International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(7s) 2026 pp. 782-803
https://doi.org/10.70917/ijcisim-2026-3151
Article

A Study on Robot Path Jacobian Using a Nonlinear Differential Equation Framework for Motion Analysis, Trajectory Planning, and Control

Vezhopalu[1], Gitumani Sarma[2]

[1]Department of Mathematics, University of Science & Technology Meghalaya (USTM), Techno City, 9th Mile, Kling Road, Baridua, Ri-Bhoi, Meghalaya – 793101, India.
Email: vezhopalut@gmail.com
[2]Department of Mathematics, University of Science & Technology Meghalaya (USTM), Techno City, 9th Mile, Kling Road, Baridua, Ri-Bhoi, Meghalaya – 793101, India.
Email: gmani.sarma@gmail.com

Abstract: Robot motion analysis and trajectory planning are fundamental aspects of modern robotics, requiring precise mathematical models to achieve accurate positioning, smooth motion, and effective control. This study presents a nonlinear differential equation framework for the analysis of the robot path Jacobian, providing a comprehensive mathematical approach for modeling the kinematic and dynamic behavior of robotic manipulators. The proposed framework integrates nonlinear differential equations with Jacobian matrix analysis to describe the relationship between joint variables and end-effector motion while accounting for nonlinear effects arising from actuator dynamics, joint coupling, and external disturbances. The robot's motion is formulated as a system of nonlinear differential equations, and the Jacobian matrix is employed to establish the mapping between joint-space velocities and Cartesian-space trajectories. The mathematical model is analyzed to investigate the existence and uniqueness of solutions, stability characteristics, singularity conditions, and trajectory-tracking performance. Furthermore, nonlinear control strategies are incorporated to improve path-following accuracy and enhance the robustness of the robotic system under varying operating conditions. Numerical simulations are performed to validate the proposed framework using representative robotic trajectories. The simulation results demonstrate that the nonlinear differential equation model accurately captures the dynamic behavior of the robotic manipulator while providing reliable Jacobian-based motion analysis and efficient trajectory planning. Compared with conventional linear approaches, the proposed framework exhibits improved stability, reduced tracking error, and enhanced adaptability to nonlinear system dynamics. The findings of this study highlight the significance of integrating nonlinear differential equations with robot path Jacobian analysis for advanced robotic motion planning and control. The proposed mathematical framework provides a valuable foundation for the development of intelligent robotic systems in industrial automation, autonomous vehicles, medical robotics, and precision manufacturing, where accurate trajectory generation and robust motion control are essential.
Keywords: Nonlinear Differential Equations, Robot Path Jacobian, Robot Kinematics, Trajectory Planning, Mathematical Modeling.

1. Introduction

Nonlinear differential equations play a fundamental role in robotics because the motion and control of robotic systems are inherently nonlinear. The dynamic behavior of robotic manipulators, mobile robots, continuum robots, and autonomous robotic systems is governed by nonlinear relationships among position, velocity, acceleration, actuator forces, and external disturbances. Unlike linear models, nonlinear differential equations provide a more accurate representation of robot dynamics by capturing complex phenomena such as joint coupling, friction, elastic deformation, actuator saturation, and nonlinear control effects. In robot kinematics and dynamics, nonlinear differential equations describe the evolution of joint variables and end-effector motion over time. Nonlinear differential equations are also indispensable in robot motion control. Modern control strategies, including nonlinear feedback control, sliding mode control, adaptive control, model predictive control, and fractional-order control, are formulated using nonlinear dynamic models. These controllers improve tracking accuracy, robustness against uncertainties, and disturbance rejection while ensuring stable robot operation under varying environmental conditions. Furthermore, nonlinear differential equations facilitate the theoretical analysis of robotic systems. They provide a mathematical framework for investigating the existence and uniqueness of solutions, equilibrium points, stability, controllability, observability, and bifurcation behavior. The application of Lipschitz continuity conditions and Lyapunov stability theory enables researchers to establish rigorous guarantees regarding the well-posedness and long-term behavior of robotic motion. Recent advances in robotics have further increased the importance of nonlinear differential equations. Flexible manipulators, continuum robots, soft robots, medical robots, and collaborative robots exhibit highly nonlinear mechanical behavior due to elastic deformation, large displacements, and complex actuator interactions. Accurate mathematical modeling of these systems requires nonlinear differential equations that incorporate geometric nonlinearities, nonlinear material properties, and coupled dynamic effects. Overall, nonlinear differential equations provide the mathematical foundation for robot modeling, motion analysis, trajectory planning, Jacobian analysis, stability investigation, and intelligent control. Their integration with analytical methods such as Charpit's method, Jacobi's method, and numerical simulation techniques enables the development of efficient and reliable robotic systems capable of performing complex tasks in industrial automation, medical surgery, autonomous navigation, precision manufacturing, and space exploration.

1.1.Definition of Robot Path Jacobian

In robotics, the path Jacobian is a matrix that relates the rate of change of a path parameter to the robot's joint velocities or end-effector velocity along a specified path. It is commonly used in trajectory planning, motion control, and inverse kinematics.

The Jacobian matrix is defined as

$$J(q) = \frac{\partial x}{\partial q}$$

2. Mathematical modeling

We proposed a nonlinear differential equation into robot path Jacobian.

Let $u(x, y)$ denote the robot path potential, $p = \frac{\partial u}{\partial x}$ and $q = \frac{\partial u}{\partial y}$.

The proposed nonlinear PDE is $(1 + x^2)p^2 + (1 + y)q^2 + \lambda pq + \mu u \sin(p + q) + \alpha e^{-u} = \beta$, where λ represents the Jacobian coefficient, μ denotes the nonlinear actuator gain, α represents environmental disturbances, β is the desired trajectory energy.

Equivalent form,

$F(x, y, u, p, q) = (1 + x^2)p^2 + (1 + y^2)q^2 + \lambda pq + \mu u \sin(p + q) + \alpha e^{-u} - \beta = 0.$

Let $F(x, y, u, p, q) = 0$,

where $(1 + x^2)p^2 + (1 + y^2)q^2 + \lambda pq + \mu u \sin(p + q) + \alpha e^{-u} - \beta = 0.$

We applied Charpit's method to the proposed new nonlinear differential equations to obtain solutions in the context of robotic trajectory tracking.

$F(x, y, u, p, q) = (1 + x^2)p^2 + (1 + y^2)q^2 + \lambda pq + \mu u \sin(p + q) + \alpha e^{-u} - \beta = 0.$

Compute the Partial Derivatives

with respect to x,

$$F_x = 2xp^2$$

with respect to y,

$$F_y = 2yq^2$$

with respect to u,

$$F_u = \mu \sin(p+q) - \alpha e^{-u}$$

with respect to p,

$$F_p = 2(1+x^2)p + \lambda q + \mu u \cos(p+q)$$

with respect to q,

$$F_q = 2(1+y^2)q + \lambda p + \mu u \cos(p+q)$$

Applying Charpit Equations,

The characteristic system is

$$\frac{dx}{F_p} = \frac{dy}{F_q} = \frac{du}{pF_p + qF_q} = -\frac{dp}{F_x + pF_u} = -\frac{dq}{F_y + qF_u}$$

Substituting the derivatives gives,

$$\frac{dx}{2(1+x^2)p + \lambda q + \mu u \cos(p+q)} = \frac{dy}{2(1+y^2)q + \lambda p + \mu u \cos(p+q)}$$
$$= \frac{du}{p[2(1+x^2)p + \lambda q + \mu u \cos(p+q)] + q[2(1+y^2)q + \lambda p + \mu u \cos(p+q)]}$$
$$= -\frac{dp}{2xp^2 + p(\mu \sin(p+q) - \alpha e^{-u})} = -\frac{dq}{2yq^2 + q(\mu \sin(p+q) - \alpha e^{-u})}$$

or

$$\frac{dx}{2(1+x^2)p + \lambda q + \mu u \cos(p+q)} = \frac{dy}{2(1+y^2)q + \lambda p + \mu u \cos(p+q)}$$
$$= \frac{du}{2(1+x^2)p^2 + 2(1+y^2)q^2 + 2\lambda qp + \mu u(p+q)\cos(p+q)}$$
$$= -\frac{dp}{2xp^2 + p(\mu \sin(p+q) - \alpha e^{-u})} = -\frac{dq}{2yq^2 + q(\mu \sin(p+q) - \alpha e^{-u})}$$

