Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Dynamic Coordination: Advancing Multi-Robot Vehicle Control Through Nonlinear Model Predictive Controllers

Pavithra M, T. Kavitha

Department of Electronics and Communication

RNSInstitute Of Technology

Bangalore,India

mpavikanth@gmail.com

Department of Electronics and Communication

New Horizon College of Engineering

Bangalore,India

drtkavitharaj@gmail.com

ARTICLE INFO

Received: 20 Dec 2024

Revised: 22 Feb 2025

Accepted: 26 Feb 2025

ABSTRACT

Dynamic coordination in multi-robots aims to achieve autonomous navigation with an effective obstacle avoidance mechanism. One of the main challenges associated with multi-robot coordination is handling the changes in the unknown environments. It is important to design an efficient coordination system which can provide reliable and safe path planning for a multi-robot system in a dynamic environment. This research presents an advanced approach for multi-robot coordination using nonlinear model predictive control (NLMPC) framework. Unlike linear MPCs, NLMPCs are highly effective in controlling nonlinear dynamics such as input/output constraints and robot parameters. In this research, the NLMPC strategy is used to train the robots to find targets and reach the destination by avoiding obstacles in the dynamic environment. The coordination mechanism and path planning approach employed in this research enables the robots to successfully search for the target and navigate through the obstacles. The proposed coordination approach follows a specific trajectory and adjusts the vehicle control parameters such as acceleration, and steering angle in order to stay on the specified path. Simulation is conducted to visualize the motion of the multi-robot vehicle through different trajectories. Results of the simulation show that the proposed strategy exhibits excellent acceleration and steering control and successfully avoids obstacles.

Keywords: Multi-Robot Vehicle, Dynamic coordination, Nonlinear Model Predictive Controllers, Nonlinear Optimization, Obstacle Avoidance, Path Planning

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

1. INTRODUCTION

The growing significance of the automotive industry has resulted in the increased deployment of autonomous robots for carrying out various tasks [1]. Gradually, autonomous robots are becoming one of the prominent aspects of daily life. These robots are extensively used in smart industrial and manufacturing processes such as self-driving, navigation assistance, rescue operations etc [2-4]. In order to leverage the advantages of robotic vehicles, it is crucial to make effective path planning along with a robust coordination strategy. In general, there are two types of path planning techniques namely local motion planning (LMP) and global motion planning (GMP) [5]. The LMP collects real-time information from local surroundings such as the outdoor environment to design an appropriate executable path. On the other hand, the GMP strategy utilizes the previous map data for exploring the motion path [6]. However, it is a challenging and complex task to obtain an optimal path for robots is a highly complicated task in robotics. One of the main constraints to design an effective path planning in robots is its ability to detect obstacles and avoid collisions, especially in dynamic environments [7]. Majority of the path planning techniques in robot vehicles are designed to achieve optimal performance in both static and dynamic environments [8]. In comparison to a static environment, it is difficult to design planning and coordination strategies for the robots in dynamic environments. This is mainly due to the fact that it is easy to assess the details related to the static environment for avoiding collision. In addition, the objects in the static environment are already predefined and this helps in easy navigation. However, in the dynamic environment the objects keep changing continuously and this increases the complexity for the researchers to design a path planning and coordination strategy wherein the robots can automatically adapt to the changes in such environments [9-10]. Although robots are designed to navigate through obstacles and make decisions based on the dynamic changes in their surroundings, it is not an easy task to avoid all possible collisions. To overcome these drawbacks, several coordination strategies are designed using different controlling mechanisms. Local navigation techniques have been proposed for interpreting the sensor data and thereby avoiding collision in dynamic environments. However, these techniques do not achieve desired performance in terms of accuracy of collision avoidance [11].. In this context, there is a great demand for an effective framework which can assist the robots to discover a safe and secure path for avoiding collision. Considering these aspects, it can be inferred that there is a great need for an efficient and productive controlling strategy. One such efficient controlling strategy is the deployment of Nonlinear Model Predictive Control (NLMPC) which can handle the nonlinear dynamics [12]. The NLMPC based controlling strategy can make the robot sense and interpret the information collected from the external environment and use it for determining its position, movement, and target [13]. By deploying an appropriate coordination mechanism, the robots can be trained to carry out activities by selecting optimal paths, avoiding obstacles without depending on manual intervention. The formulation of robot path planning encompasses the attainment of diverse objectives, incorporating various operational constraints such as the determination of specific trajectories, avoidance of obstacles, and adherence to speed limits. In the process of trajectory creation, obtaining essential information, including the current position, is crucial for making necessary decisions in the execution of desired tasks. Designing path planning criteria involves addressing numerous challenges such as identification of an optimized strategy to efficiently reach the target while ensuring feasibility and reliability. Various existing literary works have introduced different path planning and coordination strategies for multi-robot vehicles [14-16]. These strategies are designed to obtain an optimal path with shorter distance and high obstacle detection accuracy for robots. However, these strategies often struggle to achieve desired performance due to the presence of high uncertainties and nonlinearities in dynamic environments. Despite the availability of these techniques, there is still a lack of an effective approach which can effectively address the problems associated with path planning and obstacle avoidance in a dynamic environment. This research presents a robust controlling strategy for a multi-robot vehicle system using optimal path planning and dynamic coordination mechanism. The main objective is to design a robotic framework for assisting the robots to navigate and avoid collision in dynamic environments. The main contributions of this research are outlined as follows:

- A NLMPC strategy is designed in this research for an autonomous robot vehicle driving system which focuses on planning the vehicle's path using the dynamic coordinate system.

