



ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΤΜΗΜΑ ΗΜΜΥ. ΤΟΜΕΑΣ ΗΛΕΚΤΡΟΝΙΚΗΣ
ΑΣΑΦΗ ΣΥΣΤΗΜΑΤΑ (ΥΠΟΛΟΓΙΣΤΙΚΗ ΝΟΗΜΟΣΤΝΗ)

Εργασία 2

Car Control - Σειρά 3

Συντάκτης:
Χρήστος Χουτουρίδης [8997]
cchoutou@ece.auth.gr

Διδάσκοντες:
Θεοχάρης Ιωάννης
theochar@ece.auth.gr

Χαδουλός Χρήστος
christgc@auth.gr

26 Οκτωβρίου 2025

1. ΕΙΣΑΓΩΓΗ

Στην παρούσα εργασία σχεδιάστηκε και αξιολογήθηκε ένας **ελεγκτής ασαφούς λογικής (FLC)** τύπου Mamdani για το πρόβλημα *Car Control* (Σειρά 3). Το όχημα κινείται σε επίπεδο με *σκαλοπάτια/εμπόδια* και στόχος είναι να ακολουθεί μια ασφαλή τροχιά, λαμβάνοντας ως πληροφορία μόνο τις *αποστάσεις από τα εμπόδια* και τον *προσανατολισμό* του.

Ο ελεγκτής υλοποιεί κανόνες της μορφής

IF d_V is $\{S, M, L\}$ AND d_H is $\{S, M, L\}$ AND θ is $\{N, ZE, P\}$ THEN $\Delta\theta$ is $\{N, ZE, P\}$,

με τελεστές AND : min, OR : max, συνεπαγωγή min, σύνθεση/συσσώρευση sum και αποασαφοποίηση centroid (COA), όπως ζητείται.

Η ανάπτυξη έγινε σε *MATLAB*. Το FIS δημιουργήθηκε μέσω του *Fuzzy Logic Designer* (GUI) και αποθηκεύτηκε σε αρχείο *.fis*. Η προσομοίωση της κίνησης και η παραγωγή των αποτελεσμάτων έγιναν προγραμματιστικά.

1.1. Παραδοτέα

Τα παραδοτέα της εργασίας αποτελούνται από:

- Την παρούσα αναφορά.
- Τον κατάλογο **source**, με τον κώδικα της matlab.
- Το **σύνδεσμο με το αποθετήριο** που περιέχει τον κώδικα της matlab καθώς και αυτόν της αναφοράς.

2. ΥΛΟΠΟΙΗΣΗ

Πριν τη στοχοθετημένη αξιολόγηση, περιγράφεται συνοπτικά η αρχιτεκτονική της υλοποίησης. Στόχος μας ήταν ένα *αυτοτελές σενάριο προσομοίωσης* που να συνδυάζει:

- Ένα "νοητό" περιβάλλον με σκαλοπάτια/τοίχους
- Την κινηματική του οχήματος
- Τον ασαφή ελεγκτή.

Στο σχήμα 3 φαίνεται το περιβάλλον της εργασίας.

