Machine Learning Model for a Dynamic Path Planning Problem

Thong Tran Huu, Duy Nguyen Duc and Narameth Nananukul*

School of Management Technology, Sirindhorn International Institute of Technology, Thammasat University, Pathum Thani 12120, Thailand

*Corresponding author email: narameth@siit.tu.ac.th

Abstract. Due to an advancement in Industry 4.0 technology, various autonomous systems have been developed in order to increase the operational efficiency. This paper considers an application of Industry 4.0 technology to an autonomous transportation operation. The paper focuses on applying a machine learning technique to a dynamic path planning problem where real-time randomized obstacle locations are considered. The routes or the solutions from the dynamic path planning problem are determined by an A-star algorithm, which are then used to build machine learning models based on an artificial neural network. The models were developed to discover the relationship between the input and output of the dynamic path planning problem. The structure of the network which is defined by the number of intermediate layers and the number of nodes is provided, where the overall accuracy is used to evaluate the setting efficiency. The proposed methodology was tested with a problem that consists of 7 types of paths, and the number of randomized obstacles fluctuated from 1 to 8. The paths were generated based on a layout of a consumer product warehouse. The proposed model succeeded in predicting the robot paths with 98.5% prediction accuracy.

Keywords: Dynamic path planning; Artificial neural network (ANN); Machine learning; Randomized obstacle; A-star algorithm.

1. Introduction
In many areas, mobile robots have been successfully used, and the advancement in warehouse robots has received attention from researchers [1, 2]. Many multi-national companies can benefit from the advancement of warehouse robots, where they can be used to efficiently load and unload items within a warehouse. The dynamic path planning problem is an important transportation problem that must be considered in order to achieve autonomous navigation of robots in a practical transportation environment. Efficient path generation for mobile robots can reduce the delivery time and also the wear-out and maintenance cost of mobile robots. Due to the importance of the path planning aspect of mobile robots, it has received a lot of attention from researchers.

When considering path planning methodology, the A-star algorithm is a popular method for dynamic path planning, used for autonomous robots. However, when the size of the considered transportation area is large, the computational time of the A-star algorithm is large, which makes the algorithm impractical for large-scale path planning problems.

In recent years, technology development has enabled the ability to discover hidden knowledge from the vast amount of available data. For the dynamic path planning problem, machine learning technology can help make better decisions in a short time by transforming historical operational data into insight. Artificial neural network (ANN) models can be used to learn and discover the relationship between the transportation environment and the optimal path. The insight gained from the data enables the ANN to...
predict the most suitable operational path in a short amount of time which is beneficial for autonomous navigation.

In this research, a machine learning model based on an ANN is proposed, to learn and discover insight from a dynamic path planning problem where real-time randomized obstacles are considered. This is considered a novelty of this research. The paper is structured as follows: a literature review of relevant research to path planning and machine learning methodologies is provided in section 2. The experiment and environment used in this research are presented in section 3. In section 4, the main results are presented with managerial insight, and a conclusion and a possible future research direction are provided in section 5.

2. Literature Review

In this section, a literature review related to ANN and the path planning problem is presented. The A-star algorithm has been applied to transportation applications with different metrics and transportation environment. The algorithm is considered a heuristic search that can be used to determine the shortest path. The A-star algorithm, as a path planning algorithm, has been applied to many different applications, and has been successfully implemented and tested. Zhang and Zhao [3] proposed an integrated algorithm that combined the A-star and Dijkstra algorithms. The objective is to guide a mobile robot to move along the optimal path from the origin to the provided destination while the robot avoids all obstacles and arrives safely with reliability during the journey. Zhang et al. [4] considered an improved A-star algorithm for an automated guided vehicle (AGV). Their results revealed that the efficiency of the path planning process can be improved by removing redundant inflection points and nodes. Gochev et al. [5] investigated a collision avoidance method and the A-star algorithm in a dynamic environment with numerous agents for an AGV system. In their study, AGVs needed to complete different tasks and avoid unexpected obstacles that cause accidents. Haotun et al. [6] addressed the path planning problem for a robotic feeding-push machine in the dairy-farm industry. Mohan et al. [7] discussed the application of a path planning methodology for a warehouse mobile robot. An enhanced A-star algorithm was applied to determine the shortest path while monitoring the energy consumption of the robot.

