

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

Πολυτεχνική Σχολή

Τμήμα Μηχανικών Η/Υ & Πληροφορικής

Διπλωματική Εργασία

Συνεργατική εύρεση βέλτιστης διαδρομής σε περιβάλλον κινούμενων εμποδίων

Ιωάννης Σταμούλης

Α.Μ. 1059675

Επιβλέπων

Δερματάς Ευάγγελος

Συνεπιβλέποντες

Μπερμπερίδης Κωνσταντίνος, Ψαράκης Εμμανουήλ

Πάτρα, Ιούνιος 2025

Πάτρα, Ιούνιος 2025

Ιωάννης Σταμούλης – Α.Μ. 1059675

Τίτλος: «Συνεργατική εύρεση βέλτιστης διαδρομής σε περιβάλλον κινούμενων εμποδίων»

Φοιτητής: Σταμούλης Ιωάννης **Επιβλέπων:** Δερματάς Ευάγγελος

Το θέμα της εργασίας είναι η διερεύνηση των αρχών λειτουργίας ενός μοντέλου SLAM για την πλοήγηση, χαρτογράφηση και αποφυγή εμποδίων σε έναν άγνωστο χώρο ο οποίος διερμηνεύεται από ένα δίτροχο ρομπότ εξοπλισμένο με υπέρηχους αισθητήρες. Η πρόκληση που αντιμετωπίζεται βρίσκεται στη φύση του θεωρητικού μοντέλου σε σχέση με τη συμπεριφορά του ρομπότ κατά τα πειράματα μιας και η λήψη μετρήσεων από τους αισθητήρες αποδεικνύεται θορυβώδης και χρίζει φιλτραρίσματος και διόρθωσης. Πηγές θορύβου αποτελούν οι ίδιοι οι αισθητήρες, η θέση τους, οι ατέλειες στην κίνηση των τροχών του ρομπότ, οι ανακλάσεις των υπερήχων από άλλα σημεία του περιβάλλοντος, τυχόν απότομες ακμές στο περίγραμμα των εμποδίων και φυσικά ο θόρυβος από τις παραδοχές του αλγορίθμου επεξεργασίας των δεδομένων. Στόχος της διπλωματικής είναι να υλοποιήσει έναν απλό αλγόριθμο συλλογής και επεξεργασίας ηχητικών σημάτων με σκοπό την αναπαραγωγή του περιβάλλοντος χώρου και των εμποδίων μέσα σε αυτόν και να αξιολογήσει την ικανότητα διάφορων πιθανοτικών τεχνικών που βοηθούν στη βελτιστοποίηση αποφυγής εμποδίων και χαρτογράφησης με μεγαλύτερη ακρίβεια. Τα πειράματα πραγματοποιούνται στην πλατφόρμα προσομοιώσεων Webots και γίνεται χρήση των γλωσσών C και Python για συγγραφή του αλγορίθμου και απεικόνιση των δεδομένων αντίστοιχα.

Πάτρα, Ιούνιος 2025

Ιωάννης Σταμούλης – Α.Μ. 1059675

ΠΕΡΙΛΗΨΗ

Στην παρούσα εργασία θα αξιολογηθεί ένας αλγόριθμος επεξεργασίας υπερηχητικών σημάτων που λαμβάνονται από αισθητήρες υπερήχων πάνω στο ρομπότ e-puck. Το πρώτο κεφάλαιο περιγράφει την υλοποίηση και αξιολόγηση τεχνικών SLAM και αποφυγής εμποδίων σε περιβάλλον προσομοίωσης Webots, με χρήση του ρομπότ e-puck και βασική αξιοποίηση δεδομένων από αισθητήρες, εστιάζοντας στην πρακτική εφαρμογή θεωρητικών εννοιών της ρομποτικής αντίληψης και συγχώνευσης αισθητήρων. Στο δεύτερο κεφάλαιο παρουσιάζω τη θεωρία και την εφαρμογή του Μπεϋζιανού φίλτρου στη ρομποτική, με έμφαση στη χαρτογράφηση κατειλημμένου πλέγματος, αξιοποιώντας λογαριθμικές πιθανότητες και αντίστροφα μοντέλα αισθητήρων για την εκτίμηση της κατάληψης του περιβάλλοντος από θορυβώδη δεδομένα αισθητήρων. Το Κεφάλαιο 3 παρουσιάζει τους βασικότερους αλγορίθμους και τεχνικές συλλογής δεδομένων για SLAM, αναλύοντας δημοφιλείς προσεγγίσεις όπως τα Kalman και Particle Filters, τη χρήση αισθητήρων για οδομετρία όπως λέιζερ και κάμερες, καθώς και τους τύπους αισθητήρων σε σύγχρονα ρομπότ που επιτρέπουν την αντίληψη, τον εντοπισμό και τη χαρτογράφηση του περιβάλλοντος με σκοπό την αυτόνομη πλοήγηση. Στο κεφάλαιο 4 παρουσιάζω το Webots, ένα περιβάλλον εξομοίωσης ρομπότ που επιτρέπει τη γρήγορη ανάπτυξη και προγραμματισμό τρισδιάστατων εικονικών ρομπότ με φυσικές ιδιότητες, αισθητήρες και μηχανισμούς. Το webots προσφέρει έτοιμα μοντέλα και διεπαφές για τη μεταφορά προγραμμάτων σε πραγματικά ρομπότ, όπως το e-puck. Στο επόμενο κεφάλαιο δίνεται η περιγραφή των χαρακτηριστικώ του ρομπότ που χρησιμοποίησα και παρουσιάζονται οι παράμετροι που ενεργοποιήθηκαν για το σκοπό των πειραμάτων μου. Τέλος στα επόμενα κεφάλαια 6 & 7 περιγράφεται αναλυτικά το FSM που χρησιμοποιήθηκε για την επίλυση του προβλήματος και δίνονται οι περιορισμοί, παραδοχές και τα αποτελέσματα του πειράματος. Το πείραμα έδειξε πως το σχεδιασμένο ρομπότ, προσομοιωμένο στο Webots, κατάφερε να κινηθεί αυτόνομα στον χώρο, να αποφύγει εμπόδια και να δημιουργήσει έναν λειτουργικό χάρτη του περιβάλλοντος.

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Ευάγγελο Δερματά για την προθυμία του και την ικανότητα να δίνει λύσεις σε όποια προβλήματα και δυσκολίες προέκυψαν κατά τη διάρκεια εκπόνησης της εργασίας μου καθώς και για τον ενθουσιασμό του για το αντικείμενο της Ρομποτικής, πράγμα το οποίο συντέλεσε σημαντικό παράγοντα για τη συνεργασίας μας.

Η διπλωματική εργασία αυτή και η συνολική πορεία μου στο τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής του Πανεπιστημίου Πατρών θα ήταν αδύνατη χωρίς τη διαρκή υποστήριξη της οικογένειάς μου και των φίλων μου και τους ευχαριστώ θερμά έναν έναν:

Τη μητέρα, τον πατέρα και τον αδερφό μου Χρήστο, τους Δημήτριο Γερογιάννη, Γκέκα Βασιλική, Ζαφειρέλη Αντώνιο, Ζύμνη Μάρκελλο, Κουϊμτζίδη Σάββα, Κύρο Στεργιο, Παπαζαχαρία Εμμανουέλα, Σκύρλα Αγάθη, Τοπτσή Μιχάλη, Χατζημιχάλη Γιάννη, τον Γιάννη Σαρταμπάκο και τον Απόστολο Κονταρίνη.

Περιεχόμενα

Κεφάλαιο 1: 11

Εισαγωγή 11

Κεφάλαιο 1.1 Αντικείμενο της εργασίας 11

Κεφάλαιο 2: 14

Μπεϋζιανό Φίλτρο και Χάρτης Δικτύου Πληρότητας 14

2.1 Εισαγωγή στο Μπεϋζιανό Φίλτρο 14

2.1 Δυαδικά Φίλτρα Bayes με Στατική Κατάσταση 14

2.2 Χαρτογράφηση Κατειλημμένου Πλέγματος 17

Κεφάλαιο 3: 22

Αλγόριθμοι και τεχνικές συλλογής δεδομένων για SLAM 22

Κεφάλαιο 3.1: Δημοφιλείς αλγόριθμοι στο παρασκήνιο επεξεργασίας τεχνικών SLAM 22

Κεφάλαιο 3.2: Τύποι αισθητήρων σε σύγχρονα ρομπότ 24

Κεφάλαιο 3.2.1: Επισκόπηση τεχνολογιών ανίχνευσης με έναν αισθητήρα 28

Κεφάλαιο 3.2.2: Επισκόπηση τεχνολογιών ανίχνευσης με πολλαπλούς αισθητήρες 38

Κεφάλαιο 3.3 Συζητήσεις και μελλοντικές τάσεις 38

Κεφάλαιο 4: 40

Περιβάλλον εξομοίωσης και παράμετροι 40

4.1 Τι είναι το Webots; 40

4.2 Τι μπορώ να κάνω με το webots; 41

4.3 Τι πρέπει να γνωρίζει κανείς για να κάνει χρήση του webots; 41

4.4 Η διεπαφή χρήστη 41

4.5 Προσθήκη αντικειμένου 42

4.6 Ταχύμετρο και Εικονικός Χρόνος 43

4.7 Το παράθυρο 3-D 43

4.8 Μενού επιλογών 44

4.9 Χειρισμός αξόνων αντικειμένου 44

4.10 Εφαρμογή δύναμης σε στερεό 45

4.11 Εφαρμογή ροπής σε στερεό 45
4.12 Το δέντρο αντικειμένων 46
4.13 Μενού εισαγωγής αντικειμένων 46
4.14 Η κονσόλα 46
4.15 Περιγραφή επιμέρους στοιχείων του περιβάλλοντος 47

Κεφάλαιο 5: 48
Εικονικό ρομπότ που χρησιμοποιήθηκε 48
5.1 Επισκόπηση των χαρακτηριστικών του e-puck 48
5.2 Δείγματα λειτουργίας 51
5.3 Παραμετροποίηση εικονικού ρομπότ 51

Κεφάλαιο 6: 53
Η προσέγγισή μου 53
Κεφάλαιο 6.1: Λειτουργία συστήματος 53
Κεφάλαιο 6.2: Χαρτογράφηση και βήμα επεξεργασίας 58
Κεφάλαιο 6.3: Επικύρωση σωστής λειτουργίας 59

Κεφάλαιο 7: 64
Συμπεράσματα 64
Κεφάλαιο 7.1: Συμπεράσματα 64
Κεφάλαιο 7.2: Διδάγματα 66

Βιβλιογραφία 67

Παραρτήματα 74

Παράρτημα Α: 74

Παράρτημα Β: 76

Πίνακες Εικόνων

Εικόνα 2 1 15
Εικόνα 2 2 15
Εικόνα 2 3 15
Εικόνα 2 4 15
Εικόνα 2 5 16
Εικόνα 2 6 16
Εικόνα 2 7 Εάν δεν υπάρχει προηγούμενη γνώση, ορίζουμε p(X) = 0.5 ώστε να εκφραστεί πλήρης άγνοια. 16
Εικόνα 2 8 Εκ των υστέρων πιθανότητα p(m | z_1:t, x_1:t) με γνωστές θέσεις ρομπότ και παρατηρήσεις. 17
Εικόνα 2 9 Κάθε κελί mi έχει πιθανότητα p(mi) – υπολογιστική πρόκληση λόγω του τεράστιου πλήθους πιθανών συνδυασμών (π.χ. 2^10.000 χάρτες). 17
Εικόνα 2 10 Μείωση υπολογιστικής πολυπλοκότητας, εις βάρος της δυνατότητας μοντελοποίησης εξαρτήσεων μεταξύ γειτονικών κελιών. 17
Εικόνα 2 11 Η συνολική εκ των υστέρων πιθανότητα προσεγγίζεται ως γινόμενο των πιθανοτήτων κάθε κελιού 18
Εικόνα 2 12 Αναπαράσταση log-odds για την εκτίμηση κατάληψης κελιών lt,i. 18
Εικόνα 2 13 Από τον log-odds υπολογισμό μπορούμε να ανακτήσουμε την κανονική πιθανότητα. 18
Εικόνα 2 14 Εικόνα 2 15 20
Εικόνα 2 16 21

Εικόνα 3 1 22

Εικόνα 4 1 Περιβάλλον webots - κονσόλα 42
Εικόνα 4 2 Περιβάλλον webots - Προσθήκη αντικειμένου 42
Εικόνα 4 3 Περιβάλλον webots - Ταχύμετρο και εικονικός χρόνος 43
Εικόνα 4 4 Περιβάλλον webots - Παράθυρο 3D 43
Εικόνα 4 5 Περιβάλλον webots - Μενού επιλογών 44
Εικόνα 4 6 Περιβάλλον webots - Χειρισμός αξόνων αντικειμένου 44
Εικόνα 4 7 Περιβάλλον webots - Εφαρμογή ροπής σε στερεό 45
Εικόνα 4 8 Περιβάλλον webots – Μενού εισαγωγής αντικειμένων 46
Εικόνα 4 9 Περιβάλλον webots – Κονσόλα 47
Εικόνα 4 10 Εικόνα 4 11 Εικόνα 4 12 47

Εικόνα 5 1 Επισκόπηση e-puck 48
Εικόνα 5 2 Ανάκλαση υπερήχων σε εμπόδιο 49
Εικόνα 5 3 Κατεύθυνση αισθητήρων e-puck 50
Εικόνα 5 4 Αισθητήρες e-puck με αναγνωριστικά 50
Εικόνα 5 5 Σενάριο Κίνησης 51
Εικόνα 5 6 Σενάριο Κίνησης 2 51
Εικόνα 5 7 Σενάριο Κίνησης 3 51

Εικόνα 5 8 Ανάκλαση υπερηχητικού σήματος sonar αισθητήρων Εικόνα 5 9 Περιβάλλον webots - Τύποι 52
Εικόνα 5 10 Περιβάλλον webots - Στοιχεία αισθητήρα sonar 52

Εικόνα 6 1 Finite State Machine 54
Εικόνα 6 2 Σενάριο Κίνησης Πειράματος 55
Εικόνα 6 3 Σενάριο Κίνησης Πειράματος (2) 56
Εικόνα 6 4 Flow Chart Λογικής Κίνησης 57
Εικόνα 6 5 Σύστημα συντεταγμένων Ρομπότ 59
Εικόνα 6 6 Θεωρητικό περιβάλλον εξερεύνησης 61
Εικόνα 6 7 Ιδανική ανακατασκευή περιβάλλοντος 61
Εικόνα 6 8 Πειραματικό περιβάλλον εξερεύνησης 62
Εικόνα 6 9 Πειραματική Ανακατασκευή Περιβάλλοντος 62

Κεφάλαιο 1:

Εισαγωγή

Κεφάλαιο 1.1 Αντικείμενο της εργασίας

Η παρούσα διπλωματική εργασία έχει ως αντικείμενο τη μελέτη και υλοποίηση τεχνικών αποφυγής εμποδίων στο περιβάλλον εξομοίωσης Webots. Οι τεχνικές αυτές εφαρμόζονται μέσω του ρομπότ τύπου e-puck, το οποίο είναι εξοπλισμένο με αισθητήρες υπερήχων. Η έμφαση δίνεται στην αξιολόγηση της συμπεριφοράς του ρομπότ κατά την πλοήγηση σε έναν άγνωστο χώρο, σταδιακά εισάγοντας μηχανισμούς αποφυγής εμποδίων, απομνημόνευσης του περιβάλλοντος και δημιουργίας χάρτη. Καθ' όλη τη διάρκεια της διαδικασίας, η στάση (θέση και προσανατολισμός) του ρομπότ θεωρείται ως δεδομένη.

Η εργασία στοχεύει στη μελέτη και εφαρμογή μεθόδων που βασίζονται στην αρχή της ταυτόχρονης χαρτογράφησης και εντοπισμού, γνωστή στη βιβλιογραφία ως SLAM (Simultaneous Localization and Mapping). Η βασική ιδέα του SLAM είναι ότι το ρομπότ κινείται αυτόνομα σε έναν άγνωστο, κλειστό χώρο, αναγνωρίζοντας και αποφεύγοντας εμπόδια ενώ παράλληλα κατασκευάζει έναν χάρτη του περιβάλλοντος που εξερευνά. Στη συγκεκριμένη εφαρμογή, το ρομπότ αξιοποιεί δύο αισθητήρες υπερήχων και μία μονάδα αδρανειακής πλοήγησης (MPU) που ανιχνεύει αλλαγές στη θέση και τον προσανατολισμό. Αν και οι πληροφορίες μπορούν να επεξεργαστούν εντός του ρομπότ, εδώ εφαρμόζεται μοντέλο τηλεμετρίας: τα δεδομένα μεταφέρονται σε εξωτερικό υπολογιστή, όπου γίνεται η επεξεργασία, η παραγωγή του πιθανοτικού χάρτη και η απεικόνιση της τροχιάς του ρομπότ. Έτσι, μειώνονται οι υπολογιστικές απαιτήσεις του ρομπότ, χωρίς όμως η εργασία να επικεντρώνεται σε αυτό το σημείο.

Το πρόβλημα SLAM πρωτοεμφανίστηκε στις αρχές της δεκαετίας του 1980 από τους ερευνητές R.C. Smith και Cheesman. Έκτοτε, έχει προσελκύσει το έντονο ενδιαφέρον της επιστημονικής και ερευνητικής κοινότητας, με αποτέλεσμα την πρόταση και ανάπτυξη πληθώρας αλγορίθμων και μεθόδων για την επίλυσή του. Στην ουσία, το SLAM είναι ένα αλληλεξαρτώμενο πρόβλημα: για να επιτευχθεί ακριβής εντοπισμός απαιτείται ένας αξιόπιστος χάρτης, και ταυτόχρονα, για να κατασκευαστεί ένας αξιόπιστος χάρτης, χρειάζεται ακριβής γνώση της θέσης του ρομπότ. Ο ρομποτικός πράκτορας βασίζεται αποκλειστικά στις μετρήσεις των αισθητήρων του για οδομετρία, σημεία αναφοράς ή αδρανειακή πλοήγηση—, οι οποίες όμως, λόγω της φύσης τους, ενδέχεται να είναι θορυβώδεις και να παρουσιάζουν αυξανόμενα σφάλματα με την πάροδο του χρόνου.

Η παρούσα εργασία επιδιώκει τον σχεδιασμό ενός τέτοιου ολοκληρωμένου συστήματος, ενώ παράλληλα αποσκοπεί στη δημιουργία ενός συνοπτικού και πρακτικού οδηγού για όσους επιθυμούν να ασχοληθούν με αντίστοιχες υλοποιήσεις SLAM. Το ρομπότ e-puck αξιολογήθηκε υπό διαφορετικά σενάρια προσομοίωσης, με ενδεικτικό παράδειγμα την πλοήγηση και χαρτογράφηση σε κλειστό δισδιάστατο χώρο. Το σύστημα αναδεικνύει την ικανότητα επίτευξης αποτελεσμάτων ακόμα και σε συνθήκες περιορισμένων πόρων. Αν και δεν παρουσιάζεται κάποια ριζικά καινοτόμος προσέγγιση, η εργασία προσφέρει έναν σταθερό κορμό γνώσης που μπορεί να αποτελέσει εφαλτήριο για πιο σύνθετες ή πραγματικού χρόνου εφαρμογές.

Η ανάπτυξη αυτόνομων κινητών ρομπότ για χρήση σε εσωτερικούς χώρους αποτελεί ενεργό πεδίο ερευνητικής και τεχνολογικής δραστηριότητας. Για την αποτελεσματική τους λειτουργία, απαιτείται η ικανότητα συλλογής και επεξεργασίας πληροφοριών για το περιβάλλον και την ίδια τους την κατάσταση. Αυτό επιτυγχάνεται μέσω αισθητήρων διαφόρων τύπων, οι οποίοι επιτρέπουν εντοπισμό θέσης, χαρτογράφηση του χώρου, αναγνώριση εμποδίων και αυτόνομη πλοήγηση. Στο πλαίσιο αυτό, γίνεται εκτενής αναφορά στις διαθέσιμες τεχνολογίες αισθητήρων για χρήση σε ρομπότ εσωτερικού χώρου, στα πλεονεκτήματα και μειονεκτήματα της μονοαισθητήριας αντίληψης, καθώς και στους βασικούς αλγορίθμους επεξεργασίας και συγχώνευσης μετρήσεων από πολλαπλούς αισθητήρες. Επιπλέον, εξετάζονται οι τάσεις για μελλοντικές εξελίξεις στον τομέα αυτό.

Κεντρική θεωρητική βάση για τα παραπάνω αποτελεί η θεωρία Bayes, η οποία προσφέρει στατιστική μεθοδολογία ενσωμάτωσης πρότερης γνώσης σε νέα δεδομένα παρατήρησης. Στο πλαίσιο της ρομποτικής, χρησιμοποιείται η Bayesιανή διήθηση (Bayesian Filtering) για την εκτίμηση της κατάστασης ενός δυναμικού συστήματος με βάση τις εισερχόμενες μετρήσεις και τις εντολές ελέγχου. Ειδικότερα, το φίλτρο Bayes χρησιμοποιείται για την εκτίμηση της θέσης και του προσανατολισμού του ρομπότ μέσω δύο βασικών σταδίων: (α) της πρόβλεψης (prediction), με βάση τις κινήσεις που έχει εκτελέσει το ρομπότ, και (β) της ενημέρωσης (correction), με βάση τις νέες μετρήσεις από τους αισθητήρες.

Μια ειδική μορφή αυτής της μεθόδου είναι το δυαδικό Bayes φίλτρο (binary Bayes filter), το οποίο εφαρμόζεται στην τεχνική χαρτογράφησης πλεγμάτων (Occupancy Grid Mapping), όπου κάθε κελί του πλέγματος του χάρτη αντιστοιχεί σε πιθανότητα να είναι κατειλημμένο ή ελεύθερο. Οι αισθητήρες του ρομπότ συλλέγουν δεδομένα από το περιβάλλον και αυτά μετατρέπονται σε πιθανότητες κατάληψης. Για λόγους ακρίβειας και ευκολίας υπολογισμών, εφαρμόζεται η αναπαράσταση των πιθανοτήτων με λογαριθμικούς λόγους (log-odds). Η χρήση του αντιστρόφου μοντέλου αισθητήρα επιτρέπει την επικαιροποίηση των πιθανοτήτων κατοχής κάθε κελιού με βάση τις εκάστοτε μετρήσεις. Η διαδικασία αυτή επαναλαμβάνεται δυναμικά όσο το ρομπότ κινείται.

Συνοψίζοντας, η θεωρία του Bayes και τα παράγωγά της αποτελούν βασικό εργαλείο για την πλοήγηση, τον εντοπισμό και τη λήψη αποφάσεων σε δυναμικά και άγνωστα περιβάλλοντα. Το δυαδικό φίλτρο Bayes επιτρέπει αποδοτική διαχείριση αβεβαιότητας και ακριβή αναπαράσταση της κατάστασης του χώρου μέσω πιθανοτικών χαρτών.

Οι αλγόριθμοι SLAM μπορούν να ταξινομηθούν με βάση τους αισθητήρες που χρησιμοποιούν (όπως λέιζερ, υπερήχους, ή κάμερες), τις υπολογιστικές τεχνικές που υιοθετούν (όπως Kalman Filter, Particle Filter κ.ά.), και τη δομή των δεδομένων που χρησιμοποιούν. Οι δύο κύριες κατηγορίες είναι το Online-SLAM, όπου αποθηκεύονται μόνο τα απολύτως απαραίτητα δεδομένα για την τρέχουσα εκτίμηση, και το Full-SLAM, όπου διατηρείται πλήρης ιστορία μετρήσεων και καταστάσεων. Η επιλογή της μεθόδου εξαρτάται από τους διαθέσιμους πόρους και τις απαιτήσεις της εφαρμογής. Για παράδειγμα, η χρήση αισθητήρων LIDAR προσφέρει υψηλή ακρίβεια αλλά έχει αυξημένο κόστος, ενώ οι αισθητήρες υπερήχων είναι πιο οικονομικοί αλλά παρουσιάζουν περιορισμούς σε απόσταση και ευκρίνεια μετρήσεων.

Στην παρούσα εργασία, υλοποιήθηκε ένα σύστημα χαμηλού κόστους μέσω της πλατφόρμας προσομοίωσης Webots. Το ρομπότ e-puck, προγραμματισμένο με συγκεκριμένη ακολουθία εντολών, κινείται αυτόνομα σε προκαθορισμένο χώρο, καταγράφοντας δεδομένα τα οποία αποστέλλονται στον υπολογιστή για επεξεργασία. Η λειτουργία του ρομπότ βασίζεται σε Finite State Machine (FSM), που

αποτελείται από έξι βασικές καταστάσεις: εκκίνηση, διαμόρφωση, στοίχιση, επαναφορά, περιστροφή και μετάβαση προς τα εμπρός. Οι εντολές δίνονται σε κινητήρες τύπου stepper, επιτρέποντας ακριβή έλεγχο θέσης και ταχύτητας.

Η διαδικασία χαρτογράφησης βασίζεται στη μέθοδο Occupancy Grid Map. Οι μετρήσεις των αισθητήρων μετατρέπονται σε συντεταγμένες, και με τη βοήθεια γλώσσας προγραμματισμού Python και των βιβλιοθηκών serial και matplotlib, δημιουργείται δυναμικά ένας πιθανοτικός χάρτης. Ο υπολογισμός των συντεταγμένων των εμποδίων προσαρμόζεται με βάση την τρέχουσα κατεύθυνση του ρομπότ.

Η ορθή λειτουργία του συστήματος επαληθεύτηκε μέσω σειράς δοκιμών. Έγινε έλεγχος της ακρίβειας των αισθητήρων υπερήχων σε γνωστές αποστάσεις, συγχρονισμός των μοτέρ και έλεγχος της IMU για πιστότητα μέτρησης προσανατολισμού. Οι μετρήσεις συγκρίθηκαν με αναμενόμενα δεδομένα για την επιβεβαίωση της αξιοπιστίας του συστήματος.

Συνοψίζοντας, η παρούσα εργασία παρουσιάζει την ανάπτυξη ενός αυτόνομου ρομπότ χαμηλού κόστους, ικανού να χαρτογραφεί και να αποφεύγει εμπόδια σε ελεγχόμενο περιβάλλον. Παρουσιάζονται τα στάδια σχεδίασης, αξιολόγησης και πιθανοί τρόποι βελτίωσης του συστήματος. Στις μελλοντικές προτάσεις περιλαμβάνονται η χρήση 360° LIDAR, υπέρυθρων αισθητήρων με κατάλληλα φίλτρα, καθώς και πιο αποδοτικοί αλγόριθμοι πλοήγησης και SLAM, που μπορούν να βελτιώσουν περαιτέρω την ακρίβεια και την αποδοτικότητα του ρομπότ.

Κεφάλαιο 2:

Μπεϋζιανό Φίλτρο και Χάρτης Δικτύου Πληρότητας

Η θεωρία Bayes αποτελεί έναν θεμελιώδη κλάδο της μαθηματικής θεωρίας πιθανοτήτων και της στατιστικής, ο οποίος επιτρέπει την περιγραφή της αβεβαιότητας ενός γεγονότος με βάση την προηγούμενη γνώση σχετικά με τις συνθήκες που το επηρεάζουν, καθώς και τα διαθέσιμα παρατηρησιακά δεδομένα. Μια βασική εφαρμογή της θεωρίας αυτής είναι η Bayesιανή συμπερασματολογία — μια πρακτική μέθοδος για τη στατιστική ανάλυση.

Στη Bayesιανή προσέγγιση, όλες οι αβεβαιότητες (είτε αφορούν παραμέτρους είτε καταστάσεις) μοντελοποιούνται ως τυχαίες μεταβλητές. Ο στόχος είναι η εξαγωγή της κατανομής πιθανοτήτων για τις υπό εξέταση καταστάσεις, αξιοποιώντας προηγούμενη γνώση και σύνολα παρατηρήσεων. Η Bayesιανή διήθηση, ειδικότερα, εφαρμόζει το θεωρήμα του Bayes για την επίλυση προβλημάτων πιθανολογικής συμπερασματολογίας. Σε αυτό το πλαίσιο, χρησιμοποιούμε τέτοιες τεχνικές για να εκτιμήσουμε την κατάσταση ενός κινητού ρομπότ.

2.1 Εισαγωγή στο Μπεϋζιανό Φίλτρο

Στη ρομποτική, το φίλτρο Bayes χρησιμοποιείται ευρέως για την εκτίμηση πιθανοτήτων σχετικών με την πεποίθηση ενός ρομπότ για το πού βρίσκεται ή πώς είναι προσανατολισμένο. Ουσιαστικά, ο αλγόριθμος βασίζεται σε μια διαδικασία αναδρομής, στην οποία η πεποίθηση του ρομπότ για την κατάσταση του περιβάλλοντος τη χρονική στιγμή t εξαρτάται από την πεποίθησή του στη χρονική στιγμή t-1.

Ο αλγόριθμος αποτελείται από δύο βασικά στάδια:

Πρόβλεψη (**Prediction**) – Ενημέρωση βάσει των δεδομένων ελέγχου u.

Καινοτομία ή Ενημέρωση (**Correction**) – Συνδυασμός της πρόβλεψης με τη μέτρηση z μέσω του Bayes.

Έτσι, το ρομπότ μπορεί συνεχώς να επικαιροποιεί τις πεποιθήσεις του καθώς λαμβάνει νέα δεδομένα από το περιβάλλον.

