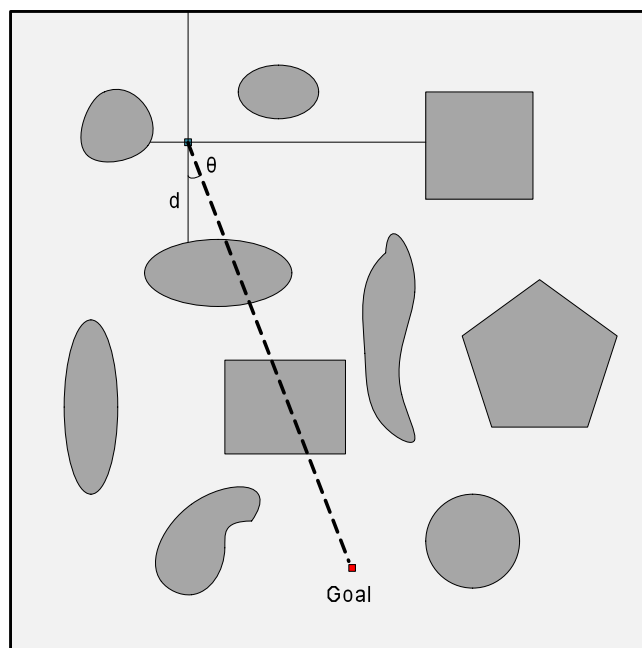


ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ ΚΑΙ ΕΥΦΥΗ ΥΠΟΛΟΓΙΣΤΙΚΑ ΣΥΣΤΗΜΑΤΑ

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 3: Ασαφή συστήματα

Αντικείμενο της εργαστηριακής άσκησης αυτής είναι ο σχεδιασμός και η υλοποίηση ασαφών συστημάτων. Πιο συγκεκριμένα, θα μελετηθεί το πρόβλημα του προσδιορισμού της τροχιάς ενός κινούμενου οχήματος με τη βοήθεια ενός ασαφούς συστήματος. Το συγκεκριμένο όχημα θα υποθέσουμε ότι είναι σημειακό και βρίσκεται σε ένα προκαθορισμένο περιβάλλον ακίνητων εμποδίων (διάστασης 400x400 pixels). Στο χώρο κίνησης θα καθορίζεται επιπλέον ένα σημείο εκκίνησης και ένας επιθυμητός στόχος, τον οποίο το κινούμενο όχημα ζητείται να προσεγγίσει, αποφεύγοντας τις συγκρούσεις με τα εμπόδια (βλ. παρακάτω Σχήμα). Οι επιτρεπτές κινήσεις του οχήματος περιορίζονται στις τέσσερις κατευθύνσεις (αριστερά, δεξιά, πάνω, κάτω) με ταχύτητες οι οποίες μπορούν να φτάσουν τα 15 pixel/κίνηση. Το ασαφές σύστημα το οποίο θα ελέγχει την κίνηση του οχήματος θα έχει δύο εισόδους και μία έξοδο. Για κάθε μία από τις τέσσερις δυνατές κατευθύνσεις κίνησης θα δέχεται στην πρώτη είσοδό του την απόσταση d (σε pixels) από το πλησιέστερο εμπόδιο κατά την κατεύθυνση αυτή και στη δεύτερη είσοδο τη γωνία θ που σχηματίζει η κατεύθυνση κίνησης με την ευθεία που ορίζει η κατεύθυνση του στόχου (0-180 μοίρες). Στην έξοδό του θα δίνει την προτεινόμενη ταχύτητα κίνησης προς την συγκεκριμένη κατεύθυνση. Αφού το ασαφές σύστημα χρησιμοποιηθεί τέσσερις φορές, μία φορά για κάθε κατεύθυνση και προτείνει τις αντίστοιχες ταχύτητες, το όχημα θα κινηθεί τελικά προς την κατεύθυνση εκείνη για την οποία το σύστημα πρότεινε τη μεγαλύτερη ταχύτητα.



