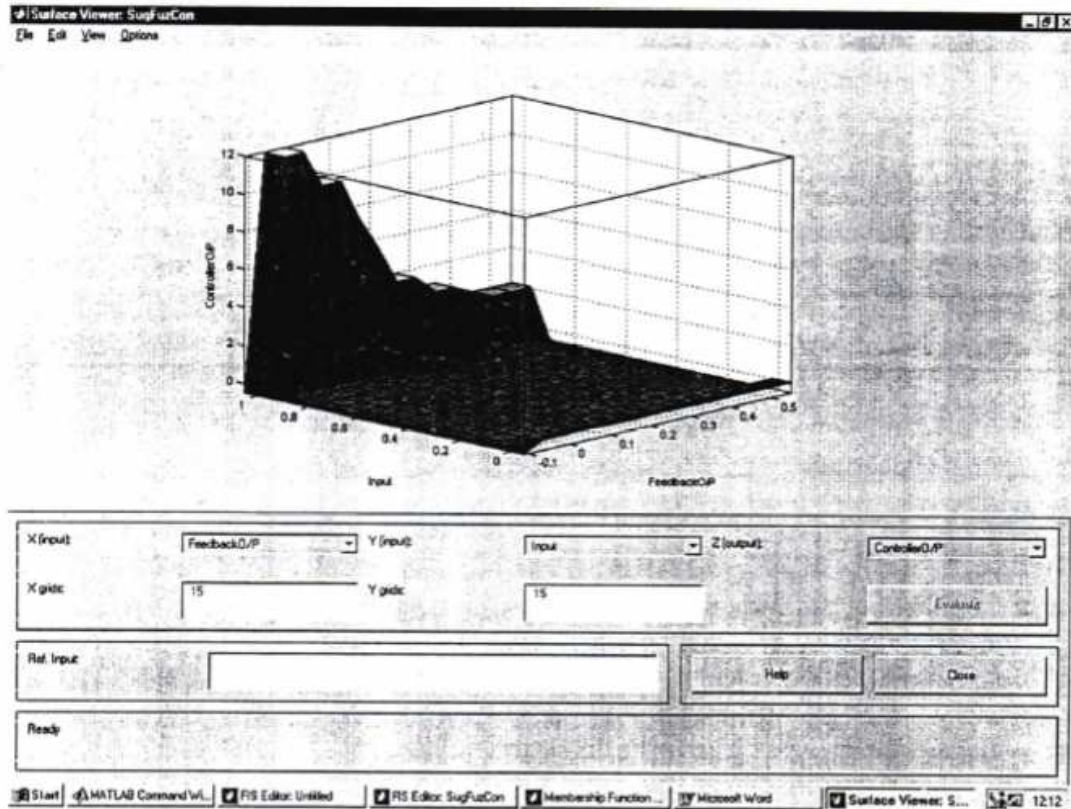
-0.11 0.55 -0.05 1.05 -2.112 7.302

Input: [0.22 1] Pot points: 101 Move: left right down up

Opened system SugFuzCon, 14 rules

Help Close

Start MATLAB Command Window FIS Editor: Untitled FIS Editor: SugFuzCon Membership Function Editor Microsoft Word - Docs... Rule Viewer: SugFuzCon 12:09



Κανόνας 1: Αν (Ανατροφοδότηση O/P είναι N) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ZPZF-0.05) (1)

Κανόνας 2: Αν (Ανατροφοδότηση O/P είναι VS) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ZPZF-0.05) (1)

Κανόνας 3: Αν (Ανατροφοδότηση O/P είναι S) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ZPZF-0.05) (1)

Κανόνας 4: Αν (Ανατροφοδότηση O/P είναι MA) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ZPZF-0.05) (1)

Κανόνας 5: Αν (Ανατροφοδότηση O/P είναι A) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ZPZF-0.05) (1)

Κανόνας 6: Αν (Ανατροφοδότηση O/P είναι B) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ZPZF-0.05) (1)

Κανόνας 7: Αν (Ανατροφοδότηση O/P είναι VB) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ZPZF-0.05) (1)

Κανόνας 8: Αν (Ανατροφοδότηση O/P είναι N) και (Είσοδος είναι ENA) τότε (Ελεγκτής O/P είναι TLV) (1)

Κανόνας 9: Αν (Ανατροφοδότηση O/P είναι VS) και (Είσοδος είναι ENA) τότε (Ελεγκτής O/P είναι ΔΕΚΑ) (1)

Κανόνας 10: Αν (Ανατροφοδότηση O/P είναι S) και (Είσοδος είναι ENA) τότε (Ελεγκτής O/P είναι ΕΞΙ) (1)

Κανόνας 11: Αν (Ανατροφοδότηση O/P είναι MA) και (Είσοδος είναι ENA) τότε (Ελεγκτής O/P είναι ΤΕΣΣΕΡΑ) (1)

Κανόνας 12: Αν (Ανατροφοδότηση O/P είναι A) και (Είσοδος είναι ENA) τότε (Ελεγκτής O/P είναι ΤΡΙΑ) (1)

Κανόνας 13: Αν (Ανατροφοδότηση O/P είναι B) και (Είσοδος είναι ENA) τότε (Ελεγκτής O/P είναι ΤΡΕΙΣ) (1)

Κανόνας 14: Αν (Ανατροφοδότηση O/P είναι VB) και (Είσοδος είναι ENA) τότε (Ελεγκτής O/P είναι ΤΡΟ) (1)

Οι πρώτοι επτά κανόνες ασχολούνται με την περίπτωση, όταν απαιτείται μία μηδενική (ZERO) είσοδος, μηδενική (ZERO) έξοδος, ενώ οι επόμενοι επτά κανόνες χρησιμοποιούνται στην περίπτωση μοναδιαίας (ONE) εισόδου, μοναδιαίας (ONE) εξόδου. Στο Γράφημα 36 παρουσιάζονται οι περιοχές δράσης του ελεγκτή.

Στη συνέχεια, η έξοδος του συγκεκριμένου ελεγκτή χρησιμοποιείται σαν μία είσοδος στο plant, για να δοκιμάσει τον ελεγκτή. Η έξοδος του plant σχεδιάζεται μαζί με την πραγματική έξοδο του συστήματος στο Γράφημα 37. Αν συγκριθεί η έξοδος του συστήματος συμπεριλαμβανομένου του ελεγκτή, με αυτή του συστήματος χωρίς τον ελεγκτή (Γράφημα 17), μπορεί να δει κάποιος ότι υπάρχει μία επαρκής βελτίωση της απόκρισης της εξόδου. Το σήμα της εξόδου στοχεύει να προσαρμόσει κοντά στο 'one-1', ενώ στην προηγούμενη περίπτωση (χωρίς τον ελεγκτή) το μέγεθος της εξόδου μπορούσε να υπερβεί μόλις το '0.5', όταν απαιτείται μία έξοδος του ENA. Επίσης, στην περίπτωση που απαιτείται μία μηδενική έξοδος (ZERO), το σήμα της εξόδου φτάνει αυτή την τιμή πολύ ταχύτερα όταν χρησιμοποιείται ο συγκεκριμένος ελεγκτής, απ' ό,τι στην περίπτωση χωρίς ελεγκτή.

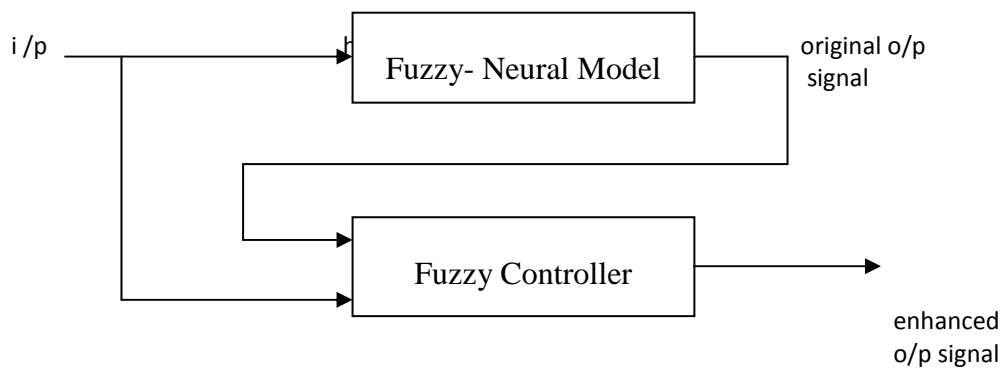
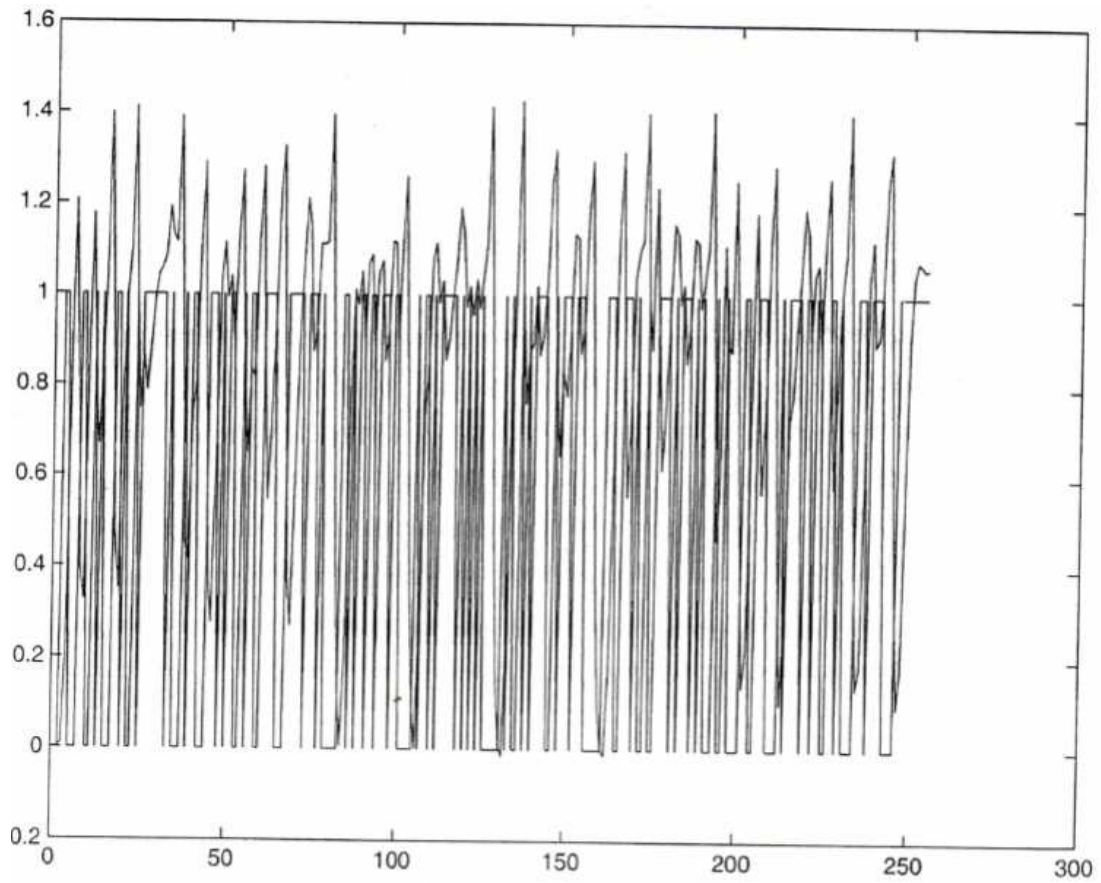
Μία πλήρης ανάλυση αυτού του Sugeno συγκεκριμένου ελεγκτή παρουσιάζεται στο Παράρτημα 17.

