

Application of the SQP algorithm in leader–follower formation control

Duong Phu Tuan¹, Nguyen Tuan Minh², Nguyen Quang Vinh^{1*}

¹Academy of Military Science and Technology, 17 Hoang Sam, Nghia Do, Hanoi, Vietnam;

²Thai Nguyen University of Technology, 666 3/2 Road, Tich Luong, Thai Nguyen, Vietnam.

*Corresponding author: vinhquang2808@gmail.com

Received 5 Apr. 2024; Revised 26 May 2025; Accepted 10 Aug. 2025; Published 25 Aug. 2025.

DOI: <https://doi.org/10.54939/1859-1043.j.mst.105.2025.11-19>

ABSTRACT

This paper proposes the application of the Sequential Quadratic Programming (SQP) algorithm to control a group of omnidirectional mobile robots using the Leader-Follower model. The method includes predicting the group's kinematics, formulating a distributed model predictive control (DMPC) problem with a cost function comprising components for formation maintenance, trajectory tracking, and collision avoidance. The cost function is then optimized in real time using the SQP algorithm. In simulations, Gaussian noise is introduced into the states and control signals to evaluate the method's effectiveness under real-world deployment conditions.

Keywords: SQP; DMPC; Leader-Follower; Omnidirectional robots.

1. INTRODUCTION

In the context of the increasing application of mobile robot systems in various domains such as autonomous delivery, environmental monitoring, and search and rescue missions, formation control of robot groups has emerged as a highly attractive research topic. Among various control strategies, the leader–follower model stands out due to its simple organizational structure and excellent scalability. In this model, one or several leader robots determine the trajectory, while the follower robots are responsible for maintaining the formation by closely tracking the relative position, orientation, and velocity with respect to the leader or nearby neighbors. However, due to the typically nonlinear dynamics of the agents and the presence of constraints such as velocity limits, safety distances, and collision avoidance, this group control problem is often formulated as a constrained nonlinear optimization problem.

Sequential Quadratic Programming (SQP) is one of the most commonly used techniques for solving nonlinear optimization problems through a sequence of quadratic programming (QP) subproblems [1]. In recent years, many studies have explored the power of SQP in multi-agent environments. Zheng et al. (2018) [2] developed SQP-based Nonlinear Model Predictive Control (NMPC) techniques for cooperative systems involving multiple vehicles, including the leader–follower structure. Similarly, Javier et al (2015) [3] applied SQP to optimize formation maneuvers while ensuring collision avoidance and minimizing energy consumption. Moreover, works such as those by Javier et al (2015) [4] and Andrea Carron et al (2023) [5] have proposed distributed and accelerated SQP methods, enabling practical applications for multi-agent systems.

Model Predictive Control (MPC) is a real-time optimal control method in which control inputs are computed by solving an optimization problem over a finite prediction horizon [6]. Optimization techniques used in MPC have evolved significantly—from basic gradient methods to advanced nonlinear programming algorithms. In the field of Distributed Model Predictive Control (DMPC), optimization techniques must support decentralized computation while maintaining feasibility and accuracy. The Alternating Direction Method of Multipliers (ADMM) has become an effective approach for distributing optimization across agents [7]. Additionally, Tube-based MPC strategies have been extensively studied to ensure robustness under disturbances and uncertainties in real-world environments [8]. Selecting an appropriate optimization technique that matches the system characteristics plays a crucial role in enhancing the performance and reliability of MPC and DMPC

in modern applications. In this study, we apply DMPC to the formation control problem of omnidirectional mobile robots under the leader–follower model, where the leader follows a predefined trajectory, and the followers maintain a stable relative position with respect to the leader and their direct neighbors. The SQP algorithm is employed to optimize the cost function of the DMPC control law deployed in each robot, with the aim of maintaining a V-shaped formation under constraints on heading angles and inter-robot distances, while ensuring collision avoidance in the presence of Gaussian noise.

2. PROBLEM FORMULATION

2.1. Graph theory

In distributed leader–follower control systems, graph theory plays a crucial role in modeling the topology and communication relationships among robots. The system can be represented using a directed graph, in which:

$$\mathcal{G} = (\mathcal{V}, \mathcal{E}, A) \quad (1)$$

Where:

$\mathcal{V} = 1, 2, \dots, M$ is a set of vertices, each representing a robot.

