

# Μελέτη και Ανάλυση Ρομποτικών Προσομοιωτών

---

Η Διπλωματική Εργασία  
παρουσιάστηκε ενώπιον  
του Διδακτικού Προσωπικού  
του Πανεπιστημίου Αιγαίου

---

Σε Μερική Εκπλήρωση  
των απαιτήσεων για την απόκτηση  
του διπλώματος του  
Μηχανικού Πληροφοριακών  
και Επικοινωνιακών Συστημάτων

---

Του

Γεώργιου Εξίζογλου – 321/2013053

Ιούνιος 2019

Η ΤΡΙΜΕΛΗΣ ΕΠΙΤΡΟΠΗ ΔΙΔΑΣΚΟΝΤΩΝ  
ΕΠΙΚΥΡΩΝΕΙ ΤΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ  
ΤΟΥ  
ΓΕΩΡΓΙΟΥ ΕΞΙΖΟΓΛΟΥ

---

ΚΑΒΑΛΛΙΕΡΑΤΟΥ ΕΡΓΙΝΑ, Επιβλέπων 07/06/2019

Αναπληρώτρια Καθηγήτρια  
Τμήμα Μηχανικών Πληροφοριακών  
και Επικοινωνιακών Συστημάτων

---

ΟΝΟΜΑΤΕΠΩΝΥΜΟ, Μέλος  
Βαθμίδα  
Τμήμα Μηχανικών Πληροφοριακών  
και Επικοινωνιακών Συστημάτων

---

ΟΝΟΜΑΤΕΠΩΝΥΜΟ, Μέλος  
Βαθμίδα  
Τμήμα Μηχανικών Πληροφοριακών  
και Επικοινωνιακών Συστημάτων

---

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ  
ΙΟΥΝΙΟΣ 2019

## Περίληψη

Από τις αρχές του 20ου αιώνα, η προσομοίωση είναι ένα σημαντικό και χρήσιμο ερευνητικό εργαλείο στον τομέα της ρομποτικής. Προσομοίωση ορίζεται η διαδικασία ανάπτυξης και σχεδίασης ενός εικονικού μοντέλου ικανό να μιμηθεί λειτουργίες κάποιου φυσικού ή θεωρητικού συστήματος με την βοήθεια υπολογιστή. Στην τωρινή εποχή της ταχείας ανάπτυξης της τεχνολογίας, πολλοί προσομοιωτές είναι πλέον διαθέσιμοι. Αν και ο κάθε προσομοιωτής είναι για διαφορετικό σκοπό, όλοι τους έχουν κάποια κοινά χαρακτηριστικά.

Στην παρούσα εργασία, γίνεται μελέτη των πιο διάσημων open-source και εμπορικών ρομποτικών προσομοιωτών και μια σύγκριση μεταξύ τους με βάση κάποια κοινά τους χαρακτηριστικά, τα οποία είναι το κόστος, οι δυνατότητες cross-platform, η 3D απεικόνιση, το γεγονός αν είναι open-source ή όχι, η δυνατότητα πολυπρακτορικής προσομοίωσης και ο εντοπισμός συγκρούσεων, με σκοπό την εύρεση του πιο ιδανικού και ολοκληρωμένου προσομοιωτή. Επίσης, διεξάγεται ένα μικρό πείραμα προσομοίωσης στους προσομοιωτές με σκοπό την σύγκριση της απόδοσής τους με βάση κάποιους μετρητές μέσω ειδικού λογισμικού.

**Λέξεις κλειδιά:** προσομοίωση, προσομοιωτής, ρομποτικός προσομοιωτής, αξιολόγηση, Πανεπιστήμιο Αιγαίου

© 2019

Με επιφύλαξη παντός δικαιώματος.

ΓΕΩΡΓΙΟΣ ΕΞΙΖΟΓΛΟΥ

Τμήμα Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ

## Abstract

Since the beginning of the 20th century, simulation has been an important and useful research tool in the field of robotics. Simulation is the process of developing and designing a virtual model capable of emulating functions of a computer-based physical or theoretical system. In the current era of rapid technology development, many simulators are now available. Although each simulator is for a different purpose, all of them have some common features.

In this thesis, we study the most popular open-source and commercial robotic simulators and compare each other based on some common features which are the cost, the cross-platform capabilities, the 3D virtualization, the fact if they are open-source or not, the ability for multi-agent simulation and the collision detection, in order to find the most ideal and complete simulator. Also, a small simulation experiment is conducted on the simulators in order to compare their performance based on some counters through a specific software.

**Keywords:** simulation, simulator, robotic simulator, evaluation, University of the Aegean

© 2019

GEORGIOS EXIZOGLU

Department of Information and Communication Systems Engineering

UNIVERSITY OF THE AEGEAN

### **ΕΥΧΑΡΙΣΤΙΕΣ**

Αρχικά, θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες στην επιβλέποντα καθηγήτριά μου, κα. Εργίνα Καβαλλιεράτου, για την ανάθεση του θέματος, την εμπιστοσύνη που μου έδειξε να εκπονηθεί αυτή η εργασία και για την άψογη συνεργασία που είχαμε όσο ήμουν μέλος της ρομποτικής ομάδας Aegean Robotics του Πανεπιστημίου Αιγαίου.

Επίσης, θα ήθελα να ευχαριστήσω τους γονείς μου, Γιάννη και Άννα, καθώς και τον αδερφό μου, Θάνο, για την πολύτιμη συμπαράστασή τους που μου προσέφεραν όλο αυτό το διάστημα και για όλες τις θυσίες που έχουν κάνει για να ολοκληρώσω τις σπουδές μου.

## Πίνακας Περιεχομένων

|                                                              |    |
|--------------------------------------------------------------|----|
| Περίληψη                                                     | 3  |
| Abstract                                                     | 4  |
| Πίνακας Περιεχομένων                                         | 6  |
| Πίνακας Εικόνων                                              | 8  |
| Κατάλογος Πινάκων                                            | 9  |
| Πίνακας Συντομογραφιών                                       | 10 |
| Κεφάλαιο 1: Εισαγωγή                                         | 11 |
| 1.1 Αντικείμενο Εργασίας                                     | 11 |
| 1.2 Σκοπιμότητα και Οφέλη                                    | 12 |
| 1.3 Παρόμοιες μελέτες                                        | 13 |
| 1.4 Δομή Εργασίας                                            | 13 |
| Κεφάλαιο 2: Βιβλιογραφική Επισκόπηση Ρομποτικών Προσομοιωτών | 15 |
| 2.1 Εισαγωγή                                                 | 15 |
| 2.2 Ορισμοί                                                  | 18 |
| 2.2.1 Προσομοίωση                                            | 18 |
| 2.2.2 Ρομποτική Προσομοίωση                                  | 18 |
| 2.2.3 Προσομοιωτής Ρομποτικής                                | 19 |
| 2.3 Υφιστάμενοι Ρομποτικοί Προσομοιωτές και Λογισμικά        | 19 |
| 2.3.1 Webots                                                 | 19 |
| 2.3.2 Simbad                                                 | 21 |
| 2.3.3 USARSim                                                | 24 |
| 2.3.4 Microsoft Robotics Developer Studio                    | 26 |
| 2.3.5 Breve                                                  | 27 |
| 2.3.6 FlightGear                                             | 28 |
| 2.3.7 MATLAB                                                 | 30 |
| 2.3.8 Player/Stage                                           | 32 |

|                                                                                                           |    |
|-----------------------------------------------------------------------------------------------------------|----|
| 2.3.9 Gazebo                                                                                              | 33 |
| 2.3.10 UberSim                                                                                            | 34 |
| 2.3.11 Actin                                                                                              | 36 |
| 2.3.12 X-Plane                                                                                            | 37 |
| Κεφάλαιο 3: Εφαρμογές Ρομποτικών Προσομοιωτών                                                             | 39 |
| 3.1 Εισαγωγή                                                                                              | 39 |
| 3.2 Προσομοίωση στον χώρο της Εκπαίδευσης                                                                 | 40 |
| 3.3 Προσομοίωση στον χώρο της Γεωργίας                                                                    | 45 |
| 3.4 Προσομοίωση στον χώρο της Ιατρικής                                                                    | 50 |
| Κεφάλαιο 4: Μελέτη Περίπτωσης Εφαρμογής Ρομποτικού Προσομοιωτή: JdeRobot framework – Program-A-Robot 2019 | 55 |
| 4.1 Εισαγωγή                                                                                              | 55 |
| 4.2 Program-A-Robot 2019                                                                                  | 56 |
| 4.2.1 Περιγραφή                                                                                           | 56 |
| 4.2.2 Πρόκληση Drone Cat Mouse                                                                            | 57 |
| 4.2.3 Συνέντευξη νικητή                                                                                   | 58 |
| Κεφάλαιο 5: Διεξαγωγή πειράματος στους ρομποτικούς προσομοιωτές                                           | 60 |
| 5.1 Εισαγωγή                                                                                              | 60 |
| 5.2 Περιγραφή πειράματος                                                                                  | 60 |
| 5.3 Αποτελέσματα πειράματος                                                                               | 63 |
| 5.4 Συμπεράσματα πειράματος                                                                               | 66 |
| Κεφάλαιο 6: Αξιολόγηση Ρομποτικών Προσομοιωτών και Παρουσίαση Αποτελεσμάτων                               | 68 |
| 6.1 Εισαγωγή                                                                                              | 68 |
| 6.2 Ανάλυση Κριτηρίων                                                                                     | 68 |
| 6.3 Αποτελέσματα και Σχολιασμός                                                                           | 69 |
| Κεφάλαιο 7: Συμπεράσματα και Προτάσεις                                                                    | 71 |
| 7.1 Συμπεράσματα                                                                                          | 71 |
| 7.2 Προτάσεις                                                                                             | 72 |
| Βιβλιογραφία                                                                                              | 74 |

## Πίνακας Εικόνων

|                                                                                                       |    |
|-------------------------------------------------------------------------------------------------------|----|
| Εικόνα 1: Λεπτομερής διαδικασία ανάπτυξης σχεδιασμού χρησιμοποιώντας προσομοιωτή [7] .....            | 16 |
| Εικόνα 2: Βήματα για μια τυπική μελέτη συστήματος με χρήση προσομοίωσης [8] .                         | 17 |
| Εικόνα 3: Λογότυπο Webots .....                                                                       | 19 |
| Εικόνα 4: Κώδικας υλοποίησης απλού feed-forward νευρωνικού δικτύου στον προσομοιωτή Simbad [15] ..... | 23 |
| Εικόνα 5: Αρχιτεκτονική του USARSim [18] .....                                                        | 25 |
| Εικόνα 6: Λογότυπο Microsoft Robotics Developer Studio.....                                           | 26 |
| Εικόνα 7: Λογότυπο FlightGear .....                                                                   | 28 |
| Εικόνα 8: Λογότυπο Simulink του MATLAB.....                                                           | 30 |
| Εικόνα 9: Λογότυπο Gazebo.....                                                                        | 33 |
| Εικόνα 10: Αρχιτεκτονική UberSim [33] .....                                                           | 35 |
| Εικόνα 11: Λογότυπο X-Plane.....                                                                      | 37 |
| Εικόνα 12: Προσομοίωση στον χώρο της Εκπαίδευσης.....                                                 | 40 |
| Εικόνα 13: Προσομοίωση στον χώρο της Γεωργίας.....                                                    | 45 |
| Εικόνα 14: Προσομοίωση στον χώρο της Ιατρικής.....                                                    | 50 |
| Εικόνα 15: Αρχική σελίδα Drone Cat Mouse πρόκλησης.....                                               | 58 |
| Εικόνα 16: Παρακολούθηση συστήματος με την χρήση του SysGauge .....                                   | 61 |
| Εικόνα 17: Μετρήσεις SysGauge για τον προσομοιωτή Webots .....                                        | 64 |
| Εικόνα 18: Μετρήσεις SysGauge για τον προσομοιωτή Microsoft Robotics Developer Studio .....           | 65 |
| Εικόνα 19: Μετρήσεις SysGauge για τον προσομοιωτή MATLAB - Simulink .....                             | 65 |
| Εικόνα 20: Μετρήσεις SysGauge για τον προσομοιωτή X-Plane .....                                       | 65 |

## Κατάλογος Πινάκων

|                                                                                        |    |
|----------------------------------------------------------------------------------------|----|
| Πίνακας 1: Πίνακας μετρήσεων του SysGauge για τους ρομποτικούς προσομοιωτές (1/2)..... | 63 |
| Πίνακας 2: Πίνακας μετρήσεων του SysGauge για τους ρομποτικούς προσομοιωτές (2/2)..... | 64 |
| Πίνακας 3: Αναλυτικός πίνακας σύγκρισης ρομποτικών προσομοιωτών .....                  | 70 |

## Πίνακας Συντομογραφιών

| <i>Συντ/φία</i> | <i>Επεξήγηση</i>                                         |
|-----------------|----------------------------------------------------------|
| <b>3D</b>       | <b>Three-Dimensional</b>                                 |
| <b>API</b>      | <b>Application Programming Interface</b>                 |
| <b>EPFL</b>     | <b>Swiss Federal Institute of Technology in Lausanne</b> |
| <b>FAA</b>      | <b>Federal Aviation Administration</b>                   |
| <b>GIS</b>      | <b>Geographic Information System</b>                     |
| <b>GPL</b>      | <b>GNU General Public Licence</b>                        |
| <b>GPS</b>      | <b>Global Positioning System</b>                         |
| <b>JDK</b>      | <b>Java Development Kit</b>                              |
| <b>ODE</b>      | <b>Open Dynamics Engine</b>                              |

## Κεφάλαιο 1: Εισαγωγή

### 1.1 Αντικείμενο Εργασίας

Ο σχεδιασμός ενός ρομπότ επιφυλάσσει πάντα ορισμένους κινδύνους, είτε λόγω κόστους, είτε αυτοί αφορούν σπατάλη χρόνου. Με την τρομερή εξέλιξη της τεχνολογίας, πλέον δύναται η δυνατότητα σχεδίασης, μοντελοποίησης και ανάπτυξης ενός ρομποτικού συστήματος με την βοήθεια ειδικών λογισμικών, που ονομάζονται ρομποτικοί προσομοιωτές.

Η προσομοίωση έχει αναγνωριστεί ως ένα σημαντικό ερευνητικό εργαλείο από την αρχή του 20<sup>ου</sup> αιώνα [1]. Στην αρχή, η προσομοίωση χρησιμοποιούταν κυρίως σε ακαδημαϊκές έρευνες. Πλέον, η χρήση της προσομοίωσης έχει γίνει ευρέως χρησιμοποιούμενη από πολλούς τομείς, όπως την εκπαίδευση, την πληροφορική, την γεωργία και την υγεία. Είναι ένα πανίσχυρο εργαλείο που χρησιμοποιείται στην ανάλυση, σχεδιασμό και υλοποίηση συστημάτων από προγραμματιστές και κατασκευαστές. Ωστόσο, υπάρχει διαθέσιμος ένας μεγάλος αριθμός προσομοιωτών, κάνοντας έτσι πιο δύσκολη την επιλογή του καταλληλότερου για την επίλυση κάποιου προβλήματος ή τον σχεδιασμό κάποιου συστήματος.

Στην παρούσα εργασία, θα γίνει αναφορά δώδεκα δημοφιλών ρομποτικών προσομοιωτών και λογισμικών, οι οποίοι συγκεκριμένα είναι οι WeBots, Simbad, USARSim, Microsoft Robotics Developer Studio, Breve, FlightGear, MATLAB - Simulink, Gazebo, Player/Stage, UberSim, Actin και X-Plane και ποια είναι τα χαρακτηριστικά τους. Επίσης, θα παρουσιαστούν περιπτώσεις χρήσης της προσομοίωσης από άλλους επιστημονικούς τομείς και πώς επωφελούνται από αυτή τη χρήση. Θα παρουσιαστεί η διεξαγωγή ενός πειράματος στους προσομοιωτές το οποίο αποσκοπεί στην σύγκριση απόδοσης των προσομοιωτών με βάση κάποιους μετρητές μέσω του ειδικού προγράμματος SysGauge. Θα γίνει σύγκριση των ρομποτικών προσομοιωτών με βάση κάποια κριτήρια και τέλος θα παρουσιαστούν τα τελικά αποτελέσματα της σύγκρισης, τα συμπεράσματα που προκύπτουν, καθώς και προτάσεις βελτίωσης για μια μελλοντική παρόμοια έρευνα.

## 1.2 Σκοπιμότητα και Οφέλη

Όσο παράξενο και αν φαίνεται, όλοι οι άνθρωποι έχουν ζήσει μια μορφή προσομοίωσης. Όλοι έχουν υποδυθεί έναν ρόλο, π.χ. κάποιος να είναι γονιός ή γιατρός ή δάσκαλος, ζώντας έτσι σε μια εικονική πραγματικότητα. Έτσι, μπορεί κανείς να κατανοήσει καλύτερα την έννοια της προσομοίωσης. Με αυτόν τον τρόπο μπορεί κανείς να ανακαλύψει και να μάθει πολλά και χρήσιμα πράγματα με πολύ αποτελεσματικό τρόπο.

Η προσομοίωση ως εργαλείο προσφέρει σημαντικά πλεονεκτήματα και οφέλη. Ανοίγει νέους δρόμους προς την επίλυση ενός προβλήματος ή τον σχεδιασμό ενός συστήματος, καθώς προάγει πολύ την δημιουργικότητα. Μπορεί κανείς να ερευνήσει, να σχεδιάσει, να μοντελοποιήσει και να ελέγξει ένα σύστημα ακόμα και αν δεν υπάρχει στην πραγματικότητα. Μπορεί να δει αποτελέσματα από ένα σύστημα χωρίς να χρειάζεται να είναι πλήρως ολοκληρωμένο. Οι πειραματισμοί μπορεί να οδηγούν σε αποτυχία, ή ακόμα και να υπάρχουν καταστροφικά αποτελέσματα, αλλά με την προσομοίωση οι επιπτώσεις μπορεί να είναι ασήμαντες, από θέμα κόστους και χρόνου. Έτσι, με την προσομοίωση μπορεί κανείς να αποφύγει ζημιές και καταστροφές, στο τελικό προϊόν, να αποφύγει αλλαγές που δεν θα είχαν καλό αποτέλεσμα και γενικά να προβλέψει καλύτερα το «τι θα γινόταν αν...». Για παράδειγμα, μπορεί κάποιος να δει, μέσα σε λίγα λεπτά, πώς μια προγραμματισμένη διαδικασία παραγωγής ενός ρομποτικού συστήματος θα εξελισσόταν στο διάστημα του επόμενου μήνα αν στον σχεδιασμό του πραγματοποιούνταν μια αλλαγή σε κάποιο μέρος του.

Ένα από τα κύρια προβλήματα που έρχεται να λύσει η προσομοίωση, εκτός από τη μείωση κόστους και χρόνου που αποτελούν εξίσου σημαντικό πλεονέκτημα, είναι η ερώτηση «τι θα γινόταν αν...». Πολλές φορές, αυτό το ερώτημα προκύπτει από την πολυπλοκότητα των συστημάτων και με αυτό τον τρόπο προκύπτουν πολλά αναπάντητα ερωτήματα που δεν μπορούν να απαντηθούν εύκολα. Με την απόλυτη ακρίβεια σε θέματα γεωμετρίας, την δυναμικότητα που προσφέρεται στα χαρακτηριστικά, καθώς και την τρισδιάστατη μοντελοποίηση ενός αντικειμένου με τέλεια λεπτομέρεια, τα όρια του τι είναι εφικτό και τι όχι επεκτείνονται, δίνοντας έτσι την ευκαιρία στην έρευνα να αντιμετωπίσει σοβαρές προκλήσεις της πραγματικότητας με την δημιουργία πειραματικών εικονικών περιβαλλόντων.

Σκόπιμο, λοιπόν, θεωρείται να απαντηθεί το ερώτημα ποιοι είναι οι ρομποτικοί προσομοιωτές και λογισμικά, ανοιχτού-κώδικα και μη, που «μπορούν να κάνουν σωστά την δουλειά τους».

### 1.3 Παρόμοιες μελέτες

Μία από τις πρώτες επισκοπήσεις ρομποτικού προσομοιωτή ήταν αυτή του N. Koenig και A. Howard. [2]. Στην επισκόπηση αυτή, οι συγγραφείς περιγράφουν το λογισμικό τρισδιάστατης προσομοίωσης Gazebo με λεπτομερή τρόπο, την χρήση του σε διάφορες μελέτες περίπτωσης καθώς και τα χαρακτηριστικά του. Ωστόσο, δεν υπήρξε κάποια σύγκριση κάποιου άλλου λογισμικού με τον Gazebo, κάποια εκτέλεση κώδικα ή τρισδιάστατη μοντελοποίηση.

Σε μια άλλη εργασία, ο Zlajpah [1], παρουσιάζει μια γενική επισκόπηση ρομποτικών προσομοιωτών για ρομποτικά συστήματα. Διάφορα λογισμικά και toolboxes περιγράφονται, αναφέροντας ποια τα χαρακτηριστικά τους και παρέχοντας μικρά παραδείγματα κώδικα. Στην συγκεκριμένη εργασία, δεν γίνεται, επίσης, κάποια σύγκριση μεταξύ των προσομοιωτών σε θέματα απόδοσης ή κάπου αλλού, παρά μόνο περιγραφή τους.

Στην εργασία του N. Michael, J. Fink, και V. Kumar [3], οι συγγραφείς παρουσιάζουν τους διάσημους ανοιχτού κώδικα προσομοιωτές Player, Stage και Gazebo ως ένα πακέτο λογισμικών, γνωστό ως PSG [4], μαζί με άλλες τέσσερις βιβλιοθήκες ρομποτικών λογισμικών. Ούτε, όμως, σε αυτήν την εργασία γίνεται κάποια σύγκριση μεταξύ του PSG με τις άλλες τέσσερις βιβλιοθήκες λογισμικών που παρουσιάστηκαν.

Τέλος, στην εργασία του J. Kramer και M. Scheutz. [5], οι συγγραφείς παρουσιάζουν μια λεπτομερή περιγραφή και σύγκριση εννέα ανοιχτού κώδικα ρομποτικών λογισμικών. Η σύγκριση αποτελείται από ένα σύνολο κριτηρίων που βασίζονται στην διαδικασία ανάπτυξης λογισμικού. Επίσης, περιγράφουν τις αποδόσεις και την ευχρηστία του κάθε εργαλείου.

### 1.4 Δομή Εργασίας

Η διπλωματική εργασία αποτελείται από τα εξής κεφάλαια:

- Κεφάλαιο 2<sup>ο</sup>: Βιβλιογραφική Επισκόπηση Ρομποτικών Προσομοιωτών  
Στο κεφάλαιο αυτό αποσαφηνίζονται όλοι οι απαραίτητοι ορισμοί που χρειάζεται να γνωρίζει ο αναγνώστης της εργασίας, ακόμα και αυτός που δεν έχει κάποια ιδιαίτερη επαφή με το αντικείμενο της ρομποτικής. Επίσης, αναφέρονται και αναλύονται δώδεκα δημοφιλείς open-source και εμπορικοί ρομποτικοί προσομοιωτές.
- Κεφάλαιο 3<sup>ο</sup>: Εφαρμογές Ρομποτικών Προσομοιωτών

Στο κεφάλαιο αυτό περιγράφεται πώς οι ρομποτικοί προσομοιωτές χρησιμοποιούνται από τομείς όπως της εκπαίδευσης, της ιατρικής και της γεωργίας για διάφορους σκοπούς που αναλύονται περισσότερο μέσα στο κεφάλαιο.

- **Κεφάλαιο 4<sup>ο</sup>: Διεξαγωγή πειράματος στους ρομποτικούς προσομοιωτές**  
Στο κεφάλαιο αυτό παρουσιάζεται η διεξαγωγή ενός πειράματος μικρής προσομοίωσης στους προσομοιωτές με σκοπό την σύγκριση απόδοσής τους με βάση κάποιους μετρητές μέσω του λογισμικού SysGauge. Συγκεκριμένα, η προσομοίωση περιλαμβάνει την προσομοίωση τεσσάρων μικρών ρομπότ με δύο ρόδες (two-wheeled robots) σε μικρό εικονικό περιβάλλον (θα είναι επίπεδο με 2-4 εμπόδια στο περιβάλλον, ανάλογα τον προσομοιωτή). Η προσομοίωση θα διαρκεί ένα λεπτό.
- **Κεφάλαιο 5<sup>ο</sup>: Αξιολόγηση Ρομποτικών Προσομοιωτών και Παρουσίαση Αποτελεσμάτων**  
Στο κεφάλαιο αυτό αναλύονται τα κριτήρια που επιλέχτηκαν για να γίνει η αξιολόγηση των ρομποτικών προσομοιωτών, ο λόγος επιλογής τους και η σημαντικότητά τους. Επίσης, παρουσιάζονται τα συνολικά αποτελέσματα της σύγκρισης που πραγματοποιήθηκε στην εργασία αυτή, καθώς και αναλυτικός σχολιασμός τους.
- **Κεφάλαιο 6<sup>ο</sup>: Συμπεράσματα και Προτάσεις**  
Στο κεφάλαιο αυτό καταγράφονται τα τελικά συμπεράσματα που προέκυψαν από την διεξαγωγή της έρευνας και σύγκρισης των ρομποτικών προσομοιωτών, από την μεριά του συγγραφέα, καθώς και προτάσεις για την διεξαγωγή μιας πιο βελτιωμένης παρομοίου χαρακτήρα και αντικειμένου έρευνας.

