

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΣΗΜΑΤΩΝ ΕΛΕΓΧΟΥ ΚΑΙ ΡΟΜΠΟΤΙΚΗΣ



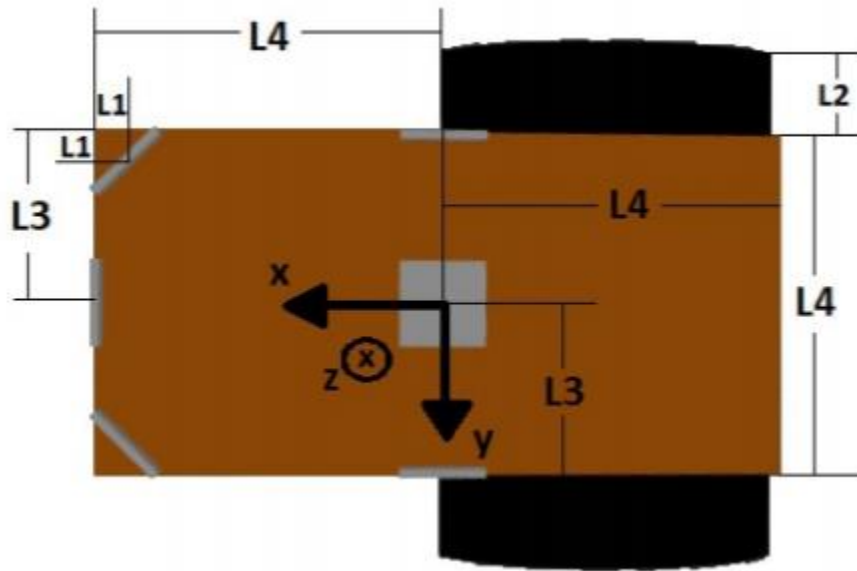
Ρομποτική II: Ευφυή Ρομποτικά Συστήματα Εξαμηνιαία Εργασία 2: Μέρος Α

Στοιχεία Φοιτητών:

Δάσκος Ραφαήλ (03116049)
Ζάρρας Ιωάννης (03116082)

Ομάδα 4

Στο πλαίσιο αυτής της εργαστηριακής άσκησης χρησιμοποιήσαμε ένα ρομπότ διαφορικής οδήγησης (differential-drive) το οποίο φαίνεται στην παρακάτω φωτογραφία.



Οι διαστάσεις που χρησιμοποιήθηκαν για το ρομπότ αυτό είναι οι εξής:

- $L_1 = 0.018$ m
- $L_2 = 0.05$ m
- $L_3 = 0.1$ m
- $L_4 = 0.2$ m
- Διάμετρος τροχών ίση με 20 cm

Ορισμένα από τα παραπάνω δεδομένα θα χρησιμοποιηθούν στον κώδικά μας για τη μελέτη της κίνησης του ρομπότ.

Σκοπός Άσκησης

Σκοπός της άσκησης μας ήταν να υλοποιήσουμε έναν αλγόριθμο παρακολούθησης εμποδίου με στόχο το ρομπότ μας να εκτελέσει τουλάχιστον μία πλήρη περιστροφή γύρω από τους τοίχους τους οποίους είχε το περιβάλλον που δημιουργήσαμε γι' αυτό.

Κατά τη διάρκεια της περιστροφής θέλουμε το ρομπότ να διατηρήσει σταθερή απόσταση από τους τοίχους που υπάρχουν γύρω του.

Α. Θεωρητική Ανάλυση

Στην αρχή δώσαμε στο ρομπότ τον κατάλληλο αρχικό προσανατολισμό ίσο με 3.5 rad όπως μας ζητούταν από την εκφώνηση της άσκησης. Ο προσανατολισμός αυτός βγαίνει έτσι καθώς έχουμε A.M. 03116082 και 03116049 και επομένως $X_1 = 2$, $X_2 = 9$.

Έτσι βγαίνει $X = X_1 + X_2 = 11$. Και $\text{angle} = \text{mod}(X, \pi)$ (in rad) $\approx 3.5014 \dots \text{rad} \approx 3.5 \text{ rad}$

Επιπλέον, μιας και έχουμε περιττό αριθμό X θα πραγματοποιήσουμε αντι-ωρολογιακή φορά (CCW) κατά τη διάρκεια της κίνησής μας.

Γενικά υπάρχουν 3 βασικά μέρη (states) στον αλγόριθμο της κίνησης του ρομπότ μας:

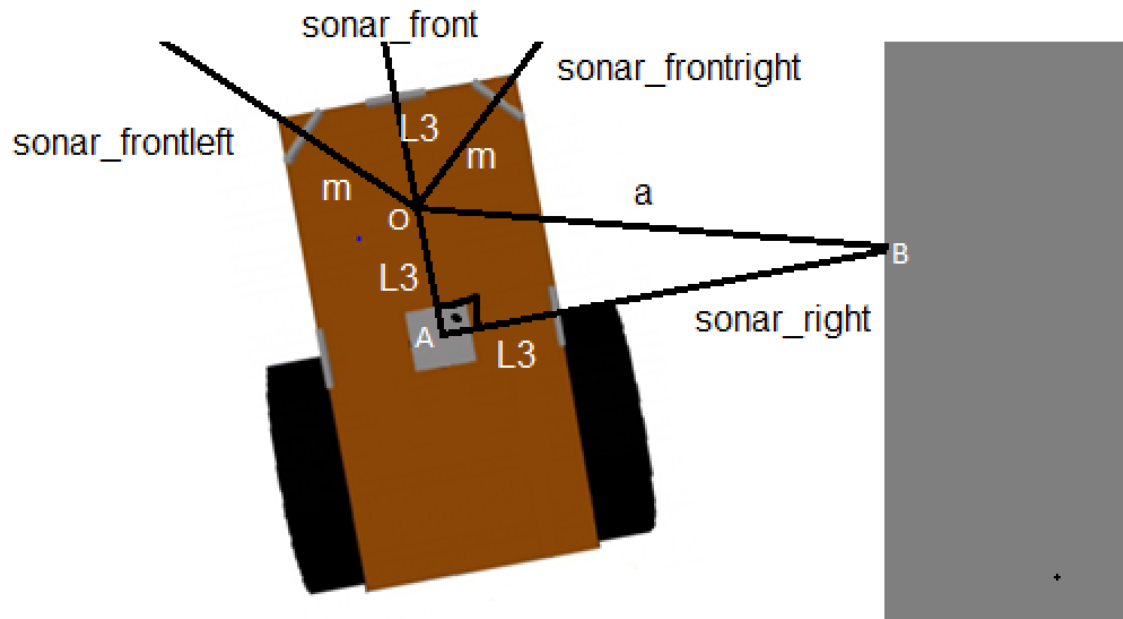
Πρώτα είναι η μεταφορά του ρομπότ μας προς τον τοίχο που «κοιτάει» διατηρώντας γραμμική ταχύτητα σταθερή και ίσης με 0.3. Αυτή η μετακίνηση γίνεται έτσι ώστε να έρθει το ρομπότ αρκετά κοντά με τον τοίχο και στη συνέχεια στο επόμενο μέρος να γυρίσει παράλληλα σε αυτόν και να αρχίσει την περιστροφή γύρω από τα εμπόδια που διακρίνει. Χρειάζεται να πούμε πως αυτή η κίνηση γίνεται μέχρι ένας από τους αισθητήρες να βρει απόσταση ίση με την επιθυμητή από τον τοίχο ώστε στη συνέχεια να συνεχίσει να τη διατηρεί αυτή εφόσον τελικά στρίβει. Για να ισχύσει

