



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ -
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Ρομποτική II: Ευφυή Ρομποτικά Συστήματα

8ο εξάμηνο, Ακαδημαϊκή περίοδος 2024-2025

Εξαμηνιαία Εργασία 2 – Μέρος Α

Αυτοκινούμενα ρομπότ: Παρακολούθηση εμποδίου
(Mobile robots: Wall following)

Φώτιος Κούτσικος : 03121082

Άγγελος Ευστρατίου : 03121113

1. Εισαγωγή

Στην παρούσα εργασία υλοποιείται αλγόριθμος παρακολούθησης εμποδίου (wall following) για ένα αυτοκινούμενο ρομπότ διαφορικής οδήγησης. Το ρομπότ έχει μοντελοποιηθεί και προσομοιωθεί στο περιβάλλον Gazebo, ενώ ο έλεγχος πραγματοποιείται με χρήση του Robot Operating System (ROS).

Ο στόχος της εργασίας είναι το ρομπότ να κινείται παράλληλα σε τοίχους, διατηρώντας μία σταθερή απόσταση και ακολουθώντας μία καθορισμένη φορά (ωρολογιακή ή αντι-ωρολογιακή). Η υλοποίηση περιλαμβάνει την ενσωμάτωση αισθητήρων υπερήχων (sonar) και IMU (δεν χρησιμοποιήθηκαν για την συγκεκριμένη εργασία), καθώς και τον σχεδιασμό ενός ελεγκτή που λαμβάνει υπόψη τα αισθητηριακά δεδομένα ώστε να κατευθύνει το ρομπότ με κατάλληλες ταχύτητες.

2. Θεωρητική Ανάλυση

2.1. Περιγραφή Συστήματος και Ρομποτικής Διάταξης

Η ρομποτική διάταξη που χρησιμοποιείται αποτελείται από ένα κινητό ρομπότ διαφορικής οδήγησης με δύο κινητήριους τροχούς διαμέτρου 20 cm. Το ρομπότ διαθέτει τοπικό σύστημα συντεταγμένων συμμετρικό ως προς το επίπεδο xz, ενώ οι βασικές γεωμετρικές του διαστάσεις δίνονται ως:

- $L_1 = 0.018$ m (μετατόπιση διαγώνιων αισθητήρων)
- $L_2 = 0.05$ m («πάχος» τροχού)
- $L_3 = 0.1$ m (απόσταση αισθητήρων από το κέντρο)
- $L_4 = 0.2$ m (διάμετρος τροχών και πλάτος ρομπότ)

Το σύστημα διαθέτει πέντε αισθητήρες υπερήχων (sonar) τοποθετημένους στις θέσεις: Αριστερά (L), Μπροστά Αριστερά (FL), Μπροστά (F), Μπροστά Δεξιά (FR), και Δεξιά (R) καθώς και ένα αισθητήρα IMU (Inertial Measurement Unit) 9 βαθμών ελευθερίας, ο οποίος παρέχει μετρήσεις προσανατολισμού, γωνιακής ταχύτητας και γραμμικής επιτάχυνσης.

Το ρομπότ περιβάλλεται από τοίχους – εμποδία που δημιουργούν ένα τραπεζοειδές σχήμα και καλείται να κινηθεί παράλληλα σε αυτά, διατηρώντας σταθερή απόσταση και εκτελώντας (τουλάχιστον) μία πλήρη περιφορά.

2.2. Στρατηγική Αλγορίθμου Παρακολούθησης Τοίχου

Ο αλγόριθμος που αναπτύχθηκε στοχεύει στο να κατευθύνει το ρομπότ ώστε να κινείται παράλληλα στα τοιχώματα, διατηρώντας μια σταθερή επιθυμητή απόσταση, η οποία ορίστηκε στα 0.5 m. Η στρατηγική βασίζεται στον υπολογισμό της επιθυμητής κατεύθυνσης μετακίνησης του ρομπότ με βάση την τοπική γεωμετρία του περιβάλλοντος, όπως αυτή προκύπτει από τις μετρήσεις των αισθητήρων sonar.

Αντίληψη περιβάλλοντος μέσω αισθητήρων

Σε κάθε χρονικό βήμα, λαμβάνονται πέντε αποστάσεις από τους αισθητήρες sonar: F, FL, FR, L, R. Οι αποστάσεις αυτές αντιστοιχούν στα σημεία όπου ανιχνεύονται εμπόδια στο περιβάλλον, από την τοπική «οπτική» του ρομπότ.

