



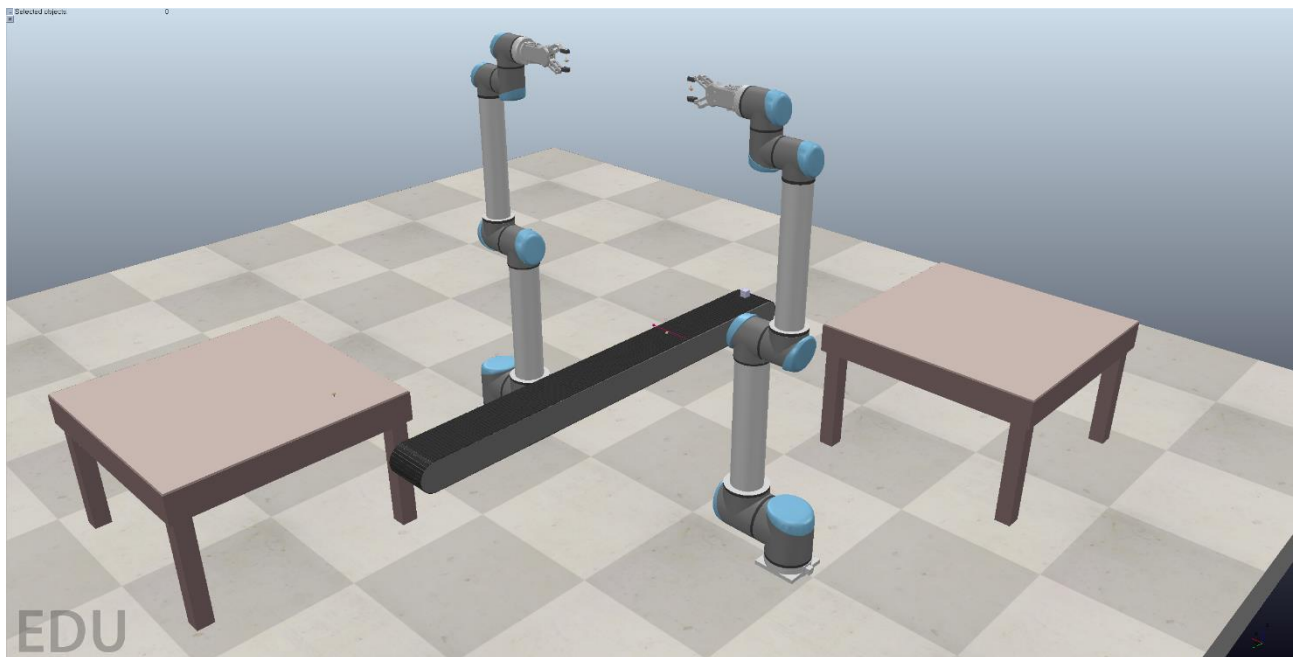
ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ

Διπλωματική Εργασία

Προσομοίωση Ρομπότ μέσω Υπολογιστή



Φοιτητής: Ευάγγελος Κουτουλούλης

AM: 45075

**Επιβλέπων
Διονύσης Κανδρής,
Καθηγητής**

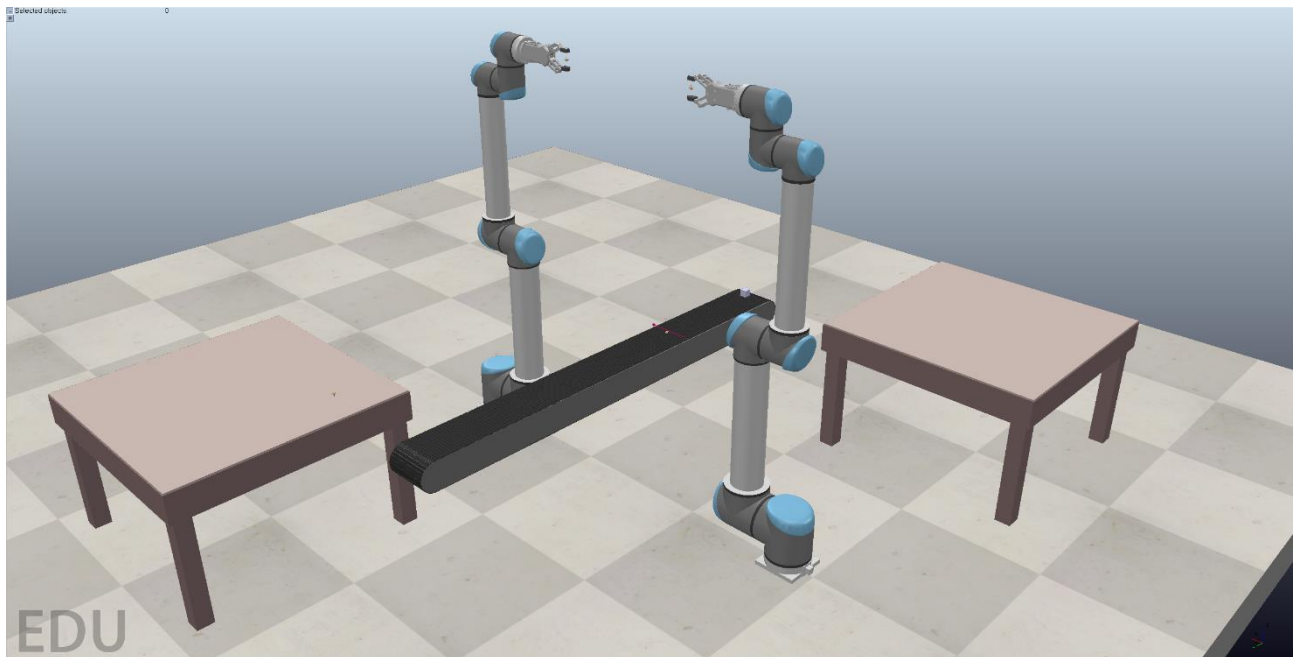
ΑΘΗΝΑ-ΑΙΓΑΛΕΩ, ΟΚΤΩΒΡΙΟΣ 2024



UNIVERSITY OF WEST ATTICA
FACULTY OF ENGINEERING
DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING

Diploma Thesis

Computer-aided Robot Simulation



Student: Evangelos Koutouloulis
Registration Number: 45075

Supervisor
Dionisis Kandris
Professor

ATHENS-EGALEO, October 2024

Η Διπλωματική Εργασία έγινε αποδεκτή και βαθμολογήθηκε από την εξής τριμελή επιτροπή:

Διονύσης Κανδρής, Καθηγητής	Γιώργος Πάτσης, Καθηγητής	Ηλίας Ζωής, Αναπληρωτής Καθηγητής
(Υπογραφή)	(Υπογραφή)	(Υπογραφή)

ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ και Ευάγγελος Κουτουλούλης, Οκτώβριος 2024

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τους συγγραφείς.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον/την συγγραφέα του και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις θέσεις του επιβλέποντος, της επιτροπής εξέτασης ή τις επίσημες θέσεις του Τμήματος και του Ιδρύματος.

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ


Ο κάτωθι υπογεγραμμένος Ευάγγελος Κουτουλούλης του Αντωνίου, με αριθμό μητρώου 45075 φοιτητής του Πανεπιστημίου Δυτικής Αττικής της Σχολής ΜΗΧΑΝΙΚΩΝ του Τμήματος ΗΛΕΚΤΡΟΛΟΓΩΝ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ,

δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της διπλωματικής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του διπλώματός μου.»

Ο Δηλών
Ευάγγελος Κουτουλούλης



Περίληψη

Η παρούσα διπλωματική εργασία εστιάζει στην προσομοίωση ρομπότ μέσω υπολογιστή, με σκοπό την ανάλυση και την αξιολόγηση των λογισμικών CoppeliaSim και Gazebo. Στην εργασία εξετάζεται η σημασία της προσομοίωσης στην ανάπτυξη και τη δοκιμή αλγορίθμων ελέγχου ρομπότ. Με τη χρήση των παραπάνω εργαλείων, οι ερευνητές και οι μηχανικοί μπορούν να εκτελούν πειράματα σε εικονικά περιβάλλοντα, περιορίζοντας τον κίνδυνο βλαβών και μειώνοντας το κόστος φυσικών πειραμάτων. Η μεθοδολογία περιλαμβάνει τη βιβλιογραφική ανασκόπηση, τη διεξαγωγή προσομοιώσεων και την ανάλυση των αποτελεσμάτων για την εξαγωγή συμπερασμάτων σχετικά με την αποδοτικότητα και την ευχρηστία των εργαλείων. Τα αποτελέσματα δείχνουν ότι και τα δύο προγράμματα προσφέρουν σημαντικές δυνατότητες στην εκπαίδευση και την έρευνα στη ρομποτική, συμβάλλοντας στην εξέλιξη της τεχνολογίας.

Λέξεις – κλειδιά

- Ρομποτική προσομοίωση
- Ρομποτική μηχανική
- Αυτοματισμός
- Έλεγχος ρομπότ
- Βιομηχανική ρομποτική
- Αλγόριθμοι
- Προγραμματισμός ρομπότ
- Προσομοίωση

Abstract

This diploma thesis focuses on computer-aided robot simulation, aiming to analyze and evaluate the software packages CoppeliaSim and Gazebo. The study examines the significance of simulation in the development and testing of robot control algorithms. By utilizing these tools, researchers and engineers can conduct experiments in virtual environments, minimizing the risk of damage and reducing the cost of physical experiments. The methodology includes a literature review, the execution of simulations, and the analysis of results to draw conclusions regarding the effectiveness and usability of the tools. The findings indicate that both programs offer significant capabilities for education and research in robotics, contributing to the advancement of technology.

Keywords




- Robot Simulation
- Robotic Engineering
- Automation
- Robot Control
- Industrial Robotics
- Algorithms
- Robot Programming
- Simulation

Περιεχόμενα

ΕΙΣΑΓΩΓΗ.....	12
Αντικείμενο της διπλωματικής εργασίας.....	13
Σκοπός και στόχοι	14
Μεθοδολογία	14
Καινοτομία	14
Δομή	15
ΚΕΦΑΛΑΙΟ 1 ^ο : CoppeliaSim.....	16
1.1 Εισαγωγή κεφαλαίου 1	16
1.2 Περιβάλλον Χρήστη	18
1.3 Σελίδες και όψεις	23
1.3.1 Διαχείριση θέσης/Προσανατολισμού	26
1.3.2 Διαχείριση Θέσης	26
1.3.3 Mouse Translation	26
1.3.4 Position	27
1.3.5 Translation	27
1.3.6 Position scaling.....	27
1.3.7 Orientation dialog/Διάλογος προσανατολισμού.....	28
1.3.8 Mouse Rotation	28
1.3.9 Orientation.....	28
1.3.10 Rotation	29
1.3.11 Object movement with the mouse	29
1.3.12 User Settings/Ρυθμίσεις χρήστη	30
1.3.13 OpenGL settings	32
1.3.14 Shortcuts/Συντομεύσεις.....	32
1.3.15 Σκηνές και μοντέλα	33
1.3.16 Σκηνές.....	34
1.3.17 Μοντέλα	35
1.3.18 Διάλογος μοντέλου/Model dialog	37
1.3.19 Περιβάλλον.....	38
1.3.20 Διάλογος περιβάλλοντος/Environment dialog	38
1.4 Διάλογος υφής/Texture dialog	39
1.4.1 Οντότητες/Entities.....	41
1.4.2 Αντικείμενα σκηνής/Scene objects.....	42
1.4.3 Ιδιότητες αντικειμένων σκηνής/Scene object properties.....	44
1.4.4 Κοινές ιδιότητες αντικειμένου/Object common properties.....	45
1.4.5 Συγκρούσιμα αντικείμενα/Collidable objects	47
1.4.6 Μετρήσιμα αντικείμενα/Measurable objects	48

Προσομοίωση Ρομπότ μέσω Υπολογιστή

1.4.7	Ανιχνεύσιμα αντικείμενα/Detectable objects	48
1.4.8	Ορατά αντικείμενα/Viewable objects.....	49
1.4.9	Convex hull	49
1.4.10	Παράθυρο διαλόγου επιλογής επιπέδων/Layer selection dialog.....	50
1.4.11	Καμερες/Cameras	50
1.4.12	Ιδιότητες κάμερας/Camera properties	53
1.4.13	Φώτα/Lights	54
1.4.14	Ιδιότητες φωτός/Light properties	55
1.4.15	Σχήματα/Shapes	56
1.4.16	Αναφορικό πλαίσιο σχήματος και κουτί οριοθέτησης/Shape reference frame and bounding box.....	58
1.4.17	Πρωτόγονα σχήματα/Primitive shapes.....	59
1.4.18	Αποσύνθεση κυρτών μερών/Convex decomposition.....	60
1.4.19	Ιδιότητες σχήματος/Shape properties.....	61
1.4.20	Ιδιότητες δυναμικής του σχήματος/Shape dynamics properties	62
1.4.21	Παράθυρο διαλόγου γεωμετρίας σχήματος/Shape geometry dialog.....	63
1.4.22	Λειτουργίες επεξεργασίας σχήματος/Shape edit modes	64
1.4.23	Λειτουργία επεξεργασίας τριγώνων/Triangle edit mode.....	65
1.4.24	Λειτουργία επεξεργασίας ακμής/Edge edit mode	69
1.4.25	Λειτουργία επεξεργασίας για σύνθετα σχήματα/Edit mode for compound shapes	71
1.4.26	Αρθρώσεις/Joints 	72
1.4.27	Καταστάσεις αρθρώσεων/Joint modes.....	74
1.4.28	Ιδιότητες αρθρώσεων/Joint properties	75
1.4.29	Δυναμικές ιδιότητες αρθρώσεων/Joint dynamics properties	76
1.4.30	Ανδρείκελα ή Ομοιώματα/Dummies 	77
1.4.31	Ιδιότητες ομοιώματος/Dummy properties.....	78
1.4.32	Γραφήματα/Graphs 	79
1.4.33	Παράθυρο διαλόγου γραφήματος/Graph dialog.....	81
1.4.34	Αισθητήρες εγγύτητας/Proximity sensors 	82
1.4.35	Τύποι αισθητήρων εγγύτητας και τρόπος λειτουργίας/Proximity sensor types and more of operation	83
1.4.36	Ιδιότητες Αισθητήρων Εγγύτητας/Proximity sensor properties.....	84
1.4.37	Παράθυρο διαλόγου όγκου αισθητήρα εγγύτητας/Proximity sensor volume dialog..	85
1.4.38	Παράθυρο διαλόγου παραμέτρων ανίχνευσης αισθητήρα εγγύτητας/Proximity sensor detection parameter dialog	86
1.4.39	Αισθητήρες όρασης/Vision sensors 	88
1.4.40	Τύποι αισθητήρων όρασης και τρόπος λειτουργίας/Vision sensor types and mode of operation.....	89

Προσομοίωση Ρομπότ μέσω Υπολογιστή	
1.4.41	Ιδιότητες αισθητήρων όρασης/Vision sensor properties 89
1.4.42	Αισθητήρες δύναμης/Force sensors  92
1.4.43	Ιδιότητες αισθητήρα δύναμης/Force sensor properties 93
1.4.44	Δέντρα OC/OC trees  94
1.4.45	Νέφη σημείων/Point clouds  96
1.4.46	Ιδιότητες νέφους σημείων/Point cloud properties 97
1.4.47	Διαδρομές/Paths 99
1.4.48	Ανίχνευση σύγκρουσης/Collision detection 102
1.4.49	Κινηματική/Kinematics 103
1.4.50	Επίλυση Εργασιών Αντίστροφης/Άμεσης Κινηματικής..... 104
1.4.51	Δυναμική/Dynamics 105
1.4.52	Διαφορές μεταξύ μηχανών φυσικής..... 106
1.4.53	Ανάπτυξη κώδικα μέσα και γύρω από το CoppeliaSim 108
1.4.54	Προσομοίωση.....110
1.4.55	Παράθυρο διαλόγου προσομοίωσης110
1.5	Δημιουργία Ρομποτικού Βραχίονα στο CoppeliaSim112
ΚΕΦΑΛΑΙΟ 2 ^ο : Gazebo 139
2.1	Τι είναι το Gazebo; 139
2.2	Αρχιτεκτονική Gazebo Sim 139
2.3	Οδηγίες εγκατάστασης για το Ubuntu 140
2.4	User Interface/Διεπαφή Χρήστη..... 142
2.4.1	GUI..... 142
2.4.2	Η Σκηνή..... 143
2.4.3	Τα Πάνελ 143
2.4.4	Τα Εργαλεία..... 145
2.4.5	Το Μενού..... 146
2.4.6	Χειριστήρια Ποντικιού 146
2.5	Επεξεργαστής Μοντέλου / Model Editor 146
2.5.1	Διεπαφή Χρήστη του Επεξεργαστή Μοντέλου / Model Editor User Interface..... 147
2.5.2	Περιορισμοί..... 148
2.6	Κατασκευή Οχήματος..... 148
2.6.1	Σασί 148
2.6.2	Μπροστινοί Τροχοί..... 149
2.6.3	Περιστρεφόμενος Τροχός..... 161
2.6.4	Προσθήκη Αισθητήρα 165
2.6.5	Προσθήκη Προσθέτου..... 168
2.6.6	Αποθήκευση του Μοντέλου 171
2.7	Εξώθηση αρχείων SVG 172

Προσομοίωση Ρομπότ μέσω Υπολογιστή

2.7.1	Χρήση του επεξεργαστή SVG του Inkscape.....	172
2.7.2	Προετοιμασία του εγγράφου.....	172
2.7.3	Σχέδιο.....	175
2.7.4	Αποθήκευση του σχεδίου.....	176
2.8	Δημιουργία Μοντέλου στο Gazebo.....	177
2.9	Επεξεργαστής Κτιρίων (Building Editor).....	179
2.9.1	Γραφική διεπαφή χρήστη (Graphical User Interface).....	180
2.9.2	Εισαγωγή κάτοψης.....	180
2.9.3	Προσθήκη χαρακτηριστικών.....	182
2.10	Καταγραφή και αναπαραγωγή (Logging and Playback).....	197
2.10.1	Αρχεία καταγραφής Gazebo.....	197
2.10.2	Καταγραφή αρχείου καταγραφής.....	199
2.10.3	Αναπαραγωγή αρχείου καταγραφής.....	200
2.10.4	Οπτική αναπαραγωγή στο GUI.....	200
2.11	Δομή μοντέλου και απαιτήσεις.....	202
2.12	Το Αποθετήριο της Βάσης Δεδομένων Μοντέλων.....	202
2.12.1	Βάση Δεδομένων.....	203
2.13	Κατασκευή ενός Κινητού Ρομπότ.....	205
2.13.1	Ρύθμιση του φακέλου του μοντέλου μας.....	205
2.13.2	Κατασκευή της Δομής του Μοντέλου.....	208
2.14	Robot Operating System (ROS).....	216
2.15	RViz.....	219
2.15.1	Προβολές (Displays).....	219
2.15.2	Προσθήκη νέας προβολής.....	219
2.15.3	Ιδιότητες Προβολής.....	220
2.15.4	Κατάσταση Προβολής.....	220
2.15.5	Διαμορφώσεις.....	223
2.15.6	Πίνακας Προβολών (Views Panel).....	223
2.15.7	Κάμερα σε τροχιά (προεπιλογή).....	224
2.15.8	Κάμερα FPS (πρώτου προσώπου).....	224
2.15.9	Ορθογραφική κάμερα από πάνω προς τα κάτω.....	224
2.15.10	Προσαρμοσμένες Προβολές.....	225
2.15.11	Συστήματα Συντεταγμένων.....	226
2.15.12	Το Σταθερό Πλαίσιο.....	226
2.15.13	Το Πλαίσιο-Στόχος.....	226
2.15.14	Εργαλεία.....	226
2.15.15	Αλληλεπίδραση (Interact).....	226
2.15.16	Μετακίνηση Κάμερας (Move Camera).....	227

<i>Προσομοίωση Ρομπότ μέσω Υπολογιστή</i>	
2.15.17 Επιλογή (Select).....	227
2.15.18 Εστίαση Κάμερας (Focus Camera).....	227
2.15.19 Μέτρηση (Measure).....	227
2.15.20 Χρόνος (Time)	227
2.16 Δημιουργία ρομποτικού βραχίονα στο Gazebo	228
ΚΕΦΑΛΑΙΟ 3 ^ο : ΣΥΜΠΕΡΑΣΜΑΤΑ	249
Βιβλιογραφία – Διαδικτυακές Πηγές	251

ΕΙΣΑΓΩΓΗ

Η ρομποτική αποτελεί έναν από τους ταχύτερα αναπτυσσόμενους τομείς της τεχνολογίας, με εφαρμογές που εκτείνονται από τη βιομηχανία και την ιατρική, μέχρι τη διαστημική εξερεύνηση και την οικιακή αυτοματοποίηση. Στον πυρήνα της ρομποτικής βρίσκεται η ανάγκη για ασφαλή και αποτελεσματική προγραμματισμένη λειτουργία ρομπότ, η οποία μπορεί να επιτευχθεί μέσω προσομοιώσεων.

Η προσομοίωση ρομπότ είναι ιδιαίτερα χρήσιμη για πολλούς λόγους. Αρχικά, επιτρέπει την ασφαλή δοκιμή και βελτίωση αλγορίθμων ελέγχου σε εικονικά περιβάλλοντα, μειώνοντας τον κίνδυνο βλαβών σε πραγματικά ρομπότ και την πιθανότητα ανθρώπινων σφαλμάτων. Επιπλέον, η προσομοίωση παρέχει τη δυνατότητα για γρήγορη επανάληψη και πειραματισμό, επιτρέποντας στους ερευνητές και τους μηχανικούς να αξιολογούν διάφορες στρατηγικές και σενάρια χωρίς το υψηλό κόστος των φυσικών πειραμάτων. Τέλος, οι προσομοιώσεις διευκολύνουν την εκπαίδευση και την εκμάθηση, προσφέροντας στους φοιτητές και τους επαγγελματίες τη δυνατότητα να κατανοήσουν τις αρχές της ρομποτικής και τις αλληλεπιδράσεις του ρομπότ με το περιβάλλον του.

Η παρούσα διπλωματική εργασία εστιάζει στη μελέτη λογισμικών προσομοίωσης ρομπότ, συγκεκριμένα στα CoppeliaSim και Gazebo. Αυτά τα εργαλεία παρέχουν ένα ευέλικτο και ισχυρό περιβάλλον για την ανάπτυξη και δοκιμή αλγορίθμων ελέγχου ρομπότ, προσφέροντας τη δυνατότητα προσομοίωσης σε διάφορες συνθήκες και σενάρια.

Ο σκοπός της εργασίας είναι να αναλυθούν οι λειτουργίες και οι δυνατότητες των παραπάνω λογισμικών, προκειμένου να κατανοηθεί η σημασία τους στην εκπαίδευση και την έρευνα της ρομποτικής. Μέσα από τη διεξαγωγή προσομοιώσεων, επιδιώκεται η εξαγωγή συμπερασμάτων σχετικά με την απόδοση και την ευχρηστία τους.

Η μεθοδολογία που ακολουθείται περιλαμβάνει τη διεξαγωγή αναλύσεων και πειραμάτων με τη χρήση των λογισμικών CoppeliaSim και Gazebo, με στόχο την εξαγωγή πολύτιμων δεδομένων για την αξιολόγηση των εργαλείων αυτών και τη συμβολή τους στην πρόοδο της ρομποτικής τεχνολογίας.

Αντικείμενο της διπλωματικής εργασίας

Το αντικείμενο της παρούσας διπλωματικής εργασίας εστιάζει στη μελέτη και ανάλυση λογισμικών προσομοίωσης ρομπότ, με συγκεκριμένη αναφορά στα CoppeliaSim και Gazebo. Η ρομποτική, ως επιστημονικός και τεχνολογικός τομέας, διαδραματίζει κεντρικό ρόλο στη σύγχρονη βιομηχανία και καθημερινή ζωή, με τις προσομοιώσεις να παρέχουν έναν κρίσιμο μηχανισμό για την ανάπτυξη και δοκιμή ρομποτικών αλγορίθμων.

Η μελέτη των εργαλείων προσομοίωσης είναι αναγκαία για την κατανόηση των δυνατοτήτων που προσφέρουν για την εκπαιδευτική διαδικασία και την έρευνα. Το CoppeliaSim, με την ευέλικτη αρχιτεκτονική του και τη δυνατότητα ενσωμάτωσης με άλλες εφαρμογές, προσφέρει στους χρήστες τη δυνατότητα προσομοίωσης πολλών ρομπότ ταυτόχρονα. Από την άλλη, το Gazebo, γνωστό για τη ρεαλιστική φυσική και τα εξαιρετικά γραφικά του, αποτελεί ένα δημοφιλές εργαλείο για τη δοκιμή ρομπότ σε εικονικά περιβάλλοντα.

Σκοπός της ανάλυσης αυτής είναι η εξαγωγή συμπερασμάτων σχετικά με την αποδοτικότητα και την ευχρηστία των εργαλείων αυτών, καθώς και η αναγνώριση των τομέων στους οποίους μπορούν να συνεισφέρουν στη ρομποτική. Με τη διεξαγωγή προσομοιώσεων και την ανάλυση των αποτελεσμάτων, η εργασία προσφέρει μια ολοκληρωμένη εικόνα για την αξία και τη σημασία των λογισμικών προσομοίωσης στην εκπαίδευση και την έρευνα της ρομποτικής.

Σκοπός και στόχοι

Ο σκοπός της παρούσας διπλωματικής εργασίας είναι να μελετήσουμε και να αναλύσουμε τα προγράμματα προσομοίωσης ρομπότ, εστιάζοντας στα CoppeliaSim και Gazebo. Επιδιώκουμε να κατανοήσουμε τις λειτουργίες, τα πλεονεκτήματα και τους περιορισμούς αυτών των εργαλείων, καθώς και να αξιολογήσουμε τη συνεισφορά τους στην εκπαίδευση και την έρευνα της ρομποτικής.

Οι στόχοι της εργασίας αναλύονται σε επιμέρους ερωτήματα προς διερεύνηση, τα οποία περιλαμβάνουν τα εξής:

- Ποιες είναι οι βασικές δυνατότητες και τα χαρακτηριστικά του CoppeliaSim και του Gazebo;
- Ποιες είναι οι διαφορές στην απόδοση και την ευχρηστία των δύο εργαλείων κατά τη διάρκεια προσομοιώσεων σε διάφορες συνθήκες;
- Πώς μπορούμε να αξιολογήσουμε την αποτελεσματικότητα και την αξιοπιστία των εργαλείων στην εκπαίδευση και την έρευνα;
- Ποιες είναι οι καλύτερες πρακτικές για την προσομοίωση ρομπότ με τη χρήση αυτών των εργαλείων;
- Ποιες προτάσεις μπορούν να παρασχεθούν για τη χρήση των εργαλείων προσομοίωσης στην εκπαίδευση και την ανάπτυξη ρομποτικών εφαρμογών;

Μεθοδολογία

Η μεθοδολογία της παρούσας διπλωματικής εργασίας περιλαμβάνει μια συνδυαστική προσέγγιση, συνδυάζοντας θεωρητική ανάλυση και πρακτική εφαρμογή μέσω προσομοιώσεων.

Αρχικά, θα διεξαχθεί βιβλιογραφική ανασκόπηση που θα εστιάζει σε άρθρα, έρευνες και άλλες πηγές σχετικά με τα προγράμματα προσομοίωσης CoppeliaSim και Gazebo. Αυτή η ανασκόπηση παρέχει το θεωρητικό πλαίσιο για την κατανόηση των λειτουργιών και των χαρακτηριστικών τους.

Στη συνέχεια, αναπτύσσονται και εκτελούνται προσομοιώσεις.

Αφού ολοκληρωθούν οι προσομοιώσεις, θα τις αναλύσουμε και θα συγκρίνουμε αποτελέσματα, ώστε να αξιολογήσουμε αποτελεσματικότητα των εργαλείων και τη διερεύνηση πιθανών παραγόντων που επηρεάζουν την απόδοση των ρομπότ σε διαφορετικά σενάρια.

Τέλος, θα συντάξουμε τα συμπεράσματα της μελέτης μας, αναδεικνύοντας τα πλεονεκτήματα και τους περιορισμούς των εργαλείων προσομοίωσης.

Καινοτομία

Σε αυτή τη διπλωματική εργασία αφενός αναλύονται 2 από τα πλέον δημοφιλή λογισμικά προσομοίωσης ρομποτικών συστημάτων, αφετέρου δε γίνεται η συγκριτική μελέτη τους.

Δομή

Η δομή της παρούσας διπλωματικής εργασίας οργανώνεται σε πέντε κύρια κεφάλαια, τα οποία αναλύουν αναλυτικά το θέμα της προσομοίωσης ρομπότ μέσω υπολογιστή.

Κεφάλαιο 1: CoppeliaSim. Αυτό το κεφάλαιο εστιάζει στο λογισμικό CoppeliaSim, αναλύοντας τις δυνατότητες και τα χαρακτηριστικά του. Περιλαμβάνει επίσης παραδείγματα εφαρμογών και μελέτες περίπτωσης που αναδεικνύουν τη χρησιμότητά του στην προσομοίωση ρομπότ.

Κεφάλαιο 2: Gazebo Στο τρίτο κεφάλαιο εξετάζεται το Gazebo, με έμφαση στις δυνατότητές του και την ευελιξία του στην προσομοίωση.

Κεφάλαιο 3: Ανάλυση Αποτελεσμάτων. Σε αυτό το κεφάλαιο συγκρίνονται οι επιδόσεις των δύο λογισμικών και εξάγονται συμπεράσματα σχετικά με την αποτελεσματικότητά τους.

ΚΕΦΑΛΑΙΟ 1^ο : CoppeliaSim

1.1 Εισαγωγή κεφαλαίου 1

Σε αυτό το κεφάλαιο αναλύεται το λογισμικό CoppeliaSim. Ο προσομοιωτής ρομποτικής CoppeliaSim, με ενσωματωμένο περιβάλλον ανάπτυξης, βασίζεται σε μια κατανεμημένη αρχιτεκτονική ελέγχου όπου κάθε αντικείμενο/μοντέλο μπορεί να ελεγχθεί ξεχωριστά μέσω ενός ενσωματωμένου σεναρίου, ενός plugin (προσθέτου), κόμβων ROS / ROS2, ενός απομακρυσμένου περιβάλλοντος διεπαφής ή μιας προσαρμοσμένης λύσης. Αυτό καθιστά το CoppeliaSim πολύ ευέλικτο και ιδανικό για πολύ-ρομποτικές εφαρμογές. Οι ελεγκτές μπορούν να γραφούν σε C/C++, Python, Java, Lua, Matlab ή Octave.

Ορισμένες από τις εφαρμογές του CoppeliaSim:

- Προσομοίωση συστημάτων αυτοματισμού εργοστασίων
- Απομακρυσμένη παρακολούθηση
- Έλεγχος υλικού
- Γρήγορη δημιουργία πρωτοτύπου και επαλήθευση
- Παρακολούθηση ασφάλειας
- Γρήγορη ανάπτυξη αλγορίθμων
- Εκπαίδευση σχετικά με τη ρομποτική
- Παρουσίαση προϊόντων

Το CoppeliaSim μπορεί να χρησιμοποιηθεί ως αυτόνομη εφαρμογή ή μπορεί εύκολα να ενσωματωθεί σε μια κύρια εφαρμογή πελάτη: το μικρό μέγεθος και η λεπτομερείς διασυνδέσεις εφαρμογών προγραμματισμού (API) καθιστούν το CoppeliaSim ιδανικό υποψήφιο για ενσωμάτωση σε υψηλότερο επίπεδο εφαρμογών. Ο ερμηνευτής σεναρίων Lua ή Python καθιστά το CoppeliaSim μια εξαιρετικά ευέλικτη εφαρμογή, δίνοντας την ελευθερία στο χρήστη να συνδυάσει τις λειτουργικότητες χαμηλού/υψηλού επιπέδου για την απόκτηση νέων λειτουργικοτήτων υψηλού επιπέδου.

Κύρια χαρακτηριστικά του CoppeliaSim:

- 5 Προγραμματιστικές προσεγγίσεις: (embedded scripts, plugins, add-ons, ROS nodes ή απομακρυσμένα API clients)
Ο προσομοιωτής και οι προσομοιώσεις είναι πλήρως προσαρμόσιμοι με 5 προγραμματιστικές προσεγγίσεις που είναι αμοιβαία συμβατές και μπορούν ακόμη να λειτουργήσουν από κοινού. Υποστηρίζονται 6 γλώσσες προγραμματισμού.
- Πανίσχυρα APIs, 7 γλώσσες:
Κανονικό API: Python, Lua και C/C++
Απομακρυσμένο API: C/C++, Python, Java, JavaScript, Matlab & Octave
Διεπαφές ROS: δημοσιοποιητές, συνδρομητές και κλήσεις υπηρεσίας. Υποστηρίζει όλα τα κανονικά μηνύματα και είναι επεκτάσιμο.
- Απομακρυσμένο API:
Μπορούμε να ελέγξουμε μια προσομοίωση ή τη προσομοίωση την ίδια απομακρυσμένα (π.χ από ένα πραγματικό ρομπότ ή ένα άλλο Ηλεκτρονικό Υπολογιστή).
- Δυναμική/Φυσική:

Προσομοίωση Ρομπότ μέσω Υπολογιστή

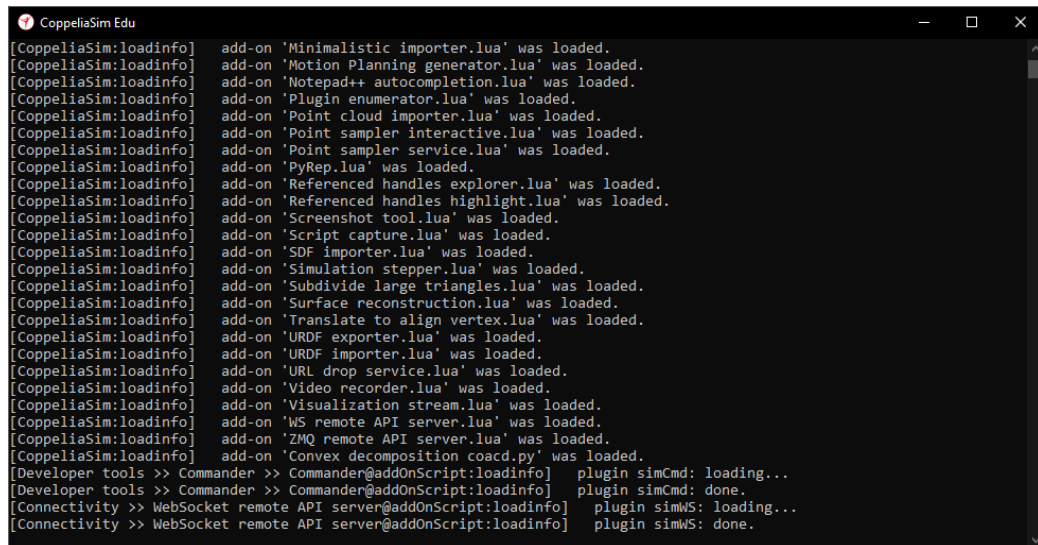
5 μηχανές φυσικής (MuJoCo, Bullet Physics, ODE, Vortex και Newton Dynamics) για γρήγορους και προσαρμοσμένους υπολογισμούς δυναμικής, για να προσομοιώσουν φυσική και αλληλεπιδράσεις αντικειμένων στον πραγματικό κόσμο (αντίδραση σε συγκρούσεις, λαβή κ.λπ.).

- Αντίστροφη/Προς τα εμπρός Κινηματική:
Υπολογισμοί αντίστροφης/προς τα εμπρός κινήματικής για οποιοδήποτε είδος μηχανισμού (κλαδωτός, κλειστός, περιττός, περιέχοντας ενσωματωμένους βρόγχους κλπ.).
- Μαλακά Σώματα:
Το CoppeliaSim υποστηρίζει μέσω της μηχανής MuJoCo μαλακά σώματα, κλωστές, σχοινιά, ρούχα, κ.λπ.
- Ανίχνευση συγκρούσεων
- Υπολογισμός ελάχιστης Απόστασης
- Cross-Platform και φορητό (Windows, MacOSX, Linux)
- Προσομοίωση Αισθητήρων προσέγγισης
- Προσομοίωση αισθητήρα όρασης
- Σχέδιο οικοδομικών μπλοκ
- Σχεδιασμός διαδρομής/κίνησης
- Καταγραφή και οπτικοποίηση δεδομένων (διαγράμματα χρόνου, X/Y διάγραμμα ή τρισδιάστατες καμπύλες)
- Προσαρμοσμένο περιβάλλον χρήστη
- Ενσωματωμένες λειτουργίες επεξεργασίας
- Εύκολη εισαγωγή/εξαγωγή δεδομένων
- RRS διεπαφή και βιβλιοθήκη κίνησης
- Ιεραρχία σκηνής με πλήρη χαρακτηριστικά
- Βολικός περιηγητής μοντέλων
- Πλήρης αλληλεπίδραση

1.2 Περιβάλλον Χρήστη

Η εφαρμογή CoppeliaSim αποτελείται από αρκετά στοιχεία. Τα βασικά της στοιχεία είναι:

- **Παράθυρο κονσόλας/Console window:**



```
CoppeliaSim Edu
[CoppeliaSim:loadinfo] add-on 'Minimalistic importer.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Motion Planning generator.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Notepad++ autocompletion.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Plugin enumerator.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Point cloud importer.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Point sampler interactive.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Point sampler service.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'PyRep.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Referenced handles explorer.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Referenced handles highlight.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Screenshot tool.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Script capture.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'SDF importer.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Simulation stepper.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Subdivide large triangles.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Surface reconstruction.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Translate to align vertex.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'URDF exporter.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'URDF importer.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'URL drop service.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Video recorder.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Visualization stream.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'WS remote API server.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'ZMQ remote API server.lua' was loaded.
[CoppeliaSim:loadinfo] add-on 'Convex decomposition coacd.py' was loaded.
[Developer tools >> Commander >> Commander@addOnScript:loadinfo] plugin simCmd: loading...
[Developer tools >> Commander >> Commander@addOnScript:loadinfo] plugin simCmd: done.
[Connectivity >> WebSocket remote API server@addOnScript:loadinfo] plugin simWS: loading...
[Connectivity >> WebSocket remote API server@addOnScript:loadinfo] plugin simWS: done.
```

Στα Linux, το CoppeliaSim ανοίγει καλώντας το executable αρχείο μέσω του τερματικού και παραμένει ορατό καθ' όλη τη διάρκεια της συνεδρίας του CoppeliaSim. Όταν ξεκινά η εφαρμογή, δημιουργείται ένα παράθυρο κονσόλας αλλά αμέσως εξαφανίζεται. Αυτό συμβαίνει από προεπιλογή και μπορεί να αλλάξει από τις ρυθμίσεις χρήστη. Το παράθυρο κονσόλας ή τερματικού εμφανίζει ποια πρόσθετα φορτώθηκαν και εάν η διαδικασία αρχικοποίησής τους ήταν επιτυχής. Το παράθυρο κονσόλας δεν είναι αλληλεπιδραστικό και χρησιμοποιείται μόνο για την εκτύπωση πληροφοριών. Ο χρήστης μπορεί να εκτυπώσει πληροφορίες απευθείας στο παράθυρο κονσόλας με την εντολή εκτύπωσης (από εντός ενός σεναρίου) ή με τις εντολές printf ή std::cout από εντός ενός προσθέτου. Επιπλέον, ο χρήστης μπορεί να δημιουργήσει προγραμματιστικά βοηθητικά παράθυρα κονσόλας για την εμφάνιση πληροφοριών που είναι για παράδειγμα συγκεκριμένες για μια προσομοίωση.

- **Παράθυρο εφαρμογής/Application window:**

Το παράθυρο της εφαρμογής είναι το κύριο παράθυρο της εφαρμογής. Χρησιμοποιείται για την εμφάνιση, την επεξεργασία, τη προσομοίωση και την αλληλεπίδραση με μια σκηνή. Το αριστερό και δεξί κουμπί του ποντικιού, η ροδέλα καθώς και το πληκτρολόγιο έχουν συγκεκριμένες λειτουργίες όταν ενεργοποιούνται στο παράθυρο της εφαρμογής. Εντός του παραθύρου της εφαρμογής, οι λειτουργίες των συσκευών εισόδου (ποντίκι και πληκτρολόγιο) μπορεί να ποικίλουν ανάλογα με το πλαίσιο ή τη θέση ενεργοποίησης.

- **Διάφοροι διάλογοι/Several dialogs:**

Δίπλα από το παράθυρο της εφαρμογής, ο χρήστης μπορεί επίσης να επεξεργαστεί και να αλληλεπιδράσει με μια σκηνή προσαρμόζοντας ρυθμίσεις διαλόγου ή παραμέτρων. Κάθε διάλογος ομαδοποιεί ένα σύνολο σχετικών λειτουργιών ή λειτουργιών που εφαρμόζονται σε ένα ίδιο στόχο. Το περιεχόμενο ενός διαλόγου ενδέχεται να εξαρτάται από το πλαίσιο (π.χ. εξαρτημένο από την κατάσταση επιλογής αντικειμένου).

- **Γραμμή εφαρμογής/Application bar:**

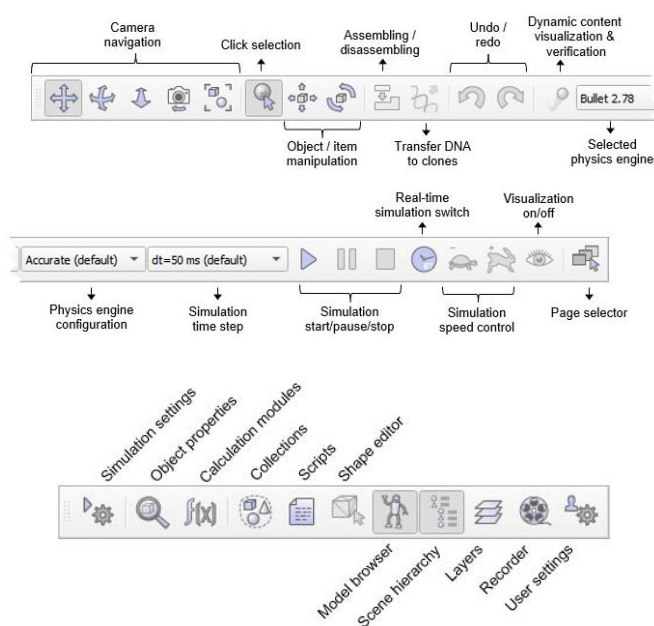
Η γραμμή εφαρμογής υποδηλώνει τον τύπο της άδειας του CoppeliaSim αντιγράφου μας, το όνομα του αρχείου της σκηνής που εμφανίζεται αυτή τη στιγμή, το χρόνο που χρησιμοποιείται για μια περάτωση αποτύπωσης (ένα πέρασμα προβολής), και την τρέχουσα κατάσταση του προσομοιωτή (κατάσταση προσομοίωσης ή τύπο της ενεργής λειτουργίας επεξεργασίας). Η γραμμή εφαρμογής, καθώς και οποιαδήποτε επιφάνεια μέσα στο παράθυρο της εφαρμογής, μπορούν επίσης να χρησιμοποιηθούν για το σύρσιμο και απόθεση αρχείων που σχετίζονται με το CoppeliaSim, στη σκηνή. Τα υποστηριζόμενα αρχεία περιλαμβάνουν "*.tnt" αρχεία (αρχεία σκηνής CoppeliaSim) και "*.tmm" αρχεία (αρχεία μοντέλου CoppeliaSim).

- **Γραμμή μενού/Menu bar:**

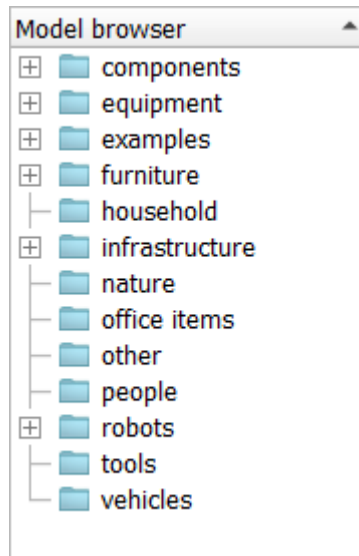
Η γραμμή μενού επιτρέπει την πρόσβαση σχεδόν σε όλες τις λειτουργίες του προσομοιωτή. Τις περισσότερες φορές, τα στοιχεία στη γραμμή μενού ενεργοποιούν έναν διάλογο. Το περιεχόμενο της γραμμής μενού εξαρτάται από τη συγκεκριμένη κατάσταση του προσομοιωτή. Οι περισσότερες λειτουργίες στη γραμμή μενού μπορούν εναλλακτικά να έχουν πρόσβαση μέσω ενός αναδυόμενου μενού, ενός διπλού κλικ σε ένα εικονίδιο στην προβολή ιεραρχίας σκηνής ή μέσω ενός κλικ σε ένα κουμπί εργαλείων.

- **Εργαλεία/Toolbars:**

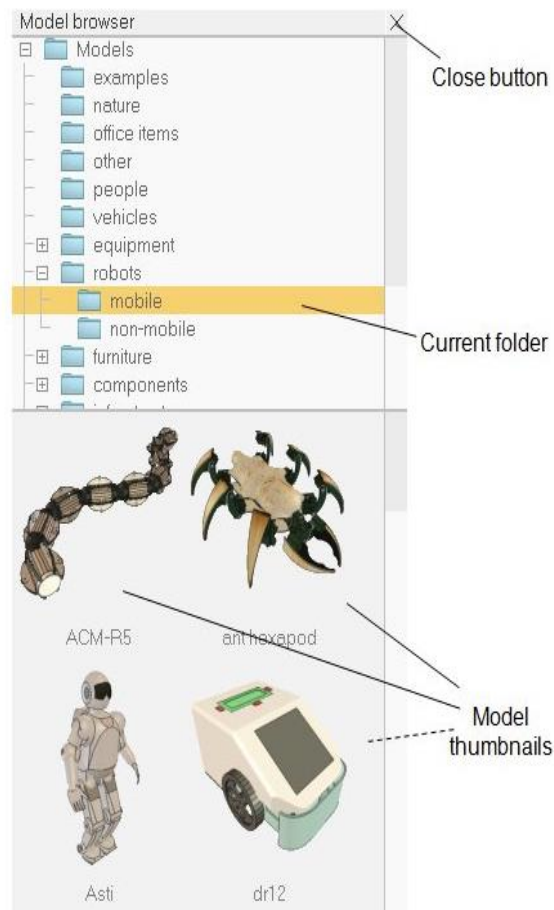
Τα εργαλεία παρέχουν λειτουργίες που συχνά είναι προσβάσιμες (π.χ. αλλαγή λειτουργίας πλοήγησης, επιλογή άλλης σελίδας, κλπ.). Κάποιες λειτουργίες στην πρώτη γραμμή εργαλείων, καθώς και όλες οι λειτουργίες στη δεύτερη γραμμή εργαλείων, μπορούν επίσης να έχουν πρόσβαση μέσω της γραμμής μενού ή του αναδυόμενου μενού. Και οι δύο γραμμές εργαλείων μπορούν να δεσμευτούν και να απελευθερωθούν, αλλά η δέσμευση λειτουργεί μόνο με τις αρχικές τους θέσεις. Οι παρακάτω εικόνες εξηγούν τη λειτουργία κάθε κουμπιού στις γραμμές εργαλείων:



- **Περιηγητής μοντέλου/Model browser:**

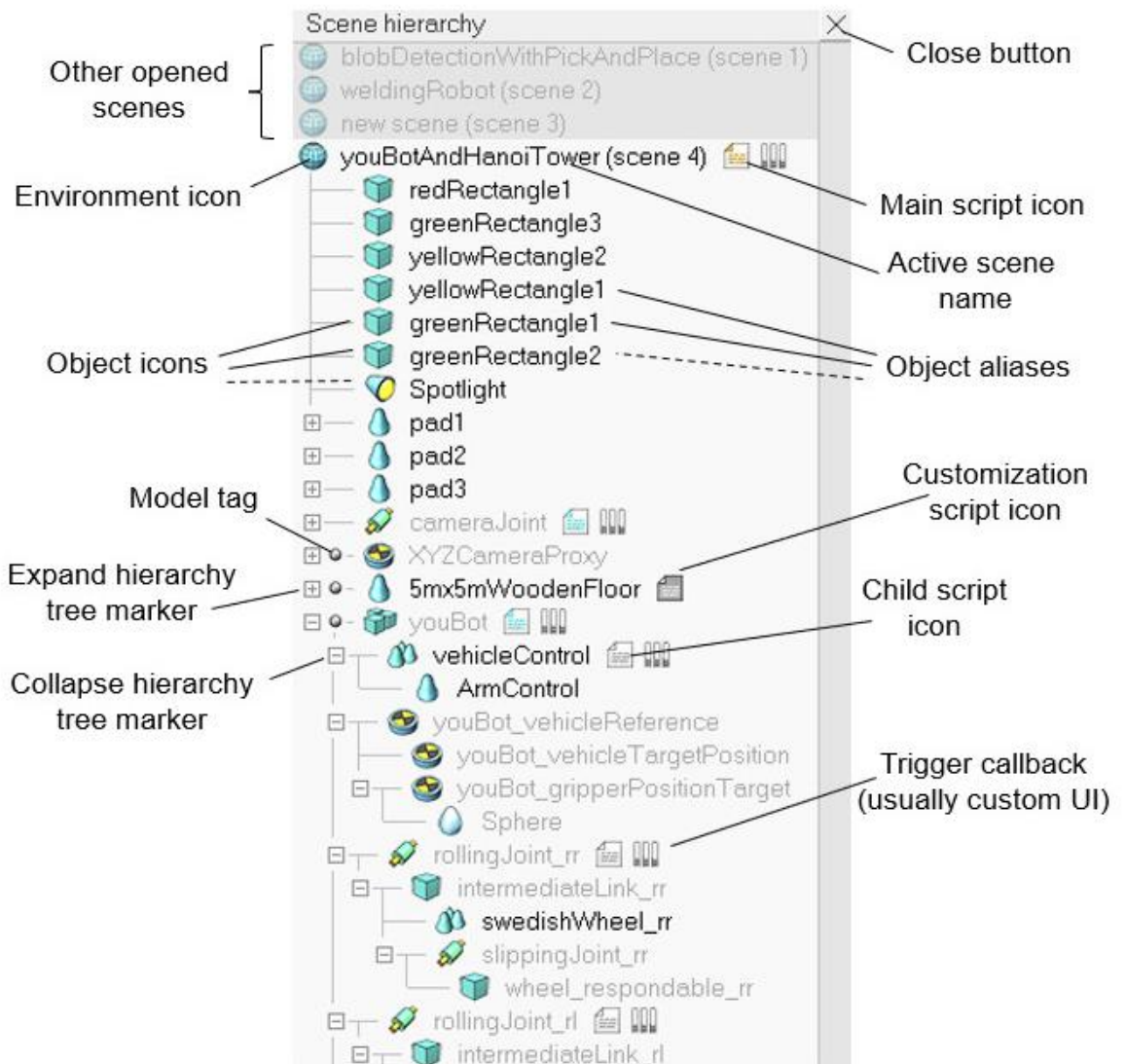


Ο περιηγητής μοντέλων είναι ορατός από προεπιλογή, αλλά μπορεί να εναλλάσσεται με το αντίστοιχο κουμπί της γραμμής εργαλείων. Εμφανίζει στο επάνω μέρος μια δομή φακέλων μοντέλων του CorreliaSim και στο κάτω μέρος εμφανίζει μια μικρογραφία των μοντέλων που περιέχονται στον επιλεγμένο φάκελο. Οι μικρογραφίες μπορούν να σύρονται και να αποθέτονται στη σκηνή για να φορτώνεται αυτόματα το σχετικό μοντέλο. Οι συλληφθείσες μικρογραφίες εμφανίζονται σκούρες εάν η περιοχή απόθεσης δεν υποστηρίζεται ή δεν είναι κατάλληλη.



- **Ιεραρχία σκηνής/Scene hierarchy:**

Η ιεραρχία της σκηνής είναι ορατή από προεπιλογή, αλλά μπορεί να εναλλάσσεται με το αντίστοιχο κουμπί των εργαλείων της. Εμφανίζει το περιεχόμενο μιας σκηνής (δηλαδή όλα τα αντικείμενα σκηνής που αποτελούν μια σκηνή). Δεδομένου ότι τα αντικείμενα σκηνής είναι δομημένα σε μια ιεραρχική δομή, η ιεραρχία της σκηνής εμφανίζει ένα δέντρο αυτής της ιεραρχίας και τα μεμονωμένα στοιχεία μπορούν να διευρυνθούν ή να συμπυκνωθούν. Ένα διπλό κλικ σε ένα εικονίδιο ανοίγει/κλείνει ένα διάλογο ιδιοτήτων που σχετίζεται με το "κλικαρισμένο" εικονίδιο. Ένα διπλό κλικ σε ένα ψευδώνυμο αντικειμένου επιτρέπει την επεξεργασία του. Η ροδέλα κύλισης του ποντικιού, καθώς και ένα σύρσιμο των γραμμών κύλισης της προβολής της ιεραρχίας της σκηνής, επιτρέπει τη μετακίνηση του περιεχομένου πάνω/κάτω ή αριστερά/δεξιά. Η επιλογή με το πλήκτρο Control και Shift υποστηρίζεται πάντα. Τα αντικείμενα στην ιεραρχία της σκηνής μπορούν να συρθούν και να αποτεθούν σε ένα άλλο αντικείμενο, προκειμένου να δημιουργηθεί μια σχέση γονιού-παιδιού. Η ιεραρχία της σκηνής θα εμφανίσει διαφορετικό περιεχόμενο εάν ο προσομοιωτής βρίσκεται σε κατάσταση επεξεργασίας.



- **Σελίδα/page:**

Κάθε σκηνή μπορεί να περιέχει μέχρι και 8 σελίδες, καθεμία από αυτές μπορεί να περιλαμβάνει έναν απεριόριστο αριθμό προβολών. Μια σελίδα μπορεί να θεωρηθεί ως δοχείο για προβολές.

- **Όψεις/views:**

Μπορεί να υπάρχει απεριόριστος αριθμός προβολών που περιέχονται σε μια σελίδα. Μια προβολή χρησιμοποιείται για να εμφανίσει τη σκηνή (την ίδια περιλαμβάνοντας ένα περιβάλλον και αντικείμενα), βλέποντας μέσω ενός ορατού αντικειμένου (π.χ. κάμερες, γραφήματα ή αισθητήρες όρασης).

- **Κείμενο πληροφοριών/information text:**

Το κείμενο πληροφοριών εμφανίζει πληροφορίες που σχετίζονται με την τρέχουσα επιλογή αντικειμένου/στοιχείου και με την κατάσταση ή τις παραμέτρους της τρέχουσας προσομοίωσης. Η εμφάνιση του κειμένου μπορεί να εναλλάσσεται με ένα από τα δύο μικρά κουμπιά στην επάνω αριστερή πλευρά μιας σελίδας. Το άλλο κουμπί μπορεί να χρησιμοποιηθεί για να εναλλάξει ένα λευκό φόντο, προσφέροντας καλύτερη αντίθεση ανάλογα με το χρώμα φόντου μιας σκηνής.

- **Γραμμή κατάστασης/status bar:**

Η γραμμή κατάστασης εμφανίζει πληροφορίες που σχετίζονται με εκτελεσθείσες λειτουργίες, εντολές και επίσης εμφανίζει μηνύματα σφαλμάτων από ερμηνευτές σεναρίων. Από εντός ενός σεναρίου, ο χρήστης μπορεί επίσης να εξάγει αλφαριθμητικά στη γραμμή κατάστασης ή την κονσόλα με τη συνάρτηση *sim.addLog*. Η γραμμή κατάστασης εμφανίζει μόνο δύο γραμμές από προεπιλογή, αλλά μπορεί να αλλάξει μέγεθός χρησιμοποιώντας την οριζόντια χειρολαβή διαχωρισμού της.

- **Γραμμή εντολών/Commander:**

Είναι ένας βρόχος ανάγνωσης-εκτέλεσης-εκτύπωσης, που προσθέτει κείμενο εισόδου στη γραμμή κατάστασης του CorreliaSim, επιτρέποντας την εισαγωγή και εκτέλεση κώδικα Python ή Lua κατά τη διάρκεια της εκτέλεσης, όπως σε ένα τερματικό. Ο κώδικας μπορεί να εκτελεστεί στο sandbox σενάριο ή σε οποιοδήποτε άλλο ενεργό σενάριο στο CorreliaSim.

- **Προσαρμοσμένες διεπαφές χρήστη/custom user interfaces:**

Οι προσαρμοσμένες διεπαφές χρήστη είναι επιφάνειες διεπαφής που καθορίζονται από τον χρήστη και μπορούν να χρησιμοποιηθούν για την εμφάνιση πληροφοριών (κείμενο, εικόνες, κλπ.) ή ενός προσαρμοσμένου διαλόγου, επιτρέποντας την αλληλεπίδραση με τον χρήστη με ένα προσαρμοσμένο τρόπο.

- **Αναδυόμενο μενού/pop-up menu:**

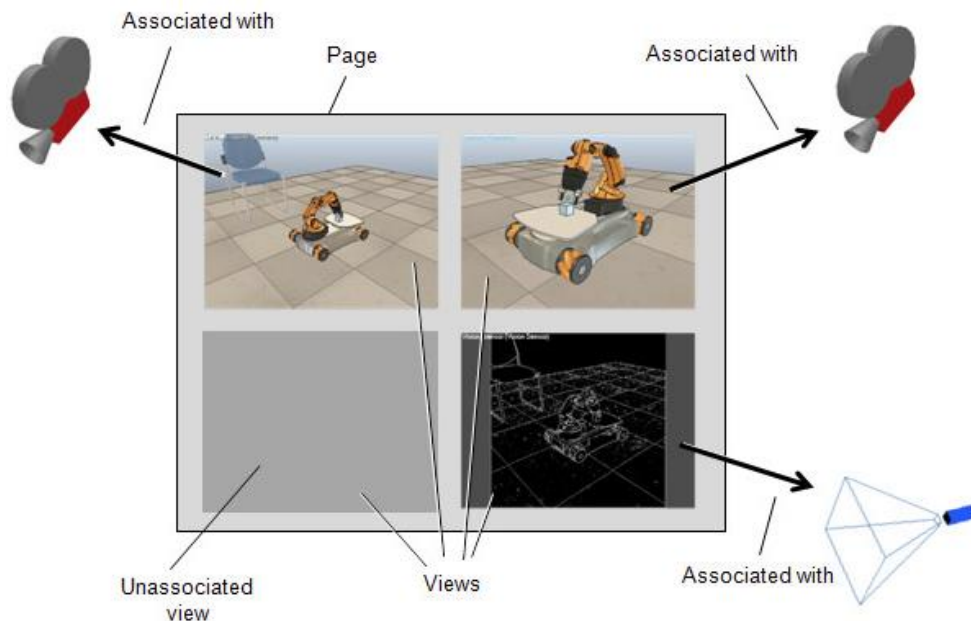
Τα αναδυόμενα μενού είναι τα μενού που εμφανίζονται μετά από ένα δεξί κλικ του ποντικιού. Για να ενεργοποιήσουμε ένα αναδυόμενο μενού, βεβαιωνόμαστε ότι το ποντίκι δεν κινείται κατά τη διάρκεια της λειτουργίας κλικ, διαφορετικά ενδέχεται να ενεργοποιηθεί η λειτουργία

Προσομοίωση Ρομπότ μέσω Υπολογιστή

περιστροφής της κάμερας. Κάθε επιφάνεια μέσα στο παράθυρο εφαρμογής (π.χ. προβολή ιεραρχίας σκηνής, σελίδα, προβολή κλπ.) μπορεί να ενεργοποιήσει ένα διαφορετικό αναδυόμενο μενού (είναι ευαίσθητο στο πλαίσιο). Το περιεχόμενο των αναδυόμενων μενού μπορεί επίσης να αλλάξει ανάλογα με την τρέχουσα κατάσταση προσομοίωσης ή τη λειτουργία επεξεργασίας. Το μεγαλύτερο μέρος της λειτουργίας του αναδυόμενου μενού μπορεί επίσης να είναι προσβάσιμο μέσω της γραμμής μενού, εκτός από το στοιχείο προβολής μενού που εμφανίζεται μόνο όταν το αναδυόμενο μενού ενεργοποιείται σε μια προβολή ή σελίδα.

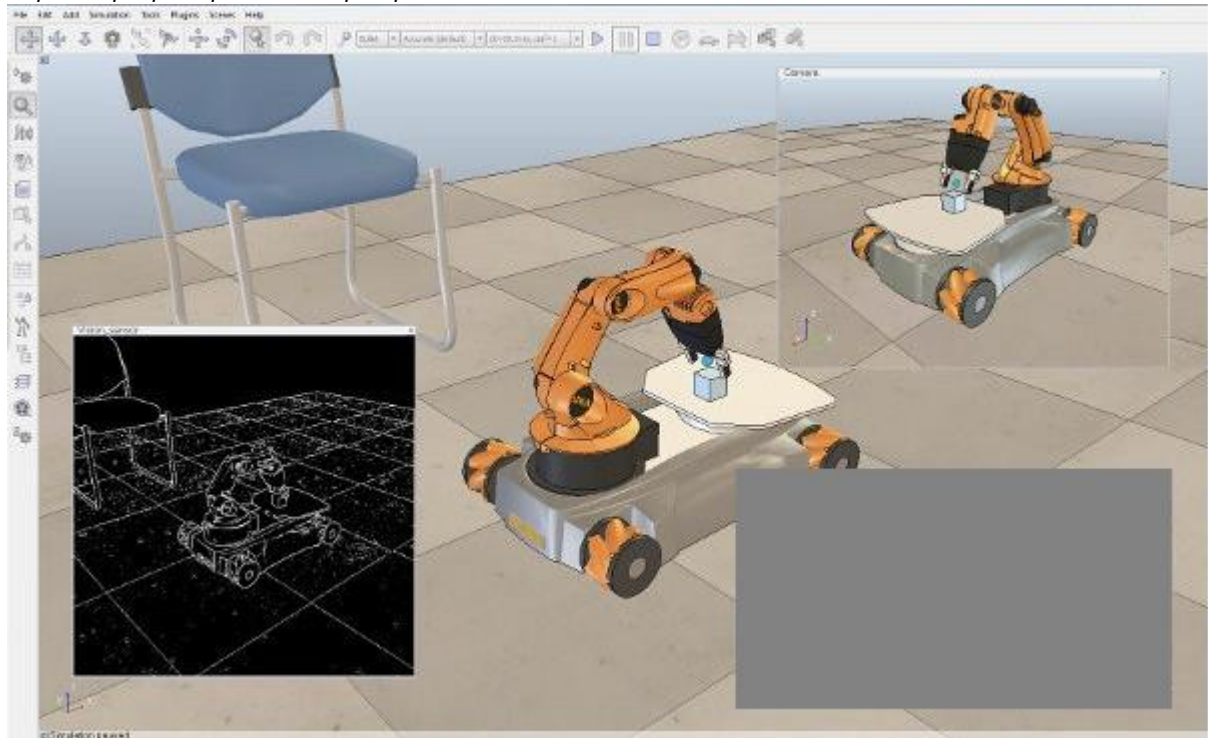
1.3 Σελίδες και όψεις

Μια σελίδα στο CoppeliaSim είναι η κύρια επιφάνεια προβολής μιας σκηνής. Μπορούμε να χρησιμοποιήσουμε μία, δύο ή όσες προβολές χρειαζόμαστε για να παρακολουθήσουμε ένα αντικείμενο. Βασική προϋπόθεση είναι το αντικείμενο να είναι αντικείμενο που μπορεί να προβληθεί. Εάν μια προβολή συνδέεται με ένα αντικείμενο κάμερας, για παράδειγμα, τότε μπορεί να εμφανίσει ό,τι βλέπει η κάμερα. Η παρακάτω εικόνα εξηγεί τη σχέση μεταξύ σελίδας, προβολής και αντικειμένου που μπορεί να προβληθεί:



Μια προβολή μπορεί να έχει σταθερή θέση μέσα σε μια σελίδα ή μπορεί να έχει αιωρούμενη θέση πάνω στη σελίδα. Η παραπάνω σχέση σελίδας, προβολής και αντικειμένου που μπορεί να προβληθεί απεικονίζεται στην παρακάτω διαμόρφωση σελίδας:

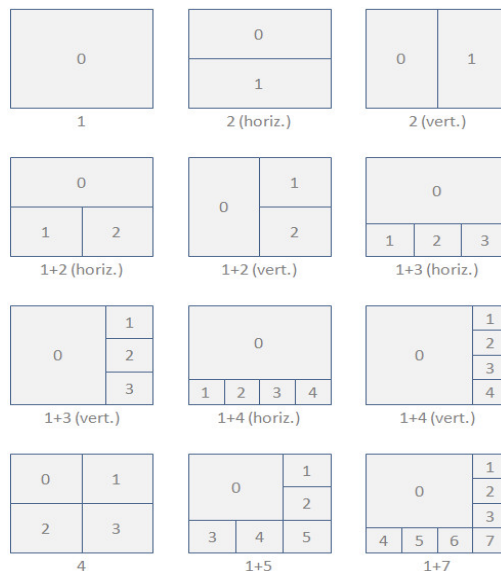
Προσομοίωση Ρομπότ μέσω Υπολογιστή



Κάθε σκηνή στο CoppeliaSim έχει οκτώ σελίδες τις οποίες μπορούμε να διαμορφώσουμε ελεύθερα. Οι επιμέρους σελίδες μπορούν να εμφανιστούν μέσω του κουμπιού της γραμμής εργαλείων επιλογής σελίδας (page selector) είτε κάνοντας δεξί κλικ στη scene και στο popup menu → view → view selector :

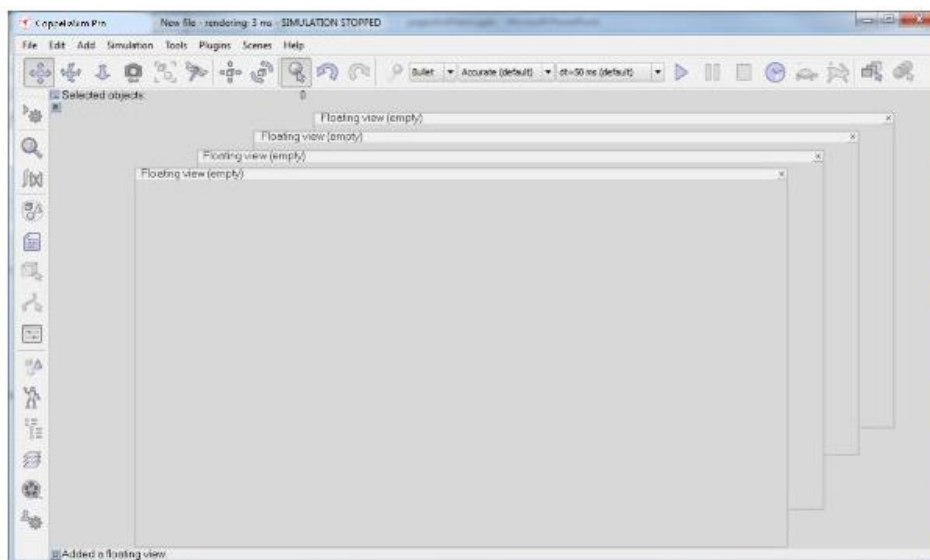


Όταν δημιουργείται μια νέα σκηνή, τότε κάθε μία από τις 8 σελίδες θα είναι προδιαμορφωμένη με διαφορετικό τρόπο. Μια σελίδα μπορεί να αφαιρεθεί με [Popup menu → Remove Page]. Μια ανύπαρκτη σελίδα (δηλαδή αφαιρεθείσα σελίδα) εμφανίζει μια επιφάνεια σκούρου γκρι. Μπορούμε να δημιουργήσουμε μια σελίδα και μια προεπιλεγμένη διαμόρφωση σελίδας με κενές προβολές με [Popup menu → Setup-Page With...]. Διατίθενται διάφορες διαμορφώσεις σελίδων, όπως φαίνεται στην παρακάτω εικόνα (οι αριθμοί υποδεικνύουν τους δείκτες προβολής):



Προσομοίωση Ρομπότ μέσω Υπολογιστή

Οι παραπάνω διαμορφώσεις σελίδων επιτρέπουν την εμφάνιση 1-8 σταθερών προβολών. Καθεμία από τις επιφάνειες ανοιχτού γκρι αντιστοιχεί σε μια κενή προβολή. Ανά πάσα στιγμή, μια σελίδα μπορεί να αφαιρεθεί με [Pop-up menu → Remove Page]. Η αφαίρεση μιας σελίδας θα αφαιρέσει επίσης όλες τις προβολές που περιέχει, αλλά δεν θα αφαιρέσει κανένα από τα συσχετισμένα αντικείμενα. Πάνω σε μια υπάρχουσα διαμόρφωση σελίδας, είναι δυνατόν να προσθέσουμε έναν απεριόριστο αριθμό αιωρούμενων προβολών με [Pop-up menu → Add → Floating View]. Η παρακάτω εικόνα δείχνει ένα τέτοιο παράδειγμα:



Μπορούμε να μετακινήσουμε και να αλλάξουμε μέγεθος στις αιωρούμενες προβολές αλλά δεν επιτρέπεται η πλοήγηση με το αριστερό ή δεξί κλικ (π.χ. περιστροφές κάμερας κ.λπ.). Αν κάνουμε διπλό κλικ σε μια αιωρούμενη προβολή μπορούμε να ανταλλάξουμε το περιεχόμενο της με τη προβολή δείκτη 0 (αυτή η ενέργεια μπορεί ωστόσο να απαγορευτεί απενεργοποιώντας την προγραμματιστικά). Για να αφαιρέσουμε μια αιωρούμενη προβολή, απλώς κάνουμε κλικ στο κουμπί της στην επάνω δεξιά πλευρά της.

Για να συσχετίσουμε ένα αντικείμενο που μπορεί να προβληθεί με μια προβολή, επιλέγουμε το αντικείμενο και στην προβολή που θέλουμε να συσχετίσουμε κάνουμε κλικ στο [Pop-up menu → View → Associate view with selected camera] ή [Pop-up menu → View → Associate view with selected version sensor] (το αναδυόμενο μενού θα προσαρμόσει αυτόματα το περιεχόμενό του ανάλογα με το τελευταίο επιλεγμένο αντικείμενο). Όταν δημιουργηθεί μια προβολή, αλλά δεν έχει συσχετιστεί ακόμη με ένα αντικείμενο που μπορεί να προβληθεί, η εντολή [Pop-up menu → Add → Camera] θα προσθέσει μια κάμερα και θα την συσχετίσει άμεσα με την προβολή (δηλαδή, θα δείχνει μέσω αυτής). Ένα αντικείμενο που μπορεί να προβληθεί μπορεί να συσχετιστεί με οποιονδήποτε αριθμό προβολών ταυτόχρονα.

Εναλλακτικά, μπορούμε επίσης να συσχετίσουμε μια προβολή με ένα αντικείμενο που μπορεί να προβληθεί ενεργοποιώντας το παρακάτω αναδυόμενο μενού σε μια προβολή: [Pop-up menu → View → View selector...]. Αυτό θα ανοίξει τον επιλογέα προβολής.

(Μόλις ένα αντικείμενο προβολής συσχετιστεί με μια προβολή, το περιεχόμενο της εικόνας του θα εμφανιστεί, εκτός από τους αισθητήρες όρασης που λειτουργούν με τον ακόλουθο τρόπο: ένας αισθητήρας όρασης πρέπει να δημιουργήσει το περιεχόμενο της εικόνας του, και αυτό μπορεί να

Προσομοίωση Ρομπότ μέσω Υπολογιστή

συμβεί μόνο όταν η κατάλληλη εντολή API κληθεί. Από προεπιλογή, το κύριο σενάριο περιέχει μια εντολή που θα χειρίζεται όλους τους αισθητήρες όρασης στη σκηνή (sim.handleVisionSensor(sim.handle_all)). Εκτός εάν ένας αισθητήρας όρασης επισημαίνεται για ρητή χειρισμό αλλά δεν χειρίζεται ρητά, τότε το περιεχόμενο της εικόνας του θα δημιουργείται κατά τη διάρκεια της προσομοίωσης.)

1.3.1 Διαχείριση θέσης/Προσανατολισμού

Μπορούμε να τροποποιήσουμε και να διαχειριστούμε τη θέση και το προσανατολισμό ενός αντικειμένου ή χρησιμοποιώντας ένα από τα παρακάτω κουμπιά της γραμμής εργαλείων:



Το πρώτο εμφανίζει το διάλογο θέσης (position dialog), ενώ το δεύτερο εμφανίζει το διάλογο προσανατολισμού (orientation dialog). Και στις δύο περιπτώσεις, μπορούμε επίσης να μετακινήσουμε τα επιλεγμένα αντικείμενα ή στοιχεία με το ποντίκι.

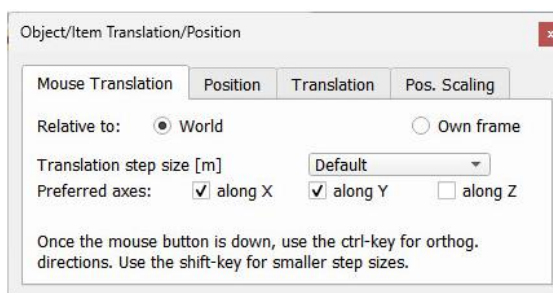
1.3.2 Διαχείριση Θέσης

Ο διάλογος θέσης γίνεται ορατός όταν επιλέγεται το κουμπί της γραμμής εργαλείων μετάφρασης αντικειμένου:



Ο διάλογος έχει τέσσερις διακριτές καρτέλες:

1.3.3 Mouse Translation



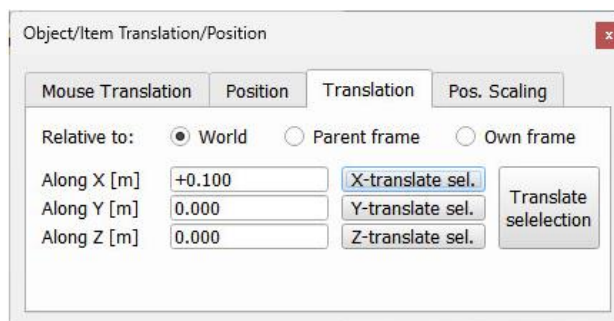
- **Relative to world/own frame:** υποδεικνύει ότι ένα σύρσιμο του ποντικιού θα μεταφράσει το επιλεγμένο αντικείμενο σε ένα επίπεδο ή γραμμή που ευθυγραμμίζεται με το απόλυτο σύστημα αναφοράς ή ευθυγραμμίζεται με το δικό του σύστημα αναφοράς.
- **Translation step size:** το μέγεθος του βήματος που χρησιμοποιείται κατά τη μετάφραση του επιλεγμένου αντικειμένου με ένα σύρσιμο του ποντικιού (δείτε το διάλογο ρυθμίσεων χρήστη για τα προεπιλεγμένα μεγέθη βήματος). Μικρότερα μεγέθη βήματος μπορούν να χρησιμοποιηθούν κατά τη διάρκεια της διαχείρισης, πατώντας το πλήκτρο shift αφού έχει πατηθεί το κουμπί του ποντικιού.
- **Preferred axes: along X/ along Y/ along Z:** υποδεικνύει ότι ένα σύρσιμο του ποντικιού επιτρέπει τη μετάφραση του επιλεγμένου αντικειμένου κατά μήκος των προτιμώμενων αξόνων του συστήματος αναφοράς που επιλέχθηκε παραπάνω. Άλλοι άξονες μπορούν να χρησιμοποιηθούν κατά τη διάρκεια της διαχείρισης, πατώντας το πλήκτρο ctrl αφού έχει πατηθεί το κουμπί του ποντικιού.

1.3.4 Position



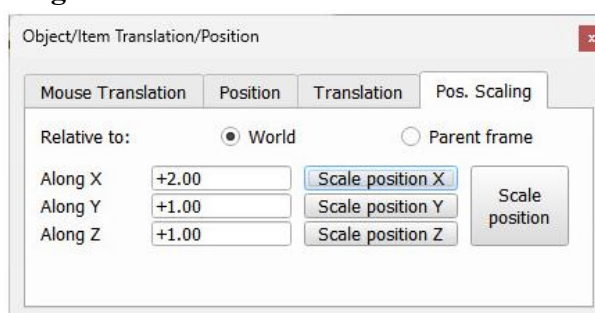
- **Relative to world/parent frame:** υποδεικνύει ότι οι συντεταγμένες είναι σχετικές με το απόλυτο σύστημα αναφοράς ή σχετικές με το σύστημα αναφοράς του γονέα.
- **X-/ Y- / Z-coordinate:** η θέση του επιλεγμένου αντικειμένου σε σχέση με το υποδεικνυόμενο σύστημα αναφοράς (παγκόσμιο ή γονικό).

1.3.5 Translation



- **Relative to world/parent frame /own frame:** υποδεικνύει ότι η μετάφραση θα είναι σχετική με το απόλυτο σύστημα αναφοράς, σχετική με το σύστημα αναφοράς του γονέα ή σχετική με το σύστημα αναφοράς του ίδιου του αντικειμένου.
- **Μετατόπιση κατά X / Y / Z:** υποδεικνύει τα επιθυμητά ποσά μετάφρασης κατά μήκος του άξονα x, y και z του υποδεικνυόμενου συστήματος αναφοράς (παγκόσμιο, γονικό ή δικό του πλαίσιο).

1.3.6 Position scaling



- **Relative to world/parent frame:** υποδεικνύει ότι η κλιμάκωση της θέσης θα είναι σχετική με το απόλυτο σύστημα αναφοράς ή σχετική με το σύστημα αναφοράς του γονέα.
- **Κλιμάκωση κατά X / Y / Z:** υποδεικνύει την επιθυμητή κλιμάκωση θέσης κατά μήκος του άξονα x, y και z του υποδεικνυόμενου συστήματος αναφοράς (παγκόσμιο ή γονικό).

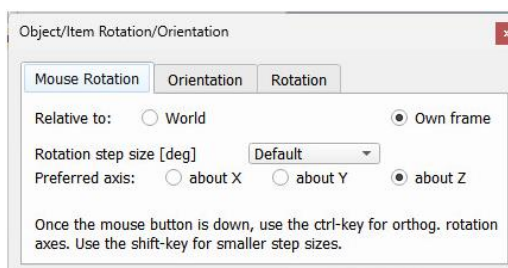
1.3.7 Orientation dialog/Διάλογος προσανατολισμού

Ο διάλογος προσανατολισμού γίνεται ορατός όταν επιλέγεται το κουμπί εργαλείων περιστροφής αντικειμένου.



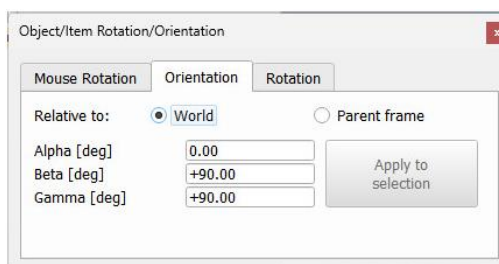
Ο διάλογος έχει τρεις διακριτές καρτέλες:

1.3.8 Mouse Rotation



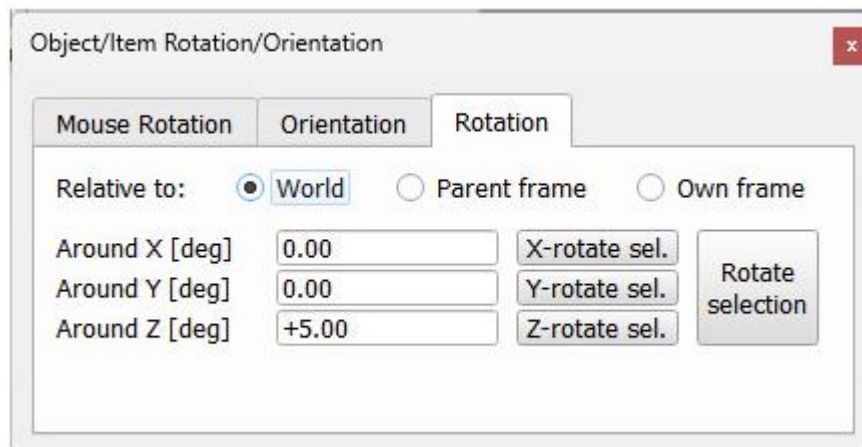
- **Relative to world/own frame:** υποδεικνύει ότι ένα σύρσιμο του ποντικιού θα περιστρέψει το επιλεγμένο αντικείμενο γύρω από έναν άξονα του απόλυτου συστήματος αναφοράς ή το δικό του σύστημα αναφοράς.
- **Rotation step size:** το μέγεθος του βήματος που χρησιμοποιείται κατά την περιστροφή του επιλεγμένου αντικειμένου με σύρσιμο του ποντικιού. Μικρότερα μεγέθη βήματος μπορούν να χρησιμοποιηθούν κατά τη διάρκεια του χειρισμού πιέζοντας το πλήκτρο shift μετά το πάτημα του κουμπιού του ποντικιού.
- **Preferred axis: about X/ about Y/ about Z:** υποδεικνύει ότι ένα σύρσιμο του ποντικιού επιτρέπει την περιστροφή του επιλεγμένου αντικειμένου γύρω από έναν προτιμώμενο άξονα του αναφερόμενου συστήματος αναφοράς. Άλλοι άξονες μπορούν να χρησιμοποιηθούν κατά το χειρισμό πιέζοντας το πλήκτρο ctrl μετά το πάτημα του κουμπιού του ποντικιού.

1.3.9 Orientation



- **Relative to world/parent frame:** υποδεικνύει ότι οι επιλεγμένες γωνίες Euler είναι σχετικές με το απόλυτο σύστημα αναφοράς ή σχετικές με το γονικό σύστημα αναφοράς.
- **Alpha / Beta / Gamma:** Γωνίες Euler του επιλεγμένου αντικειμένου σχετικά με το αναφερόμενο σύστημα αναφοράς (παγκόσμιο ή γονικό).

1.3.10 Rotation



- **Relative to world/parent frame/own frame:** υποδεικνύει ότι η περιστροφή θα είναι σχετική με το απόλυτο σύστημα αναφοράς, σχετική με το γονικό σύστημα αναφοράς, ή σχετική με το ίδιο σύστημα αναφοράς του αντικειμένου.
- **Rotate around X / Y / Z:** υποδεικνύει τις επιθυμητές ποσότητες περιστροφής γύρω από τους άξονες x-, y- και z- του αναφερόμενου συστήματος αναφοράς (παγκόσμιο, γονικό ή ίδιο πλαίσιο).

1.3.11 Object movement with the mouse

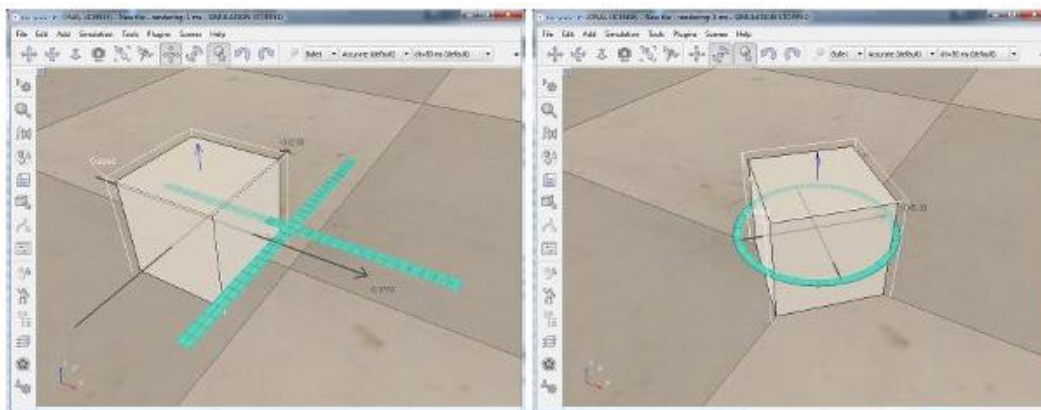
Πλην της τροποποίησης των ρυθμίσεων του αντικειμένου μέσω του διαλόγου συντεταγμένων και μετασχηματισμού, τα αντικείμενα μπορούν επίσης να τροποποιηθούν απευθείας χρησιμοποιώντας το ποντίκι: όταν επιλέγονται αντικείμενα (και μερικά αντικείμενα), τότε μπορούν να μετακινηθούν ή να περιστραφούν χρησιμοποιώντας τα ακόλουθα κουμπιά της γραμμής εργαλείων:



Από προεπιλογή, το translation πραγματοποιείται στο επίπεδο X-Y του κόσμου. Μπορεί επίσης να γίνει και σε διαφορετικό άξονα/επίπεδο και σχετικά με διαφορετικό αναφορικό πλαίσιο για κάθε ξεχωριστό αντικείμενο, ανάλογα με τις ρυθμίσεις στον διάλογο θέσης. Ένα κυανό overlay σχέδιο δείχνει το τρέχον επίπεδο ή άξονα μετάφρασης.

Προεπιλογή, το rotation γίνεται γύρω από τον ίδιο τον άξονα Z του αντικειμένου. Μπορεί επίσης να γίνει γύρω από διαφορετικό άξονα ή σχετικά με διαφορετικό αναφορικό πλαίσιο για κάθε ξεχωριστό αντικείμενο, ανάλογα με τις ρυθμίσεις στον διάλογο προσανατολισμού. Ένα κυανό overlay σχέδιο δείχνει το τρέχον άξονα περιστροφής.

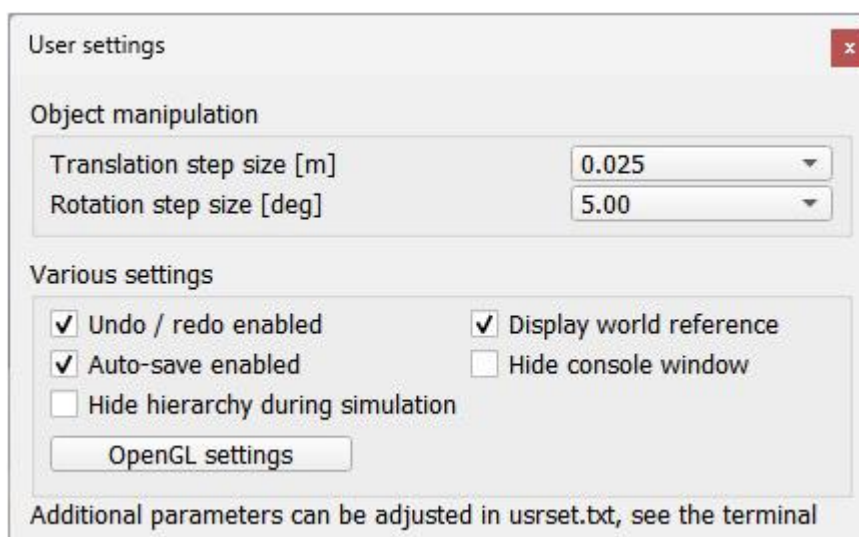
Προσομοίωση Ρομπότ μέσω Υπολογιστή



Οι προεπιλεγμένες τιμές βήματος των translation και rotation (snapping) μπορούν να προσαρμοστούν στον διάλογο ρυθμίσεων χρήστη, αλλά συνιστάται να διατηρηθούν οι τιμές των 5 εκατοστών και 5 βαθμών αντίστοιχα, ή να ρυθμιστούν οι τιμές βήματος για τα αντικείμενα μεμονωμένα. Το snapping μπορεί να απενεργοποιηθεί προσωρινά κρατώντας το πλήκτρο shift πατημένο META το πάτημα του πλήκτρου του ποντικιού κατά τη διάρκεια της μετακίνησης του αντικειμένου. Με παρόμοιο τρόπο, εναλλακτικοί άξονες translation ή rotation μπορούν να ενεργοποιηθούν προσωρινά κρατώντας το πλήκτρο ctrl πατημένο μετά το πάτημα του πλήκτρου του ποντικιού κατά τη διάρκεια της μετακίνησης του αντικειμένου.

1.3.12 User Settings/Ρυθμίσεις χρήστη

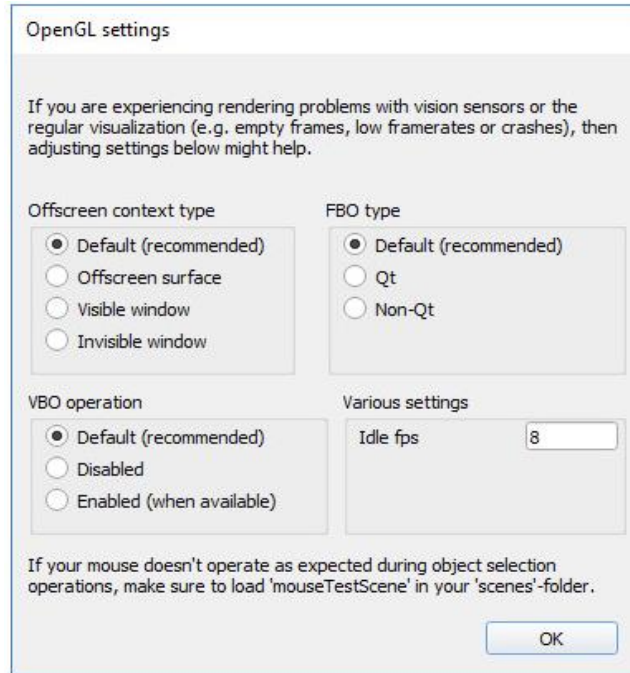
Κάποιες τιμές και ρυθμίσεις στο CoppeliaSim δεν εξαρτώνται από μια σκηνή ή ένα μοντέλο, αλλά από τον χρήστη. Οι ρυθμίσεις χρήστη βρίσκονται στο [Tools → Settings] ή κάνοντας κλικ στο ακόλουθο κουμπί εργαλειοθήκης:



- **Translation step size:** Γραμμικό βήμα μετάφρασης που χρησιμοποιείται κατά τη μετάφραση αντικειμένων στη λειτουργία διαχείρισης αντικειμένων. Συνιστάται να διατηρείτε τιμή 5 εκατοστά. Τα αντικείμενα μπορούν να έχουν συγκεκριμένα βήματα στον διάλογο συντεταγμένων και μετασχηματισμών.

- **Rotation step size:** Γωνιακό βήμα περιστροφής που χρησιμοποιείται κατά την περιστροφή αντικειμένων στη λειτουργία διαχείρισης αντικειμένων. Τα αντικείμενα μπορούν να έχουν συγκεκριμένα βήματα στον διάλογο συντεταγμένων και μετασχηματισμών.
- **Undo/redo enabled:** Ενεργοποίηση ή απενεργοποίηση της λειτουργικότητας αναίρεσης/επανάληψης. Αυτή η λειτουργία λειτουργεί με το να αποθηκεύει (σειριοποιεί) ολόκληρη τη σκηνή στη μνήμη κάθε φορά που καταγράφεται μια αλλαγή. Μνημονεύονται μόνο οι διαφορές με τα προηγούμενα σημεία αναίρεσης, για να χρησιμοποιεί λίγη μνήμη. Αυτή είναι μια πολύ αποτελεσματική και ασφαλής λειτουργία αναίρεσης/επανάληψης που υποστηρίζεται επίσης από πρόσθετα ή την κύρια εφαρμογή πελάτη, και επιτρέπεται μόνο επειδή οι διαδικασίες σειριοποίησης του CoppeliaSim είναι πολύ γρήγορες. Ωστόσο, υπάρχουν και στιγμές όταν ο υπολογιστής είναι πραγματικά παλιός ή το περιεχόμενο της σκηνής είναι εξαιρετικά μεγάλο (π.χ. πολύ λεπτομερή δεδομένα CAD) όπου αυτή η μέθοδος επιβραδύνει την ολόκληρη εφαρμογή. Σε αυτή την περίπτωση, απλώς απενεργοποιήστε τη λειτουργία αναίρεσης/επανάληψης.
- **Display world reference:** Εμφανίζει ένα μικρό παγκόσμιο αναφορικό πλαίσιο στο κάτω αριστερό μέρος μιας προβολής κάμερας. Όπως παντού στο CoppeliaSim, τα κόκκινα, πράσινα και μπλε βέλη αντιστοιχούν αντίστοιχα στους άξονες x, y και z.
- **Auto-save enabled:** Όταν είναι ενεργοποιημένη η αυτόματη αποθήκευση, κάθε ανοιγμένη σκηνή θα αποθηκεύεται τακτικά. Σε περίπτωση κολλήματος, οι αυτόματα αποθηκευμένες σκηνές μπορούν να ανακτηθούν. Η καθυστέρηση αυτόματης αποθήκευσης μπορεί να προσαρμοστεί στο αρχείο system/usrset.txt.
- **Hide console window:** Επιτρέπει την απόκρυψη ή εμφάνιση του παραθύρου κονσόλας. Από προεπιλογή, το παράθυρο κονσόλας είναι κρυμμένο. Στο Mac, αυτό το στοιχείο δεν είναι διαθέσιμο (τα πρότυπα έξοδος μπορούν να προβληθούν στη συστηματική κονσόλα (Applications/Utilities/Console)).
- **Hide hierarchy during simul.:** Αυτόματα κρύβει την ιεραρχία σκηνής κατά τη διάρκεια της εκτέλεσης της προσομοίωσης.
- **OpenGL settings:** Ανοίγει έναν διάλογο που επιτρέπει τη ρύθμιση των περισσότερων ρυθμίσεων που σχετίζονται με το OpenGL.

1.3.13 OpenGL settings



- **Offscreen context type:** Ο τύπος των πλαισίων από ανακλήσεις εκτός οθόνης. Προεπιλογή είναι αόρατα παράθυρα.
- **FBO type:** Ο τύπος των αντικειμένων καρέ πλαισίων. Προεπιλογή είναι Qt-based FBOs στο Mac και μη-Qt-based FBOs σε Windows και Linux.
- **VBO operation:** Εάν χρησιμοποιούνται Vertex Buffer Objects. Η προεπιλογή χρησιμοποιεί VBOs.
- **Idle fps:** Ο αριθμός των καρέ ανά δευτερόλεπτο όταν το πρόγραμμα είναι σε αδράνεια.

1.3.14 Shortcuts/Συντομεύσεις

Όταν η εστίαση είναι στην ιεραρχία σκηνής ή σε μια σελίδα υποστηρίζονται τα ακόλουθα πλήκτρα συντόμευσης:

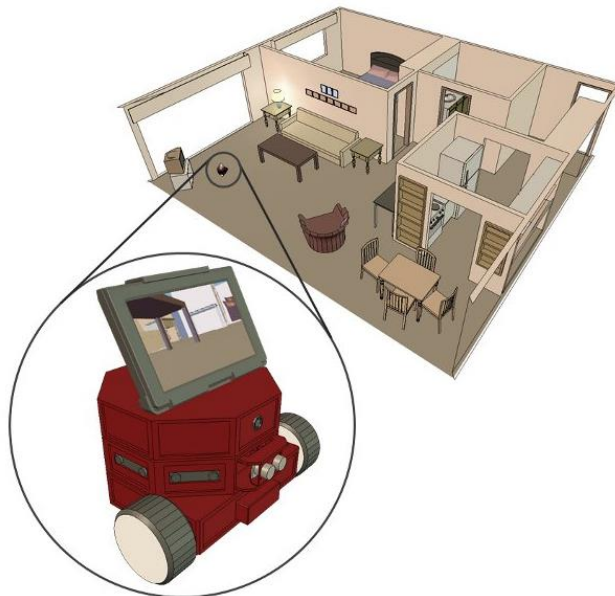
- **CTRL+A:** επιλογή όλων
- **<esc>:** απαλοιφή της επιλογής
- **CTRL+C:** αντιγραφή της επιλογής
- **CTRL+V:** επικόλληση του αντιγραφικού προθέματος
- **CTRL+X:** αποκοπή της επιλογής
- **<delete>:** διαγραφή της επιλογής
- **<backspace>:** διαγραφή της επιλογής
- **CTRL+O:** άνοιγμα σκηνής
- **CTRL+N:** άνοιγμα νέας σκηνής

Προσομοίωση Ρομπότ μέσω Υπολογιστή

- **CTRL+S**: αποθήκευση της σκηνής
- **CTRL+W**: κλείσιμο της σκηνής
- **CTRL+Q**: έξοδος από την εφαρμογή
- **CTRL+<space>**: έναρξη/διακοπή της προσομοίωσης
- **CTRL+E**: εναλλαγή μεταξύ 1) κανονικής λειτουργίας, 2) μετάφρασης αντικειμένου και 3) περιστροφής αντικειμένου με το ποντίκι
- **CTRL+D**: ανοίγει το παράθυρο ιδιοτήτων αντικειμένου
- **CTRL+G**: ανοίγει το παράθυρο του μοντέλου υπολογισμού
- **CTRL+B**: προσαρμογή της προβολής για να ταιριάζει στα επιλεγμένα αντικείμενα, ή στην ολόκληρη τη σκηνή αν δεν έχει επιλεγεί κανένα αντικείμενο. Η εστίαση πρέπει να είναι σε μια προβολή.
- **CTRL+ALT+C**: ορίζει την εστίαση στον διαχειριστή
- **CTRL+L**: εκκαθάριση της γραμμής κατάστασης (όταν η εστίαση βρίσκεται στον διαχειριστή)

1.3.15 Σκηνές και μοντέλα

Οι σκηνές και τα μοντέλα είναι τα κύρια στοιχεία προσομοίωσης του CoppeliaSim. Ένα μοντέλο είναι ένα υποστοιχείο μιας σκηνής, σαφώς σημειωμένο ως μοντέλο. Μια σκηνή μπορεί να περιέχει οποιονδήποτε αριθμό μοντέλων. Το ακόλουθο σχήμα απεικονίζει τη σχέση σκηνής-μοντέλου:



Τόσο οι σκηνές όσο και τα μοντέλα μπορούν να περιέχουν ένα ή περισσότερα από τα ακόλουθα στοιχεία:

Προσομοίωση Ρομπότ μέσω Υπολογιστή

- Αντικείμενα σκηνής (scene objects)
- Παιδικά σενάρια (Child scripts)

Οι σκηνές αποθηκεύονται σε αρχεία τύπου "*.tnt" και τα μοντέλα αποθηκεύονται σε αρχεία τύπου "*.ttm". Και οι δύο τύποι αρχείων μπορούν να ανοίξουν μεταφέροντας τα από το παράθυρο εξερεύνησης και ρίχνοντάς τα στην εφαρμογή CoppeliaSim. Μπορούν επίσης να ανοίξουν με διπλό κλικ.

1.3.16 Σκηνές

Σε σύγκριση με τα μοντέλα, μια σκηνή μπορεί να περιέχει ακριβώς τον ίδιο τύπο στοιχείων, αλλά επιπλέον περιλαμβάνει τα εξής στοιχεία, ειδικά για σκηνές:

- Το περιβάλλον
- Το κύριο σενάριο (main script)
- Σελίδες και προβολές

Το περιεχόμενο μιας σκηνής ή μιας εικόνας σκηνής μπορεί να προβληθεί μέσω ενός αντικειμένου προβολής που σχετίζεται με μια προβολή, η οποία περιέχεται σε μια σελίδα. Όταν δημιουργούμε μια νέα σκηνή ([File → New Scene]), η προεπιλεγμένη σκηνή θα περιέχει τα εξής στοιχεία:

- Πολλά αντικείμενα κάμερας: οι κάμερες επιτρέπουν την προβολή της σκηνής αν σχετίζονται με μια προβολή.
- Πολλά αντικείμενα φωτός: χωρίς φως, η σκηνή θα ήταν σχεδόν αόρατη. Το φως χρησιμοποιείται για να φωτίζει τη σκηνή.
- Πολλές προβολές: μια προβολή σχετίζεται με μια κάμερα και εμφανίζει αυτό που βλέπει η κάμερα. Οι προβολές περιέχονται σε σελίδες.
- Πολλές σελίδες: μια σελίδα περιέχει μία ή περισσότερες προβολές.
- Το περιβάλλον: το περιβάλλον αποτελείται από ιδιότητες όπως το περιβάλλον φως, την ομίχλη, το χρώμα του φόντου κ.λπ.
- Το πάτωμα: το πάτωμα αποτελείται από αντικείμενα ομαδοποιημένα σε ένα μοντέλο.
- Το προεπιλεγμένο κύριο script: το προεπιλεγμένο κύριο script θα πρέπει να επιτρέπει την εκτέλεση ελάχιστων προσομοιώσεων, χωρίς την ανάγκη για child scripts. Ένα παιδικό σενάριο που θα αντιγραφεί στη σκηνή αργότερα θα εκτελεστεί επίσης αυτόματα (καλούμενο από το κύριο σενάριο) αν σχετίζεται με ένα αντικείμενο σκηνής.

1.3.17 Μοντέλα

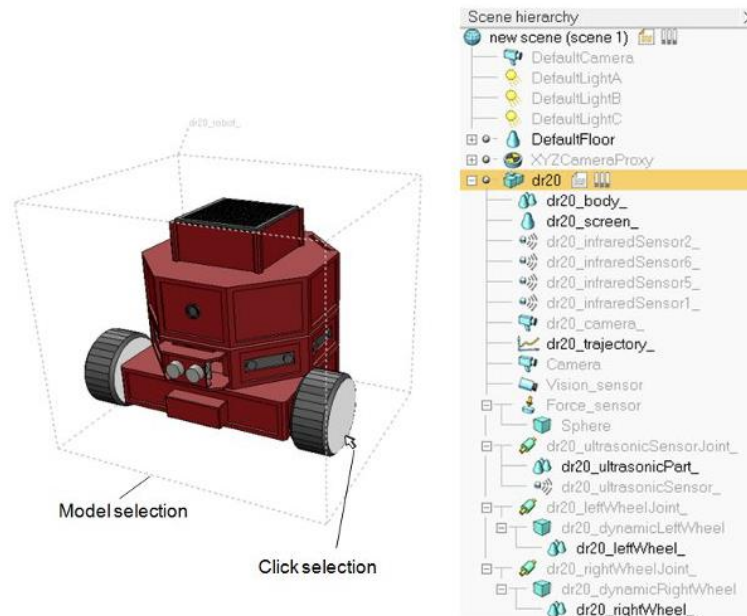
Ένα μοντέλο είναι ένα υπο-στοιχείο μιας σκηνής. Ένα μοντέλο από μόνο του δεν μπορεί να υπάρξει, εκτός από ένα αρχείο (τύπου αρχείου "*.ttm") ούτε μπορεί να προσομοιωθεί από μόνο του. Ένα μοντέλο πρέπει να περιέχεται σε μια σκηνή για να είναι λειτουργικό.

Τα μοντέλα ορίζονται από μια επιλογή αντικειμένων σκηνής που είναι δομημένα σε ένα ιεραρχικό δέντρο, όπου η βάση του δέντρου πρέπει να είναι ένα αντικείμενο με τη σημαία "το αντικείμενο είναι βάση μοντέλου". Μπορούν να φορτωθούν με την εντολή [File → Load model...]. Ωστόσο, είναι πολύ πιο εύκολο και βολικό να φορτώσουμε ένα μοντέλο με μια λειτουργία drag and drop μεταξύ του model browser και μιας προβολής σκηνής. Τα μοντέλα μπορούν να αποθηκευτούν με την εντολή [File → Save model as...] απλά πρέπει να βεβαιωθούμε ότι είναι επιλεγμένο μόνο ένα αντικείμενο με τη σημαία "το αντικείμενο είναι βάση μοντέλου" διαφορετικά το στοιχείο μενού "Save model as..." δεν θα είναι ενεργοποιημένο.

Ένα μοντέλο ορίζεται στα ακόλουθα βήματα:

- Προσκολλάμε όλα τα αντικείμενα που λογικά ανήκουν στο μοντέλο σε ένα βασικό αντικείμενο, έτσι ώστε το βασικό αντικείμενο να είναι η βάση του δέντρου μοντέλου.
- Ελέγχουμε το στοιχείο "το αντικείμενο είναι βάση μοντέλου" στις κοινές ιδιότητες αντικειμένου.
- Στον ίδιο διάλογο όπως παραπάνω, κάνοντας κλικ στο "Edit model properties", μπορούμε να ορίσουμε ειδικές υπερκείμενες ιδιότητες (π.χ. να κάνουμε ολόκληρο το μοντέλο αόρατο, μη συγκρουόμενο, κ.λπ.). Αυτό επιτρέπει να απενεργοποιήσουμε γρήγορα ορισμένες ιδιότητες για όλα τα αντικείμενα που ορίζονται στο μοντέλο.
- Για όλα τα αντικείμενα στο μοντέλο, εκτός από το βασικό αντικείμενο, ελέγχουμε το στοιχείο "επιλέξτε τη βάση του μοντέλου αντ' αυτού" στις κοινές ιδιότητες αντικειμένου. Αυτό θα προστατεύσει το μοντέλο μας: δεν θα μπορούμε να επιλέγουμε άμεσα μεμονωμένα αντικείμενα στο μοντέλο μας και θα μπορούμε να χειριστούμε το μοντέλο μας σχεδόν σαν ένα μόνο αντικείμενο.
- Για όλα τα αντικείμενα που κανονικά δεν είναι ορατά, ελέγχουμε το στοιχείο "Μην εμφανίζετε ως μέρος της επιλογής μοντέλου". Αυτό θα κάνει το πλαίσιο περιγράμματος του μοντέλου να εμφανίζεται στο σωστό μέγεθος γύρω από το μοντέλο.
- Σκεφτόμαστε το ρόλο του μοντέλου: θα μπορούμε να το συνδέσουμε με κάποιο άλλο αντικείμενο; (π.χ. το μοντέλο μας είναι ένας γρίφος που θα μπορούσαμε να συνδέσουμε με έναν καρπό χειριστή). Ή θα μπορούμε να συνδέσουμε κάποιο άλλο μοντέλο σε αυτό; (π.χ. το μοντέλο μας είναι ένας χειριστής που θα μπορούσε να δεχτεί ένα μοντέλο γρίφου). Μόλις αποφασιστεί ο ρόλος, ορίζουμε τη συμπεριφορά συναρμολόγησης του μοντέλου.

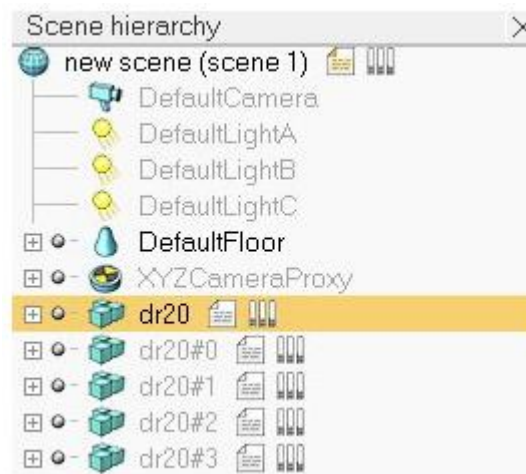
Τώρα, μεμονωμένα αντικείμενα που κατασκευάζονται στη βάση του μοντέλου δεν μπορούν να επιλεγούν πλέον στη σκηνή (η επιλογή τους θα επιλέξει τη βάση του μοντέλου αντ' αυτού), ωστόσο μπορούν ακόμα να επιλεγούν μεμονωμένα κρατώντας πατημένα τα πλήκτρα ctrl και shift κατά τη διάρκεια της επιλογής ή επιλέγοντάς τα στην ιεραρχία της σκηνής. Επιπλέον, όταν το βασικό αντικείμενο είναι επιλεγμένο, ένα διακεκομμένο πλαίσιο περιγράμματος που περικλείει ολόκληρο το μοντέλο εμφανίζεται όπως φαίνεται στην ακόλουθη εικόνα:



Παρατηρούμε την ετικέτα μοντέλου στην αριστερή πλευρά του εικονιδίου του αντικειμένου που έχει επισημανθεί ως βάση μοντέλου:



Ένα διπλό κλικ σε μια ετικέτα μοντέλου ανοίγει τον διάλογο του μοντέλου, όπου μπορούμε να ρυθμίσουμε τις ιδιότητες του μοντέλου. Είναι επίσης καλή πρακτική να καταρρέει η ιεραρχία του μοντέλου αφού έχει επεξεργαστεί το μοντέλο, ώστε να είναι εύκολος ο εντοπισμός του αριθμού των λογικά ομαδοποιημένων στοιχείων/μοντέλων.



Η ομαδοποίηση αρκετών αντικειμένων ως μοντέλου είναι επίσης σημαντική όταν ένα child script έχει πρόσβαση στα αντικείμενα προγραμματιστικά. Στο CoppeliaSim, τα αντικείμενα/μοντέλα μπορούν να αναπαραχθούν ανά πάσα στιγμή, ακόμη και κατά τη διάρκεια μιας προσομοίωσης. Για να μπορεί το αναπαραγμένο child script να έχει πρόσβαση στα σωστά αντικείμενα (όχι στα αρχικά αντικείμενα αλλά στα αναπαραγμένα), το child script πρέπει πάντα να αναπαράγεται ταυτόχρονα με τα αντικείμενα στα οποία έχει πρόσβαση. Ένας τρόπος για να διασφαλιστεί αυτό είναι να δημιουργηθεί ένα μοντέλο και να βεβαιωθούμε ότι τα child scripts που έχουν πρόσβαση σε αντικείμενα στο μοντέλο είναι συνδεδεμένα με αντικείμενα που περιέχονται στο μοντέλο. Το

Προσομοίωση Ρομπότ μέσω Υπολογιστή

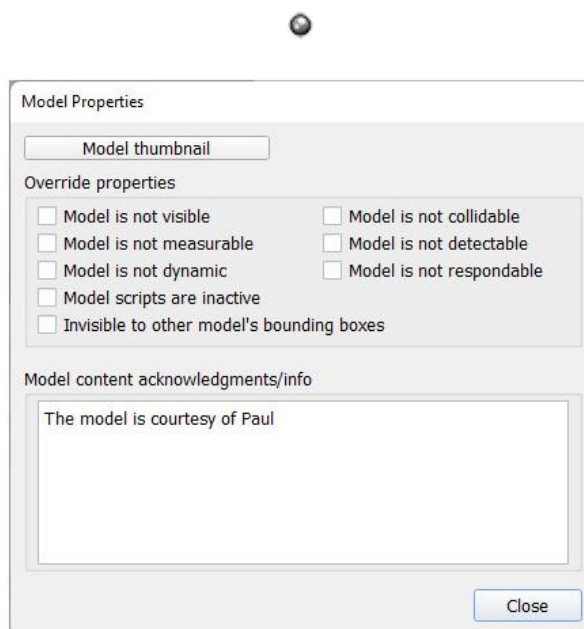
καλύτερο είναι να συνδέσουμε ένα child script (μπορεί να υπάρχουν και δευτερεύοντα child scripts) με τη βάση του μοντέλου.

Η αντιγραφή και επικόλληση ενός μοντέλου λειτουργεί ακριβώς όπως η αποθήκευση του μοντέλου και στη συνέχεια η φόρτωσή του (χρησιμοποιώντας ωστόσο μια μνήμη αντί του χώρου στο δίσκο). Τα μοντέλα μπορούν να αντιγραφούν από μια σκηνή σε μια άλλη όπως κάθε άλλο αντικείμενο. Τα αρχεία μοντέλων (αρχεία "*.ttm") υποστηρίζουν επίσης λειτουργίες μεταφοράς και απόθεσης μεταξύ του παραθύρου εξερευνητή και του παραθύρου εφαρμογής. Τα αρχεία μοντέλων μπορούν επίσης να ανοιχτούν με διπλό κλικ, οπότε θα εκκινήσουν την εφαρμογή CorreliaSim και θα φορτωθούν σε μια προεπιλεγμένη σκηνή.

Οι ιδιότητες ενός μοντέλου μπορούν να ρυθμιστούν μεμονωμένα στον διάλογο του μοντέλου.

1.3.18 Διάλογος μοντέλου/Model dialog

Οι ιδιότητες ενός μοντέλου μπορούν να ρυθμιστούν μεμονωμένα στον διάλογο του μοντέλου. Μπορεί να ανοίξει με διπλό κλικ σε ένα εικονίδιο μοντέλου στην ιεραρχία της σκηνής:



- **Μικρογραφία μοντέλου/Model thumbnail:** Όταν αποθηκεύουμε ένα μοντέλο, εμφανίζεται ένα παράθυρο διαλόγου που ζητά μικρογραφία μοντέλου (εμφανίζεται στον περιηγητή μοντέλων). Αν θέλουμε η μικρογραφία να δείχνει διαφορετική διαμόρφωση του μοντέλου μας, μπορούμε να την ορίσουμε εδώ.
- **Παράκαμψη ιδιοτήτων/Override properties:** Εδώ μπορούμε να απενεργοποιήσουμε συγκεκριμένες ιδιότητες για όλο το μοντέλο (δηλαδή για όλα τα αντικείμενα στην ιεραρχία του μοντέλου). Αυτό είναι χρήσιμο για γρήγορη απενεργοποίηση μοντέλου που απαιτεί πολύ χρόνο υπολογισμού.
- **Αναγνωρίσεις περιεχομένου μοντέλου/Πληροφορίες/Model content acknowledgments/info:** Πληροφορίες που σχετίζονται με ένα μοντέλο. Είναι καλή πρακτική να αναγνωρίζουμε τον αρχικό δημιουργό ενός μοντέλου ή εισαγόμενου πλέγματος. Όταν ανοίγουμε ένα μοντέλο με πληροφορίες αναγνώρισης, αυτές εμφανίζονται αυτόματα.

