



Autonomous mobile robots in manufacturing: Highway Code development, simulation, and testing

A. Liaqat¹ · W. Hutabarat² · D. Tiwari² · L. Tinkler³ · D. Harra¹ · B. Morgan³ · A. Taylor² · T. Lu² · A. Tiwari²

Received: 21 February 2019 / Accepted: 1 August 2019 / Published online: 29 August 2019
© The Author(s) 2019

Abstract

A dynamic and flexible manufacturing environment presents many challenges in the movement of autonomous mobile robots (AMRs), leading to delays due to the complexity of operations while negotiating even a simple route. Therefore, an understanding of rules related to AMR movement is important both from a utility perspective as well as a safety perspective. Our survey from literature and industry has revealed a gap in methodology to test rules related to AMR movement in a factory environment. Testing purely through simulations would not be able to capture the nuances of shop floor interactions whereas physical testing alone would be incredibly time-consuming and potentially hazardous. This work presents a new methodology that can make use of observations of AMR behaviour on selected cases on the shop floor and build up the fidelity of those simulations based on observations. This paper presents the development of a Highway Code for AMRs, development of simulation models for an ideal AMR (based on the rules from the Highway Code), and physical testing of real AMR in an industrial environment. Finally, a behavioural comparison of an ideal AMR and a real AMR in five scenarios (taken from the shop floor of an industrial partner) is presented. This work could enable informed decisions regarding the implementation of AMRs through identification of any adverse behaviours which could then be mitigated either through improvements on the AMR or through establishing shop floor protocols that reduce the potential impact of these behaviours.

Keywords Manufacturing · Autonomous mobile robots · Simulation · Highway Code

1 Introduction

Robots in the industrial sector have evolved from powerful, stationary machines into sophisticated, mobile platforms to address a broader range of automation needs. Autonomous mobile robots (AMRs) utilise feedback from sensors to navigate their environment [1]. This is unlike traditional automated guided vehicles (AGVs), which are

restricted to predetermined paths using magnetic/electrical wires, inertial navigation, optical sensors, or infrared sensors [2]. In further contrast, an AMR has a greater in-built intelligence and is able to detect obstacles present on its path and recalculate a route around the obstacle to get it to its destination [1]. AMRs have found applications in various industries due to their high efficiency and low operating costs. They are currently seen as a critical component of ‘Industry 4.0’ for ideas such as smart factories and self-organisation [3].

In the production of aircraft wings within Airbus factories, large-scale AGVs are utilised to move wing assembly structure and aircraft components between the manufacturing cells. As the rate of production increases, the movements and availability of the AGVs become constraints, requiring many manual interventions to deal with deadlocks (e.g. traffic jams). In order to address increased logistical movements, a more flexible system is needed to reduce the need for a dedicated floor space and manual interventions, hence the drive towards fully autonomous vehicle technology. The challenge however is to develop a

Electronic supplementary material The online version of this article (<https://doi.org/10.1007/s00170-019-04257-1>) contains supplementary material, which is available to authorized users.

✉ D. Tiwari
d.tiwari@sheffield.ac.uk

¹ Airbus UK, Broughton CH4 0DR, UK

² Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield S1 3JD, UK

³ Factory 2050, Advanced Manufacturing Research Centre, Sheffield S9 1ZA, UK

reliable system that can fully integrate into the existing factory environment addressing complex logistic operations with simple solutions available in the market.

Need for a Highway Code A literature review of relevant rules/algorithms revealed that the AMR rules are often studied in isolation, focusing on a particular category (e.g. path planning for delivery tasks in AMR) rather than as a whole [4, 5]. Thus, the idea of creating a Highway Code, which accumulates all the different rules regardless of their category, was initiated. A Highway Code, which collects and structures the rules into their respective category, would broaden the understanding of the rules that determine an AMR's operational capability. The implementation of a Highway Code would provide several benefits: making it easier and safer for humans, and human-driven vehicles to interact with AMRs; aid in greater autonomy by enabling resolution of issues such as traffic jams, right of way, without resorting to a central authority such as a fleet controller; and enhancing interoperability. The development of such a Highway Code for AMRs will require significant testing to verify the desired performance in all foreseen scenarios. To perform these tests experimentally would be incredibly time-consuming, and potentially hazardous before the interplay between the various rules is fully understood. As such, this development will initially rely on simulation.

Need for simulating AMR behaviour In order to achieve acceptance of the Highway Code in industry, it needs to be proven out. However, our survey from literature and industry has yielded no methodology to test such rules in a factory environment. Additionally, testing through simulation alone would not capture the nuances of shop floor interactions and therefore would not be acceptable to industry, whereas physical testing alone is not practical. It is therefore necessary to develop a new methodology that can make use of observations of AMR behaviour on selected cases on the shop floor and build up the fidelity of those simulations based on those observations. The methodology must allow iterations where new observations have been made. In this paper, we demonstrate how this can be done for a subset of the identified Highway Code focusing on route finding and motion deadlocks.

The experiments on AMR systems indicated that in the case of any motion deadlocks, AMRs do not follow a standard set of rules on how the vehicles interact or communicate effectively with the surroundings and between themselves. In a static environment, this can be mitigated through careful programming. A dynamic and flexible environment presents a greater challenge leading to delays in AMR movement. However, most types of obstructions (e.g. humans or other AMRs) can move themselves to accommodate the obstructed AMR, if only the AMR could communicate its intentions based on an accepted set of rules and priorities. Conversely,

with a better understanding of how humans and other vehicles are likely to behave an AMR could anticipate their next move and make decisions accordingly.

We introduce the notion of the ideal AMR and real AMR. The ideal AMR is a simulation model which is closely derived from the Highway Code, whose behaviour bears little or no relation to the actual real-world AMR. The real AMR is either an actual AMR on the shop floor or a simulation model which has been tweaked such that its behaviour is demonstrably closer to the physical system. The rest of the paper demonstrates how these concepts can accommodate the gradual build up of simulation fidelity such that relevant parts of the Highway Code can be tested in a credible way.

Related research An overview of available literature on path planning of AGVs including different path planning approaches, robot control architectures, analyses of sensor systems, and velocity estimation techniques has been provided in reference [6] and a review of methodologies to optimise scheduling, dispatching, and routing problems is presented in reference [7]. Extending this body of work, the authors have conducted a detailed review of the rules related to AMR movement (presented in Section 3). Researchers working on flexible manufacturing system have reported studies focusing on design and simulation of new algorithms focussing on pickup and delivery [8], selection of best control rules for a multiple-load AGV [9], and optimisation of AMR scheduling [10]. Modelling and simulating complex systems are often the only way for their full analysis and design. However, for non-trivial problems, this approach will raise computational issues, stiff set of equations, or numerical instability [11]. Recently, Bai et al. utilised the same simulation approach to investigate the potential benefits of centralised decision-making by solving the multi-AGV motion planning problem in a direct and centralised way. Centralised motion planning is computationally expensive; therefore, this study involved investigations in schemes that are sensitive to solution quality but insensitive to computation time [12]. A survey of performance and computational requirements of various motion planning and control techniques has been discussed for assessing compatibility and computational trade-offs between various choices at the system level [13], without considering their implications on actual shop floor scenarios. Although the above-mentioned work studies the behaviour of AMRs in individual scenarios, the approach followed in this study simulates the vehicle's behaviour in multiple scenarios that it may encounter.

