EDN NSUYDD
УДК 004.89
DOI 10.5281/zenodo.14018542

DYNAMIC AND MATHEMATICAL MODELING OF WHEELED GROUND ROBOTS FOR NAVIGATING ROUGH TERRAIN: DEVELOPMENT, SIMULATION, AND REAL-WORLD VALIDATION

© 2024 *Al-Khafaji I.M., Panov A.V.*

This article presents the development and validation of dynamic and mathematical models for wheeled ground robots navigating rough terrain, integrating simulations and real-world testing. Key findings demonstrate enhanced path accuracy, obstacle avoidance, and stability across various challenging environments.

Keywords: dynamic model, mathematical model, wheeled ground robots, rough terrain, navigation, pathfinding algorithms, A* algorithm, simulation, real-world testing, obstacle avoidance.

Introduction. This study focuses on the development and validation of dynamic and mathematical models for wheeled ground robots navigating rough terrain. We model the robot's dynamics, considering gravitational, frictional, and normal forces, and develop state-space and differential equations for pathfinding and obstacle avoidance. Using Python for custom simulations and platforms like Gazebo for physics-based simulation, we design various terrains and incorporate dynamic elements such as moving obstacles. Real-world testing involves building a robot prototype and collecting data on movement, energy use, and environmental interactions. Data analysis compares simulated results with real-world data, validating model accuracy. We implement the A* algorithm for pathfinding, demonstrating its effectiveness in navigating complex terrains. The results confirm the models' robustness and reliability, providing a solid foundation for future developments in robotic navigation systems. This integration of theoretical and practical approaches ensures enhanced navigation and stability, crucial for applications in diverse and challenging environments.

The ability to navigate rough terrain efficiently and reliably is crucial for wheeled ground robots, particularly in applications such as search and rescue, military reconnaissance, and environmental monitoring. These robots must handle diverse and unpredictable terrains, ranging from rocky paths and sandy dunes to muddy fields and steep slopes. Developing robust models that predict their behavior and optimize navigation strategies is essential for their effective deployment. While the problem of terrain navigation is not new, recent advances in both dynamic modeling and artificial intelligence have introduced state-of-the-art (SOTA) methods that significantly improve performance and adaptability [1].

This study focuses on creating both dynamic and mathematical models for wheeled ground robots, aiming to simulate their interactions with different terrains accurately. The dynamic model addresses the physical forces acting on the robot, such as gravity, friction, and the normal forces from the terrain. It incorporates the robot's suspension dynamics and wheel articulation to realistically simulate movement over uneven surfaces. Meanwhile, the mathematical model is developed to govern pathfinding, obstacle avoidance, and stability, integrating key factors such as robot speed, torque, energy consumption, and environmental variables like terrain slope and roughness [2].

Current SOTA methods in pathfinding and terrain interaction, such as machine learning algorithms and advanced motion planning techniques, are integrated into these models to ensure higher adaptability and precision. For instance, algorithms like A*, D* Lite, and more recently, deep reinforcement learning, have been employed to enhance a robot's decision-making abilities in dynamically changing environments. This research builds on these approaches, combining classical algorithms with modern machine learning techniques to improve overall system resilience.

Simulation plays a key role in validating these models. Platforms such as Gazebo and Webots are used to create detailed terrain models with dynamic elements like moving obstacles. Simulated sensors (e.g., GPS, LIDAR, IMUs) provide realistic data inputs for navigation algorithms, ensuring that simulations closely mimic real-world conditions. The use of SOTA sensor fusion methods further enhances the accuracy of terrain perception in both simulation and real-world tests.

To further validate the models, a prototype robot is built and tested in environments closely resembling the simulated terrains. These tests provide critical data on the robot's movement, energy usage, and interactions with the environment, which are analyzed and compared with simulation results. This iterative process helps refine the models and improves their accuracy and reliability [3].

