

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2024.0429000

# A Logarithmic Barrier Function-Based Newton's Model for Solving Dynamic Quadratic Programming Subject to Inequality Constraints

**JIUFAN WANG<sup>1</sup>, YUQUN ZHOU<sup>2</sup>**

<sup>1</sup>Department of Integrated System Engineering, The Ohio State University, Columbus, OH, 43210 USA (e-mail: wang.18428@osu.edu)

<sup>2</sup>Department of Industrial & System Engineering, University of Wisconsin-Madison, Madison, WI 53706 USA (e-mail: yzhou364@wisc.edu)

Corresponding author: Yuqun Zhou (e-mail: yzhou364@wisc.edu).

**ABSTRACT** This paper proposes a logarithmic barrier function-based Newton's (LBFN) model designed to solve dynamic quadratic programming problems subject to inequality constraints in real time. Departing from conventional techniques that introduce auxiliary variables via Karush-Kuhn-Tucker optimality conditions, the LBFN model utilizes parametric and adaptive logarithmic barrier functions to directly handle dynamic inequality constraints. This strategy avoids the necessity of dual variables while enabling three major features: convergence from arbitrary starting points without feasibility restrictions, globally convergent dynamics insensitive to initial conditions, and exponential reduction of solution tracking error with tunable rates. By combining a prediction mechanism that anticipates the temporal drift of the optimal solution with a Newton-type correction strategy, the proposed model ensures persistent constraint satisfaction while accurately tracking the evolving optimal trajectory. Numerical simulations confirm enhanced performance in convergence speed and solution accuracy compared to several established solvers. Additionally, an application to the simultaneous trajectory tracking and obstacle avoidance task of robot further validates the practical utility of the LBFN model.

**INDEX TERMS** Dynamic quadratic programming, inequality constraints, logarithmic barrier function, global convergence

## I. INTRODUCTION

OPERATING in highly dynamic environments, modern engineering and control systems demand instantaneous adaptation of optimal decisions to rapidly changing conditions [1]–[3]. This necessity for real-time adaptive programming is particularly critical in domains such as robotic autonomous navigation [4]–[6] and intelligent traffic management systems [7], [8]. Effective control of these systems hinges on the continuous resolution of programming problems with time-varying objectives and constraints. As applications increase in scale and complexity, the development of practical methodologies capable of accurately tracking optimal solutions under dynamic variations has become indispensable [9]–[11].

Conventional strategies primarily address static programming problem formulations, where parameters remain fixed throughout the solution process [12], [13]. Techniques like gradient-flow dynamics utilize first-order descent principles for unconstrained minimization [14], [15], while Newton-type flows accelerate convergence by incorporating second-

order curvature information [16], [17]. Constrained problems often employ active set strategies [18], [19] and sequential quadratic programming [20], [21]. However, the above-mentioned approaches are proven effective in dealing with the static programming problems; these approaches encounter limitations when deployed in dynamic programming problems with persistent parameter drifts. Because they lack an effective predictive mechanism to compensate for the solution drifts [22], [23].

A critical challenge arises because static solvers lack temporal awareness. When applied to dynamic systems, treating each programming instance as an independent snapshot, they inevitably generate substantial tracking errors [24]–[26]. For example, consider a robotic manipulator that moves between points while avoiding dynamically moving obstacles [27], [28]. If its path planner uses a static optimizer without prediction, it would solve the problem only at the current instant based on the obstacle's current position. However, during the finite computation time, e.g., 50 ms, required to compute the optimal solution, both the robot and the obstacles have

moved. The solution computed at  $t_k$  is therefore optimized for a past state ( $t_k$ ) and applied at a future state ( $t_k + \text{computation time}$ ), resulting in sub-optimality or, worse, constraint violation if the obstacle moved unexpectedly close during that interval [29], [30]. Similarly, in automated lane-changing scenarios for autonomous vehicles, a static planner reacting only to the instantaneous relative positions of surrounding vehicles without anticipating their motion trends could fail to generate a safe and efficient trajectory when neighboring vehicles accelerate or decelerate unexpectedly during the planning computation [31]. These limitations highlight two critical gaps: a lack of inherent mechanism to compensate for the temporal drift of the optimal solution between the time computation starts and ends, and a reliance on strategies for handling inequality constraints with KKT-based approaches that are computationally expensive to recompute frequently or that introduce additional variables, complicating the real-time solution process [32], [33].

To address these limitations, this paper proposes a logarithmic barrier function-based Newton's (LBFN) Model for dynamic inequality-constrained quadratic programming. Departing from the conventional paradigm of converting inequalities into equality systems via KKT conditions which requires the introduction of dual variables, the LBFN model employs parametric adaptive log-barrier functions to directly incorporate dynamic inequality constraints. This formulation fundamentally avoids the need for auxiliary variables. The proposed model integrates a prediction mechanism that estimates the future drift of the optimal solution using time derivatives of the problem parameters, along with a Newton-based correction dynamics that drives the solution toward the optimum of the barrier-augmented objective while ensuring continuous constraint satisfaction throughout the evolution. The main contributions of this work are summarized as follows:

- A novel LBFN model is constructed for real-time solving of dynamic quadratic programming subject to inequality constraints (DQP-IC), which employs parametric adaptive log-barrier functions instead of KKT-based constraint handling.
- The LBFN model exhibits initialization insensitivity, enabling convergence to the optimal trajectory from arbitrary initial points (feasible or infeasible), while maintaining exponentially decaying tracking errors with a tunable convergence rate.
- Theoretical analysis, numerical simulations, and a robotic navigation application are presented to validate the effectiveness and advantages of the proposed approach.

**Notation:** For any dynamic vector  $\mathbf{v}(t) \in \mathbb{R}^n$ , the Euclidean norm is expressed as  $\|\mathbf{v}(t)\|_2$ . Consider a function  $\hbar(\mathbf{v}(t), t)$ . The optimal value of this function is denoted by  $\hbar(\mathbf{v}^*(t), t)$ , where  $\mathbf{v}^*(t)$  signifies the optimal trajectory of the decision variable  $\mathbf{v}(t)$ . The gradient of  $\hbar$  with respect to  $\mathbf{v}(t)$  is represented by  $\nabla_{\mathbf{v}}\hbar(\mathbf{v}(t), t)$ , and its partial derivative with

respect to time  $t$  is given by  $\partial_t\hbar(\mathbf{v}(t), t)$ . The Hessian matrix of  $\hbar$  with respect to  $\mathbf{v}(t)$  is denoted by  $\nabla_{\mathbf{v}\mathbf{v}}\hbar(\mathbf{v}(t), t)$ . Furthermore, the partial derivative of the gradient  $\nabla_{\mathbf{v}}\hbar(\mathbf{v}(t), t)$  with respect to  $t$  is written as  $\nabla_{\mathbf{v}t}\hbar(\mathbf{v}(t), t)$ . The  $i$ -th component of the vector  $\mathbf{v}(t)$  is indicated by  $v_i(t)$ .

## II. PRELIMINARY AND SCHEME FORMULATION

In this part, a dynamic quadratic programming subject to inequality constraints and the LBFN model preliminaries are provided in detail.