2.1 Δυαδικά Φίλτρα Bayes με Στατική Κατάσταση

Το Δυαδικό Φίλτρο Bayes [1], [2] είναι μια εξειδικευμένη περίπτωση του βέλτιστου Bayesιανού φίλτρου, στην οποία η κατάσταση X είναι στατική και ο χώρος καταστάσεων διακριτός και δυαδικός (δηλ. κάθε κατάσταση είναι είτε 0 είτε 1). Αυτή η μορφή είναι κατάλληλη για αλγορίθμους χαρτογράφησης κατειλημμένου πλέγματος, όπου κάθε κελί μπορεί να είναι είτε κατειλημμένο είτε ελεύθερο.

$$bel_t(X) = p(X|z_{1:t}, u_{1:t}) = p(X|z_{1:t})$$

Εικόνα 2 1

Υπό την υπόθεση του Markov και εφαρμόζοντας το θεώρημα του Bayes δύο φορές στην εξίσωση 2.1, προκύπτει:

$$p(X|z_{1:t-1}) = \frac{p(z_t|X, z_{1:t-1})\, p(X|z_{1:t-1})}{p(z_t|z_{1:t-1})} = \frac{p(z_t|X)\, p(X|z_{1:t-1})}{p(z_t|z_{1:t-1})} = \\ = \frac{p(X|z_t)\, p(z_t)\, p(X|z_{1:t-1})}{p(X)\, p(z_t|z_{1:t-1})}$$

Εικόνα 2 2

Ομοίως, για το αντίθετο γεγονός, προκύπτει:

$$bel_t(\neg X) = p(\neg X|z_{1:t-1}) = \frac{p(\neg X|z_t)\, p(z_t)\, p(\neg X|z_{1:t-1})}{p(\neg X)\, p(z_t|z_{1:t-1})}$$

Εικόνα 2 3

Γενικά, το πρόβλημα της εκτίμησης της πεποίθησης αναπαρίσταται ως λόγος λογαριθμικών πιθανοτήτων (log-odds). Αυτό γίνεται για την αποφυγή προβλημάτων αποκοπής που προκύπτουν από πιθανότητες κοντά στο 0 ή το 1. Ο λόγος λογαριθμικών πιθανοτήτων της κατάστασης X ορίζεται ως ο λογάριθμος της πιθανότητας του γεγονότος διαιρεμένος με την πιθανότητα του αντίθετου γεγονότος.

$$l_t(X) = \log \frac{p(X|z_1{:}t)}{p(\neg X|z_{1:t})} = \log \frac{p(X|z_1{:}t)}{1 - p(X|z_1{:}t)}$$

Εικόνα 2 4

Χρησιμοποιώντας τις εξισώσεις 2.2 και 2.3 στην 2.4, προκύπτει:

$$
\begin{aligned}
l_t(X) &= \log\left(\frac{p(X|z_t)}{p(\neg X|z_t)}\frac{p(X|z_{1:t-1})}{p(\neg X|z_{1:t-1})}\frac{p(\neg X)}{p(X)}\right) \\
&= \log\left(\frac{p(X|z_t)}{1-p(X|z_t)}\frac{p(X|z_{1:t-1})}{1-p(X|z_{1:t-1})}\frac{1-p(X)}{p(X)}\right) \\
&= \log\frac{p(X|z_t)}{1-p(X|z_t)}+\log\frac{p(X|z_{1:t-1})}{1-p(X|z_{1:t-1})}+\log\frac{1-p(X)}{p(X)} \\
&= \overbrace{\log\frac{p(X|z_t)}{1-p(X|z_t)}}^{Inverse\ Sensor\ Model}+l_{t-1}(X)-\overbrace{\log\frac{p(X)}{1-p(X)}}^{Prior}
\end{aligned}
$$

Εικόνα 2 5

(3)Ο χώρος καταστάσεων ενός δυναμικού συστήματος είναι το σύνολο όλων των πιθανών καταστάσεων του συστήματος. Κάθε συντεταγμένη είναι μια μεταβλητή κατάστασης, και οι τιμές όλων των μεταβλητών κατάστασης συνολικά περιγράφουν την κατάσταση του συστήματος.

1: **Algorithm binary_Bayes_filter(l_{t-1} , z_t)**:

2: $l_t = l_{t-1} + \log\frac{p(X|z_t)}{1-p(X|z_t)} - \log\frac{p(x)}{1-p(x)}$

3: return l_t

Εικόνα 2 6

Το δυαδικό φίλτρο Bayes σε μορφή λογαριθμικών πιθανοτήτων (log-odds) με αντίστροφο μοντέλο μετρήσεων. Εδώ, το lt είναι οι λογαριθμικές πιθανότητες της εκ των υστέρων πεποίθησης για μια δυαδική μεταβλητή κατάστασης που δεν αλλάζει με την πάροδο του χρόνου (πηγή: Probabilistic Robotics) [2].

Αυτός ο τύπος φίλτρου αξιοποιεί το **αντίστροφο μοντέλο αισθητήρα** *p(X | zt)* αντί του εμπρόσθιου *p(zt | X)*, που είναι η πιο συνηθισμένη μορφή. Για την έναρξη της διαδικασίας απαιτείται ο καθορισμός της σταθεράς l_0, βάσει της αρχικής (προγενέστερης) πιθανότητας:

$$l_0(X) = \log\frac{p(X)}{1-p(X)}$$

Εικόνα 2 7 Εάν δεν υπάρχει προηγούμενη γνώση, ορίζουμε p(X) = 0.5 ώστε να εκφραστεί πλήρης άγνοια.

2.2 Χαρτογράφηση Κατειλημμένου Πλέγματος

Η χαρτογράφηση είναι η διαδικασία δημιουργίας αναπαραστάσεων του περιβάλλοντος μέσω αισθητήρων. Στο πλαίσιο αυτής της διατριβής, χρησιμοποιούνται **δισδιάστατοι χάρτες κατειλημμένου πλέγματος**, όπου το περιβάλλον μοντελοποιείται ως πίνακας κελιών.

Σε αυτούς τους χάρτες:

- Κάθε κελί m_i μοντελοποιείται ως μια **δυαδική τυχαία μεταβλητή**.
- Η πιθανότητα κατάληψης κυμαίνεται μεταξύ 0 (ελεύθερο) και 1 (κατειλημμένο).

$$p(m \mid z_{1:t}, x_{1:t})$$

Εικόνα 2 8 Εκ των υστέρων πιθανότητα p(m | z_1:t, x_1:t) με γνωστές θέσεις ρομπότ και παρατηρήσεις.

Σύμφωνα με τη σύμβαση:

- m: ολόκληρος ο χάρτης
- z_1:t: παρατηρήσεις μέχρι τη στιγμή t
- x_1:t: γνωστές θέσεις ρομπότ

$$m = \{m_i\} \quad \text{or} \quad m = \sum_i m_i$$

Εικόνα 2 9 Κάθε κελί mi έχει πιθανότητα p(mi) – υπολογιστική πρόκληση λόγω του τεράστιου πλήθους πιθανών συνδυασμών (π.χ. 2^10.000 χάρτες).

Η λύση είναι να σπάσουμε το πρόβλημα σε υποπροβλήματα εκτίμησης για κάθε κελί μεμονωμένα:

$$p(m_i \mid z_{1:t}, x_{1:t})$$

Εικόνα 2 10 Μείωση υπολογιστικής πολυπλοκότητας, εις βάρος της δυνατότητας μοντελοποίησης εξαρτήσεων μεταξύ γειτονικών κελιών.

Τελικά, κάθε ένα από αυτά τα προβλήματα εκτίμησης είναι ένα δυαδικό πρόβλημα. Αυτή η διάσπαση είναι βολική καθώς απαλλάσσει το πρόβλημα από την εκ των υστέρων πιθανότητα υψηλής διάστασης. Ωστόσο, εισάγει ένα νέο πρόβλημα. Συγκεκριμένα, δεν μας επιτρέπει να μοντελοποιήσουμε πιθανές

εξαρτήσεις μεταξύ γειτονικών κελιών. Έτσι, η εκ των υστέρων πιθανότητα για έναν χάρτη προσεγγίζεται ως το γινόμενο των πιθανοτήτων όλων των κελιών του χάρτη.

$$p(m|z_{1:t},x_{1:t})=\prod_i p(m_i|z_{1:t},x_{1:t})$$

Εικόνα 2 11 Η συνολική εκ των υστέρων πιθανότητα προσεγγίζεται ως γινόμενο των πιθανοτήτων κάθε κελιού.

Για κάθε τέτοιο δυαδικό πρόβλημα, εφαρμόζεται το δυαδικό φίλτρο Bayes με στατική κατάσταση, όπως στον παρακάτω πίνακα:

$$l_{t,i} = \log\frac{p(m_i|z_{1:t},x_{1:t})}{1-p(m_i|z_{1:t},x_{1:t})}$$

Εικόνα 2 12 Αναπαράσταση log-odds για την εκτίμηση κατάληψης κελιών lt,i.

Η χρήση της αναπαράστασης λογαριθμικών πιθανοτήτων προσφέρει σημαντικά οφέλη στον αλγόριθμο λόγω των αριθμητικών πλεονεκτημάτων της σε περιπτώσεις μικρών πιθανοτήτων. Σημειώστε ότι οι πιθανότητες μπορούν εύκολα να ανακτηθούν από τον λόγο λογαριθμικών πιθανοτήτων.

$$p(m_i|z_{1:t},x_{1:t}) = 1-\frac{1}{1+e^{l_{t,i}}}$$

Εικόνα 2 13 Από τον log-odds υπολογισμό μπορούμε να ανακτήσουμε την κανονική πιθανότητα.

Η βασική λειτουργία του αλγορίθμου χαρτογράφησης (occupancy_grid_mapping):

- Διατρέχει όλα τα κελιά.
- Ενημερώνει την πιθανότητα για όσα ανήκουν στο αντιληπτικό πεδίο της μέτρησης *zt*.
- Χρησιμοποιεί τη συνάρτηση *inverse_sensor_model*.

Ο αλγόριθμος βασίζεται στο φίλτρο Bayes από τον Πίνακα 2.1 ([2] Probabilistic Robotics).

Η *inverse_sensor_model* υλοποιεί το αντίστροφο μοντέλο μέτρησης *p(mi | zt, xt)*. Είναι ιδιαίτερα χρήσιμη όταν ο χώρος των μετρήσεων είναι πολύπλοκος.

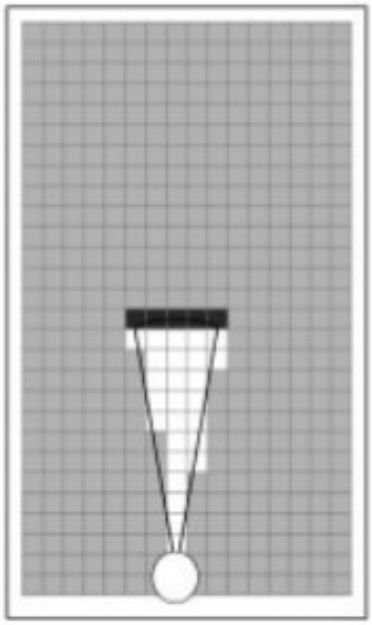
```
1.  Algorithm occupancy_grid_mapping({l_{t-1,i}}, χ_t, z_t):
2.      for all cells m_i do
3.          if m_i in perceptual field of z_t then
4.              l_{t,i} = l_{t-1,i} + inverse_sensor_model(m_i, χ_t, z_t) - l_0
5.          else
6.              l_{t,i} = l_{t-1,i}
7.          endif
8.      endfor
9.      return {l_{t,i}}
```

Ο αλγόριθμος κατειλημμένου πλέγματος, μια εκδοχή του δυαδικού φίλτρου Bayes από τον Πίνακα 2.1. (πηγή: Probabilistic Robotics) [2].

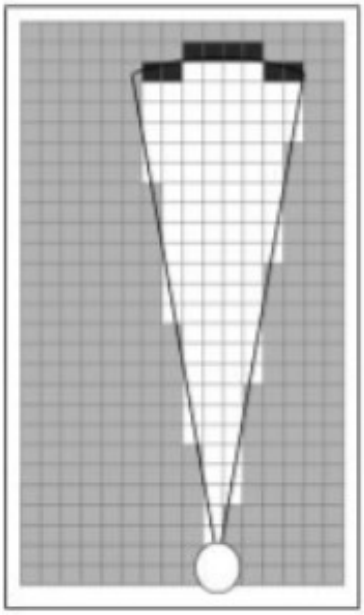
Η συνάρτηση **inverse_sensor_model** υλοποιεί το αντίστροφο μοντέλο μέτρησης της μορφής p(mi|zt, xt). Καθορίζει μια κατανομή πάνω στη δυαδική μεταβλητή κατάστασης mi, η οποία συνδέει ένα συγκεκριμένο κελί ως συνάρτηση της μέτρησης zt και της θέσης xt. Αυτό είναι χρήσιμο σε καταστάσεις όπου ο χώρος μετρήσεων είναι πολύ πιο σύνθετος από τον χώρο καταστάσεων. Μια βασική συνάρτηση για έναν ανιχνευτή απόστασης δίνεται στον κάτω Πίνακα και απεικονίζεται στις Εικόνα 2 14 και Εικόνα 2 15. [1]

```
1:  Algorithm inverse_range_sensor_model(i, x_t, z_t):
2:      Let x_i, y_i be the center-of-mass of m_i
3:      r = sqrt((x_i − x)^2 + (y_i − y)^2)
4:      φ = atan2(y_i − y, x_i − x) − θ
5:      k = argmin_j |φ − θ_{j,sens}|
6:      if r > min(z_max, z_t^k + α/2) or |φ − θ_{k,sens}| > β/2 then
7:          return l_0
8:      if z_t^k < z_max and |r − z_max| < α/2
9:          return l_occ
10:     if r ≤ z_t^k
11:         return l_free
12:     endif
```

Ένα απλό αντίστροφο μοντέλο μέτρησης για ρομπότ εξοπλισμένα με ανιχνευτές απόστασης. Εδώ, το α είναι το πάχος των εμποδίων και το β το πλάτος της δέσμης του αισθητήρα. Οι τιμές locc και lfree δηλώνουν την ποσότητα της απόδειξης που παρέχει μια ανάγνωση για τις δύο διαφορετικές περιπτώσεις (πηγή: Probabilistic Robotics) [2].



Εικόνα 2 14



Εικόνα 2 15

Γραφική απεικόνιση απλού αντίστροφου μοντέλου για αισθητήρες απόστασης. Οι παράμετροι α (πάχος εμποδίου) και β (εύρος δέσμης) ορίζουν την ακρίβεια, ενώ οι τιμές locc και lfree προσδιορίζουν την αξιοπιστία για κατάληψη ή ελευθερία.

Η **Εικόνα 2 16** δείχνει ένα παράδειγμα χάρτη κατειλημμένου πλέγματος δίπλα στο αρχιτεκτονικό σχέδιο του περιβάλλοντος χώρου. Ο χάρτης κατασκευάστηκε χρησιμοποιώντας μετρήσεις ενός ρομπότ που αποκτήθηκαν ενώ το ρομπότ εκτελούσε SLAM. Η κλίμακα του γκρι υποδεικνύει την εκ των υστέρων πιθανότητα: Το μαύρο αντιστοιχεί σε υψηλή πιθανότητα ότι ένα κελί είναι κατειλημμένο, ενώ το λευκό αντιστοιχεί σε υψηλή πιθανότητα ότι ένα κελί είναι ελεύθερο. Το γκρι φόντο αντιπροσωπεύει την προγενέστερη πιθανότητα.

Ο χάρτης δείχνει όλα τα εμπόδια που εντοπίζονται από τον αισθητήρα, αποδεικνύοντας την καταλληλότητά του για χρήση σε SLAM.



Εικόνα 2 16

Παράδειγμα χάρτη κατειλημμένου πλέγματος συγκριτικά με το αρχιτεκτονικό σχέδιο του χώρου. Μαύρα κελιά δείχνουν υψηλή πιθανότητα κατάληψης, λευκά δείχνουν ελευθερία, και γκρι αναπαριστά την αρχική πιθανότητα.

Κεφάλαιο 3:

Αλγόριθμοι και τεχνικές συλλογής δεδομένων για SLAM

Κεφάλαιο 3.1: Δημοφιλείς αλγόριθμοι στο παρασκήνιο επεξεργασίας τεχνικών SLAM

Από τη στιγμή που οι Peter Cheeseman, Jim Crowley και Hugh-Durrant Whyte έθεσαν τις πρώτες βάσεις για το SLAM εφαρμόζοντας θεωρία πιθανοτήτων σε αυτό, έχει μετατραπεί σε ένα πολύ ενεργό πεδίο στην έρευνα ρομποτικής. Η δημιουργία του SLAM έχει συντελέσει σε ένα μεγάλο αριθμό έργων αφοσιωμένα στο να δημιουργήσουν τις πιο κατάλληλες τεχνικές ικανές να αντιμετωπίσουν την αποκωδικοποίηση σε μετρήσιμα στοιχεία του περιβάλλοντος ενός ρομπότ ακόμη και στα πιο δυσπρόσιτα προς εξερεύνηση σημεία. Η έρευνα δεν περιορίζεται μόνο σε εσωτερικούς χώρους και οι τεχνικές μπορούν να κατηγοριοποιηθούν σύμφωνα με την αναπαράσταση σε χάρτη(χαρτογράφηση) και τις διάφορες τεχνικές εκτίμησης.

Υπάρχουν 2 κύριες αναπαραστάσεις χάρτη. Η πιο δημοφιλής είναι η αναπαράσταση χάρτη σε πλέγμα, η οποία χρησιμοποιείται και στη δικιά μας προσέγγιση. Η συγκεκριμένη προσέγγιση φέρει μεγάλο υπολογιστικό κόστος και απαιτεί πολλή μνήμη. Η δεύτερη πιο δημοφιλής είναι η αναπαράσταση με σημεία αναφοράς(Feature based) στην οποία το μοντέλο του χάρτη εκφράζεται σε μορφή σημείων αναφοράς στο χάρτη τα οποία βρίσκονται εντός του περιβάλλοντος που εξερευνιέται. Η μέθοδος αυτή έχει λάβει δημοφιλία κυρίως λόγω της κομψότητας σε χρήση μνήμης και υπολογιστικής ισχύος. Ωστόσο, βασίζεται βαριά σε γνώση για το περιβάλλον που έχει αποκτηθεί με άλλα μέσα και έχει ενσωματωθεί ήδη στο σύστημα. Αυτό προφανώς μειώνει το βαθμό ελευθερίας του συστήματος να κινεί το ρομπότ.

Μιας και η ρομποτική χαρτογράφηση καθιστά ένα συνεχώς αναπτυσσόμενο πεδίο, ο αριθμός των μεθόδων εκτίμησης έχει γίνει αρκετά μεγάλος και συνεχώς αυξάνεται. Οι αλγόριθμοι αυτοί που χρησιμοποιούνται ουσιαστικά διαφέρουν στους αισθητήρες με τους οποίους επιτυγχάνουν να συλλέξουν δεδομένα για το περιβάλλον τους και πως χρησιμοποιούν αυτά τα δεδομένα. Μπορεί επίσης να διαφέρουν στα φίλτρα στα οποία βασίζονται(π.χ. Kalman Filter, Particle Filter). Μεταξύ πολλών προσεγγίσεων επιλέξαμε να παρουσιάσουμε μερικές από τις πιο διαδεδομένες και δημοφιλείς λύσεις διαθέσιμες για ένα πρόβλημα εκτίμησης SLAM.

Τα Γκαουσιανα Φίλτρα αποτελούν την πιο παλιά καταγεγραμμένη υλοποίηση του φίλτρου Bayes για συνεχείς χώρους. Παρόλο τους περιορισμούς που έχουν, τα Γκαουσιανά Φίλτρα είναι μια δημοφιλής οικογένεια τεχνικών μέχρι και σήμερα. Η ιδέα που υποστηρίζουν είναι πως μια πεποίθηση αναπαρίσταται από μια κανονική κατανομή τύπου:

$$p(x) = det(2\pi\Sigma)^{\frac{-1}{2}} \exp\frac{-1}{2}(\chi-\lambda)^T \Sigma^{-1}(\chi-\lambda)$$

Εικόνα 3 1

Όπου λ είναι ο mean και Σ είναι covariance.

Δύο από τις πιο δημοφιλείς και πολυμελετημένες μέθοδοι για την υλοποίηση ενός φίλτρου Bayes είναι το **φιλτρο Kalman** [3] και το **Extended Kalman Filter**. Το Kalman Filter, γνωστό και ως Linear Quadratic Estimation, είναι ένας αναδρομικός αλγόριθμος που προσπαθεί να εκτιμήσει την κατάσταση $x \in \mathbb{R}^N$ μιας διακριτού χρόνου ελεγχόμενης διαδικασίας η οποία υπόκειται σε μια γραμμική στοχαστική διαφορική εξίσωση. Το Kalman Filter κάνει χρήση μιας σειράς μετρήσεων στο χρόνο οι οποίες περιέχουν στατιστικό θόρυβο και άλλες ανακρίβειες για να παράγει εκτιμήσεις για τυχαίες μεταβλητές σε κάθε κβάντο χρόνου. Στο πλαίσιο SLAM οι μεταβλητές προς εκτίμηση είναι η θέση του ρομπότ και οι θέσεις των σημείων αναφοράς. Το φίλτρο έχει πάρει το όνομά του από το δημιουργό του Rudolf E. Kalman.

Το **Extended Kalman Filter** είναι μια μη-γραμμική έκδοση του Kalman Filter η οποία ξεπερνά την υπόθεση πως λαμβάνουμε γραμμική εκτίμηση των mean και covariance. Το μοντέλο αυτό συντελεί μια γενίκευση του γραμμικού Γκαουσιανού μοντέλου το οποίο κρύβεται πίσω από τα Φίλτρα Kalman. Ωστόσο, η κατανομή μας παύει να είναι Γκαουσιανή όταν γίνεται χρήση μη-γραμμικών συναρτήσεων. Άρα, το βήμα με το οποίο ενημερώνεται η κατανομή δεν μπορεί να υπολογιστεί μέσω κλειστού τύπου. Ως αποτέλεσμα το Extended Kalman Filter υπολογίζει μια προσέγγιση της πραγματικής κατανομής. Έτσι, το EKF κληρονομεί από τα φίλτρα Kalman τη βασική πεποίθηση για το πως αναπαρίσταται αλλά διαφέρει από αυτά μιας και η ίδια αναπαράσταση αυτών είναι επίσης προσεγγιστική. Είναι σημαντικό να πούμε πως οι αλγόριθμοι που βασίζονται στο EKF είναι βασισμένοι στα σημεία αναφοράς και χρησιμοποιούν αλγόριθμους μέγιστης πιθανότητας για συσχέτιση δεδομένων. Η αποτελεσματικότητα των EKF πηγάζει από το γεγονός πως εκτιμούν μια πλήρως συσχετισμένη σε χάρτες με σημεία αναφοράς σε συνδυασμό με τις στάσεις ρομπότ(pose). Από την άλλη, τέτοιες μέθοδοι υποθέτουν προκατασκευασμένες προκείμενες για το θόρυβο των αισθητήρων και το μοντέλο κίνησης του ρομπότ που ωθούν το μοντέλο των προσεγγίσεων να γίνεται ολοένα και πιο ευάλωτο σε αναντιστοιχίες αφού εκτιμά λάθος τον πραγματικό κόσμο. Επίσης, πρέπει να σημειωθεί πως κάθε ένα σημείο αναφοράς στο χάρτη θα πρέπει να είναι μοναδικά αναγνωρίσιμο.

Μια βελτιστοποιημένη έκδοση του EKF προτάθηκε από τον Jose Guivant και τον Eduardo Net. Αυτοί εισήγαγαν το **Compressed Extended Kalman Filter(CEKF)**. Η προσέγγιση αυτή μειώνει την υπολογιστική πολυπλοκότητα διαιρώντας τον υπολογισμό σε 2 μέρη: το τοπικο ενεργό μέρος και τα υπόλοιπα. Έτσι, το CEKF ενημερώνει πλήρως μόνο το τοπικό ενεργό μέρος σε κάθε βήμα ενώ η απαραίτητη πληροφορία για τις υπόλοιπες καταστάσεις συμπιέζεται σε έναν δευτερεύων πίνακα συντελεστών. Αυτό ενισχύει την αποδοτικότητα του αλγορίθμου και χωρίς να μειώνει την ακρίβεια από την οποία χαρακτηρίζονται τους πλήρεις SLAM αλγόριθμους.

Εναλλακτική στις τεχνικές Kalman είναι τα Particle Filters. **Particle Filters(PF)** ή **Sequential Monte Carlo(CMS)** είναι ένα σύνολο αλγορίθμων Monte Carlo που χρησιμοποιούνται για να επιλύσουν προβλήματα φιλτραρίσματος με Bayesian statistical inference. Το Particle Filter έιναι σχεδιασμένο για τυχαίες μεταβλητές συνδεδεμένες σε μια Markov Chain, όπου το σύστημα αποτελείται από παρατηρήσιμες και μη μεταβλητές. Οι μεταβλητές παρατήρησης είναι διασυνδεδεμένες με τις κρυφές μεταβλητές(καταστάσεις) μέσω μιας μορφής συνάρτησης η οποία είναι γνωστή. Μια συνάρτηση πυκνότητας πιθανότητας PDF χρησιμοποιείται για να αναπαραστήσουμε τις πληροφορίες για τις καταστάσεις.

Το **DP-SLAM** [3] καθιστά έναν on-line αλγόριθμο που βασίζεται στα particle filters και παράγει τους χάρτες grid. Ο σκοπός του είναι να πετύχει καλή ακρίβεια SLAM χωρίς σημεία αναφοράς ενώ

ταυτόχρονα μειώνει την εκτεταμένη χρήση υπολογιστικής ισχύος. Αυτό το πετυχαίνει με το να αποφεύγει τις επανειλημμένες αντιγραφές του χάρτη για κάθε νέο σημείο που δημιουργείται. Λειτουργεί διατηρώντας μια από κοινού πιθανότητα για το χάρτη και για τη στάση του ρομπότ. Αυτό επιτρέπει στον αλγόριθμο να διατηρεί αβεβαιότητα για το χάρτη σε ένα εύρος πολλαπλών βημάτων/κβάντων χρόνου έως ότου όλα τα σημεία με τις αβεβαιότητες επιλυθούν. Ταυτοχρόνως, αποτρέπει σφάλματα από το να συσσωρεύονται στο χάρτη με την πάροδο του χρόνου.

Ο **Fast Slam** [5]**,** [6] είναι μια εναλλακτική μέθοδος για στοχαστική χαρτογράφηση και εντοπισμό. Ο Fast Slam αποτελεί μια ενδιαφέρουσα μέθοδο καθώς είναι ένας αλγόριθμος υβρίδιο που χρησιμοποιεί και Kalman και Particle Filters. Η βασική ιδέα πίσω από τον αλγόριθμο είναι πως προσπαθεί να εκμεταλλευτεί την παράμετρο ανεξαρτησίας από περιορισμούς του SLAM μοντέλου για να διαιρέσει το πρόβλημα σε μικρότερα προβλήματα αντίστοιχα μόνο εντοπισμού και μόνο χαρτογράφησης.

Οι αλγόριθμοι που μόλις αναφέραμε χρησιμοποιούν αισθητήρες απόστασης και οδομετρία για να εκτελέσουν την εξερεύνηση του χώρου. Ωστόσο, λόγω της δημοφιλίας των αλγορίθμων για την επίλυση του SLAM νέες τεχνικές σε συνδυασμό με αισθητήρες όρασης ξεπροβάλλουν συνεχώς. Τέτοιοι αλγόριθμοι συχνά αναφέρονται σαν Visual-SLAM. Ο **EKFM-SLAM** αλγόριθμος είναι ένας από αυτούς. Παρόλο που ο αλγόριθμος αυτός βασίζεται στον **EKF-SLAM** χρησιμοποιεί μια κάμερα σαν το μοναδικό αισθητήρα αντί για αισθητήρα απόστασης. Η μέθοδος Random Sample Consensus(RANSAC) χρησιμοποιείται για να ενσωματώσει την πληροφορία από το σύστημα της κάμερας, δηλαδή να εκτιμήσει την κίνηση της κάμερας μέσω οδομετρίας(Visual Odometry).

Κεφάλαιο 3.2: Τύποι αισθητήρων σε σύγχρονα ρομπότ

Εδώ παρουσιάζονται μερικές μεθόδους απόκτησης μετρήσεων με βάση τον εκάστοτε αισθητήρα που χρησιμοποιείται στο χώρο του SLAM. Επίσης, παρατίθενται αποσπάσματα που οπτικοποιούν καλύτερα τις παραμέτρους που χαρακτηρίζουν τις διάφορες συνιστώσες ενός συστήματος αυτόνομης πλοήγησης με βάση έναν ή περισσότερους αισθητήρες.

Εισαγωγή

Τον τελευταίο καιρό, έχει σημειωθεί ραγδαία ανάπτυξη των αυτόνομων κινητών ρομπότ (AMR). Λόγω της εξαιρετικής τους ικανότητας εργασίας, τα AMR χρησιμοποιούνται για την αντικατάσταση των ανθρώπων σε ορισμένα σενάρια εσωτερικών χώρων. Τα κινητά ρομπότ μπορούν να συμβάλουν στη μείωση του φόρτου της ανθρώπινης εργασίας και στη βελτίωση της αποδοτικότητας της παραγωγής. Έχουν χρησιμοποιηθεί ευρέως σε πολλές βιομηχανίες και υπηρεσίες, όπως η αποθήκευση, η εφοδιαστική, η υγειονομική περίθαλψη, η εξυπηρέτηση εστιατορίων και οι προσωπικές υπηρεσίες [15-17]. Τα βασικά στοιχεία της κινητής ρομποτικής αποτελούνται από τους τομείς της κίνησης, της αντίληψης, της γνώσης,

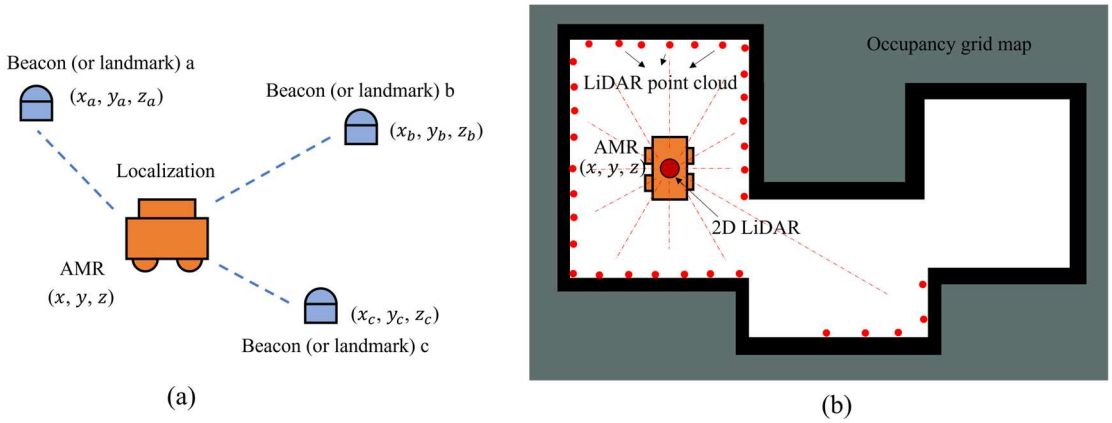
και την πλοήγηση. Το πρόβλημα της μετακίνησης αφορά κυρίως το σύστημα κίνησης του κινητού ρομπότ. Ο σχεδιασμός του συστήματος κίνησης βασίζεται στις απαιτήσεις των παρεχόμενων υπηρεσιών, όπως το περιβάλλον κίνησης, η δυνατότητα ελέγχου, η αποδοτικότητα, η σταθερότητα και άλλοι σχετικοί δείκτες. Η αντίληψη περιλαμβάνει την απόκτηση πληροφοριών σχετικά με το περιβάλλον εργασίας του κινητού ρομπότ και τον εαυτό του. Η νόηση περιλαμβάνει την ανάλυση και την επεξεργασία δεδομένων

από το σύστημα αντίληψης και την παροχή λύσεων ελέγχου στο σύστημα κίνησης για την εκτέλεση των εργασιών του κινητού ρομπότ. Η πλοήγηση του κινητού ρομπότ βασίζεται στην αντίληψη, τη νόηση και τον έλεγχο κίνησης για να μετακινηθεί από ένα σημείο εκκίνησης σε ένα σημείο-στόχο της εργασίας σε ένα περιβάλλον εργασίας, γνωστό ή άγνωστο. Η αντίληψη αποτελεί κρίσιμο μέρος της έρευνας στην κινητή ρομποτική. Για να μπορούν τα κινητά ρομπότ να εκτελούν με ασφάλεια και αποτελεσματικότητα κινητές εργασίες, πρέπει να αντιλαμβάνονται το περιβάλλον τους και να λαμβάνουν αποφάσεις με βάση αυτές τις πληροφορίες. Εάν ένα κινητό ρομπότ δεν μπορεί να αντιληφθεί το περιβάλλον του με ακρίβεια και αποτελεσματικότητα, τότε δεν θα είναι σε θέση να εκτελέσει ακόμη και απλές εργασίες.

