OPEN ACCESS

Energy Efficiency ARTICLE

The Innovation Energy

Energy-efficient robotic inspection in under-panel spaces of photovoltaic plants

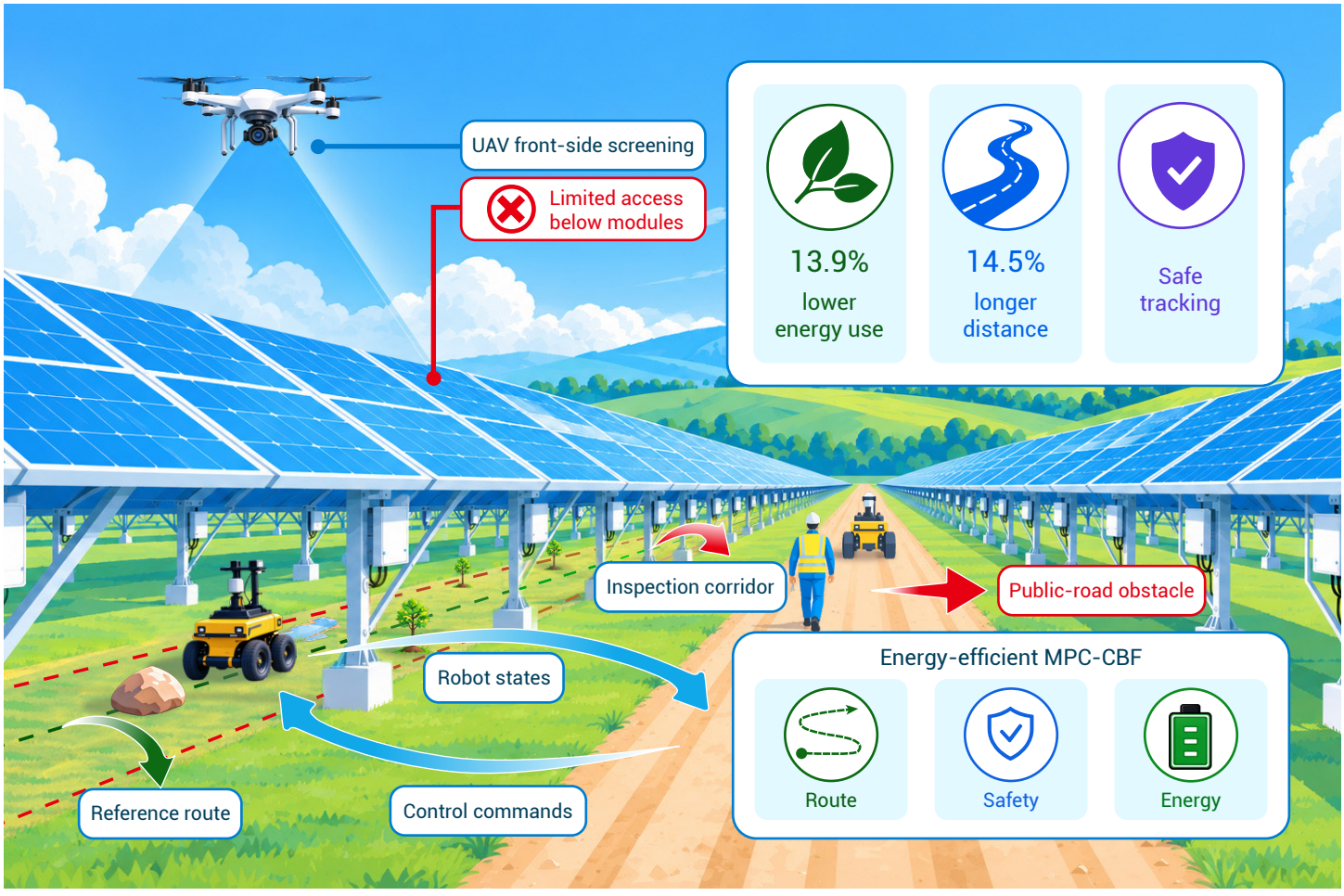
Liang Gao,[1] Shuting Le,[2] Xinhui Zhao,[1] Zongyang Lv,[3] Xun Shen,[4] Benlian Xu,[2] and Yuhu Wu[1,*]

*Correspondence: wuyuhu@dlut.edu.cn

Received: May 29, 2026; Accepted: July 8, 2026; Published Online: October 19, 2026; https://doi.org/10.59717/j.xinn-energy.2026.100190

© 2026 The Author(s). This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

GRAPHICAL ABSTRACT



PUBLIC SUMMARY

- An energy-efficient model predictive control–control barrier function (MPC-CBF) controller is developed for safe under-panel photovoltaic (PV) inspection.
- The fixed-task test shows a 13.9% energy reduction over standard MPC-CBF.
- The limited-energy test shows a 14.5% increase in completed inspection distance.

ARTICLE Energy Efficiency

OPEN ACCESS

www.the-innovation.org/the-innovation-energy

Energy-efficient robotic inspection in under-panel spaces of photovoltaic plants

Liang Gao,[1] Shuting Le,[2] Xinhui Zhao,[1] Zongyang Lv,[3] Xun Shen,[4] Benlian Xu,[2] and Yuhu Wu[1,*]

[1]School of Control Science and Engineering, Dalian University of Technology, Dalian 116024, China

[2]School of Electronic and Information Engineering, Suzhou University of Science and Technology, Suzhou 215000, China

[3]Department of Mechanical Engineering, University of Victoria, Victoria V8W 2Y2, Canada

[4]Department of Electrical Engineering and Computer Science, Tokyo University of Agriculture and Technology, Tokyo 183-8538, Japan

*Correspondence: wuyuhu@dlut.edu.cn

Received: May 29, 2026; Accepted: July 8, 2026; Published Online: October 19, 2026; https://doi.org/10.59717/j.xinn-energy.2026.100190

© 2026 The Author(s). This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

Citation: Gao L., Le S., Zhao X., et al. (2026). Energy-efficient robotic inspection in under-panel spaces of photovoltaic plants. *The Innovation Energy* **3**:100190.

Large-scale ground-mounted photovoltaic (PV) plants are expanding toward ultra-large deployment, making efficient operation and maintenance (O&M) critical for power-generation revenue, equipment reliability, and maintenance-cost reduction. Although unmanned aerial vehicle (UAV) inspection improves large-area inspection efficiency, it remains limited by endurance, under-panel accessibility, and adaptability to complex environments. A low-profile ground robot can access spaces beneath PV modules for close-range inspection, making it an important tool for intelligent PV plant O&M. This paper proposes an energy-efficient automated inspection method for a low-profile ground robot in PV module backsheet and under-panel scenarios by integrating model predictive control (MPC) and control barrier functions (CBFs). CBFs are used to describe corridor-boundary and obstacle-avoidance requirements, while robot energy consumption and the battery-usage proxy are incorporated into the MPC framework to improve inspection energy efficiency. Simulation results show that, compared with standard MPC-CBF, reduces energy consumption by 13.9% in the fixed-task test and increases the inspection distance by 14.5% in the limited-energy test, validating its potential for energy-saving inspection in large PV plants.

INTRODUCTION

With the global transition toward low-carbon and renewable energy systems, photovoltaic (PV) generation has become an important component of future energy infrastructure.[1] Recent studies indicate that PV deployment is not only increasing in scale but also extending to more diverse and space-constrained scenarios, including agricultural greenhouses,[2] mining areas,[3] and high-density ground-mounted PV plants.[4] This trend imposes higher requirements on the operation and maintenance (O&M) of PV plants. In particular, large-scale utility PV plants usually cover wide outdoor areas, involve a large number of devices, and operate under complex environmental conditions.[1] During long-term operation, PV modules and their auxiliary components may suffer from defects such as hot spots, cracks, loose connectors, junction-box failures, and loosened support structures, as illustrated in Figure 1A.[5,6] If these defects are not detected and handled in a timely manner, they may reduce power-generation efficiency, shorten equipment lifetime, and introduce potential electrical safety risks.[1,5–10] However, manual inspection of large-scale PV plants is typically time-consuming, labor-intensive, and prone to missed detections. Therefore, efficient automated inspection is important for supporting the reliable operation of PV plants.[1,6,8]

In recent years, unmanned aerial vehicle (UAV)-based inspection has become a major technical approach for automated PV plant inspection. UAVs can rapidly cover large PV arrays and, when combined with infrared imaging, electroluminescence imaging, and image recognition techniques, can detect module hot spots, shading, cracks, and other surface anomalies.[9,11–15] Existing studies indicate that such methods are suitable for large-scale screening of surface anomalies and can improve the efficiency of fault detection in PV plants. However, UAV inspection mainly acquires information from the front side of PV modules or the array surface. It is difficult for UAVs to enter low-clearance spaces beneath the modules or to perform stable close-range inspection of module backsheets, cables, connectors, supports, and under-panel passage environments. In addition, UAV inspection is constrained by endurance, weather conditions, close-range flight safety, and image degradation caused by fast flight or motion blur. Its performance may degrade under strong wind, rain, snow, dust, or visually obstructed conditions.[9,13,16,17] Therefore, UAV-based inspection alone is insufficient to cover all O&M targets in PV plants, particularly for fine-grained inspection of module backsheets and under-panel equipment.

