Tensegrity Spines for Quadruped Robots

by

Andrew Preston Sabelhaus

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Mechanical Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Alice M. Agogino, Chair
Professor Andrew Packard
Professor Claire Tomlin
Professor Murat Arcak

Summer 2019

Tensegrity Spines for Quadruped Robots

Abstract

Tensegrity Spines for Quadruped Robots

by

Andrew Preston Sabelhaus

Doctor of Philosophy in Engineering - Mechanical Engineering

University of California, Berkeley

Professor Alice M. Agogino, Chair

Robots that are designed for NASA's missions to foreign planets, as with disaster relief efforts on earth, face tough challenges with harsh environments and locomotion over extreme terrain. Soft-bodied robots could address many of these challenges by conforming and adapting to their environments. This dissertation presents the modeling, design, and control of a set of soft and flexible robotic spines that assist quadruped robots in their locomotion over extreme terrain. These spines are tensegrity systems, consisting of rigid bodies held together in a network of flexible cables, used as a practical method of producing soft behavior.

First, the geometry and possible movements of these spines are discussed. Next, the inverse statics problem is solved for these spines, in order to calculate the tensions of the cables which control the spine's vertebrae. The resulting inverse statics optimization algorithm is tested in a hardware experiment, demonstrating pseudo-static open-loop positioning of the spine. Using this model, a design of a quadruped robot with a tensegrity spine is proposed and prototyped. Simulations show that this quadruped robot's tensegrity spine can lift and position its feet, as a way to assist with locomotion and balance. Hardware experiments validate the simulation's motions of the robot's spine and feet.

Then, control systems are investigated for these tensegrity spines. A set of closed-loop controllers, which use model-predictive control (MPC) in combination with the inverse statics algorithm, are proposed and simulated against dynamics models of the spine. Two different MPC formulations are used, both of which show low-error tracking in simulation.

Finally, given the ongoing challenges with MPC, an energy-based stability criterion is derived for a class of high-dimensional, nonlinear, possibly hybrid robotic systems. These systems, termed 'statically conservative', include networks of cables in tension, similar to tensegrity spines. The stability criterion is applied to these cable networks, giving conditions for stabilizing controllers. An example controller is proposed for a cable-driven robot with slack cables. Simulations of this system and its controller to validate the stability proof. These control approaches show promise for future hardware implementation of walking locomotion in quadruped robots with tensegrity spines.

Contents

List of Figures

List of Tables

Acknowledgments

During my time at UC Berkeley, I've been given the gift of a wide, deep, and diverse network of support. Without everyone who has formed that network, it would not have been possible to do this research and write this dissertation, much less stay here in the San Francisco Bay Area. Here is a brief attempt to thank you all by name. I am equally grateful for everyone who I forget here, and everyone who provided the small interactions that have helped along the way.

For my dissertation committee, a sincere thank you for your time, advice, support, and feedback. My chair and advisor, Alice Agogino, has been the cornerstone of my time at UC Berkeley, and I am infinitely grateful for her guidance, patience, and encouragement. The thoughtful advice of Claire Tomlin, Andy Packard, and Murat Arcak are sincerely appreciated, and their feedback has made this research more clear and impactful. This dissertation is not short, and your time spent reading it will be put to good use when the results are published in improved forms.

I have been fortunate to receive funding from a wide variety of sources during my graduate career. Thank you to the NSF's Graduate Research Fellowship program, which funded my first three years at Berkeley. Thank you to NASA and the NASA Space Technology Research Fellowship program, which funded the remainder of my time here, under grant number NNX15AQ55H. Thank you to the UC CITRIS Tech for Social Good program, which helped fund the hardware needed for the experiments in this dissertation. Thank you to the other funding sources at NASA Ames Research Center, which have provided hardware funds for collaborative work with groups at Ames.

Thank you to all my collaborators at UC Berkeley, in the Berkeley Emergent Space Tensegrities (BEST) Lab and the Berkeley Institute of Design. My labmates have been wonderful co-authors and insightful teammates. Thank you in particular to the other former and current PhD students in BEST, particularly Kyunam Kim, Lee-Huang Chen, Edward Zhu, Mallory Daly, Brian Cera, Jianlan Luo, Alan Zhang, and Andrew Barkan. Thank you very much to the large cast of M.S. and M.Eng students who have worked with me over the years, and to the even larger cast of undergraduate researchers whom I've had the privilege of leading.

Thank you to all my collaborators at NASA Ames in the Intelligent Systems Division. My work with you all during the first years of my graduate career, and our shared space during the latter years with the research in this dissertation, was exciting and rewarding. Thanks in particular to Terry Fong, Jonathan Bruce, Massimo Vespignani, Vytas SunSpiral and Adrian Agogino.

Thank you to all other outside collaborators who have worked with my team over the years. In particular, I'm grateful for the time spent with the Field Innovation Team (FIT), and Desi Matel-Anderson of FIT, while exploring different applications for the quadruped robots proposed in this dissertation. I look forward to continuing to work with you over the years.

The LGBTQ community has supported me throughout my time at Berkeley with advice and mentorship, and I'll be forever grateful for the community's presence in my life. In addition to various other students and staff at GenEq, I would like to thank in particular Bill and Lar from the Markowski-Leach Foundation. I'm honored that you twice chose me for the foundation's scholarship, and that the foundation considered me to be the example that Tom Markowski and Jim Leach hoped for: an 'LGBTQ individual likely to make a substantial contribution to society.'

I want to thank the Disabled Students Program at UC Berkeley for their support, both personally and through advocacy within the university. Without your official documentation, my degree program may not have been completed, or would take a very different form. Without your unofficial guidance, mentoring, personal advice, and motivation, I may not have wanted to complete my degree. Thanks in particular to my counselor, Danny Kodmur.

Thank you, from the bottom of my heart, to my family. Your acceptance and support has made an enormous difference in my life. Though it shouldn't have to be said, many other queer students don't have the luxury of a family like you all. Thank you also to all the older members of my family with graduate degrees: my grandfather, various uncles, and both my parents. You all gave me a vision for what my life might be like during these years at Berkeley, and prepared me to succeed.

Finally, thank you to my partner, Marco Sánchez, whose kind patience made all the difference. Por fin acabamos! Nunca jamás!

Part I

Problem Statement

Chapter 1

Introduction

Of the many ongoing challenges in robotics, one of the most pressing is the use of robots in unstructured environments. Though robots have been successful in a number of domains (for example, manufacturing), significant challenges exist in using robots in spaces that are not specifically designed for their presence. In particular, designing and controlling a robot to move or walk over uneven, extreme terrain is both a need and an unsolved problem in multiple domains.

This dissertation proposes an approach to assisting the locomotion of four-legged (quadruped) robots over uneven terrain. Using the principle of *tensegrity*, this work presents the modeling, design, and control of flexible bodies for quadruped robots. These flexible bodies are referred to here as 'spines' in reference to their biological motivation.

The research in this dissertation seeks to demonstrate that such spines can be made practical in both design and manufacture, with models that match physical experiment, and which can be controlled in closed-loop. It is also demonstrated that quadrupeds with these spines are able to perform movements beneficial to walking locomotion. Along the way to these goals, new approaches to statics modeling of tensegrity structures are presented, as is a new framework for analyzing energy in physical systems with flexible, soft components.

The remaining sections in this introduction provide the problem statement of locomotion over uneven, extreme terrain, and review some current robotics solutions to the problem. The concept of a quadruped robot with a tensegrity spine is then proposed in brief. The chapter finishes with an outline of the remainder of the dissertation.

1.1 Transportation Needs in Extreme Environments

Both on Earth and on foreign planets, there are a variety of environments where human movement and transportation are limited or impossible. In space exploration, data from a variety of sources is highly valued in determining the origins, composition, and colonization potential of different planets. However, sending human operators to collect this data is impractical. Similarly, in disaster situations, large amounts of supplies must be transported to a community to facilitate relief and recovery. However, many common forms of transportation cannot reach such areas post-disaster, and it is left to humans to manually transport goods. In both cases, there are clear potential benefits of robots that move in uneven, unstructured environments and transport equipment and supplies.

1.1.1 Extreme Terrain on Foreign Planets

The US National Aeronautics and Space Administration (NASA) has long used robotics technology to explore worlds inaccessible by humans, dating back to the 1970s with the Voyager probes [184]. In modern times, space agencies send mobile robots to the surface of planets. NASA's most recent of such missions include the Mars Science Laboratory rovers, Opportunity [183], Spirit [13], and Curiosity [83]. Valuable data from these experiments has informed our world's understanding of the composition of the Martian surface, radiation properties, and past hydrodynamics.

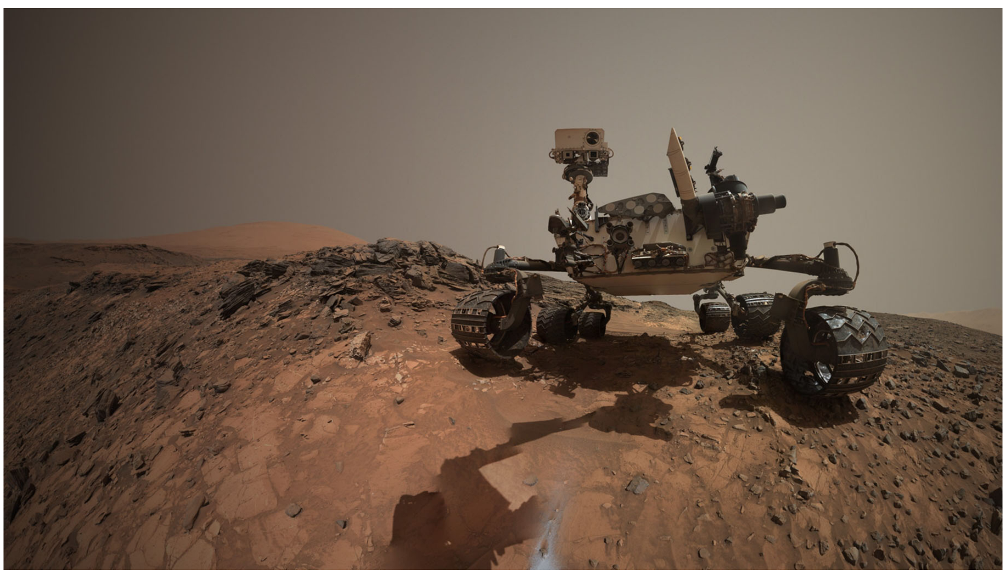


Figure 1.1: The Mars Curiosity Rover, in one of its more aggressive motions on rocky terrain [99]. The limitations of wheeled robots become evident in comparison to the variety of rocky surfaces on planets such as Mars.

Each of these rovers has employed a wheeled base to move around on the planet's surface. For example, Mars Curiosity (Fig. 1.1) has six wheels with a suspension that allows it to move on inclined surfaces. However, these robots are all limited in the regions they can traverse, due to the inherent issues with wheeled locomotion. Fig. 1.1 demonstrates a particularly aggressive slope and terrain for this robot, which was restricted to smoother, flatter surfaces for most of its life cycle, similar to its counterpart Spirit [13]. Even in these conservative motions, the robot still experienced a number of hardware failures in its wheels due to the terrain [13]. These limitations are common to wheeled rover designs [173].

A wider array of science return is available in areas that are neither smooth nor flat, such as river beds and lava tubes. Transportation of the bulky equipment required for scientific measurements (e.g., mass spectrometers) in these spaces has prompted NASA to examine other means of locomotion (Sec. 1.2.2.) Nevertheless, no other design concept than wheels has been flown by NASA.

1.1.2 Extreme Terrain in Disaster Relief Areas

In addition to locomotion and transportation needs in space, challenges exist with extreme terrain here on Earth. Humanitarian crises cause significant loss of human life, much of which occurs due to lack of supplies during and after the event occurs ([94], p.179). Disasters often significantly disrupt transportation capabilities ([94], p.43) just as transportation is needed the most. Supplies such as fuel, lighting, communications, and medical equipment are needed by the relief workers that serve the area, as are fresh water [200] and cooking fuel [71] for survivors.

However, current options to bring aid into a disaster area are limited by the ability of traditional vehicles to access the area, potentially creating crises of food, water, and healthcare in regions that cannot be accessed. All-terrain trucks are limited by wheel base and axle height in the same way as wheeled robots in space [147]. Airdrops by helicopters can be unpredictable and dangerous [80], and robotic drones have relatively small carrying capacities [4] and so are limited to lightweight payloads. More widely-used are pack animals [160], which cannot operate in the same way as machines.

Figure 1.2: A street in Chobhar, Nepal after two earthquakes struck the region in April 2015. Credit: Desiree Matel-Anderson, Field Innovation Team.

Therefore, it is most common for human relief workers to transport supplies manually, and much disaster equipment is specifically designed to be carried by people [201]. Fig. 1.2 shows one such situation, courtesty of the Field Innovation Team [65], where wheeled vehicles cannot pass through a street after an earthquake. Disaster relief experts anticipate other earthquakes to cause damage in the United States in the near future, including along the Cascadia and Seattle fault lines in the Pacific Northwest as well as along the San Andreas fault in the San Francisco Bay Area.

In the same way that robotic rovers have been used in space for decades, the idea of using robots to help in disaster relief situations is not new. Entire books have been written on the topic [138], and nascent organizations such as Roboticists Without Borders are being formed to oversee their implementation [137]. However, most of these uses have involved search and rescue, not transportation. Sec. 1.2.1 below reviews these solutions.

1.2 Robots in Extreme Environments

The issues identified above are well-known in the field of robotics, and many proposed solutions exist. The following section reviews those proposed solutions, and identifies the particular gap which the work in this dissertation seeks to address: reliable transportation of supplies and equipment over unknown and uneven terrain.

Chapter 2 reviews the literature related to this dissertation's goals: quadrupeds, spines, and control. This section instead reviews other potential robotics solutions to the issues identified above.

1.2.1 Search and Rescue Robots

The field of robotics has traditionally focused on search and rescue for research in the context of disaster relief. As differentiated from transportation of supplies during rebuilding, search and rescue refers to an immediate response to a disaster, within the first few days, where survivors may be trapped or in danger [138]. As a result, these robots are more often designed solely for locomotion over or through uneven terrain, without regard to carrying equipment or supplies.

In particular, tank-style robots are those most commonly used for search and rescue [139]. However, these robots are mostly small [59] and suffer from the same inherent issues as other wheeled vehicles: they still cannot climb over large rocks or gaps. On the other hand, more novel search and rescue robots are very small, and designed more to navigate inside rubble than to carry supplies over it. Robots such as the VelociRoach [82] and Salamandra [51] can move over uneven terrain, but exhibit potential limitations for use in transporting large payloads or sensitive equipment. However, these modes of locomotion may be adaptable to transportation in future work.

1.2.2 Flying and Hopping Robots

While disaster relief robotics has focused on identifying and assisting survivors, space robots for uneven terrain locomotion have explored more extreme methods of locomotion that would otherwise be inadvisable in sensitive disaster situations. In particular, robots that combine flying and hopping may locomote effectively over uneven terrain, but have significant trade-offs.

Hopping robots can use inertial flywheels [89], thrusters [107], or other methods to make rapid movements from one location to another. These robots have significant challenges aiming their motions [88], and risk damaging surrounding environments in sensitive situations as a result. For space applications, concerns may also exist in transporting sensitive equipment.

In addition to the previously-stated small payload capacity of flying robots in disaster relief settings, more challenges exist in space applications. In particular, Mars has a less dense atmosphere, making flight difficult [177], but still possible. These solutions are not considered here for the same reasons as with disaster relief.

1.2.3 Walking Robots

Walking locomotion is generally energy-efficient, does not involve unstable hopping, nor has the size and terrain limitations of treads and wheels. Walking robots can have a variety of different leg configurations and walking gaits. Robots with two legs (bipeds) for locomotion over rough terrain have been recently emphasized by national competitions such as the DARPA Robotics Challenge [53]. However, these robots are inherently unstable and require active control at all times, and often fall or collapse in field tests [2]. Robots with six legs (hexapods) or greater have more stability in comparison to two, and can locomote over uneven surfaces even in open-loop [134]. However, the additional degrees of freedom with each leg make closed-loop control more challenging.

Quadruped (four-legged) robots represent an efficient trade-off between stability and complexity. Four legs are the minimum number required for pseudo-static walking gaits. It is for this reason that this dissertation proposes the following concept.

1.3 Proposed Solution

This dissertation develops a flexible, actuated spine for quadruped robots that can assist in a robot's locomotion over uneven terrain in disaster relief and space applications. The spine is used as part of an early prototype of the robot Laika, named after the first dog in space (Fig. 1.3). Laika's spine is a *tensegrity* structure (discussed in Chap. 2.3), consisting of rigid vertebrae held together in a network of flexible cables. These cables are actuated by motors, changing their lengths and forces, which moves the spine's vertebrae.

Using a flexible tensegrity structure as part of a quadruped robot could address many of the challenges that are found in other quadrupeds (Sec. 2.1), while simultaneously taking advantage of the benefits of quadruped locomotion gaits. Laika's spine is inherently flexible, allowing for passive compliance with uneven terrain, while also assisting the robot in lifting and moving its feet (Chap. 6).

Though designing a robot with such a spine is appealing for multiple reasons, significant challenges also arise. Before this dissertation, no work existed that could model such a spine either statically or dynamically. No practical hardware designs existed, nor were there sys-

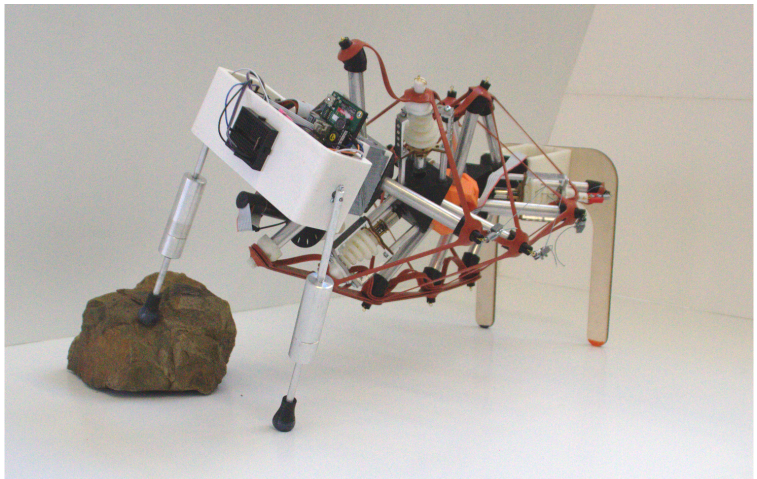


Figure 1.3: Prototype of the quadruped robot Laika, which incorporates an actuated tensegrity spine as its body. Laika's flexible spine allows it to passively balance on terrain such as the rock in this example. Current prototypes of Laika do not yet walk, as this dissertation focuses on Laika's spine, not locomotion.

tems to control the spine's movement in closed-loop. Each of these challenges are addressed, in order, in subsequent chapters.

1.4 Dissertation Outline

This dissertation is split into three parts. The first part, an extension of this introduction, addresses the problem statement and background literature for quadruped robots with tensegrity spines. The second addresses the kinematics, statics, and mechanical design of such spines, since these concepts will be shown to be inherently intertwined. The third and final part addresses dynamics models and control systems for both this spine and related control problems, presenting solutions to the major difficulties with this spine's system.

Part I contains this introduction, its continuation to a literature review in Chap. 2, and a review of Lagrangian dynamics and passivity-based control in Chap. 3. Particular emphasis is placed on statics modeling and energy-based robotics control, since this dissertation contributes progress in these two areas in addition to the spine itself. Chap. 3 reviews key theorems in prior work related to dynamics and energy-shaping control, which are needed for the new contributions in Chap. 8.

Part II presents the kinematics, statics, and design of tensegrity spines. While previous investigations into tensegrity robots of similar types relied upon either model-free or adhoc design methods, these chapters instead take a model-based and physically-validated approach. In particular, Chap. 4 performs a bio-inspired investigation into spine geometry, Chap. 5 derives the first mathematical models of these spines, and Chap. 6 presents the first designs for tensegrity spine prototypes based on these mathematical models. Chap. 5 also presents a new inverse statics optimization algorithm for these spines, and shows that the results can be used for open-loop control. These results demonstrate that tensegrity spines can be modeled accurately and designed efficiently, and that their motions can be beneficial to quadruped robots.

Part III presents dynamics models and the first closed-loop control systems for these spines. As with Part II, these controllers are model-based as opposed to ad-hoc or hand-tuned. Chap. 7 uses model-predictive control (MPC), in combination with the inverse statics algorithm from Chap. 5, as part of multiple closed-loop controllers for some example spines. Then, given the limitations of the MPC controllers, Chap. 8 presents a new framework for energy-based control of robots that have *statically conservative* forces, a phenomenon present in the slack cable dynamics of tensegrity spines. Finally, Chap. 9 discusses the future work for this project, including walking prototypes of quadrupeds with these spines, locomotion studies, scaling up soft robots to compete with larger rigid robots, manufacturing and sensing challenges, and the wide applications and extensions of the proposed energy-shaping framework.

Chapter 2

Prior Work

The work in this dissertation spans many disparate domains in robotics modeling, design, and control. The following chapter discusses the background and reviews the literature in these domains. Relevant prior work in each area - quadruped robots and locomotion, robotic spines, tensegrity structures, and control - are reviewed in order.

Taken in whole, this chapter suggests that ongoing challenges in each sub-field can be addressed by particular combinations and contributions of the approaches presented here. Robotic locomotion challenges could be addressed by constructing systems with soft components. The intimidating issues with soft robotic spines could be addressed in turn by the use of tensegrity designs. And, ongoing challenges in modeling, design, and control of tensegrity robots can be addressed in part by the contributions in this dissertation.

2.1 Quadruped Robots and Locomotion

As discussed briefly in Sec. 1.2.3, quadruped (four-legged) robots exhibit a number of desirable qualities for locomotion, particularly over uneven terrain. A large number of quadruped robots have been designed for different types of locomotion. This section reviews those robots and ongoing challenges.

One major differentiation of quadruped robot designs is the robots' purpose. Many are focused on fast locomotion over flat terrain, i.e., running [174, 93], without or without spines to assist in 'galloping' gaits [103, 150, 62]. These running robot serve a very different purpose than Laika, which is intended for reliable, safe transportation of goods.

The other class of quadrupeds focuses on walking over uneven or uncertain terrain [73, 72, 110, 148, 157, 159, 33, 87]. Such terrain could be as diverse as the stairs inside buildings [87, 182, 1] or rocky planets. However, balancing and locomotion over large obstacles can be challenging for robots built with rigid structures that cannot conform to the environment, limiting them to obstacles that are small in comparison to their total size [75, 72].

Such a limitation is evident in the work of Boston Dynamics, whose BigDog robot is arguably the leading quadruped robot for transportation [159]. Fig. 2.1 shows a variety

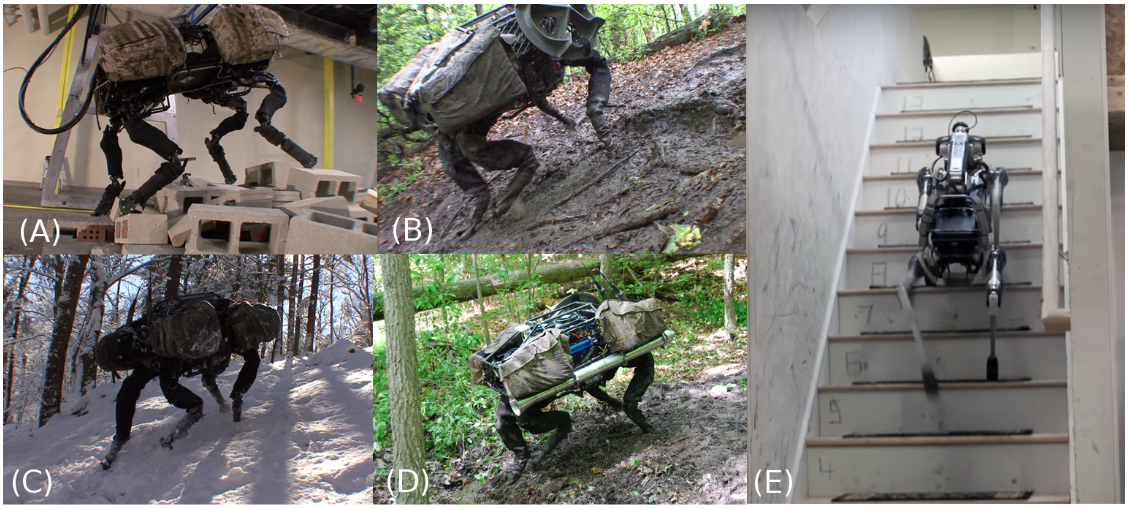


Figure 2.1: Various walking four-legged (quadruped) robots by Boston Dynamics, the leader in this field. Right-to-left, top-to-bottom, the BigDog robot walks over rubble (A), mud (B), snow (C), and a dirt path (D). In (E), the SpotMini robot walks up stairs. With the exception of the rubble in (A), all these terrains are mostly-flat slopes (B-D) or environments that only tip the robot up and down, not left-to-right (E). In the only truly uneven test, the rubble in (A), the robot is tilted to the left as it walks: since BigDog cannot distribute its weight between its shoulders and hips using a spine, it is less balanced. Images from [29].

of generations of BigDog, and its successors. By using many-jointed legs and advanced control systems, BigDog can traverse snow (Fig. 2.1(C)) and sloped, smooth dirt paths (Fig. 2.1(D)).

However, BigDog is constructed with a rigid chassis as its body, and exhibits somewhat surprising behavior as a result when locomoting over very uneven terrain. For example, Fig. 2.1(A) shows a large swaying motion of the robot as it walks over a pile of cinderblocks, also noticeable when slipping in mud (Fig. 2.1(B)). These motions presumably arise from the large moments between BigDog's shoulders and hips as its legs contact different heights of terrain: the robot must make large movements to place all four of its feet in contact with the ground. Though BigDog does still traverse these terrains, questions of the stability of gaits arise. The proposed solution in this dissertation incorporates a flexible, actuated body for precisely this purpose.

Boston Dynamics' ongoing work appears to address these issues of balance by shrinking the size of the robot's body, and widening its stance, with the new SpotMini (2.1(E)). Though doing so does allow the robot a wider range of motions, simply shrinking the robot's size counteracts the goals of transportation: the original BigDogs carried payloads, whereas SpotMini does not. This suggests an inherent limitation on rigid chassis for robots, where design tradeoffs dictate stability versus carrying capacity.

2.2 Spines for Quadruped Robots

One method to address these limitations, then, is to incorporate a flexible body into the robot, such that the weight distribution and leg placement can be partially performed by the body itself. This can be done with the goals of faster locomotion [174, 62] or other tasks such as turning [202], but which do not necessarily assist in uneven terrain settings. Robots that do incorporate spines for balance purposes can either use passive or active spine designs.

Passive spines assist in balance by conforming to the environment under the robot's legs, without the need for active control [190, 100]. Such designs have been demonstrated on inclined surfaces, for example. However, using a passive spine in the robot's body comes with significant modeling and control challenges, since the robot's legs may not be able to move into a desired position during a gait cycle. Extremely uneven terrain, such as large rocks, suggests an actively controlled spine.

Spines that are actively controlled for purposes of balance have traditionally incorporated rigid joints [20, 21]. Though this assists in positioning the robot's feet, the lack of compliance makes adapting to the environment challenging in other ways. Additionally, design issues arise. When the robot needs to rotate its spine, for example, all the torque for this rotation must pass through every joint in the spine and every actuator [20, 21], requiring heavy mechanisms with large actuators, and subsequent inefficiencies. Amplified moment arms through rigid joints when encountering large obstacles often limit mechanical designs in this way [188].

2.3 Tensegrity Robots

A solution to the all these issues of compliance, active control, force distribution, and design efficiency can possibly be found in the principle of *tensegrity*, or 'tension-integrity', structures. The definition of the word tensegrity can be vague in the literature. Here, the following is adopted:

Definition 2.3.1. Tensegrity. A tensegrity structure consists of rigid bodies suspended in a network of cables in tension such the bodies do not contact each other [179].

This definition is sometimes restricted further to structures without internal bending moments, i.e., where the bodies are single bars and cables only connect at bar ends [135, 128, 26]. However, both historical examples (Snelson's 'X-Cross', [179]) and many modern robots [130, 70, 27, 69, 46] use the broader definition, having more complicated bodies suspended in the network, as with the vertebrae of these spines.

Tensegrity structures are differentiated from other cable-driven robots by the lack of contact (either frictional, or through other types of joints) between their bodies. Instead, cables must hold each body apart. The results can be striking and counter-intuitive (Fig. 2.2). Among the variety of benefits to using tensegrity principles when designing structures are:

1. Shape Change. By adjusting the tensions on the structure's cables, for example by retracting or extending the cables, the structure can change shape.

2. Force Redistribution. Since there are many load paths through the structure's cables, there is no single mechanism that carries the whole load applied to any individual element. Therefore, the rigid elements can be more lightweight, with lower internal stresses.

3. Adjustable compliance. A tensegrity structure can be made more stiff, or more flexible, without changing shape. Chap. 5 takes advantage of this 'pretensioning' property for control.

There are many robots which take advantage of these properties. These robots are able to adjust the lengths of their cables to roll [109, 167, 106, 48, 162, 199] (Fig. 2.3a), crawl [153, 175, 194, 130, 214] (Fig. 2.3b), swim [24, 46], hop and jump [107, 127], and climb [68, 70]. Most of these examples involve locomotion: the tensegrity structure's shape change is designed to propel it along the ground or through a medium. Tensegrity robots also called 'spines' have been previously investigated [129, 131, 130], but these may more properly be named 'snakes', since they perform this type of ground locomotion.

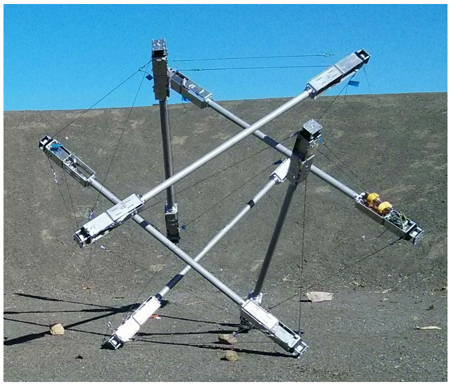
Figure 2.2: Kenneth Snelson's "Dragon", a tensegrity structure, stiffly cantilevered over a lake [181].

There has not yet been a tensegrity spine used as part of a larger robot in walking locomotion, such as is proposed here. Other work has proposed similar ideas [92], but differ in purpose and goal for their use.

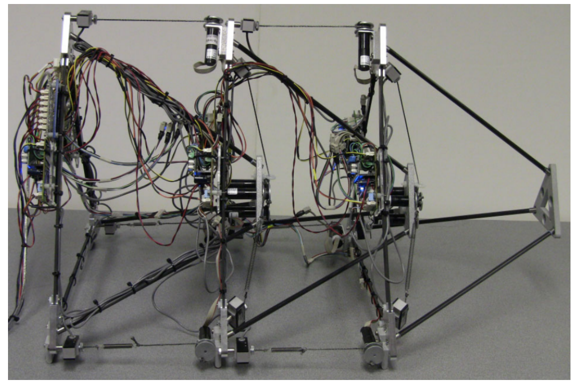
2.4 Tensegrity Robot Modeling: Kinematics and Dynamics

Since these structures by definition do not have position constraints on their bodies, their kinematics, statics, and dynamics can be expressed in relatively general formulations. This does not necessarily make it easy to design or control tensegrity robots, nor do prior mathematical models apply to all tensegrity robots. This dissertation contributes new static and dynamic models not present in the literature.

Most models for tensegrity structures begin by assuming that the structure can be represented as a graph, with force-carrying elements (bodies in compression or cables in tension)



(a) A spherical tensegrity robot designed for rolling locomotion [167].



(b) A snake tensegrity robot designed to crawl along the ground [131].

Figure 2.3: Two example tensegrity robots. (A) A spherical robot designed to roll by changing its cable lengths, consisting of only bars with cables connected at nodes. (B) A snake-like tensegrity robot designed to crawl, which has more complicated bodies suspended in its network, which cannot be easily modeled by the techniques in the seminal work of [179].

represented as edges that exist between nodes where they connect [179]. This model, similar to a truss in civil engineering, currently only allows for pin joints at nodes (no internal moments.) However, the graph structure can still represent the geometry of most systems (discussed more in Chap. 5).

Using such a formulation simplifies the kinematics and statics of tensegrity structures. This is the *form-finding* problem, and has a variety of well-known solutions. Form-finding is the process of simultaneously solving for a pose of the tensegrity structure's bodies alongside the cable forces that keep it in equilibrium [193]. A subset of this problem is solving for the cable forces in static equilibrium for a given pose. For related parallel robots, the former problem is termed *inverse kinetostatics* analysis [22, 16], and the latter subset is *inverse statics* analysis [12, 76].

For tensegrity robots like these spines, no position constraints exist on the bodies, decoupling their statics and kinematics and avoiding the need for approaches common in the parallel cable-driven robot literature such as [74, 12, 16]. This allows for the easier-to-solve *force-density* method [172, 193, 195] to be applied. However, solutions have yet to exist in the literature for tensegrity robots with internal bending moments, which occur in the spines considered here.

As mentioned above, much research on tensegrity structures and robots assumes that the bodies suspended in the structure are single 'bars', with cables only attached at endpoints. Fig. 2.2 demonstrates this "bars-only" design principle. For structures like this, equations of motion are well-known [179]. By assuming the single rods are infinitely thin, and by a clever parameterization of the states of the bars, equations of motion can be written succinctly.

For other tensegrity robots, such as the spines considered here, those models either require modifying the body's shape or else do not apply. Chapter 7 gives an alternative dynamics formulation for the spines in this paper, though it lacks some of the benefits of the method from [179].

2.5 Tensegrity Robot Control

Although there are a variety of benefits to using tensegrity structures as robots, control of such structures has proven challenging. Their dynamics are inherently nonlinear and often high-dimensional. Various saturation issues also exist, as cables within the structure provide no force in compression, and the controller cannot retract any cable to a negative rest length (discussed in Chap. 7.) Consequently, state-space control for tracking or regulation has been mostly limited to low-dimensional structures, particularly those with only bars [6, 210, 178, 179, 128], which assume, a-priori, that all cables are in tension at all times. Open-loop methods have also been used for this purpose [186, 187, 37, 68], but cannot reject disturbances.

However, extensive closed-loop control strategies have been developed for high-dimensional tensegrity robots with other control goals. In particular, when the robots are intended to roll or crawl, model-free controllers have used evolutionary algorithms [153, 95, 96, 102], central pattern generators [131, 130], Bayesian optimization [162, 108], deep reinforcement learning [215], kinodynamic motion planning [115], or hand-tuned algorithms [198]. Model-predictive control has been used for generating locomotion primitives as "expert" supervision for imitation learning of an end-to-end locomotion policy [45]. Tensegrity structures which oscillate, such as fish tails, have used resonance entrainment [26, 24]. Though these approaches are promising in their domains, they do not necessarily apply to state tracking, as is needed for precision movements of a spine as part of a more complicated gait.

2.6 Control Techniques for High-Dimensional, Nonlinear, Hybrid-Dynamics Robots

Controlling a robot with a tensegrity spine could be done with a variety of approaches, including directly attempting to control the whole robot in a specific gait. This dissertation focuses on control of the spine alone. It is assumed that by demonstrating control of the spine as a subsystem, closed-loop walking will be made simpler. Since it is somewhat unknown what types of motions would benefit walking locomotion (some discussed in Chap. 4), the state tracking problem is studied here, as it is the most extensible to different situations.

Many options for control exist in the literature for state tracking in robots with high-dimensional, nonlinear dynamics. When the robot's dynamics are also hybrid, for example in the case where a robot's cables can become slack, the problem becomes even more challenging. A brief review of some common approaches is given in this subsection, justifying the use of model-predictive control (MPC) in Chap. 7. The last section in this chapter, sec. 2.6.6, is dedicated to a more thorough discussion on the other approach in this dissertation, that of passivity-based control, since it requires somewhat more nuance than MPC.

2.6.1 Feedback Linearization

One approach to nonlinear control is that of feedback linearization, where a set of nonlinear dynamics is transformed (in a one-to-one manner) into a set of linear dynamics. This is done for both walking robots [132] and tensegrity structures [6]. However, doing so is only practical for systems with a similar number of inputs as states; otherwise, numerical issues arise with taking many derivatives of the same variable. Since the robots considered in this dissertation are highly under-actuated, feedback linearization would not be indicated here.

2.6.2 Lyapunov Function Finding / Sums-of-Squares

A more modern approach to control of highly under-actuated robots, including those which have hybrid dynamics by nature of walking locomotion, is the use of various Lyapunov techniques to generate controllers. Control Lyapunov Functions [8] can be used in certain cases, for example, though solving for control laws can be challenging. Similar approaches include the use of trajectory libraries [121]. In these cases, robots are modeled with the lowest number of degrees of freedom possible, for tractability, and are still significantly lower-dimensional than the spines considered here.

Addressing computational complexity issues can be done via various sums-of-squares programming techniques for Lyapunov function finding [120] or other optimization techniques to make problems convex [122] or lower-dimensional [119]. These are possibly the most promising approaches in the literature for robots such as the spines considered here. However, adapting these techniques on a new system (such as the spine) is a significant undertaking, one which may detract from the proof-of-concept goals in this dissertation. Implementing these approaches is left for future work.

2.6.3 Model-Predictive Control

Model-Predictive Control (MPC) is an optimization-based approach to control, commonly used to address some of the challenges experienced in systems such as these tensegrity robots. For example, using an optimization program for control can address constraints on the system (in the case of the tensegrity spine, actuator saturation and tensioned cables). In addition, computational tractability of an optimization-based controller can be addressed by using a receding horizon. These two features combined become the definition of MPC. Finally, an MPC formulation allows straightforward introduction of smoothing weights and constraints for hard-to-control systems [209, 63].

Model-predictive control for nonlinear systems (NMPC) is a well-studied topic with many implementations [7], particularly in low dimensions where nonconvex optimization is computationally possible [209]. For high-dimensional nonlinear systems, practical options include modifying the NMPC problem or using more efficient solvers [39]. Alternatively, linearized dynamics can be used at later points in the horizon [217], or linearizations can be performed at each timestep in the problem to create a linear time-varying MPC [63].

2.6.4 Model-Free Approaches and Machine Learning

As with tensegrity robots, control approaches that use machine learning (with or without models) have recently seen a strong emergence in robotics. Entire books are written on the subject [67, 54]. However, as per the discussion on NASA's needs for space, as well as the inherent sensitivity of disaster situations, controllers without stability proofs may be of concern. Most model-free techniques, and most machine learning approaches, do not come with stability proofs, nor can easily be extended to do so. Therefore, these approaches are left for future work, and model-based approaches are the focus of this dissertation.

2.6.5 Other Approaches

There are an enormous number of other approaches to nonlinear, high-dimensional, hybrid control. These include using reachability for hybrid systems to develop controllers [118], or controlling Gaussian Process-based models of robots [60]. It is impractical to list all potential options here. A current survey for tensegrity robots in particular can be found in [111].

2.6.6 Passivity-Based Control of Lagrangian Systems

In addition to the approaches mentioned above, one of the more recent control approaches for mechanical systems is that of passivity-based control (PBC). Using the concepts of *passivity* and *dissipativity* for control can address both nonlinearities and high number of states in a problem. Passivity-based control examines the energy in a system, potentially bypassing parts of the nonlinear dynamics. And, by taking advantage of interconnection properties of passive systems, a larger system can be separated into individual subsystems of lower dimension, while satisfying stability properties [10]. For these reasons, this dissertation proposes a form of energy-shaping control, motivated by passivity, in Chap. 8.

Passivity-Based Control of Robotic Systems

The use of an energy-based passivity analysis for control is a well-studied topic, first appearing in the 1980s [145]. In this framework, the total energy of a robotic system is used as a storage function (see Defn. 3.2.1) and a Lyapunov candidate. Various proposed controllers have been developed that extend this idea to adaptive control [31], output feedback [143], underactuated systems [165] and impedance control of manipulators [5], among others.

The approach of passivity-based control has been particularly successful with flexible-link robotic manipulators [146] or flexible cable-driven robots [43, 44]. Of particular interest are systems with viscoelastic joints have been stabilized by PBC [101]. The cables of the robots in this dissertation can be considered 'viscoelastic' in the sense that their tensioned-or-slack state is determined by a combination of length and stretch rate. However, the approaches in all the above assume certain forms of applied forces, such as affine-in-the-control-input dynamics [101]. The above, then, do not necessarily apply to the tensegrity spine robots studied here.

Vehicle Platoons and Equilibrium-Independent Dissipativity

The systems studied in this dissertation cannot necessarily be placed in the framework of [144, 32], due to issues with the systems' total potential energy. In the cases considered in Chapter 8, the potential energy is unbounded, and the Lyapunov candidate of the system's total energy is therefore not a Lyapunov function. Yet, equilbria clearly exist in some cases.

One promising method to address this difficulty is that of equilibrium-independent dissipativity (EID) [86], which allows for interconnections of systems without knowledge of a system-wide equilibrium point. In particular, any individual subsystem does not necessarily need to have an equilibrium; instead, it must simply be possible that an equilibrium exist upon application of a particular input.

Such an approach has been used to decouple other types of flexible robots, such as UAVs supporting a payload with flexible cables in Meissen et al. [125]. This is very similar to the tensegrity structures considered in this dissertation, where individual bodies are connected with flexible cables. In Meissen et al.'s work, some subsystems (such as a suspended payload) also had unbounded potential energy, yet control was addressed through EID and interconnections. The approach proposed in Chapter 8 differs from [125] in a variety of ways, and has different benefits versus drawbacks. These comparisons are discussed in-line later, after presentation of the new approach.

Chapter 3

Review of Lagrangian Dynamics and Passivity-Based Control

Controllers for the various mechanical systems studied in this dissertation will be model-based; that is, using the system's equations of motion. Newton's laws could be used to derive these dynamics equations. However, a more powerful form, one that lends itself to an energy-focused consideration of mechanics, will be used instead. That approach is the use of Lagrange's Equations.

The following chapter presents Lagrange's equations as background, in combination with the concepts of passivity and dissipativity. The technical details here become relevant when investigating the techniques in Chapter 8. This chapter includes a variety of theorems from prior work that are needed later. These topics are reviewed here in order to dedicate Chap. 8 to the new results without extraneous content.

3.1 Lagrange's Equations

For a physical system governed by Newtonian physics, an energy balance can be used to write the governing equations of motion (or of continuous time response behavior.) Traditionally, the mechanics/dynamics literature refers to the result as Lagrange's equations, whereas the electromechanical systems community (and control systems community) refers to them as the Euler-Lagrange equations. Both terms are equivalent for the discussion here.

3.1.1 A Variational Approach to Lagrange's Equations of Motion

The following considers what is termed the *variational* approach to Lagrange's equations [77], Ch.2. This approach uses the calculus of variations to define Lagrange's equations, without a discussion of bases or geometry. Such a discussion of bases and geometry will be temporarily deferred to later subsections.

Consider a system which is parameterized by a a set of *generalized coordinates*, $\{q_1 \ldots q_n\}$. Each generalized coordinate is an element of a subset of the real numbers, $q_j \in \mathcal{X}_j \subseteq \mathbb{R}$. Similarly, define their time derivatives as the *generalized velocities*, $\dot{q}_j \in \mathbb{R}$. There will be a quantity called the *generalized force* acting on the system, the interpretation of which is application-dependent. The generalized force will be defined according to its components in the direction of each coordinate, $Q_j \in \mathbb{R}$.

Let the system's total kinetic energy, potential energy respectively, be denoted by T and U which are implicitly functions of the generalized coordinates and generalized velocities:

$$T \in \mathbb{R}, \qquad U \in \mathbb{R}$$

Both are scalar-valued functions. Define the quantities called the *Lagrangian*, L, and the *Hamiltonian*, H, as either the difference or sum respectively of the kinetic and potential energy:

$$L := T - U, \qquad H := T + U$$

Then, the following set of equations, referred to as Lagrange's equations, hold based on a the use of Hamilton's principle from the calculus of variations. These are equivalent to Newton's laws of motion.

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \left(\frac{\partial L}{\partial q_j}\right) = Q_j, \qquad j = 1 \ldots n. \tag{3.1}$$

This forms a set of n equations, one for each coordinate. In the future, summations over all indices j (Einstein notation) will occasionally be implied, and will be clear from context.

There are significant benefits to using Lagrange's equations over Newton's laws in many contexts, including the ease of incorporating constraints, automatically dealing with signs of forces, creating distinctions between conservative and nonconservative generalized forces, and allowing general treatments of curvilinear systems in $\mathbb{E}^3$ and related manifolds [142]. However, the most powerful property in the control systems context is the passivity and stability analysis that follows easily from the use of energy to derive the equations.

3.1.2 Vector Form of Lagrange's Equations

Stability analysis for control (e.g., by Lyapunov methods) requires considering a single signal in a vector space. In multibody Lagrangian systems, the equations of motion are instead derived in terms of individual position vectors for each particle (and for rigid bodies, rotation tensors for each body.) One method to extract a single vector may be the use of the *representative particle*, either for systems of K particles in $\mathbb{E}^{3K}$ [40] or for systems of K rigid bodies in $\mathbb{E}^{12K}$ [41, 42]. Instead, for the analysis here, the generalized coordinates, velocities, and forces are simply concatenated into vectors of real numbers; behavior is then interpreted geometrically after the proofs. Such an approach maintains the purely analytical interpretation of Lagrange's equations, allowing for a more diverse set of applications where a geometric interpretation is not indicated (for example, to electrical systems.)

So, define an n-dimensional vector consisting of the generalized coordinates, as in

$$\mathbf{q} := [q_1,\ q_2,\ \ldots,\ q_n]^\top \quad \in \mathcal{X} \subseteq \mathbb{R}^n. \tag{3.2}$$

Similarly for the generalized velocities,

$$\dot{\mathbf{q}} := [\dot{q}_1,\ \dot{q}_2,\ \ldots,\ \dot{q}_n]^\top \quad \in \mathbb{R}^n, \tag{3.3}$$

and for the components of the generalized force,

$$\mathbf{Q} := [Q_1,\ Q_2,\ \ldots,\ Q_n]^\top \quad \in \mathbb{R}^n. \tag{3.4}$$

The notation $\mathbf{q}$ will be used interchangeably to refer to the set $\{q_1, \ldots, q_n\}$ as well as the element of $\mathcal{X}$, for example when used as an argument to a function. Finally, the following notation will be used for partial derivatives with respect to a vector in $\mathbb{R}^n$,

$$\frac{\partial U}{\partial \mathbf{q}} := \left[\frac{\partial U}{\partial q_1}, \ \ldots, \frac{\partial U}{\partial q_n}\right]^\top \in \mathbb{R}^n. \tag{3.5}$$

It is important to note that this is not the geometric definition of a gradient in Euclidean space, which will be discussed shortly. Then, Lagrange's equations could be equivalently expressed as

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \left(\frac{\partial L}{\partial \mathbf{q}}\right) = \mathbf{Q}. \tag{3.6}$$

This is *not* a generally-accepted form of Lagrange's equations in the dynamics literature, and does not appear in pervasive texts such as [77]. However, most past work on passivity-based control using Lagrange's equations [144, 32, 197] considers the form of eqn. (3.6), and so both the coordinate-wise and vectorized forms will be interchanged throughout this dissertation.

3.1.3 Definition of Conservative Generalized Forces

In eqn. (3.6), the concept of kinetic energy is usually well-defined and is clear for a variety of systems (for example, systems of particles.) However, U is less clear, and requires a discussion on conservativeness of a generalized force.

Definition 3.1.1. Conservative Generalized Force. A generalized force represented by its components $\{Q_1,\ \ldots,\ Q_n\}$ is *conservative* if each component can be expressed as a partial derivative of a scalar field that is only a function of the generalized coordinates,

$$Q_j = -\frac{\partial U}{\partial q_j}, \qquad j = 1,\ \ldots,\ n. \tag{3.7}$$

This definition must be met for all coordinates and all components ($\forall\ j$.)

Remark. Later, for an analysis of dissipativity in the control-systems usage of the term, an opposite sign convention will be chosen, equally valid for the definition of conservation:

$$\mathbf{Q} = \frac{\partial U}{\partial \mathbf{q}}.$$

The motivation for this convention will become clear in section 3.3.2. It is unfortunate that the two communities of dynamics and control require these different conventions; however, this dissertation will be consistent in the application of one definition versus the other depending upon context.

In eqns. (3.1) and (3.6), the quantity U is the system's potential energy. The potential energy is therefore defined as the U from all generalized forces which meet Defn. 3.1.1. It should be clear upon inspection how expressing Q_j as the above would allow it to be subsumed into U in eqns. (3.1) and (3.6).

3.1.4 The Geometry of Lagrange's Equations: Systems of Particles

Though the above refers to Lagrange's equations without discussion of bases, modeling physical systems requires doing so. The robots in this dissertation are modeled as systems of particles, which will be the realization of Lagrange's equations considered here.

The following subsection establishes three main results from the literature. First, a discussion of the geometry of an unconstrained system of particles gives context for the vague descriptions of eqns. (3.1) and (3.6). Second, potential energy and generalized forces are defined for a particle and system of particles. Finally, the definitions of conservative forces for the particles are shown to be equivalent to the concept of a conservative generalized force.

Configuration Space, Configuration Manifold, and Position Vectors

Let a mechanical system be defined in a given configuration space $\mathcal{C}$. For example, the configuration space for a system of K unconstrained particles is Euclidean space for each particle, $\mathcal{C} = \mathbb{E}^{3K}$. Let there be a vector (or a set of vectors for $K > 1$) that expresses the system's state in this space, each implicitly parameterized by the generalized coordinates $\mathbf{q}$. For example, with a system of particles,

$$\mathbf{r}_1(\mathbf{q}),\ \ldots,\ \mathbf{r}_K(\mathbf{q})\ \in \mathbb{E}^3.$$

If these particles are constrained, their motion lies on a configuration manifold $\mathcal{M}$ of lower dimension than $\mathcal{C}$, i.e. $\mathcal{M} \subseteq \mathcal{C}$. Then, the generalized coordinates are said to parameterize $\mathcal{M}$. Motions of the system are then considered as changes in $\mathbf{q}$, which change $\mathbf{r}_{1\ldots K}$ in $\mathcal{M}$. Also observe that the particles' positions are possibly a function of all the generalized coordinates $\mathbf{q}$, though smart choices of parameters decouple $\mathbf{q}$ into blocks per particle.

Potential Energy in Euclidean Space

It is now possible to be explicit about the arguments to U. In particular, the potential energy is only a function of the particles' positions:

$$U = U(\mathbf{r}_1(\mathbf{q}),\ \ldots,\ \mathbf{r}_K(\mathbf{q})).$$

The dependence of $\mathbf{r}$ on $\mathbf{q}$ will be dropped in the notation in the remaining derivations; it is always implied.

The point raised about partial derivatives versus gradients in reference to eqn. (3.5) can now be made clear via the chain rule. The gradient is defined in $\mathbb{E}^3$ for a single particle with position vector $\mathbf{r}$ as

$$\nabla U(\mathbf{r}) = \frac{\partial U}{\partial \mathbf{r}} \in \mathbb{E}^3, \tag{3.8}$$

and for particle k in a system as

$$\nabla_{\mathbf{r}_k} U(\mathbf{r}_1, \ldots, \mathbf{r}_K) = \frac{\partial U}{\partial \mathbf{r}_k} \in \mathbb{E}^3. \tag{3.9}$$

By the chain rule, we can relate the above to eqn. (3.5), since for a single particle,

$$\frac{\partial U}{\partial q_j} = \frac{\partial U}{\partial \mathbf{r}} \cdot \frac{\partial \mathbf{r}}{\partial q_j} = \nabla U(\mathbf{r}) \cdot \frac{\partial \mathbf{r}}{\partial q_j}, \tag{3.10}$$

and for a system of particles,

$$\frac{\partial U}{\partial q_j} = \sum_{k=1}^{K} \frac{\partial U}{\partial \mathbf{r}_k} \cdot \frac{\partial \mathbf{r}_k}{\partial q_j} = \sum_{k=1}^{K} \nabla_{\mathbf{r}_k} U \cdot \frac{\partial \mathbf{r}_k}{\partial q_j}. \tag{3.11}$$

As a result, another way to write the expression in eqn. (3.5) for a system of particles is

$$\frac{\partial U}{\partial \mathbf{q}} := \begin{bmatrix} \displaystyle\sum_{k=1}^{K} \nabla_{\mathbf{r}_k} U \cdot \frac{\partial \mathbf{r}_k}{\partial q_1} \\ \vdots \\ \displaystyle\sum_{k=1}^{K} \nabla_{\mathbf{r}_k} U \cdot \frac{\partial \mathbf{r}_k}{\partial q_n} \end{bmatrix} \in \mathbb{R}^n. \tag{3.12}$$

It is therefore clear that

$$\nabla_{\mathbf{r}_k} U = \frac{\partial U}{\partial \mathbf{r}_k} \in \mathbb{E}^3 \quad \neq \quad \frac{\partial U}{\partial \mathbf{q}} \in \mathbb{R}^n,$$

since they are in different vector spaces and have different dimensions. This salient point is overlooked by most standard control systems references on the stability of a Lagrangian system [32, 196, 144]. However, both expressions give the same definition of potential energy, when considered with the generalized force as follows.

Generalized Force Components in Euclidean Space

Deriving Lagrange's equations for a single particle or system of particles gives the following definition of the generalized force components in those cases [142, 77]. Letting the total force on the particle be $\mathbf{F}$ or on particle k be $\mathbf{F}_k$,

$$Q_j = \mathbf{F} \cdot \frac{\partial \mathbf{r}}{\partial q_j}, \quad \text{or for a system,} \tag{3.13}$$

$$Q_j = \sum_{k=1}^{K} \mathbf{F}_k \cdot \frac{\partial \mathbf{r}_k}{\partial q_j}. \tag{3.14}$$

The geometric definition of conservative forces in (a system of) particles is that the force is the gradient of a scalar function of the particle's position, either

$$\mathbf{F} = \nabla U(\mathbf{r}) \quad \text{or} \quad \mathbf{F}_k = \nabla_{\mathbf{r}_k} U(\mathbf{r}_1, \ldots, \mathbf{r}_K).$$

Now, the equivalence with the definition of a conservative generalized force can be established by substituting such a conservative force into eqn. (3.13) or (3.14) respectively, and noting the equivalence to eqns. (3.10) and (3.11):

$$Q_j = \nabla U \cdot \frac{\partial \mathbf{r}}{\partial q_j} \qquad = \frac{\partial U}{\partial q_j},$$

or for a system,

$$Q_j = \sum_{k=1}^{K} \nabla_{\mathbf{r}_k} U \cdot \frac{\partial \mathbf{r}_k}{\partial q_j} \quad = \frac{\partial U}{\partial q_j}.$$

The following important fact has been established: either the component-wise definition or the geometric Euclidean space definition of conservation can be used interchangeably. Summing over all components shows this result, which also emphasizes the difference in a component-wise sense of conservation (according to coordinate j) versus a particle-wise sense of conservation (according to particle k).

$$\mathbf{Q} = \frac{\partial U}{\partial \mathbf{q}} \quad \Longleftrightarrow \quad \mathbf{F}_k = \frac{\partial U}{\partial \mathbf{r}_k} \ \forall\, k. \tag{3.15}$$

It is worth briefly reiterating that the notation $\nabla_{\mathbf{r}_k} U = \frac{\partial U}{\partial \mathbf{r}_k}$ will be used interchangeably; i.e., the gradient operator ∇ will only be used in $\mathbb{E}^3$.

3.1.5 Variational Approach to the Work-Energy Theorem

The various theorems on passivity and stability of a Lagrangian system require the following lemma, which derives a form of the work-energy theorem from the variational approach (eqn.

3.1). The lemma is then shown to be equivalent to the traditional work-energy theorem for a system of particles, which will be used interchangeably in the remainder of this dissertation.

The following is adapted from [32, 144] to match the notation used here

Lemma 3.1.0.1. *The Work-Energy Principle for a Component-Wise Representation of Lagrange's Equations.*

Consider a system modeled by eqn. (3.1), with n *generalized coordinates. The instantaneous change in energy in the system, $\dot{H}$, is equal to the inner product of the generalized velocities and the generalized force component vector:*

$$\dot{H} = \sum_{j=1}^{n} \dot{q}_j Q_j = \dot{\mathbf{q}}^\top \mathbf{Q}. \tag{3.16}$$

Proof. For a system considered here, the total energy in the system is the kinetic plus potential energy, equal to the Hamiltonian H. To show eqn. (3.16), first consider multiplying the j-th of Lagrange's equations with the j-th generalized velocity, as in:

$$\dot{q}_j \left(\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_j} \right) - \dot{q}_j \left(\frac{\partial L}{\partial q_j} \right) = \dot{q}_j Q_j. \tag{3.17}$$

Then, take the following derivative:

$$\frac{d}{dt} \left(\dot{q}_j \frac{\partial L}{\partial \dot{q}_j} \right) = \ddot{q}_j \frac{\partial L}{\partial \dot{q}_j} + \dot{q}_j \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_j}. \tag{3.18}$$

Rearrange,

$$\dot{q}_j \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_j} = \frac{d}{dt} \left(\dot{q}_j \frac{\partial L}{\partial \dot{q}_j} \right) - \ddot{q}_j \frac{\partial L}{\partial \dot{q}_j}, \tag{3.19}$$

substitute into (3.17) as

$$\frac{d}{dt} \left(\dot{q}_j \frac{\partial L}{\partial \dot{q}_j} \right) - \ddot{q}_j \frac{\partial L}{\partial \dot{q}_j} - \dot{q}_j \left(\frac{\partial L}{\partial q_j} \right) = \dot{q}_j Q_j, \tag{3.20}$$

and sum over all coordinates:

$$\sum_j \frac{d}{dt} \left(\dot{q}_j \frac{\partial L}{\partial \dot{q}_j} \right) - \sum_j \ddot{q}_j \frac{\partial L}{\partial \dot{q}_j} - \sum_j \dot{q}_j \left(\frac{\partial L}{\partial q_j} \right) = \sum_j \dot{q}_j Q_j. \tag{3.21}$$

Next, take the full time derivative of the Lagrangian, noting that it has both the coordinates and velocities as arguments, and using the chain rule:

$$\frac{d}{dt} L = \sum_{j=1}^{n} \frac{\partial L}{\partial q_j} \dot{q}_j + \sum_{j=1}^{n} \frac{\partial L}{\partial \dot{q}_j} \ddot{q}_j. \tag{3.22}$$

Rearrange eqn. (3.22) as

$$-\sum_{j=1}^{n} \dot{q}_j \frac{\partial L}{\partial q_j} = \sum_{j=1}^{n} \frac{\partial L}{\partial \dot{q}_j} \ddot{q}_j - \frac{dL}{dt}, \tag{3.23}$$

and substitute into eqn. (3.21), eliminating the $\ddot{q}$ terms:

$$\sum_j \frac{d}{dt}\left(\dot{q}_j \frac{\partial L}{\partial \dot{q}_j}\right) - \sum_j \ddot{q}_j \frac{\partial L}{\partial \dot{q}_j} + \sum_j \frac{\partial L}{\partial \dot{q}_j} \ddot{q}_j - \frac{dL}{dt} = \sum_j \dot{q}_j Q_j,$$

$$\sum_j \frac{d}{dt}\left(\dot{q}_j \frac{\partial L}{\partial \dot{q}_j}\right) - \frac{dL}{dt} = \sum_j \dot{q}_j Q_j. \tag{3.24}$$

Since the derivative is a linear operator,

$$\frac{d}{dt}\left(\sum_j \dot{q}_j \frac{\partial L}{\partial \dot{q}_j} - L\right) = \sum_j \dot{q}_j Q_j. \tag{3.25}$$

The Legendre transformation, commonly used to interchange the Lagrangian and Hamiltonian formulations of mechanical systems, states the following:

$$H = \sum_j \dot{q}_j \frac{\partial L}{\partial \dot{q}_j} - L \tag{3.26}$$

Therefore, the following expression holds for the time derivative of the Hamiltonian, noting that the vectors $\dot{\mathbf{q}}$ and $\mathbf{Q}$ have been defined to produce the inner product on the right-hand side of this equation.

$$\dot{H} = \sum_j \dot{q}_j Q_j = \dot{\mathbf{q}}^\top \mathbf{Q}. \tag{3.27}$$

□

Remark. The variational (component-wise) form of the work-energy theorem is equivalent to the more commonly encountered geometric form, using forces and velocities for particles in $\mathbb{E}^3$. By expanding out eqn. (3.27),

$$
\begin{aligned}
\dot{H} &= \dot{\mathbf{q}}^\top \mathbf{Q}, \\
&= \sum_{j=1}^{n} \dot{q}_j Q_j, \\
&= \sum_{j=1}^{n} \dot{q}_j \left(\sum_{k=1}^{K} \mathbf{F}_k \cdot \frac{\partial \mathbf{r}_k}{\partial q_j} \right), \\
&= \sum_{k=1}^{k} \mathbf{F}_k \cdot \left(\sum_{j=1}^{n} \dot{q}_j \frac{\partial \mathbf{r}_k}{\partial q_j} \right).
\end{aligned}
$$

Since the chain rule for a particle's velocity gives

$$
\mathbf{v}_k = \dot{\mathbf{r}}_k = \frac{d}{dt}\mathbf{r} = \sum_j \frac{\partial \mathbf{r}_k}{\partial q_j} \frac{dq_j}{dt},
$$

the geometric work-energy theorem follows:

$$
\dot{H} = \sum_{k=1}^{K} \mathbf{F}_k \cdot \mathbf{v}_k. \tag{3.28}
$$

3.2 Passivity and Dissipativity

Having now established some energy-related principles that arise from the dynamics of a mechanical system, a framework for control can be established. The following section details the related concepts of *passivity* and *dissipativity* for input-output analysis of a nonlinear system. The treatment here is shortened and adapted from Arcak et al. [10]. Various intricacies, such as existence and uniqueness, can be found in [56] and related.

3.2.1 Definition of Passivity and Dissipativity

Consider a dynamical system of the form

$$
\dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)), \tag{3.29}
$$
$$
\mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)), \tag{3.30}
$$

with the known condition $\mathbf{g}(\mathbf{0}, \mathbf{0}) = \mathbf{0}$ and $\mathbf{h}(\mathbf{0}, \mathbf{0}) = \mathbf{0}$. Here, the state vector is $\mathbf{x}(t) \in \mathbb{R}^n$, the input vector is $\mathbf{u}(t) \in \mathbb{R}^m$, and the system's output is $\mathbf{y} \in \mathbb{R}^p$. The functions $\mathbf{g}, \mathbf{h}$ are continuously differentiable mappings, $\mathbf{g} : \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathbb{R}^n$ and $\mathbf{h} : \mathbb{R}^m \times \mathbb{R}^m \mapsto \mathbb{R}^p$.

An input-output analysis can be conducted on the system (3.29)-(3.30) via the notion of *dissipativity*, as many authors have done [10, 32].

Definition 3.2.1. Dissipativity. The system (3.29)-(3.30) is *dissipative with respect to a supply rate* $s(\cdot,\cdot) : \mathbb{R}^m \times \mathbb{R}^p \mapsto \mathbb{R}$ if there exists a function $V : \mathbb{R}^n \mapsto \mathbb{R}$ such that $V(\mathbf{0}) = 0$, $V(\mathbf{x}) \geq 0 \; \forall \mathbf{x}$, and

$$V(\mathbf{x}(\tau)) - V(\mathbf{x}(0)) \leq \int_0^\tau s(\mathbf{u}(t), \mathbf{y}(t)) \, dt \tag{3.31}$$

for every input signal $\mathbf{u}$ and every time $\tau \geq 0$ in the interval of existence of the solution $\mathbf{x}(t)$. The function $V(\mathbf{x}(t))$ is called the *storage function*.

Remark. If $V(\cdot)$ is also continuously differentiable, then eqn. (3.30) can be substituted into Defn. 3.2.1, and so the above is equivalent to

$$\dot{V}(\mathbf{x}(t)) \leq s(\mathbf{u}(t), \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)) \quad \forall\, \mathbf{x} \in \mathbb{R}^n, \; \forall\, \mathbf{u} \in \mathbb{R}^m, \tag{3.32}$$

or more succinctly with arguments dropped, $\dot{V} \leq s(\mathbf{u}, \mathbf{h}(\mathbf{u}, \mathbf{y}))$.

The choice of supply rate s defines the type of dissipativity considered. There are many important supply rates, in particular the L_2-gain supply rate [197]. However, the *passivity* supply rate will be the focus of this dissertation. It is defined as the following.

Definition 3.2.2. Passivity. Consider a system (3.29)-(3.30) with the same dimension of input and output, i.e. $m = p$. It is *passive* if it is dissipative with respect to the supply rate of

$$s(\mathbf{u}, \mathbf{y}) = \mathbf{u}^\top \mathbf{y}. \tag{3.33}$$

Two related supply rates, those of *input-strict* and *output-strict* passivity, are needed in later chapters. These add a bound on the supply rate according to either the input or the output respectively [197]. Here, $||\cdot||$ is used as the Euclidean norm.

