



ΑΡΙΣΤΟΤΕΛΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΟΝΙΚΗΣ

Ρομποτική Αναφορά
Εργασία Τμήμα Α και Τμήμα Β

Διακολουκάς Δημήτριος
ΑΕΜ 10642

Email: ddiakolou@ece.auth.gr

Περιεχόμενα

1	Εργασία Ρομποτικής: Τμήμα Α	2
1.1	Εισαγωγή και περιγραφή του προβλήματος	2
1.2	Περιγραφή υλοποίησης σε MATLAB	2
1.3	Ενδελεχής ανάλυση κώδικα και μεθοδολογίας	3
1.3.1	Γεωμετρική περιγραφή και χρονικά διαστήματα	3
1.3.2	Κατασκευή τροχιών (Trajectory Planning)	3
1.3.3	Κινηματική μοντελοποίηση	4
1.3.4	Ανάλυση τροχιάς	4
1.3.5	Μαθηματική απεικόνιση του προβλήματος	5
1.4	Παρατηρήσεις και Αποτελέσματα	5
1.4.1	Στιγμιότυπα Κίνησης Πόρτας και Πομόλου	5
1.4.2	Διαγράμματα για Οπτικοποίηση Μοντέλου Κίνησης	7
1.4.3	Ανάλυση Αποτελεσμάτων	7
2	Εργασία Ρομποτικής: Τμήμα Β	9
2.1	Περιγραφή του προβλήματος και απαιτήσεων	9
2.2	Προσέγγιση υλοποίησης σε MATLAB	10
2.2.1	Χρήσιμες εντολές από Robotics Toolbox που αξιοποιήθηκαν	10
2.3	Ενδελεχής ανάλυση και μεθοδολογία επίλυσης	10
2.3.1	Δομή του αλγορίθμου	11
2.3.2	Αριθμητική προσέγγιση μέσω διαφορικών σχέσεων	11
2.3.3	Προεπεξεργασία και αρχικές συνθήκες	11
2.3.4	Σημεία ενδιαφέροντος στον κώδικα MATLAB	12
2.3.5	Αποτελέσματα Κώδικα	12
2.3.6	Μαθηματική σύνοψη	12
2.4	Παρατηρήσεις και Αποτελέσματα	12
2.4.1	Αρχικές Συνθήκες Συστήματος	13
2.4.2	Στιγμιότυπα Κίνησης Ρομπότ, Πόρτας και Πομόλου	13
2.4.3	Διαγράμματα Τροχιών και Αξιολόγησης Κίνησης	15
2.4.4	Ανάλυση Αποτελεσμάτων	16

Κεφάλαιο 1

Εργασία Ρομποτικής: Τμήμα Α

1.1 Εισαγωγή και περιγραφή του προβλήματος

Στο πλαίσιο της παρούσας εργασίας ζητείται η προσομοίωση της κίνησης ανοίγματος μιας πόρτας με τη βοήθεια τροχιάς που περιγράφει την κίνηση του πόμολου (handle) . Η πόρτα βρίσκεται σε δωμάτιο και περιστρέφεται γύρω από κατακόρυφο άξονα στο σημείο μεντεσέ (hinge) , με αρχική θέση πλήρως κλειστή. Στην επιφάνεια της πόρτας υπάρχει τοποθετημένο ένα πλαίσιο $\{H\}$, το οποίο αντιστοιχεί στο πόμολο και περιγράφεται μέσω ενός ομογενούς μετασχηματισμού (homogeneous transformation) .

Η διαδικασία ανοίγματος της πόρτας περιλαμβάνει δύο διαδοχικά στάδια:

- Περιστροφή του πόμολου κατά -45° γύρω από τον τοπικό άξονα x_h με μηδενική αρχική και τελική ταχύτητα και επιτάχυνση.
- Περιστροφή της πόρτας κατά -30° γύρω από τον κατακόρυφο άξονα z_d , ενώ το πόμολο κινείται συγχρονισμένα ώστε να διατηρηθεί η σχετική του θέση.

Η συνολική διάρκεια της κίνησης είναι $T = 5 \text{ sec}$, ενώ στόχος είναι η δημιουργία ομαλής τροχιάς θέσης και προσανατολισμού για το πλαίσιο $\{H\}$. Για την υλοποίηση χρησιμοποιήθηκε το Robotics Toolbox του Peter Corke στο MATLAB , αξιοποιώντας τις συναρτήσεις `trpoly` και `trplot` για χρονική παρεμβολή και τρισδιάστατη απεικόνιση πλαισίων αντίστοιχα. Η έξοδος της διαδικασίας αποτελείται από δύο πίνακες τροχιάς, τον g_{oh} για το πόμολο (handle) \geq και τον g_{od} για την πόρτα (door). Αυτοί οι πίνακες περιγράφουν τη χρονικά εξαρτώμενη θέση και προσανατολισμό των αντίστοιχων αντικειμένων στο χώρο, και θα χρησιμοποιηθούν στο Τμήμα Β της εργασίας για την υλοποίηση και αξιολόγηση της αντιστροφικής κινηματικής.

1.2 Περιγραφή υλοποίησης σε MATLAB

Ο κώδικας ακολουθεί τα βήματα της περιγραφής και είναι χωρισμένος σε τμήματα:

- Ορισμός γεωμετρικών παραμέτρων πόρτας και πόμολου.
- Δημιουργία των γωνιακών προφίλ $\alpha(t)$ (στρέψη πόμολου) και $\beta(t)$ (γωνία ανοίγματος πόρτας).
- Υπολογισμός των ομογενών μετασχηματισμών $g_{od}(t)$ και $g_{oh}(t)$.

- Γραφική απεικόνιση σε κάθε χρονικό βήμα και αποθήκευση των τροχιών.

Ο κώδικας διασφαλίζει ρυθμό δειγματοληψίας $dt = 0.01$ sec (δηλαδή 100Hz), με συνολικά $n = 501$ δείγματα, διατηρώντας ακρίβεια και ομαλότητα της τροχιάς. Η χρήση `UnitQuaternion` επιτρέπει την απεικόνιση της προσανατολισμού του πλαισίου $\{H\}$ στον χρόνο.