A ground inspection robot provides an important complement for under-panel inspection in PV plants. Compared with UAVs, such a robot can travel at low speed and close range beneath PV modules and can carry sensors such as visible-light cameras, infrared cameras, and light detection and ranging (LiDAR) sensors to inspect module backsheets, connectors, cables, supports, and under-panel passages.[17] Existing studies have mainly focused on inspection path planning, robot-system design, and automated inspection frameworks. Representative examples include improved rapidly-exploring random tree star (RRT*)-based path planning for solar power plant inspection,[18] a PV-powered inspection robot system using maximum power point tracking (MPPT)-based power management,[19] a layered robotic platform for offshore PV inspection,[20] and broader automated PV plant inspection schemes based on unmanned vehicles.[21,22] These studies have advanced automated PV inspection. Nevertheless, most of them still emphasize module surface inspection, global task planning, or platform design, while the inspection problem for a ground robot operating in low-clearance spaces beneath PV modules has received comparatively limited attention.

Under-panel PV inspection has several distinctive characteristics. First, the robot operates in a low and narrow space, where PV supports, cables, junction boxes, and other fixed facilities are located on both sides. Even small deviations may degrade sensor observations or increase collision risk. Second, as illustrated in Figure 1B, temporary obstacles, such as maintenance personnel, operation vehicles, fallen objects, or local vegetation, may appear during inspection, requiring the robot to adjust its trajectory in real time according to site conditions. For PV plants, inspection routes are long and frequently repeated, whereas the battery capacity of a ground robot is limited. Frequent emergency stops, sharp turns, and unnecessary acceleration or deceleration may increase energy consumption, reduce the coverage range of a single inspection mission, and accelerate battery degradation. Therefore, ground robotic inspection for module backsheets and under-panel equipment requires not only tracking of a predefined inspection route, but also safe navigation and energy-efficient operation.

To address these requirements, model predictive control (MPC)[23–25] and control barrier functions (CBFs)[26–28] provide feasible technical support for under-panel robotic inspection. MPC can dynamically adjust the robot velocity and steering command by considering the reference route, obstacle distribution, and energy-related cost over a prediction horizon. CBFs can impose distance constraints between the robot and obstacles or passage boundaries, thereby improving navigation safety in complex environments.[23,24,26–28] In the context of under-panel PV inspection, these methods are relevant not only for improving the tracking accuracy of predefined inspection routes, but also for enabling the robot to balance safe passage with energy-efficient operation. This capability can improve the reliability and energy efficiency of automated under-panel inspection.

To address the above issues, this paper proposes an intelligent inspection scheme for photovoltaic module backsheets and under-panel corridors using a low-profile wheeled ground robot as the inspection platform. The main contributions are summarized as follows.

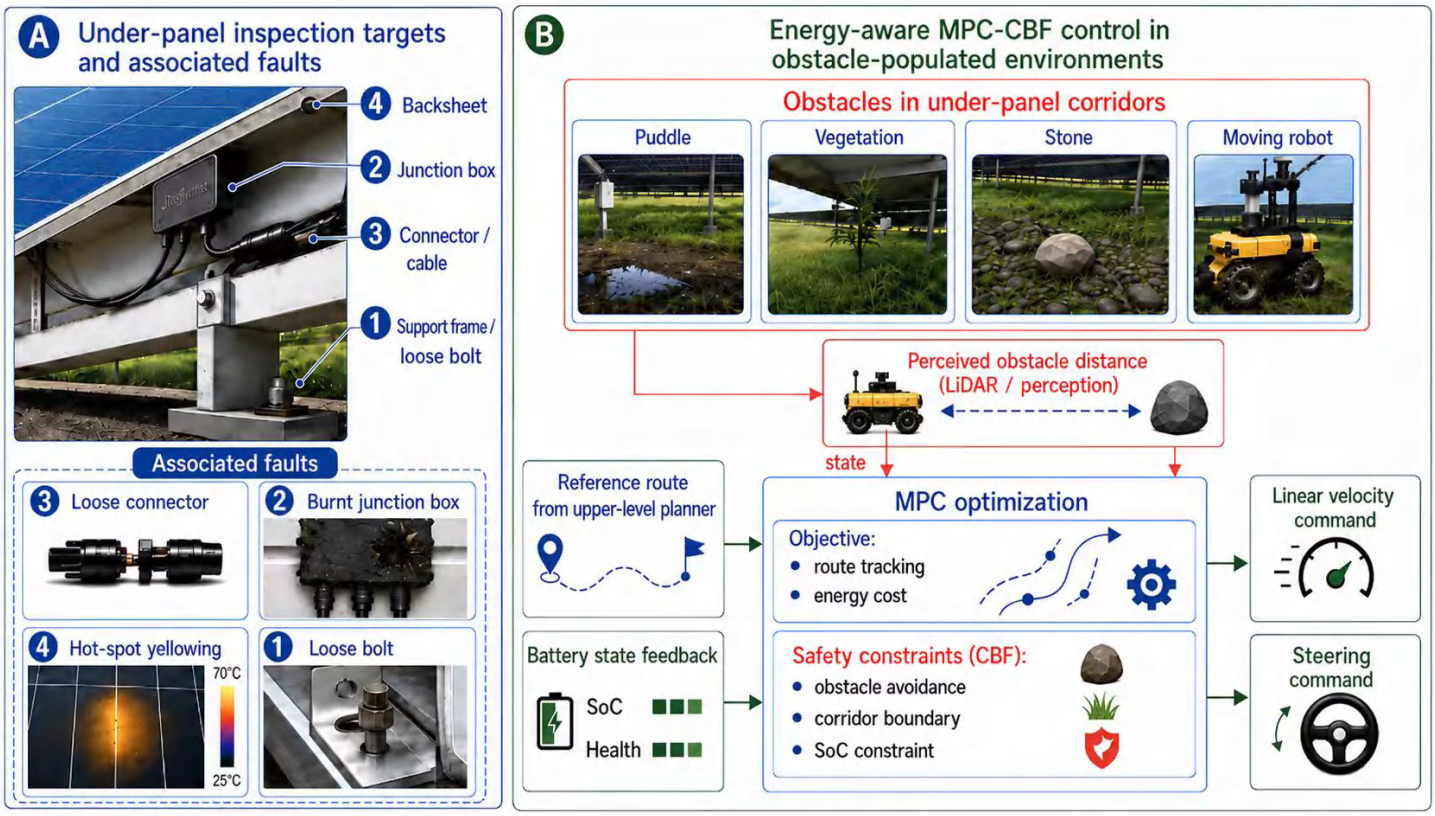


Figure 1. Under-panel inspection targets, associated faults, and energy-aware predictive safety control framework.

First, the under-panel inspection task in photovoltaic plants is formulated as a safety-critical and energy-efficient execution problem for low-profile ground robots. An energy-efficient MPC-CBF control framework is developed, in which path tracking, safety enforcement, energy consumption, and terminal energy reserve are jointly incorporated into a unified model predictive control problem.

Second, the robot kinematics and the battery state-of-charge (SoC) dynamics are embedded into a unified prediction model, allowing motion execution and energy evolution to be optimized in a coordinated manner within the same predictive framework.

Third, simulations are conducted in a representative local photovoltaic array scenario, and the proposed method is compared with a standard MPC-CBF baseline. The results show that the proposed approach reduces inspection energy consumption while maintaining basic trajectory-tracking performance and safety-critical obstacle avoidance. Moreover, under the same energy budget, it enables a longer completed inspection distance.

MODELING OF UNDER-PANEL PV INSPECTION

This section establishes the kinematic model, energy consumption model, and safety constraints for a single low-profile under-panel inspection robot. It is assumed that the reference inspection trajectory is generated in advance by the upper-level planning module, the corridor boundaries and fixed infrastructure information are provided by the power plant map, and local static or moving obstacles are perceived online using onboard LiDAR or cameras.

Under-panel inspection task description

This paper focuses on under-panel inspection of PV modules by a single low-profile ground robot. The inspection robot is required to follow a predefined route and perform close-range inspection of module backsheets, electrical devices, and the under-panel passage environment. According to the characteristics of under-panel inspection scenarios, this task is summarized into the following three technical issues.

1. Trajectory tracking for high-quality inspection data acquisition: To ensure image clarity and consistency of the viewing angle, the robot should move smoothly along the predefined inspection route.

2. Static and dynamic obstacle avoidance in complex under-panel environments: During inspection, the robot needs to travel safely through narrow and structurally complex under-panel passages, avoid static obstacles such as supports, trees, and electrical devices, and perceive and avoid moving obstacles such as maintenance personnel, service vehicles, or moving equipment in real time.

3. Energy constraints and battery-usage moderation for low-cost O&M: Large-scale PV power plants require long-distance inspection, and the endurance of the robot directly affects the coverage of each inspection task. To reduce O&M costs, the robot should reduce frequent acceleration and deceleration, sharp turns, and unnecessary detours, while jointly considering energy constraints and battery-usage moderation in execution control.

In summary, this paper formulates the under-panel PV inspection task as a safe and energy-efficient inspection execution problem that jointly considers reference trajectory tracking, static and dynamic obstacle avoidance, energy constraints, and battery-usage moderation. The following subsections further present the robot kinematic model, energy model, and obstacle-avoidance safety constraints.

