2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Kinetic-energy-optimal and Safety-guaranteed Trajectory Planning for Bridge Inspection Robot Manipulator

Tianyu Zhang[1], Yong Chang[2], Hongguang Wang[3] and Tianlong Wang[4]

***Abstract*— Bridge inspections are essential for maintaining key infrastructure and preventing structural and functional failures. Nevertheless, traditional manual inspection techniques are plagued by laboriousness, high risk, and low efficiency. Although numerous automation inspection methods have been studied, inspection performance remains challenging. The main difficulties are redundant mechanisms, complex control, high energy consumption, and limited autonomy and safety. To address these problems, we are developing a small, lightweight, electrically-driven robotic manipulator for bridge inspection named the BIRM. Here, we propose a kinetic-energy-optimal and safety-guaranteed trajectory planning for BIRM. Compared with existing methods, it simultaneously addresses energy consumption and safety. The approach formulates a quadratic programming (QP) problem by considering the robot's kinetic energy as the objective function, and the augmented Lagrange multiplier (ALM) is applied to find the solution of the QP. The proposed method completely satisfies the joint position, velocity, and acceleration limits at the speed level while considering collision avoidance. In this paper, the collision detection strategy can achieve low-complexity computation through several structural parameters of the bridge, thereby quickly adapting to environmental changes. Through simulation experiments, we validate the effectiveness and superiority of the proposed method. Through physical experiments, we demonstrate the sustainability and safety of bridge inspections in the field.**

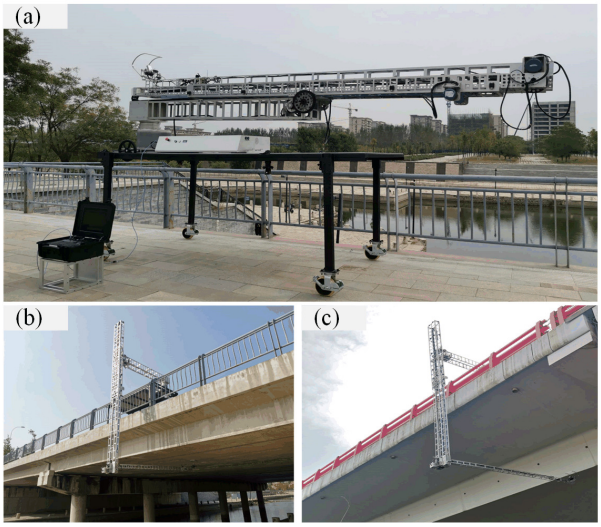

Fig. 1. Real-world experiments for kinetic-energy-optimal and safety-guaranteed trajectory planning. (a) A small, lightweight, electric-driven bridge inspection robot manipulator (BIRM). (b) shows the testing of Bridge-I. (c) shows the testing of Bridge-II.

I. INTRODUCTION

Bridge collapses have resulted in significant loss of life and property damage. Bridge cracks are a major cause of damage in collapse accidents. Therefore, inspecting and analyzing bridge cracks is crucial for maintaining them. However, traditional manual inspection methods are highly labor-intensive, risky, and inefficient [1]. Some solutions have been developed, such as truss-type bridge inspection vehicles and arm-type bridge inspection vehicles equipped with small robotic arms or mobile inspection robots [2]–[5]. However, the core structure of these solutions is hydraulic-driven robotic arms, resulting in redundant mechanisms, complex control, high energy consumption, limited autonomy, and low safety. Transitioning from a hydraulic system to an electric system can greatly reduce energy consumption. This transition can extend the operating time. Therefore, we are developing a small, lightweight, electric-driven robotic manipulator for bridge inspection, as shown in Fig.1. Minimizing energy consumption is essential for the bridge inspection robot manipulator, which enhances the manipulation continuity under limited energy in the field. The primary strategy for reducing energy consumption is trajectory planning [6]. This paper aims to ensure that the BIRM can safely and continuously inspect bridges under limited energy using kinetic-energy-optimal trajectory planning.

Trajectory planning is a highly challenging task in robotics [7]. For the trajectory planning problems of redundant robotic arms, redundancy resolution is typically used to enhance specific performances and avoid joint physical constraints of joint angles, velocities, accelerations, and obstacles to follow the end-effector trajectory. Current methods based on the pseudoinverse of the Jacobian matrix have been widely studied, but these methods cannot integrate inequality constraints. For joint physical constraints, [8], [9] used a weighted pseudoinverse to minimize the weighted norm of joint velocities, demonstrating their effectiveness in addressing joint velocity limitations. An extension of the pseudoinverse method is

*This work was supported by the National Key Research and Development Program of China (No.2019YFB1310402) and the Basic Research Program of Shenyang Institute of Automation, Chinese Academy of Sciences (No.2022JC3K06). (*Corresponding author: Hongguang Wang.*)

[1]Tianyu Zhang is with the State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China; Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China, and with the University of Chinese Academy of Sciences, Beijing 100049, China (email: zhangtianyu1@sia.cn).

[2]Yong Chang, [3]Hongguang Wang, and [4]Tianlong Wang are with the State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China, and Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China (e-mail:changyong@sia.cn; hgwang@sia.cn; wangtianlong@sia.cn).

the gradient projection method (GPM), which projects the gradient vectors of optimization criteria into the null space of the Jacobian matrix. These methods have been utilized for joint constraints and obstacle avoidance without impacting the primary task. Redundancy resolution based on quadratic programming (QP) has become a popular research topic due to its ability to simultaneously handle multiple constraints and performance metrics that are explicitly related to decision variables. With improvements in parallel computing efficiency, neural network-based inverse kinematics methods have become popular [10], [11]. However, these methods depend on the computer's performance and can be challenging to interpret in a physical sense. In addition, numerical solutions such as the augmented Lagrangian method (ALM) [12] and SQP [13], [14] work well. ALM is beneficial for problems with multiple inequality constraints. ALM addresses inequality constraints by introducing penalty functions, thereby transforming constrained problems into unconstrained problems. [15] presented an energy optimization method based on the Euler–Lagrange equation, which was designed to optimize energy along trajectories that satisfy kinematic constraints. However, this method has only been proven effective for nonredundant robotic arms.