Οι μετρήσεις αυτές μετατρέπονται από πολικές σε καρτεσιανές συντεταγμένες, με κέντρο το ρομπότ, ώστε να δημιουργηθεί ένα τοπικό σύνολο σημείων:

$$\vec{p}_i = \begin{bmatrix} r_i \cos \theta_i \\ r_i \sin \theta_i \end{bmatrix}, \quad i \in \{F, FL, FR, L, R\}$$

Όπου από τη γεωμετρία της διάταξης είναι:

$$\theta_i = \{ F : 0, FL : \frac{\pi}{4}, FR : -\frac{\pi}{4}, L : \frac{\pi}{2}, R : -\frac{\pi}{2} \}$$

Τα πέντε αυτά σημεία ενώνονται ανά δύο διαδοχικά, σχηματίζοντας τέσσερα ευθύγραμμα τμήματα, τα οποία αναπαριστούν μία εικονική απεικόνιση των τοίχων, όπως τους αντιλαμβάνεται το ρομπότ για τη δεδομένη στιγμή. Πρόκειται δηλαδή για τοπική χαρτογράφηση εμποδίων, βασισμένη στις άμεσες μετρήσεις.

Εύρεση Στόχου και Σφάλματος

Από το σύνολο των τεσσάρων ευθυγράμμων τμημάτων, το ρομπότ εντοπίζει εκείνο που απέχει την ελάχιστη απόσταση από το κέντρο του και υπολογίζει το αντίστοιχο διάνυσμα.

Έτσι, προκύπτει το σφάλμα απόστασης:

$$\varepsilon = d_{desired} - ||\vec{v}_{min}||$$

Όπου \vec{v}_{min} είναι το διάνυσμα προς το πλησιέστερο σημείο, $d_{desired}$ είναι η επιθυμητή απόσταση του ρομπότ από τα τοιχώματα (0.5 m)

Καθορισμός Επιθυμητής Πορείας

Το επιθυμητό μονοπάτι που θα ακολουθήσει το ρομπότ προκύπτει από τη σύνθεση δύο διανυσμάτων:

- Διάνυσμα διόρθωσης σφάλματος: Αν το ρομπότ δε βρίσκεται πάνω στην επιθυμητή πορεία που ορίζεται από την ελάχιστη απόσταση από τα τοιχώματα, το διάνυσμα αυτό τείνει να το επαναφέρει στη σωστή θέση. Πιο συγκεκριμένα, αν το ρομπότ είναι πιο μακριά από όσο πρέπει, αυτό το διάνυσμα κατευθύνεται προς τον τοίχο, ενώ αν είναι πολύ κοντά, κατευθύνεται προς τα έξω.
- Διάνυσμα παράλληλης κίνησης: Προκύπτει από την περιστροφή του \vec{v}_{min} κατά $+90^\circ$ (για ωρολογιακή κίνηση) ή -90° (για αντι-ωρολογιακή), ώστε το ρομπότ να διατηρεί πορεία παράλληλη με το εμπόδιο. Στη συγκεκριμένη περίπτωση, εφόσον τα τελευταία ψηφία του αριθμού μητρώου των συγγραφέων – φοιτητών είναι $2 + 3 = 5$, άρα περιττός αριθμός, τότε θα εκτελέσει αντι-ωρολογιακή κίνηση (CCW – Counter Clockwise).

Τα δύο διανύσματα συνδυάζονται με βάρος που εξαρτάται από το τετράγωνο του σφάλματος:

$$\vec{v}_{des} = \varepsilon^2 \cdot \vec{v}_{correction} + \vec{v}_{parallel}$$

Με αυτόν τον τρόπο, δημιουργείται η κατεύθυνση προς την οποία το ρομπότ θα κινηθεί: όταν βρίσκεται ήδη στην επιθυμητή απόσταση, το βάρος της διόρθωσης μηδενίζεται και συνεχίζει παράλληλα. Αν όμως πλησιάσει ή απομακρυνθεί, ενεργοποιείται αντίστοιχα το διάνυσμα διόρθωσης για να επαναφέρει τη σωστή απόσταση.