Inspection robot kinematics and tracking model

At discrete time $k \in \mathbb{N}$, the state of the inspection robot is defined as

$$\mathbf{x}(k) := \begin{bmatrix} x(k) & y(k) & \theta(k) & v(k) & s(k) \end{bmatrix}^\top \in \mathbb{R}^5, \tag{1}$$

where $(x(k), y(k)) \in \mathbb{R}^2$ denotes the planar position, $\theta(k) \in \mathbb{R}$ is the heading angle, $v(k) \in \mathbb{R}$ is the longitudinal speed, and $s(k) \in [0,1]$ is the battery state of charge (SoC).

The control input is defined as

$$\mathbf{u}(k) := \begin{bmatrix} a(k) & \omega(k) \end{bmatrix}^\top \in \mathbb{R}^2, \tag{2}$$

where $a(k) \in \mathbb{R}$ and $\omega(k) \in \mathbb{R}$ denote the commanded longitudinal acceleration and yaw rate, respectively.

For brevity, define the pose-velocity substate as

$$\mathbf{X}(k) := \begin{bmatrix} x(k) & y(k) & \theta(k) & v(k) \end{bmatrix}^\top \in \mathbb{R}^4. \tag{3}$$

The robot kinematics are described by

$$\mathbf{X}(k+1)=\mathbf{X}(k)+\Delta t\big(A(k)+B\mathbf{u}(k)\big), \tag{4}$$

where $\Delta t>0$ is the control sampling interval,

$$A(k)=\begin{bmatrix} v(k)\cos\theta(k) \\ v(k)\sin\theta(k) \\ 0 \\ 0 \end{bmatrix}, \qquad B=\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{5}$$

For the assigned inspection corridor, the reference trajectory is denoted by

$$\mathbf{X}^{\mathrm{ref}}(k):=\begin{bmatrix} x^{\mathrm{ref}}(k) & y^{\mathrm{ref}}(k) & \theta^{\mathrm{ref}}(k) & v^{\mathrm{ref}}(k) \end{bmatrix}^{\top}. \tag{6}$$

Accordingly, the tracking error is defined as

$$\mathbf{e}(k):=\mathbf{X}(k)-\mathbf{X}^{\mathrm{ref}}(k). \tag{7}$$

In addition, the admissible input set is defined as

$$\mathcal{U}:=\left\{\mathbf{u}\in\mathbb{R}^2:\begin{matrix} a_{\min}\le a\le a_{\max}, \\ \omega_{\min}\le\omega\le\omega_{\max} \end{matrix}\right\}, \tag{8}$$

and the constrained-state set is defined as

$$\mathcal{X}:=\left\{\mathbf{x}\in\mathbb{R}^5:\begin{matrix} v_{\min}\le v\le v_{\max}, \\ s\ge s_{\min} \end{matrix}\right\}. \tag{9}$$

Here, $\mathcal{U}$ characterizes the actuator limits on the longitudinal acceleration and yaw rate, while $\mathcal{X}$ specifies the admissible ranges of the speed and battery SoC.

Energy consumption and battery state model

To minimize energy consumption during the inspection process, this paper is inspired by energy-efficient modeling and penalty-design ideas,[28,29] and further tailors them to the under-panel inspection robot scenario by introducing a simplified robot power model, an SoC update equation, and a battery-usage proxy. The instantaneous power is modeled as

$$P(v,a,\omega)=P_0+\alpha_v v^2+\alpha_a a^2+\alpha_\omega\omega^2, \tag{10}$$

where P_0 represents the baseline power required by onboard perception, computation, and communication, and the remaining terms capture motion-related power induced by speed, acceleration, and steering activity.

Equation (10) is used as a control-oriented approximation rather than a high-fidelity electrochemical battery model. Here, energy consumption refers to the physical electrical energy accumulated from this power model over time.

The SoC update equation,[30] the SoC update equation is written as

$$s(k+1)=s(k)-\frac{P(v(k),a(k),\omega(k))\Delta t}{E^{\max}}, \tag{11}$$

subject to the hard constraint $s(k)\ge s_{\min}$ for all k, where $s(k)\in[0,1]$ denotes the normalized battery SoC, $s_{\min}\in(0,1)$ is the minimum allowable SoC reserve for safe operation, and $E^{\max}$ denotes the usable battery energy capacity of the robot.

Additionally, the battery-usage proxy is defined as

$$\phi(k)=\beta_P\left(\frac{P(k)}{P_{\mathrm{rated}}}\right)^2+\beta_s\left[s_{\mathrm{hlth}}-s(k)\right]_+^2+\beta_a a^2(k), \tag{12}$$

where $P(k):=P(v(k),a(k),\omega(k))$, P_{rated} is the rated power, $a(k)$ is the longitudinal acceleration, $s_{\mathrm{hlth}}\in(0,1)$ is the preferred SoC threshold for low-SoC avoidance and is typically chosen above the hard lower bound $s_{\min}$, $[x]_+=\max(x,0)$, and $\beta_P,\beta_s,\beta_a\ge 0$ are weighting coefficients associated with the power, SoC-depletion, and acceleration terms, respectively. This proxy is introduced as an operational penalty rather than a separate battery-health state variable.

In (12), the first term penalizes high power demand, the second penalizes deep discharge below the preferred health zone, and the third suppresses excessively large acceleration magnitudes. Thus, $\phi(k)$ is referred to as the battery-usage proxy: it is a dimensionless quantity that combines high-power demand, low-SoC operation, and aggressive acceleration. When included in the MPC objective, this proxy acts as a soft penalty term. Note that, for simplicity, the penalty term associated with angular velocity is not included separately in (12).

Safety constraints in under-panel corridors

In under-panel robotic inspection scenarios, the robot may encounter heterogeneous obstacles, such as on-site workers, service vehicles, and vegetation intruding into the inspection corridor. To ensure safe operation, obstacle-avoidance constraints are introduced so that sufficient clearance from surrounding obstacles is maintained.

Let r denote the safety radius of the robot, and let $p(n|k)=[x(n|k),y(n|k)]^{\top}$ denote its predicted planar position at prediction step n and time k, for $n=0,\dots,N_p$. Let $\mathcal{O}(k)$ denote the set of obstacles detected around the robot at time k. For each obstacle $o\in\mathcal{O}(k)$, let $\hat{p}_o(n|k)$ denote its predicted center position at prediction step n, for $n=0,\dots,N_p$. Static obstacles are included as a special case by assuming that the obstacle center remains constant over the prediction horizon, i.e., $\hat{p}_o(n|k)=\hat{p}_o(0|k)$ for all n. The obstacle-avoidance safety function is defined as

$$h_o^{\mathrm{obs}}(n|k)=\|p(n|k)-\hat{p}_o(n|k)\|^2-\left(r+r_o+\delta_o\right)^2, \tag{13}$$

where r_o is the radius of obstacle o, and $\delta_o>0$ is an additional safety margin. The safety condition is $h_o^{\mathrm{obs}}(n|k)\ge 0$, which provides a unified treatment of both static and dynamic obstacles through $\hat{p}_o(n|k)$.

Additionally, the active inspection corridor is represented by the local convex polytope

$$\mathcal{P}=\{p\in\mathbb{R}^2\mid Gp\le b\}, \tag{14}$$

where $p\in\mathbb{R}^2$ is a generic planar position vector, the m-th row of G is denoted by $g_m^{\top}$, and the corresponding scalar bound is b_m, with $m=1,\dots,M$.

Equation (14) represents the active local corridor polytope, not one global convex region for the full multi-turn route. At each MPC update, the segment associated with the current reference position provides G and b; these quantities are updated online when the robot moves to the next segment.

The safety function associated with the m-th corridor boundary is written as

$$h_m^{\mathrm{bd}}(n|k)=b_m-g_m^{\top}p(n|k)-\delta_m^{\mathrm{bd}}, \tag{15}$$

where $\delta_m^{\mathrm{bd}}>0$ is the boundary buffer distance. The safety condition $h_m^{\mathrm{bd}}(n|k)\ge 0$ ensures that the robot remains inside the corridor while maintaining a sufficient clearance from the boundary.

In addition, the SoC safety function is defined as

$$h^{\mathrm{soc}}(n|k)=s(n|k)-s_{\min}, \tag{16}$$

where $s(n|k)$ is the predicted SoC at prediction step n at time k obtained from (11). The corresponding safety condition is $h^{\mathrm{soc}}(n|k)\ge 0$, which ensures $s(n|k)\ge s_{\min}$.

The above models provide the motion, energy, and environmental information required for the control design of the wheeled inspection robot. The next section develops a controller that generates admissible inputs for trajectory tracking, obstacle-avoidance safety, and energy-efficient operation.

SAFETY CONTROLLER DESIGN FOR PV POWER PLANT INSPECTION

Following the energy-aware MPC-CBF control framework shown in Figure 1B, this section designs an MPC-CBF safety controller for the PV inspection robot based on the preplanned reference trajectory $\mathbf{X}^{\mathrm{ref}}$. The controller generates low-level control inputs by solving a local optimization problem, thereby achieving trajectory tracking while satisfying safety and energy constraints.

Energy-efficient MPC cost formulation

At time k, suppose that the upper-level planner provides a feasible reference trajectory over the prediction horizon N_p. The corresponding reference state and nominal input sequences are denoted by

$$\{\mathbf{X}^{\mathrm{ref}}(n|k)\}_{n=0}^{N_p}, \quad \{\mathbf{u}^{\mathrm{ref}}(n|k)\}_{n=0}^{N_p-1}.$$

Here, $\mathbf{X}^{\mathrm{ref}}(n|k)$ denotes the reference pose-and-velocity state at prediction step n, and $\mathbf{u}^{\mathrm{ref}}(n|k)$ denotes the corresponding reference input.