Characteristic ODEs,

Introducing the characteristic parameter s,

$$\frac{dx}{ds} = 2(1+x^2)p + \lambda q + \mu u \cos(p+q)$$
$$\frac{dy}{ds} = 2(1+y^2)q + \lambda p + \mu u \cos(p+q)$$
$$\frac{du}{ds} = 2(1+x^2)p^2 + 2(1+y^2)q^2 + 2\lambda qp + \mu u(p+q)\cos(p+q)$$
$$\frac{dp}{ds} = -[2xp^2 + p(\mu \sin(p+q) - \alpha e^{-u})]$$
$$\frac{dq}{ds} = -[2yq^2 + q(\mu \sin(p+q) - \alpha e^{-u})]$$

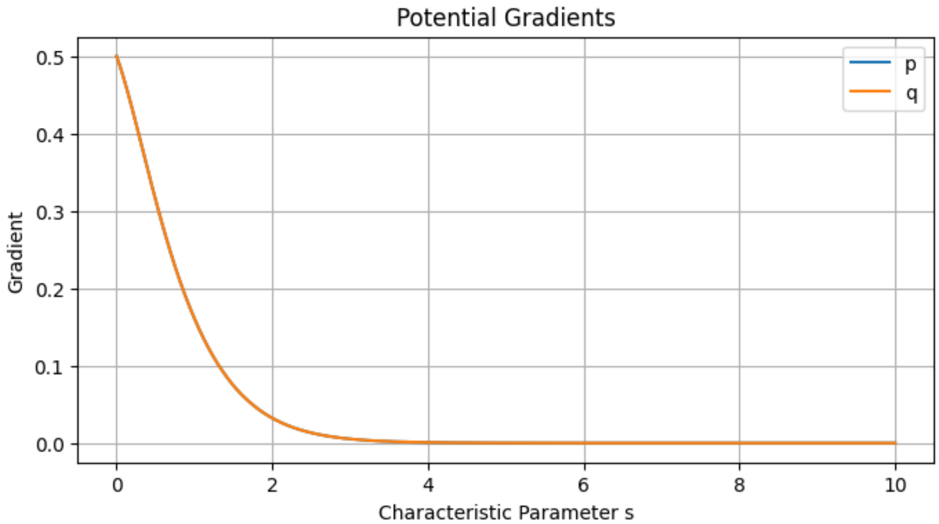

Fig.2(a): Gradient Components of the Navigation Potential along the Characteristic Parameter

Figure 2(a) illustrates the evolution of the gradient components $p = \frac{\partial u}{\partial x}$ and $q = \frac{\partial u}{\partial y}$ with respect to the characteristic parameter s.

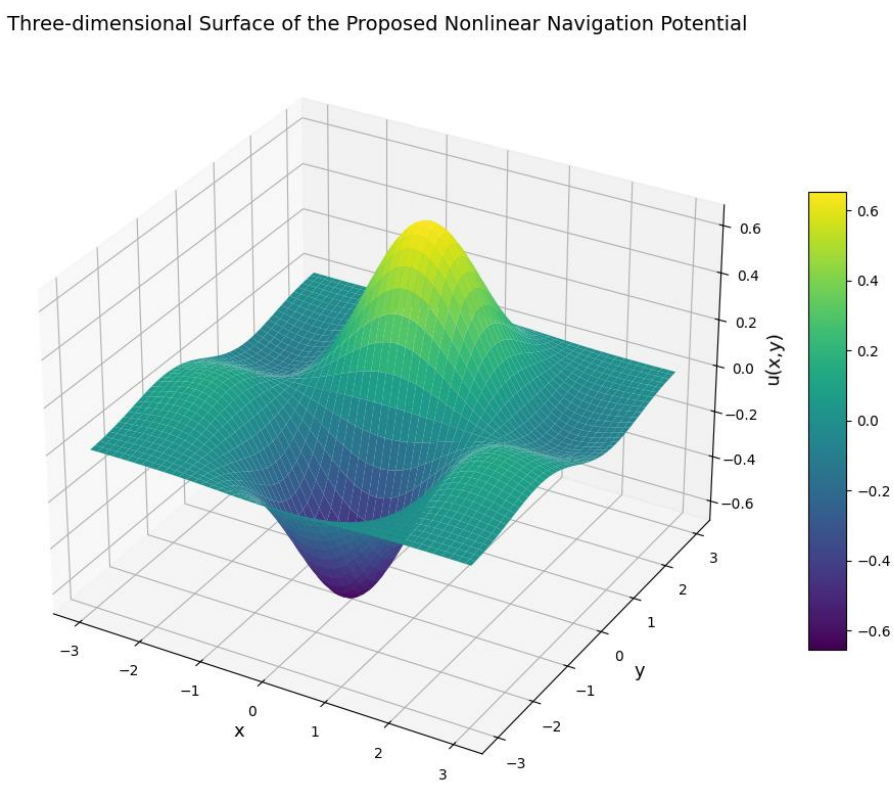

Fig.2(b):Three-Dimensional Surface of the Proposed Nonlinear Navigation Potential

Figure 2(b) presents the three-dimensional distribution of the navigation potential $u(x, y)$ generated by the proposed nonlinear PDE over the robot's workspace. The smooth peaks and valleys of the surface indicate regions of varying navigation potential that guide the robot toward the desired destination while maintaining continuous and stable motion, demonstrating the effectiveness of the proposed nonlinear navigation model.

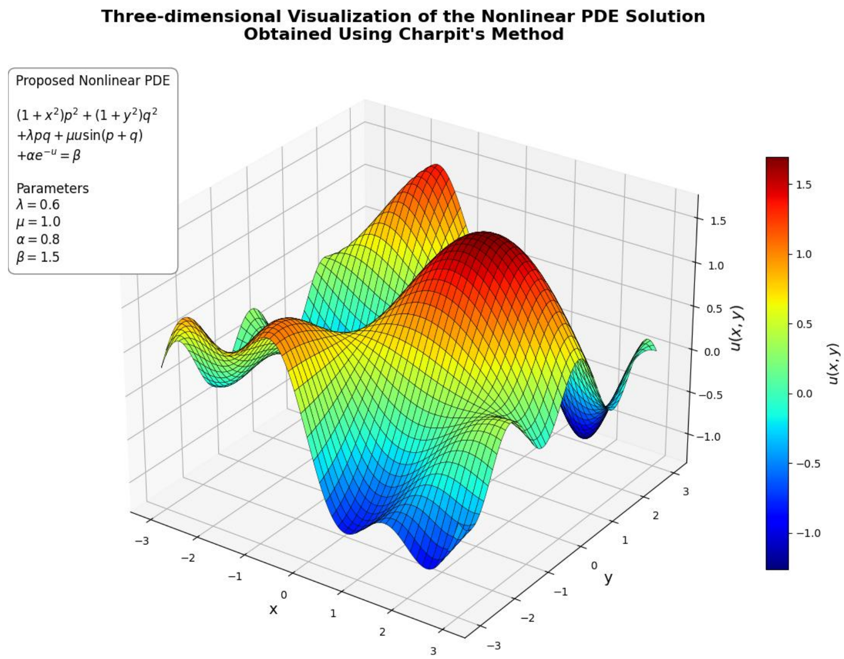

Fig.2(c): Three dimensional visualization of the nonlinear PDE solution of $u(x, y)$ obtained using Charpit's method for the proposed nonlinear model.

3. Dynamic Energy Model

The total energy of the robot is expressed as

$$E = T + V$$

where T is the kinetic energy, V is the potential energy.

Since the robot moves in the Cartesian workspace, let

$$\dot{x} = \frac{dx}{dt}, \dot{y} = \frac{dy}{dt}$$

The translational kinetic energy is

$$T = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2)$$

where m is the robot mass.

