

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ

ΡΟΜΠΟΤΙΚΑ ΣΥΣΤΗΜΑΤΑ 1

**Simultaneous Localization And Mapping
with Extended Kalman Filter
and Unknown Correspondences**

Όνοματεπώνυμο:	ΤΣΙΑΤΣΙΑΝΑΣ ΕΥΑΓΓΕΛΟΣ
Αριθμός Μητρώου:	1072369
Έτος φοίτησης:	4 ^ο
Ακ. έτος διεξαγωγής μαθήματος:	2023-2024

Περιεχόμενα

1	Εισαγωγή	1
2	Θεωρητικό υπόβαθρο	2
2.1	Εκτεταμένο Φίλτρο Κάλμαν (EKF)	2
2.2	Αλγόριθμος EKF-SLAM	3
2.3	Κινηματικό Μοντέλο Διαφορικού Ρομπότ	4
2.4	Οδομετρικό Μοντέλο	5
3	Προσομοίωση	6
3.1	Κίνηση και οδομετρία	6
3.2	Αισθητήρες Range – Bearing.....	6
3.3	EKF-SLAM	6
4	Αποτελέσματα	7

1 Εισαγωγή

Ο στόχος της παρούσας εργασίας είναι η επίλυση του προβλήματος Simultaneous Localization and Mapping (SLAM) με άγνωστες συσχετίσεις, χρησιμοποιώντας έναν αλγόριθμο με Extended Kalman Filter (EKF). Το SLAM αποτελεί ένα από τα σημαντικότερα προβλήματα στη ρομποτική, καθώς επιτρέπει σε ένα ρομπότ να δημιουργήσει μια εκτίμηση για το περιβάλλον στο οποίο κινείται και ταυτόχρονα να προσδιορίσει την κατάσταση του (θέση και προσανατολισμός) εντός αυτού, χρησιμοποιώντας μετρήσεις από αισθητήρες και μοντέλα για αυτούς και τις κινήσεις του.

Βασικός ρόλος των Kalman Filters είναι να συνδυάζουν τις προβλέψεις από ένα μοντέλο κίνησης με δεδομένα από αισθητήρες με αποτέλεσμα να προκύπτει μια αρκετά καλή προσέγγιση της πραγματικής θέσης του συστήματος. Λαμβάνουν υπόψη την αβεβαιότητα και τον θόρυβο τόσο στις μετρήσεις όσο και στα μοντέλα. Βασίζονται στο κανόνα του Bayes και το αποτέλεσμα τους εκφράζεται μέσω μιας πολυδιάστατης κανονικής κατανομής.

Τα αποτελέσματα του αλγόριθμου επιβεβαιώνονται σε προσομοίωση ενός ρομπότ διαφορικής κίνησης και παρουσιάζονται στο τέλος αυτής της εργασίας.

2 Θεωρητικό υπόβαθρο

Παρακάτω αναλύονται θεωρητικά ορισμένα σημεία που χρησιμοποιήθηκαν για την εργασία.

2.1 Εκτεταμένο Φίλτρο Κάλμαν (EKF)

Όπως αναφέρθηκε παραπάνω, η βασική χρήση των Kalman Filters είναι να υπολογίζουν μια εκτίμηση για μια κατάσταση x με βάση της μετρήσεις των αισθητήρων z και των εισόδων ελέγχου του συστήματος u .

