IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 9 September 2022, accepted 12 October 2022, date of publication 25 October 2022, date of current version 1 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3216601

RESEARCH ARTICLE

Energy-Efficient Local Path Planning of a Self-Guided Vehicle by Considering the Load Position

MOHAMMAD MOHAMMADPOUR[1], SOUSSO KELOUWANI[1], (Senior Member, IEEE), MARC-ANDRÉ GAUDREAU[1], BILEL ALLANI[2], LOTFI ZEGHMI[2], ALI AMAMOU[2], (Member, IEEE), AND MASSINISSA GRABA[2]

[1]Department of Mechanical Engineering, University of Quebec at Trois-Rivières, Trois-Rivières, QC G8Z 4M3, Canada
[2]Department of Electrical and Computer Engineering, University of Quebec at Trois-Rivières, Trois-Rivières, QC G8Z 4M3, Canada

Corresponding author: Mohammad Mohammadpour (mohammad.mohammadpour@uqtr.ca)

This work was supported in part by the Industrial Research Chair Noovelia, and in part by the Natural Science and Engineering Research Council of Canada.

ABSTRACT The local path planning, as one of the navigation stages, plays a significant role in the energy consumption of Self-Guided Vehicles (SGV). Since SGV must operate for several hours on a single battery charge to transport loads, its energy consumption is a critical issue. Therefore, this article puts forward an approach for boosting the energy efficiency of the local path planning stage using load position. Unlike other similar works which solely use robots' kinematic and kinetic constraints to develop energy-efficient local path planners, this article considers the effect of load position on SGV's dynamic. In this regard, first, the kinetic model of the differential drive SGV is developed to consider the change of SGV's Center of Mass (CoM) affected by load properties. Second, machine learning methods are used to create two learning models for online estimation of the position of CoM (PoCoM) and prediction of required energy of sample trajectories. Hence, the generated SGV's kinetic model is used to train the learning models. Finally, estimated parameters are employed to add a new constraint to extend the cost function of the local path planner. The outcomes of the study show that the proposed planner generates smoother and shorter paths to pass obstacles and corridors than a general one. Thus, SGV's energy consumption decreases by considering the load effect.

INDEX TERMS Energy efficiency, local path planning, dynamic, machine learning, self-guided vehicle.

I. INTRODUCTION

The fourth Industrial Revolution 4.0 is a promising approach based on the integration of the business and manufacturing processes, as well as the integration of all actors in the company's value chain [1]. Smart factory as a key construct of industry 4.0 aims to reach more flexible and modular production by using automation technologies which among them Self-Guided Vehicles (SGVs) take a significant role. Compared to the previous generation of Automated Guided Vehicles (AGVs) which were limited to fixed routes, SGVs can navigate dynamically and travel quickly. They are used mostly to perform repetitive heavy material delivery tasks [2]. Hence, they are being more and more used in smart factories to increase manufacturing efficiency and keep most performance indexes as high as possible. Since most SGVs rely on batteries as only sources of energy and use electric tractions to move, their energy consumption is an essential issue to improve productivity [3], [4]. The rapidly growing demand for electric vehicles is putting huge pressure on the increase of batteries price. A possible way to contribute to reduce this pressure is to improve the electric SGV energy efficiency and the battery lifetime. It is well known that lithium-ion battery lifetime is sensitive to the depth-of-discharge (DoD) [5]. Since most electric SGVs are designed to sustain a full shift work, any improvement of energy efficiency during motion will directly reduce the DoD and contribute to augment the battery lifetime. Therefore, it is vital to amend

The associate editor coordinating the review of this manuscript and approving it for publication was Jad Nasreddine.

the energy efficiency of the whole navigation stack of these vehicles.

SGV's navigation module includes five main stages. The process starts from the perception stage by determining the essential information about the robot's surroundings. Then, the mapping stage uses the collected information to model the environment. According to generated map, the localization stage determines the vehicle's location. The fourth one is the planning stage which is divided into two categories of global path planning and local path planning to define proper paths and trajectories. Consequently, the motion control stage executes appropriate torque or current to follow the generated trajectory [6].

Because both planning and motion control stages can affect energy consumption, previous works tried to augment them. In the global path planning part, A$*$ method [7] as one of the famous global planners has been improved in various ways such as using an energy-related cost function [8] or optimization algorithms [9]. In the local one, energy constraints were added to cost functions [10] or multi-sensors were used to generate optimal trajectories [11]. Using optimal control theory to achieve optimal velocity trajectory [12] and adding a model predictive control for trajectory tracking [13], [14] have been famous ways in the motion control stage. However, none of the previous works considered the effect of load position on a robot's energy consumption. Since small robots do not have big loading surfaces, it seems like a good assumption to take dynamic inertial parameters as constant during missions. But for SGV, which can have a big loading surface to transport heavy loads, the inertial parameters are of interest as they will be significantly different for each given mission.

Among SGV navigation stages, local path planning has the main role to define the SGV motion. This stage generates proper trajectories to follow the created path by the global path planner stage. Moreover, the local planner must guide SGV to avoid unexpected obstacles and return it to the defined path. So, any non-optimal trajectory can raise the SGV energy consumption. Because the main task of SGV is the load transportation, its energy consumption increases with the growth of load weight. Moreover, the load effect on SGV energy consumption will intensify if SGV receives angular velocity commands to perform a rotational motion. When loads are placed on the SGV platform, its dynamic inertial parameters such as mass, the moment of inertia, and Center of Mass (CoM) are changed. According to load properties, SGV mass distribution gets asymmetric, and the position of CoM (PoCoM) moved toward the CoM of the load itself.

One important aspect that has escaped the attention of researchers in the domain of energy-efficient motion for SGVs and AGVs is the effect of load position on energy consumption during missions. Regarding this point, a proper kinetic model is required. However, in the literature, the displacement of CoM has been considered to move just in the longitudinal direction in kinetic models [15], [16]. But in reality, when there is a load on the vehicle's platform, the CoM shifts on both longitudinal and lateral axes according to load properties. Thus, this paper has two major contributions: (i) developing the differential drive vehicle's dynamic model to consider changes of CoM in two-dimensional space. The created SGV kinetic model is employed to generate a dataset and train the machine learning (ML) models. So, the ML models help to develop a tool that estimates the required unknown dynamic parameters. (ii) Adding this tool to the local path planner algorithm to generate energy-efficient trajectories is the main contribution of this work. According to the load position on the vehicle, the first ML model finds PoCoM of loaded SGV. Then, the second ML model determines the required energy of each sample of trajectories generated by the local planner. Finally, it is added to the cost function of the local planner. Regarding the emphasis of energy on smart factories and industry 4.0, this method can be tremendously helpful.

The remainder of this paper is organized as follows. The necessity of considering load position in the local path planner is investigated in the next section. In the third section, the SGV kinetic model is developed to demonstrate the changes of PoCoM. According to the generated model, two different machine learning algorithms are employed to estimate unknown parameters and regarding results, the best one is selected in the fourth section. The fifth part describes the energy-efficient local path planner with various performed simulations and experiments to represent the efficiency of the proposed method. The sixth part discusses the results and limitations of this work and suggests some ideas for future works followed by a conclusion.

II. NECESSITY OF CONSIDERING LOAD POSITION

A. EFFECT OF A LOAD ON SGV'S COM

Figure 1 displays the SGV's platform with the displacement of the center of mass (CoM). Where subscripts i, g, l mean inertia, global, and load respectively. Regarding lateral and longitudinal axes, it is assumed that the platform can be divided into four zones. When a load is placed on the platform with the determined center of mass (CoM_l), CoM_i that is in zone B can move to a new global point (CoM_g) such as zone A. So, CoM_g is the new CoM of the platform that affects the SGV motion. The formulation is described as [17]:

$$r_{x_g} = \frac{r_{x_i} m_i + r_{x_l} m_l}{m_g} \tag{1}$$

$$r_{y_g} = \frac{r_{y_i} m_i + r_{y_l} m_l}{m_g} \tag{2}$$

where r_x and r_y are distances of CoM with respect to the middle of the wheels' axis (point P) in x and y directions respectively. In addition, m_g displays the sum of SGV mass (m_i) and load mass (m_l). By assuming the model as a two-dimensional (2D) problem, the vehicle rotation occurs around the z direction (body frame). So, the SGV's moment of inertia around this direction is:

$$I_{z_g} = I_{x_g} + I_{y_g} \tag{3}$$

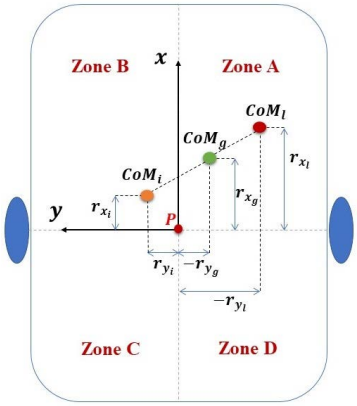


FIGURE 1. SGV's CoM_g with respect to load's CoM_l.

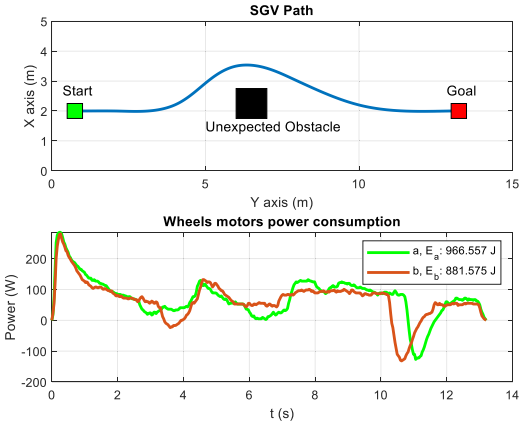


FIGURE 2. The first scenario of the experiment, avoiding an unexpected obstacle.

where I displays the moment of inertia and this parameter around x and y directions are defined as:

$$I_{x_g} = \left(I_{x_i} + m_i I_{x_i}^2\right) + \left(I_{x_l} + m_l I_{x_l}^2\right) \tag{4}$$

$$I_{y_g} = \left(I_{y_i} + m_i I_{y_i}^2\right) + \left(I_{y_l} + m_l I_{y_l}^2\right) \tag{5}$$

According to Figure 1, the load has a substantial effect on the SGV dynamic that should be considered in navigation algorithms. However, previous works have studied the SGV's dynamic by ignoring the parameter r_y ($r_y = 0$). Hence, this paper intends to develop the SGV dynamic model by considering the displacement of CoM_g in both longitudinal (x) and lateral (y) directions.

B. EFFECT OF CoM DISPLACEMENT ON SGV'S ENERGY CONSUMPTION

Rotational motions are frequent for SGVs during missions such as turning in corridors or avoiding obstacles. During a mission, unexpected obstacles might appear to cross the SGVs' defined paths such as humans and other vehicles. An increase in the number of obstacles has a direct effect on SGV energy consumption [11]. Moreover, a nonoptimal reaction to avoid obstacles might raise the distance to the destination and waste time by generating useless motion [18].