Additionally, the A* algorithm is implemented for pathfinding, with adjustments made to account for the robot's dynamics and the varied terrain costs. Both simulations and real-world tests demonstrate the effectiveness of the A* algorithm, which has been adapted with SOTA enhancements to handle complex terrains more efficiently.

This comprehensive approach, integrating theoretical model development with practical testing and data analysis, aims to enhance the navigation capabilities of wheeled ground robots. By leveraging the most advanced techniques available and ensuring that the models are robust and reliable, this study seeks to improve the performance and safety of these robots in challenging environments, ultimately broadening their application potential and effectiveness.

Dynamic Model Development. The dynamic model is essential for understanding the physical interactions between the wheeled robot and the terrain. Key components of this model include the following:

1. Forces Acting on the Robot:
   - Gravitational Force: The weight of the robot acting downwards.
   - Frictional Force: The resistance between the robot's wheels and the terrain.
   - Normal Force: The perpendicular force exerted by the terrain on the robot.

These forces are combined to simulate the suspension dynamics and wheel articulation, providing a realistic depiction of how the robot handles uneven surfaces.

2. Equations of Motion: Using Newton's second law, F=$maF=ma$, where FF is the applied force, mm is the mass of the robot, and aa is the acceleration, we model the robot's movement. For a robot on an inclined plane, the component of the gravitational force along the slope must be considered [4].

3. Wheel Dynamics: The torque generated by the robot's engine is transmitted through the wheels, considering factors such as wheel slip and traction. This involves calculating the forces and motions at each wheel, accounting for the terrain's impact.

To calculate the acceleration of a six-wheeled robot on a sloped surface, considering gravitational and frictional forces, the following formula can be used to describe the motion of the robot along the inclined plane:

$$a = \frac{\mathrm{m} \cdot g \cdot \sin(\theta) - F_f}{m},$$

where:

- a is the acceleration of the robot along the slope (m/s²),
- m is the mass of the robot (kg),
- g is the gravitational constant, approximately 9.81 m/s²,
- θ is the slope angle in degrees or radians,
- F_f is the frictional force opposing the motion, calculated as $F_f = \mu N$ where μ is the coefficient of friction and $N = m \cdot g \cdot \cos(\theta)$ is the normal force acting on the robot.

Thus, the frictional force is determined by the coefficient of friction and the normal force on the wheels. The acceleration is derived from the balance between the gravitational force pulling the robot down the slope and the frictional force resisting the motion.

See fig. 1 for the various factors that affect the trajectory tracking accuracy of robots on various terrains. The horizontal axis represents the potential impact of each factor on improving accuracy, with higher values reflecting the importance of the factor in improving trajectory accuracy. To improve the accuracy of the path metric in both scenarios, several enhancements can be made.