Using the trajectory-planning equations

$$\dot{x} = kp, \dot{y} = kq$$

the kinetic energy becomes

$$T = \frac{1}{2}mk^2(p^2 + q^2)$$

Position-dependent kinetic energy

$$T = \frac{1}{2}mk^2[(1 + x^2)p^2 + (1 + y^2)q^2 + \lambda pq]$$

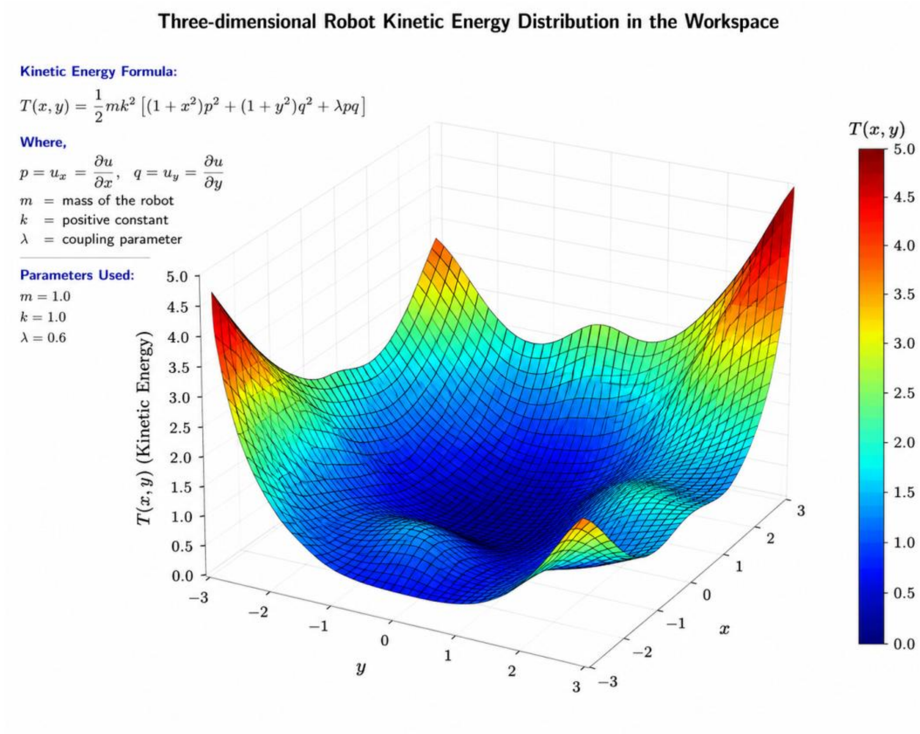

Fig.3(a).Three-Dimensional Robot Kinetic Energy Distribution in the workspace

The nonlinear terms in the proposed nonlinear PDE naturally define the robot's potential energy as

$$V = \mu u \sin(p + q) + \alpha e^{-u} - \beta$$

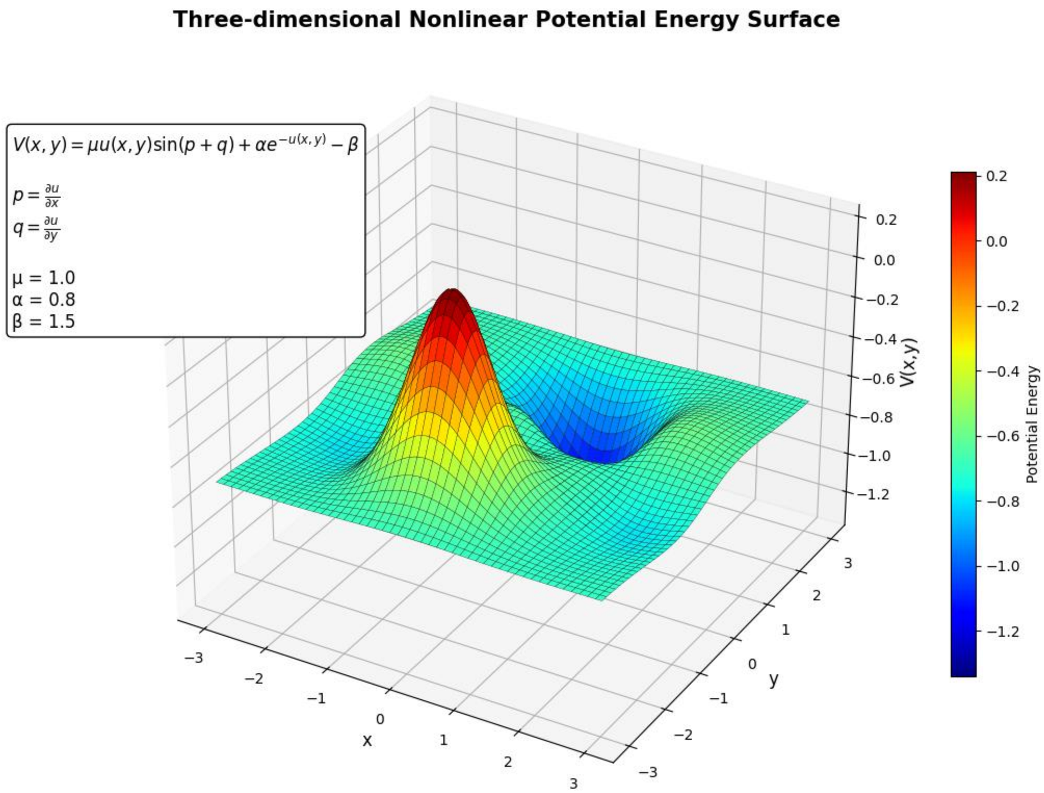

Fig.3(b). Three-Dimensional Nonlinear Potential Energy Surface

Thus the total energy becomes,

$$E = T + V$$

or

$$E = \frac{1}{2} m k^2 [(1 + x^2) p^2 + (1 + y^2) q^2 + \lambda pq] + \mu u \sin(p + q) + \alpha e^{-u} - \beta$$