1.3 Ενδεδειγμένη ανάλυση κώδικα και μεθοδολογίας

Το σύστημα που προσομοιώνεται αποτελείται από μία πόρτα (door) η οποία περιστρέφεται γύρω από κατακόρυφο άξονα (hinge) και έναν μηχανισμό πομόλου (handle) τοποθετημένο επάνω της, που στρέφεται γύρω από το δικό του τοπικό άξονα. Η χρονικά εξελισσόμενη κίνησή τους περιγράφεται μέσω ομογενών μετασχηματισμών του χώρου $SE(3)$.

1.3.1 Γεωμετρική περιγραφή και χρονικά διαστήματα

Οι βασικές παράμετροι του συστήματος είναι:

- $l = 1.0$ m: πλάτος πόρτας
- $l_o = 0.1$ m: απόσταση του πομόλου από τον μεντεσέ
- $h = 0.7$ m: ύψος τοποθέτησης του πομόλου
- $door_height = 2.0$ m: συνολικό ύψος πόρτας

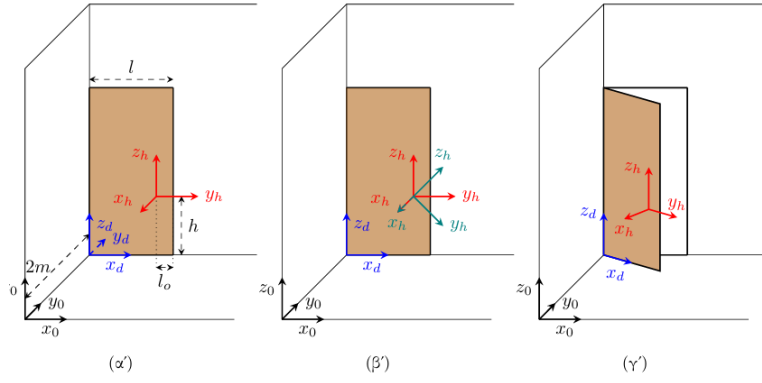
Η χρονική διάρκεια της προσομοίωσης είναι $T = T_1 + T_2 = 5$ s, όπου $T_1 = 2$ s αφιερώνεται στη στρέψη του πομόλου και $T_2 = 3$ s στο άνοιγμα της πόρτας καθώς ορίσαμε $N1 = 201$ και $N2 = 300$ καρέ για δειγματοληψία στα $dt = 0.01$ sec.

1.3.2 Κατασκευή τροχιών (Trajectory Planning)

Χρησιμοποιείται η συνάρτηση `troly` για την παραγωγή λείων, συνεχών τροχιών για τις γωνίες:

- $\alpha(t)$: περιγράφει τη στροφή του πομόλου γύρω από τον τοπικό άξονα x , αρχικά από 0° σε -45° κατά T_1 και επιστρέφει στο 0° στο T_2 .
- $\beta(t)$: δηλώνει τη στροφή της πόρτας γύρω από τον άξονα z , η οποία πραγματοποιείται μόνο στο διάστημα $[T_1, T]$ και φτάνει τα 30° .

Οι τροχιές είναι κατασκευασμένες ώστε να είναι χρονικά ομαλές και μη απότομες, κατάλληλες για εφαρμογή σε ρομποτικά συστήματα.



Σχήμα 1.1: Αναπαράσταση σχήματος πόρτας και πομόλου με τα πλαίσια $\{0\}$, $\{\Delta\}$, $\{H\}$ και τις γεωμετρικές παραμέτρους l , l_o , h

1.3.3 Κινηματική μοντελοποίηση

Το πρόβλημα επιλύεται κινηματικά μέσω συνδυασμού στοιχειωδών μετασχηματισμών του χώρου $SE(3)$. Οι σχετικές θέσεις της πόρτας και του πομόλου εκφράζονται αναλυτικά:

Πόρτα ως προς παγκόσμιο σύστημα: Η πόρτα θεωρείται προσαρτημένη σε σταθερό σημείο $(0, 2, 0)$, όπου και βρίσκεται ο άξονας περιστροφής (μεντεσές):

$$T_{D0}(t) = \text{transl}(0, 2, 0) \cdot \text{Rot}_z(\beta(t))$$

Πομόλο ως προς πόρτα: Η θέση του πομόλου καθορίζεται από μετατόπιση σε απόσταση $(l - l_o)$ κατά τον άξονα x και ύψος h , στροφή κατά -90° στον άξονα z για ευθυγράμμιση και εν συνεχεία στρέψη κατά $\alpha(t)$ στον άξονα x :

$$T_{HD}(t) = \text{transl}(l - l_o, 0, h) \cdot \text{Rot}_z\left(-\frac{\pi}{2}\right) \cdot \text{Rot}_x(\alpha(t))$$

Πομόλο ως προς παγκόσμιο σύστημα: Ο συνολικός μετασχηματισμός για το πομόλο υπολογίζεται ως:

$$T_{H0}(t) = T_{D0}(t) \cdot T_{HD}(t)$$

Ο πίνακας $T_{H0}(t) \in SE(3)$ περιέχει την πλήρη πληροφορία για τη θέση και τον προσανατολισμό του πομόλου σε κάθε χρονική στιγμή.

1.3.4 Ανάλυση τροχιάς

Από τον πίνακα $T_{H0}(t)$ εξάγονται:

- $p_h(t) = T_{H0}(t)[1 : 3, 4]$: τρισδιάστατη θέση πομόλου
- $q_h(t) = \text{UnitQuaternion}(T_{H0}(t))$: προσανατολισμός σε μορφή μοναδιαίου τεταρτογώνιου

Αυτά τα δεδομένα αποτελούν την έξοδο της προσομοίωσης και εισάγονται στην αντιστροφή κινηματικής.

