VOL. 21, NO. 2, 2022, 33-41
www.elektrika.utm.my
ISSN 0128-4428

ELEKTRIKA
Journal of Electrical Engineering

Modeling and Simulation of Autonomous Mobile Robot System for Collision Avoidance in VR Environment Using Simscape-Multibody

Hasin Abrar Biswas, Sophan Wahyudi Nawawi[*]**, Abd Alati Ahmed** and **Mohammed Alih Jibrin**

School of Electrical Engineering, Universiti Teknologi Malaysia

[*]Corresponding author: sophan@fke.utm.my

Abstract: Autonomous mobile robot has become a center of attention in the robotics community recently due to its ease of design, modeling and implementation which makes it very lucrative in industries for logistics, monitoring, mapping and other purpose. Fundamentally, collision avoidance is a major aspect of autonomous mobile robot which requires assistance from other aspect like sensor fusion, localization, path planning etc., to contribute in achieving proper functionality in an open environment hence the inherent required integration makes it hard to many researchers, robotics and robot enthusiasts to incorporate multi-dimensional functionalities in their system. Sometimes traditional software for robotics makes it harder to even acquire actual insight of their system due to lack of cross platform compatibility. Therefore, this research aims to demonstrate a way to integrate important aspect required for obstacle avoidance system for design and analysis using a compatible system which is solidworks and MATLAB simscape multibody which provides sufficient scope to properly and accurately simulate autonomous system on real time basis that can be used with actual hardware to final implementation. The findings of the study demonstrates that it is very much feasible to use such method for robotic and other complex mechanics applications and further analysis on the sensor fusion algorithm reveals some short coming of the sensor fusing algorithm and data processing which can be avoided by introducing further sensor sequencing and implementing adaptive neuro fuzzy based system.

Keywords: Autonomous mobile robot, Collision avoidance, VR simulation, Sim-Multibody

Article History: received 30 May 2022; accepted 8 August 2022; published 25 August 2022.

1. INTRODUCTION

The autonomous mobile robot is a type of system which operates in a partially unknown or unpredictable environment. This implies that the robot has to have the capability of navigate disruption free while avoiding any obstacle found with in the vicinity. One of its other features is the ability to operate without direct human intervention. Depending on the environment the mobile robot is operated, it can be either indoor or outdoor. Outdoor environment complies with much more uncertainty and unpredictable terrain behavior whereas indoor environment is relatively predictable. For indoor work frame the robot attempts to navigate based on floor plan, sonar sensing, Inertial Measurement unit and other sensors. For outdoor operation, environmental sensors are used. The sensors are mounted on the robot or also communication route can be made between the robot and sensor positioned in the environment itself. Therefore, the varied types of sensors, mounting positions, data fusion from different type of sensors which is used to estimate the robot state for localization, navigation and path planning makes it difficult task. In The fundamental problems in the filed of mobile robotics can be divided into four parts. These are, locomotion, perception, navigation and obstacle avoidance [1] [2].

Locomotion is the study of the robot on how to move from one place to other. It does not only depend on the medium the robot is travelling in but also other aspects like controllability, maneuverability, terrain conditions, stability and efficiency and many other [3]. The specifications of these aspect are depended on the operating conditions and operating media. Mobile robot's movement on air, land or water dictate what sort of locomotion mechanism would be preferred. Also, the specifics of the environment are also necessary. For example, even among the mobile robot operating in land there are many mechanisms for locomotion such as leg-based locomotion and wheel-based locomotion. Each have their own benefits. Leg based are suitable for rough terrain with many obstacles. Such terrain contains uneven ground, unsmooth and stony surfaces. One disadvantage of legged locomotion is the complexity of the mechanism and hence requirement of high computational requirement with additional platforms and equipment. On the other hand, wheel-based locomotion is one the most used method in mobile robotics due to its inherent simplicity for design and to maintain stability. One major advantage is the ease of maintaining balance using wheels. Some major disadvantages would include the inability to operate properly in stony, uneven and unsmooth grounds. To ensure a robust type of locomotion in both types of terrain, hybrid locomotion has been designed and developed for testing where both legged mechanics is involved with wheels [4]. Such form of hybrid locomotion is shows promising result on accommodating changes in terrain characteristics to achieve stability.

Perception is a very fundamental aspect of any kinds of robotics applications. It entails the idea of acquiring the data from environment, sense object around specified zone and around the body of the autonomous robot itself. The success of the key task of the autonomous mobile robot lies in the accuracy of being able to sense the key feature of the environment and object around it [5]. To acquire this, different kinds of sensors are used and the data of the sensors for processes in different ways to make sense of the data as accurately as possible [6]. Sensor used for such application are typically segmented into two ways. This includes prospective sensors and active sensors. The prospective sensors are used to measure values that are internal to the robotic system like wheel position, motor speed, joint angle, battery level etc. Typically, these sensors are encoders, gyroscopes, potentiometers, gyroscopes, compasses etc. Exteroceptive sensors include Infrared sensitive sensors, Sonar sensors, ultrasonic sensor. These types of sensors emit energy to the environment to acquire information about the environment as the response to robotic action to the environment. These sensors can achieve good performance for the ability to mange the interactions with the environment. Based on energy emission and receiving, sensors can be divided into two. Active sensors are sensors that emit energy to gather information from the environment whereas passive sensors receive energy from the environment without any emission of energy. An issue of interference may arise for the active sensors between signal and environment [7]. Passive sensors like camera, Complementary metal oxide semiconductor (CMOS), Temperature sensors etc. are also good choice for autonomous mobile robots.