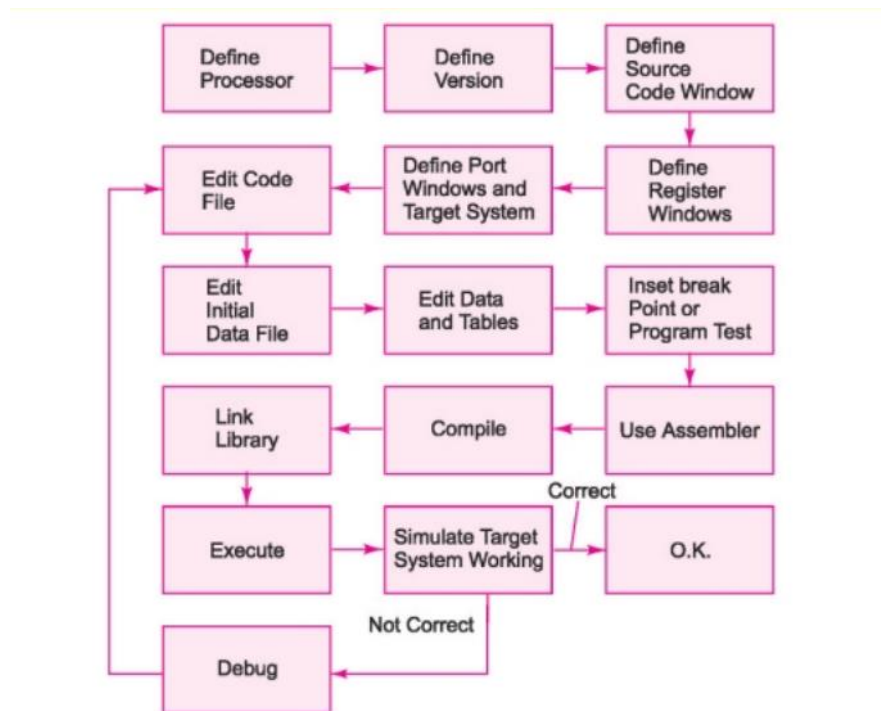
## Κεφάλαιο 2: Βιβλιογραφική Επισκόπηση Ρομποτικών Προσομοιωτών

### 2.1 Εισαγωγή

Με την χρήση σύγχρονων ρομποτικών προσομοιωτών, πλέον έχει γίνει εφικτή η δημιουργία πραγματικών εφαρμογών, χωρίς να είναι απαραίτητη η ύπαρξη του φυσικού μοντέλου. Οι ερευνητές και προγραμματιστές που χρησιμοποιούν το εργαλείο της προσομοίωσης κερδίζουν χρόνο, μπορούν να μειώσουν το κόστος κατασκευής μέσα από πειραματισμούς και μπορούν να προβλέψουν το αποτέλεσμα πριν την υλοποίηση του συστήματος. Με την πολύπλευρη σημασία που έχει αποκτήσει πλέον ο όρος της ρομποτικής προσομοίωσης, η έκταση των εφαρμογών έχει αυξηθεί σε πολλούς και διαφορετικούς τομείς. Το εργαλείο της προσομοίωσης χρησιμοποιείται στην εκπαίδευση, στην ψυχαγωγία και πιο συγκεκριμένα στον χώρο των ταινιών και των βιντεοπαιχνιδιών, στον αθλητισμό για την πρόβλεψη ενός γεγονότος, στην μετεωρολογία, στην ρομποτική και σε πολλούς άλλους τομείς.

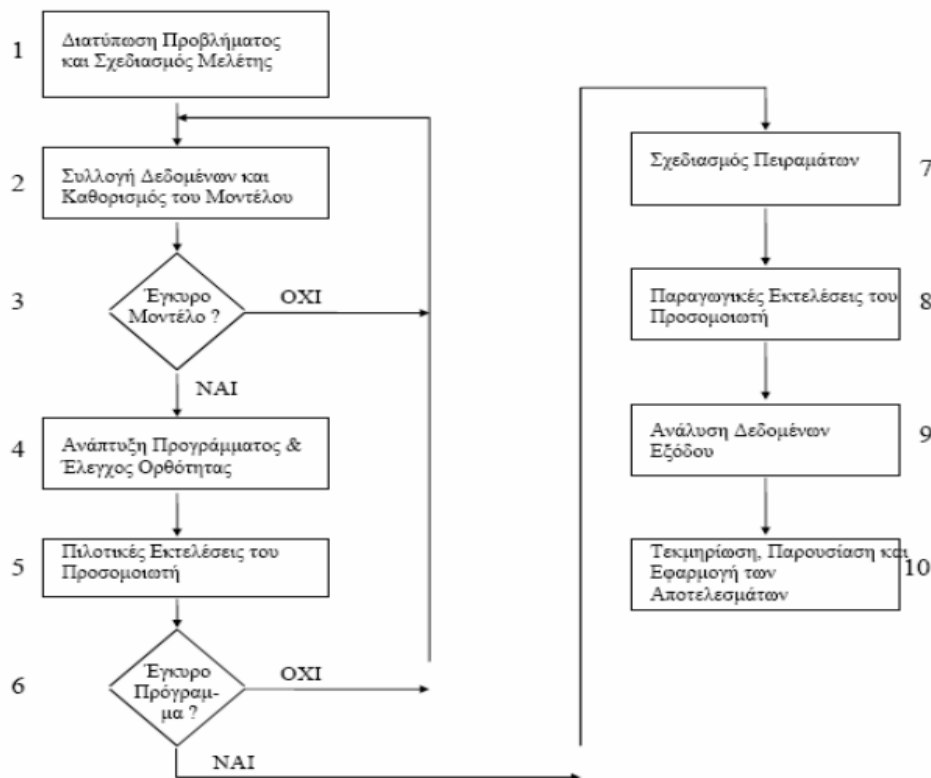
Κάποια από τα χαρακτηριστικά που παρουσιάζουν οι σύγχρονοι εμπορικοί και ανοιχτού κώδικα προσομοιωτές μπορεί να είναι άμεση αναπαράσταση εφαρμογών από ενσωματωμένες βιβλιοθήκες, αναπαράσταση φυσικών φαινομένων από έτοιμες μηχανές, υλοποίηση εφαρμογών και προγραμματισμό συστημάτων με χρήση δημοφιλών γλωσσών προγραμματισμού υψηλού επιπέδου και χρήση δισδιάστατης και τρισδιάστατης γεωμετρίας με τη βοήθεια άλλων λογισμικών. [6]

Στην παρακάτω εικόνα, απεικονίζεται μια λεπτομερή διαδικασία υλοποίησης σχεδιασμού με τη χρήση της προσομοίωσης:



Εικόνα 1: Λεπτομερής διαδικασία ανάπτυξης σχεδιασμού χρησιμοποιώντας προσομοιωτή [7]

Στην παρακάτω εικόνα, φαίνονται τα βήματα για μια τυπική μελέτη ενός συστήματος με χρήση του εργαλείου της προσομοίωσης:



Εικόνα 2: Βήματα για μια τοπική μελέτη συστήματος με χρήση προσομοίωσης [8]

Θα ήταν όμως πολύ ιδανικό το να μπορούν οι προσομοιωτές να λύνουν προβλήματα της πραγματικής ζωής με τόση ευκολία. Παρουσιάζουν κάποιους περιορισμούς που θα πρέπει κανείς να λάβει υπόψιν αν πρόκειται να χρησιμοποιήσει κάποιον. Πρώτα απ' όλα, η φυσική μηχανή, που χρησιμοποιούν σχεδόν όλοι οι προσομοιωτές, αν και είναι καλή στο να προσομοιώνει φυσικούς νόμους και συγκρούσεις αντικειμένων, δεν μπορεί να προσομοιώσει με τέλεια πιστότητα τη φυσικότητα των συνθηκών και τα αντικείμενα που θα υπάρχουν σε έναν προσομοιωμένο κόσμο, καθώς ο πραγματικός κόσμος είναι πολύ περίπλοκος, γεμάτος συνθήκες και μεταβλητές που συνεχώς μεταβάλλονται. Τα αντικείμενα στον πραγματικό κόσμο, μπορεί να αποτελούνται από άλλα μικρότερα εξαρτήματα, ενώ στον προσομοιωμένο κόσμο να είναι ένας ενιαίος κύβος ή σφαίρα. Χρειάζεται προσοχή στην λήψη αποφάσεων για το πού θα πρέπει να προστεθεί περισσότερη λεπτομέρεια για να υπάρχει μεγαλύτερη πιστότητα της προσομοίωσης σε λογικά πλαίσια. Επίσης, στον πραγματικό κόσμο, οι αισθητήρες πολλές φορές επιστρέφουν δεδομένα θορύβου, καθώς ο θόρυβος συναντάται παντού, οι ρόδες μπορεί να γλιστράνε και οι κάμερες βίντεο να επιστρέφουν εικόνες με κακά pixels. Οι ρομποτικοί αλγόριθμοι θα πρέπει να λαμβάνουν υπόψιν όλα τα παραπάνω. Θα πρέπει

να έχουν προγραμματιστεί σωστά για να ανταποκρίνονται στις συνθήκες αυτές. Τέλος, θα πρέπει να λαμβάνεται υπόψιν η διαφορά υπολογιστικής ισχύς μεταξύ του προσομοιωμένου περιβάλλοντος και του πραγματικού ρομπότ. Ένας προσομοιωτής μπορεί να τρέχει σε έναν ισχυρό υπολογιστή με γρήγορο επεξεργαστή ενώ το ρομπότ μπορεί να έχει έναν πιο αργό επεξεργαστή με πιο αργά data busses. Ένα ρομπότ που περιέχει μια προσομοιωμένη κάμερα μπορεί να επιστρέφει 60 frames το δευτερόλεπτο βίντεο ενώ μια πραγματική κάμερα μπορεί να περιοριστεί σε 1-2 frames το δευτερόλεπτο. Κατά τον σχεδιασμό αλγορίθμων, είναι σημαντικό να είναι γνωστά οι περιορισμοί και η απόδοση του τελικού υλικού (hardware) [22].

Στο κεφάλαιο αυτό θα αποφασηγιστούν χρήσιμοι και σημαντικοί ορισμοί που είναι απαραίτητοι για την ανάγνωση της εργασίας και θα παρουσιαστούν δώδεκα διάσημοι ρομποτικοί προσομοιωτές και λογισμικά με τα χαρακτηριστικά τους.

## 2.2 Ορισμοί

### 2.2.1 Προσομοίωση

Η προσομοίωση ορίζεται ως μια κατά προσέγγιση μίμηση μιας λειτουργίας μιας διαδικασίας ή ενός συστήματος μέσα στο χρόνο με τη βοήθεια υπολογιστή. Στην προσομοίωση προαπαιτείται το μοντέλο να έχει υλοποιηθεί. Το μοντέλο είναι μια καλά ορισμένη περιγραφή του υποκείμενου που προσομοιώνεται και αντιπροσωπεύει τα κύρια χαρακτηριστικά του, όπως την συμπεριφορά του, διάφορες λειτουργίες και αφηρημένες ή φυσικές ιδιότητες. [9]

### 2.2.2 Ρομποτική Προσομοίωση

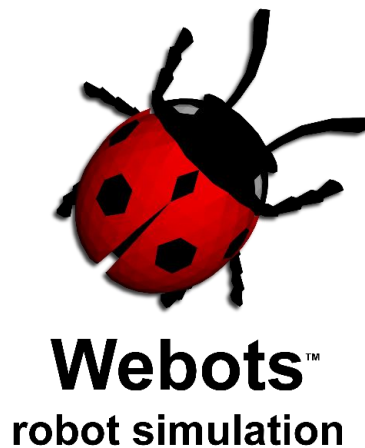
Η ρομποτική προσομοίωση αφορά την εφαρμογή της προσομοίωσης στον κλάδο της ρομποτικής. Μέσω της ρομποτικής προσομοίωσης, είναι εφικτή η δημιουργία ρομποτικών συστημάτων ελέγχου τα οποία έχουν άμεση σύνδεση με τυχαίες σταδιακές ή όχι αλλαγές οι οποίες πραγματοποιούνται μέσα από ένα εύρος πολλών και συνεχών επαναλήψεων. Επίσης, με την ρομποτική προσομοίωση, το κόστος της διαδικασίας δημιουργίας ρομποτικών συστημάτων δεν είναι υψηλό και ο χρόνος υλοποίησης μειώνεται πολύ περισσότερο. [10]

### 2.2.3 Προσομοιωτής Ρομποτικής

Ο προσομοιωτής ρομποτικής χρησιμοποιείται για την υλοποίηση εφαρμογών για ένα φυσικό ρομπότ, χωρίς να εξαρτάται από την φυσική μηχανή, μειώνοντας έτσι κόστος και χρόνο. Στις περισσότερες περιπτώσεις, οι εφαρμογές που υλοποιούνται μεταφέρονται στο ρομπότ χωρίς κάποιες τροποποιήσεις.

## 2.3 Υφιστάμενοι Ρομποτικοί Προσομοιωτές και Λογισμικά

### 2.3.1 Webots



*Εικόνα 3: Λογότυπο Webots*

Ο προσομοιωτής Webots [11] είναι ένας τρισδιάστατος ρομποτικός προσομοιωτής και θεωρείται ένας από τους πιο ανεπτυγμένους. Ανακαλύφθηκε το 1966 για πρώτη φορά και έχει υλοποιηθεί από την Cyberbotics Ltd. και την EPFL. Περιλαμβάνει και χρησιμοποιεί την φυσική μηχανή Open Dynamics Engine, η οποία παρέχει την δυνατότητα εντοπισμού τυχόν συγκρούσεων μεταξύ των μοντέλων και προσομοίωσης των χαρακτηριστικών γνωρισμάτων, σχετικά με τα αντικείμενα που εισέρχονται στην πλατφόρμα [12].

Ο προσομοιωτής Webots μπορεί να υποστηρίξει ένα σύνολο ρομποτικών κατασκευών και να τα προσομοιώσει με άριστο τρόπο. Παρακάτω φαίνονται τα πιο βασικά είδη [13]:

- AIBO ERS7 και ERS210, Sony Corporation
- BIOLOID (dog), Robotis

- Boe-Bot
- DARwIn-OP, Robotis
- E-puck
- Hemisson
- HOAP-2, Fujitsu Limited
- iCub, RobotCub Consortium
- iRobot Create, iRobot
- Katana IPR, Neuronics AG
- Khepera mobile robot I, II, III, K-Team Corporation
- KHR-2HV, KHR-3HV, Kondo
- Koala, K-Team Corporation
- Lego Mindstorms (RCX Rover model)
- Magellan
- Nao V2, V3, Aldebaran Robotics
- MobileRobots Inc Pioneer 2, Pioneer 3-DX, Pioneer 3-AT
- Puma 560, Unimate
- Scout 2
- Shrimp III, BlueBotics SA
- Surveyor SRV-1, Surveyor Corporation
- youBot, KUKA

Οι γλώσσες προγραμματισμού που χρησιμοποιούνται από την πλατφόρμα για την δημιουργία και προσομοίωση των παραπάνω ρομποτικών κατασκευών είναι οι C/C++, Java, URBI, Python, ROS και MATLAB. Όλες οι εκτελέσιμες εντολές και χειρισμοί των συστημάτων γράφονται με τις παραπάνω γλώσσες.

Ο προσομοιωτής Webots χρησιμοποιείται κυρίως για συγκεκριμένες εφαρμογές. Χρησιμοποιείται κυρίως για την υλοποίηση πρότυπων ρομποτικών οχημάτων, για έρευνα στην κίνηση ρομπότ, για πολυρομποτικές προσομοιώσεις, για πειραματικά μοντέλα προσομοίωσης και για διερεύνηση ρομποτικής συμπεριφοράς σε περιβάλλον προσομοίωσης [13]. Μεγάλη εφαρμογή βρίσκει και στην εκπαίδευση όπου δίνει την δυνατότητα σε μαθητές να αρχίσουν μια νέα εμπειρία σχετικά με την ρομποτική, την τρισδιάστατη σχεδίαση, τον προγραμματισμό και τις έννοιες της τεχνητής νοημοσύνης και της υπολογιστικής όρασης.

### 2.3.2 Simbad

Το Simbad [14] είναι ένας ανοιχτού κώδικα τρισδιάστατος Java ρομποτικός προσομοιωτής κυρίως για επιστημονικούς και εκπαιδευτικούς σκοπούς. Χρησιμοποιείται κυρίως από ερευνητές και προγραμματιστές που θέλουν ένα περιβάλλον για να μελετήσουν Τεχνητή Νοημοσύνη, Μηχανική Μάθηση και πιο γενικά αλγόριθμους τεχνητής νοημοσύνης, στα πλαίσια αυτόματων ρομπότ και πρακτόρων. Η απλότητα που διαθέτει η διεπαφή του παραμένει έτσι λόγω της αμεσότητας και της γρήγορης χρήσης που παρέχει για έρευνα και εκπαίδευση.

Στο κύριο πακέτο του Simbad περιλαμβάνονται τρία πακέτα: ο ρομποτικός προσομοιωτής Simbad, η βιβλιοθήκη νευρωνικών δικτύων PicoNode και η βιβλιοθήκη εξελικτικών αλγορίθμων PicoEvo.

Ο ρομποτικός προσομοιωτής Simbad έχει γραφτεί σε γλώσσα Java και το μόνο που απαιτείται για την εκτέλεσή του είναι ένα JDK (version 1.4.2 ή νεότερο) και Sun Java 3d (version 1.3.1). Τα δύο αυτά πακέτα είναι διαθέσιμα για τα λογισμικά Windows, Mac OSX, Linux, IRIX και AIX. Το Simbad φιλοξενείται από την SourceForge. Είναι δωρεάν προς χρήση και τροποποίηση υπό όρους GPL. [15]

Οι κύριες λειτουργίες που παρέχει ο Simbad είναι κυρίως [15]: πολυρομποτική ή μονού ρομπότ προσομοίωση, μονοσκοπικού χρώματος κάμερες, online προσομοίωση, προγραμματισμό Python με jython, επεκτάσιμη διεπαφή του χρήστη και απλοποιημένη μηχανή φυσικής.

Εκτός από Java, με τον Simbad μπορεί κανείς να προγραμματίσει και σε Python. Μια λίστα με παραδείγματα τύπου demo είναι διαθέσιμη στο μενού του Simbad αναπαριστώντας έτσι τα χαρακτηριστικά του και είναι η εξής:

- BaseDemo, demo με κάμερα αισθητήρα, sonars και bumpers. Το ρομπότ περιπλανιέται και σταματάει όταν υπάρξει κάποια σύγκρουση.
- ImagerDemo, δείχνει πώς να γίνει σύλληψη εικόνας, η επεξεργασία της και η προβολή της σε ξεχωριστό παράθυρο.
- AvoidersDemo, δείχνει διάφορα ρομπότ με sonars και bumpers που αποφεύγουν συγκρούσεις.
- BlinkingLampDemo, ένα λαμπάκι αναβοσβήνει σε κάθε ρομπότ όταν πλησιάζει κάποιο εμπόδιο.
- BumpersDemo, demo στο οποίο ρομπότ συγκρούονται.
- DifferentialKinematicDemo, demo κίνησης ρομπότ με δύο ρόδες.
- KheperaDemo, demo του Khepera ρομπότ.
- LightSearchDemo, ρομπότ στο οποίο ψάχνει φως μέσω αισθητήρα φωτός.
- PickCherriesDemo, demo όπου ρομπότ μαζεύει κεράσια.

- PushBallsDemo, δείχνει ρομπότ να σπρώχνει μπάλες.

Τα Νευρωνικά Δίκτυα αποτελούν μια άριστη επιλογή για τον τομέα του ρομποτικού ελέγχου λόγω της ικανότητάς τους να χειρίζονται δεδομένα θορύβου με μικρή υπολογισμότητα. Η βιβλιοθήκη PicoNode παρέχει ένα γενικό framework βασισμένο σε γραφήματα μαζί με δύο υλοποιήσεις: feed-forward και επαναλαμβανόμενα νευρωνικά δίκτυα. Η χρήση του PicoNode δεν είναι περιορισμένη στον ρομποτικό έλεγχο. Έχει υλοποιηθεί έτσι ώστε να γίνεται εύκολη η χρήση του. Στην παρακάτω εικόνα, φαίνεται κώδικας υλοποίησης ενός απλού feed-forward νευρωνικού δικτύου με δύο κρυμμένα αντικείμενα, δύο input κόμβους και έναν output κόμβο:

```

/* STEP 1 : Initializing and building a neural net */

// step 1a : create a network
FeedForwardNeuralNetwork network = new FeedForwardNeuralNetwork(
    new ActivationFunction_logisticSigmoid() );

// step 1b : create some neurons
Neuron in1 = new Neuron( network,
    new ActivationFunction_logisticSigmoid());
Neuron in2 = new Neuron( network,
    new ActivationFunction_logisticSigmoid());
Neuron hidden1 = new Neuron( network,
    new ActivationFunction_logisticSigmoid());
Neuron out1 = new Neuron( network,
    new ActivationFunction_logisticSigmoid());

// step 1c : declare I/O neurons
network.registerInputNeuron( in1 );
network.registerInputNeuron( in2 );
network.registerOutputNeuron( out1 );

// step 1d : create the topology (random weight values)
network.registerArc( new WeightedArc( in1 , hidden1 ,
    Tools.getArcWeightRandomInitValue()));
network.registerArc( new WeightedArc( in2 , hidden1 ,
    Tools.getArcWeightRandomInitValue()));
network.registerArc( new WeightedArc( hidden1 , out1,
    Tools.getArcWeightRandomInitValue()));

// step 1e : initialize the network (perform some integrity
// checks and internal encoding)
network.initNetwork();

/* STEP 2 : using the network (feed-forward signal) */

// step 2a : loading the input values (i.e. sensory inputs)
ArrayList inputValuesList = new ArrayList();
inputValuesList.add(new Double (0.5));
inputValuesList.add(new Double (0.5));

// step 2b : computing the output values (i.e. motor outputs)
network.step( inputValuesList );
System.out.println("Output value : " + out1.getOutputValue());

```

Εικόνα 4: Κώδικας υλοποίησης απλού feed-forward νευρωνικού δικτύου στον προσομοιωτή Simbad [15]

Η βιβλιοθήκη PicoEvo είναι ένας γενικός εξελικτικός αλγόριθμος που ενσωματώνει διάφορα είδη αλγορίθμων όπως γενετικούς αλγόριθμους, εξελικτικές στρατηγικές, γενετικούς προγραμματισμούς με βάση δέντρα ή γραφήματα. Μπορεί να

θεωρεί ένα module που μπορεί να υπάρξει μόνο του και να χρησιμοποιηθεί για οπτικοποίηση συστημάτων και εργασιών που δεν σχετίζονται με την ρομποτική [15].

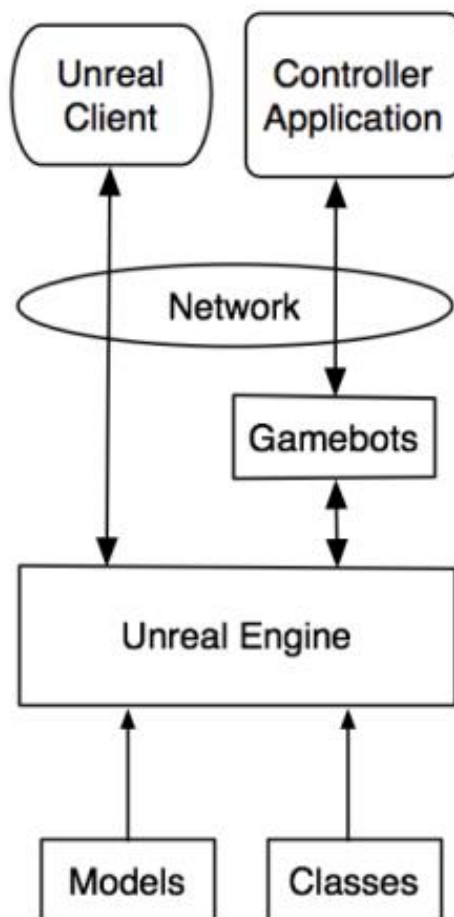
### 2.3.3 USARSim

Ο προσομοιωτής USARSim [16] (Unified System for Automation and Robotics Simulation) υλοποιήθηκε το 2002 στο Carnegie Mellon University. Ο USARSim είναι ένας δωρεάν ανοιχτού κώδικα προσομοιωτής και βασίζεται στην cross platform Unreal Engine 2.0. Το 2005 είχε δοθεί πλήρως στο National Institute of Standards & Technology (NIST) και είχε δημοσιευτεί στο κοινό υπό όρους GPL. Οι φυσικοί νόμοι προσομοιώνονται με την φυσική μηχανή Karma που είναι ενσωματωμένη στην μηχανή Unreal, προσφέροντας έτσι προσομοιώσεις φυσικών νόμων. Ένα σημαντικό πλεονέκτημα χρήσης του Unreal είναι η δυνατότητα υλοποίησης δυνατοτήτων δικτύου. Τα ρομπότ μπορούν να ελεγχθούν με οποιαδήποτε γλώσσα υποστηρίζει TCP sockets [17].

Τα χαρακτηριστικά που προσφέρει ο προσομοιωτής USARSim είναι πολλά και ενδιαφέροντα. Αρχικά, λόγω του ότι χρησιμοποιεί την μηχανή Unreal, που είναι μια από τις πιο σύγχρονες και χρησιμοποιείται ευρέως, η διαδικασία της προσομοίωσης γίνεται με καλή αίσθηση ρεαλιστικότητας και φυσικότητας. Επίσης, όπως αναφέρθηκε και προηγουμένως, είναι ανοιχτού κώδικα και δωρεάν διαθέσιμος, υπό όρους GPL. Επίσης, ο USARSim δίνει πολλές επιλογές ρύθμισης και επέκτασης. Οι χρήστες μπορούν να προσθέσουν καινούργιους αισθητήρες ή να μοντελοποιήσουν καινούργια ρομπότ. Επιπλέον, ο USARSim μπορεί να συνδεθεί με τον Player, ένα middleware λογισμικό που θα συζητηθεί στην συνέχεια της εργασίας, με το οποίο γίνεται έλεγχος πολλών διαφορετικών ρομπότ και με το Mobility Open Architecture Simulation and Tools framework (MOAST), ένα πλήρως λειτουργικό σύστημα ελέγχου. Με το MOAST, οι χρήστες μπορούν και ελέγχουν ρομποτικές πλατφόρμες. Έτσι, μπορούν να υλοποιήσουν πιο ικανά ρομπότ προσθέτοντας εντολές στο λεξιλόγιο κάποιου modular. Να σημειωθεί ότι ο προσομοιωτής USARSim είχε υλοποιηθεί, αρχικά, με την προοπτική ότι θα υλοποιούσε Urban Search and Rescue προσομοιώσεις (ως εκ τούτου και το όνομα), αλλά τελικά έγινε ένας πολυρομποτικός προσομοιωτής γενικού σκοπού. Τέλος, είναι μια ανεξάρτητη πλατφόρμα που μπορεί να εκτελεστεί σε Windows, Linux και MacOS [18].

