

## Ρομποτικά Συστήματα 2

**Τίτλος εργασίας:** Έλεγχος συστήματος τετράποδου ρομπότ μέσω γραφικής διεπαφής API

**Ονοματεπώνυμο:** Πρίντζιος Λάμπρος

**Έτος:** 5<sup>ο</sup>

**Υπεύθυνος καθηγητής:** Χατζηλυγερούδης Κωνσταντίνος

### Περιεχόμενα:

Εισαγωγή / Σκοπός και προδιαγραφές της εργασίας.....	2
Περιγραφή του περιβάλλοντος και των λειτουργιών της γραφικής διεπαφής .....	3
Έλεγχος του συστήματος του τετράποδου ρομπότ.....	10
Αποτελέσματα της προσομοίωσης .....	17

## Εισαγωγή / Σκοπός και προδιαγραφές της εργασίας

Στη συγκεκριμένη άσκηση απαιτείται η δημιουργία εφικτών (ως προς τη δυναμική) μετακινήσεων ενός τετράποδου ρομπότ, το οποίο μοντελοποιείται με τη χρήση του SRBD (single rigid body dynamics) μοντέλου. Χρειάζεται επομένως η δημιουργία ενός προβλήματος βελτιστοποίησης, η επίλυση του οποίου θα δίνει τις βέλτιστες αυτές τροχιές. Κύριο κομμάτι της εργασίας αποτελεί η υλοποίηση μιας γραφικής διεπαφής API, μέσω της οποίας ο χρήστης θα ελέγχει το τετράποδο ρομποτικό σύστημα. Έχουν οριστεί οι ακόλουθες βασικές προδιαγραφές που απαιτείται να πληροί η εφαρμογή:

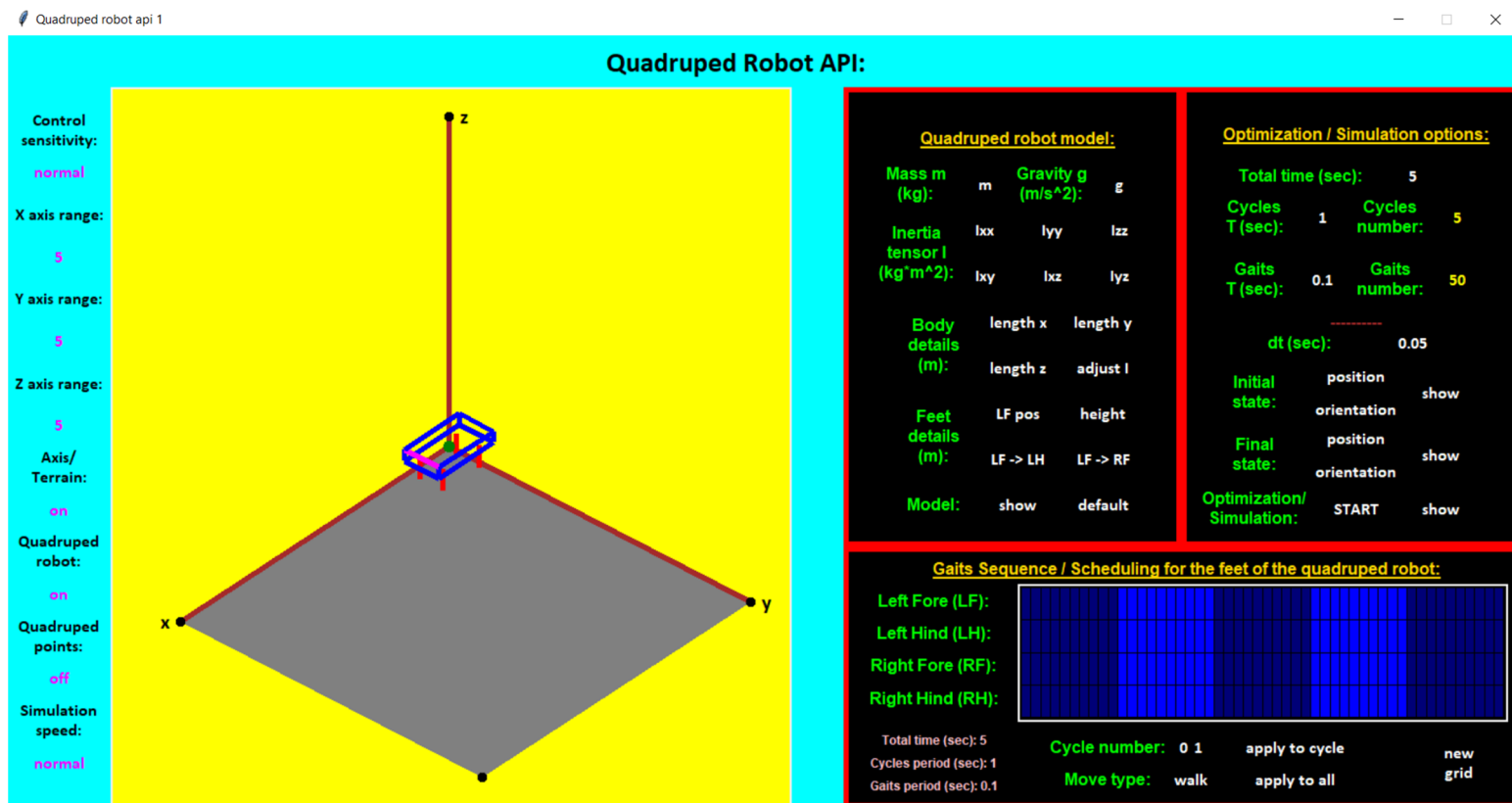
- Δυνατότητα αλλαγής της αρχικής και της τελικής/επιθυμητής κατάστασης του τετράποδου ρομπότ (ορίζοντας και τη θέση και τον προσανατολισμό σε κάθε περίπτωση).
- Δυνατότητα επιλογής των φάσεων του κάθε ποδιού. Αυτό περιλαμβάνει τον καθορισμό του αριθμού των φάσεων, της χρονικής διάρκειας εκτέλεσης της κάθε φάσης, της κατάστασης του κάθε ποδιού στην εκάστοτε φάση (δηλαδή αν βρίσκεται σε επαφή/contact με το έδαφος ή σε αιώρηση/swing) κ.ά..
- Κατασκευή προκαθορισμένων κύκλων κίνησης του τετράποδου ρομπότ (π.χ. κύκλος βαδίσματος, αναπήδησης, τρεξίματος κ.ά.).
- Μεταβολή των τεχνικών ιδιοτήτων του τετράποδου, όπως είναι η μάζα του, ο πίνακας αδράνειας, τα όρια τοποθέτησης των ποδιών κ.ά..
- Οπτικοποίηση της κίνησης του τετράποδου ρομπότ στον 3D χώρο. Για τον σκοπό αυτό χρησιμοποιώ τη native βιβλιοθήκη tkinter της python.
- Δημιουργία ρεαλιστικής επαφής του κάθε ποδιού με το έδαφος και αποφυγή ολισθήσεων. Δηλαδή όλες οι επαφές απαιτείται να βρίσκονται σε σταθερή κατάσταση (fixed mode), εκτός φυσικά των περιπτώσεων που το πόδι δεν ακουμπά στο έδαφος (ελεύθερη κατάσταση, free mode).

Ορισμένα αρχικά τεχνικά χαρακτηριστικά του τετράποδου ρομπότ δίνονται στη συνέχεια:

- Η μάζα (του σώματος, τα πόδια θεωρούνται αβαρή):  $m = 30.4213964625 \text{ kg}$
- Ο πίνακας αδράνειας (σε  $\text{kg} \cdot \text{m}^2$ ):  $I = \begin{bmatrix} 0.88201174 & -0.00137526 & -0.00062895 \\ -0.00137526 & 1.85452968 & -0.00018922 \\ -0.00062895 & -0.00018922 & 1.97309185 \end{bmatrix}$
- Οι ονομαστικές θέσεις των τεσσάρων ποδιών (αυτές δείχνουν τις σχετικές θέσεις των ποδιών μεταξύ τους, εκφράζονται σε  $m$ ):
  - Αριστερό μπροστά πόδι:  $[x \ y \ z]^T = [0.34 \ 0.19 \ -0.42]^T$
  - Αριστερό πίσω πόδι:  $[x \ y \ z]^T = [-0.34 \ 0.19 \ -0.42]^T$
  - Δεξί μπροστά πόδι:  $[x \ y \ z]^T = [0.34 \ -0.19 \ -0.42]^T$
  - Δεξί πίσω πόδι:  $[x \ y \ z]^T = [-0.34 \ -0.19 \ -0.42]^T$

Όπως θα εξηγηθεί και παρακάτω, έχω δημιουργήσει (χωρίς βλάβη της γενικότητας) ένα τερέν που εκτείνεται μονάχα στο 1<sup>ο</sup> τεταρτημόριο του  $xy$  επιπέδου. Το επίπεδο αυτό θα αποτελέσει και το πεδίο δράσης και κίνησης του ρομπότ. Η επιτάχυνση της βαρύτητας έχει πάντα κατεύθυνση προς τα αρνητικά του άξονα  $z$ . Επιπλέον, από τον πίνακα της αδράνειας μπορούμε να διαπιστώσουμε ότι το σώμα του τετράποδου δεν είναι ακριβώς ορθογώνιο παραλληλεπίπεδο, αφού τα γινόμενα αδράνειας  $I_{xy}, I_{xz}, I_{yz}$  είναι διάφορα του μηδενός. Έχουν βεβαίως πολύ μικρές τιμές, κοντά στο μηδέν, συνεπώς θεωρώ στη σχεδίαση ότι το σχήμα του σώματος προσεγγίζει αυτό ενός ορθογωνίου παραλληλεπίπεδου. Θεωρώ ακόμα ότι η κατανομή της μάζας του σώματος είναι ομοιόμορφη, άρα το κέντρο μάζας βρίσκεται στο γεωμετρικό κέντρο του σχήματος, δηλαδή στο σημείο τομής των τριών επιπέδων συμμετρίας του.

