



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΗΜΜΥ

Ρομποτική II: Ευφυή Ρομποτικά Συστήματα

Ακαδημαϊκό έτος 2022-23

Εξαμηνιαία Εργασία 2

Γιώργος Χαραλάμπους - 03119706

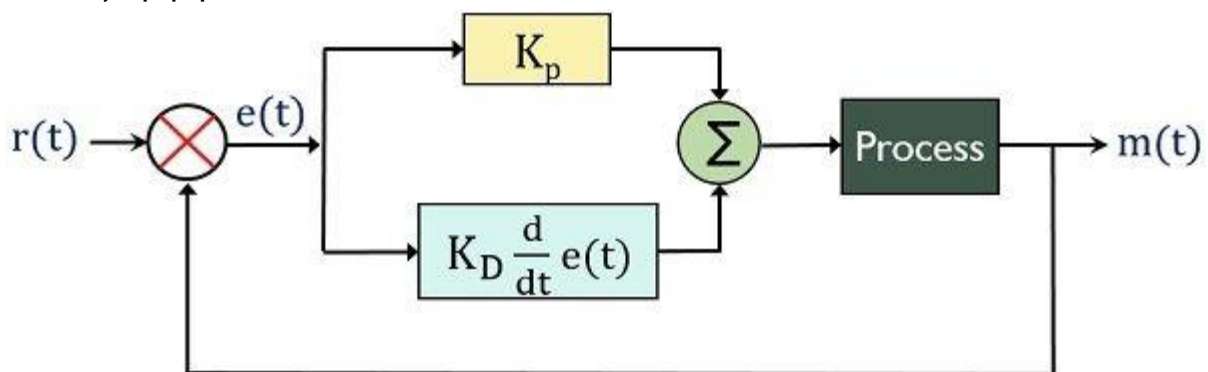
Δωροθέα Κουμίδου - 03119712

A. Θεωρητική Ανάλυση

Σκοπός: Η παρακολούθηση εμποδίου από το αυτοκινούμενο ρομπότ με επενέργηση στη γραμμική και γωνιακή ταχύτητά του και χρήση των διαθέσιμων αισθητηριακών εισόδων.

Με βάση τις τιμές των αισθητήρων (sonars) που λαμβάνουμε επιλέγεται η κατάλληλη ταχύτητα (γραμμική ή γωνιακή) που θα εκτελέσει το ρομπότ.

Για τον υπολογισμό της στιγμιαίας γραμμικής ή γωνιακής ταχύτητας του ρομπότ χρησιμοποιούμε ένα PD-ελεγκτή. Το σύστημα ελέγχου έχει την πιο κάτω μορφή:



Σχήμα 1: Block Diagram PD-Ελεγκτή

Θέτουμε σήμα ελέγχου: $u(t) = K_p * e(t) + K_d \frac{d}{dx} e(t)$

όπου οι σταθερές K_p και K_d αναφέρονται στο αναλογικό και διαφορικό κέρδος αντίστοιχα, οι οποίες καθορίζονται πειραματικά. Το σφάλμα θέσης ορίζεται ως η διαφορά μεταξύ πραγματικής και επιθυμητής θέσης, ενώ η παράγωγος του σφάλματος είναι η διαφορά τρέχοντος και προηγούμενου σφάλματος.

Για τη γραμμική ταχύτητα: $K_p = 1.3$, $K_d = 0.5$

Το σφάλμα είναι η διαφορά μεταξύ θέσης των αισθητήρων και επιθυμητής απόστασης d_0 από τον τοίχο, επομένως η ταχύτητά του ρυθμίζεται με την απόστασή του από τον τοίχο.

Για τη γραμμική ταχύτητα: $K_p = 1.2$, $K_d = 0.4$

Το σφάλμα εδώ είναι η διαφορά της y -συνιστώσας του `sonar_left` και `sonar_frontleft`, δηλαδή ο προσανατολισμός του ρομπότ σε σχέση με τον τοίχο. Όταν το σφάλμα μηδενιστεί το ρομπότ βρίσκεται παράλληλα με τον τοίχο.