Όσον αφορά την αρχιτεκτονική του προσομοιωτή USARSim, παρουσιάζει ιδιαίτερο ενδιαφέρον. Όπως αναφέρθηκε και προηγουμένως, ο USARSim χρησιμοποιεί την Unreal Engine, μια εμπορικά διαθέσιμη μηχανή που χρησιμοποιείται για υλοποίηση παιχνιδιών και έχει υλοποιηθεί από την Epic Games [19]. Οι μηχανές παιχνιδιών, ουσιαστικά, είναι συστήματα προσομοίωσης γενικού

σκοπού που μπορούν να χρησιμοποιηθούν στην υλοποίηση πολλαπλών παιχνιδιών με ίδια βάση. Είναι πολύ προσαρμόσιμες και για αυτό μπορούν να χρησιμοποιηθούν στην υλοποίηση ρομποτικών προσομοιωτών. Στην αρχιτεκτονική της Unreal Engine, υπάρχει μια διεπαφή που ονομάζεται Gamebots [20], που επιτρέπει σε μια εξωτερική εφαρμογή να ανταλλάσσει πληροφορίες και προς τις δύο κατευθύνσεις με την μηχανή. Η Unreal Engine υλοποιεί την Unreal Virtual Machine, που είναι πολύ παρόμοια σε λογική με την Java Virtual Machine. Ο κώδικας που γράφεται σε Unrealscript, μια γλώσσα που θυμίζει πολύ C++ και Javascript, μεταγλωττίζεται σε bytecode και εκτελείται από την Unreal Engine. Στην παρακάτω εικόνα, παρουσιάζεται αναλυτικότερα η αρχιτεκτονική του προσομοιωτή USARSim.



Εικόνα 5: Αρχιτεκτονική του USARSim [18]

#### 2.3.4 Microsoft Robotics Developer Studio



*Εικόνα 6: Λογότυπο Microsoft Robotics Developer Studio*

Το Microsoft Robotics Developer Studio (MRDS) κυκλοφόρησε τον Δεκέμβριο του 2006 με κύριο στόχο την παροχή ενός επαγγελματικού λογισμικού για ρομποτικό έλεγχο. Ο χρόνος προσομοίωσης μοιάζει πολύ με αυτόν μιας μηχανής παιχνιδιών, στα πλαίσια των φυσικών νόμων και δυνατοτήτων απεικόνισης. Οι απαιτήσεις του συστήματος είναι σχετικά λίγες. Είναι συμβατό μόνο σε Windows και απαιτείται το Visual Studio για την εκτέλεση. Τα ρομπότ που προσομοιώνονται στο MRDS, προγραμματίζονται σε C# ή VisualBasic.NET.

Στην προσομοίωση, το MRDS, χρησιμοποιεί την τεχνολογία DirectX για την παροχή γραφικών. Οι υπηρεσίες ορίζουν τις δικές τους οντότητες, οι οποίες δημιουργούν τα δικά τους αντικείμενα και τα τοποθετούν στον προσομοιωμένο κόσμο. Οντότητες, γενικά, δεν εννοούνται μόνο τα ρομπότ, αλλά και αντικείμενα του περιβάλλοντος, όπως εμπόδια, τοίχοι και δάπεδα. Κάθε οντότητα παρέχεται με μια συλλογή callback μεθόδων για να ανταποκρίνονται με τα γεγονότα του προσομοιωμένου κόσμου, πχ συγκρούσεις ή frame ενημερώσεις. Επίσης, χρησιμοποιεί την Ageia PhysX φυσική μηχανή, η οποία είναι μια από της υψηλότερης πιστότητας μηχανή που είναι διαθέσιμη, για την προσομοίωση φυσικών νόμων. Ένα από τα σημαντικά χαρακτηριστικά της μηχανής είναι ότι μπορεί να επιταχυνθεί με ειδικό υλικό [22]. Οι οντότητες στον προσομοιωμένο κόσμο προσδιορίζονται σε συνδυασμό με συντελεστές τριβής, μάζας και κέντρου βάρους. Στην προσομοίωση της επαφής λαμβάνεται υπόψιν η δύναμη, η ροπή, η ορμή και η αντίσταση κατά την ενημέρωση frames.

### 2.3.5 Breve

Ο Breve είναι ένας δωρεάν GPL τρισδιάστατος προσομοιωτής που επιτρέπει στους χρήστες να υλοποιούν τρισδιάστατες προσομοιώσεις αποκεντρωμένων συστημάτων (decentralized systems) και τεχνητής νοημοσύνης. Οι χρήστες μπορούν να καθορίζουν την συμπεριφορά πολυπρακτορικών συστημάτων σε τρισδιάστατο χώρο και να παρατηρούν πώς αλληλοεπιδρούν μεταξύ τους και με το περιβάλλον. Διαθέτει εντοπισμό συγκρούσεων και την μηχανή OpenGL, για την κατάλληλη προσομοίωση γραφικών και περιβαλλόντων [24].

Οι προσομοιώσεις στον Breve προσομοιωτή υλοποιούνται χρησιμοποιώντας την αντικειμενοστραφής γλώσσα προγραμματισμού *steve* [24]. Η γλώσσα *steve*, που έχει σχεδιαστεί ειδικά για υλοποιήσεις τρισδιάστατων προσομοιώσεων, περιλαμβάνει “συλλέκτη σκουπιδιών” (garbage collector), υποστήριξη για λίστες και τρισδιάστατων πινάκων, καθώς και ένα σύνολο κλάσεων που συμπεριλαμβάνονται στα χαρακτηριστικά προσομοίωσης της Breve μηχανής. Επίσης, διαθέτει μια επεκτάσιμη αρχιτεκτονική plugin η οποία επιτρέπει στους χρήστες να αλληλεπιδρούν με εξωτερικές βιβλιοθήκες και γλώσσες προγραμματισμού και να αποκτάνε πρόσβαση σε αυτές μέσω της *steve* [24]. Η *steve* γλώσσα είναι μια διαδικαστική ή δομημένη γλώσσα και πολλά από τα χαρακτηριστικά της φαίνονται παρόμοια στους χρήστες με αυτά της γλώσσας C. Μια διαφορά, γενικά, που είναι διακριτή, είναι η σύνταξη των μεθόδων που καλούνται, η οποία σύνταξη μοιάζει περισσότερο με αυτήν της Objective-C. Στην *steve*, κάθε παράμετρος μιας μεθόδου σχετίζεται με μια λέξη-κλειδί. Αυτό σημαίνει ότι κάθε παράμετρος δεν προσδιορίζεται από την σειρά που αναφέρεται στην εντολή, αλλά από τις λέξεις κλειδιά [24].

Για να αλληλοεπιδράσει κανείς με τα χαρακτηριστικά του προσομοιωτή Breve, αυτό μπορεί να γίνει με την συμπεριλαμβανόμενη ιεραρχία κλάσεων. Η υψηλότερη κλάση της ιεραρχίας είναι η κλάση που λέγεται *Object* και θεωρείται ο πρόγονος όλων των αντικειμένων που έχουν δημιουργηθεί στον Breve. Η ιεραρχία χωρίζεται σε *Abstract* κλάσεις και *Real* κλάσεις. Οι *Abstract* κλάσεις ορίζονται ως κλάσεις οι οποίες δεν έχουν κάποια φυσική αναπαράσταση στον προσομοιωμένο κόσμο, ενώ οι *Real* κλάσεις σχετίζονται με συγκεκριμένα φυσικά αντικείμενα [24].

Οι συμπεριφορές των πρακτόρων υλοποιούνται στην *steve*. Κάθε λειτουργία που διαθέτει ένας πράκτορας, μπορεί να υλοποιηθεί σαν ξεχωριστή μέθοδος, η οποία θα καλείται σε συγκεκριμένα γεγονότα. Υπάρχουν διάφοροι τρόποι για να προκληθεί ένα γεγονός στις προσομοιώσεις του Breve. Αρχικά, κάποια γεγονότα καλούνται σε κάθε επανάληψη. Υπάρχουν λειτουργίες οι οποίες εκτελούνται σε κάθε επανάληψη της μηχανής, όπως για παράδειγμα η αλλαγή ταχύτητας ή επιτάχυνσης και ο έλεγχος του περιβάλλοντος για ορισμένα σημεία ή η αλληλεπίδραση με άλλες οντότητες της προσομοίωσης. Επίσης, υπάρχουν γεγονότα που είναι προγραμματισμένα να συμβούν

σε μια συγκεκριμένη στιγμή. Αυτό επιτυγχάνεται με την βοήθεια της μεθόδου `schedule`. Άλλα γεγονότα προκαλούνται από ειδοποιήσεις. Επειδή οι οντότητες μπορεί να εμφανίζονται και να εξαφανίζονται κατά την διάρκεια της προσομοίωσης, είναι πιθανό να είναι δύσκολο για τις οντότητες να γνωρίζουν και να παρακολουθούνται μεταξύ τους. Έτσι, για να απλοποιηθεί η διαδικασία επικοινωνίας με τις οντότητες, ο Breve επιτρέπει στις οντότητες να προβάλλουν ειδοποιήσεις σχετικά με την κατάστασή τους στην οποία βρίσκονται εκείνη την στιγμή χρησιμοποιώντας την μέθοδο `notify`. Αυτές οι ειδοποιήσεις προκαλούν λειτουργίες για όλες τις οντότητες οι οποίες έχουν καταγραφεί ως παρατηρητές χρησιμοποιώντας την μέθοδο `observe`. Επιπλέον, κάποια γεγονότα προκαλούνται από συγκρούσεις. Αυτό μπορεί να επιτευχθεί, γιατί, όπως αναφέρθηκε και προηγουμένως, η μηχανή της προσομοίωσης διαθέτει το χαρακτηριστικό να εντοπίζει μια φυσική σύγκρουση μεταξύ δύο αντικειμένων. Τέλος, υπάρχουν και γεγονότα τα οποία προκαλούνται μέσω της διεπαφής του χρήστη. Οι προσομοιώσεις μπορεί να έχουν παραμετροποιηθεί και υλοποιηθεί που, με βάση τις ανταποκρίσεις και κινήσεις που πραγματοποιεί ο χρήστης, να προκαλεί και τα ανάλογα γεγονότα [24].

Κάθε προσομοίωση στον Breve ελέγχεται από ένα αντικείμενο `controller`, της κλάσης `Control`. Το αντικείμενο `controller` ορίζεται στον πηγαίο κώδικα της προσομοίωσης και είναι το μόνο αντικείμενο που δημιουργείται αυτόματα όταν μια προσομοίωση τρέχει. Άλλα αντικείμενα δημιουργούνται στην πορεία της προσομοίωσης, από το αντικείμενο `controller` στην μέθοδο `init` [24].

### 2.3.6 FlightGear



Εικόνα 7: Λογότυπο FlightGear

Ο προσομοιωτής πτήσεων FlightGear είναι ένα δωρεάν, ανοιχτού κώδικα, ανεξάρτητης πλατφόρμας λογισμικό που έχει υλοποιηθεί από την FlightGear Project

το 1997. Το FlightGear ξεκίνησε ως μια online πρόταση το 1996 από τον David Murr, στον Καναδά. Πρότεινε έναν νέο προσομοιωτή πτήσεων που θα υλοποιούνταν από εθελοντές μέσω του Διαδικτύου, σαν εναλλακτική λύση προς τους εμπορικά διαθέσιμους όπως τον Microsoft Flight Simulator. Το 1997, ο Curt Olson δημιούργησε μια ανεξάρτητης πλατφόρμας κυκλοφορία του FlightGear [25]. Από τότε μέχρι και σήμερα, έχει βελτιωθεί και επεκταθεί πέρα από τις αεροδυναμικές πτήσεις λόγω βελτίωσης γραφικών, προσθήκης καιρικών συνθηκών με ουρανό, αστέρια και ήλιο, ηλεκτρονικών συστημάτων πλοήγησης των αεροπλάνων, αεροδρομίων και δικτύων. Πλέον οι παραμετροποιήσεις των παραπάνω χαρακτηριστικών σε προσομοιώσεις του FlightGear έχουν απλοποιηθεί για διευκόλυνση του χρήστη [25].

Συγκριτικά με άλλους προσομοιωτές πτήσεων, ο FlightGear, λόγω του ότι είναι ανοιχτού κώδικα, επιτρέπει στους χρήστες να τροποποιούν και να βελτιώνουν τα πηγαία αρχεία του προσομοιωτή. Μέσω ενός ευέλικτου framework, ενθαρρύνει τους χρήστες να επεκτείνουν και να βελτιώσουν την ποιότητα της προσομοίωσης [25].

Ο προσομοιωτής FlightGear έχει ένα σύνολο εύχρηστων και ιδιαίτερων χαρακτηριστικών. Πρώτα από όλα, ο προσομοιωτής υποστηρίζει ένα περιβάλλον multiplayer, επιτρέποντας έτσι διάφορους “πιλότους” να βρίσκονται στον ίδιο κόσμο και να μοιράζονται τις εμπειρίες τους. Επίσης, υποστηρίζονται υπηρεσίες στα αεροπλανοφόρα Nimitz, Vinson, San Antonio, Foch και Eisenhower. Τα αεροπλανοφόρα διαθέτουν καταπέλτες, συρματόσχοινα, ανελκυστήρες, TACAN και FLOLS. Το TACAN, είναι συντομογραφία του Tactical air navigation system και είναι σύστημα πλοήγησης που χρησιμοποιείται από στρατιωτικά αεροσκάφη. Το FLOLS, είναι συντομογραφία του Frensel Lens Optical Landing System και είναι μια πιο βελτιωμένη έκδοση του κλασσικού optical landing συστήματος, καθώς έχει ειδικούς φακούς Frensel. Το FLOLS χρησιμοποιείται στο να παρέχει κατευθυντήριες πληροφορίες προσγείωσης στους πιλότους στην τερματική φάση προσγείωσης στα αεροπλανοφόρα. Είναι συστήματα που αποτελούνται κυρίως από χρωματιστά φώτα και πιο συγκεκριμένα, κόκκινα και πράσινα. Επίσης, ο FlightGear διαθέτει τον Atlas, μια εφαρμογή που στοχεύει στην παροχή και εμφάνιση υψηλής ποιότητας διαγραμμάτων στους χρήστες του FlightGear. Ουσιαστικά είναι ένας δισδιάστατος χάρτης ο οποίος ανιχνεύει την κίνηση που κάνει το αεροπλάνο και την εμφανίζει στον χάρτη. Σημειώνεται ότι πρέπει να γίνει λήψη του ξεχωριστά από τον FlightGear και θεωρείται ως extra χαρακτηριστικό για τον προσομοιωτή. Επιπλέον, υποστηρίζεται η ταυτόχρονη μετάδοση από πολλές οθόνες. Δηλαδή, ένας χρήστης μπορεί να έχει μια οθόνη που να δείχνει την μπροστά, μια οθόνη που να δείχνει από τα δεξιά και άλλη μία που να δείχνει από τα αριστερά. Τέλος, ο χρήστης έχει την δυνατότητα να

καταγράψει τις προσομοιώσεις που έχει πραγματοποιήσει και να τις αναλύσει αργότερα ή να την επαναλάβει [26].

### 2.3.7 MATLAB



Εικόνα 8: Λογότυπο Simulink του MATLAB

Το Simulink, ένα πρόσθετο προϊόν στο MATLAB, παρέχει ένα διαδραστικό γραφικό περιβάλλον για τη μοντελοποίηση, την προσομοίωση και την ανάλυση δυναμικών συστημάτων. Επιτρέπει την κατασκευή εικονικών πρωτοτύπων για να την διερεύνηση σχεδιαστικών εννοιών σε οποιοδήποτε επίπεδο λεπτομέρειας με ελάχιστη προσπάθεια. Για τη μοντελοποίηση, το Simulink παρέχει ένα γραφικό περιβάλλον στον χρήστη (GUI) για την κατασκευή μοντέλων ως διαγράμματα μπλοκ. Περιλαμβάνει μια ολοκληρωμένη βιβλιοθήκη προκαθορισμένων μπλοκ προς χρήση για την κατασκευή γραφικών μοντέλων συστημάτων μέσω drag-and-drop λειτουργιών. Ο χρήστης είναι σε θέση να παράγει ένα "up-and-running" μοντέλο που διαφορετικά θα απαιτούσε ώρες για να υλοποιηθεί σε κάποιο εργαστηριακό περιβάλλον [27]. Υποστηρίζει γραμμικά και μη γραμμικά συστήματα, μοντελοποιημένα σε συνεχή χρόνο, δειγματισμένο χρόνο ή συνδυασμό των δύο. Η διαδραστική φύση του Simulink ενθαρρύνει την δοκιμή διάφορων πραγμάτων. Είναι εφικτή η αλλαγή παραμέτρων κατά την υλοποίηση του συστήματος και αμέσως να είναι ορατό το αποτέλεσμα στον χρήστη. Το Simulink είναι ενσωματωμένο στο MATLAB και τα δεδομένα μπορούν εύκολα να μοιραστούν μεταξύ των προγραμμάτων. Επίσης, το Simulink επιτρέπει την ενσωμάτωση MATLAB αλγορίθμων σε μοντέλα αλλά και την εξαγωγή αποτελεσμάτων της προσομοίωσης στο MATLAB για περαιτέρω ανάλυση. Γενικά, το Simulink υποστηρίζει σχεδιασμό συστημάτων, προσομοίωση, αυτόματη παραγωγή κώδικα, έλεγχο και επαλήθευση ενσωματωμένων συστημάτων [27].

Υπάρχουν διάφορα πρόσθετα προϊόντα, που παρέχονται από την MathWorks και από προϊόντα και λογισμικά τρίτων κατασκευαστών που είναι διαθέσιμα για χρήση με το Simulink. Μερικά από αυτά είναι [27]:

- Stateflow, επιτρέπει την υλοποίηση μηχανών καταστάσεων και διαγραμμάτων ροής.
- Simulink Coder, επιτρέπει την δημιουργία πηγαίου κώδικα C για την αυτόματη υλοποίηση συστημάτων σε πραγματικό χρόνο.
- xPC Target, μαζί με συστήματα πραγματικού χρόνου βασισμένα σε αρχιτεκτονική x86 παρέχει ένα περιβάλλον για προσομοίωση και έλεγχο συστημάτων από το Simulink και το Stateflow σε πραγματικό χρόνο.
- Embedded Coder, υποστηρίζει συγκεκριμένα ενσωματωμένα συστήματα.
- HDL Coder, επιτρέπει την αυτόματη δημιουργία συνθετικών VHDL και Verilog.
- SimEvents, παρέχει μια βιβλιοθήκη από γραφικά δομικά μπλοκ για την μοντελοποίηση συστημάτων.

Το Simulink είναι ικανό για την συστηματική επαλήθευση και επικύρωση των μοντέλων και συστημάτων μέσω ειδικού ελέγχου στυλ μοντέλων, ανίχνευσης απαιτήσεων και ανάλυσης μοντέλου. Το Simulink Design Verifier επιτρέπει τον εντοπισμό σχεδιαστικών σφαλμάτων και την παραγωγή σεναρίων δοκιμαστικών περιπτώσεων για κατάλληλο έλεγχο του μοντέλου.

Στο Simulink, ο κλασσικός τρόπος υλοποίησης ενός μοντέλου είναι μέσω των στάνταρ μπλοκ που παρέχονται από την ολοκληρωμένη βιβλιοθήκη του λογισμικού. Ωστόσο, υπάρχει και ένας εναλλακτικός τρόπος και αυτός είναι μέσω των S-functions (system functions). Οι αλγόριθμοι στο MATLAB και στην C μπορούν να υλοποιηθούν και σε S-functions. Το κύριο πλεονέκτημα χρήσης των S-functions είναι ότι οι χρήστες μπορούν και δημιουργούν μπλοκ γενικής χρήσης τα οποία μπορούν να χρησιμοποιηθούν πολλές φορές σε ένα μοντέλο, με διαφορετικές παραμέτρους για κάθε αντικείμενο μπλοκ [28].

Αν και το Simulink του MATLAB θεωρείται από τις πιο διαδεδομένες επιλογές για προσομοίωση ενός μοντέλου, υπάρχουν κάποια μειονεκτήματα που περιορίζουν τους χρήστες που πρόκειται να το χρησιμοποιήσουν. Πρώτα από όλα, το λογισμικό MATLAB δεν είναι δωρεάν και η άδεια του λογισμικού είναι πολύ ακριβή για έναν ερευνητή ή προγραμματιστή που επιθυμεί να πειραματιστεί με το λογισμικό ή να υλοποιήσει ένα από ή γενικού σκοπού μοντέλο. Το MATLAB δεν ενδείκνυται για υλοποίηση μικρών, χαμηλού κόστους μοντέλων αλλά για έργα ερευνητικού σκοπού που συμμετέχουν πολλοί προγραμματιστές και ερευνητές. Επίσης, το cross

compiling ή η μετατροπή του κώδικα MATLAB σε άλλη γλώσσα είναι πολύ δύσκολη ως ακατόρθωτη. Για να επιτευχθεί, προϋποθέτει πολύ καλή γνώση MATLAB, όχι μόνο για την μετατροπή του κώδικα αλλά και για την αντιμετώπιση των οποιοδήποτε σφαλμάτων προκύψουν. Τέλος, είναι δύσκολος ο εντοπισμός αλλαγών με λογισμικά εκδόσεων και revision control systems (όπως CVS, SVN, Git, Bazaar ή Mercurial). Γενικά, είναι πιο εύκολο να εντοπιστούν οι διαφορές όταν το μοντέλο αποθηκεύεται σε .mdl φορμάτ (επειδή είναι δομικού κειμένου αρχείο), παρά όταν αποθηκεύεται σε φορμάτ .slx (επειδή είναι συμπιεσμένο με περισσότερες πληροφορίες σχετικά με το μοντέλο).

### 2.3.8 Player/Stage

Η πλατφόρμα Player Project [4] είναι ανοιχτού κώδικα με GPL άδεια, η οποία, όταν δημιουργήθηκε το 1999 στο University of Southern California, αποτελούνταν από τρία μέρη. Έναν δικτυακού τύπου ρομποτικό εξυπηρετητή, που είναι ο Player, ένα δισδιάστατο προσομοιωτή για πολλαπλά ρομπότ, αισθητήρες και αντικείμενα, τον Stage και έναν τρισδιάστατο προσομοιωτή για πολλαπλά ρομπότ, αισθητήρες και περιβάλλοντα, τον Gazebo. Στο σημείο αυτό, θα γίνει περιγραφή του Player και Stage και στην επόμενη υποενότητα θα αναλυθεί περισσότερο ο προσομοιωτής Gazebo καθώς, από το 2011, ο Gazebo είναι ανεξάρτητο λογισμικό και υποστηρίζεται από τον Willow Garage.

Όσον αφορά το λογισμικό Player, ουσιαστικά παρέχει μια απλή διεπαφή στους αισθητήρες του ρομπότ μέσω δικτύου. Δραστηριοποιείται κυρίως σε συστήματα που λειτουργούν μέσω Portable Operating System Interface for UNIX (POSIX), όπως τα Windows, Linux, MacOS και Solaris [29]. Επίσης, διαθέτει πλούσιες βιβλιοθήκες δεδομένων και πληροφοριών που παρουσιάζουν γλώσσες προγραμματισμού όπως είναι η C, C++, Tcl, Python, Java, Ruby και Common LISP και παρέχει την δυνατότητα υποστήριξης σε interfacing. Γενικά όμως, επειδή το εξωτερικό interface του Player είναι ουσιαστικά ένα TCP socket, τα προγράμματα των πελατών (client) μπορούν να γραφτούν σε οποιαδήποτε προγραμματιστική γλώσσα παρέχει υποστήριξη socket και σχεδόν όλες παρέχουν [29].

Για τον Stage, θα πρέπει να αναφερθεί ότι ουσιαστικά είναι ένας δισδιάστατος ρομποτικός προσομοιωτής, ο οποίος έχει την δυνατότητα να αναπτύσσει το περιβάλλον και τα μέρη του κόσμου της προσομοίωσης για πολυάριθμα ρομποτικά μοντέλα [29]. Οι κύριοι σκοποί του Stage είναι δύο: πρώτον, να γίνεται εφικτή η ταχεία ανάπτυξη ρομποτικών μοντέλων και αισθητήρων τα οποία στη συνέχεια θα υλοποιηθούν στην πραγματικότητα και δεύτερον να γίνονται πειράματα με χρήση των ρομποτικών συστημάτων. Ο Stage σχεδιάστηκε για να υποστηρίξει την διεξαγωγή

πειραμάτων σε πολλαπλά ρομποτικά συστήματα. Ο προγραμματισμός και ο πειραματισμός πολλαπλών ρομποτικών συστημάτων αυξάνει τα πλεονεκτήματα της προσομοίωσης, καθώς είναι αρκετά δύσκολο και ανέφικτο να γίνει κάτι παρόμοιο στην πραγματικότητα λόγω μεγάλου κόστους και κινδύνου σπατάλης χρόνου, αν δεν υπάρξει επιτυχία [29]. Γενικά, ο Stage διαθέτει σχετικά μέτρια πιστότητα. Ωστόσο, το γεγονός αυτό κάνει τον Stage να είναι πιο κατάλληλος για την προσομοίωση ρομποτικών controller και αισθητήρων που θα πρέπει να τρέξουν σε αληθινά ρομπότ. Η απλότητα χρήσης του καθώς και η χρήση αλγορίθμων που λειτουργούν ανεξάρτητα από το πλήθος των ρομποτικών συστημάτων που προσομοιώνονται ευνοούν ακόμα περισσότερο την προσομοίωση απλών και πολλών ρομπότ. Τέλος, ο Stage έχει την δυνατότητα άμεσης σύνδεσης και αλληλεξάρτησης με τα interfaces του Player Project [29].

