



**ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ  
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ  
ΔΙΟΙΚΗΣΗΣ**

## **ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**Αυτόνομη πλοήγηση ρομποτικών οχημάτων με χρήση  
ευφρών τεχνικών**

**ΙΩΑΝΝΗΣ ΧΡΟΝΟΠΟΥΛΟΣ**

**ΑΜ: 2015010161**

Χανιά, 2022



## **ΕΥΧΑΡΙΣΤΙΕΣ**

Η σελίδα αυτή μπορεί να παραλειφθεί.

## ΠΕΡΙΛΗΨΗ ΔΙΠΛΩΜΑΤΙΚΗΣ

Σκοπός της παρούσας εργασίας είναι η μελέτη, σύγκριση και αξιολόγηση του τρόπου με τον οποίο ένα αυτόνομο ρομποτικό όχημα διαφορεικής κίνησης, κινείται σε άγνωστο περιβάλλον, με ή χωρίς στατικά εμπόδια, ελεγχόμενο από ελεγκτές βασισμένους σε μεθοδολογίες ευφυούς ελέγχου (Ασαφής λογική, Νευρωνικά Δίκτυα). Το όχημα κινείται σε διαφορετικές αρένες προσομοίωσης, ακολουθώντας μια προκαθορισμένη διαδρομή από σημεία – στόχους. Για το σκοπό αυτό, δημιουργήθηκαν, με τη χρήση κατάλληλου λογισμικού, προσομοιωμένα περιβάλλοντα ποικίλης πολυπλοκότητας και δοκιμάστηκαν σε αυτά διαφορετικοί ελεγκτές. Οι ελεγκτές αξιολογήθηκαν και συγκρίθηκαν μεταξύ τους ανά κατηγορία, με βάση συγκεκριμένα κριτήρια και μελετήθηκε η επίδραση της δομής τους στη συμπεριφορά του οχήματος.

## **SUMMARY**

In the context of this work, we have developed intelligent controllers, based on Fuzzy Logic and Neural Networks, that allow a differential drive robotic vehicle, to navigate in a pre-constructed simulated environment. The vehicle moves and follows a pre-determined trajectory by simultaneously avoiding unknown obstacles. To evaluate the proposed approach, we have developed simulated scenarios with incremental difficulty. Based on these scenarios, we have performed an analysis, to study how the structure of the individual controllers affects the overall performance. Extensive simulated results are presented to compare and highlight the efficiency of the individual controllers.

## ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 1 .....	1
ΕΙΣΑΓΩΓΗ .....	1
1.1      ΑΝΤΙΚΕΙΜΕΝΟ ΤΗΣ ΕΡΓΑΣΙΑΣ .....	1
ΚΕΦΑΛΑΙΟ 2 .....	3
ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΕΠΙΣΚΟΠΗΣΗ .....	3
2.1 ΕΙΣΑΓΩΓΗ.....	3
2.2 ΑΣΑΦΗΣ ΛΟΓΙΚΗ .....	3
2.3 ΑΣΑΦΗΣ ΕΛΕΓΧΟΣ .....	4
2.3.1 ΒΑΣΙΚΕΣ ΑΡΧΕΣ ΑΣΑΦΟΥΣ ΕΛΕΓΧΟΥ.....	4
2.3.2 ΑΣΑΦΗ ΜΟΝΤΕΛΑ ΣΥΜΠΕΡΑΣΜΟΥ MAMDANI-SUGENO .....	5
2.3.3 ΕΦΑΡΜΟΓΕΣ ΤΗΣ ΑΣΑΦΟΥΣ ΛΟΓΙΚΗΣ ΣΤΟΝ ΕΛΕΓΧΟ .....	7
2.3.3.1 ΓΕΝΙΚΕΣ ΕΦΑΡΜΟΓΕΣ .....	7
2.3.3.2 ΑΣΑΦΗΣ ΕΛΕΓΧΟΣ ΣΤΗΝ ΠΛΟΗΓΗΣΗ ΑΥΤΟΝΟΜΩΝ ΡΟΜΠΟΤΙΚΩΝ ΟΧΗΜΑΤΩΝ .....	7
2.4 ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ .....	9
2.5 ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ ΣΤΟΝ ΕΛΕΓΧΟ .....	14
2.5.1 ΓΕΝΙΚΕΣ ΕΦΑΡΜΟΓΕΣ .....	15
2.5.2 ΕΦΑΡΜΟΓΕΣ ΣΤΗΝ ΕΥΡΥΤΕΡΗ ΡΟΜΠΟΤΙΚΗ.....	15
2.5.3 ΕΦΑΡΜΟΓΕΣ ΣΤΟΝ ΕΛΕΓΧΟ ΑΥΤΟΝΟΜΩΝ ΟΧΗΜΑΤΩΝ .....	16
ΚΕΦΑΛΑΙΟ 3 .....	17
ΒΑΣΙΚΕΣ ΑΡΧΕΣ ΚΑΙ ΑΛΓΟΡΙΘΜΟΙ ΚΙΝΗΣΗΣ-ΕΛΕΓΧΟΥ .....	17
3.1 ΕΙΣΑΓΩΓΗ.....	17
3.2 ΘΕΣΗ ΚΑΙ ΜΟΝΤΕΛΟ ΚΙΝΗΣΗΣ ΟΧΗΜΑΤΟΣ ΔΙΑΦΟΡΙΚΗΣ ΚΙΝΗΣΗΣ ..	17
3.3 ΡΟΜΠΟΤΙΚΟ ΟΧΗΜΑ PIONEER 3 – DX .....	19
3.4 ΠΕΡΙΒΑΛΛΟΝ ΠΡΟΣΟΜΟΙΩΤΗ CORPEDIA SIMULATOR.....	21
3.5 ΣΥΝΔΕΣΗ ΠΡΟΣΟΜΟΙΩΤΗ CORPEDIA SIMULATOR ΜΕ ΤΟ ΛΟΓΙΣΜΙΚΟ MATLAB .....	23
3.6 ΑΛΓΟΡΙΘΜΟΣ/ΚΩΔΙΚΑΣ ΔΙΑΧΕΙΡΗΣΗΣ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΠΡΟΣΟΜΟΙΩΣΗΣ .....	24

3.7 ΑΛΓΟΡΙΘΜΟΣ/ΚΩΔΙΚΑΣ ΚΙΝΗΣΗΣ ΡΟΜΠΟΤΙΚΟΥ ΟΧΗΜΑΤΟΣ .....	27
ΚΕΦΑΛΑΙΟ 4 .....	32
ΥΛΟΠΟΙΗΣΗ ΣΕΝΑΡΙΩΝ ΜΕ ΧΡΗΣΗ ΑΣΑΦΟΥΣ/ΝΕΥΡΩΝΙΚΟΥ ΕΛΕΓΚΤΗ .....	32
4.1 ΕΙΣΑΓΩΓΗ.....	32
4.2 ΣΕΝΑΡΙΑ ΚΛΙΜΑΚΟΥΜΕΝΗΣ ΠΟΛΥΠΛΟΚΟΤΗΤΑΣ.....	32
4.3 ΑΣΑΦΕΙΣ ΕΛΕΓΚΤΕΣ ΤΥΠΟΥ ΜΑΜΔΑΝΙ ΓΙΑ ΚΙΝΗΣΗ PIONEER 3-DX..	35
4.3.1 ΚΑΤΑΣΚΕΥΗ ΑΣΑΦΩΝ ΕΛΕΓΚΤΩΝ ΓΙΑ ΚΙΝΗΣΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΧΩΡΙΣ ΕΜΠΟΔΙΑ .....	36
4.3.2 ΠΡΟΣΟΜΟΙΩΣΗ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΑΣΑΦΩΝ ΕΛΕΓΚΤΩΝ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΧΩΡΙΣ ΕΜΠΟΔΙΑ .....	45
4.3.3 ΚΑΤΑΣΚΕΥΗ ΑΣΑΦΩΝ ΕΛΕΓΚΤΩΝ ΓΙΑ ΚΙΝΗΣΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΜΕ ΕΜΠΟΔΙΑ.....	48
4.3.4 ΠΡΟΣΟΜΟΙΩΣΗ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΑΣΑΦΩΝ ΕΛΕΓΚΤΩΝ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΜΕ ΕΜΠΟΔΙΑ.....	58
4.4 ΚΑΤΑΣΚΕΥΗ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ – ΝΕΥΡΩΝΙΚΩΝ ΕΛΕΓΚΤΩΝ ΓΙΑ ΚΙΝΗΣΗ PIONEER 3-DX .....	65
4.4.1 ΕΚΠΑΙΔΕΥΣΗ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ ΕΛΕΓΧΟΥ ΓΙΑ ΚΙΝΗΣΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΧΩΡΙΣ ΕΜΠΟΔΙΑ .....	65
4.4.2 ΠΡΟΣΟΜΟΙΩΣΗ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΚΙΝΗΣΗΣ ΟΧΗΜΑΤΟΣ ΜΕ ΝΕΥΡΩΝΙΚΟ ΕΛΕΓΚΤΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΧΩΡΙΣ ΕΜΠΟΔΙΑ.....	66
4.4.3 ΕΚΠΑΙΔΕΥΣΗ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ ΕΛΕΓΧΟΥ ΓΙΑ ΚΙΝΗΣΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΜΕ ΕΜΠΟΔΙΑ.....	68
4.4.4 ΠΡΟΣΟΜΟΙΩΣΗ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΚΙΝΗΣΗΣ ΟΧΗΜΑΤΟΣ ΜΕ ΝΕΥΡΩΝΙΚΟ ΕΛΕΓΚΤΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΜΕ ΕΜΠΟΔΙΑ .....	69
ΚΕΦΑΛΑΙΟ 5 .....	78
ΣΥΜΠΕΡΑΣΜΑΤΑ/ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ .....	78
5.1 ΣΥΜΠΕΡΑΣΜΑΤΑ .....	78
5.2 ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ .....	79
ΒΙΒΛΙΟΓΡΑΦΙΑ .....	80

# ΚΕΦΑΛΑΙΟ 1

## ΕΙΣΑΓΩΓΗ

### 1.1 ΑΝΤΙΚΕΙΜΕΝΟ ΤΗΣ ΕΡΓΑΣΙΑΣ

Από την Αναγέννηση μέχρι τη βιομηχανική εποχή και από εκεί μέχρι τη μετα-βιομηχανική εποχή του σήμερα, η ανθρωπότητα διένυσε μια πορεία χρόνων, προσπαθώντας να ερμηνεύσει αρχικά τη φύση και να κυριαρχήσει στη συνέχεια επάνω σε αυτή. Η στροφή στην επιστήμη και στην κατανόηση του κόσμου γύρω μας, έθεσε ως προτεραιότητα την αναζήτηση και τεκμηρίωση των φυσικών φαινομένων και της κοσμικής ζωής, έχοντας ως πυξίδα την επίλυση των πάγιων υπαρξιακών ερωτημάτων του ανθρώπινου είδους και υπήρξε το εργαλείο για το άλμα από την προ-νεωτερική ανθρωποκεντρική αντίληψη του σύμπαντος και τις μεσαιωνικές δεισιδαιμονίες.

Φτάνοντας στο σήμερα και κοιτώντας πίσω, μπορεί κανείς να παρατηρήσει τη ραγδαία τεχνολογική ανάπτυξη από τη βιομηχανική επανάσταση και μετά. Έχοντας ως αφετηρία τους κινητήρες εσωτερικής καύσης και τα πρώτα οχήματα και τρανζίστορ, τα οποία υπήρξαν συνοδοιπόροι και βοηθοί του ανθρώπου στη μετακίνηση και στη μαζική παραγωγή, πλέον η ανθρωπότητα μπορεί να μιλά για τρισδιάστατες εκτυπώσεις, αυτόματα μηχανήματα παραγωγής και αυτόνομα οχήματα, των οποίων η λειτουργία αντικαθιστά ή μειώνει αισθητά τη συμμετοχή του ανθρώπινου παράγοντα.

Σε αυτήν την κατεύθυνση της αυτοματοποίησης και αυτονομίας και με τη βοήθεια της τεχνολογικής προόδου των τελευταίων δεκαετιών, τα ρομποτικά οχήματα φαίνεται να διαδραματίζουν κεντρικό ρόλο στις εξελίξεις. Οι ανάγκες και η θέληση για αντικατάσταση του ανθρώπου- χειριστή και του ανθρώπινου λάθους ή για επέμβαση και εξερεύνηση σε περιβάλλοντα που η ανθρώπινη πρόσβαση είναι πολύ δύσκολη έως μη εφικτή, οδήγησε στη μελέτη και εξέλιξη διάφορων τεχνικών και μέσων, σε ότι αφορά την υλικοτεχνική υποστήριξη των οχημάτων και το λειτουργικό τους. Στόχος σε κάθε περίπτωση είναι η αυτόνομη πλοήγηση ενός ρομποτικού οχήματος σε άγνωστο περιβάλλον, η μελέτη του τρόπου ελέγχου του, ο σχεδιασμός της διαδρομής που ακολουθεί και ο τρόπος που αντιδρά και διαχειρίζεται τις μεταβολές του εξωτερικού περιβάλλοντος. Το όχημα καλείται δηλαδή, να διανύσει μια διαδρομή από ένα αρχικό σημείο σ' ένα τελικό σημείο, με πιθανά



ενδιάμεσα διαδοχικά σημεία – στόχους ή να ακολουθήσει μια προσχεδιασμένη διαδρομή, έχοντας παράλληλα ως προτεραιότητα την ελαχιστοποίηση του κόστους μετακίνησης (ως προς το χρόνο, την απόσταση κλπ.). Παράλληλα, οφείλει να αλληλοεπιδρά συνέχεια με το περιβάλλον του και να αξιοποιεί κατάλληλα τα δεδομένα που συλλέγει, ώστε να αποφεύγει στατικά ή δυναμικά εμπόδια και εξωτερικό «θόρυβο» και να επιτυγχάνει ομαλή και γρήγορη κίνηση στο χώρο.

Σκοπός της παρούσας εργασίας είναι η μελέτη, σύγκριση και αξιολόγηση του τρόπου με τον οποίο ένα αυτόνομο ρομποτικό όχημα διαφορικής κίνησης, κινείται σε άγνωστο περιβάλλον, με ή χωρίς στατικά εμπόδια, ελεγχόμενο από ελεγκτές βασισμένους σε μεθοδολογίες ευφυούς ελέγχου (Ασαφής λογική, Νευρωνικά Δίκτυα). Το όχημα κινείται σε διαφορετικές αρένες προσομοίωσης, ακολουθώντας μια προκαθορισμένη διαδρομή από σημεία – στόχους. Για το σκοπό αυτό, δημιουργήθηκαν, με τη χρήση κατάλληλου λογισμικού, προσομοιωμένα περιβάλλοντα ποικίλης πολυπλοκότητας και δοκιμάστηκαν σε αυτά διαφορετικοί ελεγκτές. Οι ελεγκτές αξιολογήθηκαν και συγκρίθηκαν μεταξύ τους ανά κατηγορία, με βάση συγκεκριμένα κριτήρια και μελετήθηκε η επίδραση της δομής τους στη συμπεριφορά του οχήματος.

Αναλυτικά, στο δεύτερο κεφάλαιο γίνεται βιβλιογραφική αναφορά στις αρχές της ασαφούς λογικής και των νευρωνικών δικτύων, στα χαρακτηριστικά και στο σχεδιασμό ενός ασαφούς ελεγκτή και αναλύονται ασαφείς και νευρωνικοί ελεγκτές που έχουν σχεδιαστεί και βρει εφαρμογή, από την υπάρχουσα βιβλιογραφία.

Στο τρίτο κεφάλαιο, παρατίθενται οι βασικές αρχές και το μοντέλο διαφορικής κίνησης έντροχων οχημάτων, δίνονται τα χαρακτηριστικά του οχήματος που χρησιμοποιείται στην παρούσα εργασία, λαμβάνει χώρα περιγραφή του περιβάλλοντος του προσομοιωτή και της σύνδεσης του με το λογισμικό της Matlab και αναλύονται εκτενώς οι απαραίτητοι αλγόριθμοι για την εκπόνηση των προσομοιώσεων.

Στο τέταρτο κεφάλαιο, παρουσιάζονται οι αρένες κλιμακωτής πολυπλοκότητας που χρησιμοποιούνται στις προσομοιώσεις, αναλύονται εκτενώς τα χαρακτηριστικά των ασαφών και νευρωνικών ελεγκτών που δημιουργήθηκαν, παρατίθενται με τη μορφή πινάκων και διαγραμμάτων τα αποτελέσματα των προσομοιώσεων και σχολιάζονται αναλυτικά τα αποτελέσματα τους.

Στο πέμπτο κεφάλαιο, δίνεται μια συμπερασματική σύνοψη της εργασίας και αναφέρονται σκέψεις για μελλοντικές επεκτάσεις της.

## ΚΕΦΑΛΑΙΟ 2

# ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΕΠΙΣΚΟΠΗΣΗ

### 2.1 ΕΙΣΑΓΩΓΗ

Στο παρόν κεφάλαιο, γίνεται παρουσίαση του θεωρητικού πλαισίου της ασαφούς λογικής, της δομής των ασαφών ελεγκτών στον έλεγχο συστημάτων αλλά και έντροχων οχημάτων, καθώς και των εφαρμογών του ασαφούς ελέγχου σε ρεαλιστικές συνθήκες. Ακόμη, αναλύονται τα νευρωνικά δίκτυα και οι βασικές αρχές που τα διέπουν και περιγράφονται έμπρακτες εφαρμογές ελέγχου με χρήση νευρωνικών δικτύων.

### 2.2 ΑΣΑΦΗΣ ΛΟΓΙΚΗ

Η κλασική θεωρία των συστημάτων αυτομάτου ελέγχου, συνέβαλε στην εξέλιξη και δημιουργία πολλών τεχνολογικών εφαρμογών και αγαθών. Ωστόσο, για την εφαρμογή της, είναι αναγκαία η πλήρης γνώση του αναλυτικού μοντέλου του εκάστοτε συστήματος, ενώ παράλληλα, σε σύνθετες διεργασίες, η σχέση διέγερσης-απόκρισης, εμφανίζει μη γραμμικότητα, ανακρίβειες και αντιφάσεις. Τόσο τα δεδομένα αυτά, όσο και οι αυξανόμενες απαιτήσεις της Βιομηχανίας, οδήγησαν στην ανάγκη εύρεσης μιας νέας θεωρίας, μη συμβατικών τεχνικών ελέγχου, οι οποίες είναι γνωστές με τον ευρύτερο όρο “Ευφυής Έλεγχος”.

Στον ευρύτερο αυτό κλάδο και συγκεκριμένα στον κλάδο της υπολογιστικής νοημοσύνης, υπάγονται τα ασαφή συστήματα, ένα κράμα ασαφών συνόλων και ασαφούς λογικής, τα οποία προσπαθούν να δώσουν αποτελεσματική ερμηνεία στην ασάφεια του πραγματικού κόσμου. Η ασαφής λογική, μπορεί να θεωρηθεί ότι είναι μια προσπάθεια μοντελοποίησης της φυσικής γλώσσας και της κοινής λογικής. Αποτελεί μια πλειότιμη επέκταση της Αριστοτέλειας δίτιμης λογικής, η οποία ορίζει ότι, αν μια λογική πρόταση είναι αληθής, τότε δεν είναι ψευδής και αν δεν είναι αληθής τότε είναι ψευδής. [2]

Τη θεωρία των ασαφών συνόλων και τον όρο fuzzy, θεμελίωσε ο Lofti Zadeh το 1965, εισάγοντας τα στη διεθνή βιβλιογραφία. Κεντρική ιδέα της θεωρίας είναι ότι, συνήθως, στον κόσμο που ζούμε τα αντικείμενα γύρω μας ανήκουν σε διάφορα σύνολα με διαφορετικούς βαθμούς συμμετοχής.

Η ασαφής λογική βασίζεται στην επέκταση της έννοιας του κλασικού συνόλου που ορίζεται στο δίτιμο σύνολο  $(0,1)$ , στη γενικευμένη έννοια του ασαφούς συνόλου το οποίο ορίζεται στο πλειότιμο κλειστό διάστημα  $[0,1]$ . Είναι γνωστό επίσης, ότι γενικά ένα κλασικό σύνολο  $A$  ως προς σύνολο αναφοράς  $X$ , μπορεί να αναπαρασταθεί ισοδύναμα μέσω της χαρακτηριστικής συνάρτησής του, δηλαδή:

$$I_A : x \in X \rightarrow I_A(x) = \begin{cases} 1, & \text{όταν } x \in A \\ 0, & \text{όταν } x \notin A \end{cases}, \quad \text{δηλ. } A = \{x \in X \mid I_A(x) = 1\} \quad (2.1)$$

Μέσω της συνάρτησης συμμετοχής, επεκτείνεται η έννοια του δίτιμου συνόλου, έτσι ώστε να περιλαμβάνει όλες τις πιθανές καταστάσεις, οι οποίες λεκτικά δεν μπορούν να εκφραστούν με το σύνολο  $\{0,1\}$ :

$$\mu_A : x \in X \rightarrow \mu_A(x) \in [0,1], \quad (2.2)$$

Ο αριθμός  $\mu_A(x) \in [0,1]$  εκφράζει το βαθμό συμμετοχής, με τον οποίο το στοιχείο  $x \in X$  συμμετέχει στο ασαφές υποσύνολο  $A$  του  $X$ . Αναλυτικότερα,  $\mu_A(x) = 1$ , σημαίνει ότι το  $x$  ανήκει ολοκληρωτικά στο  $A$ ,  $\mu_A(x) = 0$ , σημαίνει ότι το  $x$  δεν ανήκει καθόλου στο  $A$  και  $0 < \mu_A(x) < 1$ , σημαίνει ότι το  $x$  ανήκει κατά κάποιο βαθμό στο  $A$ . Η τιμή αλήθειας μιας πρότασης δεν παίρνει μόνον μία από τις τιμές 0 ή 1, αλλά μπορεί να παίρνει και τιμές μεταξύ 0 και 1, οπότε μια πρόταση που δεν είναι αληθής δεν σημαίνει αναγκαία ότι είναι ψευδής, αλλά μπορεί να είναι μερικά αληθής και μερικά ψευδής. [1]

## 2.3 ΑΣΑΦΗΣ ΕΛΕΓΧΟΣ

### 2.3.1 ΒΑΣΙΚΕΣ ΑΡΧΕΣ ΑΣΑΦΟΥΣ ΕΛΕΓΧΟΥ

Τα ασαφή μοντέλα, σε αντίθεση με τα κλασσικά μοντέλα ελέγχου, έχουν την ικανότητα να επεξεργάζονται ασαφή πληροφορία καθώς και δεδομένα με θόρυβο. Καθώς τα συστήματα αυτά, βασίζονται στην αντίληψη του ανθρώπου να λαμβάνει αποφάσεις με ποιοτικά κριτήρια, μέσω ενός συνόλου απλών λεκτικών κανόνων, η ασαφής λογική μοντελοποιεί το γνωσιακό φάσμα και την εμπειρία ενός έμπειρου χρήστη. Συνεπώς, τα ασαφή συστήματα ελέγχου, αποτελούν μοντέλα βασισμένα στη γνώση (knowledge based system), τα οποία απλοποιούν την πληροφορία και πλησιάζουν την ανθρώπινη λογική και σκέψη. Αυτή η φιλοσοφία δομής του ασαφούς ελέγχου, καθιστά εφικτό τον έλεγχο πολύπλοκων και μη συμβατικών συστημάτων, στα οποία οι κλασικές μέθοδοι ελέγχου δεν μπορούν να εφαρμοστούν ή αποτυγχάνουν [3].

Τα συστήματα ελέγχου, στα οποία εφαρμόζεται η ασαφής λογική, αλλά και τα ευφυή συστήματα ελέγχου εν γένει, είναι δομημένα με μία συγκεκριμένη φιλοσοφία σχεδιασμού, η οποία οριοθετείται από τρεις βασικές αρχές:

- **Ορθότητα:** Η ικανότητα εκτέλεσης των λειτουργικών απαιτήσεων του συστήματος με ασφάλεια.
- **Ευρωστία:** Η ικανότητα του συστήματος να παραμένει λειτουργικό κάτω από μη αναμενόμενες συνθήκες.
- **Επεκτασιμότητα:** Η δυνατότητα επέκτασης του υλικού και του λογισμικού χωρίς επανασχεδίαση του συστήματος από την αρχή.

Ένας ασαφής ελεγκτής απαρτίζεται από τα εξής βασικά στοιχεία:

- **Βάση γνώσης:** Σε αυτήν είναι αποθηκευμένοι οι κανόνες ελέγχου για το έλεγχο της διαδικασίας.
- **Ασαφή σύνολα:** Έχοντας ορίσει τα ασαφή σύνολα είναι δυνατή η μετάφραση των λεκτικών κανόνων της βάσης γνώσης σε μαθηματικούς κανόνες.
- **Ασαφοποιητής:** Αναλαμβάνει την μετατροπή των πραγματικών τιμών των μεταβλητών εισόδου του ελεγκτή σε ασαφή σύνολα.
- **Μηχανισμός συμπερασμού:** Εκεί παράγονται μέσω συνεπαγωγών τα ασαφή σύνολα των συμπερασμάτων.
- **Αποασαφοποιητής:** Τα ασαφή σύνολα των συμπερασμάτων μετατρέπονται σε πραγματικούς αριθμούς έτσι ώστε να είναι δυνατή η μετάδοση της δράσης ελέγχου στην διαδικασία [2].

### 2.3.2 ΑΣΑΦΗ ΜΟΝΤΕΛΑ ΣΥΜΠΕΡΑΣΜΟΥ MAMDANI-SUGENO

Τα Ασαφή Συστήματα Συμπερασμού (Fuzzy Inference Systems (FISs)) τύπου Mamdani και Sugeno, αποτελούν δημοφιλή Ασαφή Μοντέλα, τα οποία χρησιμοποιούνται στον έλεγχο. Ο ασαφής συμπερασμός, όπως έχει ήδη αναφερθεί, είναι η διαδικασία συσχέτισης μιας εισόδου με μία έξοδο, χρησιμοποιώντας τη θεωρία των ασαφών συνόλων. Μέσα από τα μοντέλα αυτά, παρουσιάζεται ο τρόπος λειτουργίας ενός ασαφούς ελεγκτή και αναλύονται τα βασικά του στοιχεία.

Το 1975, ο καθηγητής Ebrahim Mamdani κατασκεύασε ένα από τα πρώτα ασαφή συστήματα, για τον έλεγχο μιας ατμομηχανής μαζί με ένα λέβητα. Ο συμπερασμός τύπου **Mamdani** ακολουθεί την εξής διαδικασία τεσσάρων βημάτων: ασαφοποίηση (fuzzification) των μεταβλητών εισόδου, αξιολόγηση των κανόνων (rule evaluation), συνάθροιση (aggregation) του κανόνα εξόδου και τέλος, αποασαφοποίηση.

Κατά τη διαδικασία της ασαφοποίησης, ορίζεται ο βαθμός στον οποίο οι σαφείς (crisp) είσοδοι του συστήματος ανήκουν σε καθένα από τα ασαφή σύνολα. Κάθε είσοδος, η οποία

έχει αριθμητική αξία μέσα σε ένα εύρος τιμών ορισμένο από το χρήστη, συνδέεται με τις αντίστοιχες συναρτήσεις συμμετοχής [4].

Στη διαδικασία αξιολόγησης των κανόνων, οι ασαφοποιημένες πλέον εισοδοί, υπό τη μορφή συναρτήσεων συμμετοχής, συνδέονται με τις προϋποθέσεις (antecedents) των ασαφών κανόνων. Εάν ένας κανόνας έχει πολλαπλές προϋποθέσεις, τότε, με τη βοήθεια των λογικών τελεστών (KAI ή Ή), εξάγεται ένας αριθμός, ως αποτέλεσμα της αξιολόγησης. Με τη χρήση του τελεστή Ή (OR) αξιολογείται ο διαχωρισμός των προϋποθέσεων ενός κανόνα. Ο τελεστής αυτός, αν θεωρηθούν δύο ασαφή σύνολα A και B ενός κλασικού συνόλου X, εκφράζει την ένωση των δύο αυτών συνόλων, δηλαδή:

$$\mu_{A \cup B}(x) = \max[\mu_A(x), \mu_B(x)] \quad (2.3)$$

Αντίστοιχα, με τη χρήση του τελεστή KAI (AND), αξιολογείται η σύζευξη των προϋποθέσεων ενός κανόνα. Για τα ίδια παραπάνω σύνολα, ο τελεστής εκφράζει την τομή των δύο συνόλων, δηλαδή:

$$\mu_{A \cap B}(x) = \min[\mu_A(x), \mu_B(x)] \quad (2.4)$$

Το αποτέλεσμα της αξιολόγησης των προϋποθέσεων του εκάστοτε κανόνα, συνδέεται έπειτα με τη συνάρτηση συμμετοχής του επακόλουθου αποτελέσματος (consequent) του κανόνα (το σκέλος ΤΟΤΕ...). Στη συνέχεια της διαδικασίας, γίνεται η ένωση όλων των εξόδων, όλων των κανόνων. Στην ουσία, όλες οι συναρτήσεις συμμετοχής της σύνδεσης που περιγράφεται στην προηγούμενη παράγραφο, ενώνονται, σχηματίζοντας ένα ενιαίο ασαφές σύνολο για κάθε μεταβλητή εξόδου αντίστοιχα. Το ενιαίο αυτό σύνολο, απο-ασαφοποιείται, ώστε να προκύψει σαφής (crisp) έξοδος, δηλαδή ένας αριθμός. Υπάρχουν πολλές μέθοδοι για τη διαδικασία αυτή. Η πιο γνωστή, είναι η μέθοδος του **Κέντρου Βάρους**, όπου υπολογίζεται, θεωρητικά, ολοκληρώνοντας το δυσδιάστατο σχήμα του ασαφούς συνόλου και πρακτικά μέσω της σχέσης (2.5), το σημείο που αποτελεί το κέντρο βάρους του αντίστοιχου ασαφούς συνόλου, μέσα σε καθορισμένο χώρο - όρια.

$$K.B = \frac{\sum_{x=\alpha}^b \mu_A(x) * x}{\sum_{x=\alpha}^b \mu_A(x)} \quad (2.5)$$

Αυτή η συνθετότητα στον υπολογισμό των από-ασαφοποιημένων συνόλων του συμπερασμού Mamdani, δεν είναι υπολογιστικά αποτελεσματική. Το γεγονός αυτό, οδήγησε τον Michio Sugeno, στην ανάπτυξη του συμπερασμού τύπου **Sugeno**. Η διαφοροποίηση από τον Mamdani, έγκειται στο ότι, η συνάρτηση συμμετοχής του επακόλουθου αποτελέσματος (consequent) του κανόνα, λαμβάνει μια τιμή σε ένα σημείο του πεδίου ορισμού και είναι μηδέν οπουδήποτε αλλού. Έτσι, αλλάζοντας μόνο το επακόλουθο των κανόνων από ένα ασαφές σύνολο σε μια μαθηματική συνάρτηση της μεταβλητής εισόδου, οι κανόνες σε συστήματα τύπου Sugeno διαμορφώνονται ως εξής:

$$AN \ x = A \ KAI \ y = B \ TOTE \ z = f(x, y) \quad (2.6),$$

Το πιο γνωστό ασαφές μοντέλο Sugeno, κάθε μέλος του επακόλουθου του κάθε κανόνα με μια σταθερά  $k$  ( $z = k$ ).

Το σύστημα αποασαφοποιείται με την εύρεση του σταθμισμένου μέσου ως εξής [5]:

$$\Sigma. M = \frac{\sum \mu(k_i) * k_i}{\sum \mu(k_i)} \quad (2.7)$$

### 2.3.3 ΕΦΑΡΜΟΓΕΣ ΤΗΣ ΑΣΑΦΟΥΣ ΛΟΓΙΚΗΣ ΣΤΟΝ ΕΛΕΓΧΟ

#### 2.3.3.1 ΓΕΝΙΚΕΣ ΕΦΑΡΜΟΓΕΣ

Η πρακτική εφαρμογή των ασαφών ελεγκτών, βρίσκει αντικείμενο τόσο σε συστήματα τα οποία διέπονται από μη γραμμικά μέρη και είναι αδύνατον να ελεγχθούν διαφορετικά, όσο και στη βελτιστοποίηση συμβατικών ελεγκτών. Η εισαγωγή τους στα προϊόντα και τα συστήματα έχει απασχολήσει τόσο τον τομέα της έρευνας, όσο και της βιομηχανίας.

Ένα κομμάτι έρευνας που συναντάται συχνά η δοκιμή ασαφών ελεγκτών αλλά και γενικότερα ευφών τεχνικών, είναι η διαχείριση της εξερχόμενης τάσης ενός μετασχηματιστή σταθερού ρεύματος (DC Converter), όπως στην [6]. Το σύστημα ελέγχου της τάσης εφαρμόζεται με τη βοήθεια ενός μικροεπεξεργαστή, ο οποίος, βασισμένος στον ασαφή έλεγχο, καταφέρνει να σταθεροποιεί την τάση, απέναντι σε ταλαντώσεις που προκαλεί η απότομη εναλλαγή του φορτίου ή της εισερχόμενης τάσης.

Στην [7], χρησιμοποιείται σε εργαστηριακό πλαίσιο, ένας υβριδικός ελεγκτής - συγχώνευση ασαφούς και PID ελεγκτή - για τη βέλτιστη ρύθμιση των παραμέτρων του PID ελεγκτή ενός ρομποτικού βραχίονα. Έτσι, γίνεται η καλύτερη επιλογή των παραμέτρων αυτών, ώστε να μετριαστούν οι ταλαντώσεις στο σύστημα από τις αλλαγές φορτίου, αλλά και να υπάρχει ταχύτερη απόκριση από τον ελεγκτή. Η τεχνολογία αυτή αποτελεί καινοτομία σε έναν δύσκολο τομέα ελέγχου, ο οποίος έχει απασχολήσει τόσο ακαδημαϊκά όσο και σε επίπεδο βιομηχανίας.

#### 2.3.3.2 ΑΣΑΦΗΣ ΕΛΕΓΧΟΣ ΣΤΗΝ ΠΛΟΗΓΗΣΗ ΑΥΤΟΝΟΜΩΝ ΡΟΜΠΟΤΙΚΩΝ ΟΧΗΜΑΤΩΝ

Κάποιοι από τους πιο δύσκολους τομείς εφαρμογής της ρομποτικής αφορούν κίνηση οχημάτων σε επικίνδυνα και άγνωστα περιβάλλοντα. Παραδείγματα αποτελούν η έρευνα και διάσωση σε ξηρά ή θάλασσα, η υποθαλάσσια κατασκευή και χαρτογράφηση, ο κλάδος της μετεωρολογίας και της γεωπονίας και η άμυνα.