1.3.19 Περιβάλλον

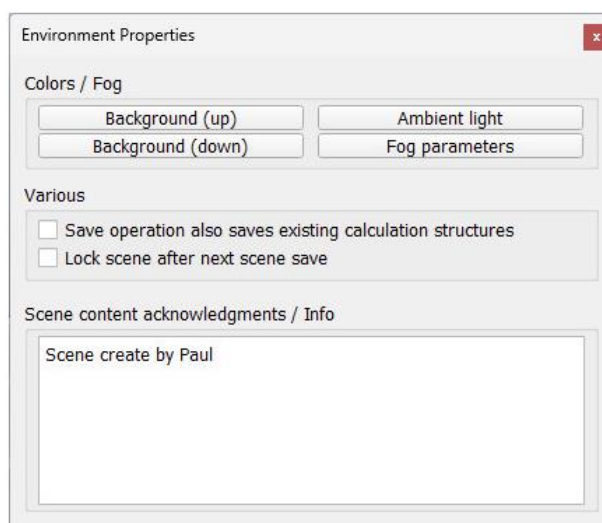
Οι ιδιότητες και οι παράμετροι του περιβάλλοντος στο CoppeliaSim ορίζουν χαρακτηριστικά που είναι μέρος μιας σκηνής, αλλά δεν είναι αντικείμενα σκηνής. Αυτές οι ιδιότητες και παράμετροι δεν αποθηκεύονται όταν αποθηκεύεται ένα μοντέλο, αλλά μόνο όταν αποθηκεύεται μια σκηνή.

Ένα περιβάλλον ορίζει τις εξής ιδιότητες και παραμέτρους:

- Χρώματα φόντου
- Παράμετροι ομίχλης (αλληλεπιδρούν μόνο με κάμερες ή αισθητήρες όρασης εάν επιλεγεί η αντίστοιχη επιλογή)
- Φωτισμός περιβάλλοντος
- Πληροφορίες δημιουργίας σκηνής
- Επιπλέον ρυθμίσεις

1.3.20 Διάλογος περιβάλλοντος/Environment dialog

Το παράθυρο διαλόγου περιβάλλοντος μπορεί να ανοίξει με την επιλογή [Tools → Environment] ή με διπλό κλικ στο εξής εικονίδιο στην ιεραρχία της σκηνής:

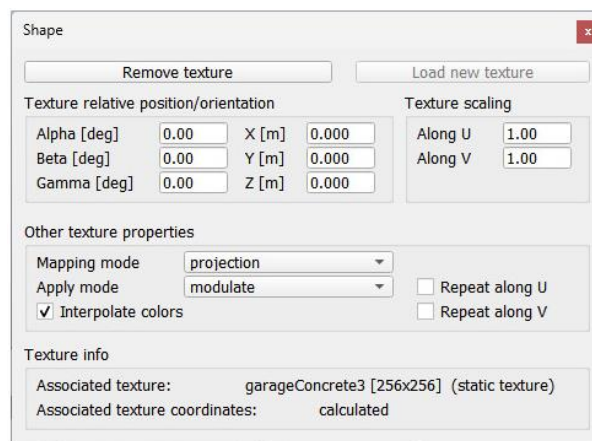


- **Background (up/down):** επιτρέπει τη ρύθμιση του χρώματος φόντου μιας σκηνής. Η επάνω συνιστώσα αντιστοιχεί στο πάνω μέρος της οθόνης (ουρανό), η κάτω συνιστώσα αντιστοιχεί στο κάτω μέρος της οθόνης. Τα χρώματα φόντου είναι ορατά μόνο όταν η λειτουργία ομίχλης είναι απενεργοποιημένη.
- **Ambient light:** επιτρέπει την προσαρμογή του φωτισμού περιβάλλοντος μιας σκηνής. Το φως του περιβάλλοντος μπορεί να θεωρηθεί ως το ελάχιστο φως μιας σκηνής, φωτίζει ένα αντικείμενο με τον ίδιο ακριβώς τρόπο από όλες τις κατευθύνσεις.
- **Fog parameters:** επιτρέπει τη ρύθμιση διαφόρων παραμέτρων ομίχλης.

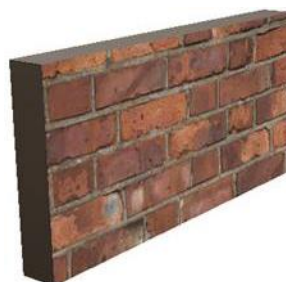
- **Save operation also saves existing calculation structures:** για υπολογισμούς αποστάσεων, ανιχνεύσεις συγκρούσεων κ.λπ. μια δομή δεδομένων υπολογίζεται στην αρχή μιας προσομοίωσης (προεπεξεργασία) ή την πρώτη φορά που ένα σχήμα εμπλέκεται σε τέτοιους υπολογισμούς, για να επιταχύνει τους υπολογισμούς. Ο υπολογισμός αυτής της δομής δεδομένων μπορεί να είναι χρονοβόρος, οπότε μπορούμε να επιλέξουμε να την αποθηκεύσουμε μαζί με τη σκηνή ή το μοντέλο. Ωστόσο, πρέπει να είμαστε ενήμεροι ότι η επιπλέον πληροφορία που θα αποθηκευτεί είναι μεγάλη και θα οδηγήσει σε μεγαλύτερα αρχεία (μερικές φορές διπλάσια ή μεγαλύτερα).
- **Lock scene after next scene save:** επιλέγουμε αυτό το στοιχείο αν θέλουμε να κλειδώσουμε τη σκηνή μας από επεξεργασία / τροποποίηση, προβολή περιεχομένου σεναρίου και εξαγωγή πόρων. Μετά την επόμενη αποθήκευση σκηνής, η σκηνή θα είναι κλειδωμένη. Πρέπει να βεβαιωθούμε ότι έχουμε αποθηκεύσει την ίδια σκηνή σε κατάσταση μη κλειδωμένη αν θέλουμε να είμαστε σε θέση να την τροποποιήσουμε σε μεταγενέστερο χρόνο.
- **Scene content acknowledgements / Info:** πληροφορίες που σχετίζονται με μια σκηνή. Είναι πάντα καλή πρακτική να αναγνωρίζουμε τον αρχικό δημιουργό μιας σκηνής, μοντέλου ή εισαγόμενου πλέγματος. Όταν ανοίγουμε μια σκηνή που περιέχει πληροφορίες αναγνώρισης, θα εμφανίζονται αυτόματα αυτές οι πληροφορίες.

1.4 Διάλογος υφής/Texture dialog

Ο διάλογος υφής μας επιτρέπει να δούμε και να τροποποιήσουμε τις ιδιότητες που σχετίζονται με μια υφή που είναι συνδεδεμένη με ένα σχήμα. Είναι προσβάσιμος κάνοντας κλικ στο στοιχείο Set Texture-item στο διάλογο ιδιοτήτων σχήματος(shape properties):



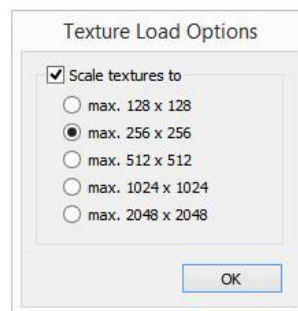
Μια υφή είναι μια εικόνα bitmap που μπορεί να εφαρμοστεί σε επιφάνειες για να τις κάνει να φαίνονται πιο ρεαλιστικές. Η ακόλουθη εικόνα απεικονίζει μια επιφάνεια με υφή:



Υποστηρίζονται οι παρακάτω 5 μέθοδοι χαρτογράφησης υφής:

- **Χαρτογράφηση προβολής:** η υφή απλά προβάλλεται στο επίπεδο X-Y του αντικειμένου (δείτε εδώ παραπάνω). Οι συντεταγμένες υφής θα υπολογιστούν.
- **Κυλινδρική χαρτογράφηση:** η υφή περιτυλίγεται γύρω από τον άξονα Z του αντικειμένου. Οι συντεταγμένες υφής θα υπολογιστούν.
- **Σφαιρική χαρτογράφηση:** η υφή χαρτογραφείται σφαιρικά πάνω στο αντικείμενο. Οι συντεταγμένες υφής θα υπολογιστούν.
- **Χαρτογράφηση κουτιού:** η υφή εφαρμόζεται σε όλες τις 6 επιφάνειες ενός αντικειμένου που μοιάζει με κουτί. Οι συντεταγμένες υφής θα υπολογιστούν
- **Εισαγόμενες συντεταγμένες υφής:** συγκεκριμένες συντεταγμένες υφής μπορούν να εισαχθούν ταυτόχρονα με την εισαγωγή του πλέγματος, μέσω της μορφής αρχείου OBJ ή DAE.

Κατά τη φόρτωση μιας νέας υφής, θα ανοίξει ο ακόλουθος διάλογος:



Συνιστάται να διατηρήσουμε τις προεπιλεγμένες ρυθμίσεις, προκειμένου να διατηρηθεί η συμβατότητα με παλαιότερες κάρτες γραφικών σε υψηλό επίπεδο (όταν είναι επιλεγμένο το "Scale texture to", όλες οι υφές θα αλλάζουν μέγεθος σε δύναμη του 2) και το μέγεθος των αρχείων σε χαμηλό επίπεδο.

Μόλις μια υφή φορτωθεί στη μνήμη, μπορεί να κλιμακωθεί, να μετατοπιστεί και να περιστραφεί πριν πραγματοποιηθεί η πραγματική χαρτογράφηση, προκειμένου να επιτευχθεί η επιθυμητή οπτική εμφάνιση.

Οι στατικές υφές μπορούν να φορτωθούν κάνοντας κλικ στο κουμπί "Load new texture". Οι ακόλουθες μορφές αρχείων υποστηρίζονται αυτή τη στιγμή:

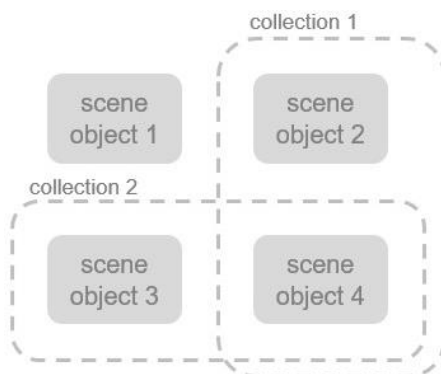
- JPEG
- PNG
- TGA
- BMP
- TIFF
- GIF

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Αντί να φορτώσουμε υφές από αρχείο, μπορούμε επίσης να επιλέξουμε ήδη φορτωμένες στατικές υφές ή ακόμη και δυναμικές υφές που δημιουργούνται από αισθητήρες όρασης. Απλώς κάνουμε κλικ στο κουμπί "Select texture from existing textures". Πρέπει να σημειώσουμε ότι οι δυναμικές υφές θα εμφανιστούν σωστά μόνο κατά τη διάρκεια της προσομοίωσης (και επίσης μόνο εάν ο σχετικός αισθητήρας όρασης λειτουργεί σωστά).

1.4.1 Οντότητες/Entities

Μια οντότητα είναι ένας όρος που αναφέρεται σε ένα αντικείμενο σκηνής ή σε μια συλλογή. Η ακόλουθη εικόνα απεικονίζει τη σχέση μεταξύ αντικειμένων σκηνής, συλλογών και οντοτήτων:



1.4.1.1 Συλλογές/Collections

Μία συλλογή είναι μία συλλογή αντικειμένων σκηνής που ορίζεται από τον χρήστη. Τόσο οι συλλογές όσο και τα αντικείμενα σκηνής είναι οντότητες. Οι συλλογές είναι χρήσιμες όταν αναφερόμαστε σε πολλά αντικείμενα όπως για παράδειγμα ένα ρομπότ, κατά την εκτέλεση μίας συγκεκριμένης λειτουργίας, όπως ο έλεγχος σύγκρουσης μεταξύ δύο ρομπότ: εάν ένα ζεύγος ανίχνευσης σύγκρουσης οριστεί ως (collectionRobot1; collectionRobot2), τότε ο αλγόριθμος ελέγχου σύγκρουσης θα ελέγχει εάν κάποιο αντικείμενο από την πρώτη συλλογή παρεμβαίνει σε κάποιο αντικείμενο από τη δεύτερη συλλογή.

Οι συλλογές είναι οντότητες συγκρουόμενες, μετρήσιμες και ανιχνεύσιμες. Αυτό σημαίνει ότι οι συλλογές:

- μπορούν να χρησιμοποιηθούν σε ανιχνεύσεις συγκρούσεων έναντι άλλων συγκρουόμενων οντοτήτων.
- μπορούν να χρησιμοποιηθούν σε υπολογισμούς ελάχιστης απόστασης με άλλες μετρήσιμες οντότητες.
- μπορούν να ανιχνευθούν από αισθητήρες εγγύτητας.

Ακόμα κι αν μια συλλογή είναι συγκρουόμενη, μετρήσιμη και ανιχνεύσιμη, αυτό δεν σημαίνει ότι όλα τα αντικείμενα που περιέχονται στη συλλογή είναι συγκρουόμενα, μετρήσιμα και ανιχνεύσιμα:

- Κατά την ανίχνευση σύγκρουσης, μόνο τα συγκρουόμενα αντικείμενα μιας συλλογής (υποσύνολο της συλλογής) ελέγχονται έναντι μιας άλλης συγκρουόμενης οντότητας.
- Κατά τη μέτρηση απόστασης, μόνο τα μετρήσιμα αντικείμενα μιας συλλογής (υποσύνολο της συλλογής) μετρούνται έναντι μιας άλλης μετρήσιμης οντότητας.
- Μόνο τα ανιχνεύσιμα αντικείμενα μιας συλλογής (υποσύνολο της συλλογής) μπορούν να ανιχνευθούν από αισθητήρες εγγύτητας.

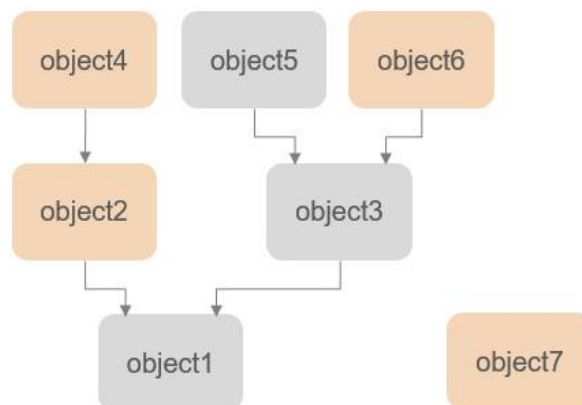
Προσομοίωση Ρομπότ μέσω Υπολογιστή

Μια συλλογή μπορεί ωστόσο να υπερισχύσει των συγκρουόμενων, μετρήσιμων και ανιχνεύσιμων ιδιοτήτων των αντικειμένων της.

Μια συλλογή ορίζεται από ένα ή περισσότερα στοιχεία που μπορούν να συνδυαστούν με προσθετικό ή αφαιρετικό τρόπο. Υποστηρίζονται τα ακόλουθα στοιχεία:

- ένα μόνο αντικείμενο
- ένα δέντρο αντικειμένων: αυτό το στοιχείο αντιπροσωπεύει όλους τους απογόνους ενός μόνο αντικειμένου (που βρίσκεται στη ρίζα του δέντρου). Αν το δέντρο μεγαλώνει ή μικραίνει, το στοιχείο θα μεγαλώνει/μικραίνει δυναμικά επίσης.
- μια αλυσίδα αντικειμένων: αυτό αντιπροσωπεύει όλα τα αντικείμενα που συναντώνται ακολουθώντας όλους τους προγόνους ενός μόνο αντικειμένου. Αν η αλυσίδα μεγαλώνει ή μικραίνει, το στοιχείο θα μεγαλώνει/μικραίνει δυναμικά επίσης.
- όλα τα αντικείμενα της σκηνής

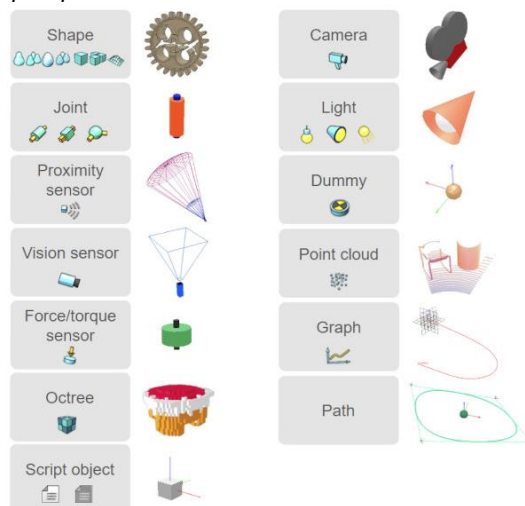
Μια συλλογή δεν είναι στατική και αξιολογείται/ενημερώνεται συνεχώς για να καθορίσει τα περιεχόμενα αντικείμενά της. Μια δεδομένη συλλογή μπορεί επίσης να οριστεί με πολλούς διαφορετικούς τρόπους, ανάλογα με τις συγκεκριμένες ανάγκες:



1.4.2 Αντικείμενα σκηνής/Scene objects

Τα κύρια στοιχεία στο CoppeliaSim που χρησιμοποιούνται για την κατασκευή μιας σκηνής προσομοίωσης είναι τα αντικείμενα σκηνής. Αυτά είναι ορατά στην ιεραρχία σκηνής και στην προβολή σκηνής. Στην προβολή σκηνής, τα αντικείμενα σκηνής έχουν μια τρισδιάστατη αναπαράσταση όπως απεικονίζεται στην ακόλουθη εικόνα:

Προσομοίωση Ρομπότ μέσω Υπολογιστή



Η ακόλουθη λίστα παρέχει μια σύντομη λειτουργική περιγραφή κάθε τύπου αντικείμενου σκηνής:

- Σχήματα/Shapes: ένα σχήμα είναι ένα άκαμπτο πλέγμα που αποτελείται από τριγωνικές όψεις.
- Αρθρώσεις/Joints: μια άρθρωση είναι ένα αντικείμενο που επιτρέπει 1 ή 3 βαθμούς ελευθερίας μεταξύ των συνδεδεμένων αντικειμένων. Μία άρθρωση μπορεί να είναι παθητική ή ενεργή (ενεργοποιητής).
- Γραφήματα/Graphs: ένα γράφημα χρησιμοποιείται για την καταγραφή και την απεικόνιση δεδομένων προσομοίωσης.
- Ανδρείκελα/Dummies: ένα dummy είναι ένα σημείο με προσανατολισμό. Τα dummies είναι πολύχρηστα αντικείμενα που μπορούν να έχουν πολλές διαφορετικές χρήσεις.
- Αισθητήρες εγγύτητας/Proximity sensors: ένας αισθητήρας εγγύτητας μπορεί να ανιχνεύσει συγκεκριμένα αντικείμενα σκηνής με γεωμετρικά ακριβή τρόπο μέσα στον όγκο ανίχνευσής του.
- Αισθητήρες όρασης /Vision sensors: ένας αισθητήρας όρασης είναι ένας αισθητήρας τύπου κάμερας, που αντιδρά στο φως, τα χρώματα και τις εικόνες.
- Αισθητήρες δύναμης/Force sensors: ένας αισθητήρας δύναμης είναι ένα αντικείμενο που μπορεί να μετρήσει δυνάμεις και ροπές που εφαρμόζονται σε αυτό.
- Κάμερες/Cameras: μια κάμερα είναι ένα αντικείμενο που επιτρέπει την παρακολούθηση της σκηνής προσομοίωσης από διάφορες οπτικές γωνίες.
- Φώτα/Lights: ένα φως είναι ένα αντικείμενο που επιτρέπει τον φωτισμό της σκηνής προσομοίωσης.
- Δέντρα OC/OC trees: ένα δέντρο OC είναι μια δομή διαμερισμού χώρου αποτελούμενη από voxels.
- Νέφη σημείων/Point Clouds: ένα νέφος σημείων είναι μια δομή δέντρου OC που περιέχει σημεία.
- Διαδρομές/Paths: μια διαδρομή είναι μια ακολουθία σημείων με προσανατολισμό στο χώρο. Η διαδρομή είναι ένα ψευδο-αντικείμενο σκηνής, καθώς αποτελείται από πολλά αντικείμενα σκηνής.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

- Αντικείμενα σεναρίων/Script objects: ένα αντικείμενο σεναρίου είναι ένα σενάριο ενσωματωμένο σε ένα αντικείμενο σκηνής ως σενάριο προσομοίωσης ή προσαρμογής.

Μερικά από τα παραπάνω αντικείμενα σκηνής μπορούν να έχουν ειδικές ιδιότητες που επιτρέπουν σε άλλα αντικείμενα ή σε διάφορες λειτουργίες να αλληλεπιδρούν με αυτά. Μπορούν να είναι:

- Συγκρουόμενα/Collidable: συγκρουόμενα αντικείμενα μπορούν να ελεγχθούν για σύγκρουση με άλλα συγκρουόμενα αντικείμενα.
- Μετρήσιμα/Measurable: μετρήσιμα αντικείμενα μπορούν να έχουν την ελάχιστη απόσταση μεταξύ αυτών και άλλων μετρήσιμων αντικειμένων υπολογισμένη.
- Ανιχνεύσιμα/Detectable: ανιχνεύσιμα αντικείμενα μπορούν να ανιχνευθούν από αισθητήρες εγγύτητας.
- Ορατά/Viewable: ορατά αντικείμενα μπορούν να παρακολουθούνται, να παρατηρούνται ή το περιεχόμενό τους να απεικονίζεται σε προβολές.

Κάθε αντικείμενο σκηνής έχει μια θέση μέσα στη σκηνή προσομοίωσης. Τα αντικείμενα μπορούν να συνδεθούν σε άλλα αντικείμενα (ή να κατασκευαστούν το ένα πάνω στο άλλο). Εάν το αντικείμενο A είναι κατασκευασμένο πάνω στο αντικείμενο B, τότε το αντικείμενο B είναι ο γονέας και το αντικείμενο A είναι το παιδί. Για να δημιουργήσουμε μια σχέση γονέα-παιδιού μεταξύ του αντικειμένου B και του αντικειμένου A, επιλέγουμε το αντικείμενο A, στη συνέχεια επιλέγουμε το αντικείμενο B (η σειρά επιλογής είναι σημαντική). Στη συνέχεια, επιλέγουμε [Edit → Set parent, keep pose(s)].

Εναλλακτικά, μπορούμε να σύρουμε και να αποθέσουμε ένα αντικείμενο πάνω σε ένα άλλο στην ιεραρχία της σκηνής για να επιτύχουμε ένα παρόμοιο αποτέλεσμα. Παρατηρούμε ότι η απόλυτη θέση του αντικειμένου A δεν άλλαξε. Ωστόσο, κοιτάζοντας την ιεραρχία της σκηνής, μπορούμε να δούμε ότι το αντικείμενο A έγινε παιδί του αντικειμένου B. Εάν τώρα μετακινήσουμε το αντικείμενο B, το αντικείμενο A θα ακολουθήσει αυτόματα, καθώς το αντικείμενο A είναι συνδεδεμένο με το αντικείμενο B. Το αντικείμενο A μπορεί να αποσυνδεθεί επιλέγοντάς το και στη συνέχεια επιλέγοντας [Edit → Set parent-less]. Κάνοντας αυτό, το αντικείμενο A θα αποσυνδεθεί χωρίς να αλλάξει η θέση του. Εναλλακτικά, μπορούμε να σύρουμε και να αποθέσουμε ένα αντικείμενο πάνω στο εικονίδιο του κόσμου για να επιτύχουμε ένα παρόμοιο αποτέλεσμα.

Κάθε αντικείμενο σκηνής έχει μια απόλυτη θέση (ή αθροιστική θέση) που είναι σχετική με το πλαίσιο αναφοράς του κόσμου και μια τοπική θέση (ή σχετική θέση) που είναι σχετική με το πλαίσιο αναφοράς του γονικού αντικειμένου. Στο παραπάνω παράδειγμα, όταν το αντικείμενο A έγινε παιδί του αντικειμένου B, η απόλυτη θέση του αντικειμένου A δεν άλλαξε, αλλά η τοπική του θέση τροποποιήθηκε.

Η απόλυτη θέση του τελευταίου επιλεγμένου αντικειμένου σκηνής εμφανίζεται στο πληροφοριακό κείμενο.

1.4.3 Ιδιότητες αντικειμένων σκηνής/Scene object properties

Ο διάλογος ιδιοτήτων αντικειμένου σκηνής βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το διάλογο με διπλό κλικ στο εικονίδιο ενός αντικειμένου στην ιεραρχία σκηνής ή με ένα κλικ στο κουμπί της γραμμής εργαλείων:

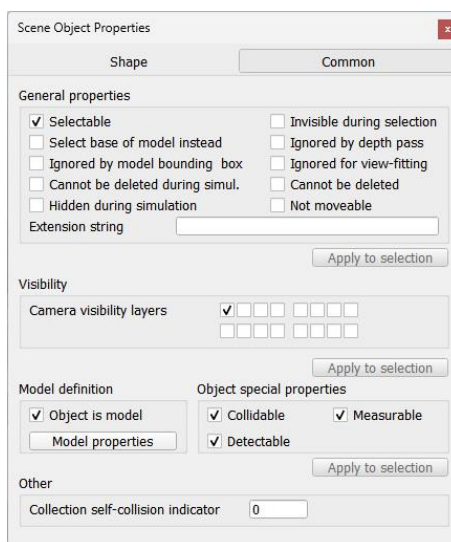


Προσομοίωση Ρομπότ μέσω Υπολογιστή

Ο διάλογος ιδιοτήτων αντικειμένου σκηνής εμφανίζει ιδιότητες που σχετίζονται με αντικείμενα (δηλαδή αντικείμενα σκηνής). Ο διάλογος είναι ευαίσθητος στο πλαίσιο και το περιεχόμενό του θα εξαρτηθεί κυρίως από την κατάσταση επιλογής του αντικειμένου σκηνής: μόνο οι ιδιότητες του τελευταίου επιλεγμένου αντικειμένου θα εμφανίζονται. Αυτές οι ιδιότητες χωρίζονται σε 2 μέρη:

- Ιδιότητες συγκεκριμένες για τον τύπο του αντικειμένου: ιδιότητες που είναι συγκεκριμένες για τον επιλεγμένο τύπο αντικειμένου.
- Κοινές ιδιότητες αντικειμένων: ιδιότητες κοινές σε όλους τους τύπους αντικειμένων.

Δύο κουμπιά στο πάνω μέρος του διαλόγου επιτρέπουν την επιλογή του επιθυμητού τύπου ιδιοτήτων για εμφάνιση. Εάν η επιλογή αντικειμένου είναι κενή, τότε όλα τα στοιχεία του διαλόγου θα είναι ανενεργά.

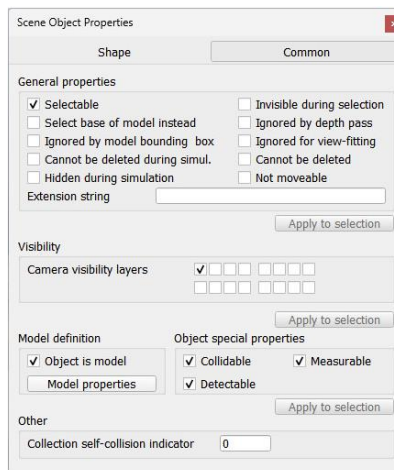


1.4.4 Κοινές ιδιότητες αντικειμένου/Object common properties

Το παράθυρο διαλόγου κοινών ιδιοτήτων αντικειμένου είναι μέρος του παραθύρου διαλόγου ιδιοτήτων αντικειμένου σκηνής, το οποίο βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το παράθυρο διαλόγου με διπλό κλικ σε ένα εικονίδιο αντικειμένου στην ιεραρχία της σκηνής ή κάνοντας κλικ στο κουμπί της γραμμής εργαλείων:



Στο παράθυρο διαλόγου ιδιοτήτων αντικειμένου σκηνής, κάνουμε κλικ στο κουμπί "Common" για να εμφανιστεί το παράθυρο διαλόγου κοινών ιδιοτήτων αντικειμένου. Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους του τελευταίου επιλεγμένου αντικειμένου. Εάν δεν έχει επιλεγεί κανένα αντικείμενο, το παράθυρο διαλόγου είναι ανενεργό. Εάν έχουν επιλεγεί περισσότερα από ένα αντικείμενα, τότε ορισμένες παράμετροι μπορούν να αντιγραφούν από το τελευταίο επιλεγμένο αντικείμενο στα άλλα επιλεγμένα αντικείμενα ("Apply to selection"):

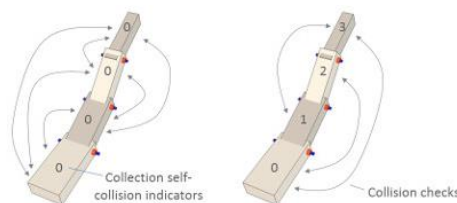


- **Selectable:** υποδεικνύει εάν το αντικείμενο μπορεί να επιλεγεί στη σκηνή. Τα αντικείμενα μπορούν πάντα να επιλέγονται στην ιεραρχία της σκηνής.
- **Invisible during selection:** όταν είναι ενεργοποιημένο, το αντικείμενο θα είναι αόρατο κατά τη διαδικασία επιλογής (δηλαδή θα μπορούμε να επιλέξουμε μέσω του αντικειμένου).
- **Ignored by depth pass:** όταν είναι ενεργοποιημένο, το αντικείμενο θα αγνοηθεί κατά τη διαδικασία απόδοσης βάθους. Η διαδικασία απόδοσης βάθους χρησιμοποιείται για την σωστή τοποθέτηση της κόκκινης σφαίρας για τις κινήσεις της κάμερας.
- **Select base of model instead:** εάν είναι ενεργοποιημένο, τότε η επιλογή του αντικειμένου στη σκηνή θα επιλέξει το πρώτο αντικείμενο με γονέα που έχει σημειωθεί ως βάση του μοντέλου αντί για αυτό. Αυτή η ιδιότητα είναι βολική όταν προστατεύουμε ένα μοντέλο από λανθασμένους χειρισμούς, επιτρέποντάς του να χειρίζεται ως μια ενιαία οντότητα μαζί με άλλα αντικείμενα.
- **Ignored by model bounding box:** όταν είναι επιλεγμένο και το αντικείμενο είναι μέρος ενός μοντέλου, τότε το πλαίσιο οριοθέτησης του μοντέλου (δηλαδή το πλαίσιο επιλογής του μοντέλου) δεν θα περιλαμβάνει αυτό το αντικείμενο. Αυτό είναι χρήσιμο για αόρατα αντικείμενα που μπορεί να κάνουν το πλαίσιο οριοθέτησης του μοντέλου να φαίνεται πολύ μεγάλο. Αυτή η ιδιότητα δεν έχει λειτουργικό αποτέλεσμα.
- **Ignored for view-fitting:** τα αντικείμενα με αυτό το στοιχείο επιλεγμένο δεν θα λαμβάνονται υπόψη κατά την προσαρμογή μιας σκηνής σε μια προβολή όταν δεν έχει επιλεγεί κανένα αντικείμενο. Συνήθως, τα δάπεδα και παρόμοια θα επισημαίνονται ως τέτοια.
- **Cannot be deleted during simul.:** όταν είναι ενεργοποιημένο, το αντικείμενο θα αγνοεί μια εντολή διαγραφής όταν τρέχει μια προσομοίωση (η διαγραφή θα λειτουργεί ωστόσο όταν ενεργοποιείται μέσω κώδικα).
- **Cannot be deleted:** όταν είναι ενεργοποιημένο, το αντικείμενο θα αγνοεί μια εντολή διαγραφής (η διαγραφή θα λειτουργεί ωστόσο όταν ενεργοποιείται μέσω κώδικα).
- **Hidden during simulation:** όταν είναι ενεργοποιημένο, το αντικείμενο κρύβεται κατά τη διάρκεια της προσομοίωσης.
- **Not moveable:** όταν είναι ενεργοποιημένο, το αντικείμενο δεν μπορεί να μετακινηθεί με το ποντίκι.
- **Extension string:** μια συμβολοσειρά που περιγράφει επιπλέον ιδιότητες αντικειμένου, που χρησιμοποιείται κυρίως από πρόσθετα.

- **Camera visibility layers:** κάθε αντικείμενο στο CoppeliaSim μπορεί να ανατεθεί σε ένα ή περισσότερα επίπεδα ορατότητας. Εάν υπάρχει τουλάχιστον ένα επίπεδο ορατότητας που ταιριάζει με τα επίπεδα του παραθύρου διαλόγου επιλογής επιπέδων, τότε το αντικείμενο θα είναι ορατό όταν το βλέπει η κάμερα. Από προεπιλογή, ένα σχήμα ανατίθεται στο πρώτο επίπεδο, ένας σύνδεσμος στο δεύτερο επίπεδο, ένα ψευδοαντικείμενο στο τρίτο επίπεδο κ.λπ.
- **Object is model base:** υποδεικνύει εάν το αντικείμενο θα πρέπει να λειτουργεί ως βάση ενός μοντέλου. Ένα αντικείμενο που έχει επισημανθεί ως βάση μοντέλου έχει ειδικές ιδιότητες (π.χ. αντιγραφή του αντικειμένου θα αντιγράψει επίσης αυτόματα όλο το δέντρο της ιεραρχίας του). Επιπλέον, όταν επιλέγεται ένα τέτοιο αντικείμενο, το πλαίσιο επιλογής εμφανίζεται ως παχιές διακεκομμένες γραμμές, περικλείοντας ολόκληρο το μοντέλο. Επιπλέον, ένα μοντέλο θα μοιράζεται τον ίδιο αναγνωριστικό με όλα τα αντίγραφα του ή τα κλώνου του. Ένα μοντέλο μπορεί να μεταφέρει το DNA του (δηλαδή να μοιράζεται μια παρουσία του εαυτού του) σε όλους τους κλώνους του, μέσω του κουμπιού μεταφοράς DNA της γραμμής εργαλείων. Ας φανταστούμε να έχουμε 100 ίδια ρομπότ στη σκηνή μας που θέλουμε να τροποποιήσουμε με παρόμοιο τρόπο: απλώς τροποποιούμε ένα από αυτά, το επιλέγουμε και μετά κάνουμε κλικ στο κουμπί μεταφοράς DNA της γραμμής εργαλείων.



- **Edit model properties:** επιτρέπει το άνοιγμα του διαλόγου μοντέλου.
- **Collidable:** επιτρέπει την ενεργοποίηση ή απενεργοποίηση της δυνατότητας ανίχνευσης σύγκρουσης για το επιλεγμένο συγκρούσιμο αντικείμενο.
- **Measurable:** επιτρέπει την ενεργοποίηση ή απενεργοποίηση της δυνατότητας υπολογισμού ελάχιστης απόστασης για το επιλεγμένο μετρήσιμο αντικείμενο..
- **Detectable:** επιτρέπει την ενεργοποίηση ή απενεργοποίηση της δυνατότητας ανίχνευσης από αισθητήρα εγγύτητας για το επιλεγμένο ανιχνεύσιμο αντικείμενο.
- **Collection self-collision indicator:** όταν εκτελείται ο υπολογισμός σύγκρουσης (ή ελάχιστης απόστασης) μεταξύ δύο πανομοιότυπων συλλογών, το CoppeliaSim θα ελέγχει κανονικά όλα τα στοιχεία της συλλογής με όλα τα άλλα στοιχεία αυτής της συλλογής. Σε ορισμένες περιπτώσεις, όπως μια κινηματική αλυσίδα, δεν θέλουμε να ελέγξουμε διαδοχικούς συνδέσμους, καθώς μπορεί να συγκρούονται συνεχώς στην διεπαφή. Σε αυτή την περίπτωση, μπορούμε να χρησιμοποιήσουμε τον δείκτη αυτοσύγκρουσης συλλογής: δύο στοιχεία της ίδιας συλλογής δεν θα ελεγχθούν μεταξύ τους αν η διαφορά των δεικτών τους είναι ακριβώς 1, 10, 100, 1000, 10000 ή 100000, όπως φαίνεται στο παρακάτω σχήμα:



1.4.5 Συγκρουόμενα αντικείμενα/Collidable objects

Συγκρουόμενα αντικείμενα είναι αντικείμενα που μπορούν να ελεγχθούν για σύγκρουση με άλλα συγκρουόμενα αντικείμενα, δηλαδή που θα καταγράψουν μια κατάσταση σύγκρουσης. Αυτό δεν

Προσομοίωση Ρομπότ μέσω Υπολογιστή

σημαίνει ότι θα αντιδράσουν στη σύγκρουση (δηλαδή ανταποκρίσιμα), κάτι που είναι διαφορετικό. Τα συγκρουόμενα αντικείμενα περιλαμβάνουν:

- Ομοιώματα/Dummies
- Σχήματα/Shapes
- Δέντρα OC/OC trees
- Νέφη σημείων/Point clouds

Τα ανδρείκελα και τα νέφη σημείων, επειδή βασίζονται σε σημεία, μπορούν να συγκρούονται μόνο με τα δέντρα OC (τα οποία βασίζονται σε όγκους).

Οι συλλογές είναι επίσης συγκρουόμενες, καθώς μπορεί να περιέχουν συγκρουόμενα αντικείμενα.

Τα συγκρουόμενα αντικείμενα μπορούν να έχουν την ιδιότητα σύγκρουσης μεμονωμένα ενεργοποιημένη ή απενεργοποιημένη (ενεργοποιημένη από προεπιλογή για μη πρωτόγονα σχήματα, δέντρα OC και νέφη σημείων). Αυτό μπορεί να οριστεί στις κοινές ιδιότητες αντικειμένων ή μέσω της συνάρτησης `API sim.setObjectSpecialProperty`.

Επιπλέον, τα συγκρουόμενα αντικείμενα μπορούν να έχουν την ιδιότητα σύγκρουσης υπερισχυμένη ανάλογα με τις σχετικές ιδιότητες μοντέλου (αν είναι μέρος ενός μοντέλου).

1.4.6 Μετρήσιμα αντικείμενα/Measurable objects

Μετρήσιμα αντικείμενα είναι αντικείμενα που μπορούν να χρησιμοποιηθούν για τον υπολογισμό της ελάχιστης απόστασης σε σχέση με άλλα μετρήσιμα αντικείμενα. Αυτά περιλαμβάνουν:

- Ανδρείκελα/Ομοιώματα/Dummies
- Σχήματα/Shapes
- Δέντρα OC/OC trees
- Νέφη σημείων/Point clouds

Οι συλλογές είναι επίσης μετρήσιμες, καθώς μπορεί να περιέχουν μετρήσιμα αντικείμενα.

Τα μετρήσιμα αντικείμενα μπορούν να έχουν την ιδιότητα μέτρησης μεμονωμένα ενεργοποιημένη ή απενεργοποιημένη (ενεργοποιημένη από προεπιλογή για μη πρωτόγονα σχήματα, δέντρα OC και νέφη σημείων). Αυτό μπορεί να οριστεί στις κοινές ιδιότητες αντικειμένων ή μέσω της συνάρτησης `API sim.setObjectSpecialProperty`.

Επιπλέον, τα μετρήσιμα αντικείμενα μπορούν να έχουν την ιδιότητα μέτρησης υπερισχυμένη ανάλογα με τις σχετικές ιδιότητες μοντέλου (αν είναι μέρος ενός μοντέλου).

1.4.7 Ανιχνεύσιμα αντικείμενα/Detectable objects

Ανιχνεύσιμα αντικείμενα είναι αντικείμενα που μπορούν να ανιχνευθούν από αισθητήρες εγγύτητας. Αυτά περιλαμβάνουν:

- Ανδρείκελα/Ομοιώματα/Dummies

Προσομοίωση Ρομπότ μέσω Υπολογιστή

- Σχήματα/Shapes
- Δέντρα OC/OC trees
- Νέφη σημείων/Point clouds

Τα ανδρείκελα και τα νέφη σημείων, επειδή βασίζονται σε σημεία, δεν μπορούν να ανιχνευθούν από αισθητήρες εγγύτητας τύπου ακτίνας ή τυχαίου τύπου.

Τα ανιχνεύσιμα αντικείμενα μπορούν να ανιχνευθούν από όλους τους αισθητήρες εγγύτητας, ή μόνο από συγκεκριμένους τύπους αισθητήρων εγγύτητας, ή από μια υποκατηγορία αισθητήρων εγγύτητας όπως αναφέρονται παρακάτω:

- Αισθητήρες εγγύτητας υπερήχων/Ultrasonic proximity sensors
- Αισθητήρες εγγύτητας υπέρυθρων/Infrared proximity sensors
- Αισθητήρες εγγύτητας λέιζερ/Laser proximity sensors
- Επαγωγικοί αισθητήρες εγγύτητας/Inductive proximity sensors
- Χωρητικοί αισθητήρες εγγύτητας/Capacitive proximity sensors

Οι συλλογές είναι επίσης ανιχνεύσιμες, καθώς μπορεί να περιέχουν ανιχνεύσιμα αντικείμενα.

Τα ανιχνεύσιμα αντικείμενα μπορούν να έχουν την ιδιότητα ανιχνευσιμότητας μεμονωμένα ενεργοποιημένη ή απενεργοποιημένη, και αυτό για όλους τους τύπους αισθητήρων εγγύτητας (ενεργοποιημένη από προεπιλογή για μη πρωτόγονα σχήματα). Αυτό μπορεί να οριστεί στις κοινές ιδιότητες αντικειμένων ή μέσω της συνάρτησης `API sim.setObjectSpecialProperty`.

1.4.8 Ορατά αντικείμενα/Viewable objects

Τα ορατά αντικείμενα είναι αντικείμενα που μπορούμε να δούμε μέσα τους, να τα δούμε είτε να προβάλλουν κάποιο περιεχόμενο εικόνας. Αυτά περιλαμβάνουν:

- Κάμερες/Cameras
- Αισθητήρες όρασης/Vision sensors

Τα αντικείμενα που μπορούν να προβληθούν μπορούν να συσχετιστούν με προβολές που θα εμφανίζουν το περιεχόμενο της εικόνας τους.

1.4.9 Convex hull

Το CoppeliaSim επιτρέπει τον υπολογισμό και την προσθήκη της κυρτής θήκης (convex hull) συγκεκριμένων αντικειμένων. Η κυρτή θήκη ενός συνόλου πλεγμάτων (meshes) είναι το μικρότερο κυρτό περίβλημα που περικλείει όλα τα πλέγματα. Το CoppeliaSim επιτρέπει την εξαγωγή της κυρτής θήκης μόνο μετρήσιμων αντικειμένων.

Για να προσθέσουμε μια κυρτή θήκη, επιλέγουμε τα αντικείμενα που θέλουμε να περιληφθούν στην κυρτή θήκη και επιλέγουμε [Modules → Geometry / Mesh → Convex hull...].

Ανάλογα με τις καταστάσεις, οι κυρτές θήκες μπορούν να μειώσουν δραστικά τους χρόνους υπολογισμού σε περιπτώσεις υπολογισμών απόστασης ή προσομοιώσεων αισθητήρων εγγύτητας. Ο εντοπισμός σύγκρουσης μπορεί επίσης να πραγματοποιηθεί γρηγορότερα αν χρησιμοποιηθεί μία πιο

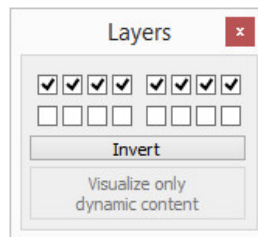
Προσομοίωση Ρομπότ μέσω Υπολογιστή

πρόχειρη προσέγγιση (δηλαδή η κυρτή θήκη) ενός σύνθετου σχήματος. Μπορούμε επίσης να εξαγάγουμε την κυρτή θήκη ενός μοντέλου (π.χ. ρομπότ), να την κάνουμε αόρατη, αλλά να εκτελέσουμε όλους τους υπολογισμούς (εντοπισμός σύγκρουσης, υπολογισμοί απόστασης, κ.λπ.) χρησιμοποιώντας την αόρατη κυρτή θήκη αντί του σύνθετου μοντέλου.

1.4.10 Παράθυρο διαλόγου επιλογής επιπέδων/Layer selection dialog

Το παράθυρο διαλόγου επιλογής επιπέδων είναι ένας βολικός τρόπος για να εμφανίσουμε ή να αποκρύψουμε συγκεκριμένα μέρη μιας σκηνής. Κάθε αντικείμενο σκηνής μπορεί να αντιστοιχιστεί σε οποιοδήποτε από τα 16 διαθέσιμα επίπεδα ορατότητας. Χρησιμοποιώντας αυτό το παράθυρο διαλόγου, μπορούμε να ενεργοποιήσουμε/απενεργοποιήσουμε κάθε επίπεδο ξεχωριστά. Αν κάποια αντικείμενα πρέπει να είναι αόρατα, μπορούμε δοκιμάσουμε να τα αποκρύψουμε σε ένα μη ενεργοποιημένο επίπεδο (π.χ. τρέχον επίπεδο+8). Όταν αποθηκεύουμε μια σκηνή, πάντα προσπαθούμε να παραμείνουμε στην προεπιλεγμένη κατάσταση ενεργοποίησης (όλα τα 8 πρώτα επίπεδα ενεργοποιημένα, τα τελευταία 8 επίπεδα απενεργοποιημένα). Ο λόγος γι' αυτό είναι ότι αν αντιγράψουμε και επικολλήσουμε ένα μοντέλο από τη σκηνή μας σε άλλη σκηνή, τότε κάποια αντικείμενα που υποτίθεται ότι ήταν αόρατα μπορεί ξαφνικά να γίνουν ορατά. Κατά τη διάρκεια της προσομοίωσης, το παράθυρο διαλόγου προσφέρει τη δυνατότητα να εμφανίζεται μόνο το δυναμικό περιεχόμενο της σκηνής (Οπτικοποίηση μόνο δυναμικού περιεχομένου) το οποίο συχνά είναι χρήσιμο για σκοπούς εντοπισμού σφαλμάτων.

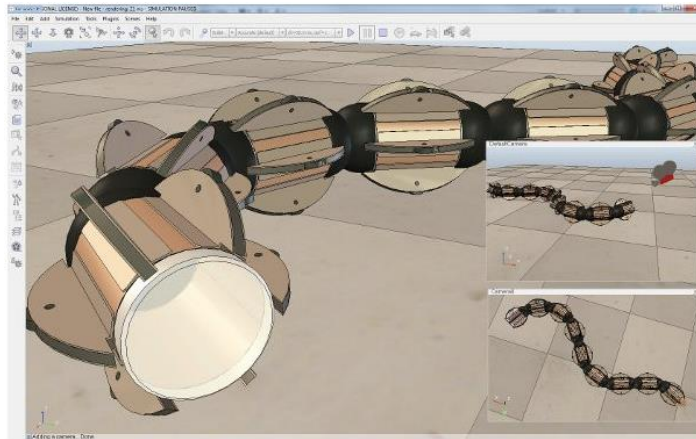
Το παράθυρο διαλόγου επιλογής επιπέδων μπορεί να προσπελαστεί με "Tools → Layers" ή κάνοντας κλικ στο ακόλουθο κουμπί της γραμμής εργαλείων:



1.4.11 Κάμερες/Cameras



Οι κάμερες είναι ορατά αντικείμενα, πράγμα που σημαίνει ότι μπορούμε να κοιτάξουμε μέσα από αυτές και να εμφανίσουμε μια προβολή του τι βλέπουν. Μπορούμε να έχουμε όσες κάμερες χρειαζόμαστε στη σκηνή μας, καθεμία από αυτές προσφέροντας μια διαφορετική προβολή της σκηνής. Το παρακάτω δείχνει ένα παράδειγμα μιας σκηνής που περιέχει πολλές κάμερες:



Πρέπει να είμαστε σίγουροι να μη μπερδεύουμε τις κάμερες με τους αισθητήρες όρασης. Οι κύριες διαφορές:

- Μια κάμερα δεν έχει συγκεκριμένη ανάλυση (δηλαδή προσαρμόζεται αυτόματα στο μέγεθος της προβολής). Ένας αισθητήρας όρασης έχει σταθερή ανάλυση.
- Το περιεχόμενο της εικόνας μιας κάμερας δεν είναι άμεσα διαθέσιμο μέσω του API (αλλά μέσω ενός μηχανισμού callback) και η επεξεργασία εικόνας δεν υποστηρίζεται άμεσα. Το περιεχόμενο της εικόνας ενός αισθητήρα όρασης μπορεί να προσπελαστεί μέσω του API και να επεξεργαστεί μέσω των συναρτήσεων callback όρασης.
- Μια κάμερα γενικά απαιτεί λιγότερο χρόνο CPU και λειτουργεί ταχύτερα από τους αισθητήρες όρασης.
- Μια κάμερα μπορεί να εμφανίσει όλους τους τύπους αντικειμένων. Ένας αισθητήρας όρασης κυρίως εμφανίζει σχήματα.

Μια κάμερα μπορεί να προστεθεί στη σκηνή με [Add → Camera]. Ωστόσο, η προσθήκη μιας κάμερας με αυτόν τον τρόπο θα προσθέσει το αντικείμενο σε μια προεπιλεγμένη στάση. Είναι καλύτερη πρακτική να προσθέτουμε κάμερες κάνοντας δεξί κλικ σε μια συγκεκριμένη προβολή μιας σκηνής και επιλέγοντας [Popup menu → Add → Camera]. Αυτό θα προσθέσει το αντικείμενο ακριβώς μπροστά από την τρέχουσα προβολή. Η προστιθέμενη κάμερα επιλέγεται αυτόματα και μπορούμε στη συνέχεια να κοιτάξουμε μέσα από αυτή με [Popup menu → View → Associate view with selected camera]. Για να λειτουργήσει αυτό, θα πρέπει να βεβαιωθούμε ότι το αναδυόμενο μενού είναι ενεργοποιημένο σε μια προβολή. Όταν δημιουργείται μια προβολή αλλά δεν έχει συσχετιστεί ακόμα με ένα ορατό αντικείμενο, η εντολή [Popup menu → Add → Camera] θα προσθέσει μια κάμερα και θα τη συσχετίσει άμεσα με την προβολή (δηλαδή, θα κοιτάξουμε μέσα από αυτή). Μια κάμερα μπορεί να συσχετιστεί με οποιοδήποτε αριθμό προβολών.

Εναλλακτικά, μπορούμε επίσης να συσχετίσουμε μία προβολή με μια κάμερα ενεργοποιώντας το παρακάτω αναδυόμενο μενού σε μια προβολή: [Popup menu → View → View selector...]. Αυτό θα μας επιτρέψει να επιλέξουμε την επιθυμητή προβολή κάμερας από ένα παράθυρο προεπισκόπησης. Όταν μια κάμερα συσχετίζεται με μια προβολή, μπορεί να χειριστεί με το ποντίκι με τα παρακάτω κουμπιά της γραμμής εργαλείων:



- **Pan button/Κουμπί μετατόπισης:** επιτρέπει τη μετατόπιση της κάμερας σε ένα κάθετο επίπεδο προς την κατεύθυνση της προβολής. Το σημείο που κάνουμε κλικ στη σκηνή θα μετατοπιστεί με τον ίδιο ρυθμό με την κίνηση του ποντικιού. Αν δεν επιλεγεί κάποιο

Προσομοίωση Ρομπότ μέσω Υπολογιστή

συγκεκριμένο σημείο (δηλαδή όταν γίνεται κλικ στο φόντο) η λειτουργία μετατόπισης θα εκτελεστεί για ένα αόρατο σημείο κοντά στην κάμερα.

- **Rotate button/Κουμπί περιστροφής:** επιτρέπει την περιστροφή της κάμερας γύρω από το σημείο που επιλέχθηκε στη σκηνή. Αν δεν επιλεγεί κάποιο συγκεκριμένο σημείο (δηλαδή όταν γίνεται κλικ στο φόντο) η λειτουργία περιστροφής θα εκτελεστεί γύρω από ένα αόρατο σημείο κοντά στην κάμερα.
- **Shift button/Κουμπί ολίσθησης:** επιτρέπει τη μετατόπιση της κάμερας κατά μήκος της κατεύθυνσης της προβολής. Αυτό έχει ένα εφέ ζουμ σε προβολές κάμερας που βρίσκονται σε λειτουργία προβολής προοπτικής, αλλά οι προβολές κάμερας σε λειτουργία ορθογωνίας προβολής μπορεί να μην δουν καμία διαφορά (οι κάμερες ωστόσο θα μετατοπιστούν κατά μήκος της κατεύθυνσης της προβολής τους).
- **Camera angle button/Κουμπί γωνίας κάμερας:** επιτρέπει την προσαρμογή της γωνίας προβολής προοπτικής μιας προβολής κάμερας (όταν βρίσκεται σε λειτουργία προβολής προοπτικής) ή επιτρέπει την προσαρμογή του μεγέθους προβολής μιας προβολής κάμερας (όταν βρίσκεται σε λειτουργία ορθογωνίας προβολής).
- **Fit-to-view button/Κουμπί προσαρμογής στην προβολή:** προσαρμόζει την κάμερα της προβολής που έχει την εστίαση ώστε να πλαισιώσει ωραία όλα τα επιλεγμένα αντικείμενα ή ολόκληρη τη σκηνή αν δεν έχει επιλεγεί κάποιο αντικείμενο.

Ενώ το αριστερό κουμπί του ποντικιού επιτρέπει τις παραπάνω λειτουργίες, ο τροχός του ποντικιού επιτρέπει τη μετατόπιση της κάμερας κατά μήκος της κατεύθυνσης της προβολής (αν η προβολή της κάμερας είναι σε λειτουργία προοπτικής προβολής) ή την προσαρμογή του μεγέθους της προβολής της κάμερας (αν η προβολή της κάμερας είναι σε λειτουργία ορθογωνίας προβολής). Το δεξί κουμπί, όταν γίνεται κλικ χωρίς κίνηση του ποντικιού, θα ενεργοποιήσει ένα αναδυόμενο μενού. Ωστόσο, αν το δεξί κουμπί παραμείνει πατημένο ενώ μετακινείτε το ποντίκι, μπορεί να επιτευχθεί περιστροφή της κάμερας γύρω από το σημείο που κάναμε κλικ.

Όταν το κουμπί μετατόπισης της κάμερας είναι ενεργοποιημένο, είναι πολύ εύκολο να περιηγηθούμε στη σκηνή σε οποιαδήποτε επιθυμητή στάση χρησιμοποιώντας:

- 1) το αριστερό κουμπί του ποντικιού για μετατόπιση κάθετα προς την κατεύθυνση της προβολής
- 2) τον τροχό του ποντικιού για μετατόπιση κατά μήκος της κατεύθυνσης της προβολής
- 3) το δεξί κουμπί του ποντικιού για περιστροφή γύρω από το σημείο που κάναμε κλικ



[Default camera navigation: (1) pan with left mouse button, (2) shift with mouse wheel, (3) rotate with right mouse button]

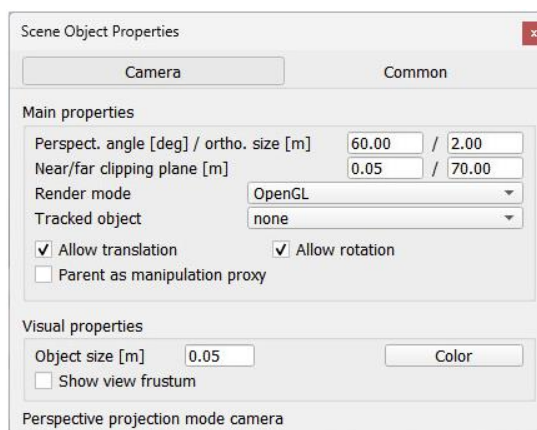
Οι κάμερες μπορούν να παρακολουθούν αυτόματα αντικείμενα (δηλαδή να τα ακολουθούν καθώς κινούνται) επιλέγοντας το αντικείμενο για παρακολούθηση και στη συνέχεια επιλέγοντας στην κατάλληλη προβολή [Pop-up menu → View → Track selected object]. Μια κάμερα που παρακολουθεί θα διατηρεί τη θέση της, αλλά θα προσαρμόζει αυτόματα την κατεύθυνση της προβολής της ώστε να κρατά πάντα το αντικείμενο προς παρακολούθηση στο πεδίο της προβολής της. Αυτό είναι διαφορετικό από το να έχει μια κάμερα συνδεδεμένη με αυτό το αντικείμενο μέσω μιας σχέσης γονέα-παιδιού.

1.4.12 Ιδιότητες κάμερας/Camera properties

Οι ιδιότητες της κάμερας είναι μέρος του διαλόγου ιδιοτήτων αντικειμένων σκηνής, που βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το παράθυρο διαλόγου με διπλό κλικ στο εικονίδιο του αντικειμένου στην ιεραρχία της σκηνής ή με κλικ στο κουμπί της γραμμής εργαλείων:



Στον διάλογο ιδιοτήτων αντικειμένων σκηνής, κάνουμε κλικ στο κουμπί Camera για να εμφανίσουμε το παράθυρο διαλόγου της κάμερας (το κουμπί Camera εμφανίζεται μόνο αν η τελευταία επιλογή είναι κάμερα). Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους της τελευταίας επιλεγμένης κάμερας. Εάν είναι επιλεγμένες περισσότερες από μία κάμερες, τότε ορισμένες παράμετροι μπορούν να αντιγραφούν από την τελευταία επιλεγμένη κάμερα στις άλλες επιλεγμένες κάμερες (Apply to selection):



- **Γωνία προοπτικής/Perspective angle:** η μέγιστη γωνία ανοίγματος του τραπεζοειδούς της προβολής, όταν η κάμερα βρίσκεται σε λειτουργία προοπτικής προβολής.
- **Ορθογραφικό μέγεθος/Orthographic size:** το μέγεθος της ορθογραφικής προβολής (ή το μέγεθος προβολής), όταν η κάμερα βρίσκεται σε λειτουργία ορθογραφικής προβολής.
- **Εγγύς / απομακρυσμένο επίπεδο αποκοπής/Near / far clipping plane:** απόσταση από το σημείο προέλευσης της κάμερας από την οποία η κάμερα αρχίζει / σταματά να βλέπει. Αν τα κοντινά / μακρινά αντικείμενα δεν πρέπει να εμφανίζονται, αυξάνουμε / μειώνουμε αυτήν την τιμή. Αυτές οι παράμετροι συνδέονται άμεσα με την ακρίβεια της απόδοσης, ειδικά όταν μια συσχετισμένη προβολή είναι σε λειτουργία προοπτικής προβολής, τότε πρέπει πάντα να είμαστε προσεκτικοί ώστε να μην υπάρχει πολύ μεγάλο χάσμα μεταξύ των απομακρυσμένων και κοντινών επιπέδων αποκοπής, διαφορετικά μπορεί να αντιμετωπίσουμε μερικά οπτικά τεχνουργήματα γνωστά ως z-fighting (όταν επιφάνειες ή pixels που βρίσκονται πιο μακριά φαίνεται να αλληλεπικαλύπτονται με επιφάνειες/pixel πιο κοντά στην κάμερα).
- **Λειτουργία απόδοσης/Render mode:** η λειτουργία απόδοσης της κάμερας.
- **Παρακολουθούμενο αντικείμενο/Tracked object:** αντικείμενο που θα παρακολουθείται από την κάμερα. Η παρακολούθηση ενός αντικειμένου δεν θα επηρεάσει τη θέση της κάμερας, μόνο ο προσανατολισμός της θα προσαρμοστεί αυτόματα για να ακολουθεί το παρακολουθούμενο αντικείμενο.
- **Επιτρέψτε τη μετάφραση/Allow translation:** επιτρέπονται οι κινήσεις μετάφρασης της κάμερας.

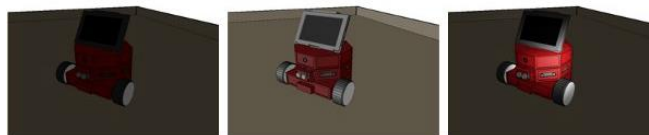
Προσομοίωση Ρομπότ μέσω Υπολογιστή

- **Επιτρέψτε την περιστροφή/Allow rotation:** επιτρέπονται οι κινήσεις περιστροφής της κάμερας.
- **Γονέας ως χειριστήριο αντιπροσώπευσης/Parent as manipulation proxy:** αν είναι ενεργοποιημένο, τότε τα κουμπιά της γραμμής εργαλείων χειρισμού της κάμερας δεν θα επηρεάζουν άμεσα τη θέση της κάμερας, αλλά θα επηρεάζουν το αντικείμενο-γονέα της κάμερας. Αυτή η επιλογή είναι χρήσιμη αν θέλουμε να έχουμε πολλές κάμερες συνδεδεμένες μεταξύ τους (π.χ. η προεπιλεγμένη σκηνή στο CoppeliaSim έχει 3 κάμερες διατεταγμένες ορθογώνια μεταξύ τους. Αν μία μετακινηθεί, οι δύο άλλες θα ακολουθήσουν).
- **Μέγεθος αντικειμένου/Object size:** μέγεθος της κάμερας. Αυτή η παράμετρος έχει μόνο οπτικό αποτέλεσμα και καμία λειτουργική σημασία.
- **Χρώμα/Color:** επιτρέπει την προσαρμογή του χρώματος της κάμερας.
- **Εμφάνιση του όγκου προβολής/Show view frustum:** αν επιλεγεί, εμφανίζεται ο όγκος προβολής.

1.4.13 Φώτα/Lights



Φώτα είναι αντικείμενα που επιτρέπουν να φωτίσετε μια σκηνή. Χωρίς κανένα φως στη σκηνή, τα αντικείμενα εμφανίζονται σε αχρωμάτιστα χρώματα (σε αυτή την περίπτωση τα αντικείμενα φωτίζονται μόνο με ένα περιβάλλον φως που μπορεί να ρυθμιστεί στο διάλογο περιβάλλοντος). Η παρακάτω εικόνα απεικονίζει την επίδραση των φώτων σε μια σκηνή:



[Scene illuminated (1) with ambient light only, (2) with a directional light, (3) with a spotlight]

Μπορεί να υπάρχουν έως οκτώ ενεργά φώτα σε μια σκηνή. Υπάρχουν τρεις διαφορετικοί τύποι φώτων:

- **Πολυκατευθυντικά φώτα/Omnidirectional lights:** ένα πολυκατευθυντικό φως θα φωτίσει τη σκηνή από μία σημειακή πηγή φωτός.
- **Φώτα σημείου/Spotlights:** ένα φως σημείου θα φωτίσει τη σκηνή μόνο προς μία συγκεκριμένη κατεύθυνση. Μπορούμε να ρυθμίσουμε τον κώνο φωτός ενός σημείου φωτός.
- **Κατευθυντικά φώτα/Directional lights:** η θέση ενός κατευθυντικού φωτός δεν είναι σημαντική. Μόνο η κατεύθυνση έχει σημασία και όλα τα αντικείμενα σε μια σκηνή θα φωτιστούν με παρόμοιο τρόπο.



[(1) omnidirectional light, (2) spotlight, (3) directional light]

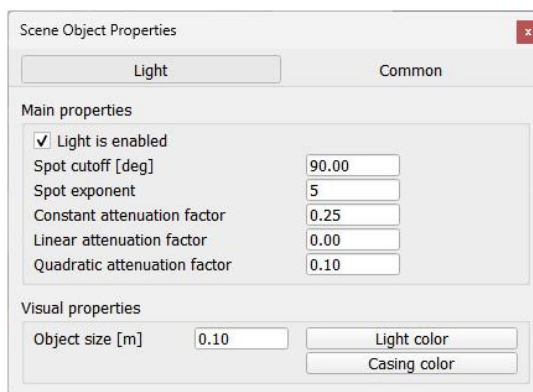
Ένα φως μπορεί να προστεθεί στη σκηνή με [Add → Light].

1.4.14 Ιδιότητες φωτός/Light properties

Οι ιδιότητες φωτός είναι μέρος του παραθύρου διαλόγου ιδιοτήτων αντικειμένων σκηνής, ο οποίος βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το παράθυρο διαλόγου με διπλό κλικ σε ένα εικονίδιο αντικειμένου στην ιεραρχία της σκηνής ή με κλικ στο κουμπί της γραμμής εργαλείων:



Στο παράθυρο διαλόγου ιδιοτήτων αντικειμένων σκηνής, κάνουμε κλικ στο κουμπί Light για να εμφανιστεί το παράθυρο διαλόγου φωτός (το κουμπί Light εμφανίζεται μόνο αν η τελευταία επιλογή είναι φως). Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους του τελευταίου επιλεγμένου φωτός. Αν είναι επιλεγμένα περισσότερα από ένα φώτα, τότε ορισμένες παράμετροι μπορούν να αντιγραφούν από το τελευταίο επιλεγμένο φως στα άλλα επιλεγμένα φώτα (Apply to selection κουμπιά):



- **Το φως είναι ενεργοποιημένο/Light is enabled:** ενεργοποιεί ή απενεργοποιεί ένα φως. Μέγιστος αριθμός ενεργοποιημένων φώτων είναι 8 σε μια σκηνή ταυτόχρονα.
- **Το φως είναι τοπικό/Light is local:** όταν αυτό το στοιχείο είναι επιλεγμένο, τότε το φως θα λειτουργεί ως τοπικό φως και θα λειτουργεί μόνο όταν φαίνεται μέσω κάμερας ή αισθητήρα όρασης που χρησιμοποιεί τοπικά φώτα (δηλ. το στοιχείο χρήσης τοπικών φώτων της κάμερας ή του αισθητήρα όρασης πρέπει να είναι ενεργοποιημένο). Επιπλέον, το φως πρέπει να είναι κατασκευασμένο πάνω από την κάμερα ή τον αισθητήρα όρασης (το φως να είναι άμεσο ή έμμεσο παιδί).
- **Spot cutoff:** στην περίπτωση που το φως είναι φως σημείου, αυτή η τιμή καθορίζει τη γωνία αποκοπής του φωτός. Μικρές γωνίες έχουν ως αποτέλεσμα στενά φώτα σημείου.
- **Spot exponent:** στην περίπτωση που το φως είναι φως σημείου, αυτή η τιμή καθορίζει την κατανομή έντασης του φωτός. Η τιμή 0 έχει ως αποτέλεσμα ένα ευρύ φως σημείου, ενώ η τιμή 128 έχει ως αποτέλεσμα ένα στενό φως σημείου.
- **Σταθερός παράγοντας απόσβεσης/Constant attenuation factor:** παράμετρος που υποδεικνύει τον σταθερό παράγοντα απόσβεσης του φωτός.
- **Γραμμικός παράγοντας απόσβεσης/Linear attenuation factor:** παράμετρος που υποδεικνύει το γραμμικό παράγοντα απόσβεσης του φωτός.






Προσομοίωση Ρομπότ μέσω Υπολογιστή


- **Τετραγωνικός παράγοντας απόσβεσης/Quadratic attenuation factor:** παράμετρος που υποδεικνύει τον τετραγωνικό παράγοντα απόσβεσης του φωτός.
- **Μέγεθος αντικειμένου/Object size:** μέγεθος του φωτός. Αυτή η παράμετρος έχει μόνο οπτική επίδραση και καμία λειτουργική σημασία.
- **Χρώμα φωτός / περιβλήματος/ Light / casing color:** επιτρέπει τη ρύθμιση του χρώματος του φωτός ή του χρώματος του περιβλήματος.

1.4.15 Σχήματα/Shapes



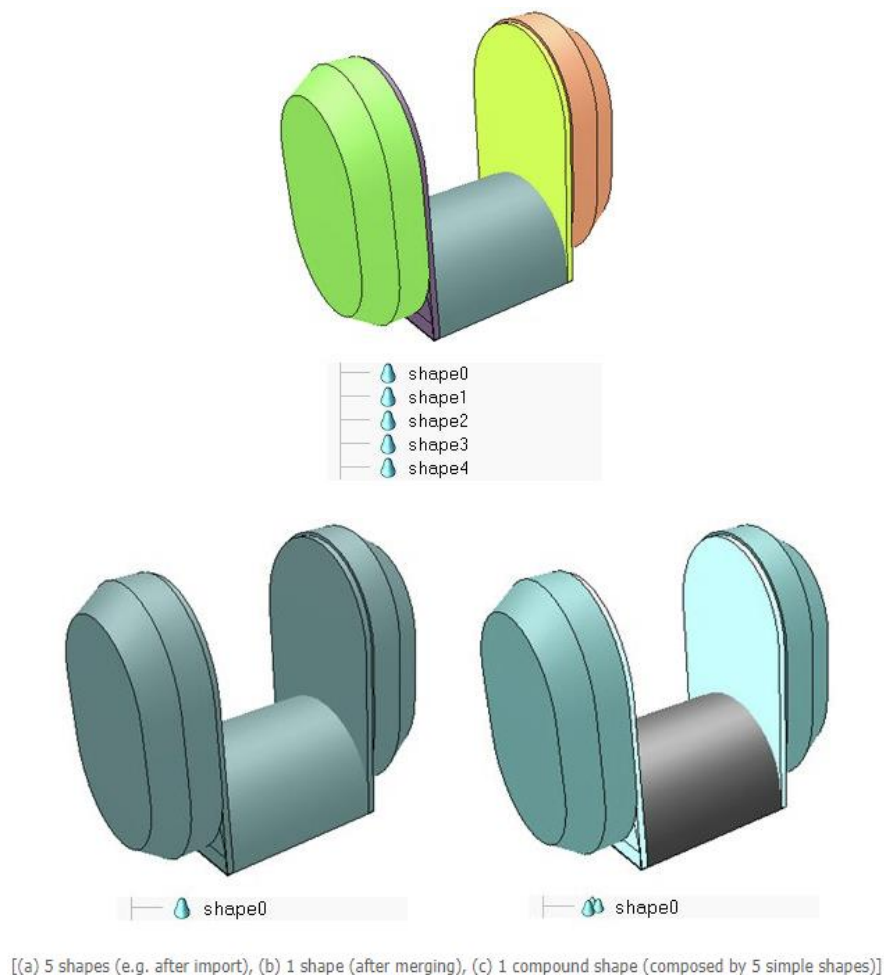
Τα σχήματα είναι άκαμπτα αντικείμενα πλέγματος που αποτελούνται από τριγωνικές όψεις. Μπορούν να εισαχθούν/εξαχθούν και να επεξεργαστούν. Υπάρχουν τέσσερις διαφορετικοί υποτύποι:

-  **Απλό σχήμα/Simple shape:** μπορεί να αναπαραστήσει οποιοδήποτε πλέγμα. Έχει ένα χρώμα και ένα σύνολο οπτικών χαρακτηριστικών. Δεν είναι βελτιστοποιημένο ούτε συνιστάται για υπολογισμούς δυναμικής απόκρισης σύγκρουσης (επειδή είναι πολύ αργό και ασταθές).
-  **Σύνθετο σχήμα/Compound shape:** μπορεί να αναπαραστήσει οποιοδήποτε πλέγμα. Έχει πολλά χρώματα και σύνολα οπτικών χαρακτηριστικών. Δεν είναι βελτιστοποιημένο ούτε συνιστάται για υπολογισμούς δυναμικής απόκρισης σύγκρουσης (επειδή είναι πολύ αργό και ασταθές).
-  **Κυρτό σχήμα/Convex shape:** αναπαριστά ένα κυρτό πλέγμα με ένα χρώμα και ένα σύνολο οπτικών χαρακτηριστικών. Βελτιστοποιημένο για υπολογισμούς δυναμικής απόκρισης σύγκρουσης (αλλά τα πρωτόγονα σχήματα συνιστώνται).
-  **Σύνθετο κυρτό σχήμα/Compound convex shape:** αναπαριστά μια ομάδα κυρτών πλεγμάτων με πολλά χρώματα και σύνολα οπτικών χαρακτηριστικών. Βελτιστοποιημένο για υπολογισμούς δυναμικής απόκρισης σύγκρουσης (αλλά τα πρωτόγονα σύνθετα σχήματα συνιστώνται).
-  **Πρωτόγονο σχήμα/Primitive shape:** αναπαριστά ένα πρωτόγονο σχήμα (ορθογώνιο παραλληλεπίπεδο, κύλινδρο ή σφαίρα (με πρόσθετες παραλλαγές ανάλογα με τη χρησιμοποιούμενη μηχανή φυσικής)). Ένα πρωτόγονο σχήμα (ή πρωτόγονο σύνθετο σχήμα) είναι ιδανικό για υπολογισμούς δυναμικής απόκρισης σύγκρουσης, καθώς θα εκτελείται πολύ γρήγορα και είναι σταθερό.
-  **Πρωτόγονο σύνθετο σχήμα/Primitive compound shape:** αναπαριστά μια ομάδα πρωτόγονων σχημάτων (ορθογώνιο παραλληλεπίπεδο, κύλινδρο ή σφαίρα). Ένα πρωτόγονο σύνθετο σχήμα (ή πρωτόγονο σχήμα) είναι ιδανικό για υπολογισμούς δυναμικής απόκρισης σύγκρουσης, καθώς θα εκτελείται πολύ γρήγορα και είναι σταθερό. Δείτε την ενότητα δυναμικής για περισσότερες πληροφορίες.

-  **Σχήμα ύψους/Heightfield shape:** μπορεί να αναπαραστήσει ένα έδαφος ως κανονικό πλέγμα, όπου μόνο τα ύψη αλλάζουν. Τα σχήματα ύψους μπορούν επίσης να θεωρηθούν ως πρωτόγονα σχήματα και είναι βελτιστοποιημένα για υπολογισμούς δυναμικής απόκρισης σύγκρουσης.

Από προεπιλογή, όλα τα εισαγόμενα σχήματα είναι απλά σχήματα. Ωστόσο, δύο ή περισσότερα σχήματα ή σύνθετα σχήματα μπορούν να ομαδοποιηθούν/απο-ομαδοποιηθούν με [Edit → Shape grouping/merging → group/ungroup]. Τα απλά σχήματα μπορούν επίσης να συγχωνευθούν/διααιρεθούν με [Edit → Shape grouping/merging → merge/divide].

Τα πρωτόγονα σχήματα είναι κυρίως λειτουργικά σχήματα. Τις περισσότερες φορές χρησιμοποιούνται μόνο από τη μηχανή φυσικής, η οποία αποδίδει πολύ καλύτερα και γρηγορότερα σε αυτά από ό,τι σε μη πρωτόγονα σχήματα (π.χ. τυχαία ή κυρτά πλέγματα). Για αυτόν τον λόγο, τα πρωτόγονα σχήματα συχνά κρύβονται σε ένα αόρατο επίπεδο (π.χ. επίπεδο 9).



Τα πρωτόγονα σχήματα μπορούν επίσης να ομαδοποιηθούν, αλλά το αποτέλεσμα σύνθετου σχήματος θα είναι πρωτόγονο μόνο εάν όλα τα συστατικά του στοιχεία είναι επίσης πρωτόγονα. Η συγχώνευση πρωτογόνων σχημάτων θα οδηγήσει σε ένα μη πρωτόγονο σχήμα.

Τα σχήματα είναι συγκρουόμενα (collidable), μετρήσιμα (measuerable) και ανιχνεύσιμα (detectable) αντικείμενα. Αυτό σημαίνει ότι τα σχήματα μπορούν:

- Να χρησιμοποιηθούν για ανίχνευση συγκρούσεων με άλλα κινητά αντικείμενα.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

- Να χρησιμοποιηθούν για υπολογισμό ελάχιστης απόστασης με άλλα μετρήσιμα αντικείμενα.
- Να ανιχνευθούν από αισθητήρες πλησιάζματος.

Οι ιδιότητες "collidable", "measurable" και "detectable" ενός σχήματος μπορούν να τροποποιηθούν στις κοινές ιδιότητες του αντικειμένου (object common properties). Επιπλέον, αυτές οι ιδιότητες μπορούν να αντικατασταθούν αν το σχήμα είναι μέρος ενός μοντέλου που τις αντικαθιστά.

Οι υπολογισμοί σχημάτων (δηλαδή υπολογισμοί συγκρούσεων, απόστασης και αισθητήρων πλησιάζματος) που είναι διαθέσιμοι στο CorreliaSim είναι επίσης διαθέσιμοι ως αυτόνομες ρουτίνες μέσω των γεωμετρικών ρουτινών του Correlia.

1.4.16 Αναφορικό πλαίσιο σχήματος και κουτί οριοθέτησης/Shape reference frame and bounding box

Ένα σχήμα έχει ένα αναφορικό πλαίσιο και ένα κουτί οριοθέτησης, όπως και κάθε αντικείμενο. Το αναφορικό πλαίσιο του σχήματος μπορεί να μετακινηθεί και το κουτί οριοθέτησης μπορεί να τροποποιηθεί, χωρίς να επηρεαστούν οι άλλες ιδιότητες του σχήματος.

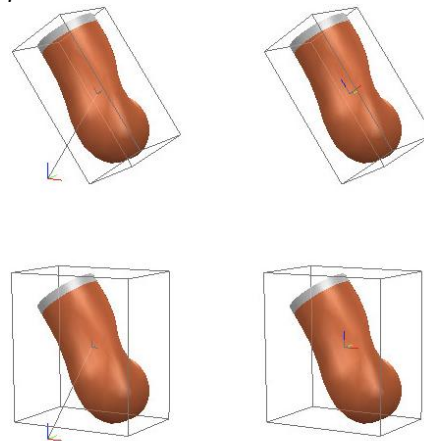
Το αναφορικό πλαίσιο των επιλεγμένων σχημάτων μπορεί να μετακινηθεί με δύο διαφορετικούς τρόπους, μέσω [Edit → Shape reference frame]:

- **μετακίνηση στην αρχή του κόσμου/relocate to world origin:** το αναφορικό πλαίσιο θα μετακινηθεί στην αρχή του κόσμου
- **μετακίνηση στο κέντρο του πλέγματος/relocate to mesh center:** το αναφορικό πλαίσιο θα μετακινηθεί στο κέντρο του κουτιού οριοθέτησης

Οι ιδιότητες αδράνειας του σχήματος δεν επηρεάζονται όταν το αναφορικό πλαίσιο μετακινείται, αλλά ο πίνακας αδράνειας και το κέντρο μάζας θα εκφράζονται σε σχέση με το νέο αναφορικό πλαίσιο.

Το κουτί οριοθέτησης των επιλεγμένων σχημάτων μπορεί να τροποποιηθεί με δύο διαφορετικούς τρόπους, μέσω [Edit → Shape bounding box]:

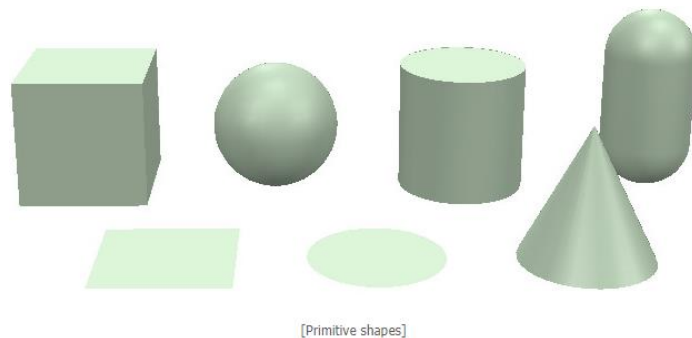
- **ευθυγράμμιση με τον κόσμο/align with world:** το κουτί οριοθέτησης θα ευθυγραμμιστεί με το αναφορικό πλαίσιο του κόσμου
- **ευθυγράμμιση με το πλέγμα/align with mesh:** το κουτί οριοθέτησης θα επιλεγεί έτσι ώστε να ελαχιστοποιηθεί ο όγκος του



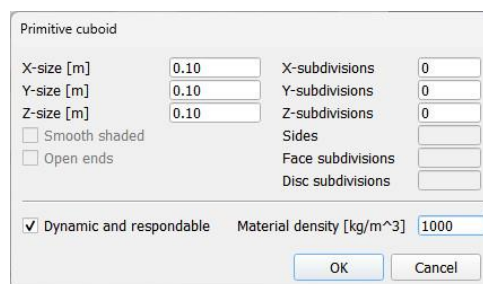
[Possible reference frame locations and bounding box orientations for a shape]

1.4.17 Πρωτόγονα σχήματα/Primitive shapes

Εκτός από την εισαγωγή πλεγμάτων από ένα εξωτερικό αρχείο, τα σχήματα μπορούν επίσης να δημιουργηθούν απευθείας μέσα στο CoppeliaSim επιλέγοντας [Add → Primitive shape]:



Οι γεωμετρικές παράμετροι των πρωτόγονων σχημάτων μπορούν να ρυθμιστούν στο παράθυρο διαλόγου πρωτόγονου σχήματος:

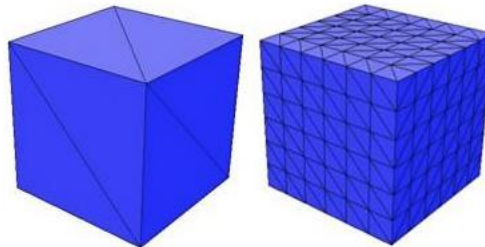


- **X- / Y- / Z-μέγεθος/size:** το μέγεθος κατά τον άξονα x - / y - / z - του αναφορικού πλαισίου του κόσμου.
- **X- / Y- / Z-υποδιαιρέσεις/subdivisions:** ο αριθμός των στοιχείων κατά τον άξονα x - / y - / z - του αναφορικού πλαισίου του κόσμου.
- **Ομαλή σκίαση/Smooth shaded:** αν το σχήμα πρέπει να εμφανίζεται ομαλό (σκίαση Gouraud) ή αιχμηρό.
- **Ανοιχτά άκρα/Open ends:** αν ο κύλινδρος έχει ανοιχτά άκρα.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

- **Πλευρές/Sides:** ο αριθμός των πλευρών ενός κυλίνδρου, δίσκου ή σφαίρας.
- **Υποδιαιρέσεις προσώπου/Face subdivisions:** ο αριθμός των υποδιαιρέσεων προσώπου ενός κυλίνδρου.
- **Υποδιαιρέσεις δίσκου/Disc subdivisions:** ο αριθμός των υποδιαιρέσεων δίσκου (δίσκος ή κύλινδρος).
- **Δυναμικό και ανταποκρινόμενο/Dynamic and respondable:** αν επιλεγεί, το δημιουργούμενο σχήμα θα είναι δυναμικά ενεργοποιημένο, δηλαδή δυναμικό και ανταποκρινόμενο. Αυτές οι ιδιότητες μπορούν να αλλάξουν αργότερα.
- **Πυκνότητα υλικού/Material density:** η πυκνότητα του υλικού.

Οι υποδιαιρέσεις και οι ιδιότητες προσώπων δεν μπορούν να φανούν άμεσα και δεν έχουν άμεσο αποτέλεσμα. Στην ακόλουθη εικόνα φαίνονται δύο παρόμοια σχήματα με διαφορετικές παραμέτρους υποδιαιρέσεων:



[Two cuboids with different subdivision parameters (as seen in the triangle edit mode)]

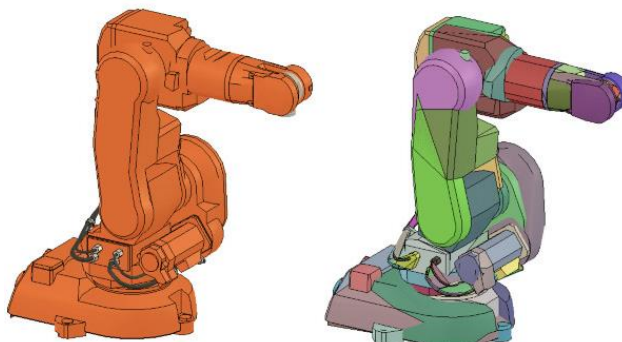
Τα σχήματα με μεγαλύτερο αριθμό υποδιαιρέσεων μπορούν να αντανακλούν το φως με πιο διαφοροποιημένο τρόπο, ανάλογα με τον επιλεγμένο renderer.

Τα χρώματα και άλλα οπτικά χαρακτηριστικά πρέπει να ρυθμιστούν στις ιδιότητες του σχήματος. Το πλέγμα του σχήματος μπορεί επίσης να ρυθμιστεί και να επεξεργαστεί στις λειτουργίες επεξεργασίας σχήματος ή στο διάλογο γεωμετρίας.

1.4.18 Αποσύνθεση κυρτών μερών/Convex decomposition

Το CorreliaSim επιτρέπει τον υπολογισμό και την προσθήκη της κυρτής αποσύνθεσης των καθορισμένων σχημάτων. Τα κυρτά σχήματα αποδίδουν ταχύτερα και είναι πιο σταθερά από τα τυχαία σχήματα, όσον αφορά τον υπολογισμό της δυναμικής απόκρισης σύγκρουσης. Ωστόσο, δεν είναι τόσο γρήγορα ούτε τόσο σταθερά όσο τα πρωτόγονα σχήματα.

Μπορούμε να δημιουργήσουμε το ισοδύναμο κυρτής αποσύνθεσης ενός σχήματος με [Modules → Geometry / Mesh → Convex decomposition]:



[Non-convex model (left) and corresponding convex-decomposed model (right)]

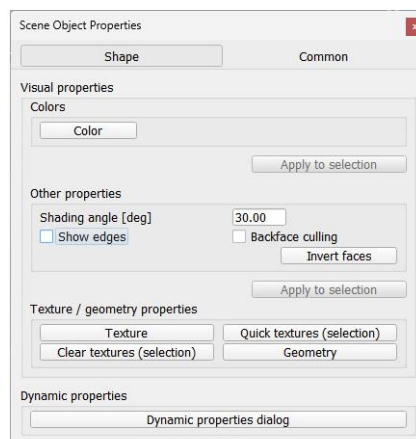
1.4.19 Ιδιότητες σχήματος/Shape properties

Οι ιδιότητες του σχήματος αποτελούν μέρος του παραθύρου διαλόγου ιδιοτήτων αντικειμένου σκηνής, που βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το διάλογο με διπλό κλικ στο εικονίδιο ενός αντικειμένου στην ιεραρχία της σκηνής ή με κλικ στο κουμπί της γραμμής εργαλείων:



[Scene object properties toolbar button]

Στον διάλογο ιδιοτήτων αντικειμένου σκηνής, κάνουμε κλικ στο κουμπί **Shape** για να εμφανιστεί ο διάλογος σχήματος (το κουμπί **Shape** εμφανίζεται μόνο εάν η τελευταία επιλογή είναι ένα σχήμα). Ο διάλογος εμφανίζει τις ρυθμίσεις και τις παραμέτρους του τελευταίου επιλεγμένου σχήματος. Εάν επιλεγούν περισσότερα από ένα σχήματα, τότε ορισμένες παράμετροι μπορούν να αντιγραφούν από το τελευταίο επιλεγμένο σχήμα στα άλλα επιλεγμένα σχήματα (**Apply to selection**).



[Shape dialog]

- **Χρώμα/Color:** Επιτρέπει την επεξεργασία των χρωμάτων του σχήματος.
- **Γωνία σκίασης/Shading angle:** Γωνία σκίασης είναι η γωνία από την οποία διακρίνονται τα μεμονωμένα πρόσωπα. Αυτό επηρεάζει μόνο την οπτική εμφάνιση του σχήματος. Μια μικρή γωνία κάνει ένα σχήμα να φαίνεται αιχμηρό, με πολλές ακμές, ενώ μια μεγάλη γωνία το καθιστά ομαλό και με λιγότερες ακμές.
- **Εμφάνιση γωνιών/Show edges:** Εμφανίζει τις ακμές σε μαύρο χρώμα.
- **Αποκοπή οπίσθιας όψης/Backface culling:** Κάθε τρίγωνο που αποτελεί ένα σχήμα έχει μια εσωτερική και μια εξωτερική όψη. Όταν είναι ενεργοποιημένο το Backface culling, οι

Προσομοίωση Ρομπότ μέσω Υπολογιστή

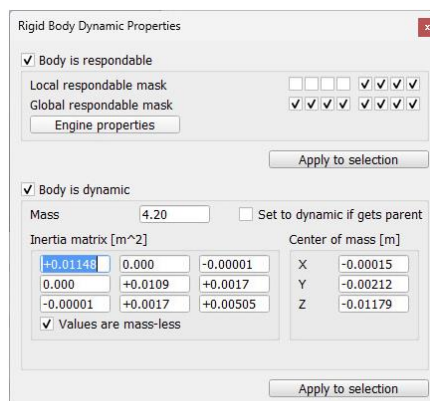
εσωτερικές όψεις δε θα εμφανίζονται. Αυτό είναι χρήσιμο για κλειστά σχήματα και για σχήματα που είναι διαφανή.

- **Αντιστροφή προσώπων/Invert faces:** Αυτό αναστρέφει όλα τα τρίγωνα. Οι εσωτερικές όψεις γίνονται εξωτερικές και αντίστροφα. Τα κυρτά σχήματα θα γίνουν μη κυρτά, εκτός από τα πρωτόγονα σχήματα.
- **Υφή/Texture:** Ανοίγει το παράθυρο διαλόγου υφής για το επιλεγμένο σχήμα. Όταν ένα σχήμα είναι συνδεδεμένο με μια υφή, θα εμφανίζεται με υφή.
- **Γρήγορη υφή(επιλογή)/Quick textures (selection):** Εφαρμόζει μια τετραγωνικά απεικονισμένη υφή σε όλα τα επιλεγμένα σχήματα. Είναι ιδιαίτερα χρήσιμο με αδιάληπτες υφές που χρησιμοποιούνται ως "βρωμιές", για να κάνουν τα αντικείμενα να φαίνονται πιο ρεαλιστικά.
- **Εκκαθάριση υφών(επιλογή)/Clear textures (selection):** Αφαιρεί την υφή από όλα τα επιλεγμένα σχήματα.
- **Γεωμετρία/Geometry:** Ανοίγει το παράθυρο διαλόγου γεωμετρίας του σχήματος για το επιλεγμένο σχήμα. Επιτρέπει τη ρύθμιση διάφορων παραμέτρων του πλέγματος.
- **Παράθυρο διαλόγου δυναμικών ιδιοτήτων/Dynamic properties dialog:** Εναλλάσσει το παράθυρο διαλόγου ιδιοτήτων δυναμικών του σχήματος. Το παράθυρο διαλόγου δυναμικών ιδιοτήτων επιτρέπει τη ρύθμιση των δυναμικών ιδιοτήτων ενός σχήματος.

Ορισμένες από τις παραπάνω παραμέτρους είναι διαθέσιμες μόνο για απλά σχήματα. Όταν επιλέγεται ένα σύνθετο σχήμα, τότε μπορούμε να επεξεργαστούμε τα οπτικά χαρακτηριστικά του μεταβαίνοντας στη λειτουργία επεξεργασίας σχήματος για σύνθετα σχήματα. Φυσικά, μπορούμε επίσης να αποομαδοποιήσουμε το σχήμα για να επεξεργαστούμε ανεξάρτητα τα στοιχεία του.

1.4.20 Ιδιότητες δυναμικής του σχήματος/Shape dynamics properties

Το παράθυρο διαλόγου δυναμικής του σχήματος αποτελεί μέρος των ιδιοτήτων του σχήματος. Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους δυναμικής του τελευταίου επιλεγμένου σχήματος. Εάν δεν έχει επιλεγεί κανένα αντικείμενο, το παράθυρο διαλόγου είναι ανενεργό. Εάν επιλεγούν περισσότερα από ένα σχήματα, τότε ορισμένες παράμετροι μπορούν να αντιγραφούν από το τελευταίο επιλεγμένο σχήμα στα άλλα επιλεγμένα σχήματα (**Apply to selection**):

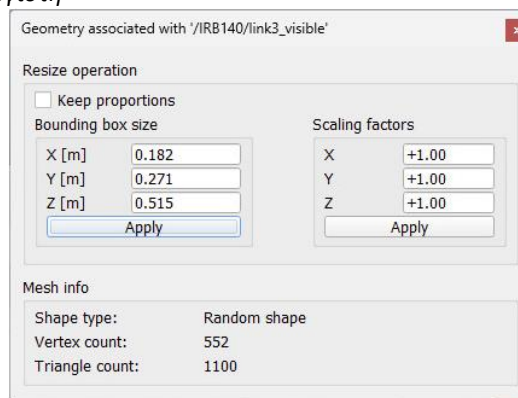


[Shape dynamics dialog]

- **Το σώμα είναι αποκρίσιμο/Body is respondable:** Αν είναι ενεργοποιημένο, τότε το σχήμα θα αντιδράει στη σύγκρουση με άλλα αποκρίσιμα σχήματα, μόνο αν οι αντίστοιχες αποκρίσιμες μάσκες αλληλεπικαλύπτονται.
- **Αποκρίσιμη μάσκα/Respondable mask:** Υποδεικνύει πότε δημιουργείται απόκριση σύγκρουσης. Η μάσκα αποτελείται από δύο 8-bit τιμές, local και global. Αν δύο σχήματα που συγκρούονται μοιράζονται οποιονδήποτε από τους γονείς τους (άμεσους ή έμμεσους), τότε χρησιμοποιούνται οι τοπικές μάσκες, αλλιώς χρησιμοποιούνται οι παγκόσμιες μάσκες. Αν οι μάσκες δύο σχημάτων, συνδυασμένες με AND (local ή global), είναι διαφορετικές από το μηδέν, τότε θα δημιουργηθεί απόκριση σύγκρουσης.
- **Ιδιότητες κινητήρα/Engine properties:** Κάθε φυσικός κινητήρας έχει συγκεκριμένες παραμέτρους σχετικές με τα σχήματα. Πρέπει να βεβαιωθούμε ότι διαβάζουμε τις αντίστοιχες τεκμηριώσεις κινητήρων για να επιτύχουμε τις καλύτερες ρυθμίσεις παραμέτρων.
- **Το σώμα είναι δυναμικό/Body is dynamic:** Όταν είναι ενεργοποιημένο, τότε η θέση του σχήματος θα επηρεαστεί σε μια δυναμική προσομοίωση.
- **Μάζα/Mass:** Η μάζα του σχήματος. Τα επιλεγμένα σχήματα μπορούν εύκολα να υπολογιστούν ([Edit → Shape mass and inertia → compute from uniform density...]) ή να κλιμακωθούν ([Edit → Shape mass and inertia → scale mass...]).
- **Ορισμός σε δυναμικό αν αποκτά γονέα/Set to dynamic if gets parent:** Όταν είναι ενεργοποιημένο και το σχήμα προσαρτάται σε άλλο αντικείμενο, τότε το σχήμα θα γίνεται αυτόματα δυναμικό. Αυτό είναι χρήσιμο για βάσεις μοντέλων που πρέπει να είναι στατικές όταν είναι μόνες τους, αλλά δυναμικές όταν συνδυάζονται με άλλο μοντέλο/αντικείμενο (π.χ. ένας ρομποτικός βραχίονας που λειτουργεί μόνος του συνήθως έχει τη βάση στατική, αλλά όταν προσαρτάται σε ένα όχημα, τότε η βάση πρέπει να γίνεται δυναμική).
- **Πίνακας αδράνειας/Inertia matrix:** Ο πίνακας αδράνειας. Οι αναπαριστώμενες τιμές είναι χωρίς μάζα από προεπιλογή (δηλ. διαιρούνται με τη μάζα του σχήματος). Ο πίνακας εκφράζεται σε σχέση με το πλαίσιο αναφοράς του σχήματος. Τα επιλεγμένα σχήματα μπορούν εύκολα να υπολογιστούν ([Edit → Shape mass and inertia → compute from uniform density...]) ή να κλιμακωθούν ([Edit → Shape mass and inertia → scale inertia...]).
- **Κέντρο μάζας/Center mass:** Η θέση του κέντρου μάζας, σε σχέση με το πλαίσιο αναφοράς του σχήματος.

1.4.21 Παράθυρο διαλόγου γεωμετρίας σχήματος/Shape geometry dialog

Το παράθυρο διαλόγου γεωμετρίας σχήματος είναι μέρος των ιδιοτήτων του σχήματος. Το παράθυρο διαλόγου επιτρέπει την προβολή μιας γεωμετρίας που σχετίζεται με ένα σχήμα και την τροποποίησή της σε ένα βαθμό. Εμφανίζει τον αριθμό των κορυφών και τριγώνων, καθώς και το μέγεθος του περιβλήματος του σχήματος. Ο αριθμός των κορυφών και τριγώνων ενός πλέγματος σχετίζεται άμεσα με τους χρόνους απεικόνισης και υπολογισμού (π.χ. κατά την ανίχνευση σύγκρουσης ή τον υπολογισμό αποστάσεων) και όσο περισσότερες κορυφές/τρίγωνα έχει ένα πλέγμα, τόσο πιο αργή θα είναι η προσομοίωση ή η εμφάνιση της σκηνής. Ενώ οι κοινές ιδιότητες αντικειμένου επιτρέπουν την κλιμάκωση αντικειμένων (συμπεριλαμβανομένων των σχημάτων), διατηρούν τις αναλογίες σταθερές σε όλες τις διαστάσεις (η κλιμάκωση κατά μήκος των αξόνων x-, y- ή z- είναι ίδια). Αυτός ο περιορισμός δεν ισχύει στο παράθυρο διαλόγου γεωμετρίας και μπορούμε ακόμα να αναστρέψουμε ένα πλέγμα κατά μήκος ενός από τους άξονές του, καθορίζοντας ένα αρνητικό παράγοντα κλιμάκωσης. Πρέπει θυμόμαστε ωστόσο ότι ορισμένα πρωτογενή σχήματα έχουν περιορισμούς σχετικά με τη μη ομοιόμορφη κλιμάκωση. Επιπλέον, τα πρωτογενή σχήματα δεν μπορούν να αναστραφούν.

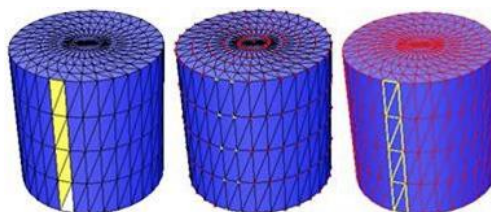


[Shape geometry dialog]

1.4.22 Λειτουργίες επεξεργασίας σχήματος/Shape edit modes

Ο καλύτερος τρόπος για να δημιουργηθούν συγκεκριμένα σχήματα στο CoppeliaSim είναι να τα σχεδιάσουμε σε μια εφαρμογή CAD (π.χ. AutoCAD, 3D Studio Max, κ.λπ.) και στη συνέχεια να τα εισάγουμε. Όταν δεν υπάρχει διαθέσιμη εξωτερική εφαρμογή CAD ή όταν απαιτούνται μόνο απλά σχήματα, ο χρήστης μπορεί να δημιουργήσει πρωτογενή σχήματα και στη συνέχεια να προσαρμόσει το δημιουργημένο σχήμα σε μία από τις τρεις υποστηριζόμενες λειτουργίες επεξεργασίας σχήματος στο CoppeliaSim:

- **Λειτουργία επεξεργασίας τριγώνων/Triangle edit mode:** Σε αυτή τη λειτουργία, τα μεμονωμένα τρίγωνα που αποτελούν το σχήμα είναι ορατά και μπορούν να χειριστούν ή να εξαχθούν.
- **Λειτουργία επεξεργασίας κορυφών/Vertex edit mode:** Σε αυτή τη λειτουργία, οι μεμονωμένες κορυφές που αποτελούν το σχήμα είναι ορατές και μπορούν να χειριστούν ή να εξαχθούν.
- **Λειτουργία επεξεργασίας ακμών/Edge edit mode:** Σε αυτή τη λειτουργία, οι μεμονωμένες ακμές που αποτελούν το σχήμα είναι ορατές και μπορούν να χειριστούν ή να εξαχθούν.



[Triangle, vertex and edge edit modes]

Τα πρωτογενή σχήματα μετατρέπονται σε κανονικά σχήματα πριν την είσοδο σε μία από τις παραπάνω λειτουργίες επεξεργασίας.

Ένα σύνθετο σχήμα δεν μπορεί να επεξεργαστεί άμεσα. Πρέπει πρώτα να αποσυναρμολογηθεί. Εναλλακτικά, μπορούμε να επεξεργαστούμε τις οπτικές παραμέτρους των στοιχείων του μέσω της λειτουργίας επεξεργασίας για σύνθετα σχήματα. Οι διάφορες λειτουργίες επεξεργασίας είναι προσβάσιμες με ένα κλικ στο κατάλληλο κουμπί της γραμμής εργαλείων:



[Shape edit mode toolbar button]

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Πρέπει να βεβαιωθούμε ότι έχουμε επιλέξει ένα αντικείμενο σχήματος πριν εισέλθουμε στη λειτουργία επεξεργασίας σχήματος. Κατά την επεξεργασία σχήματος, τα αντικείμενα δεν μπορούν να επιλεγούν και η προσομοίωση δεν μπορεί να ξεκινήσει. Μόλις ολοκληρωθεί η επεξεργασία, κάνουμε κλικ στο κουμπί της γραμμής εργαλείων της λειτουργίας επεξεργασίας σχήματος για να βγούμε από τη λειτουργία επεξεργασίας. Όταν εφαρμόζουμε αλλαγές σε ένα σχήμα, το CoppeliaSim θα διασφαλίσει ότι το τροποποιημένο σχήμα είναι συνεπές, θα αφαιρέσει αχρησιμοποίητες κορυφές και θα συγχωνεύσει κορυφές που είναι κοντά η μία στην άλλη κ.λπ. Η ακριβής συμπεριφορά μπορεί να ρυθμιστεί στις ρυθμίσεις επαλήθευσης κορυφών/τριγώνων στο παράθυρο διαλόγου ρυθμίσεων χρήστη.

Μπορούμε να μεταβούμε από τη μία λειτουργία επεξεργασίας στην άλλη και αυτό μας επιτρέπει να επιτύχουμε ειδικές λειτουργίες. Για παράδειγμα, αν θέλουμε να επιλέξουμε και να αφαιρέσουμε όλα τα τρίγωνα της άνω επιφάνειας ενός κυλίνδρου, αντί να τα επιλέξουμε ξεχωριστά στη λειτουργία επεξεργασίας τριγώνων, πραγματοποιούμε επιλογή-μετατόπιση όλων των άνω κορυφών στη λειτουργία επεξεργασίας κορυφών, στη συνέχεια μεταβαίνουμε στη λειτουργία επεξεργασίας τριγώνων και στη συνέχεια πατάμε το πλήκτρο διαγραφής.

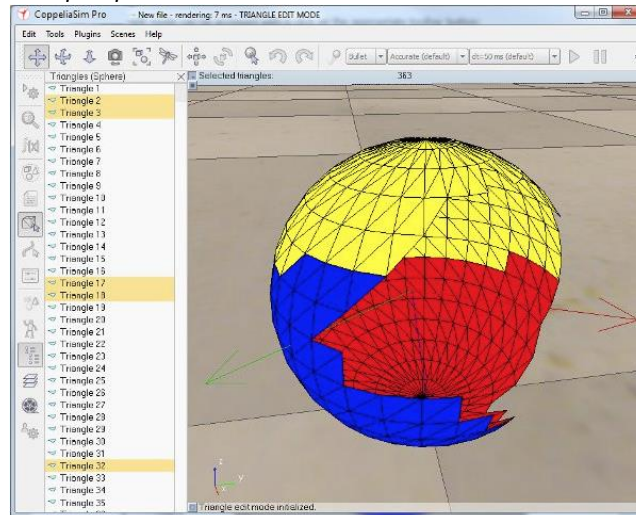
1.4.23 Λειτουργία επεξεργασίας τριγώνων/Triangle edit mode

Η λειτουργία επεξεργασίας τριγώνων μπορεί να προσπελαστεί με ένα κλικ στο κατάλληλο κουμπί της γραμμής εργαλείων:



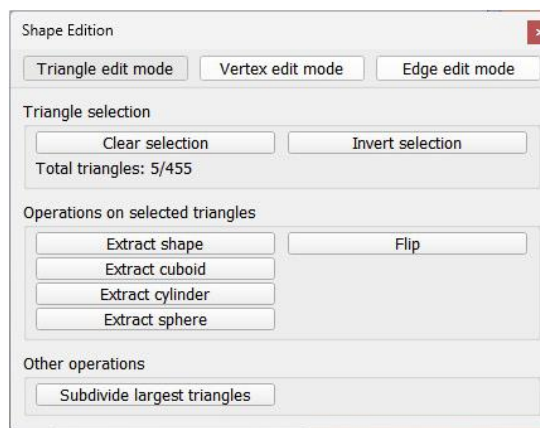
[Shape edit mode toolbar button]

Το κουμπί της παραπάνω γραμμής εργαλείων είναι ενεργό μόνο εάν έχει επιλεγεί ένα σχήμα. Εάν το τελευταίο επιλεγμένο σχήμα δεν είναι απλό, αλλά σύνθετο, θα ενεργοποιηθεί αντί για αυτό η λειτουργία επεξεργασίας σύνθετων σχημάτων. Στη λειτουργία επεξεργασίας τριγώνων όλα τα τρίγωνα που αποτελούν το σχήμα εμφανίζονται μεμονωμένα. Ένα τρίγωνο έχει δύο όψεις, την μπροστινή όψη και την πίσω όψη. Η μπροστινή όψη εμφανίζεται σε μπλε χρώμα, ενώ η πίσω όψη εμφανίζεται σε κόκκινο. Κατά την επιλογή τριγώνων (χρησιμοποιώντας την ίδια διαδικασία όπως για την επιλογή αντικειμένων), θα εμφανίζονται σε κίτρινο χρώμα και το τελευταίο επιλεγμένο τρίγωνο σε λευκό. Υποστηρίζονται οι συντομεύσεις αντιγραφής/αποκοπής/επικόλλησης/διαγραφής (ctrl-c, ctrl-x, ctrl-v, delete). Η επιλογή μπορεί να εκκαθαριστεί με το πλήκτρο esc, με το κουμπί αποεπιλογής της γραμμής εργαλείων ή με το ctrl-κλικ σε μια κενή περιοχή της σκηνής. Μια επιλογή με shift θα επιλέξει όλα τα τρίγωνα κάτω από την περιοχή επιλογής, συμπεριλαμβανομένων και των κρυφών τριγώνων (εάν επιθυμείτε να επιλέξετε μόνο ορατά τρίγωνα μέσω μιας επιλογής με shift, κρατήστε πατημένο το πλήκτρο ctrl επιπλέον του shift). Στη λειτουργία επεξεργασίας τριγώνων, το τμήμα του παραθύρου που κανονικά εμφανίζει την ιεραρχία της σκηνής χρησιμοποιείται για να εμφανίσει τα τρίγωνα του σχήματος που επεξεργάζεστε ως λίστα. Τα αντικείμενα στη λίστα μπορούν να επιλεγούν με το ποντίκι όπως για τα αντικείμενα στο παράθυρο ιεραρχίας.



[Triangle edit mode]

Μετά την είσοδο στη λειτουργία επεξεργασίας τριγώνου, τα στοιχεία του μενού επεξεργασίας ([Menu bar → Edit]) θα γίνουν συγκεκριμένα για τη λειτουργία επεξεργασίας τριγώνου και θα εμφανιστεί το ακόλουθο παράθυρο διαλόγου:



[Triangle edit mode dialog]

- **Λειτουργία επεξεργασίας τριγώνου / κορυφής / ακμής/Triangle / Vertex / Edge edit mode:** Η τρέχουσα ενεργοποιημένη λειτουργία επεξεργασίας. Ο χρήστης μπορεί να μεταβεί από μία λειτουργία επεξεργασίας στις άλλες μέσω αυτών των κουμπιών (τα επιλεγμένα στοιχεία θα διατηρηθούν).
- **Εκκαθάριση επιλογής/Clear selection:** Καθαρίζει την επιλογή των τριγώνων.
- **Αντιστροφή επιλογής/Invert selection:** Αντιστρέφει την κατάσταση επιλογής των τριγώνων.
- **Εξαγωγή σχήματος/Extract shape:** Θα δημιουργήσει ένα σχήμα βασισμένο στα επί του παρόντος επιλεγμένα τρίγωνα. Το σχήμα θα προστεθεί στη σκηνή αλλά ενδέχεται να μην είναι ορατό (το σχήμα στη λειτουργία επεξεργασίας και το νεοδημιουργηθέν σχήμα συμπίπτουν στη θέση, ή η πρώτη στρώση ορατότητας είναι απενεργοποιημένη). Η επιλογή τριγώνων θα παραμείνει.
- **Εξαγωγή κυβοειδούς/Extract cuboid:** Θα δημιουργήσει ένα ορθογώνιο πρωτογενές σχήμα βασισμένο στα επί του παρόντος επιλεγμένα τρίγωνα (θα ευθυγραμμίσει το νέο κυβοειδές με

Προσομοίωση Ρομπότ μέσω Υπολογιστή

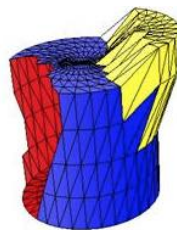
το ελάχιστο κουτί που περικλείει τα επιλεγμένα τρίγωνα). Η επιλογή τριγώνων θα παραμείνει. Αυτή η λειτουργία είναι χρήσιμη όταν ένα πολύπλοκο σχήμα δικτύου χρειάζεται να μοντελοποιηθεί με απλούστερο τρόπο για να επιτρέψει αποδοτική δυναμική προσομοίωση, για παράδειγμα.

- **Εξαγωγή κυλίνδρου/Extract cylinder:** Θα δημιουργήσει κυλινδρικό πρωτογενές σχήμα βασισμένο στα επί του παρόντος επιλεγμένα τρίγωνα (θα ευθυγραμμίσει το νέο κύλινδρο με το ελάχιστο κουτί που περικλείει τα επιλεγμένα τρίγωνα). Η επιλογή τριγώνων θα παραμείνει. Αυτή η λειτουργία είναι χρήσιμη όταν ένα πολύπλοκο σχήμα δικτύου χρειάζεται να μοντελοποιηθεί με απλούστερο τρόπο για να επιτρέψει αποδοτική δυναμική προσομοίωση, για παράδειγμα.
- **Εξαγωγή σφαίρας/Extract sphere:** Θα δημιουργήσει σφαιρικό πρωτογενές σχήμα βασισμένο στα επί του παρόντος επιλεγμένα τρίγωνα (θα ευθυγραμμίσει τη νέα σφαίρα με το ελάχιστο κουτί που περικλείει τα επιλεγμένα τρίγωνα). Η επιλογή τριγώνων θα παραμείνει. Αυτή η λειτουργία είναι χρήσιμη όταν ένα πολύπλοκο σχήμα δικτύου χρειάζεται να μοντελοποιηθεί με απλούστερο τρόπο για να επιτρέψει αποδοτική δυναμική προσομοίωση, για παράδειγμα.
- **Αναστροφή/Flip:** Αναστρέφει την πλευρά των επιλεγμένων τριγώνων (από κόκκινο σε μπλε και από μπλε σε κόκκινο).
- **Διαίρεση των μεγαλύτερων τριγώνων/Subdivide largest triangles:** Αυτό θα διαιρέσει τα μεγάλα τρίγωνα. Εκτός αν θέλουμε να μειώσουμε το μέγεθος των τριγώνων για να αποκτήσουμε ένα πιο διαφοροποιημένο φωτισμό των μεγάλων επιφανειών, δεν συνιστάται να μειώσουμε το μέγεθος των τριγώνων (εκτός αν γνωρίζουμε ακριβώς τι κάνουμε). Αντί για αυτό, μπορούμε να προσαρμόσουμε το μέγεθος των τριγώνων για τις δομές υπολογισμούς παγκοσμίως (στο παράθυρο περιβάλλοντος).