Navigation is another fundamental part of autonomous mobile robots. It is the process of the localizing of the mobile robot from the sensor data and the desired destination of the mobile robot and process of reaching the destination under some spatial and temporal constant. Additional dynamic states and stability can also be considered as constraints within navigation framework. The fundamental objective is the navigate from one specified position to another in either recognized, semi recognized or unrecognized environment. In most cases a mobile robot can not take linear and direct path towards the destination which calls for additional steps like path and motion planning, trajectory tracking etc. This implies that this process is dependent on other fundamental aspect of autonomous mobile robot such as perception which includes data acquisition and processing from different kinds of sensor that are part of the system, localization which implies the position and orientation of the robot, cognition which implies decision making i.e. path and motion planning and how to achieve the goal and finally motion control which entails giving proper input to the actuators to achieve the goal by estimating and anticipating probable changes in motion states. Other more specified aspects include computer vision aided navigation and simultaneous localization. Object recognition and feature matching are two areas of computer vision for navigation. Object recognition is used to identify position of landmark in the environment which can give the mobile robot the associated coordinate or it can also be used to measure the relative position between the robot and the landmark. On the other hand, feature detection can be used select and identify key feature in the map like edges, lines and surface shapes which helps the robot to map the environment [8][9]. Typically, three approaches are used to represent the environment which includes geometric approach, topological approach and semantic approach [10]. Geometric approach describes the surrounding by parameterizing the fundamental geometric features like lines, curves and points. It represents the environment closer to the sensor and actuator and provides the best local navigation. In [11] a principal components analysis (PCA) – Bayesian based method with grid map was used to compress the images and computational data for mapping. Topological approach includes defining reference elements of environment using distinct relation between objects within the environment by discretization. In [12] a stereo-based feature extraction technique was demonstrated while adopting topological methodology by fusing and removing new and transient landmarks over time. Semantic approaches aim to design robot representation of environment by imitating the way human understands the environment. This is done by providing semantic model of the environments to the robot where the robot has the autonomy to control its' motion hence discission making process is wider in such case and therefore more robust [13].

Obstacle avoidance is one of the central themes of autonomous mobile robot. It is vital due to it directly helps the robot to achieve the target goal with being obstructed by any obstacle or an event. It can also the called as collision avoidance system on the path of robot determined by the navigation process. Therefore, collision free design of algorithm is a requirement of mobile robot deployed for any task as it ensures safe trajectory and identifies convergence of the designed system [14]. Some of the primary algorithms used for such tasks are bug algorithm [15], vector field histogram, [16], virtual force field algorithm [17] etc. Bug algorithm is one of the earliest algorithms developed for obstacle avoidance. It provides the ability to the robot to navigate around an entire circumference of the obstacle faced and decided by the perception unit in reaching the desired position. It chooses the best path possible to navigate around the obstacle to move towards the goal. This algorithm is the easiest to implement. Vector field histogram (VFH) is an improvement on vector force field algorithm (VFF) which allows detection of unknow hindrance and obstacle in the path towards the goal. It deploys 2-stage data reduction process to compute the appropriate pathway for the robot but takes a lot of computational resources. Hybrid navigation algorithms (HNA) are also deployed as it offers very efficient performance of the robot in terms of motion states. The robot might deviate from the planned path when faced with an obstacle on its' path based on reactive navigation strategies, this however doesn't imply exit movement of the robot due to obstacle. The obstacles can come in form of dynamic and static state of their motion. As the robot moves in a convex area the static obstacles are sure to be avoided by the robot by the algorithms but the same is hard to acquire for dynamic obstacles. Similar to this process another method is new hybrid navigation Algorithm (NHNA) which also demonstrates good result [18]. To proper understand the performance of the robot in collision avoidance several factors have to be in taken account of like the types of sensors used, dynamics of the robot, sensor fusion and sensor data processing and control logic that dictates the movement of the robot upon detecting an obstacle. Often time it is possible to achieve good obstacle avoidance of a robot by simulating it in VR environment which saves a lot of time and gives more flexibility on understanding the sensor behavior and process. For such task software packages like Robot operating system (ROS) with Gazebo framework is readily available.

Some works using such softwires would include research found in [19], where the authors conclude a suboptimal solution for such strategy. Review on the available method found in [20], [21], [22] reveals that the metaheuristics method-based control actions are the most popular obstacle avoidance. In [23] an analysis of the trajectories was formed by TEB planner and the data obtained allowed determining the error between global and local plan in robot navigation with obstacle avoidance. In [24] it was suggested that parameter indicating the distance from which the robot begins to bypass the obstacle, the shortest distance from it and the time required to reach the target can be used as comparison factors for local planners. In [25] a turtle bot was used in ROS environment which compares the global RRT planner with Dijkstra algorithm form ROS. Similar studies were undertaken to investigate the performance of A* algorithm. Although most study attempts to create real time simulation of the robot in ROS, a unified platform like multibody provides the ability to analyze and develop the robot model in a single Simulink platform which is very beneficial due the robustness of Simulink and the library that comes with it which can be used to analyze the overall robot design and performance of the system on real time bases. Hence, this study aims to design and simulate the performance of a pioneer 3dx model robot in multibody in a VR environment with sonar sensors being the main sensor to identify the obstacle. The simulation is carried out real time and the control logic is designed in state flow to demonstrates real time changes.

