

How Constraints Define Control for Multi-Agent Navigation Robots

Boyang Zhang¹

University of Louisiana at Lafayette, Lafayette, LA 70503, USA,
boyang.zhang@louisiana.edu

Abstract. We present a general control framework that is based entirely on system performance constraints. This nonlinear, discontinuous control results from solving a linear matrix equation without iteration at each instant of time. No cost function, or time horizon, or linearization is involved in obtaining the control actions. The methodology is applied to controlling multi-agent navigation robots subject to collisions. Collision avoidance performance and control structure (centralized or decentralized) are realized purely by whether the constraint is defined on vector norms or vector components and by whether the constraint is partitioned between colliding agents. Numerical and analytical results are presented to compare three different constraint formulations, demonstrating the effectiveness of the control approach.

Keywords: multi-agent robots, collision avoidance, constrained control, dynamic stabilization

1 Introduction

As we enter an era of autonomy and artificial intelligence, autonomous and intelligent robots in the air, on the ground, and in the water are gaining increasing attention in transporting cargoes and humans in a diverse range of applications, such as autonomous delivery and transportation [1]. One critical capability of teams of such robots is to guarantee safety through collision avoidance, which attracts ongoing research efforts in the robotics and controls community [2].

Existing methods for multi-agent robot collision avoidance include velocity obstacles (VO) [3], artificial potential fields (APF) [4], mixed-integer programs (MIP) [5], control barrier functions (CBF) [6], and reinforcement-learning (RL) based methods [7].

With certain levels of success achieved, existing approaches may suffer from shortcomings in certain environments. For examples, VO methods (e.g., [3]) assume that robots avoid collision at constant speeds with no solution uniqueness guarantee; APF methods (e.g., [4]) assume physically unrealistic control actions for imminent crashes, with solutions prone to local optima; MIP (e.g., [5]) become computationally intractable when the number of agents is significant; CBF approaches (e.g., [6]) may not be computationally efficient when the end points do not remain static in solving the constrained optimization problem; while RL-based methods (e.g., [7]) can consume enormous computational resources yet may still not capture some outlier scenarios in the training datasets and may have a large number of physically meaningless control parameters.

In this article, we present a general control methodology that addresses the limitations of the existing methods by 1) having no premature assumptions on avoidance speeds, 2) having a relatively small number of control parameters that are physically meaningful, 3) having a solution uniqueness guarantee, and 4) having simple and efficient computations for control actions.

The methodology is based on extending Gauss's principle of least constraint (GPLC) [8] with dynamic identification, incorporation, and stabilization of active inequality constraints. The method is then applied to the collision avoidance control of multi-agent navigation robots. Constraints formulated on either vector norms or vector components and whether the interacting constraints are partitioned or not dictate the performance and structure of the control.

This paper is organized as follows. Section 2 reviews GPLC and presents GPLC control in an abstract and general manner. Then, Section 3 models the collision control of multi-agent navigation systems in three different constraint formulations. Next, Section 4 numerically and analytically compares the three formulations by investigating a baseline case in which two robots are under head-on collision. Finally, conclusions and future work are summarized in Section 5.

2 Methodology

2.1 Gauss's Principle of Least Constraint (GPLC)

In 1829, Gauss presented the principle of least constraint in his 4-page seminal paper [8]. We now review GPLC from a constrained optimization perspective.

Consider a nonlinear, time-varying dynamical system free of constraints whose equations of motion are expressed as

$$\mathbf{M}(\mathbf{q}, t) \mathbf{a} = \mathbf{f} , \quad (1)$$

where $\mathbf{M} \in \mathbb{R}^{N_{\mathbf{q}} \times N_{\mathbf{q}}}$ is the symmetric positive definite (SPD) mass matrix that is dependent on $N_{\mathbf{q}}$ coordinate positions $\mathbf{q} \in \mathbb{R}^{N_{\mathbf{q}}}$ and time t , $\mathbf{a} \in \mathbb{R}^{N_{\mathbf{q}}}$ denotes the unconstrained accelerations, and $\mathbf{f} \in \mathbb{R}^{N_{\mathbf{q}}}$ is the collection of all forces that do not result from a set of equality constraints $\mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{c}$, where $\mathbf{g} \in \mathbb{R}^{N_{\mathbf{g}}}$ are the (at-least) twice-differentiable functions that constrain the relations among the coordinate positions \mathbf{q} and the coordinate velocities $\dot{\mathbf{q}}$, and $\mathbf{c} \in \mathbb{R}^{N_{\mathbf{g}}}$ denotes constant thresholds for the constraint functions \mathbf{g} . In the context of navigation control of multi-agent robots, \mathbf{g} describes the desired system behaviors such as path following and collision avoidance and is thus a function of only \mathbf{q} and t .

Taking the second-order time derivative of $\mathbf{g}(\mathbf{q}, t) - \mathbf{c} = \mathbf{0}$ yields a system of equations that is linear in the coordinate accelerations $\ddot{\mathbf{q}}$

$$\mathbf{A} \ddot{\mathbf{q}} = \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}, t) , \quad (2)$$

where $\mathbf{A} \in \mathbb{R}^{N_{\mathbf{g}} \times N_{\mathbf{q}}}$ is the Jacobian matrix.

Note that the unconstrained equations of motion (1) is equivalent to the solution of the unconstrained quadratic programming

$$\min_{\ddot{\mathbf{q}}} \frac{1}{2} \ddot{\mathbf{q}}^T \mathbf{M} \ddot{\mathbf{q}} - \mathbf{f}^T \ddot{\mathbf{q}} \quad (3)$$

and that satisfying the constraint equations $\mathbf{g} = \mathbf{c}$ requires that $\ddot{\mathbf{q}}$ satisfies (2). We can therefore express the constrained equations of motion as a constrained quadratic programming by noting (1) [10]

$$\min_{\ddot{\mathbf{q}}} \frac{1}{2} \ddot{\mathbf{q}}^\top \mathbf{M} \ddot{\mathbf{q}} - \mathbf{f}^\top \ddot{\mathbf{q}} \quad \text{s.t.} \quad \mathbf{A} \ddot{\mathbf{q}} = \mathbf{b} \quad (4)$$