This work addressed these gaps in the existing body of research by first compiling a Highway Code for AMRs and identifying suitable motion planning algorithms to incorporate into the simulated model of an ideal AMR. This idealised simulation is then compared against physical testing of a physical AMR to identify differences in emergent behaviour. These differences are then used as the basis for simulation

model of a real AMR which mimics the physical system, and can then subsequently be used to investigate the potential benefits of implementing particular elements of the Highway Code. In this work, we illustrate this by comparing the simulated behaviour of the ideal AMR and real AMR in five real world scenarios from the shop floor. The developed simulation models could identify potential problems before implementation of AMR on the shop floor. This could be used as a tool to evaluate the benefits of implementing a particular element on the shop floor.

2 Methodology

The process of methodology has been demonstrated as a flow chart in Fig. 1 and explained in detail below:

2.1 Identification of rules for Highway Code

The Airbus production site for airplane wings in Broughton, Chester was used as a case study. The Highway Code was developed for production facilities that are not optimised towards the utility of AMRs. The process of constructing the Highway Code constituted of two phases: Phase one aimed at establishing the database of AMR rules for the Highway Code. Data was collected, four categories of AMR rules were identified (details in Section 3), and finally, classifications

were established to further define the characteristics of an individual rule and to help make the database easier to navigate. Phase two aimed at development of a toolkit in Excel. A detailed description of the development of Highway Code is discussed in [14].

2.2 Suitable algorithm for simulation

2.2.1 Data collection

Primary data was collected and validated through our industrial partner via on-site observations, and interviews/questionnaires with respondents who had a direct relation to AMRs. The shop floor information was collected by recording several parameters, such as type, size and speed of dynamic, overhead and static objects, number of pedestrians, unexpected pedestrian behaviour, junctions, slow and fast zones, size of paths for vehicle and pedestrian, possible paths for real AMR, and length and shape of paths.

2.2.2 Identification of algorithms for sampling-based motion planning

A real scenario was captured from a manufacturing site of an industrial partner. The simulation models were developed in V-REP (Virtual Robot Experimentation Platform) by Coppelia Robotics, and the programming language was Lua [15]. The

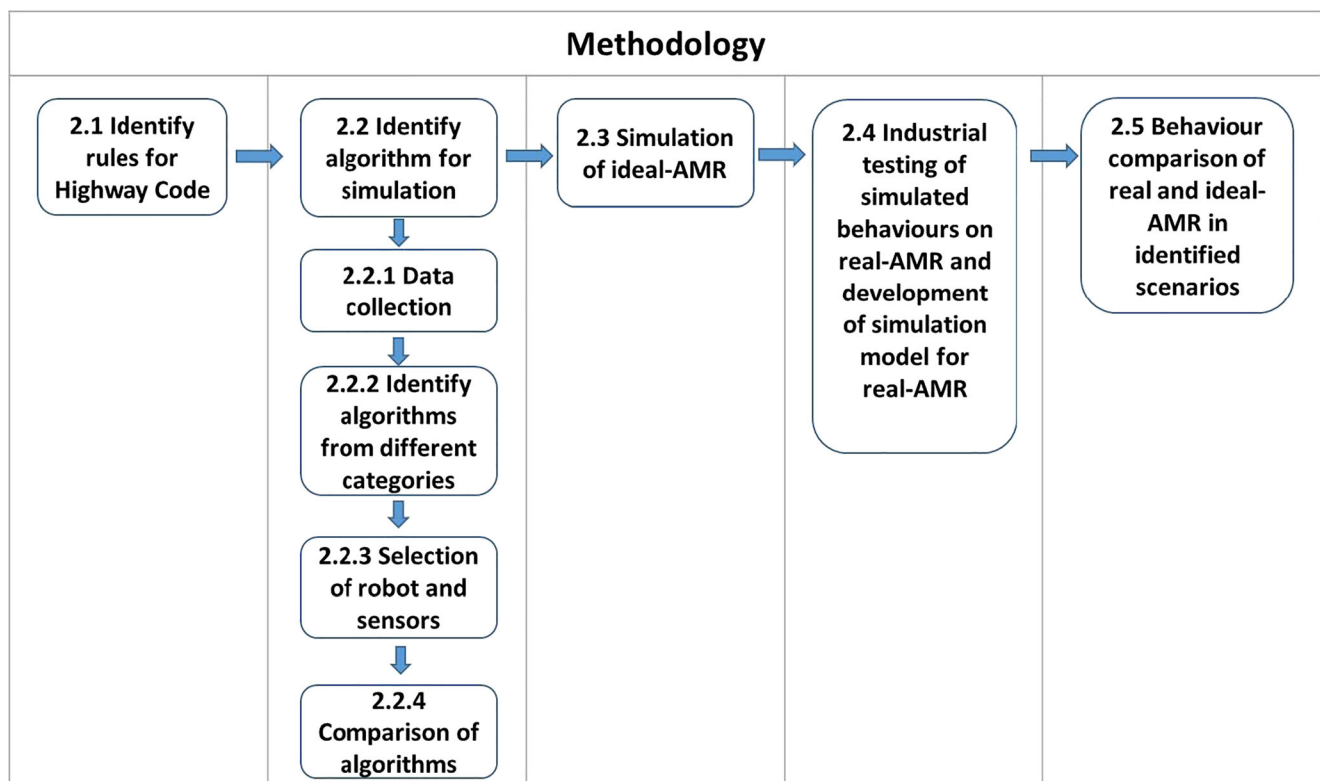


Fig. 1 A flow chart illustrating the methodology process

Open Motion Planning Library (OMPL) integrated in V-REP contains many advanced algorithms for sampling-based motion planning [16]. Three algorithms from separate categories were identified for simulation [16]: Rapidly exploring Random Tree (RRT), Probabilistic RoadMap Planner (PRM), and Kinodynamic motion Planning Interior-Exterior Cell Exploration (KPIECE). RRT motion planning algorithm is a tree-based algorithm in the category of single-query in geometric planner and is considered a suitable algorithm to solve motion planning problems involving obstacles. PRM algorithm belongs to multi-query category and is able to execute multiple threats at the same time. KPIECE algorithm belongs to control-based planner that considers dynamic and kinematic constraints, and is capable of tackling the motion planning problem for a system with complex dynamics.