Δεύτερο μέρος είναι η στροφή κατά τον άξονα του z αντίθετα με τη φορά του ρολογιού μέχρι να αποκτήσει προσανατολισμό σχεδόν παράλληλο με τον τοίχο, όπως και είναι το επιθυμητό. Αυτό όπως και το πρώτο χρησιμοποιούνται μόνο στην αρχή της προσομοίωσης καθώς από εκεί και πέρα θέλουμε το ρομπότ μας να επιτελέσει ομαλή κίνηση και επομένως θα γίνει χρήση PD ελεγκτή. Αξίζει να αναφέρουμε πως για να το πετύχουμε αυτό χρειαστήκαμε 2 προϋποθέσεις. Ο μπροστά αισθητήρας να διακρίνει μεγάλη απόσταση (σημαίνει πως δεν έχει τοίχο μπροστά του) και η απόσταση από τον αριστερό αισθητήρα που περνά από τον μπροστά δεξιά να είναι μεγαλύτερη από αυτή που περνά από τον δεξιά. Χρησιμοποιήσαμε την απόσταση από εκείνο το σημείο γιατί είναι το σημείο τομής των δύο ευθειών και επίσης είναι αυτός ο τύπος γιατί αυτό συμβαίνει οριακά όταν ξεπεράσουμε τις 90°. Βέβαια χρειάζονται και οι δύο έλεγχοι δεδομένου ότι μπορεί το ρομπότ να έχει έρθει με τέτοιο τρόπο που ήδη η απόσταση που θέλουμε να είναι πιο μεγάλη να μην είναι ή να σταματήσει πολύ πιο νωρίς γιατί κατά την περιστροφή ο μπροστά αισθητήρας το σταματάει πιο γρήγορα γιατί σταματάει να βλέπει τοίχο (όταν ο επόμενος τοίχος είναι σε πολύ μεγάλη απόσταση από το σημείο που στρίβουμε).

Τέλος έχουμε την κίνηση από εκεί και πέρα για το ρομπότ μας η οποία ελέγχεται με έναν PD-ελεγκτή. Αυτός λαμβάνει ως εισόδους ενός σημείου του ρομπότ (θα δείξουμε στη συνέχεια πως παίρνουμε και τα μήκη από αυτό) από τον τοίχο και αναλόγως επηρεάζει τη στροφική ταχύτητά του.

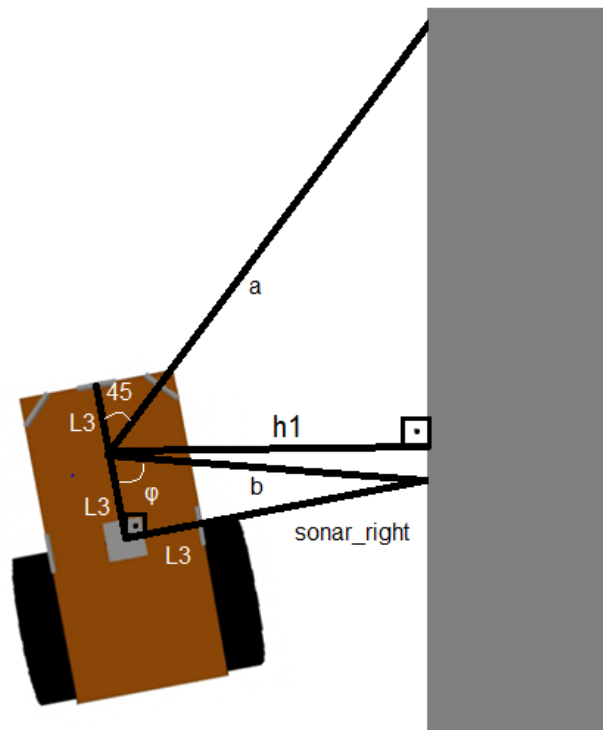
Στην παρακάτω φωτογραφία βλέπουμε όλες τις αποστάσεις των σόναρ μετρημένων από το σημείο O για να μπορούν στη συνέχεια να χρησιμοποιηθούν για τον υπολογισμό όλων των h (ύψη) που θα χρειαστούμε για την ομαλή κίνηση του ρομπότ. Έτσι θα έχουμε αρχικά ότι $m = (L_4 - L_1)$:

- $final_front = sonar_front + L_3$
- $final_frontleft = sonar_frontleft + m$
- $final_frontright = sonar_frontright + m$
- $final_right = a = \sqrt{L_3^2 + (sonar_right + L_3)^2}$ (από το τρίγωνο ABO)



Στην παρακάτω φωτογραφία είναι η απόσταση της δεξιάς πλευράς από τον τοίχο όπου υπολογίζεται με τη χρήση των δεξιά μπροστά και δεξιά αισθητήρων. Το φ το βρίσκουμε από το συνημίτονο της γωνίας στο ορθογώνιο τρίγωνο που βρίσκεται και ξέρουμε ότι και η άλλη γωνία είναι 45° οπότε από τον νόμο των συνημιτόνων και το εμβαδό του τριγώνου που έχει πλευρές a , b και ύψος h_1 . Έτσι:

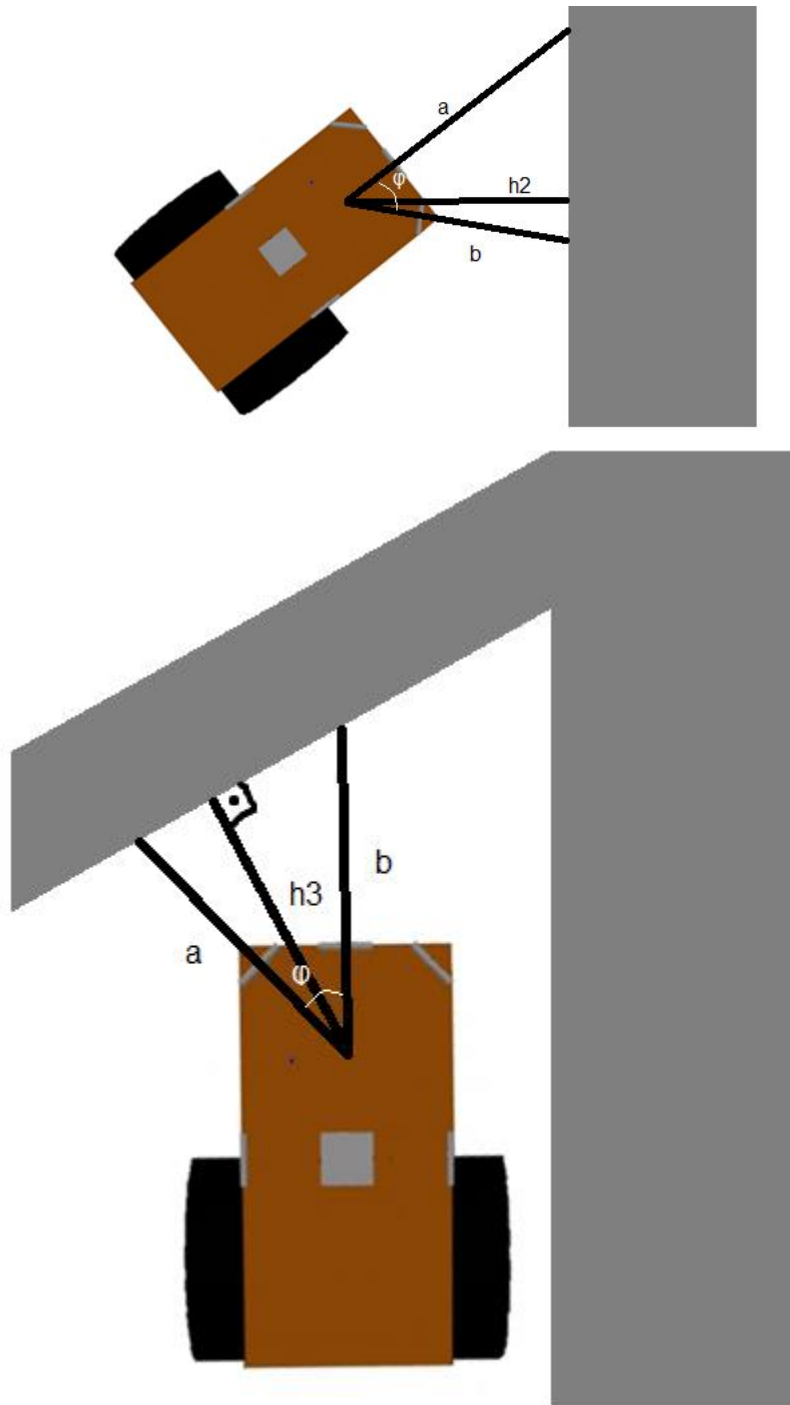
$$h_1 = \frac{a*b*\sin(135^\circ - \varphi)}{\sqrt{a^2 + b^2 - 2*a*b*\cos(135^\circ - \varphi)}}$$



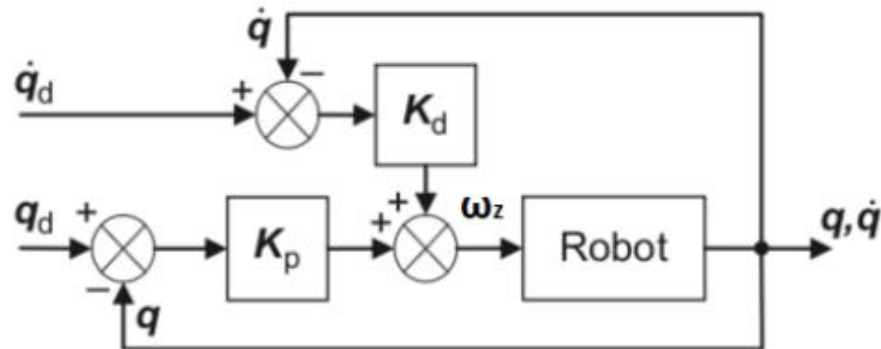
Για τις επόμενες 2 αποστάσεις του ρομπότ βγαίνουν και αυτές από τον τύπο των συνημιτόνων σε συνδυασμό με το εμβαδό με τη χρήση ημιτόνου. Έτσι θα είναι:

$$h_i = \frac{a*b*\sin(\varphi)}{\sqrt{a^2 + b^2 - 2*a*b*\cos(\varphi)}}, i = 2, 3, \varphi = 45^\circ$$

Αυτές οι δύο μετρικές θα χρειαστούν όταν το ρομπότ πλησιάζει γωνίες, τόσο στις αμβλείες όσο και τις οξείες. Θα εξηγήσουμε στη συνέχεια πώς τις χρησιμοποιούμε.



Έχοντας όλα αυτά τα χρησιμοποιούμε στον παρακάτω έλεγχο ο οποίος φαίνεται στο σχήμα:



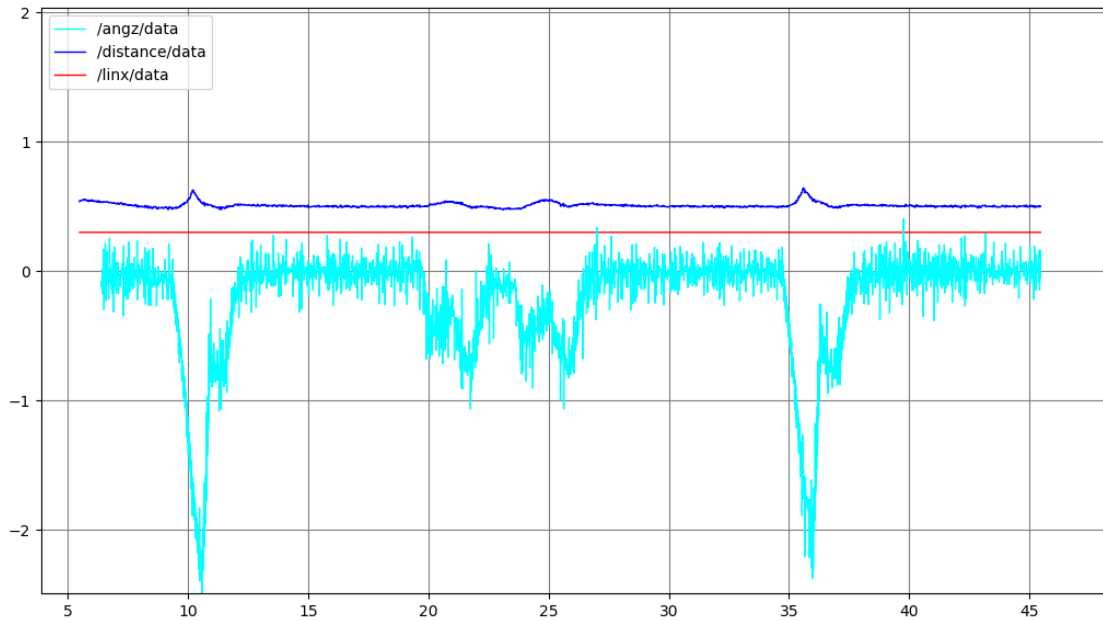
Με αυτόν ελέγχεται το ω που είναι η γωνιακή ταχύτητα κατά ψ του ρομπότ. Μιας και έχουμε γραμμικό έλεγχο η παράγωγος του σφάλματος μπορεί να υπολογιστεί ως $\text{previous_error} - \text{error}$. Όπου το $\text{error} = \min(h_1, h_2 - 0.12, h_3 - 0.4) - \text{desired_distance}$. Και επιλέξαμε επιθυμητή απόσταση ίση με 0.5. Επίσης χρειάζεται να αναφέρουμε πως βάζουμε την αφαίρεση στις τελευταίες δύο μετρικές για να αποκτήσουν μεγαλύτερη βαρύτητα κατά τον έλεγχο. Αυτό χρειάζεται αφού έτσι μπορεί η δεύτερη να βοηθήσει στην ομαλή διατήρηση της απόστασης στις αμβλείες γωνίες και η τρίτη για να βοηθήσει περισσότερο στις οξείες γωνίες καθώς παρατηρήσαμε ότι η κάθε μία εκεί έχει μεγαλύτερη επίδραση η κάθε μία. Αν δεν υπήρχαν αυτές οι δύο, το ρομπότ θα έβλεπε την τελευταία στιγμή τον τοίχο και δε θα προλάβαινε να αντιδράσει ή θα περνούσε οριακά χωρίς να τον χτυπήσει. Βέβαια αναγκαία είναι και η πρώτη που βοηθά να κρατηθεί σταθερή απόσταση από τον τοίχο στην περίπτωση ευθείας (μόνιμη κατάσταση), όταν δηλαδή το ρομπότ δε βρίσκει μπροστά του εμπόδια.

Για τις τιμές του ρυθμιστή υπολογίσαμε ως βέλτιστες τις $K_p = 6$ και $K_d = 14$ για να καταφέρουμε να πετύχουμε τόσο χαμηλό overshoot όσο και καλό χρόνο αποκατάστασης. Το μόνιμο σφάλμα είναι μικρό κατά τη διάρκεια της κίνησης με εξαίρεση τις οξείες γωνίες όπου για να κρατήσει ταχύτητα αλλά και όσο πιο χαμηλή απόσταση μπορεί, οδηγείται στη δημιουργία μιας μικρής ανύψωσης επειδή απομακρύνεται από τον τοίχο. Βλέπουν μεγαλύτερες τιμές οι αισθητήρες εξαιτίας της γωνίας. Αυτό γίνεται λίγο και στις αμβλείες γωνίες καθώς προσπαθεί να μπει σε θέση να είναι σωστή η απόσταση σου τη στιγμή εξόδου από αυτές οπότε και χάνει λίγο από τη στιγμή που τις διαπερνά.

B. Προσομοίωση

Τέλος, θα παρουσιάσουμε ορισμένα διαγράμματα για την απόσταση αλλά και την ταχύτητα του ρομπότ μας για να αποδείξουμε πως είναι καλά τα αποτελέσματά μας. Θα δείξουμε για ποιο λόγο οι τιμές που αναφέρουμε στην ανάλυση χρησιμοποιήθηκαν εν τέλει στο πρόγραμμά μας.

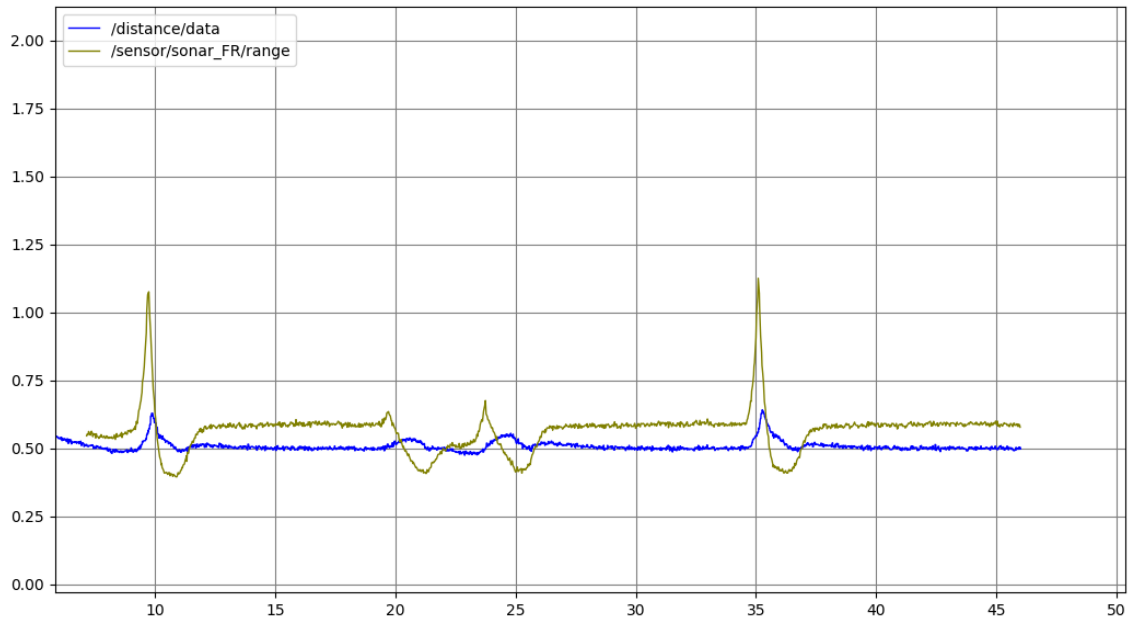
Αρχικά η ομαλή ταχύτητα του ρομπότ από τη στιγμή που αρχίζει την κίνηση του και έπειτα. Ταυτόχρονα με αυτή εμφανίζουμε και τη γωνιακή ταχύτητα αλλά και την απόσταση από τον τοίχο:



Αυτό που μπορούμε να παρατηρήσουμε είναι, όπως είπαμε, ότι έχουμε σταθερή γραμμική ταχύτητα ίση με 0.3 καθόλη τη διάρκεια της κίνησης μας. Από την άλλη για τη γωνιακή ταχύτητα βλέπουμε πως σε όλες τις γωνίες που έχουμε γίνεται αρκετά αρνητική με σκοπό να οδηγηθεί το ρομπότ να στρίψει αριστερόστροφα και να έρθει πάλι παράλληλα με τον τοίχο. Στη συνέχεια κατά τη διάρκεια την σταθερής του κίνησης (στο ενδιάμεσο των τοίχων) βλέπουμε συνεχείς διακυμάνσεις. Αυτό συμβαίνει γιατί προσπαθεί η απόσταση να διατηρηθεί σταθερή αλλά εξαιτίας μικρών σφαλμάτων που έχει από μόνη της η προσομοίωση είναι σχεδόν αδύνατο να βγει ίση με μηδέν. Σχεδόν πάντα χρειάζεται να μεταβεί είτε προς μεγαλύτερη είτε προς μικρότερη τιμή.

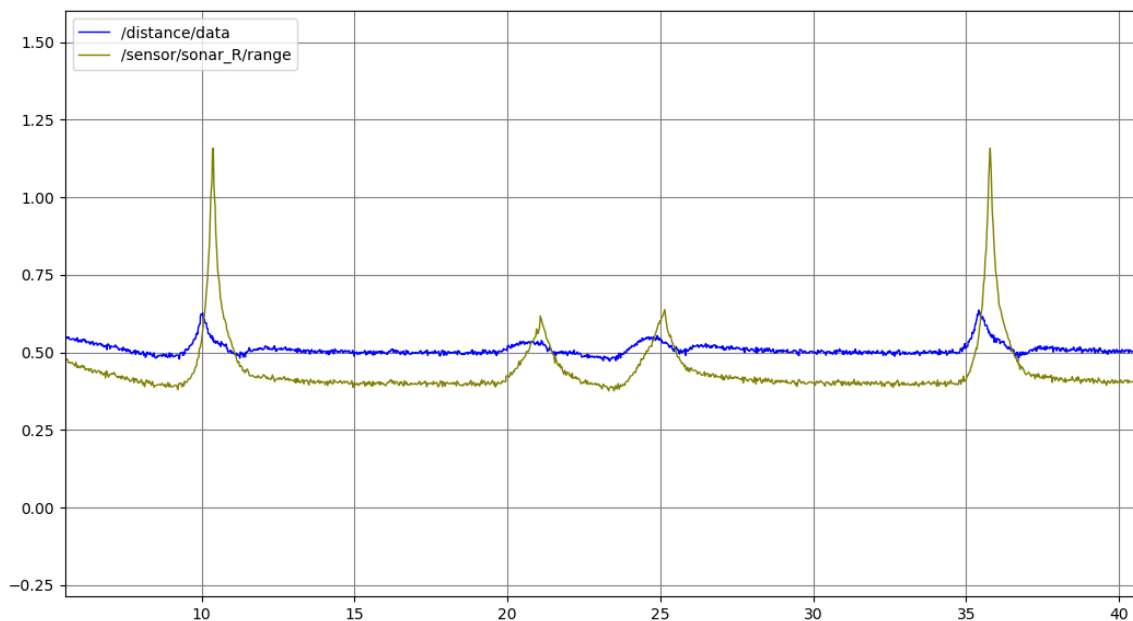
Στη συνέχεια θα παρουσιάσουμε ορισμένους από τους αισθητήρες μας και τον τρόπο με τον οποίο επηρεάζονται κατά την κίνηση μας.

Πρώτα από όλα θα παρουσιάσουμε τον μπροστά δεξιά αισθητήρα:



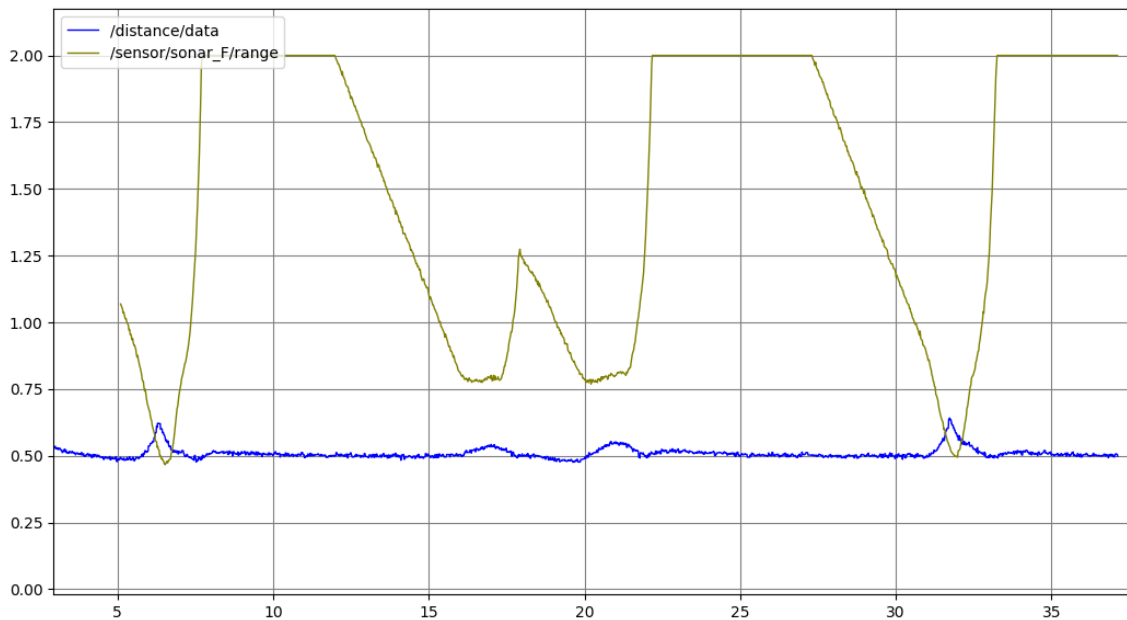
Όπως μπορούμε να παρατηρήσουμε είναι σχεδόν σταθερή η τιμή που έχουμε όταν βρισκόμαστε σε ισορροπία (κίνηση παράλληλα στους τοίχους χωρίς γωνίες). Όμως, εκτός από αυτό βλέπουμε και κάποια spikes να υπάρχουν τα οποία εμφανίζονται όταν στρίβει το ρομπότ μας. βλέπουμε που τη στιγμή που φτάνει στις οξείες γωνίες έχουμε μια αύξηση της τιμής που διαβάζει και αυτό γίνεται επειδή όσο στρίβει βλέπουμε πιο «βαθιά» μέσα στη γωνία ενώ όταν βγαίνει από αυτές είναι πιο μικρή η τιμή καθώς η φορά του ρομπότ είναι με το μπροστά δεξιά μέρος του κοντά στον τοίχο. Το ίδιο περίπου παρατηρούμε και στις αμβλείες γωνίες. Μεγαλύτερη, δηλαδή τιμή τη στιγμή που μπαίνουμε σε αυτές και μικρότερη όταν πάμε να βγούμε.

