A Survey of Path Planning and Obstacle Avoidance Techniques in Mobile Robotics

Amanzhol Bektemessov
International Engineering Technological University, Kazakhstan
amanzhol.bektemessov@gmail.com (corresponding author)

Received: 5 August 2025 | Revised: 29 August 2025 and 10 September 2025 | Accepted: 11 September 2025

DOI: https://doi.org/10.48084/etasr.13885

ABSTRACT

This paper presents a comprehensive survey of path planning and obstacle avoidance techniques in mobile robotics, addressing their theoretical foundations, algorithmic developments, and practical implementations. The study categorizes path planning strategies into classical, sampling-based, optimization-based, and learning-based approaches, highlighting their respective strengths, limitations, and applicability across different environments. Obstacle avoidance methods are similarly examined through reactive, predictive, and learning-driven paradigms, with an emphasis on sensor technologies and real-time decision-making. Integrated systems that combine global and local planning, hierarchical control architectures, and embedded execution frameworks are analyzed to demonstrate how contemporary mobile robots navigate safely and efficiently in complex, dynamic settings. Case studies, including the Robot Operating System (ROS) Navigation Stack, delivery robots, and robotic vacuums, are used to illustrate real-world deployments. Furthermore, the paper identifies ongoing challenges and open research questions related to planning under uncertainty, real-time adaptability, human-aware navigation, multi-robot coordination, and generalization through transfer learning. The discussion is supported by figures and tables summarizing algorithmic trade-offs and system architectures. This survey aims to provide researchers and practitioners with a clear taxonomy, comparative evaluation, and forward-looking insights that will inform the development of more robust, adaptive, and intelligent navigation systems in the next generation of autonomous mobile robots.

Keywords-path planning; obstacle avoidance; mobile robotics; autonomous navigation; reinforcement learning; real-time systems; sensor fusion; dynamic environments; robot control; multi-robot coordination

I. INTRODUCTION

The increasing demand for autonomous systems across various domains, ranging from warehouse automation and autonomous delivery to planetary exploration and urban mobility has elevated the importance of robust path planning and obstacle, avoidance in mobile robotics. These two interrelated functions are foundational for enabling robots to navigate safely and efficiently through complex and dynamic environments without human intervention [1].

Path planning involves determining a feasible and optimal trajectory from a start to a goal location, accounting for environmental constraints, robot kinematics, and mission objectives. Obstacle avoidance, on the other hand, focuses on ensuring real-time responsiveness to static or dynamic obstacles that may not have been initially accounted for during planning [2]. The interplay between these two components defines the efficacy of robotic autonomy in real-world deployments, particularly in GPS-denied, cluttered, or unpredictable settings [3].

Over the decades, a wide spectrum of techniques has emerged to address these challenges, including classical deterministic algorithms, sampling-based strategies, optimization-based methods, and, more recently, data-driven approaches such as reinforcement learning and deep learning [4]. Each category brings trade-offs in terms of computational complexity, adaptability, and generalizability across environments.

Given the diversity and rapid evolution of the field, there is a pressing need for a comprehensive survey that not only categorizes these approaches but also compares their strengths, limitations, and applicability across different robotic platforms and use cases [5]. Moreover, the convergence of machine learning with traditional robotics is reshaping the landscape, introducing new paradigms for learning-based navigation and predictive obstacle avoidance [6].

This paper presents a structured and critical review of existing path planning and obstacle avoidance techniques, emphasizing their theoretical foundations, implementation strategies, and performance in practical scenarios.

A. Contributions of this Survey

The primary contribution of this survey lies in its comprehensive and up-to-date synthesis of path planning and obstacle avoidance techniques in the field of mobile robotics, covering both classical approaches and recent advancements driven by machine learning and artificial intelligence. Unlike previous surveys that often treat these areas in isolation or focus on narrow algorithmic categories, this paper integrates both domains into a unified framework, offering a clearer understanding of how planning and avoidance strategies interact in real-world systems.

Specifically, the key contributions are as follows:

- A structured classification of path planning algorithms into four main families: classical, sampling-based, optimization-based, and learning-based methods, with a comparative analysis supported by Table I and citations to foundational and recent works.
- A taxonomy of obstacle avoidance strategies, including sensor-driven reactive methods, predictive control, and deep learning-based approaches, presented with practical insights into their integration with real-time systems.
- A detailed discussion of integrated navigation architectures, including Robot Operating System (ROS)-based implementations, hierarchical control layers, and embedded system constraints, supported by current case studies.
- The identification of knowledge gaps and articulation of open research challenges in Section VI, offering concrete directions for future work on generalization, uncertainty, multi-agent coordination, and sim-to-real transfer.

While this is not an experimental study, the paper offers a critical analysis and curated references from over 40 recent peer-reviewed sources (2021–2025), making it a valuable resource for researchers and practitioners seeking a consolidated overview of this fast-evolving field.

II. FUNDAMENTALS OF PATH PLANNING AND OBSTACLE AVOIDANCE

A. *Definitions and Terminology*

In the domain of mobile robotics, path planning and obstacle avoidance are distinct yet interdependent components. Path planning refers to the computational process of identifying a collision-free trajectory from an initial position to a target location while optimizing specific criteria such as path length, energy consumption, or smoothness [7]. This process typically assumes prior knowledge of the environment, which may be static or dynamically updated during operation.

In contrast, obstacle avoidance involves the real-time detection and circumvention of unforeseen obstacles that may disrupt the planned trajectory. It emphasizes responsiveness and adaptability, often relying on sensor feedback to modify motion commands during execution [8].

Additional distinctions exist between global planning, which considers the entire environment map, and local planning, which focuses on a robot's immediate surroundings [9]. Furthermore, configuration space (*C*-space) is a conceptual representation in which each point corresponds to a unique state of the robot, enabling geometric reasoning about feasible and infeasible paths.

