https://doi.org/10.3311/PPtr.36908

Periodica Polytechnica Transportation Engineering, 52(4), pp. 412–422, 2024

Designing a Trajectory Control System for Wheelchairs with a Combination of Automatic and Manual Steering Methods

Van Tan Vu[1*], Quang Thinh Nguyen[2], Chihiro Nakagawa[3], Trung Nguyen Nguyen[1], Thanh Long Le[1], Van Vinh Nguyen[1], Thi Lan Anh Hoang[4], Trong Tu Do[5], András Mihály[6], Péter Gáspár[6]

[1] Department of Automotive Mechanical Engineering, Faculty of Mechanical Engineering, University of Transport and Communications, Room 603, Building A6, no. 3, Cau Giay Street, Lang Thuong Ward, Dong Da District, 100000 Hanoi, Vietnam
[2] Japan Technology Solution Joint Stock Company, 9 Dau Ma Street, Section 2, Thi Cau Ward, 222000 Bac Ninh, Bac Ninh, Vietnam
[3] School of Engineering, Osaka Metropolitan University, 1–1 Gakuen-cho, Nakaku, Sakai, 599–8531 Osaka, Japan
[4] Emergency abdominal surgery Department, 4th floor, Building D, Viet Duc University Hospital, no. 40, Trang Thi Street, 100000 Hanoi, Vietnam
[5] Mechanical and Power Engineering Faculty, Room 304, Building M, Electric Power University, no. 235, Hoang Quoc Viet Street, Co Nhue Ward, Bac Tu Liem District, 100000 Hanoi, Vietnam
[6] Systems and Control Laboratory, Institute for Computer Science and Control, Hungarian Research Network, Kende u. 13–17, 1111 Budapest, Hungary
[*] Corresponding author, e-mail: vvtan@utc.edu.vn

Received: 18 March 2024, Accepted: 27 August 2024, Published online: 10 September 2024

Abstract

A wheelchair is a specialized form of Personal Mobility Vehicle (PMV) designed to help people with disabilities move safely to their desired locations. Unlike conventional PMVs, wheelchairs for people with disabilities are challenging to manually control in environments with heavy traffic. This study focuses on designing and feasibility testing of control systems for PMVs to meet safety and efficiency standards, even in complex environments. First, a control structure model for PMVs is proposed. This is followed by the application of a combination of First In First Out (FIFO) and Dijkstra traffic control methods to determine effective trajectories for PMVs. The proposed control system is then tested on the campus of the University of Transport and Communications. Simulation results show that when comparing the travel time of 10 PMVs, the order execution time for each PMV remains stable (within 10%) for up to 20 PMVs. However, when the number of PMVs increases to 25, 30, 35, and 40 vehicles, the order execution time increases by 35%, 55%, 103%, and 119%, respectively. These results demonstrate that the system can control a large number of PMVs, ensuring safety, flexibility, and efficiency.

Keywords

personal mobility vehicle, wheelchair, trajectory control system, intersection control method, Dijkstra method, hybrid control model, traffic control method

1 Introduction

People with disabilities are an inseparable part of modern societies. It results from the interaction between health conditions such as dementia, blindness or spinal cord injury and a range of environmental and personal factors. According to the World Health Organization, by 2023, an estimated 1.3 billion people – or 16% of the global population – had a severe disability. This number is increasing due to the rise in non-communicable diseases and people living longer. People with disabilities are a diverse group and factors such as gender, age, religion, race, ethnicity, and economic circumstances influence their life experiences and health needs (WHO, 2023).

According to statistics of the National Committee for People with Disabilities of Vietnam, the country has about 6.2 million people with disabilities, accounting for 7.06% of the population aged 2 years or older, of which 58% are female, 28.3% are children, nearly 29% are severely and especially severely disabled (Công, 2020). Because their mode of transportation is mainly handing wheelchairs or having to have a pusher to be able to move, it is still difficult to travel in crowded places. Recognizing that situation, many studies have been launched to find solutions to help people with disabilities move themselves to their desired place automatically. One of the most feasible

Cite this article as: Vu, V. T., Nguyen, Q. T., Nakagawa, C., Nguyen, T. N., Le, T. L., Nguyen, V. V., Hoang, T. L. A., Do, T. T., Mihály, A., Gáspár, P. (2024) "Designing a Trajectory Control System for Wheelchairs with a Combination of Automatic and Manual Steering Methods", Periodica Polytechnica Transportation Engineering, 52(4), pp. 412–422. https://doi.org/10.3311/PPtr.36908

measures is to turn a normal electric wheelchair into an autonomous electric wheelchair. Thus, a wheelchair is not only an assistive tool but also a PMV (Personal Mobility Vehicle) for people with disabilities to help them smoothly move to their desired position. Therefore, in this article, the authors use the phrase "PMV" instead of the phrase "automatic electric wheelchair".

There have been many studies on smart electric wheelchairs in the world, such as PerMMA Gen II wheelchair (Wang et al., 2013), MEBot (Daveler et al., 2015), Ibot (Cooper et al., 2006), etc. However, these studies only focus on free control on each device and not really on controlling PMVs at the system level, which not only creates a potential accident risk but also reduces the vehicle's ability to flow in crowded areas. In a previous research (Tan et al., 2022) the authors proposed a centralized control model to improve flexibility and safety when PMVs move in different environments.

Unlike other common PMVs, wheelchairs for people with disabilities often move around specific buildings or facilities such as hospitals, schools, shopping malls, etc. Therefore, these wheelchairs cannot use satellite mapping tools such as Google Maps and require a system that provides detailed maps of buildings and facilities, as well as navigation, so that PMVs can move comfortably without causing accidents or congestion.

