Towards Mechanical Communication in Multi-Agent Locomotive Systems: Principally Kinematic Robots on a Shared Platform

Ruijie Fu

CMU-RI-TR-23-62
November 27, 2023

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA

Thesis Committee:
Prof Howie Choset (Chair)
Dr. Matthew Travers
David Neiman

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Robotics.

Abstract

Many biological multi-agent systems exhibit a mechanism for information exchange among individuals known as mechanical communication, which leads to the emergence of collective behavior within the group. One such example is the swarming behavior of bacteria, where they form rafts and move collectively over solid surfaces using flagella. Similarly, multi-agent groups of articulated robots in a shared environment can dynamically adjust their shape and movements, influencing each other's motion through the ambient media. Recognizing the potential significance of this mechanism, we would like to investigate and harness the power of mechanical communication in the control of multi-agent locomotive systems. To begin with, this thesis focuses on a relatively simple system consisting of principally kinematic robots that share a movable platform, as opposed to more complex ambient media like air or fluid. Initially, to understand the information transmission and reception, two building-block scenarios are explored: 1) an active robot moving a passive platform, and 2) an active platform moving a passive robot. Subsequently, a multi-agent system consisting of both active and passive robots on a platform is analyzed to observe locomotion and communication dynamics.

Throughout this work, we investigate the applicability of geometric motion planning methods in studying the mechanical communication of principally kinematic robots that share a common movable platform. By closely examining the locomotion and communication of the robots, this study prepares the ground for a comprehensive discussion on the trade-offs between them.

Acknowledgments

As I reach a milestone of my academic journey, I am deeply grateful to those who have played a pivotal role in shaping my thesis and the overall trajectory of my graduate experience.

First and foremost, my heartfelt appreciation goes to my advisor, Dr. Howie Choset, whose invaluable guidance, support, and mentorship spanned my time at CMU. His wisdom, passion for research, and commitment to excellence have not only shaped my work but also enriched my understanding of the field.

I am indebted to the members of my committee, Dr. Matthew Travers and David Neiman, for their insightful feedback and dedication to refining the quality of my research. Their expertise has been invaluable in navigating the complexities of my thesis.

I am incredibly fortunate to express my gratitude to Dr. Ross Hatton and Dr. Scott Kelly, whose guidance has been instrumental throughout the journey of my thesis. Working under their mentorship has been a privilege, and I am sincerely thankful for their unwavering support and insightful discussions that have significantly shaped the development of my research.

I am fortunate to have worked alongside the talented individuals in the Biorobotics Lab and other parts of CMU, whose diverse perspectives, and collective pursuit of knowledge have created an environment conducive to innovation. Thanks to Albert Xu, Andrew Orekhov, Shuoqi Chen, Anoop Bhat, Dr. Ji Zhang, and Dr. Keenan Crane, for their very insightful discussions and feedbacks.

A heartfelt shout-out to my parents, Qian Bai and Jiangang Fu, who are absolutely the most caring and supportive parents in the whole world. And thanks to my friends, Angela Zheng, Jieqiong Yu, Qian Weng, and Alex Zheng, whose encouragement and shared laughter provided a welcome respite during the challenges of academia. Your friendship has been a source of strength.

Special thanks to my loving cats, Honey and Momo, and many other cats: Prince, Moli, Brother Cat, Sister Cat, Professor Cat, Nono, Mitten, Kaiky, Caicai, Kiwi.

Contents

1 Introduction **1**

2 Background **5**
2.1 Geometric Locomotion Model 5
2.2 Variational Gait Design 6
2.3 Lagrangian Mechanics 7

3 A 3-link Kinematic Robot on a Movable Platform **9**
3.1 An active 3-link Kinematic Robot Atop a Compliant Platform 9
3.1.1 System Setup 11
3.1.2 Momentum Conservation 13
3.1.3 Gait Optimization 16
3.2 Actively Shaking a Platform to Move a Passive 3-link Kinematic Robot 18
3.2.1 Lagrange d'Alembert Equations of Motion 18
3.2.2 Solving the forward dynamics 19
3.2.3 Prescribing acceleration in robot joints 20
3.2.4 Collapsing Singularity 23
3.2.5 Geometric View 24

4 Multiple 3-link Robots on the Same Platform **27**
4.1 Use an Active Kinematic Robot to Stimulate a Passive Robot on the Same Platform 27
4.2 Chaotic Dynamics 29
4.3 Multiple Active and Passive Robots on the Same Compliant Platform 30

5 Conclusions and Future Work **35**

Bibliography **37**

When this dissertation is viewed as a PDF, the page header is a link to this Table of Contents.

List of Figures

3.1 The configuration of a 3-link kinematic snake adapted from [9]. The fiber coordinates are x, y position and θ orientation of the chosen body frame, in this figure the center link frame. Base space consists of α_1, α_2 the two active joint angles that describes the robot shape. 10

3.2 The connection vector fields over the robot shape space that correspond to the fiber spaces of compliant platform position. Vectors along the diagonal line and at the boundary are scaled down for visualization. . 15

3.3 The constraint curvature function corresponding to the fiber spaces of compliant platform position, plotted over the robot shape space. Values along the diagonal line and at the boundary are scaled down for visualization. Illustrated over the constraint curvature function of x_p are one null-homotopic gait and one gait with winding number (1,1). 16

3.4 Prescribed circular gait of a 3-link kinematic robot that encloses a sign-definite area. 22

3.5 Calculated controlled platform trajectory that induces the prescribed gait of passive robot in Fig. 3.4. The platform is set to start at rest in this example. 23

3.6 Comparison between the desired trajectory and the resulting trajectory from the calculated platform control. 24

3.7 Resulting trajectory of the robot relative to the platform from the calculated platform control in Fig. 3.5. Because the prescribed gait of the robot (Fig. 3.4) encloses a non-zero area in θ direction, the robot has a non-zero orientation change per cycle, leading to a looped trajectory after multiple executions of the same gait. 25

3.8 Convergence of the robot to collapsing configuration after 200s of simulation. The platform shakes in a uniform circular motion, and the passive robot runs from different initial configurations. 26

4.1 The joint angle and position of a passive robot on a platform moved by an active robot over time. As we can see, after cycles of periodic motion, eventually the passive robot enters a collapsed link configuration. But even after the passive 3-link robot collapses into one link, the active robot can still stimulate position motions in the passive robot. 30

4.2 A null-homotopic gait that produces zero net pose change per cycle for the active robot. 31
4.3 Resulting trajectory of the system simulated with the active robot executing the gait in Fig. 4.2. 32
4.4 For the trajectory in Fig. 4.1, despite minor fluctuations caused by numerical methods, the overall preservation of constraints is evident as they vary on a very small scale. 32
4.5 This trajectory of active and passive robots on the same platform provides an example of the use of relatively small motions in an active robot (Fig. 4.1) to induce substantial movements in the passive robot. It demonstrates the potential of motion amplification by using an active robot to stimulate the motion in a passive robot. 33

Chapter 1

Introduction

The study of articulated robots' locomotion has traditionally focused on examining gaits, which are cyclic shape changes that generate desired motions. However, these studies often do not deal with the crucial aspect of the robot's interaction with the ambient environment. Meanwhile, research on multi-agent robot planning has been exploring the conceptualization of robots as points that can freely communicate with each other [3, 12, 18], while overlooking the intricate mechanical structures of the robots.

In contrast, nature provides numerous examples of animals, such as fish, birds, and bacteria [4, 11, 13], that move in swarms and interact with one another through the surrounding media, such as air and fluid. This interaction allows them to exchange information and communicate effectively. Inspired by these natural phenomena, we aim to harness the power of mechanical communication—information transfer through the ambient media—in the control and coordination of articulated robots.

The objective of this thesis is to explore the potential of mechanical communication in the locomotion and coordination of multi-agent systems of articulated robots. To initiate this study, we focus on a simple example: 3-link kinematic robots placed on a movable platform. These well-studied robots consist of three links, each with a non-slip constraint enforced by a wheel, resulting in a constrained velocity perpendicular to the link. The choice of this relatively uncomplicated setup allows us to emphasize the interaction between robots and the exchange of information through the ambient medium, rather than delving into the complexities of fluid dynamics.

The central question we aim to address throughout this research revolves around the trade-offs between robot locomotion and mechanical communication. We break down the information exchange process into three key components: information transmission, information reception, and the locomotion of robot. By studying these components separately, we gain insights into how each robot interacts with the platform and how information is transferred among them.

To examine the dynamics of active robots on passive platforms, we adopt a conventional geometric motion planning method that establishes a linear relationship between the robot's position configuration space and its shape space. This method is well-suited for our system with active robot joints that enable us to establish a linear matrix, which depends only on the robot shape, between the robot's shape velocity and the velocity of the robot and platform, respectively. We then leverage Stokes' theorem to convert the line integral along the connection vector field, approximating the fiber space displacement, into an area integral of the connection vector field's curl. This approach allows us to visualize the displacement as an enclosed volume under an area, particularly useful for analyzing null-homotopic gaits. In the process of utilizing the geometric motion planning method to study the system of an active robot on a passive platform, we encountered and effectively resolved two challenges: addressing gaits on a toroidal shape space that are not null-homotopic and handling gaits that pass through the singularity line. Through slight modifications to the variational optimization method, we successfully overcome these issues.