Εφαρμογή Ελέγχου Κίνησης

Αφού υπολογιστεί η επιθυμητή κατεύθυνση (διάνυσμα \vec{v}_{des}) το ρομπότ μετατρέπει αυτό το διάνυσμα σε γραμμική ταχύτητα και γωνιακή ταχύτητα χρησιμοποιώντας ένα αναλογικό ελεγκτή για τη ρύθμιση της γωνιακής ταχύτητας με βάση το σφάλμα μεταξύ της τρέχουσας και της επιθυμητής κατεύθυνσης. Η γραμμική ταχύτητα περιορίζεται μέσω ενός ελέγχου προσανατολισμού για να διατηρείται ο έλεγχος του ρομπότ, ενώ η γωνιακή ταχύτητα ελέγχεται με βάση το σφάλμα της γωνίας του ρομπότ.

Συγκεκριμένα, εφαρμόζεται P ελεγκτής για τη γωνιακή ταχύτητα :

$$\omega = -K_p \cdot e_\theta$$

Όπου e_θ είναι το γωνιακό σφάλμα, δηλαδή η γωνία μεταξύ επιθυμητού διανύσματος κατεύθυνσης και της κατεύθυνσης του ρομπότ.

Η γραμμική ταχύτητα μεταβάλλεται με βάση το σφάλμα γωνίας, προκειμένου να δίνεται προτεραιότητα στην απόκτηση σωστής κατεύθυνσης. Με αυτό τον τρόπο, όσο μεγαλύτερο το σφάλμα, τόσο μικρότερη η ταχύτητα εμπρόσθιας κίνησης, ώστε να αποφευχθεί απότομη αλλαγή πορείας. Θεωρώντας και έναν παράγοντα επιβράδυνσης λ του ρομπότ, η γραμμική ταχύτητα είναι η εξής:

$$u = u_{min} + \frac{1}{1 + \lambda e_{\theta}^2} (u_{max} - u_{min})$$

Όπου έχουν οριστεί ως μέγιστη και ελάχιστη ταχύτητα του ρομπότ να είναι 0.01 και 0.40 m/s αντίστοιχα.

3. Προσομοίωση

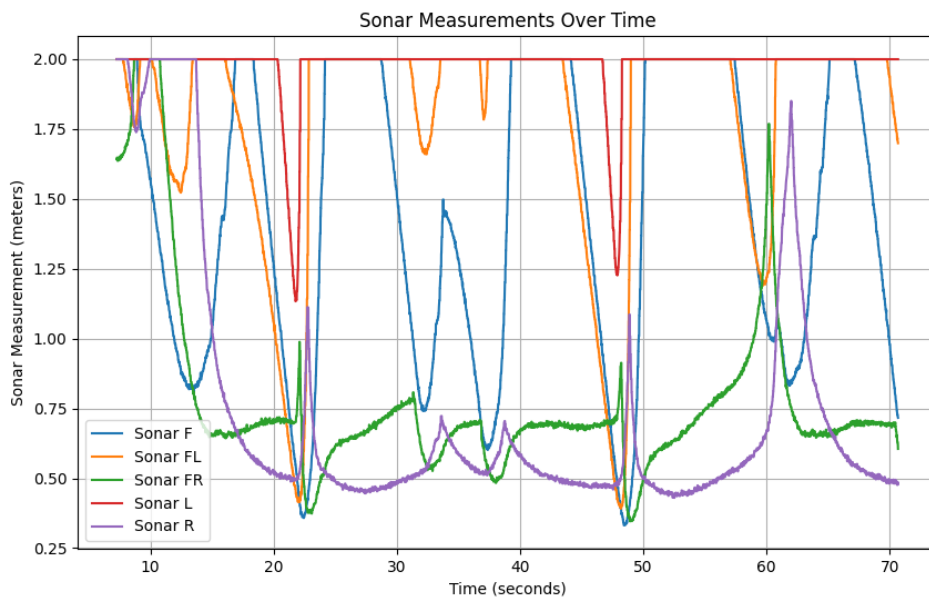
Στα αρχεία κώδικα που παρατίθενται στον φάκελο, έχουν υλοποιηθεί όλοι οι αλγόριθμοι για την πραγματοποίηση της απαιτούμενης εργασίας και η καταγραφή τιμών για διάφορα πειράματα. Συγκεκριμένα, στο αρχείο *kinematics.py* έχει γίνει η κινηματική ανάλυση του κινούμενου ρομπότ διαφορικής οδήγησης, με υπολογισμό ευθέως και αντιστρόφου κινηματικού μοντέλου, ενώ στο αρχείο *controller.py* έχει υλοποιηθεί ο αλγόριθμος παρακολούθησης εμποδίου, με τη μεθοδολογία που περιγράψαμε στο θεωρητικό μέρος της αναφοράς. Χρήσιμες συναρτήσεις για την εκτέλεση της εργασίας βρίσκονται στο αρχείο *utils.py*, ενώ υλοποιήσεις των logs και των plots στα *logger.py* και *plotter.py*.