# Περιγραφή του περιβάλλοντος και των λειτουργιών της γραφικής διεπαφής



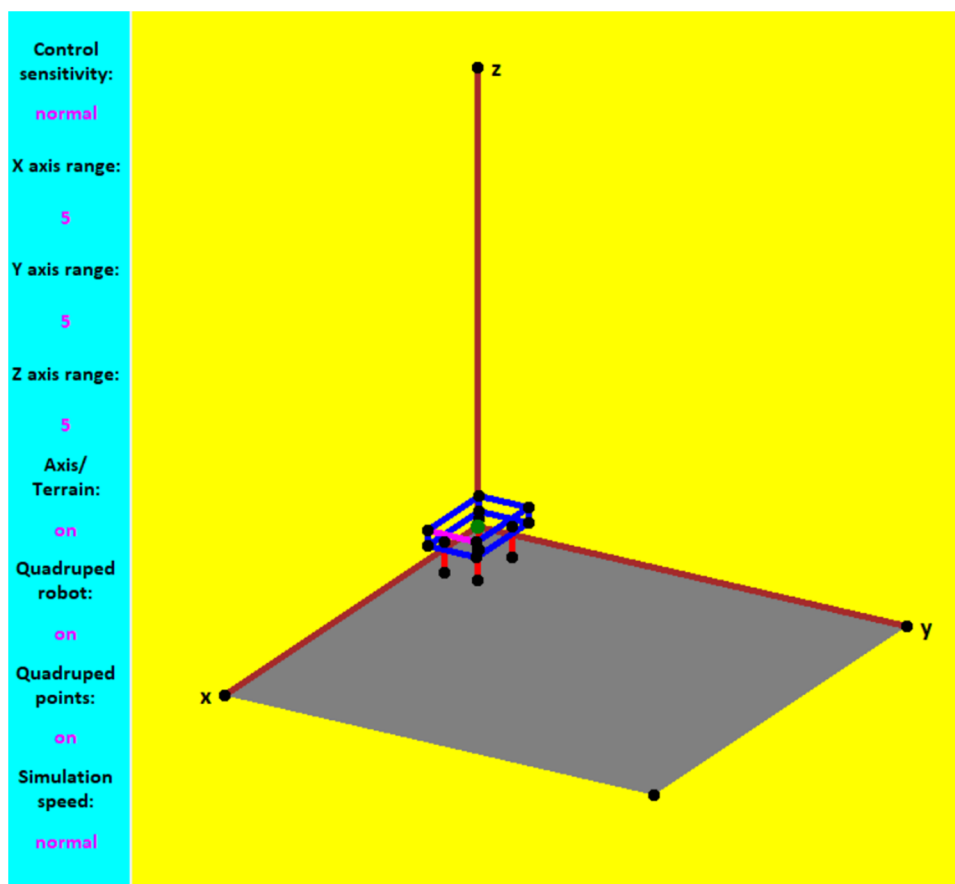
Εικόνα 1

Η γραφική διεπαφή (εικόνα 1) αποτελείται από δύο βασικές περιοχές, την περιοχή όπου οπτικοποιείται το συνολικό σύστημα (δηλαδή ο χώρος εργασίας μαζί με το τετράποδο ρομπότ), η οποία καταλαμβάνει το αριστερό μισό της εφαρμογής, και την περιοχή όπου βρίσκονται τα διάφορα μενού με επιλογές για τον χρήστη, η οποία καταλαμβάνει το δεξί μισό της εφαρμογής. Η περιοχή της οπτικοποίησης διακρίνεται σε δύο υποπεριοχές, με τη μία να περιέχει διάφορες επιλογές σχετικές με τον έλεγχο και την εμφάνιση της οπτικοποίησης και την άλλη να περιλαμβάνει έναν καμβά όπου προβάλλεται ο τριδιάστατος χώρος στον οποίον ενεργεί το τετράποδο. Η περιοχή των μενούν διακρίνεται σε τρεις υποπεριοχές, ή αλλιώς τρία υπομενού, με το πρώτο να παρέχει τη δυνατότητα αλλαγής των τεχνικών ιδιοτήτων και της στερεομετρίας του τετράποδου ρομπότ, το δεύτερο να δίνει επιλογές για τις παραμέτρους της προσομοίωσης και το τρίτο να ορίζει τα χαρακτηριστικά της κίνησης του τετράποδου κατά τη διάρκεια της προσομοίωσης. Περισσότερες λεπτομέρειες για όλα τα παραπάνω ακολουθούν αμέσως μετά:

## 1) Περιοχή της οπτικοποίησης:

Αρχικά, παρατηρούμε τη μεγάλη κίτρινη περιοχή που καταλαμβάνει το σημαντικά μεγαλύτερο μέρος της περιοχής και σε αυτήν προβάλλεται ο τριδιάστατος χώρος με το τετράποδο ρομπότ (εικόνα 2). Η σχεδίαση του τετράποδου γίνεται με πολύ απλά γεωμετρικά σχήματα, πιο συγκεκριμένα με σημεία και ευθύγραμμα τμήματα που συνδέουν τα κατάλληλα σημεία, ώστε να απεικονιστεί το σώμα του ρομπότ ως ένα ορθογώνιο παραλληλεπίπεδο και τα πόδια του ως ευθείες γραμμές. Οι άξονες  $x, y, z$  που παριστάνουν το παγκόσμιο σύστημα συντεταγμένων σχεδιάζονται με παρόμοια λογική, με ένα σημείο

να είναι η αρχή των αξόνων  $O$  και άλλα τρία σημεία να σηματοδοτούν το πέρας της σχεδίασης του κάθε θετικού ημιάξονα, δηλαδή απεικονίζω μονάχα το πρώτο οκτημόριο του τριδιάστατου χώρου. Με την εκκίνηση του προγράμματος οι άξονες σχεδιάζονται έτσι ώστε ο θετικός ημιάξονας  $z$  να κατευθύνεται προς το πάνω μέρος της οθόνης, ο θετικός ημιάξονας  $y$  προς το δεξί μέρος και ο θετικός ημιάξονας  $x$  προς τον χρήστη. Το τερέν (δάπεδο) στο οποίο κινείται το ρομπότ σχεδιάζεται με ένα ορθογώνιο παραλληλόγραμμο και εκτείνεται στο πρώτο τεταρτημόριο του  $xy$  επιπέδου. Γενικά θα αναφέρομαι συνολικά στο σύστημα των αξόνων του περιβάλλοντος του ρομπότ και στο τετράποδο ρομπότ, ως χώρο εργασίας. Ο χρήστης έχει τη δυνατότητα να μεταφέρει ολόκληρο τον χώρο εργασίας πατώντας το δεξί κλικ και σέρνοντας το ποντίκι. Ακόμα, μπορεί να περιστρέψει τον χώρο εργασίας πατώντας το αριστερό κλικ και σέρνοντας τον κέρσορα δεξιά – αριστερά για περιστροφή γύρω από τον αρχικό άξονα  $z$  και πάνω – κάτω για περιστροφή γύρω από τον άξονα  $y$ . Επίσης, με την περιστροφή της ροδέλας γίνεται μεγέθυνση και σμίκρυνση της οπτικοποίησης. Τέλος, ο χρήστης μπορεί να φέρει τον χώρο εργασίας στην αρχική του κατάσταση στον καμβά με δύο διαδοχικά δεξιά κλικ. Οι ενέργειες μεταφοράς, περιστροφής και κλιμάκωσης πραγματοποιούνται με τη χρήση ομογενών πινάκων  $4 \times 4$  που δρουν στο σύνολο των σημείων του χώρου εργασίας, όπως φαίνεται στις αντίστοιχες συναρτήσεις του κώδικα. Τέλος, ο χρήστης μπορεί να μάθει τη θέση του κάθε σημείου του χώρου εργασίας ανά πάσα στιγμή, απλά πηγαίνοντας με το ποντίκι πάνω από το επιθυμητό σημείο και διαβάζοντας τις συντεταγμένες του που εμφανίζονται στο κάτω μέρος του καμβά. Ένα μειονέκτημα της οπτικοποίησης που έχω δημιουργήσει είναι ότι δεν υπάρχει γνώση για το βάθος στο οποίο βρίσκεται το κάθε σχήμα (δεν υπάρχει π.χ. καταχωρητής όπου να αποθηκεύεται το βάθος του κάθε αντικείμενου, όπως είναι η συνήθης πρακτική στα γραφικά), με αποτέλεσμα κάποιες φορές να ζωγραφίζονται τα αντικείμενα με τη λάθος σειρά. Αυτό βέβαια επηρεάζει ελάχιστα την προοπτική αντίληψη της σχεδίασης, λόγω και του μικρού αριθμού των αντικειμένων.



Εικόνα 2

Συνεχίζοντας στην περιοχή της οπτικοποίησης, στο αριστερό μέρος υπάρχει μια λεπτή γαλάζια λωρίδα όπου βρίσκονται διάφορα κουμπιά, η λειτουργία των οποίων περιγράφεται ακολούθως:

- **Control sensitivity** (μέγεθος ευαισθησίας του ελέγχου): Δίνονται στον χρήστη οι επιλογές “low”, “normal” και “high”, κατά αύξουσα σειρά ευαισθησίας. Αυτές αναφέρονται στο πόσο έντονη ή ήπια είναι η ενέργεια που προκαλεί ο χρήστης στον χώρο εργασίας (μεταφορά, περιστροφή ή κλιμάκωση), για την ίδια κίνηση του ποντικιού.
- **X axis range, Y axis range, Z axis range** (τα εύρη των αξόνων X, Y και Z): Εδώ ρυθμίζεται το μήκος σχεδίασης για κάθε θετικό ημιάξονα, με ελάχιστη τιμή το 5 και μέγιστη τιμή το 100 σε μέτρα.
- **Axis/Terrain** (άξονες/δάπεδο): Ορίζει την ορατότητα (επιλογή “on”) ή μη (επιλογή “off”) του παγκόσμιου συστήματος συντεταγμένων των αξόνων και του τερέν/δαπέδου.
- **Quadruped robot** (τετράποδο ρομπότ): Ορίζει την ορατότητα (επιλογή “on”) ή μη (επιλογή “off”) του τετράποδου ρομπότ (σημεία και ακμές).
- **Quadruped points** (σημεία του τετράποδου ρομπότ): Ορίζει την ορατότητα (επιλογή “on”) ή μη (επιλογή “off”) των σημείων του τετράποδου ρομπότ, με τις ακμές να είναι φανερές σε κάθε περίπτωση.
- **Simulation speed** (ταχύτητα εκτέλεσης της προσομοίωσης): Δίνει τη δυνατότητα στον χρήστη να επιλέξει την ταχύτητα αναπαραγωγής της υπολογισμένης προσομοιωμένης κίνησης του τετράποδου. Για αναπαραγωγή σε πραγματικό χρόνο υπάρχει η επιλογή “normal”, ενώ για αργές ταχύτητες υφίστανται οι επιλογές “slow” και “very slow” και για γρήγορες ταχύτητες οι επιλογές “fast” και “very fast”.

## 2) Περιοχή των μενού:

Όπως προαναφέρθηκε, η περιοχή με τα μενού διακρίνεται σε τρία βασικά υπομενού που περιγράφονται και αναλύονται στη συνέχεια.

a) **Quadruped model** (μοντέλο του τετράποδου ρομπότ): Στο συγκεκριμένο υπομενού (εικόνα 3) προσδιορίζονται τα τεχνικά και στερεομετρικά χαρακτηριστικά του τετράποδου. Δίνονται στον χρήστη οι εξής επιλογές (πηγαίνοντας με τη σειρά από πάνω προς τα κάτω):

- **Mass m in kg** (μάζα): Εδώ ρυθμίζεται η μάζα  $m$  του τετράποδου (υποθέτουμε ότι όλη η μάζα συγκεντρώνεται στο σώμα, με αποτέλεσμα τα πόδια να θεωρούνται αβαρή).
- **Gravity g in m/s<sup>2</sup>** (επιτάχυνση της βαρύτητας): Με αυτήν την επιλογή καθορίζεται το μέγεθος της επιτάχυνσης της βαρύτητας (προεπιλεγμένη τιμή είναι το πεδίο βαρύτητας της γης  $g = 9.81 \frac{m}{s^2}$ ).
- **Inertia tensor I, in kg\*m<sup>2</sup>** (Πίνακας αδράνειας): Εδώ ρυθμίζονται τα στοιχεία του  $3 \times 3$

πίνακα αδράνειας  $I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}$  του σώματος

του ρομπότ. Απαιτείται να καθοριστούν μονάχα 6 (τα  $I_{xx}, I_{yy}, I_{zz}, I_{xy}, I_{xz}, I_{yz}$ ) από τα 9 συνολικά στοιχεία του πίνακα, διότι ο  $I$  είναι συμμετρικός.