Έπειτα θα δείξουμε την απόσταση που μετράει ο δεξιός αισθητήρας.

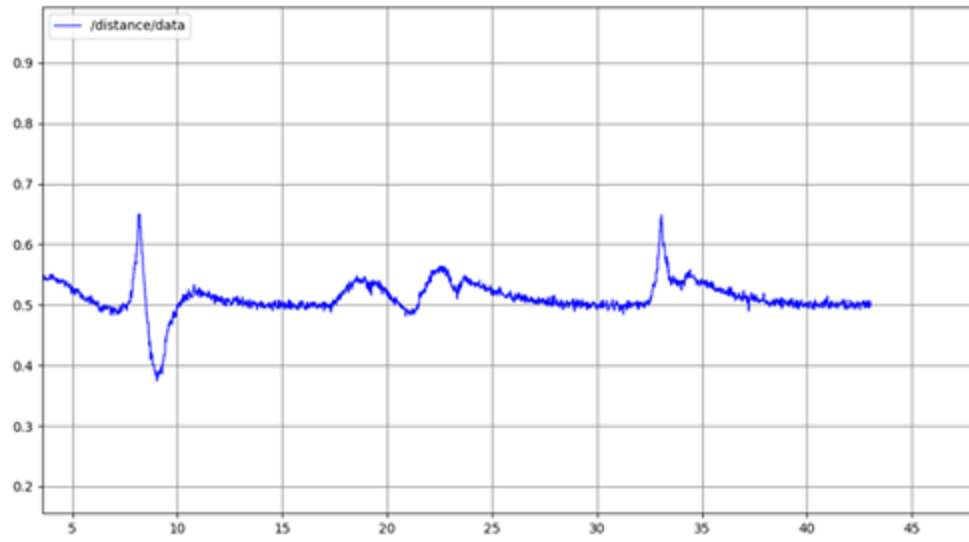


Ο αισθητήρας αυτός βλέπει μια όμοια κίνηση με την απόσταση του ρομπότ από τον τοίχο. Αυτό γίνεται γιατί στο μεγαλύτερο μέρος της κίνησης είναι και αυτός που την επηρεάζει όσο γίνεται. Βέβαια σε αντίθεση με την απόσταση, όταν είμαστε σταθερή έχει τιμή λίγο πιο μικρή από αυτή γιατί η απόσταση ισούται με την τιμή του αισθητήρα + L_3 (επομένως 0.1 μικρότερη) αλλά πάνω στις στροφές παρατηρούμε πιο εκρηκτικές αλλαγές από την απόσταση δεδομένου ότι πλέον δεν την επηρεάζει και το ρομπότ γυρίζει πιο απότομα για να αποκτήσει πάλι απόσταση ίση με 0.5.

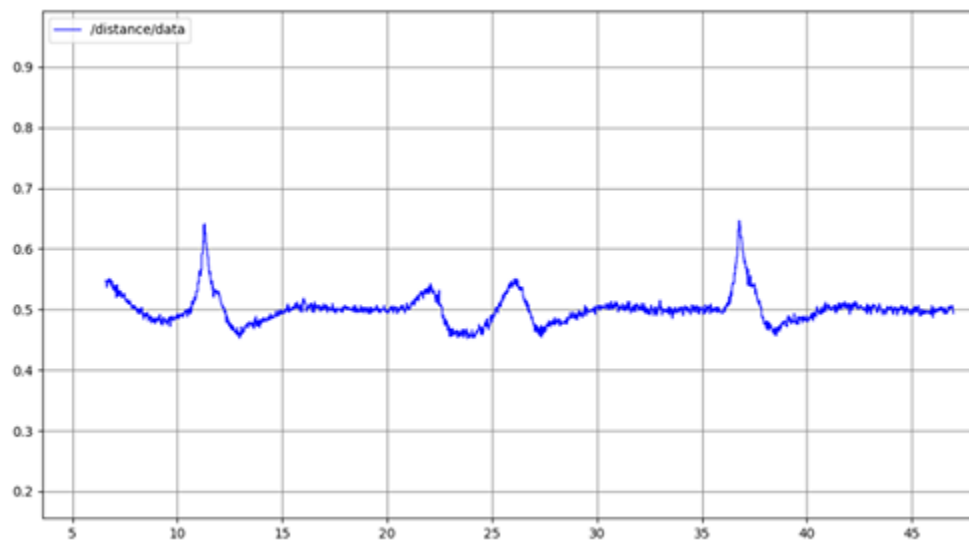
Τέλος, ο μπροστινός αισθητήρας έχει σε όλη τη διάρκεια της κίνησης του ρομπότ παράλληλα στον τοίχο, τιμή ίση με 2 (είναι η μέγιστη μέτρηση που μπορεί να δει το σόναρ μας) ενώ, όταν πλησιάζουμε στις γωνίες, βρίσκουμε μεγάλες πτώσεις μιας και αρχίζει να βλέπει τοίχους μπροστά του. Αυτό μπορούμε να το παρατηρήσουμε στο παρακάτω διάγραμμα:



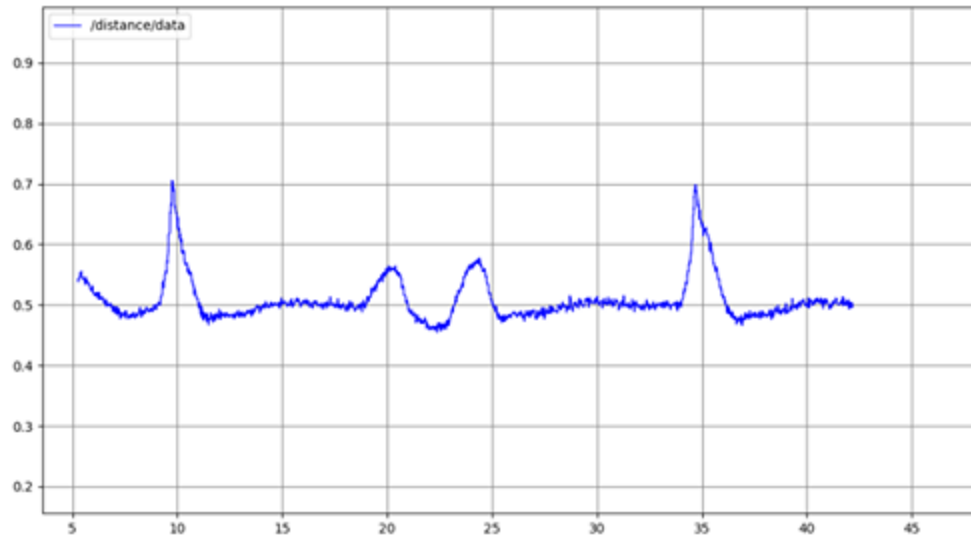
Και τελικά θα παρουσιάσουμε διάφορα διαγράμματα τα οποία θα έχουν αλλαγές στις τιμές των παραμέτρων: K_p , K_d , ευαισθησία h_2 (E_2) και ευαισθησία h_3 (E_3) (ως ευαισθησίες εννοούμε τη μεταβλητή που αφαιρείται από αυτές τις τιμές και δείχνει πόσο γρήγορα θα γίνουν αντιληπτές από τον PD-έλεγχο μας) και θα εξηγήσουμε για ποιο λόγο επιλέξαμε τις τιμές μας.



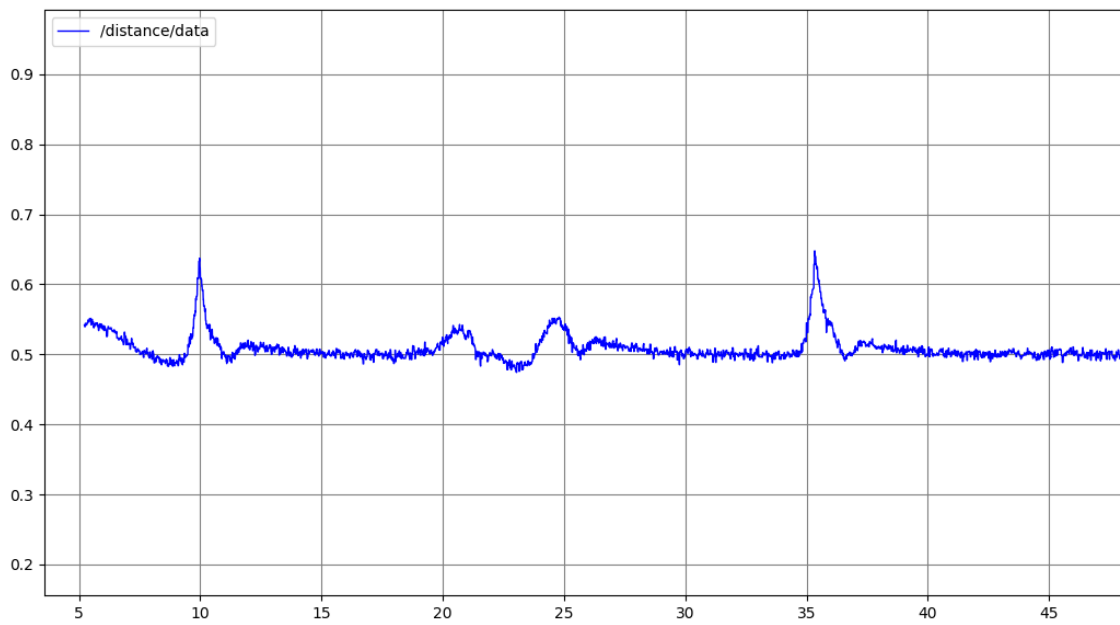
$K_p = 9, K_d = 11, E_2 = 0.12, E_3 = 0.4$



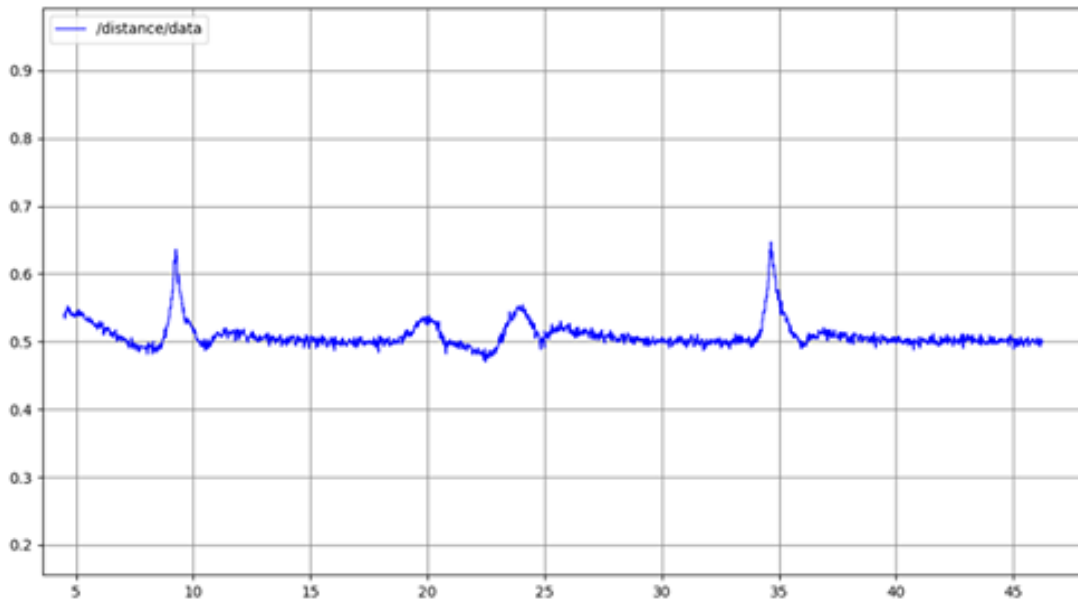
$K_p = 6, K_d = 14, E_2 = 0.07, E_3 = 0.4$