Παρόλο που η έξοδος του συστήματος βελτιώνεται, χρησιμοποιώντας αυτό το συγκεκριμένο ελεγκτή, δεν μπορεί κάποιος να αγνοήσει τον αριθμό και το ποσό υπέρβασης στο σήμα εξόδου, μαζί με την αποτυχία επίτευξης της επιθυμητής εξόδου

(είτε 0 ή 1), αν υπάρχει μία μετάβαση από το 0 στο 1 ή αντίστροφα, σε μία μικρή χρονική περίοδο. Επίσης, είχε προκύψει ένα πρόβλημα έναρξης, σχετικά με τον τρόπο δοκιμής της εξόδου του ελεγκτή στο συγκεχυμένο-νευρωνικό μοντέλο.

Το συγκεχυμένο-νευρωνικό μοντέλο, που αναπτύχθηκε νωρίτερα, έχει τέσσερις εισόδους και μία έξοδο. Μία από τις εισόδους είναι η πραγματική είσοδος (PBR) στο σύστημα, ενώ οι άλλες τρεις χρησιμοποιούνται σαν αναδρομικές (recursive) εισοδοί, δηλαδή οι προηγούμενες τρεις τελευταίες εξοδοί χρησιμοποιούνται σαν εισοδοί. Το πρόβλημα σε αυτήν την περίπτωση είναι ότι, ο ελεγκτής έχει μία μόνο έξοδο, που μπορεί να χρησιμοποιηθεί σαν μία είσοδος στο συγκεχυμένο-νευρωνικό μοντέλο. Πραγματοποιήθηκε μία απόπειρα να εισάγουμε μηδενικές τιμές στις υπόλοιπες εισόδους, για να δημιουργήσουμε μία πρώτη έξοδο και μετά να χρησιμοποιηθεί αυτή, καταλήγοντας σαν αναδρομικές (recursive) εισόδους στο συγκεχυμένο-νευρωνικό μοντέλο, αλλά υπήρχαν σημαντικές διακυμάνσεις στο σήμα εξόδου. Συνεπώς, η απόδοση του ελεγκτή έπρεπε να δοκιμαστεί στην πραγματική απόκριση του $G(s)$.

Όπως αναφέρθηκε παραπάνω, υπάρχουν περαιτέρω περιθώρια βελτίωσης, που είναι απαραίτητο να γίνουν για να βελτιωθεί η απόκριση της εξόδου. Μία πρώτη σκέψη θα ήταν, να αυξηθεί ο αριθμός των συμμετοχικών συναρτήσεων, αντιστοιχώντας σε μία από τις εισόδους (Ανατροφοδότηση O/P) και την έξοδο (Ελεγκτής O/P) του ελεγκτή. Δυστυχώς, η εργαλειοθήκη της συγκεχυμένης λογικής περιορίζει τις συμμετοχικές συναρτήσεις ανά μεταβλητή σε ένα μέγιστο αριθμό των εννιά. Ο ελεγκτής, που σχεδιάστηκε νωρίτερα, ήδη χρησιμοποιεί 7 συμμετοχικές συναρτήσεις. Συνεπώς, χρησιμοποιώντας και τις 9, μπορούμε να καταλήξουμε σε μία μόλις μικρή βελτίωση του σήματος εξόδου.



5.3 Ενίσχυση του σήματος εξόδου

Μία δεύτερη σκέψη ήταν, αντί να προσπαθήσουμε να δημιουργήσουμε ένα κατάλληλο σήμα εισόδου για το συγκεκριμένο-νευρωνικό μοντέλο για να δημιουργήσουμε την επιθυμητή έξοδο, να χρησιμοποιήσουμε την πραγματική είσοδο (PRBS) σαν είσοδο στον ελεγκτή, να δημιουργήσουμε ένα σήμα εξόδου, και μετά να χρησιμοποιήσουμε ένα συγκεκριμένο ελεγκτή για να ενισχύσουμε αυτό το σήμα, με ένα τέτοιο τρόπο ώστε, να δημιουργήσουμε το επιθυμητό σήμα στην έξοδο του ελεγκτή. Ο παραπάνω σχηματισμός παρουσιάζεται στο Σχήμα 13.

Για να το πετύχουμε αυτό, έπρεπε να ταξινομήσουμε την έξοδο του F-N μοντέλου σε 7 συμμετοχικές συναρτήσεις. 'N' αν κυμαίνεται μεταξύ -0.1 και 0, 'VS' από 0 έως 0.12, 'S' από 0.13 έως 0.19, 'MA' από 0.2 έως 0.27, 'A' για τιμές μεταξύ 0.28 και 0.38, 'B' από 0.39 έως 0.44, και 'VB' από 0.45 έως 0.55. Αυτές οι συμμετοχικές συναρτήσεις παρουσιάζονται στο Γράφημα 39. Παρατηρούμε ότι, όλες οι παραπάνω τιμές είναι προσεγγιστικές και όλες οι συναρτήσεις αλληλεπιδρούν η μία με την άλλη. Το Γράφημα 40 δείχνει τις συμμετοχικές συναρτήσεις του σήματος της PRBS εισόδου. Παρόλο που η PRBS απόκριση έχει τιμές μόνο 0 ή 1, οι συμμετοχικές συναρτήσεις κυμαίνονται με 0.5 ή τον άλλο τρόπο στην περίπτωση που ο θόρυβος εισάγεται στο σήμα εισόδου.

Οι συμμετοχικές συναρτήσεις των ελεγκτικών εξόδων είναι γραμμικές, δηλαδή κάθε μία από αυτές είναι ένας παράγοντας που πολλαπλασιάζεται στην έξοδο του F-N μοντέλου για να αποδώσει την έξοδο του ελεγκτή.

Οι συμμετοχικές συναρτήσεις είναι 'ZPZF' αν ο πολλαπλασιαστικός παράγοντας είναι 0.04, 'TPO' αν ο πολλαπλασιαστικός παράγοντας είναι 2.1, 'TPF' για 2.5, 'TRIA' για 3, 'TESSERA' για 4, 'EΞI' για 6, 'ΔEKA' για 10, και 'MTLV' για -12, και παρουσιάζονται στο Γράφημα 41.

Ακολουθούν οι 14 κανόνες που κουμαντάρουν τον συγκεκριμένο ελεγκτή:

Κανόνας 1: Αν (Ανατροφοδότηση O/P είναι N) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ZPZF-0.05) (1)

Κανόνας 2: Αν (Ανατροφοδότηση O/P είναι VS) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ZPZF-0.05) (1)

Κανόνας 3: Αν (Ανατροφοδότηση O/P είναι S) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ZPZF-0.05) (1)

Κανόνας 4: Αν (Ανατροφοδότηση O/P είναι MA) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ZPZF-0,05) (1)

Κανόνας 5: Αν (Ανατροφοδότηση O/P είναι A) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ΖΡΖΦ-0,05) (1)

Κανόνας 6: Αν (Ανατροφοδότηση O/P είναι Β) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ΖΡΖΦ) (1)

Κανόνας 7: Αν (Ανατροφοδότηση O/P είναι VB) και (Είσοδος είναι ΜΗΔΕΝ) τότε (Ελεγκτής O/P είναι ΖΡΖΦ-0,05) (1)

Κανόνας 8: Αν (Ανατροφοδότηση O/P είναι N) και (Είσοδος είναι ΕΝΑ) τότε (Ελεγκτής O/P είναι ΜΤΛV) (1)

Κανόνας 9: Αν (Ανατροφοδότηση O/P είναι VS) και (Είσοδος είναι ΕΝΑ) τότε (Ελεγκτής O/P είναι ΔΕΚΑ) (1)