This study aims to design and control a PMV trajectory system for people with disabilities to meet safety and efficiency standards. The research team focused on the following contributions:

- Propose a model of the control structure required for PMVs (Section 2).
- Propose and test simple and safe traffic control methods for PMVs in complex traffic environments (Section 3)
- Suggest the selection and application of pathfinding algorithms for PMVs in cases where the vehicle is controlled in a fully automatic state (Section 3).
- Simulate and evaluate proposals based on theoretical design using the map of the University of Transport and Communications, Hanoi, Vietnam (Section 4).

The subject of the research applies to people with disabilities, so the PMVs must be controlled automatically by a system to ensure safety, efficiency, and smoothness. The above parameters are evaluated by many factors: collision safety, smoothness when accelerating or braking, and travel time efficiency. In the study, the proposed model is applied to one or more vehicles that can move fully autonomously within the campus of the University of Transport and Communications. The automatic control system is based on the Dijkstra shortest path finding algorithm and the First-In-First-Out (FIFO) traffic control algorithm.

The structure of the article is as follows: Section 2 summarizes the methods of controlling the motion trajectory of PMVs; Section 3 describes the design of the PMV's motion trajectory control system; Section 4 shows the results of the evaluation of the designed system in a specific environment; Section 5 gives conclusions and future research directions.

2 Control methods for PMVs

2.1 Traffic control methods

In many studies of transportation networks, vehicular traffic has been assumed to follow the so-called First-In-First-Out (FIFO) principle. Different from the queueing models of traffic flow, many other well-studied traffic flow models usually assume implicit link FIFO. Let t represent the time for vehicle n to travel from home x_1 to destination x_2. This is given in Eq. (1) (Jin and Li 2007):

$$t(n;\ x_1,\ x_2) = t(n,\ x_2) - t(n,\ x_1) \tag{1}$$

Where:

- $t(n; x_1, x_2)$ is the travel time for vehicle n from x_1 to x_2.
- $t(n; x_1)$ is the time when vehicle n reaches location x_1.
- $t(n; x_2)$ is the time when vehicle n reaches location x_2.

The FIFO principle is violated when the order in which vehicle n passes home x_1 and destination x_2 is different, or when the IDs of the z_j vehicles passing positions x_1 and x_2 are different.

To improve safety of PMVs when moving through intersections, the authors use the classic FIFO traffic control method at the intersections, specifically as follows:

- Step 1: All PMVs must continuously report the current location to the system when moving to the new location.
- Step 2: All PMVs must wait for the control signal when reaching the specified intersection position.
- Step 3: PMVs that receive the signal allowed first will be moved first, PMVs coming later will have to stop at the specified position. This ensures that no collisions will occur even if the PMV comes to a complete stop.
- Step 4: When the first-arrived PMV has exited the intersection area, the next vehicle will receive the same control signal according to the FIFO principle.

This method is a traditional method applied to controlling basic mobile robot systems. The disadvantage is that the vehicle will have to stop in case many others are waiting at the same intersection, but it is absolutely safe to avoid dangerous collisions. Therefore, it is suitable for small-scale systems such as PMVs.

2.2 Dijkstra algorithm

The Dijkstra algorithm was invented by the Dutch computer scientist Edsger Dijkstra in 1956, it is an algorithm that solves the problem of finding the shortest path from one vertex to the rest of the directed and scalar graph as long as the edges are weighted non-negative (Wikipedia, 2024). The algorithm is commonly applied in global positioning system (GPS) technology (Mirzaeinia et al., 2019; Zhang and Zhao, 2014), mobile robot control systems (Julius Fusic et al., 2018), and navigation systems for taxis (Lanning et al., 2014; Ruan et al., 2014).

The Dijkstra algorithm can find all the optimal paths and the correct rate of these optimal paths is 100%, but the search speed is slow, and the time-consuming efficiency is low in multi-node screening. The distance of all the marked points *k* to the unmarked points *j* directly connected to the marked points is calculated, and the function is set in Eq. (2). In the function expression, it is the direct connection distance from point *k* to *j* (Zhou and Gao, 2019).

$$w_j = \min\left\{w_j,\ w_k + d_{kj}\right\} \tag{2}$$

Where:

- w_j is the current known shortest distance from the root node to node *j*.
- w_k is the current known shortest distance from the root node to node *k*.
- d_{kj} is the direct distance from node *k* to node *j*.

The Dijkstra algorithm can solve the problem of finding the shortest path both on scalar and directional graphs as long as the weight is not negative. The basic idea of the algorithm is as follows:

- Step 1: From the root top, initialize the distance to itself is zero, initialize the distance to the remaining nodes is "infinite".
- Step 2: Let S = {} be a collection of nodes consisting of current_node (the node being considered) and passed_node (the node being considered). The first current_node is the root peak of the problem. Cost(N) is the value of the shortest path from N to the root vertex.
- Step 3: Consider nodes adjacent to N with current_node. Let d(current node,N) be the distance between the adjacent node N and current_node. For p = d(current node,N) + cost(current_node). If p < cost(N) then cost(N) = p. Otherwise, cost(N) remains the same.
- Step 4: After reviewing all nodes adjacent to N, mark current_node as passed_node.
- Step 5: Find a new current_node with 2 conditions: not passed_node and cost(current_node) is the minimum.
- Step 6: If the set S = {} contains enough nodes of the graph, stop the algorithm. If not, go back to Step 3 (Lanning et al., 2014).

The Dijkstra algorithm has advantages such as little complexity, almost linear; the algorithm only works for directional, weighted graphs and all edges must have non-negative values. However, the algorithm also has disadvantages such as the algorithm takes longer to process than other algorithms such as (DFS, A*, etc.) but always results in the shortest route; the algorithm cannot handle negatively weighted edges.