Τα μη επανδρωμένα οχήματα, για να ανταπεξέλθουν σε αυτές τις συνθήκες, είναι ικανά να δρουν σε μη δομημένα και αβέβαια περιβάλλοντα, είτε με απομακρυσμένη καθοδήγηση χειριστή, είτε (ημι-) αυτόνομα χωρίς ανθρώπινη παρέμβαση. Για την εκπλήρωση των παραπάνω, τα οχήματα διαθέτουν τα εξής χαρακτηριστικά [8]:

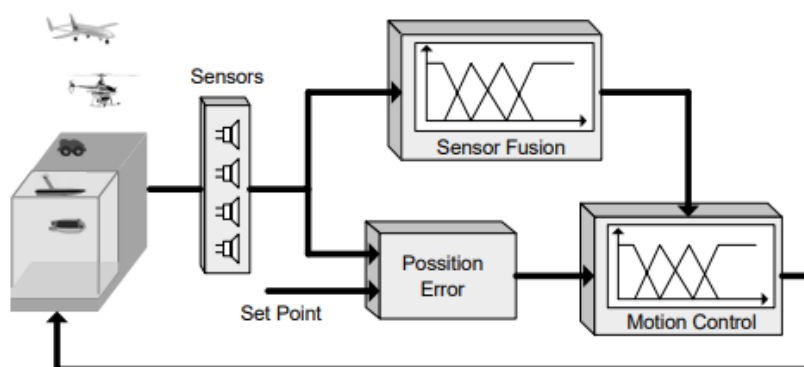
- **Αντίληψη:** η ικανότητα να συλλέγουν και να χρησιμοποιούν τη γνώση που αφορά το περιβάλλον αλλά και τον εαυτό τους. Στην πράξη, η συλλογή πληροφοριών από τους αισθητήρες και η κατάλληλη επεξεργασία τους, για την εξαγωγή χρήσιμης πληροφορίας.
- **Ευφυΐα:** η ικανότητα να λειτουργούν για ικανοποιητικό χρονικό διάστημα χωρίς την ανθρώπινη παρέμβαση, μαθαίνοντας και προσαρμόζοντας τη συμπεριφορά τους.
- **Δράση:** η πλοήγηση μεταξύ σημείων σε δυναμικά περιβάλλοντα χωρίς την ανθρώπινη παρέμβαση, αξιοποιώντας τη συλλογή δεδομένων και την εκμάθηση από αυτά.

Η αυτόνομη πλοήγηση ρομποτικών οχημάτων για να είναι επιτυχής, χρειάζεται τη διαρκή αλληλεπίδραση μεταξύ των παραπάνω βασικών χαρακτηριστικών. Για την αυτόνομη πλοήγηση οχήματος σε δυναμικό περιβάλλον με εμπόδια, η χρήση ελεγκτών ασαφούς λογικής είναι αρκετά συχνή και διαδεδομένη. Η απλή και γρήγορη αναπαράσταση της γνώσης, εκφρασμένη σε λεκτικούς κανόνες και η μη αναγκαία γνώση του μαθηματικού μοντέλου του συστήματος-σε θεωρητικό επίπεδο-, η καταλληλότητα στην αναπαράσταση της ανακρίβειας των δεδομένων των αισθητήρων και ο χαμηλός υπολογιστικός χρόνος- σε πρακτικό επίπεδο-, είναι οι λόγοι που κάνουν τα συστήματα αυτά ιδανικά για την πλοήγηση αυτόνομων οχημάτων.

Η πλοήγηση αυτόνομων ρομποτικών οχημάτων με ασαφή ελεγκτή, έχει μελετηθεί και εφαρμοστεί με διάφορες παραλλαγές στη δομή τους. Οι *Kazuo Tanaka et al.* [9], κατασκεύασαν έναν ελεγκτή για την προς τα πίσω κίνηση ενός οχήματος με τράκτορα, βασισμένο σε παράλληλα επίπεδα, όπου τα γραμμικά μέρη του μοντέλου εισάγονται και ενώνονται με τη μη γραμμικότητα του ενιαίου μοντέλου. Στη [10], προτάθηκε ένας ελεγκτής, ο οποίος βασίζεται στη λογική της αντίληψης και δράσης. Το σύστημα ελέγχου είναι εφοδιασμένο με τα πιθανά σενάρια που θα συναντήσει και τις ανάλογες κινήσεις που θα κάνει ώστε να αντιδράσει. Με αυτό τον τρόπο, προσεγγίζεται η επαφή με πραγματικά δυναμικά περιβάλλοντα, όπου δε μπορεί να είναι γνωστό εκ των προτέρων ένα καθορισμένο- καθαρό από εμπόδια και ατέλειες- μονοπάτι. Στην [11] αναπτύχθηκε παρόμοια τεχνική ελέγχου.

Το πρόβλημα της πλοήγησης οχημάτων χωρίζεται σε δύο μέρη: τη γενική πλοήγηση, δηλαδή την εύρεση ενός μονοπατιού για την οδήγηση προς ένα στόχο και την τοπική πλοήγηση, δηλαδή πώς το όχημα πορεύεται στο μονοπάτι και ταυτόχρονα αποφεύγει εμπόδια, που είναι κοντά ή πάνω σε αυτό. Ένα άλλο σημαντικό θέμα της πλοήγησης, είναι ο τρόπος διαχείρισης των δεδομένων των αισθητήρων. Αυτοί, αποτελούν τον ενδιάμεσο κρίκο μεταξύ οχήματος και εξωτερικού περιβάλλοντος, παρέχοντας ζωντανές πληροφορίες για το τελευταίο, αλλά και για το όχημα το ίδιο. Τα δεδομένα αυτά, μπορεί να εμπεριέχουν θόρυβο, να έχουν αντικρουόμενες πληροφορίες από αισθητήρα σε

αισθητήρα και χρειάζονται ειδική επεξεργασία, ώστε να μπορεί να εξαχθεί ουσιώδης πληροφορία από αυτά και να εφαρμοστεί στην κίνηση του οχήματος. Για την επίτευξη αυτού, οι *Tsourveloudis et al.* [8], πρότειναν την αρχιτεκτονική ενός γενικού μοντέλου πλοήγησης αυτόνομου ρομποτικού οχήματος με ασαφή ελεγκτή, όπως φαίνεται στο σχήμα 2.1.



Σχήμα 2.1 Αρχιτεκτονική μοντέλου πλοήγησης με ασαφή λογική [8]

Τα δεδομένα από τους αισθητήρες, διοχετεύονται σε έναν ασαφή ελεγκτή, ο οποίος επεξεργάζεται τα δεδομένα και ορίζει την πιθανότητα σύγκρουσης με κάποιο εμπόδιο, σε σχέση πάντα με τη θέση και την κατεύθυνση του οχήματος και σε ένα επίπεδο το οποίο υπολογίζει το σφάλμα κατεύθυνσης του οχήματος. Οι πληροφορίες των δύο αυτών επιπέδων, διοχετεύονται ως είσοδοι σε ένα άλλο ασαφή ελεγκτή, ο οποίος εξάγει τις εντολές για την κίνηση του οχήματος [8].

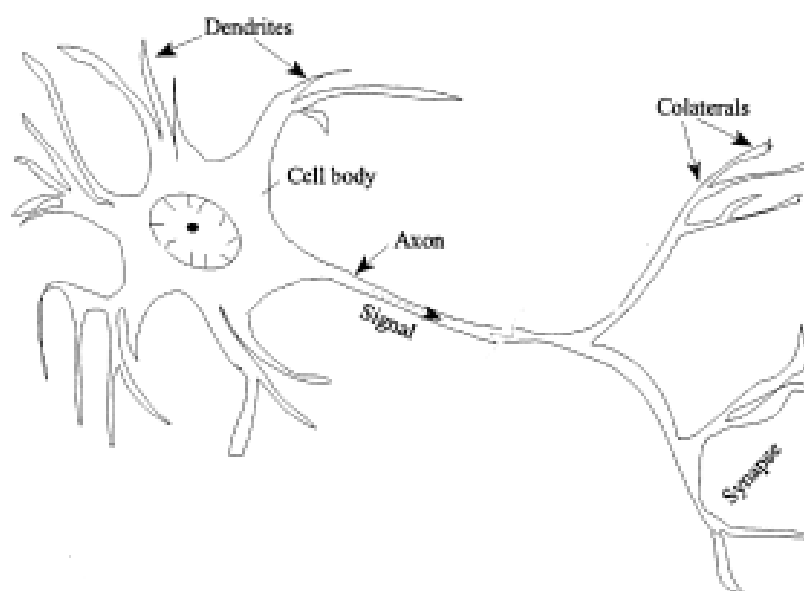
## 2.4 ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Πολλές εργασίες οι οποίες εμπεριέχουν νοημοσύνη ή αναγνώριση μοτίβων είναι εξαιρετικά δύσκολο να αυτοματοποιηθούν, ωστόσο για τους ανθρώπους αποτελεί εύκολη υπόθεση η πραγματοποίησή τους. Για παράδειγμα, οι άνθρωποι αναγνωρίζουν ποικίλα αντικείμενα, τα οποία ξεχωρίζουν ανάμεσα στο μεγάλο όγκο οπτικής πληροφορίας γύρω από αυτά, χωρίς να καταβάλουν μεγάλη προσπάθεια. Είναι λογικό ότι τα υπολογιστικά συστήματα που επιχειρούν παρόμοιες εργασίες ωφελούνται σε μεγάλο βαθμό από την κατανόηση του τρόπου με τον οποίο οι άνθρωποι εκτελούν αυτές τις εργασίες και την προσομοίωση αυτών των διαδικασιών, στο βαθμό που επιτρέπεται από φυσικούς περιορισμούς [12].

Τέτοια συστήματα είναι τα Τεχνητά Νευρωνικά Δίκτυα, τα οποία είναι δημοφιλή, λόγω των αξιοσημείωτων χαρακτηριστικών επεξεργασίας της πληροφορίας του βιολογικού συστήματος που μιμούνται, όπως η μη γραμμικότητα, ο υψηλός παραλληλισμός, η ευρωστία, η ανοχή στο σφάλμα και στην αποτυχία, η μάθηση, η ικανότητα να χειρίζονται

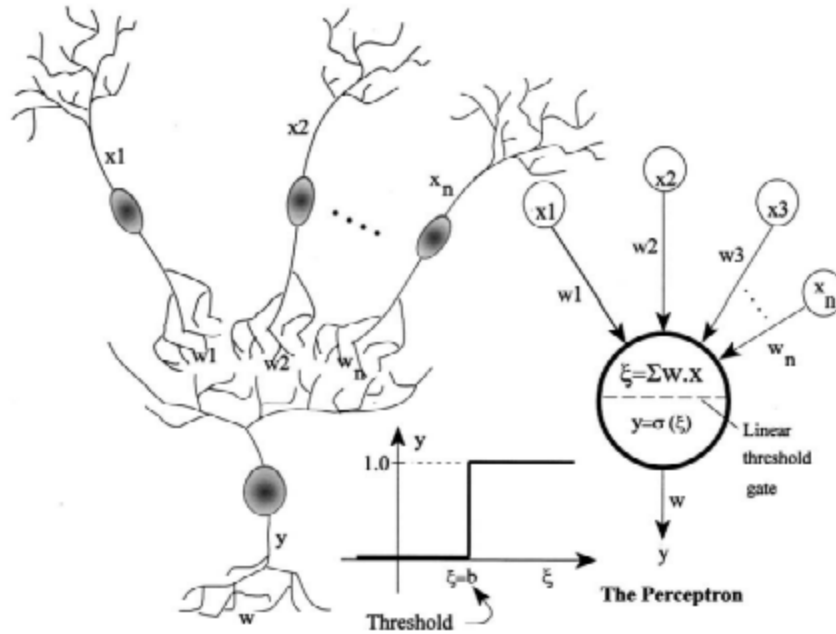


ανακριβείς και ασαφείς πληροφορίες και η ικανότητα γενίκευσης. Παρόλο που αποτελούν μια αφαίρεση των βιολογικών νευρωνικών δικτύων, δεν επιχειρούν να επαναλάβουν τη λειτουργία τους, αλλά να χρησιμοποιήσουν τις πληροφορίες που είναι γνωστές για τη λειτουργικότητά τους, με σκοπό την επίλυση σύνθετων προβλημάτων. Το ανθρώπινο νευρικό σύστημα αποτελείται από δισεκατομμύρια νευρώνες διαφορετικού τύπου και μήκους, ανάλογα με τη θέση τους στο σώμα [13].



Σχήμα 2.2 Απλουστευμένο μοντέλο βιολογικού νευρώνα[13]

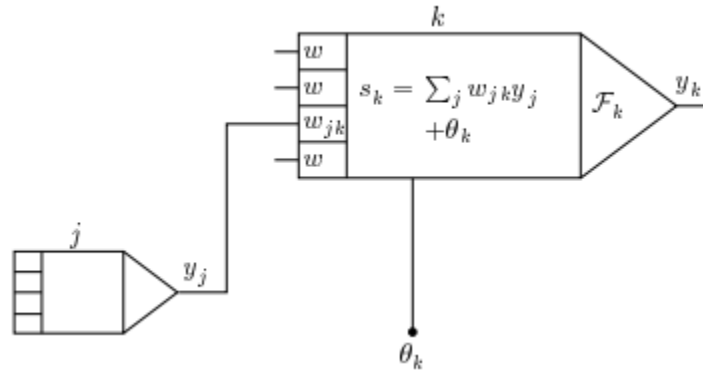
Στο σχήμα 2.2, παρουσιάζεται ένας υπεραπλουστευμένος βιολογικός νευρώνας, με τρία βασικά λειτουργικά μέρη: τους δενδρίτες, το κυτταρικό σώμα και τον άξονα. Οι δενδρίτες λαμβάνουν σήματα από άλλους νευρώνες και τα προωθούν στο κυτταρικό σώμα. Ο άξονας, λαμβάνει τα σήματα από το κυτταρικό σώμα και τα μεταφέρει μέσω των συνάψεων στους δενδρίτες των γειτονικών νευρώνων. Επειδή ένας νευρώνας έχει μεγάλο αριθμό δενδριτών/συνάψεων, μπορεί να λαμβάνει και να μεταφέρει πολλά σήματα ταυτόχρονα. Αυτά τα σήματα μπορούν είτε να βοηθήσουν είτε να αναστείλουν την πυροδότηση του νευρώνα. Η αναλογία μεταξύ τεχνητού και βιολογικού νευρώνα είναι η εξής: οι συνδέσεις μεταξύ των κόμβων αναπαριστούν τους άξονες και τους δενδρίτες, τα βάρη των συνδέσεων αντιστοιχούν στις συνάψεις και τα κατώφλια προσεγγίζουν τη δραστηριότητα στο σώμα του νευρώνα [23]. Με τη βοήθεια του σχήματος 2.3, η αντιστοιχία αυτή γίνεται περισσότερο κατανοητή.



Σχήμα 2.3 Αντιστοιχία βιολογικού και τεχνητού νευρώνα [13]

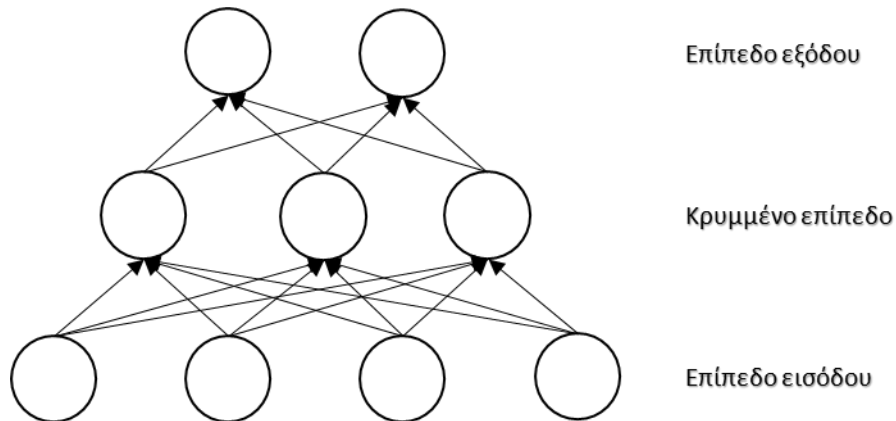
Ένα τεχνητό δίκτυο αποτελείται από τη σύμπραξη απλών μονάδων επεξεργασίας, οι οποίες επικοινωνούν στέλνοντας σήματα μεταξύ τους μέσω ενός μεγάλου αριθμού συνδέσεων με βάρη συμμετοχής. Τα νευρωνικά δίκτυα τα οποία θα αναλυθούν, βασίζονται σε παραλλαγές του Parallel Distributed Processing (PDP) και των αντίστοιχων μοντέλων. Τα παρακάτω αποτελούν τα βασικά στοιχεία των τελευταίων:

- Ένα σετ από μονάδες επεξεργασίας (νευρώνες, κελιά).
- Μια κατάσταση ενεργοποίησης  $y_k$  για κάθε μονάδα, η οποία ταυτίζεται με την έξοδο της μονάδας.
- Συνδέσεις μεταξύ των νευρώνων, αποτελούμενες από ένα βάρος  $w_{jk}$  το οποίο δείχνει την επιρροή του σήματος της μονάδας  $j$  προς την μονάδα  $k$ .
- Κανόνας διάδοσης, ο οποίος καθορίζει την ενεργή είσοδο  $s_k$  της μονάδας από τις εξωτερικές εισόδους.
- Συνάρτηση ενεργοποίησης  $F_k$ , η οποία καθορίζει το νέο επίπεδο ενεργοποίησης βασιζόμενη στην ενεργή είσοδο  $S_k(t)$  και την τρέχουσα ενεργοποίηση  $y_k(t)$ .
- Εξωτερική είσοδος  $\theta_k$  (προτίμηση, κατώφλι) για κάθε μονάδα.
- Μία μέθοδο για συλλογή της πληροφορίας (κανόνας εκμάθησης).
- Το περιβάλλον μέσα στο οποίο το σύστημα πρέπει να λειτουργήσει, παρέχοντας σήματα εισόδου και -αν είναι αναγκαίο- σήματα σφάλματος.[14]



Σχήμα 2.4 Βασικά στοιχεία νευρωνικού δικτύου[14]

Κάθε μονάδα επεξεργασίας πραγματοποιεί μια σχετικά απλή διεργασία: λαμβάνει κάποια είσοδο από τις γειτονικές μονάδες ή από εξωτερικές πηγές και τη χρησιμοποιεί για να υπολογίσει το εξερχόμενο σήμα το οποίο στη συνέχεια διανέμεται σε άλλες μονάδες. Εκτός αυτού, πραγματοποιεί και μία δεύτερη διεργασία, την προσαρμογή των βαρών. Η παραλληλία του συστήματος έγκειται στο γεγονός ότι πολλές μονάδες μπορούν να εκτελούν υπολογισμούς την ίδια στιγμή. Υπάρχουν τρεις τύποι επιπέδων που εμπεριέχουν μονάδες: το επίπεδο εισόδου (i), το οποίο λαμβάνει δεδομένα εκτός του δικτύου, το επίπεδο εξόδου (o), του οποίου τα στοιχεία στέλνουν δεδομένα εκτός του δικτύου και τα κρυμμένα επίπεδα (h), των οποίων οι εισόδους και εξόδους περιορίζονται στο εσωτερικό του δικτύου [22].



Σχήμα 2.5 Διάταξη επιπέδων και κόμβων τυχαίου νευρωνικού δικτύου

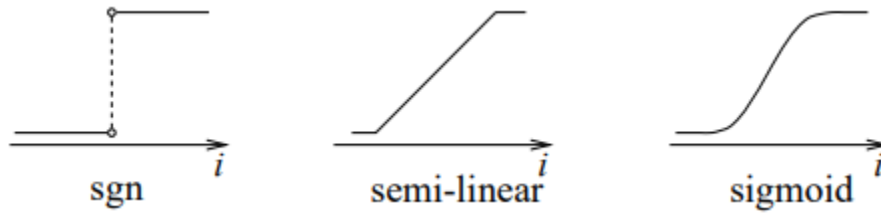
Σε ότι αφορά τις συνδέσεις των νευρώνων, στις περισσότερες περιπτώσεις, κάθε μονάδα συνεισφέρει προσθετικά στην είσοδο της μονάδας με την οποία συνδέεται. Η συνολική είσοδος στη μονάδα  $k$ , είναι το άθροισμα των εξόδων των επιμέρους μονάδων με τις οποίες είναι συνδεδεμένη, πολλαπλασιασμένες με ένα αντίστοιχο συντελεστή βάρους.

$$s_k(t) = \sum_j w_{jk}(t)y_j(t) + \theta_k(t) \quad (2.8)$$

Για την εφαρμογή της συνάρτησης  $F_k$ , η τελευταία παίρνει τη συνολική είσοδο  $s_k(t)$  και την τωρινή ενεργοποίηση  $y_k(t)$  και παράγει μια νέα τιμή της ενεργοποίησης της μονάδας  $k$ , έτσι ώστε:

$$y_k(t+1) = F_k(s_k(t)) \quad (2.9)$$

Σε γενικές γραμμές όμως, χρησιμοποιούνται οι εξής:



Σχήμα 2.6 Κοινές συναρτήσεις ενεργοποίησης [14]

Σε κάποιες περιπτώσεις, η έξοδος της μονάδας μπορεί να είναι μια στοχαστική συνάρτηση της συνολικής εισόδου της μονάδας [14].

Η εκμάθηση ενός τεχνητού νευρωνικού δικτύου ορίζεται ως η διαδικασία στην οποία οι παράμετροι του νευρωνικού δικτύου (βάρη και κατώφλια), οι οποίες αρχικά έχουν τυχαίες τιμές, αναβαθμίζονται συνεχώς και προσαρμόζονται σε μία συνεχή διαδικασία προσομοίωσης στο περιβάλλον του δικτύου. Το νευρωνικό δίκτυο τροποποιείται, ακολουθώντας συγκεκριμένα μοτίβα εκπαίδευσης, με στόχο να εκτελεί ικανοποιητικά κάποιο συγκεκριμένο σενάριο. Υπάρχουν τρεις τύποι εκμάθησης και είναι οι εξής: **εκμάθηση με επίβλεψη, εκμάθηση χωρίς επίβλεψη και ενισχυμένη εκμάθηση** [24].

Στον πρώτο τύπο εκμάθησης, υπάρχει ένας χειριστής-επιβλέπων, ο οποίος μπορεί να κρίνει εάν το σύστημα λειτουργεί στην επιθυμητή κατάσταση, ή να υποδείξει μια επιθυμητή απόκριση, να αξιολογήσει την απόκριση του συστήματος ή την ποσότητα σφάλματος. Το δίκτυο εκπαιδεύεται με την παροχή εισόδων και αντίστοιχων γνωστών επιθυμητών εξόδων. Τα ζευγάρια εισόδου-εξόδου τα παρέχει είτε ο επιβλέπων, είτε το σύστημα μέσα στο οποίο λειτουργεί το νευρωνικό δίκτυο (αυτο-επιβλεπούμενο).

Σε αντίθεση με τα προηγούμενα, κατά την εκμάθηση χωρίς επίβλεψη, δεν υπάρχει εκ των προτέρων σύνολο κατηγοριών στις οποίες πρόκειται να ταξινομηθούν τα παραγόμενα μοτίβα. Η μονάδα (έξοδος) έχει εκπαιδευτεί να ανταποκρίνεται σε συστάδες προτύπων των δεδομένων εισόδου, ανακαλύπτοντας στατιστικά εμφανή χαρακτηριστικά σε αυτά.

Σε ότι αφορά την τελευταία κατηγορία, θεωρείται ως ενδιάμεση μορφή των παραπάνω δύο τύπων μάθησης. Εδώ η μηχανή εκμάθησης κάνει κάποια ενέργεια στο περιβάλλον και λαμβάνει μια ανταπόκριση από αυτό. Το σύστημα εκμάθησης βαθμολογεί τη δράση του ως καλή (επιβράβευση) ή κακή (τιμωρείται) με βάση την απόκριση του περιβάλλοντος και ανάλογα προσαρμόζει τις παραμέτρους του. Γενικά, η προσαρμογή των παραμέτρων συνεχίζεται μέχρι την κατάσταση ισορροπίας, μετά από την οποία δε θα υπάρχουν άλλες αλλαγές σε αυτές [12].

Σε ένα νευρωνικό δίκτυο, ο τρόπος με τον οποίο συνδέονται οι μονάδες μεταξύ τους καθώς και η διάδοση των δεδομένων μέσα σε αυτό, το χωρίζουν σε 2 κατηγορίες:

- Δίκτυα με τροφοδότηση προς τα εμπρός (feed-forward networks), στο οποίο τα δεδομένα από τις μονάδες εισόδου ρέουν προς τις μονάδες εξόδου αυστηρώς προς τα εμπρός. Η επεξεργασία δεδομένων μπορεί να επεκταθεί σε πολλαπλά επίπεδα των μονάδων, αλλά δεν υπάρχουν συνδέσεις ανατροφοδότησης, δηλαδή συνδέσεις που να εκτείνονται από την έξοδο των μονάδων μέχρι την είσοδο τους, στο ίδιο ή σε προηγούμενο επίπεδο [14].
- Επαναλαμβανόμενα δίκτυα (recurrent networks), στα οποία οι έξοδοι κάποιων νευρώνων τροφοδοτούνται πίσω στους ίδιους νευρώνες ή σε νευρώνες προηγούμενων επιπέδων. Αυτό επιτρέπει τη ροή πληροφοριών τόσο προς τα εμπρός όσο και προς τα πίσω, προσδίδοντας έτσι δυναμική μνήμη στο σύστημα [13].

Ιδιαίτερη αναφορά πρέπει να γίνει στον αλγόριθμο οπισθοδιάδοσης, που χρησιμοποιείται ευρέως σε τεχνητά νευρωνικά δίκτυα με τροφοδότηση προς τα εμπρός, τα οποία διαθέτουν τουλάχιστον μία είσοδο, μία έξοδο και ένα κρυμμένο επίπεδο. Καθώς η πληροφορία ρέει από την είσοδο προς την έξοδο του δικτύου, το σφάλμα – διαφορά επιθυμητής και παρούσας τιμής- το οποίο προκύπτει στην έξοδο, τροφοδοτείται ξανά πίσω στο δίκτυο. Με αυτή τη διαδικασία επιτυγχάνεται η σταδιακή μείωση του σφάλματος, αλλάζοντας τα τυχαία-στην αρχή- βάρη συμμετοχής, μέχρι το δίκτυο να μάθει τα δεδομένα εκπαίδευσης. [12] Τα δίκτυα αυτά είναι ευέλικτα και χρησιμοποιούνται για πληθώρα εφαρμογών, όπως η μοντελοποίηση δεδομένων, η κατηγοριοποίηση, η πρόβλεψη, ο έλεγχος, η συμπίεση δεδομένων και εικόνων και η αναγνώριση μοτίβων [13]. Λόγω των πλεονεκτημάτων αυτών, επιλέγονται στην παρούσα εργασία δίκτυα με τροφοδότηση προς τα εμπρός, τα οποία χρησιμοποιούν αλγόριθμο οπισθοδιάδοσης.

## 2.5 ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ ΣΤΟΝ ΕΛΕΓΧΟ

Ο έλεγχος με τεχνικές νευρωνικών δικτύων και παράλληλης διάδοσης της πληροφορίας, βρίσκει, λόγω των πλεονεκτημάτων που έχουν ήδη αναλυθεί, μεγάλη απήχηση σε ακαδημαϊκό και μη επίπεδο, τόσο στην πλοήγηση οχημάτων όσο και σε ευρύτερες εφαρμογές, όπου η αβεβαιότητα του περιβάλλοντος, τα μη γραμμικά μοτίβα

αλλά και η αναγκαιότητα μείωσης του ανθρώπινου σφάλματος, χρίζουν αναγκαία την εφαρμογή νευρωνικών τεχνικών ελεγκτών.

### 2.5.1 ΓΕΝΙΚΕΣ ΕΦΑΡΜΟΓΕΣ

Τα υποθαλάσσια ρομποτικά οχήματα χρησιμοποιούνται τόσο στην εξερεύνηση, όσο και στη βιομηχανία, ανταποκρινόμενα σε ένα περιβάλλον υψηλά δυναμικό, με εναλλασσόμενες καταστάσεις και γενικά σε καθήκοντα τα οποία οδηγούν σε αλλαγή της συμπεριφοράς τους, με αποτέλεσμα τη διαταραχή του σήματος αντίχρευσσης της θέσης τους. Τα οχήματα τα οποία λειτουργούν με ελεγκτή PID, σε αυτές τις συνθήκες, χρειάζονται εκ νέου ρύθμιση των βασικών παραμέτρων του ελεγκτή, γεγονός το οποίο οδηγεί σε χρονική καθυστέρηση. Για την αντιμετώπιση των προβλημάτων αυτών, οι Rodrigo Hernández-Alvarado *et al.* [15], πρότειναν τη δημιουργία ενός αυτορρυθμιζόμενου PID ελεγκτή, ο οποίος θα βασίζεται σε νευρωνικά δίκτυα. Συγκεκριμένα, το νευρωνικό δίκτυο εκτιμά τις τιμές των τριών βασικών τιμών του PID ελεγκτή ( $K_p, K_i, K_d$ ), ώστε το σύστημα να σταθεροποιείται, έχοντας το μικρότερο δυνατό σφάλμα σήματος θέσης. Οι Nicolas Durand *et al.* [16], δούλεψαν πάνω στην αποφυγή σύγκρουσης μεταξύ αεροσκαφών στον εναέριο χώρο. Μελέτησαν την ύπαρξη ελεγκτών στα αεροσκάφη, τα οποία, σε περίπτωση πιθανής σύγκρουσης, οδηγούν τα αεροπλάνα σε αλλαγή της πορείας τους. Έτσι, δημιούργησαν ελεγκτή, δομημένο σε ένα πολυεπίπεδο νευρωνικό δίκτυο. Στα δεδομένα τα οποία λαμβάνονται από το ραντάρ του αεροπλάνου, βασίζονται οι είσοδοι του δικτύου, οι οποίες είναι απαραίτητες γωνίες και αποστάσεις στο δισδιάστατο καρτεσιανό επίπεδο, ενώ η έξοδος είναι η τιμή της γωνίας εκτροπής της κατεύθυνσης του αεροπλάνου. Λόγω της φύσης του προβλήματος, επιλέχθηκε μάθηση χωρίς επίβλεψη, με τη χρήση γενετικού αλγορίθμου και έτσι, το σύστημα εκπαιδεύτηκε σε διάφορα σενάρια μεταξύ δύο αεροσκαφών, χωρίς να γνωρίζει την ιδανική λύση. Στο στάδιο της αξιολόγησης, ο ελεγκτής, συγκρίθηκε με την ιδανική λύση των παραπάνω σεναρίων και τα αποτελέσματα υπήρξαν ενθαρρυντικά, αφού δεν υπήρξε μεγάλη απόκλιση με τα δεδομένα αξιολόγησης.

### 2.5.2 ΕΦΑΡΜΟΓΕΣ ΣΤΗΝ ΕΥΡΥΤΕΡΗ ΡΟΜΠΟΤΙΚΗ

Οι Daryl H. Graf *et al.* [17], δόμησαν και προσομοίωσαν έναν ελεγκτή βασισμένο σε νευρωνικά δίκτυα, που μπορεί να ενταχθεί και να προσαρμοστεί σε ρομποτικούς βραχίονες γενικής φύσεως. Η καινοτομία αυτού του ελεγκτή έγκειται στο ότι μαθαίνει αυτόνομα, τόσο τους περιορισμούς της κινηματικής του ρομπότ και του χώρου, όσο και την οριοθέτηση μεταξύ των οπτικών αισθητήρων και του βραχίονα, γεγονός που του δίνει τη δυνατότητα να προσεγγίζει ένα σημείο στο χώρο, στοχεύοντας το οπτικά, μέσω των αισθητήρων. Εκτός αυτών, αντιλαμβάνεται τους περιορισμούς που θέτουν πιθανά εμπόδια στο χώρο και είναι ευέλικτος σε αλλαγές της κινηματικής ή σε ταλαντώσεις των αρθρώσεων.

### 2.5.3 ΕΦΑΡΜΟΓΕΣ ΣΤΟΝ ΕΛΕΓΧΟ ΑΥΤΟΝΟΜΩΝ ΟΧΗΜΑΤΩΝ

Στη [18] προτάθηκε ένας αλγόριθμος για την εύρεση μονοπατιού προς το στόχο, ενός αυτόνομου ρομποτικού οχήματος. Συγκεκριμένα, ο ελεγκτής αποτελείται από δύο νευρωνικά δίκτυα, τα οποία συμβάλλουν στην επίλυση δύο διαφορετικών κομματιών του προβλήματος, τη χαρτογράφηση του περιβάλλοντος στο οποίο κινείται το ρομπότ και την επιλογή της γωνίας κατεύθυνσης, ώστε να φτάσει το όχημα στο στόχο. Το πρώτο νευρωνικό, λαμβάνει τα δεδομένα απόστασης από τα εμπόδια, από αισθητήρες υπερήχου, ως είσοδο και επιστρέφει ως έξοδο, το ελεύθερο από εμπόδια κομμάτι του χώρου εκείνη τη στιγμή. Η έξοδος αυτή, μαζί με το στόχο, τον οποίο πρέπει να φτάσει το ρομπότ, εισάγονται στο δεύτερο νευρωνικό, το οποίο δίνει τη γωνία στροφής του ρομπότ, για τη μεταφορά του στο στόχο με ταυτόχρονη αποφυγή των εμποδίων. Ο ελεγκτής δοκιμάστηκε σε περιβάλλον προσομοίωσης, για συγκεκριμένο εργαστηριακό ρομπότ, στο περιβάλλον-αρένα, όπου είχαν εκπαιδευτεί τα νευρωνικά δίκτυα, όσο και σε άγνωστα για τον ελεγκτή περιβάλλοντα, με πολύ θετικά αποτελέσματα και θέτοντας τη βάση για χρήση του ελεγκτή σε πραγματικό περιβάλλον.

Οι M.K Singha *et al.* [19], μελέτησαν και αυτοί την εύρεση και βελτιστοποίηση του μονοπατιού ενός ρομποτικού οχήματος, προς ένα σημείο-στόχο. Ο ελεγκτής τον οποίο ανέπτυξαν, δομείται από ένα νευρωνικό δίκτυο με τέσσερις εισόδους - απόσταση από εμπόδια σε τρεις κατευθύνσεις και γωνία του ρομπότ από το στόχο - και έξοδο τη γωνία εκτροπής του ρομπότ. Τα δεδομένα εκπαίδευσης λήφθηκαν από κίνηση του ρομπότ στο χώρο, με τη βοήθεια κινηματικών εξισώσεων. Ο ελεγκτής, αφού εκπαιδεύτηκε, δοκιμάστηκε σε διάφορα περιβάλλοντα προσομοίωσης, τόσο με στατικά όσο και με δυναμικά εμπόδια, για την προσέγγιση αντιμετώπισης καταστάσεων στον πραγματικό κόσμο. Στο τελευταίο στάδιο, ο ελεγκτής αντιμετώπισε την οδήγηση του ρομπότ σε εργαστηριακό περιβάλλον, αντίστοιχο της προσομοίωσης. Ο ελεγκτής αποδείχτηκε αποτελεσματικός, αποφεύγοντας τα εμπόδια και φτάνοντας στο στόχο.

## ΚΕΦΑΛΑΙΟ 3

# ΒΑΣΙΚΕΣ ΑΡΧΕΣ ΚΑΙ ΑΛΓΟΡΙΘΜΟΙ ΚΙΝΗΣΗΣ-ΕΛΕΓΧΟΥ

### 3.1 ΕΙΣΑΓΩΓΗ

Στο κεφάλαιο αυτό, αναλύεται το κινηματικό μοντέλο των οχημάτων διαφορικής κίνησης, παρουσιάζονται τα βασικά μέρη του ρομποτικού οχήματος που χρησιμοποιείται στην παρούσα εργασία και γίνεται περιγραφή του τρόπου σύνδεσης του προσομοιωτή με το λογισμικό Matlab και του βασικού αλγορίθμου που αναπτύχθηκε για την κίνηση του ρομποτικού οχήματος στο χώρο.