### A. CONSTRUCTION OF DQP-IC PROBLEM

Consider a DQP-IC problem formulated as:

$$\begin{aligned} & \underset{\mathbf{v}(t) \in \mathbb{R}^n}{\text{minimize}} \quad \hbar(\mathbf{v}(t), t) = \frac{1}{2} \mathbf{v}^\top(t) \mathcal{M}(t) \mathbf{v}(t) + \mathbf{g}^\top(t) \mathbf{v}(t) \\ & \text{subject to} \quad \mathcal{N}(t) \mathbf{v}(t) \leq \mathbf{d}(t) \end{aligned} \quad (1)$$

where  $\mathbf{v}(t) \in \mathbb{R}^n$  is a decision variable.  $t \geq 0$  denotes continuous time.  $\mathcal{M}(t) \in \mathbb{S}_{++}^n$  is a symmetric positive-definite matrix with  $\mathcal{M}(t) \succeq m_f \mathcal{I}_n$  for some  $m_f > 0$  (uniform strong convexity).  $\mathbf{g}(t) \in \mathbb{R}^n$  is a dynamic linear coefficient vector.  $\mathcal{N}(t) \in \mathbb{R}^{p \times n}$  and  $\mathbf{d}(t) \in \mathbb{R}^p$  define  $p$  linear inequality constraints. The feasible set  $\mathcal{F}(t) = \{\mathbf{v}(t) \in \mathbb{R}^n : \mathcal{N}(t)\mathbf{v}(t) \leq \mathbf{d}(t)\}$  satisfies Slater's condition for all  $t \geq 0$ .

To handle constraints, we introduce a dynamic barrier parameter  $w(t) > 0$  with  $\lim_{t \rightarrow \infty} w(t) = \infty$ , and a dynamic slack variable  $r(t) > 0$  with  $r(t) = r(0)e^{-\gamma_r t}$ ,  $\gamma_r > 0$ , ensuring  $r(t) \rightarrow 0$  asymptotically. The modified objective  $\Upsilon(\mathbf{v}(t), w(t), r(t), t)$  is defined as:

$$\begin{aligned} \Upsilon(\mathbf{v}(t), w(t), r(t), t) = & \hbar(\mathbf{v}(t), t) - \frac{1}{w(t)} \sum_{i=1}^p \log \left( r(t) \right. \\ & \left. - (\mathcal{A}_i(t)\mathbf{v}(t) - b_i(t)) \right) \end{aligned} \quad (2)$$

where  $\mathcal{A}_i(t)$  is the  $i$ -th row of  $\mathcal{N}(t)$ . The domain  $\widehat{\mathcal{D}}(t) = \{\mathbf{v}(t) \in \mathbb{R}^n : \mathcal{A}_i(t)\mathbf{v}(t) - b_i(t) < r(t)\}$  is non-empty when  $r(0) > \max_i(\mathcal{A}_i(0)\mathbf{v}(0) - b_i(0))$ .

### B. GENERAL FRAMEWORK OF LBFN MODEL

Then, an implicit evolution dynamics is constructed as:

$$\dot{\mathbf{v}}(t) = -\nabla_{\mathbf{v}\mathbf{v}}^{-1} \Upsilon [\alpha \nabla_{\mathbf{v}\mathbf{v}} \Upsilon + \nabla_{\mathbf{v}r} \Upsilon \dot{r} + \nabla_{\mathbf{v}w} \Upsilon \dot{w} + \nabla_{\mathbf{v}t} \Upsilon] \quad (3)$$

where  $\alpha > 0$  is a convergence rate parameter.  $\nabla_{\mathbf{v}\mathbf{v}} \Upsilon$  is the Hessian of  $\Upsilon$  w.r.t.  $\mathbf{v}(t)$ .  $\nabla_{\mathbf{v}r} \Upsilon$ ,  $\nabla_{\mathbf{v}w} \Upsilon$ , and  $\nabla_{\mathbf{v}t} \Upsilon$  are partial derivatives.

Define  $\delta_i(\mathbf{v}(t), t) = r(t) - (\mathcal{A}_i(t)\mathbf{v}(t) - b_i(t))$ , and the following definitions are given:

$$\nabla_{\mathbf{v}} \Upsilon = \mathcal{M}(t) \mathbf{v}(t) + \mathbf{g}(t) + \frac{1}{w(t)} \sum_{i=1}^p \frac{\mathcal{A}_i(t)^\top}{\delta_i(\mathbf{v}(t), t)} \quad (4)$$

$$\nabla_{\mathbf{v}\mathbf{v}} \Upsilon = \mathcal{M}(t) + \frac{1}{w(t)} \sum_{i=1}^p \frac{\mathcal{A}_i(t)^\top \mathcal{A}_i(t)}{\delta_i^2(\mathbf{v}(t), t)} \quad (5)$$

$$\nabla_{vr} \Upsilon = \frac{1}{w(t)} \sum_{i=1}^p \frac{\mathcal{A}_i(t)^\top}{\delta_i^2(\mathbf{v}(t), t)} \quad (6)$$

$$\nabla_{vw} \Upsilon = -\frac{1}{w^2(t)} \sum_{i=1}^p \frac{\mathcal{A}_i(t)^\top}{\delta_i(\mathbf{v}(t), t)} \quad (7)$$

$$\begin{aligned} \nabla_{vt} \Upsilon &= \dot{\mathcal{Q}}(t)\mathbf{v}(t) + \dot{\mathbf{g}}(t) + \frac{1}{w(t)} \sum_{i=1}^p \frac{\dot{\mathcal{A}}_i(t)^\top}{\delta_i(\mathbf{v}(t), t)} \\ &\quad + \frac{1}{w(t)} \sum_{i=1}^p \frac{\mathcal{A}_i(t)^\top (\dot{\mathcal{A}}_i(t)\mathbf{v}(t) - \dot{b}_i(t))}{\delta_i^2(\mathbf{v}(t), t)} \end{aligned} \quad (8)$$

Substituting into (3), it is eventually obtained the following LBFN model:

$$\begin{aligned} \dot{\mathbf{v}}(t) &= - \left( \mathcal{M}(t) + \frac{1}{w(t)} \sum_{i=1}^p \frac{\mathcal{A}_i(t)^\top \mathcal{A}_i(t)}{\delta_i^2(\mathbf{v}(t), t)} \right)^{-1} \left( \alpha \left( \mathcal{M}(t)\mathbf{v}(t) \right. \right. \\ &\quad \left. \left. + \mathbf{g}(t) + \frac{1}{w(t)} \sum_{i=1}^p \frac{\mathcal{A}_i(t)^\top}{\delta_i(\mathbf{v}(t), t)} \right) \dot{r}(t) \right. \\ &\quad \left. + \left( \frac{1}{w(t)} \sum_{i=1}^p \frac{\mathcal{A}_i(t)^\top}{\delta_i^2(\mathbf{v}(t), t)} \right) \dot{r}(t) \right. \\ &\quad \left. + \left( -\frac{1}{w^2(t)} \sum_{i=1}^p \frac{\mathcal{A}_i(t)^\top}{\delta_i(\mathbf{v}(t), t)} \right) \dot{w}(t) + \nabla_{vt} \Upsilon \right) \end{aligned} \quad (9)$$

where  $\dot{r}(t) = -\gamma_r r(t)$ , and  $\dot{w}(t) = \gamma_w w(t)$  if  $w(t) = w(0)e^{\gamma_w t}$ .