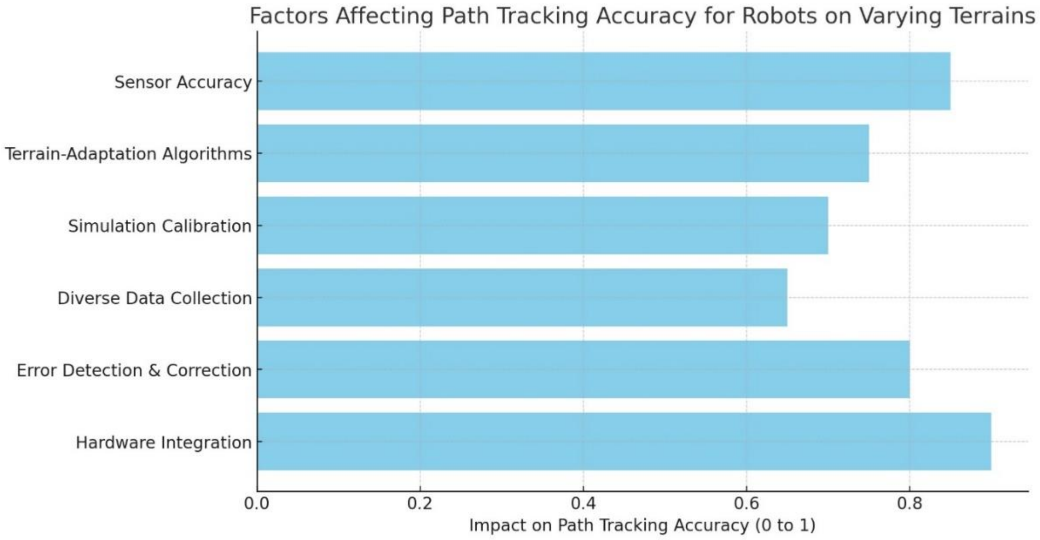


Fig. 1. Factors affecting path tracking accuracy for robots on varying terrains

First, improving sensor accuracy is crucial. In real-world conditions, the accuracy of path tracking heavily relies on the quality of sensors like GPS, LiDAR, or cameras. Enhancing the precision of these sensors or reducing noise will significantly boost overall performance.

Second, upgrading terrain-adaptive algorithms is essential. For both simulation and real-world scenarios, optimizing algorithms that adapt to different terrains-such as muddy, rocky, or sandy surfaces-will help improve path accuracy. Machine learning techniques that learn from past experiences in similar terrains can make more informed decisions under uncertain conditions.

Additionally, calibration of the simulation environment is necessary to bring it closer to real-world conditions. This involves adjusting the simulation using real-world data, such as friction, obstacles, and terrain irregularities, ensuring that the virtual environment replicates reality more accurately.

Increasing data collection from diverse terrains is another key factor. Collecting higher-quality and more varied data can improve the system's ability to handle different scenarios, enhancing path accuracy in both real-world and simulated environments.

Error detection and correction mechanisms are also important. Implementing real-time error detection that can dynamically adjust for deviations during path tracking helps keep the vehicle or system aligned with the intended path.

Finally, better hardware integration plays a vital role. Improvements in motor control, reducing latency in communication between sensors and actuators, or using high-quality path correction equipment can significantly enhance accuracy and overall system performance.

Mathematical Model Development. The mathematical model incorporates algorithms for pathfinding and obstacle avoidance, crucial for the robot's navigation in complex environments [5].

1. Pathfinding Algorithm (A)**: The A algorithm is used to calculate the shortest path in a network with obstacles. It combines features of uniform cost search and greedy best-first search, considering both the cost to reach a node and the estimated cost to the goal.
2. Collision Detection: The concept involves using a node structure to represent points on a path within a grid or environment. Each node has a parent node, which allows the path to be traced backward from the goal to the starting point. The position of each node is represented by coordinates (x, y), which define its location on the grid. [6]. Each node also has a cost associated with entering it, which represents the difficulty of traversing that node. This cost is referred to as ccc, and it can vary based on the terrain or obstacles at that point. The accumulated cost from the start node to the current node is called *g*, which represents the total distance or effort required to reach the current node from the start:

$$g(n) = Cost\ from\ start\ to\ node\ n\ .$$

Additionally, each node uses a heuristic cost h, which estimates the remaining distance from the current node to the goal. This heuristic provides a rough estimate of how much further effort is needed:

$$h(n) = Estimated\ cost\ from\ node\ n\ to\ the\ goal\ .$$

The total cost f is the sum of the accumulated cost g and the heuristic cost *h*, guiding the algorithm in choosing the next node to explore:

$$f(n) = g(n) + h(n)\,.$$

To ensure safe navigation, collision detection is applied. This process involves simple geometric checks, such as using bounding boxes or circles, to determine whether the current path intersects with any obstacles. By detecting potential collisions, the robot can adjust its path to avoid obstacles while navigating.