Το σύστημα αντίληψης του κινητού ρομπότ χρησιμοποιεί σχετικές τεχνικές ανίχνευσης για να παρέχει πληροφορίες σχετικά με το περιβάλλον, το ίδιο το ρομπότ και τη σχέση μεταξύ των δύο. Οι πληροφορίες αυτές χρησιμοποιούνται στη συνέχεια για τον σχεδιασμό της διαδρομής, τον έλεγχο της κίνησης του ρομπότ και τελικά την ολοκλήρωση της εργασίας πλοήγησης. Οι κύριες εργασίες του συστήματος αντίληψης περιλαμβάνουν τον εντοπισμό, τη δημιουργία χάρτη, την ανίχνευση αντικειμένων κ.λπ.

Εντοπισμός(localization) είναι η διαδικασία εκτίμησης της θέσης ενός κινητού ρομπότ σε σχέση με τον περιβάλλοντα χώρο εργασίας κατά τη διάρκεια της κίνησής του. Οι πληροφορίες αυτές χρησιμεύουν ως βάση για την πλοήγηση του κινητού ρομπότ [20-22]. Οι τεχνικές εντοπισμού κινητών ρομπότ περιλαμβάνουν την αδρανειακή πλοήγηση, τα παγκόσμια συστήματα εντοπισμού θέσης (GPS), την πλοήγηση με ενεργό φάρο, την πλοήγηση με σημεία αναφοράς(ορόσημα) και την αντιστοίχιση μοντέλων(model-matching).

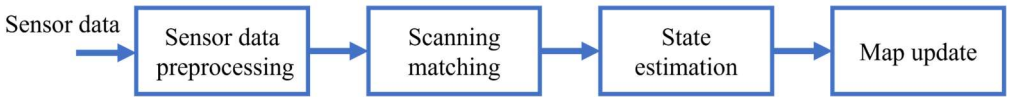
- Η **αδρανειακή πλοήγηση** είναι μια μέθοδος πλοήγησης που χρησιμοποιεί δεδομένα από αισθητήρες που ανταποκρίνονται σε αδρανειακές δυνάμεις.
- Το **παγκόσμιο σύστημα εντοπισμού θέσης** (**GPS**) είναι μια δορυφορική τεχνολογία που παρέχει ακριβείς πληροφορίες σχετικά με το γεωγραφικό μήκος, το γεωγραφικό πλάτος και το υψόμετρο για οποιαδήποτε θέση στη Γη. Είναι μια εξαιρετικά αποτελεσματική τεχνική για τον εντοπισμό κινητών ρομπότ σε εξωτερικά περιβάλλοντα. Ωστόσο, το GPS είναι περιορισμένης χρήσης για τον εντοπισμό σε εσωτερικούς χώρους, λόγω προβλημάτων μη οπτικής επαφής και αποκλεισμού του σήματος σε πολύπλοκα περιβάλλοντα [23].
- Η **πλοήγηση με ενεργό φάρο** (**active-beacon**) και η πλοήγηση με σημεία αναφοράς(ορόσημα) χρησιμοποιούν για τον εντοπισμό σχετικές σχέσεις θέσης με φάρους ή σημεία αναφοράς(ορόσημα) με γνωστές θέσεις στο περιβάλλον, όπως φαίνεται στο Σχήμα 1α. Η αντιστοίχιση μοντέλων(model-matching) είναι μια μέθοδος εντοπισμού θέσης με αντιστοίχιση δεδομένων αισθητήρων με συγκεκριμένα δεδομένα μοντέλων χαρτών. Για παράδειγμα, στον εντοπισμό 2D LiDAR, η μέτρηση LiDAR λαμβάνει δεδομένα 2D νέφους σημείων(point cloud) λέιζερ και το μοντέλο χάρτη είναι ένας χάρτης πλέγματος πληρότητας(occupancy grid map). Ο εντοπισμός του κινητού ρομπότ πραγματοποιείται με την αντιστοίχιση των δεδομένων νέφους σημείων(point cloud) με τον χάρτη πλέγματος πληρότητας(occupancy grid map), όπως φαίνεται στο Σχήμα 1β.



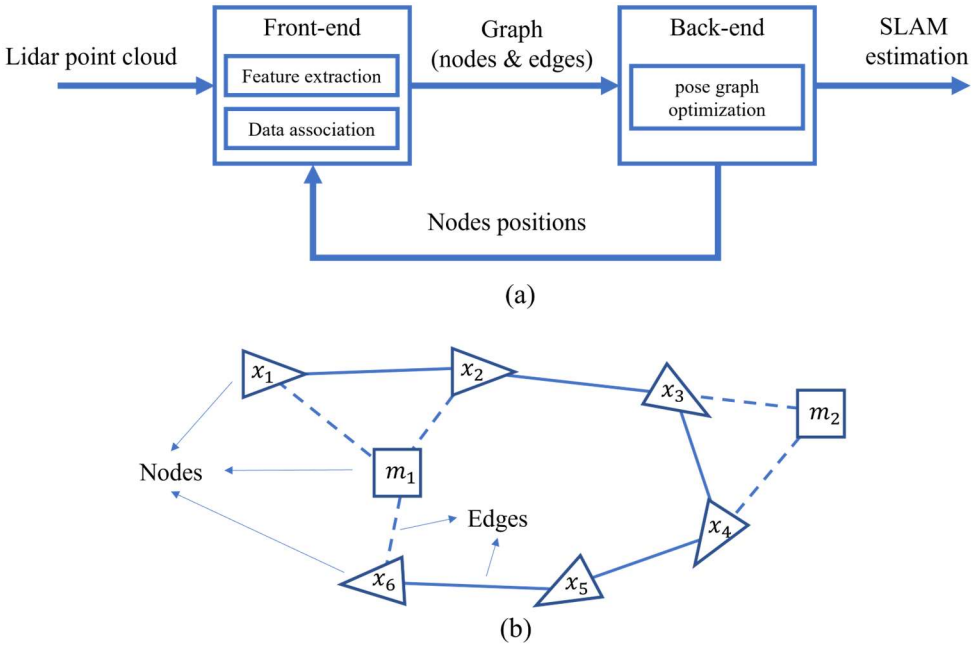
Σχήμα 1. Σχηματικό διάγραμμα της μεθόδου εντοπισμού: (α) η μέθοδος εντοπισμού με βάση τον ενεργό φάρο(active-beacon) ή τα σημεία αναφοράς(ορόσημα) και (β) η μέθοδος εντοπισμού με βάση την αντιστοίχιση μοντέλων 2D LiDAR.

Η **κατασκευή χαρτών** συνήθως περιλαμβάνει ταυτόχρονο εντοπισμό και χαρτογράφηση (SLAM), που αποτελεί κρίσιμο ερευνητικό πεδίο για τα κινητά ρομπότ. Η SLAM είναι μια τεχνική που χρησιμοποιείται από τα κινητά ρομπότ για τον προσδιορισμό της θέσης τους σε ένα άγνωστο περιβάλλον, ενώ ταυτόχρονα κατασκευάζεται ένας ακριβής χάρτης του περιβάλλοντος. Η συνιστώσα εντοπισμού εκτιμά τη θέση του ρομπότ σε έναν υπάρχοντα χάρτη, ενώ η συνιστώσα χαρτογράφησης κατασκευάζει τον χάρτη του περιβάλλοντος [24-26]. Οι δύο συνιστώσες είναι αλληλοεξαρτώμενες στην ολοκλήρωση της διαδικασίας κατασκευής του χάρτη και στην ενίσχυση της ακρίβειας μέσω συνεχών επαναλήψεων. Οι δύο πιο συχνά χρησιμοποιούμενοι αισθητήρες στο SLAM είναι το LIDAR και οι κάμερες όρασης. Οι τεχνικές SLAM μπορούν να ταξινομηθούν σε δύο κύριες κατηγορίες με βάση τη βραχυπρόθεσμη και μακροπρόθεσμη επεξεργασία βελτιστοποίησης: μέθοδοι που βασίζονται σε φίλτρα και μέθοδοι που βασίζονται στη βελτιστοποίηση γραφημάτων. Οι αλγόριθμοι SLAM που βασίζονται σε φίλτρα χρησιμοποιούν συνήθως διάφορους τύπους φίλτρων για την εκτίμηση και τη βελτιστοποίηση της τροχιάς και του χάρτη του ρομπότ, όπως φαίνεται στο Σχήμα 2. Οι δύο πιο συνηθισμένοι αλγόριθμοι που βασίζονται σε φίλτρα είναι ο αλγόριθμος SLAM με φίλτρο Kalman και ο αλγόριθμος SLAM με φίλτρο σωματιδίων(particle filter).

Ο **αλγόριθμος βελτιστοποίησης γραφημάτων** αποτελείται κυρίως από δύο διαδικασίες, το front-end και το back-end, όπως φαίνεται στο Σχήμα 3α. Ο αλγόριθμος βελτιστοποίησης γραφήματος μοντελοποιεί την τροχιά του ρομπότ και την τοπολογία του περιβάλλοντος ως γράφημα και ελαχιστοποιεί το σφάλμα βελτιστοποιώντας τους κόμβους και τις ακμές του γραφήματος για να εκτιμήσει τη θέση του ρομπότ και να αποκτήσει ένα μαο του περιβάλλοντος, όπως φαίνεται στο Σχήμα 3β. Οι κόμβοι αντιπροσωπεύουν γενικά τις μεταβλητές που πρέπει να βελτιστοποιηθούν στη διαδικασία SLAM του αισθητήρα, όπως η θέση του αισθητήρα, η θέση του σημείου χαρακτηριστικών, η εκτιμώμενη τροχιά κ.λπ. Οι ακμές αντιπροσωπεύουν τα σφάλματα ή τους περιορισμούς μεταξύ των μεταβλητές. Ο αλγόριθμος βελτιστοποίησης επαναλαμβάνεται για την ελαχιστοποίηση του σφάλματος των ακμών ώστε να βρεθεί η βέλτιστη λύση [27].



Σχήμα 2. Σχηματικό διάγραμμα της μεθόδου SLAM με βάση το φίλτρο.



Σχήμα 3. Σχηματική απεικόνιση της μεθόδου SLAM που βασίζεται στη βελτιστοποίηση γραφημάτων: (α) αναπαράσταση μιας διαδικασίας SLAM με βελτιστοποίηση γραφημάτων και (β) σχηματική αναπαράσταση του αλγορίθμου βελτιστοποίησης γραφημάτων.

Με την ανάπτυξη της τεχνητής νοημοσύνης, η εμφάνιση της ενισχυτικής μάθησης(reinforcement learning) επιτρέπει στα κινητά ρομπότ να απορρίπτουν σταδιακά τους περιορισμούς που σχετίζονται με τους εκ των προτέρων χάρτες και να επιτυγχάνουν την κρίση των ενεργειών πλοήγησης μόνο μέσω της εισαγωγής εικόνων [28,29]. Η βασική λογική είναι: να δημιουργηθεί μια σειρά από μηχανισμούς ανταμοιβής εξερεύνησης γύρω από το πρόβλημα εξερεύνησης του κινητού ρομπότ, όπως: ανταμοιβή άφιξης, ανταμοιβή σύγκρουσης, ανταμοιβή περιέργειας κ.λπ. και να δημιουργηθεί το σχετικό νευρωνικό δίκτυο μέσω μεθόδων ενισχυτικής μάθησης και να εκπαιδευτεί το δίκτυο χρησιμοποιώντας το σύνολο δεδομένων. Τέλος, υλοποιείται η απόφαση πλοήγησης του αυτόνομου κινητού ρομπότ.

Για την αποτελεσματική και ακριβή εκτέλεση της αποστολής πλοήγησης, το κινητό ρομπότ πρέπει να αποφεύγει τη σύγκρουση με εμπόδια. Η αποφυγή εμποδίων είναι ένα κρίσιμο έργο για τα αυτόνομα κινητά ρομπότ που κινούνται σε αβέβαια περιβάλλοντα [30,31]. Η αποφυγή εμποδίων επικεντρώνεται στο πρόβλημα με την απόκτηση της παγκόσμιας πληροφορίας θέσης του ρομπότ σε σχέση με το περιβάλλον, ώστε το κινητό ρομπότ να χρειάζεται να λάβει μόνο την τοπική πληροφορία θέσης κοντινών αντικειμένων γύρω από το ρομπότ σε σχέση με το ίδιο. Η ανίχνευση εμποδίων είναι ζωτικής σημασίας και για τα συστήματα αντίληψης. Το σύστημα αντίληψης του κινητού ρομπότ εκτελεί σχεδιασμό διαδρομής για την αποφυγή εμποδίων με βάση τις πληροφορίες απόστασης και κατεύθυνσης για τα

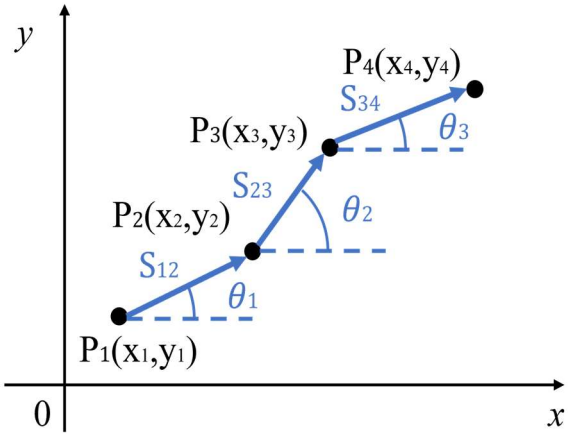
εμπόδια που λαμβάνονται από το σύστημα αντίληψης. Κατ' αρχήν, οι αισθητήρες απόστασης (αισθητήρες υπερήχων και LiDAR) είναι ιδιαίτερα κατάλληλοι για εργασίες ανίχνευσης εμποδίων. Επιπλέον, οι κάμερες όρασης μπορούν να αναγνωρίσουν εμπόδια στο περιβάλλον και ακόμη και να προβλέψουν την τροχιά δυναμικών εμποδίων.

Το σύστημα αντίληψης επιτυγχάνει αυτές τις λειτουργίες μέσω της χρήσης αισθητήρων και της εξαγωγής σχετικών πληροφοριών από τις μετρήσεις τους. Οι αισθητήρες που χρησιμοποιούνται για τη συλλογή δεδομένων κατηγοριοποιούνται σε δύο κύριες πτυχές:
ιδιοδεκτικοί/εξωδεκτικοί(proprioceptive/exteroceptive) αισθητήρες και
ενεργητικοί/παθητικοί(active/passive) αισθητήρες [17]. Οι ιδιοδεκτικοί(proprioceptive) αισθητήρες συλλέγουν δεδομένα σχετικά με το εσωτερικό του κινητού ρομπότ, όπως η ταχύτητα του κινητήρα, η γωνία περιστροφής των τροχών κ.λπ. Επί του παρόντος, οι κωδικοποιητές(encoders) [32], οι επιταχυντές(accelerometers) και γυροσκόπια(gyroscopes) [33] χρησιμοποιούνται ευρέως σε αυτόν τον τύπο αισθητήρα. Οι εξωαισθητικοί(exteroceptive) αισθητήρες αποκτούν πληροφορίες για το περιβάλλον γύρω από το κινητό ρομπότ, όπως εικόνες, ηχητικά κύματα και ασύρματα σήματα. Παραδείγματα εξωδεκτικών αισθητήρων περιλαμβάνουν αισθητήρες όρασης [34], αισθητήρες υπερήχων [35] και LiDAR [36]. Οι ενεργητικοί αισθητήρες στέλνουν ενέργεια στο εξωτερικό περιβάλλον και στη συνέχεια μετρούν την περιβαλλοντική ανατροφοδότηση, όπως οι αισθητήρες υπερήχων(sonar), LIDAR κ.λπ. Οι παθητικοί αισθητήρες μετρούν την περιβαλλοντική ενέργεια που εισέρχεται στον αισθητήρα, όπως οι αισθητήρες όρασης (κάμερες CCD (Charge Coupled Device) ή CMOS (Complementary Metal Oxide Semiconductor)).

Κεφάλαιο 3.2.1: Επισκόπηση τεχνολογιών ανίχνευσης με έναν αισθητήρα

Μονάδα αδρανειακών μετρήσεων (IMU)

Μια IMU είναι μια ηλεκτρονική συσκευή που ενσωματώνει ένα επιταχυνσιόμετρο, ένα γυροσκόπιο και ένα μαγνητόμετρο για τη μέτρηση της επιτάχυνσης, της γωνιακής ταχύτητας και της κατεύθυνσης του γεωμαγνητικού πεδίου ενός αντικειμένου [37]. Η IMU είναι ο αισθητήρας ιδιοδεκτικότητας. Το αδρανειακό σύστημα πλοήγησης (INS) είναι ένα σύστημα πλοήγησης με νεκρό σημείο επαναφοράς που χρησιμοποιεί την αδρανειακή μονάδα μέτρησης (IMU) και χρησιμοποιείται συνήθως για την πλοήγηση κινητών ρομπότ. Οι παράμετροι προσανατολισμού που μετρώνται με μαγνητόμετρο έχουν χαμηλή ακρίβεια και είναι δύσκολο να χρησιμοποιηθούν για ακριβείς εργασίες πλοήγησης. Ως εκ τούτου, οι πιο συχνά χρησιμοποιούμενες παράμετροι στις IMU είναι η επιτάχυνση και η γωνιακή ταχύτητα. Η αρχή του νεκρού υπολογισμού παρουσιάζεται στο Σχήμα 4. Με τη χρήση της εξεταζόμενης IMU, η τροχιά ενός εσωτερικού AMR προσδιορίζεται με την ολοκλήρωση της επιτάχυνσης και της γωνιακής ταχύτητας με βάση την αρχική θέση, κατεύθυνση και ταχύτητα.



Σχήμα 4. Νεκρός υπολογισμός IMU.

Μια IMU έχει δύο τυπικά πλεονεκτήματα:

- Υψηλό ρυθμό ενημέρωσης εξόδου
- Ανοσία σε εξωτερικές παρεμβολές.

Αυτά τα πλεονεκτήματα είναι αναντικατάστατα όταν απαιτείται υψηλός ρυθμός ενημέρωσης δεδομένων ή υψηλή αξιοπιστία ή όταν το εξωτερικό σήμα που λαμβάνεται από το σύστημα κινητού ρομπότ είναι αναξιόπιστο.

Ωστόσο, οι IMU έχουν περιορισμούς όταν εφαρμόζονται σε κινητά ρομπότ. Οι IMU έχουν δύο τυπικά μειονεκτήματα:

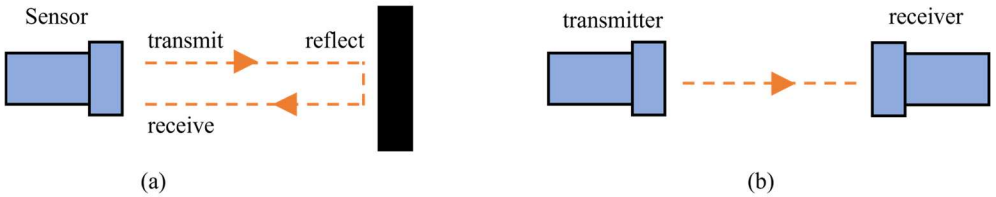
- Η διαδικασία υπολογισμού πρέπει να εξαρτάται από τις αρχικές συνθήκες.
- Το σφάλμα πλοήγησης αυξάνεται με το χρόνο, γεγονός που οφείλεται στην ανάγκη διπλής ολοκλήρωσης των μετρήσεών του.

Θα πρέπει να αναφερθεί ότι η διάρκειά και η ακρίβεια του αισθητήρα έχουν σημαντικό αντίκτυπο στην ακρίβεια της πλοήγησης με βάση την IMU [37]. Για να ξεπεραστούν αυτά τα μειονεκτήματα, μια IMU χρησιμοποιείται συχνά σε συγχώνευση με άλλους αισθητήρες και ως βοηθητικές μετρήσεις σε άλλες μέθοδοι πλοήγησης. Οι L. Luo κ.ά. [38] σχεδίασαν μια μέθοδο εντοπισμού θέσης που συνδυάζει ένα IMU, ένα Bluetooth και ένα μαγνητόμετρο για το πρόβλημα εντοπισμού θέσης κινητών ρομπότ. Στην προαναφερθείσα μέθοδο, το αδρανειακό σύστημα πλοήγησης (INS) χρησιμεύει ως πυρήνας και ο σταθμός βάσης εντοπισμού θέσης Bluetooth AOA παρατηρεί τη θέση, ενώ το μαγνητόμετρο παρατηρεί τη γωνία πορείας. Οι Alatise M. B. et al. [39] συνένωσαν τους αισθητήρες IMU και όρασης προκειμένου να επιτύχουν ακριβή εντοπισμό θέσης ενός κινητού ρομπότ. Τα δεδομένα του επιταχυνσιόμετρου χρησιμοποιούνται ως είσοδοι στον έλεγχο, ενώ τα δεδομένα του γυροσκοπίου και της εικόνας χρησιμοποιούνται ως παρατηρήσεις για την εκτίμηση της θέσης πόζας με τη χρήση εκτεταμένου φίλτρου Kalman. Οι Zhang κ.ά. [40] ανέπτυξαν ένα αλγοριθμικό πλαίσιο για τον εντοπισμό κινητών ρομπότ σε εσωτερικούς χώρους συνδυάζοντας ένα οδόμετρο τροχού, μια αδρανειακή μονάδα πλοήγησης (IMU) και Ultra-Wideband (UWB). Το πλαίσιο χρησιμοποιεί μια μέθοδο συγχώνευσης με εκτεταμένο φιλτράρισμα Kalman, όπου το οδόμετρο τροχού παρέχει πληροφορίες εντοπισμού ως πρόβλεψη και η IMU και η UWB παρέχουν πληροφορίες γωνίας στάσης και θέσης ως παρατηρήσεις.

Αισθητήρας υπερήχων/Sonar

Ο υπέρηχος αναφέρεται σε ηχητικά κύματα που δονούνται σε συχνότητες άνω των 20 kHz, οι οποίες βρίσκονται πέρα από το ανώτατο όριο της ανθρώπινης ακοής. Το sonar χρησιμοποιεί επίσης υπερήχους. Οι αισθητήρες υπερήχων είναι συνήθως αισθητήρες μέτρησης απόστασης. Οι αισθητήρες μέτρησης απόστασης μπορούν ταξινομηθούν σε δύο τύπους μεθόδων μέτρησης: τη μέθοδο μέτρησης απόστασης τύπου ανάκλασης και τη μέθοδο μέτρησης απόστασης με μονόδρομο σήμα υπερήχων, όπως φαίνεται στο Σχήμα 5. Στη μέθοδο τύπου ανάκλασης, ο αισθητήρας στέλνει ένα σήμα το οποίο στη συνέχεια ανακλάται όταν συναντήσει ένα εμπόδιο. Στη συνέχεια, ο αισθητήρας λαμβάνει το ανακλώμενο σήμα για να μετρήσει την απόσταση μεταξύ του αισθητήρα και του εμποδίου. Η μέθοδος μέτρησης απόστασης με μονόδρομο σήμα υπερήχων περιλαμβάνει την τοποθέτηση του πομπού και του δέκτη σε διαφορετικές θέσεις. Ο δέκτης μετρά την απόσταση μεταξύ των δύο δεχόμενος το σήμα από τον πομπό. Η μέτρηση της απόστασης με υπερήχους μετρά την απόσταση χρησιμοποιώντας τρία βασικά κριτήρια:

- το χρόνο
- τη φάση
- την ακουστική δόνηση.



Σχήμα 5. Μέθοδος μέτρησης απόστασης με αισθητήρα: (α) μέθοδος μέτρησης απόστασης τύπου ανάκλασης. (β) μέθοδος μέτρησης απόστασης μονής κατεύθυνσης.

Οι αισθητήρες υπερήχων χρησιμοποιούν συνήθως μια μέθοδο μέτρησης απόστασης τύπου ανάκλασης για τον προσδιορισμό της απόστασης από ένα αντικείμενο, η οποία χρησιμοποιείται για τον εντοπισμό θέσης και την πλοήγηση κινητών ρομπότ. Ωστόσο, το πλάτος της δέσμης υπερήχων του αισθητήρα sonar είναι πολύ μεγάλο για τον ακριβή προσδιορισμό της κατεύθυνσης του αντικειμένου. Επιπλέον, η κατοπτρική ανάκλαση στις επιφάνειες των αντικειμένων οδηγεί συχνά σε ανακριβείς υπολογισμούς της απόστασης, μειώνοντας την αξιοπιστία των μετρήσεων απόστασης από αισθητήρες σόναρ. Δεν είναι δυνατόν να κατασκευαστεί ένας πλήρης χάρτης πλέγματος χρησιμοποιώντας λανθασμένα μετρημένα δεδομένα σόναρ, όταν η συχνότητα των λανθασμένων μετρήσεων απόστασης είναι πολύ υψηλή. Για να ξεπεραστεί αυτό το , οι S.-J. Lee et al. [41] σχεδίασαν μια μέθοδο για την κατασκευή χαρτών πλέγματος κατάληψης για κινητά ρομπότ χρησιμοποιώντας δεδομένα σόναρ. Η μέθοδος αυτή αποδίδει βάρη σε κάθε σημείο δεδομένων σόναρ με βάση τη γεωμετρική του αξιοπιστία, ώστε να ελαχιστοποιείται η επίδραση των εσφαλμένων δεδομένων σόναρ στην κατασκευή του χάρτη πλέγματος. Τα ανώμαλα δεδομένα σόναρ εντοπίζονται και τους αποδίδεται χαμηλότερη τιμή γεωμετρικής αξιοπιστίας μέσω μορφολογικής σύγκρισης με γειτονικά δεδομένα σόναρ. Οι H. Liu κ.ά. [42] χρησιμοποίησαν αισθητήρες σόναρ για να ξεπεράσουν τους περιορισμούς της απόδοσης των αισθητήρων συσσωρεύοντας δεδομένα σόναρ για να επιτύχουν εντοπισμό σε επίπεδο δωματίου. Οι Y. Liu κ.ά. [43] παρουσίασαν ένα κινητό ρομπότ με βάση το LEGO MINDSTORM NXT, στο οποίο ένας αισθητήρας υπερήχων ήταν τοποθετημένος σε έναν περιστρεφόμενο κινητήρα 360◦ και μια προσέγγιση φιλτραρίσματος σωματιδίων χρησιμοποιήθηκε για

την υλοποίηση της λειτουργίας εντοπισμού του. Οι Mingqi Shen κ.ά. [44] παρείχαν ένα σύστημα εντοπισμού μέθοδος για αυτόνομα κινητά ρομπότ που βασίζονται σε πολλαπλούς αισθητήρες υπερήχων. Η μέθοδος υπολογίζει τις συντεταγμένες του στόχου με βάση τον λόγο τριών χρόνων πτήσης(ToF) που μετρούνται από τρεις αισθητήρες υπερήχων. Αυτό βελτιώνει την ακρίβεια εντοπισμού αποφεύγοντας την επίδραση της θερμοκρασίας του περιβάλλοντος στον εντοπισμό.



LEGO MINDSTORM NXT

Επιπλέον, οι αισθητήρες υπερήχων μπορούν επίσης να χρησιμοποιηθούν για τη μέθοδο μέτρησης της απόστασης σε μία κατεύθυνση, η οποία έχει εφαρμοστεί στον εντοπισμό κινητών ρομπότ εσωτερικού χώρου. Οι R. Li κ.ά. [35] εγκατέστησαν έναν δέκτη υπερήχων στο ρομπότ και είχαν πολλαπλούς γεννήτορες υπερήχων στη σκηνή. Για την επίτευξη του εντοπισμού του κινητού ρομπότ χρησιμοποιείται η αρχή του τριμερούς εντοπισμού. Οι Chen-Chien Hsu κ.ά. [45] παρουσίασαν μια μέθοδο εντοπισμού που βασίζεται στην πανκατευθυντική ανίχνευση υπερήχων. Η προτεινόμενη μέθοδος χρησιμοποιεί ανακλαστικούς κώνους για τη δημιουργία μιας διαδιδόμενης δέσμης υπερήχων 360◦ για την αντιμετώπιση του περιορισμού της γωνίας ανίχνευσης των συμβατικών αισθητήρων υπερήχων. Τα υπερηχητικά κύματα εκπέμπονται απευθείας σε έναν κώνο, ο οποίος τα ανακλά προς όλες τις κατευθύνσεις. Το υπερηχητικό σήμα λαμβάνεται από τον εγκατεστημένο δέκτη στο περιβάλλον. Η θέση του κινητού ρομπότ προσδιορίζεται μέσω της μεθόδου ToF (χρόνος πτήσης) και της μεθόδου εντοπισμού τριμερούς εντοπισμού.

Οι αισθητήρες υπερήχων είναι ευαίσθητοι στον περιβαλλοντικό θόρυβο και στην υποβάθμιση της ισχύος σε μεγάλη απόσταση [46] και δεν είναι αρκετά ακριβείς για τη μέτρηση λεπτομερών χαρακτηριστικών του περιβάλλοντος. Υπάρχουν σχετικά λίγες περιπτώσεις όπου οι αισθητήρες υπερήχων χρησιμοποιούνται για τον εντοπισμό θέσης σε εφαρμογές πλοήγησης υψηλής ακρίβειας σε εσωτερικούς χώρους. Ωστόσο, οι αισθητήρες υπερήχων έχουν ένα αξιοσημείωτο πλεονέκτημα στην ανίχνευση εμποδίων. Είναι εξαιρετικά ακριβείς στην ανίχνευση αντικειμένων με υψηλή διαπερατότητα ή αντανάκλαση, όπως το γυαλί και οι καθρέφτες, κάτι που μπορεί να αποτελέσει πρόκληση για πολλούς οπτικούς αισθητήρες. Οι H. Takai κ.ά. [47] χρησιμοποίησαν υπερηχητικό στερεοφωνικό σόναρ και έναν ενιαίο αισθητήρα εικόνας για την ανίχνευση εμποδίων στο χώρο εργασίας. J.-H. Jean κ.ά. [48] χρησιμοποίησαν κυρίως τις πληροφορίες περιβαλλοντικής απόστασης που συλλέγονται από αισθητήρες υπερήχων για να καθοδηγήσουν το ρομπότ κατά μήκος της βασικής γραμμής του τοίχου χρησιμοποιώντας έναν ελεγκτή διεύθυνσης που βασίζεται στη μέθοδο του πεδίου δυναμικού. Οι Grami T. et al. [49] σχεδίασαν ένα κινητό ρομπότ εξοπλισμένο με επτά αισθητήρες υπερήχων και χρησιμοποίησαν

μια προσέγγιση φιλτραρίσματος σωματιδίων για την υλοποίηση της λειτουργίας εντοπισμού του. Το κινητό ρομπότ μπορεί να κινείται ελεύθερα αποφεύγοντας τα εμπόδια. Οι Maryna Derkach et al. [50] χρησιμοποίησαν τέσσερις αισθητήρες υπερήχων για την αποφυγή εμποδίων σε κινητά ρομπότ. Ο προτεινόμενος αλγόριθμος τους χρησιμοποιεί ένα γραμμικό αναδρομικό φίλτρο Kalman για την επεξεργασία των δεδομένων των αισθητήρων, επιτρέποντας στο ρομπότ να αποφεύγει επιπλέον εμπόδια.