Τα τρίγωνα μπορούν να μεταφραστούν απευθείας με το ποντίκι, χρησιμοποιώντας το κουμπί μετάφρασης αντικειμένου/στοιχείου της γραμμής εργαλείων για να μεταφράσουμε το επιλεγμένο τρίγωνο(α) σε επίκεντρο επίπεδο προς την κατεύθυνση της θέασης:



[Object/item translation toolbar button]



[Shape in triangle edit mode]

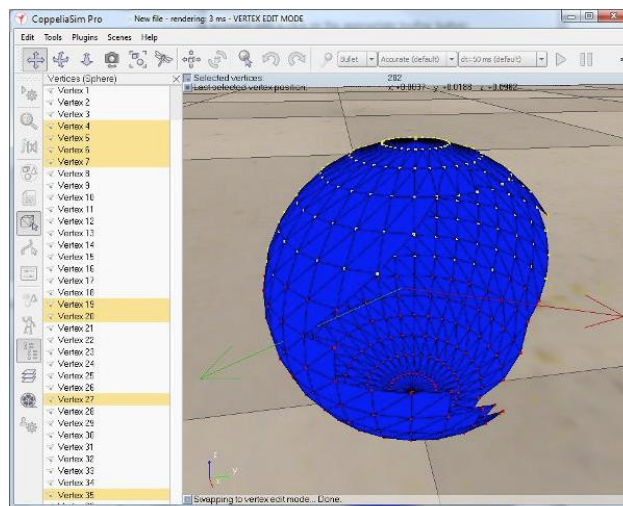
Λειτουργία επεξεργασίας κορυφής/Vertex edit mode

Η λειτουργία επεξεργασίας κορυφής μπορεί να προσπελαστεί με ένα κλικ στο κατάλληλο κουμπί της γραμμής εργαλείων:



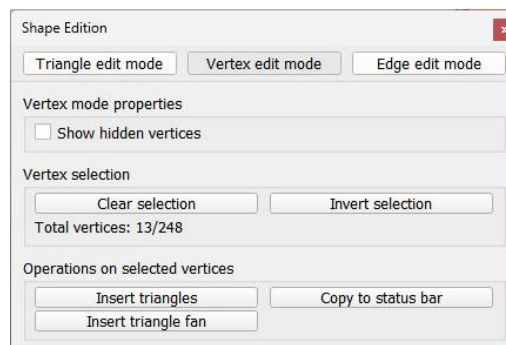
[Shape edit mode toolbar button]

Το παραπάνω κουμπί γραμμής εργαλείων είναι ενεργό μόνο εάν έχει επιλεγεί ένα σχήμα. Εάν το τελευταίο επιλεγμένο σχήμα δεν είναι απλό σχήμα, αλλά σύνθετο σχήμα, θα ενεργοποιηθεί η λειτουργία επεξεργασίας για σύνθετα σχήματα. Στη λειτουργία επεξεργασίας κορυφής όλες οι κορυφές που αποτελούν το σχήμα εμφανίζονται ξεχωριστά σε κόκκινο. Κατά την επιλογή κορυφών (χρησιμοποιώντας την ίδια διαδικασία με την επιλογή αντικειμένων), θα εμφανίζονται σε κίτρινο και η τελευταία επιλεγμένη κορυφή σε λευκό. Υποστηρίζεται η συντόμευση αντιγραφής/αποκοπής/επικόλλησης/διαγραφής (ctrl-c, ctrl-x, ctrl-v, delete). Η επιλογή μπορεί να καθариστεί με το πλήκτρο esc, με το κουμπί αποεπιλογής της γραμμής εργαλείων, ή με ctrl-κλικ σε μια κενή περιοχή της σκηνής. Μια επιλογή με shift θα επιλέξει όλες τις κορυφές κάτω από την περιοχή επιλογής, συμπεριλαμβανομένων των κρυφών κορυφών (εάν θέλουμε να επιλέξουμε μόνο ορατές κορυφές μέσω επιλογής με shift, κρατάμε πατημένο το πλήκτρο ctrl επιπλέον του shift). Στη λειτουργία επεξεργασίας κορυφής, το μέρος του παραθύρου που κανονικά εμφανίζει την ιεραρχία της σκηνής χρησιμοποιείται για την εμφάνιση των κορυφών του σχήματος που επεξεργάζεται ως λίστα. Τα στοιχεία στη λίστα μπορούν να επιλεγούν με το ποντίκι όπως και τα αντικείμενα στο παράθυρο ιεραρχίας.



[Vertex edit mode]

Μετά την είσοδο στη λειτουργία επεξεργασίας κορυφής, τα στοιχεία του μενού επεξεργασίας ([Menu bar → Edit]) θα γίνουν συγκεκριμένα για τη λειτουργία επεξεργασίας κορυφής και θα εμφανιστεί το ακόλουθο παράθυρο διαλόγου:



[Vertex edit mode dialog]

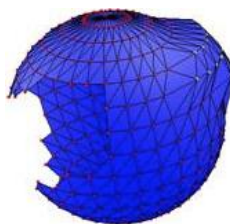
Προσομοίωση Ρομπότ μέσω Υπολογιστή

- **Λειτουργία επεξεργασίας τριγώνου / κορυφής / ακμής/ Triangle / Vertex / Edge Edit Mode:** Η τρέχουσα ενεργοποιημένη λειτουργία επεξεργασίας. Ο χρήστης μπορεί να μεταβεί από μία λειτουργία επεξεργασίας στις άλλες μέσω αυτών των κουμπιών (τα επιλεγμένα στοιχεία θα διατηρηθούν).
- **Εμφάνιση κρυφών κορυφών/Show Hidden Vertices:** όταν επιλεγεί, θα εμφανίζει επίσης κρυφές κορυφές.
- **Εκκαθάριση επιλογής/Clear selection:** καθαρίζει την επιλογή των κορυφών.
- **Αντιστροφή επιλογής/Invert selection:** αντιστρέφει την κατάσταση επιλογής των κορυφών.
- **Εισαγωγή τριγώνων/Insert Triangles:** θα εισάγει τρίγωνα μεταξύ των επιλεγμένων κορυφών. Η σειρά με την οποία επιλέγονται οι κορυφές είναι σημαντική.
- **Εισαγωγή ανεμιστήρα τριγώνων/Insert Triangle Fan:** θα εισάγει έναν ανεμιστήρα τριγώνων μεταξύ των επιλεγμένων κορυφών. Η σειρά με την οποία επιλέγονται οι κορυφές είναι σημαντική. Η τελευταία επιλεγμένη κορυφή θα είναι η κοινή κορυφή μεταξύ των νέων τριγώνων που εισάγονται.
- **Αντιγραφή στη γραμμή κατάστασης/Copy to status bar:** εκτυπώνει τις επιλεγμένες κορυφές στη γραμμή κατάστασης.

Οι κορυφές μπορούν να τοποθετηθούν με ακρίβεια μέσω του παραθύρου διαλόγου συντεταγμένων και μετασχηματισμών. Μπορούν επίσης να μεταφραστούν απευθείας με το ποντίκι, χρησιμοποιώντας το κουμπί μετάφρασης αντικειμένου/στοιχείου της γραμμής εργαλείων για να μεταφράσουμε την επιλεγμένη κορυφή/τις επιλεγμένες κορυφές σε ένα επίπεδο κάθετο προς την κατεύθυνση θέασης:



[Object/item translation toolbar button]



[Shape in vertex edit mode]

1.4.24 Λειτουργία επεξεργασίας ακμής/Edge edit mode

Η λειτουργία επεξεργασίας ακμής μπορεί να προσπελαστεί με ένα κλικ στο κατάλληλο κουμπί της γραμμής εργαλείων:

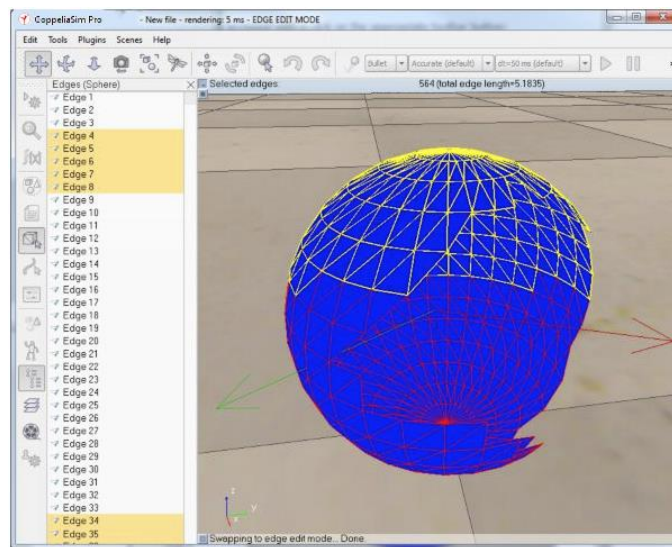


[Shape edit mode toolbar button]

Το παραπάνω κουμπί γραμμής εργαλείων είναι ενεργό μόνο εάν έχει επιλεγεί ένα σχήμα. Εάν το τελευταίο επιλεγμένο σχήμα δεν είναι απλό σχήμα, αλλά σύνθετο σχήμα, θα ενεργοποιηθεί η λειτουργία επεξεργασίας για σύνθετα σχήματα. Στη λειτουργία επεξεργασίας ακμής όλες οι ακμές που αποτελούν το σχήμα εμφανίζονται ξεχωριστά σε κόκκινο. Κατά την επιλογή ακμών

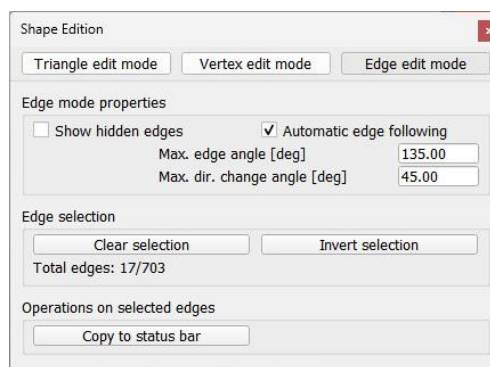
Προσομοίωση Ρομπότ μέσω Υπολογιστή

(χρησιμοποιώντας την ίδια διαδικασία με την επιλογή αντικειμένων), θα εμφανίζονται σε κίτρινο και η τελευταία επιλεγμένη ακμή σε λευκό. Οι ακμές μπορούν να διαγραφούν με το πλήκτρο delete, ωστόσο δεν μπορούν να αντιγραφούν, αποκοπουν ή επικολληθούν. Η επιλογή μπορεί να καθαριστεί με το πλήκτρο esc, με το κουμπί αποεπιλογής της γραμμής εργαλείων, ή με ctrl-κλικ σε μια κενή περιοχή της σκηνής. Μια επιλογή με shift θα επιλέξει όλες τις ακμές κάτω από την περιοχή επιλογής, συμπεριλαμβανομένων των κρυφών ακμών (εάν θέλουμε να επιλέξουμε μόνο ορατές ακμές μέσω επιλογής με shift, κρατάμε πατημένο το πλήκτρο ctrl και shift). Στη λειτουργία επεξεργασίας ακμής, το μέρος του παραθύρου που κανονικά εμφανίζει την ιεραρχία της σκηνής χρησιμοποιείται για την εμφάνιση των ακμών του σχήματος που επεξεργάζεται ως λίστα. Τα στοιχεία στη λίστα μπορούν να επιλεγούν με το ποντίκι όπως και τα αντικείμενα στο παράθυρο ιεραρχίας.



[Edge edit mode]

Μετά την είσοδο στη λειτουργία επεξεργασίας ακμής, τα στοιχεία του μενού επεξεργασίας ([Menu bar → Edit]) θα γίνουν συγκεκριμένα για τη λειτουργία επεξεργασίας ακμής και θα εμφανιστεί το ακόλουθο παράθυρο διαλόγου:

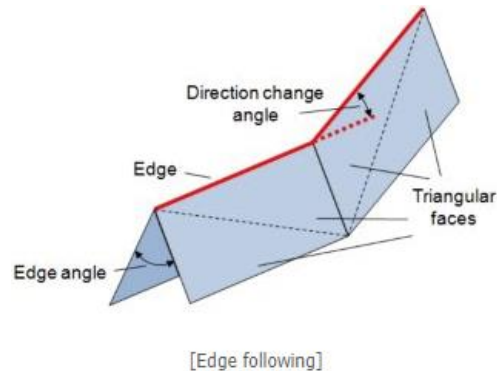


[Edge edit mode dialog]

- **Λειτουργία επεξεργασίας τριγώνου / κορυφής / ακμής/ Triangle / Vertex / Edge Edit Mode:** Η τρέχουσα ενεργοποιημένη λειτουργία επεξεργασίας. Ο χρήστης μπορεί να μεταβεί από μία λειτουργία επεξεργασίας στις άλλες μέσω αυτών των κουμπιών (τα επιλεγμένα στοιχεία θα διατηρηθούν).
- **Εμφάνιση κρυφών ακμών/Show Hidden Edges:** όταν επιλεγεί, θα εμφανίζει επίσης κρυφές ακμές.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

- **Αυτόματη παρακολούθηση ακμής/Automatic Edge Following:** όταν επιλεγεί και επιλεγεί μια ακμή, αυτή η ακμή θα παρακολουθηθεί μέχρι να δημιουργηθεί ένας βρόχος, μέχρι η ακμή να αλλάξει κατεύθυνση απότομα (**Max. dir. change angle**) ή μέχρι να εξαφανιστεί η ακμή (**Max. edge angle**). Αυτή είναι μια χρήσιμη λειτουργία όταν θέλουμε να εξάγουμε όχι μόνο μια τοπική ακμή (μια πλευρά ενός τριγώνου) αλλά μια συνολική (μια ακμή σχήματος). Μια ακμή παρακολουθείται μόνο προς μία κατεύθυνση.



- **Εκκαθάριση επιλογής/Clear selection:** καθαρίζει την επιλογή των ακμών.
- **Αντιστροφή επιλογής/Invert selection:** αντιστρέφει την κατάσταση επιλογής των ακμών.
- **Αντιγραφή στη γραμμή κατάστασης/Copy to status bar:** εκτυπώνει τις επιλεγμένες ακμές στη γραμμή κατάστασης.

1.4.25 Λειτουργία επεξεργασίας για σύνθετα σχήματα/Edit mode for compound shapes

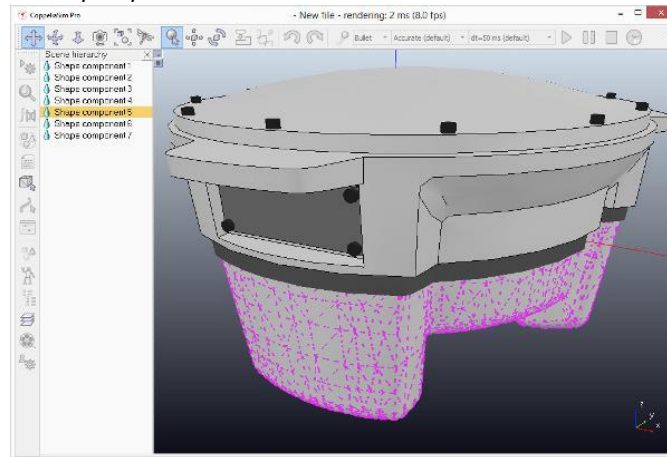
Η λειτουργία επεξεργασίας για σύνθετα σχήματα μπορεί να προσπελαστεί με ένα κλικ στο κατάλληλο κουμπί της γραμμής εργαλείων:



[Shape edit mode toolbar button]

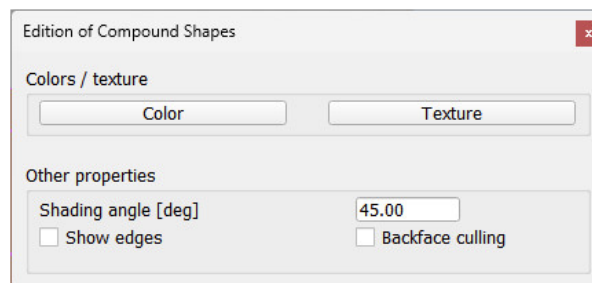
Το παραπάνω κουμπί γραμμής εργαλείων είναι ενεργό μόνο εάν έχει επιλεγεί ένα σχήμα. Εάν το τελευταίο επιλεγμένο σχήμα δεν είναι σύνθετο σχήμα, αλλά απλό σχήμα, τότε θα ενεργοποιηθεί η λειτουργία επεξεργασίας τριγώνου. Στη λειτουργία επεξεργασίας για σύνθετα σχήματα, μπορούμε να επεξεργαστούμε τις οπτικές παραμέτρους των συστατικών του σχήματος ξεχωριστά. Τα στοιχεία του σχήματος μπορούν να επιλεγούν από το παράθυρο ιεραρχίας.

Προσομοίωση Ρομπότ μέσω Υπολογιστή



[Edit mode for compound shapes]

Μετά την είσοδο στη λειτουργία επεξεργασίας για σύνθετα σχήματα, θα εμφανιστεί το ακόλουθο παράθυρο διαλόγου:

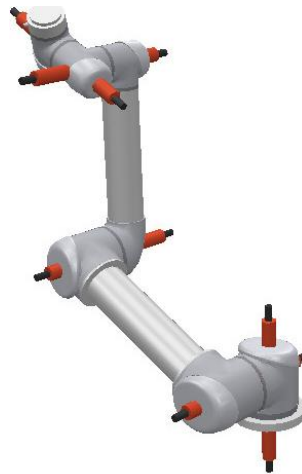


[Edit mode dialog for compound shapes]

- Ρύθμιση χρώματος: επιτρέπει την επεξεργασία των χρωμάτων των συστατικών του σχήματος.
- Ρύθμιση υφής: ανοίγει το παράθυρο υφής για το επιλεγμένο συστατικό του σχήματος.
- Γωνία σκίασης: η γωνία σκίασης είναι η γωνία από την οποία διακρίνονται οι μεμονωμένες επιφάνειες. Μια μικρή γωνία κάνει ένα σχήμα να φαίνεται αιχμηρό με πολλές ακμές, ενώ μια μεγάλη γωνία κάνει ένα σχήμα να φαίνεται λείο και με λιγότερες ακμές.
- Εμφάνιση ακμών: εμφανίζει τις ακμές σε μαύρο.
- Απόκρυψη οπίσθιων επιφανειών: κάθε τρίγωνο που αποτελεί ένα σχήμα έχει μια εσωτερική και μια εξωτερική επιφάνεια. Όταν είναι ενεργοποιημένη η απόκρυψη οπίσθιων επιφανειών, τότε οι εσωτερικές επιφάνειες δεν θα εμφανίζονται. Αυτή είναι μια χρήσιμη παράμετρος για κλειστά σχήματα και για σχήματα που είναι διαφανή.

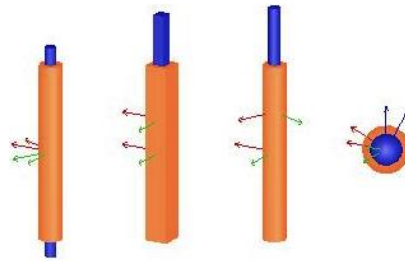
1.4.26 Αρθρώσεις/Joints

Μία άρθρωση είναι ένα αντικείμενο που επιτρέπει τη σχετική κίνηση μεταξύ του γονιού και του παιδιού/παιδιών του. Όταν δημιουργείται μια σχέση γονιού-παιδιού μεταξύ μίας άρθρωσης και ενός αντικείμενου, το αντικείμενο είναι συνδεδεμένο με το δεύτερο πλαίσιο αναφοράς της άρθρωσης, με αποτέλεσμα οι αλλαγές στη γραμμική/γωνιακή θέση της άρθρωσης να αντανακλώνται άμεσα στα παιδιά του. Οι αρθρώσεις μπορούν να λειτουργούν με διαφορετικούς τρόπους. Νέες αρθρώσεις μπορούν να προστεθούν σε μια σκηνή με [Add → Joints].



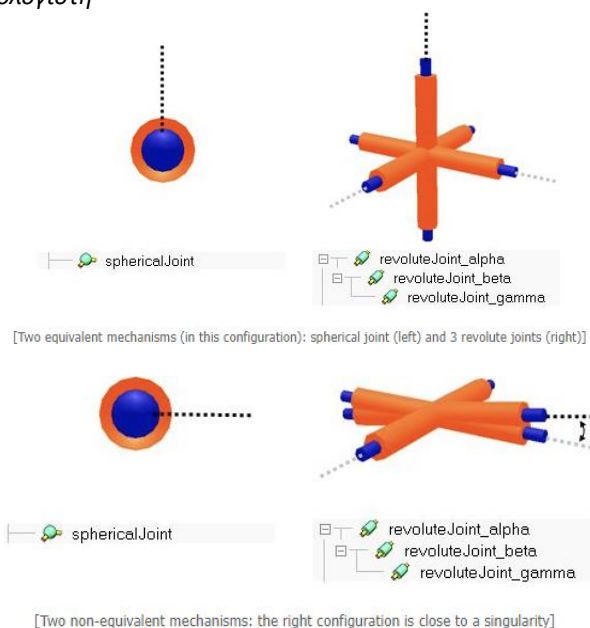
[Robotic manipulator with highlighted joints]

Τέσσερις τύποι αρθρώσεων υποστηρίζονται:



[Revolute joint, prismatic joint, screw and spherical joint]

- Περιστροφική άρθρωση/Revolute joint: μία περιστροφική άρθρωση έχει ένα βαθμό ελευθερίας και χρησιμοποιείται για να περιγράψει περιστροφική κίνηση μεταξύ αντικειμένων. Η περιστροφή συμβαίνει γύρω από τον άξονα z του πλαισίου αναφοράς του.
- Πρισματική άρθρωση/Prismatic joint: μία πρισματική άρθρωση έχει ένα βαθμό ελευθερίας και χρησιμοποιείται για να περιγράψει μεταλλακτική κίνηση μεταξύ αντικειμένων. Η μετάφραση συμβαίνει κατά μήκος του άξονα z του πλαισίου αναφοράς του.
- Βίδα/Screw: μία βίδα, η οποία μπορεί να θεωρηθεί ως συνδυασμός περιστροφικής και πρισματικής άρθρωσης με συνδεδεμένες τιμές, έχει ένα βαθμό ελευθερίας και χρησιμοποιείται για να περιγράψει μια κίνηση παρόμοια με τη βίδα. Ένας παράγοντας ηγεσίας καθορίζει την ποσότητα της μετάφρασης για μια πλήρη περιστροφή. Η μετάφραση/περιστροφή συμβαίνει κατά μήκος/γύρω από τον άξονα z του πλαισίου αναφοράς του.
- Σφαιρική άρθρωση/Spherical joint: μία σφαιρική άρθρωση έχει τρεις βαθμούς ελευθερίας και χρησιμοποιείται για να περιγράψει περιστροφική κίνηση μεταξύ αντικειμένων. Σε ορισμένες καταστάσεις, μία σφαιρική άρθρωση μπορεί να θεωρηθεί ως 3 ταυτόχρονες και ορθογώνιες αρθρώσεις, οι οποίες είναι γονείς σε μια ιεραρχική αλυσίδα. Η αναλογία όμως είναι έγκυρη μόνο όταν όλοι οι περιστροφικές αρθρώσεις διατηρούν μια προσανατολισμένη θέση διαφορετική από τους άλλους δύο: πράγματι, αν δύο αρθρώσεις έρθουν κοντά στο να συμπέσουν, ενδέχεται να προκύψει μια ασυνήθιστη κατάσταση και ο μηχανισμός θα χάσει ένα βαθμό ελευθερίας. Αυτό δεν συμβαίνει ποτέ με τις σφαιρικές αρθρώσεις, οι οποίες χειρίζονται εσωτερικά για να αποφεύγουν αυτού του είδους τις καταστάσεις:



1.4.27 Καταστάσεις αρθρώσεων/Joint modes

Μία άρθρωση μπορεί να βρίσκεται σε μία από τις ακόλουθες 3 καταστάσεις:

- **Κινηματική κατάσταση/Kinematic mode:** Η άρθρωση κινείται σε κινηματική κατάσταση και δεν χειρίζεται από τη δυναμική μονάδα. Μπορεί να αλλάξει αμέσως τη γραμμική/γωνιακή της θέση (μέσω `sim.setJointPosition`), να εμφανίσει ένα συγκεκριμένο προφίλ κίνησης (μέσω `sim.setJointTargetPosition` ή `sim.setJointTargetVelocity`) ή να χειριστεί με εξατομικευμένο τρόπο μέσω μιας συνάρτησης επανάκλησης αρθρώσεων (joint callback function).
- **Εξαρτημένη κατάσταση/Dependent mode:** Η άρθρωση εξαρτάται άμεσα από μια άλλη άρθρωση μέσω γραμμικής εξίσωσης. Η άρθρωση δε χειρίζεται από τη δυναμική μονάδα.
- **Δυναμική κατάσταση/Dynamic mode:** Η άρθρωση χειρίζεται από τη δυναμική μονάδα, αν σχηματίζει με τα συνδεδεμένα στοιχεία μια έγκυρη διάταξη, δηλαδή είναι δυναμικά ενεργοποιημένη. Σε αυτήν την περίπτωση, μπορεί να χειριστεί με διάφορους τρόπους.

Υπάρχουν πολλοί διαφορετικοί τρόποι ελέγχου της άρθρωσης, ανάλογα με την κατάσταση της άρθρωσης και την κατάσταση ελέγχου, αλλά μπορεί να διαφοροποιηθεί ανάμεσα σε ελέγχους υψηλού επιπέδου και ελέγχους χαμηλού επιπέδου:

Ο έλεγχος **υψηλού επιπέδου** επιτυγχάνεται κυρίως μέσω συγκεκριμένων συναρτήσεων API, όπως `sim.setJointPosition`, `sim.setJointTargetPosition`, `sim.setJointTargetVelocity` ή `sim.setJointTargetForce`. Ανάλογα με την κατάσταση της άρθρωσης και την κατάσταση ελέγχου, δεν είναι λογικό να χρησιμοποιηθούν όλες οι συναρτήσεις, για παράδειγμα καλούμε `sim.setJointTargetForce` σε μια κινηματική άρθρωση ή `sim.setJointTargetPosition` σε μια δυναμική άρθρωση σε έλεγχο ταχύτητας.

Ο έλεγχος **χαμηλού επιπέδου** είναι καλύτερα να υλοποιείται μέσω μιας συνάρτησης επανάκλησης άρθρωσης από μέσα στο `CoppeliaSim`. Σε αυτήν την περίπτωση, η άρθρωση θα πρέπει να είναι σε κινηματική κατάσταση ή σε δυναμική κατάσταση και το `CoppeliaSim`. Η συνάρτηση επανάκλησης θα καλείται συνεχώς από το `CoppeliaSim` για να λάβει νέες τιμές ρύθμισης.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Οι συναρτήσεις επανάκλησης άρθρωσης είναι ιδανικές για τον χαμηλού επιπέδου έλεγχο μιας άρθρωσης αφού ο ελεγκτής θα ρυθμίσει την άρθρωση σε κάθε βήμα χρόνου, δηλαδή το να παραλείψει ο ελεγκτής ένα βήμα ρύθμισης για έναν ελεγκτή χαμηλού επιπέδου δεν είναι μία επιλογή. Οι αρθρώσεις σε κινηματική κατάσταση θα χρησιμοποιήσουν το ίδιο βήμα ρύθμισης όπως το βήμα χρόνου της προσομοίωσης (προεπιλογή 50 ms). Από την άλλη πλευρά, οι αρθρώσεις σε δυναμική κατάσταση θα χρησιμοποιήσουν ένα πολύ λεπτομερές βήμα ρύθμισης όπως ορίζεται από τη δυναμική μονάδα (προεπιλογή 5 ms).

Στην περίπτωση μιας εξωτερικής εφαρμογής που επιθυμεί να πραγματοποιήσει χαμηλού επιπέδου έλεγχο μιας άρθρωσης, οι συναρτήσεις επανάκλησης άρθρωσης δεν είναι διαθέσιμες, οπότε η εξωτερική εφαρμογή θα πρέπει να διασφαλίσει ότι θα λειτουργεί συγχρονισμένα με το CoppeliaSim. Αυτό συμβαίνει μέσω της λειτουργίας βημάτων, όπου η εξωτερική εφαρμογή είναι υπεύθυνη για την ενεργοποίηση κάθε βήματος προσομοίωσης εκ των προτέρων. Επιπλέον, για έλεγχο υψηλής ακρίβειας, το βήμα χρόνου της προσομοίωσης θα πρέπει να είναι το ίδιο με το βήμα χρόνου της δυναμικής μονάδας.

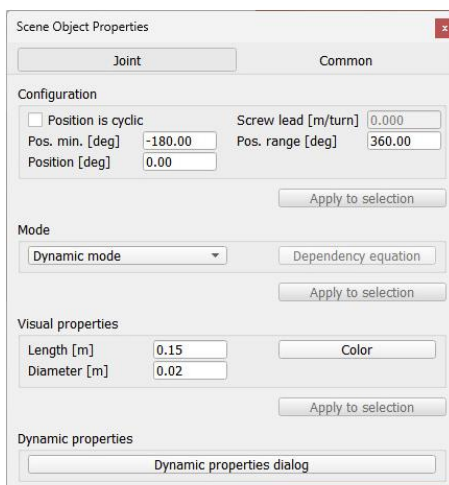
1.4.28 Ιδιότητες αρθρώσεων/Joint properties

Οι ιδιότητες αρθρώσεων είναι μέρος του διαλόγου ιδιοτήτων αντικειμένων της σκηνής, ο οποίος βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το διάλογο με διπλό κλικ στο εικονίδιο του αντικειμένου στην ιεραρχία της σκηνής ή με ένα κλικ στο κουμπί της γραμμής εργαλείων του:



[Scene object properties toolbar button]

Στο διάλογο ιδιοτήτων αντικειμένων σκηνής, κάνουμε κλικ στο κουμπί **Joint** για να εμφανίσουμε το παράθυρο διαλόγου αρθρώσεων (το κουμπί Joint εμφανίζεται μόνο αν η τελευταία επιλογή είναι μια άρθρωση). Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους της τελευταίας επιλεγμένης άρθρωσης. Αν είναι επιλεγμένες περισσότερες από μία αρθρώσεις, τότε κάποιες παράμετροι μπορούν να αντιγραφούν από την τελευταία επιλεγμένη άρθρωση στις άλλες επιλεγμένες αρθρώσεις (**Apply to selection**). Να σημειωθεί ότι αυτό έχει αποτέλεσμα μόνο μεταξύ αρθρώσεων του ίδιου τύπου ή κατάστασης):



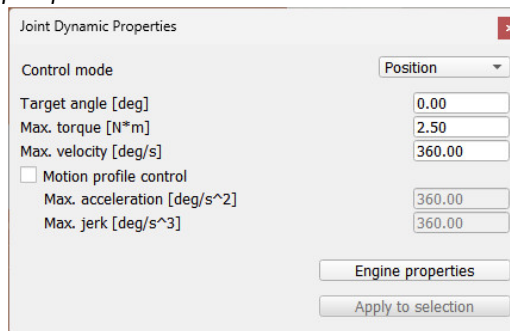
[Joint dialog]

Προσομοίωση Ρομπότ μέσω Υπολογιστή

- **Θέση είναι κυκλική/Position is cyclic:** υποδεικνύει αν η γωνιακή θέση της άρθρωσης είναι κυκλική (μεταβάλλεται μεταξύ -180 και +180 μοίρες χωρίς περιορισμό). Μόνο οι περιστροφικές αρθρώσεις μπορούν να είναι κυκλικές.
- **Βήμα βίδας/Screw lead:** η μετατόπιση κατά τον άξονα z ανά περιστροφή. Αυτή η ιδιότητα είναι διαθέσιμη μόνο με τον τύπο άρθρωσης Περιστροφική/Βίδα/Revolute/Screw, όταν το **Position is cyclic** είναι ξεμαρκαρισμένο και η άρθρωση είναι κινηματική.
- **Ελάχιστη θέση/Position minimum:** η ελάχιστη επιτρεπόμενη τιμή μιας μη κυκλικής περιστροφικής άρθρωσης, μιας βίδας ή μιας πρισματικής άρθρωσης.
- **Εύρος θέσης/Position range:** το εύρος μεταβολής μιας μη κυκλικής περιστροφικής άρθρωσης, μιας βίδας ή μιας πρισματικής άρθρωσης. Η θέση μιας τέτοιας άρθρωσης περιορίζεται μεταξύ της **Ελάχιστης θέσης/Position minimum** και της **Ελάχιστης θέσης + Εύρος θέσης/Position minimum+Position range**.
- **Θέση/Position:** γραμμική/γωνιακή θέση της άρθρωσης.
- **Λειτουργία/Mode:** μια άρθρωση μπορεί να βρίσκεται σε κινηματική, εξαρτημένη ή δυναμική κατάσταση.
- **Εξίσωση εξάρτησης/Dependency equation:** αν η άρθρωση είναι σε εξαρτημένη κατάσταση, τότε μπορεί να οριστεί μια γραμμική εξίσωση που να συνδέει την άρθρωση με μια άλλη. Οι τιμές σε αυτήν την ενότητα του διαλόγου αναφέρονται σε μέτρα ή ακτίνια.
- **Μήκος/Length:** το μήκος της άρθρωσης. Δεν έχει λειτουργική σημασία.
- **Διάμετρος/Diameter:** η διάμετρος της άρθρωσης. Δεν έχει λειτουργική σημασία.
- **Χρώμα/Color:** επιτρέπει τη ρύθμιση του χρώματος της άρθρωσης.
- **Παράθυρο διαλόγου δυναμικών ιδιοτήτων/Dynamic properties dialog:** ενεργοποιεί το παράθυρο διαλόγου δυναμικών ιδιοτήτων της άρθρωσης. Το παράθυρο διαλόγου δυναμικών ιδιοτήτων της άρθρωσης επιτρέπει τη ρύθμιση των δυναμικών ιδιοτήτων της άρθρωσης.

1.4.29 Δυναμικές ιδιότητες αρθρώσεων/Joint dynamics properties

Οι δυναμικές ιδιότητες αρθρώσεων είναι μέρος των ιδιοτήτων αρθρώσεων. Το παράθυρο διαλόγου διάλογος τους εμφανίζει τις ρυθμίσεις και τις παραμέτρους της δυναμικής της τελευταίας επιλεγμένης άρθρωσης. Αν δεν έχει επιλεγεί καμία άρθρωση ή η άρθρωση δεν είναι δυναμική, τότε ο διάλογος είναι ανενεργός. Αν είναι επιλεγμένες περισσότερες από μία αρθρώσεις, τότε κάποιες παράμετροι μπορούν να αντιγραφούν από την τελευταία επιλεγμένη άρθρωση στις άλλες επιλεγμένες αρθρώσεις:



[Joint dynamics dialog]

Υποστηρίζονται διάφορες λειτουργίες ελέγχου:

- **Ελεύθερη (χωρίς έλεγχο)/Free (no control):** η άρθρωση είναι ελεύθερη, δηλαδή μη μηχανοκίνητη.
- **Έλεγχος δύναμης/ροπής/Force/torque control:** η άρθρωση ελέγχεται στη δύναμη/ροπή, δηλαδή εφαρμόζεται μια σταθερή δύναμη/ροπή. Μπορεί να ρυθμιστεί με το `sim.setJointTargetForce`.
- **Έλεγχος ταχύτητας/Velocity control:** η άρθρωση ελέγχεται στην ταχύτητα: η καθορισμένη δύναμη/ροπή εφαρμόζεται μέχρι να επιτευχθεί η επιθυμητή ταχύτητα. Προαιρετικά, μπορεί να εφαρμοστεί ένα συγκεκριμένο προφίλ κίνησης για την ταχύτητα. Μπορεί να ρυθμιστεί με το `sim.setJointTargetVelocity` και το `sim.setJointTargetForce`.
- **Έλεγχος θέσης/Position control:** η άρθρωση ελέγχεται στη θέση: η καθορισμένη δύναμη/ροπή εφαρμόζεται και η ταχύτητα ρυθμίζεται ανάλογα, μέχρι να επιτευχθεί η επιθυμητή θέση/γωνία. Οι παράμετροι μπορούν να ρυθμιστούν με το `sim.setJointTargetPosition`, το `sim.setJointTargetForce` και το `sim.setObjectFloatParam`.
- **Έλεγχος ελατηρίου-αποσβεστήρα/Spring-damper control:** η άρθρωση ελέγχεται στη θέση με την τροποποίηση της ασκούμενης δύναμης/ροπής μέσω ενός απλού ελεγκτή ΚC, προσπαθώντας να φτάσει στη θέση/γωνία μηδενικής μετατόπισης. Οι παράμετροι μπορούν να ρυθμιστούν με το `sim.setJointTargetPosition`, το `sim.setJointTargetForce` και το `sim.setObjectFloatParam`.
- **Προσαρμοσμένος έλεγχος/Custom control:** η άρθρωση καλεί μια συνάρτηση επανάκλησης αρθρώσεως για έλεγχο, όπου ο χρήστης μπορεί να καθορίσει μια δύναμη/ροπή και μια στόχευση ταχύτητας με ευέλικτο τρόπο.

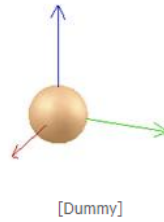
Εκτός από τις παραπάνω ρυθμίσεις, οι αρθρώσεις έχουν συγκεκριμένες παραμέτρους σχετικές με κάθε μηχανή φυσικής που μπορούν να ρυθμιστούν από το **Engine properties**.

1.4.30 Ανδρείκελα ή Ομοιώματα/Dummies

Ένα αντικείμενο ομοίωμα είναι το πιο απλό διαθέσιμο αντικείμενο: είναι ένα σημείο με προσανατολισμό και μπορεί να θεωρηθεί ως πλαίσιο αναφοράς. Τα ομοιώματα είναι αντικείμενα πολλαπλών χρήσεων ή βοηθητικά αντικείμενα: χρησιμοποιούνται μόνο τους για να εντοπίζουν συγκεκριμένα σημεία ή πλαίσια αναφοράς στη σκηνή και επίσης χρησιμοποιούνται σε ζεύγη για να

Προσομοίωση Ρομπότ μέσω Υπολογιστή

καθορίσουν κλεισίματα βρόχων ή σχέσεις άκρης-στόχου για υπολογισμούς δυναμικής ή κινηματικής. Η ακόλουθη εικόνα δείχνει ένα ομοίωμα:



Τα ομοιώματα είναι συγκρουόμενα, μετρήσιμα και ανιχνεύσιμα αντικείμενα. Αυτό σημαίνει ότι:

- μπορούν να χρησιμοποιηθούν σε ανιχνεύσεις συγκρούσεων με άλλα συγκρουόμενα αντικείμενα που είναι βασισμένα σε όγκο, όπως δέντρα OC.
- μπορούν να χρησιμοποιηθούν σε υπολογισμούς ελάχιστης απόστασης με άλλα μετρήσιμα αντικείμενα.
- μπορούν να ανιχνευθούν από αισθητήρες εγγύτητας.

Από προεπιλογή, η συγκρουόμενη, μετρήσιμη και ανιχνεύσιμη ιδιότητα ενός ομοιώματος είναι απενεργοποιημένη.

Τα ομοιώματα μπορούν να προστεθούν στη σκηνή με [Add → Dummy] ή να δημιουργηθούν από ένα αντικείμενο σχήματος στη λειτουργία επεξεργασίας κορυφών.

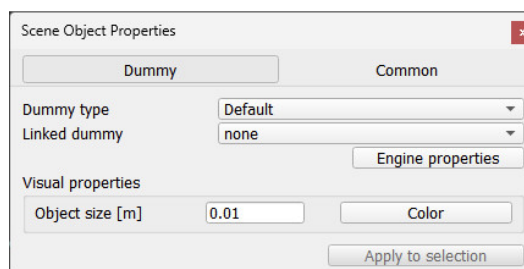
1.4.31 Ιδιότητες ομοιώματος/Dummy properties

Οι ιδιότητες ομοιώματος είναι μέρος του διαλόγου ιδιοτήτων αντικειμένου σκηνής, ο οποίος βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το παράθυρο διαλόγου με διπλό κλικ σε ένα εικονίδιο αντικειμένου στην ιεραρχία σκηνής ή με κλικ στο κουμπί της γραμμής εργαλείων του:



[Scene object properties toolbar button]

Στο παράθυρο διαλόγου ιδιοτήτων αντικειμένου σκηνής, κάνουμε κλικ στο κουμπί **Dummy** για να εμφανιστεί ο διάλογος ομοιώματος (το κουμπί **Dummy** εμφανίζεται μόνο αν η τελευταία επιλογή είναι ομοίωμα). Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους του τελευταίου επιλεγμένου ομοιώματος. Αν είναι επιλεγμένα περισσότερα από ένα ομοιώματα, τότε κάποιες παράμετροι μπορούν να αντιγραφούν από το τελευταίο επιλεγμένο ομοίωμα στα άλλα επιλεγμένα ομοιώματα (**Apply to selection**):



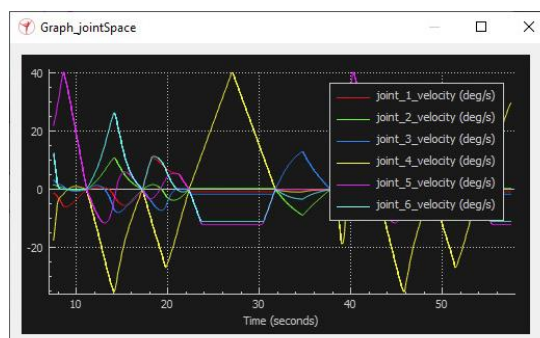
[Dummy dialog]

- **Τύπος ομοιώματος/Dummy type:**
 - *Προεπιλογή/Default:* το ομοίωμα δεν έχει εξειδικευμένη συμπεριφορά ή λειτουργία.
 - *Συναρμολόγηση/Assembly:* το ομοίωμα μπορεί να λειτουργήσει ως σημείο συναρμολόγησης.
 - *Δυναμικό, περιορισμός επικάλυψης/Dynamic, overlap constraint:* το ομοίωμα και το συνδεδεμένο αντίστοιχο του θα προσπαθήσουν να επικαλύψουν τη θέση/προσανατολισμό τους, επιτρέποντας τη δημιουργία ενός δυναμικού κλεισίματος βρόχου.
 - *Δυναμικό, περιορισμός τένοντα/Dynamic, tendon constraint:* το ομοίωμα και το συνδεδεμένο αντίστοιχο του θα συμπεριφέρονται σαν να υπάρχει μια χορδή ή ένας ελαστικός περιορισμός μεταξύ τους. Προς το παρόν υποστηρίζεται μόνο από τη μηχανή MuJoCo.
- **Συνδεδεμένο ομοίωμα/Linked dummy:** ένα προαιρετικό ομοίωμα συνδεδεμένο με αυτό. Τα συνδεδεμένα ομοιώματα μπορούν να έχουν ειδικές ιδιότητες ή συμπεριφορά.
- **Ιδιότητες μηχανής/Engine properties:** κάθε μηχανή φυσικής έχει συγκεκριμένες παραμέτρους σχετικές με τα ομοιώματα. Πρέπει να βεβαιωθούμε ότι έχουμε διαβάσει τις αντίστοιχες τεκμηριώσεις της μηχανής για να επιτύχουμε τις καλύτερες ρυθμίσεις παραμέτρων.
- **Μέγεθος αντικειμένου/Object size:** το μέγεθος του ομοιώματος. Αυτή η παράμετρος έχει μόνο οπτικό αποτέλεσμα και καμία λειτουργική σημασία.
- **Χρώμα/Color:** επιτρέπει την προσαρμογή του χρώματος ενός ομοιώματος.

1.4.32 Γραφήματα/Graphs

Τα γραφήματα είναι αντικείμενα σκηνής που μπορούν να καταγράψουν και να οπτικοποιήσουν δεδομένα από μια προσομοίωση. Τα δεδομένα καταγράφονται σε ροές δεδομένων, οι οποίες είναι σειριακές λίστες τιμών που συνδέονται με χρονικές σημάνσεις.

Οι ροές δεδομένων μπορούν να οπτικοποιηθούν άμεσα ως διαγράμματα χρόνου. Συνδυάζοντας 2 ή 3 ροές δεδομένων, μπορούμε να δημιουργήσουμε καμπύλες X/Y ή 3D καμπύλες στη σκηνή. Η ακόλουθη εικόνα απεικονίζει ένα διάγραμμα χρόνου δείχνοντας 6 ροές δεδομένων που οπτικοποιούν τις ταχύτητες αρθρώσεων:



[Examples of a time plot]

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Ένα γράφημα προστίθεται στη σκηνή με [Add → Graph]. Οι βασικές του ιδιότητες είναι προσβάσιμες και προσαρμόζονται στο παράθυρο διαλόγου γραφήματος.

Ο χρήστης είναι υπεύθυνος για τον ορισμό των ροών δεδομένων, των καμπυλών και την τροφοδότηση τους με κατάλληλα δεδομένα σε τακτική βάση (συνήθως μία φορά ανά βήμα προσομοίωσης). Το ακόλουθο παράδειγμα δείχνει πώς να καταγράψουμε και να οπτικοποιήσουμε 2 ροές δεδομένων (τη ταχύτητα δύο αρθρώσεων σε σχέση με τον χρόνο):

Με χρήση κώδικα **Python**:

```
#python

#import math

def sysCall_init():
    sim = require('sim')
    self.joint1Handle = sim.getObject('/joint1')
    self.joint2Handle = sim.getObject('/joint2')
    self.graph = sim.getObject('/Graph')
    self.joint1Vel = sim.addGraphStream(self.graph, 'joint 1 velocity', 'deg/s', 0, [1, 0, 0])
    self.joint2Vel = sim.addGraphStream(self.graph, 'joint 2 velocity', 'deg/s', 0, [0, 1, 0])

def sysCall_sensing():
    sim.setGraphStreamValue(self.graph, self.joint1Vel, 180 * sim.getJointVelocity(self.joint1Handle) / math.pi)
    sim.setGraphStreamValue(self.graph, self.joint2Vel, 180 * sim.getJointVelocity(self.joint2Handle) / math.pi)
```

ή με χρήση **Lua**:

```
--lua

function sysCall_init()
    sim = require('sim')
    joint1Handle = sim.getObject('/joint1')
    joint2Handle = sim.getObject('/joint2')
    graph = sim.getObject('/Graph')
    joint1Vel = sim.addGraphStream(graph, 'joint 1 velocity', 'deg/s', 0, {1, 0, 0})
    joint2Vel = sim.addGraphStream(graph, 'joint 2 velocity', 'deg/s', 0, {0, 1, 0})
end

function sysCall_sensing()
    sim.setGraphStreamValue(graph, joint1Vel, 180 * sim.getJointVelocity(joint1Handle) / math.pi)
    sim.setGraphStreamValue(graph, joint2Vel, 180 * sim.getJointVelocity(joint2Handle) / math.pi)
end
```

Το επόμενο παράδειγμα δείχνει πώς να καταγράψουμε και να οπτικοποιήσουμε μια καμπύλη θέσης X/Y ενός αντικειμένου:

Με χρήση κώδικα **Python**:

```
#python
```



```
def sysCall_init():
    sim = require('sim')
    self.objectHandle = sim.getObject('/object')
    self.graph = sim.getObject('/Graph')
    self.objectPosX = sim.addGraphStream(self.graph, 'object pos x', 'm', 1)
    self.objectPosY = sim.addGraphStream(self.graph, 'object pos y', 'm', 1)
    sim.addGraphCurve(self.graph, 'object pos x/y', 2, [objectPosX, objectPosY], [0, 0], 'm by m')
```

```
def sysCall_sensing():
    pos = sim.getObjectPosition(self.objectHandle)
    sim.setGraphStreamValue(self.graph, self.objectPosX, pos[0])
    sim.setGraphStreamValue(self.graph, self.objectPosY, pos[1])
```

ή με χρήση **Lua**:

--lua

```
function sysCall_init()
    sim = require('sim')
    objectHandle = sim.getObject('/object')
    graph = sim.getObject('/Graph')
    objectPosX = sim.addGraphStream(graph, 'object pos x', 'm', 1)
    objectPosY = sim.addGraphStream(graph, 'object pos y', 'm', 1)
    sim.addGraphCurve(graph, 'object pos x/y', 2, {objectPosX, objectPosY}, {0, 0}, 'm by m')
end

function sysCall_sensing()
    local pos = sim.getObjectPosition(objectHandle)
    sim.setGraphStreamValue(graph, objectPosX, pos[1])
    sim.setGraphStreamValue(graph, objectPosY, pos[2])
end
```

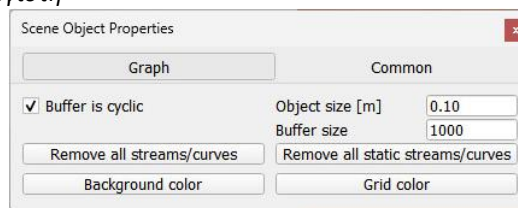
1.4.33 Παράθυρο διαλόγου γραφήματος/Graph dialog

Το παράθυρο διαλόγου γραφήματος είναι μέρος του παραθύρου διαλόγου ιδιοτήτων αντικειμένων σκηνής, το οποίο βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το παράθυρο διαλόγου με διπλό κλικ στο εικονίδιο ενός αντικειμένου στην ιεραρχία σκηνής ή με ένα κλικ στο κουμπί της γραμμής εργαλείων του:



[Scene object properties toolbar button]

Στο παράθυρο διαλόγου ιδιοτήτων αντικειμένων σκηνής, κάνουμε κλικ στο κουμπί **Graph** για να εμφανιστεί ο διάλογος γραφήματος (το κουμπί **Graph** εμφανίζεται μόνο αν η τελευταία επιλογή είναι ένα γράφημα). Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους του τελευταίου επιλεγμένου γραφήματος:



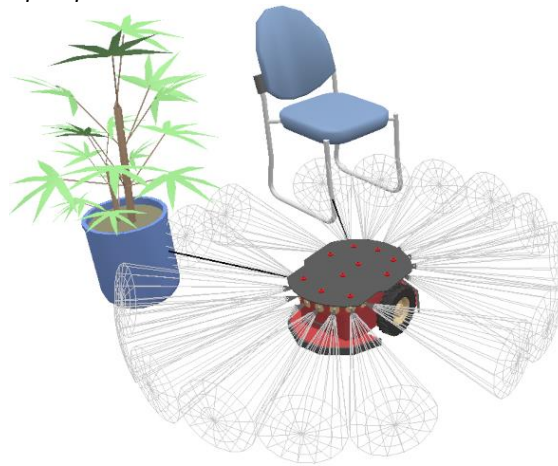
[Graph dialog]

- **Μέγεθος αντικειμένου/Object size:** το μέγεθος του αντικειμένου γραφήματος. Αυτή η παράμετρος δεν έχει λειτουργικό αποτέλεσμα.
- **Ο buffer είναι κυκλικός/Buffer is cyclic:** όταν επιλεγεί και ο αριθμός των στοιχείων σε κάθε ροή δεδομένων φτάσει το μέγεθος του buffer, τότε τα πρώτα στοιχεία θα διαγραφούν για να μπορούν να καταγραφούν νέα στοιχεία. Διαφορετικά, η καταγραφή σταματά όταν ο buffer γεμίσει.
- **Μέγεθος buffer/Buffer size:** ο αριθμός των στοιχείων που μπορεί να καταγράψει κάθε ροή δεδομένων.
- **Αφαίρεση όλων των ροών/καμπυλών/Remove all streams/curves:** διαγράφει όλες τις ροές δεδομένων και καμπύλες.
- **Αφαίρεση όλων των στατικών ροών/καμπυλών/Remove all static streams/curves:** διαγράφει όλες τις στατικές ροές δεδομένων και καμπύλες. Μια στατική ροή δεδομένων ή καμπύλη είναι μια "παγωμένη" ροή δεδομένων ή καμπύλη, που δεν μπορεί να τροποποιηθεί πλέον.
- **Χρώμα φόντου/Background color:** επιτρέπει την προσαρμογή του χρώματος φόντου στην προβολή διαγράμματος χρόνου ή X/Y διαγράμματος.
- **Χρώμα πλέγματος/Grid color:** επιτρέπει την προσαρμογή του χρώματος του πλέγματος και των διαβαθμίσεων στην προβολή διαγράμματος χρόνου ή x/y διαγράμματος.

Το ίδιο το αντικείμενο γραφήματος λειτουργεί κυρίως μέσω ενός προσαρμοσμένου script, που μπορεί να προσαρμοστεί με διάφορους τρόπους, εάν χρειαστεί. Η συμπεριφορά του γραφήματος μπορεί επίσης να τροποποιηθεί με διάφορους τρόπους κάνοντας κλικ στο εικονίδιο παραμέτρων χρήστη δίπλα στο προσαρμοσμένο script.

1.4.34 Αισθητήρες εγγύτητας/Proximity sensors

Το CoppeliaSim προσφέρει έναν πολύ ισχυρό και αποδοτικό τρόπο προσομοίωσης αισθητήρων εγγύτητας. Ο χρήστης μπορεί να μοντελοποιήσει σχεδόν οποιοδήποτε τύπο αισθητήρα εγγύτητας, από υπερηχητικούς έως υπέρυθρους, κ.λπ. Τα αντικείμενα της σκηνής που επιτρέπουν αυτή τη λειτουργικότητα είναι οι αισθητήρες εγγύτητας (οι οποίοι διαφέρουν από τους αισθητήρες όρασης) που μπορούν να ανιχνεύουν ανιχνεύσιμες οντότητες. Οι ακόλουθες εικόνες απεικονίζουν προσομοιώσεις χρησιμοποιώντας αισθητήρες εγγύτητας:



[Mobile robot using proximity sensors]

Οι αισθητήρες εγγύτητας προστίθενται στη σκηνή με [Add → Proximity sensor].

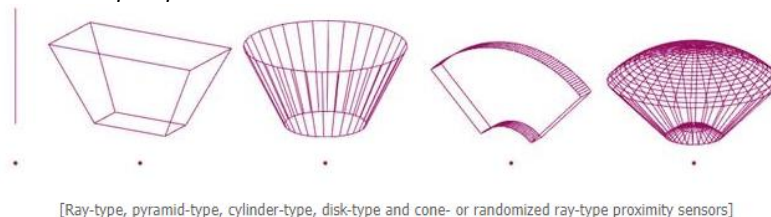
Επίσης οι ρουτίνες ανίχνευσης αισθητήρων εγγύτητας που χρησιμοποιούνται από τους αισθητήρες εγγύτητας είναι διαθέσιμες ως αυτόνομες ρουτίνες μέσω των γεωμετρικών ρουτινών Coppelia.

1.4.35 Τύποι αισθητήρων εγγύτητας και τρόπος λειτουργίας/Proximity sensor types and more of operation

Οι αισθητήρες εγγύτητας έρχονται σε 6 διαφορετικούς τύπους και μπορούν να προσαρμοστούν σε μεγάλο βαθμό:

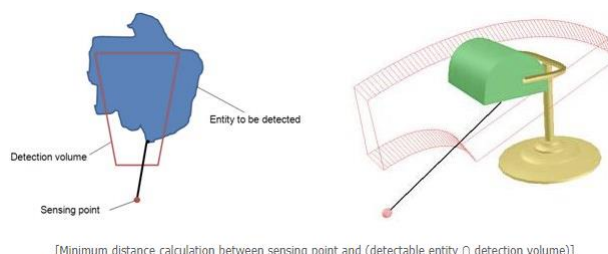
- **Τύπου ακτίνας/Ray-type:** ο αισθητήρας εγγύτητας τύπου ακτίνας είναι ιδανικός για πολύ απλή μοντελοποίηση ενός αισθητήρα εγγύτητας ή για μοντελοποίηση ενός λείζερ εύρεσης απόστασης. Είναι οι ταχύτεροι αισθητήρες εγγύτητας.
- **Τύπου ακτίνας με τυχαίο δείγμα/Randomized ray-type:** ο αισθητήρας εγγύτητας τύπου ακτίνας με τυχαίο δείγμα λειτουργεί ως αισθητήρας τύπου ακτίνας που σαρώνει τυχαία έναν όγκο κώνου. Έχει παρόμοια εμφάνιση με τον αισθητήρα τύπου κώνου.
- **Τύπου πυραμίδας/Pyramid-type:** ο αισθητήρας εγγύτητας τύπου πυραμίδας είναι ιδανικός για απλή μοντελοποίηση ενός αισθητήρα εγγύτητας με ορθογώνιο όγκο ανίχνευσης. Είναι πολύ γρήγοροι.
- **Τύπου κυλίνδρου/Cylinder-type:** ο αισθητήρας εγγύτητας τύπου κυλίνδρου είναι ιδανικός για απλή μοντελοποίηση ενός αισθητήρα εγγύτητας με περιστρεφόμενο όγκο ανίχνευσης. Είναι πολύ γρήγοροι.
- **Τύπου δίσκου/Disk-type:** ο αισθητήρας εγγύτητας τύπου δίσκου επιτρέπει την ακριβή μοντελοποίηση ενός αισθητήρα εγγύτητας με περιστρεφόμενο όγκο ανίχνευσης. Ανάλογα με την επιλεγμένη ακρίβεια και τρόπο λειτουργίας, μπορεί να είναι λίγο πιο απαιτητικοί υπολογιστικά.
- **Τύπου κώνου/Cone-type:** ο αισθητήρας εγγύτητας τύπου κώνου επιτρέπει την καλύτερη και πιο ακριβή μοντελοποίηση των περισσότερων αισθητήρων εγγύτητας. Ανάλογα με την επιλεγμένη ακρίβεια και τρόπο λειτουργίας, μπορεί να είναι λίγο πιο απαιτητικοί υπολογιστικά.

Προσομοίωση Ρομπότ μέσω Υπολογιστή



Οι αισθητήρες εγγύτητας λειτουργούν με γεωμετρικά ακριβή τρόπο: εκτελούν έναν ακριβή υπολογισμό απόστασης μεταξύ του σημείου ανίχνευσης (μικρή σφαίρα) και οποιασδήποτε ανιχνεύσιμης οντότητας που παρεμβαίνει στον όγκο ανίχνευσης (δεν εκτελούν απλώς έναν απλό έλεγχο σύγκρουσης μεταξύ των άκρων του όγκου ανίχνευσης όπως τα περισσότερα άλλα προγράμματα προσομοίωσης, αλλά έναν ακριβή υπολογισμό απόστασης εντός του όγκου ανίχνευσης). Κάθε αισθητήρας εγγύτητας θα υπολογίζει την ακόλουθη ελάχιστη απόσταση:

Σημείο ανίχνευσης <-- --> (ανιχνεύσιμη οντότητα \cap όγκος ανίχνευσης)



Εάν ένας αισθητήρας εγγύτητας ανιχνεύσει ένα αντικείμενο, τότε ενεργοποιείται ένας μηχανισμός πυροδότησης, ο οποίος θα έχει ως αποτέλεσμα την κλήση της συνάρτησης επανάκλησης πυροδότησης.

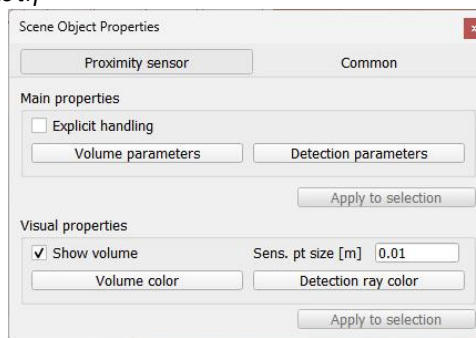
1.4.36 Ιδιότητες Αισθητήρων Εγγύτητας/Proximity sensor properties

Οι ιδιότητες των αισθητήρων εγγύτητας αποτελούν μέρος του διαλόγου ιδιοτήτων αντικειμένων σκηνής, ο οποίος βρίσκεται στο [Tools \rightarrow Scene object properties]. Μπορούμε επίσης να ανοίξουμε το παράθυρο διαλόγου με διπλό κλικ σε ένα εικονίδιο αντικειμένου στην ιεραρχία σκηνής ή με ένα κλικ στο κουμπί της γραμμής εργαλείων:



[Scene object properties toolbar button]

Στο παράθυρο διαλόγου ιδιοτήτων αντικειμένων σκηνής, κάνουμε κλικ στο κουμπί **Proximity sensor** για να εμφανιστεί το παράθυρο διαλόγου αισθητήρα εγγύτητας (το κουμπί **Proximity sensor** εμφανίζεται μόνο εάν η τελευταία επιλογή είναι αισθητήρας εγγύτητας). Ο διάλογος εμφανίζει τις ρυθμίσεις και τις παραμέτρους του τελευταίου επιλεγμένου αισθητήρα εγγύτητας. Εάν έχουν επιλεγεί περισσότεροι από ένας αισθητήρες εγγύτητας, τότε κάποιες παράμετροι μπορούν να αντιγραφούν από τον τελευταίο επιλεγμένο αισθητήρα στους υπόλοιπους επιλεγμένους αισθητήρες (**Apply to selection**):

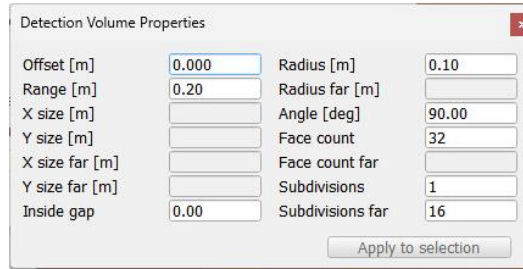


[Proximity sensor dialog]

- **Ρητός χειρισμός/Explicit handling:** υποδεικνύει εάν ο αισθητήρας πρέπει να χειρίζεται ρητά. Εάν είναι επιλεγμένο, ο αισθητήρας δεν θα χειρίζεται όταν καλείται η εντολή `sim.handleProximitySensor(sim.handle_all_except_explicit)` αλλά μόνο εάν καλείται η εντολή `sim.handleProximitySensor(sim.handle_all)` ή `sim.handleProximitySensor(proximitySensorHandle)`. Αυτό είναι χρήσιμο εάν επιθυμούμε να χειριζόμαστε τον αισθητήρα σε ένα σενάριο προσομοίωσης αντί στο κύριο σενάριο (εάν δεν είναι επιλεγμένο, ο αισθητήρας θα χειρίζεται δύο φορές, μία όταν καλείται η εντολή `sim.handleProximitySensor(sim.handle_all_except_explicit)` στο κύριο σενάριο και μία όταν καλείται η εντολή `sim.handleProximitySensor(proximitySensorHandle)` στο σενάριο προσομοίωσης).
- **Παράμετροι όγκου/Volume parameters:** εναλλάσσει το διάλογο όγκου αισθητήρα εγγύτητας. Αυτός ο διάλογος επιτρέπει την προσαρμογή του όγκου ανίχνευσης του αισθητήρα εγγύτητας.
- **Παράμετροι ανίχνευσης/Detection parameters:** ανοίγει το διάλογο παραμέτρων ανίχνευσης του αισθητήρα εγγύτητας. Αυτός ο διάλογος επιτρέπει την προσαρμογή διάφορων παραμέτρων ανίχνευσης.
- **Όγκος/Volume:** εμφανίζει ή αποκρύπτει τον όγκο ανίχνευσης.
- **Μέγεθος σημείου ανίχνευσης/Sensing point size:** το μέγεθος του σημείου ανίχνευσης. Αυτό δεν έχει καμία επίδραση στους υπολογισμούς (το σημείο ανίχνευσης, ακόμα και αν εμφανίζεται μεγάλο, πάντα θεωρείται ως σημείο).
- **Χρώμα όγκου/Volume color:** επιτρέπει την προσαρμογή του χρώματος του όγκου του αισθητήρα.
- **Χρώμα ακτίνας ανίχνευσης/Detection ray color:** επιτρέπει την προσαρμογή του χρώματος της ακτίνας ανίχνευσης του αισθητήρα.

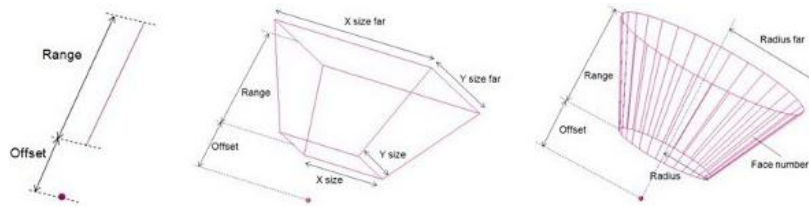
1.4.37 Παράθυρο διαλόγου όγκου αισθητήρα εγγύτητας/Proximity sensor volume dialog

Το παράθυρο διαλόγου όγκου αισθητήρα εγγύτητας αποτελεί μέρος των ιδιοτήτων του αισθητήρα εγγύτητας. Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους όγκου του τελευταίου επιλεγμένου αισθητήρα εγγύτητας. Εάν δεν είναι επιλεγμένο κανένα αντικείμενο, Το παράθυρο διαλόγου είναι ανενεργός. Εάν έχουν επιλεγεί περισσότεροι από ένας αισθητήρες εγγύτητας, τότε κάποιες παράμετροι μπορούν να αντιγραφούν από τον τελευταίο επιλεγμένο αισθητήρα στους υπόλοιπους επιλεγμένους αισθητήρες (**Apply to selection**):

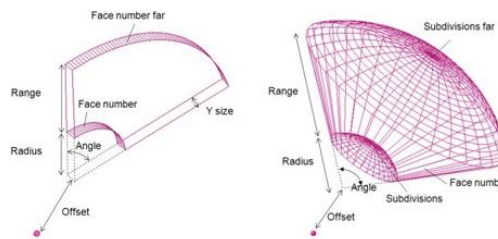


[Proximity sensor volume dialog]

Οι έξι τύποι όγκου ενός αισθητήρα εγγύτητας (ακτίνα, τυχαία ακτίνα, πυραμίδα, κύλινδρος, δίσκος και κώνος) ορίζονται μέσω διαφόρων παραμέτρων:

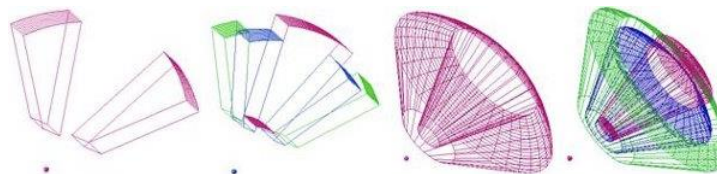


[Ray, pyramid and cylinder-type detection volume parameters]



[Disk and cone-type detection volume parameters]

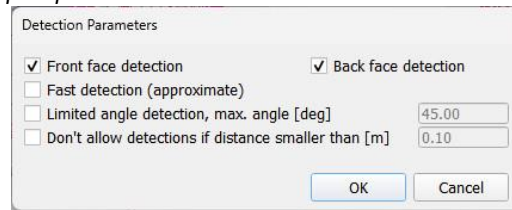
- **Face count, face count far, subdivisions, subdivisions far:** Χρησιμοποιούνται για να περιγράψουν τον όγκο ανίχνευσης με περισσότερη ή λιγότερη ακρίβεια. Περισσότερη ακρίβεια συνεπάγεται και μεγαλύτερους χρόνους υπολογισμού.
- **inside gap:** Παράμετρος διαθέσιμη μόνο για τους αισθητήρες εγγύτητας τύπου δίσκου και κώνου. Η τιμή της μπορεί να κυμαίνεται μεταξύ 0 και 1 και αντιπροσωπεύει το ποσοστό του όγκου που θα αφαιρεθεί. Αυτή η δυνατότητα μπορεί να χρησιμοποιηθεί για τη δημιουργία πολύπλοκων όγκων ανίχνευσης με τον συνδυασμό πολλών αισθητήρων:



[Inside gap functionality and its application]

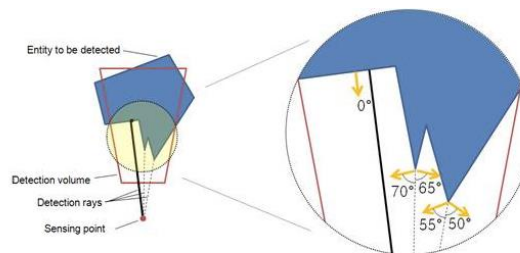
1.4.38 Παράθυρο διαλόγου παραμέτρων ανίχνευσης αισθητήρα εγγύτητας/Proximity sensor detection parameter dialog

Ο διάλογος παραμέτρων ανίχνευσης αισθητήρα εγγύτητας αποτελεί μέρος των ιδιοτήτων του αισθητήρα εγγύτητας. Ο διάλογος εμφανίζει διάφορες παραμέτρους ανίχνευσης του τελευταίου επιλεγμένου αισθητήρα εγγύτητας:

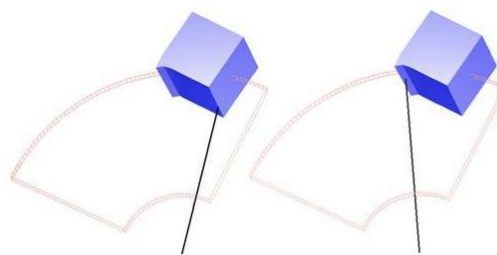


[Proximity sensor detection parameter dialog]

- **Ανίχνευση εμπρόσθιων / οπίσθιων επιφανειών/Front / back face detection:** Μπορούμε να αποφασίσουμε ποια πλευρά ενός τριγώνου θα βλέπει και θα ανιχνεύει ο αισθητήρας. Οι εμπρόσθιες επιφάνειες εμφανίζονται μπλε, ενώ οι οπίσθιες επιφάνειες εμφανίζονται κόκκινες στη λειτουργία επεξεργασίας τριγώνου.
- **Γρήγορη ανίχνευση (προσεγγιστική)/Fast detection (approximate):** Όταν επιλέγεται, η διαδικασία ανίχνευσης θα είναι πιο γρήγορη, αλλά πλέον όχι ακριβής (δηλ. το ανιχνευμένο σημείο μπορεί να μην είναι το κοντινότερο σημείο).
- **Περιορισμένη γωνιακή ανίχνευση/Limited angle detection:** Όταν επιλέγεται, μπορεί να καθοριστεί μια μέγιστη γωνία μεταξύ της ακτίνας ανίχνευσης και του διανύσματος κανονικής επιφάνειας ενός αντικειμένου. Αυτή είναι μια χρήσιμη λειτουργία κατά τη μοντελοποίηση υπερηχητικών αισθητήρων εγγύτητας. Οι υπερηχητικοί αισθητήρες εγγύτητας κανονικά δεν μπορούν να δουν επιφάνειες που δεν είναι αρκετά στραμμένες προς τον αισθητήρα. Όταν ενεργοποιείται και η μέγιστη γωνία είναι μικρή, ο χρόνος υπολογισμού μπορεί να αυξηθεί δραστικά.



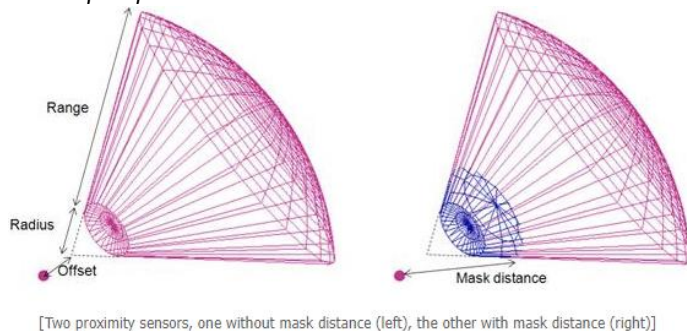
[Limited angle detection principle. Yellow arrows indicate surfaces normal vectors. Maximum angle is 30° (for example)]



[Two identical proximity sensors configurations. Only one proximity sensor has the limited angle detection turned on (right)]

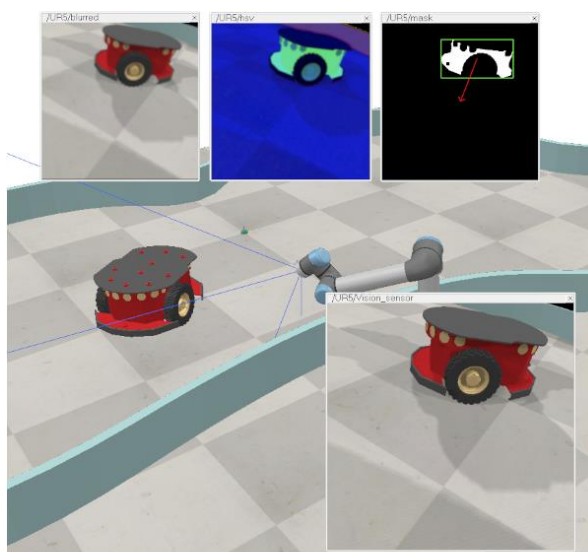
- **Μην επιτρέπετε ανιχνεύσεις αν η απόσταση είναι μικρότερη από/Don't allow detections if distance smaller than:** Κάποιοι αισθητήρες έχουν μια ελάχιστη απόσταση ανίχνευσης (π.χ. υπερηχητικοί αισθητήρες) από την οποία ο αισθητήρας δεν θα λειτουργεί πλέον (δηλ. το αντικείμενο που θα ανιχνευθεί είναι πολύ κοντά στον αισθητήρα). Για να μοντελοποιηθεί αυτή η συμπεριφορά, μπορούμε να ορίσουμε απλώς μια μετατόπιση για τον όγκο ανίχνευσης, αλλά κάνοντάς το αυτό, ο αισθητήρας θα μπορεί ακόμα να ανιχνεύει άλλα αντικείμενα που βρίσκονται πιο μακριά. Ένας πραγματικός αισθητήρας θα είχε το "πεδίο θέασης" του μπλοκαρισμένο από το κοντινότερο αντικείμενο και δεν θα ανίχνευε τίποτα. Αυτό μπορεί να μοντελοποιηθεί με το καθορισμό μιας ελάχιστης απόστασης ανίχνευσης (μία απόσταση μάσκας) που αν υποσκελιστεί θα απενεργοποιήσει απλώς την ανίχνευση.

Προσομοίωση Ρομπότ μέσω Υπολογιστή



1.4.39 Αισθητήρες όρασης/Vision sensors

Το CoppeliaSim προσφέρει, εκτός από αισθητήρες προσέγγισης και έναν άλλο τύπο αισθητήρων: τους αισθητήρες όρασης. Οι αισθητήρες όρασης, που είναι ορατά αντικείμενα, λειτουργούν με πολύ παρόμοιο τρόπο όπως τα αντικείμενα κάμερας: θα αποδίδουν τα αντικείμενα που βρίσκονται στο πεδίο τους και θα ενεργοποιούν την ανίχνευση αν οι καθορισμένα όρια υπερβούν ή δεν φτάσουν. Οι αισθητήρες όρασης θα πρέπει να χρησιμοποιούνται αντί των αισθητήρων εγγύτητας κυρίως όταν το χρώμα, το φως ή η δομή παίζει ρόλο στη διαδικασία ανίχνευσης (π.χ. υπέρυθροι αισθητήρες ή γενικότερα, αισθητήρες ευαίσθητοι στο φως (κάμερες κ.λπ.)). Ωστόσο, ανάλογα με την κάρτα γραφικών στην οποία εκτελείται η εφαρμογή ή την πολυπλοκότητα των αντικειμένων της σκηνής, οι αισθητήρες όρασης μπορεί να είναι λίγο πιο αργόί από τους αισθητήρες εγγύτητας.



[Image acquisition and processing]

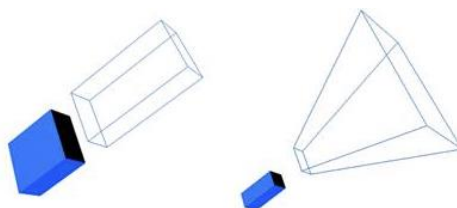
Πρέπει να βεβαιωθούμε ότι δεν συγχέουμε τους αισθητήρες όρασης με τις κάμερες. Οι κύριες διαφορές:

Ένας αισθητήρας όρασης έχει σταθερή ανάλυση. Μια κάμερα δεν έχει συγκεκριμένη ανάλυση (δηλαδή προσαρμόζεται αυτόματα στο μέγεθος της προβολής). Το περιεχόμενο της εικόνας ενός αισθητήρα όρασης μπορεί να προσπελαστεί μέσω του API και η επεξεργασία εικόνας πραγματοποιείται μέσω των λειτουργιών ανάκλησης αισθητήρα όρασης. Το περιεχόμενο της εικόνας μιας κάμερας δεν είναι άμεσα διαθέσιμο μέσω του API. Ένας αισθητήρας όρασης γενικά απαιτεί περισσότερο χρόνο CPU και λειτουργεί πιο αργά από τις κάμερες. Ένας αισθητήρας όρασης μπορεί κυρίως να εμφανίζει σχήματα. Μια κάμερα μπορεί να εμφανίζει όλους τους τύπους αντικειμένων. Οι αισθητήρες όρασης προστίθενται στη σκηνή με [Add → Vision sensor].

1.4.40 Τύποι αισθητήρων όρασης και τρόπος λειτουργίας/Vision sensor types and mode of operation

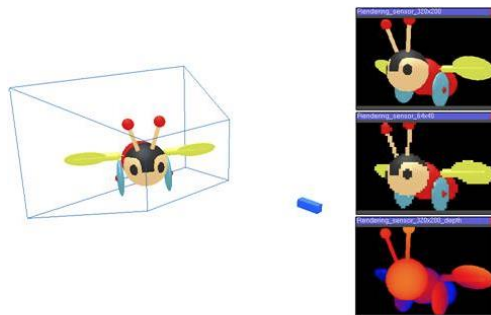
Οι αισθητήρες όρασης διατίθενται σε 2 διαφορετικούς τύπους και μπορούν να προσαρμοστούν για διάφορους σκοπούς:

- **Τύπος ορθογραφικής προβολής/Orthographic projection-type:** το πεδίο προβολής των αισθητήρων όρασης τύπου ορθογραφικής προβολής είναι ορθογώνιο. Είναι κατάλληλοι για αισθητήρες υπέρυθρων κοντινής απόστασης ή λέιζερ ανίχνευσης αποστάσεων.
- **Τύπος προοπτικής προβολής/Perspective projection-type:** το πεδίο προβολής των αισθητήρων όρασης τύπου προοπτικής προβολής είναι τραπεζοειδές. Είναι κατάλληλοι για αισθητήρες τύπου κάμερας.



[Orthogonal projection-type and perspective projection-type vision sensors]

Οι αισθητήρες όρασης, που είναι ορατά αντικείμενα, μπορούν να προβληθούν όπως τα αντικείμενα κάμερας [Popup menu → View → Associate view with selected vision sensor].



[Vision sensor and various image processings]

Οι αισθητήρες όρασης είναι πολύ ισχυροί με την έννοια ότι μπορούν να χρησιμοποιηθούν με διάφορους και ευέλικτους τρόπους. Για παράδειγμα, μπορούν να χρησιμοποιηθούν για την προβολή στατικών ή κινούμενων εικόνων από μια εξωτερική εφαρμογή ή ένα plugin. Τα plugins μπορούν επίσης να παρέχουν εξατομικευμένους αλγόριθμους επεξεργασίας εικόνας καθώς και αλγόριθμους αξιολόγησης (π.χ. συνθήκες ενεργοποίησης). Οι συναρτήσεις ανάκλησης όρασης αντιπροσωπεύουν τον μηχανισμό όπου μπορεί να πραγματοποιηθεί η επεξεργασία εικόνας και όπου μπορούν να δημιουργηθούν ενεργοποιήσεις.

1.4.41 Ιδιότητες αισθητήρων όρασης/Vision sensor properties

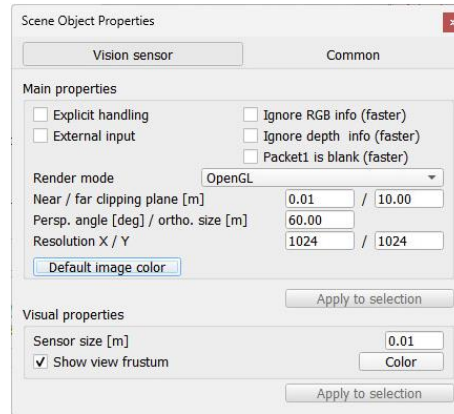
Οι ιδιότητες των αισθητήρων όρασης ανήκουν στο πλαίσιο ιδιοτήτων αντικειμένου στη σκηνή, το οποίο βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το πλαίσιο διαλόγου με διπλό κλικ σε ένα εικονίδιο αντικειμένου στην ιεραρχία σκηνής ή με κλικ στο κουμπί εργαλειοθήκης του.



[Scene object properties toolbar button]

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Στο παράθυρο διαλόγου ιδιοτήτων αντικειμένου σκηνης, κάνουμε κλικ στο κουμπί **Vision sensor** για να εμφανιστεί το παράθυρο διαλόγου αισθητήρα όρασης (το κουμπί **Vision sensor** εμφανίζεται μόνο αν η τελευταία επιλογή είναι ένας αισθητήρας όρασης). Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους του τελευταίου επιλεγμένου αισθητήρα όρασης. Αν επιλεγούν περισσότεροι από ένας αισθητήρες όρασης, τότε κάποιες παράμετροι μπορούν να αντιγραφούν από τον τελευταίο επιλεγμένο αισθητήρα όρασης στους άλλους επιλεγμένους αισθητήρες όρασης (**Apply to selection**).



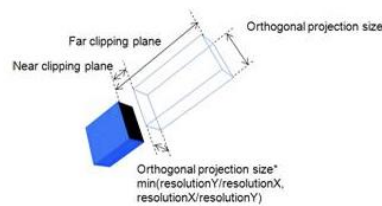
[Vision sensor dialog]

- **Εμφανής χειρισμός/Explicit handling:** υποδεικνύει εάν ο αισθητήρας θα πρέπει να χειρίζεται εμφανώς. Αν είναι επιλεγμένο, ο αισθητήρας δεν θα χειριστεί όταν κληθεί η συνάρτηση `sim.handleVisionSensor(sim.handle_all_except_explicit)`, αλλά μόνο όταν κληθεί `sim.handleVisionSensor(sim.handle_all)` ή `sim.handleVisionSensor(visionSensorHandle)`. Αυτό είναι χρήσιμο εάν επιθυμούμε να χειριζόμαστε τον αισθητήρα σε ένα σενάριο προσομοίωσης αντί για το κύριο σενάριο (εάν δεν είναι επιλεγμένος ο αισθητήρας θα χειριστεί δύο φορές, μία όταν κληθεί `sim.handleVisionSensor(sim.handle_all_except_explicit)` στο κύριο σενάριο και μία όταν κληθεί `sim.handleVisionSensor(visionSensorHandle)` στο σενάριο προσομοίωσης).
- **Εξωτερική είσοδος/External input:** εάν επιλεγεί, τότε η κανονική λειτουργία του αισθητήρα όρασης θα τροποποιηθεί για να μπορεί να χειρίζεται εξωτερικές εικόνες αντί αυτών (π.χ. εικόνες βίντεο).
- **Αγνόηση πληροφοριών RGB (ταχύτερα)/Ignore RGB info (faster):** εάν επιλεγεί, οι πληροφορίες RGB του αισθητήρα (δηλαδή το χρώμα) θα αγνοηθούν για να λειτουργεί πιο γρήγορα. Χρησιμοποιήστε αυτήν την επιλογή εάν εξαρτάστε μόνο από τις πληροφορίες βάθους του αισθητήρα.
- **Αγνόηση πληροφοριών βάθους (ταχύτερα)/Ignore depth info (faster):** εάν επιλεγεί, οι πληροφορίες βάθους του αισθητήρα θα αγνοηθούν για να λειτουργεί πιο γρήγορα. Χρησιμοποιήστε αυτήν την επιλογή εάν δεν προτίθεστε να χρησιμοποιήσετε τις πληροφορίες βάθους του αισθητήρα.
- **Packet1 είναι κενό (ταχύτερα)/Packet1 is blank (faster):** εάν επιλεγεί, τότε το CoppeliaSim δεν θα εξάγει αυτόματα συγκεκριμένες πληροφορίες από τις αποκτηθείσες εικόνες, ώστε να μπορεί να λειτουργήσει πιο γρήγορα. Χρησιμοποιήστε αυτήν την επιλογή εάν δεν προτίθεστε

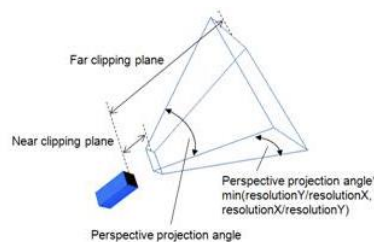
Προσομοίωση Ρομπότ μέσω Υπολογιστή

να χρησιμοποιήσετε το πρώτο πακέτο βοηθητικών τιμών που επιστρέφονται από τις συναρτήσεις API `sim.readVisionSensor` ή `sim.handleVisionSensor`.

- **Λειτουργία αποδόσεως/Render mode:** τρεις λειτουργίες είναι διαθέσιμες αυτή τη στιγμή:
 - **Παλιά OpenGL (προεπιλεγμένο)/ Legacy OpenGL (default):** αποδίδει τα ορατά χρωματικά κανάλια των αντικειμένων.
 - **Παλιά OpenGL, κωδικοποίηση χρωμάτων λαβών/Legacy OpenGL, color coded handles:** αποδίδει τα αντικείμενα κωδικοποιώντας τις λαβές τους στα χρώματα. Το πρώτο πακέτο δεδομένων που επιστρέφεται από τις συναρτήσεις API `sim.readVisionSensor` ή `sim.handleVisionSensor` αντιπροσωπεύει τις εντοπισμένες λαβές αντικειμένων (στρογγυλοποιημένες κάτω).
 - **POV-Ray:** χρησιμοποιεί το πρόσθετο POV-Ray για την απεικόνιση εικόνων, επιτρέποντας σκιές (επίσης μαλακές σκιές) και εφέ υλικού (πολύ πιο αργό).
 - **OpenGL3:** χρησιμοποιεί το πρόσθετο `simOpenGL3`. Το πρόσθετο προσφέρει ρύψη σκιών, η οποία προς το παρόν δεν είναι δυνατή φυσικά στο CoppeliaSim. Η προβολή φωτός και οι σκιές μπορούν να ρυθμιστούν για κάθε φως, μέσω της συμπληρωματικής συμβολοσειράς του αντικειμένου π.χ. `openGL3 {lightProjection {nearPlane {0.1} farPlane {10} orthoSize {8} bias {0.001} normalBias {0.005} shadowTextureSize {2048}}}`
- **Εγγύς / μακρινό επίπεδο απόκομμα/Near / far clipping plane:** η ελάχιστη / μέγιστη απόσταση από την οποία ο αισθητήρας θα είναι σε θέση να ανιχνεύσει.
- **Γωνία προοπτικής/Perspective angle:** η μέγιστη γωνία ανοίγματος του όγκου ανίχνευσης όταν ο αισθητήρας είναι σε λειτουργία προοπτικής.
- **Μέγεθος ορθογραφικού/Orthographic size:** το μέγιστο μέγεθος (κατά μήκος x ή y) του όγκου ανίχνευσης όταν ο αισθητήρας δεν είναι σε λειτουργία προοπτικής.



[Detection value parameters of the orthographic-type vision sensor]



[Detection value parameters of the perspective-type vision sensor]

- **Ανάλυση X / Y/Resolution X/Y:** επιθυμητή ανάλυση x- / y- της εικόνας που απαθανατίζεται από τον αισθητήρα όρασης. Επιλέγουμε προσεκτικά την ανάλυση ανάλογα με την εφαρμογή

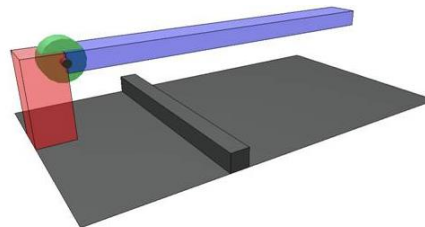
Προσομοίωση Ρομπότ μέσω Υπολογιστή

μας (υψηλή ανάλυση θα οδηγήσει σε πιο αργή λειτουργία). Με παλαιότερα μοντέλα γραφικών, η πραγματική ανάλυση μπορεί να είναι διαφορετική από αυτή που εμφανίζεται εδώ (παλαιά μοντέλα γραφικών υποστηρίζουν μόνο αναλύσεις στο 2^n , όπου n είναι 0, 1, 2 κλπ.).

- **Προεπιλεγμένο χρώμα εικόνας/Default image color:** επιτρέπει την καθορισμένη χρήση του χρώματος που θα πρέπει να χρησιμοποιηθεί στις περιοχές όπου δεν απεικονίστηκε τίποτα. Από προεπιλογή, χρησιμοποιείται το χρώμα ομίχλης περιβάλλοντος.
- **Μέγεθος αισθητήρα/Sensor size:** μέγεθος του σώματος-μέρους του αισθητήρα όρασης. Δεν έχει λειτουργική επίδραση.
- **Εμφάνιση κορυφογραμμής προβολής/Show view frustum:** εάν επιλεγεί, εμφανίζεται η κορυφογραμμή προβολής (όγκος).
- **Χρώμα/Color:** επιτρέπει τη ρύθμιση του χρώματος του αισθητήρα.

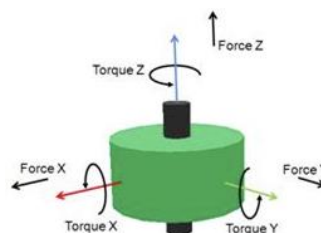
1.4.42 Αισθητήρες δύναμης/Force sensors

Οι αισθητήρες δύναμης είναι αρχικά άκαμπτοι σύνδεσμοι μεταξύ δύο σχημάτων που είναι σε θέση να μετρήσουν τις μεταδιδόμενες δυνάμεις και τις ροπές. Η ακαμψία των αισθητήρων δύναμης είναι υπό όρους, με την έννοια ότι οι αισθητήρες δύναμης μπορούν να σπάσουν αν προκύψει μια συγκεκριμένη συνθήκη (π.χ. αν ξεπεραστεί ένα όριο δύναμης ή ροπής). Το παρακάτω σχήμα απεικονίζει μια εφαρμογή που χρησιμοποιεί έναν αισθητήρα δύναμης:



Αντικείμενο αισθητήρα δύναμης (πράσινο) που μετρά τη δύναμη και τη ροπή που ασκείται από μια δοκό (μπλε) αγκυρωμένη σε έναν τοίχο (κόκκινο)

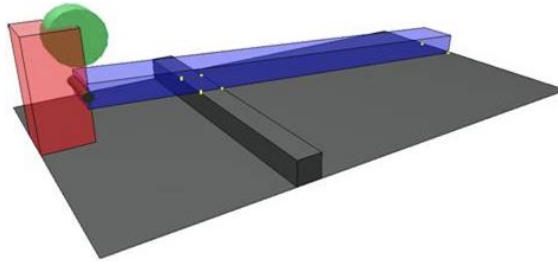
Ένας αισθητήρας δύναμης μετρά ένα ζεύγος από 3 τιμές που αντιπροσωπεύουν τη δύναμη στον αισθητήρα κατά μήκος των αξόνων x , y και z και τη ροπή στον αισθητήρα γύρω από τους άξονες x , y και z :



Δυνάμεις και ροπές που μετρώνται από έναν αισθητήρα δύναμης

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Αρχικά, ένας αισθητήρας δύναμης λειτουργεί ως άκαμπτος σύνδεσμος. Κατά τη διάρκεια της προσομοίωσης, ένας αισθητήρας δύναμης μπορεί να σπάσει όταν ξεπεραστεί ένα καθορισμένο όριο δύναμης/ροπής ή όταν πληρούνται κάποιες άλλες οριζόμενες από τον χρήστη συνθήκες. Τα παρακάτω σχήματα απεικονίζουν την κατάσταση *σπασίματος* ενός αισθητήρα δύναμης:



Σπασμένος αισθητήρας δύναμης

Ένας αισθητήρας δύναμης είναι λειτουργικός κατά τη διάρκεια της προσομοίωσης μόνο αν είναι δυναμικά ενεργοποιημένος. Οι αρθρώσεις είναι επίσης σε θέση να μετρήσουν μία δύναμη ή μία ροπή αλλά μόνο κατά μήκος/γύρω από τον άξονά τους z.

Οι αισθητήρες δύναμης προστίθενται στη σκηνή με [Add → Force sensor].

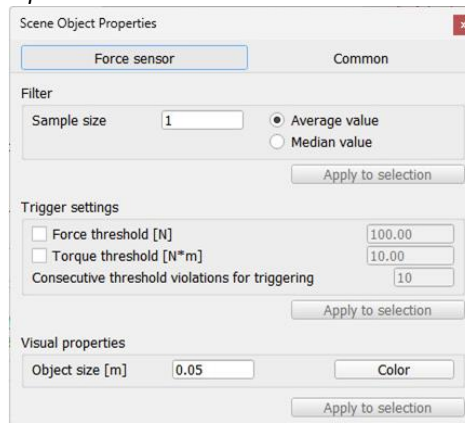
1.4.43 Ιδιότητες αισθητήρα δύναμης/Force sensor properties

Οι ιδιότητες των αισθητήρων δύναμης ανήκουν στο πλαίσιο ιδιοτήτων αντικειμένου σκηνής, το οποίο βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το παράθυρο διαλόγου με διπλό κλικ σε ένα εικονίδιο αντικειμένου στην ιεραρχία σκηνής ή με κλικ στο κουμπί εργαλειοθήκης του.



Κουμπί εργαλειοθήκης ιδιοτήτων αντικειμένου σκηνής

Στο παράθυρο διαλόγου ιδιοτήτων αντικειμένου σκηνής, κάνουμε κλικ στο κουμπί **Force sensor** για να εμφανιστεί το παράθυρο διαλόγου αισθητήρα δύναμης (το κουμπί **Force sensor** εμφανίζεται μόνο αν η τελευταία επιλογή είναι ένας αισθητήρας δύναμης). Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους του τελευταίου επιλεγμένου αισθητήρα δύναμης. Αν επιλεγούν περισσότεροι από ένας αισθητήρες δύναμης, τότε κάποιες παράμετροι μπορούν να αντιγραφούν από τον τελευταίο επιλεγμένο αισθητήρα δύναμης στους άλλους επιλεγμένους αισθητήρες δύναμης (**Apply to selection**).



Διάλογος αισθητήρα δύναμης

- **Φίλτρο/Filter:** όταν δεδομένα δύναμης ή ροπής αποκτώνται από τον αισθητήρα δύναμης, τότε συσσωρεύονται και μπορούν να φιλτραριστούν για να ληφθούν λιγότερο ασταθείς τιμές.
- **Μέγεθος δείγματος/Sample size:** ο αριθμός των τιμών που θα χρησιμοποιηθούν για το φίλτρο. *I* δεν θα φιλτράρει τιμές (ακατέργαστη έξοδος).
- **Μέση τιμή/Average value:** αν επιλεγεί, τότε ο αισθητήρας δύναμης θα δίνει μέσες τιμές (μέσος όρος τιμών **μεγέθους δείγματος**).
- **Διάμεση τιμή/Median value:** αν επιλεγεί, τότε ο αισθητήρας δύναμης θα δίνει διάμεσες τιμές (διάμεσος των τιμών **μεγέθους δείγματος**).
- **Ρυθμίσεις ενεργοποίησης/Trigger settings:** αυτή η ενότητα επιτρέπει τη ρύθμιση κάποιων αυτόματων συνθηκών ενεργοποίησης. Μια ενεργοποίηση θα καλέσει τη συνάρτηση ανάκλησης ενεργοποίησης (μέσα στην οποία, για παράδειγμα, μπορεί να σπάσει ο αισθητήρας δύναμης).
- **Όριο δύναμης/Force threshold:** το μέγεθος του διανύσματος δύναμης που ενεργοποιεί την παραβίαση ενός ορίου.
- **Όριο ροπής/Torque threshold:** το μέγεθος του διανύσματος ροπής που ενεργοποιεί την παραβίαση ενός ορίου.
- **Συνεχόμενες παραβιάσεις ορίου για θραύση/Consecutive threshold violations for breaking:** ο αριθμός των συνεχόμενων φορών που επιτρέπεται στον αισθητήρα να παραβιάσει ένα όριο πριν από την ενεργοποίηση.
- **Μέγεθος αντικειμένου/Object size:** το μέγεθος της τρισδιάστατης αναπαράστασης του αισθητήρα. Αυτό δεν έχει λειτουργική επίδραση.
- **Χρώμα/Color:** επιτρέπει τη ρύθμιση του χρώματος του αισθητήρα.

1.4.44 Δέντρα OC/OC trees

Ένα OC δέντρο είναι ένα αντικείμενο που αναπαριστά τη χωρική κατανομή. Αποτελείται από μια δομή δέντρου δεδομένων στην οποία κάθε κόμβος έχει ακριβώς οκτώ παιδιά. Οι κατειλημμένοι κόμβοι φύλλα αναπαριστώνται ως voxel. Τα OC δέντρα μπορούν να χρησιμοποιηθούν για την παροχή μιας απλουστευμένης αναπαράστασης σχημάτων ή νεφών σημείων ή μπορούν να λειτουργήσουν ως πλέγμα/χώρος κατάληψης:



Τα OC δέντρα είναι συγκρουόμενα, μετρήσιμα και ανιχνεύσιμα αντικείμενα. Αυτό σημαίνει ότι τα OC δέντρα:

- μπορούν να χρησιμοποιηθούν σε ανιχνεύσεις συγκρούσεων με άλλα συγκρουόμενα αντικείμενα.
- μπορούν να χρησιμοποιηθούν σε υπολογισμούς ελάχιστης απόστασης με άλλα μετρήσιμα αντικείμενα.
- μπορούν να ανιχνευθούν από αισθητήρες εγγύτητας.

Τα OC δέντρα μπορούν να προστεθούν στη σκηνή με [Add → OC tree] και να επεξεργαστούν μέσω των ιδιοτήτων του OC δέντρου.

Οι υπολογισμοί OC δέντρων (δηλ. συγκρούσεις, αποστάσεις και υπολογισμοί αισθητήρων εγγύτητας) που είναι διαθέσιμοι στο CoppeliaSim, είναι επίσης διαθέσιμοι ως ανεξάρτητες ρουτίνες μέσω των γεωμετρικών ρουτίνων του Coppelia.

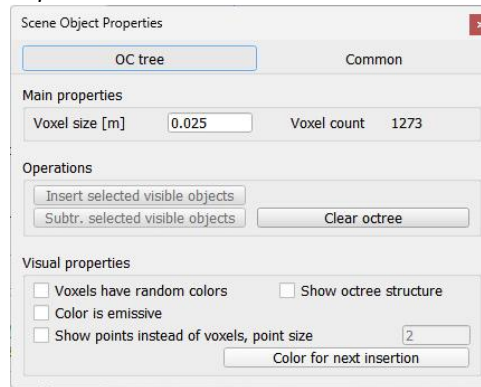
Ιδιότητες OC δέντρου/OC tree properties

Οι ιδιότητες των OC δέντρων ανήκουν στο παράθυρο διαλόγου ιδιοτήτων αντικειμένου σκηνής, το οποίο βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το παράθυρο διαλόγου με διπλό κλικ σε ένα εικονίδιο αντικειμένου στην ιεραρχία σκηνής ή με κλικ στο κουμπί της εργαλειοθήκης.



[Scene object properties toolbar button]

Στο παράθυρο διαλόγου ιδιοτήτων αντικειμένου σκηνής, κάνουμε κλικ στο κουμπί **OC tree** για να εμφανιστεί ο διάλογος OC δέντρου (το κουμπί **OC tree** εμφανίζεται μόνο αν η τελευταία επιλογή είναι ένα OC tree). Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους του τελευταίου επιλεγμένου OC δέντρου:

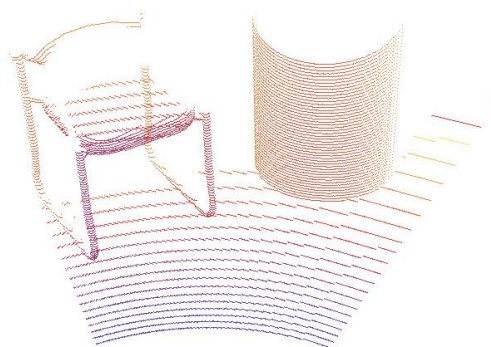


[OC tree dialog]

- **Μέγεθος voxel/Voxel size:** το μέγεθος των voxel του OC δέντρου. Όσο μικρότερο το μέγεθος, τόσο περισσότερη μνήμη θα χρησιμοποιηθεί για τη δομή δεδομένων του OC δέντρου.
- **Εισαγωγή επιλεγμένων ορατών αντικειμένων/Insert selected visible objects:** επιτρέπει την εισαγωγή voxel βάσει των επιλεγμένων, ορατών αντικειμένων. Λαμβάνονται υπόψη μόνο τα dummies, σχήματα, OC δέντρα και νέφη σημείων. Επιλέξτε όλα τα αντικείμενα που θέλετε να εισαγάγετε και επιλέξτε τελευταίο το στόχο OC δέντρο.
- **Αφαίρεση επιλεγμένων ορατών αντικειμένων/Subtract selected visible objects:** επιτρέπει την αφαίρεση voxel βάσει των επιλεγμένων, ορατών αντικειμένων. Λαμβάνονται υπόψη μόνο τα dummies, σχήματα, OC δέντρα και νέφη σημείων. Επιλέξτε όλα τα αντικείμενα που θέλετε να αφαιρέσετε και επιλέξτε τελευταίο το στόχο OC δέντρο.
- **Καθαρισμός OC δέντρου/Clear OC tree:** αφαιρεί όλα τα voxel από το OC δέντρο. Τα voxel έχουν τυχαία χρώματα: κάθε voxel θα έχει ένα τυχαίο χρώμα.
- **Εμφάνιση δομής OC δέντρου/Voxels have random colors:** εμφανίζει τη δομή δεδομένων του OC δέντρου, κυρίως για σκοπούς εντοπισμού σφαλμάτων.
- **Το χρώμα είναι εκπεμπόμενο/Color is emissive:** τα voxel θα εμφανίζονται με εκπεμπόμενα χρώματα.
- **Εμφάνιση σημείων αντί για voxel/Show points instead of voxels:** για OC δέντρα που περιέχουν μεγάλο αριθμό voxel, η εμφάνιση μπορεί να είναι αρκετά αργή. Σε αυτήν την περίπτωση, έχουμε τη δυνατότητα να εμφανίζουμε voxel σε μορφή pixel αντί για κύβους.
- **Χρώμα για επόμενη εισαγωγή/Color for next insertion:** επιτρέπει την επιλογή νέου χρώματος, το οποίο θα χρησιμοποιηθεί για νέες εισαγωγές voxel.

1.4.45 Νέφη σημείων/Point clouds

Ένα νέφος σημείων είναι ένα αντικείμενο που λειτουργεί ως δοχείο σημείων βασισμένο σε OC δέντρο:



[Point cloud]

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Τα νέφη σημείων είναι συγκρουόμενα, μετρήσιμα και ανιχνεύσιμα αντικείμενα. Αυτό σημαίνει ότι:

- μπορούν να χρησιμοποιηθούν σε ανιχνεύσεις συγκρούσεων με άλλα συγκρουόμενα αντικείμενα που είναι βασισμένα σε όγκο, όπως τα OC δέντρα.
- μπορούν να χρησιμοποιηθούν σε υπολογισμούς ελάχιστης απόστασης με άλλα μετρήσιμα αντικείμενα.
- μπορούν να ανιχνευθούν από αισθητήρες προσέγγισης.

Τα νέφη σημείων μπορούν να προστεθούν στη σκηνή με [Add → Point cloud] και να επεξεργαστούν μέσω των ιδιοτήτων νέφους σημείων.

Οι υπολογισμοί νέφους σημείων (δηλ. συγκρούσεις, αποστάσεις και υπολογισμοί αισθητήρων προσέγγισης) που είναι διαθέσιμοι στο CoppeliaSim είναι επίσης διαθέσιμοι ως ανεξάρτητες ρουτίνες μέσω των γεωμετρικών ρουτίνων του Coppelia.

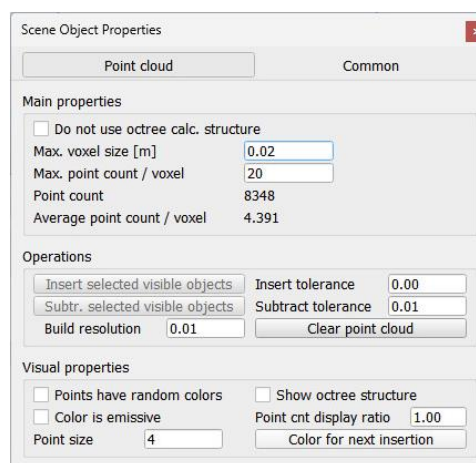
1.4.46 Ιδιότητες νέφους σημείων/Point cloud properties

Οι ιδιότητες των νεφών σημείων ανήκουν στο παράθυρο διαλόγου ιδιοτήτων αντικειμένου σκηνής, το οποίο βρίσκεται στο [Tools → Scene object properties]. Μπορούμε επίσης να ανοίξουμε το παράθυρο διαλόγου με διπλό κλικ σε ένα εικονίδιο αντικειμένου στην ιεραρχία σκηνής ή με κλικ στο κουμπί της εργαλειοθήκης.



[Scene object properties toolbar button]

Στο πλαίσιο διαλόγου ιδιοτήτων αντικειμένου σκηνής, κάνουμε κλικ στο κουμπί **Point cloud** για να εμφανιστεί το παράθυρο διαλόγου OC δέντρου (το κουμπί **Point cloud** εμφανίζεται μόνο αν η τελευταία επιλογή είναι ένα νέφος σημείων). Το παράθυρο διαλόγου εμφανίζει τις ρυθμίσεις και τις παραμέτρους του τελευταίου επιλεγμένου νέφους σημείων:



[Point cloud dialog]

- **Να μην χρησιμοποιείται η δομή υπολογισμού OC δέντρου/Do not use octree calc. structure:** Από προεπιλογή, τα νέφη σημείων χρησιμοποιούν μια δομή υπολογισμού τύπου OC δέντρου για αποτελεσματική διαχείριση των σημείων. Η εισαγωγή και η αφαίρεση σημείων μπορεί να επιβραδυνθούν από αυτή τη δομή. Εάν ενεργοποιήσουμε αυτή την

επιλογή, η εισαγωγή σημείων θα είναι πιο αποτελεσματική, αλλά το νέφος σημείων δεν θα είναι πλέον συγκρουόμενο, μετρήσιμο ούτε ανιχνεύσιμο και ορισμένες άλλες λειτουργίες μπορεί επίσης να είναι περιορισμένες. Μπορούμε πάντα να εναλλάσσουμε αυτή την επιλογή σε μεταγενέστερο στάδιο ή μέσω της συνάρτησης `sim.setPointCloudOptions`.

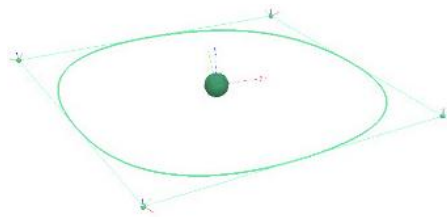
- **Μέγ. μέγεθος voxel/Max. voxel size:** Το μέγιστο μέγεθος των voxel του OC δέντρου που θα περιέχουν σημεία. Όσο μικρότερο το μέγεθος, τόσο περισσότερη μνήμη θα χρησιμοποιηθεί για τη δομή δεδομένων του OC δέντρου που βασίζεται το νέφος σημείων.
- **Μέγ. αριθμός σημείων / voxel/Max point count / voxel:** Ο μέγιστος αριθμός σημείων που μπορεί να περιέχει ένα μόνο voxel του OC δέντρου. Εάν πρέπει να αποθηκευτούν περισσότερα σημεία σε ένα τέτοιο voxel, αυτό θα χωριστεί σε 8 παιδιά voxel μέχρι να ικανοποιηθεί ο περιορισμός. Για αποτελεσματική ανίχνευση συγκρούσεων, υπολογισμούς απόστασης και ανιχνεύσεις αισθητήρων προσέγγισης, συνιστάται να διατηρούμε περίπου 10-20 σημεία αποθηκευμένα σε ένα μόνο voxel.
- **Εισαγωγή επιλεγμένων ορατών αντικειμένων/Insert selected visible objects:** Επιτρέπει την εισαγωγή σημείων βάσει των επιλεγμένων, ορατών αντικειμένων. Λαμβάνονται υπόψη μόνο τα ομοιώματα, σχήματα, OC δέντρα και νέφη σημείων. Επιλέγουμε όλα τα αντικείμενα που θέλουμε να εισαγάγουμε και επιλέγουμε τελευταίο το επιθυμητό στόχο νέφους σημείων. Τα σχήματα θα μετατραπούν προηγουμένως σε ένα προσωρινό OC δέντρο με την καθορισμένη **ανάλυση κατασκευής/build resolution**. Το στοιχείο **Εισαγωγή ανοχής/Insert tolerance** επιτρέπει την αποφυγή ύπαρξης διπλότυπων σημείων, καθορίζοντας μια ελάχιστη ανοχή απόστασης που χρησιμοποιείται για να αποφασιστεί αν το σημείο θα εισαχθεί ή όχι. Με ανοχή εισαγωγής > 0.0 η εισαγωγή σημείων θα είναι πιο αργή.
- **Αφαίρεση επιλεγμένων ορατών αντικειμένων/Subtract selected visible objects:** Επιτρέπει την αφαίρεση σημείων βάσει των επιλεγμένων, ορατών αντικειμένων. Λαμβάνονται υπόψη μόνο τα ομοιώματα, OC δέντρα και νέφη σημείων. Επιλέγουμε όλα τα αντικείμενα που θέλουμε να αφαιρέσουμε και επιλέγουμε τελευταίο το επιθυμητό στόχο νέφους σημείων. Τα ομοιώματα και τα νέφη σημείων θα χρησιμοποιούν μια τιμή ανοχής αφαίρεσης για να προσδιορίσουν τα σημεία που θα αφαιρεθούν.
- **Καθαρισμός νέφους σημείων/Clear point cloud:** Αφαιρεί όλα τα σημεία από το νέφος σημείων.
- **Τα σημεία έχουν τυχαία χρώματα/Points have random colors:** Κάθε σημείο θα έχει ένα τυχαίο χρώμα.
- **Εμφάνιση δομής OC δέντρου/Show OC tree structure:** Εμφανίζει την υποκείμενη δομή δεδομένων OC δέντρου, κυρίως για σκοπούς εντοπισμού σφαλμάτων.
- **Το χρώμα είναι εκπεμπόμενο/Color is emissive:** Τα σημεία θα εμφανίζονται με εκπεμπόμενα χρώματα.
- **Αναλογία εμφάνισης αριθμού σημείων/Point cnt display ration:** Τα νέφη σημείων που περιέχουν μεγάλο αριθμό σημείων μπορεί να επιβραδύνουν την απόδοση. Ορίζοντας μια αναλογία μικρότερη από 1, θα εμφανιστεί μικρότερος αριθμός σημείων για κάθε voxel του OC δέντρου που περιέχει σημεία. Αυτό έχει μόνο αντίκτυπο στην εμφανιζόμενη/απεικονιζόμενη εικόνα.
- **Μέγεθος σημείου/Point size:** Το μέγεθος των σημείων, σε pixel.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

- **Χρώμα για επόμενη εισαγωγή/Color for next insertion:** Επιτρέπει την επιλογή νέου χρώματος, το οποίο θα χρησιμοποιηθεί για νέες εισαγωγές σημείων.

1.4.47 Διαδρομές/Paths

Μια διαδρομή είναι ένα ψευδοαντικείμενο, που αντιπροσωπεύει μια διαδοχή σημείων με προσανατολισμό στο χώρο. Είναι ψευδοαντικείμενο, καθώς κατασκευάζεται χρησιμοποιώντας μόνο ομοιώματα και ένα προσαρμοσμένο σενάριο που περιγράφει τη λειτουργικότητα και τη συμπεριφορά του.



