Sensors and Materials, Vol. 35, No. 4 (2023) 1497–1508
MYU Tokyo

S & M 3263

Energy Prediction of Cleanroom-type Differential Drive Mobile Robot Based on Recurrent Neural Network

Sarucha Yanyong, Poom Konghuayrob, Punyavee Chaisiri, and Somyot Kaitwanidvilai*

School of Engineering, King Mongkut's Institute of Technology Ladkrabang,
1 Chalongkrung, Ladkrabang, Bangkok 10520, Thailand

(Received Novemver 29, 2022; accepted February 21, 2023)

Keywords: energy sensing, energy prediction, machine learning, recurrent neural network, autonomous mobile robot

The battery charger time is a major issue for mobile robots. The study of the power usage of each component is important for optimizing the overall power consumption. Additionally, knowing the total energy consumption before commanding a robot to execute a task is essential for effective queue management and determining which robots are ready to execute tasks or move to the charging station. In this paper, we propose an energy modeling system consisting of an energy sensing technique, logging, and a recurrent neural network prediction model. The model is configured to recognize the dynamic system of the drive unit with the support of the robot operating system. The proposed model has a prediction error of only 3.58%. The simulation and experimental results demonstrate the effectiveness of the proposed system.

1. Introduction

The revolution of Industry 4.0 is being driven by autonomous mobile robots (AMRs) on the factory floor. These vehicles are capable of moving packages along assembly lines and even warehouses.[1] Advances in technology are leading to the introduction of intelligent sensors to the robot industry, with automated guided vehicles (AGVs) being upgraded to AMRs. AMRs were developed to find unique and collision-free paths but have many complex components.[2] Moreover, the energy consumed by AMRs is increasing owing to their increased use, especially in the context of Industry 4.0, where they play a significant role as a major source of transportation in assembly lines and factories. The high energy consumption of AMRs must be addressed to improve their efficiency and effectiveness.

The lack of energy-efficient solutions is currently the most pressing challenge facing the intelligence industry. To address this issue, several studies[3-5] have been performed. One such study, conducted by Zou *et al.*,[6] investigated battery charging and swapping strategies. They applied a semi-open queueing network to model the battery charging process, and their robots needed to recharge when their remaining battery level reached 5%. Tomy *et al.*[7] utilized the Markov decision process to develop an optimized charging scheduler, which significantly

*Corresponding author: e-mail: somyot.ka@kmitl.ac.th
https://doi.org/10.18494/SAM4263

improved the battery performance. Lu *et al.*[8] proposed a hybrid power distribution system that combined a fuel cell and a lithium battery to utilize the potential of both energy sources. The fuel cell can be rapidly charged, while the lithium battery is capable of releasing energy as needed. These studies focused on battery and charging queue management. Moreover, Fu *et al.*[9] conducted research on online scheduling and charging strategies of robots in warehouses and applied the shortest path algorithm to optimize the average charging delay. However, their research did not address the energy analysis of path planning in depth, which is a new and fascinating area of study.

Hou *et al.*[10] proposed the analysis of energy modeling and power measurement in mobile robots by modeling a four-wheeled Mecanum mobile robot. The energy measurement unit was integrated into the sensor, control, and motion systems, and the results indicated that the motion system was the most significant factor in energy consumption. To demonstrate this, the motion of the robot in a stable curve for 2.8 s at a constant speed of 1 m/s was tested. Our research builds on this work by focusing on energy modeling for complex routes planned by a navigation system. The proposed system includes an energy logging and prediction system, which is implemented using a robot operating system (ROS) to promote widespread adoption. The hardware components are designed for use in cleanroom environments, which are commonly found in the hard disk drive industry. Additionally, to build a mobile robot for use in industrial environments, the motion system must be capable of moving at a wide range of speeds. A brushless DC motor, which is capable of delivering high torque at low speeds, is used in our robot. However, this makes the dynamic system more complex, and conventional energy modeling systems used in DC motors may not be sufficient. Therefore, in this paper, we propose an energy modeling system using a recurrent neural network (RNN) to address the issue.

The proposed system uses a nonlinear autoregressive network with exogenous inputs (NARX) model. NARX is a type of RNN that has been specifically designed for time series prediction and modeling. It is capable of handling input sequences of varying length and can consider the temporal dependence between past and current inputs. In contrast, feed-forward neural networks, also known as artificial neural networks (ANNs), are designed for static input–output mapping. They are effective for identifying patterns and relationships in data but lack the ability to retain past inputs.