To show the effect of load position on energy consumption caused by the local planning stage, some experiments have been performed. The SGV and load's properties are explained in Appendix C. In this regard, experiments include two scenarios. First, SGV moves in a straight line. Second, it turns into a corridor. Moreover, each scenario has two attempts with respect to the middle longitudinal axis of SGV:

1. attempt a: load is placed on the right side of the platform ($r_{x_l} = 0.3$ m, $r_{y_l} = -0.15$ m)
2. attempt b: load is placed on the left side of the platform ($r_{x_l} = 0.3$ m, $r_{y_l} = +0.15$ m)

Three containers are on each other as load (total mass = 55.5 kg). To have an objective comparison of the energy consumption for each mission, the positions of the start and goal, the global path, and the position of the unexpected obstacle are the same for the two load positions.

Figure 2 displays the SGV's path and power consumption of wheels'motors in the first scenario of experimental tests. SGV receives a goal and moves toward it in a straight line. However, there is an unexpected static obstacle to cross over the SGV path. So, the local path planner changes the SGV path to pass the obstacle (turning left) and then returns SGV on the defined path created by the global path planner (straight line). When the load is on the right side of SGV (attempt a), it consumes 966.5 J energy while the consumption decreases in attempt b by 8.8% at the same time.

Figure 3 illustrates the SGV's path and power consumption in the second scenario of the experimental tests. SGV has a rotation motion to the left to pass the corridor. The energy consumption of SGV with a load on the right side of the platform is 514.2 J which is 15.2% more than attempt b. The results of experiments represent considerable differences in energy consumption between two attempts of each scenario under the same conditions. The differences can be repeated when the load's position is changed on the platform according to the middle lateral axis of SGV. So, the local path planner can be improved to find energy-efficient trajectories by using the position of load. For instance, SGV could pass the obstacle or turn in a corridor by using different trajectories or maneuvers.

III. MOTION EQUATIONS OF SGV

Motion equations of the differential drive SGV are developed in Appendix A. Although the kinematic model is like previous works, the kinetic model is developed to consider changes of CoM on both lateral and longitude axes. The kinetic model of SGV is achieved as:

$$\begin{bmatrix} m_g & -m_g r_{y_g} \\ -m_g r_{y_g} & I_{z_g} + m_g(r_{x_g}^2 + r_{y_g}^2) \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} 0 & -m_g r_{x_g} w \\ m_g r_{x_g} w & 0 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} = \frac{1}{r} \begin{bmatrix} 1 & 1 \\ b & -b \end{bmatrix} \begin{bmatrix} \tau_r \\ \tau_l \end{bmatrix} \tag{6}$$

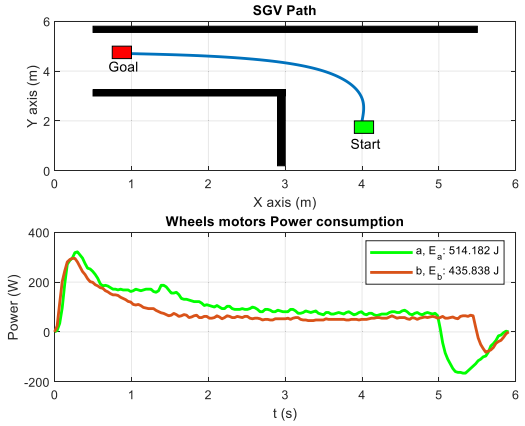


FIGURE 3. The second scenario of the experiment, turning motion in a corridor.

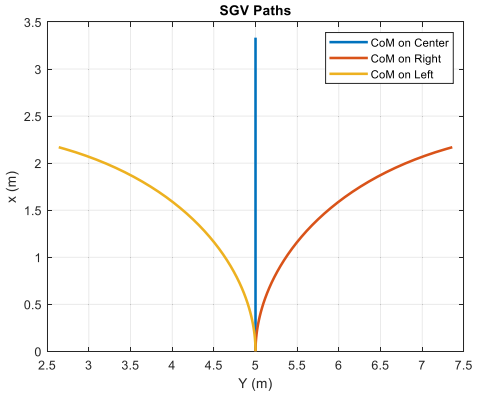


FIGURE 4. SGV paths with the open-loop controller in three scenarios.

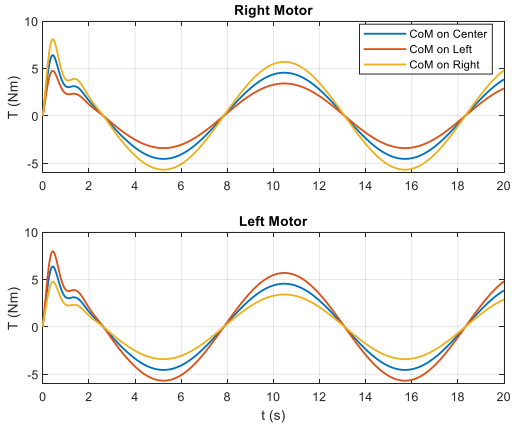


FIGURE 5. Generated torques for wheels motors in three scenarios.

where v and w represent the linear and angular velocities of SGV. Also, b and r define the distance of SGV's wheels to point P, and the radius of them respectively. Moreover, τ_r and τ_l are right and left wheels' motors torques. By determining the values and signs of r_{y_g} and r_{x_g}, the new CoM_g is founded.

An open-loop controller can be used to display the effect of CoM on the SGV's dynamic. So, it is considered that there is a load on the SGV. The load causes changes in the position of CoM_g according to the load's inertial parameters. Therefore, three scenarios are defined:

1. CoM_g on the middle longitudinal axis of SGV ($r_{x_g} = 0.1$ m, $r_{y_g} = 0$ m)
2. CoM_g on the left side of the middle longitudinal axis ($r_{x_g} = 0.\ 1$m, $r_{y_g} = 0.1$ m)
3. CoM_g on the right side of the middle longitudinal axis ($r_{x_g} = 0.1\ m$, $r_{y_g} = -0.1$ m)

Other SGV's parameters by considering load are $m_g = 150$ kg, $I_{z_g} = 6$ kgm^2, $b = 0.4$ m, $r = 0.1$ m. Moreover, left and right motors torques are considered the same as the control inputs for motion in a straight line. Noise is not considered in the simulations to demonstrate this effect clearly.

The simulations' results of three defined scenarios are displayed in Figure 4. They illustrate the effect of CoM on the SGV dynamic for the 3 positions of the load. When torques of the right and left wheels are the same, a straight trajectory is expected. So, while the CoM is on the longitudinal axis, SGV moves along a straight line (blue line). However, when the CoM is on the left and right sides (zones A and B), SGV tends to left (yellow line) and right (red line) sides respectively.

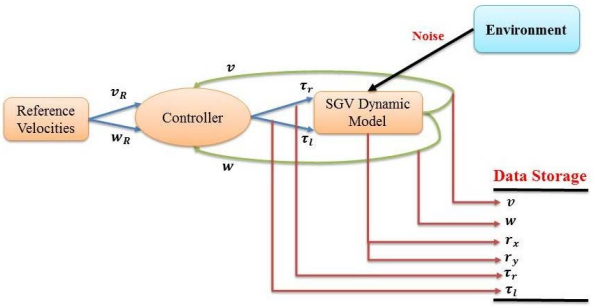
To see the effect of changes of CoM on required torques of right and left wheels motors, three defined scenarios in the previous simulations are repeated. So, linear and angular velocities of references are considered as a wave and zero respectively for a straight motion without any rotation. The controller must generate proper torques to keep the SGV in a straight line. Because SGV's dynamic inertial parameters are highly variable, a sliding-adaptive controller is required for this type of vehicle. So, the sliding-adaptive controller for differential drive vehicles designed by Mevo et al. [19] is employed. To know more information and details about the controller see the mentioned reference. Figure 5 displays the results of the simulations. Hence, by considering CoM on the SGV's longitudinal axis, torques of left and right motors are the same (blue graphs). When CoM is on the right side of the SGV's longitudinal axis, the right motor needs more torque to keep the SGV's motion in a straight line (yellow graph). However, when CoM is on the left side, the left motor consumes more torque (red graph). The results prove the effect of PoCoM on SGV's motion. Hence, it is possible to improve the SGV maneuver by considering PoCoM.

IV. PARAMETERS ESTIMATION

According to the methodology of this work, online estimation of the PoCoM (r_x, r_y) and torques (τ_r, τ_l) are required. Hence, parameter estimation methods can be employed. They are divided into three main groups [20]. The first one is the filter-based method. The most frequently used filters in the literature are the extended Kalman filter (EKF) and unscented Kalman filter (UKF) [21], [22]. The second group is the observer-based method such as recursive least squares [23], linear and nonlinear observers, and sliding mode observers [24]. The third one is the data-driven-based estimation [25]

which uses the generated data from models or collected data from experiments to create a learning model.

Regarding SGV nonlinear dynamic model (6), there are four unknown parameters $(I_{z_g}, m_g, r_{x_g}, r_{y_g})$ due to load effects. The second power of CoM's parameters (r_{x_g}, r_{y_g}) convert them into smaller values. Moreover, they are multiplied by the mass which is a large value in this type of vehicle. So, the mentioned nonlinearity conditions represent a serious challenge for the parameter identification task. Since filter-based and observer-based groups use the model directly to estimate parameters, they cannot be proper ways to address this problem. However, the data-driven-based estimation group can be an appropriate way because of its ability to address nonlinear conditions, hence; a method from the data-driven-based estimation group will be used to estimate the unknown parameters of the loaded SGV.



Model number	Feature 1	Feature 2	Feature 3	Feature 4	Output 1	Output 2
1	v	w	τ_r	τ_l	r_x	r_y
2	v	w	r_x	r_y	τ_r	τ_l

FIGURE 6. The data flow map with the dataset structure.

A. DATA-DRIVEN-BASED PARAMETER ESTIMATION

By using the designed SGV kinetic model, a dataset with known inputs and outputs can be generated. So, a supervised machine learning method is required. Moreover, according to the nonlinearity of the kinetic model that has been discussed, the learning model must handle this condition. Furthermore, the selected method must be able to overcome multi-output problems. In this regard, the Multi-Layer Perceptron (MLP) method can be a proper choice to cover all mentioned conditions. MLP can be applied to complex nonlinear problems and is compatible with large input data. It is able to achieve online quick predictions after fine training. To see whether there is a dependency between the data of the dataset concerning time, a time-series method is employed. Moreover, it can be helpful to obtain a comparison between the results of parameters estimation of a time-series method and MLP to select the best method for this project. Among time series methods, long short-term memory (LSTM) is the useful one that comprises MLP's advantages. A brief description of the two mentioned methods and their formulations are explained in Appendix B.

1) DATASET GENERATION

Dataset is an important part of machine learning methods. Training an algorithm by a dataset is the collection of data that is fed into algorithms to create a predictive model. Regarding the aim of this work, a dataset is generated by using the developed dynamic model of SGV (6). Figure 6 illustrates the data flow map of the dataset generation process and the dataset structure. According to this diagram, various scenarios are performed with different inertial parameters (mass, moment of inertia, CoM). Moreover, various reference velocities are considered to comprehend requirement torques for different conditions. Hence, the values of linear and angular velocities (v,w), torques of right and left motors (τ_r, τ_l), PoCoM (r_x, r_y) are saved in the data storage to create the dataset. Because of the existence of noise in an industrial environment, Gaussian noise is added to interfere in the process to consider uncertainties in the dataset. Since two learning models are used in this paper, two structures are generated. The first model includes four features v, w, τ_r and τ_l and two outputs r_x, r_y. However, the features of the second model are v, w, r_x and r_y and the outputs are τ_r, τ_l.