Αισθητήρας υπέρυθρων

Η υπέρυθρη ακτινοβολία είναι ένα ηλεκτρομαγνητικό κύμα με μήκος κύματος μεταξύ του ορατού φωτός και των μικροκυμάτων. Οι αισθητήρες υπέρυθρων χρησιμοποιούν έναν πομπό για την εκπομπή υπέρυθρης δέσμης και έναν δέκτη για τη μέτρηση της έντασης της δέσμης. Η ένταση της υπέρυθρης δέσμης μειώνεται καθώς αυξάνεται η απόσταση που διανύεται. Οι αισθητήρες υπέρυθρων μπορούν να χρησιμοποιηθούν για τη μέτρηση της απόστασης αξιολογώντας την ένταση της δέσμης, εκτός από τη μέθοδο ToF (χρόνος πτήσης). Παρόμοια με τους αισθητήρες υπερήχων, οι αισθητήρες υπέρυθρων χρησιμοποιούν μεθόδους μέτρησης απόστασης τύπου ανάκλασης και μονής κατεύθυνσης.

Υπάρχουν πολλές πρώτες μελέτες που εφαρμόζουν αισθητήρες υπέρυθρων στον εντοπισμό κινητών ρομπότ. Οι Eric Brassart κ.ά. [51] σχεδίασαν ένα σύστημα εντοπισμού κινητών ρομπότ βασισμένο σε υπέρυθρους πομπούς. Σε αυτό το σύστημα, ένας πομπός υπέρυθρων τοποθετημένος στην οροφή στέλνει ένα κωδικοποιημένο σήμα υπέρυθρων, το οποίο στη συνέχεια λαμβάνεται από μια κάμερα CCD τοποθετημένη στο κινητό ρομπότ. Η θέση του ρομπότ προσδιορίζεται με τη χρήση μεθόδων triangulation(τριγωνισμού) και trilateration(τριπλοποίησης). Οι J. Krejsa κ.ά. [52] παρουσίασαν μια μέθοδο εσωτερικού εντοπισμού για κινητά ρομπότ που χρησιμοποιεί υπέρυθρους πομπούς. Οι πομποί τοποθετούνται σε γνωστές θέσεις στο περιβάλλον και για τη λήψη των υπέρυθρων σημάτων χρησιμοποιείται μια συσκευή σάρωσης δακτυλιοειδούς φάρου με 16 δέκτες. Ο σαρωτής τοποθετείται στο ρομπότ για να μετρά τη σχετική γωνία μεταξύ του ρομπότ και του φάρου. Η θέση του κινητού ρομπότ εκτιμάται με τη συνένωση πληροφοριών προσανατολισμού με εντολές ελεγκτή κίνησης ή με ένα οδόμετρο μέσω ενός εκτεταμένου φίλτρου Kalman. Οι αισθητήρες υπέρυθρων μπορούν να χρησιμοποιηθούν για την παρακολούθηση στόχων όταν ο πομπός και ο δέκτης είναι τοποθετημένοι σε διαφορετικά ρομπότ. Οι Tzuu-Hseng S. Li κ.ά. [53] σχεδίασαν ένα σύστημα ελέγχου παρακολούθησης για ένα αυτόνομο κινητό ρομπότ με τη χρήση αισθητήρων υπέρυθρων. Το σύστημα περιλαμβάνει ένα κινητό ρομπότ-στόχο με έναν ανακλαστήρα υπέρυθρων και ένα κινητό ρομπότ παρακολούθησης με δέκτη υπέρυθρων και ανακλαστικό αισθητήρα. Το τελευταίο σχεδιάστηκε για να παρακολουθεί το ρομπότ-στόχο. Οι Juang J. G. et al. [54] σχεδίασαν τροχήλατα κινητά ρομπότ με ενσωματωμένους αισθητήρες υπερήχων και υπέρυθρων που μπορούν να κινούνται κατά μήκος τοίχων και να αποφεύγουν εμπόδια. Οι αισθητήρες υπερήχων και υπέρυθρων χρησιμοποιήθηκαν για την ανίχνευση εμποδίων και τον εντοπισμό άγνωστων περιβαλλόντων. Η τεχνολογία εντοπισμού με βάση τις υπέρυθρες είναι επί του παρόντος ώριμη. Ωστόσο, είναι δύσκολο για αυτήν να διεισδύσει σε αντικείμενα και είναι ευάλωτη στις ιδιαιτερότητες αυτές του περιβάλλοντός της. Επιπλέον, τα εμπόδια μπορούν να μπλοκάρουν άμεσα τις υπέρυθρες ακτίνες στη μέθοδο εντοπισμού που βασίζεται σε υπέρυθρους ραδιοπομπούς. Ως αποτέλεσμα, η εφαρμογή των αισθητήρων υπέρυθρων στον εντοπισμό κινητών ρομπότ είναι περιορισμένη.

Στην ανίχνευση εμποδίων, οι αισθητήρες υπέρυθρων χρησιμοποιούνται συχνά λόγω της ταχείας απόκρισής τους και του μικρού μεγέθους τους [55-57]. Οι Jianwei Zhao κ.ά. [58] σχεδίασαν ένα κινητό ρομπότ πολλαπλών αισθητήρων που χρησιμοποιεί αισθητήρες υπέρυθρων τοποθετημένους μπροστά από τους δύο μπροστινούς τροχούς στο κάτω μέρος του ρομπότ για την ανίχνευση λάκκων.

LiDAR

Το LiDAR (Light Detection and Ranging) χρησιμοποιεί ακτίνες λέιζερ για τη μέτρηση της θέσης και της απόστασης των αντικειμένων-στόχων. Πολλαπλές ακτίνες λέιζερ αποστέλλονται από πομπούς και, όταν πέφτουν πάνω σε αντικείμενα, ανακλώνται πίσω. Οι ανακλώμενες ακτίνες λαμβάνονται από τους δέκτες LiDAR. Το LiDAR μετρά την απόσταση που διανύει κάθε δέσμη χρησιμοποιώντας το χρόνο πτήσης (ToF) και συνδυάζει τις πληροφορίες απόστασης από όλες τις δέσμες λέιζερ για να λάβει πληροφορίες για το περιβάλλον. Το LiDAR λειτουργεί παρόμοια με το ραντάρ, αλλά με μεγαλύτερη ακρίβεια, ταχύτητα μέτρησης και απόσταση μέτρησης σε σύγκριση με το συμβατικό ραντάρ.

Στην πλοήγηση των κινητών ρομπότ εσωτερικού χώρου, τα κινητά ρομπότ χρησιμοποιούν κυρίως 2D LIDAR για την ανίχνευση περιβαλλοντικών πληροφοριών, ενώ το 3D LIDAR χρησιμοποιείται συχνότερα στον τομέα της ευφυούς οδήγησης, επειδή χρειάζεται περισσότερες πληροφορίες για επεξεργασία [52]. Το LiDAR χρησιμοποιείται ευρέως για τον εντοπισμό κινητών ρομπότ λόγω της εξαιρετικής ακρίβειας μέτρησης της απόστασης. Ο εντοπισμός με LiDAR βασίζεται σε έναν χάρτη περιβάλλοντος [53].

Καθώς η μέτρηση της απόστασης LiDAR βασίζεται στην αρχή της ανάκλασης του φωτός, η μέτρηση έχει μεγάλο σφάλμα σε περιβάλλοντα με γυάλινα εμπόδια ή εμπόδια από καθρέφτη.

Ελλείψει προηγούμενου χάρτη περιβάλλοντος, το LiDAR SLAM χρησιμοποιεί αισθητήρες για να μετρήσει το περιβάλλον και να λάβει ένα νέφος σημείων LiDAR, το οποίο στη συνέχεια χρησιμοποιείται για την εκτίμηση της θέσης του και την κατασκευή ενός χάρτη με βάση τις πληροφορίες θέσης.

Οι τυπικές μέθοδοι SLAM 2D LiDAR περιλαμβάνουν τις μεθόδους Gmapping [54], Hector SLAM [55], Karto SLAM [56] και Cartographer [57]. Το πλαίσιο του αλγορίθμου Gmapping βασίζεται στον αλγόριθμο RBPF (Rao-Blackwellized Particle Filters) και ο Gmapping πραγματοποιεί δύο κύριες βελτιώσεις, συμπεριλαμβανομένης της βελτίωσης της κατανομής των προτάσεων και της επιλεκτικής επαναδειγματοληψίας. Το πλαίσιο του αλγορίθμου Hector SLAM βασίζεται στον αλγόριθμο EKF (εκτεταμένο φίλτρο Kalman). Ο Hector SLAM μπορεί να χρησιμοποιηθεί χωρίς οδόμετρο και είναι κατάλληλος για περιβάλλοντα αερομεταφερόμενων ή ανώμαλων δρόμων, αλλά είναι επιρρεπής σε σφάλματα ολίσθησης όταν το LiDAR περιστρέφεται πολύ γρήγορα. Το Karto SLAM είναι ο πρώτος αλγόριθμος ανοικτού κώδικα που βασίζεται στον αλγόριθμο PGO (pose graph optimization). Βασίζεται στην αποτελεσματική κατασκευή γραμμικών πινάκων και στην αραιή μη επαναληπτική αποσύνθεση Cholesky για την αποτελεσματική αναπαράσταση και επίλυση μεγάλων αραιών γράφων κατεύθυνσης(pose). Το πιο σημαντικό σημείο του Karto SLAM είναι η εισαγωγή της βελτιστοποίησης back-end και της ανίχνευσης κλεισίματος βρόχων. Σε σύγκριση με το Gmapping και το Hector SLAM, το Karto SLAM πλεονεκτεί για χαρτογράφηση σε μεγάλα περιβάλλοντα. Το Cartographer βασίζεται στον αλγόριθμο PGO (pose graph optimization). Εξαλείφει τα σωρευτικά σφάλματα που δημιουργούνται κατά τη διαδικασία κατασκευής του χάρτη κυρίως μέσω της ανίχνευσης κλεισίματος βρόχων, στην οποία ο υποχάρτης αποτελεί τη βασική μονάδα. Επιταχύνει την ανίχνευση κλεισίματος βρόχων συνδυάζοντας τοπικά και παγκόσμια δεδομένα και έχει υψηλή απόδοση σε πραγματικό χρόνο, γεγονός που την καθιστά αξιόπιστη μέθοδο 2D LIDAR SLAM. Για την ανίχνευση εμποδίων, σε σύγκριση με τους αισθητήρες υπερήχων, το LiDAR έχει σημαντικά πλεονεκτήματα όσον αφορά την ακρίβεια μέτρησης της απόστασης και την αποτελεσματικότητα για τα εμπόδια.

Τεχνολογίες ραδιοσυχνοτήτων

Η τεχνολογία ανίχνευσης με βάση τη ραδιοσυχνότητα (RF) χρησιμοποιείται κυρίως για τον εντοπισμό κινητών ρομπότ σε εσωτερικούς χώρους. Το σύστημα εντοπισμού με βάση τα RF περιλαμβάνει σταθμούς φάρου (BS) που εκπέμπουν ραδιοσήματα και κινητά ρομπότ που τα λαμβάνουν. Σε αντίθεση με τα

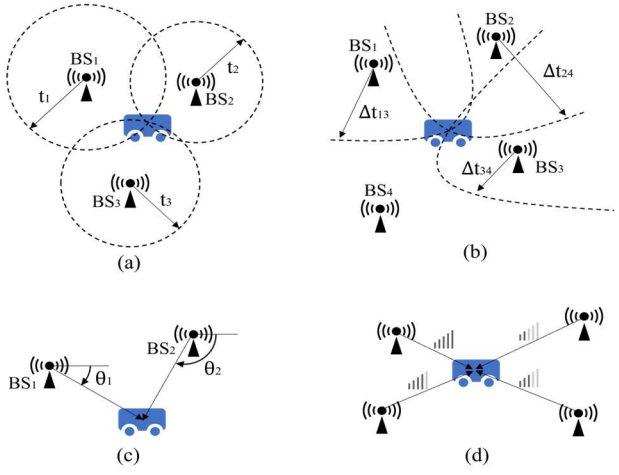
συστήματα εντοπισμού ραδιοφάρου που βασίζονται στις υπέρυθρες και τους υπερήχους, τα κύματα RF μπορούν να διαπεράσουν πόρτες και τοίχους, παρέχοντας πανταχού παρούσα κάλυψη των κτιρίων. Τα συστήματα εντοπισμού με βάση RF που εφαρμόζονται σε κινητά ρομπότ εσωτερικών χώρων συνήθως περιλαμβάνουν: [58], Bluetooth [59], Zigbee [60], Ultra-Wideband (UWB) [61] και Radio Frequency Identification (RFID). Τα πλεονεκτήματα και τα μειονεκτήματα των τεχνικών RF [62] παρουσιάζονται στον πίνακα 2.

RF	Range (m)	Cost	Advantages	Disadvantages
WiFi	250	high	Implementation simplicity, large coverage, high transmission rate	High power consumption, meterlevel accuracy, vulnerable to NLOS path
Zigbee	100	medium	Extremely low energy consumption, low system cost	Low transmission rate, high latency, vulnerable to NLOS path
Bluetooth	100	low	Implementation simplicity, low energy consumption	Small coverage, low transmission rate, vulnerable to NLOS path
UWB	100	high	High accuracy, extremely high transmission rate, low latency, immune to interference, robustness to NLOS path	High energy consumption and system cost
RFID	5	low	Extremely low energy consumption, implementation simplicity, low system cost, high accuracy with specific approach	Small coverage, vulnerable to NLOS path

Πίνακας 2. Τα χαρακτηριστικά των διαφόρων τεχνολογιών RF που χρησιμοποιούνται για τον εντοπισμό σε εσωτερικούς χώρους.

Αλγόριθμοι εντοπισμού τεχνολογίας ραδιοσυχνοτήτων

Οι τεχνολογίες εντοπισμού εσωτερικών χώρων με βάση το RF χρησιμοποιούν δύο κύριες μεθόδους: γεωμετρία και δακτυλικό αποτύπωμα. Η μέθοδος εντοπισμού με βάση τη γεωμετρία προσδιορίζει τη θέση του κινητού ρομπότ χρησιμοποιώντας γεωμετρική γνώση, είτε μετρώντας την απόσταση από σταθερούς σταθμούς φάρου (BS) είτε υπολογίζοντας τη γωνία του λαμβανόμενου σήματος. Αρκετοί τυπικοί αλγόριθμοι εντοπισμού θέσης με βάση το RF είναι ο χρόνος άφιξης (TOA), η χρονική διαφορά άφιξης (TDOA), η γωνία άφιξης (AOA) και η ένδειξη ισχύος λαμβανόμενου σήματος (RSSI), όπως φαίνεται στο Σχήμα 6.



Σχήμα 6. Τεχνολογίες εντοπισμού RF: (α) Χρόνος άφιξης (TOA). (β) Διαφορά χρόνου άφιξης (TDOA). (γ) Γωνία άφιξης (AOA). (δ) Ένδειξη ισχύος λαμβανόμενου σήματος (RSSI).

Η **TOA** είναι μια μέθοδος εντοπισμού που μετρά το χρόνο διάδοσης του σήματος. Ένα σήμα που αποστέλλεται από έναν πομπό σε ένα κινητό ρομπότ ταξιδεύει στον αέρα με την ταχύτητα του φωτός περίπου. Όταν το σήμα φτάσει σε δέκτες τοποθετημένους σε πολλαπλούς σταθμούς φάρου (BS), οι δέκτες καταγράφουν τους αντίστοιχους χρόνους άφιξης [63]. Η απόσταση διάδοσης του σήματος υπολογίζεται με τη μέτρηση του χρόνου διάδοσης. Αυτό επιτρέπει τον εντοπισμό του κινητού ρομπότ με τη μέθοδο του τριγωνισμού, όπως φαίνεται στο Σχήμα 6α. Τα σήματα που εκπέμπονται από τους πομπούς χρονοσημαίνονται για τον υπολογισμό του χρόνου διάδοσης. Είναι ζωτικής σημασίας ο συγχρονισμός της ώρας των σταθμών φάρου με την ώρα του κινητού ρομπότ [64].

Η **TDOA** είναι μια μέθοδος που χρησιμοποιείται για τον προσδιορισμό της θέσης ενός πομπού σήματος με τη μέτρηση της χρονικής διαφοράς μεταξύ της άφιξης του σήματος σε διαφορετικούς δέκτες [65]. Ο χρόνος άφιξης του ίδιου σήματος καταγράφεται από πολλαπλούς δέκτες σήματος και υπολογίζονται οι χρονικές διαφορές μεταξύ των διαφορετικών δεκτών. Αυτές οι χρονικές διαφορές χρησιμοποιούνται στη συνέχεια για την κατασκευή ενός συστήματος εξισώσεων. Κάθε εξίσωση αντιπροσωπεύει τον περιορισμό θέσης του αντίστοιχου δέκτη στον πομπό. Αυτοί οι περιορισμοί αντικατοπτρίζονται στο γράφημα συντεταγμένων ως υπερβολές. Η θέση του πομπού σήματος, δηλαδή του κινητού ρομπότ, προσδιορίζεται με την επίλυση αυτών των εξισώσεων, όπως φαίνεται στο Σχήμα 6β. Σε αντίθεση με τη μέθοδο TOA, η μέθοδος TDOA απαιτεί χρονικό συγχρονισμό μόνο μεταξύ των θέσεων φάρου, όχι μεταξύ αυτών και του κινητού ρομπότ [66].

Η **AOA** είναι μια μέθοδος που χρησιμοποιείται για τον προσδιορισμό της θέσης ενός πομπού σήματος με τη μέτρηση της γωνίας με την οποία το σήμα φτάνει σε διαφορετικούς δέκτες [67]. Ο δέκτης σήματος αποτελείται από πολλαπλά στοιχεία λήψης σήματος και η γωνία άφιξης υπολογίζεται συγκρίνοντας τους χρόνους άφιξης ή τις διαφορές φάσης των σημάτων μεταξύ των διαφορετικών στοιχείων λήψης. Η γωνία άφιξης ενός σήματος καταγράφεται από ένα πλήθος δεκτών σήματος. Στη συνέχεια, χρησιμοποιούνται γεωμετρικές σχέσεις για τον υπολογισμό της θέσης του πομπού σήματος, που είναι το κινητό ρομπότ, όπως φαίνεται στο σχήμα 6γ.

Ο δείκτης ισχύος λαμβανόμενου σήματος (RSSI) είναι μια μέτρηση της ισχύος ενός ραδιοσήματος που έχει ληφθεί. Η ισχύς του σήματος μειώνεται κατά τη διάδοση, η οποία χαρακτηρίζεται από το μοντέλο απώλειας ισχύος διάδοσης. Ως εκ τούτου, η ισχύς του λαμβανόμενου σήματος μπορεί επίσης να χρησιμοποιηθεί για τον υπολογισμό της απόστασης διάδοσης του σήματος. Ο τριγωνισμός χρησιμοποιείται στη συνέχεια για τη μέτρηση της θέσης του κινητού ρομπότ [68].

Ο **εντοπισμός με δακτυλικό αποτύπωμα** είναι μια τεχνική που χρησιμοποιείται για τον προσδιορισμό της θέσης με τη σύγκριση των σημάτων που λαμβάνονται με μια προκαθορισμένη βάση δεδομένων δακτυλικών αποτυπωμάτων [69]. Αποτελείται από την κατασκευή βάσης δεδομένων δακτυλικών αποτυπωμάτων και την εκτίμηση θέσης. Η διαδικασία περιλαμβάνει την ανάπτυξη ενός σταθμών λήψης αισθητήρων RF σε όλη την περιοχή για τη συλλογή σημάτων RF σε κάθε σημείο δειγματοληψίας και την καταγραφή των χαρακτηριστικών παραμέτρων τους. Οι θέσεις των σημείων δειγματοληψίας συσχετίζονται με αυτές τις παραμέτρους χαρακτηρισμού για την κατασκευή μιας βάσης δεδομένων δακτυλικών αποτυπωμάτων. Αυτή η βάση δεδομένων περιέχει προμετρημένες και καταγεγραμμένες αντιστοιχίες μεταξύ χαρακτηριστικών σημάτων και θέσεων. Η πιο συχνά χρησιμοποιούμενη χαρακτηριστική παράμετρος σήματος είναι το RSSI. Μόλις δημιουργηθεί η βάση δεδομένων δακτυλικών αποτυπωμάτων, οι σταθμοί αισθητήρων RF συλλέγουν σε πραγματικό χρόνο παραμέτρους ασύρματου σήματος, οι οποίες στη συνέχεια αντιστοιχίζονται και συγκρίνονται με τις πληροφορίες δακτυλικών αποτυπωμάτων στη βάση δεδομένων [70]. Η ακριβής θέση του στόχου εκτιμάται μέσω της χρήσης αλγορίθμων σύγκρισης και συμπερασμού. Σε σύγκριση με τη μέθοδο που βασίζεται στη γεωμετρία, η μέθοδος εντοπισμού δακτυλικών αποτυπωμάτων είναι πιο ακριβής. Ωστόσο, η κατασκευή της βάσης

δεδομένων δακτυλικών αποτυπωμάτων απαιτεί σημαντική προσπάθεια και πρέπει να συλλέγεται και να κατασκευάζεται εκ νέου εάν αλλάξει η θέση του σταθμού λήψης σήματος [66].

WiFi

Το WiFi είναι μια ευρέως χρησιμοποιούμενη ασύρματη τεχνολογία που ακολουθεί τα πρότυπα IEEE 802.11. Το WiFi χρησιμοποιεί συχνότητες ασύρματης επικοινωνίας 2,4 GHz και 5 GHz. Μπορεί να υποστηρίξει πολλαπλές συνδέσεις συσκευών ταυτόχρονα και χρησιμοποιείται ευρέως σε εσωτερικούς χώρους, όπως εργοστάσια και γραφεία. Ενώ η τεχνολογία WiFi προσφέρει πλεονεκτήματα όπως ευκολία, ευελιξία και υψηλή , έχει περιορισμούς όταν χρησιμοποιείται για εντοπισμό σε εσωτερικούς χώρους. Παράγοντες όπως η αστάθεια του σήματος WiFi και η διαδρομή μη οπτικής επαφής (NLOS) μπορεί να επηρεάσουν την ακρίβεια του εντοπισμού.

Για τις μεθόδους εντοπισμού θέσης με βάση το WiFi, χρησιμοποιείται συνήθως το RSSI λόγω της υψηλής ακρίβειάς του.

Zigbee

Το Zigbee είναι μια τεχνολογία ασύρματης επικοινωνίας που βασίζεται στο πρότυπο IEEE 802.15.4. Χρησιμοποιεί μια κατανεμημένη τοπολογία δικτύου που υποστηρίζει πολλαπλές συσκευές που επικοινωνούν τους, με όλους τους τερματικούς σταθμούς να επικοινωνούν απευθείας με τον συντονιστή. Το ZigBee έχει χαμηλό κόστος, χαμηλή κατανάλωση ενέργειας και χαμηλό ρυθμό μεταφοράς δεδομένων σε σύγκριση με τα πρότυπα WiFi, καθώς και μεγαλύτερο χρόνο καθυστέρησης.

Bluetooth

Το Bluetooth είναι μια τυποποιημένη τεχνολογία ασύρματης επικοινωνίας που βασίζεται στην IEEE

Πρότυπο 802.15. Χρησιμοποιείται συνήθως για τη μετάδοση δεδομένων μικρής εμβέλειας λόγω της χαμηλής κατανάλωσης ενέργειας, του χαμηλού κόστους, της υψηλής ασφάλειας και της ευκολίας χρήσης. Η τεχνολογία Bluetooth χρησιμοποιείται ευρέως σε διάφορα ηλεκτρονικά προϊόντα, αυτοκίνητα και ιατρικές συσκευές. Ωστόσο, η ακρίβεια του εντοπισμού θέσης σε εσωτερικούς χώρους επηρεάζεται από τη διαδρομή μη οπτικής επαφής (NLOS) λόγω εμποδίων που επηρεάζουν τη διάδοση του σήματος Bluetooth.

Υπερ-ευρυζωνικότητα (UWB)

Η τεχνολογία UWB (Ultra-Wideband) είναι μια τεχνολογία ασύρματης επικοινωνίας που χρησιμοποιεί σύντομα, μη συνεχή παλμικά σήματα για τη μετάδοση δεδομένων. Διαθέτει μεγάλο εύρος ζώνης συχνοτήτων και προσφέρει πλεονεκτήματα όπως χαμηλή κατανάλωση ενέργειας, γρήγορο ρυθμό μετάδοσης και ισχυρή ικανότητα προστασίας από παρεμβολές. Επιπλέον, το UWB επηρεάζεται θεωρητικά λιγότερο από διαδρομές μη οπτικής επαφής (NLOS), καθιστώντας το εξαιρετικά ακριβές και σταθερό για τον εντοπισμό σε εσωτερικούς χώρους.

Αναγνώριση ραδιοσυχνοτήτων (RFID)

Η τεχνολογία RFID (Radio Frequency Identification) χρησιμοποιεί ραδιοκύματα για τη μετάδοση δεδομένων και την αναγνώριση. Συνήθως αποτελείται από τρία στοιχεία: μια ετικέτα, έναν αναγνώστη και ένα σύστημα επεξεργασίας δεδομένων. Οι ετικέτες RFID είναι μικρές συσκευές που περιέχουν ένα τσιπ και μια κεραία χρησιμοποιούνται για την αποθήκευση και τη μετάδοση δεδομένων. Ο αναγνώστης επικοινωνεί με την ετικέτα μέσω ραδιοφώνου και διαβάζει ή γράφει τις πληροφορίες της ετικέτας. Τα δεδομένα RFID επεξεργάζονται και διαχειρίζονται από το σύστημα επεξεργασίας δεδομένων. Η

τεχνολογία RFID προσφέρει τα πλεονεκτήματα της χαμηλής κατανάλωσης ενέργειας και του χαμηλού κόστους του συστήματος. Ωστόσο, έχει μικρή απόσταση μετάδοσης σήματος, περίπου 1-5 m, γεγονός που απαιτεί την προρύθμιση πολυάριθμων σημείων φάρου για τον εντοπισμό μεγάλης περιοχής. τεχνολογία RFID πλεονεκτεί για τον εντοπισμό και την ταυτοποίηση στόχων μικρής εμβέλειας.

Περίληψη και συζήτηση

Οι αισθητήρες IMU έχουν υψηλή συχνότητα εξόδου και υιοθετούνται ευρέως στους τομείς εντοπισμού κινητών ρομπότ και SLAM. Ως ιδιοδεκτικός αισθητήρας, δεν επηρεάζεται από το περιβάλλον, αλλά δεν έχει ρόλο στην ανίχνευση εμποδίων. Για τον μετριασμό της συσσώρευσης σφαλμάτων, ένα IMU χρησιμοποιείται συχνά σε συνδυασμό με κωδικοποιητές ως βραχυπρόθεσμο οδόμετρο και αποτελεί μέρος της μεθόδου σύντηξης αισθητήρων στον εντοπισμό θέσης και το SLAM κινητών ρομπότ. Οι αισθητήρες υπερήχων είναι λιγότερο ακριβείς στην ανίχνευση απόστασης από ό,τι οι αισθητήρες υπέρυθρων και το LiDAR. Ως εκ τούτουσπάνια χρησιμοποιούνται ως πρωταρχικοί αισθητήρες στον εντοπισμό κινητών ρομπότ και στο SLAM σε πρόσφατες μελέτες. Ωστόσο, έχουν σημαντικά πλεονεκτήματα έναντι άλλων αισθητήρων στην ανίχνευση εμποδίων και χρησιμοποιούνται ευρέως στην αποφυγή εμποδίων από κινητά ρομπότ. Οι αισθητήρες υπέρυθρων, ως οπτικοί αισθητήρες, είναι ευαίσθητοι σε άλλες πηγές φωτός ή ανακλάσεις στο περιβάλλον. Ωστόσο, οι υπέρυθροι αισθητήρες είναι πολύ χαμηλού κόστους και χρησιμοποιούνται συχνά ως βοηθητικοί αισθητήρες σε κινητά ρομπότ, υπεύθυνοι για την ανίχνευση εμποδίων ή απόστασης στόχου. Οι αισθητήρες LiDAR και οι αισθητήρες όρασης είναι οι πιο συχνά χρησιμοποιούμενοι αισθητήρες στο SLAM για κινητά ρομπότ και μπορούν και οι δύο να χρησιμοποιηθούν για την πλοήγηση σε άγνωστα περιβάλλοντα. Αντιθέτως, το LiDAR έχει μεγαλύτερη ακρίβεια μέτρησης της απόστασης και οι αισθητήρες όρασης μπορούν να συλλέξουν πιο πλούσιες πληροφορίες για τις σκηνές εργασίας. Ωστόσο, η ακρίβεια των μετρήσεων LiDAR μπορεί να επηρεαστεί από διαφανή ή ανακλαστικά αντικείμενα στο περιβάλλον. Οι αισθητήρες όρασης πλεονεκτούν για την ανίχνευση εμποδίων ή στόχων της εξαιρετικής τεχνολογίας αναγνώρισης αντικειμένων που διαθέτουν. Μπορούν να φιλτράρουν αποτελεσματικά τα δεδομένα των εμποδίων, βελτιώνοντας την ακρίβεια εντοπισμού κατά την πλοήγηση του ρομπότ. Οι αισθητήρες ραδιοσυχνότητας (RF) υιοθετούνται ευρέως στον εντοπισμό κινητών ρομπότ και η ακρίβεια εντοπισμού δεν θα επηρεαστεί από περιβαλλοντικούς παράγοντες, όπως οι συνθήκες χαμηλής ορατότητας. Ωστόσο, ο εντοπισμός βάσει αισθητήρων RF απαιτεί πρόσθετους κόμβους αγκύρωσης με προηγούμενες πληροφορίες θέσης και δεν μπορεί να παρέχει πληροφορίες προσανατολισμού για κινητά ρομπότ. Μια σύγκριση των αποτελεσμάτων των σχετικών αισθητήρων παρουσιάζεται στον Πίνακα 3.

Sensors	Advantages	Disadvantages
IMU	High output frequency, free from environmental interference.	Initial conditions required, error accumulation, accelerometer is susceptible to vibration.
Ultrasonic sensor	Low cost, not influenced by transparent or reflective objects	long-range intensity attenuation, vulnerable to ambient noise.
Infrared sensor	Low cost.	Susceptible to environmental interference (light source, reflection).
LiDAR	High distance measurement accuracy.	Restricted in environments surrounded by transparent or reflective objects.
Vision-based sensor	Rich collection of information	Large computational volume, vulnerable to light.
Radio Frequency Sensor	Environment and obstacles have less effect on positioning.	Not suitable for unknown or unpredictable environments, unable to measure orientation.

Τα χαρακτηριστικά των αισθητήρων.

