Acknowledgment

In the first place, we thank God for helping us to succeed, so we would like to thank all those who helped us to do this work.

Special thanks go to: Ms.Chourouk Guettas. For her support, her effectiveness and her advices in the preparation of this report.

These thanks are also addressed to our teachers who have contributed to our training, without forgetting the members of the jury, in recognition of our gratitude.

Dedication

We give this humble report to each of our mothers, fathers, sisters, brothers, and also to all who helped us (our teacher Ms.Chourouk Guettas and our frinds Hassouna Othmane, Abdelhay Affoun, Ayachi Youcef, Lachraf Raki).

And then to all the teachers who taught us and made efforts in our education.

And finally, all the thanks and appreciation to all our helpers to get here.

Abstract

Robots can automate a wide range of physical tasks from warehouse management to space exploration. Many of these tasks require robots to navigate autonomously. Such a robot should continuously choose and execute actions from a set of available actions until it reaches the destination. The robot's decision should avoid collisions and other damaging outcomes while it ensures favorable ones, such as reduced travel distance and time. This project reviews decision-making techniques for effective navigation. In particular, it reviews different methods to acquire information about the environment and how this information can be used for effective decision making and motion planning. The goal is to make the robot avoid the difficult obstacles within a complex environment and move according to sensors. We used NetLogo to simulate the robot and test the proposed approach.

KeyWord:

Robotic, Robot, Artificial Intelligence,Obstacles, Complex Environment, Motion Planning, Exploration ,Navigation, NetLogo, Simulation.

Résumé

Les robots peuvent automatiser un large éventail de tâches physiques, allant de la gestion de l'entrepôt à l'exploration spatiale. Beaucoup de ces tâches nécessitent que les robots naviguent de façon autonome. Un tel robot doit toujours choisir et exécuter des actions à partir d'un ensemble d'actions disponibles jusqu'à ce qu'il atteigne la destination. La décision du robot devrait éviter les collisions et autres résultats néfastes, tout en assurant des conditions favorables, telles que la distance et le temps de déplacement réduits. Ce projet examine les techniques de prise de décision pour une navigation efficace. En particulier, il examine différentes méthodes pour acquérir des informations sur l'environnement et la manière dont ces informations peuvent être utilisées pour une prise de décision efficace et une planification de mouvement. L'objectif est de faire en sorte que le robot évite les obstacles difficiles dans un environnement complexe et se déplace selon les capteurs. Nous avons utilisé NetLogo pour simuler le robot et tester l'approche proposée.

Mots-clés:

Robotique, Robot, Intelligence Artificielle, Obstacles, Environnement complexe, Motion Planning, Exploration, Navigation, NetLogo, Simulation.

Contents

List of Figures

List of Tables

General Introduction

General Introduction

Today, robots are a good substitute for the human in several tasks; In heavy and cruel tasks and also in prudent tasks. Robots have been found to be advantageous over humans in many activities (inspection, wars, industrial processes, etc...). At the beginning of the robotic era, robots were small and capable of performing simple axioms or gestures. Then they were asked to adapt to their environment and to follow rigorous instructions in order to carry out more complicated tasks, which led to a rich field of research which aims to produce robust control laws and robots with Adaptive behaviors.

The word robot was first introduced in Czech playwright Karl Kapek in 1920. The title of the play at the time was a global iconic people. In the Czech language it means "hard work," although Karl was the first to use this word, but not his inventor, but Joseph's brother, who, as an assistant to his brother, derived from the Czech word "Robota" which meant forced labor or forced labor. From this date began to spread the word in the books and science fiction films first gave a scientific idea of these robots who will seize the world. And gave a great horizon and great promises to the Marvelous Man who will intervene in many things in different and complex places and environments.

There are many very complex and unorganized environments, while being mostly designed for humans. Doors, stairs, or wall can be easily overcome by humans, but they pose serious obstacles to the robot. Also, knowing that there are some places that people cannot reach, some obstacles are unavoidable, and that they are at risk, and here comes the role of the robot to make it easy and accessible to humans. From here we raise the problem of how the components of the robot work, especially sensors that help him to avoid obstacles in difficult environments.

The objective of this topic is to create a robot containing sensor that enables it to avoid obstacles. In short the robot will use these sensors for moving in the complicated environment and evade these obstacles until it gets to the goal.

Our report is subdivided into three main chapters:

- **In the first chapter:** we will try to give an overview of robotics and robots.
- **The second chapter:** conception of our work and how the robot avoiding the obstacles using sensors.
- **The third chapter:** the implementation of our system and the results obtained.

Finally, we conclude with a general conclusion.

Chapter 01:

Robotics & Robots

Chapter 1

Robotics & Robots

1.1 Introduction

A robot is a machine capable of doing pre-programmed work, either by direct instructions and control of humans or by software programs. The work of the robot and its performance is often arduous, dangerous or accurate, such as searching for mines and disposing of radioactive waste, or precise and arduous industrial work. In this chapter, we will describe this technology in general, and some points related to it.

1.2 Artificial Intelligence

Figure 1.1: *Artificial Intelligence.*

1.2.1 Definition

Artificial intelligence is the branch of computer science that develops machines and software with human-like intelligence. It is the intelligence exhibited by software or machines. The field was founded on the claim that a central property of humans is intelligence, and that it can be sufficiently well described to the extent that a machine can simulate it.
The central goals of artificial intelligence research include knowledge, reasoning, learning, planning, perception, the ability to manipulate and move objects and natural language processing. [1]

1.2.2 Why AI?

"AI can have two purposes. One is to use the power of computers to augment human thinking, just as we use motors to augment human or horse power. Robotics and expert systems are major branches of that. The other is to use a computer's artificial intelligence to understand how humans think in a humanoid way. If you test your programs not merely by what they can accomplish, but how they accomplish it, then you're really doing cognitive science; you're using AI to understand the human mind." [2]

1.3 Robotics and Robots

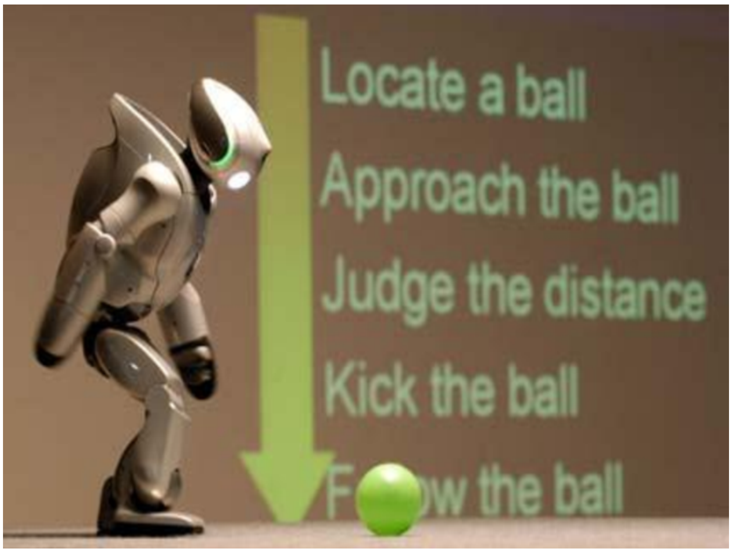


Figure 1.2: *Sony Corporation's biped entertainment robot prepares to kick a ball during an interactive program with school children in Delhi, 2004.*

1.3.1 Robotics

It is a branch of technology that deals with designing, construction, operation, and application of robots. It also deals with the computer systems for their sensory, control, information processing and feedback. These technologies deal with automated machines that can replace humans in manufacturing processes or dangerous environments.
These robots resemble humans in behavior, appearance, and/or cognition. Robotics requires a working knowledge of mechanics, electronics, and software. [3]

1.3.2 Robots

A robot is an electro-mechanical machine that can do the work of a person and that works automatically or is controlled by a computer is defined as a Robot. It is a device that can perform automatically or through some controlling devices. A robot is defined as "a machine capable of carrying out a complex series of actions automatically, especially one programmable by a computer.” (Oxford Dictionary, 2013)
Robots are of a wide range. The common feature of robots is their capability to move. They perform physical tasks. Robots have many different forms. They range from industrial robots, whose appearance is dictated by the function they are to perform. Or they can be humanoid robots, which mimic the human movement and our form.
Robots can be grouped generally as:

1. Manipulator robots (for e.g. industrial robots).
2. Mobile robots (for e.g. autonomous vehicles).
3. Self-reconfigurable robots, the robots that can conform themselves to the task at hand.

Robots may act according to their own decision making ability, provided by artificial intelligence or may be controlled directly by a human, such as remotely-controlled bomb disposal robots and robotic arms; or. However, the majority of robots fall in between these extremes, being controlled by pre-programmed computers. [4]

1.3.3 Robot Working

Human beings on a basic level are made of five major components:

- A muscle system that can move the body structure.

- A body structure itself.
- A power source that can activate the muscles and sensors.
- A sensory system which can receive information about the body and the surrounding environment.
- A brain system which can process sensory information and tell the muscles what to do.