2) SIMULATIONS AND RESULTS

The results of the parameter estimation task by using machine learning methods are displayed in Table 1. ML model 1 tries to estimate r_{x_g} and r_{y_g}; ML model 2 estimates τ_r, τ_l. According to the literature, a proper way to show the accuracy of the estimation task is RMSE (Root-mean-square deviation) [26]. The unit of the RMSE is like the unit of variables. So, RMSE's units are m and Nm respectively here. The range of changes of r_{x_g} and r_{y_g} is between -0.25 m to $+0.25$ m (total is 0.5 m). Also, the range of torques changes is between -20 Nm to $+20$ Nm. Another useful method is SMAPE (symmetric mean absolute percentage error) to measure the accuracy based on the percentage error [26]. The SMAPE range is between 0 to 100% and a lower percentage means lower error. To demonstrate the robustness of ML models when dealing with different levels of noise, three conditions are considered for the dataset. Type 1 includes data with Gaussian noise. Type 2 and 3 include Gaussian noise with more than 10% and 20% Signal-to-noise ratio (SNR) respectively. The numbers of hidden layers and neurons in training are displayed. The presented values are the optimal ones that have been achieved after some trial and error process. Moreover, the names of activation functions and solvers for both methods are presented. Regarding RMSE and SMAPE of results in Table 1, both MLP and LSTM achieve acceptable results to estimate both PoCoM and torques for different dataset types. To comprehend clearly the difference between results, they are presented in Figure 7. Blue and green lines correspond to MLP and LSTM methods. By increasing the SNR of noise from dataset 1 to dataset 3, the values of errors grow gradually. For instance, in Figure 7a, RMSE of estimation of PoCoM in x-direction for dataset type 1 is 0.08 m and 0.085 m by using MLP and LSTM methods respectively.

TABLE 1. Properties and results of parameter estimation task by MLP and LSTM methods.

Method		MLP						LSTM					
Noise type		Data with noise		Data with noise + 10% more SNR		Data with noise + 20% more SNR		Data with noise		Data with noise + 10% more SNR		Data with noise + 20% more SNR	
Numbers of neurons in three hidden layers		[20, 40, 40]						[20, 30, 40]					
Activation function		Hyperbolic tangent						Hyperbolic tangent					
Solver		Adam						Adam					
ML Model 1	Parameter	r_{x_g}	r_{y_g}	r_{x_g}	r_{y_g}	r_{x_g}	r_{y_g}	r_{x_g}	r_{y_g}	r_{x_g}	r_{y_g}	r_{x_g}	r_{y_g}
	RMSE (m)	0.08	0.06	0.083	0.067	0.09	0.079	0.085	0.065	0.089	0.07	0.092	0.075
	SMAPE (%)	17.7	12.9	18.1	13.4	18.9	15.3	18.2	13.54	18.7	13.78	19.2	14.7
ML Model 2	Parameter	τ_r	τ_l	τ_r	τ_l	τ_r	τ_l	τ_r	τ_l	τ_r	τ_l	τ_r	τ_l
	RMSE (Nm)	0.22	0.25	0.31	0.3	0.39	0.35	0.29	0.27	0.35	0.34	0.43	0.41
	SMAPE (%)	9.5	9.9	11.2	11.1	12.5	12.2	11	10.2	11.9	11.5	13.1	12.8

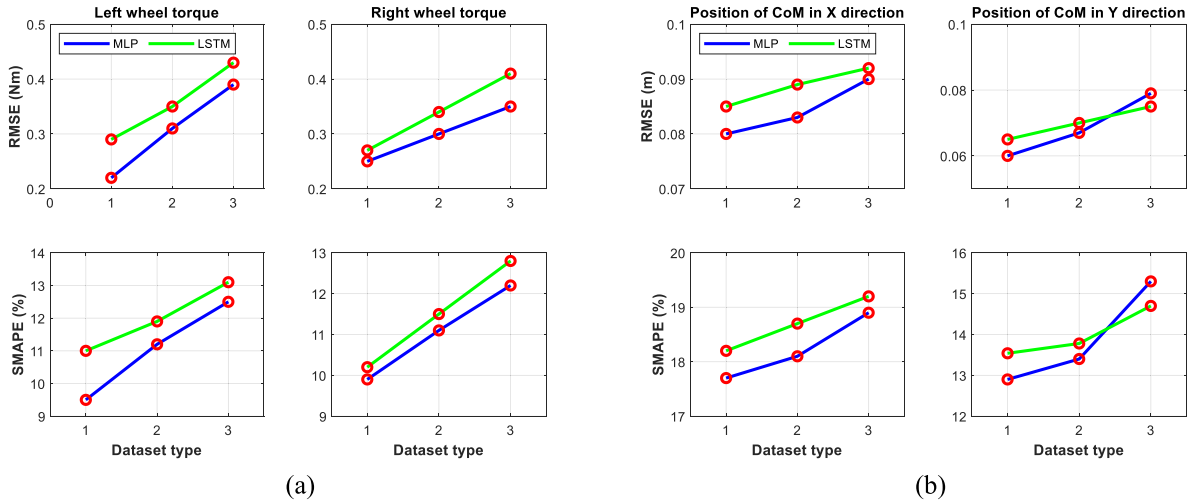


FIGURE 7. The estimation error based on the MLP and LSTM for PoCoM (a) and torques (b). (Blue and green lines indicate MLP and LSTM methods respectively).

When the SNR of noise raises 20% (dataset type 3), RMSE reaches 0.09 m and 0.092 m for MLP and LSTM methods. Also, the same results are observed in SMAPE of estimation. The few changes of RMSE and SMAPE with respect to the increase of noise' SNR prove the robustness of generated ML models to noise when they will be deployed in real-world and uncontrolled conditions.

LSTM needs previous steps of data to predict parameters. However, MLP works with data of the current time. Although the LSTM can be used to estimate PoCoM, it is not proper for LSTM which needs previous steps of data to predict parameters. However, MLP works with data of the current time. Although the LSTM can be used to estimate PoCoM, it is not proper for torque estimation. Estimation of PoCoM is performed at the beginning of the motion once. So, there is enough time (some steps) to collect data and then predict PoCoM. But the estimation of torque must be done according to the frequency rate of the controller which is 20 ms in this work. Regarding these conditions, the MLP method is selected for the parameter estimation task in this article.

V. ENERGY-EFFICIENT LOCAL PATH PLANNING

By using the generated kinetic model and two learning models, the local path planner can be improved to achieve an energy-efficient motion. Hence, Dynamic Window Approach (DWA) as a useful local planner method is employed in this article. Then, some simulations and experiments are performed to investigate the proposed method's efficiency.

A. DYNAMIC WINDOW APPROCH (DWA): AN OVERVIEW

DWA is a well-known collision avoidance navigation algorithm that was proposed by Dieter Fox et al. [27]. Also, DWA is an online reactive method and its cost function has been extended several times during the last years. In order to select safe and optimal translational (v) and rotational (w) velocities, the method directly generates their profiles by considering the

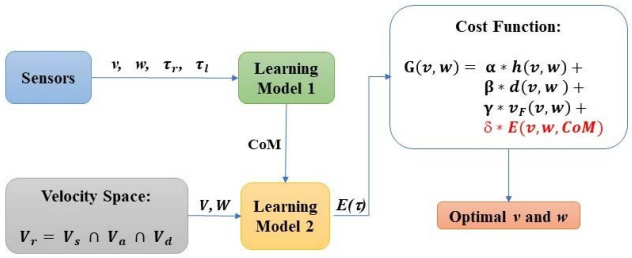


FIGURE 8. The structure of the energy-efficient DWA method.

dynamic of the robot and the range limitation of the velocity and acceleration. The main search space for suitable velocities is intersected by three subspaces: (i) space of possible velocities according to the robot kinematic constraints, V_s, (ii) space of admissible velocities that allows the robot to stop without colliding with an obstacle, V_a, and (iii) space of possible velocities by considering the robot's limited accelerations, V_d:

$$V_r = V_s \cap V_a \cap V_d \tag{7}$$

where V_r is the search space of optimal velocities, which is selected by minimizing the following objective function:

$$G(v, w) = \alpha * h(v, w) + \beta * d(v, w) + \gamma * v_F(v, w) \tag{8}$$

where h measures the alignment of the robot with the target direction, d is the distance to the closest obstacle, and v_F is the robot forward velocity. α, β, and γ are tunable constant weights. Hence, the DWA method generates a lot of possible online trajectories and then selects the optimal ones according to the objective function.

B. ENERGY-EFFICIENT DWA

To develop the energy-efficient DWA by considering PoCoM, the following model is proposed. The method's structure is displayed in Figure 8. When SGV starts moving, ML model 1 receives SGV linear and angular velocities (v, w), left and right motors' torques (τ_l, τ_r) to estimate PoCoM. Then ML model 2 uses PoCoM and vectors of linear and angular velocities (V, W) generated by the velocity space to predict the required energy of each trajectory. Finally, the energy is added to the cost function with a tunable constant weight (δ). Thus, DWA finds the optimal trajectory by minimizing the new cost function. By adding the energy constraint to DWA's cost function, the formulation of the new one is represented in the following formulation:

$$\begin{aligned} G(v, w) &= \alpha * h(v, w) + \beta * d(v, w) \\ &\quad + \gamma * v_F(v, w) + \delta * E(v, w, CoM) \end{aligned} \tag{9}$$

To be sure that the selected optimal velocities are related to the energy-efficient distance traveled to pass an obstacle, the inputs of learning Model 2 are considered as vectors. So, the cost function calculates the cost of vectors of linear and angular velocities which means the cost of possible paths.

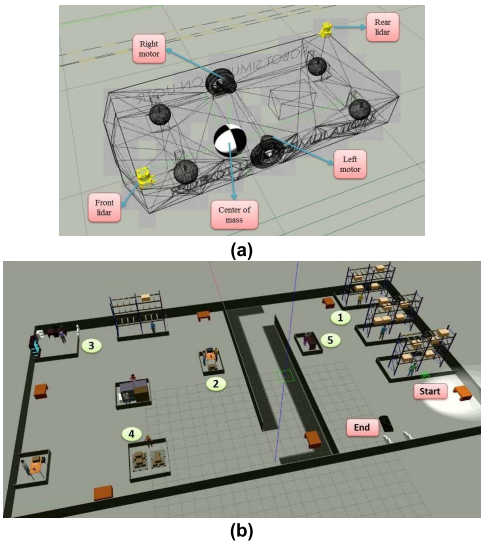


FIGURE 9. Gazebo environment: (a) SGV designed model, (b) simulation's world.

Then, it picks out the optimal vector. This condition can guarantee to achieve an energy-efficient trajectory and distance traveled at the same time. It is worth mentioning that learning models 1 and 2 do not work at the same time. Because the value of PoCoM is constant until the next SGV's mission, it is estimated once during the first steps of the motion. Then, it is used as a constant input of model 2 until the next load. So that, model 2 receives online inputs to generate the output in each step of the controller frequency.