Definition 3.2.3. Input Strict Passivity. A system is *input strictly passive* if it is dissipative with respect to a supply rate of

$$s(\mathbf{u}, \mathbf{y}) = \mathbf{u}^\top \mathbf{y} - \varepsilon ||\mathbf{u}||_2^2, \tag{3.34}$$

or equivalently, $s = \mathbf{u}^\top \mathbf{y} - \varepsilon \mathbf{u}^\top \mathbf{u}$, where $\varepsilon \in \mathbb{R} > 0$.

Definition 3.2.4. Output Strict Passivity. A system is *output strictly passive* if it is dissipative with respect to a supply rate of

$$s(\mathbf{u}, \mathbf{y}) = \mathbf{u}^\top \mathbf{y} - \varepsilon ||\mathbf{y}||_2^2, \tag{3.35}$$

or equivalently, $s = \mathbf{u}^\top \mathbf{y} - \varepsilon \mathbf{y}^\top \mathbf{y}$, where $\varepsilon \in \mathbb{R} > 0$.

A generalization of (input or output strict) passivity is that of the quadratic supply rate. This is particularly useful if the dimensions of $\mathbf{u}$ and $\mathbf{y}$ are not conformal.

Definition 3.2.5. Quadratic Supply Rate. A supply rate is *quadratic* if it can be expressed as

$$s(\mathbf{u}, \mathbf{y}) = \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \end{bmatrix}^\top \mathbf{X} \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \end{bmatrix}, \tag{3.36}$$

where $\mathbf{X} \in \mathbb{R}^{(m+p)\times(m+p)}$ is a constant matrix.

The blocks of $\mathbf{X}$ can be selected to meet the definitions 3.2.2, 3.2.3, and 3.2.4 (see [10].)

3.2.2 Memoryless Nonlinearities

In the following sections, systems without internal state will occasionally be analyzed for their dissipativity properties. These *memoryless nonlinearities* have no concept of a state '$\mathbf{x}$'. Consequently, the system's dynamics (3.29-3.30) reduce to

$$\mathbf{y}(t) = \mathbf{h}(\mathbf{u}(t)). \tag{3.37}$$

The storage function is taken to be zero for these systems. Specifically, a memoryless nonlinearity will be dissipative with respect to the supply rate s if

$$s(\mathbf{u}, \mathbf{h}(\mathbf{u})) \geq 0 \quad \forall\, \mathbf{u} \in \mathbb{R}^m. \tag{3.38}$$

Typical examples of memoryless nonlinearities are nonlinear springs and dampers [171]. For example, if a particle in one dimension with position $x \in \mathbb{R}$ is attached to a nonlinear spring and nonlinear damper, the applied forces on the particle would arise from some functions f and w,

$$F_s(x) = f(x), \quad F_d(\dot{x}) = w(\dot{x}).$$

In these cases, the function $F : \mathbb{R} \mapsto \mathbb{R}$, and so the dissipativity inequality would be for example

$$s(x, f(x)) \geq 0.$$

Linear springs and dampers would be $f(x) = kx$ and $w(\dot{x}) = c\dot{x}$, which are both clearly input and output strictly passive with $\varepsilon = \{k,\ c\} > 0$.

3.3 Passivity-Based Control of Lagrangian Systems

With this mathematical machinery in hand, the two fields of Lagrangian dynamics and dissipativity can be united. The result is termed *passivity-based control* [144], since passivity is used in these cases to show stability. Stability can arise from passivity (or dissipativity) for arbitrary dynamical systems, but here, the following proofs are presented with respect to Lagrange's equations, since that will cover all applications considered in later chapters.

The section is split into two parts. First, the passivity of a Lagrangian system, from an input generalized force to an output generalized velocity, is proven given certain conditions. Second, a stability proof is provided, given an unforced system with a stricter set of conditions. These two proofs form the basis of the work in Chap. 8, which will modify both for use with the proposed energy-shaping controller.

3.3.1 Passivity of Lagrangian Systems with Bounded Potential Energy

The following proposition, adapted from [197, 32, 144] gives the passivity of a Lagrangian system under some general circumstances. Specifically, the system is passive with respect to a generalized force as input, and generalized velocity as output, when (a) only conservative forces act on the system and (b) the potential energy is bounded. It is proposed in accordance with the dynamics community's formulations of Lagrange's equations.

Below, the full system state is both the combined generalized forces and generalized velocities,

$$\mathbf{x} := \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \in \mathcal{X} \times \mathbb{R}^n \subseteq \mathbb{R}^{2n}.$$

The use of $\mathbf{q}$ and $\dot{\mathbf{q}}$ differs slightly from the control systems terminology using $\mathbf{x}$. The full state here is of the same dimension as usually encountered in control, but the constant n has a different meaning. Treating $\mathbf{q}$ and $\dot{\mathbf{q}}$ separately is needed for some parts of the following proof, so the storage function $V(\cdot, \cdot)$ is written with two arguments for clarity. This is equivalent to the definition of passivity (3.2.2) above, substituting various constants (e.g. n vs. $2n$.)

Proposition 3.3.1. *Passivity of Euler-Lagrange systems with Bounded Potential Energy.*

Consider a system that is described by Lagrangian mechanics, eqn. (3.1), with n *generalized coordinates, velocities, and forces:* $q_j \in \mathcal{X}_j \subseteq \mathbb{R}$, $\dot{q}_j$, $Q_j \in \mathbb{R}$, $j = 1 \ldots n$. *Specifically,*

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \left(\frac{\partial L}{\partial q_j}\right) = Q_j, \qquad j = 1 \ldots n,$$

where $L = T - U$ *is the Lagrangian,* T *is the kinetic energy, and* U *is the potential energy.*

If U *is bounded below by a constant* $c \in \mathbb{R}$,

$$U \geq c,$$

and the total generalized forces are treated as the input to the system, $Q_j = u_j$, *then the system is passive from* $\dot{\mathbf{q}}$ *to* $\mathbf{Q}$, *i.e.*

$$V(\mathbf{q}(\tau)), \dot{\mathbf{q}}(\tau)) - V(\mathbf{q}(0)), \dot{\mathbf{q}}(0)) \leq \int_0^\tau \dot{\mathbf{q}}(t)^\top \mathbf{Q}(t) dt, \tag{3.39}$$

for every input signal $\mathbf{Q}$ *and every time interval* $t = [0, \tau)$ *in the interval of existence of the solution* $\mathbf{q}(t)$*, with the storage function* $V(\cdot, \cdot) : \mathcal{X} \times \mathbb{R}^n \mapsto \mathbb{R}$ *chosen to be the Hamiltonian* $H = T + U$ *minus the bounding constant for the potential energy,*

$$V(\mathbf{q}, \dot{\mathbf{q}}) := H(\mathbf{q}, \dot{\mathbf{q}}) - c. \tag{3.40}$$

Proof. Consider the storage function candidate

$$V(\mathbf{q}, \dot{\mathbf{q}}) = H(\mathbf{q}, \dot{\mathbf{q}}) + c = T(\mathbf{q}, \dot{\mathbf{q}}) + U(\mathbf{q}) - c.$$

First, justify that this is a valid storage function. The total kinetic energy in mechanical systems is always nonnegative, $T \geq 0$. So,

$$T \geq 0, \;\; U - c \geq 0 \quad \Rightarrow \quad T + U - c \geq 0 \quad \Rightarrow \quad H - c \geq 0.$$

Thus, $V(\cdot, \cdot)$ is nonnegative, and is a valid storage function.

However, in anticipation of the work in following sections where an equilibrium point $\bar{\mathbf{q}}$ will be considered, $V(\cdot, \cdot)$ is not necessarily zero for zero argument. A coordinate transformation to $\mathbf{q}'$ such that $\mathbf{q}' = \mathbf{0} \iff \mathbf{q} = \bar{\mathbf{q}}$ can easily address this problem and so is ignored here.

From Lemma 3.1.0.1,

$$\dot{H} = \dot{\mathbf{q}}^\top \mathbf{Q},$$

where $\dot{\mathbf{q}}$ and $\mathbf{Q}$ are vectors of the generalized coordinates/velocities as per eqns. (3.3)-(3.4). Observe that $\dot{V} = \dot{H}$ since the bounding constant does not change with time. Integrating from 0 to τ, and noting that the constant c subtracts away in $V(\tau) - V(0)$, gives the desired expression:

$$V(\mathbf{q}(\tau)), \dot{\mathbf{q}}(\tau)) - V(\mathbf{q}(0)), \dot{\mathbf{q}}(0)) = \int_0^\tau \dot{\mathbf{q}}(t)^\top \mathbf{Q}(t) dt, \tag{3.41}$$

establishing both the passivity and the losslessness of this formulation.

□

Remark. The losslessness noted in the equality of eqn. (3.41) versus eqn. (3.39) arises from the fact that the system does not dissipate energy. For a mechanical system, this would be interpreted as free oscillations without damping. It is intuitive, then, that adding a damping (dissipation) term gives a stronger condition, specifically output strict passivity, which then shows the system settling to an equilibrium point (or surface.)

3.3.2 Output Strict Passivity of Lagrangian Systems with Rayleigh Dissipation

The aforementioned addition of energy dissipation into a Lagrangian system can take a variety of forms. The most common, forming the basic principle of passivity-based control [197, 144, 32] is an internal force in the system that is a function of the generalized velocities. Specifically, consider if the generalized forces in the system were not themselves an input, but instead consisted of a memoryless nonlinearity added to a separate input term:

$$Q_j := -R_j(\dot{\mathbf{q}}) + u_j, \qquad j = 1 \ldots n, \tag{3.42}$$

or as vectors of components in $\mathbb{R}^n$,

$$\mathbf{Q} := -\mathbf{R}(\dot{\mathbf{q}}) + \mathbf{u}. \tag{3.43}$$

The function $\mathbf{R}$ is considered to be a memoryless nonlinearity - for example, a dashpot, which only a function of the system's velocities. Since $\mathbf{R} : \mathbb{R}^n \mapsto \mathbb{R}^n$, its passivity and input-strict passivity respectively would be given if

$$\dot{\mathbf{q}}^\top \mathbf{R}(\dot{\mathbf{q}}) \geq 0, \quad \text{or respectively,} \tag{3.44}$$

$$\exists\, \varepsilon > 0 \quad \text{s.t.} \quad \dot{\mathbf{q}}^\top \mathbf{R}(\dot{\mathbf{q}}) \geq \varepsilon \dot{\mathbf{q}}^\top \dot{\mathbf{q}}, \qquad \forall \dot{\mathbf{q}} \in \mathbb{R}^n. \tag{3.45}$$

The common interpretation of the term $\mathbf{R}$ is a damping 'force', informally termed 'Rayleigh Dissipation' [144, 77]. It is often expressed as a set of partial derivatives of the Rayleigh Dissipation Function $\mathcal{R}$, written in vectorized form similar to potential energy (eqn. 3.5)

$$\mathbf{R}(\dot{\mathbf{q}}) := \frac{\partial \mathcal{R}(\dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} \tag{3.46}$$

This form emphasizes the fact that $\mathbf{R}(\cdot)$ is only a function of $\dot{\mathbf{q}}$, and allows the discussion of $\mathcal{R}$ as a counterpart to the potential energy $U(\mathbf{q})$. This form also allows for conveniently analyzing the dissipativity of $\mathbf{R}$ if $\mathcal{R}$ is quadratic in $\dot{\mathbf{q}}$, leading to a positive-definiteness test for input-strict passivity. As an example, a one-dimensional spring-damper system would have

$$U(x) = \frac{1}{2}kx^2, \qquad \mathcal{R}(\dot{x}) = \frac{1}{2}c\dot{x}^2$$

Since none of the results in this dissertation rely on properties of $\mathcal{R}$, it is not used, and $\mathbf{R}$ is used instead to denote the total energy-dissipating, nonconservative force with no position dependence.

These definitions of $\mathbf{R}$ then give (in one way of potentially many) output strict passivity of the Lagrangian system. The following proof is adapted from [144, 197, 32] as with the passivity proof above.

Proposition 3.3.2. *Output Strict Passivity of Lagrangian Systems with Rayleigh Dissipation.*

Consider a Lagrangian system (eqn. 3.1),

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \left(\frac{\partial L}{\partial q_j}\right) = Q_j, \qquad j = 1 \ldots n.$$

Assume this system is passive, i.e., has bounded potential energy and satisfies the conditions of Prop. (3.3.1). Assume that, in addition, the generalized forces can be split into two terms,

$$\mathbf{Q} = -\mathbf{R}(\dot{\mathbf{q}}) + \mathbf{u} \tag{3.47}$$

where $\mathbf{u}$ represents an external input signal with component u_j in the direction of the j-th generalized coordinate, and $\mathbf{R} : \mathbb{R}^n \mapsto \mathbb{R}^n$ is a memoryless nonlinearity that is only a function of the generalized velocities.

Then, if $\mathbf{R}$ is input strictly passive, i.e. eqn. (3.45) holds,

$$\exists\, \varepsilon > 0 \quad s.t. \quad \dot{\mathbf{q}}^\top \mathbf{R}(\dot{\mathbf{q}}) \geq \varepsilon \dot{\mathbf{q}}^\top \dot{\mathbf{q}}, \qquad \forall \dot{\mathbf{q}} \in \mathbb{R}^n,$$

then the system (3.1) is output strictly passive, satisfying the inequality

$$V(\mathbf{q}(\tau), \dot{\mathbf{q}}(\tau)) - V(\mathbf{q}(0), \dot{\mathbf{q}}(0)) + \varepsilon \int_0^\tau \dot{\mathbf{q}}(t)^\top \dot{\mathbf{q}}(t)\, dt \leq \int_0^\tau \dot{\mathbf{q}}(t)^\top \mathbf{u}(t)\, dt, \tag{3.48}$$

for every input signal $\mathbf{u}$ and every time interval $t = [0, \tau)$ in the interval of existence of the solution $\mathbf{q}(t)$, with the same storage function $V(\cdot, \cdot)$ as in Prop. 3.3.1.

Proof. Consider the same storage function candidate as in Prop. 3.3.1,

$$V(\mathbf{q}, \dot{\mathbf{q}}) = H(\mathbf{q}, \dot{\mathbf{q}}) - c,$$

which is nonnegative by the same arguments as Prop. 3.3.1, and recall that Lemma 3.1.0.1 showed that

$$\dot{H} = \dot{\mathbf{q}}^\top \mathbf{Q}.$$

Substituting for $\mathbf{Q}$ yields the following, similar to Prop. 3.3.1, dropping excess notation:

$$\dot{H} = -\dot{\mathbf{q}}^\top \mathbf{R} + \dot{\mathbf{q}}^\top \mathbf{u}, \tag{3.49}$$

$$\Rightarrow \quad V(\tau) - V(0) = -\int_0^\tau \dot{\mathbf{q}}^\top \mathbf{R}\, dt + \int_0^\tau \dot{\mathbf{q}}^\top \mathbf{u}\, dt. \tag{3.50}$$

The input-strict passivity of the memoryless nonlinearity $\mathbf{R}$ then gives that

$$\dot{\mathbf{q}}^\top \mathbf{R} \geq \varepsilon \dot{\mathbf{q}}^\top \dot{\mathbf{q}} \Rightarrow \quad \int_0^\tau \dot{\mathbf{q}}^\top \mathbf{R}\, dt \geq \varepsilon \int_0^\tau \dot{\mathbf{q}}^\top \dot{\mathbf{q}}\, dt.$$

The desired inequality then holds:

$$V(\tau) - V(0) + \varepsilon \int_0^\tau \dot{\mathbf{q}}^\top \dot{\mathbf{q}}\, dt \le \int_0^\tau \dot{\mathbf{q}}^\top \mathbf{u}\, dt.$$

□

3.3.3 Stability of Lagrangian Systems with Rayleigh Dissipation and Strict Minima of Potential Energy

Finally, stability of the above system can be considered in the autonomous case ($\mathbf{u} \equiv \mathbf{0}$.) When a system is designed to meet the following proof, the literature terms this *passivity-based control*, which may also imply the interconnection of systems via $\mathbf{u}$ in other ways [197, 144, 32].

The derivations below depend on the particular usage of strict passivity above, and do not apply for systems that (for example) do not have this particular form of Rayleigh dissipation, or other considerations with potential energy. Therefore, the proof is best not considered as 'stability of Lagrangian systems,' as is sometimes used [144], but instead stability under certain conditions on forces and energy.

The following continues to take the variational approach to treating Lagrange's equations, as with Props. 3.3.1 and 3.3.2. For the unconstrained system of particles, proving stabilization to an equilibrium point $\bar{\mathbf{q}}$ in $\mathbb{R}^n$ will later be shown to correspond to a configuration $\bar{\mathbf{r}}_1 \dots \bar{\mathbf{r}}_K$ in $\mathbb{E}^{3K}$. In particular, the arguments to the potential energy U will not be specified; it is implied that $\mathbf{q}$ parameterizes U. Examples will be given later for (systems of) particles, where for example $U = U(\mathbf{r}_1(\mathbf{q}), \dots, \mathbf{r}_K(\mathbf{q}))$.

Proposition 3.3.3. *Stability of a Lagrangian System with Rayleigh Dissipation and Strict Minima of Potential Energy.*

Consider the same system from Prop. 3.3.2, in the form of (3.6), with Rayleigh dissipation as $\mathbf{Q} = -\mathbf{R} + \mathbf{u}$. Let the system be unforced, with $\mathbf{u} = \mathbf{0}$, so that

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \left(\frac{\partial L}{\partial \mathbf{q}}\right) = -\mathbf{R}(\dot{\mathbf{q}}).$$

Assume the system is output strictly passive, i.e. has bounded potential energy and $\mathbf{R}$ is input strictly passive (Prop. 3.3.2).

The following hold:

1. *If in addition the potential energy U has a strict local minimum at coordinates $\bar{\mathbf{q}}$ in a set $\mathcal{Q} \subseteq \mathcal{X} \subseteq \mathbb{R}^n$, i.e.*

$$U\Big|_{\mathbf{q}=\bar{\mathbf{q}}} = c, \qquad U > c \quad \forall \mathbf{q} \neq \bar{\mathbf{q}} \in \mathcal{Q}, \tag{3.51}$$

   then $\bar{\mathbf{q}}$ is an equilibrium point, and the system is locally asymptotically stable in the sense of Lyapunov around that equilibrium point in $\mathcal{Q}$.

2. *If also U is radially unbounded (proper), then the system is asymptotically stable around the point $\bar{\mathbf{q}}$ in all of $\mathcal{Q}$. If $\mathcal{Q} = \mathcal{R}^n$ then the system is globally asymptotically stable.*

Proof. First, establish the equilibrium point(s) of the system (3.6.) By the definition, equilibria exist where $\dot{\mathbf{q}} = \mathbf{0}$. Using the definition of $L = T - U$, as well as the fact that the total potential energy is not a function of the generalized velocities, Lagrange's equations at zero velocity become

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_j}\Big|_{\substack{\mathbf{q}=\bar{\mathbf{q}}\\ \dot{\mathbf{q}}=\mathbf{0}}}\right) - \frac{\partial T}{\partial q_j}\Big|_{\substack{\mathbf{q}=\bar{\mathbf{q}}\\ \dot{\mathbf{q}}=\mathbf{0}}} + \frac{\partial U}{\partial q_j}\Big|_{\mathbf{q}=\bar{\mathbf{q}}} = -R_j(\mathbf{0}), \qquad j = 1 \ldots n. \tag{3.52}$$

By definition, kinetic energy is zero at zero velocity, so all T terms drop. In vector form,

$$\frac{\partial U}{\partial \mathbf{q}}\Big|_{\mathbf{q}=\bar{\mathbf{q}}} = -\mathbf{R}(\mathbf{0}). \tag{3.53}$$

Then, since the function $\mathbf{R}$ is input strictly passive, $\mathbf{R}(\dot{\mathbf{q}} = \mathbf{0}) = \mathbf{0}$, and

$$\frac{\partial U}{\partial \mathbf{q}}\Big|_{\mathbf{q}=\bar{\mathbf{q}}} = \mathbf{0}. \tag{3.54}$$

By the assumption that U is bounded below (and assumptions on continuous differentiability giving extrema of U), there exist $\mathbf{q}$ that satisfy this property, which are subsequently the equilibrium point(s):

$$U \geq c, \;\; \exists\, \bar{\mathbf{q}} \text{ s.t. } U\Big|_{\mathbf{q}=\bar{\mathbf{q}}} = c, \qquad \Rightarrow \qquad \exists\, \bar{\mathbf{q}} \text{ s.t. } \frac{\partial U}{\partial \mathbf{q}}\Big|_{\mathbf{q}=\bar{\mathbf{q}}} = \mathbf{0}. \tag{3.55}$$

Consider the storage function $V = H - c$ as a Lyapunov candidate. First establish that it is a positive definite function. It was shown in the proof to Prop. (3.3.1) that V is nonnegative. In addition, the storage function is zero at the equilibrium point's coordinates:

$$\begin{aligned} V(\bar{\mathbf{q}}, \mathbf{0}) &= T\Big|_{\substack{\mathbf{q}=\bar{\mathbf{q}}\\ \dot{\mathbf{q}}=\mathbf{0}}} + U\Big|_{\mathbf{q}=\bar{\mathbf{q}}} - c, \\ V(\bar{\mathbf{q}}, \mathbf{0}) &= c - c = 0. \end{aligned}$$

Then specialize to the first assumption, where U has a strict local minimum at $\bar{\mathbf{q}}$ in $\mathcal{Q}$. Therefore,

$$U > c \Rightarrow U - c > 0 \qquad \forall \mathbf{q} \neq \bar{\mathbf{q}} \in \mathcal{Q}.$$

Since $T \geq 0$ for any argument,

$$T + U - c > 0 \qquad \forall \mathbf{q} \neq \bar{\mathbf{q}} \in \mathcal{Q},$$

$$V(\mathbf{q}, \dot{\mathbf{q}}) > 0 \qquad \forall \mathbf{q} \neq \bar{\mathbf{q}} \in \mathcal{Q}.$$

Therefore, V is a locally positive definite function, $V \succ 0$, around $\{\mathbf{q} = \bar{\mathbf{q}}, \dot{\mathbf{q}} = 0\}$.

Next, consider $\dot{V}$. Assuming (again) continuous differentiability of the storage function, from Prop. (3.3.2) it was shown that, with $\mathbf{u} \equiv \mathbf{0}$ in eqn. (3.48), input strict passivity of $\mathbf{R}$ gives

$$\dot{V}(\mathbf{q}, \dot{\mathbf{q}}) \leq -\varepsilon \dot{\mathbf{q}}^\top \dot{\mathbf{q}}. \tag{3.56}$$

Since the right-hand side of eqn. (3.56) is quadratic,

$$\varepsilon \dot{\mathbf{q}}^\top \dot{\mathbf{q}} \geq 0 \qquad \Rightarrow \qquad \dot{V}(\mathbf{q}, \dot{\mathbf{q}}) \leq 0.$$

Therefore, $\dot{V}$ is negative semi-definite, $\dot{V} \preceq 0$, meets all requirements for a Lyapunov function, and shows stability of the system in the sense of Lyapunov.

Then, to get asymptotic stability, LaSalle's Invariance Principle can be used in the following way. Here, since $\dot{\mathbf{q}}^\top \dot{\mathbf{q}}$ is quadratic, it is positive definite, implying that the only solution to $\dot{V} = 0$ from eqn. (3.56) is

$$\dot{V}(\mathbf{q}, \dot{\mathbf{q}}) = 0 \iff \dot{\mathbf{q}} = 0.$$

From eqns. (3.53-3.54), it was shown that

$$\dot{\mathbf{q}} = 0 \quad \Rightarrow \quad \left.\frac{\partial U}{\partial \mathbf{q}}\right|_{\mathbf{q}=\bar{\mathbf{q}}} = \mathbf{0},$$

which must occur at $\bar{\mathbf{q}}$, proven to be unique. Therefore,

$$\dot{V}(\mathbf{q}, \dot{\mathbf{q}}) = 0 \iff \{\mathbf{q} = \bar{\mathbf{q}}, \dot{\mathbf{q}} = \mathbf{0}\},$$

indicating that the equilibrium is the only point at which $\dot{V} = 0$, and so is a positively invariant set for the zero level set of this $\dot{V}$. This meets the conditions of LaSalle's Invariance Theorem, and the system is asymptotically stable.

Finally, for global stability, incorporate the second set of assumptions. Then, since

$$V(\mathbf{q}, \dot{\mathbf{q}}) = T + U + c,$$

$$\lim_{\mathbf{q}, \dot{\mathbf{q}} \to \infty} V(\mathbf{q}, \dot{\mathbf{q}}) = \lim_{\mathbf{q}, \dot{\mathbf{q}} \to \infty} T + \lim_{\mathbf{q} \to \infty} U + c.$$

By definition, the kinetic energy approaches infinity as velocity approaches infinity (for any value of generalized coordinates):

$$\lim_{\dot{\mathbf{q}} \to \infty} T = \infty.$$

The assumption gives radial unboundedness of U, and since T goes to infinity in the second argument, their sum is then radially unbounded with respect to both arguments:

$$\lim_{\mathbf{q},\dot{\mathbf{q}}\to\infty} T = \infty, \quad \lim_{\mathbf{q}\to\infty} U = \infty \quad \Rightarrow \lim_{\mathbf{q},\dot{\mathbf{q}}\to\infty} V(\mathbf{q},\dot{\mathbf{q}}) = \infty.$$

Which meets the condition of radial unboundedness of the Lyapunov candidate, and therefore the system is asymptotically stable in all of $\mathcal{Q}$, or globally if $\mathcal{Q} = \mathbb{R}^n$.

□

3.3.4 A Geometric Interpretation of Passivity and Stability for Systems of Particles

The above proof works for all formulations of Lagrange's equations. However, two concepts are left vague. First is the definition of input and output, $\mathbf{u}$ and $\dot{\mathbf{q}}$, in a physical space. Second is the concept of an 'equilibrium point' corresponding to the equilibrium configuration $\bar{\mathbf{q}}$. The following short section interprets this proof in terms of a (system of) particle(s).

Inputs and Outputs For Systems of Particles

The remark to Lemma 3.1.0.1 established that, for a particle or a system of K particles respectively,

$$\dot{\mathbf{q}}^\top \mathbf{Q} = \mathbf{F} \cdot \mathbf{v}, \qquad \text{or} \qquad \dot{\mathbf{q}}^\top \mathbf{Q} = \sum_{k=1}^{K} \mathbf{F}_k \cdot \mathbf{v}_k,$$

where the particles' positions and velocities $\mathbf{r}_k, \mathbf{v}_k \in \mathbb{E}^3$. For the single particle, the passivity inequality is clear: the system would be passive from an applied force to the resulting velocity of the particle.

For the system of particles, the passivity inequality is less clear, for the same reasons as discussed in Sec. 3.1.2. Each force has its own individual inner product with its own velocity, and there is no sense of an 'input' or 'output' vector for the system as a whole. The best interpretation here is then that of the *single representative particle* abstraction of force and velocity [40, 142]. The system would then be passive with respect to the applied force on the representative particle to the induced velocity of the representative particle.

As such, future work could extend this result to rigid bodies via [41] and systems of rigid bodies via [42].

Equilibrium Configuration for an Equilibrium Point

The vector $\bar{\mathbf{q}} \in \mathcal{Q} \subseteq \mathbb{R}^n$ is an equilibrium point, and the stability proof above shows that the system settles to a configuration defined by this point. What, however, does this represent in terms of the system's position in its configuration space, such as the system of particles in $\mathbb{E}^{3K}$?

Here, the kinetic and potential energies of the system are functions of by the particles' positions and velocities, $\mathbf{r}_k(\mathbf{q})$ and $\mathbf{v}_k(\mathbf{q}, \dot{\mathbf{q}})$. Recall that eqn. (3.15) showed

$$\mathbf{Q} = \frac{\partial U}{\partial \mathbf{q}} \quad \iff \quad \mathbf{F}_k = \frac{\partial U}{\partial \mathbf{r}_k} = \nabla_{\mathbf{r}_k} U \;\; \forall\, k.$$

The stability proof above had $\frac{\partial U}{\partial \mathbf{q}} = \mathbf{0}$ at $\bar{\mathbf{q}}$, so it is clear then that the equilibrium point is at

$$\left.\frac{\partial U}{\partial \mathbf{q}}\right|_{\mathbf{q}=\bar{\mathbf{q}}} = \mathbf{0} \quad \iff \quad \nabla_{\mathbf{r}_k} U(\mathbf{r}_1(\bar{\mathbf{q}}), \ldots, \mathbf{r}_K(\bar{\mathbf{q}})) = \mathbf{0} \;\; \forall\, k, \tag{3.57}$$

or equivalently, from the definition of a conservative force,

$$\mathbf{F}_k(\bar{\mathbf{q}}) = \mathbf{0} \quad \forall\, k,$$

which is Newton's second law.

This fact will be used in Chap. 8: the equilibrium point of the system parameterizes the equilibrium configuration in $\mathbb{E}^{3K}$ given by a Newtonian force balance. A proof of the system stabilizing to its equilibrium point also proves stabilization around a configuration in space (or on a manifold), with an appropriately defined basis.

3.3.5 Conclusion

This chapter reviewed a variety of concepts related to the use of Lagrange's equations for modeling mechanical systems, the use of passivity and dissipativity to analyze those systems, and the provable stability of those systems via such an analysis. The level of detail given here will be crucial for the extension in Chapter 8, where these proofs will be adapted for a new application, with a new concept of potential within a Lagrangian system.

The concepts in this chapter do not represent a new contribution to the literature, but instead, a clarification and standardization on the terminology and notation across different fields (mechanics, signals and systems.) By organizing this dissertation with Chap. 3 here, then Chap. 8 can focus on a novel contribution.

Part II

Kinematics, Statics, and Design

Chapter 4

Tensegrity Spines: Geometry and Movement Goals

When considering the development of a spine for a quadruped robot, two fundamental questions arise prior to undertaking any modeling or design efforts. First, what motions should such a spine perform? And second, what geometry or shape of spine best facilitates those motions?

The following chapter presents an initial analysis of answers to these two questions from the perspective of tensegrity spines. Though there is significantly more depth in these questions, the preliminary answers here allowed for selection of one specific geometry and one set of desired motions, allowing for the development of designs and control systems in later chapters.

4.1 Biological Spines as Bio-Inspiration for Spine Movements

How should a spine assist a robot in its motions? There are a number of potential options, as discussed in Chap. 2; however, most prior work focused on either dynamic running motion of a robot or passive stabilization of a gait. Instead, this research seeks to have a robot walk robustly on uneven terrain, implying the use of less dynamic motions, and of state-space control. Fortunately, as opposed to requiring a more full study of possible spine motions and their effects on a robot, biology gives inspiration as to how spines move. The biomechanics community is clear in this regard. Biological spines perform three motions in a biped or quadruped animal's body:

1. bending in the sagittal plane,

2. bending in the coronal plane, and

3. axial rotation.

These three motion primitives, also sometimes referred to as "flexion/extension", "lateral bending", and "torsion", [85], appear in a wide range of biomechanics literature on vertebral spines of humans and quadruped animals [78, 9, 149, 154, 206, 61, 79, 205]. Fig. 4.1 shows two of these motions (lateral bending and axial rotation). The biomechanics community parameterizes all positions of the spine using these three variables, though the exact resulting state of each vertebra is somewhat unclear in most cases.

(a) Spine bending [61].

(b) Spine rotation [206].

Figure 4.1: Two examples of biomechanics research on spines, showing (a) lateral bending and (b) rotation. These motion primitives, alongside flexion/extension, comprise the three motions of vertebrate spines.

Other vertebrate animals may use their spines in more exotic ways. For example, various robots have been prototyped that use more complicated curves (e.g. sinusoids) for swimming or crawling, including the Salamandra robots [50, 51] which also have four legs. However, the proposed quadruped has more biomechanical similarities to the literature cited here, in light of its intended style of locomotion.

This three variable parameterization, 'bending-bending-rotation', is therefore chosen as a reasonable design and control goal for an actuated tensegrity spine. Later, these will be used to parameterize an example state trajectory for the spine robot considered here. Ch. 6 validates this choice with a practical application: it will be shown that combinations of bending and rotation lift and reposition a quadruped robot's legs, and actively shift its balance. Future work may show benefits to different or more complicated motion patterns.

4.2 Tensegrity Spine Geometries

Creating a shape and cable connection pattern for any tensegrity structure is a non-trivial task. This is the problem of form-finding: designing the size, shape, and placement of the bars of a tensegrity alongside its cable tensions such that it remains in static equilibrium [193]. Generating potential shapes and geometries of tensegrity spines could be done in this way; however, such an undertaking would be challenging both theoretically and numerically.

Instead, since this dissertation seeks to validate the concept of tensegrity spines in general (as opposed to, for example, presenting an optimal design for tensegrity spines), the shapes of spines were designed by hand. Concepts of bio-inspiration were used when envisioning shapes during brainstorming sessions, as were the motion goals of these robots. The subset of designs that were created, and are considered here, do not necessarily represent a thorough investigation into all options, but instead, are intended as a high-level validation of the chosen design: specifically, that no other option the team could envision would have more benefits than the one that is chosen. This section presents the designs that were considered, as simple 3D-printed concepts, without actuation.

In the initial work conducted in this dissertation [169], a single spine geometry was chosen for an analysis. That shape is the “tetrahedral” spine (Fig. 4.2). A spine with nested tetrahedral vertebrae was proposed in the literature as a bio-inspired model for a human spine [66]. It contains sets of horizontal cables along its sides, top, and bottom, which correlate logically to the desired lateral bending / extension-flexion motions.

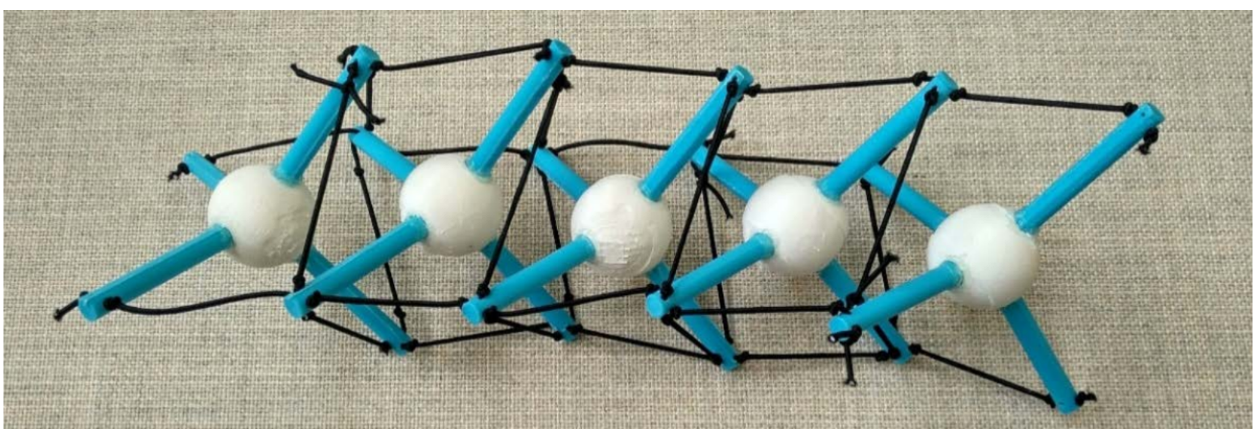


Figure 4.2: Initial concept for the geometry of a tensegrity spine, with tetrahedral vertebrae, nested within each other. This shape has been proposed as a bio-inspired model for a human spine [66]. Image used with permission from [191].

In addition to this spine, a team of students brainstormed the following six vertebra shapes and connection patterns. This team, consisting of Lara Janse van Vuuren [98], Asher Saghian [170], Shu Jun Tan [191], Robel Teweldebirhan [192], and Huajing Zhao [216], constructed the designs in Fig. 4.3. The designs concepts were motivated by the desired motions identified above. All have a similar approximately-horizontal set of cables for bending and extension/flexion, similar to the tetrahedral spine, but those in Fig. 4.3 emphasize sets of cables that would produce axial rotation.

Figure 4.3: Tensegrity spine geometries considered in the brief design study conducted here. The Curved Rod (CR) spine consists of nested, alternating cups. The W-Alternating spine has vertebrae that are mirrored images of the letter "W". The same principle is used for the "H" and "X" spines. The W-Cross and W-Extended spines are variations on the W-Alternating, with different rotations and cable connection patterns between the vertebrae. All are simple 3D-printed prototypes without actuation. Images from [170, 191] with permission.

4.3 Evaluation of Spine Geometries

Given these spine shapes, and the desired motions of the spine, a variety of techniques could be used to quantitatively or qualitatively compare their performance, up to and including simulations and fully actuated hardware prototypes. Such an approach is somewhat impractical given the goals of this dissertation, particularly considering the eventual goal of locomotion. There exists a large, unexplored gap between qualitative behavior of a given spine and its benefits for a quadruped walking over uneven terrain. Therefore, only a brief qualitative analysis was performed, focusing on identifying any clear choices for a spine geometry moving forward.

To do so, a set of high-level metrics for comparing potential spine shapes was created, in collaboration with the same team as above [98, 170, 191, 192, 216]. These metrics are one possible adaptation of the movement goals into design parameters. The team proposed the following to compare the designs:

1. Neutral Zone. As defined in [180], the neutral zone of a spine is the range over which it moves with marginal resistance. For the spines here, the neutral zone is interpreted in terms of flexibility of the spine in a given pretensioned state, for both bending (sagittal or coronal) and axial rotation.

2. Complexity. A common design parameter is the number of features required in a design, which correlates to potential for manufacturing and assembly error. This metric counts (roughly) the number of cable connections required per-vertebra in a design.

3. Component Failure. Spine geometries with a greater tendency to show cracks or materials failure are penalized. For this metric, the team both visually examined the prototypes, and penalized designs with obvious stress concentrations.

4. Stability. As opposed to stability in the control system sense, the team identifies "stability" here as the tendency for a spine to stay upright, or suspend its vertebra without significant deformation, in a given pretensioned state.

5. Cost of Shape Change. One of the goals for this spine was to move along its motions with a minimal amount of actuation in the cables, which correlates directly to the size of actuators required. The team penalized designs that required greater changes in force in the cables in order to produce a given movement.

The above are challenging to evaluate numerically, given the heterogeneity of spine geometries and the significant differences in pretension required for each design. These metrics were therefore analyzed qualitatively, with the "Pugh Matrix" ranking system for designs, which only assigns ratings of -1, 0, or +1 to each design. The team experimented with each prototype by hand, and came to the consensus decisions shown in Fig. 4.4. In the analysis, the tetrahedral spine was chosen as a control, since it has already been proposed as a tensegrity spine model, and since its bio-inspiration gave it the most clear justification for use among all the designs considered.

The analysis in Fig. 4.4 indicates a variety of competing metrics for these spines. For example, more flexible spines were also less stable. Certain spine shapes are obvious choices to eliminate - particularly, the "H" and "W"-Extended spines. Among the remaining options, there is no clear optimal geometry. One hypothesis may be, therefore, that different spines are more optimal for different applications. For example, more flexible spines may be better for quadrupeds walking over larger obstacles with smaller payloads, whereas more stable spines may be better for the reverse (smaller obstacles but heavier payloads.)

As such, the remainder of this dissertation considers the tetrahedral spine geometry. Future work in quantitatively evaluating different spine geometries, particularly with hardware prototypes on walking robots, may suggest a more favorable spine geometry.

PUGH MATRIX	Control: Tetrahedral	Curved Rod	W-Alternating	W-Extended	W-Cross	X	H
Workspace (contact)	0	+1	-1	-1	-1	-1	-1
NZ (Bending Flexibility)	0	-1	-1	-1	-1	+1	-1
NZ (Rotational Flexibility)	0	+1	0	-1	-1	0	-1
Complexity	0	-1	-1	0	-1	-1	-1
Component Failure	0	+1	+1	-1	+1	0	-1
Stability	0	-1	+1	+1	+1	0	0
Cost of Shape Change (CoSC)	0	-1	+1	+1	+1	0	0
Total	0	-1	0	-2	-1	-1	-5
Total (w/o CoSC)	0	0	-1	-3	-2	-1	-5

Figure 4.4: Qualitative evaluation of the different spine designs, using the metrics described above. These were each analyzed by hand as a group effort in [98, 170, 191, 192, 216]. The results indicate a clear competition between the various metrics: more flexible spines are less stable, etc. There is no clear optimal design given this level of analysis, but three designs stand out: the tetrahedral spine, the Curved Rod spine (if cost of shape change/actuation is neglected), and the W-Alternating spine. Image taken from [170] with permission.

Chapter 5

Inverse Statics Optimization for Design and Control of Tensegrity Spine Robots

Designing and controlling tensegrity spines for quadruped robots requires a variety of system models, used for different purposes. This chapter introduces the first of these models: static equilibrium. In doing so, the *inverse statics* problem is formulated, where a tensegrity robot's bodies are positioned in space and a set of cable tensions is calculated that keeps it in that pose. Since many solutions to these cable tensions exist, an optimization program is proposed to calculate these tensions - thus motivating the title to this chapter, *inverse statics optimization*. This procedure can be used as a simple form of open-loop control, by iteratively applying these pseudo-static cable tension solutions to a physical system. Chapter 7 also uses these solutions as part of a closed-loop controller, and the approach here arises again in calculating equilibria in Chapter 8.

The following chapter first introduces the kinematics of tensegrity structures, with the example of the tetrahedral tensegrity spine proposed in Chap. 4. Then, the static equilibrium condition and corresponding inverse statics problem are presented and solved. The resulting algorithm is adapted from the well known *force-density method* [172] to allow its application to this spine model. In that section, a reformulation of the static equilibrium condition is proposed for tensegrity structures with internal bending moments, such as this spine, showing the first feasible solutions for such structures. Finally, a hardware test of the resulting algorithm is performed, and it is shown that open-loop control of these spines tracks a trajectory with low error.

5.1 Tensegrity Kinematics

By Defn. 2.3.1, a tensegrity structure (or robot) is defined as "*a set of rigid bodies suspended in a network of cables in tension such the bodies do not contact each other [179].*"

Generally speaking, those bodies can have complicated shapes [46]. However, a useful specialization, one which is employed in the vast majority of the literature, is that of tensegrity structures as a graph. Informally, a body is represented by a set of nodes, with edges representing stiff members (bars). Cables are also edges between nodes. The kinematics of the structure are therefore reduced to a description of nodes in space. The following section provides a rigorous mathematical description.

However, the following analysis has an important caveat: it is assumed that all bodies in a tensegrity robot are rigid, with no moving joints within them. A variety of past work does not limit to this case. Tensegrity structures may be composed of multiple rigid bodies containing pin joints between them (e.g., [25, 24]. These structures have more in common with jointed mechanisms (example, 4-bar or N-bar mechanisms) than the tensegrity spine robots studied in this dissertation, and the mathematical tools to model them are much more complex. It is arguable that the definition of *tensegrity* here does not capture a sufficiently wide number of structures; however, it does capture the vast majority of tensegrity robots considered in the literature [130, 68, 70, 27, 69, 46, 109, 167, 106, 48, 162, 199, 153, 175, 194, 130, 214, 107].

5.1.1 Graph Definition of a Tensegrity Structure

A tensegrity structure can be defined by a graph $G = (V, E)$, with nodes V and edges E. There are n nodes, where each $\mathbf{a}_i \in V, i = 1...n$ represents a point in $\mathbb{E}^3$ where members of the structure connect. There are m edges, where each edge $e_k \in E, k = 1...m$ represents a structural member that connects two nodes. Forces $P = \{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$, $\mathbf{p}_i \in \mathbb{E}^3$ are applied at each node. The two dimensional case follows similarly to the 3D case here: the structure exists in a d-dimensional physical space (d=2 or d=3.) This representation is the same as an idealized truss, or a *frame*, in structural engineering.

The following assumptions arise from the above definition.

1. Nodal coordinates and external forces are expressed in a Cartesian basis, for example $\mathbf{a}_i = x_i\mathbf{E}_1 + y_i\mathbf{E}_2 + z_i\mathbf{E}_3$. The following chapter abuses notation so that $\mathbf{a}_i = [x_i, y_i, z_i]^\top \in \mathbb{R}^3$, etc.

2. Structural members $e_k,\ k = 1...m$, are perfectly rigid, do not deform, and do not change length. (Cable deformations are back-calculated after solving the initial problem.)

3. Structural members only connect two nodes; a member is exactly one edge.

4. Structural members exist as one of two types: either *bars* of which there are r, which can take either compression or tension loading, or *cables*, which can only take tension loading, of which there are s. This implies $s + r = m$.

Assumption 1 may be modified in the analysis throughout this paper by choosing a different curvilinear coordinate system in $\mathbb{E}^3$. So long as force and moment balance conditions still hold, so do the conclusions below.

As with other graphs, the connections between nodes can be defined by a *connectivity matrix*, also called an indicence matrix or branch-node matrix ([172]) in the literature. The connectivity matrix $\mathbf{C} \in \mathbb{R}^{(s+r)\times n}$ describes how nodes are connected by structural members. If member $i \in \{1, ..., (s+r)\}$ connects nodes k and j, then the k-th and j-th columns in $\mathbf{C}$ are set to 1 and -1 respectively for row i, as in

$$\mathbf{C}^{(a,b)} = \begin{cases} 1 & \text{if} \quad a = i, b = k \\ -1 & \text{if} \quad a = i, b = j \\ 0 & \text{else} \end{cases} \tag{5.1}$$

The structure is then fully defined by the tuple of the configuration matrix, all nodal coordinates, and all nodal forces. Combining each nodal coordinate into a vector according to dimension, for example as $\mathbf{x} = [x_1, \ldots, x_n] \in \mathbb{R}^n$, the tuple is then

$$(\mathbf{x},\ \mathbf{y},\ \mathbf{z},\ \mathbf{p}_x,\ \mathbf{p}_y,\ \mathbf{p}_z) \in \mathbb{R}^n, \quad \mathbf{C} \in \mathbb{R}^{(s+r)\times n}. \tag{5.2}$$

5.1.2 Kinematics of the Tensegrity Graph

It is important to highlight the following result of this discussion:

Tensegrity structures have no position constraints that define a kinematic relationship.

In effect, the bodies are free to move in all dimensions in space. Since the bodies can be positioned arbitrarily, the kinematics problem is trivial: the nodal coordinates can have any value. This is different than having a *solution* with respect to cable tensions for a particular set of nodal coordinates. That problem is the statics problem, below, which does indeed constrain nodal positions, though the constraints are not from positions.

As discussed above, there are some structures also called 'tensegrity' in the literature for which this statement is not true. For example, some structures consider pinned joints between bars [24]. Such structures are not considered here, again because other analysis tools for these mechanisms are more applicable (e.g. [12, 11].

For control, there are a variety of mundane ways to transform a system state into the kinematic parameters above. For example, assume there are b bodies in the structure, and a local frame of nodal positions $\mathbf{a}_k$ in body j. If there is a state vector $\boldsymbol{\xi}$ containing the center of mass of each body $\mathbf{r}_j$ and a set of Euler angles parameterizing the body's rotation, then a node's position in the global frame is $\mathbf{b}_{kj}$ as in

$$\mathbf{b}_{kj}(\boldsymbol{\xi}) = \mathbf{R}_j^{\phi}(\boldsymbol{\xi})\mathbf{R}_j^{\gamma}(\boldsymbol{\xi})\mathbf{R}_j^{\theta}(\boldsymbol{\xi})\mathbf{a}_k + \mathbf{r}_j(\boldsymbol{\xi}),$$

where $\mathbf{R}_j^{(\cdot)}$ are appropriate rotation matrices. The 2D model removes the y, θ, ϕ coordinates, but is otherwise expressed in the same manner.

5.1.3 Kinematics of Tetrahedral Tensegrity Spines

The tetrahedral-vertebra spine from Chap. 4 can be put in the framework described above. Consider first a two-dimensional projection of two vertebrae of that spine, as a reduced-order model for analysis. Fig. 5.1 shows a visualization of this model, with a set of dimensions chosen to match a potential hardware version of the spine, with an initial pose (horizontal) chosen arbitrarily. The graph structure is visible with nodes V in blue and edges E in black (for members in the rigid body) and red (for cable members, which can only take tension loads.)

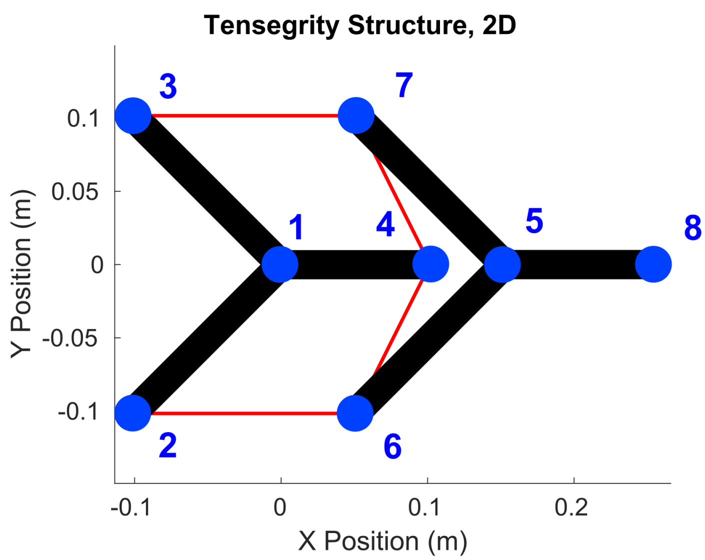


Figure 5.1: Example two-dimensional, two-vertebra tensegrity spine. Blue circles are nodes, black edges are bars (compression), red edges are cables (tension.) This model is a projection into 2D of the tetrahedral spine from Chap. 4.

Writing a connectivity matrix $\mathbf{C}$ for this spine requires differentiating between the cable members and bar members. It is assumed that the first s rows of $\mathbf{C}$ represent cables and the remaining $r = m - s$ rows are for bars [179]. For later analysis in this section, the nodes have been ordered in Fig. 5.1 according to rigid body, in groups of four. Also imposing an ordering on the bars leads to the following connectivity matrix, highlighted according to its block structure:

$$\mathbf{C} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix}, \tag{5.3}$$

where the red rows represent the $s = 4$ cables and blue rows are the $r = 6$ bars within the vertebrae.

Evaluating the movements of such a spine requires fixing part of the structure in space, or otherwise modeling its interaction with the environment. The following analysis assumes that the leftmost vertebra is fixed to the ground and does not move, and also provides the reaction forces to keep the body in static equilibrium. Columns 1-4 of $\mathbf{C}$ are highlighted with lighter colors to emphasize the nodes for the fixed left vertebra.

5.2 Tensegrity Statics using the Force Density Method

Given a configuration of the tensegrity (nodal coordinates), the forward and inverse statics problems correspond to calculating cable tensions so that the system is in static equilibrium. Both forward and inverse require an expression for the static equilibrium condition. This section derives that constraint for the tensegrity spines considered here, and along the way, introduces a new reformulation of the constraint to account for internal bending moments within a tensegrity body.

The inverse statics solutions can be used for control (later sections,) since the dynamics models in later chapters will assume that the control input arises from adjusting the cables' parameters. Specific models of the cables, such as linear elasticity, rest length, etc., are needed to describe the control input precisely. Consequently, this section focuses on calculating cable tensions independent of a specific model, under the assumption that the control input can be calculated from these tensions.

For structural networks, including cable networks, the static equilibrium condition can be readily specified by the *force density method* [172]. It has, by extension, been used for tensegrity systems as networks of force-carrying structural members in tension or compression [195]. This section briefly derives the static equilibrium condition for a structure using force density, then reformulates it for the problem at hand.

In the following, the character $\mathbf{q}$ does not refer to the generalized coordinates for Lagrange's equations as in chapter 3. This is an unfortunate clash of terminology between fields. The field-specific terminology is kept in this chapter, since the work here does not consider any dynamics of the system, and thus Lagrange's equations do not appear.

For an initial analysis of the statics of this system using the force density method, the following assumptions are imposed in addition to those for the kinematics:

5. Structural members are one-dimensional; they have zero volume and mass. (The mass of the structure is instead distributed to the nodes, for purposes of analysis.)
6. Forces are only exerted at nodes. This includes gravitational forces due to the structure's mass.
7. All connections between members (i.e., the nodes) are friction-less pin joints.
8. All nodes are fixed in space; they do not translate.

The final assumption differentiates the statics problem from the form-finding problem as discussed in Chap. 2.4. Specifically, if no static solution is found for the specified nodal coordinates and forces, then this chapter does not consider adjusting the positions of the nodes.

5.2.1 The Force Density Method for General Networks

Assume that the nodal positions and external forces are given. Let the force in member i be F_i, which acts along its axis. Define the *force density* vector $\mathbf{q}$ as

$$\mathbf{q} = [\ q_1,\ q_2,\ q_3,\ \dots\ q_{s+r}\ ]^\top \in \mathbb{R}^{(s+r)}, \tag{5.4}$$

such that if member i has length ℓ_i,

$$q_i = F_i/\ell_i. \tag{5.5}$$

As seen in [68, 172, 193, 195] the force balance condition in d=3 dimensions for static equilibrium of the structure can then be stated as

$$\begin{aligned} \mathbf{C}^\top \text{diag}(\mathbf{q})\mathbf{C}\mathbf{x} &= \mathbf{p}_x, \\ \mathbf{C}^\top \text{diag}(\mathbf{q})\mathbf{C}\mathbf{y} &= \mathbf{p}_y, \\ \mathbf{C}^\top \text{diag}(\mathbf{q})\mathbf{C}\mathbf{z} &= \mathbf{p}_z. \end{aligned} \tag{5.6}$$

With some manipulation, as also discussed in [68, 172, 195], eqn. 5.6 can be reorganized as

$$\mathbf{A}\mathbf{q} = \mathbf{p}, \tag{5.7}$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{C}^\top \mathrm{diag}(\mathbf{Cx}) \\ \mathbf{C}^\top \mathrm{diag}(\mathbf{Cy}) \\ \mathbf{C}^\top \mathrm{diag}(\mathbf{Cz}) \end{bmatrix} \in \mathbb{R}^{(nd)\times(s+r)}, \tag{5.8}$$

$$\mathbf{p} = \begin{bmatrix} \mathbf{p}_x \\ \mathbf{p}_y \\ \mathbf{p}_z \end{bmatrix} \in \mathbb{R}^{(nd)}. \tag{5.9}$$

The d=2 dimensional case follows similarly.

Here, $\mathbf{A}$ and $\mathbf{p}$ are constants. Therefore, eqn. (5.7) is a set of linear equations in $\mathbf{q}$. A value for $\mathbf{q}$ that satisfies (5.7) can then be obtained in a variety of ways, e.g. by a quadratic program [68], which produces a set of equilibrium cable forces for a given desired pose. If solutions exist, then $\bar{\mathbf{u}}$ can be calculated (for example, in the linear elastic case) as

$$\bar{u}_i = \ell_i - \frac{\ell_i q_i^*}{k_i}. \tag{5.10}$$

5.2.2 Existence and Uniqueness of Equilibrium Solutions

Issues can arise when attempting to apply eqn. (5.7) to tensegrity robots. Both the existence of solutions, and their uniqueness, must be considered.

Existence of Solutions

First, the spine considered in this work intuitively requires internal bending moments to be present in static equilibrium, violating Assumption 7. Consider, for example, the center node of the tetrahedral spine: it must resist the induced bending from the cables at the tips of the tetrahedron. In such a case, eqn. (5.7) is inconsistent in almost all poses of the spine, and no solutions exist.

Additional insight exists when examining the rank of $\mathbf{A}$. In works such as [68, 169, 70], the tensegrity structure has many more cables and bars than nodes, such that $(s+r) > (nd)$. Thus, $\mathbf{A}$ is wider than it is tall, with a null space dimension of at least $(s+r) - (nd) > 0$. However, for example with the two-dimensional two-vertebra tensegrity spine, $(s+r) = 10$ and $(nd) = 16$, so $\mathbf{A}$ is taller than it is wide, has an empty null space in almost all poses, and no solutions exist other than $\mathbf{q} = \mathbf{0}$ with $\mathbf{p}_x, \mathbf{p}_y = \mathbf{0}$.

This rank deficiency issue for static equilibrium is discussed in the literature on tensegrity structures in the context of geometry [35] and energy methods [49]. Algorithms exist for determining if a structure would have static equilibrium solutions [155, 128]. However, addressing this issue usually consists of adding cables or changing the geometry of the tensegrity structure itself (via form-finding, e.g. [195]), which is not possible given the problem statement in this work. It has also been suggested that a rigid body be replaced with an equivalent rigid frame with no bending moments; however, such a task is very challenging.

No force-density formulations of static equilibrium for tensegrity structures with internal bending moments is present in the literature.

Uniqueness of Solutions

When eqn. (5.7) has a nonzero null space, an infinite number of solutions exist. Informally, the structure can increase or decrease its stiffness by antagonistically tightening or loosening cables. The inverse statics optimization problem presented below proposes searching this null space for some optimal set of tensions.

The question of uniqueness of the equilibrium calculated via the above, with respect to spatial coordinates, is an open problem for many tensegrity systems. This chapter considers static equilibrium, but an equilibrium point in the robot's state space must be considered with respect to the equations of motion. For comparison, this chapter asks: what cable tensions, and therefore rest lengths of springs, keep the structure in equilibrium? An analysis of equilibrium points asks: given a set of rest lengths of flexible cables, what nodal coordinates achieve static equilibrium? The latter requires analyzing the dynamic equations of the cables, and is considered in Chapter 8 for general networks.

For some tensegrity structures, particularly with bars only and when pretensioned with linear-elastic cables, it has been shown that a static equilibrium is an asymptotically stable local equilibrium in the state space [185]. It has also been shown qualitatively that some tensegrity structures have multiple equilibria when position constraints on the bodies are imposed [203]. Identifying equilibria of this dynamic system is left for future work. Relatively speaking, for the spines considered here, the local region where only one equilibrium exists is "large:" no multiple equilibria have been found during all the research in this dissertation.

5.2.3 Rigid Body Reformulation of the Force Density Method

The well-known results in preceding sections did not produce any solutions for the spines in this work. Therefore, the following adaptation of the node-graph formulation of the force density method is proposed.

Here, the equilibrium condition for a tensegrity structure is expressed as a force and moment balance per rigid body. The derivation below produces a new set of linear equations for static equilibrium. Doing so neglects the internal stresses within the bodies, consistent with the assumption of rigid-body equations of motion. The process below is therefore described as a "rigid body reformulation", although prior statics work uses the term 'rigid' in different contexts [35]. The following is the first static equilibrium constraint for tensegrity robots such as these.

The rigid body reformulation requires the following two assumptions:

1. The tensegrity robot consists of b rigid bodies each with the same number of nodes, $\eta = n/b$.

2. The columns of $\mathbf{C}$ are block-ordered according to rigid body: nodes are assigned an ordering in blocks of η.

These assumptions are demonstrated in the highlighted $\mathbf{C}$ in eqn. (5.3). The robot has $b = 2$ bodies, with $\eta = 4$ nodes each, so that columns 1-4 and 5-8 correspond to each body. This is similar to the repeated 'cells' of a larger tensegrity, as the term is used in [136].

Preliminaries

Throughout the following section, certain patterns and quantities are used repeatedly. Here, they are briefly introduced and derived, so as to shorten exposition in the sequel.

Combining together sets of rows or columns of a matrix will be referred to as *collapsing* the matrix. One way to do so is pre-multiplication by a matrix that is generated using the Kronecker product in combination with the identity matrix and ones vector. For example, the matrix

$$\begin{aligned} \mathbf{K} =& (\mathbf{I}_2 \otimes \mathbf{1}_4^\top) && (5.11)\\ =& \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} && (5.12) \end{aligned}$$

would collapse rows 1-4 and 5-8 of another matrix $\mathbf{A} \in \mathbb{R}^{8\times n}$, as in $\mathbf{KA} \in \mathbb{R}^{2\times n}$.

Reformulating the constraint to neglect internal stresses requires that the force densities for the cables, $\mathbf{q}_s$, be separated from the force density vector for all members, $\mathbf{q}$. Recall that the $\mathbf{C}$ matrix has been organized with the first s rows as cables (for example, highlighted in red in eqn. (5.3)) and the remainder as rigid members (bars.) Therefore, the first s elements of $\mathbf{q}$ are $\mathbf{q}_s$. Relating these two can be done by a matrix $\mathbf{H}$, which will be used as

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}_s \\ \mathbf{0}_{r\times s} \end{bmatrix} \in \mathbb{R}^{(s+r)\times s}, \qquad \mathbf{H}\mathbf{q}_s = \begin{bmatrix} \mathbf{q}_s \\ \mathbf{0}_r \end{bmatrix}, \qquad \mathbf{q}_s = \mathbf{H}^\top \mathbf{q}. \tag{5.13}$$

Both the cable lengths and the square of the cable lengths are needed below. They can be obtained using the $\mathbf{C}$ matrix and node vectors, which are briefly given here using basic linear algebra operations for implementation speed in an algorithm. The signed distances in each dimension for each cable are

$$\mathbf{d}_x = \mathbf{H}^\top \mathbf{C}\mathbf{x}, \quad \mathbf{d}_y = \mathbf{H}^\top \mathbf{C}\mathbf{y}, \quad \mathbf{d}_z = \mathbf{H}^\top \mathbf{C}\mathbf{z}. \tag{5.14}$$

The lengths of each cable ℓ_i can then be calculated quickly by concatenating these vectors, and performing a row-wise 2-norm:

$$\mathbf{D} = \begin{bmatrix} \mathbf{d}_x & \mathbf{d}_y & \mathbf{d}_z \end{bmatrix} \in \mathbb{R}^{s\times 3}. \tag{5.15}$$

$$\ell_i = ||\mathbf{D}_{i,*}||_2, \tag{5.16}$$
$$\boldsymbol{\ell} = [\ell_1, \quad \dots, \quad \ell_s]^\top \in \mathbb{R}^s \tag{5.17}$$
$$\mathbf{L} = \text{diag}(\boldsymbol{\ell}) \in \mathbb{R}^{s \times s}. \tag{5.18}$$

A vector of the squared length of each cable can be found without using a norm:

$$\boldsymbol{\ell}^2 = \mathbf{H}^\top \text{diag}(\mathbf{Cx})\mathbf{Cx} + \mathbf{H}^\top \text{diag}(\mathbf{Cy})\mathbf{Cy} + \mathbf{H}^\top \text{diag}(\mathbf{Cz})\mathbf{Cz} \quad \in \mathbb{R}^s \tag{5.19}$$
$$\mathbf{L}^2 = \text{diag}(\boldsymbol{\ell}^2) \quad \in \mathbb{R}^{s \times s}. \tag{5.20}$$

Two-dimensional versions of these quantities follow similarly.

Removal of Anchor Nodes

For non-mobile tensegrity robots, for which reaction forces with the environment are not relevant to the problem, these forces can be neglected by eliminating the constraints associated with the body's *anchor* nodes. Anchor nodes are those that are assumed to support arbitrary reaction forces. Such an assumption is consistent with the body being rigidly fixed to the ground, and takes advantage of the static indeterminacy of the structure. For example, the 2D spine robot in Fig. 5.1 has its leftmost vertebra assumed to be rigidly fixed to the ground, nodes 1-4 act as anchor nodes: any reaction forces may exist there, such that the remainder of the robot is in equilibrium. Note that this does *not* imply the presence of reaction moments at these nodes; only reaction forces are allowed to vary.

The material in this section does not apply when external reaction forces are specified by another part of the control problem. For example, if the robot was connected to other moving components (for example, when the hips and shoulders in a quadruped with a spine are moving or the shoulder joint in [114] is moving), then the anchoring nodes have their reactions specified beforehand also.

To remove anchoring nodes from the equilibrium constraint, specify a vector of binary numbers with each entry corresponding to a node, where a 1 signifies that a node is to be kept and a 0 for an anchor node:

$$\mathbf{w} \in \{0, 1\}^n. \tag{5.21}$$

For example, removing the rigidly-fixed body of the 2D single-vertebra spine (Fig. 5.1) corresponds to

$$\mathbf{w} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}^\top. \tag{5.22}$$

Setting $\mathbf{w} = \mathbf{1}_n^\top$ recovers the original problem formulation.

Denote the number of remaining nodes as $h \leq n$, equivalent to the number of nonzeros in $\mathbf{w}$. Then, diagonalize this matrix, and remove the zero-ed rows:

$$\mathbf{W} = \text{nonzero rows}(\text{diag}(\mathbf{w})) \in \{0,1\}^{h \times n}. \tag{5.23}$$

Pattern this matrix out along the diagonal, corresponding to the number of dimensions d of the problem,

$$\mathbf{W}_f = \mathbf{I}_d \otimes \mathbf{W} \in \{0,1\}^{hd \times nd}. \tag{5.24}$$

The equilibrium constraint (eqn. 5.7) with the anchors removed can then be posed as

$$\mathbf{W}_f \mathbf{A} \mathbf{q} = \mathbf{W}_f \mathbf{p} \tag{5.25}$$

Conceptually, this eliminates the force balance constraint at any nodes with a 0 in $\mathbf{w}$.

Force balance per rigid body

The nodal force balance, eqn. (5.7), can be converted into a force balance per-body by combining the above concepts. Assume that $\mathbf{w}$ has been chosen such that the assumptions in the problem hold: i.e., after eliminating anchor nodes, the tensegrity robot has b bodies and a configuration matrix is block-organized by bodies of η nodes, and that $\mathbf{W}_f$ from eqn. (5.24) is used as per eqn. (5.25). Note here that if a body was removed via $\mathbf{W}_f$, then b must be decremented accordingly from the original problem formulation. For example, the structure in Fig. 5.1 has $b = 1$ body remaining, though $b = 2$ were initially present.