NARX models are more suitable than feed-forward neural networks when dealing with complex input–output relationships that involve time delays because NARX models can capture the dynamics of the input–output relationship over time by using information from past inputs and outputs to predict the current output, which makes NARX models particularly useful for time series data. In contrast, feed-forward neural networks may struggle to capture these dynamic and time-dependent input–output relationships. For this reason, NARX models can achieve better results for tasks that require handling dynamic and time-dependent input–output relationships such as motors. Many researchers have shown that NARX models can accurately predict the dynamics of nonlinear systems. Xin *et al.*[11] used NARX models to identify the dynamic model of an omnidirectional mobile robot, where the input data was the motor input voltage and the prediction output was the speed vector. Sahoo *et al.* [12] proposed the use of NARX models in a control system scheme to model dynamic system identification with the H_∞

filter. Altan *et al.*[13] applied NARX to the real-time control of a hexarotor unmanned aerial vehicle. Additionally, Wunsch *et al.*[14] conducted a comparison of RNNs. Their results showed that the NARX model was excellent for nonlinear time series prediction and outperformed other RNNs. They demonstrated that the NARX model was capable of capturing complexities and retaining information for over three times longer than other RNNs.

This paper is organized to cover the architecture, energy logging, ROS interfacing, dataset preparation, model design, experiment in an actual environment, and discussion of the proposed system. It also gives includes a discussion of how the NARX model is implemented within the proposed system and the results obtained from experiments conducted in an actual environment.

2. System Architecture

Our proposed system includes newly developed energy logging and energy prediction modules. The energy logging module keeps track of electrical and motion data, providing valuable insights for energy consumption analysis. The energy prediction module estimates the energy consumption of the planned route, allowing the more accurate management of robot tasks. The system architecture, which demonstrates the integration of these modules, is shown in Fig. 1.

The energy logging module is designed to track and analyze battery usage data in real time. It monitors various parameters such as power, current, voltage, battery cycles, and charging status using a microcontroller. The microcontroller transfers the data to the ROS, which is then processed by the robot's main computer.

In the ROS environment, the energy logging module is a new node, and its source code and executable file must be kept in a package specifically developed for the module. The measurement values received from the microcontroller are converted into ROS messages, and it is very important that these messages are compatible with other nodes. The ROS message structure is shown in Fig. 2.

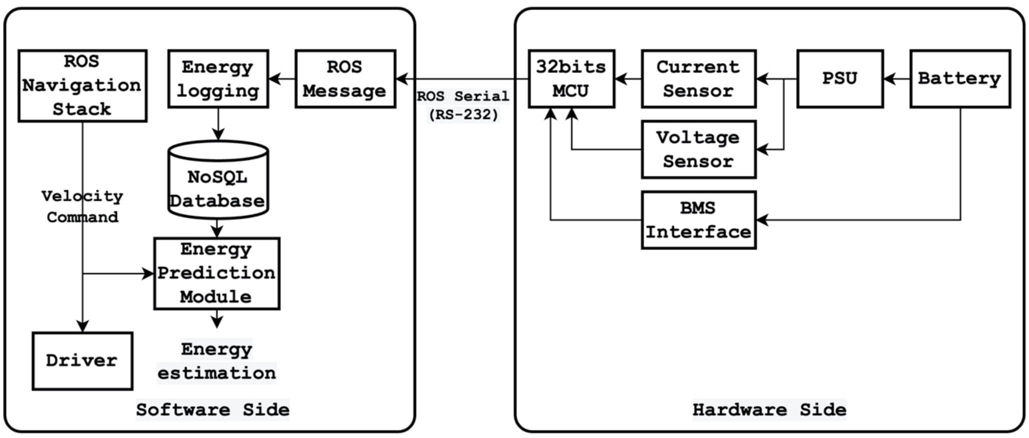


Fig. 1. Diagram of the system architecture.

```
std_msgs/Header
  uint32 seq
  time stamp
  string frame_id
float32 temperature
float32 voltage
float32 current
float32 power
float32 capacity
float32 capacity_max
float32 charge_status
uint8   cycles
```

Fig. 2. Structure of energy logging message.