Minimal energy consumption is a crucial task for field robots. To reduce energy consumption, current research focuses on performance metrics such as joint torque, velocity, and kinetic energy. The redundancy resolution with joint torque is at the acceleration level, while the redundancy resolution with joint velocity and kinetic energy is at the velocity level. However, existing velocity-level planning methods must address joint acceleration limitations, as described in [16], [17]. Reference [18] suggested an acceleration-level method to address joint acceleration constraints. However, acceleration-level approaches cannot directly address velocity-level metrics. Joint angle limits, joint velocity limits, and acceleration limits are considered in the velocity-level approach [19] using a projection neural network method. However, the method exhibits instability in terms of joint physical constraints. Additionally, to guarantee the absolute safety of the robotic arm, trajectory planning must balance joint physical constraints and collision avoidance. However, these two aspects cannot be addressed simultaneously in the existing methods with kinetic energy as the optimization objective [17].

In summary, most trajectory planning methods face a common challenge: simultaneously satisfying the physical constraints of joint position, velocity, and acceleration, along with collision avoidance. In this paper, we formulate the planning problem as a QP with kinetic energy as the objective function. We consider constraints such as kinematics, collision avoidance, and joint physical limitations and use the ALM algorithm to solve the QP numerically. To improve the convergence rate, the algorithm initializes each iteration with a first-order approximation of joint positions.

To overcome these challenges, this paper introduces a trajectory planning method that optimizes kinetic energy and ensures safety. This method utilizes a velocity-level redundancy resolution approach based on QP, using kinetic energy as the performance metric. The proposed method satisfies joint physical constraints (i.e., joint position, velocity, and acceleration limitations) while avoiding obstacles, thus achieving low energy consumption and safe joint trajectories. Initially, trajectory planning is formulated as a QP, and collision avoidance is formulated as an inequality constraint. Joint position, velocity, and acceleration limitations are unified as boundary constraints. In this paper, the PHR-ALM method [20] is used to solve the QP with equality and inequality constraints. In addition, the collision distance between the robotic arm and the bridge is characterized as a low-dimensional QP. Through QP solving, unexpected collisions are detected in real time, and distance information is utilized to formulate inequality constraints, thereby enhancing safety. Based on this method, an online trajectory planner is proposed to avoid real-time matrix inversion.

The contributions of this work are summarized as follows:

- The proposed method efficiently computes joint trajectories with low energy consumption and high safety for the bridge inspection robot manipulator.
- This method effectively addresses the challenge of handling acceleration constraints in velocity-level redundancy resolution.
- Physical experiments validate the sustainability and safety of bridge surfaces in the field, thus overcoming the limitations of existing methods.

The remainder of this paper is organized as follows. Section II describes the mathematical model of the BIRM, including kinematics and kinetic energy. Section III introduces trajectory planning based on velocity-level redundancy resolution, considering joint physical constraints and collision avoidance. Section IV introduces a planner based on the method presented in Section III. Section V validates the effectiveness of the method through experiments. Section VI summarizes the research and proposes further work.

II. Mathematical Model of the Bridge Inspection Robot Manipulator (BIRM)

To detect bridge cracks, a bridge inspection robot manipulator (BIRM) with five degrees of freedom is designed, as shown in Fig.2(a). The BIRM is kinematically redundant for inspection, and its end is equipped with a detection platform. A manipulator is termed kinematically redundant when it possesses more degrees of freedom than is needed to execute a given task. This bridge inspection requires the end of the BIRM to move along the surface of the main beam in three directions within Cartesian space. The kinematic model of the robotic arm is illustrated in Fig.2(b).

A. Task-Oriented Kinematics

For the trajectory planning of BIRM, the goal is to determine the control input $q(t) \in R^n$ that allows the manipulator to follow a desired trajectory, represented by the path $x_{ref}(t) \in R^m$. This must be achieved while meeting various constraints (i.e., kinematic and joint physical limitations),

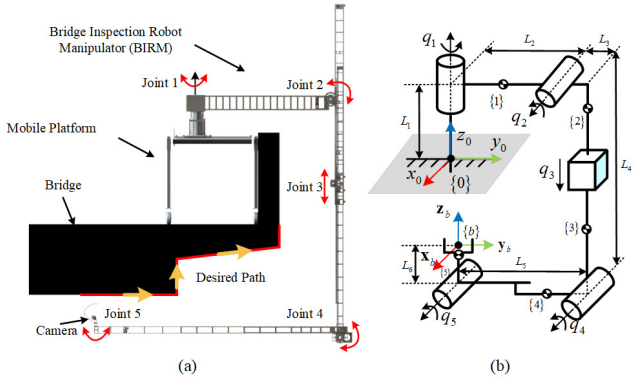

Fig. 2. Mathematical model of the bridge inspection robot manipulator (BIRM). The BIRM is mounted on a mobile platform, such as a vehicle, to collect data from the bridge's surface. Each joint of the BIRM is electrically driven.

minimizing kinetic energy, and avoiding collisions. The kinematic equation of the BIRM is

$$x_{ref}(t) = f(q(t)) \tag{1}$$

where $x_{ref}(t)$ denotes the reference trajectory of the end-effector as a function of time t, and $f(\cdot)$ describes a continuous nonlinear kinematic mapping from the joint space to the task space. In this paper, the dimension of the task space for bridge inspection is $m = 3$, and the dimension of the joint space is $n = 5$.

By taking the derivative of (1), the first-order differential kinematics of the redundant manipulator can be obtained to describe the relationship between task space and joint space.

$$\dot{x}_{ref}(t) = J_t \dot{q}(t) = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{2}$$

where vector $\dot{x}_{ref}(t)$ is the task-space velocity, vector $\dot{q}(t)$ is the joint-space velocity, and $J_t \in R^{m\times n}$ is the task-space Jacobian matrix. $\begin{bmatrix} v_x & v_y & v_z \end{bmatrix}^T$ denotes the linear velocity of the origin of the end-effector frame $\{b\}$ relative to the fixed frame $\{0\}$.