### III. THEORETICAL ANALYSIS

This part provides the theoretical analysis for the LBFN model (9).

**Theorem:** Consider the LBFN model (9) for the DQP-IC problem (1). Define the Lyapunov function candidate:

$$V(t) = \frac{1}{2} \|\nabla_{\mathbf{v}} \Upsilon(\mathbf{v}(t), w(t), r(t), t)\|_2^2 \quad (10)$$

For any initial condition  $\mathbf{v}(0) \in \widehat{\mathcal{D}}(0)$ ,  $V(t)$  satisfies:

$$\dot{V}(t) \leq -2\alpha V(t) \quad (11)$$

Moreover, the system is globally asymptotically stable, i.e.,

$$\lim_{t \rightarrow \infty} \|\nabla_{\mathbf{v}} \Upsilon(\mathbf{v}(t), w(t), r(t), t)\|_2 = 0. \quad (12)$$

**Proof:** The time derivative of  $V(t)$  is computed as:

$$\dot{V}(t) = \nabla_{\mathbf{v}} \Upsilon^\top \left( \frac{d}{dt} \nabla_{\mathbf{v}} \Upsilon \right) \quad (13)$$

The total derivative  $\frac{d}{dt} \nabla_{\mathbf{v}} \Upsilon$  expands to:

$$\frac{d}{dt} \nabla_{\mathbf{v}} \Upsilon = \nabla_{vv} \Upsilon \dot{\mathbf{v}} + \nabla_{vr} \Upsilon \dot{r} + \nabla_{vw} \Upsilon \dot{w} + \nabla_{vt} \Upsilon \quad (14)$$

Substitute the dynamics (3) into (14):

$$\begin{aligned} \frac{d}{dt} \nabla_{\mathbf{v}} \Upsilon &= \nabla_{vv} \Upsilon \left( -\nabla_{vv}^{-1} \Upsilon (\alpha \nabla_{\mathbf{v}} \Upsilon + \nabla_{vr} \Upsilon \dot{r} + \nabla_{vw} \Upsilon \dot{w} \right. \\ &\quad \left. + \nabla_{vt} \Upsilon) \right) + \nabla_{vr} \Upsilon \dot{r} + \nabla_{vw} \Upsilon \dot{w} + \nabla_{vt} \Upsilon \end{aligned} \quad (15)$$

Simplify the expression:

$$\frac{d}{dt} \nabla_{\mathbf{v}} \Upsilon = -\alpha \nabla_{\mathbf{v}} \Upsilon - \nabla_{vt} \Upsilon + \nabla_{vt} \Upsilon = -\alpha \nabla_{\mathbf{v}} \Upsilon \quad (16)$$

The cancellation occurs because the prediction terms ( $\nabla_{vr} \Upsilon \dot{r} + \nabla_{vw} \Upsilon \dot{w} + \nabla_{vt} \Upsilon$ ) are exactly compensated by the additional derivatives in the total derivative. This is a key feature of the prediction-correction structure.

Substitute (16) into (13):

$$\dot{V}(t) = \nabla_{\mathbf{v}} \Upsilon^\top (-\alpha \nabla_{\mathbf{v}} \Upsilon) = -2\alpha V(t) \quad (17)$$

Solving this ODE gives [34]:

$$V(t) = V(0)e^{-2\alpha t} \quad (18)$$

which implies  $\lim_{t \rightarrow \infty} V(t) = 0$ , proving global asymptotic stability.

### IV. EXPERIMENTAL RESULTS

This section provides two examples to demonstrate the effectiveness and advantages of the LBFN model.

#### A. NUMERICAL EXAMPLE

The quadratic objective function is designed to be strictly convex and dynamic through its parameters:

$$\begin{aligned} h(\mathbf{v}(t), t) &= \\ &\frac{1}{2} \left[ \begin{matrix} v_1(t) & v_2(t) \end{matrix} \right] \underbrace{\begin{bmatrix} 1 + 0.5 \sin(3t) & 0 \\ 0 & 2 + \cos(6t) \end{bmatrix}}_{\mathcal{P}(t)} \begin{bmatrix} v_1(t) \\ v_2(t) \end{bmatrix} + \\ &\underbrace{\begin{bmatrix} \sin(3t) & \cos(3t) \end{bmatrix}}_{\mathbf{n}^T(t)} \begin{bmatrix} v_1(t) \\ v_2(t) \end{bmatrix} \end{aligned} \quad (19)$$