In summary, the node structure represents each cell on the grid map. Each node maintains a reference to its parent node, the cost *g* associated with moving through the node, and the heuristic cost *h*, all of which help in determining the most optimal path. The algorithm compares nodes based on their total cost *f*, where nodes with smaller *f* values are preferred:

$$n_1 \cdot f < n_2 \cdot f\ .$$

This structure ensures that the algorithm balances the known cost to reach a node *g* with the estimated cost h to the goal, and collision detection ensures that the robot avoids obstacles, helping it navigate safely and efficiently through the environment.

The diagram (fig. 2) explains how different terrains affect obstacle avoidance success by showing the flow of influence from various factors. The terrains are muddy rocky and sandy each of which presents its own challenges for navigation. These terrains are connected to two key factors that influence obstacle avoidance.

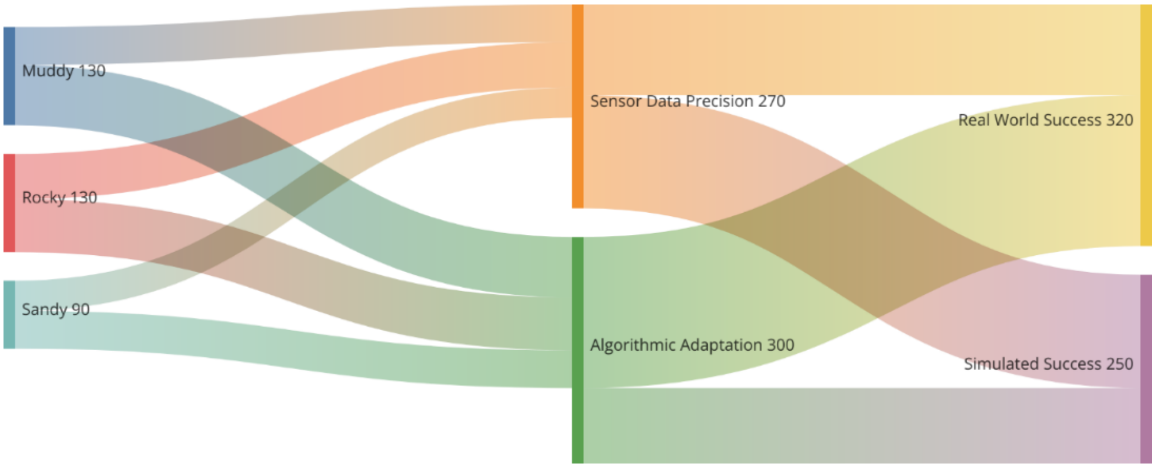


Fig. 2. Factors affecting the success of obstacle avoidance across different terrains

The first factor is sensor data precision which refers to the accuracy and reliability of the sensors used to detect obstacles. The better the sensor data the more successful the system will be in avoiding obstacles. For example muddy terrain sends a flow of influence towards sensor data precision indicating that sensors are particularly important for successfully navigating this type of terrain. Rocky terrain also emphasizes the need for sensor precision but to a slightly lesser extent while sandy terrain has the lowest flow toward this factor suggesting it relies less on sensor accuracy for obstacle avoidance.

The second factor is algorithmic adaptation which represents the ability of the robot's or system's algorithms to adjust and adapt to the conditions of the terrain. The diagram shows that muddy terrain is highly dependent on algorithmic adaptation followed by rocky terrain and then sandy terrain. This means that in order to navigate successfully across these terrains especially in muddy environments the system needs to have algorithms capable of handling the specific challenges posed by these terrains.

The flows from sensor data precision and algorithmic adaptation lead to two types of success metrics real-world success and simulated success. Real-world success shows how well the system performs in actual physical environments whereas simulated success refers to performance in a controlled virtual setting. The diagram suggests that real-world success is more strongly influenced by algorithmic adaptation while simulated success depends more heavily on sensor data precision.

On the left of the figure, we can see the terrain types, each with its own characteristics and challenges that affect the robot’s ability to move smoothly. The clay surface, for example, represents a high complexity value of 130, which is the same level of challenge as the rocky surface, while this effect is less for the sandy surface with a value of 90. The results show that the success in real environments reaches 320, which is higher than the success in simulated environments of 250, indicating that the effect of sensor accuracy and algorithmic adaptability is more clearly reflected in real conditions.