To move the end effector of the redundant manipulator along the given trajectory, we solve (2) to obtain the expected joint velocity $\dot{q}(t)$, given $\dot{x}_{ref}(t)$. Let $\dot{q}_s$ be a particular solution to (2), and let $\dot{q}_0$ be any vector in the null space of J_t. The solution for the expected joint velocity in (2) is

$$\dot{q}(t) = \dot{q}_s + \omega \dot{q}_0 \tag{3}$$

where ω is an arbitrary constant; thus, the kinematic inverse solutions for the BIRM are infinite, allowing optimization based on specific criteria to improve the corresponding performance.

B. Manipulator Kinetic Energy

This paper uses kinetic energy as an optimization performance metric to eliminate uncertainty in redundancy resolution. The overall kinetic energy of the bridge inspection robotic arm is defined as the sum of the kinetic energy of each link, as follows:

$$k = \frac{1}{2}\sum_{i=1}^{n} v_i^T g_i v_i \tag{4}$$

where $v_i \in R^6$ is the velocity vector of the link coordinate system $\{i\}\,(i = 1,2,\cdots,n)$. The origin of this link coordinate system is at the mass center of the link, as shown in Fig.2(b). $g_i \in R^{n\times n}$ is the spatial inertia matrix of link i with respect to the link coordinate system $\{i\}$, as follows:

$$g_i = \begin{bmatrix} I_i & 0 \\ 0 & m_i I \end{bmatrix} \tag{5}$$

where $I_i \in R^{3\times 3}$ is the inertia matrix of link $\{i\}$, m_i is the mass of link i, and I is the identity matrix.

We define the Jacobian matrix $J_{ib}\left(q_{1\cdots i}, 0^{n-i}\right) \in R^{6\times n}$ as the mapping from the origin velocity of the link coordinate system $\{i\}$ to the joint velocity $\dot{q}\left(q_{1\cdots i}, 0^{n-i}\right) \in R^n$.

$$v_i = J_{ib}\left(q_{1\cdots i}, 0^{n-i}\right)\dot{q}\left(q_{1\cdots i}, 0^{n-i}\right) \tag{6}$$

where $q_{1\cdots i}$ represents the joint vector composed of joints from q_1 to q_i. Substituting (6) into (4), the kinetic energy of the manipulator is defined as

$$K = \frac{1}{2}\dot{q}^T M(q)\dot{q} \tag{7a}$$

$$M(q) = \sum_{i=1}^{n} J_{ib}^T g_i J_{ib} \tag{7b}$$

where $M(q)$ is the mass matrix of the manipulator, which is symmetric and positive definite.

III. Trajectory Optimization Method

This section formulates trajectory planning as a QP-based trajectory optimization. The physical constraints on the joint position, velocity, and acceleration are unified and described as boundary constraints, and collision avoidance is described as an inequality constraint. By reformulating the QP, the ALM is employed to solve for joint velocities (decision variables) that satisfy the constraints.

A. QP Formulation

Since QP provides the capability to consider constraints and costs, the trajectory planning of the manipulator can be formulated as a QP. Consequently, the QP is defined as follows:

$$\begin{aligned} \min_{\dot{q}\in\mathbb{R}^n} \quad & K = \frac{1}{2}\dot{q}^T \mathbf{M}(q)\dot{q} \\ \text{s.t.} \quad & \mathbf{J}(q)\dot{q} = \dot{x}_{ref} \\ & \mathbf{A}\dot{q} \le \mathbf{B} \\ & q, \dot{q}, \ddot{q} \in \Omega^* \end{aligned} \tag{8}$$

where $\ddot{q}$ represents the joint acceleration, Ω^* indicates the upper and lower bounds of the joint position, velocity, and acceleration, and the matrix $\mathbf{A}$ and vector $\mathbf{B}$ define the conditions for inequality constraints. This problem regards kinetic energy as the cost function, considers kinematics as an equality constraint, and incorporates collision avoidance as an inequality constraint, with joint physical constraints represented as boundary constraints.

B. Joint Physical Constraints

In applications, it is necessary to satisfy physical constraints such as joint position limits, velocity limits, and acceleration limits to ensure safety. Generally, the approach to controlling a robot's motion along the desired trajectory $\dot{x}_{ref}$ involves computing the inverse kinematics for each discrete time step Δt. Subsequently, the control of joint velocities occurs within the time interval $[k\Delta t,(k+1)\Delta t]$, and this is added to the feedback controller.

$$\dot{q}_{k+1} = (q_{k+1} - q_k)/\Delta t \tag{9}$$

where $\dot{q}_{k+1}$ is the desired joint velocity, and q_k is the currently measured position. However, this method requires solving the inverse kinematics at each iteration. The control strategy in this paper avoids solving the inverse kinematics by directly obtaining the optimal joint velocity $\dot{q}_{k+1}$ at the current time step through solving (8). Since the controller needs to determine the position, velocity, and acceleration at a specific time step, the joint physical constraints in (8) are defined as follows:

$$q^- \le q_{k+1} \le q^+ \tag{10a}$$
$$\dot{q}^- \le \dot{q}_{k+1} \le \dot{q}^+ \tag{10b}$$
$$\ddot{q}^- \le \ddot{q}_{k+1} \le \ddot{q}^+ \tag{10c}$$

where q^-, $\dot{q}^-$, and $\ddot{q}^-$ represent the lower bounds of the robotic arm's joint angle, velocity, and acceleration, respectively, and q^+, $\dot{q}^+$, and $\ddot{q}^+$ represent the upper bounds of the robotic arm's joint angle, velocity, and acceleration, respectively.