A mobile robot is commonly used to verify and test the algorithm in a simulated environment. Aprilia et al. [1] implemented the A-star algorithm to determine the shortest path in a warehouse. In their research, a robot in a warehouse relied on the visible light communication method to define a walkway. Kusuma and Machbub [8] successfully applied the A-star algorithm for a humanoid robot. Besides finding the shortest path, the robot has an ability to search for a new route if the destination is suddenly changed by users or if the robot moves outside the waypoints. Contreras-González et al. [9] used an ANN to predict the AGV location by using data that contained the accuracy from the model and the actual location. Thus, their model gave a smooth and accurate point-to-point travel plan. Their research was tested in a field without a controlled environment. Flórezet et al. [10] proposed a management strategy for an autonomous system for robots, specifically for an AGV. The authors proposed an integrated kinematic controller, based on an adaptive ANN and a genetic algorithm, in order to implement an obstacle avoidance system. The proposed methodology can be used to manage the movement of vehicles to reach a specified level of autonomy and a controller that takes into account both kinematic and dynamic aspects. Kurdi et al. [11] considered a path planning problem with an intelligent control using an ANN. In their research, the authors considered input from different resources to construct collision-free paths for navigating robots among real-time obstacles, based on the results from an ANN. Zhang et al. [12] addressed a problem related to the line-of-sight path control of an AGV in adverse conditions by using an ANN. The ANN was employed to learn and regulate the parameters that are used to control the AGVs, in order to reduce the effects of wind-induced sideslips.

Based on the literature review, the application of machine learning for path planning with dynamic obstacles is still an active research area. This research focuses on developing an ANN model that can be used to determine the parameters that specify the relationship between the input and output of a dynamic path planning problem. This can then be applied to predict the path based on different scenarios of input data. More details of the problem description and methodology can be found in section 3.
3. Problem Statement and Methodology

3.1. Problem Statement
In this research, an A-star algorithm for a dynamic path planning problem was implemented using Excel Visual Basic (VBA), based on Teil Zwei [13]. The dynamic path planning problem considers a transportation area based on a 2-dimensional grid map implemented in an Excel worksheet, as shown in Figure 1. The size of the problem is defined by the number of rows and columns, which then are used to define a transportation area represented by a rectangle. The origin is represented by a pink square, and the destination is represented by a light green square. There are two types of obstacles considered in the path planning problem. Fixed obstacles (or fixed blocks) are represented by blue rectangles, and dynamic obstacles are green squares. The number of dynamic obstacles can be specified, and they are generated at random locations within the transportation area. When the size of the transportation area increases, the complexity of the problem increases significantly, which requires impractical runtimes to determine a solution.

![Figure 1. Grid map used for dynamic path planning.](image)

For instance, there are 3 dynamic obstacles in figure 1. The selected path is represented by a red (thick) line, which connects the starting point and the destination point. The shortest path is generated, based on the A-star algorithm to avoid collision with all obstacles (fixed and dynamic) on the map.

3.2. Methodology
In this section, the data encoding process and ANN model are introduced. An example of a transportation area is shown in figure 2, where the size is 21 rows x 21 columns in an Excel worksheet. To distinguish between the different components within the transportation area, each cell is encoded by using the following logic. A cell is encoded with “1” if it represents an origin, a destination, or an obstacle, and encoded with “0” otherwise. Note that fixed obstacles exist permanently and can be omitted from the model. Because a dynamic obstacle located on a line segment blocks the whole segment, an improved encoding scheme is proposed to remove noise from the input data. All cells from a blocked segment are encoded with “1”. Each segment is named based on the orientation, horizontal (H) or vertical (V), as shown in figure 3. This encoding scheme reduces the combination of input data and greatly reduces the amount of computation.