$$\iff \max_{\boldsymbol{\lambda}} \min_{\ddot{\mathbf{q}}} \frac{1}{2} \ddot{\mathbf{q}}^\top \mathbf{M} \ddot{\mathbf{q}} - \mathbf{f}^\top \ddot{\mathbf{q}} + \boldsymbol{\lambda}^\top (\mathbf{A} \ddot{\mathbf{q}} - \mathbf{b}) \quad (5)$$

$$\iff \max_{\boldsymbol{\lambda}} \min_{\ddot{\mathbf{q}}} \frac{1}{2} \ddot{\mathbf{q}}^\top \mathbf{M} \ddot{\mathbf{q}} - \ddot{\mathbf{q}}^\top \mathbf{f} + \frac{1}{2} \mathbf{a}^\top \mathbf{M} \mathbf{a} + \boldsymbol{\lambda}^\top (\mathbf{A} \ddot{\mathbf{q}} - \mathbf{b}) \quad (6)$$

$$\iff \max_{\boldsymbol{\lambda}} \min_{\ddot{\mathbf{q}}} \frac{1}{2} (\ddot{\mathbf{q}} - \mathbf{a})^\top \mathbf{M} (\ddot{\mathbf{q}} - \mathbf{a}) + \boldsymbol{\lambda}^\top (\mathbf{A} \ddot{\mathbf{q}} - \mathbf{b}) \quad (7)$$

where $\boldsymbol{\lambda} \in \mathbb{R}^{N_g}$ denotes the Lagrange multipliers that correspond to active constraints. Solving (5) yields a Karush-Kuhn-Tucker (KKT) system

$$\begin{bmatrix} \mathbf{M} & \mathbf{A}^\top \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{b} \end{bmatrix}, \quad (8)$$

which is a linear matrix equation. We may regard $\mathbf{f}^c \triangleq -\mathbf{A}^\top \boldsymbol{\lambda}$ as a control force since it enforces the equality constraints \mathbf{g} at a differentiated level. Equation (7) takes the form of GPLC. Therefore, we call (8) the GPLC KKT system.

2.2 Udwadia-Kalaba Equations

In 1992, Udwadia and Kalaba derived an analytical solution to GPLC (7) for the constrained accelerations $\ddot{\mathbf{q}}$ and the force of constraints \mathbf{f}^c [9]. In their approach, Lagrange multipliers $\boldsymbol{\lambda}$ are eliminated, yielding the closed-form expressions

$$\begin{aligned} \ddot{\mathbf{q}} &= \mathbf{a} + \mathbf{M}^{-1} \mathbf{K} (\mathbf{b} - \mathbf{A} \mathbf{a}), \\ \mathbf{f}^c &= \mathbf{K} (\mathbf{b} - \mathbf{A} \mathbf{a}) = \mathbf{K} \mathbf{A} (\ddot{\mathbf{q}} - \mathbf{a}), \end{aligned} \quad (9)$$

where $\mathbf{K}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{M}^{1/2} (\mathbf{A} \mathbf{M}^{-1/2})^+$, and the superscript $+$ represents the Moore-Penrose pseudo-inverse. Therefore, obtaining the constraint force \mathbf{f}^c is more computationally involved than that of the KKT-based approach (8). Furthermore, since the Lagrange multipliers are eliminated, the relative contribution of a specific constraint to the control actions cannot be determined.

2.3 Active Constraint Stabilization

When the dynamical system is subjected to inequality constraints, both the GPLC and the Udwadia-Kalaba (U-K) equations become inapplicable. Furthermore, the inevitable errors due to estimation, measurement, and/or numerical integration would drift the state trajectories away from the constraint manifolds. Zhang and Gavin [10] proposed an active-set method that dynamically identifies, incorporates, and stabilizes active inequality constraints.

Suppose the system is under the inequality constraints

$$\mathbf{g}^{\text{ineq}}(\mathbf{q}, t) \leq \mathbf{c}, \quad (10)$$

where \mathbf{c} denotes constant thresholds. At each time instant, a subset of \mathbf{g}^{ineq} has their function values exceeding the corresponding thresholds. This subset of active inequality constraints, along with the equality constraints, comprises the active constraint set and is treated as equality constraints

$$\mathbf{g} = \mathbf{c} \quad (11)$$

at the present time instant. Note that \mathbf{g} and $N_{\mathbf{g}}$ take active inequality constraints into consideration in the present and following sections.

For any linearly independent element of $g \in \mathbf{g}$, we enforce a second-order oscillator

$$\ddot{g} + 2\zeta\omega\dot{g} + \omega^2(g - c) = 0, \quad (12)$$

where ω and ζ are the natural frequency and damping ratio of the constraint oscillator and thus can be viewed as control parameters.

We apply the second-order constraint stabilization to all active g and write it into a compact form

$$\mathbf{A}(\mathbf{q}, t) \ddot{\mathbf{q}} = \hat{\mathbf{b}}(\mathbf{q}, \dot{\mathbf{q}}, t), \quad (13)$$

where $\hat{\mathbf{b}}$ now contains terms involving ω and ζ .

Substituting the stabilized constraints (13) in the place of the differentiated constraints (2) into the GPLC objective (5) and re-applying the KKT conditions, we arrive upon

$$\begin{bmatrix} \mathbf{M} & \mathbf{A}^\top \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \hat{\mathbf{b}} \end{bmatrix}. \quad (14)$$

When g becomes inactive (i.e., $g \leq c$) due to the imposed stabilization scheme, the contribution of g to the control actions vanishes, thus resulting in a discontinuous control rule.

Unlike other optimization-based control methods, where an artificial cost function is minimized over a time horizon subject to constraints, GPLC control is defined entirely by the active subset of a full set of inequality constraints at any instant. The objective function in deriving the GPLC control is the quadratic form of the unconstrained dynamics $\mathbf{M}\mathbf{a} = \mathbf{f}$, which is automatically satisfied at all times according to physical laws. Table 1 summarizes some main differences between optimization-based control methods found in the literature and the GPLC control approach.

Table 1: Feature comparison between optimization-based control and GPLC control

Feature	Optimization-based	GPLC
involves a time horizon	✓	✗
has a cost function	✓	✗
relaxed constraints	✗	✓
physically meaningful control parameters	✗	✓
involves iterative solution	✓	✗
involves linearization	✓	✗
solution uniqueness guarantee	✗	✓

3 Collision Control of Multi-Agent Navigation Robots

In this section, various constraint formulations are presented to achieve different control structures (i.e., centralized vs. decentralized) and control performance on collision avoidance (e.g., deadlock vs. deadlock resolution).