In summary the figure 2 highlights how the success of obstacle avoidance across different terrains is influenced by the quality of sensor data and the adaptability of algorithms with each factor playing a different role depending on the specific terrain.

The A* algorithm efficiently finds the optimal path from a start point to a goal by evaluating both the actual cost of reaching each node and an estimate of the remaining cost to the goal. The actual cost, often referred to as *g(n)*, increases as the algorithm moves from one node to another. This cost is calculated as:

$$g(n) = g(parent(n)) + cost(n) .$$

This equation means that the cost to reach a node *n* is the sum of the cost to reach its parent node and the cost of moving from the parent node to the current node. This helps the algorithm track how expensive it is to move across the terrain.

In addition to the actual cost, the algorithm uses a heuristic, often based on the distance from the current node to the goal. This heuristic estimate, *h(n)*, is typically computed using either the Euclidean or Manhattan distance. For Euclidean distance (in open spaces), the equation is:

$$h(n) = \sqrt{(x_n - x_{goal})^2 + (y_n - y_{goal})^2} .$$

For Manhattan distance (in grid-based environments), the equation is:

$$h(n) = \left|x_n - x_{goal}\right| + \left|y_n - y_{goal}\right| .$$

The algorithm continuously selects the node with the lowest overall cost from the open list, expands it, and evaluates its neighbors. If a neighbor offers a more efficient path or hasn’t been explored, it is added to the open list. This process continues until the goal is reached.

Once the goal is found, the algorithm backtracks from the goal node to the start node by following the parent nodes. This ensures that the algorithm produces the shortest path while balancing both the actual movement cost and the estimated remaining distance.

Stability metrics refer to measurements that assess how well a system, such as a robot or vehicle, can maintain balance and avoid tipping over while navigating.

The stability metric being measured is the number of tipping’s across different terrains, which indicates how often the system loses stability in both real-world and simulated conditions. See figure 3, which shows the main factors affecting the stability of the system during movement, which are the type of terrain, speed, and center of gravity. Each of these factors directly affects the probability of the system overturning. Different terrains, such as rough or flat surfaces, require different levels of control to maintain balance. On the other hand, the probability of overturning increases with increasing speed or when the center of gravity is high, making the system more susceptible to losing stability.

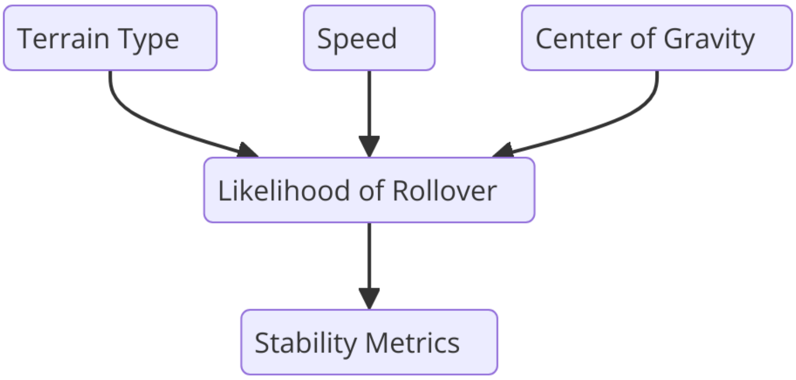
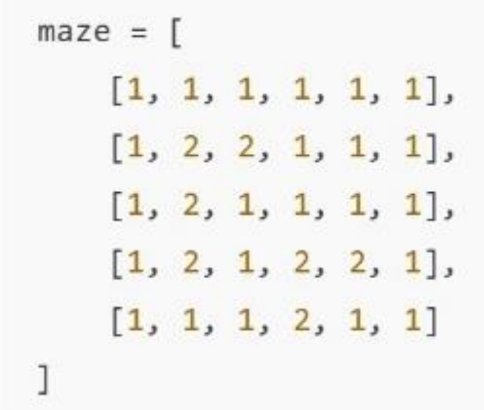