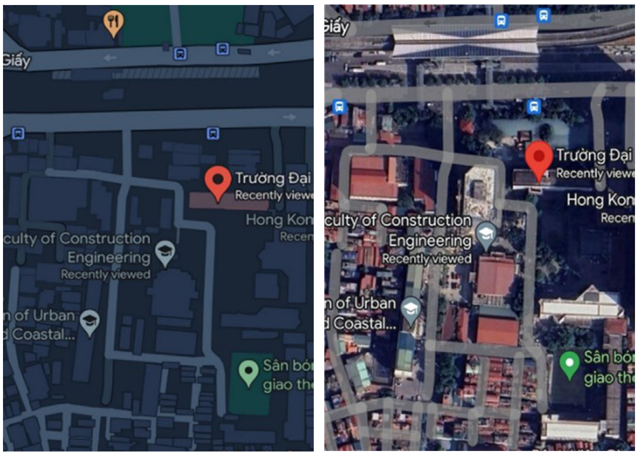
The Dijkstra algorithm can be applied in the following typical techniques: For mapping applications, the algorithm is widely deployed to measure the shortest possible distance and check the direction between two geographical areas such as Google Maps; for telephone networks, the algorithm is also widely deployed in conducting data in the network and telecommunications sectors to reduce obstacles to transmission; wherever the need for shortest path interpretation is solved in the field of robotics, transport, embedded systems, laboratories or manufacturing plants, etc., this algorithm can be applied (Swaniawski, 2022; Tyagi, 2020).

3 Designing the trajectory control system for wheelchairs and data collection

3.1 Design of trajectory control systems for wheelchairs

In this study, the authors apply the design of a wheelchair trajectory control system to a specific case on a campus of the University of Transport and Communications shown in Fig. 1(a). The main design steps are as follows:

- Step 1: Extract the University of Transport and Communications map from Google Map, as shown in Fig. 1(b).
- Step 2: Create a layout for the map of the University of Transport and Communications.
- Step 3: Divide the road grid on the layout with a distance of 1 m.



(a) (b)

Fig. 1 Map of university of transport and communications: (a) in road map mode; (b) in satellite observation mode (Google Map, 2024)

- Step 4: Create points, divide lanes.
  - The Law on Road Traffic in Vietnam stipulates people and road vehicles drive on the right side of the road. So, in this experiment, the PMVs drive on the right side.
  - According to the standards of roads and practice at the University of Transport and Communications, we divide the roads into two lanes because the width of one PMV is 554 mm and the roads at the University of Transport and Communications has a width of higher than 2 m.
  - Set each meter distance as one point on the map.
- Step 5: Design the path.

The road design ensures the following principles: One-way roads for one direction; two-way roads are as restrictive as possible; the two-way road also divides into two one-way directions; simplified intersections avoid duplication more than twice; the points ensure the vehicles do not collide with each other.

During system control, the following priorities are needed: If the system receives a first-back signal of two PMVs when reaching an intersection, priority is given to the PMV that sends the signal first. This ensures time efficiency when moving. If the system simultaneously receives signals from two PMVs when reaching an intersection, priority is always given to PMVs moving in the front lane, and then to PMVs crossing the road.

Fig. 2 shows the result of building control data based on the above steps.

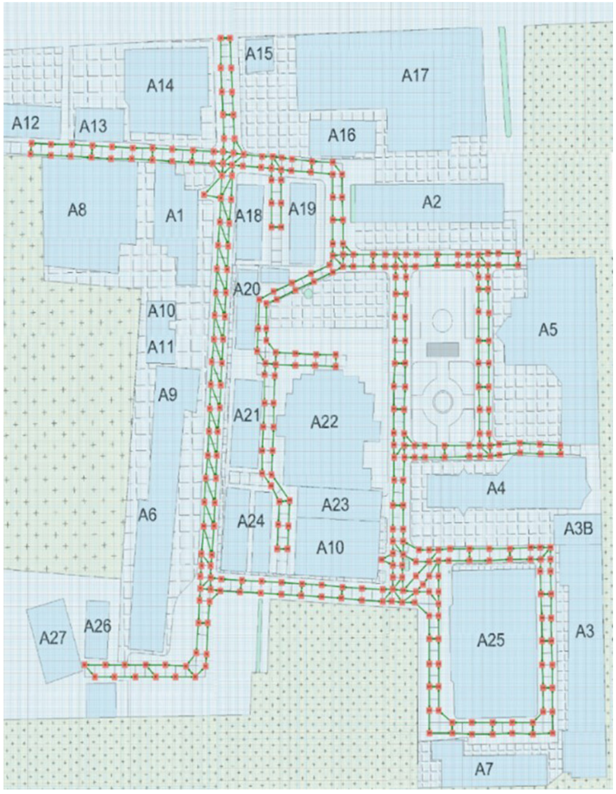


Fig. 2 PMV control map at UTC

3.2 Design and selection of safety devices on PMVs

The equipment of a typical PMV includes Lidar, Radar, Ultrasonic sensors.

Lidar (Light Detection and Ranging) is a device used to detect and measure the distance to surrounding objects and it can build a 3D map. The principle of operation of Lidar is that it emits light beams at objects and picks up reflected light. With this principle, it is possible to measure the distance of all surrounding objects by calculating the time between the emission and collection of light rays. Lidar is usually combined with other devices to serve different uses, for example with GPS to serve on self-driving vehicles. In self-driving vehicles, this device is mounted on a cylindrical housing capable of rotating 360° placed on the roof of the vehicle. Active Lidars will get 3D map of the environment and locations on the road with a 360° rotation angle. They use lasers, ultraviolet light, visible light, or infrared light to photograph the subject. This data is fed into a computer that creates a 3D map of the surroundings with high accuracy. The accuracy of this map is in centimeters because the wavelength of light used is very small and can reflect all types of surfaces, even small objects.