Η εκτέλεση των προγραμμάτων ελέγχου του ρομπότ έγινε στο περιβάλλον ROS (Robot Operating System) σε συνεργασία με το περιβάλλον προσομοίωσης Gazebo.

Στο συγκεκριμένο πείραμα που θα σχολιάσουμε, χρησιμοποιήσαμε τις εξής παραμέτρους:

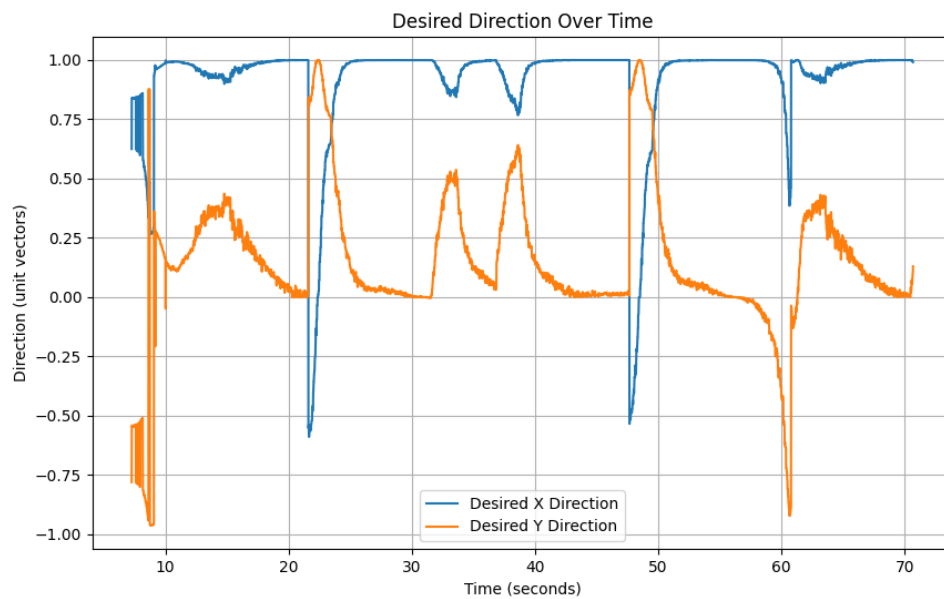
Παράμετρος	Τιμή
d_{min} (ελάχιστη απόσταση)	0.5 m
Κίνηση (CW / CCW)	CCW
V_{min}	0.01 m/s
V_{max}	0.4 m/s
K_p (controller gain)	1.0
λ (slow factor)	20.0
error fix weight	16

Ξεκινάμε με την αναπαράσταση των τιμών των αισθητήρων sonar με το χρόνο, το οποίο δείχνει τις διακυμάνσεις των αποστάσεων από κάθε τοίχο ανά πάσα στιγμή.