The rows corresponding to the remaining bodies can be collapsed by a matrix

$$\mathbf{K} = \mathbf{I}_{db} \otimes \mathbf{1}_\eta^\top \in \mathbb{R}^{(db) \times (db\eta)}, \tag{5.26}$$

which combines the per-node balance (each row) for each body in all d directions. Performing this operation on both the internal member forces (left-hand side) and external forces (right-hand side) from eqn. 5.7) produces

$$\mathbf{K}\mathbf{W}_f \mathbf{A} \mathbf{q} = \mathbf{K}\mathbf{W}_f \mathbf{p}. \tag{5.27}$$

Now, the forces in the bar members must be removed, since they have no physical meaning. Introducing $\mathbf{H}$ does so, as per eqn. (5.13),

$$\mathbf{K}\mathbf{W}_f \mathbf{A}\mathbf{H} \mathbf{q}_s = \mathbf{K}\mathbf{W}_f \mathbf{p}. \tag{5.28}$$

Therefore, the constraint from eqn. (5.7) can now be posed as

$$\mathbf{A}_f \mathbf{q}_s = \mathbf{p}_f, \tag{5.29}$$

where

$$\mathbf{A}_f = \mathbf{K}\mathbf{W}_f \mathbf{A}\mathbf{H} \quad \in \mathbb{R}^{(db) \times s}, \tag{5.30}$$

$$\mathbf{p} = \mathbf{K}\mathbf{W}_f \mathbf{p} \quad \in \mathbb{R}^{(db)}, \tag{5.31}$$

which is a linear system of equations, just as is the original.

Moment balance per rigid body

A moment balance for each body is now required, since the robot is treated as a set of rigid bodies. To do so, this section calculates the induced moment by each cable (and external force) on each node, then combines those moments per-body in the same way as with the force balance.

Moments due to all forces can be summed around any point in the structure (not necessarily the centers of mass) in static equilibrium. A convenient point is therefore the origin, so that the moment arms are simply the nodal coordinates. Moments can then be expressed using matrix multiplication as the following.

In three dimensions, the moment applied by the force in member i upon node k at coordinates $\mathbf{a}_k$, calculated around the origin, is

$$\mathbf{M}_i^k = \mathbf{a}_k \times \mathbf{F}_i \in \mathbb{R}^3.$$

Recall that the cross product can be expressed using a skew-symmetric matrix, as is common in robotics applications [140]. In this case, let

$$\mathbf{B}_k = \begin{bmatrix} 0 & -z_k & y_k \\ z_k & 0 & -x_k \\ -y_k & x_k & 0 \end{bmatrix},$$

then

$$\mathbf{M}_i^k = \mathbf{B}_k \mathbf{F}_i.$$

From the previous sections, $\mathbf{Aq} \in \mathbb{R}^{(nd)}$ are the forces applied by each member at each node, expressed component-wise in each direction. Therefore, the moment in each direction due to all members on each node (for d=3) is

$$\mathbf{M} = \mathbf{BAq} \in \mathbb{R}^{3n}, \tag{5.32}$$

with

$$\mathbf{B} = \begin{bmatrix} \mathbf{0} & -\mathbf{Z} & \mathbf{Y} \\ \mathbf{Z} & \mathbf{0} & -\mathbf{X} \\ -\mathbf{Y} & \mathbf{X} & \mathbf{0} \end{bmatrix} \in \mathbb{R}^{3n \times 3n} \tag{5.33}$$

$$\mathbf{X} = \mathrm{diag}(\mathbf{x}) \tag{5.34}$$

$$\mathbf{Y} = \mathrm{diag}(\mathbf{y}) \tag{5.35}$$

$$\mathbf{Z} = \mathrm{diag}(\mathbf{z}) \tag{5.36}$$

In two dimensions, the moment applied by member i acting on one of its anchors at node k is a scalar quantity:

$$M_i^k = -y_k F_i^x + x_k F_i^z.$$

So, by defining the moment arm matrix in two dimensions,

$$\mathbf{B}^{2D} = [-\mathbf{Y} \;\; \mathbf{X}] \quad \in \mathbb{R}^{n \times 2n}, \tag{5.37}$$

a similar result arises:

$$\mathbf{M} = \mathbf{B}^{2D}\mathbf{A}\mathbf{q} \quad \in \mathbb{R}^{n}. \tag{5.38}$$

As with the force balance per body, the moment contributions from the bar members can be removed and the moment balance at the anchor nodes can be removed. Applying the same arguments as with eqn. (5.30) modifies eqn. (5.32) to become, per body, for the cables only,

$$\mathbf{M}_c = \mathbf{K}\mathbf{W}_f\mathbf{B}\mathbf{A}\mathbf{H}\mathbf{q}_s \quad \in \mathbb{R}^{3b}. \tag{5.39}$$

In d=2 dimensions, since there is only one moment per body, the moments from the cables can be expressed as

$$\mathbf{K}^{2D} = \mathbf{I}_b \otimes \mathbf{1}_\eta^\top, \tag{5.40}$$

$$\mathbf{M}_c^{2D} = \mathbf{K}^{2D}\mathbf{W}\mathbf{B}^{2D}\mathbf{A}\mathbf{H}\mathbf{q}_s \quad \in \mathbb{R}^{b}. \tag{5.41}$$

The same can be done with the external forces,

$$\mathbf{p}_m^{2D} = \mathbf{K}^{2D}\mathbf{W}\mathbf{B}\mathbf{p}, \tag{5.42}$$

$$\mathbf{p}_m = \mathbf{K}\mathbf{W}_f\mathbf{B}\mathbf{p}. \tag{5.43}$$

So, the moment balance for the system is

$$\mathbf{A}_m\mathbf{q}_s = \mathbf{p}_m, \tag{5.44}$$

where $\mathbf{A}_m$ is defined as

$$\mathbf{A}_m = \mathbf{K}\mathbf{W}_f\mathbf{B}\mathbf{A}\mathbf{H} \qquad \text{or} \qquad \mathbf{A}_m = \mathbf{K}^{2D}\mathbf{W}\mathbf{B}^{2D}\mathbf{A}\mathbf{H}. \tag{5.45}$$

Combined static equilibrium constraint

The force and moment balance conditions, eqns. (5.29) and (5.44), can then be combined by stacking the systems of equations, as in

$$\mathbf{A}_b = \begin{bmatrix} \mathbf{A}_f \\ \mathbf{A}_m \end{bmatrix}, \quad \mathbf{p}_b = \begin{bmatrix} \mathbf{p}_f \\ \mathbf{p}_m \end{bmatrix}, \tag{5.46}$$

so that the full static equilibrium condition is

$$\mathbf{A}_b \mathbf{q}_s = \mathbf{p}_b. \tag{5.47}$$

Though the static equilibrium constraint has been fundamentally transformed from a per-node force balance into a per-body force and moment balance, the constraint is still linear. This allows the application of the same approaches to solving eqn. (5.47) as are done in prior literature for eqn. (5.7).

For the example two-dimensional, two-vertebra spine, eqn. (5.47) has solutions, whereas none existed using the vanilla force-density formulation. Its constraint matrix $\mathbf{A}_b \in \mathbb{R}^{6\times4}$ has rank 3 in all poses in the tests below, thus a null space of dimension $4-3=1$. Simulations below also show that eqn. (5.47) is consistent. So, even though $\mathbf{A}_b$ is still a 'tall' matrix, the rigid body formulation admits solutions here whereas the node-graph formulation does not.

5.3 Inverse Statics Optimization

With the static equilibrium condition in hand, an inverse statics optimization problem can be posed to find the optimal cable tensions that satisfy eqn. (5.47). The term 'inverse statics' is used here to emphasize that a control system chooses a $\mathbf{q}_s$. For comparison, the forward statics problem would specify the cable model by fixing $\bar{\mathbf{u}}$, and solving for the $\mathbf{q}$ that evolves naturally due to the applied load $\mathbf{p}$. Here, instead, the trajectory generation problem solves for an optimal $\mathbf{q}_s$ for a given load $\mathbf{p}$, and back-calculates the corresponding inputs $\bar{\mathbf{u}}$.

The following linear elastic model of the cables is used in the remainder of this chapter. For cable i with total length (node-to-node) as ℓ_i, with (scalar) force F_i, a controller is assumed to provide an input u_i, where

$$F_i = \kappa_i(\ell_i - u_i) \tag{5.48}$$

with $\kappa_i \in \mathbb{R}_+$ as the linear spring constant. Here, the sign convention is chosen so that tension forces are positive, in consideration of the same sign convention for $\mathbf{q}$.

This is equivalent to a motor retracting or extending a flexible cable. In the tests in sec. 5.5, a mechanical spring will be attached to a stiff cable that is wound around a spool. Later, given this model of the cables, the total potential energy is formulated to be used as an objective function.

5.3.1 Constraints on Equilibrium Cable Tension Solutions

The following section introduces two inter-related constraints to be imposed on the inverse statics problem. First, the cables cannot have negative tension: a minimum is imposed on the force densities for all cables. Second, given a model of the cables, an input saturation constraint is applied that prevents negative rest lengths (since negative length is not possible.) These two constraints provide both upper and lower bounds on the force density.

Minimum Tension

In order to enforce positive cable force densities, a scalar lower-bound constant is introduced, with its corresponding column vector,

$$c \in \mathbb{R}_+, \quad \mathbf{c} = c\mathbf{1}_s \in \mathbb{R}^s_+ \tag{5.49}$$

which are used to define the minimum force density in every cable as $q_i \geq c$, $\forall\ i = 1...s$. The constraint is then (in standard optimization program form)

$$-\mathbf{q}_s \leq -\mathbf{c}. \tag{5.50}$$

The magnitude of c determines the amount of pretension in the structure: a higher c, the stiffer the structure. From a control systems perspective, setting $c = 0$ is therefore possible but not advisable, since $q_i = 0$ indicates that a cable has become slack and the equations of motion of the robot change as a consequence. Choosing a larger c therefore operates the robot in a region of its state space that's further away from the switching surface of these hybrid dynamics, but a high pretensioning constraint comes at the design cost of a higher total potential energy in the cables. In addition, setting c too high may conflict with the input saturation constraint below, and result in no solutions.

Input Saturation Constraint

A maximum force density constraint can be added to the problem in order to prevent input saturation. Input saturation arises from the physical condition that an actuator cannot command negative rest length; it must always be that

$$u_i > 0 \qquad \forall\ i = 1 \ldots s. \tag{5.51}$$

For purposes of robustness, introduce a minimum cable rest length of u_i^{min}, or in vector form for all inputs, $\mathbf{u}^{min}$. Selecting a $\mathbf{u}^{min}$ can be done from physical considerations of a robots' cables: there are mechanical components at the ends of a cable, such as springs, that could not be retracted by a motor. Attempting to do so would cause collisions between the actuator and the robot's cable connectors. For example, if a mechanical spring plus its connectors attached at the end of a cable had some length ℓ_i^0, then a reasonable choice would be $u_i^{min} = \ell_i^0 + \epsilon$, where ϵ is some safety factor.

Using $\mathbf{u}^{min}$ also allows the inequality to be non-strict, as with the cable pretensioning constraint. Via the relationship between rest length and cable force (the input model from eqn. 5.48), this inequality becomes

$$u_i^{min} \leq \ell_i - \frac{F_i}{\kappa_i}, \tag{5.52}$$

and with the definition of force density (eqn. 5.5), can be written linearly as

$$u_i^{min} \leq \ell_i - \frac{\ell_i}{\kappa_i} q_i \quad \iff \quad \ell_i q_i \leq \kappa_i (\ell_i - u_i^{min}). \tag{5.53}$$

Eqn. (5.53) can therefore also be understood as $F_i \leq F_i^{max}$.

Define the matrix $\boldsymbol{\varkappa} = \text{diag}(\boldsymbol{\kappa}) \in \mathbb{R}_+^{s \times s}$, which contains the spring constants along its diagonal. The constraint then becomes, in vector form,

$$\mathbf{L}\mathbf{q}_s \leq \boldsymbol{\varkappa}(\boldsymbol{\ell} - \mathbf{u}^{min}). \tag{5.54}$$

This input saturation constraint constraint can be combined with the pretensioning constraint (eqn. (5.50)) to become

$$\begin{bmatrix} \mathbf{L} \\ -\mathbf{I}_s \end{bmatrix} \mathbf{q}_s \leq \begin{bmatrix} \boldsymbol{\varkappa}(\boldsymbol{\ell} - \mathbf{u}^{min}) \\ -\mathbf{c} \end{bmatrix}. \tag{5.55}$$

The use of the combined constraint 5.55 guarantees that, if solutions exist, they result in valid inputs to the control problem while also maintaining tension on all cables.

Finally, it is notable that if $\mathbf{u}^{min} = \mathbf{0}$, the input saturation constraint simplifies to the remarkable expression

$$\mathbf{q}_s \leq \boldsymbol{\kappa}, \tag{5.56}$$

which emphasizes the deep connection between selection of appropriate spring constants for the robots' design and the range of allowable control inputs.

5.3.2 The Inverse Statics Optimization Program

Since eqn. (5.47) is a linear equality constraint, and eqn. (5.55) is a linear inequality constraint, solving for an optimal $\mathbf{q}_s^*$ can be done with a convex program. In the following, an energy-based objective function is derived, then a quadratic program is proposed that solves the problem.

Total Cable Potential Energy

Given the linear elastic cable model of eqn. (5.48), the total potential energy in the system can be formulated. As a function of the control input, the potential energy in the cable system is

$$PE = \frac{1}{2}\sum_{i=1}^{s} \kappa_i(\ell_i - u_i)^2. \tag{5.57}$$

Relating the energy in a cable to its force density can be done by using eqn. (5.48), as

$$q_i = \frac{\kappa_i}{\ell_i}(\ell_i - u_i) \qquad \Rightarrow \qquad (\ell_i - u_i)^2 = q_i^2 \frac{\ell_i^2}{\kappa_i^2},$$

and therefore that

$$PE_i = \frac{1}{2} q_i^2 \frac{\ell_i^2}{\kappa_i}.$$

The sum in eqn. (5.57) is quadratic in $\mathbf{q}_s$, and has the form

$$PE = \frac{1}{2}\mathbf{q}_s^\top \boldsymbol{\varkappa}^{-1}\mathbf{L}^2\mathbf{q}_s. \tag{5.58}$$

The scaling factor of $\frac{1}{2}$ will be dropped for convenience below, since it does not affect the optimizer $\mathbf{q}^*$.

Solution via a Quadratic Program

With the above quadratic cost and linear constraints, the following quadratic program can be used to solve the inverse statics problem for one pose of the robot.

$$\mathbf{q}_s^* = \arg\min_{\mathbf{q}_s} \ \mathbf{q}_s^\top \mathbf{R}\mathbf{q}_s \tag{5.59}$$

$$\text{s.t.} \ \mathbf{A}_b\mathbf{q}_s = \mathbf{p}_b \tag{5.60}$$

$$\mathbf{S}\mathbf{q}_s \leq \mathbf{v}, \tag{5.61}$$

with

$$\mathbf{R} = \boldsymbol{\varkappa}^{-1}\mathbf{L}^2, \qquad \mathbf{S} = \begin{bmatrix} \mathbf{L} \\ -\mathbf{I}_s \end{bmatrix}, \qquad \mathbf{v} = \begin{bmatrix} \boldsymbol{\varkappa}(\boldsymbol{\ell} - \mathbf{u}^{min}) \\ -\mathbf{c} \end{bmatrix}. \tag{5.62}$$

Similar optimization problems have been termed “inverse kinematics” in the literature [68, 70]. However, this approach contributes two advances in comparison. First, the reformulated constraint (5.60) allows for solutions to exist here for a much wider array of tensegrity robots. No solutions existed for tensegrity spines using the approach from [68]. Second, the above problem accounts for input saturation, preventing infeasible solutions and enabling safer implementation on hardware robots.

5.4 Inverse Statics Optimization as Open-Loop Control for Tensegrity Robots

The optimization program (5.59-5.61) can generate an optimal $\mathbf{q}_s^*$ for a desired pose $\boldsymbol{\xi}$ of the robot (sec. 5.1.2). Open-loop control can therefore be performed by solving for a $\mathbf{q}_s^*(t)$ over a time series of poses, $\bar{\boldsymbol{\xi}}(t)$, $t = 1 \dots T$, and applying the corresponding control inputs $\bar{\mathbf{u}}(t)$ for each pose. The implied assumptions, such as pseudo-static motions, may or may not be valid for different problem settings.

5.4.1 Inverse Statics Optimization Algorithm

The inverse statics procedure first calculates the constraint eqn. (5.60) for each $\bar{\boldsymbol{\xi}}(t)$ in the reference state trajectory, via eqns. (5.8)-(5.47). Then (5.59-5.61) is solved for each $\mathbf{q}_s^*(t)$ via an optimization solver, and $\bar{\mathbf{u}}(t)$ is calculated using eqns. (5.48) and (5.5). The procedure follows Algorithm (1), where ℓ_i and k_i are the length and spring constant for cable i as per eqn. (5.48).

Algorithm 1 Inverse Statics Optimization

procedure INVSTAT($\bar{\boldsymbol{\xi}}$)
  for $t \leftarrow 1, T$ **do**
    $\mathbf{A}_b \leftarrow \mathbf{A}_b(\mathbf{x}(\bar{\boldsymbol{\xi}}(t)),\ \mathbf{z}(\bar{\boldsymbol{\xi}}(t)))$
    $\mathbf{p}_b \leftarrow \mathbf{p}_b(\mathbf{x}(\bar{\boldsymbol{\xi}}(t)),\ \mathbf{z}(\bar{\boldsymbol{\xi}}(t)))$
    $\mathbf{q}_s^*(t) \leftarrow \mathit{OptForceDens}(\mathbf{A}_b,\ \mathbf{p}_b)$ ▷ solve (5.59-5.61)
    for $i \leftarrow 1, s$ **do**
      $\bar{u}_i(t) \leftarrow \ell_i(t) - \dfrac{\ell_i(t) q_i^*(t)}{k_i}$
  return $\bar{\mathbf{u}}$

Algorithm (1) was implemented with MATLAB's `quadprog` solver in the software that accompanies this research[1].

5.4.2 Inverse Statics Optimization For Motion of a Tensegrity Spine

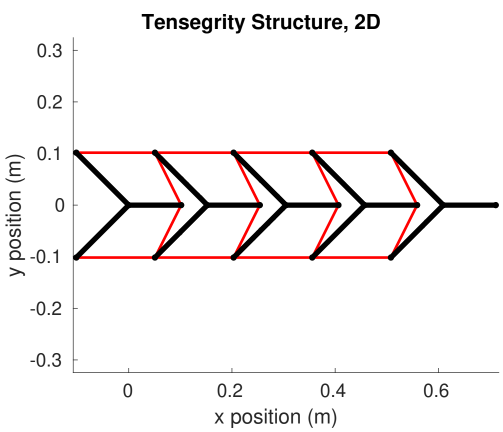
Algorithm 1 was used to generate the first control solutions for a tensegrity spine performing the bending motions suggested in Chap. 4. The following considers a five-vertebra version of the two-dimensional spine from Fig. 5.1 for clarity of exposition. Three dimensional versions of this problem follow similarly. In both cases, the $\mathbf{C}$ matrix is extended from that of sec. 5.1.3 to include sets of cables and bodies with the same block structure as for two vertebrae.

[1]https://github.com/apsabelhaus/tiso

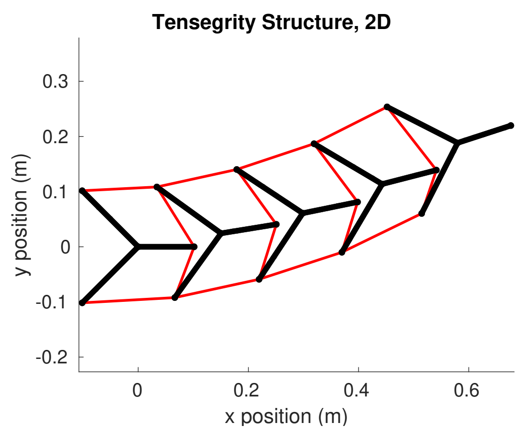
Problem Setup for a Two-Dimensional Bending Spine

The spine is assumed to have its leftmost vertebra fixed to the ground, with the remaining vertebrae under the influence of gravity. Therefore, external forces $\mathbf{p}_y$ are $-mg$ at each node. The removal of the leftmost vertebra via the $\mathbf{w}$ vector eliminates the force balance at the fixed nodes, so no reaction forces are needed and the reaction force calculation is skipped.

Many different motions are possible that might be called 'bending', as motivated by the bio-inspiration for this problem. The biomechanics literature does not commonly define a specific sequence of states of each vertebra in a spine when undergoing a motion. Therefore, the test in this chapter chooses one possible state trajectory among many: each vertebra is swept out along an arc from an initial horizontal configuration. Future work will examine different trajectories for representing the three motion primitives from Chap. 4 and their implications for design and control.



(a) Initial pose of the five-vertebra, 2D spine, for the inverse statics optimization example.



(b) Final pose of the five-vertebra, 2D spine, for the inverse statics optimization example.

Figure 5.2: Setup and kinematics for the inverse statics optimization example, with the five-vertebra two-dimensional spine. The test calculated optimal cable tensions for $T = 30$ poses between the initial (a) and final pose (b).

Fig. 5.2 shows this spine in its initial configuration and final configuration. The test calculated the optimal cable tensions for $T = 30$ poses between the initial and final. The state trajectory $\bar{\boldsymbol{\xi}}$ along this path consists of translations and rotations of each of the four moving vertebrae. Each of the bars of the vertebra were $\ell_b = 4$ inches long, and each vertebra was translated in the x-direction from the previous by $1.5\ell_b$ in their initial poses. Specifically, counting the first moving vertebra as $j = 1$, with the state vector $\bar{\boldsymbol{\xi}}(t)$ containing center-of-mass coordinates $\bar{x}_j(t)$ and $\bar{y}_j(t)$,

$$\bar{x}_j(0) = (1.5\ell_b)j \ . \tag{5.63}$$

The vertebrae are swept upward along an arc for each pose. The initial x-translations also define the radius of the rotation: $r_j = \bar{x}_j(0)$. Consequently, the reference positions of each vertebra over time, $\bar{x}_j(t)$ and $\bar{y}_j(t)$, are:

$$\bar{x}_j(t) = r_j \sin(\beta_j(t)), \quad \bar{y}_j(t) = r_j \cos(\beta_j(t)). \tag{5.64}$$

The maximum sweep angle for each vertebra $\beta_j(T) := \beta_j^{max}$ was defined as 4/5 of the previous vertebra's maximum sweep angle, counting backwards from the rightmost vertebra J. Specifically,

$$\beta_J^{max} = \frac{\pi}{10}, \qquad \beta_j^{max} = \frac{4}{5}(\beta_{j+1}^{max}), \qquad j = (J-1)\dots 1. \tag{5.65}$$

In addition, the desired rotation $\bar{\theta}_j(t)$ of each vertebra about its inertial z-axis is defined to be the same as the sweep angle $\beta_j(t)$ for that vertebra. This keeps the x-axis of each vertebra's local frame aligned with the vector r_j:

$$\bar{\theta}_j(t) = \beta_j(t). \tag{5.66}$$

The spring constant in the cables was chosen to be 4.8 lbf/in, from a set of example springs used later for prototyping. The minimum force density was chosen to be $c = 0.5$, and the minimum rest length was $u_i^{min} = 1$ cm.

Inverse Statics Optimization Results for a Two-Dimensional Bending Spine

Applying Alg. 1 to the problem above generates a set of optimal forces, force densities, and inputs. The optimal cable forces are shown in Fig. 5.3, plotted against the 30 chosen poses between the initial and final configurations. Solutions existed in all poses.

Here, the individual cables (as extrapolated from Fig. 5.1) are given labels. The horizontal top (HT) and horizontal bottom (HB) cables are the first two rows of the example $\mathbf{C}$ matrix. The remaining two are termed 'saddle' cables, from the observation that the corresponding cables in three-dimensional version of this spine resemble a saddle point.

Three important observations are possible from this data, each of which has an effect on the design and control approaches in the following sections. First, the optimal forces can occasionally be counter-intuitive upon initial inspection. As the spine bends upward, the tensions decrease in most cables (a downward trend evident in the figure.) Here as with the rest of this dissertation, the cables each adjust the structure's pose in complicated ways. However, more analysis often leads to intuition: in particular, as the spine bends up, the total moment arm from the mass of the tip vertebra becomes lower (versus the gravity vector), so reduced tensions on the top cables are a reasonable result.

Second, there are much larger variations between cables, and between sets of cables, than between poses. The leftmost set of cables (red) must support the cantilevered load of the remaining vertebrae, for example. Even within one set, some cables may support most of the structure's weight. For example, the bottom saddle cables (SB) are the only ones which

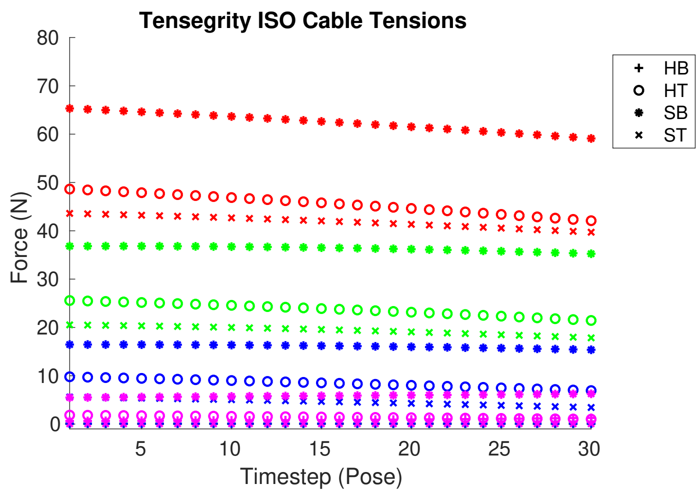


Figure 5.3: Inverse statics optimization results for the two-dimensional, five-vertebra spine performing the motion in Fig. 5.2. Cables are colored according to sets between vertebra, in order from left to right (red, green, blue, magenta.) Marker indicates cables within the set, numbered according to the **C** matrix from sec. 5.1.3. Here, HB = horizontal bottom, HT = horizontal top, SB = saddle bottom, ST = saddle top. There are significant variations according to cable, but smaller variations between poses.

provide a force in the positive y direction, and thus carry a large load. On the other hand, the horizontal bottom (HB) cables are only barely tensioned due to the chosen c, with forces so small as to not appear in the plot.

Finally, small variations in tension lead to significant changes in pose. The bend from Fig. 5.2 produces difficult-to-distinguish trends in most cables in Fig. 5.3. This has significant implications for calibration: with a small mismatch in initial cable length, errors will propogate throughout a test. Adjusting the initial lengths of cables before performing a test - manual calibration - is the most significant source of error in the hardware experiments in this dissertation.

5.5 Hardware Testing and Verification

The new inverse statics optimization algorithm was tested in a hardware experiment. These tests were designed to validate the reformulated rigid-body equilibrium constraint in the optimization problem. Results also consider the validity of using open-loop inputs for control.

The test setup consisted of a single vertebra, constrained in a two-dimensional plane, of the same form as Fig. 5.1. Choosing this simple model reduced many of the hardware difficulties of three dimensional structures (such as state tracking in higher dimensions) as well as with multiple vertebrae (such as actuator placement.) Though the test was performed on a reduced-order model, the very promising results below indicate the applicability of the approach to higher dimensions with more vertebrae.

All work in this section was performed collaboratively with Kimberly Sover and Jacob Madden. Designs and CAD models are used in this dissertation with permission from all team members.

5.5.1 Test Setup

The full test setup is shown in Fig. 5.4. The test consists of a single vertebra, mechanically constrained to two dimensions via a series of acrylic plates and bearings. The vertebra is of the same geometry as described in sec. 5.1.3, with four cables. As the motors adjust the lengths of the cables, a camera tracks the position of the markers on the vertebra and calculates its state.

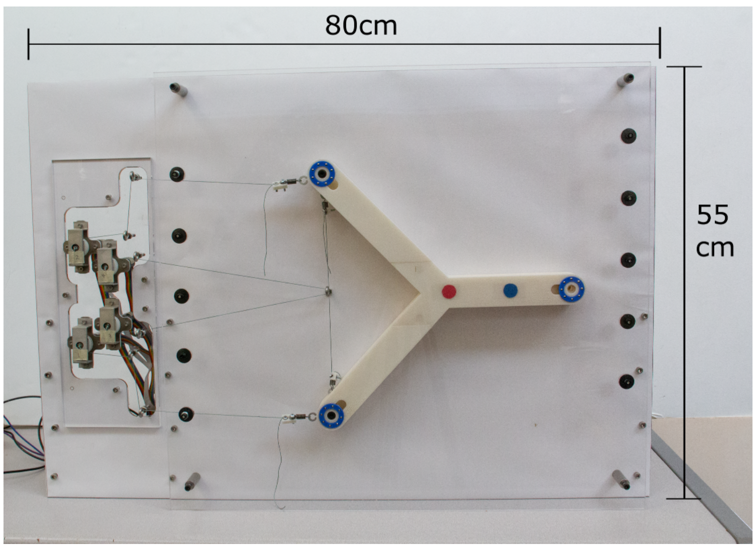


Figure 5.4: Hardware test setup for validating the proposed inverse statics optimization procedure. Motor and cable assembly on left. Shown here is the vertebra pinned in its calibration pose, via two rods that insert through the rear plate of the test setup, located underneath the red and blue tracking markers.

Mechanical Assembly

The mechanical assembly of the test setup consists of a single 3D-printed vertebra, placed in between a white-colored back plate and a clear acrylic front plate (Fig. 5.5). With the vertebra placed between the plates, and with thrust bearings to reduce friction, the vertebra was constrained to move in only three degrees of freedom: horizontal (x), vertical (y), and rotation (expressed in the vertebra's local frame, θ).

A series of standoffs and washers were used to align the two plates so the vertebra could move freely. The vertebra contains notches for attaching springs, to which the cables were

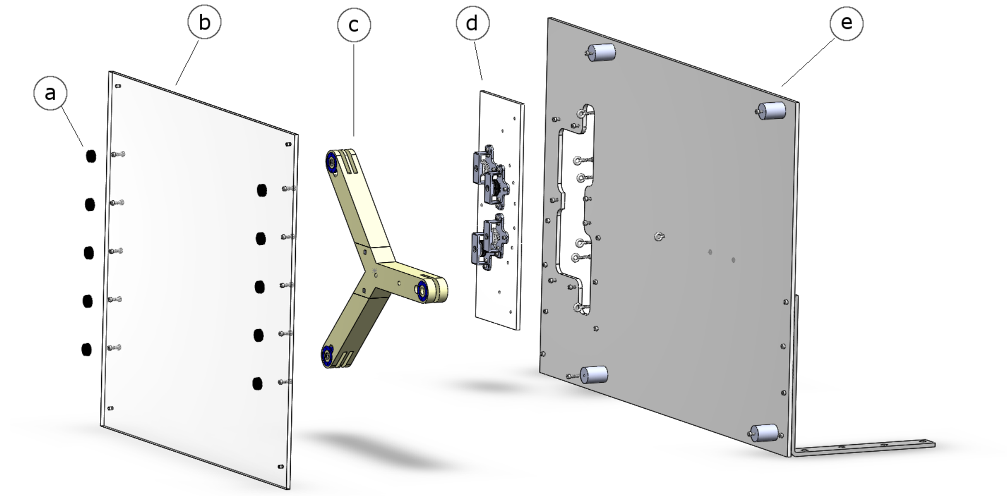


Figure 5.5: CAD render of the test setup assembly, exploded view, demonstrating its components. (a) Calibration markers for the computer vision tracker, fixed to the clear front plate. (b) Clear front plate, constraining the vertebra to two dimensional movement. (c) 3D-printed vertebra, with thrust bearings in blue. (d) Motor assembly, attached to the rear plate, controlling the vertebra's cables. (e) Rear plate supporting the test setup.

fixed via a screw adjustment mechanism. Cables were wound around spools attached to the shafts of each motor so that motor rotation provided the input u_i. A series of eye bolts were used to route the cables such that their orientation coincided with Fig. 5.1. The motors were four 30W Maxon brushless DC motors, fixed to the rear plate.

Sizing of all components was chosen to minimize calibration error (with a smaller test setup) against mechanical challenges for a much larger setup. The vertebra had bar lengths of 20cm for the two rear bars, and 16cm for the front cantilevered bar with the markers. It had a mass of 0.5kg. Spring constants were 4.8 lbf/in for the horizontal cables, and 25.4 lbf/in for the saddle cables. These were chosen to balance calibration error (stiffer springs) against spring deformation (more flexible springs.)

Electronics

The brushless DC motors were operated with the VESC 6.0 motor controller[2], one for each motor, configured for sensorless operation. These controllers accepted a pulse-width modulation signal for velocity input. A Cypress PSoC microcontroller was attached to the encoder on each motor, and controlled the system by varying that PWM signal to each motor driver. The microcontroller was connected to a laptop computer via USB, communicating over the

[2]https://vesc-project.com/

UART protocol. A webcam was also connected to that computer for the computer vision system.

Software

For control of the motors, the microcontroller generated the PWM signal via a SISO PID controller. The tracked output was the error in motors' rotation, the difference between recorded encoder ticks versus the commands sent over UART. Controller gains were calibrated as per the VESC's documentation. The laptop computer sent cable rest lengths over UART at specified intervals.

A computer vision system was used to track the state of the vertebra as it moved. The camera was pointed at the clear front plate of the test setup, with the same perspective as Fig. 5.4. The system used OpenCV libraries in ROS, running on the attached laptop computer.

Calibration of the camera frame was performed at the start of each test. The eight black markers fixed to the front acrylic plate (Fig. 5.5) were detected, and a homography transformation was automatically calculated between the camera frame and the test setup frame. That transformation was applied to all pixel values to determine the locations of the blue and red markers in the test setup frame.

The camera was operated at 640 x 480 resolution for use with the computer vision libraries. This implied a tolerance of approximately +/- 1 mm in the computer vision system, which corresponds to the width of one camera pixel in the hardware frame.

The state tracking procedure, after the automatic calibration, consisted of blob detection of the red and blue markers on the vertebra. These dots, placed at known locations in the vertebra's local frame, were used to calculate the center of mass and local rotation of the vertebra. No statistical state estimation or smoothing was implemented; the raw pixel data was used to calculate the state for each frame.

The attached laptop computer recorded the calculated vertebra state, and recorded timestamps for both states and cable rest length commands sent to the microcontroller. Data was then post-processed to correlate timestamps.

5.5.2 Experimental Results

The chosen reference trajectory of the vertebra was similar to that of sec. 5.4.2, but with the vertebra sweeping and angle from $-\pi/8$ to $\pi/8$. The inverse statics optimization algorithm was executed, results were saved, and then replayed during each hardware test. There were $N = 10$ tests performed with the same cable rest length calculations.

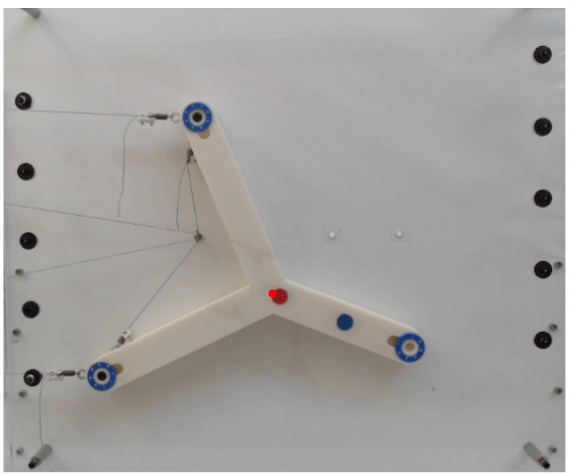
Test Procedure

The procedure for one test consisted of first calibrating the cable lengths. The vertebra was pinned in a known pose (Fig. 5.4) by fixing it in place with rods through the rear test setup plate. Each cable was then manually tensioned by visual and physical inspection, with the goal to have each cable only slightly tensioned, approximating zero force as best as possible. The offset from the cable rest lengths in that known pose was incorporated with the inverse statics optimization results before transmission to the microcontroller.

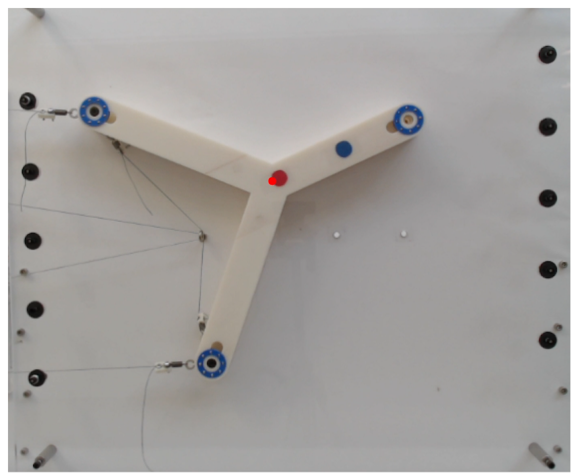
After calibrating the vertebra's cables for each test, the vertebra was unpinned and allowed to move freely, and the cable rest length commands were sent while tracking with the computer vision system. Each test was discretized into 80 poses, with a one-second pause between commanding each pose.

Vertebra Movement Results

Fig. 5.6 shows the initial pose and final pose from one representative test. During the 80-second test, the vertebra moved slowly between these positions. Visually, this test verifies that a bending motion is not only possible with a tensegrity spine, but that control inputs for such a motion can be calculated analytically.



(a) Initial pose.



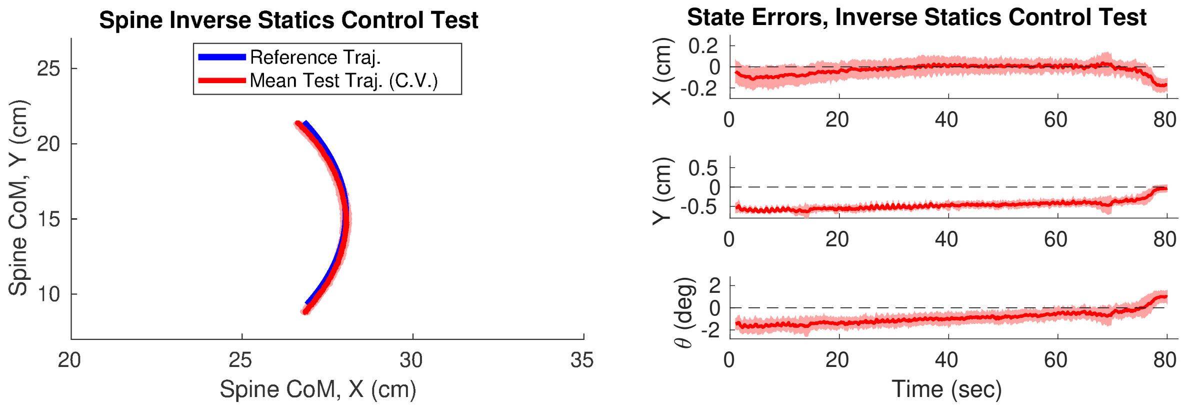
(b) Final pose.

Figure 5.6: Representative hardware test, (a) initial pose and (b) final pose of the vertebra during its trajectory. Red dot indicates the center of mass of the vertebra, calculated by the computer vision system, as it moves along its sweeping motion.

Data Analysis

The recorded states and tracking errors are plotted in Fig. 5.7. Results show extremely close tracking to the expected series of poses.

Fig. 5.7a plots the center of mass of the vertebra in the x-y plane as it moves through its test. Points are averaged per-pose over all 10 tests, with standard deviation plotted as a pink shaded circle behind each mean. Blue points correspond to the poses specified to the



(a) Averaged trajectory of the center of mass of the vertebra.

(b) State tracking errors averaged over the 10 tests.

Figure 5.7: Data analysis from the 10 experiments. Means in red, standard deviations in pink shading, reference trajectory in blue. The vertebra followed its proscribed path very closely in all experiments, with some minor drift as expected from an open-loop control test. Errors are extremely small in comparison to the test geometry, and in many cases (e.g. the x-errors) are below the 1 mm-per-pixel resolution of the computer vision system.

inverse statics optimization algorithm. Qualitatively, the vertebra moves as expected, with some offset as expected from an open-loop control test.

Fig. 5.7b plots the errors for each individual state, again with red as the mean and pink shading for standard deviation. Calculated state errors were close to measurement tolerance for the test setup, in some cases less than 1mm. All errors were extremely small overall in comparison to the size of the body and the length of the trajectory. For example, the robot moves 12.5 cm in the y-direction during its test, so with the largest y-error as 0.5 cm (Fig. 5.7b), the largest errors in the test are on the order of 4% total movement.

Both plots show an observable, small drift. This is partially due to the test setup's calibration procedure and partially due to the approximations used in the statics model. The pins which held the vertebra in place allowed for some play, so the vertebra's configuration pose was not always fixed. The drift may be due to the unmodeled dynamics of the system, including the weight of the springs and friction as the vertebra pulls upward. However, the results appear to justify the validity of the new static equilibrium constraint, giving correct poses up to some expected test error.

5.6 Conclusion

This chapter established an inverse statics algorithm for calculating control inputs that position a tensegrity robot in a desired pose. The inverse statics optimization problem introduced a reformulation of the static equilibrium constraint, to extend prior work to a much wider range of tensegrity robots. This section also addresses practical considerations for the control problem, such as input saturation, while maintaining linear inequality constraints for a convex program. Simulation examples show how the approach can be used to calculate inputs for tensegrity spines in bending motions. A hardware test validated the new static equilibrium constraint, and the inverse statics optimization algorithm which uses it, on a reduced-order hardware prototype. Open-loop control results demonstrated that this simple approach provides a reasonable method for simple control of tensegrity spines.

The work in this chapter focused on two-dimensional spines; however, physical prototypes of quadruped robots require three-dimensional spines. The core contribution in this chapter (the reformulated equilibrium constraint) applies equally to both two-dimensional and three-dimensional structures, so it is expected that the approach will also be valid when used in three dimensions. However, the sources of error described in Sec. 5.5.2 will become more problematic as the dimension and number of vertebrae increases within a spine model. As more cables are introduced, pretensioning and calibration may cause offsets in vertebra movement that are more challenging to address.

As such, future work could incorporate closed-loop control with this open-loop approach. Closing the loop using a state tracking controller would help reject the modeling error that would arise from mis-calibration. Part III of this dissertation proposes ways to do so.

Chapter 6

Design, Simulation, and Testing of an Actuated Tensegrity Spine for Quadruped Robots

This chapter presents the first prototype of an actuated tensegrity spine for quadruped robots. The robot Laika, named after the first dog in space, is designed with this tensegrity spine as its body. The following chapter presents the first set of hardware designs for Laika's spine, a physical prototype of those designs, and a set of tests in both simulation and hardware which show the benefits of the three motion primitives from Chap. 4.

First, actuation is discussed, and using the inverse statics optimization approach from the previous chapter, it is justified that Laika's spine can be controlled in a bending motion with a reduced number of actuators. Then, a prototyping method for tensioning the spine's cables, via a pretensioned elastic lattice, is introduced. Hardware designs of the prototype spine, and a version of Laika with stiff (non-actuated) legs, are presented. Simulations are performed that show specific combinations of bending and rotation can lift each of Laika's legs. Hardware experiments are performed that verify which combinations of motions lift each leg. Ongoing work and future work seeks to design a version of Laika with moving legs, so that walking locomotion can be performed.

6.1 Motion Primitives with Underactuation

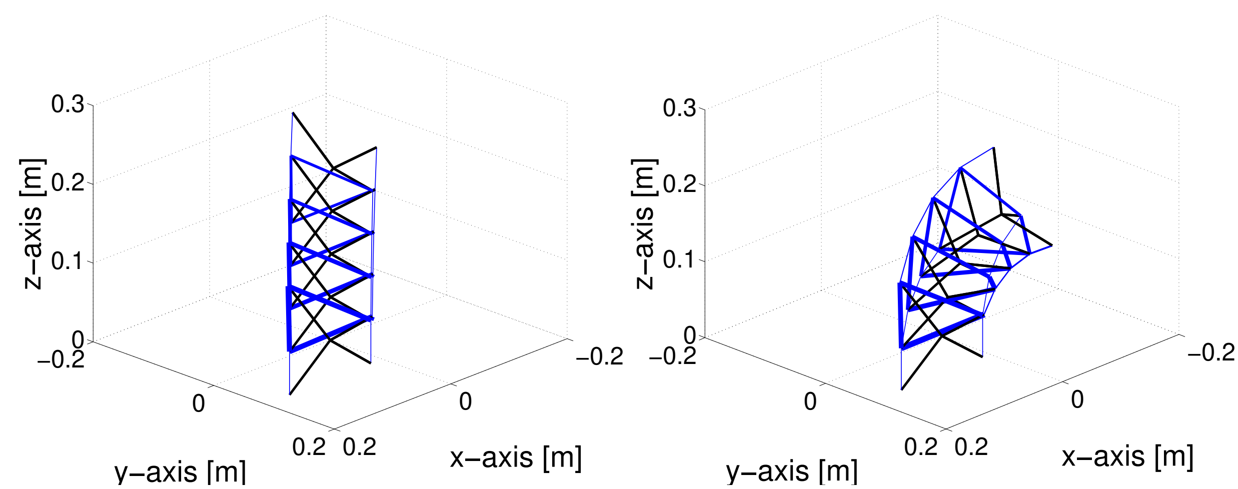
Prototypes of tensegrity robots often suffer from the challenges discussed during the inverse statics optimization test in Chap. 5. Specifically, there are a large number of cables within a structure, and controlling the lengths of each cable presents a significant mechanical design challenge. Motivated by other work in tensegrity robots which leaves some cables unactuated [167], this section evaluates the example tetrahedral tensegrity spine for patterns in motion that may allow for reduced actuation.

In the context of this work, the term 'underactuation' is used in two meanings. As with most tensegrity robots, the dynamics model of the spine has fewer possible control inputs (cables) than states (rigid body poses and velocities), and is therefore underactuated in the control systems sense of the term. However, in the context of a cable-driven system, the spine can also be *underactuated* in the sense of leaving some cables uncontrolled. The designs below will also propose to control multiple cables with one actuator, with such coupled actuation also informally implying a form of *underactuation*.

6.1.1 Inverse Statics Testing of a 3D, Five-Vertebra Tetrahedral Spine

A series of tests were conducted with a three-dimensional version of the tetrahedral-vertebra spine. These tests examined bending motions under gravity, with the spine oriented vertically. The choice of a vertical spine arises from previously-motivated research, before the application of the spine to a quadruped robot was selected; however, the principles of underactuation discussed here apply to the motion itself, not the orientation.

Using the same static equilibrium calculation as in Chap. 5, cable tensions in the spine were calculated in a series of poses in Fig. 6.1. Cable force data was then examined for trends. A linear relationship was observed between poses for the vertical cables, which run along the sides of the spine, corresponding to the horizontal cables from Chap. 5's spines.



(a) Initial pose of the vertical spine. (b) Final (bent) pose of the vertical spine.

Figure 6.1: Inverse statics simulation for a vertical five-vertebra spine performing a bending motion. This orientation was chosen for analysis based on earlier studies of the spine; however, the observations from motion primitives are generalizable to other orientations.

To examine the relationship between cable segments, consider if actuators were only located at the bottom-most vertebra's nodes for each set of vertical cables. In this case, a cable attached to a subsequent vertebra from the single actuator at the bottom, routed through the structure, could be modeled by adding the lengths of each vertical segment. Along one edge of the spine, let the length of a cable segment between two adjacent vertebrae be ℓ_i^{i+1}. Then total length of the cable that connected the bottom vertebra to vertebra $j = 2...5$ would be of the form L_j as in

$$L_j = \sum_{i=1}^{j-1} \ell_i^{i+1},$$

i.e., the sum of all the edge lengths. By assumption of linear elastic behavior of the cables, if all cables have the same spring constant, the rest length (control input $u := L_j^0$) follows this same additive relationship. For the analysis here, each vertebra had a mass of 1.6kg, and cables had spring constants of 1220 N/m.

Examining the change in cable rest length generated by the inverse statics procedure, and adding lengths along the edge, allows the calculation of relative length change between each L_j^0. Taking the difference between the initial pose (Fig. 6.1a) and final pose (Fig. 6.1b), and normalizing against the shortest cable (L_2), gives the results in Tables 6.1 and 6.2. Table 6.1 shows the results for the edge that is elongating in the test, whereas table 6.2 shows the corresponding result for the cable that is contracting.

Cable Segment j	$j = 2$	$j = 3$	$j = 4$	$j = 5$
Initial Rest Length L_j^0 [m]	0.04748	0.09497	0.14250	0.23910
Final Rest Length L_j^0 [m]	0.05978	0.11960	0.17940	0.18990
ΔL_j^0 [m]	0.01230	0.02463	0.03690	0.04920
Ratio	$\times$ 1.00	$\times$ 2.00	$\times$ 3.00	$\times$ 4.00

Table 6.1: Rest Length Ratio Computation with Elongating Vertical Cables. The shortest cable is arbitrarily normalized for illustration purposes.

Cable Segment j	$j = 2$	$j = 3$	$j = 4$	$j = 5$
Initial Rest Length L_j^0 [m]	0.04610	0.09219	0.13820	0.18410
Final Rest Length L_j^0 [m]	0.03056	0.06059	0.08745	0.11170
ΔL_j^0 [m]	0.01554	0.03160	0.05075	0.07240
Ratio	$\times$ 1.00	$\times$ 2.03	$\times$ 3.26	$\times$ 4.66

Table 6.2: Rest Length Ratio Computation with Contracting Vertical Cables. The shortest cable is arbitrarily normalized for illustration purposes.

For Table 6.1, taking any two poses along the trajectory leads to the same ratio of rest lengths. For Table 6.2, the ratio of rest lengths does change between subsequent poses, albeit by small amounts.

6.1.2 Motivation for Multiple-Spool Actuators

The relationships in Tables 6.1 and 6.2 demonstrate a fixed (or slowly varying) relationship between the amount of cable along vertical edges of the spine that would be retracted or extended during a bending motion. Consequently, if only a bending motion is desired (as opposed to arbitrary placements of vertebrae in space), a single actuator might be used. One actuator with multiple spools attached, with diameter ratios corresponding to those above, with cables routed to the appropriate vertebrae, would perform the desired amount of rest length change for a single vertical edge.

There are a variety of caveats to these observations. For example, the inverse statics algorithm varied the saddle cables' force densities also, so positioning the spine into the desired pose would not occur at the above rest lengths unless those saddle cables were also actuated. In addition, a comparison of the horizontal spine versus this vertical spine implies that gravitational forces could change the result: a horizontal spine has a different gravity vector than the vertical.

However, as discussed in Chap. 5's simulation of the five-vertebra spine, the 'bending' motion primitive does not have a precise meaning in terms of specific vertebra positions. Therefore, as opposed to attempting to re-create a very specific (arbitrarily chosen) motion, a proof-of-concept prototype could instead choose an actuation scheme that leads to bending-like motion and define the result as 'bending.' Such is the approach taken here. Future work would investigate exactly how fixed retraction/extension ratios move the spine.

6.2 Prototype Construction with Elastic Lattices

In addition to the challenge of actuation, accurately pretensioning a large three-dimensional spine presents significant challenges. As discussed in the hardware tests of Chap. 5, manually adjusting a tensegrity's cables to the desired rest lengths can be the most significant source of error in a test. One method to address assembly error in this situation is instead to manufacture a structure with components that do not require manual adjustment. This section presents such a method.

As originally proposed by other members of the Berkeley Emergent Space Tensegrities lab, in collaboration with the team working on Laika's spine, strips of elastic material could be laser-cut to specific lengths and then attached to a tensegrity structure as a set of passive (un-actuated) cables. Taking a larger sheet of material and cutting multiple segments attached to one another results in a 'lattice' of elastic material [47]. All work in this subsection was performed in collaboration with Lee-Huang Chen, Mallory C. Daly, Lara A. Janse van Vuuren, Hunter J. Garnier, Mariana I. Verdugo, Ellande Tang, Carielle U. Spangenberg, and Faraz Ghahani.

A prototype of the five-vertebra 3D spine was created that used an elastomer lattice 'jacket' that wraps around the vertebrae (Figure 6.2). The prototype's vertebrae consist of 3D-printed octahedra as the center node, into which thin-walled aluminum rods are inserted, and a bolt and screw act for 'endcaps' that clamp onto the lattice and fit into the rods. A fully assembled spine tensegrity structure utilizes one lattice, twenty bolt endcaps, and twenty of the aluminum rods. Figure 6.2 illustrates the sequence required to assemble the spine tensegrity structure using one full lattice and five vertebrae. The material used for this lattice is a 0.0625 in. thick silicone rubber, 60A durometer.

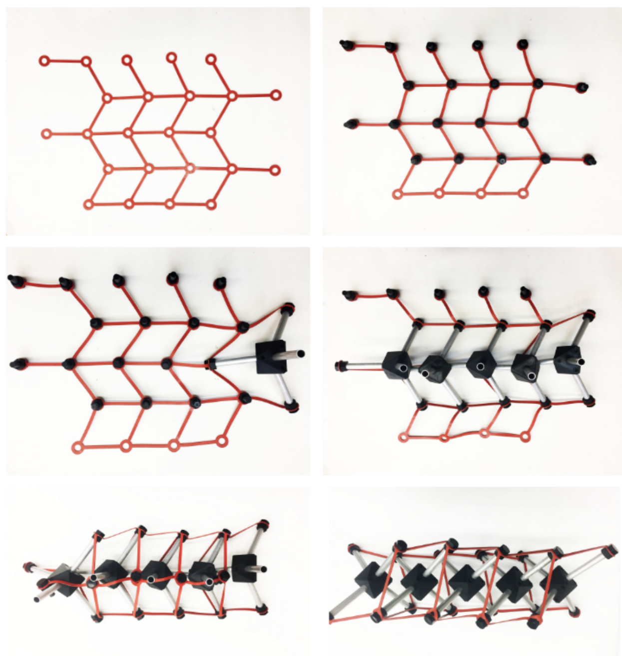


Figure 6.2: Step-by-step assembly sequence of a five-vertebra spine with a single-piece elastic lattice 'jacket.' Top left image shows the two-dimensional cutout from the silicone elastomer. Credit to Hunter Garnier, used with permission.

An initial static prototype of the spine with legs is shown in Fig. 6.3. This image demonstrates how the use of the lattice evenly pretensions the robot.

6.3 Actuated Prototype Mechanical Design

With suitable methods to address the major design and assembly concerns for an actuated tensegrity spines, a prototype was created that could demonstrate the bending and rotation motions studied here. This initial prototype of Laika consists of a spine supported by rigid hips, shoulders, and legs. As its spine bends and rotates, Laika's center of mass shifts in three dimensions. These motions are shown to lift each of the robot's legs, in preparation for locomotion. Actuated legs will be added in future work to test the spine with locomotion.

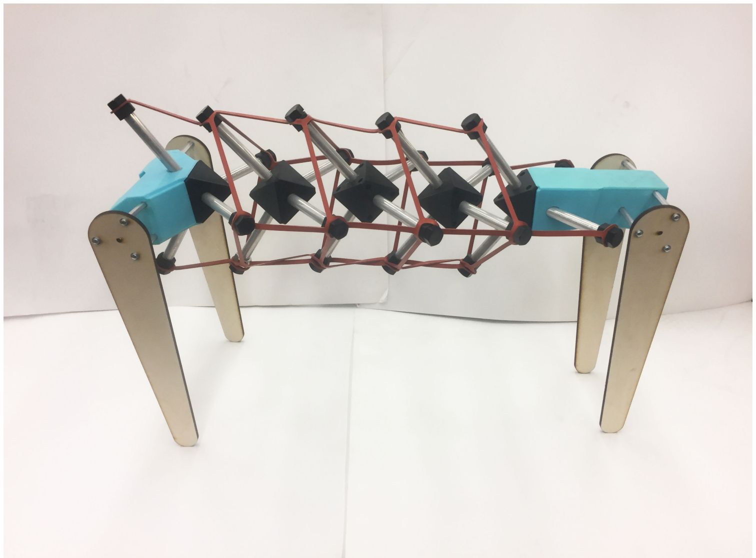


Figure 6.3: Prototype of a tensegrity spine for a quadruped robot, manufactured using an elastic lattice for a pretensioned cable structure. This prototype does not include actuators, but demonstrates how the lattice achieves even pretensioning. Credit to Hunter Garnier, used with permission.

6.3.1 Actuated Spine Topology

Laika's spine is composed of five vertebrae, out of which the second and the fourth vertebrae are *active*, with actuators, and the third vertebra in the center provides rotation. Vertebrae 1 and 5 connect to the hips and shoulders, and are *passive*, without actuation. The active vertebrae contain motors which adjust the lengths of these cables, as motivated by Sec. 6.1. Fig. 6.4 shows the lattice network of cables for the spine, where the attachment points for the cables are labeled based on the vertebra and the side of the spine they belong to.

The cables are numbered and labeled for identification in later simulation tests. These designations are chosen based on a biological view of the robot. The horizontal cables connect the different vertebrae on each of the four sides of the spine: top, bottom, left, or right, respectively (Fig. 6.4). Each set of horizontal cables consists of four individual cables, running from one vertebra's motor to attachment points on each of the other vertebrae. For example, in Fig. 6.4, four separate cables run from the actuated end at T2 to T1 and T3...5.

6.3.2 Hardware Overview

The current prototype of Laika attaches the actuated spine the same a 3D printed hip and shoulder that stand on stiff, rapid-prototyped legs as in Fig. 6.3. Varying sizes of legs can be attached. A representative model of Laika (Fig. 6.5) is approximately 52.8 cm long and stands 41.4 cm tall, and weighs 1.62 kg.

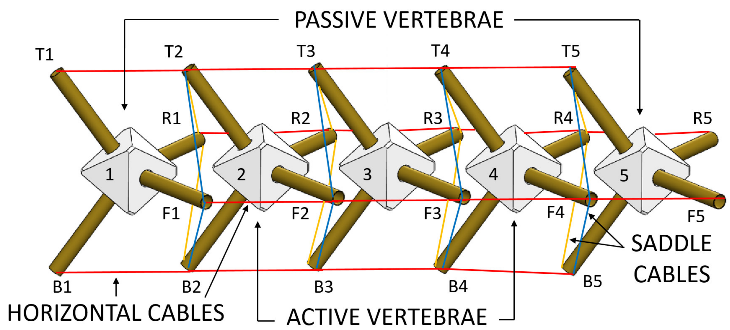


Figure 6.4: Spine Topology. Laika's spine consists of five tetrahedral vertebrae with two types of connecting cables (horizontal and saddle). The second and fourth vertebrae contain actuators for an even mass distribution.

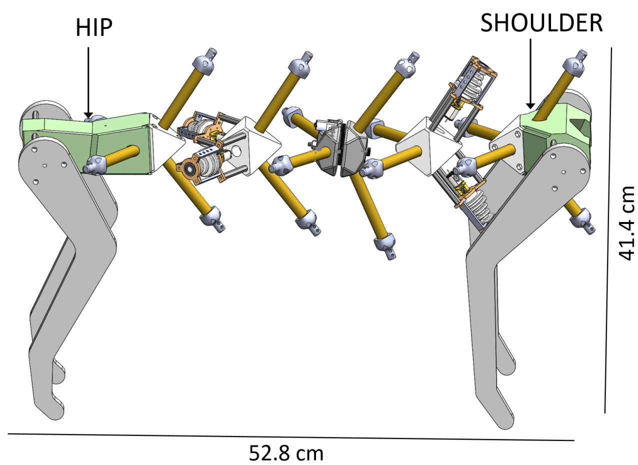


Figure 6.5: Assembly of a version of Laika with long legs. The spine is mounted onto the four legs attached to the hip and shoulder. This CAD render does not show the cables.

The robot's tension-network cables are implemented using a combination of two structural elements. First, the passive under-structure from Sec. 6.2. Then, a set of stiff cables, attached to mechanical springs, are used for actuation. These run alongside the lattice on the horizontal edges. The horizontal cables are actuated, whereas the saddle cables are only held in place by the elastomer lattice.

Two different materials are used for the lattice along different edges. The majority of the robot's lattice is a silicone rubber described above and seen alongside the motor spools in Fig. 6.7. An additional Buna-N rubber, with a higher stiffness, is used for the ventral horizontal cable. This stiffer strip of lattice counteracts the robot's weight.

There are thus three sets of properties for modeling the robot's cables. These are summarized in table 6.3. The elastomers are inexpensive materials for which no manufacturer data was available; thus, a set of tests were done to estimate a linear spring constant. The variability in these constants serves as a way to calibrate the simulations in Sec. 6.5.

Table 6.3: Hardware Prototype Material Properties.

Cable Material	Spring Constant (N/m)	Std Dev. (N/m)
Silicone Rubber	237	11
Buna-N Rubber	810	132
Mechanical Springs	187	– (exact)

All actuated components of this prototype use a brushed DC motor with a 1000:1 gearbox, are position-controlled using an encoder.

6.3.3 Vertebra and Actuator Design

Each vertebra has a 3D printed core which holds either the actuator assemblies (active vertebrae) or lattice and spring attachment points (on both passive and active.) For the passive vertebrae, the core has four rods with end caps that hold the attachment points. The active vertebrae have two rods with end caps, and two motor assemblies. These actuator assemblies have a bracket that connects to the core, and holds the motor and cable spool (Fig. 6.6).

As motivated by the inverse statics observations above, Laika's spine uses a 3D printed spool which has four different grooves, one for each cable in a given set of four horizontal cables (Fig. 6.7). The diameters of these spool grooves have a ratio of 1-1-2-3 (since the actuators are at vertebrae 2 and 4). When the spool rotates, each cable's length changes proportionally to the diameter of that groove, retracting the cables at different rates.

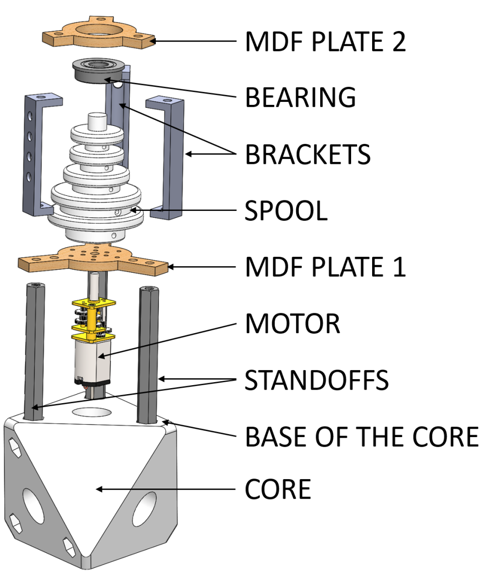


Figure 6.6: Actuator assembly. Each active vertebrae has two of these actuators that connect a motor to a spool in order to adjust the lengths of the horizontal cables.

6.3.4 Rotating Vertebra Design

An additional, different actuator is included as the middle vertebra of the spine, and provides its rotational degree-of-freedom (Fig. 6.8.) This vertebra is composed of two halves, one

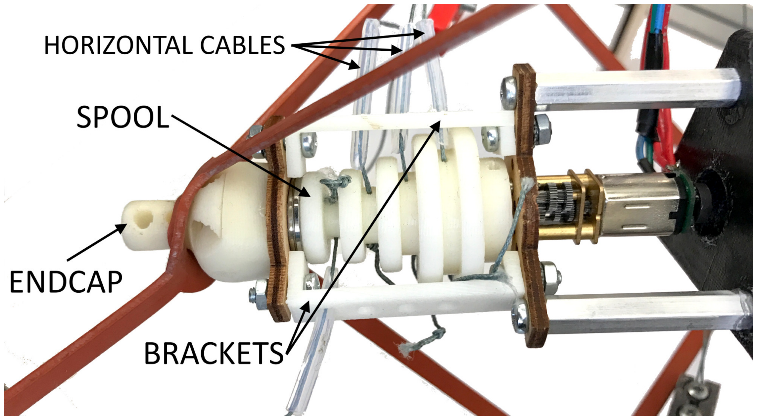


Figure 6.7: Spool prototype on an active vertebra. The spool is 3D printed such that the diameters of the gears match the varying horizontal cable rest lengths.

driving and one driven, which are connected through a shoulder screw that also acts as the shaft. The same motor as in Fig. 6.6 and 6.7 is mounted on the driving half and its torque is transmitted through a 4:1 spur gear pair to the driven half. This design was chosen for its structural and actuation simplicity, allowing for the straightforward generation of rotational motion.

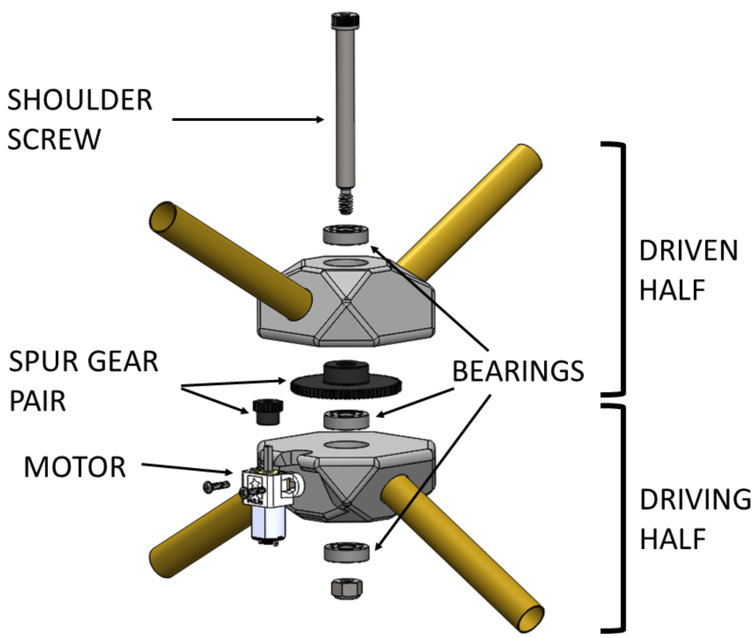


Figure 6.8: Rotating vertebra structure, used as vertebra 3 in the prototype. The moving halves of the structure allow for axial rotation of the spine.