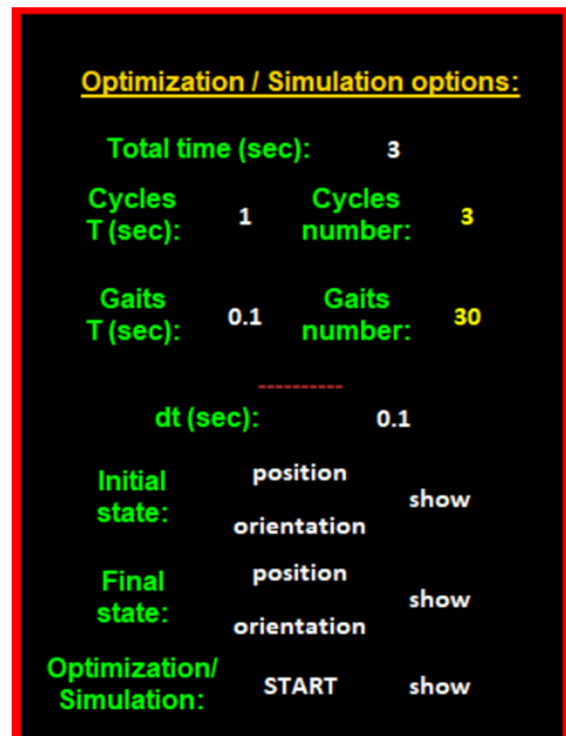
<u>Quadruped robot model:</u>			
<b>Mass m (kg):</b>	m	<b>Gravity g (m/s<sup>2</sup>):</b>	g
<b>Inertia tensor I (kg*m<sup>2</sup>):</b>	Ixx	Iyy	Izz
	Ixy	Ixz	Iyz
<b>Body details (m):</b>	length x	length y	
	length z	adjust I	
<b>Feet details (m):</b>	LF pos	height	
	LF -> LH	LF -> RF	
<b>Model:</b>	show	default	

Εικόνα 3

- **Body details in m** (σχεδιαστικές λεπτομέρειες για το σώμα): Εδώ επιλέγονται το μήκος ("length x"), το πλάτος ("length y") και το ύψος ("length z") του σώματος (ορθογώνιο παραλληλεπίπεδο στο σχήμα) του τετράποδου σε μέτρα. Υφίσταται ακόμα η επιλογή "adjust l" που προσαρμόζει τον πίνακα αδράνειας βάσει του σχήματος, της μάζας και των διαστάσεων του σώματος του ρομπότ (ειδικότερα, υπολογίζεται ο θεωρητικά προβλεπόμενος πίνακας αδράνειας για ένα ορθογώνιο παραλληλεπίπεδο σώμα). Έτσι, το μοντέλο γίνεται πιο ρεαλιστικό και λαμβάνεται υπόψη η φυσική σημασία των τιμών, οι οποίες ανταποκρίνονται καλύτερα στο σχεδιαστικό μοντέλο.
- **Feet details in m** (τεχνικές λεπτομέρειες για τα πόδια): Ορίζονται η χωρική θέση του αριστερού μπροστά ποδιού ("LF pos"), το κοινό ύψος των τεσσάρων ποδιών ("height"), η απόσταση μεταξύ του αριστερού μπροστά και του αριστερού πίσω ποδιού ("LF -> LH") και η απόσταση μεταξύ του αριστερού μπροστά και του δεξιού μπροστά ποδιού ("LF -> RF"). Οι παράμετροι αυτές αρκούν για να προσδιοριστούν πλήρως οι αρχικές θέσεις και των τεσσάρων ποδιών, θεωρώντας ότι στη σχεδιαστική φάση, πριν την εκτέλεση της προσομοίωσης, το ρομπότ βρίσκεται σε όρθια στάση με τα κατώτερα σημεία των ποδιών που μπορούν να βρεθούν σε επαφή με το έδαφος να ανήκουν σε ένα επίπεδο παράλληλο με αυτό του δαπέδου.
- **Model** (γενικές επιλογές αναφορικά με το μοντέλο): Πρόκειται για την τελευταία επιλογή, που περιλαμβάνει δύο λειτουργίες, τη "show", η οποία δείχνει στον χρήστη τα τεχνικά και σχεδιαστικά χαρακτηριστικά του τετράποδου που ισχύουν τη δεδομένη στιγμή και τη "default", που επαναφέρει το μοντέλο στις αρχικές του προκαθορισμένες ρυθμίσεις που περιεγράφηκαν στην εισαγωγή.

b) **Optimization / Simulation options** (επιλογές για τη βελτιστοποίηση / προσομοίωση): Εδώ ο χρήστης ρυθμίζει τις τιμές των παραμέτρων που έχουν άμεση σχέση με την εύρεση και την προσομοίωση της κίνησης του τετράποδου ρομπότ. Σε συνέχεια αυτού του υπομενού υπάρχουν και οι επιλογές του τρίτου υπομενού, οι οποίες θα αναλυθούν αργότερα. Έχουμε λοιπόν:

- **Total time in sec** (συνολική χρονική διάρκεια της προσομοίωσης): Ο χρήστης επιλέγει από ένα συγκεκριμένο σύνολο τιμών, που κυμαίνεται από το 1 έως και τα 20 δευτερόλεπτα, τον συνολικό χρόνο κίνησης του τετράποδου (συνολικός χρόνος της προσομοίωσης).
- **Cycles T in sec** (η χρονική περίοδος των κύκλων κίνησης): Εδώ ορίζεται η χρονική περίοδος των κύκλων κίνησης του ρομπότ, με τον χρήστη να επιλέγει την τιμή από ένα προκαθορισμένο σύνολο από 0.5 έως και 10 δευτερόλεπτα. Περισσότερα για τη σημασία της συγκεκριμένης μεταβλητής θα εξηγηθούν στο τρίτο υπομενού. Το πρόγραμμα φροντίζει έτσι ώστε ο χρήστης να μπορεί να επιλέξει μονάχα τιμές που να δίνουν τέλεια διαίρεση με τη συνολική χρονική διάρκεια της προσομοίωσης. Το αποτέλεσμα της τέλει διαίρεσης είναι ο ακέραιος αριθμός των κύκλων κίνησης της προσομοίωσης που υπολογίζεται αυτόματα από το πρόγραμμα και καταγράφεται δίπλα στην ετικέτα "Cycles number".
- **Gaits T in sec** (η χρονική περίοδος των βηματισμών): Εδώ ορίζεται η χρονική περίοδος των βηματισμών των τεσσάρων ποδιών του ρομπότ, με επιλογή από προκαθορισμένες τιμές που κυμαίνονται από τα 0.1 έως και τα 5 δευτερόλεπτα. Και γι'

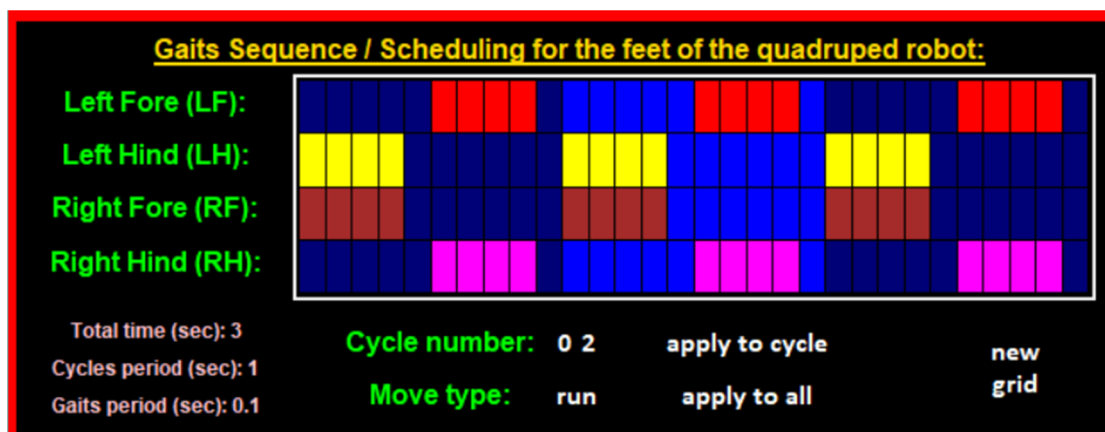


Εικόνα 4

αυτήν την παράμετρο θα δοθούν περισσότερες λεπτομέρειες στο τρίτο υπομενού. Πάλι το πρόγραμμα ρυθμίζει αυτόματα τις επιλογές του χρήστη, ώστε το αποτέλεσμα της διαίρεσης από τον συνολικό χρόνο της προσομοίωσης να είναι ακέραιος αριθμός (φαίνεται δίπλα στην ετικέτα “Gaits number”). Αυτός είναι μάλιστα ο συνολικός αριθμός βηματισμών για κάθε πόδι (είναι κοινός και για τα 4 πόδια).

- **dt in sec** (το χρονικό βήμα της προσομοίωσης): Με το συγκεκριμένο κουμπί ο χρήστης επιλέγει το χρονικό διάστημα που μεσολαβεί μεταξύ δύο διαδοχικών διακριτών βημάτων της προσομοίωσης. Οι τιμές που επιτρέπεται να επιλεγούν είναι συγκεκριμένες, με μικρότερη τα 0.01 και τη μέγιστη τα 0.1 δευτερόλεπτα.
- **Initial state** (η αρχική κατάσταση του τετράποδου): Σχετικά με την αρχική κατάσταση του ρομπότ, δίνεται η δυνατότητα για καθορισμό της αρχικής θέσης του κέντρου μάζας, με επιλογή του πλήκτρου “position” και του αρχικού προσανατολισμού του σώματος, με επιλογή του πλήκτρου “orientation”. Για τον προσδιορισμό του κέντρου μάζας χρειάζεται να δοθούν οι τρεις χωρικές συντεταγμένες της θέσης του και για τον προσανατολισμό του σώματος ο χρήστης πρέπει να εισάγει διαδοχικά τις γωνίες Euler ZYX (δηλαδή τις διαδοχικές περιστροφές του σώματος ως προς τους άξονες Z, Y και X του παγκόσμιου συστήματος συντεταγμένων). Έχω υποθέσει ότι το τετράποδο ξεκινάει την κίνησή του από την ηρεμία, δηλαδή με μηδενικές μεταφορική και γωνιακή ταχύτητα.
- **Final state** (η τελική κατάσταση του τετράποδου): Εδώ ορίζεται η επιθυμητή τελική κατάσταση του τετράποδου, δηλαδή η τελική θέση του κέντρου μάζας (με το αντίστοιχο κουμπί “position”) και ο τελικός προσανατολισμός του σώματος (με το αντίστοιχο κουμπί “orientation”) του τετράποδου. Παρομοίως με την αρχική κατάσταση υποθέτω ότι το τετράποδο καταλήγει σε ακινησία στο τέλος της προσομοίωσης.
- **Simulation** (γενικές επιλογές αναφορικά με την προσομοίωση): Εδώ υπάρχουν δύο κουμπιά, το “START”, που συνδέεται με την κρίσιμότερη λειτουργία της εφαρμογής, και το “show”. Με το πάτημα του κουμπιού “START”, το πρόγραμμα παίρνει όλα τα δεδομένα που έχει εισάγει ο χρήστης για την προσομοίωση στο δεύτερο και στο τρίτο υπομενού και επιχειρεί να βρει τη βέλτιστη τροχιά για την εκτέλεση της επιθυμητής μετακίνησης του τετράποδου. Λόγω του σημαντικού υπολογιστικού φόρτου που ακολουθεί, η γραφική διεπαφή συνήθως κολλάει και τίθεται ξανά προς χρήση με το πέρας της βελτιστοποίησης. Αμέσως μετά την εύρεση της τροχιάς, εμφανίζεται στον χρήστη ένα μήνυμα που τον ενημερώνει για τον βαθμό επιτυχίας της βελτιστοποίησης. Ο χρήστης μπορεί να παρακολουθήσει την υπολογισμένη τροχιά στην περιοχή της οπτικοποίησης πατώντας το κουμπί “show”. Η ταχύτητα της οπτικοποίησης εξαρτάται από την επιλογή “Simulation speed” που έχει αναφερθεί προηγουμένως.

c) **Gaits sequence / Scheduling for the feet of the quadruped robot** (ακολουθία βηματισμών / προγραμματισμός για την κίνηση των ποδιών του τετράποδου ρομπότ): Στο συγκεκριμένο υπομενού πραγματοποιείται η ρύθμιση του τρόπου βαδίσματος και των τεσσάρων ποδιών του ρομπότ. Τα κύρια στοιχεία του υπομενού παρουσιάζονται στη συνέχεια:



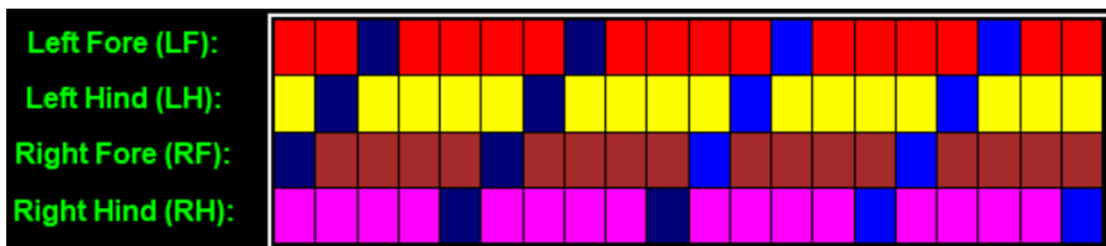
Εικόνα 5



- Πλέγμα σχεδιασμού των βηματισμών: Αποτελεί το πρώτο και σημαντικότερο κομμάτι του τρίτου υπομενού, όπου σχεδιάζονται με εύκολο και διαισθητικό τρόπο οι φάσεις των ποδιών του τετράποδου για ολόκληρη τη διάρκεια της προσομοίωσης (οι default φάσεις πριν την παρέμβαση του χρήστη είναι αιωρήσεις/swings). Πρόκειται για ένα πλέγμα 4 γραμμών, όσες και ο αριθμός των ποδιών, και αριθμού στηλών όσοι είναι οι βηματισμοί (gaits), που για κάθε πόδι το πλήθος τους είναι το ίδιο. Το πλέγμα χωρίζεται σε ορθογώνια κουτάκια, κάθε ένα από τα οποία είναι μία μονάδα βηματισμού (gait), που μπορεί να είναι σε φάση επαφής/contact ή αιώρησης/swing. Η αλλαγή της φάσης γίνεται με αριστερό κλικ του ποντικιού πάνω στη μονάδα gait, με την επαφή να έχει χρώμα κόκκινο, κίτρινο, καφέ ή μοβ αναλόγως αν το εκάστοτε πόδι είναι το μπροστά αριστερό (LF), το πίσω αριστερό (LH), το μπροστά δεξί (RF) ή το πίσω δεξί (RH) αντίστοιχα, και την αιώρηση να δηλώνεται με μία απόχρωση του μπλε. Αντί ο χρήστης να επιλέγει ένα ένα τα gaits που θέλει να μεταβάλλει, μπορεί να επιταχύνει τη διαδικασία πατώντας το δεξί κλικ του ποντικιού, που έχει ως αποτέλεσμα την αλλαγή των φάσεων σε όσα gaits το ποντίκι αγγίζει χωρίς να χρειάζεται παραπάνω κλικ (με το πάτημα της ροδέλας ορίζεται αν οι φάσεις μεταβάλλονται σε contacts ή swings). Ακόμα, το πλέγμα διακρίνεται και σε μεγαλύτερες περιοχές, τους λεγόμενους κύκλους κίνησης, που ξεχωρίζουν ο ένας από τον άλλον λόγω της διαφορετικής απόχρωσης μπλε του χρώματός τους. Αυτός ο διαχωρισμός σε κύκλους είναι βολικός όπως θα φανεί στη συνέχεια, για να δίνουμε γρήγορα στο τετράποδο προκαθορισμένες ακολουθίες συνηθισμένων βηματισμών. Συνοψίζοντας, το πλέγμα είναι ουσιαστικά ένα χρονοδιάγραμμα των βηματισμών της μετακίνησης, που εκτείνεται σε όλο το χρονικό διάστημα που έχει οριστεί (φαίνεται με τη ροζ ένδειξη "Total time (sec)"). Κάθε ορθογώνιο κουτάκι gait έχει χρονική διάρκεια όση η περίοδος των βηματισμών (φαίνεται με τη ροζ ένδειξη "Gaits period (sec)") και κάθε κύκλος κίνησης cycle διαρκεί όσο μία περίοδος κύκλων (φαίνεται με τη ροζ ένδειξη "Cycles period (sec)"). Ας σημειωθεί ότι κάθε μονάδα gait περιλαμβάνει ακέραιο πλήθος χρονικών βημάτων  $dt$ . Για να δημιουργηθεί ένα καινούριο πλέγμα χρειάζεται να πατηθεί το κουμπί "new grid", που έχει τοποθετηθεί κάτω δεξιά στο τρίτο υπομενού.
- Cycle number (ο επιλεγμένος κύκλος): Εδώ υπάρχουν δύο πλήκτρα τα οποία προσδιορίζουν τον αριθμό ενός κύκλου κίνησης, με το ένα πλήκτρο να καθορίζει τις δεκάδες και το άλλο τις μονάδες του αριθμού. Η αρίθμηση ξεκινάει από το 1 και δεν μπορεί να υπερβαίνει τον συνολικό αριθμό των κύκλων κίνησης. Η συγκεκριμένη επιλογή συνδέεται με το κουμπί "apply to cycle", όπως θα εξηγηθεί στη συνέχεια.
- Move type (ο τύπος της κίνησης): Εδώ ο χρήστης επιλέγει τον τρόπο βηματισμού του ρομπότ στη χρονική διάρκεια ενός κύκλου κίνησης. Οι ενδεικτικοί τύποι κίνησης που παρέχονται (φαίνονται και στην εικόνα 6 για δύο κύκλους κίνησης) είναι "walk" (συνηθισμένο περπάτημα, με τρία πόδια να ακουμπάνε στο έδαφος κάθε φορά), "trot" (διαγώνιο περπάτημα, με τα δύο διαγώνια πόδια να ακουμπάνε στο έδαφος κάθε φορά), "pace" (πλάγιο περπάτημα, με τα δύο πόδια της αριστερής ή της δεξιάς πλευράς να ακουμπάνε στο έδαφος κάθε φορά), "run" (τρέξιμο), "jump" (άλμα, αναπήδηση), "all C" (όλες οι φάσεις είναι επαφές/contacts), "all S" (όλες οι φάσεις είναι αιωρήσεις/swings). Ο τύπος κίνησης εισάγεται στο χρονοδιάγραμμα βηματισμών με τα κουμπιά "apply to cycle" και "apply to all", η λειτουργία των οποίων αναλύεται αμέσως μετά.
- apply to cycle, apply to all (εφαρμογή στον επιλεγμένο κύκλο, εφαρμογή σε όλους τους κύκλους): Με το κουμπί "apply to cycle" εφαρμόζεται ο τύπος κίνησης "Move type" μονάχα στον επιλεγμένο κύκλο "Cycle number", σχεδιάζοντας στο πλέγμα την κατάλληλη ακολουθία επαφών/contacts και αιωρήσεων/swings στην περιοχή που πρέπει. Με το κουμπί "apply to all", ο επιλεγμένος τύπος κίνησης εφαρμόζεται σε όλους τους κύκλους, γεμίζοντας έτσι το πλέγμα με ένα και μόνο μοτίβο κίνησης.
- new grid (δημιουργία νέου πλέγματος): Κατασκευάζει ένα καινούριο πλέγμα βηματισμών του τετράποδου ρομπότ, με default τιμές swing (S) για όλα τα gaits. Στην κατασκευή του πλέγματος λαμβάνονται υπόψη ο συνολικός χρόνος της προσομοίωσης ("Total time"), η περίοδος των κύκλων κίνησης ("Cycles T") και η περίοδος των βηματισμών ("Gaits T"), που έχουν οριστεί στο δεύτερο υπομενού.



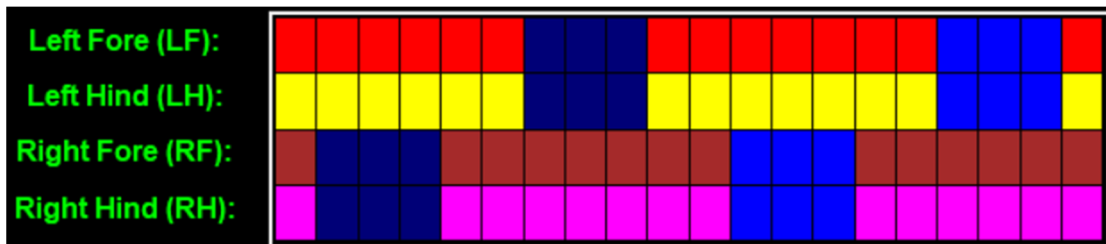
Walk



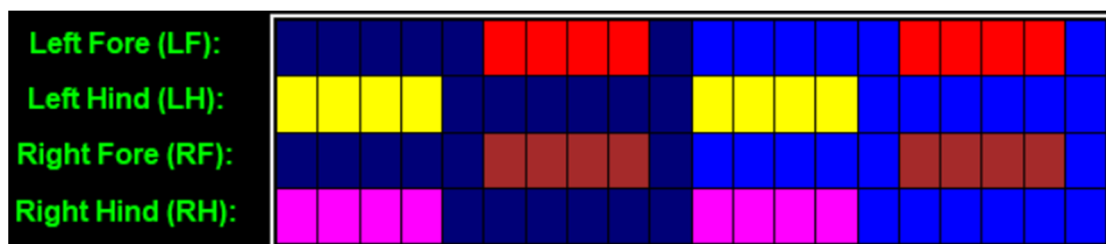
Trot



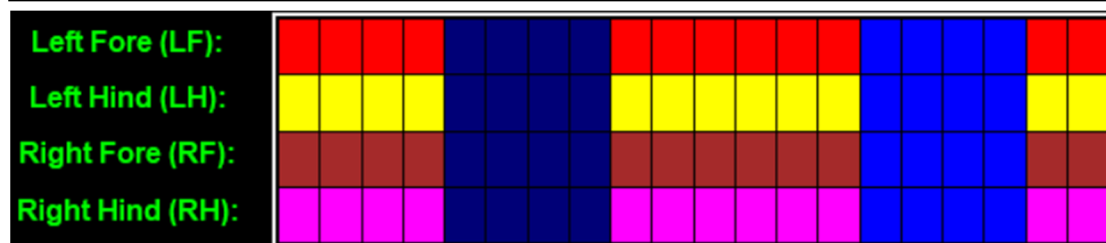
Pace



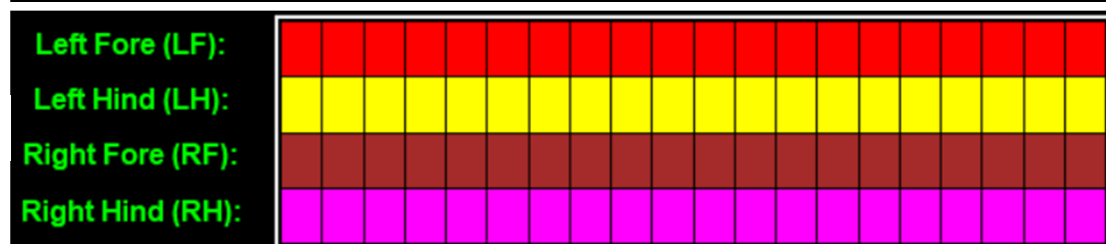
Run



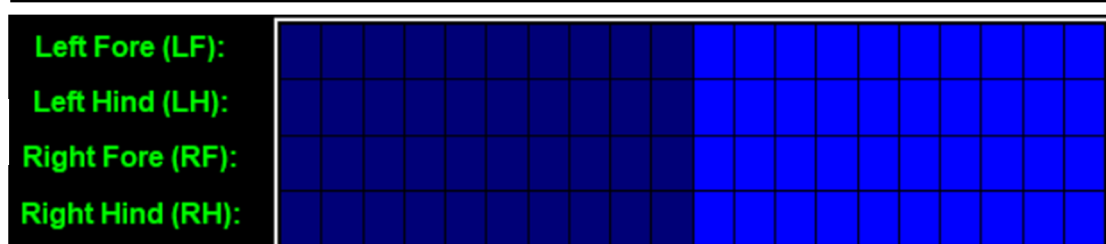
Jump



All contacts



All swings



Εικόνα 6

## Έλεγχος του συστήματος του τετράποδου ρομπότ

Η μελέτη ελέγχου του συστήματος του τετράποδου ρομπότ θα πραγματοποιηθεί σε δύο στάδια. Στο πρώτο στάδιο, θα εξαχθούν οι δυναμικές του τετράποδου και θα υπολογιστούν οι Ιακωβιανές ως προς το διάνυσμα κατάστασης και το διάνυσμα εισόδου. Στο δεύτερο στάδιο θα κατασκευαστεί το κατάλληλο πρόβλημα βελτιστοποίησης που θα βρίσκει τη βέλτιστη τροχιά του τετράποδου, βάσει ορισμένων απαιτούμενων περιορισμών.