We model N robots as two-dimensional (2-D) double integrators to demonstrate the fundamental performance of the proposed GPLC control. Each robot i , $i = 1, \dots, N$, is coordinated at $\mathbf{p}_i(t) = [x_i(t) \ y_i(t)]^\top \in \mathbb{R}^2$ with a mass m_i and is subjected to the unconstrained equations of motion

$$\begin{bmatrix} m_i & 0 \\ 0 & m_i \end{bmatrix} \begin{bmatrix} \ddot{x}_i \\ \ddot{y}_i \end{bmatrix} = \begin{bmatrix} f_i^x \\ f_i^y \end{bmatrix}, \quad (15)$$

where f_i^x and f_i^y are the external forces on agent i along X and Y axis.

We investigate the rudimentary case for collision avoidance in a 2-D multi-agent robotic system, namely, two robots subject to head-on collision under their maximum attainable speeds. This is the worst-case scenario for the two-robot setting. Each robot is enclosed within a virtual buffer of different shapes, which is determined by the constraint formulation. If the buffer size is greater than the maximum violation of this most extreme scenario, then all active collision avoidance constraints do not lead to actual inter-agent collisions. Note that the two robots can be any agent pair whose virtual buffers are actively colliding in the multi-agent system, thus we assign indices i and j to them.

In the following, the subscripts i and j denote the indices of agent i and agent j , respectively, while the sub/superscripts u, l, x, y, n, v, c, and d respectively denote the upper branch, lower branch, X component, Y component, vector-norm-based, vector-component-based, centralized, and decentralized of the corresponding variables.

3.1 Constraints Defined on Vector Norms

We use centroid-to-centroid relative distances as the metric for collision avoidance. This results in each robot being encompassed in a circular virtual buffer with a radius of r_i . Mathematically, the collision avoidance constraint is expressed in terms of a vector norm [10]

$$g_{ij} \triangleq d_{ij}^2 - (x_i - x_j)^2 - (y_i - y_j)^2 = d_{ij}^2 - \Delta \mathbf{p}_{ij}^\top \Delta \mathbf{p}_{ij} \leq 0, \quad (16)$$

where $d_{ij} = r_i + r_j$ is the virtual safety distance between the two possibly heterogeneous robots, and the notation $\Delta \mathbf{p}_{ij} \triangleq \mathbf{p}_i - \mathbf{p}_j = -\Delta \mathbf{p}_{ji}$. Without loss of generality, we investigate a homogeneous multi-agent robotic system in the present and following sections.

Differentiating (16) with respect to time once and twice, respectively, we obtain

$$\begin{aligned} \dot{g}_{ij} &= 2\Delta \mathbf{p}_{ji}^\top \Delta \dot{\mathbf{p}}_{ij} \\ \ddot{g}_{ij} &= 2\Delta \mathbf{p}_{ji}^\top \Delta \ddot{\mathbf{p}}_{ij} + 2\Delta \dot{\mathbf{p}}_{ji}^\top \Delta \dot{\mathbf{p}}_{ij} \end{aligned} \quad (17)$$

which, along with (16), are substituted into (12) to stabilize g_{ij} .

Given the physical parameters of the robots (mass m , characteristic length L , maximum attainable speed \bar{V}), we can perform a systematic dimensional study

to choose proper values of natural frequency ω , damping ratio ζ , and virtual buffer radius r for the least-effort collision avoidance according to the method presented in [10]. In Section 4, we will present an analytical approach to select control parameters ω , ζ , and r .

Note that the vector-norm-based constraint (16) is in essence one dimensional (i.e., colinear with $\Delta\mathbf{p}_{ij}$) and thus is unable to resolve deadlocks, in which the collision avoidance constraints prevent robots from pursuing their virtual leader with nonzero control actions. Deadlocks can happen, e.g., when the collision avoidance constraint is colinear with the vector-norm-based leader following constraint but in an opposite direction.

Furthermore, given the fact that robots can take on different geometries in a multi-agent system, it is important to allow the degree of flexibility to account for non-uniform buffer size along different axes. On the contrary, circular virtual buffers will be overly conservative if a robot has a much larger size along one axis than that in another axis.

Due to the aforementioned reasons, we define collision avoidance in terms of vector components in the next section to resolve deadlocks and to address possibly non-uniform-shaped robots.

3.2 Constraints Defined on Vector Components

We enclose each robot within a rectangular virtual buffer and describe collision avoidance in terms of vector components [11]

$$\mathbf{g}_{ij} = [g_{ij}^x \ g_{ij}^y]^\top \triangleq |\Delta\mathbf{p}_{ij}| = |\mathbf{p}_i - \mathbf{p}_j| \geq \mathbf{r}_{ij} \ , \quad (18)$$

where $|\cdot|$ denotes absolute value, $\mathbf{r}_{ij} = \mathbf{r}_{ji} = \mathbf{r}_i + \mathbf{r}_j = [r_{ij}^x \ r_{ij}^y]^\top \in \mathbb{R}^2$ are the sizes of the rectangular virtual buffers for actively colliding agent pair $\{i, j\}$ along the X and Y axes, and the inequality sign denotes elementwise comparison.

Decomposing constraints (18) into two branches: an upper and a lower bound on \mathbf{p}_i , we obtain

$$\mathbf{g}_{ij}^u = [g_{ij}^{xu} \ g_{ij}^{yu}]^\top \triangleq \{\Delta\mathbf{p}_{ij} \leq -\mathbf{r}_{ij}\}, \quad \mathbf{g}_{ij}^l = [g_{ij}^{xl} \ g_{ij}^{yl}]^\top \triangleq \{\Delta\mathbf{p}_{ji} \leq -\mathbf{r}_{ij}\}. \quad (19)$$

Therefore, the upper and lower branch of active \mathbf{g}_{ij} for agent i is respectively

$$\text{active } \mathbf{g}_{ij}^u = \{-\mathbf{r}_{ij} < \Delta\mathbf{p}_{ij} \leq \mathbf{0}\}, \quad \text{active } \mathbf{g}_{ij}^l = \{\mathbf{0} < \Delta\mathbf{p}_{ij} < \mathbf{r}_{ij}\}. \quad (20)$$

Note that the strict equality sign in (20) is assigned to the active upper branch in order to break the perfect symmetry of the active constraint region.