### 3.2 ΘΕΣΗ ΚΑΙ ΜΟΝΤΕΛΟ ΚΙΝΗΣΗΣ ΟΧΗΜΑΤΟΣ ΔΙΑΦΟΡΙΚΗΣ ΚΙΝΗΣΗΣ

Για να είναι γνωστή η θέση και συνεπώς η κίνηση ενός έντροχου οχήματος σε ένα δισδιάστατο επίπεδο, είναι απαραίτητος ο ορισμός δύο πλαισίων, του γενικού πλαισίου αναφοράς  $\{X_I, Y_I\}$  και του επιπέδου αναφοράς του ρομπότ  $\{X_R, Y_R\}$ .

Με τη βοήθεια του σημείου P, το οποίο βρίσκεται πάνω στο νοητό άξονα κίνησης του οχήματος, μπορεί να διατυπωθεί η θέση του ρομπότ ως προς το γενικό πλαίσιο αναφοράς, στην περίπτωση αυτή, το καρτεσιανό επίπεδο (Σχήμα 3.1), ως εξής:

$$e_i = P[x \ y \ \theta] \quad (3.1)$$

όπου x και y η θέση του ρομπότ στο καρτεσιανό επίπεδο και  $\theta$  ο προσανατολισμός του, δηλαδή η γωνιακή διαφορά των δύο πλαισίων αναφοράς.

Τα δύο επίπεδα συσχετίζονται μέσω του πίνακα περιστροφής:

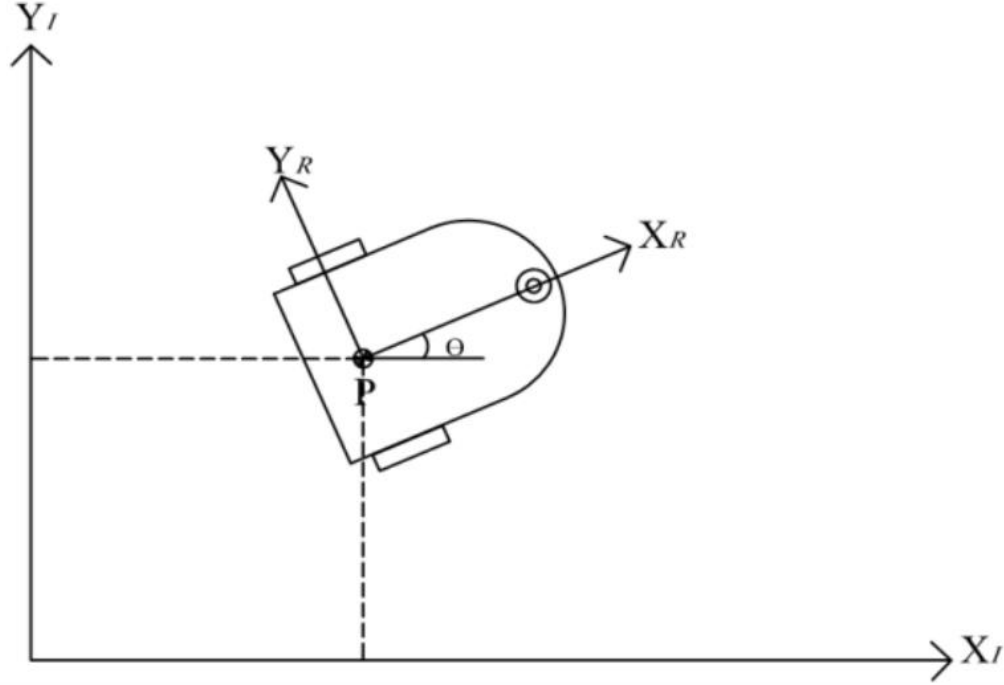
$$R(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

Από τις (3.1) και (3.2), οι εξισώσεις θέσεις και ταχύτητας του οχήματος δίνονται ως εξής:



$$e_R = R(\theta) * e_i \quad (3.3)$$

$$\dot{e}_R = R(\theta) * \dot{e}_i \quad (3.4)$$



Σχήμα 3.1 Θέση-πλαίσια αναφοράς οχήματος σε δισδιάστατο καρτεσιανό επίπεδο[20]

Το ρομποτικό όχημα το οποίο χρησιμοποιείται στην παρούσα εργασία, εκτελεί διαφορική κίνηση, έχοντας δύο ανεξάρτητης κίνησης τροχούς και ένα τροχό τύπου caster, ο οποίος προσφέρει ισορροπία στο σύστημα και στρίβει ως προς κατακόρυφο άξονα, ανάλογα με τη φορά κίνησης του οχήματος. Σε αντίθεση με άλλες μορφές κίνησης, όπου μπορεί π.χ να υπάρχει στροφή του τροχού ως προς άξονα περιστροφής, στη διαφορική οδήγηση, η επιθυμητή κατεύθυνση δίνεται από το συνδυασμό της γωνιακής ταχύτητας του αριστερού και δεξιού τροχού. Αν το όχημα πρέπει να μετακινηθεί ευθεία εμπρός, τότε οι τροχοί έχουν ταχύτητα ίδιου μέτρου, αν πάει προς τα δεξιά, ο αριστερός τροχός έχει μεγαλύτερο μέτρο ταχύτητας και αντίστροφα.

Για τη μετακίνηση του ρομποτικού οχήματος σε ένα χώρο, είναι απαραίτητο να γνωρίζει ο χειριστής, ανά πάσα στιγμή που βρίσκεται, δηλαδή τη θέση και τον προσανατολισμό του. Θεωρείται αρχικά η διάταξη  $q_k = [x_k \ y_k \ u_k]$  και  $v_k$  και  $\omega_k$  η γραμμική και γωνιακή ταχύτητα τη χρονική στιγμή  $t_k$ . Ολοκληρώνοντας κατά Euler:

$$x_{k+1} = x_k + v_k T_s \cos \theta_k \quad (3.5)$$

$$y_{k+1} = y_k + v_k T_s \sin \theta_k \quad (3.6)$$

$$\theta_k = \theta_k + \omega_k T_s \quad (3.7)$$

$$\text{όπου } v_k T_s = \Delta_s \text{ και } \omega_k T_s = \Delta_\theta \quad (3.8)$$

$$\text{με } T_s = T_{k+1} - T_k \quad (3.9)$$

Αν  $\Delta\varphi_R$  και  $\Delta\varphi_L$  ο αριθμός των στροφών που έχει εκτελέσει αντίστοιχα ο δεξιός και αριστερός τροχός του οχήματος κατά τη χρονική περίοδο  $T_s$ , η γραμμική και γωνιακή μετατόπιση του ρομπότ δίνεται ως εξής:

$$\Delta_s = \frac{r}{2}(\Delta\varphi_R + \Delta\varphi_L), \quad \Delta_\theta = \frac{r}{d}(\Delta\varphi_R - \Delta\varphi_L) \quad (3.10)$$

Όπου  $r$  η ακτίνα του τροχού και  $d$  η αξονική απόσταση των δύο τροχών.

Λύνοντας ως προς  $v_k$  και  $\omega_k$  από σχέσεις 3.8, 3.9, εξάγεται το κινηματικό μοντέλο του οχήματος.

Η θέση ενός οχήματος διαφορικής κίνησης τη χρονική στιγμή  $t_k$  μπορεί να γίνει γνωστή, αν ξεκινώντας από γνωστό σημείο, αθροιστεί η αυξανόμενη διανυόμενη απόσταση. Έτσι, η θέση και ο προσανατολισμός ενός οχήματος στο χώρο χαρακτηρίζεται από την εξής σχέση [20]:

$$\begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \theta_{k-1} \end{bmatrix} + \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta_s \\ \Delta_\theta \end{bmatrix} \quad (3.11)$$

### 3.3 ΡΟΜΠΟΤΙΚΟ ΟΧΗΜΑ PIONEER 3 – DX

Για το περιβάλλον προσομοίωσης, χρησιμοποιείται το μοντέλο CAD του ρομποτικού οχήματος Pioneer 3 - DX, με όλες τις δυναμικές λειτουργίες του (Σχήμα 3.2).



Σχήμα 3.2 Ρομποτικό όχημα Pioneer 3-dx [21]

Πρόκειται για όχημα βασισμένο στην ανεξάρτητη διαφορική κίνηση δύο τροχών, δύο κινητήρων και ενός τροχού-οδηγού, τύπου caster, ενώ διαθέτει 16 αισθητήρες υπερήχου (sonar) για την ανίχνευση εμποδίων. Φέρει τη δυνατότητα επεκτάσεων, όπως αισθητήρων laser scanner, κάμερας και ρομποτικού βραχίονα. Είναι ιδανικό για χρήση στην εκπαίδευση και την έρευνα. Στον Πίνακα 3.1 παρουσιάζονται τα βασικά λειτουργικά του χαρακτηριστικά.

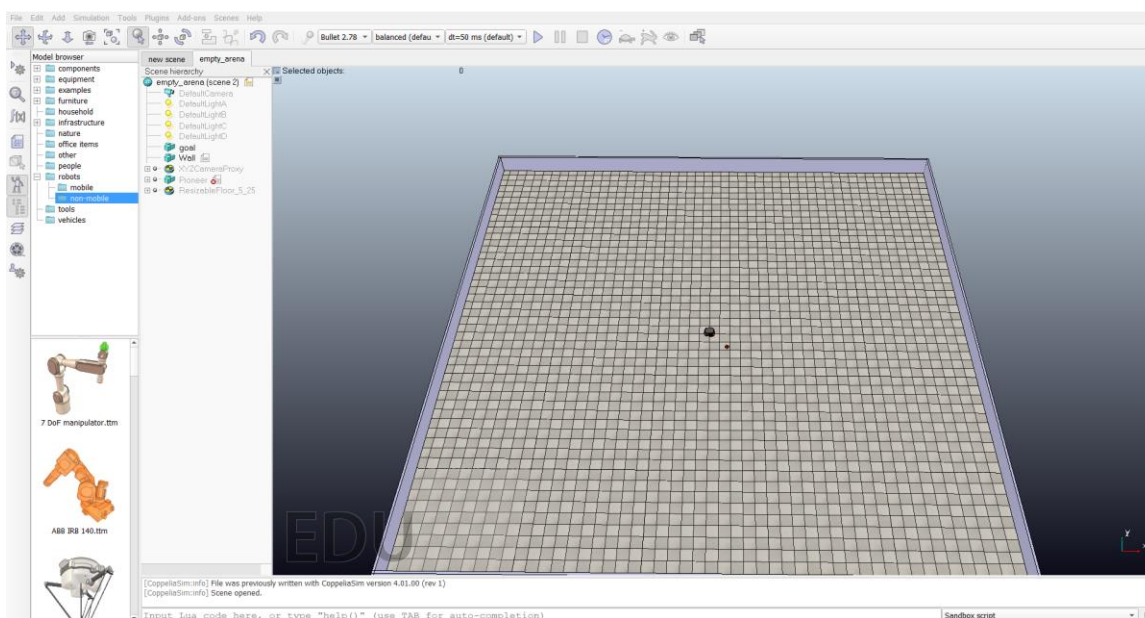
<p><b>Construction</b></p> <p>Body: 1.6 mm aluminum (powder-coated)</p> <p>Tires: Foam-filled rubber</p>
<p><b>Operation</b></p> <p>Robot Weight: 9 kg</p> <p>Operating Payload: 17 kg</p>
<p><b>Differential Drive Movement</b></p> <p>Turn Radius: 0 cm</p> <p>Swing Radius: 26.7 cm</p> <p>Max. Forward/Backward Speed: 1.2 m/s</p> <p>Rotation Speed: 300°/s</p> <p>Max. Traversable Step: 2.5 cm</p> <p>Max. Traversable Gap: 5 cm</p> <p>Max. Traversable Grade: 25%</p> <p>Traversable Terrain: Indoor, wheelchair accessible</p>
<p><b>Power</b></p> <p>Run Time: 8-10 hours w/3 batteries (with no accessories)</p> <p>Charge Time: 12 hours (standard) or 2.4 hrs (optional high-capacity charger)</p> <p>Available Power Supplies:</p> <p>5 V @ 1.5 A switched</p> <p>12 V @ 2.5 A switched</p>
<p><b>Microcontroller I/O</b></p> <p>System Serial</p> <p>32 digital inputs, 8 digital outputs,</p> <p>7 analog inputs, 3 serial expansion ports</p>
<p><b>User Control Panel</b></p> <p>MIDI programmable piezo buzzer</p> <p>Main power indicator</p> <p>Battery charge indicator</p> <p>AUX power switches</p> <p>System reset Motor enable pushbutton</p>

Πίνακας 3.1 Δομικά στοιχεία οχήματος Pioneer 3-dx

### 3.4 ΠΕΡΙΒΑΛΛΟΝ ΠΡΟΣΟΜΟΙΩΤΗ CORPELIA SIMULATOR

Το λογισμικό Coppelia Simulator της Coppelia Robotics, αποτελεί συνέχεια του λογισμικού προσομοίωσης V-REP και είναι γνωστό για τη δυνατότητα εφαρμογής του, τόσο σε τομείς της εκπαίδευσης και της έρευνας, όσο και σε εργοστασιακές διαδικασίες. Βασίζεται σε αρχιτεκτονική κατακεντρωμένη ελέγχου, αφού κάθε αντικείμενο/μοντέλο, μπορεί να ελεγχθεί ξεχωριστά με τη χρήση κώδικα, έχοντας ποικίλες επιλογές τις οποίες διαθέτει το λογισμικό, από ενσωματωμένο κώδικα (embedded script) μέχρι απομακρυσμένο έλεγχο (remote API), με τη βοήθεια δηλαδή άλλου λογισμικού. Οι διάφορες δυνατότητες ελέγχου κάνουν τον προσομοιωτή ευέλικτο και ικανό να έχει εφαρμογή σε διαφορετικά προβλήματα. Οι γλώσσες προγραμματισμού που υποστηρίζονται είναι οι: C/C++, Python, Java, Lua, Matlab ή Octave.

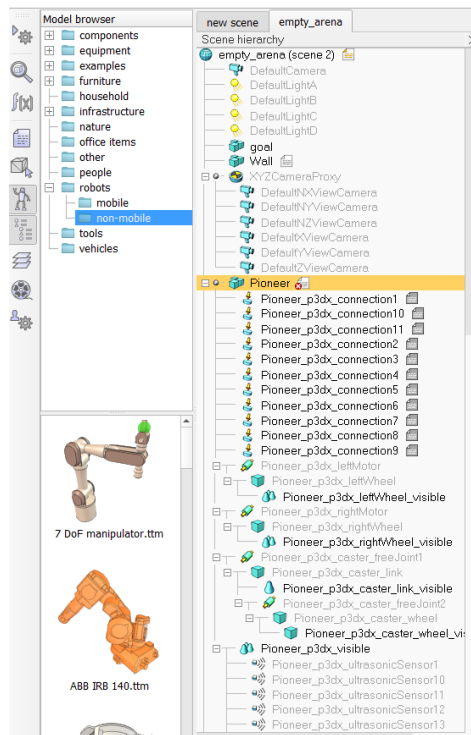
Στον προσομοιωτή είναι δυνατό να χρησιμοποιηθούν αρένες, τοιχώματα, αντικείμενα, τα οποία είναι στατικά ή δυναμικά, όπως εμπόδια, μπάλες κλπ, να χρησιμοποιηθούν ρομπότ, τόσο έντροχα όσο και βραχίονες, τα οποία μπορούν να φέρουν διαφόρων ειδών αισθητήρες, αλλά και κάμερες. Στο σχήμα 3.3 απεικονίζεται το γραφικό περιβάλλον του προσομοιωτή.



Σχήμα 3.3 Ενδεικτικό περιβάλλον προσομοίωσης

Στο πάνω μέρος βρίσκονται οι μπάρες εργαλείων, με τη βοήθεια των οποίων ο χρήστης αλληλοεπιδρά με το περιβάλλον και εκτελεί σημαντικές λειτουργίες, όπως το να ανοίξει ένα αρχείο ή να το σώσει, να προσθέσει αντικείμενα στην προσομοίωση (Add), να αλλάξει την οπτική γωνία που βλέπει το χώρο, να διαβάσει και να δώσει συντεταγμένες θέσης και προσανατολισμού σε κάποιο αντικείμενο, να αρχίσει, να κάνει παύση ή να σταματήσει την προσομοίωση κ.ά.

Τα αντικείμενα που υπάρχουν ενσωματωμένα στο πρόγραμμα και είναι διαθέσιμα για άμεση χρήση, βρίσκονται στην αριστερή στήλη. Ενδεικτικές επιλογές αποτελούν τα εξής: ρομπότ -κινητά και μη-, εμπόδια σε διάφορα σχήματα, ποικίλες μορφοποιήσεις του εδάφους κ.ά. Επιπλέον, στα αριστερά βρίσκεται το ιεραρχικό δέντρο του περιβάλλοντος, το οποίο εμφανίζει τα αντικείμενα που χρησιμοποιούνται εκείνη τη χρονική στιγμή που είναι ανοιχτό το περιβάλλον προσομοίωσης. Στην περίπτωση μίας αρένας, όπως αυτής του σχήματος 3.3, εμφανίζεται η κάμερα του χρήστη, ο στόχος, τα τοιχώματα, το δάπεδο, ο φωτισμός και το ρομποτικό όχημα. Τα βασικά μέρη του δέντρου αποτελούνται από παρακλάδια, όπως φαίνεται αναλυτικά στο σχήμα 3.4.



Σχήμα 3.4 Αναλυτική απεικόνιση ιεραρχικού δέντρου προσομοίωσης

Το ρομποτικό όχημα, αποτελείται από διαφορετικά μικρότερα αντικείμενα. Ενδεικτικά, απαρτίζεται ιεραρχικά από τους κινητήρες και τους αντίστοιχους τροχούς, τους αισθητήρες και τις συνδέσεις στη μητρική πλακέτα του οχήματος. Όπως έχει αναφερθεί, κάθε μικρότερο μέρος του δέντρου μπορεί να διαθέτει, εφόσον είναι επιθυμητό και χρήσιμο, το δικό του ενσωματωμένο κώδικα (script) σε γλώσσα Lua, όπως για παράδειγμα φαίνεται στο σχήμα 3.4 ότι φέρουν οι συνδέσεις της πλακέτας.

### 3.5 ΣΥΝΔΕΣΗ ΠΡΟΣΟΜΟΙΩΤΗ COPPELIA SIMULATOR ΜΕ ΤΟ ΛΟΓΙΣΜΙΚΟ MATLAB

Για τις ανάγκες τις εργασίας, έγινε χρήση του απομακρυσμένου ελέγχου, δηλαδή το περιβάλλον προσομοίωσης λάμβανε εντολές, εξωτερικά, από το λογισμικό της Matlab. Για να συμβεί αυτό, αναπτύχθηκε αντίστοιχος κώδικας στο περιβάλλον της Matlab, ο οποίος απεικονίζεται αναλυτικά στο σχήμα 3.5 και εξηγείται παρακάτω.

```

8  function [simulation_steps, vp2goal, Pioneer] = empty_arena(NET) %FIS
9  % Connection Matlab/Vrep
10 disp('Program started')
11 %build the object and load the library
12 %initialiseRoboticsToolbox
13 %We use the loopback
14 %IP address / connection Port / wait Until Connected /
15 %do Not Reconnect Once Disconnected / time Out In (ms)
16
17 sim = remApi('remoteApi');
18
19
20 sim.simxFinish(-1);
21 clientID=sim.simxStart('127.0.0.1',19999,true,true,5000,5);
22
23 sim.simxStartSimulation(clientID,sim.simx_opmode_oneshot_wait);
24
25 if clientID == -1
26     error('the connection to the server was not possible');
27 else
28     disp('Connected to remote API server');
29
30     %Requests a start of a simulation (clientID / operationMode)
31 [errorCode]=sim.simxStartSimulation(clientID,sim.simx_opmode_oneshot_wait);
32 if errorCode ~=0
33     error('Vrep error of requests a start of a simulation');
34 else
35     disp('OK request simulation');
36 end
37 end

```

Σχήμα 3.5 Απομακρυσμένη σύνδεση Matlab – Coppelia Simulator

Πρώτο βήμα για τον έλεγχο με Matlab και τη σύνδεση με τον προσομοιωτή, είναι να εισαχθούν τα αρχεία με τις κατάλληλες βιβλιοθήκες στο αντίστοιχο μονοπάτι (path) της μνήμης, στο οποίο βρίσκεται ο κώδικας (script). Έπειτα, με τη συνάρτηση `remApi('remoteApi')`, η βιβλιοθήκη φορτώνεται στη Matlab και περνιέται στη μεταβλητή `sim`. Η μεταβλητή `clientID` κρατάει την τιμή της διεύθυνσης της σύνδεσης. Στο τέλος, κλείνεται οποιαδήποτε ανοιχτή σύνδεση υπήρχε από πριν και δίνεται εντολή (`simxStart`), για να ενεργοποιηθεί ο απομακρυσμένος έλεγχος μέσω των βιβλιοθηκών και να ξεκινήσει η σύνδεση CoppeliaSim-Matlab. Ελέγχεται το αποτέλεσμα της σύνδεσης και ξεκινάει η προσομοίωση. Σημειώνεται ότι, για οποιοδήποτε προσομοίωση, το περιβάλλον του

coppelia simulator πρέπει να είναι ανοιχτό στην επιθυμητή αρένα και στην κατάσταση «play».

### 3.6 ΑΛΓΟΡΙΘΜΟΣ/ΚΩΔΙΚΑΣ ΔΙΑΧΕΙΡΗΣΗΣ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΠΡΟΣΟΜΟΙΩΣΗΣ

Στην ενότητα αυτή, γίνεται περιγραφή και ανάλυση του αλγορίθμου αυτού, με τη βοήθεια του οποίου καθορίζονται όλες οι αναγκαίες μεταβλητές του περιβάλλοντος προσομοίωσης, ώστε να είναι έτοιμο το όχημα να κινηθεί και να υλοποιήσει ένα σενάριο.

Πρώτο και βασικό βήμα αποτελεί η αρχικοποίηση υπό μορφή μεταβλητών, κάποιων κύριων αντικειμένων/παραμέτρων του περιβάλλοντος προσομοίωσης, δηλαδή ολόκληρου του ρομποτικού οχήματος, αλλά και των επιμέρους χρήσιμων κομματιών του, δηλαδή των δύο κινητήρων, του προορισμού-στόχου και των 16 υπερηχητικών αισθητήρων του οχήματος. Οι μεταβλητές αυτές είναι απαραίτητες για την επίτευξη πειραμάτων στη συνέχεια, αφού με τη με τη χρήση αυτών, καθορίζεται η θέση του ρομπότ στο χώρο, ελέγχεται η ταχύτητα του, αλλάζει το σημείο – στόχος, λαμβάνονται δεδομένα από τους αισθητήρες κ.ά.

Στο σχήμα 3.6 απεικονίζεται ενδεικτικά ένα κομμάτι του κώδικα στο περιβάλλον της Matlab, με τη βοήθεια του οποίου έγιναν όλες οι απαραίτητες αρχικοποιήσεις. Στο σχήμα δίνεται η εκχώρηση, σε μεταβλητές, του οχήματος, των δύο κινητήρων, καθώς και του αντικειμένου - στόχου.

Με τη βοήθεια της συνάρτησης `simxGetObjectHandle`, γίνεται η σύνδεση των αντικειμένων του προσομοιωτή με τη Matlab. Η συνάρτηση δέχεται σαν όρισμα τη διεύθυνση της σύνδεσης με τη Matlab, η οποία υπάρχει ήδη από τον κώδικα της ενότητας 3.5, το όνομα του αντικειμένου για το οποίο απαιτείται χειρισμός, ακριβώς όπως αναγράφεται στο ιεραρχικό δέντρο του προσομοιωτή, και το είδος της συνάρτησης που καθορίζει τον τρόπο μεταφοράς των δεδομένων μεταξύ Matlab και Coppelia Simulator. Ως έξοδο, η συνάρτηση επιστρέφει έναν αριθμό σφάλματος και τη μεταβλητή χειρισμού του αντικειμένου που επιλέχτηκε σε κάθε περίπτωση. Για κάθε αντικείμενο, ελέγχεται ο αριθμός σφάλματος εάν είναι διάφορος του μηδέν. Εάν ναι, τότε σημαίνει ότι έχει υπάρξει κάποιο σφάλμα στη σύνδεση μεταξύ των προγραμμάτων και τυπώνεται το αντίστοιχο μήνυμα σφάλματος. Εάν όχι, συνεχίζεται η ροή του κώδικα κανονικά.

```

6   %Pioneer
7   [errorCode, h.Pioneer] = sim.simxGetObjectHandle(clientID,'Pioneer',...
8       sim.simx_opmode_oneshot_wait);
9   if errorCode ~= 0
10      error('Error simxGetObjectHandle function for Pionner object');
11  end
12
13  %Left Motor
14  [errorCode, h.leftMotor] = sim.simxGetObjectHandle(clientID,...
15      'Pioneer_p3dx_leftMotor', sim.simx_opmode_oneshot_wait);
16  if errorCode ~= 0
17      error('Error simxGetObjectHandle function with left motor Pionner object');
18  end
19
20  %Right Motor
21  [errorCode, h.rightMotor] = sim.simxGetObjectHandle(clientID,...
22      'Pioneer_p3dx_rightMotor', sim.simx_opmode_oneshot_wait);
23  if errorCode ~= 0
24      error('Error simxGetObjectHandle function with right motor Pionner object');
25  end
26
27  %Goal
28  [errorCode, h.goal] = sim.simxGetObjectHandle(clientID,'goal',...
29      sim.simx_opmode_oneshot_wait);
30  if errorCode ~= 0
31      error('Error simxGetObjectHandle function for goal object');
32  end

```

Σχήμα 3.6 Αρχικοποίηση βασικών μεταβλητών

Ο τρόπος λειτουργίας της συνάρτησης `simxGetObjectHandle` βρίσκεται στο εγχειρίδιο του site της Coppelia Robotics, μαζί με όλες τις υπόλοιπες συναρτήσεις και με παραλλαγές για όλες τις διαθέσιμες γλώσσες προγραμματισμού. Στο σχήμα 3.7, παρατίθεται απόσπασμα του εγχειριδίου για τη συνάρτηση αυτή.

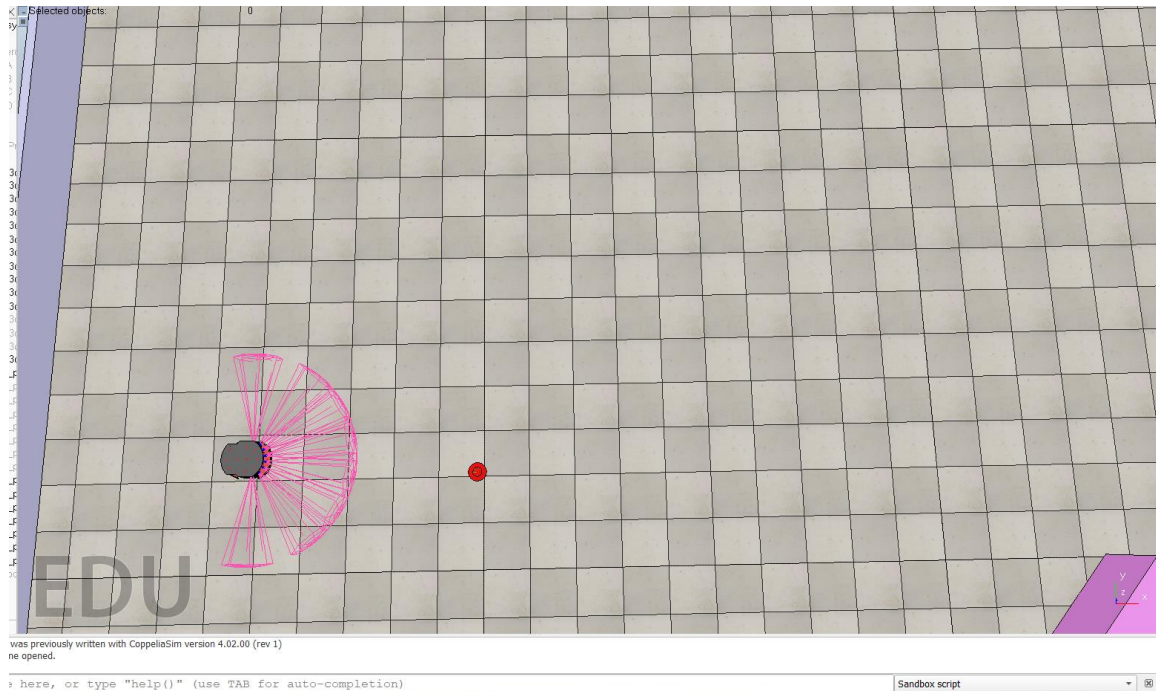
**`simxGetObjectHandle` (regular API equivalent: `sim.getObjectHandle`)**

Description	Retrieves an object handle based on its name.
Matlab synopsis	<code>[number returnCode,number handle]=simxGetObjectHandle(number clientID,string objectName,number operationMode)</code>
Matlab parameters	<b>clientID</b> : the client ID. refer to <a href="#">simxStart</a> . <b>objectName</b> : name of the object. <b>operationMode</b> : a <a href="#">remote API function operation mode</a> . Recommended operation mode for this function is <code>simx_opmode_blocking</code>
Matlab return values	<b>returnCode</b> : a <a href="#">remote API function return code</a> <b>handle</b> : the handle
Other languages	C/C++, Python, Java, Octave, Lua

Σχήμα 3.7 Εγχειρίδιο χρήσης συνάρτησης `simxGetObjectHandle`

Στο σχήμα 3.8, απεικονίζονται σε γραφικό περιβάλλον, οι βασικές μεταβλητές οι οποίες αρχικοποιήθηκαν, με ενεργοποιημένους αισθητήρες.





Σχήμα 3.8 Ρομποτικό όχημα, αισθητήρες και στόχος, στο περιβάλλον προσομοίωσης.

Στη συνέχεια του κώδικα, είναι απαραίτητο να ληφθεί η θέση (x,y,z) του αντικειμένου - στόχου και του οχήματος, όπως αρχικά έχουν διαταχθεί στο χώρο, καθώς και η γωνίες euler του οχήματος ( $\alpha$ ,  $\beta$ ,  $\gamma$ ). Στο σχήμα 3.9, παρουσιάζεται, ο κώδικας για την εισαγωγή σε μεταβλητή, ενός αντικειμένου στο χώρο, ενδεικτικά του αντικειμένου – στόχου.

```

44  %% Retrieves goal position
45  [errorCode, goal_pos] = sim.simxGetObjectPosition(clientID,h.goal,-1,sim.simx_opmode_streaming);%first call: streaming
46  pause(0.1)
47  [errorCode, goal_pos] = sim.simxGetObjectPosition(clientID, h.goal, -1,sim.simx_opmode_buffer);%other call: buffer
48  if errorCode ~= 0
49      error('Error with simxGetObjectPosition for goal');
50  end
51

```

Σχήμα 3.9 Εκχώρηση των συντεταγμένων της θέσης του αντικειμένου – στόχου σε μεταβλητή

Με τη βοήθεια της συνάρτησης `simGetObjectPosition`, λαμβάνεται η θέση σε συντεταγμένες τριών αξόνων, του αντικειμένου - στόχου στο χώρο. Η συνάρτηση δέχεται ως όρισμα τη μεταβλητή σύνδεσης Matlab και Coppeliasim, τη μεταβλητή που αντιστοιχεί στο εν λόγω αντικείμενο, τον αριθμό -1, ο οποίος ορίζει ότι οι συντεταγμένες θα ληφθούν με βάση το απόλυτο σύστημα (αρχή των αξόνων) και την ανάλογη συνάρτηση μετάδοσης δεδομένων, η οποία διαφέρει στα δύο καλέσματα, αφού το πρώτο γίνεται για ενεργοποίηση της συνάρτησης σε σχέση με το αντικείμενο και το δεύτερο για τη μετάδοση δεδομένων.

Αφού ληφθούν όλες οι απαραίτητες θέσεις και εκχωρηθούν σε μεταβλητές, αρχικοποιείται η ταχύτητα των τροχών (ίση με μηδέν), εκχωρείται σε δομή τύπου `struct` το διάνυσμα με τις συντεταγμένες των στόχων στους οποίους θα επιχειρήσει να κατευθυνθεί το όχημα και ως τελευταίο βήμα, το όχημα τίθεται μέσω συνάρτησης στην επιθυμητή

αρχική θέση και προσανατολισμό. Υπό αυτές τις επιλεγμένες από το χρήστη συνθήκες, θα γίνει η έναρξη του εκάστοτε σεναρίου – προσομοίωσης.

Αν το σενάριο εμπεριέχει την ύπαρξη εμποδίων, τότε υπάρχει ξεχωριστό κομμάτι αλγορίθμου, το οποίο προστίθεται στον κύριο κώδικα, για την ενεργοποίηση της μετάδοσης σήματος από τους αισθητήρες. Η διαδικασία αυτή επιτυγχάνεται με τη χρήση της αντίστοιχης συνάρτησης από τη βιβλιοθήκη, η οποία καλείται αρχικά δύο φορές, μία φορά για την έναρξη της μετάδοσης του σήματος και μία φορά για την καταγραφή των πρώτων δεδομένων από τους αισθητήρες.

### 3.7 ΑΛΓΟΡΙΘΜΟΣ/ΚΩΔΙΚΑΣ ΚΙΝΗΣΗΣ ΡΟΜΠΟΤΙΚΟΥ ΟΧΗΜΑΤΟΣ

Στην ενότητα αυτή, γίνεται αναλυτική περιγραφή των σημαντικότερων βημάτων του αλγορίθμου, με τη βοήθεια του οποίου ένα ρομποτικό όχημα κινείται σε άδεια αρένα ή σε αρένα με εμπόδια και καταφέρνει να φτάσει στο σημείο προορισμού, με τη βοήθεια κάποιου ασαφής ή νευρωνικού ελεγκτή, των συναρτήσεων της Matlab και τη σύνδεση της Matlab με τον προσομοιωτή CoppeliaSim. Ο αλγόριθμος αυτός αποτέλεσε τη βάση για την υλοποίηση των στόχων της παρούσας εργασίας.

Ο βασικός αλγόριθμος κίνησης, μέσα στον οποίο καλούνται συναρτήσεις, γίνονται υπολογισμοί και εξάγονται αποτελέσματα, με βάση τα οποία ελέγχεται το ρομπότ στον προσομοιωτή, αποτελεί στην ουσία μια συνάρτηση, η οποία μπορεί να κληθεί μέσα σε άλλα script. Η συνάρτηση αυτή, λαμβάνει ως είσοδο ένα ασαφές σύστημα ή ένα νευρωνικό δίκτυο και εξάγει τον αριθμό των βημάτων της εκάστοτε προσομοίωσης, την τελευταία καταγεγραμμένη απόσταση του ρομπότ από το στόχο και μια δομή (struct), με όλες τις απαραίτητες πληροφορίες από την προσομοίωση.