$\mathcal{E} = \{(i, j), i, j \in \mathcal{V}, i \neq j\}$ is a set of directed edges (i, j) , representing the communication link from robot i to robot j .

$A = [a_{ij}] \in \mathbb{R}^{M \times M}$ is the adjacency matrix, in which $a_{ij} = 1, (i, j) \in \mathcal{E}$ if robot i received information of robot j , other 0.

2.2. Kinematic model of omnidirectional mobile robots

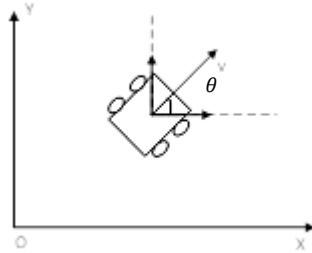


Figure 1. Robot model coordinate system.

An omnidirectional mobile robot is a type of robot capable of moving freely in a 2D plane in any direction without the need to rotate its body. Unlike conventional wheeled robots (e.g., differential drive robots), this type of robot uses Mecanum wheels or omni wheels to achieve three degrees of freedom of motion in the plane: Forward/backward along the X-axis; Left/right along the Y-axis, and Rotation around the Z-axis (θ)

Let the state vector of the robot at time t be:

$$\mathbf{x}(t) = [x \ y \ \theta]^T \quad (2)$$

Where:

x, y : The position coordinates of the robot in the global coordinate frame;

θ : The heading angle (orientation) of the robot.

Control input is

$$\mathbf{u}(t) = [v_x \ v_y \ \omega]^T \quad (3)$$

Where:

v_x, v_y : Linear velocity along the x and y axes in the robot's local coordinate frame;

ω : Angular velocity around the z-axis.

The kinematic model of the robot is represented as follows:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} \quad (4)$$

2.3. Distributed model predictive controller DMPC

Model Predictive Control (MPC) is an advanced control strategy in which, at each sampling instant, the system solves a finite-horizon, open-loop optimal control problem using the current state of the system as the initial condition. The result is an optimal control sequence, of which only the first control input is applied to the actual system [6].

In Distributed Model Predictive Control (DMPC), the optimization problem is solved in a distributed manner, where each robot solves only its own local optimization problem but can exchange information with neighboring robots [9].

Each robot individually optimizes its actions over a prediction horizon N_p . However, a robot does not rely solely on its own data for optimization - it must also exchange information with neighboring robots to ensure synchronization across the system. Each robot solves its optimization problem without needing full information about the entire system.

The DMPC control process can be described in the following steps:

- Future state prediction: Each robot uses its kinematic model to predict its future states.
- Local optimization: Each robot solves its own optimization problem (with information from neighboring robots) over the prediction horizon N_p .
- Information sharing: Each robot shares its predicted state information with neighboring robots.
- Control execution: Each robot executes only the first control action of its optimized sequence (and repeats the process at the next time step).

The cost function in DMPC plays a critical role in optimizing the robots' actions within a distributed system, helping achieve control objectives such as formation maintenance, collision avoidance, and inter-robot consensus.

The general DMPC cost function not only evaluates the tracking error between the actual and desired states but also includes other elements such as control effort and constraints related to interactions among the robots.

The cost function for each robot can be expressed as follows:

$$L_i(\mathbf{x}_i, \{\mathbf{x}_j\}_{j \in \mathcal{N}_i}, \mathbf{u}_i) = L_{ii}(\mathbf{x}_i, \mathbf{u}_i) + \sum_{j \in \mathcal{N}_i} L_{ij}(\mathbf{x}_i, \mathbf{x}_j) + L_{if} \quad (5)$$

Where:

L_{ii} is the cost function related to the robot's states and control inputs i ;

L_{ij} is the cost function related to the robot's state inputs j neighboring (robots that exchange information);

\mathcal{N}_i is the set of robot's neighboring i ;

L_{if} is the terminal cost function of the robot i .

To determine the control input \mathbf{u}_i , it is necessary to solve the optimization problem of the cost function $L_i \rightarrow \min$.