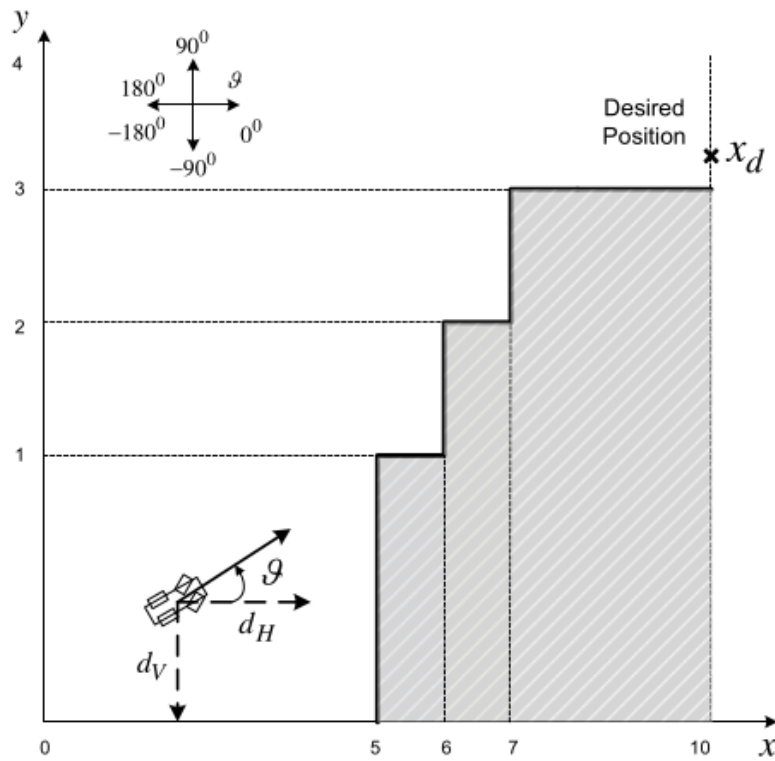
Για να διατηρηθεί καθαρή διεπαφή, το περιβάλλον και η κινηματική ενσωματώθηκαν σε δύο βοηθητικές συναρτήσεις, ενώ ο ελεγκτής ορίστηκε σε ανεξάρτητη συνάρτηση/αντικείμενο FIS. Η προσομοίωση εκτελείται από ένα κύριο script που διακρητοποιεί τον χρόνο, τροφοδοτεί τον ελεγκτή με μετρήσεις και ενημερώνει την κατάσταση.

2.1. wallDistance

Η `wallDistance(corners, x, y)` υλοποιεί τον "αισθητήρα" του περιβάλλοντος. Δεδομένων των *γωνιών* που ορίζουν τα σκαλοπάτια (πίνακας `corners` με σημεία (x_i, y_i)), υπολογίζει:

- d_h : την *οριζόντια* απόσταση μέχρι τον επόμενο τοίχο στα δεξιά του οχήματος (x -διεύθυνση),
- d_v : την *κάθετη* απόσταση μέχρι τον πλησιέστερο "πάτο" κάτω από το όχημα (y -διεύθυνση).

Ο αλγόριθμος ταξινομεί κατάλληλα τα `corners` (με `sortrows`) και επιλέγει τα *ενεργά* τμήματα τοίχων ανάλογα με την τρέχουσα θέση (x, y) . Τιμές $d_h, d_v \leq 0$ αντιστοιχούν σε σύγκρουση, ενώ ∞ δηλώνει απουσία τοίχου προς την εξεταζόμενη κατεύθυνση. Οι μετρήσεις κατόπιν *κορεσμού* $[0, 1]$ χρησιμοποιούνται αυτούσιες ως είσοδοι του FIS, όπως ορίζει η εκφώνηση.



Σχήμα 1: Απεικόνιση του περιβάλλοντος με σκιασμένες περιοχές.

2.2. moveCar

Η $\text{moveCar}(x, y, \theta, \Delta\theta)$ υλοποιεί τη στοιχειώδη κινηματική: πρώτα εφαρμόζεται η διόρθωση γωνίας $\theta \leftarrow \theta + \Delta\theta$, έπειτα ενημερώνεται η θέση

$$x \leftarrow x + v T_s \cos(\theta), \quad y \leftarrow y + v T_s \sin(\theta),$$

με σταθερό μέτρο ταχύτητας v και περίοδο δειγματοληψίας T_s . Η γωνία χρησιμοποιείται σε μοίρες ώστε να είναι συμβατή με τα διαστήματα της εκφώνησης.