C. SIMULATION

Gazebo is a powerful 3D simulator to calculate physics, generate sensor data, and provide convenient interfaces to make a specific robot [28]. By changing the inertial parameters of a created robot in Gazebo, the robot's dynamic is affected like a real robot. In addition, it is able to simulate a specific environment with all details. So, Gazebo is a proper choice to create the 3D SGV model and the industrial environment with consideration of obstacles. Figure 9a displays the differential drive SGV model in Gazebo with inertial dynamic properties like the real one which represented in Appendix C. It includes two lidars to perceive the environment and two motors to move SGV. Moreover, CoM is displayed. The Robot Operating System (ROS) [29] has been used as the navigation stack role. ROS is a software development kit that helps the robot to create robot applications such as drivers, algorithms, and node creation and destruction for various operations. Hence, cooperation with ROS and Gazebo is used to create an ideal environment to perform simulations.

Figure 9b displays the simulation environment in Gazebo. SGV begins moving from the start point and then stops in five stations (1-5) respectively followed by the end point. The load's inertia parameters are changed in each station. However, the geometry of the load is rectangular for the whole scenario. There are some unexpected static obstacles to

TABLE 2. Results of simulations of SGV motion by considering load changes and effect of the energy-efficient local planner.

S	Start point	End point	Total mass (kg)	CoM_l (cm)		Real CoM_g (cm)		Estimated CoM_g (cm)		Simulation time (s)		Distance traveled (m)		Energy consumption (J)		Energy efficiency (%)
				r_{x_l}	r_{y_l}	r_{x_g}	r_{y_g}	r_{x_g}	r_{y_g}	*E0*	*E*	*E0*	*E*	*E0*	*E*	
a	Start	1	150	44	22	20	10	17.4	8.7	23.0	20.8	15.5	15.1	1521	1363	10
b	1	2	125	29	-15	10	-5	12.5	-3.9	43.8	40.2	25.5	24.1	2689	2244	16
c	2	3	165	-20	-20	-10	-10	-11.6	-11.1	24.7	25.1	16.8	16.5	1310	1260	4
d	3	4	133	-15.5	21	-6	8	-5.9	9.2	22.4	21.2	16.3	15.8	1405	1315	6
e	4	5	145	21	16	9	7	8.5	6.6	46.5	43.4	24.7	23.3	2770	2366	14
f	5	End	115	-28	4	-8	4	-9.8	3.4	19.6	18.5	12.5	12.2	1287	1015	11
Totally										177.8	169.2	109.3	105.1	10982	9563	13
															r_x	r_y
										SMAPE of estimation (%)					13	14.2

cross over the SGV's path. SGV must follow the global path generated by the global planner besides avoiding obstacles. The scenario is implemented in two conditions. First, SGV's navigation uses the general DWA ($\delta = 0$) as the local planner algorithm. Second, the proposed DWA method ($\delta \neq 0$) is applied.

According to the direct relation between torque and current, torque is linearly proportional to current [14]:

$$\tau = Ki_M \tag{10}$$

where i_M and k denote current and torque's constant respectively. Motors' torques are found by the motion controller and the required currents of a motor can be obtained by using (10). The power consumption of the motor (P_M) is defined as:

$$P_M = V_M i_M \tag{11}$$

where V_M is the voltage of the motor. So, the total energy consumption of the motor (E_T) can be displayed as:

$$E_T = \int P_M dt \tag{12}$$

The simulation is designed to imitate a real mission in a warehouse. Hence, SGV moves between sections of a warehouse and each time it transports a new load with different mass and position. Figure 10 represents the SGV traveled path. Yellow circles with numbers 1-5 display the stations where SGV must stop, completing each section of the total mission (*a* to *f*). The direction of SGV's movement is illustrated by gray arrows. Green, red and black rectangles show the positions of the start, end, and obstacles respectively. The obstacle in blue is only present for section *e*, it is not present when the SGV is doing section *b*. *E0* implies the situation of using the general DWA ($\delta = 0$) and *E* means when the energy-efficient DWA is applied ($\delta \neq 0$).

Figures 11 displays the difference between velocities and Figure 12 represents the difference between motors' torques in simulations. Subscripts *R* and *L* indicate right and left motors respectively. To have a better comparison between the general DWA and the energy-efficient one, the whole results are presented in Table 2. It includes six sections (*a*-*f*)

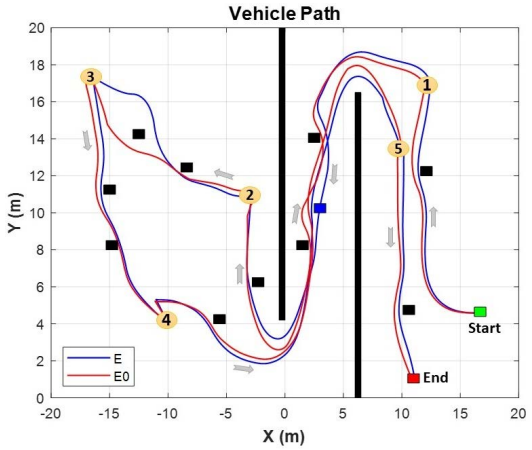


FIGURE 10. SGV path in the simulation (E and E0 are related to energy-efficient DWA and the general one respectively).

that display the simulations' properties and results of SGV movement between two stations. Hence, the station's number and total mass (sum of SGV's mass and load's mass) are presented. Moreover, the simulation time, the traveled distance, and SGV's energy consumption between two stations for both situations of the simulation's scenario are displayed. Loads and real global CoM (CoM_l, real CoM_g) are illustrated with the outputs of parameter estimation (estimated CoM_l). The results are represented by SMAPE which are 13% and 14.2% for estimations of r_{x_g} and r_{y_g}. The column related to energy efficiency shows the percentage of reduction of energy consumption by using the energy-efficient local planner. So, the positive value of this column demonstrates the advantage of the proposed method than the general one. The total results are displayed in green color. Hence, SGV terminates the scenario in 169.2 s by traveling 105.1 m and consuming 9563 J energy while using the energy-efficient DWA. However, SGV consumes 10982 J energy to travel 109.3 m in 177.9 s by applying the general DWA. Hence, in addition to the reduction in time and distance traveled of SGV, the energy consumption decreases by 13% while activating the proposed method.

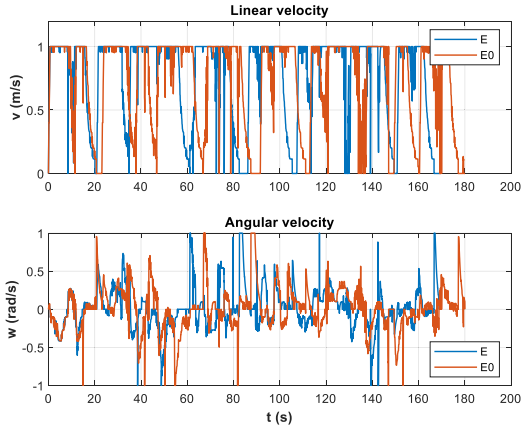


FIGURE 11. Linear and angular velocities of SGV in the simulation.

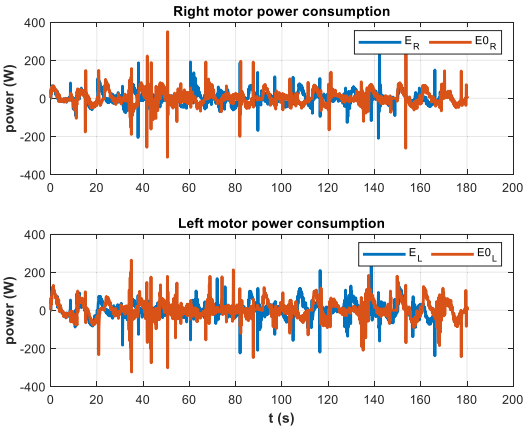


FIGURE 12. Power consumption of SGV's left and right motors in the simulation.

D. EXPERIMENTAL VALIDATION

In order to experimentally validate the simulations in the previous part, several experiments are carried out by using an industrial SGV. Five storage containers are used as loads. The experimental tests are performed in the lab of the Hydrogen Research Institute (IRH) of the University of Quebec at Trois-Rivières (UQTR). The floor is painted concrete leveled to avoid slip. The properties of SGV, loads, and test environment are presented in Appendix C. Regarding the space limitation of the lab, the environment of simulations and experiments are different. However, their logics of scenarios are the same where load's mass and position are changed during missions and SGV must pass corridors and unexpected obstacles. Although simulations aim to see the output of the proposed method in a more complicated world, experiments intend to find this effect on long-time missions. SGV starts from the station, passes waypoints 1 and 2, then returns to the station. Two unexpected obstacles are considered to cross over SGV's path. The first obstacle appears when SGV departs from the station into waypoint 1 and the second one becomes visible when SGV moves from waypoint 2 to the station. The effect of optimal or nonoptimal motions on SGV's energy consumption intensifies by the maximum load. Thus, there are four or five loads in each attempt. Moreover, the loads' weight and positions are constant during each attempt. When SGV returns to the station at the end of an attempt, their weight and positions are changed for the next attempt. In total, six attempts are performed with the same scenario and different loads' mass and position for both energy-efficient DWA and the general one. As shown earlier (see Figure 1), the loading surface can be divided into 4 zones with respect to point P and x and y axes. Attempts 1 to 4 present a different load in each of the 4 zones. In attempt 5, the global CoM is considered close to point P. Also, attempt 6 is created to see the result compared to attempt 2 when their CoM_g is the same however loads' heights are different. Hence, the experiments cover all of the possible conditions that might happen for SGV. Because SGV performs long-time scenarios in the industry,

TABLE 3. Results of experimental tests of SGV motion by considering load changes and effect of the energy-efficient local planner.

At	TM (kg)	NoL	CoM_l (cm)		$RCoM_g$ (cm)		$ECoM_g$ (cm)		ET (min)		DT (m)		DoD (%)		EC (KJ)		EE (%)
			r_{x_l}	r_{y_l}	r_{x_g}	r_{y_g}	r_{x_g}	r_{y_g}	$E0$	E	$E0$	E	$E0$	E	$E0$	E	
1	174.5	2	5.5	12	10.3	5.5	9.1	5.2	75	67.8	1485	1477	15.2	12.8	299	259	13.4
		3	34.5	12													
2	156	4	30.5	-11.5	12.5	-6.5	14.9	-6.9	76.4	73.3	1488	1481	14.4	13.0	275	253	8.3
3	156	4	-27	11.5	-15	4.5	-13.4	3.8	75.9	71.3	1489	1485	14.2	12.7	321	270	11.1
4	174.5	5	-15	-10	-9.8	-6.3	-18	-10.5	69.7	71.2	1481	1483	14.6	14.7	271	275	-1.6
5	174.5	5	5	4	0.8	1.2	1.2	2	72.9	68.8	1482	1479	14.8	13.6	274	255	7
6	156	2	12	-11	12.6	-6.3	14.8	-7.2	76.2	71.5	1486	1480	14.4	12.9	268	240	10.5
		2	50	-11													
Totally									446	424	8911	8889	14.6	13.3	1708	1550	9.25

		r_x	r_y
SMAPE of estimation (%)	With scenario 4	26	23.6
	Without scenario 4	19.5	18.3

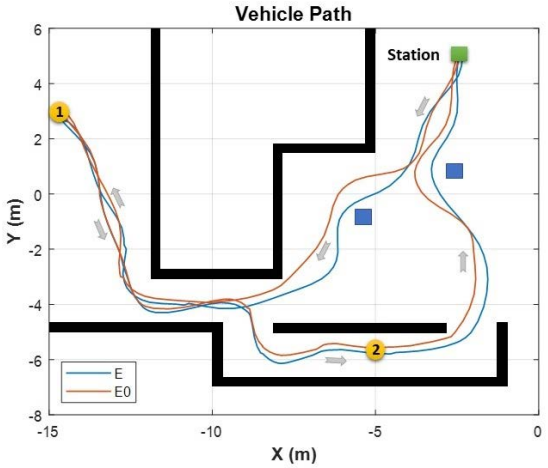


FIGURE 13. SGV path in the experiment (attempt 1, first iteration).