Radar (Radio Detection and Ranging): This device determines the relative velocity between an object and a vehicle using electromagnetic waves. During measurement, it will emit a signal and wait for the signal to respond backwards. Compared to Lidar, Radar uses longer waves and lower signal energy. However, it cannot describe the shape of the scanned space. The feedback signals may be problematic for objects that are not metal or have a special shape.

Ultrasonic sensors: These are mounted on different sides of the vehicle to detect objects very close to the vehicle or used to determine the distance to other vehicles during parking. These sensors provide parking assist, collision warning, lane departure warning, and other functions.

Fig. 3 depicts the design of Lidar sensor control models widely used in the field of robotics including PMVs. In this sensor control design, users can define about 30 control models for a safety sensor, with three levels of early warning (blue), near warning (orange), stop danger warning (red). Based on such a safe control principle, the system recognizes the PMV's position on the map, commands ON/OFF control and controls sensor samples from those 30 control samples (Swaniawski, 2022). The locations on the control map of the system will be preset in the system.

3.3 Map design

The map for PMV trajectory control design includes a total of 378 nodes, clear directions of movement, 21 stations, 2 waiting stops (Home), 65 traffic control nodes (Intersection) including 11 important intersections, all of which are detailed in Fig. 4. The design data is used to test the safety of the system's traffic control. The authors designed a total of 65 large and small traffic control nodes. In this article, the authors would like to present the inspection results of 11 major intersections as shown in Fig. 4.

Fig. 5 shows traffic control button no.1. Red color represents potential collision locations that need to be controlled. Orange color represents the gateway locations of the intersection as well as the vehicle location and communication system to decide whether to enter the intersection or not.

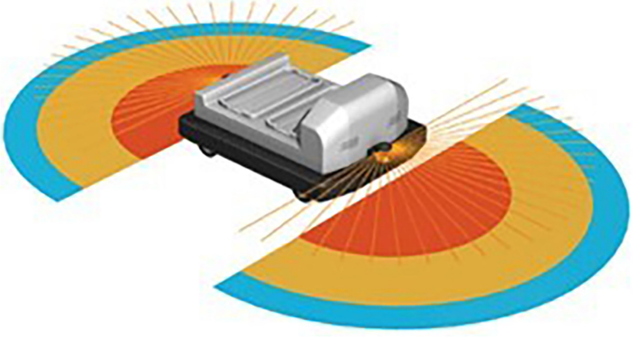


Fig. 3 Lidar and sensors on a PMV (Hokuyo, 2024)

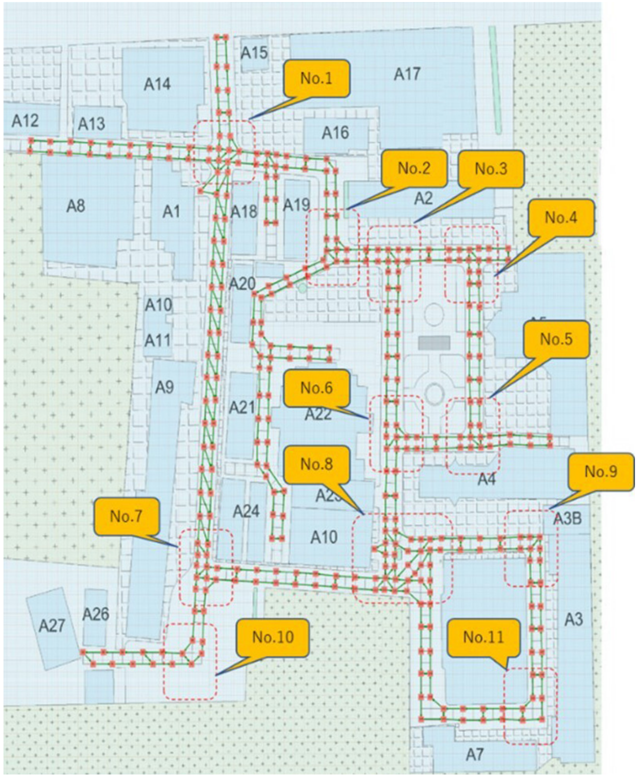


Fig. 4 Important traffic control nodes layout

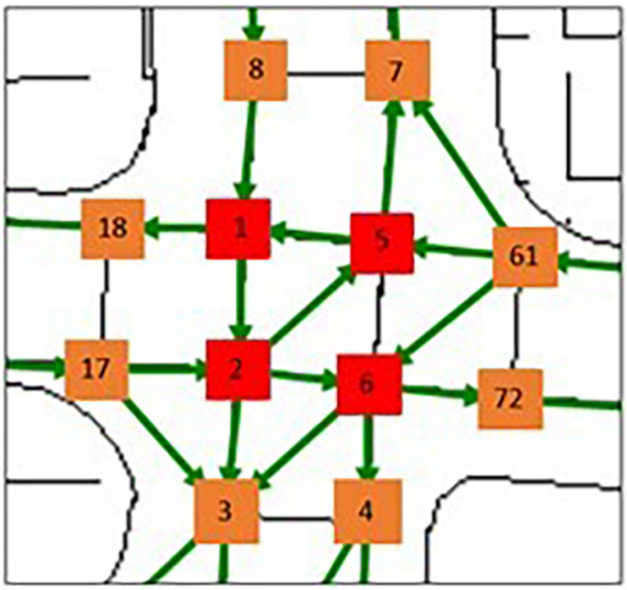


Fig. 5 Example for traffic control no.1

This intersection has many different directions. In terms of technology, they can be divided for control, but to ensure the safety of the system and simplify the control design, here we choose the option of a large intersection. The original assumption was that all PMVs move quickly through the intersection, neither waiting nor deliberately stopping inside (the red spots) of the intersection. The remaining intersections are also designed similarly to no.1.

3.4 PMV trajectory control system and simulation tools

PMV control system is a management software that synthesizes all mobile devices such as PMVs to ensure safety with traffic control features, improve user efficiency with the shortest path-finding feature, stable maneuverability feature with optimal speed control feature. The PMV control system as well as the control simulation tools below are provided by Japan Technology Solution Joint Stock Company (JTS).

