Holistic Soft Material Systems for Advancing Robotic and Wearable Sensing Functionality

Anthony Wertz

CMU-RI-TR-25-04
February 2025

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

Thesis Committee:
Carmel Majidi (chair)
Oliver Kroemer
Sarah Bergbreiter
Cynthia Hipwell (Texas A&M University)

Submitted in partial fulfillment of the requirements for the Degree of Doctor of Philosophy

To my wonderful wife Rachael and amazing children Benjamin and Maria. I love you all very much and could not have accomplished this without you! Thank you for standing by and supporting me throughout this journey even when it was hard.

Abstract

The integration of soft and multifunctional materials in emerging technologies is becoming more widespread due to their ability to enhance or improve functionality in ways not possible using typical rigid alternatives. This trend is evident in various fields. For example, wearable technologies are increasingly designed using soft materials to improve mechanical compatibility with biological systems and employing conformable interfaces for electrodes for enhanced signal integrity and user comfort. Likewise, surgical tools are leveraging soft material systems to reduce the risk of tissue damage through their inherent compliance. Soft material systems are also being incorporated into robots to improve safety in human-robot interactions, as in co-working and assistive applications.

However, the same lack of rigidity and complex constitutive properties that make these material systems useful in emerging applications also present challenges in fully exploiting their capabilities. Soft substrates are continuously deformable and, without rigid constraints or simplifying operational assumptions, state inference can be difficult or impossible. In systems that exploit dynamic material properties, such as those using shape-memory alloys for actuation or thermoplastic polymers for stiffness tuning, system behavior is challenging to model from a controls perspective due to internal states that are difficult or impossible to measure in real time.

Exploiting these materials effectively requires improved sensor integration and device co-design. Here I discuss how sensing can be integrated to harness the inherent functionality of these non-traditional materials while preserving their novel properties and minimizing unnecessary design complexity. I first examine integration into existing systems, highlighting both the potential benefits and challenges of approaching sensorization in this way. Then I propose a more holistic design approach that embraces a synergistic relationship between material systems and their embedded sensors.

Acknowledgments

My journey to complete this work has been shaped by countless individuals. Although words cannot fully express my gratitude to everyone who has supported me in this endeavor, I would at least like to acknowledge those whose guidance, wisdom, and support were instrumental in bringing this research to fruition. Rome was not built in a day, nor was it built alone. Science progresses through collaboration and the dissemination of knowledge, each contribution adding to the foundation laid by those before us. It is my sincere hope that this work represents meaningful advances that support future discoveries. As I acknowledge those who most closely influenced my progress, I remain mindful of the broader community of scholars, mentors, and support structures outside academia whose collective efforts make such scientific pursuits possible.

I would like to thank my advisor Carmel for his support and mentorship throughout this process. Carmel goes to great lengths to empower his students with the direction and resources necessary to pursue the topics that most excite them. On research, career, and life, he has provided meaningful guidance and advice and has done everything he could to help me succeed. His advising strikes a delicate balance between providing the necessary support while allowing me to maintain autonomy and develop into an independent researcher, scientist, and engineer.

I am very grateful to my exceptional thesis committee, who provided invaluable feedback and guidance. Oliver, who served on both my qualifying and thesis committees, offered excellent ideas and critiques that helped me refine my research direction. Sarah's sharp analytical insights and thorough feedback helped identify gaps, assumptions, and missed opportunities. Cynthia's expertise in friction and adhesion helped sharpen both my research questions and my experimental approach.

I must also recognize my former supervisor and advisor Artur Dubrawski, with whom I worked in the Auton Lab at CMU prior to entering the PhD program. Artur contributed greatly to my growth as a scientist, and without him I would never have been able to reach this point in my education and career. Artur is an amazing person who goes above and beyond to help the people around him succeed, as he as done with me. I am incredibly grateful for my experience working with him and others in the Auton Lab.

They say it takes a village to raise a child; I think the same is true for a PhD student. As far as villages go, the Soft Machines Lab was an excellent place to come of age. The exceptional caliber of my colleagues, combined with the lab's rich interdisciplinary environment, provided fertile ground for collaboration and discovery. I am deeply grateful to every member of SML, past and present, who has contributed to my growth through thoughtful feedback during presentations, practical assistance in the lab, and enlightening discussions in the office. There is not a single person I can think of, neither veteran members nor the new recruits, who has not shaped my development as a scientist.

I want to extend a special thank you to some of my closest collaborators, each of whom uniquely shaped my research journey. Drew provided exceptional mentorship during the challenging COVID period, going well beyond what was required of him to actively teach, mentor, and help me develop. Mason, with his vastly different expertise, proved to be an ideal collaborator - our complementary skill sets enabled us to demonstrate interesting technological advances. During my internship, Dylan's sharp insights and guidance, coupled with the independence he granted me, helped me flourish as a researcher. Sarvesh's enthusiasm and support were instrumental in bringing the final chapter of my research to fruition. Each of these partnerships has not only advanced my research, but also enriched my scientific perspective. Within the lab, I also benefited greatly from the advice and support of many others outside, including Dinesh, Michael, Zach, Nathan, and Andrew.

Most of all, I want to thank my incredible family. My wife Rachael, especially, sacrificed so much to help me reach this milestone. During my time at CMU, we welcomed two kids into our family and navigated the trials of COVID together. She took on countless responsibilities while working to keep our home running and her husband sane. I'm grateful to my parents, John and Nancy, for their encouragement over the years, and to my extended family — Tom, Colleen, Thom, and Matt — whose invaluable support made this achievement possible. And of course, my wonderful children, Benjamin and Maria, whom I get to come home to every day. I am blessed with such a supportive and loving family, and I will always cherish that.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Soft and multifunctional materials are making their way into many new technologies and are the subject of a large body of emerging research [56, 60, 71]. Soft robotics is an especially exciting branch of research that incorporates soft materials into the space of robotics, a field typically composed almost entirely of hard materials and some semi-rigid or soft interfaces for wheels or contacts. Whitesides gives an excellent definition of the field [124], and Yasa et al. cover some more recent advances [128]. In soft robots, material properties play a much more fundamental role in robot operation, rather than being largely relegated to structural purposes only [116, 67]. This academic field has dramatically accelerated in the last decade as researchers have devised novel ways to characterize the behavior of soft systems, model them, and incorporate them into interesting devices [43, 51]. This is not a trivial point, as soft systems offer a number of unique challenges that are not typically encountered in rigid robotics. For one, soft matter is complicated to model and simulate due to its high degree of mechanical deformability [128]. In contrast, rigid systems can be treated as non-deformable, simplifying the analysis of dynamics and device design. The kinematic degrees of freedom of a rigid robot can be largely equated to the discrete number of joints in the system. In stark contrast, soft systems are often described as infinite degree-of-freedom systems because they can undergo significant deformation anywhere. This makes kinematic modeling very challenging, actuation must be carefully thought out, and contact simulation is still a considerable open challenge [72]. Soft robot actuation itself is also typically quite varied and complicated [134], and in itself difficult to model [4]. Another set of challenges faced in soft systems involves changing material properties and hysteresis on timescales relevant to a robot's operation or lifespan [20, 4]. In rigid systems, these aspects can often be ignored, as the materials used typically do not undergo significant degradation or exhibit hysteretic behavior. However, repeated strain cycles will typically weaken polymers and reduce their

elastic modulus. In addition, if the material exhibits significant viscoelasticity, the effects of both strain and relaxation should be considered for the application.

With all of the challenges of soft robotics and soft devices in general, it is worth mentioning some of their advantages as well. Soft substrates interface much more naturally and reliably with biological organisms [128]. Rigid biomedical devices such as physiologic monitors, implants, wearables, and prosthetics are typically rigid, bulky, uncomfortable, and interface poorly with the human body largely due to the extreme disparity in modulus that prevents robust and reliable mating. As Cianchetti et al. discuss, this is changing rapidly with recent advances in soft electronics and prosthetics [22]. There are still numerous challenges in reliable kinematics [4], state inference [4, 47], and control [25, 119]. Soft materials also enable capabilities unattainable in their rigid counterparts, such as significant shape change or deformation [53, 115], self-sensing [135, 37], and self-healing [113]. This is partly why they can be so incredibly difficult to model, simulate, and fabricate, but it also serves as a motivating factor for their incorporation in many systems where such properties are desirable.

One major hurdle to the effective use of soft and multifunctional material systems is the ability to monitor the state of the system well enough to take advantage of devices such as soft continuum end-effectors [2, 19], shape-memory alloy actuators [23], and stiffness-tunable materials [120]. Incorporating enough sensory feedback can quickly complicate the device circuitry, require too much physical space in the design, and begin to degrade material performance. The stiffening that will occur in a continuum limb if too many tubes, wires, or other structures are incorporated is one example of this.

In robotics, the material properties of interfacial contacts have largely been treated as static, even in soft robots: a rigid robot might have a rubber foot or boot; a vehicle might have a rubber wheel; and a soft robot might have a silicone limb. However, this is not how biological organisms function. Humans have the ability to dynamically modulate fingertip friction to optimize grip for particular manipulation tasks [131, 3]. Slugs actively secrete viscous mucous to assist in their locomotion and allow adhesion to resist succumbing to the crashing waves or strong currents [16]. Geckos can modulate the amount of total contact area to quickly engage or disengage an exceptional range of adhesion, allowing vertical and inverted locomotion [92]. Some soft systems attempt to mimic this tunability in some way. Jamming robots can be used to gently form fit and then manipulate irregular objects [12]. Controllable suckers take inspiration from octopi to grip objects [70] or pick-and-place operations. Electromagnets allow one to grip oddly shaped or flat metal objects, or even non-conductive surfaces through electroadhesion [35].

In this work, I will report on a number of soft systems in which sensors were integrated, and I will discuss some of the benefits and pitfalls of different approaches. The final chapters

will explore two interesting device designs that attempt to build a stronger connection between the use of soft materials to enable new device functionality and the sensor and electronics integration required to fully take advantage of them. I propose a fundamental exploration of a novel interface design that takes advantage of the modulus control of shape-memory polymers to provide a controllable stiffness interface for contact adaptation and sensing. I hypothesized that this controllability could improve a robot's tactile sensing capabilities by interrogating an object of interest with materials of different elastic moduli. I further hypothesized that these tunable interfaces could improve a robot's ability to adapt to varying contact scenarios by increasing or decreasing its contact stiffness and, by extension, its contact friction. The aim of this work is to demonstrate how the lessons learned in a number of soft-system sensor integrations can provide guidance on how we can best design new systems that incorporate such materials effectively.

- In chapter 2, I discuss some foundations in dynamics modeling, model calibration, and control of soft systems incorporating shape memory alloys, and look at the instrumentation of such systems that enables this control.
- Chapter 3 will highlight my relevant work in low-power electronics, energy harvesting, and battery-free sensing enabled by soft thermoelectric generators.
- A novel integrated soft sensor design is presented in chapter 4, which better integrates the sensor infrastructure into the material design process.
- Chapter 5 introduces a novel tunable-stiffness robot interface that acts as both a tactile sensor and provides the ability to adapt to different contact requirements.
- In chapter 6, I discuss the results of this work as well as some directions for future research.

Part I

Instrumenting soft systems to extend capabilities

In the chapters that follow, I discuss my work on two projects in which electronics were designed and implemented to extend the capability or to enable functionality of soft devices. We first look at a soft robot limb in chapter 2 whose controls were previously manually specified through trial-and-error. The intent was to replace this time-consuming process through sensorization to enable dynamics modeling, simulation, and optimization of open-loop controls based on a user-specified or robot-learned target trajectory. In chapter 3, a soft, wearable, energy-harvesting device is exploiting using an electronics platform I designed to enable continuous battery-free biometric sensing and wireless communication.

Chapter 2

Soft robot modeling, simulation, and control

In this chapter, I will discuss the use of sensorization as a means of fully exploiting the capabilities of a soft robot limb, designed by my collaborator Andrew Sabelhaus (Soft Robotics Control Lab, Boston University). The complete robot is made up of five links in the shape of an "M", forming three limbs and two shoulders. It also bears some resemblance to an elephant, for which it was named Horton. Each segment uses an antagonistic pair of shape-memory alloy (SMA) actuator coils to enable curling. The design of the limb constrains all bending to a single axis, and stiffer polymers are used to minimize extension or compression, so they can be safely ignored. The complete Horton robot is depicted in figure 2.1.

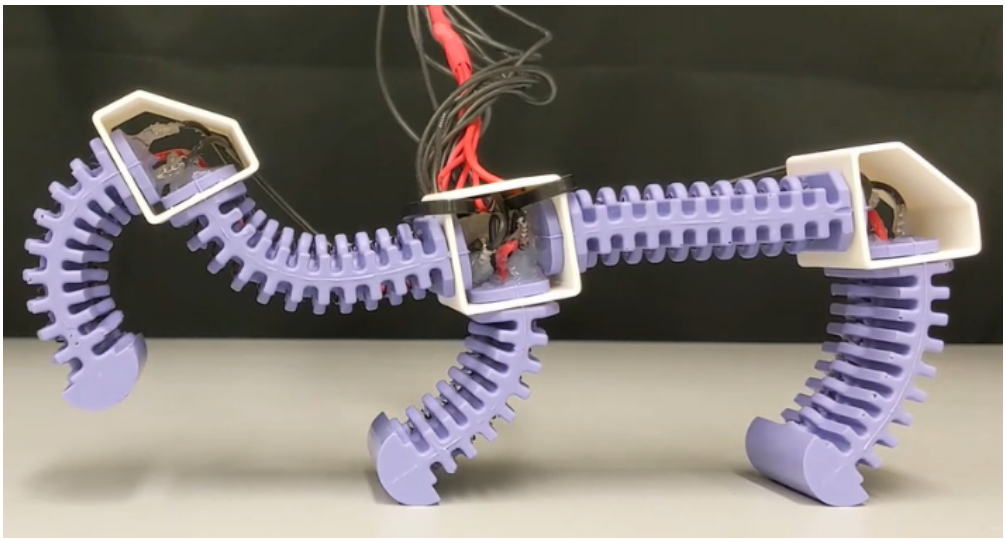


Figure 2.1: Horton robot, consisting of five identical segments antagonistically actuated using shape memory alloy (SMA) coils. Courtesy of Andrew Sabelhaus.

Prior to this work, the robot's actuation was primarily hand-tuned to achieve specific motions and gaits, but this process is fickle, time-consuming, and not particularly robust. The goal of the project was to employ sufficient sensorization to adequately infer the robot's state and enable dynamics modeling and optimization-based control design. This enables one to *derive* controls to achieve specific actions, instead of having to *intuit* them. What follows is a complete description of the sensorization that was performed, the calibration that was required, and the open-loop control performance that was achieved for a single-limb segment. We showed that overall good open-loop tracking can be achieved by making some simplified approximations of the soft limb and its SMA coils. We justified the use of a discretized rigid manipulator model for the dynamics with joint torques proportional to wire temperature. Then, we proposed a method to calibrate this model from experimental data and demonstrate that the simulation aligns well with a hardware test. Finally, we use a direct collocation optimization with the robot's nonlinear dynamics to generate feasible state-input trajectories from a desired reference. Three experiments validate our approach for a single-segment robot in hardware: first using a hand-derived reference trajectory, then with two teach-and-repeat tests.

What we will discover in this work is that, while it was ultimately a successful process, it also highlights some of the deficiencies of such an after-the-fact approach to sensorizing soft systems. The setup requires multiple deformation models for different calibration and operational regimes, the SMA temperature model does not accurately track the internal state, and the setup does not offer substantial room for additional sensorization without affecting device performance. Some of these pitfalls can be avoided by employing a tighter design between the electronics and the soft system, a topic that we will explore further in part II.

2.1 Introduction

Soft robots may outperform their rigid counterparts in tasks requiring biomimetic deformability and safe, robust environmental interaction [54, 66]. However, practical use of soft robots requires performing similarly-advanced motions as rigid robots. Many soft robots struggle to match more complicated trajectories due to limitations in actuation, design [90], and modeling [13] for high-degree-of-freedom state spaces. Generating feasible motions, and corresponding inputs, requires tractable models that accurately reproduce hardware behavior. This is particularly challenging for soft robots actuated with thermally-responsive materials like shape-memory alloy (SMA) becoming increasingly popular within the field [41, 5]. Though shape-memory actuators require minimal added hardware while providing high work density, unlike cable-driven and pneumatic systems, they are especially difficult to model and simulate for robotic applications [30].