$$p(x | z, u)$$

Σε αντίθεση με τα απλά KF τα οποία προϋποθέτουν γραμμικά μοντέλα κίνησης και παρατήρησης, τα εκτεταμένα φίλτρα Κάλμαν χρησιμοποιούν αναπτύγμα Taylor 1^{ης} τάξης ώστε να προσεγγίσουν γραμμικά τα μοντέλα. Η τελική εκτίμηση που προκύπτει είναι αποτέλεσμα δυο βασικών βημάτων. Πρώτα, με βάση την αρχική εκτίμηση της θέσης του ρομπότ σε συνδυασμό με την είσοδο του ελέγχου και με χρήση του κινηματικού μοντέλου υπολογίζεται μια νέα εκτίμηση για την θέση του ρομπότ. Αυτή διορθώνεται στο δεύτερο βήμα κατά το οποίο χρησιμοποιούνται οι μετρήσεις των αισθητήρων και το μοντέλο παρατήρησης για να προκύψει η τελική εκτίμηση της θέσης. Με βάση το μοντέλο παρατήρησης υπολογίζεται το κέρδος Κάλμαν το οποίο υποδεικνύει το ποσοστό ενσωμάτωσης της εκτίμησης του δεύτερου βήματος στην εκτίμηση του πρώτου. Στον παρακάτω πίνακα παρουσιάζονται τα βήματα του αλγορίθμου[1].

```
EKF_Algorithm( $\mu$ ,  $\Sigma$ ,  $u_t$ ,  $z_t$ ):  
     $\bar{\mu}_t = g(u_t, \mu_{t-1})$   
     $\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t$   
     $K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1}$   
     $\mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t))$   
     $\Sigma_t = (I - K_t H_t) \bar{\Sigma}_t$   
    return  $\mu_t, \Sigma_t$ 
```

Όπου:

- $p(x) = N(\mu_t, \Sigma)$
- $g(u, x)$ το μη γραμμικό μοντέλο κίνησης
- $G(u, x)$ η Ιακοβιανή του μοντέλου κίνησης
- $h(x)$ το μη γραμμικό μοντέλο παρατήρησης
- $H(x)$ η Ιακοβιανή του μοντέλο παρατήρησης

2.2 Αλγόριθμος EKF-SLAM

Ο αλγόριθμος EKF-SLAM βασίζεται σε αυτόν που αναλύθηκε στην προηγούμενη ενότητα. Ωστόσο, πέραν της κατάστασης του ρομπότ \mathbf{x} , η προσέγγιση περιλαμβάνει και τον χάρτη \mathbf{m} [1].

$$p(\mathbf{x}_t, \mathbf{m} \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t})$$

Επομένως η εκτεταμένη κατάσταση \mathbf{y} του συστήματος που εκτιμάται είναι:

$$\mathbf{y}_t = \begin{bmatrix} \mathbf{x}_t \\ \mathbf{m} \end{bmatrix} = [x \quad y \quad \theta \quad lm_{1,x} \quad lm_{1,y} \quad \dots \quad lm_{n,x} \quad lm_{n,y}]^T$$

Όπου $\mathbf{x}_t = [x \quad y \quad \theta]$ η κατάσταση του ρομπότ την χρονική στιγμή t και $[lm_{1,x} \quad lm_{1,y} \quad \dots \quad lm_{n,x} \quad lm_{n,y}]$ οι συντεταγμένες των διακριτών σημειακών στοιχείων που αποτελούν τον χάρτη. Στον παρακάτω πίνακα παρουσιάζεται ο ψευδοκώδικας του αλγορίθμου.

```

EKF_SLAM( $\mu_{t-1}, \Sigma_{t-1}, \mathbf{u}_t, \mathbf{z}_t$ )
   $\mu_t = g(\mu_{t-1}, \mathbf{u}_t)$ 
   $\Sigma_t = G_t \Sigma_{t-1} G_t^T + R_t$ 
  for every observed landmark  $lm$  in  $\mathbf{z}$ :
     $\mu_{lm} = \text{landmark position estimate}$ 
    for every previously observed landmark  $k$ :
       $\mathbf{z}_k = h(\mu_t, \mu_{lm})$ 
       $\Psi_k = H_t^k \Sigma_t [H_t^k]^T + Q_t$ 
       $\pi_k = (\mathbf{z}_{lm} - \mathbf{z}_k)^T \Psi_k^{-1} (\mathbf{z}_{lm} - \mathbf{z}_k)$ 
     $\pi_{lm} = a$ 
     $j = \text{argmin}_k(\pi_k)$ 
    If  $j \neq k$ : add new landmark
     $K_{lm} = \Sigma_t [H_t^j]^T \Psi_j^{-1}$ 
     $\mu_t = \mu_t + K_{lm} (\mathbf{z}_{lm} - \mathbf{z}_j)$ 
     $\Sigma_t = (I - K_{lm} H_t^j) \Sigma_t$ 
  return  $\mu_t, \Sigma_t$ 