From (4) and (11), the finite-horizon prediction model of the robot is given by

$$\mathbf{x}(n+1|k) = f_d(\mathbf{x}(n|k), \mathbf{u}(n|k)), \ \ n = 0, \ldots, N_p - 1, \tag{17}$$

with the initial condition $\mathbf{x}(0|k) = \mathbf{x}(k)$. Here, the mapping $f_d(\cdot,\cdot)$ is determined jointly by the kinematic state update and the battery SoC update, and is given by

$$f_d(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} x + \Delta t v \cos\theta \\ y + \Delta t v \sin\theta \\ \theta + \Delta t \omega \\ v + \Delta t a \\ s - \dfrac{\Delta t}{E^{\max}} P(v, a, \omega) \end{bmatrix}, \tag{18}$$

where $\mathbf{x} = [x\ y\ \theta\ v\ s]^\top$, $\mathbf{u} = [a\ \omega]^\top$, and $P(v,a,\omega)$ is the instantaneous power function defined in (10). Eq. (17) unifies robot motion and energy evolution into a single predictive framework, which provides the state evolution basis for the subsequent execution-layer optimization.

Therefore, the predicted tracking error over the prediction horizon is defined as

$$\mathbf{e}(n|k) = \mathbf{X}(n|k) - \mathbf{X}^{\mathrm{ref}}(n|k), \tag{19}$$

where $\mathbf{X}(n|k)$ denotes the pose-velocity substate extracted from $\mathbf{x}(n|k)$.

To penalize rapid input variations and to improve motion smoothness in narrow under-panel inspection passages, the input increment is introduced as follows:

$$\Delta\mathbf{u}(n|k) = \begin{cases} \mathbf{u}(0|k) - \mathbf{u}(k-1), & n = 0, \\ \mathbf{u}(n|k) - \mathbf{u}(n-1|k), & n = 1, \ldots, N_p - 1. \end{cases} \tag{20}$$

By jointly considering tracking performance, control smoothness, energy consumption, and battery usage stress, the stage cost at prediction step n is defined as

$$\begin{aligned} L(n|k) = {} & \mathbf{e}^\top(n|k) Q \mathbf{e}(n|k) + \tilde{\mathbf{u}}^\top(n|k) R \tilde{\mathbf{u}}(n|k) \\ & + \Delta\mathbf{u}^\top(n|k) S \Delta\mathbf{u}(n|k) + \rho_P \bar{P}(n|k) \Delta t \\ & + \rho_\phi \bar{\phi}(n|k) + \rho_\xi 1^\top \xi(n|k), \end{aligned} \tag{21}$$

where $\tilde{\mathbf{u}}(n|k) = \mathbf{u}(n|k) - \mathbf{u}^{\mathrm{ref}}(n|k)$. The matrices $Q \succeq 0$, $R \succ 0$, and $S \succeq 0$ weight the tracking-error, control-deviation, and input-increment terms, respectively. The scalars $\rho_P > 0$, $\rho_\phi > 0$, and $\rho_\xi > 0$ are penalty coefficients. In (21), $\xi(n|k)$ denotes the aggregate nonnegative slack vector associated with the safety constraints, and 1 denotes the all-ones vector of compatible dimension. Therefore, $1^\top \xi(n|k)$ penalizes the sum of all slack variables and helps preserve numerical feasibility when the local constraints become temporarily restrictive.

To mitigate the difficulty of weight tuning caused by heterogeneous physical units, the power-related index and the battery usage index are normalized as

$$\bar{P}(n|k) = \frac{P(n|k)}{P_{\mathrm{rated}}}, \qquad \bar{\phi}(n|k) = \frac{\phi(n|k)}{\phi_{\mathrm{nom}}}, \tag{22}$$

where $P(n|k)$ and $\phi(n|k)$ are computed from (10) and (12), respectively. Here, P_{rated} denotes the rated power, and ϕ_{nom} is used to normalize the dimensionless battery-usage proxy. In particular, $\phi(n|k)$ is the predicted battery-usage proxy obtained by evaluating (12) along the predicted state-input pair $(\mathbf{x}(n|k), \mathbf{u}(n|k))$. This normalization enables a more systematic tradeoff among tracking performance, control smoothness, and energy-related metrics within a unified optimization framework.

To reduce the terminal deviation from the reference, the terminal cost is chosen as

$$V_f(\mathbf{X}(N_p|k)) = \|\mathbf{X}(N_p|k) - \mathbf{X}^{\mathrm{ref}}(N_p|k)\|_{P_f}^2, P_f \succ 0. \tag{23}$$

Accordingly, the finite-horizon cost function at time k is formulated as

$$J_{opt}(k) = \sum_{n=0}^{N_p-1} L(n|k) + V_f(\mathbf{X}(N_p|k)). \tag{24}$$

Compared with conventional MPC formulations that focus solely on trajectory tracking, the cost function formulated in (21)–(24) incorporates tracking accuracy, control deviation, control smoothness, energy consumption, and battery-usage moderation in a unified manner. This formulation better captures the practical control requirements of the under-panel PV inspection robot.

Safety assurance for the inspection robot

In this subsection, the safety functions constructed are adopted as candidate CBFs to maintain system safety over the prediction horizon. With a slight abuse of notation, these safety functions are also viewed here as functions of a generic state $\mathbf{x}$ at prediction step n. At each prediction step n, the condition $h(\mathbf{x}, n|k) \geq 0$ indicates that the corresponding safety constraint is satisfied. Accordingly, the time-varying safe set over the prediction horizon is defined as

$$\mathcal{C}_{n|k} = \left\{ \mathbf{x} \in \mathbb{R}^5 : \begin{array}{l} h_o^{\mathrm{obs}}(\mathbf{x}, n|k) \geq 0, \ \forall o \in \mathcal{O}(k), \\ h_m^{\mathrm{bd}}(\mathbf{x}, n|k) \geq 0, \ \forall m = 1, \ldots, M, \\ h^{\mathrm{soc}}(\mathbf{x}, n|k) \geq 0 \end{array} \right\}. \tag{25}$$

To ensure that the predicted states remain within the time-varying safe set (25) over the prediction horizon in discrete time, the following CBF propagation condition is imposed at each prediction step for every safety function defining $\mathcal{C}_{n|k}$:

$$h(\mathbf{x}(n+1|k), n+1|k) \geq \alpha h(\mathbf{x}(n|k), n|k), \ \ \alpha \in [0,1) \tag{26}$$

This condition implies that if $\mathbf{x}(n|k) \in \mathcal{C}_{n|k}$ and the applied control input is chosen such that (26) holds for all constituent safety functions, then $\mathbf{x}(n+1|k) \in \mathcal{C}_{n+1|k}$.[26,27,31] By recursively enforcing this condition over the prediction horizon, safety of the predicted state sequence can be guaranteed.

Based on this property, the safety functions defined in section "Safety constraints in under-panel corridors" are incorporated in a unified manner into the safety-constrained predictive control formulation. Specifically, for $n = 0, \ldots, N_p - 1$, the corresponding constraints are written as

$$h_o^{\mathrm{obs}}(n+1|k) \geq \alpha_o h_o^{\mathrm{obs}}(n|k) - \xi_o^{\mathrm{obs}}(n|k), \tag{27a}$$

$$h_m^{\mathrm{bd}}(n+1|k) \geq \alpha_b h_m^{\mathrm{bd}}(n|k) - \xi_m^{\mathrm{bd}}(n|k), \tag{27b}$$

$$h^{\mathrm{soc}}(n+1|k) \geq \alpha_s h^{\mathrm{soc}}(n|k) - \xi^{\mathrm{soc}}(n|k). \tag{27c}$$

where α_o, α_b, and $\alpha_s \in [0,1)$ are the decay parameters associated with the corresponding CBFs, and $\xi_o^{\mathrm{obs}}(n|k)$, $\xi_m^{\mathrm{bd}}(n|k)$, and $\xi^{\mathrm{soc}}(n|k)$ are nonnegative slack variables. To avoid relaxing safety at the imminent prediction step, these slack variables are set to zero at $n = 0$, so that the first predicted state must satisfy the safety constraints without relaxation. For compact notation in (21), these slack variables are collected into the aggregate vector $\xi(n|k)$.

Safety-critical economic MPC formulation

Based on the cost function (24) and the safety constraints (27), this subsection formulates the safety-critical economic MPC problem.

In addition, a terminal SoC threshold s_{ter} is introduced to specify the minimum allowable SoC at the end of the prediction horizon.

The terminal predicted state is also The terminal predicted state is also required to lie in the admissible state set $\mathbf{X}$.

