Navigation of Mobile Robot

Subjects: Physics, Applied
Contributor: Kailash Kumar Borkar, Turki Aljrees, Saroj Kumar Pandey, Ankit Kumar, Mukesh Kumar Singh, Anurag Sinha, Kamred Udham Singh, Vandana Sharma

Mobile robot navigation is a study area with several potential applications, both military and nonmilitary. Nonetheless, a great deal of problems remain, and many of them will probably need to be addressed using either a novel theory or another approach. The autonomous navigation of mobile robots has been studied by several researchers. Robots may encounter static interior environments or dynamic environments that change swiftly. The mobile robot's goals may include reaching a particular place, following a predetermined course, or mapping out a space for later use. Many studies have been conducted throughout the world regarding this. Different methodologies were proposed on this topic, each with its advantages and disadvantages based on the recursive research. Navigation of a mobile robot involves determining the robot's trajectory through an environment to reach a desired goal location while avoiding obstacles.

Keywords: dynamic analysis ; path planning ; navigation ; WMR

1. Indoor Navigation

It is conducted groundbreaking research on robotic vehicle navigation, which highlighted the importance of incorporating knowledge of the device's internal environment to navigate effectively. Early vision systems developed for mobile robot navigation relied heavily on the geometry of space and other metric data to drive the vision processes and enable self-localization. To model the internal environment, extremely complex algorithms were used to create CAD models. It is believed that these programmers can be grouped into three general categories [1].

1.1. Map-Based Navigation

There are navigation systems that rely on topological or geometrical maps of the environment, known as model-based or map-based navigation systems. These systems typically use a geometric model or a topological representation of the surroundings to plan the movement of the robot. In a geometric model-based system, the environment is represented as a 3D geometric model using point clouds or meshes, which are built using the robot's sensors, such as cameras or LiDAR. The robot can then plan a collision-free path through the model using algorithms like A*, RRT, or PRM. In a topological-map-based system, the environment is represented as a graph where nodes represent significant sites and edges represent routes between them. The robot's starting point and goal are connected using a series of nodes and edges that avoid hazards. The Dijkstra's algorithm or A graph search algorithm can be used for this purpose. Model-based and map-based navigation systems have several advantages, including the ability to handle complex environments with many obstacles, plan efficient paths that avoid obstacles, and adapt to changes in the environment. They can also be used with other robot capabilities, such as object recognition or manipulation, to perform complex tasks.

However, these systems also have some limitations. They require accurate models of the environment, which can be time-consuming and difficult to create. They can also be sensitive to errors in the model, such as occlusions or inaccuracies in the sensor data. Finally, they may not be suitable for environments that are constantly changing or where the robot must navigate in real time.

1.2. Map-Building-Based Navigation

These systems use sensors to create their own topological or geometric representations of the world, which they subsequently use to aid with navigation, i.e., systems that use sensors as navigational aids. These systems frequently combine several sensors, such as cameras, LiDAR, and sonar, to provide a real-time map of the area. The robot plans its path and moves around the region using this map. The sensors aboard the robot are utilized in a sensor-based navigation system to gather information about the surroundings. A topological or geometric representation of the environment that is utilized for navigation may be made using this data. For example, a topological map might consist of nodes that represent significant locations in the environment, and edges that represent the paths between these locations. A geometric map might consist of a 3D point cloud that represents the environment's geometry. Once the map is created, the robot can use it to plan its path through the environment. This is typically carried out using algorithms such as A*, RRT, or PRM, which find a collision-free trajectory through the map. The robot's sensors are used to update the map in real time as the robot navigates through the environment, allowing it to respond to changes in the environment and avoid obstacles. A control method called fuzzy logic control (FLC) employs fuzzy sets and rules to make choices. Given that it permits imprecise or ambiguous inputs and outputs, it is very helpful when working with complicated or uncertain systems. A fuzzifier, a rule basis, and a fuzzified are the three primary parts of FLC. The rule base determines the output using a collection of IF–THEN rules, the fuzzifier transforms the fuzzy output into a crisp value, and the fuzzifier maps the input variables to fuzzy sets. Navigation is based on a representation of the environment in C. The firefly method (FA) is a metaheuristic optimization method that draws inspiration from firefly behavior. Finding a function's global optimum is one of its main applications in problem-solving. In FA, each firefly stands in for a potential remedy to the issue, and their behavior is determined by their allure and closeness to other fireflies. A firefly's attractiveness is defined by the objective function that is being optimized, and its proximity is determined by its location in the solution space. FLC and FA work well together to solve optimization and control issues. FLC, for instance, may be used to regulate a mobile robot's speed or steering angle, whereas FA can be used to tailor the FLC settings for a particular job or environment [2].