- Εξαγωγή δυναμικών και υπολογισμός Ιακωβιανών του μοντέλου του τετράποδου ρομπότ:

Για τη μοντελοποίηση του τετράποδου ρομπότ και των δυναμικών του ακολουθούμε το SRBD (single rigid body dynamics) μοντέλο. Δηλαδή θεωρούμε ότι ολόκληρη η μάζα του ρομπότ βρίσκεται στο σώμα του με συνέπεια τα πόδια να μην έχουν καθόλου μάζα, που αποτελεί μια λογική παραδοχή αφού συνήθως τα πόδια έχουν αμελητέα μάζα σε σχέση με το σώμα. Το διάνυσμα κατάστασης του σώματος καθώς και το συνολικό διάνυσμα κατάστασης του τετράποδου αναγράφονται ακολούθως:

$$x_{body} = \begin{bmatrix} p_{com} \\ \dot{p}_{com} \\ q \\ \omega \end{bmatrix} \in \mathbb{R}^{13} \quad \text{και} \quad x_{quad} = \begin{bmatrix} p_{com} \\ \dot{p}_{com} \\ q \\ \omega \\ p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} \in \mathbb{R}^{25}$$

Όπου  $p_{com} \in \mathbb{R}^3$  είναι η θέση του κέντρου μάζας (λογίζεται ως η θέση του σώματος),  $\dot{p}_{com} \in \mathbb{R}^3$  είναι η μεταφορική ταχύτητα του κέντρου μάζας,  $q \in \mathbb{R}^4$  είναι ο προσανατολισμός του σώματος σε μορφή quaternion και  $\omega \in \mathbb{R}^3$  είναι η γωνιακή ταχύτητα του σώματος. Ακόμα,  $p_1, p_2, p_3, p_4 \in \mathbb{R}^3$  είναι οι θέσεις των τεσσάρων ποδιών (με τη σειρά αριστερό μπροστά, αριστερό πίσω, δεξί μπροστά, δεξί πίσω). Ας σημειωθεί ότι τα διανύσματα  $p_{com}, \dot{p}_{com}, q$  υπολογίζονται ως προς το παγκόσμιο σύστημα συντεταγμένων, ενώ το  $\omega$  βρίσκεται με αναφορά το τοπικό σύστημα συντεταγμένων του σώματος του ρομπότ. Ο έλεγχος του συστήματος γίνεται με τις δυνάμεις  $f_1, f_2, f_3, f_4 \in \mathbb{R}^3$  (υπολογίζονται και αυτές ως προς το παγκόσμιο σύστημα συντεταγμένων) που ασκούνται στα αντίστοιχα αβαρή πόδια του τετράποδου, στα σημεία επαφής τους με το έδαφος, οπότε το διάνυσμα ελέγχου γράφεται ως εξής:

$$u = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \in \mathbb{R}^{12}$$

Για τον σχηματισμό του δυναμικού μοντέλου του συστήματος παραγωγίζω ως προς τον χρόνο μονάχα το διάνυσμα κατάστασης του σώματος, διότι τα πόδια έχουν μηδενική μάζα. Άρα οδηγούμαστε στο:

$$\dot{x}_{body} = f(x_{quad}, u)$$

Όπου  $\dot{x}_{body} = \begin{bmatrix} \dot{p}_{com} \\ \ddot{p}_{com} \\ \dot{q} \\ \dot{\omega} \end{bmatrix}$ , με  $\ddot{p}_{com}$  να είναι η μεταφορική επιτάχυνση του κέντρου μάζας,  $\dot{\omega}$  η γωνιακή επιτάχυνση του σώματος και  $\dot{q}$  η χρονική παράγωγος του quaternion προσανατολισμού του σώματος. Αποδεικνύονται οι παρακάτω σχέσεις:

$$\ddot{\mathbf{p}}_{com} = \frac{\mathbf{f}_{total}}{m}$$

$$\dot{\boldsymbol{\omega}} = I^{-1}(R_w^T \boldsymbol{\tau}_{total} - \boldsymbol{\omega} \times I \boldsymbol{\omega})$$

$$\dot{\mathbf{q}} = \frac{1}{2} \mathbf{q} \odot \begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} = \frac{1}{2} L(\mathbf{q}) \begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} = \frac{1}{2} L(\mathbf{q}) \begin{bmatrix} 0_{1 \times 3} \\ I_{3 \times 3} \end{bmatrix} \boldsymbol{\omega} = \frac{1}{2} L(\mathbf{q}) H \boldsymbol{\omega}, \quad \mu \varepsilon H = \begin{bmatrix} 0_{1 \times 3} \\ I_{3 \times 3} \end{bmatrix}$$

Στις παραπάνω σχέσεις,  $\mathbf{f}_{total} = m\mathbf{g} + \sum_{i=1}^4 \mathbf{f}_i$  είναι η συνολική δύναμη που ασκείται στο ρομπότ, με  $\mathbf{g} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}$  το διάνυσμα της βαρύτητας στη διεύθυνση του άξονα  $z$ , και  $\boldsymbol{\tau}_{total} = \sum_{i=1}^4 (\mathbf{p}_i - \mathbf{p}_{com}) \times \mathbf{f}_i$  είναι η συνολική ροπή. Ακόμα,  $R_w$  είναι ο πίνακας περιστροφής που ισοδυναμεί με το quaternion  $\mathbf{q}$ , εκφρασμένος στο παγκόσμιο σύστημα συντεταγμένων. Αποδεικνύεται ότι ο πίνακας  $R_w$  μπορεί να υπολογιστεί με τον παρακάτω πολλαπλασιασμό πινάκων:

$$R_w = H^T L(\mathbf{q}) R^T(\mathbf{q}) H, \quad \text{οπότε} \quad R_w^T = H^T R(\mathbf{q}) L^T(\mathbf{q}) H = s^2 I_{3 \times 3} - 2s\hat{\mathbf{v}} + \hat{\mathbf{v}}^2 + \mathbf{v}\mathbf{v}^T$$

Όπου  $L(\mathbf{q}) = \begin{bmatrix} s & -\mathbf{v}^T \\ \mathbf{v} & sI + \hat{\mathbf{v}} \end{bmatrix}$  και  $R(\mathbf{q}) = \begin{bmatrix} s & -\mathbf{v}^T \\ \mathbf{v} & sI - \hat{\mathbf{v}} \end{bmatrix}$ , μπορούν να ονομαστούν αντίστοιχα αριστερός και δεξιός πίνακες – τελεστές για το quaternion  $\mathbf{q} = \begin{bmatrix} s \\ \mathbf{v} \end{bmatrix}$  ( $s$  είναι το βαθμωτό μέγεθος και  $\mathbf{v} \in \mathbb{R}^3$  είναι η διανυσματική ποσότητα του  $\mathbf{q}$ ). Επίσης,  $\hat{\mathbf{v}} = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}$  είναι ο αντισυμμετρικός πίνακας που παράγεται από το διάνυσμα  $\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$ .

Χρειαζόμαστε επιπλέον και τις μερικές παραγώγους των συγκεκριμένων δυναμικών ως προς τα ορίσματά τους  $\mathbf{x}_{quad}$  και  $\mathbf{u}$ , συνεπώς απαιτείται ο υπολογισμός των Ιακωβιανών  $A = \frac{\partial \mathbf{f}}{\partial \mathbf{x}_{quad}} \in \mathbb{R}^{13 \times 25}$  και  $B = \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \in \mathbb{R}^{13 \times 12}$ . Έχουμε:

$$A = \frac{\partial \mathbf{f}}{\partial \mathbf{x}_{quad}} = \begin{bmatrix} 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 4} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 4} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{4 \times 3} & 0_{4 \times 3} & \frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{q}} & \frac{\partial \dot{\mathbf{q}}}{\partial \boldsymbol{\omega}} & 0_{4 \times 3} & 0_{4 \times 3} & 0_{4 \times 3} & 0_{4 \times 3} \\ \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{p}_{com}} & 0_{3 \times 3} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{q}} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \boldsymbol{\omega}} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{p}_1} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{p}_2} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{p}_3} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{p}_4} \end{bmatrix}$$

Όπου:

$$\frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{q}} = \frac{1}{2} \cdot \frac{\partial}{\partial \mathbf{q}} (L(\mathbf{q}) H \boldsymbol{\omega}) = \frac{1}{2} \cdot \frac{\partial}{\partial \mathbf{q}} \begin{bmatrix} -\mathbf{v}^T \boldsymbol{\omega} \\ s\boldsymbol{\omega} + \hat{\mathbf{v}} \boldsymbol{\omega} \end{bmatrix} = \frac{1}{2} \cdot \begin{bmatrix} 0 & -\boldsymbol{\omega}^T \\ \boldsymbol{\omega} & -\hat{\boldsymbol{\omega}} \end{bmatrix}, \quad \frac{\partial \dot{\mathbf{q}}}{\partial \boldsymbol{\omega}} = \frac{1}{2} L(\mathbf{q}) H$$

$$\frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{p}_{com}} = I^{-1} R_w^T \sum_{i=1}^4 \hat{\mathbf{f}}_i, \quad \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{p}_i} = -I^{-1} R_w^T \hat{\mathbf{f}}_i, \quad i = 1, 2, 3, 4, \quad \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \boldsymbol{\omega}} = I^{-1} (\widehat{I\boldsymbol{\omega}} - \widehat{\boldsymbol{\omega}} I)$$

$$\text{για } \boldsymbol{\tau}_{total} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \text{ είναι } \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{q}} = I^{-1} \frac{\partial}{\partial \mathbf{q}} (R_w^T \boldsymbol{\tau}_{total}) =$$

$$I^{-1} \begin{bmatrix} 4s\tau_1 + 2v_3\tau_2 - 2v_2\tau_3 & 4v_1\tau_1 + 2v_2\tau_2 + 2v_3\tau_3 & 2v_1\tau_2 - 2s\tau_3 & 2s\tau_2 + 2v_1\tau_3 \\ -2v_3\tau_1 + 4s\tau_2 + 2v_1\tau_3 & 2v_2\tau_1 + 2s\tau_3 & 2v_1\tau_1 + 4v_2\tau_2 + 2v_3\tau_3 & -2s\tau_1 + 2v_2\tau_3 \\ 2v_2\tau_1 - 2v_1\tau_2 + 4s\tau_3 & 2v_3\tau_1 - 2s\tau_2 & 2s\tau_1 + 2v_3\tau_2 & 2v_1\tau_1 + 2v_2\tau_2 + 4v_3\tau_3 \end{bmatrix}$$