However, it is important to acknowledge that the geometric motion planning approach does have limitations when applied to passive robots on active platforms. This method relies on symmetry or invariance with respect to certain configuration variables to reduce the dynamic system to a first-order system. Specifically, when analyzing the kinematic robot shape and position, the non-holonomic constraints demonstrate symmetry concerning the robot's relative position to the platform. Similarly, when considering the platform, our Lagrangian is independent of the platform's position. As we move on to study an active platform with a passive robot, the dynamic system lacks symmetry with respect to the robot's shape space due to our control over the platform. As a result, we must utilize the Lagrange d'Alembert principle to analyze this system fully and account for the complexities arising from the active platform.

Throughout the investigation, we encounter an intriguing phenomenon where periodic shaking of the platform causes the three-link robot to collapse into a singular configuration of a straight line. Despite this collapse, we find that the active platform can still drive the robot, allowing it to reach any desired position, although control of the passive robot is limited to only two dimensions.

Subsequently, we extend our analysis to study the multi-robot systems on the same movable platform. We observe that active robots can influence the platform's acceleration while maintaining their independent positions, and the platform can then influence the passive robots. We implemented simulations and controls of a variety of system setups, and analyzed the chaotic dynamics of the systems. This comprehensive study provides valuable insights and lays the groundwork for exploring more complex and practical systems in the future, such as kinematic robots with backdrivable joints that allow active robots to influence each other, or robots operating in more intricate fluid environments.

Chapter 2

Background

In this chapter, we present the essential mathematical preliminaries required to comprehensively study the system in our work. We introduce fundamental terms from differential geometry to accurately describe our system's configuration space. Subsequently, we delve into variational methods used for analyzing kinematic systems, followed by an exploration of Lagrangian mechanics, which proves valuable in handling dynamic systems with high degrees of freedom.

2.1 Geometric Locomotion Model

In the study of locomotion for mobile robots, it is common practice to split the configuration space Q into a base space M and a fiber space G [16]. The position variable $g \in G$ indicates the system's location in the world, while the shape variable $r \in M$ describes the relative arrangements of the system's components. In systems with first-order dynamics, there is a linear relationship between the system's shape velocity and position velocity, given by equation

$$\mathring{g} = -\boldsymbol{A}(r)\dot{r} \tag{2.1}$$

in which $\mathring{g}$ is the system's body frame position velocity, $\dot{r}$ is the system's internal shape velocity, and $\boldsymbol{A}(r)$ is the local connection. Each row of the matrix $\boldsymbol{A}(r)$ represents a covector field that is defined over the shape space and corresponds to a

local derivative of a single position component with respect to the shape r [7].

A gait ϕ is defined as a periodic function $\phi : [0, T] \to M$ that maps from time to the shape space such that $\phi(t) = \phi(t + T)$. For the $SE(2)$ position space, we can approximate the displacement g_ϕ of a gait by the exponential of the surface integral of the constraint curvature function over the gait, i.e.,

$$g_\phi \approx \exp\left(\iint_{\phi_a} D(-\boldsymbol{A}(r))dr\right) \tag{2.2}$$

in which $D(-\boldsymbol{A}) = -\mathrm{d}\boldsymbol{A} + \sum[\boldsymbol{A}_i, \boldsymbol{A}_{j>i}]$ is the curvature of the local connection, which corresponds to the Lie bracket of the full-configuration-space control vector fields associated with the joints, and the surface integral is known as the corrected body velocity integral (cBVI) [8]. The exterior derivative $-d\boldsymbol{A}$ captures the net motion resulted from the change in the coupling between shape and position motion over the shape space, while the local Lie bracket $\sum[\boldsymbol{A}_i, \boldsymbol{A}_{j>i}]$ captures the "parallel parking effect" induced by noncommutativity in position space.

2.2 Variational Gait Design

[14] introduced a variational optimization method for identifying gaits that optimize the efficiency $\frac{g_\phi}{T}$, which we defined in the literature as the displacement g_ϕ generated by a gait over the time period T required to execute the gait at a specified average instantaneous power. In the case of drag-dominated systems [15] and inertia-dominated systems [10], the duration T is proportional to respectively the path-length effort s of the gait and the square root of s, which is weighted by a Riemannian metric.

Given a gait parameterization p, we want to maximize the gait efficiency where the gradient of the efficiency ratio is zero, i.e.,

$$\nabla_p \frac{g_\phi}{s} = \frac{1}{s}\nabla_p g_\phi - \frac{g_\phi}{s^2}\nabla_p s = \mathbf{0} \tag{2.3}$$

By selecting an appropriate initial value of p, it is possible to achieve the maximum efficiency gait by finding the equilibrium of a dynamic system, expressed as:

$$\dot{p} = \nabla_p g_\phi - \frac{g_\phi}{s}\nabla_p s + \nabla_p \sigma \tag{2.4}$$

where the gradient $\nabla_p g_\phi$ acts as "internal pressure" from the constraint curvature function, driving the expansion of the gait displacement. The term $\nabla_p s$, on the other hand, acts as "surface tension" that considers the path-length cost and limits the growth of the gait. The term $\nabla_p \sigma$ is defined based on the shape space metric $\mathcal{M}$, which enforces even sampling of points around the perimeter of the gait.

By utilizing the generalized form of Leibniz Rule [6], the gradient of displacement can be calculated from the *interior product* of the boundary gradient with the integrand [14], which reduces to a simple multiplication between the outward component of $\nabla_p \phi$ and the scalar magnitude of the total Lie bracket for a two-dimensional shape space

$$\nabla_p g_\phi \approx \nabla_p \iint_{\phi_a} D(-\boldsymbol{A}) = \oint_{\phi} (\nabla_{p_\perp} \phi)(D(-\boldsymbol{A})) \tag{2.5}$$

In our implementation, we represent the gait using waypoints. For each waypoint p_i on the gait, consider the triangle it forms with its neighboring points. The base of the triangle determines the local tangent direction $e_{||}$ as $p_{i+1} - p_{i-1} = \ell e_{||}$, where ℓ is the scalar length of the base and $e_{||}$ a unit vector. We also define a local normal direction $e_\perp$ that is orthogonal to $e_{||}$. The gradient of displacement at p_i can be evaluated as

$$\nabla p_i g_\phi = \frac{\ell}{2} e_\perp \tag{2.6}$$

By substituting (2.6) into (2.5), one can observe that the constraint curvature function puts pressure on the gait trajectory to displace in a given direction.

2.3 Lagrangian Mechanics

Lagrangian mechanics offers a streamlined method for analyzing mechanical systems, beginning with the Lagrangian quantity and deriving the system's equations of motion [1]. However, this process often reaches its limits, as the resulting equations can become highly nonlinear and challenging to handle. To address this, prior research in the geometric mechanics of locomotion has aimed to uncover common geometric

features, such as symmetries, to simplify and express these equations in a standardized form. For our system, we employ the Lagrangian approach to study systems with less symmetries, and integrate it with the geometric concepts discussed in the previous section.

The conventional Euler-Lagrangian equations of motion are generally written as

$$\frac{\partial L}{\partial p_i} - \frac{d}{dt}\frac{\partial L}{\partial \dot{p}_i} = \tau_i \tag{2.7}$$

for each p_i coordinate of the configuration variable p, where the Lagrangian L is defined as $L = T - V$ the difference between kinetic energy and the potential energy in the system. Note that although we include τ_i the external force into the equation, in most of the cases we only use the equations of motions for passive coordinates.

In the case of having non-holonomic constraints in the system [2], we can take the constraints into consideration according to the Lagrange-d'Alembert principle. The updated equations of motion appends virtual constraint forces on the system using Lagrange multipliers in order to enforce the constraint:

$$\frac{\partial L}{\partial p_i} - \frac{d}{dt}\frac{\partial L}{\partial \dot{p}_i} = \tau_i + \sum_j \lambda_j \frac{\partial f_j}{\partial \dot{p}_i} \tag{2.8}$$

where $f_j = 0$ are the constraints of the system.

Chapter 3

A 3-link Kinematic Robot on a Movable Platform

3.1 An active 3-link Kinematic Robot Atop a Compliant Platform

We start with the dynamic behavior of a 3-link kinematic snake robot on a compliant movable platform. The 3-link kinematic snake robot is a planar mechanism comprising three links, with each link incorporating a wheel oriented along its length, as illustrated in Fig. 3.1. The wheels possess non-slip capabilities, allowing the robot to maneuver on the ground by exploiting the non-holonomic constraints induced by these wheels. The robot configuration space can be represented as $Q = G \times M$, where $G = SE(2)$ denotes the position space in the world and $M = \mathbb{R}^2$ stands for the shape space [9]. (If we assume no joint limits, M can be represented as $M = \mathbb{T}^2$.)