2. METHODOLOGY

2.1 Scope of research

The following research demonstrates design and simulation collision avoidance of pioneer 3dx model robot in VR environment using sim-Multibody. Here, only static environment is taken into consideration and a basic sensor fusion technique is deployed. The control logic for the collision avoidance is considered as instantaneous control logic and not a continuous controller. For sensor configuration only sonar sensor was used.

2.2 Modelling and simulation process

The following demonstrates the flow of the design and implementation

- **CAD model preparation**

The CAD model of the robot pioneer 3DX is demonstrated as in Figure 1.

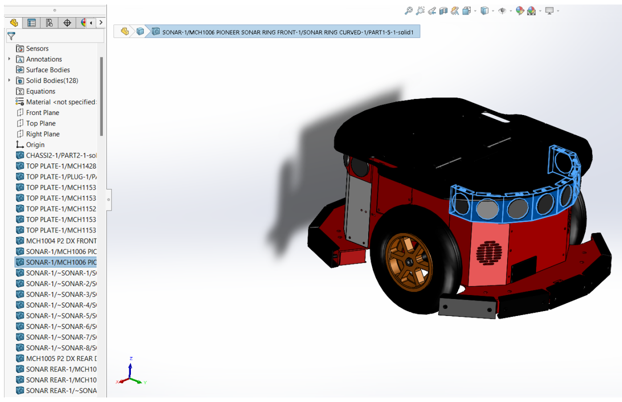

Figure 1. CAD assembly

The CAD model is the assembly of separate parts of the pioneer 3DX robot model as demonstrated. In CAD assembly proper joint characteristics have to be assigned in order to ensure correct representation in the simscape multibody environment. Main points to consider at this point is to reduction the number of assembly parts by combining multiple parts with the similar function as one single part. For example, wheel, axile and rim can be modelled as single wheel sub-assembly and thus minimizing the total number of parts in the assembly. The graphic setting should also be considered and adjusted as it requires additional computing for high resolution model. Most importantly, the mate and joint characteristics of the parts with main body has to be assigned properly. Following the process, the model derived is as demonstrated in figure 2.

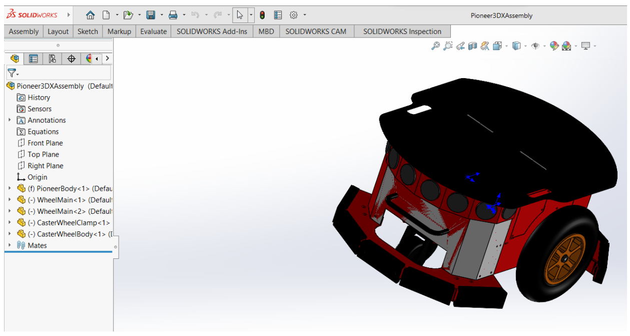

Figure 2. modified CAD model

- **Importing CAD model to sandscape environment**

After achieving the modified CAD model, the file has to be saved as .XML file format which is then opened by MATLAB simscape multibody and demonstrated in figure 3.

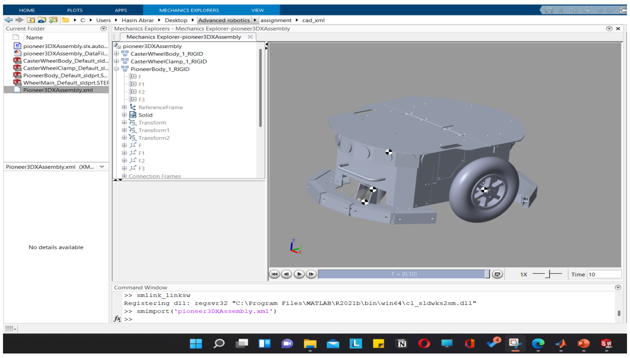

Figure 3. .XML file in simscape environment

The CAD assembly files and part files has to be in the same folder as the .XML file. The folder is then selected as opening path in MATLAB.

From the .XML file we can create a Simulink model of the CAD in form of block diagram as presented in figure 4.

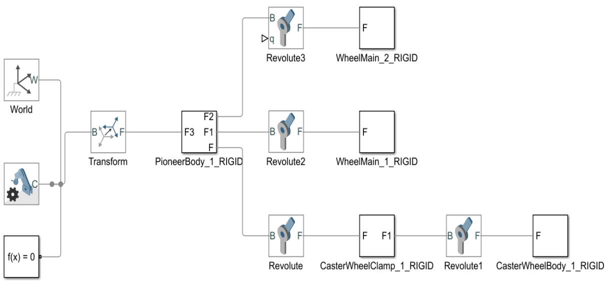

Figure 4. Block diagram presentation of the robot

- **Testing CAD feature and verifying simscape model.**

In order to ensure the correct joint and mate characteristics have been translated properly in the simscape multibody environment some basic test with respect to each body mechanics of the robot is carried out and upon finding error the model is updated. The testing process includes the following tests:

1. No actuations to wheel angular velocity – no movement of the robot
2. Adding actuation to wheel angular velocity – noticeable robot movement and displacement depends on actuating wheel which includes the following:
    a. $W_r = W_l > 0$, straight forward
    b. $W_r = W_l < 0$, straight reverse
    c. $W_r = -W_l > 0$, Turn left in place
    d. $W_r > W_l$, curve left
    e. $W_l > W_r$, curve right