### 2.3.9 Gazebo



*Εικόνα 9: Λογότυπο Gazebo*

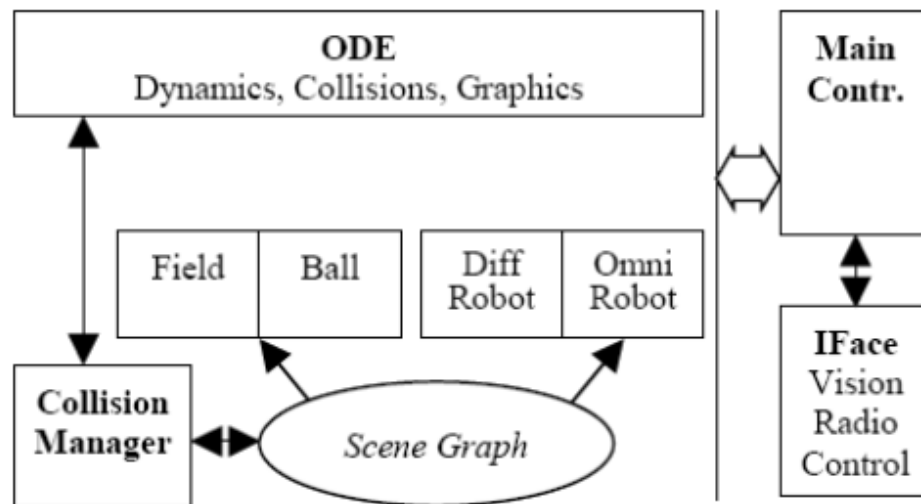
Ο Gazebo [30] είναι ένας δυναμικός τρισδιάστατος προσομοιωτής με την ικανότητα να προσομοιώνει με ακρίβεια και αποτελεσματικότητα πολλαπλά ρομπότ και αισθητήρες σε πολύπλοκα εσωτερικά και εξωτερικά περιβάλλοντα. Παρόλο που είναι αρκετά παρόμοιος με μηχανές παιχνιδιών, ο Gazebo προσφέρει προσομοίωση φυσικών νόμων σε πολύ υψηλό βαθμό πιστότητας, αισθητήρων και διεπαφών τόσο για χρήστες όσο και για προγράμματα [31]. Είναι ανοιχτού κώδικα προσομοιωτής και ήταν μέρος του Player Project από το 2004 μέχρι το 2011. Από το 2011, ο Gazebo είναι ανεξάρτητο λογισμικό και υποστηρίζεται από τον Willow Garage. Περιλαμβάνει την φυσική μηχανή Open Dynamics Engine (ODE) και την βιβλιοθήκη OpenGL [31].

Ο προσομοιωτής Gazebo παρουσιάζει κάποια σημαντικά χαρακτηριστικά και πλεονεκτήματα. Αρχικά, κάποιες από τις βασικές χρήσεις του προσομοιωτή είναι κυρίως για δοκιμές αλγόριθμων ρομποτικής, σχεδιασμό και μοντελισμό ρομπότ και έλεγχο πραγματικών σεναρίων μέσω ρομπότ και αισθητήρων, σχεδιασμένα από τον ίδιο τον προσομοιωτή [31]. Γενικά, ο Gazebo περιέχει πολλαπλές φυσικές μηχανές, μια ολοκληρωμένη βιβλιοθήκη έτοιμων ρομποτικών μοντέλων και περιβάλλοντων, ένα σύνολο αισθητήρων και γραφικές, εύκολες προς τον χρήστη προγραμματιστικές διεπαφές που διευκολύνουν τον προγραμματιστή και ερευνητή στην χρήση του προσομοιωτή. Κάτι που παρουσιάζει μεγάλο ενδιαφέρον και θεωρείται τεράστιο συν του προσομοιωτή Gazebo είναι το γεγονός ότι μπορεί να αναβαθμιστεί με την χρήση και ενσωμάτωση του ROS (Robot Operating System) [31].

Με την ενσωμάτωση του ROS στον Gazebo, ο Gazebo αναβαθμίζεται και αποκτάει επιπλέον χαρακτηριστικά. Αυτό επιτυγχάνεται μέσω ενός συνόλου πακέτων ROS που ονομάζονται `gazebo_ros_pkgs`. Παρέχουν κατάλληλα wrappers και διεπαφές για την ανάλογη προσομοίωση ρομπότ με χρήση ROS μηνυμάτων, υπηρεσιών και δυναμικής αναμόρφωσης. Είναι εφικτή η υλοποίηση με `catkin`, ένα επίσημο build system του ROS και διάδοχος του `roscpp`, που είναι επίσης επίσημο build system του ROS. Μειώνεται ο διπλότυπος κώδικας καθώς και κώδικας από προηγούμενες εκδόσεις ROS και Gazebo. Τέλος, δίνεται καλύτερη υποστήριξη για controllers με την χρήση του `ros_control`.

### 2.3.10 UberSim

Ο UberSim [32] είναι ένας ανοιχτού κώδικα, με GPL άδεια, ρομποτικός προσομοιωτής, με υψηλή πιστότητα και ικανός για προσομοίωση πολλαπλών ρομποτικών συστημάτων. Δημιουργήθηκε το 2000 στο πανεπιστήμιο Carnegie Mellon, με έμφαση στην υλοποίηση μικρών ρομπότ για προσομοίωση ποδοσφαίρου με ρομπότ. Η αρχική εστίαση του προσομοιωτή ήταν οι ομάδες CMDragons RoboCup, αν και ο τελικός στόχος ήταν να αναπτυχθεί ένας προσομοιωτής για πολλούς τύπους και μεγέθη ρομποτικών πλατφορμών [17].



Εικόνα 10: Αρχιτεκτονική UberSim [33]

Στην παραπάνω εικόνα, φαίνεται η αρχιτεκτονική δομή του προσομοιωτή UberSim με τα στοιχεία από τα οποία αποτελείται. Γενικά, ο UberSim είναι κατασκευασμένος γύρω από την βιβλιοθήκη Open Dynamics Engine (ODE), για να είναι εφικτή η παροχή γραφικών, φυσικών νόμων και παραμετροποιημένων αντικειμένων για τον ορισμό του κάθε αντικειμένου στον κόσμο της προσομοίωσης [33]. Το στοιχείο Scene Graph αποτελεί την πρωταρχική δομή δεδομένων για την συλλογή όλων των σχετικών ODE πληροφοριών και όλων των πληροφοριών της προσομοίωσης. Ο διαχειριστής συγκρούσεων (collision manager) παρέχει έναν αποδοτικό εντοπισμό συγκρούσεων και χρησιμοποιώντας τα χαμηλού επιπέδου primitives της ODE ανανεώνει το Scene Graph με σκοπό την παραγωγή κατάλληλης ένδειξης για σύγκρουση. Ο Main Controller είναι ο κύριος αγωγός τους συστήματος του προσομοιωτή. Ουσιαστικά, επιβλέπει και διατηρεί τα interfaces επικοινωνίας του ρομποτικού λογισμικού, παράγει την προσομοίωση και τις ανάλογες ενδείξεις συγκρούσεων την κατάλληλη στιγμή και πραγματοποιεί κατάλληλη διαμόρφωση του συστήματος όταν χρειάζεται [33].

Γενικά, σαν προσομοιωτής, ο UberSim έχει πολλές δυνατότητες λόγω της υψηλής του πιστότητας και της χρήσης της ODE για ταχεία υλοποίηση μικρών ρομπότ με κεντρική όραση (vision centric). Επίσης, έχει χρησιμοποιηθεί και σε διεξαγωγή ποδοσφαίρου με ρομπότ. Ωστόσο, σημαντική σημείωση αποτελεί το γεγονός ότι από το 2007 έχει σταματήσει οριστικά η βελτίωση και συντήρηση του προσομοιωτή [17].

### 2.3.11 Actin

Το Actin [34] είναι ένα εμπορικό λογισμικό υψηλής πιστότητας και βασισμένο στην C++ με σκοπό τον έλεγχο και την προσομοίωση ρομπότ. Μπορεί να προσομοιώνει δυναμικά τρισδιάστατο περιβάλλον και η χρήση του προορίζεται κυρίως για την μοντελοποίηση ρομποτικών αρθρώσεων μέσω αλγορίθμων πραγματικού χρόνου για την επίτευξη συγκεκριμένης κίνησης. Επίσης, με το Actin, είναι εφικτή η πραγματοποίηση Κινηματικής και Δυναμικής Προσομοίωσης, προσομοίωση Monte Carlo και δυναμική προσομοίωση βασισμένη σε φυσικούς νόμους με τεχνικές βελτιστοποίησης παραμέτρων [35]. Χρησιμοποιείται σε τομείς όπως της βιομηχανίας, της ιατρικής περίθαλψης, της γεωργίας και του στρατού [35].

Ένα από τα σημαντικά πλεονεκτήματα του Actin είναι το γεγονός ότι οι χρήστες μπορούν να χρησιμοποιήσουν το Actin και για τον σχεδιασμό ρομποτικών κινήσεων και ελέγχου του ρομπότ αλλά και για την προσομοίωση των ίδιων των ρομπότ. Ουσιαστικά, πραγματοποιείται ολοκληρωμένη προσομοίωση και των ρομπότ αλλά και των κινήσεων που υλοποίησαν οι προγραμματιστές. Επίσης, παρέχει υποστήριξη σε μια πληθώρα λειτουργικών συστημάτων και πιο συγκεκριμένα σε Windows, Linux, OS X, VxWorks, RT Linux, RTOS32 και RTX64 καθώς και σε ενσωματωμένα συστήματα. Πολύ χρήσιμο αποτελεί το γεγονός ότι ο χρήστης μπορεί να εισάγει ένα μοντέλο μορφής CAD στο πρόγραμμα και να δημιουργήσει έτσι αυτόματα ένα σύστημα ελέγχου για να μπορεί να πειραματιστεί διαδραστικά με το προσομοιωμένο ρομπότ. Κατά την προσομοίωση, μπορεί να πραγματοποιηθεί ευθεία κινηματική ανάλυση και αντίστροφη κινηματική ανάλυση πολλών βαθμών ελευθερίας και πιο συγκεκριμένα περισσότερων από 100 βαθμούς, ρομποτικών συστημάτων σε πραγματικό χρόνο. Μπορεί να υπολογίσει τις συντεταγμένες μεταξύ των ρομπότ αλλά και άλλων πρόσθετων προσομοιωμένων ρομπότ και αντικειμένων. Το Actin, εκτός από ρομποτικά συστήματα, μπορεί να προσομοιώσει και διάφορα είδη αισθητήρων, όπως αισθητήρες εικόνας, ήχου και εντοπισμού φωτός. Τα στοιχεία της προσομοίωσης διαμορφώνονται με την χρήση XML. Υπάρχουν κατάλληλα sockets για TCP/IP και UDP/IP επικοινωνία.

Γενικά, το λογισμικό Actin αποτελεί ένα προχωρημένου επιπέδου toolkit για την προσομοίωση ρομποτικών συστημάτων και η χρήση του θα πρέπει να γίνεται από προγραμματιστές που έχουν εκπαιδευτεί από ειδικούς εκπαιδευτές μέσω κατάλληλων σεμιναρίων. Η ίδια εταιρεία δημιουργίας του Actin, η Energid, παρέχει πλήρη εκπαίδευση επί πληρωμή σε προγραμματιστές και ερευνητές για να είναι έτοιμοι να χρησιμοποιήσουν το Actin.

### 2.3.12 X-Plane



*Εικόνα 11: Λογότυπο X-Plane*

Ο προσομοιωτής πτήσεων X-Plane πιστοποίησης FAA υλοποιήθηκε από την Laminar Research και είναι ένα ευέλικτο εμπορικό λογισμικό που προορίζεται για σχεδίαση αεροσκαφών, εκπαίδευση πτήσεων, έλεγχο πτήσεων και μοντελοποίηση αποτυχιών. Υποστηρίζει τα λειτουργικά συστήματα Windows, Linux και OS X αλλά είναι διαθέσιμο και για κινητές συσκευές Android και iOS. Περιλαμβάνει έναν τεράστιο αριθμό εμπορικών και στρατιωτικών αεροσκαφών καθώς και πολλά σενάρια και περιβάλλοντα που είναι εμπνευσμένα από την πραγματικότητα. Το λογισμικό, γενικά, μπορεί να χρησιμοποιηθεί από χρήστες, προγραμματιστές και ερευνητές είτε για λόγους ψυχαγωγίας και testing καθώς διαθέτει προεγκατεστημένα μοντέλα αεροσκαφών και σεναρίων, είτε για ερευνητικούς λόγους για να προγραμματιστεί και να μοντελοποιηθεί κάποιο καινούργιο αεροσκάφος ή για να αναλυθεί η πορεία κάποιου σεναρίου με αεροσκάφος σε ένα περιβάλλον για τυχόν αποτυχίες.

Το X-Plane διατίθεται σε δύο εκδόσεις. Μια η οποία προορίζεται για προσωπική και μη εμπορική χρήση, την Global και μια που είναι για επαγγελματική χρήση, την Professional. Και οι δύο εκδόσεις είναι επί πληρωμή αλλά σαφώς η Professional είναι ακριβότερη και αυτό συμβαίνει γιατί παρέχει δυνατότητα εμπορικής χρήσης, έλεγχο πιστοποίησης FFA και υπηρεσία GPS σε πραγματικό χρόνο.

Επίσης, το X-Plane, εκτός από τους απλούς χρήστες, χρησιμοποιείται και από αεροπορικές δυνάμεις, από κατασκευαστές αεροσκαφών καθώς και από διαστημικούς οργανισμούς είτε για εκπαιδευτικούς λόγους είτε για την προσομοίωση μιας

σχεδίασης αεροσκάφους ή ελέγχου πτήσης. Το λογισμικό διαθέτει μια τεράστια πληθώρα παραμετροποιήσεων και επιλογών για προσομοίωση και σχεδίαση αεροσκαφών και σεναρίων, φέρνοντας έτσι πολύ καλά την εικονικότητα με την πραγματικότητα. Η χρήση του προτείνεται με joystick και πληκτρολόγιο καθώς οι επιλογές και οι χειρισμοί είναι πάρα πολλοί.

## Κεφάλαιο 3: Εφαρμογές Ρομποτικών Προσομοιωτών

### 3.1 Εισαγωγή

Ο κλάδος της ρομποτικής βασίζει τον λόγο ύπαρξής του στην διαδικασία της προσομοίωσης. Πιο συγκεκριμένα, μέσω των ρομποτικών προσομοιωτών ο άνθρωπος έχει καταφέρει να εκτελέσει χρονοβόρες και ακριβές διαδικασίες μέσα σε σύντομο χρονικό διάστημα και με λιγότερο κόστος. Ειδικότερα, οι ρομποτικοί προσομοιωτές δύνανται να δοκιμάσουν διάφορες μεθοδολογίες σε μεταβαλλόμενα, δυναμικά περιβάλλοντα και παράλληλα να κάνουν ευκολότερη τη συλλογή αναδραστικών δεδομένων, ενώ ταυτόχρονα επιτρέπουν την ανάπτυξη των ρομποτικών συστημάτων ελέγχου.

Ρομποτικούς προσομοιωτές μπορούμε να εντοπίζουμε σε διάφορους κλάδους της ανθρώπινης καθημερινότητας, όπως για παράδειγμα στην ιατρική, την εκπαίδευση, την γεωργία κλπ. Μέσω των προσομοιωτών, ο οποιοσδήποτε μπορεί να οργανώσει κάθε βήμα της διαδικασίας που θέλει να επιτελέσει ο προσομοιωτής και στη συνέχεια απλά να ελέγχει την υλοποίηση της διαδικασίας αυτής. Ωστόσο, μέσω ενός προσομοιωτή μπορεί να γίνει κι ένα είδος πρόβλεψης για το μέλλον, γιατί επιτρέπει τη δημιουργία ενός εικονικού περιβάλλοντος, συλλέγοντας απαραίτητα δεδομένα και εξάγοντας ανάλογα συμπεράσματα, για το ποιο θα ήταν το αποτέλεσμα μιας συγκεκριμένης επιλογής. Τέλος, πολύ σημαντική είναι και η δυνατότητα συνεργασίας των ρομποτικών προσομοιωτών από διάφορους κλάδους, επιτρέποντας έτσι την διεπιστημονική προσέγγιση για την αντιμετώπιση ενός προβλήματος.

Βέβαια, ένα ρομποτικός προσομοιωτής μπορεί να κρύβει και κινδύνους κατά τη χρήση του. Πρώτο πρόβλημα αποτελεί η ανάπτυξη του κατάλληλου λογισμικού από τον προσομοιωτή, καθώς υπάρχει πλήθος εφαρμογών, οι οποίες πιθανόν να συμβαδίζουν με τη λειτουργία του λογισμικού. Από την άλλη όμως, υπάρχουν και λογισμικά τα οποία αδυνατούν να υλοποιηθούν λόγω της έλλειψης των αντίστοιχων προδιαγραφών. Το πρόβλημα δεν γίνεται εντονότερο με την συνεχή αύξηση περιβαλλόντων προσομοίωσης, η οποία αυξάνει την επεξεργαστική ισχύ αλλά και το εύρος των εφαρμογών που απαιτούνται για την υλοποίηση των προγραμμάτων. Παράλληλα, υπάρχουν και αρκετοί περιορισμοί για τη λειτουργία των ρομποτικών προσομοιωτών καθώς, ένας προγραμματιστής πρέπει να αφιερώσει αρκετό χρόνο δουλεύοντας πάνω σ έναν προσομοιωτή, διότι, σε αντίθεση με τον εικονικό κόσμο, ο

πραγματικός κόσμος είναι απρόβλεπτος και τα δεδομένα συλλογής αμέτρητα. Επιπλέον, πολλά δεδομένα του φυσικού κόσμου είναι αρκετά δύσκολο να μοντελοποιηθούν ή ακόμα και να αλληλοεπιδράσουν με τα δεδομένα του εικονικού κόσμου ενός προσομοιωτή.

Αν θα έπρεπε όμως να διεξαχθεί ένα γενικό πόρισμα για την σημασία ενός ρομποτικού προσομοιωτή, θα μπορούσαμε να πούμε πως αυτός όχι μόνο είναι σημαντικός για την ανθρώπινη ζωή, αλλά και απαραίτητος πλέον καθώς διανύουμε μια εποχή έκρηξης τεχνολογικών επιτευγμάτων τα οποία ο άνθρωπος από μόνος του αδυνατεί να παρακολουθήσει. Όλος αυτός ο καταγισμός πληροφοριών αναγκάζει την ανθρωπότητα να βρίσκεται σε μια διαρκή αναζήτηση τρόπων επεξεργασίας τους, ώστε να μπορεί να τις διαχειριστεί. Θα πρέπει λοιπόν να αποδεχτούμε πως η τεχνολογία είναι εδώ για να μας βοηθήσει και όχι για να μας κάνει τη ζωή δυσκολότερη. Αυτό όμως δεν σημαίνει πως πρέπει να γίνεται αλόγιστη χρήση των τεχνολογικών μέσων, ούτε να παραβιάζονται νομικές αλλά και ηθικές αρχές και αξίες στον βωμό της εξέλιξης. Μπορούμε να αξιοποιήσουμε οποιοδήποτε τεχνολογικό επίτευγμα αρκεί να σεβόμαστε τον συνάνθρωπο αλλά και την κοινωνία ολόκληρη.

### 3.2 Προσομοίωση στον χώρο της Εκπαίδευσης



Εικόνα 12: Προσομοίωση στον χώρο της Εκπαίδευσης

Τα τελευταία χρόνια όλο και περισσότερα μοντέλα ρομποτικής προσομοίωσης έχουν εισβάλλει στον χώρο της εκπαίδευσης. Ακόμα και στην χώρα μας πανεπιστήμια, τεχνολογικά ιδρύματα ακόμα και σχολεία δειλά δειλά εξοικειώνονται

με τα συγκεκριμένα συστήματα κάνοντάς τα κομμάτι της μάθησης. Αλλά σε πάρα πολλές χώρες παγκοσμίως δεν μπορεί να νοηθεί η εκπαίδευση χωρίς τη χρήση οποιουδήποτε τεχνολογικού μέσου.

Ο ρόλος ενός προσομοιωτή στην εκπαίδευση είναι πολύπλευρος. Αρχικά, αποτελεί πολύτιμο εργαλείο στα χέρια των εκπαιδευτικών, οι οποίοι μπορούν να οργανώσουν οποιαδήποτε διάλεξη με τον τρόπο που οι ίδιοι επιθυμούν και το εκάστοτε μοντέλο να προχωρά κάθε φορά στην παρουσίασή της. Ταυτόχρονα, ένας προσομοιωτής μπορεί να εμπλουτίσει την εκπαιδευτική διδασκαλία πραγματοποιώντας πειράματα είτε σε εικονικό είτε σε πραγματικό περιβάλλον. Μια ακόμη σημαντική λειτουργία των ρομποτικών συστημάτων είναι η δυνατότητα διάδρασης: οι μαθητές συμμετέχουν ενεργά στο μάθημα, μπορούν και οι ίδιοι να δώσουν εντολές στα μοντέλα ρομποτικής, να αναλάβουν εργασίες που θα υλοποιηθούν με την βοήθεια των μοντέλων αυτών κ.ο.κ Ταυτόχρονα με τη συγκεκριμένη τακτική θα προωθηθεί και η συνεργασία μεταξύ όχι μόνο των μαθητών, αλλά και μεταξύ μαθητών και καθηγητών.

Στα πλαίσια αυτά αναπτύχθηκε η εκπαιδευτική ρομποτική. Ως εκπαιδευτική ρομποτική ορίζουμε την καινοτόμα μαθησιακή τεχνολογία η οποία συνδυάζει βασικά στοιχεία επιστημών (ιστορία, μαθηματικά, φυσικές επιστήμες κλπ.) με νέες τεχνολογίες (π.χ. λογισμικά υπολογιστών) και την ανθρώπινη συμπεριφορά [38]. Η ενασχόληση των μαθητών με την ρομποτική πραγματοποιείται μέσω τριών δραστηριοτήτων κατασκευαστικού και προγραμματιστικού χαρακτήρα.

Για την πραγματοποίηση αυτών των δραστηριοτήτων απαιτούνται τα δομικά υλικά (Mindstorms) [39]. Πρώτα όμως οι μαθητές καταγράφουν την πορεία που θέλει να ακολουθήσει το μοντέλο τους δίνοντάς του συγκεκριμένες εντολές και στη συνέχεια με τη χρήση των Lego Mindstorms ορίζουν τις δραστηριότητες της συγκεκριμένης κατασκευής. Στόχος αυτής της διαδικασίας είναι η ολοκλήρωση ενός έργου και κατ' επέκταση η επίλυση ενός προβλήματος. Τα Lego Mindstorms αποτελεί ένα είδος λογισμικού το οποίο βασίζεται στην γλώσσα προγραμματισμού LOGO του Papert και δημιουργήθηκε το 1996. Στη συνέχεια εμφανίστηκε το λογισμικό LEGO Mindstorms Education NXT, το οποίο ανήκει στην κατηγορία λογισμικών τρίτης γενιάς. Το συγκεκριμένο λογισμικό βασίζεται στη χρήση εικονιδίων τα οποία χρησιμοποιούνται παγκοσμίως από επιστήμονες, μηχανικούς και άλλους επαγγελματίες οι οποίοι επιθυμούν να σχεδιάσουν, να δοκιμάσουν και να ελέγξουν αντίστοιχα συστήματα προσομοιωτών.

Η μεθοδολογία του λογισμικού αυτού που ακολουθείται από τους μαθητές είναι η εξής:

Βήμα 1<sup>ο</sup>: Έρευνα (καινούρια ιδέα)

- Οι μαθητές εντοπίζουν την ύπαρξη ενός προβλήματος.
- Στη συνέχεια γίνεται συζήτηση για τις πιθανές λύσεις που θα μπορούσε να έχει το πρόβλημα αυτό.
- Κατόπιν γίνεται προσπάθεια κατανόησης όλων των απαιτήσεων που θα έχει η επίλυση του συγκεκριμένου προβλήματος.
- Τελευταίο στάδιο του συγκεκριμένου βήματος είναι η σύλληψη του αρχικού τύπου του μοντέλου.

#### Βήμα 2<sup>ο</sup>: Σχεδιασμός-(Πρόταση)

- Σε αυτό το σημείο οι μαθητές καταγράφουν τη στρατηγική που θα ακολουθήσουν.
- Στη συνέχεια πρέπει να καταγράψουν και τις προδιαγραφές που θα έχει το μοντέλο αυτό.
- Τέλος έχουμε την ανάπτυξη χρονοδιαγράμματος υλοποίησης της δραστηριότητας.

#### Βήμα 3<sup>ο</sup>: Ανάπτυξη πρωτότυπου (Βελτίωση)

- Εδώ οι μαθητές καλούνται να κατασκευάσουν το πρωτότυπο μοντέλο ρομποτικής προσομοίωσης.
- Στη συνέχεια οφείλουν να κάνουν έναν επανέλεγχο όλων των σχεδίων που ακολούθησαν για τυχόν διορθώσεις.

#### Βήμα 4<sup>ο</sup> Δοκιμή (Λύση)

- Οι μαθητές προχωρούν στη δοκιμασία του μοντέλου που κατασκεύασαν στον φυσικό κόσμο.

Μελετώντας την εφαρμογή των ρομποτικών συστημάτων στη χώρα μας παρατηρούμε πως τα τελευταία χρόνια γίνονται προσπάθειες από το Υπουργείο Παιδείας με σκοπό τη διαμόρφωση ενός Ψηφιακού Σχολείου. Ο χαρακτήρας του σχολείου αυτού απαιτεί καινοτόμες διδακτικές μεθόδους χρήση νέων τεχνολογιών και όσο το δυνατόν μεγαλύτερη αξιοποίηση ψηφιακών διδακτικών μέσων. Αναμφίβολα

στα Ανώτατα Εκπαιδευτικά Ιδρύματα (Πανεπιστήμια, Τ.Ε.Ι, Ι.Ε.Κ. κλπ.) γίνεται συστηματική χρήση των Νέων Τεχνολογιών, και στα τμήματα όμως της Πρωτοβάθμιας και Δευτεροβάθμιας εκπαίδευσης γίνονται προσπάθειες οι μαθητές να έρθουν πιο κοντά με την Τεχνολογία και να είναι καλοί γνώστες χειρισμού του Ηλεκτρονικού Υπολογιστή (Παραπομπή ΝΔ).