1.3.5 Μαθηματική απεικόνιση του προβλήματος

Η κινηματική περιγραφή του συστήματος συνοψίζεται στην εξής εξίσωση:

$$T_{H0}(t) = \underbrace{\text{transl}(0, 2, 0)}_{\text{μεντεσές}} \cdot \underbrace{\text{Rot}_z(\beta(t))}_{\text{πόρτα}} \cdot \underbrace{\text{transl}(l - l_0, 0, h)}_{\text{θέση πομόλου}} \cdot \underbrace{\text{Rot}_z\left(-\frac{\pi}{2}\right)}_{\text{ευθυγράμμιση}} \cdot \underbrace{\text{Rot}_x(\alpha(t))}_{\text{στρέψη}}$$

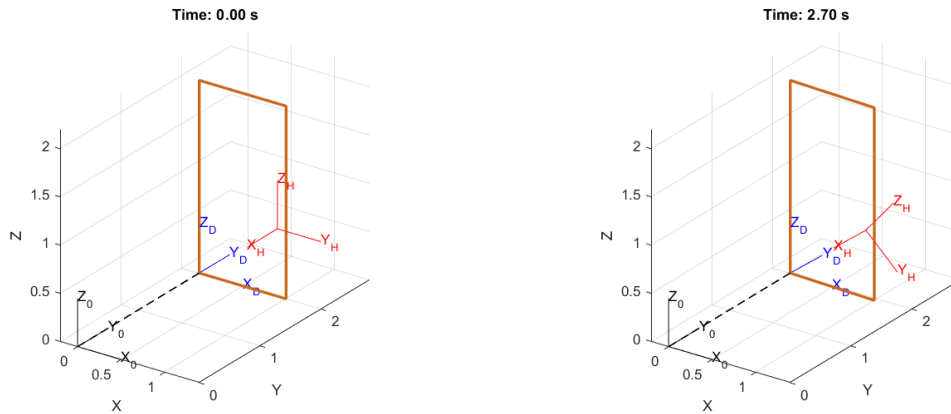
Η τροχιά $T_{H0}(t)$ ανήκει στον χώρο SE(3) και χρησιμοποιείται σε συνδιασμό με την $T_{D0}(t)$ ως αναφορά για τον σχεδιασμό της inverse kinematics στο Τμήμα Β της εργασίας.

1.4 Παρατηρήσεις και Αποτελέσματα

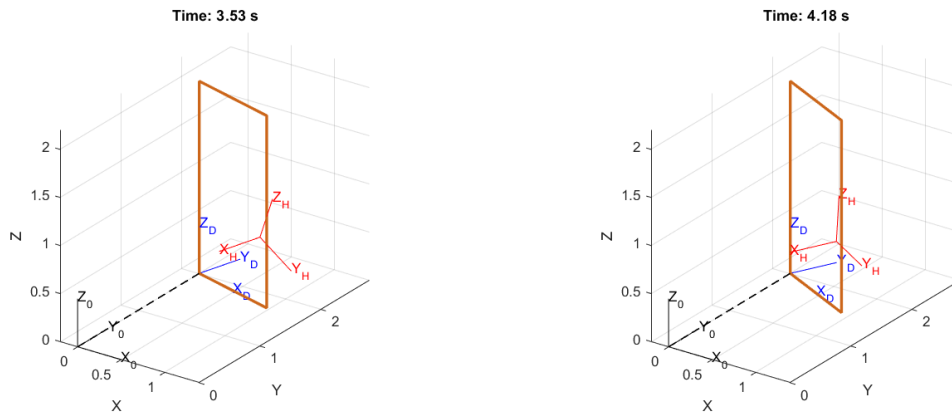
Στην παρούσα ενότητα παρουσιάζονται και αναλύονται τα αποτελέσματα της προσομοίωσης με βάση τα δεδομένα τροχιάς και τα αντίστοιχα διαγράμματα.

1.4.1 Στιγμιότυπα Κίνησης Πόρτας και Πομόλου

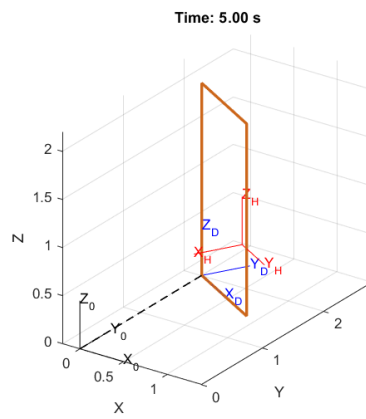
Τα παρακάτω στιγμιότυπα δείχνουν την εξέλιξη του συστήματος σε κρίσιμες χρονικές στιγμές:



Σχήμα 1.2: Αριστερά: $t = 0.00s$ (κλειστή πόρτα, αρχή στρέψης). Δεξιά: $t \approx 2.7s$ (τέλος στρέψης, κατά $0.7s$ αρχή ανοίγματος).

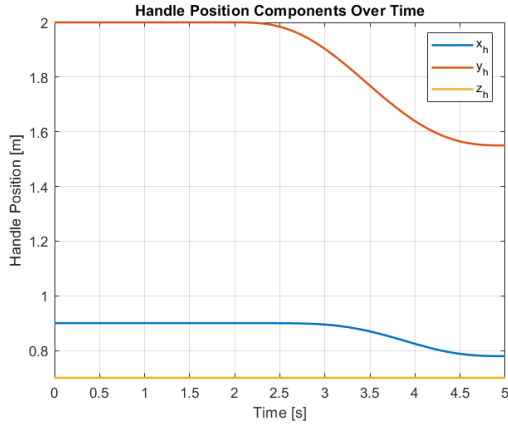


Σχήμα 1.3: Αριστερά: $t \approx 3.5s$. Δεξιά: $t \approx 4.2s$. Παρατηρείται σημαντική περιστροφή της πόρτας.