To make the energy logging module functional, a new ROS package is developed, named the energy logging package, which provides the energy logging node. Additionally, this package includes a logging plugin for recording the data in a NoSQL database as a dataset for machine learning, as shown in the flowchart of data acquisition in Fig. 3(a).

The NoSQL database, containing a large number of energy consumption records, is transformed into datasets. The input datasets, consisting of two dimensions, the *X*-axis representing the linear velocity in m/s and the *Z*-axis representing the angular velocity in rad/s, are used in the training process. The goal of the training process is to optimize the model's parameters such as weight and bias to accurately fit the relationship between the input and output data, where the output represents the power usage in *W* associated with the operation of the motor. The training continues until either the maximum number of epochs is reached or the mean squared error (MSE) reaches saturation, indicating that the model has achieved convergence and is producing the best result based on the input data. This process is shown in Fig. 3(b).

The architecture of the NARX model, as depicted in Fig. 4, includes an input layer, a feedback layer, an output layer, input delays, and output delays.[15] During the training process, the inputs to the model are derived from the energy logging module, which measures the linear velocity (v) and angular velocity (ψ) of the mobile robot as it moves. The output (P) is the power usage of the robot. The unique feature of this model is its ability to use the output as the input for the next iteration. Equation (1) shows the inputs, including the current input, the delayed input ($t - n_u$), and the delayed outputs ($t - 1$ to $t - n_y$), which are used to predict the current output. The hyperparameters in the training process are given in Table 1.

As a result of the training process, the model parameters, including weights and biases, are adjusted and optimized. During the implementation of the trained model, the input is provided by the path planning algorithm in the form of 2D data, consisting of the planned linear velocity and angular velocities. The model uses these inputs to estimate the power required at each point along the path until the final destination is reached.

$$y(t) = f\left(y(t-1), y(t-2), \ldots, y(t-n_y), u(t-1), u(t-2), \ldots, u(t-n_u)\right) \tag{1}$$

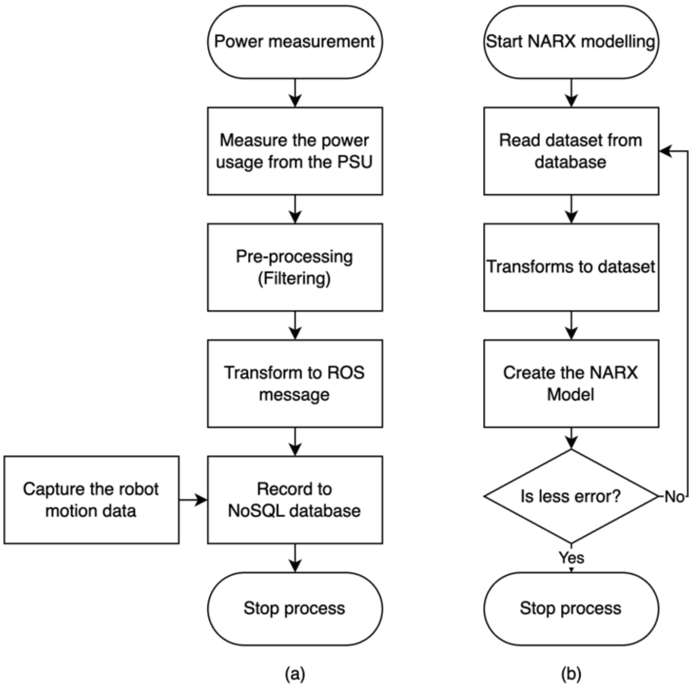


Fig. 3. (a) Flow chart for the data acquisition and (b) neural network modeling process.

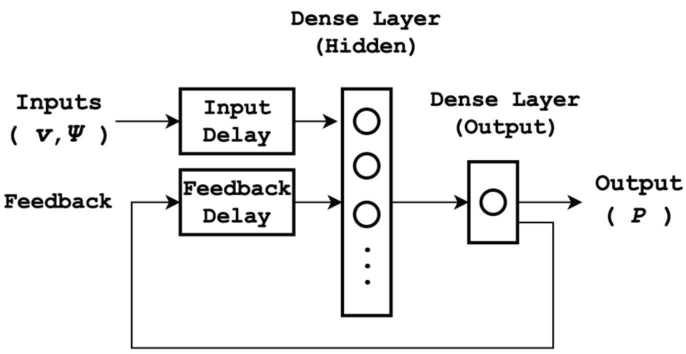


Fig. 4. Architecture of NARX neural network model.