B. *Problem Formulation*

The path planning and obstacle avoidance problem in mobile robotics can be formulated as a constrained optimization task in a configuration space C, where each configuration $q \in C$ represents a unique state of the robot [10]. The goal is to compute a feasible path $\tau(t):[0,T] \rightarrow C_{free}$, such that:

$$\tau(0) = q_{start}, \quad \tau(T) = q_{goal} \tag{1}$$

$$\tau(t) \in C_{free}, \quad \forall t \in [0,T] \tag{2}$$

Here, $C_{free} \subset C$ denotes the collision-free subset of the configuration space, and $C_{obs} \subset C$ denotes the obstacle region such that $C = C_{free} \cup C_{obs}$ and $C_{free} \cap C_{obs} = \neg 0$ [11].

The cost of a trajectory is defined by a function:

$$J[\tau] = \int_0^T L(\tau(t), \dot{\tau}(t))dt \tag{3}$$

where L is a Lagrangian representing the cost associated with motion, such as energy or time [12]. The problem is to find a trajectory τ^* that minimizes $J[\tau]$ while satisfying kinematic, dynamic, and environmental constraints.

C. *Survey Methodology*

To enhance the transparency and reproducibility of this review, we detail below the methodology employed for searching, selecting, and analyzing the literature included in this survey.

1) *Search Strategy and Data Sources*

The literature was gathered from reputable academic databases, including IEEE Xplore, ScienceDirect, SpringerLink, MDPI, and arXiv, which are widely recognized for hosting high-quality publications in robotics and artificial intelligence. The search was conducted using combinations of the following keywords: path planning, obstacle avoidance, mobile robots, motion planning, reactive navigation, learning-based planning, and robot navigation systems. Boolean operators (AND, OR) and filters were applied to refine the results to include only peer-reviewed articles, conference proceedings, and surveys.

2) *Selection Criteria*

We focused on papers published between January 2021 and August 2025 to ensure the review reflects the most recent advances. Articles were selected based on the following inclusion criteria:

- Relevance to mobile robot navigation (not limited to Unmanned Aerial Vehicles (UAVs) or underwater robots).
- Clear technical contribution to path planning or obstacle avoidance.
- Inclusion of experimental results, case studies, or comparative evaluations.
- Duplicates, non-English publications, short abstracts, and articles lacking technical depth were excluded.

3) Survey Scope

In total, over 350 papers were initially screened, from which 44 core references were selected and analyzed in depth. These references represent a balanced distribution across classical, learning-based, and integrated navigation approaches, ensuring comprehensive coverage of the topic.

III. CLASSIFICATION OF PATH PLANNING TECHNIQUES

This section provides a comprehensive classification of path planning techniques employed in mobile robotics, highlighting the evolution from rule-based to learning-driven approaches. These techniques are organized into four primary categories: classical, sampling-based, optimization-based, and learning-based methods. Each category addresses specific challenges in robot navigation, such as computational complexity, environmental uncertainty, and adaptability to dynamic conditions. The categorization also reflects the trade-offs between completeness, optimality, and real-time feasibility, which are critical when selecting an appropriate planning strategy for a given application. Figure 1 presents a taxonomy of these techniques, offering a visual overview of their relationships, algorithmic representatives, and operational characteristics.

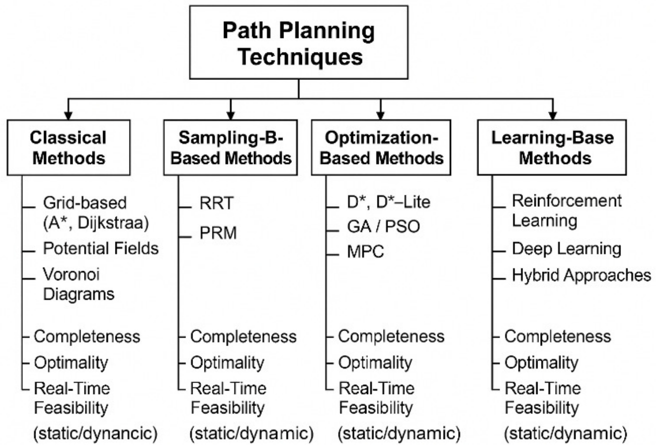

Fig. 1. Taxonomy of path planning techniques in mobile robotics.

A. Classical Approaches

Classical path planning techniques rely on deterministic algorithms that operate on discrete or continuous representations of the environment. Grid-based methods such as Dijkstra's algorithm and A* search, offer complete and optimal solutions on discretized maps, but they suffer from high computational costs in large or high-resolution environments [12]. Potential field methods treat the robot as a particle influenced by artificial attractive and repulsive forces, but they often fall into local minima [13]. Voronoi diagrams generate paths along the medial axis, maximizing clearance from obstacles, although their complexity increases with irregular boundaries [14]. These methods form the foundational basis for subsequent algorithmic advancements.

B. Sampling-Based Approaches

Sampling-based planners, such as Rapidly-exploring Random Trees (RRT) and Probabilistic Roadmaps (PRM), are designed to address high-dimensional and continuous configuration spaces where explicit representation is infeasible. RRT incrementally builds a tree rooted at the start configuration by sampling random states and connecting them with feasible motions, offering fast exploration but lacking optimality [15]. PRM, in contrast, constructs a roadmap by randomly sampling nodes and connecting collision-free edges, suitable for multi-query problems [16]. These methods do not guarantee completeness but are probabilistically complete, meaning the probability of finding a path increases with computation time.

C. Optimization-Based Approaches

Optimization-based planners model the path planning task as a cost minimization problem. D and D-Lite** are incremental search algorithms that dynamically update paths in changing environments, making them suitable for real-time applications [17]. Genetic Algorithms (GA) and Particle Swarm Optimization (PSO) apply bio-inspired heuristics to evolve candidate paths toward optimality through generations of recombination and mutation or social influence, respectively [18]. Model Predictive Control (MPC) formulates a finite-horizon optimization problem at each step, minimizing a cost function subject to system dynamics and constraints:

$$\min_{u(t)} \int_0^T L\left(x(t),u(t)\right)dt \quad s.t. \quad \dot{x} = f\left(x,u\right) \tag{4}$$

MPC offers smooth and adaptive trajectories but requires high computational resources [19].