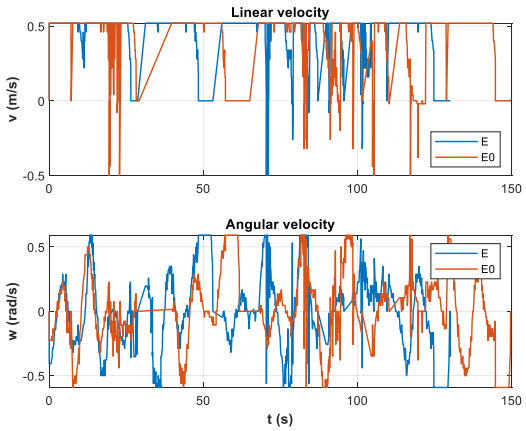


FIGURE 14. Linear and angular velocities of SGV in the experiment (attempt 1, first iteration).

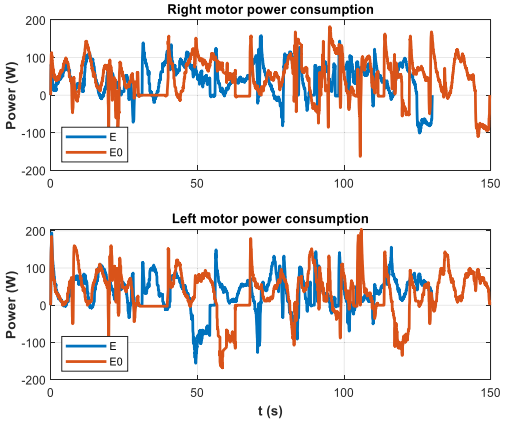


FIGURE 15. Power consumption of SGV's left and right motors in the experiment (attempt 1, first iteration).

so each attempt is repeated 28 times (nonstop) here to be guaranteed more than an hour of motion. This condition helps to consider uncertainties of localization and mapping that might raise the time of missions. Hence, they can affect the efficiency of motion planning. Moreover, the percentage of the energy efficiency due to more than an hour of motion is more applicable for the comparison and is more similar to a real mission than just a few minutes scenario.

Because of long-time tests and the high number of repetitions, the results of the first iteration of scenario 1 are just displayed in Figures 13-15 where red and blue colors mean general DWA ($E0$) and energy-efficient DWA (E) respectively. Hence, SGV's path is represented in Figure 13 where the green rectangular shows the position of the station, and the blue ones are related to the positions of unexpected obstacles. In addition, two yellow circles display the positions of waypoints 1 and 2. The direction of SGV's movement is illustrated by gray arrows. According to the generated paths, the one generated by the proposed method (blue) is more smother than the general DWA (red). When SGV passes an obstacle, the energy-efficient DWA tries to generate optimal trajectories to create a shorter distance than the obstacle to pass it. Figure 14 demonstrates linear and angular velocities of SGV which are different by using the proposed method and Figure 15 displays the power consumption of left and right motors.

Table 3 represents the details and results of experimental tests. It includes the number of loads (NoL) and the total mass (TM) which is the sum of SGV and loads mass in each attempt (At). The column related to CoM_l explains the distance of loads' CoM with respect to point P. The real CoM_g is calculated by using (1) and (2). Also, the estimated CoM_g is achieved by using the generated learning model 1 which is explained in the previous section. Moreover, the experiments' time (ET), distance traveled (DT), DoD, and motors energy consumption (EC) columns display the sum of 28 times of repetitions in each attempt. The difference between motors' energy consumption due to the two used methods are illustrated as the energy efficiency (EE).

The estimations' results are acceptable except in scenario 4 where the error is more than 8 cm in the x-direction and 4 cm in the y-direction. The SMAPE of the whole estimations of r_{x_g} and r_{y_g} are 26% and 23.6%. However, they become 19.5% and 18.3% without considering scenario 4. The simulation time, SGV distance traveled, and motors' energy consumption of proposed methods are less than the general DWA in all scenarios except scenario 4. Thus, by achieving an incorrect estimation, the proposed method cannot be energy-efficient and even makes SGV consumes more energy. SGV travels 8911 m in 446 min by consuming 1708 KJ energy while using the general local planner. However, it travels 8889 m in 424 min with 1550 KJ energy consumption and by using the energy-efficient one. Moreover, the average of the battery's DoD becomes 13.3% which is 9% less than the general DWA method with 14.6% DoD. In the end, the proposed method helps SGV to achieve 9.25% energy efficiency during experiments and finish the missions almost 22 min sooner.

VI. DISCUSSION AND FUTURE WORKS

Since energy-efficient motion is a vital subject to raise the robot's operation without charging, previous studies have

proposed different ways to improve it. However, few of them have surveyed the local planning part. The energy-efficient DWA was proposed by Xie et al. [10]. They added the robot's energy consumption model to DWA's cost function and achieved different percentages of the reduction in the energy consumption by changing the constant gains. In more recent work on this topic, Szczepanski et al. [30] suggested an energy-efficient local path planning based on predictive Artificial Potential Field (APF). Also, their method could reduce the energy consumption of a small robot. However, none of these works, as similar works to this paper, considered the effect of load on local planning to achieve an energy-efficient motion. Hence, the changes in mass, the moment of inertia, and CoM during a mission were ignored. Since the main role of SGV is load transportation, this paper proposed a method to consider the changes of dynamic inertial parameters to achieve an energy-efficient mission.

The previous section surveys the effect of the proposed method on the energy consumption of SGV. The results of simulations and experimental tests display 13% and 9.25% reduction in energy consumption. Moreover, this method helps SGV to complete the mission by consuming less time and distance traveled. However, when there is a high estimation error (experimental tests, attempt 4), this method cannot be useful and even increase energy consumption. So, the high accuracy of the parameter estimation is essential for this work. Concerning this matter, some improvements are suggested for future works. Hence, besides the parameter estimation task, a classification method can be employed to make sure that the estimated sign of r_x and r_y (positive or negative) are correct. A combination of the result of estimation and classification methods guarantees the precision of the final result. Another point that is perceived from the experimental results, is the effect of loads' height on the results. Whereas the positions of real global CoM of loads in attempts 2 and 6 are the same, their energy consumption is different. Four containers are on each other in attempt 2. However, they are divided into 2×2 in attempt 6. The SGV completed the mission in less time and energy consumption in attempt 6, showing the fact that the height of the CoM influences the dynamic properties of the SGV. Since this work develops the SGV kinetic model in 2D, it is not able to consider changes of CoM in the z-direction. So, a 3D kinetic model might be more useful when there are changes in loads' height. When SGV performs rotational motion with a heavy and long load, the effect of the moment of inertia would appears. Although this study tries to find energy-efficient motion by considering PoCoM, the estimated moment of inertia can be employed in future works to raise the energy efficiency of motion.

An interesting topic that can be considered in future work is the geometry of obstacles. The proposed local planner in this paper finds the optimal path to avoid obstacles by achieving the optimal vector of trajectories and considering PoCoM. In other words, it finds the optimal side to pass an obstacle (left or right) by considering PoCoM. It just calculates the required energy according to the front side of the obstacle and assumes the left and right sides of the obstacle are the same. However, sometimes the obstacle has an asymmetric geometry, and the optimal side is unknown to the local planner. In this condition, machine learning algorithms such as classification methods can be employed to predict the type of geometry and help the local planner to find an energy-efficient side to pass obstacles. This study deals with static obstacles which cross over SGV's pass. However, dynamic obstacles might appear in environments. Hence, our future work will discuss the proposed method by considering dynamic obstacles.

Although SGV performs a mission for more than an hour in this study, it must work much more time in industrial surroundings and warehouses on a single battery charge. Hence, the proposed method can decrease the total energy consumption of SGV on a workday.

VII. CONCLUSION

The current study proposed a new methodology to make energy-efficient local path planning for SGVs' navigation. First, this work used the newton-Euler method to develop a SGV kinetic model that takes into account the position and the weight of the added load. Hence, it included displacement of CoM on both longitudinal and lateral axes. Second, the created kinetic model was employed to generate a dataset with known inputs and outputs. The dataset was used to train two learning models which estimated CoM and required torque respectively. Regarding the nonlinearity of the kinetic model and the supervised dataset, MLP and LSTM methods were tried for the parameter estimation task. However, MLP was selected as the main parameters estimator with respect to the problem's condition and results of estimation. Finally, the two trained models were added to DWA as the local path planner of SGV navigation. DWA could generate trajectories by considering CoM and the required energy of sample trajectories. The simulations and experiment results proved the energy efficiency of the proposed method in comparison with the general DWA. This method can be used as a tool for all created local path planner algorithms in real-time missions. It will help them to consider the load position on SGVs to choose energy-efficient motion during turning in corridors or avoiding obstacles.

APPENDIX A SGV'S MOTION EQUATIONS (PROOF OF EQUATION 6)

The motion equations of differential drive SGV are achieved in the following.

A. KINEMATIC MOTION EQUATIONS

According to Figure 16, it is assumed that there is a load on SGV. Therefore, SGV's center of mass (point G) moves to a new position. The position of point G is illustrated by r_x and r_y which are the distance between point G to the middle of the wheels' axis (point P) in x and y directions respectively. Moreover, OXY and Pxy are reference coordinates (fixed) and body coordinates (moving) respectively.

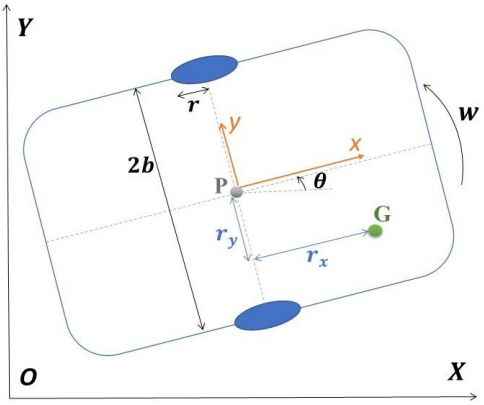


FIGURE 16. SGV free body diagram.