Let the control input sequence be denoted by $\mathbf{U}_k := \{\mathbf{u}(n|k)\}_{n=0}^{N_p-1}$ and the slack-variable sequence be denoted by $\Xi_k := \{\xi(n|k)\}_{n=0}^{N_p-1}$. Accordingly, at each sampling instant k, the following finite-horizon optimization problem is solved:

$$\min_{\mathbf{U}_k, \Xi_k} \quad J_{opt}(k), \tag{28a}$$

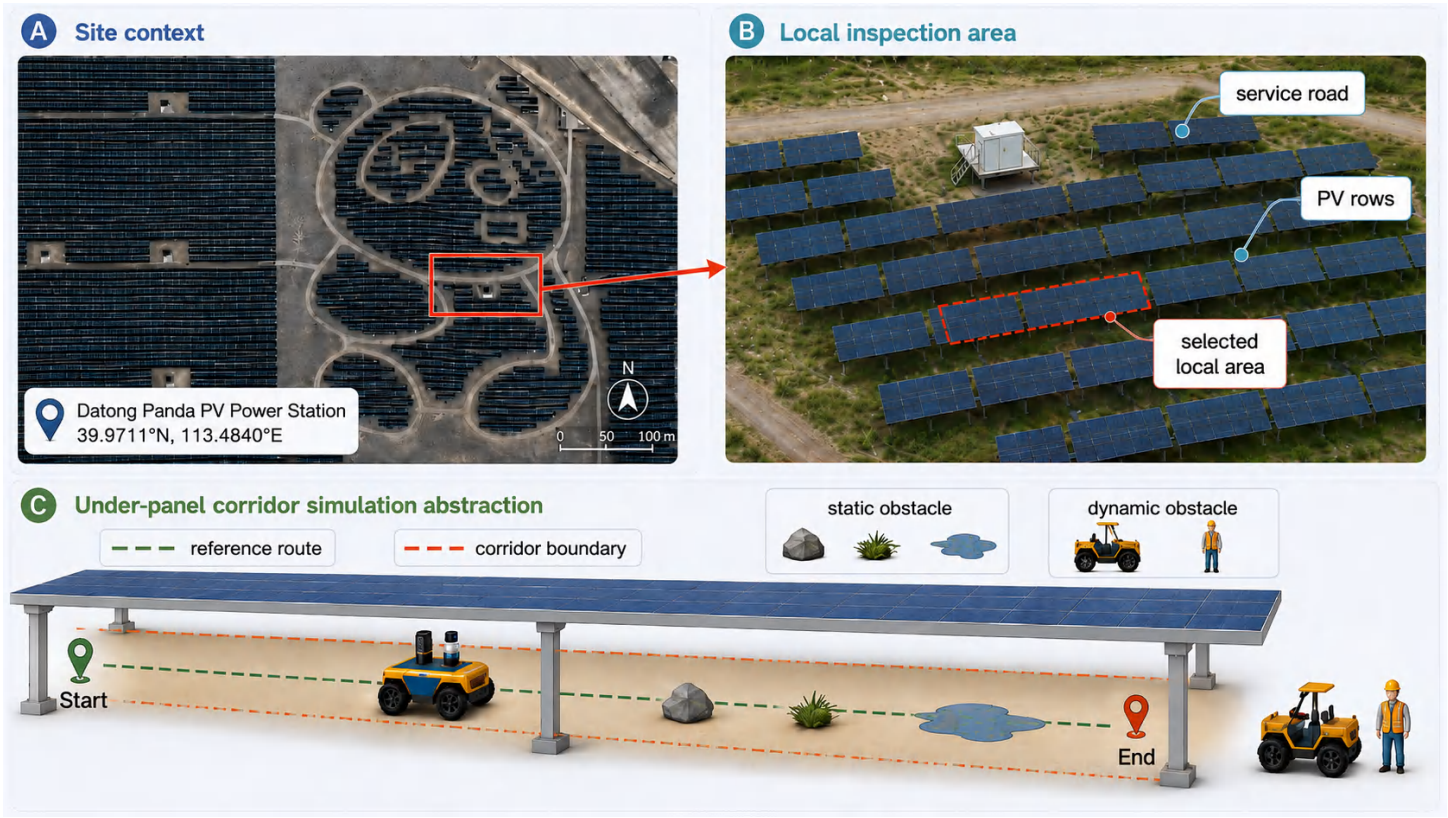


Figure 2. Simulation scenario for under-panel PV inspection. (A) Satellite view of the selected local area in the Datong Panda PV Power Station. (B) Enlarged local inspection area used as the prototype. (C) Corridor-level simulation abstraction with the reference route, corridor boundary, static obstacles, and dynamic obstacle.

$$\text{s.t.} \quad \mathbf{x}(0|k) = \mathbf{x}(k), \quad \text{for } n = 0, \ldots, N_p - 1, \tag{28b}$$

$$\mathbf{x}(n+1|k) = f_d(\mathbf{x}(n|k), \mathbf{u}(n|k)), \tag{28c}$$

$$\mathbf{u}(n|k) \in \mathcal{U}, \ \mathbf{x}(n|k) \in \mathcal{X}, \tag{28d}$$

$$\xi(n|k) \geq 0, \ \xi(0|k) = 0, \tag{28e}$$

$$\text{the safety constraints in (27)}, \tag{28f}$$

$$\mathbf{x}(N_p|k) \in \mathcal{X}, \ s(N_p|k) \geq s_{\text{ter}}. \tag{28g}$$

After solving (28), only the first element of the optimal input sequence is applied to the robot, namely,

$$\mathbf{u}(k) = \mathbf{u}^*(0|k). \tag{29}$$

The optimization is then repeated at the next sampling instant using updated state measurements.

Problem (28) is a nonlinear optimization problem because the prediction model contains trigonometric terms and the obstacle-related constraints are generally nonconvex. For online implementation, this paper solves (28) using sequential quadratic programming (SQP),[29] with the optimal solution at the previous time step used as a warm start. The control framework defined by (28) provides an online control strategy for the robot operating in constrained inspection environments while balancing safety, feasibility, and economic performance.

SIMULATION RESULTS AND DISCUSSION

This section verifies the proposed energy-efficient MPC-CBF controller in an under-panel PV inspection scenario. Ideally, the proposed method should be compared with existing unmanned ground vehicle-based PV inspection methods. However, related studies mainly focus on global path planning or inspection framework design,[17,18,21] and rarely address local obstacle-avoidance execution. Since MPC-CBF has been applied to safety-critical inspection-robot trajectory tracking,[32] standard MPC-CBF[25] is adopted as the baseline. For a fair comparison, the baseline uses the same robot model, prediction horizon, sampling time, state and input constraints, obstacle prediction model, corridor-boundary constraints, CBF decay factors, slack variables, and slack penalty weight as the proposed controller. The difference is that the baseline removes the energy-consumption and battery-usage proxy terms from the objective function.

Simulation scenario and parameter setup

To emulate a realistic under-panel PV inspection task, a geometrically scaled 100 m × 80 m scenario is constructed from a selected local array area of the Datong Panda PV Power Station in Shanxi, China. As shown in Figure 2A & B, the red-boxed satellite-image region and its enlarged local view provide the geometric prototype, including PV panel rows, under-panel passages, and a nearby service road. This region is mapped to the simulation domain as an under-panel inspection area, where the corridor boundaries are defined by fixed infrastructure and PV support columns and the reference route is generated within the traversable area. The resulting scenario includes several PV panel rows with different lengths, narrow under-panel corridors, and a transformer area. The reference inspection trajectory is assumed to be generated in advance by an upper-level planning module, while the controller is only responsible for real-time trajectory tracking and local collision avoidance. To reflect typical O&M conditions, temporary obstacles are introduced into the scenario. Specifically, static obstacles are modeled as under-panel puddles, stones, trees, or other objects that may appear near the inspection corridor, whereas the dynamic obstacle is modeled as a small service vehicle moving along a public road. Therefore, the robot is expected to temporarily deviate from the predefined route to avoid these obstacles and then return to the reference trajectory.

The comparison is conducted in terms of tracking accuracy, obstacle safety margins, instantaneous power, total and motion-related energy consumption, completed inspection distance, SoC evolution, and cumulative

Table 1. Simulation parameter settings.

Parameter	Symbol	Value	Description
Sampling time	Δt	0.8 s	Control and simulation sampling interval
Prediction horizon	N_p	5	Number of MPC prediction steps
Tracking weight	Q	diag(60,60,8,6)	Weight for tracking error
Control weight	R	diag(1.2,0.8)	Weight for control deviation
Input-increment weight	S	diag(8,5)	Weight for input variation
Terminal weight	P_f	diag(80,80,10,8)	Weight for terminal tracking error
Robot radius	r	0.34 m	Equivalent collision radius
Corridor half-width	w_c	1.25 m	Half-width of the inspection corridor
Safety margins	δ_o, δ_b	0.20, 0.36 m	Obstacle and corridor-boundary margins
Initial velocity	$v(0)$	0 m/s	Robot starts from rest
Speed bounds	$v_{\min}, v_{\max}$	0, 0.55 m/s	Longitudinal speed limits
Acceleration bounds	$a_{\min}, a_{\max}$	−0.16, 0.20 m/s^2	Longitudinal acceleration limits
Yaw-rate bounds	$\omega_{\min}, \omega_{\max}$	±0.34 rad/s	Yaw-rate limits
Local obstacle range	R_o	12 m	Range for local obstacle selection
Moving-obstacle speed	v_o	0.08 m/s	Speed of the service vehicle
Minimum safe SoC	$s_{\min}$	0.20	Lower SoC threshold for safety evaluation
Terminal SoC threshold	s_{ter}	0.20	Terminal SoC threshold, set equal to $s_{\min}$ in simulations
Healthy SoC threshold	s_{hlth}	0.40	Preferred SoC threshold in the battery-usage proxy
Battery capacity	$E_{\max}$	600 Wh	Battery capacity used in SoC update
Baseline power	P_0	50 W	Sensing, computation, and communication power
Power coefficients	$\alpha_v, \alpha_a, \alpha_\omega$	60, 380, 320	Velocity, acceleration, and yaw-rate terms
Rated power	P_{rated}	300 W	Power normalization for the battery-usage proxy
Nominal battery-usage proxy	ϕ_{nom}	1.0	Normalization constant for the battery-usage proxy
Battery-proxy weights	$\beta_P, \beta_s, \beta_a$	1.0, 4.0, 0.45	High-power, low-SoC, and acceleration penalties
Energy-cost weights	ρ_P, ρ_ϕ	180, 16	Proposed controller energy-related weights
Slack penalty weight	ρ_ξ	5.0×10^6	Penalty for relaxed CBF constraints
CBF decay factors	$\alpha_o, \alpha_b, \alpha_s$	0.66, 0.86, 0.74	Obstacle, boundary, and SoC CBF factors