1.3. Mapless Navigation

These navigation systems merely distinguish items in the environment or follow those objects by creating movements based on visual observations, as opposed to using any explicit representation of the space in which navigation is to take place. The issue of a mobile robot traveling in an interior environment that is unknown or just partially understood was raised by [3]. Based on the integration of fundamental behaviors, a navigation strategy for uncharted territory was devised. The bulk of these qualities are achieved via fuzzy inference methods. Two separate methods of avoiding obstacles are included in the recommended navigator: one for convex obstacles and the other for concave ones. To develop basic behaviors like "reaching the middle of the collision-free space" and "wall-following", zero-order Takagi–Sugeno fuzzy inference systems are used. This method is fairly straightforward and intuitive, but one can never eliminate the possibility that the rules derived from straightforward human expertise are somewhat inadequate. It is possible to optimize the parameters of a fuzzy inference system by minimizing a cost function using a method based on a backpropagation-like algorithm. It derive set of rules from the experimental data without using any empirical methods, and it is especially crucial to keep in mind this last aspect. In the case of a partially known environment, a hybrid technique is used to benefit from both global and local navigation algorithms. A fuzzy inference method that compares the actual situation to one that has been previously remembered online is used to coordinate these tactics. To design the journey, an algorithm and visibility graph are employed. Fuzzy controllers are made available for both the virtual robot's planned path following in the hypothetical environment and the actual robot's navigation when the real environment is local compared to the recalled one. Both methods are employed with the help of the tiny mobile robot Khepera, which possesses ultrasonic sensors. To correctly maneuver a tiny, four-wheeled, indoor mobile robot in real time without prior knowledge of the surroundings, Jazayeriet al. [4] described the implementation of an intelligent system. Finding the robot's optimum route to its objective was achieved using a recurrent neural network. Using a laser range finder scenario and a modified dead reckoning method, a precise grid-based map was produced. The presentation of a motion control technique came last. The mobile robot Resquake used these strategies, and they were tested. Intelligent autonomous mobile robots must be able to perceive and recognize 3D inside spaces where they reside or operate, according to a study by [5]. The presentation by [6] stated that intelligent autonomous mobile robots must be able to observe and comprehend the 3D interior environment where they live or work. Robotic systems are increasingly being used in crowded environments, where accurately differentiating between multiple objects can be a significant challenge. While monocular and binocular vision sensors are commonly used in mobile robotics, they face issues related to visual intensity variations, a lack of feature information, and correspondence problems. To address these challenges, a group of researchers developed a unique 3D sensing system using laser-structured lighting. This approach is preferred due to its robustness in navigating complex environments and its simplicity in extracting feature information. The proposed active trinocular vision system is composed of a flexible multistripe laser projector and two cameras. The system uses trinocular epipolar constraints to represent the laser projector as a virtual camera and establish matching pairs of line characteristics between the two actual camera images. This allows for the extraction of 3D information from a single image of the patterned scene. To ensure robust feature matching, the researchers proposed a novel correspondence-matching approach based on line grouping and probabilistic voting. The proposed sensor system was tested in several experiments to demonstrate its simplicity, efficacy, and precision in 3D environment sensing and recognition. Overall, the new 3D sensing system provides a promising solution to the challenges posed by crowded environments and can potentially improve the performance of robotic systems in these scenarios.

2. Outdoor Navigation

Outdoor navigation frequently includes features like obstacle avoidance, landmark recognition, map construction and updating, and location estimation. But, at least in the research on outdoor navigation that has so far been published, seldom is the complete map of an environment known in advance; thus, the system must deal with the objects as they appear in the image without knowing where they belong. Based on how organized the environment is, the two types of outdoor navigation—outdoor navigation in structured surroundings and outdoor navigation in unstructured situations—can still be separated. Mobile robots' outdoor navigation generally involves course planning, global localization, and obstacle avoidance. The objective is to locate the robot in a global coordinate system and plot a path to a destination while avoiding environmental hazards. The method of pinpointing the location of a robot using sensor measurements is known as global localization. Numerous methods, including GPS, visual odometry, and laser-based localization, can be used to accomplish this. Once the robot's location is established, a path to the desired place may be planned using path-planning algorithms.