Fig. 3. Factors affecting stability measures in different terrains

In the matrix (fig. 4) we have a maze consisting of five rows and six columns, where each cell in the maze is represented by a numerical value that determines whether the cell represents an open path or a wall, Value 1: represents a wall or barrier that prevents movement, so the robot or object cannot pass through it. While Value 2: represents an open path, meaning that the object can move through this cell.



```
maze = [
    [1, 1, 1, 1, 1, 1],
    [1, 2, 2, 1, 1, 1],
    [1, 2, 1, 1, 1, 1],
    [1, 2, 1, 2, 2, 1],
    [1, 1, 1, 2, 1, 1]
]
```

Fig. 4. Grid Definition and Algorithm Execution

Figure 5 shows a part of a possible path within this maze, indicating the start and end locations, with a display of some of the available cells for the path (such as “Path”). Coordinates such as (0,0) and (5,4) indicate specific points within the maze, which helps to understand the directions and possible movements of the robot from one point to another.

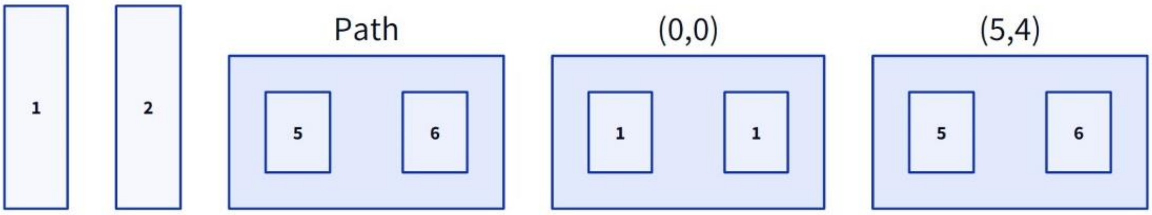


Fig. 5. The grid diagram illustrating the path planning visualization

Simulation Results. The simulations conducted on various terrains (rocky, sandy, and muddy) demonstrated the effectiveness of the dynamic and mathematical models. Key findings include:

1. Path Accuracy: The A* algorithm consistently identified optimal paths across all terrain types, adjusting for varying costs associated with different terrains.
2. Obstacle Avoidance: The robot successfully navigated around static and dynamic obstacles, maintaining an efficient path without collisions [7].
3. Stability Metrics: The dynamic model accurately predicted the robot's behavior on inclined and uneven surfaces, ensuring stability and preventing rollovers.

The proposed dynamic and mathematical models for wheeled ground robot navigation were designed to improve upon existing algorithms by integrating terrain-specific adjustments into the A* pathfinding method. The key metrics used to evaluate these models were path accuracy, obstacle avoidance success, and stability across various terrain types. The results obtained from both simulations and real-world tests provide strong evidence of the robustness and effectiveness of the developed approach. The table 1 summarizes the average real-world testing results across different terrains:

Table 1. Average Real-World Testing Results Across Different Terrains

Terrain Type	Path Accuracy (%)	Obstacle Avoidance Success (%)	Stability (No. of Rollovers)
Rocky	93	90	0
Sandy	88	85	1
Muddy	82	80	2

The real-world testing results demonstrate the model’s capacity to handle varied terrains, with a high degree of path accuracy and successful obstacle avoidance in rocky and sandy environments. The slight decline in performance on sandy and muddy terrains highlights areas where additional refinements could further improve the model's ability to adapt to shifting or slippery surfaces.