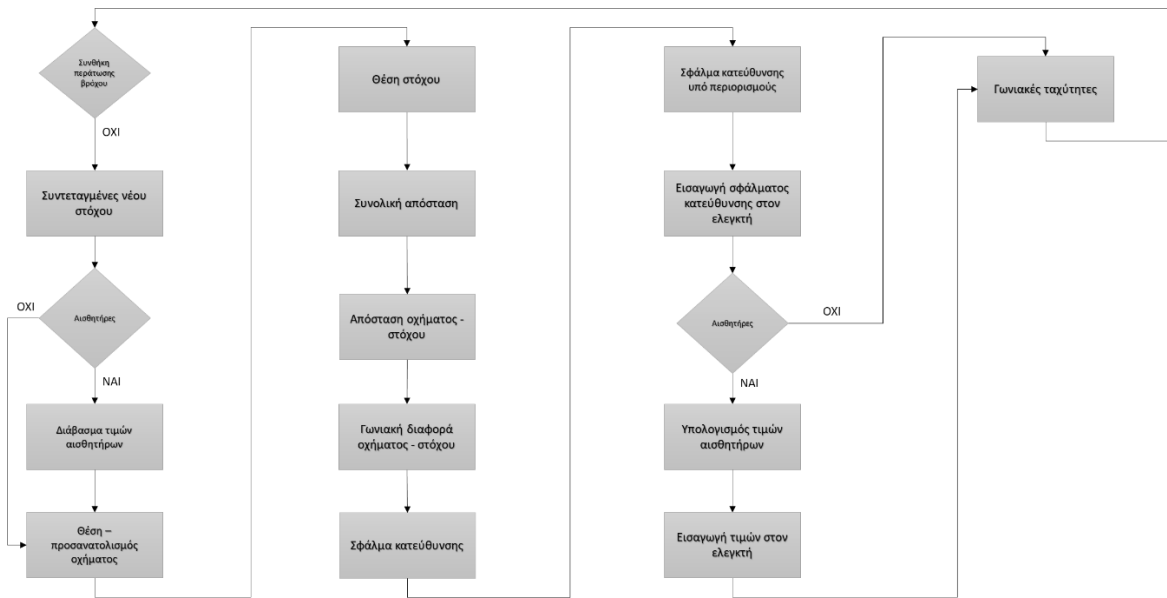
Το κύριο κομμάτι αλγορίθμου, για την εκτέλεση προσομοιώσεων, αφορά το βρόχο κίνησης while, με συνθήκη τερματισμού, τη τιμή της μεταβλητής stop να είναι ίση με 1. Ο βρόχος while αφορά την κίνηση μεταξύ αρχικής θέσης και σημείων στόχων, μέχρι το όχημα να περάσει από όλα τα σημεία που του έχουν ανατεθεί ως στόχοι και ως εκ τούτου να τερματίσει ο βρόχος, θέτοντας τη μεταβλητή stop ίση με 1. Υπάρχουν περιορισμοί, ώστε αν το όχημα πλησιάσει πολύ κοντά στο τοίχωμα ή στα εμπόδια, να τερματίζει ο βρόχος και συνεπώς η προσομοίωση.

#### Δομή βρόχου while υπό μορφή ψευδοκώδικα

- Εκχώρηση συντεταγμένων επόμενου στόχου
- Εάν υπάρχει χρήση αισθητήρων, διάβασμα τιμών αισθητήρων
- Λήψη συντεταγμένων θέσης και προσανατολισμού οχήματος
- Λήψη συντεταγμένων θέσης του αντικειμένου – στόχου
- Υπολογισμός συνολικής απόστασης που έχει διανυθεί έως τώρα

- Υπολογισμός ευκλείδειας απόστασης μεταξύ οχήματος και στόχου
- Υπολογισμός γωνιακής διαφοράς οχήματος και στόχου
- Υπολογισμός σφάλματος κατεύθυνσης
- Εκ νέου υπολογισμός σφάλματος κατεύθυνσης μετά από περιορισμούς
- Εισαγωγή του σφάλματος κατεύθυνσης στον ελεγκτή
- Εάν υπάρχει χρήση αισθητήρων, υπολογισμός των κατάλληλων τιμών των αισθητήρων
- Εισαγωγή τιμών αισθητήρων στον ελεγκτή υπό μορφή ομάδων
- Εξαγωγή γωνιακών ταχυτήτων από τον ελεγκτή

Στο σχήμα 3.10 παρουσιάζεται υπό μορφή διαγράμματος ροής ο παραπάνω ψευδοκώδικας.



Σχήμα 3.10 Διάγραμμα ροής βρόχου κίνησης οχήματος

Αναλυτικά ο βρόχος while και οι υπολογισμοί σε αυτόν περιγράφονται στη συνέχεια.

Αρχικά, διαβάζονται οι συντεταγμένες του επόμενου στόχου και τίθεται το αντικείμενο-στόχος της αρένας στο αντίστοιχο σημείο. Διαβάζονται μέσω κατάλληλων συναρτήσεων οι τιμές στους αισθητήρες, ανανεώνονται οι μεταβλητές θέσης και προσανατολισμού οχήματος και θέσης στόχου και υπολογίζεται η συνολική απόσταση που έχει διανύσει το όχημα έως την τρέχουσα στιγμή. Η απόσταση που κάνει κάθε φορά το όχημα μεταξύ δύο διαδοχικών επαναλήψεων (steps), μετρίεται και προστίθεται στο ήδη υπάρχον άθροισμα απόστασης, βρίσκοντας έτσι την εκ νέου συνολική απόσταση κάθε φορά. Στη συνέχεια, υπολογίζεται η ευκλείδεια απόσταση μεταξύ οχήματος και στόχου ως εξής:

$$vp2goal = \sqrt{(x_{goal} - x_{vehicle})^2 + (y_{goal} - y_{vehicle})^2} \quad (3.12)$$

Ο υπολογισμός της γωνιακής διαφοράς μεταξύ ρομπότ και στόχου γίνεται με τη βοήθεια της atan2 ως εξής:

$$\theta = \tan^{-1} \frac{y_{goal} - y_{vehicle}}{x_{goal} - x_{vehicle}} \quad (3.13)$$

Τελικώς, αν θεωρήσουμε ως  $\phi_{vehicle}$  τη μεταβλητή που δηλώνει τον προσανατολισμό του οχήματος στο χώρο, δηλαδή τη γωνία  $\gamma$ , αφού το όχημα περιστρέφεται γύρω από τον κατακόρυφο άξονα  $z$ , εξάγεται το σφάλμα κατεύθυνσης ως η αφαίρεση του προσανατολισμού και της γωνιακής διαφοράς:

$$\phi = \phi_{vehicle} - \theta \quad (3.14)$$

Επειδή η  $\tan^{-1}$  αφορά το καρτεσιανό επίπεδο σε εύρος γωνίας  $[-\pi, \pi]$ , ενώ το όχημα μπορεί να βρίσκεται παντού στο χώρο, σε εύρος  $[0, 2\pi]$ , πρέπει να ληφθούν οι εξής περιορισμοί:

$$\begin{cases} \phi = \phi + 2\pi, & \text{εάν } \phi < -\pi \\ \phi = \phi - 2\pi, & \text{εάν } \phi \geq \pi \end{cases} \quad (3.15)$$

Το σφάλμα κατεύθυνσης, αφού υπολογιστεί εκ νέου μετά τους περιορισμούς, μετατρέπεται από ακτίνια σε μοίρες, με τη βοήθεια συνάρτησης της Matlab.

Εάν γίνεται πλοήγηση σε περιβάλλον με εμπόδια, τότε το επόμενο βήμα στον αλγόριθμο είναι ο έλεγχος των τιμών του κάθε αισθητήρα και στη συνέχεια η ομαδοποίηση τους για να εισαχθούν στον ελεγκτή. Για την πλοήγηση στην παρούσα εργασία, χρησιμοποιούνται οι 8 (μπροστινοί) από τους 16 συνολικά αισθητήρες του οχήματος.

Για κάθε αισθητήρα, εάν η ένδειξη η οποία έχει καταγραφεί από προηγούμενες εντολές, είναι μικρότερη από 1, τότε η τιμή της αντίστοιχης μεταβλητής του αισθητήρα τίθεται ίση με 1, που σημαίνει ότι δεν έχει ανιχνευθεί κάποιο εμπόδιο.

Αν ωστόσο η τιμή είναι μεγαλύτερη ή ίση με 1, σημαίνει ότι έχει ανιχνευθεί κάποιο αντικείμενο στο πεδίο του αισθητήρα. Τότε, η τιμή που εκχωρείται στη μεταβλητή μέτρησης, είναι ίση με τη νόρμα της ένδειξης του αισθητήρα που έχει σταλεί στη matlab.

Στο σχήμα 3.11 παρουσιάζεται ενδεικτικά απόσπασμα κώδικα που αφορά όλη τη διαδικασία ελέγχου για τον αισθητήρα no 1.

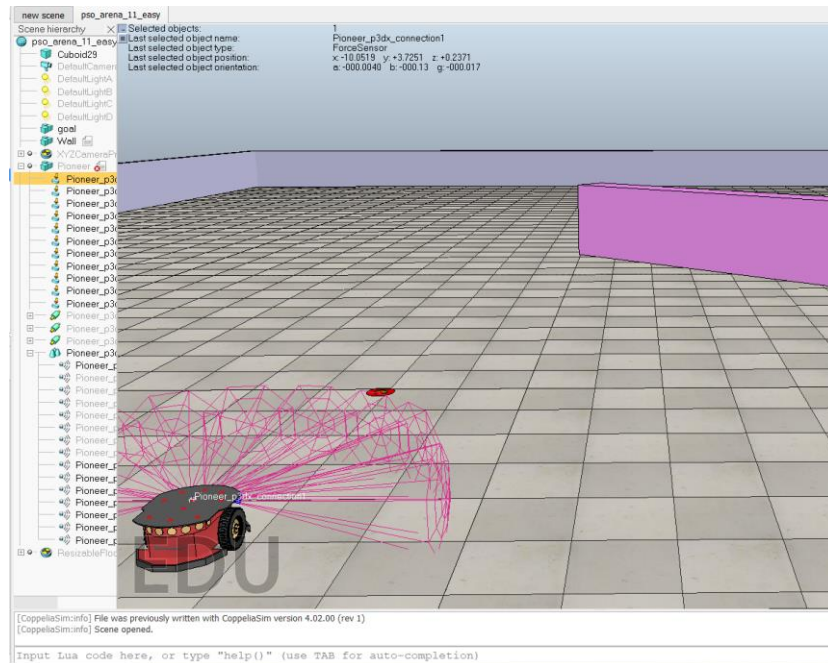
```

301 -     if detect1<1
302 -         s1 = 1;
303 -     else
304 -         s1 = norm(detectedP1);
305 -     end

```

Σχήμα 3.11 Έλεγχος τιμών αισθητήρα νο 1

Κάθε αισθητήρας έχει ένα εύρος μέτρησης, οι ακτίνες του δηλαδή καταλαμβάνουν ένα συγκεκριμένο νοητό όγκο στο χώρο, μέσα στον οποίο, αν υπάρξει αντικείμενο, ενεργοποιείται ο αισθητήρας. Οι ακτίνες των αισθητήρων που χρησιμοποιούνται εδώ, έχουν κωνικό σχήμα, στον όγκο που καταλαμβάνουν (σχήμα 3.12). Η νόρμα υπολογίζεται, ώστε να κανονικοποιηθεί η μέτρηση στο σχήμα τριών διαστάσεων και να μετατραπεί σε αριθμό στο διάστημα από μηδέν έως ένα.



Σχήμα 3.12 Κωνικό εμβαδό αισθητήρων

Για λόγους μείωσης του όγκου της πληροφορίας και απλούστευσης του προβλήματος, δεν εισάγονται και οι 8 τιμές των αισθητήρων στον ελεγκτή, αλλά η διαδικασία αυτή γίνεται σε ομάδες. Οι αισθητήρες 1 έως 3 είναι αυτοί που βρίσκονται αριστερά στο όχημα, οι αισθητήρες 4 και 5 κεντρικά και 6 έως 8 δεξιά. Για κάθε ομάδα, θα ληφθεί η μικρότερη τιμή από τις αντίστοιχες μεταβλητές των αισθητήρων, όπως αυτές έχουν προκύψει από τη διαδικασία ελέγχου που έχει προηγηθεί. Συνεπώς, προκύπτουν τρεις τελικές τιμές, μία από κάθε ομάδα και αυτές εισάγονται στον ελεγκτή.

Τέλος, ως αποτέλεσμα, εξάγονται από τον ελεγκτή οι τιμές των γωνιακών ταχυτήτων αριστερού και δεξιού τροχού και μέσω κατάλληλων συναρτήσεων στέλνονται στον προσομοιωτή, ώστε να κινηθεί αναλόγως το όχημα.

Σημειώνεται ότι, όταν το όχημα φτάσει σε μια επιθυμητή ακτίνα -ορισμένη από το χρήστη-, δηλαδή μια πολύ κοντινή απόσταση από το στόχο, τότε το σημείο-στόχος θεωρείται ότι «πιάστηκε» και η ίδια διαδικασία (βρόχος) συνεχίζεται, για διαφορετικό σημείο- στόχο. Ο βρόχος τερματίζει όταν το όχημα περάσει από όλα τα σημεία στόχους, εκτός αν υπάρξει ανεπιθύμητη συμπεριφορά στην πορεία, όπως περιεγράφηκε παραπάνω.

## ΚΕΦΑΛΑΙΟ 4

# ΥΛΟΠΟΙΗΣΗ ΣΕΝΑΡΙΩΝ ΜΕ ΧΡΗΣΗ ΑΣΑΦΟΥΣ/ΝΕΥΡΩΝΙΚΟΥ ΕΛΕΓΚΤΗ

### 4.1 ΕΙΣΑΓΩΓΗ

Στο κεφάλαιο αυτό, παρουσιάζονται τα σενάρια τα οποία δημιουργήθηκαν για τον πειραματισμό, καθώς και οι αντίστοιχες αρένες διαφορετικής πολυπλοκότητας για την εφαρμογή τους, αναλύεται η δομή των διάφορων ασαφών και νευρωνικών ελεγκτών που κατασκευάστηκαν, για χρήση σε περιβάλλοντα με ή χωρίς εμπόδια, παρουσιάζονται υπό μορφή πινάκων και διαγραμμάτων τα αποτελέσματα των προσομοιώσεων και εξάγονται συμπεράσματα.

### 4.2 ΣΕΝΑΡΙΑ ΚΛΙΜΑΚΟΥΜΕΝΗΣ ΠΟΛΥΠΛΟΚΟΤΗΤΑΣ

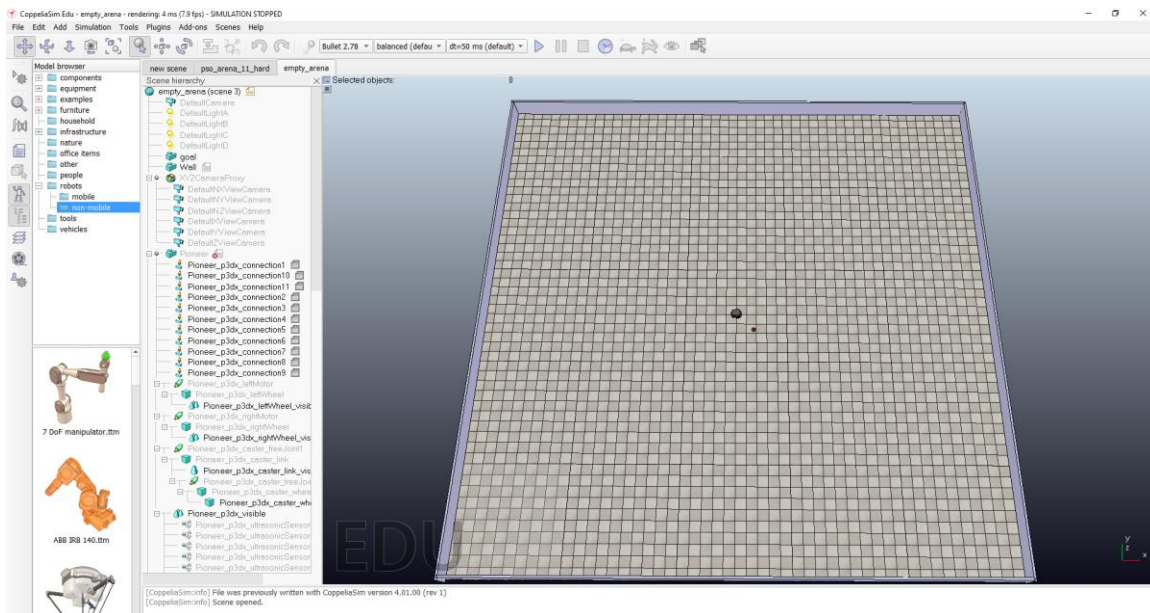
Για την υλοποίηση των προσομοιώσεων και την εξαγωγή πλήρων συμπερασμάτων σε διαφορετικές συνθήκες, δημιουργήθηκαν αρένες βαθμωτής δυσκολίας, οι οποίες αντιπροσωπεύουν πιθανές καταστάσεις στις οποίες μπορεί να βρεθεί ένα ρομποτικό όχημα. Για τις ανάγκες αυτές, χρησιμοποιήθηκε χώρος πειραματισμού διαστάσεων 25 x 25 m, εμπόδια δύο διαφορετικών διαστάσεων και συγκεκριμένα: 5 x 0.5 x 1 m και 8 x 0.5 x 1 m και τοίχιο περιβολής του χώρου πειραματισμού με διαστάσεις 25.75 x 25.10 x 1 m.

Για να μπορεί το όχημα να ολοκληρώνει σε ικανοποιητικό βαθμό τα σενάρια, χωρίς να παγιδεύεται και να είναι αναποτελεσματικό, έγιναν κάποιες παραδοχές, μετά από πρώιμο πειραματισμό. Συνεπώς, μία αρένα με τις παραπάνω διαστάσεις στοιχείων, μπορεί για τις ανάγκες της εργασίας να θεωρηθεί πλήρης, όταν βρίσκονται σε αυτή δώδεκα εμπόδια (max capacity). Με βάση αυτήν την παραδοχή, οι αρένες δημιουργήθηκαν και κατηγοριοποιήθηκαν με γνώμονα την ποσοστιαία πληρότητα σε εμπόδια, η οποία ορίζεται ως το κλάσμα του αριθμού των εμποδίων μιας αρένας, ως προς το μέγιστο αριθμό εμποδίων που θεωρείται ότι μπορεί να φέρει ένας χώρος πειραματισμού. Οι αρένες, συμπερασματικά, αντιπροσωπεύουν σενάρια με διαφορετική πολυπλοκότητα και κλιμακούμενη δυσκολία.

No.	ΠΛΗΡΟΤΗΤΑ (%)	ΒΑΘΜΟΣ ΠΟΛΥΠΛΟΚΟΤΗΤΑΣ
1.	0	Πολύ εύκολος
2.	8	Εύκολος
3.	25	Μέτριας δυσκολίας
4.	58	Δύσκολος
5.	83	Πολύ δύσκολος

Πίνακας 4.1 Σενάρια κλιμακούμενης πολυπλοκότητας ως προς την πληρότητα της αρένας

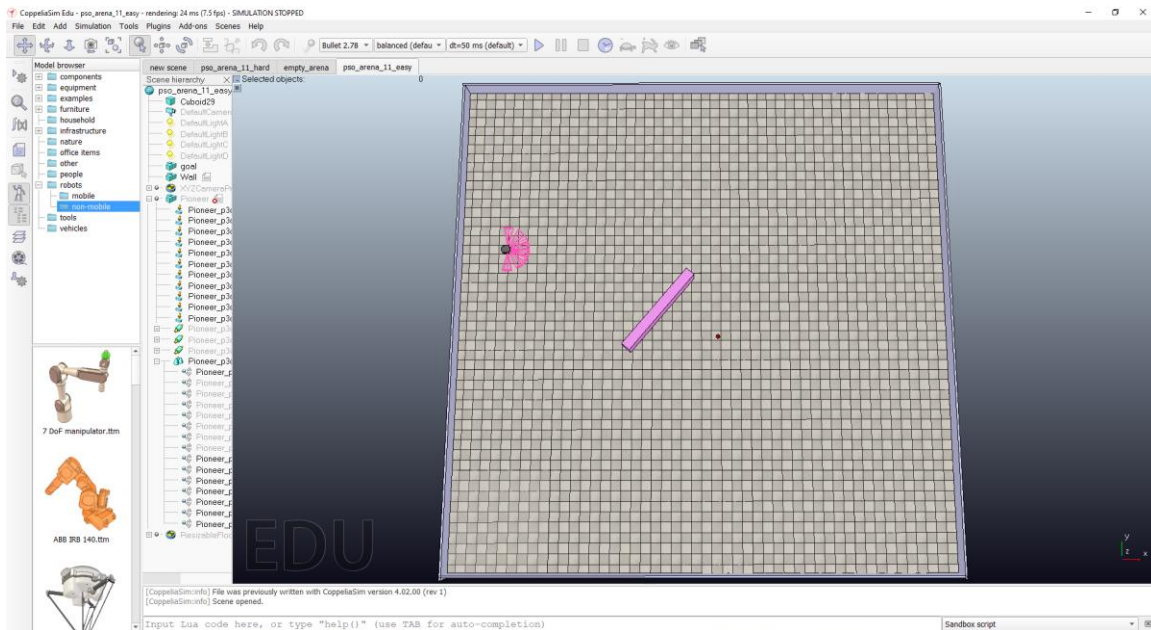
Στη συνέχεια παρουσιάζονται οι αντίστοιχοι χώροι πειραματισμού, που αντιπροσωπεύουν τα σενάρια διαφορετικής δυσκολίας. Συγκεκριμένα στο Σχήμα 4.1 παρουσιάζεται η απλούστερη εκδοχή ενός χώρου πειραματισμού, με πληρότητα 0%, αποτελούμενος από ένα άδειο χώρο και τοιχώματα. Είναι ο πρώτος χώρος πειραματισμού στον οποίο δοκιμάζεται η συμπεριφορά ασαφών και νευρωνικών ελεγκτών, πριν εισαχθούν σε περιβάλλον εμποδίων, με πολυπλοκότερους ελεγκτές.



Σχήμα 4.1 Αρένα χωρίς εμπόδια (0%), βαθμός δυσκολίας “πολύ εύκολος”

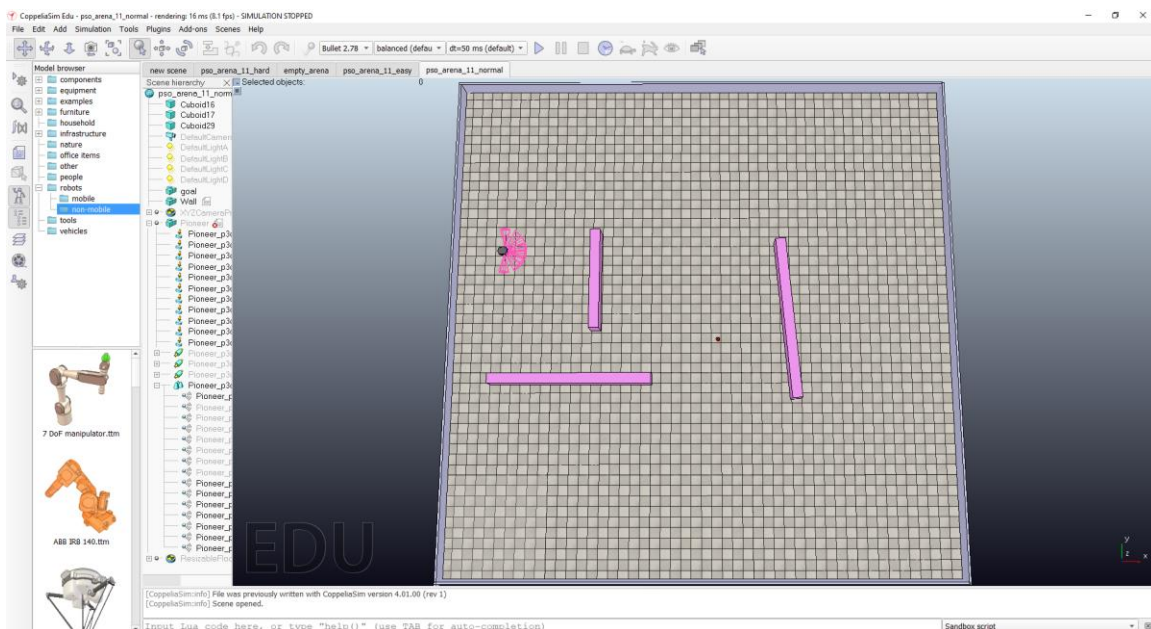
Στο σχήμα 4.2, παρουσιάζεται το λιγότερο πολύπλοκο σενάριο σε περιβάλλον με εμπόδια, μια αρένα πληρότητας 8%, με ένα εμπόδιο. Χρησιμοποιείται κυρίως για την επιβεβαίωση σεναρίων, για την επιβεβαίωση δηλαδή ότι ένας ελεγκτής προς δοκιμή, αποφεύγει το εμπόδιο και φτάνει στο στόχο.



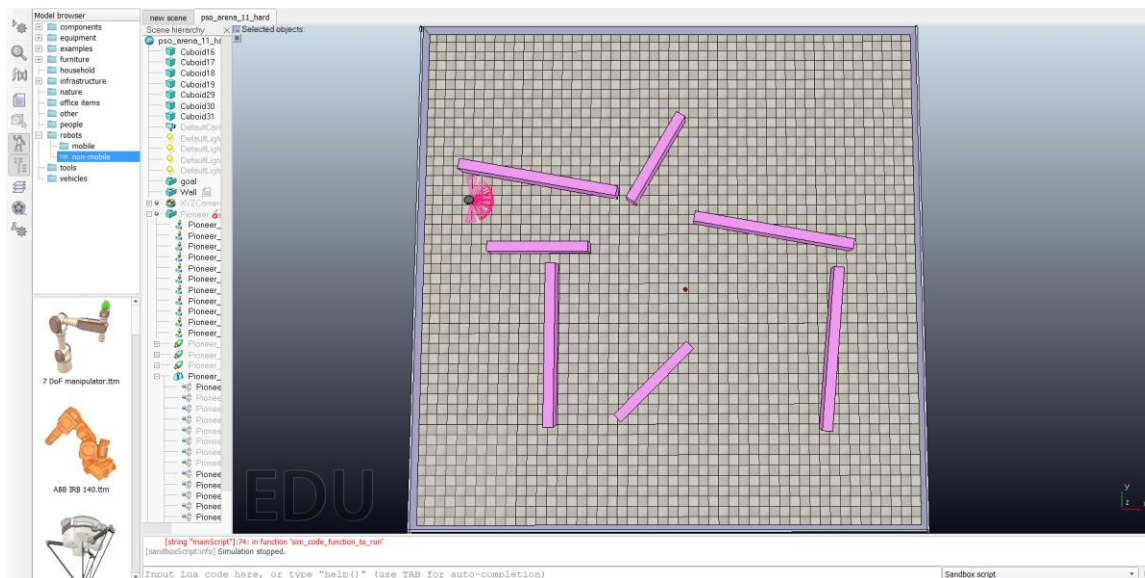


Σχήμα 4.2 Αρένα με ένα εμπόδιο (8%), βαθμός «εύκολος»

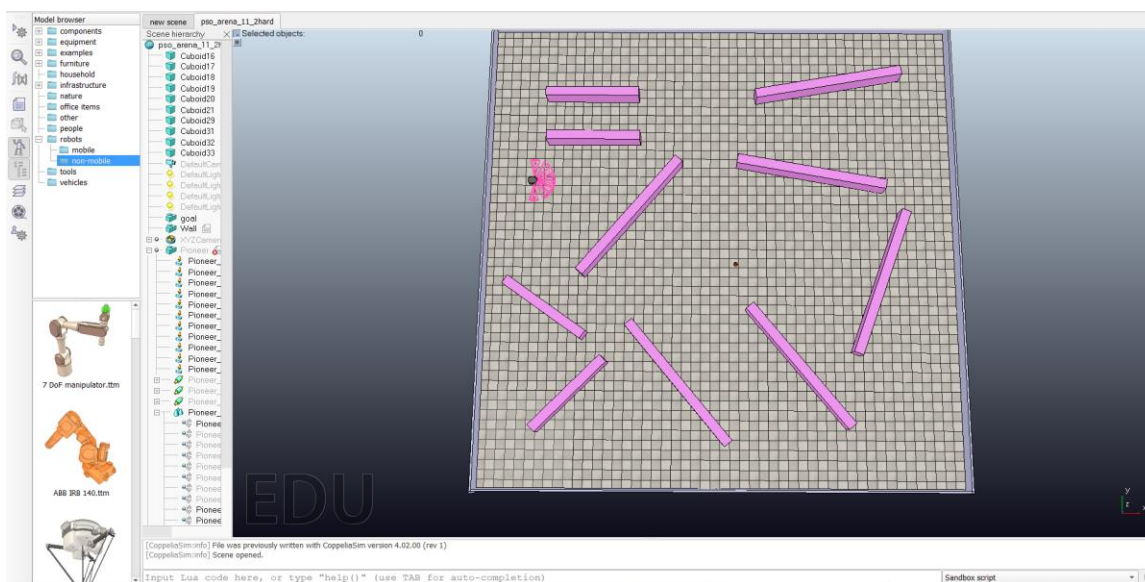
Στα σχήματα 4.3, 4.4, 4.5, απεικονίζονται οι αρένες που αφορούν τα σενάρια με μέτρια έως μεγάλη πληρότητα σε εμπόδια και ο τρόπος διάταξης των εμποδίων σε κάθε περίπτωση. Τα εμπόδια μπορεί να είναι παράλληλα ή κάθετα μεταξύ τους, να σχηματίζουν γωνίες, που να επιτρέπουν ή να μην επιτρέπουν τη διέλευση του οχήματος. Ο τρόπος διάταξης τους αντιπροσωπεύει διάφορες καταστάσεις στις οποίες μπορεί να βρεθεί το όχημα και να πρέπει να επιλύσει ο ελεγκτής.



Σχήμα 4.3 Αρένα με τρία εμπόδια (25%), βαθμός «μέτριας δυσκολίας»



Σχήμα 4.4 Αρένα με επτά εμπόδια (58%), βαθμός «δύσκολος»



Σχήμα 4.5 Αρένα με 10 εμπόδια (83%), βαθμός «πολύ δύσκολος»

### 4.3 ΑΣΑΦΕΙΣ ΕΛΕΓΚΤΕΣ ΤΥΠΟΥ ΜΑΜΔΑΝΙ ΓΙΑ ΚΙΝΗΣΗ PIONEER 3-DX

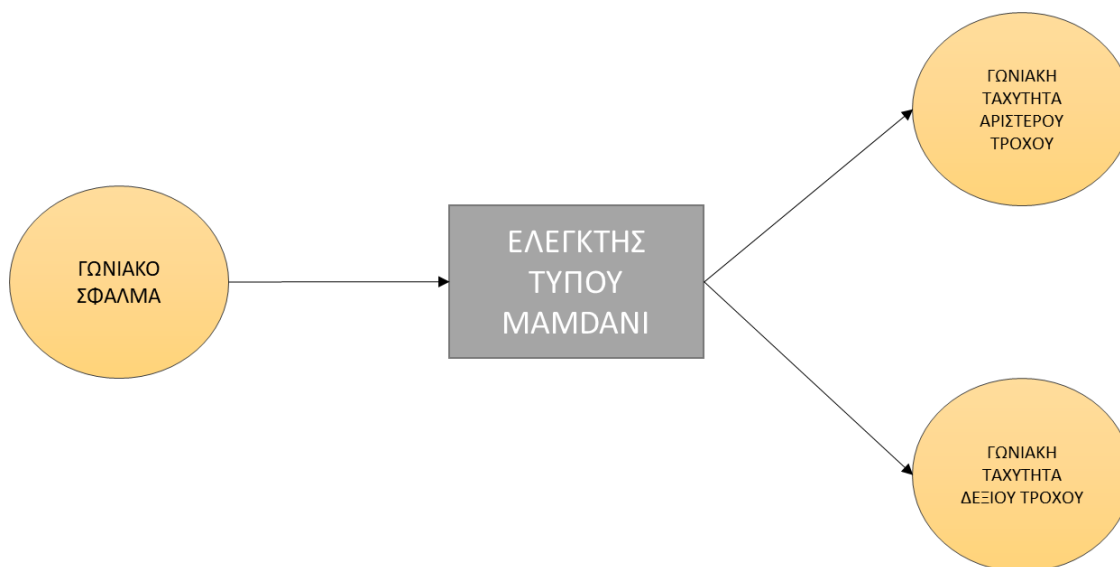
Η κίνηση και συμπεριφορά του ρομποτικού οχήματος στο χώρο, εξαρτάται από το είδος και την ποιότητα του αντίστοιχου ελεγκτή του. Για της ανάγκες της παρούσας εργασίας, αναπτύχθηκαν σε πρώτο στάδιο, ελεγκτές ασαφούς λογικής, με αυξανόμενη πολυπλοκότητα και πληροφορία, ξεκινώντας δηλαδή από τη δημιουργία απλών σχετικά

ελεγκτών και καταλήγοντας σε πιο σύνθετους, με σκοπό την παρατήρηση της αλληλεπίδρασης του οχήματος με τους στόχους και το περιβάλλον, υπό την καθοδήγηση δομικά διαφορετικών ελεγκτών. Τελικός στόχος, η σύγκριση και εξαγωγή συμπερασμάτων, καθώς και η χρήση κάποιων από αυτούς για μετέπειτα εκπαίδευση νευρωνικών δικτύων.

Η παρουσίαση των ελεγκτών μπορεί να χωριστεί, λόγω διαφορών στην κύρια δομή τους, σε αυτούς που αφορούν περιβάλλον χωρίς εμπόδια και σε αυτούς για περιβάλλον με εμπόδια. Άλλωστε, η συλλογιστική πορεία για τη δημιουργία τους, ακολούθησε αυτή την κατεύθυνση.

#### 4.3.1 ΚΑΤΑΣΚΕΥΗ ΑΣΑΦΩΝ ΕΛΕΓΚΤΩΝ ΓΙΑ ΚΙΝΗΣΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΧΩΡΙΣ ΕΜΠΟΔΙΑ

Η δομή όλων των ελεγκτών, στην αρένα με τη χαμηλότερη πολυπλοκότητα, έχουν ως είσοδο στο σύστημα, το γωνιακό σφάλμα κατεύθυνσης, το οποίο υπολογίζεται από τον αλγόριθμο της ενότητας 3.7 και ως έξοδο, τις γωνιακές ταχύτητες των δύο διαφορεικά περιστρεφόμενων τροχών.