Σχήμα. Περιβάλλον κίνησης οχήματος

Στο πλαίσιο της 3^{ης} εργαστηριακής άσκησης ζητείται η υλοποίηση της παραπάνω εφαρμογής στο περιβάλλον της MATLAB, κάνοντας χρήση (αν επιθυμείτε) των έτοιμων συναρτήσεων και παραδειγμάτων χώρων κίνησης που έχουν υλοποιηθεί για το σκοπό αυτό και που διατίθενται στο εργαστήριο. Χρησιμοποιώντας το Fuzzy Logic Toolbox της MATLAB και (αν επιθυμείτε) το γραφικό περιβάλλον του θα πρέπει να ορίσετε ένα ασαφές σύστημα τύπου Mamdani με τις κατάλληλες συναρτήσεις συμμετοχής των μεταβλητών εισόδου και εξόδου και τους κατάλληλους ασαφείς κανόνες το οποίο θα ελέγχει την κίνηση του οχήματος.

Οι συναρτήσεις που έχουν υλοποιηθεί μετακινούν επαναληπτικά το σημειακό όχημα από την αρχική θέση προς την κατεύθυνση εκείνη και με την ταχύτητα εκείνη που καθορίζονται από το ασαφές σύστημα. Επιπλέον, έχουν τη δυνατότητα να εμφανίζουν στην οθόνη το χώρο κίνησης και την τροχιά την οποία διαγράφει το όχημα, για προεπισκόπηση και έλεγχο των αποτελεσμάτων. Πιο αναλυτικά το προγραμματιστικό περιβάλλον που έχει υλοποιηθεί αποτελείται από 6 μεταβλητές global και 5 συναρτήσεις.

Οι μεταβλητές είναι οι εξής:

map_size;	Η διάσταση σε pixels του τετράγωνου χώρου κίνησης (η προτεινόμενη διάσταση είναι 400 pixels)
map	Ο χώρος κίνησης ως ένας τετραγωνικός διδιάστατος πίνακας Αν ένα στοιχείο του πίνακα είναι 1, τότε στο σημείο αυτό υπάρχει εμπόδιο, ενώ αν είναι 0 τότε δεν υπάρχει εμπόδιο. Τα σημεία που αντιστοιχούν στην αρχική θέση του οχήματος και στη θέση του στόχου έχουν τιμή 0.25. Τέλος για τη γραφική απεικόνιση της τροχιάς του οχήματος τα σημεία από τα οποία πέρασε τίθενται στην τιμή 0.5.
init_position_x init_position_y	Η αρχική θέση του οχήματος. init_position_x είναι η οριζόντια μετατόπιση του οχήματος από το αριστερό όριο του χώρου κίνησης και init_position_y η κατακόρυφη μετατόπιση από το πάνω όριο του χώρου κίνησης.
goal_position_x goal_position_y	Η θέση του στόχου, ορισμένη με τον ίδιο τρόπο όπως η αρχική θέση του οχήματος.

Οι συναρτήσεις είναι οι εξής:

function create_map(plan)

Κατασκευάζει τον πίνακα **map** με βάση ένα γενικό σχέδιο που ορίζει ο πίνακας **plan**. Ο πίνακας **plan** είναι ένας πίνακας αυθαίρετης διάστασης αρκετά μικρότερες της **map_size**, με δυνατή τιμή κάθε στοιχείου του 0, 1, 2 ή 3. Το 0 αντιστοιχεί σε ελεύθερο χώρο, το 1 σε εμπόδιο το 2 στην αρχική θέση του οχήματος και το 3 στη θέση του στόχου. Ο πίνακας πρέπει να έχει ακριβώς ένα στοιχείο με την τιμή 2 και ένα με την τιμή 3. Η συνάρτηση διαμερίζει τον χώρο διάστασης **map_size** σε ισομεγέθη ορθογώνια ανάλογα με τη διάσταση του πίνακα **plan** και χαρακτηρίζει κάθε ένα ως ελεύθερο χώρο ή ως χώρο εμποδίων ανάλογα με την τιμή του αντίστοιχου στοιχείου του πίνακα **plan**. Η αρχική θέση του οχήματος και η θέση του στόχου τίθενται να είναι το κέντρο του ορθογωνίου του αντίστοιχου τμήματος για το οποίο ο πίνακας **plan**, έχει