6.4 Test Setups in Simulation and Hardware

In order to use Laika's spine to balance and lift its feet, in anticipation of developing a walking gait, both simulations and hardware tests were performed that demonstrated and correlated the spine's behavior with the robot's feet. Though both simulation and hardware have the capability to bend the robot in the sagittal plane, pulling the robot's dorsal (top) and ventral (bottom) cables, only coronal (left/right) bending is shown in this proof-of-concept. Shifts in the spine's center-of-mass in the coronal plane are what differentiate which foot is lifted.

The simulation, using numerical integration of rigid-body mechanics, can predict foot position as a function of spine motions. Although there is some intuition about which motions of the spine would lift which foot (for example, how the center of mass shifts as the robot bends), these simulations were used to quantify exactly how much movement would be required for different foot positions. These tests were performed using the NASA Tensegrity Robotics Toolkit[1].

Hardware tests were performed to show Laika's spine lifting its feet and to calibrate the simulation for future use. Multiple tests were performed per foot, all using the same lattice under the same conditions.

6.4.1 Simulation Setup

The NASA Tensegrity Robotics Toolkit (NTRT, or NTRTsim) is an open source package for modeling, simulation, and control of tensegrity robots based on the Bullet Physics engine [34]. Prior work has validated both the kinematics [38] and dynamics [131] of the simulator, and it has been extensively used in prior tensegrity robotics work [68, 70, 114, 105, 48, 169, 130], although minor technical details (due to friction, etc.) are present. Mirletz et al. (2015) describes the simulation parameters and models used for the core tenesgrity library in NTRT [131].

Cables tensions are modeled in NTRT as virtual spring-dampers, as in:

$$F_i = \begin{cases} k(\ell_i - u_i) - c\dot{\ell}_i, & \text{if} \quad k(\ell_i - u_i) - c\dot{\ell}_i \geq 0 \\ 0, & \text{if} \quad k(\ell_i - u_i) - c\dot{\ell}_i < 0, \end{cases} \tag{6.1}$$

where F_i is the tension force applied by cable i when its total length is ℓ_i, and the spring's rest length is u_i. The simulations controlled the rest lengths u_i, as if motors retracted the cables. This model corrects for cables that may go slack.

Like the hardware, the robot model consists of a spine with its rotating vertebra plus shoulders, hips, and legs (Fig. 6.9). The spine is rendered as a simplified model consisting of cylindrical rods. This is similar to the structure used in [169, 130]. The rotating vertebra is constructed out of two constrained rigid bodies, placed inside the spine, and is position-controlled by specifying a rotation between the two halves.

[1] http://irg.arc.nasa.gov/tensegrity/ntrt

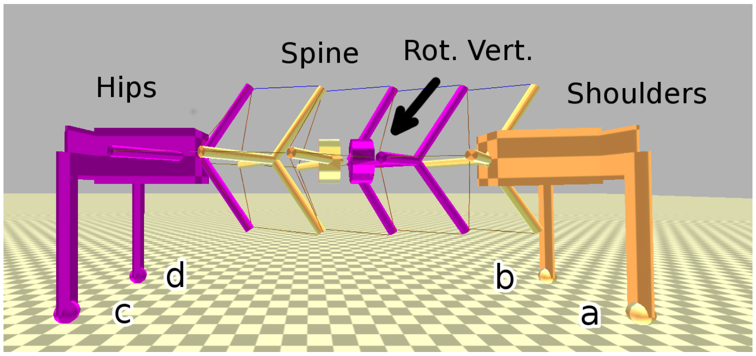


Figure 6.9: Model of Laika and its spine in the NASA Tensegrity Robotics Toolkit simulation, with the rotating vertebra. Feet labeling are: (a) front right, (b) front left, (c) back right, (d) back left.

6.4.2 Simulation Test Procedure

The initial simulations presented in this work represent a small subset of the robot's motions that may lift a foot, chosen in order to make a hardware comparison tractable. Specifically, bending in the coronal plane is restricted to two simple motions (left, or right), leaving the rotational degree-of-freedom as the primary variable.

Thus, simulation tests followed a two-step procedure. First, one set of horizontal cables were retracted to a percent P of their original rest length, as in:

$$u_i(t) = P\ u_i(0), \qquad t > 0 \tag{6.2}$$

in order to create a set left-or-right bend. Using observations from a first round of simulations, and the prototype itself, a horizontal length-change of $P = 80\%$ best represented a bending motion that sufficiently shifted the spine's center-of-mass.

Then, once the robot settled, the center vertebra was slowly rotated until one foot left the ground. Positions of the feet were recorded alongside the angle of rotation of the center vertebra. The center vertebra rotations for quasi-static motion were commanded as a slow ramp-input over 40 sec. up to 60 deg. in either direction, as in:

$$\theta(t) = \pm \left(\frac{t}{40}\right)\left(\frac{\pi}{3}\right) \tag{6.3}$$

Multiple tests were performed in order to calibrate the simulation parameters against hardware, as is common for this simulator [38]. Since one of the major assumptions in the simulation is the linearity of the cable spring force, the variation in the spring constant for the robot's cables (Table 6.3) provides a convenient method of calibration.

Five spring constants were tested with each of the four foot-lifting motions, varying both the silicone and Buna-N cable constants simultaneously. This adjusts the spine's overall tension, calibrating for both the unknowns and nonlinearities in the materials as well as modeling simplifications. Table 6.4 shows the spring constants chosen, evenly spaced from -2 to +2 standard deviations of the mean from Table 6.3.

Table 6.4: Five spring constant test points (in N/m) for the simulation, adjusting the overall tension of the robot.

Material	Low (-2σ)	Med-Low	Mean (μ)	Med-High	High ($+2\sigma$)
Silicone	216	227	237	248	258
Buna-N	547	678	810	941	1073

6.4.3 Hardware Test Setup and Procedure

The prototype of Laika was set up in placed of a camera, with off-board power and control (Fig. 6.10). The robot's control system consisted of microcontroller connected to a power supply, with two connected motors: one for a horizontal cable set, and one for the center vertebra. Power cables were wound through the spine to connect the motors to the controller. During testing, the motors' encoders were used to track rotations, which were converted to percent-length-change in the controller.

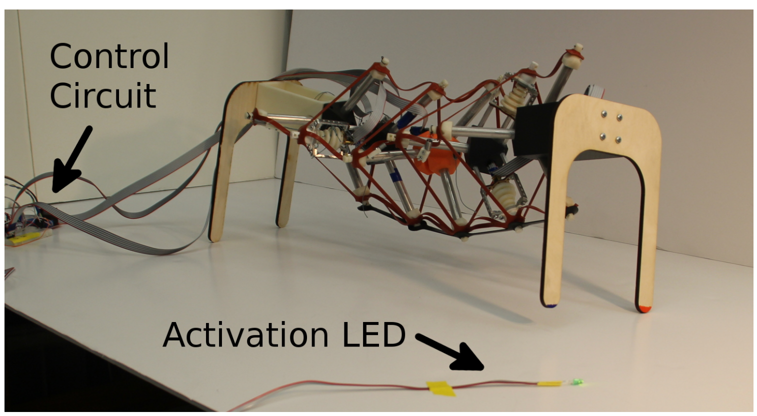


Figure 6.10: One test of Laika's spine. An offboard control circuit powered the motors and tracked rotations, and an LED was activated at the start of the test such that the video camera (out-of-frame) could correlate timestamps.

For each test, the spine was actuated along the same trajectories as in sec. 6.4.2 for the simulation. Switching between tests involved rotating the robot, re-routing the motors' cables, and if needed, attaching and detaching the required horizontal cables. The robot was rotated by 180 degrees when switching between the tests for its anterior and posterior ends. This required re-routing the motors' cables, causing a slight change in center-of-mass.

Prior to each test, the single set of actuated horizontal cables were tensioned until just past slackness, as an approximation to the simulation's initial conditions. Markers were

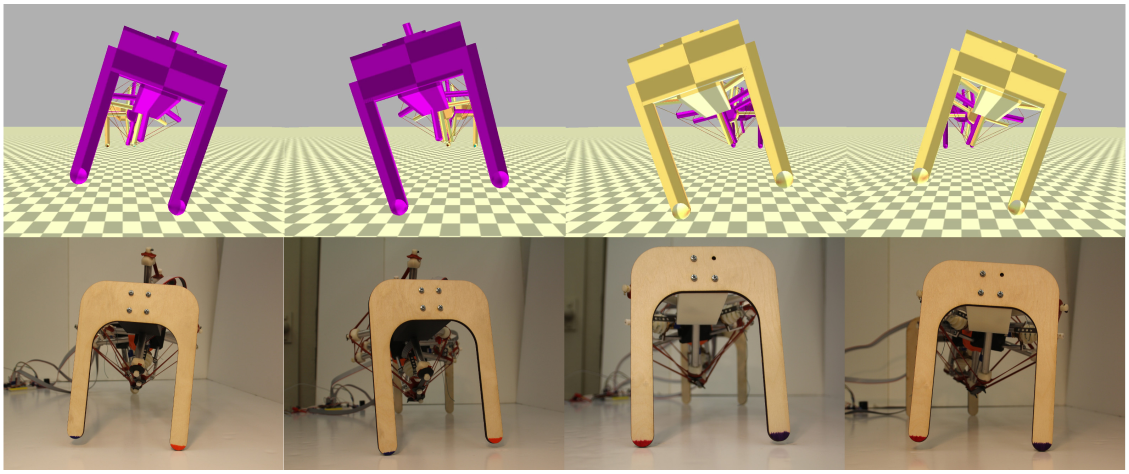


Figure 6.11: Lifting of each of Laika's feet in simulation (top) and hardware (bottom). Images left-to-right are for feet A, B, C, D (Front Right, Front Left, Back Right, Back Left). Hardware images taken just as liftoff occurs.

placed on the bed of the test setup, and the robot was re-positioned to the same state between tests.

For each test, the video camera recorded the feet as the controller tracked the rotations. A small LED, placed within the viewing frame of the camera, was activated at the start of the test, and a time series of data was recorded from the microcontroller. After each test, the video was analyzed for the time at which the LED activated, and at which the desired foot just began to lift. This time difference was then indexed into the controller data to find the rotation at that time.

6.5 Results: Laika's Spine Lifts Its Feet

By choosing four combinations of rotation direction and coronal-plane bending, Laika's spine was able to lift each of its four feet. Each of the four motions are summarized in Table 6.5, and shown in Fig. 6.11. These motions can be interpreted as a rotation lifting one diagonal set of legs, and bending shifting the robot's mass to raise one foot or the other.

Table 6.5: Motion combinations of spine for foot lifting.

Bend Dir. / Cabled Pulled	Vert. Rotation Dir.	Foot Lifted?
Left Bend / Horiz. Right	(+), CCW	A, Front Right
Right Bend / Horiz. Left	(+), CCW	C, Back Left
Left Bend / Horiz. Right	(-), CW	B, Front Left
Right Bend / Horiz. Left	(-), CW	D, Back Right

6.5.1 Foot Position and Required Rotation

The results of five hardware tests per foot are plotted against the simulation results in Fig. 6.12. The center-vertebra rotations at foot lift-off, observed in hardware, are plotted as black vertical lines representing the minimum and maximum data points. Colored curves are simulation data at the different lattice tension levels from Table 6.4.

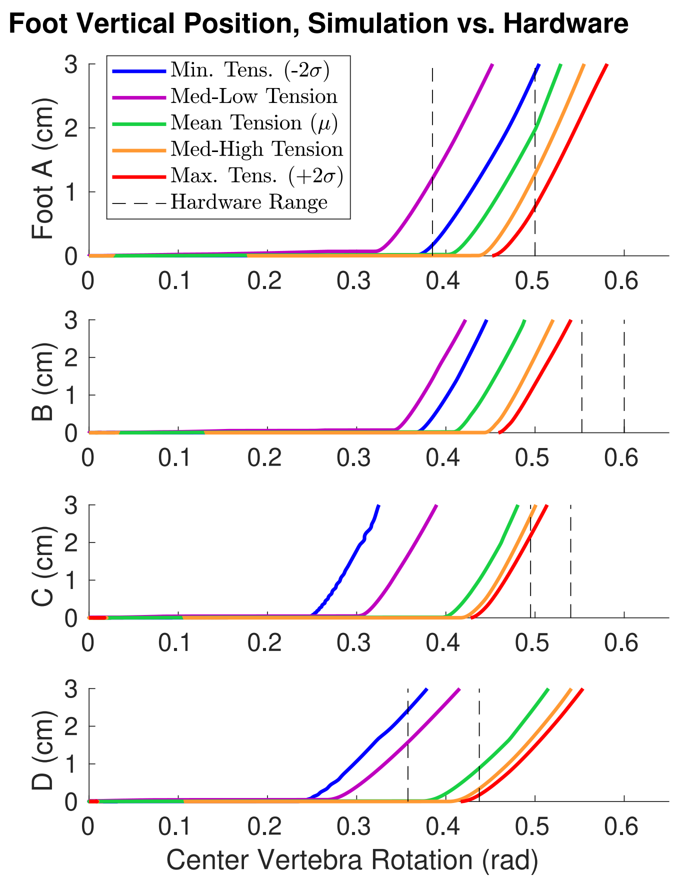


Figure 6.12: Simulations and hardware results of foot-lifting tests for Laika's spine. Black dashed lines represent the range of lift-off points in hardware. Colored curves represent vertical position of each foot (A, B, C, D) at varying levels of lattice tension in simulation. The highest-tension simulation result (red) matches hardware most closely, and represents a calibration of the simulation for future work.

The rotations for foot lift-off are summarized in Table 6.6. The hardware minimum and maximum correspond to the black dashed lines in Fig. 6.12. The simulation data listed are the points where each curve leaves the y-axis in Fig. 6.12, with the variation arising from lattice tension.

Across all tests, the maximum-tension lattice (red lines in Fig. 6.12) was most representative of hardware, falling within the range for feet A and D and closest to the range for feet

Table 6.6: Range of center vertebra rotations that produced foot lift-off, in simulation and hardware. All angles in radians (abs. val.)

Foot	Simul., Min	Simul., Max	HW, Min	HW, Max
A	0.33	0.47	0.44	0.50
B	0.35	0.47	0.57	0.60
C	0.25	0.44	0.51	0.54
D	0.25	0.43	0.41	0.43

B and C. The simulation results for feet B and C fall close to the hardware range, but not within. Such results can be expected with the small amount of testing that was performed in hardware, and with the variation in the test setup.

6.5.2 Discussion

Simulation data produced a calibration (highest lattice tension) that can reasonably be used for future work in developing balancing motions and gait cycles of the robot. The error in the B/C foot simulations, which did not strictly lie within the hardware range, can be attributable to the simplifications made in the simulation model. These simplifications include a combination of frictional effects at the robot's feet, the variation due to manual tensioning of the hardware robot's cables, and the simplified geometry in the simulation.

In addition to the differences observed between simulation and hardware for feet B and C, there were also differences between each foot with respect to lift-off angle as well as height after lift-off. These differences are expected, due to the spine's geometry.

Feet C and D, the back feet, lifted with less rotation than their front-opposite counterparts (C versus B, and D versus A.) This anterior-posterior difference is attributed to Laika's asymmetry in that direction, with more weight (due to the spine) at the robot's shoulders.

Feet B and C, which lifted with clockwise rotation, raised more rapidly after the initial lift-off. Such a difference is expected due to the geometry of the robot's saddle cables. These cables do not lie completely in the transverse plane of the spine: they pull the vertebrae forward and backward as well (Fig. 6.4.) Thus, when the center vertebra rotates, it also creates a small amount of additional bending in both the horizontal (coronal) and front-back vertical (sagittal) plane, as its saddle cables adjust. The clockwise/counterclockwise difference is attributed to center-of-mass shifts in the transverse (left-right vertical) plane from this extra bending.

6.6 Ongoing Work

An actuated tensegrity spine, as a body of a quadruped robot, can lift and balance a robot's feet. Ongoing work seeks to demonstrate full locomotion of such a robot, using the results above as part of a gait cycle. This work is proceeding in two ways: first, incorporating moving legs into the robot, and second, reposition and redesigning the actuators for reliability.

Fig. 6.13 shows an early design of Laika's successor, named Belka, after the second dog in space. Belka incorporates motors at each of its shoulder joints to move each leg in a single degree of freedom. It is anticipated that, when these designs are completed, that the left-lifting motions can be combined with leg-swinging motions to produce a walking gait.

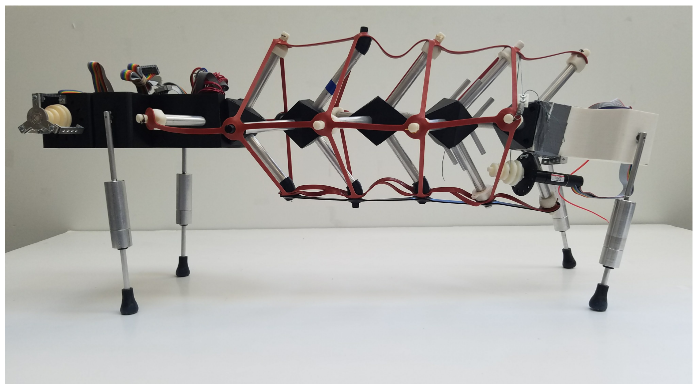
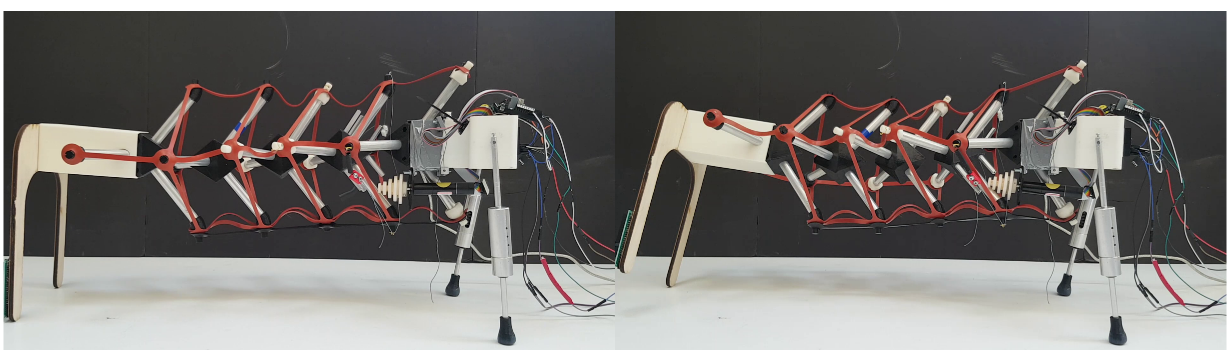


Figure 6.13: Prototype for Belka, the successor to Laika, with actuated legs. Credit to authors of [124], used with permission.

Additionally, Fig. 6.14 demonstrates a new prototype of a rotational actuator for Belka. Belka's designs alleviate some issues observed with the rotating vertebra on Laika, such as motor stall, by adjusting the rest lengths of saddle cables.



(a) Belka, with its spine in a neutral position. (b) Belka, with its spine rotated.

Figure 6.14: Belka, the successor to Laika, with a new rotational actuation mechanism that replaces Laika's rotating vertebra. The additional spool underneath Belka's shoulders adjusts saddle cables to rotate the spine. Credit to authors of [124] with permission.

There is much work left before walking locomotion can be implemented. For example, a gait cycle has yet to be investigated. However, the results with Laika present promising directions towards walking.

Part III

Dynamics and Control

Chapter 7

Model-Predictive Control of Tensegrity Spines

This dissertation seeks to demonstrate that tensegrity spines can be modeled, designed, manufactured, and controlled in ways such that their use in quadruped robots are practical. To that end, this part of the dissertation addresses closed-loop control of these structures. The following chapter presents two controllers for tensegrity spine robots, using model-predictive control (MPC), one of which also incorporates inverse statics optimization. Chapter 8 takes an alternative approach to addressing some of the fundamental analysis and control issues encountered with tensegrity spines, though does not yet provide a controller for the spines themselves.

This chapter presents two controllers for Laika's spine, both of which track state-space trajectories using the spine's dynamics model. These controllers use combinations of model-predictive control (MPC) with the inverse statics (IS) algorithm from Chap. 5. Both frameworks are motivated by the practical challenges with computational complexity of nonlinear, optimization-based control. The first controller employs a variety of smoothing and tuning terms in the MPC optimization problem. It was the first controller to demonstrate closed-loop tracking of a state space trajectory with a tensegrity spine.

The second controller incorporates an inverse statics optimization problem to generate reference input trajectories that are then used with MPC. The second approach is significantly more general, with less tuning than the smoothing approach (Table 7.1), and with more favorable computational characteristics since the inverse statics are solved offline. This approach contributes a new architecture for addressing computationally-complex state tracking problems in robots such as these spines. The controller's novelty arises from both this new solution to the inverse statics problem as well as its interconnection with MPC, forming a new approach to control of such systems (Sec. 7.3.3 and Fig. 7.3b).

Table 7.1: Controller formulations: MPC with smoothing vs. MPC with tracking of inverse statics (IS) input trajectories

Controller formulation	# Tuning constants	Time discr.	Simulation setup	Max. Error	Refs.
Smoothing terms	14	$1e^{-3}$ sec.	3 vertebra, 3D	< 0.5 cm	[166]
IS input traj. tracking	5	$1e^{-5}$ sec.	1 vertebra, 2D	See Sec. 7.4.4	-

The two controller formulations presented in this paper have different benefits with respect to tuning and performance. The five tuning constants (column two) of the more general controller with the inverse statics optimization algorithm are straightforward to chose. All have physical interpretations (e.g., minimum cable tension, vertebra anti-collision distance) or are common to many optimal control problems (e.g., the $\mathbf{Q}$ and $\mathbf{R}$ weighting matrices in eqns. (7.37)-(7.38), and MPC horizon length.)

7.1 Motivation

When quadruped robots are constructed with spine-like flexible bodies that include actuation, significant control challenges are often encountered [174]. Few dynamics-based closed-loop control approaches have been developed for robots like these. Instead, authors commonly use kinematics-only models [123, 90], model-free control using machine learning [202, 62, 92], decoupled controllers for different parts of the robot [174], or the replaying of open-loop inputs [126] among other approaches. This chapter presents one of the first closed-loop, model-based controllers for actuated spines for quadruped robots.

7.1.1 Control Challenges for Tensegrity Robots: Nonlinearities, Saturation, and Constraints

Control challenges for tensegrity robots are commonly due to dynamics which are inherently nonlinear and often high-dimensional. Various saturation issues also exist, as cables within the structure exert no force in compression, and the controller cannot retract the spine's flexible cables to a negative 'rest length' (defined in Sec. 7.2.2.) Consequently, state-space control for tracking or regulation has been mostly limited to low-dimensional structures, particularly those with only bars [6, 210, 178, 179, 128], which assume, a-priori, that all cables are initially tensioned. Open-loop methods have also been used for this purpose [186, 187, 37, 68], but cannot reject disturbances.

This chapter proposes controllers based on model-predictive control (MPC) for three primary reasons. First, using an optimization program for control can address constraints on the system (actuator saturation and tensioned cables). Second, computational tractability can be addressed by using a receding horizon. These two features define an MPC problem. Finally, an MPC formulation allows straightforward introduction of smoothing weights and constraints for hard-to-control systems [209, 63].

Model-predictive control for nonlinear systems (NMPC) is a well-studied topic with many implementations [7], particularly in low dimensions where nonconvex optimization is computationally feasible [209]. For high-dimensional nonlinear systems, practical options include modifying the NMPC problem or using more efficient solvers [39]. Alternatively, linearized dynamics can be used at later points in the horizon [217], or linearizations can be performed at each timestep in the problem to create a linear time-varying MPC [63]. A time-varying linearization is used in this chapter for computational tractability purposes. As opposed to attempting to solve an NMPC problem in real time for this high-dimensional system, the proposed control architecture addresses linearization error via two other methods. In particular, one approach here includes smoothing terms, while the other tracks an approximated input reference trajectory generated by inverse statics.

7.1.2 Controller Verification via Simulations

This chapter employs a set of simulations that demonstrate the performance of the proposed controllers and show proof-of-concept. There are fewer traditional sources of modeling error in this problem than in other tensegrity robotics control problems, which commonly involve locomotion on the ground [153, 215]. Locomotion requires modeling complex interactions with the ambient environment [24]. This spine instead moves freely in space without surface contact, therefore simulation inaccuracies due to contact friction modeling are not present. Here, the most significant sources of error are anticipated to arise from unmodeled actuator dynamics and manufacturing differences of hardware prototypes versus the nonlinear dynamics model. Prior work has confirmed that rigid body models of free-standing tensegrity robots match hardware results reasonably well under similar conditions [25, 38].

This chapter uses a three-dimensional model of the spine, with multiple vertebrae, for evaluating the smoothing controller (from [166].) Meanwhile, a reduced-order two-dimensional model, with only a single moving vertebra, is used for the controller with the IS optimization for input trajectory generation. Although the controller with the IS optimization is tested on a lower-dimensional system than the smoothing controller, the tracked states of the vertebrae are the same. Therefore, the results are compared quantitatively in Sec. 7.4, and the limitations of this comparison are discussed in Sec. 7.5.

7.2 Spine Model and Movement Goals

The geometry of the spine is the same as that considered in Chap. 5 and Chap. 6: the tetrahedral vertebra. The following section briefly describes the state-space model used for both the two-dimensional and three-dimensional spine, as well as the desired state trajectory to be tracked. Full knowledge of the system states at each timestep is assumed; the controllers in this chapter are state-feedback.

7.2.1 Vertebra Geometry and State Space

Each spine vertebra is a rigid body, approximated by a system of point masses (Fig. 7.1), as has been justified in past literature [38, 68]. The local frame of one vertebra contains point mass k at position $\mathbf{a}_k$, $k = 1 \ldots K$ (Fig. 7.1.) The local frames of each node from Fig. 7.1a and 7.1b are, in centimeters,

$$\begin{bmatrix}\mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3 & \mathbf{a}_4 & \mathbf{a}_5\end{bmatrix} = \begin{bmatrix} 0 & 13 & -13 & 0 & 0 \\ 0 & 0 & 0 & 13 & -13 \\ 0 & -7.5 & -7.5 & 7.5 & 7.5 \end{bmatrix} \tag{7.1}$$

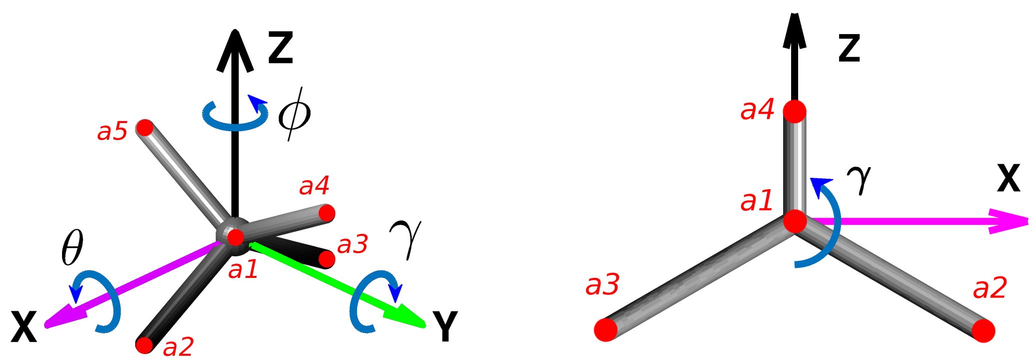
$$\begin{bmatrix}\mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3 & \mathbf{a}_4\end{bmatrix} = \begin{bmatrix} 0 & 13 & -13 & 0 \\ 0 & -7.5 & -7.5 & 7.5 \end{bmatrix}. \tag{7.2}$$

The mass at each node is assigned to evenly distribute the $m = 0.13$ kg mass of each vertebra.

The robot's state space is parameterized by the coordinates of the center of mass and a set of Euler angles (3-2-1) for each vertebra, in addition to their respective time-derivatives. The continuous-time equations of motion have the form

$$\dot{\boldsymbol{\xi}} = \mathbf{g}(\boldsymbol{\xi}, \mathbf{u}), \tag{7.3}$$

where $\boldsymbol{\xi} \in \mathbb{R}^{36}$ in 3D (for three moving vertebrae) or $\mathbb{R}^6$ in 2D (for one moving vertebra) is the state vector, and $\mathbf{u} \in \mathbb{R}^{24}$ in 3D or $\mathbb{R}^4$ in 2D is the input vector, which has the same dimension as number of cables.



(a) A single 3D spine vertebra in its local coordinate system.

(b) A single 2D spine vertebra in its local coordinate system.

Figure 7.1: Geometry of spine vertebrae in both 3D and 2D. Point mass locations $\{\mathbf{a}_1 ... \mathbf{a}_4, \mathbf{a}_5\}$ shown in red. Certain coordinates (X-axis and θ, γ rotations) are flipped from the right-hand convention in order to match a simulation environment used in prior research [68, 169, 168].

The system state $\boldsymbol{\xi}$ parameterizes the position of each point mass within a vertebra. For a local frame of particle positions $\mathbf{a}_k$ in vertebra j, the particle's position in the global frame is $\mathbf{b}_{kj}$ as in

$$\mathbf{b}_{kj}(\boldsymbol{\xi}) = \mathbf{R}_j^{\phi}(\boldsymbol{\xi})\mathbf{R}_j^{\gamma}(\boldsymbol{\xi})\mathbf{R}_j^{\theta}(\boldsymbol{\xi})\mathbf{a}_k + \mathbf{r}_j(\boldsymbol{\xi}),$$

with the vertebra's center of mass $\mathbf{r}_j$ and rotation matrices $\mathbf{R}_j$ a function of the generalized coordinates. The 2D model removes the y, θ, ϕ coordinates, but is otherwise expressed in the same manner.

The continuous-time function $\mathbf{g}(\boldsymbol{\xi}, \mathbf{u})$ can be symbolically solved by considering $\mathbf{b}_{kj}$ as a system of particles. These models have J vertebrae and K point masses per vertebra. Lagrange's equations were used to express the dynamics of the system. With the particles' total kinetic energy T, gravitational potential energy U, and Lagrangian $L = T - U$,

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\xi}_i} - \frac{\partial L}{\partial \xi_i} = \sum_{j=1}^{J}\sum_{k=1}^{K} \mathbf{F}_{kj} \cdot \frac{\partial \mathbf{b}_{kj}}{\partial \xi_i}, \quad i = 1...6J, \tag{7.4}$$

The right-hand side and left-hand side of (7.4) are solved symbolically, then equated to solve for $\mathbf{g}(\boldsymbol{\xi}, \mathbf{u})$, noting that $\dot{\xi}_i = \xi_{i+6J}$.

7.2.2 Cable Model as System Inputs

The cables suspending the vertebrae provide the control input to the system. Unlike work such as [210, 178, 6], it is not assumed that the controller specifies forces in the robot's cables, since this becomes challenging to implement on physical hardware. Instead, the control inputs are the rest lengths of a virtual spring-damper. Specifically, let the vector between the two connection points of cable i be $\boldsymbol{\ell}_i$, with scalar length $\ell_i = ||\boldsymbol{\ell}_i||$. Then the applied force due to a cable, directed away from its attachment point, is

$$\mathbf{F}_i = F_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i,$$

so that tension forces are positive. Here $\hat{\boldsymbol{\ell}}_i$ is a normalized direction vector. The scalar tension force on cable i is a rectified spring-damper, so the cables apply no compression forces:

$$F_i = \begin{cases} k(\ell_i - u_i) - c\dot{\ell}_i, & \text{if} \quad k(\ell_i - u_i) - c\dot{\ell}_i \geq 0 \\ 0, & \text{if} \quad k(\ell_i - u_i) - c\dot{\ell}_i < 0, \end{cases} \tag{7.5}$$

where the input u_i is the rest length of cable i. In addition, the controller cannot command a negative u_i, since retracting a cable to a negative length is not physically possible.

7.2.3 Reference State Trajectory

The desired trajectory $\bar{\boldsymbol{\xi}}$ for the spine robot is a bending motion in the X-Z plane, consisting of translations and rotations for each moving vertebra (Fig. 7.2.) As no a-priori dynamic trajectories were available for this model, the controllers in Sec. 7.3 do not include the tracking of vertebrae velocities. Consequently, this trajectory is not guaranteed to be dynamically feasible. However, this sequence of states has been observed as the output of prior qualitative simulation studies in Chap. 6, and is therefore judged as a reasonable control goal.

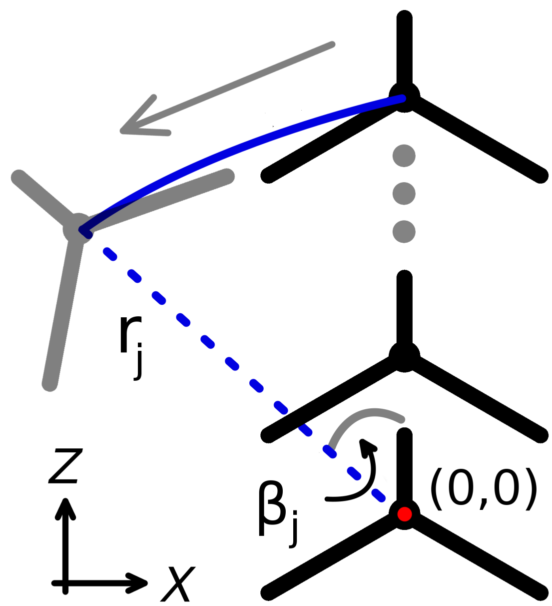


Figure 7.2: Bending trajectory for the j-th vertebra of the spine in the X-Z plane. The vertebra rotates counterclockwise around the origin at a constant radius r_j (dashed blue line), swept out by angle β_j (solid gray line). Solid blue line shows the center of mass of the vertebra.

The desired state trajectory $\bar{\boldsymbol{\xi}}$ separates the vertebrae by 10cm vertically in their starting positions. For vertebra $j = 1 \dots J$,

$$\bar{z}_j(0) = 0.1j \; . \tag{7.6}$$

These initial heights also define the radius of the rotation: $r_j = \bar{z}_j(0)$. Consequently, the reference positions of each vertebra over time, $\bar{x}_j(t)$ and $\bar{z}_j(t)$, are:

$$\bar{x}_j(t) = r_j \sin(\beta_j(t)), \quad \bar{z}_j(t) = r_j \cos(\beta_j(t)). \tag{7.7}$$

In addition, the desired rotation $\bar{\gamma}_j(t)$ of each vertebra about its inertial Y-axis is defined to be the same as the sweep angle $\beta_j(t)$ for that vertebra,

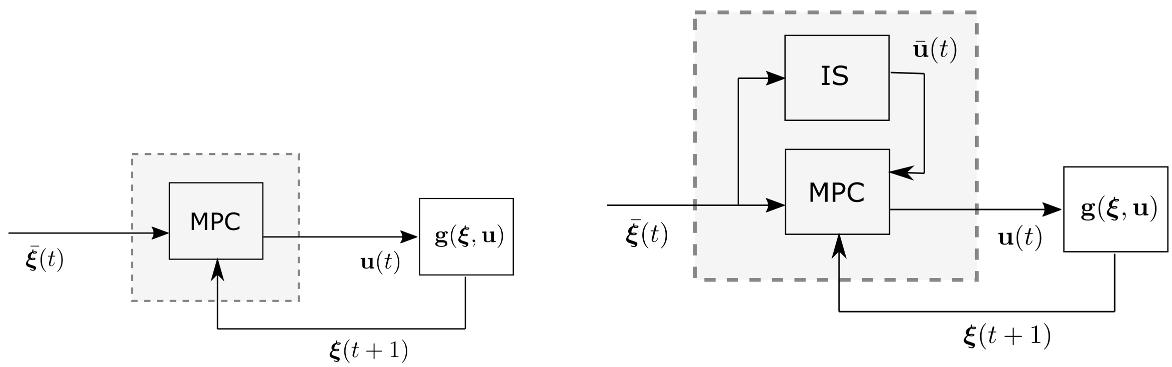
$$\bar{\gamma}_j(t) = \beta_j(t). \tag{7.8}$$

The maximum sweep angles for each vertebra are the following. The 2D model with one vertebra only uses β_1^{max}.

$$[\beta_1^{max}, \quad \beta_2^{max}, \quad \beta_3^{max}] = \left[\frac{\pi}{16}, \quad \frac{\pi}{12}, \quad \frac{\pi}{8}\right]. \tag{7.9}$$

7.3 Controller Formulations

This chapter introduces two controllers for the tensegrity spine robots under consideration. The first uses a model-predictive control (MPC) law, and incorporates smoothing terms into the optimization problem (Fig. 7.3a.) The second uses the inverse statics routine, Algorithm (1) from Chap. 5, for reference input trajectory generation, and a simplified version of the model-predictive control law to close the control loop (Fig. 7.3b.) Both controllers incorporate a linearization of the equations of motion, eqn. (7.3), in the control calculations; however, all are simulated against the ground-truth nonlinear system.



(a) Block diagram for the model-predictive controller with smoothing terms, no input trajectory generation.

(b) Block diagram of the proposed controller that combines inverse statics (IS) for input trajectory generation with model-predictive control (MPC) to close the loop.

Figure 7.3: Block diagrams of the two controllers considered in this chapter. Both controllers are simulated against the ground-truth nonlinear dynamics $\mathbf{g}(\boldsymbol{\xi}, \mathbf{u})$.

These controllers prioritize practicality over theoretical guarantees. For this reason, neither formulation contains terminal constraints, and thus stability can only be shown experimentally, not proven.

The following sections use subscripts (e.g., $\mathbf{u}_t$) to represent predicted values of vectors at a time instance, and parentheses (e.g., $\mathbf{u}(t)$) to represent a measured or applied value at that time instance. Note that these are the same for the reference trajectory ($\bar{\mathbf{u}}(t) = \bar{\mathbf{u}}_t$) and so the notation is used interchangeably. Superscripts (e.g., $\boldsymbol{\xi}^{(i)}$) index into a vector.

7.3.1 Model-Predictive Controller Formulation

For both controllers, the MPC block generates a control input $\mathbf{u}(t)$ via the following. At each timestep t, a constrained finite-time optimal control problem (CFTOC) is solved, generating the sequence of optimal control inputs $\mathbf{U}^*_{t\rightarrow t+N|t} = \{\mathbf{u}^*_{t|t}, ..., \mathbf{u}^*_{t+N|t}\}$, over a horizon of N timesteps ahead. The notation $t+k|t$ represents a value at the timestep $t+k$, as predicted at timestep t (from [28], Ch. 4.) The first input $\mathbf{u}^*_{t|t}$ is applied, as in $\mathbf{u}(t) = \mathbf{u}^*_{t|t}$, closing the loop. The following sections define this CFTOC problem for each case, fully specifying the controllers.

7.3.2 Controller with MPC and Smoothing Terms

The first controller adapts the standard linear time-varying MPC formulation by adding a variety of hand-tuned weights and constraints. This is a common approach to establishing proof-of-concept control [63].

Constrained Finite-Time Optimal Control Problem Formulation

The following CFTOC problem is solved at each timestep t using a quadratic programming solver.

$$
\begin{aligned}
\min_{\mathbf{U}_{t\rightarrow t+N|t}} \quad & p(\boldsymbol{\xi}_{t+N|t}, \Delta\boldsymbol{\xi}_{t+N|t}) + \sum_{k=0}^{N-1} q(\boldsymbol{\xi}_{t+k|t}, \Delta\boldsymbol{\xi}_{t+k|t}, \Delta\mathbf{u}_{t+k|t}) && (7.10)\\
\text{s.t.} \quad & \boldsymbol{\xi}_{t+k+1|t} = \mathbf{A}_t\boldsymbol{\xi}_{t+k|t} + \mathbf{B}_t\mathbf{u}_{t+k|t} + \mathbf{c}_t && (7.11)\\
& \Delta\boldsymbol{\xi}_{t+k|t} = \boldsymbol{\xi}_{t+k|t} - \boldsymbol{\xi}_{t+k-1|t} && (7.12)\\
& \Delta\mathbf{u}_{t+k|t} = \mathbf{u}_{t+k|t} - \mathbf{u}_{t+k-1|t} && (7.13)\\
& \boldsymbol{\xi}_{t|t} = \boldsymbol{\xi}(t) && (7.14)\\
& \mathbf{u}^{min} \leq \mathbf{u}_{t+k} \leq \mathbf{u}^{max} && (7.15)\\
& ||\mathbf{u}_{t|t} - \mathbf{u}_{t-1}||_\infty \leq w_1 && (7.16)\\
& ||\mathbf{u}_{t+k|t} - \mathbf{u}_{t|t}||_\infty \leq w_2, \; k = 1..(N-1) && (7.17)\\
& ||\mathbf{u}_{t+N|t} - \mathbf{u}_{t|t}||_\infty \leq w_3 && (7.18)\\
& ||\Delta\boldsymbol{\xi}^{(1:6)}_{t+k|t}||_\infty \leq w_4 && (7.19)\\
& ||\Delta\boldsymbol{\xi}^{(13:18)}_{t+k|t}||_\infty \leq w_5 && (7.20)\\
& ||\Delta\boldsymbol{\xi}^{(25:30)}_{t+k|t}||_\infty \leq w_6 && (7.21)\\
& \boldsymbol{\xi}^{(3)}_{t+k|t} + w_7 \leq \boldsymbol{\xi}^{(15)}_{t+k|t} && (7.22)\\
& \boldsymbol{\xi}^{(15)}_{t+k|t} + w_7 \leq \boldsymbol{\xi}^{(27)}_{t+k|t}. && (7.23)
\end{aligned}
$$

Here, $N = 10$ is the horizon length and $w_1 ... w_7$ are constant scalar weights. The functions p and q represent the terminal cost and stage cost of the objective function, not to be confused with the inverse statics force balance of Chap. 5. The objective function, and the use and purpose of the constraints, are given in subsections 7.3.2 to 7.3.2.

Dynamics Constraint

The dynamics constraint (7.11) consists of a time-varying linearization of the system, as in:

$$\mathbf{A}_t = \frac{\partial \mathbf{g}(\boldsymbol{\xi}, \mathbf{u})}{\partial \boldsymbol{\xi}}\Big|_{\substack{\boldsymbol{\xi}=\boldsymbol{\xi}_{t-1} \\ \mathbf{u}=\mathbf{u}_{t-1}}} \tag{7.24}$$

$$\mathbf{B}_t = \frac{\partial \mathbf{g}(\boldsymbol{\xi}, \mathbf{u})}{\partial \mathbf{u}}\Big|_{\substack{\boldsymbol{\xi}=\boldsymbol{\xi}_{t-1} \\ \mathbf{u}=\mathbf{u}_{t-1}}} \tag{7.25}$$

$$\mathbf{c}_t = \mathbf{g}(\boldsymbol{\xi}_{t-1}, \mathbf{u}_{t-1}) - \mathbf{A}_t \boldsymbol{\xi}_{t-1} - \mathbf{B}_t \mathbf{u}_{t-1}. \tag{7.26}$$

This linearization (7.24-7.26) is implemented as a finite difference approximation. This approach is chosen due to computational issues with calculating additional analytical derivatives of the dynamics. The calculated $\mathbf{A}_t, \mathbf{B}_t, \mathbf{c}_t$ are used over the entire horizon. For the start of the simulation, $\mathbf{u}_0 = \mathbf{0}$ is used. Since these linearizations are not at equilibrium points, the linear system is affine, with $\mathbf{c}_t$ being a constant vector offset.

Here, the continuous-time linearized dynamics are used as a constraint, and are not discretized. Since the timesteps in the simulations below are small ($dt = 0.001$ sec.), a discretization does not significantly alter the values of $\mathbf{A}_t, \mathbf{B}_t, \mathbf{c}_t$.

Other Constraints

The remaining constraints are either smoothing terms, constraints motivated by the physical robot, or miscellaneous housekeeping terms.

Constraints (7.12) and (7.13) define the $\Delta \mathbf{u}$ and $\Delta \boldsymbol{\xi}$ variables, which are used for the smoothing constraints on the inputs and states. Constraint (7.14) assigns the state variable at the start of the optimization horizon, $\boldsymbol{\xi}_{t|t}$, to the actual observed value of the state from the previous simulation timestep, $\boldsymbol{\xi}(t)$.

Constraint (7.15) is a bound on the inputs, limiting the length of the cable rest lengths, with $\mathbf{u}^{min}, \mathbf{u}^{max} \in \mathbb{R}^{24}$ but having the same value for all inputs (Table 7.2).

Constraints (7.16-7.21) are smoothing terms to compensate for the lack of an reference input trajectory. Of these, (7.16-7.18) are for the inputs, where $\mathbf{u}_{t-1}$ is the most recent input at the start of the CFTOC problem. Constraints (7.19-7.21) are smoothing terms on the states, limiting the deviation between successive states in the trajectory. These reduce linearization error, and are split so that the positions and angles of each vertebra could be weighted differently. No velocity terms are constrained.

Finally, since states $\{\boldsymbol{\xi}^{(3)}, \boldsymbol{\xi}^{(15)}, \boldsymbol{\xi}^{(27)}\}$ are the vertebra z-positions, constraints (7.22-7.23) prevent vertebra collisions.

Objective Function

The objective function has a terminal cost p and a stage cost q defined as the following. Here, shortened notation such as $\|\Delta\boldsymbol{\xi}_{t+k|t}\|^2_{\mathbf{S}^k}$ denotes a weighted quadratic term, as in $(\Delta\boldsymbol{\xi}_{t+k|t})^\top \mathbf{S}^k (\Delta\boldsymbol{\xi}_{t+k|t})$.

$$p(\boldsymbol{\xi}_{t+N|t}, \Delta\boldsymbol{\xi}_{t+N|t}) = \|\boldsymbol{\xi}_{t+N|t} - \bar{\boldsymbol{\xi}}_{t+N|t}\|^2_{\mathbf{Q}^N} + \|\Delta\boldsymbol{\xi}_{t+N|t}\|^2_{\mathbf{S}^N}, \tag{7.27}$$

$$q(\boldsymbol{\xi}_{t+k|t}, \Delta\boldsymbol{\xi}_{t+k|t}, \Delta\mathbf{u}_{t+k|t}) = \|\boldsymbol{\xi}_{t+k|t} - \bar{\boldsymbol{\xi}}_{t+k|t}\|^2_{\mathbf{Q}^k} + \|\Delta\boldsymbol{\xi}_{t+k|t}\|^2_{\mathbf{S}^k} + w_8\|\Delta\mathbf{u}_{t+k|t}\|_\infty. \tag{7.28}$$

As before, w_8 is a scalar, while $\mathbf{Q}$ and $\mathbf{S}$ are constant diagonal weighting matrices which are exponentiated by the timestep in the optimization horizon. Here, $\mathbf{Q}$ penalizes the tracking error in the states, $\mathbf{S}$ penalizes the deviation in the states at one timestep to the next, and w_8 penalizes the deviations in the inputs from one timestep to the next. These matrices are diagonal, with blocks corresponding to the Cartesian and Euler angle coordinates, with zeros for all velocity states, according to vertebra.

Raising each diagonal element of $\mathbf{Q}$ or $\mathbf{S}$ to the power of k or N puts a heavier penalty on terms farther away on the horizon. These are defined as:

$$\begin{aligned}
&\bar{\mathbf{Q}}^k = diag(w_9^k,\ w_9^k,\ w_9^k \mid w_{10}^k,\ w_{10}^k,\ w_{10}^k \mid 0...0) \in \mathbb{R}^{12\times 12} \\
&\bar{\mathbf{S}}^k = diag(w_{11}^k,\ w_{11}^k,\ w_{11}^k \mid w_{11}^k,\ w_{11}^k,\ w_{11}^k \mid 0...0) \in \mathbb{R}^{12\times 12} \\
&\mathbf{Q}^k = \mathbf{I}_3 \otimes \bar{\mathbf{Q}}^k, \qquad \mathbf{S}^k = \mathbf{I}_3 \otimes \bar{\mathbf{S}}^k.
\end{aligned} \tag{7.29}$$

Table 7.2 lists all the constants for this controller, including the constraints and the objective function, with units.

7.3.3 Controller with MPC and Inverse Statics Optimization

A major contribution of this chapter is a controller that combines the inverse statics (IS) optimization, via Algorithm (1), with an MPC block. As shown in Fig. 7.3b, the IS block generates a reference input trajectory $\bar{\mathbf{u}}$ that is tracked alongside $\bar{\boldsymbol{\xi}}$ as part of the MPC problem. This approach contributes a new method to address computational complexity and tuning. The IS solutions can be solved offline, reducing the load on the MPC optimization problem. This approach also significantly reduces hand-tuning. As discussed in this section, the controller is formulated for the 2D, single-vertebra spine model.

Table 7.2: Smoothing controller weights and constants.

Constant:	Value:		Interpretation:
N	10	no units	Horizon Length
u_{min}	0.0	meters (cable)	Min. Cable Length
u_{max}	0.20	meters (cable)	Max. Cable Length
w_1	0.01	meters (cable)	Input Smooth., Horiz. Start
w_2	0.01	meters (cable)	Input Smooth., Horiz. Middle
w_3	0.10	meters (cable)	Input Smooth., Horiz. End
w_4	0.02	meters and radians	State Smooth., Bottom Vert.
w_5	0.03	meters and radians	State Smooth., Mid. Vert.
w_6	0.04	meters and radians	State Smooth., Top Vertebra
w_7	0.02	meters	Vertebra Anti-Collision
w_8	1	no units	Input Smoothing
w_9	25	no units	State Tracking, Vertebra Pos.
w_{10}	30	no units	State Tracking, Vert. Angle
w_{11}	3	no units	Input Difference Penalty

Constrained Finite-Time Optimal Control Problem Formulation

The following CFTOC problem is solved at each timestep t using a quadratic programming solver.

$$\min_{\mathbf{U}_{t\to t+N|t}} \quad p(\boldsymbol{\xi}_{t+N|t}) + \sum_{k=0}^{N-1} q(\boldsymbol{\xi}_{t+k|t}, \mathbf{u}_{t+k|t}) \tag{7.30}$$

$$\text{s.t.} \quad \boldsymbol{\xi}_{t+k+1|t} = \mathbf{A}_t \boldsymbol{\xi}_{t+k|t} + \mathbf{B}_t \mathbf{u}_{t+k|t} + \mathbf{c}_t \tag{7.31}$$

$$\boldsymbol{\xi}_{t|t} = \boldsymbol{\xi}(t) \tag{7.32}$$

$$\mathbf{u}_{t+k|t} \geq \mathbf{u}^{min} \tag{7.33}$$

$$\boldsymbol{\xi}^{(2)}_{t+k|t} \geq w_1. \tag{7.34}$$

This formulation (7.30-7.34) is significantly simpler than the smoothing formulation (7.10-7.23), with only one scalar tuning weight w_1, and a much smaller horizon length (Table 7.3). As above, p and q represent the terminal cost and stage cost of the objective function. The following sections define the objective function, and use and purposes of the constraints.

Dynamics Constraint

Constraint (7.31) enforces the time-varying linearized system dynamics, just as with the smoothing controller, via eqns. (7.24-7.26). The two-dimensional controller also applies a zero-order hold to (7.31) for increased prediction fidelity. However, due to the small timesteps involved, the values of $\mathbf{A}_t$, $\mathbf{B}_t$, and $\mathbf{c}_t$ remained mostly unchanged after this operation, with no noticeable effect on simulation results.

Other Constraints

The remaining constraints have the same interpretations as their counterparts in the smoothing controller formulation. Constraint (7.32) assigns the initial condition at the starting time of the CFTOC problem. Constraint (7.33) is a linear constraint on the inputs so that the cables cannot have negative rest lengths. Finally, constraint (7.34) denotes a minimum bound on the second element in the state, the z-position, which prevents collision between the moving vertebra and the static vertebra.

Objective Function

The objective function for this formulation is comprised of quadratic weights on the state and input tracking errors. As opposed to the smoothing formulation, which included non-traditional terms, the objective function here is exactly the same as with standard MPC. Using similar notation as in equations (7.27) and (7.28),

$$p(\boldsymbol{\xi}_{t+N|t}) = \|\boldsymbol{\xi}_{t+N|t} - \bar{\boldsymbol{\xi}}_{t+N|t}\|^2_{\mathbf{Q}}, \tag{7.35}$$

$$q(\boldsymbol{\xi}_{t+k|t}, \mathbf{u}_{t+k|t}) = \|\boldsymbol{\xi}_{t+k|t} - \bar{\boldsymbol{\xi}}_{t+k|t}\|^2_{\mathbf{Q}} + \|\mathbf{u}_{t+k|t} - \bar{\mathbf{u}}_{t+k|t}\|^2_{\mathbf{R}}. \tag{7.36}$$

Here, $\mathbf{Q}$ and $\mathbf{R}$ are constant diagonal weighing matrices which penalize state and input tracking errors respectively, defined similarly to the smoothing formulation, but do not vary with the horizon step as with the $\mathbf{Q}^k$ terms in eqn. (7.28). Specifically, these weights are

$$\mathbf{Q} = diag(w_2,\ w_2,\ w_2 \mid 0...0) \in \mathbb{R}^{6\times 6}, \tag{7.37}$$

$$\mathbf{R} = diag(w_3,\ w_3,\ w_3,\ w_3) \in \mathbb{R}^{4\times 4}. \tag{7.38}$$

As with eqn. (7.29), the $\mathbf{Q}$ matrix does not penalize velocity states. Table 7.3 lists all the constants for this controller, including the constraints and the objective function, with units.

7.3.4 Controller Comparison

The differences between the two controller formulations (Sec. 7.3.2 and 7.3.3) are summarized in Table 7.1. In addition to the inherent difference between the tracking of one vertebra versus

Table 7.3: Input tracking controller weights and constants.

Constant:	Value:		Interpretation:
N	4	no units	Horizon Length
u_{min}	0.0	meters (cable)	Min. Cable Length
w_1	0.075	meters (vertebra position)	Vertebra Anti-Collision
w_2	1	no units	State Tracking Penalty
w_3	10	no units	Input Tracking Penalty

3 vertebrae, and the difference between the 2D and 3D models that are tracked, three major considerations are present.

First, the controller with the IS optimization is much more general, and does away with the smoothing terms. This reduces the complexity of the CFTOC problem, thus removing most of the need for tuning optimization weights (compare Table 7.2 versus Table 7.3). Second, the controller with the IS optimization moves some computational load offline, since the MPC problem now has fewer terms. Third, in contrast to those benefits, the MPC plus IS controller required a faster simulation rate as tested here, with the discretization timestep of $dt = 1e^{-5}$ versus $1e^{-3}$ for the smoothing controller. These three changes represent the tradeoffs between tuning requirements and performance implications of either controller.

7.4 Simulation Results

Two sets of simulations are presented here, one for the controller with MPC and smoothing terms, and one for the controller with MPC and inverse statics reference input generation/tracking. All simulation work used the YALMIP toolbox in MATLAB [116], with Gurobi as the solver. All code is available online[1].

Both controllers tracked the vertebrae states with sufficiently low error as to justify their use. The smoothing controller tracked with lower error, after an initial transient response, but had higher computational complexity and tuning requirements. The more general input-tracking controller exhibited lag, and thus larger tracking errors, but with lower computational overhead and with significantly less hand-tuning.

7.4.1 Noise Model

For both models and controllers, simulations are also performed with noise, in order to test closed-loop performance. Process noise is implemented by adding a sample from a normally-distributed random variable to the system dynamics during the simulation. For example, denoting the timestep as Δt, the forward-Euler-simulated model for the 2D spine is

$$\boldsymbol{\xi}_{t+1} = \boldsymbol{\xi}_{t|t} + \boldsymbol{g}(\boldsymbol{\xi}_{t|t}, \boldsymbol{u}^*_{t|t})(\Delta t) + \boldsymbol{E}\boldsymbol{\epsilon}_t \tag{7.39}$$

[1] https://github.com/BerkeleyExpertSystemTechnologiesLab/ultra-spine-simulations

where $\boldsymbol{\epsilon}_t$ is a sample drawn from $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}_2, \mathbf{I}_2) \in \mathbb{R}^2$ at time t. The the weighting matrix $\boldsymbol{E}$ scales the variance of the random variable. The simulation of the 3D model, using Runge-Kutta integration, works similarly.

$\boldsymbol{E}$ has different weights according to the position (and angle) states or the velocities, and is sized appropriately for either the 3D or 2D state model:

$$\boldsymbol{E}^{3D} = \mathbf{1}_3 \otimes \begin{bmatrix} w_{12}\mathbf{1}_6 & \mathbf{0}_6 \\ \mathbf{0}_6 & w_{13}\mathbf{1}_6 \end{bmatrix} \in \mathbb{R}^{36\times 2} \tag{7.40}$$

$$\boldsymbol{E}^{2D} = \begin{bmatrix} w_{14}\mathbf{1}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & w_{15}\mathbf{1}_3 \end{bmatrix} \in \mathbb{R}^{6\times 2} \tag{7.41}$$

Table 7.4 lists the values for $\{w_{12} \dots w_{15}\}$. These values are selected such that the standard deviation of $\boldsymbol{\epsilon}$ is scaled to roughly 33% of the maximum position displacement (or maximum velocity, respectively) of the robot between timesteps in the reference state trajectory.

Table 7.4: Noise model weights.

Constant:	Value:		Interpretation:
w_{12}	$5e^{-4}$	meters, rad.	Noise std. dev., positions/angles, 3D model
w_{13}	$2e^{-4}$	m/s, rad/s	Noise std. dev., velocities, 3D model
w_{14}	$1.6e^{-5}$	meters, rad.	Noise std. dev., positions/angles, 2D model
w_{15}	$6.6e^{-6}$	m/s, rad/s	Noise std. dev., velocities, 2D model

7.4.2 Computational Performance

The optimization problem for the MPC plus smoothing controller, applied to the 3D model (from Sec. 7.3.2) took $0.5 - 1$ sec. to solve at each timestep, using the Gurobi solver. The optimization problem for the MPC plus inverse statics reference input tracking controller, applied to the 2D model (from sec. 7.3.3), took $0.15 - 0.2$ sec. to solve at each timestep. The inverse statics optimization procedure (Alg. 1) is performed offline before the closed-loop tests begin, so is not timed; however, it solves rapidly enough for practical use.

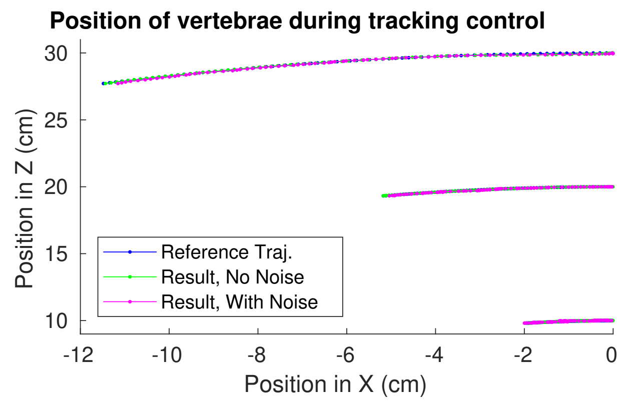
The optimization problem for the smoothing controller, applied to the 3D model (from sec. 7.3.2) took $0.5 - 1$ sec. to solve at each timestep, using the Gurobi solver. The optimization problem for the reference input tracking controller, applied to the 2D model (from sec. 7.3.3), took $0.15 - 0.2$ sec. to solve at each timestep. The inverse statics inputs for the reference input tracking controller are calculated offline, so are not included in these statistics, but are of low enough computational load so as to be negligible in comparison.

7.4.3 Controller with MPC and Smoothing Terms

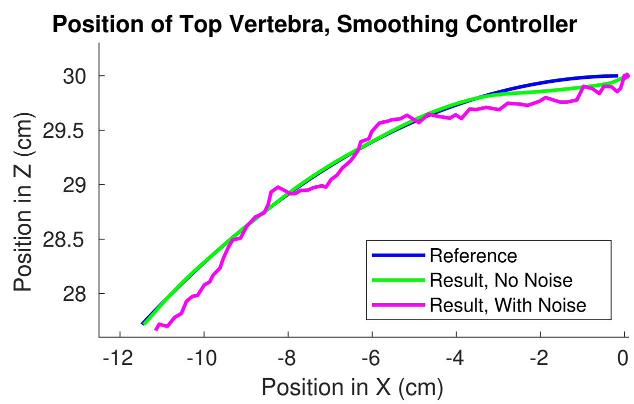
Fig. 7.4a shows the paths of the vertebrae in the 3D, three-vertebra simulation, using the smoothing constraint controller, in the X-Z plane as they sweep through their counterclockwise bending motion.

Fig. 7.4a includes the reference trajectory (blue), the resulting trajectory with the smoothing MPC controller and no noise (green), and a representative result of controller with added noise (magenta). Fig. 7.4b shows a larger view of the top vertebra center of mass, which had the largest tracking errors of the three vertebrae, and which is used for comparison with the 2D single-vertebra model below.

The tracking errors for each state, for each vertebra, for both simulations (with and without noise) are shown in Fig. 7.5. In both simulations, an initial transient is observed in the X-position and γ-angle states. This is possibly due to a zero initial velocity of the vertebrae, requiring the spine to rapidly move at the start of its simulation to "catch up" with the trajectory. After that, all errors trend to zero, with the expected oscillations in the simulation with noise.



(a) Positions of all three vertebrae.



(b) Position of the top vertebra.

Figure 7.4: Positions in the X-Z plane for the 3D, three-vertebra model with the smoothing controller, as the robot performs a counterclockwise bend.

7.4.4 Controller with MPC and Inverse Statics Optimization

Fig. 7.6a shows the path of the single vertebra in the 2D simulation, using the controller with MPC plus inverse statics reference input tracking, as it sweeps through its counterclockwise bending motion. As with Fig. 7.4a and 7.4b, the reference state trajectory is included (in blue) alongside results from the controller with no noise (green) and from a representative simulation with noise (magenta.) The vertebra follows the path of the of the kinematic states, but experiences some accumulation of lag. The results show that the closed-loop controller is noise-insensitive, alongside accurate tracking, but that the lag occurs in all circumstances.

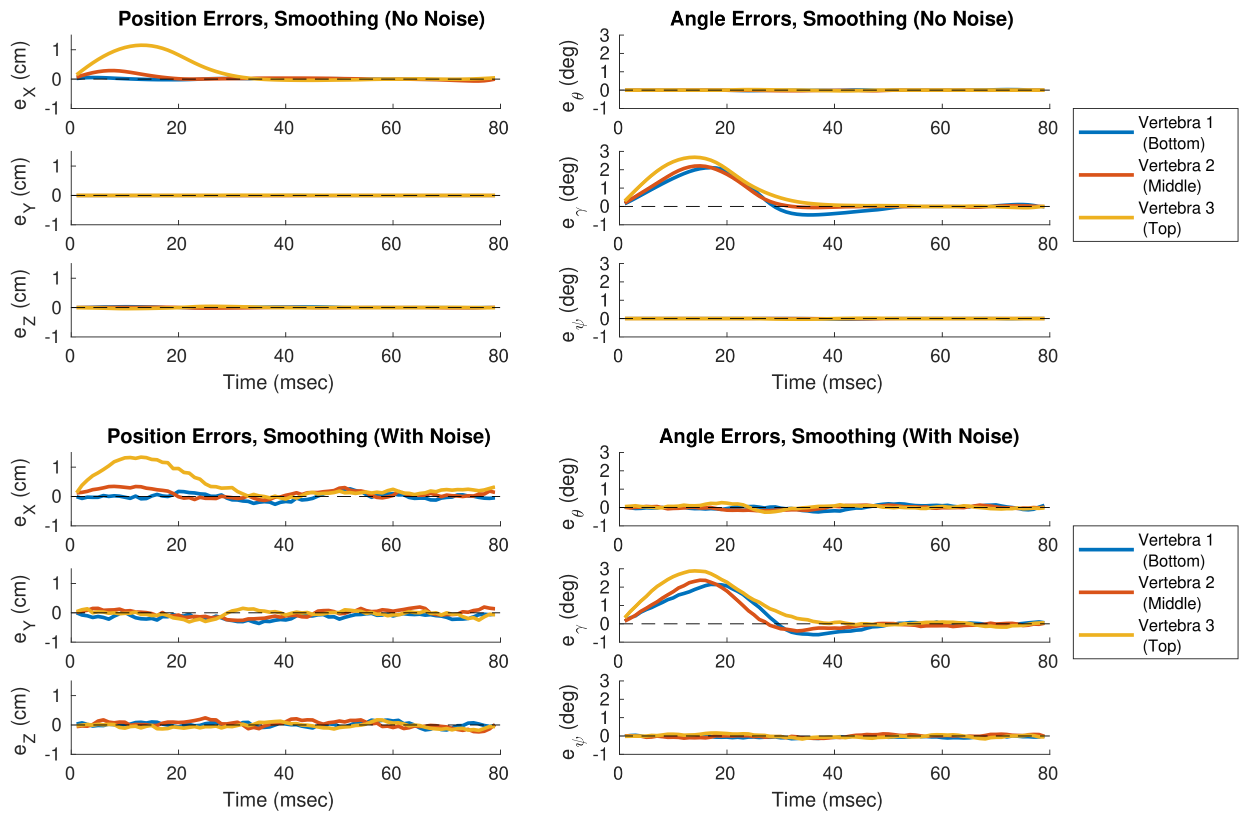
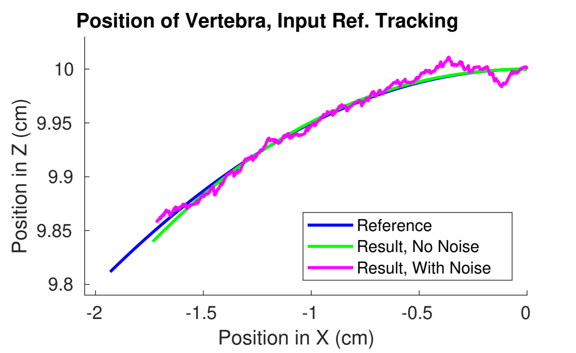


Figure 7.5: Tracking errors in system states for the 3D, three-vertebra model using the smoothing controller, with and without noise. Position states (x, y, z) on the left with units of cm, Euler angles (θ, γ, ψ) on the right with units of degrees.