Κεφάλαιο 3.2.2: Επισκόπηση τεχνολογιών ανίχνευσης με πολλαπλούς αισθητήρες

Διάφοροι αισθητήρες μπορούν να συλλέγουν διάφορες μορφές δεδομένων μέτρησης. Η συγχώνευση αισθητήρων επιτρέπει την πληρέστερη απόκτηση αισθητηριακών πληροφοριών, βελτιώνοντας έτσι την αποτελεσματικότητα του εντοπισμού κατά την ολοκλήρωση της εργασίας. Είναι σημαντικό να σημειωθεί ότι κάθε αισθητήρας έχει τις δικές του υπεροχές, περιορισμούς και πιθανά σενάρια. Σημειώνεται ότι η συγχώνευση πληροφοριών αισθητήρων είναι σε θέση να βελτιώσει την ακρίβεια εντοπισμού θέσης του κινητού ρομπότ, την αποτελεσματικότητα αναγνώρισης εμποδίων και άλλα χαρακτηριστικά, αξιοποιώντας τα δυνατά σημεία και αντισταθμίζοντας τις αδυναμίες. Η προσέγγιση αυτή μπορεί να ικανοποιήσει τις απαιτήσεις διαφορετικών εργασιακών περιβαλλόντων. Όσον αφορά τους πολλαπλούς αισθητήρες, οι αισθητήρες που συχνά αναμειγνύονται με τους οπτικούς αισθητήρες είναι το σόναρ [125,126], οι αποστασιόμετρα λέιζερ [127,128], η αναγνώριση ραδιοσυχνότητας (RFID) [128,129], οι αδρανειακοί αισθητήρες [130,131], το GPS [132,133]. Ωστόσο, στην περίπτωση των αισθητήρων GPS, βασίζονται σε δορυφορικά συστήματα για την επίτευξη παγκόσμιου εντοπισμού θέσης. Σε εσωτερικά περιβάλλοντα, τα αντικείμενα επηρεάζονται από τα κτίρια καθώς και από άλλα σήματα, προκαλώντας σημαντική μείωση της ακρίβειας εντοπισμού τους, γεγονός που σημαίνει ότι το GPS μπορεί να είναι ακατάλληλο για τον εντοπισμό θέσης κινητών ρομπότ σε εσωτερικά σκοτεινά περιβάλλοντα [134,135].

Κεφάλαιο 3.3 Συζητήσεις και μελλοντικές τάσεις

Το σύστημα αντίληψης ενός AMR προσφέρει πληροφορίες σχετικά με τη συσχέτιση του ρομπότ με το περιβάλλον, οι οποίες είναι απαραίτητες για την επιτυχή πλοήγηση. Για να διασφαλιστεί ότι το κινητό ρομπότ εσωτερικού χώρου ολοκληρώνει με επιτυχία το έργο πλοήγησης, είναι απαραίτητη η λογική επιλογή και βελτιστοποίηση των αισθητήρων και των αλγορίθμων που χρησιμοποιούνται στο σύστημα αντίληψης με βάση τις ειδικές ανάγκες του σεναρίου.

Οι μέθοδοι εξαγωγής συμπερασμάτων τροχιάς με βάση την IMU μπορούν να χρησιμοποιηθούν μόνο για μικρές χρονικές περιόδους και μπορούν να χρησιμοποιηθούν για την υποβοήθηση άλλων συστημάτων εντοπισμού. Για τον εντοπισμό κινητών ρομπότ, τα συστήματα εντοπισμού θέσης που βασίζονται σε φάρο (συμπεριλαμβανομένων αισθητήρων RF, αισθητήρων υπερήχων και αισθητήρων υπέρυθρων) μπορούν να παρέχουν σταθερές και ακριβείς πληροφορίες απόλυτης θέσης. Ωστόσο, απαιτούν τακτική εγκατάσταση και συντήρηση και η ευρεία ανάπτυξη μπορεί να οδηγήσει σε αυξημένο κόστος. Το LiDAR ή οι κάμερες όρασης μπορούν να χρησιμοποιηθούν για ακριβή και σταθερό εντοπισμό θέσης. Ωστόσο, ο εντοπισμός τους βασίζεται σε περιβαλλοντικές πληροφορίες που εξάγονται από τους αισθητήρες, γεγονός που τους καθιστά ευάλωτους σε αλλαγές στο περιβάλλον. Για σενάρια με συχνές αλλαγές, οι μέθοδοι εντοπισμού που βασίζονται σε φάρο είναι πιο κατάλληλες.

Για τη SLAM ενός κινητού ρομπότ, τόσο οι κάμερες LIDAR όσο και οι κάμερες όρασης έχουν τα δικά τους πλεονεκτήματα. Τα δεδομένα νέφους σημείων λέιζερ απαιτούν λιγότερους υπολογισμούς, είναι εύκολο να υπολογιστούν, είναι ακριβή στην αποστασιομέτρηση και είναι λιγότερο ευαίσθητα σε μεγάλες αλλαγές στο φωτισμό. Ωστόσο, οι περιβαλλοντικές πληροφορίες που συλλέγονται από το LiDAR δεν είναι επαρκείς και ο χάρτης που κατασκευάζεται με τη χρήση 2D LiDAR περιέχει μόνο πληροφορίες απόστασης για το περιβάλλον. Ο μηχανισμός περιστροφής της συσκευής είναι απλός, γεγονός που επηρεάζει αρνητικά τη σταθερότητα της εσωτερικής δομής της με την πάροδο του χρόνου. Το Vision SLAM παρέχει μια πιο λεπτομερή περιγραφή του περιβάλλοντος, συμπεριλαμβανομένων των

πληροφοριών υφής, η οποία βελτιώνει την ακρίβεια και επιτρέπει την αναγνώριση των διαφορών μεταξύ των περιβαλλόντων. Ενώ το laser SLAM έχει περιορισμούς, το vision SLAM, εξοπλισμένο με κάμερα, προσφέρει μεγαλύτερη σταθερότητα όσον αφορά την εσωτερική δομή. Τα κινητά ρομπότ με μία μόνο κάμερα ως αισθητήρα σε εσωτερικά περιβάλλοντα περιορίζονται από τις περιβαλλοντικές συνθήκες και στερούνται robustness.

Για την ανίχνευση εμποδίων, οι κάμερες όρασης είναι αποτελεσματικές για την αναγνώριση εμποδίων, καθώς παρέχουν πλούσιες περιβαλλοντικές πληροφορίες. Οι αισθητήρες υπερήχων και το LiDAR είναι ακριβείς στη μέτρηση της απόστασης και της θέσης των εμποδίων.

Μελλοντικές τάσεις

Η τεχνολογία της αντίληψης, επωφελούμενη από την επιστήμη των υπολογιστών, την τεχνητή νοημοσύνη και τις οπτικές προσεγγίσεις, θα αναπτυχθεί με ταχείς ρυθμούς και αξίζει να δοθεί έμφαση στις ακόλουθες πτυχές:

1: Σύντηξη πληροφοριών πολλαπλών πηγών. Η μελέτη και η ανάπτυξη των κινητών ρομπότ έχουν επεκτείνει τις πιθανές εφαρμογές τους, μαζί με τις αυξημένες απαιτήσεις για τη χρήση τους. Ωστόσο, ένας μόνο αισθητήρας μπορεί να δυσκολευτεί να ικανοποιήσει τις απαιτήσεις πλοήγησης σε πολύπλοκες σκηνές λόγω των εγγενών περιορισμών του. Στην κατεύθυνση αυτή, λοιπόν, η σύντηξη πολλαπλών αισθητήρων καθίσταται ένας κρίσιμος τομέας έρευνας και ανάπτυξης για τα AMR σε εσωτερικούς χώρους. Η σύντηξη πολλαπλών αισθητήρων μπορεί να βελτιώσει την ακρίβεια, την αποτελεσματικότητα και τη σταθερότητα του εντοπισμού, της κατασκευής χάρτη και της ανίχνευσης εμποδίων παρέχοντας πιο ολοκληρωμένες περιβαλλοντικές πληροφορίες στο σύστημα αντίληψης. Η σύντηξη πολλαπλών αισθητήρων παρέχει ακριβείς και ολοκληρωμένες πληροφορίες στο σύστημα λήψης αποφάσεων, επιτρέποντας στα κινητά ρομπότ να προσαρμόζονται καλύτερα σε πολύπλοκα περιβάλλοντα και να ολοκληρώνουν εργασίες πλοήγησης πιο αποτελεσματικά και σταθερά.

2: Ευέλικτες και αποτελεσματικές στρατηγικές βελτιστοποίησης. Όταν τα κινητά ρομπότ εργάζονται σε διάφορα πολύπλοκα και δυναμικά περιβάλλοντα, υπάρχει μεγάλη ποσότητα άγνωστου θορύβου μοντέλου και τραχύτητας στις πληροφορίες παρατήρησης που λαμβάνονται από τους αισθητήρες. Για να διασφαλιστεί η αποτελεσματικότητα, η σταθερότητα και η ακρίβεια ενός συστήματος AMR, θα πρέπει να διερευνηθούν κατάλληλες στρατηγικές βελτιστοποίησης με βάση το συγκεκριμένο περιβάλλον και τις δυναμικές αλλαγές. Επιπλέον, υπάρχουν πολλές σύνθετες μη γραμμικές λειτουργίες στη διαδικασία βελτιστοποίησης. Για τη βελτίωση της υπολογιστικής απόδοσης κατά την επεξεργασία δεδομένων μεγάλης κλίμακας ή πολυτροπικών δεδομένων, μπορούν να υιοθετηθούν μέθοδοι όπως ο παράλληλος υπολογισμός και ο κατανεμημένος υπολογισμός.

Κεφάλαιο 4:

Περιβάλλον εξομοίωσης και παράμετροι

Εγκατάσταση webots και εισαγωγή παραμέτρων

1. Download the "webots-R2023b_setup.exe" installation file from our website.
2. Double click on this file.
3. Follow the installation instructions.
4. (Optional) Follow the programming language setup instructions, if you plan on using specific languages such as Python or Java.
5. It is possible to install Webots silently from an administrator DOS console, by typing:
6. webots-R2023b_setup.exe /SUPPRESSMSGBOXES /VERYSILENT /NOCANCEL /NORESTART /ALLUSERS
7. Additional setup options are available and documented from:
8. webots-R2023b_setup.exe /?
9. Once installed, if you observe 3D rendering anomalies or if Webots crashes, it is strongly recommend to upgrade your graphics driver.
10. Windows SmartScreen
11. It may be possible that Windows Defender SmartScreen will display a warning when starting the Webots installer:

4.1 Τι είναι το Webots;

Το webots είναι ένα περιβάλλον εξομοίωσης ρομπότ. Προσφέρει μία λύση για τη γρήγορη ανάπτυξη πρωτοτύπων και επιτρέπει στο χρήστη να δημιουργεί τρισδιάστατα εικονικά περιβάλλοντα που περιλαμβάνουν ιδιότητες της Φυσικής όπως π.χ. μάζα, συντελεστές τριβής, αρθρώσεις κ.α. Ο χρήστης μπορεί να κάνει τόσο χρήση παθητικών αντικειμένων όσο και ενεργητικών όπως κινούμενα ρομπότ. Τα ρομπότ μπορούν να είναι τροχοφόρα, αρθρωτά ή ακόμη και ιπτάμενα. Επιπροσθέτως, μπορούν να κάνουν χρήση μιας πληθώρας αισθητήρων και μηχανισμών όπως αισθητήρες απόστασης, κινητήρες τροχών, ψηφιακές κάμερες, μοτέρ, αισθητήρες αφής, πομπούς, δέκτες και πολλών άλλων. Τέλος, ο χρήστης μπορεί να προγραμματίσει κάθε ρομπότ ξεχωριστά έτσι ώστε να υλοποιεί την επιθυμητή λογική. Η πλατφόρμα webots διαθέτει μια πληθώρα από έτοιμα εικονικά μοντέλα από ρομπότ έτοιμα προς χρήση όπως επίσης και ικανοποιητικό αριθμό από διεπαφές οι οποίες αντιστοιχούν σε πραγματικά ρομπότ όπως για παράδειγμα e-puck. Με αυτόν τον τρόπο, όταν επιτυχθεί η επιθυμητή συμπεριφορά στο εικονικό μοντέλο, ο χρήστης μπορεί να μεταφέρει το πρόγραμμά του στο πραγματικό αντίστοιχο του ρομπότ. Δίνεται επίσης και η δυνατότητα να προσθέσει κανείς τις δικές του διεπαφές.

4.2 Τι μπορώ να κάνω με το webots;

Το webots δίνει τη δυνατότητα να αναπτύξει κανείς όχι μόνο εκπαιδευτικές εργασίες αλλά και να κάνει έρευνα που σχετίζεται με τα κινούμενα ρομπότ. Παραδείγματα χρήσης στη βιομηχανία που έχουν γίνει με χρήση του webots στο παρελθόν είναι:

- Κατασκευή πρωτοτύπων για την αυτοκινητοβιομηχανία, την αεροναυτιλία, βιομηχανία παιχνιδιών κ.α.
- Έρευνα πάνω στα ρομποτικά τρένα και τους σιδηροδρόμους.
- Έρευνα με βάση σμήνος πρακτόρων ή συνεργατικά δίκτυα ρομπότ κ.α.
- Έρευνα πάνω στην μηχανική μάθηση, τα νευρωνικά δίκτυα και την τεχνητή νοημοσύνη.
- Ανάπτυξη μεθόδων διδασκαλίας της ρομποτικής και του προγραμματισμού.
- Διαγωνισμούς ρομπότ

4.3 Τι πρέπει να γνωρίζει κανείς για να κάνει χρήση του webots;

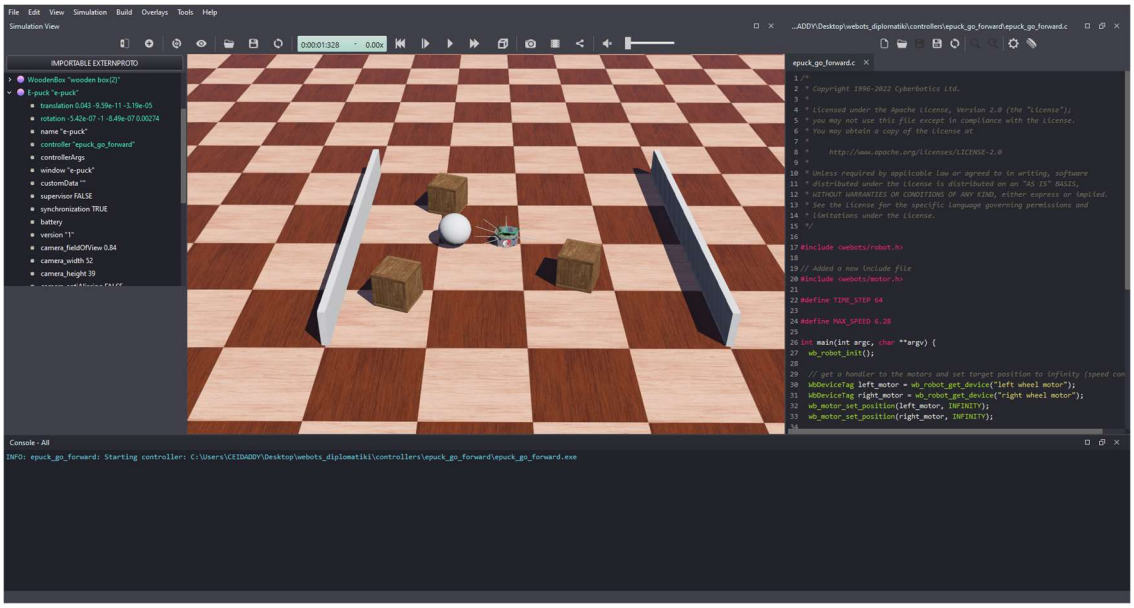
Για να κάνει κανείς χρήση της πλατφόρμας θα πρέπει να έχει μια βασική γνώση προγραμματιστικών εννοιών και εννοιών της φυσικής και της κίνησης. Προτεινόμενες και υποστηριζόμενες γλώσσες προγραμματισμού στο webots είναι οι C, C++, Java, Python, MATLAB. Δεν είναι απαραίτητη η γνώση σχεδιασμού τρισδιάστατων μοντέλων μιας και προσφέρονται έτοιμα μοντέλα και ρομπότ για να χρησιμοποιήσει κανείς. Ωστόσο, για να σχεδιάσει και να χρησιμοποιήσει κανείς δικά του τρισδιάστατα μοντέλα εντός του webots θα πρέπει να μπορεί να τα σχεδιάσει και να έχει γνώση της VRML97 γλώσσας προγραμματισμού.

4.4 Η διεπαφή χρήστη

Το γραφικό περιβάλλον Webots αποτελείται από 4 παράθυρα.

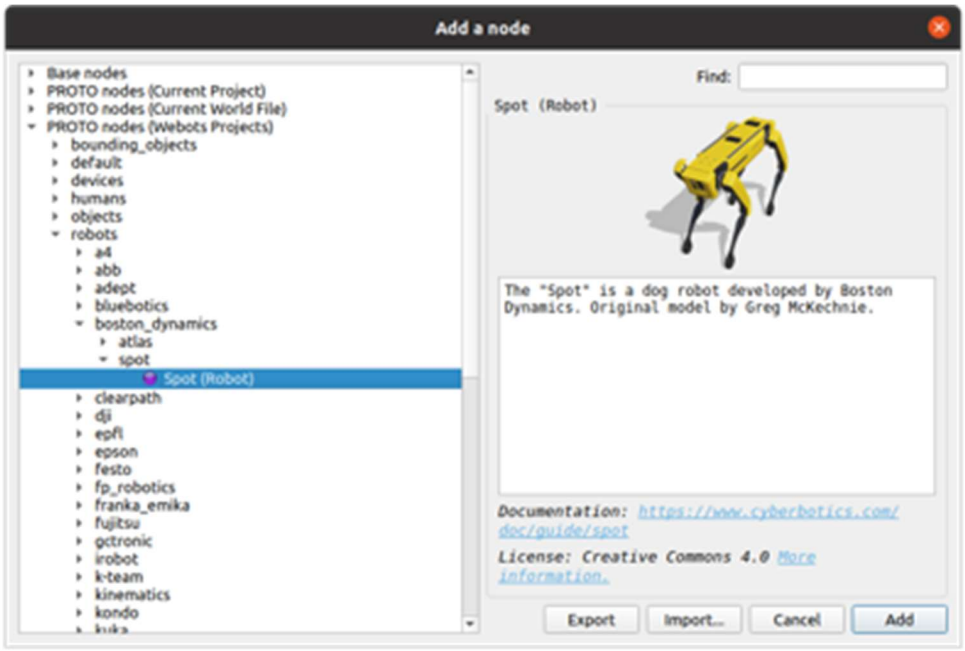
1. Το **παράθυρο 3-D**(κέντρο) το οποίο σχεδιάζει τα γραφικά πάνω στην οθόνη μας και το οποίο μας επιτρέπει να αλληλοεπιδρούμε διαδραστικά με το περιβάλλον και την προσομοίωση.
2. Το **δέντρο αντικειμένων**(αριστερά) το οποίο είναι μια ιεραρχική απεικόνιση του εικονικού κόσμου και στο οποίο βρίσκονται όλες οι παράμετροι του περιβάλλοντος. Αργότερα θα χρησιμοποιηθεί για να κάνω παραμετροποίηση του ρομπότ και των αισθητήρων που θα χρησιμοποιήσω.
3. Ο **επεξεργαστής κειμένου**(δεξιά) ο οποίος μου επιτρέπει να επεξεργαστώ κώδικα.

4. Η **κονσόλα**(κάτω) στην οποία μπορώ να παρατηρήσω την έξοδο του compiler και πιθανές εξόδους του ελεγκτή.



Εικόνα 4 1 Περιβάλλον webots - κονσόλα

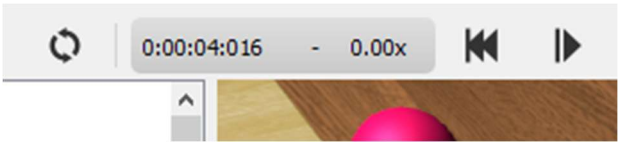
4.5 Προσθήκη αντικειμένου



Εικόνα 4 2 Περιβάλλον webots - Προσθήκη αντικειμένου

Το παράθυρο αυτό εμφανίζεται εάν πατήσω το κουμπί **Add** . Μου επιτρέπει να προσθέσω αντικείμενα από τη βιβλιοθήκη webots στον εικονικό μου κόσμο. Η λίστα με τα αντικείμενα που εμφανίζεται στο άνω παράθυρο αποτελείται μόνο από τα αντικείμενα που μπορώ να εισάγω στο συγκεκριμένο επίπεδο ιεραρχίας του «**δέντρου αντικειμένων**» το οποίο το έχω επιλέξει από πριν. Κάθε νέο αντικείμενο προστίθεται με τις προεπιλεγμένες τιμές του και τις οποίες μπορώ αργότερα να τροποιοποιήσω. Όλα τα αντικείμενα μιας προσομοίωσης μπορούν να εξαχθούν σαν τοπικά αρχεία .wbo ή .wrl και να εισαχθούν αργότερα από την ίδια ή κάποια άλλη προσομοίωση webots.

4.6 Ταχύμετρο και Εικονικός Χρόνος

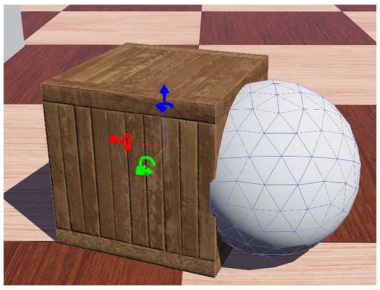


Εικόνα 4 3 Περιβάλλον webots - Ταχύμετρο και εικονικός χρόνος

Το ταχύμετρο δείχνει την ταχύτητα της προσομοίωσης στον υπολογιστή μου. Φαίνεται στη μπάρα εργαλείων πάνω από το **παράθυρο 3-D** και αντιπροσωπεύει την ταχύτητα μεταξύ προσομοίωσης και πραγματικού χρόνου. Με άλλα λόγια είναι η ταχύτητα του εικονικού κόσμου. Αν λ.χ. το ταχύμετρο δείχνει 2, αυτό σημαίνει πως η προσομοίωση του υπολογιστή μας τρέχει 2 φορές πιο γρήγορα από ότι θα έτρεχε ένα πραγματικό ρομπότ. Η ταχύτητα της προσομοίωσης δε φαίνεται όταν χρησιμοποιώ βήμα-προς-βήμα εκτέλεση κινήσεων στον εικονικό κόσμο. Το φορμάτ ώρας του ταχύμετρου είναι Ω:ΛΛ:ΔΔ:ΜΜΜ όπου Ω: ώρες, Λ: λεπτά, Δ: δευτερόλεπτα, Μ: milliseconds.

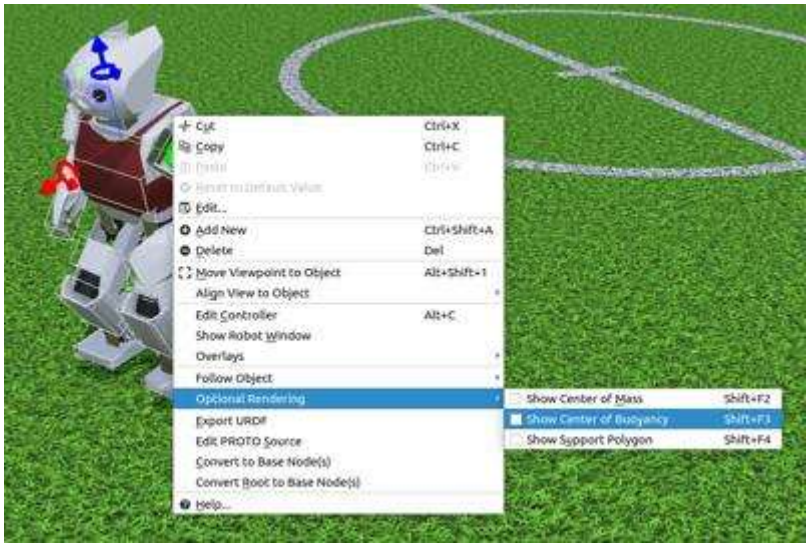
4.7 Το παράθυρο 3-D

Για να επιλέξω ένα στερεό αντικείμενο κάνω αριστερό κλικ. Το περίγραμμα του αντικειμένου θα εμφανιστεί με λευκές γραμμές. Οι γραμμές αυτές εμφανίζονται ροζ εάν το αντικείμενο βρίσκεται σε επαφή με κάποιο άλλο και μπλε όταν το αντικείμενο είναι αδρανές. Με διπλό κλικ μπορώ να ανοίξω το παράθυρο του ρομπότ εάν το αντικείμενο είναι ρομπότ.



Εικόνα 4 4 Περιβάλλον webots - Παράθυρο 3D

4.8 Μενού επιλογών

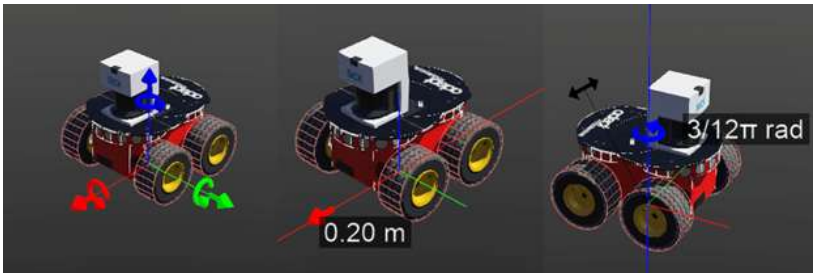


Εικόνα 4 5 Περιβάλλον webots - Μενού επιλογών

Μπορώ να κάνω δεξί κλικ σε ένα αντικείμενο εντός του **παράθυρου 3-D** για να το επιλέξω και να ανοίξω το μενού επιλογών. Το μενού αυτό μου επιτρέπει να χειραγωγήσω το ρομπότ ως προς όλες του τις παραμέτρους από τις αντίστοιχες επιλογές.

4.9 Χειρισμός αξόνων αντικειμένου

Κατά την επιλογή ενός αντικειμένου εμφανίζονται 3 χρωματιστά βέλη στο **παράθυρου 3-D** (βλ. Εικόνα 8). Τα βέλη αυτά μπορούν να χρησιμοποιηθούν για να μετακινήσω ή να περιστρέψω το αντικείμενο κατά τον εκάστοτε άξονα. Για να μετακινήσω ένα αντικείμενο αρκεί να κάνω κλικ στο αντίστοιχο βέλος και να το σύρω προς την επιθυμητή θέση. Μια ετικέτα θα εμφανιστεί δείχνοντάς μου τη σχετική μετακίνηση ή γωνία υπό την οποία υποβάλλουμε το αντικείμενο.



Εικόνα 4 6 Περιβάλλον webots - Χειρισμός αξόνων αντικειμένου

4.10 Εφαρμογή δύναμης σε στερεό

Για να ασκήσω δύναμη σε ένα στερεό αντικείμενο τοποθετώ τον δείκτη στο σημείο που θέλω να την ασκήσω, κρατάω πατημένο το **Alt** κάνοντας αριστερό κλικ ταυτόχρονα και σέρνω το δείκτη. Με αυτό τον τρόπο σχεδιάζω ένα τρισδιάστατο διάνυσμα το άκρο του οποίου βρίσκεται στο επίπεδο που διαπερνά παράλληλα το σημείο άσκησης της δύναμης. Η δύναμη υπολογίζεται ως:

$$\boldsymbol{F}\ \textbf{[N]} = \textbf{WorldInfo.dragForceScale} * \textbf{Solid.mass} * \boldsymbol{d^3}$$

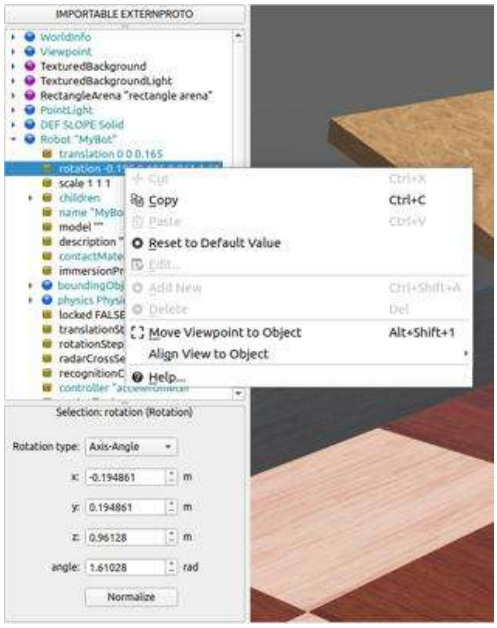
Όπου **d** είναι το διάνυσμα, σε μέτρα, που δημιούργησα.

4.11 Εφαρμογή ροπής σε στερεό

Για να ασκήσω ροπή σε ένα στερεό αντικείμενο τοποθετώ τον δείκτη στο σημείο που θέλω να την ασκήσω, κρατάω πατημένο το Alt κάνοντας δεξί κλικ ταυτόχρονα και σέρνω το δείκτη. Με αυτό τον τρόπο σχεδιάζω ένα τρισδιάστατο διάνυσμα με αρχή το κέντρο μάζας και το άκρο του οποίου βρίσκεται στο επίπεδο που διαπερνά παράλληλα το κέντρο αυτό. Το στερεό περιστρέφεται προς την κατεύθυνση του βέλους και η ροπή υπολογίζεται ακολούθως:

$$\boldsymbol{T}\ \textbf{[Nm]} = \textbf{WorldInfo.dragTorqueScale} * \textbf{Solid.mass} * \boldsymbol{d^3}$$

Όπου **d** είναι το διάνυσμα, σε μέτρα, που δημιουργήσαμε.



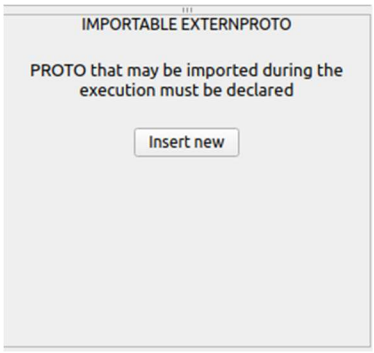
Εικόνα 4 7 Περιβάλλον webots - Εφαρμογή ροπής σε στερεό

4.12 Το δέντρο αντικειμένων

Εδώ (βλ. Εικόνα 9)περιέχονται όλες οι πληροφορίες και τα δεδομένα του εικονικού μου κόσμου συμπεριλαμβανομένου ρομπότ και στοιχείων περιβάλλοντος και η γραφική αναπαράστασή τους. Η δομή του είναι αυτή ενός αρχείου VRML97 αρχείου. Αποτελείται από μια λίστα αντικειμένων με τα αντίστοιχα πεδία τους να αποτελούνται είτε από κείμενο, αριθμητικές τιμές ή και άλλα αντικείμενα. Το δέντρο αντικειμένων διαθέτει επίσης λειτουργίες copy, paste, εναλλαγή μεταξύ προεπιλεγμένων τιμών επιμέρους πεδίων και εξερεύνηση ιδιοτήτων μέσω Documentation. Αυτό καθιστά την τροποποίηση του περιβάλλοντος μια ιδιαίτερα εύκολη διαδικασία αφού μου δίνονται όλες οι ελευθερίες τροποποίησης παραμέτρων των αντικειμένων και μεταφοράς μεταξύ τους σε επιμέρους στοιχεία.