Consider shape variable $r = (\alpha_1, \alpha_2) \in M$ that represents the joint angles and position variable $g = (x, y, \theta) \in G$ that describes the robot pose, each of the three links' non-holonomic constraint can be expressed as

$$-\sin(\theta_i)\dot{x}_i + \cos(\theta_i)\dot{y}_i = 0, \;\; i = 1, 2, 3 \tag{3.1}$$

where θ_i is the orientation of the i^{th} link and $(\dot{x}_i, \dot{y}_i)$ the velocity at the link center.

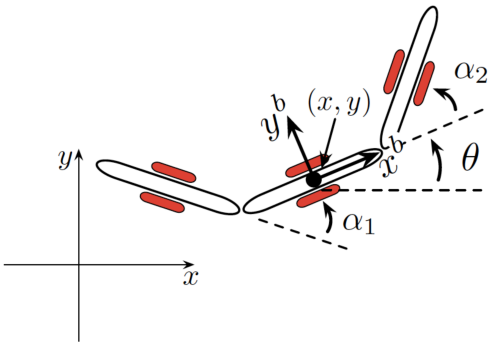


Figure 3.1: The configuration of a 3-link kinematic snake adapted from [9]. The fiber coordinates are x, y position and θ orientation of the chosen body frame, in this figure the center link frame. Base space consists of α_1, α_2 the two active joint angles that describes the robot shape.

Expressing the constraint equations in Pfaffian form, we can get:

$$w_r(r, g)\dot{r} + w_g(r, g)\dot{g} = 0 \tag{3.2}$$

where $w_r \in \mathbb{R}^{3\times 2}$ and $w_g \in \mathbb{R}^{3\times 3}$ depend only on our configuration variables. Note that our constraints are invariant to the fiber space. We can further eliminate g from our reconstruction equation by replacing $\dot{g}$ with the body velocity $\mathring{g}$ to get

$$\mathring{g} = A(r)\dot{r} \tag{3.3}$$

where $A \in \mathbb{R}^{3\times 2}$ is a Jacobian matrix that links the robot's shape and position spaces. Let each link of the robot have length ℓ, and defining the robot frame by the pose of the middle of its center link, we can calculate

$$A(r) = D \begin{bmatrix} \ell \cos(\frac{\alpha_2}{2}) \sec(\frac{\alpha_1}{2}) & \ell \cos(\frac{\alpha_1}{2}) \sec(\frac{\alpha_2}{2}) \\ 0 & 0 \\ 2 \sin(\frac{\alpha_2}{2}) \sec(\frac{\alpha_1}{2}) & 2 \sin(\frac{\alpha_1}{2}) \sec(\frac{\alpha_2}{2}) \end{bmatrix} \tag{3.4}$$

where $D = -\frac{1}{4} \csc(\frac{1}{2}(\alpha_1 - \alpha_2))$. As we can see, A depends only on our base variable r and not the fiber variable g. In this case, the number of non-holonomic constraints in the system happens to equal to the degree of freedom of the fiber space G, allowing the shape velocity of the robot to fully decide the robot body velocity, making the

3-link kinematic robot a principally kinematic system.

3.1.1 System Setup

By situating the kinematic robot on a compliant platform, we effectively place the robot in a noninertial frame. For our analysis, we consider a compliant platform with a radius, R, that has a uniform mass distribution. Notably, the moment of inertia of the platform ($\propto R^4$) grows more rapidly than its mass ($\propto R^2$) as R increases. Thus, for the sake of simplicity, we focus on the translation of the platform, disregarding rotations. As a consequence of introducing the compliant platform, the configuration space of the system becomes $Q = G \times M \times G_p$, where the additional fiber component $G_p = \mathbb{R}^2$ represents the position space of the platform.

Consider the configuration variables $r = (\alpha_1, \alpha_2) \in M, g = (x, y, \theta) \in G, g_p = (x_p, y_p) \in G_p$, where r denotes the robot joint angles, g describes the robot pose relative to the platform, and g_p the platform translation in the world. Tony Dear [5] pointed out that the stratified fiber bundle structure allows us to derive a second mechanical connection that lifts trajectories from the robot shape space M to platform position space G_p.

More specifically, since we defined g as the robot position relative to the platform, Eqn.(3.3) still holds as the non-holonomic constraints depend only on the relative motion between robot and surface. In addition, note that the dynamic of the system does not depend on the platform position. The generalized momentum linked to coordinates (x_p, y_p) is expressed as $(p_{xp}, p_{yp}) = (\frac{dL}{d\dot{x}_p}, \frac{dL}{d\dot{y}_p})$, where L is the Lagrangian of our system. The compliant dimensions of the platform position space should adhere to generalized momentum conservation because they do not appear in the Lagrangian of the system and are not subject to non-holonomic constraints.

Let each link have evenly distributed mass m, moment of inertia $J = \frac{1}{12}m\ell^2$ and the platform mass M_p, the Lagrangian of our system can be calculated as:

$$L = \frac{1}{2}m \sum_{i=1,2,3} (\dot{x}_i^2 + \dot{y}_i^2) + \frac{1}{2}J \sum_{i=1,2,3} \dot{\theta}_i^2 + \frac{1}{2}M_p(\dot{x}_p^2 + \dot{y}_p^2) \tag{3.5}$$

where (x_i, y_i, θ_i) is the pose of link i in the base frame.

As we can see from the equation, the Lagrangian of our system is invariant to x_p

and y_p, the position variables of the platform. In addition, the platform position is not subject to constraints, and there is no external force on the platform. We can write down the Euler-Lagrange equations of motion that correspond to generalized coordinates in x_p and y_p:

$$\frac{\partial L}{\partial x_p} - \frac{d}{dt}\frac{\partial L}{\partial \dot{x}_p} = 0 \tag{3.6}$$

$$\frac{\partial L}{\partial y_p} - \frac{d}{dt}\frac{\partial L}{\partial \dot{y}_p} = 0 \tag{3.7}$$

Because $\frac{\partial L}{\partial x_p} = \frac{\partial L}{\partial y_p} = 0$, the equations of motion turns out to $\frac{d}{dt}p_{xp} = \frac{d}{dt}p_{yp} = 0$. Assuming that our system starts at rest, we have $p_{xp} = p_{yp} = 0$, which symplifies to:

$$-\frac{1}{2}lm(\theta'(\sin(\alpha_2+\theta) - \sin(\theta-\alpha_1)) + \alpha_1'\sin(\theta-\alpha_1) + \alpha_2'\sin(\alpha_2+\theta)) + (3m+M)x_p' + 3mx' = 0 \tag{3.8}$$

$$\frac{1}{2}lm(\theta'(\cos(\alpha_2+\theta) - \cos(\theta-\alpha_1)) + \alpha_1'\cos(\theta-\alpha_1) + \alpha_2'\cos(\alpha_2+\theta)) + (3m+M)y_p' + 3my' = 0 \tag{3.9}$$

By segregating the variables, we can perform a similar operation as done with the robot's shape and position space, yielding the matrix form:

$$B(\theta,\alpha_1,\alpha_2)\begin{bmatrix}\theta'\\ \alpha_1'\\ \alpha_2'\end{bmatrix} + (3m)\mathbb{I}_2\begin{bmatrix}x'\\ y'\end{bmatrix} + (3m+M)\mathbb{I}_2\begin{bmatrix}x_p'\\ y_p'\end{bmatrix} = 0 \tag{3.10}$$

in which

$$B(\theta,\alpha_1,\alpha_2) = \begin{bmatrix} -\frac{\ell m}{2}(\sin(\alpha_2+\theta) - \sin(\theta-\alpha_1)) & -\frac{\ell m}{2}\sin(\theta-\alpha_1) & -\frac{\ell m}{2}\sin(\alpha_2+\theta) \\ \frac{\ell m}{2}(\cos(\alpha_2+\theta) - \cos(\theta-\alpha_1)) & \frac{\ell m}{2}\cos(\theta-\alpha_1) & \frac{\ell m}{2}\cos(\alpha_2+\theta)\end{bmatrix} \tag{3.11}$$

Furthermore, notice that if we pre-multiply the matrix B by a rotation matrix

$$R(-\theta) = \begin{bmatrix}\cos(\theta) & \sin(\theta)\\ -\sin(\theta) & \cos(\theta)\end{bmatrix} \tag{3.12}$$

we get

$$B_1(\alpha_1, \alpha_2) = R(-\theta)B(\theta, \alpha_1, \alpha_2) = \frac{\ell m}{2}\begin{bmatrix} -(\sin(\alpha_1) + \sin(\alpha_2)) & \sin(\alpha_1) & -\sin(\alpha_2) \\ -\cos(\alpha_1) + \cos(\alpha_2) & \cos(\alpha_1) & \cos(\alpha_2) \end{bmatrix} \tag{3.13}$$

that only depends on α_1 and α_2. Thus, if we consider the instantaneous velocity of robot and platform in the local robot body frame, call them $\xi_1 = R(\theta)[x'\ y']^\top, \xi_2 = R(\theta)[x'_p\ y'_p]^\top$, we then have a very nice linear equation whose matrix coefficents depend only on the base variables:

$$-B_1(\alpha_1, \alpha_2)\begin{bmatrix} \theta' \\ \alpha'_1 \\ \alpha'_2 \end{bmatrix} = (3m)\xi_1 + (3m + M)\xi_2 \tag{3.14}$$

Rearranging the terms and plugging in Eqn.(3.3), we can get a connection between the robot shape space and platform position space in the form of

$$\xi_2 = A_p(r)\dot{r} \tag{3.15}$$

3.1.2 Momentum Conservation

Compared to Dear's derivation using generalized momentum conservation illustrated above, we instead utilized the momentum conservation on x and y position to directly build a relationship between the robot position and platform position space.