2.3. carFLC

Ο ασαφής ελεγκτής είναι Mamdani με:

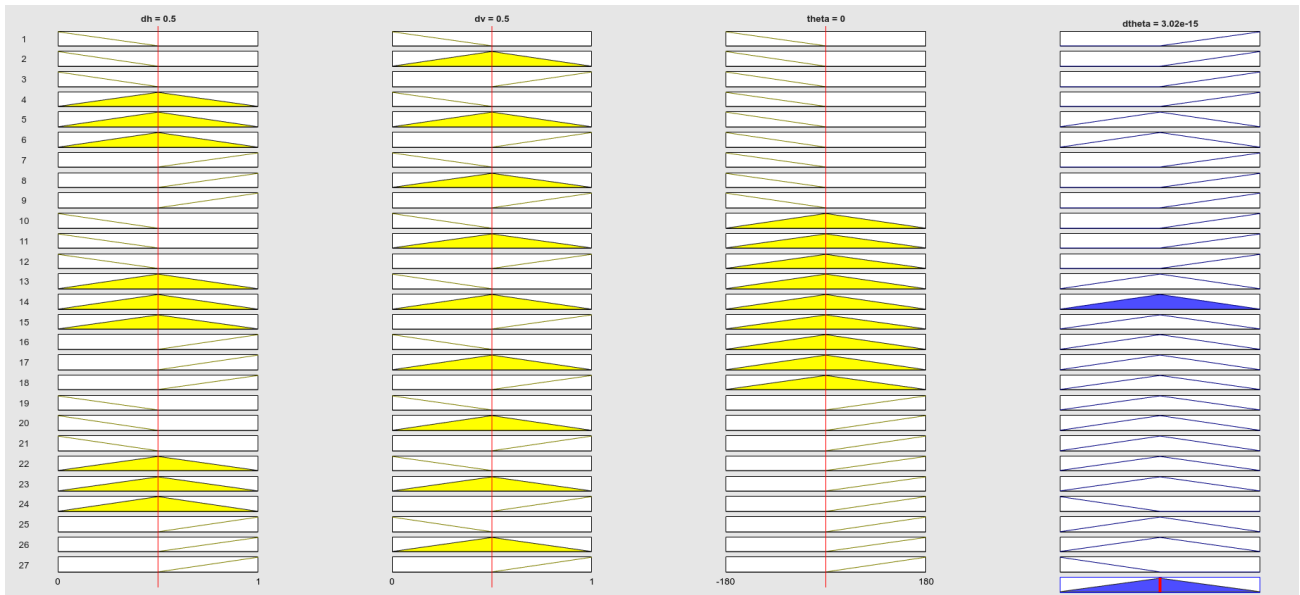
- **Εισόδους:** $d_h \in [0, 1]$, $d_v \in [0, 1]$, $\theta \in [-180, 180]$ (μοίρες).
- **Έξοδο:** $\Delta\theta \in [-130, 130]$ (μοίρες).

Για d_h, d_v χρησιμοποιούνται τριγωνικές MF $\{S, M, L\}$, ενώ για θ και $\Delta\theta$ οι MF $\{N, ZE, P\}$ όπως δίνονται στην εκφώνηση. Οι τελεστές είναι:

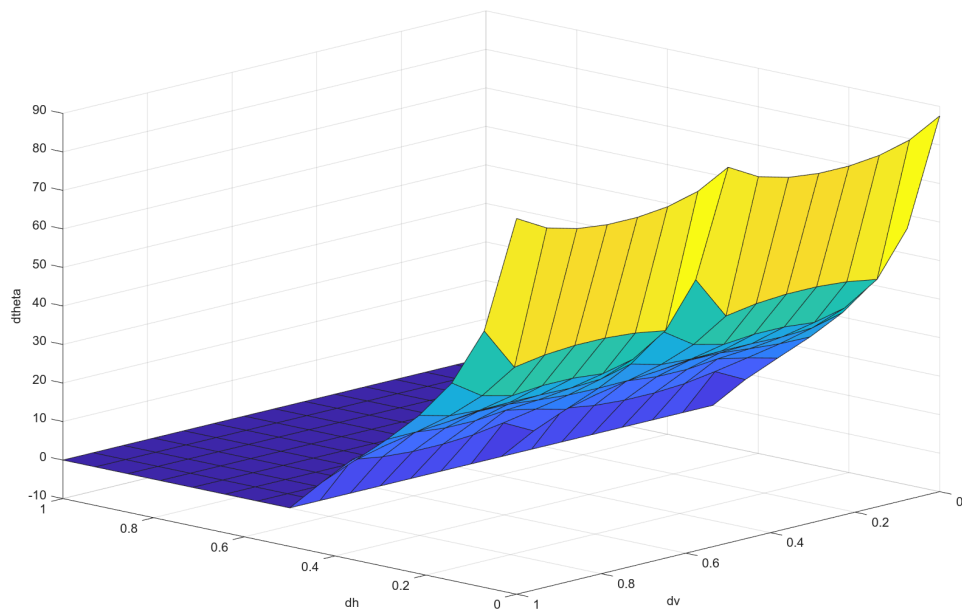
$$\text{AND} = \min, \quad \text{OR} = \max, \quad \text{Implication} = \min, \quad \text{Aggregation} = \max, \quad \text{Defuzzification} = \text{centroid}.$$