By defining v as the linear velocity in x direction and ω as the angular velocity of the vehicle around z direction the kinematic model is described:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{1}$$

By describing the variable generalized vector q, the Jacobian matrix s, and u:

$$q = \begin{bmatrix} X \\ Y \\ \theta \end{bmatrix}; \quad S = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix}; \; u = \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{2a, b, c}$$

Equation (1) can be described as:

$$\dot{q} = Su \tag{3}$$

The linear velocity of the vehicle in the inertial coordinate is defined:

$$\vec{v}_P = \dot{X}\hat{I} + \dot{Y}\hat{J} \tag{4}$$

where $\hat{I}$ and $\hat{J}$ are unit vectors of inertial coordinate. Moreover, $\hat{i}$ and $\hat{j}$ are unit vectors of body coordinate. By defining the unit vector of body coordinate in the inertial coordinate:

$$\hat{j} = \cos\theta\hat{J} - \sin\theta\hat{I} \tag{5}$$

Vector projection of $\vec{v}_P$ on the direction $\hat{j}$ is:

$$\vec{v}_P \cdot \hat{j} = 0 \tag{6}$$

By substituting (4) and (5) in (6):

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ 0 \end{bmatrix} \cdot \begin{bmatrix} -\sin\theta \\ \cos\theta \\ 0 \end{bmatrix} = 0 \tag{7}$$

So, velocity constraint which is a type of nonholonomic motion family is defined in the inertial coordinate:

$$-\sin\theta\dot{X} + \cos\theta\dot{Y} = 0 \tag{8}$$

Equation (8) can be rewritten as a closed form:

$$A\dot{q} = 0 \tag{9}$$

where A is the constraint matrix:

$$A = [\,-\sin\theta \quad \cos\theta \quad 0\,] \tag{10}$$

$$\dot{q} = [\,\dot{X} \quad \dot{Y} \quad \dot{\theta}\,]^T \tag{11}$$

The relation between wheels' angular velocities and vehicle's velocities according to point P can be described:

$$v = \frac{r}{2}(\dot{\varphi}_r + \dot{\varphi}_l) \tag{12}$$

$$\omega = \frac{r}{2b}(\dot{\varphi}_r - \dot{\varphi}_l) \tag{13}$$

where $\dot{\varphi}_r$ and $\dot{\varphi}_l$ are angular velocities of right and left wheels, b is the distance of each wheel to point P, and r is the radius of each wheel. Equations (12) and (13) can be written according to right and left wheels' angular velocities respectively:

$$\dot{\varphi}_r = \frac{1}{r}v + \frac{b}{r}\omega$$

$$\dot{\varphi}_l = \frac{1}{r}v - \frac{b}{r}\omega \tag{14a, b}$$

B. KINETIC MOTION EQUATIONS

The principle of virtual work should be used to find generalized forces. So, by defining the virtual movement in the body coordinate:

$$\delta\varphi_r = \frac{1}{r}\delta x + \frac{b}{r}\delta\theta$$

$$\delta\varphi_l = \frac{1}{r}\delta x - \frac{b}{r}\delta\theta \tag{15a, b}$$

The virtual work of external forces and moments acting on the vehicle can be calculated:

$$f_x\delta x + f_y\delta y + \tau_\theta\delta\theta = \tau_r\delta\varphi_r + \tau_l\delta\varphi_l + \lambda\delta y \tag{16}$$

where f_x, f_y and τ_θ are generalized forces, τ_r and τ_l are torques of right and left wheels' motors, and λ is the constraint force. Substituting (15) in (16):

$$f_x\delta x + f_y\delta y + \tau_\theta\delta\theta = \frac{1}{r}(\tau_r + \tau_l)\,\delta x + \frac{1}{r}(\tau_r - \tau_l)\,\delta\theta + \lambda\delta y \tag{17}$$

So, the generalized forces can be calculated from (17):

$$f_x = \frac{1}{r}(\tau_r + \tau_l)$$

$$\tau_\theta = \frac{1}{r}(\tau_r - \tau_l)$$

$$f_y = \lambda \tag{18a, b, c}$$

The constrained force is appeared because of the impossibility of wheels slipping in the y direction. To define the constraint force in the inertial coordinate, it can be described:

$$\lambda\hat{j} = \lambda(-\sin\theta\hat{I} + \cos\theta\hat{J}) \tag{19}$$

Equation (19) can be rewritten in the matrix form:

$$\lambda\hat{j} = \begin{bmatrix} -\sin\theta \\ \cos\theta \\ 0 \end{bmatrix}\lambda = A^T\lambda \tag{20}$$

Moreover, it is clear that:

$$S^T A^T = 0 \tag{21}$$

Equation (21) shows the properties of the Natural Orthogonal Complement. So, f_x and f_y can be defined in X and Y directions:

$$\begin{aligned} f_X &= f_x \cos\theta - \lambda \sin\theta \\ &= \frac{1}{r}(\tau_r + \tau_l)\cos\theta - \lambda \sin\theta \end{aligned} \tag{22}$$

$$\begin{aligned} f_Y &= f_x \sin\theta + \lambda \cos\theta \\ &= \frac{1}{r}(\tau_r + \tau_l)\sin\theta + \lambda \cos\theta \end{aligned} \tag{23}$$

To find the motion equations:

$$\vec{r}_P = X\hat{I} + Y\hat{J} \tag{24}$$

$$\vec{r}_{G/P} = (r_x\hat{i} + r_y\hat{j}) \tag{25}$$

$$\begin{aligned} \vec{r}_G &= \vec{r}_P + \vec{r}_{G/P} \\ &= (X + r_x\cos\theta - r_y\sin\theta)\hat{I} \\ &\quad + (Y + r_x\sin\theta + r_y\cos\theta)\hat{J} \end{aligned} \tag{26}$$

where $\vec{r}_P$ and $\vec{r}_G$ are the position of points P and G in the inertial coordinate, and $\vec{r}_{G/P}$ is the position of point G related to point P. By using (26), linear velocity $\vec{v}_G$ and acceleration $\vec{a}_G$ of CoM are defined:

$$\begin{aligned} \vec{v}_G &= (\dot{X} - r_x\dot{\theta}\sin\theta - r_y\dot{\theta}\cos\theta)\hat{I} \\ &\quad + (\dot{Y} + r_x\dot{\theta}\cos\theta - r_y\dot{\theta}\sin\theta)\hat{J} \end{aligned} \tag{27}$$

$$\begin{aligned} \vec{a}_G &= (\ddot{X} - r_x\ddot{\theta}\sin\theta - r_x\dot{\theta}^2\cos\theta - r_y\ddot{\theta}\cos\theta + r_y\dot{\theta}^2\sin\theta)\hat{I} \\ &\quad + (\ddot{Y} + r_x\ddot{\theta}\cos\theta - r_x\dot{\theta}^2\sin\theta - r_y\ddot{\theta}\sin\theta - r_y\dot{\theta}^2\cos\theta)\hat{J} \end{aligned} \tag{28}$$

To define the translation motion of the vehicle, the Newton method is used:

$$\sum \vec{F} = m\vec{a}_G \tag{29}$$

By dividing (29) into components of X and Y directions:

$$m(a_G)_X = f_X \tag{30}$$

$$m(a_G)_Y = f_Y \tag{31}$$

By substituting (22), (23), and (28) in (30) and (31):

$$\begin{aligned} &m(\ddot{X} - r_x\ddot{\theta}\sin\theta - r_x\dot{\theta}^2\cos\theta - r_y\ddot{\theta}\cos\theta + r_y\dot{\theta}^2\sin\theta) \\ &\quad = \frac{1}{r}(\tau_r + \tau_l)\cos\theta - \lambda\sin\theta \end{aligned} \tag{32}$$

and

$$\begin{aligned} &m(\ddot{Y} + r_x\ddot{\theta}\cos\theta - r_x\dot{\theta}^2\sin\theta - r_y\ddot{\theta}\sin\theta - r_y\dot{\theta}^2\cos\theta) \\ &\quad = \frac{1}{r}(\tau_r + \tau_l)\sin\theta + \lambda\cos\theta \end{aligned} \tag{33}$$

To define the rotational motion of the vehicle, the Euler method is used:

$$\sum \vec{M}_P = I_G\ddot{\theta}\hat{k} + \vec{r}_{G/P} \times m\vec{a}_G \tag{34}$$

By substituting (25) and (28) in (34):

$$\begin{aligned} \sum \vec{M}_P &= I_G\ddot{\theta}\hat{k} + m(r_x\hat{i} + r_y\hat{j}) \\ &\quad \times [(\ddot{X} - r_x\ddot{\theta}\sin\theta - r_x\dot{\theta}^2\cos\theta - r_y\ddot{\theta}\cos\theta \\ &\quad + r_y\dot{\theta}^2\sin\theta)\hat{I} + (\ddot{Y} + r_x\ddot{\theta}\cos\theta - r_x\dot{\theta}^2\sin\theta \\ &\quad - r_y\ddot{\theta}\sin\theta - r_y\dot{\theta}^2\cos\theta)\hat{J}] \end{aligned} \tag{35}$$

By defining $\hat{I}$ and $\hat{J}$ in the body coordination in (35), then simplifying and sorting:

$$\begin{aligned} &(I_G + mr_x^2 + mr_y^2)\ddot{\theta} - m(r_x\sin\theta + r_y\cos\theta)\ddot{X} \\ &\quad + m(r_x\cos\theta - r_y\sin\theta)\ddot{Y} = \frac{b}{r}(\tau_r - \tau_l) \end{aligned} \tag{36}$$

So, the closed-form of the motion equation that is obtained by using the Newton-Euler method:

$$M(q)\ddot{q} + C(q,\dot{q}) = B(q)\tau + A^T\lambda \tag{37}$$

where:

$$M(q) = \begin{bmatrix} m & 0 & M(q)_{13} \\ 0 & m & M(q)_{23} \\ M(q)_{31} & M(q)_{32} & M(q)_{33} \end{bmatrix} \tag{38a}$$

$$M(q)_{13} = -m(r_x\sin\theta + r_y\cos\theta) \tag{38b}$$

$$M(q)_{23} = m(r_x\cos\theta - r_y\sin\theta) \tag{38c}$$

$$M(q)_{31} = -m(r_x\sin\theta + r_y\cos\theta) \tag{38d}$$

$$M(q)_{32} = m(r_x\cos\theta - r_y\sin\theta) \tag{38e}$$

$$M(q)_{33} = I_G + m(r_x^2 + r_y^2) \tag{38f}$$

$$C(q,\dot{q}) = \begin{bmatrix} -m(r_x\cos\theta - r_y\sin\theta)\dot{\theta}^2 \\ -m(r_x\sin\theta + r_y\cos\theta)\dot{\theta}^2 \\ 0 \end{bmatrix} \tag{39}$$

$$B(q) = \begin{bmatrix} \cos\theta & \cos\theta \\ \sin\theta & \sin\theta \\ b & -b \end{bmatrix} \tag{40}$$

$$\tau = \begin{bmatrix} \tau_r \\ \tau_l \end{bmatrix} \tag{41}$$

The constraint force λ can be removes in (37) by using (21). So, the Jacobian matrix S is multiplied in (38):

$$S^T M(q)\ddot{q} + S^T C(q,\dot{q}) = S^T B(q)\tau + S^T A^T\lambda \tag{42}$$

So, λ is eliminated:

$$S^T M(q)\ddot{q} + S^T C(q,\dot{q}) = S^T B(q)\tau \tag{43}$$

By substituting the derivative of (2) in (37) and sorting:

$$(S^T MS)\dot{u} + (S^T M\dot{S})u + S^T C = S^T B\tau \tag{44}$$

where:

$$\dot{u} = \begin{bmatrix} \dot{v} \\ \ddot{\theta} \end{bmatrix} \tag{45}$$

By simplifying and sorting (44), the kinetic motion equation of the differential drive SGV is achieved:

$$\begin{bmatrix} m & -mr_y \\ -mr_y & I_G + m(r_x^2 + r_y^2) \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} 0 & -mr_x w \\ mr_x w & 0 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} = \frac{1}{r} \begin{bmatrix} 1 & 1 \\ b & -b \end{bmatrix} \begin{bmatrix} \tau_r \\ \tau_l \end{bmatrix} \quad (46)$$

APPENDIX B FORMULATIONS OF MLP AND LSTM METHODS

The formulations of MLP and LSTM methods' equations are described in the following.