Σε αυτό το σημείο όμως τίθεται το κρίσιμο ερώτημα του ρόλου του εκπαιδευτικού [40]. Σαφέστατα η διδασκαλία χωρίς τον εκπαιδευτικό δεν μπορεί να νοηθεί. Στην περίπτωση της Πληροφορικής όμως τα πράγματα είναι πιο σύνθετα. Ήδη από πολύ μικρή ηλικία τα παιδιά κατέχουν βασικές γνώσεις για το πώς να χειριστούν ένα υπολογιστή. Στην προκειμένη περίπτωση όμως δεν έχουμε να κάνουμε με μια απλή περιήγηση στο διαδίκτυο, αλλά με διαδικασίες πιο σύνθετες όπως είναι ο προγραμματισμός και η κατασκευαστική ικανότητα. Στα πλαίσια αυτά ο εκπαιδευτικός καλείται αρχικά να εξηγήσει τις βασικές έννοιες του προγραμματισμού στους μαθητές και στη συνέχεια να τους καθοδηγήσει με σωστές τεχνικές βήμα προς βήμα με σκοπό την κατασκευή ενός προγράμματος εντολών τις οποίες θα υλοποιήσει το μοντέλο ρομποτικής. Όλη αυτή η διαδικασία μπορεί να λάβει χώρα και εντός αλλά και εκτός σχολικής τάξης και αυτό μπορεί να γίνει με την ανάθεση ομαδικών εργασιών, διαθεματικών δραστηριοτήτων, επισκέψεις σε μουσεία τεχνολογίας κ.ο.κ Με αυτόν τον τρόπο οι μαθητές θα εξοικειωθούν με την επιστήμη της ρομποτικής ενώ παράλληλα διασκεδάζουν, συνεργάζονται με τους μαθητές τους και δημιουργούν. [39]

Εξίσου σημαντική όμως είναι η συνεισφορά της ρομποτικής και στη μάθηση άλλων γνωστικών αντικειμένων, όπως η φυσική, τα μαθηματικά, η ιστορία κλπ. Είναι ευτυχές το γεγονός ότι στην Αθήνα, αλλά και σε άλλες μεγάλες πόλεις της Ελλάδας, πραγματοποιούνται σεμινάρια από την εταιρεία ‘Διερευνητική Μάθηση’ τα οποία στοχεύουν στην κατανόηση από τους μαθητές αλλά και το εκπαιδευτικό προσωπικό του τρόπου ανάπτυξη μαθησιακών εφαρμογών. Η διαδικασία εκπαίδευσης αποτελείται από τα εξής στάδια:

- Στάδιο εμπλοκής: διατύπωση προβλήματος και διάλογος μαθητών με σκοπό την επίλυσή του.
- Στάδιο πειραματισμού: οι μαθητές έρχονται σε επαφή με άλλους πιο απλούς ρομποτικούς προσομοιωτές με σκοπό την κατανόηση του τρόπου λειτουργίας τους αλλά και τις δυνατότητές του.
- Στάδιο διερεύνησης: οι μαθητές επαναπροσδιορίζουν το πρόβλημα με σκοπό την καλύτερη κατανόησή του και διατυπώνουν ερωτήματα τα οποία οδηγούν στην επίλυση του.

- Στάδιο σύνθεσης και δημιουργίας: οι μαθητές ενώνουν τα στοιχεία που συνέλεξαν φτιάχνοντας έτσι ένα πλάνο επίλυσης του προβλήματος.
- Στάδιο αξιολόγησης: οι μαθητές παρουσιάζουν το πλάνο που κατασκεύασαν και οι συμμαθητές τους το αξιολογούν.

Τελειώνοντας, θα πρέπει να γίνει ειδική αναφορά στην συνεισφορά των προσομοιωτών της ρομποτικής σε έναν ξεχωριστό κλάδο, τον κλάδο της Ειδικής Αγωγής. Ειδική Αγωγή είναι μια πρόσφατη σχετικά επιστήμη η οποία έχει ως σκοπό της την ένταξη παιδιών με ειδικές ανάγκες στο οικογενειακό, σχολικό και κοινωνικό πλαίσιο όπου ζουν. Σύμφωνα με τον Νόμο 3699/2008, η εκπαίδευση του κλάδου της Ειδικής Αγωγής περιλαμβάνει το σύνολο των εκπαιδευτικών υπηρεσιών που παρέχονται σε μαθητές με επιβεβαιωμένες ειδικές εκπαιδευτικές ανάγκες και με ή χωρίς αναπηρία. Δεν χωρά αμφιβολία πως για τους περισσότερους η αλληλεπίδραση με τον ηλεκτρονικό υπολογιστή είναι μια διαδικασία εύκολη που αποτελείται από τρία βήματα: την είσοδο, την επεξεργασία και την έξοδο, οι οποίες πραγματοποιούνται με τη χρήση συγκεκριμένων συστημάτων που βασίζονται στις ανθρώπινες αισθήσεις. Όταν όμως έχουμε να κάνουμε με χρήστες οι οποίοι έχουν οποιαδήποτε μορφή αναπηρίας καταλαβαίνουμε πως η χρήση οποιουδήποτε τεχνολογικού μέσου είναι, αν όχι αδύνατη, πάρα πολύ δύσκολη.

Για την εξυπηρέτηση του συγκεκριμένου πληθυσμού αναπτύχθηκαν οι Υποστηρικτικές Τεχνολογίες (Assistive Technologies) [40]. Γενικά θα μπορούσαμε να πούμε πως στις Υποστηρικτικές Τεχνολογίες ανήκει οποιαδήποτε τεχνολογία χρησιμοποιείται από ΑμεΑ, με σκοπό την εξυπηρέτηση αναγκών. Στον κόσμο της πληροφορικής όμως ως Υ.Τ. ορίζουμε κάθε συσκευή ή λογισμικό που βοηθούν ΑμεΑ να χρησιμοποιούν τον υπολογιστή. Οι Υ.Τ. έρχονται να αντικαταστήσουν ή/και να ενισχύσουν κάποιες από τις ανθρώπινες αισθήσεις ή ικανότητες (όραση, ακοή κινητικές και γνωσιακές ικανότητες).

Ξεκινώντας από τα άτομα με προβλήματα όρασης υπάρχουν πολλοί προσομοιωτές της ανθρώπινης φωνής με διάφορες λειτουργίες όπως μετατροπές κειμένων σε ομιλία, αναγνώστες οθόνες, μεταφραστές Braille, λογισμικά αναγνώρισης ομιλίας κλπ. Από την άλλη, τα άτομα με προβλήματα ακοής μπορούν να αξιοποιούν συστήματα εναλλακτικής επικοινωνίας με τον υπολογιστή τα οποία βασίζονται σε προσομοιωτές που χρησιμοποιούν σύμβολα αντί για γράμματα και μεταφράζουν οποιαδήποτε ομιλία στη νοηματική γλώσσα. Τέλος, για τα άτομα με κινητικά προβλήματα υπάρχουν προσομοιωτές που τους διευκολύνουν στη χρήση του υπολογιστή (πχ διευκόλυνση κλικ).

Σύμφωνα με πρόσφατες παιδαγωγικές έρευνες υποστηρίζεται πως η μάθηση είναι πιο αποτελεσματική όταν συνδυάζει την δημιουργικότητα, την συνεργασία και

την σύνδεση του θεωρητικού μέρους με την ανθρώπινη καθημερινότητα. Επομένως, θα ήταν ανώφελο να εθελotuφλούμε και να μην αναγνωρίζουμε την αναγκαιότητα της τεχνολογικής επιμόρφωσης των μαθητών. Ακολουθώντας λοιπόν της επιταγές του σύγχρονου κόσμου το Υπουργείο Παιδείας προσπαθεί να εντάξει όσο το δυνατόν περισσότερο τις νέες τεχνολογίες στο σχολικό περιβάλλον, πάντα με γνώμονα το συμφέρον των μαθητών και της κοινωνίας ολόκληρης. Πάντα όμως πρέπει να τηρούνται τα μέτρα προφύλαξης των μαθητών και να αποφεύγεται η υπέρμετρη χρήση των υπολογιστών καθώς αυτή μπορεί να φέρει αντίθετα αποτελέσματα

### 3.3 Προσομοίωση στον χώρο της Γεωργίας



*Εικόνα 13: Προσομοίωση στον χώρο της Γεωργίας*

Σε μια χώρα όπως η Ελλάδα που η οικονομία της θα έπρεπε να βασίζεται πρωτίστως στην γεωργία τίθεται το ζήτημα εκσυγχρονισμού των μέσων παραγωγής. Η επιστήμη της ρομποτικής έρχεται να βοηθήσει για μία ακόμα φορά όλους αυτούς που έχουν ως αντικείμενο εργασίας τους τη γη. Όλοι γνωρίζουμε πόσο δύσκολες και χρονοβόρες είναι οι αγροτικές δραστηριότητες, οι οποίες λαμβάνουν χώρα σε περιβάλλοντα δύσκολα με απροσδιόριστες συνθήκες. Κάθε κομμάτι γης έχει τις δικές του ανάγκες, πράγμα που απαιτεί την ειδημοσύνη εκείνων που θα ασχοληθούν με αυτό. Το γεγονός όμως ότι οι εργασίες των αγροτών είναι συγκεκριμένες (όργωμα, ύδρευση κλπ.) επιτρέπει στις νέες τεχνολογίες να εισχωρήσουν και να αυτοματοποιήσουν κάθε μορφής δραστηριότητας.

Προσομοιωτές μπορούν να πραγματοποιήσουν οποιαδήποτε εργασία εκτελεί ένας απλός αγρότης. Αυτό μπορούμε εύκολα να το διαπιστώσουμε αν μελετήσουμε βήμα προς βήμα κάθε γεωργική ενέργεια. Αρχικά, έχουμε την εγκατάσταση της φυτείας, η οποία περιλαμβάνει τις εξής διαδικασίες: προετοιμασία εδάφους, χαρτογράφηση χώρου σποράς, έλεγχος σποράς και επανασπορά. Για κάθε μια από τις παραπάνω περιπτώσεις υπάρχουν ειδικές μηχανές που μπορούν να φέρουν εις πέρας κάθε ανάγκη. Για παράδειγμα για το όργωμα υπάρχουν ειδικά οχήματα μικρού μεγέθους και τηλεκατευθυνόμενα. Παράλληλα, για τη διαδικασία της χαρτογράφησης χρήσιμα μπορούν να φανούν τα RTK GPS, τα οποία τοποθετούνται στον σπορέα, και οι αισθητήρες υπέρυθρων ακτινών, οι οποίοι τοποθετούνται στην σπαρτική μηχανή. Και τα δύο είναι σε θέση να καταγράψουν τη θέση του σπόρου με ακρίβεια. Για την διαδικασία του ελέγχου της καλλιέργειας, η οποία μπορεί να γίνει με τη χρήση ενός αυτοματοποιημένου συστήματος το οποίο διαθέτει αισθητήρες δύνανται να εποπτεύει την κατάσταση της καλλιέργειας. Οι προσομοιωτές ρομποτικής όμως μπορούν να αναλάβουν και πιο σύνθετες διαδικασίες όπως είναι το βοτάνισμα. Για παράδειγμα, το μηχάνημα KVL αναλαμβάνει την καταστροφή ζιζανίων γύρω από οποιαδήποτε καλλιέργεια. Από την άλλη, υπάρχουν και συστήματα υπεύθυνα για τους ψεκασμούς, την άρδευση, την συγκομιδή κλπ.

Σε αυτό το σημείο θα καταγραφούν διάφορα παραδείγματα από την εφαρμογή των ρομποτικών συστημάτων στον κλάδο της γεωργίας. Αρχικά έχουμε τον αυτόνομο Γεωργικό Ελκυστήρα (HAKO), ο οποίος αποτελεί κομμάτι του εξελιγμένου καλλιεργητικού συστήματος, το οποίο βασίζεται σε καλλιεργητικές τεχνικές διαχείρισης της γεωργικής ακρίβειας. Οι τεχνικές αυτές βασίζονται στον διαχωρισμό σε τμήματα μιας καλλιέργειας, τα οποία διαθέτουν ίδιες ανάγκες, ενώ παράλληλα στοχεύουν στην επέκταση των καλλιεργητικών τεχνικών σε υψηλότερη χωρική ανάλυση έτσι ώστε να είναι σε θέση να μπορούν να αναγνωρίζουν τις μελλοντικές ανάγκες όχι μόνο της καλλιέργειας, αλλά και κάθε φυτού ξεχωριστά. Τα συστήματα που χρησιμοποιεί κυρίως είναι μηχανήματα σποράς ακριβείας με ταυτόχρονη καταγραφή πλέγματος σποράς, μηχανήματα μηχανικής καταπολέμησης ζιζανίων και μηχανήματα ψεκασμού και κοπής.

Ένα ακόμη παράδειγμα αποτελεί το αυτόνομο υποστηρικτικό όχημα ZEYS. Πρόκειται για μια ρομποτική μηχανή χαμηλού κόστους, το οποίο σχεδιάστηκε με σκοπό την εκτέλεση προγραμματισμένων ενεργειών εντός των ορίων της καλλιέργειας. Η ισχυρή του πλατφόρμα συνδυάζεται με μια αποδοτική μονάδα πληροφοριών με σκοπό την ταχεία λήψη ερεθισμάτων από τις περιφερειακές μονάδες. Τα προτερήματα αυτής της μηχανής είναι το χαμηλό της κόστος, η μαζική χρήση, η μεγάλη ευελιξία του (μπορεί εκτελέσει ποικίλες δραστηριότητες όπως ανίχνευση ζιζανίων, μέτρηση υγρασίας κλπ.) και ο κατάλληλος σχεδιασμός του

(μικρό βάρος και ακρίβεια αρθρώσεων). Το πιο σημαντικό πλεονέκτημα του όμως είναι το γεγονός της ελευθερίας στην πλατφόρμα του, η οποία βασίζεται σε μια ανοιχτή αρχιτεκτονική που μπορεί να φιλοξενήσει οποιοδήποτε υποσύστημα, με σκοπό την εκτέλεση οποιασδήποτε λειτουργίας. Ταυτόχρονα, μπορεί να φιλοξενήσει και επιπλέον εξοπλισμό για επιπλέον αγροτικές εργασίες [42].

Ξέχωρα όμως από τις καλλιέργειες, ρομποτικά συστήματα υπάρχουν και στα θερμοκήπια. Ειδικότερα, στα θερμοκήπια πραγματοποιούνται μαζικές γεωργικές παραγωγές προϊόντων σε περιοχές με δυσμενείς κλιματολογικές συνθήκες. Στόχος των θερμοκηπίων είναι η δημιουργία ιδανικών κλιματολογικών συνθηκών στο χώρο έτσι ώστε να ευδοκιμήσουν οι καλλιέργειες. Για να γίνει όμως αυτό απαιτείται πλήθος φυτοφαρμάκων και άλλων χημικών προϊόντων τα οποία καθιστούν το περιβάλλον τοξικό και άκρως επικίνδυνο για τον άνθρωπο.

Την λύση έρχεται να δώσει το Project Aurora. Πρόκειται για ένα είδος ρομπότ στιβαρό και χαμηλού κόστους και προσφέρεται για αυτόνομη πλοήγηση στον χώρο του θερμοκηπίου. Τον πλήρη έλεγχο των κινήσεων του έχει ο χειριστής, ο οποίος επιβλέπει την πορεία του συστήματος σε μια οθόνη που έχει τοποθετήσει στον σταθμό εργασίας. Έτσι μπορεί να ελέγχει την κατάσταση του θερμοκηπίου, να συλλέγει και να επεξεργάζεται δεδομένα σ' ένα περιβάλλον φιλικό προς αυτόν. Οι λειτουργίες του συστήματος αυτού όμως δεν σταματούν εδώ. Το ρομπότ αναλαμβάνει και την ανάλυση της κατάστασης της υγείας του φυτού, μελετώντας τα συμπτώματα που μπορεί να παρουσιάζει, με τη χρήση ειδικών αισθητήρων. Κατόπιν της διάγνωσης, ακολουθεί η θεραπεία σε τοπικό επίπεδο, η οποία βασίζεται στην εργαστηριακή μελέτη που έχει προηγηθεί στις σωστές περιβαλλοντικές συνθήκες. Παράλληλα, αναλαμβάνει και την προληπτική προστασία των φυτών του θερμοκηπίου με τον ψεκασμό της καλλιέργειας. Για να γίνει αυτό, θα πρέπει να τοποθετηθεί στα άκρα του μηχανήματος ειδικός εξοπλισμός (συσκευή πένσας-ψαλιδιού και συσκευή ψεκασμού, κάμερες κλπ.). Για τα θερμοκήπια όμως υπάρχει και η εναλλακτική επιλογή του προτεινόμενου κηπουρού. Πρόκειται για μια εφαρμογή που ελέγχει την κατάσταση της υγείας των φυτών αλλά και των περιβαλλοντικών συνθηκών του εσωτερικού των θερμοκηπίων. Παράλληλα, το σύστημα αυτό έχει τη δυνατότητα θεραπείας των ασθενούντων φυτών, και αν ακόμη ο τρόπος αντιμετώπισης της ασθένειας δεν είναι σίγουρος, το ρομπότ προχωρά σε συλλογή δείγματος από το φυτό και το μεταφέρει στο εργαστήριο για περαιτέρω μελέτη. Με αυτό τον τρόπο καμία ανθρώπινη ζωή δεν κινδυνεύει και κάθε στάδιο της καλλιέργειας είναι πλήρως ελεγχόμενο.

Πριν από τη βιομηχανική επανάσταση, ο κλάδος της γεωργίας απαιτούσε πολύ ανθρώπινο μόχθο και η παραγωγή περιοριζόταν στην οικογενειακή κατανάλωση. Με την εισχώρηση των μηχανών όμως στην παραγωγή, το κόστος και ο

χρόνος παραγωγής, το εργατικό δυναμικό και ο ανθρώπινος κόπος μειώθηκαν σημαντικά, ενώ η παραγωγή αυξήθηκε σε πολύ μεγάλο βαθμό. Πριν την είσοδο των μηχανών λοιπόν στις καλλιέργειες, ο γεωργός γνώριζε το χωράφι του με το να περπατά διαρκώς κατά τη διάρκεια της βλάστησης και διαχειρίζονταν κάθε τμήμα του σύμφωνα με τις ανάγκες του. Με την εκμηχάνιση της γεωργίας όμως ο γεωργός έχασε αυτή την άμεση επαφή με το χωράφι του, καθώς ο ρόλος του περιορίστηκε απλώς στη διαχείριση συστημάτων που αναλάμβαναν εκείνα τις γεωργικές δραστηριότητες. Στην πορεία της εκβιομηχάνισης όμως προέκυψαν προβλήματα κατά τη διαδικασία της καλλιέργειας. Αρχικά, οι αγρότες γνώριζαν πως οι καλλιέργειες δεν είναι ομοιόμορφες και πως κάθε τμήμα τους έχει τις δικές του ανάγκες. Τα μηχανήματα όμως είχαν την φιλοσοφία της τυπικής διαχείρισης των χωραφιών με βάση τους μέσους όρους παραγωγής. Όπως είναι λογικό, σε πολλά σημεία της καλλιέργειας, η ποσότητα λιπάσματος ή ύδρευσης, για παράδειγμα, δεν συμβάδιζε με τις ανάγκες του φυτού. Αυτό οδηγούσε στη μείωση παραγωγής και τη σπατάλη πρώτων υλών.

Τη λύση στο πρόβλημα ήρθαν να δώσουν οι νέες τεχνολογίες οι οποίες επέτρεψαν την μέτρηση χωρικής και χρονικής παραλλακτικότητας των παραμέτρων παραγωγής και του εδάφους.

Βέβαια, όπως συμβαίνει σε κάθε ανθρώπινο επίτευγμα, έτσι και στην προκειμένη περίπτωση υπάρχουν αρνητικά και θετικά αποτελέσματα για τον άνθρωπο μεμονωμένα, αλλά και ως σύνολο στην κοινωνία. Έτσι αναπτύχθηκε η γεωργία ακριβείας. Η γεωργία ακριβείας αφορά τη διαχείριση της χωρικής και χρονικής παραλλακτικότητας των αγρών, με σκοπό τη βελτίωση της αποδοτικότητας των αγροκτημάτων σε συνδυασμό με τη μείωση των αρνητικών επιπτώσεων στο περιβάλλον από την μη ορθολογική χρήση των εισροών. Πρόκειται δηλαδή για ένα σύστημα διαχείρισης αγροκτημάτων βασιζόμενο στις αρχές της πληροφορικής και της ηλεκτρολογίας εφαρμοσμένες όμως στον κλάδο της γεωργίας. Αυτό το σύστημα μεριμνά στην βοήθεια του γεωργού να πάρει τις σωστές αποφάσεις για την καλύτερη διαχείριση του αγροκτήματος του [43]. Η καλύτερη διαχείριση μισ καλλιέργειας συνεπάγεται με τη βελτίωση της οικονομικής απόδοσης της καλλιέργειας αυτής, την αύξηση της παραγωγής, την μείωση των εισροών και τη βελτίωση τυχόν αρνητικών επιπτώσεων της γεωργίας στο περιβάλλον. Όλα αυτά γιατί οι νέες τεχνολογίες δίνουν τη δυνατότητα οι καλλιεργητικές φροντίδες να γίνονται με βάση τις μέσες τιμές παραγωγής και γονιμότητας του χωραφιού και ταυτόχρονα να εφαρμόζεται και διαφορετική δόση εισροών σε κάθε τμήμα του χωραφιού ανάλογα με τις ανάγκες του.

Με τη βελτίωση της ακρίβειας των συστημάτων γεωγραφικού εντοπισμού (GPS) και την ανάπτυξη των αισθητήρων μέτρησης της παραγωγής καλλιεργειών, δόθηκε η δυνατότητα χαρτογράφησης της παραγωγής, με την αρχή να εντοπίζεται το

1980. Στις εφαρμογές αυτές, η μέτρηση της ροής και της υγρασίας των φυτών γίνονταν κατά τη διάρκεια της λειτουργίας της μηχανής συγκομιδής. Ταυτόχρονα υπήρχε η δυνατότητα συνδυασμού των παραπάνω διαδικασιών με καταγραφή της ταχύτητας εργασίας και τη γεωγραφική θέση της μηχανής. Τέλος, η εκτίμηση του πλάτους εργασίας γινόταν από τον ίδιο τον χειριστή ή από άτομο που ήταν πάνω στη μηχανή. Όλες αυτές οι πληροφορίες απετέλεσαν τη βάση για την παραγωγή χαρτών που εμφανίζουν τη χωρική κατανομή της παραγωγής με χρήση λογισμικού γεωγραφικού συστήματος πληροφοριών (GIS).

Υπάρχουν όμως αρκετές χώρες στις οποίες πρέπει να γίνουν αρκετές αλλαγές όπως επιπλέον μείωση του κόστους παραγωγής μέσω της περαιτέρω εκμηχάνισης της γεωργίας και ταυτόχρονη μείωση των εργατικών ημερομισθίων. Τις χώρες που αφορούν αυτά τα μέτρα τις εντοπίζουμε στο Νότιο ημισφαίριο, και δυστυχώς στις χώρες αυτές συμπεριλαμβάνεται και η Ελλάδα. Παράλληλα, σε όλες τις περιοχές ανά τον κόσμο, αφού έχει προηγηθεί η συλλογή δεδομένων μέσω των ηλεκτρονικών συστημάτων καταγραφής, θα πρέπει να γίνει και η επεξεργασία των δεδομένων που συλλέχθηκαν με σκοπό την ανεύρεση ζωνών διαχείρισης (management zones) εντός του αγροτεμαχίου, που χαρακτηρίζονται από κοινά εδαφολογικά ή αγρονομικά χαρακτηριστικά.

Καταλαβαίνουμε λοιπόν πως βάση της εφαρμογής της Αγροτικής Ακριβείας αποτελεί η δημιουργία ζωνών διαχείρισης των αγροτεμαχίων. Για να γίνει αυτό αξιοποιούνται τα δεδομένα που συλλέχθηκαν από τα συστήματα καταγραφής. Παράλληλα, αξιοποιούνται αναλύσεις των φυσικών και χημικών ιδιοτήτων της καλλιέργειας, εικόνες από τον αέρα (αεροπλάνα ή δορυφόρους) και διάφορα μηχανήματα που μπορούν να χαρτογραφήσουν άμεσα τις ιδιότητες του αγρού. Παράδειγμα μηχανήματος που υλοποιεί αντίστοιχες λειτουργίες είναι το Veris που δύναται να χαρτογραφήσει την ηλεκτρική αγωγιμότητα του εδάφους με περάσματα στον αγρό, ελκόμενο από γεωργικό ελκυστήρα. Μια ακόμα σπουδαία χρήση των ζωνών διαχείρισης των αγροτεμαχίων, είναι η δυνατότητα καλλιεργητικών επεμβάσεων με μεταβαλλόμενες δόσεις (Variable Rate Application). Εξάλλου αυτός είναι και ο σκοπός της Γεωργίας Ακριβείας, δηλαδή η εφαρμογή μεταβλητών καλλιεργητικών φροντίδων για κάθε τμήμα του αγρού, ανάλογα με τις πραγματικές του, ανά περιοχή, ανάγκες. Όλες αυτές οι λειτουργίες συμβάλλουν στην εξοικονόμηση πόρων, στην καλύτερη διαχείριση των αγροτικών εκμεταλλεύσεων σε επίπεδο παραγωγού, αλλά και στην προστασία του περιβάλλοντος από την αλόγιστη χρήση των εισροών, που είναι ιδιαίτερα σημαντικό για τη χώρα μας.

Μπορεί όλα τα παραπάνω να έχουν κάνει την εργασία των αγροτών πιο εύκολη, αυτό όμως δεν σημαίνει πως δεν κρύβουν και κινδύνους και δυσκολίες. Αρχικά ο χειρισμός όλων αυτών των μηχανημάτων απαιτεί εξειδικευμένες γνώσεις

όχι μόνο γεωργικού περιεχομένου. Οι χειριστές θα πρέπει να είναι καλοί γνώστες πληροφορικής ώστε να μπορούν να κατευθύνουν σωστά οποιονδήποτε ρομποτικό προσομοιωτή δίνοντας του τις κατάλληλες εντολές. Παράλληλα, θα πρέπει να έχουν και ηλεκτρολογικές γνώσεις, οι οποίες σχετίζονται με τη λειτουργία του συστήματος. Σε κάθε περίπτωση οφείλουμε να αποδεχτούμε πως οι νέες τεχνολογίες δεν ήρθαν για να αντικαταστήσουν τον άνθρωπο, αλλά για να τον διευκολύνουν. Συνεπώς, κάθε ρομποτικό σύστημα οφείλει να προγραμματίζεται με γνώμονα όχι μόνο το ατομικό όφελος του ιδιοκτήτη, αλλά και το κοινό καλό.