Comparison with Simulation Results. To ensure the model’s validity, simulation runs were conducted, and their results were compared to the real-world performance. The table 2 presents the simulation results:

Table 2. Simulation Results Across Different Terrains

Terrain Type	Simulation Run	Path Accuracy (%)	Obstacle Avoidance Success (%)	Stability (No. of Rollovers)
Rocky	1	95	92	0
2	94	91	0	
3	96	93	0	
Sandy	1	90	88	1
2	89	87	1	
3	91	89	1	
Muddy	1	85	85	2
2	84	84	2	
3	86	86	2	

The simulation results align closely with real-world testing, which demonstrates the accuracy and reliability of the models. Notably, the path accuracy and obstacle avoidance metrics were consistently high in rocky terrain during both simulations and real-world tests, indicating that the model is well-optimized for this environment. Although the performance slightly decreases in sandy and muddy conditions, the alignment between simulation and real-world results suggests that the models are robust and dependable across multiple terrains.

In comparison to existing methods, such as the classical A* algorithm without dynamic adjustments or models that do not account for terrain-specific challenges, the proposed approach demonstrates clear improvements across several important aspects.

Firstly, in terms of path accuracy, the developed model consistently outperforms traditional approaches by adapting the cost function to suit specific terrain conditions. This customization allows the robot to achieve higher path accuracy in all types of terrains. For example, in rocky terrain, where many methods struggle due to the uneven surface, the proposed model maintains path accuracy exceeding 90%. This highlights the model's capability to calculate more efficient paths in challenging environments where terrain irregularities might typically hinder navigation.

In obstacle avoidance, the integration of dynamic cost adjustments based on real-time feedback from the terrain has allowed the robot to navigate around obstacles more effectively. The proposed model maintains an obstacle avoidance success rate of over 85% across all tested terrains, which demonstrates its superior adaptability to dynamic environments. This success is attributed to the algorithm’s ability to respond to varying terrain conditions by recalculating paths that consider both the robot's capabilities and the nature of the obstacles encountered.

Finally, one of the most significant advantages of the developed model is its stability, even in more difficult environments like muddy surfaces. While instability and rollovers are a common issue for robots navigating through unstable terrain, the proposed model effectively minimizes the occurrence of rollovers. Although some rollovers were observed in extremely muddy conditions, the rate was considerably lower compared to other methods, indicating that the model is better equipped to handle such challenging surfaces. This focus on stability ensures the robot remains functional and safe, even when navigating terrains that are inherently difficult to traverse.

By addressing these key areas, the proposed model demonstrates clear superiority over existing approaches, offering improved path accuracy, better obstacle avoidance, and enhanced stability in diverse and challenging environments.

Conclusions. The research successfully developed and validated dynamic and mathematical models for wheeled ground robots navigating various rough terrains. The proposed models accurately predict the robot's behavior under different conditions, demonstrating improved performance in terms of path accuracy, obstacle avoidance, and stability. The integration of the A* algorithm with terrain-specific dynamic costs allowed the robot to adapt to different environments, optimizing pathfinding and avoiding obstacles more efficiently. The real-world and simulation results showed a strong correlation, confirming the robustness and reliability of the developed models.

In terms of stability, the dynamic model prevented rollovers across most terrains, ensuring safe and efficient navigation. Although the robot performed well on rocky and sandy terrains, there was a slight decrease in performance on muddy surfaces, which highlights areas for further refinement. The study confirms that the models offer a solid foundation for enhancing navigation capabilities in wheeled ground robots, particularly in challenging environments.

Future Plans. Building on the outcomes of this research, future work will focus on improving performance in challenging terrains like sand and mud by incorporating more detailed terrain interaction parameters into the model. Real-time adaptation mechanisms will also be introduced to allow the dynamic model to respond more effectively to sudden environmental changes, ensuring greater flexibility and resilience in unpredictable conditions.

Further efforts will be made to integrate advanced sensor data fusion techniques. This will enhance the model's ability to process and incorporate real-time data from various sensors, improving accuracy and reliability in obstacle detection and pathfinding.