2.4. Algorithm SQP

Sequential Quadratic Programming (SQP) is a powerful nonlinear optimization algorithm used

to solve constrained nonlinear optimization problems. The method iteratively approximates the original problem by a sequence of Quadratic Programming (QP) subproblems, in which:

- The objective function is approximated using a second-order Taylor expansion (with the Hessian matrix or its approximation);
- The constraints are linearized at the current point.

At each iteration, a QP subproblem is solved to find an optimal search direction, and the solution is then updated along that direction.

Find out:
$$\min_{x \in \mathbb{R}^n} f(x) \quad (6)$$

Subject to:
$$\mathbf{h}_i(\mathbf{x}) = 0, i = 1, \dots, m \quad (7)$$

$$\mathbf{g}_j(\mathbf{x}) \leq 0, j = 1, \dots, p \quad (8)$$

At each iteration, SQP solves the following QP problem:

Approximation of the objective function:

$$\min_d \nabla f(\mathbf{x}_k)^\top \mathbf{d} + \frac{1}{2} \mathbf{d}^\top \mathbf{B}_k \mathbf{d} \quad (9)$$

Constraint linearization:

$$\mathbf{h}(\mathbf{x}_k) + \nabla \mathbf{h}(\mathbf{x}_k)^\top \mathbf{d} = 0 \quad (10)$$

$$\mathbf{g}(\mathbf{x}_k) + \nabla \mathbf{g}(\mathbf{x}_k)^\top \mathbf{d} \leq 0 \quad (11)$$

Where:

\mathbf{B}_k is the approximate Hessian matrix of the Lagrangian function at iteration k ;

\mathbf{d} is the search direction vector.

The Lagrangian function is defined as:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\mathbf{x}) + \sum_{i=1}^m \lambda_i h_i(\mathbf{x}) + \sum_{j=1}^p \mu_j g_j(\mathbf{x}) \quad (12)$$

Where:

$f(\mathbf{x})$ is the objective function;

$h_i(\mathbf{x})$ is the equality constraints;

$g_j(\mathbf{x})$ is the inequality constraints;

λ_i, μ_j are Lagrange multipliers, treated as auxiliary variables.

3. DESIGN OF THE DMPC CONTROLLER USING SQP

In formation movement, it is necessary to ensure consensus on both distance and heading angle, combined with collision avoidance. Based on this, the cost function of the DMPC control law is constructed as follows:

$$J_i(\mathbf{x}_i, \hat{\mathbf{x}}_j, \mathcal{N}_i) = \sum_{k=0}^{N_p-1} \left[\left\| \mathbf{x}_i(k) - \mathbf{x}_{ref,i}(k) \right\|_{Q_c}^2 + \sum_{j \in \mathcal{N}_i} \left\| \mathbf{x}_i(k) - \hat{\mathbf{x}}_j(k) \right\|_{Q_c}^2 + \left\| \mathbf{u}_i(k) \right\|_R^2 \right] + \left\| \mathbf{x}_i(N_p) - \mathbf{x}_{ref,i}(N_p) \right\|_{Q_f}^2 \quad (13)$$

Where:

$\mathbf{x}_i(k)$ is the predicted state of robot i at time step k ;

$\mathbf{x}_{ref,i}(k)$ is the reference point of the predicted trajectory that robot i followed;

$\hat{x}_j(k)$ is the predicted state of robots j ($j \in \mathcal{N}_i$ - The set of robots that exchange information with robot i);

$u_i(k)$ is the predicted control input of robot i at time step k ;

N_p is the prediction horizon length;

Q, Q_c, R, Q_f are the control weighting matrices corresponding to the trajectory tracking cost, consensus constraint cost, control effort cost, and terminal cost, respectively.

By discretizing the kinematic model of the robot, we obtain:

$$\begin{bmatrix} x(k+1) \\ y(k+1) \\ \theta(k+1) \end{bmatrix} = \begin{bmatrix} x(k) \\ y(k) \\ \theta(k) \end{bmatrix} + T_s \begin{bmatrix} \cos\theta_k & -\sin\theta_k & 0 \\ \sin\theta_k & \cos\theta_k & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x^k \\ v_y^k \\ \omega_k \end{bmatrix} \quad (14)$$

$$x_i(k+1) = f(x_i(k), u_i(k)) \quad (15)$$

$$J_i(x_i(k), U_i) = \sum_{k=0}^{N_p-1} l_i(x_i(k), u_i(k), \hat{x}_{\mathcal{N}_i}(k)) + V_{f,i}(x_i(N_p)) \quad (16)$$

Where:

l_i is the cost of trajectory maintenance, consensus, and control;

$V_{f,i}$ is the terminal cost.