Consider the system containing one active 3-link kinematic robot on a passive movable platform, where the kinematic robot is actuated only by torques in the joints. Due to the total external force on the system of robot and platform equal to zero, the velocity of the system center of mass remains constant. Let each link have evenly distributed mass m and the platform mass M_p, we can write the momentum of the

system as

$$p_x = m \sum_{i=1,2,3} (\dot{x}_i + \dot{x}_p) + M_p \dot{x}_p$$
$$p_y = m \sum_{i=1,2,3} (\dot{y}_i + \dot{y}_p) + M_p \dot{y}_p \tag{3.16}$$

where (x_i, y_i) is the relative position of each link to the platform.

Moreover, recall that the body frames described by the coordinates in any two valid coordinate sets (i.e. those for which a local connection can be derived) are related by a shape-dependent transformation [8]. Let's define the robot frame to be the center of mass frame $g_c = (x_c, y_c, \theta_c)$, where $x_c = \frac{1}{3}\sum_i x_i, y_c = \frac{1}{3}\sum_i y_i, \theta_c = \frac{1}{3}\sum_i \theta_i$. The center of mass frame, related to the center link frame in the previous section by only a shape-dependent transformation, is also a valid body frame. Similar to Eqn.(3.2), we can write down the Pfaffian form of constraints in Eqn.(3.1) in terms of g_c and r, and further eliminate g from our reconstruction equation by replacing $\dot{g}_c$ with the body velocity $\mathring{g}_c$ to get

$$\mathring{g}_c = A_c(r)\dot{r} \tag{3.17}$$

In terms of the new body frame, the momentum can be further simplified to

$$p_x = 3m\dot{x}_c + (3m + M_p)\dot{x}_p$$
$$p_y = 3m\dot{y}_c + (3m + M_p)\dot{y}_p \tag{3.18}$$

Consider the momentum conservation, $\frac{d}{dt}p_x = 0$, $\frac{d}{dt}p_y = 0$, let the initial momentum of the system be p_{x0}, p_{y0}, we can easily derive the platform velocity from robot center of mass velocity as

$$\dot{x}_p = \frac{p_{x0} - 3m\dot{x}_c}{3m + M_p}$$
$$\dot{y}_p = \frac{p_{y0} - 3m\dot{y}_c}{3m + M_p} \tag{3.19}$$

In the common case of the system starting at rest, the initial momentum of the

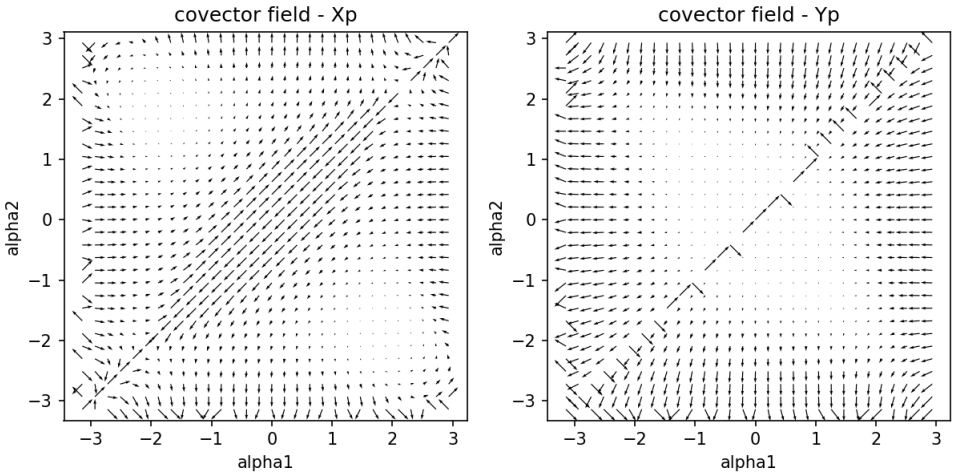


Figure 3.2: The connection vector fields over the robot shape space that correspond to the fiber spaces of compliant platform position. Vectors along the diagonal line and at the boundary are scaled down for visualization.

system is zero, we have:

$$\begin{aligned} \dot{x}_p &= -\frac{3m}{3m + M_p}\dot{x}_c \\ \dot{y}_p &= -\frac{3m}{3m + M_p}\dot{y}_c \end{aligned} \tag{3.20}$$

If we define a "body velocity" $\mathring{g}_p$ for platform as the platform velocity in the instantaneous robot frame, we can extend the linear relationship to

$$\mathring{g}_p = -\frac{3m}{3m + M_p} A_c(r)\dot{r} \tag{3.21}$$

and apply geometric methods to design gaits of the robot that moves the platform. The corresponding covector field is illustrated in Fig. 3.2.

In addition to being more straightforward and having simpler mathematical derivation, our method using regular momentum conservation chooses the center of mass frame as the robot body frame, which coincides with the minimum perturbation frame for kinematic robots [8]. The body velocity integral calculated in this frame best approximates the displacement of robot in the position space, making it more accurate for designing gaits of the robot that moves the compliant platform.

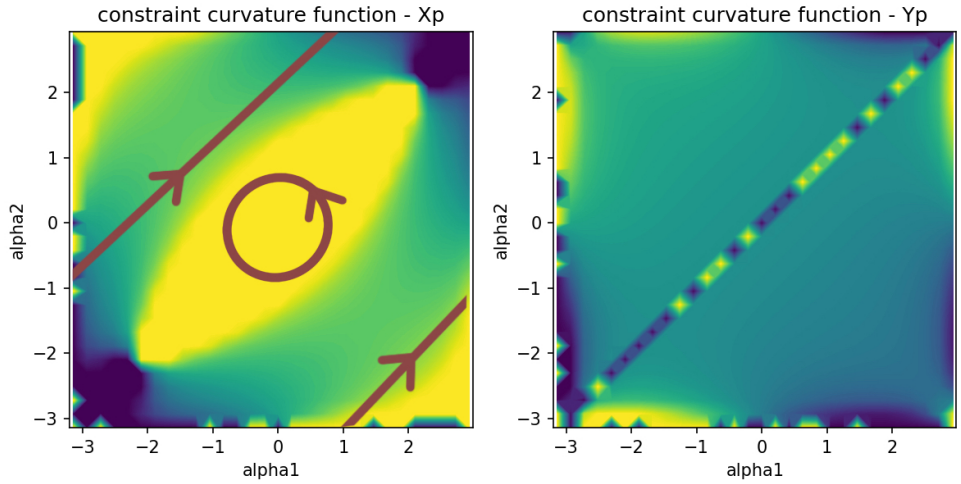


Figure 3.3: The constraint curvature function corresponding to the fiber spaces of compliant platform position, plotted over the robot shape space. Values along the diagonal line and at the boundary are scaled down for visualization. Illustrated over the constraint curvature function of x_p are one null-homotopic gait and one gait with winding number (1,1).

3.1.3 Gait Optimization

Now that we have calculated the connection between robot shape space and robot position space, as well as the connection between robot shape space and the platform position space, we can perform the regular geometric motion planning methods, such as the variational optimization method, to design efficient gaits that are able to move the platform in a desired direction. We developed methods to overcome two special cases, specifically targeting gaits that wind around the cyclic dimensions of shape spaces and gaits that cross singularities, as introduced below. For the sake of space limitation, we will not provide full description of the methods here.

Geometric Motion Planning on Toroidal Shape Space

Most prior works on such geometric analysis do not account for cyclic degrees of freedom in the shape space. In our system, however, if we consider the robot without joint limits, our base space effectively becomes a torus. By considering the underlying toroidal topology, we are now able to explore a broader range of gaits compared to

previous approaches. For instance, the gait with winding number $(1, 1)$ depicted in Fig. 3.3 would have been classified as an open loop without considering the cyclic dimensions, but in the toroidal shape space, it manifests as a closed loop that generates periodic motion.

Moreover, with the quantitative evaluation of gait efficiency presented in [15] and [10], we can extend the variational optimization method to get a pipeline for generating optimal gaits in systems with cyclic degrees of freedom in their shape spaces (e.g., torus). The pipeline comprises three distinct elements:

- determine an optimal choice of body frame that accounts for the cyclic nature of the shape space
- select a seed gait with an appropriate winding number
- perform a constrained variational optimization on the system that takes into consideration the topology of the shape space

Crossing Singularity

Note that when $\alpha_1 = \alpha_2$ in Eqn.(3.4), our connection matrix becomes infinite, indicating that the robot falls into a singular configuration. As evident from the covector field, these C-shaped singular configurations appear as a diagonal line. Dealing with this singularity line during gait design would be highly inconvenient. However, the singular configuration is not entirely insurmountable if we approach it with a specific speed. Computing and inspecting w_r and w_g from Eqn.(3.2) reveals that the singularity occurs because one side of the equations is consistently zero while the other side is not. Nevertheless, if we additionally satisfy the condition $\dot{\alpha}_1 + \dot{\alpha}_2 = 0$, both sides become zero, enabling us to cross the singularity line. In the covector field, this crossing is depicted as crossing the singularity line perpendicularly. For optimization on the gait family that crosses the singularity, please refer to [10].