- The data for training the robot vehicle is collected from different sources such as Li DAR, cameras, GPS, and IMU to understand and interpret the vehicle's environment in dynamic surroundings.
- The sensor data is used to localize the vehicle within the trajectory and a path planning algorithm is deployed to define the start and goal position within the dynamic coordinate system.
- The NLMPC controller is optimized to adjust the acceleration and steering angle and to update its control inputs based on the latest measurements from the vehicle. In this way, the vehicle is trained to react to unforeseen disturbances or changes in the environment and avoid possible collisions.

The manuscript is further organized as follows. Section 2 provides an overview of existing works related to path planning and coordination of robots. Section 3 provides a brief description of the proposed research methodology which includes the description of the NLMPC strategy along with the steps involved in the implementation stages. Section 4 provides the performance evaluation details along with the simulation details. Section 5 outlines the observations obtained from the existing works as conclusion and highlights the potential directions for future scope.

2. RELATED WORKS

A significant amount of research work has been dedicated to path planning and coordination in multi-robot vehicles. Fundamental path planning techniques that are used in robots require a detailed description of the environment. These techniques rely on the map that provides the details about the surrounding environment. Some of the most commonly employed maps include point cloud map, diagram map, grid map describing the occupancy and Euclidean signed distance fields [17-19]. These maps use four main techniques namely discrete path searching, trajectory generation and optimization, trajectory tracking, and local planning. However, the effectiveness of the path planning process depends on the accuracy and reliability of these maps. This is one of the critical drawbacks of map-based techniques since it is not only challenging but also practically not feasible to customize the maps for a dynamic environment. The works presented in [20-22]. Few works have also emphasized on the application of different motion control techniques such as model predictive controls (MPC), adaptive neuro-fuzzy inference (ANFIS) techniques, and fuzzy logic controllers (FLC) [23-26]. However, these strategies are characterized by their highly complex behavior, which restricts the adaptability of these techniques. Several research works have also used navigation algorithms which are designed to plan desired paths and provide important instructions to the robots [27-28]. The work presented in [29] designed an optimal coordination strategy for multi-robots. The study focused on designing a trajectory planning and tracking strategy for achieving effective coordination in multi-robots. Initially, a method for planning a secure reference trajectory is designed which involves projecting the unsafe segment of the current distributed optimization trajectory onto the outer boundary of the obstacle region in real-time. Subsequently, a distributed back stepping tracking control scheme is introduced, utilizing a novel multiplicity-integral-type Barrier-Lyapunov function. The proof establishes that all robot systems can steer clear of unforeseen equilibriums, ultimately reaching the global optimal position while effectively avoiding collisions with dynamic obstacles. A novel approach for a robot kinematic model is presented in [30] for path planning and collision avoidance. A novel bidirectional alternating jump point search A* algorithm (BAJPSA*) is proposed in this study for finding an optimized path for the robot to traverse. A kinematic model is designed in this research for robots utilizing the dynamic window approach (DWA). This research presents an adaptive navigation strategy and introduces a novel path tracking evaluation function to enhance accuracy and optimality in path tracking. In order to enhance obstacle avoidance security, we modify the decision rules and obstacle avoidance rules for individual robots and augment the decision avoidance capability of multi-robot systems. Additionally, this research employs the mainstream prioritization method to coordinate local dynamic path planning in the multi-robot systems, resolving collision conflicts and simplifying the algorithm's obstacle avoidance complexity. Experimental results demonstrate that the proposed distributed multi-mobile robot motion planning method performs effectively in providing superior navigation and obstacle avoidance strategies within complex dynamic environments. This method serves as a valuable technical reference for practical applications. As observed

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

from the existing works, most of these algorithms have focused on identifying the shortest path and obstacle avoidance. Few studies have concentrated on avoiding dynamic obstacles without considering the degree of smoothness of the trajectory. Besides, these techniques consider inexplicit representation of the obstacle since the obstacles occupy only a smaller area of the environment. This reduces the flexibility of the path planning techniques in dynamic environments. These drawbacks motivate this research to develop an efficient dynamic coordination technique by leveraging the ability of NLMPCs to handle nonlinear dynamics. A brief overview of the proposed approach is presented in the research methodology section.

3. PROPOSED RESEARCH METHODOLOGY

The proposed research intends to find an optimized path for multi-robots in both static and dynamic environments. An autonomous robot vehicle driving system is designed in this research using MATLAB that involves several key components and steps. This system focuses on planning the vehicle's path using the dynamic coordinate system, which simplifies the complex task of path planning in a dynamic and continuous environment. The work of the proposed approach is illustrated in figure 3.1 and the steps involved in the implementation process are discussed in below subsections:

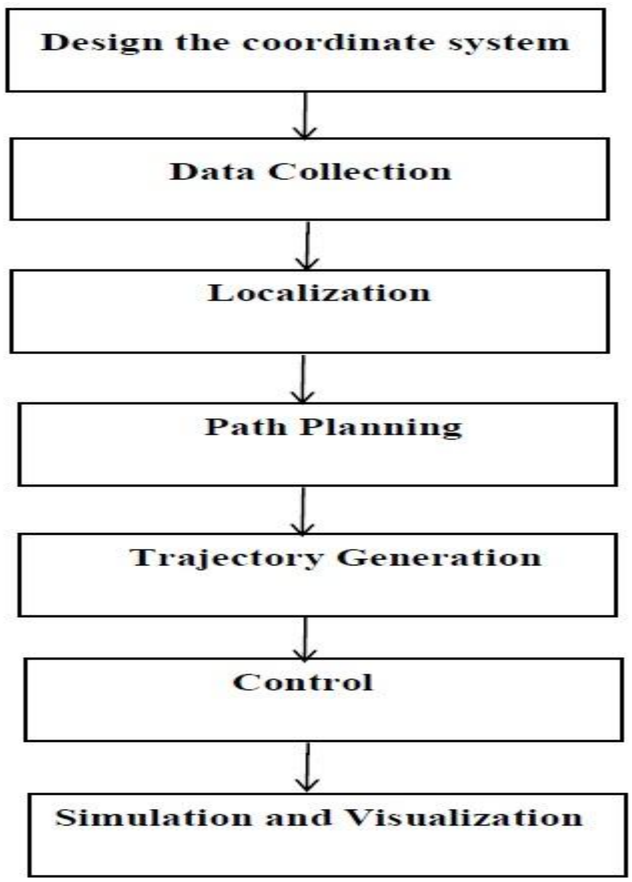

Figure 3.1 Workflow of the proposed approach

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

3.1 Designing a Coordinate System

The autonomous robot vehicle driving system is designed using a dynamic coordinate system which is a mathematical representation of a vehicle's position on a road, defined by the road's reference line (center line). It uses two coordinates: 's' (longitudinal) and 'd' (lateral). 's' represents the distance along the reference line, while 'd' represents the lateral distance from the reference line. The coordination in multi-robots is considered as a motion-planning problem which is formulated as a distributed coordination problem which aims to identify collision-free trajectories for the robots. The preliminary objective of considering a distributed coordination problem is to eliminate the necessity for a central coordinator and enable scalability of the problem with the increasing number of robots, allowing them to coordinate and independently address smaller sub-problems simultaneously. In centralized coordination, the constraints defined for collision avoidance establish interconnections among the robots. This centralization is avoided in this work by opting for distributed coordination which allows the robots to coordinate with other vehicles and solve smaller sub-problems in parallel. This reduces the computational time and reduces the complexity. The variables used in the distributed coordination are optimized using the NLMPC algorithm, which only optimizes the state variables and keeps the state variables fixed. The optimization problem is formulated in the NLMPC framework as follows:

For each time instance 't', the robot 'i' determines the trajectory on its own represented as z^i based on the variables provided over the horizon. The variables are denoted as follows: $[\lambda_{ij}(1), \ldots, \lambda_{ij}(N)]$, $[\lambda_{ji}(1), \ldots, \lambda_{ji}(N)]$ and $[s_{ij}(1), \ldots, s_{ij}(N)]$. For each robot $i \in v$, $j \in N_i$, the optimization using NLMPC is given in equation 1:

$$\min_{u^i(\cdot|t)} J^i(z^i, u^i) \quad \ldots\ldots (1)$$

Subjected to the constraints: $z^i(k+1|t) = f(z^i(k|t), u^i(k|t))$ ….1(a)

$$z^i(0|t) = z^i(t) \quad \ldots\ldots 1\,(b)$$

$$z^i(k/t) \in Z,\ u^i(k/t) \in U \quad \ldots.\ 1\,(c)$$

Where, the $u^i(\cdot|t)$ represents the sequence of control inputs over the NLMPC planning horizon N for i^{th} robot, $z^i(k|t)$ and $u^i(k|t)$ are the variables of the robot at step k are predicted at time t. The optimized trajectory is then computed along with collision avoidance wherein the collision avoidance problem is optimized as shown in equation 2.

$$A^i(z^i(k|t))^T \lambda_{ji}^*(k|t) + s_{ij}^*(k|t) = 0 \quad \ldots.\ (2)$$

Where A^i is the constant value for the i^{th} robot and is the function of z^i. The variables λ_{ij}, λ_{ji}, and s_{ij} are the known values used for collision avoidance along with the planning horizon. It was observed during the computation that in comparison to the centralized formulation, the distributed coordination optimizes only the decision variable using NMPC optimization for the i^{th} robot state z^i. In this case, the number of decision variables in the formulation of the coordination and collision avoidance problem is maintained constant.

3.2 Data Collection

The data for designing the path planning and coordination system is collected from various sensors, such as Li DAR, cameras, GPS, and IMU, to perceive the vehicle's environment. A high-definition map of the road network is obtained which provides all coordination related details, including lane information and reference lines.

3.3 Localization

The data collected from the sensors is utilized to localize the vehicle within the map. Further, the coordinates of the vehicle 's' and 'd' are defined within the dynamic system based on its position on the road. This localization map is used to develop adaptive path planning and the motion is controlled using two aspects namely an actuator and a command. These two aspects help the model to move swiftly in the optimized path. In this research, target localization for the multi-robots is achieved in a distributed manner wherein the robots act based on the local information obtained

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

by the sensors. Besides, in this work, localization is not achieved by the centralized module which uses a centralized system such as vision modules to instruct the robots. The mutual localization method employed in this research is exploited for localizing both robots and targets to overcome the drawbacks such as limited range, blind spots, and anisotropy. This validates the fact that the proposed approach is suitable for handling dynamic environmental constraints wherein each robot anticipates the trajectories based on the local information obtained from the sensors.