Hence, the first- and second-order derivatives of the decomposed collision avoidance constraints are

$$\dot{\mathbf{g}}_{ij}^u = -\dot{\mathbf{g}}_{ij}^l = \Delta\dot{\mathbf{p}}_{ij}, \quad \ddot{\mathbf{g}}_{ij}^u = -\ddot{\mathbf{g}}_{ij}^l = \Delta\ddot{\mathbf{p}}_{ij}, \quad (21)$$

which are substituted into (12) for constraint stabilization.

Two robots are identified to have colliding virtual buffers when both $g_{ij}^x \in [g_{ij}^{xu} \ g_{ij}^{xl}]$ and $g_{ij}^y \in [g_{ij}^{yu} \ g_{ij}^{yl}]$ are active. This indicates that the collision avoidance control actions corresponding to (19) are in a 2-D plane, while those corresponding to (16) are along a line \mathbf{p}_{ij} . Therefore, vector-component-based constraint formulation (18) can resolve deadlocks and non-uniform-shaped issues naturally, given that proper control parameters are selected.

3.3 Constraints Partitioning

When the number of robots in the multi-agent system increases significantly, a decentralized control architecture becomes more favorable and even mandatory. Both the vector-norm-based and vector-component-based constraint formulations introduced in Section 3.1 and 3.2 involve interacting agents and thus are not readily applicable to a decentralized framework.

To develop a decentralized architecture while maintaining the advantages posed by the vector-component-based approach, we partition the collision avoidance constraints (19) between colliding robot pairs $\{i, j\}$ [12]

$$\ddot{\mathbf{g}}_{ij}^u = \Delta\ddot{\mathbf{p}}_{ij} = 2\ddot{\mathbf{p}}_i, \quad \ddot{\mathbf{g}}_{ij}^l = -\Delta\ddot{\mathbf{p}}_{ij} = -2\ddot{\mathbf{p}}_i, \quad (22)$$

by assuming that each agent i have access to the relative positions $\Delta\mathbf{p}_{ij}$ and the relative velocities $\Delta\dot{\mathbf{p}}_{ij}$ with respect to its neighbor j , and that $\ddot{\mathbf{p}}_i = -\ddot{\mathbf{p}}_j$ (due to the homogeneity assumption introduced earlier) in a collision avoidance.

Equation (22) implies that the \mathbf{A} matrix of the vector-component-based, decentralized framework is scaled by a factor of 2 compared to its counterpart of the vector-component-based, centralized framework presented in Section 3.2 when constructing the KKT system (14) and solving for the control forces $-\mathbf{A}^\top\boldsymbol{\lambda}$.

4 Results and Discussions

Table 2: Comparison of three different constraint formulations

Formulation	Framework	Constraint	Section
1	centralized	vector-norm-based	3.1
2	centralized	vector-component-based	3.2
3	decentralized	vector-component-based	3.3

We compare the performance of three constraint formulations as listed in Table 2 by numerically and analytically investigating two robots subject to maximum-speed, head-on collisions under the same simulation settings. The initial positions and the initial velocities for the two robots are respectively

$$\mathbf{p}_1(0) = \begin{bmatrix} -10 \\ 0 \end{bmatrix}, \quad \mathbf{p}_2(0) = \begin{bmatrix} 10 \\ 0 \end{bmatrix}, \quad \dot{\mathbf{p}}_1(0) = \begin{bmatrix} \bar{V} \\ 0 \end{bmatrix}, \quad \dot{\mathbf{p}}_2(0) = \begin{bmatrix} -\bar{V} \\ 0 \end{bmatrix}. \quad (23)$$

The numerical studies are conducted in Matlab using a fourth order Runge-Kutta integrator with a constant integration time step Δt . The parameters used in these studies are presented in Table 3.

In the numerical simulations, we compute the instantaneous power due to the control forces, $(-\mathbf{A}^\top\boldsymbol{\lambda})^\top\dot{\mathbf{q}}$, where $\dot{\mathbf{q}}$ contains all coordinate velocities, and integrate it to obtain the time series of the total energy due to the work done by the control forces. Since we assume there exists no other forces, the kinetic energy change results purely from the control forces when robots avoid collisions. When the two

Table 3: The parameters and their values used in a two-robot system subject only to initial velocities and collision avoidance constraints.

Quantity	Variable (Unit)	Value
mass	m (kg)	10
maximum speed	\bar{V} (m/s)	10
buffer size	r (m)	6
natural frequency	ω (rad/s)	2
simulation time	T (s)	4
integration step	Δt (s)	0.005

virtual buffers are not in collision, the two agents remain in constant speeds at the repelling speeds when the two virtual buffers instantly detach.

For better comparison, the virtual buffer size along X and Y axis for Formulations 2 and 3 are set to equal, leading to the circular virtual buffer of Formulation 1 being inscribed to the square virtual buffers of Formulations 2 and 3. Note that this problem is of perfect symmetry.

4.1 Underdamped Constraint Dynamics, $\zeta = 0.5$

Figure 1 illustrates the numerical results for two robot under head-on maximum-speed collision, where the only constraint, two agents avoid collision, is subjected to underdamped constraint stabilization dynamics with a $\zeta = 0.5$. The animation of robots' position trajectories and the X components of their control forces can be found online.¹

Figure 1a clearly shows that Formulation 1 cannot resolve deadlocks as the colliding and repelling velocities are colinear, while both Formulations 2 and 3 that use vector-component-based constraints can handle deadlocks as both agents turn to their right and away from each other given the control law. Within the same time frame, Formulation 2 achieves the largest departure distances. This is also supported by Figure 1g that Formulation 2 results in the largest constant speed after collision avoidance. Note that no traffic rule is specified for both robots. The "turn-to-right" behaviors result purely from the natural evolution of system dynamics subject to the constraints (19).

Figures 1c and 1d indicates that the onset of collision avoidance occurs at $t = 0.4$ seconds and the inherent nature of nonlinear and discontinuous for the proposed methodology. Figures 1g and 1h demonstrate that the robots move in constant velocities after virtual buffers depart from each other.