The PMV trajectory control system has the list of system control simulation tools as follows:

Database: a database platform to store control design data and daily execution data to support control monitoring, ensuring good system operation and timely data retrieval in case of abnormal errors. Databases are also an important part of managing one and more PMVs in the future and are part of IoT (Internet of Things) technology in the transportation sector.

PMV Simulator: a software that simulates all basic operations of PMV equipment in reality such as communication with the system, moving/stopping and energy consumption, charging operations, etc.

PMV Map designer: a software that allows users to design and create Map data to control PMVs intuitively and easily with features such as creating points, creating paths, creating commands, creating intersections, defining points on the map, etc.

PMV order manager: a software that commands PMVs to go to desired locations.

PMV Monitor: a software that simulates the position of PMVs on the designed Map platform, indicating the location, time, logs controlling the actual communication between PMVs and the system.

4 Simulation and evaluation

Two methods are available to evaluate whether the designed system works: 1) Use the actual PMVs for commissioning and testing; 2) Use simulation tools for testing. Due to the difficulty in using actual equipment for testing, the authors use simulation methods to test the system with specific test items:

- Feasibility of the system: This characteristic is the most important evaluation criterion showing whether the system can control PMVs or not. Here, the authors will check whether PMVs in all preset locations can reach any other location with the shortest path detected by the system using the Dijkstra wayfinding algorithm.
- Safety of the system: This characteristic is the second most important evaluation criterion showing that the system is capable of controlling multiple PMVs at the same time, without colliding at any intersection. It is necessary to use FIFO control to check that at each intersection when one PMV enters the interior area, the other PMVs will stop at the boundary positions, and when the PMV exits the intersection, the other PMVs will pass through the intersection in turn.
- System flexibility: This characteristic is the third most important evaluation criterion demonstrating that the system is capable of controlling a large number of PMV vehicles with high efficiency. To evaluate the flexibility of the system in detail, it is necessary to carefully prepare and install the system. Therefore, in order to simplify and save time, the authors stopped at the measurement step to survey the controllability (time, shortest distance) by simulating the same control map, the same number of PMV vehicles, the same system and simulation tools.

4.1 Testing the feasibility of the designed system

To test the feasibility of the designed system, all stations that can be walked with the shortest path have been examined as shown in Fig. 6 and summarized in Table 1. Table 1 summarizes the points that the PMV has passed through, which have been marked on the map. The first column is the serial number, the second column is the starting point, the third column is the destination point of the PMV, and the fourth column lists the points the PMV has passed through. This data is extracted from the system and analyzed to verify if the PMVs have traveled the shortest path. If they have, "OK" is marked in the fifth column. The purpose of Table 1 is to provide a clear and concise summary of the PMV routes and to confirm whether they have followed the shortest paths as calculated