The figure 5 demonstrates actuation to left wheel in the block diagram

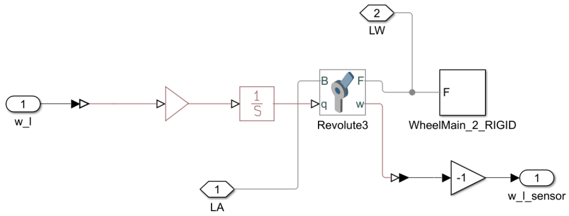

Figure 5. Adding actuation to both wheels

The joint and associated rigid body can be joined together as block and actuation can be directly actuated. The figure 6 demonstrates the wheel block diagram with equal actuation to both wheels:

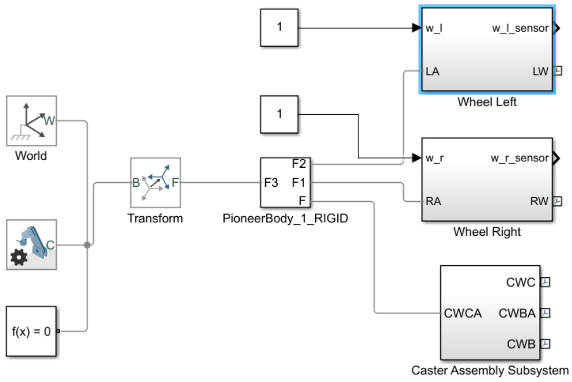

Figure 6. Equal actuation to wheel block

- **Adding kinematics relation to outer world with robot model**

The kinematic relation has to be established of the robot body with to translate the dynamic behaviour of the inner robot mechanism with the environment and vice versa. In order to do that proper parameter which dictates the robot behaviour of the system has to be identified and mathematically represented. For the current system at hand, the robot wheel velocity and orientations are off parameter of interest which dictates the motion of the robot acts on the environment. The figure 7 demonstrates the mathematical derivation and free body diagram of the robot.

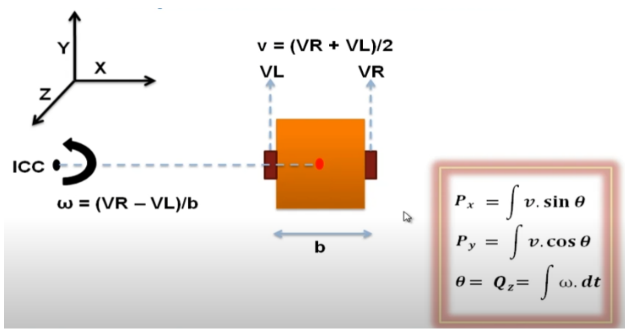

Figure 7. FBD and kinematic equation derivation [14]

Based on the kinematic relation a Simulink model can be derived as shown:

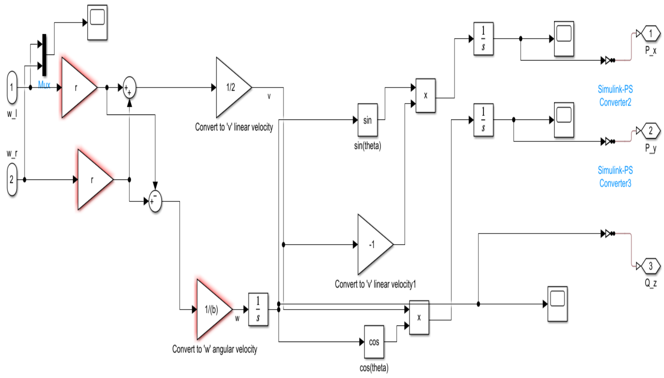

Figure 8. Kinematic subsystem

The kinematic subsystem here translates the inner rigid body mechanics of the robot to the environment to demonstrates dynamical behaviour in the reference frame considered. Therefore, the kinematic system is integrated with the overall robot block diagram as demonstrated in figure 9:

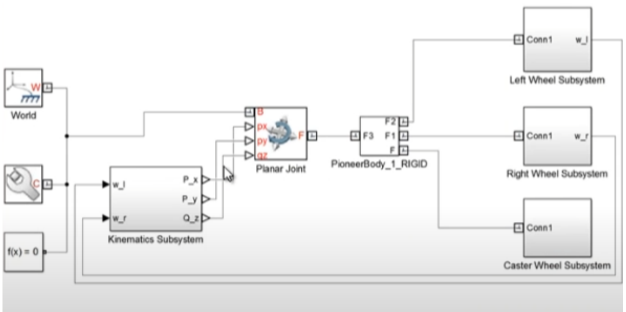

Figure 9. Kinematic subsystem integrated with the robot model

- **Actuator assembly with controller**

To ensure stability in the actuator performance a controller block has to be added. A simple control block is integrated with the actuator as demonstrated in figure 10:

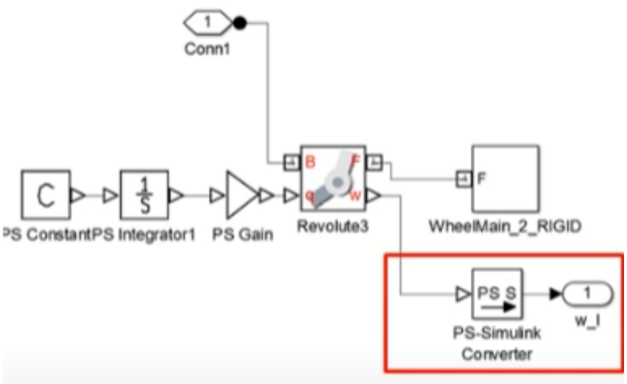

Figure 10. Control implementation in the actuator

The controller block is a simple PI controller where the proportional and integral gain values have been selected based on the manual tuning and the gain values are selected based on the maximum actuator output values. To ensure stability, the gain values are kept within the range such that, the control signal does not cross maximum actuator output.