D. Learning-Based Approaches

Learning-based path planners leverage experience or data to generate efficient and adaptive trajectories. Reinforcement learning enables agents to learn optimal policies through interaction with the environment, using reward signals to guide learning [20]. This approach is effective in dynamic or partially known environments but demands extensive training. Deep learning-based methods learn mappings from sensory inputs to actions or trajectories, facilitating end-to-end planning in complex domains. Hybrid systems combine learned models with classical planners to enhance generalization and safety. However, these approaches often lack interpretability and may struggle with generalization beyond trained scenarios, highlighting the need for robust validation frameworks.

E. Comparison and Trade-Offs

Each path planning strategy exhibits unique trade-offs in terms of performance, generalization, and computational efficiency. Classical methods are fast and predictable but limited in scalability. Sampling-based planners excel in high-dimensional spaces but can be suboptimal. Optimization-based methods balance adaptivity and optimality but incur computational overhead. Learning-based approaches offer flexibility in unstructured environments, yet demand significant training and may lack guarantees. Table I provides a comparative analysis of these techniques across key metrics, including completeness, optimality, scalability, and real-time applicability, helping inform appropriate algorithm selection based on robotic application requirements.

F. Interpretation of Comparative Evaluation Metrics in Path Planning

In the context of robotic path planning, evaluation metrics such as completeness, optimality, and scalability serve as fundamental indicators for assessing algorithm performance across varied environments and constraints:

- Completeness denotes an algorithm's ability to guarantee a path solution if one exists. Deterministic algorithms like A* and Dijkstra ensure completeness by exhaustively exploring the configuration space [2, 7]. Conversely, sampling-based methods such as RRT and PRM offer probabilistic completeness, which implies a solution is likely but not guaranteed in finite time [8, 10].
- Optimality refers to the algorithm's capability to find the most cost-efficient path, typically with respect to distance, energy, or time. Classical algorithms like Dijkstra and A*, when configured with admissible heuristics, yield optimal solutions. MPC [16] also maintains high optimality by solving constrained optimization problems in real time. In contrast, reactive strategies such as potential fields may become trapped in local minima, resulting in suboptimal paths.
- Scalability reflects an algorithm's adaptability to complex, high-dimensional environments. Sampling-based methods (RRT/PRM) and learning-based approaches excel in scalability due to their ability to generalize across larger state spaces. Classical algorithms, despite their optimality and completeness, often struggle with scalability due to increased computational overhead as the map size grows.

As Table I illustrates, classical methods (e.g., A*, Dijkstra) [2, 7] are best suited for structured, static environments, where optimality and completeness are prioritized over scalability and adaptability. Potential field [3, 5, 14, 15] and Voronoi-based techniques [4, 9] are lightweight but lack robustness in complex scenarios. Sampling-based methods like RRT/PRM are favorable for real-time planning in expansive, dynamic environments, albeit at the expense of guaranteed optimality.

Among adaptive approaches, D* and D*-Lite strike a balance between completeness and adaptability [11, 13], making them well-suited for dynamic or partially known environments. MPC stands out for its integration of predictive control and constraints, achieving high performance across all criteria except computational load [16]. Learning-based methods, including reinforcement learning and deep learning [6, 18, 19], show strong promise due to their scalability and adaptability, although training complexity and interpretability remain challenges.

Hybrid systems aim to combine the strengths of classical and learning-based models, offering a balanced trade-off across all dimensions [20]. Overall, Table I highlights that no single method dominates across all metrics, emphasizing the importance of context-aware algorithm selection in mobile robotic applications.

The landscape of path planning techniques in mobile robotics is rich and diverse, with each category offering distinct advantages depending on task requirements and environmental complexity. From deterministic classical methods to adaptive learning-based strategies, the choice of algorithm must balance factors such as efficiency, reliability, and scalability. This classification not only aids in understanding existing approaches but also provides a foundation for developing hybrid systems that leverage the strengths of multiple paradigms for enhanced navigation performance.

TABLE I. COMPARATIVE ANALYSIS OF PATH PLANNING TECHNIQUES

Method category	**Completeness**	**Optimality**	**Computation time**	**Scalability**	**Adaptability to dynamic environments**	**Reference**
Classical (e.g., A*, Dijkstra)	High	High	High (for large maps)	Low	Low	[2, 7]
Potential fields	Low	Low (local minima)	Low	Medium	Medium	[3, 5, 14, 15]
Voronoi diagrams	High	Medium	Medium	Low	Low	[4, 9]
RRT / PRM	Probabilistic	Low to medium	Low	High	Medium	[8, 10]
D, D-Lite**	High	Medium	Medium	Medium	High	[11, 13]
GA / PSO	Medium	Medium to high	Medium to high	Medium	Medium	[14, 15]
MPC	High	High	High	Medium	High	[16]
Reinforcement learning	Variable	Variable	High (training)	High	High	[18]
Deep learning	Variable	Variable	Low (inference)	High	High	[6, 19]
Hybrid systems	Medium to high	Medium to high	Medium	High	High	[20]

IV. OBSTACLE DETECTION AND AVOIDANCE TECHNIQUES

This section explores the diverse range of techniques used for obstacle detection and avoidance, a critical capability for ensuring safe and reliable navigation in mobile robotics. As robots operate in increasingly complex and dynamic environments, the ability to perceive, interpret, and respond to obstacles in real time has become essential. This section categorizes obstacle avoidance strategies into sensor-based perception systems, reactive mechanisms, predictive modeling approaches, and learning-based frameworks. Each category offers unique advantages depending on the application context, environmental uncertainty, and computational constraints. Figure 2 provides a high-level overview of these strategies, showing the flow from sensor inputs through various decision-making layers to motion planning outputs.