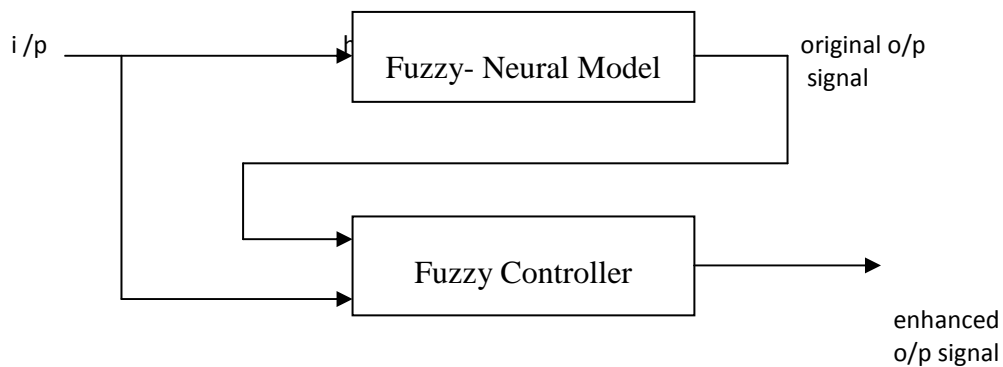
Κανόνας 10: Αν (Ανατροφοδότηση O/P είναι S) και (Είσοδος είναι ΕΝΑ) τότε (Ελεγκτής O/P είναι ΕΞΙ) (1)

Κανόνας 11: Αν (Ανατροφοδότηση O/P είναι MA) και (Είσοδος είναι ΕΝΑ) τότε (Ελεγκτής O/P είναι ΤΕΣΣΕΡΑ) (1)

Κανόνας 12: Αν (Ανατροφοδότηση O/P είναι A) και (Είσοδος είναι ΕΝΑ) τότε (Ελεγκτής O/P είναι ΤΡΙΑ) (1)

Κανόνας 13: Αν (Ανατροφοδότηση O/P είναι Β) και (Είσοδος είναι ΕΝΑ) τότε (Ελεγκτής O/P είναι ΤΡΦ) (1)

Κανόνας 14: Αν (Ανατροφοδότηση O/P είναι VB) και (Είσοδος είναι ΕΝΑ) τότε (Ελεγκτής O/P είναι ΤΡΟ) (1)



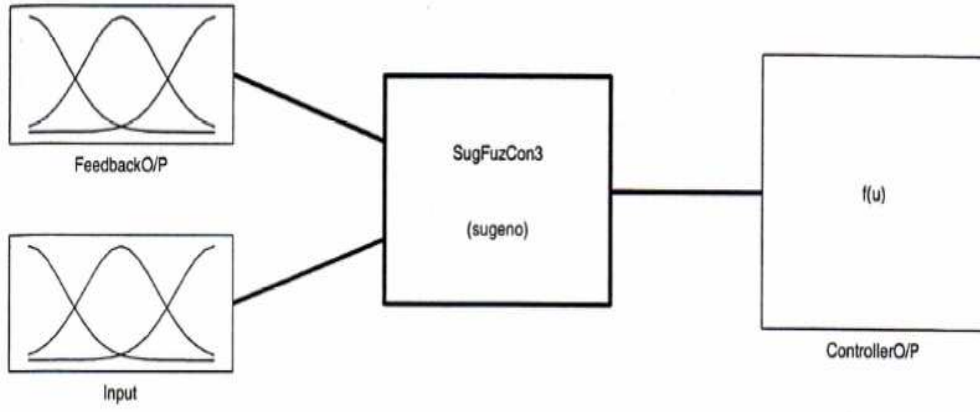
Σχήμα 13: Συγκεχυμένος ελεγκτής που ενισχύει την έξοδο του μοντέλου F-N

Το Γράφημα 38 δείχνει τις εισόδους και τις εξόδους του ελεγκτή με είσοδο “Ανατροφοδότηση O/P” να είναι η έξοδος του συγκεχυμένου-νευρωνικού μοντέλου, “Είσοδος” η πραγματική είσοδος στο σύστημα (PRBS απόκριση), και “Ελεγκτής O/P” η έξοδος του ελεγκτή.

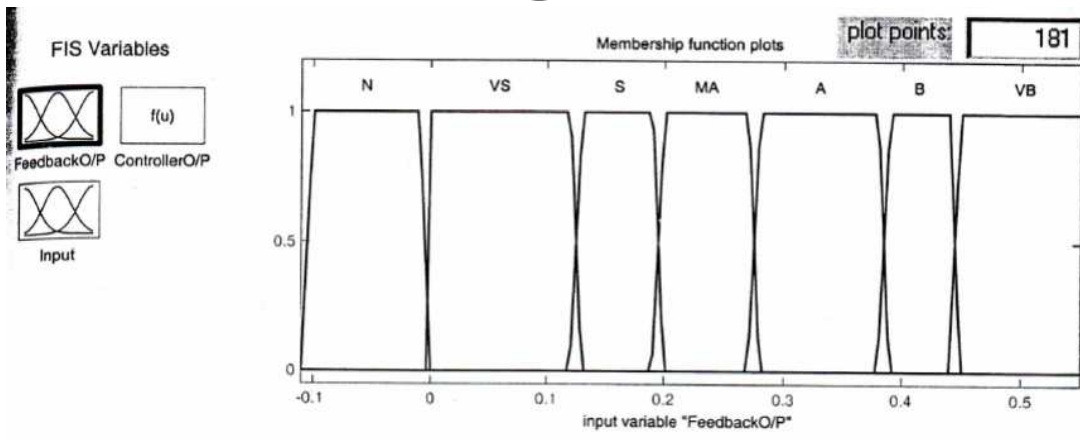
Ο ελεγκτής δουλεύει ως εξής:

Πρώτα, λαμβάνει την έξοδο του συγκεχυμένου-νευρωνικού μοντέλου μαζί με την πραγματική είσοδο (PBRS) του συστήματος και κοιτάει την τιμή της πραγματικής εισόδου. Αν η είσοδος στο σύστημα είναι “ENA”, πολλαπλασιάζει το σήμα εξόδου από το συγκεχυμένο-νευρωνικό μοντέλο με έναν κατάλληλο παράγοντα , έτσι ώστε η παραγόμενη τιμή είναι περίπου “ENA”.

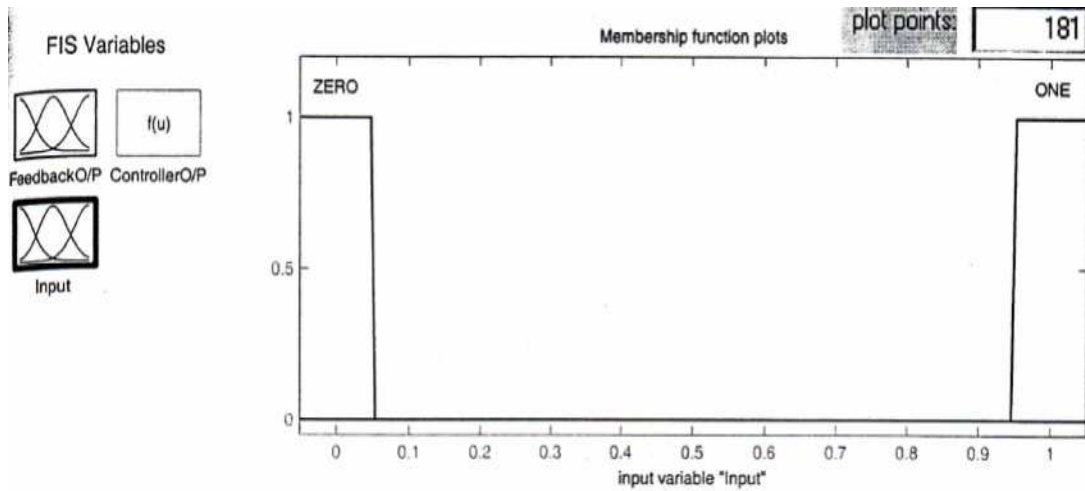
Αν η είσοδος στο σύστημα είναι “ΜΗΔΕΝ”, πολλαπλασιάζει το σήμα εξόδου από το συγκεχυμένο-νευρωνικό μοντέλο, με έναν μικρό δεκαδικό αριθμό, έτσι ώστε η παραγόμενη τιμή (Ελεγκτή O/P) είναι προσεγγιστικά “ΜΗΔΕΝ”.



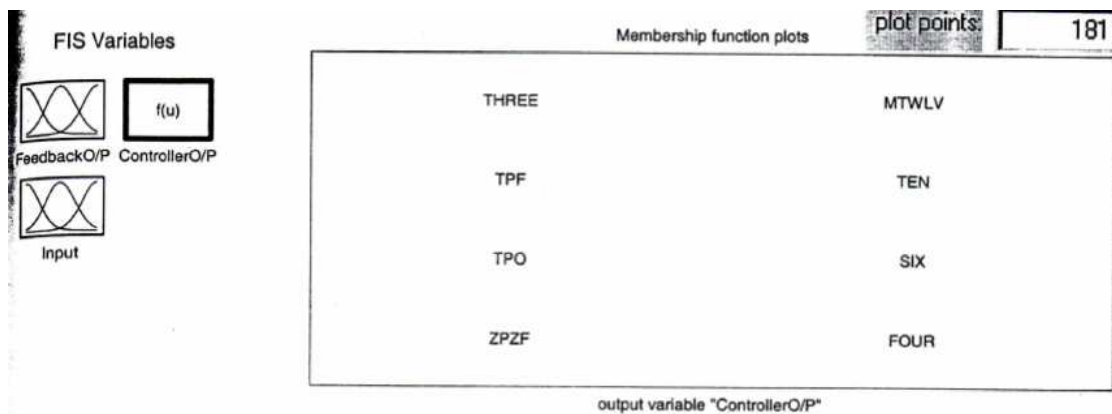
39



40



41



Ο κύριος στόχος των κανόνων είναι να ταυτοποιήσουν τους διαφορετικούς συνδυασμούς της εισόδου και να δράσουν ανάλογα. Στο Γράφημα 42, μπορεί κάποιος να δει τους 14 κανόνες του ελεγκτή, όπως παρουσιάζονται από τον Συντάκτη Κανόνων και τον Θεατή Κανόνων στην εργαλειοθήκη συγκεκριμένης λογικής, ενώ στο Γράφημα 43, κάποιος μπορεί να παρατηρήσει τις περιοχές δραστηριότητας του ελεγκτή. Πλήρεις λεπτομέρειες του συγκεκριμένου ελεγκτή δίνονται στο Παράρτημα 8.