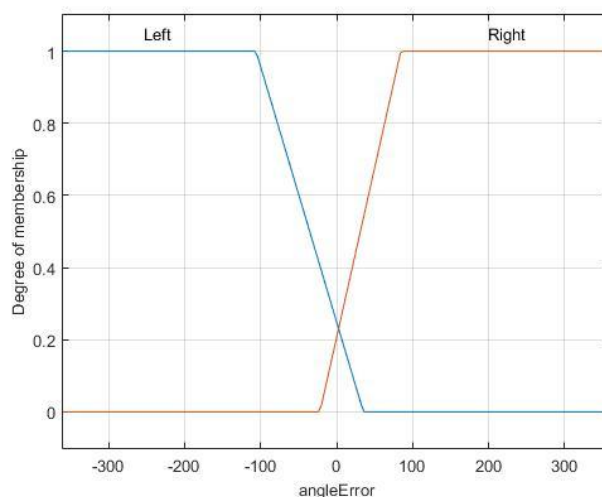


Σχήμα 4.6 Δομή ασαφούς ελεγκτή σε άδεια αρένα

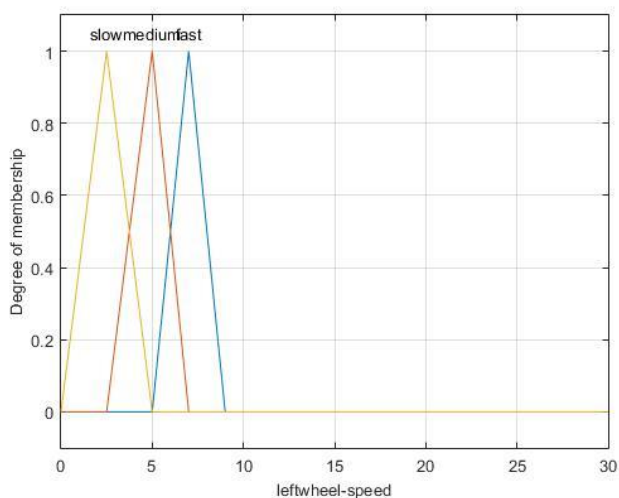
#### Ασαφής Ελεγκτής 1 (fuzzy 1)

Ο πρώτος ελεγκτής που κατασκευάστηκε, εκπροσωπεί την πιο απλουστευμένη λογική στην κίνηση του ρομποτικού οχήματος σε άδεια αρένα. Τα ασαφή σύνολα εισόδου επιλέχτηκαν να είναι δύο τραπέζια, τα οποία αφορούν τις παραμέτρους «δεξιά» και «αριστερά», δηλαδή τη θέση του στόχου σε σχέση με τη θέση του οχήματος την εκάστοτε

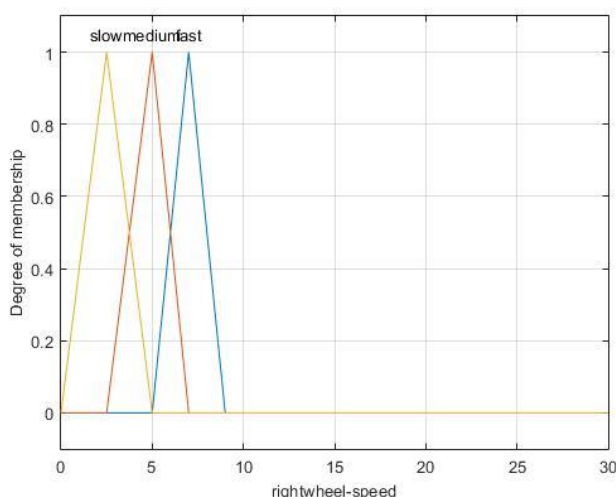
στιγμή. Σε ότι αφορά την έξοδο, επελέγησαν τρία σύνολα για τον κάθε τροχό, «αργά», «μέτρια», «γρήγορα», ως υποκατηγορίες της ταχύτητας των τροχών. Σημειώνεται ότι όλες οι ταχύτητες στα σχήματα που ακολουθούν, είναι σε rad/s. Στα σχήματα 4.7, 4.8, 4.9, παρουσιάζονται τα ασαφή σύνολα για τις μεταβλητές εισόδου και εξόδου, για τον ασαφή ελεγκτή 1. Ανάλογα με τις τιμές κάθε παραμέτρου στην είσοδο και έξοδο, καταλαμβάνεται και το αντίστοιχο εμβαδό που αντιστοιχεί σε κάθε συνάρτηση συμμετοχής.



Σχήμα 4.7 Είσοδος «σφάλμα κατεύθυνσης» ασαφούς ελεγκτή «fuzzy 1»



Σχήμα 4.8 Έξοδος «ταχύτητα αριστερού τροχού» ασαφούς ελεγκτή «fuzzy1»



Σχήμα 4.9 Εξοδος «ταχύτητα δεξιού τροχού» ασαφούς ελεγκτή «fuzzy1»

Μετά την εισαγωγή των συναρτήσεων συμμετοχής εισόδου-εξόδου, διατυπώνονται οι λεκτικοί κανόνες, με βάση τους οποίους θα διαμορφώνεται η σύζευξη των ασαφών συνόλων και θα καθορίζεται η στιγμιαία συμπεριφορά του οχήματος.

Οι κανόνες συνθέτονται με βάση την εξής λογική:

ΕΑΝ...ΣΥΝΘΗΚΗ...ΤΟΤΕ ... ΕΝΤΟΛΗ 1, ΕΝΤΟΛΗ 2.

Η σχετική βάση κανόνων παρουσιάζεται στον Πίνακα 4.1.

ΛΕΚΤΙΚΟΙ ΚΑΝΟΝΕΣ				
	<u>Σφάλμα</u> <u>Κατεύθυνσης</u>		<u>Ταχύτητα</u> <u>Αριστερού</u> <u>Τροχού</u>	<u>Ταχύτητα</u> <u>Δεξιού</u> <u>Τροχού</u>
<b>Εάν</b>	Αριστερά	<b>Τότε</b>	Αργά	Μέτρια
	Δεξιά		Μέτρια	Αργά

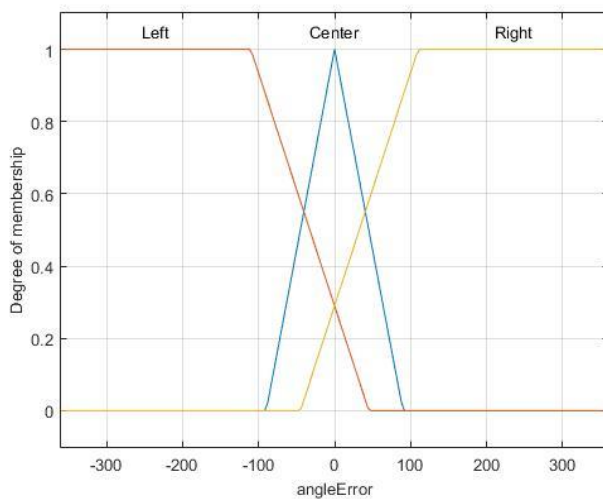
Πίνακας 4.2 Λεκτικοί κανόνες συστήματος «fuzzy1»

Έχοντας δημιουργήσει τον παραπάνω ελεγκτή, ακολουθεί ο σταδιακός εμπλουτισμός του με πληροφορία, εξάγοντας καινούργια συστήματα, τα: fuzzy2, fuzzy 3, fuzzy 4 και fuzzy0.

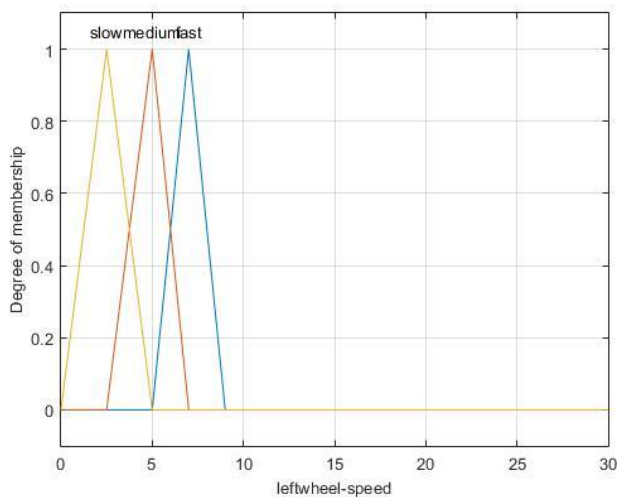
### Ασαφής Ελεγκτής 2 (fuzzy 2)

Σε αυτόν τον ελεγκτή, εισήχθη άλλη μια παράμετρος στην είσοδο, η τριγωνική συνάρτηση συμμετοχής «κεντρικά», δημιουργώντας ένα καινούργιο σύνολο, που αφορά ένα εύρος τιμών, το οποίο είναι σημαντικό να διαχωριστεί σε μερικό βαθμό από τις συναρτήσεις «αριστερά», «δεξιά». Το σύνολο αφορά τις γωνιακές τιμές (σε μοίρες)  $[-90,90]$ , λόγω όμως της επιλογής τριγωνικής συνάρτησης, μεγάλο βαθμό συμμετοχής έχει

στις τιμές  $[-45,45]$ . Στα σχήματα 4.10, 4.11, 4.12, παρουσιάζονται τα σύνολα των παραμέτρων εισόδου – εξόδου, για τον ασαφή ελεγκτή 2.

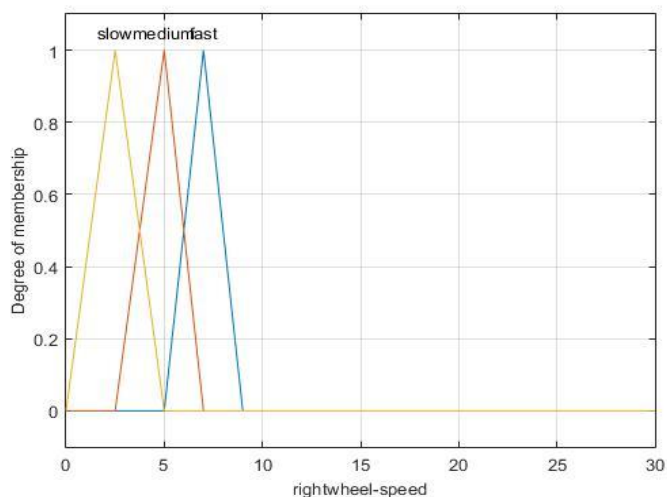


Σχήμα 4.10 Είσοδος «σφάλμα κατεύθυνσης» συστήματος «fuzzy2», τριών ασαφών συνόλων



Σχήμα 4.11 Εξοδος «ταχύτητα αριστερού τροχού» ασαφούς ελεγκτή «fuzzy2»





Σχήμα 4.12 Έξοδος «ταχύτητα δεξιού τροχού» ασαφούς ελεγκτή «fuzzy2»

Με την εισαγωγή επιπλέον ασαφούς συνόλου στην είσοδο, δημιουργείται ανάγκη για επιπλέον κανόνες, οι οποίοι εξακολουθούν να έχουν την ίδια λογική στη δομή τους με πριν. Η βάση κανόνων διαμορφώνεται όπως ακριβώς παρουσιάζεται στον πίνακα 4.3.

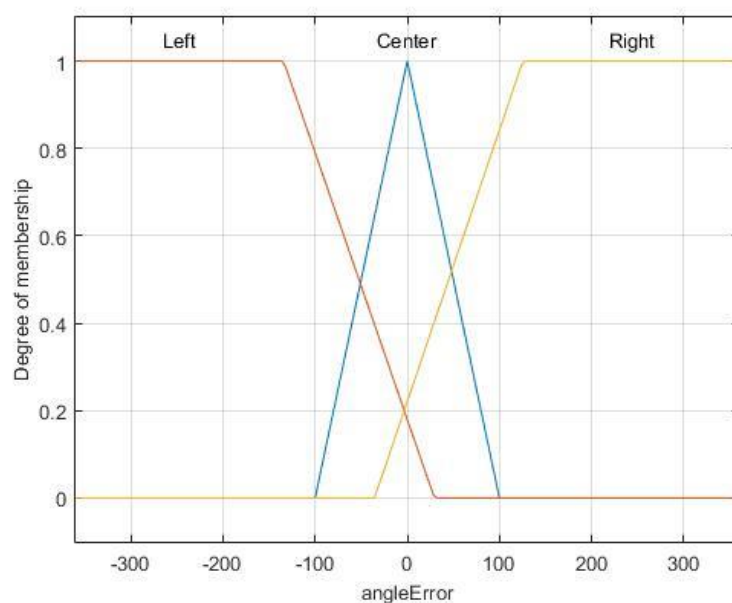
ΛΕΚΤΙΚΟΙ ΚΑΝΟΝΕΣ				
	<u>Σφάλμα</u> <u>Κατεύθυνσης</u>		<u>Ταχύτητα</u> <u>Αριστερού</u> <u>Τροχού</u>	<u>Ταχύτητα</u> <u>Δεξιού</u> <u>Τροχού</u>
<b>Εάν</b>	Αριστερά	<b>Τότε</b>	Αργά	Μέτρια
	Δεξιά		Μέτρια	Αργά
	Κεντρικά		Γρήγορα	Γρήγορα

Πίνακας 4.3 Λεκτικοί κανόνες συστήματος «fuzzy 2»

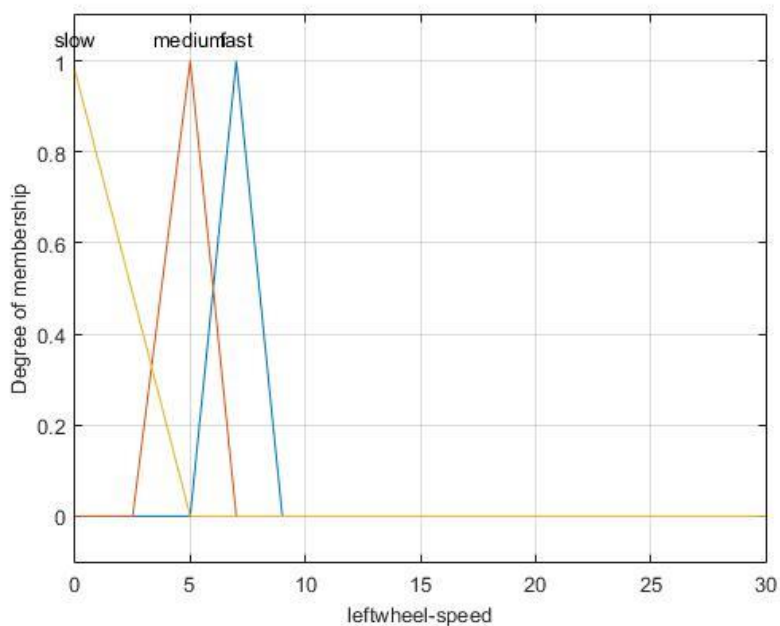
### Ασαφής Ελεγκτής 3 (fuzzy 3)

Δημιουργώντας τον τρίτο ελεγκτή, σκοπός ήταν η παρέμβαση σε πολύ μικρό βαθμό, σε κάποια από τα μέρη του συστήματος «fuzzy2», ώστε να παρατηρηθεί, εάν και πόση διαφορά θα υπάρξει στο πρακτικό κομμάτι. Έτσι, τα όρια του συνόλου εισόδου «κεντρικά», μετατοπίστηκαν στο εύρος  $[-100,100]$ , ενώ μετατοπίστηκε και το τρίγωνο της εξόδου «αργά», με τρόπο ώστε η κορυφή του να αντιστοιχεί σε ταχύτητα 0. Έτσι, αν για παράδειγμα η ταχύτητα κάποιου τροχού είναι 0 rad/s, η συνάρτηση συμμετοχής για το σύνολο «αργά» ενεργοποιείται πλήρως σε αντίθεση με τα προηγούμενα συστήματα. Στα

σχήματα 4.13, 4.14, 4.15, παρουσιάζονται τα ασαφή σύνολα για τις μεταβλητές εισόδου – εξόδου του ασαφή ελεγκτή 3.

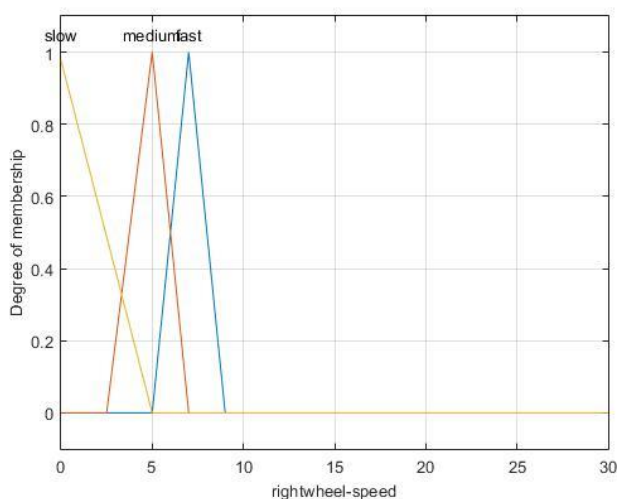


Σχήμα 4.13 Είσοδος «σφάλμα κατεύθυνσης» συστήματος «fuzzy3»



Σχήμα 4.14 Εξόδος «ταχύτητα αριστερού τροχού» ασαφούς ελεγκτή «fuzzy3»

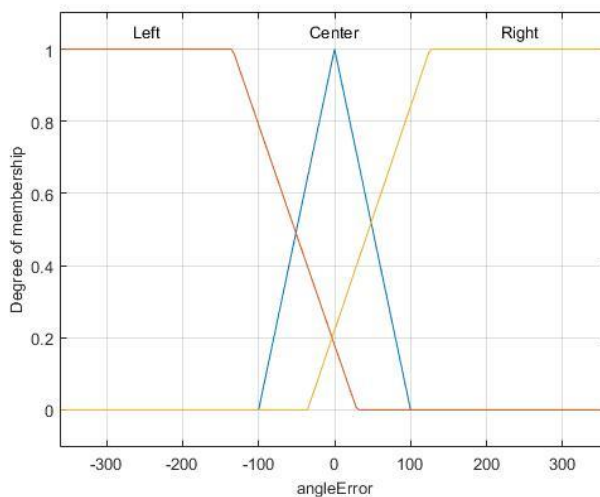




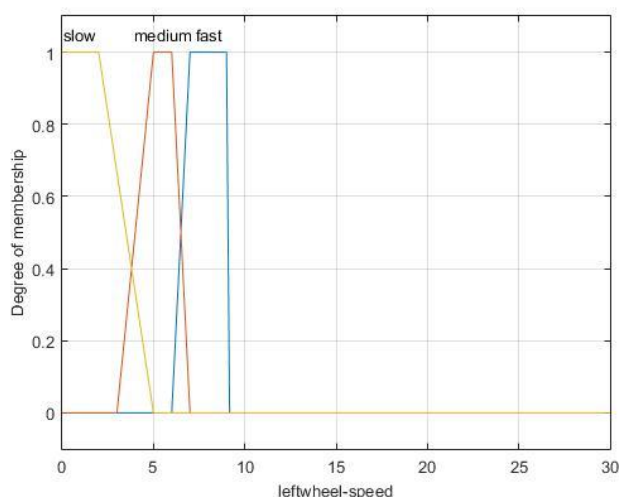
Σχήμα 4.15 Έξοδος «ταχύτητα δεξιού τροχού» ασαφούς ελεγκτή «fuzzy3»

#### **Ασαφής Ελεγκτής 4(fuzzy 4)**

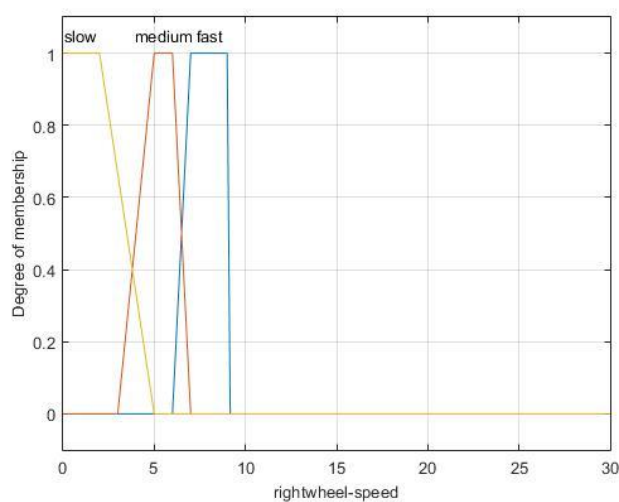
Το ασαφές σύστημα fuzzy4, σχεδιάστηκε, έχοντας ως δομή την είσοδο του συστήματος fuzzy3, αλλάζοντας όμως πλήρως τις εξόδους. Οι τριγωνικές συναρτήσεις μετατράπηκαν σε τραπέζια, δίνοντας έτσι τη δυνατότητα στο σύστημα να έχει μεγαλύτερες τιμές ενεργοποίησης για μεγαλύτερο εύρος τιμών. Στα σχήματα 4.16, 4.17, 4.18, παρουσιάζονται τα ασαφή σύνολα για τις μεταβλητές εισόδου – εξόδου του ασαφή ελεγκτή 4, όπου είναι εμφανείς οι αλλαγές.



Σχήμα 4.16 Είσοδος «σφάλμα κατεύθυνσης» συστήματος «fuzzy4»



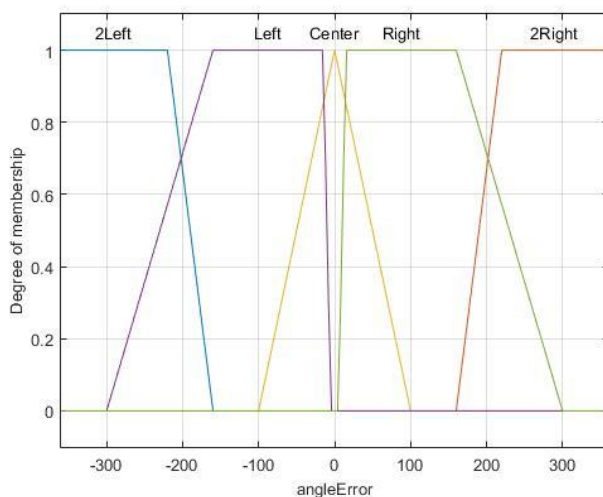
Σχήμα 4.17 Έξοδος «ταχύτητα αριστερού τροχού» ασαφούς ελεγκτή «fuzzy4»



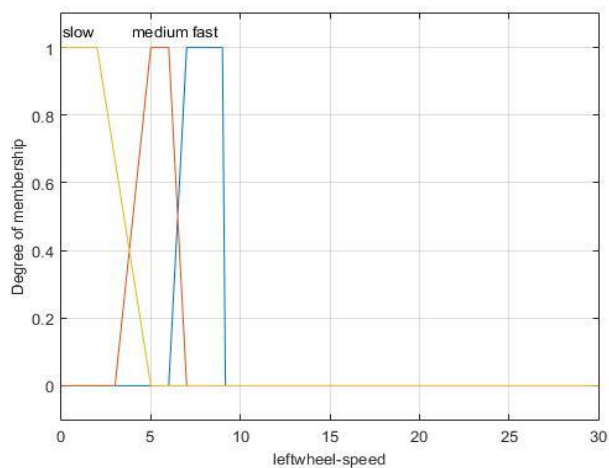
Σχήμα 4.18 Έξοδος «ταχύτητα δεξιού τροχού» ασαφούς ελεγκτή «fuzzy4»

### **Ασαφής Ελεγκτής 0(fuzzy 0)**

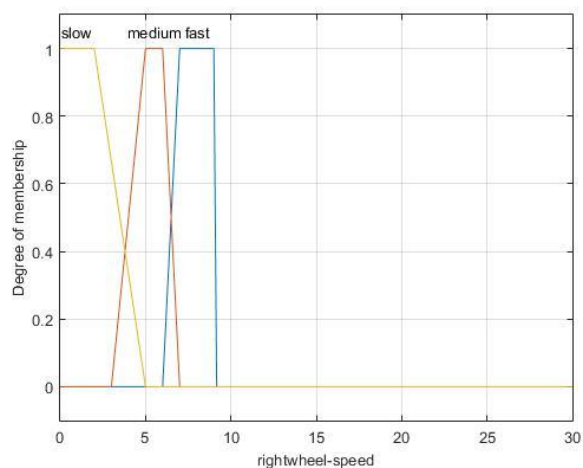
Το ασαφές σύστημα αυτό, σχεδιάστηκε ως το πιο σύνθετο, αφού σε συνέχεια του fuzzy4, προστέθηκε επιπλέον πληροφορία στην είσοδο του συστήματος, με δύο σύνολα ακόμα («πολύ δεξιά», «πολύ αριστερά»), αυξάνοντας τους κανόνες αλλά και τις επιλογές του ρομποτικού οχήματος στην έξοδο. Στα σχήματα 4.19, 4.20, 4.21, παρουσιάζονται τα σύνολα αυτά για τις μεταβλητές εισόδου – εξόδου του ασαφούς ελεγκτή 0, όπου είναι εμφανείς οι αλλαγές στη μεταβλητή εισόδου, ενώ στον πίνακα 4.4, φαίνεται η εμπλουτισμένη, σε σχέση με τα προηγούμενα συστήματα, βάση κανόνων, του ελεγκτή 0. Οι κανόνες ακολουθούν την ίδια λογική δομή με τα προηγούμενα συστήματα.



Σχήμα 4.19 Είσοδος «σφάλμα κατεύθυνσης» συστήματος «fuzzy0»



Σχήμα 4.20 Έξοδος «ταχύτητα αριστερού τροχού» ασαφούς ελεγκτή «fuzzy0»



Σχήμα 4.21 Έξοδος «ταχύτητα δεξιού τροχού» ασαφούς ελεγκτή «fuzzy0»

	ΛΕΚΤΙΚΟΙ ΚΑΝΟΝΕΣ			
	<u>Σφάλμα</u> <u>Κατεύθυνσης</u>		<u>Ταχύτητα</u> <u>Αριστερού</u> <u>Τροχού</u>	<u>Ταχύτητα</u> <u>Δεξιού</u> <u>Τροχού</u>
<b>Εάν</b>	Αριστερά	<b>Τότε</b>	Αργά	Μέτρια
	Δεξιά		Μέτρια	Αργά
	Κεντρικά		Γρήγορα	Γρήγορα
	Πολύ αριστερά		Αργά	Πολύ γρήγορα
	Πολύ δεξιά		Γρήγορα	Αργά

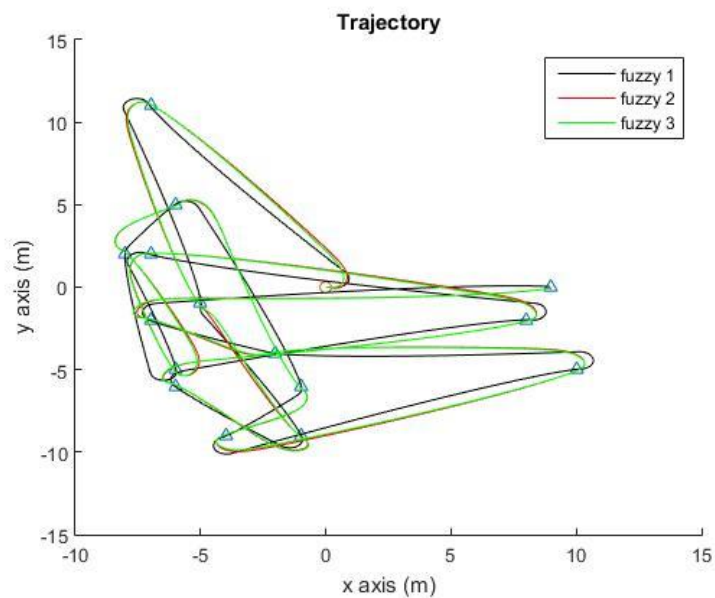
Πίνακας 4.4 Λεκτικοί κανόνες συστήματος «fuzzy0»

#### 4.3.2 ΠΡΟΣΟΜΟΙΩΣΗ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΑΣΑΦΩΝ ΕΛΕΓΚΤΩΝ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΧΩΡΙΣ ΕΜΠΟΔΙΑ

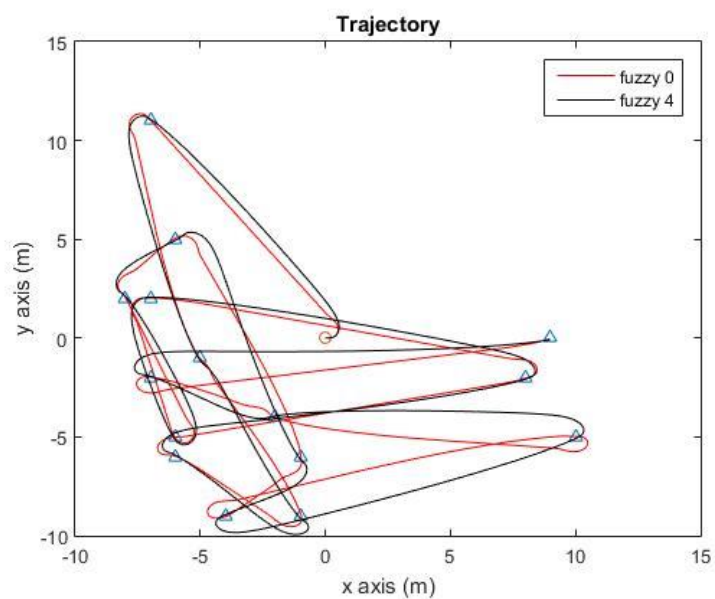
Στην προηγούμενη παράγραφο, παρουσιάστηκε η δομή των ασαφών ελεγκτών που δημιουργήθηκαν για πλοήγηση οχήματος σε άδεια арένα, πληρότητας 0% (σχήμα 4.1). Προκειμένου να αξιολογηθεί η συμπεριφορά και η απόδοση τους, το ρομποτικό όχημα, φέροντας τον αντίστοιχο ελεγκτή κάθε φορά, κλήθηκε να περάσει από 15 ίδια σημεία στο χώρο. Ως μεταβλητές αξιολόγησης ορίστηκαν ο συνολικός χρόνος εκτέλεσης του σεναρίου και η συνολική απόσταση που διένυσε το εκάστοτε όχημα. Τα αποτελέσματα των προσομοιώσεων παρουσιάζονται στον πίνακα 4.5. Ακόμη, στα σχήματα 4.22, 4.23, 4.24, 4.25, απεικονίζεται η τροχιά και η μέση ταχύτητα του οχήματος για τους ελεγκτές 1,2,3 και 4,5 ξεχωριστά.

Ασαφής ελεγκτής	Χρόνος εκτέλεσης (s)	Συνολική διανυόμενη απόσταση (m)
0	56.13	166.84
1	112.44	180.21
2	76.58	180.37
3	70.14	180.76
4	69.66	168.12

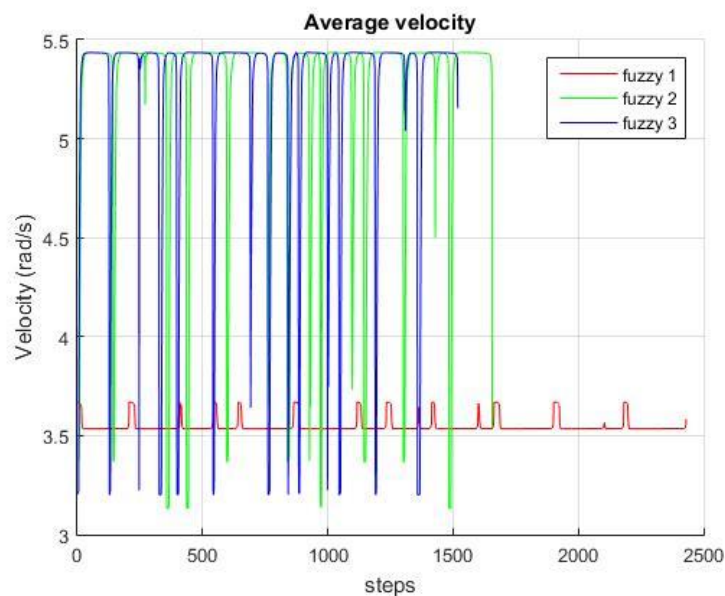
Πίνακας 4.5 Αποτελέσματα προσομοιώσεων ασαφών ελεγκτών σε арένα βαθμού πολυπλοκότητας «πολύ εύκολος»



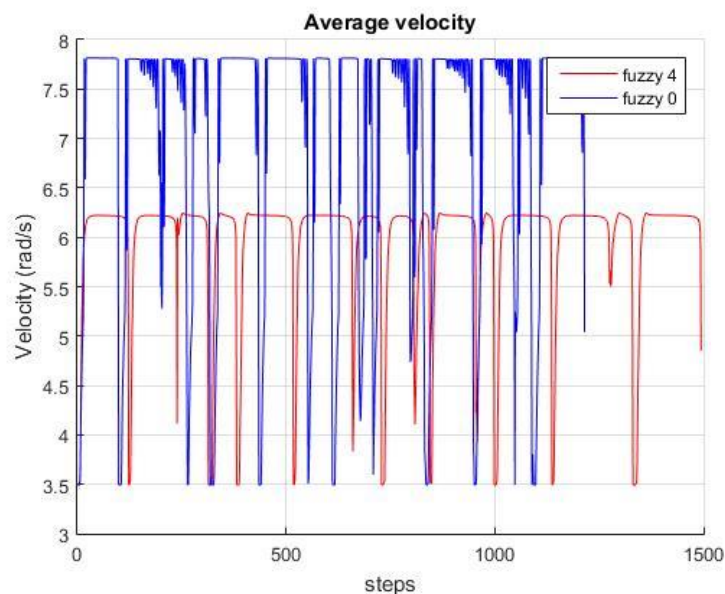
Σχήμα 4.22 Τροχιά οχήματος για τους ασαφείς ελεγκτές 1,2,3 σε αρένα πληρότητας 0%



Σχήμα 4.23 Τροχιά οχήματος για τους ασαφείς ελεγκτές 0 και 4 σε αρένα πληρότητας 0%



Σχήμα 4.24 Μέση ταχύτητα τροχών οχήματος για τους ασαφείς ελεγκτές 1,2,3 σε αρένα πληρότητας 0%



Σχήμα 4.25 Μέση ταχύτητα τροχών οχήματος για τους ασαφείς ελεγκτές 0 και 4 σε αρένα πληρότητας 0%

Παρατηρώντας τα αποτελέσματα χρόνου και συνολικής διαδρομής, είναι εμφανές ότι ο ελεγκτής fuzzy1, είναι ο λιγότερο αποδοτικός ελεγκτής, ως προς το χρόνο εκτέλεσης του σεναρίου, με χρόνο εκτέλεσης περίπου διπλάσιο από τους υπόλοιπους ελεγκτές. Λαμβάνοντας υπόψη και τα διαγράμματα τροχιάς και μέσης ταχύτητας, παρότι έχει μειωμένη πληροφορία στην είσοδο, φαίνεται να κινείται πιο άμεσα από στόχο σε στόχο, ωστόσο στρίβει με πολύ πιο αργό ρυθμό από τα υπόλοιπα συστήματα, ενώ και οι ταχύτητες

που αναπτύσσει είναι αρκετά χαμηλές σε σχέση με τους άλλους ελεγκτές. Γι' αυτό εν τέλει εκτελεί την ίδια περίπου απόσταση με τους ελεγκτές 2 και 3, αλλά σε πολύ πιο αργό χρόνο.