Αξίζει να αναφερθεί ότι οι σταθερές K_p , K_d έχουν οριστεί με δοκιμές ανάλογα με το χρόνο μεταβατικής απόκρισης και το μόνιμο σφάλμα που επιθυμούμε. Αυξάνοντας το K_p , μειώνεται το σφάλμα μόνιμης κατάστασης και υπάρχει μικρή αλλαγή στο χρόνο αποκατάστασης. Αυξάνοντας το K_d , μειώνεται ο χρόνος αποκατάστασης και η ποσοστιαία υπερύψωση.

B. Προσομοίωση

Σημειώσεις:

- Βάσει του τελευταίου ψηφίου του αριθμού μητρώου κάθε μέλους της ομάδας μας υπολογίζουμε το $X = 8$.

Έτσι ορίζουμε στο αρχείο **mymobibot_world_wf.launch** αρχικό προσανατολισμό ρομπότ:

$$angle = \text{mod}(8, \pi) \approx 1.716815 \text{ rad}$$

- Η φορά περιστροφής είναι ωρολογιακή (CW), έτσι στο κώδικά μας ασχολούμαστε με τους αισθητήρες *sonar_front*, *sonar_frontleft*, *sonar_left* για να ρυθμίσουμε την απόσταση από τον τοίχο.
- Επιλέξαμε ως απόσταση ασφαλείας τα **0.3m**.
- Μέγιστη γραμμική ταχύτητα έχουμε ορίσει τα **0.3m/s**.
- Μέγιστη γωνιακή ταχύτητα έχουμε ορίσει τα **0.5rad/s**.

Σύντομη επεξήγηση κώδικα (αρχείο «*follower.py*»)

Για την υλοποίηση της Θεωρητικής Ανάλυσης χωρίζουμε τη κίνηση του ρομπότ σε 2 φάσεις. Η μετάβαση από τη μία φάση στη άλλη γίνεται μέσω μιας boolean μεταβλητής *correction_flag* .

1) Φάση 1 (Μεταφορική κίνηση):

Στη κατάσταση αυτή βρίσκεται το ρομπότ όταν δεν ανιχνεύει εμπόδιο σε απόσταση μικρότερη από την απόσταση ασφαλείας. Το ρομπότ έχει μηδενική γωνιακή ταχύτητα στον άξονα z και η γραμμική του ταχύτητα στον άξονα x καθορίζεται από το σήμα ελέγχου. Ωστόσο, για να υπάρχει ευστάθεια στο σύστημά μας, φροντίζουμε να υπάρχει ένα ανώτατο όριο ταχύτητας στο ρομπότ:

```
control_signal = min( max_linvel ,  $K_p$  * error +  $K_d$  * error_derivative )
```

Όταν το σφάλμα γίνει μικρότερο από μια επιθυμητή τιμή, το ρομπότ έχει πλησιάσει σε τοίχο ή γωνία και πρέπει να διορθώσει τον προσανατολισμό του. Η γραμμική του ταχύτητα μηδενίζεται, η μεταβλητή `correction_flag=True` και γίνεται η μετάβαση στη Φάση 2.

2) Φάση 2 (Περιστροφική κίνηση):

Στη κατάσταση αυτή βρίσκεται το ρομπότ όταν εντοπίσουν οι αισθητήρες του εμπόδιο σε απόσταση ίση με την απόσταση ασφαλείας. Το ρομπότ έχει μηδενική γραμμική ταχύτητα στον άξονα x και η γωνιακή του ταχύτητα στον άξονα z καθορίζεται από το σήμα ελέγχου. Ωστόσο, για να υπάρχει ευστάθεια στο σύστημά μας, φροντίζουμε να υπάρχει ένα ανώτατο όριο ταχύτητας στο ρομπότ:

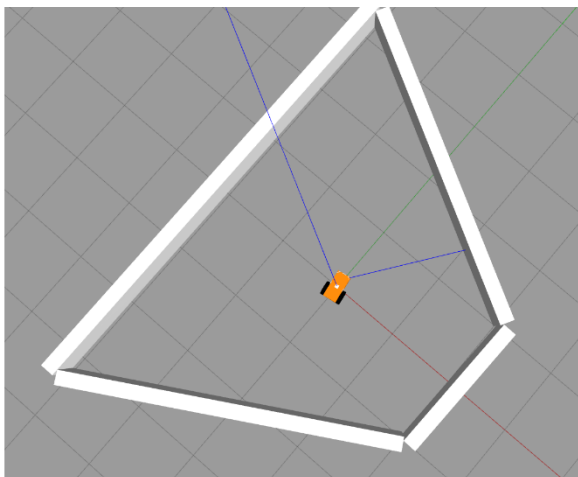
`control_signal = min(max_angvel , Kp2 * error2 + Kd2* error_derivative)`