[Simple path containing 4 control points]

Τα αντικείμενα διαδρομής μπορούν να προστεθούν στη σκηνή με [Add → Path]. Μια διαδρομή αποτελείται από σημεία ελέγχου που ορίζουν την καμπύλη της στο χώρο. Τα σημεία ελέγχου μπορούν να μετακινηθούν, να αντιγραφούν/επικολληθούν ή να διαγραφούν. Οι βασικές ιδιότητες μιας διαδρομής είναι προσβάσιμες και ρυθμίζονται μέσω των παραμέτρων χρήστη της (υλοποιημένες μέσω μιας συνάρτησης ανάκλησης ρύθμισης χρήστη), στην ιεραρχία σκηνής:



[User parameter icon]

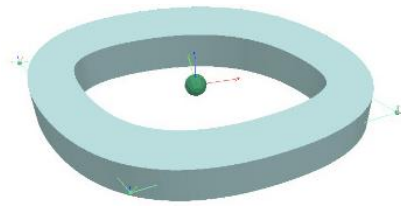
Μια διαδρομή μπορεί επίσης να δημιουργήσει αυτόματα εξωθημένα σχήματα. Αυτή η λειτουργία ενεργοποιείται μέσω των παραμέτρων χρήστη της και το προφίλ σχήματος, το χρώμα και άλλες λεπτομέρειες μπορούν να ρυθμιστούν στο προσαρμοσμένο σενάριό της, το οποίο χρησιμοποιεί τη συνάρτηση API `sim.generateShapeFromPath`.

```
--lua
```

```
function path.shaping(path, pathIsClosed, upVector)
-- following section generates a square extrusion shape:
local section = {0.02, -0.02, 0.02, 0.02, -0.02, 0.02, -0.02, -0.02, 0.02, -0.02}
local color={0.7, 0.9, 0.9}
local options = 0
if pathIsClosed then
    options = options | 4
end
local shape = sim.generateShapeFromPath(path, section, options, upVector)
sim.setShapeColor(shape, nil, sim.colorcomponent_ambient_diffuse, color)
return shape
```

```
end
```

Προσομοίωση Ρομπότ μέσω Υπολογιστή



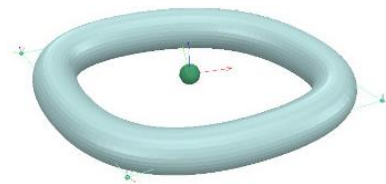
[Simple path generating an extruded square shape]

Για να δημιουργήσουμε ένα εξωθημένο κυκλικό σχήμα, χρησιμοποιούμε τον ακόλουθο κώδικα για να δημιουργήσουμε τα δεδομένα της διατομής:

```
--lua

local section = {}
local radius = 0.02
local sides = 32
local da = math.pi * 2 / sides
for i = 0, sides - 1 do
    section[2 * i + 1] = radius * math.cos(da * i)
    section[2 * i + 2] = radius * math.sin(da * i)
end
-- the section should be closed (first and last points perfect overlap):
section[#section + 1] = section[1]

section[#section + 1] = section[2]
```



[Simple path generating an extruded circular shape]

Τα δεδομένα της διαδρομής αποθηκεύονται μέσα στο αντικείμενο διαδρομής ως προσαρμοσμένα δεδομένα. Μπορούν να προσεγγιστούν με:

```
#python

# control point data (each one has x, y, z, qx, qy, qz, qw (position and quaternion)):
ctrlPts = sim.unpackDoubleTable(sim.readCustomBufferData(pathHandle, 'PATHCTRLPTS'))

# path data (each one has x, y, z, qx, qy, qz, qw (position and quaternion)):

pathData = sim.unpackDoubleTable(sim.readCustomBufferData(pathHandle, 'PATH'))
```

ή με Lua:

```
--lua

-- control point data (each one has x, y, z, qx, qy, qz, qw (position and quaternion)):
local ctrlPts = sim.unpackDoubleTable(sim.readCustomBufferData(pathHandle, 'PATHCTRLPTS'))

-- path data (each one has x, y, z, qx, qy, qz, qw (position and quaternion)):

local pathData = sim.unpackDoubleTable(sim.readCustomBufferData(pathHandle, 'PATH'))
```

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Διατίθενται διάφορες συναρτήσεις API σχετικές με τις διαδρομές. Για παράδειγμα, για να ακολουθήσει ένα αντικείμενο μια διαδρομή σε θέση και προσανατολισμό, θα μπορούσαμε να χρησιμοποιήσουμε το ακόλουθο σενάριο:

```
#python
```

```
import numpy as np
```

```
def sysCall_init():
```

```
    sim = require('sim')
    self.objectToFollowPath = sim.getObject('..')
    self.path = sim.getObject('/Path')
    pathData = sim.unpackDoubleTable(sim.readCustomBufferData(self.path, 'PATH'))
```

```
    m = np.array(pathData).reshape(len(pathData) // 7, 7)
    self.pathPositions = m[:, :3].flatten().tolist()
    self.pathQuaternions = m[:, 3:].flatten().tolist()
```

```
    self.pathLengths, self.totalLength = sim.getPathLengths(self.pathPositions, 3)
    self.velocity = 0.04 # m/s
    self.posAlongPath = 0
    self.previousSimulationTime = 0
    sim.setStepping(True)
```

```
def sysCall_thread():
```

```
    while not sim.getSimulationStopping():
        t = sim.getSimulationTime()
        self.posAlongPath += self.velocity * (t - self.previousSimulationTime)
        self.posAlongPath %= self.totalLength
        pos = sim.getPathInterpolatedConfig(self.pathPositions, self.pathLengths, self.posAlongPath)
        quat = sim.getPathInterpolatedConfig(self.pathQuaternions, self.pathLengths,
            self.posAlongPath, None, [2, 2, 2, 2])
        sim.setObjectPosition(self.objectToFollowPath, pos, self.path)
        sim.setObjectQuaternion(self.objectToFollowPath, quat, self.path)
        self.previousSimulationTime = t
```

```
    sim.step()
```

ή με Lua:

```
--lua
```

```
function sysCall_init()
```

```
    sim = require('sim')
    objectToFollowPath = sim.getObject('..')
    path = sim.getObject('/Path')
    pathData = sim.unpackDoubleTable(sim.readCustomBufferData(path, 'PATH'))
    local m = Matrix(#pathData // 7, 7, pathData)
    pathPositions = m:slice(1, 1, m:rows(), 3):data()
    pathQuaternions = m:slice(1, 4, m:rows(), 7):data()
    pathLengths, totalLength = sim.getPathLengths(pathPositions, 3)
    velocity = 0.04 -- m/s
    posAlongPath = 0
    previousSimulationTime = 0
    sim.setStepping(true)
end
```

```
function sysCall_thread()
```

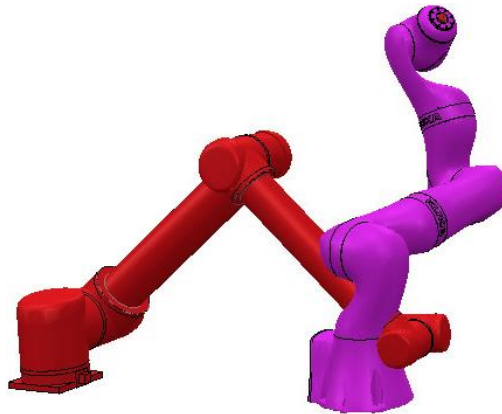
```
    while not sim.getSimulationStopping() do
        local t = sim.getSimulationTime()
```

Προσομοίωση Ρομπότ μέσω Υπολογιστή

```
posAlongPath = posAlongPath + velocity * (t - previousSimulationTime)
posAlongPath = posAlongPath % totalLength
local pos = sim.getPathInterpolatedConfig(pathPositions, pathLengths, posAlongPath)
local quat = sim.getPathInterpolatedConfig(pathQuaternions, pathLengths,
    posAlongPath, nil, {2, 2, 2, 2})
sim.setObjectPosition(objectToFollowPath, pos, path)
sim.setObjectQuaternion(objectToFollowPath, quat, path)
previousSimulationTime = t
sim.step()
end
end
```

1.4.48 Ανίχνευση σύγκρουσης/Collision detection

Το CoppeliaSim μπορεί να ανιχνεύσει συγκρούσεις μεταξύ δύο συγκρουόμενων οντοτήτων με ευέλικτο τρόπο. Ο υπολογισμός είναι ακριβής υπολογισμός παρεμβολής. Η ανίχνευση σύγκρουσης, όπως δηλώνει το όνομά της, ανιχνεύει μόνο συγκρούσεις. Ωστόσο, δεν αντιδρά άμεσα σε αυτές.

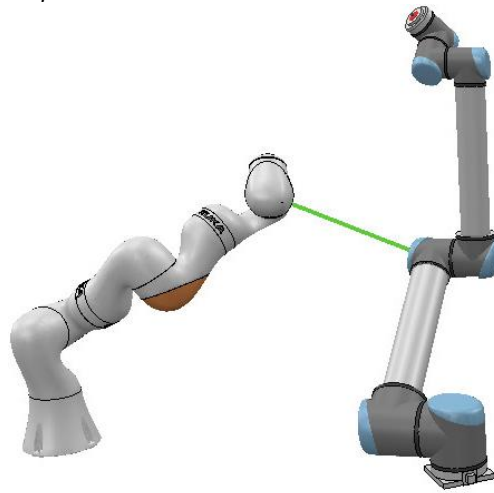


[Collision detection between two manipulators]

Χρησιμοποιώντας τη συνάρτηση API `sim.checkCollision`, μπορούμε εύκολα να ανιχνεύσουμε σύγκρουση μεταξύ οντοτήτων, για παράδειγμα ανίχνευση σύγκρουσης μεταξύ ενός ρομπότ και του περιβάλλοντός του, από ένα σενάριο προσομοίωσης, σε κάθε βήμα προσομοίωσης.

Υπολογισμός Απόστασης/Distance calculation

Το CoppeliaSim μπορεί να μετρήσει την ελάχιστη απόσταση μεταξύ δύο μετρήσιμων οντοτήτων με ευέλικτο τρόπο. Ο υπολογισμός είναι ακριβής υπολογισμός ελάχιστης απόστασης.



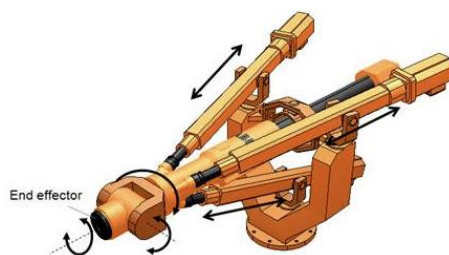
[Minimum distance calculation between two manipulators]

Χρησιμοποιώντας τη συνάρτηση API `sim.checkDistance`, μπορούμε εύκολα να υπολογίσουμε την ελάχιστη απόσταση μεταξύ οντοτήτων, για παράδειγμα μεταξύ ενός ρομπότ και του περιβάλλοντός του, από ένα σενάριο προσομοίωσης, σε κάθε βήμα προσομοίωσης

1.4.49 Κινηματική/Kinematics

Η λειτουργικότητα κινηματικής του CoppeliaSim είναι μια εξαιρετικά ικανή και ευέλικτη λύση για τη διαχείριση ενός ευρέος φάσματος μηχανισμών τόσο σε κατάσταση αντίστροφης κινηματικής (IK) όσο και σε κατάσταση άμεσης κινηματικής (FK). Η λειτουργικότητα αυτή παρέχεται μέσω των Κινηματικών Ρουτινών Coppelia, μια συλλογή συναρτήσεων C++ που μπορούν να ενσωματωθούν σε ανεξάρτητες εφαρμογές και να προσεγγιστούν μέσω του κινηματικού προσθέτου (plugin).

Το πρόβλημα IK(αντίστροφης κινηματικής) περιλαμβάνει την εύρεση των τιμών των αρθρώσεων που αντιστοιχούν σε μια συγκεκριμένη θέση και/ή προσανατολισμό ενός στοιχείου του σώματος, συνήθως του τελικού εκτελεστή. Αυτό μπορεί να θεωρηθεί ως μετασχηματισμός από τις συντεταγμένες του χώρου εργασίας στις συντεταγμένες του χώρου των αρθρώσεων. Για παράδειγμα, στην περίπτωση ενός σειριακού χειριστή, το πρόβλημα είναι να καθοριστούν οι τιμές όλων των αρθρώσεων δεδομένης της θέσης (και/ή προσανατολισμού) του τελικού εκτελεστή. Το αντίστροφο πρόβλημα, η εύρεση της θέσης και/ή του προσανατολισμού του τελικού εκτελεστή δεδομένων των τιμών των αρθρώσεων, είναι γνωστό ως πρόβλημα FK(άμεσης κινηματικής) και συχνά θεωρείται ευκολότερο από το IK, αν και αυτό μπορεί να μην ισχύει για όλες τις μηχανές, όπως δείχνει το ακόλουθο παράδειγμα:



[Non-trivial FK problem: determining the end effector pose given the joint values indicated by the arrows.]

Η κινηματική λειτουργία προσεγγίζεται μέσω του κινηματικού προσθέτου και επιτρέπει τη ρύθμιση σύνθετων κινηματικών εργασιών σε ξεχωριστό περιβάλλον IK, επιτρέποντας τον καθαρό διαχωρισμό

Προσομοίωση Ρομπότ μέσω Υπολογιστή

από άλλες πτυχές του μοντέλου προσομοίωσης (όπως η δυναμική, για παράδειγμα). Οι υπολογισμοί βασίζονται στον Ιακωβιανό και υποστηρίζονται διάφορες μέθοδοι επίλυσης, συμπεριλαμβανομένων της ψευδοαντίστροφης Ιακωβιανής, της αποσβεσμένης μέσης τετραγωνικής ψευδοαντίστροφης και των προβολών στον χώρο των μηδενικών, μεταξύ άλλων.

1.4.50 Επίλυση Εργασιών Αντίστροφης/Άμεσης Κινηματικής

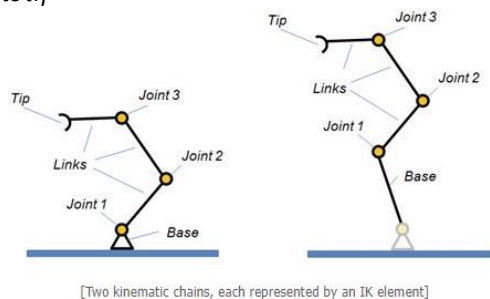
Το CoppeliaSim επιλύει εργασίες αντίστροφης και άμεσης κινηματικής χρησιμοποιώντας στοιχεία IK και ομάδες IK. Δείγματα σκηνών σχετικών με την IK και την FK μπορείτε να βρείτε στο φάκελο scenes/kinematics.

Μια εργασία IK ορίζεται από μια ομάδα IK, η οποία αποτελείται από ένα ή περισσότερα στοιχεία IK:

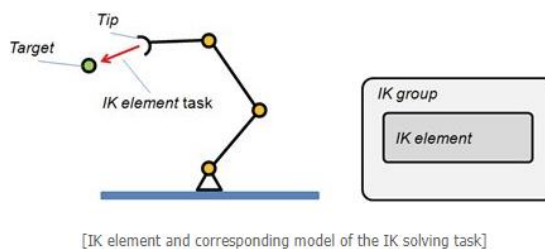
- **Στοιχείο IK/IK element:** Ένα στοιχείο IK αντιπροσωπεύει μια απλή κινηματική αλυσίδα, αποτελούμενη από τουλάχιστον μία άρθρωση. Η αλυσίδα ορίζεται από μια βάση, μια άκρη που υποδεικνύει τον τελικό εκτελεστή και έναν στόχο για την άκρη να ακολουθήσει. Το στοιχείο IK περιγράφει έναν γυμνό Ιακωβιανό πίνακα, με γραμμές που αντιπροσωπεύουν περιορισμούς και στήλες που αντιπροσωπεύουν βαθμούς ελευθερίας. Αποτελείται από τα ακόλουθα στοιχεία:
 - Μία Βάση που αντιπροσωπεύει την αρχή της κινηματικής αλυσίδας.
 - Αρκετούς συνδέσμους (οποιοδήποτε αντικείμενο εκτός από αρθρώσεις), περιλαμβάνοντας παθητικές αρθρώσεις που θεωρούνται ως άκαμπτες αρθρώσεις (αρθρώσεις με σταθερή γραμμική/γωνιακή θέση).
 - Αρκετές αρθρώσεις.
 - Μία άκρη τυπικά το τελευταίο αντικείμενο στην κινηματική αλυσίδα και συχνά ο τελικός εκτελεστής. Το αντικείμενο της άκρης πρέπει να συνδέεται με έναν στόχο.
 - Ένας στόχος αντιπροσωπεύει την επιθυμητή θέση και/ή προσανατολισμό της άκρης. Το αντικείμενο του στόχου πρέπει να συνδέεται με την άκρη.
- **Ομάδα IK/IK group:** Μια ομάδα IK περιέχει ένα ή περισσότερα στοιχεία IK. Για την επίλυση μιας απλής κινηματικής αλυσίδας χρειάζεται μια ομάδα IK που περιέχει ένα στοιχείο IK. Η ομάδα IK ορίζει τις γενικές ιδιότητες επίλυσης, όπως ο επιλυτής, ο αριθμός των επαναλήψεων κ.λπ.

Για να διευκολυνθεί η αποσφαλμάτωση της λειτουργικότητας IK, μπορούμε να ενεργοποιήσουμε την εμφάνιση ενός παραθύρου αποσφαλμάτωσης, που αντιπροσωπεύει το περιεχόμενο του κόσμου IK, μέσω [Tools > Visualize IK world].

Οι παρακάτω εικόνες απεικονίζουν δύο κινηματικές αλυσίδες όπως περιγράφονται από τα στοιχεία IK:



Στόχος ενός στοιχείου IK είναι να έχει τον στόχο να ακολουθεί την άκρη, υπολογίζοντας τις κατάλληλες τιμές αρθρώσεων για την κινηματική αλυσίδα:



1.4.51 Δυναμική/Dynamics

Το CoppeliaSim υποστηρίζει πέντε διαφορετικές μηχανές φυσικής: τη βιβλιοθήκη φυσικής Bullet, τη μηχανή Open Dynamics Engine, τη μηχανή φυσικής MuJoCo, τη μηχανή Vortex Studio και τη μηχανή Newton Dynamics. Υπάρχουν μικρές διαφορές μεταξύ των μηχανών, αλλά μπορούμε να αλλάξουμε ανά πάσα στιγμή από τη μία στην άλλη, ανάλογα με τις ανάγκες μας. Ο λόγος για την μεγάλη ποικιλία στην υποστήριξη των μηχανών φυσικής στο CoppeliaSim είναι ότι η προσομοίωση φυσικής είναι μια πολύπλοκη εργασία που μπορεί να επιτευχθεί με έμφαση σε διάφορες πτυχές της προσομοίωσης, όπως η ακρίβεια/ρεαλισμός, η ταχύτητα ή με την υποστήριξη διάφορων χαρακτηριστικών:

- **Βιβλιοθήκη φυσικής Bullet:** Μια ανοιχτού κώδικα μηχανή φυσικής που προσφέρει ανίχνευση σύγκρουσης 3D, δυναμική άκαμπτων σωμάτων και δυναμική μαλακών σωμάτων. Χρησιμοποιείται σε παιχνίδια και σε οπτικά εφέ σε ταινίες. Συχνά θεωρείται ως *μηχανή φυσικής για παιχνίδια*.



- **Open Dynamics Engine (ODE):** Μια ανοιχτού κώδικα μηχανή φυσικής με δύο κύρια συστατικά: δυναμική άκαμπτων σωμάτων και ανίχνευση σύγκρουσης. Έχει χρησιμοποιηθεί σε πολλές εφαρμογές και παιχνίδια. Συχνά θεωρείται ως *μηχανή φυσικής για παιχνίδια*.



- **MuJoCo:** Μια ανοιχτού κώδικα μηχανή φυσικής τελευταίας τεχνολογίας που παράγει υψηλής πιστότητας προσομοιώσεις φυσικής. Υποστηρίζει μια μεγάλη ποικιλία χαρακτηριστικών όπως φυσική μαλακών σωμάτων, σχοινιά, κορδόνια/ελαστικά, τένοντες, διάφορους συγκεκριμένους περιορισμούς (π.χ. περιορισμούς εξάρτησης μεταξύ μεμονωμένων αρθρώσεων) κ.λπ. Μέσω έγχυσης XML, μπορούμε να καλύψουμε επιπλέον πολλές ιδιαιτερότητες του MuJoCo.

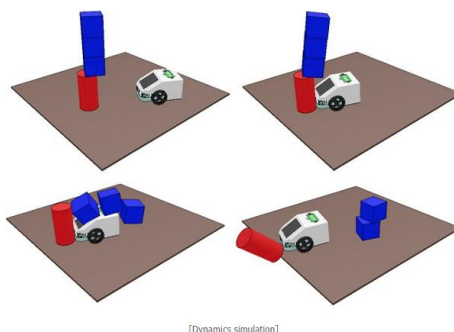
- **VortexÂ® Studio:** Μια κλειστού κώδικα εμπορική μηχανή φυσικής που παράγει υψηλής πιστότητας προσομοιώσεις φυσικής. Το Vortex προσφέρει παραμέτρους πραγματικού κόσμου (δηλαδή αντίστοιχες με φυσικές μονάδες) για μεγάλο αριθμό φυσικών ιδιοτήτων καθιστώντας αυτή τη μηχανή τόσο ρεαλιστική όσο και ακριβή. Το Vortex χρησιμοποιείται κυρίως σε εφαρμογές υψηλής απόδοσης/ακρίβειας στη βιομηχανία και την έρευνα.



- **Newton Dynamics:** Το Newton Dynamics είναι μια βιβλιοθήκη προσομοίωσης φυσικής που προσφέρει ρεαλιστική φυσική και είναι διαλειτουργική σε πολλές πλατφόρμες. Υλοποιεί έναν ντετερμινιστικό επιλύτη που δεν βασίζεται σε παραδοσιακές μεθόδους LCP ή επαναληπτικές μεθόδους, αλλά διαθέτει τη σταθερότητα και την ταχύτητα και των δύο αντίστοιχα.



Μια μηχανή φυσικής επιτρέπει την προσομοίωση των αλληλεπιδράσεων μεταξύ αντικειμένων που πλησιάζουν τις αλληλεπιδράσεις πραγματικών αντικειμένων. Επιτρέπει στα αντικείμενα να πέφτουν, να συγκρούονται, να αναπηδούν, αλλά επιτρέπει επίσης σε έναν χειριστή να αρπάξει αντικείμενα, σε έναν ιμάντα μεταφοράς να προωθήσει κομμάτια ή σε ένα όχημα να κυλά με ρεαλιστικό τρόπο πάνω σε ανώμαλο έδαφος. Τα παρακάτω σχήματα απεικονίζουν μια τέτοια προσομοίωση:



Σε αντίθεση με πολλά άλλα πακέτα λογισμικού προσομοίωσης, το CoppeliaSim δεν είναι ένας καθαρός προσομοιωτής φυσικής μηχανής. Μπορεί να θεωρηθεί ως ένας υβριδικός προσομοιωτής που συνδυάζει κινηματική και δυναμική για να επιτύχει την καλύτερη απόδοση για διάφορα σενάρια προσομοίωσης. Σήμερα, οι μηχανές φυσικής εξακολουθούν να βασίζονται σε πολλές προσεγγίσεις και είναι σχετικά ανακριβείς και αργές. Όπου είναι δυνατόν, θα πρέπει να προσπαθήσουμε να χρησιμοποιήσουμε την κινηματική (π.χ. για ρομποτικούς χειριστές) και όπου δεν είναι εφικτό να βασισόμαστε στη δυναμική μόνο (π.χ. μια αρπάγη ρομποτικού χειριστή, κινητά ρομπότ).

1.4.52 Διαφορές μεταξύ μηχανών φυσικής

Καθεμία από τις μηχανές φυσικής που υποστηρίζονται στο CoppeliaSim λειτουργεί με διαφορετικό τρόπο και κάθε μία από αυτές θα παράγει ελαφρώς ή εντελώς διαφορετικά αποτελέσματα. Τα αποτελέσματα εξαρτώνται από τους συγκεκριμένους αλγόριθμους και τις ρουτίνες υπολογισμού της μηχανής αλλά και από τις συγκεκριμένες ιδιότητες/παραμέτρους της μηχανής που χρησιμοποιούνται. Αν και θα ήταν δυνατόν να αφαιρεθούν πολύ χονδρικά ορισμένες ιδιότητες (π.χ. ο συντελεστής τριβής για ένα σώμα), το αποτέλεσμα συχνά δεν θα ήταν το αναμενόμενο: σπάνια υπάρχει μία προς

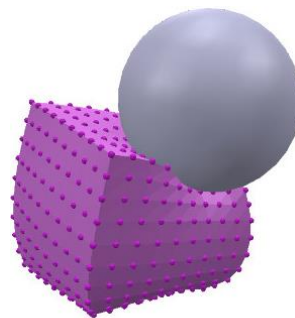
μία αντιστοιχία μεταξύ των ιδιοτήτων/παραμέτρων των μηχανών και για αυτόν το λόγο ο χρήστης μπορεί να τις προσαρμόσει βάσει της μηχανής. Οι προεπιλεγμένες ιδιότητες/παραμέτροι της μηχανής έχουν επιλεγεί για να αποδίδουν καλύτερα στις περισσότερες καταστάσεις αλλά ανάλογα με το έργο και την εφαρμογή συχνά θα χρειαστεί να τις ρυθμίσουμε για σταθερότητα ή ρεαλισμό. Αυτό γίνεται για τα σχήματα, τις αρθρώσεις και τις καθολικές ιδιότητες της μηχανής μέσω ενός επεξεργαστή JSON.

Εκτός από τις δύο κύριες διαφορές των μηχανών παραπάνω (αλγόριθμοι/ρουτίνες υπολογισμού & ιδιότητες) υπάρχουν χαρακτηριστικά (ή μειονεκτήματα) που προσφέρονται μόνο από ορισμένες μηχανές φυσικής. Σχετικά με το CoppeliaSim:

- **Ρεαλιστικές μάζες και αδράνεια:** ενώ οι μηχανές MuJoCo, Vortex και Newton επιτρέπουν τη χρήση ρεαλιστικών τιμών για μάζες και αδράνεις, οι μηχανές Bullet και ODE από την άλλη συχνά απαιτούν ισοστάθμιση μάζας και αδρανείας (δηλαδή τη διατήρηση μικρών διαφορών μάζας και αδρανείας μεταξύ των συνδεδεμένων σχημάτων).
- **Κινητήριες αρθρώσεις:** η παράμετρος *armature* είναι ίσως μία από τις πιο σημαντικές παραμέτρους της μηχανής MuJoCo και μπορεί να έχει μεγάλο αντίκτυπο στο πώς συμπεριφέρεται μια κινητήρια άρθρωση: προσθέτει μια αδράνεια ρότορα (ή ανακλώμενη αδράνεια) σε μια άρθρωση, κάνοντας την προσομοίωση πιο σταθερή και συχνά αυξάνοντας το φυσικό ρεαλισμό. Πρέπει να σημειωθεί ότι αυτό συχνά απαιτεί αυξημένη μέγιστη δύναμη ή ροπή που μπορεί να παρέχει η άρθρωση.
- **Διαδοχικές αρθρώσεις:** οι διαδοχικές αρθρώσεις που είναι δυναμικά ενεργοποιημένες υποστηρίζονται μόνο από τη μηχανή MuJoCo. Με τις άλλες μηχανές, πρέπει να εισάγονται βοηθητικές μάζες μεταξύ διαδοχικών αρθρώσεων.
- **Τροποποίηση περιεχομένου δυναμικά:** η τροποποίηση του περιεχομένου μιας σκηνής κατά τη διάρκεια της προσομοίωσης, όπως η προσθήκη/αφαίρεση ενός αντικειμένου, η αλλαγή της ιεραρχίας της σκηνής ή η δυναμική επαναφορά ενός αντικειμένου, γίνεται χωρίς προβλήματα με όλες τις μηχανές, εκτός από τη μηχανή MuJoCo που απαιτεί πλήρη επανεκκίνηση κάθε φορά: αυτή η λειτουργία είναι σχετικά αργή, αλλά θα διατηρήσει την κατάσταση της προσομοίωσης εκείνη τη στιγμή (π.χ. θέσεις και ταχύτητες αντικειμένων). Μια κοινή και συχνή κατάσταση όπου αυτό συμβαίνει είναι οι εφαρμογές ρομπότ με τροχούς που κινούνται σε όλες τις κατευθύνσεις, όπου απαιτείται δυναμική επαναφορά αντικειμένου σε κάθε βήμα προσομοίωσης.
- **Μη κυρτά σχήματα:** τα μη κυρτά πλέγματα είναι δύσκολα και αργά στην προσομοίωση γενικά και συχνά ασταθή. Για αυτόν τον λόγο, πρέπει να αποφεύγονται με κάθε κόστος. Ορισμένες μηχανές ωστόσο τα υποστηρίζουν, ενώ άλλες θα χρησιμοποιούν την κυρτή αναπαράστασή τους για την προσομοίωση (οι μηχανές Newton και MuJoCo).
- **Στατικά σχήματα:** τα στατικά σχήματα που είναι προγραμματικά μετακινούμενα θα αλληλεπιδρούν με τα δυναμικά σχήματα. Αυτή η αλληλεπίδραση περιλαμβάνει την αντίδραση σύγκρουσης και τη μετάδοση τριβής. Η μηχανή MuJoCo ωστόσο δεν θα μεταδίδει δυνάμεις τριβής από στατικά σχήματα όταν μετακινούνται, εκτός αν το σχήμα σημειωθεί ως κινηματικό: `sim.setObjectIntParam(shapeHandle, sim.shapeintparam_kinematic, 1)`. Σε αυτή την περίπτωση, το στατικό σχήμα αναπαρίσταται στο MuJoCo ως σώμα *mocap* συνδεδεμένο με ένα δυναμικό σώμα μέσω ενός περιορισμού συγκόλλησης.
- **Επιφανειακή ταχύτητα:** οι περισσότερες μηχανές υποστηρίζουν επιφανειακές ταχύτητες, οι οποίες χρησιμοποιούνται συχνά για την προσομοίωση ενός απλού μάντα μεταφοράς μοντελοποιημένου ως κυλινδρικό σχήμα με επιφανειακή ταχύτητα. Αυτό δεν είναι δυνατό με

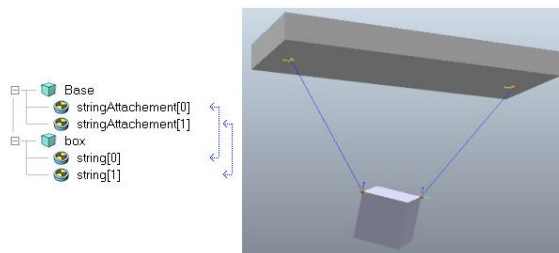
τη μηχανή MuJoCo, η οποία υποστηρίζει μόνο ιμάντες μεταφοράς, όπου κάθε στοιχείο του ιμάντα είναι στατικό και κινηματικό και προγραμματικά μετακινούμενο.

- **Μαλακά σώματα:** μόνο η μηχανή MuJoCo υποστηρίζει μαλακά σώματα, συμπεριλαμβανομένων των ρούχων και των σχοινιών. Αυτά δημιουργούνται στο CoppeliaSim προσθέτοντας σύνθετα στοιχεία μέσω έγχυσης XML. Πρόσθετοι περιορισμοί και συμπεριφορές μπορούν επίσης να επιτευχθούν μέσω έγχυσης XML, γνωρίζοντας ότι τα παραγόμενα αρχεία XML έχουν συνεπή ονομασία αντικειμένων, βασισμένα στα ψευδώνυμα και τις λαβές αντικειμένων του CoppeliaSim: `sim.getObjectAlias(objectHandle, 4)`. Τα παραγόμενα αρχεία MuJoCo μπορούν να βρεθούν και να ελεγχθούν στο `sim.getStringParam(sim.stringparam_mujocodir)`.



[Soft body simulation]

- **Καλώδια/σχοινιά και ελαστικά:** μόνο η μηχανή MuJoCo υποστηρίζει περιορισμούς τύπου τένοντα μεταξύ δύο σχημάτων. Αυτοί υλοποιούνται μέσω συνδεδεμένων ομοιωμάτων, όπου ο τύπος σύνδεσης είναι Dynamics, tendon constraint. Αυτοί οι τένοντες μπορούν να κινηθούν μέσω ενός μεσολαβητικού αρθρωτού άξονα. Άλλες μηχανές θα πρέπει να ανατρέξουν στη μοντελοποίηση ενός σχοινιού μέσω δύο σφαιρικών και μίας αρθρωτής άρθρωσης, με κατάλληλες βοηθητικές μάζες μεταξύ τους.



[Strings/elastics]

1.4.53 Ανάπτυξη κώδικα μέσα και γύρω από το CoppeliaSim

Το CoppeliaSim είναι ένας υψηλά προσαρμόσιμος προσομοιωτής: κάθε πτυχή της προσομοίωσης μπορεί να προσαρμοστεί. Αυτό επιτυγχάνεται μέσω μιας εκτενούς διεπαφής προγραμματισμού εφαρμογών (API). Υποστηρίζονται 7+ διαφορετικές προσεγγίσεις προγραμματισμού, καθεμία με συγκεκριμένα πλεονεκτήματα (και φυσικά μειονεκτήματα) σε σύγκριση με τις άλλες, αλλά όλες είναι συμβατές μεταξύ τους (δηλαδή μπορούν να χρησιμοποιούνται ταυτόχρονα ή ακόμα και συνδυαστικά). Ο κώδικας ελέγχου ενός μοντέλου, μιας σκηνής ή ακόμα και του ίδιου του προσομοιωτή μπορεί να τοποθετηθεί στις εξής τοποθεσίες:

Προσομοίωση Ρομπότ μέσω Υπολογιστή

- ένα ενσωματωμένο σενάριο (δηλαδή προσαρμογή μιας προσομοίωσης μέσω σεναρίων): αυτή η μέθοδος, η οποία περιλαμβάνει τη συγγραφή σεναρίων Lua ή Python, είναι πολύ εύκολη και ευέλικτη. Αυτή η μέθοδος επιτρέπει την προσαρμογή μιας συγκεκριμένης προσομοίωσης, μιας σκηνης προσομοίωσης και μέχρι ενός βαθμού του ίδιου του προσομοιωτή. Αυτή είναι η πιο εύκολη και συχνά χρησιμοποιούμενη προγραμματιστική προσέγγιση.
- ένα πρόσθετο ή το sandbox: αυτή η μέθοδος η οποία περιλαμβάνει τη συγγραφή σεναρίων Lua ή Python επιτρέπει τη γρήγορη προσαρμογή του ίδιου του προσομοιωτή. Τα πρόσθετα (ή το sandbox) μπορούν να ξεκινήσουν αυτόματα και να τρέχουν στο παρασκήνιο ή μπορούν να καλούνται ως λειτουργίες (π.χ. βολικό όταν γράφονται εισαγωγείς/εξαγωγείς). Τα πρόσθετα δεν πρέπει να είναι συγκεκριμένα για μια συγκεκριμένη προσομοίωση ή μοντέλο, αλλά θα πρέπει να προσφέρουν μια πιο γενική, λειτουργικότητα που σχετίζεται με τον προσομοιωτή.
- ένα πρόσθετο (δηλαδή προσαρμογή του προσομοιωτή και/ή μιας προσομοίωσης μέσω ενός προσθέτου): αυτή η μέθοδος βασίζεται βασικά στη συγγραφή ενός προσθέτου για το CoppeliaSim. Συχνά, τα πρόσθετα χρησιμοποιούνται μόνο για να παρέχουν μια προσομοίωση με προσαρμοσμένες εντολές API και γι' αυτό χρησιμοποιούνται σε συνδυασμό με τις πρώτες μεθόδους. Άλλες φορές, τα πρόσθετα χρησιμοποιούνται για να παρέχουν στο CoppeliaSim μια ειδική λειτουργικότητα που απαιτεί είτε δυνατότητα γρήγορου υπολογισμού (τα σενάρια είναι συνήθως πιο αργά από τις μεταγλωττισμένες γλώσσες), είτε μια συγκεκριμένη διασύνδεση με συσκευή υλικού (π.χ. ένα πραγματικό ρομπότ) ή μια ιδιαίτερη διασύνδεση επικοινωνίας.
- η εφαρμογή πελάτη (δηλαδή η προσαρμογή του προσομοιωτή και/ή μιας προσομοίωσης μέσω μιας προσαρμοσμένης εφαρμογής πελάτη): αυτό συνίσταται βασικά στη συγγραφή μιας προσαρμοσμένης κύριας εφαρμογής για τη βιβλιοθήκη CoppeliaSim. Αυτό επιτρέπει, για παράδειγμα, να ξεκινήσουμε/να κλείσουμε το CoppeliaSim αρκετές φορές σειριακά ή με συγκεκριμένο τρόπο.
- ένας απομακρυσμένος API πελάτης (δηλαδή η προσαρμογή του προσομοιωτή και/ή μιας προσομοίωσης μέσω μιας εφαρμογής απομακρυσμένου API πελάτη): αυτή η μέθοδος επιτρέπει σε μια εξωτερική εφαρμογή (π.χ. που βρίσκεται σε ένα ρομπότ, σε έναν άλλο υπολογιστή, κλπ.) να συνδεθεί με το CoppeliaSim με πολύ εύκολο τρόπο, χρησιμοποιώντας εντολές απομακρυσμένου API.
- ένας κόμβος ROS (δηλαδή η προσαρμογή του προσομοιωτή και/ή μιας προσομοίωσης μέσω ενός κόμβου ROS): αυτή η μέθοδος επιτρέπει σε μια εξωτερική εφαρμογή (π.χ. που βρίσκεται σε ένα ρομπότ, σε έναν άλλο υπολογιστή, κλπ.) να συνδεθεί με το CoppeliaSim μέσω του ROS, του Robot Operating System.
- ένας κόμβος που μιλά TCP/IP, ZeroMQ, κλπ.: αυτή η μέθοδος επιτρέπει σε μια εξωτερική εφαρμογή (π.χ. που βρίσκεται σε ένα ρομπότ, σε έναν άλλο υπολογιστή, κλπ.) να συνδεθεί με το CoppeliaSim μέσω διαφόρων μέσων επικοινωνίας.

Ο ακόλουθος πίνακας περιγράφει λεπτομερώς τα αντίστοιχα πλεονεκτήματα και μειονεκτήματα κάθε μεθόδου:

Προσομοίωση Ρομπότ μέσω Υπολογιστή

	Embedded script	Add-on / sandbox script	Plugin	Client application	Remote API client	ROS / ROS2 node	ZeroMQ node
Control entity is external (i.e. can be located on a robot, different machine, etc.)	No	No	No	No	Yes	Yes	Yes
Supported programming language	Lua, Python	Lua, Python	C/C++	C/C++, Python	C/C++, Python, Java, JavaScript, Matlab, Octave	Any ¹	Any
Code execution speed	Relatively fast ²	Relatively fast ²	Fast	Fast	Depends on programming language	Depends on programming language	Depends on programming language
Communication lag	None ³	None ³	None	None	Yes	Yes	Yes
Communication channel	Python: ZeroMQ ⁴	Python: ZeroMQ ⁴	None	None	ZeroMQ or WebSockets	ROS / ROS2	ZeroMQ
Control entity can be fully contained in a scene or model, and is highly portable	Yes	No	No	No	No	No	No
Stepped operation ⁴	Yes, inherent	Yes, inherent	Yes, inherent	Yes, inherent	Yes	Yes	Yes
Non-stepped operation ⁴	Yes, via threads	Yes, via threads	No (threads available, but API access forbidden)	No (threads available, but API access forbidden)	Yes	Yes	Yes

¹ Depends on ROS / ROS2 bindings

² Depends on the programming language, but the execution of API functions is very fast

³ Lua scripts are executed in CoppeliaSim's main thread, Python scripts are executed in separate processes

⁴ Stepped as in *synchronized* with each simulation step

1.4.54 Προσομοίωση

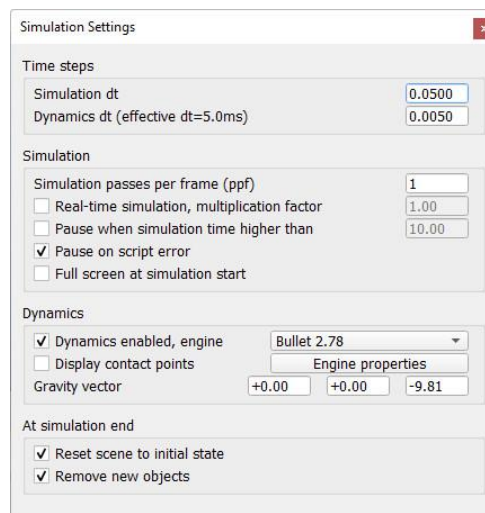
Μια προσομοίωση στο CoppeliaSim μπορεί να ξεκινήσει, να τεθεί σε παύση και να διακοπεί με την [Simulation → Start/Pause/Stop simulation] ή μέσω των σχετικών κουμπιών της γραμμής εργαλείων:



[Simulation start/pause/stop toolbar buttons]

1.4.55 Παράθυρο διαλόγου προσομοίωσης

Το παράθυρο διαλόγου προσομοίωσης μπορεί να προσπελαστεί με [Simulation → Simulation settings] ή κάνοντας κλικ στο ακόλουθο κουμπί εργαλειοθήκης:



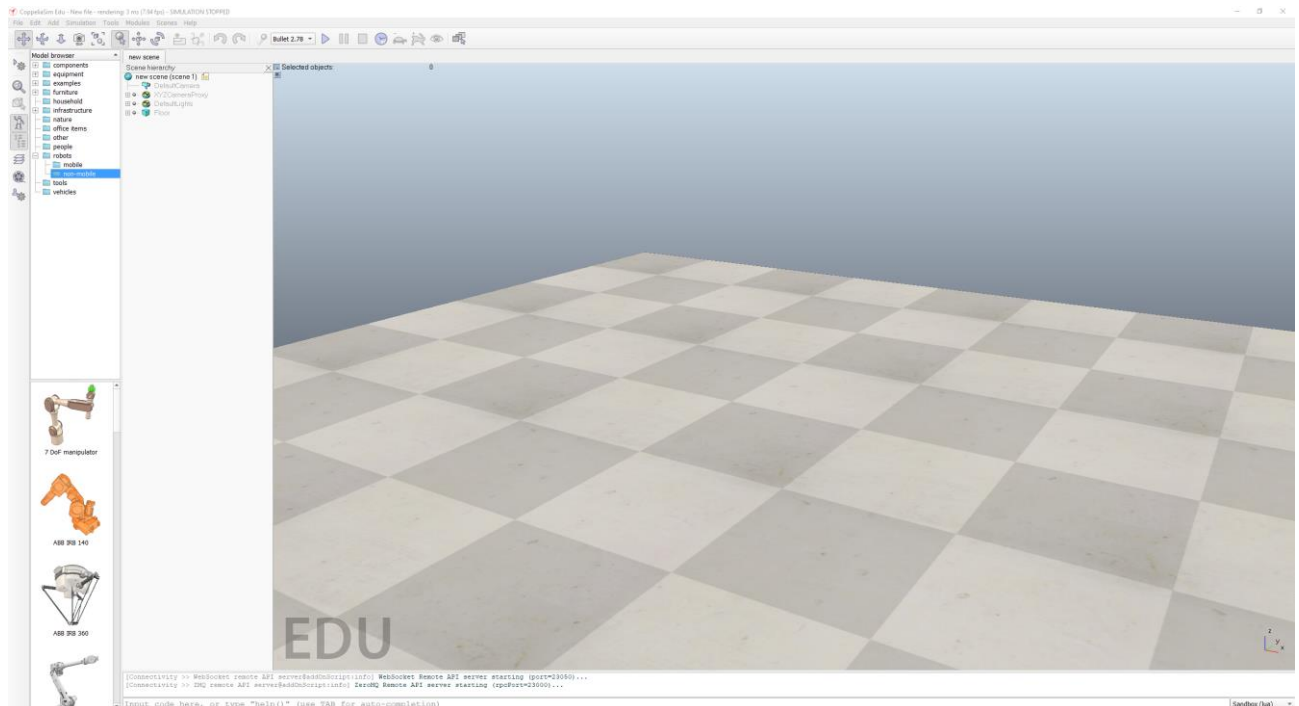
- **Simulation dt:** το βήμα χρόνου της προσομοίωσης. Κάθε φορά που εκτελείται το κύριο σενάριο ο χρόνος προσομοίωσης αυξάνεται κατά το dt προσομοίωσης. Συνιστάται ιδιαίτερα να διατηρείται ένα προεπιλεγμένο dt προσομοίωσης των 50ms. Πρέπει να λάβουμε υπόψιν ότι το dt προσομοίωσης δεν μπορεί να είναι μικρότερο από το dt δυναμικής.
- **Dynamics dt:** το βήμα χρόνου που χρησιμοποιεί η μηχανή φυσικής. Η χρήση μεγάλων βημάτων χρόνου οδηγεί σε ταχύτερες αλλά λιγότερο ακριβείς προσομοιώσεις. Αντίθετα, τα μικρά βήματα χρόνου θα οδηγήσουν σε πιο ακριβείς προσομοιώσεις, αλλά θα απαιτήσουν περισσότερο χρόνο. Συνιστάται ιδιαίτερα να διατηρείται ένα προεπιλεγμένο dt δυναμικής των 5ms. Σε συνδυασμό με ένα προεπιλεγμένο dt προσομοίωσης των 50ms, αυτό οδηγεί σε 10 βήματα υπολογισμού δυναμικής ανά πέρασμα προσομοίωσης.
- **Περάσματα προσομοίωσης ανά καρτέ/ Simulation passes per frame (ppf):** ο αριθμός των περασμάτων προσομοίωσης για ένα πέρασμα απόδοσης. Μία τιμή του 10 σημαίνει ότι το κύριο σενάριο εκτελείται 10 φορές (10 βήματα προσομοίωσης) πριν ανανεωθεί η οθόνη.
- **Προσομοίωση σε πραγματικό χρόνο, συντελεστής πολλαπλασιασμού/Real-time simulation, multiplication factor:** αν επιλεγεί, τότε ο χρόνος προσομοίωσης θα περιορίζεται στο πραγματικό χρόνο (όταν είναι εφικτό). Ένας συντελεστής πολλαπλασιασμού X θα περιορίζει τον χρόνο προσομοίωσης στο 10* του πραγματικού χρόνου.
- **Παύση όταν ο χρόνος προσομοίωσης υπερβαίνει/Pause when simulation time is higher than:** επιτρέπει τον καθορισμό ενός χρόνου προσομοίωσης στον οποίο η προσομοίωση θα παύσει (π.χ. για να αναλυθούν κάποια αποτελέσματα σε συγκεκριμένο χρόνο προσομοίωσης).
- **Παύση σε σφάλμα σεναρίου/Pause on script error:** αν ενεργοποιηθεί, τότε η προσομοίωση θα παύσει όταν παρουσιαστεί σφάλμα σε ενσωματωμένο σενάριο.
- **Πλήρης οθόνη κατά την έναρξη της προσομοίωσης/Full screen at simulation start:** αν ενεργοποιηθεί, τότε η προσομοίωση ξεκινά σε πλήρη οθόνη. Σε πλήρη οθόνη οι διάλογοι και τα μηνύματα δεν θα εμφανίζονται ή δεν θα είναι ορατά και μόνο το αριστερό κουμπί του ποντικιού θα είναι ενεργό. Για αυτόν τον λόγο, αυτή η λειτουργία συνιστάται μόνο όταν η σκηνή είναι κατάλληλα διαμορφωμένη και τελική. Η πλήρης οθόνη μπορεί να εγκαταλειφθεί με το πλήκτρο esc και να εναλλάσσεται μέσω της bool παραμέτρου sim.booparam_fullscreen κατά την προσομοίωση. Σε Linux και MacOS, η λειτουργία πλήρους οθόνης μπορεί να υποστηρίζεται μόνο μερικώς και η επιστροφή σε κανονική λειτουργία μπορεί να αποτύχει σε ορισμένα συστήματα.
- **Ενεργοποίηση δυναμικής/Dynamics enabled:** ενεργοποιεί ή απενεργοποιεί τη μηχανή φυσικής. Επιτρέπει την επιλογή μιας μηχανής φυσικής. Κάθε μηχανή έχει συγκεκριμένες δυνάμεις και αδυναμίες. Μία προσομοίωση που τρέχει καλά με μία μηχανή δεν σημαίνει πάντα ότι θα τρέχει καλά και με άλλες μηχανές.
- **Εμφάνιση σημείων επαφής/Display contact points:** εμφανίζει ή αποκρύπτει σημεία επαφής που υπολογίζονται από τη μηχανή φυσικής.
- **Ιδιότητες μηχανής/Engine properties:** κάθε μηχανή έχει συγκεκριμένες παραμέτρους που μπορούν να ρυθμιστούν παγκοσμίως. Φροντίζουμε να διαβάσουμε τις αντίστοιχες τεκμηριώσεις των μηχανών για την επίτευξη των βέλτιστων ρυθμίσεων παραμέτρων.
- **Διάνυσμα βαρύτητας/Gravity vector:** το μέγεθος και η κατεύθυνση μιας σταθερής δύναμης που εφαρμόζεται σε όλα τα μη στατικά σχήματα (π.χ. βαρύτητα).

Προσομοίωση Ρομπότ μέσω Υπολογιστή

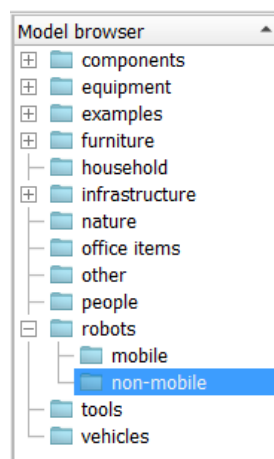
- **Επαναφορά σκηνής στην αρχική κατάσταση/Reset scene to initial state:** όταν είναι επιλεγμένο, όλα τα αντικείμενα θα έχουν τις περισσότερες παραμέτρους και ιδιότητες επαναρυθμισμένες σε αυτό που ήταν κατά την έναρξη της προσομοίωσης.
- **Αφαίρεση νέων αντικειμένων/Remove new objects:** όταν είναι επιλεγμένο, τα αντικείμενα σκηνής που προστίθενται κατά τη διάρκεια μιας προσομοίωσης θα αφαιρεθούν στο τέλος της προσομοίωσης.

1.5 Δημιουργία Ρομποτικού Βραχίονα στο CoppeliaSim

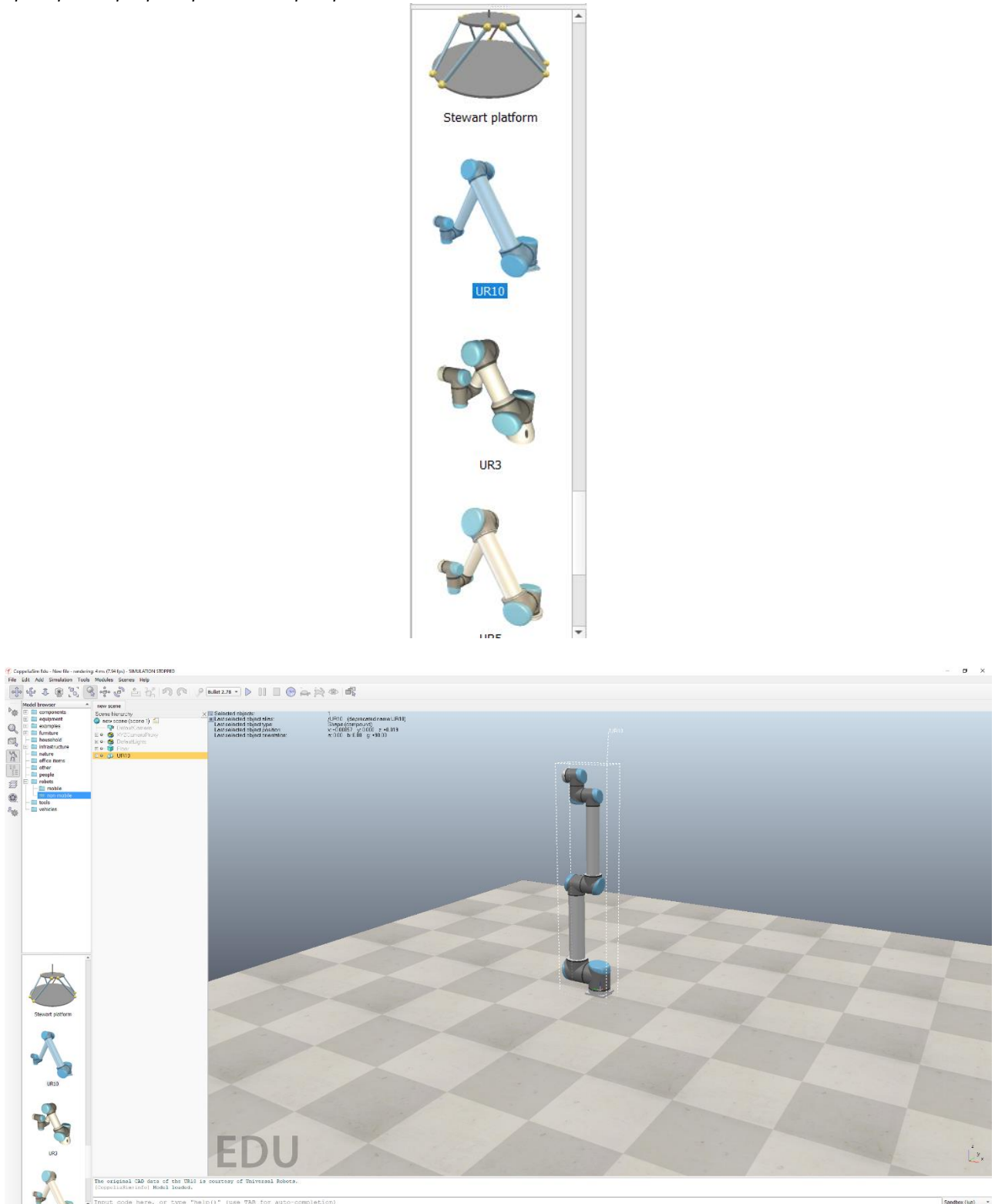
Για τη δημιουργία ενός ρομποτικού βραχίονα θα εκκινήσουμε το CoppeliaSim. Μας εμφανίζει μια κενή σκηνή.



Επιλέγουμε από το Model browser το robots → non-mobile και διαλέγουμε ποιο θέλουμε να χρησιμοποιήσουμε. Για τη δική μας προσομοίωση θα χρησιμοποιήσουμε το UR10. Το επιλέγουμε και το σύρουμε στο Scene hierarchy.

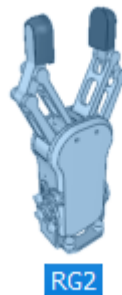
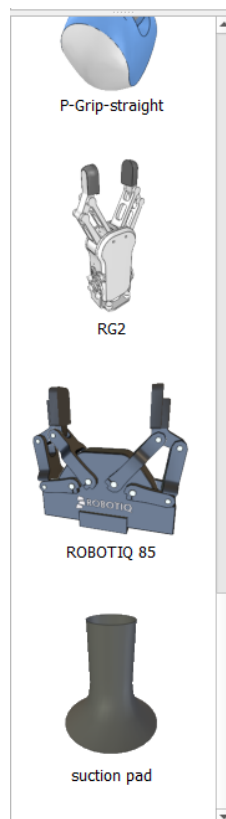
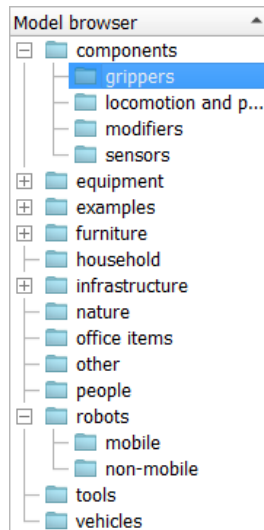


Προσομοίωση Ρομπότ μέσω Υπολογιστή



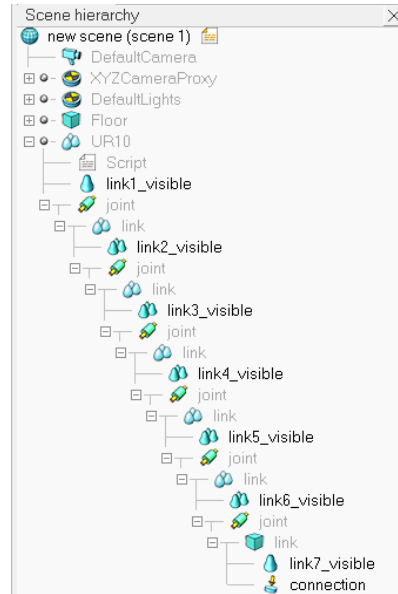
Αφού το εισάγουμε επόμενο βήμα είναι να του προσθέσουμε μια λαβή (Gripper). Πηγαίνουμε στο Model browser και επιλέγουμε το components → grippers. Θα χρησιμοποιήσουμε τη λαβή RG2.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

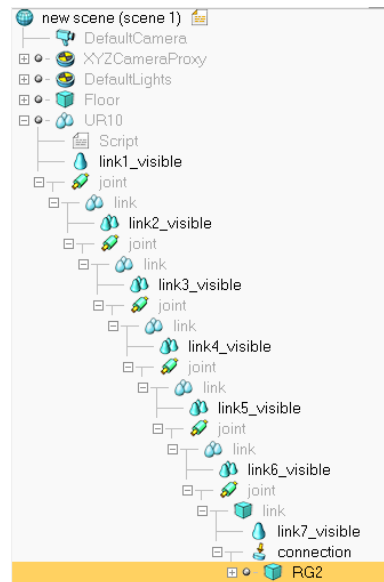


Αναπτύσσουμε το UR10 ώστε να δούμε από τι αποτελείται.

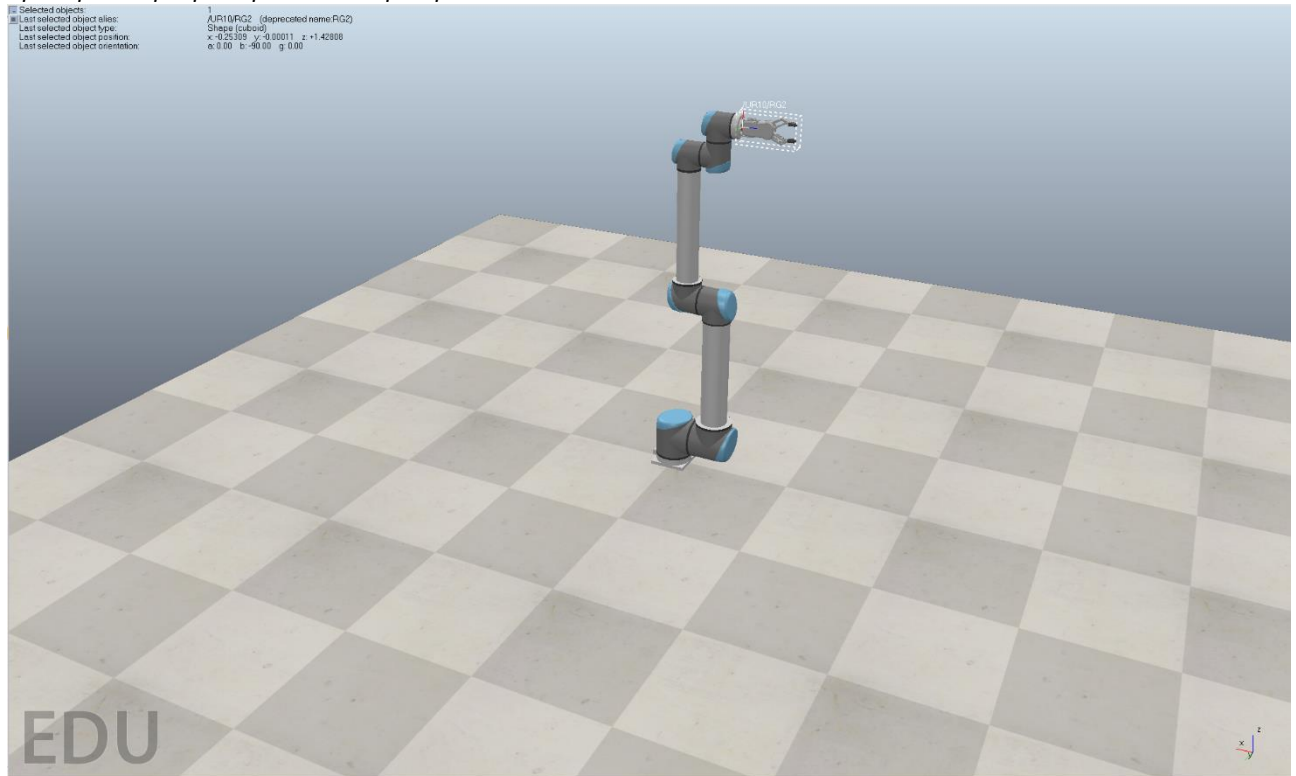
Προσομοίωση Ρομπότ μέσω Υπολογιστή



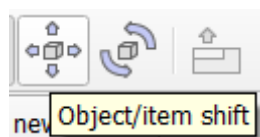
Θέλουμε η λαβή να πάει στο τελευταίο σημείο του ρομπότ μας ώστε να βρίσκεται στην άκρη του. Θα σύρουμε τη λαβή μας και θα την αφήσουμε πάνω στο connection.



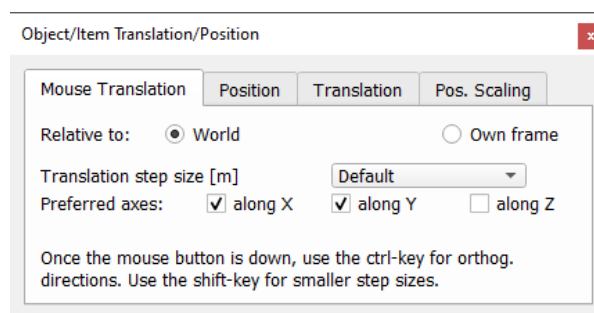
Προσομοίωση Ρομπότ μέσω Υπολογιστή



Επιλέγουμε το RG2 και πατάμε το Object/item shift και έπειτα το Object/item rotate ώστε να ευθυγραμμίσουμε σωστά τη λαβή μας.



Στο παράθυρο που μας εμφανίζεται πηγαίνουμε στη καρτέλα Position και επιλέγουμε Relative to: Parent frame έτσι ώστε να το ευθυγραμμίσουμε με το connection. Για να το κάνουμε αυτό εισάγουμε το 0 στις συντεταγμένες X, Y, Z.



Προσομοίωση Ρομπότ μέσω Υπολογιστή

Object/Item Translation/Position

Mouse Translation Position Translation Pos. Scaling

Relative to: World Parent frame

X-coord. [m] -0.25309 Apply X to sel.

Y-coord. [m] -0.00011 Apply Y to sel.

Z-coord. [m] +1.42808 Apply Z to sel.

Apply to selection

Object/Item Translation/Position

Mouse Translation Position Translation Pos. Scaling

Relative to: World Parent frame

X-coord. [m] +0.00008 Apply X to sel.

Y-coord. [m] -0.00011 Apply Y to sel.

Z-coord. [m] +0.0045 Apply Z to sel.

Apply to selection

Object/Item Translation/Position

Mouse Translation Position Translation Pos. Scaling

Relative to: World Parent frame

X-coord. [m] 0.000 Apply X to sel.

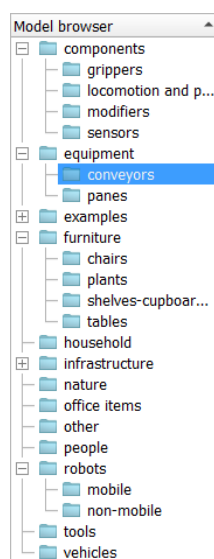
Y-coord. [m] 0.000 Apply Y to sel.

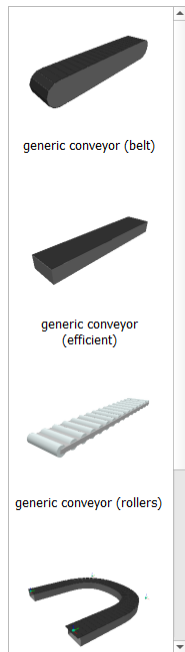
Z-coord. [m] 0.000 Apply Z to sel.

Apply to selection

Θα προσθέσουμε μια ταινία μεταφοράς από την οποία ο ρομποτικός βραχίονας μας θα πιάνει ένα κύβο και θα το μεταφέρει σε ένα τραπέζι όπου και θα το αφήνει.

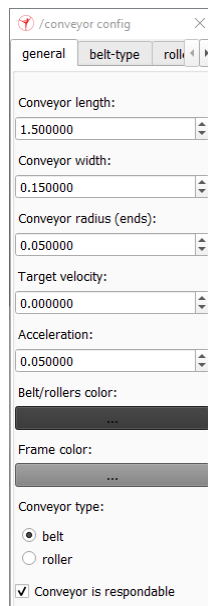
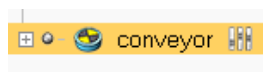
Για να το κάνουμε αυτό θα πάμε στο Model browser → equipment → conveyors και θα επιλέξουμε και θα σύρουμε στη σκηνή μας το generic conveyor (belt).

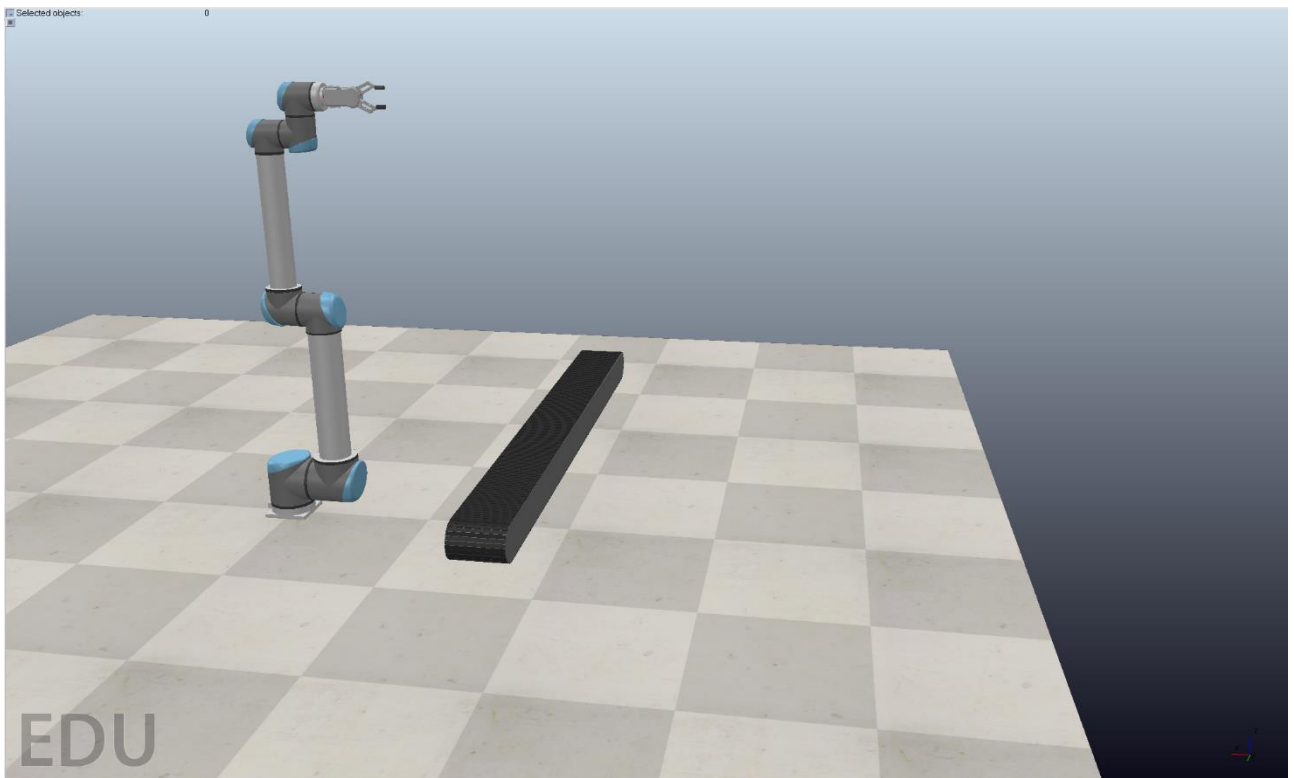
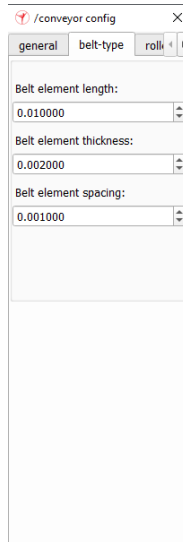




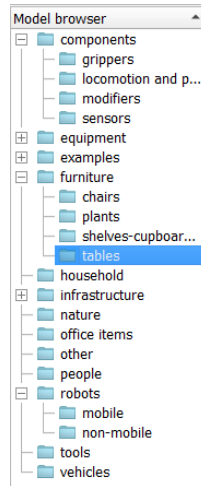
Με τη βοήθεια του Object/item shift και Object/item rotate πηγαίνουμε τη ταινία μεταφοράς στο σημείο που θέλουμε και στο μέγεθος που θέλουμε.

Πατώντας στις 3 γραμμές δίπλα από το conveyor μας εμφανίζεται το παράθυρο ώστε να το διαμορφώσουμε.





Για να προσθέσουμε το τραπέζι θα πάμε στο Model browser → furniture → tables και θα επιλέξουμε το customizable table. Το σύρουμε στη σκηνή μας. Πατάμε διπλό κλικ πάνω στο script για να του αλλάξουμε τη γεωμετρία και έπειτα με το Object/item shift θα το μετακινήσουμε στο σημείο που θέλουμε (κάπου που να φτάνει και το ρομπότ μας).

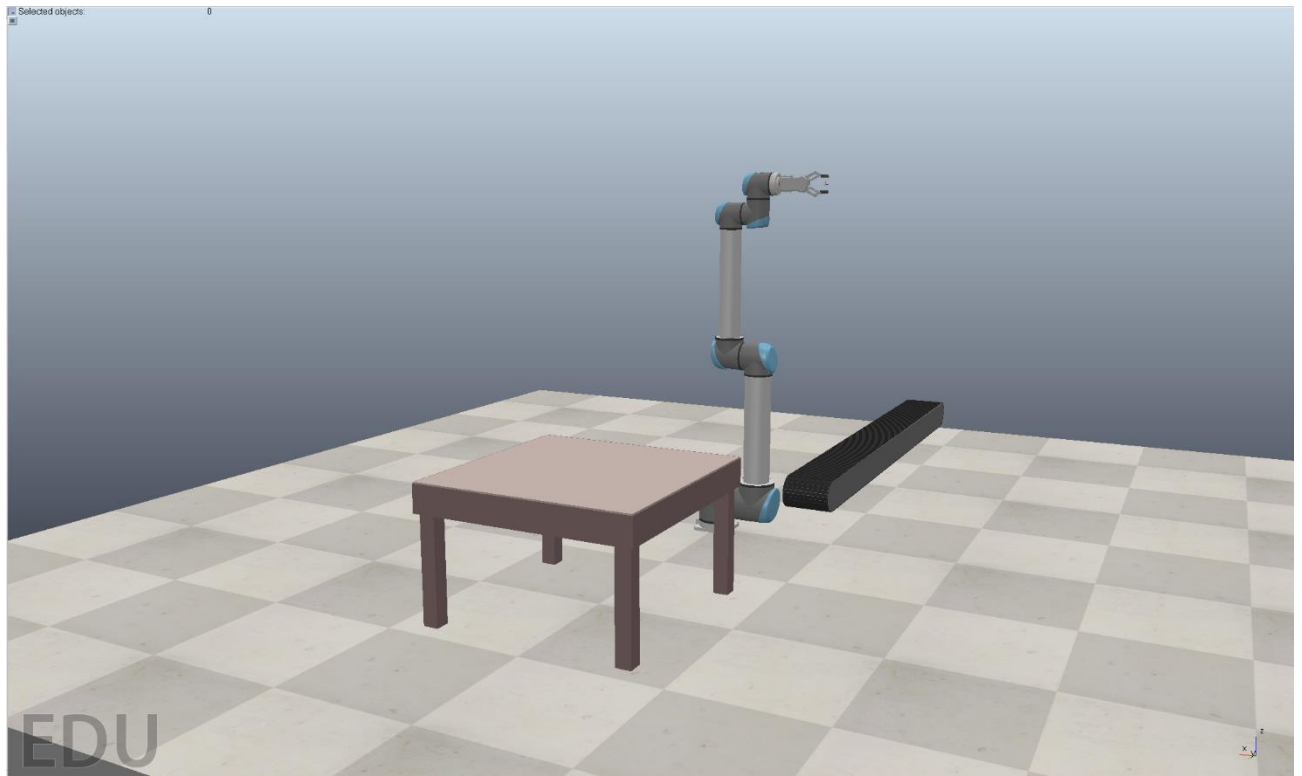


Προσομοίωση Ρομπότ μέσω Υπολογιστή

```
Customization script "/customizableTable/Script"
lua
1 |sim=require'sim'
2
3 |function sysCall_init()
4 |   local tableConfig={}
5 |   tableConfig.length=0.8
6 |   tableConfig.width=0.8
7 |   tableConfig.height=0.4827
8 |   tableConfig.color1={0.8,0.73,0.7}
9 |   tableConfig.color2={0.57,0.47,0.47}
10
11 |   local model=sim.getObject('.')
12 |   local xJoint=sim.getObject('./xJoint1')
13 |   local yJoint=sim.getObject('./yJoint1')
14 |   local tableTop=sim.getObject('./top')
15 |   local tableP=sim.getObject('./table')
16
17 |   local p=sim.getObjectPosition(model)
18 |   sim.setObjectPosition(model,{p[1],p[2],tableConfig.height})
19 |   sim.setShapeBB(tableP,{tableConfig.length,tableConfig.width,0.1})
20 |   sim.setShapeColor(tableP,'',sim.colorcomponent_ambient_diffuse,tableConfig.color2)
21 |   sim.setObjectPosition(tableP,{0,0,-0.05},model)
22 |   sim.setShapeBB(tableTop,{tableConfig.length-0.005,tableConfig.width-0.005,0.1})
23 |   sim.setShapeColor(tableTop,'',sim.colorcomponent_ambient_diffuse,tableConfig.color1)
24 |   sim.setObjectPosition(tableTop,{0,0,-0.045},model)
25 |   for i=1,4 do
26 |     local foot=sim.getObject('./foot'.i)
27 |     sim.setShapeBB(foot,{0.055,0.055,tableConfig.height-0.05})
28 |     sim.setShapeColor(foot,'',sim.colorcomponent_ambient_diffuse,tableConfig.color2)
29 |     p=sim.getObjectPosition(foot,model)
30 |     sim.setObjectPosition(foot,{p[1],p[2],-0.05-(tableConfig.height-0.05)/2},model)
31 |   end
32 |   sim.setJointPosition(xJoint,tableConfig.length/2-0.07)
33 |   sim.setJointPosition(yJoint,tableConfig.width/2-0.07)
34 | end
35
```

Αλλάζουμε το length το width και το height και το κινούμε στο χώρο.

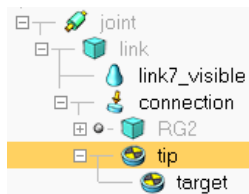
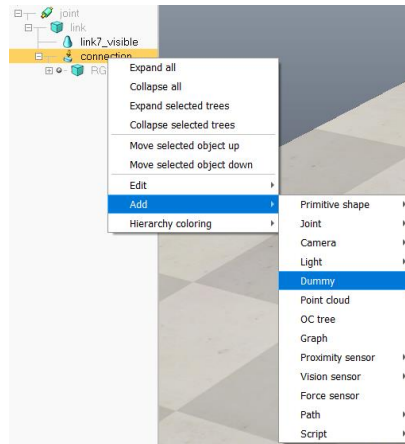
Η σκηνή μας τελικά θα είναι έτσι:



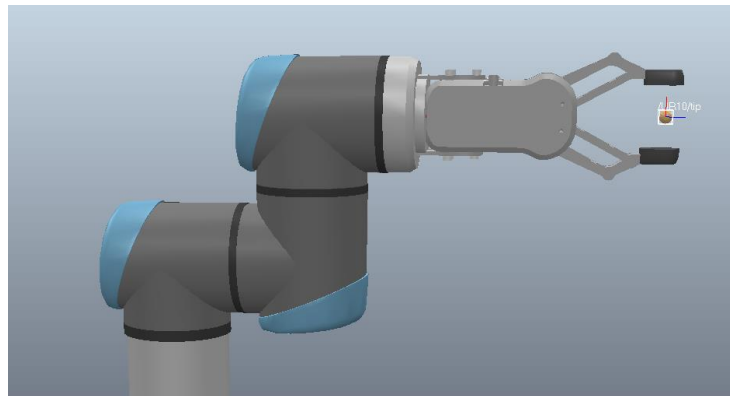
Για να κινήσουμε το ρομπότ μας θα χρειαστεί να χρησιμοποιήσουμε κώδικα. Μέσα σε αυτόν θα υπάρχουν ορισμένες συντεταγμένες. Αυτές θα είναι οι γωνίες στις οποίες οι αρθρώσεις του βραχίονα θα πρέπει να γυρίσουν προκειμένου να πιάσουν το κύβο από τη ταινία μεταφοράς και μετέπειτα να τον αφήσουν πάνω στο τραπέζι. Για να βρούμε τις γωνίες θα χρησιμοποιήσουμε Αντίστροφη Κινηματική (Inverse Kinematics). Για να το κάνουμε αυτό ξεκινάμε με το να επιλέξουμε το

Προσομοίωση Ρομπότ μέσω Υπολογιστή

connection και να κάνουμε δεξί κλικ → add → dummy. Έπειτα θα το ονομάσουμε tip. Αυτό θα είναι η άκρη του βραχίονα μας. Θα επιλέξουμε τώρα το tip και θα κάνουμε το ίδιο ονομάζοντας το νέο dummy ως target. Αυτός θα είναι ο στόχος που θα ακολουθάει η άκρη του βραχίονα μας.

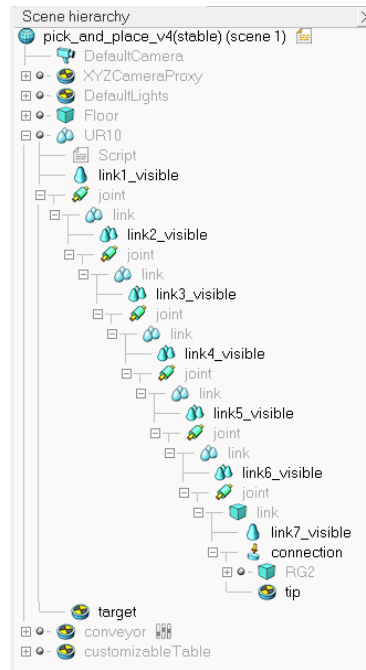


Θα χρησιμοποιήσουμε πάλι το Object/item shift και Object/item rotate ώστε να φέρουμε το tip στην άκρη του βραχίονα.

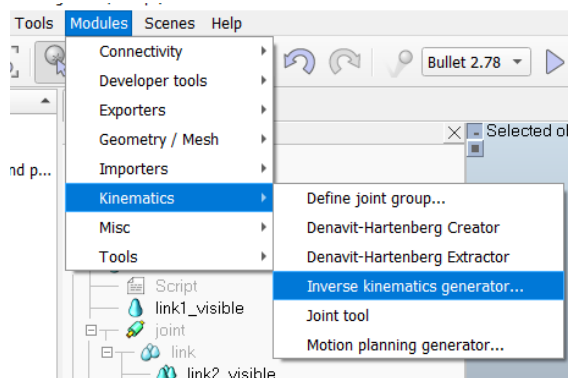


Έπειτα, θα ευθυγραμμίσουμε με τον ίδιο τρόπο το target με το tip και μετά θα σύρουμε το target στο UR10.

Προσομοίωση Ρομπότ μέσω Υπολογιστή



Για να κάνουμε την αντίστροφη κινηματική τώρα θα πατήσουμε Modules → Kinematics → Inverse kinematics generator...



Στο παράθυρο που μας εμφανίζει αν πατήσουμε στο Robot model και επιλέξουμε το UR10, θα πάρει αυτόματα τα Robot base, Robot tip και Robot target.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

The dialog box is titled "Inverse Kinematics generator" and contains the following fields and options:

- Robot model: (empty dropdown)
- Robot base: (empty dropdown)
- Robot tip: (empty dropdown)
- Robot target: (empty dropdown)
- Joint group: (empty dropdown)
- (leave empty to use all joints in the tip-target chain)
- Constraint: Position: X Y Z; Orientation: Alpha+Beta Gamma
- Solver: Max. iterations: 10; Damping factor: 0.100000
- Actively avoid joint limits
- Abort on joint limits hit
- Allow error
- Output errors
- Handling: Enabled; During simulation; When not simulating
- Script: IK variables (ikBase, ikTip, ikTarget, ...)
- Generate button

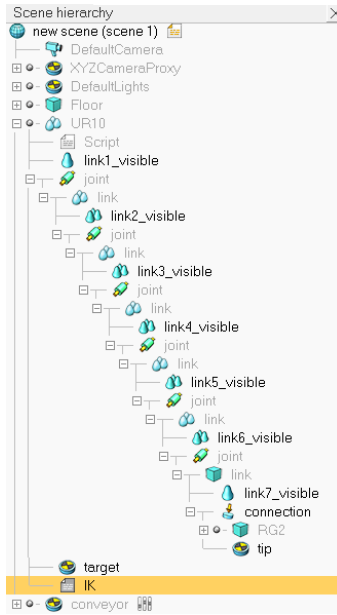
Ελέγχουμε αν έχει πάρει τα σωστά και πατάμε Generate.

The dialog box is titled "Inverse Kinematics generator" and contains the following fields and options:

- Robot model: UR10
- Robot base: /UR10
- Robot tip: /UR10/tip
- Robot target: /UR10/target
- Joint group: (empty dropdown)
- (leave empty to use all joints in the tip-target chain)
- Constraint: Position: X Y Z; Orientation: Alpha+Beta Gamma
- Solver: Max. iterations: 10; Damping factor: 0.100000
- Actively avoid joint limits
- Abort on joint limits hit
- Allow error
- Output errors
- Handling: Enabled; During simulation; When not simulating
- Script: IK variables (ikBase, ikTip, ikTarget, ...)
- Generate button

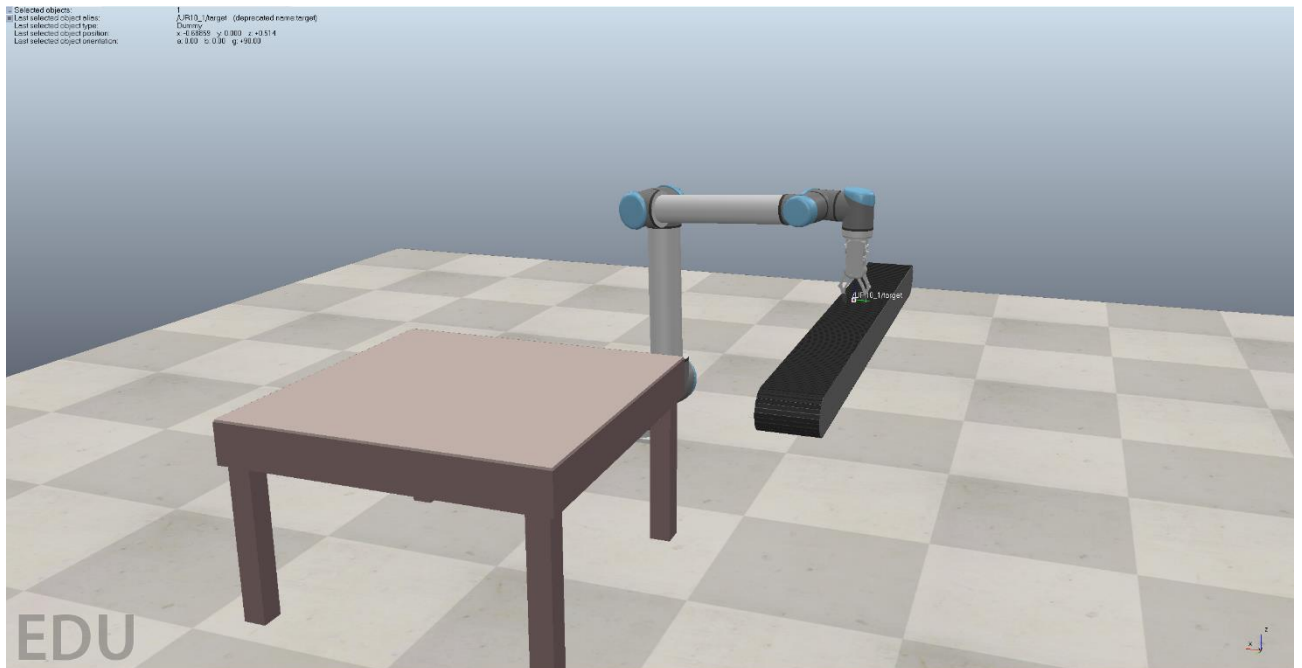
Αν το κάναμε σωστά θα έχει δημιουργηθεί ένα script αρχείο ΙΚ.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

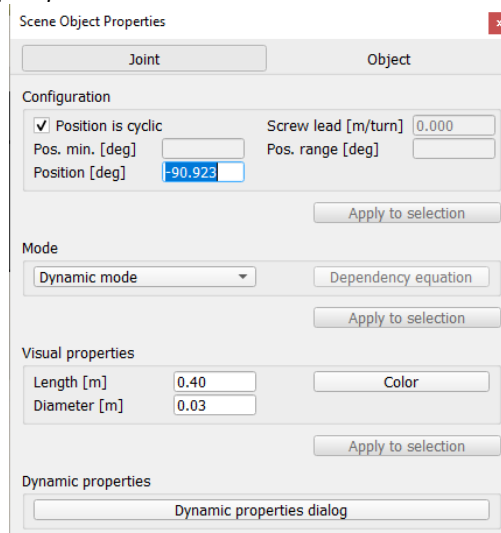


Τώρα αν πιάσουμε και μετακινήσουμε το target θα μετακινείται και όλος ο βραχίονας μας.

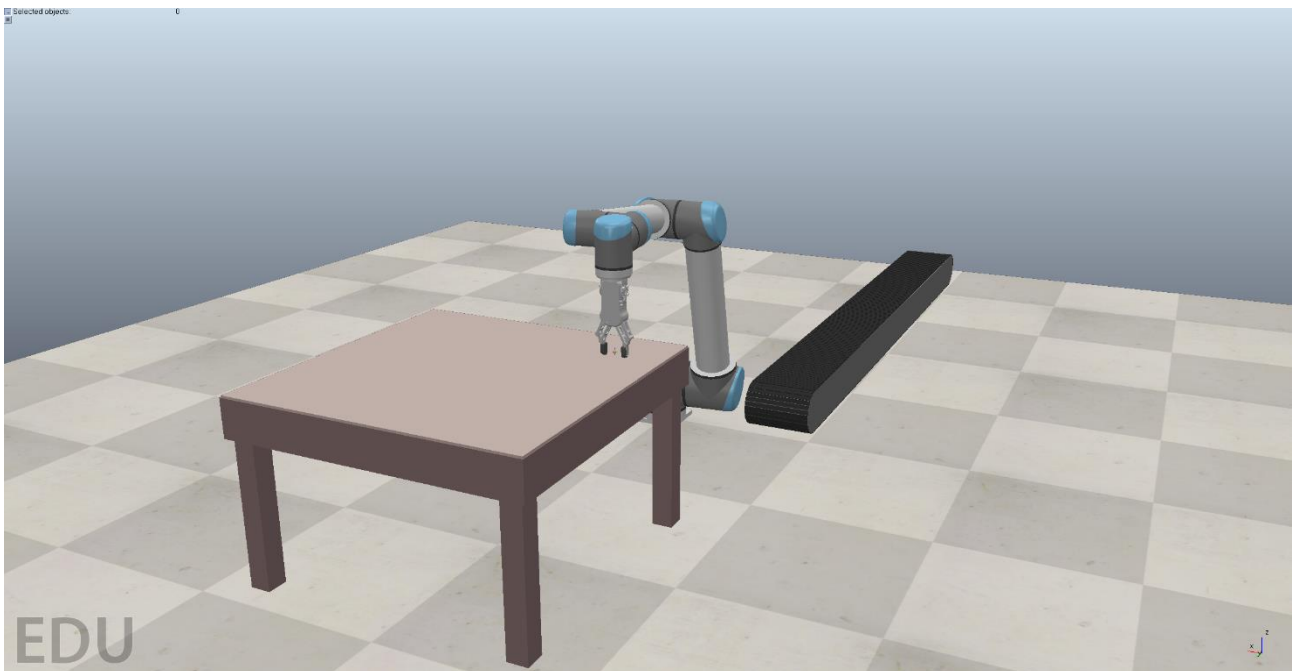
Μετακινούμε το target πάνω στη ταινία μεταφοράς στο σημείο που θα θέλαμε να πιάσουμε το κύβο και κοιτάμε ένα τα joint και σημειώσουμε τη γωνία της κάθε άρθρωσης. Τα σημειώνουμε και έπειτα μετακινούμε το target πάνω στο τραπέζι στο σημείο που θα θέλαμε να αφήσουμε το κύβο.



Προσομοίωση Ρομπότ μέσω Υπολογιστή



Οι αρθρώσεις είναι 6, οπότε κρατάμε 6 μοίρες. Προκειμένου να τα βλέπουμε σε μοίρες επιλέγουμε το Position is cyclic.



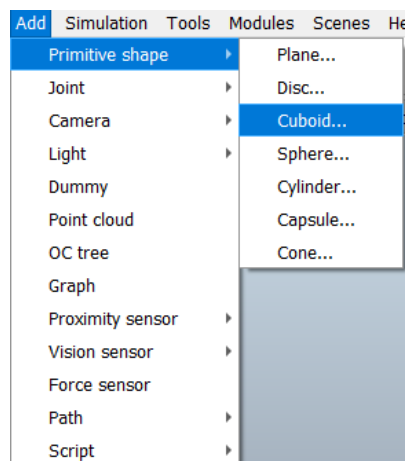
Πατάμε διπλό κλικ στο Script του UR10 και το ανοίγουμε. Παρατηρούμε ότι είναι γραμμένο σε γλώσσα LUA. Μέσα σε αυτό βρίσκονται εντολές που κινούν το βραχίονα μας.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

```
Simulation script "/UR10/Script"
LUA
1  sim=require'sim'
2
3  function moveToConfig(handles,maxVel,maxAccel,maxJerk, targetConf)
4      local params = {
5          joints = handles,
6          targetPos = targetConf,
7          maxVel = maxVel,
8          maxAccel = maxAccel,
9          maxJerk = maxJerk,
10     }
11     sim.moveToConfig(params)
12 end
13
14 function sysCall_thread()
15     local jointHandles={-1,-1,-1,-1,-1,-1}
16     for i=1,6,i do
17         jointHandles[i]=sim.getObject('../joint',{index=i-1})
18     end
19
20     local vel=120
21     local accel=40
22     local jerk=80
23     local maxVel={vel*math.pi/180,vel*math.pi/180,vel*math.pi/180,vel*math.pi/180,vel*math.pi/180,vel*math.pi/180}
24     local maxAccel={accel*math.pi/180,accel*math.pi/180,accel*math.pi/180,accel*math.pi/180,accel*math.pi/180,accel*math.pi/180}
25     local maxJerk={jerk*math.pi/180,jerk*math.pi/180,jerk*math.pi/180,jerk*math.pi/180,jerk*math.pi/180,jerk*math.pi/180}
26
27     local targetPos1={90*math.pi/180,90*math.pi/180,-90*math.pi/180,90*math.pi/180,90*math.pi/180,90*math.pi/180}
28     moveToConfig(jointHandles,maxVel,maxAccel,maxJerk, targetPos1)
29
30     local targetPos2={-90*math.pi/180,45*math.pi/180,90*math.pi/180,135*math.pi/180,90*math.pi/180,90*math.pi/180}
31     moveToConfig(jointHandles,maxVel,maxAccel,maxJerk, targetPos2)
32
33     local targetPos3={0,0,0,0,0,0}
34     moveToConfig(jointHandles,maxVel,maxAccel,maxJerk, targetPos3)
35 end
```

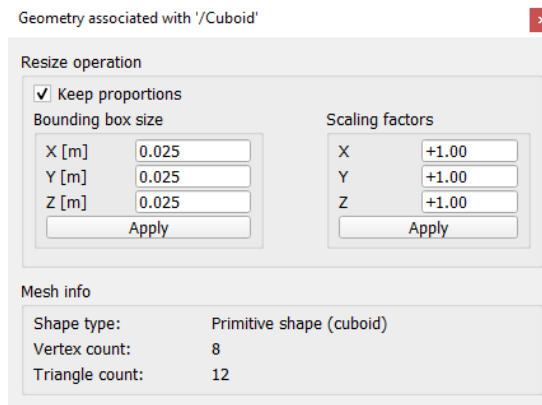
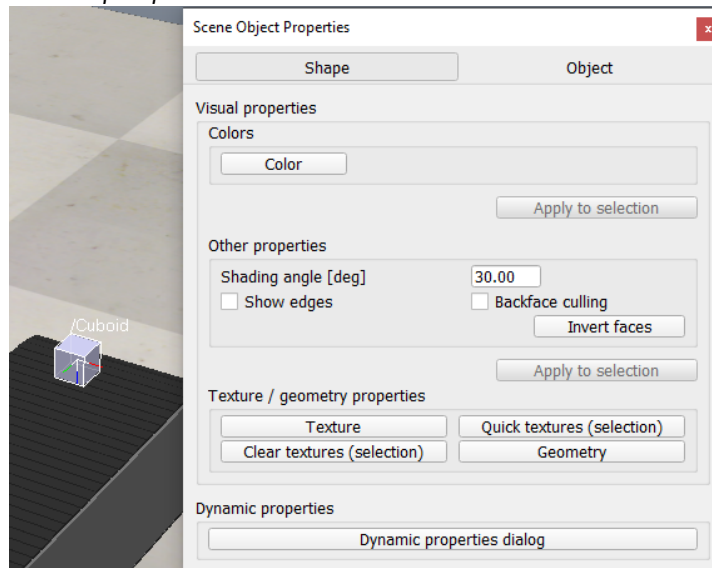
Θα αλλάξουμε το targetPos1 και θα βάλουμε τις μοίρες που σημειώσαμε για το σημείο που επιθυμούμε να πιάσουμε το κύβο, ενώ στο targetPos2 θα βάλουμε τις συντεταγμένες που θέλουμε να το αφήνει. Το targetPos3 είναι 0 για να επιστρέψει στην αρχική του θέση.

Τώρα θα προσθέσουμε ένα κύβο τον οποίο και θα πιάει ο βραχίονας μας από τη ταινία μεταφοράς και θα το μεταφέρει στο τραπέζι. Για να το κάνουμε αυτό πατάμε Add --> Primitive Shape --> Cuboid.

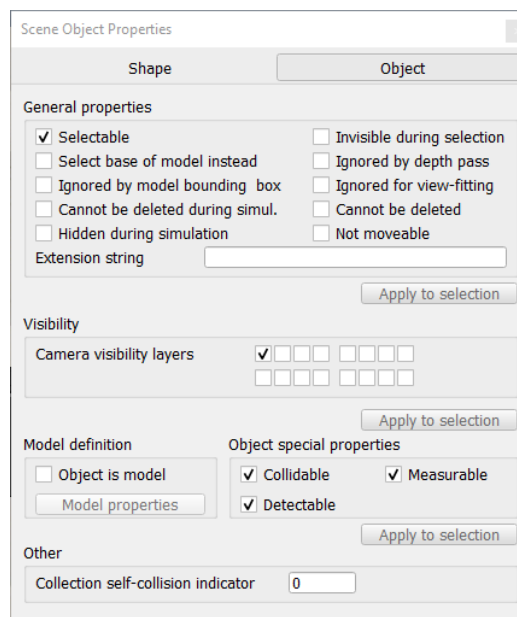


Αφού το τοποθετήσουμε πατάμε διπλό κλικ στο εικονίδιο του και στο παράθυρο Scene Object Properties πατάμε Geometry και αλλάζουμε το μέγεθος του σε 0.025.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

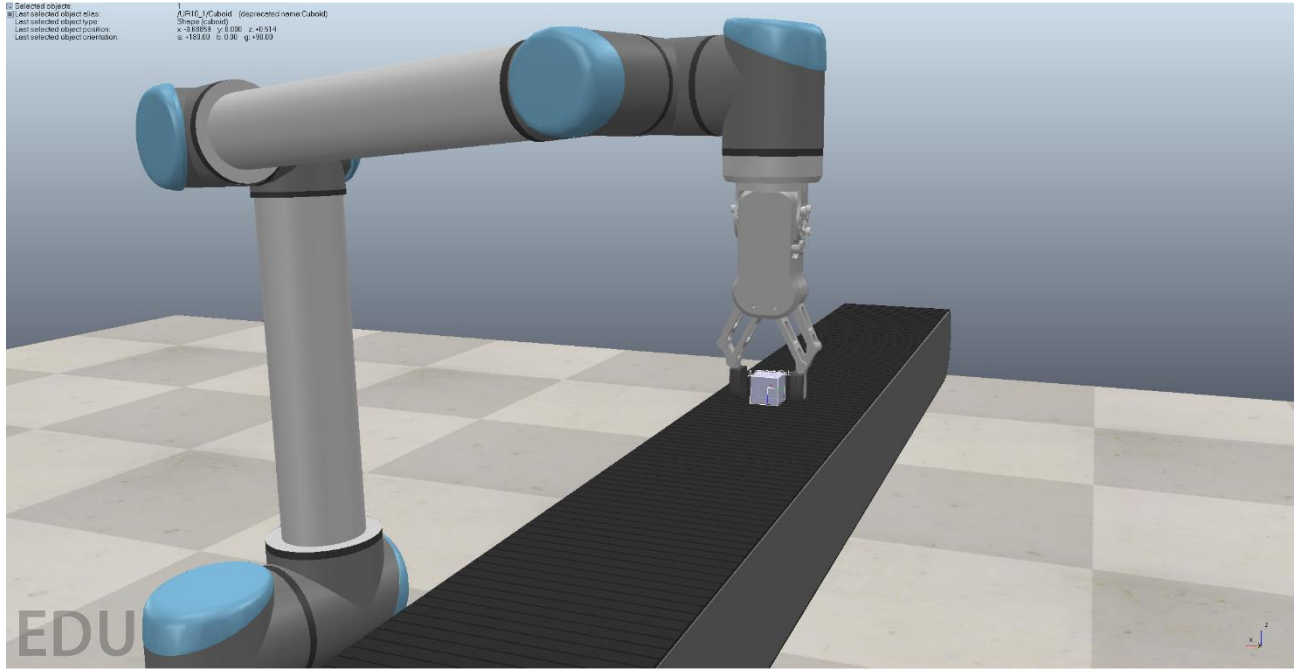


Επίσης πάμε στη καρτέλα Object και επιλέγουμε στο Object special properties τα Collidable, Measurable, Detectable ώστε να μπορεί αργότερα να ανιχνευθεί από τον αισθητήρα μας.



Πηγαίνουμε τον κύβο μας στο σημείο όπου η λαβή θα τον πιάσει. Ένας γρήγορος τρόπος να γίνει αυτό είναι να το κάνουμε child του target και μετά να το φέρουμε στο ίδιο σημείο.

Προσομοίωση Ρομπότ μέσω Υπολογιστή



Για να κινηθεί τώρα ο βραχίονας μας θα φτιάξουμε το script.

```
Simulation script "/UR10_1/Script"
1  sim = require 'sim'
2
3  function moveToConfig(handles, maxVel, maxAccel, maxJerk, targetConf)
4      local params = {
5          joints = handles,
6          targetPos = targetConf,
7          maxVel = maxVel,
8          maxAccel = maxAccel,
9          maxJerk = maxJerk,
10     }
11     sim.moveToConfig(params)
12 end
13
14 function sysCall_thread()
15
16     local jointHandles = {-1, -1, -1, -1, -1, -1}
17     for i = 1, 6 do
18         jointHandles[i] = sim.getObject('/..joint', {index = i-1})
19     end
20
21     local vel = 20000
22     local accel = 300
23     local jerk = 400
24     local maxVel = {vel * math.pi/180, vel * math.pi/180, vel * math.pi/180, vel * math.pi/180, vel * math.pi/180, vel * math.pi/180}
25     local maxAccel = {accel * math.pi/180, accel * math.pi/180, accel * math.pi/180, accel * math.pi/180, accel * math.pi/180, accel * math.pi/180}
26     local maxJerk = {jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180}
27
28     -- Position1 (above the cube)
29     local targetPos1 = {76.231*math.pi/180, -0.341*math.pi/180, 76.795*math.pi/180, 13.546*math.pi/180, -90.00*math.pi/180, -13.769*math.pi/180}
30     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos1)
31     sim.wait(0.05)
32
33     -- Position2 (grabbing position)
34
35     local targetPos2 = {76.231*math.pi/180, -1.649*math.pi/180, 86.813*math.pi/180, 4.836*math.pi/180, -90.00*math.pi/180, -13.769*math.pi/180}
36     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos2)
37     sim.wait(0.05)
38
39     -- Move back up after picking
40     local targetPos3 = {76.231*math.pi/180, -1.649*math.pi/180, 86.813*math.pi/180, 4.836*math.pi/180, -90.00*math.pi/180, -13.769*math.pi/180}
41     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos3)
42     sim.wait(0.05)
43
44     -- Move to the release position
45     local targetPos4 = {6.617*math.pi/180, 20.369*math.pi/180, 52.084*math.pi/180, 17.547*math.pi/180, -90.00*math.pi/180, -83.383*math.pi/180}
46     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos4)
47     sim.wait(0.05)
48
49     -- Move to the release point
50     local releasePos = {6.617*math.pi/180, 18.262*math.pi/180, 63.801*math.pi/180, 7.938*math.pi/180, -90.00*math.pi/180, -83.383*math.pi/180}
51     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, releasePos)
52     sim.wait(0.05)
53
54     -- Return to initial position
55     local initialPos = {0*math.pi/180, 0*math.pi/180, 0*math.pi/180, 0*math.pi/180, 0*math.pi/180, 0*math.pi/180}
56     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, initialPos)
57     sim.wait(0.05)
58
59 end
```

Πατάμε το σήμα του Play και ξεκινάει η προσομοίωση.



Τώρα θα γράψουμε ένα script για τη λαβή ώστε μόλις φτάνει στη σωστή θέση να κλείνει. Για να το καταφέρουμε αυτό θα στέλνουμε ένα σήμα από το script του βραχίονα, όταν θα είναι στη κατάλληλη θέση, ώστε να κλείνει η λαβή και άλλο ένα σήμα όταν θα φτάνει ο βραχίονας στη θέση που είναι να αφήσει τον κύβο.

```

Simulation script "/UR10_1/RG2/Script"
LUA
1  sim = require 'sim'
2
3  function sysCall_init()
4
5      motorHandle = sim.getObject('../openCloseJoint')
6
7      -- Initialize motor variables
8
9      motorVelocity = 0.7 -- m/s, set a suitable value for gripper closing speed
10     motorForce = 20 -- N, set a suitable value for gripper force
11
12     -- Initialize open gripper
13
14     sim.setJointTargetVelocity(motorHandle, motorVelocity)
15 end
16
17 function sysCall_actuation()
18
19     -- Check if the signal to control the gripper has been set
20     local gripperControl = sim.getInt32Signal('gripperControl')
21
22     if gripperControl == 1 then
23
24         -- Close the gripper by applying force to the motor joint
25         local v = -motorVelocity -- Closing velocity
26         sim.setJointTargetForce(motorHandle, motorForce)
27         sim.setJointTargetVelocity(motorHandle, v) -- Close the gripper
28
29         -- Clear the release signal so it only happens once
30         sim.clearInt32Signal('gripperControl')
31     end
32
33     -- Check if the signal to release the object has been set
34     local releaseSignal = sim.getInt32Signal('releaseObject')
35     if releaseSignal == 1 then
36
37         -- Open the gripper by reversing the motor
38         sim.setJointTargetVelocity(motorHandle, motorVelocity) -- Open the gripper
39
40
41         -- Clear the release signal so it only happens once
42         sim.clearInt32Signal('releaseObject')
43     end
44 end

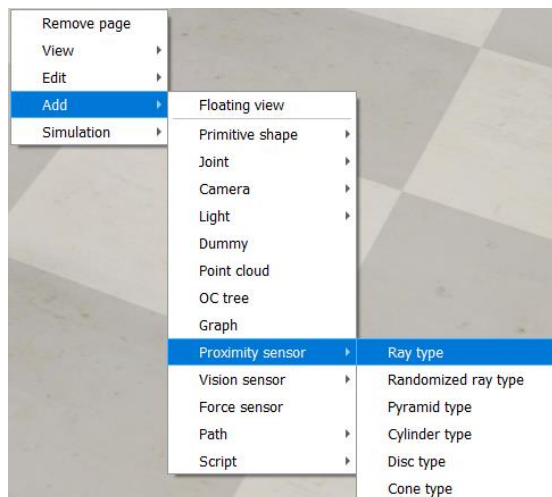
```

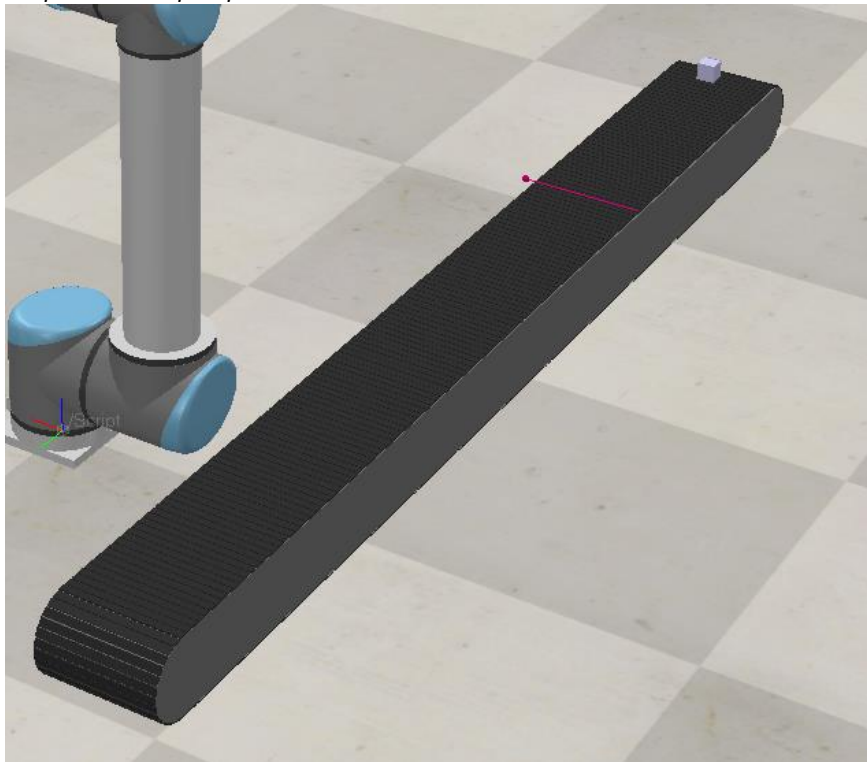
Προσομοίωση Ρομπότ μέσω Υπολογιστή

```
Simulation script 'UR10_1/Script'
1 sim = require 'sim'
2
3 function moveToConfig(handles, maxVel, maxAccel, maxJerk, targetConf)
4     local params = {
5         joints = handles,
6         targetPos = targetConf,
7         maxVel = maxVel,
8         maxAccel = maxAccel,
9         maxJerk = maxJerk,
10    }
11    sim.moveToConfig(params)
12 end
13
14 function sysCall_thread()
15     local jointHandles = {-1, -1, -1, -1, -1, -1}
16     for i = 1, 6 do
17         jointHandles[i] = sim.getObject('../joint', {index = i-1})
18     end
19
20     local vel = 20000
21     local accel = 300
22     local jerk = 400
23     local maxVel = {vel * math.pi/180, vel * math.pi/180, vel * math.pi/180, vel * math.pi/180, vel * math.pi/180, vel * math.pi/180}
24     local maxAccel = {accel * math.pi/180, accel * math.pi/180, accel * math.pi/180, accel * math.pi/180, accel * math.pi/180, accel * math.pi/180}
25     local maxJerk = {jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180}
26
27     -- Position1 (above the cube)
28     local targetPos1 = {76.231*math.pi/180, -0.341*math.pi/180, 76.795*math.pi/180, 13.546*math.pi/180, -90.00*math.pi/180, -13.769*math.pi/180}
29     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos1)
30     sim.wait(0.05)
31
32     -- Position2 (grabbing position)
33     local targetPos2 = {76.231*math.pi/180, -1.649*math.pi/180, 86.813*math.pi/180, 4.836*math.pi/180, -90.00*math.pi/180, -13.769*math.pi/180}
34     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos2)
35     sim.wait(0.05)
36
37     -- Close the gripper to pick up the object
38     sim.setInt32Signal('gripperControl', 1) -- Signal to close the gripper
39     sim.wait(0.05)
40
41     -- Move back up after picking
42     local targetPos3 = {76.231*math.pi/180, -1.649*math.pi/180, 86.813*math.pi/180, 4.836*math.pi/180, -90.00*math.pi/180, -13.769*math.pi/180}
43     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos3)
44     sim.setInt32Signal('resume_conveyor', 1)
45     sim.wait(0.05)
46
47     -- Move to the release position
48     local targetPos4 = {6.617*math.pi/180, 20.369*math.pi/180, 52.084*math.pi/180, 17.547*math.pi/180, -90.00*math.pi/180, -83.383*math.pi/180}
49     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos4)
50     sim.wait(0.05)
51
52     -- Move to the release point
53     local releasePos = {6.617*math.pi/180, 18.262*math.pi/180, 63.801*math.pi/180, 7.938*math.pi/180, -90.00*math.pi/180, -83.383*math.pi/180}
54     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, releasePos)
55     sim.wait(0.05)
56
57     -- Open the gripper to release the object
58     sim.setInt32Signal('releaseObject', 1) -- Signal to open the gripper
59     sim.wait(0.05)
60
61     -- Return to initial position
62     local initialPos = {0*math.pi/180, 0*math.pi/180, 0*math.pi/180, 0*math.pi/180, 0*math.pi/180, 0*math.pi/180}
63     moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, initialPos)
64     sim.wait(0.05)
65 end
```

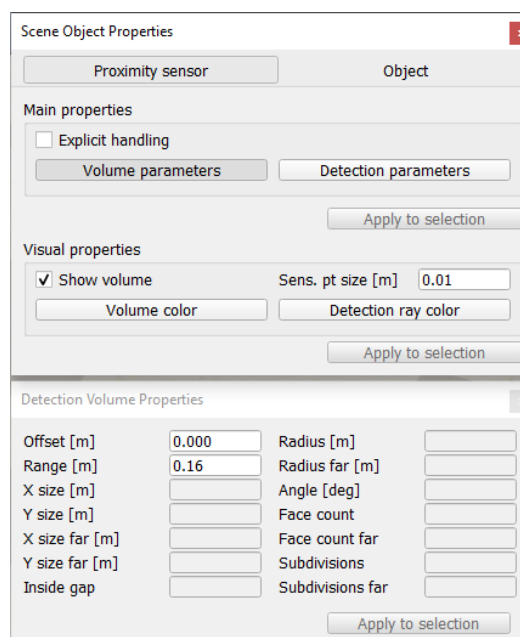
Επόμενος μας στόχος είναι να κάνουμε την ταινία μεταφοράς να κινείται. Ο κύβος που θα είναι πάνω της όταν θα περάσει από έναν αισθητήρα εγγύτητας θα σταματάει η ταινία μεταφοράς, μέχρι η λαβή να πιάσει το κύβο και τότε θα ξαναξεκινάει.

Πατάμε δεξί κλικ Add --> Proximity sensor --> Ray type.



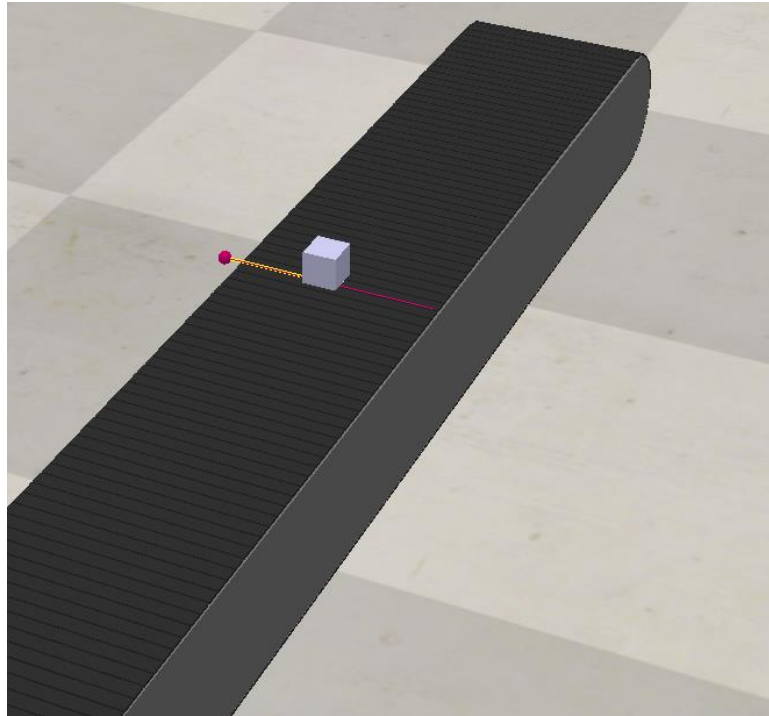


Με διπλό κλικ πάνω στον αισθητήρα ανοίγει το παράθυρο διαλόγου Scene Object Properties μέσα από το οποίο μπορούμε να κάνουμε διάφορες ρυθμίσεις.



Όταν ο κύβος ακουμπήσει την ακτίνα του αισθητήρα γίνεται κίτρινη ώστε να μας δείξει ότι κάτι εντοπίστηκε.

Προσομοίωση Ρομπότ μέσω Υπολογιστή



Στο κυρίως script γράφουμε το παρακάτω κώδικα.

```
Simulation script "/Script"
LUA
1  sim = require('sim')
2
3  function sysCall_init()
4
5      conveyorHandle = sim.getObject('/conveyor')
6      raySensorHandle = sim.getObject('/ProxSensor_1')
7
8      -- Set initial conveyor speed
9      conveyorSpeed = 0.15 -- Conveyor belt speed, can be adjusted
10     sim.writeCustomTableData(conveyorHandle, '__ctrl__', {vel=conveyorSpeed}) -- Start conveyor
11 end
12
13 function sysCall_actuation()
14     -- Check if the ray sensor detects the cube
15     local detectionState = sim.readProximitySensor(raySensorHandle)
16
17     if detectionState > 0 then
18         -- Stop the conveyor when a cuboid is detected
19         sim.writeCustomTableData(conveyorHandle, '__ctrl__', {vel=0.0}) -- Stop the conveyor belt
20
21         -- Log detection event for debugging
22         sim.addStatusBarMessage('Proximity sensor detected an object.')
23
24         -- Assign the task to UR10
25
26         sim.setInt32Signal('UR10_1_start', 1) -- Signal UR10_1 to start
27         sim.addStatusBarMessage('UR10_1 starts task.')
28
29     elseif detectionState == 0 then
30
31     end
32
33     -- Check to resume conveyor
34     if sim.getInt32Signal('resume_conveyor') == 1 then
35         sim.clearInt32Signal('resume_conveyor') -- Clear the signal
36         sim.writeCustomTableData(conveyorHandle, '__ctrl__', {vel=conveyorSpeed}) -- Resume conveyor
37     end
38 end
```

Στην αρχή δίνουμε αρχικές συνθήκες στη ταινία μεταφοράς μας και έπειτα όταν ο αισθητήρας μας εντοπίζει ένα αντικείμενο τότε στέλνει σήμα να σταματήσει η ταινία μεταφοράς και να ξεκινήσει η διαδικασία ώστε ο βραχίονας να πιάσει το κύβο.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Μέσα στο script του βραχίονα προσθέτουμε τα παρακάτω.