The proposed planner in this paper uses the currently measured joint position q_k and joint velocity $\dot{q}_k$ as inputs. The planner takes the decision variable $\dot{q}_{k+1}$ in (8) as the velocity command for the next time interval. We make a first-order approximation of these constraints, so (11) is detailed as follows:

$$q^- \le q_k + \dot{q}_{k+1}\Delta t \le q^+ \tag{11a}$$
$$\dot{q}^- \le \dot{q}_{k+1} \le \dot{q}^+ \tag{11b}$$
$$\ddot{q}^- \le \frac{\dot{q}_{k+1} - \dot{q}_k}{\Delta t} \le \ddot{q}^+ \tag{11c}$$

Since the decision variable of the optimization problem is the joint velocity $\dot{q}_{k+1}$, (11) is integrated into the following expression:

$$\Omega^- \le \dot{q}_{k+1} \le \Omega^+ \tag{12}$$

where

$$\Omega^- = \arg\max\left(\ddot{q}^-\Delta t + \dot{q}_k, (q^- - q_k)/\Delta t, \dot{q}^-\right)$$
$$\Omega^+ = \arg\min\left(\ddot{q}^+\Delta t + \dot{q}_k, (q^+ - q_k)/\Delta t, \dot{q}^+\right)$$

C. Obstacle Avoidance

Due to the road conditions and bridge structure, the mobile platform cannot strictly follow the given reference trajectory, which results in a significant error in the moving direction. The error leads to a large offset, which introduces a collision between the BIRM and bridges. It is necessary for effective collision distance detection and collision strategies to avoid collisions. In this paper, obstacles are represented as convex polytope sets O_i in the workspace, as shown in Fig.3.

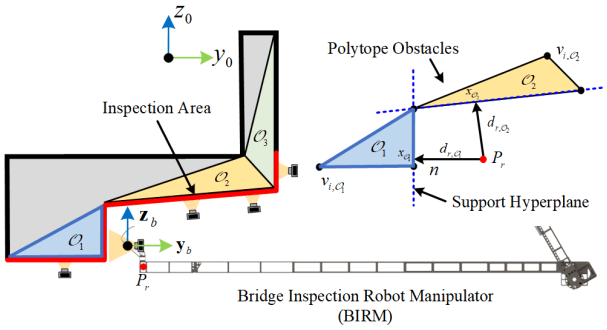

Fig. 3. Schematic diagram of collision detection

Convex polytopes described with a V-representation exhibit lower complexity under known vertices. Thus, this paper utilizes V-representation to describe the geometric shape of obstacles and defines x as a point inside the obstacle, with the specific description as follows:

$$x \in O_i = conv\{V_1, \cdots, V_m\} \tag{13}$$

where V_i is the convex hull vertex of the convex polytope determined by the geometric parameters of the bridge, the collision vector n is defined as the vector between any point $P_r \in R^d$ of the BIRM and the closest distance point x_{O_i} on the convex polytope O_i. Here, d_{r,O_i} represents the distance between them, as follows:

$$d_{r,O_i} = \left(n^T n\right)^{1/2} \tag{14}$$

and $\dot{d}_{r,O_i}$ represents their velocity.

Any collision vector is orthogonal to its supporting hyperplane, which divides the convex polytope and the point on the robot into two sides. Thus, the collision vector n can be determined to maximize the distance from the supporting hyperplane to P_r. This problem is also described as a typical QP, defined as follows:

$$\begin{aligned} \max_{n\in R^d} \quad & n^T n \\ s.t. \quad & (V_i - P_r)^T n \ge n^T n \quad ,\forall i \in \{1,\cdots,m\} \end{aligned} \tag{15}$$

Letting $e = n/n^T n$, solving the maximization problem in (15) is equivalent to solving the minimization problem in (16).

$$\begin{aligned} \min_{e\in R^d} \quad & e^T e \\ s.t. \quad & (V_i - P_r)^T e \ge 1 \quad ,\forall i \in \{1,\cdots,m\} \end{aligned} \tag{16}$$

The problem is characterized as a low-complexity QP problem. By the L-BFGS algorithm [21], the solution to this problem converges linearly, yielding the precise vector e and consequently obtaining the collision vector n in (15) and collision distance d_{r,O_i} (14).

Based on the computed collision distance, this paper adopts a velocity damper approach to prevent robot collisions, thereby constraining the joint velocity before the

manipulator contacts obstacles.

$$\dot{d}_{r,O_i} = n^T \dot{P}_{r,O_i}(t) \leq \delta\left(d_{r,O_i} - d_s\right) \tag{17}$$

where $\dot{P}_{r,O_i}(t)$ is the linear velocity vector of any point on the robot, d_s is the minimum stop distance, and δ is the positive gain associated with time-varying collision sensitivity.

Through a similar derivation with (6), we define the Jacobian matrix $J_{r,O_i}(q_{1\ldots i}) \in R^{3\times i}$ as the mapping between the velocity of any point $\dot{P}_{r,O_i}$ on link i and the joint velocities $\dot{q}_{1\ldots i} \in R^i$, expressed as:

$$\dot{P}_{r,O_i} = J_{r,O_i}\dot{q}_{1\ldots i} \tag{18}$$

Combining (17) and (18), we obtain

$$J_{O_i}\dot{q}_{1\ldots i} \leq \delta\left(d_{r,O_i} - d_s\right) \tag{19a}$$

$$J_{O_i}(q_{1\ldots i}) = n^T J_{r,O_i} \tag{19b}$$

where $J_{O_i}(q_{1\ldots i}) \in R^{1\times i}$. Therefore, the inequality constraint in (8) can be expressed as:

$$\begin{pmatrix} J_{O_1}(\hat{q}_{O_1}) & 0^{1\times(n-k_1)} \\ \vdots & \vdots \\ J_{O_j}(\hat{q}_{O_j}) & 0^{1\times(n-k_j)} \end{pmatrix} \dot{q}(t) \leq \begin{pmatrix} \delta\left(d_{r,O_1} - d_s\right) \\ \vdots \\ \delta\left(d_{r,O_j} - d_s\right) \end{pmatrix} \tag{20}$$

where the vector $\dot{q}(t) \in R^n$ represents the joint velocity, and the vector $\hat{q}_{O_j}$ represents the joint angles corresponding to j obstacles. We employ the L-BFGS algorithm [21] to obtain distance in real time. This distance detects unexpected collisions and is applied to the trajectory planner. Collision constraints are transformed into linear inequality constraints, ensuring the absolute safety of the BIRM.