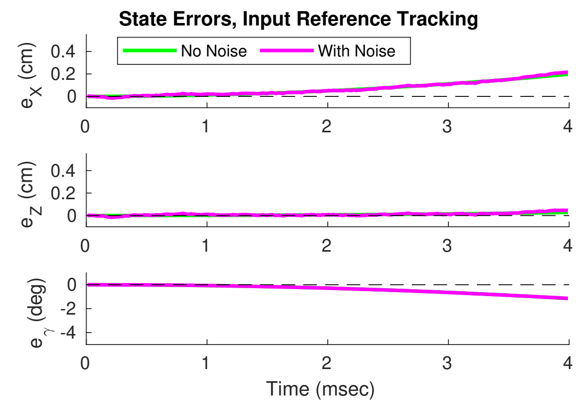
The tracking errors for each state are shown in Fig. 7.6b, using the same convention as Fig. 7.5. The controller accumulates lag throughout the simulation, and the errors do not converge. This is expected, since the tracked inputs are inverse statics and not dynamics, and this simulation setup violates the assumption of quasi-static movement. Since the results presented here are used to compare with the smoothing controller, the simulations use the same setup with only dynamic movement. It is expected that given a setup where the controller has the opportunity to settle, the errors would converge.

7.4.5 Control of Different Spines

The proposed controller that combines MPC with inverse statics for reference input tracking has significantly fewer tuning parameters. It is thus easily extendable to different sizes and shapes of spines, whereas a large amount of tuning may have otherwise been required. In order to illustrate this, the controller was tested on a different 2D spine, with a different size and shape of vertebra.



(a) Positions in the X-Z plane as the robot performs a counterclockwise bend. The vertebra tracks the trajectory, but accumulates more lag than the smoothing controller.



(b) Tracking errors in system states. Position errors are in cm, rotation errors are in degrees. The drift shown here arises from controller lag.

Figure 7.6: Simulations results for the 2D single-vertebra model, using the controller with MPC plus inverse statics reference input trajectory generation/tracking.

Control results (Fig. 7.7) show equivalent performance to the original vertebrae of Sec. 7.4, despite the size and geometry change. For these tests, no changes were made to the inverse statics algorithm, nor to any of the constants in Table 7.3.

This differently-shaped spine still retained the same number of point masses, and is symmetric (to satisfy the assumptions of the inverse statics algorithm), but is now larger and heavier, with different angles between its bars. The vertebra weighed a total of 0.2 kg, and the positions of its point masses (nodes) are, in cm,

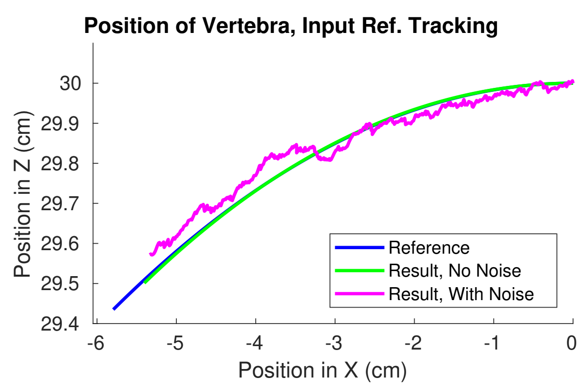


Figure 7.7: Additional test of the reference input tracking controller with the larger, differently-shaped vertebra. This controller tracks this vertebra in the same way as Fig. 7.6a with no need to change any tuning constants.

$$\begin{bmatrix}\boldsymbol{a}_1 & \boldsymbol{a}_2 & \boldsymbol{a}_3 & \boldsymbol{a}_4\end{bmatrix} = \begin{bmatrix}0 & 20 & -20 & 0 \\ 0 & -20 & -20 & 20\end{bmatrix} \tag{7.42}$$

The reference state trajectory is adjusted to match the new size, with the same β_1^{max} but a height and radius of $\bar{z}_1(0) = r_1 = 0.3$ m. Simulated noise is scaled in the same way. Accounting for the number of timesteps and distance traveled in comparison to the top vertebra of the 3D model, the weights for the noise are $w_{14} = 5e^{-4}$ and $w_{15} = 2e^{-4}$.

7.5 Discussion

Both controllers exhibit state tracking characteristics which could be used in different environments for effective closed-loop control. The smoothing controller tracked with lower error, after an initial transient response, but had higher computational complexity and tuning requirements. The controller with inverse statics tracking, which is more general, exhibited lag and thus larger tracking errors, but with lower computational overhead and with significantly less hand-tuning. These are the first controllers that track a state-space trajectory of a tensegrity spine robot in closed-loop, and the first which shows noise insensitivity.

7.5.1 Computational Performance

The lengths of time taken to solve the optimization problem for each controller (0.5-1 sec. and 0.15-0.2 sec.) were longer than the timesteps of each respective simulation ($1e^{-3}$ and $1e^{-5}$ sec.). Thus, the optimization procedure will need to be made more efficient before using this controller in hardware. One approach that may reduce solver time is the calculation of a symbolic Jacobian for the $\mathbf{A}_t$ and $\mathbf{B}_t$ matrices, reducing the computational load in the linearization.

7.5.2 Tracking Performance Comparison

The controller with MPC and the inverse statics optimization removed the need for hand-tuned smoothing terms, but exhibited lag in tracking a highly-dynamic state trajectory. This motivates the use of either controller in different settings. The MPC plus smoothing controller may be appropriate for high-performance dynamic tracking, when the control system designer is able to tune the weights and constraints. In contrast, the MPC plus inverse statics optimization controller may be appropriate for more pseudo-static movements, but can be implemented more reliably and on more systems without the tuned smoothing terms.

Both approaches demonstrated noise insensitivity as well as some robustness to model mismatch (since both controllers utilize a time-varying linearization.) However, these controllers have yet to be tested with unknown external loads or disturbances. It is anticipated that such settings may have more impact on the controller with MPC plus inverse statics, since it relies more heavily on open-loop behavior. In this case, approaches may exist for tuning disturbance rejection, such as increasing pretension in the reference input trajectory via the lower bound c in eqn. (5.50).

7.5.3 Limitations Of Comparison

The results provided here compare the top vertebra of the 3D model to the single vertebra in the 2D model. This comparison is chosen to demonstrate the largest errors of each simulation. Thus, Fig. 7.4b and 7.6a represent the same geometry of state trajectory, but do not represent the exact same system model.

Though the controller with inverse statics optimization is prototyped in a reduced-order version of the spine, the formulation is general enough to be applied to a multiple-vertebra, 3D spine. However, such simulations have not been implemented, and as such, it is unknown if some combination of both optimization problems in Sec. 7.3.2 and 7.3.3 may still be required for the higher-dimensional system.

7.5.4 Future Work

Future work will focus in two areas. First, performance improvements are needed. In addition to the computational aspects mentioned above, better tracking may be achieved using inverse dynamics instead of inverse statics solutions. Using higher-fidelity models, or more sophisticated numerical techniques, may allow for a lower-frequency controller to show good tracking performance.

In addition, hardware experiments using such a lower-frequency controller will be conducted in future work. Significant mechanical design challenges remain before an appropriate physical prototype can be constructed, particularly with actuation (the dimension of $\mathbf{u}$) and sensing (since this chapter uses state feedback.) Work is ongoing on the robot from Chap. 6, which may eventually be the test platform for these experiments.

Chapter 8

Stability and Control of Lagrangian Systems with Statically Conservative Forces

The tensegrity spine robot studied in this dissertation exhibits a number of properties that make control challenging, such as high dimensionality, nonlinearities, input saturation, and the hybrid behavior of cables that become slack. The previous chapter addressed these issues through model-predictive control, using a constrained optimization program. However, the MPC approach also showed significant drawbacks. The controller had computational challenges and was not real time. More importantly, no stability proof was possible, given the variety of approximations for tractability.

This chapter proposes a different method for addressing similar challenges in robotics control. Here, an energy-based approach, modified from the passivity-based control framework reviewed in Chap. 3, addresses the same issues in ways that do not come with computational challenges, while also providing a stability proof. The energy-based analysis inherently applies to nonlinear systems, distributed control is used for issues of dimensionality, and hybrid system behavior is directly accounted for in the stability analysis.

The class of systems considered here are those with *statically conservative* applied forces, a relatively obscure concept that nonetheless captures a wide range of mechanical models. In particular, this chapter proves that models of slack or viscoelastic cables in a cable-driven robot are statically conservative. These systems are then analyzed in the new framework. Though a controller for tensegrity spines is not derived here, the proposed framework addresses each major challenge in control of such systems (nonlinearities, dimensionality, hybrid behavior), with a clear framework for future application to systems of rigid bodies.

The following chapter first addresses the concept of statically conservative forces, adapting the theory from Chap. 3 to the new case. The new stability proofs are then applied to particles with damped central forces, as a model for a robot's cables. A simplified model of a cable-driven robot, as an example with this structure, is then controlled in setpoint regulation.

8.1 Motivation: Challenges with Slack Cables in Cable-Driven Robots

A variety of challenges remain in the control of tensegrity spines using the optimization-based control approaches from Chap. 7. These challenges are not necessarily due to tensegrity spines themselves, but instead are representative of larger problems in robotics:

1. Hybrid dynamics. The cables of a tensegrity spine can go slack, changing the equations of motion in a system.
2. High dimensionality. The structure consists of many rigid bodies, with a high-dimensional state space, making computational procedures such as model-predictive control either inefficient or impractical.
3. Stability. Tensegrity spines are nonlinear in addition to hybrid, and so the linearized controllers in Chap. 7 could not provide stability guarantees.

Identifying these challenges prompts the use of inherently nonlinear control systems, particularly those which can be developed in a distributed manner to eliminate dimensionality problems. The following chapter develops such a controller, based on energy-theoretic concepts of stability. Though the results here have not yet been applied to tensegrity spines, the fundamental framework and its verification on a related system address each problem listed above. The results prompt future use of these controllers on tensegrity spines.

8.1.1 Lagrange's Equations Facilitate Energy Analysis

Lagrange's equations of motion were used in Chap. 7 to derive the dynamics of a tensegrity spine. Recall from Chap. 3 that Lagrange's equations are a method of writing the dynamics of mechanical systems from an energy-focused perspective, and are equivalent to using Newton's laws. A mechanical system with n degrees of freedom is parameterized by its *generalized coordinates* $\mathbf{q} = [q_1 \dots q_n]^\top \in \mathcal{X}^n \subseteq R^n$, the corresponding *generalized velocities* $\dot{\mathbf{q}} \in \mathbb{R}^n$, and the applied *generalized forces* $\mathbf{Q} = [Q_1 \dots Q_n]^\top \in \mathbb{R}^n$, all of which have application-dependent interpretations (Chap. 3.) The system is modeled via its kinetic (T) and potential (U) energy. The sum of these is the Hamiltonian (H), and the difference is the Lagrangian (L),

$$T \in \mathbb{R}, \qquad U \in \mathbb{R}, \qquad L := T - U, \qquad H := T + U.$$

Then, the following set of equations, referred to as Lagrange's equations, hold based on a the use of Hamilton's principle from the calculus of variations.

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \left(\frac{\partial L}{\partial q_j}\right) = Q_j, \qquad j = 1 \dots n. \tag{8.1}$$

Equivalently, expressed in vector form,

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \left(\frac{\partial L}{\partial \mathbf{q}}\right) = \mathbf{Q}, \tag{8.2}$$

with the partial derivative notation discussed in Chap. 3.

8.1.2 Using Passivity-Based Control for Stability

There are a variety of ways that systems of this form could be analyzed for stability, including optimization-based methods. However, an intuitive approach is analyzing the system's total energy H as a Lyapunov candidate (Chap. 3), or for passivity, a storage function candidate. An energy analysis is the underlying principle of passivity-based control (PBC) of Lagrangian systems.

To do so, the generalized force is assumed to depend on the system states, $\mathbf{Q} = \mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}})$, and is assumed to be a *memoryless nonlinearity* such that it does not have its own internal state. The commonly-used presentation of PBC for Lagrangian systems [196, 144, 32] then requires that $\mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}})$ can be additively separated into one component that is a function of coordinates and another a function of velocities, chosen with a negative sign convention,

$$\mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}}) = -\mathbf{Q}^s(\mathbf{q}) - \mathbf{Q}^d(\dot{\mathbf{q}}).$$

The coordinates-only component must then also be a conservative (generalized) force, as in

$$Q_j^s = \frac{\partial U}{\partial q_j}, \qquad j = 1, \ldots, n, \tag{8.3}$$

incorporating $\mathbf{Q}^s$ into the potential energy U, and leaving the remaining nonconservative component to determine the energy change in the system (Lemma 3.1.0.1 and Prop. 3.3.2) via the work-energy principle,

$$\dot{H} = -\dot{\mathbf{q}}^\top \mathbf{Q}^d(\dot{\mathbf{q}}).$$

Stability then comes if $\mathbf{Q}^d(\dot{\mathbf{q}})$ is an input-strictly passive (definition in Sec. 3.2.1) memoryless nonlinearity,

$$\dot{\mathbf{q}}^\top \mathbf{Q}^d(\dot{\mathbf{q}}) \geq \gamma^2 \dot{\mathbf{q}}^\top \dot{\mathbf{q}}, \qquad \gamma \in \mathbb{R} > 0, \qquad \forall\, \dot{\mathbf{q}} \qquad \Rightarrow \qquad \dot{H} \leq 0,$$

via the arguments related to LaSalle's Invariance Theorem in Prop. 3.3.3.

The canonical example of this situation is a one-dimensional ($q \in \mathbb{R}$) nonlinear spring-damper system [171],

$$F(q, \dot{q}) = -F^s(q) - F^d(\dot{q}),$$

with the linear case $F^s = kq$ and $F^d = c\dot{q}$ clearly stable via the above with a positive damping constant $c > 0$.

8.1.3 Energy Analysis when Forces are not Separable

However, what happens if the (generalized) force is not separable into one conservative and another input-strictly-passive nonconservative component? For example, a flexible cable with length ℓ that goes slack could have a model (autonomous version of that from Chap. 7) of the form

$$F = \begin{cases} k\ell + c\dot{\ell}, & \text{if} \quad k\ell + c\dot{\ell} \geq 0 \\ 0, & \text{if} \quad k\ell + c\dot{\ell} < 0, \end{cases} \tag{8.4}$$

which cannot be separated into one function of only ℓ and another of only $\dot{\ell}$. This is the source of the hybrid dynamics of the tensegrity spines in this dissertation: cables have different constitutive equations depending on both cable length ℓ and length change $\dot{\ell}$.

The following issues arise.

1. The potential energy U may not be bounded, which is a requirement for applying Prop. 3.3.3. The tensegrity spine in Chap. 7 only has gravitational potential energy for U, which is unbounded below.

2. Despite this, there may still exist equilibrium points. The spine may oscillate with cables becoming slack then re-tensioning, but if pretensioned properly, could settle to an equilibrium.

3. The equilibrium point $\bar{\mathbf{q}}$, if it exists, is not at the point specified in Prop. 3.3.3. By taking Lagrange's equations and substituting for zero velocity (well known to be the condition for equilibrium [77]), the kinetic energy is $T = 0$, and the equilibrium condition for equation (8.2) becomes

$$\left.\frac{\partial U}{\partial \mathbf{q}}\right|_{\mathbf{q}=\bar{\mathbf{q}}} = \mathbf{Q}(\bar{\mathbf{q}}, \mathbf{0}). \tag{8.5}$$

   In comparison, as in Prop. 3.3.3, with $\mathbf{Q}$ separable and the damping force $\mathbf{Q}^d(\mathbf{0}) = \mathbf{0}$, the equilibrium is at the minimum of the potential energy:

$$\left.\frac{\partial U}{\partial \mathbf{q}}\right|_{\mathbf{q}=\bar{\mathbf{q}}} = \mathbf{0}.$$

Formulating a Lyapunov (or storage) function candidate that has its minimum at the equilibrium would require combining the right-hand side of eqn. (8.5) with the left-hand side. The following chapter does exactly so, proposing the new concept of *augmented potential energy* that has the equilibrium as its minimum, and giving conditions under which the augmented potential energy decreases via a modified version of the work-energy theorem.

Defining and using the augmented potential energy for stability analysis requires conditions on $\mathbf{Q}(\mathbf{q}, \mathbf{0})$ so that it can be combined with U. Specifically, eqn. (8.5) implies that $\mathbf{Q}(\mathbf{q}, \mathbf{0})$ must be the gradient of some scalar potential function; i.e, $\mathbf{Q}$ must be a gradient of some scalar function when the system is at rest. This is the definition of a *statically conservative* force adopted here.

Definition 8.1.1. Statically Conservative Force.

A (generalized) force $Q_j(\mathbf{q}, \dot{\mathbf{q}})$ that is a function of both the generalized coordinates and generalized velocities, in vector form $\mathbf{Q} = [Q_1 \ldots Q_n]^\top \in \mathbb{R}^n$, is *statically conservative* if it is the gradient of a scalar potential function when at rest. Specifically, defining

$$\sigma_j(\mathbf{q}) := Q_j(\mathbf{q}, \mathbf{0}), \qquad \boldsymbol{\sigma} := [\sigma_1 \ldots \sigma_n]^\top \in \mathbb{R}^n, \tag{8.6}$$

then $\mathbf{Q}$ is statically conservative if

$$\exists\, U^f \quad \text{s.t.} \quad \frac{\partial U^f}{\partial q_j} = \sigma_j, \quad \forall\, j = 1 \ldots n, \quad \text{equivalently} \quad \frac{\partial U^f}{\partial \mathbf{q}} = \boldsymbol{\sigma}. \tag{8.7}$$

It appears that this concept is relatively obscure in the literature, with no commonly-accepted nomenclature. The use of the term ‘statically conservative’ above comes from [58], one of the few papers that considers a case similar to this one.

8.1.4 Other Approaches to Similar Problems

The above analysis is for an autonomous system, one in which the control loop has already been closed. Such will be the case with the cables in the application below. However, there are other assumptions that could be made on $\mathbf{Q}$ that lead to stabilizing control strategies. For example, assuming that $\mathbf{Q}$ is affine in an external input allows incorporating it into the Lagrangian in some cases [32],

$$\mathbf{Q}(\mathbf{q}, \mathbf{u}) = \mathbf{J}(\mathbf{q})\mathbf{u}, \qquad \bar{L} := L + \mathbf{q}^\top \mathbf{Q}(\mathbf{q}, \mathbf{u}),$$

or interconnecting the system with forces that have internal state [101]. However, none of these are readily able to incorporate forces with coupled $\mathbf{q}$ and $\dot{\mathbf{q}}$ terms.

Other approaches to solving this equilibrium problem in the context of passivity have recently been investigated in the literature. Specifically, Meissen et. al. [125] proposed a controller that uses equilibrium-independent dissipativity (EID) for the similar issue of an a-priori unknown equilibrium point. In that example, with a quadcopter carrying a cable-suspended payload, the cables are considered as a separate interconnected subsystem for which equilibrium is addressed in the EID framework. The analysis (and resulting controller) proposed in this chapter can therefore be viewed as a different approach to a similar problem, with particles suspended in networks of cables, where the system is analyzed as a whole but distributed control is developed. The approach here also not only accounts for flexibility in the cables, but exploits the energy dissipation therein for a stability proof.

8.1.5 Controlling Systems of Particles as Progress Towards Tensegrity Robots

Recall from Chap. 7 that a tensegrity spine can be represented by a system of particles with integrable constraints imposed between them. These constraints define the shape of a rigid body in the tensegrity spine, giving an approximation to rigid-body dynamics. Without these constraints, these systems become a *cable network*, where particles move freely. Cable networks have the convenient property that all forces between particles are *central forces* (see Sec. 8.3), and it will be shown later that these forces are always statically conservative. This chapter therefore chooses to address cable networks as an application for the proposed control theory.

By addressing cable networks as a first application of the theory in this chapter, the approach can be adapted for the more general case of constrained systems of particles, which can then model a tensegrity spine as in Chap. 7. The control systems in this section do not directly apply to tensegrity spines, and as such, are not used for those spines. Nor is an unconstrained cable network intended as a model for a tensegrity structure. However, it is hypothesized that the static conservation properties of cable networks will also apply to tensegrity structures upon further analysis.

Though tensegrity spines are not studied in this chapter, a different cable-driven robot is modeled as a cable network system in Sec. 8.6 for purposes of validating the theory presented here. The approximation of the example cable-driven robot's rigid body as a point mass is only valid under certain conditions, such as small moments of inertia, and when forces are applied close to the body's center of mass. This is the case in some cable-driven robots [218], but nonetheless introduces modeling error. Future work will examine the implications of this approximation in various settings.

8.2 Passivity and Stability of Lagrangian Systems with Statically Conservative Forces

The following sections derive passivity and stability conditions for Lagrangian systems with applied forces that are statically conservative. Though the eventual goal for this control purpose is stability, the first proposition addresses passivity: it is expected that, in future work, there may be control frameworks which make use of interconnections with these systems. The passivity condition is also a subset of the stability condition, and thus the two are addressed in sequence.

The following relies heavily on the proofs from prior work in Chap. 3. Any missing background, context, or definitions are given there - for example, with equilibrium points versus configurations, the use of partial derivatives in $\mathbb{R}^n$ versus the gradient in $\mathbb{E}^3$, etc.

8.2.1 Passivity of Lagrangian Systems with Statically Conservative Forces

For a passivity analysis, assume that a system modeled by Lagrange's equations (8.2) has an external input added to its generalized force, as with Prop. 3.3.1. The following then holds.

Proposition 8.2.1. *Passivity of Lagrangian Systems with Statically Conservative Forces.*

Consider a system that is described by Lagrangian mechanics, eqn. (8.2), with n *generalized coordinates, velocities, and forces: $q_j \in \mathcal{X}_j \subseteq \mathbb{R}$, $\dot{q}_j$, $Q_j \in \mathbb{R}$, $j = 1 \ldots n$. Assume that there is an additional generalized force that is an input to the system, u_j, $j = 1 \ldots n$, that a negative sign convention is chosen for Q_j, and that $Q_j = Q_j(\mathbf{q}, \dot{\mathbf{q}})$ is a function of both the coordinates and velocities and is a memoryless nonlinearity (stateless.) Specifically,*

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \left(\frac{\partial L}{\partial q_j}\right) = -Q_j(\mathbf{q}, \dot{\mathbf{q}}) + u_j, \qquad j = 1 \ldots n,$$

or in vector form

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \left(\frac{\partial L}{\partial \mathbf{q}}\right) = -\mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{u}, \tag{8.8}$$

where $L = T - U$ is the Lagrangian, T is the kinetic energy, and U is the potential energy.

If the following hold:

1. *The generalized force $\mathbf{Q}$ is statically conservative, i.e. (from Defn. 8.1.1),*

$$\boldsymbol{\sigma}(\mathbf{q}) := \mathbf{Q}(\mathbf{q}, \mathbf{0}), \qquad \exists\, U^f \ \ s.t. \quad \frac{\partial U^f}{\partial \mathbf{q}} = \boldsymbol{\sigma},$$

2. *The* augmented potential energy*, defined as*

$$\bar{U} := U + U^f, \tag{8.9}$$

is bounded below by a constant, $\bar{U} \geq c \in \mathbb{R}$,

3. *The difference between $\mathbf{Q}$ and its statically conservative component $\boldsymbol{\sigma}$, defined as*

$$\mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) := \mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}}) - \boldsymbol{\sigma}(\mathbf{q}), \tag{8.10}$$

satisfies the following inequality:

$$\dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) \geq 0 \quad \forall \mathbf{q}, \dot{\mathbf{q}}, \tag{8.11}$$

or equivalently when written as the memoryless nonlinearity $\mathbf{y} = \mathbf{S}(\mathbf{v}_1, \mathbf{v}_2)$, is dissipative with respect to the passivity-like quadratic supply rate $\mathbf{v}_2^\top \mathbf{y}$:

$$s(\mathbf{v},\mathbf{y}) = \begin{bmatrix}\mathbf{v}_1\\ \mathbf{v}_2\\ \mathbf{y}\end{bmatrix}^\top \mathbf{X} \begin{bmatrix}\mathbf{v}_1\\ \mathbf{v}_2\\ \mathbf{y}\end{bmatrix} \geq 0 \quad \forall\, \mathbf{v}_1, \mathbf{v}_2, \qquad X = \begin{bmatrix}0 & 0 & 0\\ 0 & 0 & \frac{1}{2}\mathbf{I}\\ 0 & \frac{1}{2}\mathbf{I} & 0\end{bmatrix},$$

Then, the system is passive from $\mathbf{u}$ *to* $\dot{\mathbf{q}}$*, i.e. that*

$$V(\mathbf{q}(\tau)), \dot{\mathbf{q}}(\tau)) - V(\mathbf{q}(0)), \dot{\mathbf{q}}(0)) \leq \int_0^\tau \dot{\mathbf{q}}^\top \mathbf{u}\, dt \tag{8.12}$$

for every input signal $\mathbf{u}$ *and every time interval* $t = [0, \tau)$ *in the interval of existence of the solution* $\mathbf{q}(t)$*, with the storage function* $V(\cdot,\cdot) : \mathcal{X} \times \mathbb{R}^n \mapsto \mathbb{R}$ *chosen to be the augmented total energy* $\bar{H} = T + \bar{U}$ *minus the bounding constant for the augmented potential energy,*

$$V(\mathbf{q}, \dot{\mathbf{q}}) := \bar{H}(\mathbf{q}, \dot{\mathbf{q}}) - c. \tag{8.13}$$

Proof. Consider the storage function candidate,

$$V(\mathbf{q}, \dot{\mathbf{q}}) = \bar{H}(\mathbf{q}, \dot{\mathbf{q}}) - c.$$

First, justify that this is a valid storage function. The total kinetic energy in mechanical systems is always nonnegative, $T \geq 0$. So,

$$T \geq 0,\ \bar{U} - c \geq 0 \quad \Rightarrow \quad T + \bar{U} - c \geq 0 \quad \Rightarrow \quad \bar{H} - c \geq 0.$$

Thus, $V(\cdot,\cdot)$ is nonnegative, and is a valid storage function. As with Prop. 3.3.3, the storage function $V(\cdot,\cdot)$ is not necessarily zero for zero argument, which can be accounted for via a coordinate transformation (not performed here.)

Next, from Lemma 3.1.0.1, the time derivative of the total system energy is

$$\dot{H} = -\dot{\mathbf{q}}^\top \mathbf{Q} + \dot{\mathbf{q}}^\top \mathbf{u}. \tag{8.14}$$

The time derivative of the storage function candidate is

$$\dot{V} = \dot{H} + \dot{U}^f, \tag{8.15}$$

so via substitution,

$$\dot{V} = -\dot{\mathbf{q}}^\top \mathbf{Q} + \dot{\mathbf{q}}^\top \mathbf{u} + \dot{U}^f. \tag{8.16}$$

Expanding $\dot{U}^f$ using the chain rule, since U^f is only a function of $\mathbf{q}$,

$$\dot{U}^f = \sum_{j=1}^n \frac{\partial U^f}{\partial q_j} \dot{q}_j = \sum_{j=1}^n \sigma_j \dot{q}_j = \dot{\mathbf{q}}^\top \boldsymbol{\sigma}. \tag{8.17}$$

Substituting eqn. (8.17) into the expression for the time derivative of the storage function candidate (8.16),

$$\dot{V} = -\dot{\mathbf{q}}^\top \mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}}) + \dot{\mathbf{q}}^\top \mathbf{u} + \dot{\mathbf{q}}^\top \boldsymbol{\sigma}(\mathbf{q}) \tag{8.18}$$

$$\dot{V} = -\dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) + \dot{\mathbf{q}}^\top \mathbf{u}. \tag{8.19}$$

The dissipativity assumption on the difference in generalized forces gives that $\dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) \geq 0$, so therefore

$$-\dot{\mathbf{q}}^\top \mathbf{S} \leq 0 \qquad \forall\, \mathbf{q}, \dot{\mathbf{q}}, \tag{8.20}$$

$$\dot{V} - \dot{\mathbf{q}}^\top \mathbf{u} \leq 0 \qquad \forall\, \mathbf{q}, \dot{\mathbf{q}}, \tag{8.21}$$

$$V(\mathbf{q}(\tau), \dot{\mathbf{q}}(\tau)) - V(\mathbf{q}(0), \dot{\mathbf{q}}(0)) \leq \int_0^\tau \dot{\mathbf{q}}(t)^\top \mathbf{u}(t)\, dt \qquad \forall\, \mathbf{q}, \dot{\mathbf{q}}, \tag{8.22}$$

and the map from $\mathbf{u}$ to $\dot{\mathbf{q}}$ is passive.

$\square$

Remark. The above proof may at first seem to require finding a U^f, of which there may be many. In practice, this is not necessary: the above can be satisfied only examining $\mathbf{Q}$ and $\boldsymbol{\sigma}$. Determining if a U^f exists can done via the common 'curl test' for forces in $\mathbb{E}^3$. Here, the partial derivative for the generalized force $\boldsymbol{\sigma}$ becomes a gradient for a force $\mathbf{f}$ in $\mathbb{E}^3$ (see Chap. 3),

$$\exists\, U^f \text{ s.t. } \nabla U^f = \mathbf{f} \text{ iff } \operatorname{curl}(\mathbf{f}) = \mathbf{0}.$$

Similarly, proving that $\bar{U}$ has a lower bound can be done by instead proving a strict minimum via the second partial derivative test (as is needed anyway for the stability proof below.) Taking advantage of the fact that the Hessian of a scalar function is also the Jacobian of its gradient, a lower bound exists if

$$\mathbf{J}(\nabla U + \mathbf{f}) \succ 0.$$

Therefore, U^f is never explicitly needed, and in addition, the proof holds independent of one particular choice of a U^f over another.

8.2.2 Stability of Lagrangian Systems with Statically Conservative Forces

Assume now that the loop has been closed for the Lagrangian system with statically conservative forces, and that it is autonomous: there is no input $\mathbf{u}$. The following proposition gives a set of conditions under which the system is (asymptotically, globally) stable.

Proposition 8.2.2. *Stability of Lagrangian Systems with Statically Conservative Forces.* *Consider an unforced system in the form of (8.8), i.e. with* $\mathbf{u} \equiv \mathbf{0}$*,*

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \left(\frac{\partial L}{\partial q_j}\right) = -Q_j(\mathbf{q}, \dot{\mathbf{q}}), \quad j = 1 \ldots n. \tag{8.23}$$

Assume that U^f*,* $\bar{U}$*, and* $\mathbf{S} := \mathbf{Q} - \boldsymbol{\sigma}$ *meet the three conditions of existence, boundedness, and dissipativity from Prop. 8.2.1. If also:*

4. *The augmented potential energy* $\bar{U}(\mathbf{q})$ *has a strict local minimum at coordinates* $\bar{\mathbf{q}}$ *in a set* $\mathcal{Q} \subseteq \mathcal{X} \subseteq \mathbb{R}^n$*, i.e., with both* U *and* U^f *in* $\mathcal{C}^1$*,*

$$\bar{U}\Big|_{\mathbf{q}=\bar{\mathbf{q}}} = c, \qquad \bar{U} > c \quad \forall \mathbf{q} \neq \bar{\mathbf{q}} \in \mathcal{Q}, \tag{8.24}$$

Then $\bar{\mathbf{q}}$ *is an equilibrium point in* $\mathcal{Q}$*, and the system is stable in the sense of Lyapunov around that equilibrium point.*

5. *The inequality for* $\mathbf{S}$ *is strict in the second argument, i.e.,*

$$\dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) > 0 \qquad \forall\, \dot{\mathbf{q}} \neq \mathbf{0}, \quad \forall\, \mathbf{q} \tag{8.25}$$

Then the system is also locally asymptotically stable around $\bar{\mathbf{q}}$ *in* $\mathcal{Q}$*.*

6. *The augmented potential energy* $\bar{U}$ *is radially unbounded (proper),*

Then the system is also asymptotically stable in the sense of Lyapunov in all of $\mathcal{Q}$*, or if* $\mathcal{Q} = \mathbb{R}^n$*, globally asymptotically stable.*

Proof. The proof follows the form of Prop. 3.3.3.

First, establish the equilibrium point(s) of the system (8.23.) By the definition, equilibria exist where $\dot{\mathbf{q}} = \mathbf{0}$. Using the definition of $L = T - U$, as well as the fact that the total potential energy is not a function of the generalized velocities, Lagrange's equations (8.23) at zero velocity become

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_j}\Big|_{\dot{\mathbf{q}}=\mathbf{0}}\right) - \frac{\partial T}{\partial q_j}\Big|_{\dot{\mathbf{q}}=\mathbf{0}} + \frac{\partial U}{\partial q_j} = -Q_j(\mathbf{q}, \mathbf{0}), \qquad j = 1 \ldots n. \tag{8.26}$$

By definition, kinetic energy is zero at zero velocity, so all T terms drop. In vector form,

$$\frac{\partial U}{\partial \mathbf{q}} + \mathbf{Q}(\mathbf{q}, \mathbf{0}) = \mathbf{0}. \tag{8.27}$$

Substituting the definition of the augmented potential energy, where

$$\boldsymbol{\sigma}(\mathbf{q}) := \mathbf{Q}(\mathbf{q}, \mathbf{0}), \qquad \frac{\partial U^f}{\partial \mathbf{q}} = \boldsymbol{\sigma}, \qquad \bar{U} := U + U^f,$$

establishes that any equilibrium point therefore must exist at zero points of the augmented potential energy:

$$\frac{\partial U}{\partial \mathbf{q}} + \frac{\partial U^f}{\partial \mathbf{q}} = \mathbf{0}, \qquad \Rightarrow \qquad \frac{\partial \bar{U}}{\partial \mathbf{q}} = \mathbf{0}. \tag{8.28}$$

By the assumption that $\bar{U}$ is bounded below (and assumptions on continuous differentiability giving extrema of $\bar{U}$), there exist $\mathbf{q}$ that satisfy this property, which are subsequently the equilibrium point(s):

$$\bar{U} \geq c, \ \ \exists\, \bar{\mathbf{q}} \ \text{ s.t. } \ \bar{U}\big|_{\mathbf{q}=\bar{\mathbf{q}}} = c, \qquad \Rightarrow \qquad \exists\, \bar{\mathbf{q}} \ \text{ s.t. } \ \frac{\partial U}{\partial \mathbf{q}}\bigg|_{\mathbf{q}=\bar{\mathbf{q}}} = \mathbf{0}. \tag{8.29}$$

Consider the storage function $V = \bar{H} - c$ as a Lyapunov candidate. First establish that it is a positive definite function. It was shown in the proof to Prop. (8.2.1) that V is nonnegative. In addition, the storage function is zero at the equilibrium point's coordinates:

$$\begin{aligned} V(\bar{\mathbf{q}}, \mathbf{0}) &= \ T\big|_{\substack{\mathbf{q}=\bar{\mathbf{q}} \\ \dot{\mathbf{q}}=\mathbf{0}}} + \bar{U}\big|_{\mathbf{q}=\bar{\mathbf{q}}} - c, \\ V(\bar{\mathbf{q}}, \mathbf{0}) &= c - c = 0. \end{aligned}$$

Given assumption 4, the augmented potential energy $\bar{U}$ has a strict local minimum at $\bar{\mathbf{q}}$ in $\mathcal{Q}$. Therefore,

$$\bar{U} > c \Rightarrow \bar{U} - c > 0 \qquad \forall \mathbf{q} \neq \bar{\mathbf{q}} \in \mathcal{Q}.$$

Since $T \geq 0$ for any argument and $T > 0$ for $\dot{\mathbf{q}} \neq 0$,

$$T + \bar{U} - c > 0 \qquad \forall \mathbf{q} \neq \bar{\mathbf{q}} \in \mathcal{Q}, \ \ \forall \dot{\mathbf{q}} \neq \mathbf{0}$$

$$V(\mathbf{q}, \dot{\mathbf{q}}) > 0 \qquad \forall \mathbf{q} \neq \bar{\mathbf{q}} \in \mathcal{Q}, \ \ \forall \dot{\mathbf{q}} \neq \mathbf{0}.$$

Therefore, $V \succ 0$ is a locally positive definite function around $\{\mathbf{q} = \bar{\mathbf{q}}, \dot{\mathbf{q}} = \mathbf{0}\}$.

Next, consider the time derivative $\dot{V}$ of the Lyapunov candidate. Assuming (again) continuous differentiability of V, it was derived in Prop. 8.2.1 that (with $\bar{\mathbf{u}} = 0$),

$$\dot{V} = -\dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) \leq 0 \qquad \forall \mathbf{q}, \dot{\mathbf{q}} \tag{8.30}$$

Therefore, $\dot{V} \preceq 0$, so V meets all conditions for a Lyapunov function, and the system is stable in the sense of Lyapunov around $\bar{\mathbf{q}}$.

For asymptotic stability, assumption 5 gives strictness of the above inequality in the second argument, and can be written as

$$\dot{\mathbf{q}} \neq \mathbf{0} \ \Rightarrow \ \dot{\mathbf{q}}^\top \mathbf{S} > 0. \tag{8.31}$$

Consider the contrapositive of this statement, which must also be true:

$$\dot{\mathbf{q}}^\top \mathbf{S} \leq 0 \;\Rightarrow\; \dot{\mathbf{q}} = \mathbf{0}. \tag{8.32}$$

Since it is also known that $\dot{\mathbf{q}}^\top \mathbf{S} \geq 0$ always, the left-hand side of (8.32) is reduced to the case when $\dot{\mathbf{q}}^\top \mathbf{S}$ is equal to zero, and

$$\dot{\mathbf{q}}^\top \mathbf{S} = 0 \;\Rightarrow\; \dot{\mathbf{q}} = \mathbf{0}.$$

It is clear also that

$$\dot{\mathbf{q}} = \mathbf{0} \;\Rightarrow\; \dot{\mathbf{q}}^\top \mathbf{S} = 0,$$

then only solution to $\dot{V} = 0$ is

$$\dot{V}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{0} \iff \dot{\mathbf{q}} = \mathbf{0}.$$

LaSalle's Invariance Principle can then be used in the following way. From eqns. (8.27)-(8.29), it was shown that

$$\dot{\mathbf{q}} = \mathbf{0} \quad \Rightarrow \quad \frac{\partial \bar{U}}{\partial \mathbf{q}} = \mathbf{0}.$$

It was also shown that the only solution here is the equilibrium point $\bar{\mathbf{q}}$, and that $\bar{\mathbf{q}}$ is a unique point in $\mathcal{Q}$ via the assumption on a strict local minimum of $\bar{U}$,

$$\frac{\partial \bar{U}}{\partial \mathbf{q}} = 0 \qquad \iff \qquad \mathbf{q} = \bar{\mathbf{q}} \qquad \mathbf{q} \in \mathcal{Q}$$

Therefore,

$$\dot{V}(\mathbf{q}, \dot{\mathbf{q}}) = 0 \iff \{\mathbf{q} = \bar{\mathbf{q}}, \dot{\mathbf{q}} = \mathbf{0}\} \qquad \mathbf{q} \in \mathcal{Q}$$

indicating that the equilibrium point $\bar{\mathbf{q}}$ is the only point in $\mathcal{Q}$ at which $\dot{V} = 0$, and so is a positively invariant set for the zero level set of this $\dot{V}$. This meets the conditions of LaSalle's Invariance Theorem, and the system is asymptotically stable around the point specified by $\bar{\mathbf{q}}$ in $\mathcal{Q}$.

Finally, for global stability, incorporate assumption 6. Then, since

$$V(\mathbf{q}, \dot{\mathbf{q}}) = T + \bar{U} + c,$$

$$\lim_{\mathbf{q}, \dot{\mathbf{q}} \to \infty} V(\mathbf{q}, \dot{\mathbf{q}}) = \lim_{\mathbf{q}, \dot{\mathbf{q}} \to \infty} T + \lim_{\mathbf{q} \to \infty} \bar{U} + c.$$

By definition, the kinetic energy approaches infinity as velocity approaches infinity (for any value of generalized coordinates):

$$\lim_{\dot{\mathbf{q}}\to\infty} T = \infty.$$

The assumption gives radial unboundedness of $\bar{U}$, and since T goes to infinity in the second argument, their sum is then radially unbounded with respect to both arguments:

$$\lim_{\mathbf{q},\dot{\mathbf{q}}\to\infty} T = \infty, \quad \lim_{\mathbf{q}\to\infty} \bar{U} = \infty \quad \Rightarrow \lim_{\mathbf{q},\dot{\mathbf{q}}\to\infty} V(\mathbf{q},\dot{\mathbf{q}}) = \infty.$$

Which meets the condition of radial unboundedness of the Lyapunov candidate. The system is therefore asymptotically stable around $\bar{\mathbf{q}}$ in all of $\mathcal{Q}$, and if $\mathcal{Q} = \mathbb{R}^n$, is globally asymptotically stable.

□

Remark. The use of propositional logic (arguing the contrapositive of the strictness condition on $\dot{\mathbf{q}}^\top\mathbf{S}$) is not necessarily common in proofs such as these, and may seem out of place. However, it provides a much cleaner solution than attempting to investigate the cases when

$$\dot{\mathbf{q}}^\top\mathbf{S}(\mathbf{q},\dot{\mathbf{q}}) = 0,$$

since this is a nonlinear system of equations.

For example, in the cable-driven robot application below, it could be conceptually possible that at some configuration with nonzero velocity, the cables 'cancel out' to give $\dot{V} = 0$ even though $\dot{\mathbf{q}} \neq 0$. Arguing against this case might then need tools from (for example) graph theory to compare behavior of various cables with the strictness assumption. Instead, calling upon the contrapositive above eliminates the need to examine these cases, showing that it's not possible for a nonzero velocity to give $\dot{V} = 0$, giving the 'only if' part of the asymptotic stability proof via LaSalle's Invariance Theorem.

Remark. The example of an inseparable force given at the start of this chapter, that of the slack cable in eqn. (8.4),

$$F = \begin{cases} k\ell + c\dot{\ell}, & \text{if} \quad k\ell + c\dot{\ell} \geq 0 \\ 0, & \text{if} \quad k\ell + c\dot{\ell} < 0, \end{cases},$$

will be shown to be statically conservative. A nonlinear version of this force will also satisfy the dissipation inequality of eqn. (8.11) as well as the strictness condition of eqn. (8.25) given a set of conditions on the scalar spring-like and damper-like components. The controller example at the end of this chapter gives stability of a system with these forces using the above proposition.

8.3 Particles with Damped Central Forces: Model

Modeling a mechanical system as a system of particles is a useful and informative abstraction for control, and often is used as a step toward controlling a system with rigid body dynamics.

Common examples include quadcopters with a suspended payload [125], or cable-driven robots. In this chapter, a cable-driven robot will be considered as an application using this model. The following section establishes a model and some properties related to statically conservative forces for a (system of) particles.

One highly useful framework for which the concept of augmented potential energy can be applied to a system of particles is that of *central forces*. A central force, commonly encountered in the orbital mechanics community, is one where the force is directed along a unit vector between two particles, and only depends on the distance between them (formal definition given below.) A wide variety of commonly encountered forces are central, including gravity, electrostatics, and spring forces.

However, unlike orbital mechanics, for which the energy of a body does not dissipate in any appreciable manner, it is often necessary to model some energy dissipation in a network of cables in order to obtain physical realistic models: an unforced, weighted cable net settles to an equilibrium state. Therefore, a model of central forces *with damping* [52] can be used. Though forces in systems of particles have been studied since the time of Keppler, incorporating damping or dissipation is still an open problem [156, 17]. The work in this chapter provides a framework for stability in these cases.

The following three sections establish the needed properties for an autonomous system, with no input. When applied later in Sec. 8.6, an input will be allowed, a controller will be proposed, and the loop will be closed before performing the needed analysis.

8.3.1 System Model

Single particles

Consider first a single particle of mass m in three-dimensional Euclidean space (Fig. 8.1). Assuming a basis in $\mathbb{E}^3$, the particle's position $\mathbf{r}$ is described by three generalized coordinates,

$$\mathbf{r} = \mathbf{r}(q_1, q_2, q_3),$$

and has a velocity that is a function of the generalized velocities,

$$\mathbf{v} = \dot{\mathbf{r}} = \dot{\mathbf{r}}(\dot{\mathbf{q}}).$$

As an example, in the Cartesian basis,

$$\mathbf{q} = [x_1, x_2, x_3]^\top, \quad \mathbf{r} = x_1\mathbf{E}_1 + x_2\mathbf{E}_2 + x_3\mathbf{E}_3, \quad \mathbf{v} = \dot{x}_1\mathbf{E}_1 + \dot{x}_2\mathbf{E}_2 + \dot{x}_3\mathbf{E}_3.$$

For simplicity, it will be assumed that $\mathbf{v}$ is not a function of the generalized coordinates $\mathbf{q}$, though this does not hold for some choices of basis (e.g., polar coordinates.) This assumption will be relaxed in future work.

Next, assume there are a set of $i = 1 \dots s$ other points fixed in space, $\mathbf{b}_i \in \mathbb{E}^3$, such as anchor points for cables, from which forces will act on the particle. The vector between the particle and one of these points - again, with cables as a motivation - is

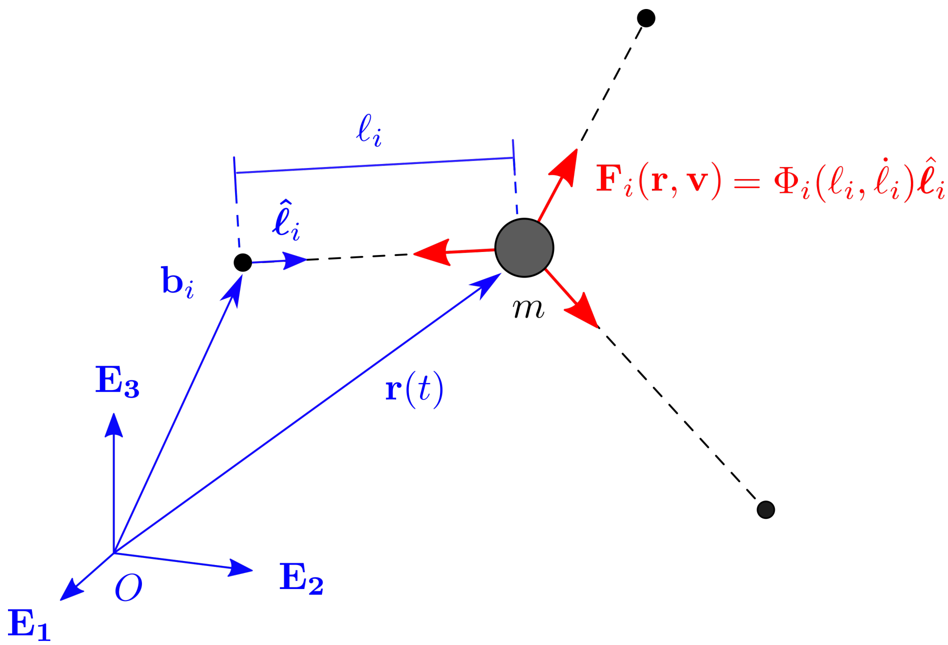


Figure 8.1: A particle of mass m with forces directed from fixed points. Each force $i = 1 \ldots s$ has an anchor point $\mathbf{b}_i$ fixed in space, a total (scalar) length between particle and anchor ℓ_i, and a unit vector $\hat{\boldsymbol{\ell}}_i$. A central force with damping has the form $\mathbf{F}_i = \Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i$, where the function $\Phi_i(\cdot,\cdot) : \mathbb{R}^2 \mapsto \mathbb{R}$ describes the scalar applied force. The system-of-particles model allows $\mathbf{b}_i$ to be other particles $\mathbf{r}_j$.

$$\boldsymbol{\ell}_i = \mathbf{r} - \mathbf{b}_i,$$

which has a scalar distance ('length' of the cable) as ℓ_i, with the unit vector from anchor to point mass as $\hat{\boldsymbol{\ell}}_i$,

$$\ell_i = ||\boldsymbol{\ell}_i||, \qquad \hat{\boldsymbol{\ell}}_i = \frac{1}{\ell_i}\boldsymbol{\ell}_i,$$

where $||\cdot||$ is the Euclidean norm. As the particle moves in time, these distances change (cables extend and retract.) The *length change*, $\dot{\ell}$, will be defined as the rate of change of ℓ,

$$\dot{\ell}_i = \frac{d}{dt}||\boldsymbol{\ell}_i||.$$

A quick calculation expresses $\dot{\ell}_i$ in terms of $\mathbf{r}$ and $\mathbf{v}$ (implicitly, $\mathbf{q}$ and $\dot{\mathbf{q}}$.) The time derivative of the direction vector is

$$\frac{d}{dt}\boldsymbol{\ell}_i = \dot{\boldsymbol{\ell}}_i = \frac{d}{dt}(\mathbf{r} - \mathbf{b}_i) = \mathbf{v},$$

noting that $\dot{\mathbf{b}}_i = 0$. By the full derivative of a norm, the time derivative of the length is

$$\dot{\ell}_i = \frac{d}{dt}||\boldsymbol{\ell}_i|| = \frac{\dot{\boldsymbol{\ell}}_i \cdot \boldsymbol{\ell}_i}{||\boldsymbol{\ell}_i||} = \frac{\mathbf{v} \cdot \boldsymbol{\ell}_i}{||\boldsymbol{\ell}_i||} = \mathbf{v} \cdot \hat{\boldsymbol{\ell}}_i.$$

Therefore, any length and length change are expressed as functions of the particle's position $\mathbf{r}$ and velocity $\mathbf{v}$,

$$\hat{\boldsymbol{\ell}}_i(\mathbf{r}), \quad \ell_i(\mathbf{r}), \quad \dot{\ell}_i(\mathbf{r}, \mathbf{v}).$$

Consider a force applied on the particle, directed toward an anchor point (Fig. 8.1.) The following definitions describe such forces.

Definition 8.3.1. Central Force. A force applied to the particle is a *central force* if it is directed along the vector from one point to that particle, and is only a function of the distance from that point:

$$\mathbf{F}_i(\mathbf{r}) := \phi_i(\ell_i)\hat{\boldsymbol{\ell}}_i, \tag{8.33}$$

where $\phi_i(\cdot) : \mathbb{R} \mapsto \mathbb{R}$ is the *scalar force*.

Definition 8.3.2. Central Force With Damping. A force applied to the particle is a *central force with damping* if it is central, but also is a function of the rate of change of the distance between anchor to particle:

$$\mathbf{F}_i(\mathbf{r}, \mathbf{v}) := \Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i, \tag{8.34}$$

where $\Phi_i(\cdot, \cdot) : \mathbb{R}^2 \mapsto \mathbb{R}$ will also be called the *scalar force*.

The term 'scalar force' arises since $||\hat{\boldsymbol{\ell}}|| = 1 \Rightarrow ||\mathbf{F}|| = \phi$ or Φ respectively.

If considered as a particle in a network of cables, central forces with damping capture the behavior of a wide variety of idealized, massless models. For example, the traditional linear spring-damper system arises with

$$\Phi_i(\ell_i, \dot{\ell}_i) = k\ell_i + c\dot{\ell}_i.$$

A fact that will be used in the passivity and stability proofs later is the result of the inner product of $\mathbf{F}$ with $\mathbf{v}$: the work done by the force on the particle. For those proofs, the following calculation will occur:

$$\mathbf{F}_i \cdot \mathbf{v} = (\Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i) \cdot \mathbf{v} = \Phi_i(\ell_i, \dot{\ell}_i)(\mathbf{v} \cdot \hat{\boldsymbol{\ell}}_i) = \Phi_i(\ell_i, \dot{\ell}_i)\dot{\ell}_i, \tag{8.35}$$

by the definition of length change. Intuitively, then, the work done by a cable will be its scalar force Φ_i multiplied by its length change, a somewhat remarkable result that will allow treating each cable independently when analyzing the system's energy exchange.

Systems of particles

Extending the model above to a system of K particles can be done by allowing some anchors $\mathbf{b}_i$ to also be point masses in $\mathbb{E}^3$, with position vector $\mathbf{r}_k$ and mass m_k. The length vector and its time derivative associated with force i directed from particle k to particle j are

$$\boldsymbol{\ell}_i = \mathbf{r}_j - \mathbf{r}_k, \qquad \dot{\boldsymbol{\ell}}_i = \mathbf{v}_j - \mathbf{v}_k,$$

and its scalar length change is

$$\dot{\ell}_i = \dot{\boldsymbol{\ell}}_i \cdot \frac{\boldsymbol{\ell}_i}{||\boldsymbol{\ell}_i||} = (\mathbf{v}_j - \mathbf{v}_k) \cdot \hat{\boldsymbol{\ell}}_i.$$

All other equations are the same as for the single particle case.

As with the single particle, it will be useful to note that the total work of this force on both particles becomes the product of its scalar force and the cable's velocity. If cable i is defined along a length vector $\boldsymbol{\ell}_i = \mathbf{r}_j - \mathbf{r}_k$, then its applied force on particle j and k will be designated as $\mathbf{F}_i^j$ and $\mathbf{F}_i^k$. Note then that

$$\mathbf{F}_i^j = \Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i, \qquad \mathbf{F}_i^k = -\Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i,$$

since the length vector points "toward" particle j and "away" from particle k. The work done by this force on the respective particles is

$$\mathbf{F}_i^j \cdot \mathbf{v}_j = \Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i \cdot \mathbf{v}_j \tag{8.36}$$

$$\mathbf{F}_i^k \cdot \mathbf{v}_k = -\Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i \cdot \mathbf{v}_k. \tag{8.37}$$

The total work is the sum, $\mathbf{F}_i^j \cdot \mathbf{v}_j + \mathbf{F}_i^k \cdot \mathbf{v}_k$, so

$$\mathbf{F}_i^j \cdot \mathbf{v}_j + \mathbf{F}_i^k \cdot \mathbf{v}_k = \Phi_i(\ell_i, \dot{\ell}_i)(\hat{\boldsymbol{\ell}}_i \cdot \mathbf{v}_j - \hat{\boldsymbol{\ell}}_i \cdot \mathbf{v}_k) \tag{8.38}$$

$$\mathbf{F}_i^j \cdot \mathbf{v}_j + \mathbf{F}_i^k \cdot \mathbf{v}_k = \Phi_i(\ell_i, \dot{\ell}_i)\dot{\ell}_i, \tag{8.39}$$

and again, analyzing the system's change in energy can be done per-cable (or per-force) as opposed to per-particle.

8.3.2 Central Forces with Damping are Statically Conservative

In addition to the observation that networks of cables inherently consist of central forces with damping, examining systems with these forces is also motivated by the following. In all cases, central forces with damping are statically conservative. To show this, it is first shown here that central forces are always conservative. This is a textbook result that bears briefly repeating.

Here, recall from Chap. 3 that the definition of conservation in $\mathbb{E}^3$ is equivalent to the component-wise definition using generalized forces, under the required component-wise projection from $\mathbf{F}$ to $\mathbf{Q}$ (see Sec. 3.1.4) for all particles k,

$$\mathbf{Q} = \frac{\partial U}{\partial \mathbf{q}} \quad \iff \quad \mathbf{F}^k = \frac{\partial U}{\partial \mathbf{r}_k} \;\; \forall\, k.$$

For the remainder of this chapter, all forces will be worked with in $\mathbb{E}^3$ with the understanding that Prop. 8.2.1 and 8.2.2 are satisfied via this equivalence.

Claim 8.3.1. *Central forces acting on a single particle,*

$$\mathbf{F}_i(\mathbf{r}) = \phi_i(\ell_i)\hat{\boldsymbol{\ell}}_i, \qquad \textit{where} \quad \boldsymbol{\ell}_i(\mathbf{r}) = \mathbf{r} - \mathbf{b}_i,$$

are conservative, i.e.,

$$\exists\, U_i(\mathbf{r}) \quad s.t. \quad \nabla_{\mathbf{r}} U_i(\mathbf{r}) = \mathbf{F}_i(\mathbf{r}).$$

Proof. The following is a textbook proof [142]. Propose the following potential energy function, where a is a constant,

$$U_i(\mathbf{r}) := \int_a^{\ell_i} \phi_i(\tau) d\tau.$$

Differentiating,

$$\nabla_{\mathbf{r}} U_i(\mathbf{r}) = \frac{\partial U_i}{\partial \ell_i}\frac{\partial \ell_i}{\partial \mathbf{r}} = \frac{\partial U_i}{\partial \ell_i}\hat{\boldsymbol{\ell}}_i,$$

since as described in Sec. 8.3.1,

$$\frac{\partial \ell_i}{\partial \mathbf{r}} = \frac{\partial}{\partial \mathbf{r}}||\boldsymbol{\ell}_i|| = \frac{\boldsymbol{\ell}_i}{||\boldsymbol{\ell}_i||} = \hat{\boldsymbol{\ell}}_i.$$

The partial derivative of U_i recovers the scalar force,

$$\frac{\partial U_i}{\partial \ell_i} = \frac{\partial}{\partial \ell_i}\int_a^{\ell_i} \phi_i(\tau) d\tau = \phi_i(\ell_i),$$

giving the central force as the gradient of the proposed potential energy function:

$$\nabla_{\mathbf{r}} U_i(\mathbf{r}) = \phi_i(\ell_i)\hat{\boldsymbol{\ell}}_i = \mathbf{F}_i.$$

□

The following extension to systems of particles is equally well-known. For systems of particles, one potential function is used for all particles; there is no separate per-particle potential. Instead, the sum below is over cables i.

Claim 8.3.2. *Consider a set of central forces $i = 1 \ldots s$ acting on a system of K particles, where force i acts between particles j and k with $\boldsymbol{\ell}_i = \mathbf{r}_k - \mathbf{r}_j$, as in*

$$\mathbf{F}_i^k(\mathbf{r}_k, \mathbf{r}_j) = \phi_i(\ell_i)\hat{\boldsymbol{\ell}}_i, \qquad \mathbf{F}_i^j(\mathbf{r}_k, \mathbf{r}_j) = -\phi_i(\ell_i)\hat{\boldsymbol{\ell}}_i.$$

Let particle k have $u = 1...m$ of these forces directed towards it, and $v = 1...p$ directed away, such that the total force on that particle is

$$\mathbf{F}^k = \sum_{u=1}^{m} \mathbf{F}_u^k + \sum_{v=1}^{p} \mathbf{F}_v^k \tag{8.40}$$

$$= \sum_{u=1}^{m} \phi_u^k(\ell_u)\hat{\boldsymbol{\ell}}_u - \sum_{v=1}^{p} \phi_v^k(\ell_v)\hat{\boldsymbol{\ell}}_v, \tag{8.41}$$

where $\phi_{(\cdot)}^k$ represents the appropriate scalar force acting on particle k. These central forces are conservative, i.e., the total force on particle k is the gradient of a scalar potential function, for all particles,

$$\exists\, U_i(\mathbf{r}_1, \ldots, \mathbf{r}_K), \quad U = \sum_{i=1}^{s} U_i \qquad \text{s.t.} \qquad \nabla_{\mathbf{r}_k} U = \mathbf{F}^k, \qquad \forall\, k = 1 \ldots K.$$

Proof. Propose the same potential energy function as with Claim 8.3.1, now recognizing that the lengths are functions of multiple particles' positions,

$$U_i(\mathbf{r}_1, \ldots, \mathbf{r}_K) := \int_a^{\ell_i} \phi_i(\tau)d\tau, \qquad U = \sum_{i=1}^{s} U_i.$$

Consider if $\boldsymbol{\ell}_u = \mathbf{r}_k - \mathbf{r}_j$. Then,

$$\nabla_{\mathbf{r}_k} U_u(\mathbf{r}_1, \ldots, \mathbf{r}_K) = \frac{\partial U_u}{\partial \ell_u}\frac{\partial \ell_u}{\partial \mathbf{r}_k} = \frac{\partial U_u}{\partial \ell_u}\hat{\boldsymbol{\ell}}_u \tag{8.42}$$

$$= \phi_u^k(\ell_u)\hat{\boldsymbol{\ell}}_u \tag{8.43}$$

$$= \mathbf{F}_u^k. \tag{8.44}$$

Consider instead if $\boldsymbol{\ell}_v = \mathbf{r}_j - \mathbf{r}_k$. Then,

$$\nabla_{\mathbf{r}_k} U_v(\mathbf{r}_1, \ldots, \mathbf{r}_K) = \frac{\partial U_v}{\partial \ell_v}\frac{\partial \ell_v}{\partial \mathbf{r}_k} = -\frac{\partial U_v}{\partial \ell_v}\hat{\boldsymbol{\ell}}_v \tag{8.45}$$

$$= -\phi_v^k(\ell_v)\hat{\boldsymbol{\ell}}_v \tag{8.46}$$

$$= \mathbf{F}_v^k. \tag{8.47}$$

The gradient of a cable's the potential function U_i with respect to particle position $\mathbf{r}_k$ will only be nonzero for u or v, so

$$\begin{aligned}\nabla_{\mathbf{r}_k} U &= \sum_{i=1}^{s} \nabla_{\mathbf{r}_k} U_i && (8.48)\\ &= \sum_{u=1}^{m} \nabla_{\mathbf{r}_k} U_u + \sum_{v=1}^{p} \nabla_{\mathbf{r}_k} U_v && (8.49)\\ &= \sum_{u=1}^{m} \mathbf{F}_u^k + \sum_{v=1}^{p} \mathbf{F}_v^k && (8.50)\\ &= \mathbf{F}^k. && (8.51)\end{aligned}$$

The argument can be repeated for all $k = 1 \ldots K$ particles.

□

Given these two proofs, the following important result can be established: central forces with damping are statically conservative. For systems of particles that are only under the action of either conservative forces, or central forces with damping, the augmented potential energy always exists in the form needed for Prop. 8.2.1 and 8.2.2.

To do so, a formal definition of the statically conservative component of a force in $\mathbb{E}^3$ will be useful. Though this is potentially one of many decompositions of a force $\mathbf{F}$ into a conservative and nonconservative component, it will be the only one considered here.

Definition 8.3.3. Statically conservative component of a force. Given a force $\mathbf{F}$ that is a function of a particle's (or many particles') position and velocity, $\mathbf{F}(\mathbf{r}, \mathbf{v})$, its *statically conservative component* is defined as

$$\mathbf{f}(\mathbf{r}) := \mathbf{F}(\mathbf{r}, \mathbf{0}), \tag{8.52}$$

if $\mathbf{f}$ is conservative.

In the remainder of this chapter, the statically conservative component of a central force with damping will use the same notation for its scalar force,

$$\phi_i(\ell_i) := \Phi_i(\ell_i, 0),$$

as for central forces.

Claim 8.3.3. *Central forces with damping acting on a single particle,*

$$\mathbf{F}_i(\mathbf{r}, \mathbf{v}) = \Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i, \qquad \textit{where} \quad \boldsymbol{\ell}_i(\mathbf{r}) = \mathbf{r} - \mathbf{b}_i,$$

are statically conservative, i.e.,

$$\exists\, U_i^f(\mathbf{r}) \quad s.t. \quad \nabla_{\mathbf{r}} U_i^f(\mathbf{r}) = \mathbf{f}_i(\mathbf{r}),$$

where

$$\mathbf{f}_i(\mathbf{r}) := \mathbf{F}_i(\mathbf{r}, \mathbf{0}) = \Phi_i(\ell_i, 0)\hat{\boldsymbol{\ell}}_i \tag{8.53}$$

$$:= \phi_i(\ell_i)\hat{\boldsymbol{\ell}}_i \tag{8.54}$$

is the force's statically conservative component.

Proof. Propose the potential function

$$U_i^f(\mathbf{r}) := \int_a^{\ell_i} \phi_i(\tau) d\tau.$$

The remainder follows from the same derivation as Claim 8.3.1.