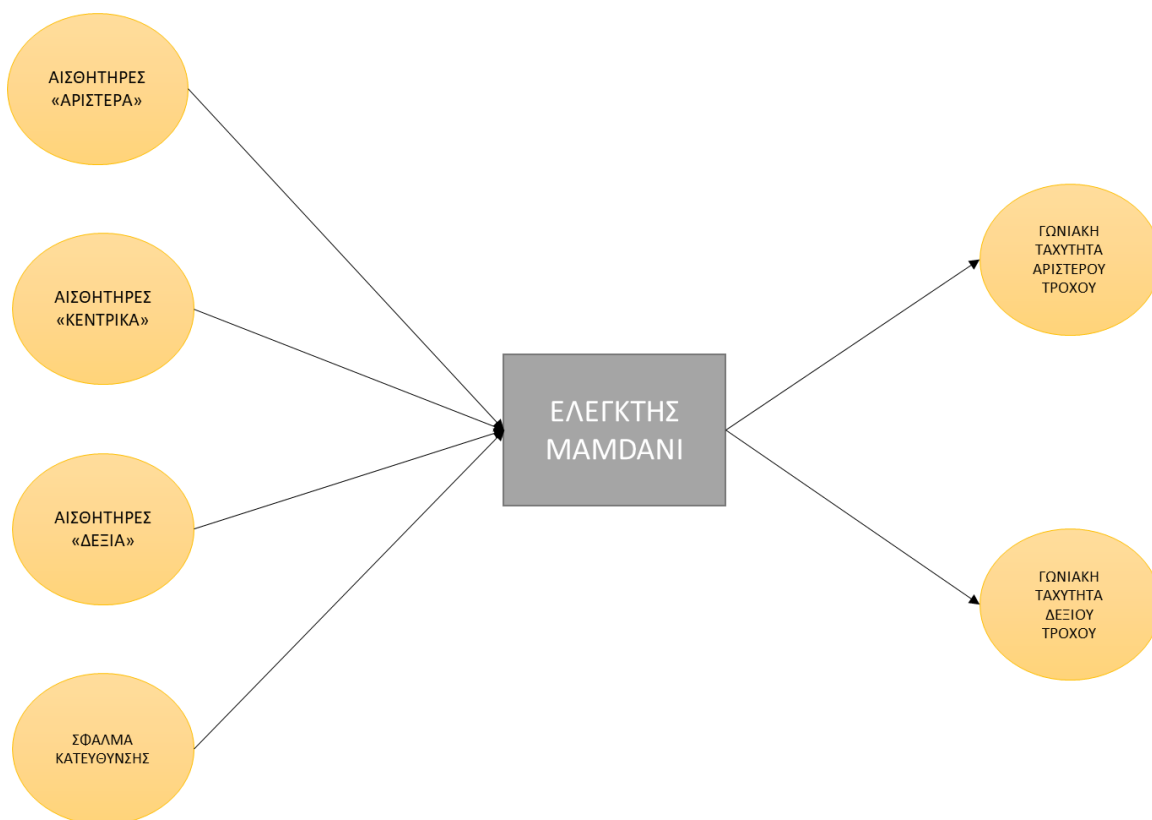
Οι ελεγκτές 2, 3 και 4 έχουν παρόμοια συμπεριφορά, αν θεωρήσουμε ότι τα αποτελέσματα τους δεν έχουν πολύ μεγάλη απόκλιση. Η τροχιά τους είναι ομαλή και γι' αυτό η τιμή της συνολικής απόστασης που διανύουν είναι σχεδόν ίδια, με τον ελεγκτή 4 να είναι πιο αποδοτικός σε αυτό το κομμάτι. Οι μικρές αλλαγές στην είσοδο και έξοδο του συστήματος 3 σε σχέση με αυτή του 2, φαίνεται να έφεραν αισθητή αλλαγή στο χρόνο εκτέλεσης του σεναρίου από τον ελεγκτή 3. Όπως αποδεικνύεται στο σχήμα 4.24, οι αλλαγές σταθεροποίησαν σε αρκετές περιπτώσεις τις ανά διαστήματα σημαντικές μεταπτώσεις της μέσης ταχύτητας του ελεγκτή 2.

Ο ελεγκτής 4, όπως έχει αναφερθεί, διαφέρει με τον ελεγκτή 3 ως προς τις συναρτήσεις εξόδου. Παρατηρώντας τα αποτελέσματα, βλέπει κανείς ότι δεν επέρχεται σημαντική βελτίωση στο χρόνο εκτέλεσης σε σχέση με τον ελεγκτή 3, ωστόσο, η αλλαγή στις εξόδους οδηγεί το όχημα σε αύξηση των τιμών της μέσης ταχύτητας του. Το γεγονός αυτό φαίνεται να τον βοηθάει στην επίτευξη γρήγορων στροφών στο χώρο, με αποτέλεσμα να διανύσει μικρότερη απόσταση από τους ελεγκτές 2 και 3.

Πιο αποδοτικός ελεγκτής από τους πέντε χαρακτηρίζεται ο fuzzy 0, καθώς είναι ο πιο γρήγορος και αυτός με τη μικρότερη διανυόμενη απόσταση. Η προσθήκη επιπρόσθετης πληροφορίας στην είσοδο, οδήγησε σε πιο απότομη και μη ομαλή συμπεριφορά από κόμβο σε κόμβο, σε ότι αφορά την τροχιά, αλλά ταυτόχρονα στην ανάπτυξη πιο υψηλών ταχυτήτων από κάθε άλλο ελεγκτή. Έτσι η απόδοση του θεωρείται η ζητούμενη, σε σύγκριση με όλους τους άλλους ελεγκτές.

### **4.3.3 ΚΑΤΑΣΚΕΥΗ ΑΣΑΦΩΝ ΕΛΕΓΚΤΩΝ ΓΙΑ ΚΙΝΗΣΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΜΕ ΕΜΠΟΔΙΑ**

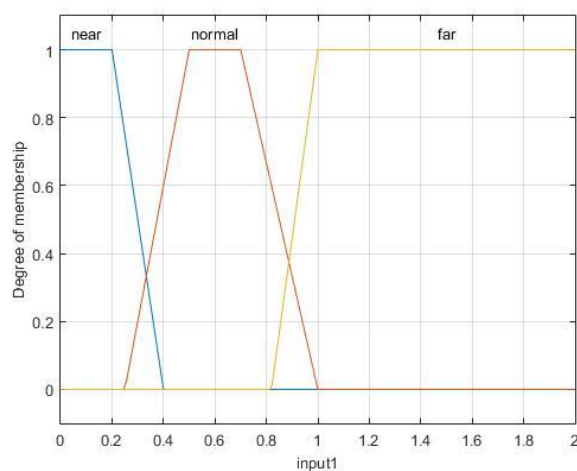
Για την ανάπτυξη ελεγκτών, οι οποίοι να ανταπεξέρχονται σε περιβάλλοντα με σταθερά εμπόδια, χρησιμοποιούνται δεδομένα τα οποία ανιχνεύονται από τους οχτώ αισθητήρες του οχήματος, όπως αναλύονται στο τρίτο κεφάλαιο. Σε ότι αφορά την κατασκευή, επιλέχθηκαν, αρχικά, οι τρεις από τους πέντε καλύτερους ελεγκτές παραπάνω -fuzzy0, fuzzy3, fuzzy4- ως προς το συνδυασμό μικρότερου χρόνου και απόστασης. Στην είσοδο του ασαφούς συστήματος του καθενός, προστέθηκαν, τρεις παράμετροι, οι οποίοι αντιπροσωπεύουν τους οχτώ αισθητήρες, σε τρεις ομάδες, όπως περιγράφεται στην ενότητα 3.7. Η δομή ενός ασαφούς ελεγκτή για κίνηση σε περιβάλλον με εμπόδια παρουσιάζεται στο σχήμα 4.26.



Σχήμα 4.26 Δομή ασαφούς ελεγκτή σε περιβάλλον με εμπόδια

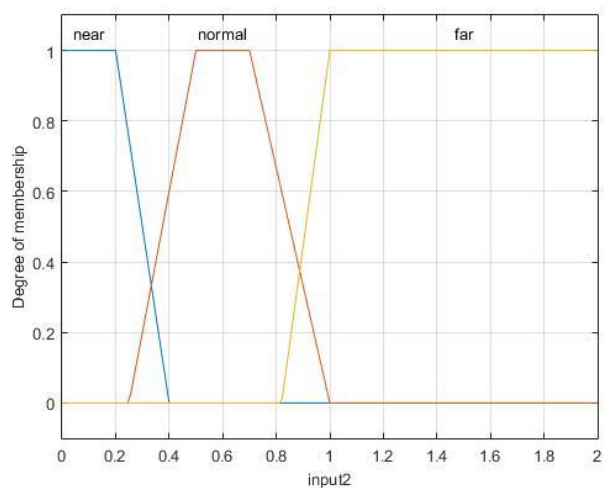
### Ασαφής Ελεγκτής 3 (Fuzzy 3)

Στα σχήματα 4.27 – 4.32 παρουσιάζονται τα ασαφή σύνολα των παραμέτρων εισόδου και εξόδου, του ασαφούς ελεγκτή 3, για περιβάλλον με εμπόδια. Η δομή είναι ίδια με τον ασαφή ελεγκτή 3 της παραγράφου 4.3.1, με μόνη διαφορά την προσθήκη των παραμέτρων των αισθητήρων και των αντίστοιχων συνόλων.

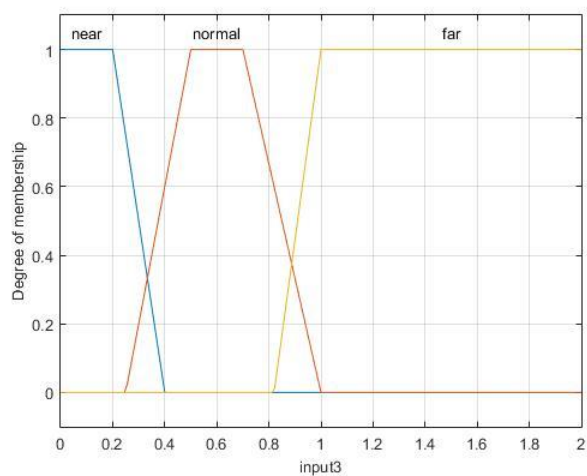


Σχήμα 4.27 Είσοδος αισθητήρων «αριστερά» συστήματος «fuzzy3»

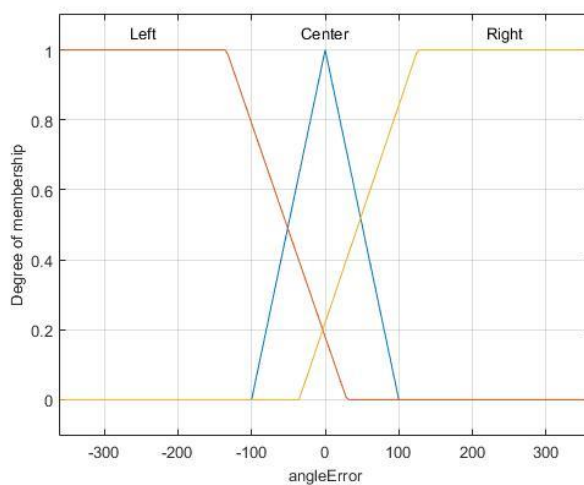




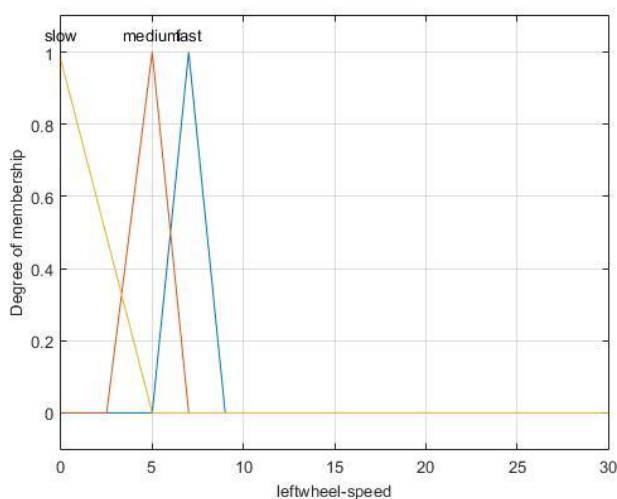
Σχήμα 4.28 Είσοδος αισθητήρων «κεντρικά» συστήματος «fuzzy3»



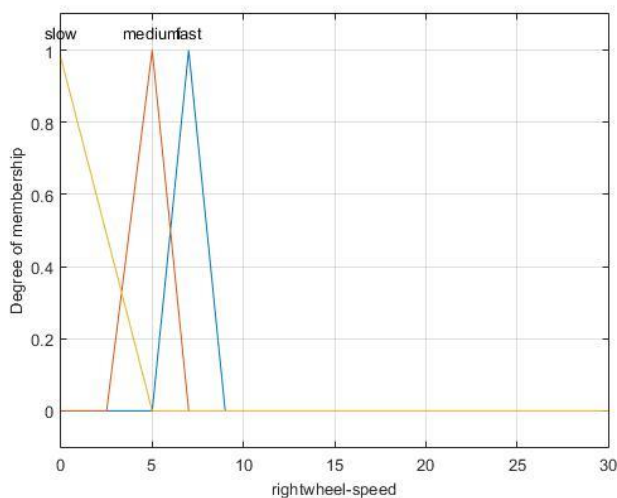
Σχήμα 4.29 Είσοδος αισθητήρων «δεξιά» συστήματος «fuzzy3»



Σχήμα 4.30 Είσοδος «σφάλμα κατεύθυνσης» συστήματος «fuzzy3»



Σχήμα 4.31 Έξοδος «αριστερή ταχύτητα τροχού» συστήματος «fuzzy3»



Σχήμα 4.32 Έξοδος «ταχύτητα δεξιού τροχού» ασαφούς ελεγκτή «fuzzy3»

Σύμφωνα με τα παραπάνω σύνολα εισόδου -εξόδου, οι κανόνες που εισάγονται στο σύστημα και είναι απαραίτητοι για να καλύψουν κάθε πιθανό συνδυασμό συνόλων - συναρτήσεων συμμετοχής και κατ' επέκταση συνθηκών, είναι 32. Η δομή των κανόνων ακολουθεί την εξής λογική:

ΕΑΝ...ΣΥΝΘΗΚΗ 1 ΚΑΙ ΣΥΝΘΗΚΗ 2 ΚΑΙ ΣΥΝΘΗΚΗ 3...ΤΟΤΕ... ΕΝΤΟΛΗ 1, ΕΝΤΟΛΗ 2

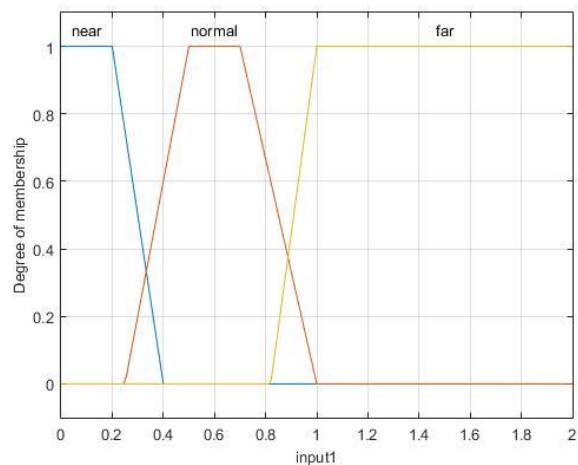
Ενδεικτικά αναφέρονται κάποιοι από τους κανόνες στον παρακάτω πίνακα.

ΛΕΚΤΙΚΟΙ ΚΑΝΟΝΕΣ							
	<u>Ομάδα</u> <u>αισθητήρων</u> <u>«αριστερά»</u>	<u>Ομάδα</u> <u>αισθητήρων</u> <u>«κεντρικά»</u>	<u>Ομάδα</u> <u>αισθητήρων</u> <u>«δεξιά»</u>	<u>Σφάλμα</u> <u>Κατεύθυν</u> <u>σης</u>		<u>Ταχύτητα</u> <u>Αριστερού</u> <u>Τροχού</u>	<u>Ταχύτητα</u> <u>Δεξιού</u> <u>Τροχού</u>
<b>Εάν</b>	Μακριά	Μακριά	Μακριά	-	<b>Τότε</b>	Μέτρια	Γρήγορα
	Μακριά	Μέτρια	Μακριά	-		Αργά	Γρήγορα
	Μακριά	Μέτρια	Μέτρια	-		Μέτρια	Αργά
	Μέτρια	Μέτρια	Μακριά	-		Γρήγορα	Αργά
	Μέτρια	Μακριά	Μακριά	-		Μέτρια	Αργά
	Κοντά	Μέτρια	Κοντά	-		Αργά	Μέτρια
	Μέτρια	Κοντά	Μέτρια	-		Μέτρια	Αργά
	Μέτρια	Μακριά	Κοντά	-		Αργά	Αργά
	Μέτρια	Μέτρια	Μέτρια	-		Αργά	Αργά
	Μακριά	Μακριά	Μακριά	Κεντρικά		Γρήγορα	Γρήγορα
	Μακριά	Μακριά	Μακριά	Δεξιά		Γρήγορα	Αργά
	Μακριά	Μακριά	Μακριά	Αριστερά		Αργά	Γρήγορα

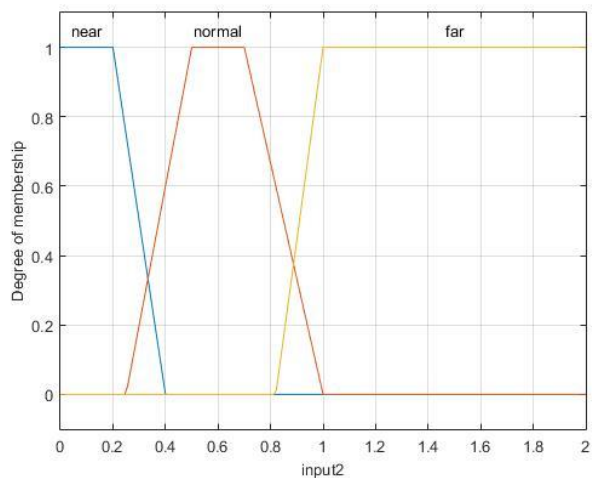
Πίνακας 4.6 Λεκτικοί κανόνες συστήματος «fuzzy 3» με εμπόδια

**Ασαφής Ελεγκτής 4 (Fuzzy 4)**

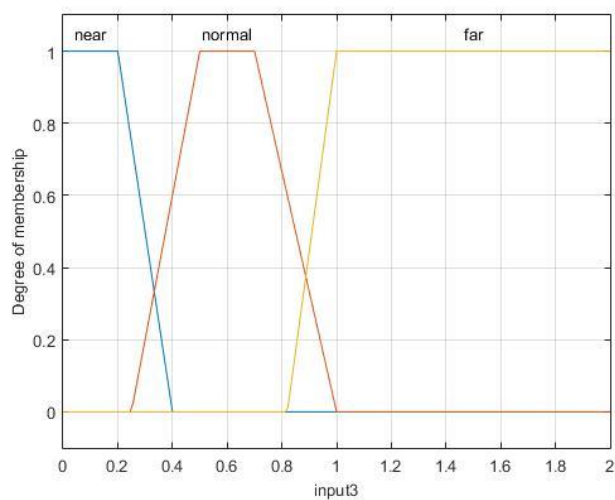
Στα σχήματα 4.33 – 4.38 παρουσιάζονται τα ασαφή σύνολα των παραμέτρων εισόδου και εξόδου, του ασαφούς ελεγκτή 4, για περιβάλλον με εμπόδια. Η δομή είναι ίδια με τον ασαφή ελεγκτή 4 της παραγράφου 4.3.1, με μόνη διαφορά την προσθήκη των παραμέτρων των αισθητήρων και των αντίστοιχων συνόλων.



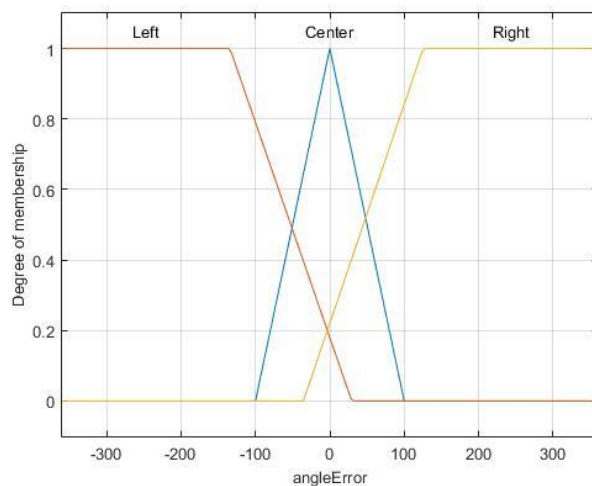
Σχήμα 4.33 Είσοδος αισθητήρων «αριστερά» συστήματος «fuzzy4»



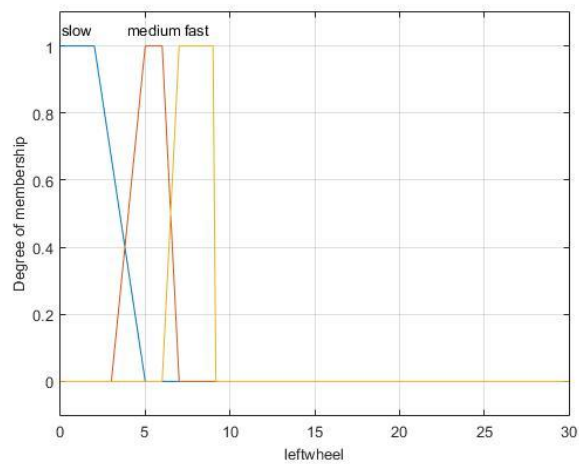
Σχήμα 4.34 Είσοδος αισθητήρων «κεντρικά» συστήματος «fuzzy4»



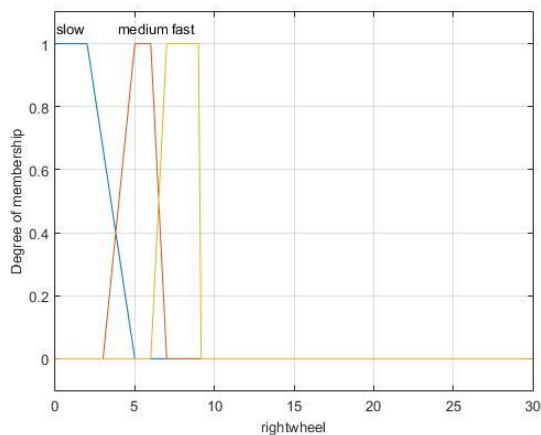
Σχήμα 4.35 Είσοδος αισθητήρων «δεξιά» συστήματος «fuzzy4»



Σχήμα 4.36 Είσοδος «σφάλμα κατεύθυνσης» συστήματος «fuzzy4»



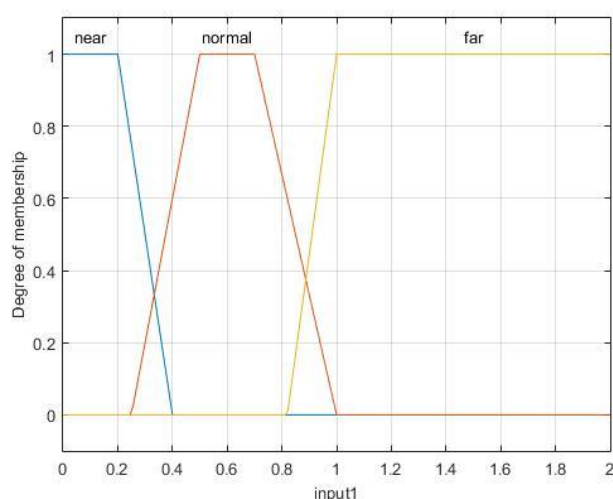
Σχήμα 4.37 Εξοδος «ταχύτητα αριστερού τροχού» ασαφούς ελεγκτή «fuzzy4»



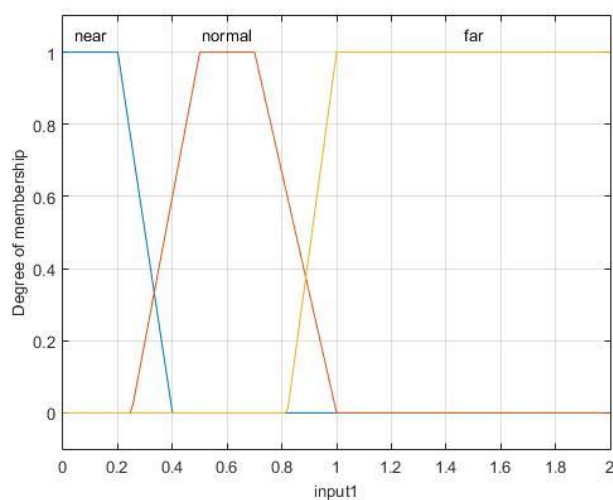
Σχήμα 4.38 Εξοδος «ταχύτητα δεξιού τροχού» ασαφούς ελεγκτή «fuzzy4»

### Ασαφής Ελεγκτής 0 (Fuzzy 0)

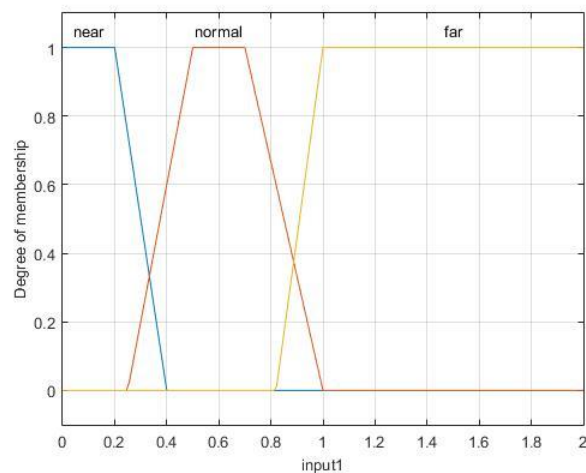
Στα σχήματα 4.39 – 4.42 παρουσιάζονται τα ασαφή σύνολα των παραμέτρων της εισόδου, του ασαφούς ελεγκτή 0, για περιβάλλον με εμπόδια. Η δομή είναι ίδια με τον ασαφή ελεγκτή 0 της παραγράφου 4.3.1, με μόνη διαφορά την προσθήκη των παραμέτρων των αισθητήρων και των αντίστοιχων συνόλων. Σε ότι αφορά την έξοδο, παρότι αρχικός σκοπός ήταν η διατήρηση της ως έχει, παρατηρήθηκε μετέπειτα, ότι η πληροφορία για τις ταχύτητες των τροχών δεν επαρκούσε για την ομαλή πορεία του οχήματος στο χώρο και την επίτευξη των σεναρίων, συνεπώς εισήχθη τέταρτο σύνολο («μηδέν») στην έξοδο, προσθέτοντας πληροφορία και κανόνες στο σύστημα. Οι αλλαγές αυτές απεικονίζονται στα σχήματα 4.43 και 4.44.



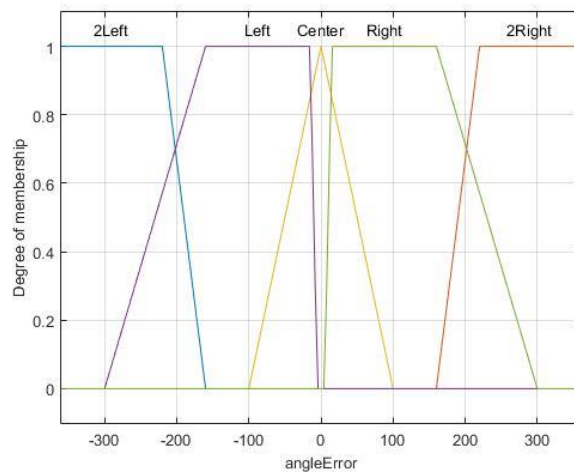
Σχήμα 4.39 Είσοδος αισθητήρων «αριστερά» συστήματος «fuzzy0»



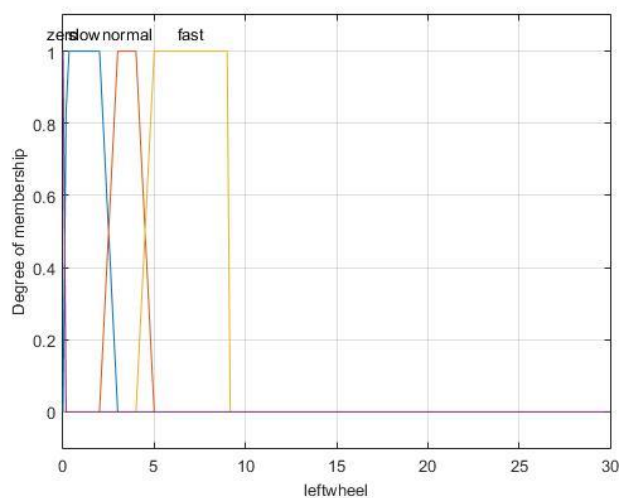
Σχήμα 4.40 Είσοδος αισθητήρων «κεντρικά» συστήματος «fuzzy0»



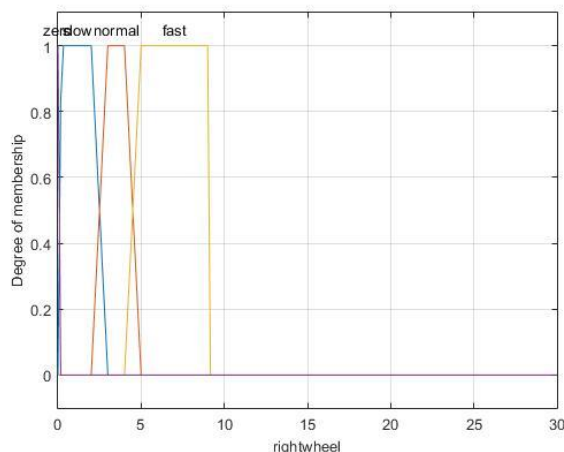
Σχήμα 4.41 Είσοδος αισθητήρων «δεξιά» συστήματος «fuzzy0»



Σχήμα 4.42 Είσοδος «σφάλμα κατεύθυνσης» συστήματος «fuzzy0»



Σχήμα 4.43 Εξοδος «ταχύτητα αριστερού τροχού» ασαφούς ελεγκτή «fuzzy0»



Σχήμα 4.44 Έξοδος «ταχύτητα δεξιού τροχού» ασαφούς ελεγκτή «fuzzy4»

Σύμφωνα με τις παραπάνω παραμέτρους, οι κανόνες που εισάγονται στο σύστημα και είναι απαραίτητοι για να καλύψουν κάθε πιθανό συνδυασμό συνόλων - συναρτήσεων και κατ' επέκταση συνθηκών, είναι 32. Παρουσιάζονται ενδεικτικά στον πίνακα 4.7 κάποιοι από αυτούς, ενώ αναγράφονται και κανόνες που περιλαμβάνουν τις προσθήκες που έγιναν στην έξοδο του συστήματος 0. Η δομή στη λογική των κανόνων είναι η ίδια με τα προηγούμενα συστήματα της παραγράφου.

ΛΕΚΤΙΚΟΙ ΚΑΝΟΝΕΣ							
	<u>Είσοδος</u> <u>Αισθητήρων</u> <u>1</u>	<u>Είσοδος</u> <u>Αισθητήρων</u> <u>2</u>	<u>Είσοδος</u> <u>Αισθητήρων</u> <u>3</u>	<u>Σφάλμα</u> <u>Κατεύθυνσης</u>		<u>Ταχύτητα</u> <u>Αριστερού</u> <u>Τροχού</u>	<u>Ταχύτητα</u> <u>Δεξιού</u> <u>Τροχού</u>
<b>Εάν</b>	Μακριά	Μακριά	Μακριά	-	<b>Τότε</b>	Μέτρια	Γρήγορα
	Μακριά	Μέτρια	Μακριά	-		Αργά	Γρήγορα
	Μακριά	Μέτρια	Μέτρια	-		Μέτρια	Αργά
	Μέτρια	Μέτρια	Μακριά	-		Γρήγορα	Αργά
	Μέτρια	Μακριά	Μακριά	-		Μέτρια	Αργά
	Κοντά	Μέτρια	Κοντά	-		Μηδέν	Μέτρια
	Μέτρια	Κοντά	Μέτρια	-		Μέτρια	Μηδέν
	Μέτρια	Μακριά	Μακριά	Πολύ αριστερά		Μηδέν	Γρήγορα
	Μέτρια	Μέτρια	Μέτρια	Πολύ δεξιά		Γρήγορα	Μηδέν
	Μακριά	Μακριά	Μακριά	Κεντρικά		Γρήγορα	Γρήγορα
	Μακριά	Μακριά	Μακριά	Δεξιά		Γρήγορα	Αργά
	Μακριά	Μακριά	Μακριά	Αριστερά		Αργά	Γρήγορα

Πίνακας 4.7 Λεκτικοί κανόνες συστήματος «fuzzy 0» με εμπόδια



#### 4.3.4 ΠΡΟΣΟΜΟΙΩΣΗ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΑΣΑΦΩΝ ΕΛΕΓΚΤΩΝ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΜΕ ΕΜΠΟΔΙΑ

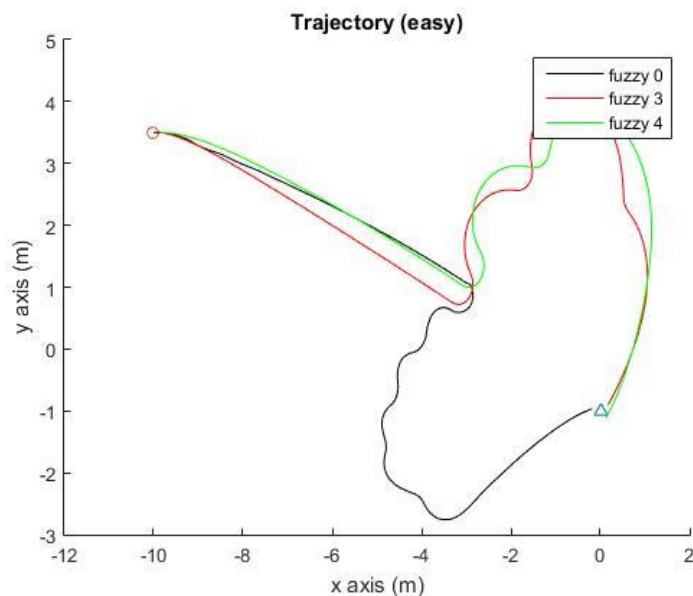
Στην προηγούμενη παράγραφο, παρουσιάστηκε η δομή των ασαφών ελεγκτών που δημιουργήθηκαν για πλοήγηση οχήματος σε αρένες με σταθερά εμπόδια. Η συμπεριφορά και η απόδοση του κάθε ελεγκτή, αξιολογείται στις αρένες πληρότητας 8%, 25%, 58% και 83% (σχήμα 4.2 – 4.5 αντίστοιχα). Φέροντας τον αντίστοιχο ελεγκτή κάθε φορά, το ρομποτικό όχημα κλήθηκε να περάσει από μία συγκεκριμένη ακολουθία σημείων, η οποία διαφέρει ανάλογα με την αρένα, είναι ίδια όμως και για τους τρεις ελεγκτές σε κάθε περίπτωση. Ως μεταβλητές αξιολόγησης ορίστηκαν ο συνολικός χρόνος εκτέλεσης του σεναρίου και η συνολική απόσταση που διένυσε το εκάστοτε όχημα. Ως σημείο εκκίνησης ορίζεται το  $[x_1, y_1] = [-10, 3.5]$ .