2.2.3 Selection of robot and proximity sensors

The kinematics of the simulated real AMR is based on differential drive, mimicking that of the physical AMR. Eight proximity sensors were added to represent the 360° laser scanning ability of the real AMR. The data from the shop floor study (e.g. size of the AMR, detection distance of sensors) was integrated so that the simulation models represented the actual shop floor environment.

2.2.4 Behaviour comparison

The simulated behaviour of AMR was compared with path data for real AMR. The path data for real AMR was obtained by capturing the videos of AMR's movement from industrial sites, dividing video into frames, and then processing each frame to obtain the coordinates of the AMR using image processing tools in MATLAB. The coordinates were obtained by using built-in functions in image processing tools. The coordinates for the simulated path data from the simulation model were extracted by using a built-in function in V-REP. The actual path data and the simulated path data of AMR were compared and analysed in MATLAB for discrepancies in positions and direction angles of real-world and the simulated robot. The comparison revealed that RRT was the most suitable algorithm for simulating behaviour of real AMR.

2.3 Simulation model of ideal AMR

After identifying the most suitable algorithm for simulating real AMR behaviour, simulation models for ideal AMR were developed. The method adopted for the production of simulation models was 'modular', meaning that the overall real AMR behaviour was categorised and simulated as eight individual behaviours (identified from Highway Code) and after that they were integrated into one simulation model that exhibited all eight behaviours. The data from the shop floor

study was integrated so that the simulation models represented the actual shop floor environment as accurately as possible and provided an understanding as to the kinds of scenarios that an AMR on the shop floor may encounter.

2.4 Industrial testing and simulation model of real AMR

Physical testing at the manufacturing site of an industrial partner was performed. The real AMR chosen for this trial was a MiR200 from Mobile Industrial Robots ApS which features two laser scanners to provide a 360° protection field and LiDAR data for navigation, and a structured-light infrared camera to detect overhead obstacles. The industrial testing revealed the behaviours or lack of them in real AMR. Based on this, the simulation model for real AMR was developed by modifying the previously developed simulation model of ideal AMR.

2.5 Comparison of behaviours of real AMR and ideal AMR in identified scenarios

Finally, five scenarios were identified from the shop floor of an industrial partner that were considered to potentially prohibit the implementation of real AMR on the shop floor. The behaviours of real AMR and ideal AMR (including any resulting delays, blockages) were compared in these five scenarios.

3 Development of Highway Code

The process of constructing the Highway Code comprised two phases:

Phase one aimed at establishing the database of AMR rules for the Highway Code. Following the literature review, 53 rules/algorithms related to AMR were identified that were most relevant for an aerospace manufacturing environment. Next, four categories of AGV rules were identified, and classifications were established, to further define the characteristics of an individual rule and to help make the database easier to navigate.

Phase two aimed at development of Highway Code in Excel toolkit that involved the following steps: Identification of a suitable software development methodology, identification of features required for the functionality of the toolkit, a risk assessment to identify and mitigate potential risks, and finally, development and validation of toolkit features. A detailed description of the development of the Highway Code along with a list of algorithms/rules is provided in [14].

In this paper, only a subset of the full Highway Code is investigated. The purpose of this paper is not a thorough test of the Highway Code but to present the methodology on how we are currently conducting the test. We hope to publish the

result of the full test of the Highway Code in due course. The rules were categorised into four categories, a brief overview is as follows:

1. **Traffic regulation:** This category included rules related to AMR traffic systems in a manufacturing environment, explored by testing distinct set of algorithms: coordination planning algorithm, incremental coordination algorithm, and complete coordination algorithm [17]. A set of traffic control rules for the navigation of AMRs is also discussed in [4, 18].
2. **Dispatchment:** This category included rules related to pickup, dispatch, and delivery. Twelve heuristic dispatching rules have been presented in literature including Shortest Travel Time/Distance Rule, First Comes-First Serve Rule, and Longest Idle Vehicle Rule [5, 19].
3. **Load selection:** Ho and Chien presented nine rules on load selection including prioritising the load that an AMR has to carry [20], e.g. should the load which is heaviest is prioritised before the lighter loads, or loads that are going to area X are prioritised instead of area Y [21]? This category also included the best control rule for multiple-load AMR systems [7].
4. **Routing method:** Routing method is the most discussed topic related to AMR rules. A review of methodologies to optimise scheduling and routing in AMRs is provided in reference [7]. This category included hybrid multi-objective genetic algorithms for dynamic scheduling and routing of jobs and AMRs in flexible manufacturing systems [18, 22].

4 Identification of suitable algorithm for motion planning

Discussion with the industrial partners revealed that rules 45 and 46 from the Highway Code were crucial in avoiding congestion on the shop floor. Rule 45 states that ‘by using features such as camera or sensors the vehicle can determine the distance of an obstacle. To avoid collision a shutdown criterion is set where obstacles cannot be closer than the max distance to the vehicle’. Rule 46 states that ‘vehicle will seek to find the shortest distance possible without consideration to anything, such as other vehicles, outside of its programmed environment’. These two rules formed the basis of simulation models depicting the behaviour of AMRs.

4.1 Observed interactions of real AMR

A real scenario, based on rules 45 and 46 from the Highway Code, was captured from a manufacturing site of an industrial partner (Fig. 2a). It was as follows: An AMR was carrying

tools towards tools calibration room. A large vehicle stopped in AMR’s path and at the same time, a person walked through the path; AMR was able to detect both obstacles and updated the position of the walking person. It recalculated a new path to avoid both obstacles. The obstacles are larger than the detecting distance, but AMR was still able to avoid them.

4.2 Simulation models

The simulation models developed in V-REP utilised the following three algorithms (details in Section 3.1) from the OMPL [16]: RRT, PRM, and KPIECE. The programme of the simulated robot is illustrated in the flow chart depicted in Fig. 2b. The programme starts with the status as 1, where the robot plans the path from current position to goal position without considering obstacles. While the robot is following the original path, the sensors of simulated robot keep checking if there is any obstacle on its path. If sensors detect any obstacle (anywhere in the vicinity except at the front) within a distance of 0.5 m, the robot stops for 30 s. After 30 s, if sensors do not detect any obstacle and the robot has not reached the destination, the programme goes back to status 1. However, if an obstacle is detected at the front within a distance of 1 m, the status is set to 2, and a path planning algorithm is implemented for robot to plan a new path to avoid the obstacles. After overtaking the obstacles, if there are no other obstacles on the path, the robot follows the path. The programme ends when the robot has reached the goal position.