```

Στο πρώτο βήμα του αλγορίθμου χρησιμοποιείται κατάλληλα το κινηματικό μοντέλο του ρομπότ ώστε να ενημερώνεται η κατάσταση, χωρίς να επηρεάζει την προσέγγιση των στοιχείων του χάρτη. Στο δεύτερο βήμα, καθώς τα στοιχεία δεν είναι γνωστά εξ' αρχής και η συσχέτιση μεταξύ των μετρήσεων και των στοιχείων που έχουν βρεθεί σε προηγούμενα βήματα είναι άγνωστη, υπολογίζεται η συσχέτιση μεταξύ τους με βάση την ομοιότητα μέγιστης πιθανότητας. Επομένως για κάθε στοιχείο που παρατηρείται στο διάνυσμα \mathbf{z} χρησιμοποιείται το μοντέλο παρατήρησης και εκτιμάται η θέση του στο χώρο με βάση την εκτίμηση της θέσης του ρομπότ από το πρώτο βήμα. Έπειτα ελέγχεται αν υπάρχει κάποιο στοιχείο που έχει παρατηρηθεί σε προηγούμενη χρονική στιγμή και συσχετίζεται με το νέο σύμφωνα με την απόσταση Mahalanobis. Αν αυτό υπάρχει ήδη στο διάνυσμα εκτεταμένης κατάστασης τότε η τιμή του ανανεώνεται, αλλιώς ένα νέο

στοιχείο προστίθεται στην αναπαράσταση του χάρτη. Στη συνέχεια ακολουθούν τα βήματα του αλγορίθμου για την διόρθωση της θέσης του ρομπότ όπως στον αλγόριθμο της προηγούμενης ενότητας.

2.3 Κινηματικό Μοντέλο Διαφορικού Ρομπότ

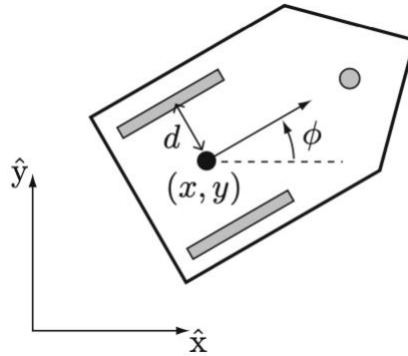


Figure 1. Αναπαράσταση ρομπότ διαφορικής κίνησης

Η κατάσταση του ρομπότ είναι:

$$q = \begin{bmatrix} \theta \\ x \\ y \end{bmatrix}$$

Με $[x, y]$ οι συντεταγμένες του κέντρου του ρομπότ και θ ο προσανατολισμός του. Τα σήματα ελέγχου για κάθε τροχό, είναι:

$$u = \begin{bmatrix} u_L \\ u_R \end{bmatrix}$$

Οι κινηματικές εξισώσεις είναι:

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{r}{2} \cos(\theta) & \frac{r}{2} \cos(\theta) \\ \frac{r}{2} \sin(\theta) & \frac{r}{2} \sin(\theta) \\ -\frac{r}{2d} & \frac{r}{2d} \end{bmatrix} \begin{bmatrix} u_L \\ u_R \end{bmatrix}$$

Όπου d η απόσταση των τροχών από το κέντρο περιστροφής του ρομπότ και r η ακτίνα των τροχών[2].