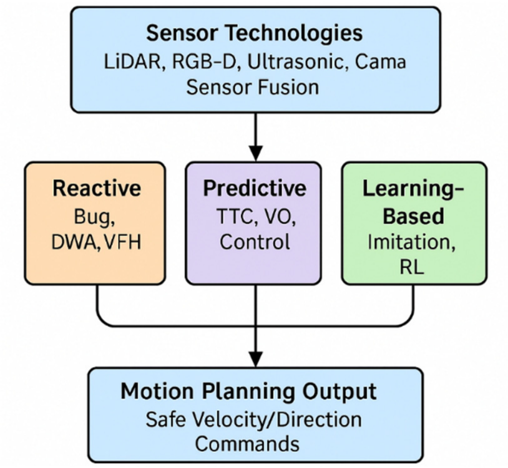

Fig. 2. Pipeline representation of obstacle detection and avoidance strategies in mobile robotics.

A. Sensor Technologies

Effective obstacle detection in mobile robotics relies heavily on sensor technologies. LiDAR provides high-resolution 3D point clouds, enabling accurate distance measurements in structured and unstructured environments [21]. RGB-D cameras offer depth maps combined with visual context, useful for semantic understanding [22]. Ultrasonic sensors are low-cost and ideal for short-range detection but suffer from poor angular resolution [23]. Vision-based systems, particularly those powered by convolutional neural networks, can recognize object types and locations [24]. Sensor fusion integrates data from multiple modalities to improve perception robustness, often using techniques such as Kalman filtering or Bayesian inference [25], enhancing obstacle recognition in dynamic and cluttered environments.

B. Reactive Obstacle Avoidance

Reactive techniques generate immediate responses to sensed obstacles without global planning. Bug algorithms follow wall-following or boundary-tracing strategies, enabling basic navigation around obstacles but can be inefficient [26]. The Dynamic Window Approach (DWA) computes a robot's velocity commands within a feasible dynamic window by optimizing a cost function:

$$\begin{aligned} G(u,w) &= \alpha \cdot heading(v,w) \\ &+\beta \cdot clearance(v,w) \\ &-\gamma \cdot velocity(v,w) \end{aligned} \tag{5}$$

where v and w are linear and angular velocities [27]. Vector Field Histogram (VFH) builds histograms from obstacle densities to find safe navigation directions [28]. Reactive methods are fast and simple but often lack global awareness.

C. Predictive and Model-Based Approaches

Model-based methods incorporate future state predictions to avoid collisions proactively. The Time-to-Collision (TTC) metric estimates the remaining time before impact with an obstacle based on relative velocity:

$$TTC = \frac{d}{\left|v_{rel}\right|} \tag{6}$$

where d is distance and v_{rel} is relative velocity [29]. Velocity Obstacle (VO) methods define velocity sets that lead to collisions and compute evasive maneuvers in real time [30]. Nonlinear control-based strategies apply control laws (e.g., Lyapunov functions) to ensure stable and collision-free trajectories [31]. These approaches are effective in dynamic environments, especially when obstacle trajectories are predictable, but require accurate models and more computation.

D. Learning-Based Avoidance

Learning-based techniques allow robots to develop obstacle avoidance behaviors from data rather than explicit programming. Imitation learning trains agents to mimic expert demonstrations, enabling reactive behaviors in complex scenes without extensive engineering [32]. In contrast, reinforcement learning enables robots to learn navigation policies by trial and error, receiving rewards for safe and efficient movement through obstacle-laden environments [33]. These methods excel in unstructured or unknown settings and can generalize to new scenarios. However, they require large amounts of training data or simulation time and often lack theoretical safety guarantees, which limits their deployment in safety-critical domains.

E. Benchmarks and Evaluation Metrics

Evaluating obstacle avoidance strategies necessitates standardized benchmarks and metrics. Key indicators include navigation safety (e.g., number of collisions per trajectory), real-time responsiveness (e.g., reaction time to dynamic obstacles), and computational load (e.g., CPU/GPU usage during planning) [21, 26, 30]. Public benchmarks like the Gazebo simulations, TurtleBot real-world navigation tests, and datasets such as KITTI and ScanNet provide controlled settings for fair comparisons. Additionally, robustness to sensor noise, adaptability to unforeseen changes, and energy efficiency are increasingly important metrics for modern robotic systems, especially in real-time embedded platforms and resource-constrained environments.

V. INTEGRATED PATH PLANNING AND OBSTACLE AVOIDANCE SYSTEMS

This section focuses on the integration of path planning and obstacle avoidance into cohesive robotic navigation systems capable of operating in real-world environments. Unlike isolated algorithms, integrated systems coordinate global trajectory generation with local obstacle responsiveness, while adhering to real-time processing constraints and hardware limitations.

This integration ensures robust performance across diverse scenarios, from structured indoor environments to dynamic outdoor settings. The design typically follows a hierarchical architecture, combining strategic, tactical, and reactive control layers, each responsible for different aspects of navigation. Figure 3 presents an overview of this architecture, illustrating the flow of information from sensor inputs to motion control, and showcasing practical implementations in various robotic platforms.

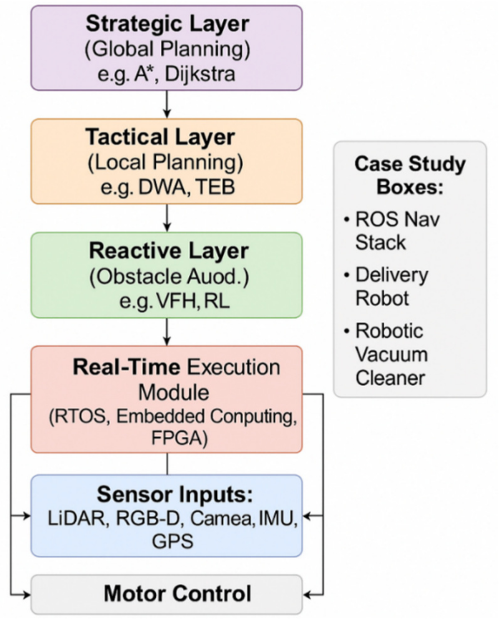

Fig. 3. Integrated path planning and obstacle avoidance architecture in mobile robots.