Η έξοδος του συγκεκριμένου ελεγκτή σχεδιάζεται κατά μήκος της PRBS εισόδου του συστήματος στο Γράφημα 44. Όπως παρατηρήθηκε, το σήμα εξόδου είναι πολύ κοντά στο σήμα εισόδου, με λιγότερες και ασθενέστερες υπερβάσεις (σφάλματα), σε σύγκριση με την έξοδο της διάταξης στην Παράγραφο 5.2.

Φαίνεται ότι, είναι ευκολότερο για ένα συγκεκριμένο ελεγκτή να ενισχύσει ένα σήμα εξόδου, με σκοπό να παράγει μία απόκριση στην έξοδο, από το να δημιουργήσει ένα κατάλληλο σήμα εισόδου για το συγκεκριμένο-νευρωνικό μοντέλο, για να παράγει την επιθυμητή έξοδο.

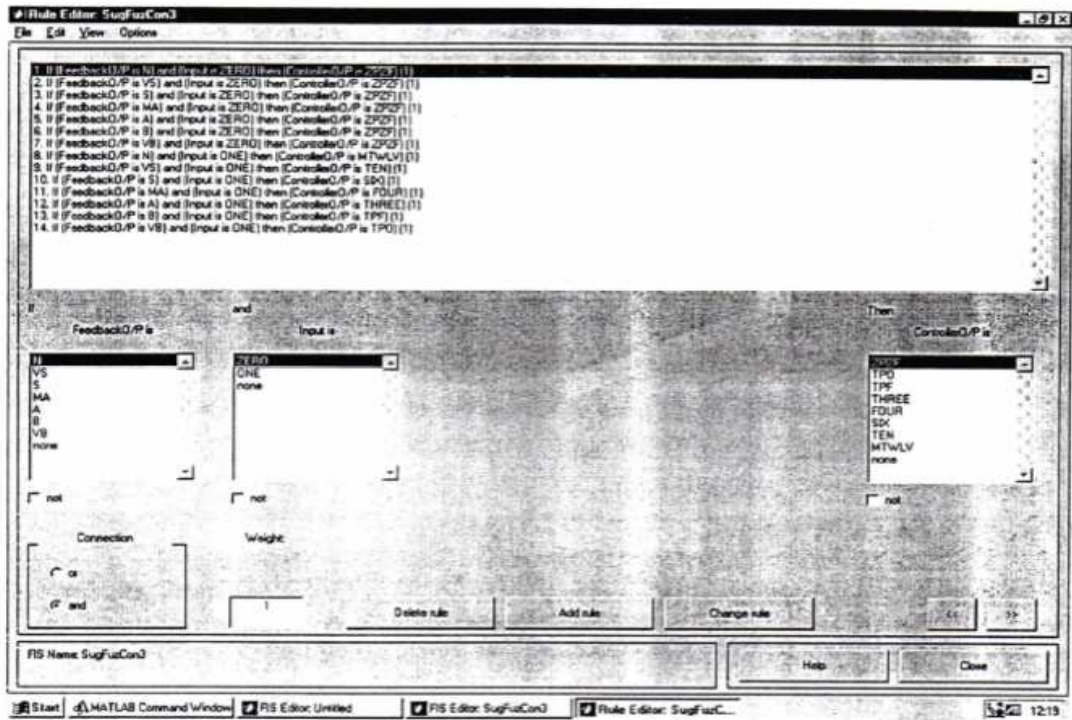
5.4 ΜΕΘΟΔΟΙ ΓΙΑ ΠΕΡΑΙΤΕΡΩ ΒΕΛΤΙΩΣΗ

Μολονότι ο κύριος στόχος, δηλαδή το να παράγουμε ένα σήμα εξόδου όσο το δυνατόν πιο κοντά στο σήμα εισόδου, επετεύχθη, υπάρχει πάντα περιθώριο για περαιτέρω βελτίωση. Οι ερευνητές έχουν αναπτύξει διάφορες συγκεκριμένες-νευρωνικές μεθόδους για έλεγχο, και υπάρχουν επίσης εκτεταμένες εργασίες που πραγματοποιήθηκαν, για να εφαρμοστούν γενετικοί αλγόριθμοι σε εφαρμογές συγκεκριμένου ελέγχου, γνωστές ως Προσαρμοστικός Συγκεκριμένος Έλεγχος.

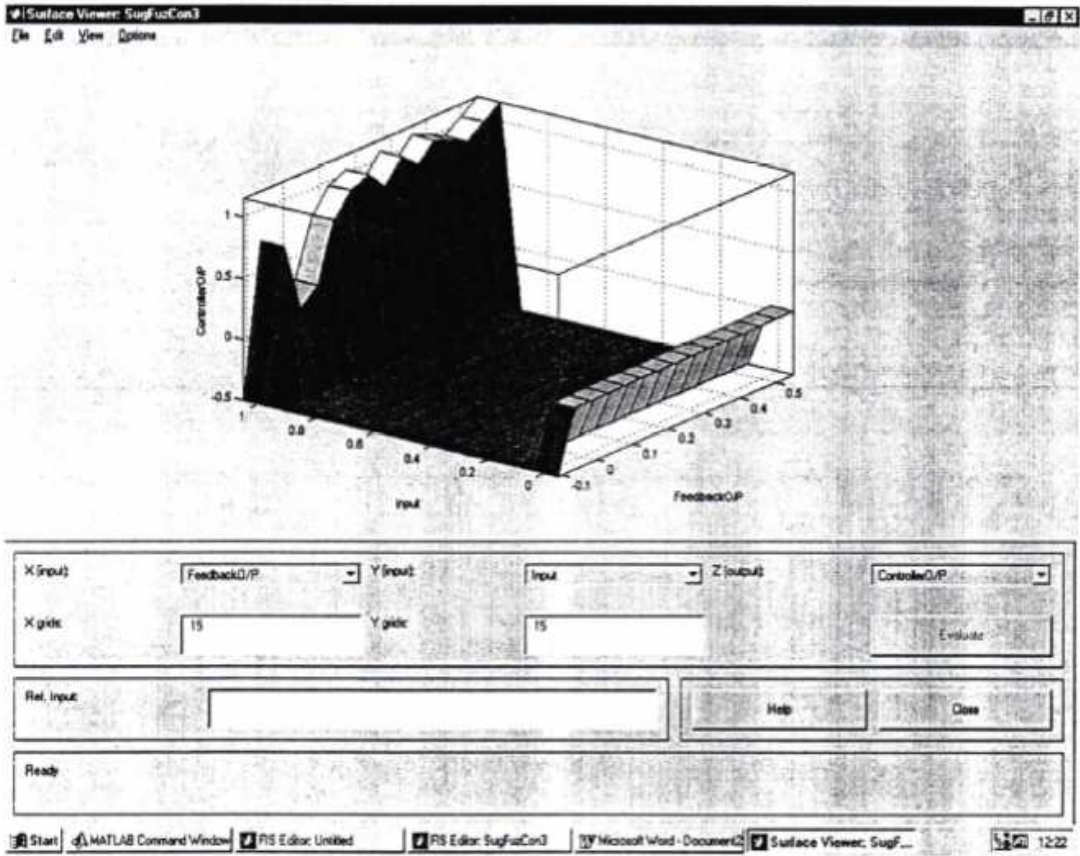
Οι επόμενες δύο παράγραφοι εισάγουν μία μέθοδο για κάθε περίπτωση:

- Αντίστροφη Εκμάθηση
- Αλγόριθμος Πίσω (Ανάστροφης) Διάδοσης σε Συστήματα Συγκεκριμένης Λογικής