It is evident that Formulation 3 has a slightly faster response time, and both Formulations 2 and 3 achieve collision avoidance more than 0.6 seconds faster than that of Formulation 1. Further, Formulation 3 achieves almost the same peak magnitude as that of Formulation 1 but with a faster peak time, and Formulation 3 arrives at the same peak time as that of Formulation 2 but with about a twice magnitude. This is due to the fact that the \mathbf{A} matrix in Formulation 3 is scaled by a factor of 2 compared to that of Formulation 2 and that both Formulations 2 and

¹ <https://youtu.be/Gkg-I2PrMWs>

3 have maneuverability in a 2-D plane rather than a 1-D line as for Formulation 1, which is verified in Figures 1f and 1h since the position and velocity in Y direction stays unchanged $\forall t$.

It is also evident from Figures 1e and 1f that Formulation 3 attains the largest relative distance and the fastest peak response while Formulation 1 achieves the smallest relative distance and the slowest peak response.

It is clear from Figure 1b that all three Formulations dissipate kinetic energies during collision avoidance, which is also supported by the observation that the constant velocities after collision avoidance are less than the initial colliding velocities. Because the imposed underdamped constraint dynamics exhibits oscillatory responses of g , the control forces, the velocities, and thus the total energy also exhibit oscillations.

Based on the observations on Figure 1 and the above discussions, we may conclude that Formulation 2 has the best overall collision avoidance performance due to its fast response time, the least maximum control force magnitude (the least work done), and the largest departure distance.

An Analytical Perspective. The initial conditions for the second-order constraint dynamics (12) with $0 < \zeta < 1$ can be analytically determined given the initial positions and initial velocities of the two colliding agents at the onset of constraint violation at $t = t_o$ [10]

$$\begin{bmatrix} g(t) - c \\ \dot{g}(t) \end{bmatrix} = e^{-\zeta\omega t} \begin{bmatrix} \cos(\omega_d t) + \eta \sin(\omega_d t) & \frac{1}{\omega_d} \sin(\omega_d t) \\ \frac{-\omega_d}{\sqrt{1-\zeta^2}} \sin(\omega_d t) & \cos(\omega_d t) - \eta \sin(\omega_d t) \end{bmatrix} \begin{bmatrix} g(t_o) - c \\ \dot{g}(t_o) \end{bmatrix}, \quad (24)$$

where $\eta = \zeta/\sqrt{1-\zeta^2}$, $\omega_d = \omega\sqrt{1-\zeta^2}$ denotes the damped natural frequency of the constraint dynamics.

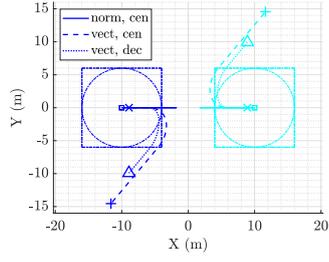
By substituting the first time instant t_{\max} solved from setting $\dot{g}(t) = 0$ into (24) for $g(t)$, we can obtain the maximum constraint value g_{\max} as

$$g_{\max} = \frac{\dot{g}(t_o)}{\omega} e^{\frac{-\zeta}{\sqrt{1-\zeta^2}} \arctan \frac{\sqrt{1-\zeta^2}}{\zeta}} + c \quad (25)$$

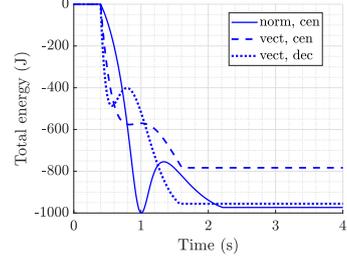
for initial conditions $g(t_o) - c = 0$ and $\dot{g}(t_o) > 0$.

In this case study, $t_o = (10-6)/10 = 0.4$ second, $g(t_o) = [6 - (-6)]^2 = 144 \text{ m}^2$, $\dot{g}(t_o) = -2 \times [6 - (-6)] \times [10 - (-10)] = 480 \text{ m}^2/\text{s}$, and $g_{\max} = c = (2r)^2 = 144 \text{ m}^2$ when the two robots have their centroids overlap with each other. Substituting these numeric values into (25) yields the critical frequency $\omega_{\text{crit}} \approx 1.821 \text{ rad/s}$ for Formulation 1. If the actual robot collision occurs at a minimum centroid-to-centroid distance squared $\|\Delta \mathbf{p}_{ij}\|_2^2 = 36 \text{ m}^2$, then it is equivalent to setting $g_{\max} = (2r)^2 - 36 = 108 \text{ m}^2$. Substituting this g_{\max} into (25) yields $\omega_{\min} \approx 2.428 \text{ rad/s}$ for $\zeta = 0.5$.

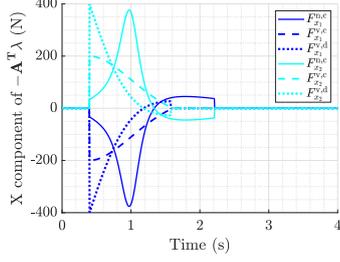
Figure 2 illustrates the simulated trajectories subject to $\zeta = 0.5$ and the analytically determined ω_{crit} and ω_{\min} for Formulation 1. The results are desired that reflect the analytical solutions. Figure 2 demonstrates that both Formulations 2 and 3 have nonzero pairwise distances in X and Y directions when the pairwise distance is close to zero for Formulation 1.



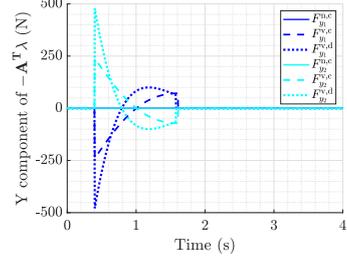
(a) \square : initial positions; \times , $+$, \triangle : end positions.



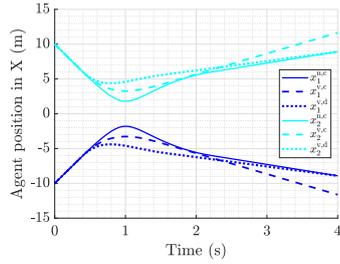
(b) Total energy done by control forces.