C. MLP

The MLP is one of the most popular feed-forward artificial neural network (ANN) architectures for supervised multi-estimation and nonlinear problems. The back-propagation (BP) model is the most used among different types of MLP models. MLP includes at least three layers: (i): input layer(s), (ii): hidden layer(s), and (iii): an output layer(s). It works by approximating the non-linear relationship between the input and the output by adjusting the weight values internally. It has two main processes. Firstly, input signals are transformed into output layers by using a nonlinear activation function. Then, the prediction error is back-propagated to tune the weights. The process continues until the prediction error converges to a stable value.

The data transformation and the back-propagation process are expressed respectively in the following equations [31].

$$y_j^m(n) = f\left(\sum_{k=0}^{p} w_{ji}^k(n)\, y_i^k(n)\right) \quad (1)$$

$$w(n+1) = w(n) - \eta \frac{\partial E_p}{\partial w(n)} + \alpha \Delta w(n) \quad (2)$$

where $y_j^m(n)$ shows the output of the mth neuron of the jth layer and $f(x)$ is the activation function. p defines the total number of the ith layer, and $w_{ji}^k(n)$ is the connection weight from the kth node of ith layer at the nth iteration. Moreover, $w(n)$ defines the weights at nth iteration and η is the learning rate. E_p denotes the prediction error, and α is the momentum parameter that can accelerate the convergence.

D. LSTM

LSTM is a sort of Recurrent Neural Network (RNN) in the field of Deep Learning that has been specifically developed for the use of addressing sequential prediction problems. It includes a cell to remember values over arbitrary time intervals and three gates of forget f_g, input i_g and output o_g to regulate the flow of ∈information into and out of the cell. To solve the problem of disappearing gradients, LSTM structure is a beneficial solution. A detailed description can be found in [32].

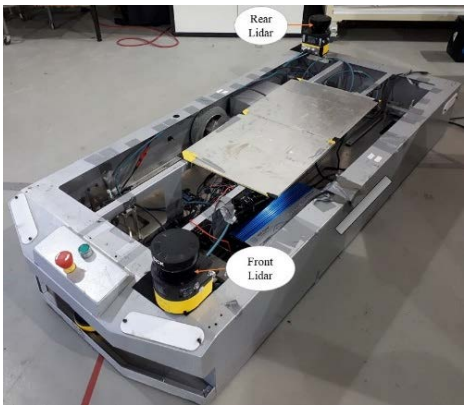


FIGURE 17. The Self-Guided Vehicle.

TABLE 4. SGV characteristic.

Parameter	Value
Vehicle Weight (m)	82 kg
Moment of inertia around z-direction (I_z)	5.5 $kg.m^2$
Distance of each wheel to point P (b)	38 cm
Wheel's radius	10 cm
Dimensions (length × width × height)	165 × 76 × 23 cm
Position of Center of Mass (length, width) with respect to point P	-4±1 cm, -2±1 cm

The equations of LSTM method are displayed as [33]:

$$g_c(t) = phi(W_{gx} a(t) + W_{gh} h(t-1) + b_g \quad (3)$$

$$i_g(t) = \sigma(W_{ix} a(t) + W_{ih} h(t-1) + b_i \quad (4)$$

$$f_g(t) = \sigma(W_{fx} a(t) + W_{fh} h(t-1) + b_f \quad (5)$$

$$o_g(t) = \sigma(W_{ox} a(t) + W_{oh} h(t-1) + b_o \quad (6)$$

$$s(t) = g_c(t) i_g(t) + s(t-1) f_g(t) \quad (7)$$

$$h(t) = \phi(s(t)) o_g(t) \quad (8)$$

where W_{gx}, W_{ix}, W_{fx} and W_{ox} indicate the correlation weight matrices; $a(t)$ and $h(t)$ represents the inputs and outputs of an LSTM block at time step t; Also, b_g, b_i, b_f, b_o are biases of input node, input gate, forget gate, and output gate respectively. σ and ϕ indicate sigmoid and hyperbolic tangent nonlinear activation functions.

APPENDIX C PROPERTIES OF SGV, LOADS AND ENVIRONMENT

E. SGV

The SGV has a differential drive and is configured by a Mini PC with Ubuntu 18.04 LTS operating system and ROS Navigation Stack. According to Figure 17, SGV includes two lidars to perceive the environment and two motors to move it. Moreover, the properties of SGV are displayed in Table 4.

F. LOADS

Five storage containers are considered as loads on the platform. Figure 18 shows them on the platform and their

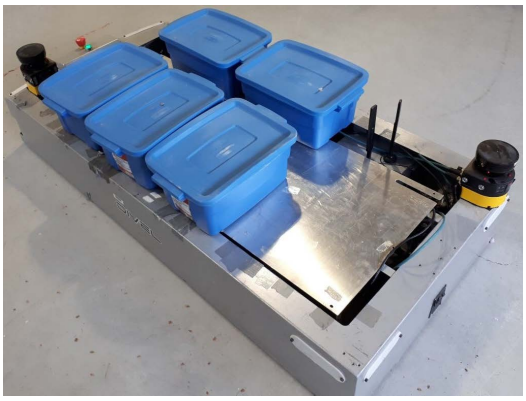


FIGURE 18. Loads on the platform.

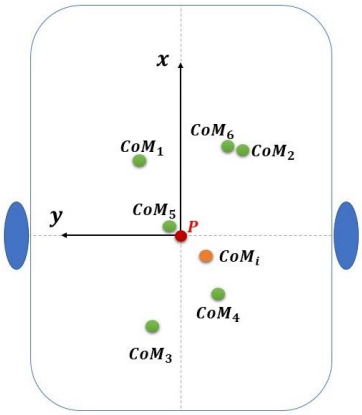


FIGURE 19. Real global CoM (CoM_g) on the platform for six scenarios.

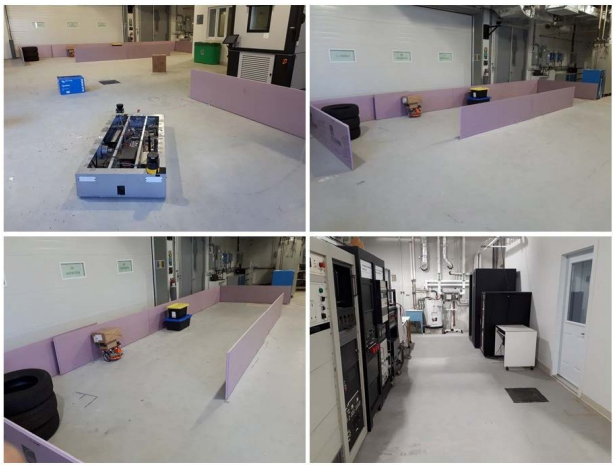


FIGURE 20. The experimental test's environment.

properties are displayed in Table 5. According to the scenarios' details, their positions are different on the platform.

Figure 19 displays the position of real global CoM (approximately) on the platform (green points) for all attempts where subscripts 1 to 6 mean attempt' number. Moreover, the red

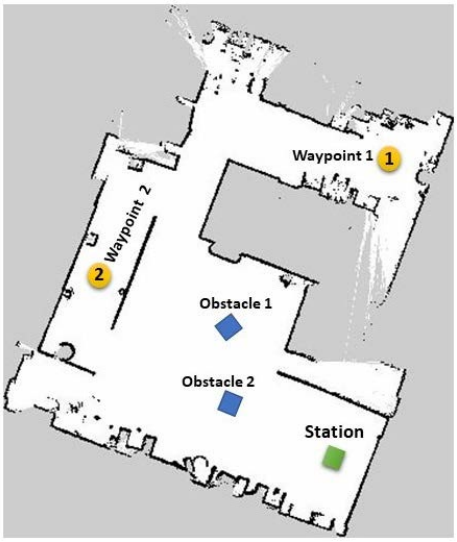


FIGURE 21. The map of experimental test's environment on rviz.

TABLE 5. Characteristics of each Load.

Parameter	Value
Weight	18.5 kg
Dimensions (length × width × height)	30 × 21 × 18 cm
Position of Center of Mass (length, width)	15±0.5 cm, 10.5±0.5 cm

and orange points show the middle of the wheels' axis (P) and the inertial CoM (CoM_i) respectively.

G. ENVIRONMENT

The environment of experimental test and its map are presented in Figures 20 and 21. Positions of the station, two obstacles, and waypoints 1 and 2 are displayed on the map.

REFERENCES

[1] E. A. Oyekanlu, A. C. Smith, W. P. Thomas, G. Mulroy, D. Hitesh, M. Ramsey, D. J. Kuhn, J. D. Mcghinnis, S. C. Buonavita, N. A. Looper, M. Ng, A. Ng'oma, W. Liu, P. G. Mcbride, M. G. Shultz, C. Cerasi, and D. Sun, "A review of recent advances in automated guided vehicle technologies: Integration challenges and research areas for 5G-based smart manufacturing applications," *IEEE Access*, vol. 8, pp. 202312–202353, 2020, doi: 10.1109/ACCESS.2020.3035729.

[2] X. Liu, G. Wang, and K. Chen, "Nonlinear model predictive tracking control with C/GMRES method for heavy-duty AGVs," *IEEE Trans. Veh. Technol.*, vol. 70, no. 12, pp. 12567–12580, Dec. 2021, doi: 10.1109/TVT.2021.3123176.

[3] S. Riazi, K. Bengtsson, and B. Lennartson, "Energy optimization of large-scale AGV systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 2, pp. 638–649, Apr. 2021.

[4] M. Graba, S. Kelouwani, L. Zeghmi, A. Amamou, K. Agbossou, and M. Mohammadpour, "Investigating the impact of energy source level on the self-guided vehicle system performances, in the industry 4.0 context," *Sustainability*, vol. 12, no. 20, p. 8541, Oct. 2020.

[5] J. Peng, X. Zhou, S. Jia, Y. Jin, S. Xu, and J. Chen, "High precision strain monitoring for lithium ion batteries based on fiber Bragg grating sensors," *J. Power Sources*, vol. 433, Sep. 2019, Art. no. 226692.

[6] B. K. Patle, A. Pandey, D. R. K. Parhi, and A. Jagadeesh, "A review: On path planning strategies for navigation of mobile robot," *Defence Technol.*, vol. 15, pp. 582–606, Aug. 2019.