3.4 Path Planning using Rapidly Exploring Random Trees Algorithm

For path planning, a start and goal position is defined in the initial stages within the dynamic coordinate system. Further, a path is planned from the start to the goal using a path planning algorithm. The path planning algorithm is designed in such a way that it considers the constraints of the vehicle's kinematic, traffic rules, and obstacle avoidance. Path planning in multi-robots can be challenging considering the dynamics of robotic architecture. For mobile robot vehicles, the objective of path planning and construction is to find an optimized path in dynamic and static environments, in such a way that the path begins at a starting point (S) and ends at a target point (T). The environment where the robot is operating is considered as a two dimensional space and path is constructed for local and global path planning scenarios. In global path planning cases, the paths are constructed in known environments and the position of the objects/obstacles must be known beforehand. This is also known as a static environment. For global paths, the model of the environment where the robot is placed is defined accurately and the path is constructed based on the previous information. This improves the convergence towards the target point. Hence, computation time in this scenario is not emphasized and quality of path is focused more. In local path planning, the path is constructed where the environment is not known. In such cases, sensors are used to detect the surrounding objects to avoid the possible collision. This research uses a RRT (Rapidly-Exploring Random Trees) algorithm for path planning. The RRT planner is executed on an occupancy map which defines the start and goal states for the robot's path. The algorithm further plans a path from the start to the goal, visualizes the map, and shows the computed path for the robots to traverse in dynamic environments. The RRT is a sampling-based algorithm used for path planning in robotics. The principle behind the design of the RRT algorithm is to construct a tree incrementally by randomly sampling the configuration space and connecting these samples to existing nodes in the tree. The process involved in the computation of the RRT algorithm can be explained using mathematical terms as follows:

Initially, a state space is created for the RRT algorithm wherein X represents the configuration space (in this case, a 3D space for position and a 4D space for orientation). The terms X-start and X-goal represent the start state and goal state with position and orientation respectively. The algorithm involves manipulating states in the state space wherein it continuously checks for collisions, and constructs a tree structure. The key components are: State space 'X', tree 'T' consisting of nodes and edges, sampling module which randomly selects the states x- rand in X. The algorithm finds the nearest nodes x- nearest in T and creates respective edges. Mathematically, the RRT algorithm uses various distance metrics (like Euclidean distance) to measure distances between states and determine the nearest nodes. Initially, the length of the path for the robots is determined between two consecutive paths and by adding all the segments.. For a path p, the two consecutive points 'p_i' and 'p_{i+1}' are represented as $p_i = (x_i, y_i)$ and $p_{i+1} = (x_{i+1}, y_{i+1})$. The overall line segment can be calculated using the below equation.

$$dis(p_i, p_{i+1}) = \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2} \quad ..... (3)$$

Since the path incorporates multiple consecutive segments, the overall length with respect to path 'p' can be determined using equation 4.

$$Length\ (p) = \sum_{i=0}^{n} dis\ (p_i\ ,\ p_{i+1}) \quad .....(4)$$

Where 'n' is the number of iterations. While the target point is specified, there is no predefined information regarding the position and shape of obstacles. Consequently, prior knowledge of the environment has limited influence in this

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

scenario, and the robot is dynamically guided by data obtained from sensors. Therefore, a local path construction framework is deemed more effective for mobile robot swarms in such situations. In this research, the efficacy of the path planning is ensured by employing collision-checking functions to ensure the feasibility of newly added nodes and edges in the tree. The algorithm aims to iteratively expand the tree towards the goal state while navigating around obstacles in the state space. The pseudocode of the RRT algorithm his given below:

Pseudocode of the RRT Algorithm:

Initialization:

Start with a tree T containing only the start node x- start

Add x start to the tree T

Iterative Sampling:

Repeat until a goal is found or a maximum iteration limit is reached:

Sample a random state x- rand in the state space X.

Find the nearest node x- nearest in the tree T to x rand.

Steer from x- nearest toward x -rand within a specified distance.

If the path from x -nearest to the new state x- new is collision-free:

Add x- new to the tree T.

Create an edge between x- nearest and x -new.

Termination:

If the goal state x- goal is within a certain threshold distance from a node in the tree T:

Return the path from x -start to x- goal by tracing back the tree edges.

End

3.5 Trajectory Generation

From the obtained planned path, a continuous trajectory is generated that the multi-robot vehicle must follow. Different vehicle dynamics, such as acceleration and curvature constraints are considered, to ensure a smooth and feasible trajectory. Further, the trajectory is optimized to minimize jerk, handle the changes in the speed changes, or other performance criteria such as steering angle control. The RRT algorithm explores a feasible tool path trajectory by performing local optimization and by constructing the optimal path. The adoption of the NLMPC is highly beneficial in generating trajectories that account for the dynamic nature of the environment. NLMPC allows robots to predict their future states and interactions, optimizing trajectories based on the system's nonlinear dynamics. NLMPC is particularly useful when dealing with complex, nonlinear systems, such as vehicles. It can handle situations where traditional linear control methods may not be effective. However, it requires a good understanding of the system dynamics and a careful selection of the prediction horizon and cost function to achieve the desired control objectives. Additionally, real-time computational resources are often required, making it suitable for applications where computational power is available. The controlling strategy designed using the NLMPC controller is discussed in the next section.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