D. Reformulation and Solution

The boundary constraints are integrated in Section III.B and inequality constraints in Section III.C, the QP is reformulated as

$$\begin{aligned} \min_{\dot{q}_{k+1}\in\mathbb{R}^n} \quad & \frac{1}{2}\dot{q}_{k+1}^T \mathbf{M}(q_{k+1})\dot{q}_{k+1} \\ \text{s.t.} \quad & \mathbf{J}(q_{k+1})\dot{q}_{k+1} = \dot{x}_{k+1} \\ & \mathbf{A}\dot{q}_{k+1} \leq \mathbf{B} \\ & \Omega^- \leq \dot{q}_{k+1} \leq \Omega^+ \end{aligned} \tag{21}$$

where,

$$\mathbf{A} = \begin{pmatrix} J_{O_1}(\hat{q}_{O_1}) & 0^{1\times(n-k_1)} \\ \vdots & \vdots \\ J_{O_j}(\hat{q}_{O_j}) & 0^{1\times(n-k_j)} \end{pmatrix} \in R^{j\times n} \tag{22}$$

$$\mathbf{B} = \begin{pmatrix} \delta\left(d_{r,O_1} - d_s\right) \\ \vdots \\ \delta\left(d_{r,O_j} - d_s\right) \end{pmatrix} \in R^j \tag{23}$$

$$\begin{aligned} \Omega^- &= \arg\max\left(\ddot{q}^-\Delta t + \dot{q}_k, (q^- - q_k)/\Delta t, \dot{q}^-\right) \\ \Omega^+ &= \arg\min\left(\ddot{q}^+\Delta t + \dot{q}_k, (q^+ - q_k)/\Delta t, \dot{q}^+\right) \end{aligned} \tag{24}$$

Using the PHR-ALM [20] method, we solve optimization problems with equality and inequality constraints. The main idea of this method is to transform the optimization problem with constraints into an unconstrained optimization problem. For the problem defined in (21), the PHR augmented Lagrangian function is defined as

$$\begin{aligned} L_\rho(\dot{q}_{k+1}, \lambda, \mu) = & \tfrac{1}{2}\dot{q}_{k+1}^T \mathbf{M}(q_{k+1})\dot{q}_{k+1} \\ & + \frac{\rho}{2}\left\| J(q_{k+1})\dot{q}_{k+1} - \dot{x}_{k+1} + \frac{\lambda}{\rho}\right\|^2 \\ & + \frac{\rho}{2}\left\| \max\left[\mathbf{A}\dot{q}_{k+1} - \mathbf{B} + \frac{\mu_1}{\rho}, 0\right]\right\|^2 \\ & + \frac{\rho}{2}\left\| \max\left(\dot{q}_{k+1} - \Omega^+ + \frac{\mu_2}{\rho}, 0\right)\right\|^2 \\ & + \frac{\rho}{2}\left\| \max\left(\Omega^- - \dot{q}_{k+1} + \frac{\mu_3}{\rho}, 0\right)\right\|^2 \end{aligned} \tag{25}$$

where $\|\cdot\|^2$ represents the square of the 2-norm, $\rho > 0$ is the penalty parameter, $\mu \geq 0$ is the dual variable for the equality constraints, and λ is the dual variable for inequality constraints.

We employ the L-BFGS [21] to solve the unconstrained optimization problem in (25) by continuously updating the dual variables to obtain the optimal joint velocities. We set the initial values of λ, μ, and ρ as $\lambda = \mu = 0, \rho = 1$. However, the convergence rate is affected by the initial values of the joint velocities in each iteration. To solve this problem, we use the first-order approximation of joint positions as the initial point.

IV. Proposed Discrete-Time Planner

Using the QP described in (21), this section proposes a position-servo-based discrete-time planner for achieving minimal kinetic energy control. The diagram of the planner is illustrated in Fig.4. This method discretizes the Cartesian space trajectory, iteratively optimizes the cost function of the manipulator kinetic energy, and calculates the distance between the robotic arm and obstacles in each iteration. The planner comprehensively considers joint physical constraints and obstacle avoidance. The planner determines the optimal joint velocities in discrete time, adapting to current conditions based on updated information.

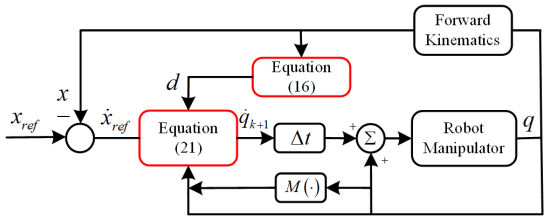

Fig. 4. Diagram of the position-servo-based discrete-time planner

The expected path x_{ref} is used as the planner's input, as shown in Fig.4. Starting from the starting point x_0, the optimal joint velocity is determined through iterative solving until reaching the endpoint x_f, as shown in Fig.5. This process realizes the motion of the end effector along the desired path from x_0 to x_f, while meeting joint physical constraints and avoiding collisions with the bridge structure. Path points P1, P2, P3, and P4 are determined by the structural parameters of the bridge.

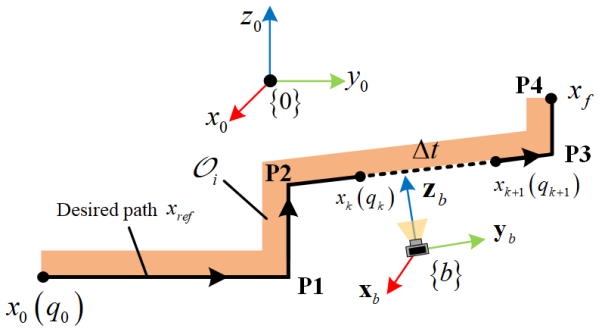

Fig. 5. Schematic diagram of end-effector trajectory planning for the BIRM. A discretized end-effector trajectory is parameterized.