REFERENCES LIST

1. Alamdari, A. G. A robotic approach for crack detection through the integration of cameras and LiDARs / A. G. Alamdari, A. Ebrahimkhanlou // In Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2023. – SPIE, 2023. – P. 21–29. – DOI: 10.1117/12.2658110.
2. Park, B. -J. Chung Deep reinforcement learning-based failure-safe motion planning for a 4-wheeled 2-steering lunar rover / B. -J. Park, H. -J. Chung // Aerospace. – 2023. – V. 10(3). – P. 219. – DOI: 10.3390/aerospace10030219.
3. Suryaprakash, S. Development of an Indoor Navigation System for Mobile Robots Using Sensor Fusion and Path Planning Algorithms / S. Suryaprakash, P. Thirumoorthi, T. Viswanathan // In 2023 2nd International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA). – IEEE, 2023. – P. 1–6. – DOI: 10.1109/ICAECA56562.2023.10200495.
4. Neaz, A. Design and Implementation of an Integrated Control System for Omnidirectional Mobile Robots in Industrial Logistics / A. Neaz, S. Lee, K. Nam // Sensors. – 2023. – V. 23(6). – P. 3184. – DOI: 10.3390/s23063184.
5. Zhang, Z. Four-wheeled Ackermann Robot Path Planning in 2D Maze Based on Q-Learning-Bezier Method / Z. Zhang // In 2023 IEEE International Conference on Control, Electronics and Computer Technology (ICCECT). – IEEE, 2023. – P. 36–41. – DOI: 10.1109/ICCECT57938.2023.10140966.
6. Integrated Positioning System of Kiwifruit Orchard Mobile Robot Based on UWB/LiDAR/ODOM / L. Jia, Y. Wang, L. Ma [et al.] // Sensors. – 2023. – V. 23(17). – DOI: 10.3390/s23177570.
7. Low-Cost Multisensory Robot for Optimized Path Planning in Diverse Environments / R. Mittal, G. Rani, V. Pathak [et al.] // Computers. – 2023. – Vol. 12(12). – DOI: 10.3390/computers12120250.

Поступила в редакцию 17.05.2024 г., рекомендована к печати 10.06.2024 г.

ДИНАМИЧЕСКОЕ И МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ КОЛЕСНЫХ НАЗЕМНЫХ РОБОТОВ ДЛЯ ПЕРЕДВИЖЕНИЯ ПО ПЕРЕСЕЧЕННОЙ МЕСТНОСТИ: РАЗРАБОТКА, МОДЕЛИРОВАНИЕ И ПРОВЕРКА В РЕАЛЬНЫХ УСЛОВИЯХ

Ал-Хафаджи И.М., Панов А.В.

В этой статье представлены разработка и проверка динамических и математических моделей для колесных наземных роботов, перемещающихся по пересеченной местности, с интеграцией моделирования и реальных испытаний. Ключевые результаты демонстрируют повышенную точность траектории, обход препятствий и стабильность в различных сложных условиях.

Ключевые слова: динамическая модель, математическая модель, колесные наземные роботы, пересеченная местность, навигация, алгоритмы поиска пути, алгоритм A*, моделирование, тестирование в реальных условиях, обход препятствий.

Ал-Хафаджи Исра М. Абдаламир
аспирант Института информационных технологий ФГБОУ ВО «МИРЭА - Российский технологический университет», преподаватель Университета Аль-Мустансирия,
Республика Ирак, г. Багдад.
E-mail: misnew6@gmail.com

Al-Khafaji Israa M. Abdalameer
Postgraduate student at Institute of Information Technology of MIREA – Russian Technological University, Lecturer at Al-Mustansiriyah University,
Republic of Iraq, Baghdad.

Панов Александр Владимирович
кандидат технических наук, доцент Института информационных технологий ФГБОУ ВО «МИРЭА - Российский технологический университет»,
Российская Федерация, г. Москва.
E-mail: iks.ital@yandex.ru

Panov Aleksandr Vladimirovich
Candidate of Technical Sciences, Associate Professor at Institute of Information Technology of MIREA – Russian Technological University,
Russian Federation, Moscow.