2.4 Οδομετρικό Μοντέλο

Το οδομετρικό μοντέλο είναι ένας τρόπος υπολογισμού της κίνησης ενός ρομπότ στον χώρο ο οποίος βασίζεται σε οδομετρικά δεδομένα από αισθητήρες έναντι των σημάτων ελέγχου[2]. Ωστόσο, παρότι επιρρεπής σε σφάλματα, είναι συνήθως πιο ακριβής από τα κινηματικά μοντέλα καθώς αυτά πλήττονται επιπλέον από διαφορές μεταξύ των ενεργητών και του προσεγγιστικού μαθηματικού μοντέλου. Για αυτό το λόγω χρησιμοποιούνται συνήθως σε αλγορίθμους φίλτρων (όπως ο EKF_SLAM) ως σήματα ελέγχου. Αυτό μπορεί να αξιοποιηθεί καθώς η σχετική κίνηση που καταγράφεται από την οδομετρία σε χρονικό διάστημα dt προσεγγίζει με αρκετή ακρίβεια την πραγματική μεταβολή της κατάστασης του ρομπότ. Για τον υπολογισμό της σχετικής αυτής κίνησης απαιτείται πρώτα ο υπολογισμός του body twist V_b . Για ένα ρομπότ διαφορικής κίνησης, αυτό υπολογίζεται με βάση την περιστροφή κάθε ρόδας $\Delta\phi$, ως:

$$V_b = \begin{bmatrix} v_{bx} \\ v_{by} \\ \omega_b \end{bmatrix} = r \begin{bmatrix} 1/2 & 1/2 \\ 0 & 0 \\ 1/(2d) & 1/(2d) \end{bmatrix} \begin{bmatrix} \Delta\phi_L \\ \Delta\phi_R \end{bmatrix}$$

Στη συνέχεια με την ολοκλήρωση του Body Twist μπορούμε να υπολογίσουμε την σχετική μετατόπιση $\Delta q_b = [\Delta x_b \ \Delta y_b \ \Delta\omega_b]^T$ που προκύπτει από την μεταβολή $\Delta\phi$, ως:

$$\text{Αν } \omega_b = 0: \Delta q_b = V_b$$

$$\text{Αν } \omega_b \neq 0: \Delta q_b = \begin{bmatrix} (v_{bx} \sin(\omega_b) + v_{bx}(\cos(\omega_b) - 1))/\omega_b \\ (v_{by} \sin(\omega_b) + v_{bx}(1 - \cos(\omega_b)))/\omega_b \\ \omega_b \end{bmatrix}$$

Αυτή η μετατόπιση είναι εκφρασμένη στο πλαίσιο βάσης του ρομπότ, μπορούμε να την εκφράσουμε στο πλαίσιο σταθερό πλαίσιο (World Frame) μέσω του παρακάτω μετασχηματισμού αξιοποιώντας τον αρχικό προσανατολισμό του ρομπότ:

$$\Delta q = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \Delta q_b$$

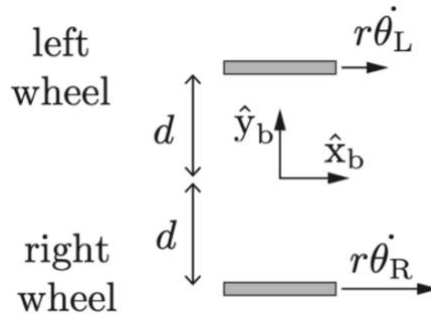


Figure 2. Γεωμετρική αναπαράσταση οδομετρικού μοντέλου

3 Προσομοίωση

Για την προσομοίωση του ρομπότ και την εφαρμογή του αλγορίθμου υλοποιήθηκε ένας προσομοιωτής σε Python.

