ÉCOLE NATIONALE SUPÉRIEURE DES MINES DE SAINT-ÉTIENNE

MASTER'S 1 REPORT

Implementation of a robot behavior learning simulator

Author:
Kushagra Singh BISEN

Supervisor:
Dr. Mihaela JUGANARU-MATHIEU

A thesis submitted in fulfillment of the requirements for the degree of Master's 1 in Cyber-Physical and Social Systems

in the

Set of Intelligent Robots Project
Institu Henri Fayol, IT and Intelligent Systems department

August 22, 2021

Declaration of Authorship

I, Kushagra Singh BISEN, declare that this thesis titled, "Implementation of a robot behavior learning simulator" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

“Thanks to my solid academic training, today I can write hundreds of words on virtually any topic without possessing a shred of information, which is how I got a good job in journalism.”

Dave Barry

ÉCOLE NATIONALE SUPÉRIEURE DES MINES DE SAINT-ÉTIENNE

Abstract

Faculty Name
Institu Henri Fayol, IT and Intelligent Systems department

Master's 1 in Cyber-Physical and Social Systems

Implementation of a robot behavior learning simulator

by Kushagra Singh BISEN

The Thesis Abstract is written here (and usually kept to just this page). The page is kept centered vertically so can expand into the blank space above the title too…

Acknowledgements

The acknowledgments and the people to thank go here, don't forget to include your project advisor…

Contents

List of Figures

List of Tables

List of Abbreviations

LAH	**L**ist **A**bbreviations **H**ere
WSF	**W**hat (it) **S**tands **F**or

Physical Constants

Speed of Light $c_0 = 2.997\,924\,58 \times 10^8\,\mathrm{m\,s^{-1}}$ (exact)

List of Symbols

a	distance	m
P	power	W ($J\,s^{-1}$)
ω	angular frequency	rad

For/Dedicated to/To my…

Chapter 1

Context of the Work

1.1 Introduction

The work is based on an Industry 4.0 scenario, which is a cyber-physical environment consisting of various different actors and objects involved. The different actors involved are either stationary or mobile. Moreover, complexity of the environment increases when we account for heterogeneous actors with various decision making capabilities. Robots with various manufacturers present various transform frames, different software and sensors. Due to the heterogeneous nature of the robots involved, we can not depend on information we receive from the robot, as this particular information will differ from a robot to other based upon it's configuration. The problem is solved by creating a digital twin which records the information of the environment as well as the robot. The simulator notes the state of the robot and obstacles it surrounds as it passes through the obstacle grid.

1.1.1 Motivation

The structure of a dynamic Industry 4.0 environment is highly volatile, the structure is defined through a stationary frame that has been declared before. The decision making capabilities of the robot to navigate the environment while avoiding obstacles and other robots can have a great impact on the performance and the utility of the environment.

1.2 Robot

1.2.1 What is a Robot?

The origin of the word robot can be found in Czech playwriter Karel Chapek's play titled "Rossum's Universal Robots (R.U.R)" in 1921. Thw word robot results from combining the Czech words *rabota* meaning compulsory work and *robotnik* meaning an agricultural bound labor. A **robot** is a system existing in a physical world, with decision making capabilities of varying extent, can sense the environment it's in to achieve some goals. A goal can be differ according to the need of autonomous behaviour. Essentially, robot is a cyber-physical system combining sensing, actuation, and computation. With the advancements in technology and materials essential to build a robot, we can see numerous different robots with different applications. Robots such as,

- a self-foldabel / self-actuated robot developed at MIT Sung, 2016
- a lightweight aerial robot developed at University of Penn

- consumer-grade drones by DJI
- Autonomous Vehicles developed at Google.
- Autonomous Surface Vehicles by ASV Global

Robots help humans to do *dirty*, *dull*, and *dangerous* tasks that no human wishes to do, although they are important to be done. As any machines, in an Industry 4.0 environment humans can integrate robots into the development/production process, thus these processes can be optimized. Optimizing robots with different applications can help us to exploit robot technologies to leviate pressure imposed by growing population by using in applications such as,

- mobility-on-demand
- automated highways
- drone swams for servillance
- truck platoons for long distance logistics

Along with these mobile wheel bearing vehicular robots, we hav other robots such as,

- autonomous behaviour on any terrain for search and rescue with Big Dog robots.
- Persobal Robots for help with menial tasks, for example, iCub Robot.
- Emotional Robots wit Human Computer Interface designed to ease the interaction for example, Pepper Robot.

1.3 Autonomous Behavior

For an entity to display auto behavior in an environment, it must be able to model and percieve the world it is in, be able to process information and perform required actions and plan it's behavior in adverse conditions. The level of such autonomy varies with different use cases. These challenges are solved by deploying perception module, action module and decision-making module. These three modules will be mounted and developed on an cyber-physical system, thus differentiating cyber-physical systems in this case with pure artificial intelligence. Architectures employed in Robotics combine the three modules to be used by the developer to develop such CPS systems.