V. Experiments and Results

To validate the feasibility and effectiveness, the proposed method was implemented on a 5-DOF bridge inspection robot manipulator. The model parameters for both the bridge and the manipulator are well known and employed in both simulation and physical experiments. In this paper, robot kinematics based on the product of exponentials (PoE) formula are used, and the screw axis parameters are shown in Table I. These parameters include vital structural parameters of the manipulator. This formula does not require specifying coordinate systems for each link; only joint vectors need to be specified, making it easy to derive the Jacobian matrix. Furthermore, the parameters associated with the kinetic energy of the BIRM are provided in Table II.

TABLE I

The screw axes of the BIRM based on the PoE.

i	ω_i	v_i
1	(0,0,1)	(0,0,0)
2	(1,0,0)	(0,L_1, $-L_2$)
3	(0,0,0)	(0,0,-1)
4	(1,0,0)	(0,L_1-L_4,$-L_2-L_3$)
5	(1,0,0)	(0,L_1-L_4+0.327,$-L_2-L_3+L_5$)

[1] L_1, L_2, L_3, L_4, and L_5 are 0.614, 2.4585, 0.173, 4.03, and 4.52885, respectively, with units in meters.

TABLE II

Parameters associated with the kinetic energy of the BIRM

Link i	Mass(Kg)	Moment of Interia (I_{xx}, I_{yy}, I_{xx},I_{xy}, I_{xz}, I_{yz}) ($Kg \cdot m^2$)
1	59.65	(72.9794, 73.7094, 1.9668, -0.1382, -2.6269, 2.3639)
2	30.30	(34.0063, 1.0416, 34.6787, 0.1824, -0.0048, 0.0621)
3	23.62	(52.9395, 0.4909, 53.2322, 0.5128, -0.0301, -0.6393)
4	6.28	(16.6706, 16.6349, 0.0833, 0.0001, -0.0018, -0.4930)
5	0.75	(0.000768, 0.002554, 0.0023, -0.000017, 0.00005, 0)

A. Simulated Experiments and Results

In this section, we present comparative results for planning with three methods: Our method, Method (1) [9], and Method (2) [19]. All three methods use kinetic energy as the optimization criterion. The simulation experiments were performed on an Intel® Core™ i7-1165G7 CPU@4.7 GHz PC with 16 GB of RAM. Method (1) [9] considered only the joint position and velocity without acceleration or collision avoidance. Method (2) [19] considered the joint position, velocity, and acceleration without collision avoidance. Our method considered the joint position, velocity, and acceleration and collision avoidance. This paper provides the necessary joint acceleration limits for analyzing the comparative results. The range of the acceleration is from -0.25 to 0.25, with unit deg$/s^2$.

The simulation environments are illustrated in Fig.6. The bridge is modeled offline as a set of convex polyhedron obstacles. The structure of the bridge determines the parameters of the convex polyhedron. We use a specific type of bridge structure as an example. We input the trajectory x_{ref} of the task space and the discrete-time Δt. The proposed planner controls the manipulator to move along the trajectory. The joint data for the three methods were recorded in Fig.7. Due to the absence of obstacle constraints in Method (1) and Method (2), there were failures in the planning results. We only recorded one set of successful data.

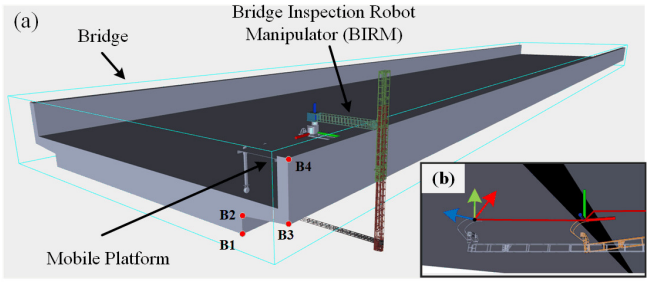

Fig. 6. (a) Environment setup. (b) shows the simulation results, with the red lines indicating the end-effector's following trajectory. B1, B2, B3, and B4 represent key structure parameters of Bridge-I.

According to Fig.7, the planning time of our proposed method is significantly better than that of the other methods. Additionally, the joint positions and velocities of the three methods are within the limitation range. However, the acceleration of joint 2 in Method (1) and Method (2) exceeds this limitation. Moreover, the motion range of joints 3, 4, and 5 is significantly greater than the other joints, while the inertial masses of joints 3, 4, and 5 are significantly smaller than those of the other joints. Therefore, we argue that the main advantage of minimal kinetic energy is that it tends to move joints with lower inertia, reducing energy consumption, rather than joints with higher inertia.

The above results indicated that the proposed planner can satisfy equality, inequality, and boundary constraints and can effectively meet acceleration constraints in the velocity-level planning method.

B. Physical Experiments and Results

We conducted physical experiments on two different types of bridges, as shown in Fig.1. The proposed planner is deployed in the EtherCAT bus motion controller, which