3.1 Κίνηση και οδομετρία

Τα σήματα ελέγχου δημιουργήθηκαν με βάση κάποιους στόχους στο δισδιάστατο χώρο. Αυτά χρησιμοποιήθηκαν αρχικά με αλλοίωση ως ταχύτητες για το κινηματικό μοντέλο και προσομοίωση της πραγματικής θέσης του ρομπότ έγινε με αριθμητική ολοκλήρωση RK4. Η αλλοίωση προστέθηκε ώστε να ενσωματωθεί η μαθηματική ανακρίβεια του μοντέλου σε σχέση με το πραγματικό ρομπότ. Στη συνέχεια σε αυτά τα σήματα προστέθηκε θόρυβος με σκοπό να προσομοιωθεί ο θόρυβος των οδομετρικών αισθητήρων και έπειτα χρησιμοποιήθηκαν για τον υπολογισμό της οδομετρίας με ολοκλήρωση του Body Twist.

3.2 Αισθητήρες Range - Bearing

Οι θέσεις των στοιχείων του χάρτη με βάση την κατάσταση του ρομπότ q υπολογίζεται από το παρακάτω μοντέλο:

$$\begin{bmatrix} r \\ \varphi \end{bmatrix} = \begin{bmatrix} \left((lm_x - x)^2 + (lm_y - y)^2 \right)^{\frac{1}{2}} \\ atan2(lm_y - y, lm_x - x) - \theta \end{bmatrix}$$

Όπου $[lm_x, lm_y]$ οι συντεταγμένες των στοιχείων του χάρτη στις οποίες προστέθηκε θόρυβος κανονικής κατανομής με στόχο να προσομοιωθεί η αβεβαιότητα των μετρήσεων των αισθητήρων. Αυτό το μοντέλο χρησιμοποιήθηκε και για ως μοντέλο παρατήρησης στον αλγόριθμο EKF-SLAM.

3.3 EKF-SLAM

Εφαρμόστηκε ο αλγόριθμος που επιγράφηκε σε προηγούμενη ενότητα, οι μετρήσεις z λήφθηκαν από το παραπάνω μοντέλο του αισθητήρα. Ως μοντέλο κίνησης χρησιμοποιήθηκε η στοιχειώδης μεταβολή της κατάστασης που προκύπτει από την οδομετρία, ενώ ως μοντέλο παρατήρησης το μοντέλο που περιεγράφηκε παραπάνω χωρίς την προσθήκη θορύβου. Επιπλέον, το όριο για την απόσταση Mahalanobis για την αναγνώριση ενός νέου στοιχείου βρέθηκε πειραματικά και είναι ανάλογο του θορύβου των μοντέλων κίνησης και παρατήρησης.

4 Αποτελέσματα

Οι παράμετροι που χρησιμοποιήθηκαν για τις προσομοιώσεις είναι: $r=0.5$, $d=0.1$, $dt=0.1$, $lidar_width=[\pi\epsilon\omega\varsigma 2\pi]$, $lidar_dist=10$, $lidar_noise=0.001$, $mahalanobis_distance_threshold=[0.5\epsilon\omega\varsigma 2]$, $odometry_noise=0.1$.