Table 1
Hyperparameters.

Parameter	Value
Layer number	3
Time delay number (d)	5
Hidden layer neuron number (N)	6
Weight (w) and bias (b)	Random
Activation functions	Tan-Sigmoid
Training algorithm	Levenberg–Marquardt
Learning rate	Damping parameter (λ) in typical range 0.01–100
Maximum epochs	1000

Here, $u(t)$ is the input at time t, $y(t)$ is the output at time t, n_u and n_y are the input and output memory orders, respectively.

In the ROS, path planning is divided into two processes: global path planning and local path planning. Global path planning is performed using algorithms such as the A* algorithm. The path is calculated using the cost map, which is a 2D data representation that combines the map and LiDAR readings to display current obstacles. The output of these algorithms is a path represented as a series of waypoints that the robot can follow to reach its destination.

Local path planning is focused on a small area and designed to provide collision-free navigation. The conventional algorithm used for local path planning is the dynamic window approach (DWA). This algorithm calculates velocity commands for the robot by considering its current position and the environment in which it is operating.

Another local path planning algorithm is the timed elastic band (TEB). It operates by considering the robot's current position and velocity, then generating a set of candidate paths that take into account the robot's dynamics and environment. In both the DWA and TEB implementations, the algorithms generate a set of waypoints from the current position to the end of the area. The waypoints represent the X and Y positions of the robot for each step. However, TEB differs from DWA in that it plans the velocity commands, including the linear and angular velocities, before the first movement and adjusts the speed as the environment changes.

Our proposed system leverages this feature of TEB by incorporating the planned velocity commands into an energy prediction model to predict the power usage for each waypoint. The series of outputs represents the power usage profile, which can be used to calculate the energy consumption of the planned path as shown in Fig. 5.

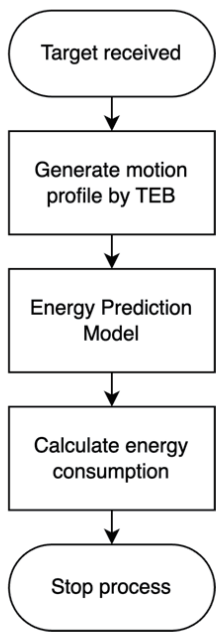


Fig. 5. Energy prediction process.

3. Experimental Results

The proposed system was implemented on an industrial-grade robot, as shown in Fig. 6, which is intended for deployment in a hard disk drive factory. The experiment was conducted in two processes: the training and evaluation processes. The training process focused on creating the dataset for the training process by controlling the mobile robot within any speed range. The energy logging module was used to record the linear and angular velocities as the input, as shown in Figs. 7(a) and 7(b), respectively, and the power usage as the output, as shown in Fig. 8(a). The recorded data were stored in a NoSQL database for later use in the training process.

In the training process, 50000 data points were used to adjust the model. As the principal step, the dataset was divided into 70% for training, 15% for validation, and the remaining 15% for testing. Table 2 shows the parameters during training, which consist of 35000 data points for training, 7500 data points for validation, and 7500 data points for testing, with each data point sampled at a rate of 10 ms. During training, the neural network achieved its best MSE of 0.0194 on the training set, 0.0186 on the validation set, and 0.0234 on the testing set. The neural network achieved its best performance after 215 epochs. The performance of each epoch is also shown in Fig. 9(a). The figure shows the trend of decreasing MSE until it reaches a stable value, which means that the best model has been found. The performance of the model was verified using the test data; the test data was fed into the model to generate the power output, as shown in Fig. 9(b).

To demonstrate the energy prediction performance, the proposed model was tested in an actual environment. The test cases consisted of seven destinations with a time delay of 10 s between points. The robot navigated through all the destinations to obtain the motion planning profile, which simulated the transfer of equipment between machines. The motion planning profile of the navigation algorithm takes into account obstacles and the shape of the room, as illustrated in Fig. 10. This information is derived from the actual environment. The profile can become more complex when there are rapid changes in velocity. This complexity makes it challenging to test the proposed model. The proposed energy prediction model takes the motion planning profile, which is the profile velocity with respect to time, as the input and returns the estimated power profile as the output, as illustrated in Fig. 11.

Fig. 6. (Color online) Cleanroom-type mobile robot.