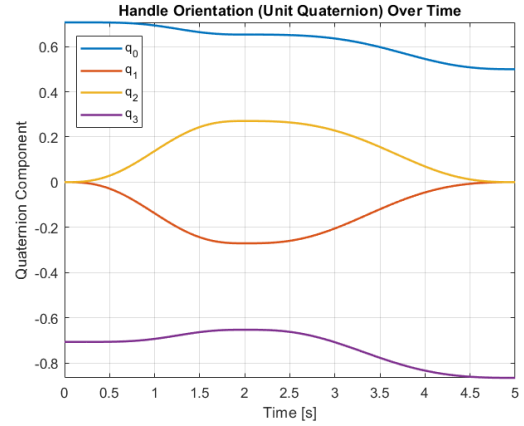


Σχήμα 1.4: Τελική φάση κίνησης με πόρτα πλήρως ανοιχτή $t = 5s$ και πομόλο επανελθόν στη θέση ισορροπίας.

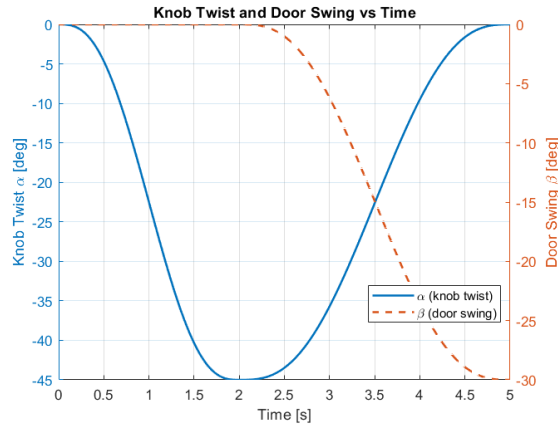
1.4.2 Διαγράμματα για Οπτικοποίηση Μοντέλου Κίνησης



Σχήμα 1.5: Θέση πομόλου x_h , y_h , z_h ως προς τον χρόνο.



Σχήμα 1.6: Συνιστώσες του μοναδιαίου τεταρτογωνίου q_0 έως q_3 .



Σχήμα 1.7: Γωνίες στρέψης πομόλου $\alpha(t)$ και περιστροφής πόρτας $\beta(t)$.

1.4.3 Ανάλυση Αποτελεσμάτων

Από την ανάλυση των παραπάνω διαγραμμάτων προκύπτουν τα εξής:

- **Θέση πομόλου:** Παρατηρείται ότι η z_h διατηρείται σταθερή καθ' όλη τη διάρκεια, επιβεβαιώνοντας πως δεν υπάρχει κατακόρυφη μετατόπιση. Η x_h εμφανίζει μικρή μεταβολή κατά την επαναφορά, ενώ η y_h μειώνεται σημαντικά μετά το πέρας της στρέψης, φανερώνοντας τη μετακίνηση της πόρτας προς τα πίσω.
- **Προσανατολισμός πομόλου:** Οι καμπύλες q_0 έως q_3 μεταβάλλονται ομαλά, χωρίς ασυνέχειες ή απότομες μεταβολές, επιβεβαιώνοντας σωστή κινηματική αναπαράσταση της περιστροφής του πομόλου μέσω μοναδιαίου τεταρτογωνίου.
- **Γωνιακά προφίλ $\alpha(t)$ και $\beta(t)$:** Το μοντέλο ακολουθεί πιστά το κινηματικό σενάριο. Πρώτα ολοκληρώνεται η στροφή του πομόλου (α), και μόνο τότε ξεκινά το

άνοιγμα της πόρτας (β), κάτι που φαίνεται από τη διακριτή χρονική διαδοχή των γωνιών.

- **Οπτικά στιγμιότυπα:** Απεικονίζουν επιτυχώς την κίνηση του συστήματος σε κρίσιμα χρονικά σημεία. Επιβεβαιώνεται ότι το σύστημα ανταποκρίνεται φυσιολογικά στο σενάριο, χωρίς αριθμητικά σφάλματα ή αντικρουόμενες μετακινήσεις.

Η συνολική συμπεριφορά της πόρτας και του πομόλου επιβεβαιώνει την εγκυρότητα του κινηματικού μοντέλου και τη σωστή υλοποίηση των τροχιών και φαίνεται πως ο κώδικας επιλύει ακριβώς το πρόβλημα που ζητήθηκε. Ο αντίστοιχος κώδικας βρίσκεται στο αρχείο *PartA.m*.

Κεφάλαιο 2

Εργασία Ρομποτικής: Τμήμα Β

2.1 Περιγραφή του προβλήματος και απαιτήσεων

Σε συνέχεια του Τμήματος Α, στο Τμήμα Β της εργασίας εξετάζεται η επίλυση του προβλήματος της αντιστροφικής κινηματικής για τον ρομποτικό βραχίονα UR10, προκειμένου να ακολουθήσει την προκαθορισμένη τροχιά του πόμολου (handle) που υπολογίστηκε στο Τμήμα Α.

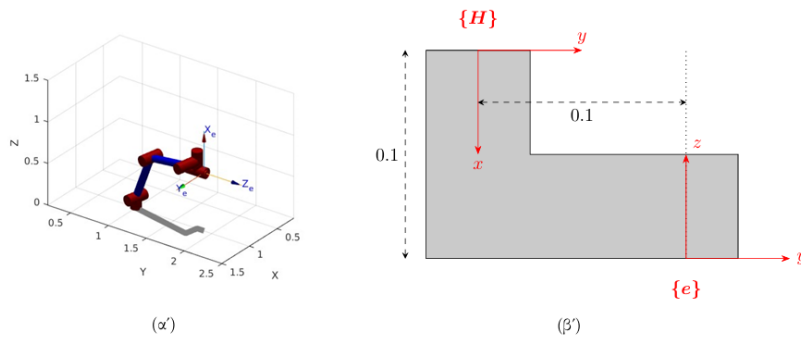
Ο βραχίονας UR10 διαθέτει 6 βαθμούς ελευθερίας και προσομοιώνεται με χρήση του Robotics Toolbox και ο κώδικας του κινηματικού αυτού μοντέλου δόθηκε ως βοήθημα στην υλοποίηση του Τμήματος Β και είναι *included* στο ίδιο *path* με τον κώδικα MATLAB του Τμήματος Β ως *ur10robot.m*. Στόχος είναι να υπολογιστεί για κάθε χρονική στιγμή το διάνυσμα αρθρώσεων $\mathbf{q}(t) \in \mathbb{R}^6$ το οποίο τοποθετεί το άκρο του ρομπότ (end-effector) έτσι ώστε να συμπίπτει με το πλαίσιο $\{H\}$ του πόμολου.