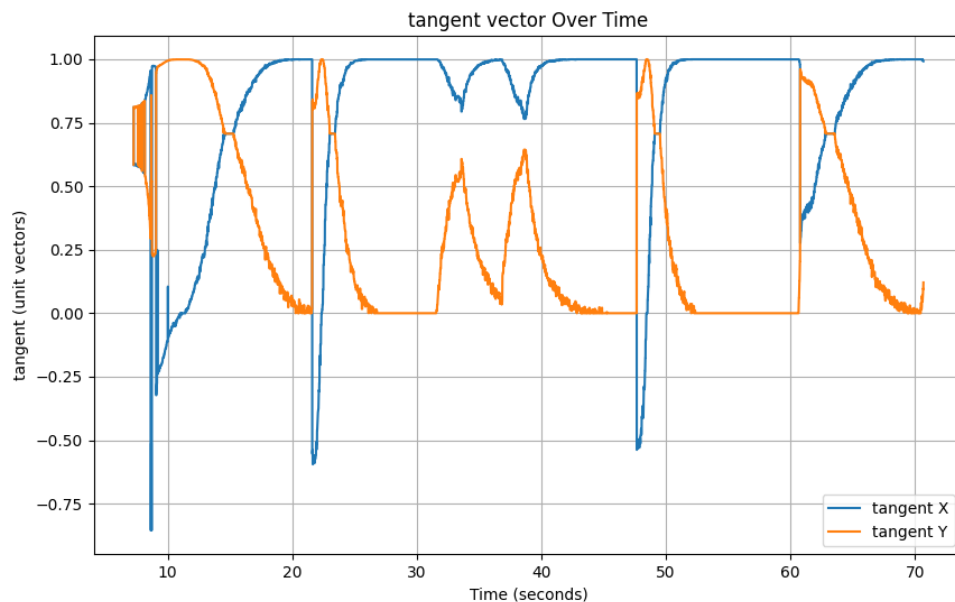
Όπως μπορούμε να δούμε, οι αισθητήρες έχουν ένα μέγιστο εύρος στα 2 m, όπου μπορούν να διακρίνουν κάποιο εμπόδιο και η ελάχιστη απόσταση από τοίχο συνολικά βρίσκεται σε $\sim 0.3 - 0.35$ m. Αυτό σημαίνει ότι ο αλγόριθμος δουλεύει σωστά και τηρεί την επιθυμητή απόσταση, διορθώνοντας όταν πρέπει. Επίσης, δεδομένου ότι το ρομπότ κινείται με CCW φορά, μπορούμε να διακρίνουμε ότι οι αισθητήρες με φορά προς τα αριστερά είναι αυτοί που παρουσιάζουν τις μικρότερες τιμές, ένα λογικό αποτέλεσμα, γιατί το ρομπότ έχει πάντα το κοντινότερο εμπόδιο δεξιά του και κινείται παράλληλα σε αυτό. Παρατηρούμε επίσης πως ο μπροστινός αισθητήρας έχει αυξομειώσεις, που σηματοδοτούν την άφιξη του ρομπότ σε μία γωνία και την αλλαγή κατεύθυνσής του. Η αναπαράσταση των τιμών του FL αισθητήρα έχει δύο μεγάλες πτώσεις που εξηγούνται από την άφιξη του ρομπότ σε μία εκ των δύο οξείων γωνιών στο τραπεζοειδές σχήμα που δημιουργούν τα τοιχώματα μεταξύ τους. Καταλαβαίνουμε ότι το ρομπότ εκτελεί σωστά την απαιτούμενη εργασία, επισκέπτεται όλες τις γωνίες που σχηματίζουν τα εμπόδια μεταξύ τους και κινείται παράλληλα στα τοιχώματα.

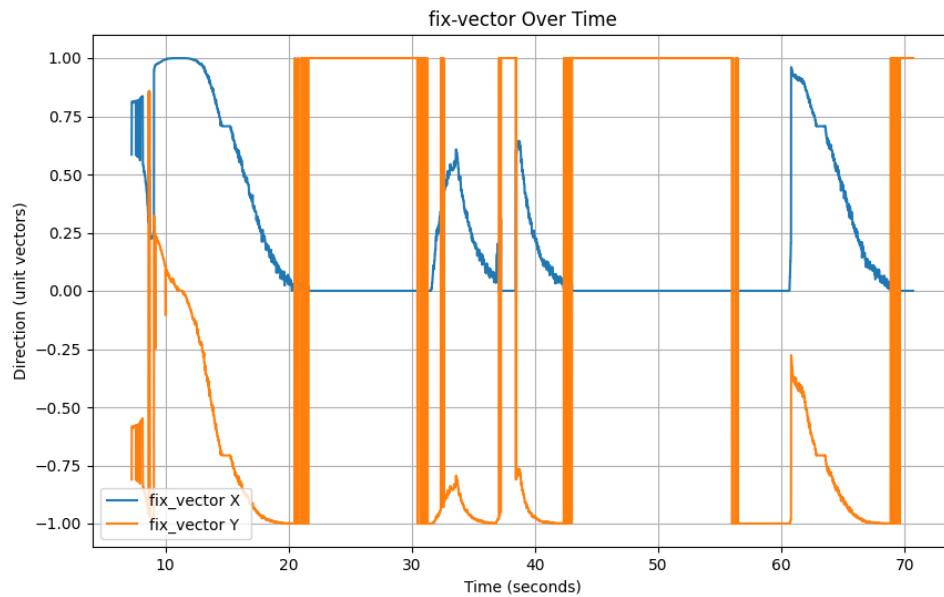
Στη συνέχεια, η αναπαράσταση της επιθυμητής κατεύθυνσης (στον άξονα X και Y) δείχνει τη στρατηγική του ρομπότ για τη διατήρηση της παράλληλης πορείας του προς τα εμπόδια. Συγκεκριμένα, όπως εξηγήθηκε και στο θεωρητικό μέρος, με βάση τα σήματα των αισθητήρων sonar, το ρομπότ αποφασίζει ποια είναι η κατάλληλη κατεύθυνση για κάθε στιγμή, προκειμένου να κινηθεί παράλληλα στον κοντινότερο τοίχο. Το ρομπότ πραγματοποιεί συνεχείς προσαρμογές για να ευθυγραμμιστεί σωστά, όπως φαίνεται και στο επόμενο διάγραμμα.