We assume that:

- Each robot is constrained within a closed control set \mathcal{U} and a closed state set \mathcal{X} ;
- Each robot sends its predicted state $\hat{x}_{\mathcal{N}_i}$ to its neighboring robots at each time step;
- There exists a terminal control law $\mathcal{K}_{f,i}(x)$ such that:

$$x_i \in \mathcal{X}_f \Rightarrow f_i(x_i, \mathcal{K}_{f,i}(x_i)) \in \mathcal{X}_f \quad (17)$$

$$V_{f,i}(f(x_i, \mathcal{K}_{f,i}(x_i))) - V_{f,i}(x_i) \leq -l_i(x_i(k), \mathcal{K}_{f,i}(x_i), \hat{x}_{\mathcal{N}_i}) \forall x_i \in \mathcal{X}_f \quad (18)$$

The Lyapunov function for the entire system is defined as the total optimal cost at each time step:

$$V(x(k)) = \sum_{i=1}^N J_i^*(x_i(k)) \quad (19)$$

where $J_i^*(x_i(k))$ is the optimal value of the system at time step k .

When each robot applies the optimal control law $u_i^*(k)$:

$$V(x(k+1)) - V(x(k)) = \sum_{i=1}^N (J_i^*(x_i(k+1)) - J_i^*(x_i(k))) \quad (20)$$

$$\leq \sum_{i=1}^N -l_i(x_i(k), u_i^*(k), \hat{x}_{\mathcal{N}_i}) < 0 \quad (21)$$

Since the cost decreases at each step, the Lyapunov function decreases over time, and the system is asymptotically stable in the sense of Lyapunov.

Based on the constructed cost function, this paper employs the SQP algorithm as follows:

At each time step, each follower robot formulates a local optimization problem based on a finite prediction model as follows:

$$\min_{u_i(k), \dots, u_i(k+N_p-1)} J_i = \sum_s^{N_p-1} (J_i^{cons} + J_i^{track}) + \|x_i(N_p) - x_{ref,i}(N_p)\|_{Q_f}^2 \quad (22)$$

Subject to:

Input control:

$$\mathbf{v}_{min} \leq \mathbf{v}_i(k+s) \leq \mathbf{v}_{max}, \omega_{min} \leq \omega_i(k+s) \leq \omega_{max} \quad (23)$$

Where:

J_i^{cons} is the consensus constraint, $J_i^{cons}(\mathbf{s}) = \sum_{j \in \mathcal{N}_i} \|\mathbf{x}_i(\mathbf{s}) - \hat{\mathbf{x}}_j(\mathbf{s})\|_{Q_c}^2$;

J_i^{track} is the trajectory, $J_i^{track}(\mathbf{s}) = \|\mathbf{x}_i(k) - \mathbf{x}_{ref,i}(k)\|_Q^2$;

Q_f is the terminal state convergence weighting matrix.

The SQP solver will iteratively solve a sequence of approximate QP problems of the form:

$$\min_{\mathbf{d}} \nabla J_i(\mathbf{U})^T \mathbf{d} + \frac{1}{2} \mathbf{d}^T \mathbf{H} \mathbf{d} \quad (24)$$

st: $\mathbf{lb} \leq \mathbf{U} + \mathbf{d} \leq \mathbf{ub}$

Where:

$\nabla J_i(\mathbf{U})$ is the gradient of the cost function with respect to the control variables

$$\nabla J_i(\mathbf{U}) \approx \frac{J_i(\mathbf{U} + \epsilon \mathbf{e}_j) - J_i(\mathbf{U})}{\epsilon} \quad (25)$$

\mathbf{e}_j is the unit vector (1 at index j , 0 elsewhere);

$\epsilon \ll 1$ is a small perturbation step (typically 10^{-6} to 10^{-8});

\mathbf{H} is the approximate Hessian matrix of the Lagrangian function;

\mathbf{lb}, \mathbf{ub} are the lower and upper bounds of linear velocity and angular velocity.