$Kp = 6, Kd = 14, E2 = 0.12, E3 = 0.48$



$Kp = 6, Kd = 14, E2 = 0.12, E3 = 0.4$

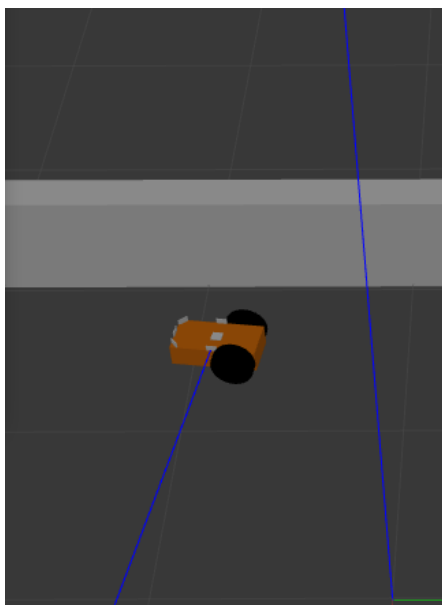


$$Kp = 6, Kd = 16, E2 = 0.12, E3 = 0.4$$

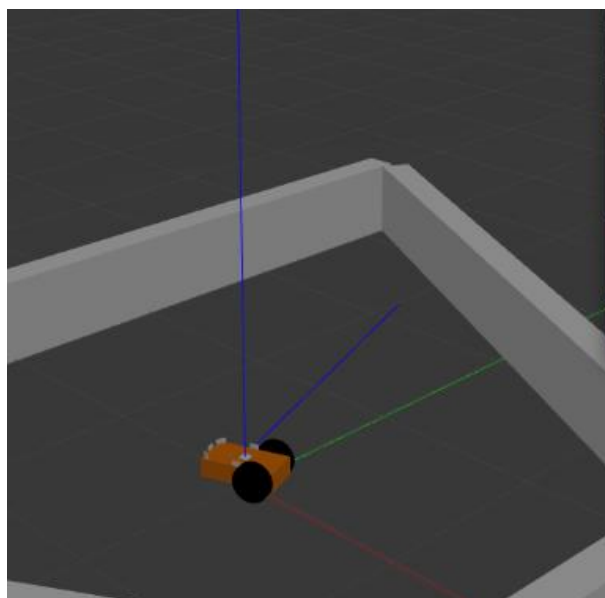
Αξιολόγηση τιμών: παρατηρούμε ότι η μεταβολή του Kd δεν επηρεάζει τόσο πολύ τα αποτελέσματα του ελέγχου μας δεδομένου ότι δε μεταβάλλονται τόσο πολύ οι γραφικές. Το μόνο που μπορούμε να παρατηρήσουμε είναι πως όσο αυξάνεται η τιμή του έχουμε μια μερική εξομάλυνση της γραφικής μας, βέβαια είναι πιο αργά τα spikes, ο χρόνος ανύψωσης επηρεάζεται. Από την άλλη, αύξηση του Kp μας κάνει μεν να πάμε πιο γρήγορα στην τελική μας τιμή αλλά προστίθενται αρκετές ταλαντώσεις στο ενδιαμέσο της κίνησης οπότε και δεν είναι επιθυμητό. Μείωση του E_2 για να επηρεάζει λιγότερο την κίνησή μας αυτή η τιμή οδηγεί να γίνουν πιο αργά οι στροφές και στις αμβλείες και στις οξείες γωνίες που συνεπάγεται να αναγκαστεί το σύστημά μας να απομακρυνθεί από τον τοίχο καθώς θα φτάσει πολύ κοντά σε αυτόν μετά την στροφή. Αύξησή του, από την άλλη, θα μας επηρέαζε η μετρική και στην κίνηση στην ευθεία και δε θα ήταν επιθυμητό το αποτέλεσμα μας. Τέλος, αύξηση του E_3 για να προσπαθήσουμε να αντιμετωπίσουμε ακόμα καλύτερα την απόσταση στις οξείες γωνίες οδηγεί να έχουμε μεγαλύτερα spikes καθώς η ανιχνεύονται πιο νωρίς από ότι πρέπει και στριβεί σε προγενέστερη χρονική στιγμή. Αντιθέτως, μείωσή του θα μας οδηγούσε να τα δούμε πιο αργά και το ρομπότ θα έβρισκε στον τοίχο στην προσπάθεια αποφυγής.

Για τους λόγους που αναφέραμε θα επιλέξουμε και θα χρησιμοποιήσουμε στον κώδικά μας τις τιμές $Kp = 6$, $Kd = 14$, $E_2 = 0.12$, $E_3 = 0.4$.

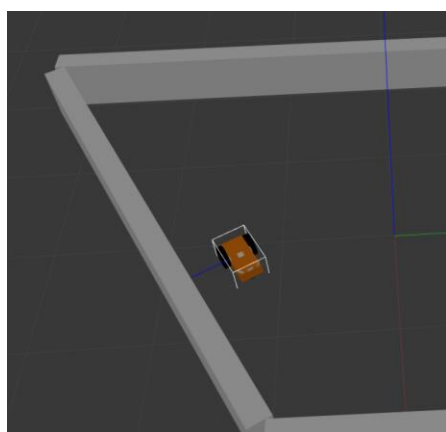
Τέλος θα δείξουμε μερικές εικόνες που λάβαμε από το Gazebo κατά το τρέξιμο του αλγορίθμου παρακολούθησης εμποδίων.



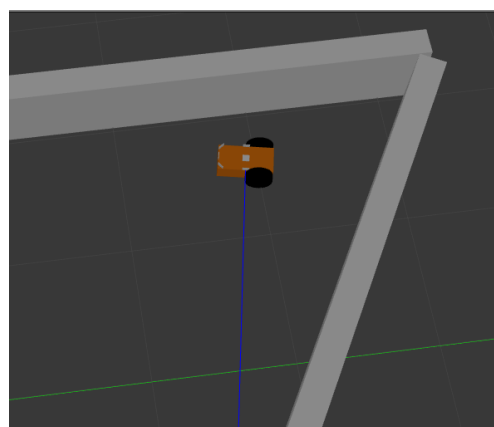
Μετά τη στροφή



Αρχική Θέση του ρομπότ



Ευθεία κίνηση παράλληλα σε τοίχο



Οξεία στροφή