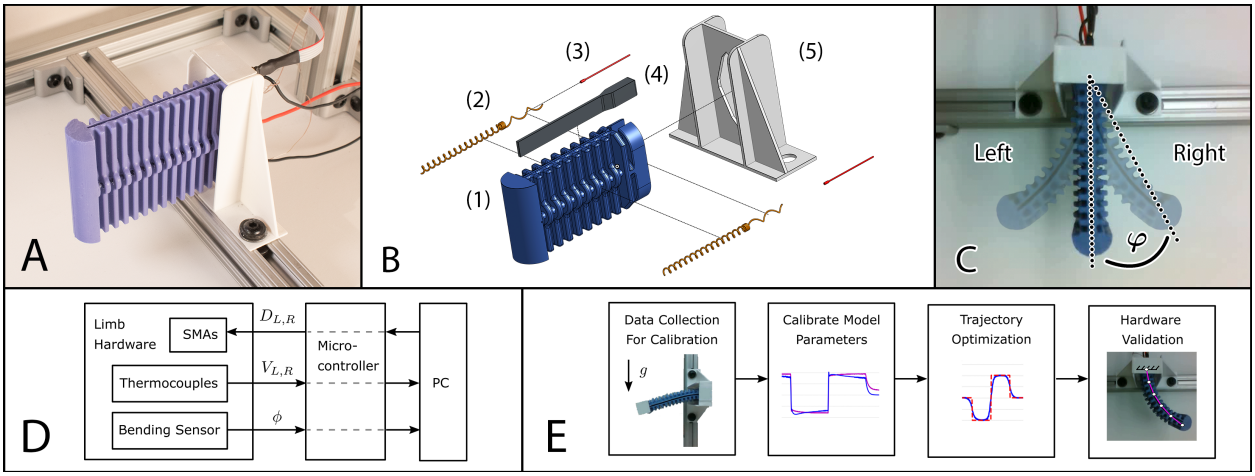


Figure 2.2: The limb (A) consists of cast bulk silicone (B1) actuated by two antagonistic SMA coils (B2). Thermocouples (B3) and a bend sensor (B4) are included. A bracket (B5) holds the limb horizontally to remove gravitational loading on the bending axis. Bending (C) is achieved using PWM signals from a microcontroller (D) to actuate the SMA coils through Joule heating. (E) Our procedure collects hardware data, calibrates the model, optimizes trajectories, then validates on hardware.

We propose a modeling and trajectory generation framework for soft limbs with antagonistic thermal actuators (Fig. 2.2). From prior work and first-principle approximations we develop a three-part dynamics model based on a rigid manipulator, a map from actuator temperature to joint torque, and Joule heating. We construct a single segment of a soft limb with two SMA actuators, motivated by [82] with the addition of temperature and displacement sensing, and calibrate the model using our hardware. We develop and solve an optimization problem to generate feasible state and control trajectories. Open-loop hardware

tests demonstrate that our approach can be used to re-create "teach and repeat" motions of the limb.

We focus on open-loop operation, as feedback control has been extensively studied for soft robots with thermal [127, 65] and controllable-force actuators [69, 26]. Trajectory optimization offers two distinct benefits in comparison to feedback without pre-planned trajectories [46]: First, dynamic feasibility is verified *a priori*, a challenging requirement for many state-feedback techniques (e.g., model-predictive control [94]) in soft and flexible robots. Second, objectives of interest, like minimum energy expenditure or minimum time, can be incorporated directly through costs or constraints. Since thermal actuators have significant energy requirements [90], these are often opposing goals.

This article contributes a trajectory optimization technique for a thermally-actuated soft planar robot limb, comprising:

1. An approximated model of the manipulator and actuator dynamics with low computational cost,
2. An optimization routine that uses the model for generating feasible trajectories, and
3. A validation of the approach, faithfully re-creating three motions of the limb in hardware.

This article applies our approach to a single-segment robot with two actuators. Single-segment planar soft robots are found in a variety of settings, such as multi-fingered hands with a rigid base [118], and are routinely used as benchmarks [110]. Our approach makes motion planning possible when thermal actuators are used in these applications. This is the first demonstration of a re-usable computational trajectory generation approach for any thermally-actuated soft robot.

2.2 Related work

We consider soft limbs actuated with SMA coils that, due to their size and work density advantages [36, 67], are promising for compact, untethered robots [82]. However, other soft actuation mechanisms (such as pneumatics and cables) can directly control applied force [69, 34], whereas thermal actuation only occurs indirectly [93]. Since traditional robotics models do not capture thermal actuator dynamics, little progress has been made toward dynamic trajectory generation for these mechanisms. Prior work includes $A*$ to optimize SMA arrays [73] and evolutionary algorithms for rigid SMA-actuated robots [45]. Neither test their results on hardware, nor address feasibility. There have been attempts at open-loop SMA task-space operation [9], though not for soft manipulators. To our knowledge, no prior work has computationally generated feasible state-input trajectories of a soft robot limb with

thermal actuators.

Many models of soft-bodied robot arm kinematics and dynamics are available [95] with trade-offs between complexity, computational requirements, and physical accuracy. More physically accurate models include the discrete elastic rod (DER) method [33, 39, 40]. More computationally-tractable models include the constant-curvature framework [121, 26]. We use one of the simplest possible representations: the rigid manipulator, motivated by approximations of the above alongside promising contemporary results [34].

Dynamics for thermal actuators are often based on first principles and constitutive models [21, 36]. However, these often require measurements of stress and strain [30], which adds to computational complexity and presents a challenge for robot design. This article investigates if temperature alone can approximate a more complicated stress/strain actuator response, since temperature is more readily modeled with Joule heating and convective cooling [10, 93]. Our simple temperature-to-stress model makes computational trajectory generation possible for this class of robots.

2.3 Hardware platform

Our soft limb (Fig. 2.2A, B) is derived from prior work in a legged robot [82], with the intent to eventually be employed in that setting. The limb body was cast from bulk silicone elastomer (Smooth-Sil 945, Smooth-On) and embedded with nickel-titanium alloy SMA actuator coils (Flexinol, Dynalloy) and a capacitive bend sensor (single axis, Bend Labs). Thermocouples for measuring SMA wire temperature were bonded to the bracketed side of the actuators with thermally conductive, electrically insulating epoxy (MG 8329TCF). The limb was mounted with a 3D-printed bracket oriented with the bending axis parallel to gravity (Fig. 2.2) so gravitational loading can be ignored.

Sensing and control were performed with an offboard microcontroller (nRF52-DK, Nordic Semiconductor). The thermocouples (attached to an amplifier, MAX31855) and the bending sensor both communicated digitally with the microcontroller (Fig. 2.2D). Current through the SMA actuators was controlled using pulse-width modulation (PWM) to N-channel power MOSFETs connected to a 7V power supply.

2.4 Dynamics modeling

The approach in this article makes approximations to each of the three relevant dynamics phenomena of our robot limb: the manipulator body, the discretized joint torques, and the actuator temperature. These are combined into a final set of equations of motion, taking

a pulse-width modulation (PWM) voltage on the SMA wires as input and predicting the manipulator's bending angle as output.

2.4.1 Rigid manipulator model

This article employs a simplified model of a discretized rigid manipulator for the robot's body. Two observations motivate this approach. First, recent work has demonstrated relatively accurate simulations of soft fluid-driven limbs as discretized manipulators [34]. Second, the discretized rigid manipulator arises from the discrete elastic rod (DER) model, previously shown to accurately model SMA-driven soft robots [33, 39], under certain approximating assumptions. In particular, our robot does not experience significant centerline extension. The DER with no stretching is dynamically equivalent to a serial-chain rigid manipulator with nonlinear springs at each discretized joint. Then, modeling the bending forces with *linear* springs instead, the dynamics become the flexible manipulator model (Fig. 2.3) of

$$\mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + k(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}) + \sigma\dot{\boldsymbol{\theta}} = \mathbf{0}, \tag{2.1}$$

with the conventional mass and Coriolis/centrifugal terms M, C and spring constant k. We include linear damping σ for energy dissipation in the soft material as in [39]. The vector $\boldsymbol{\theta}$ are angles between the n discretized segments: $\boldsymbol{\theta} = [\theta_1,\ \theta_2\ , \ldots, \theta_n]$, hereafter referred to as joint angles.

The discretized manipulator model in eqn. (2.1) actuates by changing the set-point angle $\bar{\boldsymbol{\theta}}$ of the torsional springs, as in the DER dynamics [39]. Moreover, as in [39], we assume that this change occurs as a function of temperature in our two SMAs, $\mathbf{T} = [T_l, T_r]$, where r indicates the right-side actuator and l the left-side actuator. Since k distributes through our linear spring, we re-write the generalized force (generalized torque) due to actuation as $\mathbf{f}(\mathbf{T}) = k\bar{\boldsymbol{\theta}}(\mathbf{T})$, to be specified and calibrated later, arriving at

$$\mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + k\boldsymbol{\theta} + \sigma\dot{\boldsymbol{\theta}} = \mathbf{f}(\mathbf{T}). \tag{2.2}$$

2.4.2 SMA actuator model: temperature-force relationship

Eqn. (2.2) takes $\mathbf{f}(\mathbf{T})$ to be a static stateless mapping, i.e., neglects the internal constitutive properties of the SMA in favor of reduced computational complexity. To justify this highly simplified relationship, we consider the constitutive model for a single SMA, and determine

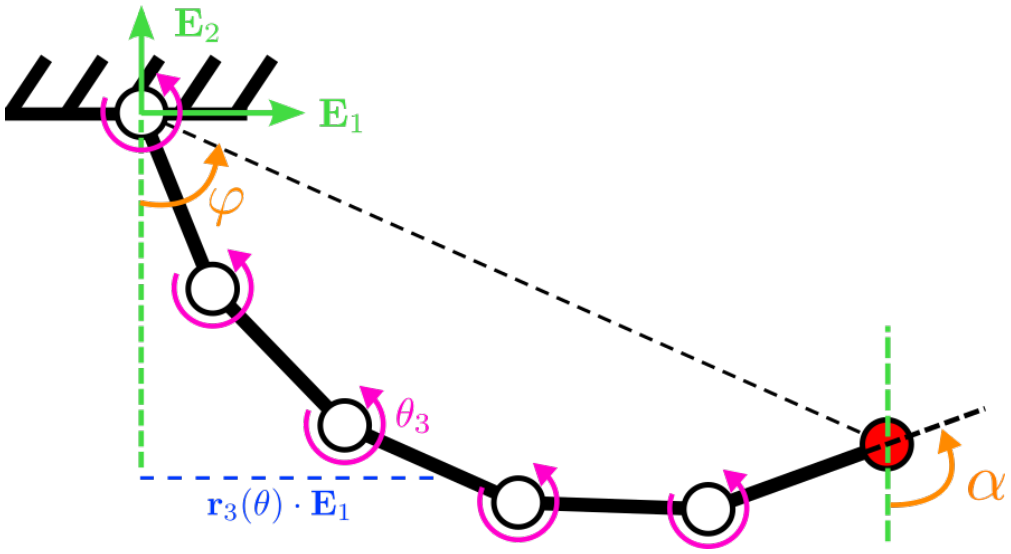


Figure 2.3: Fixed-base manipulator model with uniformly spaced revolute joints rotating normal to the plane. Orthogonal basis vectors $\mathbf{E}_1$ and $\mathbf{E}_2$ define the inertial frame centered on the first joint. The bend angle φ is estimated from the sensor's tip tangent angle α (eqn. 2.8) for model calibration (Sec. 2.5) along with link center of mass offsets (e.g., $\mathbf{r_3}(\boldsymbol{\theta}) \cdot \mathbf{E_1}$).

what approximations are implied. The SMA strain-stress-temperature model is well known in the literature, for example, as in [21] for one wire,

$$\tau - \tau_0 = G(\xi)(\gamma - \gamma_0) + \frac{\Theta}{\sqrt{3}}(T - T_0) + \frac{\Omega(\xi)}{\sqrt{3}}(\xi - \xi_0) \tag{2.3}$$

with the wire's shear stress τ, shear modulus G, shear strain γ, coefficient of thermal expansion Θ, temperature T, phase transformation coefficient Ω, and martensite fraction ξ. Quantities with *naught* subscript, e.g., τ_0, are the values at ambient temperature. In order, if the following approximations are assumed:

1. Thermal expansion is negligible (as in [21]),
2. Change in strain is small in comparison to change in stress, i.e., $G(\xi)(\gamma - \gamma_0) << (\tau - \tau_0)$ for our range of interest in ξ, so $G(\xi)(\gamma - \gamma_0) \approx 0$,
3. Martensite fraction is proportional to wire temperature, $\xi \propto T - T_0$,

then $\tau \propto T - T_0$. Lastly, assuming that τ acts uniformly across joints and that it induces the generalized torques in eqn. (2.2), then $\mathbf{f} \propto \boldsymbol{\tau}$. We do not consider any center-line dependence as would be expected in a tendon-driven system [15]; rather, this is accounted for via calibration of the torsional spring constant in the manipulator model. Lumping each scaling factor into a parameter $\beta \in \mathbb{R}$, then $\mathbf{f} = \beta(T - T_0)\mathbb{1}_\mathbf{n}$, where $\mathbb{1}_\mathbf{n}$ is a vector of ones of

length n. Therefore, with two SMA actuators acting antagonistically,

$$\mathbf{f}(\mathbf{T}) = \beta_r(T_r - T_0)\mathbb{1}_\mathbf{n} - \beta_l(T_l - T_0)\mathbb{1}_\mathbf{n}. \tag{2.4}$$

These assumptions are simplistic, but our hardware validation testing suggests they are a useful approximation.

2.4.3 SMA actuator model: temperature dynamics

Finally, we design a relationship between our control inputs, the PWM duty cycles $\mathbf{u} = [D_l, D_r] \in [0,1]^2$, and the SMA wire temperatures. As with stress vs. temperature, we consider each wire individually. It has been shown in the literature [73, 45, 10] that SMA wire temperature can be approximately modeled by Joule heating in the form

$$\dot{T} = -\frac{h_c A_c}{C_v}(T - T_0) + \frac{1}{C_v}P \tag{2.5}$$

with specific heat capacity C_v, ambient heat convection coefficient h_c, surface area A_c, ambient temperature T_0, and input electrical power P. For current-controlled SMAs, $P = \rho J^2$, where ρ is resistance and J is current density. For our PWM input, we assume that the duty cycle D modulates the fraction of time current is conducting through the SMA and that current is constant when flowing, so $P = \rho J^2 D$.

Our embedded thermocouple is bonded to the SMA wire using a small amount of thermally conductive, electrically insulating epoxy; this adds thermal mass. We therefore model the measured temperature V with an additional linear time delay. As a result, both $\dot{T}$ and $\dot{V}$ are linear systems, of the form

$$\dot{T} = a_1(T - T_0) + a_2 D\,, \tag{2.6}$$

$$\dot{V} = a_3(V - T)\,, \tag{2.7}$$

where $a_1 = -h_c A_c / C_v$ and $a_2 = \rho J^2 / C_v$. This linear model is equivalent to that used in [65]. The measurement temperature V is the state tracked in the simulated system dynamics, but T is readily computed by rearranging the terms in eqn. 2.7. The full model dynamics are depicted in Fig. 2.4.

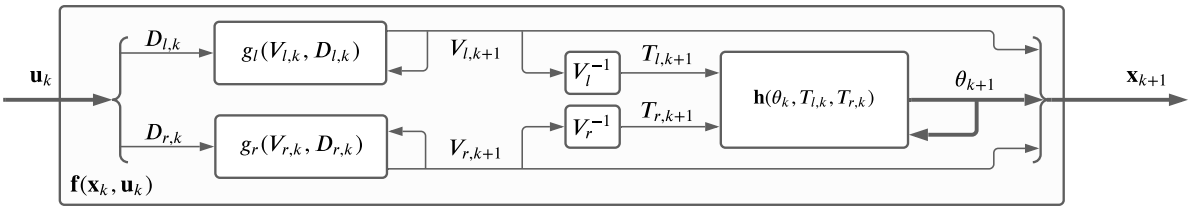


Figure 2.4: The combined dynamics f is composed of two thermal actuator blocks g_l, g_r, eqns. (2.6)-(2.7), and the serial manipulator dynamics h, eqn. (2.2). Wire temperature is obtained from measured temperature (V_j^{-1}) via $T_j = V_j - \dot{V}_j/a_3$.