3.5 Control using NLMPC

Implementing a robust control algorithm which can effectively handle nonlinear dynamics is crucial for path planning and achieving dynamic coordination in multi-robot vehicles. The control algorithm is not only responsible for generating control signals for the robots but it is also responsible for following the generated trajectory and continuously adjusting the vehicle parameters such as acceleration, steering, to stay on the planned path. This research implements a NLMPC based strategy to control the acceleration and steering angle of the robot in order to achieve the good trajectory. NMPCs operate based on nonlinear and non-quadratic constraints for controlling the process and optimizing the performance. The formulation of NLMPCs involves developing iterative solutions to address both open-loop and closed-loop control challenges, taking into account input constraints and system dynamics. A key strength of NLMPC lies in its capacity to manage dynamic system constraints, encompassing variables that may pertain to inputs or states. The principle of NLMPC revolve around three primary principles: (a) Utilizing a nonlinear model within the system model to dynamically predict future processes in real-time applications, (b) employing an optimal control law for real-time computations by utilizing an optimal control sequence to enhance system performance and attain desired outcomes, and (c) implementing a receding horizon approach where the initial value of the control sequence is applied, followed by a shift in the horizon by one instance, leading to the calculation of new sequences based on this adjusted perspective. In this research, the NLMPC involves developing a predictive model for the multi-robot vehicle and optimizing a control sequence over a finite prediction horizon to minimize a cost function.

3.5.1 System Modeling

Initially, a nonlinear model of the vehicle dynamics is assessed and this model captures how the vehicle responds to changes in acceleration and steering inputs. Such a model can be quite complex and may include differential equations representing the motion of the vehicle. As discussed in the previous section, the primary purpose of employing the NLMPC model is to establish the control law governing the system's operation. Ensuring the precision of the model is crucial for achieving stable control. The NLMPC model exhibits a high degree of flexibility in its structure and operation, allowing for design without strict constraints. When detailed information about the dynamic environment is accessible, it can be mathematically expressed in the state space form. Conversely, when only partial information is obtainable, a black box model can be employed to represent the system's state. A cost function is defined to quantify the performance of the control inputs. For a vehicle, the cost function may include terms related to safety, comfort, and efficiency. For instance, to minimize the time taken to reach a destination the cost function is optimized to avoid sudden accelerations and maintain a safe following distance from other vehicles. The performance of the NMPC is evaluated based on the cost index which is defined as follows:

$$J = \sum_{j=N_1}^{N_2} p_k \left\| \hat{y}(k+j \mid k) - y_d(k+j) \right\|^2 + \sum_{j=1}^{N_u} q_k \left\| \Delta u(k+j-1) \right\|^2 + J_1 \quad .... (5)$$

Where, J is the initial term is used for computing the square error between the estimated output y_d (k+j) and the predicted future output y^(k + j|k). The output is predicted based on the feedback obtained at an instant k. The feedback error is calculated over a specific prediction horizon which lies between N_1 and N_2 samples. The NLMPC controller estimates the dynamic behavior of the control system by assessing the state of the model at time ‘t’. The input state of the multi-robot vehicle is determined based on the behavior of the predicted system and is measured as a function of different input and state constraints, with an aim to minimize the cost function. The NLMPC predicts the input in an open-loop system, it is considered for a finite horizon wherein the input is different from the closed loop system. The control problem is formulated over a finite prediction horizon. This means that the controller predicts how the system will evolve over a short period into the future, typically a few seconds. For the dynamic coordination system, the

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

NLMPC controller solves an optimization problem at each time step. It searches for the control inputs (acceleration and steering) that minimize the cost function over the prediction horizon, subject to the system dynamics and any constraints (e.g., physical limits on acceleration and steering). The controller then applies only the first control inputs from the optimal sequence to the real system. The optimization process is repeated at the next time step, considering the updated system state and predictions. NLMPC is a receding horizon control strategy, which means it continually updates its control inputs based on the latest measurements from the vehicle. This allows it to react to unforeseen disturbances or changes in the environment. By sensing the changes using the active sensors deployed in the autonomous robot vehicle driving system, the NLMPC instructs the robots for detecting obstacles and collision avoidance. The sensor-based techniques measure the distance of the object with a limited number of resources and hence are quite advantageous.

4. RESULTS AND DISCUSSION

This section discusses the details of the simulation analysis and results of performance evaluation.

4.1 Experimental Details

For simulation analysis, a multi-robot vehicle scenario is constructed in a dynamic environment for a robot driving simulation. Simulation is conducted using a MATLAB environment. For simulation, different vehicle parameters such as length, width, lane width, waypoints, and initial configurations are determined. A simulation loop is executed to control the robot's behavior. The NLMPC calculates and evaluates various trajectory options considering the current state of the robot, other actors, and constraints. Further, the proposed strategy checks for collisions between the robot and other objects in the environment and visualizes the scenario with trajectories and updates the robot's position and orientation. The NLMPC is designed with 30 prediction horizons, 7 states, and 2 inputs. The sample time of the controller object is set to 0.1 seconds and a manipulated variable rate (MVR) is considered to minimize the cost function. In this research, the minimum and maximum value of the first manipulated variable (Acceleration) is fixed between -2 to 2. Correspondingly, the minimum and maximum value of the second manipulated variable (Steering Angle) is fixed between -1.13 to 1.13.The loop is simulated from 1 to 31 (for 31 stages) for optimal trajectory generation. Using MATLAB, the stage cost function is specified as vehicleCostFcnLC for each stage which specifies the length of the parameter for each stage as 2. The simulation data is obtained from the nonlinear MPC object using the function nl_{obj}. The validateFcns function is defined with arguments nl_{obj}, x_0, u_0, and simdata to validate the prediction model functions at an arbitrary operating point. Further, a new optimal trajectory is generated by the NLMPC using optimal acceleration and steering angle controls. The optimized trajectory is visualized and corresponding control inputs are defined. Based on the visualization, the plots showing acceleration and steering inputs are generated over time, before and after optimization.