- **Importing CAD model as VRM**

The modified CAD model has to be exported as VRM file and imported by Simulink 3D animation environment. Upon imported in the 3D environment it is investigated further. The figure 11 demonstrates the imported 3D model

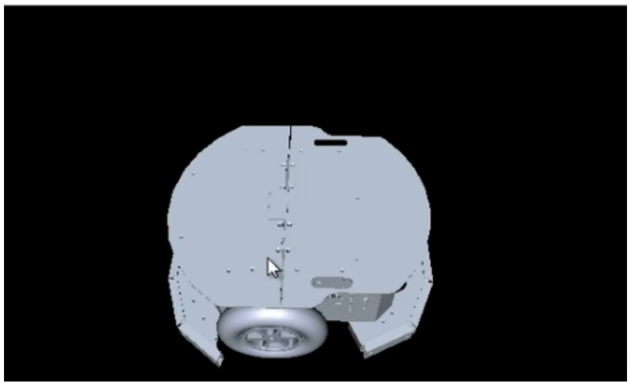

Figure 11. VRM file imported in the 3D environment

Once VRM file is imported a 3D environment has to be selected in the Simulink 3D animation block. For this case a VR file from MATLAB official site has been used. VR environment is static and connected to the robot block as demonstrated in figure 12.

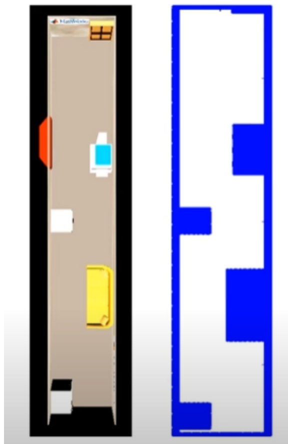

Figure 12. VR environment

Robot model is then imported in the VR environment as demonstrated in figure 13.

- **Designing Sensor and senor peripheral**

The robot has eight sonar sensors mounted on top in eight direction each thirty-degree angle apart from each other starting from ninety degrees to counter clockwise ninety degrees thus the robot sensor covers 180 degrees view from the front. The figure 14 demonstrates the sensor distribution around the robot model.

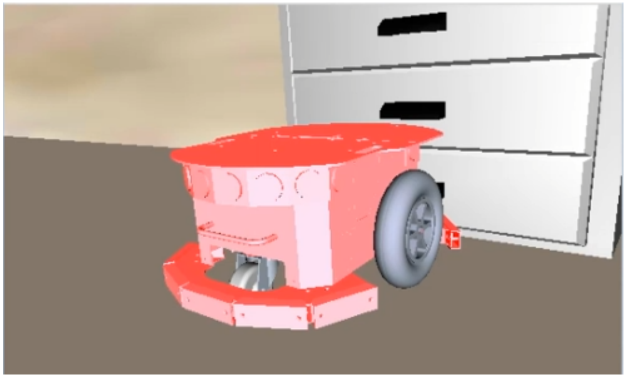

Figure 13. Robot in VR environment

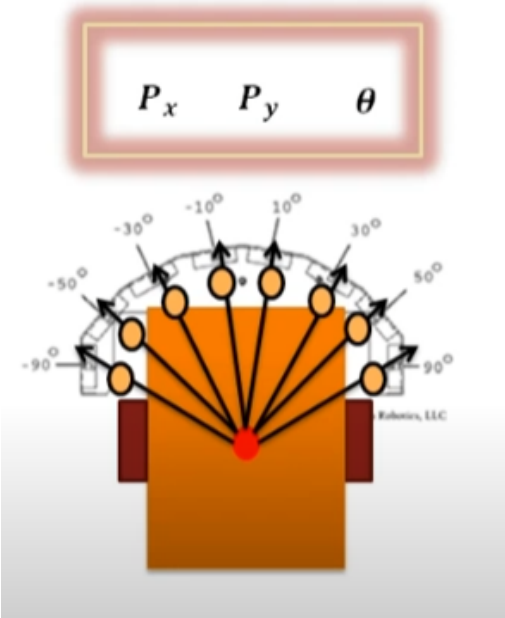

Figure 14. Sensor configuration [14]

The robot has eight sonar sensors mounted on top in eight direction each thirty-degree angle apart from each other starting from ninety degrees to counter clockwise ninety degrees thus the robot sensor covers 180 degrees view from the front. Sonar sensor used to as it provides sufficient performance for indoor use. The sensor senses when an obstacle is found and update the control logic and the control logic chooses an appropriate action by the actuator depending on the sensor input. The sensor position and orientation are also included with the sensor data. The figure 15 demonstrates this.