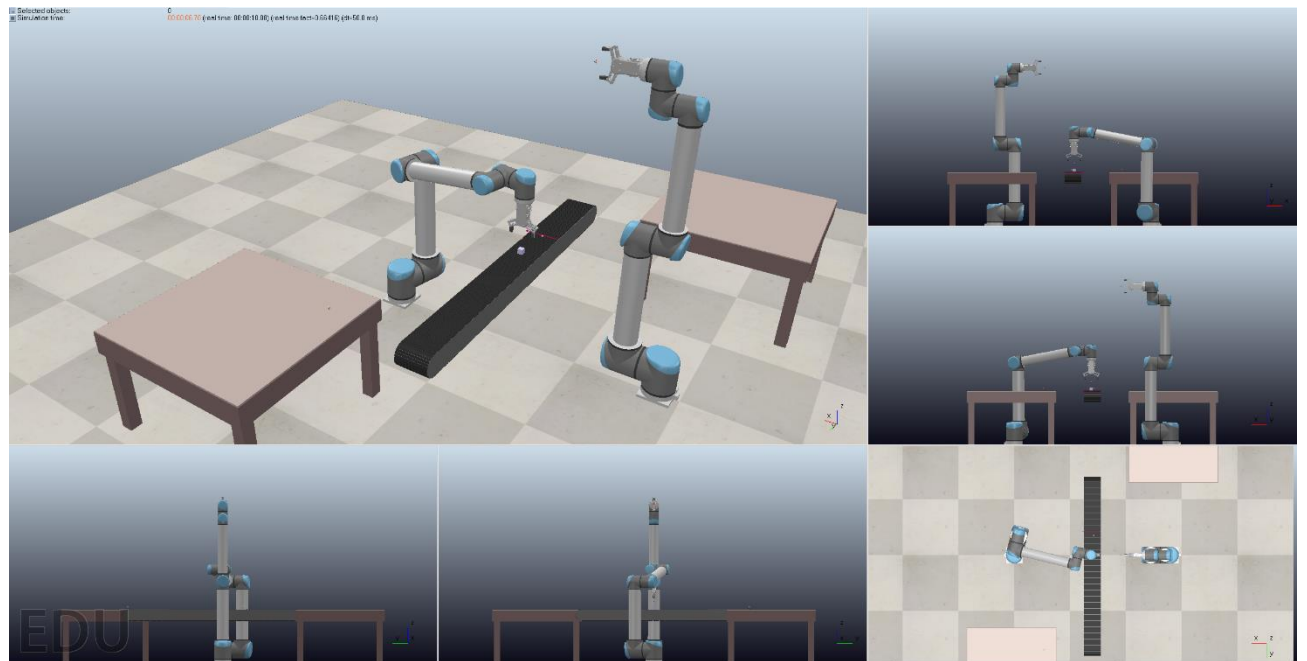
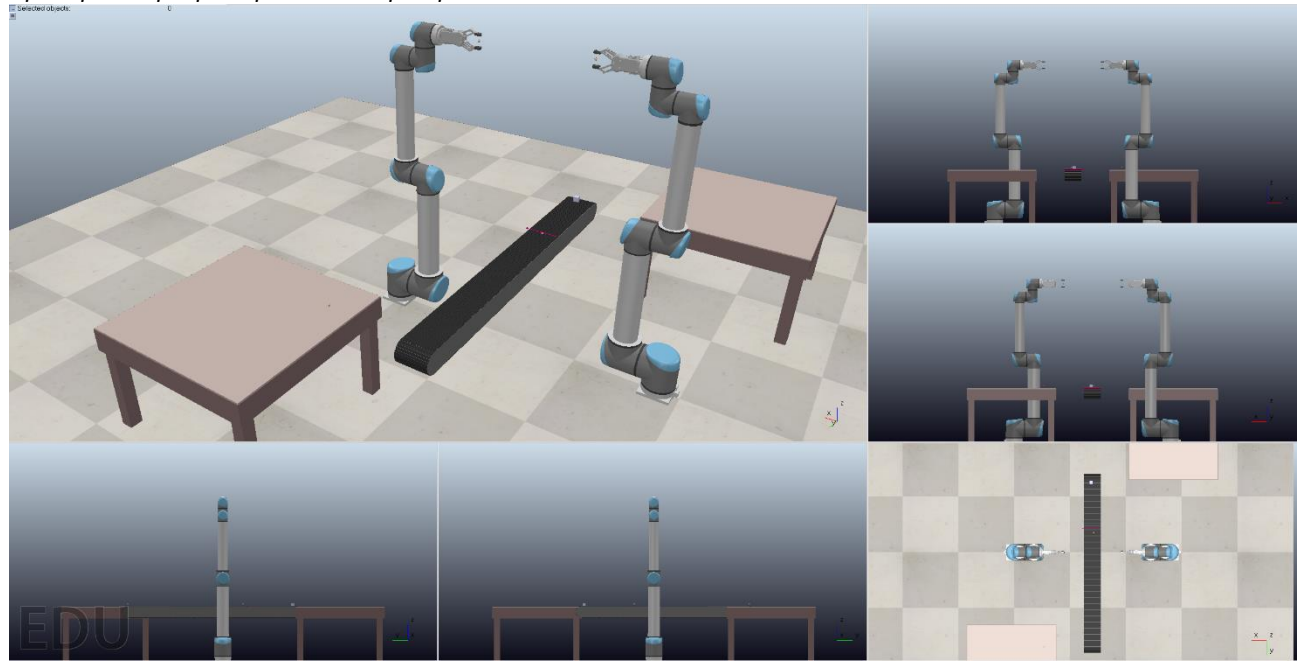
```
Simulation script "/UR10_1/Script"
LUA
9   maxJerk = maxJerk,
10  }
11  sim.moveToConfig(params)
12  end
13
14  function sysCall_thread()
15  local jointHandles = {-1, -1, -1, -1, -1, -1}
16  for i = 1, 6 do
17  jointHandles[i] = sim.getObject('../joint', {index = i-1})
18  end
19
20  local vel = 20000
21  local accel = 300
22  local jerk = 400
23  local maxVel = {vel * math.pi/180, vel * math.pi/180, vel * math.pi/180, vel * math.pi/180, vel * math.pi/180, vel * math.pi/180}
24  local maxAccel = {accel * math.pi/180, accel * math.pi/180, accel * math.pi/180, accel * math.pi/180, accel * math.pi/180, accel * math.pi/180}
25  local maxJerk = {jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180, jerk * math.pi/180}
26
27  -- Wait for the task signal from the central controller
28  while true do
29  local taskSignal = sim.getInt32Signal('UR10_1_start')
30  if taskSignal == 1 then
31  sim.clearInt32Signal('UR10_1_start') -- Clear the task signal
32  break -- Start the pick-up sequence
33  end
34  end
35
36  -- Position1 (above the cube)
37  local targetPos1 = {76.231*math.pi/180, -0.341*math.pi/180, 76.795*math.pi/180, 13.546*math.pi/180, 0, 0}
38  moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos1)
39  sim.wait(0.05)
40
41  --Position2 (grabbing position)
42  local targetPos2 = {76.231*math.pi/180, -1.649*math.pi/180, 86.813*math.pi/180, 4.836*math.pi/180, 0, 0}
43  moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos2)
44  sim.wait(0.05)
45
46  -- Close the gripper to pick up the object
47  sim.setInt32Signal('gripperControl', 1) -- Signal to close the gripper
48  sim.wait(0.05)
49
50  -- Move back up after picking
51  local targetPos3 = {76.231*math.pi/180, -1.649*math.pi/180, 86.813*math.pi/180, 4.836*math.pi/180, 0, 0}
52  moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos3)
53  sim.setInt32Signal('resume conveyor', 1) --Signal to resume the conveyor
54  sim.wait(0.05)
55
56  -- Move to the release position
57  local targetPos4 = {6.617*math.pi/180, 20.369*math.pi/180, 52.084*math.pi/180, 17.547*math.pi/180, 0, 0}
58  moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, targetPos4)
59  sim.wait(0.05)
60
61  -- Move to the release point
62  local releasePos = {6.617*math.pi/180, 18.262*math.pi/180, 63.801*math.pi/180, 7.938*math.pi/180, 0, 0}
63  moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, releasePos)
64  sim.wait(0.05)
65
66  -- Open the gripper to release the object
67  sim.setInt32Signal('releaseObject', 1) -- Signal to open the gripper
68  sim.wait(0.05)
69
70  -- Return to initial position
71  local initialPos = {0*math.pi/180, 0*math.pi/180, 0*math.pi/180, 0*math.pi/180, 0*math.pi/180, 0*math.pi/180}
72  moveToConfig(jointHandles, maxVel, maxAccel, maxJerk, initialPos)
73  sim.wait(0.05)
74
75  -- Signal that the task is done
76  sim.setInt32Signal('UR10_1_done', 1)
77  end
```

Ουσιαστικά περιμένει ο βραχίονας μας σήμα από το κυρίως script. Όταν αυτό έρθει, ξεκινάει η διαδικασία για το πιάσιμο του κύβου και το σήμα για τη ταινία μεταφοράς να συνεχίσει.

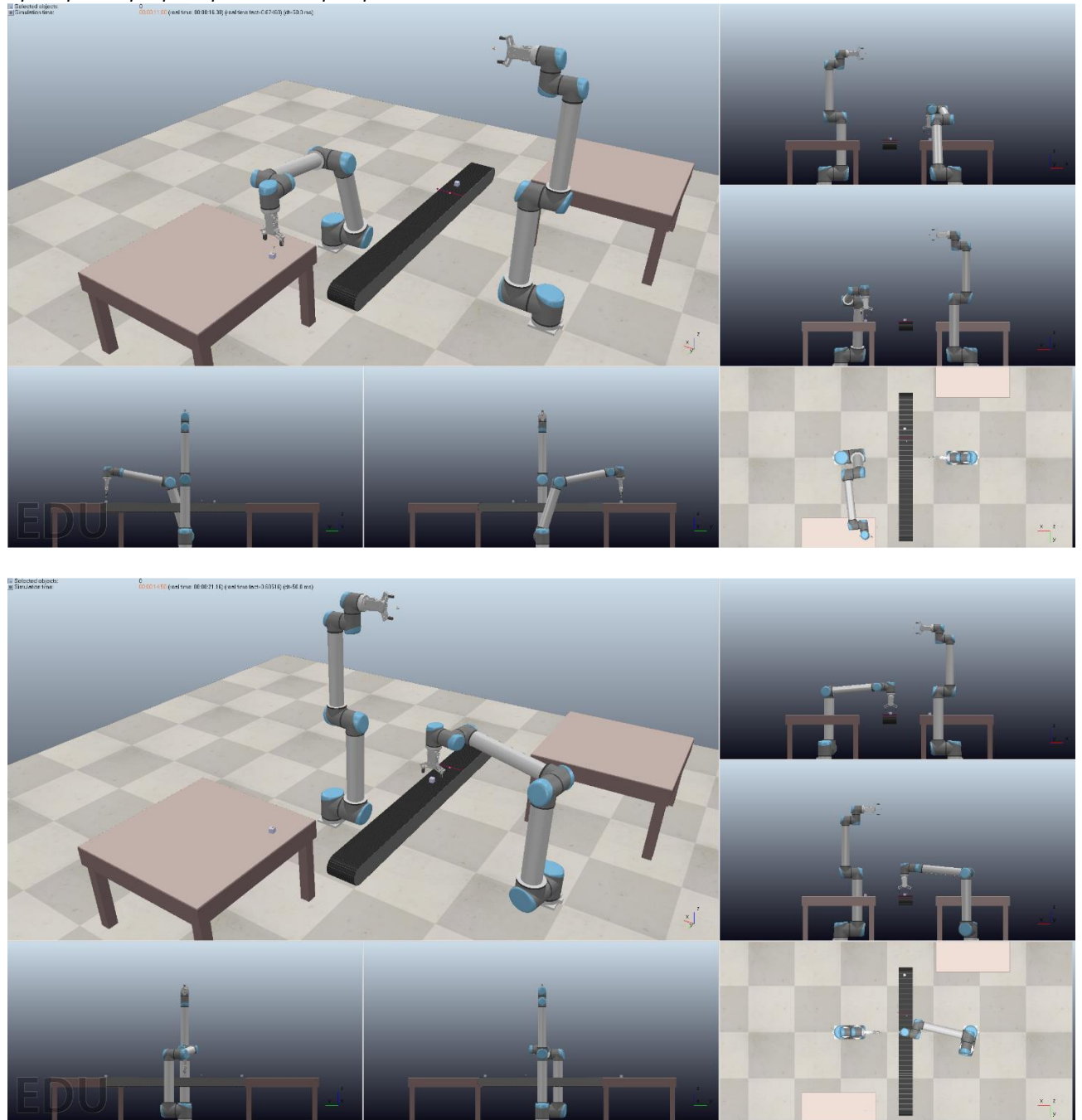
Προσθέτουμε άλλον ένα βραχίονα και κάνουμε την ίδια διαδικασία.

Τελικός στόχος μας ήταν να έχουμε μία ταινία μεταφοράς πάνω στην οποία θα έρχονται κύβοι. Όταν θα τους ανιχνεύει ο αισθητήρας εγγύτητας, η ταινία μεταφοράς, θα σταματάει και θα εκκινείτε ο πρώτος διαθέσιμος βραχίονας. Αφού πιάσει το κύβο θα εκκινείται ξανά η ταινία μεταφοράς και ο βραχίονας θα αφήνει το κύβο πάνω σε ένα τραπέζι. Κατά τη διάρκεια αυτής της διαδικασίας αν ένας κύβος ανιχνευθεί από τον αισθητήρα τότε θα σταματάει η ταινία μεταφοράς και πρώτος διαθέσιμος βραχίονας θα καλείται να πιάσει το κύβο (στη περίπτωση μας θα είναι ο δεύτερος διότι ο πρώτος θα είναι ακόμα απασχολημένος).

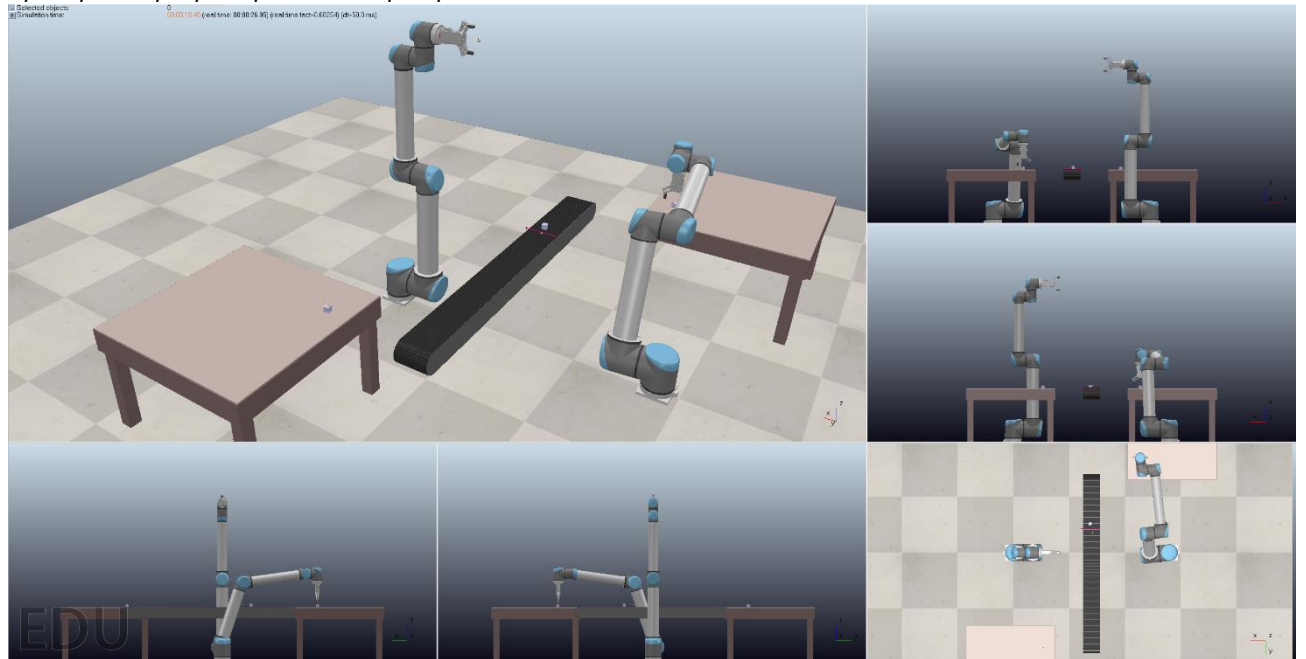
Προσομοίωση Ρομπότ μέσω Υπολογιστή



Προσομοίωση Ρομπότ μέσω Υπολογιστή



Προσομοίωση Ρομπότ μέσω Υπολογιστή



ΚΕΦΑΛΑΙΟ 2^ο : Gazebo

2.1 Τι είναι το Gazebo;

Το Gazebo είναι ένας τρισδιάστατος δυναμικός προσομοιωτής με τη δυνατότητα να προσομοιώνει με ακρίβεια και αποδοτικότητα πληθυσμούς ρομπότ σε πολύπλοκα εσωτερικά και εξωτερικά περιβάλλοντα. Παρόλο που είναι παρόμοιο με τις μηχανές παιχνιδιών, το Gazebo προσφέρει προσομοίωση φυσικής σε πολύ υψηλότερο επίπεδο πιστότητας, μια σειρά από αισθητήρες και διεπαφές τόσο για χρήστες όσο και για προγράμματα.

Τυπικές χρήσεις του Gazebo περιλαμβάνουν:

- δοκιμή αλγορίθμων ρομποτικής,
- σχεδιασμό ρομπότ,
- εκτέλεση δοκιμών παλινδρόμησης με ρεαλιστικά σενάρια.

Μερικά βασικά χαρακτηριστικά του Gazebo περιλαμβάνουν:

- πολλαπλές μηχανές φυσικής,
- μια πλούσια βιβλιοθήκη μοντέλων ρομπότ και περιβαλλόντων,
- μια μεγάλη ποικιλία αισθητήρων,
- εύχρηστες προγραμματιστικές και γραφικές διεπαφές.

2.2 Αρχιτεκτονική Gazebo Sim

Το Gazebo Sim είναι το κύριο σημείο εισόδου της εφαρμογής Gazebo, που γενικά περιλαμβάνει τη χρήση όλων των άλλων βιβλιοθηκών του Gazebo. Ως εκτελέσιμο, τρέχει μια προσομοίωση εκκινώντας δύο διεργασίες: τη διεργασία του backend server και τη διεργασία του frontend client.

Η βασική λειτουργικότητα του Gazebo Sim είναι η επικοινωνία client-server και τα διάφορα plugins που φορτώνει για να διευκολύνει την προσομοίωση. Τα plugins είναι βιβλιοθήκες που φορτώνονται κατά την εκτέλεση. Επιτρέπουν στο Gazebo Sim να αξιοποιεί άλλες βιβλιοθήκες του Gazebo. Υπάρχουν πολλοί τύποι plugins στο πλαίσιο του Gazebo. Για παράδειγμα, υπάρχουν plugins που εισάγουν νέες μηχανές φυσικής ή μηχανές rendering.

Επειδή φορτώνονται κατά την εκτέλεση, το Gazebo δεν χρειάζεται να γίνει recompile για να προσθέσουμε ή να αφαιρέσουμε plugins. Το Gazebo Sim περιλαμβάνει πολλά plugins εξ ορισμού (plugins server, plugins GUI Gazebo, plugins GUI Gazebo Sim και άλλα), όλα από τα οποία είναι προαιρετικά και μπορούν να αφαιρεθούν από τον χρήστη. Οι χρήστες μπορούν επίσης να προσθέσουν περισσότερα plugins και ακόμη και να γράψουν τα δικά τους plugins που θα μεταγλωττιστούν σε αρχεία βιβλιοθηκών.

Οι βιβλιοθήκες του Gazebo είναι αρθρωτές. Τα plugins επιτρέπουν στο Gazebo Sim να αξιοποιεί άλλες βιβλιοθήκες. Για παράδειγμα, το Gazebo Physics και το Gazebo Sim είναι ανεξάρτητα το ένα από το άλλο, επομένως το Gazebo Sim έχει ένα plugin φυσικής που χρησιμοποιεί το Gazebo Physics ως βιβλιοθήκη και ενσωματώνει τις ιδιαιτερότητες του Gazebo, επιτρέποντας στη φυσική να είναι ένα σύστημα που λειτουργεί στον βρόχο προσομοίωσης.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Η βιβλιοθήκη Gazebo Physics χρησιμοποιείται μόνο στο plugin φυσικής, όχι σε άλλα plugins ούτε στον πυρήνα του Gazebo Sim. Ορισμένες βιβλιοθήκες χρησιμοποιούνται μόνο από ένα plugin ή ένα σε κάθε διεργασία (frontend και backend). Ωστόσο, ορισμένες βασικές βιβλιοθήκες χρησιμοποιούνται από όλα τα plugins, όπως το Gazebo Common που παρέχει κοινή λειτουργικότητα όπως καταγραφή, χειρισμό συμβολοσειρών, αλληλεπίδραση με το σύστημα αρχείων κ.λπ., σε όλα τα plugins. Άλλες τέτοιες βιβλιοθήκες περιλαμβάνουν το Gazebo Plugin, το Gazebo Math, το SDF format και άλλα.

Υπάρχουν ορισμένα plugins τόσο στις διεργασίες frontend όσο και backend που φορτώνονται εξ ορισμού σε κάθε έκδοση του Gazebo. Πολλά άλλα plugins είναι προαιρετικά. Ένα κοινό προαιρετικό plugin είναι το plugin αισθητήρων. Τόσο τα προαιρετικά όσο και τα προεπιλεγμένα plugins μπορούν να αφαιρεθούν ή να προστεθούν ανά πάσα στιγμή το Gazebo Sim θα συνεχίσει να λειτουργεί με περιορισμένη λειτουργικότητα. Αυτές οι επιδείξεις στην Παραμετροποίηση του Server και την Παραμετροποίηση του GUI παρουσιάζουν αυτή τη λειτουργικότητα.

Η διαδικασία προσομοίωσης απεικονίζεται στο διάγραμμα παρακάτω και εξηγείται περαιτέρω στις ενότητες της διαδικασίας Backend και Frontend που ακολουθούν.

Το Gazebo χωρίζεται στις πέντε παρακάτω βιβλιοθήκες

- **Φυσικής:** διατηρεί και ανανεώνει τη φυσική κατάσταση της προσομοίωσης. Rendering: οπτικοποιεί την κατάσταση προσομοίωσης.
- **Αισθητήρες:** Δημιουργία δεδομένων του αισθητήρα. Μεταφορές: Χειρίζεται την επικοινωνία μεταξύ διεργασιών.
- **GUI:** Επιτρέπει την οπτικοποίηση και τις πράξεις χειραγώγησης της προσομοίωσης.

Αυτές οι βιβλιοθήκες χρησιμοποιούνται από δύο κύριες διαδικασίες:

- **Διακομιστής (server):** Τρέχει το βρόχο φυσικής και δημιουργίας δεδομένων του αισθητήρα.
 - Εκτελέσιμα: gzserver
 - Βιβλιοθήκες: Φυσική, Αισθητήρες, Rendering, Μεταφορές
- **Πελάτης (client):** Παρέχει αλληλεπίδραση του χρήστη και την απεικόνιση μιας προσομοίωσης.
 - Εκτελέσιμα: gzclient
 - Βιβλιοθήκες: Μεταφορές, Rendering, GUI

2.3 Οδηγίες εγκατάστασης για το Ubuntu

Πρώτα εγκαθιστούμε κάποια απαραίτητα εργαλεία:

Ανοίγουμε το τερματικό:

Προσομοίωση Ρομπότ μέσω Υπολογιστή

```
diplwmatiki@diplwmatiki: ~  
bash: /opt/ros/noetic/setup.bash: No such file or directory  
bash: /opt/ros/melodic/setup.bash: No such file or directory  
diplwmatiki@diplwmatiki:~$
```

Πληκτρολογούμε τις παρακάτω εντολές:

```
sudo apt-get update
```

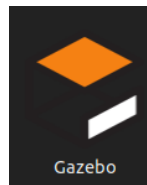
```
sudo apt-get install lsb-release wget gnupg
```

```
diplwmatiki@diplwmatiki: ~  
bash: /opt/ros/noetic/setup.bash: No such file or directory  
bash: /opt/ros/melodic/setup.bash: No such file or directory  
diplwmatiki@diplwmatiki:~$ sudo apt-get update  
diplwmatiki@diplwmatiki:~$ sudo apt-get install lsb-release wget gnupg
```

Έπειτα εγκαθιστούμε τη τελευταία έκδοση του Gazebo πληκτρολογώντας τη παρακάτω εντολή:

```
diplwmatiki@diplwmatiki: ~  
bash: /opt/ros/noetic/setup.bash: No such file or directory  
bash: /opt/ros/melodic/setup.bash: No such file or directory  
diplwmatiki@diplwmatiki:~$ curl -sSL http://get.gazebosim.org | sh
```

Όλες οι βιβλιοθήκες πρέπει να είναι έτοιμες για χρήση και η εφαρμογή gazebo sim πρέπει να είναι έτοιμη για εκτέλεση.



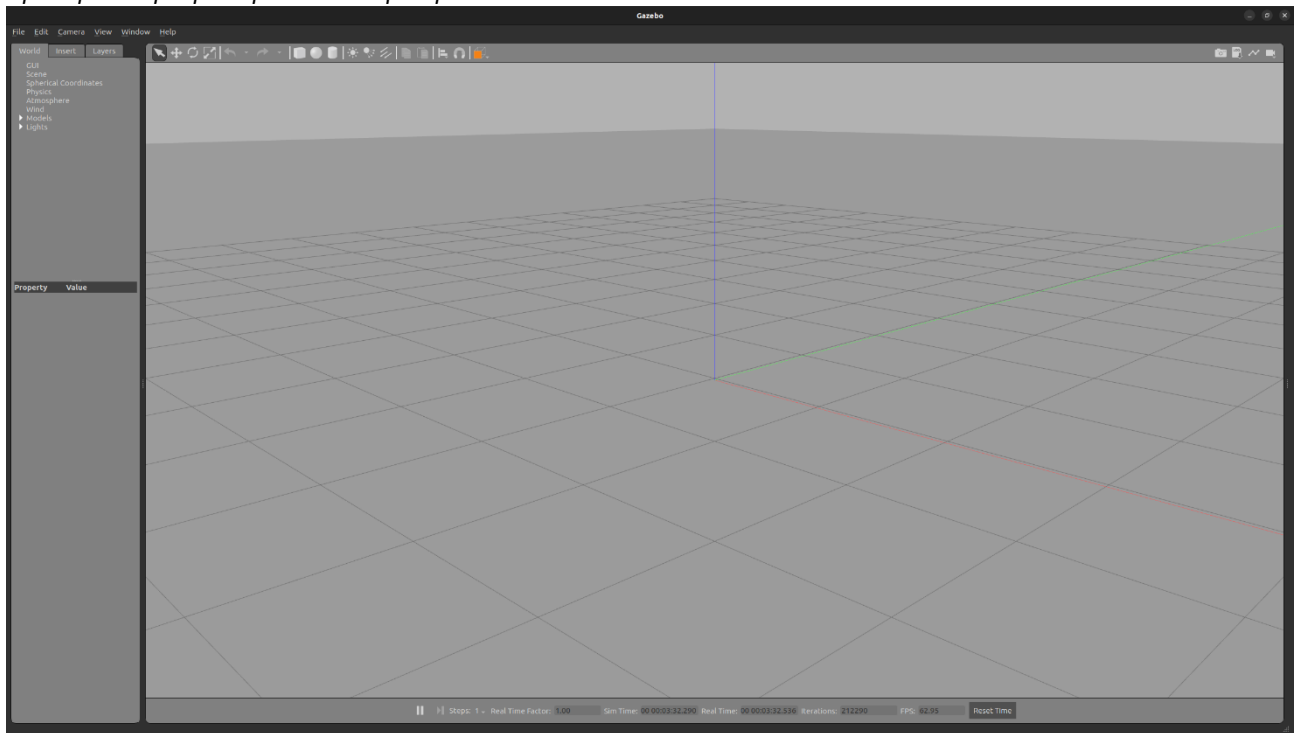
Το παραπάνω σύμβολο είναι το σύμβολο της εφαρμογής Gazebo. Κάνουμε κλικ πάνω του και το εκτελούμε.

2.4 User Interface/Διεπαφή Χρήστη

Αυτή είναι μια εισαγωγή στη Γραφική Διεπαφή Χρήστη (Graphical User Interface) του Gazebo, ή GUI.

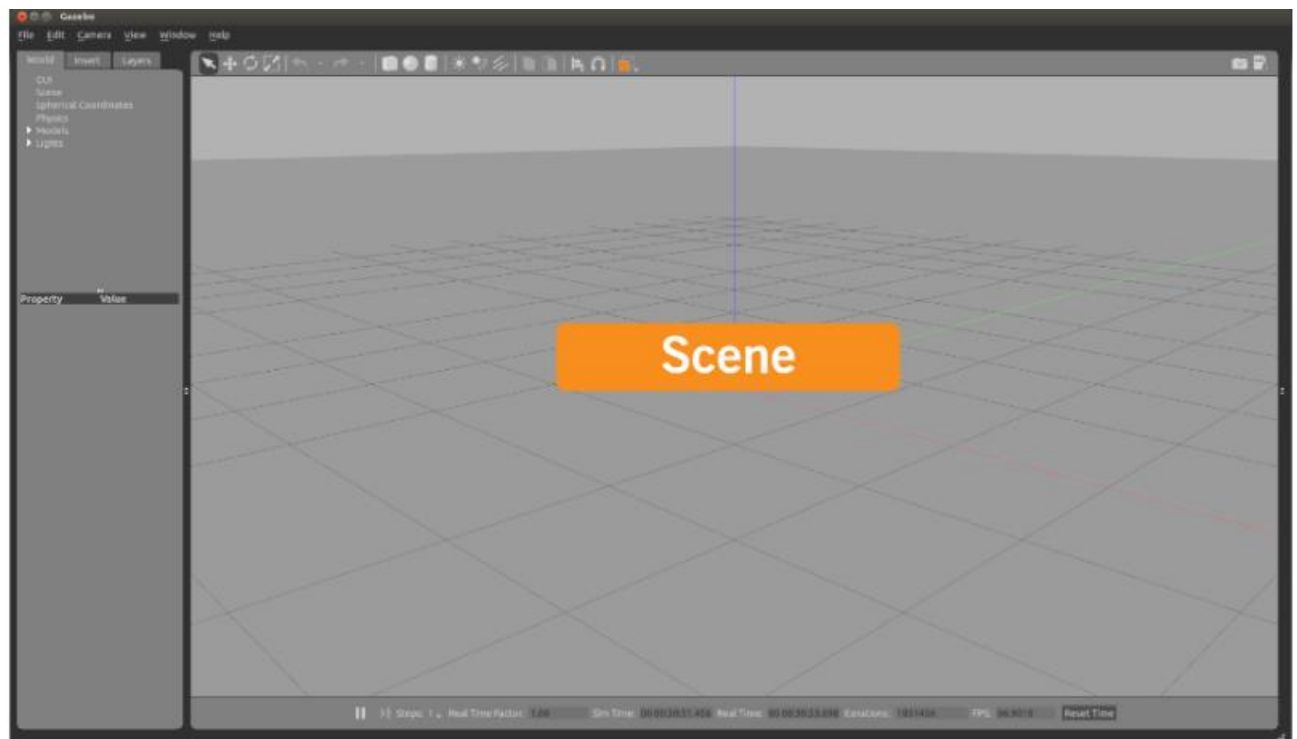
2.4.1 GUI

Αυτό είναι αυτό που θα πρέπει να βλέπουμε:



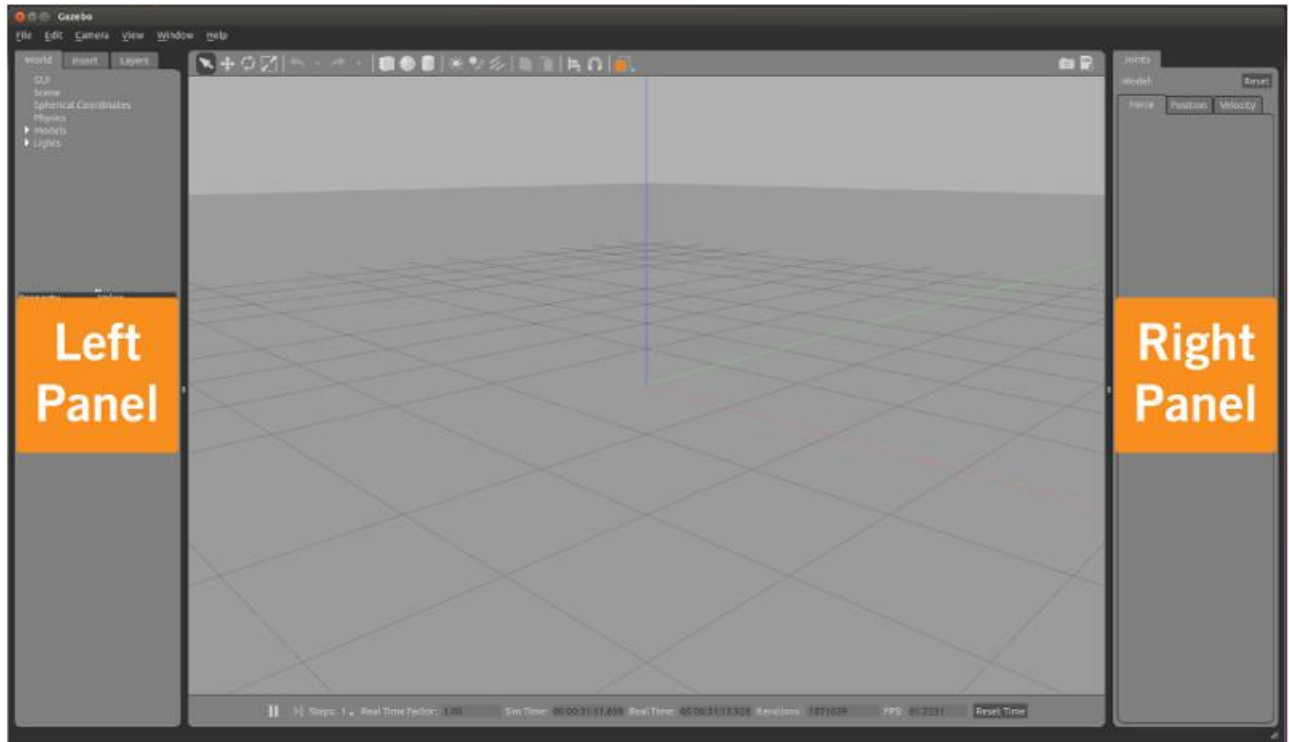
2.4.2 Η Σκηνή

Η σκηνή είναι το κύριο μέρος του προσομοιωτή. Εδώ είναι όπου τα προσομοιωμένα αντικείμενα κινούνται και αλληλεπιδρούν με το περιβάλλον.



2.4.3 Τα Πάνελ

Τα πλαϊνά πάνελ —δεξιά και αριστερά— μπορούν να εμφανιστούν, να κρυφτούν ή να αλλάξουν μέγεθος, σύροντας τη γραμμή που τα χωρίζει από τη Σκηνή.



2.4.3.1 Αριστερό Πάνελ

Το αριστερό πάνελ εμφανίζεται από προεπιλογή όταν ξεκινάμε το Gazebo. Υπάρχουν τρεις καρτέλες στο πάνελ:

- **WORLD:** Η καρτέλα World εμφανίζει τα μοντέλα που βρίσκονται αυτήν τη στιγμή στη σκηνή και μας επιτρέπει να βλέπουμε και να τροποποιούμε τις παραμέτρους των μοντέλων, όπως η θέση τους. Μπορούμε επίσης να αλλάξουμε τη γωνία της κάμερας επεκτείνοντας την επιλογή "GUI" και προσαρμόζοντας τη θέση της κάμερας.
- **INSERT:** Η καρτέλα Insert είναι εκεί όπου προσθέτουμε νέα αντικείμενα (μοντέλα) στην προσομοίωση. Για να δούμε τη λίστα των μοντέλων, ίσως χρειαστεί να κάνουμε κλικ στο βελάκι για να επεκτείνουμε το φάκελο. Κάνουμε κλικ (και αφήνουμε) στο μοντέλο που θέλουμε να εισάγουμε και κάνουμε ξανά κλικ στη Σκηνή για να το προσθέσουμε.
- **LAYERS:** Η καρτέλα Layers οργανώνει και εμφανίζει τις διάφορες ομάδες οπτικοποίησης που είναι διαθέσιμες στην προσομοίωση, αν υπάρχουν. Ένα επίπεδο μπορεί να περιέχει ένα ή περισσότερα μοντέλα. Αν ενεργοποιήσουμε ή απενεργοποιήσουμε ένα επίπεδο, τα μοντέλα σε αυτό το επίπεδο θα εμφανιστούν ή θα κρυφτούν.

Αυτή είναι μια προαιρετική λειτουργία, οπότε αυτή η καρτέλα θα είναι άδεια στις περισσότερες περιπτώσεις.

2.4.3.2 Δεξί Πάνελ (κρυφό από προεπιλογή)

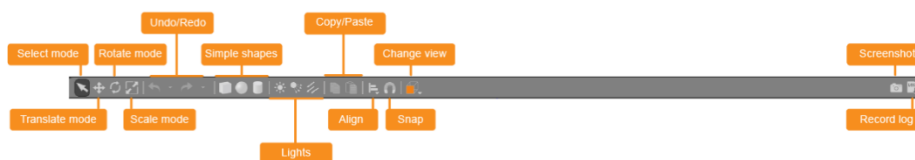
Το δεξί πάνελ είναι κρυφό από προεπιλογή. Κάνουμε κλικ και σύρουμε τη γραμμή για να το ανοίξουμε. Το δεξί πάνελ μπορεί να χρησιμοποιηθεί για να αλληλοεπιδράσουμε με τα κινητά μέρη ενός επιλεγμένου μοντέλου (τις αρθρώσεις). Αν δεν υπάρχουν επιλεγμένα μοντέλα στη Σκηνή, το πάνελ δεν εμφανίζει καμία πληροφορία.

2.4.4 Τα Εργαλεία

Η διεπαφή του Gazebo διαθέτει δύο γραμμές εργαλείων. Η μία βρίσκεται ακριβώς πάνω από τη Σκηνή και η άλλη ακριβώς από κάτω.

2.4.4.1 Άνω Γραμμή Εργαλείων

Η κύρια γραμμή εργαλείων περιλαμβάνει μερικές από τις πιο χρησιμοποιούμενες επιλογές για την αλληλεπίδραση με τον προσομοιωτή, όπως κουμπιά για: επιλογή, μετακίνηση, περιστροφή και κλιμάκωση αντικειμένων· δημιουργία απλών σχημάτων (π.χ. κύβος, σφαίρα, κύλινδρος)· και αντιγραφή/επικόλληση.



- Λειτουργία Επιλογής/Select mode: Πλοήγηση στη σκηνή
- Λειτουργία Μετακίνησης/Translate mode: Επιλογή μοντέλων που θέλουμε να μετακινήσουμε
- Λειτουργία Περιστροφής/Rotate mode: Επιλογή μοντέλων που θέλουμε να περιστρέψουμε
- Λειτουργία Κλιμάκωσης/Scale mode: Επιλογή μοντέλων που θέλουμε να κλιμακώσουμε
- Αναίρεση/Επανάληψη/Undo/Redo: Αναίρεση/επανάληψη ενεργειών στη σκηνή
- Απλά σχήματα/Simple shapes: Εισαγωγή απλών σχημάτων στη σκηνή
- Φώτα/Lights: Προσθήκη φωτισμού στη σκηνή
- Αντιγραφή/επικόλληση/Copy/Paste: Αντιγραφή/επικόλληση μοντέλων στη σκηνή
- Ευθυγράμμιση/Align: Ευθυγράμμιση μοντέλων μεταξύ τους
- Στερέωση/Snap: Στερέωση ενός μοντέλου σε άλλο
- Αλλαγή προβολής/Change view: Προβολή της σκηνής από διάφορες γωνίες

2.4.4.2 Κάτω Γραμμή Εργαλείων

Η κάτω γραμμή εργαλείων εμφανίζει δεδομένα σχετικά με την προσομοίωση, όπως το χρόνο της προσομοίωσης και τη σχέση του με τον πραγματικό χρόνο. Ο "χρόνος προσομοίωσης" αναφέρεται στο πόσο γρήγορα περνάει ο χρόνος στον προσομοιωτή όταν εκτελείται μια προσομοίωση. Η

Προσομοίωση Ρομπότ μέσω Υπολογιστή

προσομοίωση μπορεί να είναι πιο αργή ή πιο γρήγορη από τον πραγματικό χρόνο, ανάλογα με το πόση υπολογιστική ισχύς απαιτείται για την εκτέλεση της προσομοίωσης.

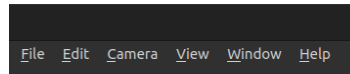
Ο "πραγματικός χρόνος" αναφέρεται στο πραγματικό χρόνο που περνάει στη πραγματική ζωή καθώς τρέχει ο προσομοιωτής. Η σχέση μεταξύ του χρόνου προσομοίωσης και του πραγματικού χρόνου είναι γνωστή ως "παράγοντας πραγματικού χρόνου/real time factor" (RTF). Είναι ο λόγος του χρόνου προσομοίωσης προς τον πραγματικό χρόνο. Ο RTF είναι ένα μέτρο του πόσο γρήγορα ή αργά τρέχει η προσομοίωσή μας σε σύγκριση με τον πραγματικό χρόνο.

Η κατάσταση του κόσμου στο Gazebo υπολογίζεται μία φορά ανά επανάληψη. Μπορούμε να δούμε τον αριθμό των επαναλήψεων στη δεξιά πλευρά της κάτω γραμμής εργαλείων. Κάθε επανάληψη προχωρά την προσομοίωση κατά έναν σταθερό αριθμό δευτερολέπτων, που ονομάζεται μέγεθος βήματος ή step size. Από προεπιλογή, το μέγεθος βήματος είναι 1 ms. Μπορούμε να πατήσουμε το κουμπί παύσης για να διακόψουμε την προσομοίωση και να προχωρήσουμε λίγα βήματα τη φορά χρησιμοποιώντας το κουμπί βήματος.



2.4.5 Το Μενού

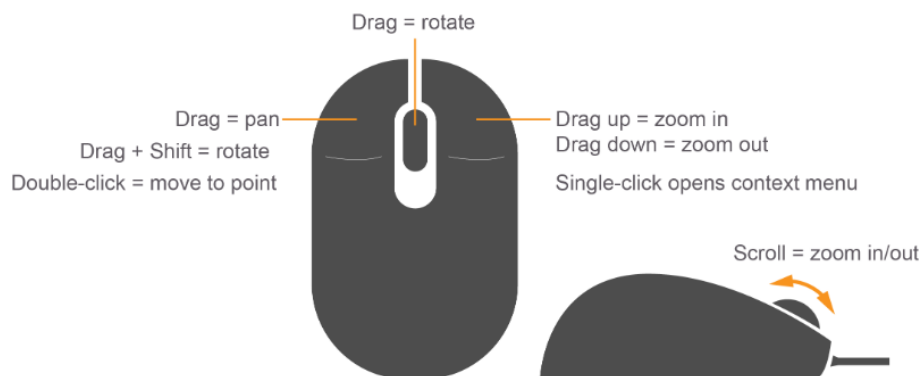
Το Gazebo έχει ένα μενού εφαρμογής στο πάνω μέρος. Ορισμένες από τις επιλογές του μενού υπάρχουν στη Γραμμή Εργαλείων ή ως επιλογές του μενού περιβάλλοντος με δεξί κλικ στη σκηνή.



2.4.6 Χειριστήρια Ποντικιού

Το ποντίκι είναι πολύ χρήσιμο κατά την πλοήγηση στη σκηνή. Παρακάτω είναι οι βασικές λειτουργίες του ποντικιού για πλοήγηση στη σκηνή και αλλαγή της γωνίας προβολής.

Κάνοντας δεξί κλικ σε μοντέλα θα ανοίξει ένα μενού περιβάλλοντος με διάφορες επιλογές.



2.5 Επεξεργαστής Μοντέλου / Model Editor

Τώρα θα κατασκευάσουμε ένα απλό ρομπότ. Θα φτιάξουμε ένα όχημα με ρόδες και θα προσθέσουμε έναν αισθητήρα που θα επιτρέπει στο ρομπότ να ακολουθεί ένα αντικείμενο.

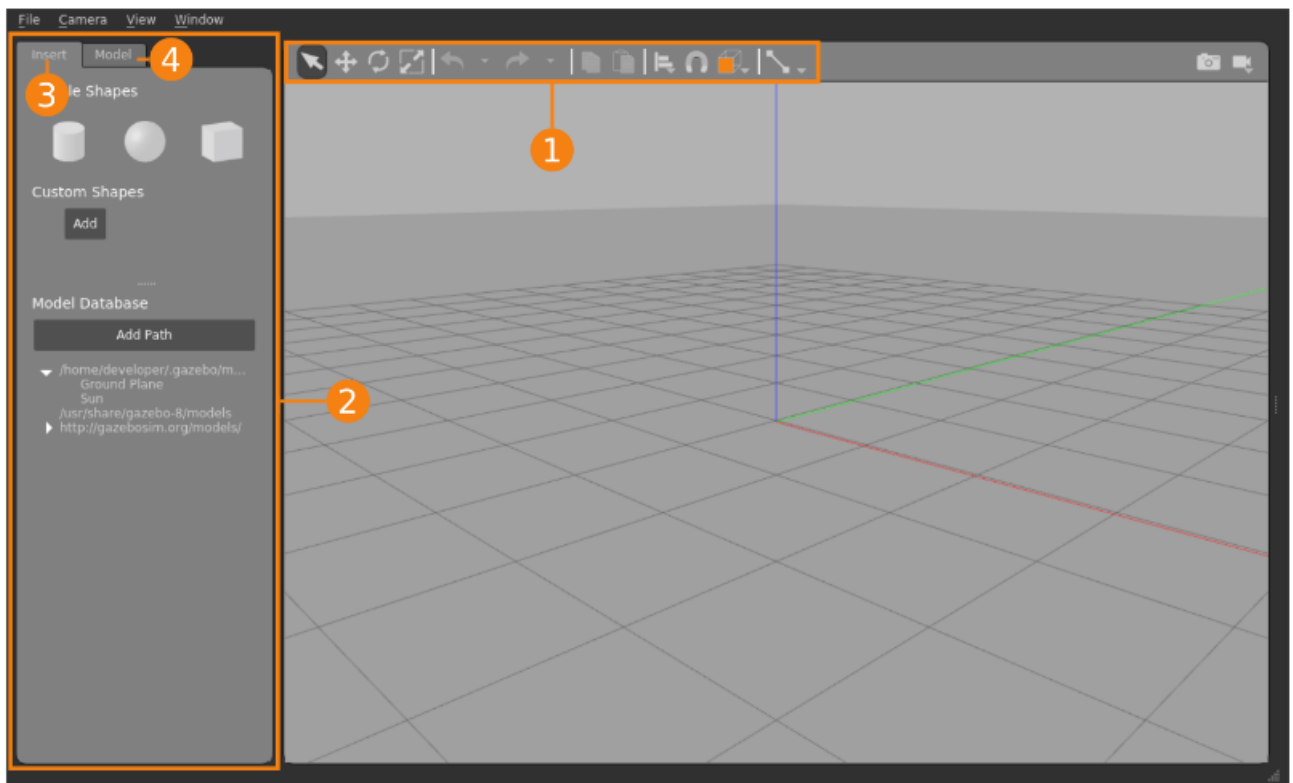
Προσομοίωση Ρομπότ μέσω Υπολογιστή

Ο Επεξεργαστής Μοντέλου μας επιτρέπει να κατασκευάσουμε απλά μοντέλα απευθείας στη Γραφική Διεπαφή Χρήστη (GUI).

2.5.1 Διεπαφή Χρήστη του Επεξεργαστή Μοντέλου / Model Editor User Interface

Για να εισέλθουμε στο Model Editor, κάνουμε κλικ στο Edit από τη γραμμή μενού και επιλέγουμε Model Editor. Εναλλακτικά, χρησιμοποιούμε το συνδυασμό πλήκτρων Ctrl+M. Η φυσική και η προσομοίωση θα διακοπούν αμέσως μόλις μπούμε στο Model Editor.

Η διεπαφή του Model Editor μοιάζει με την κύρια διεπαφή του Gazebo, αλλά έχει κάποιες μικρές διαφορές. Το αριστερό πάνελ και η πάνω γραμμή εργαλείων περιέχουν τώρα μόνο εργαλεία για την επεξεργασία και τη δημιουργία μερών του μοντέλου. Η κάτω γραμμή εργαλείων που εμφανίζει δεδομένα προσομοίωσης είναι κρυφή καθώς η προσομοίωση έχει σταματήσει.



1. **Γραμμή Εργαλείων/Toolbar** - Περιέχει εργαλεία για την επεξεργασία του μοντέλου
2. **Παλέτα/Palette** - Γνωστό και ως Αριστερό Πάνελ. Έχει δύο καρτέλες για την επεξεργασία του μοντέλου.
3. **Καρτέλα Insert/Insert Tab** - Εργαλεία για την προσθήκη συνδέσμων και ενσωματωμένων μοντέλων
4. **Καρτέλα Model/Model Tab** - Επιτρέπει την επεξεργασία των ιδιοτήτων και του περιεχομένου του μοντέλου

2.5.1.1 Παλέτα (Αριστερό Πάνελ)

Η Παλέτα έχει δύο καρτέλες:

- **INSERT:** Η καρτέλα Insert είναι εκεί όπου προσθέτουμε νέα μέρη (συνδέσμους και μοντέλα) στο Model Editor. Υπάρχουν τρεις ενότητες:
 - **Απλά Σχήματα:** Αυτά είναι πρωτογενή γεωμετρικά σχήματα που μπορούμε να εισάγουμε για να σχηματίσουμε έναν σύνδεσμο στο μοντέλο.
 - **Προσαρμοσμένα Σχήματα:** Το κουμπί Add μας επιτρέπει να εισάγουμε

Προσομοίωση Ρομπότ μέσω Υπολογιστή

προσαρμοσμένα πλέγματα για τη δημιουργία συνδέσμου στο μοντέλο.

Υποστηρίζονται τα αρχεία COLLADA (.dae), 3D Systems (.stl), Wavefront (.obj) και W3C SVG (.svg).

- **Βάση Δεδομένων Μοντέλων:** Έχει μια λίστα μοντέλων που μπορούμε να εισάγουμε στο Model Editor με τον ίδιο τρόπο όπως τα απλά σχήματα. Όταν εισαχθούν, ονομάζονται Ενσωματωμένα Μοντέλα/Nested Models.
- **MODEL:** Η καρτέλα Model μας επιτρέπει να ορίσουμε το όνομα και τις βασικές παραμέτρους του μοντέλου που κατασκευάζουμε. Εμφανίζει μια λίστα με τους συνδέσμους, τις αρθρώσεις, τα ενσωματωμένα μοντέλα και τα πρόσθετα που υπάρχουν στο μοντέλο. Οι παράμετροι μπορούν να τροποποιηθούν χρησιμοποιώντας τον Επιθεωρητή Συνδέσμων/Link Inspector.

2.5.1.2 Γραμμή Εργαλείων

Όπως στη λειτουργία Προσομοίωσης/Simulation mode, η κύρια Γραμμή Εργαλείων στον Model Editor περιλαμβάνει εργαλεία για την αλληλεπίδραση με τα αντικείμενα στη σκηνή.

Τα διαθέσιμα εργαλεία είναι επιλογή, μετάφραση, κλιμάκωση, περιστροφή, αναίρεση & επανάληψη, αντιγραφή & επικόλληση, ευθυγράμμιση, στερέωση, προσαρμογή προβολής και δημιουργία αρθρώσεων.

2.5.2 Περιορισμοί

Ο Επεξεργαστής Μοντέλου υποστηρίζει τις περισσότερες από τις βασικές εργασίες κατασκευής μοντέλων που μπορούν να γίνουν γράφοντας SDF. Ωστόσο, υπάρχουν ορισμένες δυνατότητες που δεν είναι ακόμη διαθέσιμες:

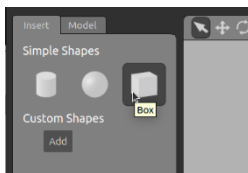
- επεξεργασία ενσωματωμένων μοντέλων και συνδέσμων μέσα σε ενσωματωμένα μοντέλα,
- προσθήκη και επεξεργασία ορισμένων τύπων γεωμετρίας, συμπεριλαμβανομένων των επιπέδων (Plane) και των πολυγραμμών (Polyline),
- υποστήριξη για χάρτες ύψους (heightmaps),
- λειτουργίες CAD.

2.6 Κατασκευή Οχήματος

Θα δημιουργήσουμε βήμα προς βήμα ένα μοντέλο απλού οχήματος στον Επεξεργαστή Μοντέλου.

2.6.1 Σασί

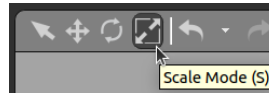
1. Ξεκινάμε δημιουργώντας το σασί του οχήματος. Στην καρτέλα Insert στο αριστερό πάνελ, κάνουμε κλικ μία φορά στο εικονίδιο του κουτιού και το τοποθετούμε στη Σκηνή.



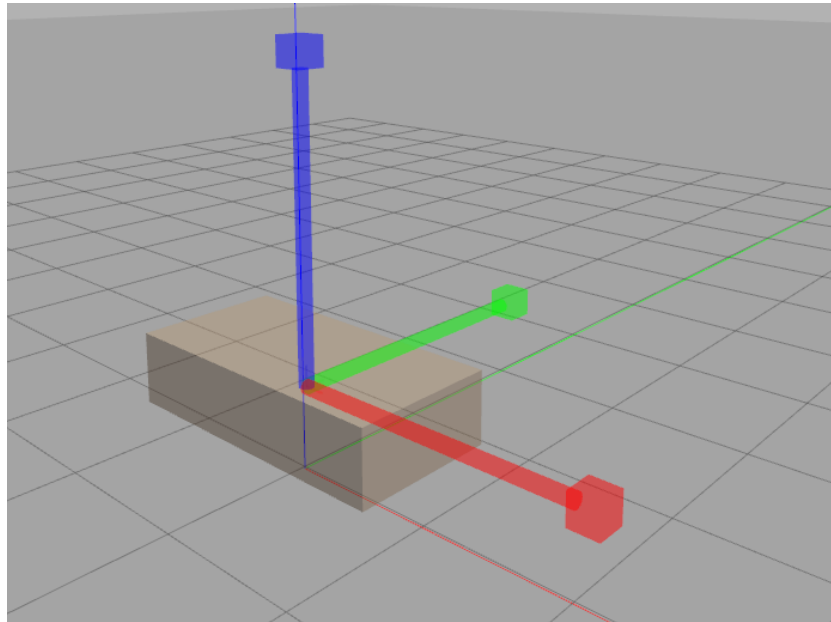
2. Έπειτα αλλάζουμε το μέγεθος του κουτιού ώστε να μοιάζει περισσότερο με το σχήμα του σασί ενός αυτοκινήτου. Μπορούμε να το κάνουμε αυτό επιλέγοντας το εργαλείο Scale που βρίσκεται στην επάνω Γραμμή Εργαλείων. Επιλέγουμε το κουτί στη Σκηνή και εμφανίζεται ένας δείκτης RGB πάνω από το κουτί. Ο κόκκινος δείκτης αντιπροσωπεύει τον άξονα X, ο πράσινος τον Y και ο μπλε τον Z. Μετακινούμε το ποντίκι πάνω από τον κόκκινο δείκτη για να τον επισημάνουμε, και στη συνέχεια κάνουμε κλικ και σύρουμε για να κάνουμε το σασί μακρύτερο κατά μήκος του άξονα X. Κλιμακώνουμε το σασί ώστε να είναι περίπου 2 μέτρα

Προσομοίωση Ρομπότ μέσω Υπολογιστή

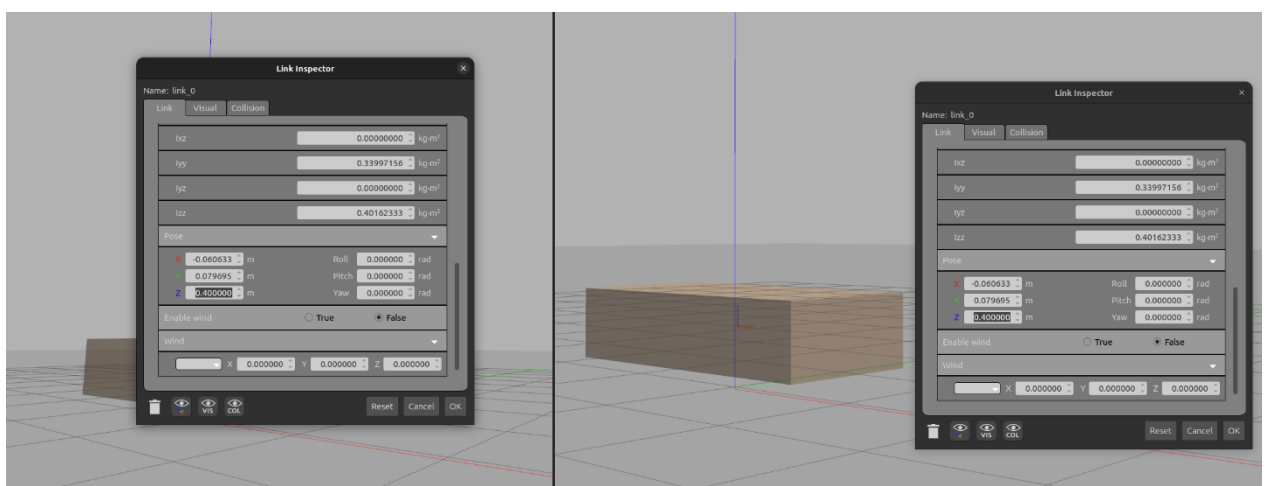
μακρύ. Μπορούμε να το εκτιμήσουμε κοιτάζοντας τα πλέγματα 1x1 μέτρου στο έδαφος.



3. Στη συνέχεια, ισοπεδώνουμε το σασί χρησιμοποιώντας το εργαλείο Scale. Κάνουμε κλικ και σύρουμε τον μπλε δείκτη προς τα κάτω, ώστε το σασί να γίνει περίπου το μισό από το αρχικό του μέγεθος.



4. Θέλουμε να χαμηλώσουμε το σασί πιο κοντά στο έδαφος. Για να δώσουμε ακριβείς μετρήσεις, θα χρησιμοποιήσουμε τον Επιθεωρητή Συνδέσμων (Link Inspector). Κάνουμε διπλό κλικ στο κουτί για να ανοίξουμε τον Επιθεωρητή (Inspector). Κάνουμε κύλιση προς τα κάτω στην καρτέλα Link για να βρούμε τις παραμέτρους Θέσης (Pose) και αλλάζουμε τον άξονα Z σε 0,4m. Στη συνέχεια, κάνουμε κλικ έξω από το πλαίσιο (χωρίς να πατήσουμε Enter). Κάνουμε κλικ στο OK για να αποθηκεύσουμε τις αλλαγές και κλείνουμε τον Επιθεωρητή.

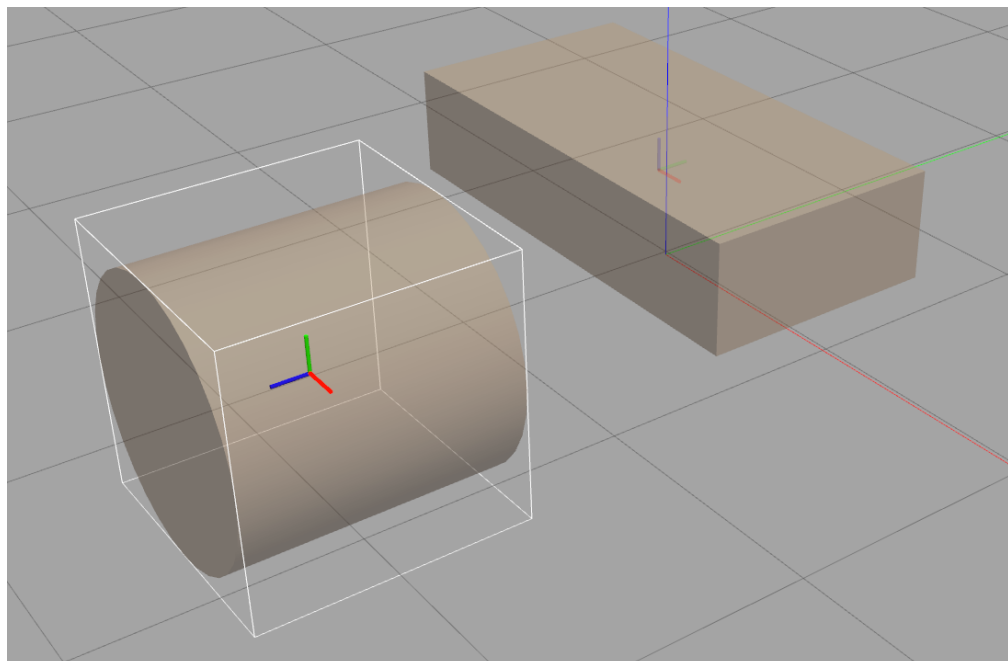
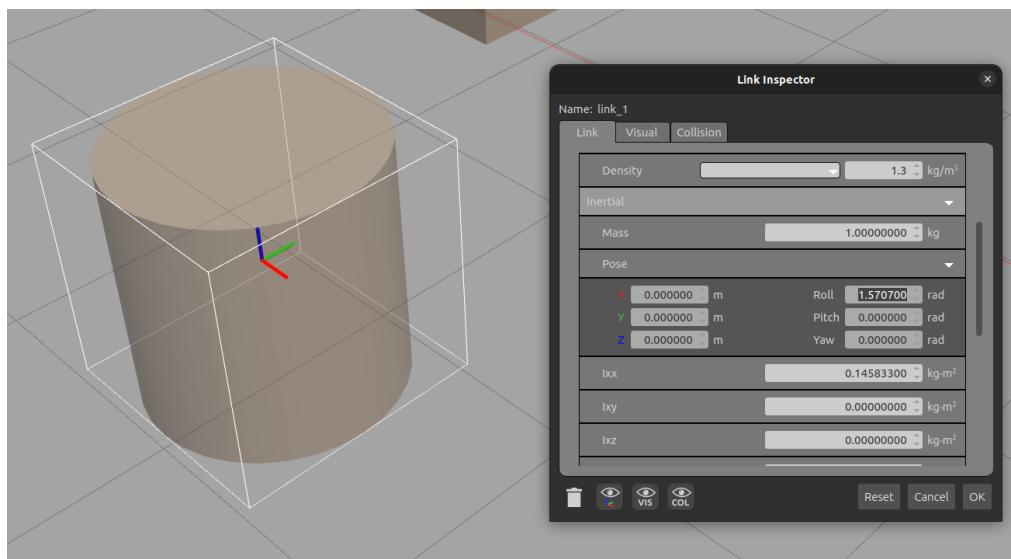
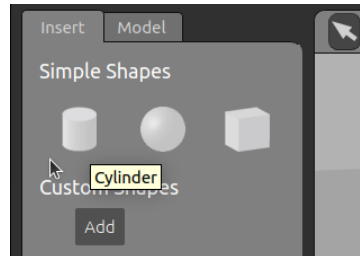


2.6.2 Μπροστινοί Τροχοί

1. Συνεχίζουμε με τους μπροστινούς τροχούς. Ξεκινάμε εισάγοντας έναν κύλινδρο από τη καρτέλα Insert στο αριστερό πάνελ.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

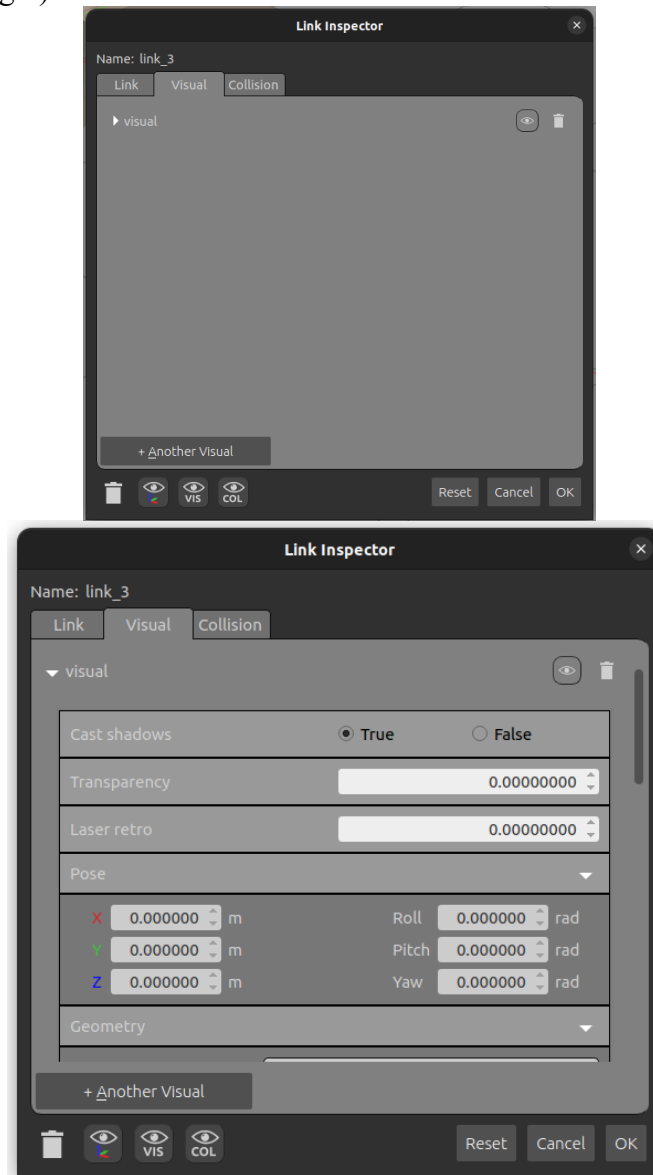
2. Ο κύλινδρος στην προεπιλεγμένη του θέση δεν θα κυλάει καλά. Ας τον περιστρέψουμε κατά μήκος του άξονα X χρησιμοποιώντας τον Επιθεωρητή Συνδέσμων (Link Inspector). Κάνουμε διπλό κλικ στον κύλινδρο, κάνουμε κύλιση προς την ενότητα Θέσης (Pose) στο κάτω μέρος και αλλάζουμε την Περιστροφή (Roll) σε 1.5707 ακτίνια (90 μοίρες). Στη συνέχεια, κάνουμε κλικ έξω από το πλαίσιο.

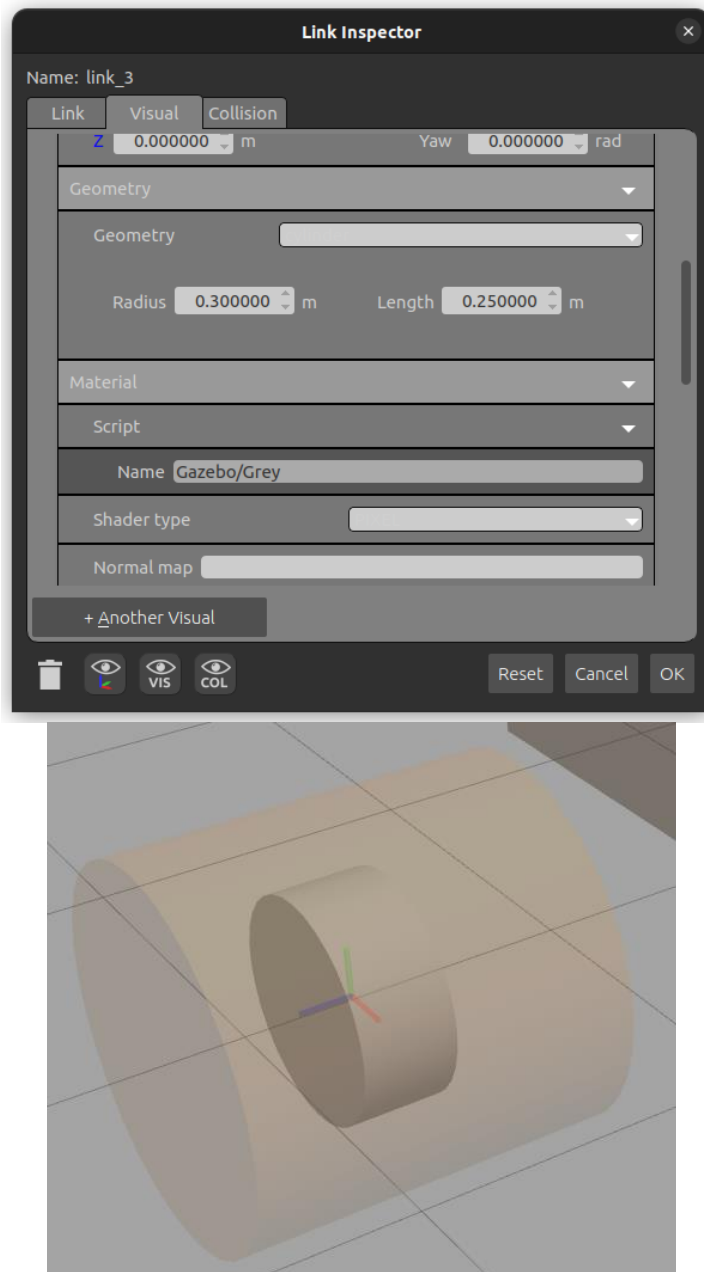


3. Αλλάζουμε το μέγεθος του τροχού δίνοντας ακριβείς διαστάσεις. Πηγαίνουμε στη καρτέλα Visual για να δούμε τη λίστα με τα οπτικά στοιχεία αυτού του συνδέσμου. Θα πρέπει να υπάρχει μόνο ένα. Επεκτείνουμε το οπτικό στοιχείο κάνοντας κλικ στο μικρό

Προσομοίωση Ρομπότ μέσω Υπολογιστή

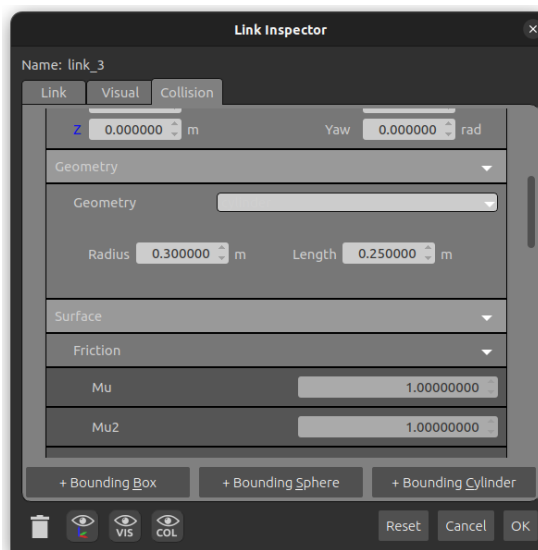
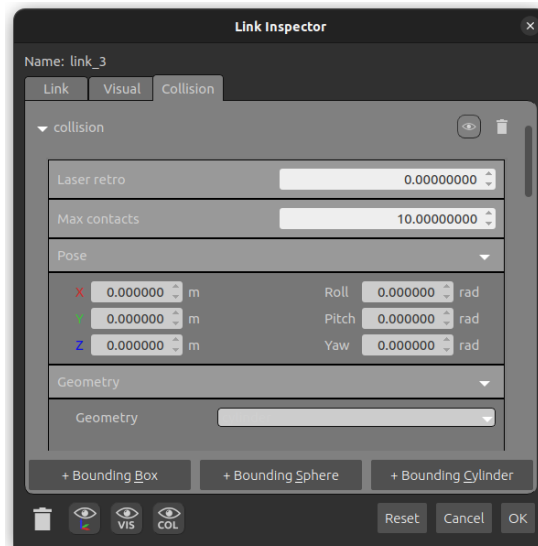
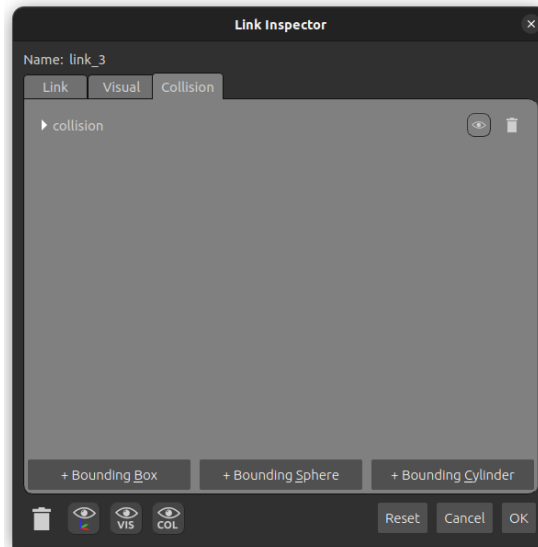
βελάκι δίπλα στην ετικέτα του οπτικού στοιχείου. Κάνουμε κύλιση προς τα κάτω στην ενότητα Γεωμετρίας (Geometry) και αλλάζουμε την Ακτίνα (Radius) σε 0.3m και το Μήκος (Length) σε 0.25m.



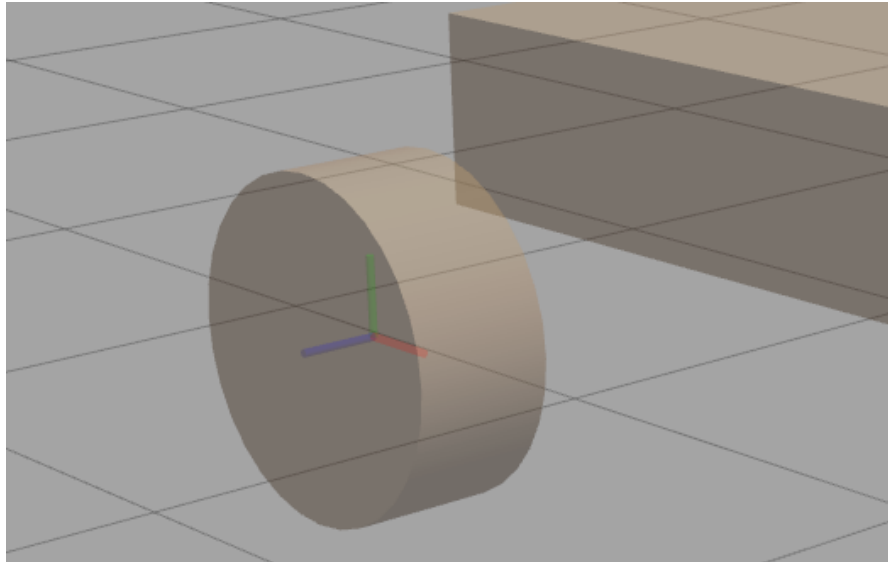


4. Τώρα θα πρέπει να βλέπουμε έναν μικρότερο κύλινδρο μέσα σε έναν μεγαλύτερο. Αυτό είναι αναμενόμενο, καθώς έχουμε αλλάξει μόνο τη γεωμετρία του οπτικού στοιχείου και όχι την περιοχή σύγκρουσης (collision). Το "visual" είναι η γραφική αναπαράσταση του συνδέσμου και δεν επηρεάζει τη προσομοίωση της φυσικής. Από την άλλη, η "περιοχή σύγκρουσης" χρησιμοποιείται από τη μηχανή φυσικής για τον έλεγχο συγκρούσεων. Για να ενημερώσουμε επίσης την περιοχή σύγκρουσης του τροχού, πηγαίνουμε στην καρτέλα Collision, επεκτείνουμε το μοναδικό στοιχείο σύγκρουσης και εισάγουμε τις ίδιες διαστάσεις γεωμετρίας: Ακτίνα (Radius): 0.3m και Μήκος (Length): 0.25m. Κάνουμε κλικ στο OK για να αποθηκεύσουμε τις αλλαγές και να κλείσουμε τον Επιθεωρητή.

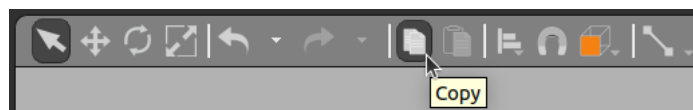
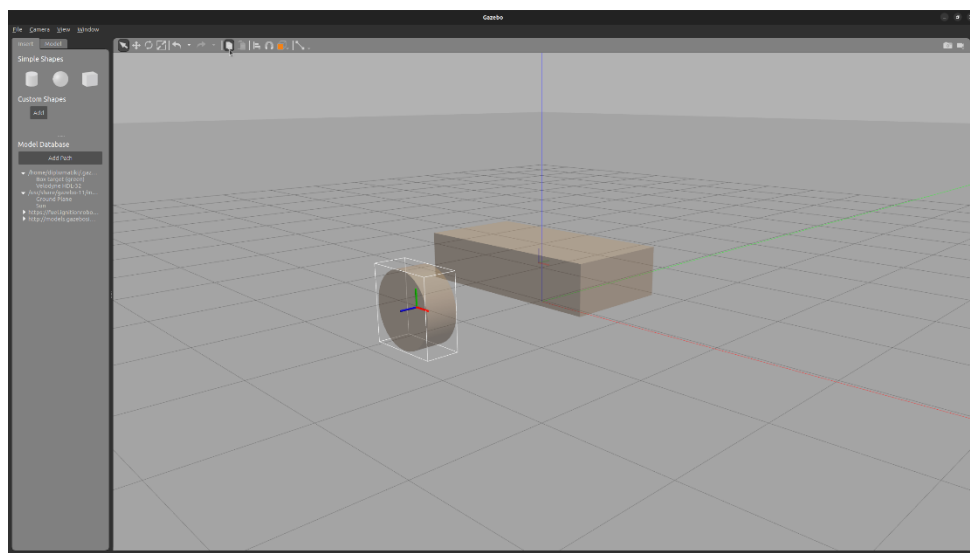
Προσομοίωση Ρομπότ μέσω Υπολογιστή



Προσομοίωση Ρομπότ μέσω Υπολογιστή

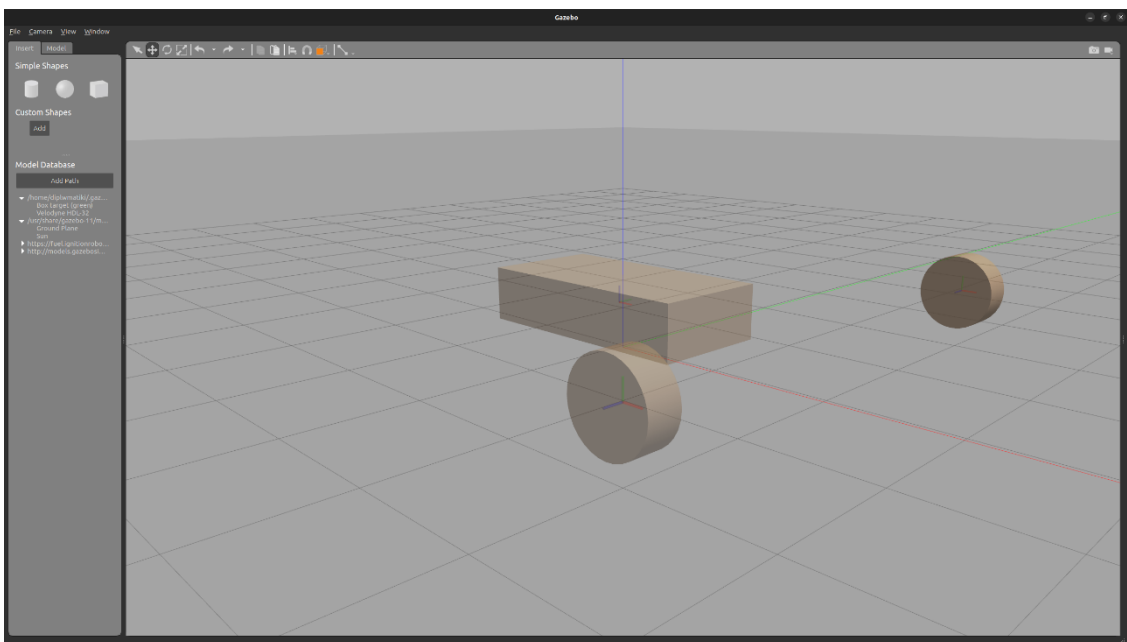
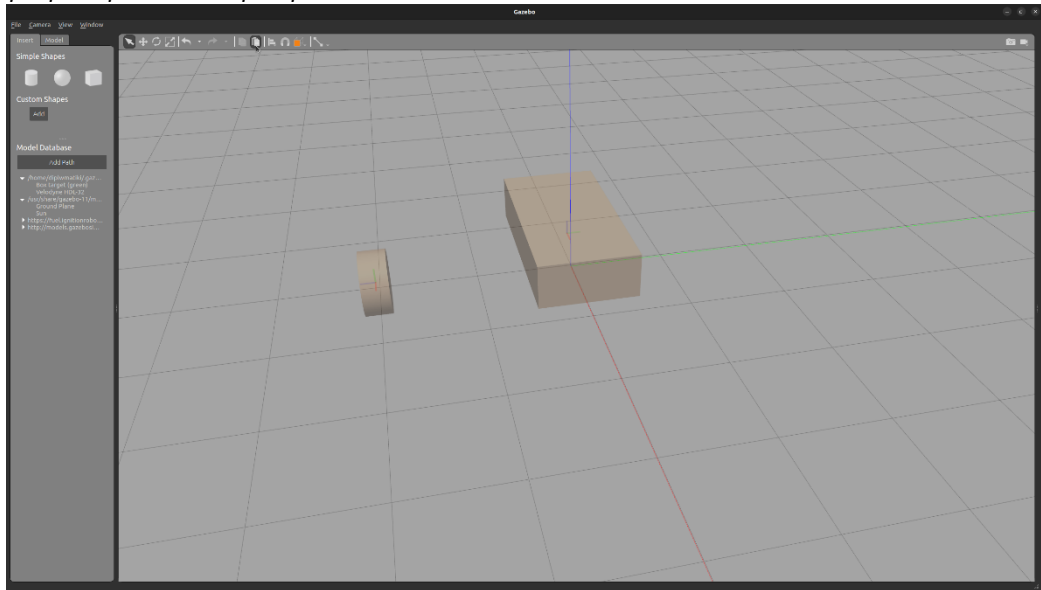


5. Αφού δημιουργήσαμε τον πρώτο τροχό, θα τον χρησιμοποιήσουμε ως πρότυπο για να φτιάξουμε έναν ακόμα. Επιλέγουμε τον τροχό και κάνουμε κλικ στο εικονίδιο Αντιγραφή (Copy) στην επάνω γραμμή εργαλείων.



6. Κάνουμε κλικ στο εικονίδιο Επικόλληση (Paste) και μετακινούμε το ποντίκι πίσω στη Σκηνή για να εισάγουμε το αντίγραφο του τροχού.

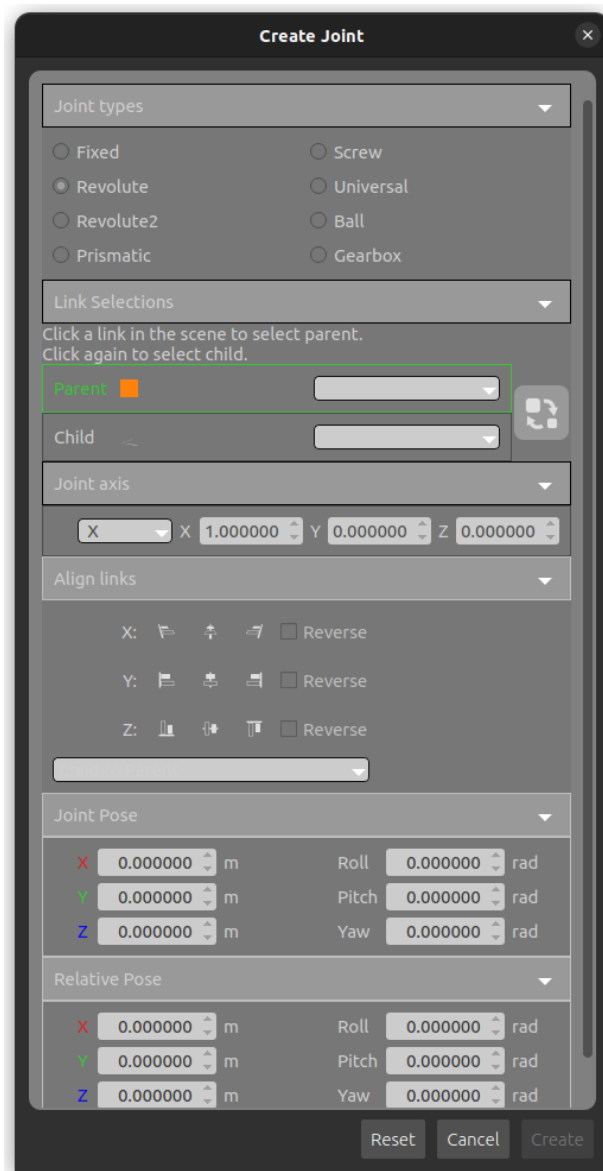
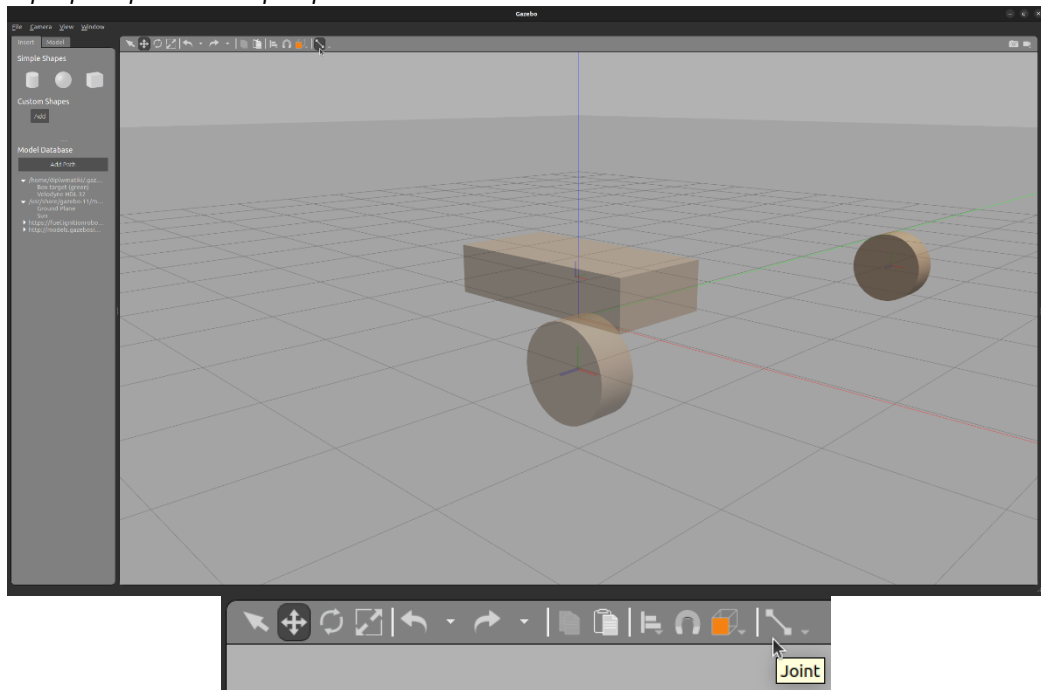
Προσομοίωση Ρομπότ μέσω Υπολογιστή



Ας βεβαιωθούμε τώρα ότι το όχημα θα κινηθεί σωστά ευθυγραμμίζοντας το σασί κατά μήκος του θετικού άξονα X (ο κόκκινος δείκτης στη σκηνή). Όταν προσθέτουμε τους τροχούς στο επόμενο βήμα, βεβαιωνόμαστε ότι βρίσκονται στο άκρο του οχήματος που εκτείνεται κατά μήκος του θετικού άξονα X.

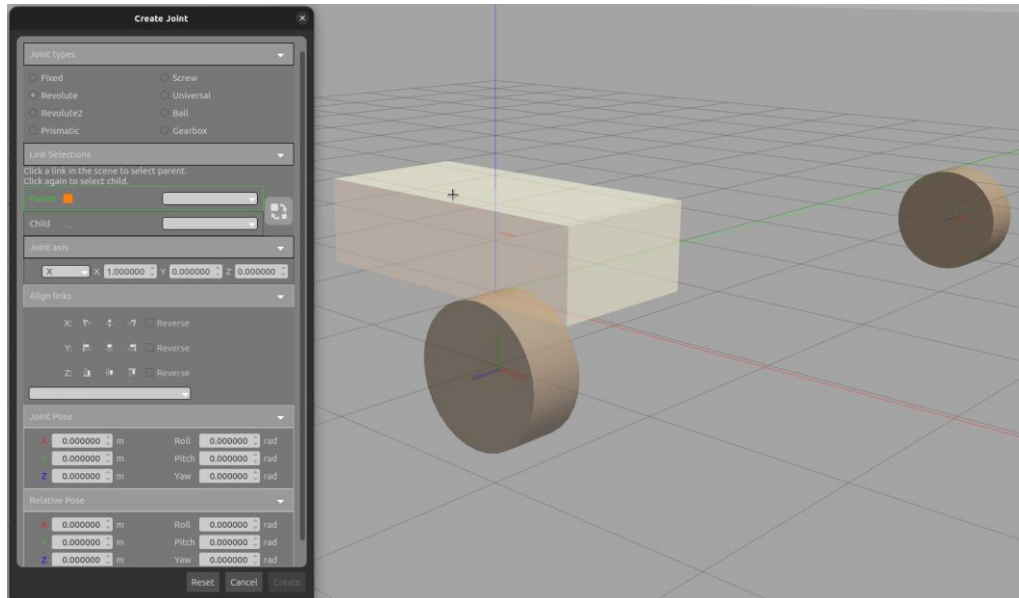
7. Το σασί και οι τροχοί είναι προς το παρόν ελεύθερα κινούμενα σώματα. Για να περιορίσουμε την κίνησή τους, θα προσθέσουμε αρθρώσεις (joints) μεταξύ κάθε τροχού και του σασί. Κάνουμε κλικ στο εικονίδιο **Joint** στην επάνω γραμμή εργαλείων για να ανοίξουμε τον διάλογο **Δημιουργίας Αρθρώσεων (Joint Creation)**.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

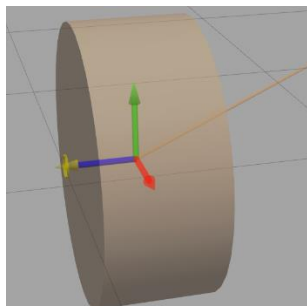
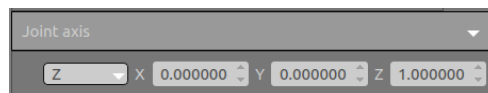


Προσομοίωση Ρομπότ μέσω Υπολογιστή

8. Το παράθυρο διαλόγου Δημιουργίας Αρθρώσεων/Create Joint περιέχει τις ιδιότητες της άρθρωσης που συνήθως καθορίζονται για μια άρθρωση. Πριν ρυθμίσουμε οποιαδήποτε από τις ιδιότητες, μας ζητείται να επιλέξουμε τους συνδέσμους γονέα και παιδιού της άρθρωσης. Μετακινούμε το ποντίκι πάνω από το σασί στη σκηνή για να το δούμε να επισημαίνεται και κάνουμε κλικ πάνω του για να το ορίσουμε ως τον γονέα της άρθρωσης.



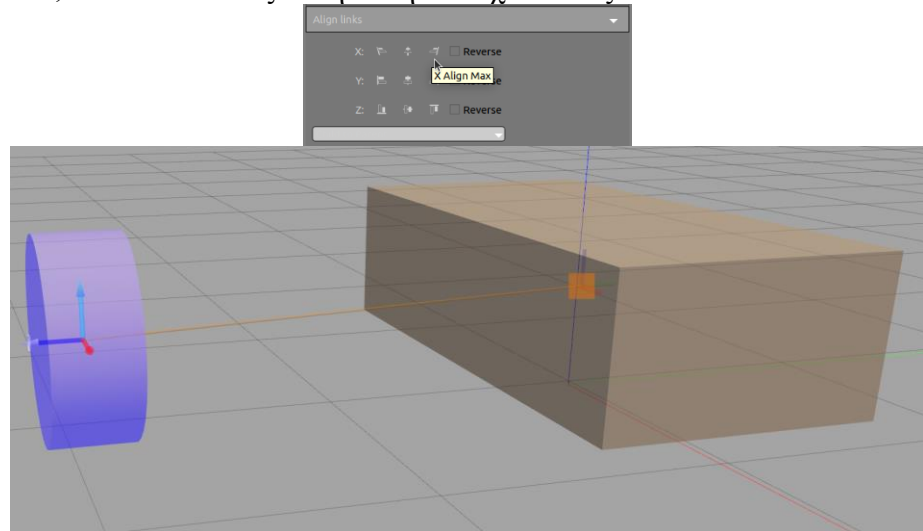
9. Στη συνέχεια, μετακινούμε το ποντίκι στον μπροστινό αριστερό τροχό. Μια γραμμή θα πρέπει τώρα να εκτείνεται από την αρχή του σασί μέχρι την άκρη του ποντικιού. Κάνουμε κλικ στον τροχό για να τον ορίσουμε ως παιδί της άρθρωσης. Μια νέα άρθρωση δημιουργείται. Από προεπιλογή, είναι μια περιστροφική άρθρωση (revolute joint), όπως φαίνεται στην ενότητα Τύποι Αρθρώσεων (Joint Types) στο παράθυρο διαλόγου, που είναι ο τύπος άρθρωσης που χρειαζόμαστε.
10. Τώρα, πρέπει να ρυθμίσουμε τον άξονα περιστροφής του τροχού. Στο παράθυρο διαλόγου Δημιουργίας Αρθρώσεων (Joint Creation), βρίσκουμε την ενότητα Άξονας Αρθρώσεων (Joint Axis) και αλλάζουμε τον άξονα σε Z (0, 0, 1). Δίνουμε προσοχή στην οπτική ένδειξη RGB της άρθρωσης πάνω στον τροχό. Θα πρέπει να δούμε έναν κίτρινο δακτύλιο να εμφανίζεται πάνω από το μπλε βέλος της οπτικής άρθρωσης, υποδεικνύοντας ότι αυτός είναι ο άξονας περιστροφής.



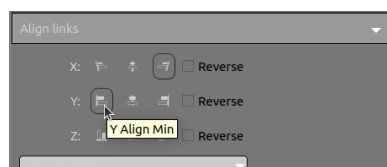
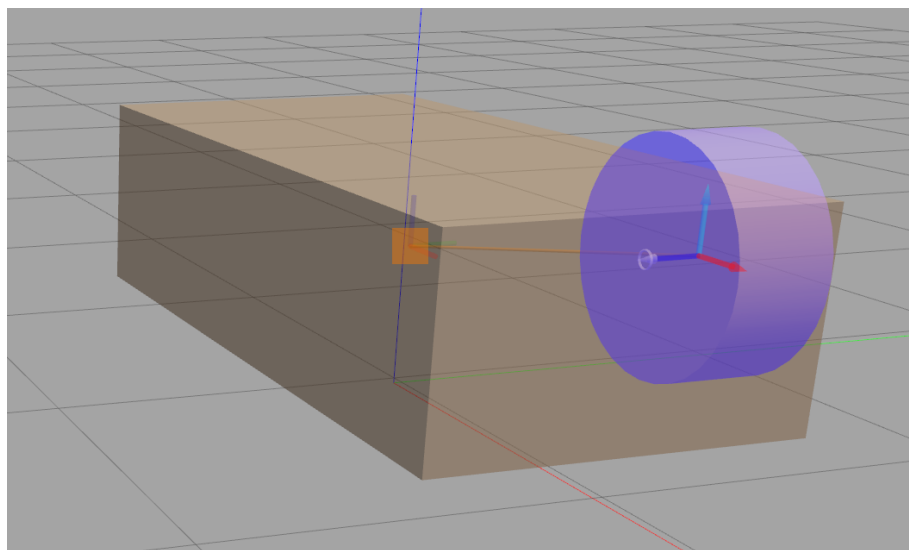
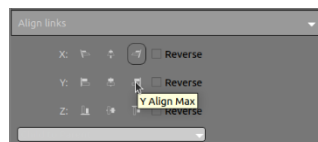
11. Για να ευθυγραμμίσουμε το τροχό δίπλα στο σασί, θα χρησιμοποιήσουμε τις διάφορες επιλογές ευθυγράμμισης στην ενότητα Ευθυγράμμιση Συνδέσμων (Align Links) στο παράθυρο διαλόγου Δημιουργίας Αρθρώσεων (Joint Creation). Πρώτα θα

Προσομοίωση Ρομπότ μέσω Υπολογιστή

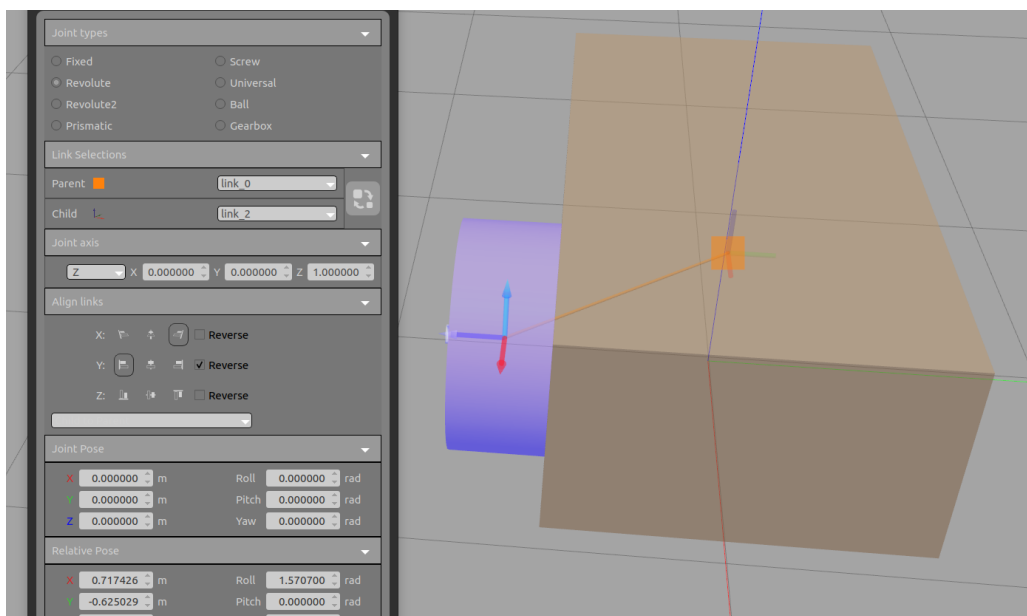
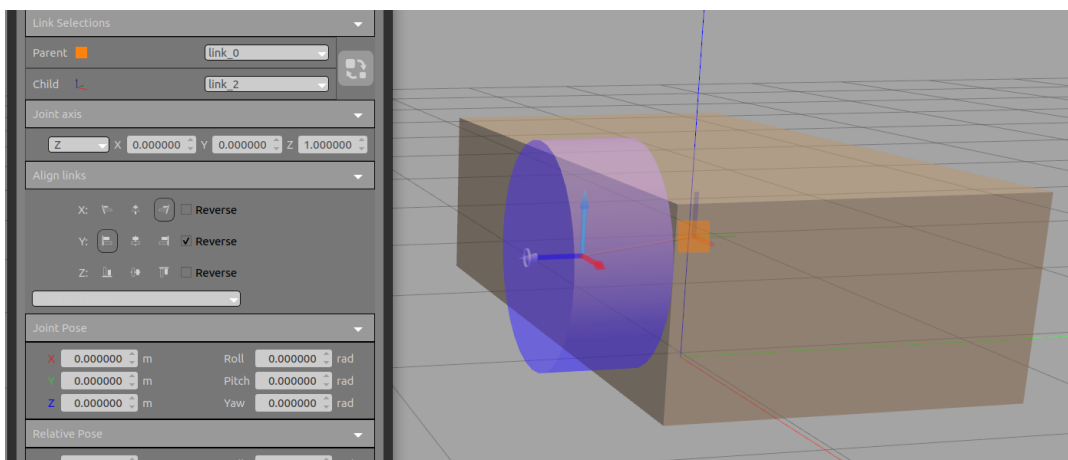
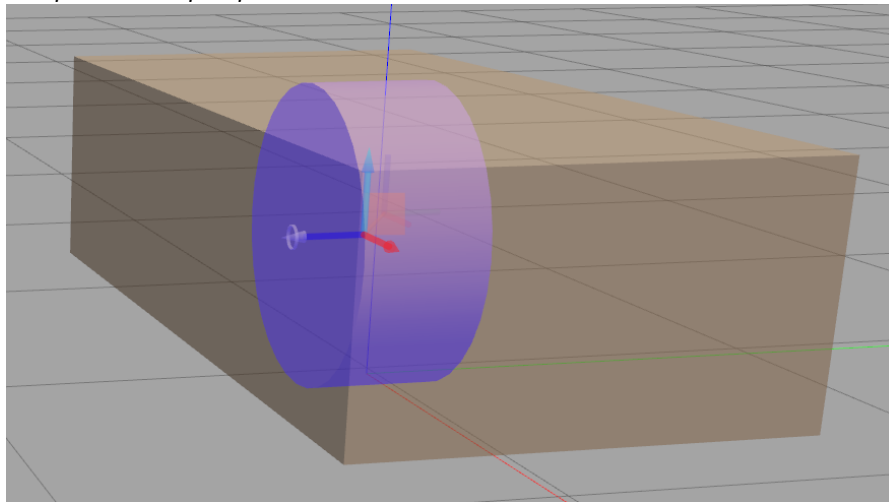
ευθυγραμμίσουμε κατά μήκος του άξονα X, οπότε κάνουμε κλικ στην επιλογή X Align Max για να δούμε το αποτέλεσμα της ευθυγράμμισης. Ο κύλινδρος (τροχός) θα πρέπει να επισημανθεί, υποδεικνύοντας ότι η θέση του έχει αλλάξει.



12. Στο παράδειγμά μας, θέλουμε να τοποθετήσουμε τον τροχό ακριβώς δίπλα στο σασί. Για να φέρουμε το τροχό πιο κοντά, κάνουμε κλικ στην επιλογή Y Align Max. Ωστόσο, αυτό δεν είναι ακόμα το επιθυμητό αποτέλεσμα. Κάνουμε κλικ στην επιλογή Reverse δίπλα στις επιλογές ευθυγράμμισης του άξονα Y για να ευθυγραμμίσουμε το ελάχιστο του τροχού (αντίστροφο του μέγιστου) με το μέγιστο του σασί. Σημειώνουμε ότι η επιλογή Reverse εφαρμόζεται στο σύνδεσμο παιδιού, καθώς η προεπιλεγμένη διαμόρφωση ευθυγράμμισης που εμφανίζεται στη λίστα είναι Child to Parent. Αν η διαμόρφωση ήταν Parent to Child, τότε η επιλογή Reverse θα εφαρμοζόταν στον σύνδεσμο γονέα. Τέλος, πατάμε Create για να ολοκληρώσουμε τη δημιουργία της άρθρωσης.

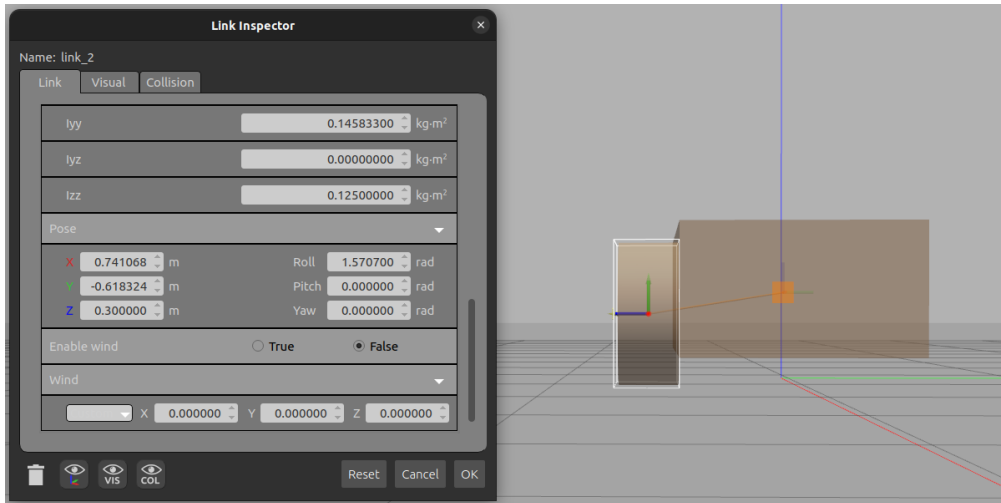
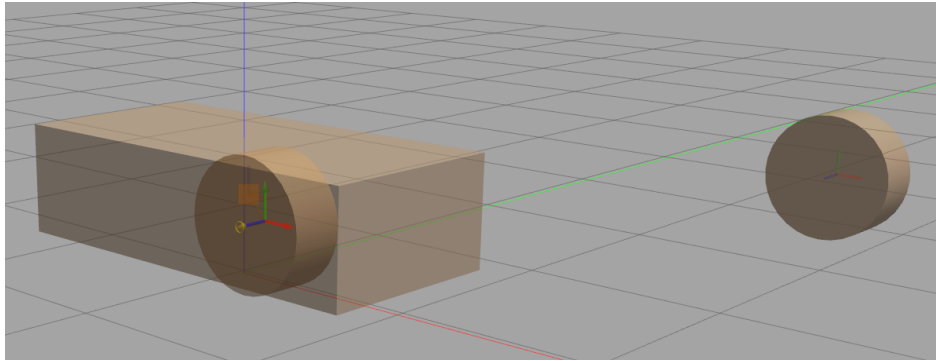


Προσομοίωση Ρομπότ μέσω Υπολογιστή

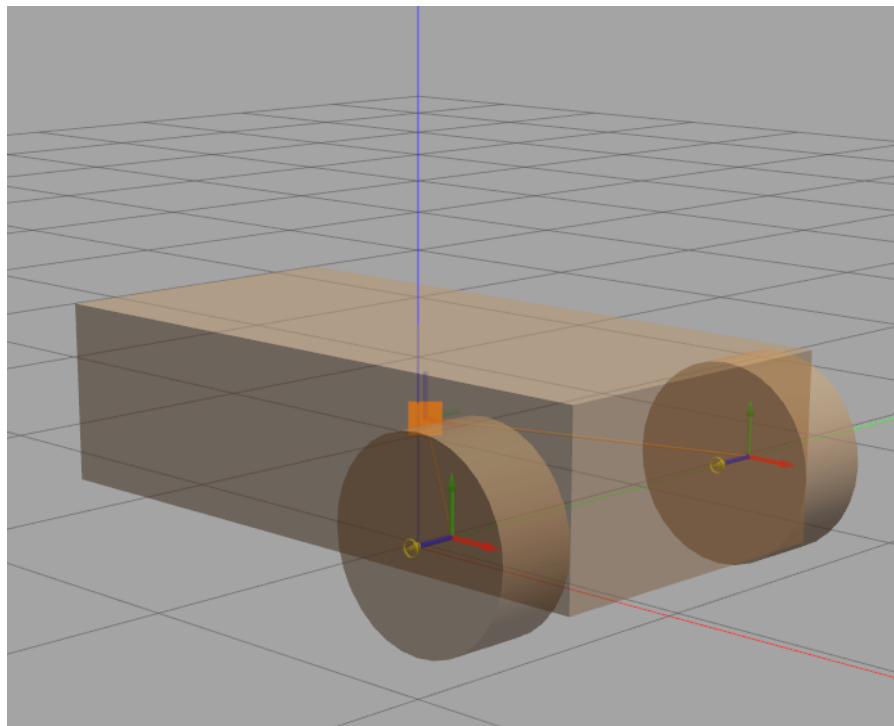
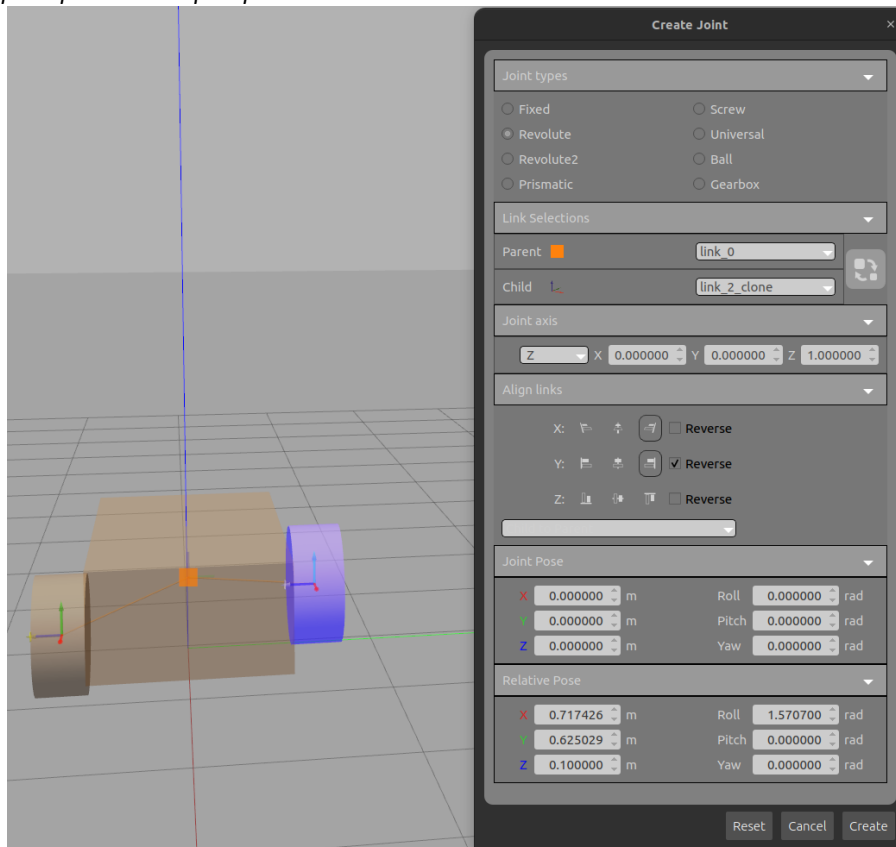


13. Για να τοποθετήσουμε τον τροχό πάνω από το έδαφος, ανοίγουμε τον Επιθεωρητή Συνδέσμων (Link Inspector) κάνοντας διπλό κλικ στο τροχό. Μπορούμε να χρησιμοποιήσουμε την ενότητα Pose στο κάτω μέρος του διαλόγου για να μετακινήσουμε τον τροχό. Δεδομένου ότι ο τροχός έχει ακτίνα 0.3m, αλλάζουμε τη θέση Z σε 0.3m για να τοποθετηθεί στο έδαφος και πατάμε OK.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

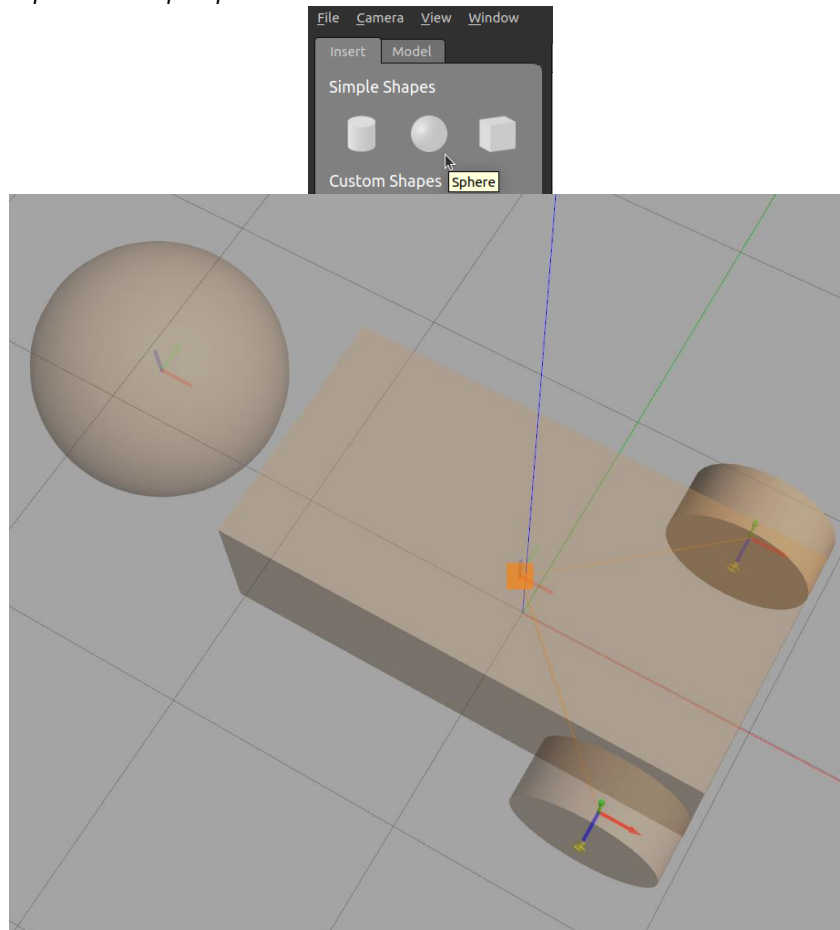


14. Στη συνέχεια, επαναλαμβάνουμε τη διαδικασία δημιουργίας άρθρωσης και τη διαμόρφωση του άξονα για τον άλλον μπροστινό τροχό, διασφαλίζοντας ότι α) το σασί είναι ο γονέας της άρθρωσης και ο τροχός είναι το παιδί, β) ο άξονας περιστροφής έχει ρυθμιστεί στον άξονα Z και γ) χρησιμοποιούμε την επιλογή Y Align Min για να ευθυγραμμίσουμε τον δεξιό τροχό, καθώς βρίσκεται στην άλλη πλευρά του σασί.