Robots are made up of the same components as above. A typical autonomous robot has a sensor system, a movable physical structure, a power supply and a computer brain that controls all of these elements. Basically, robots are man-made versions of the animal life. They are machines that can replicate human and animal behavior. [5]

1.3.4 Robot Learning

Robot learning is an intersecting research field between robotics and machine learning. It studies techniques that allow robots to acquire skills and adapt to its environment by learning various algorithms. Learning can take place either by self-exploration or through guidance (from a human teacher), like in robot learning that learns by imitation. [6]

1.3.5 The Actuator

All robots have a movable body (almost all). Some have motorized wheels only, while others may have a dozen of movable parts (that are typically made of plastic or metal). Like bones in a human body, the individual segments are connected together with the help of joints.
Robots use actuators to spin wheels and jointed pivot. Some robots use solenoids and electric motors as actuators; others some use a pneumatic system (a system driven by compressed gases); yet others use a hydraulic system. A robot may even use all of these actuator types together.
Robots need a power source to be able to drive the actuators. Most robots have a battery or they plug into an electricity source. Pneumatic robots need air compressors or compressed air tanks and hydraulic robots need a pump that pressurizes the hydraulic fluid. The actuators are wired to an electrical circuit. The circuit powers these electrical motors and solenoids directly. It also activates the hydraulic system by manipulating electrical valves. The valves determine the pressurized fluid's path through the machine. [7]

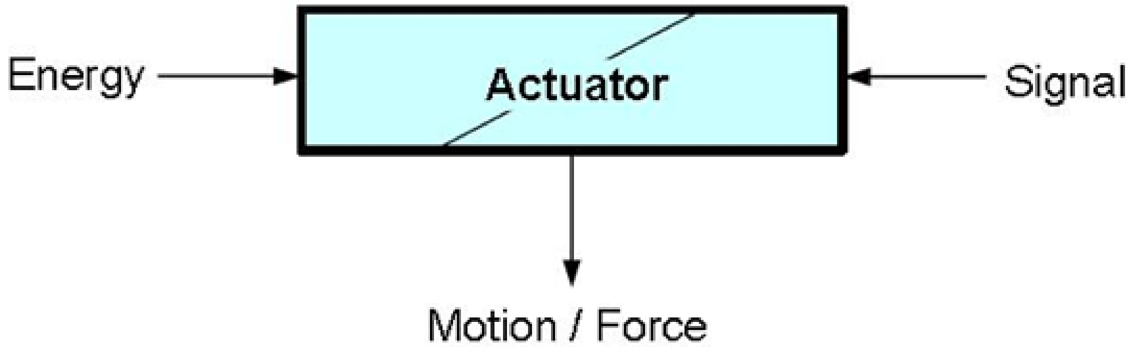


Figure 1.3: *Actuator Diagram.*

1.4 Obstacle-Avoiding Robot

1.4.1 Introduction:

An obstacle avoiding robot works on the principle that on having obstacle in its path, it stops there before colliding with the obstacle and makes backward motion for some time and then takes right or left turn and follows its path.
Uses/applications: Obstacle avoider is a key feature in any kind of robot, household appliance, pc peripheral, or vehicle that needs to move around autonomously, including: remote web cameras, surveillance robots, personal robot assistants, vehicle security and advanced personal warning systems, etc...

1.4.2 Advantages & Disadvantages of Obstacle Avoidance Robot

Advantage

- When ever the robot detects any obstacle, it automatically deflects its.
- position to the left or to the right and follows the path without human guidance.
- Easy microcontroller programming.
- It is a low cost circuit.

Disadvantages

- Is a waste of time project.

- It is not recommended to keep the range too long because it will make the robot move forward and backward as it senses any obstacle, even away from it.
- It is not in human control. [8]

1.5 The Robots Applications

1.5.1 Introduction

Currently, robots perform a number of different jobs in numerous fields and the amount of tasks delegated to robots is rising progressively. The best way to split robots into types is a partition by their application

1.5.2 Applications

- **Industrial Robots:**

  These robots play a role in the manufacturing industry. These arsenals are usually created specifically for applications such as material handling, painting, welding and others. If we evaluated only by application then this type of robot could also consist of some auto-directed robots and other robots.

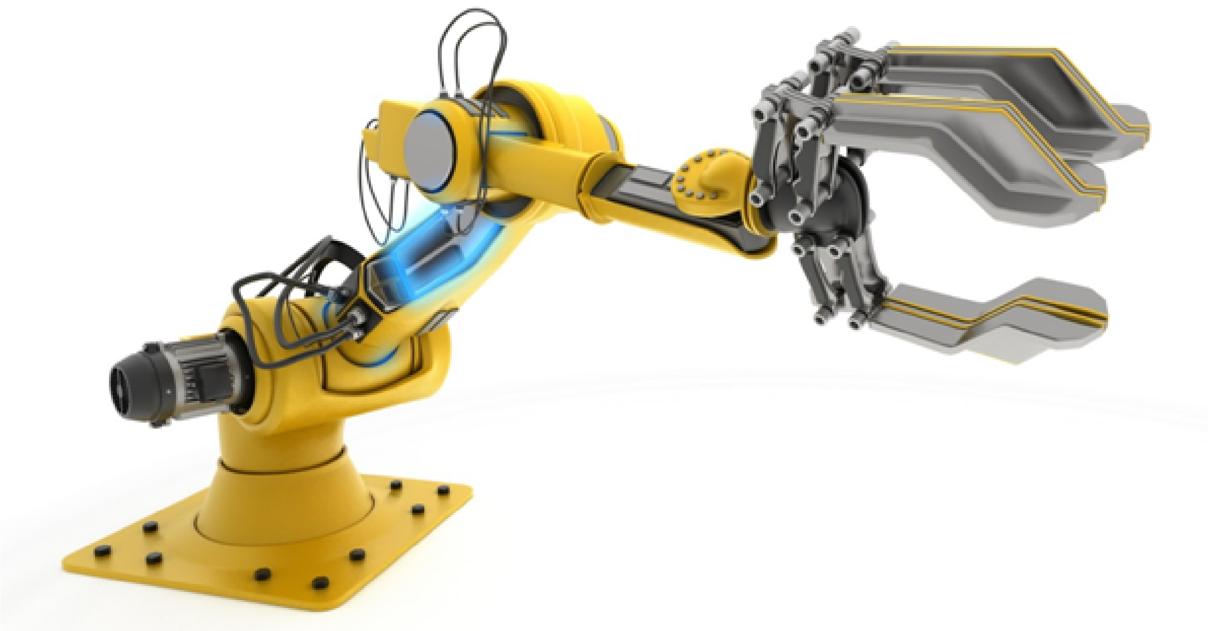


Figure 1.4: *Industrial Robots.*

- **Domestic or Household Robots:**

  Robots that are used at home. This type of robot is made up of many different gears for example - robotic pool cleaners, robotic robbers, robotic vacuum cleaners, sewage robots cleaners and other robots that can perform various household tasks. Also, a number of robots can be considered for scrutiny and remote presence of local robots if played in this type of environment.

Figure 1.5: *Domestic or Household Robots.*

- **Medical Robots:**

  Robots used in medicine and medical institutes. First of all robots are surgical treatment. Also, a number of robotic-oriented cars may have lift proponents.

Figure 1.6: *Medical Robots.*

- **Service Robots:**

  Robots that cannot be classified into other types by practice. These could be various data collection robots, robots ready to expose technologies, robots used for research, etc...

Figure 1.7: *Service Robots.*

- **Military Robots:**

  Robots have been brought into play in the armed and military forces. This type of robot consists of robots of destruction of the bomb, various robots of expedition, drones of exploration. Often, robots initially produced for the purposes of the armed forces and armed forces can be used in law enforcement, exploration and rescue and other associated fields.

Figure 1.8: *Military Robots.*

- **Entertainment Robots:**

  These types of robots are used for entertainment. This is an extremely broad category. It starts with robot models such as robosapien or photo frames running and ends with real heavyweights like arms of articulated robots used as motion simulators.

Figure 1.9: *Entertainment Robots.*

- **Space Robots:**

  We would like to distinguish robots used in space as a separate type. This type of robot would consist of the robots used on the Canadarm that were brought into play in space shuttles, the International Space Station, as well as Mars explorers and other robots used in space exploration and other activities .

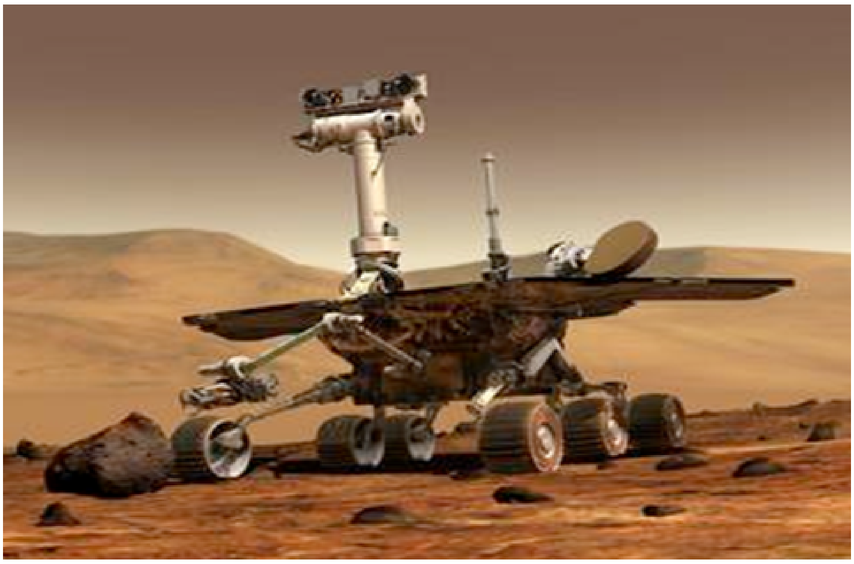


Figure 1.10: *Space Robots.*

- **Hobby and Competition Robots:**

  The robots created by the students. Sumo-bots, Line seguiteurs, robots prepared only for learning, fun and robots prepared for competitions.[9]

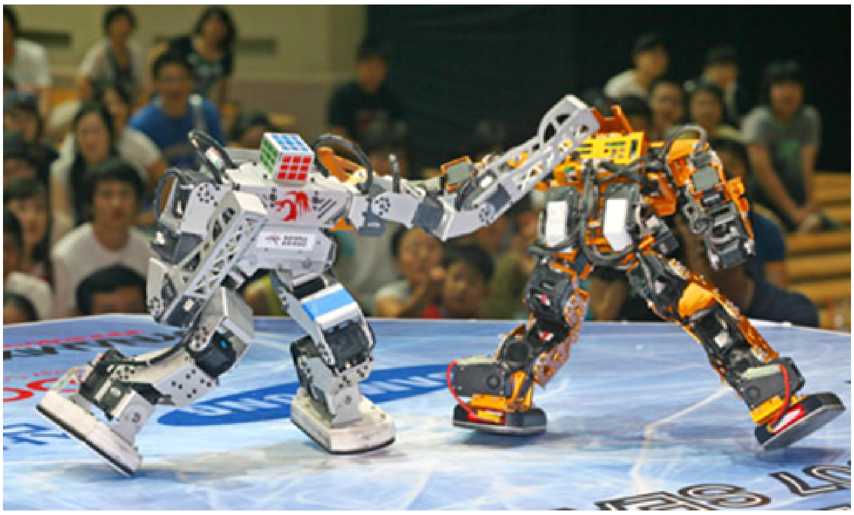


Figure 1.11: *Hobby and Competition Robots.*

1.6 Conclusion

In this chapter we have presented an overview of robotics and robot, we have defined the artificial intelligence and obstacle avoidance robot with its advantages and disadvantage also some applications robots.

In the next chapter, we will be presenting our conception of the robot and how the robot avoiding the obstacles using sensors.

Chapter 02 :

Conception

Chapter 2

Conception

2.1 Introduction

The robots have a number of sensors and their number varies from one robot to another. In this work we will describe a robot consisting of five sensors .The robot moves according to sensor devices for discovering the complex environment and avoiding the obstacles in it.