Όταν το σφάλμα γίνει μικρότερο από μια επιθυμητή τιμή, το ρομπότ έχει διορθώσει τον προσανατολισμό του και βρίσκεται παράλληλα στον τοίχο. Η γωνιακή του ταχύτητα μηδενίζεται, η μεταβλητή `correction_flag=False` και γίνεται η μετάβαση στη Φάση 1 ξανά.

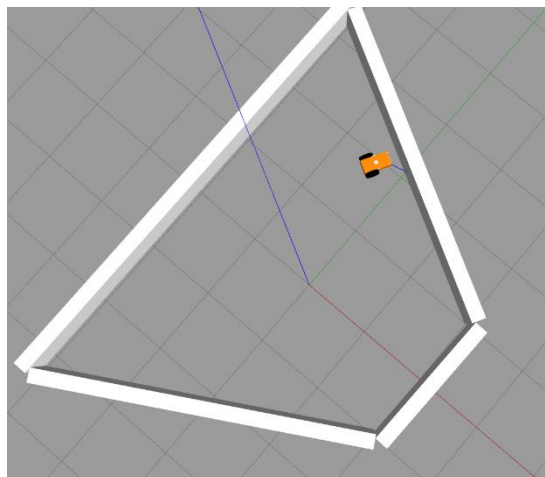
Σχόλια και Παρατηρήσεις:

- ✓ Το σφάλμα για το σήμα ελέγχου στη φάση 1 είναι της τάξης των 5mm, ενώ στη φάση 2 έχουμε σφάλμα της τάξης των 9mm.
- ✓ Η απόσταση από τον τοίχο διατηρείται σταθερή κατά την διάρκεια της μεταφορικής κίνησης του ρομπότ, παρόλα αυτά υπάρχει μια απόκλιση από την απόσταση ασφαλείας κυρίως λόγω των ασυνεχειών στις γωνίες.
- ✓ Όταν η γωνία είναι αρκετά κλειστή λαμβάνονται υπόψη και οι αισθητήρες στη δεξιά πλευρά του ρομπότ για καλύτερο έλεγχο και εντοπισμό.
- ✓ Οι σταθερές K_p και K_d ιδανικά μπορούν να βελτιστοποιηθούν. Επιπλέον για μηδενισμό του μόνιμου σφάλματος, δύναται να τοποθετηθεί PID ελεγκτής. Οι τιμές των K_p , K_d θα πρέπει να διαφοροποιηθούν για να μην πέσει το σύστημα σε αστάθεια.

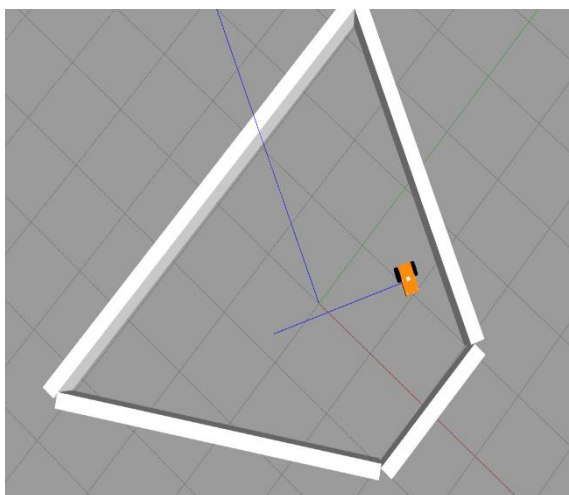
Αποτελέσματα προσομοίωσης:



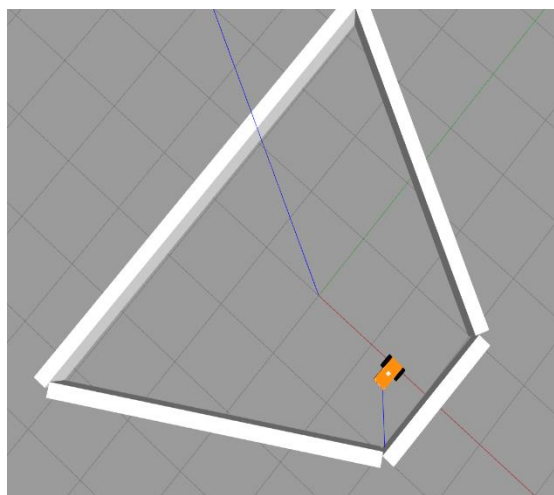
Εικόνα 1



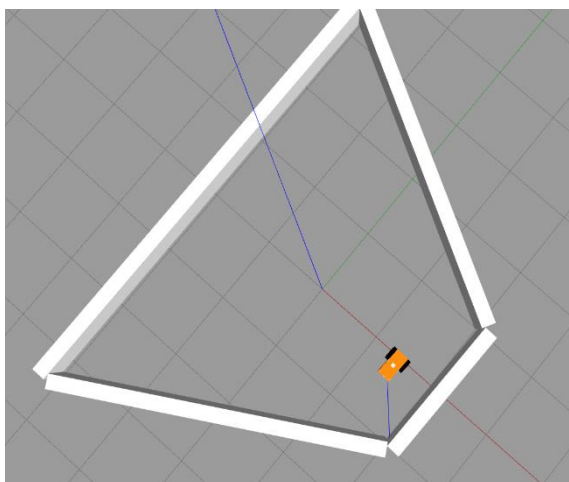
Εικόνα 2



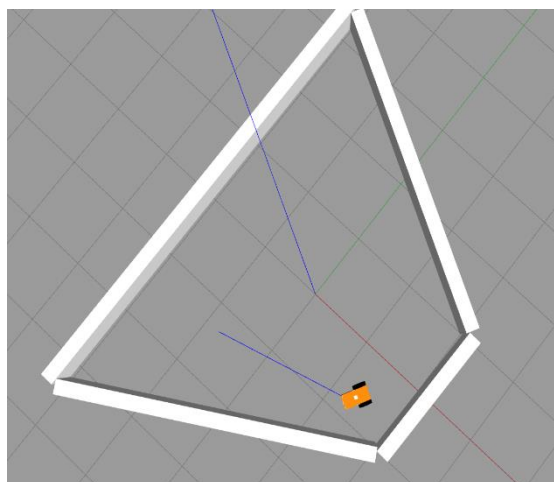
Εικόνα 3



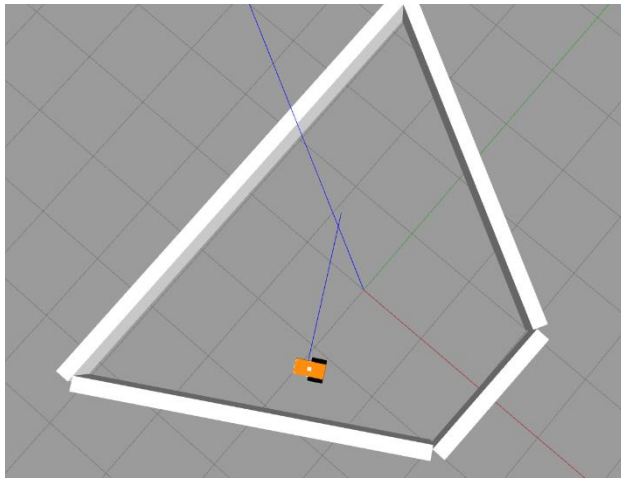
Εικόνα 4



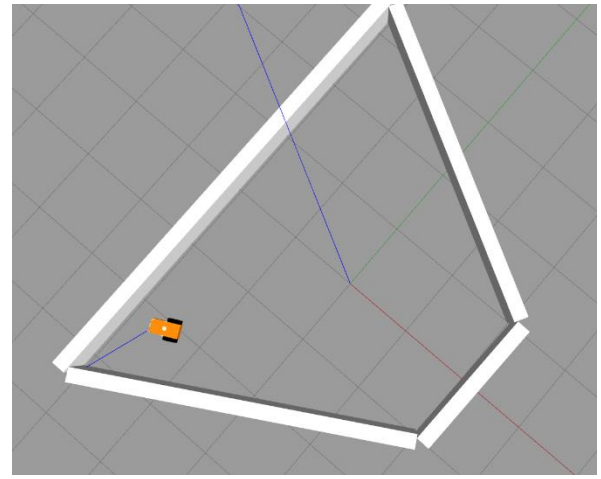