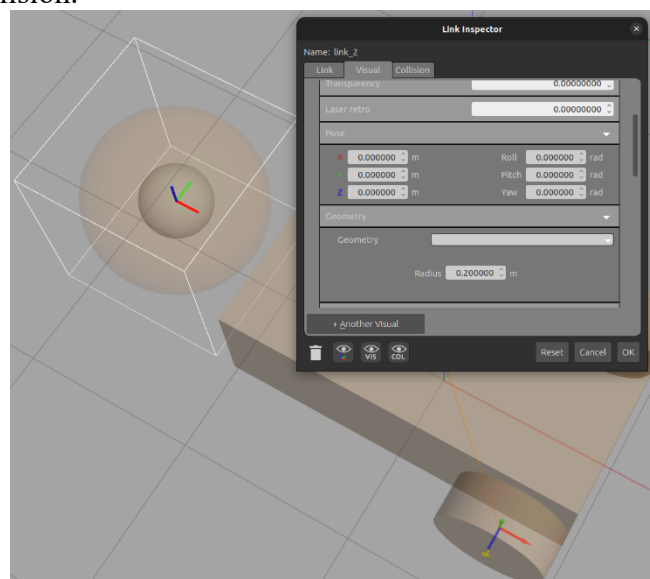


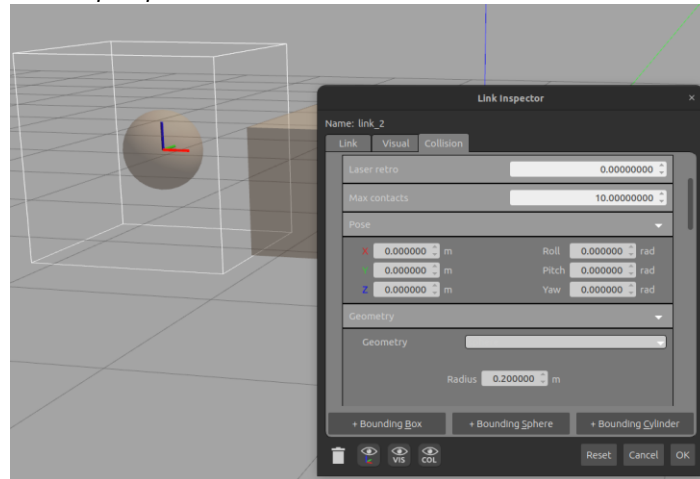
2.6.3 Περιστρεφόμενος Τροχός

1. Για να δημιουργήσουμε έναν περιστρεφόμενο τροχό (caster wheel) για το όχημα, κάνουμε κλικ στο κουμπί Σφαίρα (Sphere) στο Αριστερό Πάνελ και την εισάγουμε στη Σκηνή.

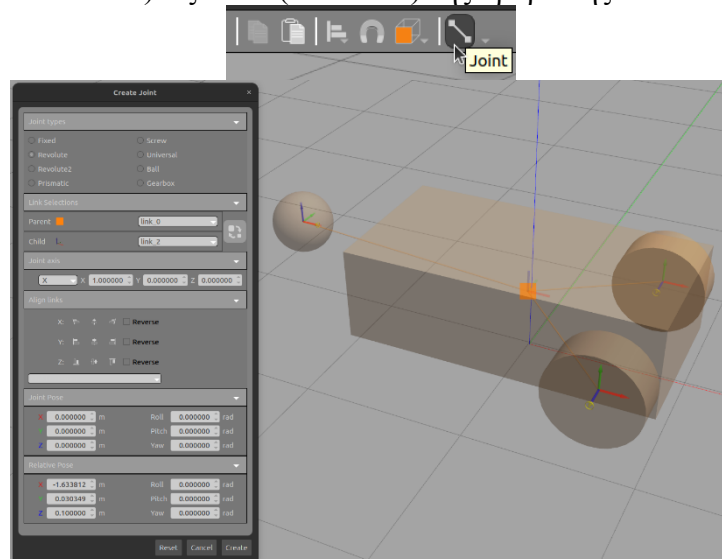


2. Αλλάζουμε το μέγεθος της σφαίρας δίνοντάς της ακριβείς διαστάσεις, όπως κάναμε και για τους μπροστινούς τροχούς. Πηγαίνουμε στην καρτέλα Visual για να δούμε τη λίστα των οπτικών στοιχείων αυτού του συνδέσμου. Επεκτείνουμε το μοναδικό οπτικό στοιχείο. Κάνουμε κύλιση προς τα κάτω στην ενότητα Geometry και αλλάζουμε την Ακτίνα (Radius) σε 0.2m. Επίσης, επιβεβαιώνουμε ότι έχουμε κάνει το ίδιο για το collision στην καρτέλα Collision.

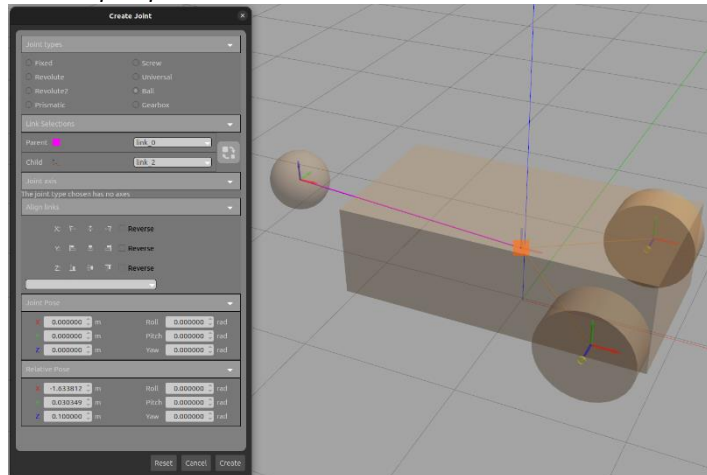




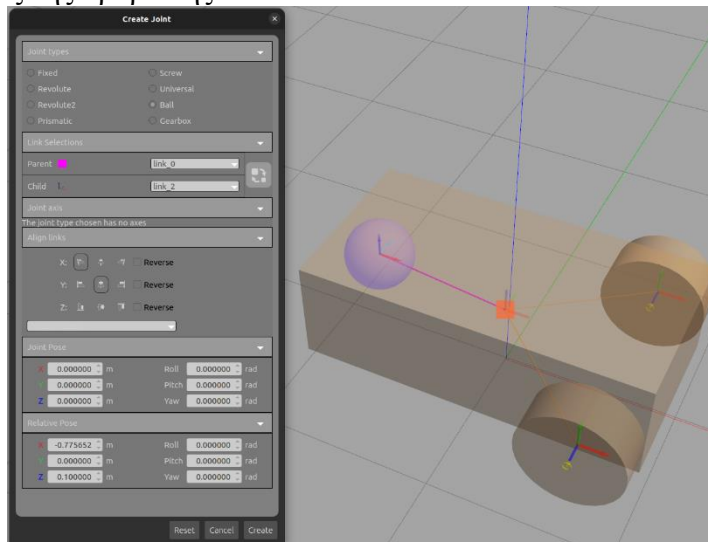
3. Για να δημιουργήσουμε μια άρθρωση (joint) μεταξύ του περιστρεφόμενου τροχού (caster wheel) και του σασί ανοίγουμε το παράθυρο διαλόγου Δημιουργίας Αρθρώσεων (Joint Creation) κάνοντας κλικ στο εικονίδιο Joint στην επάνω γραμμή εργαλείων μετακινούμε το ποντίκι στη Σκηνή και επιλέγουμε το σασί ως γονέα (parent link) της άρθρωσης και τη σφαίρα (caster wheel) ως παιδί (child link) της άρθρωσης.



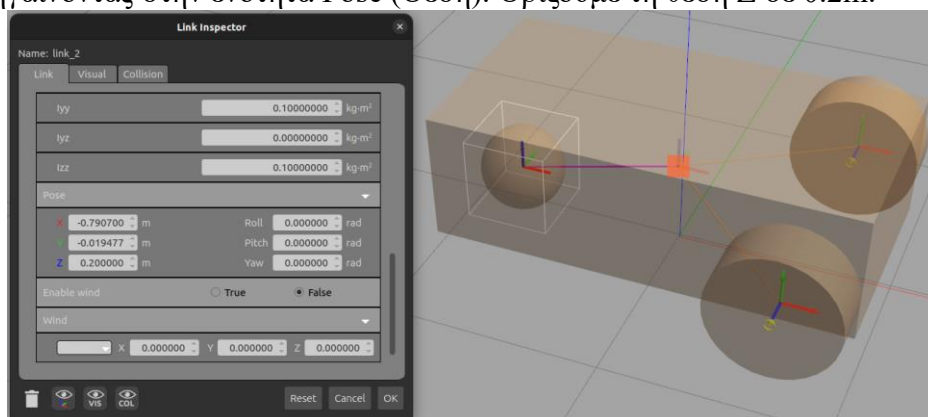
4. Σε αντίθεση με τις αρθρώσεις των μπροστινών τροχών, ένας περιστρεφόμενος τροχός (caster wheel) κινείται προς όλες τις κατευθύνσεις και δεν έχει συγκεκριμένο άξονα περιστροφής. Στο Gazebo, αυτό προσομοιώνεται χρησιμοποιώντας μια σφαιρική άρθρωση (ball joint). Στην ενότητα Joint Types στον διάλογο δημιουργίας αρθρώσεων, επιλέγουμε την επιλογή Ball joint. Θα παρατηρήσουμε ότι η οπτική αναπαράσταση της άρθρωσης στη σκηνή αλλάζει χρώμα, υποδεικνύοντας ότι έχει οριστεί διαφορετικός τύπος άρθρωσης.



5. Στη συνέχεια, ευθυγραμμίζουμε τον περιστρεφόμενο τροχό (caster wheel) ώστε να είναι κεντραρισμένος με το σασί και να τοποθετηθεί στο πίσω μέρος. Στην ενότητα Align Links στο διάλογο δημιουργίας αρθρώσεων, επιλέγουμε την επιλογή Y Align Center για να κεντράρουμε τον τροχό και το σασί κατά μήκος του άξονα Y. Στη συνέχεια, επιλέγουμε την επιλογή X Align Min για να μετακινήσουμε τον περιστρεφόμενο τροχό στο πίσω μέρος του οχήματος. πατάμε το κουμπί Create για να ολοκληρώσουμε τη διαδικασία δημιουργίας της άρθρωσης.



6. Τέλος, τοποθετούμε το περιστρεφόμενο τροχό (caster wheel) ακριβώς πάνω από το έδαφος. Το κάνουμε αυτό ανοίγοντας τον Επιθεωρητή Συνδέσμων (Link Inspector) και πηγαίνοντας στην ενότητα Pose (Θέση). Ορίζουμε τη θέση Z σε 0.2m.



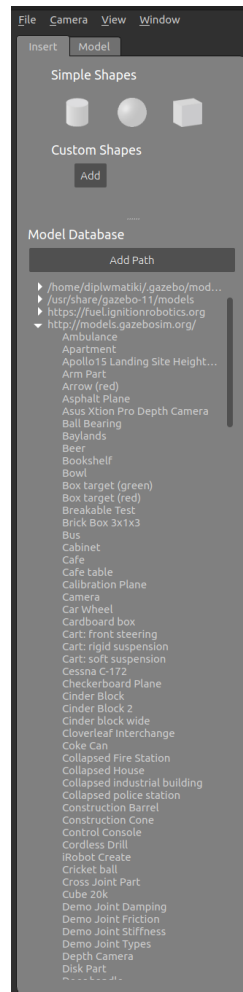
2.6.4 Προσθήκη Αισθητήρα

Ο αισθητήρας που θα προσθέσουμε στο όχημα είναι ένας αισθητήρας κάμερας βάθους, ο οποίος θα μας βοηθήσει να ανιχνεύουμε αντικείμενα μπροστά από το όχημα.

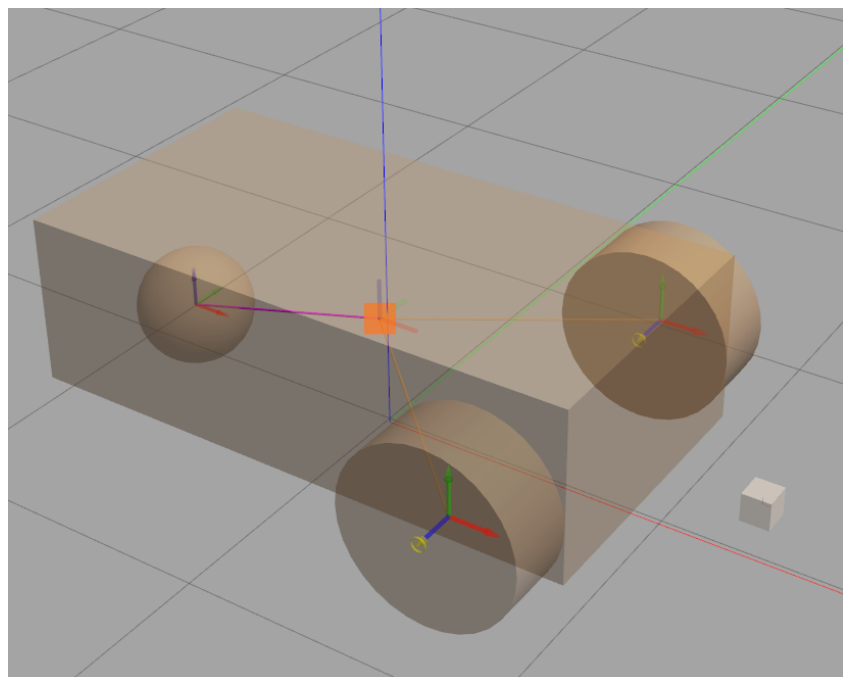
1. Πηγαίνουμε στην Παλέτα (Αριστερό Πάνελ) και επιλέγουμε την καρτέλα Insert για να δούμε μια λίστα με τα διαθέσιμα μοντέλα στη Βάση Δεδομένων Μοντέλων (Model Database).



2. Τα μοντέλα είναι οργανωμένα με βάση τη διαδρομή στην οποία βρίσκονται. Η πρώτη λίστα περιέχει μοντέλα που είναι διαθέσιμα στον τοπικό υπολογιστή, όπως φαίνεται από τη διαδρομή στον τίτλο. Εντοπίζουμε τη λίστα με τη διαδρομή <http://gazebosim.org/models/> και την επεκτείνουμε για να δούμε τα διαθέσιμα μοντέλα από την online βάση δεδομένων.

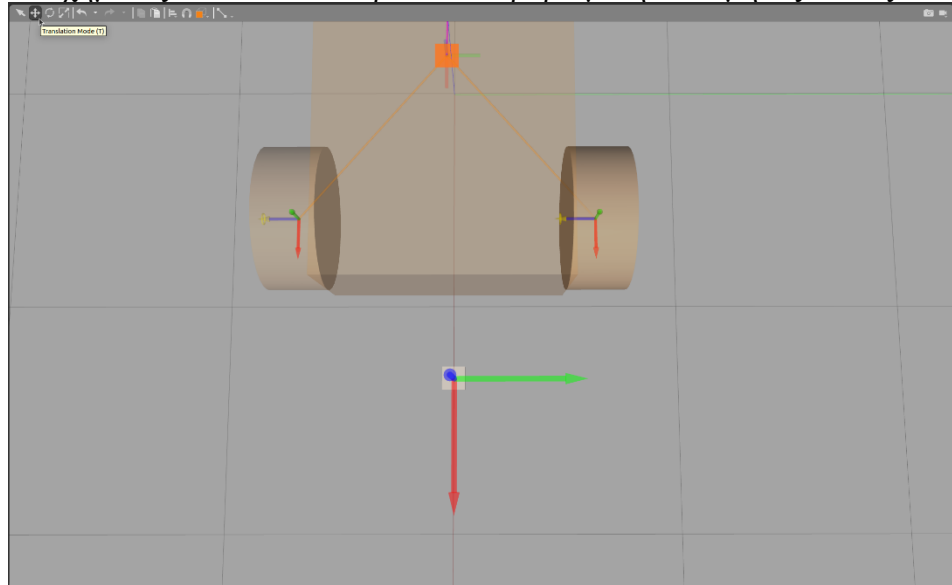


3. Τα μοντέλα είναι ταξινομημένα αλφαβητικά. Βρίσκουμε την Depth Camera στη λίστα και κάνουμε κλικ σε αυτήν για να ξεκινήσει η λήψη του μοντέλου. Αυτό μπορεί να πάρει μερικά δευτερόλεπτα, ανάλογα με τη συνδεσιμότητα του δικτύου.
4. Μόλις ολοκληρωθεί η λήψη, θα δούμε το μοντέλο της κάμερας βάθους να εμφανίζεται στη σκηνή. Φαίνεται σαν ένας μικρός κύβος. Μετακινούμε το ποντίκι στη σκηνή και κάνουμε κλικ σε έναν κενό χώρο μπροστά από το όχημα για να εισάγουμε την κάμερα βάθους.

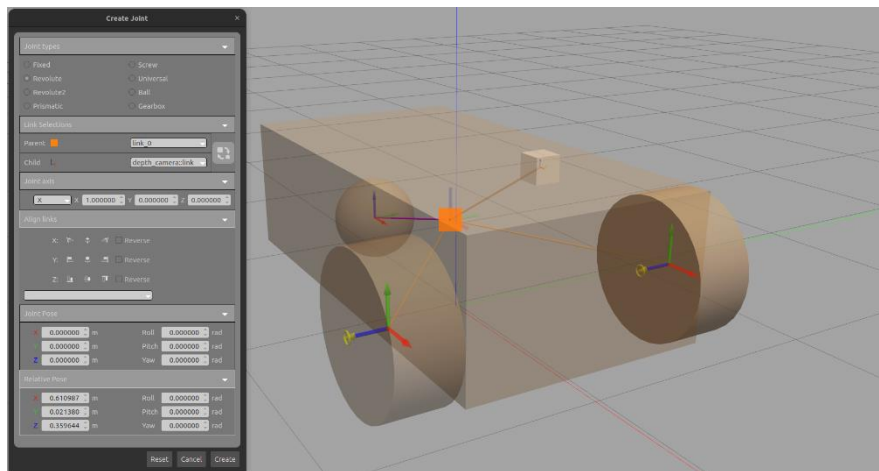


Προσομοίωση Ρομπότ μέσω Υπολογιστή

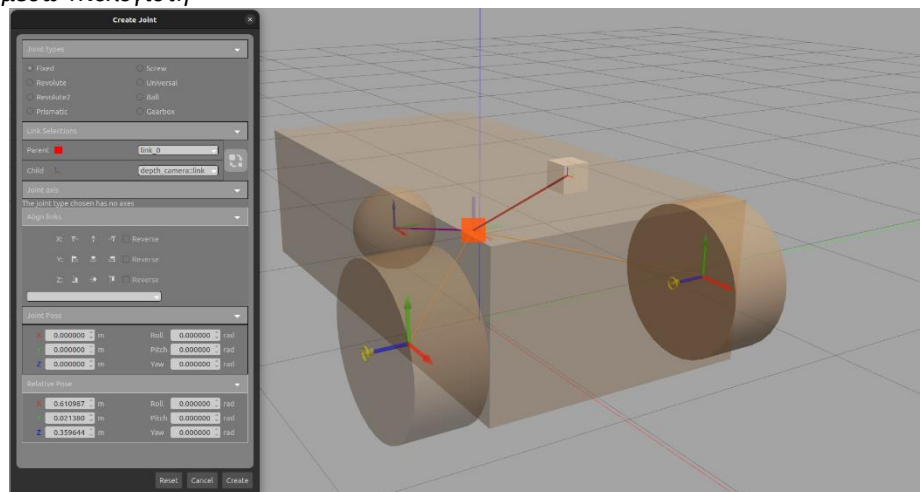
- Επιλέγουμε το εργαλείο Μετάφρασης (Translate) από την επάνω γραμμή εργαλείων και μετακινούμε την κάμερα βάθους έτσι ώστε να βρίσκεται πάνω στο σασί, στο μπροστινό μέρος του οχήματος, και να είναι περίπου κεντραρισμένη κατά μήκος του άξονα Y.



- Για να σταθεροποιήσουμε την κάμερα βάθους στο σασί, κάνουμε κλικ στο εικονίδιο Joint στην επάνω γραμμή εργαλείων για να ανοίξουμε τον διάλογο δημιουργίας άρθρωσης (Joint Creation). Μετακινούμε το ποντίκι στη σκηνή και επιλέγουμε το σασί ως γονέα (parent link) και την κάμερα βάθους ως παιδί (child link).



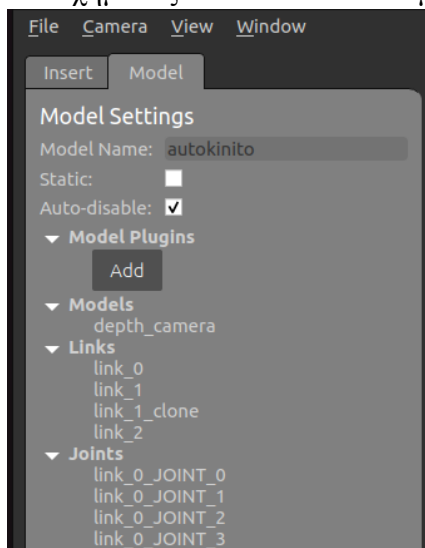
- Στο παράθυρο διαλόγου δημιουργίας αρθρώσεων, στην ενότητα Τύποι Αρθρώσεων (Joint Types), επιλέγουμε την επιλογή Fixed joint και κάνουμε κλικ στο Create για να ολοκληρώσουμε τη δημιουργία της άρθρωσης.



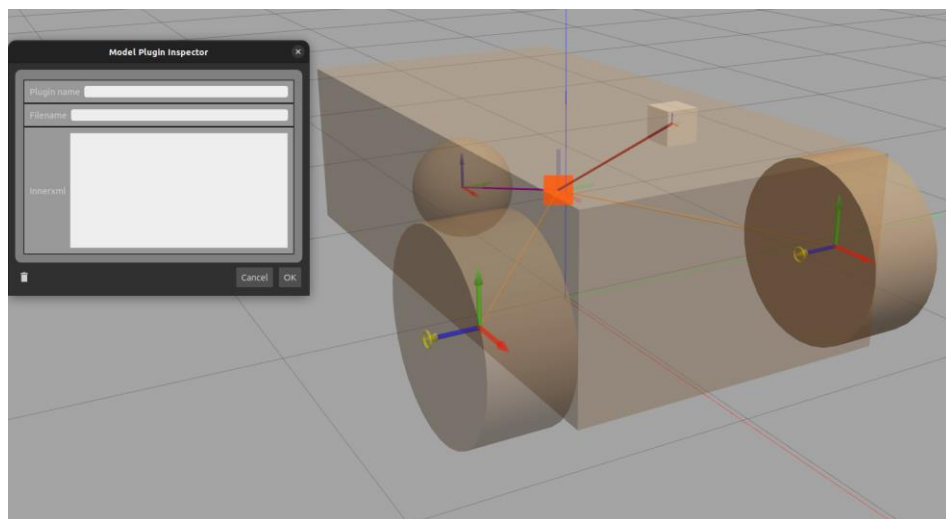
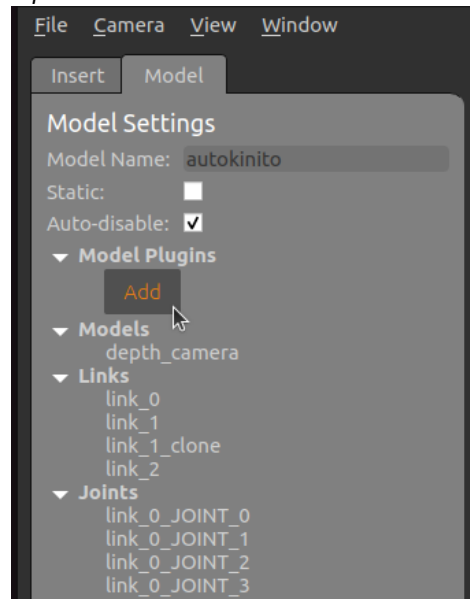
2.6.5 Προσθήκη Προσθέτου

Το όχημα που έχουμε κατασκευάσει μέχρι στιγμής είναι πλήρες με όλα τα φυσικά εξαρτήματα και τους αισθητήρες. Ωστόσο, δεν θα κάνει πολλά παρά να παραμένει ακίνητο και να παράγει δεδομένα βάθους στην προσομοίωση. Τα πρόσθετα (plugins) είναι ένας εξαιρετικός τρόπος για να ενισχύσουμε το μοντέλο με κάποια αυτονομία, επιτρέποντας του να εκτελεί υπολογισμούς, όπως επεξεργασία δεδομένων από αισθητήρες, σχεδιασμό διαδρομής και έλεγχο.

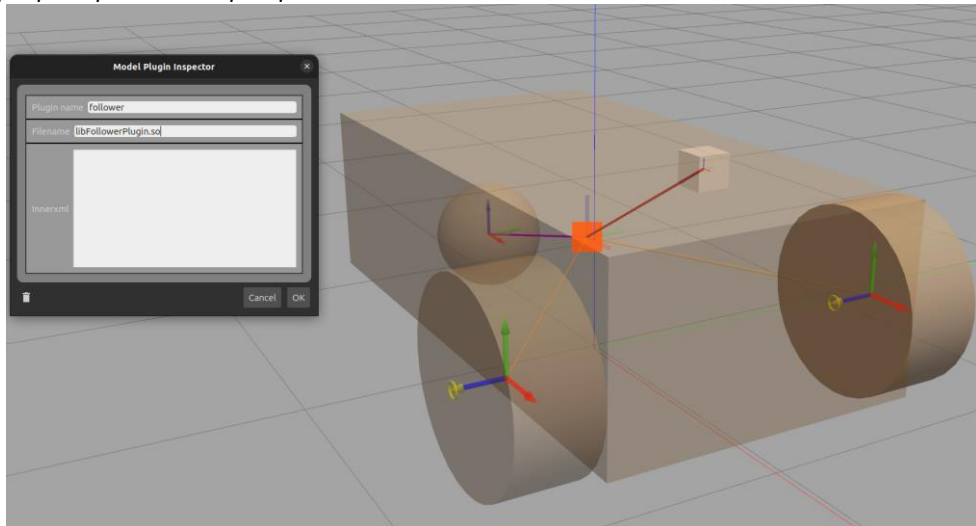
1. Πηγαίνουμε στο αριστερό πάνελ και επιλέγουμε την καρτέλα Model για να δούμε τα μέρη που αποτελούν το μοντέλο του οχήματος που κατασκευάσαμε.



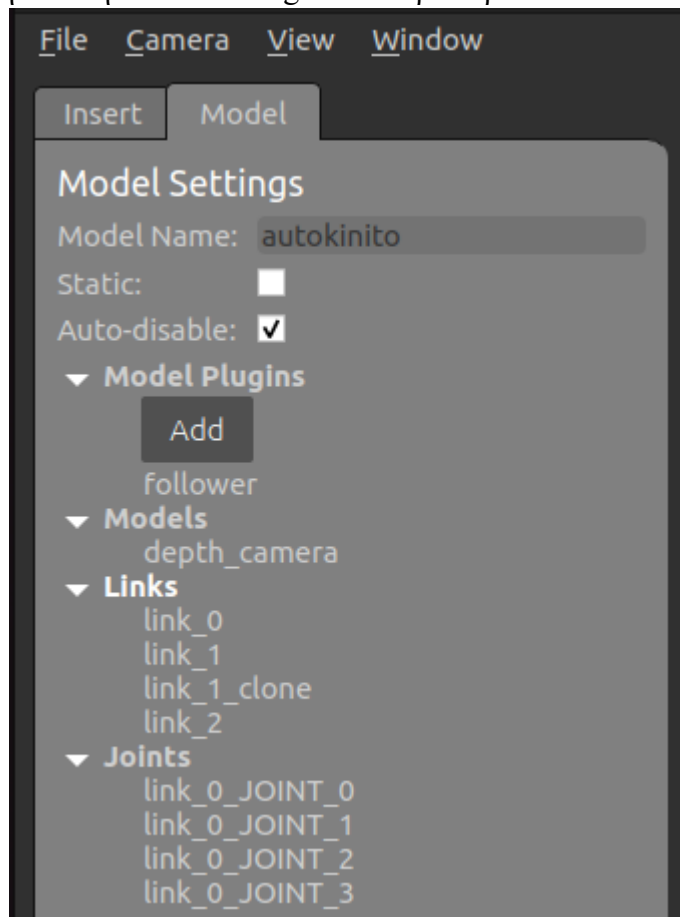
2. Κάτω από την ενότητα Model Plugins, θα πρέπει να βλέπουμε ένα κουμπί Add. Κάνουμε κλικ σε αυτό για να ανοίξει ο Model Plugin Inspector, που μας επιτρέπει να προσθέσουμε ένα νέο πρόσθετο στο μοντέλο.



3. Αρχικά, δίνουμε στο πρόσθετο ένα όνομα. Εισάγουμε το όνομα follower στο πεδίο Plugin Name. Το όνομα του πρόσθετου πρέπει να είναι μοναδικό μέσα στο συγκεκριμένο μοντέλο.
4. Το πρόσθετο που θα χρησιμοποιήσουμε ονομάζεται libFollowerPlugin.so, οπότε το εισάγουμε στο πεδίο Filename. Το όνομα του αρχείου αντιστοιχεί στο πραγματικό όνομα της βιβλιοθήκης του πρόσθετου που είναι αποθηκευμένο στον τοπικό μας υπολογιστή. Υπάρχει με τη μορφή δυναμικά συνδεδεμένης βιβλιοθήκης κοινόχρηστου αντικειμένου (shared object library), γι' αυτό και η επέκταση .so (στο Linux). Αν χρησιμοποιούμε το Gazebo σε άλλα λειτουργικά συστήματα, η επέκταση θα αντικατασταθεί αυτόματα με την κατάλληλη.

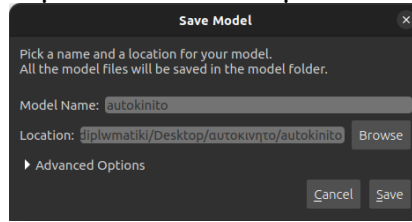


5. Το πρόσθετο follower δεν απαιτεί επιπλέον παραμέτρους, οπότε μπορούμε να αφήσουμε κενό το πεδίο Innerxml. Σημείωση: Αυτό είναι ένα απλό πρόσθετο για σκοπούς επίδειξης. Συνήθως, τα πρόσθετα έχουν διάφορες παραμέτρους. Για παράδειγμα, ένα πρόσθετο διαφορικής οδήγησης (differential drive plugin) απαιτεί τον ορισμό των ονομάτων των αρθρώσεων που ελέγχουν τους αριστερούς και δεξιούς τροχούς, ώστε να μπορεί να κινεί το όχημα προς τη σωστή κατεύθυνση. Στην περίπτωση του πρόσθετου follower, κάνει πολλές υποθέσεις για τον τύπο του μοντέλου στο οποίο είναι συνδεδεμένο και προσπαθεί να βρει αυτόματα τις αρθρώσεις και τον αισθητήρα.
6. Κάνουμε κλικ στο OK για να προσθέσουμε το πρόσθετο. Το πρόσθετο θα εμφανιστεί τώρα κάτω από την ενότητα Model Plugins στο αριστερό πάνελ.

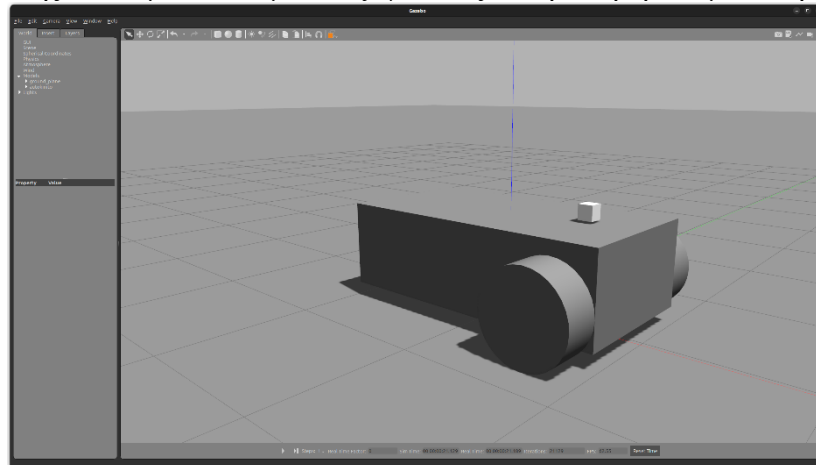


2.6.6 Αποθήκευση του Μοντέλου

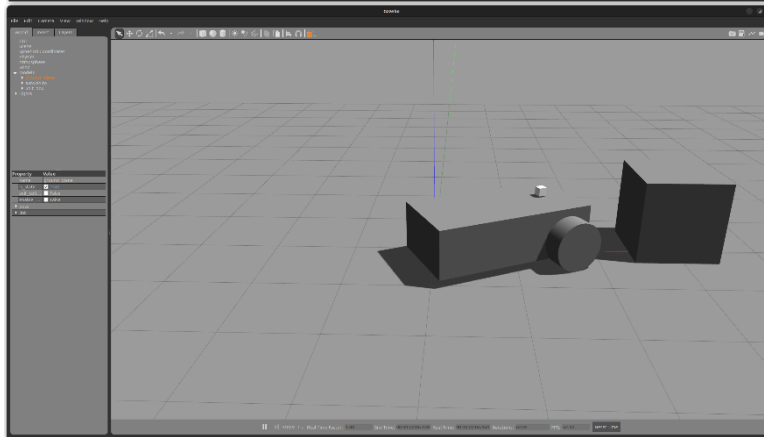
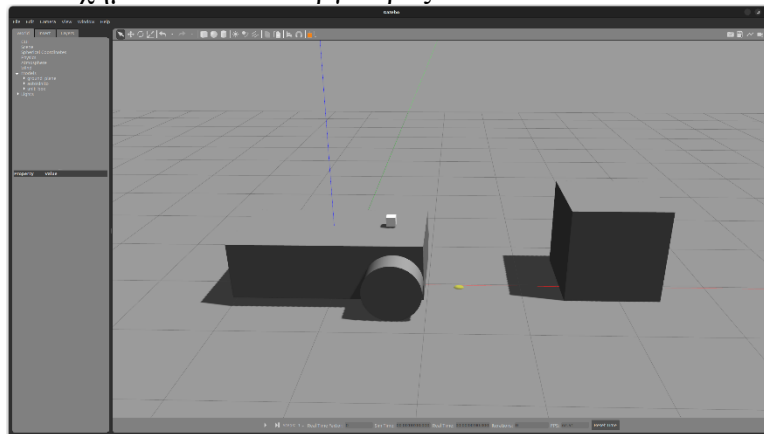
1. Για να αποθηκεύσουμε το μοντέλο, πηγαίνουμε στο μενού File και επιλέγουμε Save As. Εισάγουμε ένα όνομα για το μοντέλο και κάνουμε κλικ στο Save.



2. Για να βγούμε από τον Επεξεργαστή Μοντέλου, πηγαίνουμε ξανά στο μενού File και επιλέγουμε Exit Model Editor. Το Gazebo θα επιστρέψει στη λειτουργία κανονικής προσομοίωσης. Πατάμε το κουμπί Play για να ξεκινήσει η προσομοίωση.



3. Για να ελέγξουμε αν το πρόσθετο λειτουργεί, εισάγουμε ένα κουτί μπροστά από το όχημα και παρατηρούμε το όχημα να κινείται αργά προς το κουτί.



4. Εάν θέλουμε να επεξεργαστούμε το μοντέλο αργότερα κάνουμε δεξί κλικ πάνω στο μοντέλο

Προσομοίωση Ρομπότ μέσω Υπολογιστή
και επιλέγουμε Edit Model από το μενού που εμφανίζεται.

2.7 Εξώθηση αρχείων SVG

Θα περιγράψουμε τη διαδικασία εξώθησης αρχείων SVG, τα οποία είναι δισδιάστατες εικόνες, για τη δημιουργία τρισδιάστατων πλεγμάτων (meshes) για τα μοντέλα μας στο Gazebo. Μερικές φορές είναι πιο εύκολο να σχεδιάσουμε μέρος ενός μοντέλου σε ένα πρόγραμμα όπως το Inkscape ή το Illustrator.

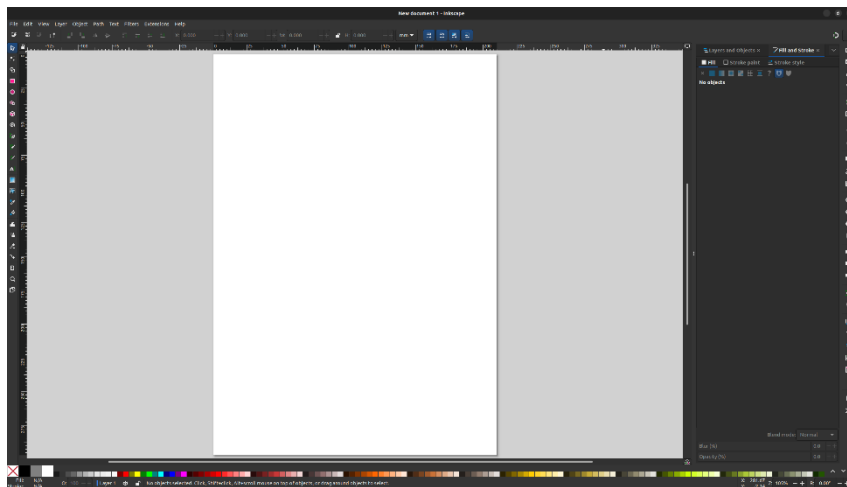
Θα δείξουμε πώς να δημιουργήσουμε έναν προσαρμοσμένο τροχό ως αρχείο .svg στο Inkscape και να τον εισάγουμε στο Gazebo, ώστε να μπορεί να συνδεθεί με ένα ρομπότ.

2.7.1 Χρήση του επεξεργαστή SVG του Inkscape



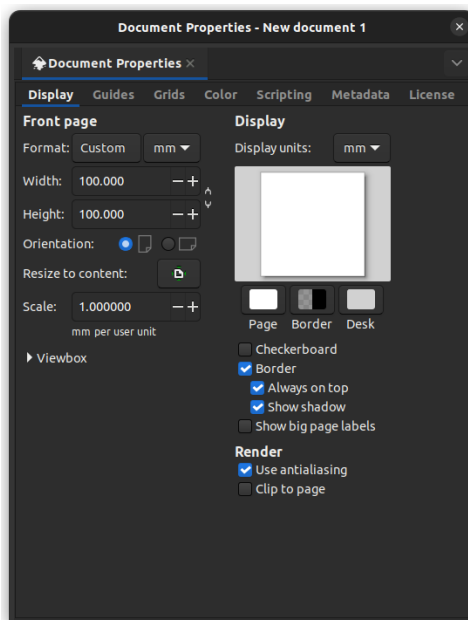
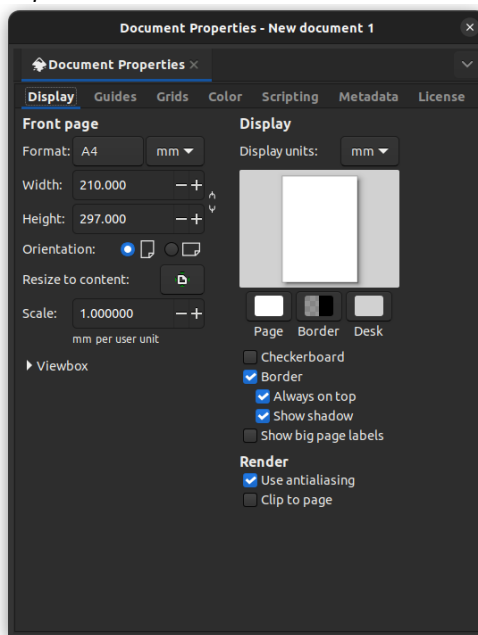
2.7.2 Προετοιμασία του εγγράφου

Ανοίγουμε το Inkscape. Αυτό θα δημιουργήσει ένα κενό έγγραφο.

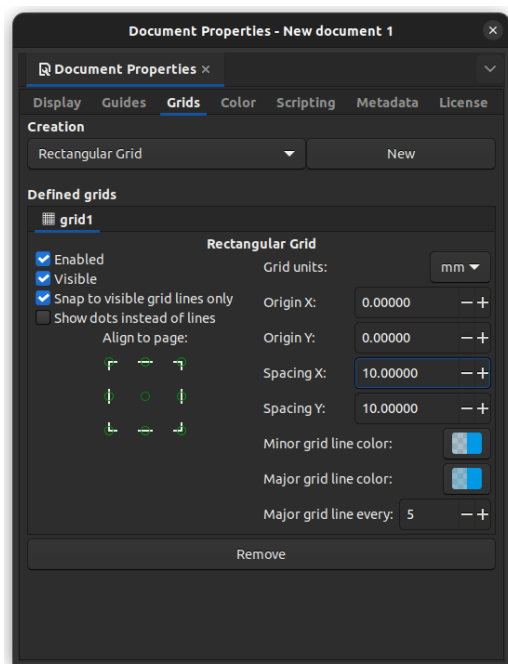
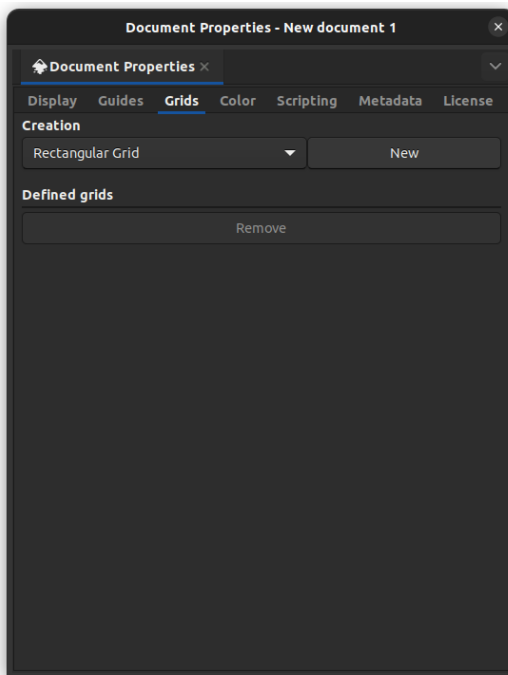


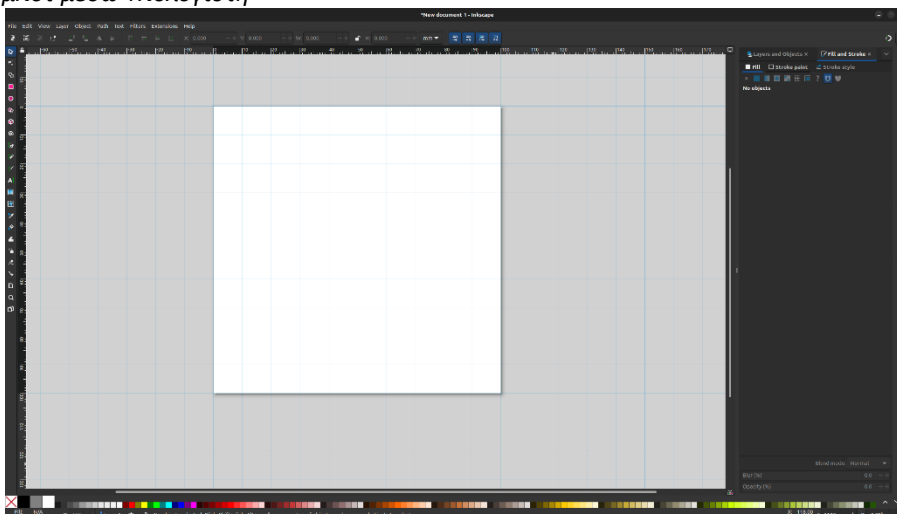
Αλλάζουμε το μέγεθος του εγγράφου για να προσαρμόσουμε καλύτερα τον τροχό μας. Πηγαίνουμε στο μενού File --> Document Properties, επιλέγουμε την καρτέλα Page και αλλάζουμε το μέγεθος του εγγράφου σε 100.0 x 100.0 mm.

Προσομοίωση Ρομπότ μέσω Υπολογιστή



Στο ίδιο παράθυρο διαλόγου, επιλέγουμε την καρτέλα Grids, πατάμε το κουμπί New για να δημιουργήσουμε ένα προσαρμοσμένο πλέγμα. Ενεργοποιούμε τις επιλογές Enabled, Visible, και Snap to visible grid lines only. Αλλάζουμε το Spacing X και το Spacing Y σε 10.

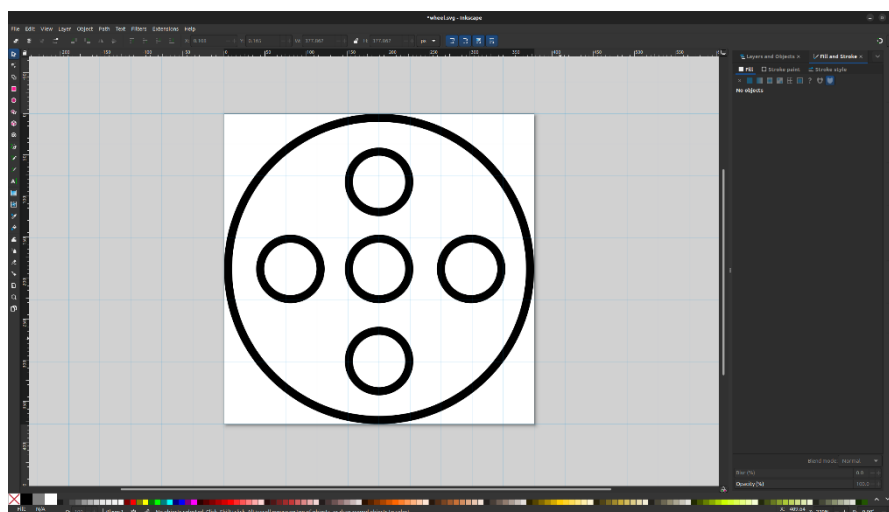




2.7.3 Σχέδιο

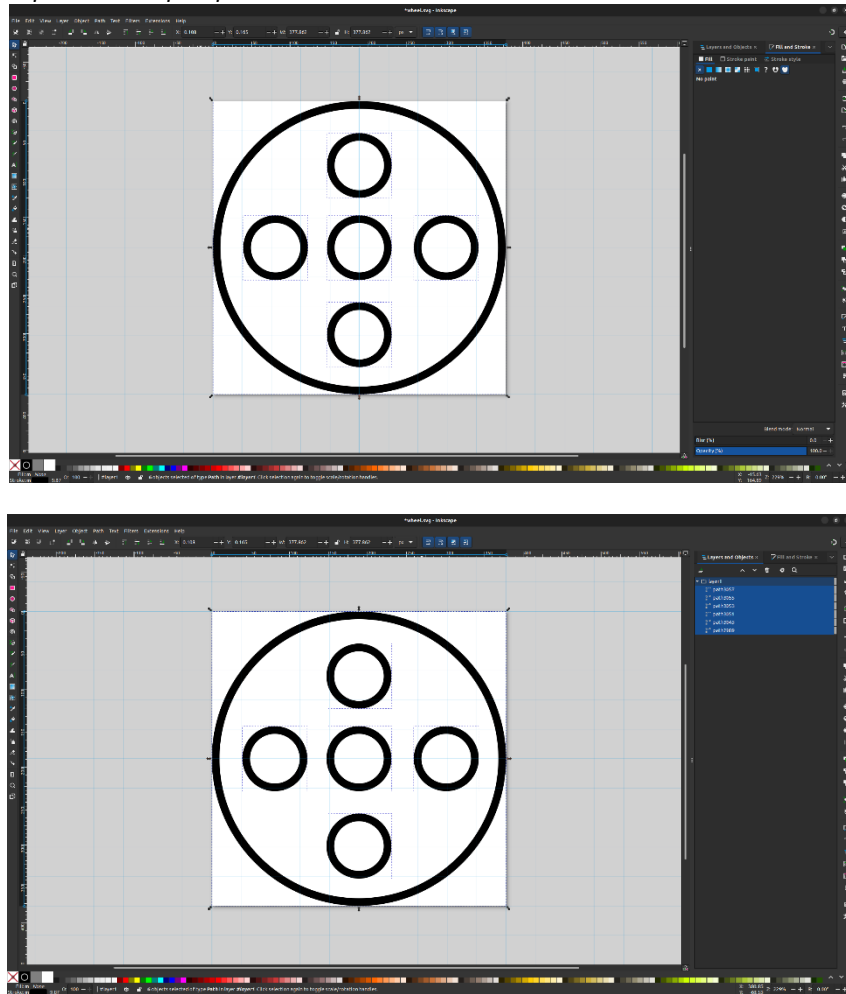
Χρησιμοποιούμε τα διάφορα εργαλεία (πέννα, κείμενο, αστέρια, σχήματα κλπ.) για να δημιουργήσουμε τη γεωμετρία μας. Σε αυτό το παράδειγμα, ο τροχός δημιουργείται από κύκλους (κρατώντας πατημένο το πλήκτρο Shift, ξεκινάμε τον κύκλο από το κέντρο, και χρησιμοποιώντας το πλήκτρο CTRL διατηρούμε τη στρογγυλότητα του σχήματος). Είναι δυνατό να συνδυάσουμε σχήματα, διασφαλίζοντας ότι οι διαδρομές (paths) είναι κλειστές και ότι το κομμάτι έχει κατάλληλο πάχος.

Σημείωση: Ένα ραβδόμορφο σχήμα ή δύο κύκλοι που αγγίζουν ο ένας τον άλλον δεν θα δημιουργήσουν έγκυρα μοντέλα για το Gazebo. Οι διαδρομές SVG πρέπει να δημιουργούν ένα κλειστό περίγραμμα με τρύπες, όπου οι τρύπες δεν αγγίζουν το περίγραμμα ή άλλες τρύπες.



Το Gazebo εισάγει μόνο διαδρομές (paths), αλλά είναι εύκολο στο Inkscape να μετατρέψουμε οποιοδήποτε σχήμα σε διαδρομή. Επιλέγουμε Select All από το μενού Edit. Στη συνέχεια, επιλέγουμε το Path --> Object to Path από το μενού. Αυτό θα μετατρέψει κάθε αντικείμενο σε ξεχωριστές διαδρομές και υπο-διαδρομές. Αυτή η μετατροπή είναι μη αναστρέψιμη, οπότε αν μετατρέψουμε κείμενο σε διαδρομές, δεν θα μπορούσαμε να τροποποιήσουμε το κείμενο αργότερα.

Προσομοίωση Ρομπότ μέσω Υπολογιστή



Το Gazebo δεν υποστηρίζει ομαδοποίηση αντικειμένων. Χρησιμοποιούμε την επιλογή Ungroup από το μενού Object για να χωρίσουμε τις ομάδες διαδρομών.

2.7.4 Αποθήκευση του σχεδίου

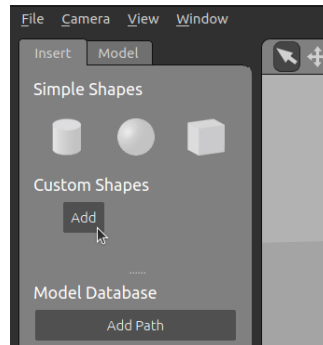
Αποθηκεύουμε το σχέδιό μας σε ένα αρχείο SVG που θα χρησιμοποιήσουμε αργότερα στο Gazebo. Χρησιμοποιούμε την επιλογή Save από το μενού File.

2.8 Δημιουργία Μοντέλου στο Gazebo

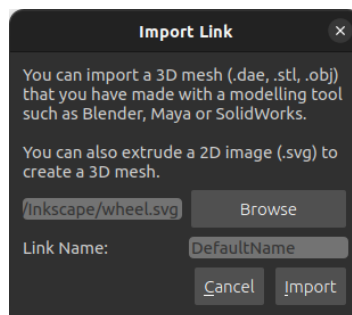
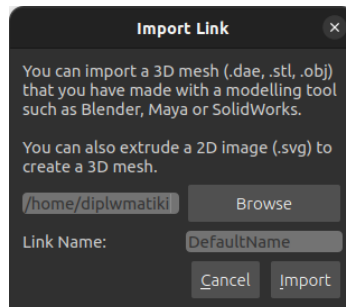
Το SDF format δεν υποστηρίζει απευθείας αρχεία SVG. Υποστηρίζει 2D πολυγραμμές. Ο Επεξεργαστής Μοντέλων του Gazebo έχει έναν μηχανισμό εισαγωγής που εξάγει τις πολυγραμμές από τα αρχεία SVG και τις αποθηκεύει ως αρχείο SDF μοντέλου.

Εκκινούμε το Gazebo και επιλέγουμε τον Model Editor από το μενού Edit για να εισέλθουμε στη λειτουργία Επεξεργαστή Μοντέλων.

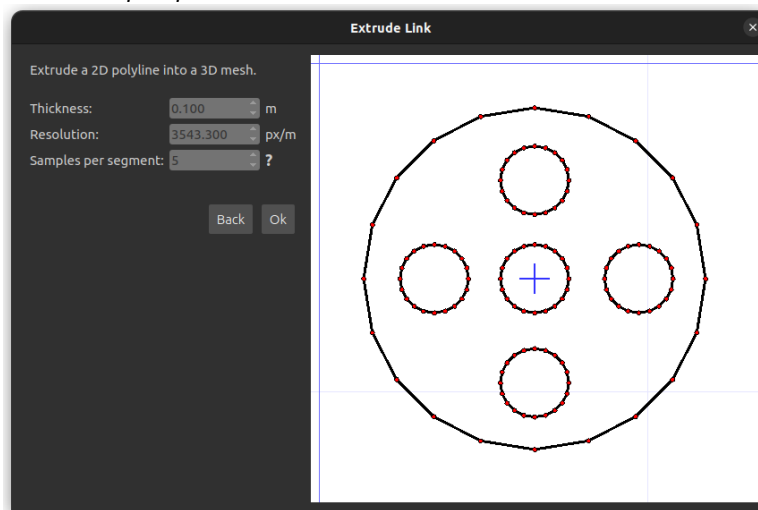
Πατάμε το κουμπί Add στην ενότητα Custom Shapes της καρτέλας Insert.



Ανοίγει το παράθυρο διαλόγου Import Link, από όπου μπορούμε να επιλέξουμε το αρχείο SVG πατώντας το κουμπί Browse.



Αφού επιλέξουμε το αρχείο, πατάμε το κουμπί Import για να ανοίξει το παράθυρο διαλόγου Extrude Link.



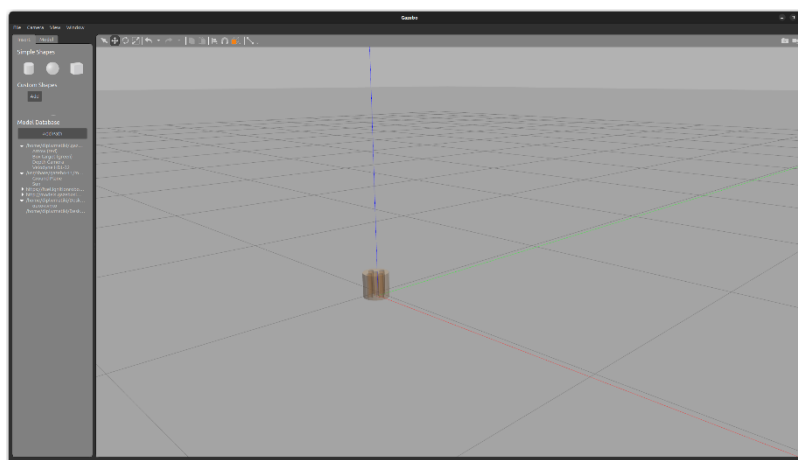
Το παράθυρο διαλόγου μας επιτρέπει να ορίσουμε τις παραμέτρους της εξώθησης:

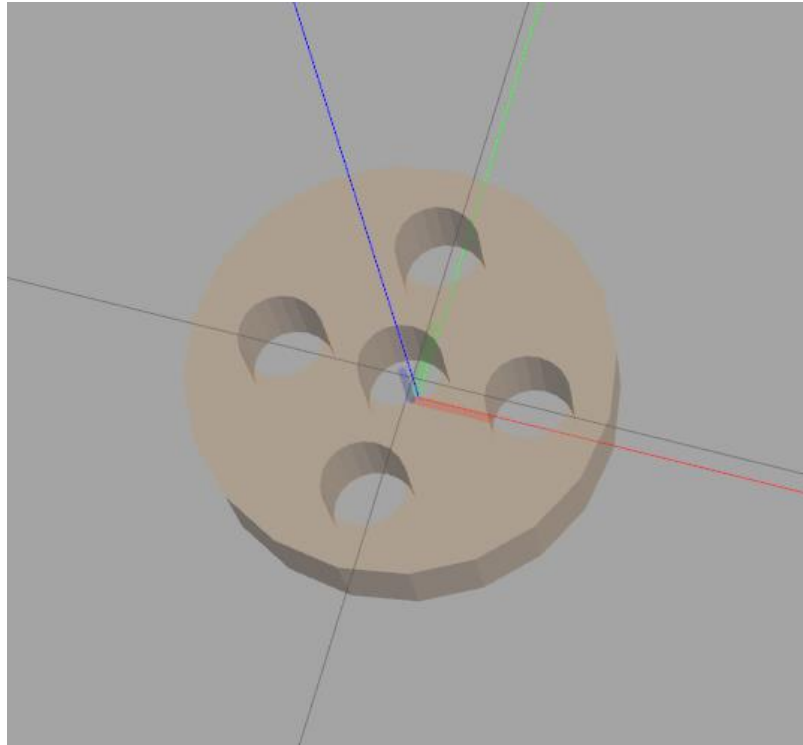
- **Thickness:** Καθορίζει το πάχος του συνδέσμου (link). Αυτό αντιστοιχεί στο ύψος της εξώθησης στον άξονα Z. Για τη διαδρομή SVG που φαίνεται στα δεξιά, ο άξονας εξώθησης εκτείνεται προς τα έξω από την οθόνη.
- **Resolution:** Πόσα pixels στο αρχείο SVG αντιστοιχούν σε ένα μέτρο. Η προεπιλεγμένη τιμή (3543,3 px/m) αντιστοιχεί σε 90 dpi (κουκκίδες ανά ίντσα/dots per inch), που είναι η προεπιλεγμένη ανάλυση για πολλά προγράμματα, όπως το Inkscape.
- **Samples per segment:** Αυτό υποδεικνύει σε πόσα τμήματα θα διαιρεθεί κάθε καμπύλη διαδρομή στο αρχείο SVG. Όσο περισσότερα τμήματα, τόσο πιο περίπλοκος θα είναι ο σύνδεσμος. Δεν επηρεάζει τις ευθείες διαδρομές.

Στα δεξιά, μπορούμε να δούμε τη διαδρομή που εξάγεται από το αρχείο SVG. Τα κόκκινα σημεία είναι οι κορυφές της τριγωνοποίησης του εξωθημένου τρισδιάστατου μοντέλου.

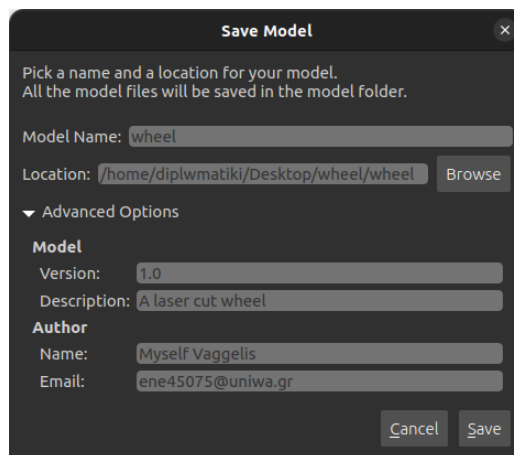
Ορίζουμε το πάχος του τροχού σε 0,025 m και πατάμε OK. Ο νέος σύνδεσμος εμφανίζεται στην τρισδιάστατη προβολή.

Ένας νέος σύνδεσμος δημιουργείται, ο οποίος περιλαμβάνει ένα προεπιλεγμένο σχήμα σύγκρουσης (collision shape) που είναι αντίγραφο του παραγόμενου τρισδιάστατου πλέγματος.





Στη συνέχεια, επιλέγουμε Exit Model Editor από το μενού File. Το Gazebo θα μας ζητήσει να αποθηκεύσουμε το νέο μοντέλο στον δίσκο. Πατάμε το κουμπί Save and Exit στον διάλογο εξόδου και θα εμφανιστεί ο διάλογος Save Model.

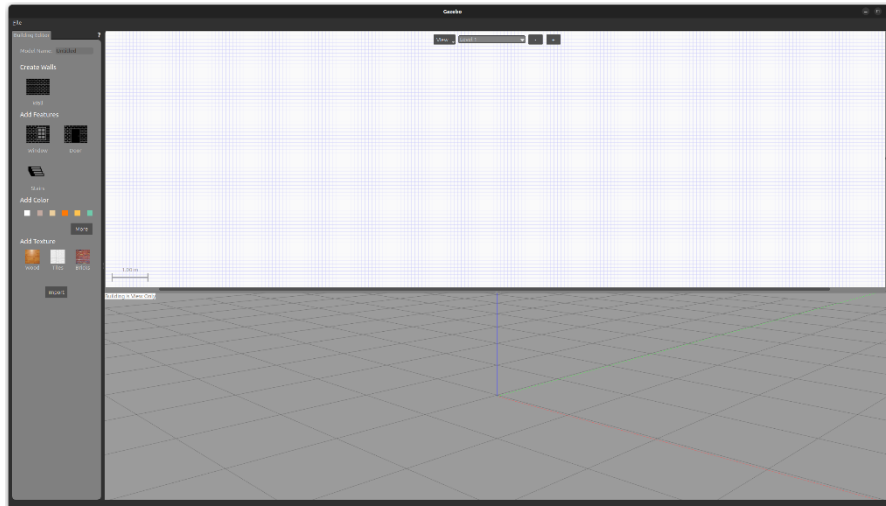


Ορίζουμε το όνομα του νέου μοντέλου σε "Wheel" και συμπληρώνουμε τις πληροφορίες στην ενότητα Advanced Options. Στη συνέχεια, πατάμε το κουμπί Save για να αποθηκεύσουμε το μοντέλο.

2.9 Επεξεργαστής Κτιρίων (Building Editor)

Θα περιγράψουμε τη διαδικασία κατασκευής ενός κτηρίου χρησιμοποιώντας το Building Editor.

Στο μενού Edit, πηγαίνουμε στην επιλογή Building Editor ή πατάμε Ctrl+B για να ανοίξουμε τον επεξεργαστή.



2.9.1 Γραφική διεπαφή χρήστη (Graphical User Interface)

Ο επεξεργαστής αποτελείται από τις εξής 3 περιοχές:

1. **Παλέτα (Palette):** Εδώ μπορούμε να επιλέξουμε χαρακτηριστικά και υλικά για το κτίριό μας.
2. **2D Προβολή (2D View):** Σε αυτή την περιοχή μπορούμε να εισάγουμε ένα σχέδιο κάτοψης για να το χρησιμοποιήσουμε ως οδηγό (προαιρετικό) και να προσθέσουμε τοίχους, παράθυρα, πόρτες και σκάλες.
3. **3D Προβολή (3D View):** Εδώ μπορούμε να δούμε μια προεπισκόπηση του κτιρίου μας. Μπορούμε επίσης να αναθέσουμε χρώματα και υφές σε διάφορα μέρη του κτιρίου.

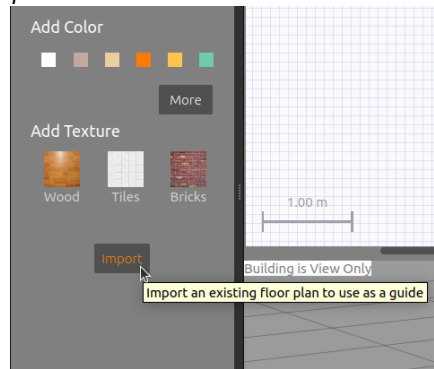


2.9.2 Εισαγωγή κάτοψης

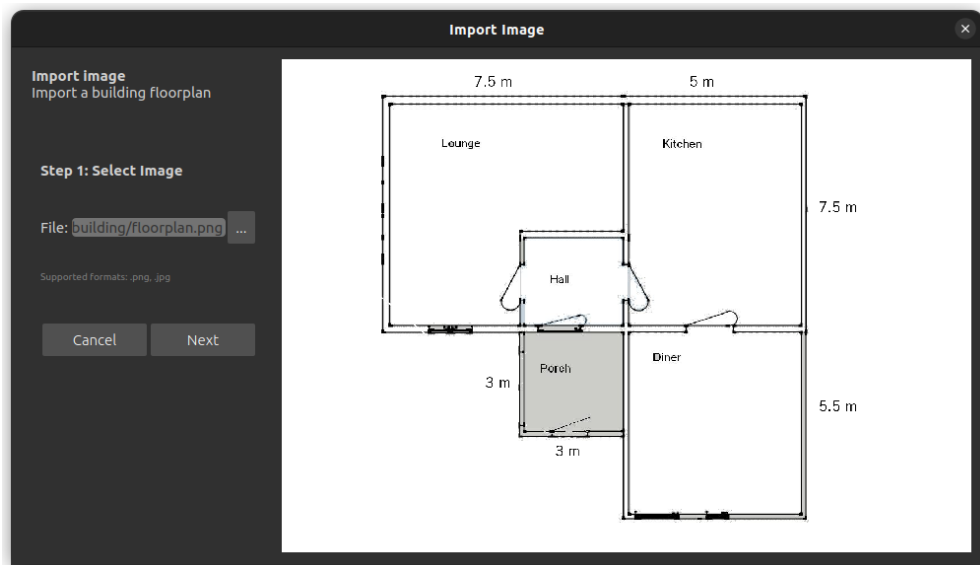
Μπορούμε να δημιουργήσουμε μια σκηνή από το μηδέν ή να χρησιμοποιήσουμε μια υπάρχουσα εικόνα ως πρότυπο για να σχεδιάσουμε από πάνω της. Αυτή η εικόνα μπορεί να είναι, για παράδειγμα, μια δισδιάστατη σάρωση λέιζερ ενός κτιρίου.

1. Κάνουμε κλικ στο κουμπί Import. Ο διάλογος Import Image θα εμφανιστεί.

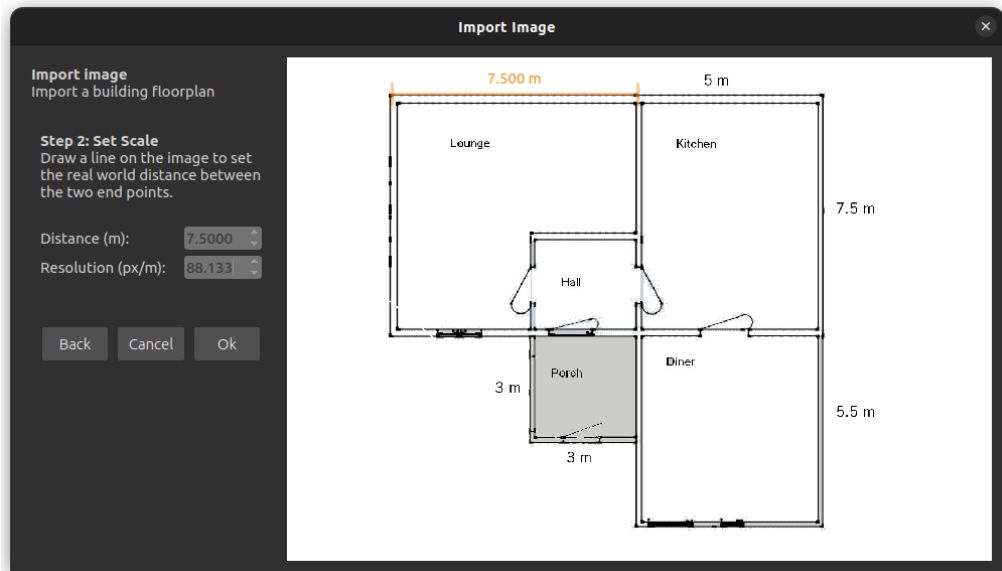
Προσομοίωση Ρομπότ μέσω Υπολογιστή



2. Βήμα 1: Επιλέγουμε την εικόνα που έχουμε αποθηκεύσει στον υπολογιστή μας και κάνουμε κλικ στο Next.

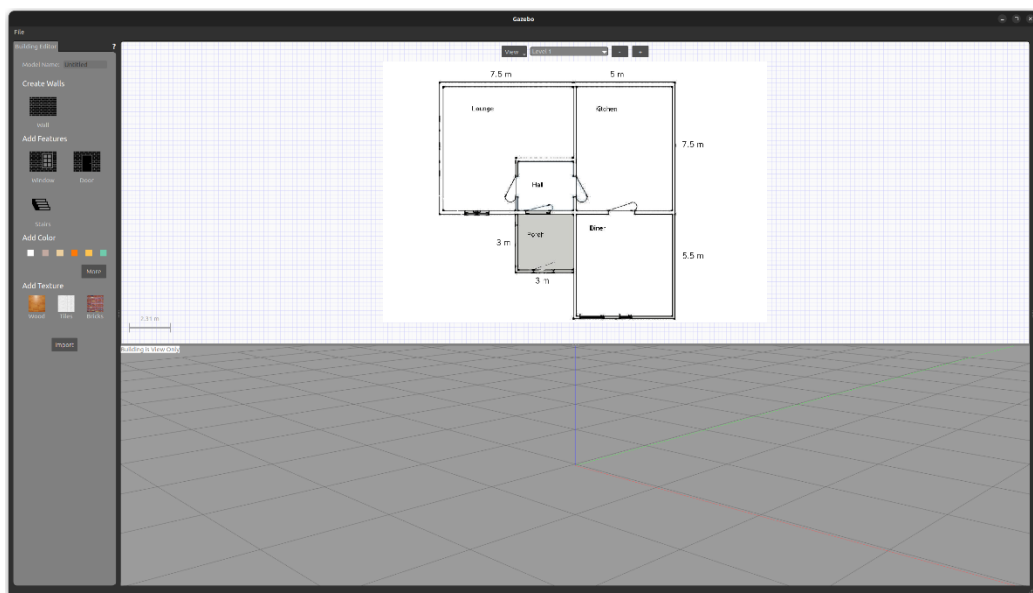


3. **Βήμα 2:** Για να διασφαλίσουμε ότι οι τοίχοι που θα σχεδιάσουμε από πάνω είναι στη σωστή κλίμακα, πρέπει να ορίσουμε την ανάλυση της εικόνας σε pixels ανά μέτρο (px/m). Αν γνωρίζουμε την ανάλυση, μπορούμε να την εισάγουμε απευθείας στο διάλογο και να κάνουμε κλικ στο Ok. Στην περίπτωση που δεν γνωρίζουμε την ανάλυση αλλά ξέρουμε την πραγματική απόσταση μεταξύ δύο σημείων στην εικόνα (π.χ., ο πάνω τοίχος είναι 7,5 μέτρα), ακολουθούμε τα παρακάτω βήματα:
 - a. Κάνουμε κλικ/απελευθερώνουμε σε ένα άκρο του τοίχου. Καθώς μετακινούμε το ποντίκι, θα εμφανιστεί μια πορτοκαλί γραμμή.
 - b. Κάνουμε κλικ/απελευθερώνουμε στο άλλο άκρο του τοίχου για να ολοκληρώσουμε τη γραμμή.
 - c. Εισάγουμε την απόσταση σε μέτρα στο διάλογο (7,5 m σε αυτή την περίπτωση). Η ανάλυση θα υπολογιστεί αυτόματα με βάση τη γραμμή που σχεδιάσαμε.
 - d. Κάνουμε κλικ στο Ok.



4. Η εικόνα θα εμφανιστεί στην 2D Προβολή στη σωστή κλίμακα.

Συμβουλή: Αφού προσθέσουμε περισσότερους ορόφους, μπορούμε να εισάγουμε μια κάτοψη για κάθε έναν επαναλαμβάνοντας την ίδια διαδικασία.

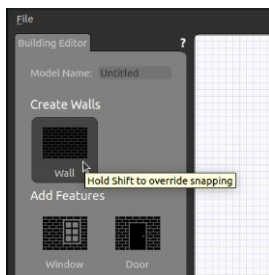


2.9.3 Προσθήκη χαρακτηριστικών

2.9.3.1 Προσθήκη τοίχων

Για να σχεδιάσουμε όλους τους τοίχους στην κάτοψη, κρατάμε στο νου μας ότι θα προσθέσουμε παράθυρα και πόρτες αργότερα, επομένως μπορούμε να σχεδιάσουμε τους τοίχους πάνω από αυτά τα στοιχεία χωρίς πρόβλημα. Ακόμα κι αν οι τοίχοι δεν είναι τέλειοι, μπορούμε να τους επεξεργαστούμε αργότερα.

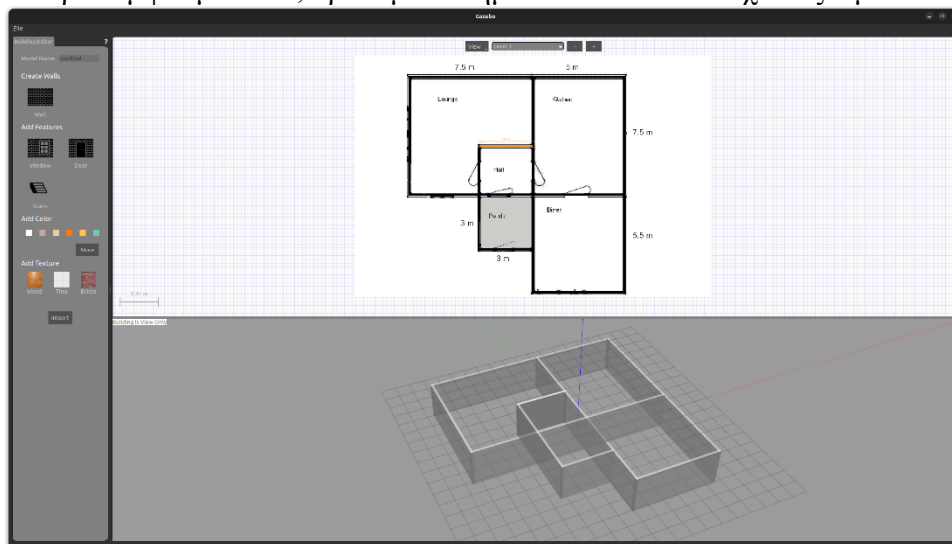
1. Στην Παλέτα, κάνουμε κλικ στο Wall.



2. Στην 2D Προβολή, κάνουμε κλικ/απελευθερώνουμε οπουδήποτε για να ξεκινήσουμε έναν τοίχο. Καθώς μετακινούμε το ποντίκι, εμφανίζεται το μήκος του τοίχου.
3. Κάνουμε ξανά κλικ για να τελειώσουμε τον τρέχοντα τοίχο και να ξεκινήσουμε έναν γειτονικό.
4. Κάνουμε διπλό κλικ για να ολοκληρώσουμε τον τοίχο χωρίς να ξεκινήσουμε καινούργιο.

Συμβουλές:

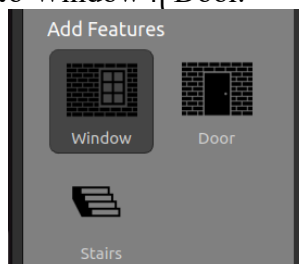
- Μπορούμε να κάνουμε δεξί κλικ ή να πατήσουμε Esc για να ακυρώσουμε το τρέχον τμήμα τοίχου.
- Οι τοίχοι, από προεπιλογή, κουμπώνουν στις γωνίες σε αυξήσεις των 15° και 0,25 μέτρων. Για να παρακάμψουμε αυτό, κρατάμε πατημένο το Shift ενώ σχεδιάζουμε.



2.9.3.2 Προσθήκη παραθύρων και πορτών

Τα παράθυρα και οι πόρτες είναι απλά ανοίγματα στους τοίχους. Για να τα εισάγουμε, ακολουθούμε τα παρακάτω βήματα:

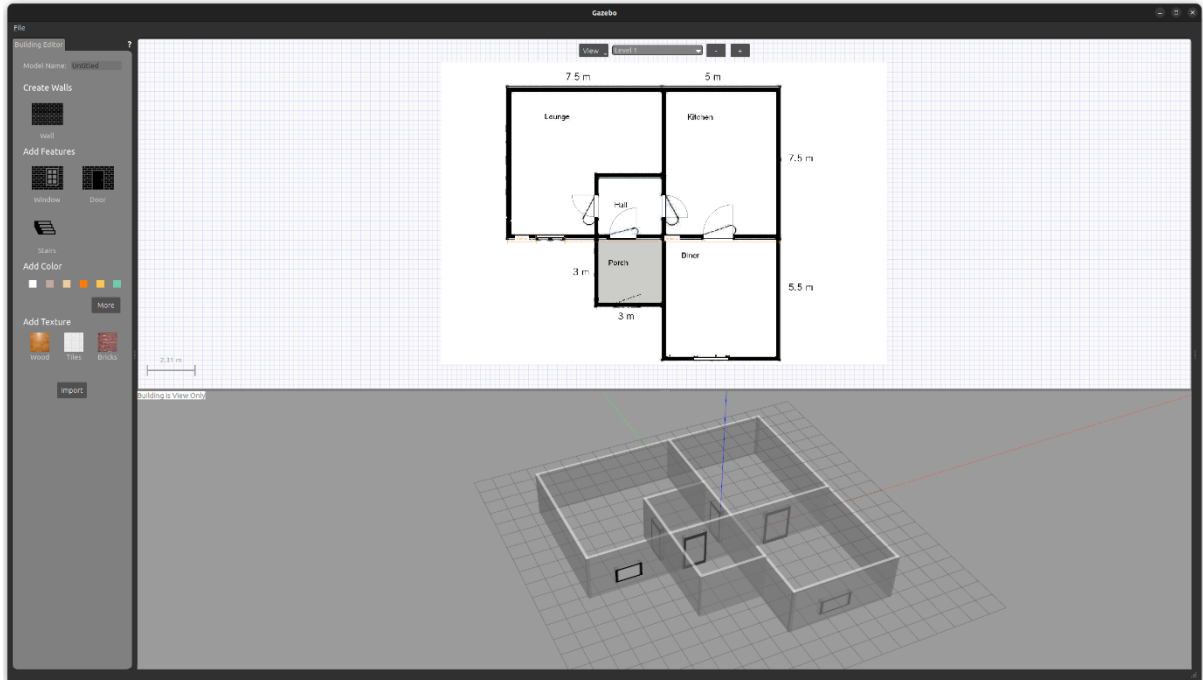
1. Στην Παλέτα, κάνουμε κλικ στο Window ή Door.



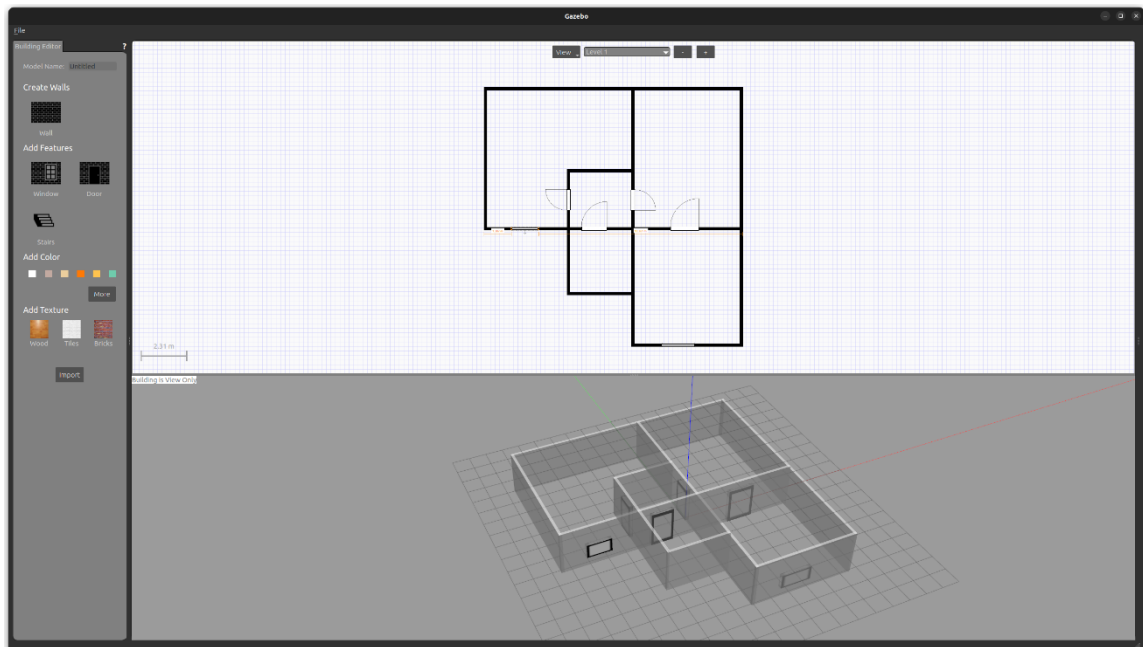
2. Καθώς μετακινούμε το ποντίκι στην 2D Προβολή, το στοιχείο που πρόκειται να εισαχθεί μετακινείται μαζί με αυτό, όπως και το αντίστοιχο στη 3D Προβολή.
3. Κάνουμε κλικ στη θέση που θέλουμε για να τοποθετήσουμε το στοιχείο.

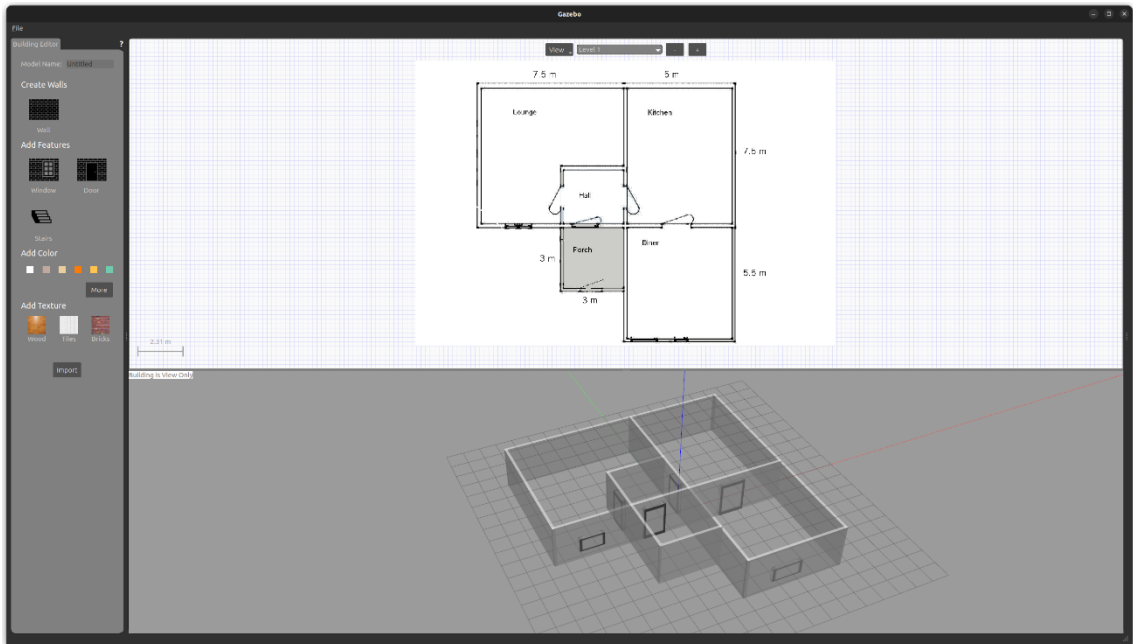
Προσομοίωση Ρομπότ μέσω Υπολογιστή

Συμβουλή: Τα παράθυρα και οι πόρτες κουμπώνουν αυτόματα στους τοίχους καθώς τα τοποθετούμε.



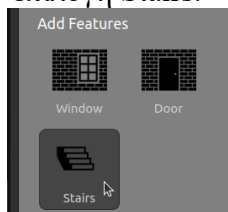
Συμβουλή: Μπορεί να είναι δύσκολο να δούμε πού βρίσκονται τα χαρακτηριστικά στην κάτοψη αφού έχουν σχεδιαστεί οι τοίχοι από πάνω της. Για να το κάνουμε πιο εύκολο, στην κορυφή της 2D Προβολής, μπορούμε να επιλέξουμε να εμφανίσουμε ή να αποκρύψουμε την κάτοψη ή τα χαρακτηριστικά για το τρέχον επίπεδο. Μπορούμε επίσης να χρησιμοποιήσουμε συντομεύσεις πληκτρολογίου για να εναλλάξουμε την ορατότητα, F για την κάτοψη και G για τα χαρακτηριστικά.



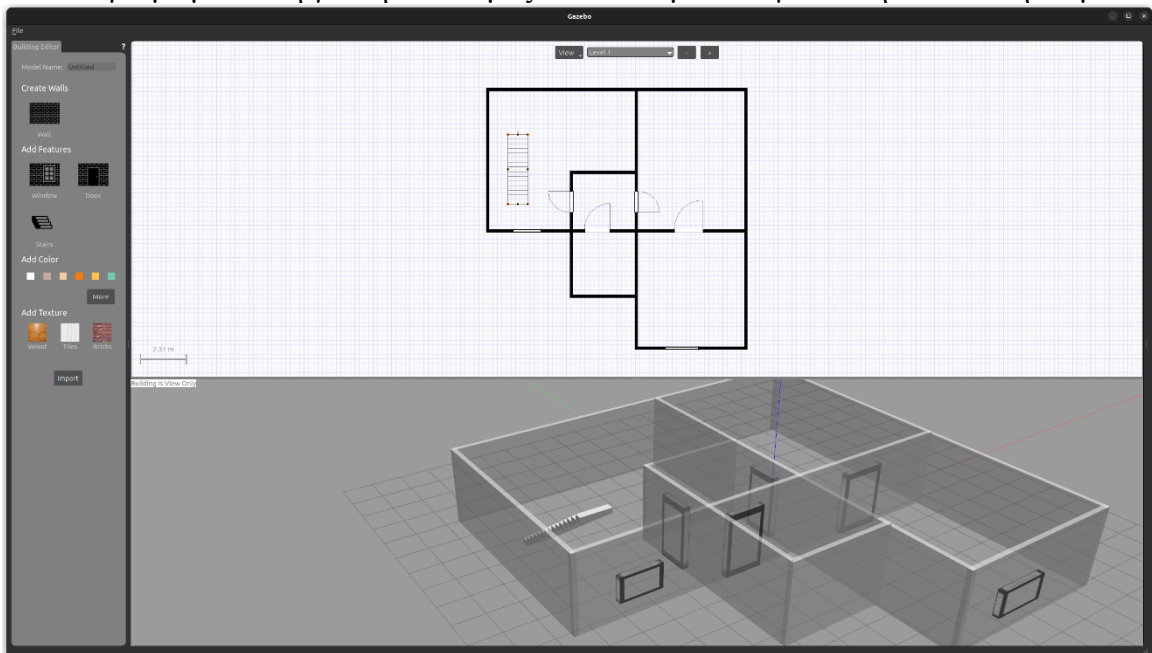


2.9.3.3 Προσθήκη σκάλας

1. Στην Παλέτα, κάνουμε κλικ στην επιλογή Stairs.



2. Καθώς μετακινούμε το ποντίκι στην 2D Προβολή, η σκάλα που θα εισαχθεί κινείται μαζί του, όπως και το αντίστοιχο στοιχείο στην 3D Προβολή.
3. Επιλέγουμε μια θέση για τη σκάλα μας και κάνουμε κλικ για να την τοποθετήσουμε.

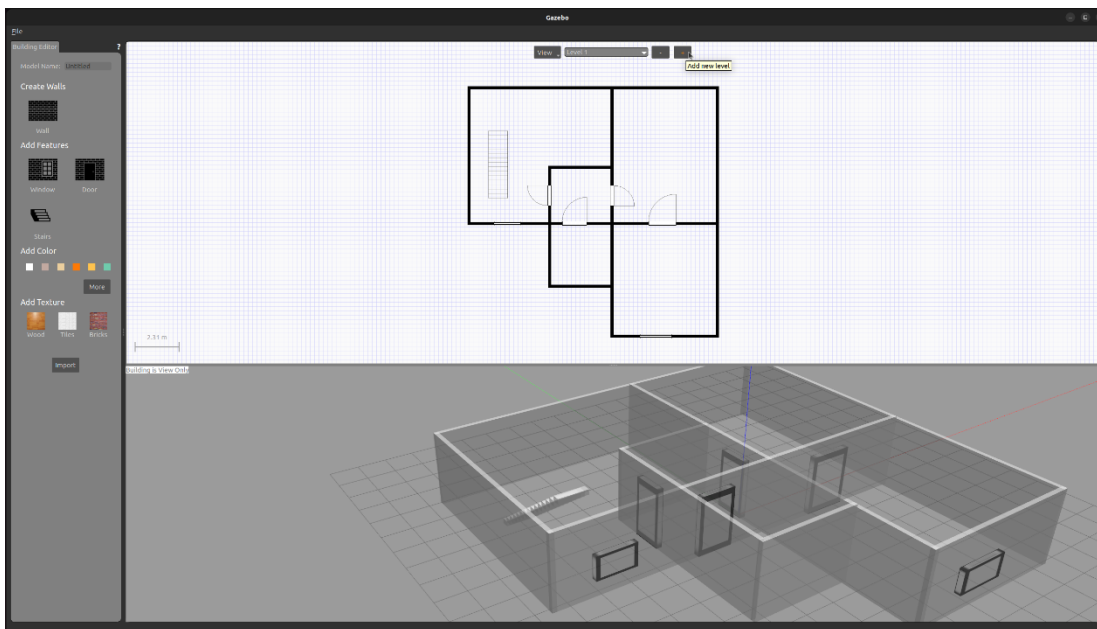


Προσομοίωση Ρομπότ μέσω Υπολογιστή

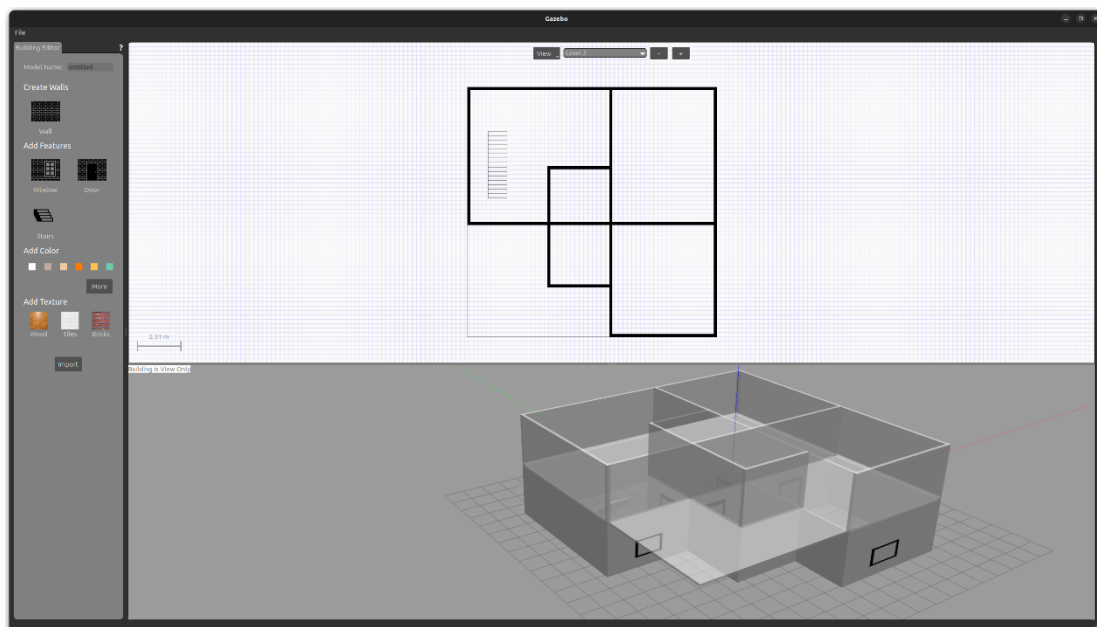
2.9.3.4 Προσθήκη επιπέδων

Έχουμε σχεδόν τελειώσει με το Επίπεδο 1. Θα προσθέσουμε ένα νέο επίπεδο στο κτίριό μας, έτσι ώστε η σκάλα να καταλήξει κάπου.

Στην κορυφή της 2D Προβολής, κάνουμε κλικ στο + για να προσθέσουμε ένα επίπεδο. Εναλλακτικά, κάνουμε δεξί κλικ στην 2D Προβολή και επιλέγουμε Add a level.



Όταν προστίθεται ένα νέο επίπεδο, το δάπεδο εισάγεται αυτόματα. Αν υπάρχουν σκάλες στο επίπεδο από κάτω, θα δημιουργηθεί αυτόματα μια τρύπα πάνω από τη σκάλα όταν αποθηκευτεί το κτίριο.



Σημείωση: Προς το παρόν, όλα τα δάπεδα είναι ορθογώνια.

Συμβουλές:

Πριν προσθέσουμε ένα νέο επίπεδο, βεβαιωνόμαστε ότι έχουμε τοίχους στο τρέχον επίπεδο για να χτίσουμε από πάνω.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Όλοι οι τοίχοι από το επίπεδο από κάτω αντιγράφονται στο νέο επίπεδο με προεπιλεγμένα υλικά. Κανένα άλλο χαρακτηριστικό δεν αντιγράφεται. Μπορούμε να διαγράψουμε χειροκίνητα τους τοίχους που δεν θέλουμε.

2.9.3.5 *Επεξεργασία του κτιρίου μας*

Σημείωση: Πρέπει να είμαστε προσεκτικοί κατά την επεξεργασία του κτιρίου, καθώς ο επεξεργαστής δεν διαθέτει λειτουργία αναίρεσης (undo) των ενεργειών μας.

Όλες οι μετρήσεις γίνονται σε μέτρα.

2.9.3.6 *Αλλαγή επιπέδων*

Αφού προσθέσαμε ένα επίπεδο, μεταφερθήκαμε αυτόματα στο νέο επίπεδο στην 2D Προβολή. Μπορούμε να επιστρέψουμε στο Επίπεδο 1 επιλέγοντάς το από το αναπτυσσόμενο μενού στην κορυφή της 2D Προβολής.

Συμβουλή: Το επίπεδο που είναι επιλεγμένο στην 2D Προβολή θα εμφανίζεται ως ημιδιαφανές στην 3D Προβολή, ενώ όλα τα επίπεδα κάτω από αυτό θα εμφανίζονται αδιαφανή. Τα επίπεδα πάνω από το τρέχον θα είναι κρυφά, αλλά παραμένουν μέρος του κτιρίου!

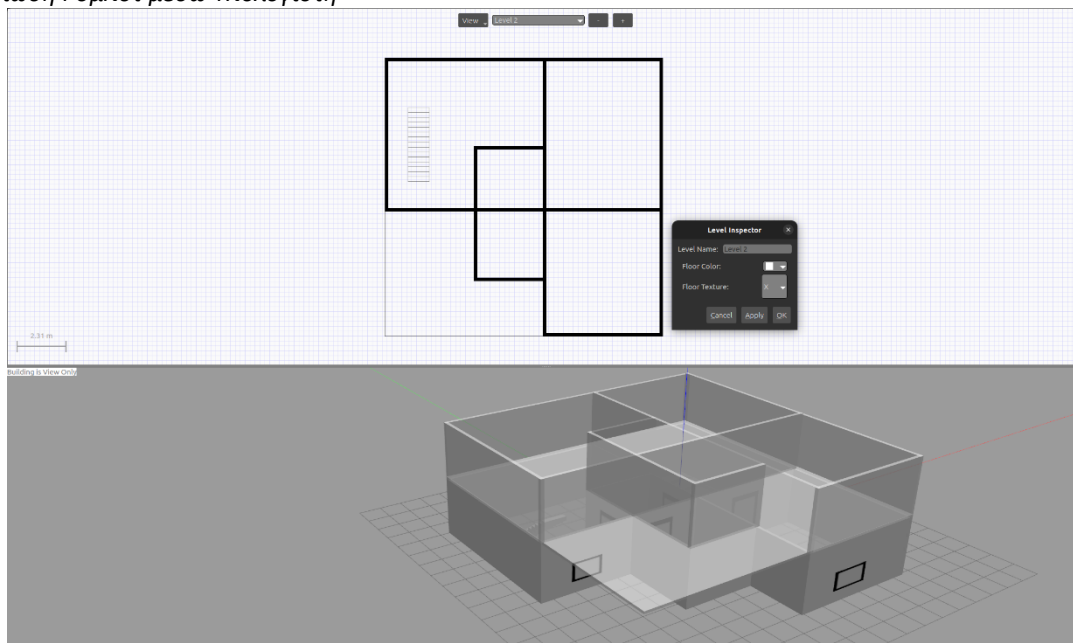
2.9.3.7 *Επεξεργασία διαμόρφωσης επιπέδων*

Μπορούμε επίσης να επεξεργαστούμε τις ρυθμίσεις των επιπέδων εάν το επιθυμούμε.

- Κάνουμε διπλό κλικ στη 2D Προβολή για να ανοίξουμε τον Inspector με επιλογές διαμόρφωσης επιπέδων. Εναλλακτικά, κάνουμε δεξί κλικ και επιλέγουμε Open Level Inspector.

Αν έχουμε προσθέσει επίπεδα που δεν θέλουμε ή αν έχουμε κάνει λάθος στο τρέχον επίπεδο και θέλουμε να το ξεκινήσουμε από την αρχή, μπορούμε να διαγράψουμε το τρέχον επίπεδο.

- Για να διαγράψουμε το τρέχον επίπεδο πατάμε το κουμπί - στην κορυφή της 2D Προβολής ή κάνουμε δεξί κλικ και επιλέγουμε Delete Level.



2.9.3.8 Επεξεργασία τοίχων

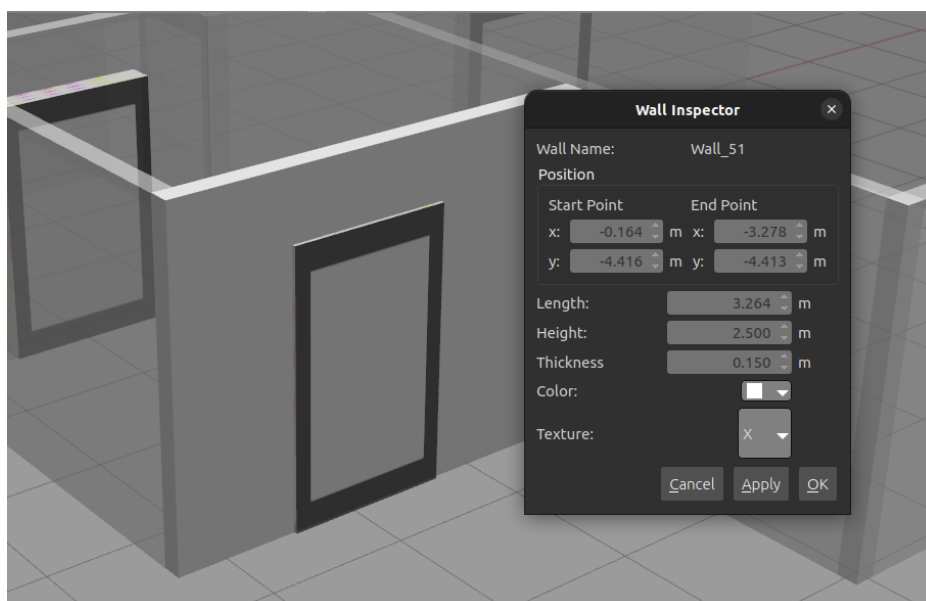
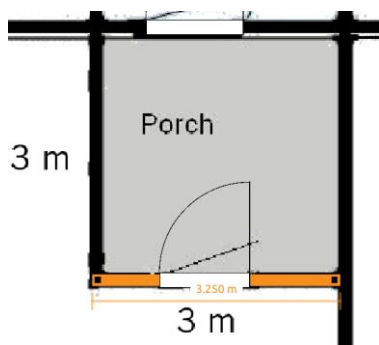
Έχουμε σχεδιάσει πολλούς τοίχους, αλλά ίσως να μην είναι ακριβώς όπως τους θέλαμε. Για να τους επεξεργαστούμε:

- Στη 2D Προβολή, κάνουμε κλικ στον τοίχο που θέλουμε να επεξεργαστούμε.
 - a. Μετακινούμε το τοίχο σύροντάς τον σε μια νέα θέση.
 - b. Αλλάζουμε το μέγεθος ή περιστρέφουμε τον τοίχο σύροντας ένα από τα άκρα του.

Συμβουλή: Από προεπιλογή, οι τοίχοι κουμπώνουν σε γωνίες 15° και αυξήσεις 0,25 μέτρων. Για να παρακάμψουμε αυτό, κρατάμε πατημένο το Shift ενώ σχεδιάζουμε.
- Κάνουμε διπλό κλικ σε έναν τοίχο στην 2D Προβολή για να ανοίξουμε τον Inspector με επιλογές διαμόρφωσης. Εναλλακτικά, κάνουμε δεξί κλικ και επιλέγουμε Open Wall Inspector. Επεξεργαζόμαστε τα πεδία και πατάμε Apply για να δούμε τις αλλαγές.
- Για να διαγράψουμε έναν τοίχο, είτε πατάμε το πλήκτρο Delete όσο είναι επιλεγμένος, είτε κάνουμε δεξί κλικ στον τοίχο και επιλέγουμε Delete.

Συμβουλή: Η επεξεργασία ενός τοίχου λαμβάνει υπόψη και τους συνημμένους τοίχους.

Συμβουλή: Η διαγραφή ενός τοίχου διαγράφει αυτόματα όλες τις πόρτες και τα παράθυρα που είναι προσαρτημένα σε αυτόν.

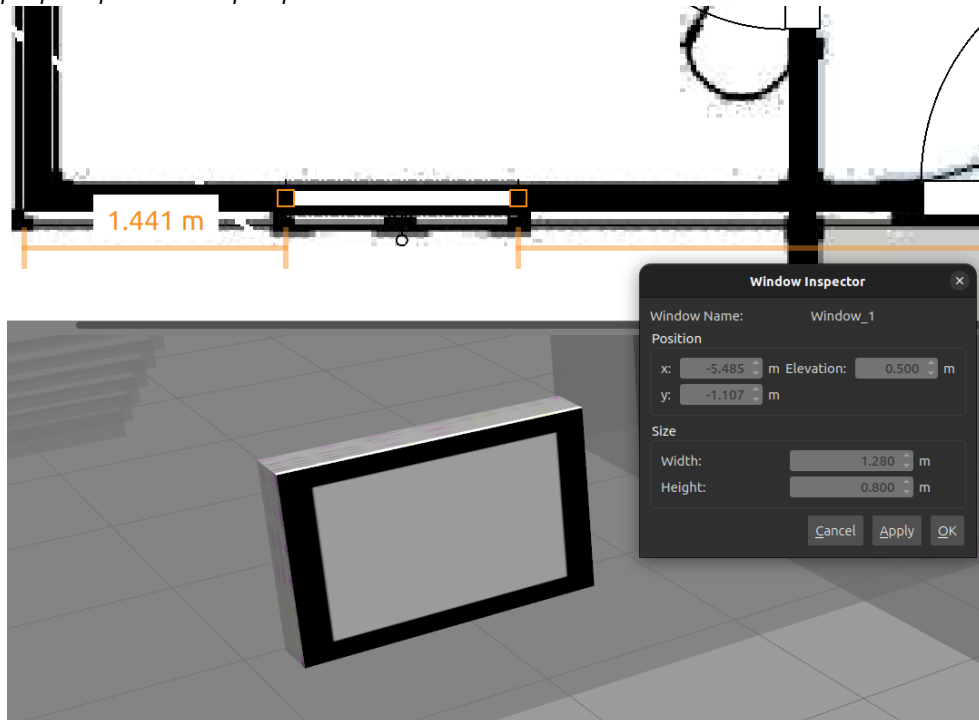


2.9.3.9 Επεξεργασία παραθύρων και πορτών

Ας παίξουμε τώρα με τα παράθυρα και τις πόρτες. Όπως κάναμε για τους τοίχους, μπορούμε να επεξεργαστούμε παράθυρα και πόρτες με μεγαλύτερη ακρίβεια με διάφορους τρόπους.

- Στην 2D Προβολή, κάνουμε κλικ στο στοιχείο που θέλουμε να επεξεργαστούμε.
 - a. Μετακινούμε το στοιχείο σύροντάς το σε νέα θέση. Θυμόμαστε ότι τα παράθυρα και οι πόρτες κουμπώνουν αυτόματα στους τοίχους, και δεν έχει νόημα να τα τοποθετήσουμε μακριά από αυτούς, καθώς αντιπροσωπεύουν ανοίγματα στον τοίχο.
 - b. Περιστρέφουμε το στοιχείο σύροντας τη λαβή περιστροφής του. Προς το παρόν, η κατεύθυνση δεν έχει σημασία όσο το στοιχείο είναι συνδεδεμένο σε τοίχο.
 - c. Αλλάζουμε το πλάτος του στοιχείου σύροντας ένα από τα άκρα του.
- Κάνουμε διπλό κλικ στο στοιχείο στην 2D Προβολή για να ανοίξουμε τον Inspector με επιλογές διαμόρφωσης. Εναλλακτικά, κάνουμε δεξί κλικ και επιλέγουμε Open Window/Door Inspector.
- Για να διαγράψουμε ένα στοιχείο, είτε πατάμε το πλήκτρο Delete όσο είναι επιλεγμένο, είτε κάνουμε δεξί κλικ στο στοιχείο και επιλέγουμε Delete.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

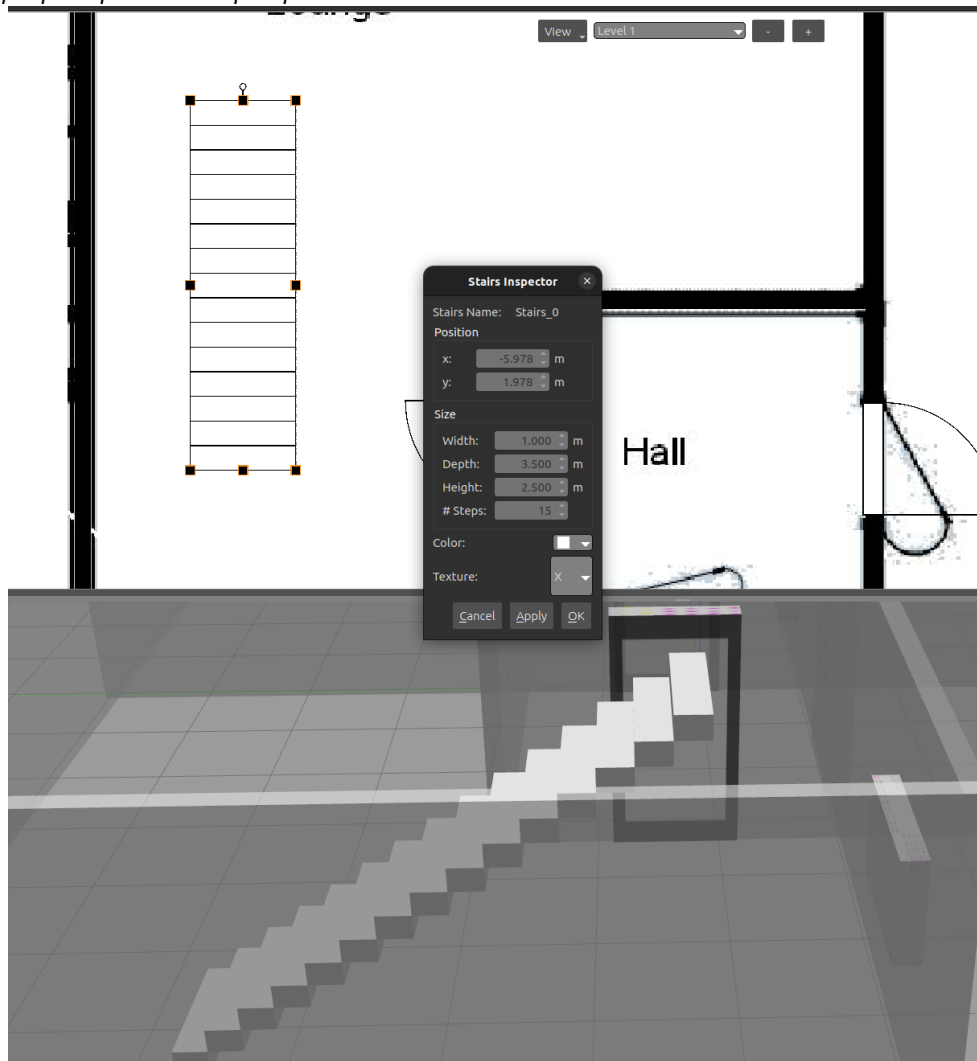


2.9.3.10 Επεξεργασία σκάλας

Τέλος, θα επεξεργαστούμε τη σκάλα που εισαγάγαμε νωρίτερα. Δεδομένου ότι δεν βρίσκεται στην κάτοψη, μπορούμε να είμαστε δημιουργικοί και να αλλάξουμε το μέγεθός της όπως θέλουμε.

- Στην 2D Προβολή, κάνουμε κλικ στη σκάλα για να την επιλέξουμε.
 - a. Μετακινούμε τη σκάλα σύροντάς την σε μια νέα θέση.
 - b. Περιστρέφουμε τη σκάλα σε πολλαπλάσια των 90° σύροντας τη λαβή περιστροφής.
 - c. Αλλάζουμε το μέγεθος της σκάλας σύροντας έναν από τους κόμβους των άκρων της.
- Κάνουμε διπλό κλικ στη σκάλα στην 2D Προβολή για να ανοίξουμε τον Inspector με επιλογές διαμόρφωσης. Εναλλακτικά, κάνουμε δεξί κλικ και επιλέγουμε Open Stairs Inspector.
- Για να διαγράψουμε τη σκάλα, είτε πατάμε το πλήκτρο Delete όσο είναι επιλεγμένη, είτε κάνουμε δεξί κλικ και επιλέγουμε Delete.

Συμβουλή: Στην 2D Προβολή, οι σκάλες είναι ορατές τόσο στο αρχικό όσο και στο τελικό επίπεδο.



2.9.3.11 Προσθήκη χρωμάτων και υφών

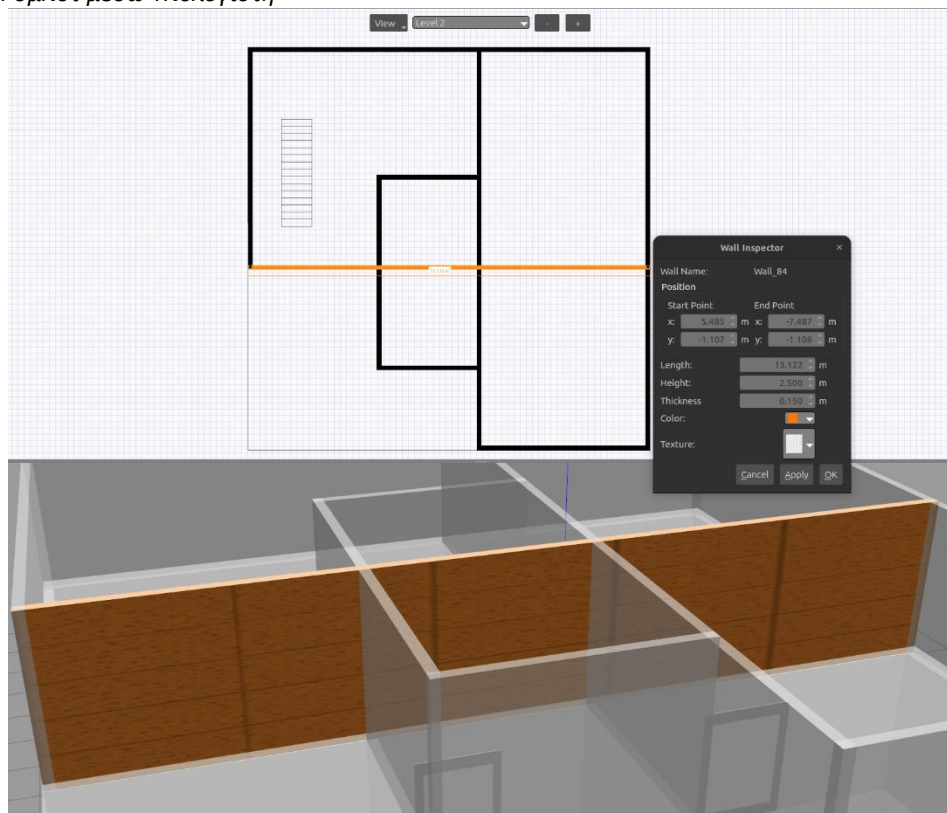
Τώρα που όλα τα στοιχεία έχουν τοποθετηθεί και έχουν το σωστό μέγεθος, μπορούμε να προσθέσουμε χρώματα και υφές στους τοίχους, τα δάπεδα και τις σκάλες. Θυμόμαστε ότι τα παράθυρα και οι πόρτες είναι απλά ανοίγματα στους τοίχους και επομένως δεν μπορούν να έχουν υλικά.