Το `.fis` δημιουργήθηκε με το *Fuzzy Logic Designer* και αποθηκεύτηκε ως `CarFLC.fis`. Στο σχήμα 2 φαίνεται ένα στιγμιότυπο με τους κανόνες που χρησιμοποιήθηκαν για τον ελεγκτή. Δυστυχώς, από την προηγούμενη εργασία μέχρι και αυτή, δεν προλάβαμε να γίνουμε ειδικοί, επομένως οι κανόνες βασίζονται και πάλι στη δική μας "φτωχή" προσέγγιση.

Στο σχήμα 3 βλέπουμε και την επιφάνεια των κανόνων ως συνάρτηση των εισόδων των αποστάσεων.



Σχήμα 2: Ενδεικτικό στιγμιότυπο κανόνων.



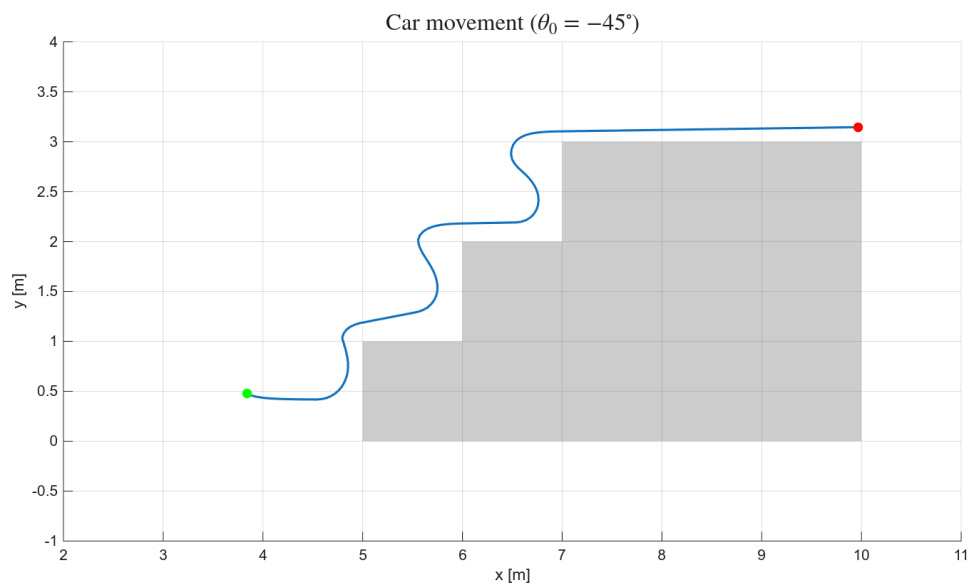
Σχήμα 3: Ενδεικτική επιφάνεια βάσης κανόνων ($\Delta\theta$ ως συνάρτηση δύο εισόδων).