3.2 Actively Shaking a Platform to Move a Passive 3-link Kinematic Robot

To further investigate the interplay of multiple robots on the same platform, we must first address another intriguing problem: actively inducing motion in a passive snake robot situated on the platform. In the previous section, we calculated the connection between the robot's shape space and its corresponding position space, as well as the connection between the robot's shape space and the platform's position space.

Although it may be tempting to straightforwardly invert the connection matrix to deduce the forward dynamics of the system, this approach encounters a challenge in the current setup where we have an active platform instead of a passive one. This divergence introduces external force into the system, disrupting momentum conservation in the (x_p, y_p) generalized coordinates and consequently alters the constrained subspace.

In this case, the nonholonomic constraints, although still exist and define the relationship between the robot joint velocity and the robot position velocity relative to the platform, are not sufficient to determine the kinematics and dynamics of the system. In other words, there are more than one solution of the system velocity that satisfies the constraints. Among the multiple results, the system behaves according to the principle of least action, also known as the stationary action principle.

3.2.1 Lagrange d'Alembert Equations of Motion

To tackle this issue, we employ the Lagrange d'Alembert principle as a valuable tool to study the dynamics of this system. The Lagrangian of the system comprises the kinetic energy of both the robot and the platform, as previously mentioned in Eqn.(3.5). Additionally, we introduce virtual constraint forces using Lagrange multipliers to account for the non-holonomic constraints. This enables us to formulate the Euler-Lagrange equations of motion in the form of

$$\frac{\partial L}{\partial p_j} - \frac{d}{dt}\frac{\partial L}{\partial \dot{p}_j} = \sum_{i=1,2,3} \lambda_i \frac{\partial f_i}{\partial \dot{p}_j} \tag{3.22}$$

where p_j can be each of the non-controllable variables in our configuration space, $f_i = 0$ is the non-holonomic constraint on link i, and λ_i is the corresponding Lagrange multiplier.

Consider our configuration variable $X = [g\ r\ g_p]^\top = [x\ y\ \theta\ \alpha_1\ \alpha_2\ x_p\ y_p]^\top$. Define our state variable $q = [X\ \dot{X}]^\top$, we can write the unforced state-space representation of our linear system as

$$\begin{bmatrix} \mathbb{I} & 0 \\ 0 & H(q) \end{bmatrix} \dot{q} + \begin{bmatrix} -\dot{X} \\ K(q) - W(p)^\top \Lambda \end{bmatrix} = 0 \tag{3.23}$$

where $\Lambda = [\lambda_1, \lambda_2, \lambda_3]^\top$ is the Lagrange multiplier, $H(q) \in \mathbb{R}^{7\times7}, K(q) \in \mathbb{R}^7, W(p) \in \mathbb{R}^{3\times7}$ depend only on q and not $\dot{q}$. More specifically, H and W depend only on $\theta, \alpha_1, \alpha_2$, and K depends only on $\theta, \alpha_1, \dot{\theta}, \alpha_2, \dot{\alpha}_1, \dot{\alpha}_2$. Due to the limit of space, we will only refer to the matrices with symbols in this section.

3.2.2 Solving the forward dynamics

Let's consider our dependent variable $p = [g\ r]^\top = [x\ y\ \theta\ \alpha_1\ \alpha_2]^\top$ and controlled variable $u = g_p = [x_p\ y_p]^\top$. We can write the equations of motion into matrix form:

$$A(p)\ddot{p} + B(p)\ddot{u} + C(p, p') = U(p)^\top \Lambda \tag{3.24}$$

where $A \in \mathbb{R}^{5\times5}, B \in \mathbb{R}^{5\times2}, C \in \mathbb{R}^5, W \in \mathbb{R}^{3\times5}$ can be extracted from sectioning the matrices in Eqn.(3.23) as

$$\begin{aligned} H_{7\times7}(\theta, r) &= \begin{bmatrix} A_{5\times5}(\theta, r) & B_{5\times2}(\theta, r) \\ D_{2\times5}(\theta, r) & E_{2\times2}(\theta, r) \end{bmatrix} \\ K_{7\times1}(\theta, \dot{\theta}, r, \dot{r}) &= \begin{bmatrix} C_{5\times1}(\theta, \dot{\theta}, r, \dot{r}) \\ F_{2\times1}(\theta, \dot{\theta}, r, \dot{r}) \end{bmatrix} \\ U_{3\times7}(\theta, r) &= \begin{bmatrix} W_{3\times5}(\theta, r) & V_{3\times2}(\theta, r) \end{bmatrix} \end{aligned} \tag{3.25}$$

A, B, C, W do not depend on the position x, y of the robot or the position x_p, y_p of the platform. This observation validates the phenomena depicted in testing animations, where the system's dynamics remain position-invariant. In other words,

the platform's location in the world and the robot's location on the platform has no impact on the overall system dynamics.

Additionally, when we perform rotation matrix multiplication on matrices A, B, and C, similar to that in Eqn.(3.13) we can further eliminate θ and obtain coefficient matrices that are solely dependent on the robot shape variable r. This finding aligns with the notion that rotating both the platform input and the robot's orientation yields essentially identical results.

Note that the determinant of $A(\theta, r)$ is equal to

$$det(A) = \frac{1}{192} l^6 m^5 (-9\cos(2\alpha_1) - 9\cos(2\alpha_2) + \cos(2(\alpha_1 + \alpha_2)) + 37) \tag{3.26}$$

suggesting A to be invertible. We can derive the Lagrange multipliers as a function of θ, r:

$$\Lambda = (WA^{-1}W^\top)^{-1}(W\ddot{p} + WA^{-1}B\ddot{u} + WA^{-1}C) \tag{3.27}$$

Recall that matrix W encodes the non-holonomic constraints on the robot variables, namely $W\dot{p} = 0$. Taking a derivative of $W\dot{p}$, we can derive $\frac{d}{dt}(W\dot{p}) = \dot{W}\dot{p} + W\ddot{p} = 0$. Plugging in to Eqn.(3.27), we can obtain

$$\Lambda = (WA^{-1}W^\top)^{-1}(-\dot{W}\dot{p} + WA^{-1}B\ddot{u} + WA^{-1}C) \tag{3.28}$$

Hereby we have implicitly solved the Lagrange multipliers Λ as a function of our configuration variables and their first order derivatives. We can plug it back into Eqn.(3.24) to arrive at the full dynamic update equation:

$$A\ddot{p} + (B - W^\top(WA^{-1}W^\top)^{-1}WA^{-1}B)\ddot{u} + C - W^\top(WA^{-1}W^\top)^{-1}(-\dot{W}\dot{p} + WA^{-1}C) = 0 \tag{3.29}$$

3.2.3 Prescribing acceleration in robot joints

Given an instantaneous desired acceleration in the robot joints, we want to find the input acceleration of the platform that can produce the prescribed robot joint acceleration. It is intuitive to apply a method similar to that in forward dynamics:

write down matrix form of dynamic equation using Lagrange d'Alembert equations of motion, calculate the instantaneous values of the Lagrange multipliers, and then solve for the required input acceleration. Following this method, we can reach an equation similar to Eqn.(3.24),

$$A_1[\ddot{g} \quad \ddot{g}_p]^\top + (B_1 - W_1^\top(W_1A_1^{-1}W_1^\top)^{-1}W_1A_1^{-1}B_1)\ddot{r} + C - W_1^\top(W_1A_1^{-1}W_1^\top)^{-1}(-\dot{W}\dot{p} - U_2\dot{r} + W_1A_1^{-1}C) = 0 \tag{3.30}$$

where the coefficient matrices are again extracted from

$$\begin{aligned} H_{7\times7} &= \begin{bmatrix} H_{11(3\times3)} & H_{12(3\times2)} & H_{13(3\times2)} \\ H_{21(2\times3)} & H_{22(2\times2)} & H_{23(2\times2)} \\ H_{31(2\times3)} & H_{32(2\times2)} & H_{33(2\times2)} \end{bmatrix} \\ U_{3\times7} &= \begin{bmatrix} U_{1(3\times3)} & U_{2(3\times2)} & U_{3(3\times2)} \end{bmatrix} \end{aligned} \tag{3.31}$$

as

$$\begin{aligned} A_1 &= \begin{bmatrix} H_{11} & H_{13} \\ H_{21} & H_{23} \end{bmatrix} \\ B_1 &= \begin{bmatrix} H_{12} \\ H_{22} \end{bmatrix} \\ W_1 &= \begin{bmatrix} U_1 & U_3 \end{bmatrix} \end{aligned} \tag{3.32}$$