The output of the ANN is the selected path, which consists of a series of cells from the origin to the destination. The selected path is encoded with a number (e.g., 0, 1, 2) that is used to represent a path id. In figures 2 and 3, the selected path is represented by a red (thick) line which connects the origin to the endpoint. The path is generated based on the A-star algorithm to avoid collisions with 4 obstacles, which exist on segments 1H2, 3V1, 2V2, and 3H2. An example of the input dataset is shown as sample 1 in table 1, where the selected path was encoded as type 6.
To prepare input data, different scenarios of input data were generated from the Excel worksheet. First, the locations of fixed obstacles were defined. Then, the locations of dynamic obstacles were generated randomly. The locations of obstacles were encoded and used as input for an ANN. Next, the A-star algorithm was executed to generate the path from the origin to the destination. The path was encoded with a number, which is used to represent an output of the ANN. The process was repeated until there were sufficient training data. The dataset was divided into training and testing sets for the machine learning model. Eighty percent of the data was used for training to develop an ANN model for path prediction, while twenty percent of the data was used for evaluating the accuracy of the ANN model. The number of samples used in the data set is based on the problem complexity. In this study, 194 samples that contain 12 segments with dynamic obstacles ranging from 1 to 8 obstacles were used to train the ANN model. An example of an ANN model is represented in figure 4. The model is based on a feed forward network where intermediate layers are allowed. For the current study, an ANN model was developed to determine a path, based on the generalization capability of the ANN model, for new datasets.

### Table 1. Example of input and output data from an ANN.

| Sample | 1H1 | 1H2 | 2H1 | 2H2 | 3H1 | 3H2 | 1V1 | 1V2 | 2V1 | 2V2 | 3V1 | 3V2 | Type |
|--------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|
| 1      | 0   | 1   | 0   | 0   | 0   | 0   | 0   | 1   | 0   | 1   | 0   | 0   | 6    |
| 2      | 0   | 1   | 0   | 0   | 0   | 1   | 0   | 0   | 1   | 1   | 0   | 0   | 4    |

To validate the accuracy of the trained ANN model, the input and output datasets described in Section 3 were used to train the ANN model by using the pattern recognition function of MATLAB R2019.

#### 4. Experimental Results

To validate the accuracy of the trained ANN model, the input and output datasets described in Section 3 were used to train the ANN model by using the pattern recognition function of MATLAB R2019.

##### 4.1. Data Preparation

In this study, the parameters of the transportation problem include the number of fixed blocks, obstacles, and segments, as shown in table 2.

### Table 2. Basic parameters.

| Parameter                    | Value          |
|------------------------------|----------------|
| Number of fixed blocks       | 4              |
| Number of obstacles          | from 1 to 8    |
| Number of segments           | 12             |
In the training process, 194 datasets were generated with 7 distinct types of paths, as shown in table 3. Path type 6 has the highest number of datasets because it has the shortest length. It is selected if there are no obstacles on the path.

| Path type | Number of datasets | One hot encoding | %  |
|-----------|--------------------|------------------|----|
| 1         | 18                 | [1 0 0 0 0 0 0]   | 9% |
| 2         | 19                 | [0 1 0 0 0 0 0]   | 10%|
| 3         | 14                 | [0 0 1 0 0 0 0]   | 7% |
| 4         | 29                 | [0 0 0 1 0 0 0]   | 15%|
| 5         | 20                 | [0 0 0 0 1 0 0]   | 10%|
| 6         | 67                 | [0 0 0 0 0 1 0]   | 35%|
| 7         | 27                 | [0 0 0 0 0 0 1]   | 14%|
| **Total** | **194**            |                  |    |

4.2. ANN Configuration, Result, and Discussion

Different ANN configurations were used to build ANN models that capture the dependency between the input and output. An ANN was configured, based on the number of intermediate layers and the number of nodes. The overall accuracy of each configuration is listed in table 4. The training stage was limited to 1000 iterations, without any limit on the runtime.

| Number of intermediate layers | Number of nodes per layer | Overall Accuracy (%) |
|-------------------------------|---------------------------|----------------------|
| 1                             | 11                        | 93.8                 |
| 2                             | [11 7]                    | 95.4                 |
| 3                             | [11 7 6]                  | 98.5                 |
| 4                             | [11 7 6 5]                | 85.6                 |