Η χρονική διάρκεια παραμένει $T = 5$ s και χρησιμοποιείται η ίδια δειγματοληψία με το Τμήμα Α: $n = 501$ δείγματα, με περίοδο $dt = 0.01$ s.

Το ρομπότ αρχικά τοποθετείται στη θέση:

$$\mathbf{q}_0 = [-1.7752, -1.1823, 0.9674, 0.2149, 1.3664, 1.5708]^T \text{ rad}$$

και η σχετική θέση του άκρου ως προς το πόμολο είναι σταθερή και περιγράφεται με τον μετασχηματισμό g_{he} (σταθερό hand-eye offset).



Σχήμα 2.1: (α') Απεικόνιση του ρομπότ UR10 στην αρχική του θέση με το πλαίσιο του άκρου $\{e\}$. (β') Σχηματική απεικόνιση του σταθερού μετασχηματισμού g_{he} μεταξύ του πλαισίου του πόμολου $\{H\}$ και του άκρου $\{e\}$.

2.2 Προσέγγιση υλοποίησης σε MATLAB

Ο κώδικας του Τμήματος B έχει την εξής δομή:

- Φόρτωση των τροχιών $g_H(t)$ και $g_D(t)$ από το Τμήμα A που έχουν αποθηκευτεί ως δεδομένα .mat μορφής.
- Υλοποίηση επανάληψης αντιστροφικής κινηματικής με χρονικό βήμα dt .
- Υπολογισμός τροχιάς θέσεων $q(t)$ και ταχυτήτων $\dot{q}(t)$ αρθρώσεων.
- Αποθήκευση και απεικόνιση της τελικής τροχιάς του ρομπότ.

2.2.1 Χρήσιμες εντολές από Robotics Toolbox που αξιοποιήθηκαν

- `ur10.fkine(q)`: Υπολογίζει τον ομογενή μετασχηματισμό του άκρου (forward kinematics) για τη θέση αρθρώσεων \mathbf{q} .
- `ur10.jacobe(q)`: Επιστρέφει την Ιακωβιανή πίνακα του άκρου (6x6) στο παγκόσμιο σύστημα.
- `tr2delta(A, B)`: Υπολογίζει εξαδιάστατο διάνυσμα σφάλματος μετασχηματισμού από A προς B στο χώρο $SE(3)$.
- `pinv(J)`: Υπολογίζει τη ψευδοαντίστροφη του πίνακα J , απαραίτητη για την επίλυση υπερπροσδιορισμένων γραμμικών συστημάτων.
- `transl(T)`: Εξάγει το διάνυσμα θέσης από ομογενή πίνακα μετασχηματισμού T .
- `UnitQuaternion(T)`: Μετατρέπει μετασχηματισμό $T \in SE(3)$ σε μοναδιαίο τεταρτογώνιο (quaternion).
- `trplot(T, ...)`: Γραφική απεικόνιση συστήματος συντεταγμένων του μετασχηματισμού T .
- `ur10.plot(q)`: Γραφική απεικόνιση του ρομπότ στη θέση q (offline rendering).
- `ur10.animate(q)`: Ενημερώνει την κίνηση του ρομπότ δυναμικά (real-time animation).

2.3 Ενδεδλεχής ανάλυση και μεθοδολογία επίλυσης

Το πρόβλημα του Τμήματος B όπως προαναφέρθηκε αφορά την υλοποίηση της αντιστροφικής κινηματικής για έναν ρομποτικό βραχίονα UR10, με σκοπό να εκτελέσει την επιθυμητή τροχιά του πομόλου όπως αυτή προδιαγράφηκε στο Τμήμα A. Η επίλυση υλοποιείται με τη χρήση της Ιακωβιανής του άκρου (Jacobian-based IK), με έμφαση στην αριθμητική συνέπεια και την σταθερότητα της λύσης.

2.3.1 Δομή του αλγορίθμου

Η αντιστροφή κινηματικής εφαρμόζεται για κάθε χρονική στιγμή t_k , με στόχο να υπολογιστεί μια ακολουθία γωνιών αρθρώσεων \mathbf{q}_k που να ικανοποιεί την επιθυμητή θέση και προσανατολισμό του άκρου. Για τον σκοπό αυτό χρησιμοποιούνται τα εξής:

- Η επιθυμητή τροχιά του πομόλου $g_H(t)$, προερχόμενη από το Τμήμα Α.
- Ο σταθερός μετασχηματισμός *hand-eye* g_{he} μεταξύ του άκρου του ρομπότ και του πομόλου.
- Η αριθμητική Ιακωβιανή J του άκρου στο παγκόσμιο πλαίσιο.

Ο επιθυμητός μετασχηματισμός του άκρου είναι:

$$g_e^{\text{des}}(t) = g_H(t) \cdot g_{he}$$

2.3.2 Αριθμητική προσέγγιση μέσω διαφορικών σχέσεων

Η αριθμητική διαδικασία βασίζεται στην εξίσωση του σφάλματος:

$$\Delta T_k = \text{tr2delta}(g_e^{\text{curr}}(t_k), g_e^{\text{des}}(t_k)) \in \mathbb{R}^6$$

η οποία αποδίδει ένα εξαδιάστατο σφάλμα (3 για θέση, 3 για προσανατολισμό).