##### Αρένα βαθμού «εύκολος»

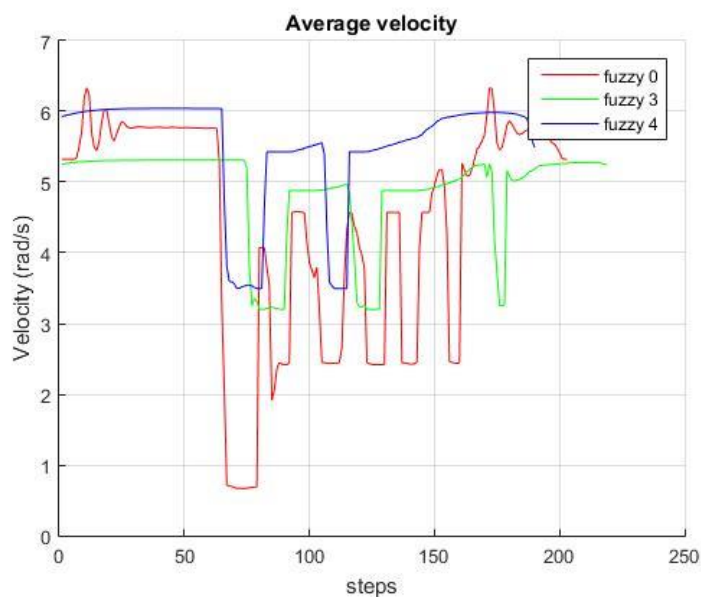
Η αρένα αυτή έχει το χαρακτήρα της επιβεβαίωσης του σεναρίου, για τους τρεις ελεγκτές και δε χρησιμοποιείται για λεπτομερή σύγκριση μεταξύ τους. Ωστόσο, για να υπάρχει πλήρης εικόνα των ελεγκτών, παρατίθενται τα αποτελέσματα σε αυτήν στον πίνακα 4.8. Στα σχήματα 4.45 και 4.46, δίνονται τα γραφήματα τροχιάς και μέσης ταχύτητας του οχήματος για τους τρεις ελεγκτές, στην αρένα αυτή. Σημείο στόχο αποτελεί το σημείο  $[x_2, y_2] = [0, -1]$ .

Ασαφής ελεγκτής	Χρόνος εκτέλεσης (s)	Συνολική διανυόμενη απόσταση (m)
Fuzzy0	9.36	17.53
Fuzzy3	10.15	27.08
Fuzzy4	8.80	26.49

Πίνακας 4.8 Αποτελέσματα προσομοιώσεων ασαφών ελεγκτών σε αρένα βαθμού πολυπλοκότητας «εύκολος»



Σχήμα 4.45 Τροχιά οχήματος για τους ασαφείς ελεγκτές 0,3,4 σε αρένα πληρότητας 8%



Σχήμα 4.46 Μέση ταχύτητα τροχών για τους ασαφείς ελεγκτές 0,3,4 σε αρένα πληρότητας 8%

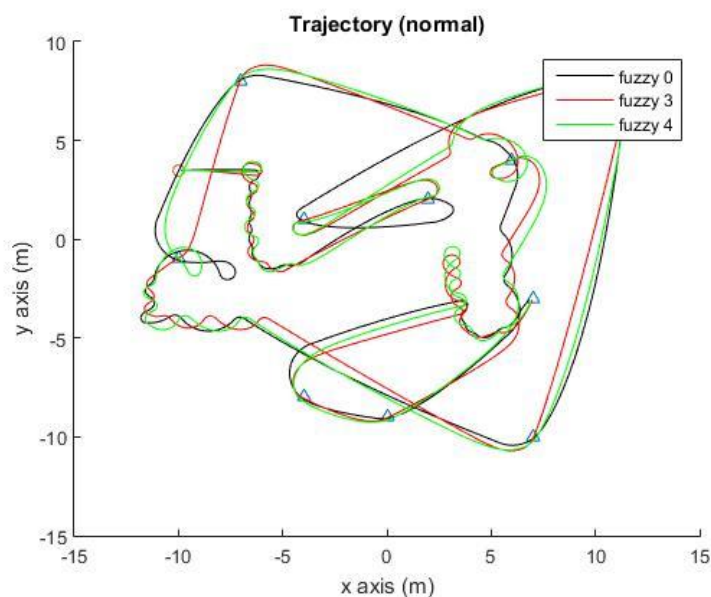
Αποδεικνύεται ότι το ρομποτικό όχημα αποφεύγει το εμπόδιο και φθάνει στον τελικό στόχο και με τους τρεις ελεγκτές. Και οι τρεις είναι ικανοί να χρησιμοποιηθούν για το σκοπό που δημιουργήθηκαν. Η πρώτη εντύπωση που δίνεται από τα αποτελέσματα του πίνακα 4.8 και τα σχήματα τροχιάς και μέσης ταχύτητας, είναι ότι ο ελεγκτής 4 είναι ο πιο αποδοτικός σε ότι αφορά το χρόνο εκτέλεσης, αλλά ο ελεγκτής 0 είναι αυτός που κινείται στο χώρο διανύοντας τη μικρότερη απόσταση. Ο ελεγκτής 4 εμφανίζει υψηλότερες ταχύτητες και μικρότερες ταλαντώσεις σε αυτές, σε σχέση με τον ελεγκτή 0.

### Αρένα βαθμού «μέτριας δυσκολίας»

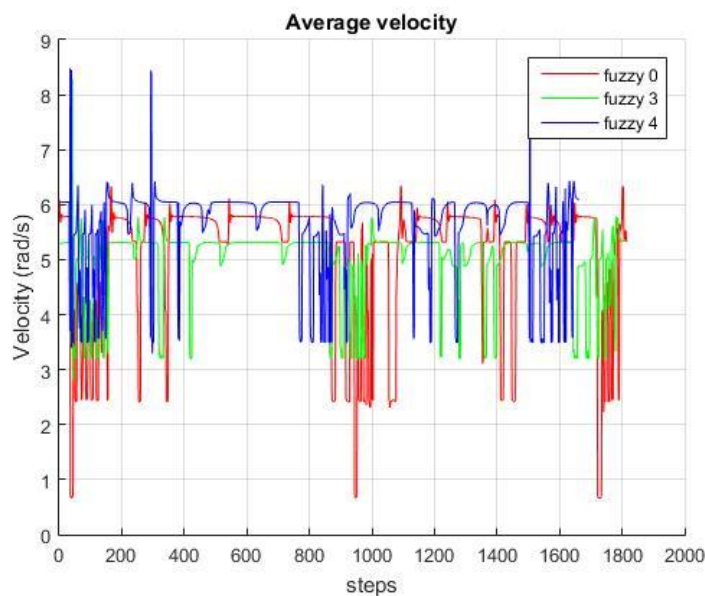
Στον πίνακα 4.9, παρουσιάζονται τα αποτελέσματα των προσομοιώσεων για την αρένα πληρότητας 25%. Στα σχήματα 4.47 και 4.48, δίνονται τα γραφήματα τροχιάς και μέσης ταχύτητας του οχήματος για τους τρεις ελεγκτές, στην αρένα αυτή.

Ασαφής ελεγκτής	Χρόνος εκτέλεσης (s)	Συνολική διανυόμενη απόσταση (m)
Fuzzy0	84.58	163.29
Fuzzy3	83.69	178.77
Fuzzy4	77.04	178.49

Πίνακας 4.9 Αποτελέσματα προσομοιώσεων ασαφών ελεγκτών σε αρένα βαθμού πολυπλοκότητας «μέτριας δυσκολίας»



Σχήμα 4.47 Τροχιά οχήματος για τους ασαφείς ελεγκτές 0,3,4 σε αρένα πληρότητας 25%



Σχήμα 4.48 Μέση ταχύτητα τροχών για τους ασαφείς ελεγκτές 0,3,4 σε αρένα πληρότητας 25%

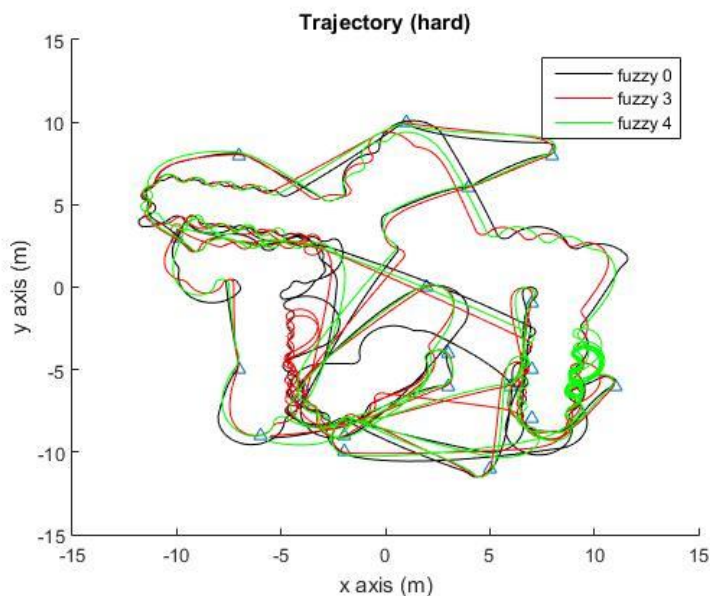
Με τη βοήθεια των παραπάνω αποτελεσμάτων, συμπεραίνει κανείς ότι η συμπεριφορά που παρουσίασαν οι ελεγκτές στην αρένα πληρότητας 8%, είναι η ίδια και σε αρένα μεγαλύτερης πληρότητας. Ο ελεγκτής 4 είναι ο πιο γρήγορος αλλά ο ελεγκτής 0 διανύει τη μικρότερη απόσταση. Ο ελεγκτής 4 κατέγραψε υψηλή ταχύτητα με λιγότερες ταλαντώσεις σε σχέση με τον ελεγκτή 0. Αντιθέτως, ο τελευταίος οδηγεί το όχημα σε ομαλή τροχιά με «έξυπνες» επιλογές αποφυγής περιττών σημείων στο χώρο.

#### Αρένα βαθμού «δύσκολος»

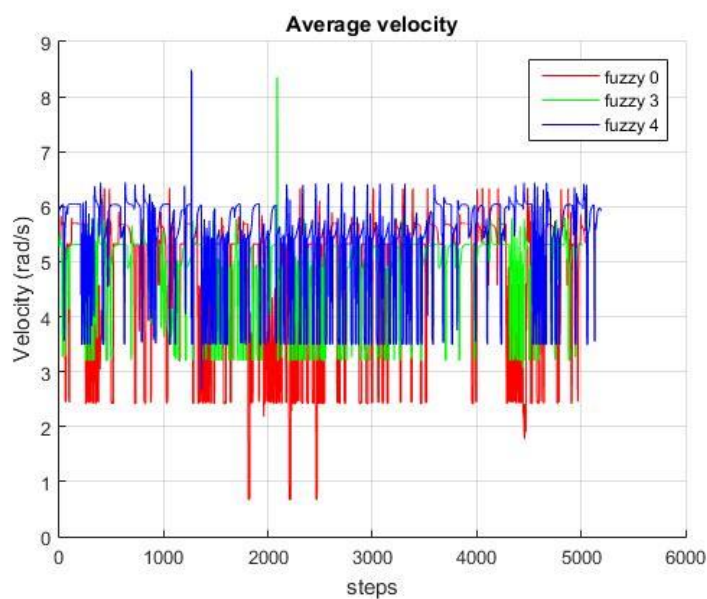
Στον πίνακα 4.10, παρουσιάζονται τα αποτελέσματα των προσομοιώσεων για την αρένα πληρότητας 58%. Στα σχήματα 4.49 και 4.50, δίνονται τα γραφήματα τροχιάς και μέσης ταχύτητας του οχήματος για τους τρεις ελεγκτές, στην αρένα αυτή.

Ασαφής ελεγκτής	Χρόνος εκτέλεσης (s)	Συνολική διανυόμενη απόσταση (m)
Fuzzy0	236.32	355.28
Fuzzy3	236.09	362.70
Fuzzy4	243.95	418.27

Πίνακας 4.10 Αποτελέσματα προσομοιώσεων ασαφών ελεγκτών σε αρένα βαθμού πολυπλοκότητας «δύσκολος»



Σχήμα 4.49 Τροχιά οχήματος για τους ασαφείς ελεγκτές 0,3,4 σε αρένα πληρότητας 58%



Σχήμα 4.50 Μέση ταχύτητα τροχών για τους ασαφείς ελεγκτές 0,3,4 σε αρένα πληρότητας 58%

Σε αυτό το σενάριο αυξημένης πληρότητας εμποδίων - υψηλής δυσκολίας, οι ελεγκτές 3 και 4 φαίνεται να τα καταφέρνουν καλύτερα σε σχέση με τον ελεγκτή 0. Εκτέλεσαν το σενάριο στον ίδιο χρόνο, ωστόσο ο ελεγκτής 0 ήταν ο πιο αποδοτικός, αφού διένυσε μικρότερη απόσταση. Η υψηλή αντίληψη του σε ότι αφορά τη διαδρομή που πρέπει να

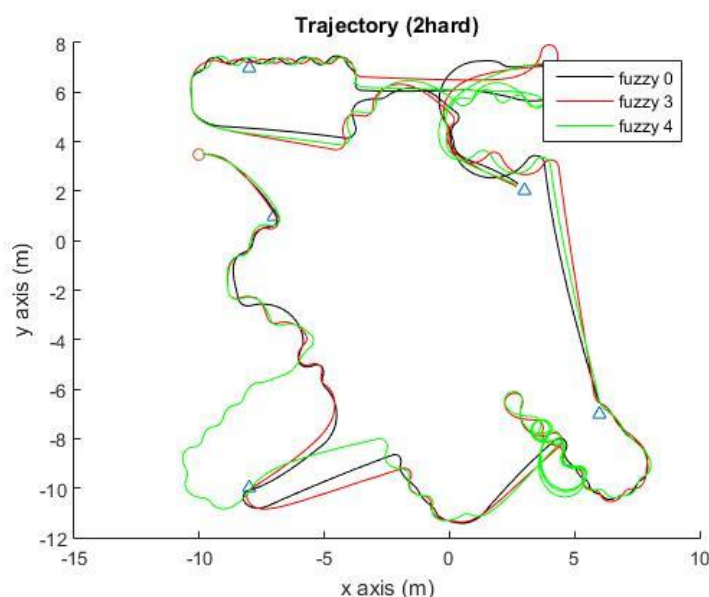
ακολουθήσει, η οποία οφείλεται πιθανότατα στην αυξημένη πληροφορία της μεταβλητής εισόδου, επιβεβαιώνεται και σε αρένα υψηλής πολυπλοκότητας. Ο ελεγκτής 4 φαίνεται να δυσκολεύεται σε αρένα με πυκνή παρουσία εμποδίων, μπλέκοντας σε βρόχους που τον καθυστερούν, σε ότι αφορά την τροχιά του, παρότι στον τομέα της ταχύτητας, παρουσιάζει ελάχιστη αύξηση των ταλαντώσεων σε σχέση με τα λιγότερο πολύπλοκα σενάρια. Ο ελεγκτής 3, παρότι έχει παρόμοια δομή με τον 4, δεν μπλέκει σε βρόχους που να τον καθυστερούν σημαντικά, πιθανότατα λόγω της πιο απλής πληροφορίας στις παραμέτρους εξόδου.

#### Αρένα βαθμού «πολύ δύσκολος»

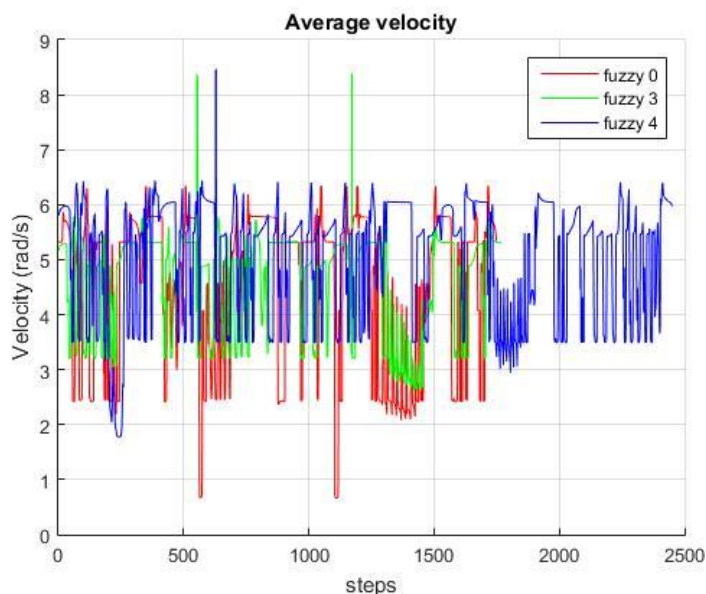
Στον πίνακα 4.11, παρουσιάζονται τα αποτελέσματα των προσομοιώσεων για την αρένα πληρότητας 83%. Στα σχήματα 4.51 και 4.52, δίνονται τα γραφήματα τροχιάς και μέσης ταχύτητας του οχήματος για τους τρεις ελεγκτές, στην αρένα αυτή.

Ασαφής ελεγκτής	Χρόνος εκτέλεσης (s)	Συνολική διανυόμενη απόσταση (m)
Fuzzy0	84.05	120.89
Fuzzy3	84.79	125.04
Fuzzy4	118.03	183.64

Πίνακας 4.11 Αποτελέσματα προσομοιώσεων ασαφών ελεγκτών σε αρένα βαθμού πολυπλοκότητας «δύσκολος»



Σχήμα 4.51 Τροχιά οχήματος για τους ασαφείς ελεγκτές 0,3,4 σε αρένα πληρότητας 83%



Σχήμα 4.52 Μέση ταχύτητα τροχών για τους ασαφείς ελεγκτές 0,3,4 σε αρένα πληρότητας 83%

Οι ελεγκτές, στην αρένα υψηλότερης δυσκολίας, είχαν την ίδια συμπεριφορά με το προηγούμενο σενάριο πληρότητας 58%, συνεπώς δεν υπάρχουν καινούργια συμπεράσματα.

### Γενικά συμπεράσματα

Και οι τρεις ελεγκτές ανταποκρίνονται επαρκώς στις διάφορες αρένες δυσκολίας.

Για μια αρένα μικρής πληρότητας σε εμπόδια, η επιλογή ενός ελεγκτή τύπου fuzzy4, είναι ιδανική, σε σχέση με τους άλλους ελεγκτές, αν κριτήριο αποτελεί ο χρόνος εκτέλεσης του σεναρίου. Η δομή του σε είσοδο και έξοδο φαίνεται να του εξασφαλίζει υψηλές ταχύτητες και λιγότερο συχνές ταλαντώσεις σε αυτές, με μικρό εύρος μεταπτώσεων σε σχέση με τους άλλους ελεγκτές. Αν ωστόσο κριτήριο αποτελεί η συνολική διανυόμενη απόσταση, επιλογή αποτελεί ο ελεγκτής fuzzy0, ο οποίος έχει τόσο ομαλή τροχιά στην κίνηση του όσο και «έξυπνες» επιλογές σε πιο δύσκολα σημεία της αρένας. Η αυξημένη πληροφορία στην είσοδο του, φαίνεται να τον οδηγεί σε πιο βέλτιστες αποφάσεις σε σχέση με τη διαδρομή που ακολουθεί. Η μεγάλες μεταπτώσεις στην ταχύτητα του, ωστόσο, φαίνεται να του στερούν χρονική απόδοση.

Σε ότι αφορά τα περιβάλλοντα αυξημένης πληρότητας σε εμπόδια και συνεπώς δυσκολίας, ο ελεγκτής fuzzy 4 δεν μπορεί να ανταπεξέλθει επαρκώς, σε σχέση με τους

ελεγκτές fuzzy0 και fuzzy3, ως προς το κριτήριο του χρόνου αλλά και της διαδρομής. Τόσο ένας πιο σύνθετος ελεγκτής όσο και ένας πιο απλός φαίνεται ότι μπορούν να εκτελέσουν με παρόμοιο τρόπο το σενάριο. Ωστόσο, ο σύνθετος ελεγκτής fuzzy0 εξασφαλίζει, σε ένα πολύπλοκο περιβάλλον, μικρότερη συνολική διανυόμενη απόσταση, με μικρότερες ταλαντώσεις στην κίνηση γύρω από εμπόδια και πιο άμεση κίνηση από στόχο σε στόχο.

#### **4.4 ΚΑΤΑΣΚΕΥΗ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ – ΝΕΥΡΩΝΙΚΩΝ ΕΛΕΓΚΤΩΝ ΓΙΑ ΚΙΝΗΣΗ PIONEER 3-DX**

Ως συνέχεια της υλοποίησης και εφαρμογής ελεγκτών ασαφούς λογικής, δημιουργήθηκαν και αξιολογήθηκαν ελεγκτές, οι οποίοι βασίζονται σε νευρωνικά δίκτυα. Για την υλοποίηση των διάφορων δικτύων νευρώνων, χρησιμοποιήθηκαν καταγεγραμμένα δεδομένα προσομοίωσης των ασαφών ελεγκτών που παρουσιάστηκαν στην προηγούμενη ενότητα. Συγκεκριμένα, για τη δημιουργία νευρωνικών ελεγκτών, με σκοπό την πλοήγηση σε αρένα χωρίς εμπόδια, χρησιμοποιήθηκαν τα δεδομένα προσομοίωσης του πιο γρήγορου ασαφή ελεγκτή (fuzzy 0), ενώ για πλοήγηση σε περιβάλλον με εμπόδια, του καλύτερου σε συνδυασμό χρόνου – απόστασης, για την αρένα με δύσκολο βαθμό πολυπλοκότητας (πληρότητα 58%). Ο λόγος που επιλέχτηκε η αρένα πληρότητας 58%, ως η πηγή των δεδομένων εκπαίδευσης, είναι ότι, τα δεδομένα είναι σημαντικό να καλύπτουν ένα ικανοποιητικό φάσμα πολυπλοκότητας του περιβάλλοντος προσομοίωσης. Αυτά τα δεδομένα, συνεπώς, αποτελούν μία μέση απεικόνιση της συμπεριφοράς του οχήματος. Οι ελεγκτές οι οποίοι δημιουργούνται, εξετάζονται στη συνέχεια για την αποτελεσματικότητά τους, τόσο σε πιο εύκολα όσο και σε πιο δύσκολα σενάρια.

Ο αλγόριθμος ελέγχου, ο οποίος χρησιμοποιείται για την υλοποίηση των προσομοιώσεων και την καθοδήγηση του οχήματος, είναι ο ίδιος με αυτόν των ασαφών ελεγκτών, με μόνη αλλαγή ότι οι εισοδοί και εξοδοί της συνάρτησης που παρουσιάζεται στο κεφάλαιο 3, αφορούν κάποιο νευρωνικό δίκτυο, αντί για ασαφές σύστημα.

##### **4.4.1 ΕΚΠΑΙΔΕΥΣΗ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ ΕΛΕΓΧΟΥ ΓΙΑ ΚΙΝΗΣΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΧΩΡΙΣ ΕΜΠΟΔΙΑ**

Οι νευρωνικοί ελεγκτές για την εκτέλεση σεναρίου σε περιβάλλον χωρίς εμπόδια, δημιουργήθηκαν στο περιβάλλον της Matlab, με χρήση του NN Toolbox και των διαθέσιμων συναρτήσεων. Ως δεδομένα εισόδου δόθηκαν, υπό τη μορφή πινάκων, τέσσερις παράμετροι: οι καταγεγραμμένες συντεταγμένες του οχήματος στους άξονες x και y, το σφάλμα κατεύθυνσης και η απόσταση του οχήματος από το στόχο. Ως έξοδος των δικτύων, δόθηκε η ταχύτητα αριστερού και δεξιού τροχού. Τα δεδομένα αυτά αποτελούν και τους νευρώνες εισόδου και εξόδου αντίστοιχα. Το 70% των δεδομένων αυτών χρησιμοποιήθηκαν για εκπαίδευση του δικτύου, το 15% για τεστ του δικτύου και το 15% για αξιολόγηση του.



Εκπαιδεύτηκαν τρία διαφορετικά νευρωνικά δίκτυα, κλιμακούμενης πολυπλοκότητας, ως προς τα επίπεδα και τον αριθμό των νευρώνων, με την εξής δομή:

Απλός ελεγκτής **netSimple**: Ένα επίπεδο εισόδου με 4 νευρώνες, ένα κρυμμένο επίπεδο με 10 νευρώνες και ένα επίπεδο εξόδου με 2 νευρώνες.

Κανονικός ελεγκτής **netNormal**: Ένα επίπεδο εισόδου με 4 νευρώνες, δύο κρυμμένα επίπεδα με 4 και 15 νευρώνες σε αντίστοιχη σειρά και ένα επίπεδο εξόδου με 2 νευρώνες.

Σύνθετος ελεγκτής **netComplex**: Ένα επίπεδο εισόδου με 4 νευρώνες, τέσσερα κρυμμένα επίπεδα με 3, 10, 10 και 2 νευρώνες σε αντίστοιχη σειρά και ένα επίπεδο εξόδου με 2 νευρώνες.

Οι εκπαίδευση αποτελείται από τις εξής παραμέτρους:

- Τύπος νευρωνικού δικτύου: Δίκτυο με τροφοδότηση προς τα εμπρός, με χρήση του αλγορίθμου οπισθοδιάδωσης.
- Αλγόριθμος – συνάρτηση εκπαίδευσης: Resilient backpropagation (trainrp)
- Συνάρτηση ενεργοποίησης: Εφαπτόμενη σιγμοειδής (Hyperbolic tangent sigmoid ή tansig)
- Μέγιστος αριθμός εποχών (επαναλήψεων): 1000
- Μέθοδος αξιολόγησης εκπαίδευσης: Ελάχιστο Τετραφωνικό σφάλμα
- Στόχος αξιολόγησης (performance goal): 0
- Κατώφλι τερματισμού εκπαίδευσης για τη συνάρτηση κλίσης (minimum gradient descent): 0.01.

#### 4.4.2 ΠΡΟΣΟΜΟΙΩΣΗ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΚΙΝΗΣΗΣ ΟΧΗΜΑΤΟΣ ΜΕ ΝΕΥΡΩΝΙΚΟ ΕΛΕΓΚΤΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΧΩΡΙΣ ΕΜΠΟΔΙΑ

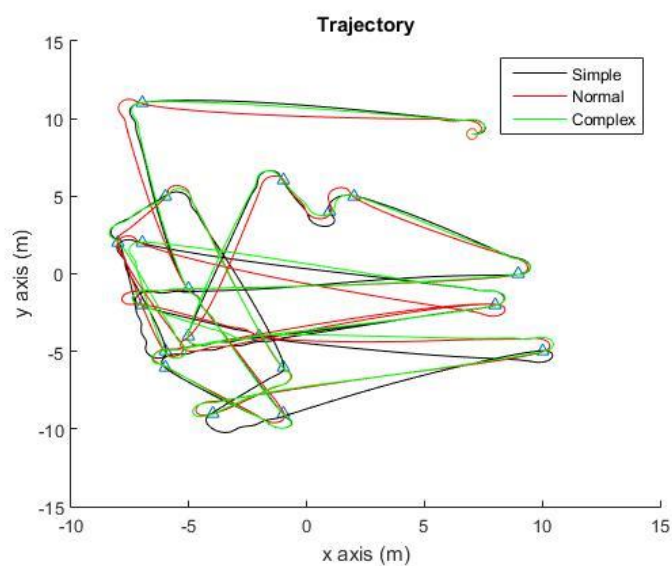
Προκειμένου να αξιολογηθεί η αποτελεσματικότητα ελέγχου του οχήματος από το εκάστοτε δίκτυο νευρώνων της προηγούμενης ενότητας, κλήθηκε το ρομποτικό όχημα να «πιάσει» 20 σημεία, 15 στα οποία εκπαιδεύτηκε και 5 σημεία επιπλέον, τα οποία ήταν τυχαία σε επιλογή, ώστε να διαφέρουν από αυτά της εκπαίδευσης του. Το σημείο εκκίνησης, σε σχέση με την προσομοίωση των ασαφών ελεγκτών άλλαξε, ώστε να μην υπάρξει το ίδιο μοτίβο διαδρομής στην αρχή του πειράματος. Ως σημείο εκκίνησης ορίστηκε το  $[x_1, y_1] = [0, 0]$ .

Ως μεταβλητές αξιολόγησης ορίστηκαν ο συνολικός χρόνος εκτέλεσης του σεναρίου και η συνολική απόσταση που διένυσε το όχημα, υπό τον έλεγχο του αντίστοιχου δικτύου. Τα αποτελέσματα των προσομοιώσεων παρουσιάζονται στον πίνακα 4.12. Ακόμη, στα σχήματα 4.53 και 4.54, απεικονίζεται η τροχιά και η μέση ταχύτητα του οχήματος για τον αντίστοιχο ελεγκτή.

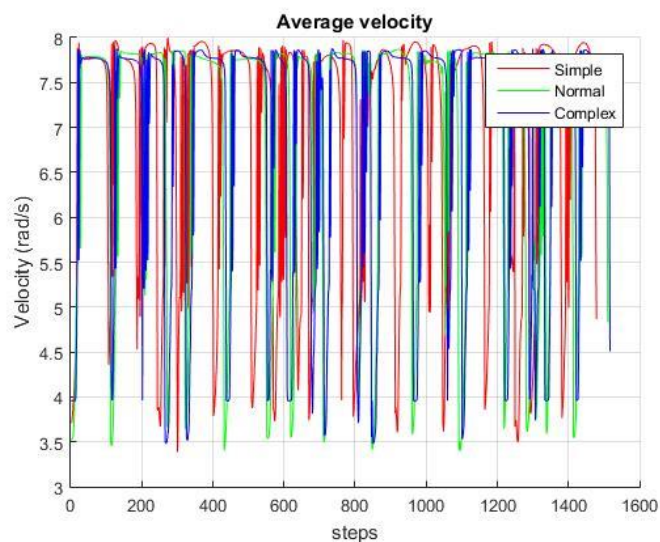
**Αρένα βαθμού «πολύ εύκολος»**

Νευρωνικός ελεγκτής	Χρόνος εκτέλεσης (s)	Συνολική διανυόμενη απόσταση (m)
Simple	68.73	237.58
Normal	70.54	223.16
Complex	70.85	221.22

Πίνακας 4.12 Αποτελέσματα προσομοιώσεων νευρωνικών ελεγκτών σε αρένα βαθμού πολυπλοκότητας «πολύ εύκολος»



Σχήμα 4.53 Τροχιά οχήματος για τους νευρωνικούς ελεγκτές σε αρένα πληρότητας 0%



*Σχήμα 4.54 Μέση ταχύτητα τροχών για τους νευρωνικούς ελεγκτές σε αρένα πληρότητας 0%*

Σύμφωνα με τον πίνακα αποτελεσμάτων και τα διαγράμματα τροχιάς και μέσης ταχύτητας, εξάγονται συμπεράσματα για τον τρόπο λειτουργίας και για την απόδοση των δικτύων στην καθοδήγηση του οχήματος.

Οι ελεγκτές παρουσιάζουν παρόμοια συμπεριφορά ως προς το χρόνο εκτέλεσης του σεναρίου. Ο απλός ελεγκτής περνάει από τα σημεία-στόχους γρηγορότερα από τους άλλους δύο ελεγκτές, ωστόσο η διαφορά μεταξύ των χρόνων τους δε θεωρείται πολύ μεγάλη. Ο κανονικός και ο σύνθετος ελεγκτής, διανύουν αρκετά μικρότερη απόσταση συνολικά, σε σχέση με τον απλό ελεγκτή. Παρατηρώντας τα σχήματα 4.53 και 4.54, ο απλός ελεγκτής αναπτύσσει την πιο υψηλή ταχύτητα, με μικρότερου εύρους διακυμάνσεις σε σχέση με τους άλλους δύο ελεγκτές. Ωστόσο, δεν αντιλαμβάνεται τόσο αποδοτικά το περιβάλλον του, ώστε να διαγράψει το όχημα διαδρομή ίδιας απόδοσης με τους δύο άλλους ελεγκτές. Ο κανονικός και ο σύνθετος ελεγκτής, αντιλαμβάνονται καλύτερα το χώρο και κινούνται πιο άμεσα όσο και πιο ομαλά από στόχο σε στόχο, ενώ εμφανίζουν ελάχιστες ταλαντώσεις στην κίνηση τους.

Συνεπώς, για ένα πρόβλημα με στόχο τη μείωση του χρόνου εκτέλεσης του σεναρίου, η χρήση πολύπλοκων δικτύων είναι μη αναγκαία, αφού ένας απλός νευρωνικός ελεγκτής του ενός επιπέδου κινείται πιο γρήγορα μεταξύ των στόχων. Για ένα πρόβλημα με στόχο την ελάχιστη διανυόμενη απόσταση, ένας πιο σύνθετος ελεγκτής, δύο ή παραπάνω επιπέδων, αποτελεί καλύτερη επιλογή. Η αύξηση των διόδων μετάδοσης της πληροφορίας εξασφαλίζει σοφότερες επιλογές στη διαδρομή που ακολουθεί στο χώρο.

#### 4.4.3 ΕΚΠΑΙΔΕΥΣΗ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ ΕΛΕΓΧΟΥ ΓΙΑ ΚΙΝΗΣΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΜΕ ΕΜΠΟΔΙΑ

Οι νευρωνικοί ελεγκτές για την εκτέλεση σεναρίου σε περιβάλλον με εμπόδια, δημιουργήθηκαν και αυτοί στο περιβάλλον της Matlab. Ως δεδομένα εισόδου δόθηκαν, υπό τη μορφή πινάκων, επτά παράμετροι: οι καταγεγραμμένες συντεταγμένες του οχήματος στους άξονες x και y, το σφάλμα κατεύθυνσης, η απόσταση του οχήματος από το στόχο και τα δεδομένα των τριών ομάδων των αισθητήρων, αριστερά, κεντρικά και δεξιά. Ως έξοδος των δικτύων, δόθηκε η ταχύτητα αριστερού και δεξιού τροχού. Τα δεδομένα αυτά αποτελούν και τους νευρώνες εισόδου και εξόδου αντίστοιχα. Το 70% των δεδομένων αυτών χρησιμοποιήθηκαν για εκπαίδευση του δικτύου, το 15% για τεστ του δικτύου και το 15% για αξιολόγηση του.

Εκπαιδεύτηκαν πέντε διαφορετικά νευρωνικά δίκτυα, κλιμακούμενης πολυπλοκότητας, ως προς τα επίπεδα και τον αριθμό των νευρώνων, με την εξής δομή:

Απλός ελεγκτής **netSimple**: Ένα επίπεδο εισόδου με 7 νευρώνες, ένα κρυμμένο επίπεδο με 10 νευρώνες και ένα επίπεδο εξόδου με 2 νευρώνες.

Κανονικός ελεγκτής **netNormal**: Ένα επίπεδο εισόδου με 7 νευρώνες, δύο κρυμμένα επίπεδα με 8 και 13 νευρώνες σε αντίστοιχη σειρά και ένα επίπεδο εξόδου με 2 νευρώνες.

Σύνθετος ελεγκτής **netComplex**: Ένα επίπεδο εισόδου με 7 νευρώνες, τέσσερα κρυμμένα επίπεδα με 3, 10, 12 και 5 νευρώνες σε αντίστοιχη σειρά και ένα επίπεδο εξόδου με 2 νευρώνες.

Αρκετά σύνθετος ελεγκτής **net2Complex**: Ένα επίπεδο εισόδου με 7 νευρώνες, πέντε κρυμμένα επίπεδα με 2, 5, 10, 8 και 2 νευρώνες σε αντίστοιχη σειρά και ένα επίπεδο εξόδου με 2 νευρώνες.

Πολύ σύνθετος ελεγκτής **net22Complex**: Ένα επίπεδο εισόδου με 7 νευρώνες, έξι κρυμμένα επίπεδα με 5, 10, 12, 5, 8 και 3 νευρώνες σε αντίστοιχη σειρά και ένα επίπεδο εξόδου με 2 νευρώνες.

Οι παράμετροι εκπαίδευσης είναι οι ίδιοι με την ενότητα 4.4.1.