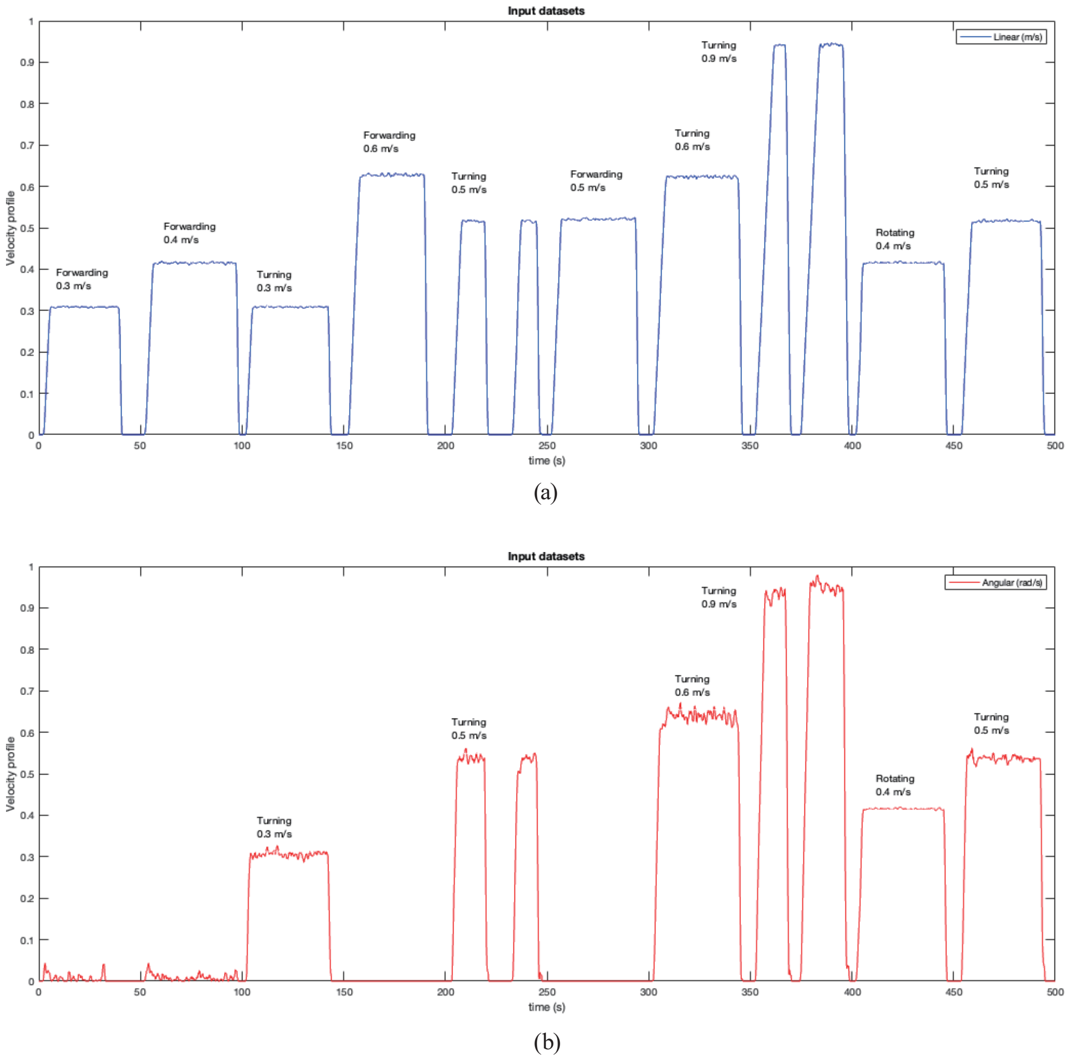


Fig. 7. (Color online) Input dataset: (a) linear velocity (m/s) and (b) angular velocity (rad/s).