[7] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Syst. Sci. Cybern.*, vol. SSC-4, no. 2, pp. 100–107, Jul. 1968.
[8] M. Chaudhari, L. Vachhani, and R. Banerjee, "Towards optimal computation of energy optimal trajectory for mobile robots," *IFAC Proc. Volumes*, vol. 47, no. 1, pp. 82–87, 2014.
[9] F. Valero, F. Rubio, and C. Llopis-Albert, "Assessment of the effect of energy consumption on trajectory improvement for a car-like robot," *Robotica*, vol. 37, no. 11, pp. 1998–2009, Nov. 2019.
[10] L. Xie, C. Henkel, K. Stol, and W. Xu, "Power-minimization and energy-reduction autonomous navigation of an omnidirectional Mecanum robot via the dynamic window approach local trajectory planning," *Int. J. Adv. Robot. Syst.*, vol. 15, no. 1, 2018, Art. no. 1729881418754563.
[11] A. Alajlan, K. Elleithy, M. Almasri, and T. Sobh, "An optimal and energy efficient multi-sensor collision-free path planning algorithm for a mobile robot in dynamic environments," *Robotics*, vol. 6, no. 2, p. 7, Mar. 2017.
[12] H. Kim and B.-K. Kim, "Minimum-energy translational trajectory planning for battery-powered three-wheeled omni-directional mobile robots," in *Proc. 10th Int. Conf. Control, Autom., Robot. Vis.*, Dec. 2008, pp. 1730–1735, doi: 10.1109/ICARCV.2008.4795788.
[13] M. Salazar, A. Alessandretti, A. P. Aguiar, and C. N. Jones, "An energy efficient trajectory tracking controller for car-like vehicles using model predictive control," in *Proc. 54th IEEE Conf. Decis. Control (CDC)*, Dec. 2015, pp. 3675–3680, doi: 10.1109/CDC.2015.7402789.
[14] A. Parra, D. Tavernini, P. Gruber, A. Sorniotti, A. Zubizarreta, and J. Perez, "On nonlinear model predictive control for energy-efficient torque-vectoring," *IEEE Trans. Veh. Technol.*, vol. 70, no. 1, pp. 173–188, Jan. 2021, doi: 10.1109/TVT.2020.3022022.
[15] M. A. Ali and M. Mailah, "A simulation and experimental study on wheeled mobile robot path control in road roundabout environment," *Int. J. Adv. Robot. Syst.*, vol. 16, no. 2, 2019, Art. no. 1729881419834778.
[16] M. F. Jaramillo-Morales, S. Dogru, and L. Marques, "Generation of energy optimal speed profiles for a differential drive mobile robot with payload on straight trajectories," in *Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot. (SSRR)*, Nov. 2020, pp. 136–141.
[17] J. L. Meriam, L. G. Kraige, and J. N. Bolton, *Engineering Mechanics: Dynamics*. Hoboken, NJ, USA: Wiley, 2020.
[18] M. Mohammadpour, L. Zeghmi, S. Kelouwani, M.-A. Gaudreau, A. Amamou, and M. Graba, "An investigation into the energy-efficient motion of autonomous wheeled mobile robots," *Energies*, vol. 14, no. 12, p. 3517, Jun. 2021.
[19] B. B. Mevo, M. R. Saad, and R. Fareh, "Adaptive sliding mode control of wheeled mobile robot with nonlinear model and uncertainties," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2018, pp. 1–5.
[20] Y. Jin and Chen, "Advanced estimation techniques for vehicle system dynamic state: A survey," *Sensors*, vol. 19, no. 19, p. 4289, Oct. 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/19/4289
[21] G. Reina, M. Paiano, and J.-L. Blanco-Claraco, "Vehicle parameter estimation using a model-based estimator," *Mech. Syst. Signal Process.*, vol. 87, pp. 227–241, Mar. 2017, doi: 10.1016/j.ymssp.2016.06.038.
[22] X. Jin and G. Yin, "Estimation of lateral tire–road forces and sideslip angle for electric vehicles using interacting multiple model filter approach," *J. Franklin Inst.*, vol. 352, no. 2, pp. 686–707, Feb. 2015, doi: 10.1016/j.jfranklin.2014.05.008.
[23] R. Rajamani, G. Phanomchoeng, D. Piyabongkarn, and J. Y. Lew, "Algorithms for real-time estimation of individual wheel tire-road friction coefficients," *IEEE/ASME Trans. Mechatronics*, vol. 17, no. 6, pp. 1183–1195, Dec. 2012, doi: 10.1109/TMECH.2011.2159240.
[24] C. Ren, X. Li, X. Yang, and X. Ma, "Extended state observer-based sliding mode control of an omnidirectional mobile robot with friction compensation," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9480–9489, Dec. 2019, doi: 10.1109/TIE.2019.2892678.
[25] Y. Wei, X. Zhang, Y. Shi, L. Xia, S. Pan, J. Wu, M. Han, and X. Zhao, "A review of data-driven approaches for prediction and classification of building energy consumption," *Renew. Sustain. Energy Rev.*, vol. 82, pp. 1027–1047, Feb. 2018, doi: 10.1016/j.rser.2017.09.108.
[26] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *Int. J. Forecasting*, vol. 22, no. 4, pp. 679–688, 2006.
[27] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robot. Autom. Mag.*, vol. 4, no. 1, pp. 23–33, Mar. 1997, doi: 10.1109/100.580977.
[28] K. Takaya, T. Asai, V. Kroumov, and F. Smarandache, "Simulation environment for mobile robots testing using ROS and gazebo," in *Proc. 20th Int. Conf. Syst. Theory, Control Comput. (ICSTCC)*, Oct. 2016, pp. 96–101, doi: 10.1109/ICSTCC.2016.7790647.
[29] Accessed: Jan. 20, 2022. [Online]. Available: https://www.ros.org/
[30] R. Szczepanski, T. Tarczewski, and K. Erwinski, "Energy efficient local path planning algorithm based on predictive artificial potential field," *IEEE Access*, vol. 10, pp. 39729–39742, 2022.
[31] H. You, Z. Ma, Y. Tang, Y. Wang, J. Yan, M. Ni, K. Cen, and Q. Huang, "Comparison of ANN (MLP), ANFIS, SVM, and RF models for the online classification of heating value of burning municipal solid waste in circulating fluidized bed incinerators," *Waste Manage.*, vol. 68, pp. 186–197, Oct. 2017, doi: 10.1016/j.wasman.2017.03.044.
[32] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," 2015, *arXiv:1506.00019*.
[33] C. Zheng, S. Wang, Y. Liu, C. Liu, W. Xie, C. Fang, and S. Liu, "A novel equivalent model of active distribution networks based on LSTM," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 9, pp. 2611–2624, Sep. 2019, doi: 10.1109/TNNLS.2018.2885219.

MOHAMMAD MOHAMMADPOUR received the B.S. degree in mechanical engineering and the master's degree in aerospace engineering from the Amirkabir University of Technology, Tehran, Iran, in 2011 and 2014, respectively. He is currently pursuing the Ph.D. degree with the Department of Mechanical Engineering, University of Quebec at Trois-Rivières (UQTR), Trois-Rivières, QC, Canada. His research interests include mobile robotics, motion planning, machine learning, dynamic, and control.

SOUSSO KELOUWANI (Senior Member, IEEE) received the Ph.D. degree in robotics systems from the Ecole Polytechnique de Montreal, in 2011. He completed his Postdoctoral Internship on fuel cell hybrid electric vehicles with the University of Quebec at Trois-Rivières (UQTR), in 2012. He developed expertise in the optimization and intelligent control of vehicular applications. He has been a Full Professor of mechatronics with the Department of Mechanical Engineering, since 2017, and a member of the Hydrogen Research Institute. He is the holder of the Canada Research Chair in Energy Optimization of Intelligent Transport Systems and the Noovelia Research Chair in Intelligent Navigation of Autonomous Industrial Vehicles. He holds four patents in USA and Canada. He has published more than 100 scientific articles. His research interests include optimizing energy systems for vehicle applications, advanced driver assistance techniques, and intelligent vehicle navigation taking into account Canadian climatic conditions. He is a member of the Order of Engineers of Quebec. He is the Winner of the Canada General Governor Gold Medal, in 2003. In 2019, his team received the First Innovation Prize in Partnership with DIVEL, awarded by the Association des Manufacturiers de la Mauricie et Center-du-Québec for the development of an autonomous and natural navigation system. In 2017, he received the Environment Prize from the Gala des Grands Prix d'Excellence en Transport, the Association Québécoise du Transport (AQTr), for the development of hydrogen range extenders for electric vehicles. He was the Co-President and the President of the Technical Committee of the IEEE International Conferences on Vehicular Power and Propulsion in Chicago, USA, in 2018, and in Hanoi, Vietnam, in 2019.

MARC-ANDRÉ GAUDREAU received the Ph.D. degree in industrial acoustics from the École de Technologie Supérieure (ÉTS), Montréal, QC, Canada, in 2016. He has been a Professor with the Department of Mechanical Engineering, University of Quebec at Trois-Rivières (UQTR), since2017, where he started the novel DUAL approach in the Drummondville Campus of UQTR. He is the Scientific Co-Director of the UQTR's Research Center on Intelligent Manufacturing. He is the holder of an Industrial Chair in the development of manufacturing intelligence, powered by Noovelia On top of his original expertise, he is working with the Canada Research Chair in Energy Optimization of Intelligent Transport Systems with Prof. Kelouwani on energy optimization of autonomous industrial vehicles.

BILEL ALLANI received the B.S. degree in industrial and automatic computing from the National Institute of Applied Sciences and Technologies, Tunisia, in 2020. He is currently pursuing the master's degree in electrical and computer engineering with the University of Quebec at Trois-Rivières (UQTR), Trois-Rivières, QC, Canada. His research interests include multiple mobile robot systems and charging strategies.

LOTFI ZEGHMI received the B.S. degree in control systems from the University of Science and Technology Houari Boumediene (USTHB), Algiers, Algeria, in 2017, and the master's degree in electrical engineering from the University of Quebec at Trois-Rivières (UQTR), Trois-Rivières, QC, Canada, in 2020. He is currently a Research Assistant with the UQTR. His research interests include mobile robotics and control systems.

ALI AMAMOU (Member, IEEE) received the B.S. degree in industrial computing and automatic science from the National Institute of Applied Sciences and Technology, Tunis, Tunisia, in 2013, the M.S. degree in embedded systems science from the Arts et Métiers ParisTech University, Aix en Provence, France, in 2014, and the Ph.D. degree in electrical engineering from the University of Quebec at Trois-Rivières (UQTR), Trois-Rivières, QC, Canada, in 2018. In May 2018, he started as a Postdoctoral Fellow at the Hydrogen Research Institute. His main research interests include optimization of energy systems for stationary and mobile applications, hybridization of energy sources for vehicular applications, and eco-energy navigation of the low-speed autonomous electric vehicle.

MASSINISSA GRABA received the B.S. degree in control and automation engineering from the Institute of Electrical and Electronic Engineering (IGEE-UMBB), Algeria, and the M.Sc. degree in smart aerospace and autonomous systems from the Université de Paris-Saclay, France, and Poznan University of Technology, Poland. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the University of Quebec at Trois-Rivières (UQTR), Trois-Rivières, QC, Canada. His research interest includes safe and energy-efficient trajectory planning of self-guided industrial vehicles in the context of sustainable manufacturing.