1.3.1 Perception

For a robot to initiate any form important autonomous behavior of decision making, the robot should know where the robot is present in the given Industry 4.0 environment. A robot uses different sensors to infer it's pose in the environment. The different sensors provide measurements of the environment and extract meaningful information for autonomous behavior. Proprioceptive sensor in a robot is used to determine the coordiante location of the robot relative to the frame it is in. These coordinates when changed define the movement of the robot. Another exteroceptive sensor is used to acquire information regarding the environment the robot is currently present in by calculating light intensity and sound amplitude to measure the distance from theh nearest obstacle. The perception module will save the information about the map to be inherited in other modules.

1.3.2 Action

Action module decides the force and orientation for a robot to perform the task assigned. Action module deals with low level control of the robot's motor. In presence of a predefined goal, the action module will calculate the rotational and forward velocities to reach the goal. Action module comprises of various equations responsible to calculate the linear and angular velocities.

1.3.3 Decision Making

In order to achieve a higher order goal, the robot will use the action and perception modules to initiate *navigation* to reach a predefined goal. **Perception** module has provided the neccesary information to the robot about the environment and location of obstacles. **Action** module provides the neccesary equations to calculate the velocities to pursue the motion towards the goal. Delibrative planning is executed by the decision-making module to compute a path that does not collide with the obstacles and respects robot's motion constraints. In real Industry 4.0 environment, we've multiple robots and mobile entities. Collaboration, Communication and Coordination among the robots for path planning to calculate efficient algorithms for calculating linear and angular velocities are an interesting subject for research. For example, collective movement between robots as well as aerial unmanned vehicles such as drones can be initiated by either having a distributed architecture or a centralized leader-follower control. A decentralized system is prone to failure much more than a leader-follower control system. A decision-making algorithm could be written in the leader-follower control system to be in a certain range or perform a certain action around the leader robot. Multi Robot control and coordination allows us to recreate this.

The simplest example of an autonomous robot would be the Roomba robot, which is employed to clean our houses. The robot would use the sensors to infer the world i.e the room around him and initiate navigation to the most suitable location. A simple roomba robot had, 1) A cliff sensor to make sure the robot is not falling off a level field. 2) A bump sensor to retrack it's behaviour and initiate recovery motion. 3) A wall sensor to detect the walls and distance to it. 4) An optical sensor to detect the odometry and force being exerted by the motor. The behaviour displayed by the simple robot is,

- **Wall-Following**: The robot would follow the wall if it bumps into something.
- **Straight** : The robot would go straight and turn a random angle if it bumps into something.
- **Dirty Spot** : If the robot is in contact with a dirty spot, it will spiral around the position untill it bumps into something.

This behaviour is very naive and primitive in relation to what robots can do in current scenario and advancements in the last two decades are huge.

1.3.4 History

The field of robotics emerged from combining and taking influences from various different fields which were present at a time around 1950s. Robotics can see it's influences from,

- **Control Theory** : Control theory Wikipedia contributors, 2021a develops methods for control of dynamic systems and engineered processes in an environment. The objective is to build a model or an algorithm to decide the behaviour of the entity once a particular trigger event (or an action request) has been initiated, the decided behaviour will influence the state transition to drive it into a desired state while minimizing any delay, overshoot, steady-state behavior while ensuring stability often to achieve a degree of optimality. A controller is present in the robot to decide this behaviour. In robotics, the most important part of controk theory is to control the feedback.

- **Cybernetics** : Cybernetics Wikipedia contributors, 2021b is an interdisciplinary approach important to regulate the structure and control of a device. The outcomes of the actions are further fed into the feedback loop. It is the integration of sensing, action and environment.

- **Artificial Intelligence** : In 1950s, artificial intelligence dealt with planning processes and reasoning which integrated to develop into robotics.

Robotics shifted in 1980s, with introduction of new ideas such as 1) reactive contro 2) hybrid control and 3) behaviour based control which produced faster and intelligent machines. Artificial Intelligence's approach to solve a robotics problem shifted from initiating a delibrative planning algorithm such as shortest distance or a checkers playing computer to focus on the basics of what an intelligent machine would look and behave like. By taking inspiration from organisms in nature, it was found that they follow a simple pattern of reactive rules to an external stimuli. This gave rise to reactive behaviours being applied into robotics. Behaviour based robotics was a key concept for the development of robotics at that point of time. We could use this behaviour based robotics concept and employ it in concepts such as swarm robotics and collective intelligence, thus birthing a new research area in robotics itself. As years passed, and we had improvements in computation and hardware design for the sensors, we moved away from the behaviour based robotics. We still see concepts such as wall-following, obstacle avoidance is based upon behaviour based control paradigm. A huge trend recently is in leveraging neural networks and machine learning to improve the perception and control of the robot. Researchers are trying to integrate machine learning end to end in the robot itself promoting it's intelligent behaviour. As we use the robot's as a blackbox, with an end to end architecture it is very difficult for us to understand why a robot is doing what it is doing at a certain position. In our work, we are trying to record this behaviour of the robot while considering it as a black box with an end to end architecture and learning it's behaviour by making a digital twin.