2.5 Model Calibration

The dynamics model of eqns. (2.2, 2.4, 2.6, 2.7) requires calibration from data. Physical parameters (mass, mass moments of inertia, geometry) were measured using laboratory scales and our CAD model. For the remainder, we make the simplifying assumption of *constant curvature* [121] to map the angular displacement output of our bending sensor, α, to the bending angle of the limb, φ (Fig. 2.3), which is

$$\varphi = \alpha/2. \tag{2.8}$$

We first calibrate the two passive dynamics parameters, spring constant k and damping constant σ, then use actuated data to calibrate the generalized force and temperature models.

2.5.1 Linear torsional spring constant

To calibrate the spring constants k, we reoriented the limb with the bending axis parallel to the ground and measured the deflection in static equilibrium under gravitational loading. Here, with $\dot{\boldsymbol{\theta}}^{eq} = \mathbf{0}$ and $\mathbf{f}(\mathbf{T}^{eq}) = \mathbf{0}$, the manipulator dynamics (eqn. 2.2) simplify to $k\boldsymbol{\theta}^{eq} + \mathbf{f_g}(\boldsymbol{\theta}^{eq}) = \mathbf{0}$, picking up a new term due to gravity. We use an exponential map for the kinematics of the link centers of mass $\mathbf{r}_i(\boldsymbol{\theta}^{eq})$. Gravitational potential energy is then $U_g(\boldsymbol{\theta}^{eq}) = mg\sum_{i=1}^{n} \mathbf{r_i}(\boldsymbol{\theta}^{eq}) \cdot \mathbf{E_1}$, and so we computed $\mathbf{f_g}(\boldsymbol{\theta}^{eq}) = -\nabla_{\boldsymbol{\theta}} U_g(\boldsymbol{\theta}^{eq})$. A least-squares fit then gives $k = -\boldsymbol{\theta}^{eq} \setminus \mathbf{f_g}(\boldsymbol{\theta}^{eq})$ from hardware data of $\boldsymbol{\theta}^{eq}$.

However, our bend sensor only provides a scalar measurement of φ^{eq}, not the joint angles $\boldsymbol{\theta}^{eq}$, and the constant curvature assumption needed for eqn. (2.8) is only a rough approximation for our manipulator under gravity. To obtain θ_i^{eq} from φ^{eq}, we treat our manipulator as a discretized version of an Euler-Bernoulli beam under gravitational loading. The comparable loading condition is a moment applied at each joint, arising from gravitational generalized force at links further along the cantilever. A static equilibrium calculation, augmented with

a correction factor $\lambda \in \mathbb{R}^+$ to partially account for eqn. (2.8)'s approximation, gives

$$\theta_i^{eq} = \lambda \varphi^{eq} \frac{(N-i+1)^2}{\sum_{j=1}^{N}(N-j)^2} = \lambda \varphi^{eq} b_i \tag{2.9}$$

Given an observation $\{\varphi^{eq}, \boldsymbol{\theta}^{eq}\}$ from a simulation of eqn. (2.2) as $k\boldsymbol{\theta}^{eq} + \mathbf{f_g}(\boldsymbol{\theta}^{eq}) = \mathbf{0}$, an estimate for the correction factor is $\lambda^* = \theta_i^{eq}/(\varphi^{eq} b_i)$. We iterated between calculating k then re-estimating λ^* via simulation, starting from $\lambda^* = 1$ until both converged (at $\lambda^* \approx 1.22$).

2.5.2 Damping constant

To calibrate the damping constant σ, we displaced the tip of the limb to 45°, released, and collected hardware data $\varphi_{1\ldots t}^d$ during passive oscillation until the limb came to rest. With no actuation or external loading, eqn. (2.2) simplifies to

$$\mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + k\boldsymbol{\theta} + \sigma\dot{\boldsymbol{\theta}} = \mathbf{0}\,. \tag{2.10}$$

We use a nested, two-step optimization (Alg. 1) to estimate σ without needing to map $\varphi \to \boldsymbol{\theta}$. First, we note that we can optimize the parameters of a scalar damped sinusoid to fit a bend angle trajectory $\varphi_{1\ldots t}$ as $\sigma^* = \texttt{fitDS}(\varphi_{1\ldots t})$. However, we observed that a naïve use of $\sigma = \sigma_d^* = \texttt{fitDS}(\varphi_{1\ldots t}^d)$ in simulations of eqn. (2.10) overdamps the response. The outer loop of our optimization, `fitData`, therefore minimizes the difference between the damping constant estimate from hardware, σ_d^*, and the damping constant estimate from manipulator simulations $\sigma_{mdl}^* = \texttt{fitDS}(\varphi_{1\ldots t}^{mdl} | \sigma_{mdl})$. Our implementation of `fitData` obtains $\varphi_{1\ldots t}^{mdl}$ by rolling out eqn. (2.10) given a σ_{mdl}. Therefore, $\sigma = \sigma_{mdl}^*$ is the manipulator (simulation) damping constant that best re-creates the magnitude of damping observed from a hardware fit. All future simulations of eqn. (2.2) used σ_{mdl}^*.

Algorithm 1: Nested optimization to find σ by matching the damping observed in hardware data $\varphi_{1\ldots t}^d$.

Procedure $\texttt{fitDS}(\varphi_{1\ldots t}) \to \sigma^*$**:**
- $\zeta^*, \omega_n^* \leftarrow \arg\min ||\varphi_{1\ldots t} - Ae^{-\zeta\omega_n t}\sin(\omega_n\sqrt{1-\zeta^2}t + \phi) + b||_2^2$
- $\sigma^* \leftarrow \zeta^*\omega_n^*$

Procedure $\texttt{fitData}(\varphi_{1\ldots t}^d) \to \sigma_{mdl}^*$**:**
- $\sigma_d^* \leftarrow \texttt{fitDS}(\varphi_{1\ldots t}^d)$
- $\sigma_{mdl}^* \leftarrow \arg\min ||\sigma_d^* - \texttt{fitDS}(\varphi_{1\ldots t}^{mdl} | \sigma_{mdl})||_2^2$

2.5.3 Thermal actuator constants

Thermal actuator constants were calculated with the limb reoriented back (no gravity). Here, the constant curvature assumption holds, and we can calculate θ_i from φ directly. With some trigonometry, and assuming identical joint angles,

$$\theta_i = 2\varphi/(n+1), \qquad \forall i = 1 \dots n. \tag{2.11}$$

To calibrate the constants for heat transfer, we collected data at static equilibria where our actuators were heated. Theoretically, if we operate our system at $\mathbf{f}(\mathbf{T}^{eq}) \neq \mathbf{0}$, $\dot{\varphi} = 0$, and $\dot{\mathbf{V}} = \mathbf{0}$, three conditions arise. Assuming we have held equilibrium for a sufficient amount of time, the wire temperature is equal to the measured temperature, $\mathbf{V}^{eq} = \mathbf{T}^{eq}$. Second, the dynamics in eqns. (2.2), (2.4) reduce to

$$k\boldsymbol{\theta}^{eq} = \frac{2k\varphi^{eq}}{(n+1)}\mathbb{1}_{\mathbf{n}} = \left[\beta_r(T_r^{eq} - T_0) - \beta_l(T_l^{eq} - T_0)\right]\mathbb{1}_{\mathbf{n}} \tag{2.12}$$

and so a fixed temperature maps to one fixed robot pose. Third, $\mathbf{V}^{eq} \neq 0 \Rightarrow \varphi^{eq} \neq 0$. Together, these observations allow us to calibrate $\{a_{1,j}, a_{2,j}, a_{3,j}, \beta_j\}$ for both actuators j without modeling dynamic motions.

We developed a simple PI feedback controller from φ to $\mathbf{D}$ to stabilize the limb around $\varphi^{eq} \neq 0$. Using this controller, we generated three calibration datasets of the form $\mathcal{C} = \{\varphi, \varphi^{eq}, \mathbf{V}, \mathbf{D}\}_{1\dots t}$ by randomly selecting φ^{eq} values in a range and operating our controller for some time at each. The first two datasets involved motion in which only one SMA was activated: $\mathcal{C}^r$ used $\varphi^{eq} \in (0, 45°)$ where the controller applied $D_l = 0$, vice-versa for a set $\mathcal{C}^l$. The third set $\mathcal{C}^m$ had both actuators activated.

Heat transfer coefficients

We independently fit the three parameters $\{a_{1j}, a_{2j}, a_{3j}\}$ in eqns. (2.6) and (2.7) for each SMA actuator j using $\mathcal{C}^j$. To obtain a time series for $T_{j,1\dots t}$ from $\mathcal{C}^j$, we observe that eqn. (2.12) implies equilibrium bending angle should be a linear scaling factor of temperature. That is, rearranging eqn. (2.12) becomes $T_j^{eq} = b\varphi^{eq} + T_0$ for some $b \in \mathbb{R}$. We examined $\mathcal{C}^j$ to find the most promising point $(\varphi^{eq*}, V_j^{eq*})$ where $\dot{V}_j \approx 0$, then with some algebra, $b = (V^{eq*} - T_0)/\varphi^{eq*}$. Lastly, we used this static relationship as a rough approximation for dynamic wire temperature not at equilibrium, $T_{j,1\dots t} = b\varphi_{1\dots t} + T_0$. With known trajectories for T_j, V_j, D_j, we used the `DiffEqParamEstim.jl` Julia package to fit the parameters, first using `two_stage_method` (a two-stage collocation procedure) to find a rough estimate, then

`optimize` for refinement, both with an L_2 residual loss.

Actuator force coefficients

To find β_j using $\mathcal{C}^j$, we observe that eqn. (2.12) also implies $\beta_j = [k/(T_j^{eq} - T_0)]\theta_i^{eq}$. We create a better estimate of $T_{j,1\ldots t}$ by simulating eqns. (2.6)-(2.7) using our calibrated $a_{1\ldots3,j}$ on the data in $\mathcal{C}^j$. Then, we picked out timepoints where $\dot{\varphi} \approx 0$ in $\mathcal{C}^j$, and took an average T^{eq} over a window around those points. Since we know k, stacking all t observations in the vector $\mathbf{\Gamma}_j^{eq}$ we have

$$\mathbf{\Gamma}_j^{eq} = \begin{bmatrix} \left(k/(T_{j1}^{eq} - T_0)\right)\theta_{i1}^{eq} \\ \vdots \\ \left(k/(T_{jt}^{eq} - T_0)\right)\theta_{it}^{eq} \end{bmatrix}, \tag{2.13}$$

where we then computed a least-squares fit as $\beta_j = \mathbb{1}_{\mathbf{t}} \setminus \mathbf{\Gamma}_j^{eq}$ to get force coefficients for each actuator j.

2.5.4 Model Validation

With all parameters identified, the dynamics can be compared to the hardware dataset $\mathcal{C}^m$ that includes activation of both actuators. We simulated our model in open loop using the inputs $\mathbf{D}_{1\ldots t}$ in $\mathcal{C}^m$, one subset plotted in Fig. 2.5. Though many approximations were made in both the dynamics derivation and calibration, the simulation faithfully predicts the limb's motion.

2.6 Trajectory optimization

Using this calibrated model, we propose the following optimization routine that finds feasible state-input trajectories $(\mathbf{x}^*_{1\ldots N}, \mathbf{u}^*_{1\ldots N})$ given an initial (likely infeasible) reference $(\mathbf{x}_{ref})$. We define the state vector at time k, $\mathbf{x_k}$, with the joint angles $\boldsymbol{\theta}$, measured wire temperatures V_l and V_r, and their derivatives. The input vector $\mathbf{u_k} = \mathbf{D}$ contains PWM duty cycles for the left and right actuators:

$$\mathbf{x} = \begin{bmatrix} \boldsymbol{\theta} & V_l & V_r & \dot{\boldsymbol{\theta}} & \dot{V}_l & \dot{V}_r \end{bmatrix}^\top, \quad \mathbf{u} = \begin{bmatrix} D_l & D_r \end{bmatrix}^\top \tag{2.14}$$

Our nonlinear optimization program takes a direct collocation approach [46] to find $\mathbf{x}^*_{1\ldots N}, \mathbf{u}^*_{1\ldots N}$, using a quadratic-cost objective (eqn. 2.15) and the constraints in eqns. (2.16)-

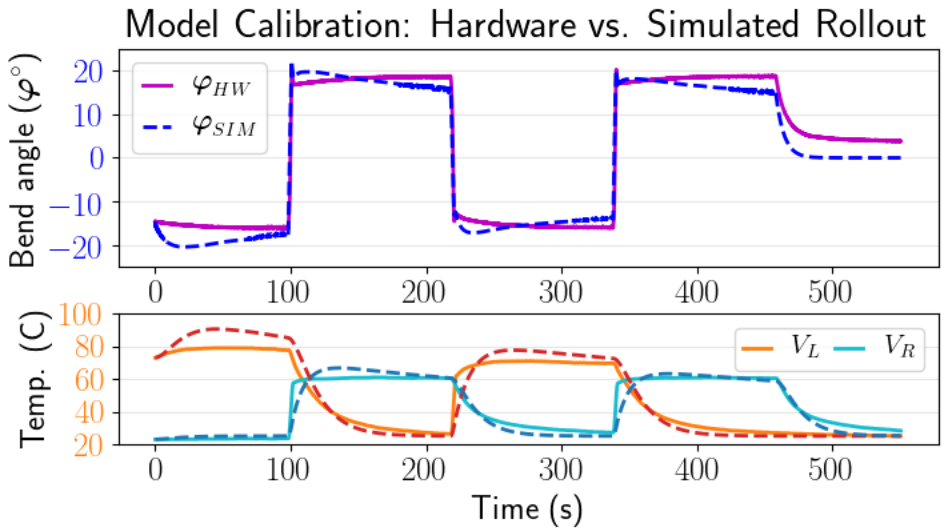


Figure 2.5: A simulation of the calibrated model (dashed) against a hardware test for the same inputs (solid) aligns qualitatively.

(2.20). We used IPOPT to solve the combined problem below, where N is the number of knot points, $\tilde{\mathbf{x}}_\mathbf{k} = \mathbf{x_k} - \mathbf{x_{k,ref}}$ is the deviation from the reference state at discrete time k, and $\mathbf{f}$ is the discrete dynamics function aggregated from the actuator and limb dynamics (Fig. 2.4). Objective weights were $\mathbf{Q} = 100\ \mathrm{diag}[\mathbb{1}_\mathbf{n}, \mathbf{0}]$ for the n-segment limb, only weighting $\boldsymbol{\theta}$, and $\mathbf{R} = 2\mathbf{I}$. We do not include a terminal constraint to avoid an infeasible problem; instead, we use a large terminal weight $\mathbf{Q_N} = 1000\mathbf{Q}$.

$$\mathbf{x}^*_{1\ldots N}, \mathbf{u}^*_{1\ldots N} = \arg\min_{\mathbf{x},\mathbf{u}} \frac{1}{2}\sum_{k=1}^{N-1} \left(\tilde{\mathbf{x}}_\mathbf{k}{}^\top \mathbf{Q}\tilde{\mathbf{x}}_\mathbf{k} + \mathbf{u_k}{}^\top \mathbf{R}\mathbf{u_k}\right) + \frac{1}{2}\tilde{\mathbf{x}}_\mathbf{N}{}^\top \mathbf{Q_N}\tilde{\mathbf{x}}_\mathbf{N} \tag{2.15}$$

$$\text{s.t.} \quad \mathbf{x_1} = \mathbf{x}_{\text{init}} \tag{2.16}$$

$$\mathbf{x_{k+1}} = \mathbf{f}(\mathbf{x_k}, \mathbf{u_k}) \tag{2.17}$$

$$\mathbf{0} \le \mathbf{u_k} \le \mathbb{1}_\mathbf{2} \tag{2.18}$$

$$\mathbf{T_k} < T_{max}\mathbb{1}_\mathbf{2} \tag{2.19}$$

$$\mathbf{T_k} > T_{warm}\mathbb{1}_\mathbf{2} \qquad \forall k > k_{warm}\,. \tag{2.20}$$

Our constraints include the physical limits of PWM duty cycle (eqn. 2.18) and a maximum of temperature (eqn. 2.19), chosen as a conservative $T_{max} = 100\,°\mathrm{C}$ to prevent damage to our 90°C SMAs. We also include a warmup constraint (eqn. 2.20) with $T_{warm} = 45\,°\mathrm{C}$, $k_{warm} = 20$sec., since we observed a better model fit to actuator force at higher temperatures.

Note that the dynamics and optimization are expressed in joint space, not task space.