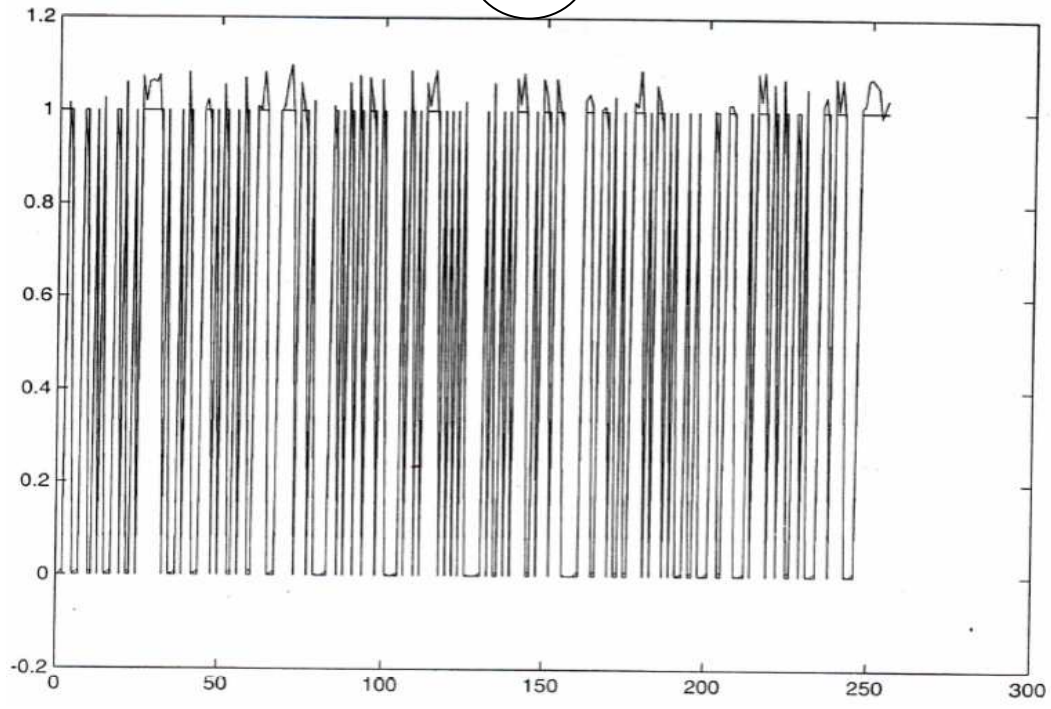
42



43



44



5.4.1 Αντίστροφη εκμάθηση

Η ανάπτυξη της αντίστροφης εκμάθησης, που είναι επίσης γνωστή και ως, γενική εκμάθηση, για σχεδίαση συγκεχυμένων-νευρωνικών ελεγκτών, περιλαμβάνει δύο φάσεις. Στη φάση εκμάθησης μία on-line ή μία off-line τεχνική χρησιμοποιείται για να διαμορφώσει την αντίστροφη δυναμική του plant. Το λαμβανόμενο νευρωνικό-συγκεχυμένο μοντέλο, το οποίο αντιπροσωπεύει την αντίστροφη δυναμική του plant, μετά χρησιμοποιείται για να δημιουργήσει ελεγκτικές δράσεις στη φάση εφαρμογής. Αυτές οι δύο φάσεις μπορούν να ενεργούν ταυτόχρονα, και ως εκ τούτου, αυτή η σχεδιαστική μέθοδος ταιριάζει απόλυτα με το κλασσικό προσαρμοστικό ελεγκτικό σχέδιο.

Θεωρώντας ότι η τάξη του plant (ο αριθμός των καταστατικών μεταβλητών) είναι γνωστή και όλες οι καταστατικές μεταβλητές είναι μετρήσιμες, έχουμε:

$$x(k+1)=f(x(k), u(k)) \quad (\text{Εξίσωση 1})$$

όπου, $x(k+1)$ είναι η κατάσταση σε χρόνο $k+1$, $x(k)$ είναι η κατάσταση σε χρόνο k , και $u(k)$ είναι το σήμα ελέγχου σε χρόνο k .

Παρατήρηση: Για παιδαγωγικούς σκοπούς θεωρούμε ότι η $u(k)$ είναι μία μονοδιάστατη συνάρτηση. Παρομοίως, η κατάσταση σε χρόνο $k+2$ εκφράζεται ως:

$$x(k+2)=f(x(k+1),u(k+1))=f(f(x(k),u(k)) \quad (\text{Εξίσωση 2})$$

Γενικά έχουμε:

$$x(k+n)=F(x(k),U) \quad (\text{Εξίσωση 3})$$

όπου, n είναι η τάξη του plant, F είναι μία πολλαπλή σύνθετη συνάρτηση της f , και U είναι οι δράσεις (ενέργειες) ελέγχου από k έως $k+n-1$, η οποία είναι ίση με $[u(k), u(k+1), \dots, u(k+n-1)]^T$. Η προηγούμενη εξίσωση υποδεικνύει το γεγονός ότι, δεδομένης της εισόδου ελέγχου u από χρόνο k σε $k+n-1$, η κατάσταση του plant θα μετακινηθεί από $x(k)$ σε $x(k+n)$ σε ακριβώς n χρονικά βήματα. Επιπλέον, θεωρούμε ότι, η αντίστροφη δυναμική του plant υπάρχει, και η U μπορεί να εκφραστεί σαν μία κατηγορηματική (αναλυτική) συνάρτηση του $x(k)$ και $x(k+n)$:

$$U=G(x(k),x(k+n)) \text{ (Εξίσωση 4)}$$

Αυτή η εξίσωση ουσιαστικά, λέει ότι, υπάρχει μία μοναδική ακολουθία εισόδου U , καθορισμένη από τη σχεδίαση του G , που μπορεί να οδηγήσει το plant από την κατάσταση $x(k)$ στη $x(k+n)$, σε n χρονικά βήματα. Το πρόβλημα τώρα είναι να βρεθεί ο τρόπος αντίστροφης σχεδίασης του G .

Θεωρούμε την υπόθεση όπου, η καταστατική εξίσωση στην Εξίσωση 1 είναι γραμμική. Στους όρους των γραμμικών συστημάτων η εξίσωση 1 μπορεί να γραφεί σαν:

$$x(k+1)=Ax(k)+Bu(k) \text{ (Εξίσωση 5)}$$

όπου, A και B είναι $n \times n$ και $n \times 1$ πίνακες αντίστοιχα. Επαναλαμβάνοντας την προηγούμενη εξίσωση, λαμβάνουμε την κατάσταση σε $k+n$:

$$x(k+n)=A^n x(k)+WU \text{ (Εξίσωση 6)}$$

όπου, $W=[A^{n-1}B \dots AB \quad B]$ είναι ο πίνακας ελεγχιμότητας. Αν ο W είναι μη-μοναδικός, τότε το σύστημα είναι ελέγξιμο και η U μπορεί να υπολογιστεί ως εξής:

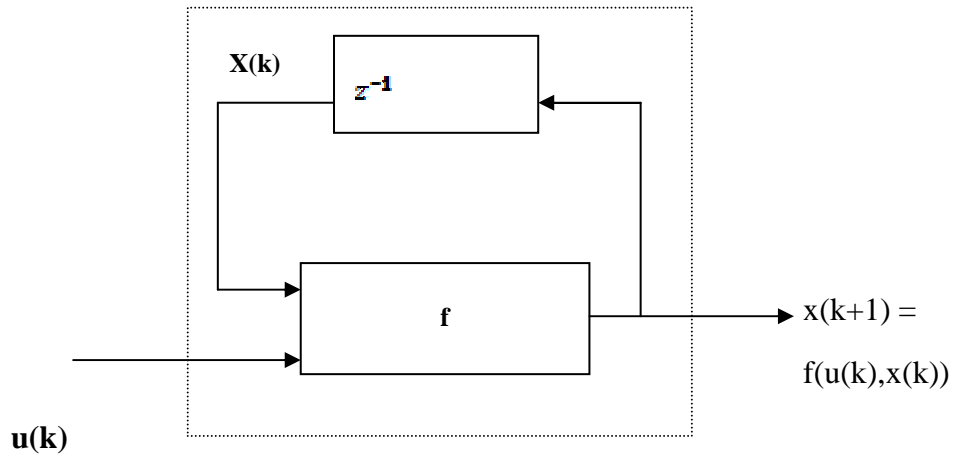
$$U=W^{-1}[x(k+n)-A^n x(k)] \text{ (Εξίσωση 7)}$$

Με άλλα λόγια, η ελεγχιμότητα σε ένα γραμμικό σύστημα είναι ισοδύναμη με την αντίστροφη κατάσταση που αναφέρθηκε στα προηγούμενα.

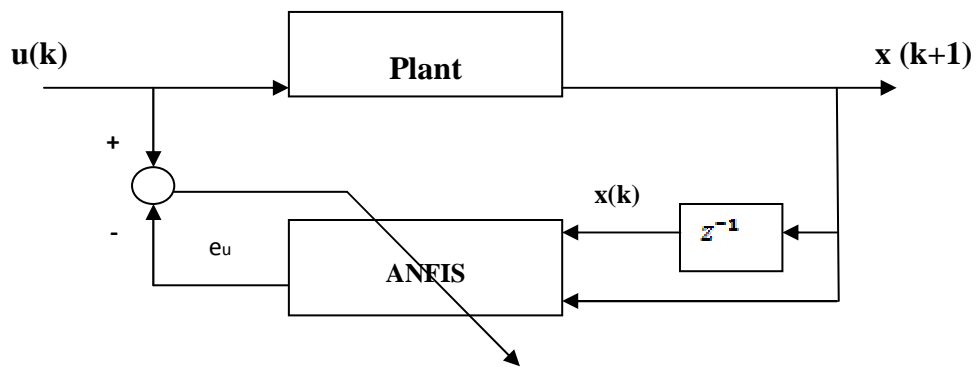
Παρόλο που ο αντίστροφος σχεδιασμός του G στην Εξίσωση 4 υπάρχει κατ' υπόθεση, δεν έχει πάντα μία διεξοδικά στενή (κλειστή) μορφή. Επομένως, αντί να αναζητούμε διεξοδικά, μεθόδους επίλυσης της Εξίσωσης 4, μπορούμε να χρησιμοποιήσουμε ένα προσαρμοστικό δίκτυο ή το ANFIS με $2n$ εισόδους και n εξόδους για να προσεγγίσουμε την αντίστροφη σχεδίαση G , σύμφωνα με τα γενικά ζεύγη εκπαιδευτικών δεδομένων:

$$[x(k)^T, x(k+n)^T U^T] \text{ (Εξίσωση 8)}$$

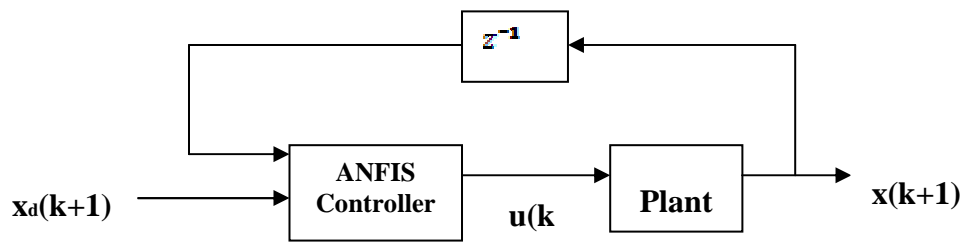
Plant



(a)



(b)



(c)

Σχήμα 14: Διάγραμμα κορμός (Συγκρότημα) για τη μέθοδο αντίστροφης εκμάθησης. (a) Κορμός plant, (b) Φάση εκπαίδευσης, (c) Φάση εφαρμογής

Το Σχήμα 14 απεικονίζει την κατάσταση, στην οποία το n ισούται με 1. Στο Σχήμα 14 (a) παρουσιάζεται ένα plant block, στο οποίο η έξοδος του plant είναι μία συνάρτηση μίας προηγούμενης κατάστασης $x(k)$ και είσοδο $u(k)$. Χρησιμοποιούμε z^{-1} block για να αναπαραστήσουμε το διαχειριστή καθυστέρησης ανά μονάδα χρόνου. Το Σχήμα 14 (b) είναι το διάγραμμα block κατά τη διάρκεια της φάσης εκπαίδευσης. Το Σχήμα 14 (c) είναι το διάγραμμα block κατά τη διάρκεια της φάσης εφαρμογής.