### 3.4 Προσομοίωση στον χώρο της Ιατρικής



Εικόνα 14: Προσομοίωση στον χώρο της Ιατρικής

Αναμφίβολα ο κλάδος που επωφελήθηκε περισσότερο από τα ρομποτικά συστήματα είναι ο κλάδος της ιατρικής. Αρχικά, στον τομέα της χειρουργικής χάρη στη ρομποτική τεχνολογία, η ψηφιακή ανάλυση έδωσε τη δυνατότητα μεταφοράς των δεδομένων σε μεγάλες αποστάσεις. Έτσι δημιουργήθηκε η τηλεχειρουργική, δηλαδή η εξ αποστάσεως, θα μπορούσαμε να πούμε επέμβαση, η οποία δεν απαιτεί την ταυτόχρονη παρουσία γιατρού και ασθενούς στον ίδιο χώρο. Παράλληλα, με την ρομποτική χειρουργική υπερβαίνονται και οι περιορισμοί της λαπαροσκοπικής επέμβασης, καθώς ο γιατρός διαθέτει πιο εύκαμπτα εργαλεία, περισσότερη ελευθερία κινήσεων και ακριβέστερη εικόνα του χειρουργικού πεδίου. Σημαντική όμως είναι

και η συμβολή της ρομποτικής χειρουργικής και στην υλοποίηση επεμβάσεων περιορισμένου πεδίου (π.χ. παιδοχειρουργική).

Παίρνοντας τα πράγματα από την αρχή, πριν από περίπου 20 χρόνια ξεκίνησε η εφαρμογή μιας νέας πολιτικής στον ιατρικό χώρο. Η εφαρμογή αυτή έφερε την ονομασία ελάχιστα επεμβατική χειρουργική, με στόχο την υλοποίηση χειρουργικών επεμβάσεων με όσο το δυνατόν λιγότερο τραυματισμό και χρόνο ανάρρωσης των ασθενών. Κατά την διαδικασία της ελάχιστα επεμβατικής χειρουργικής, μικροσκοπικές κάμερες, προσαρμοσμένες σε όργανα, καλούνται ενδοσκόπια ή λαπαροσκοπικά να εισέλθουν στο εσωτερικό του οργανισμού από μικρές τομές που έχουν προηγηθεί, καθοδηγούμενες από τους γιατρούς, οι οποίοι παρακολουθούν την πορεία τους από μια οθόνη. Η εφαρμογή αυτή γνώρισε μεγάλη επιτυχία στον κλάδο της ιατρικής με αποτέλεσμα την δημιουργία ενός ξεχωριστού τομέα, του τομέα της λαπαροσκοπικής ιατρικής, ο οποίος βοήθησε σημαντικά τις συνθήκες νοσηλείας και φροντίδας των ασθενών καθώς ο μετεγχειρητικός πόνος, η απώλεια αίματος, οι μολύνσεις και οι επιπλοκές σχεδόν εξαλείφθηκαν.

Ωστόσο, η λαπαροσκοπική χειρουργική είχε κάποιες αδυναμίες από τη στιγμή που ο χειρουργός δεν έρχεται σε άμεση επαφή με τον ασθενή, καθώς οι κινήσεις του καθοδηγούνται από το βίντεο. Έτσι, ο γιατρός αδυνατεί να ψηλαφίσει το εσωτερικό του οργανισμού του ασθενούς και ταυτόχρονα η χρήση μακριών και άκαμπτων εργαλείων περιορίζουν την ευελιξία και την δυνατότητα σύνθετων χειρισμών. Με αυτόν τον τρόπο οδηγούμαστε στο συμπέρασμα ότι η λαπαροσκοπική χειρουργική απαιτεί μακροχρόνια εκπαίδευση, ακρίβεια και επιδεξιότητα. Συνεπώς πραγματοποιήθηκε έρευνα ώστε ο χειρουργός να μπορέσει να βελτιώσει τις ικανότητές του ή ακόμα και να αντικατασταθεί η παρουσία του, όπου οι συνθήκες το επιτρέπουν. Επομένως, παρουσιάζει ιδιαίτερη δυσκολία η προσομοίωση επιτυχημένης επέμβασης στην λαπαροσκοπική χειρουργική.

Φτάνουμε λοιπόν με αυτό τον τρόπο στην υποβοηθούμενη υπολογιστική χειρουργική, όπου πρωταγωνιστής μιας επέμβασης είναι ο χειρουργός, του οποίου όμως οι αισθήσεις και οι ικανότητες έχουν αναβαθμιστεί ποιοτικά με σκοπό την βελτίωση της απόδοσης του στο χειρουργείο. Η αναβάθμιση αυτή γίνεται με τεχνητές αισθήσεις, με τρισδιάστατη όραση και αφή και με τρισδιάστατα διαγνωστικά και απεικονιστικά μηχανήματα.

Χαρακτηριστικό παράδειγμα τέτοιου μηχανήματος είναι ο ρομποτικός βραχίονας, ο οποίος πραγματοποιεί ελεγχόμενες κινήσεις από τον υπολογιστή υπακούοντας τις εντολές του γιατρού. Άλλο παράδειγμα αποτελούν τα ελεγχόμενα από τον επόπτη συστήματα. Σύμφωνα με αυτά, το χειρουργείο προγραμματίζεται από πριν έτσι ώστε οι εντολές που θα επιτελέσει το ρομπότ να υπάρχουν ήδη και να

εκτελούνται αυτόματα, πάντα υπό την επίβλεψη του ιατρού. Στη συνέχεια έχουμε τα συστήματα τηλεχειρουργικής, όπου οι επεμβάσεις γίνονται από απόσταση με τη βοήθεια εξελιγμένων αισθητήρων που συνεχώς ανατροφοδοτούν την εικόνα του χειρουργικού πεδίου στον πραγματικό χώρο και χρόνο. Τέλος έχουμε τα συστήματα μοιραζόμενου ελέγχου τα οποία απαιτούν μεγαλύτερη συμμετοχή του ιατρού. Η συγκεκριμένη τεχνολογία βασίζεται στην ιδέα της υποβοηθούμενης χειρουργικής, γιατί τον μεγαλύτερο ρόλο στο χειρουργείο τον έχει ο γιατρός. Αντίθετα, το ρομπότ περιορίζεται απλώς στην παροχή βοήθειας.

Τα ρομποτικά συστήματα όμως προκειμένου να μπορούν να αξιοποιηθούν στον κλάδο της χειρουργικής θα πρέπει να πληρούν κάποιες προϋποθέσεις. Αρχικά θα πρέπει να έχουν αποτελεσματικό έλεγχο της δύναμης και της ταχύτητας του τελικού σημείου δράσης ενώ ταυτόχρονα θα πρέπει να έχουν περιορισμένο χώρο εργασίας για την αποφυγή τραυματισμών. Κατόπιν, θα πρέπει να έχουν και ελεγχόμενη δύναμη από τον χειρουργό, κάθε ρομπότ να είναι πλήρως εξοπλισμένο και κάθε κίνησή του να ελέγχεται καθολικά από τον ιατρό και να υπάρχει η δυνατότητα ασφαλούς λειτουργίας σε περίπτωση ατυχήματος. Τέλος, θα πρέπει τα εξαρτήματα να είναι πλήρως αποστειρωμένα για λόγους υγιεινής και να υπάρχει προστασία από την μαγνητική παρεμβολή λόγω της χρήσης άλλων χειρουργικών εργαλείων με μαγνητικό πεδίο.

Σε αυτό το σημείο θα γίνει αναφορά για τις διάφορες κατηγορίες συστημάτων που υπάρχουν για τα χειρουργικά ρομπότ. Αρχικά έχουμε τα παθητικά συστήματα τα οποία λειτουργούν υποστηρικτικά σε μια επέμβαση, χωρίς να έχουν καμιά ενεργό συμμετοχή σ' αυτή καθώς όλη η εργασία γίνεται από τον χειρουργό. Χαρακτηριστικό παράδειγμα τέτοιου συστήματος είναι το Acrobot που αναπτύχθηκε από τον B.L. Davies με σκοπό την υποβοήθηση στις ορθοπεδικές επεμβάσεις. Το ρομπότ αυτό προσφέρει μεγάλη ακρίβεια αλλά και ασφάλεια στον χειρουργό καθώς περιορίζει την τρισδιάστατη κίνηση (τακτική hands-on robotics). Το σημαντικότερο όμως πλεονέκτημα του συστήματος αυτού είναι παρέχει εργαλεία για τον χειρισμό μαλακών ιστών ενώ ταυτόχρονα καθιστά δυνατή την προσέγγιση της ελάχιστα επεμβατικής χειρουργικής. Αλλά παρόμοια συστήματα είναι το acrobotmpdeller, το acrobotplanner, το acrobotnavigator κλπ.

Μια άλλη κατηγορία συστημάτων είναι τα ημιενεργά, τα οποία έχουν επίσης βοηθητικό ρόλο καθώς κρατούν σταθερή θέση στο εργαλείο ώστε να είναι δυνατή η ακριβής καθοδήγησή του. Χαρακτηριστικό παράδειγμα τέτοιου τύπου εργαλείου είναι το Neuromate, το οποίο αναπτύχθηκε από την Integrated Surgical Systems και χρησιμοποιείται κυρίως στον κλάδο της νευροχειρουργικής. Πιο συγκεκριμένα, περιλαμβάνει έναν ειδικό βραχίονα με πέντε βαθμούς ελευθερίας κι ένα σύστημα σχεδιασμού βασισμένο σε υπολογιστή, ενώ το λογισμικό του επιτρέπει έναν ακριβή

σχεδιασμό που στηρίζεται σε δεδομένα-εικόνες που συλλέχθηκαν από αξονικές ή μαγνητικές τομογραφίες. Το σύστημα αυτό παρέχει τη δυνατότητα τρισδιάστατης αναπαράστασης των ανατομικών δομών του ασθενών και των σημείων εκείνων του εγκεφάλου που ο χειρουργός στοχεύει να παρέμβει. Άλλα συστήματα ανάλογης λειτουργίας είναι το pathfinder, το veebot κλπ.

Επόμενη κατηγορία είναι τα ενεργά ρομποτικά συστήματα, τα οποία υλοποιούν χειρουργικές επεμβάσεις χωρίς την άμεση παρέμβαση του χειρουργού. Χαρακτηριστικό παράδειγμα τέτοιου μοντέλου είναι το ROBODOC, το οποίο επιτρέπει στους γιατρούς να σχεδιάζουν προεγχειρητικά τις επεμβάσεις σ' ένα τρισδιάστατο εικονικό περιβάλλον και στη συνέχεια να προχωρά στην εγχείρηση έτσι όπως αυτή αρχικά σχεδιάστηκε. Κύρια εφαρμογή του συστήματος είναι στην αρθροπλαστική ισχίου όπου η πάσχουσα άρθρωση αντικαθίσταται από μια τεχνητή πρόθεση. Το σύστημα αυτό αποτελείται από δύο υποσυστήματα, το καθένα με δική του λειτουργία. Το πρώτο είναι το ORTHODOC το οποίο είναι υπεύθυνο για τον προεγχειρητικό σχεδιασμό βάσει ενός λογισμικού το οποίο εμφανίζει στον γιατρό μια τρισδιάστατη πληροφορία και εύκολο έλεγχο. Το δεύτερο σύστημα είναι το ROBODOC, το οποίο εκτελεί την χειρουργική υποβοήθηση μετατρέποντας την αξονική τομογραφία της άρθρωσης σε μια τρισδιάστατη εικόνα της άρθρωσης. Κατόπιν ο χειρουργός επιλέγει το μοντέλο, το μέγεθος και τον τύπο των εμφυτευμάτων από την ψηφιακή βιβλιοθήκη που διαθέτει και με τη βοήθεια του συστήματος προχωρά στην ακριβή εφαρμογή του. Άλλα παρόμοια συστήματα είναι το CyberKnife κλπ.

Έπειτα, έχουμε τα συστήματα απεικόνισης και πλοήγησης στο χώρο τα οποία συνήθως χρησιμοποιούνται πριν από την εγχείρηση. Ειδικότερα, τα συστήματα αυτά έχουν ως ρόλο να σχεδιάζουν τον τρόπο εκτέλεσης της επέμβασης μέσω μιας οθόνης, η οποία αναπαριστά τον τρόπο εκτέλεσης της επέμβασης έχοντας όμως προηγηθεί η συλλογή δεδομένων από σαρωτές τομογράφους. Παράδειγμα αυτού του συστήματος είναι το HipNav, το οποίο δίνει τη δυνατότητα στον χειρουργό να προσδιορίσει τη βέλτιστη τοποθέτηση του εμφυτεύματος στον ασθενή. Το μοντέλο αυτό όμως περιλαμβάνει τρεις συνιστώσες: ένα προεγχειρητικό σχεδιασμό, έναν προσομοιωτή κίνησης κι ένα διεγχειρητικό σύστημα παρακολούθησης και καθοδήγησης. Ο προσομοιωτής αυτός στοχεύει στη μείωση εξάρθρώσεων κατόπιν της ολικής αρθροπλαστικής του ισχίου και στον προσδιορισμό και πιθανόν την αύξηση του εύρους της κίνησης, την μείωση της φθοράς και την τροχιά σε πραγματικό χρόνο της πυέλου και της κοτύλης κατά τη διάρκεια του χειρουργείου. Άλλα παρόμοια μοντέλα είναι το Navitrak, το Optotrak κλπ.

Τέλος, έχουμε τα συστήματα που λειτουργούν με τηλεχειρισμό, δηλαδή η εργασία εκτελείται από ένα άτομο που βρίσκεται από απόσταση, έτσι ώστε να έχουμε

μια αλληλεπίδραση ανθρώπου και μηχανής με στόχο την εκτέλεση μιας εργασίας. Παράδειγμα αυτής της λειτουργίας είναι η τηλεχειρουργική, όπου ο χειρουργός πραγματοποιεί την επέμβαση μπροστά σε μια κονσόλα ελέγχου μακριά από τον ασθενή. Μοντέλα τέτοιου τύπου είναι το Zeus ο οποίος διαθέτει ρομποτικούς βραχίονες προσαρμοσμένους πάνω στο χειρουργικό τραπέζι και έναν υπολογιστή ελεγκτή. Ο βραχίονας αυτός καθοδηγεί το ενδοσκόπιο μέσα στο σώμα του ασθενούς με τη βοήθεια φωνητικών εντολών από τον χειρουργό, έτσι ώστε αυτός να διαθέτει μια τρισδιάστατη σταθερή και μεγεθυμένη εικόνα του χειρουργικού πεδίου. Άλλα συστήματα είναι το DaVinci, το EndoWrist κλπ.

Η αξιοποίηση των ρομποτικών συστημάτων έχει προσφέρει πάρα πολλά οφέλη στο τομέα της ιατρικής. Αρχικά το ιατρικό προσωπικό έχει μεγαλύτερη άνεση κατά τη διάρκεια μια επέμβασης, έχει καλό συντονισμό στις κινήσεις του και έτσι μπορεί να πραγματοποιεί δύσκολους χειρουργικούς χειρισμούς. Επιπλέον, οι χειρουργοί έχουν στη διάθεσή της οπτικά πεδία, που χωρίς τα ρομποτικά συστήματα δεν θα είχε πρόσβαση και παράλληλα, μπορούν να έχουν τον πλήρη έλεγχο πριν αλλά και κατά διάρκεια ενός χειρουργείου. Έτσι, αποφεύγονται τα λάθη ή και τυχόν επιπλοκές στην υγεία του ασθενούς. Ταυτόχρονα, από τη ρομποτική χειρουργική επωφελείται και ο ασθενής, καθώς βιώνει μια ελάχιστα τραυματική μέθοδο, χάνει ελάχιστο αίμα, μειώνεται ο μετεγχειρητικός πόνος και νιώθει μεγαλύτερη ασφάλεια καθώς μειώνονται οι πιθανότητες εμφάνισης επιπλοκών στην υγεία του.

Ωστόσο, αυτή η ευρεία χρήση της ρομποτικής κρύβει και αρκετούς κινδύνους αλλά και μειονεκτήματα τα οποία πρέπει να λάβουμε σοβαρά υπόψη. Αρχικά έχει μεγάλο κόστος η χρήση της, και για να γίνει εφαρμογή της, ειδικά στα δημόσια νοσοκομεία, φαντάζει ουτοπία. Επιπλέον, η εφαρμογή της απαιτεί την χρήση ενός πολύ μεγάλου όγκου διαφορετικών συστημάτων, δύσκολων στον χειρισμό. Παράλληλα, ο ηλεκτρονικός χαρακτήρας των συστημάτων σημαίνει πως αυτά είναι πιθανό να εκδηλώσουν τις αδυναμίες που θα εκδήλωνε μια οποιαδήποτε ηλεκτρονική συσκευή (π.χ. αναστολή λειτουργίας σε περίπτωση διακοπής παροχής ρεύματος, μπλοκάρισμα κλπ.). Τέλος, ο χειρισμός τέτοιων μηχανημάτων απαιτεί εξειδικευμένες γνώσεις από το ιατρικό προσωπικό, πράγμα που δυσχεράνει ακόμη περισσότερο την εφαρμογή τους.

Αναμφίβολα, η πρόσφορα των νέων τεχνολογιών στην ιατρική έχει συμβάλλει στην εξέλιξη του κλάδου κάνοντας τις συνθήκες και τον τρόπο εργασίας των ιατρών ευκολότερη. Θα πρέπει όμως να γνωρίζουμε πως κάθε εξελικτικό στάδιο έχει κι ένα τίμημα για αυτούς που το υφίστανται.

## Κεφάλαιο 4: Μελέτη Περίπτωσης Εφαρμογής Ρομποτικού Προσομοιωτή: JdeRobot framework – Program-A-Robot 2019

### 4.1 Εισαγωγή

Το JdeRobot είναι μια πλατφόρμα ανοιχτού λογισμικού, με άδεια GPL και LGPL, με σκοπό την υλοποίηση ρομποτικών εφαρμογών και εφαρμογών υπολογιστικής όρασης. Είναι βασισμένο στην C++ και την Python και παρέχει ένα καταναμημένο προγραμματιστικό περιβάλλον μέσω του οποίου προκύπτουν εφαρμογές που αποτελούνται από μια συλλογή πολλών ταυτόχρονων και ασύγχρονων χαρακτηριστικών. Μπορούν να εκτελούνται σε διαφορετικούς υπολογιστές και να συνδέονται μέσω ICE middleware ή ROS μηνυμάτων. Το JdeRobot απλοποιεί την πρόσβαση σε συσκευές υλικού από το πρόγραμμα εφαρμογής. Οι μετρήσεις από αισθητήρες ή εντολές για κίνηση κάποιου móτορα πραγματοποιούνται με το κάλεσμα τοπικών συναρτήσεων. Αυτές οι συναρτήσεις δημιουργούν το API για το Hardware Abstraction Layer. Χρησιμοποιεί και λογισμικά τρίτων όπως τον προσομοιωτή Gazebo, ROS, OpenGL, GTK, Qt, Player, Stage, GSL, OpenCV, PCL, Eigen και Ogre.

Παρακάτω φαίνονται τα ρομπότ και οι συσκευές που υποστηρίζονται αυτή την στιγμή καθώς και τι προγραμματιστικά εργαλεία και βιβλιοθήκες περιλαμβάνει το JdeRobot:

- RGBD αισθητήρες: Kinect and Kinect2 from Microsoft, Asus Xtion.
- Ρομπότ με ρόδες: TurtleBot from Yujin Robot and Pioneer from MobileRobotics Inc.
- ArDrone quadrotor από Parrot.
- Laser Scanners: LMS από SICK, URG από Hokuyo και RPLidar.
- Gazebo προσομοιωτής.
- Κάμερες firewire, κάμερες USB, αρχεία βίντεο (όπως mpeg, avi), κάμερες IP (όπως Axis).
- Περιλαμβάνει τηλεχειριστές για διάφορα ρομπότ, προβάλλοντας τους αισθητήρες τους και με δυνατότητα ελέγχου στους móτορες. Κάποιοι μπορεί να είναι web-based και να μπορούν να τρέξουν και από κινητές συσκευές.
- Εργαλείο Scratch2JdeRobot για τον προγραμματισμό ρομπότ και drone.

- Μια κάμερα βαθμονόμησης, ένα εργαλείο για την ρύθμιση φίλτρων χρώματος, μια βιβλιοθήκη για υλοποίηση controller και μια βιβλιοθήκη για την επεξεργασία υπολογιστικής όρασης.

Στο κεφάλαιο αυτό, θα αναλυθεί η μελέτη περίπτωσης στην οποία, με την χρήση του JdeRobot, δημιουργούνται ειδικές ασκήσεις και προκλήσεις προσομοίωσης οι οποίες φιλοξενούνται από το JdeRobot Academy και χρησιμοποιούνται για την διεξαγωγή της ρομποτικής δοκιμασίας Program-A-Robot του διαγωνισμού Aegean Robotics.

## 4.2 Program-A-Robot 2019

### 4.2.1 Περιγραφή

Η JdeRobot Academy είναι μια επέκταση του JdeRobot. Είναι μία εκπαιδευτική πλατφόρμα όπου υπάρχουν πολλές ασκήσεις σχετικά με την υπολογιστική όραση, αυτόνομα αυτοκίνητα, κινητά ρομπότ και drones. Οι χρήστες μπορούν να δημιουργήσουν το δικό τους λογαριασμό και να εγγραφούν στις ασκήσεις που τους ενδιαφέρουν και να αλληλοεπιδράσουν με αυτές προκειμένου να εξασκηθούν και να τεστάρουν τις γνώσεις τους. Οι περισσότερες ασκήσεις χρησιμοποιούν τον προσομοιωτή Gazebo για την εμφάνιση και μοντελοποίηση των αντικειμένων και του περιβάλλοντος των ασκήσεων και το Jupyter Notebook για την ανάπτυξη και τον προγραμματισμό των αντικειμένων και ρομπότ της άσκησης.

Η ομάδα AegeanRobotics του Εργαστηρίου Τεχνητής Νοημοσύνης και Υποστήριξης Αποφάσεων του Τμήματος Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων του Πανεπιστημίου Αιγαίου διοργάνωσε για τρίτο συνεχόμενο έτος τον διαγωνισμό AegeanRobotics 2019. Ο διαγωνισμός διεξήχθη το Σαββατοκύριακο 13 και 14 Απριλίου 2019 στο Καρλόβασι Σάμου. Περιείχε τέσσερις δοκιμασίες: την δοκιμασία RoboSumo, την δοκιμασία Freestyle, την δοκιμασία Line Follower και την δοκιμασία Program-A-Robot.

Η δοκιμασία Program-A-Robot οργανώθηκε σε συνεργασία με το Πανεπιστήμιο Rey Juan Carlos της Μαδρίτης. Στοχεύει σε προπτυχιακούς φοιτητές και απόφοιτους μηχανικών σχολών, μεταπτυχιακούς φοιτητές και διδακτορικούς και στόχος του είναι να προάγει την ρομποτική και την υπολογιστική όραση μέσω ειδικών ασκήσεων και να δώσει την ευκαιρία του ανταγωνισμού με άλλους συμμετέχοντες. Διεξήχθη για πρώτη φορά το 2016 ως ρομποτικός διαγωνισμός στο Πανεπιστήμιο Rey Juan Carlos. Στον τρίτο διαγωνισμό AegeanRobotics Competition

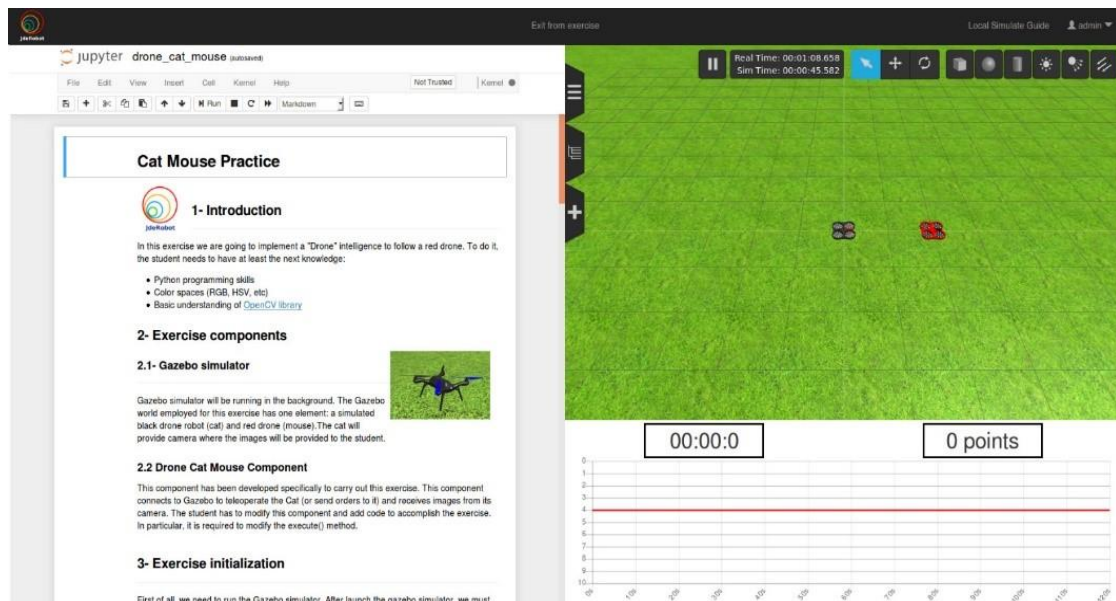
2019, η δοκιμασία Program-A-Robot αποτελούνταν από μια άσκηση προγραμματισμού ενός αυτόνομου drone. Το όνομα της άσκησης ήταν Drone Cat Mouse.

Στην δοκιμασία Program-A-Robot, υπήρχαν τρεις φοιτητές που συμμετείχαν και ένας φοιτητής που το διοργάνωσε και έδωσε συμβουλές και καθοδήγηση στους συμμετέχοντες. Οι συμμετέχοντες ήταν ο Πρόδρομος Αϊβατίδης και ο Μάριος Βασιλείου από το Πανεπιστήμιο Αιγαίου και ο Ραφαήλ Γιαβρίμης από το Πανεπιστήμιο του Surrey. Ο φοιτητής-προπονητής ήταν ο Γεώργιος Εξίζογλου από το Πανεπιστήμιο Αιγαίου. Καθώς ολοκληρώθηκε η πρόκληση, ο Προδρόμος Αϊβατίδης βγήκε πρώτος με 1160 πόντους, έπειτα ο Μάριος Βασιλείου δεύτερος με 1060 πόντους και στην τρίτη θέση ο Ραφαήλ Γιαβρίμης με 670 πόντους.