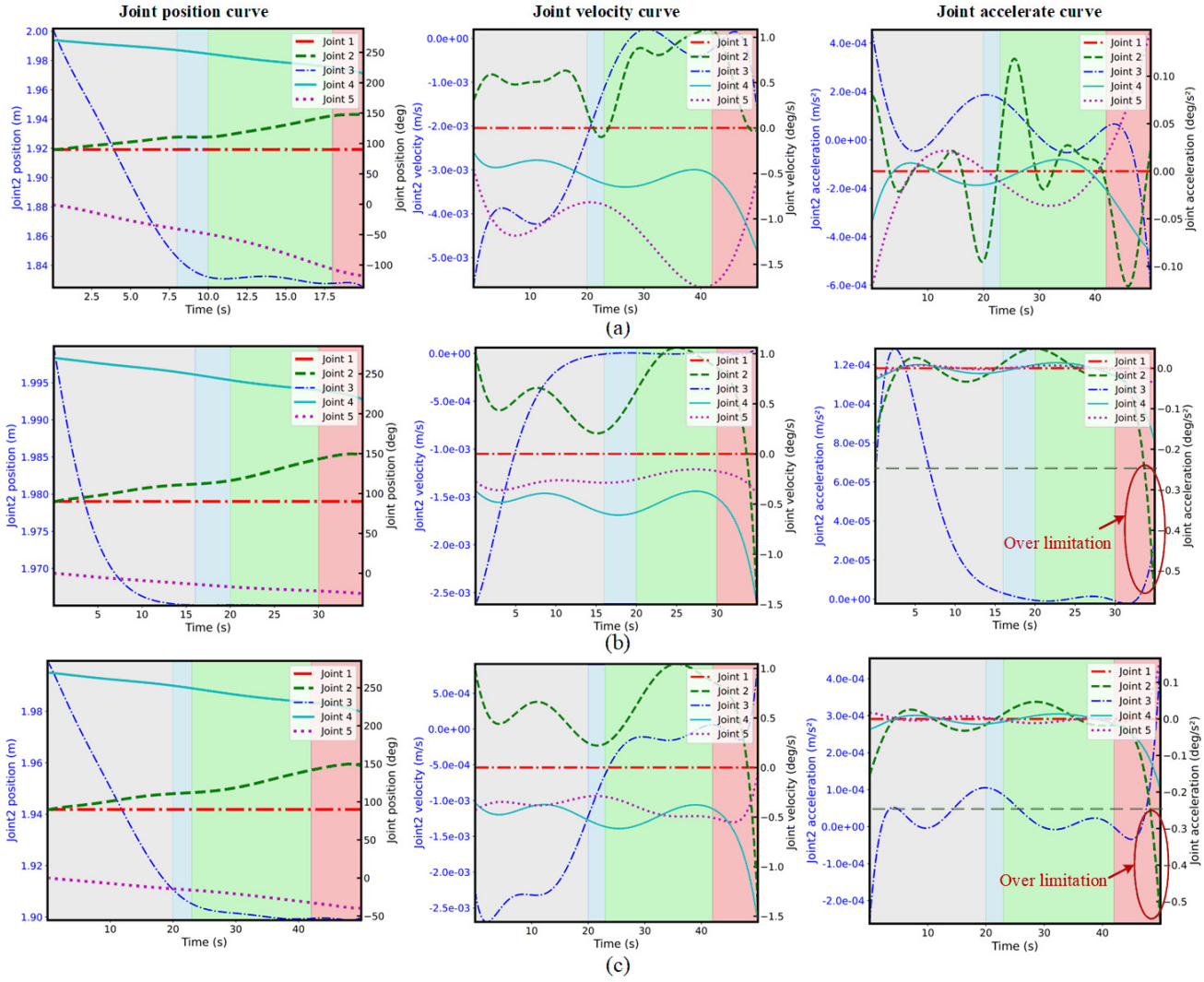

Fig. 7. Simulation experiments results of joint position, velocity, and acceleration for three methods: (a) Ours, (b) Method (1), and (c) Method (2). The trajectory was divided into four phases, with the waypoints P1(0, -0.3815, -2.843), P2(0, -0.3815, -2.2353), P3(0, 1.5185, -2.043), and P4(0, 1.51850, -1.343).

simultaneously records joint data from joint sensors to calculate kinetic energy. Similarly, we specify the joint position, velocity, and acceleration limits, and the structural parameters of the bridge are known. We only need several key points of the bridge and then use the method in Section 3. C for environment modeling. The information of the model was input into the controller. Additionally, the initial configuration of the manipulator relative to the bridge is not fixed, which does not affect the manipulator's movement along the reference trajectory. Through analysis, the data results for bridge-I are consistent with those for bridge-II, so this section presents and analyses the data for bridge-I, as shown in Fig.8.

To compare the results of various algorithms, the horizontal axis in Fig.8-b and (c) shows the number of discrete points in the end-effector trajectory. Here, each discrete point corresponds to a joint angle vector. As shown in Fig.8-b, the results indicate that the kinetic energy of our method is significantly better than that of the other methods. To verify the stability of trajectory planning, the end-effector tracking errors are recorded, as shown in Figure 8-c. The error represents the deviation between the planned end-effector trajectory and the actual trajectory. The end-effector error of Method (1) is larger than that of Method (2) and our method. Method (1) did not consider acceleration limits, which results in excessive acceleration within the given time, thus generating significant displacement of the end effector. The results indicate that our method demonstrates satisfactory performance in bridge inspection.

VI. Conclusion

In this paper, we proposed kinetic-energy-optimal and safety-guaranteed trajectory planning for the bridge inspection robot manipulator. The method simultaneously addressed energy consumption and safety concerns and ensured joint position, velocity, and acceleration limits while

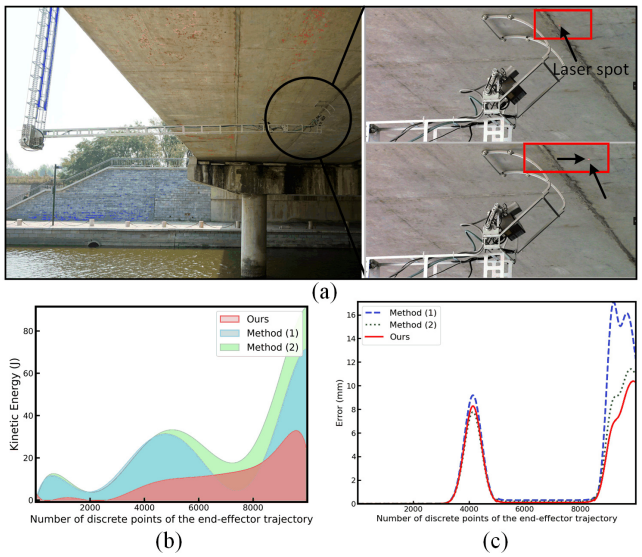

Fig. 8. Physical experiments and results for Bridge-I. (a) Real-world experiments. (b) Kinetic energy. (c) End-effector tracking error.