A. *Fusion of Global and Local Planning*

Modern robotic navigation systems frequently combine global and local planning to balance long-term path optimality with real-time responsiveness. Global planners compute an initial trajectory using environmental maps, whereas local planners adapt this path in response to dynamic obstacles. This dual-layered approach enhances robustness, enabling navigation through uncertain environments [34]. A common implementation is the use of A* for global path generation and DWA or Timed Elastic Band (TEB) for local adjustments. The integration is often guided by a cost function:

$$J_{total} = \lambda_g J_{global} + \lambda_l J_{local} \tag{7}$$

where λ_g and λ_l are weighting factors balancing global and local priorities.

B. *Layered Architectures and Hierarchical Control*

Layered and hierarchical control architectures structure robot behavior into distinct levels, improving modularity and scalability. Typically, the system consists of a strategic layer (mission planning), a tactical layer (path planning), and a reactive layer (obstacle avoidance) [35]. This separation allows high-level decisions to guide motion goals, whereas low-level controllers respond rapidly to environmental changes. Each layer communicates through structured interfaces, ensuring that commands remain consistent and context-aware. Such designs are prevalent in both research and commercial platforms due to their ability to handle multiple objectives and heterogeneous data streams in parallel, while maintaining real-time safety guarantees and behavior generalization.

C. *Real-Time Constraints and Embedded Systems*

Path planning and avoidance systems must satisfy strict real-time constraints, particularly in embedded platforms with limited processing capabilities. Algorithms must operate within bounded execution windows to ensure safe and timely decisions. Trade-offs arise between planning depth and computational latency, especially in dynamic environments [36]. Embedded systems require optimized implementations, often using techniques like pre-computed lookup tables, efficient data structures, and parallel processing on GPUs or FPGAs. Real-time operating systems (RTOS) also play a vital role in maintaining deterministic behavior and managing priority tasks [37]. Failure to meet timing deadlines can result in unsafe behaviors, especially in high-speed or human-centric robotic applications.

D. *Case Studies: ROS Navigation Stack, Autonomous Delivery Robots, Robotic Vacuums*

The ROS Navigation Stack exemplifies an open-source, integrated framework combining global planning, local obstacle avoidance, and real-time sensor updates through modular nodes [38]. It employs costmaps, localization via Adaptive Monte Carlo Localization (AMCL), and a layered planning hierarchy to enable robust autonomous navigation. Autonomous delivery robots, such as sidewalk bots, apply similar architectures but emphasize pedestrian interaction, crosswalk compliance, and cloud-based map updates. Robotic vacuum cleaners represent embedded, cost-constrained implementations with onboard Simultaneous Localization and Mapping (SLAM), reactive path planning, and minimal computation overhead [39]. These case studies demonstrate how integrated systems vary significantly in complexity depending on the operational environment and performance requirements, yet share common structural principles.

VI. CHALLENGES AND OPEN RESEARCH PROBLEMS

A. *Planning under Uncertainty*

One of the fundamental challenges in robotic navigation is planning under uncertainty, where the robot lacks complete or accurate information about the environment or its own state. Sources of uncertainty include sensor noise, dynamic obstacles, localization drift, and imperfect maps [40]. Traditional deterministic planners struggle in such conditions. Probabilistic methods, such as Partially Observable Markov Decision Processes (POMDPs), attempt to model uncertainty explicitly but are computationally intensive. Efficient approximations, risk-aware planning, and robust decision-making strategies are ongoing areas of research. A key open problem is designing scalable, uncertainty-resilient algorithms that ensure safety and goal achievement under ambiguous observations.

B. Real-Time Planning in Dynamic Environments

Real-time path planning in dynamic environments demands continuous adaptation to moving obstacles, changing terrain, and variable goals. Algorithms must generate safe and feasible trajectories within strict timing constraints, often with incomplete future knowledge [41]. Existing solutions include dynamic re-planning (e.g., D*, TEB), reactive policies, and predictive models. However, achieving both low latency and optimality remains a trade-off. Balancing computational efficiency with responsiveness is a key challenge, particularly for embedded systems. Open research problems include developing lightweight models capable of predicting motion patterns, integrating temporal reasoning, and ensuring stability in environments where obstacle behavior is non-deterministic or adversarial.

C. Human-Aware Navigation and Social Compliance

As robots increasingly operate in human-populated environments, navigation systems must incorporate models of human behavior and social norms. Human-aware navigation involves predicting pedestrian trajectories, avoiding uncomfortable proximity, and adhering to socially acceptable motion patterns (e.g., yielding, queueing) [42]. Failure to do so can lead to unsafe or disruptive interactions. Current approaches rely on crowd simulation, intent prediction, and reinforcement learning from human demonstrations. However, modeling human-robot interactions with cultural sensitivity, contextual awareness, and explainability remains an open challenge. Future work must bridge robotics with cognitive science to enable socially compliant navigation in shared spaces.

D. Multi-Robot Coordination and Swarm Planning

Coordinating multiple robots to achieve joint objectives introduces spatial, temporal, and communication complexities. Multi-robot planning involves collision avoidance, task allocation, formation control, and shared mapping [43]. Centralized strategies offer optimality but are less scalable, whereas decentralized methods face coordination and conflict resolution issues. Swarm robotics adds further complexity, requiring self-organization and emergent behavior based on local rules [44]. Current limitations include communication delays, global consistency maintenance, and resilience to individual robot failures. Open research challenges lie in designing scalable, fault-tolerant coordination protocols that leverage distributed learning, consensus algorithms, and bio-inspired strategies for robust collective navigation.