battery-usage proxy. Let $p(k) = [x(k), y(k)]^\top$ and $d_o(k) = \|p(k) - p_o(k)\|$. The main evaluation indices used in the figures and tables are defined as follows. The tracking and safety indices are $e_p(k) = \|[e_x(k), e_y(k)]^\top\|_2$, $e_{\text{rms}} = \sqrt{N^{-1}\sum_{k=0}^{N-1} e_p^2(k)}$, $e_{\max} = \max_k e_p(k)$, and $m_o^{\min} = \min_k (d_o(k) - r - r_o - \delta_o)$. The energy and battery-usage indices are $P_{\max} = \max_k P(k)$, $E = (\Delta t/3600)\sum_{k=0}^{N-1} P(k)$, $E_{\text{mot}} = (\Delta t/3600)\sum_{k=0}^{N-1} \max(P(k) - P_0, 0)$, $B_{\text{cum}} = \Delta t \sum_{k=0}^{N} \phi(k)$, $E_{\text{m}} = E/D_{\text{comp}}$, and $B_{\text{m}} = B_{\text{cum}}/D_{\text{comp}}$. Here, N denotes the number of executed sampling intervals in the corresponding simulation run. The quantities $e_x(k)$ and $e_y(k)$ are the position-error components of $\mathbf{e}(k)$ in (7). The index $o \in \{\text{st}, \text{dyn}\}$ denotes the static obstacle or the moving service vehicle, and D_{comp} is the completed inspection distance in the limited-energy case.

The computational indices t_{avg} and $t_{\max}$ denote the average and maximum computation times per MPC update, respectively.

Two cases are considered. Case 1 requires both controllers to complete the same full inspection task and therefore evaluates energy saving under a fixed route length. Case 2 uses the same available energy budget for both controllers, and therefore evaluates the completed inspection distance under the same available energy budget. The simulation parameters used in both cases are listed in Table 1. In the power model, α_a and α_ω are set larger than α_v to penalize acceleration and yaw-rate terms more strongly than steady low-speed motion. In the battery-usage proxy, β_P, β_s, and β_a weight high-power, low-SoC, and acceleration terms, respectively; β_s is set larger to emphasize low-SoC avoidance. The terminal threshold is set to $s_{\text{ter}} = s_{\min} = 0.20$, so no additional terminal SoC reserve is imposed. All simulations were conducted in MATLAB on a desktop computer equipped with an Intel(R) Core(TM) i7-4970 CPU at 3.60 GHz. The maximum number of SQP iterations was set to 100 at each sampling step, and the termination tolerances for optimality, step size, and constraint violation were set to 10^{-6}.

Case 1: fixed-task energy comparison

Case 1 requires both controllers to complete the same predefined inspection trajectory. Both simulations start from the same initial state of charge, $s(0) = 0.80$. As shown in Figure 3, the standard MPC-CBF and the proposed controller both track the reference route and temporarily deviate from it when the robot approaches the static obstacles and the moving service vehicle. Since the same CBF-based safety constraints are imposed in both controllers, the two trajectories remain within the corridor and avoid the local obstacles. The difference lies mainly in the recovery behavior after avoidance: the standard MPC-CBF tends to return to the reference more rapidly, whereas the proposed controller accepts a smoother local deviation to avoid high-power control actions. Figure 4A compares $e_p(k)$, while Table 2 reports the corresponding tracking and safety-margin indices e_{rms}, $e_{\max}$, and $m_o^{\min}$. The

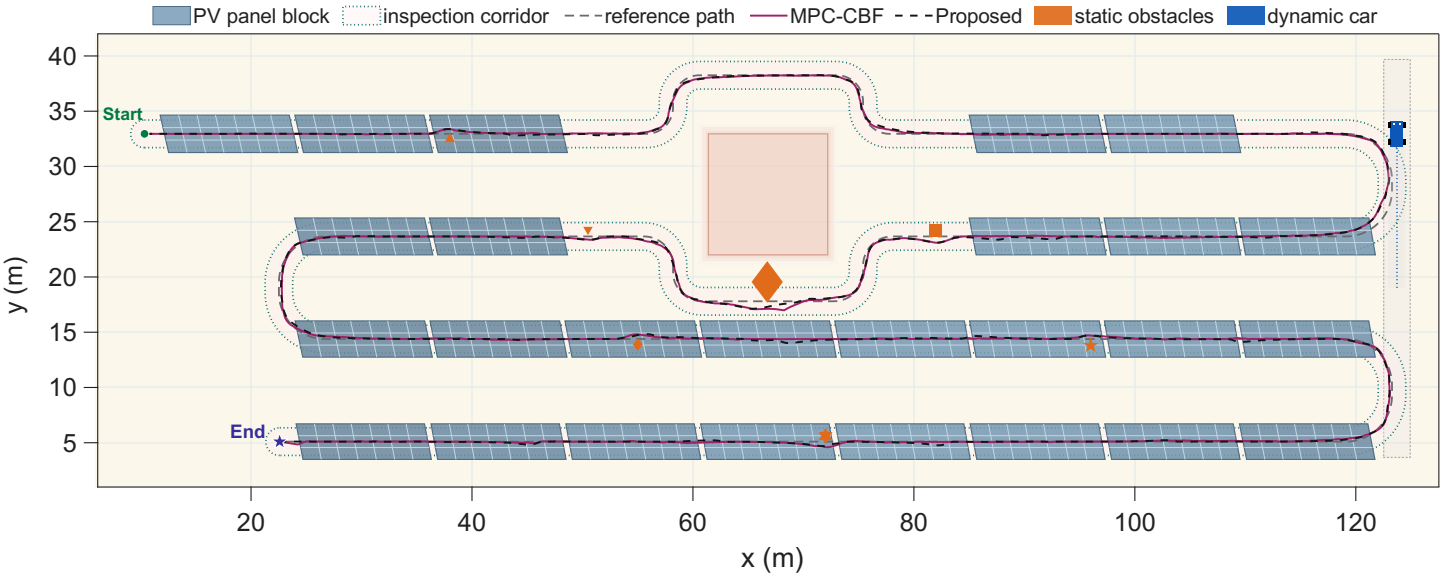


Figure 3. Trajectory comparison under the fixed inspection task.