□

Claim 8.3.4. *Consider a set of central forces with damping $i = 1 \dots s$ acting on a system of K particles, where force i acts between particles j and k with $\boldsymbol{\ell}_i = \mathbf{r}_k - \mathbf{r}_j$, as in*

$$\mathbf{F}_i^k(\mathbf{r}_k, \mathbf{r}_j, \mathbf{v}_k, \mathbf{v}_j) = \Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i, \qquad \mathbf{F}_i^j(\mathbf{r}_k, \mathbf{r}_j, \mathbf{v}_k, \mathbf{v}_j) = -\Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i.$$

Let particle k have $u = 1...m$ of these forces directed towards it, and $v = 1...p$ directed away, such that the total force on that particle is

$$\mathbf{F}^k = \sum_{u=1}^{m} \mathbf{F}_u^k + \sum_{v=1}^{p} \mathbf{F}_v^k \tag{8.55}$$

$$= \sum_{u=1}^{m} \Phi_u^k(\ell_u, \dot{\ell}_u)\hat{\boldsymbol{\ell}}_u - \sum_{v=1}^{p} \Phi_v^k(\ell_v, \dot{\ell}_v)\hat{\boldsymbol{\ell}}_v. \tag{8.56}$$

These central forces with damping are statically conservative, i.e., the sum of all statically conservative force components on particle k is the gradient of a scalar potential function, for all particles,

$$\exists\, U_i^f(\mathbf{r}_1, \dots, \mathbf{r}_K), \quad U^f = \sum_{i=1}^{s} U_i^f \qquad \text{s.t.} \qquad \nabla_{\mathbf{r}_k} U^f = \mathbf{f}^k, \qquad \forall\, k = 1 \dots K,$$

where $\mathbf{f}_i(\mathbf{r}_k, \mathbf{r}_j) := \mathbf{F}_i(\mathbf{r}_k, \mathbf{r}_j, \mathbf{0}, \mathbf{0})$ is the statically conservative component of each force just as in Claim 8.3.3.

Proof. Propose the same potential function as in Claim 8.3.2,

$$U_i^f(\mathbf{r}_1, \ldots, \mathbf{r}_K) := \int_a^{\ell_i} \phi_i(\tau) d\tau, \qquad U^f = \sum_{i=1}^{s} U_i^f.$$

The remainder follows from the same derivation as Claim 8.3.2.

□

Finally, the following lemma proves that the first condition for the passivity and stability proofs, that of existence of the augmented potential, holds for all systems of particles considered in the model above.

Lemma 8.3.0.1. *Existence of the augmented potential energy for systems of particles under the action of central forces with damping.*

Consider a system of particles. If these particles are only subject to either conservative forces, or central forces with damping, the augmented potential energy always exists for the system such that the system's equilibria are at the extrema of the augmented potential energy.

Proof. Let all conservative forces be subsumed into the potential energy U. Via Claim 8.3.4, there exists a U^f for the set of all central forces with damping. Therefore,

$$\bar{U} = U + U^f,$$

with all forces accounted for and no remaining nonzero forces at $\bar{\mathbf{q}}$.

□

This lemma can be interpreted, informally, that the passivity and stability approach in this chapter can be applied to general networks under the influence of gravity, no matter the possible couplings between generalized coordinates and generalized velocities.

8.3.3 Equations of Motion for the System of Particles

The remainder of this chapter will consider the common case of Lemma 8.3.0.1, where the system of particles is only under the action of gravity and central forces with damping. This model will be used for the cable-driven robot in later sections. Lagrange's equations for this system are briefly repeated here for future reference.

For the system of K particles, with the force from cable i on particle k as $\mathbf{F}_i^k$, with the gravitational force as $-m_k g \mathbf{E}_3$, the Lagrangian is

$$T = \frac{1}{2} \sum_{k=1}^{K} m_k \mathbf{v}_k \cdot \mathbf{v}_k, \qquad U = \sum_{k=1}^{K} m_k g \mathbf{r}_k \cdot \mathbf{E}_3, \qquad L = T - U,$$

and Lagrange's equations are

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \left(\frac{\partial L}{\partial q_j}\right) = -\sum_{k=1}^{K}\sum_{i=1}^{s}\mathbf{F}_i^k \cdot \frac{\partial \mathbf{r}_k}{\partial q_j} \qquad j = 1 \dots 3K, \tag{8.57}$$

where $\mathbf{F}_i^k$ is taken to be $\mathbf{F}_i^k = \mathbf{0}$ if cable i does not act on particle k. Equivalently, the sum over i could be expressed as eqns. (8.55)-(8.56). Note that a negative sign convention is used here to indicate that tension forces (away from the particle) are positive.

To analyze these equations in the vector form needed for Prop. 8.2.1 and 8.2.2, note that the right-hand side of eqn. (8.57) are the generalized forces in the direction of each generalized coordinate,

$$Q_j = \sum_{k=1}^{K}\sum_{i=1}^{s}\mathbf{F}_i^k \cdot \frac{\partial \mathbf{r}_k}{\partial q_j}.$$

So, although $\mathbf{F}_i^k$ are forces in $\mathbb{E}^3$, the Q_j can be stacked into a vector $\mathbf{Q} \in \mathbb{R}^{3K}$ to recover the basic form of Lagrange's equations as considered above,

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \left(\frac{\partial L}{\partial \mathbf{q}}\right) = -\mathbf{Q}. \tag{8.58}$$

The control systems discussed in this chapter do not require that eqn. (8.58) is solved, or even fully expressed: in particular, the time derivative of $\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right)$ does not need to be calculated. Although it would be trivial to do so for the system of particles, it is not done here, in order to highlight this surprising consequence of the control scheme: no full model is needed. It is only required that Lagrange's Equations can be posed, since the augmented potential energy and the force difference $\mathbf{S}$ are all that is needed for analyzing the stability of an equilibrium point.

8.4 Particles with Damped Central Forces: Energy Analysis

For the system of particles under the influence of both gravity and central forces with damping, the system's augmented potential energy can be written and analyzed for its minimum points. This constitutes an implicit part of using the propositions above, and additionally, addresses one of the motivations for using this approach over the framework of equilibrium-independent dissipativity (EID) [86, 125]. Here, finding the system's equilibrium point is a solved problem; in particular, the inverse statics approach from Chap. 5 does so.

In addition, conditions on the convexity and strict minimum of $\bar{U}$ can be posed in terms of the cables' scalar forces. This section therefore addresses the first two of the three needed conditions for Prop. 8.2.1 as well as the strictness needed for Prop. 8.2.2.

8.4.1 Augmented Potential and Equilibrium for Particles with Statically Conservative Forces

For the system (8.57), Lemma 8.3.0.1 gives the total augmented potential energy as the sum of gravitational potential and the cables' statically conservative potentials,

$$\bar{U} = U + \sum_{i=1}^{s} U_i^f = \sum_{k=1}^{K} mg\mathbf{r}_k \cdot \mathbf{E}_3 + \sum_{i=1}^{s} U_i^f, \tag{8.59}$$

where it is relevant to note again that one sum is over particles, and the other over cables.

Recall from Chap. 3 that the relationship between the partial derivatives with respect to the coordinates $\mathbf{q}$ versus the partial derivatives with respect to particles' positions $\mathbf{r}_k$, $k = 1 \dots K$, is given by eqn. (3.12), repeated here with $\bar{U}$:

$$\frac{\partial \bar{U}}{\partial \mathbf{q}} = \begin{bmatrix} \displaystyle\sum_{k=1}^{K} \nabla_{\mathbf{r}_k} \bar{U} \cdot \frac{\partial \mathbf{r}_k}{\partial q_1} \\ \vdots \\ \displaystyle\sum_{k=1}^{K} \nabla_{\mathbf{r}_k} \bar{U} \cdot \frac{\partial \mathbf{r}_k}{\partial q_{3K}} \end{bmatrix} \in \mathbb{R}^{3K}. \tag{8.60}$$

This identity gave the result in eqn. (3.57) from Sec. 3.3.4 that the equilibrium point is located at the zero of the partial derivatives of U with respect to $\mathbf{q}$ or respect to $\mathbf{r}$ equivalently. Substituting the augmented potential energy now gives the same result,

$$\left.\frac{\partial \bar{U}}{\partial \mathbf{q}}\right|_{\mathbf{q}=\bar{\mathbf{q}}} = \mathbf{0} \quad \iff \quad \nabla_{\mathbf{r}_k} \bar{U}(\mathbf{r}_1(\bar{\mathbf{q}}), \dots, \mathbf{r}_K(\bar{\mathbf{q}})) = \mathbf{0} \;\; \forall\, k. \tag{8.61}$$

In the case of separable forces, the gradients with respect to each particle simply gave the conservative forces in static equilibrium. Now, substituting for the definition of the augmented potential energy instead,

$$\begin{aligned} \nabla_{\mathbf{r}_k} \bar{U} = \mathbf{0} \quad &\iff \quad \nabla_{\mathbf{r}_k} U + \nabla_{\mathbf{r}_k} U^f = \mathbf{0} && (8.62) \\ &\iff \quad \nabla_{\mathbf{r}_k} U + \mathbf{f}^k(\bar{\mathbf{r}}) = \mathbf{0} && (8.63) \\ &\iff \quad \nabla_{\mathbf{r}_k} U = -\mathbf{F}^k(\bar{\mathbf{r}}, \mathbf{0}) \qquad \forall\, k. && (8.64) \\ & && (8.65) \end{aligned}$$

This derivation demonstrates the following fact:

If all forces in the system of particles are conservative or statically conservative, a configuration $\{\bar{\mathbf{r}}_k, \mathbf{0}\}, k = 1 \dots K$, is an equilibrium point of the system if and only if gradient of the augmented potential energy equals zero at the $\bar{\mathbf{q}}$ defining those points; equivalently, if *and only if the static Newtonian force balance holds at that point.*

In all uses of Prop. 8.2.2, then, the equilibrium point does not have to be found via an analysis of the augmented potential energy. Instead, any other method may be used to find $\bar{\mathbf{q}}$ or equivalently $\bar{\mathbf{r}}_1 \dots \bar{\mathbf{r}}_K$, particularly those based on a static Newtonian force balance. Consider then the system of particles with gravity and central forces with damping, using the convention of the cable direction indexing from Claim 8.3.4,

$$\nabla_{\mathbf{r}_k} U = -\mathbf{f}^k(\bar{\mathbf{r}}_k) \qquad \forall\ k,$$

$$-mg\mathbf{E}_3 = \sum_{u=1}^{m} \phi_u^k(\ell_u)\hat{\boldsymbol{\ell}}_u - \sum_{v=1}^{p} \phi_v^k(\ell_v)\hat{\boldsymbol{\ell}}_v \qquad \forall\ k. \tag{8.66}$$

Eqn. (8.66) exactly describes the force balance considered in the inverse statics problem of Chap. 5, where cable forces counteract gravity. The directedness of the graph structure of the network, via the connectivity matrix in eqn. (5.1), is visible in the cables' forces' directions in the right-hand side of eqn. (8.66).

It is therefore assumed in the remainder of this chapter that $\bar{\mathbf{q}}$ is known, since the problem of finding cables forces that satisfy eqn. (8.66) has been previously addressed in this dissertation. Such an assumption does imply that $\phi_i(\ell_i)$ will be modified by a controller to produce the desired point, as will be performed when a control input is introduced in later sections. This is the approach used in all later examples and simulations, and will be implied for the remainder of this chapter.

Using such an approach does however induce additional required assumptions on ϕ_i, the cables' scalar forces. Since the force density method (inverse statics optimization) calculates equilibrium cable forces, finding a corresponding input (implied to be present in ϕ_i) requires that ϕ_i be surjective: given a found $\bar{\phi}_i$, there must exist a $\bar{\mathbf{q}}$. For uniqueness, needed for stability, ϕ_i must also be invertible (with respect to whatever input.) The linear elastic case satisfies this assumption via eqn. (5.10), and so when the slack-cable model is pretensioned at its equilibrium point, can be used to obtain $\bar{\mathbf{q}}$.

8.4.2 Minima of the Augmented Potential for Networks of Central Forces With Damping

To apply Prop. 8.2.2, it must be shown that the augmented potential $\bar{U}$ has a (strict) minimum with respect to $\mathbf{r}_1 \dots \mathbf{r}_K$ at the system's equilibrium configuration. Fortunately, a set of straightforward conditions on the cables' statically conservative scalar forces $\phi_i(\ell_i)$ give this property. The following section presents conditions using the second partial derivative test for both a single particle and a system of particles, then gives a lemma for when there is a (strict) minimum at a given configuration.

The following formal definition will be used throughout this section. Though it is used in the context of cables providing forces between particles, the same concept applies to any central forces with damping.

Definition 8.4.1. Pretension.
A cable i whose force is represented by $\mathbf{F}_i = \Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i$ is *pretensioned* at a point $\mathbf{q} \iff \{\mathbf{r}_1, \ldots \mathbf{r}_K\}$ if its statically conservative scalar force is positive, i.e.

$$\Phi_i(\ell_i, 0) > 0, \qquad \text{equivalently} \qquad \phi_i(\ell_i) > 0.$$

The term 'pretension' is abundant in the literature, and this definition makes formal the concept of a positive force at zero velocity. It is obvious, but worth noting, that tension (positive force) and pretension are not the same: a cable that is pretensioned may have a positive statically conservative scalar force but a negative scalar force, i.e.

$$\phi_i(\ell_i) > 0, \qquad \Phi_i(\ell_i, \dot{\ell}_i) < 0$$

This could be caused by some length change $\dot{\ell}_i$ which is not part of the cable's conservative potential U_i^f.

Eigenvalues of the Hessians of central forces with damping

Checking the convexity of a cable's U_i^f requires calculating its Hessian. Calculating the eigenvalues of this Hessian has been done to some extent in the past (e.g., [125]), but is made more general here for a system of K particles.

The following derivative will be needed for the calculations. For convenience, the remainder of this section temporarily assumes that $\boldsymbol{\ell}_i$ is expressed in a Cartesian basis so that matrix calculus can be used. Future work will derive this same result using tensor calculus to allow for other bases in $\mathbb{E}^3$.

For a cable with direction vector $\boldsymbol{\ell}_i = \mathbf{r}_k - \mathbf{r}_j$,

$$\frac{\partial \hat{\boldsymbol{\ell}}_i}{\partial \mathbf{r}_k} = \frac{\partial}{\partial \mathbf{r}_k}\left(\frac{\boldsymbol{\ell}_i}{||\boldsymbol{\ell}_i||}\right) = \frac{1}{\ell_i}(\mathbf{I} - \hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top) \tag{8.67}$$

$$\frac{\partial \hat{\boldsymbol{\ell}}_i}{\partial \mathbf{r}_j} = \frac{\partial}{\partial \mathbf{r}_j}\left(\frac{\boldsymbol{\ell}_i}{||\boldsymbol{\ell}_i||}\right) = -\frac{1}{\ell_i}(\mathbf{I} - \hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top). \tag{8.68}$$

The eigenvalues of the Hessian can now be calculated in both the single-particle and multi-particle cases.

Lemma 8.4.0.1. *Eigenvalues of the Hessian of a single cable's statically conservative potential, for a single particle.*

Consider the single-particle system. A cable that applies a central force with damping $\mathbf{F}_i(\mathbf{r},\mathbf{v})$ *has an associated statically conservative scalar force* $\mathbf{f}_i(\mathbf{r})$ *and conservative potential* $U_i^f(\mathbf{r})$ *where as defined above,*

$$\mathbf{f}_i(\mathbf{r}) := \mathbf{F}_i(\mathbf{r},\mathbf{0}), \qquad \nabla U_i^f(\mathbf{r}) = \mathbf{f}_i(\mathbf{r}) = \phi_i(\ell_i)\hat{\boldsymbol{\ell}}_i.$$

The eigenvalues of the Hessian of U_i^f*, also those of the Jacobian of* $\mathbf{f}_i$*, are*

$$\sigma(\mathbf{H}(U_i^f)) = \sigma(\mathbf{J}(\mathbf{f}_i)) = \left\{\frac{\partial\phi_i(\ell_i)}{\partial\ell_i}, \frac{\phi_i(\ell_i)}{\ell_i}, \frac{\phi_i(\ell_i)}{\ell_i}\right\}. \tag{8.69}$$

Proof. The Hessian of a scalar function is also the Jacobian (transposed) of its gradient. Via the product rule, where the argument to the function $\phi_i(\ell_i)$ will be dropped to become ϕ_i for clarity,

$$\mathbf{J}(\mathbf{f}_i(\mathbf{r})) = \phi_i\frac{\partial\hat{\boldsymbol{\ell}}_i}{\partial\mathbf{r}} + \frac{\partial\phi_i}{\partial\mathbf{r}}\hat{\boldsymbol{\ell}}_i. \tag{8.70}$$

Using the chain rule,

$$\frac{\partial\phi_i}{\partial\mathbf{r}} = \frac{\partial\phi_i}{\partial\ell_i}\frac{\partial\ell_i}{\partial\mathbf{r}} = \frac{\partial\phi_i}{\partial\ell_i}\hat{\boldsymbol{\ell}}_i^\top,$$

and substituting eqn. (8.67) since $\boldsymbol{\ell}_i = \mathbf{r} - \mathbf{b}_i$ where $\mathbf{b}_i$ is the (constant) anchor point,

$$\mathbf{J}(\mathbf{f}_i) = \frac{\phi_i}{\ell_i}\left(\mathbf{I} - \hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top\right) + \frac{\partial\phi_i}{\partial\ell_i}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top \tag{8.71}$$

$$= \frac{\phi_i}{\ell_i}\mathbf{I} - \frac{\phi_i}{\ell_i}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top + \frac{\partial\phi_i}{\partial\ell_i}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top \tag{8.72}$$

$$= \frac{\phi_i}{\ell_i}\mathbf{I} + \left(\frac{\partial\phi_i}{\partial\ell_i} - \frac{\phi_i}{\ell_i}\right)\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top. \tag{8.73}$$

Since the matrix $\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top$ is a dyad, it has rank 1, and since $||\hat{\boldsymbol{\ell}}|| = 1$, it has eigenvalues

$$\sigma(\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top) = \{||\hat{\boldsymbol{\ell}}||, 0, 0\} = \{1, 0, 0\},$$

therefore

$$\sigma\left(\left(\frac{\partial\phi_i}{\partial\ell_i} - \frac{\phi_i}{\ell_i}\right)\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top\right) = \left\{\left(\frac{\partial\phi_i}{\partial\ell_i} - \frac{\phi_i}{\ell_i}\right), 0, 0\right\}.$$

A matrix $b\mathbf{I}$ has the property that, when added to another matrix $\mathbf{A}$, as in $\mathbf{A} + b\mathbf{I}$, the eigenvalues of the sum are the eigenvalues of $\mathbf{A}$ added to the constant b. This is shown since all nonzero vectors are eigenvectors of $b\mathbf{I}$,

$$b\mathbf{I}\mathbf{v} = b\mathbf{v} \quad \Rightarrow \quad b\mathbf{v} = b\mathbf{v} \ \ \forall \mathbf{v} \neq \mathbf{0}, \qquad \therefore (\mathbf{A} + b\mathbf{I})\mathbf{v} = (\lambda + b)\mathbf{v}, \ \ \forall \lambda \in \sigma(\mathbf{A}).$$

Therefore, the eigenvalues of the Jacobian, which is a sum of this form with $b = \left(\frac{\phi_i}{\ell_i}\right)$, are

$$\sigma(\mathbf{J}(\mathbf{f}_i)) = \left\{ \left(\frac{\phi_i}{\ell_i} + \frac{\partial \phi_i}{\partial \ell_i} - \frac{\phi_i}{\ell_i} \right), \frac{\phi_i}{\ell_i}, \frac{\phi_i}{\ell_i} \right\} = \left\{ \frac{\partial \phi_i}{\partial \ell_i}, \frac{\phi_i}{\ell_i}, \frac{\phi_i}{\ell_i} \right\}. \tag{8.74}$$

□

Remark. The eigenvalue with multiplicity two,

$$\frac{\phi_i}{\ell_i},$$

has appeared before in this dissertation: it is the *force density* in a cable (Chap. 5.) The force density method, as used in Chap. 5, was initially developed in the 1970s to transform a set of nonlinear force balance equations for a cable network into a linear system [172]. Therefore, the reappearance of this quantity in an entirely different context, related to the dynamic equilibrium of the system, is remarkable.

The multiple-particle case is more involved, since the statically conservative potential U^f cannot be split according to cable: one cable's potential gives the forces for two particles. However, an analysis can still be conducted considering only the cable's scalar forces.

Lemma 8.4.0.2. *Eigenvalues of the Hessian of the statically conservative potential for a system of particles with multiple cables.*

Consider a system of K *particles, with position vectors* $\mathbf{r}_k$ *expressed in a Cartesian coordinate system. The set of cables which apply central forces with damping, where the force due to cable* i *on particle* k *is* $\mathbf{F}_i^k$*, has an associated statically conservative potential* $U^f = \sum U_i^f$ *which gives the total statically conservative force on each particle* $\mathbf{f}^k$ *as*

$$\nabla_{\mathbf{r}_k} U^f = \mathbf{f}^k, \qquad \mathbf{f}^k = \sum_{u=1}^{m} \phi_u^k(\ell_u)\hat{\boldsymbol{\ell}}_u - \sum_{v=1}^{p} \phi_v^k(\boldsymbol{\ell}_v)\hat{\boldsymbol{\ell}}_v,$$

where cable u *is directed toward particle* k *and* v *is directed away, and* $\phi_{(\cdot)}^k$ *represents the appropriate scalar force for the cable attached to particle* k.

Let there be a total of s_k *cables attached to each particle* k*. Then, the eigenvalues of the Hessian of* U^f*, also those of the Jacobian of the partial derivative of* U^f *with respect to the generalized coordinates* $\mathbf{q}$*, are*

$$\sigma(\mathbf{H}(U^f)) = \sigma\left(\mathbf{J}\left(\frac{\partial U^f}{\partial \mathbf{q}}\right)\right) = \bigcup_{k=1}^{K} \left\{ \left(\sum_{i=1}^{s_k} \frac{\partial \phi_i^k}{\partial \ell_i} \right), \left(\sum_{i=1}^{s_k} \frac{\phi_i^k}{\ell_i} \right) \right\} \tag{8.75}$$

where each $\sum_{i=1}^{s_k} \frac{\phi_i^k}{\ell_i}$ has multiplicity two. Informally, the eigenvalues are, for each particle, the sum of the derivatives of the scalar forces and sum of the force densities of all cables attached to that particle.

Proof. The Jacobian of the partial derivative vector $\frac{\partial U^f}{\partial \mathbf{q}}$ is

$$\mathbf{J}\left(\frac{\partial U^f}{\partial \mathbf{q}}\right) = \begin{bmatrix} \frac{\partial^2 U^f}{\partial^2 q_1} & \cdots & \frac{\partial^2 U^f}{\partial q_1 \partial q_{3K}} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 U^f}{\partial q_{3K} \partial q_1} & \cdots & \frac{\partial^2 U^f}{\partial^2 q_{3K}} \end{bmatrix} \in \mathbb{R}^{3K \times 3K} \tag{8.76}$$

with $\mathbf{q} \in \mathbb{R}^{3K}$, three generalized coordinates per particle.

Assume the following for the dynamics of the system:

1. No integrable constraints are imposed on the system (each particle has no kinematic relationship with any other.) This gives that each particle in the cable network moves freely, an assumption implicit until this point, so that three coordinates are needed per particle.

2. The generalized coordinates are ordered within $\mathbf{q}$ in blocks of three per particle, i.e., $\{q_1, q_2, q_3\}, \ldots, \{q_{3K-2}, q_{3K-1}, q_{3K}\}$, each block parameterizing only one particle.

In this case, the second partial derivative $\frac{\partial^2 U^f}{\partial q_h \partial q_j} = 0$ if h and j parameterize different particles, then the Jacobian is block structured per particle:

$$\mathbf{J}\left(\frac{\partial U^f}{\partial \mathbf{q}}\right) = \begin{bmatrix} \begin{bmatrix} \frac{\partial^2 U^f}{\partial^2 q_1} & \cdots & \frac{\partial^2 U^f}{\partial q_1 \partial q_3} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 U^f}{\partial q_3 \partial q_1} & \cdots & \frac{\partial^2 U^f}{\partial^2 q_3} \end{bmatrix} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \begin{bmatrix} \frac{\partial^2 U^f}{\partial^2 q_{3K-2}} & \cdots & \frac{\partial^2 U^f}{\partial q_{3K-2} \partial q_{3K}} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 U^f}{\partial q_{3K} \partial q_{3K-2}} & \cdots & \frac{\partial^2 U^f}{\partial^2 q_{3K}} \end{bmatrix} \end{bmatrix}. \tag{8.77}$$

Define the set of coordinates for particle k as $\mathbf{q}^k := [q_{3k-2}, q_{3k-1}, q_{3k}]^\top$, and the block of the Jacobian for that particle as

$$\mathbf{J}^k\left(\frac{\partial U^f}{\partial \mathbf{q}^k}\right) := \begin{bmatrix} \frac{\partial^2 U^f}{\partial^2 q_{3k-2}} & \cdots & \frac{\partial^2 U^f}{\partial q_{3k-2} \partial q_{3k}} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 U^f}{\partial q_{3k} \partial q_{3k-2}} & \cdots & \frac{\partial^2 U^f}{\partial^2 q_{3k}} \end{bmatrix}, \tag{8.78}$$

allowing the Jacobian for the whole system to be written as

$$\mathbf{J}\left(\frac{\partial U^f}{\partial \mathbf{q}}\right) = \begin{bmatrix} \mathbf{J}^1\left(\frac{\partial U^f}{\partial \mathbf{q}^1}\right) & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{J}^K\left(\frac{\partial U^f}{\partial \mathbf{q}^K}\right) \end{bmatrix}. \tag{8.79}$$

Since $\mathbf{J}$ is block diagonal, the set of its eigenvalues are the union of the set of eigenvalues of each block,

$$\sigma\left(\mathbf{J}\left(\frac{\partial U^f}{\partial \mathbf{q}}\right)\right) = \bigcup_{k=1}^{K} \sigma\left(\mathbf{J}^k\left(\frac{\partial U^f}{\partial \mathbf{q}^k}\right)\right). \tag{8.80}$$

So, then, consider each $\mathbf{J}^k$. By definition

$$\frac{\partial U^f}{\partial q_j} = \nabla_{\mathbf{r}_k} U^f \cdot \frac{\partial \mathbf{r}_k}{\partial q_j},$$

when the generalized coordinates are ordered into blocks. Employing the same assumption of Cartesian coordinates as in Lemma 8.4.0.1, the partial with respect to $\mathbf{q}^k$ can be analyzed instead as the partial with respect to $\mathbf{r}_k$, neglecting the basis transformation from $\mathbb{E}^3$ to $\mathbb{R}^3$:

$$\frac{\partial U^f}{\partial \mathbf{q}^k} \;\Rightarrow\; \nabla_{\mathbf{r}_k} U^f.$$

With this assumption, the gradient of the statically conservative potential with respect to each particle can be considered:

$$\mathbf{J}^k\left(\frac{\partial U^f}{\partial \mathbf{q}^k}\right) = \mathbf{J}^k\left(\nabla_{\mathbf{r}_k} U^f\right) = \mathbf{J}(\mathbf{f}^k). \tag{8.81}$$

Substituting for the sum of all cable's forces on one particle, and distributing the derivatives over the sum,

$$\begin{aligned} \mathbf{J}(\mathbf{f}^k) &= \mathbf{J}\left(\sum_{u=1}^{m} \mathbf{f}_u^k - \sum_{v=1}^{p} \mathbf{f}_v^k\right) && (8.82) \\ &= \sum_{u=1}^{m} \mathbf{J}(\phi_u^k(\ell_u)\hat{\boldsymbol{\ell}}_u) - \sum_{v=1}^{p} \mathbf{J}(\phi_v^k(\boldsymbol{\ell}_v)\hat{\boldsymbol{\ell}}_v) && (8.83) \end{aligned}$$

Following the same derivation as Lemma 8.4.0.1, the Jacobian of the cables directed either towards or away from the particle are

$$\mathbf{J}(\mathbf{f}_u^k) = \frac{\phi_u^k}{\ell_u}\mathbf{I} + \left(\frac{\partial \phi_u^k}{\partial \ell_u} - \frac{\phi_u^k}{\ell_u}\right)\hat{\boldsymbol{\ell}}_u\hat{\boldsymbol{\ell}}_u^\top \tag{8.84}$$

$$\mathbf{J}(\mathbf{f}_v^k) = -\frac{\phi_v^k}{\ell_v}\mathbf{I} - \left(\frac{\partial \phi_v^k}{\partial \ell_v} - \frac{\phi_v^k}{\ell_v}\right)\hat{\boldsymbol{\ell}}_v\hat{\boldsymbol{\ell}}_v^\top, \tag{8.85}$$

where eqn. (8.68) was used for the force directed away from the particle.

Substitution back into eqn. (8.83) shows that the negative signs for the force pointed away cancel out, and the Jacobian is then the sum of that over any cables attached to the particle under consideration. Specifically, recalling that each particle is defined to have s_k cables attached ($m + p = s_k$),

$$\mathbf{J}(\mathbf{f}^k) = \sum_{i=1}^{s_k}\left(\frac{\phi_i^k}{\ell_i}\mathbf{I} + \left(\frac{\partial \phi_i^k}{\partial \ell_i} - \frac{\phi_i^k}{\ell_i}\right)\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top\right) \tag{8.86}$$

$$\mathbf{J}(\mathbf{f}^k) = \sum_{i=1}^{s_k}\frac{\phi_i^k}{\ell_i}\mathbf{I} + \sum_{i=1}^{s_k}\left(\frac{\partial \phi_i^k}{\partial \ell_i} - \frac{\phi_i^k}{\ell_i}\right)\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^\top. \tag{8.87}$$

And, by the same derivation as Lemma 8.4.0.1,

$$\sigma\left(\mathbf{J}(\mathbf{f}^k)\right) = \left\{\left(\sum_{i=1}^{s_k}\frac{\partial \phi_i^k}{\partial \ell_i}\right), \left(\sum_{i=1}^{s_k}\frac{\phi_i^k}{\ell_i}\right), \left(\sum_{i=1}^{s_k}\frac{\phi_i^k}{\ell_i}\right)\right\} \tag{8.88}$$

The eigenvalues of the Hessian of the total statically conservative potential are then the union of these eigenvalues, for all particles k with each of their cable sums considered,

$$\mathbf{H}(U^f) = \bigcup_{k=1}^{K}\sigma\left(\mathbf{J}^k\left(\frac{\partial U^f}{\partial \mathbf{q}^k}\right)\right) \tag{8.89}$$

$$= \bigcup_{k=1}^{K}\sigma(\mathbf{J}(\mathbf{f}^k)) \tag{8.90}$$

$$= \bigcup_{k=1}^{K}\left\{\left(\sum_{i=1}^{s_k}\frac{\partial \phi_i^k}{\partial \ell_i}\right), \left(\sum_{i=1}^{s_k}\frac{\phi_i^k}{\ell_i}\right)\right\}, \tag{8.91}$$

where the sum of the normalized scalar forces for each particle has multiplicity two.

□

Remark. The result for the single particle is consistent with the result for multiple particles when combining all cables together, as opposed to analyzing each cable separately. However, the system of particles can only be considered as the total U^f, not each U_i^f, due to the couplings between cables and particles.

Observing this difference also highlights that, when considering the convexity of the statically conservative potential in either case, there are a large number of complicated conditions under which all eigenvalues are nonnegative (or positive.) For example, one particle may experience both one $\phi_i^k < 0$ and another $\phi_i^k > 0$ while still maintaining all positive eigenvalues. Consequently, the results below employ the simpler requirement that all scalar forces are positive and increasing, which is an artificial but useful restriction.

In addition, for the system of particles, the result demonstrates that all particles must have at least one cable attached (or else there will always be at least one zero eigenvalue.)

Minimum via convexity of the augmented potential for networks of central forces with damping

The second partial derivative test, examining the eigenvalues of the Hessian of a function, gives conditions under which the function is convex. The following presents two lemmas that consider two similar but slightly different approaches to using those conditions to meet the strict minimum requirement for $\bar{U}$ in Prop. 8.2.2.

Lemma 8.4.0.3. *Existence of a strict minimum of the augmented potential energy at the equilibrium point for a single particle under central forces with damping.*

Consider the augmented potential energy $\bar{U}$ for a single particle under the action of both gravity and central forces with damping,

$$\bar{U}(\mathbf{r}) = mg\mathbf{r} \cdot \mathbf{E}_3 + \sum_{i=1}^{s} U_i^f(\mathbf{r}).$$

If, within a set $\mathcal{B} \subseteq \mathbb{E}^3$ that does not include the anchor points $\mathbf{b}_i$ of any cable, the following conditions hold:

1. *The system's static force balance holds at a configuration $\mathbf{r} = \bar{\mathbf{r}} \in \mathcal{B}$, i.e.*

$$\nabla \bar{U}(\mathbf{r})|_{\mathbf{r}=\bar{\mathbf{r}}} = \mathbf{0},$$

2. *All cables i have scalar forces ϕ_i that are differentiable, nonnegative, and increasing for arguments $\ell_i(\mathbf{r})$ with $\mathbf{r} \in \mathcal{B}$,*

3. *There is at least one cable j that is pretensioned has and which has ϕ_j strictly increasing for arguments $\ell_j(\mathbf{r})$ with $\mathbf{r} \in \mathcal{B}$,*

then the augmented potential energy $\bar{U}$ has a strict minimum in $\mathcal{B}$ at $\mathbf{r} = \bar{\mathbf{r}}$, and in addition, $\bar{\mathbf{r}}$ is the unique equilibrium configuration for the system in $\mathcal{B}$.

Proof. A scalar field U_i^f is convex in a set $\mathcal{B}$ if its Hessian $\mathbf{H}(U^f)$ is positive semidefinite in $\mathcal{B}$, and is strictly convex if its Hessian is positive definite. Assumption 2 gives that, for all cables,

$$\exists \, \frac{\partial \phi_i}{\partial \ell_i}, \qquad \frac{\partial \phi_i}{\partial \ell_i} \geq 0, \qquad \frac{\phi_i}{\ell_i} \geq 0 \qquad \forall \, i = 1 \dots s, \tag{8.92}$$

since the cable lengths ℓ_i are always positive when $\mathbf{r} \neq \mathbf{b}_i$. These are the eigenvalues of $\mathbf{H}(U_i^f)$ by Lemma 8.4.0.1, thus $\mathbf{H}(U_i^f) \succeq 0$, and all U_i^f are convex in $\mathcal{B}$. Similarly, Assumption 3 gives that

$$\forall \, \mathbf{r} \in \mathcal{B}, \quad \exists \, \mathbf{f}_j(\mathbf{r}) \quad \text{s.t.} \quad \frac{\partial \phi_j}{\partial \ell_j} > 0, \quad \frac{\phi_j}{\ell_j} > 0. \tag{8.93}$$

The eigenvalues for that cable's Hessian are then all positive, $\mathbf{H}(U_j^f) \succ 0$, and U_j^f is strictly convex.

Consider the sum $\bar{U}$. Since $mg\mathbf{r} \cdot \mathbf{E}^3$ is affine, it is convex, and therefore $\bar{U}$ is the sum of convex functions. In addition, there is always at least one U_i^f which is strictly convex. The sum of convex functions is convex, and with at least one summand strictly convex is also then strictly convex. Under these conditions $\bar{U}$ is strictly convex.

Finally, Assumption 1 gives that $\bar{\mathbf{r}}$ is an extremum of $\bar{U}(\mathbf{r})$ in $\mathcal{B}$. Since $\bar{U}(\mathbf{r})$ is strictly convex, it has at most one extremum, which is a minimum. Therefore, $\bar{\mathbf{r}}$ is a strict minimum, and is unique in $\mathcal{B}$. It was shown in Prop. 8.2.2 that $\bar{\mathbf{r}}$ is therefore an equilibrium, and since it is unique, is the single equilibrium configuration in $\mathcal{B}$.

□

Extending this result to the system of particles involves the same argument; however, since the total U^f must be considered together, requires slightly different phrasing.

Lemma 8.4.0.4. *Existence of a strict minimum of the augmented potential energy at the equilibrium point for the system of particles under central forces with damping.*

Consider the augmented potential energy $\bar{U}$ for the system of K *particles under the action of both gravity and central forces with damping,*

$$\bar{U} = \sum_{k=1}^{K} mg\mathbf{r}_k \cdot \mathbf{E}_3 + \sum_{i=1}^{s} U_i^f,$$

where some cables may be attached to one particle and one anchor point, and the others attached to two particles. Consider also a region $\mathcal{Q} \subseteq \mathbb{R}^{3K}$ of the generalized coordinates' space that parameterizes a set of points $\mathcal{B} \subseteq \mathbb{E}^{3K}$ in the configuration space of the particles' position vectors. If $\mathcal{B}$ does not include any configurations where $\mathbf{r}_j = \mathbf{r}_k$ nor the anchor points $\mathbf{b}_i$ of any cable, and the following conditions hold:

1. *The system's static force balance holds at a configuration $\{\bar{\mathbf{r}}_1, \dots, \bar{\mathbf{r}}_K\} \in \mathcal{B}$ parameterized by $\bar{\mathbf{q}} \in \mathcal{Q}$, i.e.*

$$\nabla_{\mathbf{r}_k} \bar{U}(\mathbf{r}_1(\mathbf{q}), \ldots, \mathbf{r}_K(\mathbf{q}))|_{\mathbf{q}=\bar{\mathbf{q}}} = \mathbf{0} \qquad \forall\, k = 1 \ldots K,$$

2. *All cables i have scalar forces ϕ_i that are differentiable, nonnegative, and increasing for arguments $\ell_i(\mathbf{r}_j, \mathbf{r}_k)$ for $\{\mathbf{r}_1, \ldots, \mathbf{r}_K\} \in \mathcal{B}$,*

3. *For all particles k, there is at least one cable h attached to k that is pretensioned and which has $\phi_h(\ell_h)$ strictly increasing for arguments $\ell_h(\mathbf{r}_j, \mathbf{r}_k)$ for $\{\mathbf{r}_1, \ldots, \mathbf{r}_K\} \in \mathcal{B}$,*

then the augmented potential energy $\bar{U}$ has a strict minimum in $\mathcal{B}$ at $\{\bar{\mathbf{r}}_1, \ldots, \bar{\mathbf{r}}_K\}$ and equivalently $\bar{\mathbf{q}} \in \mathcal{Q}$, and in addition, $\{\bar{\mathbf{r}}_1, \ldots, \bar{\mathbf{r}}_K\}$ is the unique equilibrium configuration for the system in $\mathcal{B}$.

Proof. As with Lemma 8.4.0.3, consider the definiteness of $\mathbf{H}(U^f)$, the Hessian of the statically conservative potential of all the cables. Assumption 2 gives that, just as with the single particle lemma,

$$\exists\, \frac{\partial \phi_i}{\partial \ell_i}, \qquad \frac{\partial \phi_i}{\partial \ell_i} \geq 0, \qquad \frac{\phi_i}{\ell_i} \geq 0 \qquad \forall\, i = 1 \ldots s, \tag{8.94}$$

since all $\ell_i > 0$ when $\mathbf{r}_j \neq \mathbf{r}_k$ $\forall j, k$ and $\mathbf{r}_k \neq \mathbf{b}_i$ if i is a cable attached to an anchor point. The eigenvalues of $\mathbf{H}(U^f)$ are (via Lemma 8.4.0.2):

$$\sigma(\mathbf{H}(U^f)) = \bigcup_{k=1}^{K} \left\{ \left(\sum_{i=1}^{s_k} \frac{\partial \phi_i^k}{\partial \ell_i} \right), \left(\sum_{i=1}^{s_k} \frac{\phi_i^k}{\ell_i} \right) \right\},$$

and so since eqn. (8.94) holds for all cables irrespective of their attached particles or anchor points,

$$\frac{\partial \phi_i}{\partial \ell_i} \geq 0 \quad \forall i \quad \Rightarrow \sum_{i=1}^{s_k} \frac{\partial \phi_i^k}{\partial \ell_i} \geq 0 \quad \forall k \tag{8.95}$$

$$\frac{\phi_i}{\ell_i} \geq 0 \quad \forall i \quad \Rightarrow \sum_{i=1}^{s_k} \frac{\phi_i^k}{\ell_i} \geq 0 \quad \forall k, \tag{8.96}$$

$$\Rightarrow \quad \lambda \in \sigma(\mathbf{H}(U^f)) \geq 0, \tag{8.97}$$

so the Hessian is positive semidefinite, and U^f is convex in the spaces $\mathcal{Q}$ and equivalently $\mathcal{B}$.

Then, via Assumption 3,

$$\forall\, \{\mathbf{r}_1, \ldots \mathbf{r}_K\} \in \mathcal{B}, \quad \exists\, \mathbf{f}_h^k(\mathbf{r}_j, \mathbf{r}_k) \quad \text{s.t.} \quad \frac{\partial \phi_h^k}{\partial \ell_h} > 0, \quad \frac{\phi_h^k}{\ell_h} > 0. \tag{8.98}$$

Therefore, for each particle, eqns. (8.95)-(8.96) become

$$\exists\, h \quad \text{s.t.} \quad \frac{\partial \phi_h^k}{\partial \ell_h} > 0 \quad \Rightarrow \sum_{i=1}^{s_k} \frac{\partial \phi_i^k}{\partial \ell_i} > 0 \quad \forall k \tag{8.99}$$

$$\exists\, h \quad \text{s.t.} \quad \frac{\phi_h^k}{\ell_h} > 0 \quad \Rightarrow \sum_{i=1}^{s_k} \frac{\phi_i^k}{\ell_i} > 0 \quad \forall k, \tag{8.100}$$

i.e., at least one cable makes the inequality strict for both eigenvalues for every particle. All eigenvalues for the Hessian are then positive,

$$\Rightarrow \quad \lambda \in \sigma(\mathbf{H}(U^f)) > 0, \tag{8.101}$$

so the Hessian is positive definite, and U^f is strictly convex in the spaces $\mathcal{Q}$ and equivalently $\mathcal{B}$. By the same arguments concerning $\bar{U}$ as with Lemma 8.4.0.3, the system then has the unique equilibrium configuration $\{\bar{\mathbf{r}}_1 \dots \bar{\mathbf{r}}_K\}$.

□

As briefly mentioned above, this condition is restrictive: $\bar{U}$ could very well still be strictly convex when some cables' scalar forces (or their derivatives) are negative. A network-style analysis, using techniques such as expressing the interconnections of particles with the connectivity matrix $\mathbf{C}$ from Chap. 5, may be able to more eloquently describe the needed conditions. Doing so is left for future work.

8.5 Particles with Damped Central Forces: Stability

The preceding sections gave conditions on $\bar{U}$ that are needed to use the stability proof from Prop. 8.2.2. The remaining conditions, related to the time derivative of $\bar{U}$, are derived here. This section first considers the difference between a force and its statically conservative component, then assembles all the above lemmas into a stability proof.

8.5.1 Augmented Total Energy Exchange for Particles with Damped Central Forces

Prop. 8.2.1 and 8.2.2 require analyzing the time derivative of the augmented total energy, $\dot{V} = \dot{H} + \dot{U}^f$. From Sec. 8.2,

$$\dot{V} = -\dot{\mathbf{q}}^\top \mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}}) + \dot{\mathbf{q}}^\top \boldsymbol{\sigma}(\mathbf{q}) \tag{8.102}$$

$$= -\dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}), \tag{8.103}$$

where $\mathbf{S} := \mathbf{Q} - \boldsymbol{\sigma}$ is the difference between a (generalized) force and its statically conservative component.

As discussed in the background in Chap. 3, the generalized forces $\mathbf{Q} \in \mathbb{R}^{3K}$ and total forces on the particles $\mathbf{F}^k \in \mathbb{E}^3$ can be interchanged:

$$\dot{\mathbf{q}}^\top \mathbf{Q} = \sum_{k=1}^{K} \mathbf{F}^k \cdot \mathbf{v}_k,$$

for the sum over K particles. This identity also holds for the statically conservative component $\boldsymbol{\sigma}$, which becomes the forces $\mathbf{f}^k$ in $\mathbb{E}^3$ by substitution. Eqns. (8.102)-(8.103) therefore are equivalently,

$$\dot{V} = -\left[\sum_{k=1}^{K} \mathbf{F}^k \cdot \mathbf{v}_k - \sum_{k=1}^{K} \mathbf{f}^k \cdot \mathbf{v}_k\right] \tag{8.104}$$

$$= -\sum_{k=1}^{K} \mathbf{G}^k \cdot \mathbf{v}_k, \tag{8.105}$$

where the force difference for particle k in $\mathbb{E}^3$ has been defined as

$$\mathbf{G}^k := \mathbf{F}^k - \mathbf{f}^k. \tag{8.106}$$

There is a clear parallel, then, between the work-energy theorem and the proposed modification of that theorem:

$$H = T + U, \qquad \dot{H} = -\sum_{k=1}^{K} \mathbf{F}^k \cdot \mathbf{v}_k$$

$$V = T + U + U^f, \qquad \dot{V} = -\sum_{k=1}^{K} \mathbf{G}^k \cdot \mathbf{v}_k,$$

with the negative sign convention on the forces, as above, differing from the usual convention in the field of dynamics in order to use the framework of dissipativity.

For the system of particles with damped central forces, the augmented total energy balance can be further specified by defining

$$g_i(\ell_i, \dot{\ell}_i) := \Phi_i(\ell_i, \dot{\ell}_i) - \phi_i(\ell_i), \tag{8.107}$$

i.e., the difference in the scalar forces (total applied versus statically conservative component) for a cable. As with Φ_i and ϕ_i, the difference $g_i(\cdot,\cdot) : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$ is a scalar function. Then $\mathbf{G}_i^k$ for cable i with $\boldsymbol{\ell}_i = \mathbf{r}_k - \mathbf{r}_j$, for particle k, is

$$\mathbf{G}_i^k = \mathbf{F}_i^k - \mathbf{f}_i^k = \Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i - \phi_i(\ell_i)\hat{\boldsymbol{\ell}}_i = g_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i,$$

with the signed flipped for $\mathbf{G}_i^j$. The following lemma then characterizes the total exchange of augmented potential energy in terms of g_i of the cables.

Lemma 8.5.0.1. *Augmented Total Energy Exchange for Particles with Damped Central Forces.*

Consider a system of particles only acted upon by gravity and central forces with damping, which have dynamics described by eqn. (8.57). The amount of augmented total energy exchanged for the system over all particles, $-\dot{V}$, is equal to

$$-\dot{V} = \sum_{k=1}^{K} \mathbf{G}^k \cdot \mathbf{v}_k = \sum_{i=1}^{s} \dot{\ell}_i g_i(\ell_i, \dot{\ell}_i), \tag{8.108}$$

i.e., $\dot{V}$ can be analyzed per-particle or per-cable equivalently.

Proof. Expand the left-hand side of eqn. (8.108). Using the convention above where particle k has $u = 1 \dots m$ cables attached of which $\boldsymbol{\ell}_u = \mathbf{r}_k - \mathbf{r}_j$ and $v = 1 \dots p$ cables of which $\boldsymbol{\ell}_v = \mathbf{r}_j - \mathbf{r}_k$,

$$\sum_{k=1}^{K} \mathbf{G}^k \cdot \mathbf{v}_k = \sum_{k=1}^{K} \left[\left(\sum_{u=1}^{m} \mathbf{G}_u^k + \sum_{v=1}^{p} \mathbf{G}_v^k \right) \cdot \mathbf{v}_k \right] \tag{8.109}$$

$$= \sum_{k=1}^{K} \left[\sum_{u=1}^{m} g_u^k(\ell_u, \dot{\ell}_u) \hat{\boldsymbol{\ell}}_u \cdot \mathbf{v}_k + \sum_{v=1}^{p} g_v^k(\ell_v, \dot{\ell}_v) \hat{\boldsymbol{\ell}}_v \cdot \mathbf{v}_k \right]. \tag{8.110}$$

In comparison, consider the one cable $\boldsymbol{\ell}_i = \mathbf{r}_h - \mathbf{r}_j$. When attached to two particles, it always emits exactly one u term, for particle h, and exactly one v term, for particle j, in eqn. (8.110). Consequently, the sum can be expanded over all particles h and j, and terms can be grouped per cable:

$$\sum_{k=1}^{K} \left[\sum_{u=1}^{m} g_u^k(\ell_u, \dot{\ell}_u) \hat{\boldsymbol{\ell}}_u \cdot \mathbf{v}_k + \sum_{v=1}^{p} g_v^k(\ell_v, \dot{\ell}_v) \hat{\boldsymbol{\ell}}_v \cdot \mathbf{v}_k \right] = \sum_{i=1}^{s} \left(g_i(\ell_i, \dot{\ell}_i) \hat{\boldsymbol{\ell}}_i \cdot \mathbf{v}_h - g_i(\ell_i, \dot{\ell}_i) \hat{\boldsymbol{\ell}}_i \cdot \mathbf{v}_j \right). \tag{8.111}$$

Since

$$\dot{\ell}_i = \hat{\boldsymbol{\ell}}_i \cdot \mathbf{v}_h - \hat{\boldsymbol{\ell}}_i \cdot \mathbf{v}_j,$$

the terms on the right-hand side of eqn. (8.111) become

$$g_i(\ell_i, \dot{\ell}_i) \hat{\boldsymbol{\ell}}_i \cdot \mathbf{v}_h - g_i(\ell_i, \dot{\ell}_i) \hat{\boldsymbol{\ell}}_i \cdot \mathbf{v}_j = g_i(\ell_i, \dot{\ell}_i) \dot{\ell}_i, \tag{8.112}$$

proving the result:

$$\sum_{i=1}^{s}\left(g_i(\ell_i,\dot{\ell}_i)\hat{\boldsymbol{\ell}}_i\cdot\mathbf{v}_h - g_i(\ell_i,\dot{\ell}_i)\hat{\boldsymbol{\ell}}_i\cdot\mathbf{v}_j\right) = \sum_{i=1}^{s}\dot{\ell}_i g_i(\ell_i,\dot{\ell}_i). \tag{8.113}$$

The case where a cable connects to an anchor as one of its points, $\boldsymbol{\ell}_i = \mathbf{r}_k - \mathbf{b}_i$, follows similarly since $\dot{\mathbf{b}}_i = 0$.

□

Remark. The above lemma demonstrates the convenience of the central forces with damping framework. The somewhat complicated and unintuitive expression for $\dot{V}$, in terms of generalized coordinates and forces, is now instead the sum of scalar functions per cable:

$$\dot{V} = -\dot{\mathbf{q}}^{\top}\mathbf{S}(\mathbf{q},\dot{\mathbf{q}}) = -\sum_{i=1}^{s}\dot{\ell}_i g_i(\ell_i,\dot{\ell}_i).$$

The claim at the beginning of this chapter - that the framework here allows for developing a distributed control system - is now more clearly motivated. Each cable can be analyzed independently, and therefore individually, since no ℓ_i or $\dot{\ell}_i$ terms interact between cables. The controller proposed below will therefore considers each ℓ_i (and $\dot{\ell}_i$, in future work) for feedback in a given ϕ_i.

8.5.2 Stability of Systems of Particles with Damped Central Forces

Given this lemma, a stability proof can be constructed for these systems of particles with damped central forces. As discussed in Sec. 8.4.2, there are potentially many other ways to meet these conditions. However, applying the per-cable conditions of pretension, etc., allows for scalar distributed controllers in later applications.

Proposition 8.5.1. *Stability of Systems of Particles with Damped Central Forces.*

Consider a system of particles only acted upon by gravity and central forces with damping, which have dynamics described by eqn. (8.57), rewritten here:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \left(\frac{\partial L}{\partial q_j}\right) = -\sum_{k=1}^{K}\sum_{i=1}^{s}\mathbf{F}_i^k\cdot\frac{\partial \mathbf{r}_k}{\partial q_j} \qquad j = 1\ldots 3K,$$

where force i acts between particles j and k with $\boldsymbol{\ell}_i = \mathbf{r}_k - \mathbf{r}_j$, as in

$$\mathbf{F}_i^k(\mathbf{r}_k,\mathbf{r}_j,\mathbf{v}_k,\mathbf{v}_j) = \Phi_i(\ell_i,\dot{\ell}_i)\hat{\boldsymbol{\ell}}_i, \qquad \mathbf{F}_i^j(\mathbf{r}_k,\mathbf{r}_j,\mathbf{v}_k,\mathbf{v}_j) = -\Phi_i(\ell_i,\dot{\ell}_i)\hat{\boldsymbol{\ell}}_i.$$

Consider also a region $\mathcal{Q}\subseteq\mathbb{R}^{3K}$ of the generalized coordinates' space that parameterizes a set of points $\mathcal{B}\subseteq\mathbb{E}^{3K}$ in the configuration space of the particles' position vectors. If $\mathcal{B}$ does not include any configurations where $\mathbf{r}_j = \mathbf{r}_k$ nor the anchor points $\mathbf{b}_i$ of any cable, and the following conditions hold:

1. *The system's static force balance holds at a configuration* $\{\bar{\mathbf{r}}_1, \ldots, \bar{\mathbf{r}}_K\} \in \mathcal{B}$ *parameterized by* $\bar{\mathbf{q}} \in \mathcal{Q}$*, i.e.*

$$\nabla_{\mathbf{r}_k} \bar{U}(\mathbf{r}_1(\mathbf{q}), \ldots, \mathbf{r}_K(\mathbf{q}))|_{\mathbf{q}=\bar{\mathbf{q}}} = \mathbf{0} \qquad \forall\, k = 1 \ldots K,$$

2. *All cables* i *have scalar forces* ϕ_i *that are differentiable, nonnegative, and increasing for arguments* $\ell_i(\mathbf{r}_j, \mathbf{r}_k)$ *for* $\{\mathbf{r}_1, \ldots, \mathbf{r}_K\} \in \mathcal{B}$*,*

3. *For all particles* k*, there is at least one cable* h *attached to* k *that is pretensioned and which has* $\phi_h(\ell_h)$ *strictly increasing for arguments* $\ell_h(\mathbf{r}_j, \mathbf{r}_k)$ *for* $\{\mathbf{r}_1, \ldots, \mathbf{r}_K\} \in \mathcal{B}$*,*

4. *The scalar force difference for each cable, considered as a memoryless nonlinearity* $z = g_i(x, y)$*, satisfies the following inequality for all cables,*

$$y g_i(x, y) \geq 0 \qquad \forall\, x, y, \quad \forall\, i,$$

5. *The scalar force difference for each cable additionally satisfies, for all cables,*

$$y g_i(x, y) > 0 \qquad \forall\, x, \quad \forall\, y \neq 0, \quad \forall\, i,$$

then the configuration $\{\bar{\mathbf{r}}_1, \ldots, \bar{\mathbf{r}}_K\}$ *is a locally asymptotically stable equilibrium in* $\mathcal{B}$*. In addition, if*

6. *All scalar conservative forces* ϕ_i *are radially unbounded:*

$$\lim_{\ell_i \to \infty} \phi_i(\ell_i) = \infty \qquad \forall\, i,$$

then $\{\bar{\mathbf{r}}_1, \ldots, \bar{\mathbf{r}}_K\}$ *is asymptotically stable in all of* $\mathcal{B}$*.*

Proof. The augmented potential energy always exists in this system, in the form needed to satisfy Prop. 8.2.1, via Lemma 8.3.0.1. The conditions above also satisfy Lemma 8.4.0.4, which gives that the system has a strict minimum of $\bar{U}$ at $\{\bar{\mathbf{r}}_1, \ldots, \bar{\mathbf{r}}_K\}$, which is then in turn a unique equilibrium configuration in $\mathcal{B}$.

Via Lemma 8.5.0.1,

$$\dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) = \sum_{i=1}^{s} \dot{\ell}_i g_i(\ell_i, \dot{\ell}_i).$$

Assumption 4 above gives that

$$yg_i(x,y) \geq 0 \quad \forall\, i, x, y \quad \Rightarrow \quad \sum_{i=1}^{s} \dot{\ell}_i g_i(\ell_i, \dot{\ell}_i) \geq 0 \quad \forall\, i, \ell_i, \dot{\ell}_i, \tag{8.114}$$

$$\Rightarrow \quad \dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) \geq 0 \quad \forall\, \mathbf{q} \in \mathcal{Q},\ \forall\, \dot{\mathbf{q}}, \tag{8.115}$$

due to the dependence of ℓ_i on $\mathbf{r}_k, \mathbf{r}_j$ and in turn $\mathbf{q}$, similarly with $\dot{\ell}_i$ on $\mathbf{v}_k, \mathbf{v}_j$ and in turn $\dot{\mathbf{q}}$.

Incorporating Assumption 5 gives strictness for this inequality in the same way,

$$yg_i(x,y) > 0 \quad \forall\, x,\ \forall\, y \neq 0,\ \forall i, \quad \Rightarrow \quad \sum_{i=1}^{s} \dot{\ell}_i g_i(\ell_i, \dot{\ell}_i) > 0 \quad \forall\, \ell_i,\ \forall \dot{\ell}_i \neq 0,\ \forall i \tag{8.116}$$

$$\Rightarrow \quad \dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) > 0 \quad \forall\, \mathbf{q} \in \mathcal{Q},\ \forall\, \dot{\mathbf{q}} \neq \mathbf{0}. \tag{8.117}$$

The above satisfy all conditions of Props. 8.2.1 and 8.2.2, giving the desired local asymptotic stability of $\{\mathbf{r}_1, \ldots, \mathbf{r}_K\} \in \mathcal{B}$, with $\mathbf{v}_1, \ldots \mathbf{v}_K = \mathbf{0}$.

If in addition assumption 6 holds, then

$$\begin{aligned}
\lim_{\mathbf{q}\to\infty} \ell_i(\mathbf{r}_j(\mathbf{q}), \mathbf{r}_k(\mathbf{q})) = \infty \quad &\Rightarrow \quad \lim_{\mathbf{q}\to\infty} \phi_i(\ell_i) = \infty \\
&\Rightarrow \quad \lim_{\mathbf{q}\to\infty} \int_a^{\ell_i} \phi_i(\tau) d\tau = \infty \\
&\Rightarrow \quad \lim_{\mathbf{q}\to\infty} U_i^f = \infty \\
&\Rightarrow \quad \lim_{\mathbf{q}\to\infty} \sum_{i=1}^{s} U_i^f = \infty \\
&\Rightarrow \quad \lim_{\mathbf{q}\to\infty} \bar{U} = \infty,
\end{aligned}$$

since gravitational potential energy is also unbounded as $\mathbf{q} \to \infty$. This meets the globalness condition of Prop. 8.2.2, so the region of asymptotic stability becomes at least $\mathcal{B} \subseteq \mathbb{E}^{3K}$.

□

Remark. Assumption 6 does not give that $\mathcal{B}$ is a region of attraction. It should instead be understood to show that the local region of stability is not an arbitrary region around the equilibrium, but extends (at least) to $\mathcal{B}$. A condition on global asymptotic stability is not possible here, since in $\mathbb{E}^{3K}$ there are always configurations where, for example, particles intersect with their anchor points ($\mathbf{r}_k = \mathbf{b}_k$). At these configurations, some cable lengths are zero, and the Hessian of the augmented potential energy becomes semidefinite.

For the cable-driven robot with slack cables, studied below, this stability proof cannot be applied directly. In particular, Assumption 5 will not hold. Instead, the following slightly different proof can be used. It is less general than the above case.

Corollary 8.5.1.1. *Stability of Systems of Particles with Damped Central Forces, Zero Compression Force.*

Consider the same system as in Prop. 8.5.1. If the cables' scalar forces do not satisfy the strictness condition of Assumption 5, but instead, the following hold:

1. *The region $\mathcal{B}$ (and equivalently the parameter space $\mathcal{Q}$) is defined such that, in $\mathcal{B}$, there always exists one cable per particle whose length is increasing when that particle is in motion:*

$$\forall\, \mathbf{q} \in \mathcal{Q} \text{ and } \mathbf{v}_1, \ldots, \mathbf{v}_K \neq 0, \quad \exists\, \dot{\ell}_j > 0$$

2. *All cables' scalar force differences satisfy (same notation as Prop. 8.5.1)*

$$y g_i(x, y) > 0 \qquad \forall\, x > 0,\, \forall\, y > 0,\, \forall\, i$$

then the system is locally asymptotically stable in $\mathcal{B}$ as per Prop. 8.5.1.

Proof. Assumption 1 gives that, via the dependence of all respective $\dot{\ell}_j$ on the $\mathbf{v}$ of their attached particles and in turn $\dot{\mathbf{q}}$,

$$\dot{\mathbf{q}} \neq \mathbf{0} \;\Rightarrow\; \mathbf{v}_1, \ldots, \mathbf{v}_K \neq 0 \quad \Rightarrow\; \exists\; \dot{\ell}_j > 0 \quad \forall\, \mathbf{q} \in \mathcal{Q}.$$

Also note again that $\ell_i(\mathbf{r}_k, \mathbf{r}_j) > 0\ \forall \mathbf{r}_k, \mathbf{r}_j \in \mathcal{B}$, i.e., lengths are always positive when the particles do not overlap or intersect. Therefore,

$$\mathbf{q} \in \mathcal{Q} \;\Rightarrow\; \ell_i > 0 \quad \forall\, i.$$

Incorporating the second assumption, with the inequality from Prop. 8.5.1 Assumption 4 still holding, the above makes the inequality strict:

$$\begin{aligned} y g_i(x, y) > 0 \quad \forall\, x > 0,\, \forall\, y > 0\, \forall i \quad &\Rightarrow \exists\; \dot{\ell}_j g_j(\ell_j, \dot{\ell}_j) > 0 \quad \forall\, \dot{\mathbf{q}} \neq \mathbf{0}\, \forall\, \mathbf{q} \in \mathcal{Q} \\ &\Rightarrow \sum_{i=1}^{s} \dot{\ell}_i g_i(\ell_i, \dot{\ell}_i) > 0 \quad \forall\, \dot{\mathbf{q}} \neq \mathbf{0}\, \forall\, \mathbf{q} \in \mathcal{Q} \\ &\Rightarrow \dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) > 0 \quad \forall\, \dot{\mathbf{q}} \neq \mathbf{0}\, \forall\, \mathbf{q} \in \mathcal{Q}. \end{aligned}$$

The above satisfy all conditions of Props. 8.2.1 and 8.2.2, giving the desired local asymptotic stability of $\{\mathbf{r}_1, \ldots, \mathbf{r}_K\} \in \mathcal{B}$, with $\mathbf{v}_1, \ldots \mathbf{v}_K = \mathbf{0}$, or equivalently in all of $\mathcal{B}$ with radial unboundedness.

□

Remark. An informal statement of this proof is:

Find a region of the state space where moving particles imply that one attached cable is always extending. Then if the strictness condition on the scalar force difference, g_i, holds for positive arguments for all cables, the system is asymptotically stable in that region.

8.5.3 Cable Stretch Rates Within The Convex Hull of Anchor Points

The above corollary motivates the question: when is it the case that at least one cable is always extending if a particle is in motion? That is, where in $\mathbb{E}^{3K}$ is it true that

$$\mathbf{v}_1, \ldots, \mathbf{v}_K \neq 0 \quad \Rightarrow \quad \exists\, \dot{\ell}_j > 0.$$

For the single particle, the answer can be shown geometrically: one cable is always extending when the particle is in motion within the convex hull of its anchor points. Figure 8.2 shows a diagram of this lemma to help with intuition, since the proof relies on the Separating Hyperplane Theorem [30].

Lemma 8.5.1.1. *Inner products of vertex vectors with the normal to a separating hyperplane.*

Consider a finite set of vectors $\mathcal{X} = \{\mathbf{b}_i \in \mathbb{E}^3\}$, and their convex hull, $\mathcal{P} = \mathrm{Conv}(\mathcal{X})$. Let a point $\mathbf{r} \in \mathbb{E}^3$ be in the interior of $\mathcal{P}$. Define the following:

- $\mathcal{W} = \{\mathbf{w}_i \in \mathbb{E}^3\} \subseteq \mathcal{X}$, *the set of vertices of $\mathcal{P}$, by definition a subset of $\mathcal{X}$,*
- $\mathcal{L} = \{\boldsymbol{\ell}_i \in \mathbb{E}^3 \mid \boldsymbol{\ell}_i = \mathbf{w}_i - \mathbf{r}\}$, *the set of vectors pointing to $\mathbf{r}$ from these vertices, by definition a subset of $\mathcal{L}' = \{\boldsymbol{\ell}'_i \in \mathbb{E}^3 \mid \boldsymbol{\ell}'_i = \mathbf{b}_i - \mathbf{r}\}$, the corresponding vectors from $\mathcal{X}$.*
- *An arbitrary nonzero vector $\mathbf{v}$ pointing from $\mathbf{r}$, as in $\mathbf{v}' = \mathbf{r} + \mathbf{v}$ with respect to the origin,*
- $\mathcal{A} = \{a_i \in \mathbb{R} \mid a_i = \mathbf{v} \cdot \boldsymbol{\ell}'_i\}$, *the inner products of this arbitrary vector with each $\boldsymbol{\ell}'_i$,*

Then, at least one element of $\mathcal{A}$ is greater than zero and at least one is less than zero: $\exists\, a_i > 0$ and $\exists\, a_k < 0$.

Proof. Without loss of generality, change coordinates such that $\mathbf{r} = \mathbf{0}$, so that the polytope $\mathcal{P}$ now has vertices $\{\mathbf{w}_i - \mathbf{r} = \boldsymbol{\ell}_i\} = \mathcal{L}$. The following uses $\mathcal{P}$ to denote the new polytope, abusing notation for clarity.