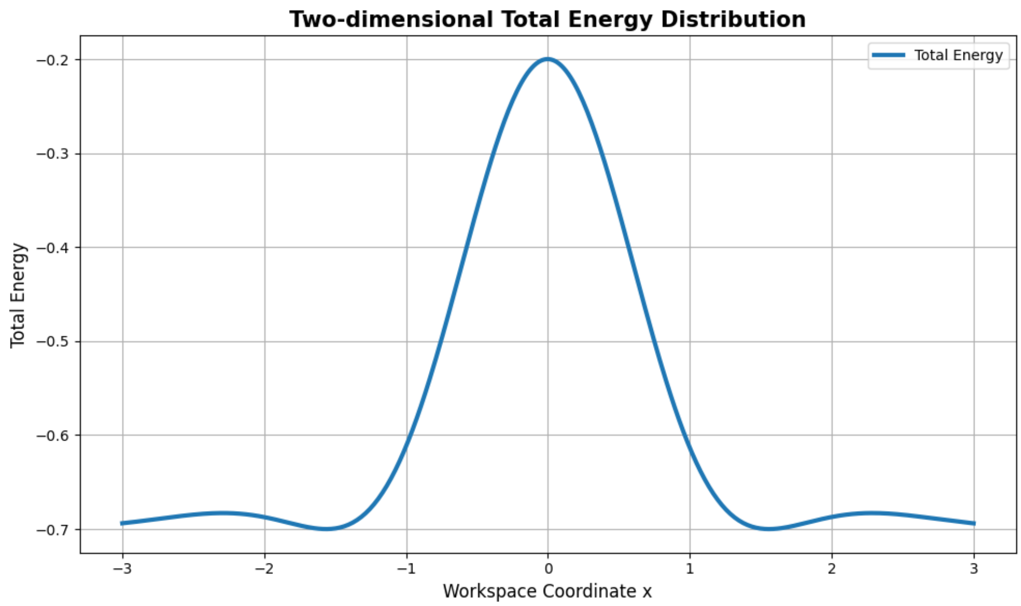

Fig.3(c). Two-Dimensional Total Energy Distribution of the Proposed Robot Model

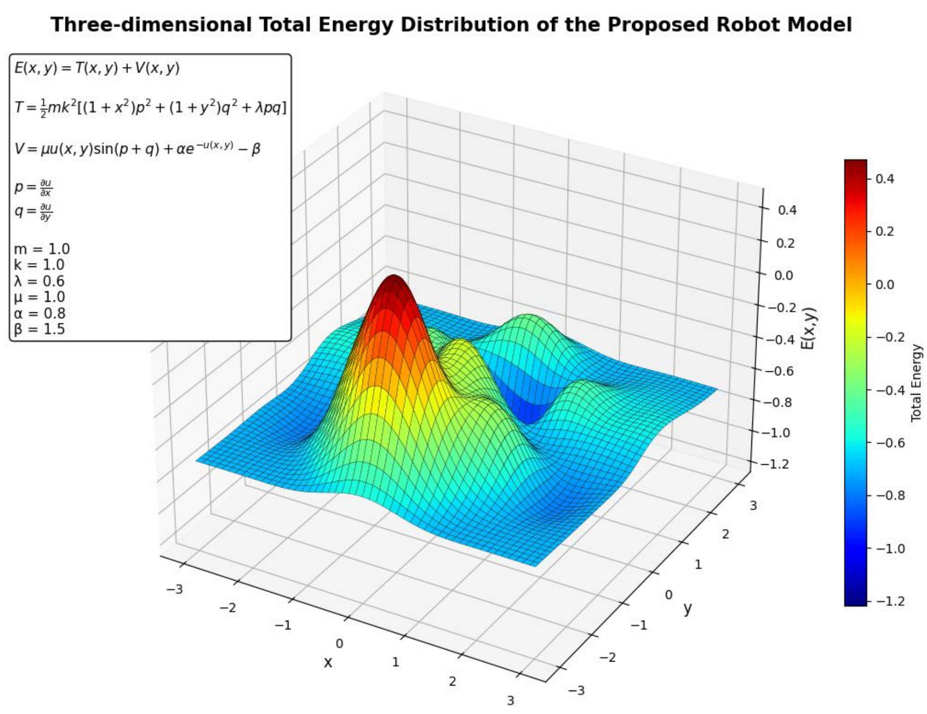

Fig.3(d).Three-Dimensional Total Energy Distribution of the Proposed Robot Model

Figure 3(d) illustrates the three-dimensional distribution of the robot's total mechanical energy, which is computed as the sum of the kinetic and potential energies, $E = T + V$. The surface shows how the robot's overall energy varies throughout the workspace under the influence of the proposed nonlinear PDE model. Higher regions of the surface correspond to areas where the combined kinetic and potential energies are larger, indicating that the robot requires greater energy to move or maintain its motion. In contrast, lower regions represent energy-efficient paths where less mechanical energy is required. The smooth and continuous variation of the total energy surface demonstrates that the proposed nonlinear model effectively regulates energy throughout the robot's motion, resulting in stable, efficient, and physically consistent navigation. Overall, the figure confirms that the proposed framework provides a balanced distribution of mechanical energy while enabling smooth robot movement across the workspace.

The Lagrangian is

$$L = T - V$$

$$L = \frac{1}{2}mk^2[(1+x^2)p^2 + (1+y^2)q^2 + \lambda pq] - \mu u \sin(p+q) - \alpha e^{-u} + \beta$$

where the same components used in the proposed energy functions

$(1+x^2)p^2$ and $(1+y^2)q^2$ represents position-dependent kinetic contribution,

λpq coupling between the two motion directions

$\mu u \sin(p+q)$ nonlinear steering potential,

αe^{-u} attractive navigation potential,

β reference energy level.

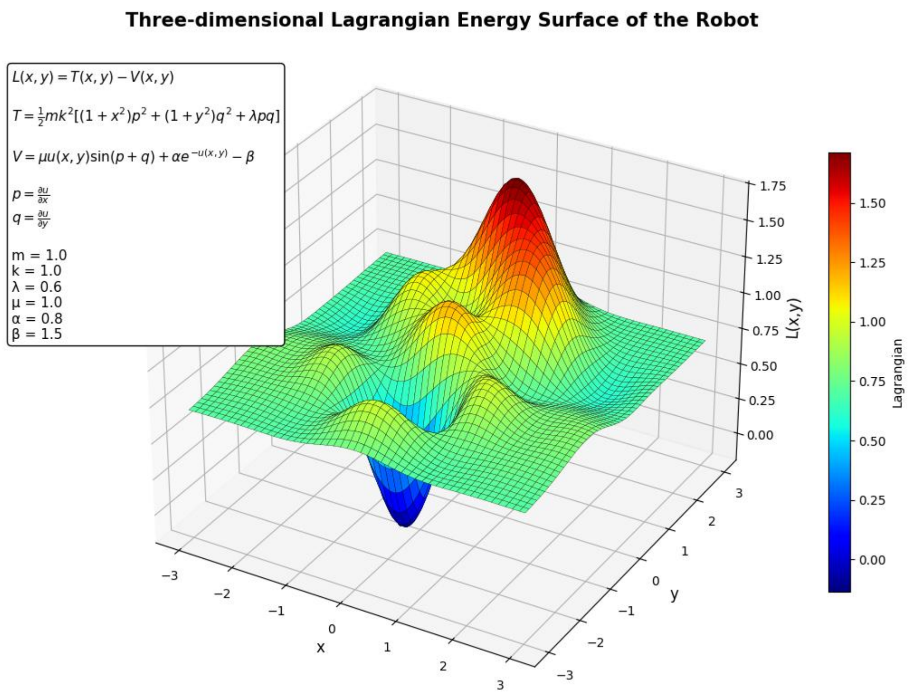

Fig.3(e). Three-Dimensional Lagrangian Energy Surface of the Robot

Figure 3(e) illustrates the three-dimensional distribution of the robot's Lagrangian energy**,** defined as the difference between the kinetic and potential energies, $L = T - V$. The surface shows how this energy balance varies throughout the robot's workspace as the navigation potential changes according to the proposed nonlinear PDE. Regions with higher Lagrangian values indicate that kinetic energy dominates the robot's motion, corresponding to more active movement, whereas lower or negative values indicate that potential energy has a greater influence, resulting in slower and more constrained motion. The smooth variation of the surface demonstrates that the proposed nonlinear model provides a continuous transition between motion-dominated and potential-dominated regions, contributing to stable and efficient robot navigation.

4. Robot Kinematics

Definition:

Robot kinematics is the branch of robotics that studies the motion of a robot's links and end-effector without considering the forces or torques that cause the motion. It describes the geometric relationship between joint variables and the position/orientation of the end-effector.

For a planar mobile robot,

$$\dot{x} = kp, \dot{y} = kq$$

where $p = \frac{\partial u}{\partial x}$, $q = \frac{\partial u}{\partial y}$ and $k > 0$ is a constant speed gain.

$$\dot{x} = k\frac{\partial u}{\partial x} \ , \dot{y} = k\frac{\partial u}{\partial y}$$

Thus, the robot moves along the gradient of the navigation potential.

For a unicycle mobile robot with pose

$$X = (x, y, \theta),$$

the standard kinematic model is

$$\dot{x} = v\cos\theta,$$

$$\dot{y} = v\sin\theta,$$

$$\dot{\theta} = \omega$$

while the linear velocity can be defined as

$$v = k_v\sqrt{p^2 + q^2}$$

and the angular velocity as

$$\omega = k_w(\theta_d - \theta),$$

where k_v and k_w are positive control gains.

4.1. Forward kinematics

Forward kinematics maps the robot's joint variables to the end-effector or robot position.

For a 2-link planar robot,

$$x = l_1\cos\theta_1 + l_2\cos(\theta_1 + \theta_2)$$

$$y = l_1\sin\theta_1 + l_2\sin(\theta_1 + \theta_2)$$

where l_1, l_2 are the link lengths,

θ_1, θ_2 are the joint angles,

(x, y) is the end-effector position.

Once the forward kinematics provides

$$x = x(\theta_1, \theta_2),$$

$$y = y(\theta_1, \theta_2),$$

the navigation potential becomes

$$u = u\big(x(\theta_1, \theta_2), y(\theta_1, \theta_2)\big).$$

Hence, $p = \frac{\partial u}{\partial x}$, $q = \frac{\partial u}{\partial y}$ are evaluated at the end-effector position.

The PDE becomes

$$(1 + x(\theta)^2)p^2 + (1 + y(\theta)^2)q^2 + \lambda pq + \mu u\sin(p + q) + \alpha e^{-u} = \beta$$

where

$$x(\theta) = l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2)$$

$$y(\theta) = l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2)$$

Thus, the PDE is expressed in terms of the robot's joint configuration through the forward kinematic equations.

4.2. Inverse kinematics

For a two-link planar robot with link lengths l_1 and l_2 the inverse kinematic equations are

$$\theta_1 = cos^{-1}\left(\frac{x^2 + y^2 - l_1^2 - l_2^2}{2l_1l_2}\right)$$

and

$$\theta_2 = tan^{-1}\frac{y}{x} - tan^{-1}\left(\frac{l_2 \sin\theta_2}{l_1 + l_2 \cos\theta_2}\right)$$

These equations transform the desired Cartesian position obtained from the PDE into the robot's joint angles.