In this chapter we will give a summary on multi-agent systems, and then we expose our work by presenting a description of our work and general architecture of our system.

2.2 Multi-Agent System (MAS)

A multi-agent system (MAS) or système multi-agents (SMA) in French , is a distributed system composed of a set of autonomous agents that interact in a common environment and work together in complex modes of cooperation, competition or coexistence to obtain A global object. There are several propositions for the elements that make up the multi agent system we take a decomposition as follows:
The decomposition of an SMA in four dimensions which correspond to the four volleys A, E, I and O:

- Agent: Definition of modeling or architectures of system components.
- Environment: Environment, in which agents are immersed, composed of objects that are perceived and manipulated by agents, and which obey physical laws.
- Interactions: sets of infrastructures, languages and protocols of interactions between agents.

- Organization: structure of agents into hierarchical groups, relationships. [10]

2.2.1 Cognitive Agent

A cognitive agent is capable of performing more complex operations on its own. A multi-agent cognitive system includes a small number of agents. Each agent has a reasoning capacity based on knowledge, an ability to handle various information related to the field of application and information related to the management of interactions with other agents and the environment.

2.2.2 Reactive Agent

A reactive agent tends to simplicity and is directly related to its perceptions and responds to changes in its environment. But it is also capable of performing complex and coordinated group actions. The sum of the members is capable of evolved actions but each individual taken separately has a weak representation of the environment and has no global goals. [11]

2.3 Description of our Work

Our work objective is developing a control center unit, which allows the robot to move towards his target .The robot moves according to sensor devices for discovering the complex environment and avoiding the obstacles in it.
The robot has five sensors each one is oriented toward a specific angle, the main goal is to create a set of rules that will takes the right decision with each step.

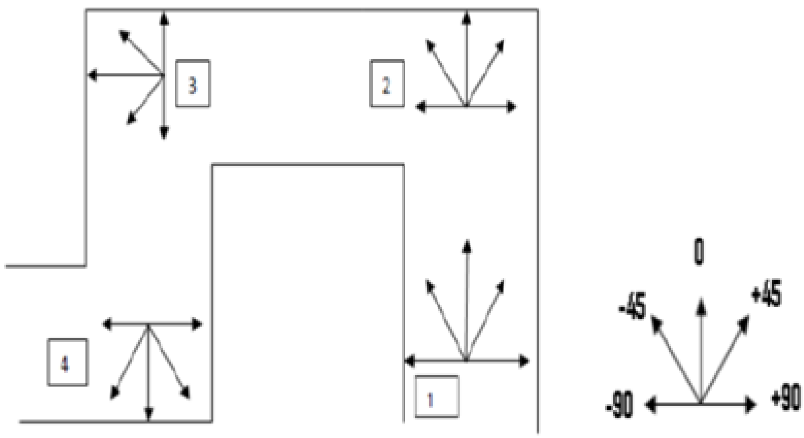


Figure 2.1: *Description of our Work.*

2.4 General Architecture of our System

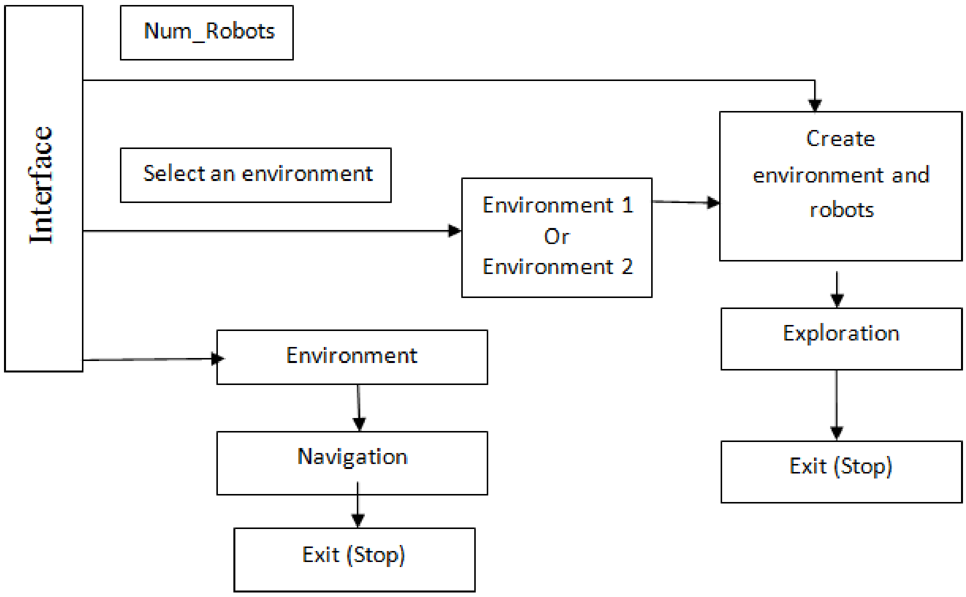


Figure 2.2: *General Architecture of our System.*

2.4.1 Environment

The environment is the space that must be explored by the robot to achieve a goal that is in this environment.

The environment may be uncertain or even unknown to the modular robot, it may contain several types of obstacles that prevent the robot from moving towards its goal.

The obstacle is modeled by an obstacle object, it is represented by a square (a square having the size of a square). The obstacle may consist of several neighboring obstacle objects which may be arranged in different forms. The obstacle handicaps the movement of the robot, for this the robot must reconfigure to overcome this obstacle.

The objective of the obstacles is to create a difficult environment in our simulation.

In our simulation have 2 environments: Difficult Environment and Maze.

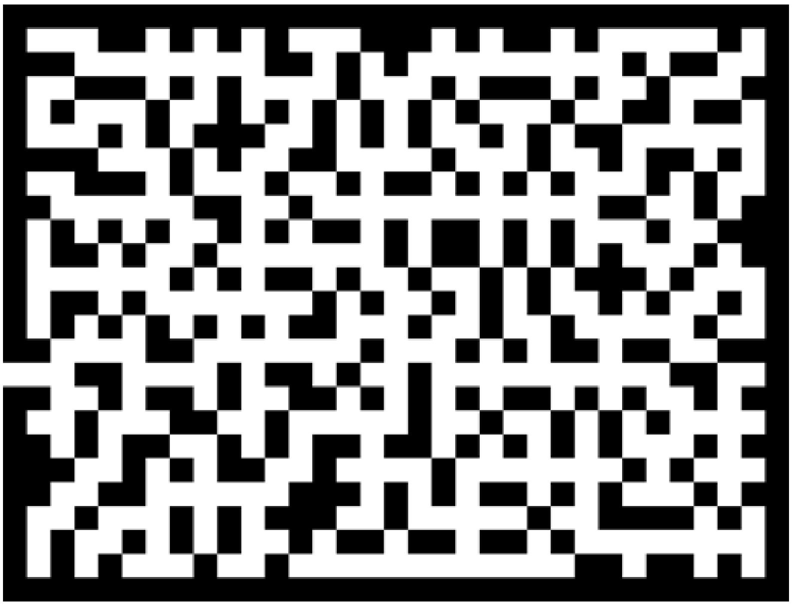


Figure 2.3: *Difficult Environment.*

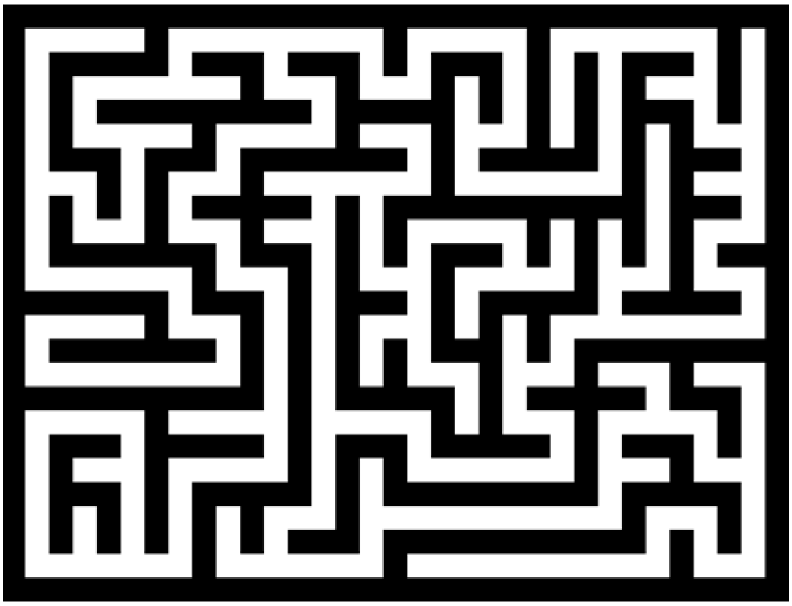


Figure 2.4: *The Maze.*

2.4.2 Motion Planning

The robot's movement depends on the information's collected by the sensors, that practically survey the existence of obstacles and alert the robot to avoid it.

The robot avoids the obstacles until he arrives to the stop point.

2.4.3 Description of the Algorithm

Our algorithm depends on sensor devices installed on the robot, which they are distributed & fixed in the most important angles as the picture below shows (Figure 2.5).

Figure 2.5: *Most Important Angles.*

- Like you've remarked, devices are away from each other with 45 degrees angle each, that allows the robot to recognize the obstacles in front and sides.
- every sensor acts with such a simple theory which is coming with 1 output if there is an obstacle in the concerned angle range and 0 if not, then the algorithm comes up with the suitable way for our robot starting from a table of probabilities filled with outputs of each angle.
- The five inputs are represented by variables X1, X2, X3, X4, X5 corresponding to angles -90,-45, 0, +45, +90, respectively .

State	X1	X2	X3	X4	X5	Output
(01)	1	0	0	0	0	Straight
(02)	1	1	0	0	0	Straight
(03)	1	1	1	0	0	Right
(04)	1	1	1	1	0	Right
(05)	1	1	1	1	1	Go Back
(06)	0	1	0	0	0	Straight
(07)	0	1	1	0	0	Right
(08)	0	1	1	1	0	Right
(09)	0	1	1	1	1	Left
(10)	1	1	1	1	1	Go back
(11)	0	0	1	0	0	Right
(12)	1	0	1	0	0	Right
(13)	1	1	1	0	0	Right
(14)	1	1	1	1	0	Right
(15)	1	1	1	1	1	Go back
(16)	0	0	0	1	0	Straight
(17)	1	0	0	1	0	Straight
(18)	1	1	0	1	0	Straight
(19)	1	1	1	1	0	Right
(20)	1	1	1	1	1	Go back
(21)	0	0	0	0	1	Straight
(22)	1	0	0	0	1	Straight
(23)	1	1	0	0	1	Straight
(24)	1	1	1	0	1	Go back
(25)	1	1	1	1	1	Go back

Table 2.1: Possible Situations

2.5 Exploration

In exploration, the robot moves according to the previous cases (states) where each step implements the algorithm and thus moves in this environment or maze to search or reach points randomly distributed.