Given a nonzero $\mathbf{v}$, its normal hyperplane that passes through the origin (now, $\mathbf{0} = \mathbf{r}$) is defined as the set of points

$$\mathcal{H} = \{\mathbf{h}_k \in \mathbb{E}^3 \mid \mathbf{h}_k \cdot \mathbf{v} = 0\}$$

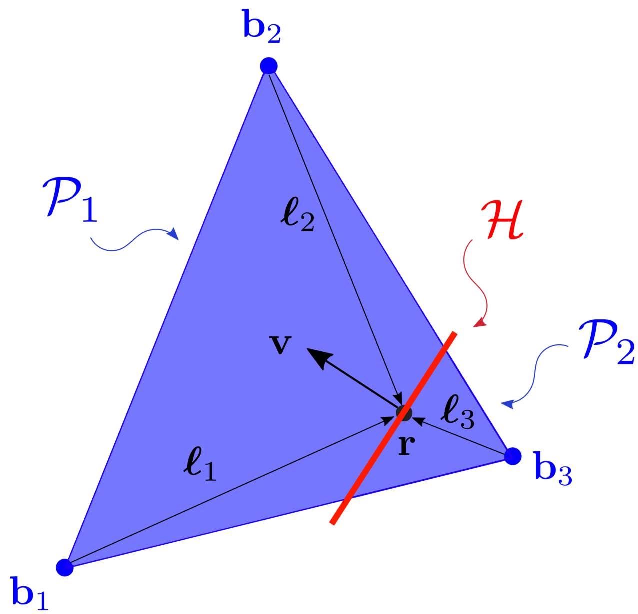


Figure 8.2: Diagram of the quantities from Lemma 8.5.1.1. The particle is at position $\mathbf{r}$ in the interior of the convex hull of the cables' anchor points $\mathbf{b}_i$. The proof shows that at least one $\dot{\ell}_i = \mathbf{v} \cdot \hat{\boldsymbol{\ell}}_i$ will be positive and one will be negative, for any direction of the particle velocity $\mathbf{v}$. The proof proceeds by partitioning the convex hull using the normal hyperplane $\mathcal{H}$ for $\mathbf{v}$. The result can visually intuited with this example $\mathbf{v}$ in the figure, and the angles of each $\boldsymbol{\ell}_i$ with respect to $\mathbf{v}$. It is clear that $\mathbf{v} \cdot \hat{\boldsymbol{\ell}}_1 < 0$ and $\mathbf{v} \cdot \hat{\boldsymbol{\ell}}_2 < 0$, whereas $\mathbf{v} \cdot \hat{\boldsymbol{\ell}}_3 > 0$. Since $\mathbf{r}$ is in the interior of $\mathcal{P}$, any direction of $\mathbf{v}$ results in at least one $\boldsymbol{\ell}_i$ residing in each partition.

Let the hyperplane partition $\mathcal{P}$. This results in three proper, convex subsets by the property of partitioning Euclidean space by a hyperplane. Denote the subsets

$$\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2 \cup \mathcal{H}' \tag{8.118}$$

where $\mathcal{H}' = \mathcal{H} \cup \mathcal{P}$ is the portion of the hyperplane that lies within $\mathcal{P}$. Here, $\mathcal{P}_1$ and $\mathcal{P}_2$ are convex polyhedra since $\mathcal{P}$ is a convex polyhedron, are disjoint since each subset is proper, and are open since neither contain their boundary $\mathcal{H}'$.

Next, show that at least one of the original vertices in $\mathcal{L}$ is contained within $\mathcal{P}_1$, and at least one (different) vertex is contained within $\mathcal{P}_2$. Since $\mathbf{0}$ is in the interior of $\mathcal{P}$ under the change of coordinates, there exists an open ball of radius $r > 0$ centered at $\mathbf{0}$ which is completely contained in $\mathcal{P}$. Therefore, there are points within $\mathcal{P}$ along a vector in any direction, with a distance that is slightly smaller than the radius $0 < \varepsilon < r$ of this ball. Taking this direction to be the unit vector $\hat{\mathbf{v}} = \frac{\mathbf{v}}{||\mathbf{v}||}$, then the points

$$\mathbf{p}_1 = \varepsilon \hat{\mathbf{v}} \subset \mathcal{P}, \qquad \mathbf{p}_2 = -\varepsilon \hat{\mathbf{v}} \subset \mathcal{P}$$

are both contained within $\mathcal{P}$. By the chosen partition, then $\mathbf{p}_1 \subseteq \mathcal{P}_1$ and $\mathbf{p}_2 \subseteq \mathcal{P}_2$, and thus both sets are nonempty in addition to being disjoint.

Since $\mathcal{P}_{1,2}$ are constructed from convex hulls around finite sets of points, and are nonempty, they are polyhedra with their own sets of vertices $\mathcal{W}_1$ and $\mathcal{W}_2$. The partitioning of $\mathcal{P}$ may add new vertices to $\mathcal{W}_{1,2}$ in addition to the original set of $\mathcal{L}$. These vertices can only be drawn from the hyperplane, where the partitioning occurs. Denote these new vertices $\mathcal{W}_{\mathcal{H}} \subset \mathcal{H}$. Then, elements of $\mathcal{W}_{1,2}$ must be drawn either from $\mathcal{W}_{\mathcal{H}}$ or the original vertices $\mathcal{L}$, as in

$$\mathcal{W}_1 \subset (\mathcal{L} \cup \mathcal{W}_{\mathcal{H}}), \qquad \mathcal{W}_2 \subset (\mathcal{L} \cup \mathcal{W}_{\mathcal{H}}) \tag{8.119}$$

Since $\mathcal{W}_{\mathcal{H}}$ are in a hyperplane which does not contain $\mathbf{p}_{1,2}$, in order for $\mathbf{p}_{1,2}$ to be convex combinations of $\mathcal{W}_{1,2}$, other vertices must exist in $\mathcal{W}_{1,2}$ other than those in $\mathcal{W}_{\mathcal{H}}$. Specifically, at least one element of $\mathcal{L}$ must be present in each set of $\mathcal{W}_{1,2}$ in order to express $\mathbf{p}_{1,2} \notin \mathcal{H}$, so

$$\exists \, \{\boldsymbol{\ell}_i, \boldsymbol{\ell}_j\} \in \mathcal{L} \quad \text{s.t.} \quad \boldsymbol{\ell}_i \in \mathcal{W}_1 \subset \mathcal{P}_1, \quad \boldsymbol{\ell}_j \in \mathcal{W}_2 \subset \mathcal{P}_2$$

Finally, the Hyperplane Separation Theorem [30] states that, given two disjoint, convex, open, nonempty sets $\mathcal{F}$ and $\mathcal{G}$, there exists a hyperplane defined by a normal vector $\mathbf{z}$ and an offset c, as in $\bar{\mathcal{H}} = \{\bar{\mathbf{h}}_k \in \mathbb{E}^3 \mid \bar{\mathbf{h}}_k \cdot \mathbf{z} = c\}$, such that

$$\forall \, \mathbf{x} \in \mathcal{F}, \ \forall \, \mathbf{y} \in \mathcal{G} : \qquad \mathbf{x} \cdot \mathbf{z} > c, \qquad \mathbf{y} \cdot \mathbf{z} < c$$

By construction, $\mathcal{H}$ is one such hyperplane for $\mathcal{P}_1$ and $\mathcal{P}_2$, both of which are disjoint, convex, open, and nonempty, so

$$\forall \, \mathbf{x} \in \mathcal{P}_1, \ \forall \, \mathbf{y} \in \mathcal{P}_2 : \qquad \mathbf{x} \cdot \mathbf{v} > 0, \qquad \mathbf{y} \cdot \mathbf{v} < 0$$

or vice-versa depending on the naming convention for the subsets. This must also hold, then, for all the sets' vertices $\mathcal{W}_{1,2}$, and so

$$\boldsymbol{\ell}_i \cdot \mathbf{v} > 0, \qquad \boldsymbol{\ell}_j \cdot \mathbf{v} < 0$$

These inner products are elements of $\mathcal{A}$, as in $a_k = \boldsymbol{\ell}_k \cdot \mathbf{v}$, therefore $a_i > 0$ and $a_j < 0$, completing the proof.

□

The above lemma gives that, for the single-particle case, Corollary 8.5.1.1 will hold for $\mathcal{B} = \mathrm{Conv}(\mathbf{b}_i)$.

8.6 Energy-Shaping Control of a Cable-Driven Robot with Slack Cables

The stability conditions above apply to systems of particles, which could represent a large number of systems ranging from cable networks to soft robotic skins to finite-element models of cloth. The following section applies the stability propositions to the motivating example of a robot driven by cables, which may go slack at points in the state space. This application consists of only one particle with multiple anchored cables.

Models of cable driven robots, suspended in space, are commonly represented with rigid bodies and applied cable forces [44, 104, 218], etc. However, if the robot is very small in comparison to its cables' lengths, its rotational inertia could be neglected, and instead the body could be approximated by a point mass (e.g. [125].) This approximated case is useful for exposition of the proposed technique in this chapter, since it also addresses major challenges related to tensegrity spines. Reformulating the conditions above for rigid bodies is left for future work.

So, consider a point-mass cable-driven robot, of the form shown in Fig. 8.1. Its model is that considered up to this point in the chapter, with $K = 1$ particle. Specifically, with particle position $\mathbf{r}$ and velocity $\mathbf{v}$, and with a set of cables $\boldsymbol{\ell}_i = \mathbf{r} - \mathbf{b}_i$ with anchor points $\mathbf{b}_i$, the robot is described by Lagrange's equations (8.57) rewritten here as

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \left(\frac{\partial L}{\partial q_j}\right) = -\sum_{i=1}^{s} \mathbf{F}_i \cdot \frac{\partial \mathbf{r}}{\partial q_j} \qquad j = 1 \ldots 3,$$

where the force from cable i is

$$\mathbf{F}_i(\mathbf{r}, \mathbf{v}) = \Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i.$$

The control goal considered here is setpoint regulation of the robot: moving it to a pose in its state space, i.e., a desired position $\bar{\mathbf{r}} \in \mathbb{E}^3$. To do so, this section first specifies the control input u_i for each cable, and gives a more specific definition of the cable model $\Phi_i(\ell_i, \dot{\ell}_i)$. A proof, using Corollary 8.5.1.1, is given for this case. Finally, a distributed output-feedback controller is proposed, meeting stability condition. Simulations then validate the result for a variety of initial conditions.

8.6.1 Cable Model

There are a variety of different possible models for incorporating control inputs into this system. As mentioned before, a number of authors assume direct control over the tension in the cables (for example, [15, 43, 44], among many others.) This is not appropriate for designs with low-bandwidth motors that control cables. Instead, all models considered here assume that the rest length of the cable can be controlled:

$$\Phi_i(\ell_i, \dot{\ell}_i, u_i) := \begin{cases} k_i(\ell_i - u_i) + c_i\dot{\ell}_i, & \text{if} \quad k_i(\ell_i - u_i) + c_i\dot{\ell}_i \geq 0 \\ 0, & \text{if} \quad k_i(\ell_i - u_i) + c_i\dot{\ell}_i < 0, \end{cases} \tag{8.120}$$

with $k_i, c_i > 0 \in \mathbb{R}$, and u_i are inputs to the system, so $\mathbf{u} = [u_1, \ldots, u_s]$ is the input vector.

Assume that a distributed output-feedback controller is proposed for the system, of the form $u_i = u_i(\ell_i)$, where the controller is a function of only that cable's current length. The closed loop model of a cable then only depends on ℓ_i and $\dot{\ell}_i$,

$$\Phi_i(\ell_i, \dot{\ell}_i) = \begin{cases} k_i(\ell_i - u_i(\ell_i)) + c_i\dot{\ell}_i, & \text{if} \quad k_i(\ell_i - u_i(\ell_i)) + c_i\dot{\ell}_i \geq 0 \\ 0, & \text{if} \quad k_i(\ell_i - u_i(\ell_i)) + c_i\dot{\ell}_i < 0. \end{cases} \tag{8.121}$$

Incorporating control inputs as cable stretch

Define the amount of stretch for cable i as the difference between its current length and rest length, with $u_i(\ell_i)$ assumed to be a pre-specified memoryless function,

$$\varepsilon_i(\ell_i) := \ell_i - u_i(\ell_i). \tag{8.122}$$

Substitution into eqn. (8.121) emphasizes that the cable model is still only a (possibly) nonlinear function of the length and its time derivative,

$$\Phi_i(\ell_i, \dot{\ell}_i) = \begin{cases} k_i\varepsilon_i(\ell_i) + c_i\dot{\ell}_i, & \text{if} \quad k_i\varepsilon_i(\ell_i) + c_i\dot{\ell}_i \geq 0 \\ 0, & \text{if} \quad k_i\varepsilon_i(\ell_i) + c_i\dot{\ell}_i < 0, \end{cases} \tag{8.123}$$

and can be analyzed with the tools in this chapter. It is clear that the existence of the augmented potential energy (Lemma 8.3.0.1) holds for the model (8.123), since ε_i can be incorporated into the function ϕ_i via composition.

Time derivative of the cable stretch

This model incorporates the control input (equivalently, cable stretch) into only the spring term. Though this is an extremely common assumption [55, 19, 18] and is employed in the most common simulator for tensegrity robots [38], it may not be accurate with large changes in control input. Intuitively, the rate of change in a cable's stretch is both a function of its (time varying) length and (time varying) rest length treated as the input,

$$\frac{d}{dt}\varepsilon_i = \frac{d}{dt}(\ell_i - u_i) = \dot{\ell}_i - \dot{u}_i,$$

A more physically realistic model then incorporates the time derivative of the total stretch for the damping term, as opposed to just that of the length:

$$\Phi_i(\ell_i, \dot{\ell}_i) = \begin{cases} k_i\varepsilon_i(\ell_i) + c_i\dot{\varepsilon}_i(\dot{\ell}_i), & \text{if} \quad k_i\varepsilon_i(\ell_i) + c_i\dot{\varepsilon}_i(\dot{\ell}_i) \geq 0 \\ 0, & \text{if} \quad k_i\varepsilon_i(\ell_i) + c_i\dot{\varepsilon}_i(\dot{\ell}_i) < 0, \end{cases} \tag{8.124}$$

where it is implicitly assumed that $u_i(\ell_i)$ is chosen such that its time derivative $\dot{u}_i$ is only a function of $\dot{\ell}_i$ and not ℓ_i, as in

$$\frac{d\dot{u}_i}{d\ell_i} = 0.$$

Though this is the case with the proposed controller in later sections, it is not the case in general, and the more general cases are left for future work with more optimal controllers.

The analysis in this chapter will consider $c_i\dot{\varepsilon}_i(\dot{\ell}_i)$ as some nonlinear, memoryless function of $\dot{\ell}_i$. It will be shown that, under the condition that the controller guarantees

$$\operatorname{sgn}(\dot{\varepsilon}_i(\dot{\ell}_i)) = \operatorname{sgn}(\dot{\ell}_i),$$

then a controller which stabilizes model (8.123) will also work for model (8.124). Ensuring this will be reduced to an additional minor condition on the proposed controller.

The filtered sum model

The above observation - that incorporating a control input corresponds to modeling a nonlinear spring/damper - prompts a further specification of the $\Phi_i(\ell_i, \dot{\ell}_i)$ model. In addition, one of the original issues raised with control of a slack cable system is that of hybrid behavior due to the function (8.124). The following subsection incorporates both these phenomena into one model: that of the filtered sum.

First, notice that saturation can be written using the Heaviside step function, $H(x)$,

$$y(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \quad \Longleftrightarrow \quad y(x) = xH(x), \tag{8.125}$$

in other words, as a unit ramp. This piecewise linear function can therefore capture the slack behavior of the cables.

Consequently, consider the following cable force model that is a *filtered sum* of nonlinear memoryless functions,

$$\Phi_i(\ell_i, \dot{\ell}_i) := A_i(B_i(\ell_i) + C_i(\dot{\ell}_i)), \tag{8.126}$$

where $A_i(\cdot), B_i(\cdot), C_i(\cdot) : \mathbb{R} \mapsto \mathbb{R}$ are each scalar. Writing the model (8.124) in terms of (8.126) determines each function as the following.

$$\begin{aligned} A_i(x) &:= xH(x) \\ B_i(x) &:= k_i\varepsilon_i(x) \\ C_i(x) &:= c_i\dot{\varepsilon}_i(x). \end{aligned}$$

Again, it is assumed that u_i has been specified, so that for example $\dot{\varepsilon}_i(x)$ is itself a memoryless nonlinear function. Notice also that this model specifies the statically conservative scalar force as

$$\phi_i(\ell_i) := A_i(B_i(\ell_i)). \tag{8.127}$$

Differentiability and hybrid behavior

The ramp function is continuous, though not continuously differentiable,

$$\frac{d}{dx}\left(xH(x)\right) = H(x) \quad \forall x \neq 0.$$

It may first seem that using $xH(x) \in \mathcal{C}^0$ as part of Φ_i might introduce theoretical complications with differentiability. However, the augmented potential energy framework offers a way around this problem, so that the slack-cable robot can be analyzed without resorting to hybrid systems analysis techniques.

First recall that as discussed in Sec. 3.3.2, a (generalized) force will not need to be a $\mathcal{C}^1$ function for analyzing the energy exchange in the system. It is instead the line integral of that function, U^f, that must be in $\mathcal{C}^1$, only requiring that its gradient $\nabla U^f = \mathbf{f}$ be continuous. Issues of differentiability only arise when analyzing U^f for a strict minimum. The differentiability requirement specifically arises with ϕ_i for positive eigenvalues of $\mathbf{H}(U^f)$ as per Lemma 8.4.0.4.

Since the differentiability requirement is only needed for the minimum of U^f, then the requirement only applies to the static (no velocity) model ϕ_i, not the dynamic model with Φ_i. Since the proof for Lemma 8.4.0.4 only considers a subset $\mathcal{B}$ of the state space, then a controller could be chosen so that ϕ_i is always differentiable in $\mathcal{B}$. To do so, note that the ramp function is always differentiable for its argument $x > 0$, corresponding to a pretensioned cable. Equivalently: though a cable may become slack when in dynamic motion (e.g., $H(0)$ will be relevant), if it remains *pretensioned* in $\mathcal{B}$, the proof holds in $\mathcal{B}$.

The proposed framework in this chapter can therefore elegantly deal with hybrid system behavior:

If a dynamical system is hybrid in a region of its configuration space, but has a static system model (no velocity) that is not hybrid in that same region, an augmented potential energy analysis could show stability without analyzing the hybrid phenomena.

That said, doing so may be challenging in general. The slack cable model is a special case, where the hybrid behavior can be expressed in terms of continuous, piecewise differentiable functions. More work is needed to properly examine the implications here.

8.6.2 Asymptotic Stability of the Cable-Driven Robot with Slack Cables

Having specified the cable model Φ_i and its statically conservative component ϕ_i as per eqns. (8.126)-(8.127), conditions can be given on the nonlinear functions A_i, B_i, and C_i for asymptotic stability of the cable-driven robot around a setpoint $\bar{\mathbf{r}}$. The following gives one proof where the loop is assumed to have been closed (i.e., $u_i(\ell_i)$ specified.) This is a restrictive proof, but is convenient for distributed control.

Proposition 8.6.1. *Stability of the Cable-Driven Robot with Slack Cables via the Filtered Sum Model.*

Consider a point mass model of a cable-driven robot (Fig. 8.1). Specifically, with particle position $\mathbf{r}$ *and velocity* $\mathbf{v}$*, and with a set of cables* $\boldsymbol{\ell}_i = \mathbf{r} - \mathbf{b}_i$ *with anchor points* $\mathbf{b}_i$*, the robot's dynamics are described by Lagrange's equations (8.57) rewritten here as*

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \left(\frac{\partial L}{\partial q_j}\right) = -\sum_{i=1}^{s} \mathbf{F}_i \cdot \frac{\partial \mathbf{r}}{\partial q_j} \qquad j = 1 \ldots 3,$$

where the force from cable i *is*

$$\mathbf{F}_i(\mathbf{r}, \mathbf{v}) := \Phi_i(\ell_i, \dot{\ell}_i)\hat{\boldsymbol{\ell}}_i.$$

Let these forces be described by the filtered-sum cable model with its corresponding statically conservative component,

$$\Phi_i(\ell_i, \dot{\ell}_i) := A_i(B_i(\ell_i) + C_i(\dot{\ell}_i)), \qquad \phi_i(\ell_i) := A_i(B_i(\ell_i)),$$

where A_i*,* B_i*, and* C_i *are memoryless nonlinear functions.*

Consider also a set of points $\mathcal{B} \subseteq \mathbb{E}^3$ *in the particle's configuration space. If* $\mathcal{B}$ *does not include the anchor points* $\mathbf{b}_i$ *of any cable, and the following conditions hold:*

1. *The system's static force balance holds at a point* $\mathbf{r} = \bar{\mathbf{r}} \in \mathcal{B}$*, i.e.*

$$\nabla \bar{U}(\mathbf{r})|_{\mathbf{r}=\bar{\mathbf{r}}} = \mathbf{0},$$

2. *The region* $\mathcal{B}$ *is defined such that, in* $\mathcal{B}$*, there always exists one cable whose length is increasing when the particle is in motion:*

$$\forall\, \mathbf{r} \in \mathcal{B} \text{ and } \mathbf{v} \neq \mathbf{0}, \quad \exists\, \dot{\ell}_j > 0$$

3. *For all cables* i*, the function* C_i *is input strictly passive, i.e. combining Defn. 3.2.3 and eqn. (3.38),*

$$xC_i(x) \geq \gamma_i^2 x^2, \qquad \forall\, x,\ \forall i, \quad \gamma_i \neq 0 \in \mathbb{R}$$

4. *For all cables* i*, the function* A_i *is weakly increasing over its entire domain,*

$$x_1 \geq x_2 \quad \Rightarrow \quad A_i(x_1) \geq A_i(x_2), \qquad \forall\, x_1, x_2,\ \forall i,$$

5. *For all cables* i*, the functions* A_i *and* B_i *are differentiable, positive, and strictly increasing for positive argument:*

$$A_i(x) > 0,\ \ B_i(x) > 0 \qquad A_i'(x) > 0,\ \ B_i'(x) > 0 \qquad \forall x > 0,\ \ \forall i,$$

then the point $\bar{\mathbf{r}}$ is a locally asymptotically stable equilibrium point in $\mathcal{B}$. If in addition,

6. *For all cables i, the functions A_i and B_i are both unbounded at infinity, i.e.*

$$\lim_{x\to\infty} A_i(x) = \infty, \quad \lim_{x\to\infty} B_i(x) = \infty, \quad \forall i,$$

then the point $\bar{\mathbf{r}}$ is asymptotically stable in all of $\mathcal{B}$.

Proof. This proof uses the conditions of Corollary 8.5.1.1 and in turn Prop. 8.5.1. The first assumption above directly gives the existence of the equilibrium needed for that proposition. Since the set $\mathcal{B}$ does not contain any anchor points, then all lengths are positive, $\ell_i > 0 \; \forall i$, and the assumptions give

$$\ell_i > 0 \;\; \Rightarrow \;\; B_i(\ell_i) > 0 \;\; \Rightarrow \;\; A_i(B_i(\ell_i)) > 0 \;\; \Rightarrow \;\; \phi_i(\ell_i) > 0 \;\; \forall i, \quad \forall \mathbf{r} \in \mathcal{B},$$

then via the chain rule and the assumptions on the scalar functions' derivatives,

$$\ell_i > 0 \;\; \Rightarrow \;\; \frac{\partial B_i}{\partial \ell_i} > 0 \;\; \Rightarrow \;\; \frac{\partial A_i}{\partial B_i}\frac{\partial B_i}{\partial \ell_i} > 0 \;\; \Rightarrow \;\; \frac{\partial \phi_i}{\partial \ell_i} > 0 \;\; \forall i, \quad \forall \mathbf{r} \in \mathcal{B},$$

meeting the second and third conditions of Prop. 8.5.1.

Next, consider the scalar force difference for a cable, written as a memoryless function

$$g_i(x, y) = A_i(B_i(x) + C_i(y)) - A_i(B_i(x)).$$

Show that $y g_i(x, y) \geq 0 \;\; \forall i, x, y$. Consider first the case that $y > 0$. Since C_i is passive,

$$C_i(y) \geq 0 \tag{8.128}$$
$$B_i(x) + C_i(y) \geq B_i(x), \tag{8.129}$$

for all x. Since A_i is weakly increasing for all arguments,

$$A_i(B_i(x) + C_i(y)) \geq A_i(B_i(x)) \tag{8.130}$$
$$A_i(B_i(x) + C_i(y)) - A_i(B_i(x)) \geq 0 \tag{8.131}$$
$$g_i(x, y) \geq 0, \tag{8.132}$$

and since $y > 0$ is being considered,

$$\Rightarrow y g_i(x, y) \geq 0.$$

Next, examine the case where $y < 0$. Again since C_i is passive,

$$C_i(y) \leq 0 \tag{8.133}$$
$$B_i(x) + C_i(y) \leq B_i(x) \tag{8.134}$$
$$A_i(B_i(x) + C_i(y)) \leq A_i(B_i(x)) \tag{8.135}$$
$$A_i(B_i(x) + C_i(y)) - A_i(B_i(x)) \leq 0 \tag{8.136}$$
$$g_i(x, y) \leq 0, \tag{8.137}$$

and since $y < 0$ in this case,

$$\Rightarrow yg_i(x, y) \geq 0.$$

It is clear from inspection that $y = 0 \;\Rightarrow\; yg_i(x, y) = 0$. Therefore, the dissipativity condition holds on g_i,

$$yg_i(x, y) \geq 0 \quad \forall\, x, y,\ \forall i,$$

and condition 4 from Prop. 8.5.1 is satisfied.

Finally, show the strengthened inequality: $yg_i(x, y) > 0 \quad \forall x > 0, \forall y > 0, \forall i$. This considers only $y > 0$ and $x > 0$. With the input strict passivity of C_i,

$$C_i(y) \geq \gamma_i^2 y > 0 \tag{8.138}$$
$$B_i(x) + C_i(y) \geq B_i(x) + \gamma_i^2 y > B_i(x). \tag{8.139}$$

Also, since the conditions on B_i give that

$$x > 0 \quad \Rightarrow \quad B_i(x) > 0,$$

then eqn. (8.139) is strict:

$$B_i(x) + C_i(y) \geq B_i(x) + \gamma_i^2 y > B_i(x) > 0 \tag{8.140}$$

The assumptions give that A_i is strictly increasing for strictly positive arguments, so applying A_i to the above does not affect the inequality:

$$A_i(B_i(x) + C_i(y)) \geq A_i(B_i(x) + \gamma_i^2 y) > A_i(B_i(x)) \tag{8.141}$$
$$A_i(B_i(x) + C_i(y)) - A_i(B_i(x)) \geq A_i(B_i(x) + \gamma_i^2 y) - A_i(B_i(x)) > 0 \tag{8.142}$$
$$g_i(x, y) \geq A_i(B_i(x) + \gamma_i^2 y) - A_i(B_i(x)) > 0, \tag{8.143}$$

and with $y > 0$ gives the desired strictness condition on the cable's scalar force difference:

$$yg_i(x, y) > 0 \qquad \forall x > 0,\ \forall y > 0,\ \forall i.$$

The final condition on Corollary 8.5.1.1 is met, so $\bar{\mathbf{r}}$ is a locally asymptotically stable equilibrium point in $\mathcal{B}$.

If assumption 6 also holds, since $\phi_i(\ell_i) = A_i(B_i(\ell_i))$,

$$\lim_{x\to\infty} A_i(x) = \infty, \quad \lim_{x\to\infty} B_i(x) = \infty \qquad \Rightarrow \lim_{x\to\infty} A_i(B_i(x)) = \infty,$$

meeting the radial unboundedness requirement for Prop. 8.5.1 and extending the region of asymptotic stability to all of $\mathcal{B}$.

□

8.6.3 The P+ Energy-Shaping Controller for Slack-Cable Robots

Given the above proof, the cable model with control inputs, eqn. (8.123) or (8.124) can finally be considered. This subsection establishes three results, in order.

First, a set of conditions on possible output feedback controllers $u_i(\ell_i)$ are given. With those conditions in hand, a specific set of controllers are proposed. These will be called "P+" controllers, by analogy with the "proportional plus gravity compensation" approach for a robotic arm applied instead as "proportional plus pretensioning" for the cable-driven robot. Finally, this subsection also includes a short note about the region $\mathcal{B}$ where the proof holds with respect to the actuator saturation issues described with the model-predictive controllers in Chap. 7.

Conditions on controllers for asymptotic stability

Corollary 8.6.1.1. *Conditions on control inputs u_i for asymptotic stability of the closed-loop filtered-sum model.*

Consider the cable-driven robot discussed in Prop. 8.6.1, with the corresponding region of the state space $\mathcal{B}$. Let the cables have the filtered sum model that incorporates cable slackness, eqn. (8.123), i.e.,

$$A_i(x) := xH(x), \qquad B_i(x) := k_i\varepsilon_i(x), \qquad C_i(x) := c_i x,$$

where ε_i is the cable stretch for cable i, with an output feedback controller specified as u_i:

$$\varepsilon_i(\ell_i) = \ell_i - u_i(\ell_i).$$

If the following hold for the chosen controllers $u_i(\ell_i)$:

1. *The controllers apply an equilibrium control input $\bar{u}_i = u_i(\bar{\ell}_i(\bar{\mathbf{r}}))$ at the desired equilibrium point $\bar{\mathbf{r}}$, such that $\bar{u}_i$ satisfy the static equilibrium force balance and the system is pretensioned at equilibrium,*

$$\nabla \bar{U}|_{\mathbf{r}=\bar{\mathbf{r}}} = \mathbf{0}, \qquad \bar{u}_i < \bar{\ell}_i(\bar{\mathbf{r}})),$$

2. *The controllers keep the system pretensioned,*

$$u_i(\ell_i) < \ell_i \qquad \forall \ell_i > 0,$$

3. *The controllers' derivatives with respect to their input satisfy*

$$\frac{du_i}{d\ell_i} < 1 \qquad \forall \ell_i > 0,$$

then Prop. 8.6.1 holds and the point $\bar{\mathbf{r}}$ is locally asymptotically stable in $\mathcal{B}$ defined in that proposition. If also:

4. *The controllers apply infinite stretch at an infinite length of the cable,*

$$\lim_{\ell_i \to \infty} (\ell_i - u_i(\ell_i)) = \infty,$$

then the point $\bar{\mathbf{r}}$ is asymptotically stable in all of $\mathcal{B}$.

Finally, if instead the filtered sum model instead uses the time derivative of stretch for the damping term as per eqn. (8.124),

$$C_i(x) = c_i \dot{\varepsilon}_i(x),$$

then $\bar{\mathbf{r}}$ is still (locally) asymptotically stable in $\mathcal{B}$ under the additional condition that the controller guarantees

$$\mathrm{sgn}(\dot{\ell}_i) = \mathrm{sgn}(\dot{\varepsilon}_i(\dot{\ell}_i)) \qquad \forall \dot{\ell}_i.$$

Proof. Assumption 1 gives the first condition for Prop. 8.6.1 by inspection. Similarly, all conditions on A_i are given by inspection:

$$x_1 \geq x_2 \quad \Rightarrow \quad x_1 H(x_1) \geq x_2 H(x_2) \qquad \forall x_1, x_2,$$

$$x > 0 \quad \Rightarrow \quad A_i(x) = x > 0, \qquad \frac{dA_i}{dx} = 1 > 0,$$

$$\lim_{x \to \infty} x H(x) = \infty.$$

And, the function $C_i(x)$ is input strictly passive, also again by inspection, since it is linear: let $\gamma_i^2 = c_i$.

Then, consider the conditions on B_i. Expanding the definition, with the spring constant $k_i > 0$,

$$B_i(\ell_i) = k_i \varepsilon_i(\ell_i) = k_i(\ell_i - u_i(\ell_i)).$$

Therefore, under the pretensioning assumption,

$$\begin{aligned} u_i(\ell_i) < \ell_i \quad \forall \ell_i > 0 \quad &\Rightarrow \quad \ell_i - u_i(\ell_i) > 0, && \forall \ell_i > 0 \\ &\Rightarrow \quad k_i(\ell_i - u_i(\ell_i)) > 0, && \forall \ell_i > 0 \\ &\Rightarrow \quad B_i(\ell_i) > 0, && \forall \ell_i > 0. \end{aligned}$$

Additionally, the derivative of B_i is

$$B_i'(\ell_i) = k_i \left(1 - \frac{du_i}{d\ell_i}\right),$$

so incorporating the assumption on the derivative of u_i,

$$\begin{aligned} \frac{du_i}{d\ell_i} < 1 \quad \forall \ell_i > 0 \quad &\Rightarrow \quad \left(1 - \frac{du_i}{d\ell_i}\right) > 0 && \forall \ell_i > 0, \\ &\Rightarrow k_i \left(1 - \frac{du_i}{d\ell_i}\right) > 0 && \forall \ell_i > 0, \\ &\Rightarrow B_i'(\ell_i) > 0 && \forall \ell_i > 0. \end{aligned}$$

These meet all the conditions of Prop. 8.6.1, so $\bar{\mathbf{r}}$ is a locally asymptotically stable equilibrium point in $\mathcal{B}$. Incorporating the assumption on the applied stretch at infinite length gives the unboundedness requirement on B_i for Prop. 8.6.1:

$$\lim_{\ell_i \to \infty} (\ell_i - u_i(\ell_i)) = \infty \qquad \Rightarrow \qquad \lim_{\ell \to \infty} B_i(\ell_i) = \infty,$$

and so $\bar{\mathbf{r}}$ is asymptotically stability in all of $\mathcal{B}$.

Finally, consider the case that the time derivative of stretch is used for the damping term, with the damping constant $c_i \in \mathbb{R} > 0$,

$$C_i(\dot{\ell}_i) = c_i \dot{\varepsilon}_i(\dot{\ell}_i).$$

If $\operatorname{sgn}(\dot{\ell}_i) = \operatorname{sgn}(\dot{\varepsilon}_i(\dot{\ell}_i))$, $\forall \dot{\ell}_i$, then C_i will be input strictly passive because

$$\begin{aligned} \dot{\ell}_i > 0 \quad &\Rightarrow \quad \dot{\varepsilon}_i(\dot{\ell}_i) > 0 \quad \Rightarrow \quad \dot{\ell}_i \dot{\varepsilon}_i(\dot{\ell}_i) > 0, \\ \dot{\ell}_i < 0 \quad &\Rightarrow \quad \dot{\varepsilon}_i(\dot{\ell}_i) < 0 \quad \Rightarrow \quad \dot{\ell}_i \dot{\varepsilon}_i(\dot{\ell}_i) > 0, \end{aligned}$$

and therefore in either case

$$\Rightarrow \quad c_i \dot{\ell}_i \dot{\varepsilon}_i(\dot{\ell}_i) > \gamma_i^2 \dot{\ell}_i^2$$

for some (possibly small) constant γ_i. Clearly $\dot{\ell}_i = 0 \Rightarrow c_i \dot{\ell}_i \dot{\varepsilon}_i(\dot{\ell}_i) = 0$, and so

$$\dot{\ell}_i C_i(\dot{\ell}_i) \geq \gamma_i^2 \dot{\ell}_i^2 \quad \forall \dot{\ell}_i.$$

This gives the input strict passivity condition on C_i, meeting the requirements of Prop. 8.6.1.

□

Remark. These conditions on the controller are relatively unsurprising. They are similar - almost the same - as if the system did not have the slackness term, and so heavily resemble a linear spring-mass-damper network. However, this proof addresses the challenging-to-handle, nondifferentiable nonlinearity of slack cable behavior, and in doing so also addresses hybrid behavior of the system. It is only after building up all the material in this chapter that the simple proof above will hold.

Proportional-plus-pretensioning (P+) controllers

The above could be satisfied by a variety of controllers. Here, a simple choice is presented that stabilizes the system of interest. The proposed controller stabilizes both the model that incorporates length change for damping (8.123) or stretch rate for damping (8.124).

Corollary 8.6.1.2. *Asymptotic stability of the cable-driven robot with P+ controllers within the convex hull of its anchor points.*

Consider the cable-driven robot discussed in Prop. 8.6.1 and Corollary 8.6.1.1. Choose the region of interest in the robot's state space to be $\mathcal{B} = Conv(\mathbf{b}_i)$, the convex hull of the cables' anchor points, minus the anchor points themselves. Let the cables have the filtered sum model that incorporates cable slackness, eqn. (8.123), i.e.,

$$A_i(x) := xH(x), \qquad B_i(x) := k_i\varepsilon_i(x), \qquad C_i(x) := c_i x \;\text{ or }\; c_i\dot{\varepsilon}_i(x),$$

where ε_i is the cable stretch for cable i, with an output feedback controller specified as u_i:

$$\varepsilon_i(\ell_i) = \ell_i - u_i(\ell_i).$$

Assume positive spring and damping constants, $k_i > 0$ and $c_i > 0$. Designate a desired equilibrium point $\bar{\mathbf{r}} \in \mathcal{B}$. Then, choose a proportional plus pretensioning *"P+" control law for each cable:*

$$u_i(\ell_i) = \kappa_i(\ell_i - \bar{\ell}_i) + \bar{u}_i, \tag{8.144}$$

where $\bar{\ell}_i = \ell_i(\bar{\mathbf{r}})$ and $\bar{u}_i < \bar{\ell}_i$ is a pretensioned solution the system's equilibrium force balance at $\bar{\mathbf{r}}$, and the controller gains κ_i are chosen to be in the range

$$1 > \kappa_i > \frac{\bar{u}_i}{\bar{\ell}_i} > 0. \tag{8.145}$$

Then the cable-driven robot is asymptotically stable around the single equilibrium point $\bar{\mathbf{r}}$, within the convex hull of its anchor points $\mathcal{B}$.

Proof. First, it is evident from observation that the equilibrium input is applied at $\bar{\mathbf{r}}$, since

$$u_i(\bar{\ell}_i) = \kappa_i(\bar{\ell}_i - \bar{\ell}_i) + \bar{u}_i = \bar{u}_i.$$

Then, expanding the control law,

$$u_i(\ell_i) = \kappa_i \ell_i - \kappa_i \bar{\ell}_i + \bar{u}_i.$$

The amount of cable stretch applied by the controller is therefore

$$\varepsilon_i(\ell_i) = \ell_i - u_i(\ell_i) = \ell_i - \kappa_i \ell_i + \kappa_i \bar{\ell}_i - \bar{u}_i.$$

Since

$$\kappa_i > \frac{\bar{u}_i}{\bar{\ell}_i} \quad \Rightarrow \quad \kappa_i \bar{\ell}_i - \bar{u}_i > 0,$$

and recalling that length is always positive in this choice of $\mathcal{B}$,

$$\kappa_i < 1, \quad \ell_i > 0 \quad \Rightarrow \quad \ell_i > \kappa_i \ell_i \quad \Rightarrow \quad \ell_i - \kappa_i \ell_i > 0,$$

$$\ell_i - \kappa_i \ell_i + \kappa_i \bar{\ell}_i - \bar{u}_i > 0 \quad \Rightarrow \quad \ell_i > u_i(\ell_i),$$

satisfying the pretensioning requirement. Similarly,

$$\frac{du_i}{d\ell_i} = \kappa_i < 1,$$

giving the condition on the controller's derivative and satisfying the local asymptotic stability requirements for Corollary 8.6.1.1 for the model with $C_i(\dot{\ell}_i) = c_i \dot{\ell}_i$. Noting also the limit of the cable's stretch is

$$\lim_{\ell_i \to \infty} (\ell_i - u_i(\ell_i)) = \lim_{\ell_i \to \infty} (1 - \kappa_i)\ell_i = \infty,$$

since $(1 - \kappa_i) > 0$, then $\bar{\mathbf{r}}$ is an asymptotically stable equilibrium in all of $\mathcal{B}$.

Consider the case when $C_i(\dot{\ell}_i) = c_i \dot{\varepsilon}_i(\dot{\ell}_i)$. The closed-loop stretch rate is

$$\dot{\varepsilon}_i(\dot{\ell}_i) = \dot{\ell}_i - \dot{u}_i(\dot{\ell}_i) = (1 - \kappa_i)\dot{\ell}_i,$$

and since $(1 - \kappa_i) > 0$, so $\mathrm{sgn}(\dot{\varepsilon}_i) = \mathrm{sgn}(\dot{\ell}_i)$ and Corollary 8.6.1.1 holds for this model as well.

□

Remark. As mentioned in Sec. 8.4.2, finding a pretensioned $\bar{u}_i$ can be done via the force density method (inverse statics optimization) discussed in Chap. 5. The simulations below calculate a set of $\bar{u}_i$ in this way.

P+ as energy shaping

The use of the term *energy shaping* arises from prior work ([144], etc.), where the systems' energy is *shaped* to have a minimum at the desired equilibrium point. Here, such shaping corresponds to choosing $\bar{u}_i$ at the desired $\bar{\mathbf{r}}$, then also ensuring that the augmented potential energy is strictly convex.

However, a notable difference between this proposed controller and that of prior work is the lack of a derivative term $\dot{\ell}_i$ in feedback. When a system does not naturally have damping or energy dissipation, such a "D" term is required. The resulting "PD+" controller is ubiquitous for that reason. In contrast, here, the force difference g_i includes that damping. The P+ controllers in this section take advantage of the natural energy dissipation in the system so that no derivative control effort is required. Consequently, another interpretation of the augmented potential energy framework is the exploiting of the dissipation already present in the system.

Control input saturation limits the region of asymptotic stability

A final note on this controller concerns input saturation. Motivated by the mechanical system that may apply u_i, for example a motor with a spool that retracts or extends a cable, control inputs $u_i < 0$ are physically unrealistic. In other words - cables cannot have negative rest length. It is therefore worth considering the regions in which the controller above may saturate. The following corollary derives the region in which the proof above holds, given the controller saturation condition $u_i \geq 0$.

Corollary 8.6.1.3. *Asymptotic stability region for the cable-driven robot with slack cables, considering controller saturation.*

Consider a system which satisfies Corollary 8.6.1.2. If the controller saturation condition

$$u_i \geq 0$$

is applied, i.e. that cables cannot have negative rest lengths, the region of provable asymptotic stability is reduced to

$$\mathcal{B}^* = \mathcal{B} \setminus \left(\bigcup_{i=1}^{s} \mathcal{R}_i \right), \tag{8.146}$$

where $\mathcal{R}_i$ is an open ball around anchor point i with radius r_i, given by

$$\mathcal{R}_i = \{\mathbf{r} : \ell_i(\mathbf{r}) < r_i\}, \quad r_i = \bar{\ell}_i - \frac{\bar{u}_i}{\kappa_i}. \tag{8.147}$$

In other words, the region of asymptotic stability is the convex hull of the anchor points, minus a ball around each anchor point where the controller for that anchor's cable would saturate.

Proof. The control law would apply a negative u_i when

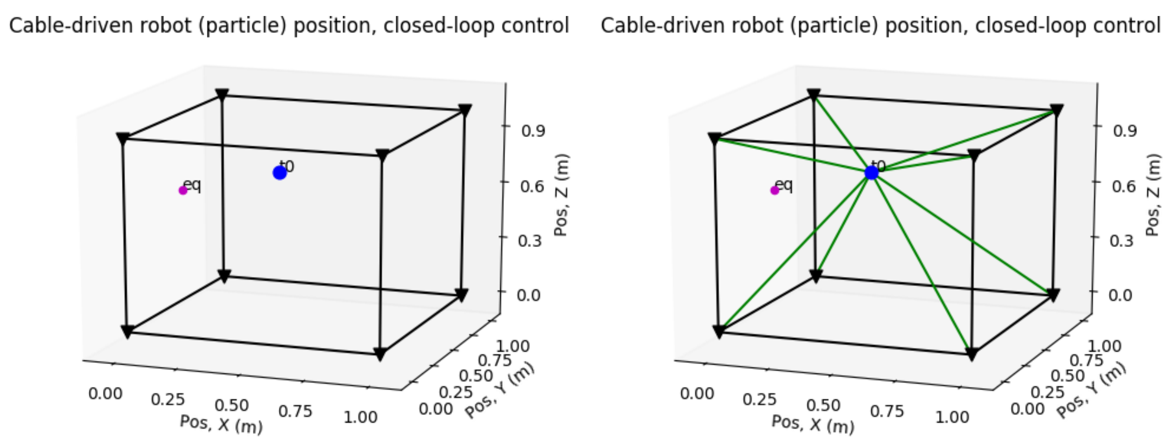
$$\begin{aligned} u_i &< 0 \\ \kappa_i(\ell_i - \bar{\ell}_i) + \bar{u}_i &< 0 \\ \kappa_i \ell_i &< \kappa_i \bar{\ell}_i - \bar{u}_i \\ \ell_i &< \bar{\ell}_i - \frac{\bar{u}_i}{\kappa_i} \\ \ell_i &< r_i. \end{aligned}$$

In the region where $\ell_i < r_i$, assume the controller instead applies $u_i = 0$ as a saturated input. In these cases, the condition of a strictly increasing ϕ_i is lost. Removing the corresponding points for each cable from the convex hull of anchors gives the set defined in eqn. (8.146).

□

8.6.4 Simulations of the P+ Controller on a Cable-Driven Robot

Simulations were performed that verify the proposed controller in its associated region of asymptotic stability (Fig. 8.3.) The following subsection presents the example setup, a set of results with a variety of initial conditions, and a numerical analysis of the Lyapunov candidate function $V(\cdot)$ that verifies the stability properties.



(a) Bounding box for the cable-driven robot simulations.

(b) Bounding box with cables, as the convex hull of anchor points.

Figure 8.3: Simulation setup for the cable-driven robot with slack cables, example initial condition (blue dot). The robot is placed within a cube with eight connected cables (one from each vertex), and a desired equilibrium condition (purple dot) is specified.

Cable-driven robot simulation setup

Corollary 8.5.1.1 designates that the stability proof holds when the robot is within the convex hull of its anchor points. A common layout for cable anchors is a box or a cube, such as is used with pick-and-place machines [113, 112]. Simulations in this section therefore use a cube, with eight cables attached to the robot (one from each corner of the cube.) Fig. 8.3 shows this setup, with the convex hull (left) and the robot in an initial pose with cables attached (right.) The cube is 1 meter in each dimension, and the robot has a mass of 4 kg. Cable spring and damping constants k_i and c_i are varied among all cables, and $\bar{u}_i$ for the equilibrium point is calculated using the force density method from Chap. 5. Controller gains κ_i were chosen within the range specified by Corollary 8.6.1.2.

Cable-driven robot simulation results

A variety of initial conditions were tested, where $\mathbf{r}_0 \in \text{Conv}(\mathbf{b}_i)$ and $\mathbf{v}_0 \neq \mathbf{0}$. All showed the robot stabilize to the desired equilibrium point.

Fig. 8.4 shows three points along the resulting trajectory in the simulation from the initial point in Fig. 8.3b. An initial velocity was chosen arbitrarily. At various points, cables are tensioned (green) or slack (red), highlighting the hybrid behavior of the system's equations of motion.

In all simulations, initial conditions were chosen such that the robot does not exit the cube due to an initial velocity. As emphasized before, $\mathcal{B} = \text{Conv}(\mathbf{b}_i)$ is not an invariant set, but instead a region where the asymptotic stability proof holds. If the particle exists the cube, the asymptotic behavior is reduced to stability in the sense of Lya-

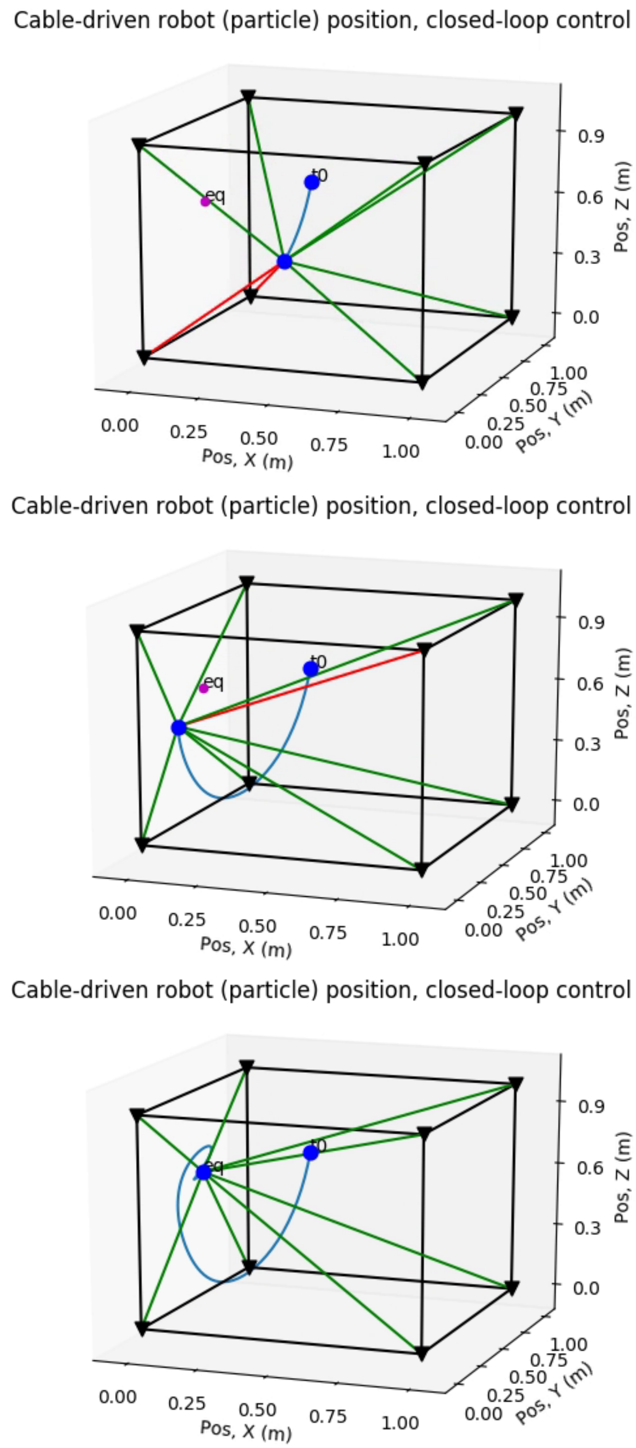
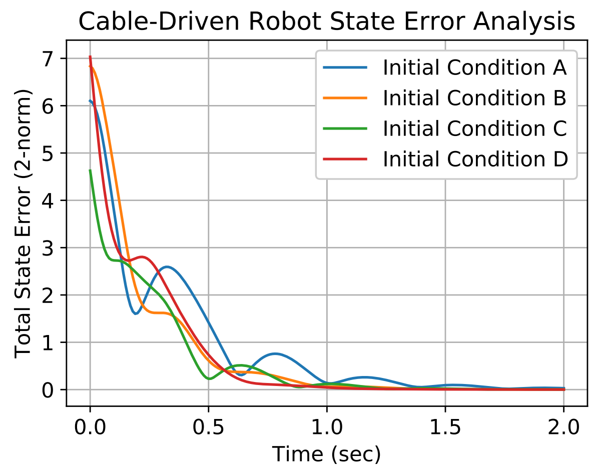


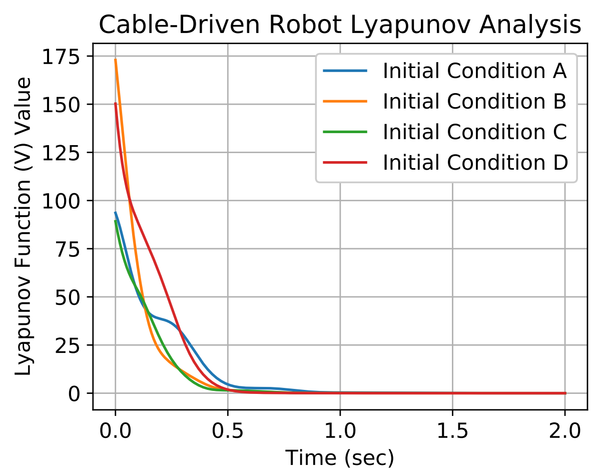
Figure 8.4: Three timepoints of the simulation example for one initial condition of the cable-driven robot.

punov, since there may not be any extending cables for certain velocities of the particle.

Fig. 8.5a shows the total error in the system states for four initial conditions. In all cases, the error converges to zero. It is also evident from this plot that the system is not necessarily exponentially stable; future work may derive a different controller (or set of controller gains) that may make the system overdamped.



(a) State error analysis (2-norm) for the cable-driven robot simulations under a variety of initial conditions. All converge to zero.



(b) Analysis of the Lyapunov candidate for the corresponding tests in Fig. 8.5a. All are descresent.

Figure 8.5: Data analysis for the cable-driven robot simulation tests.

Augmented potential energy and Lyapunov analysis with slack cables

An analysis of the Lyapunov candidate $V(\cdot)$ was also performed for each of the simulation tests. To do so, the statically conservative potential U_i^f for each cable was required, and is derived here. Although $\bar{U}$ and in turn U_i^f are not needed for the stability proof, they are needed here to confirm the results.

From Prop. 8.6.1, the statically conservative scalar force for this cable model is

$$\phi_i(\ell_i) = A_i(B_i(\ell_i)),$$

where

$$A_i(x) = xH(x), \quad B_i(x) = k_i(x - u(x)).$$

The statically conservative potential (from Claim 8.3.1) is then

$$\begin{aligned} U_i^f(\ell_i(\mathbf{r})) &= \int_a^{\ell_i} \phi_i(\tau) d\tau \\ &= \int_a^{\ell_i} A_i(B_i(\tau)) d\tau. \end{aligned}$$

It was proven above that $B_i(\ell_i) \geq 0$ always, and so the Heaviside step function can be dropped,

$$
\begin{aligned}
U_i^f &= \int_a^{\ell_i} B_i(\tau) d\tau \\
&= \int_a^{\ell_i} k_i(\tau(1-\kappa_i) + \kappa_i \bar{\ell}_i - \bar{u}_i) d\tau \\
&= \frac{1}{2} k_i \alpha_i \ell_i^2 + k_i \beta_i \ell_i,
\end{aligned}
$$

where the constants have been defined as

$$
\alpha_i := (1-\kappa_i), \quad \beta_i := \kappa_i \bar{\ell}_i - \bar{u}_i.
$$

This is not the same potential energy as is usually defined for a spring, which would have integrated over ε_i as opposed to ℓ_i. This observation highlights one of the advantageous properties of the approach in this section: it does not matter which U_i^f is chosen, since there are infinite many where $\nabla U_i^f = \mathbf{f}_i$. The above physically-unusual U_i^f is equally valid for analysis.

Fig. 8.5b shows the time series of $V(\cdot)$, using the derived U_i^f for each cable, over each simulation test. A numerical analysis confirmed that $V(\cdot)$ decreases with time for all tests. In each of the curves in Fig. 8.5b, there are smooth 'kinks' that correspond to cables becoming slack or re-tensioned. One interpretation is that the transition from slack-to-tensioned appears as a force impulse: in this case, acceleration may not be continuously differentiable, but velocity and position are at least in $\mathcal{C}^1$.

8.7 Conclusion

The control framework above has a variety of notable qualities that make it attractive for practical use. In comparison to alternative nonlinear control techniques such as feedback linearization, or more traditional passivity-based control, this energy-shaping approach:

- did not require a full derivation of the system's equations of motion,
- does not require inverting the dynamics of the system, and
- did not require deriving the exact equation of the Lyapunov candidate itself (though this was done in one case for verification purposes.)

The proposed P+ controller in particular also:

- has stabilizing control gains within a range defined by the system model's parameters, and is therefore somewhat robust to modeling error,

- is a set of distributed, output-feedback laws, requiring only the most basic sensors for feedback (length of each cable), without requiring state estimation,
- consists of scalar computations, lending itself readily to real-time control, and
- and has an intuitive interpretation of the controller gains and region of stability, leading to a flexible and intuitive tuning procedure.

Tuning and Robustness to Modeling Error

The proposed P+ controller for cable-driven robots will exhibit some robustness to modeling error. If various constants change (such as k_i, c_i, m, etc.), it is clear that the system's stability will hold if κ_i remains within the correct range. However, the equilibrium point will change.

The only model-dependent constants in the feedback laws u_i are the equilibrium inputs $\bar{u}_i$. These depend on the cable's corresponding potential energy U_i^f and the particle's mass m, but only to the extent that a force balance for the system holds at $\bar{\mathbf{r}}$. Variation in the model will thus change the force balance conditions, and change the $\bar{u}_i$ needed for the desired equilibrium point $\bar{\mathbf{r}}$.

However, for small changes in the modeling parameters, and if $\bar{\mathbf{r}}$ is not close to a face of $\mathcal{B}$, it is likely that the nominal $\bar{u}_i$ will give a different $\bar{\mathbf{r}}^*$. In other words - modeling error does not affect stability in most cases, just the equilibrium settling point, given the assumption that the true model does not eliminate the equilibrium entirely.

The controller tuning process depends on these same modeling parameters. The gains κ_i must be chosen from a limited range, between some amount of pretension $\bar{u}_i/\bar{\ell}_i$ and a gain of 1. However, this range can be adjusted by the pretensioning tuning constant when solving for $\bar{u}_i$ via the force density method. An increased pretension decreases $\bar{u}_i$, leading to a greater range for κ_i. The limitations on the gains correspondingly limit the transient performance of the controller.

Similarly, for the input constraints, the tuning constants and pretension chosen will affect the saturation region $u_i < 0$. Therefore, a control system designer can balance practical considerations such as transient performance versus region of stability.

Limitations of the Cable Network Model for Robots

For the proof-of-concept control in this chapter, an example cable-driven robot was modeled as a point mass. Though such an assumption is prevalent in the literature in a variety of cases, it is a significant simplification, and is not validated here on a hardware experiment. Instead, as mentioned in Sec. 8.1.5, this controller and its simulation test apply only under assumptions such as small moments of inertia of the robot, and when forces are applied close to the body's center of mass.

The test in this chapter simulates a robot with a small body but long cables. The workspace is one meter long in each dimension of the cube, much larger than the scales of the spine vertebra prototypes in Chap. 6. When the robot is physically small in comparison to a large workspace, the robot's moment balance does not contribute a significant constraint on the statics or dynamics, and can therefore be reasonably ignored [163]. This is also a reasonable assumption when a robot's moments of inertia are small enough that the rotational dynamics are "fast" in comparison to the dynamics of the robot's position in $\mathbb{E}^3$. Such is the case in this chapter, where the robot's small mass is intended to correspond to a small moment of inertia of some robot design.

In addition, a point-mass approximation can be useful for exploratory studies that then later transfer to robots with meaningful wrench balance requirements [161]. The point mass assumption has been used for other proof-of-concept work in passivity-based control, such as with quadcopters holding a suspended payload [125]. By demonstrating proof-of-concept here for a simplified system, future work would adapt the framework for rigid bodies without the limiting point-mass cable network assumptions.

Applications to Tensegrity Spines

Although this work addresses cable networks and systems of particles, the theory does not yet apply to rigid bodies. As discussed in Sec. 8.1, the work in this chapter is therefore not meant to represent tensegrity spines, but instead serves as progress towards eventual control of such structures. In order to apply the framework to tensegrity spines, either constraints must be imposed on the system of particles, or rigid bodies must instead be modeled with rotational inertia and angular velocity. To do the latter, rigid body twists may be useful to combine linear and angular motion into one signal.

Future Directions

The analysis above, including Prop. 8.2.2 up to Corollary 8.6.1.2, considers the problem of setpoint regulation assuming full knowledge of the system model. Future work will address tracking, and use adaptive control for modeling error purposes, as has been developed for passivity-based control of other robots [144]. Now that the basic framework of the augmented potential energy has been formulated, these extensions may be directly adaptable from prior work.

The P+ controller is straightforward and easy to implement, but is not optimal, and does not directly address performance requirements. Future controllers may be designed so that the region of provable asymptotic stability is also an invariant set. This could be done either by using more advanced nonlinear techniques to keep the conditions on the cables satisfied in a larger region, or prevent the particle from exiting the convex hull. Such approaches may include, for example, using passivity-based model predictive control for each u_i.

Chapter 9

Future Work

This dissertation established a set of modeling techniques, mechanical designs, and control systems for both tensegrity spines and quadruped robots that incorporated those spines as their bodies. Both static and dynamic models of the spine were derived, and an inverse statics routine calculated the tensions in the robots' cables for both design and control purposes. An initial design of a quadruped robot with a flexible tensegrity spine was simulated and prototyped, showing that the spine can control the robot's balance and lift its feet. Finally, control systems were presented for both the spine itself as well as similar robotic systems with statically conservative forces.

The research here has rich potential for future work, not only for quadruped robots with tensegrity spines, but also for soft robotic locomotion, control of soft systems, and applications to transportation. Ongoing work on the hardware for this specific quadruped robot, discussed in Chap. 6, would lead to tests of unique locomotion gaits as well as more systematic studies of robotic transportation over rough terrain. Mechanical designs with soft actuators and sensors would allow this robot to scale up, towards building soft robots large enough to move out of the lab and into the field. Control systems research on state estimation and localization for soft walking robots, such as those with tensegrity spines, would allow large-scale soft robots to function autonomously in open environments. Finally, theoretical advances in passivity-based control would allow for efficient, stabilizing, low-level controllers for soft walking robots. Such work includes a full investigation of the proposed energy-theoretic concepts in Chap. 8, as well as incorporating optimal control and predictive control with energy shaping.

9.1 Locomotion of Quadruped Robots with Tensegrity Spines

The prototypes of quadruped robots with tensegrity spines from Chap. 6 provide foundational concepts for the mechanisms and processes that would govern their locomotion.

Future work on these designs, once walking locomotion is demonstrated, would allow for a variety of investigations into walking gaits, terrain, and transportation.

9.1.1 Design for Locomotion: Belka and Future Soft Robots

As mentioned in Chap. 6, a variety of ongoing work on Belka, the successor to the robot Laika, seeks to demonstrate proof-of-concept walking locomotion. In addition to new mechanisms for the spine's rotational movement, and single-degree-of-freedom legs for the robot, there are a variety of unsolved problems with current designs. Ongoing work will seek to answer research questions concerning manufacturing, assembly, robustness, and general frameworks for developing these and other soft walking robots.

Though the designs for Laika address assembly of the robot's cables by incorporating a pretensioned elastic lattice [47], many other challenges in manufacturing and assembly have yet to be addressed. Future work would investigate manufacturing of a soft cable-driven robot's materials and actuators, and seek to develop procedural methods for generating soft walking robot designs. A variety of similar fields, such as microrobotics and bio-inspired robotics, have methods such as smart composite microstructures [14, 208] for this purpose. Soft cable-driven robots of Belka's size and composition are currently designed by hand; however, combining the statics modeling from Chap. 5 with the new manufacturing methods in Chap. 6 could allow for efficient procedurally-generated designs. Tensegrity robots, with their graph-structured statics and dynamics, would be an ideal candidate for such a system. Moving from hand-designed to procedurally-generated soft cable-driven robots would increase robustness by exploring a wide range of designs at a much faster pace.

9.1.2 Locomotion Studies over Rough Terrain

Having developed a robot such as Belka, which would be able to demonstrate walking locomotion on flat terrain, future work would then turn toward studies of locomotion over uneven or rough terrain. These environments are the initial motivation for quadruped robots with tensegrity spines, and consequently, extensive studies would quantify and draw conclusions about the use of soft spines during locomotion.

Studies of robots on uneven or rough terrain currently tend toward applications handpicked by researchers, focused on evaluating control systems in specific settings. These tests include terrain that is randomly generated [141] or randomly assembled [33, 207]. However, with soft quadrupeds such as Belka, tests would instead be focused on the different purpose of evaluating robot designs, not control systems, and with an orientation towards the eventual applications of space or disaster relief. Consequently, future work would study evaluation criteria for the performance of different robot designs in these tasks, in collaboration with NASA, for specific mission goals. Co-developing a platform similar to NIST's stepfield pallets [97] for space exploration would allow quantitative comparisons, in contrast to one-off proof-of-concept work.

The specific research questions to be answered using these terrain tests would address the core thesis of this dissertation: that of the usefulness and implications of a soft spine in a walking robot. For example, how much do soft components benefit a robot, in what settings, and with what movements? Experiments would test different goals, such as reaching an endpoint with the lowest rate of failure, following a predefined planned path, reaching a desired pose, or traversing the largest possible obstacle. Comparisons with stiff-bodied robots would facilitate complementary co-missions with different robot morphologies that make the best use of different design paradigms in different mission stages.

9.1.3 Design for Transportation of Supplies

The vast majority of research on quadruped locomotion, to date, has focused on locomotion for the sake of locomotion. Such research answers important, basic questions concerning bio-inspired robotics design, control systems, and mechanisms. However, little work has addressed the eventual uses of walking robots, such as design for transportation. Future work on soft walking robots would pursue this path, since there is much potential for these robots to overcome the locomotion challenges with rigid robots, and therefore serve as reliable forms of transportation over rough terrain.

Transporting scientific equipment (in NASA missions) or supplies (in disaster relief) requires a robot with carrying capacity. Beyond the saddle bags present in the military versions of BigDog [159], there have been few proposed ways to add a payload to a quadruped robot. Even fewer studies have examined designs or control systems for important considerations such as shocks or sensitivity of equipment. There are some initial concepts to do so for soft robots or tensegrity structures - for example, placing flexible containers of water or fuel in the empty space inside the robot's spine - but future work would develop new designs and frameworks for the explicit goals of a given use case.

Future design work of this type would involve collaborations for field tests, both when validating a robot's design as well as evaluating the human-centered design component of an eventual mission or task. This research has an ongoing collaboration with both NASA and the Field Innovation Team [65] to do these tests of transporting supplies and equipment. Designing for both disaster relief worker interaction, as well as deployment in space, would help the field of robotics progress from academic to practical implementations.