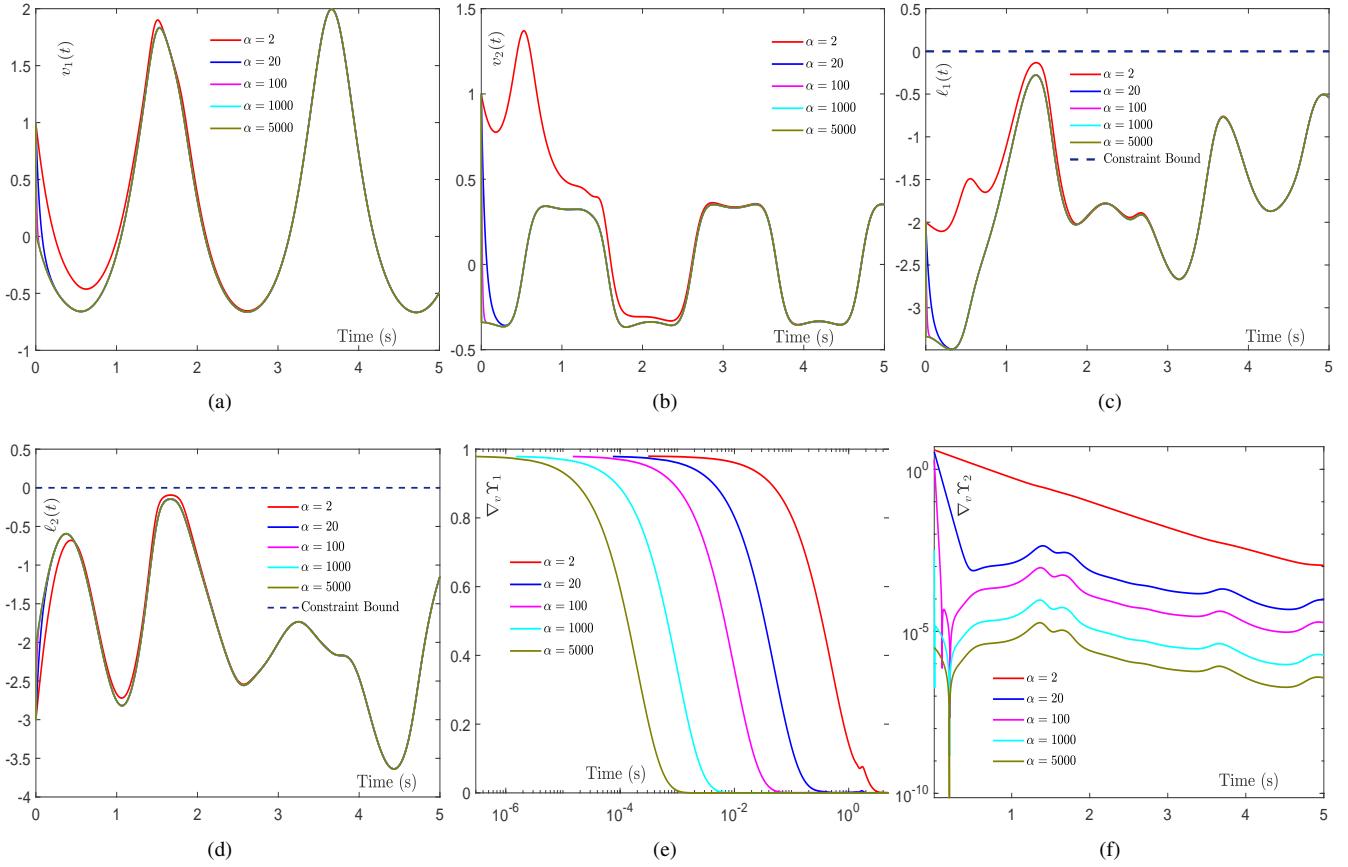
Notice that the Hessian Matrix  $\mathcal{P}(t)$  has a diagonal structure ensuring the convexity. To be specific,  $\mathcal{P}_{11}(t) = 1 + 0.5 \sin(3t)$  indicates the periodic variation between [0.5, 1.5], and  $\mathcal{P}_{22}(t) = 2 + \cos(6t)$  indicates the higher frequency variation between [1, 3]. Moreover, the eigenvalues of Hessian Matrix  $\mathcal{P}(t)$  are  $\lambda_1(t) \in [0.5, 1.5]$  and  $\lambda_2(t) \in [1, 3]$ . The problem remains strictly convex  $\forall t \in \mathbb{R}$  because:

$$\mathcal{P}(t) \succ 0$$

$$\det(\mathcal{P}(t)) = (1 + 0.5 \sin(3t))(2 + \cos(6t)) \geq 0.5 > 0$$

This ensures the strictly convex property of the objective function. Meanwhile, this also ensures a unique global minimum exists at each time instant.

Furthermore, the dynamic inequality constraints  $\mathcal{G}(t)\mathbf{v}(t) \leq$



**FIGURE 1.** Simulative outcomes of the presented LBFN model (9) to solve the optimal solution trajectory of the DQP-IC problem (21), with convergent parameter  $\alpha$  varies from 2 to 5000,  $\gamma_w = 10$  and  $\gamma_r = 5$ . (a) Convergence behaviors of  $v_1(t)$ . (b) Convergence behaviors of  $v_2(t)$ . (c) Constraint behaviors of  $g_1(t) = \sin(2t)v_1(t) + v_2(t) - (2 + \cos(2t))$ . (d) Constraint behaviors of  $g_2(t) = -\cos(2t)v_1(t) + (2t/10)v_2(t) - (2 - \sin(4t))$ . (e) The first entry of the absolute gradient error. (f) The second entry of the absolute gradient error.

$\mathbf{h}(t)$  are defined by:

$$\begin{bmatrix} \sin(2t) & 1 \\ -\cos(2t) & 2t/10 \end{bmatrix} \begin{bmatrix} v_1(t) \\ v_2(t) \end{bmatrix} \leq \begin{bmatrix} 2 + \cos(2t) \\ 2 - \sin(4t) \end{bmatrix} \quad (20)$$

In this case, this specific example is shown as

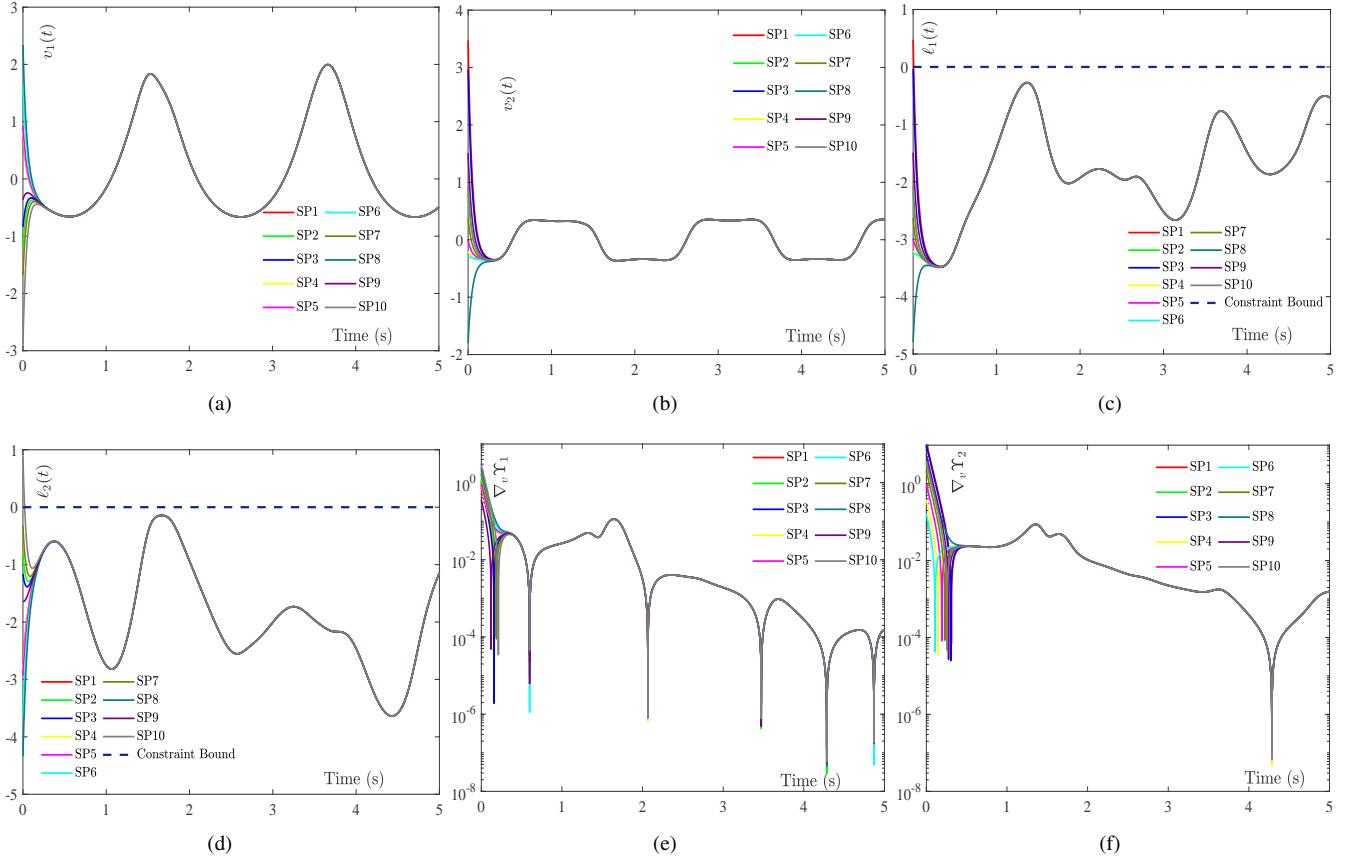
$$\begin{aligned} \text{minimize}_{\mathbf{v}(t) \in \mathbb{R}^n} \quad & \mathbf{h}(\mathbf{v}(t), t) = \frac{1}{2} \mathbf{v}^\top(t) \mathcal{P}(t) \mathbf{v}(t) + \mathbf{n}^\top(t) \mathbf{v}(t) \\ \text{subject to} \quad & \ell(\mathbf{v}(t), t) : \mathcal{G}(t) \mathbf{v}(t) \leq \mathbf{h}(t) \end{aligned} \quad (21)$$