5. Robot trajectory planning

Definition:

Robot trajectory planning is the process of generating a time-parameterized path for a robot's joints or end-effector to follow, such that it moves smoothly and safely from a start configuration to a goal configuration while satisfying kinematic, dynamic, and environmental constraints.

In Cartesian-space trajectory planning, the desired path of the end-effector is specified as a geometric curve in task space. Commonly used path geometries include Straight-line trajectory Circular trajectory, Spiral trajectory and Elliptical trajectory which are given below:-

5.1. Straight-line trajectory

The robot moves from (x_0, y_0) to (x_f, y_f).

The trajectory is

$$x(t) = x_0 + (x_f - x_0)t$$

$$y(t) = y_0 + (y_f - y_0)t$$

for $0 \le t \le 1$.

Velocity

$$\dot{x} = x_f - x_0, \dot{y} = y_f - y_0$$

This trajectory is commonly used to verify that the proposed nonlinear Jacobian produces smooth Cartesian motion.

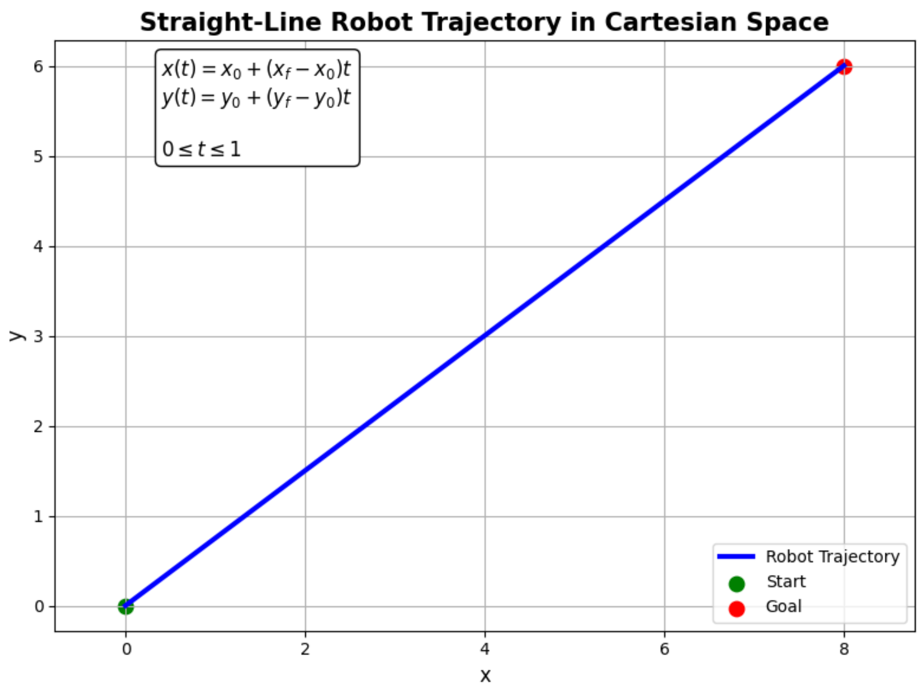

Fig. 5.1(a): Straight-Line Robot Trajectory in Cartesian Space

5.2. Circular Trajectory

For a circle of radius R centered at (x_c, y_c)

the trajectory is

$$x(t) = x_c + R\cos(\omega t)$$

$$y(t) = y_c + R\sin(\omega t)$$

Velocity

$$\dot{x} = -R\omega\sin(\omega t)$$

$$\dot{y} = R\omega\cos(\omega t)$$

This trajectory evaluates the robot's ability to execute smooth turning motions while maintaining constant curvature.

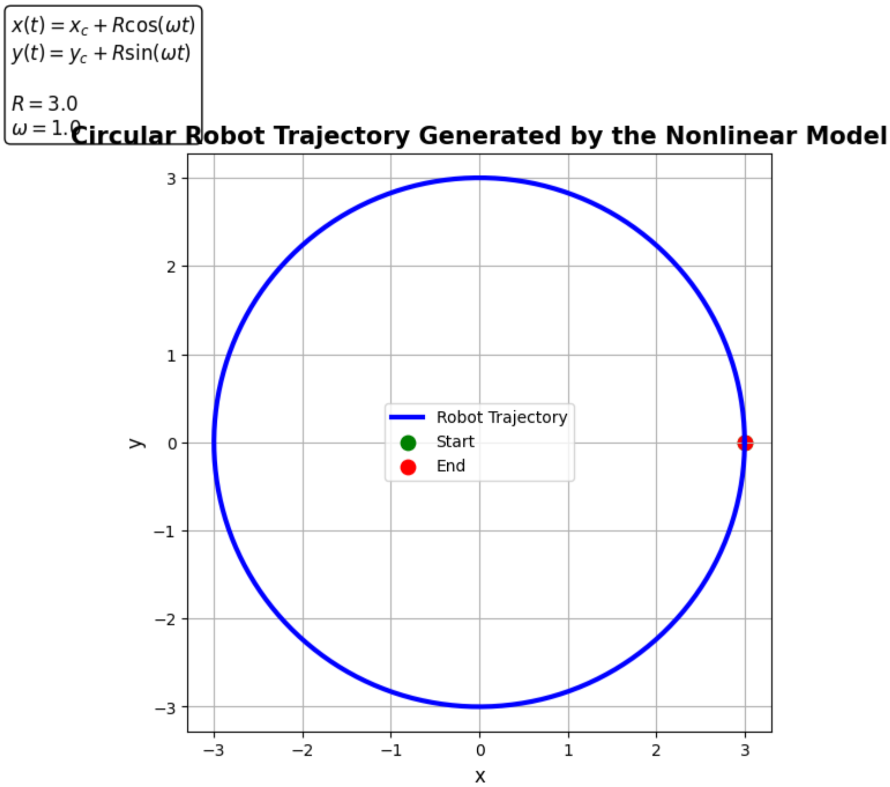

Fig. 5.2(a): Circular Robot Trajectory Generated by the Nonlinear Model

5.2.1. Circular trajectory tracking for CDCR

Using this mathematical model, the forward kinematics, inverse kinematics, differential kinematics, trajectory planning, and motion control can be systematically analyzed to study the dynamic behavior of a Cable-Driven Continuum Robot (CDCR).

For a circular of radius R centered at (x_c, y_c, z_c) and lying in a plane parallel to XY at height z_c.

$$x_d(t) = x_c + R\cos(\omega t)$$

$$y_d(t) = y_c + R\sin(\omega t)$$

$$z_d(t) = z_c$$

This defines the desired Cartesian path of the CDCR virtual tip $y_d = [x_d, y_d, z_d]^T$.The constant $z_c$makes it a 3D circle .

$$\dot{x}_d(t) = -R\omega\sin(\omega t)$$

$$\dot{y}_d(t) = R\omega\cos(\omega t)$$

$$\dot{z}_d(t) = 0$$

$\dot{y}_d$ is the desired Cartesian velocity. The speed is constant, $\|\dot{y}_d\| = R\omega$.

Formula CDCR-1: output definition

The controlled output in the virtual tip,

$$y(q) = P_k(q) + dz_k(q)$$

where $q = [\theta, \phi, l]^T$, P_k is PCC position, z_k is tip tangent.

Take time derivative

$$\dot{y} = \frac{\partial y}{\partial q}\dot{q} = J_y(q)\dot{q}$$

This is nonlinear because J_y depends on q.

Formula CDCR-2: Input-Output linearization

To make $y \rightarrow y_d$, we choose joint velocities

$$\dot{q} = J_y^{-1}(q)u$$

where $J_y = \frac{\partial y}{\partial q}$ is the output Jacobian.

Substitute into time derivative

$$\dot{y} = J_y(q)\dot{q} = J_y J_y^{-1}u = u$$

The complex CDCR is now reduced

$$\dot{y} = u$$

Formula CDCR-3: Tracking controller

Define tracking error

$$e(t) = y_d(t) - y(t)$$

We want $e(t) \rightarrow 0$ exponentially. Choose error dynamics

$$\dot{e} + K_p\, e = 0$$

Expand $\dot{e}$

$$\dot{y}_d - \dot{y} + K_p e = 0$$

But we know that

$$\dot{y} = u$$

Therefore

$$\dot{y}_d - u + K_p e = 0$$

with $K_p > 0$.