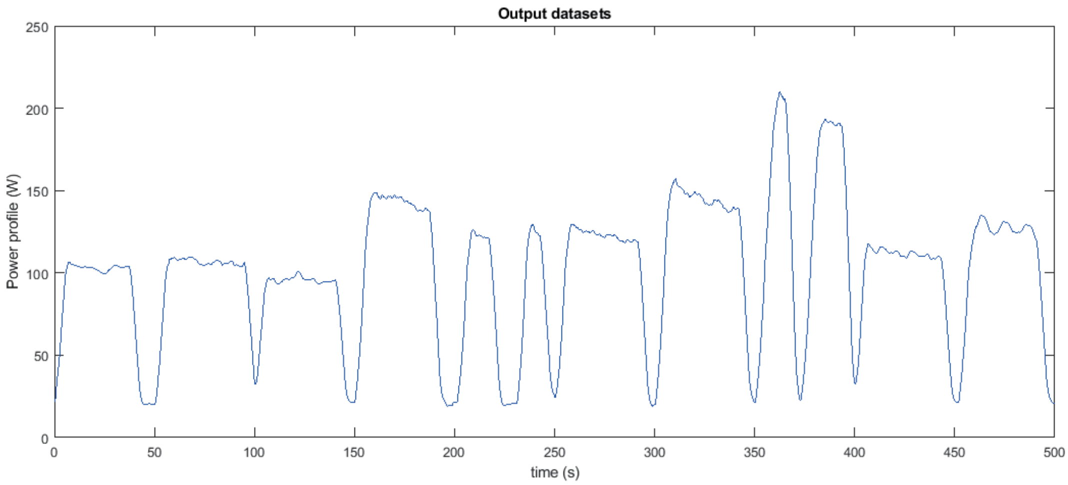
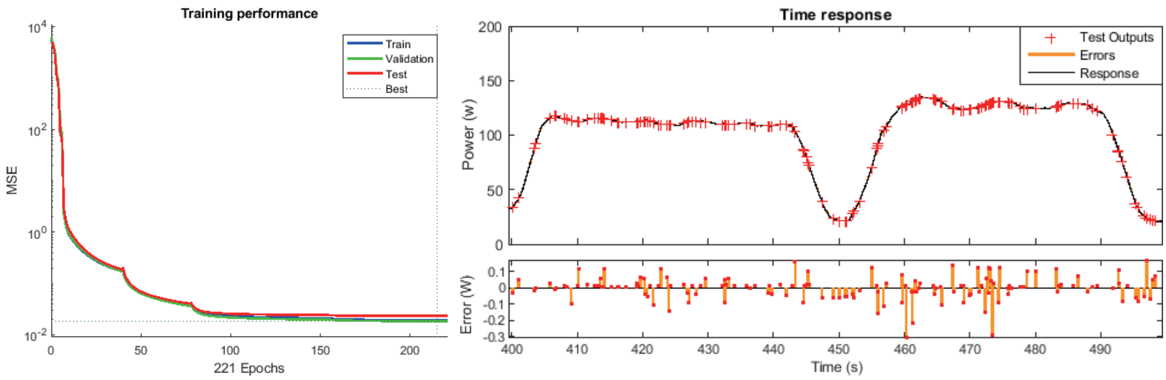


Fig. 8. (Color online) Output dataset.

Table 2
Training parameters.

Parameter	Value
Number of training data	35000
Number of validation data	7500
Number of test data	7500
Sampling time (ms)	10
Best training phase (MSE)	0.0194
Best validation phase (MSE)	0.0186
Best testing phase (MSE)	0.0234
Best epoch	215



(a) (b)

Fig. 9. (Color online) (a) Training performance and (b) time response prediction of testing dataset.

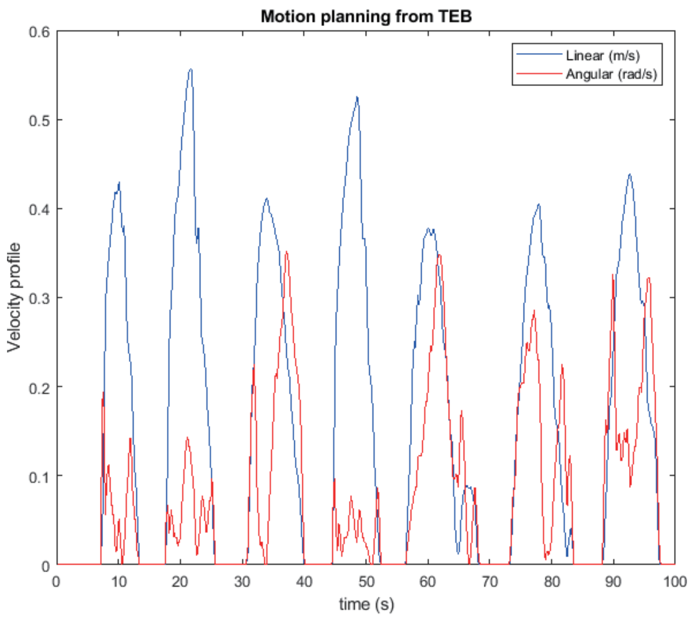


Fig. 10. (Color online) Motion planning profile obtained from TEB.