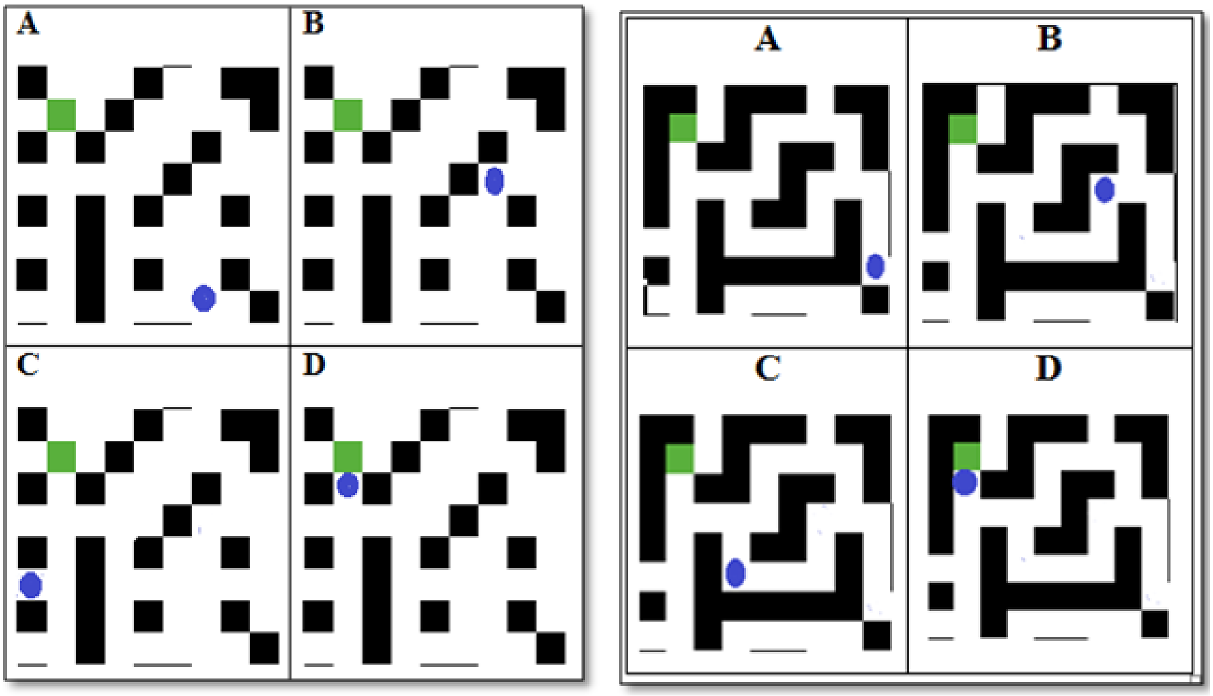


Figure 2.6: *Example of exploration in environment and maze.*

2.6 Navigation

In navigation, the robot also move according to the previous cases (states), but it but starts from the starting point to the other point and looks for the nearest path to that point using gps by applying an algorithm and then goes directly to it.

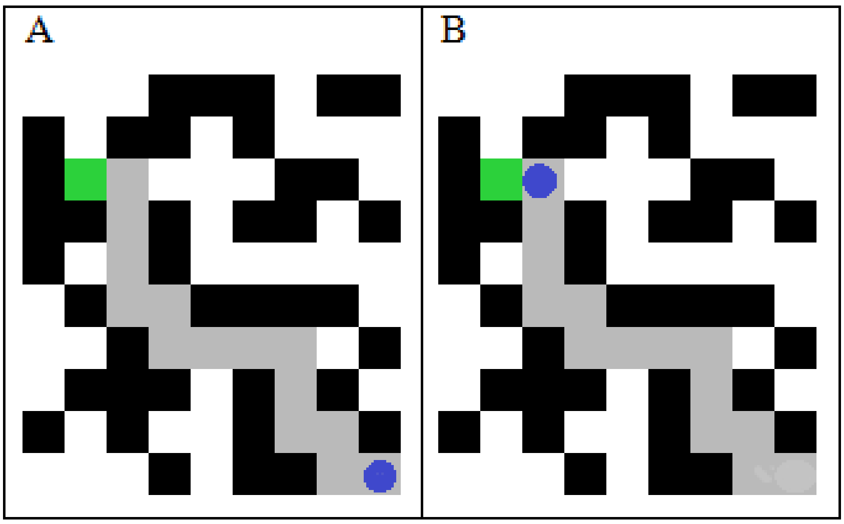


Figure 2.7: *Example of Navigation.*

Explain its Own Algorithm:

- At first we give the robot the location of the exit point,
- Using GPS, it identifies all possible movements of the robot when it is moving (The robot is constant in its location),
- The odds are arranged by the number of moves (steps), from one step and then two steps etc ..., example: the probability of finding 4 paths it could move2 steps in them.
- Until the robot moves through the step to the exit point,
- Then the determining of the the probabilities stops,
- Specifies the path that gets it to the point,
- The specified route is therefore the closest route to the exit point,
- Then it goes directly to the point.

2.7 Conclusion

We have devoted this chapter to presenting our work, first describing our project, defining its tasks, then designing our system and modeling the robot.

In the next chapter we will present the implementation of the System and the results obtained.

Chapter 03 :

Implementation & Results

Chapter 3

Implementation & Results

3.1 Introduction

After looking at the design of our system in the last chapter in this chapter for the first time we offer the implementation of the robot movement we have chosen, and then simulate the behavior of the robot during its navigation, the robot can face obstacles and which it must avoid to continue its movement.
We will describe the different data structures and algorithms used. But before you submit the results, take a look at our system and deal with them.

3.2 Programming Language Used

We realized our project under the NetLogo platform (5.3.1). In this section, we will give a general view on Netlogo and our reasons behind this choice.

3.2.1 Presentation of NetLogo

- NetLogo is a programmable modeling environment for simulating natural and social phenomena. It was authored by Uri Wilensky in 1999 and has been in continuous development ever since at the Center for Connected Learning and Computer-Based Modeling.

- NetLogo is a programming environment designed to model decentralized systems. It is well suited to the modeling of complex systems composed of hundreds, thousands of agent acting in parallel, it is entirely written in java, with integration of some OpenGL libraries. [12]

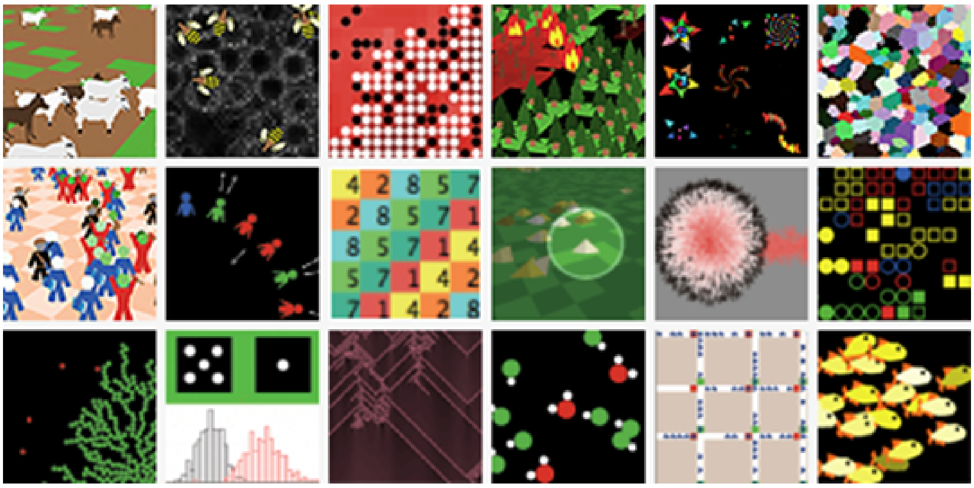


Figure 3.1: *Screenshots NetLogo.*

3.2.2 "Agent" Concepts

- The NetLogo world is made up of "Agent", who can follow instructions.
- The activities of the different "Agent" execute simultaneously.
- There are 3 types of Agent:
  1. Turtles: agents moving around the world.
  2. Patches: an environment in which the turtles can locate and move.
  3. The observer: looks from outside the world of turtles and patches.

3.2.3 Motivation of Choice of NetLogo

We chose this platform for the services it offers us:

- It allows us to design an interface for the user in a special and interactive way.
- A specific language for multi-agent simulation.
- A language adapted to model complex systems.
- A language modeling of decentralized systems.
- A modeling and programmable environment to simulate social and normal phenomena.
- NetLogo 3D allows you to create a 3D world.

- Netlogo is particularly suited for modeling complex systems that develop over time.
- Completely programmable.
- Simple language structure.

3.3 Data Structures Used

The first step in the realization of our system is the choice of the different data structures necessary for the implementation of the different objects.

3.3.1 Local Variables

- Turtle Structure
  - Size: integer // is the size.
  - Xcor: integer // indicates the initial position in the X axis.
  - Ycor: integer // indicates the initial position in the Y-axis.
  - Zcor: integer // indicates the initial position in the Z axis (in 3D).
- The Environment (Patches), Structure:
  - Size: integer // represent the size of the patch.
  - Pcolor: integer // represent the color of the patch.
  - Pxcor: integer // represent the position of the patch in the X axis.
  - Pycor: integer // represent the position of the patch in the Y axis.
  - Pzcor: integer // represent the position of the patch in the Z axis (in 3D).
- Structure Obstacle
  - Size: integer // represent the size of the patch.
  - Pcolor: integer // represent the color of the patch.
  - Pxcor: integer // represent the position of the patch in the X axis.
  - Pycor: integer // represent the position of the patch in the Y axis.
  - Pzcor: integer // represent the position of the patch in the Z axis (in 3D).