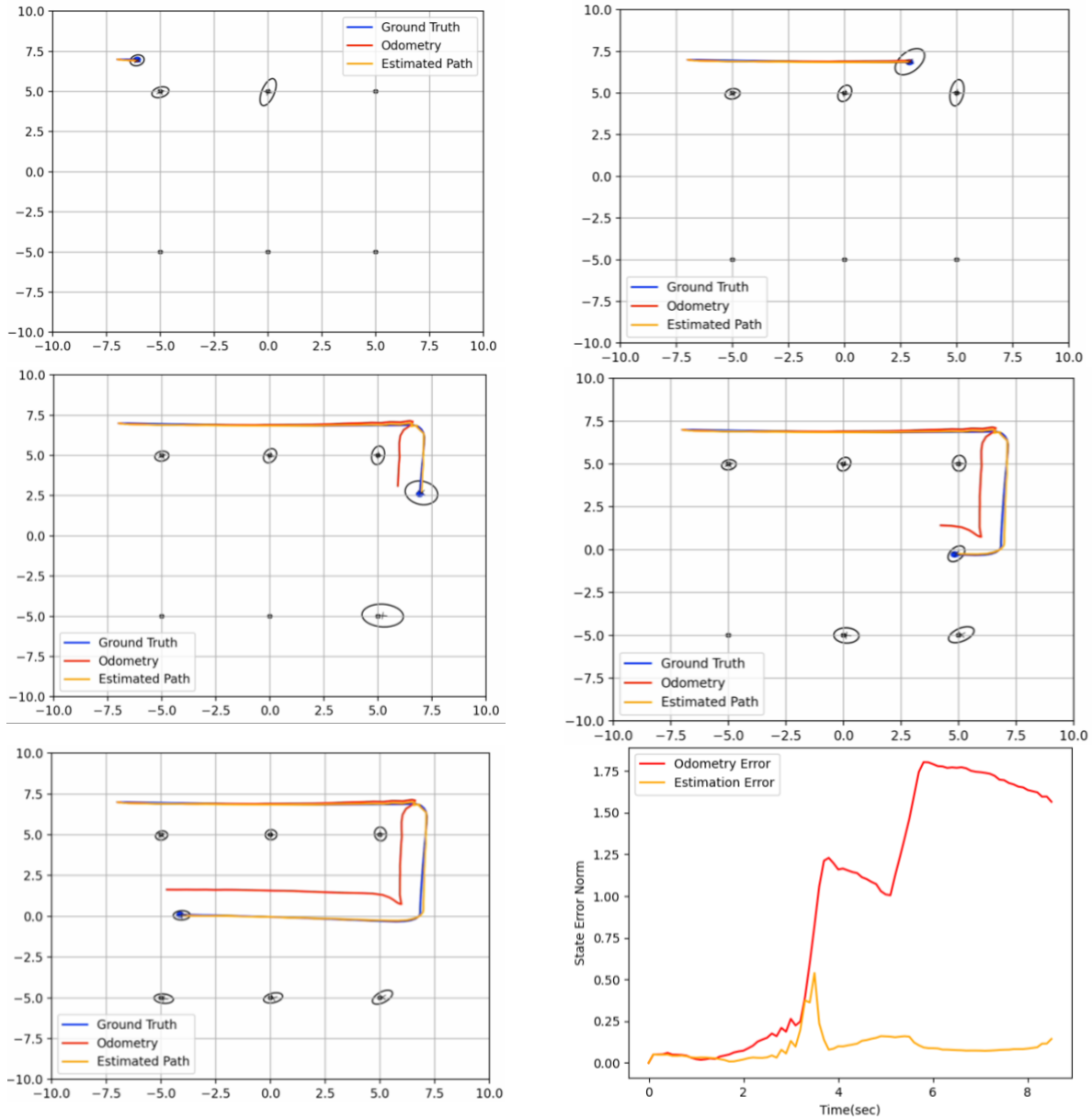


Table 1. Πείραμα 1, διδιάστατη αναπαράσταση προσομοίωσης, σφάλμα κατάστασης

Στο πείραμα 1 παρουσιάζεται η εξέλιξη της εφαρμογής του αλγορίθμου σε ένα περιβάλλον με ακριβώς ορισμένο χάρτη και σημεία ελέγχου τροχιάς. Παρατηρούμε την ανανέωση της εκτίμησης για την θέση των στοιχείων του χάρτη με την πάροδο του χρόνου, ενώ παρατηρούμε πως η εκτίμηση της τροχιάς είναι αρκετά κοντά στην τροχιά που έχει διαγράψει το ρομπότ. Μάλιστα στον πίνακα των σφαλμάτων παρατηρούμε στο 3.7 sec μια σημαντική διόρθωση της εκτίμησης για την θέση του ρομπότ, καθώς και την απόκλιση της οδομετρίας σε σχέση με την πραγματική θέση.