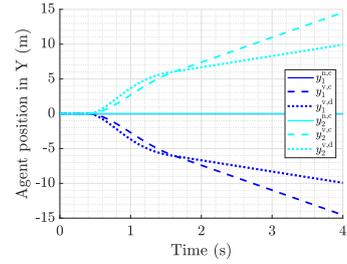
(c) X component of control forces.



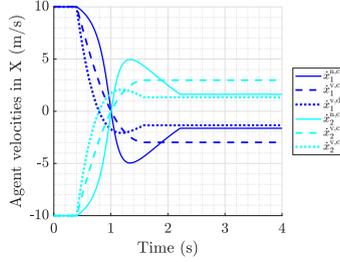
(d) Y component of control forces.



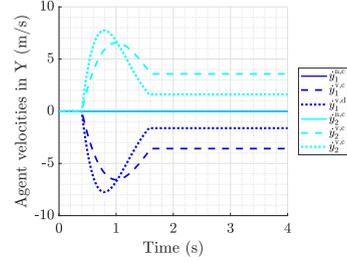
(e) Positions along X axis.



(f) Positions along Y axis.

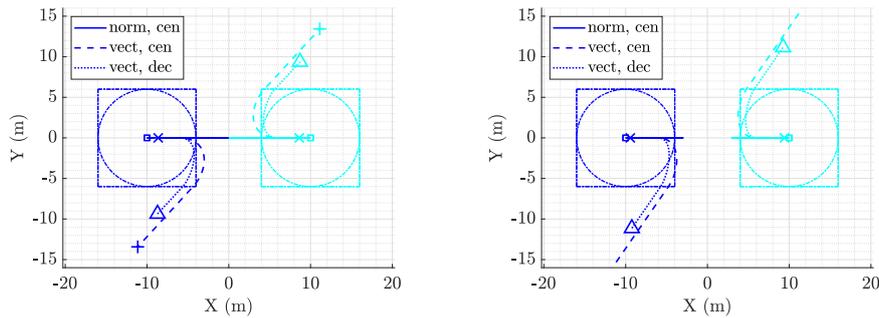


(g) Velocities along X axis.



(h) Velocities along Y axis.

Fig. 1: Comparison of three constraint formulations (**solid lines**: vector-norm-based, centralized; **dashed lines**: vector-norm-based, centralized; **dotted lines**: vector-component-based, decentralized) for 2 robots subject to **underdamped** constraint dynamics under the same settings. The superscripts n, v, c, d stands for norm, vector, centralized, and decentralized, respectively. The **dash-dot lines** in (a) denote the boundaries of the virtual buffers at $t = 0$ second.



(a) Critical natural frequency $\omega_{\text{crit}} = 1.821$ rad/s for Formulation 1 when the two robots have their centroids overlap. (b) Minimum pairwise distance is 6 m for Formulation 1 using the analytically determined $\omega_{\text{min}} = 2.428$ rad/s.

Fig. 2: Vector-norm-based constraint formulation for 2 robots subject to **underdamped** constraint dynamics under different natural frequencies.

Similarly, by noting the constraint activation criteria (20) for the vector-component-based constraints (18), the initial conditions along X and Y axis are respectively $g^x(t_o^x) = 6 - (-6) = 12$ m, $\dot{g}^x(t_o^x) = 10 - (-10) = 20$ m/s, $g_{\text{max}}^x = c^x = 2r^x = 12$ m, $g^y(t_o^y) = 0 - 0 = 0$ m, $\dot{g}^y(t_o^y) = 0 - 0 = 0$ m/s, and $g_{\text{max}}^y = c^y = 2r^y = 12$ m for Formulations 2 and 3. Note that the virtual buffers of the two robots are in collision when both g_{ij}^x and g_{ij}^y are active. Thus, the onset time $t_o^x = t_o^y = t_o$. Substituting the initial conditions into (24) we can see that the constraint dynamics along X and Y axis has the same angular frequencies ω_d and exponential decay rate $e^{-\zeta\omega t}$.

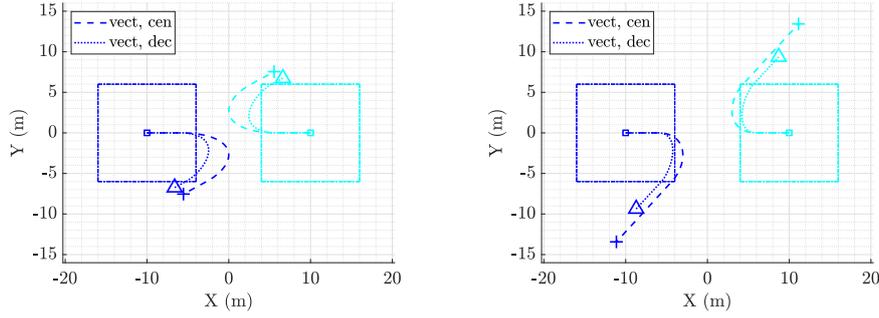
Hence, we can analytically compute the critical frequency $\omega_{\text{crit}} \approx 0.911$ rad/s at which the relative distance in X axis is zero and the minimum frequency $\omega_{\text{min}} \approx 1.821$ rad/s to ensure a 6-m minimum relative distance in X axis for Formulation 2. Figure 3 numerically verifies the analytical solutions for ω_{crit} and ω_{min} . Note that the pairwise distances of Formulation 3 is always larger than those of Formulation 2 in both cases, with a cost of higher magnitude of peak control forces.

4.2 Overdamped Constraint Dynamics, $\zeta = 1.1$

Numerical results of the same types as in Figure 1 for the same two-robot system under head-on collision subject to an overdamped constraint dynamics with a $\zeta = 1.1$ are shown in Figure 4. The animation of robots' position trajectories and the X components of their control forces is available online.²

Since the collision constraint dynamics is overdamped, g approaches its constant threshold (in this case, the size of virtual buffer or its squared, depending on which Formulation is used) as t goes to ∞ . Therefore, the end positions of all three Formulations stay where the virtual buffers are close to but never detach from each other (due to the fact that the constraint dynamics is overdamped)

² <https://youtu.be/iOu5oBTdPYc>



(a) Critical natural frequency $\omega_{\text{crit}} = 0.911$ rad/s for Formulation 2 when the relative distance in X axis is zero. (b) Min pairwise distance is 6 m in X axis for Formulation 2 using the analytically determined $\omega_{\text{min}} = 1.821$ rad/s.