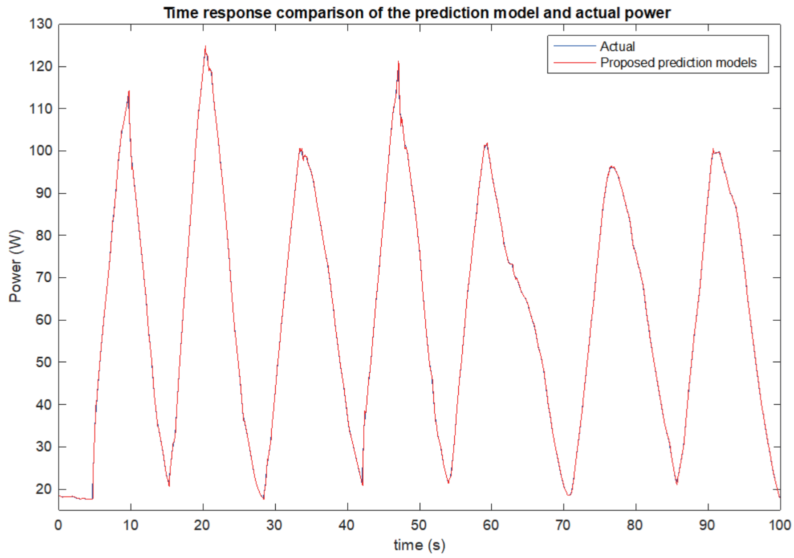


Fig. 11. (Color online) Time response comparison of proposed prediction model and actual power.

Table 3
Testing parameters.

Parameter	Value
Number of test data	10000
Testing time (s)	100
Sampling time (ms)	10
Best testing phase (MSE)	0.1668
Maximum error (W)	5W (During overshoot)
Actual energy consumption (J)	6297
Estimated energy consumption (J)	6523
Error (%)	3.58

We propose the use of the trapezoidal rule for calculating the energy consumption from the power data by integrating the power over a specific time interval. The trapezoidal rule is a numerical integration method that approximates the definite integral of a function by dividing the interval into small subintervals and approximating each area as a trapezoid. Therefore, the energy consumption can be accurately calculated by applying this method, as expressed by Eq. (2). To evaluate the precision of the estimated energy consumption, the percentage error in the energy is computed by Eq. (3). The results are presented in Table 3.

$$\int_{t_0}^{t_n} P(t)\,dt = \frac{1}{2}*\left(P(t_0)+P(t_1)\right)*(t_1-t_0)+\frac{1}{2}*\left(P(t_1)+P(t_2)\right)*(t_2-t_1)+\cdots + \frac{1}{2}*\left(P(t_{n-1})+P(t_n)\right)*(t_n-t_{n-1}) \quad (2)$$

Here, $P(t)$ is the power at time t and $t_n - t_{n-1}$ is the sampling time.

$$Percentage\ Energy\ Error = 100 * \frac{(Estimated\ Energy - Actual\ Energy)}{Actual\ Energy} \quad (3)$$

4. Conclusion

In this study, we developed a novel system for an AMR that receives its target destination from an automated system and transports an object. Predicting the energy consumption of the robot leads to the optimization of the efficiency of the overall robot system. Our proposed TEB path planning algorithm not only generates the path waypoints but also plans the motion along the route, which is a significant advantage of our energy modeling system. The proposed system implements a brushless DC motor, which is more complex than a conventional DC motor, but the proposed NARX model still provides an accurate estimate of energy consumption with an estimation error of 3.58%. Despite the complexity of the motion system and the test cases, our approach is still effective.

Acknowledgments

This work was supported by the Thailand Research Fund (TRF) through Research and Researcher for Industry (RRI) and Seagate Technology (Thailand) Co., Ltd. under research grant no. PHD62I0015.