E. Generalization and Transfer Learning

Robotic navigation systems often suffer from overfitting to specific environments, limiting their deployment across diverse or unseen settings. Generalization refers to the robot's ability to apply learned behaviors to new tasks or domains, whereas transfer learning involves adapting knowledge from one environment to another [45]. Deep learning models trained in simulation frequently fail in real-world conditions due to domain gaps [46]. Techniques such as domain adaptation, meta-learning, and sim-to-real transfer aim to address this issue, but challenges persist [47]. Open problems include minimizing data requirements, maintaining safety during adaptation, and ensuring that transferred policies remain interpretable and robust across varying scenarios [48].

F. Knowledge Gaps and Contribution of this Survey

While numerous review articles have explored aspects of path planning or obstacle avoidance independently, there remains a lack of integrated surveys that systematically address both areas in the context of real-time mobile robotics, particularly under modern constraints such as embedded system deployment, dynamic multi-agent environments, and learning-based control. Previous works have often focused exclusively on classical or sampling-based methods without considering the rapid evolution of learning-based strategies or hybrid architectures combining global and local planning.

Moreover, existing surveys frequently overlook the challenges posed by uncertainty, human-aware navigation, and sim-to-real transfer, which are becoming increasingly critical as robots transition from controlled lab environments to unstructured real-world settings. Few reviews provide detailed comparisons of system-level implementations or reference up-to-date open-source navigation frameworks, such as the ROS2 Navigation Stack.

This paper addresses these gaps by offering a unified framework that categorizes algorithms, evaluates their performance across key criteria (completeness, scalability, real-time feasibility), and presents integrated architectures supported by recent case studies. By incorporating research published between 2021 and 2025, this survey reflects the most recent advancements and provides a timely reference for researchers and practitioners working at the intersection of autonomy, artificial intelligence, and real-world mobile robotics.

VII. CONCLUSION

This survey has provided a structured and comprehensive analysis of path planning and obstacle avoidance techniques in mobile robotics, integrating both foundational and emerging approaches. The paper systematically categorized existing methods into classical, sampling-based, optimization-based, and learning-based paradigms, offering comparative insights into their operational principles, computational efficiency, and application suitability. In parallel, a detailed taxonomy of obstacle avoidance strategies—ranging from reactive and predictive models to learning-enabled systems—was presented, highlighting their role in real-time navigation under varying environmental constraints. A distinguishing contribution of this survey lies in its examination of integrated planning architectures, particularly those combining global and local planners within hierarchical frameworks that adhere to real-time and embedded-system constraints.

In addition to summarizing the state of the art, the paper identified critical knowledge gaps, including the need for robust planning under uncertainty, socially aware navigation in human-centric environments, scalable multi-robot coordination, and the generalization of learned policies across domains. To address these, future research should focus on developing hybrid models that integrate symbolic reasoning with deep learning, adaptive frameworks capable of sim-to-real transfer, and safety-assured planning mechanisms in dynamic, unstructured contexts. By bridging algorithmic developments with system-level implementations, this survey contributes a current and forward-looking reference for researchers and practitioners working toward the next generation of intelligent autonomous mobile robots.

ACKNOWLEDGMENT

This work was supported by the Science Committee of the Ministry of Higher Education and Science of the Republic of Kazakhstan within the framework of grant AP19679910, "Development of an experimental model of an autonomous unmanned mobile robot with off-road capability," International Engineering Technological University, Almaty.

REFERENCES

[1] M. A. A. Noman *et al.*, "A computer vision-based lane detection technique using gradient threshold and hue-lightness-saturation value for an autonomous vehicle," *International Journal of Electrical and Computer Engineering*, vol. 13, no. 1, pp. 347–357, Feb. 2023, https://doi.org/10.11591/ijece.v13i1.pp347-357.

[2] K. Katona, H. A. Neamah, and P. Korondi, "Obstacle Avoidance and Path Planning Methods for Autonomous Navigation of Mobile Robot," *Sensors*, vol. 24, no. 11, Jun. 2024, Art. no. 3573, https://doi.org/10.3390/s24113573.

[3] M. S. Qasim, A. B. Ayoub, and A. I. Abdulla, "NMPC Based-Trajectory Tracking and Obstacle Avoidance for Mobile Robots," *International Journal of Robotics and Control Systems*, vol. 4, no. 4, pp. 2026–2040, Nov. 2024, https://doi.org/10.31763/ijrcs.v4i4.1605.

[4] A. Gharbi, "A dynamic reward-enhanced Q-learning approach for efficient path planning and obstacle avoidance in mobile robotics," *Applied Computing and Informatics*, Jan. 2024, https://doi.org/10.1108/ACI-10-2023-0089.

[5] L. C. Sousa *et al.*, "Obstacle Avoidance Technique for Mobile Robots at Autonomous Human-Robot Collaborative Warehouse Environments," *Sensors*, vol. 25, no. 8, Apr. 2025, Art. no. 2387, https://doi.org/10.3390/s25082387.

[6] B. Omarov *et al.*, "Electronic Stethoscope for Heartbeat Abnormality Detection," in *Smart Computing and Communication: 5th International Conference*, Paris, France, 2020, pp. 248–258, https://doi.org/10.1007/978-3-030-74717-6_26.

[7] K. Cai, C. Wang, J. Cheng, C. W. D. Silva, and M. Q.-H. Meng, "Mobile Robot Path Planning in Dynamic Environments: A Survey." arXiv, Mar. 22, 2021, https://doi.org/10.48550/arXiv.2006.14195.

[8] L. Dong, Z. He, C. Song, and C. Sun, "A review of mobile robot motion planning methods: from classical motion planning workflows to reinforcement learning-based architectures." arXiv, Feb. 23, 2022, https://doi.org/10.48550/arXiv.2108.13619.

[9] P. Wenzel, T. Schön, L. Leal-Taixé, and D. Cremers, "Vision-Based Mobile Robotics Obstacle Avoidance With Deep Reinforcement Learning." arXiv, Mar. 08, 2021, https://doi.org/10.48550/arXiv.2103.04727.