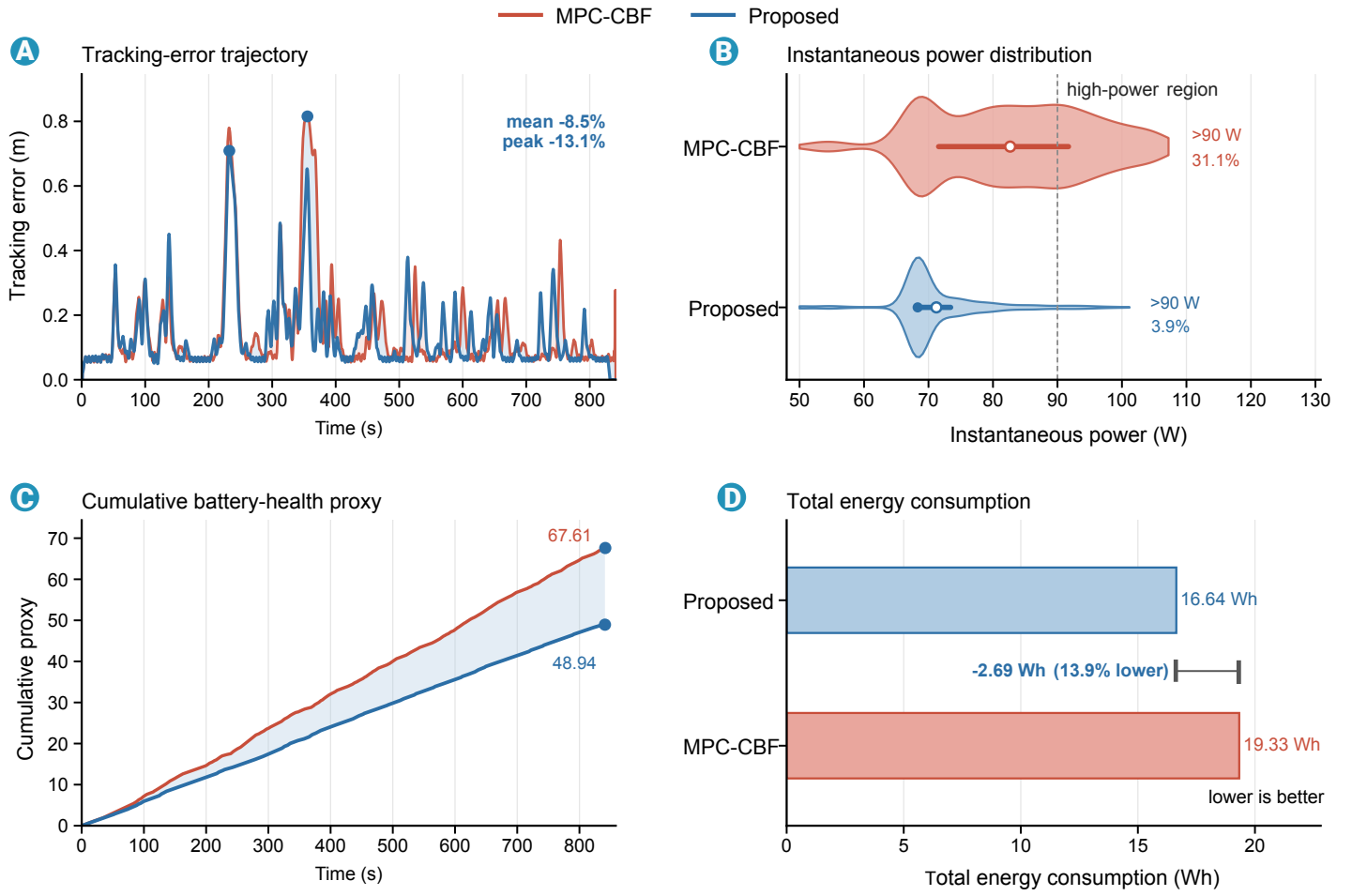


Figure 4. Simulation results under the fixed inspection task.

proposed controller gives a similar tracking profile and slightly lower mean and peak errors, indicating that the energy-related terms do not degrade path-tracking capability. The positive $m_{\mathrm{dyn}}^{\mathrm{min}}$ values also confirm that the moving service vehicle is avoided in both cases.

The energy-related results are shown in Figure 4B-D The proposed controller substantially reduces the high-power operating portion, lowers the cumulative battery-usage proxy B_{cum} from 67.61 to 48.94, and decreases the total energy consumption E from 19.33 to 16.64Wh. Table 2 further shows that the motion-related energy E_{mot} excluding the baseline power P_0 decreases from 7.64 to 4.96Wh. Thus, for the same inspection route and

Table 2. Quantitative comparison under the fixed inspection task.

Controller	RMSE (m) e_{rms}	Max. error (m) e_{max}	Static margin (m) m_{st}^{min}	Dynamic margin (m) m_{dyn}^{min}	Motion energy (Wh) E_{mot}	Peak power (W) P_{max}	Avg. t (s) t_{avg}	Max. t (s) t_{max}
MPC-CBF	0.216	0.815	0.180	0.238	7.64	120.34	0.528	2.164
Proposed	0.183	0.724	0.143	0.188	4.96	113.26	0.756	2.278

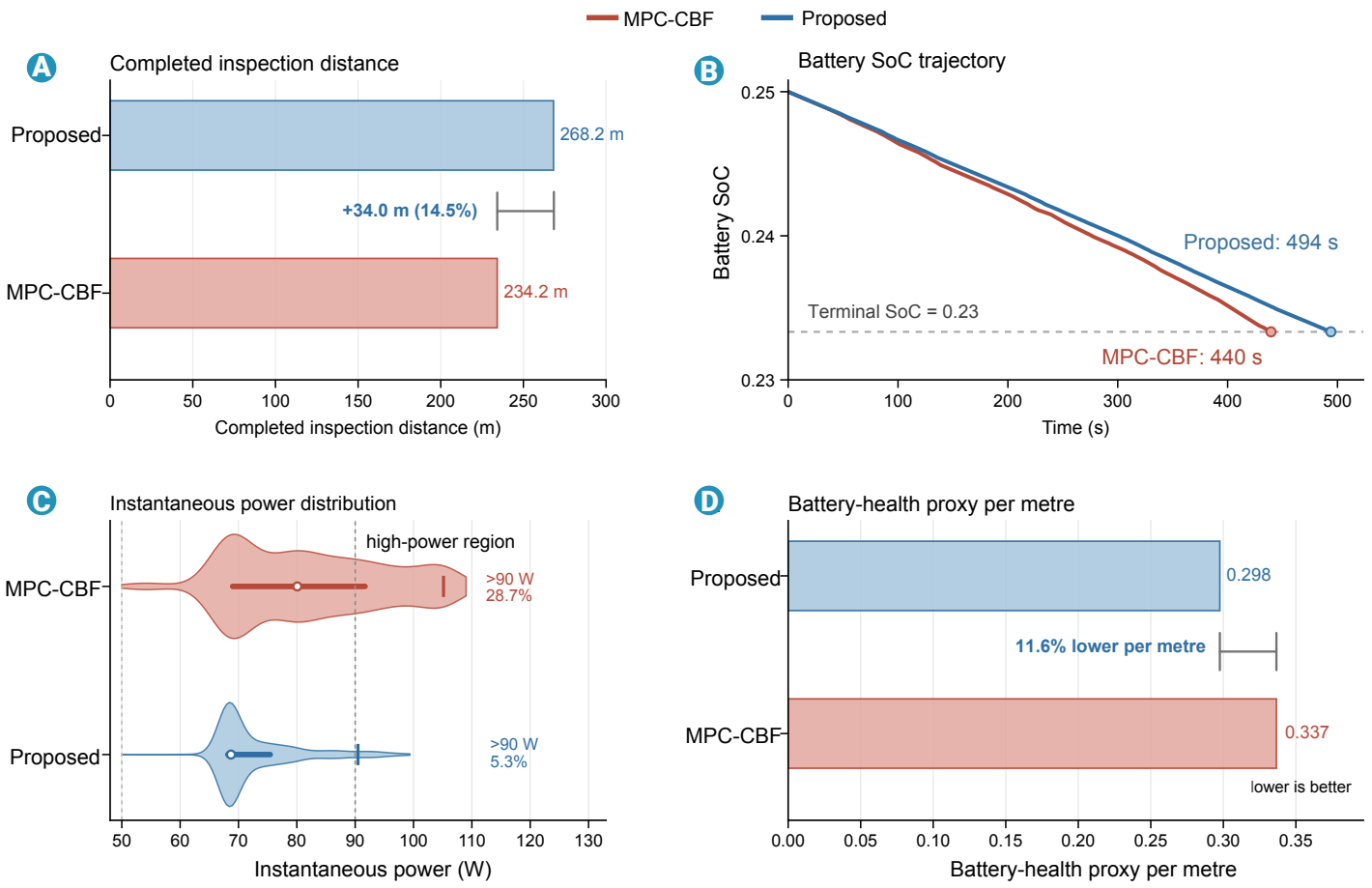


Figure 5. Simulation results under the limited energy budget.

Table 3. Quantitative comparison under the limited energy budget.

Controller	RMSE (m) e_{rms}	Max. error (m) e_{max}	Static margin (m) m_{st}^{min}	Dynamic margin (m) m_{dyn}^{min}	Energy/ m (Wh/m)] E_m	Peak power (W)] P_{max}	Avg. t (s) t_{avg}	Max. t (s) t_{max}
MPC-CBF	0.998	1.608	0.011	0.244	0.043	120.34	0.511	2.212
Proposed	1.053	1.608	0.009	0.194	0.037	114.87	0.769	2.456

similar tracking and safety performance, the proposed controller consumes less electrical energy mainly through reduced high-power operation and smoother acceleration commands.

Case 2: limited-energy endurance comparison

In this limited-energy test, both controllers start from the same initial SoC of 0.25, and each simulation is stopped when the common 10 Wh energy budget is exhausted. This budget corresponds to an SoC decrease of 10 Wh/E_{max} ≈ 0.017; therefore, the terminal SoC is about 0.23, which remains above the lower-bound thresholds s_{min} and s_{ter}. These thresholds are constraints rather than target terminal SoC values, so Case 2 evaluates the completed inspection distance under the same available energy budget rather than whether the whole route can be finished.

The resulting endurance performance is summarized in Figure 5. Under the same available energy budget, the proposed controller completes 268.2m, which is 34.0m longer than the standard MPC-CBF result. The SoC trajectory also shows a longer operating time under the same energy budget. In addition, the proposed controller reduces the high-power operating portion and lowers B_m from 0.337 to 0.298. This trend is consistent with the fixed-task result and is associated with fewer high-power samples and smoother control transients over the executed route. The quantitative energy, tracking, and safety indices in Table 3 are computed over the executed portion before each controller exhausts the energy budget. The larger RMSE and maximum-error values in Table 3 compared with the fixed-task case are mainly due to the limited-energy evaluation protocol: each controller is evaluated over the executed segment before the common energy budget is exhausted, rather than over the same completed route. The proposed controller travels farther and tends to use smoother, less aggressive recovery motions after avoid-

ance, which can slightly increase the average tracking error; nevertheless, its maximum error remains comparable to the baseline and the safety margins remain positive. Although the proposed controller travels farther, E_m decreases from 0.043 to 0.037Wh/m. These results verify that the endurance improvement is not obtained by sacrificing local obstacle avoidance or basic trajectory-tracking performance. As a complementary indication of real-time feasibility, the computation-time columns in Tables 2 & 3 show that the average computation time remains below the sampling period of 0.8s. The computation time exceeds one sampling period only occasionally under complex maneuvers, such as turning or active obstacle-avoidance constraints; in these rare overrun steps, the control command available from the previous MPC update is applied.