At each iteration, the control vector is updated according to the following formula:

$$\mathbf{U}^{(t+1)} = \mathbf{U}^{(t)} + \alpha^{(t)} \mathbf{d}^{(t)} \quad (26)$$

with $\alpha^{(t)} \in (0,1]$ is the step size, determined using a line search method to ensure the cost function decreases.

Here is the pseudo-code for implementing DMPC using the SQP optimization algorithm:

Algorithm DMPC using SQP

- 1: // Initialization: at time step $k = 0$
 - 2: Initialize the state of the robot and its neighbors: $\mathbf{x}_i(0), \mathbf{x}_j(0)$
 - 3: Set the initial estimate for the control sequence: $\hat{\mathbf{u}}_i(h|0)$ với $h \in \{0, \dots, N_p - 1\}$
 - 4: **For** each time step $k > 0$ **do**
 - 5: Get the current state $\mathbf{x}_i(k)$
 - 6: Predict future states $\hat{\mathbf{x}}_i(k+h|k)$ based on the current control sequence $\hat{\mathbf{u}}_i(k+h|k)$
 - 7: Send $\hat{\mathbf{x}}_i(k+h|k)$ to neighboring robots
 - 8: Receive predicted states $\hat{\mathbf{x}}_j(k+h|k)$ from neighbors
 - 9: Solve the local optimization problem using SQP to obtain the optimal control sequence $\hat{\mathbf{u}}_i^*(k+h|k)$
 - 9.1 **Loop:** Until convergence
 - 9.2 - Linearize dynamics and constraints around $(\hat{\mathbf{x}}_i, \hat{\mathbf{u}}_i)$
- $$\mathbf{x}_i(k+1) \approx f(\hat{\mathbf{x}}_i(k), \hat{\mathbf{u}}_i(k)) + A_k \Delta \mathbf{x}(k) + B_k \Delta \mathbf{u}(k)$$
-

$$A_k = \frac{\partial f}{\partial x}, B_k = \frac{\partial f}{\partial u}$$

9.3 - Perform a second-order approximation of the cost function around (\hat{x}_i, \hat{u}_i)

$$J(U) \approx J(\hat{U}) + \nabla J^T \Delta U + \frac{1}{2} \Delta U^T H \Delta U$$

9.4 - Solve the QP subproblem to obtain $\Delta u = \{\Delta u(0), \dots, \Delta u(N - 1)\}$

9.5 - Update: $\hat{u}_i(h) \leftarrow \hat{u}_i(h) + \alpha \Delta u(h)$

$$\hat{x}_i(h + 1) \leftarrow f(\hat{x}_i(h), \hat{u}_i(h))$$

where $\alpha \in (0, 1]$ is the step size

9.6 **End Loop**

10: Apply the first control input: $\hat{u}_i^*(k|k)$

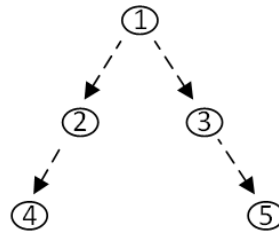
11: Update the state: $x_i(k + 1)$

12: **End for**

4. SIMULATION RESULTS

The simulation conducted in this study uses a sinusoidal and circular trajectory for the leader robot (with a predefined path), while four follower robots track this trajectory and form a V-shaped formation under constraints on distance consensus, heading alignment, and collision avoidance. Based on graph theory, the communication model among the five robots is constructed, consisting of one leader and four followers moving in a V-shaped formation.

The communication structure is assumed as follows:



Leader robot No. 1 communicates only with two neighboring follower robots, No. 2 and No. 3.

Follower robot No. 2 sends information to follower robot No. 4.

Follower robot No. 3 sends information to follower robot No. 5.

At any given time, information is transmitted directionally as shown in the figure, and the transmitted information contains the predicted states of each robot after each computation within the prediction horizon \mathcal{N}_p .

The adjacency matrix A of the model is

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

where $a_{21} = a_{31} = a_{42} = a_{53} = 1$ representing the connectivity and direction of information flow between the robots.