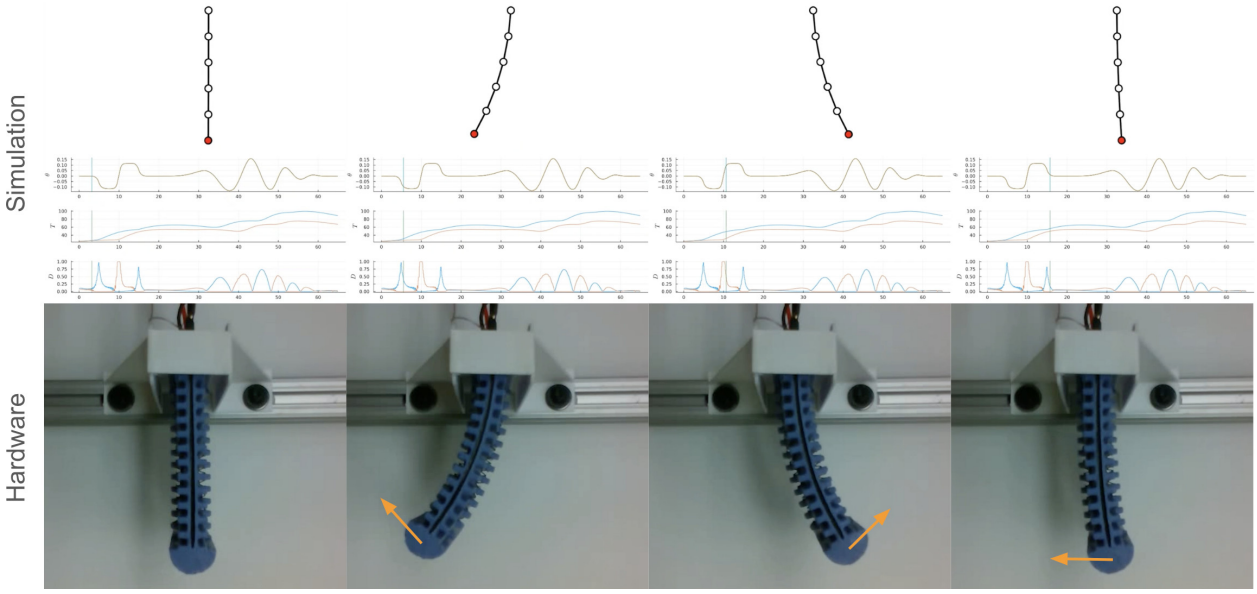


Figure 2.6: Frames from a simulated trajectory (top) and hardware tracking (bottom).

To determine the corresponding bend angle trajectory $\varphi^*_{1\ldots N}$, the forward kinematics are used to compute the tip position $\mathbf{t}_{1\ldots N}$, and then $\varphi = \arctan(\mathbf{t}\cdot\mathbf{E_2}/\mathbf{t}\cdot\mathbf{E_1})$. Lastly, the wire temperatures $\mathbf{T}$ for eqns. (2.19)-(2.20) were calculated from the states V_j in $\mathbf{x}$ via rearranging eqn. (2.7) into $T_j = V_j - \dot{V}_j/a_3$.

2.7 Results

We performed three experiments where a trajectory was optimized in software then executed in hardware. Our first test serves as a validation of the concept, where we specified the φ_{REF} in Fig. 2.7 by hand, consisting of two step inputs and a decaying sinusoid. Converting $\varphi_{REF} \rightarrow \boldsymbol{\theta}_{ref}$ via eqn. (2.11), we solved (2.15)-(2.20) to obtain the feasible trajectory φ^*. The corresponding open-loop inputs $\mathbf{u}^*$ were executed in hardware five times, plotted in Fig. 2.7 as the mean result $(\cdot)_{HW}$ and a shaded 95% confidence interval. Figure 2.6 shows a few frames of an optimized trajectory (top) and the hardware tracking (bottom).

The second two experiments were 'teach and repeat' (T&R) tests, where an initial trajectory φ_{TCH} was obtained by moving the limb by hand, with no actuation, and recording bend angle measurements (Fig. 2.9). These two dynamically infeasible φ_{TCH} were optimized to φ^* in Fig. 2.8, and as with the step/sinusoid test, the inputs $\mathbf{u}^*$ were executed in hardware. The first trajectory ('Example 1') tests faster motions within a small range of angles, whereas the second ('Example 2') tests a wider range of angles with small holds throughout.

All three experiments show that our procedure can recreate a variety of intricate motions.

Table 2.1: Tracking errors for the three hardware tests, expressed in absolute error $|\varphi^* - \varphi_{HW}|$.

Trajectory	**Mean**	**Median**	**90% Percentile**
Validation	3.44°	2.33°	7.43°
T&R 1	3.90°	2.73°	10.61°
T&R 2	5.27°	5.09°	9.20°

The mean tracking error between φ^* and $\varphi_{HW/RPT}$ remained relatively small (3° to 5°, Table 2.1) in comparison to our inexpensive sensor's capabilities. *In situ* we typically observed around 2° measurement error using this Bendlabs sensor, consistent with the accuracy noted in prior work [61]. Tracking errors were larger in all tests in regions where either (a) the actuators were at low temperatures, during the warm-up period, (b) the desired angles were large, at the limits of our calibration range, or (c) the motions are very dynamic, and our calibration assumptions are violated. Both overshoot (at fast motions or large angles) and undershoot (around $\varphi = 0$) were observed.

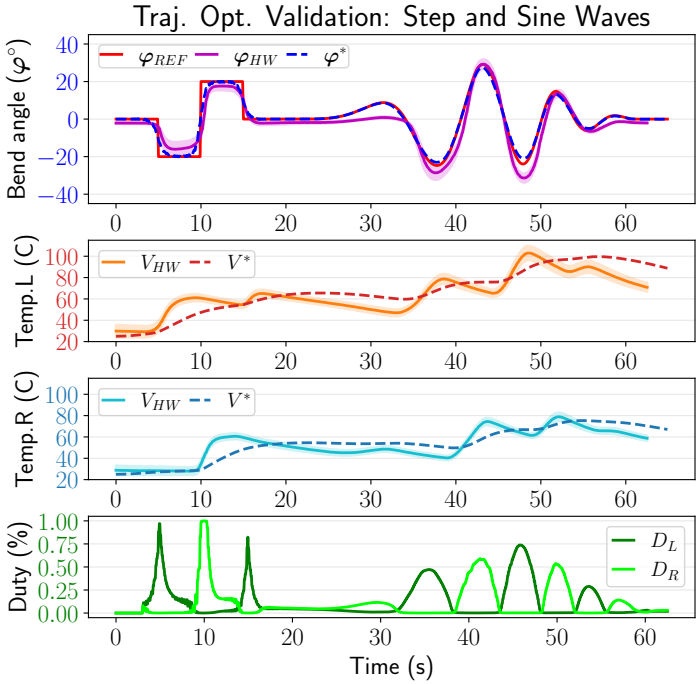
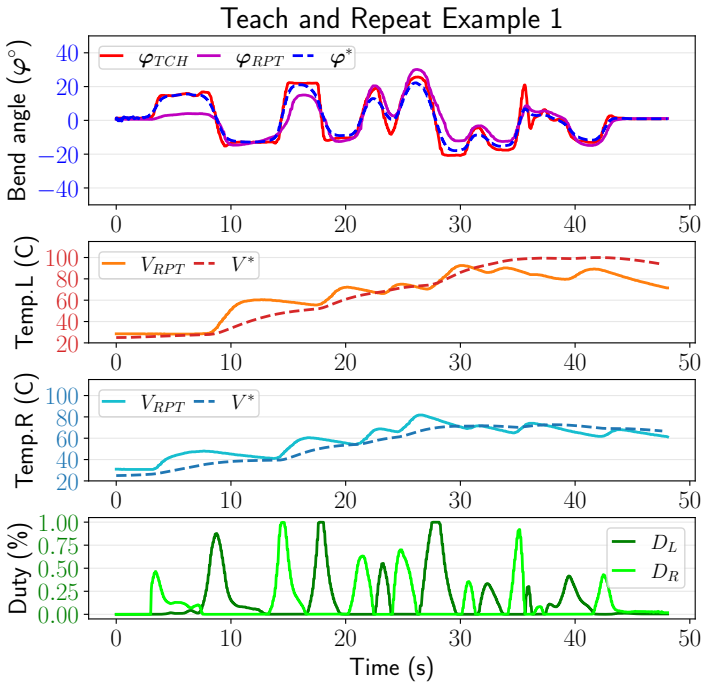
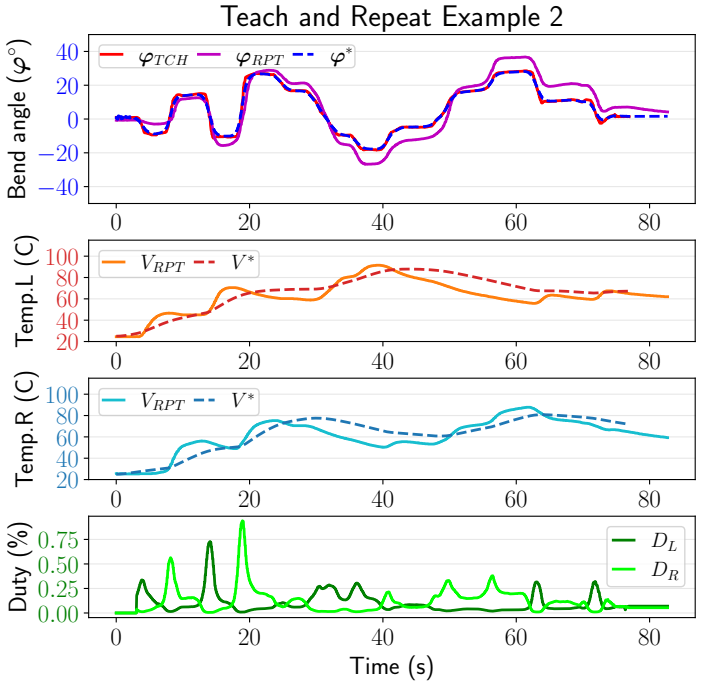


Figure 2.7: Trajectory optimization and hardware validation for step inputs and a decaying sine wave. Optimization produced the feasible state trajectory φ^* (blue dashed) including expected SMA measurement temperatures V^* (dashed). Five hardware rollouts (φ_{HW}, magenta) qualitatively align with the optimized trajectory.

The measured temperatures V_{HW} were similar in magnitude to the optimized state trajectory V^*, but considerably more dynamic, indicating some mismatch in the relationship between the measurement (V) versus wire (T) temperatures. This may be expected, since



(a)



(b)

Figure 2.8: Two teach-and-repeat tests show that our method can recreate desired motions with relatively small error (Table 2.1). Temperatures (orange, cyan) are more dynamic in hardware, since we model V with a lag.

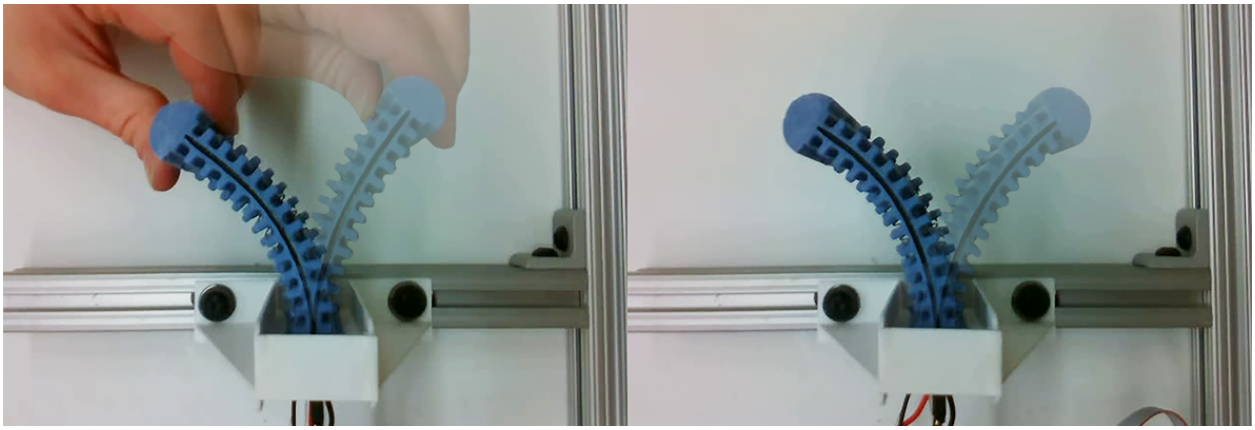


Figure 2.9: Teach and repeat tests show the use of our trajectory optimization procedure. Given an infeasible demonstration from moving the limb by hand (left), our optimization returns a feasible state/input result, which was tested open-loop in hardware (right).

the calibration from Sec. 2.5.3 optimizes for bend angle, not temperature alignment.

2.8 Discussion & Conclusion

This article demonstrates the first example of generating and optimizing motion trajectories for a soft thermally-actuated robot limb. Our approach does not require external sensing nor computationally-challenging models, and open-loop control is simple to integrate with minimal electronics. With this method, a single-segment limb performed open-loop tracking of a trajectory, including re-creation of 'teach and repeat' motions. In addition to showing proof-of-concept, the errors observed in Table 2.1 are reasonably small in comparison to range of motion. For example, with T&R 2, median absolute error vs. angle range is $\approx 5°/50° = 10\%$. Since there is growing evidence that soft robots use *embodied intelligence* to compensate for imprecise motions through their intrinsic mechanical compliance and deformation [54], our approach may be sufficient for many tasks such as locomotion [82].

2.8.1 Limitations

The methodology in this article is designed for planar soft robots, and is only verified for a single-segment soft robot with two actuators. A 3D implementation would require changing the calibration procedure, and multi-segment limbs pose challenges including greater computational complexity (with more discretized links) and propagation of modeling errors. However, research exists on both 3D calibration [42] and multi-segment soft robot modeling using a discretized manipulator [34], which we may adapt for future work.

Our results show that average wire temperatures increase throughout the execution of a trajectory, mostly due to slow convective cooling of the thermal actuators. Our approach may therefore be limited for long-term operation due to the T_{max} constraint in eqn. (2.19). Stiffness is also increasing, which may require increased power consumption, and sacrifices compliance. Active cooling may reduce these effects [21], though at significant design cost, similar to issues with pneumatics or cable actuation. Despite these limitations, our approach applies as-is to untethered soft robots with minor sensing additions (such as in [81]) which also use SMAs.

2.8.2 Sources of Error

Our approximations, which make proof-of-concept trajectory optimization possible, also introduce modeling error. The linear thermal actuator model is particularly simplistic, since the motion induced by a temperature change was nonlinear at low temperatures. The warm-up constraint in eqn. (2.20) only partially avoids this problem. Using instead a constitutive model of shape-memory materials would capture the energy absorption due to phase change. In addition, while the measured (V) vs. wire (T) temperature model in eqns. (2.6)-(2.7) was needed for our sensor design, it effectively adds a low-pass filter to temperature, causing less dynamic temperature predictions in the hardware tests (Fig. 2.7, 2.8). However, our goal is task space (φ) tracking performance, which showed similar response times between hardware and simulation.

Our test setup also inherently introduces imprecision. The gravitational loading for the spring constant k calibration routinely showed $\approx 2^\circ$ differences between hardware trials, vs. a 10° total deflection. Future work will improve calibration procedures. Other assumptions, such as a known ambient temperature T_0, may be eliminated with more sensing.