Θεωρούμε ότι, το προσαρμοστικό δίκτυο πραγματικά, αντιγράφει τη σχεδίαση εισόδου-εξόδου της αντίστροφης δυναμικής του G . Στη συνέχεια, δεδομένης της τρέχουσας κατάστασης $x(k)$ και της επιθυμητής μέλλουσας κατάστασης $x_d(k+n)$, το προσαρμοστικό δίκτυο θα δημιουργήσει μία υπολογισμένοι (εκτιμημένη) U' :

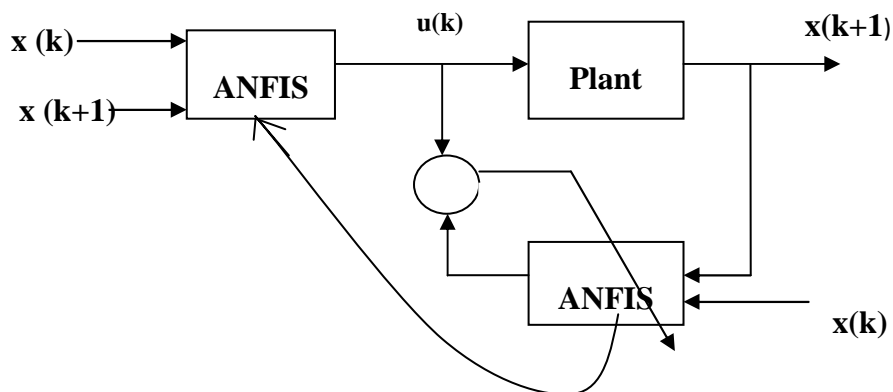
$$U' = G'(x(k), x_d(k+n)).$$

Μετά από n βήματα, αυτή η ελεγκτική ακολουθία μπορεί να οδηγήσει την κατάσταση $x(k)$ στην επιθυμητή κατάσταση $x_d(k+n)$, θεωρώντας ότι, η συνάρτηση G' του προσαρμοστικού δικτύου, είναι ακριβώς η ίδια, όπως της αντίστροφης σχεδίασης του G . Αυτή η φάση εφαρμογής παρουσιάζεται στο διάγραμμα κορμού (block) του Σχήματος 14 (b). Αν η μελλοντική επιθυμητή κατάσταση $x_d(k+n)$ δεν είναι εφικτή εκ των προτέρων, μπορούμε αντ' αυτής να χρησιμοποιήσουμε την τρέχουσα επιθυμητή κατάσταση $x_d(k)$, στο Σχήμα 14 (b). Αυτό σημαίνει ότι, η τρέχουσα επιθυμητή κατάσταση θα εμφανιστεί μετά από n χρονικά βήματα και ολόκληρο το σύστημα συμπεριφέρεται σαν ένα καθαρό n -βημάτων χρονικής καθυστέρησης σύστημα.

Όταν η G' δεν είναι κοντά στη G , η ελεγκτική ακολουθία U' δεν μπορεί να φέρει την κατάσταση στην $x_d(k+n)$ ακριβώς, στα επόμενα n χρονικά βήματα. Όσο περισσότερα ζεύγη δεδομένων χρησιμοποιούνται για να βελτιώσουν τις παραμέτρους στο προσαρμοστικό δίκτυο, τόσο πιο πολύ θα συγκλίνει η G' στην G και ο έλεγχος θα γίνεται όλο και περισσότερο πιο ακριβής, καθώς η εκπαιδευτική διαδικασία θα προχωράει.

Για off-line εφαρμογές, πρέπει να συλλέξουμε ένα σύνολο ζευγών εκπαιδευτικών δεδομένων και μετά να εκπαιδύσουμε το προσαρμοστικό δίκτυο στη μέθοδο δέσμης-παρτίδας (batch mode). Για on-line εφαρμογές, για να ασχοληθούμε με χρονικά μεταβαλλόμενα συστήματα, οι ελεγκτικές δράσεις στην Εξίσωση 8, παράγονται κάθε

n χρονικά βήματα, ενώ η on-line εκμάθηση λαμβάνει χώρα σε κάθε χρονικό βήμα. Εναλλακτικά, μπορούμε να δημιουργήσουμε την ελεγκτική ακολουθία σε κάθε χρονικό βήμα και να εφαρμόσουμε μόνο το πρώτο μέρος του plant.



Duplicate

Σχήμα 15: Διάγραμμα κορμός για on-line αντίστροφη εκμάθηση

Το Σχήμα 15 είναι ένα διάγραμμα κορμός (block) για on-line εκμάθηση, όταν το n ισοδυναμεί με 1. Η διακεκομμένη γραμμή (καμπυλωτή) στο διάγραμμα δείχνει ότι τα δύο τμήματα του ANFIS είναι ακριβή αντίγραφα το ένα του άλλου. (Για απλότητα έχουμε μετακινήσει τον διαχειριστή μοναδιαίας χρονικής καθυστέρησης z^{-1}).

Η λογική πίσω από την αντίστροφη εκμάθηση φαίνεται απλή. Εντούτοις, υποτίθεται η ύπαρξη αντίστροφης δυναμικής για ένα plant, το οποίο γενικώς δεν είναι έγκυρο και βάσιμο. Επιπλέον, η ελαχιστοποίηση του δικτυακού σφάλματος $[[U-U']]$ δεν εγγυάται ελαχιστοποίηση του συνολικού σφάλματος του συστήματος.

5.4.2 Εκπαίδευση των συστημάτων συγκεκριμένης λογικής χρησιμοποιώντας πίσω (ανάστροφη) διάδοση

Μετά από μία σχετικά ήσυχη περίοδο την δεκαετία του 1970, η έρευνα πάνω στα τεχνητά νευρωνικά δίκτυα έχει κερδίσει το ισχυρό δημόσιο ενδιαφέρον κατά την περίοδο των τελευταίων ετών. Ένας λόγος, για την αναζωπύρωση του ενδιαφέροντος, είναι η ανακάλυψη ενός ισχυρού εκπαιδευτικού αλγόριθμου για πολυεπίπεδα νευρωνικά δίκτυα, ο αποκαλούμενος αλγόριθμος πίσω (ανάστροφης) διάδοσης. Ουσιαστικά, η βασική ιδέα του αλγόριθμου πίσω (ανάστροφης) διάδοσης μπορεί να εφαρμοστεί σε οποιοδήποτε δίκτυο εμπρόσθιας τροφοδοσίας. Συνεπώς, αν μπορούμε

να παραστήσουμε τα συστήματα συγκεκριμένης λογικής σαν δίκτυα εμπρόσθιας τροφοδοσίας, μπορούμε να χρησιμοποιήσουμε την ιδέα της πίσω (ανάστροφης) διάδοσης για να τα εκπαιδεύσουμε.

Κέντρο μέσου αποσυγκεχυμενοποιητή

Ο αποσυγκεχυμενοποιητής πραγματοποιεί σχεδίαση από συγκεκριμένα σύνολα στο V στο καθαρό σημείο y , που ανήκει στο V .

$$y = \frac{\sum_{i=1}^M y - 1 (\mu_B 1(y - 1))}{\sum_{i=1}^M (\mu_B 1(y - 1))}$$

όπου, y^{-1} είναι το κέντρο του συγκεκριμένου συνόλου, το οποίο είναι το σημείο στο V , στο οποίο το $\mu_G^1(y)$ επιτυγχάνει τη μέγιστη κλίμακα του.

Παραγωγή συμπερασματικού κανόνα

$$\mu_{F1} |_{x_1 \dots x_{Fn}} | (x) = \mu_{F1} | (x_1) \dots \mu_{Fn} | (x_n)$$

και

$$\mu_{A \rightarrow B}(x, y) = \min \{ \mu_A(x) \mu_B(y) \}.$$

Συγκεχυμενοποιητής Singleton

Το A' είναι ένα συγκεκριμένο singleton με υποστήριξη x , το οποίο είναι, $\mu_{A'}(x') = 1$ για $x' = x$ και $\mu_{A'}(x) = 0$ για όλα τα υπόλοιπα x που ανήκουν στο U με $x' \neq x$.