The steps involved in the initialization and the scenario setup are discussed as follows:

Step 1: Initialization

Clears the workspace, initializes the random number generator, and adds necessary function paths

Step 2: Creating Robot Driving Scenario

A scenario is defined for the robot to drive in, including defining the vehicle's dimensions, lane widths, waypoints, reference path, trajectory generator, and obstacle geometries.

Step 3: Trajectory Planning and Collision Avoidance

The trajectories for the multi-robot vehicle are defined and the terminal states based on cruise control, lane change, and vehicle following behaviors are evaluated. In addition, the cost of terminal states are also evaluated which eliminates invalid trajectories violating constraints, and updates collision information.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

Step 4: Visualization

The scene, trajectories and obstacles are visualized using MATLAB's visualization capabilities.

Step 5: Optimization using NLMPC

The NLMPC is used to optimize the acceleration and steering angle of the vehicle. The algorithm updates the simulation with optimized control inputs.

Step 6: Collision Visualization

The collision states before and after optimization are plotted for avoidance.

Step 7: Path Planning using RRT

The RRT algorithm is used to plan a path from a starting point to a goal while avoiding obstacles in the environment. The generated path on the map is displayed for the analysis.

4.2 Simulation Results

The performance of the proposed approach is simulated in terms of different evaluation metrics such as control outputs (acceleration and steering angle control), collision detection, and path planning. The output graphs of the proposed approach are illustrated in below figures:

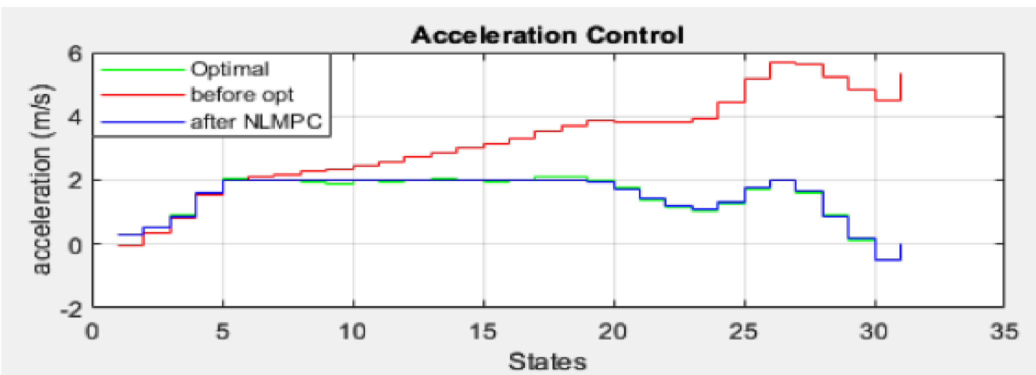

Figure 4.1 Acceleration control using NLMPC

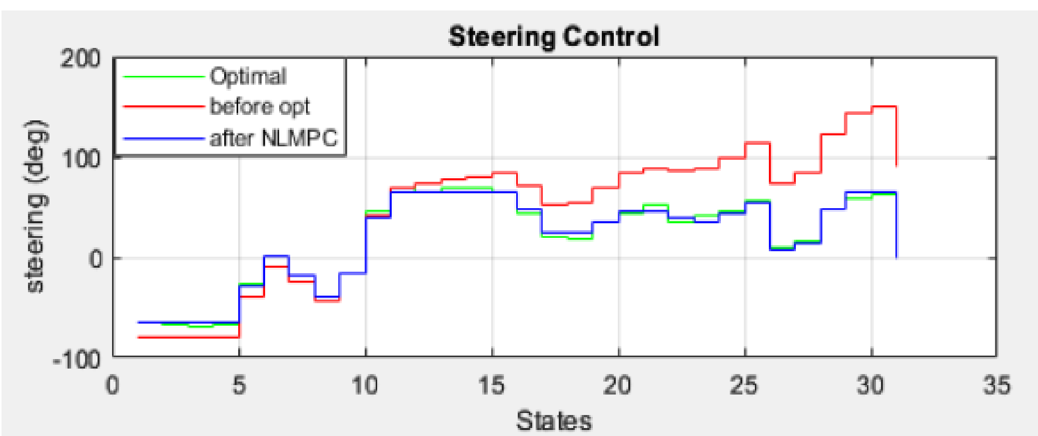

Figure 4.2 Steering angle control using NLMPC

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

It can be observed from figures 4.1 and 4.2 that the proposed NLMPC approach provides an optimal control of both acceleration and steering angle. The control of both parameters are significantly improved after employing the NLMPC strategy in comparison to before optimization. The successful control of acceleration and steering angle will help in avoiding the collisions in the dynamic environment. The performance of the NLMPC in terms of collision avoidance before and after optimization are shown in figures 4.3 and 4.4 respectively.