Όπως μπορούμε να διακρίνουμε, το ρομπότ όταν συναντάει γωνία τοίχου αποφασίζει ότι πρέπει να αλλάξει κατεύθυνση. Θεωρούμε ως κατεύθυνση του ρομπότ το $[1, 0]$ και γι' αυτό το λόγο παρατηρούμε τις τιμές των X και Y να διατηρούνται κατά κύριο λόγο σταθερές σε αυτές τις τιμές.

Τώρα, βλέπουμε τα δύο διανύσματα υπεύθυνα για την τήρηση της επιθυμητής πορείας και κατεύθυνσης του ρομπότ.

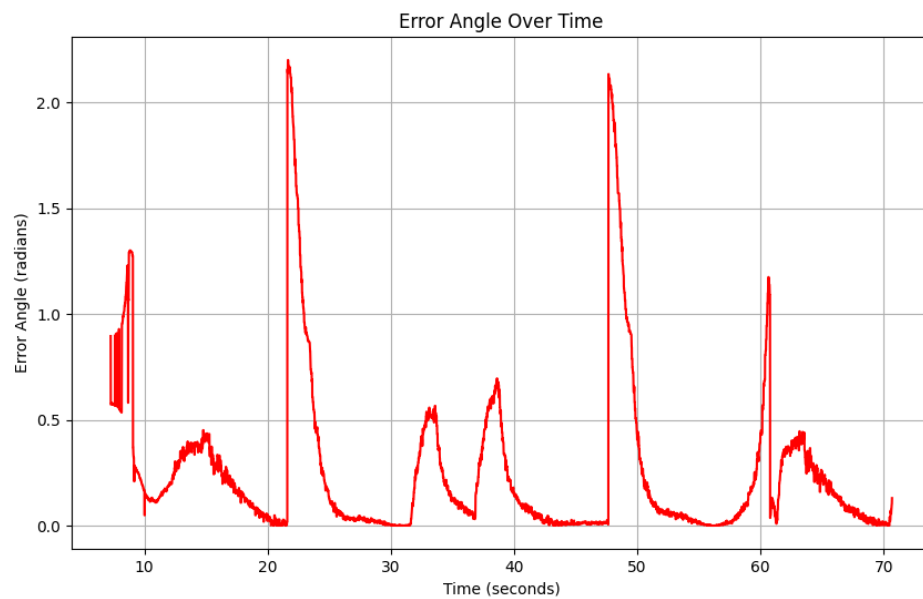




Παρατηρούμε τον τρόπο με τον οποίο τα δύο διανύσματα συνεργάζονται προκειμένου να επιτύχει το ρομπότ επιθυμητή πορεία. Αναλυτικότερα, το fix vector είναι το διάνυσμα που αντιστοιχεί στη διόρθωση του σφάλματος από την επιθυμητή απόσταση και πορεία και, όπως φαίνεται από το διάγραμμα, παρουσιάζει μεγάλες αυξομειώσεις που σηματοδοτούν ότι το ρομπότ πρέπει να κάνει έντονες διορθώσεις στην κατεύθυνσή του, προκειμένου να διατηρήσει επιθυμητή απόσταση. Αυτό συμβαίνει διότι το ρομπότ, όταν διαφέρει σημαντικά από την επιθυμητή πορεία ή απόσταση, πρέπει να κάνει μεγάλες περιστροφές ή να επαναπροσανατολιστεί γρήγορα. Αυτές οι μεγάλες μεταβολές καταγράφονται ως απότομες αλλαγές στην κατεύθυνση του fix vector. Αντίστοιχα για το tangent vector, δηλαδή το διάνυσμα εφαπτόμενο στην επιθυμητή πορεία, φαίνεται να υπάρχουν μικρότερες διακυμάνσεις, γεγονός που υποδηλώνει ότι το ρομπότ προσπαθεί να διατηρήσει την πορεία του χωρίς μεγάλες αλλαγές, ακολουθώντας τη γραμμή του τοίχου.