4.13 Μενού εισαγωγής αντικειμένων

Το μενού αυτό μπορεί να ανοιχτεί κάνοντας κλικ στο κουμπί “Insert new” της Εικόνας 10 που βρίσκεται πάνω από το δέντρο αντικειμένων. Κάθε αρχείο περιβάλλοντος που χρησιμοποιώ κάνει χρήση τα αντίστοιχα PROTO αντικείμενα(assets) και πρέπει να περιλαμβάνει τα URL από τα οποία αντλεί τα αντικείμενα αυτά. Για να εισάγω κάποιο νέο αντικείμενο κάνω κλικ στο κουμπί και ακολουθώ τη διαδικασία επιλέγοντας το αρχείο που θέλω να εισάγω. Το webots αναλαμβάνει να ελέγξει το αρχείο αυτό και να το ενσωματώσει στο περιβάλλον με κάθε σώσιμο του αρχείου ή με κάθε νέα εκτέλεση κώδικα εντός του περιβάλλοντος.

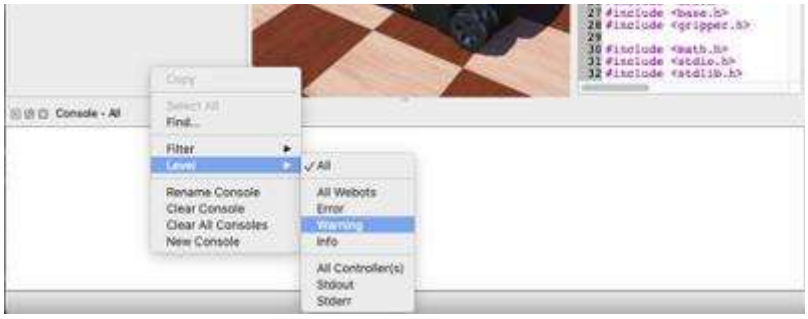


Εικόνα 4 8 Περιβάλλον webots – Μενού εισαγωγής αντικειμένων

4.14 Η κονσόλα

Η κονσόλα Webots (Εικόνα 11) τυπώνει όλα τα μηνύματα που προέρχονται από το Webots όπως ειδοποιήσεις για σφάλματα compilation, σφάλματα κατά το άνοιγμα κάποιου περιβάλλοντος και γενικά δρα σαν τη διεπαφή μεταξύ χρήστη και περιβάλλοντος. Για να διασφαλιστεί πως η έξοδος της κονσόλας είναι ντετερμινιστική, η έξοδος που προέρχεται από τους ελεγκτές δεν τυπώνεται αμέσως αλλά καθώς λαμβάνεται τότε ομαδοποιείται ανά ελεγκτή και τυπώνεται στο τέλος κάθε βήματος προσομοίωσης. Αυτό σημαίνει πως εάν η προσομοίωση περιέχει ένα ρομπότ Α και ένα ρομπότ Β, η έξοδος του ρομπότ Α θα

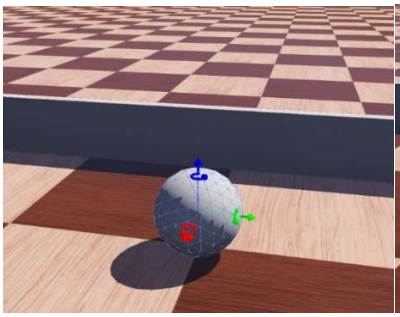
τυπωθεί στην κονσόλα πριν την έξοδο του ρομπότ Β ανεξαρτήτως εάν η έξοδος του ρομπότ Β παρελήφθη νωρίτερα από αυτή του Α.



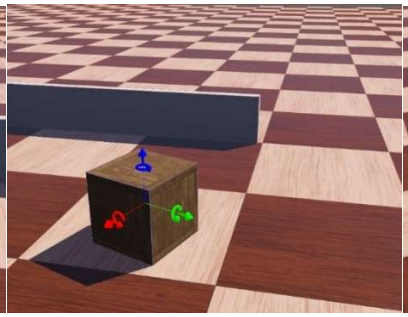
Εικόνα 4 9 Περιβάλλον webots – Κονσόλα

4.15 Περιγραφή επιμέρους στοιχείων του περιβάλλοντος.

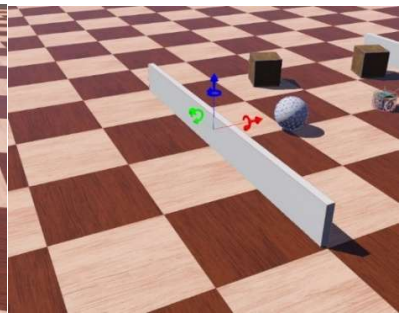
Τα εμπόδια που διαθέτει η πλατφόρμα είναι σχήματα όλων των ειδών όπως σφαίρες, κύβοι, παραλληλεπίπεδα, κύλινδροι κ.α. Η χρήση τοίχου θα είναι και η πιο εκτεταμένη καθώς είναι πιο εύκολη η παραμετροποίηση και τοποθέτησή του. Παράδειγμα χειρισμού αντικειμένων.



Εικόνα 4 10



Εικόνα 4 11



Εικόνα 4 12

Κεφάλαιο 5:

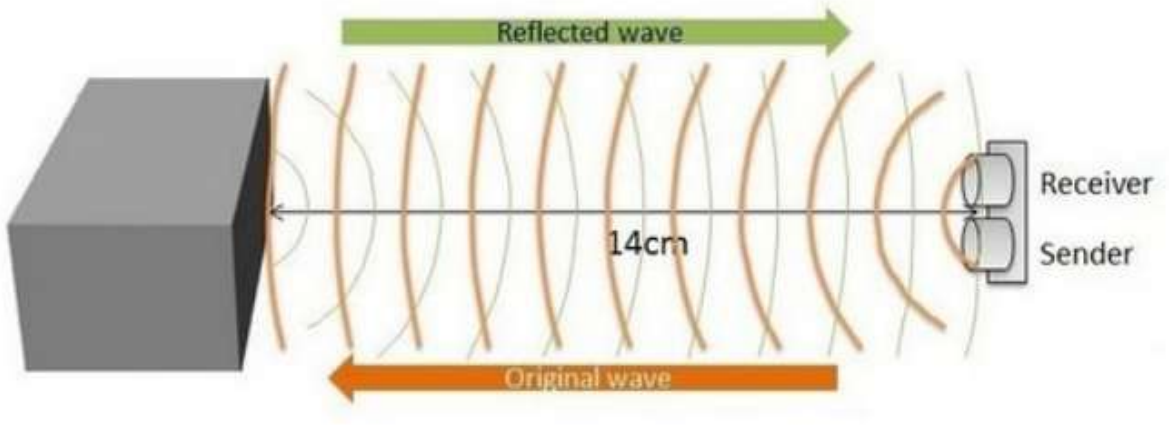
Εικονικό ρομπότ που χρησιμοποιήθηκε

5.1 Επισκόπηση των χαρακτηριστικών του e-puck



Εικόνα 5 1 Επισκόπηση e-puck

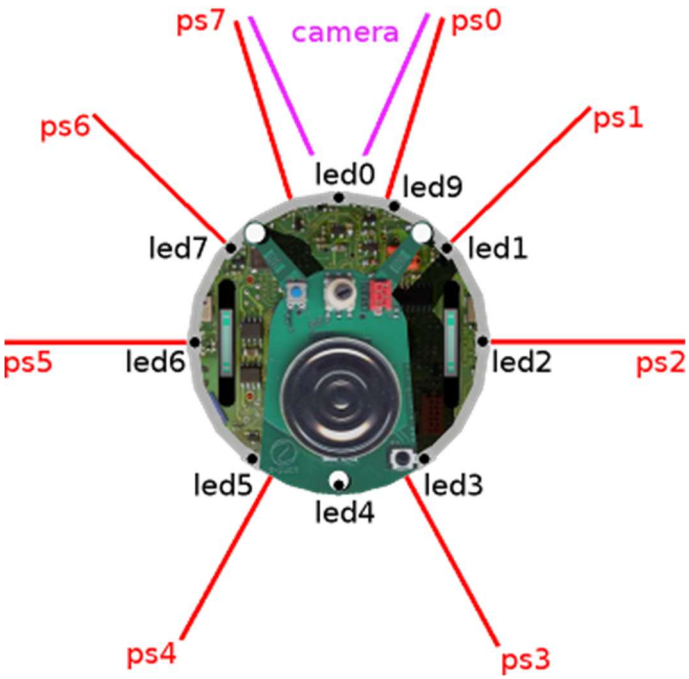
Το e-puck **Εικόνα 15** είναι ένα ανοιχτού λογισμικού τροχοφόρο ρομπότ το οποίο αναπτύχθηκε στο EPFL για διδακτικούς σκοπούς. Μας δίνει τη δυνατότητα να χειριστούμε όλα τα ηλεκτρονικά εξαρτήματα του σε χαμηλό επίπεδο καθώς επίσης και να το τροποποιήσουμε χωρίς περιορισμούς. Το εικονικό μοντέλο του περιλαμβάνει διαφορικά μοτέρ τροχών(προσομοιώνονται επίσης οι κωδικοποιητές τροχού σαν αισθητήρες θέσης), υπέρυθρους αισθητήρες για υπολογισμό εγγύτητας και μετρήσεις, επιταχυνσιόμετρο, γυροσκόπιο, ψηφιακή κάμερα, 8 LEDs, επικοινωνία Bluetooth και αισθητήρες εδάφους.



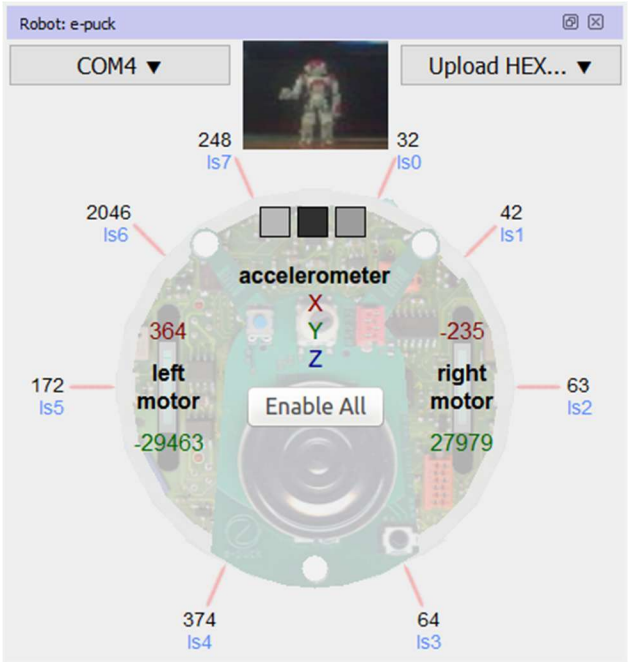
Εικόνα 5 2 Ανάκλαση υπερήχων σε εμπόδιο

Στον παρακάτω πίνακα φαίνονται τα χαρακτηριστικά του πραγματικού ρομπότ.

Feature	Description
Size & Weight	7.4 cm in diameter, 4.5 cm high & 150 g
Processor	Microchip dsPIC 30F6014A @ 60MHz (about 15 MIPS)
IR sensors	8 infra-red sensors measuring ambient light and proximity of obstacles in a 4 cm range
Camera	color camera with a max resolution of 640x480 (typical use: 52x39 or 640x1)
Microphones	3 omni-directional microphones for sound localization
Accelerometer	3D accelerometer along the X, Y and Z axes
Gyroscope	3D gyroscope along the X, Y and Z axes



Εικόνα 5 3 Κατεύθυνση αισθητήρων e-puck



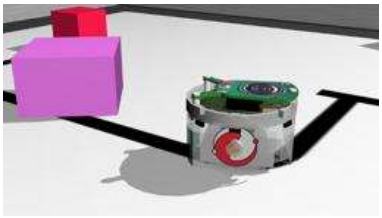
Εικόνα 5 4 Αισθητήρες e-puck με αναγνωριστικά

5.2 Δείγματα λειτουργίας

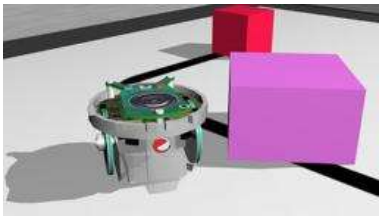
Το webots διαθέτει μια πληθώρα από έτοιμες υλοποιήσεις οι οποίες χρησιμοποιούν το ρομπότ e-puck. Μεταξύ άλλων υλοποιείται συμπεριφορά βασισμένη σε όχημα Braitenberg, Line Following και Line Following με εμπόδια.



Εικόνα 5 5 Σενάριο Κίνησης



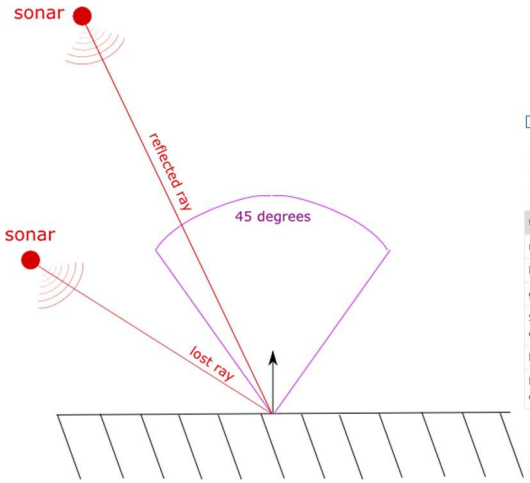
Εικόνα 5 6 Σενάριο Κίνησης 2



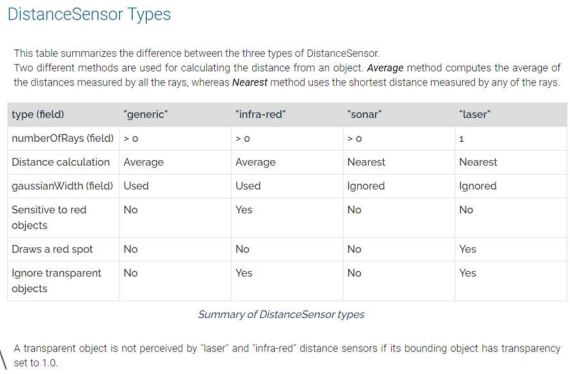
Εικόνα 5 7 Σενάριο Κίνησης 3

5.3 Παραμετροποίηση εικονικού ρομπότ

Για τα πειράματα που θα διεξάγω θα χρησιμοποιηθούν αισθητήρες υπερήχων. Οι αισθητήρες αυτοί στο εικονικό ρομπότ υποστηρίζονται τροποποιώντας μερικές τιμές στο PROTO file του περιβάλλοντος εξομοίωσης distance_sensor_numberOfRays 2. Για να ενεργοποιήσω τους αισθητήρες αρχικά ανοίγω το δέντρο αντικειμένων στο αριστερό μέρος της οθόνης και επιλέγουμε τους αισθητήρες από την αντίστοιχη δομή που αναφέρεται στο ρομπότ. Ύστερα επιλέγω εάν θα ενεργοποιήσουμε και ποιους αισθητήρες. Θα γίνει χρήση του μπροστινού αισθητήρα και του πλαϊνού έτσι ώστε να αποφύγω επικάλυψη στα σημεία που ανιχνεύω και να δώσω τη ρεαλιστική προσέγγιση της υλοποίησης. Στην πραγματικότητα η ανάκλαση ενός υπέρηχου μπορεί να διαβαστεί και από κάποιον άλλο αισθητήρα ξεχωριστά από αυτόν που εκπέμπει το σήμα αρχικά. Επιπροσθέτως υπάρχει εξτρα λογική που πρέπει να ενσωματωθεί για την ανίχνευση του ίδιου σημείου από 2 αισθητήρες καθώς η επεξεργασία τους από το φίλτρο που υλοποιείται θα αναθέτει 2 φορές πιθανότητα ύπαρξης ή όχι εμποδίου στο grid.



Εικόνα 5 8 Ανάκλαση υπερηχητικού σήματος sonar



DistanceSensor Types

This table summarizes the difference between the three types of DistanceSensor.
Two different methods are used for calculating the distance from an object. ***Average*** method computes the average of the distances measured by all the rays, whereas ***Nearest*** method uses the shortest distance measured by any of the rays.

type (field)	"generic"	"infra-red"	"sonar"	"laser"
numberOfRays (field)	> 0	> 0	> 0	1
Distance calculation	Average	Average	Nearest	Nearest
gaussianWidth (field)	Used	Used	Ignored	Ignored
Sensitive to red objects	No	Yes	No	No
Draws a red spot	No	No	No	Yes
Ignore transparent objects	No	Yes	No	Yes

Summary of DistanceSensor types

A transparent object is not perceived by "laser" and "infra-red" distance sensors if its bounding object has transparency set to 1.0.

Εικόνα 5 9 Περιβάλλον webots - Τύποι αισθητήρων

DistanceSensor Types

This table summarizes the difference between the three types of DistanceSensor.
Two different methods are used for calculating the distance from an object. ***Average*** method computes the average of the distances measured by all the rays, whereas ***Nearest*** method uses the shortest distance measured by any of the rays.

type (field)	"generic"	"infra-red"	"sonar"	"laser"
numberOfRays (field)	> 0	> 0	> 0	1
Distance calculation	Average	Average	Nearest	Nearest
gaussianWidth (field)	Used	Used	Ignored	Ignored
Sensitive to red objects	No	Yes	No	No
Draws a red spot	No	No	No	Yes
Ignore transparent objects	No	Yes	No	Yes

Summary of DistanceSensor types

A transparent object is not perceived by "laser" and "infra-red" distance sensors if its bounding object has transparency set to 1.0.

Εικόνα 5 10 Περιβάλλον webots - Στοιχεία αισθητήρα sonar

Κεφάλαιο 6:

Η προσέγγισή μου

Κεφάλαιο 6.1: Λειτουργία συστήματος

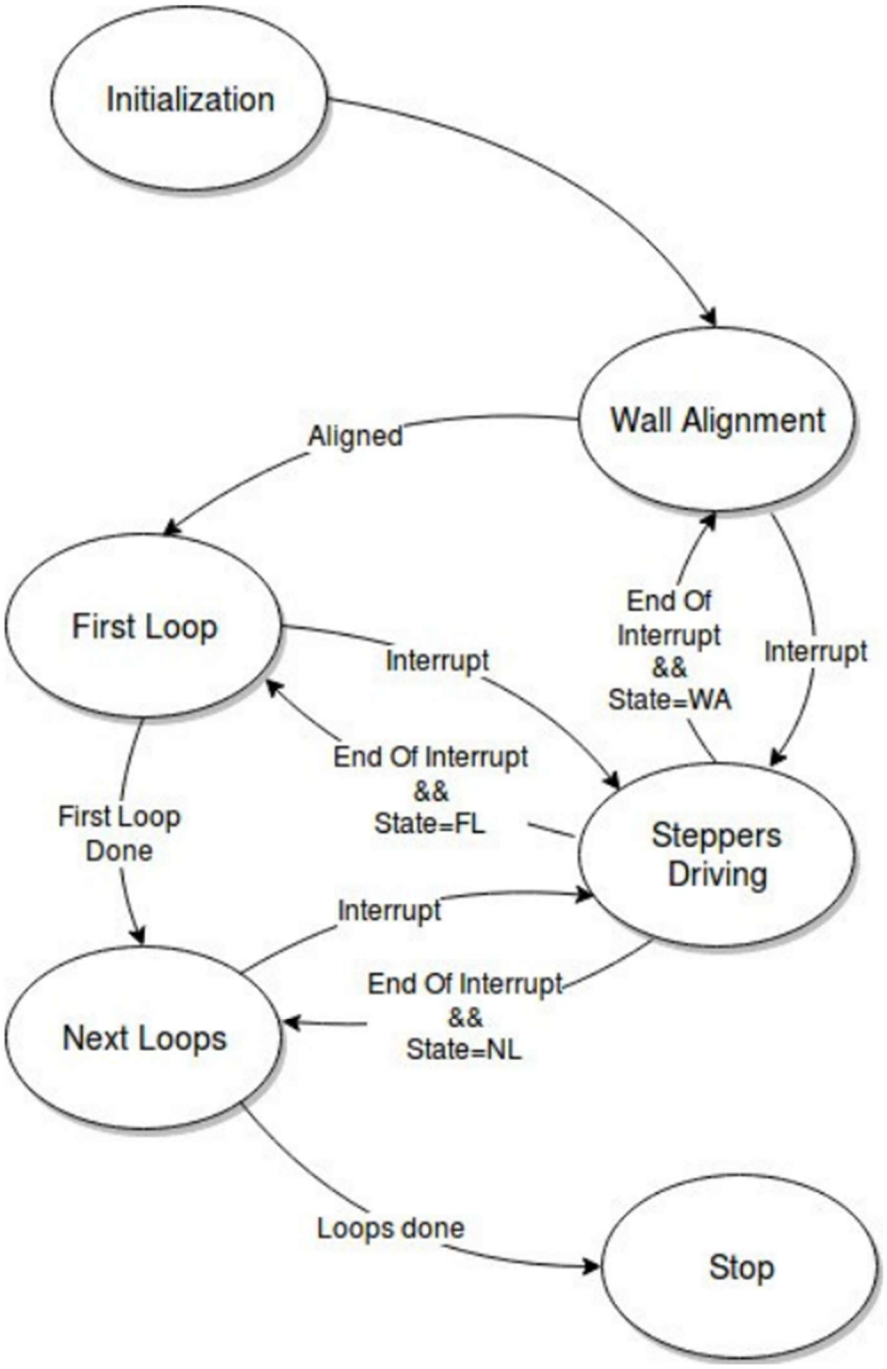
Οι προηγούμενες και πολλές άλλες προσεγγίσεις έχουν αποδείξει όλες την ικανότητά τους στην έρευνα και επίλυση SLAM προβλημάτων. Σε αυτό το σημείο μπορούμε να πούμε πως οι αλγόριθμοι SLAM χωρίζονται σε 3 κατηγορίες. Με βάση τους αισθητήρες που κάνουν χρήση, με βάση τη μέθοδο που χρησιμοποιούν ή με βάση τη δομή τους. Οι αλγόριθμοι που χαρακτηρίζονται από τους αισθητήρες τους, χρησιμοποιούν λέιζερ, υπέρηχους, αισθητήρες όρασης και οδομετρίας. Οι αλγόριθμοι που χαρακτηρίζονται από τις υπολογιστικές μεθόδους που χρησιμοποιούν μπορεί να κάνουν για παράδειγμα χρήση του φιλτρου Kalman ή του Particle Filter. Τέλος, οι αλγόριθμοι που χαρακτηρίζονται από τη δομή τους είναι είτε Online-SLAM ο οποίος προαποθηκεύει μόνο τα απαραίτητα σημεία αναφοράς, είτε fullSLAM ο οποίος αποθηκεύει κάθε σημείο αναφοράς μέσω εξερεύνησης.

Η επιλογή του αλγορίθμου προς χρήση είναι πάρα πολύ σημαντική και συνήθως το πρώτο βήμα για να επιλύσει κανείς το πρόβλημα SLAM. Η απόφαση αυτή καθορίζεται και από τις ικανότητες του ρομπότ βάσει των αισθητήρων που διαθέτει. Κάποιοι αισθητήρες όπως LIDAR για παράδειγμα μπορεί να είναι ακριβείς σε μεγάλο βαθμό στις μετρήσεις τους. Έχουν επίσης μεγαλύτερη εμβέλεια κάλυψης καθώς μπορούν με ένα πέρασμα να λάβουν πολύ μεγαλύτερο ποσοστό τους περιβάλλοντος σαν δεδομένα. Βέβαια, το κόστος τους είναι ιδιαίτερα υψηλό σε σχέση με χαμηλότερης ακρίβειας αισθητήρες τόσο για την αγορά τους όσο και στην κατανάλωση ενέργειας και στο βάρος τους. Από την άλλη οι αισθητήρες υπέρηχων έχουν εξαιρετικά χαμηλό κόστος, μπορούν να κάνουν μετρήσεις με ακρίβεια για τις περιορισμένες αποστάσεις στις οποίες λειτουργούν.

Στόχος μου ήταν να σχεδιάσω μια χαμηλού κόστους λύση για ένα αυτοματοποιημένο όχημα το οποίο θα καταφέρει με επιτυχία να προσεγγίσει την επίλυση ενός SLAM προβλήματος. Κατανοώντας τις ανάγκες της εργασίας αποφάσισα να προσομοιώσω ένα ρομπότ όπως το e-puck εντός του περιβάλλοντος webots διότι μου δίνει μια άμεση επαφή με την πραγματική κατάσταση του προβλήματος η οποία μοντελοποιείται με μεγάλη ακρίβεια στο συγκεκριμένο προσομοιωτή. Το αυτόνομο ρομπότ αυτό καταφέρνει να κινείται στο χώρο αποφεύγοντας όποιο εμπόδιο συναντά. Επιπροσθέτως, συλλέγει πληροφορία για το περιβάλλον του, τη στάση του, την απόσταση που ταξίδεψε από το προηγούμενο βήμα της προσομοίωσης του καθώς και απόσταση από τυχόν εμπόδια, εάν αυτά είναι παρόντα. Τέλος, το ρομπότ επεξεργάζεται την πληροφορία και επιστρέφει μέσω του περιβάλλοντος ένα νέφος με μετρήσεις το οποίο θα παρουσιάσουμε αργότερα. Σε αυτό το σημείο να σημειωθεί πως στον πραγματικό κόσμο η επεξεργασία μπορεί να γίνει εκτός του ρομπότ με το ίδιο να παίζει έναν ρόλο αναμεταδότη συλλέγοντας απλά πληροφορία και στέλνοντάς τη προς επεξεργασία σε κάποιον υπολογιστή.

Παρακάτω περιγράφεται η λειτουργία του FSM που τρέχει το ρομπότ καθώς επίσης δίνονται και οι βασικές ενέργειες που εκτελεί το ρομπότ επιγραμματικά.

Εξίσου σημαντική με τους αισθητήρες είναι και η σχεδίαση της λογικής και της αρχιτεκτονικής του συστήματος του ρομπότ για να επεξεργαστεί τα δεδομένα από τους αισθητήρες, να κινήσει το ρομπότ και να κατασκευάσει τον επιθυμητό χάρτη.



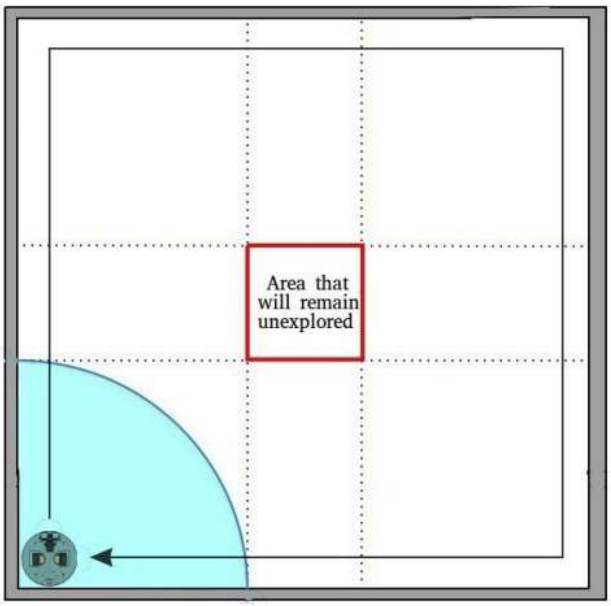
Εικόνα 6 1 Finite State Machine

Το πρόγραμμα που ελέγχει τις κινήσεις του ρομπότ αποτελείται από συναρτήσεις που κάθε μία ελέγχει και μια λειτουργία όπως οι send_measurements(), side_Sonar_Read() ή driveStraightDistance(). Οι ακριβείς λειτουργίες δίνονται στο παράρτημα για τις συναρτήσεις. Δεν είναι τίποτε άλλο παρά ένα fsm το οποίο υποκινεί τη λογική του ρομπότ και αποτελείται από 6 καταστάσεις. Αυτές οι καταστάσεις καλούν τις συναρτήσεις και τροποποιούν τις αντίστοιχες παραμέτρους έτσι ώστε να καθορίσουν την κατάλληλη κατάσταση. Το fsm αυτό λειτουργεί άρα σαν μια μηχανή Mealy. Στην πάνω εικόνα φαίνεται το fsm μαζί με τις προϋποθέσεις αλλαγής κατάστασης.

Το αρχικό σημείο είναι το Initialization State, το οποίο θα παραμείνει ενεργό μέχρι το ρομπότ να πάρει μετρήσεις από το σύστημα. Ύστερα μεταφέρεται στην κατάσταση Wall Allignment η οποία παραμένει ενεργή μέχρι να ευθυγραμμιστεί το ρομπότ με τον τοίχο στα αριστερά του. Η ενεργή κατάσταση τώρα είναι η First Loop. Μόλις το ρομπότ τελειώσει τη διαδρομή του και εξερευνήσει όλο το χώρο η κατάσταση γίνεται Next Loop. Το πρόγραμμα τελειώνει μόλις μεταβεί στην κατάσταση Stop.

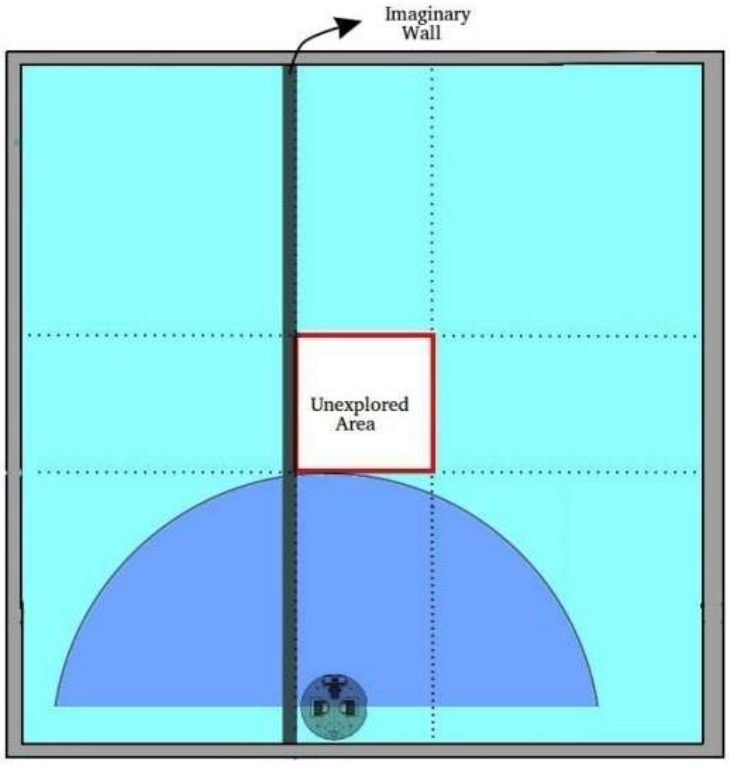
Ας δούμε την κατάσταση Steppers Driving. Στην εργασία μας σχεδιάσα τα stepper motors να είναι interrupt driven. Αυτό με ωφελεί ιδιαίτερα μιας και μπορώ έτσι να κάνω όποια κίνηση θέλω με το όχημα μου καθώς μπορώ να εκτελώ οποιαδήποτε άλλη εργασία ταυτόχρονα. Για αυτό το λόγο η κατάσταση Stepper Driving είναι προσβάσιμη από οποιαδήποτε άλλη κατάσταση εκτός βέβαια από την Initialization κάθε φορά που έχουμε Interrupt.