3.4 Implementation of the Software

3.4.1 Software Procedures

- Procedure (envi1): This procedure allows us to create environment.
- The initialization procedure (setup): This procedure allows us to set up robots with their number setting using the slider.
- Procedure (E-Move): This procedure allows the robot to move in the environment.
- Procedure (envi2): This procedure allows us to create a maze.
- Procedure (E-Move1): This procedure allows the robot to move in the maze.
- Procedure (draw-obstacle): This procedure allows us to draw obstacle.
- Procedure (erase-obstacle): This procedure allows us to erase obstacle.
- Procedure (envi3): This procedure allows us to create environment.
- Procedure (envi4): This procedure allows us to create a simple environment.
- Procedure (find-an-optimal-path): This procedure allows the robot to find an optimal path.
- Procedure (N-Move): This procedure allows the robot to move in the environment.
- Procedure (test-capt): This procedure allows us to test the state of capture.

3.5 Presentation of the Software

The handling of our software is carried out using a clearer interface and easier to process. This interface is composed of two parts:

1. ***The first part contains (Figure 3.2):***

   Information that helps the user to start the software.

   The buttons that allow user interaction with the software, these buttons are:

   (envi1, setup, E-Move), (envi2, setup, E-Move1),

   (envi3, envi4, find-an-optimal-path, N-Move)

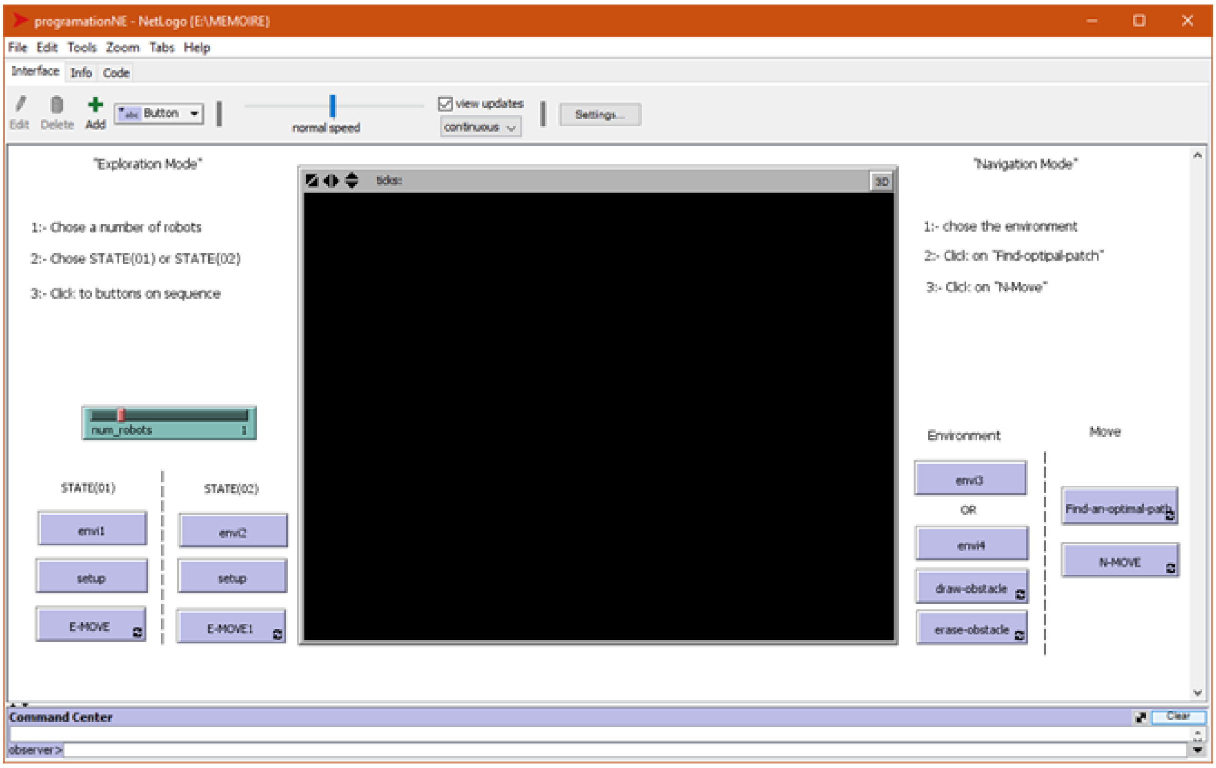


Figure 3.2: *Software Interface.*

2. ***The second part contains:*** a simulation window which shows the environment.

   We used three types of environments to see the behavior of the robot in the following cases:

   - **Case 1: (Environment1). (Figure 3.3):**we have in this environment a robot and obstacles and some point in order robot to find it.

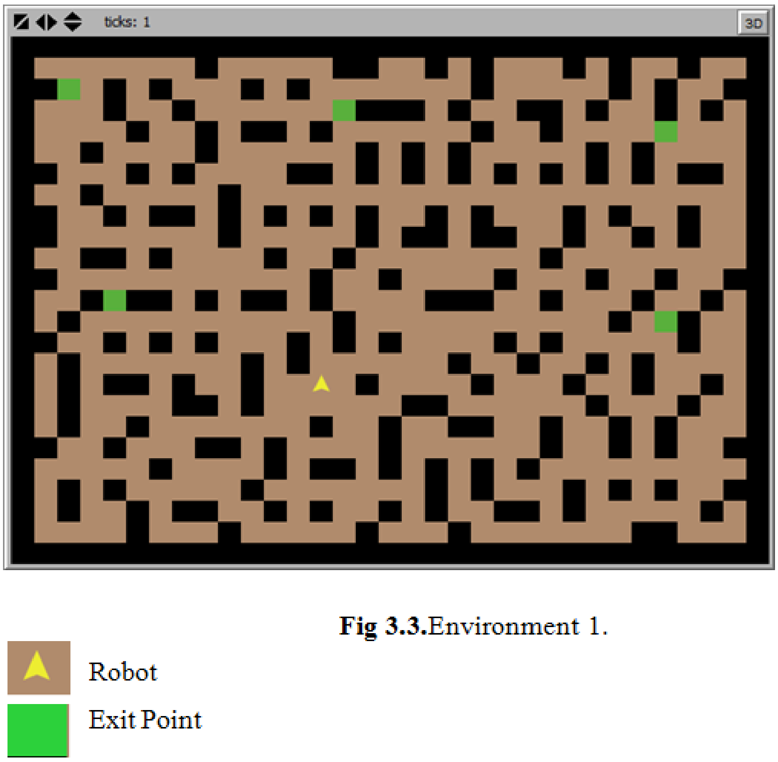


Figure 3.3: *Environment 1.*

- **Case 2: (Environment2 "Maze"). (Figure 3.4):**we have a maze and robot and obstacles and some point in order robot to find it.

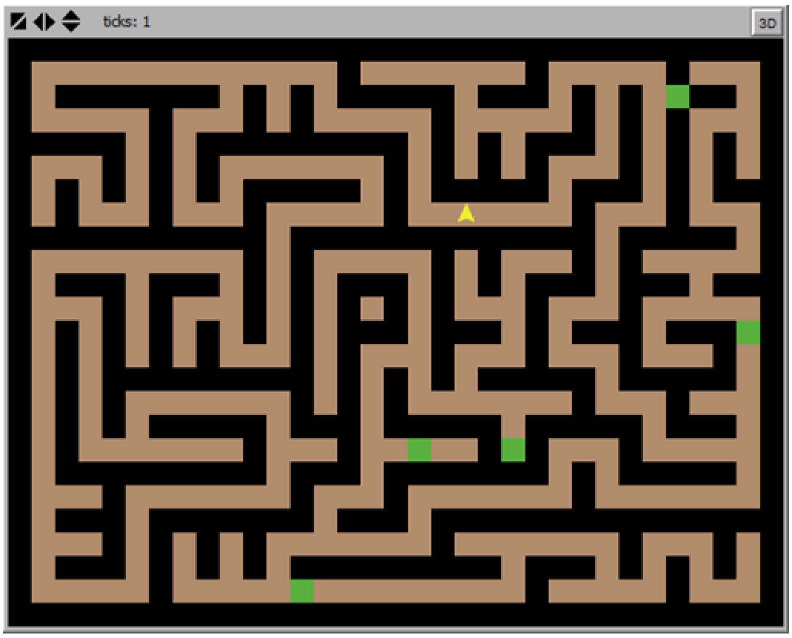


Figure 3.4: *Environment 2(Maze).*

- **Case 3: (Environment3). (Figure 3.5):**we have an environment containing obstacles and starting point of the robot (red point) and a point that the robot is trying to find it(green point).

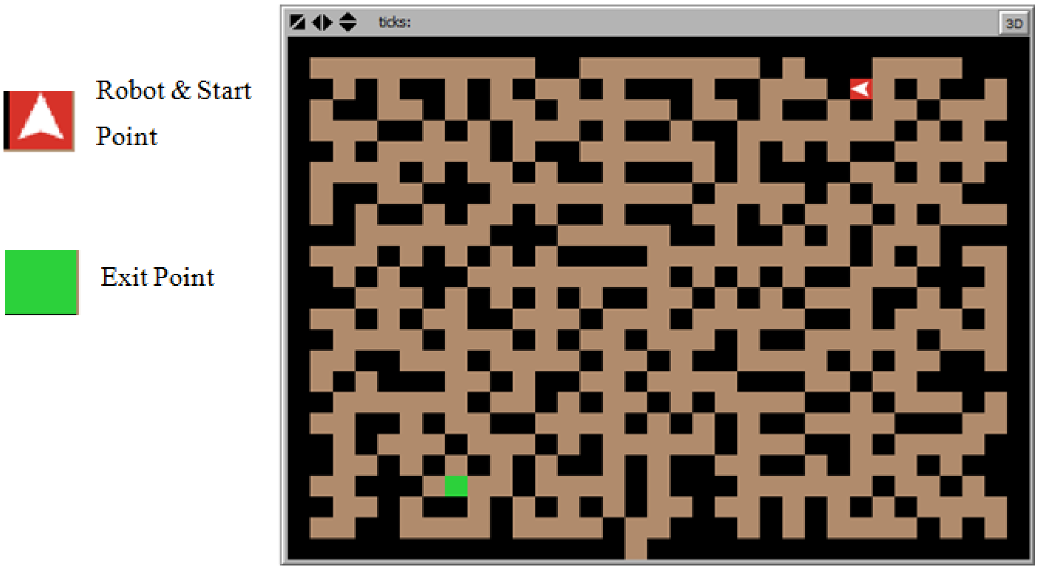


Figure 3.5: *Environment 3.*

- **Case 4: (Environment4). (Figure 3.6):**This case allows the user to draw the environment:

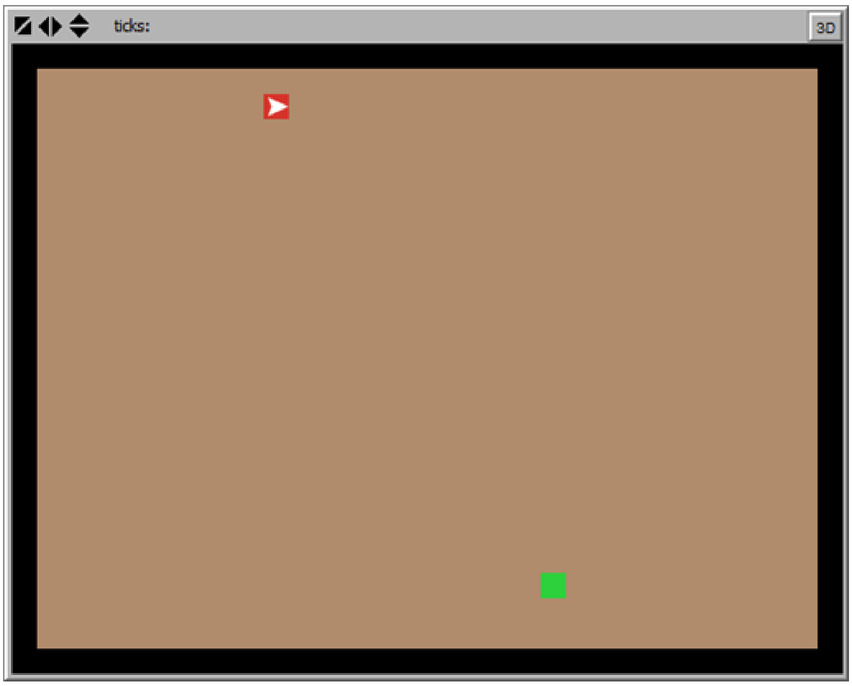


Figure 3.6: *Environment 4.*

3.6 Using The Software

- The button (envi1): allows us to create environment.
- The button (setup): allows us to set up robots with their number setting using the slider.
- The slider (num-robots):for Determine the number of robots
- The button (E-Move): allows the robot to move in the environment.
- The button (envi2): allows us to create a maze.
- The button (E-Move1): allows the robot to move in the maze.
- The button (draw-obstacle): allows us to draw obstacle.
- The button (erase-obstacle): allows us to erase obstacle.
- The button (envi3): allows us to create environment.
- The button (envi4): allows us to create environment.
- The button (find-an-optimal-path): allows the robot to find an optimal path.
- The button (N-Move): allows the robot to move in the environment.

And we have added simulations in 3D for complex environment in exploration mode.

We have: The buttons that allow user interaction with the software, these buttons are: (clear-all, genere, setup, Move) *Using the software in 3D:*

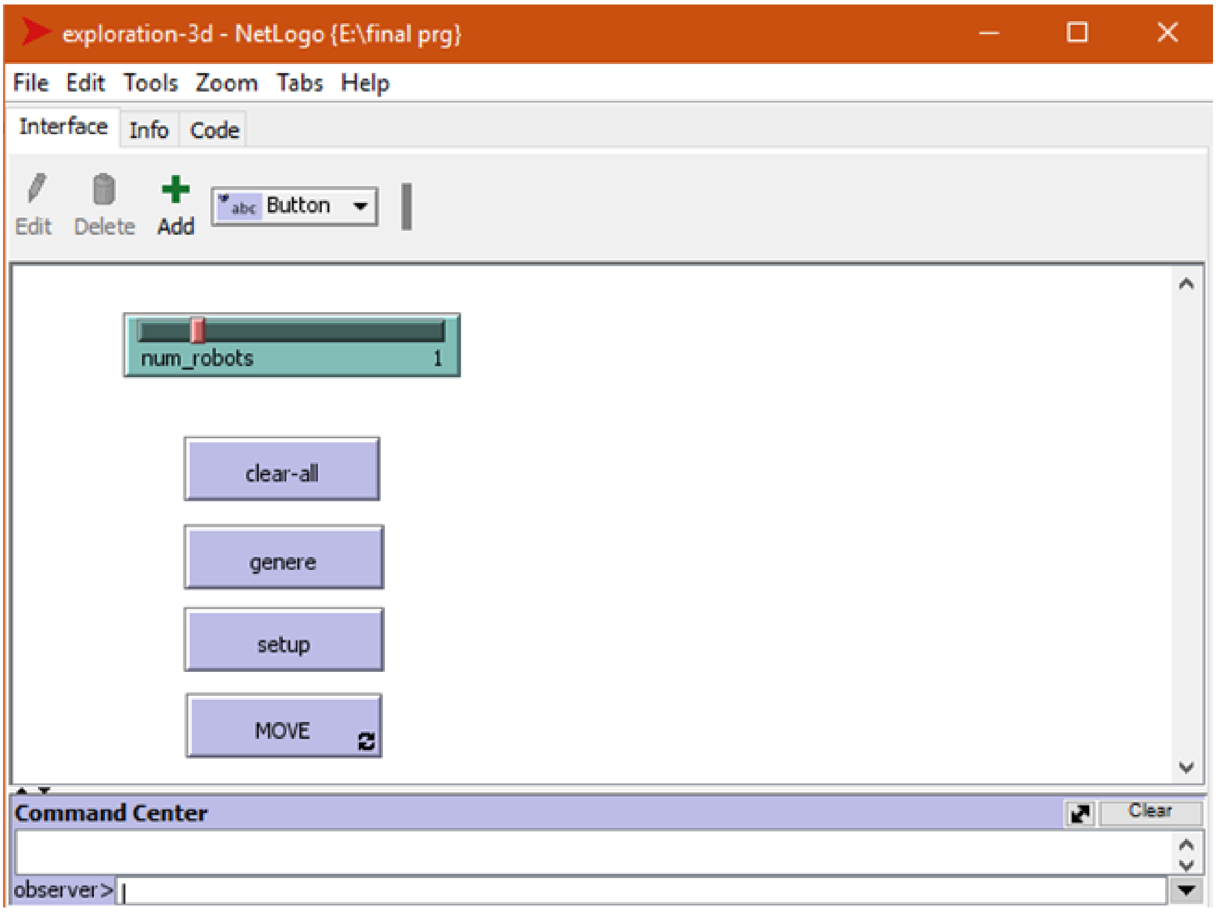


Figure 3.7: *3D Software Interface.*

- The slider (num-robots):for Determine the number of robots.
- The button (clear-all): allows us to clear the environment.
- The button (genere): allows us to create environment.
- The button (setup): allows us to set up robots with their number setting using the slider.
- The button (Move): allows the robot to move in the environment.

And simulation window which shows the environment in 3D:

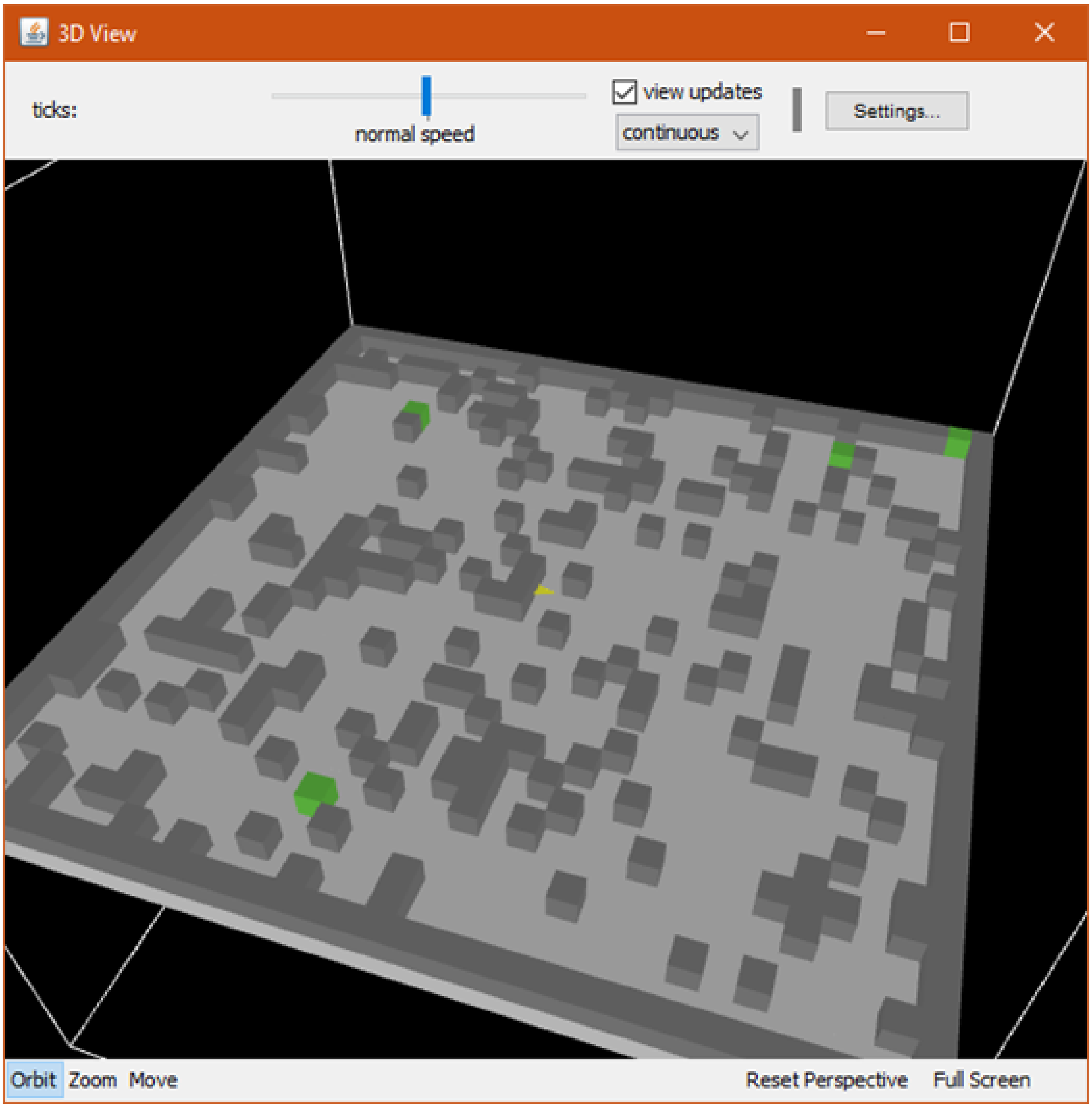


Figure 3.8: *3D Environment.*

3.7 Results Obtained

Our simulation consists of several types of environments and the following results show the ability of the robot to explore their environment and avoid obstacles.