Συμβουλή: Το προεπιλεγμένο χρώμα είναι το λευκό και η προεπιλεγμένη υφή είναι καμία.

Υπάρχουν δύο τρόποι για να προσθέσουμε χρώματα και υφές στο κτίριό μας.

Από τους Inspectors

Μπορούμε να προσθέσουμε χρώμα και υφή στους τοίχους, τις σκάλες και τα δάπεδα χρησιμοποιώντας τον Wall Inspector, Stairs Inspector και Level Inspector αντίστοιχα. Απλώς ανοίγουμε τον inspector, επιλέγουμε τα υλικά μας και πατάμε Apply για να εφαρμοστούν οι αλλαγές.

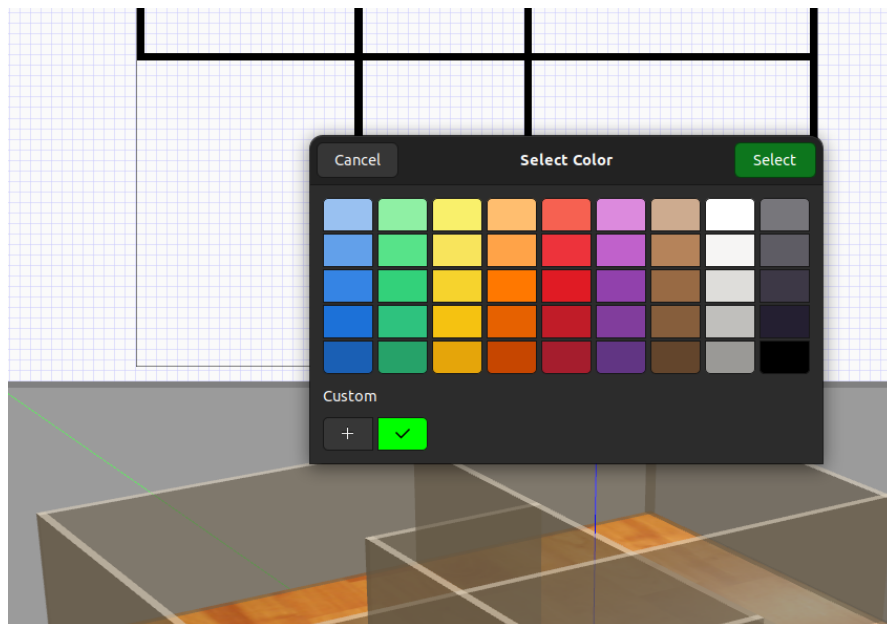
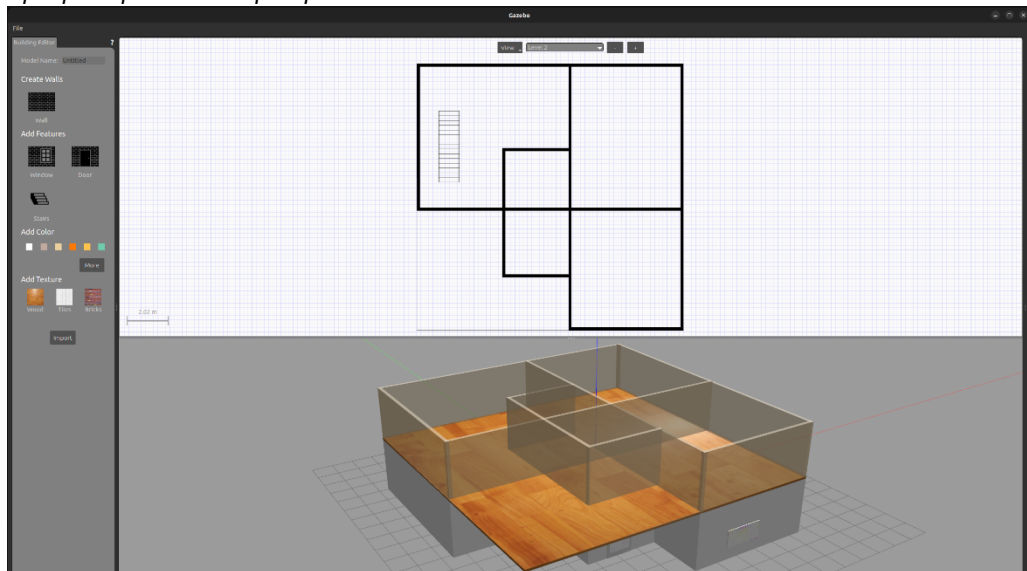


2.9.3.12 Από την Παλέτα

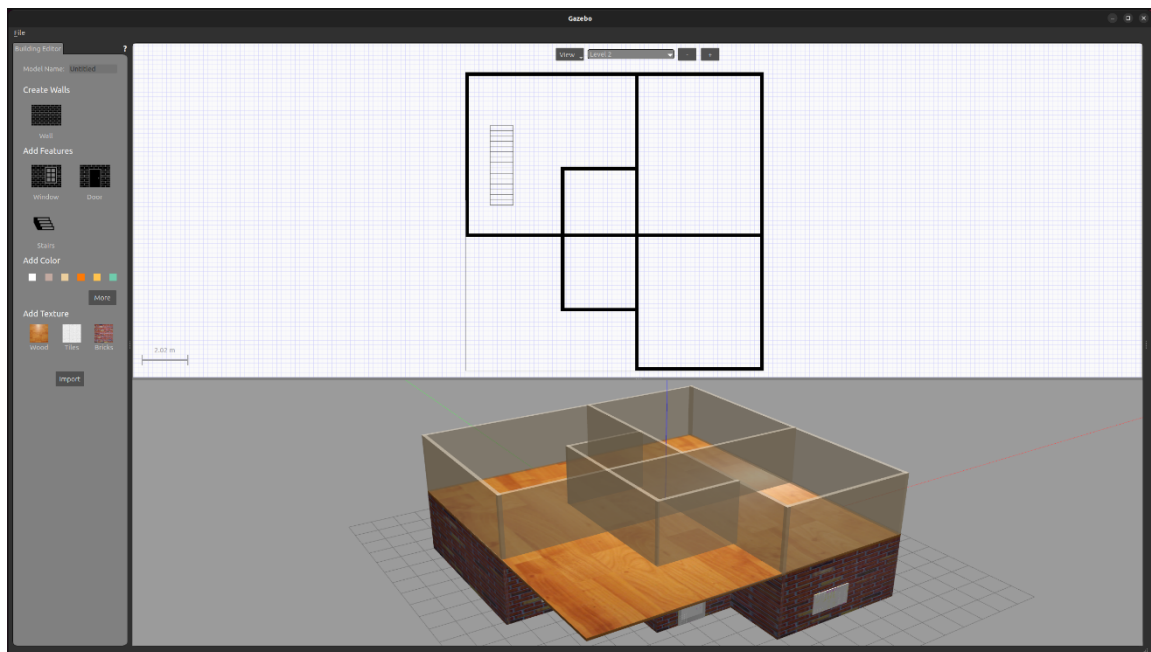
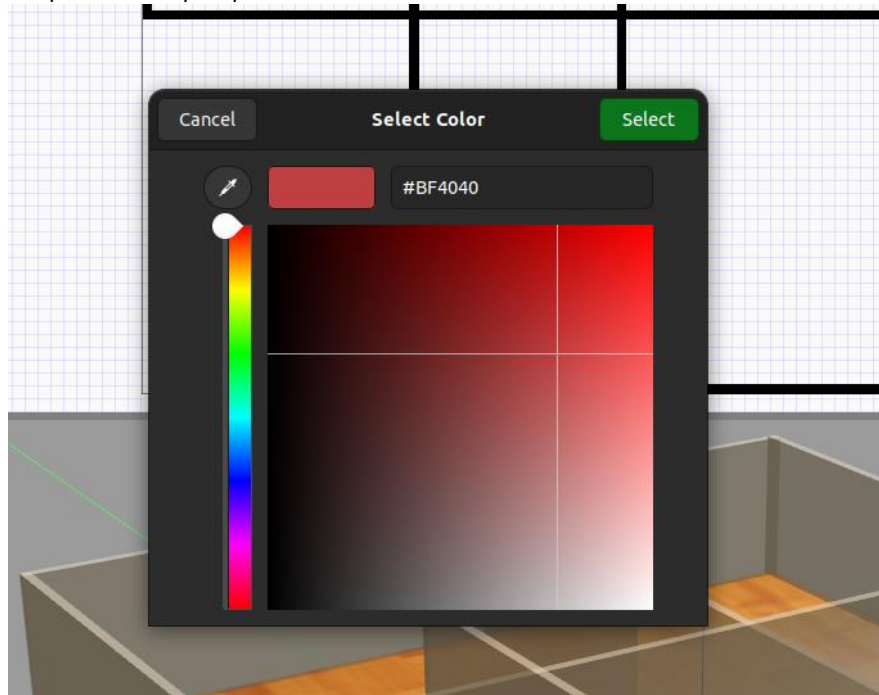
Μπορούμε να επιλέξουμε χρώματα και υφές από την Παλέτα και να τα αντιστοιχίσουμε σε αντικείμενα στο κτίριό μας κάνοντας κλικ πάνω τους στην 3D Προβολή.

1. Κάνουμε κλικ σε ένα χρώμα ή υφή από την Παλέτα.
2. Καθώς μετακινούμε το ποντίκι στην 3D Προβολή, τα στοιχεία πάνω από τα οποία περνάει το ποντίκι θα επισημαίνονται, δείχνοντας μια προεπισκόπηση του επιλεγμένου υλικού.
3. Κάνοντας κλικ στο επισημασμένο στοιχείο, εφαρμόζουμε το επιλεγμένο υλικό σε αυτό. Μπορούμε να κάνουμε κλικ σε όσα στοιχεία θέλουμε για να τους αναθέσουμε το ίδιο υλικό.
4. Όταν τελειώσουμε με το επιλεγμένο υλικό, είτε κάνουμε δεξί κλικ στην 3D Προβολή, είτε κάνουμε κλικ έξω από οποιοδήποτε στοιχείο για να εξέλθουμε από τη λειτουργία ανάθεσης υλικών.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

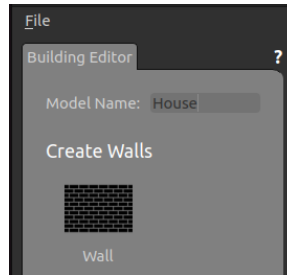


Προσομοίωση Ρομπότ μέσω Υπολογιστή



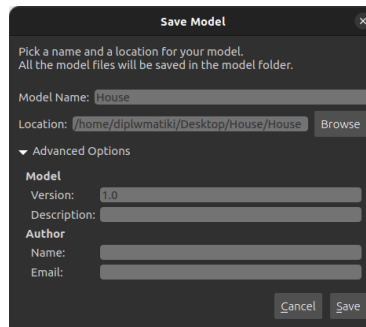
2.9.3.13 Αποθήκευση του κτιρίου μας

Η αποθήκευση θα δημιουργήσει έναν φάκελο, καθώς και αρχεία SDF και config για το κτίριό μας. Πριν αποθηκεύσουμε, δίνουμε στο κτίριό μας ένα όνομα από την Παλέτα.



Στο επάνω μενού, επιλέγουμε File και στη συνέχεια Save As (ή πατάμε Ctrl+S). Θα εμφανιστεί ένα παράθυρο διαλόγου όπου μπορούμε να επιλέξουμε την τοποθεσία για το μοντέλο μας.

Συμβουλή: Στην ενότητα Advanced Options, μπορούμε να ορίσουμε κάποια μεταδεδομένα (meta-data) για το κτίριό μας.



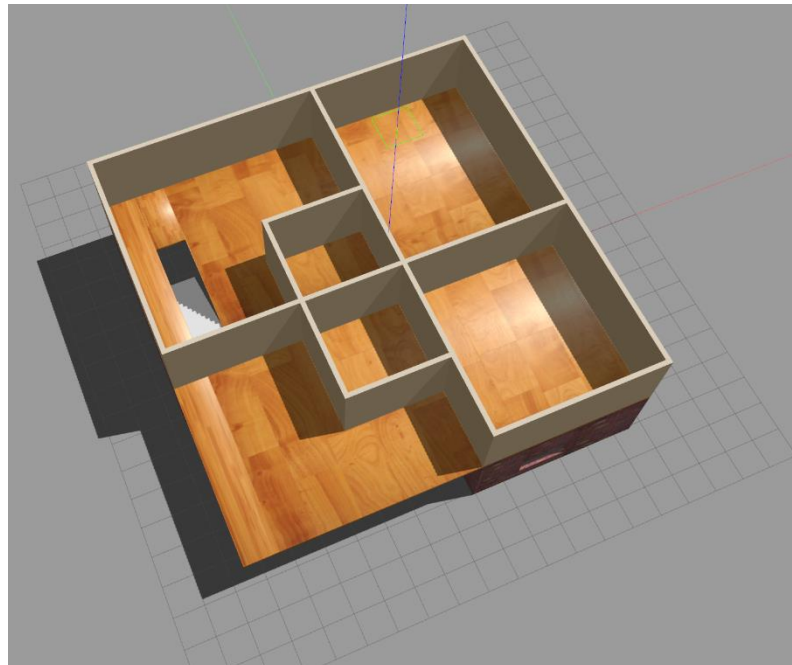
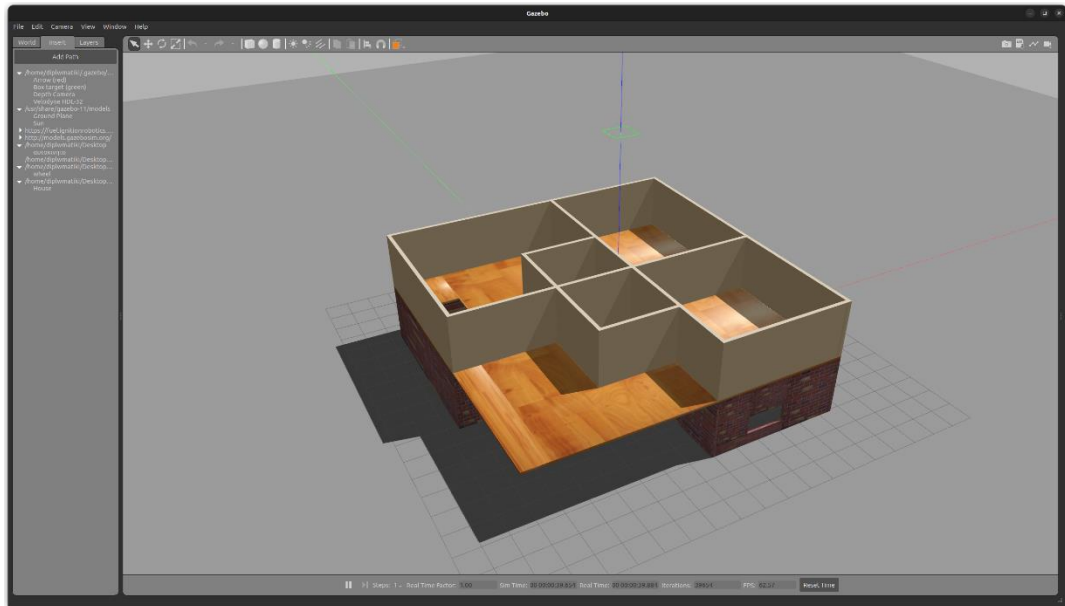
2.9.3.14 Έξοδος

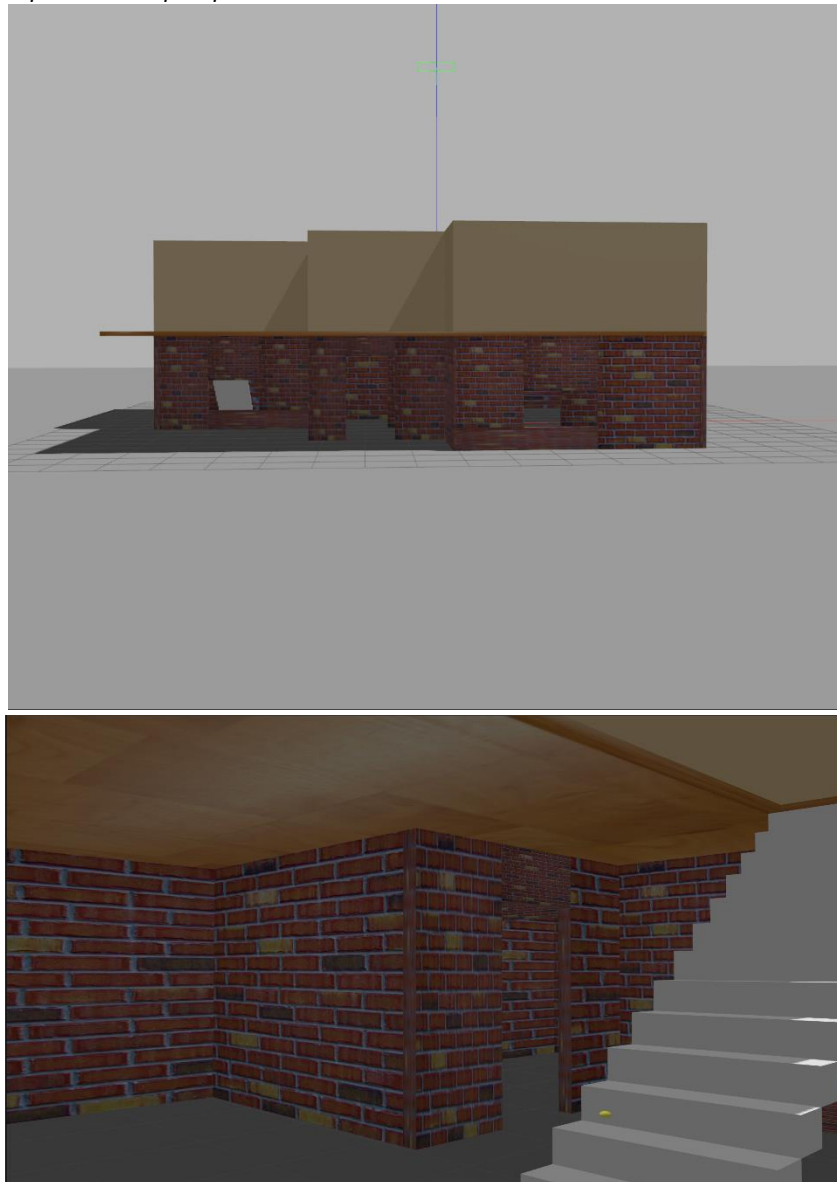
Σημείωση: Μόλις βγούμε από τον Επεξεργαστή Κτιρίων (Building Editor), το κτίριο δεν θα είναι πλέον επεξεργάσιμο.

Όταν ολοκληρώσουμε τη δημιουργία του κτιρίου μας και το έχουμε αποθηκεύσει, πηγαίνουμε στο μενού File και επιλέγουμε Exit Building Editor.

Το κτίριό μας θα εμφανιστεί στο κύριο παράθυρο. Στο μέλλον, μπορούμε να το βρούμε στην καρτέλα Insert.

Προσομοίωση Ρομπότ μέσω Υπολογιστή





2.10 Καταγραφή και αναπαραγωγή (Logging and Playback)

2.10.1 Αρχεία καταγραφής Gazebo

Τα αρχεία καταγραφής του Gazebo είναι συμπιεσμένα αρχεία .log, τα οποία περιέχουν μια αρχική πλήρη περιγραφή ολόκληρου του κόσμου, ακολουθούμενη από μια σειρά από "καταστάσεις κόσμου" (world states).

Η αρχική περιγραφή περιέχει πλήρεις πληροφορίες για τα πάντα στον κόσμο, από τη σκηνή μέχρι τις οντότητες που υπάρχουν.

Καταστάσεις κόσμου: Καταγράφονται κάθε φορά που κάτι αλλάζει στην προσομοίωση και περιέχουν πληροφορίες μόνο για ό,τι άλλαξε, όπως:

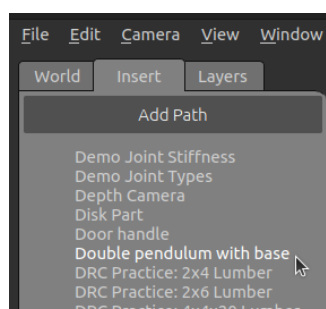
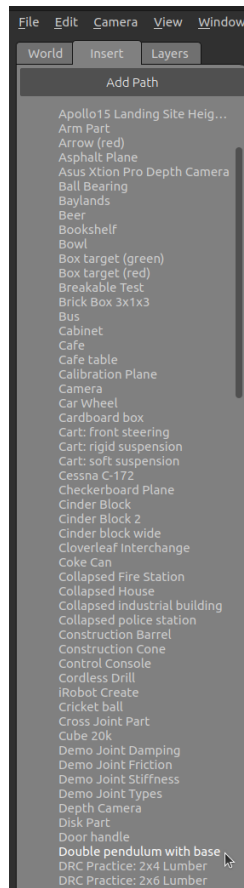
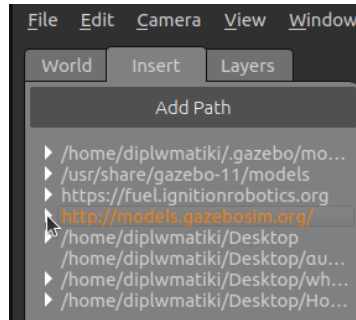
Στατιστικά προσομοίωσης, όπως η τρέχουσα ώρα προσομοίωσης και ο αριθμός των φυσικών επαναλήψεων.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Η τρέχουσα κατάσταση κάθε μοντέλου στη σκηνή, καθώς και οι καταστάσεις κάθε συνδέσμου και άρθρωσης. Αυτό περιλαμβάνει στιγμιαία θέση, ταχύτητα, επιτάχυνση και δυνάμεις.

Η τρέχουσα θέση κάθε φωτός στον κόσμο.

Για το παράδειγμα μας θα χρησιμοποιήσουμε ένα διπλό εκκρεμές. Κάνουμε κλικ στη καρτέλα Insert και έπειτα στο <http://models.gazebosim.org/>. Στη λίστα που θα μας ανοίξει θα επιλέξουμε το **double pendulum with base**.

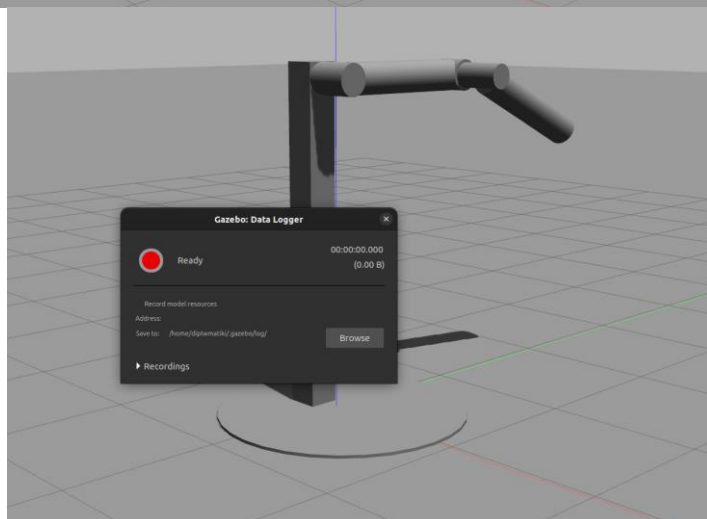
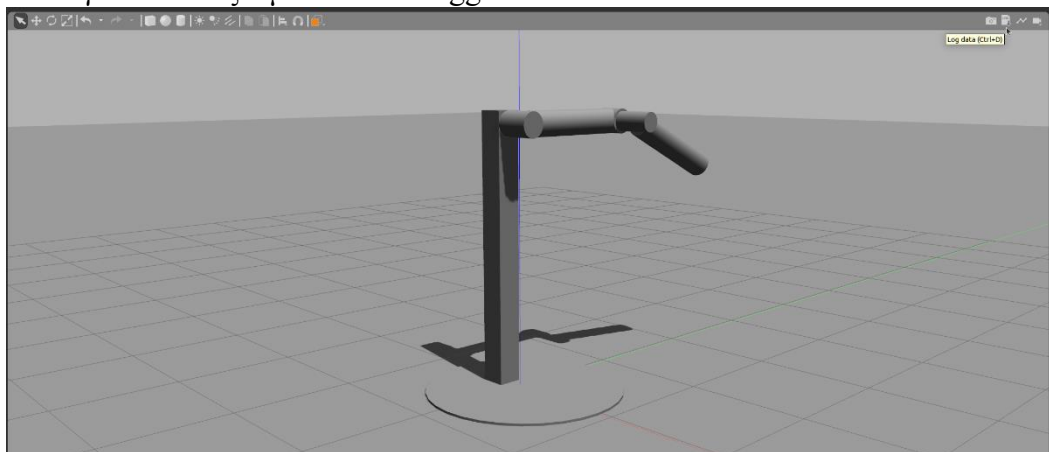




2.10.2 Καταγραφή αρχείου καταγραφής

Καταγραφή μέσω της γραφικής διεπαφής (GUI)

1. Ξεκινάμε την προσομοίωσή μας. Για παράδειγμα, μπορούμε να έχουμε έναν απλό κόσμο με ένα διπλό εκκρεμές.
2. Κάνουμε κλικ στο εικονίδιο καταγραφής που βρίσκεται στο επάνω δεξί μέρος ή πατάμε Ctrl+D για να ανοίξουμε το Data Logger.

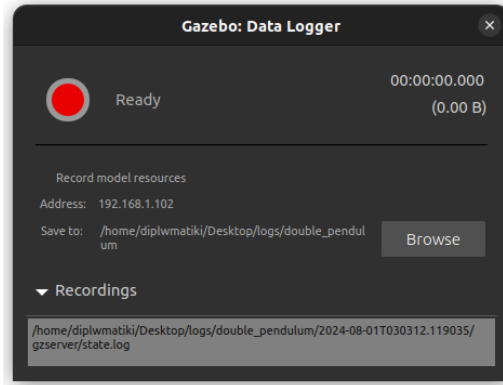


Προσομοίωση Ρομπότ μέσω Υπολογιστή

- Μπορούμε να επιλέξουμε τον φάκελο στον οποίο θα αποθηκευτεί το αρχείο καταγραφής μας κάνοντας κλικ στο κουμπί Browse. Από προεπιλογή, τα αρχεία καταγραφής αποθηκεύονται στον φάκελο `~/gazebo/log`. Σε αυτό το παράδειγμα, θα το αποθηκεύσουμε στον φάκελο `~/logs/double_pendulum/`.
- Κάνουμε κλικ στο κόκκινο κουμπί για να ξεκινήσουμε την καταγραφή. Στα δεξιά, θα δούμε τον αριθμό των byte στο αρχείο καταγραφής να αυξάνεται καθώς η καταγραφή προχωρά.

Σημείωση: Για λόγους αποδοτικότητας, καταγράφονται μόνο τα μοντέλα και τα φώτα που κινούνται με την πάροδο του χρόνου. Αν η σκηνή είναι στατική, ο αριθμός των byte στο αρχείο καταγραφής δεν θα αυξάνεται. Αυτό σημαίνει επίσης ότι ο αριθμός των δειγμάτων στο αρχείο καταγραφής μπορεί να διαφέρει από τον αριθμό των επαναλήψεων στην προσομοίωση.

- Κάνουμε ξανά κλικ στο κόκκινο κουμπί για να σταματήσουμε την καταγραφή.
- Επεκτείνουμε την ενότητα Recordings για να δούμε τη διαδρομή προς το αρχείο `state.log` που δημιουργήθηκε. Θα βρισκείται μέσα σε έναν κατάλογο με χρονική σήμανση.



2.10.3 Αναπαραγωγή αρχείου καταγραφής

Αφού έχουμε ένα αρχείο καταγραφής, μπορούμε να το αναπαράγουμε οπτικά ή να το εξετάσουμε με διάφορους τρόπους.

2.10.4 Οπτική αναπαραγωγή στο GUI

Προς το παρόν, δεν είναι δυνατή η άμεση αναπαραγωγή αρχείου καταγραφής μέσω της GUI. Η αναπαραγωγή πρέπει να ξεκινήσει από τη γραμμή εντολών. Για να ξεκινήσουμε την αναπαραγωγή, ανοίγουμε το τερματικό και εκκινούμε το Gazebo με την επιλογή `-p` για να ορίσουμε ένα αρχείο καταγραφής. Για παράδειγμα, το αρχείο που καταγράψαμε προηγουμένως:

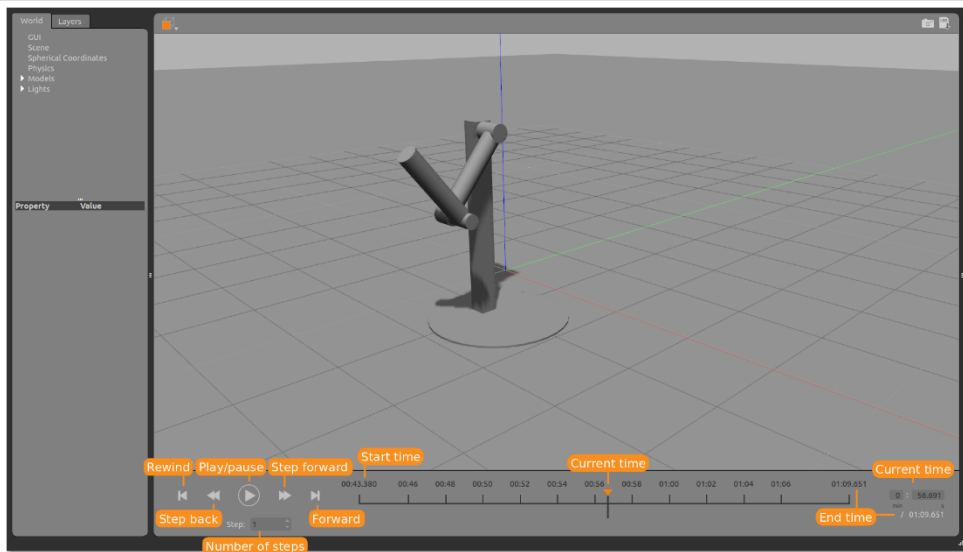
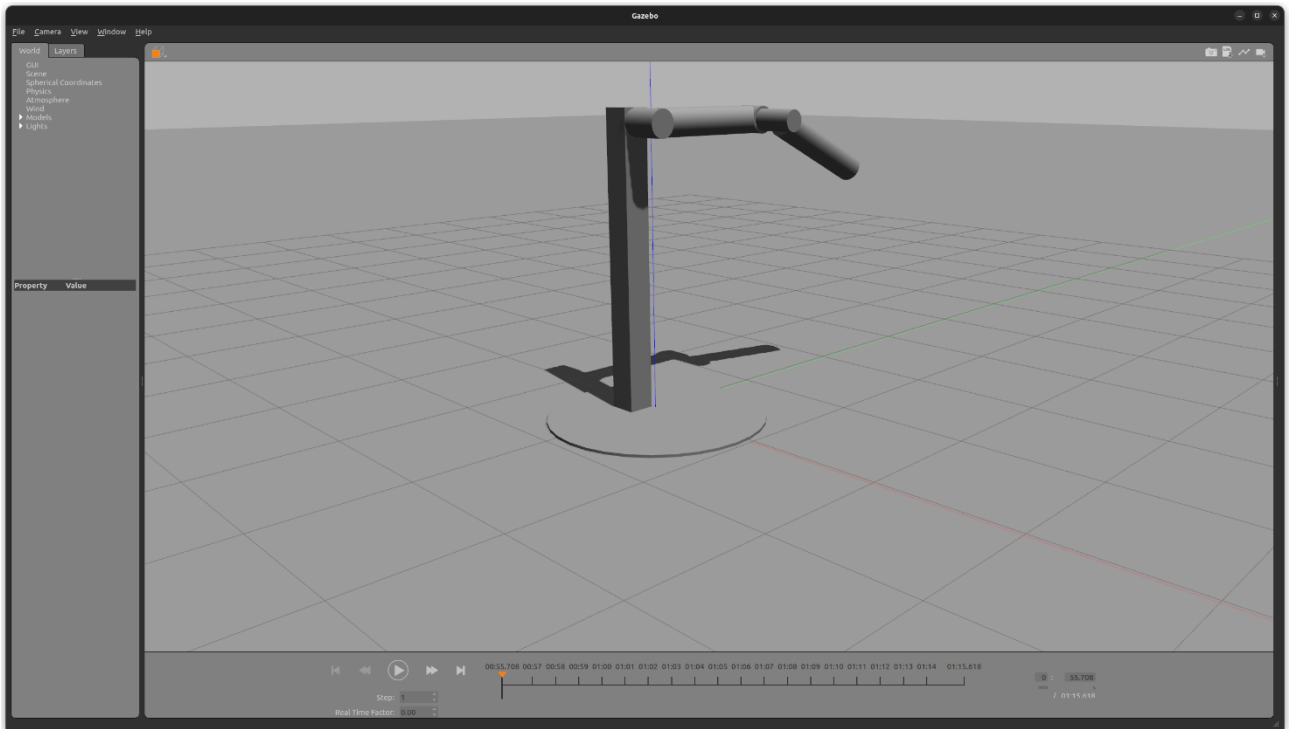
```
gazebo -u -p ~/Desktop/logs/double_pendulum/2024-08-01T032741.922666/gzserver/state.log
```

Συμβουλή: Η επιλογή `-u` ξεκινά την αναπαραγωγή σε κατάσταση παύσης.

Το Gazebo θα ανοίξει σε λειτουργία αναπαραγωγής. Μπορούμε να αναπαράγουμε, να κάνουμε παύση, να κάνουμε γρήγορη επαναφορά ή να προχωρήσουμε βήμα-βήμα μέσα από την αναπαραγωγή.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

```
diplwmatiki@diplwmatiki: ~  
bash: /opt/ros/noetic/setup.bash: No such file or directory  
bash: /opt/ros/melodic/setup.bash: No such file or directory  
diplwmatiki@diplwmatiki:~$ gazebo -u -p ~/Desktop/logs/double_pendulum/2024-08-0  
1T032741.922666/gzserver/state.log  
diplwmatiki@diplwmatiki:~$
```



- Χρησιμοποιούμε τα κουμπιά Play / Pause για να σταματήσουμε την αναπαραγωγή.
- Χρησιμοποιούμε τα κουμπιά Rewind / Forward για να μεταβούμε στην αρχή ή το τέλος του

Προσομοίωση Ρομπότ μέσω Υπολογιστή αρχείου.

- Χρησιμοποιούμε τα κουμπιά Step back / Step forward για να παραλείψουμε δείγματα. Ο αριθμός των δειγμάτων που παραλείπονται κάθε φορά που πατάμε ένα κουμπί βήματος μπορεί να αλλάξει στο πλαίσιο παρακάτω. Τα δείγματα μπορεί να απέχουν οποιονδήποτε αριθμό επαναλήψεων και δευτερολέπτων μεταξύ τους.
- Σύρουμε τον δείκτη της τρέχουσας χρονικής στιγμής και τον αφήνουμε σε άλλο σημείο για να μετακινηθούμε στο αρχείο καταγραφής.
- Εισάγουμε έναν συγκεκριμένο χρόνο στα δεξιά για να μεταβούμε σε αυτό το δείγμα.

2.11 Δομή μοντέλου και απαιτήσεις

Επισκόπηση

Το Gazebo μπορεί να φορτώσει δυναμικά μοντέλα στη προσομοίωση είτε προγραμματιστικά είτε μέσω του GUI. Τα μοντέλα υπάρχουν στον υπολογιστή μας, αφού έχουν κατέβει ή δημιουργηθεί από εμάς. Αυτό το εκπαιδευτικό υλικό περιγράφει τη δομή των φακέλων μοντέλων του Gazebo και τα απαραίτητα αρχεία μέσα σε έναν φάκελο μοντέλου.

Τα μοντέλα στο Gazebo ορίζουν μια φυσική οντότητα με δυναμικές, κινηματικές και οπτικές ιδιότητες. Επιπλέον, ένα μοντέλο μπορεί να έχει ένα ή περισσότερα πρόσθετα (plugins), τα οποία επηρεάζουν τη συμπεριφορά του. Ένα μοντέλο μπορεί να αναπαριστά οτιδήποτε, από ένα απλό σχήμα μέχρι ένα σύνθετο ρομπότ. Ακόμα και το έδαφος είναι ένα μοντέλο.

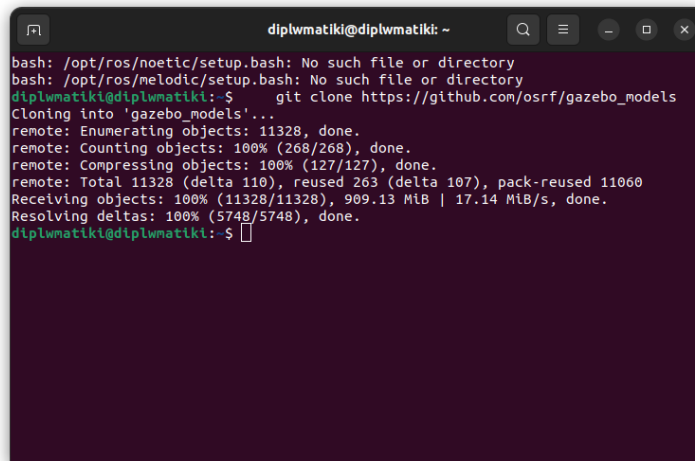
Το Gazebo βασίζεται σε μια βάση δεδομένων για την αποθήκευση και τη συντήρηση των μοντέλων που είναι διαθέσιμα για χρήση στην προσομοίωση. Η βάση δεδομένων μοντέλων είναι ένας πόρος που υποστηρίζεται από την κοινότητα, οπότε παρακαλούμε να ανεβάζουμε και να συντηρούμε τα μοντέλα που δημιουργούμε και χρησιμοποιούμε.

2.12 Το Αποθετήριο της Βάσης Δεδομένων Μοντέλων

Η βάση δεδομένων μοντέλων είναι ένα αποθετήριο στο GitHub που βρίσκεται εδώ.

Μπορούμε να κλωνοποιήσουμε το αποθετήριο χρησιμοποιώντας:

```
git clone https://github.com/osrf/gazebo_models
```



```
diplwmatiki@diplwmatiki: ~  
bash: /opt/ros/noetic/setup.bash: No such file or directory  
bash: /opt/ros/melodic/setup.bash: No such file or directory  
diplwmatiki@diplwmatiki: $ git clone https://github.com/osrf/gazebo_models  
Cloning into 'gazebo_models'...  
remote: Enumerating objects: 11328, done.  
remote: Counting objects: 100% (268/268), done.  
remote: Compressing objects: 100% (127/127), done.  
remote: Total 11328 (delta 110), reused 263 (delta 107), pack-reused 11060  
Receiving objects: 100% (11328/11328), 909.13 MiB | 17.14 MiB/s, done.  
Resolving deltas: 100% (5748/5748), done.  
diplwmatiki@diplwmatiki: $
```

2.12.1 Βάση Δεδομένων

- **Database**

- **database.config**: Μεταδεδομένα για τη βάση δεδομένων. Αυτό τώρα δημιουργείται αυτόματα από το CMakeLists.txt
- **model_1**: Ένας φάκελος για το model_1
 - **model.config**: Μεταδεδομένα για το model_1
 - **model.sdf**: SDF περιγραφή του μοντέλου
 - **model.sdf.erb**: Ruby ενσωματωμένη περιγραφή SDF του μοντέλου
 - **meshes**: Ένας φάκελος για όλα τα αρχεία COLLADA και STL
 - **materials**: Ένας φάκελος που πρέπει να περιέχει μόνο τους υποφακέλους textures και scripts
 - **textures**: Ένας φάκελος για αρχεία εικόνων (jpg, png, κ.λπ.)
 - **scripts**: Ένας φάκελος για OGRE scripts υλικών
 - **plugins**: Ένας φάκελος για τα αρχεία πηγαίου κώδικα και κεφαλίδων των plugins

2.12.1.1 Φάκελος Plugins

Είναι ένας προαιρετικός φάκελος που περιέχει όλα τα plugins για το μοντέλο.

2.12.1.2 Φάκελος Meshes

Είναι ένας προαιρετικός φάκελος που περιέχει όλα τα αρχεία COLLADA και/ή STL για το μοντέλο.

2.12.1.3 Φάκελος Υλικών (Material)

Είναι ένας προαιρετικός φάκελος που περιέχει όλες τις υφές, εικόνες και OGRE scripts για το μοντέλο. Οι εικόνες υφής πρέπει να τοποθετηθούν στον υποφάκελο textures, και τα αρχεία OGRE scripts στον φάκελο scripts.

2.12.1.4 Διαμόρφωση Βάσης Δεδομένων (Database Config)

Αυτό είναι το αρχείο database.config στη ρίζα της βάσης δεδομένων μοντέλων. Το αρχείο αυτό περιέχει πληροφορίες σχετικά με την άδεια χρήσης των μοντέλων, ένα όνομα για τη βάση δεδομένων και μια λίστα με όλα τα έγκυρα μοντέλα.

Σημείωση: Το αρχείο database.config είναι απαραίτητο μόνο για διαδικτυακά αποθετήρια. Ένας

φάκελος γεμάτος μοντέλα στον τοπικό υπολογιστή μας δεν χρειάζεται το αρχείο database.config.

Η μορφή αυτού του database.config είναι:

```
<?xml version='1.0'?>
<database>
  <name>name_of_this_database</name>
  <license>Creative Commons Attribution 3.0 Unported</license>
  <models>
    <uri>file://model_directory</uri>
  </models>
</database>
```

- <name>

Το όνομα της βάσης δεδομένων. Αυτό χρησιμοποιείται από το GUI και άλλα εργαλεία.

- <license>

Η άδεια χρήσης για τα μοντέλα μέσα στη βάση δεδομένων.

- <models>

Μια λίστα με όλα τα URIs των μοντέλων μέσα στη βάση δεδομένων.

- <uri>

Το URI για ένα μοντέλο, αυτό πρέπει να είναι της μορφής file://model_directory_name

2.12.1.5 Διαμόρφωση Μοντέλου (Model Config)

Κάθε μοντέλο πρέπει να έχει ένα αρχείο model.config στη ρίζα του φακέλου του μοντέλου, το οποίο περιέχει μεταπληροφορίες σχετικά με το μοντέλο.

Η μορφή αυτού του model.config είναι:

```
<?xml version="1.0"?>
<model>
  <name>My Model Name</name>
  <version>1.0</version>
  <sdf version='1.5'>model.sdf</sdf>
  <author>
    <name>My name</name>
    <email>name@email.address</email>
  </author>
  <description>
    A description of the model
```

Προσομοίωση Ρομπότ μέσω Υπολογιστή

</description>

</model>

- <name> υποχρεωτικό

Το όνομα του μοντέλου.

- <version> υποχρεωτικό

Η έκδοση αυτού του μοντέλου.

Σημείωση: Αυτή δεν είναι η έκδοση του SDF που χρησιμοποιεί το μοντέλο. Αυτή η πληροφορία διατηρείται στο αρχείο model.sdf.

- <sdf> υποχρεωτικό

Το όνομα ενός αρχείου SDF ή URDF που περιγράφει αυτό το μοντέλο. Το χαρακτηριστικό **έκδοση** δείχνει ποια έκδοση SDF χρησιμοποιεί το αρχείο και δεν είναι απαραίτητο για URDFs. Μπορούν να χρησιμοποιηθούν πολλά στοιχεία <sdf> για να υποστηριχθούν πολλαπλές εκδόσεις SDF.

- <author> υποχρεωτικό

- <name> υποχρεωτικό

Το όνομα του συγγραφέα του μοντέλου.

- <email> υποχρεωτικό

Η διεύθυνση email του συγγραφέα.

- <περιγραφή> υποχρεωτικό

Η περιγραφή του μοντέλου πρέπει να περιλαμβάνει:

- Τι είναι το μοντέλο (π.χ., ρομπότ, τραπέζι, φλιτζάνι)

- Τι κάνουν τα plugins (λειτουργικότητα του μοντέλου)

- <depend> προαιρετικό

Όλες οι εξαρτήσεις για αυτό το μοντέλο. Συνήθως πρόκειται για άλλα μοντέλα.

- <μοντέλο> προαιρετικό

- <uri> υποχρεωτικό

Το URI της εξάρτησης του μοντέλου.

- <έκδοση> υποχρεωτικό

- Η έκδοση του μοντέλου.

2.13 Κατασκευή ενός Κινητού Ρομπότ

2.13.1 Ρύθμιση του φακέλου του μοντέλου μας

Θα δημιουργήσουμε το δικό μας μοντέλο, το οποίο πρέπει να ακολουθεί τους κανόνες μορφοποίησης για τη δομή φακέλων της Βάσης Δεδομένων Μοντέλων του Gazebo.

1. Δημιουργούμε έναν φάκελο μοντέλου:

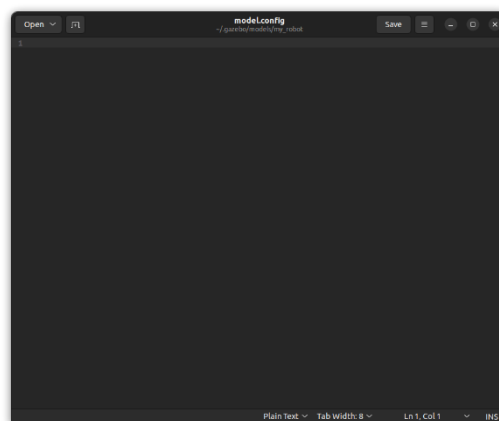
```
mkdir -p ~/.gazebo/models/my_robot
```

```
diplwmatiki@diplwmatiki: ~  
bash: /opt/ros/noetic/setup.bash: No such file or directory  
bash: /opt/ros/melodic/setup.bash: No such file or directory  
diplwmatiki@diplwmatiki: ~$ mkdir -p ~/.gazebo/models/my_robot  
diplwmatiki@diplwmatiki: ~$
```

2. Δημιουργούμε ένα αρχείο config για το μοντέλο:

`gedit ~/.gazebo/models/my_robot/model.config`

```
diplwmatiki@diplwmatiki: ~  
bash: /opt/ros/noetic/setup.bash: No such file or directory  
bash: /opt/ros/melodic/setup.bash: No such file or directory  
diplwmatiki@diplwmatiki: ~$ mkdir -p ~/.gazebo/models/my_robot  
diplwmatiki@diplwmatiki: ~$ gedit ~/.gazebo/models/my_robot/model.config  
diplwmatiki@diplwmatiki: ~$
```



3. Επικολλούμε τα ακόλουθα περιεχόμενα:

```
<?xml  
<model>  
  <name>My  
  <version>1.0</version>
```

```
version="1.0"?>
```

```
Robot</name>
```

Προσομοίωση Ρομπότ μέσω Υπολογιστή

<sdf

version='1.4'>model.sdf</sdf>

<author>

<name>My

Name</name>

<email>me@my.email</email>

</author>

<description>

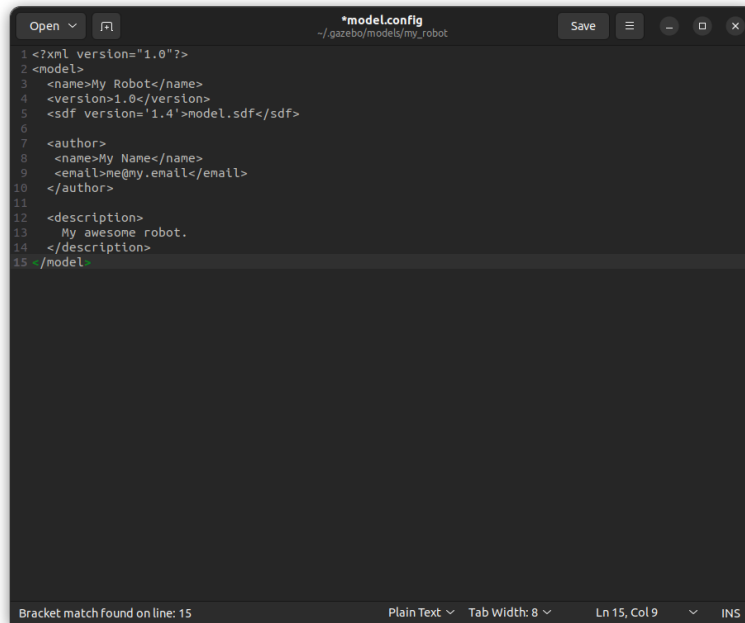
My

awesome

robot.

</description>

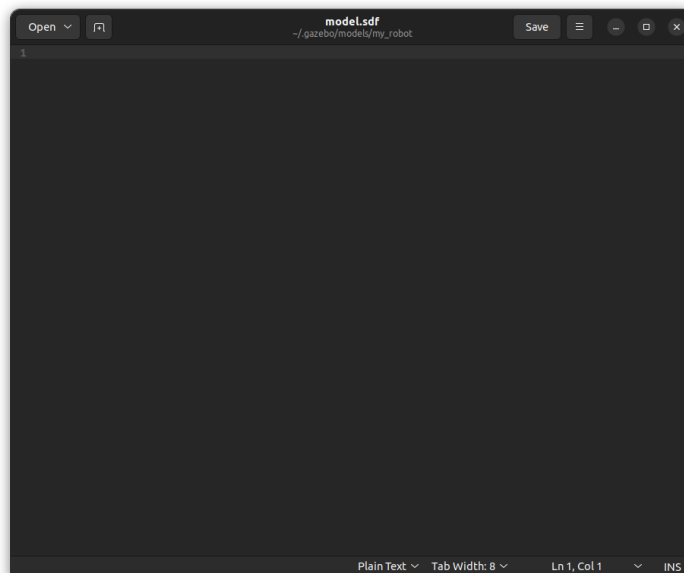
</model>



```
1 <?xml version="1.0"?>
2 <model>
3   <name>My Robot</name>
4   <version>1.0</version>
5   <sdf version="1.4">model.sdf</sdf>
6
7   <author>
8     <name>My Name</name>
9     <email>me@my.email</email>
10  </author>
11
12  <description>
13    My awesome robot.
14  </description>
15 </model>
```

4. Δημιουργούμε ένα αρχείο ~/.gazebo/models/my_robot/model.sdf:

gedit ~/.gazebo/models/my_robot/model.sdf

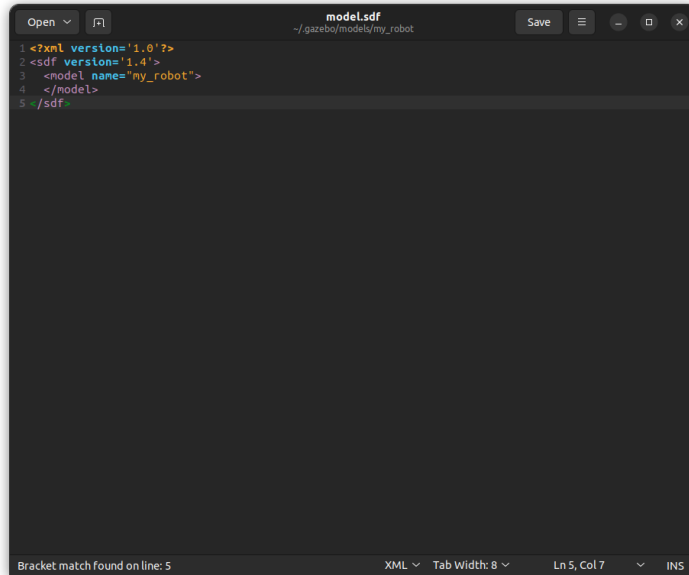


```
1
```

Προσομοίωση Ρομπότ μέσω Υπολογιστή

5. Επικολλούμε τα ακόλουθα:

```
<?xml version='1.0'?>
<sdf version='1.4'>
  <model name="my_robot">
    </model>
  </sdf>
```



Σε αυτό το σημείο έχουμε το βασικό περιεχόμενο για ένα μοντέλο. Το αρχείο **model.config** περιγράφει το ρομπότ με κάποια επιπλέον μεταδεδομένα. Το αρχείο **model.sdf** περιέχει τις απαραίτητες ετικέτες για να γίνει η παρουσίαση ενός μοντέλου με το όνομα **my_robot**, χρησιμοποιώντας το Gazebo συνδεδεμένο με την έκδοση 1.4 του SDF.

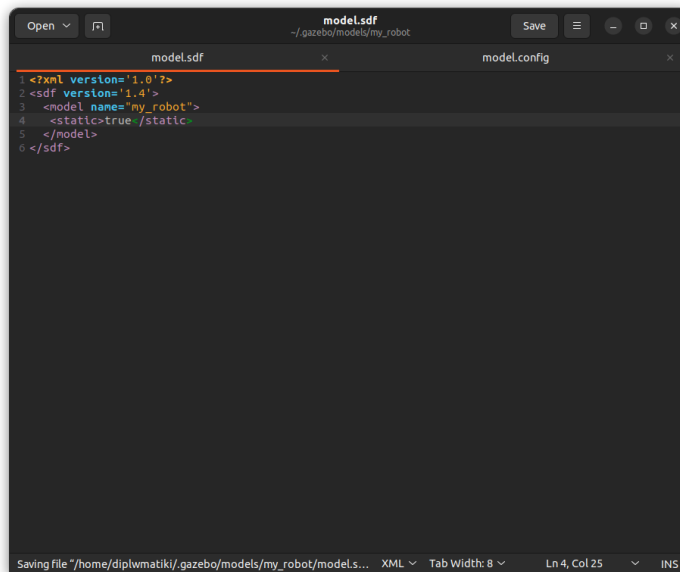
2.13.2 Κατασκευή της Δομής του Μοντέλου

Αυτό το βήμα θα δημιουργήσει μια ορθογώνια βάση με δύο τροχούς.

Είναι σημαντικό να ξεκινήσουμε απλά και να κατασκευάσουμε το μοντέλο βήμα προς βήμα. Το πρώτο βήμα είναι να διατάξουμε τα βασικά σχήματα του μοντέλου. Για να το κάνουμε αυτό, θα ορίσουμε το μοντέλο μας ως στατικό, που σημαίνει ότι θα αγνοηθεί από τη μηχανή φυσικής. Ως αποτέλεσμα, το μοντέλο θα παραμείνει σε ένα σημείο και θα μας επιτρέψει να ευθυγραμμίσουμε σωστά όλα τα εξαρτήματα.

1. Κάνουμε το μοντέλο στατικό προσθέτοντας το στοιχείο `<static>true</static>` στο αρχείο `~/gazebo/models/my_robot/model.sdf`:

```
<?xml version='1.0'?>
  <sdf version='1.4'>
    <model name="my_robot">
      <static>true</static>
    </model>
  </sdf>
```

2. Προσθέτουμε τη ορθογώνια βάση επεξεργάζοντας το αρχείο `~/gazebo/models/my_robot/model.sdf`:

```

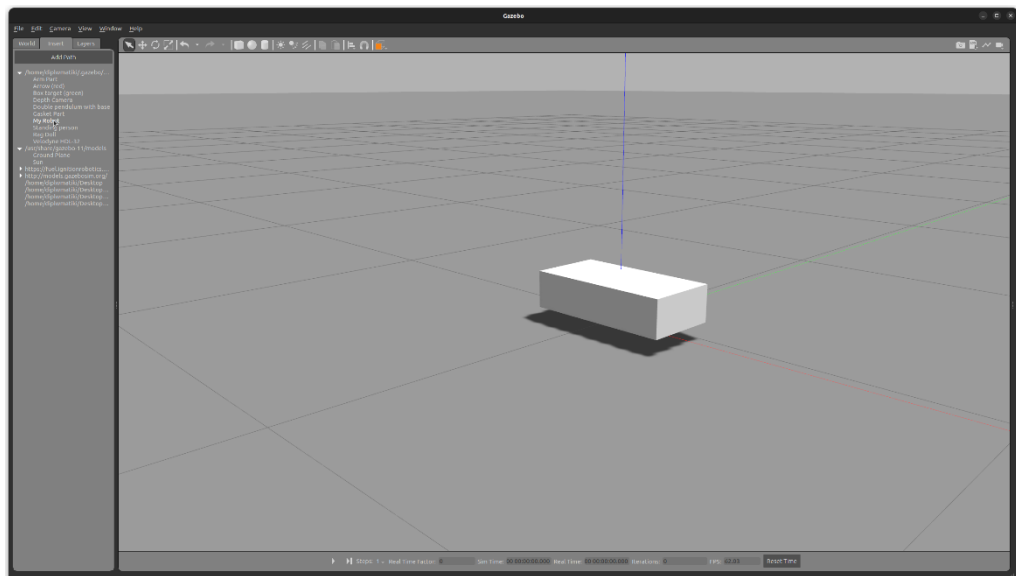
<?xml version='1.0'?>
  <sdf version='1.4'>
    <model name='my_robot'>
      <static>true</static>
      <link name='chassis'>
        <pose>0 0 .1 0 0 0</pose>
        <collision name='collision'>
          <geometry>
            <box>
              <size>.4 .2 .1</size>
            </box>
          </geometry>
        </collision>
        <visual name='visual'>
          <geometry>
            <box>
              <size>.4 .2 .1</size>
            </box>
          </geometry>
        </visual>
      </link>
    </model>
  </sdf>
    
```

```

1 <?xml version='1.0'?>
2 <sdf version='1.4'>
3   <model name='my_robot'>
4     <static>true</static>
5     <link name='chassis'>
6       <pose>0 0 .1 0 0 0</pose>
7
8       <collision name='collision'>
9         <geometry>
10          <box>
11            <size>.4 .2 .1</size>
12          </box>
13        </geometry>
14      </collision>
15
16      <visual name='visual'>
17        <geometry>
18          <box>
19            <size>.4 .2 .1</size>
20          </box>
21        </geometry>
22      </visual>
23    </link>
24  </model>
25 </sdf>
    
```

Εδώ έχουμε δημιουργήσει ένα κουτί με διαστάσεις 0.4 x 0.2 x 0.1 μέτρα. Το στοιχείο collision καθορίζει το σχήμα που χρησιμοποιείται από τη μηχανή ανίχνευσης συγκρούσεων. Το στοιχείο visual καθορίζει το σχήμα που χρησιμοποιείται από τη μηχανή απεικόνισης. Στις περισσότερες περιπτώσεις χρήσης, τα στοιχεία collision και visual είναι τα ίδια. Η πιο συνηθισμένη χρήση διαφορετικών στοιχείων collision και visual είναι η ύπαρξη ενός απλοποιημένου στοιχείου collision συνδυασμένου με ένα στοιχείο visual που χρησιμοποιεί ένα πολύπλοκο mesh. Αυτό θα βοηθήσει στη βελτίωση της απόδοσης.

3. Δοκιμάζουμε το μοντέλο μας εκτελώντας το Gazebo και εισάγοντας το μοντέλο μας μέσω της διεπαφής Insert Model στο GUI. Θα πρέπει να δούμε ένα λευκό κουτί να αιωρείται 0.1 μέτρα πάνω από το έδαφος.



4. Τώρα μπορούμε να προσθέσουμε έναν περιστρεφόμενο τροχό στο ρομπότ. Ο περιστρεφόμενος τροχός είναι μια σφαίρα χωρίς τριβή. Αυτός ο τύπος τροχίσκου είναι καλύτερος από την προσθήκη τροχού με άρθρωση, καθώς επιβάλλει λιγότερους περιορισμούς στη μηχανή φυσικής.

```
<collision name='caster_collision'>
  <pose>-0.15 0 -0.05 0 0 0</pose>
  <geometry>
    <sphere>
      <radius>.05</radius>
    </sphere>
  </geometry>

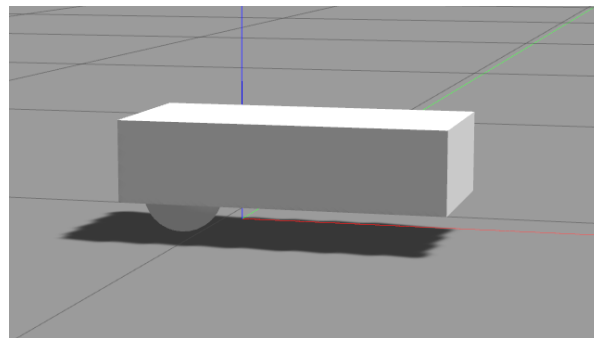
  <surface>
    <friction>
      <ode>
        <mu>0</mu>
        <mu2>0</mu2>
        <slip1>1.0</slip1>
        <slip2>1.0</slip2>
      </ode>
    </friction>
  </surface>
</collision>

<visual name='caster_visual'>
  <pose>-0.15 0 -0.05 0 0 0</pose>
  <geometry>
    <sphere>
      <radius>.05</radius>
    </sphere>
  </geometry>
</visual>
```

```

model.sdf
~/gazebo/models/my_robot
Save
model.sdf
model.config
19     <size>.4 .2 .1</size>
20   </box>
21 </geometry>
22 </visual>
23   <collision name='caster_collision'>
24     <pose>-0.15 0 -0.05 0 0 0</pose>
25     <geometry>
26       <sphere>
27         <radius>.05</radius>
28       </sphere>
29     </geometry>
30
31     <surface>
32       <friction>
33         <ode>
34           <mu>0</mu>
35           <mu2>0</mu2>
36           <slip1>1.0</slip1>
37           <slip2>1.0</slip2>
38         </ode>
39       </friction>
40     </surface>
41   </collision>
42
43   <visual name='caster_visual'>
44     <pose>-0.15 0 -0.05 0 0 0</pose>
45     <geometry>
46       <sphere>
47         <radius>.05</radius>
48       </sphere>
49     </geometry>
50   </visual>
51 </link>
52 </model>
53 </sdf>
XML Tab Width: 8 Ln 50, Col 20 INS

```



5. Τώρα ας προσθέσουμε έναν αριστερό τροχό. Τροποποιούμε το αρχείο `~/gazebo/models/my_robot/model.sdf` ώστε να είναι το εξής:

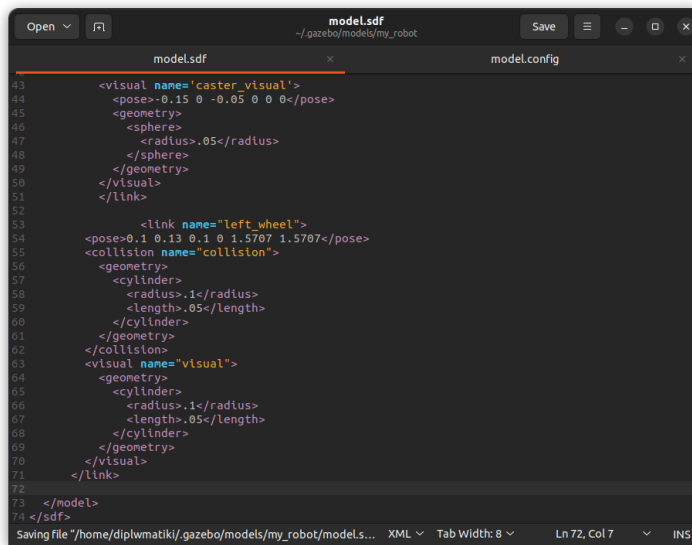
```

<link name="left_wheel">
  <pose>0.1 0.13 0.1 0 1.5707 1.5707</pose>
  <collision name="collision">
    <geometry>
      <cylinder>
        <radius>.1</radius>
        <length>.05</length>
      </cylinder>
    </geometry>
  </collision>
  <visual name="visual">
    <geometry>
      <cylinder>
        <radius>.1</radius>
        <length>.05</length>
      </cylinder>
    </geometry>
  </visual>
</link>

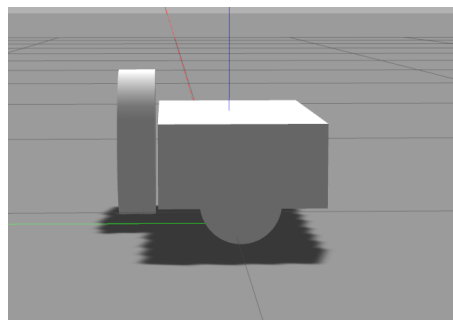
```

Προσομοίωση Ρομπότ μέσω Υπολογιστή

```
</geometry>  
</visual>  
</link>
```



```
model.sdf  
model.config  
43 <visual name='caster_visual'>  
44 <pose>-0.15 0 -0.05 0 0 0</pose>  
45 <geometry>  
46 <sphere>  
47 <radius>.05</radius>  
48 </sphere>  
49 </geometry>  
50 </visual>  
51 </link>  
52  
53 <link name="left_wheel">  
54 <pose>0.1 0.13 0.1 0 1.5707 1.5707</pose>  
55 <collision name="collision">  
56 <geometry>  
57 <cylinder>  
58 <radius>.1</radius>  
59 <length>.05</length>  
60 </cylinder>  
61 </geometry>  
62 </collision>  
63 <visual name="visual">  
64 <geometry>  
65 <cylinder>  
66 <radius>.1</radius>  
67 <length>.05</length>  
68 </cylinder>  
69 </geometry>  
70 </visual>  
71 </link>  
72 </model>  
73 </sdf>  
74 </sdf>  
Saving file "/home/diplwmatiki/gazebo/models/my_robot/model.sdf" XML Tab Width: 8 Ln 72, Col 7 INS
```

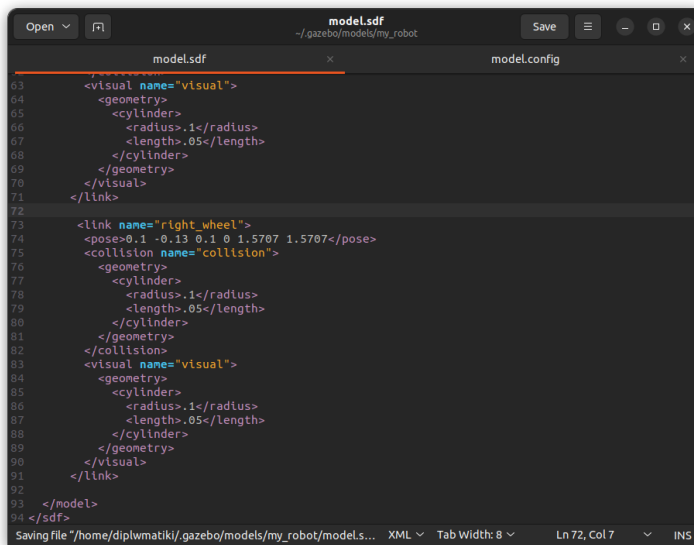


- Μπορούμε να φτιάξουμε έναν δεξιό τροχό αντιγράφοντας τον αριστερό τροχό και προσαρμόζοντας τη θέση του συνδέσμου του τροχού (wheel link):

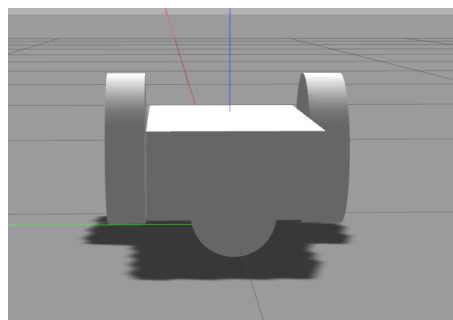
```
<link name="right_wheel">  
<pose>0.1 -0.13 0.1 0 1.5707 1.5707</pose>  
<collision name="collision">  
<geometry>  
<cylinder>  
<radius>.1</radius>  
<length>.05</length>  
</cylinder>  
</geometry>  
</collision>  
<visual name="visual">  
<geometry>  
<cylinder>  
<radius>.1</radius>  
<length>.05</length>  
</cylinder>
```

Προσομοίωση Ρομπότ μέσω Υπολογιστή

```
</geometry>  
</visual>  
</link>
```



```
model.sdf  
model.config  
63 <visual name="visual">  
64 <geometry>  
65 <cylinder  
66 <radius>.1</radius>  
67 <length>.05</length>  
68 </cylinder>  
69 </geometry>  
70 </visual>  
71 </link>  
72  
73 <link name="right_wheel">  
74 <pose>0 1 -0.13 0 1 0 1.5707 1.5707</pose>  
75 <collision name="collision">  
76 <geometry>  
77 <cylinder>  
78 <radius>.1</radius>  
79 <length>.05</length>  
80 </cylinder>  
81 </geometry>  
82 </collision>  
83 <visual name="visual">  
84 <geometry>  
85 <cylinder>  
86 <radius>.1</radius>  
87 <length>.05</length>  
88 </cylinder>  
89 </geometry>  
90 </visual>  
91 </link>  
92 </model>  
93 </sdf>  
94 </sdf>
```



7. Κάνουμε το μοντέλο δυναμικό ορίζοντας το <static> σε false και προσθέτουμε δύο αρθρώσεις περιστροφής (hinge joints) για τον αριστερό και τον δεξιό τροχό:

```
<static>false</static>
```

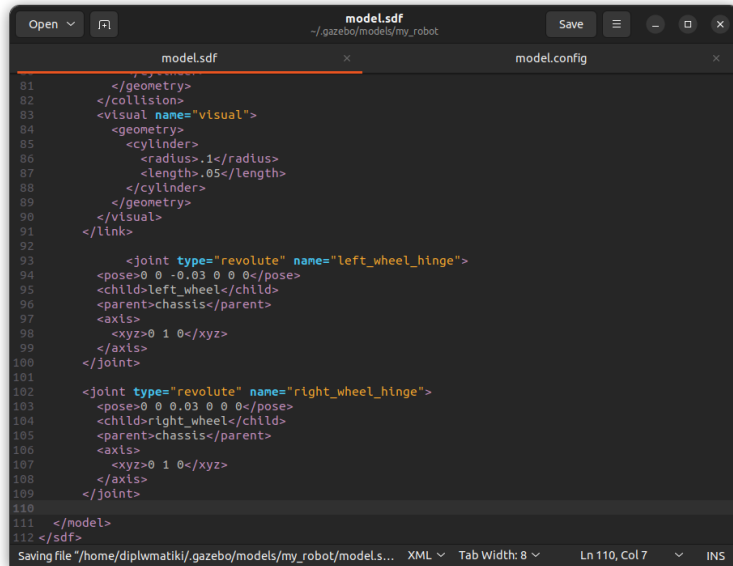
.....

```
<joint type="revolute" name="left_wheel_hinge">  
  <pose>0 0 -0.03 0 0 0</pose>  
  <child>left_wheel</child>  
  <parent>chassis</parent>  
  <axis>  
    <xyz>0 1 0</xyz>  
  </axis>  
</joint>
```

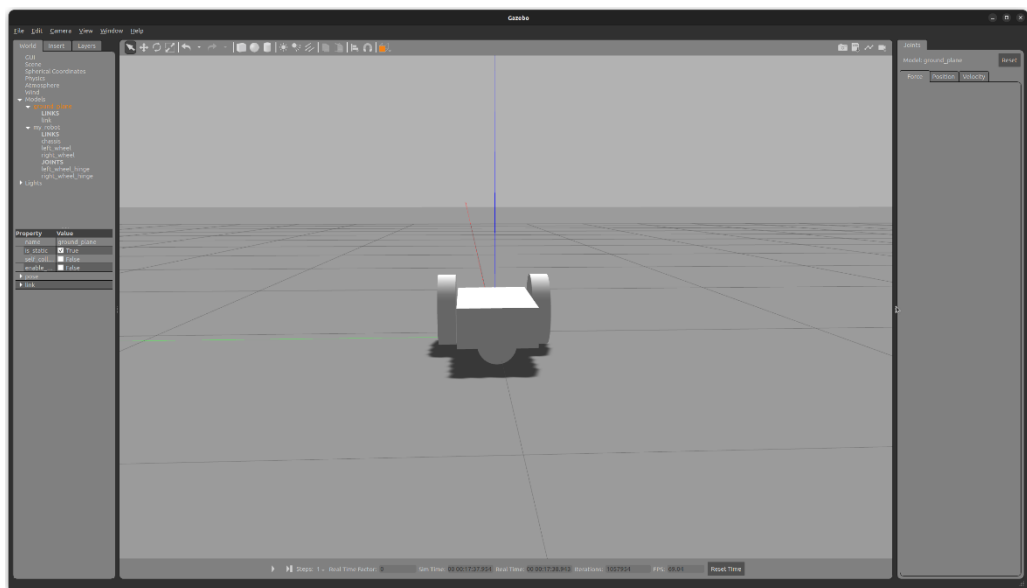
```
<joint type="revolute" name="right_wheel_hinge">  
  <pose>0 0 0.03 0 0 0</pose>  
  <child>right_wheel</child>  
  <parent>chassis</parent>
```

Προσομοίωση Ρομπότ μέσω Υπολογιστή

```
<axis>  
<xyz>0 1 0</xyz>  
</axis>  
</joint>
```

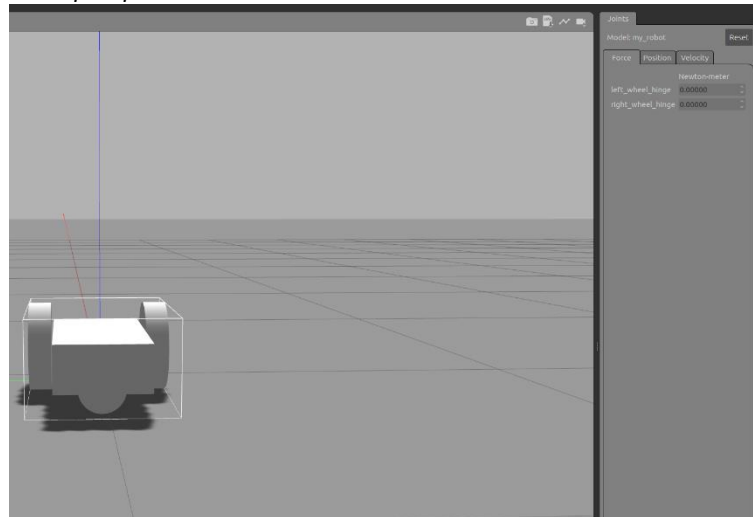


```
model.sdf  
model.config  
81 </geometry>  
82 </collision>  
83 <visual name="visual">  
84 <geometry>  
85 <cylinder>  
86 <radius>.1</radius>  
87 <length>.05</length>  
88 </cylinder>  
89 </geometry>  
90 </visual>  
91 </link>  
92  
93 <joint type="revolute" name="left_wheel_hinge">  
94 <pose>0 0 -0.03 0 0 0</pose>  
95 <child>left_wheel</child>  
96 <parent>chassis</parent>  
97 <axis>  
98 <xyz>0 1 0</xyz>  
99 </axis>  
100 </joint>  
101  
102 <joint type="revolute" name="right_wheel_hinge">  
103 <pose>0 0 0.03 0 0 0</pose>  
104 <child>right_wheel</child>  
105 <parent>chassis</parent>  
106 <axis>  
107 <xyz>0 1 0</xyz>  
108 </axis>  
109 </joint>  
110  
111 </model>  
112 </sdf>  
Saving file "/home/diplmatiki/gazebo/models/my_robot/model.sdf" XML Tab Width: 8 Ln 110, Col 7 INS
```

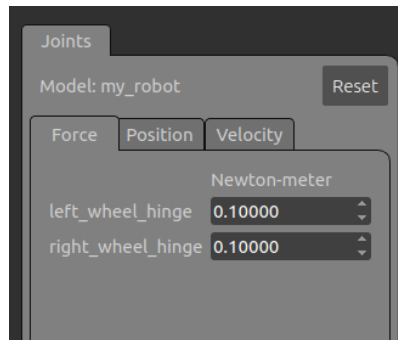


Οι δύο αρθρώσεις περιστρέφονται γύρω από τον άξονα y $\langle xyz \rangle 0 1 0 \langle /xyz \rangle$ και συνδέουν κάθε τροχό με το πλαίσιο (chassis).

8. Κάνουμε κλικ στις τελείες στα δεξιά της οθόνης και τις σύρουμε προς τα αριστερά.



9. Θα πρέπει να εμφανιστεί ένα νέο παράθυρο που περιέχει διάφορους ελεγκτές για κάθε άρθρωση. (Σημείωση: Βεβαιωνόμαστε ότι το μοντέλο που θέλουμε να ελέγξουμε είναι επιλεγμένο).



10. Στην καρτέλα Force, αυξάνουμε τη δύναμη που εφαρμόζεται σε κάθε άρθρωση σε περίπου 0.1N-m. Το ρομπότ θα πρέπει να αρχίσει να κινείται.

2.14 Robot Operating System (ROS)

Το ROS (Robot Operating System) είναι ένα ανοιχτού κώδικα πλαίσιο που μας παρέχει εργαλεία, βιβλιοθήκες και πρότυπα για τη διευκόλυνση της ανάπτυξης σύνθετων ρομποτικών συστημάτων. Παρά το όνομά του, το ROS δεν είναι αυτόνομο λειτουργικό σύστημα, αλλά λειτουργεί ως middleware, δηλαδή ως ενδιάμεσο επίπεδο ανάμεσα στο λειτουργικό σύστημα (όπως το Linux ή τα Windows) και τις εφαρμογές ρομποτικής, επιτρέποντας την επικοινωνία και την ενσωμάτωση των διαφορετικών μερών που συνθέτουν το λογισμικό ενός ρομπότ.

Βασικά Χαρακτηριστικά του ROS και του ROS 2:

Αρχιτεκτονική με Κόμβους:
Το ROS και το ROS 2 βασίζονται στην αρχιτεκτονική των κόμβων. Σε αυτήν την αρχιτεκτονική, κάθε κόμβος αντιπροσωπεύει μια συγκεκριμένη λειτουργία, όπως ο έλεγχος ενός κινητήρα ή η επεξεργασία εικόνων από κάμερα. Οι κόμβοι είναι αρθρωτοί, επιτρέποντας την επαναχρησιμοποίηση και την καλύτερη διαχείριση του κώδικα. Οι κόμβοι μπορούν να κατανέμονται σε διαφορετικές μηχανές στο δίκτυο, διευκολύνοντας έτσι την κλιμάκωση του συστήματος.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Μηχανισμοί Επικοινωνίας: Οι κόμβοι επικοινωνούν μεταξύ τους χρησιμοποιώντας μηνύματα, και τόσο το ROS όσο και το ROS 2 υποστηρίζουν:

Θέματα (για μηνύματα δημοσίευσης/εγγραφής): Οι κόμβοι μπορούν να δημοσιεύουν δεδομένα ή να εγγράφονται σε θέματα για να λαμβάνουν δεδομένα, όπως δεδομένα αισθητήρων.

Υπηρεσίες (για αιτήματα/απαντήσεις): Οι υπηρεσίες επιτρέπουν την άμεση ανταλλαγή δεδομένων μέσω αιτημάτων και απαντήσεων.

Ενέργειες (για μακροχρόνιες εργασίες): Στις περιπτώσεις που έχουμε εργασίες μεγάλης διάρκειας, όπως η κίνηση ενός ρομποτικού βραχίονα, οι ενέργειες μας παρέχουν συνεχή ανατροφοδότηση.

Αφαίρεση Υλικού: Το ROS και το ROS 2 υποστηρίζουν την αφαίρεση του υλικού, επιτρέποντάς μας να αλληλοεπιδρούμε με διάφορους αισθητήρες και ενεργοποιητές μέσω μιας κοινής διεπαφής, χωρίς να χρειάζεται να γράφουμε εξατομικευμένα προγράμματα οδήγησης.

Σύστημα Πακέτων: Το ROS και το ROS 2 οργανώνουν τον κώδικα σε πακέτα. Αυτά τα πακέτα περιέχουν βιβλιοθήκες, εργαλεία και άλλα απαραίτητα αρχεία για συγκεκριμένες λειτουργίες. Το οικοσύστημα πακέτων είναι πλούσιο, και μπορούμε να βρούμε πολλά έτοιμα πακέτα για να καλύψουμε προβλήματα όπως η πλοήγηση, η αντίληψη και η χειραγώγηση.

Υποστήριξη Προσομοίωσης: Τόσο το ROS όσο και το ROS 2 συνδέονται με προσομοιωτές όπως το Gazebo, δίνοντάς μας τη δυνατότητα να προσομοιώσουμε ρομποτικά περιβάλλοντα και αλληλεπιδράσεις χωρίς την ανάγκη φυσικού υλικού. Αυτό είναι πολύ χρήσιμο για τη δοκιμή αλγορίθμων σε εικονικό περιβάλλον.

Υποστήριξη για Κατανεμημένα Συστήματα: Το ROS και το ROS 2 παρέχουν την υποδομή για την ανάπτυξη κατανεμημένων συστημάτων, επιτρέποντάς μας να τρέχουμε κόμβους σε διαφορετικές μηχανές. Το ROS 2, με τη χρήση του DDS (Data Distribution Service), προσφέρει βελτιωμένη επικοινωνία για κατανεμημένα συστήματα, καθώς και δυνατότητες πραγματικού χρόνου.

Ποιότητα Υπηρεσιών (QoS) στο ROS 2: Το ROS 2 μας δίνει τον έλεγχο πάνω στην ποιότητα της επικοινωνίας μέσω των ρυθμίσεων QoS (Quality of Service), επιτρέποντάς μας να προσαρμόζουμε την αξιοπιστία, την καθυστέρηση και άλλες παραμέτρους ανάλογα με τις ανάγκες της εφαρμογής μας.

Ασφάλεια στο ROS 2: Σε αντίθεση με το ROS 1, το ROS 2 υποστηρίζει χαρακτηριστικά ασφαλείας όπως κρυπτογράφηση και αυθεντικοποίηση, γεγονός που το καθιστά πιο κατάλληλο για κρίσιμες εφαρμογές, όπως αυτόνομα οχήματα ή βιομηχανικά ρομπότ.

Εφαρμογές του ROS και του ROS 2: Αυτόνομα Οχήματα: Το ROS χρησιμοποιείται ευρέως στην αυτόνομη οδήγηση για τη διαχείριση αισθητήρων, τον εντοπισμό αντικειμένων και τον σχεδιασμό διαδρομών. Το ROS 2, με τις δυνατότητες πραγματικού χρόνου, είναι ακόμη πιο κατάλληλο για τέτοιες εφαρμογές.

Drones: Τόσο το ROS όσο και το ROS 2 χρησιμοποιούνται για τον έλεγχο της πτήσης και την αποφυγή εμποδίων στα drones.

Βιομηχανικά Ρομπότ: Στη βιομηχανία, το ROS και το ROS 2 είναι χρήσιμα για τον έλεγχο ρομποτικών βραχιόνων που εκτελούν εργασίες όπως η συγκόλληση ή η βαφή.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Έρευνα και Ακαδημαϊκά Ιδρύματα: Το ROS είναι διαδεδομένο στην ακαδημαϊκή έρευνα και στην ανάπτυξη νέων ρομποτικών αλγορίθμων. Το ROS 2 προσφέρει επιπλέον πλεονεκτήματα λόγω της συμβατότητάς του με πολλές πλατφόρμες και της ασφάλειάς του.

Ρομπότ Υπηρεσιών: Στην υγειονομική περίθαλψη και σε άλλες υπηρεσίες, ρομπότ χρησιμοποιούν το ROS για να παρέχουν φροντίδα ή να εκτελούν εργασίες μεταφοράς. Το ROS 2, με την υποστήριξη πραγματικού χρόνου, είναι ιδανικό για ρομπότ που χρειάζονται γρήγορες αντιδράσεις.

Πλεονεκτήματα του ROS 2 έναντι του ROS 1: Πραγματικός Χρόνος: Το ROS 2 έχει σχεδιαστεί για να υποστηρίζει εφαρμογές σε πραγματικό χρόνο, κάτι που έλειπε από το ROS 1.

Ασφάλεια: Το ROS 2 προσφέρει ενσωματωμένη ασφάλεια, επιτρέποντας ασφαλή επικοινωνία μεταξύ των κόμβων.

Υποστήριξη Πολλαπλών Πλατφορμών: Το ROS 2 μπορεί να εκτελείται σε Linux, Windows, macOS, αλλά και σε λειτουργικά συστήματα πραγματικού χρόνου.

Κλιμάκωση και Αξιοπιστία: Το ROS 2, χάρη στο DDS, μπορεί να υποστηρίξει μεγάλης κλίμακας και κατανεμημένα συστήματα με καλύτερη αξιοπιστία.

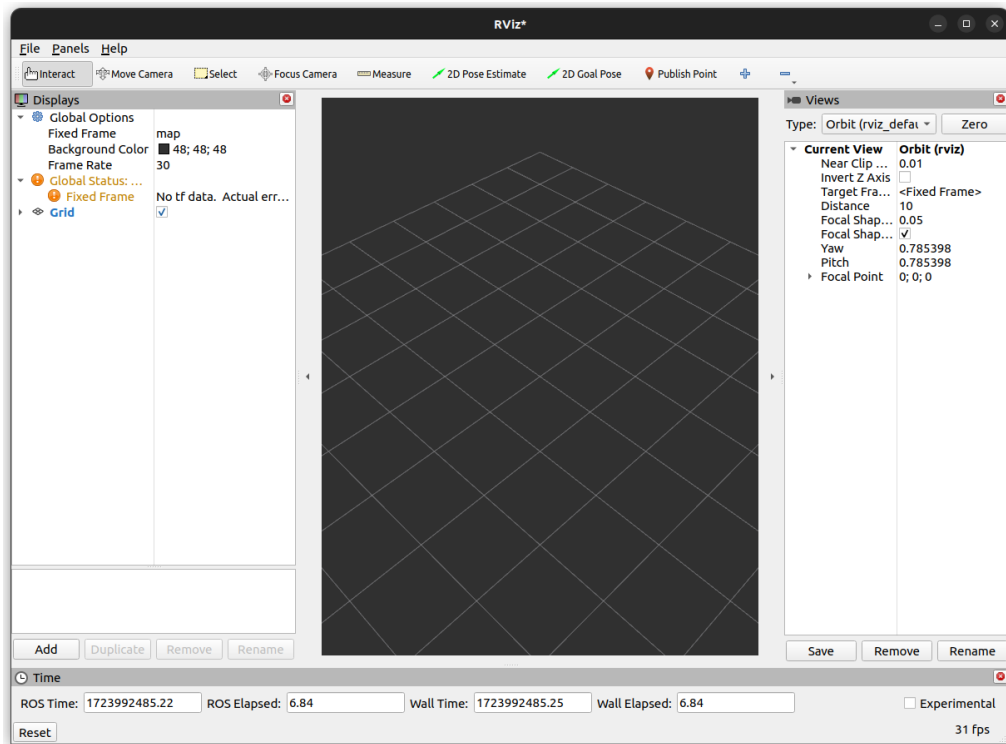
Ιστορία του ROS: Το ROS δημιουργήθηκε το 2007 από το ερευνητικό εργαστήριο Willow Garage, και έχει εξελιχθεί σε ένα από τα πιο διαδεδομένα εργαλεία για την ανάπτυξη ρομποτικών συστημάτων. Σήμερα, συντηρείται από την Open Robotics, με τη βοήθεια μιας παγκόσμιας κοινότητας προγραμματιστών. Το ROS 2 αναπτύχθηκε για να αντιμετωπίσει ορισμένους περιορισμούς του ROS 1, προσφέροντας βελτιωμένες δυνατότητες, όπως η υποστήριξη πραγματικού χρόνου και η ασφάλεια.

Συμπέρασμα:

Το ROS και το ROS 2 αποτελούν ισχυρά εργαλεία για την ανάπτυξη ρομποτικών συστημάτων. Μας προσφέρουν μια πλούσια υποδομή για την επικοινωνία και την ενσωμάτωση διαφόρων ρομποτικών εξαρτημάτων, κάνοντας την ανάπτυξη ρομποτικών εφαρμογών πιο γρήγορη και αποδοτική. Με το ROS 2, έχουμε ακόμα περισσότερα πλεονεκτήματα, όπως καλύτερη απόδοση σε πραγματικό χρόνο, ασφάλεια και δυνατότητες για μεγάλης κλίμακας έργα.

Το Σύστημα Λειτουργίας Ρομπότ (ROS) είναι ένα σύνολο βιβλιοθηκών λογισμικού και εργαλείων για την ανάπτυξη εφαρμογών ρομποτικής. Από τους οδηγούς και τους προηγμένους αλγόριθμους έως τα ισχυρά εργαλεία για προγραμματιστές, το ROS παρέχει τα ανοιχτού κώδικα εργαλεία που χρειαζόμαστε για το επόμενο έργο ρομποτικής μας.

2.15 RViz



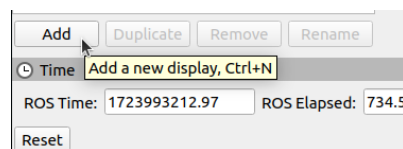
Το μεγάλο παράθυρο στο κέντρο είναι η 3D προβολή. Στα αριστερά είναι η λίστα Displays, η οποία θα εμφανίσει όποια προβολή έχουμε φορτώσει. Αυτή τη στιγμή περιέχει μόνο τις global options και ένα Grid, με τα οποία θα ασχοληθούμε αργότερα. Στα δεξιά βρίσκονται μερικά από τα άλλα πάνελ, τα οποία περιγράφονται παρακάτω.

2.15.1 Προβολές (Displays)

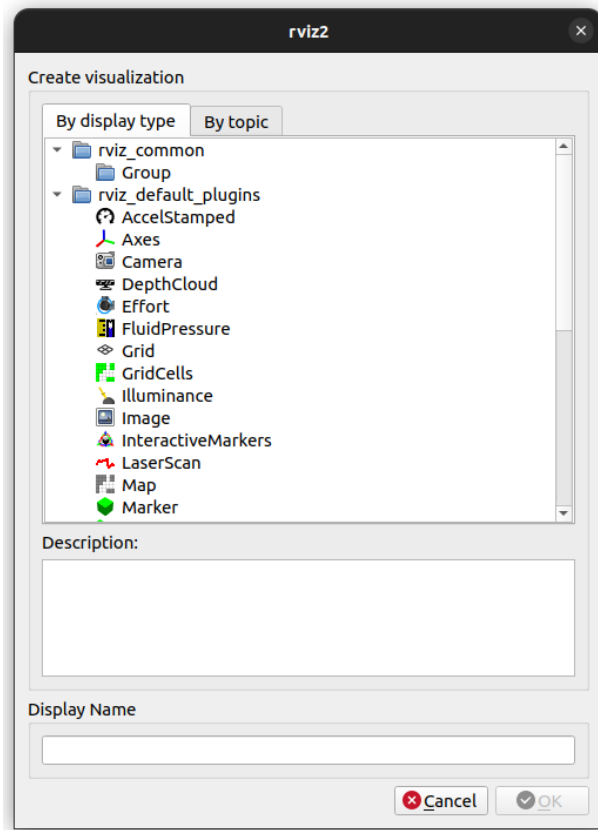
Μια προβολή είναι κάτι που σχεδιάζει κάτι στον 3D κόσμο και πιθανότατα έχει διαθέσιμες επιλογές στη λίστα displays. Ένα παράδειγμα είναι ένα point cloud, η κατάσταση του ρομπότ κ.λπ.

2.15.2 Προσθήκη νέας προβολής

Για να προσθέσουμε μια νέα προβολή, κάνουμε κλικ στο κουμπί Add στο κάτω μέρος:



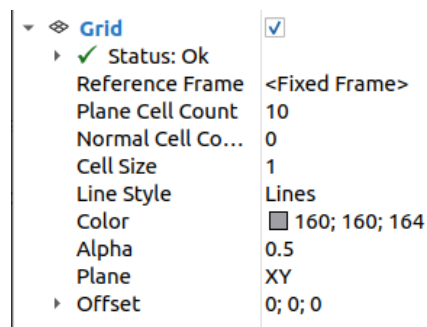
Αυτό θα εμφανίσει το παράθυρο διαλόγου για την προσθήκη νέας προβολής:



Η λίστα στη κορυφή περιέχει τον τύπο της προβολής. Ο τύπος καθορίζει τι είδους δεδομένα θα οπτικοποιήσει αυτή η προβολή. Το πλαίσιο κειμένου στο μέσο δίνει μια περιγραφή του επιλεγμένου τύπου προβολής. Τέλος, πρέπει να δώσουμε στην προβολή ένα μοναδικό όνομα. Αν έχουμε, για παράδειγμα, δύο σαρωτές λέιζερ στο ρομπότ μας, μπορούμε να δημιουργήσουμε δύο προβολές Laser Scan με ονόματα όπως "Laser Base" και "Laser Head".

2.15.3 Ιδιότητες Προβολής

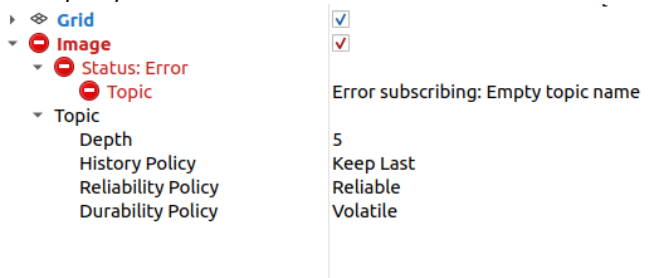
Κάθε προβολή έχει τη δική της λίστα ιδιοτήτων. Για παράδειγμα:



2.15.4 Κατάσταση Προβολής

Κάθε προβολή έχει τη δική της κατάσταση για να μας βοηθήσει να καταλάβουμε αν όλα είναι εντάξει ή όχι. Η κατάσταση μπορεί να είναι μία από τις εξής: OK, Warning, Error, ή Disabled. Η κατάσταση υποδεικνύεται στον τίτλο της προβολής από το χρώμα του φόντου, καθώς και στην κατηγορία Status που μπορούμε να δούμε όταν η προβολή είναι εκτεταμένη.

Προσομοίωση Ρομπότ μέσω Υπολογιστή



Η κατηγορία Status επεκτείνεται επίσης για να εμφανίσει συγκεκριμένες πληροφορίες κατάστασης. Αυτές οι πληροφορίες διαφέρουν για κάθε προβολή και τα μηνύματα θα πρέπει να είναι κατανοητά από μόνα τους.

Όνομα	Περιγραφή	Μηνύματα που Χρησιμοποιούνται
Axes	Εμφανίζουμε ένα σύνολο αξόνων	-
Effort	Δείχνουμε την προσπάθεια που καταβάλλεται σε κάθε περιστροφική άρθρωση ενός ρομπότ	sensor_msgs/msg/JointStates
Camera	Δημιουργούμε ένα νέο παράθυρο απεικόνισης από την προοπτική μιας κάμερας και υπερθέτουμε την εικόνα	sensor_msgs/msg/Image, sensor_msgs/msg/CameraInfo
Grid	Εμφανίζουμε ένα 2D ή 3D πλέγμα κατά μήκος ενός επιπέδου	-
Grid Cells	Σχεδιάζουμε κελιά από ένα πλέγμα, συνήθως εμπόδια από έναν costmap της στοίβας πλοήγησης	nav_msgs/msg/GridCells
Image	Δημιουργούμε ένα νέο παράθυρο απεικόνισης με μια εικόνα. Δεν χρησιμοποιούμε CameraInfo	sensor_msgs/msg/Image
InteractiveMarker	Εμφανίζουμε 3D αντικείμενα από έναν ή περισσότερους διακομιστές Interactive Marker και αλληλοεπιδρούμε με αυτά	visualization_msgs/msg/InteractiveMarker
Laser Scan	Δείχνουμε δεδομένα από μια σάρωση λέιζερ, με διάφορες	sensor_msgs/msg/LaserScan

Προσομοίωση Ρομπότ μέσω Υπολογιστή

	επιλογές για τρόπους απεικόνισης, συσσώρευση, κ.λπ.	
Map	Εμφανίζουμε έναν χάρτη στο επίπεδο του εδάφους	nav_msgs/msg/OccupancyGrid
Markers	Επιτρέπουμε στους προγραμματιστές να εμφανίζουν αυθαίρετα πρωτόγονα σχήματα μέσω ενός θέματος	visualization_msgs/msg/Marker, visualization_msgs/msg/MarkerArray
Path	Δείχνουμε μια διαδρομή από τη στοίβα πλοήγησης	nav_msgs/msg/Path
Point	Σχεδιάζουμε ένα σημείο ως μικρή σφαίρα	geometry_msgs/msg/PointStamped
Pose	Σχεδιάζουμε μια θέση ως βέλος ή άξονες	geometry_msgs/msg/PoseStamped
Pose Array	Σχεδιάζουμε ένα «σύννεφο» βελών, ένα για κάθε θέση σε έναν πίνακα θέσεων	geometry_msgs/msg/PoseArray
Point Cloud(2)	Εμφανίζουμε δεδομένα από ένα point cloud, με διάφορες επιλογές για τρόπους απεικόνισης, συσσώρευση, κ.λπ.	sensor_msgs/msg/PointCloud, sensor_msgs/msg/PointCloud2
Polygon	Σχεδιάζουμε το περίγραμμα ενός πολύγωνου ως γραμμές	geometry_msgs/msg/Polygon
Odometry	Συσσωρεύουμε τις θέσεις οδομέτρησης με την πάροδο του χρόνου	nav_msgs/msg/Odometry
Range	Εμφανίζουμε κόνους που αντιπροσωπεύουν μετρήσεις εμβέλειας από αισθητήρες υπερήχων ή IR	sensor_msgs/msg/Range
RobotModel	Εμφανίζουμε μια οπτική αναπαράσταση ενός ρομπότ στη σωστή θέση (όπως	-

	ορίζεται από τους τρέχοντες μετασχηματισμούς TF)	
TF	Εμφανίζουμε την ιεραρχία μετασχηματισμών tf2	-
Wrench	Σχεδιάζουμε ένα εργαλείο ως βέλος (δύναμη) και βέλος + κύκλο (ροπή)	geometry_msgs/msg/WrenchStamped
Twist	Σχεδιάζουμε μια περιστροφή ως βέλος (γραμμική) και βέλος + κύκλο (γωνιακή)	geometry_msgs/msg/TwistStamped

2.15.5 Διαμορφώσεις

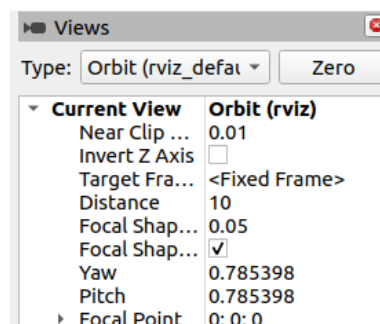
Διαφορετικές διαμορφώσεις προβολών είναι συχνά χρήσιμες για διαφορετικές χρήσεις του οπτικοποιητή. Μια διαμόρφωση χρήσιμη για ένα πλήρες PR2 δεν είναι απαραίτητα χρήσιμη για ένα καρότσι δοκιμής. Προς αυτή την κατεύθυνση, ο οπτικοποιητής μας επιτρέπει να φορτώνουμε και να αποθηκεύουμε διαφορετικές διαμορφώσεις.

Μια διαμόρφωση περιέχει:

- Προβολές και τις ιδιότητές τους
- Ιδιότητες εργαλείων
- Το σημείο προβολής και τις ρυθμίσεις για την 3D απεικόνιση

2.15.6 Πίνακας Προβολών (Views Panel)

Υπάρχουν διάφοροι τύποι καμερών διαθέσιμοι στον οπτικοποιητή.



Οι τύποι καμερών περιλαμβάνουν τόσο διαφορετικούς τρόπους ελέγχου της κάμερας όσο και διαφορετικούς τύπους προβολής (Ορθογραφική ή Προοπτική).

2.15.7 Κάμερα σε τροχιά (προεπιλογή)

Η κάμερα σε τροχιά περιστρέφεται απλά γύρω από ένα σημείο εστίασης, ενώ κοιτάζει πάντα προς αυτό το σημείο. Το σημείο εστίασης απεικονίζεται ως ένας μικρός δίσκος όταν μετακινούμε την κάμερα.

Χειρισμοί:

- **Αριστερό κουμπί του ποντικιού:** Κάνουμε κλικ και σύρουμε για να περιστρέψουμε γύρω από το σημείο εστίασης.
- **Μεσαίο κουμπί του ποντικιού:** Κάνουμε κλικ και σύρουμε για να μετακινήσουμε το σημείο εστίασης στο επίπεδο που σχηματίζεται από τους άξονες πάνω και δεξιά της κάμερας. Η απόσταση μετακίνησης εξαρτάται από το σημείο εστίασης – αν υπάρχει αντικείμενο στο σημείο εστίασης και κάνουμε κλικ πάνω του, θα παραμείνει κάτω από το ποντίκι μας.
- **Δεξί κουμπί του ποντικιού:** Κάνουμε κλικ και σύρουμε για να κάνουμε ζουμ προς/από το σημείο εστίασης. Σύρσιμο προς τα πάνω κάνει ζουμ προς τα μέσα, ενώ προς τα κάτω κάνει ζουμ προς τα έξω.
- **Τροχός κύλισης:** Ζουμάrouμε προς/από το σημείο εστίασης.

2.15.8 Κάμερα FPS (πρώτου προσώπου)

Η κάμερα FPS είναι μια κάμερα πρώτου προσώπου, οπότε περιστρέφεται σαν να κοιτάμε με το κεφάλι μας.

Χειρισμοί:

- **Αριστερό κουμπί του ποντικιού:** Κάνουμε κλικ και σύρουμε για περιστροφή. Κρατάμε Control και κάνουμε κλικ για να επιλέξουμε το αντικείμενο κάτω από το ποντίκι και να το κοιτάξουμε απευθείας.
- **Μεσαίο κουμπί του ποντικιού:** Κάνουμε κλικ και σύρουμε για να μετακινηθούμε κατά μήκος του επιπέδου που σχηματίζεται από τους άξονες πάνω και δεξιά της κάμερας.
- **Δεξί κουμπί του ποντικιού:** Κάνουμε κλικ και σύρουμε για να μετακινηθούμε κατά μήκος του εμπρόσθιου άξονα της κάμερας. Σύρσιμο προς τα πάνω κινείται προς τα εμπρός, ενώ προς τα κάτω κινείται προς τα πίσω.
- **Τροχός κύλισης:** Μετακίνηση προς τα εμπρός/πίσω.

2.15.9 Ορθογραφική κάμερα από πάνω προς τα κάτω

Η ορθογραφική κάμερα από πάνω προς τα κάτω κοιτάζει πάντα κατά μήκος του άξονα Z (στο πλαίσιο του ρομπότ) και είναι μια ορθογραφική προβολή, που σημαίνει ότι τα αντικείμενα δεν μικραίνουν όσο απομακρύνονται.

Χειρισμοί:

- **Αριστερό κουμπί του ποντικιού:** Κάνουμε κλικ και σύρουμε για περιστροφή γύρω από τον άξονα Z.
- **Μεσαίο κουμπί του ποντικιού:** Κάνουμε κλικ και σύρουμε για να μετακινήσουμε την κάμερα κατά μήκος του επιπέδου XY.
- **Δεξί κουμπί του ποντικιού:** Κάνουμε κλικ και σύρουμε για να κάνουμε ζουμ στην εικόνα.
- **Τροχός κύλισης:** Ζουμάrouμε στην εικόνα.

Κάμερα σε τροχιά XY

Όμοια με την κάμερα σε τροχιά, αλλά το σημείο εστίασης περιορίζεται στο επίπεδο XY.

Χειρισμοί:

Δείτε τους χειρισμούς της κάμερας σε τροχιά.

Κάμερα τρίτου προσώπου που ακολουθεί

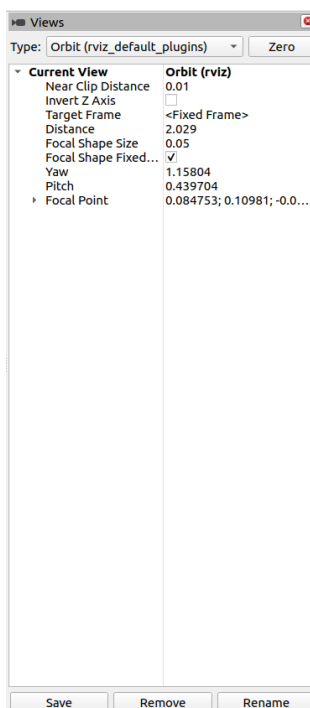
Η κάμερα διατηρεί μια σταθερή γωνία θέασης προς το πλαίσιο-στόχο. Σε αντίθεση με την κάμερα σε τροχιά XY, η κάμερα στρίβει όταν το πλαίσιο-στόχος περιστρέφεται. Αυτό μπορεί να είναι χρήσιμο αν κάνουμε χαρτογράφηση 3D σε έναν διάδρομο με γωνίες, για παράδειγμα.

Χειρισμοί:

Ίδιοι με τους χειρισμούς της κάμερας σε τροχιά.

2.15.10 Προσαρμοσμένες Προβολές

Το πάνελ προβολών μας επιτρέπει επίσης να δημιουργούμε διαφορετικές ονοματισμένες προβολές, οι οποίες αποθηκεύονται και μπορούμε να εναλλάσσουμε μεταξύ τους. Μια προβολή αποτελείται από ένα πλαίσιο-στόχο, τύπο κάμερας και θέση κάμερας. Μπορούμε να αποθηκεύσουμε μια προβολή κάνοντας κλικ στο κουμπί Save στο πάνελ προβολών.



Μια προβολή αποτελείται από:

- Τύπο ελεγκτή προβολής
- Ρύθμιση προβολής (θέση, προσανατολισμός, κ.λπ., πιθανώς διαφορετική για κάθε τύπο ελεγκτή προβολής)

Προσομοίωση Ρομπότ μέσω Υπολογιστή

- Το πλαίσιο-στόχο

Οι προβολές αποθηκεύονται ανά χρήστη, όχι στα αρχεία ρυθμίσεων.

2.15.11 Συστήματα Συντεταγμένων

Το RViz χρησιμοποιεί το σύστημα μετασχηματισμών tf για να μετασχηματίσει δεδομένα από το πλαίσιο συντεταγμένων στο οποίο φτάνουν σε ένα παγκόσμιο πλαίσιο αναφοράς. Υπάρχουν δύο πλαίσια συντεταγμένων που είναι σημαντικά στον οπτικοποιητή: το πλαίσιο-στόχος και το σταθερό πλαίσιο.

2.15.12 Το Σταθερό Πλαίσιο

Το πιο σημαντικό από τα δύο πλαίσια είναι το σταθερό πλαίσιο. Το σταθερό πλαίσιο είναι το πλαίσιο αναφοράς που χρησιμοποιείται για να προσδιορίσει το παγκόσμιο πλαίσιο. Συνήθως αυτό είναι ο χάρτης, ο κόσμος, ή κάτι παρόμοιο, αλλά μπορεί επίσης να είναι, για παράδειγμα, το πλαίσιο οδομέτρησης (odometry frame).

Αν το σταθερό πλαίσιο έχει οριστεί λανθασμένα, για παράδειγμα στη βάση του ρομπότ, τότε όλα τα αντικείμενα που έχει δει το ρομπότ θα εμφανίζονται μπροστά του, στη θέση που ανιχνεύθηκαν σε σχέση με το ρομπότ. Για σωστά αποτελέσματα, το σταθερό πλαίσιο δεν πρέπει να κινείται σε σχέση με τον κόσμο.

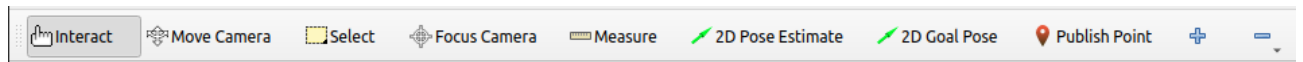
Αν αλλάξουμε το σταθερό πλαίσιο, όλα τα δεδομένα που προβάλλονται αυτήν τη στιγμή διαγράφονται αντί να μετασχηματιστούν εκ νέου.

2.15.13 Το Πλαίσιο-Στόχος

Το πλαίσιο-στόχος είναι το πλαίσιο αναφοράς για την προβολή της κάμερας. Για παράδειγμα, αν το πλαίσιο-στόχος είναι ο χάρτης, θα βλέπουμε το ρομπότ να κινείται γύρω από τον χάρτη. Αν το πλαίσιο-στόχος είναι η βάση του ρομπότ, το ρομπότ θα παραμένει στο ίδιο σημείο ενώ όλα τα άλλα θα κινούνται σε σχέση με αυτό.

2.15.14 Εργαλεία

Ο οπτικοποιητής διαθέτει μια σειρά από εργαλεία που μπορούμε να χρησιμοποιήσουμε στη γραμμή εργαλείων. Οι παρακάτω ενότητες θα δώσουν μια σύντομη εισαγωγή σε αυτά τα εργαλεία. Μπορούμε να βρούμε περισσότερες πληροφορίες στο Help -> Show Help panel.



2.15.15 Αλληλεπίδραση (Interact)

Αυτό το εργαλείο μας επιτρέπει να αλληλοεπιδράσουμε με το οπτικοποιημένο περιβάλλον. Μπορούμε να κάνουμε κλικ σε αντικείμενα και, ανάλογα με τις ιδιότητές τους, να τα επιλέξουμε, να τα μετακινήσουμε και πολλά άλλα.

Συντόμευση πληκτρολογίου: i

2.15.16 Μετακίνηση Κάμερας (Move Camera)

Το εργαλείο Μετακίνησης Κάμερας είναι το προεπιλεγμένο εργαλείο. Όταν είναι επιλεγμένο και κάνουμε κλικ μέσα στην προβολή 3D, η γωνία θέασης αλλάζει σύμφωνα με τις επιλογές και τον τύπο κάμερας που έχουμε επιλέξει στο πάνελ Προβολών. Δείτε την προηγούμενη ενότητα Πάνελ Προβολών για περισσότερες πληροφορίες.

Συντόμευση πληκτρολογίου: m

2.15.17 Επιλογή (Select)

Το εργαλείο Επιλογής μας επιτρέπει να επιλέξουμε αντικείμενα που εμφανίζονται στην προβολή 3D. Υποστηρίζει επιλογή με ένα σημείο, καθώς και επιλογή περιοχής με κλικ και σύρσιμο κουτιού. Μπορούμε να προσθέσουμε σε μια επιλογή με το πλήκτρο Shift και να αφαιρέσουμε από την επιλογή με το πλήκτρο Ctrl. Αν θέλουμε να μετακινήσουμε την κάμερα ενώ επιλέγουμε, χωρίς να αλλάξουμε πίσω στο εργαλείο Μετακίνησης Κάμερας, μπορούμε να κρατήσουμε πατημένο το πλήκτρο Alt. Το πλήκτρο f θα εστιάσει την κάμερα στην τρέχουσα επιλογή.

Συντόμευση πληκτρολογίου: c

2.15.18 Εστίαση Κάμερας (Focus Camera)

Η εστίαση κάμερας μας επιτρέπει να επιλέξουμε μια τοποθεσία στον οπτικοποιητή. Η κάμερα θα εστιάσει σε αυτό το σημείο αλλάζοντας τον προσανατολισμό της, χωρίς να αλλάξει τη θέση της.

Συντόμευση πληκτρολογίου: c

2.15.19 Μέτρηση (Measure)

Με το εργαλείο μέτρησης μπορούμε να μετρήσουμε την απόσταση μεταξύ δύο σημείων στον οπτικοποιητή. Το πρώτο κλικ μετά την ενεργοποίηση του εργαλείου ορίζει το σημείο εκκίνησης και το δεύτερο το σημείο τερματισμού της μέτρησης. Η απόσταση που προκύπτει θα εμφανιστεί στο κάτω μέρος του παραθύρου του RViz. Σημειώνουμε ότι το εργαλείο μέτρησης λειτουργεί μόνο με αντικείμενα που έχουν αποδοθεί στον οπτικοποιητή και δεν μπορεί να χρησιμοποιηθεί σε κενό χώρο.

2.15.20 Χρόνος (Time)

Το πάνελ Χρόνου είναι κυρίως χρήσιμο όταν τρέχουμε σε προσομοιωτή, καθώς μας επιτρέπει να δούμε πόσος ROS Χρόνος έχει περάσει, σε σύγκριση με τον πραγματικό χρόνο (Wall Clock). Το πάνελ χρόνου μας επιτρέπει επίσης να επαναφέρουμε την εσωτερική χρονική κατάσταση του οπτικοποιητή, γεγονός που επαναφέρει όλες τις προβολές καθώς και την εσωτερική προσωρινή μνήμη δεδομένων του tf.