#### 4.4.4 ΠΡΟΣΟΜΟΙΩΣΗ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΚΙΝΗΣΗΣ ΟΧΗΜΑΤΟΣ ΜΕ ΝΕΥΡΩΝΙΚΟ ΕΛΕΓΚΤΗ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΜΕ ΕΜΠΟΔΙΑ

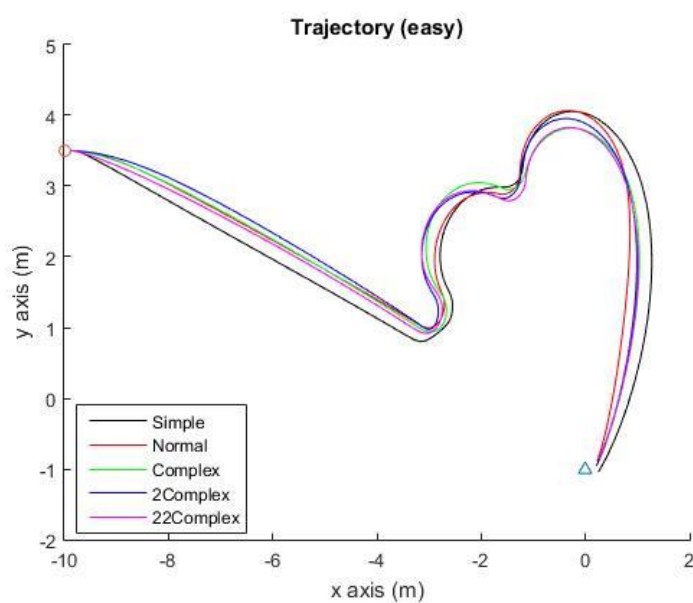
Στην προηγούμενη ενότητα, παρουσιάστηκε η δομή των νευρωνικών ελεγκτών που κατασκευάστηκαν για πλοήγηση οχήματος σε αρένες με σταθερά εμπόδια. Για την αξιολόγηση της συμπεριφοράς και της απόδοσης του οχήματος, υπό τον έλεγχο των νευρωνικών δικτύων ελέγχου, κλήθηκε το όχημα, όπως και στην αξιολόγηση των ασαφών ελεγκτών, να δράσει στις αρένες πληρότητας 8%, 25%, 58% και 83%. Φέροντας τον αντίστοιχο ελεγκτή κάθε φορά, το ρομποτικό όχημα πέρασε από μία συγκεκριμένη ακολουθία σημείων, η οποία διαφέρει ανάλογα με την αρένα, είναι ίδια όμως και για τους τρεις ελεγκτές σε κάθε περίπτωση. Ως μεταβλητές αξιολόγησης ορίστηκαν ο συνολικός χρόνος εκτέλεσης του σεναρίου και η συνολική απόσταση που διένυσε το εκάστοτε όχημα.

##### Αρένα βαθμού «εύκολος»

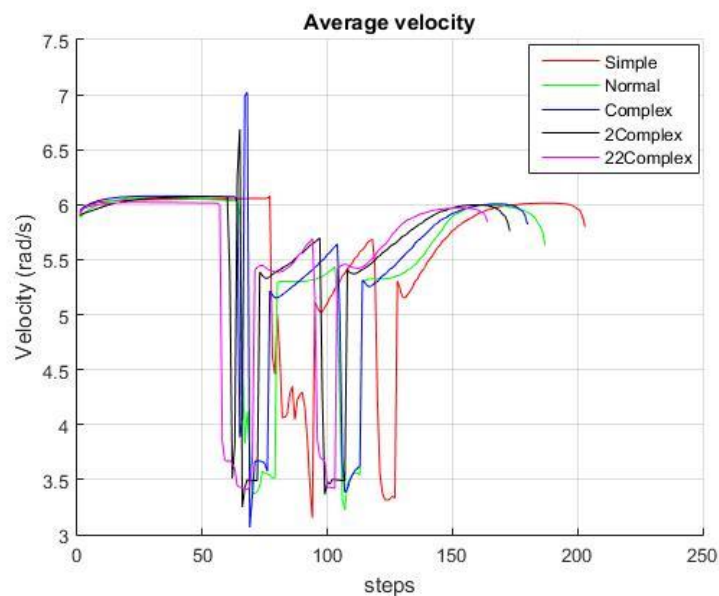
Η αρένα αυτή έχει το χαρακτήρα της επιβεβαίωσης του σεναρίου, για τους τρεις ελεγκτές και δε χρησιμοποιείται για λεπτομερή σύγκριση μεταξύ τους. Ωστόσο, για να υπάρχει πλήρης εικόνα των ελεγκτών, παρατίθενται τα αποτελέσματα σε αυτήν στον πίνακα 4.13. Στα σχήματα 4.55 και 4.56, δίνονται τα γραφήματα τροχιάς και μέσης ταχύτητας του οχήματος για τους τρεις ελεγκτές, στην αρένα αυτή. Σημείο εκκίνησης αποτελεί το σημείο  $[x_1, y_1] = [-10, 3.5]$  και σημείο στόχο το σημείο  $[x_2, y_2] = [0, -1]$ .

Νευρωνικός ελεγκτής	Χρόνος εκτέλεσης (s)	Συνολική διανυόμενη απόσταση (m)
Simple	9.81	29.02
Normal	8.78	26.21
Complex	8.57	26.37
2Complex	8.74	24.65
22Complex	8.87	26.58

Πίνακας 4.13 Αποτελέσματα προσομοιώσεων νευρωνικών ελεγκτών σε αρένα βαθμού πολυπλοκότητας «εύκολος»



Σχήμα 4.55 Τροχιά οχήματος για τους νευρωνικούς ελεγκτές σε αρένα πληρότητας 8%



Σχήμα 4.56 Μέση ταχύτητα τροχών για τους νευρωνικούς ελεγκτές σε αρένα πληρότητας 8%

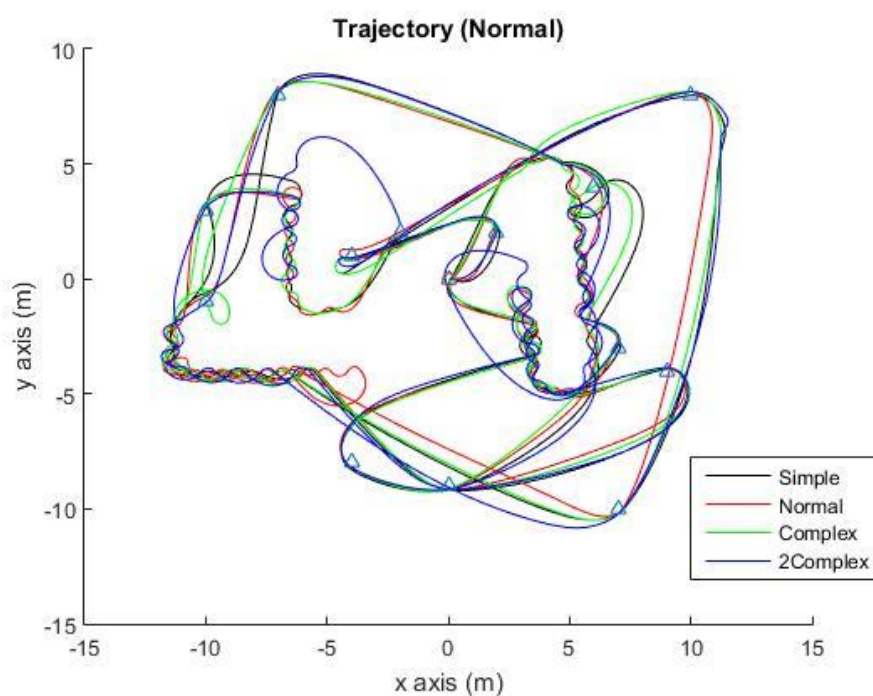
Όλοι οι ελεγκτές ανταποκρίνονται στην αποφυγή του εμποδίου και στην επιτυχή κίνηση από τον αρχικό στον τελικό στόχο, συνεπώς μπορούν πλέον να δοκιμαστούν σε πιο σύνθετα σενάρια. Ως πρώτο συμπέρασμα, παρατηρείται ότι, πέραν του απλού ελεγκτή, ο οποίος είναι και ο λιγότερο αποδοτικός, οι υπόλοιποι εμφανίζουν παρόμοια συμπεριφορά και στα δύο κριτήρια αξιολόγησης.

#### **Αρένα βαθμού «μέτριας δυσκολίας»**

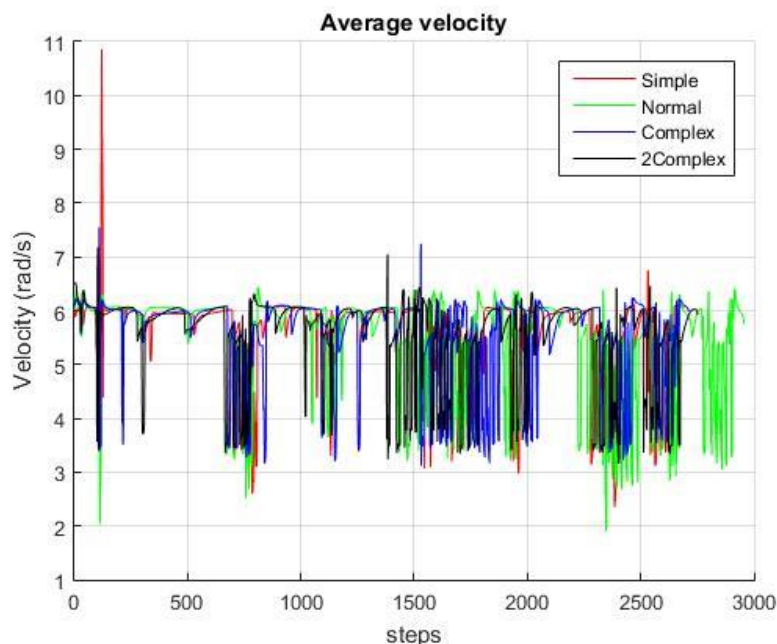
Στον πίνακα 4.14, παρουσιάζονται τα αποτελέσματα των προσομοιώσεων για την αρένα πληρότητας 25%. Στα σχήματα 4.57 και 4.58, δίνονται τα γραφήματα τροχιάς και μέσης ταχύτητας του οχήματος για τους αντίστοιχους ελεγκτές, στην αρένα αυτή. Οι ελεγκτές δοκιμάστηκαν στα 10 σημεία που είχε δοκιμαστεί ο ασαφής ελεγκτής σε αυτή την αρένα και σε 5 επιπρόσθετα καινούρια σημεία. Σημείο εκκίνησης ορίστηκε το  $[x_1, y_1] = [0, 0]$ .

Νευρωνικός ελεγκτής	Χρόνος εκτέλεσης (s)	Συνολική διανυόμενη απόσταση (m)
Simple	127.21	231.36
Normal	138.53	256.54
Complex	130.02	235.92
2Complex	137.22	249.61
22Complex	-	-

Πίνακας 4.14 Αποτελέσματα προσομοιώσεων νευρωνικών ελεγκτών σε αρένα βαθμού πολυπλοκότητας «μέτριας δυσκολίας»



Σχήμα 4.57 Τροχιά οχήματος για τους νευρωνικούς ελεγκτές σε αρένα πληρότητας 25%



Σχήμα 4.58 Μέση ταχύτητα τροχών για τους νευρωνικούς ελεγκτές σε αρένα πληρότητας 25%

Σύμφωνα με τα παραπάνω αποτελέσματα, σε περιβάλλον με μικρή πληρότητα σε εμπόδια, τόσο ένας σύνθετος όσο κι ένας πιο απλός ελεγκτής μπορεί να ανταπεξέλθει και να ολοκληρώσει το σενάριο με παρόμοια απόδοση. Οι ελεγκτές Simple και Complex παρουσιάζουν παρόμοια αποτελέσματα ως προς τα δύο κριτήρια και είναι αρκετά πιο αποδοτικοί από τους ελεγκτές Normal και 2Complex. Ο πολύ σύνθετος ελεγκτής 2Complex δεν κατάφερε να ολοκληρώσει το σενάριο.

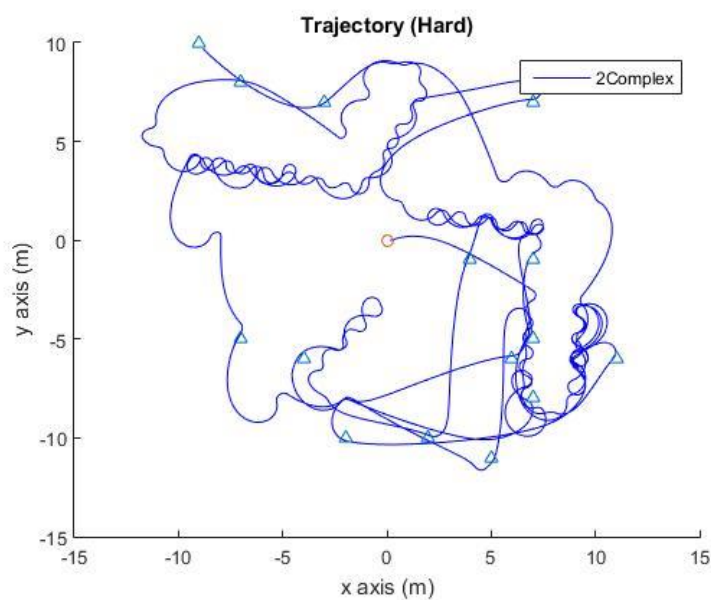
### Αρένα βαθμού «δύσκολος»

Στον πίνακα 4.15, παρουσιάζονται τα αποτελέσματα των προσομοιώσεων για την αρένα πληρότητας 58%. Στα σχήματα 4.59 και 4.60, δίνονται τα γραφήματα τροχιάς και μέσης ταχύτητας του οχήματος για τους αντίστοιχους ελεγκτές, στην αρένα αυτή. Οι ελεγκτές δοκιμάστηκαν στα 10 σημεία που είχε δοκιμαστεί ο ασαφής ελεγκτής σε αυτή την αρένα και σε 5 επιπρόσθετα καινούρια σημεία. Σημείο εκκίνησης ορίστηκε το  $[x_1, y_1] = [0, 0]$ .

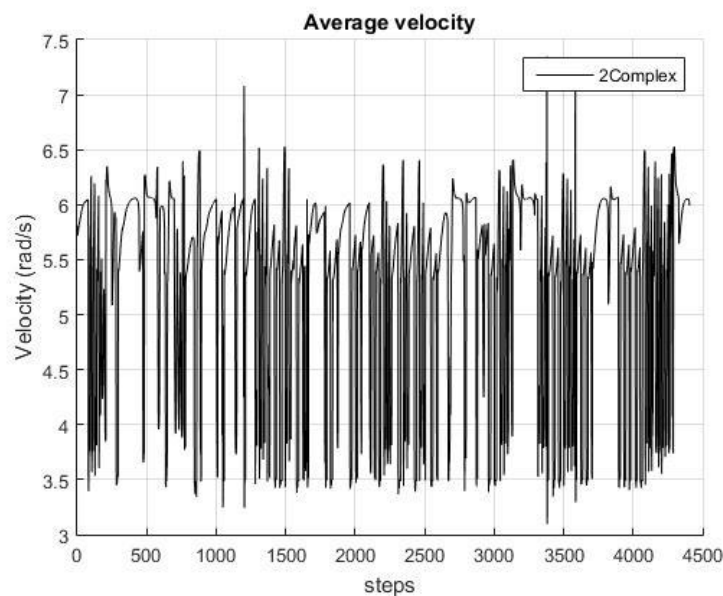


Νευρωνικός ελεγκτής	Χρόνος εκτέλεσης (s)	Συνολική διανυόμενη απόσταση (m)
Simple	-	-
Normal	-	-
Complex	-	-
2Complex	137.22	249.61
22Complex	-	-

Πίνακας 4.15 Αποτελέσματα προσομοιώσεων νευρωνικών ελεγκτών σε αρένα βαθμού πολυπλοκότητας «δύσκολος»



Σχήμα 4.59 Τροχιά οχήματος για τους νευρωνικούς ελεγκτές σε αρένα πληρότητας 58%



Σχήμα 4.60 Μέση ταχύτητα τροχών για τους νευρωνικούς ελεγκτές σε αρένα πληρότητας 58%

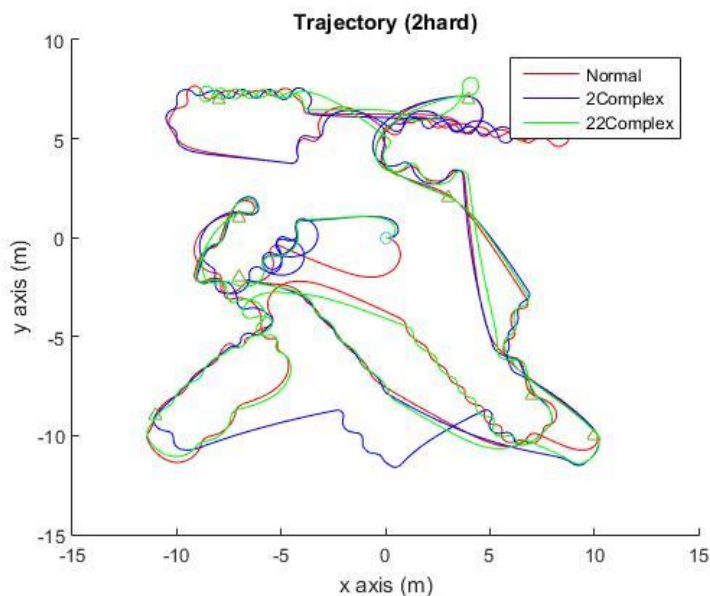
Με βάση τον πίνακα αποτελεσμάτων, μόνο ο ελεγκτής 2Complex ολοκληρώνει το σενάριο. Παρότι πρόκειται για την αρένα από την οποία προέρχονται τα δεδομένα εκπαίδευσης, φαίνεται ότι οι περισσότεροι ελεγκτές δεν μπόρεσαν να αντιμετωπίσουν την προσθήκη έξτρα σημείων στο σενάριο (εκτός δηλαδή αυτών που εκπαιδεύτηκαν) και την επερχόμενη αλλαγή στο μοτίβο των σημείων – στόχων.

#### Αρένα βαθμού «πολύ δύσκολος»

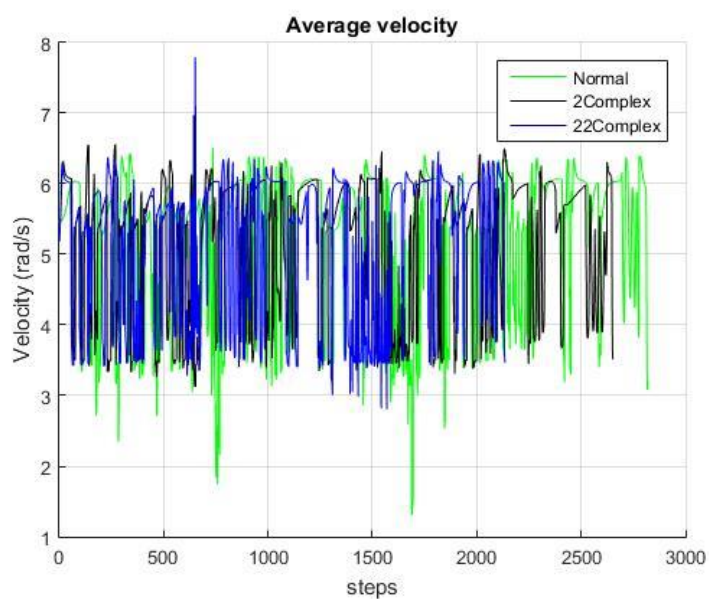
Στον πίνακα 4.16, παρουσιάζονται τα αποτελέσματα των προσομοιώσεων για την αρένα πληρότητας 83%. Στα σχήματα 4.61 και 4.62, δίνονται τα γραφήματα τροχιάς και μέσης ταχύτητας του οχήματος για τους αντίστοιχους ελεγκτές, στην αρένα αυτή. Οι ελεγκτές δοκιμάστηκαν στα 4 από τα 6 σημεία στα οποία είχαν δοκιμαστεί οι ασαφείς ελεγκτές σε αυτή την αρένα και σε 4 επιπρόσθετα καινούρια σημεία. Σημείο εκκίνησης ορίστηκε το  $[x_1, y_1] = [0, 0]$ .

Νευρωνικός ελεγκτής	Χρόνος εκτέλεσης (s)	Συνολική διανυόμενη απόσταση (m)
Simple	-	-
Normal	137.77	201.85
Complex	-	-
2Complex	138.67	205.68
22Complex	118.72	175.06

Πίνακας 4.16 Αποτελέσματα προσομοιώσεων νευρωνικών ελεγκτών σε αρένα βαθμού πολυπλοκότητας «πολύ δύσκολος»



Σχήμα 4.61 Τροχιά οχήματος για τους νευρωνικούς ελεγκτές σε αρένα πληρότητας 83%



Σχήμα 4.62 Μέση ταχύτητα τροχών για τους νευρωνικούς ελεγκτές σε αρένα πληρότητας 83%

Στο συγκεκριμένο σενάριο, εξετάζεται η συμπεριφορά του οχήματος υπό διαφορετικούς νευρωνικούς ελεγκτές, σε ένα αρκετά απαιτητικό περιβάλλον, με υψηλή πληρότητα σε εμπόδια. Με βάση τον πίνακα αποτελεσμάτων, οι ελεγκτές Normal, 2Complex και 22Complex, ολοκλήρωσαν με επιτυχία το σενάριο, σε αντίθεση με τους ελεγκτές Simple και Complex, οι οποίοι δεν μπόρεσαν να οδηγήσουν το όχημα ως το τελικό σημείο με ασφάλεια, χωρίς δηλαδή την ύπαρξη σύγκρουσης με κάποιο εμπόδιο ή τον εγκλωβισμό σε κάποιο επαναλαμβανόμενο βρόχο κινήσεων κατά την πλοήγηση.

Ο πολύ σύνθετος ελεγκτής είναι αυτός που εκτελεί το σενάριο στον ταχύτερο χρόνο αλλά και παράλληλα διανύοντας τη μικρότερη απόσταση. Τα αποτελέσματα των άλλων δύο ελεγκτών απέχουν κατά πολύ από αυτά του πολύ σύνθετου. Η δυνατότητα της επεξεργασίας της πληροφορίας σε πολλά επίπεδα φαίνεται ότι δίνει στον ελεγκτή 22Complex την ικανότητα να δρα πολύ πιο αποδοτικά από απλούστερους ελεγκτές σε περιβάλλοντα αυξημένης δυσκολίας. Το όχημα αναγνωρίζει πολύ πιο αποτελεσματικά τα μοτίβα στην κίνηση και επιλέγει διαδρομές που του αποφέρουν σημαντική εξοικονόμηση διαδρομής, κάτι που φαίνεται να του επιφέρει και σημαντικό κέρδος στο συνολικό χρόνο εκτέλεσης του σεναρίου, αφού οι ταχύτητες που αναπτύσσει δεν ακολουθούν πολύ διαφορετική πορεία από εκείνη του ελεγκτή 2Complex.

### **Γενικά συμπεράσματα**

Οι ελεγκτές παρουσιάζουν διαφορετική συμπεριφορά, η οποία εξαρτάται από την πολυπλοκότητα του εκάστοτε σεναρίου. Σε περιβάλλον με πολύ λίγα εμπόδια, η επιλογή ενός απλού ελεγκτή του ενός επιπέδου καλύπτει τις ανάγκες κίνησης του ρομποτικού οχήματος στο χώρο.

Αντιθέτως, όσο αυξάνεται η δυσκολία, η απόδοση πιο απλών δικτύων είναι φθίνουσα. Κάποιοι ελεγκτές μάλιστα, όπως ο πιο απλός αλλά και ο σύνθετος, με τα 4 επίπεδα, δεν κατάφεραν να ολοκληρώσουν κανένα από τα δύο τελευταία, πιο δύσκολα σενάρια. Παρότι ο κανονικός ελεγκτής δύο επιπέδων εκτέλεσε το πιο δύσκολο σενάριο με επιτυχία και μάλιστα με ικανοποιητικά αποτελέσματα, κρίνεται πιο ασφαλές να γίνεται χρήση δικτύων με αρκετούς νευρώνες και επίπεδα. Η αύξηση διάδοσης της πληροφορίας και οι πολυάριθμες αλλαγές μεταξύ των βαρών των νευρώνων, φαίνεται να οδηγούν σε αναγνώριση των μοτίβων και σε αρκετά πιο αποτελεσματική αντιμετώπιση των σεναρίων, τόσο ως προς το χρόνο όσο και ως προς την απόσταση. Ο ελεγκτής φαίνεται να αναγνωρίζει πιο σύντομες οδούς για την κίνηση από σημείο σε σημείο, ενώ αναπτύσσει παράλληλα υψηλή ταχύτητα με σταθερές και μικρότερου εύρους μεταπτώσεις.

## ΚΕΦΑΛΑΙΟ 5

# ΣΥΜΠΕΡΑΣΜΑΤΑ/ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

### 5.1 ΣΥΜΠΕΡΑΣΜΑΤΑ

Σκοπός της παρούσας εργασίας είναι η παρατήρηση, η μελέτη και η ανάλυση της κίνησης ενός αυτόνομου ρομποτικού οχήματος διαφορετικής κίνησης σε άγνωστα για τους ελεγκτές περιβάλλοντα. Η πλοήγηση γίνεται με διαφορετικής δομής ασαφείς και νευρωνικούς ελεγκτές, οι οποίοι κατασκευάστηκαν και δοκιμάστηκαν στις προσομοιώσεις. Η αξιολόγηση και σύγκριση των ελεγκτών γίνεται ανά κατηγορία (οι ασαφείς ελεγκτές μεταξύ τους και ομοίως οι νευρωνικοί) και οι μεταβλητές αξιολόγησης είναι ο χρόνος εκτέλεσης του σεναρίου και η συνολική απόσταση που διανύθηκε.

Σε ότι αφορά τους ασαφείς ελεγκτές, σε περιβάλλον άδειας αρένας, κατασκευάστηκαν πέντε ελεγκτές κλιμακούμενης πολυπλοκότητας ως προς τη δομή τους. Ο πιο σύνθετος ελεγκτής που κατασκευάστηκε, εξασφαλίζει την ταχύτερη διαδρομή με τη λιγότερη απόσταση. Σε γενικές γραμμές, οι πιο σύνθετοι ελεγκτές οδηγούν και στην πιο επιθυμητή συμπεριφορά του οχήματος. Σε ότι αφορά αρένες με διαφορετική πυκνότητα εμποδίων, επιλέχθηκαν οι τρεις πιο αποδοτικοί ελεγκτές από πριν και προσαρμόστηκαν στα δεδομένα περιβάλλοντος με εμπόδια. Ο ελεγκτής μέτριας πολυπλοκότητας εξασφαλίζει ταχύτερη απόδοση στις αρένες μικρής δυσκολίας, ωστόσο ο πιο σύνθετος ελεγκτής οδηγεί το όχημα στο να διανύσει τη μικρότερη απόσταση. Στις αρένες πιο υψηλής πολυπλοκότητας, ο πιο σύνθετος ελεγκτής είναι και ο πιο αποδοτικός ως προς τα δύο κριτήρια.

Σε ότι αφορά τους ελεγκτές νευρωνικών δικτύων, σε περιβάλλον άδειας αρένας, κατασκευάστηκαν τρεις ελεγκτές με αυξανόμενη πολυπλοκότητα στη δομή τους. Ο πιο απλός ελεγκτής φαίνεται να εξασφαλίζει την ταχύτερη κίνηση του οχήματος. Οι δύο πιο σύνθετοι ελεγκτές, ωστόσο, έχουν παρόμοια συμπεριφορά και εξασφαλίζουν αρκετά μικρότερη συνολική διανυόμενη απόσταση. Σε ότι αφορά τα κλιμακούμενης πολυπλοκότητας περιβάλλοντα με εμπόδια, σχεδιάστηκαν πέντε ελεγκτές, διαφορετικής και αυξανόμενης πολυπλοκότητας ως προς τα επίπεδα και τον αριθμό των νευρώνων.

Σύμφωνα με τα αποτελέσματα, για σενάρια μικρής δυσκολίας, η χρήση ενός απλού ή μέτριου σε σύνθεση ελεγκτή κρίνεται ικανοποιητική. Για σενάρια αυξημένης δυσκολίας, ένας αρκετά σύνθετος ελεγκτής είναι αναγκαίος για την ολοκλήρωση του σεναρίου, ενώ είναι σαφώς πιο αποδοτικός από έναν απλό ή μέτριας σύνθεσης ελεγκτή. Παρατηρήθηκε, ότι όσο αυξανόταν η πολυπλοκότητα των σεναρίων, κάποιοι ελεγκτές, κυρίως οι πιο απλοί, δεν μπορούσαν να ολοκληρώσουν το σενάριο, να περάσουν δηλαδή με επιτυχία από όλους τους στόχους. Μάλιστα, στην αρένα από την οποία αντλήθηκαν τα δεδομένα εκπαίδευσης, μόνο ένας ελεγκτής κατάφερε να οδηγήσει το όχημα σε ολοκλήρωση του σεναρίου. Πιθανότατα, τα δίκτυα που σχεδιάστηκαν, δεν είναι τόσο ικανά να ανταποκριθούν σε ένα τελείως διαφορετικό σενάριο. Ωστόσο, σε γενικές γραμμές, τα δίκτυα οδήγησαν ικανοποιητικά αποτελέσματα το όχημα.

## 5.2 ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Μία πιθανή συνέχεια των πεπραγμένων της παρούσας εργασίας, θα μπορούσε να είναι η δημιουργία κάποιων σεναρίων σε εργαστηριακό περιβάλλον, τα οποία να προσεγγίζουν σε πολυπλοκότητα τα περιβάλλοντα που δημιουργήθηκαν στην εργασία αυτή και η εφαρμογή των ελεγκτών σε πραγματικό περιβάλλον ή η ενίσχυση τους για να κινούνται ομαλά σε αυτά.

Σε επίπεδο βελτιστοποίησης, θα μπορούσαν να επιλεγούν κάποιοι ασαφείς ή νευρωνικοί ελεγκτές, οι οποίοι είχαν αρκετά καλά ή τα καλύτερα αποτελέσματα ανά περίπτωση και να βελτιστοποιηθούν με κάποιο αλγόριθμο βελτιστοποίησης (π.χ γενετικό αλγόριθμο, αλγόριθμο σμήνους σωματιδίων), αλλάζοντας τις συναρτήσεις συμμετοχής και τα βάρη τους αντίστοιχα. Στη συνέχεια, θα μπορούσε να γίνει σύγκριση με τους αρχικούς και να εξαχθούν συμπεράσματα.

## ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Γιάννης Α. Θεοδώρου, Εισαγωγή στην Ασαφή Λογική, Εκδόσεις Τζιόλα, Θεσσαλονίκη 2010.
- [2] Σ. Βολογιαννίδης, “Ευφυής Έλεγχος, Θεωρία και Εφαρμογές”, Διδακτικές σημειώσεις, Τμήματος Πληροφορικής και Επικοινωνιών, ΤΕΙ Σερρών, 2009.
- [3] Γεωργάκης Θωμάς, “Ασαφής λογική για ευφυή έλεγχο με εφαρμογές σε οχήματα αυτόματης πλοήγησης”, Διπλωματική εργασία, Πανεπιστήμιο Θεσσαλίας, Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Η/Υ, Βόλος 2017.
- [4] Β. Καμπουρλάζος, Α. Παπακώστας: Εισαγωγή στην υπολογιστική νοημοσύνη, μια ολιστική προσέγγιση, Ελληνικά Ακαδημαϊκά Ηλεκτρονικά Συγγράμματα και Βοηθήματα.
- [5] M. Negnevitsky, Artificial Intelligence - A Guide to Intelligent Systems, 2nd Ed (2005).
- [6] Tarun Gupta, R. R. Boudreaux, R. M. Nelms, John Y. Hung, “Implementation of a Fuzzy Controller for DC–DC Converters Using an Inexpensive 8-b Microcontroller,” *IEEE Transactions on Industrial Electronics*, vol. 44, Issue 5, 1997.
- [7] H. A. Malki, D. Misir, D. Feigenspan and Guanrong Chen, "Fuzzy PID control of a flexible-joint robot arm with uncertainties from time-varying loads," in *IEEE Transactions on Control Systems Technology*, vol. 5, no. 3, pp. 371-378, May 1997.
- [8] N.Tsourveloudis, L. Doitsidis, K. Valavanis, “Autonomous Navigation of Unmanned Vehicles: A Fuzzy Logic Perspective,” *Cutting Edge Robotics*, pp. 291-310, Pro Literatur Verlag, ISBN:3-86611-038-3, 2005.
- [9] K. Tanaka and T. Kosaki, "Design of a stable fuzzy controller for an articulated vehicle," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 27, no. 3, pp. 552-558, June 1997.
- [10] Wei Li, "Fuzzy logic-based 'perception-action' behavior control of a mobile robot in uncertain environments," *Proceedings of 1994 IEEE 3rd International Fuzzy Systems Conference*, 1994, pp. 1626-1631 vol.3.
- [11] H. Seraji and A. Howard, "Behavior-based robot navigation on challenging terrain: A fuzzy logic approach," in *IEEE Transactions on Robotics and Automation*, vol. 18, no. 3, pp. 308-321, June 2002,
- [12] A.D.Dongare, R.R.Kharde, Amit D.Kachare, Introduction to Artificial Neural Network, *International Journal of Engineering and Innovative Technology (IJEIT)*, Volume 2, Issue 1, July 2012.
- [13] I.A. Basheer, M. Hajmeer, Artificial neural networks: fundamentals, computing, design, and application, *Journal of Microbiological Methods*, 43 (2000) 3–31.

- [14] Ben Krose, Patrick van der Smagt, An introduction to Neural Networks, 8<sup>th</sup> edition, November 1996.
- [15] Hernández-Alvarado, Rodrigo, Luis G. García-Valdovinos, Tomás Salgado-Jiménez, Alfonso Gómez-Espinosa, and Fernando Fonseca-Navarro. 2016. "Neural Network-Based Self-Tuning PID Control for Underwater Vehicles" *Sensors* 16, no. 9: 1429.
- [16] Nicolas Durand, Jean-Marc Alliot, Joseph Noailles. Collision avoidance using neural networks learned by genetic algorithms. IEA-AEI 1996, 9th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert systems, Jun 1996, Nagoya, Japan.
- [17] Graf and LaLonde, "A neural controller for collision-free movement of general robot manipulators," *IEEE 1988 International Conference on Neural Networks*, 1988, pp. 77-84 vol.1.
- [18] Janglová D. Neural Networks in Mobile Robot Motion. *International Journal of Advanced Robotic Systems*. March 2004.
- [19] M.K. Singha and D.R. Parhi, Path optimisation of a mobile robot using an artificial neural network controller, *International Journal of Systems Science*, vol. 42, Issue 1, 2010.
- [20] Sandeep Kumar Malu & Jharna Majumdar, Kinematics, Localization and Control of Differential Drive Mobile Robot, *Global Journal of Researches in Engineering: H Robotics & Nano-Tech*, vol. 14, Issue 1, 2014.
- [21] <https://www.generationrobots.com/media/Pioneer3DX-P3DX-RevA.pdf>
- [22] Stuart J. Russell & Peter Norvig, Τεχνητή Νοημοσύνη, Μια Σύγχρονη προσέγγιση, Εκδόσεις Κλειδάριθμος, 2<sup>η</sup> έκδοση.
- [23] Chris M. Bishop, Neural networks and their applications, *Review of Scientific Instruments* 65, 1803 (1994)
- [24] Neha Gupta, Artificial Neural Network, *Network and Complex Systems*, Vol.3, No.1, 2013-*Selected from International Conference on Recent Trends in Applied Sciences with Engineering Applications*.