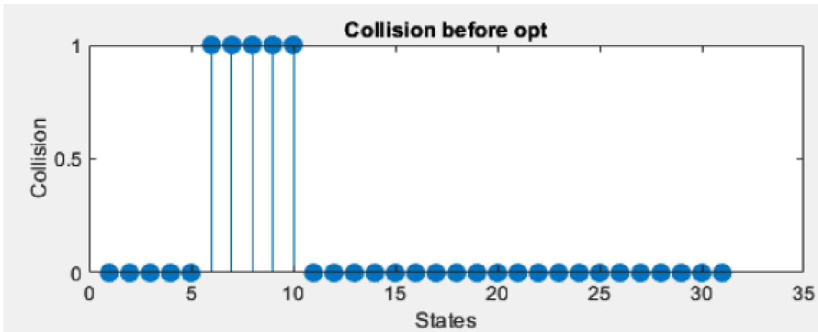

Figure 4.3 Collision avoidance before optimization

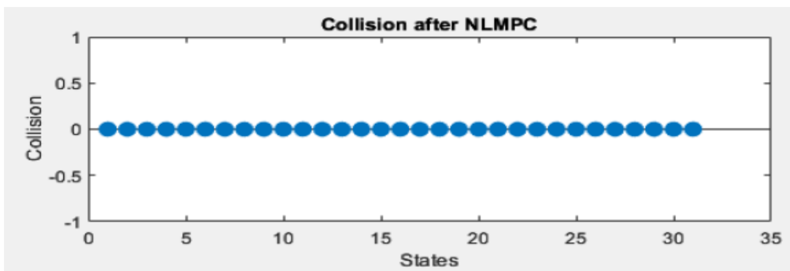

Figure 4.4 Collision avoidance after optimization using NLMPC

As inferred from figure 4.3, the robot vehicle exhibits poor collision avoidance performance which is improved after optimization using the NLMPC, which is illustrated in figure 4.4. Correspondingly, the path planning of the RRT algorithm is shown in figure 4.5.

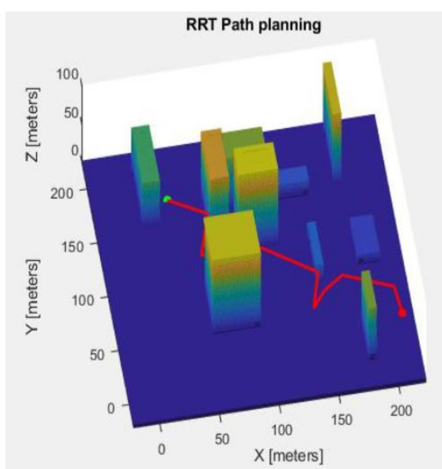

Figure 4.5 Path planning using RRT algorithm

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

The RRT algorithm identifies an optimal path for the multi-robot vehicle to reduce the time, avoid obstacles and possible collision.

5. CONCLUSION

This paper presents an efficient NLMPC approach for path planning and dynamic coordination in multi-robot vehicles for obstacle detection and collision avoidance. The proposed dynamic coordination approach overcomes the drawbacks of centralized coordination approach and trains the robot to reach the target by avoiding collisions. The RRT algorithm selects an optimal path for navigation and finds a fixed trajectory for the robot to navigate. The NLMPC approach is designed to control different vehicle parameters such as acceleration, and steering angle to maintain the predefined trajectory. A multi-robot vehicle scenario is designed and simulated using the MATLAB platform and the scene, trajectories and obstacles are visualized to determine the effectiveness of the proposed approach. Simulation results show that the control of acceleration and steering angle is improved after optimization using the NLMPC controller. Results validate the efficacy of the proposed approach. For future research, this research intends to leverage reinforcement learning mechanisms and evolutionary optimization algorithms for optimizing the performance of obstacle detection and collision avoidance in multi-robot vehicles in a dynamic environment.

REFERENCES

[1] Liu, B. (2021). Recent advancements in autonomous robots and their technical analysis. *Mathematical Problems in Engineering*, *2021*, 1-12.

[2] Liu, Z., Liu, Q., Xu, W., Wang, L., & Zhou, Z. (2022). Robot learning towards smart robotic manufacturing: A review. *Robotics and Computer-Integrated Manufacturing*, *77*, 102360.

[3] Ghodsian, N., Benfriha, K., Olabi, A., Gopinath, V., & Arnou, A. (2023). Mobile Manipulators in Industry 4.0: A Review of Developments for Industrial Applications. *Sensors*, *23*(19), 8026.

[4] Arents, J., & Greitans, M. (2022). Smart industrial robots control trends, challenges and opportunities within manufacturing. *Applied Sciences*, *12*(2), 937.

[5] Guzzi, J., Chavez-Garcia, R. O., Nava, M., Gambardella, L. M., & Giusti, A. (2020). Path planning with local motion estimations. *IEEE Robotics and Automation Letters*, *5*(2), 2586-2593.

[6] Zhou, C., Huang, B., & Fränti, P. (2022). A review of motion planning algorithms for intelligent robots. *Journal of Intelligent Manufacturing*, *33*(2), 387-424.

[7] Guo, B., Guo, N., & Cen, Z. (2022). Obstacle avoidance with dynamic avoidance risk region for mobile robots in dynamic environments. *IEEE Robotics and Automation Letters*, *7*(3), 5850-5857.

[8] Chen, G., Liu, D., Wang, Y., Jia, Q., & Zhang, X. (2018). Path planning method with obstacle avoidance for manipulators in a dynamic environment. *International Journal of Advanced Robotic Systems*, *15*(6), 1729881418820223.