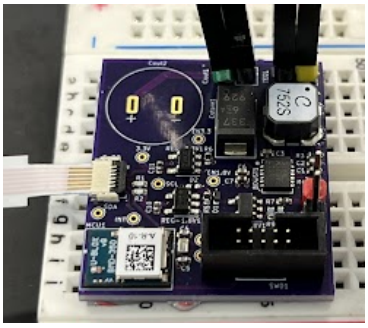
Chapter 3

Battery-free sensing using soft thermoelectric generators

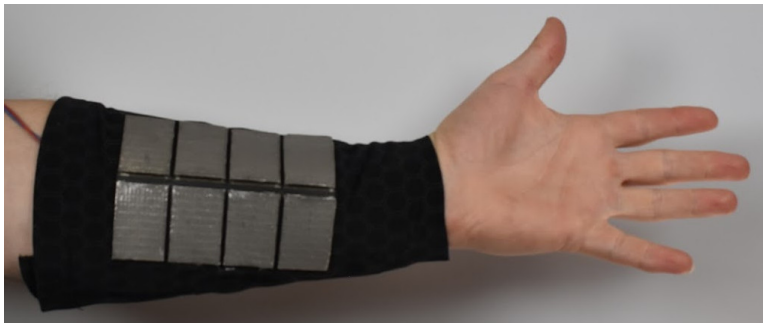
Sophisticated functionality in soft robotics requires some level of compute and processing. Often, this is done offline or off-board on a separate device to limit the complexity of design and integration in the soft robotics platform itself. However, to enable greater robot autonomy and untether such devices from a bench power supply, a better integration of the compute and sensor electronics is necessary. This is also true of wearable devices, which can perform better with some level of built-in compute capability. These devices almost universally need to be untethered to avoid restricting movement. Most untethered devices operate on batteries, but as the load-carrying capabilities of soft robots can be quite low, it is desirable to minimize the added power required for added compute. Similarly, in wearable applications, low-power devices are desirable to limit the size of device batteries, which add bulk and affect device comfort. For this reason, low-power electronics are of great importance for such applications and represent an important enabling technology for tighter device integration.

For wearable applications, we can take this concept further: batteries require space, use precious resources, and need to be recharged or replaced at regular intervals, all undesirable qualities. This means that operating them can be inconvenient at times and impossible in other scenarios. However, it turns out that our bodies can provide plenty of energy in the form of heat that would otherwise be wasted. In effect, with proper design of low-power electronics coupled with energy harvesting technologies, we can demonstrate significant capabilities on-device without any battery at all. One reason wearables have not really capitalized on this free energy is that the means of harvesting it have been difficult and inefficient with existing technologies.

This space offers a unique opportunity to showcase an excellent use-case for soft systems:



(a) TEGSense PCB.



(b) TEG sleeve.

Figure 3.1: Thermoelectric generator powered, battery-free, photoplethysmography sensor. The PCB is shown in (a), with the sleeve in (b) with generators attached.

comfortable, biocompatible, and, importantly, conformal interfaces with the skin. My collaborator Mason Zadan developed soft thermoelectric devices (TEDs) that can be worn to harvest energy from wasted body heat. In this scenario, they are typically referred to as thermoelectric *generators* (TEGs). Commercially available TEGs, perhaps better known as Peltier coolers, are typically made with ceramic interfaces that limit their ability to make close contact with the skin. This is necessary to ensure efficient heat transfer through the TEG, which is how it is able to harvest waste heat to produce electricity. However, the device alone cannot be used directly to power electronics as it typically outputs a very small voltage on the order of tens of millivolts.

As part of our collaboration, I developed an electronics (figure 3.1a) platform capable of using the low voltage generated by an arm-worn array of TEGs (figure 3.1b) to continuously power a biometric sensor and wireless communication, completely battery-free. In this way, we use the electronics to enable some new and interesting functionality of these soft devices. The sensor, which we term TEGSense, was based on a low-power motion sensing platform that I developed previously (Posey). In this chapter, I will discuss both and highlight some of our achievements with TEGSense. The work contributed to the following articles:

- Zadan, Mason, Dinesh K, Patel, Andrew P, Sabelhaus, Jiahe, Liao, Anthony, Wertz, Lining, Yao, Carmel, Majidi. "Liquid Crystal Elastomer with Integrated Soft Thermoelectrics for Shape Memory Actuation and Energy Harvesting". *Advanced Materials.* (2022): 2200857.
- Zadan, Mason, Anthony, Wertz, Dylan, Shah, Dinesh K, Patel, Wuzhou, Zu, Youngshang, Han, Jeff, Gelorme, Hing Jii, Mea, Lining, Yao, Mohammad H, Malakooti, others. "Stretchable Thermoelectric Generators for Self-Powered Wearable Health Monitoring". *Advanced Functional Materials.* (2024): 2404861.

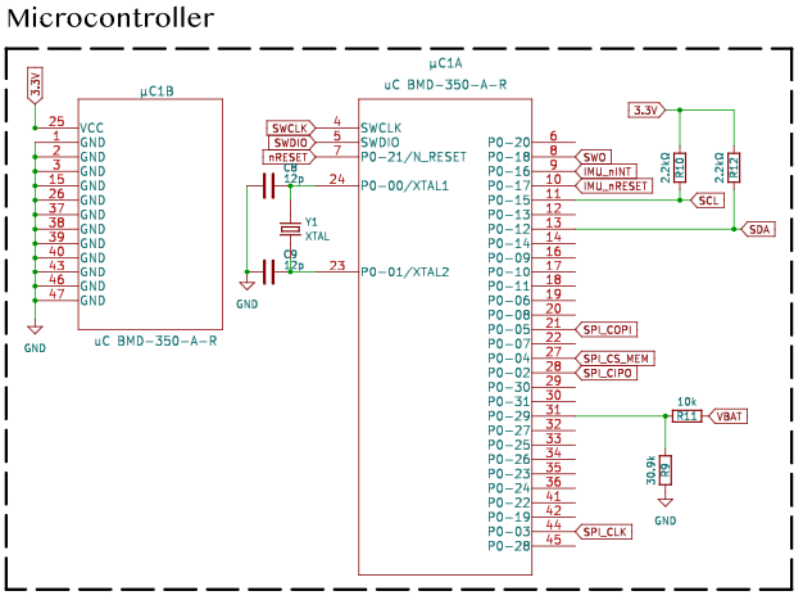


Figure 3.2: Microcontroller schematic.

3.1 Core low-power electronics framework

To ensure both autonomy and power efficiency, the hardware framework is based on an efficient family of Arm Cortex-M4 and -M33 microprocessor units (MPUs) developed by Nordic Semiconductor: nRF52832, nRF52840, and nRF5340. These were chosen for their capable processors, exceptional vendor support, and strong open source commitment. The primary modules used are the u-blox BMD-350 (nRF52832), BMD-380 (nRF52840) and NORA-B126 (nRF5340), which combine the MPU, a chip antenna, and all necessary support components in a single module with optimized radio performance. Bluetooth low energy (BLE) is the primary communication link between devices and an operator, so understanding radio performance proved to be critical to robust operation. The size and continuity of the ground plane on the printed circuit board (PCB) had to be taken into account to achieve good radiation performance in such small form factor designs. For lower power consumption, an external low-frequency crystal oscillator is added. For device connection, I2C resistances are chosen so that the device has a short rise-time to support higher-speed operation (400kHz) while minimizing power usage, although SPI is preferred where possible. The main microcontroller schematic is shown in figure 3.2.

Lithium polymer (LiPo) batteries are used to provide high power density and rechara-bility. A thermistor is used to monitor the battery temperature for safe charging and use. Charge control and power delivery are achieved using a integrated controller and power path management chip (Texas Instruments BQ24072) to properly charge the battery and manage

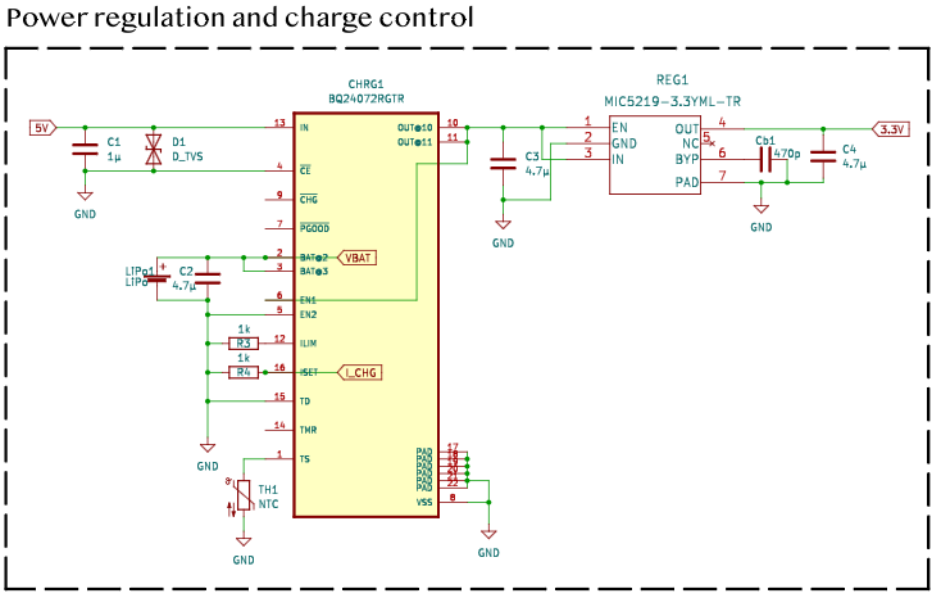


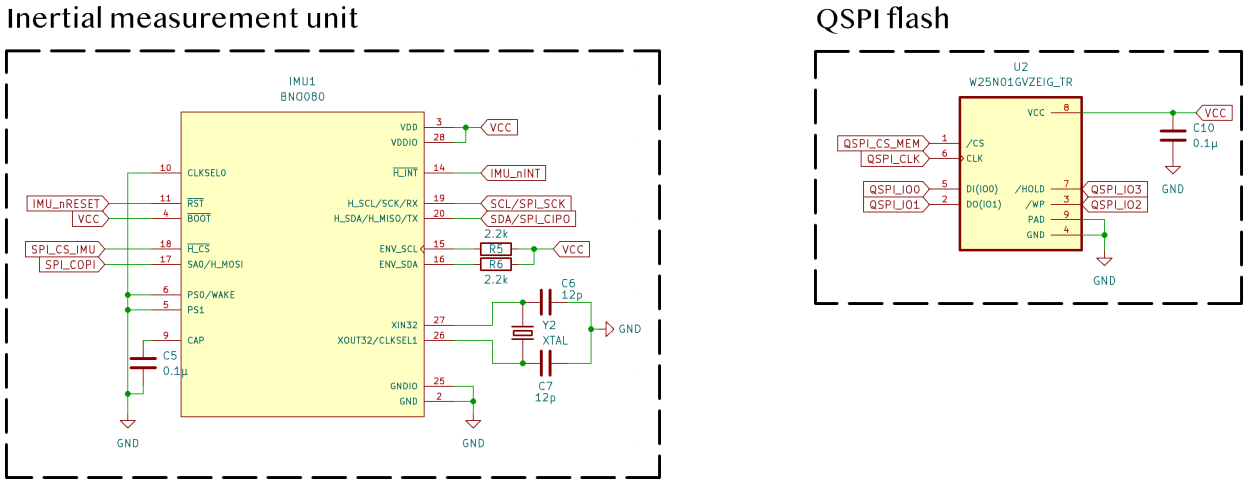
Figure 3.3: Power path schematic.

power delivery whether the input is connected or not. Transient-voltage-suppression (TVS) diodes are used to protect against electrostatic discharge (ESD) and voltage spikes, especially when connecting and disconnecting external power. A schematic of the power path and the 3.3 V regulator is shown in figure 3.3. Power utilization can be reduced further by the use of a lower voltage switched power supply instead of a linear regulator.

The platform firmware is built on the open source Zephyr real-time operating system (RTOS). BLE is used for low-power, high-throughput communication, telemetry, and localization using external beacons. The bulk of the firmware is developed in C and C++, with Python bindings built to interface with the control and data collection software.

3.2 Incorporating inertial measurement units for motion tracking

On top of the core platform we added to each sensor a 9 degree-of-freedom (9-DoF) inertial measurement unit (IMU, CEVA BNO086) to capture accelerations and fused pose estimates at $50-200$ Hz (schematic in figure 3.4a), although the platform can be tuned to support higher rates. The firmware scans for external iBeacon advertisements which can be used to track gross movement throughout a research environment. Data is logged to on-board flash (schematic in figure 3.4b) and can be downloaded at the end of the day. The system design was designed to be low power to facilitate high-rate tracking throughout the day



(a) IMU schematic. (b) Flash schematic.

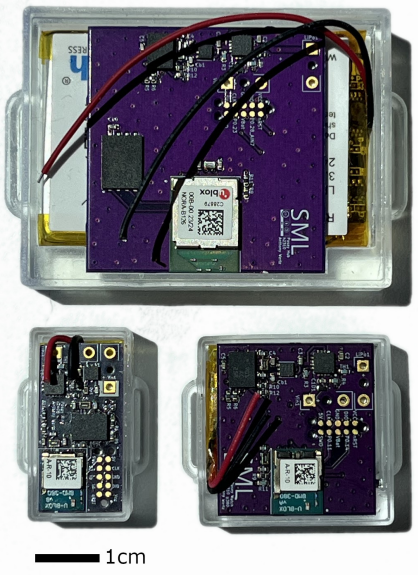
Figure 3.4: Peripheral schematics for the (a) inertial measurement unit (IMU) and (b) flash storage.

with reasonably small batteries: around 60 mAh for the ring, 290 mAh for the watches, and 970 mAh for the waist.

Figure 3.5a shows photos of the three sensors that were developed. The largest (top) has the most powerful microcontroller, including a second core specifically designated to handle the BLE traffic efficiently. It was meant to be used as a hub through which all sensors can connect and log data to the on-board NOR flash. This is why it requires a large battery, although its physical dimensions are still smaller than a deck of cards (about 55 mm × 42 mm × 13 mm). The watch and ring have almost equivalent hardware, except that the ring uses the BMD-350 (nRF52832) and the watch upgrades to the BMD-380 (nRF52840). The modules are round the same size and require about the same amount of power, so in future revisions the ring can be transitioned to use the upgraded module as well. The main reason for using two separate boards as opposed to simply recycling the ring PCB for the watch is to increase the size of the ground plane to improve radio performance. Because the modules use chip antennas, radio performance is largely dictated by the size and quality of the ground plane. Although external antennas could be used, they are typically quite large.

In any case, the size is still kept minimal for the two devices. The watch measures about 28 mm × 28 mm × 11 mm and the ring 28 mm × 15 mm × 10 mm. A photo of the ring device placed on a finger is shown in figure 3.5b for a better sense of scale.

Each of the three types of devices (ring, watch, and hub) was designed with the same core framework. However, as the complexity of the on-board computation increases with each device type, separate but compatible microcontroller modules were used in the most recent update of the devices, currently under development. For the ring, the original u-blox BMD-350



(a) Sensors assembled and in their enclosures

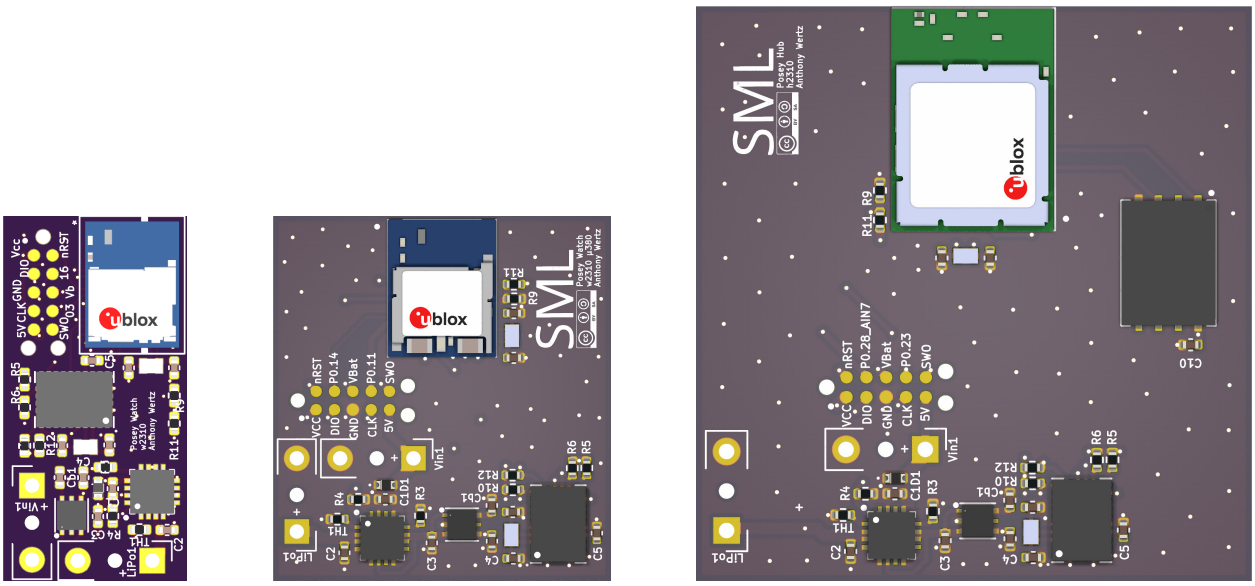
(b) Ring sensor on hand for scale.