#### 4.2.2 Πρόκληση Drone Cat Mouse

Στην πρόκληση του Drone Cat Mouse, το drone Cat πρέπει να προγραμματιστεί για να ψάξει, να κυνηγήσει και να μείνει όσο το δυνατόν πιο κοντά σε ένα άλλο κινητό ασύρματο ρομπότ (drone Mouse) χωρίς να συγκρουστεί. Η εφαρμογή του διαιτητή μετρά την απόσταση μεταξύ των δύο drones και υπολογίζει μια βαθμολογία που βασίζεται σε αυτήν. Όσοι περισσότεροι είναι ο χρόνος που μένει το drone Cat κοντά στο drone Mouse, τόσο περισσότερη βαθμολογία θα υπολογίζεται. Το drone Mouse είναι κόκκινο, για ευκολότερη ανίχνευση στις εικόνες.

Στην πρόκληση υπάρχει έτοιμος ένας τρισδιάστατος κόσμος που έχει δημιουργηθεί στον προσομοιωτή Gazebo. Σε αυτό τον κόσμο υπάρχουν δύο ρομπότ: το AR.Drone που λειτουργεί ως drone Mouse (σε κόκκινο χρώμα) και το drone Cat (σε μαύρο χρώμα). Δεν υπάρχουν εμπόδια γύρω από τον κόσμο παρά μόνο το έδαφος. Ο κώδικας αναπτύσσεται μέσα στα αντίστοιχα κελιά του Jupyter Notebook της JdeRobot Academy WebIDE. Στην αρχή του Notebook, το αυτόνομο ποντίκι θα ξεκινήσει. Στην παρακάτω εικόνα, εμφανίζεται η κύρια σελίδα της πρόκλησης Drone Cat Mouse όπου το Jupyter Notebook βρίσκεται στα αριστερά, ο προσομοιωτής Gazebo με τα δύο drones και το περιβάλλον βρίσκονται πάνω δεξιά και η εφαρμογή του διαιτητή είναι η οθόνη μέτρησης με ζωντανό σκορ κάτω δεξιά:



Εικόνα 15: Αρχική σελίδα Drone Cat Mouse πρόκλησης

#### 4.2.3 Συνέντευξη νικητή

Όταν ολοκληρώθηκε με επιτυχία ο διαγωνισμός Program-A-Robot, ο Γεώργιος Εξίζογλου πήρε συνέντευξη από τον νικητή του διαγωνισμού, τον κ. Πρόδρομο Αϊβατίδη. Ο κ. Αϊβατίδης ανέφερε τα παρακάτω:

«Ο διαγωνισμός Program-A-Robot μου φάνηκε αρκετά πρωτοποριακός καθώς έδινε τη δυνατότητα στους συμμετέχοντες να λύσουν το πρόβλημα που περιέγραφε εξ αποστάσεως. Αυτό βέβαια είναι αρκετά σύνηθες στις μέρες μας, όμως το επίπεδό του ήταν υψηλότερο καθώς δεν περιελάμβανε την λύση ενός μαθηματικού προβλήματος για την εξαγωγή συγκεκριμένων αποτελεσμάτων ανάλογα με την είσοδο, όπως συνηθίζεται, αλλά περιελάμβανε την προσομοίωση δύο drones εκ των οποίων το ένα δρούσε τυχαία και το άλλο έπρεπε να προγραμματιστεί από τον συμμετέχοντα για να ακολουθεί το πρώτο από κοντά χωρίς όμως να το ακουμπήσει. Διαβάζοντας την ιστοσελίδα με τις οδηγίες και με την πολύτιμη βοήθεια του κ. Γεώργιου Εξίζογλου, κατάλαβα πως το drone που έπρεπε να προγραμματίσω είχε δύο κάμερες, μία που κοιτούσε μπροστά και μία που κοιτούσε κάτω, οι οποίες θα με βοηθούσαν στην επίλυση του προβλήματος. Στην αρχή είχα την εντύπωση πως δεν θα τα κατάφερνα καθώς η εμπειρία μου στην ψηφιακή επεξεργασία εικόνας περιοριζόταν σε κάποιες διδακτικές ώρες ενός μαθήματος στην σχολή που φοιτώ. Η ιστοσελίδα όμως με τις οδηγίες του διαγωνισμού και ο κ. Εξίζογλου, με βοήθησαν αρκετά να εκτελέσω κάποια παραδείγματα για να δω στην πράξη τις λειτουργίες και έπειτα να συνεχίσω με την εισαγωγή του μοντέλου που είχα στο μυαλό μου. Για την επίλυση του

προβλήματος έπρεπε αρχικά να λάβω δεδομένα από τις εικόνες που λάμβανα ανά πολύ μικρά χρονικά διαστήματα. Έτσι όταν λάμβανα μία εικόνα την μετέτρεπα κατάλληλα με σκοπό να λάβω μόνο τα pixels που περιείχαν το drone που έπρεπε να κυνηγήσω ή αλλιώς "το ποντίκι". Το εύκολο κομμάτι ήταν πως το ποντίκι ήταν πλήρως κόκκινο στην προσομοίωση, ενώ το περιβάλλον δεν είχε καθόλου κόκκινο και έτσι μπορούσα να την μετατρέψω εύκολα. Αν δεν υπήρχε καθόλου κόκκινο στην εικόνα μου άλλαζα κάμερα και περιέστρεφα το δικό μου drone ή αλλιώς "γάτα" μέχρι να βρω. Όσον αφορά το κυνήγι του ποντικιού μετρούσα τα κόκκινα pixels στην εικόνα με σκοπό να καταλάβω πόσο κοντά ή μακριά βρίσκεται το ποντίκι σε σχέση με τη γάτα. Με αυτό το τρόπο κινούσα τη γάτα γρήγορα ή αργά ώστε να μένω κοντά στο ποντίκι χωρίς να το ακουμπάω. Θα μπορούσαμε να πούμε πως η ταχύτητα της γάτας είναι αντιστρόφως ανάλογη του ποσού των κόκκινων pixels στην εικόνα. Εν συνεχεία, υπολόγιζα ένα μέσο όρο κατακόρυφα και οριζόντια με σκοπό να καταλάβω ποιο είναι το κέντρο του ποντικιού σε σχέση με το κέντρο της εικόνας μου. Με αυτό τον τρόπο μπορούσα να υπολογίσω αν η γάτα έπρεπε να κινηθεί προς τα πάνω, προς τα κάτω, αριστερά ή δεξιά. Για το κυνήγι, κινούσα τη γάτα και οριζόντια αλλά ταυτόχρονα την περιέστρεφα κιόλας για να πετύχω να πλησιάσω όσο το δυνατόν περισσότερο. Όσον αφορά την αποφυγή σύγκρουσης, όταν τα κόκκινα pixels ήταν πολλά στην εικόνα το ποντίκι ήταν αρκετά κοντά άρα η γάτα έπρεπε να μην κινηθεί ή και να απομακρυνθεί σε περίπτωση που το ποντίκι την πλησίαζε. Αφού έχω ήδη ορίσει την ταχύτητα με τον τρόπο που περιέγραψα προηγουμένως, έμενε να απομακρύνεται η γάτα. Όταν λοιπόν το ποντίκι πήγαινε προς τη γάτα, η τελευταία φρόντιζε να κρατήσει το ποντίκι στο πλάνο της κάμεράς της με την κατάλληλη στροφή αλλά η ταχύτητά της ήταν αρνητική ώστε να κινείται όπισθεν και επίσης υπήρχε μία αντίθετη περιστροφή και άνοδος ανάλογα με το που βρισκόταν το ποντίκι. Δηλαδή αν το ποντίκι την πλησίαζε από κάτω αριστερά, η γάτα θα κινούταν με όπισθεν, δεξιά και προς τα πάνω με σκοπό να το αποφύγει, πάντα όμως κρατώντας το στο πλάνο της για να μπορεί να το κυνηγήσει μόλις απομακρυνθεί και πάλι.»

## Κεφάλαιο 5: Διεξαγωγή πειράματος στους ρομποτικούς προσομοιωτές

### 5.1 Εισαγωγή

Για να γίνει σωστή και αντικειμενική σύγκριση της απόδοσης λογισμικών, είναι σκόπιμο να γίνει καταγραφή κάποιων στατιστικών μετρήσεων, κατά την διάρκεια εκτέλεσης των λογισμικών και στη συνέχεια να γίνει σύγκριση των μετρήσεων αυτών μεταξύ τους. Μερικά χαρακτηριστικά στα οποία θα πρέπει να δίνεται προσοχή, καθώς επηρεάζουν την απόδοση ενός λογισμικού, είναι η ταχύτητα εκτέλεσης, η δέσμευση επεξεργαστικής ισχύς, η χρήση μνήμης κλπ.

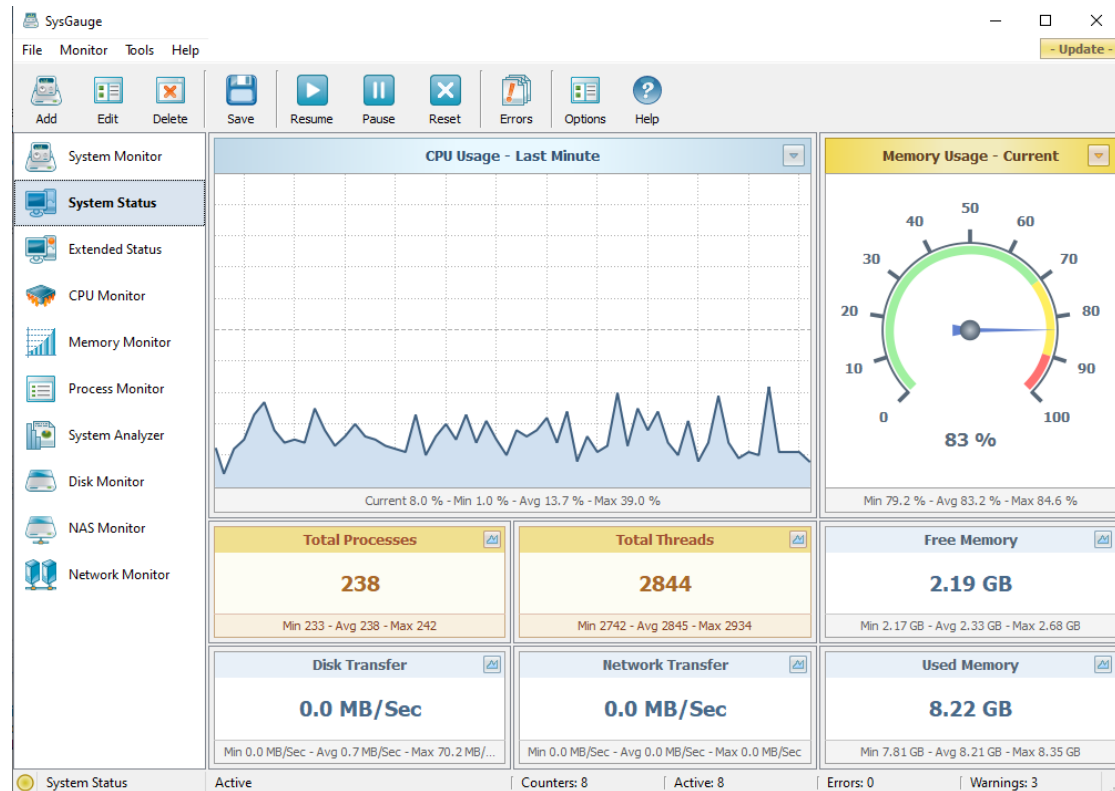
Στο συγκεκριμένο κεφάλαιο θα γίνει διεξαγωγή ενός πειράματος στους δώδεκα ρομποτικούς προσομοιωτές και λογισμικά που παρουσιάστηκαν στο Κεφάλαιο 2 της συγκεκριμένης εργασίας. Το πείραμα θα διεξαχθεί με χρήση του λογισμικού SysGauge.

### 5.2 Περιγραφή πειράματος

Το πείραμα που θα διεξαχθεί στους προσομοιωτές θα έχει να κάνει με την προσομοίωση ενός συγκεκριμένου ίδιου σεναρίου στον κάθε προσομοιωτή, στο οποίο σενάριο θα υπάρχει ένα εικονικό περιβάλλον με δύο έως πέντε εμπόδια και θα υπάρχουν περίπου τέσσερα ρομπότ με δύο ρόδες το καθένα (two-wheeled robots). Τα πιο συνηθισμένα ρομποτικά μοντέλα με δύο ρόδες είναι τα Pioneer 3-DX ή Pioneer 3-AT. Η προσομοίωση θα εκτελείται για ένα λεπτό ακριβώς. Στο διάστημα αυτό της προσομοίωσης, το λογισμικό SysGauge θα εκτελείται και θα κρατάει στατιστικές μετρήσεις για κάποια συγκεκριμένα χαρακτηριστικά που θα αναλυθούν παρακάτω. Τέλος, το μηχάνημα το οποίο θα φιλοξενεί την διεξαγωγή του πειράματος διαθέτει επεξεργαστή Intel Core i7 4700MQ 2.40GHz, μνήμη RAM 8.00GB Dual-Channel DDR3 798MHz και αποθηκευτικό δίσκο HDD 698GB Western Digital WDC.

Το SysGauge [44] είναι ένα ειδικό λογισμικό παρακολούθησης συστημάτων και απόδοσης σχετικά με την χρήση επεξεργαστικής ισχύς, την χρήση μνήμης του συστήματος, τον ρυθμό μεταφοράς δικτύου, την απόδοση λειτουργικού συστήματος, την χρήση αποθηκευτικού χώρου, την δραστηριότητα εγγραφής και διαβάσματος του

δίσκου καθώς και τον ρυθμό μεταφοράς εγγραφής και διαβάσματος του δίσκου. Στην παρακάτω εικόνα φαίνεται η καρτέλα System Status του SysGauge που δείχνει την κατάσταση του συστήματος που παρακολουθεί:



Εικόνα 16: Παρακολούθηση συστήματος με την χρήση του SysGauge

Όπως φαίνεται και στην Εικόνα 15, το SysGauge διαθέτει καρτέλες που ουσιαστικά κατηγοριοποιεί τις χρήσεις που μπορεί να κάνει ο κάθε χρήστης. Αν για παράδειγμα ένας χρήστης επιθυμεί την παρακολούθηση της κατάστασης του συστήματος, θα ανοίξει την καρτέλα System Monitor και System Status ενώ αν θέλει την παρακολούθηση χρήσης της επεξεργαστικής ισχύς του συστήματος, θα χρησιμοποιήσει την καρτέλα CPU Monitor.

Για το πείραμα, το SysGauge διαθέτει μια λειτουργία με την οποία μπορεί και συγκρίνει σε πραγματικό χρόνο την απόδοση λογισμικών, ή αλλιώς διαδικασιών (processes) που τρέχουν στο σύστημα. Η σύγκριση αυτή γίνεται για συγκεκριμένα χαρακτηριστικά.

Τα χαρακτηριστικά με τα οποία το SysGauge πραγματοποιεί την σύγκριση αναλύονται παρακάτω:

- Χρήση επεξεργαστικής ισχύς: εμφανίζει το ποσοστό που χρησιμοποιεί το σύστημα για την εκτέλεση της διαδικασίας.
- Χρήση μνήμης: εμφανίζει σε Megabytes την ποσότητα μνήμης που χρησιμοποιεί η διαδικασία για την εκτέλεσή της.
- Μέτρηση αριθμών threads: εμφανίζει το πλήθος των threads που είναι ενεργά για την εκτέλεση της διαδικασίας. Γενικά, κάθε διαδικασία έχει έστω και ένα ενεργό thread, αλλά οι περισσότερες διαδικασίες χρησιμοποιούν περισσότερα.
- Μέτρηση αριθμών handles: εμφανίζει το πλήθος των handles που είναι ανοιχτά για την εκτέλεση της διαδικασίας.
- Ρυθμός εμφάνισης σφαλμάτων σελίδας (page fault): εμφανίζει τον ρυθμό εμφάνισης σφαλμάτων σελίδας. Το σφάλμα σελίδας είναι ένα είδος εξαίρεσης (exception) και πιο συγκεκριμένα μια παγίδευση του λειτουργικού συστήματος. Συμβαίνει όταν η Μονάδα Διαχείρισης Μνήμης του συστήματος διαπιστώνει πως μια σελίδα δεν αντιστοιχίζεται στην φυσική μνήμη και αναγκάζει την Κεντρική Μονάδα Επεξεργασίας να καταφύγει σε παγίδευση του λειτουργικού συστήματος. Ουσιαστικά, σελίδες είναι οι μονάδες σταθερού μεγέθους που διαιρούνται οι εικονικές διευθύνσεις.
- Χρήση μνήμης από την nonpaged pool: εμφανίζει σε Megabytes την μνήμη που χρησιμοποιήθηκε από την nonpaged pool για την εκτέλεση της διαδικασίας.
- Ρυθμός κίνησης δεδομένων: υπολογίζει και εμφανίζει τον ρυθμό δεδομένων ανά δευτερόλεπτο που η συγκεκριμένη διαδικασία διαβάζει και γράφει για Input/Output λειτουργίες.
- Ρυθμός διαβάσματος δεδομένων: υπολογίζει και εμφανίζει τον ρυθμό δεδομένων ανά δευτερόλεπτο που η συγκεκριμένη διαδικασία για Input/Output λειτουργίες.
- Ρυθμός εγγραφής δεδομένων: υπολογίζει και εμφανίζει τον ρυθμό δεδομένων ανά δευτερόλεπτο που η συγκεκριμένη διαδικασία γράφει για Input/Output λειτουργίες.

Από τους παραπάνω μετρητές, περισσότερο ενδιαφέρον παρουσιάζουν οι δύο πρώτοι, καθώς η απόδοση του κάθε προσομοιωτή εξαρτάται σε μεγάλο βαθμό από την επεξεργαστική ισχύ και το ποσοστό μνήμης που χρησιμοποιεί.

### 5.3 Αποτελέσματα πειράματος

Στους δύο παρακάτω πίνακες, φαίνονται συγκεντρωτικά οι μετρήσεις που έγιναν από το SysGauge για τον κάθε προσομοιωτή κατά την διεξαγωγή του πειράματος που περιγράφηκε προηγουμένως:

| Μετρητής                           | WeBots     | Simbad     | Microsoft Robotics Developer Studio | FlightGear |
|------------------------------------|------------|------------|-------------------------------------|------------|
| Χρήση επεξεργαστικής ισχύς         | 5.0%       | 0.4%       | 12.4%                               | 9.8%       |
| Χρήση μνήμης                       | 369.6 MB   | 171.MB     | 267.6 MB                            | 1.97 GB    |
| Μέτρηση αριθμών threads            | 21         | 59         | 80                                  | 21         |
| Μέτρηση αριθμών handles            | 5209       | 2970       | 1466                                | 2717       |
| Ρυθμός εμφάνισης σφαλμάτων σελίδας | 0 F/Sec    | 89 F/Sec   | 44482 F/Sec                         | 1385 F/Sec |
| Χρήση μνήμης από την nonpaged pool | 0.1 MB     | 0.0 MB     | 0.1 MB                              | 0.1 MB     |
| Ρυθμός κίνησης δεδομένων           | 0.0 MB/Sec | 0.0 MB/Sec | 0.0 MB/Sec                          | 0.3 MB/Sec |
| Ρυθμός διαβάσματος δεδομένων       | 0.0 MB/Sec | 0.0 MB/Sec | 0.0 MB/Sec                          | 0.3 MB/Sec |
| Ρυθμός εγγραφής δεδομένων          | 0.0 MB/Sec | 0.0 MB/Sec | 0.0 MB/Sec                          | 0.0 MB/Sec |

Πίνακας 1: Πίνακας μετρήσεων του SysGauge για τους ρομποτικούς προσομοιωτές (1/2)

| Μετρητής                   | MATLAB-Simulink | Gazebo   | X-Plane |
|----------------------------|-----------------|----------|---------|
| Χρήση επεξεργαστικής ισχύς | 14.0%           | 4.1%     | 11.1%   |
| Χρήση μνήμης               | 1.57 GB         | 323.7 MB | 1.17 GB |
| Μέτρηση αριθμών threads    | 164             | 20       | 32      |
| Μέτρηση αριθμών handles    | 48633           | 4133     | 9968    |

|                                    |             |            |            |
|------------------------------------|-------------|------------|------------|
| Ρυθμός εμφάνισης σφαλμάτων σελίδας | 38138 F/Sec | 0 F/Sec    | 7743 F/Sec |
| Χρήση μνήμης από την nonpaged pool | 0.4 MB      | 0.1 MB     | 0.1 MB     |
| Ρυθμός κίνησης δεδομένων           | 17.1 MB/Sec | 0.0 MB/Sec | 0.1 MB/Sec |
| Ρυθμός διαβάσματος δεδομένων       | 17.1 MB/Sec | 0.0 MB/Sec | 0.1 MB/Sec |
| Ρυθμός εγγραφής δεδομένων          | 2.8 MB/Sec  | 0.0 MB/Sec | 0.0 MB/Sec |

Πίνακας 2: Πίνακας μετρήσεων του SysGauge για τους ρομποτικούς προσομοιωτές (2/2)

Παρακάτω φαίνονται μερικές ενδεικτικές εικόνες από τις μετρήσεις που συγκεντρώθηκαν για τους ορισμένους προσομοιωτές από το SysGauge:

| Counter                    | Webots     |
|----------------------------|------------|
| Process CPU Usage          | 5.0 %      |
| Process Memory Used        | 369.6 MB   |
| Process Thread Count       | 21         |
| Process Handle Count       | 5209       |
| Process Page Fault Rate    | 0 F/Sec    |
| Process Nonpaged Pool Used | 0.1 MB     |
| Process Data Rate          | 0.0 MB/Sec |
| Process Read Rate          | 0.0 MB/Sec |
| Process Write Rate         | 0.0 MB/Sec |

Εικόνα 17: Μετρήσεις SysGauge για τον προσομοιωτή Webots

### DssHost32 - Process Counters

| Counter                    | Average     |
|----------------------------|-------------|
| Process CPU Usage          | 12.4 %      |
| Process Memory Used        | 267.6 MB    |
| Process Thread Count       | 80          |
| Process Handle Count       | 1466        |
| Process Page Fault Rate    | 44482 F/Sec |
| Process Nonpaged Pool Used | 0.1 MB      |
| Process Data Rate          | 0.0 MB/Sec  |
| Process Read Rate          | 0.0 MB/Sec  |
| Process Write Rate         | 0.0 MB/Sec  |

Εικόνα 18: Μετρήσεις SysGauge για τον προσομοιωτή Microsoft Robotics Developer Studio

### MATLAB - Process Counters

| Counter                    | Average     |
|----------------------------|-------------|
| Process CPU Usage          | 14.0 %      |
| Process Memory Used        | 1.57 GB     |
| Process Thread Count       | 164         |
| Process Handle Count       | 48633       |
| Process Page Fault Rate    | 38139 F/Sec |
| Process Nonpaged Pool Used | 0.4 MB      |
| Process Data Rate          | 17.1 MB/Sec |
| Process Read Rate          | 17.1 MB/Sec |
| Process Write Rate         | 2.8 MB/Sec  |

Εικόνα 19: Μετρήσεις SysGauge για τον προσομοιωτή MATLAB - Simulink

### X-Plane - Process Counters

| Counter                    | Average    |
|----------------------------|------------|
| Process CPU Usage          | 11.1 %     |
| Process Memory Used        | 1.17 GB    |
| Process Thread Count       | 32         |
| Process Handle Count       | 9968       |
| Process Page Fault Rate    | 7743 F/Sec |
| Process Nonpaged Pool Used | 0.1 MB     |
| Process Data Rate          | 0.1 MB/Sec |
| Process Read Rate          | 0.1 MB/Sec |
| Process Write Rate         | 0.0 MB/Sec |

Εικόνα 20: Μετρήσεις SysGauge για τον προσομοιωτή X-Plane

Όπως είναι φανερό, κάποιοι από τους ρομποτικούς προσομοιωτές που περιγράφηκαν στο Κεφάλαιο 2 λείπουν στους πίνακες σύγκρισης και πιο συγκεκριμένα οι USARSim, Breve, Player/Stage, UberSim και Actin. Δυστυχώς, υπήρξαν κάποιες τεχνικές δυσκολίες με τους συγκεκριμένους προσομοιωτές και δεν έγινε εφικτή η σύγκρισή τους με τους υπόλοιπους προσομοιωτές.

Πιο συγκεκριμένα, οι προσομοιωτές USARSim, Breve και UberSim δεν ήταν διαθέσιμοι πουθενά στο διαδίκτυο και δεν βρέθηκαν σε κανένα διάστημα την στιγμή εκπόνησης της συγκεκριμένης εργασίας. Το πιο πιθανόν είναι να έχει σταματήσει η παροχή υποστήριξης και ανάπτυξης των λογισμικών αυτών και αυτός να είναι ο λόγος που δεν ήταν διαθέσιμοι. Άξιζαν όμως να αναφερθούν για την σπουδαιότητά τους. Από την άλλη, ο Player και Stage δεν υποστηρίζουν τα Windows και δεν θα ήταν αντικειμενική η σύγκρισή τους με τους υπόλοιπους προσομοιωτές από άλλο λειτουργικό σύστημα και μηχανήμα. Για αυτόν το λόγο, θεωρήθηκε καλύτερο να μην συγκριθούν με τους υπόλοιπους προσομοιωτές. Τέλος, ο προσομοιωτής Actin είναι εμπορικός και δεν υπήρχε στο διαδίκτυο καμία διαθέσιμη έκδοσή του, ούτε κάποια δωρεάν Demo. Σημειώνεται ότι, ο Γεώργιος Εξίζογλου ήρθε σε επαφή με την εταιρία υλοποίησης του προσομοιωτή, Energid, αλλά δεν δέχτηκαν να συνεργαστούν και να αποστείλουν κάποια δοκιμαστική έκδοση ή κάποιο demo για την διεξαγωγή του πειράματος. Λόγω όμως ότι είναι αρκετά σύγχρονο λογισμικό, με μεγάλη χρήση σε σημαντικούς τομείς, άξιζε να αναφερθεί και να περιγραφούν τα χαρακτηριστικά του.