Ακολουθώντας, εφαρμόζεται η σχέση:

$$\dot{\mathbf{q}}_k = J^\dagger(\mathbf{q}_k) \cdot \frac{\Delta T_k}{dt}$$

όπου J^\dagger είναι η ψευδοαντίστροφη της Ιακωβιανής.

Η ενημέρωση της θέσης αρθρώσεων γίνεται με μέθοδο Euler:

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \dot{\mathbf{q}}_k \cdot dt$$

2.3.3 Προεπεξεργασία και αρχικές συνθήκες

Το πρόγραμμα αρχικοποιείται με:

- q_0 : Αρχική θέση του UR10 ώστε να βρίσκεται κοντά στο πόμολο.
- Δειγματοληψία: $n = 501$ χρονικά σημεία με $dt = 0.01$ s και $T = 5$ s.

Οι μετασχηματισμοί $g_H(t)$ και $g_D(t)$ φορτώνονται από αρχεία που παράχθηκαν στο Τμήμα Α. Το g_{he} είναι σταθερός μετασχηματισμός καθορισμένος ως:

$$g_{he} = \begin{bmatrix} 0 & 0 & -1 & 0.1 \\ 0 & 1 & 0 & 0.1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.3.4 Σημεία ενδιαφέροντος στον κώδικα MATLAB

Η κάθε χρονική στιγμή περιλαμβάνει τα εξής βήματα (έχει προσθεθεί και ψευδοκώδικας για σαφήνεια):

1. Υπολογισμός του επιθυμητού μετασχηματισμού: $g_e_des = g_H(:, :, k) * g_he$
2. Υπολογισμός του παρόντος μετασχηματισμού: $g_e_curr = ur10.fkine(q)$
3. Υπολογισμός σφάλματος: $deltaT = tr2delta(g_e_curr.T, g_e_des)$
4. Υπολογισμός Ιακωβιανής: $J = ur10.jacobe(q)$
5. Λύση κινηματικής: $q_dot = pinv(J) * (deltaT / dt)$
6. Ενημέρωση αρθρώσεων: $q = q + q_dot * dt$

2.3.5 Αποτελέσματα Κώδικα

Η προσομοίωση παράγει τις παρακάτω τροχιές:

- $q(t)$: Θέσεις αρθρώσεων UR10 (6 διαστάσεων).
- $\dot{q}(t)$: Ταχύτητες αρθρώσεων.
- $p_e(t), q_e(t)$: Θέση και προσανατολισμός του άκρου.
- $p_{eh}(t), q_{eh}(t)$: Σχετική θέση και προσανατολισμός του άκρου ως προς το πόμολο.

Οι τροχιές είναι συνεχείς και ομαλές, επιτυγχάνοντας επιτυχή επίλυση του προβλήματος.

2.3.6 Μαθηματική σύνοψη

Ο συνολικός αλγόριθμος μπορεί να συνοψιστεί ως:

$$\begin{aligned} g_e^{des}(t) &= g_H(t) \cdot g_{he} \\ \Delta T &= tr2delta(g_e^{curr}, g_e^{des}) \\ \dot{q} &= J^\dagger \cdot \frac{\Delta T}{dt} \\ q(t + \Delta t) &= q(t) + \dot{q} \cdot dt \end{aligned}$$

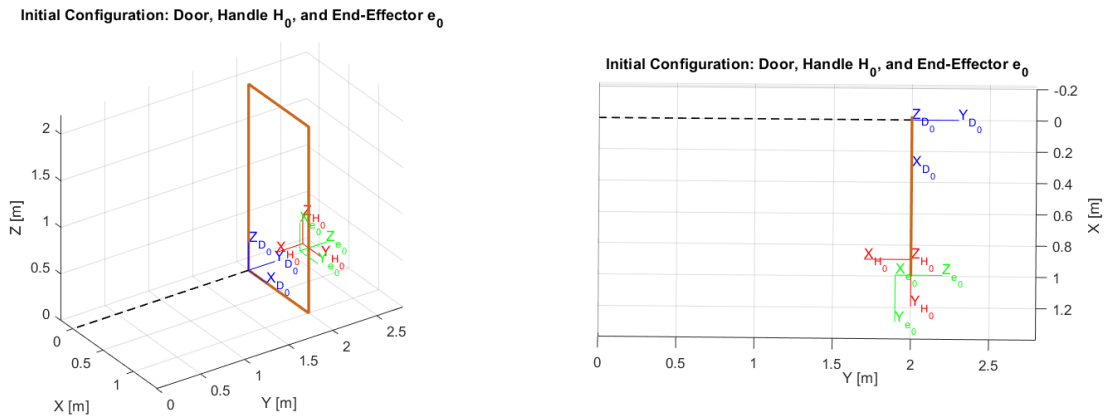
Η επίλυση ανήκει στον χώρο των μεθόδων αντιστροφής κινηματικής με χρήση διαφορικών σχέσεων και προσαρμοστικών τεχνικών σε πραγματικό χρόνο, και βασίζεται αρκετά στη δυναμική του Robotics Toolbox.

2.4 Παρατηρήσεις και Αποτελέσματα

Στην παρούσα ενότητα παρουσιάζονται και αναλύονται τα αποτελέσματα της προσομοίωσης για το Τμήμα Β, όπου έγινε χρήση αντιστροφικής κινηματικής για τον βραχίονα UR10 ώστε να ακολουθήσει το πλαίσιο του πομόλου $\{H\}$ σε όλη τη διάρκεια της τροχιάς.