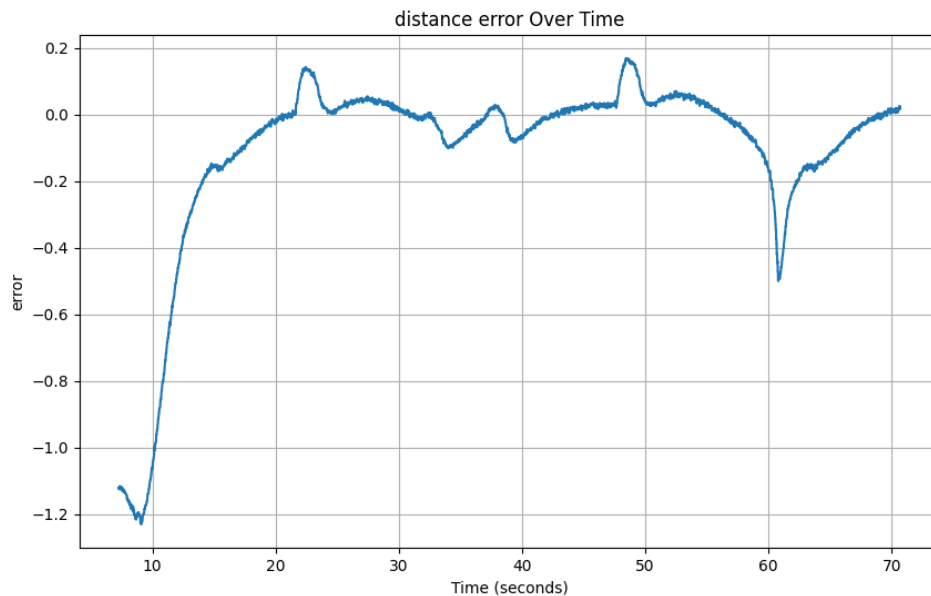
Τα δύο αυτά διανύσματα προσφέρουν στην κίνηση του ρομπότ προκειμένου να κινείται προς τη σωστή κατεύθυνση και διατηρώντας την επιθυμητή απόσταση από τα εμπόδια. Το tangent vector έχει σταθερή προσφορά στην κίνηση του ρομπότ, έτσι ώστε να εξασφαλιστεί πάντα η κίνηση του «προς τα εμπρός», παράλληλα δηλαδή στα τοιχώματα. Το fix vector έχει συμβολή στην κίνηση ανάλογη της ανάγκης του ρομπότ για βελτίωση της κατεύθυνσης. Δηλαδή, όσο μεγαλύτερη είναι η απόκλιση της γωνίας του ρομπότ από την επιθυμητή γωνία, τόσο πιο έντονη θα είναι και η διόρθωση της κατεύθυνσης. Έτσι, υπάρχει μεγάλη εξάρτηση της κίνησης από το σφάλμα γωνίας (error angle) και, ακριβέστερα, το τετράγωνο αυτής.

Η αναπαράσταση του error angle στο χρόνο είναι η εξής:



Όπως μπορούμε να δούμε, το σφάλμα γωνίας είναι μεγάλο στις στροφές της πορείας του ρομπότ και έτσι, στις δεδομένες χρονικές στιγμές που παρουσιάζονται τα spikes, το ρομπότ θα δώσει μεγάλη προτεραιότητα στην αλλαγή κατεύθυνσης.