[9] Wei, K., & Ren, B. (2018). A method on dynamic path planning for robotic manipulator autonomous obstacle avoidance based on an improved RRT algorithm. *Sensors*, *18*(2), 571.

[10] Hu, Y., Wang, Y., Hu, K., & Li, W. (2023). Adaptive obstacle avoidance in path planning of collaborative robots for dynamic manufacturing. *Journal of Intelligent Manufacturing*, *34*(2), 789-807.

[11] Sanchez-Lopez, J. L., Wang, M., Olivares-Mendez, M. A., Molina, M., & Voos, H. (2019). A real-time 3d path planning solution for collision-free navigation of multirotor aerial robots in dynamic environments. *Journal of Intelligent & Robotic Systems*, *93*, 33-53.

[12] Wang, M., Luo, J., & Walter, U. (2016). A non-linear model predictive controller with obstacle avoidance for a space robot. *Advances in Space Research*, *57*(8), 1737-1746.

[13] Sathya, A., Sopasakis, P., Van Parys, R., Themelis, A., Pipeleers, G., & Patrinos, P. (2018, June). Embedded nonlinear model predictive control for obstacle avoidance using PANOC. In *2018 European control conference (ECC)* (pp. 1523-1528). IEEE.

[14] Madridano, A., Al-Kaff, A., Martín, D., & De La Escalera, A. (2021). Trajectory planning for multi-robot systems: Methods and applications. *Expert Systems with Applications*, *173*, 114660.

[15] Tang, B., Xiang, K., Pang, M., & Zhanxia, Z. (2020). Multi-robot path planning using an improved self-adaptive particle swarm optimization. *International Journal of Advanced Robotic Systems*, *17*(5), 1729881420936154.

[16] Firoozi, R., Ferranti, L., Zhang, X., Nejadnik, S., & Borrelli, F. (2020). A distributed multi-robot coordination algorithm for navigation in tight environments. *arXiv preprint arXiv:2006.11492*.

[17] Freda, L., Gianni, M., Pirri, F., Gawel, A., Dubé, R., Siegwart, R., & Cadena, C. (2019). 3D multi-robot patrolling with a two-level coordination strategy. *Autonomous Robots*, *43*, 1747-1779.

[18] Huang, S. K., Wang, W. J., & Sun, C. H. (2021). A path planning strategy for multi-robot moving with path-priority order based on a generalized Voronoi diagram. *Applied Sciences*, *11*(20), 9650.

[19] Zhao, W., Lin, R., Dong, S., & Cheng, Y. (2022). A Study of the Global Topological Map Construction Algorithm Based on Grid Map Representation for Multirobot. *IEEE Transactions on Automation Science and Engineering*.

[20] Lin, S., Liu, A., Wang, J., & Kong, X. (2022). A review of path-planning approaches for multiple mobile robots. *Machines*, *10*(9), 773.

[21] Ni, J., Wang, X., Tang, M., Cao, W., Shi, P., & Yang, S. X. (2020). An improved real-time path planning method based on the dragonfly algorithm for a heterogeneous multi-robot system. *IEEE Access*, *8*, 140558-140568.

[22] Mutawe, S., Hayajneh, M., BaniHani, S., & Al Qaderi, M. (2021). Simulation of Trajectory Tracking and Motion Coordination for Heterogeneous Multi-Robots System. *Jordan Journal of Mechanical & Industrial Engineering*, *15*(4).

[23] Gupta, S., Chaudhary, S., Maurya, D., Joshi, S. K., Tripathy, N. S., & Shah, S. V. (2022, August). Segregation of Multiple Robots Using Model Predictive Control With Asynchronous Path Smoothing. In *2022 IEEE Conference on Control Technology and Applications (CCTA)* (pp. 1378-1383). IEEE.

[24] Yao, P., Wei, Y., & Zhao, Z. (2022). Null-space-based modulated reference trajectory generator for multi-robots formation in obstacle environments. *ISA transactions*, *123*, 168-178.

[25] Rafai, A. N. A., Adzhar, N., & Jaini, N. I. (2022). A review on path planning and obstacle avoidance algorithms for autonomous mobile robots. *Journal of Robotics*, *2022*.

[26] Kumar, P. B., Muni, M. K., & Parhi, D. R. (2020). Navigational analysis of multiple humanoids using a hybrid regression-fuzzy logic control approach in complex terrains. *Applied Soft Computing*, *89*, 106088.

[27] Sharma, K., & Doriya, R. (2020). Path planning for robots: An elucidating draft. *International journal of intelligent robotics and applications*, *4*, 294-307.

[28] Yang, L., Li, P., Qian, S., Quan, H., Miao, J., Liu, M., ... & Memetimin, E. (2023). Path Planning Technique for Mobile Robots: A Review. *Machines*, *11*(10), 980.

[29] An, L., Yang, G. H., & Wasly, S. (2023). Obstacle Avoidance in Distributed Optimal Coordination of Multi-Robot Systems: A Trajectory Planning and Tracking Strategy. *IEEE Transactions on Control of Network Systems*.

[30] Yang, Z., Li, J., Yang, L., Wang, Q., Li, P., & Xia, G. (2023). Path planning and collision avoidance methods for distributed multi-robot systems in complex dynamic environments. *Mathematical Biosciences and Engineering*, *20*(1), 145-178.