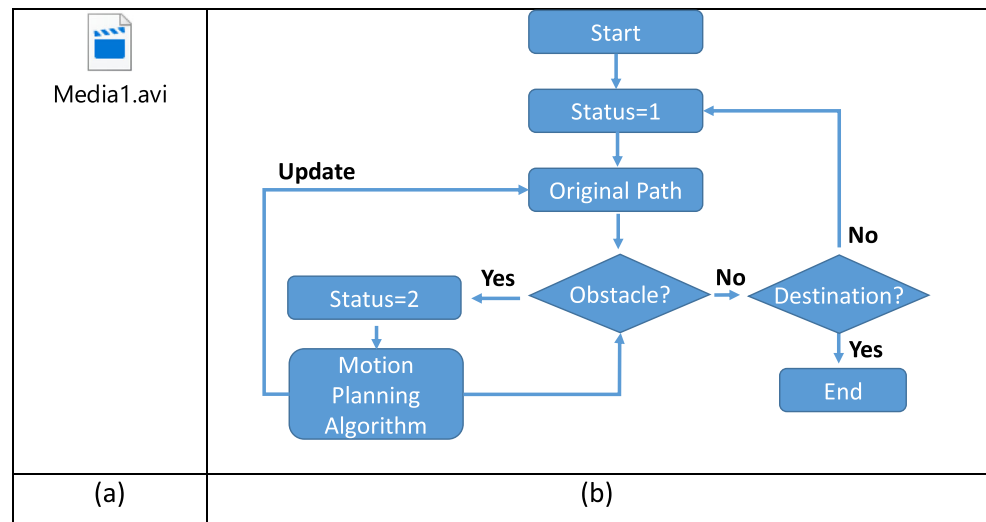
4.3 Comparison between path data from real AMR and simulated path data

Details of obtaining the simulated path data and path data from real AMR are discussed in Section 2.4. The simulated path data was compared with path data obtained from real AMR using MATLAB. Two comparison methods were developed to quantify the discrepancies between the real-world and the simulated path: First method calculated the difference of the positions between two paths, and the second method analysed the direction angles of real AMR and the simulated robot.

4.3.1 Analysis of the position discrepancies

To begin the algorithm, the x - and y -coordinates of real AMR and simulated AMR must be aligned and normalised to the same length scale. The positional discrepancies between the simulated path data and path data from real AMR were analysed by plotting data from both the paths in same graph and matching the coordinates for starting and end points. The y -coordinates were matched for both the paths and the error p was defined as the difference between x -coordinates of their respective paths. The total error throughout the path is given by Eq. 1:

Fig. 2 **a** Video for the real scenario captured from a manufacturing site of an industrial partner (see ESM 11). **b** Flow chart of the programme of the simulated robot



$$\text{Total error} = \sum_{y=0}^{y=N} Ep = \sum_{y=0}^{y=N} |x(\text{real}) - x(\text{sim})| \quad (1)$$

where $x(\text{real})$ and $x(\text{sim})$ are x -coordinates for real and simulated data respectively, N is the number of simulations, and Ep is the error in position. Due to the stochastic nature of path planning algorithms, the simulated path varied each time the algorithm was implemented. Therefore, the total error of the simulated path is different each time the simulation model is executed. The model was executed for 80 simulations and the mean error of each algorithm was calculated to compare the behaviour of algorithms. The moving average was also calculated to identify the number of simulations required to obtain the actual mean error of each algorithm. The moving average of the error is shown in Fig. 3a. After initial fluctuations, the mean errors converged after 40 simulations. As the error (Ep) was of stochastic nature, standard deviation and mean error were obtained as shown in Fig. 3b. The RRT algorithm depicted smallest average and mean error as compared with KPIECE and PRM algorithms and was concluded to be suitable for simulating the behaviour of real AMR.

4.3.2 Analysis of the direction angle discrepancies

The method of quantifying the position discrepancies may not be sufficient in some scenarios, such as shown in Fig. 4a, where black line represents the real robot path, and red and blue lines represent two simulated paths. Even though the positional error obtained from two simulated paths is similar, red path is a better match with real path as compared to blue path. This demonstrates the need for an analysis of the discrepancies in the angle of travel direction. The comparison of direction angles of the real AMR and simulated path data was done by keeping the y -

coordinates constant. The discrepancy of direction angles was obtained by Eqs. 2, 3, and 4:

$$\theta = \tan^{-1} \left(\frac{\Delta y}{\Delta x} \right) = \tan^{-1} \left(\frac{y_2 - y_1}{x_2 - x_1} \right) \quad (2)$$

$$E = |\theta(\text{real}) - \theta(\text{sim})| \quad (3)$$

$$\text{Total error} = \sum_{y=0}^{y=N} E \quad (4)$$

where θ is the direction angle of a path, $\theta(\text{real})$ and $\theta(\text{sim})$ are direction angles of real and simulated paths, E is the error in direction angle of travel, and N is the number of simulations. The error of direction angle was also calculated in moving average to demonstrate the mean error of direction angle in each algorithm. As shown in Fig. 4b, the mean error converges after 40 simulations. Once again, the RRT algorithm had the smallest error of direction angle and KPIECE algorithm has the largest error of direction angle. Thus, RRT algorithm was again proved to be suitable for simulating the behaviour of real AMR.

In both the analysis methods, V-REP and MATLAB were run on a standard PC and a complete analysis run took approximately 20 min to convergence.

5 Simulation model for ideal AMR based on RRT algorithm

Rules 45 and 46 from the Highway Code formed the basis of identification of eight behaviours of an ideal AMR. An ideal AMR would follow these eight behaviours on how to interact effectively with the surroundings especially to solve any motion deadlocks. Simulation models were developed based on these behaviours using RRT algorithm.