2.4. runSimulation

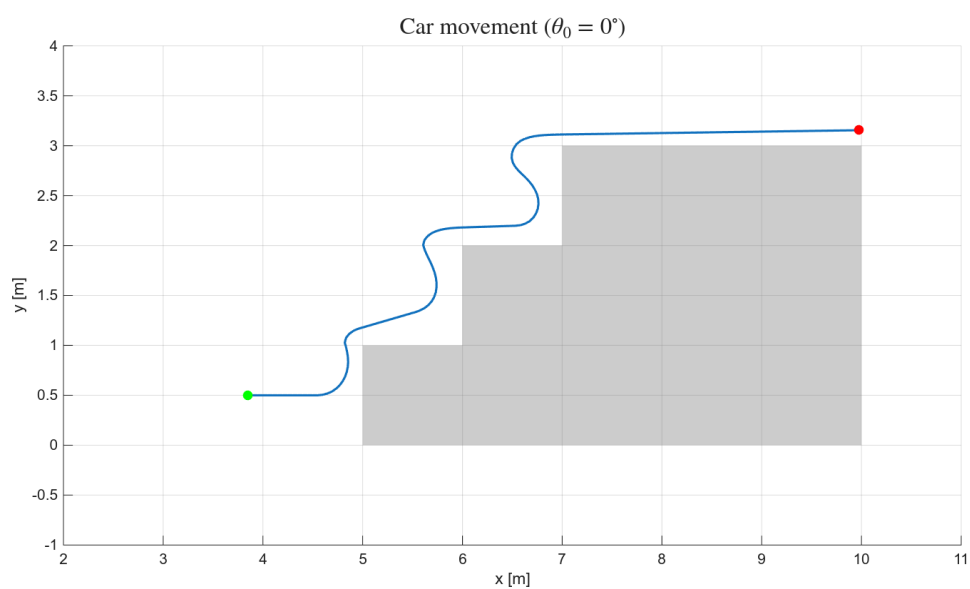
Το κύριο script εκτελεί την προσομοίωση: για κάθε χρονικό βήμα καλεί διαδοχικά `wallDistance` $\rightarrow$ `evalfis` $\rightarrow$ `moveCar`, αποθηκεύοντας την τροχιά. Η διαδικασία επαναλαμβάνεται για τις τρεις αρχικές γωνίες $\theta_0 \in \{-45^\circ, 0^\circ, 45^\circ\}$. Στην δική μας έκδοση το όχημα εκκινεί από τη θέση $(x_{init}, y_{init}) = (3.8, 0.5)$. Η εκτέλεση τερματίζει είτε σε σύγκρουση (οριακή τιμή d_H ή $d_V \leq 0$) είτε όταν το όχημα φτάσει στο $(x_d, y_d) = (10, 3.2)$.

3. ΖΗΤΟΥΜΕΝΑ

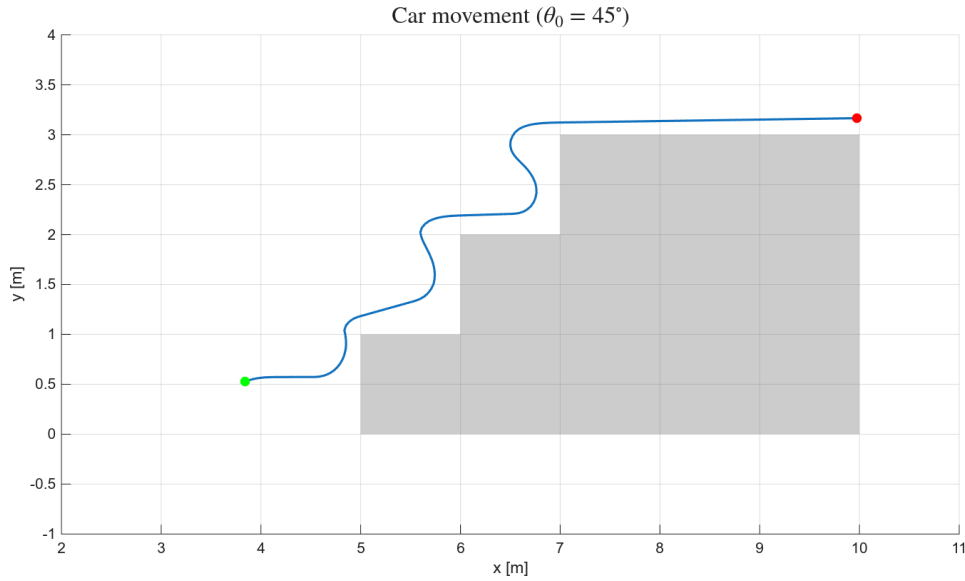
Τα ζητούμενα (τροχιές για διαφορετικές αρχικές γωνίες) παράγονται αυτόματα από το `runSimulation` και αποθηκεύονται ως εικόνες. Παρακάτω παρατίθενται οι τροχιές από την εκτέλεση του script.