Fig. 3: Vector-component-based constraint formulations for 2 robots subject to **underdamped** constraint dynamics under different natural frequencies.

subject to velocities close to an infinitesimally positive value. This also explains why all three Formulations dissipate almost the same amount of energy in Figure 4b during collision avoidance, since both initial and end velocities are the same and close to each other, respectively.

Based on the observations on Figure 4 and the above discussions, we may conclude that both Formulations 2 and 3 have similarly better collision avoidance performance in terms of response time and departure distances than those of Formulation 1 and that the maximum control force magnitude (the amount of work done) of Formulation 3 is about twice that of Formulation 2. Therefore, Formulation 2 again has the best overall performance.

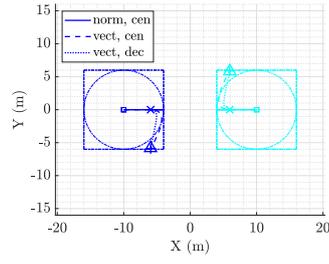
An Analytical Perspective. For this overdamped case with $\zeta = 1.1$, we can analogously solve for the critical frequency for actual collision from the analytical solution of the maximum constraint value

$$g_{\text{max}} = \frac{\dot{g}(t_o)}{\omega} e^{\frac{-\zeta}{\sqrt{\zeta^2-1}}} \ln(\zeta + \sqrt{\zeta^2-1}). \quad (26)$$

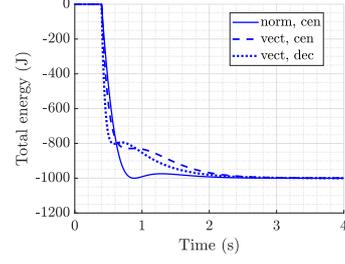
for initial conditions $g(t_o) - c = 0$ and $\dot{g}(t_o) > 0$.

Hence, for Formulation 1, $\omega_{\text{crit}} \approx 1.149$ rad/s, and $\omega_{\text{min}} \approx 1.533$ rad/s, and for Formulations 2 and 3, $\omega_{\text{crit}} \approx 0.575$ rad/s, and $\omega_{\text{min}} \approx 1.149$ rad/s. Figure 3 presents the trajectories subject to $\zeta = 1.1$ and the analytically determined ω_{crit} and ω_{min} for Formulations 2 and 3. The numerical results are desired that reflect the analytical solutions.

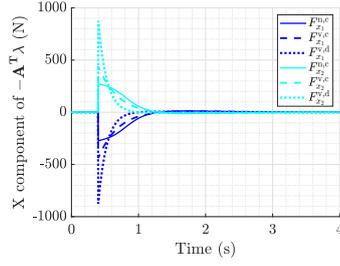
Note that a square-shaped virtual buffer is assumed with the same valued ζ and ω along X and Y directions in Formulations 2 and 3. Since the minimum pairwise distance along X axis, 6 m, is half the size of the virtual buffer in X direction, $2r = 12$ m, for all three Formulations, ω_{min} in Formulation 2 coincides with ω_{crit} in Formulation 1. Note that for all three Formulations, the virtual buffers will never detach completely (i.e., $g - c$ is always greater than 0) in collision avoidance since $\zeta > 1$.



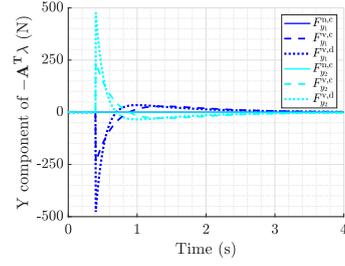
(a) \square : initial positions; \times , $+$, \triangle : end positions.



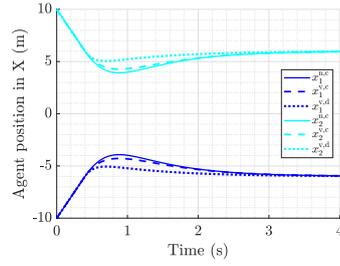
(b) Total energy done by control forces.



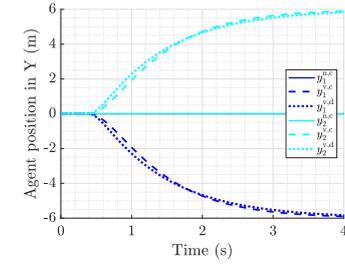
(c) X component of control forces.



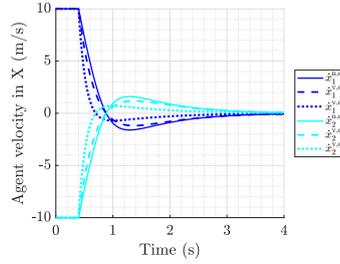
(d) Y component of control forces.



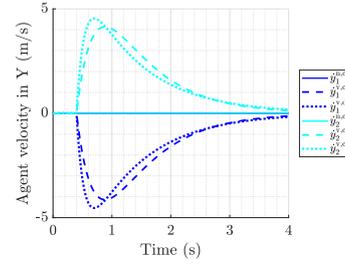
(e) Positions along X axis.



(f) Positions along Y axis.

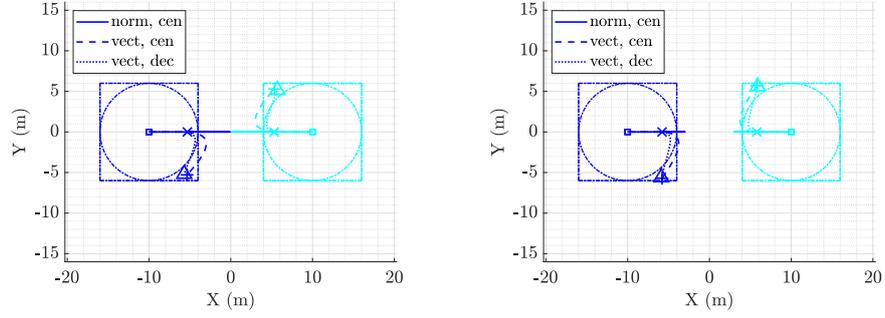


(g) Velocities along X axis.