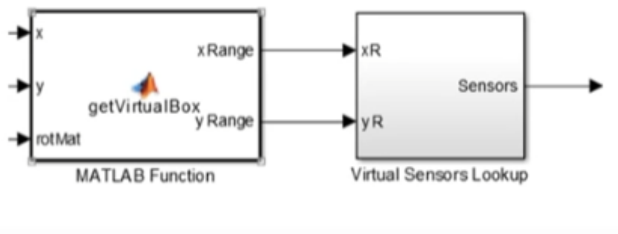

Figure 15. sensor block diagram

- **Designing and connecting control action with Sensor**

The eight-sensor based on their input range can be fused together to dictate specific control action for the robot. For example, if the robot is facing an obstacle right in front from the input of the sensors mounted at the front the best course of action for the robot will be to reverse and similarly the sensor orientation and input would dictate the control action of the robot which is represented by the figure 16:

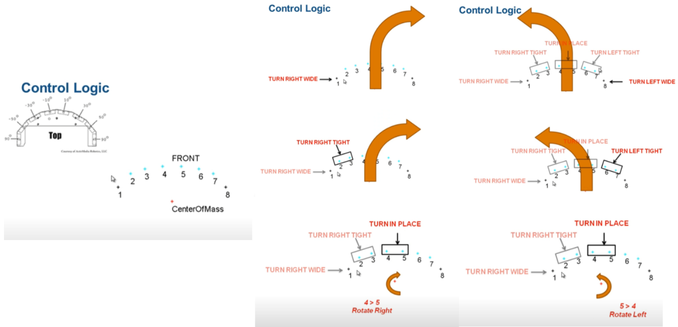

Figure 16. Control action design based on sensor output [14]

The following actions are then modelled as a dynamic state of the robot for a given sensor input which is designed in state flow. The sensor fusion is implemented using simple mathematical inequality relation of range and orientation which is as shown by the figure 17. The stateflow model is shown in figure 18.

```
frontSensors = SonarSensors(4:5) < SensorRanges(4:5);
leftMostSensors = [SonarSensors(1)] < SensorRanges(1);
leftMiddleSensors = [SonarSensors(2:3)] < SensorRanges(2:3);
rightMiddleSensors = [SonarSensors(6:7)] < SensorRanges(6:7);
rightMostSensors = [SonarSensors(8) ]< SensorRanges(8);
```

Figure 17. Inequality-based sensor fusion

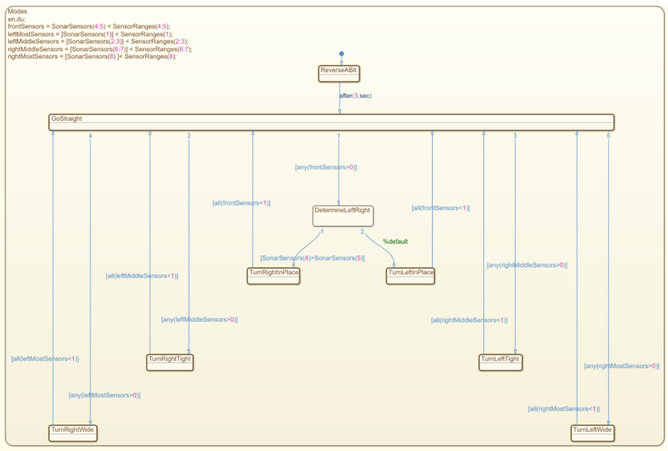

Figure 18. State flow model of control law

The overall model of the robot in form of block diagram is as demonstrated in figure 19.

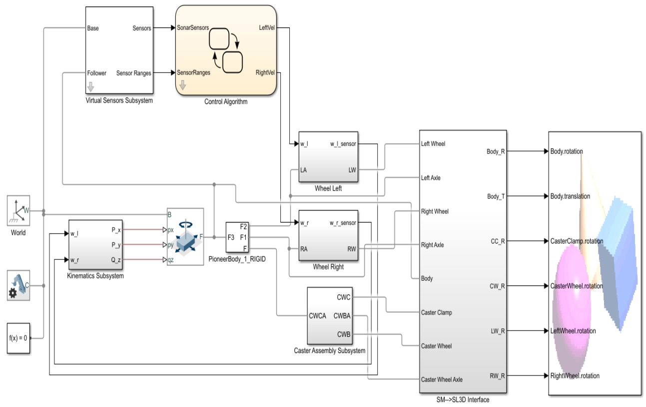

Figure 19. Overall model

3.0 RESULT AND DISCUSSION

The result of the system is evaluated on the control law based on the sensor fusion demonstrated in the previous section. The sensor demonstrates the inputs with overlapping based on the obstacle found at front which is demonstrated in figure 20.