Τα κύρια μέρη του fsm είναι τα FirstLoop και NextLoops. Για να κατανοήσω καλύτερα τις συγκεκριμένες καταστάσεις εγκαινιάζω το σκεπτικό του Path planning. Αυτή η τεχνική είναι ένα σημαντικό προαπαιτούμενο για την κίνηση των αυτόνομων ρομπότ αφού με βοηθά να βρω τη βέλτιστη διαδρομή μεταξύ 2 σημείων. Το μονοπάτι που επέλεξα είναι να ακολουθήσει το ρομπότ τον τοίχο αριστερά, πράγμα το οποίο θα με βοηθήσει να καταλάβω καλύτερα την επιφάνεια του χώρου που εξετάζω. Κάνοντας χρήση των αισθητήρων μπορώ να παρατηρήσω και το περιβάλλον γύρω από το ρομπότ παρέχοντας επιπλέον χρήσιμη πληροφορία για τη χαρτογράφηση. Αυτή η διαδικασία λαμβάνει χώρα κατά την κατάσταση First Loop. Ωστόσο, λόγω του περιορισμού του αισθητήρα υπερήχων μπορεί να έχω και περιοχές που μένουν ανεξερεύνητες στην κατάσταση First Loop. Αυτό συμβαίνει στις περιπτώσεις όπου η απόσταση της περιφέρειας είναι μεγαλύτερη από 8m. Σε αυτή την περίπτωση χρειάζομαι έναν αριθμό επαναλήψεων ακόμη για να εξερευνήσω την επιπλέον περιοχή. Χρησιμοποιώ την εξίσωση **#Loops_needed=floor(track_length/2).** Η εξίσωση προκύπτει από περιορισμούς σε αντίστοιχες πραγματικές πλατφόρμες και είναι εμπειρική καθώς μπορεί ανά πλατφόρμα να μεταβάλλεται. Αίτιο είναι η ικανότητα αποθήκευσης περιορισμένων δεδομένων σε κάποιους μικροελεγκτές και στη συγκεκριμένη περίπτωση χρησιμοποιείται για ευκολία στη μετατροπή των μετρήσεων μου αργότερα σε πιο εύχρηστα δεδομένα.



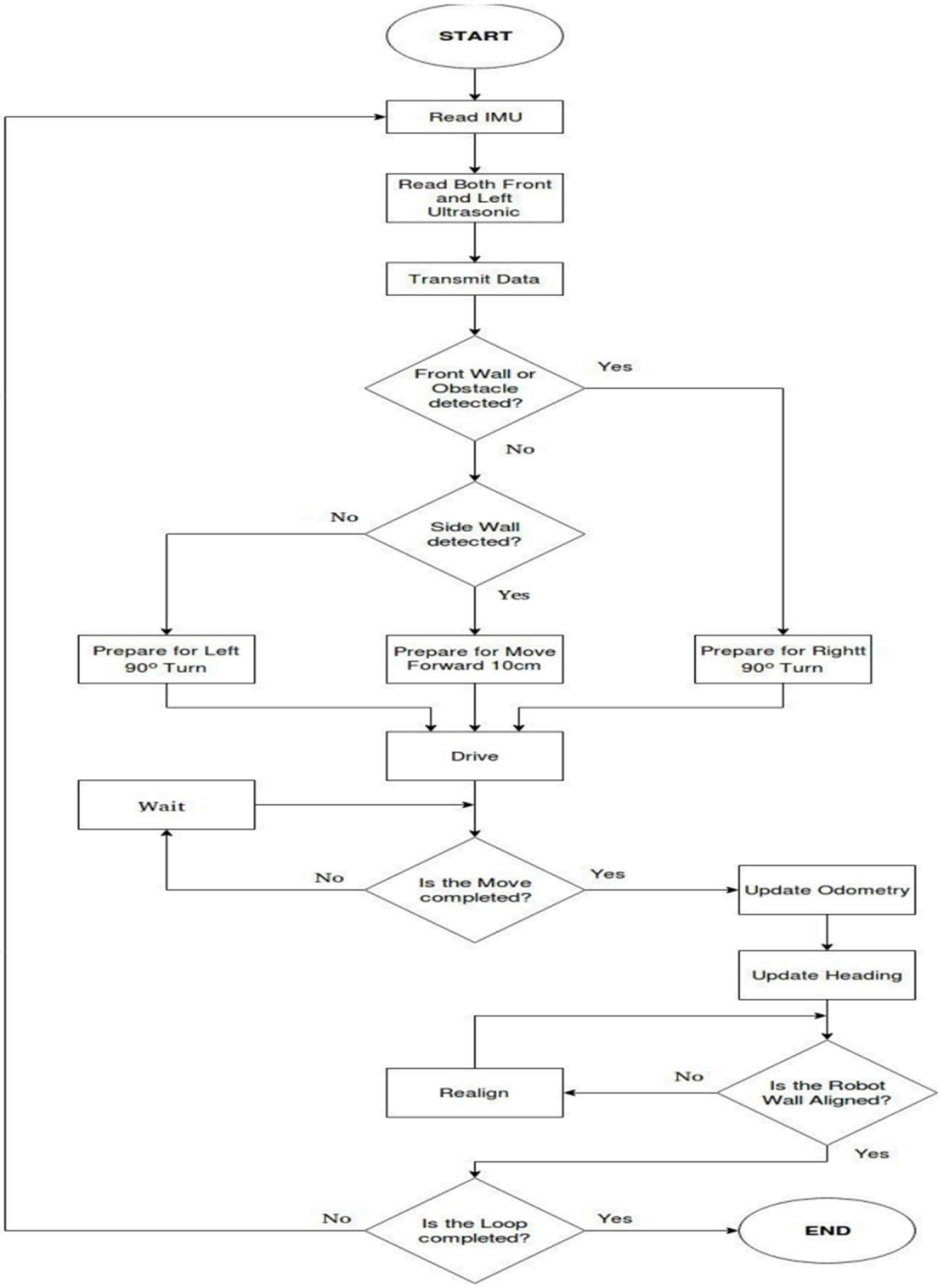
Εικόνα 6 2 Σενάριο Κίνησης Πειράματος

Αφού υπολογιστούν οι αριθμοί επαναλήψεων για να καλύψω όλη την περίμετρο το ρομπότ υπολογίζει τις νέες συντεταγμένες του «φανταστικού» τοίχου που θα ακολουθήσει. Όπως φαίνεται πάνω. Η τρέχουσα κατάσταση του fsm τότε μεταβαίνει σε NextLoops. Εκείνη τη στιγμή το ρομπότ ξεκινά να κινείται κατευθυνόμενο στις υπολογισμένες συντεταγμένες του πρώτου «φανταστικού» τοίχου και συνεχίζει ακολουθώντας το και παρατηρώντας την μη-εξερευνημένη περιοχή του χώρου. Σημαντικό είναι να αναφέρουμε πως το ρομπότ κινείται παρόμοια στις FirstLoop και NextLoops. Στην First Loop ακολουθεί έναν πραγματικό τοίχο και στην Next loops ακολουθεί έναν «φανταστικό» τοίχο μιας και η αντίληψη του για αυτόν έχει προέλθει από τον υπολογισμό με βάση τις μετρήσεις που έχει για αυτόν οι οποίες φυσικά παρουσιάζουν σφάλμα.



Εικόνα 6 3 Σενάριο Κίνησης Πειράματος (2)

Για να κατανοήσω καλύτερα πως λειτουργεί εξετάζω το παρακάτω σχήμα που δείχνει όλα τα αναγκαία βήματα για την πλοήγηση του ρομπότ [1].



Εικόνα 6 4 Flow Chart Λογικής Κίνησης

Το flow chart δείχνει πως κινείται το ρομπότ μου. Όταν η κατάσταση μου είναι είτε First loop είτε Next loop το ρομποτ υπολογίζει τη γωνία που κινείται και ύστερα κοιτάζει για εμπόδια ή τοίχους τριγύρω και μετρά την απόσταση με υπέρηχους. Σε αυτό το σημείο το ρομπότ πρέπει αφού αποθηκεύσει αυτά τα

δεδομένα να αποφασίσει με βάση τις προηγούμενες μετρήσεις αν θα στρίψει αριστερά, δεξιά ή αν θα συνεχίσει ευθεία. Εάν δεν υπάρχει εμπόδιο ή τοίχος μπροστά και έχει τοίχο αριστερά τότε κινείται μπροστά. Αν μπροστά εμφανίζεται τοίχος, τότε το ρομπότ στρίβει δεξιά και όταν δεν ανιχνεύει κανέναν τοίχο, τότε στρίβει αριστερά. Το ρομπότ χρησιμοποιεί τον αισθητήρα για να στρίψει ακριβώς 90 μοίρες.

Προσπαθώντας να ολοκληρώσει την κίνηση, το ρομπότ ανανεώνει τις συντεταγμένες του και την πορεία του. Αξίζει να αναφέρω πως οι αρχικές συντεταγμένες θεωρώ ότι είναι [0,0]. Το ρομπότ αυτή τη στιγμή ελέγχει εάν είναι ευθυγραμμισμένο με τον τοίχο αριστερά. Σε περίπτωση που δεν είναι το ρομπότ επανέρχεται σε ευθεία με τον τοίχο και συνεχίζει. Σημειωτέων πως θα μεταβεί στην κατάσταση NextLoops και θα ολοκληρώσει το σκαναρισμα του First Loops μόλις βρεθεί πάλι στις αρχικές συντεταγμένες.

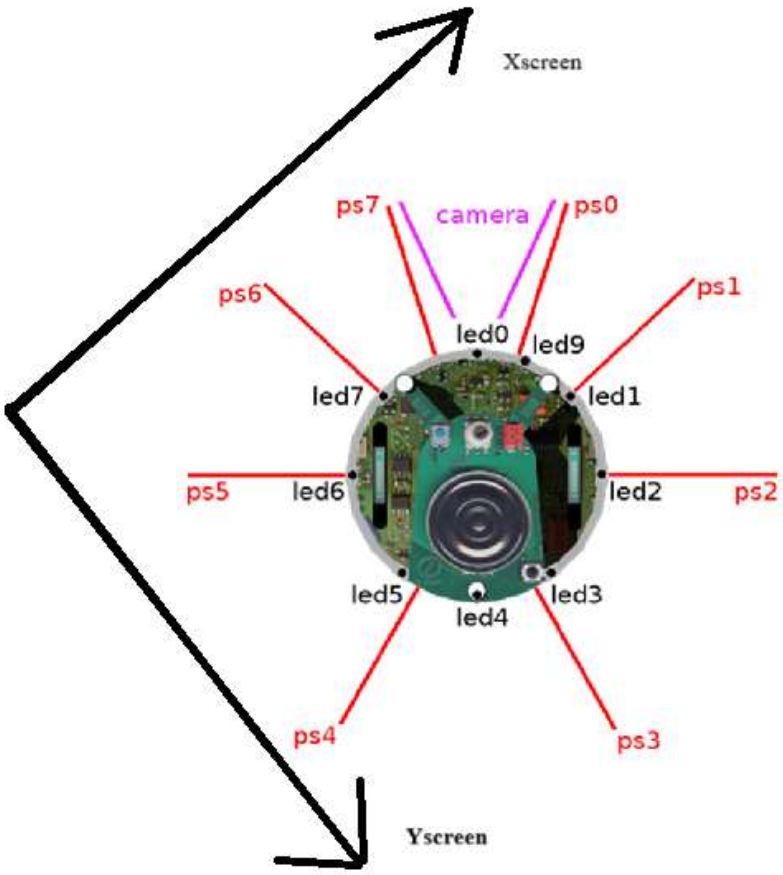
Κεφάλαιο 6.2: Χαρτογράφηση και βήμα επεξεργασίας

Το κομμάτι της χαρτογράφησης βασίζεται στην τεχνική του χάρτη δικτύου πληρότητας (Occupancy Grid Map).

Έγινε χρήση Python όπως επίσης και χρήση μερικών άλλων βιβλιοθηκών για να καταφέρουμε επεξεργασία των δεδομένων. Οι 2 πιο σημαντικές ήταν η serial library η οποία βρίσκεται ενσωματωμένη στο webots και ευθύνεται για την επικοινωνία του ρομπότ με τον έξω κόσμο και η δεύτερη ήταν η matplotlib η οποία σε συνδυασμό με το pyplot package μας βοήθησε να οπτικοποιήσουμε τα δεδομένα του ρομπότ για να μπορέσουμε να δούμε το περιβάλλον όπως το αντιλαμβάνεται μέσα από τους αισθητήρες του το ρομπότ.

Αρχικά, το σύστημα αποθηκεύει σε έναν πίνακα 19 μετρήσεις περί ενός σημείου για να προσδιορίσει τον περιβάλλοντα χώρο σε μορφή radians. Στη συνέχεια δημιουργεί ένα δεύτερο πίνακα και ο οποίος αποτελείται από 3 κελιά τα οποία συγκρατούν τις συντεταγμένες στις οποίες πιστεύει το ρομπότ ότι βρίσκεται καθώς επίσης και την κατεύθυνση του κάθε φορά που γίνεται μια νέα μέτρηση για το περιβάλλον του. Τα δεδομένα αυτά σε συνδυασμό με τις προηγούμενες μετρήσεις πέριξ του ρομπότ χρησιμοποιούνται για να ενημερωθεί το στιγμιότυπο του χάρτη. Τέλος, καθορίζεται το μέγεθος του δικτύου στο χάρτη με κάθε κελί να είναι ενημερωμένο σχετικά με την κατάληψη του ή όχι από κάποιο εμπόδιο.

Σημειωτέο πως όλες οι μετρήσεις από τους αισθητήρες έχουν παρθεί με βάση το σύστημα συντεταγμένων του ρομπότ και άρα θα πρέπει να μετασχηματιστούν για να ανταποκρίνονται στην πραγματικότητα. Αυτό μπορεί να επιτευχθεί κάνοντας επαναπροσδιορισμό της κατεύθυνσης του ρομπότ(recalibration).



Εικόνα 6 5 Σύστημα συντεταγμένων Ρομπότ

Έτσι, με κάθε λήψη μέτρησης από το ρομπότ έχω αποκωδικοποίηση δεδομένων και αντιστοίχιση τους σε πραγματικά δεδομένα καθώς και ενημέρωση του χάρτη. Το σύστημα αποφασίζει σε κάθε βήμα ποια κελιά του χάρτη θα ενημερωθούν υπολογίζοντας και την L-norm απόσταση προς όλα τα κελιά από το ρομπότ. Μόλις λάβω την πληροφορία πως κάποια κελιά είναι εντός απόστασης από τους αισθητήρες, τότε υπολογίζεται η πιθανότητα να είναι κατειλλημένα από κάποιο εμπόδιο ή από κάποιον τοίχο. Είναι σημαντικό να τονίσουμε πως η εκδοχή του χάρτη που γνωρίζει το ρομπότ κάθε φορά εμφανίζεται μέσω της διαδικασίας SLAM. Η προσομοίωση μου και ο σχεδιασμός του χάρτη λαμβάνει τέλος όταν το ρομπότ μπει στην τελική κατάσταση.

Κεφάλαιο 6.3: Επικύρωση σωστής λειτουργίας

Η επιλογή των λειτουργικών μερών του ρομπότ και ο τρόπος λειτουργίας του έχουν συζητηθεί σε προηγούμενα κεφάλαια. Ωστόσο, η κατασκευή από μόνη της δεν αρκεί. Για να δείξω την αρτιότητα του συστήματος θα πρέπει να επικυρώσω τη λειτουργία του και για αυτό υποβάλλω το ρομπότ σε δοκιμασίες που θα φέρουν εις πέρας το σκοπό αυτό. Έτσι, θα σιγουρευτώ πως το σχέδιο και η λειτουργικότητα του είναι ορθά κατασκευασμένα από όλες τις απόψεις. Εδώ πρέπει να σημειωθεί πως το ρομπότ χρησιμοποιήθηκε αποκλειστικά για τη συλλογή δεδομένων και κάνω την παραδοχή πως σε ένα τέτοιο

σύστημα αρκούμαι και στο σχεδιασμό των αποτελεσμάτων με χρήση κώδικα και όχι αμιγούς χρήσης της πλατφόρμας webots η οποία στο συγκεκριμένο κομμάτι παρουσιάζει μερικούς περιορισμούς αφού ακόμα βρίσκεται υπό ανάπτυξη. Συγκεκριμένα, η επικύρωση λειτουργίας των αισθητήρων γίνεται δοκιμάζοντας να κάνω μετρήσεις σε γνωστές αποστάσεις τόσο πριν όσο και μετά την προσομοίωση στο ίδιο εικονικό μοντέλο ρομπότ για να βεβαιώσω την εγκυρότητα στις μετρήσεις. Δοκιμάζεται επίσης σενάρια εκκίνησης και τερματισμού τα οποία μας δείχνουν την απόκριση από τους κινητήρες του οχήματος και τυχόν αστοχίες ή μικρά σφάλματα για να βεβαιωθώ πως λειτουργούν σωστά.

Οι αισθητήρες είναι ικανοί να ανιχνεύσουν αντικείμενα έως μια απόσταση 4000mm στο webots, αλλά αυτό το νούμερο δεν αντιστοιχεί σε μια υλοποίηση χαμηλού κόστους στον πραγματικό κόσμο και για αυτό το τροποποίησα στα 30cm περίπου κάνοντας τις αντίστοιχες μετατροπές στον πίνακα του αισθητήρα. Επίσης, διευκολύνει στην κατασκευή του εικονικού κόσμου καθώς η τοποθέτηση και τροποποίηση του περιβάλλοντος απαιτεί λιγότερα βήματα. Επιπλέον, συγχρονίστηκαν τα μοτέρ του ρομπότ κάνοντας χρήση μερικών έτοιμων αποστάσεων σε συνδυασμό με τη απόσταση που μετρά ο προσομοιωτής ότι ταξίδεψε το ρομπότ και έγινε ρύθμιση όπου χρειάστηκε πριν την προσομοίωση για να σιγουρευτούμε πως το ρομπότ θα εκκινείται κινούμενο απόλυτα ευθεία.

Παρόμοια μέτρηση έγινε και στη μονάδα IMU για τον προσανατολισμό του ρομπότ ελέγχοντας κάθε φορά

imu_values = imu.getRollPitchYaw()

print(f"Roll: {imu_values[0]}, Pitch: {imu_values[1]}, Yaw: {imu_values[2]}")

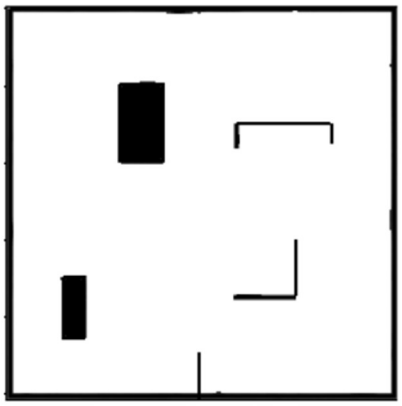
Οι επόμενες δοκιμές στο fsm εμφάνισαν διάφορα προβλήματα τα οποία εάν δε δοκιμάζοντας μεμονωμένα και στη συνέχεια εξ ολοκλήρου τη λειτουργία του fsm δε θα μπορούσαν να εμφανιστούν. Οι δοκιμές αποτελούνταν από ταυτόχρονη αξιολόγηση τόσο της κατάστασης του fsm όσο και των εξόδων των αισθητήρων.

Τέλος, το σύστημά μας αντιπαραβάλλεται με μέτρο τις τιμές που διαβάζουν οι αισθητήρες με δεδομένα που παράγω για το εικονικό περιβάλλον και τα οποία αποτελούν την ιδανική εκδοχή που θα έπρεπε να έχουν οι μετρήσεις μας και βρήκα πως η απόκλιση προσεγγίζει σε καλό βαθμό ένα παρόμοιο πιο κοντά στο ιδανικό ρομπότ που επιτελεί το έργο καλύτερα από το δικό μου. Οι δοκιμές αυτές έγιναν σε μεμονωμένα σημεία του χάρτη και δη σε ακραίες περιπτώσεις όπου η περιστροφή του ρομπότ και οι μετρήσεις των αισθητήρων ήταν σε θέσεις με μεγάλη δυσκολία π.χ. η ανίχνευση ακμής σε εμπόδιο όταν ο αισθητήρας ευθυγραμμίζεται τέλεια με τη γωνία του εμποδίου/τοίχου, ή σε περίπτωση που πρέπει να κάνω $2^{η}$ μέτρηση ενός εμποδίου κατά τη στροφή του ρομπότ.

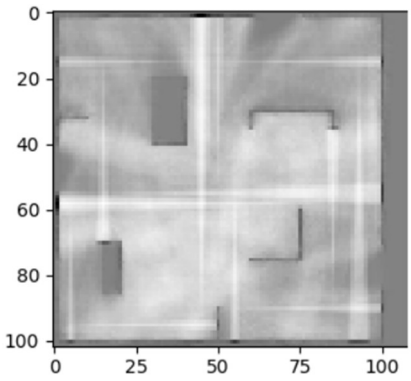
Οι δοκιμές του συστήματος στην εξ' ολοκλήρου λειτουργία του ήταν επιτυχείς με πολύ μικρά ποσοστά αποτυχημένων προσομοιώσεων τα οποία ωστόσο οφείλονται σε αστοχίες του προσομοιωτή για σωστή εκκίνηση και σωστή καταγραφή δεδομένων πράγματα τα οποία έχουν διορθωθεί. Το αποτέλεσμα είναι μια καλή προσέγγιση της δομής του περιβάλλοντος υπό εξέταση. Σημείο που επιδέχεται σημαντικής βελτίωσης είναι η ακρίβεια περιστροφής η οποία λόγω βήματος προσομοίωσης και περιορισμών του εικονικού κόσμου και της δειγματοληψίας που μπορούμε να κάνουμε σε αυτόν είναι η περιστροφή του ρομπότ. Αν και, σε όλα τα σενάρια που εξετάστηκαν δεν κρίθηκε αναγκαίο να γίνει κάποια μεγάλη αλλαγή για σφάλματα που μπορούμε να υπολογίσουμε είναι εμφανής η ανάγκη εισαγωγής περισσότερων σταδίων επεξεργασίας των δεδομένων που συλλέγω. Το ρομπότ χρίζει πολλών βελτιώσεων για να προσομοιάζει ένα πραγματικά μοντέρνο μοντέλο SLAM και κρίνεται πως από την απόδοση του, τις μετρήσεις και την αντιστοιχία αυτών στον πραγματικό κόσμο όπως και από τα διάφορα σενάρια κίνησης

έχουμε καλύψει ένα ικανοποιητικό ποσοστό αντιστοίχισης του εικονικού κόσμου στις μετρήσεις που έχω από το ρομπότ.

Εδώ παρουσιάζεται μια τελειοποιημένη δομή του περιβάλλοντος δεξιά όπως θα έπρεπε να αντιλαμβάνεται το ρομπότ. Αριστερά έχουμε την αναπαράσταση του περιβάλλοντος υπό εξέταση.

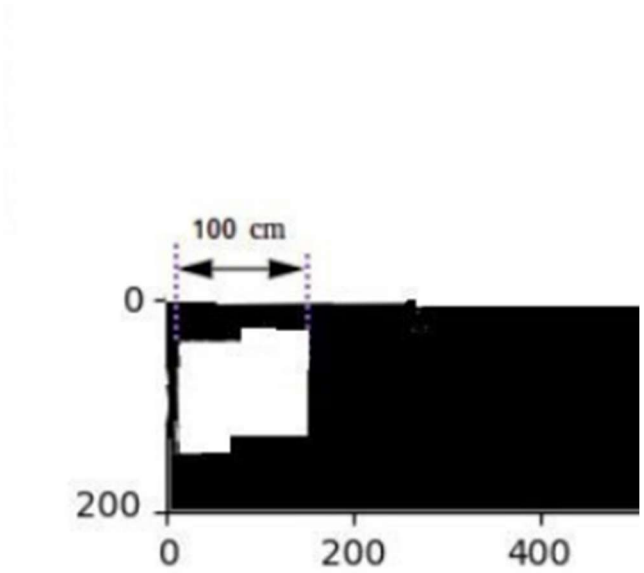


Εικόνα 6 6 Θεωρητικό περιβάλλον εξερεύνησης

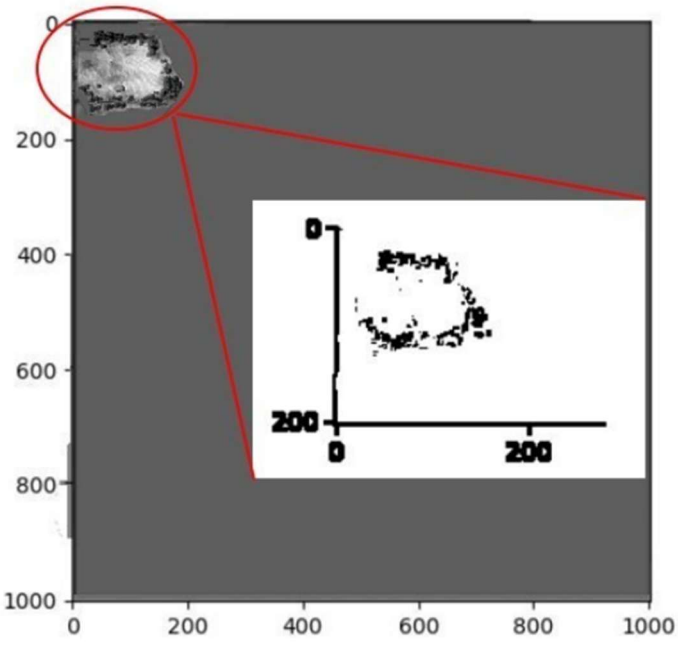


Εικόνα 6 7 Ιδανική ανακατασκευή περιβάλλοντος

Η προσομοίωση μου με παρόμοιο τρόπο σε περιβάλλον που αντιστοιχεί στην αριστερή εικόνα κάτω και στη 2η εμφανίζεται μια αναπαράσταση που έχουν παράξει τα δεδομένα του ρομπότ κατά την προσομοίωση αυτού στο περιβάλλον webots. Η προσομοίωση είναι αποτέλεσμα ενός κύκλου του μηχανισμού που περιγράφηκε παραπάνω και απεικονίζει το νέφος σημείων(point cloud) που ανακατασκευάζουν τα σχεδόν «ωμά» δεδομένα μου.



Εικόνα 6 8 Πειραματικό περιβάλλον εξερεύνησης



Εικόνα 6 9 Πειραματική Ανακατασκευή Περιβάλλοντος

Παρατηρώ πως η αντιστοιχία των επιφανειών δεν είναι όπως στην πραγματικότητα. Αυτό οφείλεται στο γεγονός ότι τα δεδομένα που συλλέξαμε χρειάζονται περαιτέρω επεξεργασία και από έξτρα φίλτρα όπως π.χ. Canny edge detection για να πετύχουμε ένα πιο αξιόλογο αποτέλεσμα. Αυτό οφείλεται στη θορυβώδη φύση του αισθητήρα υπερήχων, το βήμα της προσομοίωσης, τη δομή του περιβάλλοντος και τις

ιδιαιτερότητες του όσο αφορά τα μειονεκτήματα του σε σχέση με τις τεχνικές που κάναμε χρήση για να τον διερμηνεύσουμε.

Τυχόν εμπόδια στα οποία το ρομπότ είναι αναγκασμένο να περιτρυγυρίζει γύρω από αυτά και συγκεκριμένα σε σημεία με πολλές ακμές μας δίνουν μια άλλη προοπτική στο πρόβλημά μας καθώς μπορούν να αναπαράγουν ένα περιβάλλον με περισσότερα δεδομένα. Λόγω του ότι το ρομπότ πιθανώς θα διαβάσει περισσότερες από μια φορές το ίδιο σημείο πέριξ του εμποδίου και θα ενημερώσει την πιθανότητα ενός κελιού στο χάρτη ξανά και ξανά, είναι πιθανό ο περιβάλλων χώρος να διαβαστεί με μεγαλύτερη ακρίβεια συνεισφέροντας σε ένα αποτέλεσμα με πιο πλουσιες λεπτομέρειες και μεγαλύτερη ανάλυση. Το συγκεκριμένο βήμα μπορεί επίσης -απουσία κατάλληλης επεξεργασίας- να δημιουργήσει ανακρίβειες στην ενημέρωση της πιθανότητας ύπαρξης ή μη εμποδίου σε κάποιο σημείο του grid λόγω σύγκλισης ή μη της πιθανότητας ύπαρξης εμποδίων. Αυτό εξαρτάται και από τις εκάστοτε τεχνικές υπό χρήση.

Κεφάλαιο 7:

Συμπεράσματα

Παρακάτω παρουσιάζονται όλα τα συμπεράσματα καθώς επίσης και τα διδάγματα που αποκομίσαμε από την παρούσα εργασία.

Κεφάλαιο 7.1: Συμπεράσματα

Το πρώτο κεφάλαιο της εργασίας προσφέρει μια συνοπτική επισκόπηση του προβλήματος, παρουσιάζοντας επιγραμματικά τις τεχνικές που αναλύονται εκτενώς στα επόμενα κεφάλαια. Παρουσιάζεται τόσο το πλήθος των τεχνικών όσο και ο βαθμός εμβάθυνσης σε κάθε μία.

Για την υλοποίηση, επιλέχθηκε η πλατφόρμα προσομοίωσης Webots, με κύρια γλώσσα προγραμματισμού τη C, σε συνεργασία με την Python για την εξαγωγή και επεξεργασία των αποτελεσμάτων. Μέσα από την ενασχόλησή μου κατέληξα στο συμπέρασμα ότι η συγκεκριμένη προσέγγιση είναι ιδιαίτερα χρήσιμη όταν απαιτείται ταχεία εξοικείωση με το εργαλείο πρωτοτυποποίησης. Πέρα από τις δυνατότητες υλοποίησης, το οπτικό περιβάλλον της πλατφόρμας διευκολύνει την ανάπτυξη, καθώς επιτρέπει στον χρήστη να επικεντρωθεί στην ανάλυση των τεχνικών και όχι στη ρύθμιση των παραμέτρων του περιβάλλοντος.

Το Webots παρέχει έτοιμα assets, ρομπότ, εμπόδια και περιβάλλοντα, κάτι που μειώνει σημαντικά τον χρόνο ανάπτυξης. Ωστόσο, έχει αυξημένες απαιτήσεις σε υπολογιστική ισχύ, καθώς χρειάζεται υπολογιστή με ισχυρή κάρτα γραφικών για την ομαλή προβολή των προσομοιώσεων και την αξιόπιστη αναπαράσταση των κινήσεων του ρομπότ. Αυτό μειώνει την ανάγκη για συνεχείς μετρήσεις σεναρίων, αφού το περιβάλλον προσομοίωσης παρέχει άμεση οπτική επιβεβαίωση.

Αξίζει να σημειωθεί ότι, ως εργαλείο ανοιχτού κώδικα, το Webots παρουσιάζει συχνά σφάλματα ή απρόβλεπτες συμπεριφορές, γεγονός που μπορεί να επηρεάσει την αξιοπιστία των πειραμάτων (π.χ. τυχαίες αποκλίσεις στην κίνηση των τροχών για λόγους ρεαλιστικότητας). Επιπλέον, οι παράμετροι προσομοίωσης παίζουν καθοριστικό ρόλο και είναι σημαντικό να οριστούν από νωρίς, αφού κάθε αλλαγή τους απαιτεί τη δημιουργία νέου περιβάλλοντος, γεγονός που δυσκολεύει τη σύγκριση μεταξύ διαφορετικών σεναρίων.