2.16 Δημιουργία ρομποτικού βραχίονα στο Gazebo

Για να χρησιμοποιήσουμε το βραχίονα UR10 της Universal Robotics θα χρησιμοποιήσουμε το παρακάτω git.

```
https://github.com/ros-industrial/universal_robot
```

Εγκατάσταση

Χρησιμοποιώντας apt

```
sudo apt-get install ros-noetic-universal-robots
```

```
mkdir -p ~/catkin_ws/src
```

```
catkin_make
```

```
source devel/setup.bash
```

```
cd ~/catkin_ws/src
```

```
git clone -b noetic-devel https://github.com/ros-industrial/universal_robot.git
```

```
cd ~/catkin_ws
```

```
rosdep
```

update

```
rosdep install --rosdistro noetic --ignore-src --from-paths src
```

```
catkin_make
```

```
# activate this workspace source /catkin_ws/devel/setup.bash
```

Για τη λαβή θα χρησιμοποιήσουμε το παρακάτω git.

```
https://github.com/LearnRoboticsWROS/robotiq_description/tree/main
```

Για να το εγκαταστήσουμε θα κάνουμε τα παρακάτω βήματα.

```
cd ~/catkin_ws/src
```

```
git clone https://github.com/LearnRoboticsWROS/robotiq_description.git
```

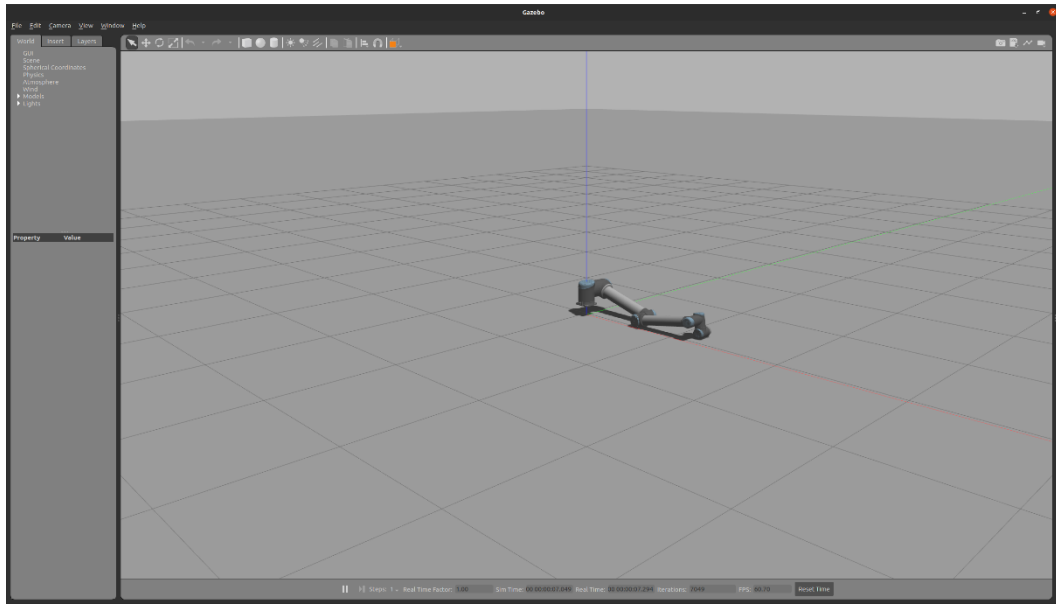
```
catkin_make
```

Αν τρέξουμε τώρα την εντολή:

```
roslaunch ur_gazebo ur10_bringup.launch
```

Προσομοίωση Ρομπότ μέσω Υπολογιστή

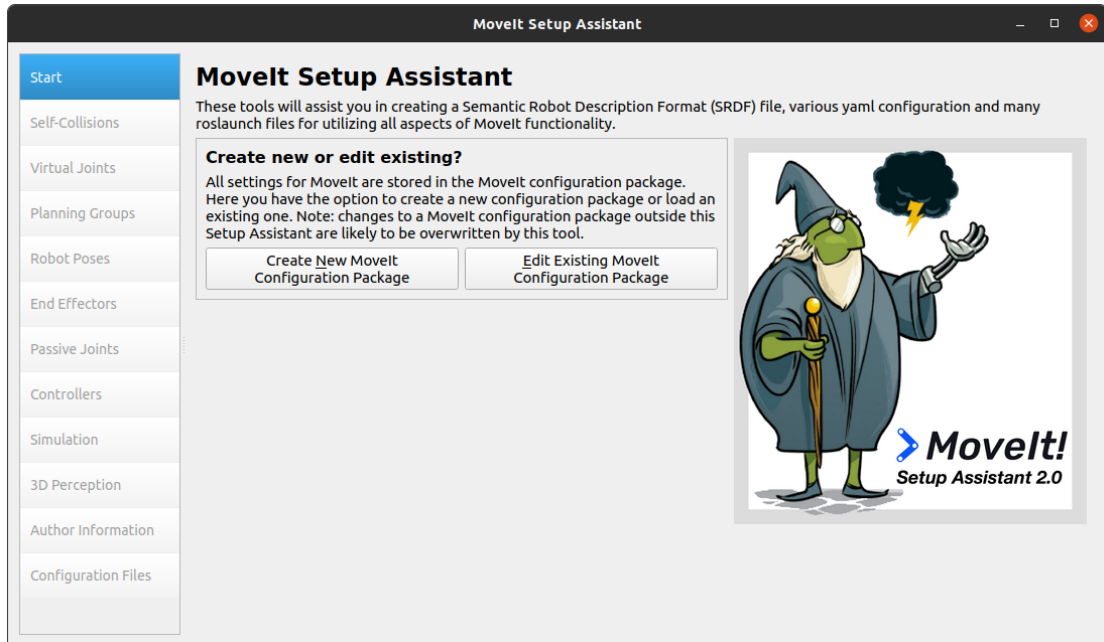
θα δούμε να ανοίγει το περιβάλλον του Gazebo και στη σκηνή θα υπάρχει ένας ρομποτικός βραχίονας UR10.



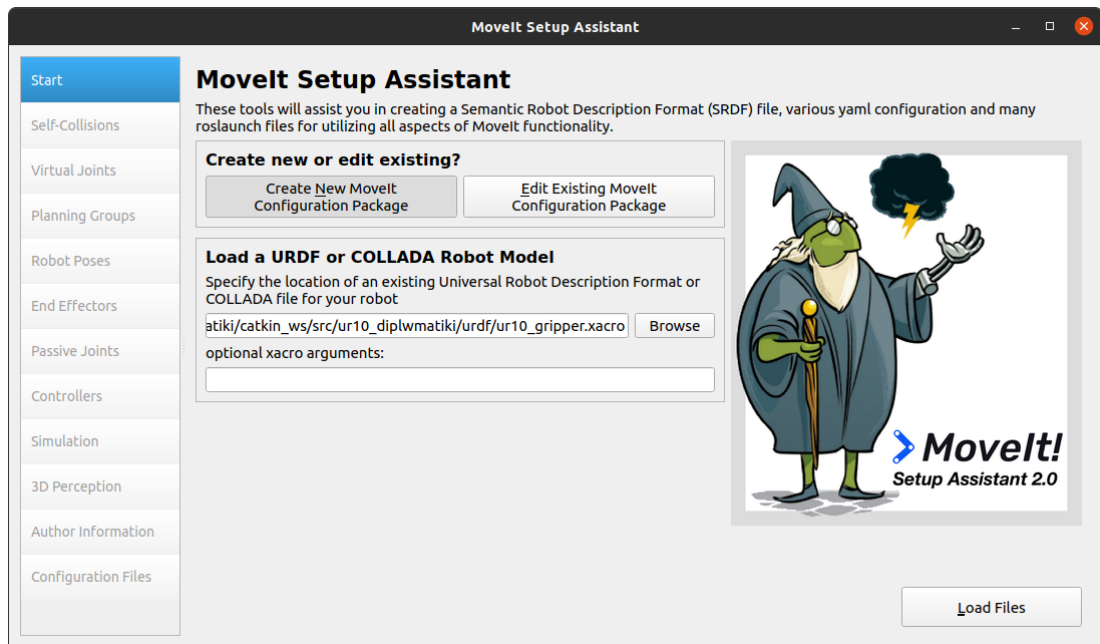
Εκκινούμε το Moveit

```
roslaunch moveit_setup_assistant setup_assistant.launch
```

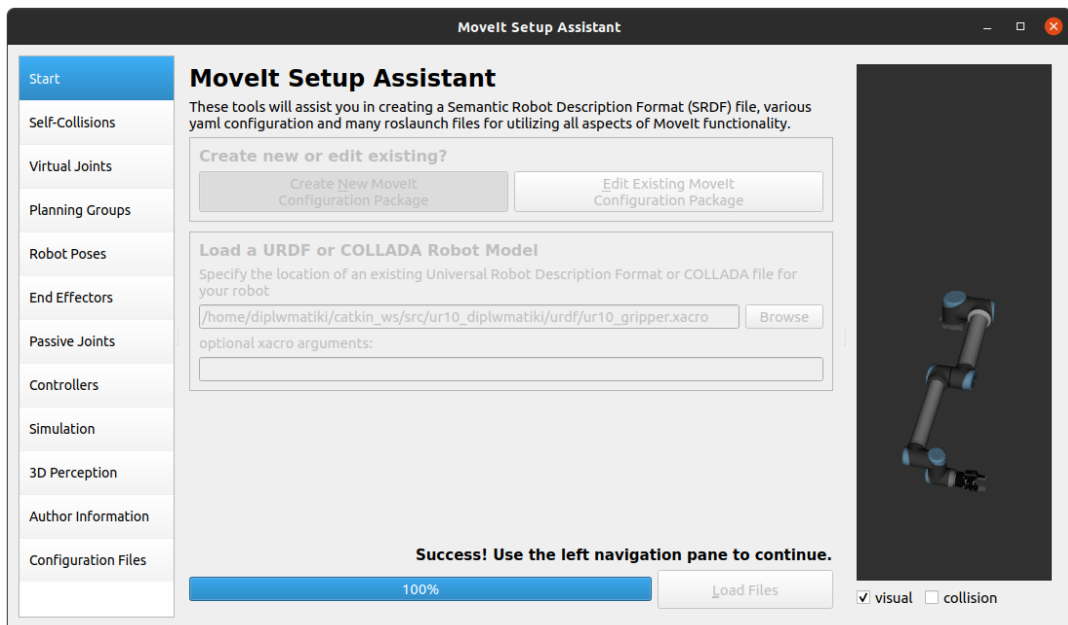
Το παράθυρο που μας εμφανίζεται είναι:



Πατάμε Create New MoveIt Configuration Package και “φορτώνουμε” το URDF στο οποίο έχουμε προσθέσει τη λαβή μας και πατάμε Load Files.

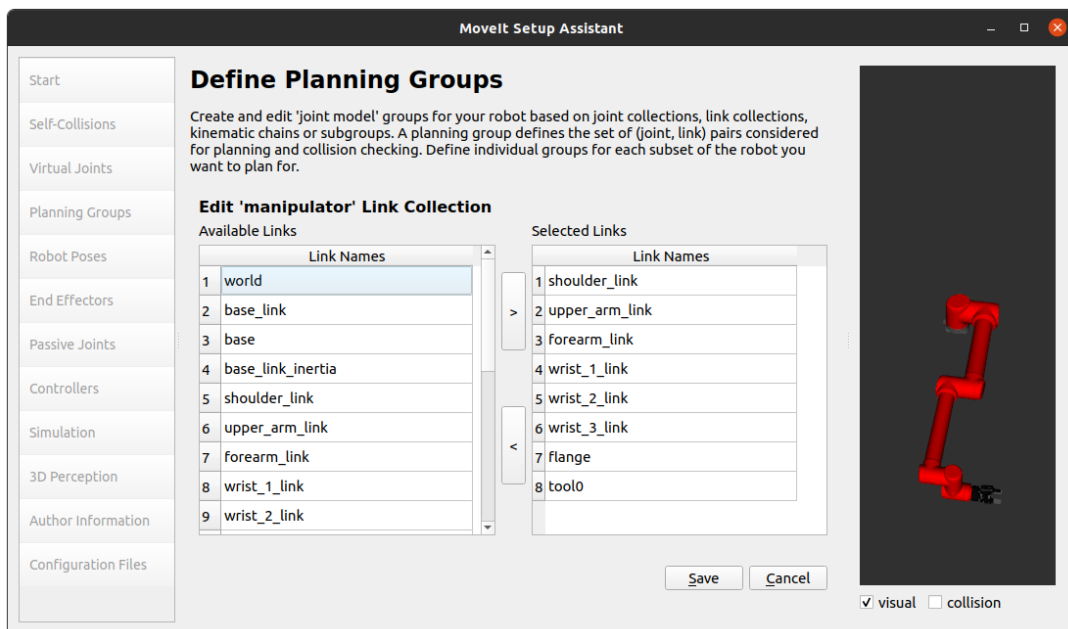
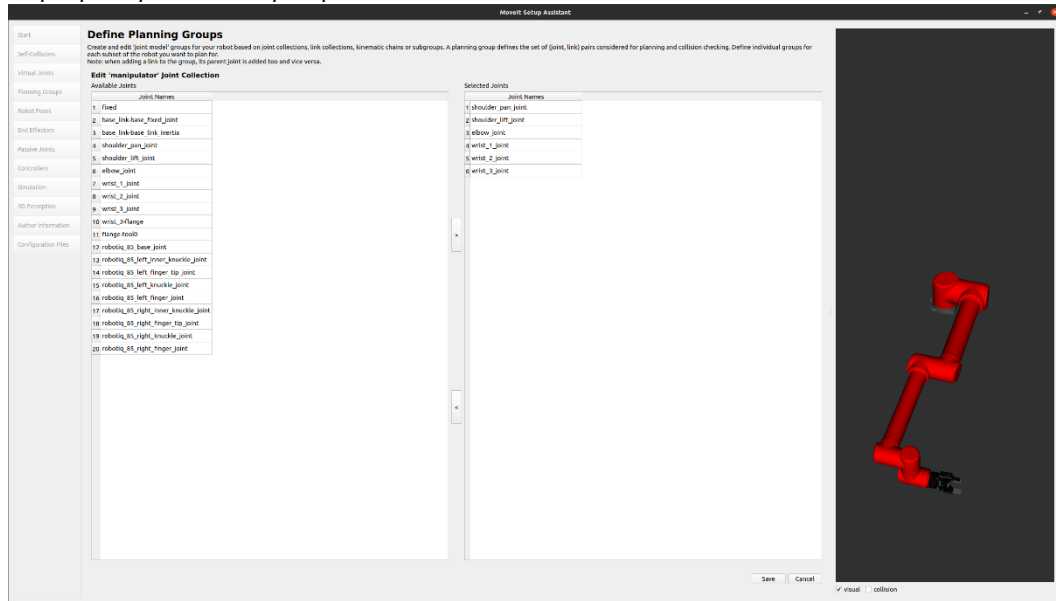


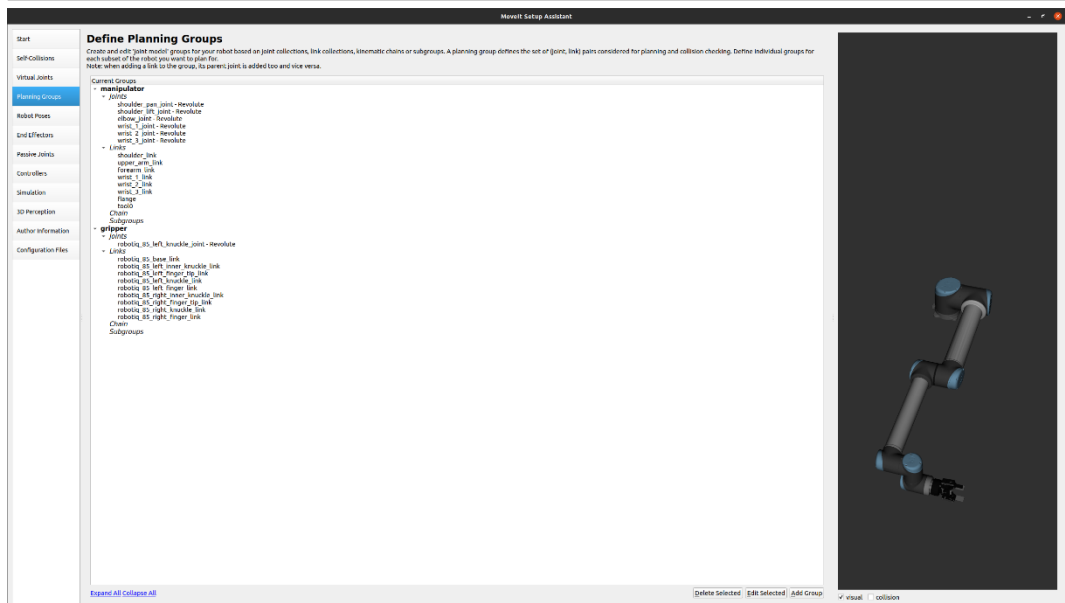
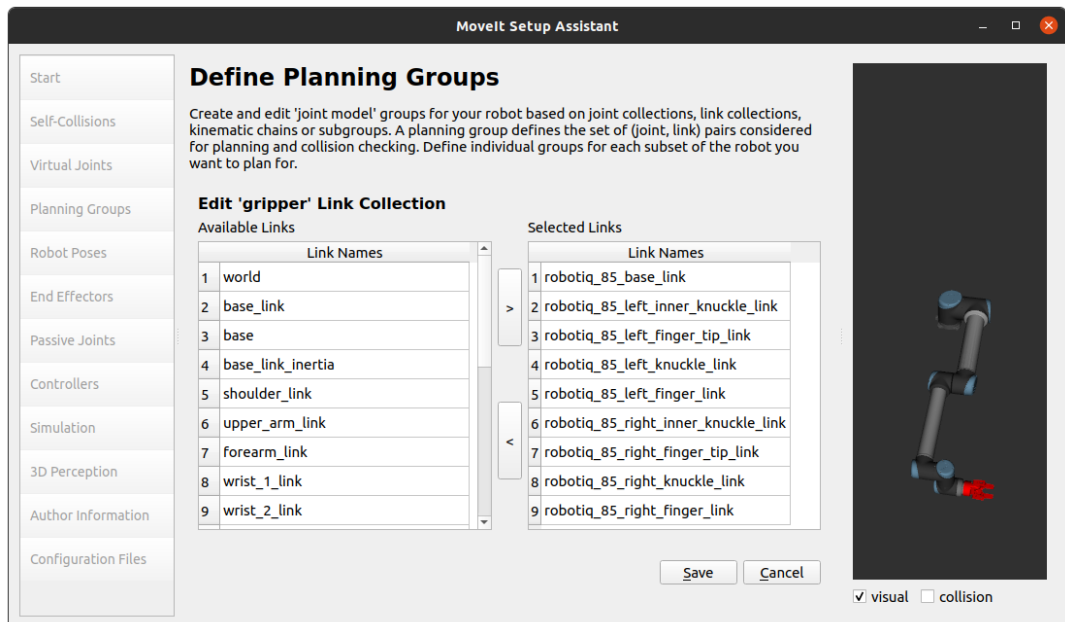
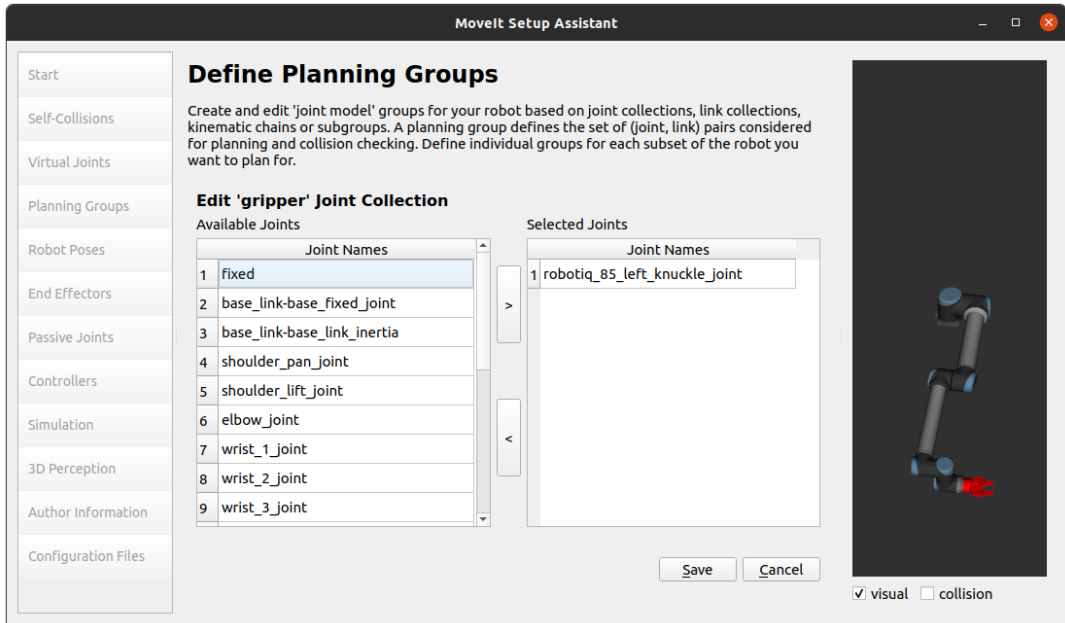
Αν έχουμε κάνει σωστά τη διαδικασία θα δούμε να εμφανίζεται ο βραχίονας μαζί με τη λαβή.



Στη καρτέλα Self-Collisions πατάμε Generate Collision Matrix. Έπειτα στη καρτέλα Planning Groups, πατάμε Add Group. Στο Group Name γράφουμε Manipulator, στο Kinematic Solver επιλέγουμε το kdl_kinematics_plugin/KDLKinematicsPlugin και στο OMPL Planning επιλέγουμε το RRT και πατάμε Add Joints. Κάνουμε την ίδια διαδικασία ξανά μόνο που αυτή τη φορά θα ονομάσουμε το group Gripper. Αφού τα κάνουμε αυτά κάνουμε διπλό κλικ στο Joints και στο Links και των δυο Groups (manipulator-gripper) και τα φτιάχνουμε όπως παρακάτω:

Προσομοίωση Ρομπότ μέσω Υπολογιστή

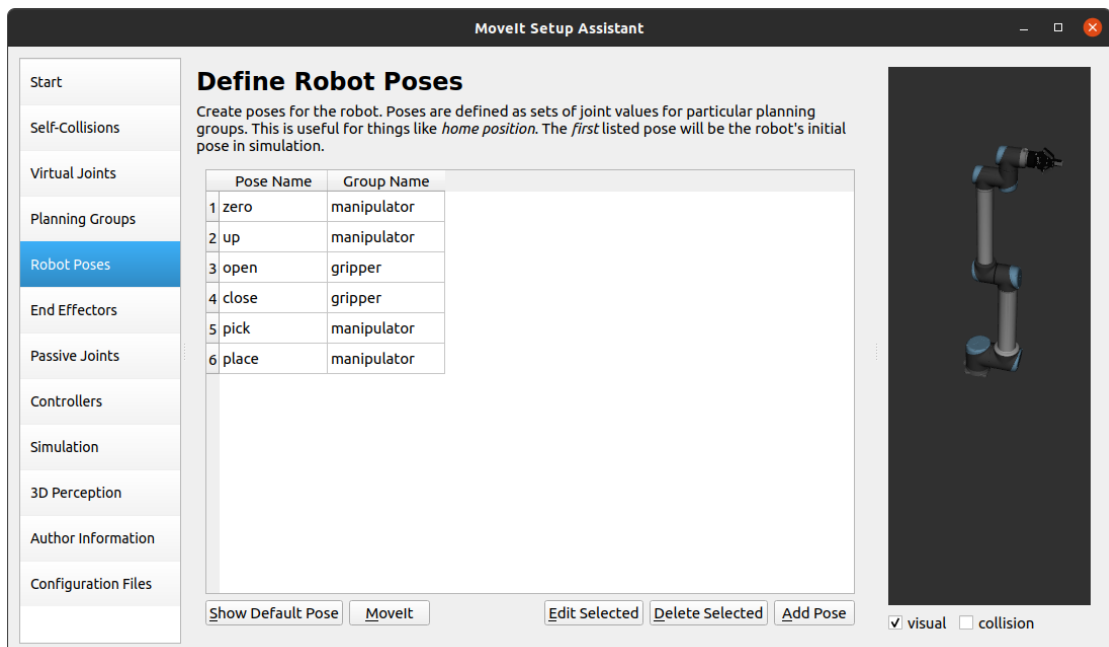


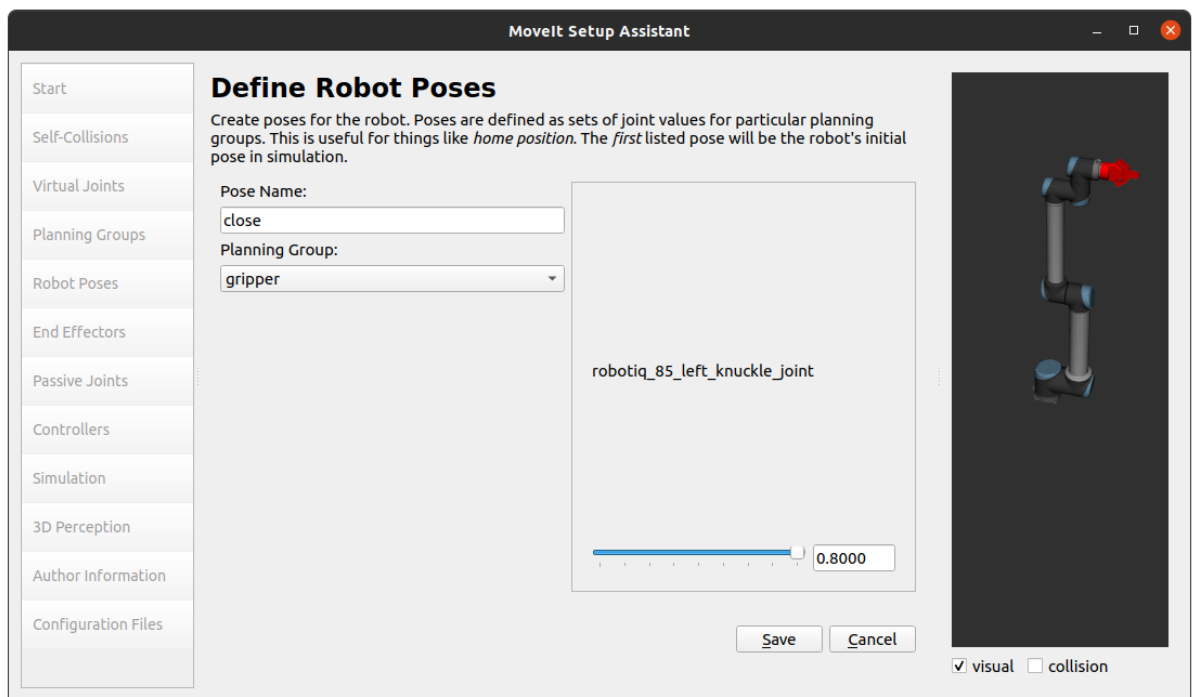
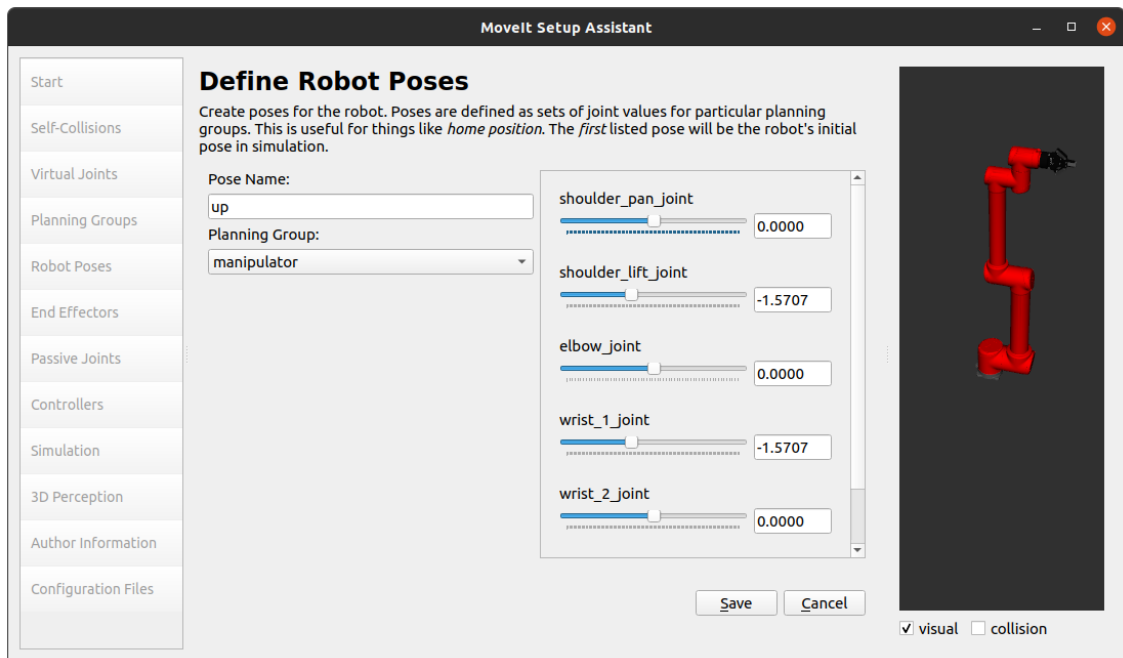


Προσομοίωση Ρομπότ μέσω Υπολογιστή

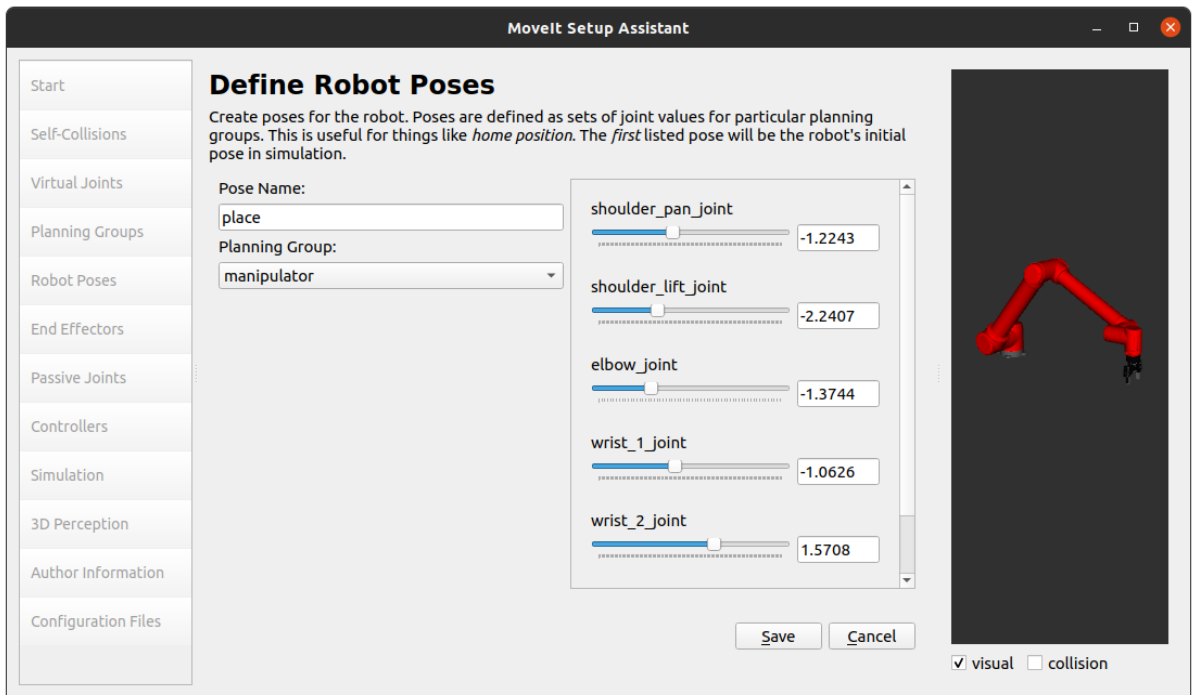
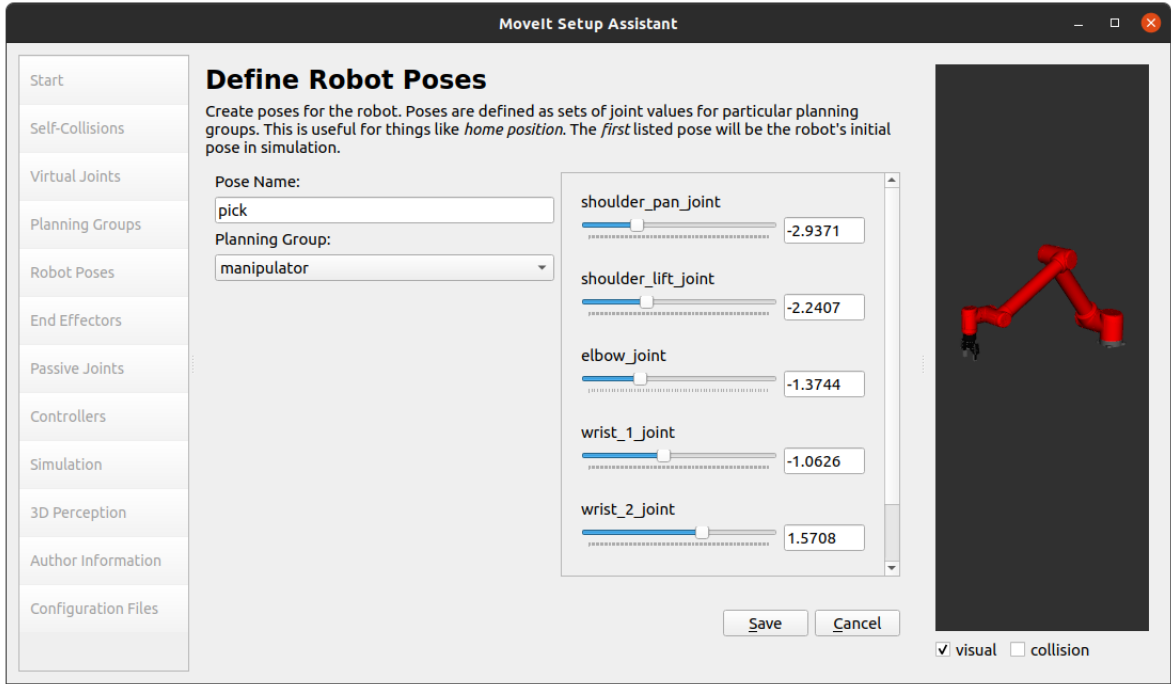
Στην καρτέλα Robot Poses, θα φτιάξουμε τις θέσεις στις οποίες θα επιθυμούμε να πάει ο βραχίονας καθώς και 2 θέσεις για τη λαβή, την ανοιχτή και τη κλειστή. Πατάμε Add Pose, επιλέγουμε όνομα για τη πόζα και ποιο Planning Group θέλουμε. Αυτές θα είναι:

- Για το manipulator:
 - zero: όλα θα είναι 0.
 - up:
 - shoulder_lift_joint: -1.5707
 - wrist_1_joint: -1.5707
 - όλα τα άλλα 0
 - pick:
 - shoulder_pan_joint: -2.9371
 - shoulder_lift_joint: -2.2407
 - elbow_joint: -1.3744
 - wrist_1_joint: -1.0626
 - wrist_2_joint: 1.5708
 - place:
 - shoulder_pan_joint: -1.2243
 - shoulder_lift_joint: -2.2407
 - elbow_joint: -1.3744
 - wrist_1_joint: -1.0626
 - wrist_2_joint: 1.5708
- Για το gripper:
 - open: θα το αφήσουμε στο 0 ώστε να είναι πλήρως ανοιχτή η λαβή.
 - close: θα το βάλουμε 0.8 γιατί από εκεί και έπειτα υπάρχει σύγκρουση και μας βγάζει μήνυμα.

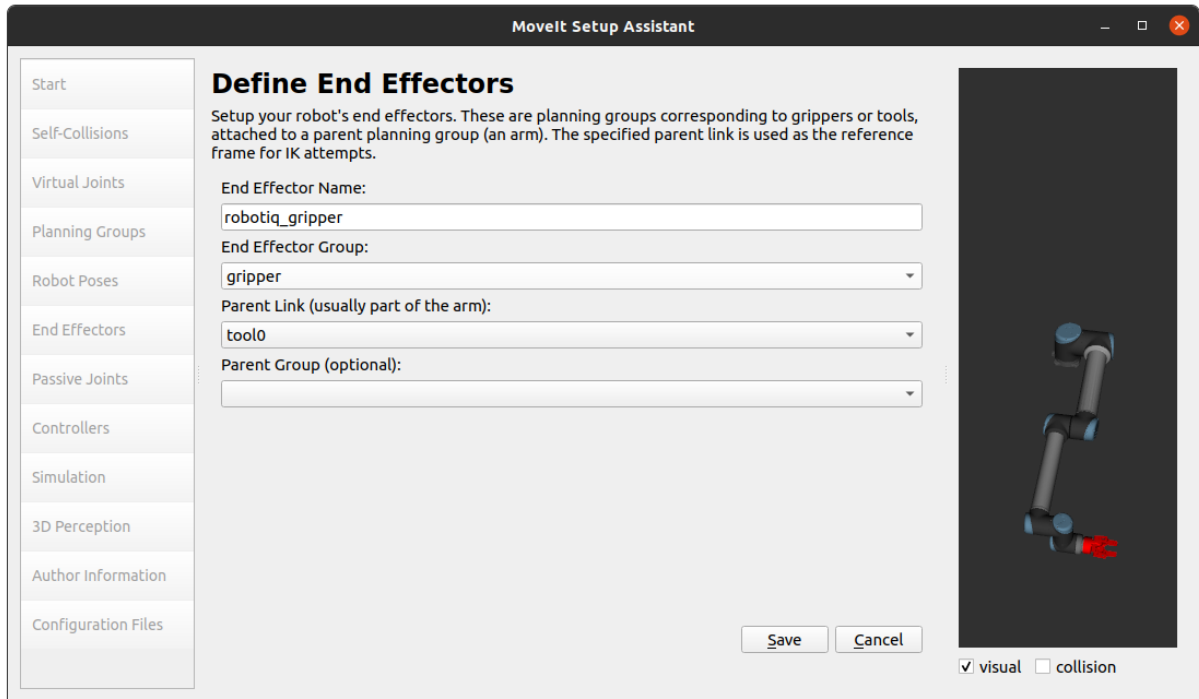




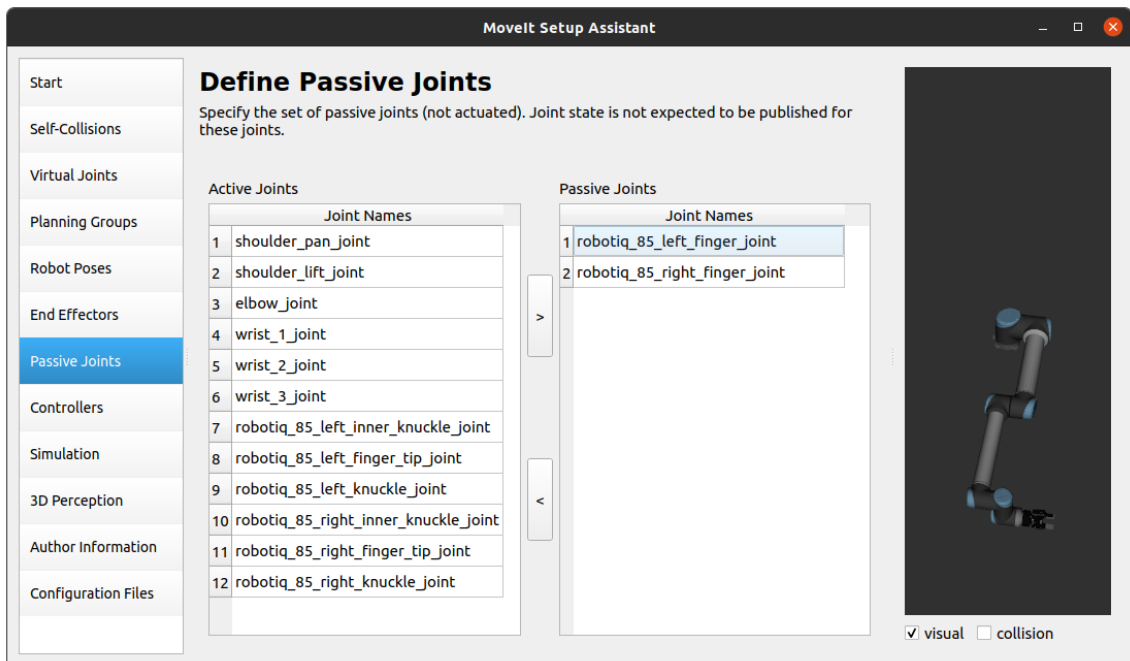
Προσομοίωση Ρομπότ μέσω Υπολογιστή



Στην καρτέλα End Effectors, εισάγουμε στο End Effector Name: `robotiq_gripper`, επιλέγουμε το gripper από το End Effector Group:, στο Parent Link επιλέγουμε το `tool0`.



Στη καρτέλα Passive Joints, επιλέγουμε τα robotiq_85_left_finger_joint, robotiq_85_right_finger_joint έτσι ώστε όταν κινείται ο ένας βραχίονας της λαβής να κινείται και ο άλλος.



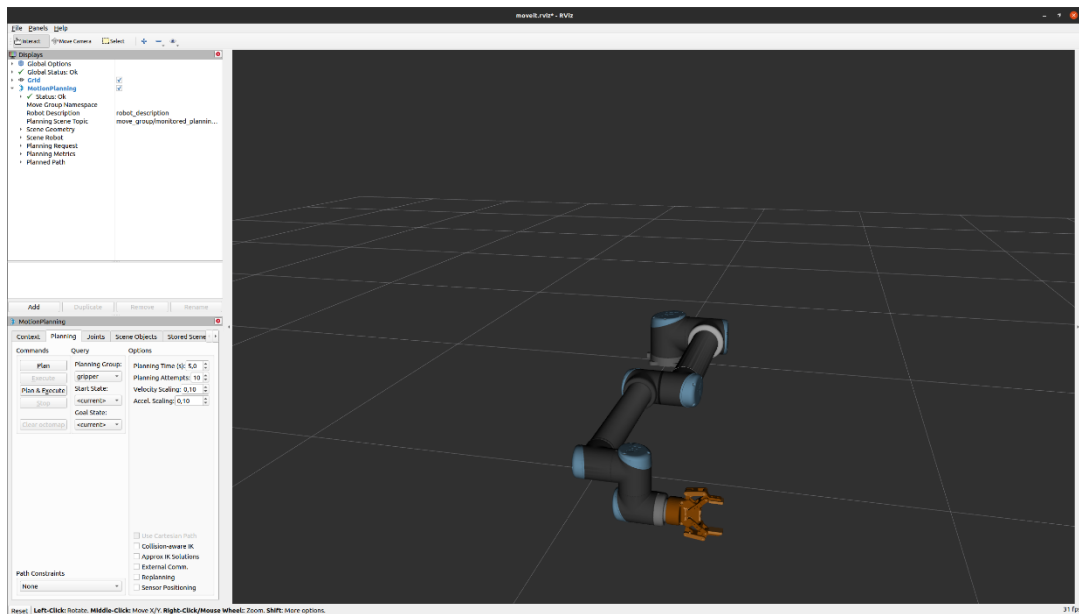
Κάνουμε Generate Package και Exit Setup Assistant. Δημιουργήθηκε, ένα πακέτο με όλο το config.

Ένας άλλος τρόπος για να βρούμε τις γωνίες στις οποίες θα πρέπει να περιστραφούν οι αρθρώσεις του βραχίονα προκειμένου να πάει στις θέσεις pick και place είναι να τρέξουμε μία demo προσομοίωση στο Rviz.

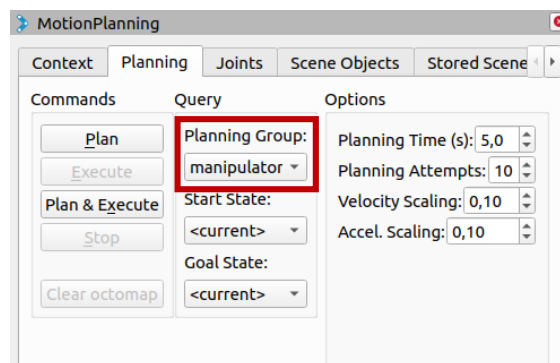
Ανοίγουμε το terminal και τρέχουμε την εντολή:

```
roslaunch ur10_gripper_moveit_config demo.launch
```

Προσομοίωση Ρομπότ μέσω Υπολογιστή
Ανοίγει το παράθυρο του Rviz:

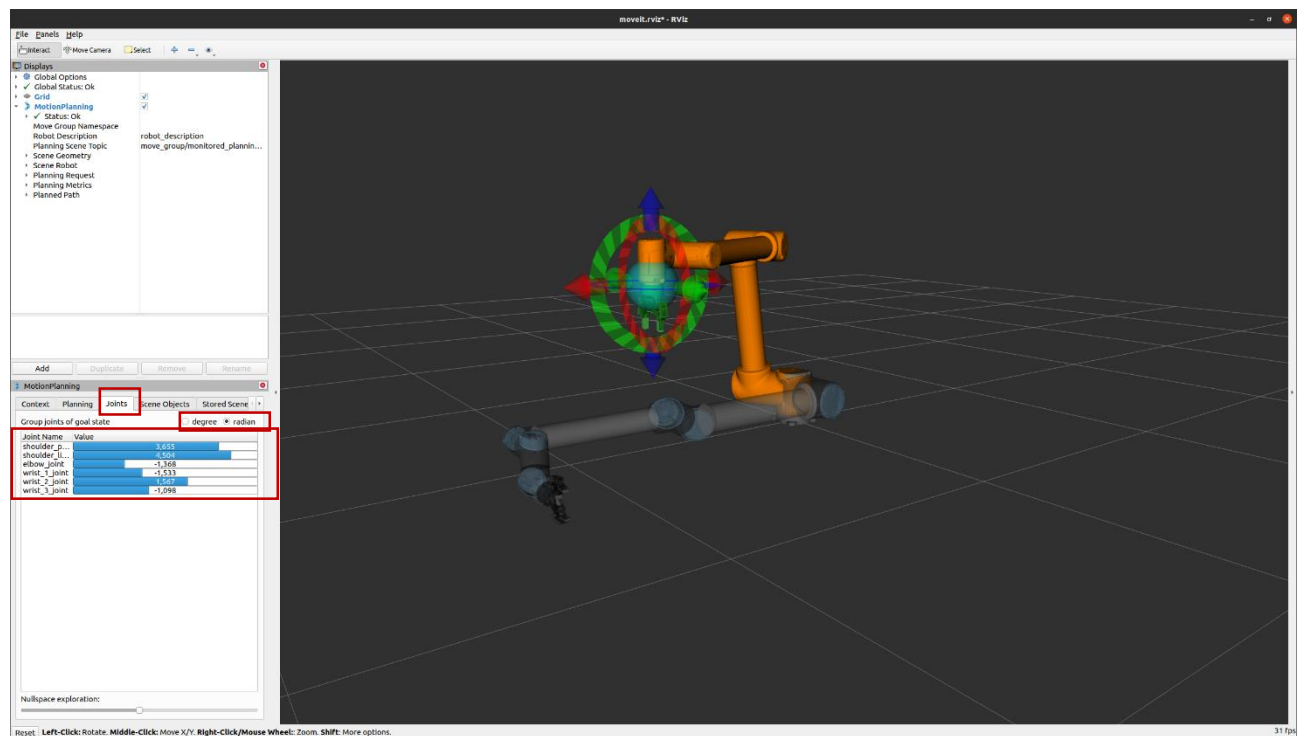
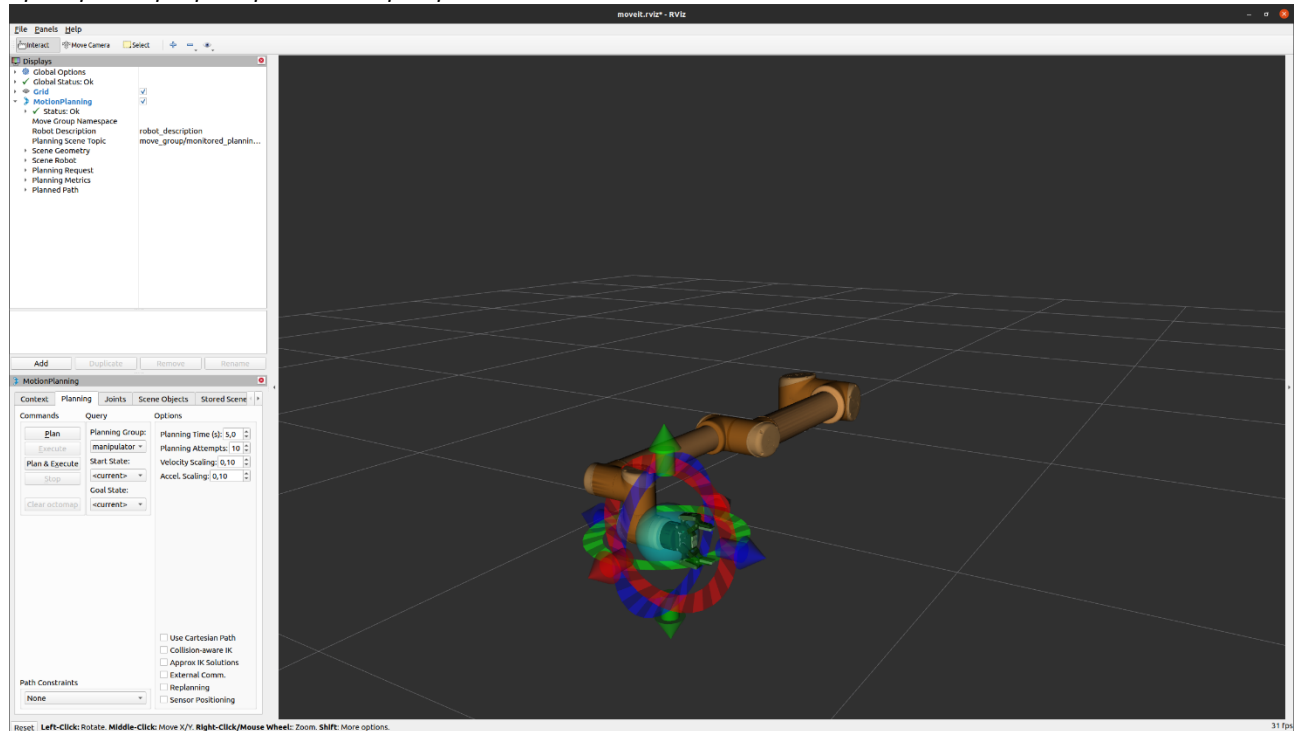


Επιλέγουμε το manipulator απο το Planning Group.



Εμφανίζεται στην άκρη του βραχίονα μια γαλάζια σφαίρα με 3 δαχτυλίδια και 3 βέλη. Κουνάμε τη σφαίρα και τη πηγαίνουμε στη κατάλληλη θέση για να πιάσουμε το αντικείμενο που θέλουμε.

Προσομοίωση Ρομπότ μέσω Υπολογιστή

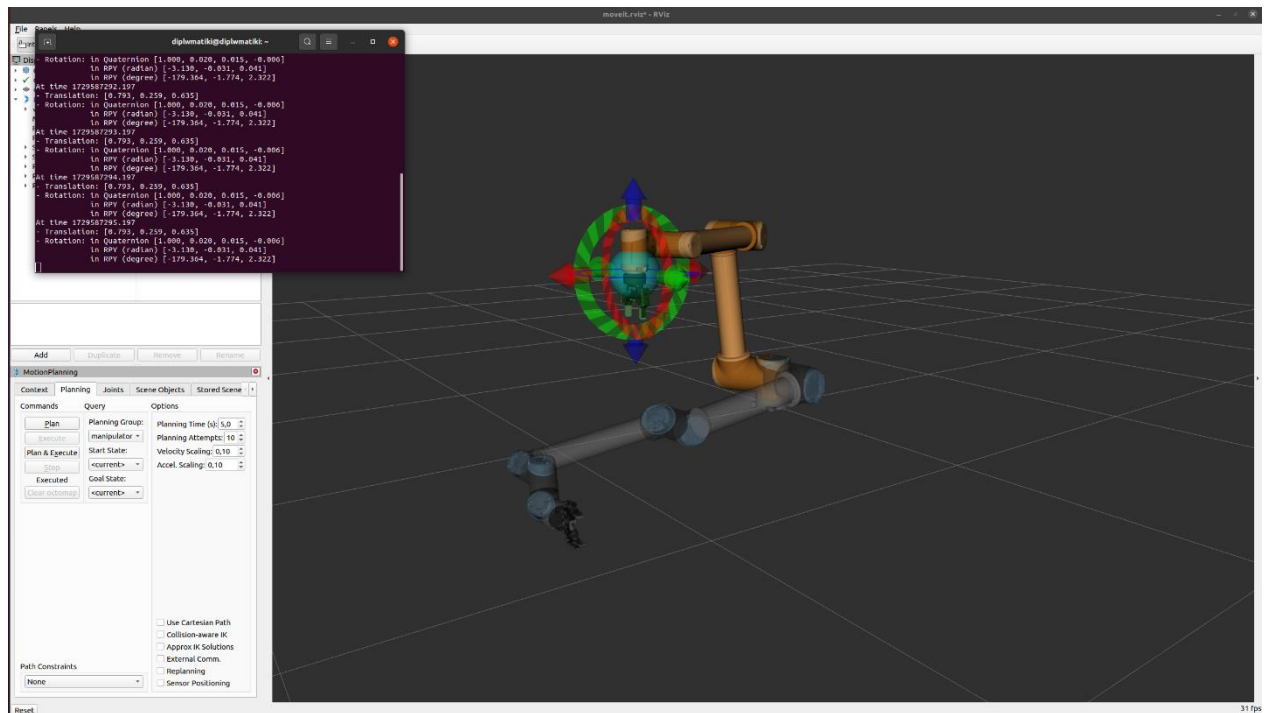
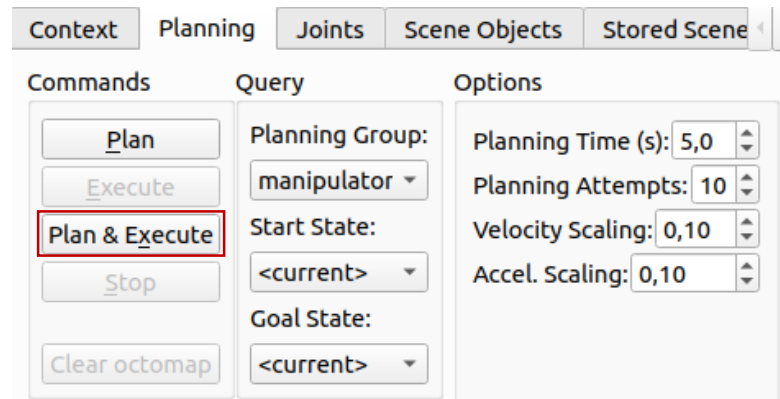


Μπορούμε να πάμε στη καρτέλα Joints και να χειριστούμε κάθε άρθρωση ώστε να διαμορφώσουμε τη θέση του. Αυτή θα είναι η θέση που θα πιάνουμε το αντικείμενο. Για να τη σημειώσουμε επιλέγουμε τα radians και σημειώνουμε για κάθε άρθρωση τη τιμή της, τις οποίες μπορούμε να εισάγουμε στις πόζες στο MoveIt Setup Assistant στη καρτέλα με τις πόζες.

Εάν θέλουμε να σημειώσουμε το Orientation και Translation της άκρης του βραχίονα μας (End Effector) έτσι ώστε να φτιάξουμε ένα script που να πηγαίνει αυτόματα στις θέσεις pick και place θα ανοίξουμε το terminal και θα γράψουμε την παρακάτω εντολή:

Προσομοίωση Ρομπότ μέσω Υπολογιστή
roslaunch tf_echo /base_link /tool0

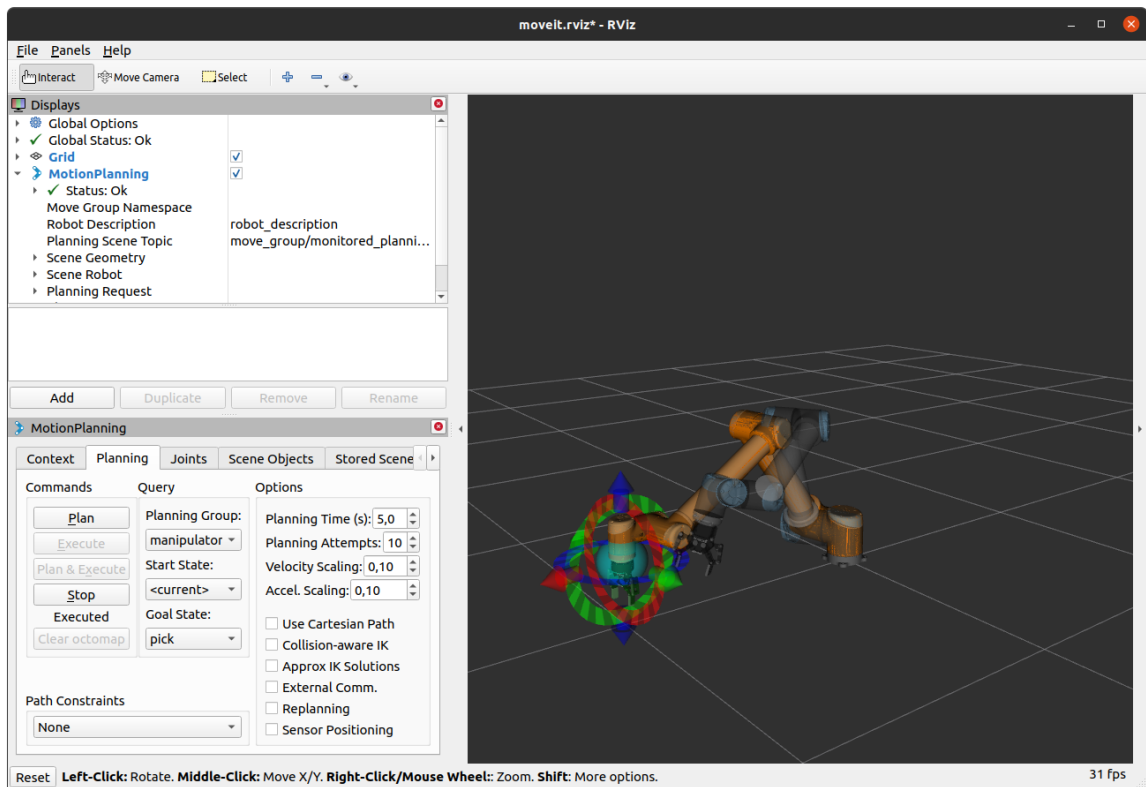
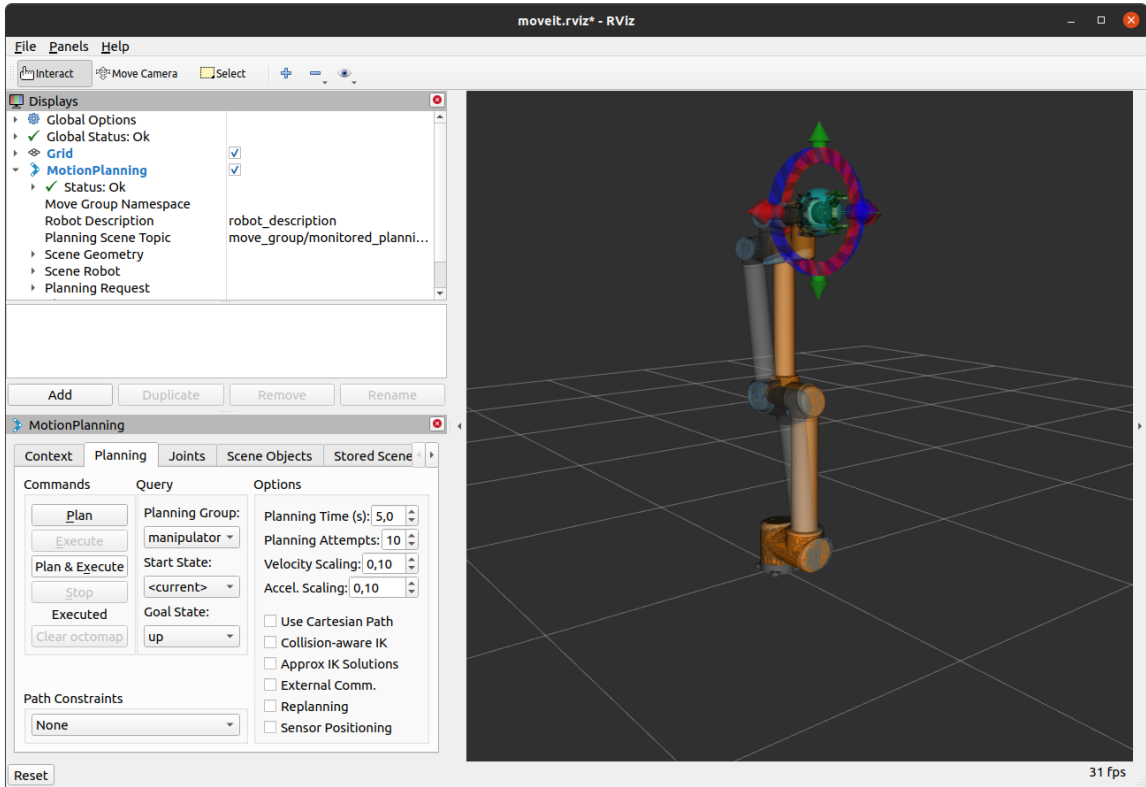
Με αυτή την εντολή βλέπουμε τη θέση του end effector(tool0) σε σχέση με τη βάση μας (base_link). Αφού τρέξουμε αυτή την εντολή, πατάμε στο Rviz, Plan & Execute και περιμένουμε να φτάσει στη τελική του θέση ο βραχίονας. Όταν θα φτάσει θα σημειώσουμε τις συντεταγμένες από το terminal.



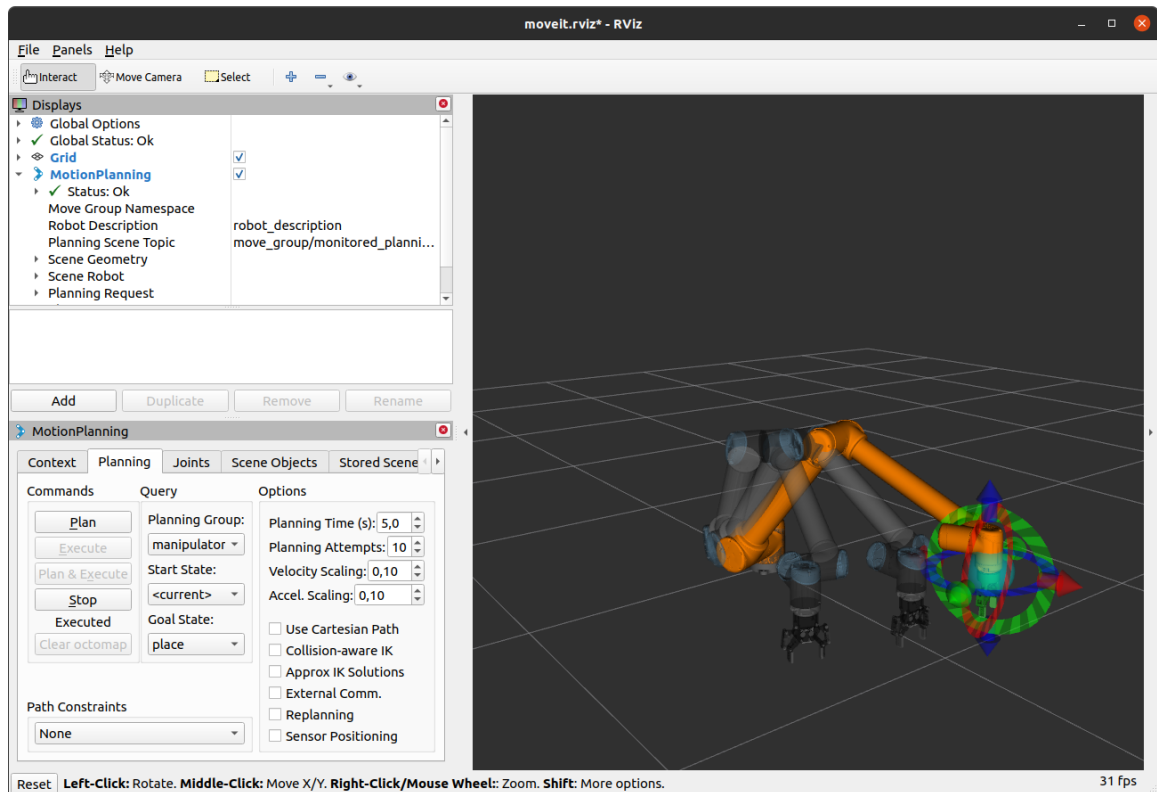
Αυτά που μας ενδιαφέρουν είναι το Translation και το Rotation σε radians.

Τώρα στο Rviz, αν επιλέξουμε το manipulator στο Planning Group και στο Goal State: μια από τις πόζες που φτιάξαμε (zero, up, pick, place) μπορούμε να τεστάρουμε τις πόζες μας αν είναι σωστές.

Προσομείωση Ρομπότ μέσω Υπολογιστή



Προσομοίωση Ρομπότ μέσω Υπολογιστή



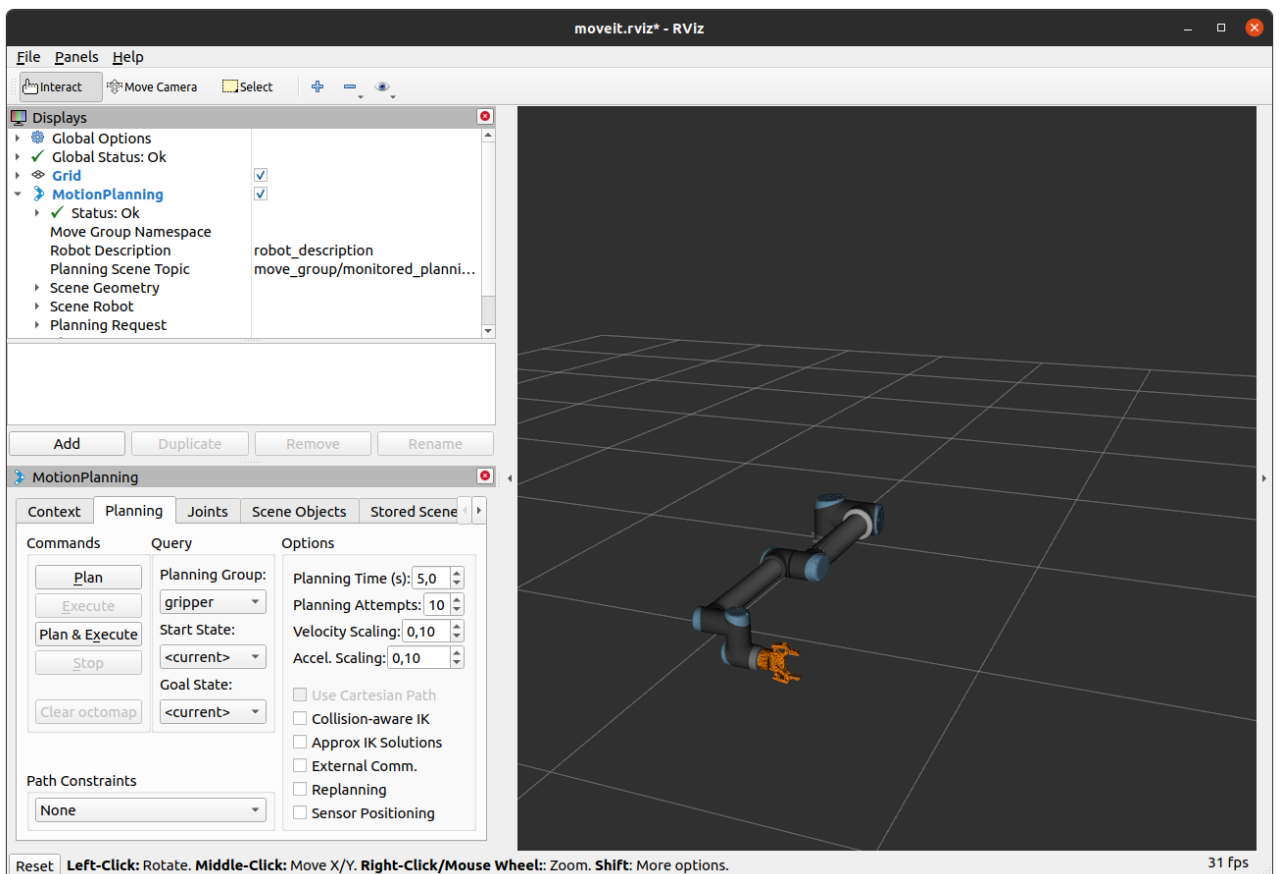
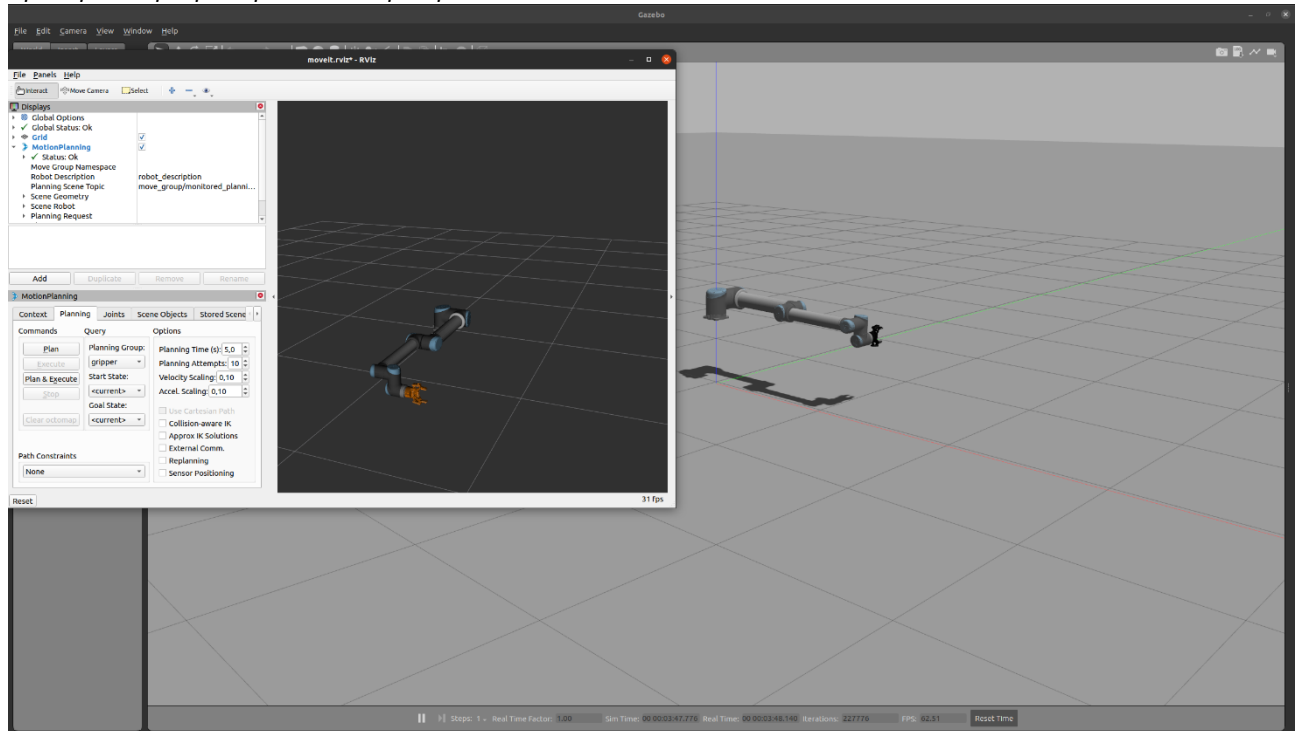
Τώρα θα τρέξουμε τη προσομοίωση στο Rviz και στο Gazebo. Για να το κάνουμε αυτό θα τρέξουμε το αρχείο με το script που καλεί το βραχίονα μας και όλα όσα χρειαζόμαστε για να ανοίξει το Rviz και το Gazebo.

Για να το κάνουμε αυτό τρέχουμε τη παρακάτω εντολή στο terminal:

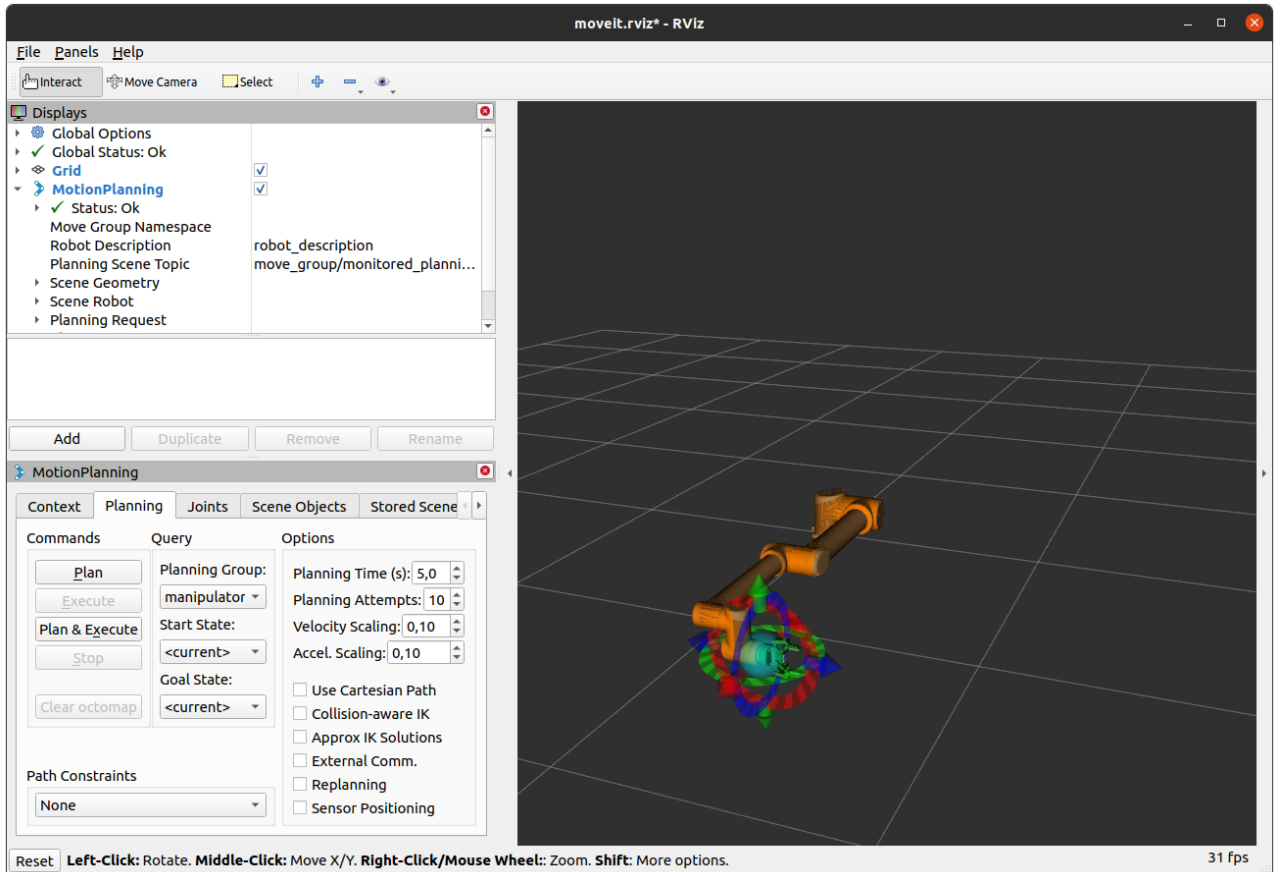
```
roslaunch ur10_diplwmatiki spawn_ur10_gripper.launch
```

Βλέπουμε ότι μας ανοίγει το Gazebo, πατάμε το play στο κάτω μέρος του παραθύρου και ανοίγει και το RViz.

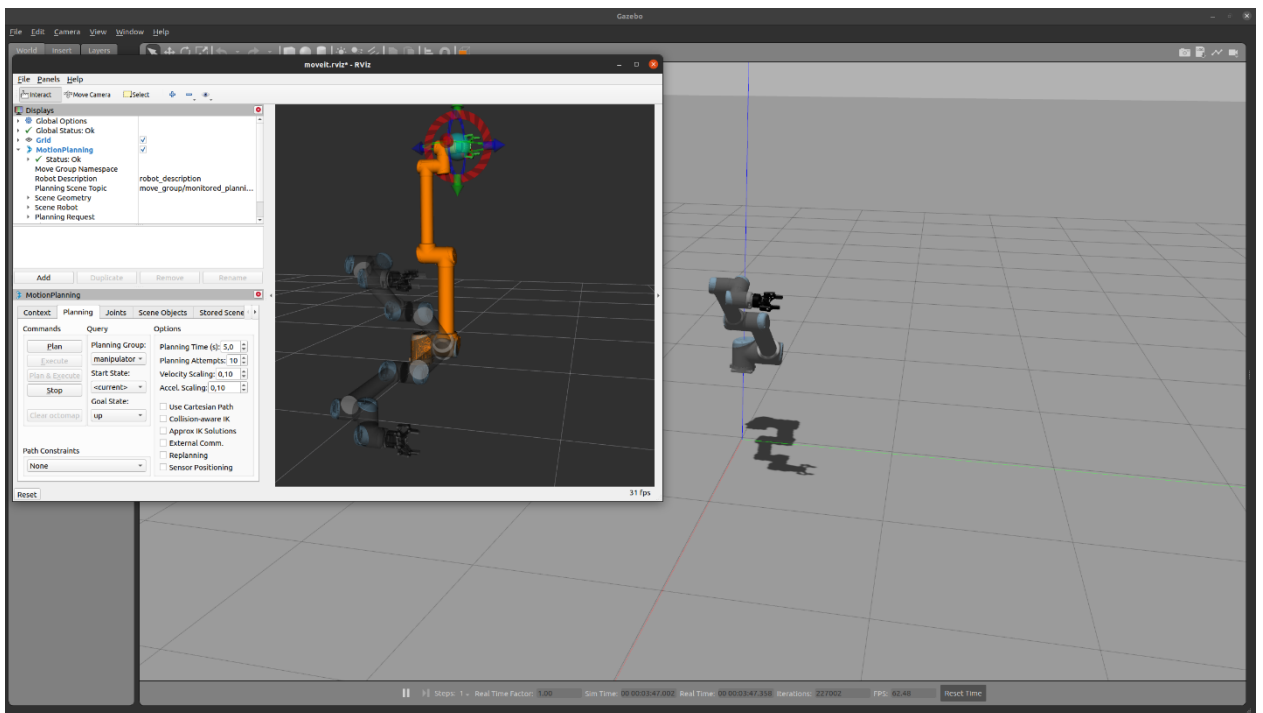
Προσομοίωση Ρομπότ μέσω Υπολογιστή



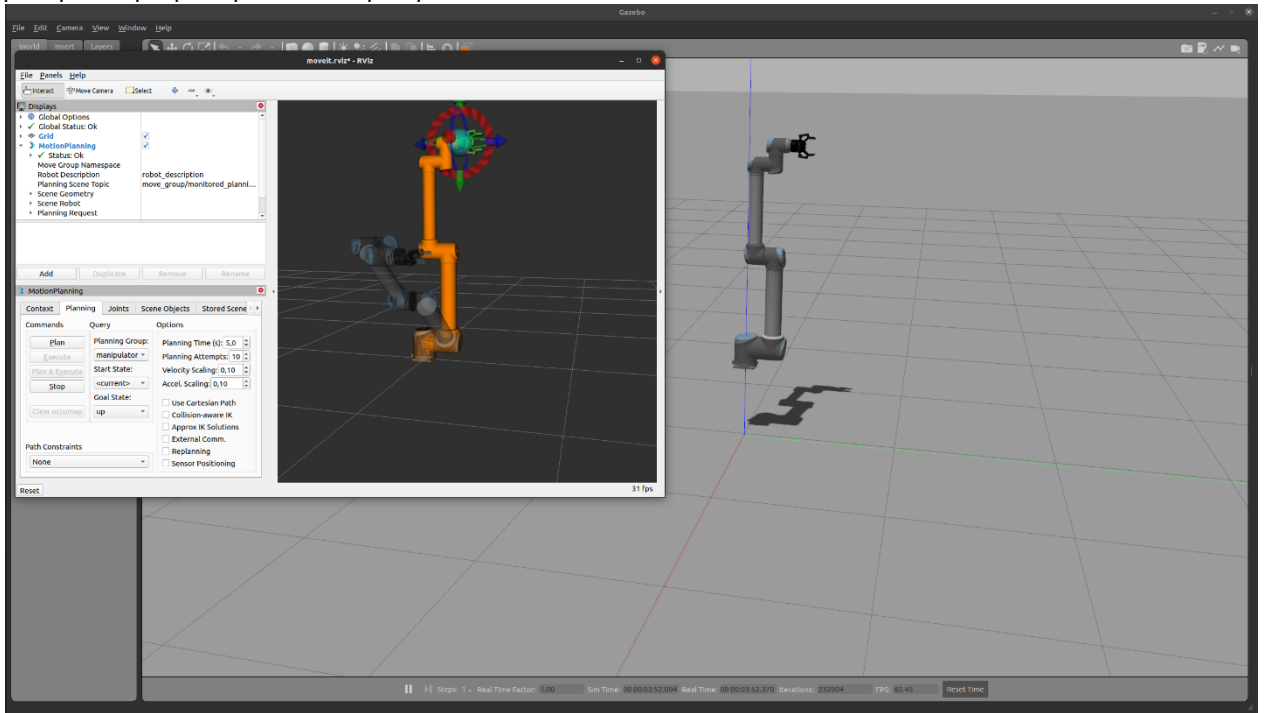
Προσομοίωση Ρομπότ μέσω Υπολογιστή



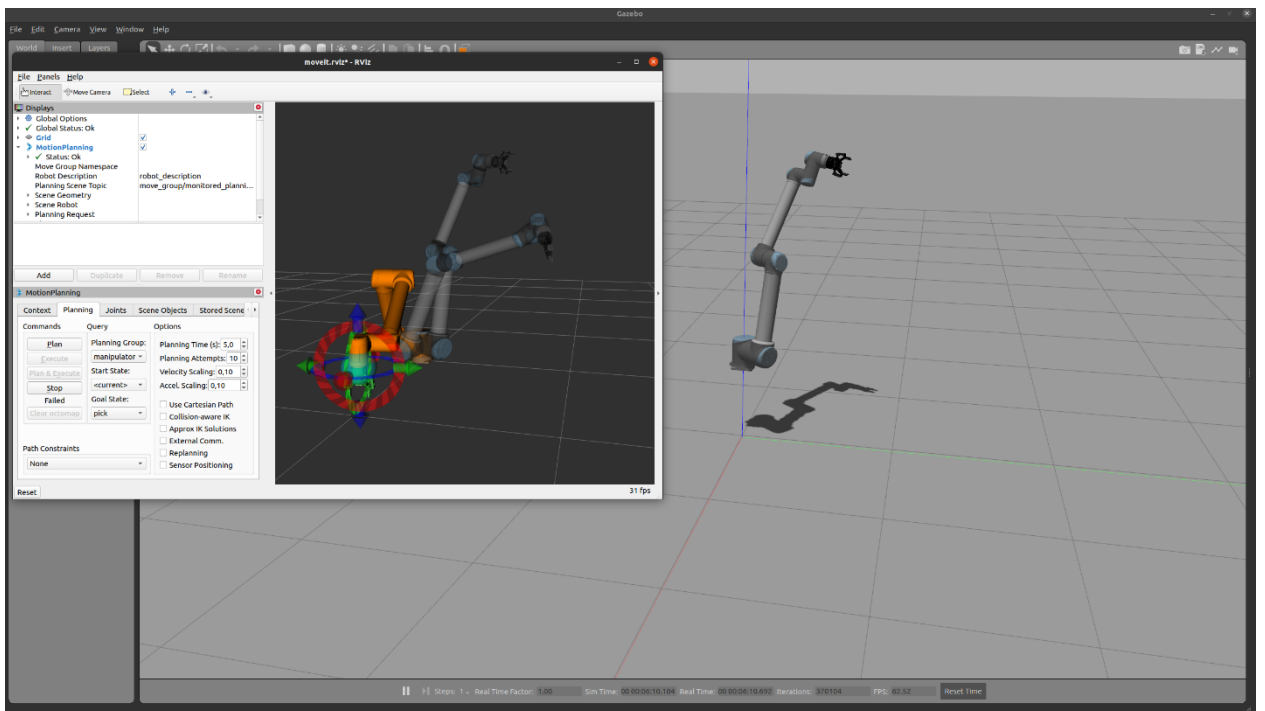
Επιλέγουμε το manipulator στο Planning Group, επιλέγουμε τη θέση up στο Goal State και πατάμε Plan & Execute. Παρατηρούμε στο Gazebo ότι ο βραχίονας πηγαίνει στη θέση αυτή.



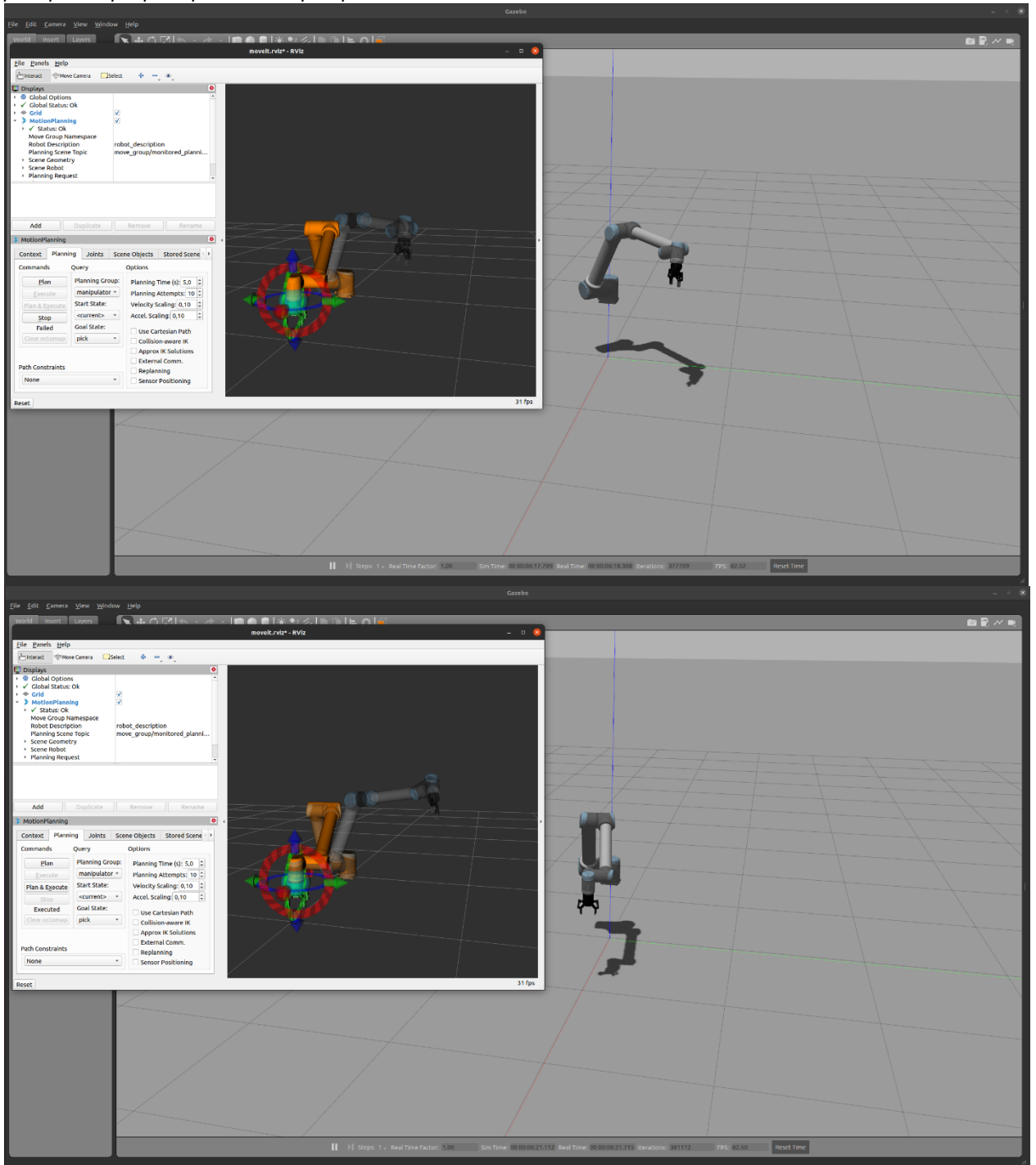
Προσομείωση Ρομπότ μέσω Υπολογιστή



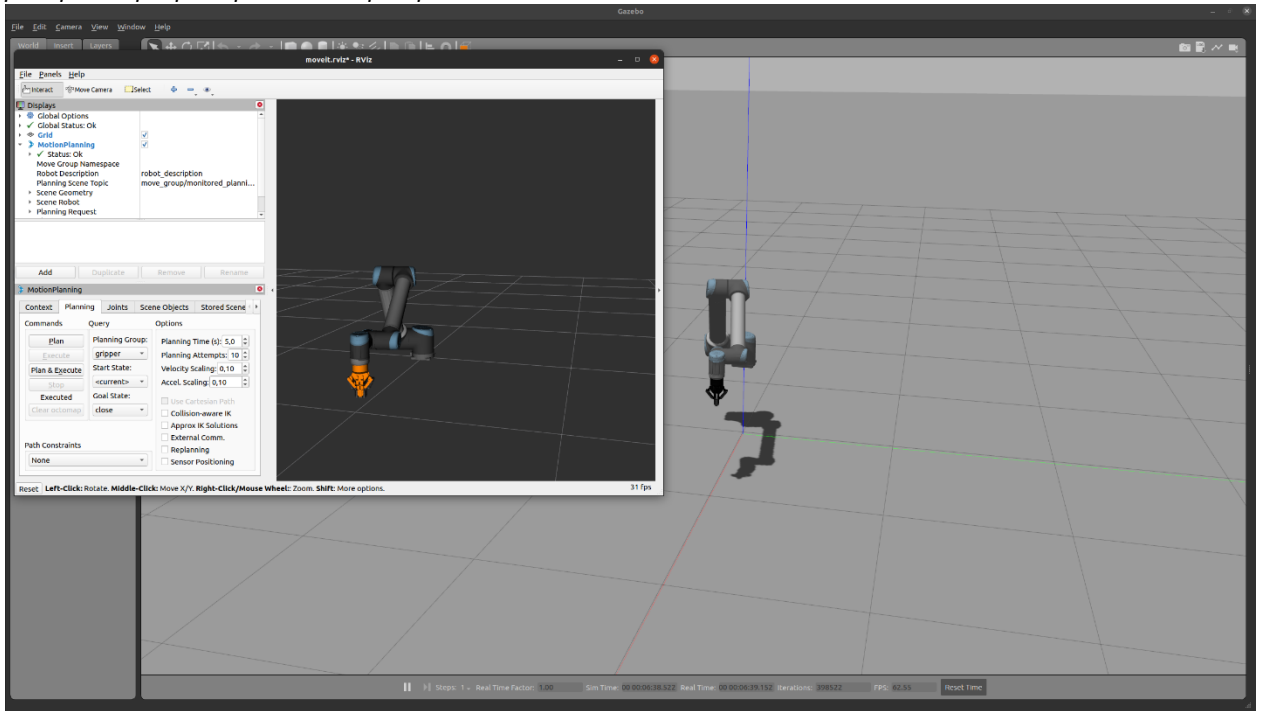
Έπειτα, διαλέγουμε να πάει στη θέση pick και αφού πάει εκεί επιλέγουμε το gripper στο Planning Group και close στο Goal State.



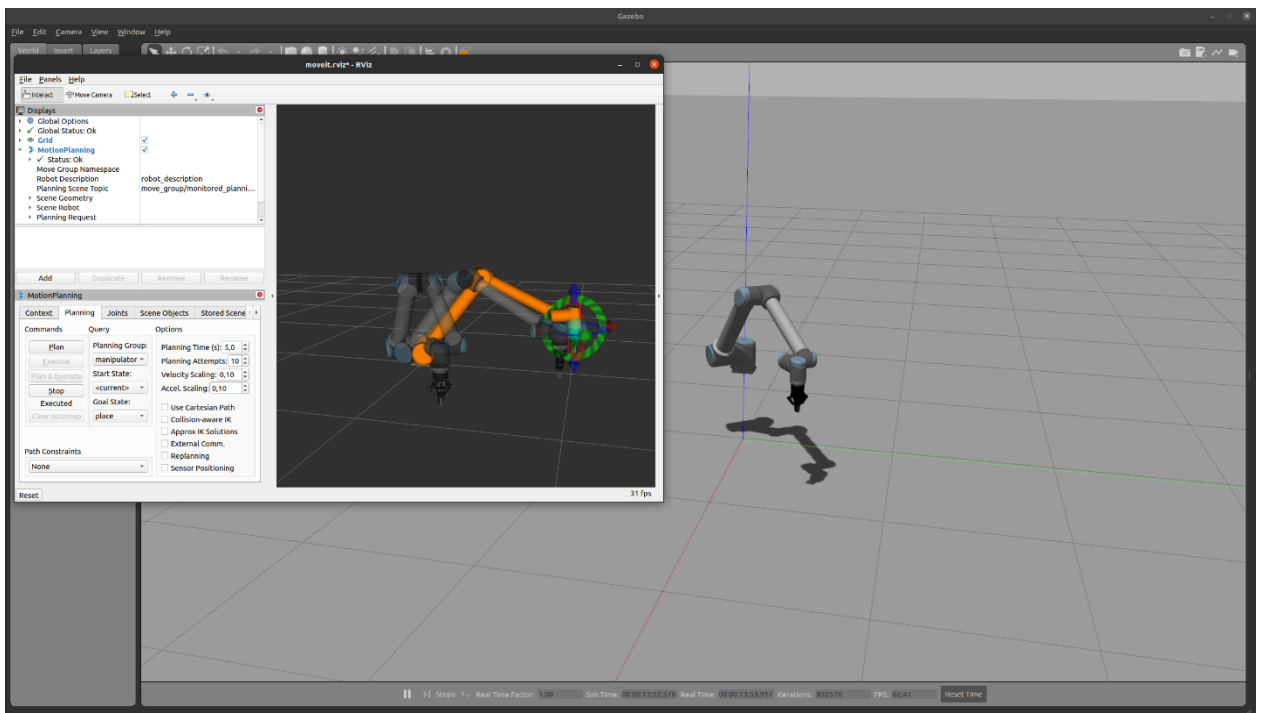
Προσομοίωση Ρομπότ μέσω Υπολογιστή



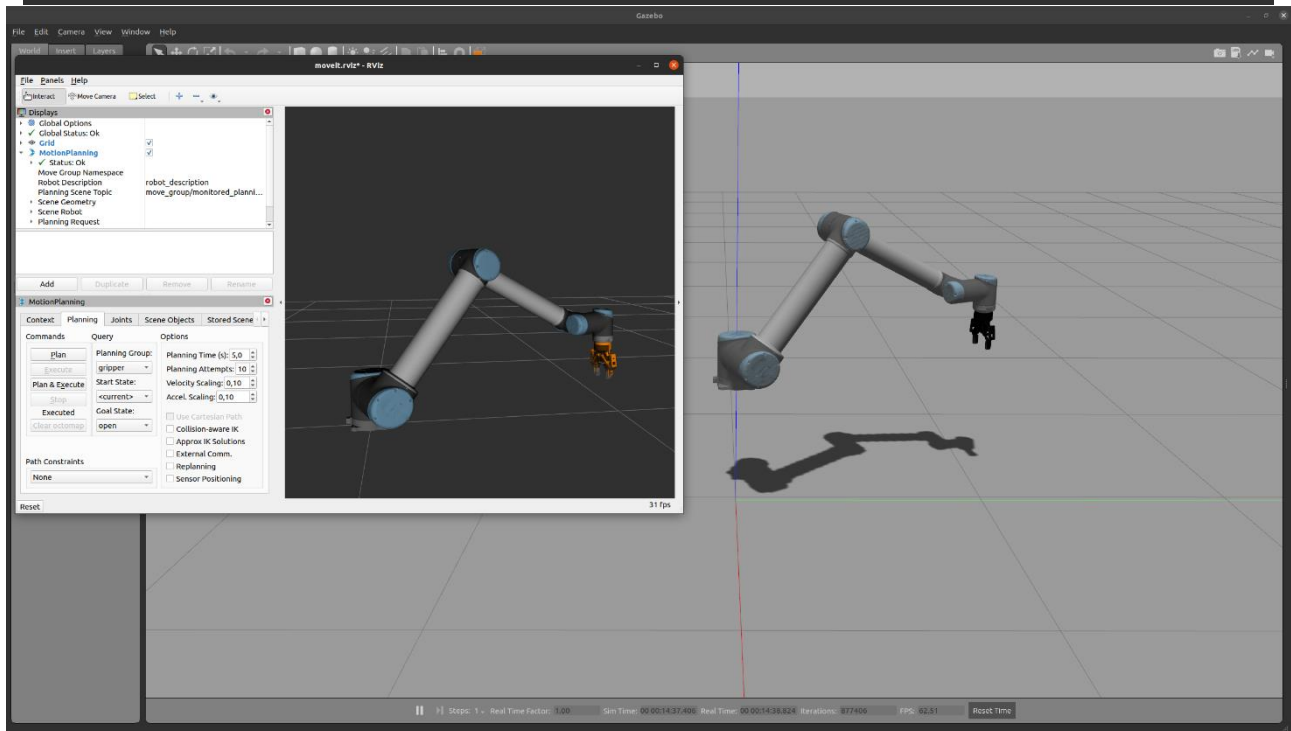
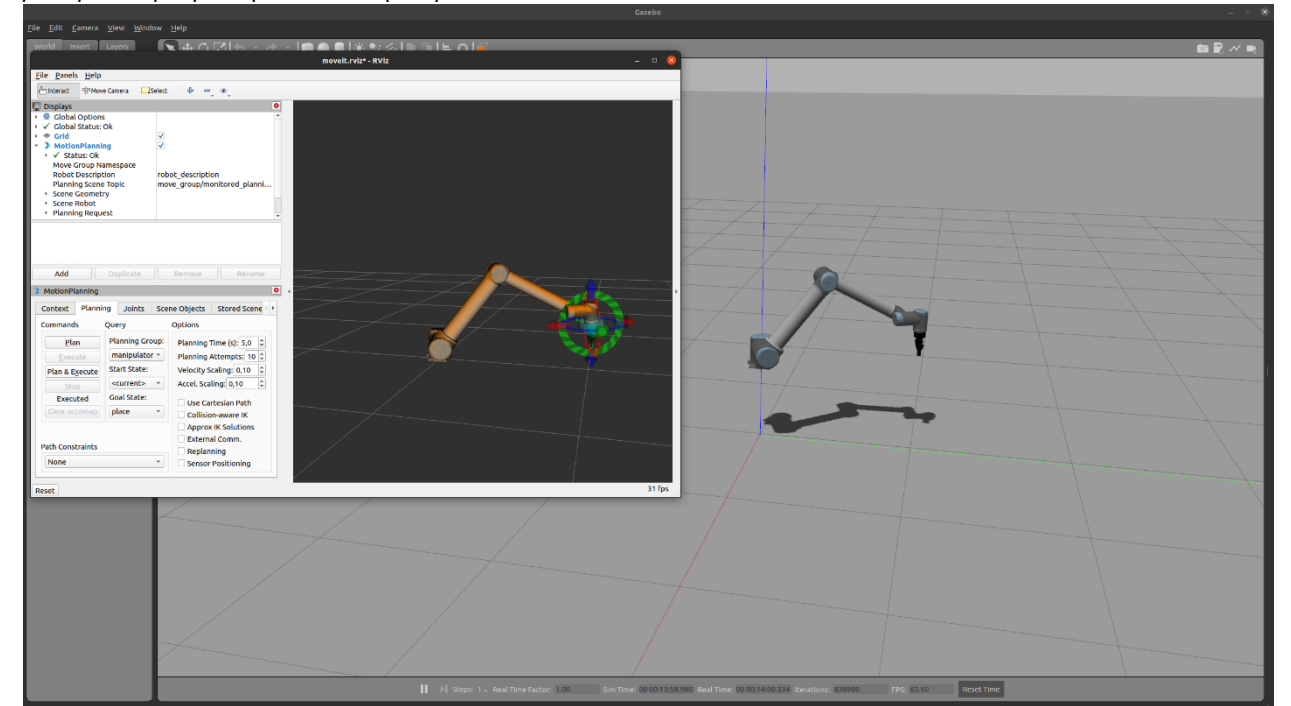
Προσομοίωση Ρομπότ μέσω Υπολογιστή



Μετά, θα το βάλουμε να πάει στη θέση place και να ανοίξει το βραχίονα.

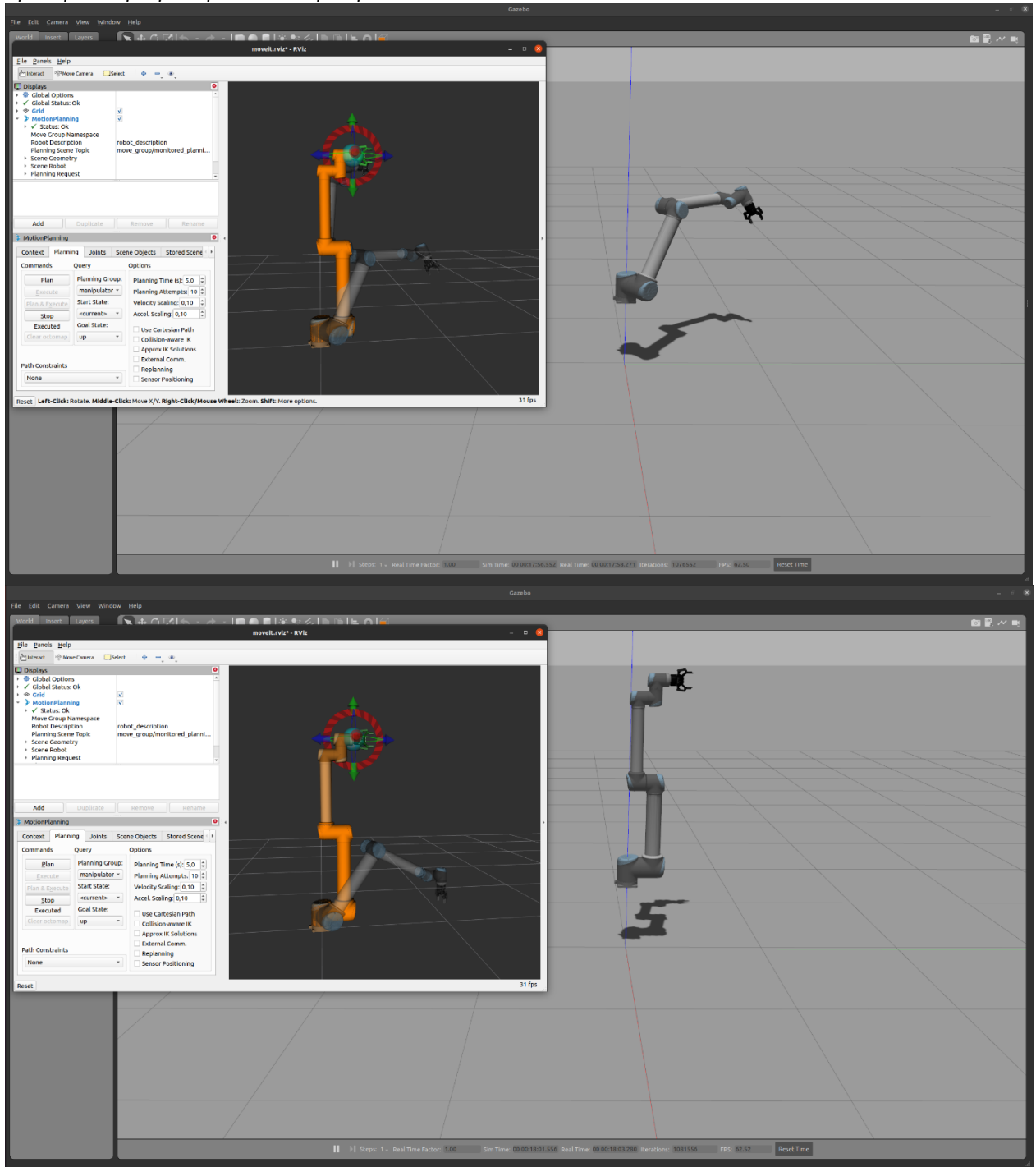


Προσομοίωση Ρομπότ μέσω Υπολογιστή



Τέλος, πηγαίνουμε το βραχίονα πάλι στην όρθια θέση.

Προσομοίωση Ρομπότ μέσω Υπολογιστή



ΚΕΦΑΛΑΙΟ 3^ο : ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα εργασία, αναλύσαμε και συγκρίναμε δύο κορυφαία εργαλεία προσομοίωσης ρομπότ: το CoppeliaSim και το Gazebo. Κάθε εργαλείο έχει τα δικά του πλεονεκτήματα και περιορισμούς, και η επιλογή του κατάλληλου εργαλείου εξαρτάται από τις συγκεκριμένες ανάγκες και απαιτήσεις του χρήστη, όπως αναλύεται ακολούθως:

Ολοκλήρωση με το ROS

- Στο Gazebo, παρατηρούμε ότι η ενσωμάτωσή του με το ROS είναι ιδιαίτερα βαθιά και ισχυρή. Είναι το προεπιλεγμένο εργαλείο προσομοίωσης για το ROS, κάτι που διευκολύνει την ενσωμάτωση ρομποτικών συστημάτων και αισθητήρων με έτοιμα plugins που υποστηρίζουν την επικοινωνία ROS-Gazebo. Η δυνατότητα δημοσίευσης και εγγραφής σε ROS θέματα και υπηρεσίες καθιστά το Gazebo εξαιρετικά πρακτικό για όσους εργάζονται με ρομποτικές εφαρμογές που βασίζονται στο ROS.
- Στο CoppeliaSim, η υποστήριξη του ROS είναι λιγότερο άμεση και απαιτείται μεγαλύτερη προσπάθεια για τη λειτουργία του. Παρότι μπορεί να λειτουργήσει μέσω plugin, χρειάζεται να τρέχει παράλληλα σε άλλο τερματικό, γεγονός που καθιστά την εμπειρία χρήσης λιγότερο ενσωματωμένη. Παρ' όλα αυτά, η λειτουργία ROS είναι εφικτή και προσφέρει ευελιξία μέσω ενός εκτεταμένου API.

Δημιουργία Κόσμων και Σύνθεση Ρομπότ

- Στο CoppeliaSim, μπορούμε να δημιουργήσουμε πολύ εύκολα κόσμους και μοντέλα ρομπότ, καθώς η πλατφόρμα προσφέρει γραφική διεπαφή και λειτουργίες "drag-and-drop". Αυτό διευκολύνει τη γρήγορη προτυποποίηση και την οπτική αναπαράσταση των ρομποτικών στοιχείων. Η δυνατότητα αυτή είναι εξαιρετικά χρήσιμη για όσους από εμάς αναζητούμε έναν πιο άμεσο τρόπο δημιουργίας και τροποποίησης προσομοιώσεων.
- Στο Gazebo, η διαδικασία δημιουργίας κόσμων είναι πιο σύνθετη και απαιτεί τη χρήση XML αρχείων όπως τα SDF και URDF για τη διαχείριση και προσαρμογή μοντέλων. Ενώ μπορούμε να αποκτήσουμε πρόσβαση σε μια online βάση δεδομένων μοντέλων, η χρήση του απαιτεί μεγαλύτερη εξοικείωση με τεχνικά εργαλεία όπως το Blender ή το Google Sketchup.

Φυσική Προσομοίωση

- Το Gazebo προσφέρει προηγμένη φυσική προσομοίωση, χρησιμοποιώντας μηχανές όπως ODE, Bullet και Simbody, που μας επιτρέπουν να προσομοιώσουμε με μεγάλη ακρίβεια δυνάμεις, τριβή, βαρύτητα και άλλες φυσικές επιρροές. Αυτό το καθιστά ιδανικό για εφαρμογές όπου χρειαζόμαστε απόλυτη ακρίβεια στη φυσική προσομοίωση.
- Στο CoppeliaSim, η προσομοίωση είναι λιγότερο ακριβής, αλλά προσφέρει μεγάλη ευελιξία, επιτρέποντάς μας να επιλέξουμε μεταξύ πολλών φυσικών μηχανών. Αυτό καθιστά το CoppeliaSim πιο ελαφρύ και εύχρηστο για περιπτώσεις που δεν απαιτούμε απόλυτη ακρίβεια, αλλά θέλουμε ευελιξία και αποδοτικότητα.

Χρήση Πόρων και Απόδοση

- Στο Gazebo, παρατηρούμε ότι η απαίτηση σε πόρους είναι μεγαλύτερη λόγω της υψηλής ακρίβειας που προσφέρει στη φυσική προσομοίωση. Αυτό συνεπάγεται μεγαλύτερη κατανάλωση CPU και μνήμης, ειδικά σε πολύπλοκες προσομοιώσεις με πολλούς αισθητήρες και αλληλεπιδράσεις.
- Το CoppeliaSim από την άλλη, καταναλώνει λιγότερους πόρους και μπορεί να λειτουργήσει ομαλότερα σε συστήματα με χαμηλότερη ισχύ. Αυτό το καθιστά ιδανικό για περιπτώσεις που χρειαζόμαστε πιο ελαφριά προσομοίωση.

Προγραμματιστική Υποστήριξη και Αυτοματοποίηση

- Στο CoppeliaSim, η υποστήριξη για πολλαπλές γλώσσες προγραμματισμού, όπως Lua,

Προσομοίωση Ρομπότ μέσω Υπολογιστή

Python, C++, και Java, μας δίνει τη δυνατότητα να αυτοματοποιήσουμε και να ελέγξουμε τις προσομοιώσεις με μεγαλύτερη ευκολία. Η ευελιξία αυτή είναι πολύτιμη, ειδικά σε ερευνητικά και εκπαιδευτικά περιβάλλοντα όπου χρειαζόμαστε γρήγορες προσαρμογές.

- Το Gazebo υποστηρίζει επίσης Python και C++, αλλά η διαδικασία προγραμματιστικής διαχείρισης είναι πιο περίπλοκη. Ωστόσο, η ενσωμάτωσή του με το ROS παρέχει ισχυρά εργαλεία για την αυτοματοποίηση των ρομποτικών εφαρμογών

Εν κατακλείδι, σε εφαρμογές που απαιτούν υψηλή ακρίβεια στη φυσική προσομοίωση και βαθιά ενσωμάτωση με το ROS, το Gazebo είναι η κατάλληλη επιλογή. Αντιλαμβανόμαστε ότι αν και είναι πιο απαιτητικό σε πόρους, προσφέρει εξαιρετική ακρίβεια και ευελιξία στη συνεργασία με ρομποτικά συστήματα. Αν, από την άλλη, δίνουμε προτεραιότητα στην ευκολία χρήσης, την ταχύτερη προτυποποίηση και τη χαμηλότερη κατανάλωση πόρων, τότε το CoppeliaSim είναι η καλύτερη επιλογή. Μπορεί να μην προσφέρει την ίδια ακρίβεια στη φυσική, αλλά είναι εξαιρετικά αποδοτικό και εύχρηστο σε εκπαιδευτικά και ερευνητικά περιβάλλοντα

Βιβλιογραφία – Διαδικτυακές Πηγές

- Διονύσης Κανδρής, Αναστασία Βελώνη. Βιομηχανική Πληροφορική και Αυτοματισμός, Εκδόσεις Τζιόλα, 2023.
- Coppelia Robotics, "CoppeliaSim User Manual," <https://manual.coppeliarobotics.com/>
- Open Robotics, "Gazebo: Robot Simulation Made Easy," <https://gazebosim.org/>
- ROS Wiki, "ROS Documentation," <http://wiki.ros.org/>
- ROS Noetic Ninjemys, "ROS Noetic Documentation", <https://wiki.ros.org/noetic>
- MoveIt, <https://moveit.ai/>
- Open Robotics, "Robot Operating System (ROS)," <https://www.ros.org/>
- GitHub Repository, "ROS: GitHub Repository," <https://github.com/ros/ros>
- GitHub Repository, "universal_robot", https://github.com/ros-industrial/universal_robot
- GitHub Repository, "robotiq_description", https://github.com/LearnRoboticsWROS/robotiq_description
- GitHub Repository "roboticsgroup_upatras_gazebo_plugins". https://github.com/roboticsgroup/roboticsgroup_upatras_gazebo_plugins