Συνάρτηση Caussian

$$\mu_{Fi}^1(x_i) = \alpha_i^1 \exp[-(x_i - x_i^{-1}) / \sigma_i^1]$$

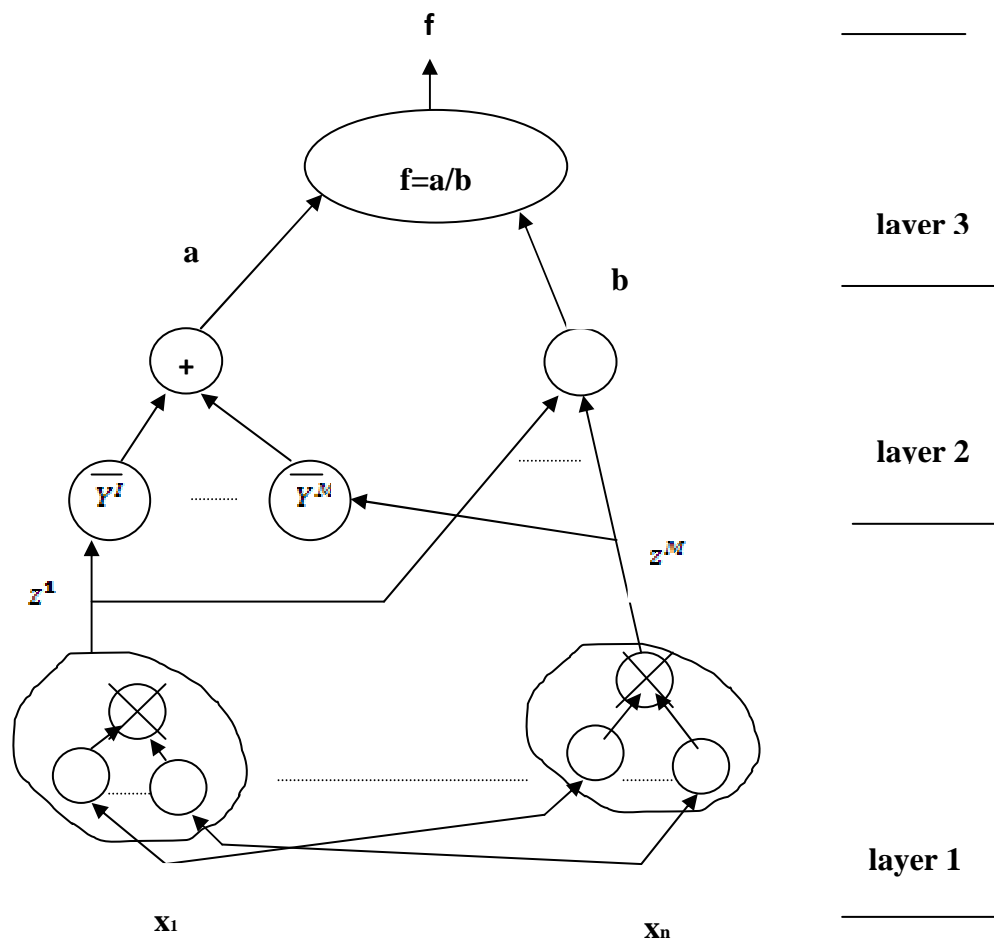
όπου, α_i^1 είναι ένας αυξητικός παράγοντας, x_i^{-1} είναι το σημείο στο οποίο η συμμετοχική συνάρτηση φτάνει τη μέγιστη τιμή της, και σ_i^1 είναι το εύρος της συνάρτησης Caussian.

Σύστημα συγκεκριμένης λογικής

Το σύστημα συγκεκριμένης λογικής με κέντρο μέσου αποσυγκεχυμενοποιητή, η παραγωγή συμπερασματικού κανόνα, ο συγκεχυμενοποιητής singleton και η συμμετοχική συνάρτηση Caussian έχουν την ακόλουθη μορφή (Εξίσωση 10):

$$f(\underline{x}) = \sum_{l=1}^M \mu_l \cdot \frac{a_l}{b_l} \exp\left(-\frac{(x_i - x_i^l)^2}{\sigma_l^2}\right)$$

Τα συστήματα συγκεκριμένης λογικής στη μορφή της Εξίσωσης 10 είναι καθολικοί προσεγγιστές, οι οποίοι είναι ικανοί να προσεγγίζουν ομοιόμορφα οποιαδήποτε μη γραμμική συνάρτηση πέρα από τη U, σε οποιονδήποτε βαθμό ακριβείας, αν η U είναι συνεπτυγμένη. Κατά συνέπεια, μπορεί να αναπτυχθεί ένας εκπαιδευτικός αλγόριθμος πίσω διάδοσης για αυτού του είδους τα συστήματα συγκεκριμένης λογικής. Παρατηρώντας, τη λειτουργική μορφή της εξίσωσης 10, κάποιος μπορεί να δει ότι παρουσιάζεται σαν ένα δίκτυο τριών επιπέδων εμπρόσθιας τροφοδοσίας, όπως παρουσιάζεται στο Σχήμα 16.



$$\mu = \exp\left[-\frac{(x - x_i^l)^2}{\sigma_i^2}\right]$$

Σχήμα 16: Δικτυακή αναπαράσταση των συστημάτων συγκεκριμένης λογικής

Με αυτή την δικτυακή αναπαράσταση των συστημάτων συγκεκριμένης λογικής, γίνεται σαφής η εφαρμογή της ιδέας της πίσω (ανάστροφης) διάδοσης για να ρυθμίσουμε τις παραμέτρους Y , X , και σ , για να εκπαιδευτεί το σύστημα συγκεκριμένης λογικής. Η προέλευση του αλγόριθμου πίσω διάδοσης δίνεται μετά.

ΠΑΡΑΤΗΡΗΣΗ: Από αυτό το σημείο και παραπέρα το Y ισούται με τη μπάρα y , και το X ισούται με την μπάρα x .

Εκπαιδευτικός αλγόριθμος πίσω διάδοσης για συστήματα συγκεκριμένης λογικής

Υποθέτουμε ότι μας δίνεται ένα ζευγάρι εισόδου-εξόδου (x^p, d^p) , η αποστολή μας είναι να ορίσουμε ένα σύστημα συγκεκριμένης λογικής $f(x)$ στη μορφή της εξίσωσης 10, όπως είναι η:

$$e^p = 0.5 [f(x^p) - d^p]^2 \quad (\text{Εξίσωση 11})$$

που ελαχιστοποιείται. Θεωρούμε, ότι το $a_i^l = 1$ και το M είναι δεδομένο: Συνεπώς, το πρόβλημα είναι η εκπαίδευση των παραμέτρων Y^l , X_i^l , και σ_i^l , τέτοια ώστε η e^p να ελαχιστοποιείται.

ΠΑΡΑΤΗΡΗΣΗ: Στην ακόλουθη χρησιμοποιούμε e , f , και d για να δηλώσουμε υποδηλώσουμε τα e^p , $f(x^p)$, και d^p , αντίστοιχα.

Για να εκπαιδύσουμε την Y χρησιμοποιούμε:

$$Y^I(k+1) = Y^I(k) - a(de/dY^I) \mid k \text{ (Εξίσωση 12)}$$

όπου, $I = 1, 2, 3, \dots, M$, $k = 0, 1, 2, \dots$, και a είναι σταθερό βηματικό μέγεθος. Από το Σχήμα 16 βλέπουμε ότι το f (και άρα το e) εξαρτάται από το Y^I μόνο μέσω του a , όπου $f=a/b$, $a = \sum_{j=1}^M (Y^I z^j)$, $b = \sum_{j=1}^M (z^j)$ και $z^j = \prod_{i=1}^n (\exp(-(x_i - X_i^j)/\sigma_i^j)^2)$. Συνεπώς, χρησιμοποιώντας τον αλυσιδωτό κανόνα, έχουμε:

$$de/dY^I = (f-d)(df/da)(da/dY^I) = (f-d)z^I/b \text{ (Εξίσωση 13)}$$

Αντικαθιστώντας την εξίσωση 13 στην 12, αποκτούμε τον εκπαιδευτικό αλγόριθμο για Y^I :

$$Y^I(k+1) = Y^I(k) - a(f-d)z^I / b \text{ (Εξίσωση 14)}$$

όπου, $I=1,2,\dots,M$ και $k=0,1,2,\dots$

Για να εκπαιδύσουμε το X_i^I , χρησιμοποιούμε:

$$X_i^I(k+1) = X_i^I(k) - a(de/dX_i^I) \mid k \text{ (Εξίσωση 15)}$$

όπου, $i=1, 2, \dots, n$, $I = 1, 2, 3, \dots, M$, $k = 0, 1, 2, \dots$. Από το Σχήμα 16 βλέπουμε ότι το f (και άρα το e) εξαρτάται από το X_i^I μόνο μέσω του z^I . Συνεπώς, χρησιμοποιώντας τον αλυσιδωτό κανόνα, έχουμε:

$$de/dX_i^I = (f - d)(df/dz^I)(dz^I/dX_i^I) = (f-d)((Y^I-f)/b) z^I (2(x_i^p - X_i^I)/\sigma_i^{I2}) \text{ (Εξίσωση 16)}$$

Αντικαθιστώντας την εξίσωση 16 στην 15, αποκτούμε τον εκπαιδευτικό αλγόριθμο για X_i^I :

$$X_i^I(k+1) = X_i^I(k) - a ((f-d)/b) (Y^I-f)/b) z^I (2(x_i^p - X_i^I(k))/\sigma_i^{I2}(k)) \text{ (Εξίσωση 17)}$$

όπου, $i=1,2,\dots,n$, $I=1,2,\dots,M$, και $k=0,1,2,\dots$

Χρησιμοποιώντας την ίδια μέθοδο, όπως προηγουμένως, αποκτούμε το ακόλουθο εκπαιδευτικό αλγόριθμο για σ_i^l (Εξίσωση 18):

$$\sigma_i^l(k+1) = \sigma_i^l(k) - \text{ade}/d \sigma_i^l \mid k = \sigma_i^l(k) - (a(f-d)/b) (Y^l - f) z^l (2(x_i^p - X_i^l(k))^2 / \sigma_i^{l3}(k))$$

όπου, $i=1,2,\dots,n$, $I=1,2,\dots,M$, και $k=0,1,2,\dots$

Ο εκπαιδευτικός αλγόριθμος (Εξ. 14), (Εξ. 17), (Εξ. 18) εκτελεί μία διαδικασία σφάλματος πίσω διάδοσης. Για να εκπαιδεύσει το Y^l , το ομαλοποιημένο σφάλμα $(f-d)/b$ διαδίδεται αντίστροφα στο επίπεδο του Y^l . Μετά το Y^l ενημερώνεται χρησιμοποιώντας την εξίσωση 14, στην οποία το z^l είναι η είσοδος στο Y^l .