References

1 M. Javaid, A. Haleem, R. P. Singh, and R. Suman: Cognit. Rob. **1** (2021) 58. https://doi.org/10.1016/j.cogr.2021.06.001
2 G. Fragapane, R. de Koster, F. Sgarbossa, and J. O. Strandhagen: Eur. J. Oper. Res. **294** (2021) 405. https://doi.org/10.1016/j.ejor.2021.01.019
3 M. Selmair, T. Maurer, C. H. Lai, and D. Grant: J. Power Sour. **521** (2022) 230894. https://doi.org/10.1016/j.jpowsour.2021.230894
4 W. Gong, J. Xiao, S. Han, X. Wu, N. Wu, and S. Chen: Energy Rep. **8** (2022) 940. https://doi.org/10.1016/j.egyr.2022.02.025
5 D. G. Logan, J. Pentzer, S. N. Brennan, and K. Reichard: J. Power Sour. **212** (2012) 130. https://doi.org/10.1016/j.jpowsour.2012.03.041
6 B. Zou, X. Xu, Y. Y. Gong, and R. de Koster: Eur. J. Oper. Res. **267** (2018) 733. https://doi.org/10.1016/j.ejor.2017.12.008
7 M. Tomy, B. Lacerda, N. Hawes, and J. L. Wyatt: Rob. Auton. Syst. **133** (2020) 103629. https://doi.org/10.1016/j.robot.2020.103629
8 X. Lu, Y. Wu, J. Lian, and Y. Zhang: Renewable Energy **171** (2021) 881. https://doi.org/10.1016/j.renene.2021.02.135
9 X. Fu, Z. Cheng, and J. Wang: Comput. Ind. Eng. **153** (2021) 107097. https://doi.org/10.1016/j.cie.2021.107097
10 L. Hou, L. Zhang, and J. Kim: Energies **12** (2018) 27. https://doi.org/10.3390/en12010027
11 L. Xin, Y. Wang, and H. Fu: Symmetry **12** (2020) 1430. https://doi.org/10.3390/sym12091430
12 H. K. Sahoo, P. K. Dash, and N. P. Rath: Appl. Soft Comput. **13** (2013) 3324. https://doi.org/10.1016/j.asoc.2013.02.007
13 A. Altan, O. Aslan, and R. Hacioglu: Proc. 2018 6th Int. Conf. Control Engineering & Information Technology (IEEE, 2018) 1–6. https://doi.org/10.1109/CEIT.2018.8751829
14 A. Wunsch, T. Liesch, and S. Broda: Hydrol. Earth Syst. Sci. **25** (2021) 1671. https://doi.org/10.5194/hess-25-1671-2021
15 M. Wei, M. Ye, J. B. Li, Q. Wang, and X. Xu: IEEE Access **8** (2020) 189236. https://doi.org/10.1109/access.2020.3031340

About the Authors

Sarucha Yanyong received his B.Eng degree in mechatronics engineering and his M.Eng degree in electrical engineering from King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand, in 2012 and 2014, respectively. He is currently working towards a Ph.D. degree in electrical engineering at KMITL. His research interests are in the areas of mobile robotics, machine learning, control systems, and artificial intelligence. (sarucha.y@gmail.com)

Poom Konghuayrob received his B.Eng, M.Eng, and Ph.D. degrees in electrical engineering from King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand, in 2012, 2013, and 2018, respectively. Currently, he works at KMITL in the Department of Electrical Engineering. His research interests are in the areas of robust control, DC/AC inverters, solar cell fields, and artificial intelligence. (poom.ko@kmitl.ac.th)

Punyavee Chaisiri received his B.Eng., M.Eng., and Ph.D. degrees from King Mongkut's Institute of Technology Ladkrabang (KMITL), Thailand in 2008, 2011, and 2020, respectively. He has been working at KMITL's Department of Electrical Engineering since 2012, and his research interests include high-voltage engineering applications and sensors. (punyavee.ch@kmitl.ac.th)

Somyot Kaitwanidvilai received his B.Eng and M.Eng degrees in electrical engineering from King Mongkut's Institute of Technology Ladkrabang (KMITL), Thailand, in 1996 and 2000, respectively. He received his Ph.D. degree in mechatronics engineering from the Asian Institute of Technology in 2005. Currently, he works at KMITL in the Department of Electrical Engineering. His research interests include artificial intelligence in power and energy systems, control systems, mechatronics, power electronics, and robust and adaptive control in power systems. (somyot.ka@kmitl.ac.th)