Figure 3.5: Photos of the assembled sensors (a) and the ring sensor on a finger (b).

(nRF53832 ARM Cortex-M4) was used. For a marginally large footprint (7.5 by 9.5mm compared with 6.4 by 8.65mm) the u-blox BMD-380 is used (nRF53840 ARM Cortex-M4) to expand the on-board flash and RAM footprint, and boost the antenna performance. The hub unit utilizes the u-blox NORA-B126 (nRF5340 dual-core ARM Cortex-M33) with a power amplifier and low-noise amplifier at the antenna front-end to improve processing capability and device sensitivity. All of these devices operate at low power and, for this sensor suite, they are interchangable, with no firmware changes required for operation. This demonstrates the flexibility of this configuration for many applications, as the processing power can be easily scaled to meet the requirements of the specific application with minimal changes to the design. CAD renderings of the board designs are shown in figure 3.6

3.2.1 Stroke rehabilitation

The original purpose that spurred the development of these sensors was as part of a collaboration with researchers at Virginia Tech. The suite of motion sensors I developed was intended to track rehabilitation progress for stroke victims with loss of function in the upper extremities. The sensors included a waist-worn hub unit about the size of a credit card used to track hip motion and collect telemetry from peripheral sensors; two wrist-worn watches; and one ring. Analysis of the wearable sensor data would be used to identify where lost function is improving and where it is not, so that therapy can be tuned on the fly. This real-time



(a) Ring sensor (b) Waist sensor (c) Hub sensor

Figure 3.6: CAD renderings of the ring (a), waist (b), and hub (c) sensor PCBs.

personalization is expected to improve patient outcomes by reducing the time required to regain strength and dexterity, and improve function by identifying and focusing on areas of weakness.

Surprisingly, reliable inertial measurement units for high-throughput, low-power, and small-footprint design were not available commercially. The available sensors are expensive and provide little control over how the operation can be tuned for different purposes, for example adding in the iBeacon tracking. The Posey framework has thus contributed a completely open and reproducible framework to the research community for future studies that require some level of motion tracking. This collaboration is still in progress, but we hope it begins to bear fruit and contribute to the greater community soon.

3.2.2 Assistive feeding robots

In a new collaboration with Akhil Padmanabha (Soft Machines Lab and Robotic Caregiving and Human Interaction Lab, Carnegie Mellon University), the small form factor and research-ready design of the Posey devices found a new purpose. Multiple Posey IMUs will be combined in a wearable platform to improve the ability of a robot assistant to feed someone unable to feed themselves. The expectation is that sensor data will enable better prediction of when the user will want the next bite of food. This would be more similar to how a human assistant would conduct the task, rather than relying physical or verbal commands from the operator to initiate the next event.

3.3 Battery-free sensing and communication

Soft devices can be advantageous in wearable applications due to their ability to conform to irregular shapes and maintain closer contact with the skin. This leads to more comfortable wearables and less contact noise for sensing. However, as suggested at the beginning of this chapter, this property is also an *enabling* quality for some applications. In the beginning of this chapter, I briefly discussed the potential for thermoelectric devices to be used in wearables to harvest heat and generate electricity. This depends on a good thermal interface between the skin and the TEG, which is difficult to achieve with the commercially available rigid devices. In collaboration with Mason Zadan, who developed soft and flexible generators in his doctoral research, I developed the electronics platform necessary to take full advantage of these devices. The platform is based on the work in the Posey project with the addition of a PPG sensor and step-up converter to enable energy harvesting from a low-voltage TEG input.

3.3.1 Electronics for efficient harvesting from TEGs

Using flexible TEGs as the starting point, I developed the electronics necessary to implement a fully body-powered biometric sensor, which we refer to as TEGSense, capable of capturing three-second red and infrared (IR) photoplethymograph (PPG) waveforms using an off-the-shelf photonic sensor (Analog Devices MAX30101). It is built on the same core platform described previously but skirts the battery or charge controller and instead uses a low-voltage DC/DC step-up converter (Analog Devices LTC3108) to extract energy from an array of thermoelectric generators woven into a wearable sleeve and in direct contact with the body to charge a bank of capacitors. Briefly, a thermoelectric generator is a transducer capable of converting a temperature differential across two sides of the device into an electrical potential, and vice versa. In this way, electricity can be generated from the difference in temperature between the human body and the environment. A schematic of this step-up converter with the TEG input is shown in figure 3.7. When sufficiently charged, a hysteretic voltage switch (Microchip MIC2779) powers the microcontroller and sensor to capture the PPG waveform and telemeter it over BLE. The TEGs are flexible liquid-metal-based devices described by Zadan, Malakooti, and Majidi [132].

The device (figure 3.1a) is worn on the TEG-embedded sleeve (figure 3.1b) so it can harvest waste heat from the skin to power the finger-worn PPG sensor and the BLE-enabled microcontroller to enable continuous operation without a battery. The board is shown in figure 3.1. We also used an Arduino to collect the input voltage of the TEG and the output

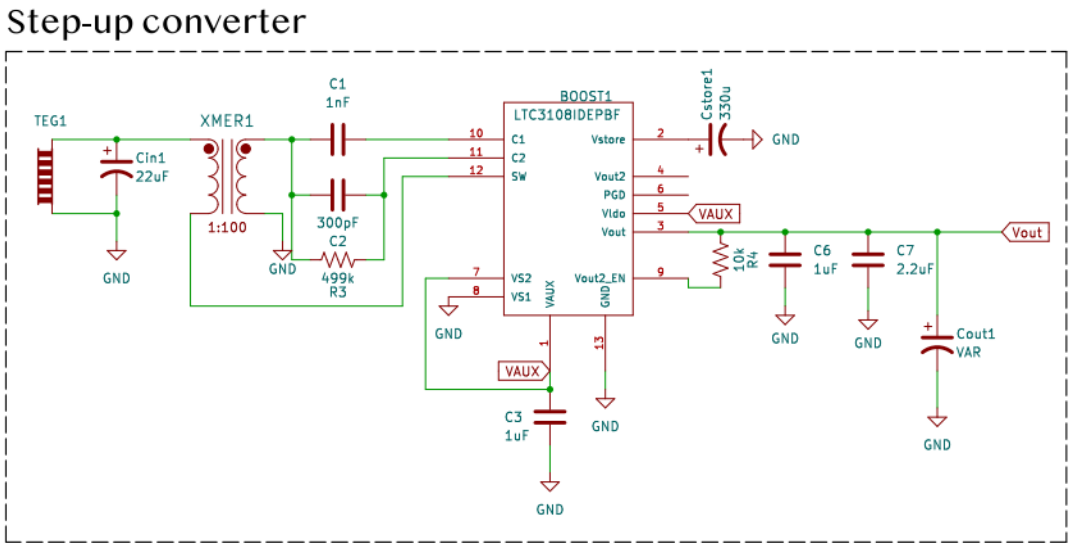


Figure 3.7: TEG step-up converter and capacitor charging.

voltage of the capacitor bank during experiments, since the TEGSense board is asleep during charging and therefore cannot be used to collect these measurements.

There were a number of challenges in the design of this device. The effective series resistance (ESR) of the capacitors turned out to be of great importance. Originally, an electrostatic double-layer capacitor (ELDC, or supercapacitor) was used to store a large amount of energy with a small footprint. However, the ESR is relatively large, on the order of a few hundred ohms, making the high current load required when the microcontroller switches on difficult to supply without browning out the supply voltage. Aluminum polymer capacitors with ESR on the order of tens of milliohms were used instead, first in parallel to the ELDC, then alone in a bank. The high ESR of the ELDC caused an excessive power loss when the energy was to be used, so we decided on removing it completely. Another challenge was to ensure that the voltage-regulating devices had a very low quiescent current to avoid significant leaching of the charge collected in the capacitors during charging. Due to the very low power generation, even regulators with off-currents on the order of tens of microamps were too high. Furthermore, because a significant portion of the board is completely powered down during operation, instead of being held in a deep sleep, care had to be taken to avoid reverse current leakage through the microcontroller, a problem manifested in an earlier revision of the design.

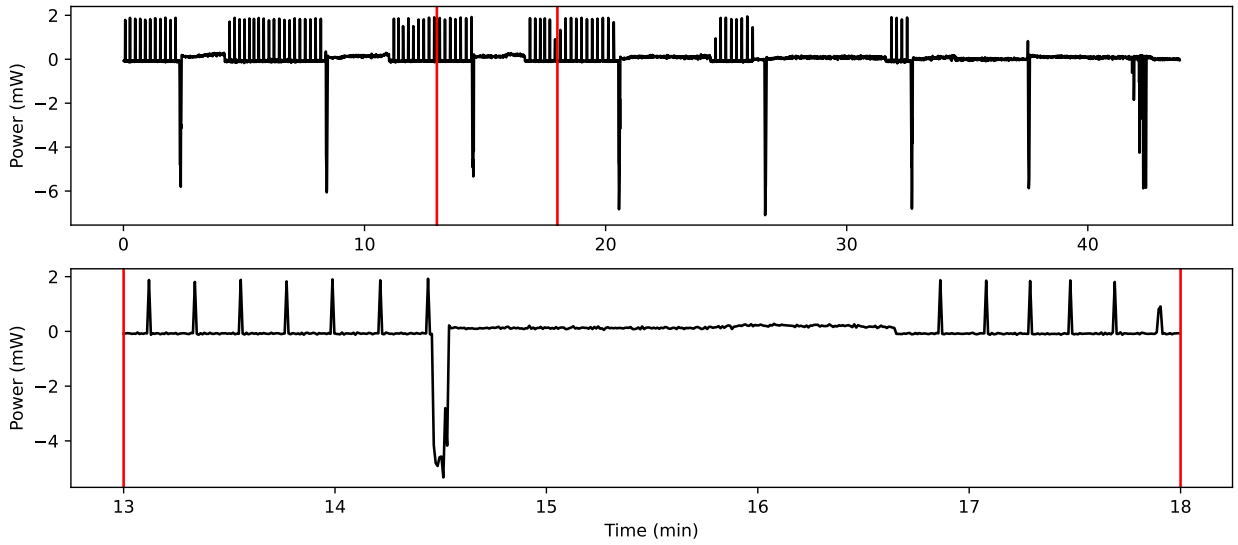


Figure 3.8: PPG sensor power utilization.

3.3.2 Waveform collection and transmission

When in operation, the device buffers three seconds of red and IR PPG waveforms. In parallel, it advertises the TEGSense service and waits for an external device to establish a connection. Once connected and after the sensing is complete, the raw waveform data is sent to the external device via BLE. The device disconnects and then goes to sleep. In the event that the voltage drops below the hysteric cut-off at 3.3 V, the power to the sensor and microcontroller is completely cut off. Although devices can run at a lower voltage, we always wanted 3.3 V supplied to the PPG sensor to ensure that it would operate consistently across collection events.

Figure 3.8 shows the power draw of a sequence of sensing and telemetry events (top) and the zoomed view of a single event (bottom). The power disparity is very apparent in this plot, showing why this has been a very difficult task using current wearable generators: for any moderately power-consuming device (of which a photonic emmitter and receiver is one), the ability of wearable devices to use energy typically far exceeds a generator's ability to collect it. Despite the large footprint, the TEG array is only generating between 100 – 250 µW, whereas the sensing and data transmission require around 4 mW. This goes to show why reducing power utilization and especially energy leakage wherever possible is so important to a device like this. Even a voltage regulator with a "low" quiescent current of around 10 µW consumes 10 % of the generated energy.

Figure 3.9 shows the PPG waveforms collected over BLE. From these waveforms, we are able to easily detect the user's heart rate, which was perhaps a bit higher than usual walking

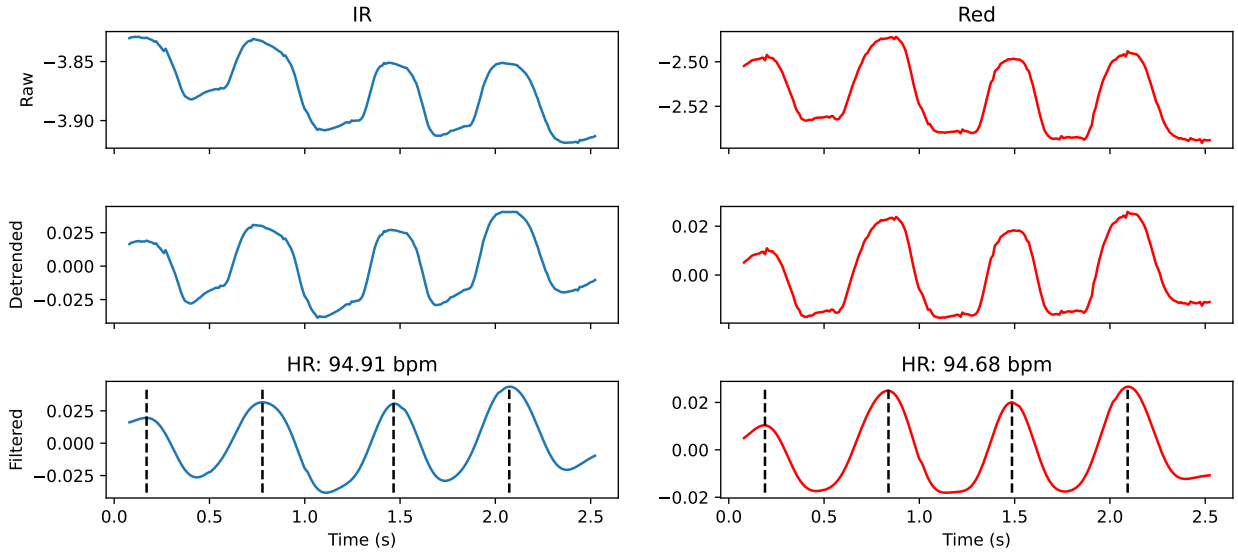
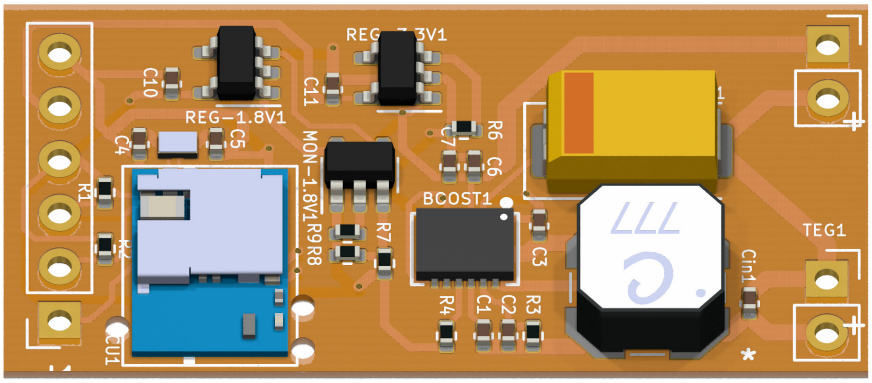


Figure 3.9: PPG sensor power utilization sensor readings.

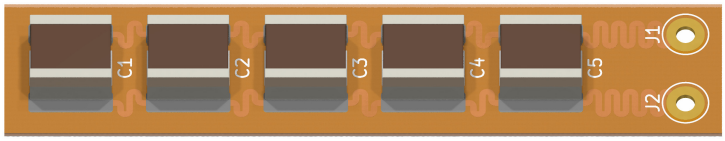
back and forth briskly in the cold. Blood oxygen saturation becomes a bit more difficult to determine because three seconds is a very short time frame to collect enough cardiac cycles to derive a robust estimate. But this at least demonstrates the framework set in place to enable this operation. For more information, see [133]

3.3.3 Conclusions and future directions

In this work, we demonstrated that thermoelectric devices, when redesigned with a novel soft material architecture, could be used as wearable energy harvesters to continuously power biometric sensors entirely battery-free. We demonstrated photonic sensing and wireless communication powered completely from energy harvested from the body, a feat that could not be accomplished with previous generations of wearables. The was a considerable challenge because both PPG sensing and wireless communication are relatively energy intensive tasks. Depending on the temperature gradient between the user's body and the environment, the TEG output voltage does not necessarily reach a level sufficient to charge the capacitors in a reasonable amount of time. This is despite the large footprint. Some work could be devoted to improving the device design with TEGs to better extract energy on the body. Additional work could be done to perform lower power PPG sensing, either through a custom designed circuit or through new application-specific integrated circuits (ASICs) that come to market. Finally, better integration into the glove can be achieved by constructing flexible or even stretchable circuits that can be more easily embedded into the fabric. Figure 3.10 shows CAD renderings of a flexible variant of the TEGSense sensor and capacitor bank that might



(a) Flexible TEGSense variant.