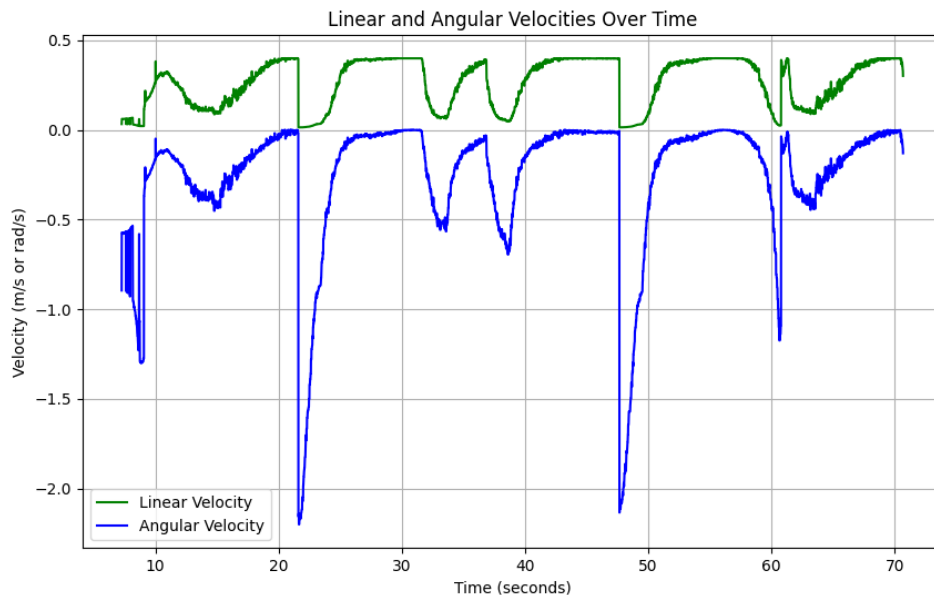
Συνδυάζοντας όλα αυτά, το ρομπότ κινείται αυτόνομα και ακολουθεί την επιθυμητή πορεία με σφάλμα:



Οι αρνητικές τιμές του σφάλματος υποδηλώνουν απόκλιση από την επιθυμητή πορεία, αλλά από την «εσωτερική μεριά», δηλαδή προς το κέντρο του τραπεζοειδούς, ενώ οι θετικές απόκλιση από την «εξωτερική μεριά», αυτή των

τοιχωμάτων. Γι' αυτό το λόγο όπως μπορούμε να διακρίνουμε, το error ξεκινάει από -1.2 m, διότι το ρομπότ ξεκινάει την κίνησή του από το κέντρο της διάταξης και κινείται προς την επιθυμητή πορεία. Όπως βλέπουμε, το ρομπότ αντιδράει σωστά σε αποκλίσεις από την επιθυμητή πορεία που συναντιούνται στις γωνίες και προσαρμόζεται ανάλογα με το σφάλμα. Αξιοσημείωτο είναι παράλληλα ότι σε δύο περιπτώσεις γωνίας το error angle παρουσιάζει καμπύλη προς τα πάνω, ενώ σε άλλες δύο καμπύλη προς τα κάτω. Αυτό το αποτέλεσμα δεν είναι τυχαίο και έχει να κάνει με το είδος της γωνίας, καθώς σε οξείες γωνίες, το ρομπότ «προσπερνάει» το desired path και μετά προσαρμόζεται και γι' αυτό το λόγο είναι καμπύλη προς τα πάνω, δηλαδή πλησίασε τα εμπόδια παραπάνω από το επιθυμητό, ενώ αντιθέτως σε αμβλείες γωνίες «προλαμβάνει» και στρίβει πριν από το ιδανικό και άρα παρουσιάζονται καμπύλες προς τα κάτω.

Τέλος, το διάγραμμα γραμμικής και γωνιακής ταχύτητας στο χρόνο:



Παρατηρούμε ότι η γωνιακή ταχύτητα λαμβάνει μόνο μη θετικές τιμές, καθώς εκτελείται CCW κίνηση και άρα οι περιστροφές είναι αρνητικής γωνίας. Βλέπουμε επίσης στις δύο οξείες γωνίες την ανάγκη για γρήγορη μεταβολή της κατεύθυνσης και επομένως σημειώνεται κατά απόλυτη τιμή η μέγιστη γωνιακή ταχύτητα σε εκείνα τα χρονικά σημεία. Αντίστοιχα, στη γραμμική ταχύτητα παρατηρούμε ότι το ρομπότ προσαρμόζει σωστά την ταχύτητά του ανάλογα με την κατάσταση, επιβραδύνοντας στις στροφές και επιταχύνοντας κατά την ευθύγραμμη κίνηση, τηρώντας τους περιορισμούς ταχύτητας με μέγιστη και ελάχιστη 0.1 και 0.4 m/s.