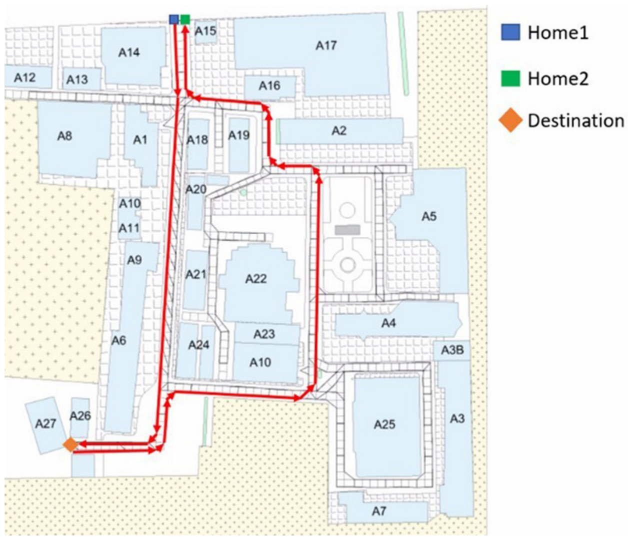


Fig. 6 The Dijkstra algorithm results in finding the shortest path

Table 1 Results of testing the path finding algorithm

No	Start position	Destination	Route data	Results (OK/NG)
0	1	9	1>2>3>4>5>7>9	OK
1	A1	A7	37>38>40>32>45>1>49>51>53>55>57>59>301>63>65>300>68>71>302>73> 304> 299>99>101>103>105>107>109>11>133	OK
2	A2	A10	268>266>264>262>263>280>226>283>285>287>289>291>293>190>191>193> 195>197>187>186>321	OK
3	A3	A2	240>297>168>170>172>295>176>178>181>182>185>196>194>192>189>188>294> 292>290>288>286>284>282>281>265	OK
4	A4	A3	207>206>204>202>200>199>306>188>191>193>195>197>187>184>183>121> 179>296>175>173>169>167>240	OK
5	A5	A6	278>277>275>274>257>272>270>268>266>264>262>260>258>256>253>251> 249>247>245>243>241>174>144>86>61>6>4	OK
6	A6	A5	300>68>71>302>73>304>299>99>101>103>105>170>109>111>112>123>125>184> 185>196>194>192>189>188>294>292	OK
7	A7	A18	142>141>138>136>134>133>131>129>126>117>114>123>125>196>194>192> 189>188>294>292>290>288>286>284	OK
8	A8	A2	30>28>26>24>21>20>17>2>6>72>95>96>177>242>244>246>248>250>252> 254 >255>308>259>261>263>265>267>269>268	OK
9	A11	A3	49>50>52>54>56>58>60>62>64>66>68>71>302>73>304>299>99>101>103> 105 >107>109>111>113>115>117>119>120>179	OK
10	A12	A7	35>36>34>43>30>28>26>24>21>20>17>2>6>72>95>96>177>242>244>246> 248>250>252>254>255>308>259>261>263>280>226>283>285>287>289>29 1> 293>190>191>193>195>197>187>186>124>122>112>115>116>130>132>135> 137>139>140	OK
11	A13	A27	29>30>29>26>24>21>20>17>3>38>40>32>45>47>49>51>53>55>57>59>301> 96> 65>300>69>70>67>74>76>78>80>82>84	OK
12	Home	A10	16>14>12>9>8>1>2>6>72>95>96>177>242>244>246>248>250>252>254>255>30 8>259>261>263>280>226>283>285>287	OK
13	A18	A5	237>238>236>234>177>242>244>246>248>250>252>254>255>308>259>261> 263>265>267>269>271>273>231>232>276	OK
14	A19	A26	238>236>234>177>144>86>61>6>4>38>40>32>45>47>49>51>53>57>59>301>63 >65>300>69>70>67>74>76>78>80>82	OK
15	A20	A4	46>48>50>52>54>56>58>60>62>64>66>68>71>302>73>304>299>99>101>103>1 05>107>109>111>112>123>125>184>185	OK
16	A10	A27	321>124>122>112>110>108>106>104>102>100>98>97>94>302>74>76>78>80>82 >84>303>88>90>92	OK
17	A22	Home	326>325>324>322>320>319>317>315>313>311>309>255>239>253>251>249> 247 >245>243>241>174>144>86>61>7>10>11	OK
18	A23	A3	343>341>339>337>335>333>329>322>320>319>317>315>313>311>309>255> 308> 259>261>263>280>226>283>285>287	OK
19	A24	A8	345>347>348>346>343>341>339>337>335>333>329>322>320>319>317>315> 313 >311>309>255>239>253>251>249>247	OK
20	A26	A23	90>92>93>91>89>87>85>83>81>79>77>75>304>299>99>101>103>105>107> 109>111>112>123>125>184>185>196>194>192>189>188>294>292>290>288> 286>284>282>281>265>262> 260>258>256>239>254>127>310>312>314>316>31 8>300>331	OK
21	A27	A7	92>93>91>89>87>85>83>81>79>77>75>304>299>99>101>103>105>107>109>111 >113>115>116>130>132>135>137>139	OK
22	A8	A10	30>28>26>24>21>20>17>2>6>72>95>96>177>242>244>246> 248>250>252>254>255>308>259>261>263>280>226>283>285	OK

Table 1 Results of testing the path finding algorithm (continued)

No	Start position	Destination	Route data	Results (OK/NG)
23	A2	A7	168>266>264>262>263>280>226>283>>285287>289>291>293>190>191>193> 195 >197>187>186>124>122>112>115>116	OK
24	A3	Home	240>297>168>170>172>295>176>178>181>182>185>196>194>192>189>188> 294> 292>290>288>286>284>282>281>265	OK
25	A6	A2	300>68>71>302>73>304>299>99>101>103>105>107>109>111>112>123>125>184> 185>196>194>192>189>188>294>292	OK
26	A5	A27	278>277>275>274>257>231>223>228>305>224>222>220>218>216>204>202> 200>199>306>188>191>193>195>197>187	OK
27	A18	A7	237>238>236>234>177>242>244>246>248>250>252>254>255>308>259>261> 263>280>226>283>285>287>289>291>293	OK
28	A2	A3	268>266>264>262>263>280>226>286>285>287>289>291>293>190>191>193> 195>197>187>184>183>121>179>296>175	OK
29	A19	A4	238>236>234>177>242>244>246>248>250>252>254>255>308>259>261>263> 265>267>269>271>273>231>223>228>305	OK
30	A7	Home	142>141>138>136>134>133>131>129>126>117>114>123>125>184>185>196> 194>1 92>189>188>294>292>290>288>286	OK
31	A27	A2	92>93>91>89>87>8>83>81>79>77>75>304>299>99>101>103>105>107>109> 111> 112>123>125>184>185>196>194>192	OK
32	A1	A3	37>38>40>32>45>47>49>51>53>55>57>59>301>63>65>300>68>71>302>73> 304>299>99>101>103>105>107>109>111	OK
33	A5	A8	278>277>275>274>257>272>270>268>266>264>262>260>258>256>253>251> 249>247>245>243>241>174>144>86>61>5>1	OK
34	A26	A3	92>93>91>89>87>85>83>81>79>77>75>304>299>99>101>103>105>107>109>111 >113>115>117>119>120>179>296>175	OK
35	A4	A12	207>206>204>202>200>199>306>294>292>290>288>286>284>282>281>265> 262>260>258>256>253>251>249>247>245	OK
36	A12	A22	35>36>34>43>30>28>26>24>21>20>17>2>6>72>95>96>177>242>244>246> 248 >250>252>254>127>310>312>314>316	OK
37	Home	A27	16>14>12>9>8>1>2>3>38>40>32>45>47>49>51>53>55>57>59>301>63>65> 300 >69>70>67>74>76>78>80>82>84>303>88	OK
38	A3	A24	240>297>168>170>172>295>176>178>181>182>185>196>194>192>189>188> 294> 292>290>288>286>284>282>281>265	OK
39	A8	A23	30>28>26>24>21>20>17>2>6>72>95>96>17>242>244>246>248>250>252>254>1 27>310>312>314>316>318>300>331>332	OK
40	A2	A1	268>266>264>262>260>258>256>253>251>249>247>245>243>241>174>144> 86>61>6>3>37	OK

by the system. This verification is essential for demonstrating the effectiveness and accuracy of the control system.

In the test, the authors used the control system, one PMV simulator to simulate the activity of PMV controlled by the system with the start position and destination position, the route data are collected and confirmed if the routes are the shortest based on the Dijkstra algorithm. The results in Table 1 show that basically the design of the system allows the vehicle to choose to calculate the path from any location defined on the map. This is consistent with the computational principle of the Dijkstra Algorithm.