Διότι:

$$\frac{\partial}{\partial \mathbf{v}} (\hat{\mathbf{v}} \boldsymbol{\omega}) = \frac{\partial}{\partial \mathbf{v}} \begin{bmatrix} v_2\omega_3 - v_3\omega_2 \\ v_3\omega_1 - v_1\omega_3 \\ v_1\omega_2 - v_2\omega_1 \end{bmatrix} = \begin{bmatrix} 0 & \omega_3 & -\omega_2 \\ -\omega_3 & 0 & \omega_1 \\ \omega_2 & -\omega_1 & 0 \end{bmatrix} = -\hat{\boldsymbol{\omega}}.$$

$$\frac{\partial}{\partial \mathbf{p}_{com}} [\sum_{i=1}^4 (\mathbf{p}_i - \mathbf{p}_{com}) \times \mathbf{f}_i] = \frac{\partial}{\partial \mathbf{p}_{com}} [-\sum_{i=1}^4 (\mathbf{p}_{com} \times \mathbf{f}_i)] = \frac{\partial}{\partial \mathbf{p}_{com}} [-\sum_{i=1}^4 (\hat{\mathbf{p}}_{com} \cdot \mathbf{f}_i)] = \sum_{i=1}^4 \hat{\mathbf{f}}_i.$$

$$\frac{\partial}{\partial \mathbf{p}_i} [\sum_{i=1}^4 (\mathbf{p}_i - \mathbf{p}_{com}) \times \mathbf{f}_i] = \frac{\partial}{\partial \mathbf{p}_i} [\sum_{i=1}^4 (\mathbf{p}_i \times \mathbf{f}_i)] = \frac{\partial}{\partial \mathbf{p}_i} [\sum_{i=1}^4 (\hat{\mathbf{p}}_i \cdot \mathbf{f}_i)] = -\hat{\mathbf{f}}_i.$$

$$\frac{\partial}{\partial \boldsymbol{\omega}} (\boldsymbol{\omega} \times I \boldsymbol{\omega}) = \frac{\partial}{\partial \boldsymbol{\omega}} (\hat{\boldsymbol{\omega}} \cdot I \boldsymbol{\omega}) = \frac{\partial}{\partial \boldsymbol{\omega}} (\hat{\boldsymbol{\omega}}) \cdot I \boldsymbol{\omega} + \hat{\boldsymbol{\omega}} \cdot \frac{\partial}{\partial \boldsymbol{\omega}} (I \boldsymbol{\omega}) = -\hat{I} \hat{\boldsymbol{\omega}} + \hat{\boldsymbol{\omega}} I.$$

$$R_w^\top = s^2 I_{3 \times 3} - 2s \hat{\mathbf{v}} + \hat{\mathbf{v}}^2 + \mathbf{v} \mathbf{v}^\top = \begin{bmatrix} 2s^2 + 2v_1^2 - 1 & 2v_1v_2 + 2sv_3 & 2v_1v_3 - 2sv_2 \\ 2v_1v_2 - 2sv_3 & 2s^2 + 2v_2^2 - 1 & 2v_2v_3 + 2sv_1 \\ 2v_1v_3 + 2sv_2 & 2v_2v_3 - 2sv_1 & 2s^2 + 2v_3^2 - 1 \end{bmatrix}.$$

Και:

$$B = \frac{\partial \mathbf{f}}{\partial \mathbf{u}} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ \frac{\partial \dot{\mathbf{p}}_{com}}{\partial \mathbf{f}_1} & \frac{\partial \dot{\mathbf{p}}_{com}}{\partial \mathbf{f}_2} & \frac{\partial \dot{\mathbf{p}}_{com}}{\partial \mathbf{f}_3} & \frac{\partial \dot{\mathbf{p}}_{com}}{\partial \mathbf{f}_4} \\ 0_{4 \times 3} & 0_{4 \times 3} & 0_{4 \times 3} & 0_{4 \times 3} \\ \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{f}_1} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{f}_2} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{f}_3} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{f}_4} \end{bmatrix}$$

Όπου:

$$\frac{\partial \dot{\mathbf{p}}_{com}}{\partial \mathbf{f}_i} = \frac{I_{3 \times 3}}{m} \alpha \mathbf{f}_i \neq \mathbf{0} \text{ ή } 0_{3 \times 3} \alpha \mathbf{f}_i = \mathbf{0}, i = 1, 2, 3, 4$$

$$\frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{f}_i} = I^{-1} R_w^\top (\hat{\mathbf{p}}_i - \hat{\mathbf{p}}_{com}), i = 1, 2, 3, 4$$

$$\Delta \text{ιότι: } \frac{\partial \boldsymbol{\tau}_{total}}{\partial \mathbf{f}_i} = \frac{\partial}{\partial \mathbf{f}_i} [\sum_{i=1}^4 (\mathbf{p}_i - \mathbf{p}_{com}) \times \mathbf{f}_i] = \frac{\partial}{\partial \mathbf{f}_i} [\sum_{i=1}^4 (\hat{\mathbf{p}}_i - \hat{\mathbf{p}}_{com}) \cdot \mathbf{f}_i] = \hat{\mathbf{p}}_i - \hat{\mathbf{p}}_{com}.$$

Σε όλους τους παραπάνω τύπους ισχύει ότι, αν το πόδι  $i$  βρίσκεται σε επαφή με το έδαφος τότε του ασκείται η δύναμη  $\mathbf{f}_i \neq \mathbf{0}$ , ενώ αν αιωρείται δεν του ασκείται καμία δύναμη δηλαδή  $\mathbf{f}_i = \mathbf{0}$ .

- Σχηματισμός του μη γραμμικού προβλήματος βελτιστοποίησης τροχιάς:

Πρέπει να βρω τη βέλτιστη τροχιά του τετράποδου ρομπότ, δηλαδή τις βέλτιστες κινήσεις των ποδιών του, έτσι ώστε να μετακινηθεί από την αρχική κατάσταση στην τελική, διατηρώντας παράλληλα την ισορροπία του και υπακούοντας κάθε χρονική στιγμή στους περιορισμούς που θα οριστούν στη συνέχεια. Για την ακρίβεια, στο πρόβλημα βελτιστοποίησης που κατασκευάζω δεν θέτω κάποια αντικειμενική συνάρτηση προς ελαχιστοποίηση (η αντικειμενική συνάρτηση είναι μηδενική), που

σημαίνει ότι με ενδιαφέρει αποκλειστικά η ικανοποίηση των περιορισμών. Θεωρώ πως η βέλτιστη τροχιά αποτελείται από διακριτά σημεία κόμβους (knot points)  $(\mathbf{x}_k, \mathbf{u}_k)$ , για την εύρεση των οποίων χρησιμοποιώ τον solver της python βιβλιοθήκης cyipopt. Θέτω τον συνολικό χρόνο της μετακίνησης μέσω της εφαρμογής, έστω ότι είναι  $t_f$ , με χρονικό βήμα  $dt$ . Τότε, τα knot points υπολογίζονται  $K = \left\lfloor \frac{t_f}{dt} \right\rfloor + 1$  στο πλήθος. Γνωρίζοντας ότι η διάσταση του διανύσματος κατάστασης  $\mathbf{x}$  είναι  $N = 25$  και ότι η διάσταση του διανύσματος εισόδου  $\mathbf{u}$  είναι  $M = 12$ , υπολογίζεται ότι ο αριθμός των μεταβλητών βελτιστοποίησης είναι  $x_{dim} = K \cdot N + (K - 1) \cdot M = 37K - 12$  (διότι υπάρχουν  $K$  σημεία κόμβοι και δεν απαιτείται είσοδος ελέγχου για το τελευταίο σημείο). Το διάνυσμα  $\mathbf{x}_{opt}$  στο οποίο συγκεντρώνεται το σύνολο των μεταβλητών απόφασης φαίνεται παρακάτω:

$$\mathbf{x}_{opt} = \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_K \\ \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_{K-1} \end{bmatrix} \in \mathbb{R}^{x_{dim} \times 1}$$

Στο πρόβλημα βελτιστοποίησης που ορίζω στην κλάση “trajectory\_optimization” του αρχείου “quadruped\_robot\_api.py” υπάρχουν και ισοτικοί και ανισοτικοί περιορισμοί. Όπως συμβαίνει και με τις μεταβλητές βελτιστοποίησης μπορούμε να τους τοποθετήσουμε σε ένα διάνυσμα στήλης, το οποίο παίρνει την παρακάτω μορφή:

$$\mathbf{c} = \begin{bmatrix} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_{eq_{dim}+ineq_{dim}} \end{bmatrix} \in \mathbb{R}^{(eq_{dim}+ineq_{dim}) \times 1}$$

Οι ισοτικοί περιορισμοί διακρίνονται στις εξής τρεις ομάδες:

- 1) Ισοτικοί περιορισμοί για την επιβολή των μη γραμμικών δυναμικών του συστήματος (με τη χρήση αριθμητικής ολοκλήρωσης Euler), που είναι  $eq_1 = K - 1$  στο πλήθος λόγω των  $K$  knot points. Λαμβάνουν την ακόλουθη μορφή:

$$\mathbf{c}_j = \mathbf{x}_{body,k+1} - (\mathbf{x}_{body,k} + \dot{\mathbf{x}}_{body,k} \cdot dt), j = 1, \dots, eq_1$$

$$\text{Όπου } \mathbf{x}_{quad,k} = \begin{bmatrix} \mathbf{x}_{body,k} \\ \mathbf{p}_{1,k} \\ \mathbf{p}_{2,k} \\ \mathbf{p}_{3,k} \\ \mathbf{p}_{4,k} \end{bmatrix}, k = 0, 1, \dots, K, \mathbf{u}_k = \begin{bmatrix} \mathbf{f}_{1,k} \\ \mathbf{f}_{2,k} \\ \mathbf{f}_{3,k} \\ \mathbf{f}_{4,k} \end{bmatrix}, k = 0, 1, \dots, K - 1 \text{ και } \dot{\mathbf{x}}_{body,k} = \mathbf{f}(\mathbf{x}_{quad,k}, \mathbf{u}_k).$$

Σχετικά με το διάνυσμα  $\mathbf{u}_k$ , αναλόγως των φάσεων των ποδιών, οι δυνάμεις που τους ασκούνται είναι μη μηδενικές (σε περίπτωση επαφής/contact) ή μηδενικές (σε περίπτωση αιώρησης/swing).

- 2) Ισοτικοί περιορισμοί για το κλείδωμα του κάθε ποδιού σε μία συγκεκριμένη θέση  $(x, y)$  στο επίπεδο του δαπέδου, όταν αυτό βρίσκεται σε επαφή με το έδαφος. Το πλήθος  $eq_2$  των συγκεκριμένων περιορισμών είναι το άθροισμα, για όλα τα πόδια, των χρονικών στιγμών  $dt$  (αντίστοιχα των knot points  $k$ ) στις οποίες το πόδι βρίσκεται σε φάση επαφής/contact, υπό την προϋπόθεση ότι την επόμενη χρονική στιγμή βρίσκεται επίσης σε φάση επαφής. Παίρνουν την εξής μορφή:

$$\mathbf{c}_{eq_1+j} = \bar{\mathbf{p}}_{i,k+1} - \bar{\mathbf{p}}_{i,k}, j = 1, \dots, eq_2$$

Όπου με  $\bar{\mathbf{p}}_{i,k}, \bar{\mathbf{p}}_{i,k+1}$  συμβολίζω τις  $(x, y)$  θέσεις του  $i$  ποδιού στα knot points  $k$  και  $k + 1$  αντίστοιχα. Δεν χρειάζεται να αναφερθούμε εδώ στη συντεταγμένη  $z$ , αφού εξασφαλίζεται ότι στην επαφή του κάθε ποδιού με το έδαφος θα παραμένει σε σταθερή μηδενική τιμή με τη βοήθεια των περιορισμών που έχουν επιβληθεί στις ανάλογες μεταβλητές βελτιστοποίησης.

3) Ισοτικοί περιορισμοί για τη διασφάλιση του μοναδιαίου μέτρου (κανονικοποίηση) των quaternions του κάθε knot point, που είναι  $eq_3 = K$  στο πλήθος. Γράφονται ως εξής:

$$c_{eq_1+eq_2+j} = \mathbf{q}_k^\top \mathbf{q}_k - 1, j = 1, \dots, eq_3$$

Όπου  $\mathbf{q}_k$  είναι ο προσανατολισμός του σώματος του τετράποδου σε μορφή quaternion στο knot point  $k$  της τροχιάς.

Προφανώς, πρέπει να ισχύει:  $c_j = 0, j = 1, \dots, eq_{dim}$ , δηλαδή θέλουμε οι ισοτικοί περιορισμοί να είναι ίσοι με το μηδέν.