Εικόνα 5



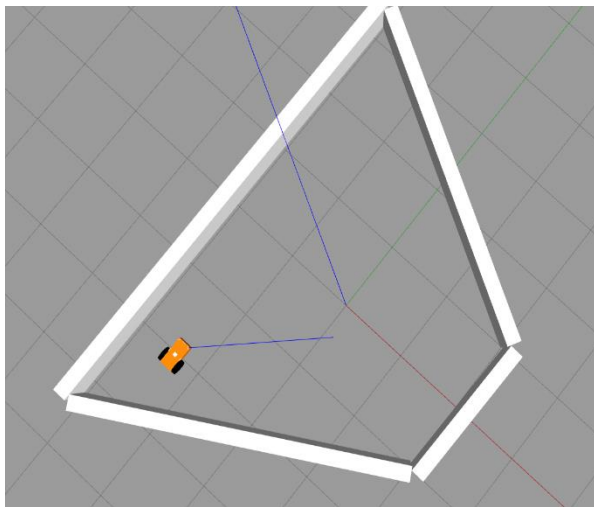
Εικόνα 6



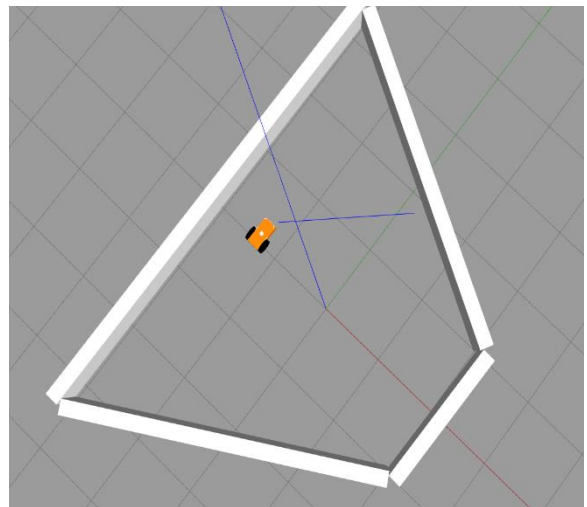
Εικόνα 7



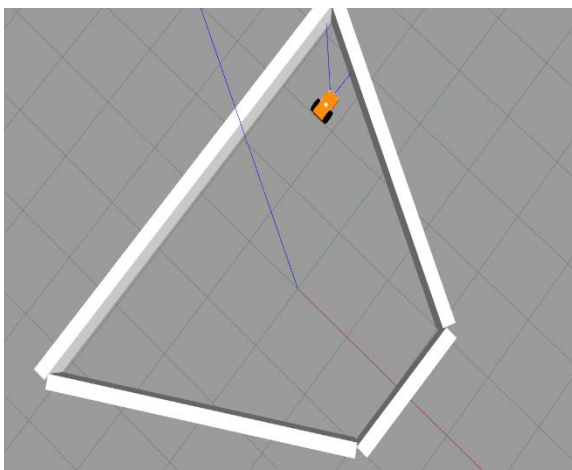
Εικόνα 8



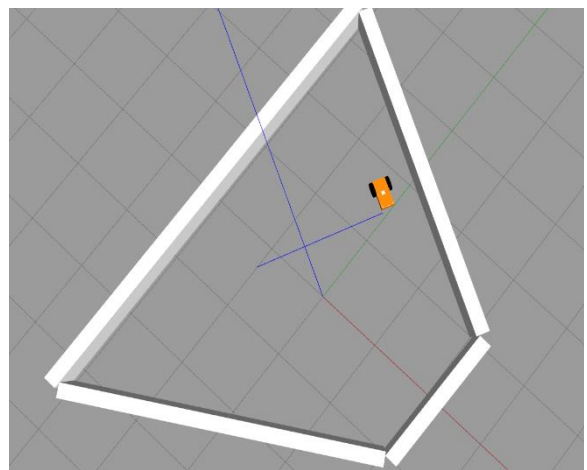
Εικόνα 9



Εικόνα 10



Εικόνα 11



Εικόνα 12