[10] A.-T. Nguyen and C.-T. Vu, "Obstacle Avoidance for Autonomous Mobile Robots Based on Mapping Method." arXiv, Sep. 14, 2021, https://doi.org/10.48550/arXiv.2109.06773.

[11] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, Jun. 2011, https://doi.org/10.1177/0278364911406761.

[12] B. Omarov, M. Baikuvekov, D. Sultan, N. Mukazhanov, M. Suleimenova, and M. Zhekambayeva, "Ensemble Approach Combining Deep Residual Networks and BiGRU with Attention Mechanism for Classification of Heart Arrhythmias," *Computers, Materials & Continua*, vol. 80, no. 1, pp. 341–359, Jul. 2024, https://doi.org/10.32604/cmc.2024.052437.

[13] K. Karur, N. Sharma, C. Dharmatti, and J. E. Siegel, "A Survey of Path Planning Algorithms for Mobile Robots," *Vehicles*, vol. 3, no. 3, pp. 448–468, Sep. 2021, https://doi.org/10.3390/vehicles3030027.

[14] Y. Tang, M. A. Zakaria, and M. Younas, "Path Planning Trends for Autonomous Mobile Robot Navigation: A Review," *Sensors*, vol. 25, no. 4, Feb. 2025, Art. no. 1206, https://doi.org/10.3390/s25041206.

[15] O. Misir and M. Celik, "Visual-based obstacle avoidance method using advanced CNN for mobile robots," *Internet of Things*, vol. 31, May 2025, Art. no. 101538, https://doi.org/10.1016/j.iot.2025.101538.

[16] X. Dong, Y. Wang, C. Fang, K. Ran, and G. Liu, "FHQ-RRT*: An Improved Path Planning Algorithm for Mobile Robots to Acquire High-Quality Paths Faster," *Sensors*, vol. 25, no. 7, Jul. 2025, Art. no. 2189, https://doi.org/10.3390/s25072189.

[17] F. de A. M. Pimentel and P. T. Aquino-Jr, "Evaluation of ROS Navigation Stack for Social Navigation in Simulated Environments," *Journal of Intelligent & Robotic Systems*, vol. 102, no. 4, Jul. 2021, Art. no. 87, https://doi.org/10.1007/s10846-021-01424-z.

[18] A. Thirugnanam, J. Zeng, and K. Sreenath, "Safety-Critical Control and Planning for Obstacle Avoidance between Polytopes with Control Barrier Functions." arXiv, May 31, 2022, https://doi.org/10.48550/arXiv.2109.12313.

[19] X. Xiao, B. Liu, G. Warnell, and P. Stone, "Motion Planning and Control for Mobile Robot Navigation Using Machine Learning: a Survey." arXiv, Feb. 26, 2022, https://doi.org/10.48550/arXiv.2011.13112.

[20] Í. Elguea-Aguinaco, I. Inziarte-Hidalgo, S. Bøgh, and N. Arana-Arexolaleiba, "A Review on Reinforcement Learning for Motion Planning of Robotic Manipulators," *International Journal of Intelligent Systems*, vol. 2024, 2024, Art. no. 1636497, https://doi.org/10.1155/int/1636497.

[21] R. Raj and A. Kos, "Dynamic Obstacle Avoidance Technique for Mobile Robot Navigation Using Deep Reinforcement Learning," *International Journal of Emerging Trends in Engineering Research*, vol. 11, no. 9, pp. 307–314, Sep. 2023, https://doi.org/10.30534/ijeter/2023/031192023.

[22] L. Da, J. Turnau, T. P. Kutralingam, A. Velasquez, P. Shakarian, and H. Wei, "A Survey of Sim-to-Real Methods in RL: Progress, Prospects and Challenges with Foundation Models." arXiv, Mar. 08, 2025, https://doi.org/10.48550/arXiv.2502.13187.

[23] J. K. Verma and V. Ranga, "Multi-Robot Coordination Analysis, Taxonomy, Challenges and Future Scope," *Journal of Intelligent & Robotic Systems*, vol. 102, no. 1, Apr. 2021, Art. no. 10, https://doi.org/10.1007/s10846-021-01378-2.

[24] Y. Huang, S. Huang, H. Wang, and R. Meng, "3D Path Planning and Obstacle Avoidance Algorithms for Obstacle-Overcoming Robots." arXiv, Sep. 02, 2022, https://doi.org/10.48550/arXiv.2209.00871.

[25] Y. Wang, C. Fu, R. Huang, K. Tong, Y. He, and L. Xu, "Path planning for mobile robots in greenhouse orchards based on improved A* and fuzzy DWA algorithms," *Computers and Electronics in Agriculture*, vol. 227, no. 2, Dec. 2024, Art. no. 109598, https://doi.org/10.1016/j.compag.2024.109598.

[26] T. Mohanraj, T. Dinesh, B. Guruchandhramavli, S. Sanjai, and B. Sheshadhri, "Mobile robot path planning and obstacle avoidance using hybrid algorithm," *International Journal of Information Technology*, vol. 15, no. 8, pp. 4481–4490, Dec. 2023, https://doi.org/10.1007/s41870-023-01475-5.

[27] G. K. Teja, P. K. Mohanty, and S. Das, "Review on path planning methods for mobile robot," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 239, no. 14, pp. 5547–5580, Jul. 2025, https://doi.org/10.1177/09544062251330083.

[28] J. C. Tejada, A. Toro-Ossaba, A. López-Gonzalez, E. G. Hernandez-Martinez, and D. Sanin-Villa, "A Review of Multi-Robot Systems and Soft Robotics: Challenges and Opportunities," *Sensors*, vol. 25, no. 5, Mar. 2025, Art. no. 1353, https://doi.org/10.3390/s25051353.

[29] L. Antonyshyn, J. Silveira, S. Givigi, and J. Marshall, "Multiple Mobile Robot Task and Motion Planning: A Survey," *ACM Computing Surveys*, vol. 55, no. 10, Feb. 2023, Art. no. 213, https://doi.org/10.1145/3564696.