The matrix coefficient of $[\ddot{g} \quad \ddot{g}_p]^\top$ is the singular matrix A_1. To avoid inverting the singular matrix, we can instead continue to write Λ as a function of the control variable g_p that is to be solved as in Eqn.(3.28) and calculate the instantaneous control acceleration as

$$\ddot{x}_p = A_2^{-1}(-B_1\ddot{r} - C + W^\top(WA^{-1}W^\top)^{-1}(-\dot{W}\dot{p} + WA^{-1}C) \tag{3.33}$$

where

$$A_2 = \begin{bmatrix} [H_{11} \quad H_{21}]^\top & B - W^\top(WA^{-1}W^\top)^{-1}WA^{-1}B \end{bmatrix} \tag{3.34}$$

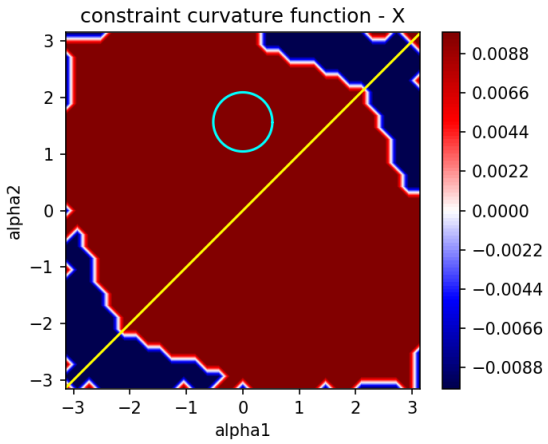


Figure 3.4: Prescribed circular gait of a 3-link kinematic robot that encloses a sign-definite area.

Pipeline Demo

Consider a passive 3-link kinematic robot on an active platform. Given that we would like to control the robot to move along the local x direction, we can prescribe a gait for the robot that encloses a sign-definite area in the constraint curvature function. As an example, we choose the circular gait in Fig. 3.4 that avoids the singularity line as our desired trajectory.

Without loss of generality, we let the platform start at rest from world location $(0, 0)$, and calculate the platform control in order to stimulate the desired acceleration of the trajectory in the passive robot, as shown in Fig. 3.5.

We can then plug in the derived control of platform into simulation (3.29) and compare the result to desired trajectory, as a validation of our pipeline. As shown in Fig. 3.6, the resulting trajectory from our calculated control of the platform differs from the desired trajectory only up to numerical error.

As a side note, when we prescribe the gait, we chose one off the singularity line, resulting in a non-zero orientation change per cycle (Fig. 3.7). After we execute the gait long enough, the resulting trajectory forms a loop. In order for the robot to move farther, we can jump to another gait before the trajectory closes on itself.

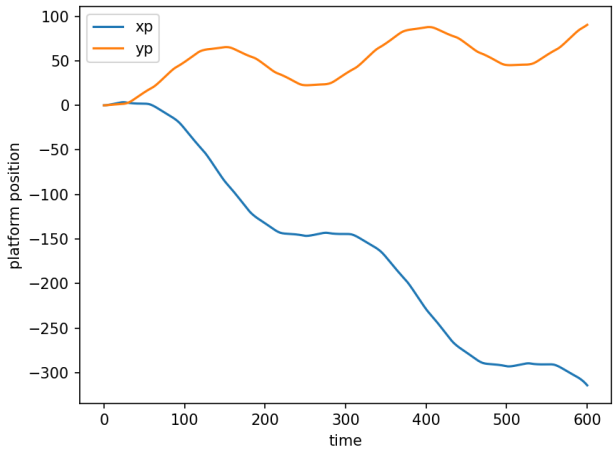
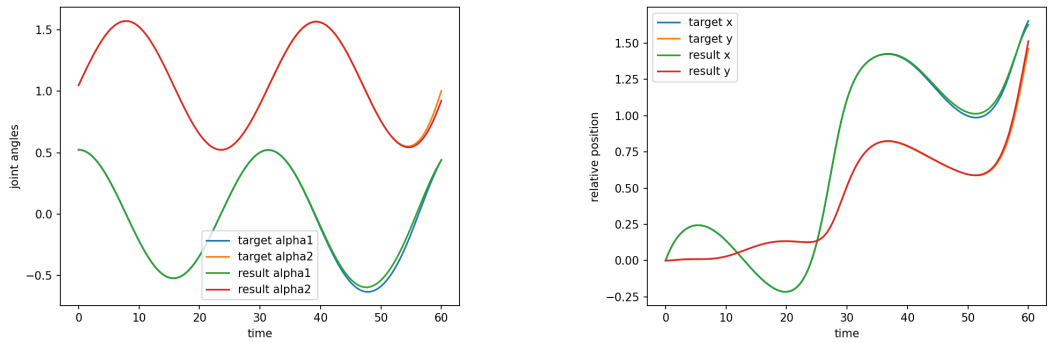


Figure 3.5: Calculated controlled platform trajectory that induces the prescribed gait of passive robot in Fig. 3.4. The platform is set to start at rest in this example.

3.2.4 Collapsing Singularity

The scatter plot (Fig. 3.8) clearly reveals an important issue arising from our system setup: after subjecting the platform to periodic shaking for a considerable duration, the passive robot experienced the collapse of at least one link, which resulted in the robot stuck in a $r = (\pi, \pi)$ or $(\pi, 0)$ singular configuration. This development poses a significant challenge to our study, as our primary objective is to stimulate the movement of a passive robot by utilizing an active robot on the same platform. Normally, the design of a gait, a rhythmic motion pattern for the active robot, would be pursued to achieve this goal. However, in the current scenario, there is a high likelihood that the passive robot will be trapped in a singular configuration with collapsed links, which is entirely undesirable.

To thoroughly investigate the tendency of the robot to fall into this undesirable singular configuration, we conducted a series of analyses. By substituting the configurations $(\alpha_1, \alpha_2, \dot{\alpha}_1, \dot{\alpha}_2) = (\pi, \pi, 0, 0)$ and $(\pi, 0, 0, 0)$ into the dynamic updating equations and performing simplifications, we found that the coefficient associated with $\ddot{u}$ has no effect on the $\theta, \alpha_1, \alpha_2$ dimensions. This finding implies that irrespective of the controlled input applied, the configurations are local fixed points of our system. In other words, the robot is prone to reaching these specific configurations regardless of



(a) joint angles (b) robot position

Figure 3.6: Comparison between the desired trajectory and the resulting trajectory from the calculated platform control.

the input, which further validates our concern about the likelihood of it encountering the collapsed link singular configuration.

With this property in mind, in the next section, where we attempt to control an active robot to induce a desired gait in a passive robot on the same platform, we expect to directly calculate the control of active robot from dynamics, rather than designing a gait for the active robot, which tend to result in passive robot link collapse.

3.2.5 Geometric View

As a side note, we can adopt a similar approach to solve for the forward dynamics of an active robot on a compliant platform, which we studied using the geometric motion planning method in the previous section. The Lagrange equations of motion provide the complete dynamics of the system. By representing the configuration variables as state variables and control variables, we can determine which dimensions are active and then proceed to solve for the forward dynamics and control.

One might wonder if it is also possible to apply geometric methods to solve the system of a passive robot on an active platform, which would offer a nicely linear representation and provide a clear visualization of the system. However, the answer is not positive. To achieve linearity, a conserved quantity, as discussed before, becomes

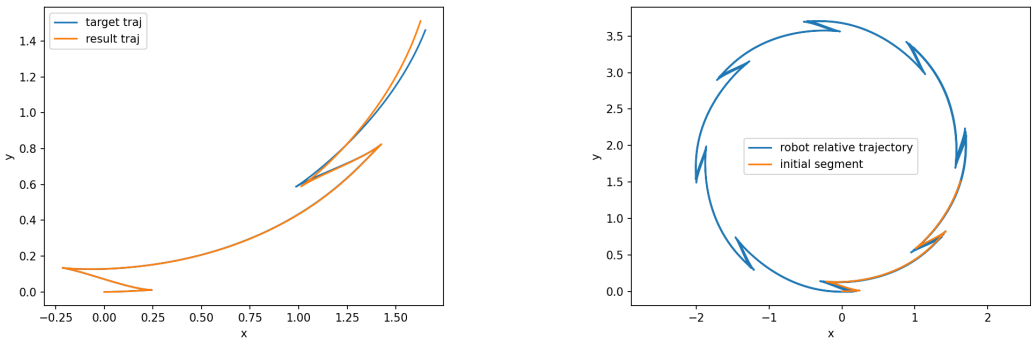


Figure 3.7: Resulting trajectory of the robot relative to the platform from the calculated platform control in Fig. 3.5. Because the prescribed gait of the robot (Fig. 3.4) encloses a non-zero area in θ direction, the robot has a non-zero orientation change per cycle, leading to a looped trajectory after multiple executions of the same gait.

necessary. Specifically, the generalized momentum conservation requires that not only do we maintain symmetry in the generalized coordinates, but also the constraints must be invariant to the coordinates. Unfortunately, satisfying this requirement is rare, although we were successful in achieving it when dealing with active robot joints, in which case the fiber spaces that correspond to the robot and platform positions both have this symmetric property.