την τιμή 2 ή 3 αντίστοιχα. Αντίστοιχα ενημερώνονται και οι μεταβλητές **init_position_x**, **init_position_y**, **goal_position_x** και **goal_position_y**.

Πέντε διαφορετικά παραδείγματα αυξανόμενου βαθμού δυσκολίας για τον πίνακα **plan** περιέχονται στο αρχείο **plans.m** τα οποία και θα πρέπει να χρησιμοποιήσετε σε συνδυασμό με το ασαφές σύστημα που θα κατασκευάσετε.

Αφού κατασκευαστεί ο χώρος κίνησης **map** μπορεί να απεικονιστεί γραφικά με την κλήση **imshow(map)**.

function distance = distance_to_obstacles(current_position_x, current_position_y)

Υπολογίζει και επιστρέφει τον 1x4 πίνακα **distance**, ο οποίος περιέχει τις αποστάσεις σε pixels της θέσης **current_position_x**, **current_position_y** από το πλησιέστερο εμπόδιο, κατά την ανατολική, βόρεια, δυτική και νότια κατεύθυνση αντίστοιχα.

**function angle = angle_to_goal(current_position_x, current_position_y,
goal_position_x, goal_position_y)**

Υπολογίζει και επιστρέφει τον 1x4 πίνακα **angle**, ο οποίος περιέχει τις γωνίες που σχηματίζει η θέση **current_position_x**, **current_position_y** με τη θέση **goal_position_x**, **goal_position_y** σε σχέση με την ανατολική, βόρεια, δυτική και νότια κατεύθυνση αντίστοιχα.

function [x, y] = do_step(current_position_x, current_position_y, step_x, step_y)

Μετακινεί το όχημα από τη θέση **current_position_x**, **current_position_y** κατά **step_x** ή κατά **step_y** pixels κατά την οριζόντια ή κατακόρυφη διεύθυνση αντίστοιχα. Αν **step_x > 0** η μετατόπιση γίνεται ανατολικά, και δυτικά αν **step_x < 0**. Αντίστοιχα, αν **step_y > 0** η μετατόπιση γίνεται νότια, και βόρεια αν **step_y < 0**. Μόνο μια από τις μεταβλητές **step_x** και **step_y** επιτρέπεται να είναι διάφορη του μηδενός σε κάθε κλήση.

function approach_goal(fis, max_iterations)

Χρησιμοποιεί το ασαφές σύστημα **fis** προκειμένου να καθοδηγήσει το όχημα από την αρχική του θέση στη θέση του στόχου. Η προσέγγιση γίνεται με επαναληπτικό τρόπο, και σε κάθε βήμα το όχημα μετακινείται προς μία από τις τέσσερις κατευθύνσεις ανάλογα με το αποτέλεσμα της κλήσης του ασαφούς συστήματος. Η επαναληπτική διαδικασία θα τερματίσει ύστερα από το πολύ **max_iterations** επαναλήψεις, ή αν το όχημα φτάσει στο στόχο νωρίτερα.

Ολοκληρωμένο το πρόγραμμα ελέγχου του οχήματος το οποίο κάνει χρήση των παραπάνω συναρτήσεων βρίσκεται στο αρχείο **control_vehicle.m**. Το πρόγραμμα αυτό για την λειτουργία του προϋποθέτει την ύπαρξη του ασαφούς συστήματος, αποθηκευμένου στο αρχείο **my_fuzzy_system.fis**, το οποίο καλείστε να υλοποιήσετε