9.2 Mechanical Design for Soft Walking Robots

Building soft robots for the tasks above will require overcoming intimidating challenges in sensing, actuation, and size scales. Future work would expand beyond the principles of tensegrity structures, and develop new sensors and actuators that take advantage of the interplay between soft and hard subsystems. Robots that incorporate both soft and hard components could then be designed much larger, so that large walking soft robots could compare to the current state-of-the-art in rigid robots.

9.2.1 Soft Sensing

As evidenced by the research in this dissertation, soft robots made from tensegrity structures have simple models, with rigid body dynamics, made possible by the use of cables to hold the structures' bodies apart. Sensors for these cables' tensions, then, are a major required component for state estimation, and are also required for designs that incorporate other soft systems alongside tensegrity structures. Future work will examine a variety of sensors for this purpose, including capacitive force sensing [204], liquid metal strain sensors [152, 84], and microfluidic pressure sensors [176]. However, these designs are not currently focused on the size and force scales required for large walking quadruped robots. Therefore, research on these sensors will also include redesigns, extensions, and combinations to new regimes of much higher force loads, enabling the use of soft sensors for larger robots.

9.2.2 Soft Actuation

New actuation methods will also be needed for soft walking robots. For robots which use tensegrity structures for their soft systems, this will involve replacing the robot's cables with soft actuators. As with soft sensors, there are a variety of designs that show promise for this purpose, including soft pneumatic systems [212, 64, 164, 211], hydraulically amplified dielectric actuators [3], and shape memory alloys [91, 133]. Like current soft sensors, these actuators are also not focused on the size and force requirements for large soft quadrupeds. Future work will both apply current techniques as well as extend those to new scales.

One focus of new mechanical designs may be continuing work on twisted-helix linear actuators [213]. These designs combine the efficient size and mass properties of twisted-cable actuators with large force transmission and large displacements, making them well-situated for cable-driven robots. Incorporating these designs into soft systems would allow the use of more traditional force creation methods (e.g., DC motors) with soft components.

9.2.3 Scaling up Soft Walking Robots

Incorporating both sensors and actuators for soft walking robots at large size scales is an unsolved challenge. Current state-of-the-art quadruped designs require large motors for even moderately-sized robots such as the MIT Cheetah [174]. A significant goal of future research, then, is bridging this gap. What soft materials and designs could withstand the large loads required to transport supplies? How do soft systems produce precise movements? How does a designer create a softer version of the Boston Dynamics BigDog? The mechanics of tensegrity structures may facilitate this transition, through their intentional placement throughout sections of a robot that carry large forces. In the process of more thoroughly validating this dissertation's main thesis, new methods for large, soft designs would be made useful and available to the larger world community.

9.3 Control Systems for Soft Walking Robots

Much more work is required for reliable, robust locomotion of soft walking quadrupeds, both with tensegrity spines and with other types of soft structures. Control systems for movement on rough and uneven terrain, algorithms for state estimation and localization, and techniques to develop walking gaits will all be developed in future work.

9.3.1 Reliable Quadruped Movements on Rough Terrain

This research seeks to develop walking robots for purposes of transporting supplies or equipment in sensitive applications. Most work on control systems for quadrupeds has instead focused on fast running [93, 151] or other types of rapid movement over uneven terrain. Consequently, a focus of future work will be control systems which prioritize steady, stable, robust locomotion in extreme environments. Walking in pseudo-static locomotion may involve control for topics ranging from robot balance in three dimensions to specific state tracking in the robot's soft components. It is unknown which types of controllers would be best for these applications; however, since slower locomotion is less constrained by time requirements, optimization-based control techniques for rigid quadrupeds [23, 57] may be re-appropriated for soft systems.

9.3.2 State Estimation and Localization

Estimating a soft robot's state is much more complicated than doing so with a rigid-link robot. Research is only beginning on state estimation for control of soft robots; however, some results have been shown for other tensegrity robots, again due to their convenient graph-like rigid-body dynamics [36]. Future work would bring such algorithms to soft walking robots, so that closed-loop control could be performed in hardware. Research in localization would bring current two-dimensional mobile robot algorithms into three dimensions in extreme terrain, and solve new challenges with estimating terrain characteristics alongside the soft robot's state.

9.3.3 Gait Generation for Robots with Soft Bodies

To date, all observations on Laika's movements have been experimentally derived, using behaviors suggested by biology (Chap. 4.) However, what motions would most help such a robot? How should a robot's spine move, and what gaits are best for different situations? Recently, a variety of results have been shown for deep reinforcement learning of walking gaits for both quadruped robots [81] and tensegrity structures [117]. Future work would use these techniques in machine learning to generate gaits for walking robots with soft, flexible bodies.

9.4 Energy-based Control for High-Dimensional, Nonlinear, Hybrid Systems

Finally, in tandem with research on control for locomotion, future work will continue to investigate the energy-theoretic control concepts proposed in Chap. 8. These controllers, which provide certificates of stability, would be used alongside high-level planning algorithms for locomotion. The analysis of a system's *augmented potential energy* would have wide-reaching implications in the field of robotics, where forces are not easily decomposable into conservative and nonconservative components, and may find a variety of diverse applications.

9.4.1 Augmented Potential Energy for Control

The initial investigation into the concept of augmented potential energy, in Chap. 8, proposes a concept parallel to the energy analysis in passivity-based control. Since its inception in the 1980s [145], passivity-based controllers have been extended to adaptive control laws for rejection of modeling error, trajectory tracking, underactuated systems, and much more [144]. All these are required for the practical use of soft robots, and as such, the concepts of augmented potential energy analysis will be extended to these same goals. Many results may require new methods of analyzing energy dissipation.

In addition, analysis has yet to be performed on interconnections of systems that dissipate augmented potential energy. Future work would show how these systems would fit into the passivity theorem for interconnections [10]. There are rich commonalities between the techniques in Chap. 8 and that of equilibrium-independent passivity [86], and ongoing work would seek to combine these two concepts and compare with controllers such as those derived in [125]. The energy-theoretic concepts from this dissertation may form a counterpart to the energy analysis of rigid-link robots, but now for soft systems.

9.4.2 Optimal and Predictive Energy-Shaping Control

In addition to the use of the controllers in Chap. 8 for different stability goals, and their use in interconnections, future work will seek to combine energy shaping with optimal and predictive control for better performance. At the moment, most energy-based control systems show stability, but performance is an ongoing issue. Recent work has sought to combine passivity with (for example) optimal model-predictive control, showing better responses in test systems [158, 189]. Using predictive control for decomposed subsystems in a control problem may give high performance (similar to Chap. 7) with lower computational cost, while certifying stability for the system at large (as per Chap. 8). The resulting control systems will not only have strong applications in soft robotics, but also in general mechanical systems that are nonlinear, high-dimensional, and hybrid.

9.5 Conclusion and Future Prospects

The work in this dissertation presents a first set of modeling, design, and control frameworks for quadruped robots with tensegrity spines. Along the way, new concepts in energy and control were presented, which have future applications to many soft robots. The ongoing and future work from this dissertation will not only develop and test these robots with flexible spines in their intended environments, but will also extend to other soft robots and walking robots with soft bodies. Continued work on large robots with both soft and hard components may make quadruped locomotion both practical and efficient in harsh environments, leading the way for more scientific discoveries on foreign planets, and better disaster response on Earth.

Bibliography

[1] Evan Ackerman. Boston Dynamics' SpotMini Is All Electric, Agile, and Has a Capable Face-Arm, 2016.

[2] Evan Ackerman and Erico Guizzo. DARPA Robotics Challenge: A Compilation of Robots Falling Down - IEEE Spectrum, 2015.

[3] E. Acome, S. K. Mitchell, T. G. Morrissey, M. B. Emmett, C. Benjamin, M. King, M. Radakovitz, and C. Keplinger. Hydraulically amplified self-healing electrostatic actuators with muscle-like performance. *Science*, 359(6371):61–65, jan 2018.

[4] Aid and International Development Forum. Drone Technology Revolutionising Disaster Relief, 2016.

[5] Alin Albu-Schäffer, Christian Ott, and Gerd Hirzinger. A Unified Passivity-based Control Framework for Position, Torque and Impedance Control of Flexible Joint Robots. *The International Journal of Robotics Research*, 26(1):23–39, jan 2007.

[6] J.B. Aldrich, R.E. Skelton, and K. Kreutz-Delgado. Control synthesis for a class of light and agile robotic tensegrity structures. In *Proceedings of the 2003 American Control Conference, 2003.*, volume 6, pages 5245–5251. IEEE, 2003.

[7] Frank Allgöwer and Alex Zheng. *Nonlinear Model Predictive Control*, volume 26. Birkhäuser Basel, Basel, 2000.

[8] Aaron D. Ames, Kevin Galloway, Koushil Sreenath, and Jessy W. Grizzle. Rapidly Exponentially Stabilizing Control Lyapunov Functions and Hybrid Zero Dynamics. *IEEE Transactions on Automatic Control*, 59(4):876–891, apr 2014.

[9] William J Anderst, Emma Baillargeon, William F Donaldson Iii, Joon Y Lee, and James D Kang. Validation of a Non-Invasive Technique to Precisely Measure In Vivo Three-Dimensional Cervical Spine Movement. *Spine*, 36(6):393–400, 2011.

[10] Murat Arcak, Chris Meissen, and Andrew Packard. *Networks of Dissipative Systems*. SpringerBriefs in Electrical and Computer Engineering. Springer International Publishing, Cham, 2016.

[11] Marc Arsenault and Clement M. Gosselin. Kinematic, Static, and Dynamic Analysis of a Planar One-Degree-of-Freedom Tensegrity Mechanism. *Journal of Mechanical Design*, 127(6):1152, 2005.

[12] Marc Arsenault and Clement M. Gosselin. Kinematic, Static, and Dynamic Analysis of a Spatial Three-Degree-of-Freedom Tensegrity Mechanism. *Journal of Mechanical Design*, 128(5):1061, sep 2006.

[13] R E Arvidson, J F Bell, P Bellutta, N A Cabrol, J G Catalano, J Cohen, L S Crumpler, D J Des Marais, T A Estlin, W H Farrand, R Gellert, J A Grant, R N Greenberger, E A Guinness, K E Herkenhoff, J A Herman, K D Iagnemma, J R Johnson, G. Klingelhöfer, R Li, K A Lichtenberg, S A Maxwell, D W Ming, R V Morris, M S Rice, S W Ruff, A Shaw, K L Siebach, P A de Souza, A W Stroupe, S W Squyres, R J Sullivan, K P Talley, J A Townsend, A Wang, J R Wright, and A S Yen. Spirit Mars Rover Mission: Overview and selected results from the northern Home Plate Winter Haven to the side of Scamander crater. *Journal of Geophysical Research*, 115(E7):E00F03, sep 2010.

[14] Daniel M. Aukes, Benjamin Goldberg, Mark R. Cutkosky, and Robert J. Wood. An analytic framework for developing inherently-manufacturable pop-up laminate devices. *Smart Materials and Structures*, 23(9):094013, sep 2014.

[15] Reza Babaghasabha, Mohammad A. Khosravi, and Hamid D. Taghirad. Adaptive robust control of fully-constrained cable driven parallel robots. *Mechatronics*, 25:27–36, feb 2015.

[16] Nicola Pio Belfiore and Pasquale Simeone. Inverse kinetostatic analysis of compliant four-bar linkages. *Mechanism and Machine Theory*, 69:350–372, nov 2013.

[17] Donald C. Benson. Motion in a Central Force Field with Drag or Tangential Propulsion. *SIAM Journal on Applied Mathematics*, 42(4):738–750, aug 1982.

[18] James Bern, Pol Banzet, Roi Poranne, and Stelian Coros. Trajectory Optimization for Cable-Driven Soft Robot Locomotion. In *Proceedings of Robotics: Science and Systems*, 2019.

[19] James M. Bern, Grace Kumagai, and Stelian Coros. Fabrication, modeling, and control of plush robots. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 2017-Septe, pages 3739–3746. IEEE, sep 2017.

[20] K Berns, W Ilg, M Deck, and R Dillmann. The mammalian-like quadrupedal walking machine BISAM. In *AMC'98 - Coimbra. 1998 5th International Workshop on Advanced Motion Control. Proceedings (Cat. No.98TH8354)*, volume 1, pages 429–433. IEEE, 1998.

[21] Karsten Berns, Winfried Ilg, M. Deck, J. Albiez, and Rüdiger Dillmann. Mechanical construction and computer architecture of the four-legged walking machine BISAM. *IEEE/ASME Transactions on Mechatronics*, 4(1):32–38, 1999.

[22] Lionel Birglen and Clément M. Gosselin. Kinetostatic analysis of underactuated fingers. *IEEE Transactions on Robotics and Automation*, 20(2):211–221, 2004.

[23] Gerardo Bledt, Patrick M. Wensing, and Sangbae Kim. Policy-regularized model predictive control to stabilize diverse quadrupedal gaits for the MIT cheetah. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 2017-Septe, pages 4102–4109. IEEE, sep 2017.

[24] Thomas Bliss, Tetsuya Iwasaki, and Hilary Bart-Smith. Central Pattern Generator Control of a Tensegrity Swimmer. *IEEE/ASME Transactions on Mechatronics*, 18(2):586–597, apr 2013.

[25] Thomas Bliss, Jeffrey Werly, Tetsuya Iwasaki, and Hilary Bart-Smith. Experimental Validation of Robust Resonance Entrainment for CPG-Controlled Tensegrity Structures. *IEEE Transactions on Control Systems Technology*, 21(3):666–678, may 2013.

[26] Thomas K. Bliss, Tetsuya Iwasaki, and Hilary Bart-Smith. Resonance entrainment of tensegrity structures via CPG control. *Automatica*, 48(11):2791–2800, nov 2012.

[27] Valter Bohm, Tobias Kaufhold, Florian Schale, and Klaus Zimmermann. Spherical mobile robot based on a tensegrity structure with curved compressed members. In *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 1509–1514. IEEE, jul 2016.

[28] Francesco Borrelli. *Constrained Optimal Control of Linear and Hybrid Systems*, volume 290 of *Lecture Notes in Control and Information Sciences*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[29] Boston Dynamics. BigDog - The Most Advanced Rough-Terrain Robot on Earth, 2016.

[30] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[31] Bernard Brogliato, Ioan-Doré Landau, and Rogelio Lozano-Leal. Adaptive motion control of robot manipulators: A unified approach based on passivity. *International Journal of Robust and Nonlinear Control*, 1(3):187–202, jul 1991.

[32] Bernard Brogliato, Bernhard Maschke, Rogelio Lozano, and Olav Egeland. *Dissipative Systems Analysis and Control*. Communications and Control Engineering. Springer London, London, 2007.

[33] Jonas Buchli, Mrinal Kalakrishnan, Michael Mistry, Peter Pastor, and Stefan Schaal. Compliant quadruped locomotion over rough terrain. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 814–820. IEEE, oct 2009.

[34] BulletPhysicsEngine. http://www.bulletphysics.org/, 2013.

[35] C.R. Calladine. Buckminster Fuller's Tensegrity structures and Clerk Maxwell's rules for the construction of stiff frames. *International Journal of Solids and Structures*, 14(2):161–172, jan 1978.

[36] Ken Caluwaerts, Jonathan Bruce, Jeffrey M. Friesen, and Vytas SunSpiral. State estimation for tensegrity robots. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1860–1865. IEEE, may 2016.

[37] Ken Caluwaerts and Juan Pablo Carbajal. Energy conserving constant shape optimization of tensegrity structures. *International Journal of Solids and Structures*, 58:117–127, apr 2015.

[38] Ken Caluwaerts, Jérémie Despraz, A. Iscen, Andrew P Sabelhaus, Jonathan Bruce, Benjamin Schrauwen, and Vytas SunSpiral. Design and control of compliant tensegrity robots through simulation and hardware validation. *Journal of The Royal Society Interface*, 11(98):20140520–20140520, jul 2014.

[39] Mark Cannon. Efficient nonlinear model predictive control algorithms. *Annual Reviews in Control*, 28(2):229–237, jan 2004.

[40] James Casey. Geometrical derivation of Lagrange's equations for a system of particles. *American Journal of Physics*, 62(9):836–847, sep 1994.

[41] James Casey. On the advantages of a geometrical viewpoint in the derivation of Lagrange's equations for a rigid continuum. *Theoretical, Experimental, and Numerical Contributions to the Mechanics of Fluids and Solids; Journal of Applied Mathematics and Physics (ZAMP)*, pages 805–847, 1995.

[42] James Casey and Oliver M. O'Reilly. Geometrical Derivation of Lagrange's Equations for a System of Rigid Bodies. *Mathematics and Mechanics of Solids*, 11(4):401–422, aug 2006.

[43] R J Caverly, J R Forbes, and D Mohammadshahi. Dynamic Modeling and Passivity-Based Control of a Single Degree of Freedom Cable-Actuated System. *Control Systems Technology, IEEE Transactions on*, PP(99):1, 2014.

[44] Ryan James Caverly and James Richard Forbes. Dynamic Modeling and Noncollocated Control of a Flexible Planar Cable-Driven Manipulator. *IEEE Transactions on Robotics*, 30(6):1386–1397, dec 2014.

[45] Brian Cera and Alice M. Agogino. Multi-Cable Rolling Locomotion with Spherical Tensegrities Using Model Predictive Control and Deep Learning. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–9. IEEE, oct 2018.

[46] Bingxing Chen and Hongzhou Jiang. Swimming Performance of a Tensegrity Robotic Fish. *Soft Robotics*, 00(00), apr 2019.

[47] Lee-Huang Chen, Mallory C. Daly, Andrew P. Sabelhaus, Lara A. Janse van Vuuren, Hunter J. Garnier, Mariana I. Verdugo, Ellande Tang, Carielle U. Spangenberg, Faraz Ghahani, Adrian K. Agogino, and Alice M. Agogino. Modular Elastic Lattice Platform for Rapid Prototyping of Tensegrity Robots. In *ASME International Design Engineering Technical Conference (IDETC) 41st Mechanisms and Robotics Conference*, 2017.

[48] Lee-huang Chen, Kyunam Kim, Ellande Tang, Kevin Li, Richard House, Erik Jung, Alice M Agogino, Adrian Agogino, and Vytas SunSpiral. Soft Spherical Tensegrity Robot Design Using Rod-Centered Actuation and Control. In *ASME International Design Engineering Technical Conference (IDETC) Mechanisms and Robotics Conference*, Charlotte, NC, 2016. American Society of Mechanical Engineers.

[49] Robert Connelly and Walter Whiteley. The Stability of Tensegrity Frameworks. *International Journal of Space Structures*, 7(2):153–163, 1992.

[50] Alessandro Crespi and Auke Jan Ijspeert. Salamandra robotica: a biologically inspired amphibious robot that swims and walks. In *Artificial Life Models in Hardware*, pages 35–64. Springer, 2009.

[51] Alessandro Crespi, Konstantinos Karakasiliotis, Andre Guignard, and Auke Jan Ijspeert. Salamandra Robotica II: An Amphibious Robot to Study Salamander-Like Swimming and Walking Gaits. *IEEE Transactions on Robotics*, 29(2):308–320, apr 2013.

[52] L. Cveticanin. Some conservation laws for orbits involving variable mass and linear damping. *Journal of Guidance, Control, and Dynamics*, 17(1):209–211, jan 1994.

[53] Defense Advanced Research Projects Agency. DARPA Robotics Challenge - DRC Finals, 2015.

[54] Marc Peter Deisenroth, Gerhard Neumann, and Jan Peters. A Survey on Policy Search for Robotics. *Foundations and Trends in Robotics*, 2(1-2):1–142, 2011.

[55] D.C. Deno, R.M. Murray, K.S.J. Pister, and S.S. Sastry. Fingerlike biomechanical robots. In *Proceedings 1992 IEEE International Conference on Robotics and Automation*, pages 566–572. IEEE Comput. Soc. Press.

[56] Charles A. Desoer and M. Vidyasagar. *Feedback Systems: InputOutput Properties*. Academic Press, New York, 1975.

[57] Jared Di Carlo, Patrick M. Wensing, Benjamin Katz, Gerardo Bledt, and Sangbae Kim. Dynamic Locomotion in the MIT Cheetah 3 Through Convex Model-Predictive Control. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–9. IEEE, oct 2018.

[58] Menashe Dornay, Ferdinando A. Mussa-Ivaldi, Joseph McIntyre, and Emilio Bizzi. Stability constraints for the distributed control of motor behavior. *Neural Networks*, 6(8):1045–1059, jan 1993.

[59] Dexter Duckworth, Brandon Shrewsbury, and Robin Murphy. Run the Robot Backward: Lessons Learned in Nuclear Forensics. In *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 1–6. IEEE, oct 2013.

[60] Duy Nguyen-Tuong and J Peters. Local Gaussian process regression for real-time model-based robot control. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 380–385. IEEE, sep 2008.

[61] J Dvorak, M M Panjabi, D G Chang, R Theiler, and D Grob. Functional radiographic diagnosis of the Lumbar Spine: Flexion-Extension and Lateral Bending, 1991.

[62] Peter Eckert, Alexander Sprowitz, Hartmut Witte, and Auke Jan Ijspeert. Comparing the effect of different spine and leg designs for a small bounding quadruped robot. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3128–3133. IEEE, may 2015.

[63] P. Falcone, M. Tufo, F. Borrelli, J. Asgari, and H.E. Tseng. A linear time varying model predictive control approach to the integrated vehicle dynamics control problem in autonomous systems. *2007 46th IEEE Conference on Decision and Control*, pages 2980–2985, 2007.

[64] Wyatt Felt, Matthew A. Robertson, and Jamie Paik. Modeling vacuum bellows soft pneumatic actuators with optimal mechanical performance. *2018 IEEE International Conference on Soft Robotics, RoboSoft 2018*, pages 534–540, 2018.

[65] Field Innovation Team. About FIT — The Field Innovation Team, 2017.

[66] Tom Flemons. The Bones of Tensegrity, 2012.

[67] Vincent François-Lavet, Peter Henderson, Riashat Islam, Marc G. Bellemare, and Joelle Pineau. An Introduction to Deep Reinforcement Learning. *Foundations and Trends® in Machine Learning*, 11(3-4):219–354, 2018.

[68] Jeffrey Friesen, Alexandra Pogue, Thomas Bewley, Mauricio de Oliveira, Robert Skelton, and Vytas Sunspiral. DuCTT: A tensegrity robot for exploring duct systems. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4222–4228. IEEE, may 2014.

[69] Jeffrey M. Friesen, John L. Dean, Thomas Bewley, and Vytas Sunspiral. A Tensegrity-Inspired Compliant 3-DOF Compliant Joint. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–9. IEEE, may 2018.

[70] Jeffrey M Friesen, Paul Glick, Michael Fanton, Pavlo Manovi, Alexander Xydes, Thomas Bewley, and Vytas Sunspiral. The second generation prototype of a Duct Climbing Tensegrity robot, DuCTTv2. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2123–2128. IEEE, may 2016.

[71] Fuel Relief Fund. Fuel Relief Fund's Disaster Response, 2016.

[72] Y. Fukuoka, H. Kimura, and Avis H Cohen. Adaptive Dynamic Walking of a Quadruped Robot on Irregular Terrain Based on Biological Concepts. *The International Journal of Robotics Research*, 22(3-4):187–202, mar 2003.

[73] Y Fukuoka, H Nakamura, and H Kimura. Biologically-inspired adaptive dynamic walking of the quadruped on irregular terrain. In *Proceedings of the 1999 IEEE International Conference on Control Applications (Cat. No.99CH36328)*, volume 1, pages 490–495. IEEE, 1999.

[74] Bingtuan Gao, Honggang Song, Jianguo Zhao, Shaoxiong Guo, Lixia Sun, and Yi Tang. Inverse kinematics and workspace analysis of a cable-driven parallel robot with a spring spine. *Mechanism and Machine Theory*, 76:56–69, jun 2014.

[75] Christian Gehring, Stelian Coros, Marco Hutter, Michael Bloesch, Peter Fankhauser, Markus A. Hoepflinger, and Roland Siegwart. Towards automatic discovery of agile gaits for quadrupedal robots. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4243–4248. IEEE, may 2014.

[76] Michele Giorelli, Federico Renda, Marcello Calisti, Andrea Arienti, Gabriele Ferri, and Cecilia Laschi. Neural Network and Jacobian Method for Solving the Inverse Statics of a Cable-Driven Soft Arm with Nonconstant Curvature. *IEEE Transactions on Robotics*, 31(4):823–834, 2015.

[77] Herbert Goldstein, Charles P. Poole Jr., and John L. Safko. *Classical Mechanics.* Addison Wesley, 3rd edition, 2002.

[78] S. Gracovetsky. An hypothesis for the role of the spine in human locomotion: A challenge to current thinking. *Journal of Biomedical Engineering*, 7(3):205–216, jul 1985.

[79] W Graf, C de Waele, and P P Vidal. Functional anatomy of the head-neck movement system of quadrupedal and bipedal mammals. *Journal of Anatomy*, 186:55–74, 1995.

[80] Jerry Gray. The Rwanda Disaster: Relief, U.S. Aid Airdrop Off to a Ragged Start, 1994.

[81] Tuomas Haarnoja, Sehoon Ha, Aurick Zhou, Jie Tan, George Tucker, and Sergey Levine. Learning to Walk via Deep Reinforcement Learning. *Robotics: Science and Systems XV*, 2019.

[82] Duncan W Haldane and Ronald S Fearing. Running beyond the bio-inspired regime. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4539–4546. IEEE, may 2015.

[83] Donald M Hassler, Cary Zeitlin, R. F. Wimmer-Schweingruber, Bent Ehresmann, Scot Rafkin, Jennifer L Eigenbrode, David E Brinza, Gerald Weigle, S. Bottcher, E. Bohm, Soenke Burmeister, Jingnan Guo, J. Kohler, Cesar Martin, Guenther Reitz, Francis A Cucinotta, M.-H. Kim, David Grinspoon, Mark A Bullock, Arik Posner, J. Gomez-Elvira, Ashwin Vasavada, John P Grotzinger, M. S. Team, O. Kemppinen, D. Cremers, J. F. Bell, L. Edgar, J. Farmer, A. Godber, M. Wadhwa, D. Wellington, I. McEwan, C. Newman, M. Richardson, A. Charpentier, L. Peret, P. King, J. Blank, M. Schmidt, S. Li, R. Milliken, K. Robertson, V. Sun, M. Baker, C. Edwards, B. Ehlmann, K. Farley, J. Griffes, H. Miller, M. Newcombe, C. Pilorget, M. Rice, K. Siebach, K. Stack, E. Stolper, C. Brunet, V. Hipkin, R. Leveille, G. Marchand, P. S. Sanchez, L. Favot, G. Cody, A. Steele, L. Fluckiger, D. Lees, A. Nefian, M. Martin, M. Gailhanou, F. Westall, G. Israel, C. Agard, J. Baroukh, C. Donny, A. Gaboriaud, P. Guillemot, V. Lafaille, E. Lorigny, A. Paillet, R. Perez, M. Saccoccio, C. Yana, C. Armiens-Aparicio, J. C. Rodriguez, I. C. Blazquez, F. G. Gomez, S. Hettrich, A. L. Malvitte, M. M. Jimenez, J. Martinez-Frias, J. Martin-Soler, F. J. Martin-Torres, A. M. Jurado, L. Mora-Sotomayor, G. M. Caro, S. N. Lopez, V. Peinado-Gonzalez, J. Pla-Garcia, J. A. R. Manfredi, J. J. Romeral-Planello, S. A. S. Fuentes, E. S. Martinez, J. T. Redondo, R. Urqui-O'Callaghan, M.-P. Z. Mier, S. Chipera, J.-L. Lacour, P. Mauchien, J.-B. Sirven, H. Manning, A. Fairen, A. Hayes, J. Joseph, S. Squyres, R. Sullivan, P. Thomas, A. Dupont, A. Lundberg, N. Melikechi, A. Mezzacappa, T. Berger, D. Matthia, B. Prats, E. Atlaskin, M. Genzer, A.-M. Harri, H. Haukka, H. Kahanpaa, J. Kauhanen, O. Kemppinen, M. Paton, J. Polkko, W. Schmidt, T. Siili, C. Fabre, J. Wray, M. B. Wilhelm, F. Poitrasson, K. Patel, S. Gorevan, S. Indyk, G. Paulsen, S. Gupta, D. Bish, J. Schieber, B. Gondet, Y. Langevin, C. Geffroy, D. Baratoux, G. Berger, A. Cros, C. D'Uston, O. Forni, O. Gasnault, J. Lasue, Q.-M. Lee, S. Maurice, P.-Y. Meslin, E. Pallier, Y. Parot, P. Pinet, S. Schroder, M. Toplis, E. Lewin, W. Brunner, E. Heydari, C. Achilles, D. Oehler, B. Sutter, M. Cabane, D. Coscia, G. Israel, C. Szopa, G. Dromart, F. Robert, V. Sautter, S. Le Mouelic, N. Mangold, M. Nachon, A. Buch, F. Stalport, P. Coll, P. Francois, F. Raulin,

S. Teinturier, J. Cameron, S. Clegg, A. Cousin, D. DeLapp, R. Dingler, R. S. Jackson, S. Johnstone, N. Lanza, C. Little, T. Nelson, R. C. Wiens, R. B. Williams, A. Jones, L. Kirkland, A. Treiman, B. Baker, B. Cantor, M. Caplinger, S. Davis, B. Duston, K. Edgett, D. Fay, C. Hardgrove, D. Harker, P. Herrera, E. Jensen, M. R. Kennedy, G. Krezoski, D. Krysak, L. Lipkaman, M. Malin, E. McCartney, S. McNair, B. Nixon, L. Posiolova, M. Ravine, A. Salamon, L. Saper, K. Stoiber, K. Supulver, J. Van Beek, T. Van Beek, R. Zimdar, K. L. French, K. Iagnemma, K. Miller, R. Summons, F. Goesmann, W. Goetz, S. Hviid, M. Johnson, M. Lefavor, E. Lyness, E. Breves, M. D. Dyar, C. Fassett, D. F. Blake, T. Bristow, D. DesMarais, L. Edwards, R. Haberle, T. Hoehler, J. Hollingsworth, M. Kahre, L. Keely, C. McKay, M. B. Wilhelm, L. Bleacher, W. Brinckerhoff, D. Choi, P. Conrad, J. P. Dworkin, M. Floyd, C. Freissinet, J. Garvin, D. Glavin, D. Harpold, A. Jones, P. Mahaffy, D. K. Martin, A. McAdam, A. Pavlov, E. Raaen, M. D. Smith, J. Stern, F. Tan, M. Trainer, M. Meyer, M. Voytek, R. C. Anderson, A. Aubrey, L. W. Beegle, A. Behar, D. Blaney, F. Calef, L. Christensen, J. A. Crisp, L. DeFlores, B. Ehlmann, J. Feldman, S. Feldman, G. Flesch, J. Hurowitz, I. Jun, D. Keymeulen, J. Maki, M. Mischna, J. M. Morookian, T. Parker, B. Pavri, M. Schoppers, A. Sengstacken, J. J. Simmonds, N. Spanovich, M. d. l. T. Juarez, C. R. Webster, A. Yen, P. D. Archer, J. H. Jones, D. Ming, R. V. Morris, P. Niles, E. Rampe, T. Nolan, M. Fisk, L. Radziemski, B. Barraclough, S. Bender, D. Berman, E. N. Dobrea, R. Tokar, D. Vaniman, R. M. E. Williams, A. Yingst, K. Lewis, L. Leshin, T. Cleghorn, W. Huntress, G. Manhes, J. Hudgins, T. Olson, N. Stewart, P. Sarrazin, J. Grant, E. Vicenzi, S. A. Wilson, V. Hamilton, J. Peterson, F. Fedosov, D. Golovin, N. Karpushkina, A. Kozyrev, M. Litvak, A. Malakhov, I. Mitrofanov, M. Mokrousov, S. Nikiforov, V. Prokhorov, A. Sanin, V. Tretyakov, A. Varenikov, A. Vostrukhin, R. Kuzmin, B. Clark, M. Wolff, S. McLennan, O. Botta, D. Drake, K. Bean, M. Lemmon, S. P. Schwenzer, R. B. Anderson, K. Herkenhoff, E. M. Lee, R. Sucharski, M. A. d. P. Hernandez, J. J. B. Avalos, M. Ramos, C. Malespin, I. Plante, J.-P. Muller, R. Navarro-Gonzalez, R. Ewing, W. Boynton, R. Downs, M. Fitzgibbon, K. Harshman, S. Morrison, W. Dietrich, O. Kortmann, M. Palucis, D. Y. Sumner, A. Williams, G. Lugmair, M. A. Wilson, D. Rubin, B. Jakosky, T. Balic-Zunic, J. Frydenvang, J. K. Jensen, K. Kinch, A. Koefoed, M. B. Madsen, S. L. S. Stipp, N. Boyd, J. L. Campbell, R. Gellert, G. Perrett, I. Pradler, S. VanBommel, S. Jacob, T. Owen, S. Rowland, E. Atlaskin, H. Savijarvi, C. M. Garcia, R. Mueller-Mellin, J. C. Bridges, T. McConnochie, M. Benna, H. Franz, H. Bower, A. Brunner, H. Blau, T. Boucher, M. Carmosino, S. Atreya, H. Elliott, D. Halleaux, N. Renno, M. Wong, R. Pepin, B. Elliott, J. Spray, L. Thompson, S. Gordon, H. Newsom, A. Ollila, J. Williams, P. Vasconcelos, J. Bentz, K. Nealson, R. Popa, L. C. Kah, J. Moersch, C. Tate, M. Day, G. Kocurek, B. Hallet, R. Sletten, R. Francis, E. McCullough, E. Cloutis, I. L. ten Kate, R. Kuzmin, R. Arvidson, A. Fraeman, D. Scholes, S. Slavney, T. Stein, J. Ward, J. Berger, and J. E. Moores. Mars' Surface Radiation Environment Measured with the Mars Science Laboratory's Curiosity Rover. *Science*, 343(6169):1244797–1244797, jan 2014.

[84] Tess Hellebrekers, Kadri Bugra Ozutemiz, Jessica Yin, and Carmel Majidi. Liquid Metal-Microelectronics Integration for a Sensorized Soft Robot Skin. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5924–5929. IEEE, oct 2018.

[85] R.J. Hindle, M.J. Pearcy, A.T. Cross, and D.H.T Miller. Three-dimensional kinematics of the human back. *Clinical Biomechanics*, 5(4):218–228, 1990.

[86] George H. Hines, Murat Arcak, and Andrew K. Packard. Equilibrium-independent passivity: A new definition and numerical certification. *Automatica*, 47(9):1949–1956, sep 2011.

[87] Shigeo Hirose, Yasushi Fukuda, Kan Yoneda, Akihiko Nagakubo, Hideyuki Tsukagoshi, Keisuke Arikawa, Gen Endo, Takahiro Doi, and Ryuichi Hodoshima. Quadruped walking robots at Tokyo Institute of Technology. *IEEE Robotics & Automation Magazine*, 16(2):104–114, jun 2009.

[88] Benjamin Hockman and M. Pavone. Stochastic Motion Planning for Hopping Rovers on Small Solar System Bodies. *International Symposium on Robotics Research (ISRR)*, 2017.

[89] Benjamin J. Hockman, Andreas Frick, Robert G. Reid, Issa A.D. Nesnas, and Marco Pavone. Design, Control, and Experimentation of Internally-Actuated Rovers for the Exploration of Low-gravity Planetary Bodies. *Journal of Field Robotics*, 34(1):5–24, jan 2017.

[90] Tomislav Horvat, Konstantinos Karakasiliotis, Kamilo Melo, Laura Fleury, Robin Thandiackal, and Auke J. Ijspeert. Inverse kinematics and reflex based controller for body-limb coordination of a salamander-like robot walking on uneven terrain. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 195–201. IEEE, sep 2015.

[91] W Huang. On the selection of shape memory alloys for actuators. *Materials & Design*, 23(1):11–19, feb 2002.

[92] Dawn Hustig-Schultz, Vytas SunSpiral, and Mircea Teodorescu. Morphological design for controlled tensegrity quadruped locomotion. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4714–4719. IEEE, oct 2016.

[93] Dong Jin Hyun, Sangok Seok, Jongwoo Lee, and Sangbae Kim. High speed trot-running: Implementation of a hierarchical controller using proprioceptive impedance control on the MIT Cheetah. *The International Journal of Robotics Research*, 33(11):1417–1445, sep 2014.

[94] International Federation of Red Cross and Red Crescent Societies. World Disasters Report: Focus on ethics in aid. Technical report, IFRC, 2003.

[95] Atil Iscen, Adrian Agogino, Vytas SunSpiral, and Kagan Tumer. Flop and roll: Learning robust goal-directed locomotion for a Tensegrity Robot. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2236–2243. IEEE, sep 2014.

[96] Atil Iscen, Ken Caluwaerts, Jonathan Bruce, Adrian Agogino, Vytas SunSpiral, and Kagan Tumer. Learning Tensegrity Locomotion Using Open-Loop Control Signals and Coevolutionary Algorithms. *Artificial Life*, 21(2):119–140, may 2015.

[97] A Jacoff, A Downs, A Virts, and E Messina. Stepfield Pallets: Repeatable Terrain for Evaluating Robot Mobility. In *Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems - PerMIS '08*, page 29, New York, New York, USA, 2008. ACM Press.

[98] Lara A. Janse van Vuuren. *Design, Manufacturing, and Testing of an Actuated Leg for a Tensegrity Quadruped Robot*. M.eng, University of California, Berkeley, 2017.

[99] Jet Propulsion Laboratory. Looking Up at Mars Rover Curiosity in 'Buckskin' Selfie, 2015.

[100] Mohammad Hasan H Kani, Mostafa Derafshian, Hamed Jalaly Bidgoly, and Majid Nili Ahmadabadi. Effect of flexible spine on stability of a passive quadruped robot: Experimental results. In *2011 IEEE International Conference on Robotics and Biomimetics*, pages 2793–2798. IEEE, dec 2011.

[101] Manuel Keppler, Dominic Lakatos, Alexander Werner, Florian Loeffl, Christian Ott, and Alin Albu-Schaffer. Visco-Elastic Structure Preserving Impedance (VESπ) Control for Compliantly Actuated Robots. In *2018 European Control Conference (ECC)*, pages 255–260. IEEE, jun 2018.

[102] Mark Khazanov, Julian Jocque, and John Rieffel. Evolution of Locomotion on a Physical Tensegrity Robot. In *Artificial Life 14: Proceedings of the Fourteenth International Conference on the Synthesis and Simulation of Living Systems*, pages 232–238. The MIT Press, jul 2014.

[103] M. Khoramshahi, A. Sprowitz, A. Tuleu, M. N. Ahmadabadi, and A. J. Ijspeert. Benefits of an active spine supported bounding locomotion with a small compliant quadruped robot. In *2013 IEEE International Conference on Robotics and Automation*, pages 3329–3334. IEEE, may 2013.

[104] Mohammad A. Khosravi and Hamid D. Taghirad. Dynamic Analysis and Control of Fully-Constrained Cable Robots with Elastic Cables: Variable Stiffness Formulation.

In *Transactions of the Canadian Society for Mechanical Engineering*, volume 35, pages 161–177. 2015.

[105] Kyunam Kim, Adrian K. Agogino, Deaho Moon, Laqshya Taneja, Aliakbar Toghyan, Borna Dehghani, Vytas SunSpiral, and Alice M. Agogino. Rapid prototyping design and control of tensegrity soft robot for locomotion. In *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, pages 7–14. IEEE, dec 2014.

[106] Kyunam Kim, Adrian K. Agogino, Aliakbar Toghyan, Deaho Moon, Laqshya Taneja, and Alice M. Agogino. Robust learning of tensegrity robot control for locomotion through form-finding. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5824–5831. IEEE, sep 2015.

[107] Kyunam Kim, Lee-Huang Chen, Brian Cera, Mallory Daly, Edward Zhu, Julien Despois, Adrian K. Agogino, Vytas SunSpiral, and Alice M. Agogino. Hopping and rolling locomotion with spherical tensegrity robots. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 2016-Novem, pages 4369–4376. IEEE, oct 2016.

[108] Jonathan Kimber, Zongliang Ji, Aikaterini Petridou, Thomas Sipple, Kentaro Barhydt, James Boggs, Luke Dosiek, and John Rieffel. Low-Cost Wireless Modular Soft Tensegrity Robots. In *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, pages 88–93. IEEE, apr 2019.

[109] Yuusuke Koizumi, Mizuho Shibata, and Shinichi Hirai. Rolling tensegrity driven by pneumatic soft actuators. In *2012 IEEE International Conference on Robotics and Automation*, pages 1988–1993. IEEE, may 2012.

[110] J. Zico Kolter, Pieter Abbeel, and Andrew Y. Ng. Hierarchical apprenticeship learning with application to quadruped locomotion. *Advances in Neural Information Processing Systems 20*, 1:1–8, 2007.

[111] Erik Komendera. *A survey of the computational modeling and control of tensegrity robots*. Ph.d. preliminary exam in computational modeling of complex systems, University of Colorado, 2012.

[112] Samir Lahouar, Erika Ottaviano, Said Zeghoul, Lotfi Romdhane, and Marco Ceccarelli. Collision free path-planning for cable-driven parallel robots. *Robotics and Autonomous Systems*, 57(11):1083–1093, nov 2009.

[113] Johann Lamaury and Marc Gouttefarde. Control of a large redundantly actuated cable-suspended parallel robot. In *2013 IEEE International Conference on Robotics and Automation*, pages 4659–4664. IEEE, may 2013.

[114] Steven Lessard, Jonathan Bruce, Erik Jung, Mircea Teodorescu, Vytas SunSpiral, and Adrian Agogino. A lightweight, multi-axis compliant tensegrity joint. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 630–635. IEEE, may 2016.

[115] Zakary Littlefield, David Surovik, Massimo Vespignani, Jonathan Bruce, Weifu Wang, and Kostas E Bekris. Kinodynamic planning for spherical tensegrity locomotion with effective gait primitives. *The International Journal of Robotics Research*, may 2019.

[116] J. Lofberg. YALMIP : a toolbox for modeling and optimization in MATLAB. In *2004 IEEE International Conference on Robotics and Automation*, pages 284–289. IEEE, 2004.

[117] Jianlan Luo, Riley Edmunds, Franklin Rice, and Alice M. Agogino. Tensegrity Robot Locomotion Under Limited Sensory Inputs via Deep Reinforcement Learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6260–6267. IEEE, may 2018.

[118] John Lygeros, Claire Tomlin, and Shankar Sastry. Controllers for reachability specifications for hybrid systems. *Automatica*, 35(3):349–370, mar 1999.

[119] Anirudha Majumdar, Amir Ali Ahmadi, and Russ Tedrake. Control design along trajectories with sums of squares programming. In *2013 IEEE International Conference on Robotics and Automation*, pages 4054–4061. IEEE, may 2013.

[120] Anirudha Majumdar, Amir Ali Ahmadi, and Russ Tedrake. Control and verification of high-dimensional systems with DSOS and SDSOS programming. In *53rd IEEE Conference on Decision and Control*, pages 394–401. IEEE, dec 2014.

[121] Anirudha Majumdar and Russ Tedrake. Funnel libraries for real-time robust feedback motion planning. *International Journal of Robotics Research*, 36(8):947–982, 2017.

[122] Anirudha Majumdar, Ram Vasudevan, Mark M. Tobenkin, and Russ Tedrake. Convex optimization of nonlinear feedback controllers via occupation measures. *The International Journal of Robotics Research*, 33(9):1209–1230, aug 2014.

[123] Soroush Maleki, Atoosa Parsa, and M. Nili Ahmadabadi. Modeling, control and gait design of a quadruped robot with active spine towards energy efficiency. In *2015 3rd RSI International Conference on Robotics and Mechatronics (ICROM)*, pages 271–276. IEEE, oct 2015.

[124] Jonathan Marr, Nigel Mevana, and Holly Stein. *Tensegrity Soft Robots for NASA Missions and Disaster Relief on Earth: Tensegrity-Spine Quadrupedal Robot*. M.eng, University of California, Berkeley, 2018.

[125] Chris Meissen, Kristian Klausen, Murat Arcak, Thor I. Fossen, and Andrew Packard. Passivity-based Formation Control for UAVs with a Suspended Load. *20th IFAC World Congress*, 50(1):13150–13155, jul 2017.

[126] Kenji Miki and Katsuyoshi Tsujita. A study of the effect of structural damping on gait stability in quadrupedal locomotion using a musculoskeletal robot. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1976–1981. IEEE, oct 2012.

[127] S Mintchev, D Zappetti, J Willemin, and D Floreano. A Soft Robot for Random Exploration of Terrestrial Environments. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7492–7497. IEEE, may 2018.

[128] Josep M. Mirats Tur and Sergi Hernàndez Juan. Tensegrity frameworks: Dynamic analysis review and open problems. *Mechanism and Machine Theory*, 44(1):1–18, jan 2009.

[129] B Mirletz, In-Won Park, Thomas E Flemons, Adrian K Agogino, Roger D Quinn, and Vytas SunSpiral. Design and Control of Modular Spine-Like Tensegrity Structures. In *The 6th World Conference of the International Association for Structural Control and Monitoring (6WCSCM)*, 2014.

[130] Brian T. Mirletz, Perry Bhandal, Ryan D. Adams, Adrian K. Agogino, Roger D. Quinn, and Vytas SunSpiral. Goal-Directed CPG-Based Control for Tensegrity Spines with Many Degrees of Freedom Traversing Irregular Terrain. *Soft Robotics*, 2(4):165–176, dec 2015.

[131] Brian T. Mirletz, In-Won Park, Roger D. Quinn, and Vytas SunSpiral. Towards bridging the reality gap between tensegrity simulation and robotic hardware. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5357–5363. IEEE, sep 2015.

[132] K. Mitobe, N Mori, K. Aida, and Y Nasu. Nonlinear feedback control of a biped walking robot. In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 3, pages 2865–2870. IEEE, 1995.

[133] Jaronie Mohd Jani, Martin Leary, Aleksandar Subic, and Mark A. Gibson. A review of shape memory alloy research, applications and opportunities. *Materials & Design*, 56:1078–1113, apr 2014.

[134] E.Z. Moore, D. Campbell, F. Grimminger, and M. Buehler. Reliable Stair Climbing in the Simple Hexapod RHex '. In *2002 IEEE International Conference on Robotics and Automation (ICRA)*, number May, pages 2222–2227. IEEE, 2002.

[135] Keith W. Moored, Stuart A. Taylor, Thomas K. Bliss, and Hilary Bart-Smith. Optimization of a tensegrity wing for biomimetic applications. In Yuji Matsuzaki, editor, *Proceedings of the 45th IEEE Conference on Decision and Control*, volume 6173, pages 2288–2293. IEEE, dec 2006.

[136] R Motro. *Tensegrity: Structural systems for the future*. Butterworth-Heinemann, 2003.

[137] Robin R. Murphy. Roboticists Without Borders: Center for Robot-Assisted Search and Rescue (CRASAR) at Texas A&M University.

[138] Robin R. Murphy. *Disaster Robotics*. The MIT Press, 2014.

[139] Robin R. Murphy, Karen L. Dreger, Sean Newsome, Jesse Rodocker, Brian Slaughter, Richard Smith, Eric Steimle, Tetsuya Kimura, Kenichi Makabe, Kazuyuki Kon, Hisashi Mizumoto, Michinori Hatayama, Fumitoshi Matsuno, Satoshi Tadokoro, and Osamu Kawase. Marine heterogeneous multirobot systems at the great Eastern Japan Tsunami recovery. *Journal of Field Robotics*, 29(5):819–831, sep 2012.

[140] Richard M. Murray, Zexiang Li, and Shankar S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[141] Quan Nguyen, Ayush Agrawal, Xingye Da, William Martin, Hartmut Geyer, Jessy Grizzle, and Koushil Sreenath. Dynamic Walking on Randomly-Varying Discrete Terrain with One-step Preview. In *Robotics: Science and Systems XIII*. Robotics: Science and Systems Foundation, jul 2017.

[142] Oliver M. O'Reilly. *Intermediate Dynamics for Engineers*. Cambridge University Press, Cambridge, 2008.

[143] Romeo Ortega, Antonio Loria, Rafael Kelly, and Laurent Praly. On passivity-based output feedback global stabilization of euler-lagrange systems. *International Journal of Robust and Nonlinear Control*, 5(4):313–323, 1995.

[144] Romeo Ortega, Antonio Loría, Per Johan Nicklasson, and Hebertt Sira-Ramírez. *Passivity-based Control of Euler-Lagrange Systems*. Communications and Control Engineering. Springer London, London, 1998.

[145] Romeo Ortega and Mark W. Spong. Adaptive motion control of rigid robots: A tutorial. *Automatica*, 25(6):877–888, 1989.

[146] Christian Ott, A. Albu-Schaffer, Andreas Kugi, and Gerd Hirzinger. On the Passivity-Based Impedance Control of Flexible Joint Robots. *IEEE Transactions on Robotics*, 24(2):416–429, apr 2008.

[147] OX Global Vehicle Trust. The OX: All-Terrain Vehicle, 2016.

[148] Luther R. Palmer and David E. Orin. Quadrupedal running at high speed over uneven terrain. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 303–308. IEEE, oct 2007.

[149] M Panjabi, J Dvorak, J Duranceau, I Yamamoto, M Gerber, W Rauschning, and H.U. Bueff. Three-dimensional movements of the upper cervical spine. *Spine*, 13(7):726–730, 1988.

[150] Hae-Won Park, Meng Yee Chuah, and Sangbae Kim. Quadruped bounding control with variable duty cycle via vertical impulse scaling. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, number Iros, pages 3245–3252. IEEE, sep 2014.

[151] Hae-Won Park, Sangin Park, and Sangbae Kim. Variable-speed quadrupedal bounding using impulse planning: Untethered high-speed 3D Running of MIT Cheetah 2. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5163–5170. IEEE, may 2015.

[152] Yong-Lae Park, Carmel Majidi, Rebecca Kramer, Phillipe Bérard, and Robert J. Wood. Hyperelastic pressure sensing with a liquid-embedded elastomer. *Journal of Micromechanics and Microengineering*, 20(12):125029, dec 2010.

[153] C. Paul, F.J. Valero-Cuevas, and H. Lipson. Design and control of tensegrity robots for locomotion. *IEEE Transactions on Robotics*, 22(5):944–957, oct 2006.

[154] Mark Pearcy, Ian Portek, and Janis Shepherd. Three-Dimensional X-ray Analysis of Normal Movement in the Lumbar Spine. *Spine*, 9(3):294–297, apr 1984.

[155] S. Pellegrino and C.R. Calladine. Matrix analysis of statically and kinematically indeterminate frameworks. *International Journal of Solids and Structures*, 22(4):409–428, jan 1986.

[156] C P Pesce. The Application of Lagrange Equations to Mechanical Systems With Mass Explicitly Dependent on Position. *Journal of Applied Mechanics*, 70(5):751, 2003.

[157] Dimitris Pongas, Michael Mistry, and Stefan Schaal. A Robust Quadruped Walking Gait for Traversing Rough Terrain. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, volume 307, pages 1474–1479. IEEE, apr 2007.

[158] Tobias Raff, Christian Ebenbauer, and Frank Allgöwer. *Nonlinear Model Predictive Control: A Passivity-Based Approach*, pages 151–162. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[159] Marc Raibert, Kevin Blankespoor, Gabriel Nelson, and Rob Playter. BigDog, the Rough-Terrain Quadruped Robot. *IFAC Proceedings Volumes*, 41(2):10822–10825, 2008.

[160] Raqim Foundation. Checklist For Effective Disaster Response, 2010.

[161] A.T. Riechel and I. Ebert-Uphoff. Force-feasible workspace analysis for underconstrained, point-mass cable robots. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, pages 4956–4962 Vol.5. IEEE, 2004.

[162] John Rieffel and Jean-baptiste Mouret. Adaptive and Resilient Soft Tensegrity Robots. *Soft Robotics*, 5(3):318–329, jun 2018.

[163] Nicolas Riehl, Marc Gouttefarde, Sébastien Krut, Cédric Baradat, and François Pierrot. Effects of non-negligible cable mass on the static behavior of large workspace cable-driven parallel mechanisms. In *2009 IEEE International Conference on Robotics and Automation*, pages 2193–2198. IEEE, may 2009.

[164] Matthew A. Robertson and Jamie Paik. Low-inertia vacuum-powered soft pneumatic actuator coil characterization and design methodology. In *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, number 513836, pages 431–436. IEEE, apr 2018.

[165] Jose Guadalupe Romero, Alejandro Donaire, Romeo Ortega, and Pablo Borja. Global stabilisation of underactuated mechanical systems via PID passivity-based control. *Automatica*, 96:178–185, 2018.

[166] Andrew P Sabelhaus, Abishek K Akella, Zeerek A Ahmad, and Vytas SunSpiral. Model-Predictive Control of a Flexible Spine Robot. In *2017 American Control Conference (ACC)*. IEEE, may 2017.

[167] Andrew P Sabelhaus, Jonathan Bruce, Ken Caluwaerts, Pavlo Manovi, Roya Fallah Firoozi, Sarah Dobi, Alice M. Agogino, and Vytas SunSpiral. System design and locomotion of SUPERball, an untethered tensegrity robot. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2015.

[168] Andrew P. Sabelhaus, Lara Janse van Vuuren, Ankita Joshi, Edward Zhu, Hunter J. Garnier, Kimberly A. Sover, Jesus Navarro, Adrian K. Agogino, and Alice M. Agogino. Design, Simulation, and Testing of a Flexible Actuated Spine for Quadruped Robots. In *arXiv:1804.06527v2*, 2018.

[169] Andrew P Sabelhaus, Hao Ji, Patrick Hylton, Yakshu Madaan, ChanWoo Yang, Alice M Agogino, Jeffrey Friesen, and Vytas SunSpiral. Mechanism Design and Simulation of the ULTRA Spine: A Tensegrity Robot. In *ASME International Design Engineering Technical Conference (IDETC) 39th Mechanisms and Robotics Conference*. ASME, aug 2015.

[170] Asher Saghian. *Design and Evaluation of the Curved Rod and W-Cross Tensegrity Spines*. M.eng, University of California, Berkeley, 2017.

[171] Shankar Sastry. *Nonlinear Systems*, volume 10 of *Interdisciplinary Applied Mathematics*. Springer New York, New York, NY, 1999.

[172] H. J. Schek. The force density method for form finding and computation of general networks. *Computer Methods in Applied Mechanics and Engineering*, 3(1):115–134, jan 1974.

[173] Aravind Seeni, Bernd Schafer, Bernhard Rebele, and Nikolai Tolyarenko. Robot Mobility Concepts for Extraterrestrial Surface Exploration. In *2008 IEEE Aerospace Conference*, pages 1–14. IEEE, mar 2008.

[174] Sangok Seok, Albert Wang, Meng Yee Michael Chuah, Dong Jin Hyun, Jongwoo Lee, David M Otten, Jeffrey H Lang, and Sangbae Kim. Design Principles for Energy-Efficient Legged Locomotion and Implementation on the MIT Cheetah Robot. *IEEE/ASME Transactions on Mechatronics*, 20(3):1117–1129, jun 2015.

[175] Mizuho Shibata, Fumio Saijyo, and Shinichi Hirai. Crawling by body deformation of tensegrity structure robots. In *2009 IEEE International Conference on Robotics and Automation*, ICRA'09, pages 4375–4380, Piscataway, NJ, USA, may 2009. IEEE.

[176] Hee-Sup Shin, Jaiyoung Ryu, Carmel Majidi, and Yong-Lae Park. Enhanced performance of microfluidic soft pressure sensors with embedded solid microspheres. *Journal of Micromechanics and Microengineering*, 26(2):025011, feb 2016.

[177] Robin Shrestha, Moble Benedict, Vikram Hrishikeshavan, and Inderjit Chopra. Hover Performance of a Small-Scale Helicopter Rotor for Flying on Mars. *Journal of Aircraft*, 53(4):1160–1167, jul 2016.

[178] R Skelton and J.M. Mirats-Tur. Nonlinear control of non-minimal tensegrity models. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 6662–6667. IEEE, dec 2009.

[179] Robert E Skelton and Mauricio C de Oliveira. *Tensegrity systems*. Springer, 2009.

[180] Theodoor H. Smit, Manon SLM van Tunen, Albert J van der Veen, Idsart Kingma, and Jaap H van Dieën. Quantifying intervertebral disc mechanics: a new definition of the neutral zone. *BMC Musculoskeletal Disorders*, 12(1):38, dec 2011.

[181] Kenneth Snelson and Eleanor Heartney. *Kenneth Snelson, Art and Ideas*. Marlborough Gallery, NY, 2013.

[182] Alexander Spröwitz, Alexandre Tuleu, Massimo Vespignani, Mostafa Ajallooeian, Emilie Badri, and Auke Jan Ijspeert. Towards dynamic trot gait locomotion: Design, control, and experiments with Cheetah-cub, a compliant quadruped robot. *The International Journal of Robotics Research*, 32(8):932–950, 2013.

[183] S W Squyres, A H Knoll, R E Arvidson, J W Ashley, J. F. Bell, W M Calvin, P R Christensen, B C Clark, B A Cohen, P. A. de Souza, L Edgar, W H Farrand, I Fleischer, R Gellert, M P Golombek, J Grant, J Grotzinger, A Hayes, K. E. Herkenhoff, J. R. Johnson, B. Jolliff, G. Klingelhofer, A. Knudson, R. Li, T. J. McCoy, S. M. McLennan, D. W. Ming, D. W. Mittlefehldt, R. V. Morris, J. W. Rice, C. Schroder, R. J. Sullivan, A. Yen, and R. A. Yingst. Exploration of Victoria Crater by the Mars Rover Opportunity. *Science*, 324(5930):1058–1061, may 2009.

[184] E.C. Stone. The Voyager missions to the outer system. *Space Science Reviews*, 21(2):91125, nov 1977.

[185] Cornel Sultan. Stiffness formulations and necessary and sufficient conditions for exponential stability of prestressable structures. *International Journal of Solids and Structures*, 50(14-15):2180–2195, jul 2013.

[186] Cornel Sultan, Martin Corless, and Robert E Skelton. Symmetrical reconfiguration of tensegrity structures. *International Journal of Solids and Structures*, 39(8):2215–2234, apr 2002.

[187] Cornel Sultan and Robert Skelton. Deployment of tensegrity structures. *International Journal of Solids and Structures*, 40(18):4637–4657, 2003.

[188] Vytas SunSpiral, D.W. Wheeler, Daniel Chavez-Clemente, and David Mittman. Development and field testing of the FootFall planning system for the ATHLETE robots. *Journal of Field Robotics*, 29(3):483–505, may 2012.

[189] Adnan Tahirovic and Gianantonio Magnani. General Framework for Mobile Robot Navigation Using Passivity-Based MPC. *IEEE Transactions on Automatic Control*, 56(1):184–190, jan 2011.

[190] Takashi Takuma, Masahiro Ikeda, and Tatsuya Masuda. Facilitating multi-modal locomotion in a quadruped robot utilizing passive oscillation of the spine structure. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4940–4945. IEEE, oct 2010.

[191] Shu Jun Tan. *Design, Prototype and Evaluation of the Under-Actuated Lightweight Tensegrity Robotic Assistive (ULTRA) Spine Robot: W-Extended, W-Alternating, and X-Alternating Spines.* M.eng, University of California, Berkeley, 2017.

[192] Robel Bahlbi Teweldebirhan. *Simulation and Performance Evaluation of Tensegrity Spine Designs.* M.eng, University of California, Berkeley, 2017.

[193] A.G. Tibert and S Pellegrino. Review of Form-Finding Methods for Tensegrity Structures. *International Journal of Space Structures*, 18(4):209–223, dec 2003.

[194] Brian R Tietz, Ross W Carnahan, Richard J Bachmann, Roger D Quinn, and Vytas SunSpiral. Tetraspine: Robust terrain handling on a tensegrity robot using central pattern generators. In *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 261–267. IEEE, jul 2013.

[195] Hoang Chi Tran and Jaehong Lee. Advanced form-finding of tensegrity structures. *Computers & Structures*, 88(3-4):237–246, feb 2010.

[196] Arjan van der Schaft. *L2 - Gain and Passivity Techniques in Nonlinear Control*. Communications and Control Engineering. Springer London, London, 2000.

[197] Arjan van der Schaft. *L2-Gain and Passivity Techniques in Nonlinear Control*. Communications and Control Engineering. Springer International Publishing, Cham, 2017.

[198] Massimo Vespignani, Chiara Ercolani, Jeffrey Michael Friesen, and Jonathan Bruce. Steerable Locomotion Controller for Six-strut Icosahedral Tensegrity Robots. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2886–2892. IEEE, oct 2018.

[199] Massimo Vespignani, Jeffrey M Friesen, Vytas SunSpiral, and Jonathan Bruce. Design of SUPERball v2, a Compliant Tensegrity Robot for Absorbing Large Impacts. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2865–2871. IEEE, oct 2018.

[200] John T Watson, Michelle Gayer, and Maire A Connolly. Epidemics after natural disasters, jan 2007.

[201] We Care Solar. WE CARE Solar — The Power to Save Lives, http://wecaresolar.org/.

[202] Karl Weinmeister, Peter Eckert, Hartmut Witte, and A.-J. Ijspeert. Cheetah-cub-S: Steering of a quadruped robot using trunk motion. In *2015 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 1–6. IEEE, oct 2015.

[203] P. Wenger and D. Chablat. Kinetostatic analysis and solution classification of a class of planar tensegrity mechanisms. *Robotica*, 37(7):1214–1224, jul 2019.

[204] Edward L. White, Michelle C. Yuen, Jennifer C. Case, and Rebecca K. Kramer. Low-Cost, Facile, and Scalable Manufacturing of Capacitive Sensors for Soft Systems. *Advanced Materials Technologies*, 2(9):1700072, sep 2017.

[205] Hans-Joachim Wilke, Annette Kettler, and Lutz E Claes. Are Sheep Spines a Valid Biomechanical Model for Human Spines? *Spine*, 22(20):2365–2374, oct 1997.

[206] JM Willems, GA Jull, and JK-F Ng. An in vivo study of the primary and coupled rotations of the thoracic spine. *Clinical Biomechanics*, 11(6):311–316, sep 1996.

[207] Alexander W Winkler, Carlos Mastalli, Ioannis Havoutis, Michele Focchi, Darwin G Caldwell, and Claudio Semini. Planning and execution of dynamic whole-body locomotion for a hydraulic quadruped on challenging terrain. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5148–5154. IEEE, may 2015.

[208] R. J. Wood, S. Avadhanula, R. Sahai, E. Steltz, and R. S. Fearing. Microrobot Design Using Fiber Reinforced Composites. *Journal of Mechanical Design*, 130(5):052304, 2008.

[209] Karl Worthmann, Mohamed W Mehrez, Mario Zanon, George K I Mann, Raymond G Gosine, and Moritz Diehl. Model Predictive Control of Nonholonomic Mobile Robots Without Stabilizing Constraints and Costs. *IEEE Transactions on Control Systems Technology*, 24(4):1394–1406, jul 2016.

[210] A.S. Wroldsen, M.C. de Oliveira, and R.E. Skelton. Modelling and control of non-minimal non-linear realisations of tensegrity systems. *International Journal of Control*, 82(3):389–407, mar 2009.

[211] Dian Yang, Mohit S. Verma, Ju-Hee So, Bobak Mosadegh, Christoph Keplinger, Benjamin Lee, Fatemeh Khashai, Elton Lossner, Zhigang Suo, and George M. Whitesides. Buckling Pneumatic Linear Actuators Inspired by Muscle. *Advanced Materials Technologies*, 1(3):1600055, jun 2016.

[212] Michelle C. Yuen, Rebecca Kramer-Bottiglio, and Jamie Paik. Strain sensor-embedded soft pneumatic actuators for extension and bending feedback. In *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, pages 202–207. IEEE, apr 2018.

[213] Kyle Zampaglione, Andrew P Sabelhaus, Lee-huang Chen, Alice M. Agogino, and Adrian K Agogino. DNA-Structured Linear Actuators. In *ASME International Design Engineering Technical Conference (IDETC) 40th Mechanisms and Robotics Conference*. ASME, aug 2016.

[214] D Zappetti, S Mintchev, J Shintake, and D Floreano. Bio-inspired Tensegrity Soft Modular Robots. In *Biomimetic and Biohybrid Systems. Living Machines*, pages 497–508. 2017.

[215] Marvin Zhang, Xinyang Geng, Jonathan Bruce, Ken Caluwaerts, Massimo Vespignani, Vytas SunSpiral, Pieter Abbeel, and Sergey Levine. Deep reinforcement learning for tensegrity robot locomotion. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 634–641. IEEE, may 2017.

[216] Huajing Zhao. *Model-Predictive Control for Reduced-Error Trajectory Tracking of a 2D Simulated Tensegrity Spine Robot*. M.eng, University of California, Berkeley, 2017.

[217] A. Zheng. A computationally efficient nonlinear MPC algorithm. In *American Control Conference*, volume 97, pages 1623–1627 vol.3. IEEE, 1997.

[218] B. Zi, B.Y. Duan, J.L. Du, and H. Bao. Dynamic modeling and active control of a cable-suspended parallel robot. *Mechatronics*, 18(1):1–12, feb 2008.