Path-planning algorithms take into account the robot's kinematic and dynamic constraints, as well as the geometry of the environment, to generate a path that is both feasible and optimal. The A* search, Dijkstra's algorithm, and rapidly-exploring random trees (RRT) are a few popular path-planning techniques. Once a path has been created, the robot is guided around environmental obstacles using obstacle avoidance algorithms. This can be achieved using sensor-based techniques, such as sonar, LiDAR, or vision, to detect obstacles and generate control signals that avoid collisions. Outdoor navigation also involves dealing with uncertainties in the environment, such as changes in lighting conditions or moving obstacles. To address these uncertainties, some outdoor navigation systems use adaptive algorithms that adjust the robot's behavior in real-time based on sensor feedback.

2.1. Outdoor Navigation in Structured Environments

One of the first outdoor navigation systems for a car that could drive itself at 30 km/h was reported [7] in a somewhat constrained set of circumstances. This system used two vertically positioned stereo cameras to detect potential impediments. The main navigational tactic was avoiding obstacles. In organized environments, road-following is typically required for outdoor navigation. Being able to discern between the lines that separate the lanes of the road from the berm, the texture of the road surface, and the surrounding surfaces, among other things, is necessary for being able to follow a road. Systems that follow roads often utilize simple environment models that just include information on vanishing points, lane lengths, and road widths.

2.2. Outdoor Navigation in Unstructured Environments

An outdoor area without any predictable characteristics that may be observed and used for navigation is referred to as an unstructured environment. The vision system can only use a broad description of the probable environmental obstructions under these conditions [8]. Outdoor navigation in unstructured environments can be a challenging task for mobile robots. It involves navigating through an environment that is unknown, unpredictable, and contains various obstacles such as rocks, bushes, and trees. To accomplish this task, the robot needs to perform simultaneous localization and mapping (SLAM), plan a path to a goal location, and avoid obstacles along the way.

Building an environment map and simultaneously locating the robot inside the map are both steps in the SLAM process. This can be accomplished by creating a 3D or 2D map of the surroundings using sensor data from devices like LiDAR, sonar, or cameras. Using methods like the extended Kalman filter (EKF) or the particle filter, the position of the robot on the map is approximated. Path-planning algorithms can be used to plot a path to a destination after an environment map has been created and the robot's position has been calculated. These algorithms provide a viable and ideal path by taking into consideration the environment's barriers as well as the robot's kinematic and dynamic restrictions.

The A* search method, Dijkstra's algorithm, and rapidly-exploring random trees (RRT) are a few examples of path-planning algorithms.

Obstacle avoidance is an essential component of outdoor navigation in unstructured environments. This involves detecting and avoiding obstacles along the planned path using sensors such as LiDAR or sonar. The robot's control system generates steering commands that enable it to avoid obstacles while following the planned path. The control system can be based on a variety of algorithms, such as potential fields, dynamic window approach, or artificial potential fields. To handle uncertainties in the environment, adaptive algorithms can be used to adjust the robot's behavior in real-time based on sensor feedback. This enables the robot to respond to changes in the environment such as moving obstacles or changes in lighting conditions.

References

1. Yukihiko, O.; Uchiyama, H.; Potter, W.D. A mobile robot for corridor navigation: A multi-agent approach. In Proceedings of the 42nd Annual Southeast Regional Conference, ACM-SE 42, New York, NY, USA, 2–3 April 2004.
2. Hung, L.C.; Chung, H.Y. Design of hierarchical fuzzy logic control for mobile robot system, Robotics and Automation. IEEE Trans. 2006, 18, 235–239.
3. Qin, H.; Shao, S.; Wang, T.; Yu, X.; Jiang, Y.; Cao, Z. Review of Autonomous Path Planning Algorithms for Mobile Robots. Drones 2023, 7, 211.
4. Maurović, I.; Seder, M.; Lenac, K.; Petrović, I. Path planning for active SLAM based on the D* algorithm with negative edge weights. IEEE Trans. Syst. Man Cybern. Syst. 2017, 48, 1321–1331.
5. Cao, Y.; Wang, Y.; Vashisth, A.; Fan, H.; Sartoretti, G.A. CAtNIPP: Context-aware attention-based network for informative path planning. PMLR 2023, 205, 1928–1937.
6. Ekrem, Ö.; Aksoy, B. Trajectory planning for a 6-axis robotic arm with particle swarm optimization algorithm. Eng. Appl. Artif. Intell. 2023, 122, 106099.
7. Muñoz, J.; López, B.; Quevedo, F.; Barber, R.; Garrido, S.; Moreno, L. Geometrically constrained path planning for robotic grasping with Differential Evolution and Fast Marching Square. Robotica 2023, 41, 414–432.
8. Shi, K.; Wu, Z.; Jiang, B.; Karimi, H.R. Dynamic path planning of mobile robot based on improved simulated annealing algorithm. J. Frankl. Inst. 2023, 360, 4378–4398.