(b) Flexible capacitor bank variant.

Figure 3.10: Reconfiguration of the TEGSense circuit for use in flexible device. The circuits are laid out on flexible copper-clad polyimide with large vias to make it easier to connect printed liquid metal ink interconnects.

be integrated into future sleeve designs using liquid-metal-based ink interconnects.

Part II

Integrating materials and electronics in design

In part I, the design of a complete soft system was based on separate, predominantly isolated designs of the soft components and the accompanying sensing and electronics. But this approach had some shortcomings. Devices designed this way quickly became bulky, negatively impacting the material properties. Also, the level of sensorization achievable post hoc was typically not sufficient to fully exploit the soft system's capabilities. In the chapters that follow, I will explore a different avenue, one in which material systems are designed more holistically as part of an electromechanical system. Advanced materials can enhance the capabilities of the system electronics and vice versa. However, this must not overly complicate the system design or fabrication, which often happens when electronics need to be incorporated after-the-fact. The intent of this work is not to fly in the face of modularity as a guiding principle in design, which is a very important concept needed to enable complicated devices. Rather, I will suggest that, at least when it comes to soft systems, some level of application-specific design and integration is necessary to enable more advanced capabilities.

Chapter 4

Encapsulation of stretchable sensors for enhanced functionality

In chapter 2 I described how a soft robot limb could be modeled and controlled with a single bend angle given by an embedded capacitive bend sensor. However, I also pointed out a major pitfall of this approach: a single bend angle provides a disappointingly incomplete measure of robot state. Restrictive assumptions about the form deformations can take must be made. For example, we used a beam model in the calibration under a gravitational load, but a constant-curvature model was assumed for actuation. This process does not generalize well to other scenarios in which those assumptions become invalid, especially in the face of contacts or other external forces. What would really be desirable would be a number of bend sensors along the axis of the robot to arrive at a distributed measurement of the robot state, as opposed to a total bend angle. Adding additional sensors becomes tricky: they are bulky, each sensor requires more wiring, and the device complexity starts to grow pretty quickly.

In this chapter, I propose a framework for enabling greater incorporation of these materials with intrinsic sensing capabilities in a way that enables distributed sensing with sensor tunability and response isolation, all while maintaining simple connectivity and minimizing the effect on the device's material properties. Furthermore, I demonstrate a topology that supports distributed bend sensing at multiple nodes with minimal design complexity and only a four- or five-wire interface, as shown in figure 4.1. As it turns out, this is a very general approach that not only works with different filter designs, here I will discuss two notch filters, but also with different sensors. Stretchable strain gauges, printable pressure sensors, and thermistor temperature sensors can all be placed on the same, shared interface, greatly simplifying system integration in soft systems that need numerous sensors.

Figure 4.1: A stretchable strain gauge built using a simple notch filter.

4.1 Overview

This work proposes the use of a narrow-band notch filter with a strain-responsive notch frequency for use in the detection of material strain, especially for soft-material systems. The design builds on the rich and growing body of existing research using strain-responsive capacitive and resistive materials, providing a highly customizable framework on which to build distributed sensing systems. This work investigates how incorporating strain-sensitive components into a notch filter offers great flexibility in tuning the sensor's strain sensitivity independent of the particular capacitor or resistor design. Additionally, encapsulating sensing elements inside a notch filter offers a straightforward means of connecting multiple sensors in a sensing network with minimal wiring and interface requirements. The active filters used are small and simple enough to fit in a small area with minimal rigid components, making them ideal for incorporation into highly stretchable or bendable soft systems. Direct ink writing (DIW) fabrication can be used to further reduce rigid components.

4.2 Theory of operation

A *notch* filter refers to a band-reject filter (alternatively, band-stop) with a high-attenuation and a narrow bandwidth signal rejection. An ideal notch filter would not attenuate any

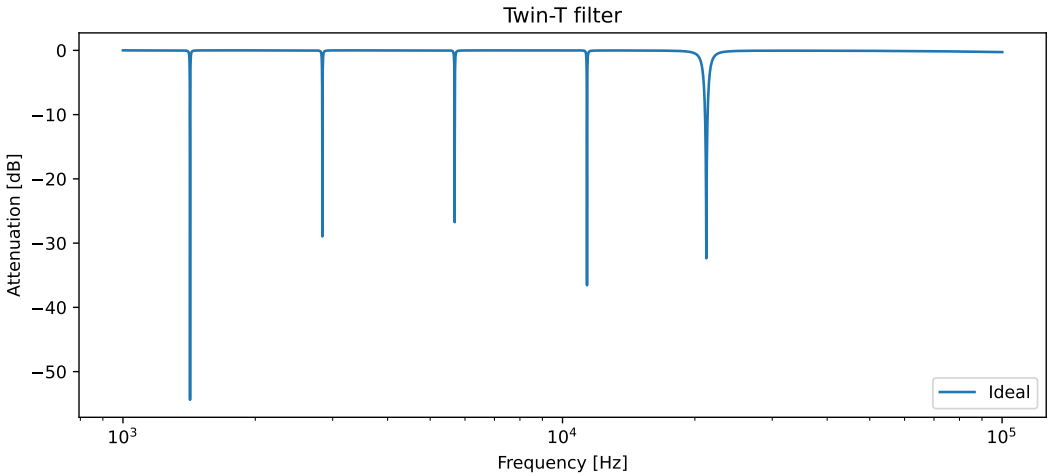


Figure 4.2: Transfer function of five spectrally isolated notch filters using an active twin-T notch topology and ideal operational amplifier model.

frequency components except those in a narrow range, in some cases a single frequency. A familiar example would be rejecting main-line switching interference from a signal, where a 60 Hz would be desirable in the US. An infinitesimally small bandwidth is not realizable in practice, but the degree to which this constraint is satisfied in a practical circuit can be quantified using the *quality factor*, or Q-factor. This and other common filtering concepts can be understood more thoroughly with any number of analog filtering textbooks, such as that by Williams [125]. The Q-factor can be interpreted as the filter frequency over the half-power bandwidth, i.e.,

$$Q = \frac{f_0}{\mathrm{BW}_{3\,\mathrm{dB}}}. \tag{4.1}$$

A narrower bandwidth, indicating a tighter notch, is indicated by a higher Q-factor. If the notch frequency is unknown, it can be estimated by looking at the magnitude response of the transfer function, as in figure 4.2.

In a typical filter implementation, the frequencies to notch out are static, as in the case of the mains hum rejection. Designing that notch frequency to be strain-responsive instead, by replacing particular passive components with strain-responsive variants, allows inference of strain on those components by determining the new notch frequency and comparing it to the zero-strain baseline. Separation of the notch filters spectrally, which can be accomplished by the appropriate selection of the fixed components, allows for the isolation and identification of sensor responses in a distributed network. Additionally, the full network can be interrogated with a small frequency chirp using only four or five wires.

Using two well-known notch filter topologies, the subsequent sections describe in some

detail how an individual sensing cell ("sencel") can be implemented, how the strain-response response can be tuned, and how multiple sensors can be used together on a single interface. Figure 4.2 shows a simulated transfer function for a network of five sensors connected on the same network.

4.3 Background and related works

Distributed sensor networks are especially important in soft electronic skin and soft robotic applications. Yin, Hellebrekers, and Majidi show the utility of multimodal sensing in closed-loop manipulator control [129]. Dou et al. discuss the need for distributed sensing in soft robot manipulators to infer robot configuration and enable tactile sensing over a large contact area [27]. The work of Truby, Della Santina, and Rus shows just how difficult it can be to infer the state on a continuum manipulator [117]. Even after a kinematic model has been derived, machine learning is still needed to achieve accurate proprioception. Shah et al. further demonstrate the utility of sensor integration in soft robotics applications [98]. Soft or deformable robots have many more possible state configurations compared to traditional rigid devices, so dense sensorization is often necessary.

Similarly, distributed sensing is useful in electronic skin applications as Shih et al. discuss in [100]. In part, distributed networks enable sensing over a wide area or complicated geometry, useful in virtual reality and haptic applications, as indicated by Yin et al. [130], and tactile sensing per Roberts, Zadan, and Majidi [89]. It also enables estimation of the shape of the object, which is important in a variety of biomedical applications and more generally applications dealing with biological organisms that take on varied and irregular forms. Shah et al. implemented shape-sensing sheets by combining inertial measurement units and capacitive sensors at nodes along a planar substrate [97]. However, as they noted, it can be difficult to scale such systems due to addressing or interfacing challenges of the discrete sensor chips, requiring different schematics and circuit board layouts for a particular configurations.

There is plenty of prior work on soft, stretchable sensors, especially in the area of capacitive and resistive sensing technologies. A prevalent strategy for designing such devices is the use of liquid-metal embedded elastomers (LMEE), as described by Majidi et al. in [68], as it proves to be a versatile tool for the implementation of stretchable circuits and integration of soft devices with rigid counterparts. For example, highly stretchable strain gauges can be designed using LMEE. Nesser and Lubineau review many examples of capacitive strain gauges [76], and Wu et al. discuss resistive and capacitive sensors and strategies to incorporate them into designs [126]. Souri et al. go into even greater detail on the incorporation of such

devices in wearable applications [104].

How exactly to distribute the sensing is a different question offering its own set of challenges. As seen in some of the aforementioned works (e.g., [11, 97]) working with traditional rigid sensor chips distributed across many sensors can be challenging both to wire appropriately and to scale, as many devices have limits to the available addressing, requiring multiple communication buses or complicated multiplexing. In [98] and work by Tee et al. [111], sensors are handled directly without an interfacing chip. This requires a new data wire for each sensor, drastically limiting how many sensors could be physically connected in a particular application and increasing the stretchable wire count, a notable failure point in stretchable electronics. Sensors are often distributed in matrices as Kaltenbrunner et al. did in [44] to reduce the data lines required for interrogation from n to $\sqrt{n}$. However, this is the best case reduction, assuming that the devices are connected in a matrix fashion. Doing so may greatly limit the distribution geometries achievable and might not be relevant at all for oddly-shaped structures as occur frequently in biology. In addition, $\sqrt{n}$ may still be quite large for a small wearable system to interface with.

In other instances, clever mechanical design and signal processing techniques might be used to decouple localized signal contributions and infer a distributed response over a single sensor. White, Yuen, and Kramer measured localized deformation on a monolithic capacitive sensor taking advantage of the reduced electrical transmission with distance when capacitor electrodes were made to have fairly high resistance [123], with a more mathematically rigorous description of this effect by Tairych and Anderson in [109, 108]. Tairych and Anderson derived the analytic expression for three capacitive strain sensors in series, but even with so few sensors, there was considerable cross-talk and scaling up that approach would be quite cumbersome [109]. Similarly, Sonar et al. performed touch localization on a two-dimensional substrate using the same concept [103]. The array can then be interrogated by varying the input frequency and training a machine learning model to classify the individual "pixel" locations. This approach does not easily lend itself to complicated geometries, larger sensing areas, or reconfigurability, and requires data collection and retraining for any new device or geometry. Bai et al. use light-absorbing dyes embedded in a stretchable fiber core to distribute sensing along the device that change the color of the received light depending on where deformation occurred and what kind [7]. The design is greatly limited in the geometry supported. Hellebrekers, Kroemer, and Majidi performs contact localization by modeling changes in the magnetic field with the deformation of a magnetized polymer [38]. A black-box machine learning model is trained on experimental data to associate a modified magnetic field with a mechanical indentation in a particular region of the polymer. There is a limit to the number of sensors that can be connected in a single system, and quite a bit of data is

required for collection as described by Bhirangi et al. [11].

Artificial intelligence methods are commonly used to model soft system sensor behaviors, including hysteresis and fatigue, including work by Kim and Park [48] building on top of more traditional models, such as the Preisach model [105]. As they detailed, the small signals coming directly from typical soft strain sensors are noisy and not always easy to process. Funabashi et al. use graph convolutional networks to utilize tactile information distributed over an Allegro hand [29]. However, such methods typically fail to isolate particular sensors in a single network when their signals mutually interfere and their response depends on the particular network of sensors in a given configuration. Instead, it is desirable to keep the sensor feedback isolated between devices.

In 2022 Kim, Kim, and Park demonstrated a distributed soft sensor network with a single input and a single output that used passive inline LC bandpass filters to selectively interrogate different sensors by varying the frequency of an input sinusoid and analyzing the attenuation of the signal at the output [50]. Strain-sensitive resistive elements were used to modulate the attenuation based on the applied force. However, to maintain sensor bandwidths to reasonably low frequencies, high component values are needed for the inductor and capacitor selection. This is at odds with the increasing impedance of inductors at higher excitation frequencies and higher inductances. Sensor bandwidths in [50] ranged from about 280 kHz to 3 MHz, which requires fairly high-rate acquisition and computationally expensive processing to process raw sensor data. Similarly, Kim, Kim, and Park also demonstrated a modular sensor network for distributed force and human touch detection [49]. The same passive LC bandpass filters were utilized in parallel, each with a force-sensitive series resistance, forming a network of notch filters in which the depth of the notch was modulated by the force contact and the frequency shifted by capacitive effects of a finger in contact. The network suffered from similar issues as the bandpass implementation, namely the operating range varied from 500 kHz to 5 MHz this time. Furthermore, the use of a passive network made decoupling complicated, requiring a compensation algorithm to reduce the effects of new sensors, an approach likely to be brittle in the face of system reconfigurations or component degradation. In both cases, the LC networks offer much more limited options for frequency tuning, as component values for inductors and capacitors in usable ranges are fairly limited and have high tolerances.

In general, existing approaches illuminate the central challenges in distributed sensing for soft robots and wearable devices.

- **Scalability and wiring**: Integrated sensor chips often have finite addressing capabilities, forcing multiplexing, complex PCB layouts, multiple communication buses, or one-wire-per-sensor solutions.

- **Reconfigurability constraints**: Systems that rely on large amounts of training data or fixed hardware addresses can be difficult to scale to larger networks or adapt for new geometries, sensor types, or reconfigured placements.
- **Signal decoupling and interference**: Passive LC filter networks can reduce wiring, but suffer from crosstalk and require careful component selection to achieve reasonable performance.

This work builds on previous efforts to incorporate soft sensors in distributed networks. We develop a framework that addresses three main challenges inherent in previous work:

1. **Active-filtering and buffering**: Each sensor node incorporates a high input-impedance active filter with a buffered output, decoupling each node from all others in the network. This approach eases reconfiguration and supports heterogeneous sensor networks as well.
2. **Tunable notch filter topology**: Strain-sensitive elements are embedded in notch filters whose center frequencies and strain sensitivities can be adjusted independently of the native properties of the sensor, expanding the design space.
3. **Operational frequency in accessible ranges**: By selecting appropriate component values, we achieve sensing performance at much more moderate operational frequencies than previously demonstrated with the passive LC filter designs, allowing for better spectral usage and operation in frequency ranges much more accessible to compute-limited devices.

In this work, we describe the framework design under two different notch filter topologies, derive the device gauge factors and theoretical supported sensor densities, and demonstrate precise strain tracking and full sensor decoupling, even when heterogenous sensor designs are used together and even when sensor bandwidths overlap. In operation, it is very similar to strain detection in fiber Bragg gratings described by Ngiejungbwen, Hamdaoui, and Chen [77] that form notch filters by reflecting particular light wavelengths. However, silica and plastic optical fibers used in these scenarios typically have low yield strains on the order of a few percent and such systems typically are not reconfigurable. Park, Park, and Park try to address tunability by designing multi-material systems [80], but this too is much more complicated than tuning by simply replacing a few passive components in the filter design. The network design described here offers many advantages for both on-body sensing in areas that undergo much larger strains, such as joints, or when wearability is a large factor. This robust design and ease of use may serve as a useful model for future distributed sensor designs that require minimal interface requirements, easy tuning, dense networks, and reliable interfacing.