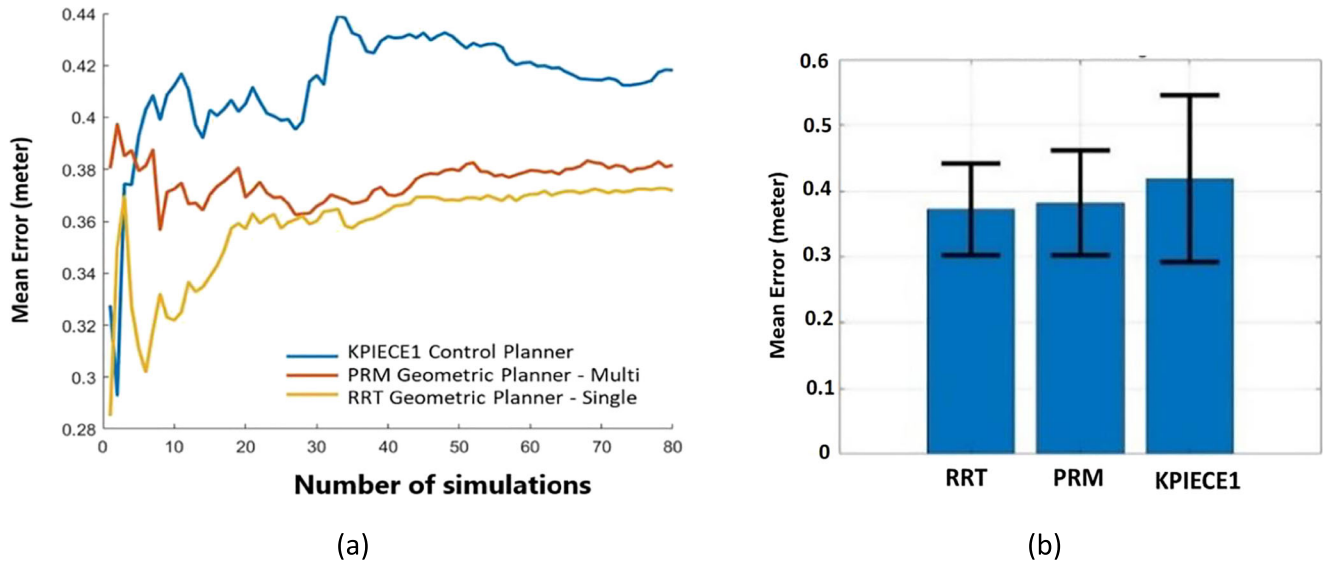


Fig. 3 **a** The moving average of positional error obtained from KPIECE, PRM, and RRT algorithms. **b** Standard deviation for KPIECE, PRM, and RRT algorithms

The method adopted for the production of the model was ‘modular’, which means that the individual behaviours, shown in Fig. 5, were converted into individual simulations and then they were integrated into a single simulation model.

The eight behaviours of an ideal AMR are described as:

Behaviour 1: It is aware of an obstacle at the front and will stop if the obstacle becomes close.

Behaviour 2: It is able to travel around an obstacle and, after completing a successful overtake, is able to return to its previously defined path and continue its task.

Behaviour 3: When travelling around an obstacle, it is able to identify whether or not the path used to overtake the obstacle is clear. If the path is not clear, then it should wait until it is safe to perform the overtaking manoeuvre.

Behaviour 4: When approached from behind, it is able to stop and allow the faster obstacle to overtake it. It should then begin moving again when it notices that the obstacle has successfully passed, or after a given amount of time has passed.

Behaviour 5: It is aware of its own height and therefore is able to assess whether or not it can fit underneath overhead obstacles.

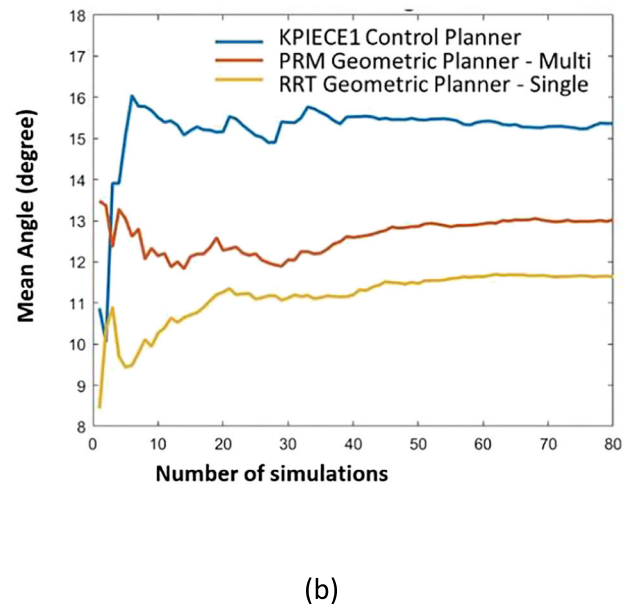
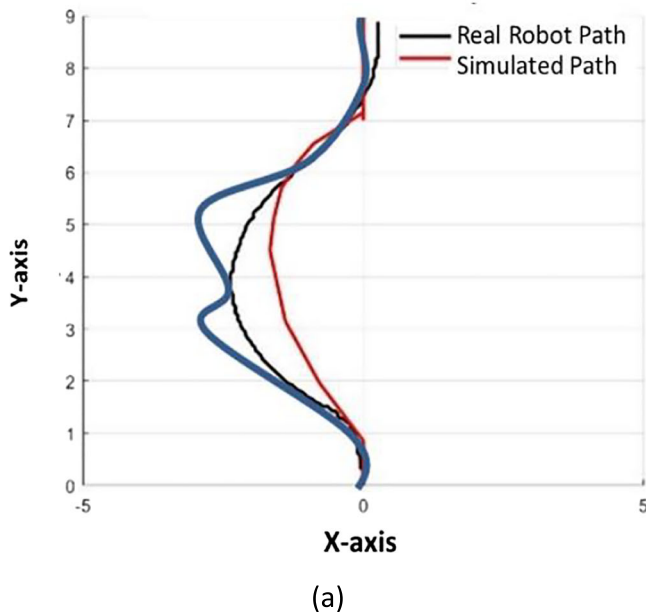


Fig. 4 **a** A scenario where black line represents the real robot path, and red and blue lines represent two simulated paths. **b** The mean error in direction angles obtained for KPIECE, PRM, and RRT algorithms