To test the accuracy of an ANN model, data were separated into two disjoint sets: training set (80%) and testing set (20%). This splitting procedure helps to avoid overfitting because the model learns from the training dataset and is tested by the testing dataset which was not used during the learning stage. According to table 4, the overall accuracy from a configuration with 3 intermediate layers is the highest (98.5%). Having more number of intermediate layers will not lead to an increase in accuracy, as shown in table 4, the accuracy of ANN with 4 intermediate layers is lower than 98.5%.

Therefore, the best number of intermediate layers is 3, and the number of nodes for intermediate layers 1, 2, and 3 are 11, 7, and 6, respectively. The best ANN model is shown in Figure 5.

5. Conclusions

This paper considers an application of Industry 4.0 technology to an autonomous transportation operation. A methodology for determining solutions from a dynamic path planning problem was proposed. The method was implemented with models based on an ANN. The input and output data encoding scheme was presented. The A-star algorithm was used to generate the output paths that were used in the training dataset. Different configurations of the ANN were studied to determine the best configuration. The proposed methodology was tested with a transportation environment based on a consumer product warehouse with a dimension of 21x21. The number of obstacles fluctuated from 1 to
8. The proposed model succeeded in predicting the robot paths with 98.5% prediction accuracy. Further study is also needed to improve the performance of the methodology in more complex environments, especially when considering larger problems.

Acknowledgments
This study was supported by the Thammasat University Research Fund, Contract No. TUGR 2/33/2562.

References
[1] Aprilia B S, Kurniawan E, Ramdhani M and Rizal A 2019 Design and implementation A* Algorithm on movement system robot in the warehouse Journal of Physics: Conference Series 1367 012066
[2] Aggarwal S and Kumar N 2020 Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges Computer Communications 149 270–99
[3] Zhang Z and Zhao Z 2014 A multiple mobile robots path planning algorithm based on A-star and Dijkstra algorithm International Journal of Smart Home 8 75-86
[4] Zhang Y, Li L-L, Lin H-C, Ma Z and Zhao J 2017 Development of path planning approach based on improved A-star algorithm in AGV system Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering IoT as a Service 276–9
[5] Gochev I, Nadzinski G and Stankovski M 2017 Path planning and collision avoidance regime for a multi-agent system in industrial robotics Machines, Technologies, Materials 11 519-22
[6] Haotun L, Xiaoxi C, Gang W, Shixiong L, Liu Y and Zheng Y 2018 Research of dynamic path planning of feeding-pushing robot based on A star algorithm ASABE Annual International Meeting: American Society of Agricultural and Biological Engineers 1
[7] Mohan L J and Ignatious J 2018 Navigation of mobile robot in a warehouse environment International Conference on Emerging Trends and Innovations in Engineering and Technological Research 1-5
[8] Kusuma M and Machbub C 2019 Humanoid robot path planning and rerouting using A-star search algorithm International Conference on Signals and Systems 110-5
[9] Contreras-González A-F, Hernández-Vega J-I, Hernandez-Santos C and Palomares-Gorham D-G 2016 A method to verify a path planning by a back-propagation artificial neural network LANMR 98-105
[10] Flórez C A C, Rosário J M and Amaya D 2018 Control structure for a car-like robot using artificial neural networks and genetic algorithms Neural Computing and Applications 1-14
[11] Kurdi M M, Dadykin A K, Elzein I and Ahmad I S 2018 Proposed system of artificial neural network for positioning and navigation of UAV-UGV Electric Electronics, Computer Science, Biomedical Engineerings' Meeting 1-6
[12] Zhang Y, Zhang Y, Liu Z, Yu Z and Qu Y 2018 Line-of-Sight Path Following Control on UAV with Sideslip Estimation and Compensation Chinese Control Conference 4711-6
[13] Zwei T 2015 A Search Algorithm with VBA https://www.vitoshacademy.com/tag/excel-vba-a-star/.