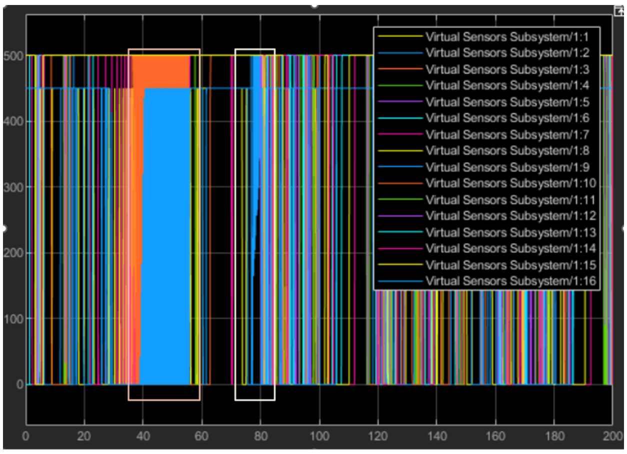

Figure 20. Sensor output

The sensor output from the senor block demonstrates the sensor data from vlook up function being called and integrated to for a possible obstacle position during the simulation. The data demonstrates two regions based on density. The denser region in the graph demonstrates situation where multiple sensors of different region simultaneously gave input of varied obstacle position in a coordinate which is properly categorized in the sensor fusion algorithm which made it harder for the control law to dictate the appropriate control action for the given situation and on the other hand the mid dense regions show relatively discrete condition for control law to act on actuator due to ease of decoding form sensor data. In addition, the upper and lower bound in the sensor look up data with respect to time suggests an amount of delay between multiple sensor data input of different kind such delay of sensor data along with obstacle position being a narrow range of the sensor fused region which caused the mobile robot being undecisive about the control action and hence the control action takes very quick frequent state changes with respect to its dynamics in order to properly identify the obstacle position relative to robot motion. This result in the robot moving left and right with varied angular speed of the actuator. Thus, an unstable behaviour can be observed during this part of the simulation by the robot. Therefore, the actuator of the robot also shows the effect of this, as figure 21 demonstrates the state changes with regards to sensor data input shown in figure 20. From, figure 21 it is observable how the state dynamics of the robot changes in the dense region in figure 20. The frequent change in sensor data with regards to obstacle position results in the frequent change in the left and right wheel actuation states demonstrated by the circle in figure 21. Although the overall figure shows good stability and performance between right and left wheel based on the associated sensor input data by sensor fusion algorithm. The denser region depicts the states where sensor data could not yield a appropriate response for a given obstacle and hence the wheel actuation frequency was high and seeming irregular. The lesser dense region on the graph in sensor input corresponds to the lesser changes in state dynamics of the robot as demonstrated in the figure 21.

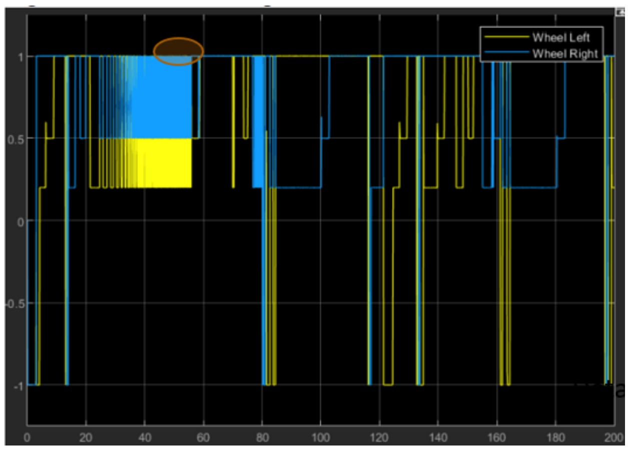

Figure 21. Dynamic state change

The controller on the actuator was a simple controller which cannot address this kind of non-linearity demonstrated by figure 22 which was found by focusing on the figure 21. Therefore, closer look in the circular region shows the imbalance in left and right wheel demonstrated by figure 22

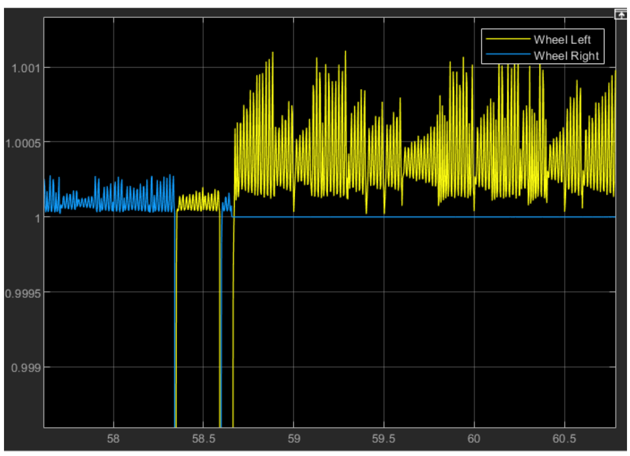

Figure 22. Left and right wheel controller in real time

The figure shows highly non-linear behaviour by the left wheel which is also noticed in the real time simulation. The control law could be design using fuzzy logic to address the dense fuzzification on the sensor input when multiple sensors simultaneously provide data. Better controller should be implemented to address such issue. A fuzzy based PID control addressing to such conditions can be specially help in such situations. In addition, an Adaptive Neuro fuzzy system (ANFIS) can be developed and integrated with the locomotion controller the system. Such configuration can be useful to deal with the nonlinearity of the system real time as well as can address proper tuning parameters for the actuator. Another method to resolve this issue could be sequencing the sensor data with deliberate delay which would make the response of the robot slower but it will benefit the robot in determining reacting to obstacle in more subtle and stable manner.

4. CONCLUSION

The overall objective was to model and simulate a mobile robot for collision avoidance in a MATLAB simscape Multibody for its better integrability, which is was achieved and demonstrated by the papers. Better control law in form of fuzzy PID could have been developed to address the non-linearity and better performance. In addition, the sensor sequencing with known delay can also be adopted along with adaptive neuro fuzzy system to address the uncertainties.

5. REFERENCES

[1] D. G. Lowe, ``Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91_110, Nov. 2004.

[2] M. B. Alatise and G. P. Hancke, "A Review on Challenges of Autonomous Mobile Robot and Sensor Fusion Methods," in IEEE Access, vol. 8, pp. 39830-39846, 2020, doi: 10.1109/ACCESS.2020.2975643.