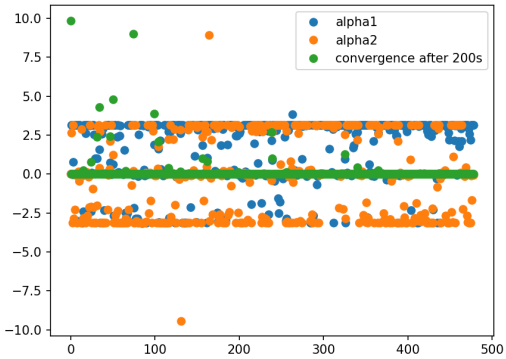


Figure 3.8: Convergence of the robot to collapsing configuration after 200s of simulation. The platform shakes in a uniform circular motion, and the passive robot runs from different initial configurations.

Chapter 4

Multiple 3-link Robots on the Same Platform

Now, let's explore the scenario where we place multiple robots on the same movable platform. Within this setup, certain robots are active, navigating by altering their shapes, while others are passive, moving in response to the platform's motion induced by the active robots. We show that we can essentially study the interaction between each robot and its environment individually by leveraging the insights gained from the two basic cases presented in the preceding section.

4.1 Use an Active Kinematic Robot to Stimulate a Passive Robot on the Same Platform

To begin with, let's consider the minimal case where there is one active robot and one passive robot on the movable platform. The joint motions of the active robot, call it robot 1, cause the platform to shake, subsequently inducing motion in the passive robot, call it robot 2. For this case, we'll denote the control variable as $u = [\alpha_1^{R1}\ \alpha_2^{R2}]^\top$ and the passive variable as $p = [x^{R1}\ y^{R1}\ \theta^{R1}\ x^{R2}\ y^{R2}\ \theta^{R2}\ x_p\ y_p]^\top$.

The Lagrangian of the system can be represented by

$$L = \frac{1}{2}m \sum_{i=1,2} \sum_{j=1,2,3} ((\dot{x}_j^{Ri})^2 + (\dot{y}_j^{Ri})^2) + \frac{1}{2}J \sum_{i=1,2} \sum_{j=1,2,3} (\theta_j^{Ri})^2 + \frac{1}{2}M_p(\dot{x}_p^2 + \dot{y}_p^2) \quad (4.1)$$

where $(x_j^{Ri}, y_j^{Ri}, \theta_j^{Ri})$ denote the pose of the j^{th} link of robot i in the world frame. To calculate the forward dynamics of the system, we can represent the state function retrieved from Lagrange-d'Alembert equations of motion (3.22) in the form of

$$\begin{bmatrix} A_{5\times5}(\theta_1, r_1) & 0_{5\times5} & B_{5\times2}(\theta_1, r_1) \\ 0_{5\times5} & A_{5\times5}(\theta_2, r_2) & B_{5\times2}(\theta_2, r_2) \\ D_{2\times5}(\theta_1, r_1) & D_{2\times5}(\theta_2, r_2) & E_{2\times2}(\theta_1, \theta_2, r_1, r_2) \end{bmatrix} \begin{bmatrix} \ddot{g}_1 \\ \ddot{r}_1 \\ \ddot{g}_2 \\ \ddot{r}_2 \\ \ddot{g}_p \end{bmatrix} + \begin{bmatrix} C_{5\times1}(\theta_1, \dot{\theta}_1, r_1, \dot{r}_1) \\ C_{5\times1}(\theta_2, \dot{\theta}_2, r_2, \dot{r}_2) \\ F_{2\times1} \end{bmatrix} = W(p)^\top \Lambda \quad (4.2)$$

where A, B, C, D are sub-matrices that depend only on the configuration of a single robot, as mentioned in Eqn.(3.25). For forward dynamics, we can directly derive a full dynamic update equations from this equation in the same way as Eqn.(3.29).

Notice that as we take the derivative of Lagrangian with respect to the configuration variables of robot 1 or their derivatives, the result is independent from the configuration variables of robot 2, leading to the zeros in Eqn.(4.2). When we consider a more complicated system with more than two robots on the same platform, in the state function $J\ddot{p} + K = W(p)^\top \Lambda$, the coefficient matrix J of configuration variable has $J_{ij} = 0$ as long as p_i and p_j are configuration variables that belong to different robots.

Thus, as we analyze the equation for the control problem, we can decompose it into two distinct parts that respectively describes the dynamics of the two robots. This decomposition allows us to address two separate problems we have already solved: 1. Using the active robot to shake the platform, and 2. Utilizing the platform to stimulate motion in a passive robot. As expected, when we input periodic gaits to the active robot, the platform shakes with periodic motion, which, in turn, affects the passive robot—potentially leading to its collapse. To overcome this situation, we can design a control mechanism to prescribe the trajectory of the passive robot. Notably,

we are primarily controlling only two degrees of freedom that significantly influence the passive robot, and we can prescribe the trajectory in two dimensions.

Pipeline Demo

Suppose again we want to control the passive robot to execute a circular gait, as prescribed in Fig. 3.4. From Eqn.(4.2) we can extract row 6-10 to get

$$A \begin{bmatrix} \ddot{g}_2 \\ \ddot{r}_2 \end{bmatrix} + B\ddot{g}_p + C = U^\top \Lambda \tag{4.3}$$

which is in the same form as Eqn.(3.24), from which we can calculate the platform trajectory (Fig. 3.5) needed to induce the gait.

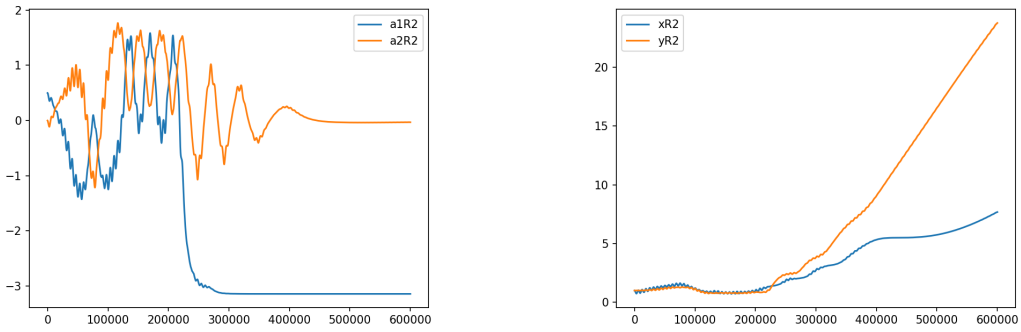
Having established the necessary platform trajectory and identified the desired passive robot gait, let's examine the entire system comprising an active robot, a passive robot, and a platform. The active robot propels the system using joint torques, ensuring that the total external force acting on the system remains zero, which results in the conservation of momentum across the entire system. Similar to Eqn.(3.19), we can calculate the required instantaneous active robot velocity as

$$\dot{x}_c^{R1} = \frac{1}{3m}(p_{x0} - 3m\dot{x}_c^{R2} - (6m + M)\dot{x}_p)$$
$$\dot{y}_c^{R1} = \frac{1}{3m}(p_{y0} - 3m\dot{y}_c^{R2} - (6m + M)\dot{y}_p)$$

which can then be controlled using the connection.

4.2 Chaotic Dynamics

Note that we can carefully choose the gait executed on the active robot to produce a zero net rotation or displacement per cycle. The gait drawn in Fig. 4.2, for example, produces zero pose change over a cycle. When the active robot executes a gait that has zero net rotation per cycle, it results in exactly the same relative velocity of the active robot to the platform in the world frame, leading to a periodicity induced in the net center of mass of the platform and passive robot.



(a) passive robot joint angles

(b) passive robot position

Figure 4.1: The joint angle and position of a passive robot on a platform moved by an active robot over time. As we can see, after cycles of periodic motion, eventually the passive robot enters a collapsed link configuration. But even after the passive 3-link robot collapses into one link, the active robot can still stimulate position motions in the passive robot.

With the simulated resulting trajectory of platform and passive robot within a time period visualized in Fig. 4.3, we can take samples of the trajectory at multiples of $T = \frac{\pi}{8}$, the time period of the gait executed by the active robot, to obtain a stroboscopic Poincaré section [17]. Initially the points seem to form a closed trajectory, which correspond to the seemingly periodic platform trajectory with a lower frequency, and later disperse into more complicated distributions.