Convergence characteristics are comprehensively evaluated under varying convergence parameter  $\alpha$ , as shown in Figure 1, which reveals critical dynamics governing the system behavior. At  $\alpha = 2$ , both variables  $v_1(t)$  and  $v_2(t)$  exhibits persistent oscillations and slow convergence around 3 seconds. Remarkably, increasing  $\alpha$  to the extreme case 5000 demonstrated rapid convergence in under  $10^{-3}$  seconds, confirming the theoretically predicted exponential convergence rate proportional to  $\alpha$ . Simultaneously, constraint satisfaction behaviors are critically dependent on  $\alpha$ , where  $\alpha = 5000$  enabled near-instantaneous boundary tracking within 0.1-0.2 s while lower  $\alpha$  values exhibited delays. Gradient errors in Figures 1(e) and (f) intuitively show the exponential decay patterns, in which a larger  $\alpha$  consumes less convergent time

and lower gradient errors, and it eventually reaches an error floor of  $10^{-7}$  at  $\alpha = 5000$ .

Initialization robustness is rigorously tested across ten distinct starting points spanning  $\mathbf{v}_i(0) \in [-1.5, 1.5]$ , as shown in Figure 2. All trajectories rapidly converged to a standard solution manifold within 0.5 s, regardless of initial states. The LBFN model maintained strict feasibility throughout the solution process, correcting worst-case initial constraint violations within 0.3 s while demonstrating remarkable consistency in gradient error reduction exceeding three orders of magnitude across all initializations. The observed exponential decay rate followed precisely the theoretically predicted  $e^{-2\alpha t}$  pattern, establishing the approach's robustness against initialization variance.

Comparative analysis against established solvers revealed the LBFN model's superior performance in dynamic programming problems, as shown in Figure 3. The comparative approaches include the active-set (AS) approach, the continuous gradient descent (CGD) approach in [35], the SQP approach in [20], the SeDuMi approach in [36], and the Newton flows (NF) approach [37]. To be specific, the AS and NF approaches show constraint violations during transition periods ( $t \in [1, 2]$  s and  $t \in [7, 8]$  s), the SeDuMi approach



**FIGURE 2.** Simulative outcomes of the presented LBFN model (9) to solve the optimal solution trajectory of the DQP-IC problem (21), under 10 different starting points (SP), with  $v_i(0) \in [-1.5, 1.5]$ , the convergent parameter  $\alpha$  set to 20,  $\gamma_w = 10$  and  $\gamma_r = 5$ . (a) Convergence behaviors of  $v_1(t)$ . (b) Convergence behaviors of  $v_2(t)$ . (c) Constraint behaviors of  $g_1(t) = \sin(2t)v_1(t) + v_2(t) - (2 + \cos(2t))$ . (d) Constraint behaviors of  $g_2(t) = -\cos(2t)v_1(t) + (2t/10)v_2(t) - (2 - \sin(4t))$ . (e) The first entry of the absolute gradient error. (f) The second entry of the absolute gradient error.

displays transient violations at constraint boundaries, and the CGD approach develops steady-state offsets from parameter drift; the LBFN model maintains continuous constraint satisfaction without violations. Gradient errors convergence metrics confirm distinct advantages, where the LBFN model maintains exponential decay to  $10^{-8}$  while other comparative approaches fail to stabilize  $10^{-4}$ . These results confirm the theoretical predictions of global convergence with tunable rates and persistent feasibility.

## B. APPLICATION ON DYNAMIC ROBOT NAVIGATION

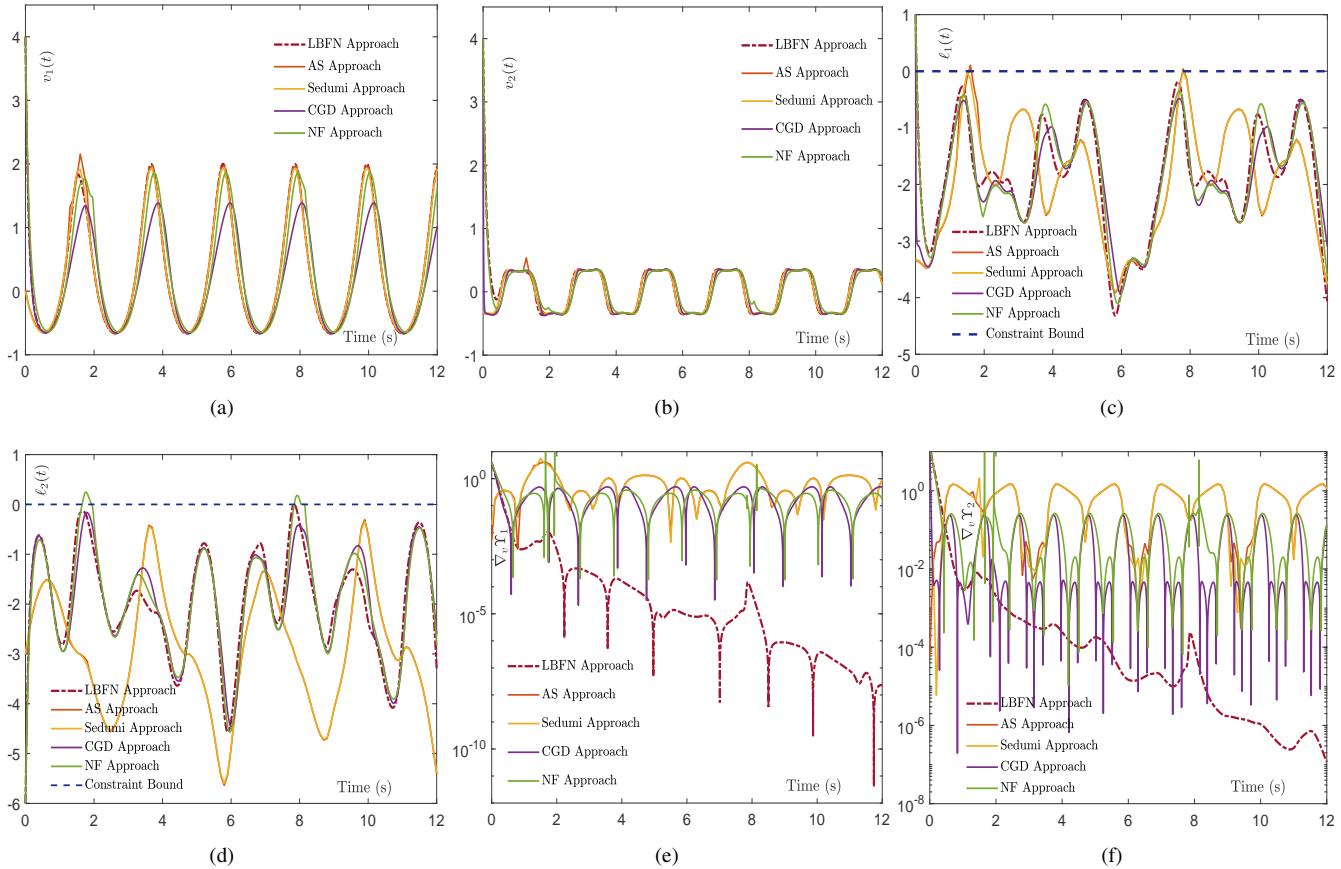
In this part, to further verify the effectiveness of the constructed LBFN model, a dynamic robot navigation task is designed, in which the robot is required to track the dynamic target in real time. This is a DQP-IC problem formulated as follows.