Through experience we find the following results:

3.7.1 Exploration in Complex Environment

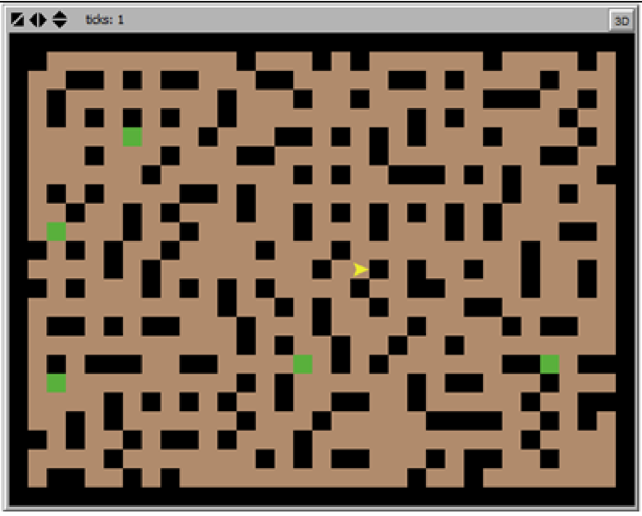
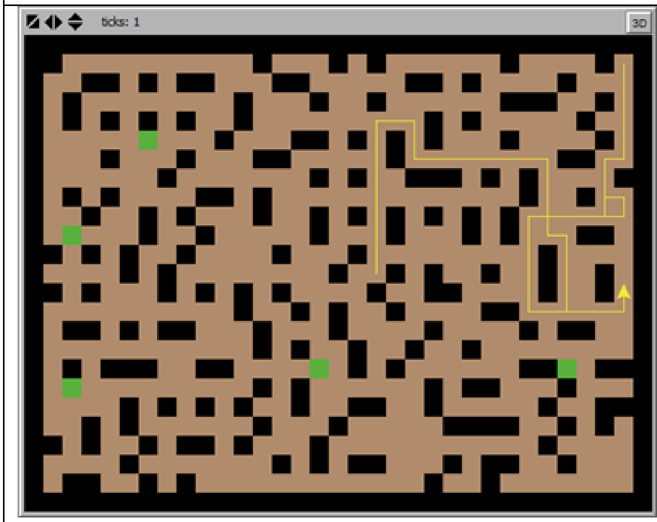
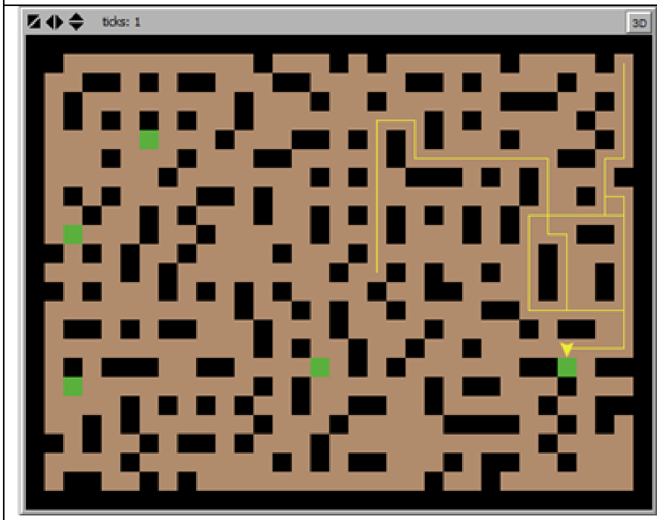
	☒ Initial Outlook
	☒ Move and explore the environment while avoiding obstacles in a random way.
	☒ Find the point in a random way.

Figure 3.9: *Environment 1.*

3.7.2 Exploration in a Maze

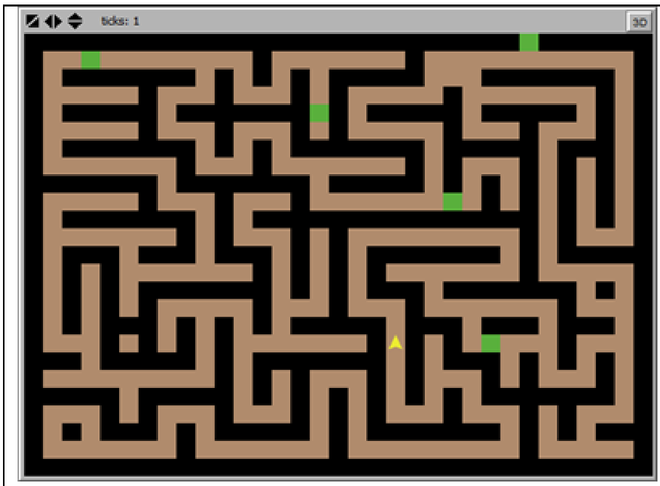
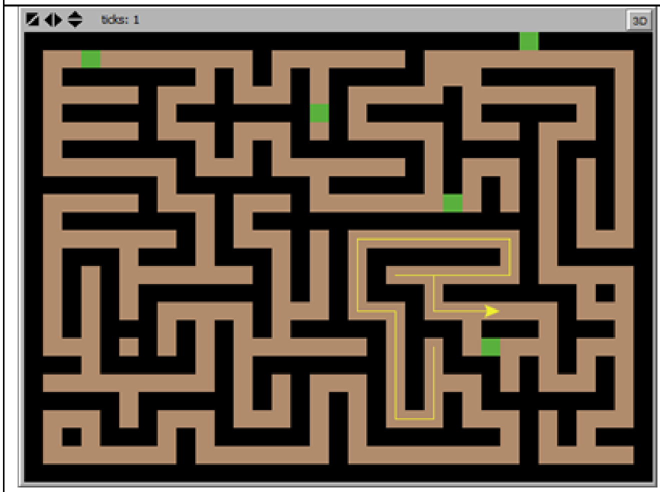
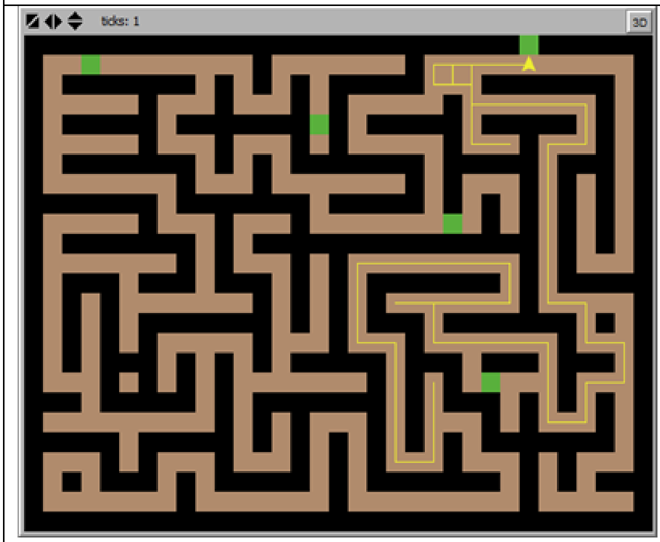
	☒ Initial Outlook
	☒ Move and explore the maze while avoiding obstacles in a random way.
	☒ Find the point in a random way.

Figure 3.10: *Environment 2.*

3.7.3 Navigation in a Complex Environment

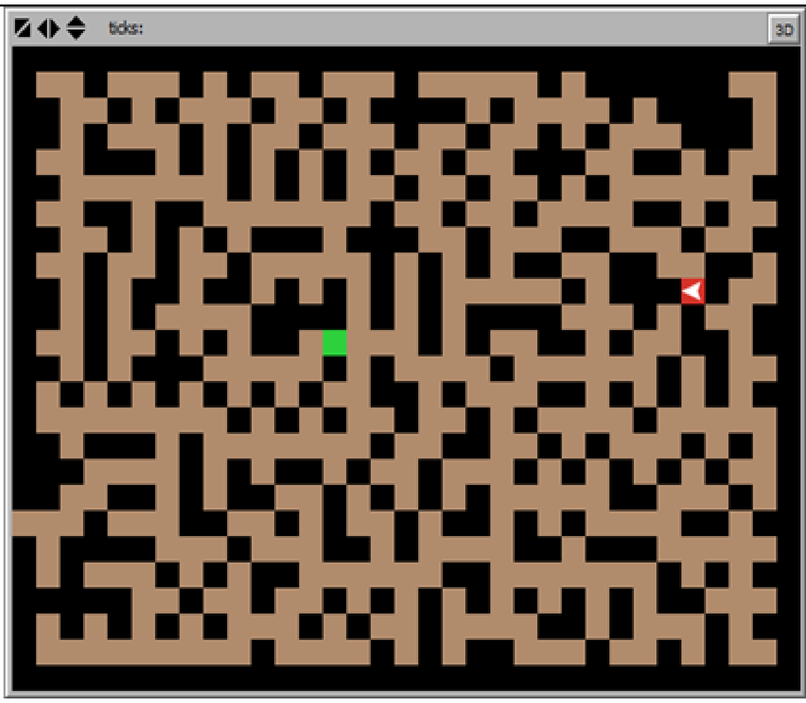
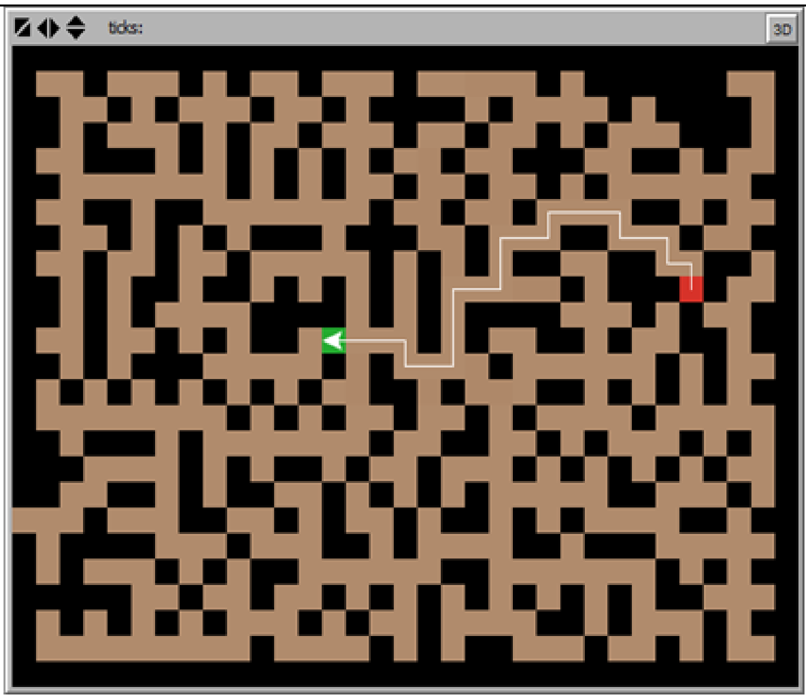
	☒ Initial Outlook
	☒ Find an optimal path and move directly to the point.