Based on the simulation results, it can be observed that the SQP algorithm demonstrates strong performance in terms of formation accuracy and control effectiveness. When Gaussian noise is introduced, the trajectories and errors exhibit slight oscillations (depending on the noise intensity);

however, the system still converges, and the robots are able to maintain the desired formation. Therefore, SQP is a reliable choice for practical DMPC applications.

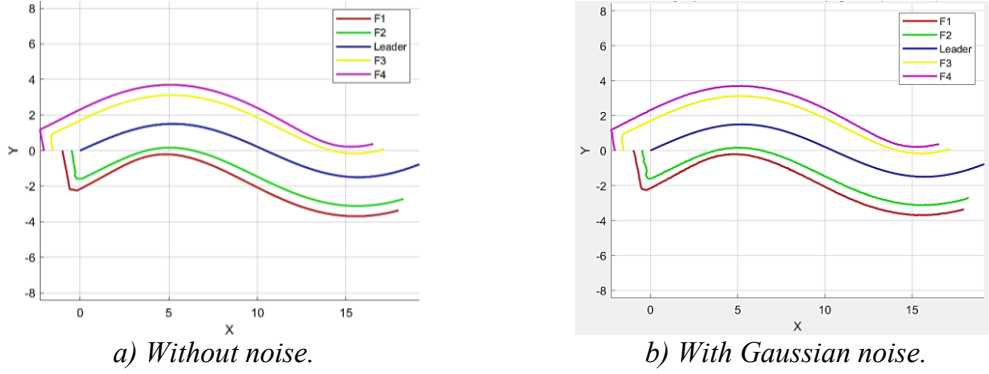


Figure 2. The sinusoidal movement trajectory of the robots maintains a V-shaped formation.

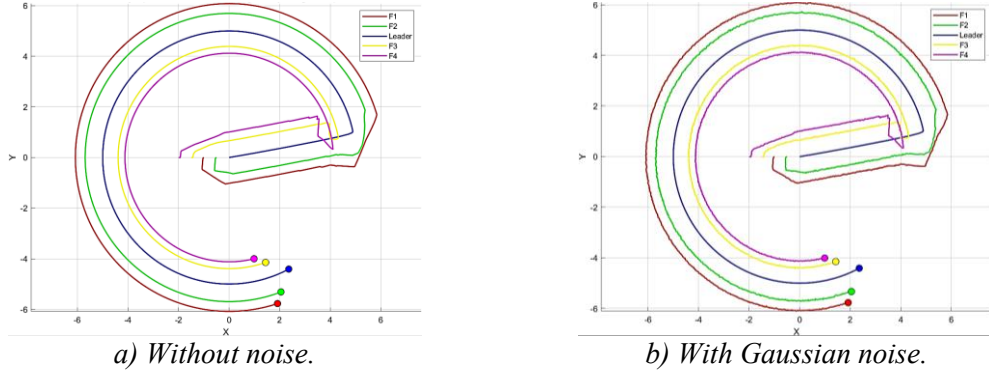


Figure 3. The circular movement trajectory of the robots maintains a V-shaped formation.

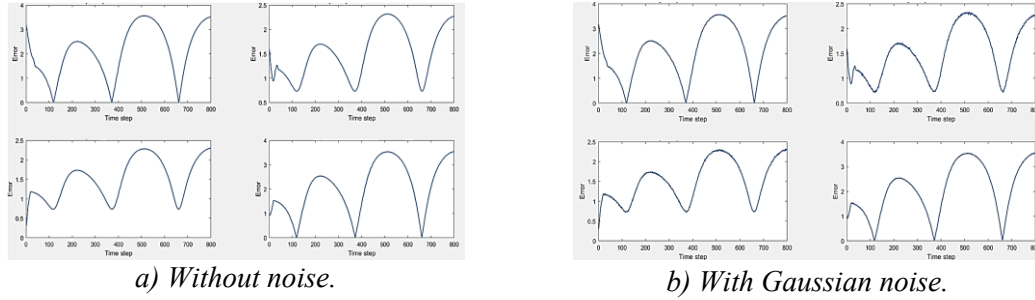


Figure 4. Formation deviation of each robot moving within the formation.