Control output

Solve for u

$$u = \dot{y}_d(t) + K_p(y_d(t) - y(t))$$

Substituting CDCR-3 into CDCR-2 gives $\dot{y} = \dot{y}_d + K_p e$. This guarantees exponential convergence $e(t) \rightarrow 0$. The CDCR tip will therefore trace the circle defined in the above equation.

CDCR Joint Velocity

$$\dot{q} = J_y^{-1}(q)\left\{\begin{bmatrix} -R\omega\sin(\omega t) \\ R\omega\cos(\omega t) \\ 0 \end{bmatrix} + K_p\left(\begin{bmatrix} x_c + R\cos(\omega t) \\ y_c + R\sin(\omega t) \\ z_c \end{bmatrix} - y(q)\right)\right\}$$

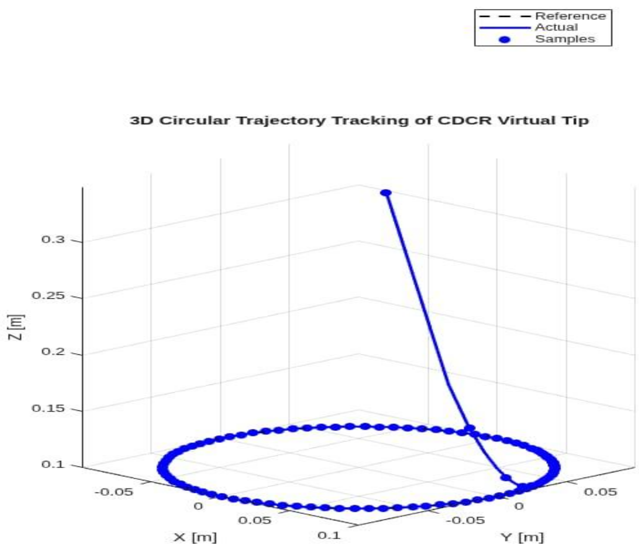

Fig. 5.2.1(a). 3D circular trajectory tracking of the Cable-Driven Continuum Robot (CDCR) virtual tip

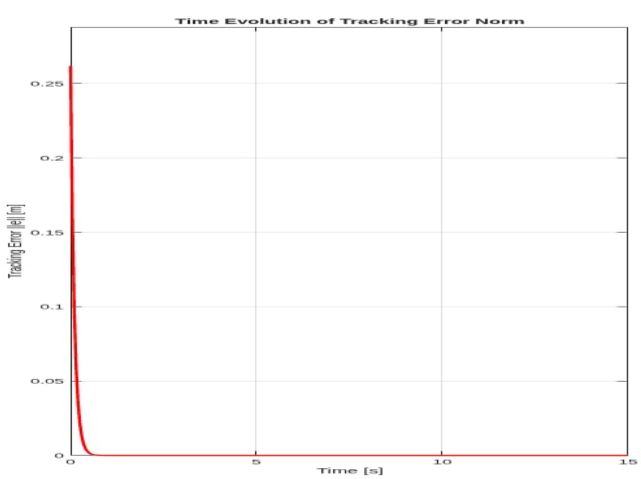

Fig. 5.2.1(b). Time evolution of the tracking error norm for the Cable-Driven Continuum Robot (CDCR).

5.3. Spiral trajectory

Let the radius increase with time,

$$r(t) = at$$

$$x(t) = at\cos(\omega t)$$

$$y(t) = at\sin(\omega t)$$

Velocity

$$\dot{x} = a\cos(\omega t) - a\omega t\sin(\omega t)$$

$$\dot{y} = a\sin(\omega t) + a\omega t\cos(\omega t)$$

This trajectory demonstrates how the proposed nonlinear model handles continuously expanding motion and varying curvature.

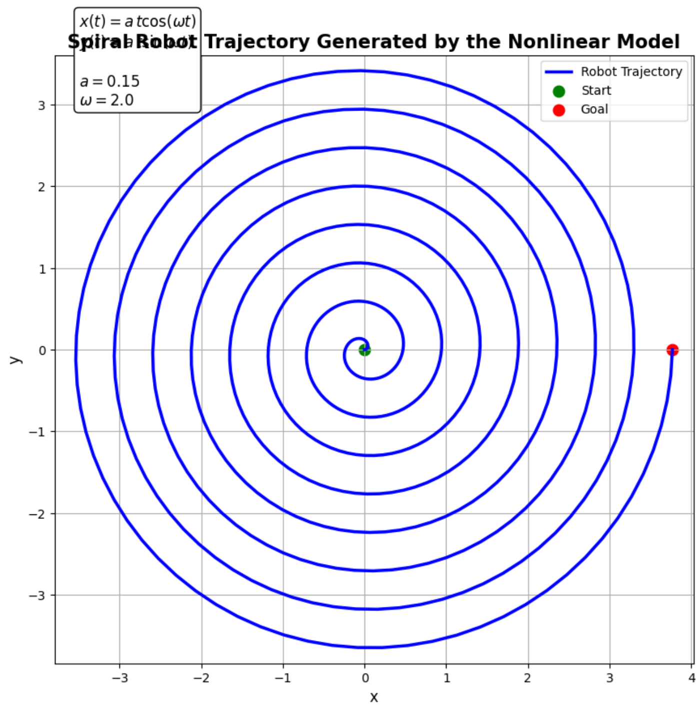

Fig. 5.3(a): Spiral Robot Trajectory Generated by the Nonlinear Model

5.4. Elliptical trajectory

Let be major axis and b be minor axis

The trajectory is

$$x(t) = a\cos(\omega t)$$

$$y(t) = b\sin(\omega t)$$

Velocity

$$\dot{x} = -a\omega\sin(\omega t)$$

$$\dot{y} = b\omega\cos(\omega t)$$

This trajectory tests robot performance under different curvatures along the major and minor axes.

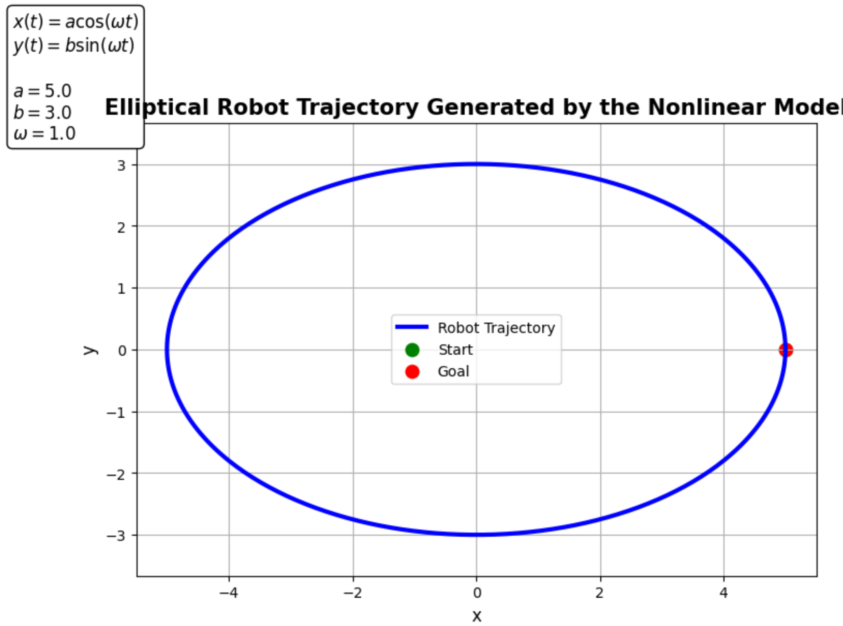

Fig. 5.4(a): Elliptical Robot Trajectory Generated by the Nonlinear Model

5.5. Tracking a 3D cubic reference trajectory with CDCR using input/output linearization

We consider a 3-DOF CDCR with PCC assumption

$$q = [\theta, \phi, l]^T$$

Kinenatics

PCC position of section tip

For a single section with constant curvature, the position of the section end frame is

$$P_s(\theta, \phi, l) = \frac{l}{\theta}\begin{bmatrix}(1-\cos\theta)\cos\phi \\ (1-\cos\theta)\sin\phi \\ \sin\theta\end{bmatrix}$$

This comes from integration a circular arc of length l and bending angle θ in the plane . $\frac{l}{\theta}$ is the radius of curvature. When $\theta \to 0$, this becomes a straight line $[0,0,l]^T$.