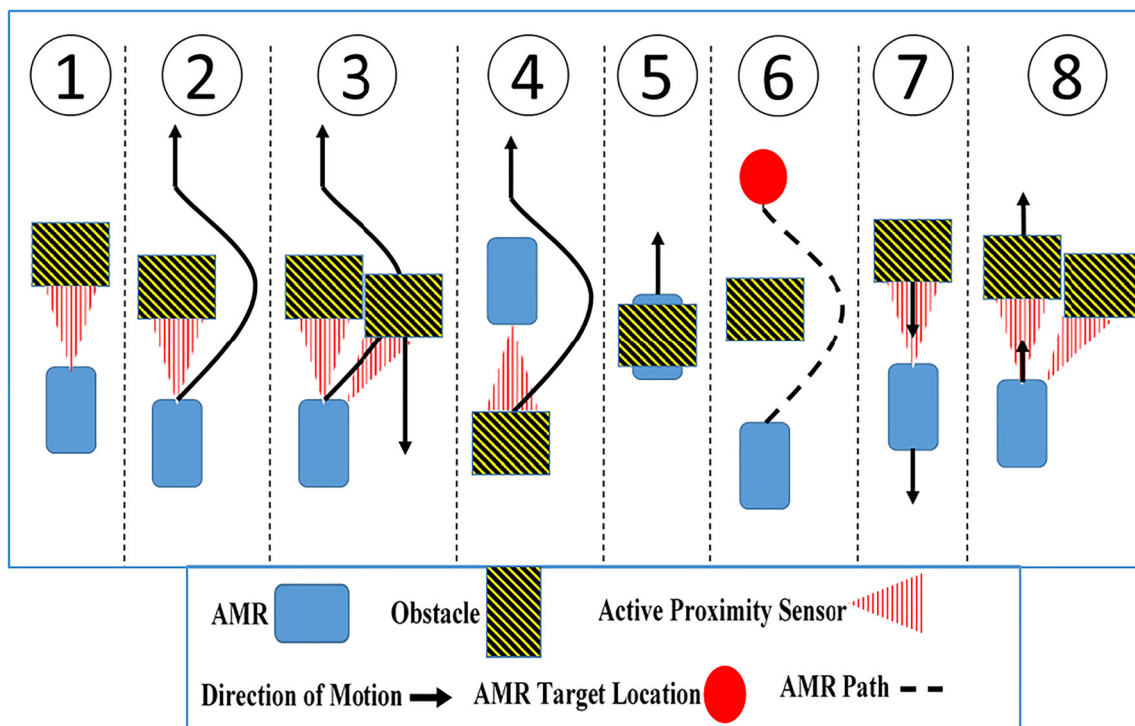


Fig. 5 Diagrammatic representation of the behaviours of an ideal AMR

Behaviour 6: It is able to utilise a path planning algorithm to find a path through a predetermined map of the shop floor that will take it to its target location. It is also able to travel along this path, without the path being directly visible to the vision sensor on board.

Behaviour 7: If an obstacle gets too close to the front, then it should reverse backwards.

Behaviour 8: Building on behaviour 3, it should be able to cancel any queued overtaking algorithms if its original path becomes clear before the overtaking algorithm is executed.

The simulation model for an ideal AMR was tested using functional or black-box testing method in which test-cases are derived from the specification of the entity to be tested [23]. The black-box testing method used in this work was Category Partitioning; further details can be found in [24].

6 Physical testing of real AMR and development of simulation model of real AMR

In order to identify differences with the real AMR behaviour, the eight simulated behaviours of ideal AMR were tested on a real AMR at the manufacturing site of an industrial partner. The real AMR chosen for this trial was a MiR200 from Mobile Industrial Robots ApS which

features two laser scanners to provide a 360° protection field and LiDAR data for navigation, and a structured-light infrared camera to detect overhead obstacles. Figure 6 shows the screenshots from the AMR user interface and photographs from testing. In testing, it was found that the real AMR exhibited five of the eight characteristic behaviours of an ideal AMR. The following deviations were found:

Behaviour 4: The real AMR did not take account of obstacles (pedestrians) approaching from behind unless this entered the safety field of the laser scanners which triggered an emergency stop. This can be seen in Fig. 6 (behaviour 4), where planned path of the real AMR is unchanged despite being closely followed by a pedestrian approaching from behind.

Behaviour 7: The real AMR did not reverse when encountering an on-coming obstacle. However, as shown in Fig. 6 (behaviour 7) the real AMR planned a route to travel back along the path in order to afford room to manoeuvre. This appears to be a result of replanning a route given a snapshot of the current environment without consideration for the obstacle's trajectory.

Behaviour 8: In this test, an obstacle was placed in the path of the real AMR, causing it to replan a route to overtake. However, as the real AMR began to overtake, this obstacle was removed. From the screenshot shown in Fig. 6 (behaviour 8), the real AMR

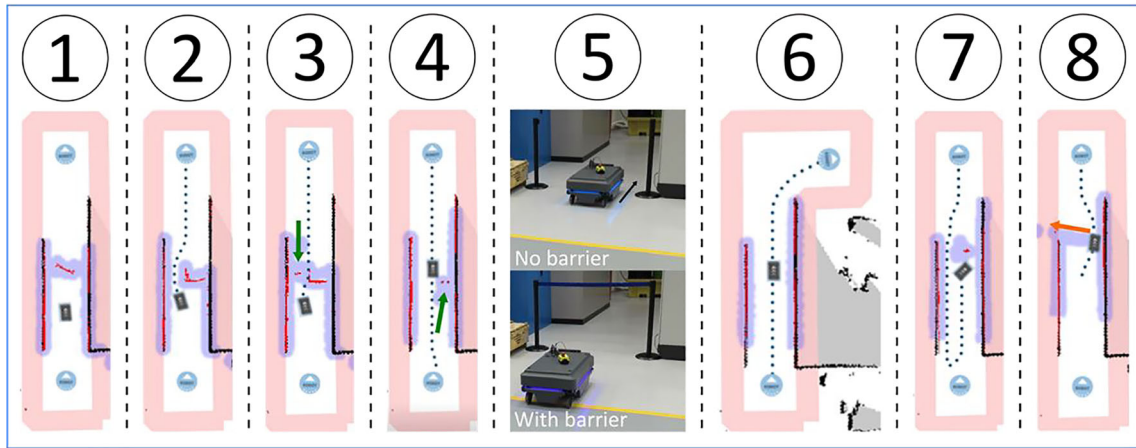


Fig. 6 Screenshots of characteristic behaviour of real AMR from the AMR user interface. Areas in pink represent forbidden region bounding the test area, black represents walls in the map, red represents objects

detected via LiDAR, and purple represents the identified obstacles. Green and orange arrows represent the movement of pedestrians and obstacles, respectively

registered the region where the obstacle had been, despite laser data confirming that the obstacle is no longer present. This can be attributed to a practical measure to reduce the computation overhead in navigation, whereby a new path is only generated when the current path is blocked. As the path shown in Fig. 6 (behaviour 8) is still valid as the real AMR executes this, even though it is not optimal. The deviation in behaviour 8 between the real AMR and ideal AMR only resulted in a lag between the external environment and the internal map; this lag was neglected and behaviour 8 was included.

These observations show a clear departure from those ideal behaviours laid out in Section 6. However, this may be attributed to a practical need for the AMR to prioritise the execution of the mission rather than overall traffic throughput as it cannot distinguish pedestrians or vehicles, and static obstacles or understand their intentions. As such, commercially available installations frequently defer to a fleet manager agent which can be afford a holistic view of the environment by aggregating the sensor data from multiple AMRs or external sensors (e.g. RFID-tracking, vision systems).

7 Simulation model for real AMR

Based on the results of this testing, the simulation model for real AMR was created by modifying that of the ideal AMR by removing behaviours 4 and 7. The model for real AMR was only a modification of simulation model

for ideal AMR and therefore regression testing was sufficient for verification [23]. The model was validated, by recording feedback from industrial partners and end-users of AMR, where the model secured a mean score of ‘4’ (on a scale from 1 to 5) across the eight scenarios. Multiple scenarios that could potentially prohibit the implementation of real AMR (details in Section 7), were identified from the shop floor of the industrial partner. These scenarios were executed in the models to depict the behaviour of real AMR and were analysed to identify any favourable and/or adverse characteristics which may impact the shop floor activities. This demonstrates the value of real AMR simulation studies where adverse behaviour can be identified and then mitigated, either through improvements on the real AMR or through establishing shop floor protocols that reduces the potential impact of these behaviours. Such improvements can then be further simulated and tested.