[30] S. Nahavandi *et al.*, "A Comprehensive Review on Autonomous Navigation." arXiv, Dec. 24, 2022, https://doi.org/10.48550/arXiv.2212.12808.

[31] K. Li *et al.*, "Deep Reinforcement Learning-based Obstacle Avoidance for Robot Movement in Warehouse Environments." arXiv, Sep. 23, 2024, https://doi.org/10.48550/arXiv.2409.14972.

[32] N. Ü. Akmandor, H. Li, G. Lvov, E. Dusel, and T. Padır, "Deep Reinforcement Learning based Robot Navigation in Dynamic Environments using Occupancy Values of Motion Primitives." arXiv, Aug. 17, 2022, https://doi.org/10.48550/arXiv.2208.08034.

[33] M. Badamasi Aremu, I. K. Kabir, G. Ahmed, and S. El-Ferik, "Autonomous Mobile Robot Path Planning Techniques—A Review: Classical and Heuristic Techniques," *IEEE Access*, vol. 13, pp. 117999–118022, 2025, https://doi.org/10.1109/ACCESS.2025.3579863.

[34] Í. R. da Costa Barros and T. P. Nascimento, "Robotic Mobile Fulfillment Systems: A survey on recent developments and research opportunities," *Robotics and Autonomous Systems*, vol. 137, Mar. 2021, Art. no. 103729, https://doi.org/10.1016/j.robot.2021.103729.

[35] H. Wang, L. He, S. Zhang, R. Bai, and Y. Wang, "Mobile Robot Path Planning Considering Obstacle Gap Features," *Applied Sciences*, vol. 15, no. 11, Jun. 2025, Art. no. 5979, https://doi.org/10.3390/app15115979.

[36] L. Yan, T. Stouraitis, and S. Vijayakumar, "Decentralized Ability-Aware Adaptive Control for Multi-Robot Collaborative Manipulation," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2311–2318, Apr. 2021, https://doi.org/10.1109/LRA.2021.3060379.

[37] R. Karthikeyan and B. Sheela Rani, "An innovative approach for obstacle avoidance and path planning of mobile robot using adaptive deep reinforcement learning for indoor environment," *Knowledge-Based Systems*, vol. 326, Sep. 2025, Art. no. 114058, https://doi.org/10.1016/j.knosys.2025.114058.

[38] M. Dorigo, G. Theraulaz, and V. Trianni, "Swarm Robotics: Past, Present, and Future [Point of View]," *Proceedings of the IEEE*, vol. 109, no. 7, pp. 1152–1165, Jul. 2021, https://doi.org/10.1109/JPROC.2021.3072740.

[39] N.-T. Tran, T.-L. Bui, and V.-L. Trinh, "An Approach for Dynamic Obstacle Avoidance in Autonomous Mobile Robots Operating in Unstructured Indoor Environments," *Engineering, Technology & Applied Science Research*, vol. 15, no. 3, pp. 23506–23513, Jun. 2025, https://doi.org/10.48084/etasr.10999.

[40] W. Zhu, X. Guo, D. Owaki, K. Kutsuzawa, and M. Hayashibe, "A Survey of Sim-to-Real Transfer Techniques Applied to Reinforcement Learning for Bioinspired Robots," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 7, pp. 3444–3459, Jul. 2023, https://doi.org/10.1109/TNNLS.2021.3112718.

[41] Y. Wang, A. R. Srinivasan, Y. M. Lee, and G. Markkula, "Modeling Pedestrian Crossing Behavior: A Reinforcement Learning Approach With Sensory Motor Constraints," *IEEE Transactions on Intelligent Transportation Systems*, 2025, https://doi.org/10.1109/TITS.2025.3581693.

[42] L. H. Zain and R. E. Shalaby, "Real-Time Obstacle Avoidance for a Mobile Robot Using CNN-Based Sensor Fusion." arXiv, Sep. 09, 2025, https://doi.org/10.48550/arXiv.2509.08095.

[43] Q. Wu, X. Gong, K. Xu, D. Manocha, J. Dong, and J. Wang, "Towards Target-Driven Visual Navigation in Indoor Scenes via Generative Imitation Learning," *IEEE Robotics and Automation Letters*, vol. 6, no. 1, pp. 175–182, Jan. 2021, https://doi.org/10.1109/LRA.2020.3036597.

[44] A. B. Alshammari, "Dynamic Rewards in Reinforcement Learning for Robotic Navigation," *Engineering, Technology & Applied Science Research*, vol. 15, no. 4, pp. 25766–25771, Aug. 2025, https://doi.org/10.48084/etasr.11986.

[45] A. B. Altayeva, B. S. Omarov, A. Z. Aitmagambetov, B. B. Kendzhaeva, and M. A. Burkitbayeva, "Modeling and exploring base station characteristics of LTE mobile networks," *Life Science Journal*, vol. 11, no. 6, pp. 227–233, Jun. 2014.

[46] B. Omarov, A. Tursynova, and M. Uzak, "Deep Learning Enhanced Internet of Medical Things to Analyze Brain Computed Tomography Images of Stroke Patients," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 8, pp. 668–676, Aug. 2023, https://doi.org/10.14569/IJACSA.2023.0140874.

[47] B. Hahn, "Enhancing Obstacle Avoidance in Dynamic Window Approach via Dynamic Obstacle Behavior Prediction," *Actuators*, vol. 14, no. 5, May 2025, Art. no. 207, https://doi.org/10.3390/act14050207.

[48] M. Fachri, D. Prasetyo, F. A. Damastuti, N. Ramadhani, S. M. S. Nugroho, and M. Hariadi, "Crowd navigation for dynamic hazard avoidance in evacuation using emotional reciprocal velocity obstacles," *IAES International Journal of Artificial Intelligence*, vol. 13, no. 2, pp. 1371–1379, Jun. 2024, https://doi.org/10.11591/ijai.v13.i2.pp1371-1379.