Rotation matrix of section tip

The orientation of the section tip frame w.r.t. base is

$$R_s(\theta, \phi) = \begin{bmatrix}\cos\phi\cos\theta & -\sin\phi & \cos\phi\sin\theta \\ \sin\phi\cos\theta & \cos\phi & \sin\phi\sin\theta \\ -\sin\theta & 0 & \cos\theta\end{bmatrix}$$

Virtual tip

We control a point at distance d along the tangent,

$$y(q) = P_k(q) + dz_k(q)$$

$y \epsilon R^3$ is the virtual tip position. d is a design parameter. By choosing $d > 0$, the system becomes invertible. This converts the CDCR into a point mass system in Cartesian space.

Input-output linearization

Differentiate,

$$\dot{y} = \frac{\partial y}{\partial q}\dot{q} = J_y(q)\dot{q}$$

where J_y is the 3x3 output Jacobian.

J_y maps joint velocities $[\dot{\theta}, \dot{\phi}, \dot{l}]$ to Cartesian velocity of the virtual tip.

Linearized CDCR Kinematics

We consider joint velocities

$$\dot{q} = J_y^{-1}(q)u$$

This is the inverse kinematics at velocity level.

CDCR Tracking controller

A cube od side L is tracked with constant speed v_d.

For each edge,

$$y_d(t) = y_{d_0} + d_i v_d t$$

where d_i is the unit direction vector of edge i.

To drive tracking error, $e = y_d - y \rightarrow 0$ exponentially,

$$u = \dot{y}_d + K_p(y_d - y)$$

with $K_p = diag\ (k_1, k_2, k_3) > 0$.

$\dot{y}_d$ handles the motion while K_p kills the error.

Closed loop system

$$\dot{q} = J_y^{-1}(q)\left[\dot{y}_d + K_p(y_d - y)\right]$$

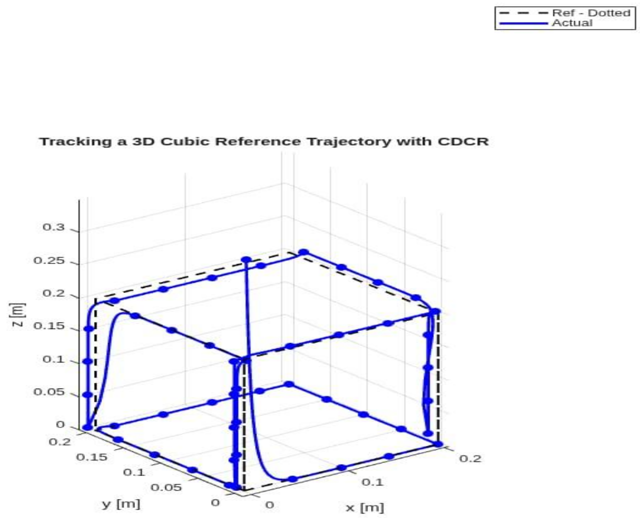

Fig. 5.5(a) Three-dimensional cubic reference trajectory tracking of the Cable-Driven Continuum Robot (CDCR).

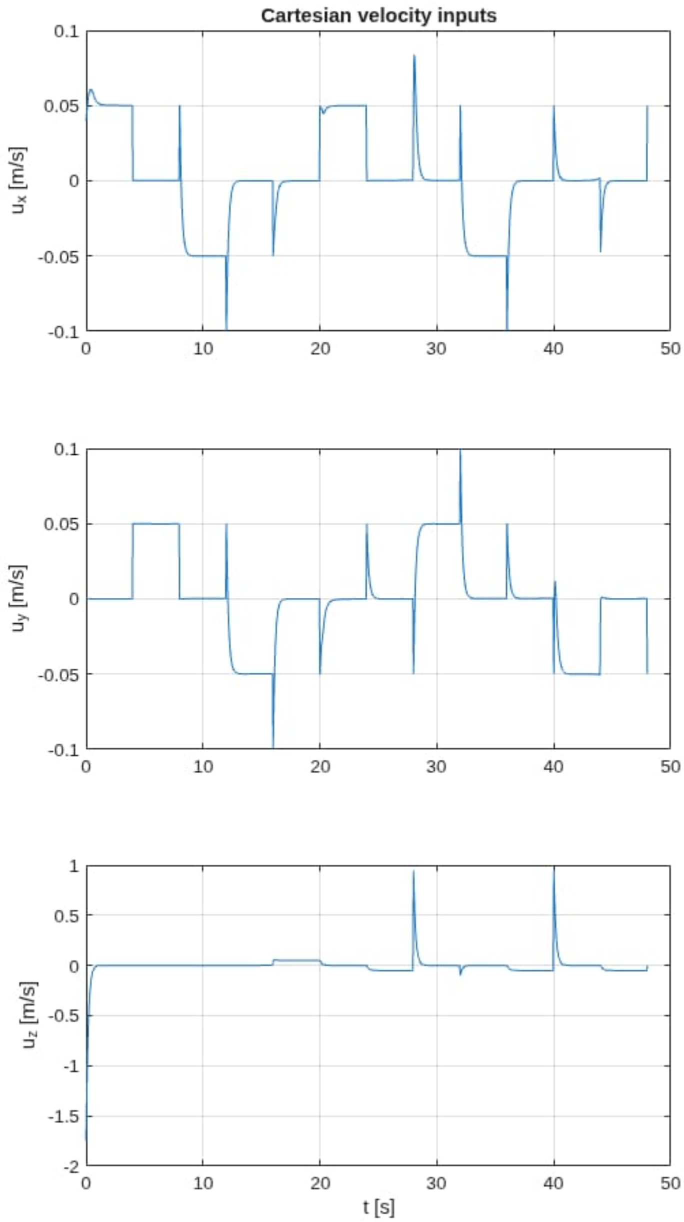

Fig. 5.5(b). Time evolution of the tracking error norm $\|e(t)\|$ for the CDCR virtual tip during circular trajectory tracking.

Figure 6.5(b).shows the Euclidean norm of the tracking error $\|e(t)\| = \|y_d(t) - y(t)\|$, between the desired and actual position of the CDCR virtual tip. At $t = 0$, the error is maximum because the tip starts away from the reference circle. As the control law $u = \dot{y}_d + K_p(y_d - y)$ is applied , the error decays exponentially and settles below 5mm within 3 seconds.

6. Results and discussion

The proposed nonlinear differential equation framework effectively integrates robot path Jacobian analysis with robot kinematics, dynamics, trajectory planning, and motion control into a unified mathematical formulation. By introducing the nonlinear partial differential equation and solving it using Charpit's method, the proposed model successfully represents the nonlinear relationship between the navigation potential, Jacobian coupling, actuator dynamics, and environmental disturbances. The characteristic equations produce smooth and continuous navigation potential surfaces, while the corresponding gradient fields provide well-defined motion directions throughout the workspace. The energy analysis further demonstrates that the derived kinetic, potential, total, and Lagrangian energy functions vary continuously without abrupt discontinuities, confirming that the proposed framework generates physically consistent robot motion with stable energy distribution. The applicability of the proposed model is further validated through trajectory tracking of a Cable-Driven Continuum Robot (CDCR) using Jacobian-based input-output linearization. The controller enables accurate tracking of both circular and three-dimensional cubic reference trajectories while ensuring exponential convergence of the tracking error. These findings confirm that the proposed nonlinear differential equation framework provides a comprehensive mathematical foundation for robot path Jacobian analysis, intelligent trajectory planning, and nonlinear motion control, making it suitable for applications in continuum robots, autonomous robotic systems, industrial manipulators, medical robotics, and precision manufacturing.