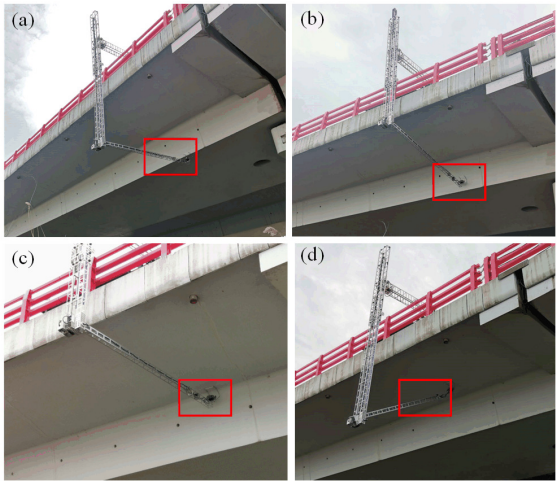

Fig. 9. Physical experiment for Bridge-II. (a)-(d) shows the process of the robot's motion.

considering collision avoidance. Our method effectively addresses the challenges of acceleration constraints and collision avoidance in a velocity-level planner. We reduced the 3D Euclidean space to 2D to optimize the collision checking efficiency, thereby enhancing safety. The experimental results demonstrated that our method can generate joint trajectories with low energy consumption and high safety. Experiments on two types of bridges validated the sustainability and safety of bridge surfaces in the field. However, the tracking accuracy of the end-effector is greatly influenced by the arm's structure. Therefore, we plan to consider structural stiffness to enhance tracking accuracy and deploy an autonomous mobile platform for complete autonomous inspection.

REFERENCES

[1] T. Ikeda, S. Minamiyama, S. Yasui, K. Ohara, A. Ichikawa, S. Ashizawa, A. Okino, T. Oomichi, and T. Fukuda, "Stable camera position control of unmanned aerial vehicle with three-degree-of-freedom manipulator for visual test of bridge inspection," *Journal of Field Robotics*, vol. 36, no. 7, pp. 1212–1221, 2019.

[2] J.-K. Oh, A.-Y. Lee, S. M. Oh, Y. Choi, B.-J. Yi, and H. W. Yang, "Design and control of bridge inspection robot system," in *2007 International Conference on Mechatronics and Automation*. IEEE, 2007, pp. 3634–3639.

[3] J.-K. Oh, G. Jang, S. Oh, J. H. Lee, B.-J. Yi, Y. S. Moon, J. S. Lee, and Y. Choi, "Bridge inspection robot system with machine vision," *Automation in Construction*, vol. 18, no. 7, pp. 929–941, 2009.

[4] R. Xie, J. Yao, K. Liu, X. Lu, Y. Liu, M. Xia, and Q. Zeng, "Automatic multi-image stitching for concrete bridge inspection by combining point and line features," *Automation in Construction*, vol. 90, pp. 265–280, 2018.

[5] H. Zhang, J. Tan, L. Liu, Q. J. Wu, Y. Wang, and L. Jie, "Automatic crack inspection for concrete bridge bottom surfaces based on machine vision," in *2017 Chinese Automation Congress (CAC)*. IEEE, 2017, pp. 4938–4943.

[6] M. Lorenz, J. Paris, F. Schöler, J.-P. Barreto, T. Mannheim, M. Hüsing, and B. Corves, "Energy-efficient trajectory planning for robot manipulators," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, vol. 58189. American Society of Mechanical Engineers, 2017, p. V05BT08A067.

[7] C. R. Garrett, R. Chitnis, R. Holladay, B. Kim, T. Silver, L. P. Kaelbling, and T. Lozano-Pérez, "Integrated task and motion planning," *Annual review of control, robotics, and autonomous systems*, vol. 4, pp. 265–293, 2021.

[8] T. F. Chan and R. V. Dubey, "A weighted least-norm solution based scheme for avoiding joint limits for redundant joint manipulators," *IEEE transactions on Robotics and Automation*, vol. 11, no. 2, pp. 286–292, 1995.

[9] D. P. Pagnotta, A. Monteriù, A. Freddi, S. Longhi, and A. Maciejewski, "Redundancy resolution scheme for manipulators subject to inequality constraints," *International Journal of Control, Automation and Systems*, vol. 21, no. 2, pp. 575–590, 2023.

[10] Z. Xu, S. Li, X. Zhou, S. Zhou, T. Cheng, and Y. Guan, "Dynamic neural networks for motion-force control of redundant manipulators: An optimization perspective," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 2, pp. 1525–1536, 2020.

[11] S. Li, M. Zhou, and X. Luo, "Modified primal-dual neural networks for motion control of redundant manipulators with dynamic rejection of harmonic noises," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 10, pp. 4791–4801, 2017.

[12] D. P. Bertsekas, *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.

[13] Z. Pan, M. Liu, X. Gao, and D. Manocha, "Joint search of optimal topology and trajectory for planar linkages," *The International Journal of Robotics Research*, vol. 42, no. 4-5, pp. 176–195, 2023.

[14] N. Jorge and J. W. Stephen, *Numerical optimization*. Spinger, 2006.

[15] J. Gregory, A. Olivares, and E. Staffetti, "Energy-optimal trajectory planning for robot manipulators with holonomic constraints," *Systems & Control Letters*, vol. 61, no. 2, pp. 279–291, 2012.

[16] S. Li, Y. Zhang, and L. Jin, "Kinematic control of redundant manipulators using neural networks," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2243–2254, 2016.

[17] Z. Zhang and Y. Zhang, "Variable joint-velocity limits of redundant robot manipulators handled by quadratic programming," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 2, pp. 674–686, 2012.

[18] D. Guo and Y. Zhang, "Acceleration-level inequality-based man scheme for obstacle avoidance of redundant robot manipulators," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 12, pp. 6903–6914, 2014.

[19] Y. Zhang, S. Li, J. Gui, and X. Luo, "Velocity-level control with compliance to acceleration-level constraints: A novel scheme for manipulator redundancy resolution," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 3, pp. 921–930, 2017.

[20] R. T. Rockafellar, "Augmented lagrange multiplier functions and duality in nonconvex programming," *SIAM Journal on Control*, vol. 12, no. 2, pp. 268–285, 1974.

[21] D. C. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," *Mathematical programming*, vol. 45, no. 1-3, pp. 503–528, 1989.