5. CONCLUSIONS

This paper has presented the implementation of the Sequential Quadratic Programming (SQP) algorithm for optimizing the cost function of a Distributed Model Predictive Control (DMPC) controller in the formation control of omnidirectional mobile robots using a V-shaped Leader-Follower strategy under constraints in a noise-affected environment. The results demonstrate the potential of integrating DMPC controllers with SQP-based optimization algorithms for practical deployment.

Future research directions include extending the model to incorporate full system dynamics as well as obstacle avoidance components to further validate the algorithm's feasibility in real-world applications. Another promising direction is to vary the prediction horizon length to assess the method's adaptability to problems requiring real-time responsiveness.

REFERENCES

- [1]. Nocedal, J., & Wright, S, "Numerical Optimization" (2nd ed.), Springer, (2006).
- [2]. Zheng, Y., Wang, L., & Yang, G. "Distributed optimal coordination for multi-agent systems with leader-follower structure via a novel SQP method", Journal: IEEE Transactions on Cybernetics, (2018)
- [3]. Wang, Xinghu, Dabo Xu, and Yiguang Hong. "Consensus control of nonlinear leader-follower multi-agent systems with actuating disturbances." Systems & Control Letters 73, (2014).
- [4]. Javier Alonso-Mora, Stuart Baker and Daniela Rus. "Multi-robot navigation in formation via sequential convex programming", IEEE/RSJ International Conference on Intelligent Robots and Systems, (2015).
- [5]. Andrea Carron, Danilo Saccani, Lorenzo Fagiano, Melanie N. Zelinger, "Multi-agent Distributed Model Predictive Control with Connectivity Constraint", IFAC-PapersOnline 56-2, (2023).
- [6]. E.F. Camacho, C.Bordons, "Model Predictive Control" (2nd ed). Springer, (2006).
- [7]. Boyd, S., Parikh, N., Chu, E., Peleato, B., & Eckstein, J, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers", Foundations and Trends in Machine Learning, (2011).
- [8]. Mayne, D. Q., Seron, M. M., & Raković, S. V, "Robust model predictive control of constrained linear systems with bounded disturbances", Automatica, (2005).
- [9]. J. M. Maestre, R. Negenborn, " Distributed Model Predictive Control Made Easy", Springer, (2014).
- [10]. Tran Thi Minh Dung, "Nghiên cứu về thuật toán đồng thuận Cho hệ thống đa đối tượng". Journal of Science and Technology – The University of Danang, Vol. 6, No. 103, (2016) (in Vietnamese).
- [11]. Trinh Hoang Minh, Nguyen Minh Hieu, "Điều khiển hệ đa tác tử", Hanoi University of Science and Technology, (2021) (in Vietnamese).
- [12]. Nguyen Thi Tuyet Hoa, Nguyen Tuan Minh, "Điều khiển nhóm phân tán cho hệ đa tác tử dựa trên lý thuyết đồ thị cứng ứng dụng trong theo dõi mục tiêu di động", TNU journal of science and technology, (2023) (in Vietnamese).
- [13]. H. Do, H. Nguyen, C. Nguyen, M. Nguyen, and M. Nguyen, "Formation control of multiple unmanned vehicles based on graph theory: A comprehensive review", EAI Endorsed Transactions on Mobile Communications and Applications, (2022).

TÓM TẮT

Ứng dụng thuật toán SQP trong điều khiển nhóm leader-follower

Bài báo đề xuất ứng dụng thuật toán SQP (Sequential Quadratic Programming) để điều khiển nhóm robot di động đa hướng có mô hình Lead-Follower. Phương pháp bao gồm dự đoán động học của toàn nhóm, xây dựng bài toán điều khiển dự đoán mô hình phân tán (DMPC) với hàm chi phí gồm các thành phần duy trì đội hình bám quỹ đạo, tránh va chạm, đồng thời giải bài toán tối ưu hàm chi phí này theo thời gian thực bằng SQP. Trong mô phỏng có sử dụng nhiễu Gauss tác động lên trạng thái và tín hiệu điều khiển để kiểm tra hiệu quả của phương pháp khi triển khai trong thực tế.

Từ khóa: SQP; DMPC; Dẫn-Theo; Rô bốt di động đa hướng.