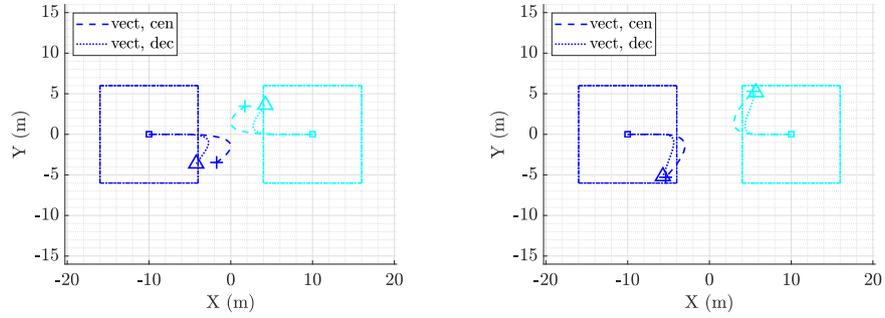
(h) Velocities along Y axis.

Fig. 4: Comparison of three constraint formulations (**solid lines**: vector-norm-based, centralized; **dashed lines**: vector-norm-based, centralized; **dotted lines**: vector-component-based, decentralized) for 2 robots subject to **overdamped** constraint dynamics under the same settings. The superscripts n, v, c, d stands for norm, vector, centralized, and decentralized, respectively. The **dash-dot lines** in (a) denote the boundaries of the virtual buffers at $t = 0$ second.



(a) Critical natural frequency $\omega_{\text{crit}} = 1.149$ rad/s for Formulation 1 when the two robots have their centroids overlap. (b) Minimum pairwise distance is 6 m for Formulation 1 using the analytically determined $\omega_{\text{min}} = 1.533$ rad/s.

Fig. 5: Vector-norm-based constraint formulation for 2 robots subject to **over-damped** constraint dynamics under different natural frequencies.



(a) Critical natural frequency $\omega_{\text{crit}} = 0.575$ rad/s for Formulation 2 when the relative distance in X axis is zero. (b) Min pairwise distance is 6 m in X axis for Formulation 2 using the analytically determined $\omega_{\text{min}} = 1.149$ rad/s.

Fig. 6: Vector-component-based constraint formulations for 2 robots subject to **overdamped** constraint dynamics under different natural frequencies.

5 Conclusions and Future Work

In this paper, a versatile control approach is presented for multi-agent navigation robotic systems with an emphasis on collision avoidance among agents. The method is based on extending Gauss’s principle of least constraint (GPLC) with dynamically identifying, incorporating, and stabilizing active inequality constraints. The control law results from solving a KKT system, a linear matrix equation, without iteration or linearization at each point in time. The state trajectories are controlled entirely by constraints that describe system behaviors without looking ahead to a time horizon and without minimizing an artificial cost function. These unique features bestow computational simplicity and efficiency to the presented methodology.

Three different constraint formulations are presented and compared on two robot subject to head-on maximum-speed collision using numerical and analytical approaches: **1)** constraint defined on *vector norms* in a *centralized* architecture, **2)** constraint determined on *vector components* in a *centralized* structure, and **3)** constraint specified by *vector components* in a *decentralized* framework. Each robot is enclosed in a virtual buffer whose active collision gives rise to control actions, making the collision avoidance constraints relaxed. The control parameters, the natural frequency and damping ratio of the constraint stabilization dynamics and the virtual buffer size, are physically meaningful and can be selected by an analytical method introduced in Section 4.

Preliminary results have demonstrated that the present GPLC control is robust to unknown external disturbances and errors in modeling, measurement, and estimation. Future research include systematic investigation of GPLC control’s robustness to those inaccuracies and adaptiveness to time-varying control parameters. Note that incorporating external disturbances, and modeling, measurement, and estimation errors into the GPLC control framework is a straightforward task and that these factors can break the perfect symmetry of the problem and thus contribute to resolve deadlocks. Another future line of research is to integrate nonlinear robot dynamics and actuator mechanisms into GPLC control synthesis for multi-agent navigation robots and other robotic applications.

References

1. Y. Liu and R. Bucknall, “A survey of formation control and motion planning of multiple unmanned vehicles,” *Robotica*, vol. 36, no. 7, pp. 1019–1047, 2018.
2. S. Huang, R. S. H. Teo, and K. K. Tan, “Collision avoidance of multi unmanned aerial vehicles: A review,” *Annual Reviews in Control*, vol. 48, pp. 147–164, 2019.
3. J. A. Douthwaite, S. Zhao, and L. S. Mihaylova, “Velocity obstacle approaches for multi-agent collision avoidance,” *Unmanned Systems*, vol. 7, no. 01, pp. 55–64, 2019.
4. M. T. Wolf and J. W. Burdick, “Artificial potential functions for highway driving with collision avoidance,” in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2008, pp. 3731–3736.
5. D. Mellinger, A. Kushleyev, and V. Kumar, “Mixed-integer quadratic program trajectory generation for heterogeneous quadrotor teams,” in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 477–483.
6. J. Grover, C. Liu, and K. Sycara, “Deadlock analysis and resolution in multi-robot systems (extended version),” *arXiv preprint arXiv:1911.09146*, 2019.

7. S. Ouahouah, M. Baga, J. Prados-Garzon and T. Taleb, "Deep-Reinforcement-Learning-Based Collision Avoidance in UAV Environment," *IEEE Internet of Things Journal*, vol. 9, no. 6, pp. 4015-4030, 2022.
8. C. F. Gauß, "Über ein neues allgemeines grundgesetz der mechanik," *J. fü Die Reine Und Angewandte Mathematik*, vol. 4, pp. 232-235, 1829.
9. F. E. Udwardia and R. E. Kalaba, "A new perspective on constrained motion," *Proc.: Math. Phys. Sci.*, vol. 439, no. 1906, pp. 407-410, 1992
10. Zhang, B. and Gavin, H.P. Gauss's Principle with Inequality Constraints for Multi-agent Navigation and Control. *IEEE Transactions on Automatic Control*, vol. 67, no. 2, pp. 810-823, 2022.
11. Zhang, B. and Gavin, H.P. Natural Deadlock Resolution for Multi-agent Multi-swarm Navigation. *Proceedings of the 60th IEEE Conference on Decision and Control (CDC)*, pp. 5958-5963, 2021.
12. Zhang, B. and Gavin, H.P. Decentralized Control of Multiagent Navigation Systems. *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 5, pp. 922-925, 2022.