8 Behaviour comparison of ideal AMR and real AMR

Five scenarios identified from the shop floor of an industrial partner, which could potentially prohibiting the implementation of real AMR: simultaneous overtaking, blocked overtaking, completely blocked path, dynamic obstacles obscured by a corner, and multiple dynamic obstacles. These five scenarios were implemented in simulation models of ideal AMR and real AMR and compared in order to understand the implications of implementing the proposed Highway Code. The finding of these simulations are listed as follows:

1. Blocked path

Scenario: The AMR travels along the vehicle path; a large load that obstructs the vehicle path, and the pedestrian path moves towards the AMR. Fig. 7 depicts the videos for Blocked Path scenario for both real AMR and ideal AMR.

Behaviour of real AMR

- When the load reaches the AMR, the AMR stops.
- The load cannot get around the AMR and therefore the load has to reverse all the way back up the path until it reaches a wider area.
- The AMR approaches the load and overtakes it.
- The load continues its planned route.

Discussion: Due to lack of behaviour 7, the load is hindered by the real AMR. The load has to reverse until it reaches a wider area resulting in delays on the shop floor.

Fig. 7 (a) Video showing Blocked Path for real-AMR (b) and ideal AMR. The red lines, yellow blocks and blue circle represent the planned path of AMR, obstacles and the destination respectively.

Behaviour of ideal AMR

- When the load reaches the AMR, the AMR stops.
- As the load continues to move towards the AMR, it begins to reverse.
- The AMR continue to reverse until either the load or it can get out of the way.

2. Blocked overtaking

Scenario: The AMR travels along the vehicle path; a faster truck is approaching it from behind. Fig. 8 shows the videos for this scenario.

Behaviour of real AMR

- AMR does not realise that an obstacle has approached it from behind and continues to follow its path.
- The truck tries to overtake the AMR but realises that there is an obstacle in the pedestrian path and therefore moves in behind the AMR again.
- The truck follows the AMR until it can either overtake the AMR, or reach its destination.

Discussion: Due to the lack of behaviour 4, the truck which was initially moving faster than the AMR is hindered by the real AMR, resulting in delays on the shop floor. The faster moving truck makes several attempts to overtake the AMR; this could result in collisions.

Fig. 8(a) Video showing Blocked Overtaking by real-AMR (b) and ideal AMR. The red lines, yellow blocks and blue circle represent the planned path of AMR, obstacles and the destination respectively.

Behaviour of ideal AMR

- AMR realises that an obstacle is approaching from behind and stops to allow it to overtake.
- The truck attempts to overtake the AMR and because it has stopped moving, there is enough time for the truck to overtake.

3. Simultaneous overtaking

Scenario: The AMR travels along the vehicle path; a faster truck begins to travel in the same direction along the vehicle path; the faster truck is obstructed by the AMR. The AMR is also obstructed by a static obstacle at the front. As the AMR is overtaking the static obstacle, a pedestrian approaches it from the front. Fig. 9 depicts the videos for this scenario for both the real AMR and ideal AMR.

Behaviour of real AMR

- AMR does not realise that an obstacle has approached it from behind and begins to overtake the static obstacle.
- The truck sees the static obstacle and begins to overtake it as well.
- As the AMR is overtaking the static obstacle, a pedestrian approaches it from the front.
- AMR identifies the pedestrian and stops.
- The truck has to stop.
- The pedestrian has to move out of the way of the AMR and wait for both the AMR and the dynamic obstacle to re-join the vehicle path.

Discussion: Due to lack of behaviours 4 and 7, both the dynamic obstacle and the pedestrian are hindered by the real AMR, which may result in collisions or time being wasted on the shop floor.

Fig. 9 (a) Video showing Simultaneous Overtaking by real-AMR and (b) ideal AMR. The red lines, yellow blocks and blue circle represent the planned path of AMR, obstacles and the destination respectively.

Behaviour of ideal AMR

- AMR realises that an obstacle is approaching from behind and stops to allow it to overtake.
- The faster truck overtakes both the AMR and the static obstacle.
- Ideal AMR begins to overtake the static obstacle.
- As it is overtaking the static obstacle, a pedestrian approaches it from the front.
- The pedestrian continues walking towards it, so it reverses out of the way.
- Once the pedestrian is out of way, ideal AMR overtakes the static obstacle and re-joins the vehicle path.

4. Dynamic obstacles obscured by a corner

Scenario: The AMR travels along the vehicle path and moves towards a blind corner; a truck approaches the same corner from the other direction; the truck takes the corner slightly quicker than it should do just as the AMR reaches the corner (Fig. 10)

Behaviour of real AMR

- As the truck moves quickly towards real AMR, the real AMR stops;
- The truck cannot stop in time and collides with the AMR.

Behaviour of an ideal AMR

- As the truck moves quickly towards ideal AMR, the AMR stops;
- The truck cannot stop in time and continues moving towards the AMR;
- AMR realises that the truck is getting too close and reverses out of the way.

Discussion: Due to lack of behaviour 7, both the truck and the real AMR could potentially be damaged if the truck is not able to stop before collision.

Fig. 10 (a) Video showing Dynamic Obstacles Obscured by a Corner for real-AMR (b) and ideal AMR. The red lines, yellow blocks and blue circle represent the planned path of AMR, obstacles and the destination respectively.

5. Multiple dynamic obstacles

Scenario: The AMR travels along the vehicle path; multiple trucks approach the AMR along the vehicle path in the opposite direction; when the trucks reaches the AMR, no overtaking is possible due to obstacles obscuring the pedestrian path.

Behaviour of real AMR

- It stops;
- The trucks are stuck waiting for the pedestrian path to be clear or have to reverse back to allow the AMR to overtake them.

Behaviour of ideal AMR

- It stops;
- As the trucks continue moving towards it, it begins to reverse;
- The trucks proceed until either the AMR can overtake them or until the AMR has reversed sufficiently.

Discussion: The downfall of the real AMR here is a lack of behaviour 7. This scenario could perhaps be the most common scenario on the shop floor, especially during times of high pedestrian density such as during shift changeovers, when the pedestrian path will be inaccessible for overtaking. In this scenario, the truck(s) is hindered by the real AMR and again, the delay could potentially be costly to the manufacturing process. Fig. 11 depicts the videos for this scenario for both the AMRs.

Fig 11 (a) Video for Multiple Dynamic Obstacles scenario for real-AMR and (b) for ideal AMR. The red lines, yellow blocks and blue circle represent the planned path of AMR, obstacles and the destination respectively.