$$\begin{aligned} & \underset{\mathbf{g}(t) \in \mathbb{R}^2}{\text{minimize}} \quad \frac{1}{2} k_t \|\mathbf{g}(t) - \mathbf{g}_{\text{Tar}}(t)\|^2 \\ & \quad + \sum_i^7 \exp \left( -\frac{\|\mathbf{g}(t) - \mathbf{g}_{\text{Obs}_i}(t)\|^2}{k_i} \right), \end{aligned} \quad (22)$$

subject to  $\mathcal{A}\mathbf{g}(t) \leq \mathbf{d}$

where  $\mathcal{A} = [1 \ 0; 0 \ 1; -1 \ 0; 0 \ -1]$  and  $\mathbf{d} = [100; 100; 0; 0]$ , and this inequality denotes the physical bounds of the map of the robot navigation task. In this experiment, the constructed LBFN model  $\alpha$  convergent parameter is set to 1000, guaranteeing the swiftly convergent performance and meeting the real-time control for the dynamic robot navigation task (22). Moreover, the moving target's trajectory is set  $\mathbf{g}_{\text{Tar}}(t) = [50 + 40 \sin(0.25t), 50 + 25 \cos(0.25t)]^\top$ . The 7 obstacles  $\mathbf{g}_{\text{Obs}_i}(t)$  with  $i = 1, 2, \dots, 7$ , separately located at [20; 25] m, [60; 20] m, [70; 60] m, [50; 85] m, [30; 72] m, [38; 35] m and [90; 40] m. The parameters for the obstacles are set  $k_1 = 0.8$ ,  $k_2 = 0.6$ ,  $k_3 = 1.0$ ,  $k_4 = 1.2$ ,  $k_5 = 1.2$ ,  $k_6 = 0.8$  and  $k_7 = 0.9$ , and the parameter for the target  $k_t$  is set as 0.3. The whole task time duration is set as  $T = 30$  s. As mentioned above, the parameter settings are provided.

As depicted in Figure 4, the snapshots at the time  $t = 10$  s,  $t = 20$  s, and  $t = 30$  are offered to present the experimental results. The dynamic target is observed to move along an elliptical pattern trajectory, with seven stationary obstacles involved. Figures 4(a), (d), and (g) provide the intuitive trajectory tracking result, where the robot starts from the starting point and swiftly converges to the moving target, and then follows the moving target in real time, while avoiding the



**FIGURE 3.** Comparative simulation among the presented LBFN model (9), AS Approach, Sedumi Approach, CGD Approach and NF Approach for solving the optimal solution trajectory of the DQP-IC problem (21), with  $\alpha = 10$ ,  $\gamma_w = 10$  and  $\gamma_r = 5$ . (a) Convergence behaviors of  $v_1(t)$ . (b) Convergence behaviors of  $v_2(t)$ . (c) Constraint behaviors of  $g_1(t) = \sin(2t)v_1(t) + v_2(t) - (2 + \cos(2t))$ . (d) Constraint behaviors of  $g_2(t) = -\cos(2t)v_1(t) + (2t/10)v_2(t) - (2 - \sin(4t))$ . (e) The first entry of the absolute gradient error. (f) The second entry of the absolute gradient error.

obstacles located in the map. Further, Figures 4(b), (e), and (h) provide the potential force field on contour, and Figures 4(c), (f), and (i) provide the potential function landscape, for different time instants. This intuitively demonstrates that under the control of the constructed LBFN model, the robot can excellently accomplish the given dynamic robot navigation task in real time.