Figure 3.11: *Environment 3.*

3.7.4 Navigation in a Complex Environment(User Draw)

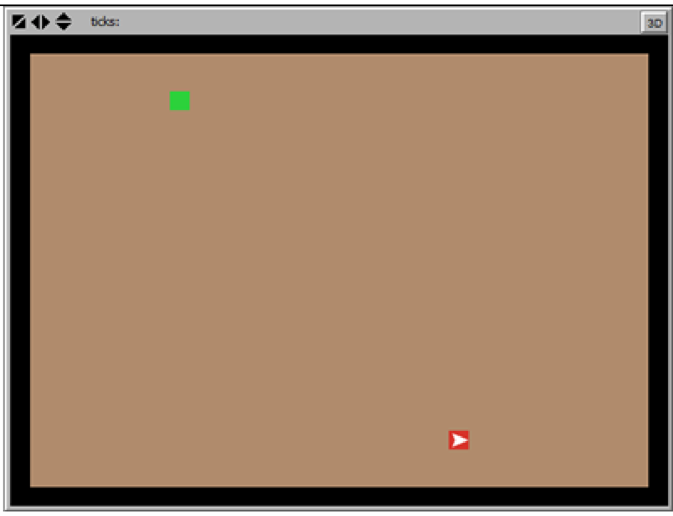
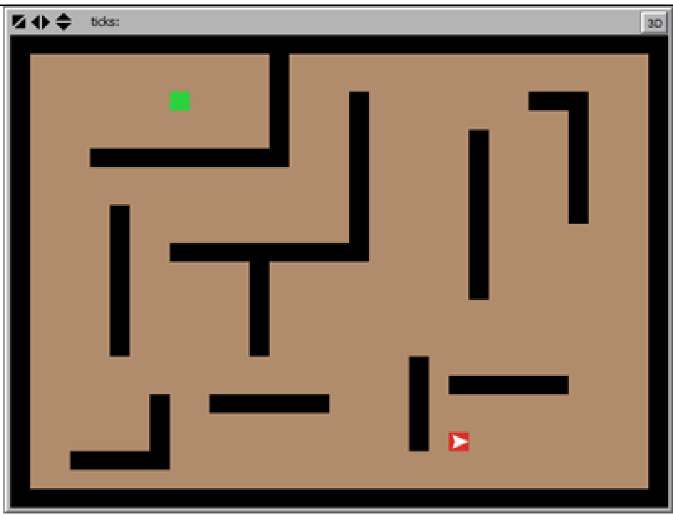
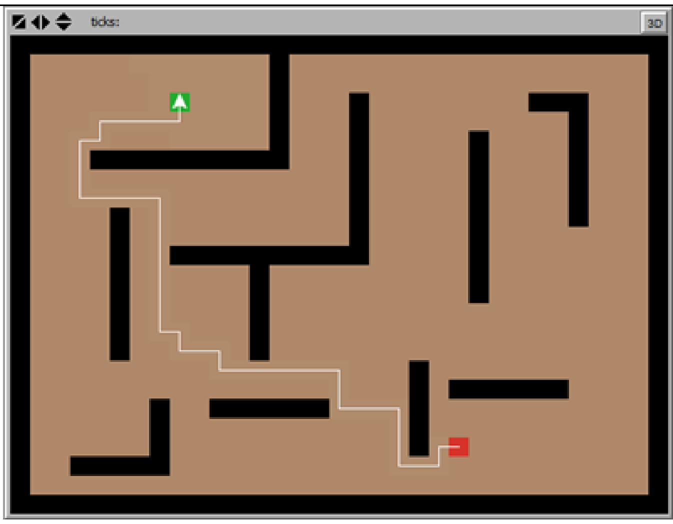
	☒ Initial Outlook
	☒ The user draw the environment.
	☒ Find an optimal path and move directly to the point.

Figure 3.12: *Environment 4.*

3.7.5 Exploration in Complex Environment (3D Simulation)

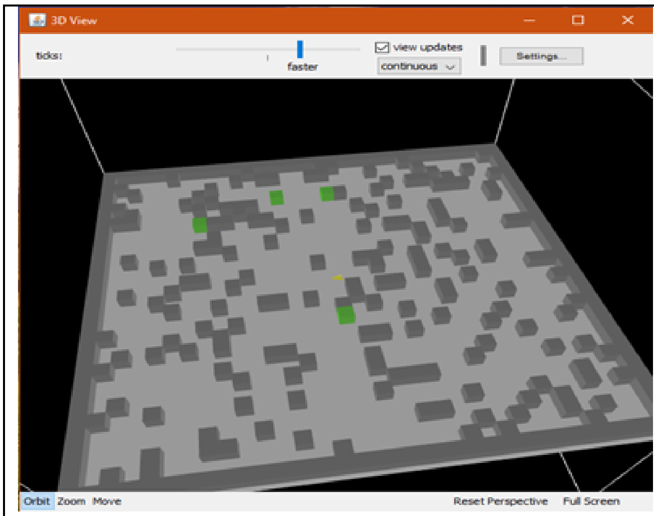
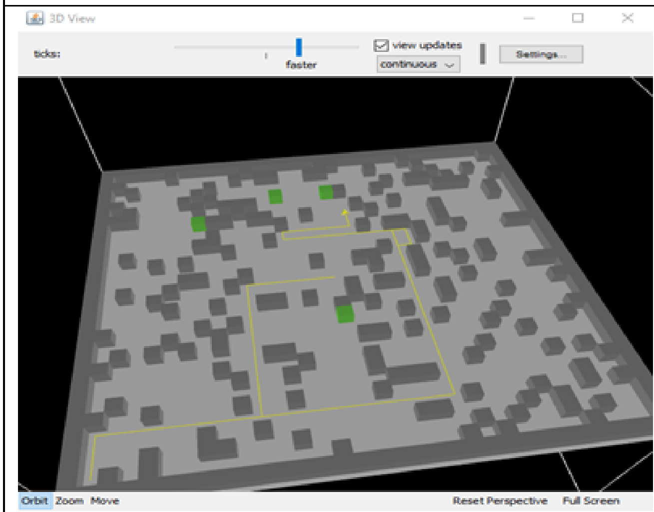
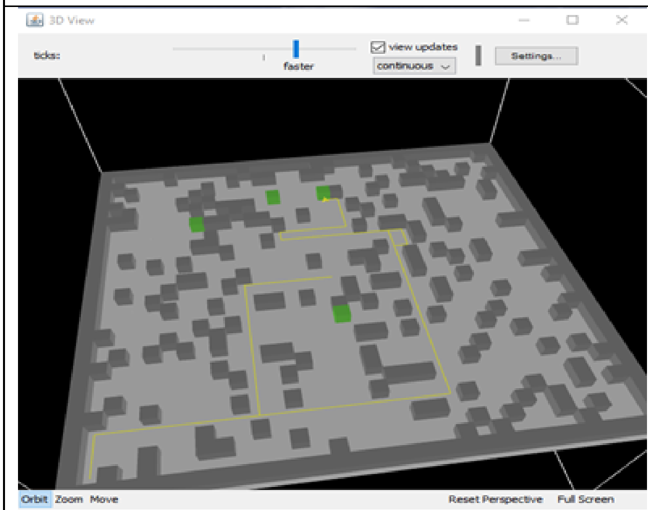
	☒ Initial Outlook
	☒ Move and explore the environment while avoiding obstacles in a random way.
	☒ Find the point in a random way.

Figure 3.13: *Environment 3D.*

3.8 Conclusion

In this chapter we have presented the different components of our software and the relations between them, with the implementation of the procedures used, and some results obtained.

Finally, we come to realize a console for the robot to avoiding obstacles in different environments from under the Netlogo platform.

General Conclusion

General Conclusion

In this report we started the field of research, about robots and their navigation in a complex environment and avoid obstacles by sensors.

Our goal is to make the robot avoid the difficult obstacles within an environment,We have shown the core of our work, where we have described the project, presenting the design of the system, also its implementation, and the results obtained.

The simulation was done in NetLogo 5.3.1 language (preview 2D and 3D).

We have achieved the primary goal of this project (moving and avoidance of obstacles) and the results obtained are encouraging from the point of view that:

- Moving and exploring in a complex the environment: It is done by using sensors of robot.
- Avoid obstacles: go right, left or back.
- Navigation in a Complex environment: use sensors and gps of robot.

Like other possibilities for this work, the robot can provide the ability to move things, or avoid them in other ways (from above the obstacle, underneath it, destroy it ...), or create modules using soft materials that give the robot the ability to change its shape, and adapt to various potential environments.

References

Bibliography

[1] Accessed 26 March 2017,
http://en.wikipedia.org/wiki/Artificial_intelligence.

[2] Accessed 26 March 2017,
https://fr.scribd.com/presentation/337846681/DSMirrorsArtificialIntelligence-ppt.

[3] Accessed 26 March 2017,
http://en.wikipedia.org/wiki/Robotics.

[4] J. Grefenstette and A. Schultz. *"An Evolutionary Approach to Learning in Robots"*. Machine Learning Workshop on Robot Learning, New Brunswick, 1994.

[5] Accessed 26 March 2017,
http://science.howstuffworks.com/robot2.htm.

[6] Marija Seder. *"Hierarchical Path Planning of Mobile robots in Complex Indoor Environments"*. June 2011, University of Zagreb.

[7] Sejal Jaiswal. *"Obstacle Detection and Avoidance by a Mobile Robot"*. May 2014.

[8] Lokendra Singh Khinva Singh Siddhant Rathore. *"Obstacle observing robot"*. May 2016.

[9] Accessed 23 April 2017,
https://www.elprocus.com/robots-types-applications/

[10] J. P. Briot et Y. Demazeau. *"Principes et architecture des systèmes multi-agents"*. Traité IC2, série Informatique et Systèmes d'Information, 2001.

[11] F. Klein. *"Contrôle d'un Système Multi-Agents Réactif par Modélisation et Apprentissage de sa Dynamique Globale"*. Autre [cs.OH], Université Nancy II, 2009.

[12] Accessed 30 May 2017,
https://ccl.northwestern.edu/netlogo/docs/