4.3 Multiple Active and Passive Robots on the Same Compliant Platform

Consider when there are more than one passive robots on the platform. It is crucial to remember that, according to the state function (4.1), active robots lack the ability to directly influence passive robots. Their only means of interaction is through the platform; active robots can shake the platform, and in turn, the platform affects the passive robots. However, due to the platform having only two degrees of freedom, our control over passive robots is restricted to two dimensions. This limitation implies that when dealing with multiple passive robots, we can only prescribe the gait of one

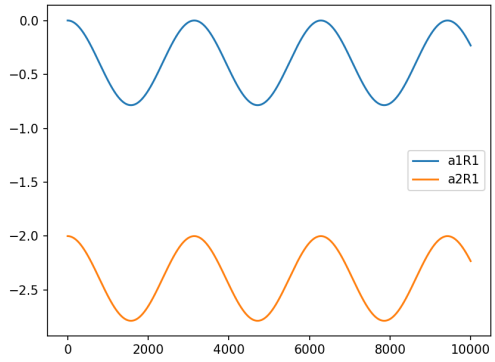
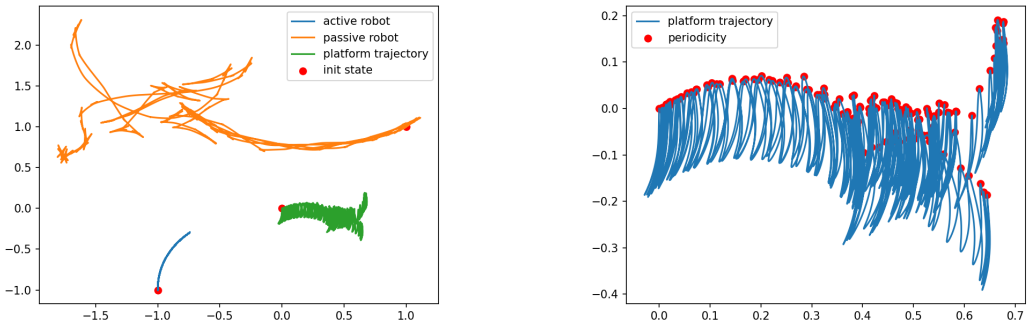


Figure 4.2: A null-homotopic gait that produces zero net pose change per cycle for the active robot.

robot at a time. In theory, it is also possible to control α_1 for two passive robots separately, but this approach may have limited practical utility.

Now, let's consider the scenario with multiple active robots. The motion of these active robots is governed by joint torques, ensuring a system-wide momentum conservation with a net external force of zero. With the knowledge of intended trajectories for passive robots and their corresponding platform trajectories, we can derive two equations from the momentum conservation that imposes constraints on the velocities of the active robots. In the case of a singular active robot, the velocity required to induce the desired gait in the passive robot is uniquely determined. However, when faced with two or more active robots, the velocities become under-constrained. This implies the flexibility to arbitrarily set the x and y velocities of all but one active robot, and then calculate the necessary velocity for the remaining robot to maintain control of the platform. This surplus degree of freedom allows for a strategic trade-off between robot locomotion and mechanical communication for the active robots.



(a) Simulated trajectory within 5000 seconds (b) Poincaré section of platform trajectory

Figure 4.3: Resulting trajectory of the system simulated with the active robot executing the gait in Fig. 4.2.

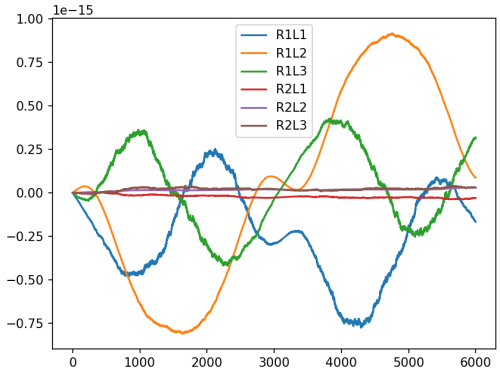


Figure 4.4: For the trajectory in Fig. 4.1, despite minor fluctuations caused by numerical methods, the overall preservation of constraints is evident as they vary on a very small scale.

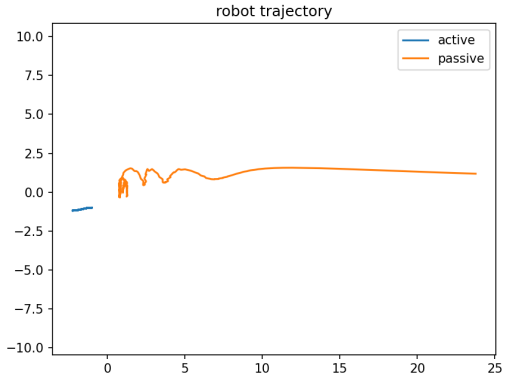


Figure 4.5: This trajectory of active and passive robots on the same platform provides an example of the use of relatively small motions in an active robot (Fig. 4.1) to induce substantial movements in the passive robot. It demonstrates the potential of motion amplification by using an active robot to stimulate the motion in a passive robot.

Chapter 5

Conclusions and Future Work

We studied a multi-agent system comprising 3-link kinematic robots on a movable platform. We first analyzed the individual interactions of each active and passive robot with the platform, and then integrated their behaviors to characterize the dynamics of an entire multi-agent system as a whole. We applied geometric motion planning methods to design gaits for active robots to move a passive platform. And for a passive robot on an active platform, we employed the Lagrange-d'Alembert principles to establish the complete dynamics of the system in the form of equations of motion. We also explored why geometric methods are exclusively suitable in the case of active robots. Through this investigation, we have highlighted the relationship between symmetry and conservation, and emphasized the crucial role of mechanical communication in the locomotion and coordination of multi-agent systems with articulated robots.

The study of 3-link kinematic robots on movable platforms has allowed us to dissect the dynamics of this multi-agent system, revealing the limitations of geometric motion planning techniques and the interaction between robot agents in the same environment. By drawing inspiration from nature's swarming behaviors and information exchange through ambient media, we have explored how articulated robots can interact with their environment to transfer information and achieve coordinated movements. These findings offer valuable insights for designing more sophisticated and practical robotic systems in the future. For example, in the current system that we studied, principally kinematic robots' position relative to the platform depends

only on the joint velocity, which results in the active robots' position not affected by each other. With backdrivable joints, robots with controllable joints can also be affected by the platform motion, making the multi-agent system more complex and intriguing. Moving forward, our work also lays the groundwork for exploring robots navigating intricate fluid environments, opening new possibilities for robotics applications across various domains. By embracing the power of mechanical communication, we can potentially enable multi-agent robot systems to work together seamlessly and efficiently in complex, real-world scenarios.

Bibliography

[1] Ralph Abraham and Jerrold E. Marsden. *Hamiltonian and Lagrangian systems*. Addison-Wesley Publishing Company, Inc, 1987. 2.3

[2] Anthony M. Bloch. *Nonholonomic Mechanics*, pages 235–313. Springer New York, 2015. ISBN 978-1-4939-3017-3. 2.3

[3] Lucian Buşoniu, Robert Babuška, and Bart De Schutter. *Multi-agent Reinforcement Learning: An Overview*, pages 183–221. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. 1

[4] Kearns DB. A field guide to bacterial swarming motility. *Nat Rev Microbiol*, 8 (9):634–644, 2010. 1

[5] Tony Dear. *Extensions of the Principal Fiber Bundle Model for Locomoting Robots*. PhD thesis, Carnegie Mellon University, 2018. 3.1.1

[6] Harley Flanders. Differentiation under the integral sign. *American Mathematical Monthly*, 80:615–627, 1973. 2.2

[7] R. L. Hatton and H. Choset. Geometric motion planning: The local connection, stokes theorem, and the importance of coordinate choice. *The International Journal of Robotics Research*, 30(8):988–1014, 2011. 2.1

[8] R. L. Hatton and H. Choset. Nonconservativity and noncommutativity in locomotion. *European Physical Journal Special Topics*, 224(17-18):3141–3174, 2015. 2.1, 3.1.2, 3.1.2

[9] Ross L Hatton and Howie Choset. Connection vector fields for underactuated systems. In *2008 2nd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*, pages 451–456. IEEE, 2008. (document), 3.1, 3.1

[10] Ross L. Hatton, Zachary Brock, Shuoqi Chen, Howie Choset, Hossein Faraji, Ruijie Fu, Nathan Justus, and Suresh Ramasamy. The geometry of optimal gaits for inertia-dominated kinematic systems. *IEEE Transactions on Robotics*, 38(5): 3279–3299, 2022. 2.2, 3.1.3, 3.1.3

[11] Nagy M. Graving J.M. et al. Li, L. Vortex phase matching as a strategy for

schooling in robots and in fish. *Nat Commun*, 11:5408, 2020. 1

[12] Qingbiao Li, Fernando Gama, Alejandro Ribeiro, and Amanda Prorok. Graph neural networks for decentralized multi-robot path planning. *arXiv preprint arXiv:1912.06095*, 2019. 1

[13] Walczak A. Del Castello L. et al. Mora, T. Local equilibrium in bird flocks. *Nature Phys*, 12:1153–1157, 2016. 1

[14] Suresh Ramasamy and Ross L Hatton. Soap-bubble optimization of gaits. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 1056–1062. IEEE, 2016. 2.2, 2.2

[15] Suresh Ramasamy and Ross L Hatton. The geometry of optimal gaits for drag-dominated kinematic systems. *IEEE Transactions on Robotics*, 35(4):1014–1033, 2019. 2.2, 3.1.3

[16] Elie A Shammas, Howie Choset, and Alfred A Rizzi. Geometric motion planning analysis for two classes of underactuated mechanical systems. *The International Journal of Robotics Research*, 26(10):1043–1073, 2007. 2.1

[17] G. Williams. CRC Press, 1997. 4.2

[18] Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Başar. Fully decentralized multi-agent reinforcement learning with networked agents, 2018. 1