Σχετικά με τις τεχνικές διήθησης Bayes, η εργασία καταλήγει σε δύο βασικές προσεγγίσεις:

- Είτε στην βελτιστοποίηση των υπολογισμών πιθανοτήτων κατάληψης κελιών (κάτι που ήταν περιορισμένο λόγω του απλού μοντέλου δυαδικού φίλτρου Bayes που χρησιμοποιήθηκε),
- Είτε στην εξερεύνηση φίλτρων που επιτρέπουν εστίαση στην υλοποίηση του σεναρίου κίνησης και στον τρόπο συλλογής δεδομένων.

Η ταυτόχρονη μεταβολή τόσο των φίλτρων όσο και των παραμέτρων αποτελεί πολυδιάστατο πρόβλημα, το οποίο εξετάστηκε κυρίως βιβλιογραφικά, λόγω του τεράστιου αριθμού πιθανών συνδυασμών.

Σε αυτό το πλαίσιο, οι χάρτες κατειλημμένου πλέγματος (occupancy grid maps) προκύπτουν ως κομψή λύση, προσφέροντας έναν εύκολο τρόπο εισαγωγής στις πιθανοτικές τεχνικές για την εξερεύνηση και

χαρτογράφηση άγνωστου χώρου. Παράλληλα, διευκολύνουν την κατανόηση της υπολογιστικής πολυπλοκότητας του προβλήματος.

Οι τεχνικές αυτές εξετάζονται βιβλιογραφικά στο δεύτερο κεφάλαιο, ενώ αξιοποιούνται στο πείραμα για την οπτικοποίηση των δεδομένων και την κατασκευή του χάρτη. Σημαντικός περιορισμός είναι ότι η αρχική θέση και ο προσανατολισμός του ρομπότ πρέπει να είναι γνωστά, κάτι που μειώνει τις ελευθερίες και τη δυνατότητα λεπτομερούς χαρτογράφησης.

Οι προσομοιώσεις χρησιμοποιούν αισθητήρες υψηλής ακρίβειας, και το περιβάλλον επιτρέπει μεγάλη παραμετροποίηση. Ως εκ τούτου, προκύπτει ότι μια πιο πλήρης προσέγγιση απαιτεί τη χρήση διαφορετικών τύπων αισθητήρων και περισσότερες τεχνικές σε όλα τα στάδια: συλλογή, επεξεργασία και χαρτογράφηση.

Στο τέταρτο κεφάλαιο, η βιβλιογραφική επισκόπηση επιβεβαιώνει ότι η υλοποίηση και εφαρμογή αλγορίθμων για την συλλογή, επεξεργασία και χαρτογράφηση δεδομένων αποτελεί πολυδιάστατο και σύνθετο πρόβλημα.

Για τον λόγο αυτό, επέλεξα να περιοριστώ σε 2D χώρους αντί για τρισδιάστατα μοντέλα. Αυτό μου επιτρέπει να ασχοληθώ με θεμελιώδεις έννοιες της εξερεύνησης και χαρτογράφησης, όπως η συνεχής γνώση του προσανατολισμού με κλασικές τεχνικές, αντί να καταπιαστώ με πιο σύνθετα προβλήματα όπως η παραμετροποίηση αλγορίθμων μηχανικής μάθησης ή η συνδυαστική χρήση πολλαπλών αισθητήρων.

Εστιάζοντας σε απλούστερες παραμέτρους και χρησιμοποιώντας ένα είδος αισθητήρα, εξετάζω σε βάθος τις βασικές εξαρτήσεις των τεχνικών και δημιουργώ μια ισχυρή βάση για μελλοντική ενασχόληση με πιο σύνθετες ή αφηρημένες προσεγγίσεις.

Κατά την προσομοίωση του αλγορίθμου του Κεφαλαίου 6, και κάνοντας τις απαραίτητες παραδοχές για τον προσανατολισμό, γίνεται γρήγορα φανερό πως οι ατέλειες του περιβάλλοντος και η εισαγωγή τυχαιότητας στις παραμέτρους κίνησης του ρομπότ από τον προσομοιωτή (και συνεπώς και στις μετρήσεις των αισθητήρων) δημιουργούν νέα προβλήματα. Τα δεδομένα που περιμένουμε να προκύψουν ως «καθαρές» και ευανάγνωστες μετρήσεις μέσω της συνάρτησης οπτικοποίησης, τελικά εμφανίζονται θορυβώδη και σε πολλές περιπτώσεις μη χρήσιμα.

Για να εξαχθούν αξιόπιστα συμπεράσματα, απαιτείται η επεξεργασία και ο μετασχηματισμός αυτών των δεδομένων με κατάλληλες τεχνικές. Το τελικό αποτέλεσμα δείχνει ότι τόσο η επιλογή του κατάλληλου αλγορίθμου όσο και του προσομοιωτικού περιβάλλοντος παίζουν εξίσου σημαντικό ρόλο. Στο Webots, τα σφάλματα εντοπίζονται σχετικά εύκολα, αλλά είναι δύσκολο να περιοριστούν, καθώς πολλές παράμετροι είναι κρυμμένες πίσω από το γραφικό περιβάλλον, το οποίο «αφαιρεί» λεπτομέρειες της λειτουργίας μηχανισμών που μπορεί να ευθύνονται για αυτά τα σφάλματα.

Άλλα εργαλεία, όπως το ROS, θα μπορούσαν να προσφέρουν ένα πιο ευέλικτο και λιγότερο εξαρτημένο από τρισδιάστατα μοντέλα περιβάλλον, ή ακόμη και να επιτρέψουν την εξέταση της θεωρητικής απόδοσης ενός αλγορίθμου χωρίς την ανάγκη σύνθετης προσομοίωσης. Αυτό είναι ιδιαίτερα χρήσιμο όταν η έμφαση της εργασίας δίνεται στην ανάλυση θεμελιωδών εννοιών.

Η παρούσα διπλωματική εργασία παρουσιάζει έναν τρόπο κατασκευής ενός χαμηλού κόστους αυτόνομου ρομποτικού συστήματος για χαρτογράφηση, διατηρώντας παράλληλα χαμηλό και το κόστος της πλατφόρμας ανάπτυξης. Περιγράφονται παρόμοιοι μηχανισμοί και τεχνικές που έχουν αναπτυχθεί, προσφέροντας στον αναγνώστη μια γενική εικόνα για την πλοήγηση σε άγνωστο περιβάλλον με γνωστή θέση και προσανατολισμό.

Γίνεται αναλυτική παρουσίαση όλων των υποσυστημάτων, της συνολικής απόδοσης του συστήματος, καθώς και των πειραματικών αποτελεσμάτων της αυτόνομης λειτουργίας. Παρά την ικανοποιητική απόδοση του ρομπότ τόσο στη χαρτογράφηση όσο και στην αποφυγή εμποδίων, υπάρχουν περιθώρια βελτίωσης για την επίτευξη βέλτιστης λειτουργίας με το συγκεκριμένο μοντέλο αισθητήρα υπερήχων.

Εναλλακτικά, μπορούν να χρησιμοποιηθούν πιο ακριβείς αισθητήρες, όπως υπέρυθροι σε συνδυασμό με φίλτρα low-pass, ή αισθητήρες LIDAR με πεδίο 180° ή ακόμα και πλήρους κάλυψης 360°, αντί του περιορισμένου εύρους που χρησιμοποιήθηκε εδώ. Προτείνεται επίσης η χρήση ROS για προσομοιώσεις και η σύγκριση με άλλες παρόμοιες υλοποιήσεις, καθώς στην παρούσα εργασία αυτό γίνεται μόνο περιορισμένα στο Κεφάλαιο 6 μέσω του Webots.

Τέλος, αν και οι λύσεις που υλοποιούνται στην πράξη έχουν υψηλότερο κόστος από μια απλή προσομοίωση, τα πλεονεκτήματα που προσφέρουν στη μαθησιακή διαδικασία και στην επίτευξη των στόχων – όπως η αξιόπιστη χαρτογράφηση και η αποφυγή εμποδίων – είναι καθοριστικά και δεν θα πρέπει να αγνοούνται σε μελλοντικές μελέτες τέτοιων προβλημάτων.

Κεφάλαιο 7.2: Διδάγματα

Καθώς φτάνω στο τέλος του προπτυχιακού κύκλου σπουδών μου και ολοκληρώνω τη συγγραφή αυτής της διπλωματικής εργασίας, δεν μπορώ παρά να αναλογιστώ όλα τα πολύτιμα διδάγματα που αποκόμισα κατά τη διάρκεια της εκπόνησής της. Η πορεία αυτή υπήρξε για μένα μια γόνιμη διαδικασία μάθησης και προσωπικής αναζήτησης, μέσα από την οποία εμβάθυνα σε θεωρητικά ζητήματα, καλλιέργησα ερευνητική σκέψη και ανέπτυξα τεχνικές δεξιότητες. Μέσα από τη μελέτη, την πειραματική διαδικασία και την επίμονη ενασχόληση, κατάφερα να κατανοήσω σε βάθος το αντικείμενο και να σχεδιάσω ένα σύστημα που ανταποκρίνεται στους στόχους της εργασίας.

Ο κύριος Δερματάς συνέβαλε καθοριστικά στην ολοκλήρωση της διπλωματικής μου εργασίας, προσφέροντας με απλό και κατανοητό τρόπο πολύτιμες επισημάνσεις στα πιο κρίσιμα σημεία της.

Τον ευχαριστώ θερμά.

Βιβλιογραφία

[1] S. Thrun, «Probabilistic robotics,» 2002.

[2] S. Thrun, «Learning Occupancy Grid Maps With Forward Sensor Models».

[3] M. Shadnan και A. Khan, «Simultaneous localization and mapping (SLAM) with an autonomous robot».

[4] M. Monternerlo και S. Thrun, «Simultaneous Localization and Mapping with Unknown Data Association Using FastSLAM».

[5] M. Montemerlo, S. Thrun, D. Koller και B. Wegbreit, «FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem,» 2002.

[6] W. Jeong και K. M. Lee, «CV-SLAM: A new Ceiling Vision-based SLAM technique».

[7] . Institute of Electrical and Electronics Engineers., The proceedings of 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems : IROS 2006 : Beijing, China, October 9-15, 2006, IEEE, 2006, p. 5774.

[8] A. Eliazar και R. Parr, «DP-SLAM: Fast, Robust Simultaneous Localization and Mapping Without Predetermined Landmarks».

[9] T. Collins, J. Collins και C. Ryan, «Occupancy Grid Mapping: An Empirical Evaluation».

[10] N. Barak, I. Gandhi, N. Gaba και S. Aggarwal, «Two Dimensional Mapping by using Single Ultrasonic Sensor Digital Holography View project Two Dimensional Mapping by using Single Ultrasonic Sensor».

[11] D. Arun Patil και S. Vinod Agiwal, «Design and Implementation of Mapping Robot using Digital Magnetic Compass and Ultrasonic Sensor».

[12] L. Armesto, «SLAM Based on Kalman Filter for Multi-rate Fusion of Laser and Encoder Measurements,» 2004.

[13] J. Ø. Amsen, «Improving Navigation and Mapping with Arduino robot».

[14] M. Abdel, K. Jaradat και R. Langari, «Line Map Construction using a Mobile Robot with a Sonar Sensor».

[15] Y. Qu, M. Yang, J. Zhang, W. Xie, B. Qiang, and J. Chen, “An Outline of Multi-Sensor Fusion Methods for Mobile Agents Indoor Navigation,” *Sensors*, vol. 21, no. 5, p. 1605, Feb. 2021, doi: 10.3390/s21051605.

[16] D. Bose, K. Mohan, M. CS, M. Yadav, and D. K. Saini, “Review of Autonomous Campus and Tour Guiding Robots with Navigation Techniques,” *Australian Journal of Mechanical Engineering*, vol. 21, no. 5, pp. 1580–1590, Oct. 2023, doi: 10.1080/14484846.2021.2023266.

[17] M. B. Alatise and G. P. Hancke, “A Review on Challenges of Autonomous Mobile Robot and Sensor Fusion Methods,” *IEEE Access*, vol. 8, pp. 39830–39846, 2020, doi: 10.1109/ACCESS.2020.2975643.

[20] H. Obeidat, W. Shuaieb, O. Obeidat, and R. Abd-Alhameed, “A Review of Indoor Localization Techniques and Wireless Technologies,” *Wireless Personal Communications*, vol. 119, no. 1, pp. 289–327, Jul. 2021, doi: 10.1007/s11277-021-08209-5.

[21] P. K. Panigrahi and S. K. Bisoy, “Localization strategies for autonomous mobile robots: A review,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, pp. 6019–6039, Sep. 2022, doi: 10.1016/j.jksuci.2021.02.015.

[22] J. Huang, S. Junginger, H. Liu, and K. Thurow, “Indoor Positioning Systems of Mobile Robots: A Review,” *Robotics*, vol. 12, no. 2, p. 47, Mar. 2023, doi: 10.3390/robotics12020047.

[23] T. Kim Geok *et al.*, “Review of Indoor Positioning: Radio Wave Technology,” *Applied Sciences*, vol. 11, no. 1, p. 279, Dec. 2020, doi: 10.3390/app11010279.

[24] X. Han, S. Li, X. Wang, and W. Zhou, “Semantic Mapping for Mobile Robots in Indoor Scenes: A Survey,” *Information*, vol. 12, no. 2, p. 92, Feb. 2021, doi: 10.3390/info12020092.

[25] W. Chen *et al.*, “SLAM Overview: From Single Sensor to Heterogeneous Fusion,” *Remote Sensing*, vol. 14, no. 23, p. 6033, Nov. 2022, doi: 10.3390/rs14236033.

[26] W. Chen *et al.*, “An Overview on Visual SLAM: From Tradition to Semantic,” *Remote Sensing*, vol. 14, no. 13, p. 3010, Jun. 2022, doi: 10.3390/rs14133010.

[27] C. Chou, H. Li, and D. Song, “Encoder-Camera-Ground Penetrating Radar Sensor Fusion: Bimodal Calibration and Subsurface Mapping,” *IEEE Transactions on Robotics*, vol. 37, no. 1, pp. 67–81, Feb. 2021, doi: 10.1109/TRO.2020.3010640.

[28] H. Shi, L. Shi, M. Xu, and K.-S. Hwang, “End-to-End Navigation Strategy With Deep Reinforcement Learning for Mobile Robots,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 4, pp. 2393–2402, Apr. 2020, doi: 10.1109/TII.2019.2936167.

[29] H. Zeng, X. Song, and S. Jiang, “Multi-Object Navigation Using Potential Target Position Policy Function,” *IEEE Transactions on Image Processing*, vol. 32, pp. 2608–2619, 2023, doi: 10.1109/TIP.2023.3263110.

[30] Y. Wang, X. Li, J. Zhang, S. Li, Z. Xu, and X. Zhou, “Review of wheeled mobile robot collision avoidance under unknown environment,” *Science Progress*, vol. 104, no. 3, Jul. 2021, doi: 10.1177/00368504211037771.

[31] H. Qin, S. Shao, T. Wang, X. Yu, Y. Jiang, and Z. Cao, “Review of Autonomous Path Planning Algorithms for Mobile Robots,” *Drones*, vol. 7, no. 3, p. 211, Mar. 2023, doi: 10.3390/drones7030211.

[32] B.-S. Choi and J.-J. Lee, “Mobile robot localization in indoor environment using RFID and sonar fusion system,” in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, Oct. 2009, pp. 2039–2044. doi: 10.1109/IROS.2009.5354104.

[33] M. Ouyang, Z. Cao, P. Guan, Z. Li, C. Zhou, and J. Yu, “Visual-Gyroscope-Wheel Odometry With Ground Plane Constraint for Indoor Robots in Dynamic Environment,” *IEEE Sensors Letters*, vol. 5, no. 3, pp. 1–4, Mar. 2021, doi: 10.1109/LSENS.2021.3057088.

[34] A. Morar *et al.*, “A Comprehensive Survey of Indoor Localization Methods Based on Computer Vision,” *Sensors*, vol. 20, no. 9, p. 2641, May 2020, doi: 10.3390/s20092641.

[35] R. Li, Z. Du, Y. Zhao, and S. Liu, “Design and implementation of mobile robot ultrasonic localization system,” in *2016 Chinese Control and Decision Conference (CCDC)*, IEEE, May 2016, pp. 5347–5352. doi: 10.1109/CCDC.2016.7531954.

[36] T. H. Chan, H. Hesse, and S. G. Ho, “LiDAR-Based 3D SLAM for Indoor Mapping,” in *2021 7th International Conference on Control, Automation and Robotics (ICCAR)*, IEEE, Apr. 2021, pp. 285–289. doi: 10.1109/ICCAR52225.2021.9463503.

[37] K. Borodacz, C. Szczepański, and S. Popowski, “Review and selection of commercially available IMU for a short time inertial navigation,” *Aircraft Engineering and Aerospace Technology*, vol. 94, no. 1, pp. 45–59, Jan. 2022, doi: 10.1108/AEAT-12-2020-0308.

[38] L. Luo, Z. Feng, W. Hong-Wei, and C. Long, “Crawler Robot Indoor Positioning Based on a Combination of Bluetooth and IMU,” in *2022 6th International Conference on Robotics, Control and Automation (ICRCA)*, IEEE, Feb. 2022, pp. 34–39. doi: 10.1109/ICRCA55033.2022.9829003.

[39] M. Alatise and G. Hancke, “Pose Estimation of a Mobile Robot Based on Fusion of IMU Data and Vision Data Using an Extended Kalman Filter,” *Sensors*, vol. 17, no. 10, p. 2164, Sep. 2017, doi: 10.3390/s17102164.

[40] S. Zhang, X. Tan, and Q. Wu, “Self-Positioning for Mobile Robot Indoor Navigation Based on Wheel Odometry, Inertia Measurement Unit and Ultra Wideband,” in *2021 5th International Conference on Vision, Image and Signal Processing (ICVISP)*, IEEE, Dec. 2021, pp. 105–110. doi: 10.1109/ICVISP54630.2021.00027.

[41] S.-J. Lee, K. Lee, and J.-B. Song, “Development of advanced grid map building model based on sonar geometric reliability for indoor mobile robot localization,” in *2014 11th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, IEEE, Nov. 2014, pp. 292–297. doi: 10.1109/URAI.2014.7057437.

[42] H. Liu, F. Sun, B. Fang, and X. Zhang, “Robotic Room-Level Localization Using Multiple Sets of Sonar Measurements,” *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 1, pp. 2–13, Jan. 2017, doi: 10.1109/TIM.2016.2618978.

[43] Y. Liu *et al.*, “Mobile Robot Localisation and Navigation Using LEGO NXT and Ultrasonic Sensor,” in *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, IEEE, Dec. 2018, pp. 1088–1093. doi: 10.1109/ROBIO.2018.8665350.

[44] C.-C. Hsu, H.-C. Chen, C.-C. Wong, and C.-Y. Lai, “Omnidirectional Ultrasonic Localization for Mobile Robots,” *Sensors and Materials*, vol. 34, no. 2, p. 453, Feb. 2022, doi: 10.18494/SAM3419.

[45] M. Shen, Y. Wang, Y. Jiang, H. Ji, B. Wang, and Z. Huang, “A New Positioning Method Based on Multiple Ultrasonic Sensors for Autonomous Mobile Robot,” *Sensors*, vol. 20, no. 1, p. 17, Dec. 2019, doi: 10.3390/s20010017.

[46] C.-C. Hsu, H.-C. Chen, C.-C. Wong, and C.-Y. Lai, “Omnidirectional Ultrasonic Localization for Mobile Robots,” *Sensors and Materials*, vol. 34, no. 2, p. 453, Feb. 2022, doi: 10.18494/SAM3419.

[47] S. Yuan *et al.*, “DSmT-Based Ultrasonic Detection Model for Estimating Indoor Environment Contour,” *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 7, pp. 4002–4014, Jul. 2020, doi: 10.1109/TIM.2019.2943715.

[48] H. Takai, M. Miyake, K. Okuda, and K. Tachibana, “A simple obstacle arrangement detection algorithm for indoor mobile robots,” in *2010 2nd International Asia Conference on Informatics in*

Control, Automation and Robotics (CAR 2010), IEEE, Mar. 2010, pp. 110–113. doi: 10.1109/CAR.2010.5456759.

[49] J.-H. Jean and J.-L. Wang, "Development of an indoor patrol robot based on ultrasonic and vision data fusion," in *2013 IEEE International Conference on Mechatronics and Automation*, IEEE, Aug. 2013, pp. 1234–1238. doi: 10.1109/ICMA.2013.6618090.

[50] T. Grami and A. Sghaier Tlili, "Indoor Mobile Robot Localization based on a Particle Filter Approach," in *2019 19th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, IEEE, Mar. 2019, pp. 47–52. doi: 10.1109/STA.2019.8717214.

[51] M. Derkach, D. Matiuk, and I. Skarga-Bandurova, "Obstacle Avoidance Algorithm for Small Autonomous Mobile Robot Equipped with Ultrasonic Sensors," in *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, IEEE, May 2020, pp. 236–241. doi: 10.1109/DESSERT50317.2020.9125019.

[52] T.-L. Habich, M. Stuede, M. Labbe, and S. Spindeldreier, "Have I been here before? Learning to Close the Loop with LiDAR Data in Graph-Based SLAM," in *2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, IEEE, Jul. 2021, pp. 504–510. doi: 10.1109/AIM46487.2021.9517565.

[53] J. Meng *et al.*, "Efficient and Reliable LiDAR-Based Global Localization of Mobile Robots Using Multiscale/Resolution Maps," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–15, 2021, doi: 10.1109/TIM.2021.3093933.

[54] G. Grisetti, C. Stachniss, and W. Burgard, "Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 34–46, Feb. 2007, doi: 10.1109/TRO.2006.889486.

[55] S. Kohlbrecher, O. von Stryk, J. Meyer, and U. Klingauf, "A flexible and scalable SLAM system with full 3D motion estimation," in *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*, IEEE, Nov. 2011, pp. 155–160. doi: 10.1109/SSRR.2011.6106777.

[56] K. Konolige, G. Grisetti, R. Kümmerle, W. Burgard, B. Limketkai, and R. Vincent, "Efficient Sparse Pose Adjustment for 2D mapping," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, Oct. 2010, pp. 22–29. doi: 10.1109/IROS.2010.5649043.

[57] W. Hess, D. Kohler, H. Rapp, and D. Andor, “Real-time loop closure in 2D LIDAR SLAM,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, May 2016, pp. 1271–1278. doi: 10.1109/ICRA.2016.7487258.

[58] J. Biswas and M. Veloso, “WiFi localization and navigation for autonomous indoor mobile robots,” in *2010 IEEE International Conference on Robotics and Automation*, IEEE, May 2010, pp. 4379–4384. doi: 10.1109/ROBOT.2010.5509842.

[59] D. Schwarz, M. Schwarz, J. Stückler, and S. Behnke, “Cosero, Find My Keys! Object Localization and Retrieval Using Bluetooth Low Energy Tags,” 2015, pp. 195–206. doi: 10.1007/978-3-319-18615-3_16.

[60] A. Loganathan, N. Ahmad, and P. Goh, “Self-Adaptive Filtering Approach for Improved Indoor Localization of a Mobile Node with Zigbee-Based RSSI and Odometry,” *Sensors*, vol. 19, no. 21, p. 4748, Nov. 2019, doi: 10.3390/s19214748.

[61] J. Xin, G. Xie, B. Yan, M. Shan, P. Li, and K. Gao, “Multimobile Robot Cooperative Localization Using Ultrawideband Sensor and GPU Acceleration,” *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 4, pp. 2699–2710, Oct. 2022, doi: 10.1109/TASE.2021.3117949.

[62] B. Yang and E. Yang, “A Survey on Radio Frequency based Precise Localisation Technology for UAV in GPS-denied Environment,” *Journal of Intelligent & Robotic Systems*, vol. 103, no. 3, p. 38, Nov. 2021, doi: 10.1007/s10846-021-01500-4.

[63] Junyang Shen, A. F. Molisch, and J. Salmi, “Accurate Passive Location Estimation Using TOA Measurements,” *IEEE Transactions on Wireless Communications*, vol. 11, no. 6, pp. 2182–2192, Jun. 2012, doi: 10.1109/TWC.2012.040412.110697.

[64] I. Guvenc and C.-C. Chong, “A Survey on TOA Based Wireless Localization and NLOS Mitigation Techniques,” *IEEE Communications Surveys & Tutorials*, vol. 11, no. 3, pp. 107–124, 2009, doi: 10.1109/SURV.2009.090308.

[65] K. C. Ho, “Bias Reduction for an Explicit Solution of Source Localization Using TDOA,” *IEEE Transactions on Signal Processing*, vol. 60, no. 5, pp. 2101–2114, May 2012, doi: 10.1109/TSP.2012.2187283.

[66] F. Zafari, A. Gkelias and K. K. Leung, "A Survey of Indoor Localization Systems and Technologies," in IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2568-2599, thirdquarter 2019, doi: 10.1109/COMST.2019.2911558.

[67] Y. Wang and K. C. Ho, “An Asymptotically Efficient Estimator in Closed-Form for 3-D AOA Localization Using a Sensor Network,” *IEEE Transactions on Wireless Communications*, vol. 14, no. 12, pp. 6524–6535, Dec. 2015, doi: 10.1109/TWC.2015.2456057.

[68] S. Mazuelas *et al.*, “Robust Indoor Positioning Provided by Real-Time RSSI Values in Unmodified WLAN Networks,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 5, pp. 821–831, Oct. 2009, doi: 10.1109/JSTSP.2009.2029191.

[69] S. He and S. . -H. G. Chan, "Wi-Fi Fingerprint-Based Indoor Positioning: Recent Advances and Comparisons," in IEEE Communications Surveys & Tutorials, vol. 18, no. 1, pp. 466-490, Firstquarter 2016, doi: 10.1109/COMST.2015.2464084.

[70] P. Davidson and R. Piché, "A Survey of Selected Indoor Positioning Methods for Smartphones," in IEEE Communications Surveys & Tutorials, vol. 19, no. 2, pp. 1347-1370, Secondquarter 2017, doi: 10.1109/COMST.2016.2637663.

Παραρτήματα

Παράρτημα Α:

Συναρτήσεις:

Εδώ παρουσιάζονται οι συναρτήσεις που χρησιμοποιεί το σύστημα.

Void send_measurements():

Λειτουργία που αποστέλλει μετρήσεις του περιβάλλοντος του ρομπότ. Στέλνουμε την τεχνητή κατεύθυνση (**AHeading**) και την απόσταση που διένυσε το ρομπότ, προκειμένου να υπολογίσουμε τις συντεταγμένες του στο πλέγμα. Στέλνουμε επίσης όλες τις μετρηθείσες αποστάσεις των αντικειμένων γύρω από το ρομπότ, για να τις επεξεργαστούμε και να δημιουργήσουμε τον αντίστοιχο χάρτη. Η μορφή με την οποία αποστέλλονται οι πληροφορίες πρέπει να είναι η εξής, ώστε να αντιστοιχεί σωστά όταν διαβάζεται από το script της Python.

Void end():

Η λειτουργία βοηθά στον τερματισμό του script Python και στην εμφάνιση του τελικού χάρτη όταν το ρομπότ σταματά να κινείται και να συλλέγει νέες πληροφορίες.

Void side_Sonar_Read():

Αυτή η λειτουργία χρησιμοποιείται για να υπολογίσει την απόσταση μεταξύ του ρομπότ και του τοίχου στα αριστερά του. Με αυτόν τον τρόπο, μπορούμε να αντιληφθούμε εάν υπάρχει τοίχος, ώστε να μπορούμε να τον ακολουθήσουμε. Ο αισθητήρας που είναι υπεύθυνος για αυτή την εργασία βρίσκεται στο πίσω αριστερό μέρος του ρομπότ. Αυτές οι πληροφορίες απόστασης, καθώς και οι πληροφορίες που μετρώνται από τη λειτουργία που ακολουθεί, όταν η θέση του σέρβο είναι 180 μοίρες, θα μας βοηθήσουν να ευθυγραμμίσουμε το ρομπότ με τον τοίχο δίπλα του.

Void front_Sonar_Read():

Αυτή η λειτουργία μας βοηθά να μετρήσουμε την απόσταση μεταξύ του ρομπότ και των αντικειμένων, εάν υπάρχουν. Συλλέγει 19 μετρήσεις από έναν υπερηχητικό αισθητήρα που είναι τοποθετημένος πάνω σε έναν κινητήρα σέρβο. Οι 19 μετρήσεις αντιστοιχούν στο εύρος [0-180] μοίρες, με βήμα 10 μοιρών. Η θέση των 0 μοιρών βρίσκεται στη δεξιά πλευρά του ρομπότ, η θέση των 90 μοιρών κοιτάζει μπροστά από το ρομπότ και η θέση των 180 μοιρών είναι στην αριστερή πλευρά του ρομπότ. Είναι σημαντικό να αναφερθεί ότι, για πιο ακριβείς μετρήσεις, επαναλαμβάνουμε τη διαδικασία δύο φορές και αποθηκεύουμε τον μέσο όρο των δύο αποτελεσμάτων.

int CMtoSteps(float cm):

Λειτουργία που μετατρέπει την απόσταση που θα διανύσει το ρομπότ από εκατοστά σε αριθμό βημάτων.

void driveStraightDistance(int steps):

Λειτουργία που οδηγεί το ρομπότ προς τα εμπρός για έναν συγκεκριμένο αριθμό βημάτων.

turnAbsolute(float target):7

Λειτουργία που στρέφει το ρομπότ σε μια συγκεκριμένη γωνία. Υπάρχει μια απόκλιση εντός 0.5 μοιρών από την επιθυμητή γωνία.

void get_heading(float *head):

Αυτή η λειτουργία καλείται κάθε φορά που πρέπει να υπολογιστεί η κατεύθυνση του ρομπότ. Χρησιμοποιείται για να στρίψει το ρομπότ αριστερά ή δεξιά κατά 90 μοίρες με σφάλμα +-0.5 μοιρών.

void CheckIMU(int *state):

Ελέγχει εάν το IMU έχει σταθεροποιηθεί και δίνει σταθερή κατεύθυνση. Εάν όχι, τότε το ρομπότ δεν μπορεί να ξεκινήσει την πλοήγηση.

Παράρτημα Β:

https://cyberbotics.com/doc/reference/distancesensor?tab-language=python

https://cyberbotics.com/doc/reference/distancesensor?tab-language=python#sonar-sensors

https://cyberbotics.com/doc/reference/distancesensor?tab-language=python#distancesensor-types

https://en.wikipedia.org/wiki/Ultrasonic_transducer

https://en.wikipedia.org/wiki/Sensor

https://en.wikipedia.org/wiki/Calibration

https://github.com/carlos-argueta/rse_prob_robotics?tab=readme-ov-file

https://github.com/carlos-argueta/rse_prob_robotics/tree/main/rse_gaussian_filters

https://github.com/carlos-argueta/rse_prob_robotics/tree/main/rse_occupancy_grid_mapping

https://en.m.wikipedia.org/wiki/Occupancy_grid_mapping