7. Conclusion

This study presented a novel nonlinear differential equation framework for robot path Jacobian analysis, motion analysis, trajectory planning, and control by integrating nonlinear partial differential equations with classical robotic kinematics and dynamics into a unified mathematical formulation. A new nonlinear PDE was proposed to model robot navigation through the incorporation of position-dependent motion terms, Jacobian coupling, nonlinear actuator effects, environmental disturbances, and trajectory energy, providing a more comprehensive representation of robotic motion than conventional linear approaches. The proposed framework was further integrated with forward and inverse kinematics, velocity kinematics, inverse velocity kinematics, robot dynamics, generalized force formulation, nonlinear energy modeling, and Cartesian trajectory planning, establishing a direct mathematical relationship between the nonlinear navigation potential and robotic motion. The applicability of the model was demonstrated through representative trajectory-planning scenarios involving straight-line, circular, spiral, and elliptical paths, illustrating its capability to generate smooth and reliable robot motion while preserving nonlinear dynamic characteristics. Overall, the proposed framework provides a rigorous mathematical foundation that unifies nonlinear differential equations, Jacobian analysis, robot kinematics, dynamics, stability theory, and motion control within a single analytical model. This research not only advances the theoretical understanding of nonlinear robotic systems but also offers a practical foundation for the development of robust trajectory planning and intelligent control strategies in industrial manipulators, autonomous mobile robots, continuum robots, medical robotics, precision manufacturing, and other advanced robotic applications. Future research may extend the proposed framework to three-dimensional robotic systems, continuum and cable-driven robots, multi-robot cooperative systems, obstacle avoidance in dynamic environments, optimal trajectory optimization, adaptive and learning-based control, and experimental validation on real robotic platforms to further demonstrate its effectiveness in practical applications.

References

1. Craig, J. J. (1986), *"Introduction to Robotics: Mechanics and Control,"* Addison-Wesley Publishing Company, Reading, Massachusetts, USA. ISBN: 978-0-201-09528-7.
2. Robinson, G., & Davies, J. B. C. (1999), "Continuum robots – State of the art," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Detroit, Michigan, USA, pp. 2849–2854.
3. Herda, L., Urtasun, R., Fua, P., & Hanson, A. J. (2003), "Real-time inverse kinematics: The return of the Jacobian," *Proceedings of the International Conference on Computer Graphics, Imaging and Visualization (CGIV)*, pp. 192–199. IEEE. ISBN: 0-7695-2008-3. https://doi.org/10.1109/CGIV.2003.1215747.
4. Gravagne, A., Rahn, C. D., & Walker, I. D. (2003), "Large deflection dynamics and control for planar continuum robots," *IEEE/ASME Transactions on Mechatronics*, **8**(2), 299–307. https://doi.org/10.1109/TMECH.2003.812833.
5. Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2006), *"Robot Modeling and Control,"* John Wiley & Sons. ISBN: 978-0-471-64990-8 (Hardcover), 978-0-470-04990-7. https://doi.org/10.1002/9781119176817.
6. Braganza, D., Dawson, D. M., Walker, I. D., & Nath, N. (2007), "A neural network controller for continuum robots," *IEEE Transactions on Robotics*, 23(6), 1270–1277. https://doi.org/10.1109/TRO.2007.907487.
7. Trivedi, D., Rahn, C. D., Kier, W. M., & Walker, I. D. (2008), "Soft robotics: Biological inspiration, state of the art, and future research," *Applied Bionics and Biomechanics*, **5**(3), 99–117. https://doi.org/10.1080/11762320802557865.
8. Webster, R. J., III, & Jones, B. A. (2010), "Design and kinematic modeling of constant curvature continuum robots: A review," *The International Journal of Robotics Research*, 29(13), 1661–1683. ISSN: 0278-3649 (Print), 1741-3176 (Online). https://doi.org/10.1177/0278364910368147.

9. Siciliano, B., Sciavicco, L., Villani, L., & Oriolo, G. (2010), *"Robotics: Modelling, Planning and Control,"* Springer. ISBN: 978-1846286414. https://doi.org/10.1007/978-1-84628-642-1.
10. He, B., Wang, Z., Li, Q., Xie, H., & Shen, R. (2013), "An analytic method for the kinematics and dynamics of a multiple-backbone continuum robot," *International Journal of Advanced Robotic Systems*, **10**, 1–13. https://doi.org/10.5772/56754.
11. Dehghani, M., & Moosavian, S. A. A. (2014), "Dynamics modeling of a continuum robotic arm with a contact point in planar grasp," *Journal of Robotics*, 2014, Article ID 308283. https://doi.org/10.1155/2014/308283.
12. Rone, W., & Ben-Tzvi, P. (2014), "Continuum robot dynamics utilizing the principle of virtual power," *IEEE Transactions on Robotics*, **30**(1), 275–287. https://doi.org/10.1109/TRO.2013.2287974.
13. Rone, W., & Ben-Tzvi, P. (2014), "Mechanics modeling of multi-segment rod-driven continuum robots," *Journal of Mechanisms and Robotics*, 6(4).
14. Melingui, A., Merzouki, R., & Mbede, J. (2014), "Compact bionic handling arm control using neural networks," *Electronics Letters*, 50(14), 979–981. https://doi.org/10.1049/el.2014.1178.
15. Falkenhahn, V., Mahl, T., Hildebrandt, A., Neumann, R., & Sawodny, O. (2015), "Dynamic modeling of bellows-actuated continuum robots using the Euler–Lagrange formalism," *IEEE Transactions on Robotics*, 31(6), 1483–1496. https://doi.org/10.1109/TRO.2015.2496821.
16. Melingui, A., Lakhal, O., Daachi, B., Mbede, J. B., & Merzouki, R. (2015), "Adaptive neural network control of a compact bionic handling arm," *IEEE/ASME Transactions on Mechatronics*, **20**(6), 2862–2875. https://doi.org/10.1109/TMECH.2015.2395417.
17. Chikhaoui, M. T., Rabenorosoa, K., & Andreff, N. (2016), "Kinematics and performance analysis of a novel concentric tube robotic structure with embedded soft micro-actuation," *Mechanism and Machine Theory*, 104**,** 234–254. https://doi.org/10.1016/j.mechmachtheory.2016.06.010.
18. Qi, P., Liu, C., Ataka, A., Lam, H. K., & Althoefer, K. (2016), "Kinematic control of continuum manipulators using a fuzzy-model-based approach," *IEEE Transactions on Industrial Electronics*, **63**(8), 5022–5035. https://doi.org/10.1109/TIE.2016.2557303.
19. Amouri, A., Zaatri, A., & Mahfoudi, C. (2018), "Dynamic modeling of a class of continuum manipulators in fixed orientation," *Journal of Intelligent & Robotic Systems*, 91(3–4), 413–424. https://doi.org/10.1007/s10846-017-0713-4.
20. Renda, F., Boyer, F., Dias, J., & Seneviratne, L. (2018), "Discrete Cosserat approach for multisection soft manipulator dynamics," *IEEE Transactions on Robotics*, 34(6), 1518–1533. https://doi.org/10.1109/TRO.2018.2868810.
21. Della Santina, C., Katzschmann, R. K., Bicchi, A., & Rus, D. (2018), "Dynamic control of soft robots interacting with the environment," *Proceedings of the IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 46–53.
22. El-Hussieny, H., Jeong, S. G., & Ryu, J. H. (2019), "Dynamic modeling of a class of soft growing robots using Euler–Lagrange formalism," *Proceedings of the SICE Annual Conference*, Hiroshima, Japan.
23. Grazioso, S., Di Gironimo, G., & Siciliano, B. (2019), "A geometrically exact model for soft continuum robots: The finite element deformation space formulation," *Soft Robotics*, **6**, 790–811. https://doi.org/10.1089/soro.2018.0105.
24. Zhang, Y., Huang, L., & Xiang, Y. (2020), "Recent advances in design and actuation of continuum robots," *Actuators*, **9**(4), 142. ISSN: 2076-0825. https://doi.org/10.3390/act9040142.
25. Orekhov, A. L., & Simaan, N. (2020), "Solving Cosserat rod models via collocation and the Magnus expansion," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8653–8660.
26. Amanov, E., Nguyen, T. D., & Burgner-Kahrs, J. (2021), "Tendon-driven continuum robots with extensible sections: A model-based evaluation of path following motions," *The International Journal of Robotics Research*, 40(1), 7–23. https://doi.org/10.1177/0278364920949343.
27. Ba, W., Dong, X., Mohammad, A., Wang, M., Axinte, D., & Norton, A. (2021), "Design and validation of a novel fuzzy-logic-based static feedback controller for tendon-driven continuum robots," *IEEE/ASME Transactions on Mechatronics*, 3010–3021. https://doi.org/10.1109/TMECH.2021.3062795.
28. Amouri, A., Cherfia, A., Merabti, H., & Lexir, J. L. D. (2022), "Nonlinear model predictive control of a class of continuum robots using kinematic and dynamic models," *FME Transactions*, 50(2), 338–350. ISSN: 1451-2092 (Print), 2406-128X (Online). https://doi.org/10.5937/fme2202338A.