Στις παρακάτω εικόνες παρουσιάζονται δυο πειράματα με τυχαία ομοιόμορφη κατανομή των στοιχείων του χάρτη στο χώρο καθώς και των στόχων για την τροχιά κίνησης. Παρατηρούμε επίσης την αρκετά καλή προσέγγιση της κατάστασης του ρομπότ και του χάρτη. Μάλιστα παρατηρήθηκε πως για χάρτη με μεγάλο αριθμό στοιχείων η ακρίβεια του αλγορίθμου βελτιώνεται.

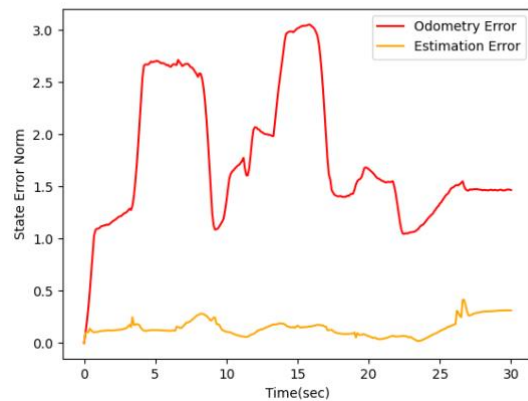
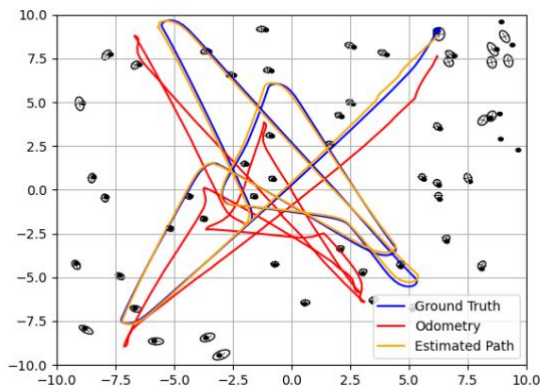
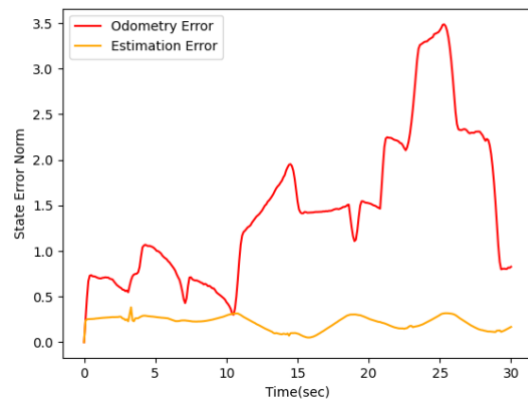
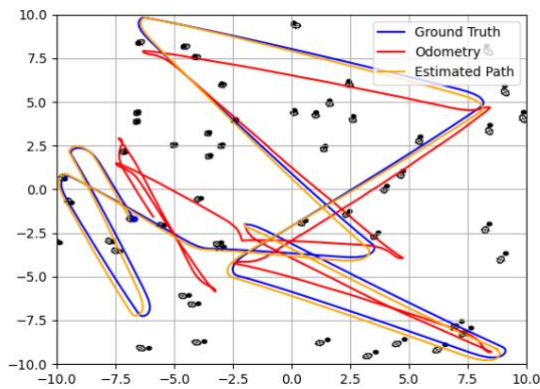


Table 2. Πίνακας πειραμάτων μεγάλου αριθμού στοιχείων με ομοιόμορφη κατανομή

5 Βιβλιογραφία

- [1] Sebastian Thrun, Wolfram Burgard, and Dieter Fox, *Probabilistic Robotics*. 2005.
- [2] Kevin. Lynch and F. C. . Park, *Modern robotics : mechanics, planning, and control*. Cambridge University Press, 2023.