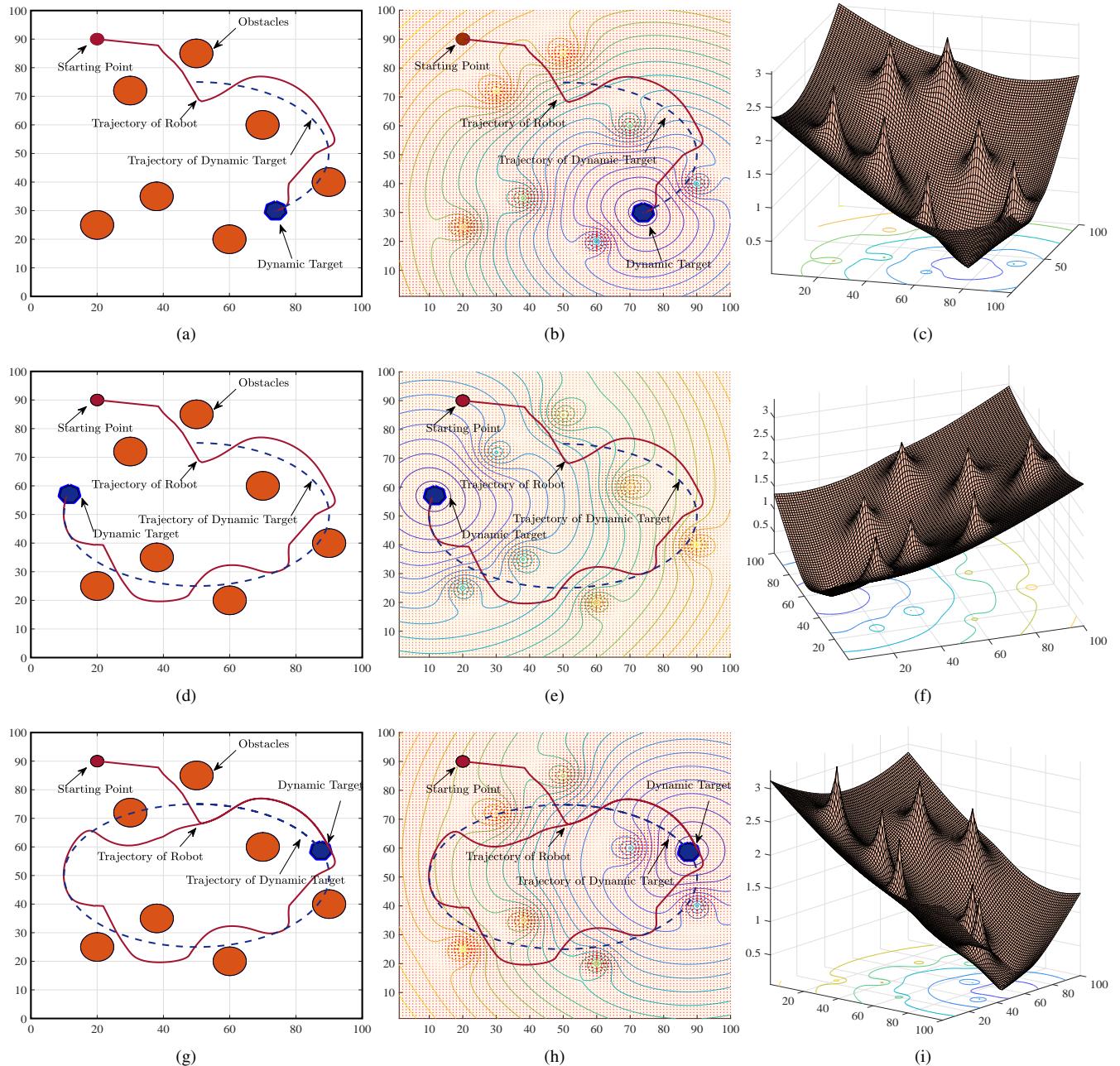
## V. CONCLUSIONS

This paper has established a logarithmic barrier function-based Newton's (LBFN) model for real-time solving of dynamic quadratic programming problems subject to inequality constraints. The proposed model fundamentally advances beyond traditional approaches by employing parametric adaptive log-barrier functions to directly handle inequality constraints, thereby avoiding the need for auxiliary variables typically required in KKT-based methods. The LBFN model demonstrates three key advantages that significantly enhance its practical applicability: (1) guaranteed global convergence from arbitrary initial points without feasibility restrictions, (2) continuous constraint satisfaction throughout the solution trajectory, and (3) exponentially decaying tracking error with

explicitly tunable convergence rates. These properties make the model particularly suitable for dynamic environments where both computational efficiency and constraint adherence are critical. Numerical experiments comprehensively validated the model's performance, showing superior convergence speed and solution accuracy compared to established solvers across various operational scenarios. The comparative analysis revealed that the LBFN model maintains consistent constraint satisfaction while achieving significantly lower tracking errors. The practical effectiveness of the approach was further demonstrated through a dynamic robot navigation application, where the model successfully enabled simultaneous trajectory tracking and obstacle avoidance in real-time operation.

## REFERENCES

- [1] D. Liu, S. Xue, B. Zhao, B. Luo and Q. Wei, "Adaptive Dynamic Programming for Control: A Survey and Recent Advances," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 1, pp. 142–160, Jan. 2021.
- [2] Y. E. Wang, "Theory of Broadband Transmission of Frequency Modulated Signals through High-Q Resonators with Continuous Time-Varying Matching Networks," *IEEE Access*, doi: 10.1109/ACCESS.2025.3595056.
- [3] D. Hanover, A. Loquercio, L. Bauersfeld, A. Romero, R. Penicka, Y. Song, G. Cioffi, E. Kaufmann and D. Scaramuzza, "Autonomous Drone Racing:



**FIGURE 4.** Experimental results of the dynamic robot navigation task (22) accomplished by the constructed LBFN model in real time, with the task duration being  $T = 30$  s, involving 7 obstacles. (a) Robot motion trajectory at  $t = 10$  s. (b) Potential force field on contour at  $t = 10$  s. (c) Potential function landscape at  $t = 10$  s. (d) Robot motion trajectory at  $t = 20$  s. (e) Potential force field on contour at  $t = 20$  s. (f) Potential function landscape at  $t = 20$  s. (g) Robot motion trajectory at  $t = 30$  s. (h) Potential force field on contour at  $t = 30$  s. (i) Potential function landscape at  $t = 30$  s.

- A Survey," *IEEE Transactions on Robotics*, vol. 40, pp. 3044–3067, May 2024.
- [4] J. Park, M. Kang, Y. Lee, J. Jung, H. -T. Choi and J. Choi, "Multiple Autonomous Surface Vehicles for Autonomous Cooperative Navigation Tasks in a Marine Environment: Development and Preliminary Field Tests," *IEEE Access*, vol. 11, pp. 36203–36217, 2023.
- [5] K. Harlow, H. Jang, T. D. Barfoot, A. Kim and C. Heckman, "A New Wave in Robotics: Survey on Recent MmWave Radar Applications in Robotics," *IEEE Transactions on Robotics*, vol. 40, pp. 4544–4560, Sept. 2024.
- [6] Y. Liufu et al., "Collaborative Physics-Informed Neural Dynamics Approach for Autonomous Vehicles with Nonconvex Safety-Critical Constraints", *IEEE Transactions on Intelligent Transportation Systems*, doi:

10.1109/TITS.2025.3572336.

- [7] J. Nur Fadila, N. Haliza Abdul Wahab, A. Alshammary, A. Aqarni, A. Al-Dhaqm and N. Aziz, "Comprehensive Review of Smart Urban Traffic Management in the Context of the Fourth Industrial Revolution," *IEEE Access*, vol. 12, pp. 196866–196886, 2024.
- [8] C. Creß, Z. Bing and A. C. Knoll, "Intelligent Transportation Systems Using Roadside Infrastructure: A Literature Survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 7, pp. 6309–6327, Jul. 2024.
- [9] M. N. Elsayed, O. De Silva, A. Jayasiri, G. K. I. Mann and R. Gosine, "Optimization-Based Maneuvering Target Tracking Using Multiple Model Horizon Scenario Tree With Model Interaction," *IEEE Transactions on*