2.4.1 Αρχικές Συνθήκες Συστήματος

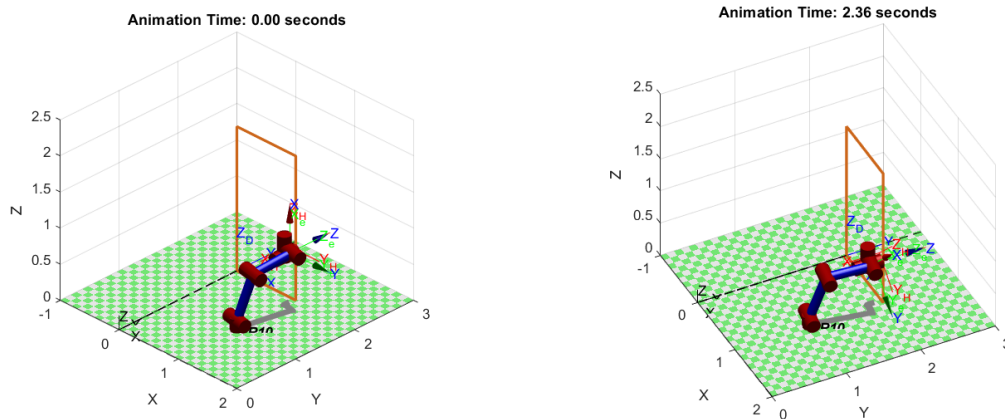
Ακολουθούν δύο όψεις της αρχικής θέσης του συστήματος:



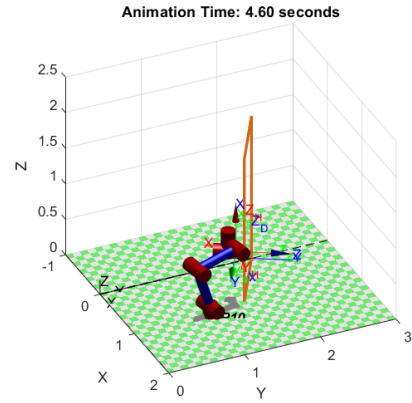
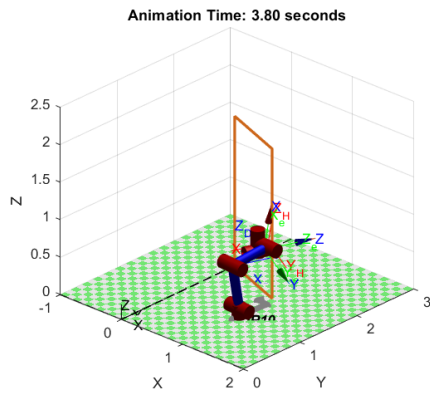
Σχήμα 2.2: Απεικόνιση αρχικής θέσης του ρομπότ, της πόρτας και του πομόλου. Αριστερά: τρισδιάστατη προβολή, δεξιά: προβολή στο επίπεδο XY.

2.4.2 Στιγμιότυπα Κίνησης Ρομπότ, Πόρτας και Πομόλου

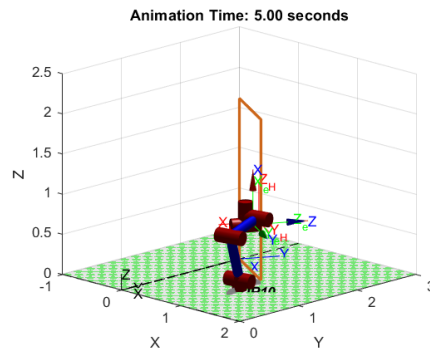
Παρουσιάζονται χαρακτηριστικά στιγμιότυπα της εξέλιξης του ρομπότ και του συστήματος:



Σχήμα 2.3: Αριστερά: $t = 0.00s$, αρχική θέση UR10. Δεξιά: $t \approx 2.36s$, ολοκλήρωση στρέψης πομόλου.

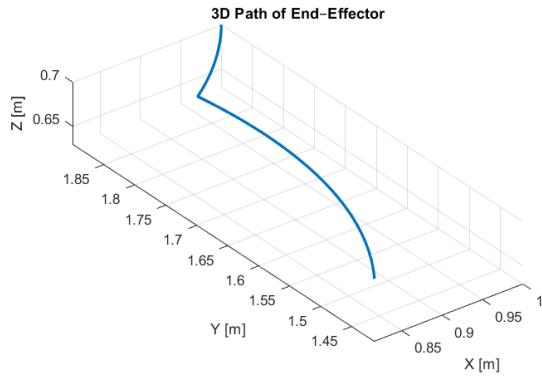


Σχήμα 2.4: Αριστερά: $t \approx 3.80s$, εν εξελίξει άνοιγμα πόρτας. Δεξιά: $t \approx 4.60s$, κοντά στο τελικό άνοιγμα.

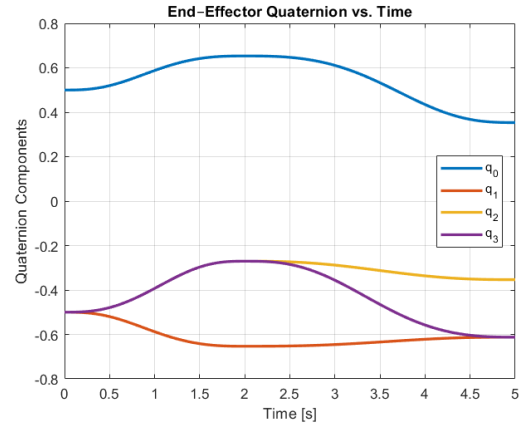


Σχήμα 2.5: Τελική φάση: $t = 5.00s$ με πλήρως ανοιχτή πόρτα και συνεχή επαφή του UR10 με το πλαίσιο $\{H\}$.