CONCLUSIONS

This paper investigated an energy-efficient under-panel PV inspection strategy for a low-profile ground robot, where trajectory tracking, local obstacle avoidance, corridor-boundary protection, and energy-related penalties are integrated through an MPC-CBF-based predictive framework. Simulations based on a local array of the Datong Panda PV Power Station show that, compared with standard MPC-CBF, the proposed method reduces total energy consumption from 19.33 to 16.64Wh in the fixed-task test and increases the completed inspection distance from 234.2 to 268.2m under the same energy budget, corresponding to improvements of 13.9% and 14.5%, respectively. These results demonstrate its potential for energy-saving under-panel PV inspection, while future work will consider field validation and tighter coupling with onboard perception.

REFERENCES

1. Høiaas I., Grujic K., Imenes A.G., et al. (2022). Inspection and condition monitoring of large-scale photovoltaic power plants: A review of imaging technologies. *Renew. Sustain. Energy Rev.* **161**:112353. DOI:10.1016/j.rser.2022.112353
2. Zhang H., Chen Y., Jin Y., et al. (2025). Greenhouse-integrated solar photovoltaics for sustainable future energy landscapes. *Innov. Energy* **2**:100124. DOI:10.59717/j.xinn-energy.2025.100124
3. He T., Chen Y., Zhao Y., et al. (2025). Harnessing the solar photovoltaic potential in global mining areas: Energy locations for the future. *The Innovation* **6**:100987. DOI:10.1016/j.xinn.2025.100987
4. Shao C., Yang K., Zhang X., et al. (2025). Land constraint makes solar trackers less effective for photovoltaic power generation. *Innov. Energy* **2**:100103. DOI:10.59717/j.xinn-energy.2025.100103
5. Akram M.W., Li G., Jin Y., et al. (2022). Failures of photovoltaic modules and their detection: A review. *Appl. Energy* **313**:118822. DOI:10.1016/j.apenergy.2022.118822
6. Tang W., Yang Q., Dai Z., et al. (2024). Module defect detection and diagnosis for intelligent maintenance of solar photovoltaic plants: Techniques, systems and perspectives. *Energy* **297**:131222. DOI:10.1016/j.energy.2024.131222
7. Polymeropoulos I., Bezyrgiannidis S., Vrochidou E., et al. (2024). Enhancing solar plant efficiency: A review of vision-based monitoring and fault detection techniques. *Technologies* **12**(10):175. DOI:10.3390/technologies12100175
8. Kuznetsov P., Kotelnikov D., Yuferev L., et al. (2022). Method for the automated inspection of the surfaces of photovoltaic modules. *Sustainability* **14**:11930. DOI:10.3390/su141911930
9. Michail A., Livera A., Tziolis G., et al. (2024). A comprehensive review of unmanned aerial vehicle-based approaches to support photovoltaic plant diagnosis. *Heliyon* **10**:e23983. DOI:10.1016/j.heliyon.2024.e23983
10. Buerhop C., van Dyk E.E., Vorster F.J., et al. (2025). Enhancing the cost- and time-effectiveness of field PV module inspection by UV-fluorescence imaging. *IEEE J. Photovolt.* **15**:30–39. DOI:10.1109/JPHOTOV.2024.3492286
11. Addabbo P., Angrisano A., Bernardi M.L., et al. (2018). UAV system for photovoltaic plant inspection. *IEEE Aerosp. Electron. Syst. Mag.* **33**:58–67. DOI:10.1109/MAES.2018.170145
12. Li X., Li W., Yang Q., et al. (2020). An unmanned inspection system for multiple defects detection in photovoltaic plants. *IEEE J. Photovolt.* **10**:568–576. DOI:10.1109/JPHOTOV.2019.2955183
13. de Oliveira A.K.V., Aghaei M., and Rüther R. (2022). Automatic inspection of photovoltaic power plants using aerial infrared thermography: A review. *Energies* **15**:2055. DOI:10.3390/en15062055
14. Barraz Z., Sebari I., Lamrini N., et al. (2025). Fast and automatic solar module geo-labeling for optimized large-scale photovoltaic systems inspection from UAV thermal imagery using deep learning segmentation. *Cleaner Eng. Technol.* **28**:101048. DOI:10.1016/j.clet.2025.101048
15. Mahdavipour Z. (2024). Defect inspection of photovoltaic solar modules using aerial electroluminescence (EL): A review. *Sol. Energy Mater. Sol. Cells* **278**:113210. DOI:10.1016/j.solmat.2024.113210
16. Solend T.A., Rødningsby A., and Moen J. (2025). Impacts of infrared thermographic image blurring on UAV inspection efficiency of solar power plants. *Sol. Energy* **299**:113673. DOI:10.1016/j.solener.2025.113673
17. Figgis B., Azid S.I., and Parlevliet D.A. (2025). Review of unmanned ground vehicles for PV plant inspection. *Sol. Energy* **291**:113404. DOI:10.1016/j.solener.2025.113404
18. Wang F., Gao Y., Chen Z., et al. (2023). A path planning algorithm of inspection robots for solar power plants based on improved RRT*. *Electron.* **12**:4455. DOI:10.3390/electronics12214455
19. Wang Y. (2017). Design of inspection robot system based on photovoltaic effect. *Adv. Mater. Mach. Elect. Eng.* **114**:259–262. DOI:10.2991/ammee-17.2017.51
20. Hu H., Chen Q., Xie Y., et al. (2025). A novel robotic platform for photovoltaic panel inspection with progressive structural configuration and constant speed regulation. *EEICE* :1535–1538. DOI:10.1109/EEICE65049.2025.11034170
21. De Benedetti M.M., Bascetta L., Falsone A., et al. (2024). Automated photovoltaic power plant inspection via unmanned vehicles. *IEEE Trans. Control Syst. Technol.* **32**:399–412. DOI:10.1109/TCST.2023.3316093
22. Amorim J.S.J.C., Neto A.F.S., Chaves R.S., et al. (2025). Collaborative inspection of solar panel farms using YOLOv5-based computer vision and UGV-UAV integration. *J. Intell. Robot. Syst.* **111**:66. DOI:10.1007/s10846-025-02265-w
23. Velasco-Sánchez E., Recalde L.F., Guevara B.S., et al. (2024). Visual servoing NMPC applied to UAVs for photovoltaic array inspection. *IEEE Robot. Autom. Lett.* **9**:2766–2773. DOI:10.1109/LRA.2024.3360876
24. Xing J., Cioffi G., Hidalgo-Carrió J., et al. (2023). Autonomous power line inspection with drones via perception-aware MPC. *IROS* :1086–1093. DOI:10.1109/IROS55552.2023.10341871
25. Gao L., Gao Y., He M., et al. (2025). Safety-critical model predictive control for trajectory tracking and obstacle avoidance of wheeled mobile robot formations. *Trans. Inst. Meas. Control* **48**:1–14. DOI:10.1177/01423312251331274
26. Jiang C. and Guo Y. (2024). Incorporating control barrier functions in distributed model predictive control for multirobot coordinated control. *IEEE Trans. Control Netw. Syst.* **11**:547–557. DOI:10.1109/TCNS.2023.3290430
27. Zeng J., Zhang B. and Sreenath K. (2021). Safety-critical model predictive control with discrete-time control barrier function. *ACC* :3882–3889. DOI:10.23919/ACC50511.2021.9483029
28. Kim Y., He C.R. and Singh T. (2024). Control barrier function based energy optimal obstacles avoidance for point-to-point maneuvers. *IFAC-PapersOnLine* **58**:857–862. DOI:10.1016/j.ifacol.2025.01.103
29. Zhu L., Tao F., Fu Z., et al. (2025). Safety-involved co-optimization of speed trajectory and energy management for fuel cell-battery electric vehicle in car-following scenarios. *Complex Intell. Syst.* **11**:89. DOI:10.1007/s40747-024-01698-4
30. Nie Z., Jia Y., Wang W., et al. (2022). Eco-co-optimization strategy for connected and automated fuel cell hybrid vehicles in dynamic urban traffic settings. *Energy Convers. Manag.* **263**:115690. DOI:10.1016/j.enconman.2022.115690
31. Gao L., Gao Y.-F., Wang X.-F., et al. (2026). Space-constrained multi-robot formation control via an enhanced discrete-time control Lyapunov and barrier function approach. *IEEE Trans. Autom. Sci. Eng.* **23**:6190–6203. DOI:10.1109/TASE.2026.3669894
32. Wei L., Liao Y., Liu J., et al. (2025). Safety-critical model predictive trajectory tracking control for transformer inspection robots based on control barrier functions. *PowerCon* :1–6. DOI:10.1109/POWERCON66300.2025.11294480

FUNDING AND ACKNOWLEDGMENTS

This work was supported in part by the Joint Fund of the National Natural Science Foundation of China under Grant U24A20263, in part by the National Natural Science Foundation of China under Grant 62473071, and in part by the Major Program of the National Natural Science Foundation of China under Grant 62595804.

AUTHOR CONTRIBUTIONS

Liang Gao designed the study, developed the method, and drafted the manuscript. Xinhui Zhao performed the simulations. Yuhu Wu supervised the research and revised the manuscript. Shuting Le, Zongyang Lv, Xun Shen, and Benlian Xu contributed to technical discussions, result analysis, and manuscript revision.

DECLARATION OF INTERESTS

The authors declare no competing interests.

DATA AND CODE AVAILABILITY

Data are available from the corresponding author upon reasonable request.