The comparison of behaviours of real AMR and ideal AMR in five scenarios from the shop floor identified potential problems before implementation of AMR on the shop floor. The comparison also highlighted that the absence of behaviours 4 and 7 can result in delay and collisions on the shop floor which could potentially be costly to the manufacturing process. This comparison could enable a manufacturing end user to make informed decisions regarding the implementation of the AMR on their shop floor.

9 Conclusions and future work

Efficient and safe movement of AMRs in a dynamic and flexible manufacturing environment may cause challenges due to

the lack of understanding of rules related to AMR movement. This motivated this research work, leading to the development of a Highway Code for AMRs, and simulation, physical testing, and comparison of behaviours of real and ideal AMRs. This study demonstrated the use of AMR simulation studies to identify adverse behaviour hence allowing their mitigation either through improvements on the real AMR or through establishing shop floor protocols that reduce the potential impact of these behaviours. Such improvements can then be further simulated and tested.

The methodology presented in this paper is currently being used to develop and test other parts of the Highway Code. In non-linear systems optimal input depends on system parameters to be identified; therefore, future work would include detailed experimentation by applying optimal experiment

design for output error (OE) models as discussed in reference [25]. We are not yet in position to conduct the necessary full factorial testing to investigate the impact of input noise, outliers, and output constraints in our real world non-linear system. Our study is exploratory and scenario-based, intended to prove the feasibility and to justify the high cost of future large-scale experiments.

Acknowledgements This project was supported by the Royal Academy of Engineering under the Research Chairs and Senior Research Fellowships scheme. The authors would also like to acknowledge Airbus for their support.

Data availability The underlying data can be accessed at <https://doi.org/10.15131/shef.data.7713116>.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Siegwart R (2011) Introduction to autonomous mobile robots. <https://doi.org/10.1017/CBO9781107415324.004>
- AGV vs. AMR - What's the Difference? <http://www.mobile-industrial-robots.com/en/resources/whitepapers/agv-vs-amr-whats-the-difference/>. Accessed 12 February 2019
- Lasi H, Fettke P, Kemper HG, Feld T, Hoffmann M (2014) Industry 4.0. *Bus Inf Syst Eng* 6:239–242. <https://doi.org/10.1007/s12599-014-0334-4>
- Park JH, Huh UY (2016) Path planning for autonomous mobile robot based on safe space. *J Electr Eng Technol* 11:1921–1718. <https://doi.org/10.5370/JEET.2016.11.3.1921>
- Corréa AI, Langevin A, Rousseau LM (2007) Scheduling and routing of automated guided vehicles: a hybrid approach. *Comput Oper Res* 34:1688–1707. <https://doi.org/10.1016/j.cor.2005.07.004>
- Anavatti SG, Francis SL, Garratt M (2015) Path-planning modules for autonomous vehicles: current status and challenges. *IEEE Int Conf Adv Mech, Int Man, Ind Aut(ICAMIMIA)*:205–214
- Fazlollahtabar H, Saidi-Mehrabad M (2015) Methodologies to optimize automated guided vehicle scheduling and routing problems: a review study. *J Intell Robot Syst Theory Appl* 77:525–545. <https://doi.org/10.1007/s10846-013-0003-8>
- Li MP, Kuhl M (2017) Design and simulation analysis of PDER: a multiple load AGV dispatching algorithm, Proc. 2017 Winter Simul. Conf
- Azimi P, Haleh H, Alidoost M (2010) The selection of the best control rule for a multiple-load AGV system using simulation and fuzzy MADM in a flexible manufacturing system. *Model Simul Eng* 2010:821701–821712. <https://doi.org/10.1155/2010/821701>
- Awad F, Naserllah M, Omar A, Abu-Hantash A, Al-Taj A (2018) Collaborative indoor access point localization using autonomous mobile robot swarm. *Sensors (Basel)* 18(2). <https://doi.org/10.3390/s18020407>
- Nedić N, Dragan P, Cristiano F, Stojanović V, Pavlovic A (2017) Simulation of hydraulic check valve for forestry equipment. *Int J Heavy Veh Syst* 24:3. <https://doi.org/10.1504/IJHVS.2017.084875>
- Li B, Liu H, Xiao D, Yu G, Zhang Y (2017) Centralized and optimal motion planning for large-scale AGV systems: a generic approach. *Adv Eng Softw* 106:33–46. <https://doi.org/10.1016/j.advengsoft.2017.01.002>
- Paden B, Čáp M, Yong S, Yershov D, Frazzoli E (2016) A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Trans Intell Transp Syst* 1(1):33–55. <https://doi.org/10.1109/TIV.2016.2578706>
- Hedborg O (2017) The development of the AGV Highway Code, Dissertation, Cranfield University
- Rohmer E, Singh SPN, Freese M (2013) V-REP: a versatile and scalable robot simulation framework. *IEEE Int Conf Intell Robot Syst*:1321–1326. <https://doi.org/10.1109/IROS.2013.6696520>
- The Open Motion Planning Library. <http://ompl.kavrakilab.org/>. Accessed 12 February 2019
- Olmi R (2011) Traffic management of automated guided vehicles in flexible manufacturing systems, Dissertation, University of Ferrara
- Bourbakis NG (1997) A traffic priority language for collision-free navigation of autonomous mobile robots in dynamic environments. *IEEE Trans Syst Man Cybern B Cybern* 27:573–587. <https://doi.org/10.1109/3477.604097>
- Egbelu PJ, Tanchoco JMA (1984) Characterization of automatic guided vehicle dispatching rules. *Int J Prod Res* 22:359–374. <https://doi.org/10.1080/00207548408942459>
- Ho YC, Chien SH (2006) A simulation study on the performance of task-determination rules and delivery-dispatching rules for multiple-load AGVs. *Int J Prod Res* 44:4193–4222. <https://doi.org/10.1080/00207540500442401>
- Lee J, Tangjarukij M, Zhu Z (1996) Load selection of automated guided vehicles in flexible manufacturing systems. *Int J Prod Res* 34:3383–3400. <https://doi.org/10.1080/00207549608905096>
- Umar UA, Ariffin MKA, Ismail N, Tang SH (2015) Hybrid multiobjective genetic algorithms for integrated dynamic scheduling and routing of jobs and automated-guided vehicle in flexible manufacturing systems environment. *Int J Adv Manuf Technol* 81: 2123–2141. <https://doi.org/10.1007/s00170-015-7329-2>
- Nidhra S, Dondeti J (2012) Black box and white box testing techniques. *Int J Embed Syst Appl* (2):29–50. <https://doi.org/10.1016/j.profnurs.2010.10.006>
- Taylor A (2018) Simulation of the movement of autonomous mobile robots around a shop-floor, Dissertation, The University of Sheffield
- Stojanovic V, Filipovic V (2014) Adaptive input design for identification of output error model with constrained output. *Circ Syst Signal Process* 33(1):97–113. <https://doi.org/10.1007/s00034-013-9633-0>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.