4.2 Check the computation of the designed system

In testing the safety of the system, the authors used two or more PMV simulators to simulate the activity of PMVs in each area of intersections, thereby observing and evaluating whether PMVs have received orders at intersections according to FIFO methods. The results of the experiment have been summarized in Table 2. The authors emphasize that the Intersection IDs are detailed in Fig. 4, which includes 11 intersections. Each intersection is arranged and depicted as an example in Fig. 5. In Table 2, the Intersection IDs correspond to the paths shown in Fig. 6.

Table 2 Intersection test results

Intersection ID	Number of PMV	Control method	Test result
1	2	FIFO	OK
2	2	FIFO	OK
3	2	FIFO	OK
4	2	FIFO	OK
5	2	FIFO	OK
6	2	FIFO	OK
7	2	FIFO	OK
8	3	FIFO	OK
9	2	FIFO	OK
10	2	FIFO	OK
11	2	FIFO	OK

By referring to Fig. 4, we can identify each Intersection ID and see how they relate to the paths in Fig. 6.

Test results showed that at intersections, when the number of PMVs is from 2 to 3, the FIFO method solved the problem of controlling their trajectory well. In addition, the test results are saved as screenshots of the laboratory screen, so it is possible to check and refer to the actual control system image.

4.3 Testing the flexibility of the system

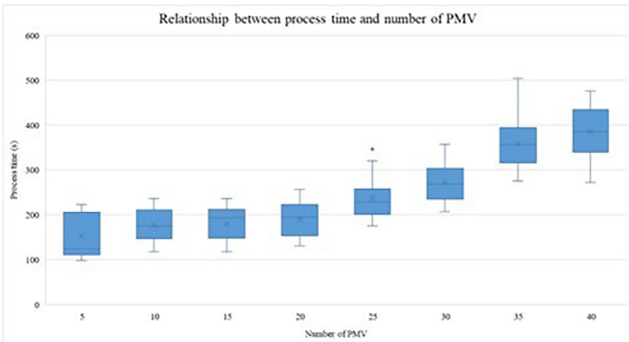
In testing the flexibility of the system, the authors used one or more PMV simulators to measure whether the system is capable of safe control when the number of PMVs increases from 5, 10, 15, 20, 25, 30, 35, 40. The vehicles receive command signals to follow the following positions: Home 1 > Random Destinations (A2, A4, A5, A8, A7, A27) >Home 2. When the PMV appears on the map designed in Section 3 at Home 1 location, it receives the command to go to randomly selected Destination locations from the previous set of locations, then return to Home 2 location and automatically go out of the map. When the vehicle moves in the map, it will be controlled by the Dijkstra algorithm and stopped at intersections according to FIFO principles as explained in Section 2. In each experiment, the time it takes for any PMV from receiving commands to drive from Home 1 position and return to Home 2 is measured as Process time, the time from when the first PMV starts from Home 1 to when the last one returns to Home 2 is measured and considered the Total process time.

Fig. 7 shows the result of experiments on actual simulation time on the map designed in Section 3. To simplify the simulation, the travel time between two points when simulating is 1 s, every 1 s the PMV can move to the next point on the map and will be stopped or passed through the intersection based on the control signal received from the system. In Fig. 7, the horizontal axis represents the number of PMV vehicles used in each experiment, the vertical axis represents the distribution of time the commands completed.

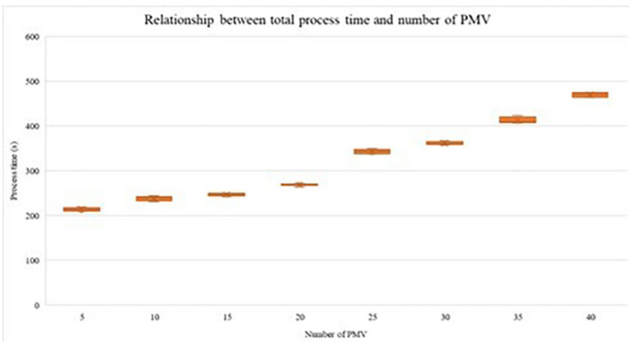
The result of Fig. 7 shows the travel time of each PMV, as the number of PMVs increases, the greater the travel time deviation. This shows that as the number of PMVs increases, the probability of having to stop at intersections increases, and as a result, the order execution time of each PMV (Fig. 7(a)) as well as the total execution time of all orders (Fig. 7(b)) will increase. Specifically, in this case, if the time of 10 PMVs is taken as the comparative time, when the number of PMVs is from 5 to 20, the order execution time of each PMV hardly changes in the range of no more than 10%, showing that with such a number of PMVs will not be affected much. But when the number increased up to 25, 30, 35, 40 PMVs, the order execution time of all vehicles increased from 35%, 55%, 103%, 119%, respectively. This suggests that the greater the number of PMVs, the probability of having to stop PMVs due to traffic control increases by a quadratic function.

5 Conclusions

This paper aims to create an automatic wheelchair trajectory control system for people with disabilities so that



(a)



(b)

Fig. 7 The process time results with the number of PMVs up to 40, (a) Process time, (b) Total process time they can integrate more into their daily life in public. The authors proposed a control structure model required for PMVs. Simple and safe traffic control methods for PMVs in complex traffic environments are proposed and tested. The authors designed and built the system on a specific scale so that the wheelchair can be controlled automatically within a certain range at the campus of the University of Transport and Communications in Hanoi, Vietnam. Finally, the authors simulated and evaluated the safety and flexibility of the designed system. The simulation results achieved show that the system has been designed to ensure standard conditions such as efficiency, safety, and flexibility of the system when the number of vehicles reaches 40 PMVs. Actual evaluation results have shown the clear effectiveness of the control method combining Dijkstra and FIFO algorithms to control motion trajectories for PMVs in general and wheelchairs in particular.