- Aerospace and Electronic Systems*, doi: 10.1109/TAES.2025.3592177.
- [10] G. R. Prudvi Kumar and S. Dasarathan, "Metaheuristic-Tuned Droop Control for PV-Based DC Microgrid Optimization Under Dynamic Load Conditions," *IEEE Access*, doi: 10.1109/ACCESS.2025.3592751.
- [11] J. Fan, L. Jin, P. Li, J. Liu, Z.-G. Wu and W. Chen, "Coevolutionary Neural Dynamics Considering Multiple Strategies for Nonconvex Optimization," *Tsinghua Science and Technology*, doi: 10.26599/TST.2025.9010120.
- [12] Y. Liufu et al., "Modified Gradient Projection Neural Network for Multiset Constrained Optimization," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 9, pp. 9413–9423, Sept. 2023.
- [13] J. M. Altschuler, J. Bok and K. Talwar, "On the Privacy of Noisy Stochastic Gradient Descent for Convex Optimization," *SIAM Journal on Computing*, vol. 53, no. 4, 969–1001, 2024.
- [14] H. Wen and X. He, "Arbitrary-Time Stable Gradient Flows Using Neurodynamic System for Continuous-Time Optimization," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 70, no. 12, pp. 4559–4563, Dec. 2023.
- [15] R. Chen and X. Li, "Implicit Runge-Kutta Methods for Accelerated Unconstrained Convex Optimization," *IEEE Access*, vol. 8, pp. 28624–28634, 2020.
- [16] Jing Ren, K. A. McIsaac and R. V. Patel, "Modified Newton's Method Applied to Potential Field-Based Navigation for Mobile Robots," *IEEE Transactions on Robotics*, vol. 22, no. 2, pp. 384–391, Apr. 2006.
- [17] D. Niu, Y. Hong and E. Song, "A Dual Inexact Nonsmooth Newton Method for Distributed Optimization," *IEEE Transactions on Signal Processing*, vol. 73, pp. 188–203, 2025.
- [18] H. Kim and H. Park, "Nonnegative Matrix Factorization Based on Alternating Nonnegativity Constrained Least Squares and Active Set Method," *SIAM Journal on Matrix Analysis And Applications*, vol. 30, no. 2, pp. 713–730, 2008.
- [19] G. Cimini and A. Bemporad, "Exact Complexity Certification of Active-Set Methods for Quadratic Programming," *IEEE Transactions on Automatic Control*, vol. 62, no. 12, pp. 6094–6109, Dec. 2017.
- [20] F. E. Curtis and M. L. Overton, "A Sequential Quadratic Programming Algorithm for Nonconvex, Nonsmooth Constrained Optimization," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 474–500, 2012.
- [21] A. S. Berahas, M. Xie and B. Zhou, "A Sequential Quadratic Programming Method With High-Probability Complexity Bounds for Nonlinear Equality-Constrained Stochastic Optimization," *SIAM Journal on Optimization*, no. 35, no. 1, pp. 240–269, 2025.
- [22] X. Shi, X. Xu, J. Cao, and X. Yu, "Finite-Time Convergent Primal-Dual Gradient Dynamics With Applications to Distributed Optimization," *IEEE Transactions on Cybernetics*, vol. 53, no. 5, pp. 3240–3252, May 2023.
- [23] J. I. Poveda and M. Krstic, "Fixed-time Gradient-Based Extremum Seeking," in Proceedings of the American Control Conference, 2020, pp. 2838–2843.
- [24] I. Fatkhullin and B. Polyak, "Optimizing Static Linear Feedback: Gradient Method," *SIAM Journal on Control and Optimization*, vol. 59, no. 5, pp. 3887–3911, 2021.
- [25] M. Mestari, M. Benzirar, N. Saber and M. Khouil, "Solving Nonlinear Equality Constrained Multiobjective Optimization Problems Using Neural Networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 10, pp. 2500–2520, Oct. 2015.
- [26] T. Breiten and L. Pfeiffer, "On the Turnpike Property and The Receding-Horizon Method for Linear-Quadratic Optimal Control Problems," *SIAM Journal on Control and Optimization*, vol. 58, no. 2, pp. 1077–1102, 2020.
- [27] Y. Hu, L. Wei, X. Guo, Z. Li, Y. Wei, and Q. Fang, "Priority-Based Reward Mechanism for Dual-Robot Path Planning in Dynamic Environments," *IEEE Access*, vol. 13, pp. 92519–92530, 2025.
- [28] M. Liu, Y. Li, Y. Chen, Y. Qi, and L. Jin, "A Distributed Competitive and Collaborative Coordination for Multirobot Systems," *IEEE Transactions on Mobile Computing*, vol. 23, no. 12, pp. 11436–11448, Dec. 2024.
- [29] R. J. Roesthuis and S. Misra, "Steering of Multisegment Continuum Manipulators Using Rigid-Link Modeling and FBG-Based Shape Sensing," *IEEE Transactions on Robotics*, vol. 32, no. 2, pp. 372–382, Apr. 2016.
- [30] J. Lin, D.-H. Zhai, Y. Xiong and Y. Xia, "Safety Control for UR-Type Robotic Manipulators via High-Order Control Barrier Functions and Analytical Inverse Kinematics," *IEEE Transactions on Industrial Electronics*, vol. 71, no. 6, pp. 6150–6160, Jun. 2024.
- [31] S. Jing, Y. Zhao, X. Zhao, F. Hui and A. J. Khattak, "An Efficient High-Risk Lane-Changing Scenario Edge Cases Generation Method for Autonomous Vehicle Safety Testing," *IEEE Transactions on Intelligent Vehicles*, vol. 10, no. 1, pp. 459–471, Jan. 2025.
- [32] L. Jin, L. Wei, and S. Li, "Gradient-Based Differential Neural-Solution to Time-Dependent Nonlinear Optimization," *IEEE Transactions on Automatic Control*, vol. 68, no. 1, pp. 620–627, Jan. 2023.
- [33] Y. Liufu et al., "Adaptive Noise-Learning Differential Neural Solution for Time-Dependent Equality-Constrained Quadratic Optimization," *IEEE Transactions on Neural Networks and Learning Systems*, doi: 10.1109/TNNLS.2025.3561415.
- [34] S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [35] J. Sirignano and K. Spiliopoulos, "Stochastic Gradient Descent in Continuous Time, *SIAM Journal on Financial Mathematics*, vol. 8, no. 1, pp. 933–961, 2017.
- [36] Y. Labit, D. Peaucelle and D. Henrion, "SeDuMi Interface 1.02: A tool for Solving LMI Problems with SeDuMi," in Proceedings. IEEE international symposium on computer aided control system design. IEEE, 2002: 272–277.
- [37] D. Niu, Y. Hong and E. Song, "A Dual Inexact Nonsmooth Newton Method for Distributed Optimization," *IEEE Transactions on Signal Processing*, vol. 73, pp. 188–203, 2025.

**JIUFAN WANG** is currently a PhD student in Operations Research at Ohio State University, Columbus, USA. He received the B.S degree from the Civil Aviation University of China, China and the M.S. in Computer Science with a focus on Computational Operations Research from William & Mary, USA. His research focuses on optimization algorithms, particularly Ant Colony Optimization for vehicle path planning and Hypercube Queuing Model for healthcare.



**YUQUN ZHOU** is an experienced Data Scientist with a Ph.D. in Industrial Engineering (Operations Research) from the University of Wisconsin-Madison. With over three years at Bayer Crop Science, he has delivered significant business impact through advanced optimization and machine learning solutions, including 5M in cost savings and a 40% improvement in scheduling efficiency. As a certified PMP and AWS specialist, Mark combines deep expertise in operations research, large-



scale optimization, and cloud-based deployment with proven ability to lead complex, cross-functional projects. His work has resulted in multiple publications and two international patents in resource optimization and scheduling systems.

\*\*\*