Οι ανισοτικοί περιορισμοί χωρίζονται στις ακόλουθες δύο ομάδες:

1) Ανισοτικοί περιορισμοί για τη δημιουργία των κώνων τριβής (friction cones). Η μετακίνηση του ρομπότ οφείλεται στην κάθετη στο έδαφος δύναμη  $f_n$  που ασκείται στο κάθε πόδι (στην κατεύθυνση του άξονα  $z$ ) ως αντίδραση της κατακόρυφης δύναμης που ασκεί το πόδι στο δάπεδο, καθώς και στις δύο δυνάμεις στατικής τριβής  $fr_x$  και  $fr_y$  στις κατευθύνσεις των αξόνων  $x$  και  $y$  αντίστοιχα. Το τετράποδο δεν πρέπει να ολισθαίνει, κατά συνέπεια απαιτούμε το μέτρο της συνισταμένης δύναμης στατικής τριβής να μην υπερβαίνουν το όριο της μέγιστης στατικής τριβής  $\mu \cdot f_n$ , όπου  $\mu$  είναι ο συντελεστής στατικής τριβής. Αυτό σημαίνει ότι το διάνυσμα της ολικής δύναμης  $\mathbf{f}_i = fr_x \cdot \hat{\mathbf{x}} + fr_y \cdot \hat{\mathbf{y}} + f_n \cdot \hat{\mathbf{z}}$  που ασκείται στο εκάστοτε πόδι  $i$ , τοποθετείται πάντα εντός μία κωνικής περιοχής, ο όγκος της οποίας αυξάνεται (που σημαίνει χαλαρότερος περιορισμός) όσο αυξάνεται ο συντελεστής  $\mu$ . Για ευκολία γραμμικοποιούμε τους περιγραφόμενους περιορισμούς, με αποτέλεσμα η περιοριστική περιοχή να μετατρέπεται σε ανάποδη πυραμίδα, στην οποία είναι εγγεγραμμένη η κωνική περιοχή. Καταλήγουμε λοιπόν στους ακόλουθους τέσσερις περιορισμούς για κάθε πόδι, το οποίο βρίσκεται σε επαφή με το έδαφος στο knot point  $k$  (συνολικά  $ineq_1$  περιορισμοί):

$$c_{eq_{dim}+j} = \begin{bmatrix} fr_x - \mu \cdot f_n \\ -fr_x - \mu \cdot f_n \\ fr_y - \mu \cdot f_n \\ -fr_y - \mu \cdot f_n \end{bmatrix} = \begin{bmatrix} (\hat{\mathbf{x}}^\top - \mu \hat{\mathbf{z}}^\top) \cdot \mathbf{f}_{i,k} \\ (-\hat{\mathbf{x}}^\top - \mu \hat{\mathbf{z}}^\top) \cdot \mathbf{f}_{i,k} \\ (\hat{\mathbf{y}}^\top - \mu \hat{\mathbf{z}}^\top) \cdot \mathbf{f}_{i,k} \\ (-\hat{\mathbf{y}}^\top - \mu \hat{\mathbf{z}}^\top) \cdot \mathbf{f}_{i,k} \end{bmatrix}, j = 1, \dots, ineq_1$$

Όπου  $fr_x = \hat{\mathbf{x}}^\top \cdot \mathbf{f}_{i,k}$ ,  $fr_y = \hat{\mathbf{y}}^\top \cdot \mathbf{f}_{i,k}$ ,  $f_n = \hat{\mathbf{z}}^\top \cdot \mathbf{f}_{i,k}$ , όπου ισχύει πως  $\mathbf{u}_k = \begin{bmatrix} f_{1,k} \\ f_{2,k} \\ f_{3,k} \\ f_{4,k} \end{bmatrix}$ , με  $i$  να είναι το

κατάλληλο πόδι. Εδώ απαιτούμε να ισχύει  $c_{eq_{dim}+j} \leq 0, j = 1, \dots, ineq_1$ .

2) Ανισοτικοί περιορισμοί για την οριοθέτηση των κινήσεων των ποδιών, ώστε να είναι όσο το δυνατόν ρεαλιστική και φυσιολογική η μετακίνηση του τετράποδου. Είναι  $ineq_2 = K$  στο πλήθος, όσα και τα knot points. Έχουν την ακόλουθη μορφή:

$$c_{eq_{dim}+ineq_1+j} = R_{w,k}^\top \cdot (\mathbf{f}_{i,k} - \mathbf{p}_{com,k}), i = 1, 2, 3, 4 \text{ και } j = 1, \dots, ineq_2$$

Όπου  $f_{i,k} - p_{com,k}$  είναι το διάνυσμα της απόστασης του ποδιού  $i$  από το κέντρο μάζας του σώματος, στο knot point  $k$ . Η απόσταση αυτή είναι εκφρασμένη στο παγκόσμιο σύστημα συντεταγμένων, οπότε τη μετατρέπω στο σύστημα συντεταγμένων του σώματος πολλαπλασιάζοντάς τη με τον ανάστροφο (που είναι και ο αντίστροφος) του πίνακα περιστροφής  $R_{w,k}$  (ο πίνακας προσανατολισμού του σώματος στο  $k$  knot point). Θέλουμε  $lowerbounds_i \leq c_{eq_{dim}+ineq_1+j} \leq upperbounds_i, j = 1, \dots, ineq_2$ , όπου  $lowerbounds_i$  είναι τα κάτω όρια τοποθέτησης του  $i$  ποδιού και  $upperbounds_i$  είναι τα άνω όρια τοποθέτησης του  $i$  ποδιού. Για τα όρια έχω ορίσει:

$$lowerbounds_i = \begin{bmatrix} p_{x,i} - p_{com,x} - \frac{bodylength}{2} \\ p_{y,i} - p_{com,y} - \frac{bodywidth}{2} \\ -1.3 \cdot \left( feetheight + \frac{bodyheight}{2} \right) \end{bmatrix}$$

$$upperbounds_i = \begin{bmatrix} p_{x,i} - p_{com,x} + \frac{bodylength}{2} \\ p_{y,i} - p_{com,y} + \frac{bodywidth}{2} \\ -0.7 \cdot \left( feetheight + \frac{bodyheight}{2} \right) \end{bmatrix}$$

Όπου  $p_{x,i}, p_{y,i}$  είναι οι  $x, y$  συντεταγμένες του  $i$  ποδιού,  $p_{com,x}, p_{com,y}$  είναι οι  $x, y$  συντεταγμένες του κέντρου μάζας του σώματος,  $bodylength, bodywidth, bodyheight$  είναι τα μήκη του σώματος κατά μήκος των αξόνων  $x, y, z$  και  $feetheight$  είναι το ύψος των ποδιών.

Ο solver της βιβλιοθήκης cygort απαιτεί να περάσουμε στον κώδικα και τις μερικές παραγώγους των περιορισμών  $c$  ως προς τις μεταβλητές απόφασης/βελτιστοποίησης  $x_{opt}$ . Ο πίνακας της Ιακωβιανής  $J \in \mathbb{R}^{(eq_{dim}+ineq_{dim}) \times x_{dim}}$  που προκύπτει γράφεται με τον ακόλουθο τρόπο:

$$J = \begin{bmatrix} \frac{\partial c_1}{\partial x_1} & \dots & \frac{\partial c_1}{\partial x_K} & \frac{\partial c_1}{\partial u_1} & \dots & \frac{\partial c_1}{\partial u_{K-1}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial c_{eq_{dim}+ineq_{dim}}}{\partial x_1} & \dots & \frac{\partial c_{eq_{dim}+ineq_{dim}}}{\partial x_K} & \frac{\partial c_{eq_{dim}+ineq_{dim}}}{\partial u_1} & \dots & \frac{\partial c_{eq_{dim}+ineq_{dim}}}{\partial u_{K-1}} \end{bmatrix}$$

Ακολουθώς αναγράφονται οι επιμέρους μερικές παράγωγοι του πίνακα της Ιακωβιανής  $J$  (όλες οι υπόλοιπες που δεν αναφέρονται είναι μηδενικές). Για τους ισοτικούς περιορισμούς έχουμε:

1) Ισοτικοί περιορισμοί των μη γραμμικών δυναμικών:

$$\frac{\partial c_j}{\partial x_k} = -[I_{13 \times 13} \quad 0_{13 \times 12}] - A|_{x_k, u_k} \cdot dt, j = 1, \dots, eq_1$$

$$\frac{\partial c_j}{\partial x_{k+1}} = [I_{13 \times 13} \quad 0_{13 \times 12}], j = 1, \dots, eq_1$$

$$\frac{\partial c_j}{\partial u_k} = -B|_{x_k, u_k} \cdot dt, j = 1, \dots, eq_1$$



- 2) Ισοτικοί περιορισμοί για το κλείδωμα των θέσεων των ποδιών σε επαφή με το έδαφος:

$$\frac{\partial \mathbf{c}_{eq1+j}}{\partial \bar{\mathbf{p}}_{i,k}} = -\mathbf{I}_{2 \times 2}, j = 1, \dots, eq_2$$

$$\frac{\partial \mathbf{c}_{eq1+j}}{\partial \bar{\mathbf{p}}_{i,k+1}} = \mathbf{I}_{2 \times 2}, j = 1, \dots, eq_2$$

Όπου  $\bar{\mathbf{p}}_{i,k}$  είναι οι  $(x, y)$  θέσεις του  $i$  ποδιού που εμπεριέχονται στη μεταβλητή βελτιστοποίησης  $\mathbf{x}_k$ .

- 3) Ισοτικοί περιορισμοί για την κανονικοποίηση των quaternions:

$$\frac{\partial \mathbf{c}_{eq1+eq2+j}}{\partial \mathbf{q}_k} = 2\mathbf{q}_k^\top, j = 1, \dots, eq_3$$

Όπου  $\mathbf{q}_k$  το quaternion που εμπεριέχεται στη μεταβλητή βελτιστοποίησης  $\mathbf{x}_k$ .

Για τους ανισοτικούς περιορισμούς έχουμε:

- 1) Ανισοτικοί περιορισμοί των κώνων τριβής:

$$\frac{\partial \mathbf{c}_{eqdim+j}}{\partial \mathbf{f}_{i,k}} = \begin{bmatrix} (\hat{\mathbf{x}} - \mu\hat{\mathbf{z}})^\top \\ (-\hat{\mathbf{x}} - \mu\hat{\mathbf{z}})^\top \\ (\hat{\mathbf{y}} - \mu\hat{\mathbf{z}})^\top \\ (-\hat{\mathbf{y}} - \mu\hat{\mathbf{z}})^\top \end{bmatrix}, j = 1, \dots, ineq_1$$

Όπου  $\mathbf{f}_{i,k}$  είναι το διάνυσμα της δύναμης που ασκείται στο  $i$  πόδι, το οποίο βρίσκεται στη μεταβλητή βελτιστοποίησης  $\mathbf{u}_k$ .

- 2) Ανισοτικοί περιορισμοί οριοθέτησης των κινήσεων των ποδιών:

$$\frac{\partial \mathbf{c}_{eqdim+ineq1+j}}{\partial \mathbf{p}_{com,k}} = -\mathbf{R}_{w,k}^\top, i = 1, 2, 3, 4 \text{ και } j = 1, \dots, ineq_2$$

$$\frac{\partial \mathbf{c}_{eqdim+ineq1+j}}{\partial \mathbf{f}_{i,k}} = \mathbf{R}_{w,k}^\top, i = 1, 2, 3, 4 \text{ και } j = 1, \dots, ineq_2$$

$$\frac{\partial \mathbf{c}_{eqdim+ineq1+j}}{\partial \mathbf{q}_k} = \frac{\partial}{\partial \mathbf{q}_k} [\mathbf{R}_{w,k}^\top \cdot (\mathbf{f}_{i,k} - \mathbf{p}_{com,k})], i = 1, 2, 3, 4 \text{ και } j = 1, \dots, ineq_2$$

Όπου  $\mathbf{p}_{com,k}$  και  $\mathbf{q}_k$  είναι η θέση του κέντρου μάζας και το quaternion που βρίσκονται στη μεταβλητή βελτιστοποίησης  $\mathbf{x}_k$  και το  $\mathbf{f}_{i,k}$  εξηγήθηκε προηγουμένως.