1.4 Robot Control Architectures

The building block of an autonomous system such as a robot is a perception-action loop. Perception-Action loop is a cycle for the robot to process the thing that he experiences and the things that he does, simultaneously improving the decision making process of the robot. The robot is in a continous interaction with it's environment.

- **Reactive Control** : Reactive Control deals with using the sensors to present the current estimate of the world, recovery behavior rules in case of collision produce actions which are simple and fast to compute.

- **Delibrative Control** : Delibrative Control deals with predicting the future state of the robot, developing a plan for the same for the robot to decide the *sequence of actions* to pursue in order to follow the sequence. In our work, we have used delibrative planning to move the robot from a starting position to the goal position via different algorithms, for example A*, Dijkstra and Greedy Best First Search Algorithm is used to compute the path in the work.

Complicated actions require complicated control architectures, which combine the three elemental modules of autonomy which are perception, plan and action. Examples of such control architecture are,

- **Finite State Machines** : They are reactive and follow a sequential path. Consisting of a finite set of states and transitions between these states. A simple example in this case would be 'pick up the trash' robots.
- **Subsumption Architecture** : They are reactive and follow a concurrent path, which means that the three modules of perception action and decision making will execute simultaneously.
- **Sense-Plan-Act** : They follow a delibrative path planning method which use the modules in a step by step way.

1.5 Motion Control in Robots

Motion Control in a robot deals with the process through which the delibrative or reactive control is passed down to actuators of the robot and finally to the motor resulting in the physical change of state in a direction robot's decided to move to. The action component of the autonomy controls the robot with the motion control module.

1.5.1 Actuators

Actuators are used to convert a signal from a circuit board, e.g. from a Raspberry Pi or an Arduino into physical mechanical motion. Actuators serve various purposes, such as Locomotion, Manipulation, heating and sound emission. Other examples of electrical-to-mechanical actuators are, DC Motors, stepper motors and loudspeakers. *For example,* in a normal example of an automated car a driver can steer and acclerate, thus making 2 control points. There are examples of for uncertainty and noise in actuators. Examples of such are wheel slip, slack in the motion and environmental factors such as wind, friction and other natural uncontrollable distractions. Actuators influence a certain single degree of freedom of the robot. A robot is controllable in all of it's degree of freedom if it has an actuator at each and every degree of freedom. *Degree of Mobility* is the number of DOFs which are operable by the actuators. If the DOF is equal to robot's DOM the robot is a *holonomic robot* and if the number of DOF is greater than it's DOM then it is a *non-holonomic robot*. Otherwise, if robot's DOM is more than it's DOF, the robot's actuation is redundant. The robot in our project is a **Differential Drive Robot** which can actuate with left and right wheels without any interference. DDR has 3 DOF but DOM is 2, differential drive robots are *non-holonomic* in nature.

Kinematics

Kinematics are important for us to infer and calculate the motion, i.e behaviour of the robot when it reaches a goal. Forward Kinematics decide where the robot would end up in the coordinate frame of the environment (x, y, θ) given that we have the control parameters and the time of movement. Reverse kinematics deal with finding the control parametetrs of the robot once the desired final pose (x, y, θ) is provided.

Appendix A

Frequently Asked Questions

A.1 How do I change the colors of links?

The color of links can be changed to your liking using:

`\hypersetup{urlcolor=red}`, or

`\hypersetup{citecolor=green}`, or

`\hypersetup{allcolor=blue}`.

If you want to completely hide the links, you can use:

`\hypersetup{allcolors=.}`, or even better:

`\hypersetup{hidelinks}`.

If you want to have obvious links in the PDF but not the printed text, use:

`\hypersetup{colorlinks=false}`.

Bibliography

Sung, C. (2016). "Computational design of foldable robots via composition". In: URL: https://dspace.mit.edu/handle/1721.1/113734.

Wikipedia contributors (2021a). *Control theory — Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/w/index.php?title=Control_theory&oldid=1039308545. [Online; accessed 21-August-2021].

— (2021b). *Cybernetics — Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/w/index.php?title=Cybernetics&oldid=1037814219. [Online; accessed 22-August-2021].