Σχήμα 4: Πορεία οχήματος για αρχική γωνία $\theta_0 = -45^\circ$.



Σχήμα 5: Πορεία οχήματος για αρχική γωνία $\theta_0 = 0^\circ$.



Σχήμα 6: Πορεία οχήματος για αρχική γωνία $\theta_0 = 45^\circ$.

3.1. Σχολιασμός

Η λειτουργία του ελεγκτή είναι *σταθερή και ομαλή*: παρατηρούνται ομαλές καμπύλες χωρίς απότομες γωνίες, καθώς οι MF επικάλυψης εξομαλύνουν τις μεταβάσεις κανόνων. Η σύγκλιση προς το επιθυμητό x_d επιτυγχάνεται και στις τρεις περιπτώσεις, με την τροχιά να ακολουθεί πρακτικά τα εμπόδια.

Ωστόσο, η συμπεριφορά δεν είναι ιδανική για δύο λόγους:

1. Ο ελεγκτής **δεν** λαμβάνει άμεση πληροφορία για το στόχο (x_d, y_d) . Βασίζεται μόνο στις αποστάσεις από εμπόδια και στη γωνία, άρα η κάθετη διόρθωση (y) δεν “οδηγείται” ρητά από το σφάλμα έως τον στόχο.
2. Οι αποστάσεις d_h, d_v **ψαλιδισμένες** στο $[0, 1]$ περιορίζουν τη δυναμική περιοχή (lossy αισθητηριακή πληροφορία σε μεγάλες αποστάσεις), κάτι που επιβαρύνει την προσέγγιση ειδικά σε ύψος.

Παρά τους περιορισμούς, ο FLC επιτυγχάνει αποφυγή εμποδίων και καλή ευθυγράμμιση με χαμηλό υπολογιστικό κόστος και υψηλή ερμηνευσιμότητα κανόνων.

4. ΣΥΜΠΕΡΑΣΜΑΤΑ

Αναπτύξαμε έναν Mamdani FLC για το πρόβλημα *Car Control* (Σειρά 3) με εισόδους (d_h, d_v, θ) και έξοδο $\Delta\theta$. Ο ελεγκτής, με κανόνες τύπου IF-THEN και COA αποασαφοποίηση, κατεύθυνε το όχημα με ομαλές τροχιές, αποφεύγοντας τα εμπόδια και επιτυγχάνοντας τη μετάβαση στο επιθυμητό επίπεδο. **Στα θετικά**, είχαμε την ομαλότητα ελέγχου, ερμηνευσιμότητα κανόνων, επαναληψιμότητα σε διαφορετικές αρχικές γωνίες, απλή παραμετροποίηση. **Στα αρνητικά**, είχαμε την έλλειψη ρητής πληροφορίας στόχου (x_d, y_d) , περιορισμένη δυναμική πληροφορία αποστάσεων λόγω κορεσμού $[0, 1]$.

Θεωρούμε πως το σύστημα θα βελτιωνόταν αρκετά με:

- Προσθήκη επιπλέον εισόδου με σφάλμα προς τον στόχο (πχ. $e_x = x_d - x$, $e_y = y_d - y$),
- Ανακλιμάκωση ή προσαρμοστικό κορεσμό των αποστάσεων,
- Χρήση υβριδικού ελεγκτής (FLC για αποφυγή εμποδίων + PD/TSK για καθοδήγηση στον στόχο).