## 5.4 Συμπεράσματα πειράματος

Ορισμένοι ρομποτικοί προσομοιωτές που περιγράφηκαν στο Κεφάλαιο 2 της εργασίας, συγκρίθηκαν σχετικά με την απόδοσή τους σε ορισμένες μετρήσεις με την βοήθεια του λογισμικού SysGauge. Η σύγκριση αυτή έγινε με την διεξαγωγή ενός πειράματος, σύμφωνα με το οποίο οι ρομποτικοί προσομοιωτές θα έπρεπε να πραγματοποιήσουν ένα συγκεκριμένο σενάριο προσομοίωσης για ένα λεπτό. Στο τέλος του χρόνου αυτού, συγκεντρώνονταν οι μετρήσεις, με το SysGauge και τέλος έγινε η σύγκριση. Σημειώνεται ότι, κατά την διάρκεια της προσομοίωσης, ο κάθε προσομοιωτής εκτελούνταν μόνος του και όχι ταυτόχρονα με άλλον. Συγκεντρώνοντας τα αποτελέσματα, παρατηρείται ότι ο προσομοιωτής MATLAB - Simulink χρησιμοποίησε την μεγαλύτερη επεξεργαστική ισχύ του συστήματος. Έπειτα, οι προσομοιωτές πτήσεων FlightGear και X-Plane αλλά και ο προσομοιωτής Microsoft Robotics Developer Studio σημείωσαν την μεγαλύτερη χρήση μνήμης του συστήματος, συγκριτικά με τους υπόλοιπους προσομοιωτές και συγκεκριμένα ο FlightGear χρησιμοποίησε την περισσότερη. Για την μέτρηση αριθμών handles, ο MATLAB – Simulink είχε την μεγαλύτερη μέτρηση ενώ για τον ρυθμό εμφάνισης

σφαλμάτων σελίδας ο MRDS ήταν πρώτος. Στην χρήση μνήμης από την nonpaged pool αλλά και για τον ρυθμό κίνησης δεδομένων, ο MATLAB – Simulink έκανε είχε την μεγαλύτερη τιμή. Τέλος, για τον ρυθμό διαβάσματος δεδομένων, όσο και για τον ρυθμό εγγραφής δεδομένων, την μεγαλύτερη τιμή σημείωσε ο ρομποτικός προσομοιωτής MATLAB – Simulink.

## Κεφάλαιο 6: Αξιολόγηση Ρομποτικών Προσομοιωτών και Παρουσίαση Αποτελεσμάτων

### 6.1 Εισαγωγή

Στο Κεφάλαιο 5, έγινε σύγκριση και αξιολόγηση των προσομοιωτών όσον αφορά την απόδοσή τους. Στο συγκεκριμένο κεφάλαιο, θα γίνει αξιολόγηση των ρομποτικών προσομοιωτών, που περιγράφηκαν στο Κεφάλαιο 2, με βάση κάποια συγκεκριμένα χαρακτηριστικά, που ουσιαστικά θα αποτελέσουν τα κριτήρια της αξιολόγησης αυτής.

### 6.2 Ανάλυση Κριτηρίων

Οι ρομποτικοί προσομοιωτές που αναλύθηκαν, παρουσίασαν ορισμένα χαρακτηριστικά. Για την συγκεκριμένη αξιολόγηση, έγινε επιλογή των πιο αντιπροσωπευτικών, κοινών και χρήσιμων χαρακτηριστικών που χρειάζεται να έχει ένας ρομποτικός προσομοιωτής για τις σημερινές ανάγκες.

Τα κριτήρια με τα οποία θα γίνει η αξιολόγηση των ρομποτικών προσομοιωτών είναι τα εξής:

- **Κόστος:** δείχνει αν ο ρομποτικός προσομοιωτής είναι εμπορικός και κοστίζει για να γίνει χρήση του ή είναι ανοιχτού λογισμικού και είναι δωρεάν.
- **Λειτουργικό σύστημα:** φαίνονται τα λειτουργικά συστήματα που υποστηρίζει ο ρομποτικός προσομοιωτής. Στην συγκεκριμένη αξιολόγηση θα εξετάσουμε αν ο προσομοιωτής υποστηρίζει τα τρία δημοφιλέστερα Windows, Linux και OS X.
- **3D απεικόνιση:** δείχνει αν ο ρομποτικός προσομοιωτής διαθέτει τρισδιάστατη απεικόνιση ή όχι. Όσο καλύτερη είναι η απεικόνιση, τόσο μεγαλύτερη πιστότητα έχει.
- **Open-source:** δείχνει αν ο προσομοιωτής είναι ανοιχτού λογισμικού ή όχι. Συνήθως αν είναι, είναι και δωρεάν για χρήση.

- Πολυπρακτορική προσομοίωση: δείχνει αν ο προσομοιωτής μπορεί να εκτελέσει προσομοίωση με πολλαπλά ρομπότ στο ίδιο περιβάλλον προσομοίωσης.
- Εντοπισμός συγκρούσεων: δείχνει αν ο ρομποτικός προσομοιωτής διαθέτει την δυνατότητα εντοπισμού συγκρούσεων των ρομπότ είτε μεταξύ τους είτε με εμπόδια του περιβάλλοντος της προσομοίωσης.

### 6.3 Αποτελέσματα και Σχολιασμός

Στον παρακάτω συγκεντρωτικό πίνακα, φαίνονται όλοι οι ρομποτικοί προσομοιωτές που περιγράφηκαν στην εργασία αυτή με τα χαρακτηριστικά που διαθέτουν. Η αξιολόγηση των προσομοιωτών γίνεται με βάση αυτά τα χαρακτηριστικά που εν τέλει γίνονται τα κριτήρια αυτής της αξιολόγησης. Οι παρατηρήσεις και τα συμπεράσματα που γίνονται διακριτά είναι ότι αρχικά, υπάρχουν αρκετοί ρομποτικοί προσομοιωτές οι οποίοι είναι ανοιχτού λογισμικού και συνεπώς δωρεάν. Επομένως, μπορεί κανείς να πραγματοποιήσει μια προσομοίωση για διάφορους σκοπούς χωρίς να χρειάζεται να ακριβοπληρώσει το λογισμικό. Έπειτα, φαίνεται ότι τα Linux είναι το πιο δημοφιλές λειτουργικό σύστημα που υποστηρίζουν οι ρομποτικοί προσομοιωτές και το Mac OS X το λιγότερο δημοφιλές. Πλέον, οι ρομποτικοί προσομοιωτές φαίνεται ότι οι περισσότεροι διαθέτουν τρισδιάστατη απεικόνιση, καθώς στην συγκεκριμένη αξιολόγηση με τους προσομοιωτές που επιλέχθηκαν σχεδόν όλοι διαθέτουν τρισδιάστατη απεικόνιση εκτός από Player/Stage. Όσον αφορά το open-source κομμάτι, φαίνεται ότι υπάρχει ένα πλήθος προσομοιωτών που είναι ανοιχτού κώδικα και ορισμένοι που δεν είναι. Ισχύει όμως ότι οι εμπορικοί προσομοιωτές είναι πιο επαγγελματικού χαρακτήρα και διαθέτουν περισσότερες λειτουργίες σε σχέση με τους ανοιχτού κώδικα προσομοιωτές. Επίσης, για την πολυπρακτορική προσομοίωση, παρατηρείται ότι υπάρχουν προσομοιωτές που διαθέτουν αυτήν την δυνατότητα και προσομοιωτές που δεν την διαθέτουν. Τέλος, παρατηρείται το ίδιο και για τον εντοπισμό συγκρούσεων, που άλλοι προσομοιωτές διαθέτουν αυτήν την δυνατότητα και άλλοι όχι.

Σημειώνεται ότι στην στήλη Λειτουργικό σύστημα, το W σημαίνει Windows, το L σημαίνει Linux και το M σημαίνει Mac OS X.

| <i>Προσομοιωτής</i>                        | <i>Κόστος</i> | <i>Λειτουργικό σύστημα</i> | <i>3D απεικόνιση</i> | <i>Open-Source</i> | <i>Πολυπρακτορική προσομοίωση</i> | <i>Εντοπισμός συγκρούσεων</i> |
|--------------------------------------------|---------------|----------------------------|----------------------|--------------------|-----------------------------------|-------------------------------|
| <i>WeBots</i>                              | Μεσαίο        | WLM                        | Ναι                  | Όχι                | Ναι                               | Ναι                           |
| <i>Simbad</i>                              | Δωρεάν        | WLM                        | Ναι                  | Ναι                | Ναι                               | Ναι                           |
| <i>USARSim</i>                             | Δωρεάν        | WLM                        | Ναι                  | Ναι                | Ναι                               | Όχι                           |
| <i>Microsoft Robotics Developer Studio</i> | Χαμηλό        | W                          | Ναι                  | Όχι                | Ναι                               | Ναι                           |
| <i>Breve</i>                               | Δωρεάν        | WLM                        | Ναι                  | Ναι                | Ναι                               | Ναι                           |
| <i>FlightGear</i>                          | Δωρεάν        | WLM                        | Ναι                  | Ναι                | Όχι                               | Όχι                           |
| <i>MATLAB</i>                              | Υψηλό         | WLM                        | Ναι                  | Όχι                | Ναι                               | Όχι                           |
| <i>Gazebo</i>                              | Δωρεάν        | WLM                        | Ναι                  | Ναι                | Ναι                               | Ναι                           |
| <i>Player/Stage</i>                        | Δωρεάν        | LM                         | Όχι                  | Ναι                | Ναι                               | Ναι                           |
| <i>UberSim</i>                             | Δωρεάν        | WL                         | Ναι                  | Ναι                | Όχι                               | Ναι                           |
| <i>Actin</i>                               | Μεσαίο        | WL                         | Ναι                  | Όχι                | Ναι                               | Ναι                           |
| <i>X-Plane</i>                             | Μεσαίο        | WLM                        | Ναι                  | Όχι                | Όχι                               | Όχι                           |

Πίνακας 3: Αναλυτικός πίνακας σύγκρισης ρομποτικών προσομοιωτών

## Κεφάλαιο 7: Συμπεράσματα και Προτάσεις

### 7.1 Συμπεράσματα

Σκοπός της συγκεκριμένης εργασίας είναι η μελέτη και η σύγκριση ρομποτικών προσομοιωτών και λογισμικών. Αρχικά, αναφέρθηκε ότι ρομποτικός προσομοιωτής είναι ένα λογισμικό ή πλατφόρμα που πραγματοποιεί προσομοίωση. Προσομοίωση είναι μια κατά προσέγγιση μίμηση μιας λειτουργίας μιας διαδικασίας ή ενός συστήματος μέσα στο χρόνο με τη βοήθεια ενός υπολογιστή. Η ρομποτική προσομοίωση παρουσιάζει οφέλη, όπως για παράδειγμα την μείωση κόστους υλοποίησης ρομποτικών συστημάτων και του κινδύνου σε περίπτωση αποτυχίας. Παράλληλα όμως και κάποια μειονεκτήματα, όπως την δυσκολία χειρισμού των προσομοιωτών και το κόστος που μπορεί να έχει η απόκτηση ενός ρομποτικού προσομοιωτή, γιατί συνήθως αν είναι επί πληρωμή, κοστίζει αρκετά.

Στην συνέχεια έγινε περιγραφή και ανάλυση δώδεκα διάσημων και αξιοσημείων ρομποτικών προσομοιωτών και λογισμικών. Περιγράφηκαν τα χαρακτηριστικά του κάθε προσομοιωτή, ποια ρομπότ υποστηρίζουν και ποια λειτουργικά συστήματα, ποια τα πλεονεκτήματα που διαθέτουν και ποια τα μειονεκτήματα.

Έπειτα, αναφέρθηκε η χρήση ρομποτικών προσομοιωτών στους τομείς της εκπαίδευσης, της γεωργίας και της ιατρικής. Και οι τρεις αυτοί τομείς, έχουν υποβοηθηθεί σημαντικά από την ρομποτική και έχουν εξελιχθεί σε σημαντικό βαθμό. Αναφέρθηκαν στον κάθε τομέα ρομποτικά συστήματα και τεχνικές που έχουν αναπτυχθεί με την βοήθεια της ρομποτικής. Αν και διευκολύνουν και αυτοματοποιούν καταστάσεις και εργασίες, χρειάζονται ιδιαίτερη προσοχή όχι μόνο κατά τον χειρισμό τους αλλά και κατά την υλοποίησή τους. Και εκεί έρχεται η προσομοίωση, η οποία κάνει εφικτή την υλοποίηση τέτοιων σύνθετων ρομποτικών συστημάτων. Με γρήγορο και ασφαλή τρόπο, μπορεί να μοντελοποιήσει και να υλοποιήσει το ρομποτικό σύστημα και να είναι έτοιμο για χρήση.

Στη συνέχεια, παρουσιάστηκε η διεξαγωγή ενός πειράματος, το οποίο είχε ως σκοπό την σύγκριση των ρομποτικών προσομοιωτών που παρουσιάστηκαν στην εργασία με βάση την απόδοσή τους. Το πείραμα είχε να κάνει με την προσομοίωση ενός σεναρίου όπου θα υπήρχε ένα μικρό περιβάλλον και δύο έως πέντε μικρά

ρομπότ που το καθένα έχει δύο ρόδες. Με την ολοκλήρωση του πειράματος αυτού, παρατηρήθηκε στα αποτελέσματα ότι τις μεγαλύτερες μετρήσεις στα χαρακτηριστικά που γινόταν οι σύγκριση των προσομοιωτών τις σημείωσε ο προσομοιωτής MATLAB – Simulink.

Εκτός από την διεξαγωγή του πειράματος, πραγματοποιήθηκε και μια αξιολόγηση των ρομποτικών προσομοιωτών με βάση κάποια κοινά και σημαντικά χαρακτηριστικά. Παρατηρήθηκε ότι το λειτουργικό σύστημα που υποστηρίζουν περισσότερο οι ρομποτικοί προσομοιωτές είναι τα Linux και το λιγότερο τα Mac OS X, υπάρχουν αρκετοί προσομοιωτές που είναι ανοιχτού λογισμικού και συνεπώς δωρεάν, σχεδόν όλοι προσομοιωτές διαθέτουν τρισδιάστατη απεικόνιση και επίσης ορισμένοι διαθέτουν πολυπρακτορική προσομοίωση και δυνατότητα εντοπισμού συγκρούσεων και άλλοι δεν διαθέτουν.

Τέλος, με την ολοκλήρωση της εργασίας παρατηρείται ότι αν και υπάρχουν ανοιχτού λογισμικού ρομποτικοί προσομοιωτές, οι περισσότεροι έχουν δύσκολο και δυσνόητο χειρισμό και εγκατάσταση και δυσκολεύουν πολύ νέους χρήστες να τους χρησιμοποιήσουν αποξενώνοντάς τους. Πλέον, μόνο οι εμπορικοί προσομοιωτές έχουν προχωρήσει και αναβαθμιστεί παρέχοντας οργανωμένη καθοδήγηση, εκπαίδευση, ενημέρωση, εγχειρίδιο χειρισμού κλπ. Οι περισσότεροι προσομοιωτές διαθέτουν τα πιο σημαντικά χαρακτηριστικά, όπως πολυπρακτορική προσομοίωση και δυνατότητα εντοπισμού συγκρούσεων αλλά παίζει ρόλο και το γραφικό κομμάτι και οι διεπαφές που διαθέτει ο κάθε προσομοιωτής και σε αυτό το κομμάτι νικούν οι εμπορικοί προσομοιωτές, καθώς είναι πιο όμορφοι και κατανοητοί. Γενικά, η κοινωνία δεν έχει αγκαλιάσει ακόμα την ρομποτική προσομοίωση όπως θα έπρεπε και αυτό συμβαίνει είτε λόγω της δυσλειτουργίας που προσφέρουν οι περισσότεροι, είτε λόγω του μεγάλου κόστους που απαιτείται για να αποκτηθούν. Η χρήση τους παρουσιάζεται και περιορίζεται κυρίως σε μεγάλες ιδιωτικές εταιρείες. Θα έρθει όμως η στιγμή που η ρομποτική προσομοίωση θα είναι το απαραίτητο εργαλείο κάθε προγραμματιστή και ερευνητή.

## 7.2 Προτάσεις

Με την ολοκλήρωση της συγκεκριμένης εργασίας, αν κάποιος πρόκειται να εκπονήσει μια εργασία με παρόμοιο θέμα, μπορεί να λάβει υπόψιν ορισμένες προτάσεις. Αρχικά, θα μπορούσε να επιλέξει αν όχι περισσότερους ρομποτικούς προσομοιωτές, τουλάχιστον διαφορετικούς. Σίγουρα σε μια έρευνα, όσα περισσότερα είναι τα αντικείμενα στα οποία γίνεται μια μελέτη και σύγκριση, γίνεται πιο ολοκληρωμένη. Επίσης, αν είναι εφικτό, θα ήταν καλό να γίνει επιλογή πιο σύγχρονων ρομποτικών προσομοιωτών, για να είναι διαθέσιμοι, να υπάρχει

υποστήριξη από την εταιρεία υλοποίησης του λογισμικού και να υπάρχει ένα εγχειρίδιο οδηγιών, διότι οι προσομοιωτές είναι σύνθετα λογισμικά με πολλές λειτουργίες και συνήθως δύσκολο χειρισμό, αν δεν υπάρχει βοήθεια από κάποιον. Επίσης, μια σημαντική πρόταση θα ήταν η αλλαγή του πειράματος. Θα υπήρχε μεγάλο ενδιαφέρον να διεξαχθεί ένα διαφορετικό πείραμα στους ρομποτικούς προσομοιωτές και να ελεγχθούν σε κάτι άλλο. Μπορεί το κεντρικό ζητούμενο να ήταν και πάλι η αναζήτηση του καλύτερου προσομοιωτή όσον αφορά την απόδοση, αλλά θα υπήρχε ένα άλλο σενάριο στο οποίο μπορεί να προέκυπτε άλλος προσομοιωτής νικητής. Το λογισμικό SysGauge, αν και πάρα πολύ εύχρηστο και δυνατό λογισμικό, μπορεί να παραληφθεί και οι μετρήσεις να αποκτηθούν με διαφορετικό τρόπο. Όσον αφορά την αξιολόγηση των ρομποτικών προσομοιωτών, είναι εφικτό να γίνει αξιολόγησή τους με περισσότερα ή και διαφορετικά χαρακτηριστικά. Στη συγκεκριμένη εργασία, θεωρήθηκε ότι αυτά αποτελούν τα πιο σημαντικά. Τέλος, μια πρόταση που θα παρουσίαζε ενδιαφέρον είναι η διεξαγωγή προσομοίωσης ενός ρομποτικού συστήματος και χρήσης αυτού στον τομέα που το αφορά. Για παράδειγμα, θα είχε τεράστιο ενδιαφέρον, η προσομοίωση ενός ρομποτικού βραχίονα και η χρήση αυτού σε κάποιο χώρο που χρειάζεται να υπάρχει, όπως για παράδειγμα σε κάποιο εργαστήριο ή σε κάποιο θερμοκήπιο για την συλλογή συγκομιδής. Ωστόσο, η συγκεκριμένη πρόταση δεν προτείνεται να συνδυαστεί με το αντικείμενο της συγκεκριμένης εργασίας, διότι δεν είναι σκοπός της εργασίας η προσομοίωση ρομποτικών συστημάτων αλλά η μελέτη και η σύγκρισή τους με βάση κάποια χαρακτηριστικά τους. Αν βέβαια είναι εφικτό να συνδυαστούν, θα αποτελούσαν έναν καινοτόμο και ολοκληρωμένο συνδυασμό καθώς θα κάλυπταν όλη την διαδικασία της προσομοίωσης, από την υλοποίηση του ρομποτικού συστήματος μέχρι και την χρήση του στην πραγματική ζωή.

## Βιβλιογραφία

- [1] Žlajpah, Leon. "Simulation in robotics." *Mathematics and Computers in Simulation* 79.4 (2008): 879-897.
- [2] N. Koenig and A. Howard. Design and Use Paradigms for Gazebo, An Open-Source Multi-Robot Simulator. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2149–2154, 2004.
- [3] N. Michael, J. Fink, and V. Kumar. Experimental testbed for large multirobot teams. *IEEE Robotics Automation Magazine*, 15(1): 53–61, March 2008.
- [4] Player/Stage/Gazebo [Online]. Διαθέσιμο: <http://playerstage.sourceforge.net/>
- [5] J. Kramer and M. Scheutz. Development environments for autonomous mobile robots: A survey. *Autonomous Robots*, 22:101–132, February 2007.
- [6] Νικόλαος Φ. Μιχαηλίδης, Προσομοίωση Ρομποτικών Εφαρμογών με χρήση του λογισμικού “V-Rep”, Διατριβή που υπεβλήθη για την μερική ικανοποίηση των απαιτήσεων για την απόκτηση Μεταπτυχιακού Διπλώματος Ειδίκευσης, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, 2014.
- [7] Kamal, Raj. *Embedded systems: architecture, programming and design*. Tata McGraw-Hill Education, 2011.

- [8] Κολιπέτσας, Κωνσταντίνος Ε. *Προσομοίωση ρομποτικών συστημάτων στη γεωργία*. MS thesis. 2011.
- [9] Wikipedia Contributors. "Simulation." *Wikipedia*, Wikimedia Foundation, 26 Mar. 2019, [en.wikipedia.org/wiki/Simulation](http://en.wikipedia.org/wiki/Simulation).
- [10] Wikipedia Contributors. "Robotics Simulator." *Wikipedia*, Wikimedia Foundation, 27 May 2019, [en.wikipedia.org/wiki/Robotics\\_simulator](http://en.wikipedia.org/wiki/Robotics_simulator).
- [11] Webots [Online]. Διαθέσιμο: <http://www.cyberbotics.com/products/webots/>
- [12] Olivier, Michel. "Cyberbotics LTD-webotstm: Professional mobile robot simulation." *International Journal of Advanced Robotic Systems* 1.1 (2004): 40-43.
- [13] Wikipedia Contributors. (2019, May 12), from Wikipedia website: <https://en.wikipedia.org/wiki/Webots>.
- [14] Simbad Project Home [Online]. Διαθέσιμο: <http://simbad.sourceforge.net/index.php>
- [15] Hugues, Louis, and Nicolas Bredeche. "Simbad: an autonomous robot simulation package for education and research." *International Conference on Simulation of Adaptive Behavior*. Springer, Berlin, Heidelberg, 2006.
- [16] USARSim [Online]. Διαθέσιμο: <https://sourceforge.net/projects/usarsim/>

- [17] Harris, Adam, and James M. Conrad. "Survey of popular robotics simulators, frameworks, and toolkits." *2011 Proceedings of IEEE Southeastcon*. IEEE, 2011.
- [18] Carpin, Stefano, et al. "USARSim: a robot simulator for research and education." *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, 2007.
- [19] Epic games, "Unreal engine," [www.epicgames.com](http://www.epicgames.com) .
- [20] Gamebots [Online]. Διαθέσιμο: <http://gamebots.sourceforge.net/>
- [21] Microsoft Robotics Developer Studio [Online]. Διαθέσιμο: <https://www.microsoft.com/en-us/download/details.aspx?id=29081>
- [22] Johns, Kyle, and Trevor Taylor. *Professional microsoft robotics developer studio*. John Wiley & Sons, 2009.
- [23] Jackson, Jared. "Microsoft robotics studio: A technical introduction." *IEEE robotics & automation magazine* 14.4 (2007): 82-87.
- [24] Klein, Jon. "Breve: a 3d environment for the simulation of decentralized systems and artificial life." *Proceedings of the eighth international conference on Artificial life*. 2003.
- [25] Perry, Alexander R. "The FlightGear flight simulator." *Proceedings of the USENIX Annual Technical Conference*. 2004.

- [26] Basler, Michael, et al. "The FlightGear Manual." *The FlightGear Manual* (2010).
- [27] "MATLAB Simulink." *www.tutorialspoint.com*. N. p., 2019. Web. 31 July 2019.
- [28] Van Schijndel, AWM van, and M. H. De Wit. "Advanced simulation of building systems and control with simulink." *Proc. 8th Int. IBPSA Conf.* 2003.
- [29] Gerkey, Brian, Richard T. Vaughan, and Andrew Howard. "The player/stage project: Tools for multi-robot and distributed sensor systems." *Proceedings of the 11th international conference on advanced robotics*. Vol. 1. 2003.
- [30] "Gazebo Simulator" [Online]. Διαθέσιμο: <http://gazebosim.org/>.
- [31] "Gazebo : Tutorials." *Gazebosim.org*. N. p., 2019. Web. 19 Aug. 2019.
- [32] "UberSim Simulator" [Online]. Διαθέσιμο: <http://www.cs.cmu.edu/~robosoccer/ubersim/>.
- [33] Browning, Brett, and Erick Tryzelaar. "Übersim: a multi-robot simulator for robot soccer." *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*. ACM, 2003.
- [34] "Actin Simulator" [Online]. Διαθέσιμο: <https://www.energid.com/>

- [35] Kumar, Kishy, and Parminder Singh Reel. "Analysis of contemporary robotics simulators." *2011 International Conference on Emerging Trends in Electrical and Computer Technology*. IEEE, 2011.
- [36] "X-Plane Simulator" [Online]. Διαθέσιμο: <https://www.x-plane.com/>
- [37] "X-Plane 11 Desktop Manual | X-Plane." *X-plane.com*. N. p., 2019. Web. <https://www.x-plane.com/manuals/desktop/>
- [38] Φράγκου, Σ. "Εκπαιδευτική ρομποτική: παιδαγωγικό πλαίσιο μεθοδολογία ανάπτυξης διαθεματικών συνθετικών εργασιών." *Γξεγνξηάδνπ, Ε. Γνπιή, & Γ* (2009).
- [39] "WRO Hellas Home - Εκπαιδευτική Ρομποτική Στην Ελλάδα." WRO HELLAS. N. p., 2019. Web. <https://wrohellas.gr/>.
- [40] Γ. Κορρέζ. "Εργαστήριο εκπαιδευτικής ρομποτικής με χρήση των Lego Mindstorms NXT", *5ο Πανελλήνιο Συνέδριο Καθηγητών Πληροφορικής, Ιωάννινα*, 1-3 Απριλίου 2011.
- [41] Βότσης, Κωνσταντίνος Χ. *Αξιοποίηση των ΤΠΕ στη διδασκαλία παιδιών που βρίσκονται στο φάσμα του αυτισμού*. MS thesis. 2018.
- [42] Roak Ely, Mekai Ely, Mark Cieslikowski. "Agrobots - Robots in Agriculture". 2007.
- [43] Gemtos, T., et al. "Precision farming experience in Europe and the Greek potential." *Greek conference on information and communication technology in agriculture*. 2002.

- [44] “SysGauge” [Online]. Διαθέσιμο: <https://www.sysgauge.com/index.html>