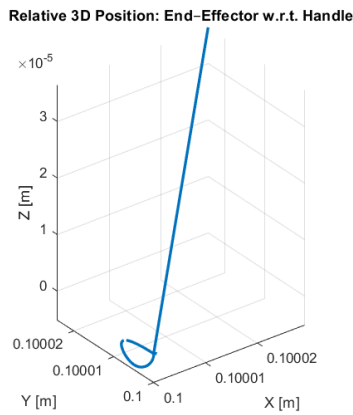
2.4.3 Διαγράμματα Τροχιών και Αξιολόγησης Κίνησης



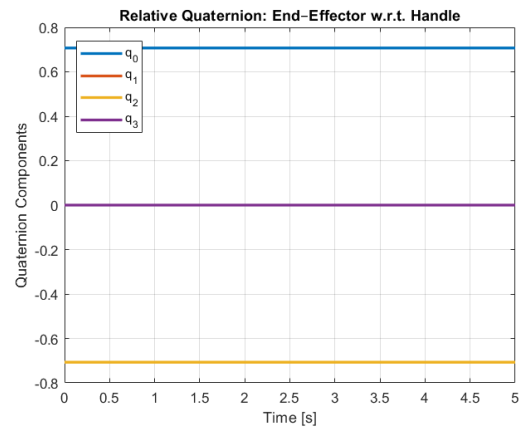
Σχήμα 2.6: Τρισδιάστατη τροχιά του άκρου (end-effector) κατά την εκτέλεση της κίνησης. Παρατηρείται συνεχής και λεία μετάπτωση.



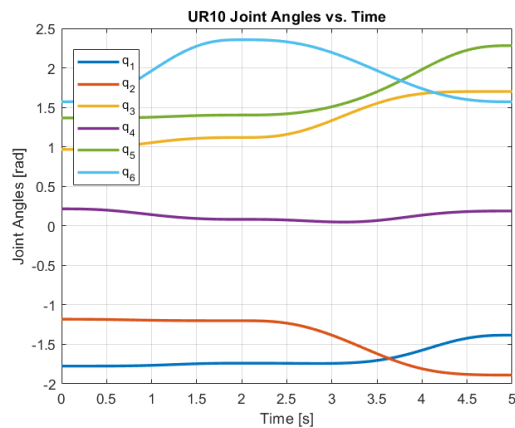
Σχήμα 2.7: Συνιστώσες του μοναδιαίου τεταρτογωνίου προσανατολισμού του άκρου. Η ομαλή μεταβολή επιβεβαιώνει σωστή περιστροφή.



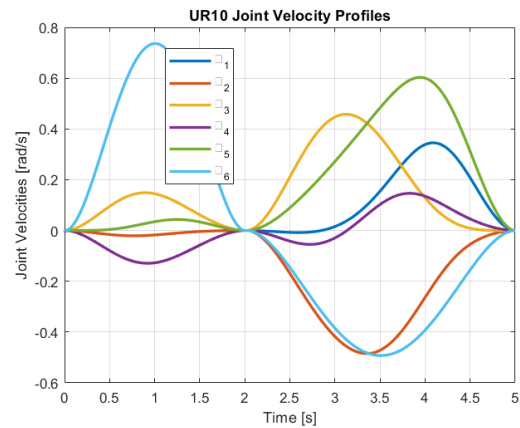
Σχήμα 2.8: Σχετική θέση του άκρου ως προς το πλαίσιο του πομόλου $\{H\}$. Η τροχιά κυμαίνεται σε τάξη 10^{-5} , επιβεβαιώνοντας ακρίβεια.



Σχήμα 2.9: Σχετικός προσανατολισμός: ο τεταρτογωνικός προσανατολισμός παραμένει πρακτικά σταθερός, αποδεικνύοντας σταθερότητα σύνδεσης.



Σχήμα 2.10: Χρονική εξέλιξη των γωνιών αρθρώσεων του UR10. Όλες οι τροχιές είναι λείες και συνεχείς, με ρεαλιστικά προφίλ.



Σχήμα 2.11: Ταχύτητες αρθρώσεων: οι μεταβολές παραμένουν ομαλές, χωρίς απότομες αλλαγές που θα έθεταν σε κίνδυνο τον ελεγκτή.

2.4.4 Ανάλυση Αποτελεσμάτων

Από την παρατήρηση των διαγραμμάτων και στιγμιοτύπων, προκύπτουν οι ακόλουθες παρατηρήσεις:

- **Διατήρηση επαφής με το πλαίσιο {H}:** Το γράφημα της σχετικής θέσης του άκρου δείχνει μεταβολές στην τάξη του 10^{-5} , δηλαδή κάτω από χιλιοστά, γεγονός που αποδεικνύει ότι το άκρο ακολουθεί με απόλυτη ακρίβεια την τροχιά του πομόλου.
- **Σταθερότητα στον προσανατολισμό:** Η διατήρηση σταθερού τεταρτογωνίου στον σχετικό προσανατολισμό του άκρου ως προς το {H} επιβεβαιώνει την επιτυχή σταθεροποίηση του *hand-eye* offset σε όλη τη διάρκεια της κίνησης.
- **Ομαλότητα τροχιών:** Οι γωνίες αρθρώσεων και οι ταχύτητες παρουσιάζουν φυσική και συνεχή εξέλιξη, όπως απαιτείται για ρομποτικά συστήματα που αλληλεπιδρούν με περιβάλλον.
- **Οπτική τεκμηρίωση κίνησης:** Από τα χρονικά στιγμιότυπα του ρομπότ επιβεβαιώνεται ότι η τροχιά ακολουθεί επακριβώς την πορεία του πομόλου, ενώ το ρομπότ τροποποιεί στάση χωρίς απώλεια σταθερότητας.
- **Πιστότητα αντιστροφικής κινηματικής:** Η λύση του προβλήματος μέσω ψευδοαντίστροφης Ιακωβιανής αποδεικνύεται επαρκής, προσφέροντας ακριβή παρακολούθηση χωρίς αριθμητικά φαινόμενα όπως ιδιομορφίες ή ασυνέχειες.

Συμπερασματικά, η αριθμητική προσέγγιση αντιστροφικής κινηματικής υλοποιείται επιτυχώς, με τη χρήση εργαλείων του Robotics Toolbox και τον κώδικα του αρχείου *PartB.m* ενώ οι αρχικές συνθήκες του συστήματος στατικά προκύπτουν στο αρχείο *debug.m*.

Βιβλιογραφία

- [1] https://elearning.auth.gr/pluginfile.php/526639/mod_resource/content/1/ur10robot.m
- [2] <https://petercorke.com/toolboxes/robotics-toolbox/>