This problem needs to be solved in the future, upgrading the Dijkstra algorithm so that the wheelchair can know how to turn itself when there are obstacles or traffic jams in its way.

6 Acknowledgement

The research was supported by the National Research, Development and Innovation Office through the project "Cooperative emergency trajectory design for connected autonomous vehicles" (NKFIH: 2019-2.1.12-TÉT_VN). The research was also supported by the International project (Vietnam-Japan) "Research and Development of Personal Mobility Vehicles" between University of Transport and Communications and Osaka Metropolitan University.

References

Công, P. (2020) "Việt Nam có 6,2 triệu người khuyết tật, chiếm hơn 7% dân số", (Vietnam has 6.2 million people with disabilities, accounting for more than 7% of the population) [online] Available at: https://dantri.com.vn/an-sinh/viet-nam-co-62-trieu-nguoi-khuyet-tat-chiem-hon-7-dan-so-20201230123147508.htm [Accessed: 26 August 2024] (in Vietnamese)

Cooper, R. A., Boninger, M. L., Cooper, R., Kelleher, A. (2006) "Use of the INDEPENDENCE 3000 IBOTTM transporter at home and in the community: A case report", Disability and Rehabilitation: Assistive Technology, 1(1–2), pp. 111–117. https://doi.org/10.1080/09638280500167571

Daveler, B., Salatin, B., Grindle, G. G., Candiotti, J., Wang, H., Cooper, R. A. (2015) "Participatory design and validation of mobility enhancement robotic wheelchair", Journal of Rehabilitation Research and Development, 52(6), pp. 739–750. https://doi.org/10.1682/JRRD.2014.11.0278

Google Map (2024) "Map of university of tranformation and communication", [online] Available at: https://maps.app.goo.gl/16Cc9nvX9JUpu22A9 [Accessed: 03 July 2024]

Hokuyo (2024) "Hokuyo Lidar sensors", [online] Available at: https://www.mathworks.com/help/lidar/hokuyo-lidar-sensors.html?s_tid=CRUX_lftnav [Accessed: 26 August 2024]

Jin, W., Li, L. (2007) "First-in-first-out is violated in real traffic", Proceedings of Transportation Research Board Annual Meeting, pp. 1–25. [pdf] Available at: https://its.uci.edu/~wjin/publications/TRB07-1174.pdf [Accessed: 26 August 2024]

JTS "Japan Technology Solution Joint Stock Company", [online] Available at: https://innovations-jts.com [Accessed: 26 August 2024]

Julius Fusic, S., Ramkumar, P., Hariharan, K. (2018) "Path planning of robot using modified dijkstra Algorithm", In: 2018 National Power Engineering Conference (NPEC), Madurai, India, pp. 1–5. ISBN 978-1-5386-3803-3 https://doi.org/10.1109/NPEC.2018.8476787

Lanning, D. R., Harrell, G. K., Wang, J. (2014) "Dijkstra's algorithm and Google maps", Proceedings of the 2014 ACM Southeast Regional Conference, pp. 1–3. ISBN 9781450329231 https://doi.org/10.1145/2638404.2638494

Mirzaeinia, A., Shahmoradi, J., Roghanchi, P., Hassanalian, M. (2019) "Autonomous routing and power management of drones in gps-denied environments through Dijkstra algorithm", AIAA Propulsion and Energy 2019 Forum, ISBN 9781624105906 https://doi.org/10.2514/6.2019-4462

Ruan, C., Luo, J., Wu, Y. (2014) "Map navigation system based on optimal Dijkstra algorithm", In: 2014 IEEE 3rd International Conference on Cloud Computing and Intelligence Systems, Shenzhen, China, pp. 559–564. ISBN 978-1-4799-4719-5 https://doi.org/10.1109/CCIS.2014.7175798

Swaniawski, C. (2022) "Application of Dijkstra's algorithm", [online] Available at: https://morioh.com/a/cdc55aea2dba/applications-of-dijkstras-shortest-path-algorithm [Accessed: 26 August 2024]

Tan, V. V., Thinh, N. Q., Chihiro, N. (2022) "Proposing a model of centralized control system on personal mobility vehicles", Journal of Science and Technology, HaUI, 58(6A), pp. 62–67. https://doi.org/10.57001/huih5804.70

Tyagi, N. (2020) "Dijkstra's algorithm: the shortest path algorithm", Analytic Steps, Machine Learning, Dec. 14. [online] Available at: https://www.analyticssteps.com/blogs/dijkstras-algorithm-shortest-path-algorithm [Accessed: 26 August 2024]

Wang, H., Candiotti, J., Shino, M., Chung, C.-S., Grindle, G. G., Ding, D., Cooper, R. A. (2013) "Development of an advanced mobile base for personal mobility and manipulation appliance generation II robotic wheelchair", The Journal of Spinal Cord Medicine, 36(4), pp. 333–346. https://doi.org/10.1179/2045772313Y.0000000094

WHO (2023) "Disability", [online] Available at: https://www.who.int/news-room/fact-sheets/detail/disability-and-health [Accessed: 26 August 2024]

Wikipedia (2024) "Dijkstra's algorithm", [online] Available at: https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm [Accessed: 26 August 2024]

Zhang, Z. Zhao, Z. (2014) "A multiple mobile robots path planning algorithm based on a-star and Dijkstra algorithm", International Journal of Smart Home, 8(3), pp. 75–86, https://doi.org/10.14257/ijsh.2014.8.3.07

Zhou, M., Gao, N. (2019) "Research on optimal path based on Dijkstra algorithms", Atlantis Press, pp. 884–892. ISBN 978-94-6252-708-9 https://doi.org/10.2991/icmeit-19.2019.141