Πέρα των ισοτικών και ανισοτικών περιορισμών χρειάζεται να ορίσουμε και τα φράγματα των τιμών των μεταβλητών βελτιστοποίησης (στη συνάρτηση "run\_optimization\_simulation"). Αρχικά, πρέπει να εξασφαλίζεται η ακριβής παρακολούθηση της αρχικής και τελικής κατάστασης (για την τελευταία δεν με ενδιαφέρει η γωνιακή ταχύτητα). Επίσης, τα πόδια δεν επιτρέπεται να διαπερνούν το έδαφος, και στην επαφή τους με αυτό πρέπει να διαθέτουν μηδενική  $z$  συντεταγμένη. Επιπλέον, οι δυνάμεις που ασκούνται στα πόδια (σε φάση επαφής) πρέπει να έχουν φορά προς τα πάνω (το έδαφος σπρώχνει τα πόδια), ενώ τα πόδια που βρίσκονται στον αέρα δεν δέχονται καμία δύναμη. Τέλος, οι συνιστώσες των δυνάμεων δεν επιτρέπεται να υπερβαίνουν σε μέτρο την ποσότητα  $10mg$ .

## Αποτελέσματα της προσομοίωσης

Ακολουθως περιγράφονται κάποια χαρακτηριστικά παραδείγματα εκτέλεσης της βελτιστοποίησης τροχιάς / προσομοίωσης. Επειδή απαιτείται αρκετή ώρα για να τρέξουν (από 3 λεπτά γι' αυτά με τον μικρό συνολικό χρόνο προσομοίωσης ή/και το μεγάλο χρονικό βήμα, μέχρι και μισή ώρα γι' αυτά με τον μεγάλο συνολικό χρόνο προσομοίωσης ή/και το μικρό χρονικό βήμα), έχω καταγράψει σε βίντεο τις συγκεκριμένες προσομοιώσεις, που μπορούν να βρεθούν στον φάκελο “simulations\_videos”.

- Στο πρώτο παράδειγμα (βίντεο “quadruped\_robot\_jump.mp4”) έχω αρχική θέση

$$\mathbf{p}_{com,i} = \begin{bmatrix} 1.00 \\ 1.00 \\ 0.42 \end{bmatrix} \text{ με αρχικό προσανατολισμό τον}$$

default (με την μπροστά όψη του ρομπότ να δείχνει προς τα θετικά του άξονα  $x$  και τα επίπεδα συμμετρίας του παραλληλεπίπεδου σώματος να είναι τα επίπεδα  $xy, xz, yz$ ), και τελική θέση

$$\mathbf{p}_{com,f} = \begin{bmatrix} 3.00 \\ 2.50 \\ 0.42 \end{bmatrix} \text{ με τελικό προσανατολισμό τον}$$

default, περιστρεμμένο κατά  $45^\circ$  ως προς τον άξονα  $z$ . Το είδος της κίνησης είναι “jump” (αναπήδηση) και εφαρμόζεται και στους δύο κύκλους της προσομοίωσης. Η προσομοίωση διαρκεί συνολικό χρόνο  $t_f = 2 \text{ sec}$  και έχει χρονικό βήμα  $dt = 0.05 \text{ sec}$ . Τα τεχνικά χαρακτηριστικά του ρομπότ παρουσιάζονται στην εικόνα 7 (είναι αυτά που έχουν δοθεί από την άσκηση).

- Στο δεύτερο παράδειγμα (βίντεο “quadruped\_robot\_walk.mp4”) η αρχική και η τελική κατάσταση είναι ίδιες με το πρώτο παράδειγμα, με τη μόνη εξαίρεση ότι ο αρχικός προσανατολισμός είναι ο default περιστρεμμένος κατά  $90^\circ$  ως προς τον άξονα  $z$ . Το είδος της κίνησης είναι “walk” (περπάτημα) και στους δύο κύκλους κίνησης της προσομοίωσης. Η προσομοίωση διαρκεί συνολικό χρόνο  $t_f = 3 \text{ sec}$  και έχει χρονικό βήμα  $dt = 0.1 \text{ sec}$ . Τα τεχνικά χαρακτηριστικά του τετράποδου είναι ίδια με αυτά του πρώτου παραδείγματος.

- Στο τρίτο παράδειγμα (βίντεο “quadruped\_robot\_run.mp4”) η αρχική θέση είναι  $\mathbf{p}_{com,i} = \begin{bmatrix} 1.00 \\ 3.00 \\ 0.42 \end{bmatrix}$

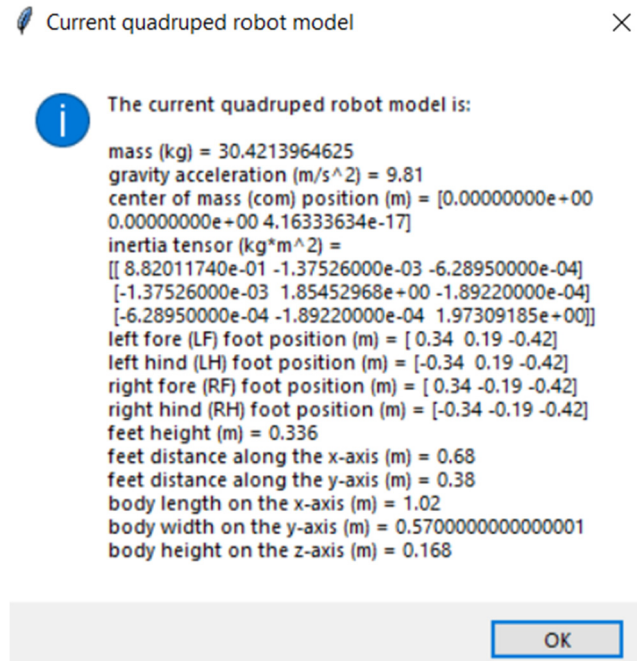
και η τελική θέση είναι η  $\mathbf{p}_{com,f} = \begin{bmatrix} 5.00 \\ 3.00 \\ 0.42 \end{bmatrix}$ , με τους προσανατολισμούς να είναι default και στις δύο

περιπτώσεις. Το είδος της κίνησης είναι “run” (τρέξιμο) καθ' όλη τη διάρκεια της προσομοίωσης, που σημαίνει ότι το τετράποδο πρέπει να τρέξει σε μια ευθεία. Η προσομοίωση διαρκεί συνολικό χρόνο  $t_f = 2 \text{ sec}$  και έχει χρονικό βήμα  $dt = 0.1 \text{ sec}$ . Τα τεχνικά χαρακτηριστικά του τετράποδου είναι ίδια με αυτά του πρώτου παραδείγματος.

- Στο τέταρτο παράδειγμα (βίντεο “quadruped\_robot\_trot\_run.mp4”) η αρχική θέση ορίστηκε

$$\mathbf{p}_{com,i} = \begin{bmatrix} 1.00 \\ 4.00 \\ 0.42 \end{bmatrix} \text{ με default αρχικό προσανατολισμό και έχω τελική θέση } \mathbf{p}_{com,f} = \begin{bmatrix} 6.00 \\ 1.50 \\ 0.42 \end{bmatrix} \text{ με τελικό}$$

προσανατολισμό τον default περιστρεμμένο κατά  $-30^\circ$  ως προς τον άξονα  $z$ . Η προσομοίωση διαρκεί συνολικό χρόνο  $t_f = 3 \text{ sec}$  με χρονικό βήμα  $dt = 0.05 \text{ sec}$  και τα τεχνικά χαρακτηριστικά του



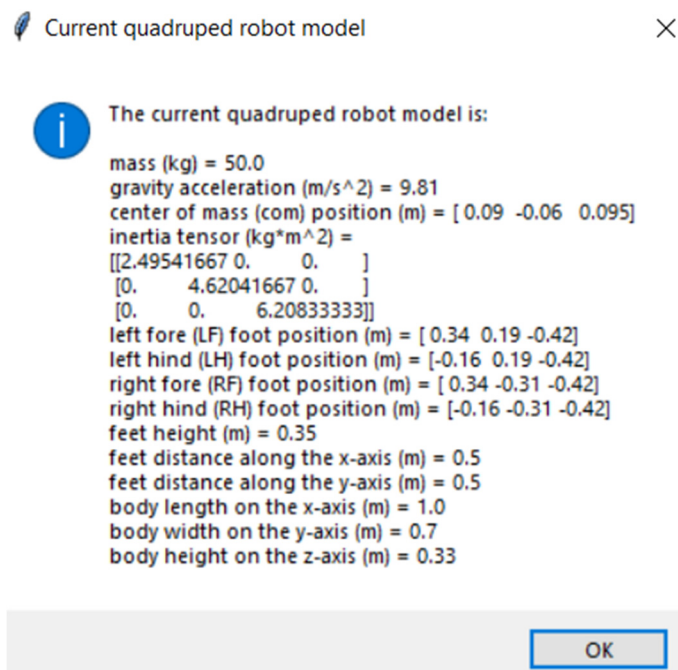
Εικόνα 7

τετράποδου είναι ίδια με αυτά του πρώτου παραδείγματος. Αυτή τη φορά υπάρχουν τρεις κύκλοι κίνησης, με τον πρώτο να είναι “trot” (διαγώνιο περπάτημα) και τους δύο τελευταίους να είναι “run” (τρέξιμο), δηλαδή η κίνηση του τετράποδου ρομπότ είναι μικτή.

- Στο πέμπτο και τελευταίο παράδειγμα (βίντεο “quadruped\_robot\_race.mp4”) έχω αλλάξει τη μοντελοποίηση του ρομπότ, τα νέα τεχνικά χαρακτηριστικά του οποίου δίνονται στην εικόνα 8. Η αρχική

θέση είναι  $\mathbf{p}_{com,i} = \begin{bmatrix} 1.00 \\ 1.00 \\ 0.52 \end{bmatrix}$  και η τελική θέση είναι η  $\mathbf{p}_{com,f} = \begin{bmatrix} 1.70 \\ 2.00 \\ 0.52 \end{bmatrix}$ , με τους προσανατολισμούς να

είναι default και στις δύο περιπτώσεις. Η προσομοίωση διαρκεί συνολικό χρόνο  $t_f = 3 \text{ sec}$  με χρονικό βήμα  $dt = 0.05 \text{ sec}$  και συνίσταται από τρεις κύκλους κίνησης. Σε όλους τους κύκλους έχει οριστεί η κίνηση “race” (πλάγιο περπάτημα). Όπως φαίνεται και στο βίντεο, η αρχική και η τελική κατάσταση αναγκάζουν το τετράποδο να κινηθεί σαν κάβουρας.



Εικόνα 8

Ας σημειωθεί ότι, για οικονομία χρόνου, έχω ορίσει μέγιστο αριθμό επαναλήψεων τις 100 στον αλγόριθμο βελτιστοποίησης. Παρατήρησα ότι στην πλειοψηφία των περιπτώσεων, ειδικά όταν η μετακίνηση του ρομπότ γίνεται σε μεγάλη απόσταση και εμπεριέχει και περιστροφή, δεν είναι δυνατόν να βρεθεί βέλτιστη τροχιά που να ικανοποιεί όλους τους περιορισμούς του προβλήματος πριν την ολοκλήρωση των 100 επαναλήψεων. Εντούτοις, όπως φαίνεται χαρακτηριστικά και στα βίντεο, η κίνηση του σώματος και των ποδιών του τετράποδου είναι αρκετά ρεαλιστικές και οι βασικοί περιορισμοί (παρακολούθηση των δυναμικών του μοντέλου, κλείδωμα των ποδιών στην επαφή τους με το έδαφος κ.ά.), καθώς και το είδος της κίνησης που έχει οριστεί (περπάτημα, αναπήδηση κ.ά.), αποτυπώνονται πολύ ικανοποιητικά στην προσομοίωση.