Για να εκπαιδευτεί το X_i^l και το σ_i^l , το ομαλοποιημένο σφάλμα $(f-d)/b$ φορές $(Y-f)$ και το z^l διαδίδεται αντίστροφα στην μονάδα διεργασίας του επιπέδου 1, του οποίου η έξοδος είναι το z^l . Στη συνέχεια το X_i^l και το σ_i^l ενημερώνονται, χρησιμοποιώντας τις εξισώσεις 17 και 18 αντίστοιχα, στις οποίες οι εναπομείνουσες μεταβλητές X_i^l , x_i^p , και σ_i^l , μπορούν να ληφθούν τοπικά.

Η εκπαιδευτική διαδικασία για το συγκεκριμένο σύστημα του Σχήματος 16 είναι μία διαδικασία δύο περασμάτων. Πρώτα, για μία δεδομένη είσοδο x^p , υπολογίζουμε προς τα εμπρός κατά μήκος του δικτύου (το οποίο είναι το σύστημα συγκεκριμένης λογικής) για να αποκτήσουμε τα z^l ($I = 1, 2, \dots, M$), a , b και f . Στη συνέχεια εκπαιδεύουμε τις παραμέτρους του δικτύου Y^l , X_i^l και σ_i^l ($i=1, 2, \dots, n$, $I=1, 2, \dots, M$) εκτελώντας προς τα πίσω, χρησιμοποιώντας τις Εξισώσεις 14, 17 και 18, αντίστοιχα. Για πολλαπλά ζεύγη εισόδου-εξόδου, παίρνουμε (x^p, d^p) με $p=1,2,\dots$, μπορούμε να εκπαιδεύσουμε το σύστημα για έναν ή περισσότερους παλινδρομικούς (μπρος-πίσω) κύκλους για ένα ζεύγος εισόδου-εξόδου πριν μετακινηθούμε στο επόμενο ζεύγος.

5.5 Συζήτηση και συμπεράσματα για τον συγκεκριμένο έλεγχο

Ένας αριθμός πλεονεκτημάτων και αδυναμιών παρατηρήθηκαν κατά τη διάρκεια της έρευνας στο πεδίο του συγκεκριμένου ελέγχου κατά την διάρκεια του ακαδημαϊκού έτους.

Οι συγκεκριμένοι ελεγκτές πρέπει να χρησιμοποιούνται μόνο σε πολύ πολύπλοκες και υψηλής μη-γραμμικότητας εφαρμογές, όπου δεν μπορούν να εφαρμοστούν οι κλασσικοί ελεγκτές. Ο λόγος είναι ότι, ένας μεγάλος αριθμός παραδοσιακών

ελεγκτών για ένα ευρύ φάσμα εφαρμογών έχει ήδη αναπτυχθεί, και γενικά, αποδίδουν καλύτερα στον έλεγχο, σε σύγκριση με τους συγκεχυμένους ελεγκτές.

Αυτό, όμως, δεν συμβαίνει όταν μία εφαρμογή περιλαμβάνει πολύπλοκα συστήματα, υψηλής μη-γραμμικότητας, τα οποία δεν μπορούν ή είναι πολύ δύσκολο να περιγραφούν από ένα μαθηματικό σύνολο εξισώσεων. Αυτό κάνει τη σχεδίαση ενός παραδοσιακού ελεγκτή πολύ δύσκολη, αν όχι αδύνατη, και αυτή είναι η περίπτωση όπου, οι συγκεχυμένοι ελεγκτές μπορούν να έχουν σπουδαία χρησιμότητα.

Η εξειδικευμένη γνώση είναι η βάση ενός συγκεχυμένου ελεγκτή. Κάποιος ειδικός είναι απαραίτητος, ώστε να παρέχει όλες τις σχετικές πληροφορίες για τη δομή, τη λειτουργία, τη δυναμική, και τους στόχους του συστήματος για να ελεγχθεί. Επίσης, απαιτείται να προσδιοριστούν όλες οι απαραίτητες δράσεις ελέγχου που πρέπει να πραγματοποιηθούν για κάθε συγκεκριμένη κατάσταση του συστήματος.

Το επόμενο βήμα θα είναι να δημιουργηθεί ένας αριθμός συμμετοχικών συναρτήσεων και ένα σύνολο κανόνων για να ελέγχουν τον συγκεχυμένο ελεγκτή, εξαρτώμενοι μόνο από τις πληροφορίες που παρέχονται από τον ειδικό.

Οι συγκεχυμένοι ελεγκτές μπορούν να χρησιμοποιηθούν σε μία εφαρμογή, χωρίς την ανάγκη μαθηματικής περιγραφής του συστήματος. Αυτό τους επιτρέπει να χρησιμοποιηθούν με συστήματα υψηλής μη-γραμμικότητας, όπου είναι πολύ δύσκολο να δημιουργηθούν μαθηματικές αναπαραστάσεις.

Από την άλλη μεριά, υπάρχει ένας αριθμός κρίσιμων λειτουργιών (επιχειρήσεων) (έλεγχος πτήσεων, παραγωγή πυρηνικής ενέργειας), όπου είναι πολύ δύσκολο να δημιουργηθεί ένας συντελεστής με όλες τις απαραίτητες πληροφορίες, για να ορίσουν συγκεκριμένα τη λειτουργία αυτών των συστημάτων και να καθορίσουν τις κατάλληλες δράσεις ελέγχου.

Ένα άλλο σημείο, που πρέπει να βελτιωθεί είναι ότι, οι συγκεχυμένοι ελεγκτές από μόνοι τους δεν είναι καθόλου προσαρμοστικοί, δεν έχουν δηλαδή ικανότητες προσαρμογής και εναρμόνισης. Αν ένας συγκεχυμένος ελεγκτής σχεδιάζεται για να ελέγξει ένα συγκεκριμένο σύστημα, περισσότερο πιθανό θα ήταν να αποτύγχανε να το πράξει, ακόμα και στην περίπτωση που υπάρχει μία ελαφρά αλλαγή στις παραμέτρους.

Αυτό παρακίνησε έναν αριθμό ερευνητών να συνδυάσουν συγκεχυμένους ελεγκτές με ήδη γνωστά συστήματα εναρμόνισης (προσαρμογής), όπως νευρωνικά δίκτυα και γενετικοί αλγόριθμοι, στοχεύοντας στο να βελτιώσουν τους συγκεχυμένους ελεγκτές, εισάγοντας ένα βαθμό ευελιξίας και κάνοντας τα προσαρμόσιμα.

Τώρα, ένας αριθμός συγκεχυμένων προσαρμοστικών ελεγκτών (συγκεχυμένοι-νευρωνικοί, συγκεχυμένοι και γενετικοί αλγόριθμοι) έχουν αναπτυχθεί, οι οποίοι

διατηρούν τα βασικά πλεονεκτήματα της συγκεκριμένης λογικής και ενισχύονται με ένα βαθμό νοημοσύνης.

Ελεγκτές που μπορούν να λειτουργήσουν με γλωσσικές μεταβλητές, χωρίς την ανάγκη για πολύπλοκες μαθηματικές λειτουργίες, με την ικανότητα να διερευνούν και να εξετάζουν (αίσθηση του περιβάλλοντος) και να παίρνουν τις κατάλληλες αποφάσεις (σαρώνουν τη βάση δεδομένων τους και παίρνουν τις απαιτούμενες ενέργειες ελέγχου).

Η τρέχουσα έρευνα πραγματοποιείται στον ευφυή έλεγχο, στοχεύοντας στη σχεδίαση ελεγκτών, οι οποίοι μπορούν να δημιουργούν νέους κανόνες, εκτός από αυτούς που είναι ήδη εγκατεστημένοι στη βάση δεδομένων τους. Αυτοί οι ελεγκτές πρέπει να μπορούν να εκτιμήσουν μία κατάσταση και να μπορούν λαμβάνουν την κατάλληλη ενέργεια ελέγχου, ακόμα κι αν αυτό σημαίνει ότι, απαιτείται τότε να οριστεί ένας νέος κανόνας ή απαιτείται μία τροποποίηση στις συμμετοχικές συναρτήσεις του ελεγκτή.

BIBLIOGRAPHY

1. Neuro-Fuzzy and Soft Computing.

J.-S.R. Jang

C.-T. Sun

E. Mizutani

2. Fuzzy Logic Toolbox. The Mathworks Inc.

3. Introduction to Fuzzy Logic and Neural Networks.

S. Kartalopoulos

4. Fuzzy-Neural Control.

J.Nie

R. Linkens

5. Neural Networks, (teaching material from the University of Central Lancashire)

M.R. Varley

6. Adaptive Fuzzy Systems and Control.

Li-Xin Wang

7. An Introduction to Fuzzy Control.

D. Driankov

H. Hellendoorn

M. Reinfrank

8. Fuzzy Rule-Based Modeling with Applications in Geophysical, Biological and Engineering Systems.

Andras Bardossy

Lucien Duckstein