[3] F. Rubio, F. Valero, and C. Llopis-Albert, ``A review of mobile robots: Concepts, methods, theoretical framework, and applications," *Int. J. Adv.Robotic Syst.*, vol. 16, no. 2, Apr. 2019, Art. no. 172988141983959

[4] G. Endo and S. Hirose, ``Study on roller-walker (multi-mode steering control and self-contained locomotion)," in *Proc. ICRA. Millennium Conf. IEEE Int. Conf. Robot. Automat. Symposia*, vol. 3, Apr. 2000, pp. 2808_2814.

[5] D. Kortenkamp, ``Perception for mobile robot navigation: A survey of the state of the art," NASA, Houston, TX, USA, Tech. Rep. 576344779,May 1994, pp. 446_453.

[6] B. Silva, R. M. Fisher, A. Kumar, and G. P. Hancke, ``Experimental link quality characterization of wireless sensor networks for underground monitoring," *IEEE Trans Ind. Information.*, vol. 11, no. 5, pp. 1099_1110, Oct. 2015.

[7] M. Gilmartin, ``Introduction to autonomous mobile robots," *Robotica*, vol. 23, pp. 271_272, Mar. 2005.

[8] D. G. Lowe, ``Distinctive image features from scale-invariant keypoints,"*Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91_110, Nov. 2004.

[9] R. Lagisetty, N. K. Philip, R. Padhi, and M. S. Bhat, ``Object detection and obstacle avoidance for mobile robot using stereo camera," in *Proc. IEEE Int. Conf. Control Appl. (CCA)*, Aug. 2013, pp. 605_610.

[10] R. Barber, J. Crespo, C. Gomez, A. C. Hernamdez, and M. Galli, *Mobile Robot Navigation in Indoor Environments: Geometric, Topological, and Semantic Navigation*. London, U.K.: IntechOpen, Nov. 2018, pp. 1_25.

[11] F. Shamsfakhr, B. Sadeghi Bigham, and A. Mohammadi, ``Indoor mobile robot localization in dynamic and cluttered environments using artificial landmarks," *Eng. Compute.*, vol. 36, no. 2, pp. 400_419, Mar. 2019.

[12] J. A. Cetto and A. Sanfeliu, ``Topological map learning for a mobile robot in indoor environments," in *Proc. Spanish Symp. Pattern Recognit. Image Anal. (SNRFAI)*, Jan. 2001, pp. 221_226.

[13] J. Crespo, R. Barber, and O. M. Mozos, ``Relational model for robotic semantic navigation in indoor environments," *J. Intell. Robotic Syst.*, vol. 86, nos. 3_4, pp. 617_639, Jan. 2017.

[14] M. Mouad, L. Adouane, D. Khadraoui, and P. Martinet, ``Mobile robot navigation and obstacles avoidance based on planning and re-planning algorithm," *IFAC Proc. Volumes*, vol. 45, no. 22, pp. 622_628, 2012.

[15] K. Shahzad, S. Iqbal, and P. Bloodsworth, ``Points-based safe path planning of continuum robots," *Int. J. Adv. Robotic Syst.*, vol. 12, no. 7, p. 107, Jul. 2015.

[16] W. J. Yim and J. B. Park, ``Analysis of mobile robot navigation using vector _eld histogram according to the number of sectors, the robot speed and the width of the path," in *Proc. 14th Int. Conf. Control, Autom. Syst. (ICCAS)*, Oct. 2014, pp. 1037_1040.

[17] G. Li, A. Yamashita, H. Asama, and Y. Tamura, "An efficient improved artificial potential field based regression search method for robot path planning," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, Aug. 2012, pp. 1227_1232.

[18] Y. Zhu, T. Zhang, J. Song, and X. Li, ``A new hybrid navigation algorithm for mobile robots in environments with incomplete knowledge," *Knowl.-Based Syst.*, vol. 27, pp. 302_313, Mar. 2012.

[19] Chaari, I., Koubaa, A., Bennaceur, H., Ammar, A., Alajlan, M., & Youssef, H. (2017). Design and performance analysis of global path planning techniques for autonomous mobile robots in grid environments. International Journal of Advanced Robotic Systems. https://doi.org/10.1177/1729881416663663

[20] Raja, P & Pugazhenthi, S. (2012). Optimal path planning of mobile robots: A review. International Journal of the Physical Sciences. 7. 10.5897/IJPS11.1745.

[21] Koubaa A. et al. (2018) Background on Artificial Intelligence Algorithms for Global Path Planning. In: Robot Path Planning and Cooperation. Studies in Computational Intelligence, vol 772. Springer, Cham. https://doi.org/10.1007/978-3-319-77042-0_2.

[22] R. Lagisetty, N. K. Philip, R. Padhi, and M. S. Bhat, ``Object detection and obstacle avoidance for mobile

robot using stereo camera," in *Proc. IEEE Int. Conf. Control Appl. (CCA)*, Aug. 2013, pp. 605_610.

[23] P. Marin-Plaza, A. Hussein, D. Martin, and A. de la Escalera, Global and Local Path Planning Study in a ROS-Based Research Platform for Autonomous Vehicles, Journal of Advanced Transportation, vol. 2018, Article ID 6392697, 10 pages, 2018. https://doi.org/10.1155/2018/6392697.

[24] Kim, B. & Pineau, J. Int J of Soc Robotics (2016) 8: 51. https://doi.org/10.1007/s12369-015-0310-2

[25] Gurel, Canberk Suat & Rajendran Sathyam, Rajendra Mayavan & Guha, Akash. (2018). ROS-based Path Planning for Turtlebot Robot using Rapidly Exploring Random Trees (RRT*).