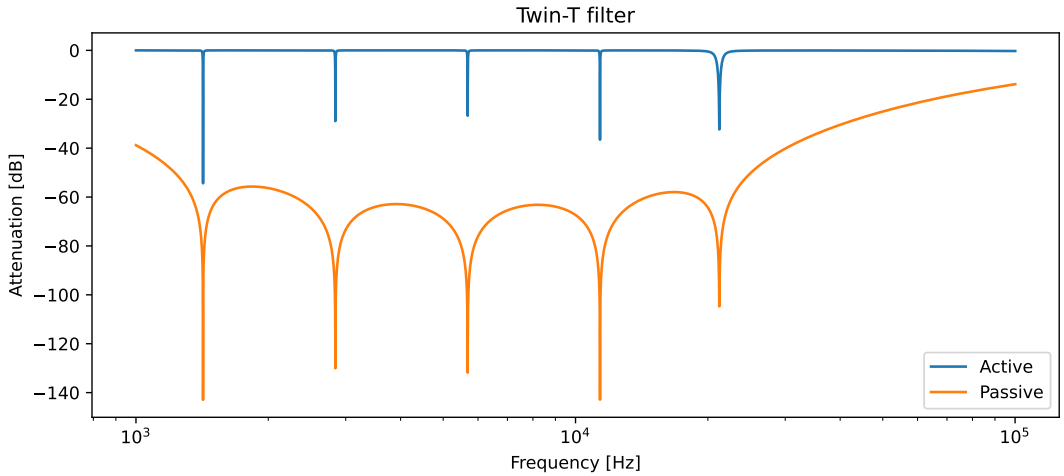


Figure 4.3: Transfer functions for active and passive twin-T notch filters designed with the same notch frequencies.

4.4 Notch filter topologies and theory

There are a multitude of topologies for notch filters that can be used to implement this sensor design. The following is not an exhaustive evaluation, the particular realization is chosen not out of necessity but rather due to particular design constraints. Two common variants are the focus of this section: the active twin-T network and the Bainter notch. While the particulars in implementing a notch do not matter so much, an evaluation of these two options will expose some points to consider when making a determination. An important exception is that of an active versus a passive implementation. If only a few sensors are used in sequence, a passive implementation can still work with the remainder of the approach explained here. Figure 4.3 shows a comparison between an active and passive variant of the same notch network. The main drawback of a passive network is the low Q (limited to $Q = 1/4$ in a twin-T implementation) and substantial broadband attenuation. However, passive topology can still be effective with only a few sensors, enough spectral separation, and sufficient ADC resolution to recover the transfer function.

The general form for a second-order notch transfer function is

$$H(s) = \frac{s^2 + \left(\frac{\omega_z}{Q_z}\right) s + \omega_z^2}{s^2 + \left(\frac{\omega_p}{Q_p}\right) s + \omega_p^2} \tag{4.2}$$

where ω_z and Q_z are the radial frequency and quality factor of the zeros (respectively), and ω_p and Q_p are those of the poles. To form a symmetric notch at a particular frequency ω, it is necessary to design a filter with pole and zero frequencies set to ω. If $\omega_p < \omega_z$, the

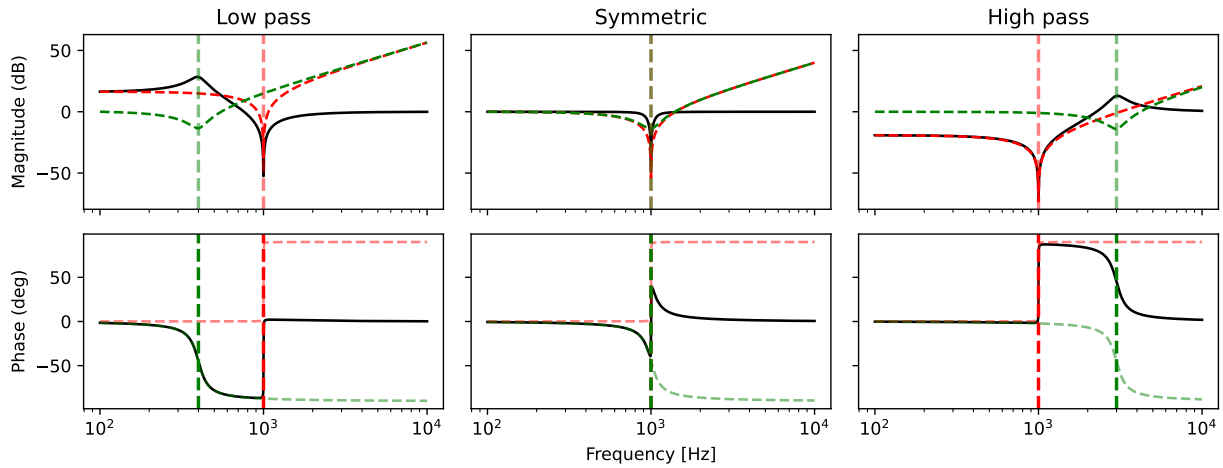


Figure 4.4: Magnitude and phase response of the low-pass, symmetric, and high-pass notch filters. The magnitude and phase are shown by the black, solid lines. The magnitude and response of the numerator and denominator are shown in red and green dashed lines, respectively. The pole frequency is indicated by a green, dashed, vertical line. The zero frequency (fixed at 1 kHz) is indicated by a red, dashed, vertical line.

denominator begins to decrease sooner than it does in the denominator, resulting in a gain before the notch and a unity gain afterward. This condition is referred to as a low-pass notch. In the other case, when the pole frequency is higher, i.e., $\omega_p > \omega_z$, the response before the notch is greater in the denominator, leading to greater passband attenuation before the notch, returning to unity gain after the pole. This is the high-pass notch condition. All three scenarios are shown in figure 4.4.

The other important consideration is the ratio of Q_z to Q_p. High quality factors are desired for both, as this provides a sharper transition and narrow bandwidth, closer to the ideal single frequency response. However, Q_z must be greater than Q_p by at least an order of magnitude. In the case where they are equal, the numerator and denominator effectively decrease at the same rate, maintaining a unity response, thereby producing no notch at all. When Q_p is larger than Q_z, the denominator decreases faster, leading to a *gain* instead of attenuation, i.e., an inverted notch, which is not desirable. Ideally, Q_z is reasonably large (e.g., $Q_z \geq 10^3$), allowing for a deep notch, and Q_p is at least 10, providing a narrow bandwidth. At most, Q_p is one order of magnitude lower than Q_z, otherwise the notch depth will begin to reduce.

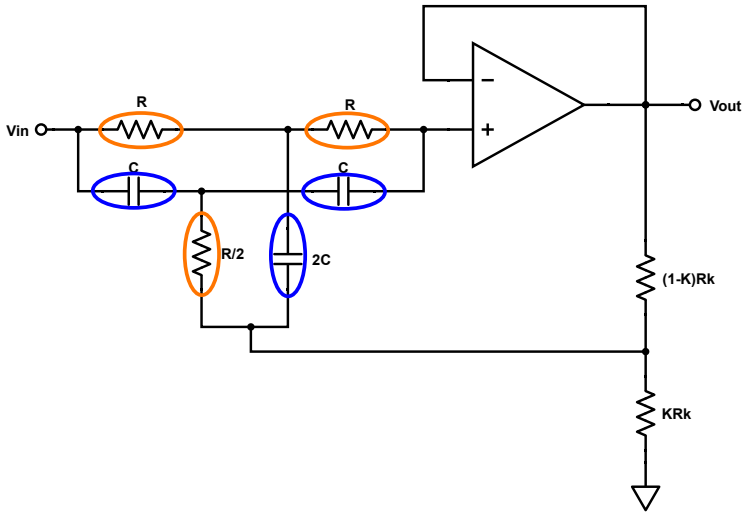


Figure 4.5: Active twin-T notch filter topology. The components chosen to be strain-responsive are circled in blue, and fixed elements used to design the notch frequency are circled in orange.

4.4.1 Twin-T notch filter

The active twin-T notch filter is a very simple circuit to analyze and implement. It is so named due to the parallel tees implementing low- and high-pass filters, summed at the output. Intuitively, setting equal cutoff frequencies yields a notch. An active variant buffers the output with an opamp, shown schematically in figure 4.5.

The transfer function for the twin-T notch is a cumbersome third-order function. However, in the special case presented in the schematic, with the resistors and capacitors matched at the given ratios, there is a pole-zero cancellation which makes the transfer function second-order and more convenient to analyze. For this variant, the transfer function is

$$H(s) = \frac{s^2 + \left(\frac{1}{RC}\right)^2}{s^2 + \left(\frac{4(1-K)}{RC}\right) s + \left(\frac{1}{RC}\right)^2} \tag{4.3}$$

Comparing this with the standard form, it is clear that the pole and zero frequencies are the same, forming a notch at

$$f_{notch} = \frac{1}{2\pi RC} \tag{4.4}$$

and the Q-factor of the pole is

$$Q_p = \frac{1}{4(1-K)}. \tag{4.5}$$

The first-order term in the numerator completely disappeared, equivalent to a high Q-factor in the numerator $Q_z \gg \omega_z$, allowing a deep notch. Choosing a value of K close to (but less than) one yields an arbitrarily large Q-factor. Note that if $(1 - K)R_k$ is not much less than R, another amplifier will be needed to buffer this feedback.

Frequency response to strain and effective gauge factor

To make the notch frequency strain-responsive, either the resistors, the capacitors, or both need to be made strain-responsive. Choosing component values of R and C in the ratios indicated by the schematic causes a pole-zero cancellation that drops the order of the transfer function down to second-order, so it is convenient to analyze. It does not matter which element is made strain-responsive. Choosing a resistive or capacitive strain gauge is a design decision that will be chosen for the particular application, as both options have their advantages and disadvantages. Wu et al. provide an excellent review of resistive and capacitive sensing approaches and their relative strengths and weaknesses in [126], and Nesser and Lubineau give a more detailed evaluation of capacitive strain sensing in particular [76]. For the data presented in this document, all implementations use simple parallel plate capacitive sensors, but a more complete analysis is provided here.

In addition, to help tune the design for a specific application, it is worth noting that the passive elements do not need to be completely replaced completely. Instead, they can be added in series or in parallel to fixed elements to better tune the response. Thus, there are three cases to consider: complete replacement of the passive element with a strain-responsive variant; an *additive* fixed element; or a *reciprocal* fixed element. The *additive* and *reciprocal* descriptions are needed to analyze the behavior of the circuit under equivalent modifications to different passive elements. In concrete terms, adding a resistor R_{new} in series to another resistor R_{old} is considered an *additive* modification, because the new resistance is the sum, i.e., $R = R_{old} + R_{new}$. However, if that resistor is added in parallel, the contribution becomes *reciprocal* in nature, i.e.,

$$\frac{1}{R} = \frac{1}{R_{old}} + \frac{1}{R_{new}}, \tag{4.6}$$

hence the name. These exact modifications are realized when capacitive elements are changed as well, but the initiating topologies are reversed. Adding capacitances in parallel yields an *additive* effect on total capacitance, while adding them in series yields a *reciprocal* change. From the perspective of the analysis, this *effect* is a much more useful concept than the topology that produced it, and greatly simplifies the interpretation.

To understand the response of the strain sensor, it is helpful to determine what the

effective gauge factor is, that is, what the frequency response to strain is. Suppose that a capacitive element will be completely replaced with a strain-responsive variant with the gauge factor G_c defined as

$$G_c = \frac{\Delta C/C_0}{\varepsilon}. \tag{4.7}$$

with strain $\varepsilon = \Delta\ell/\ell_0$ and C_0 is the capacitance at zero strain (i.e., $\varepsilon = 0$). For a parallel plate capacitor the gauge factor is ideally unity. Rearranging equation 4.7, noting that $\Delta C = C + C_0$, we can derive the capacitance at a particular strain as

$$C = C_0(1 + G_c\varepsilon). \tag{4.8}$$

Recalling the notch frequency from equation 4.4, we can calculate the fractional change in the notch frequency given a particular strain as

$$\begin{aligned}
\frac{f - f_0}{f_0} &= \frac{\frac{1}{2\pi R}\left(\frac{1}{C} - \frac{1}{C_0}\right)}{\frac{1}{2\pi R}\frac{1}{C_0}} \\
&= \frac{C_0 - C}{CC_0}C_0 \\
&= \frac{C_0 - C}{C} \\
&= \frac{C_0 - C_0(1 + G_c\varepsilon)}{C_0(1 + G_c\varepsilon)} \\
&= \frac{-G_c\varepsilon}{1 + G_c\varepsilon}
\end{aligned} \tag{4.9}$$

Therefore, the effective gauge factor for the case of directly replacing a filter element with a strain-responsive variant is

$$G_{f,\text{direct}} = \frac{\Delta f/f_0}{\varepsilon} = \frac{-G_c}{1 + G_c\varepsilon}. \tag{4.10}$$

The sensor gauge factor is completely determined by that of the strain gauge used and the strain itself. This nonlinearity is important to note when inferring the strain as no configuration will yield a linear strain response and, except when considering very small strains, linearization will produce too much error. Note also that this effective gauge factor is exactly the same if instead the resistors are made to be strain responsive, yielding instead

$$G_{f,\text{direct}} = \frac{-G_r}{1 + G_r\varepsilon}. \tag{4.11}$$

If both are strain-responsive, the gauge factors simply sum up, yielding

$$G_{f,\text{direct}} = \frac{-G_{rc}}{1 + G_{rc}\varepsilon} \tag{4.12}$$

where $G_{rc} = G_r + G_c$. Using elements with positive gauge factors implies a negative frequency response to strain. This is the case for a parallel plate capacitor which has a gauge factor around $G_c = 1$. Using a comb capacitor instead with a gauge factor of around $G_c = -2$ would lead to an increase in frequency in response to strain. It also introduces a pole in the gauge factor at $G_c\varepsilon = 1$ as the capacitance drops to zero and causes the notch frequency to approach infinity. The choice of a negative or positive gauge factor adds some flexibility, depending on whether it is desirable to have the response grow more slowly (positive) or more quickly (negative) with increasing strain.

The situation is somewhat more complicated when considering the use of fixed elements in conjunction with the strain-responsive component. Take the case of a capacitor used with a fixed element in an *additive* configuration (i.e., connected in parallel so the capacitances sum). In this case, the total capacitance given a strain is

$$C = C_{fixed} + C_0(1 + G_c\varepsilon). \tag{4.13}$$

To normalize the analysis, it is helpful to introduce a constant k which indicates the ratio of the size of the fixed component with respect to the zero-strain value, i.e.,

$$k = \frac{C_{fixed}}{C_0} \tag{4.14}$$

and therefore

$$\begin{aligned} C &= C_{fixed} + C_0(1 + G_c\varepsilon) \\ &= kC_0 + C_0(1 + G_c\varepsilon) \\ &= C_0(k + 1 + G_c\varepsilon). \end{aligned} \tag{4.15}$$

Performing the same calculations used to derive equations 4.9 and 4.12 we arrive at the following gauge factor:

$$G_{f,\text{additive}} = \frac{-G_c}{k + 1 + G_c\varepsilon}. \tag{4.16}$$

In this way, it is possible to use a fixed component to reduce the sensitivity of the response, which may be helpful if large strains are expected or if the elements are close to a singularity.

As an example, using a small form factor parallel plate capacitor might yield a very small capacitance, on the order of a few picofarads. For reference, the small and large capacitors fabricated for the twin-T implementation described later have capacitances of about 6 pF and 12 pF, respectively. Due to variability of the fabrication, it might be difficult to precisely control the notch frequency as relatively small deviations of even one picofarad accounts for a 14 % error in the specification, translating to about a 12 % change in the notch frequency. This additional factor can also be useful in conjunction with strain sensors with negative gauge factors to remove the pole in the response. It is worth pointing out now that these models have applicability only in a physically sensible domain. In the case of the negative gauge factor, there is no point where the total capacitance can drop below zero, so in reality this model is only valid up to a strain $\varepsilon = C_0/G_c$ when the capacitance goes to zero, which lands at $G_c\varepsilon = -1$. The true gauge factor is instead represented piecewise as

$$G_{f,\text{additive}} = \begin{cases} \dfrac{-G_c}{k+1+G_c\varepsilon} & 0 \leq \varepsilon < \dfrac{C_0}{G_c} \\ 0 & \text{otherwise} \end{cases}. \tag{4.17}$$

Depending on the device, compression (i.e., $\varepsilon < 0$) may be permissible. The previous analysis still holds, but depending on the design and geometry, the sign and magnitude of the gauge factor may change. The notation used implied that the gauge response is linear with strain, but the results largely still hold with minor modifications. As before, these results are identical for resistive strain-gauges when adding fixed resistance in an *additive* (i.e., series) configuration.

The same analysis can be performed once again for the reciprocal notation. The form is only a bit more complicated:

$$G_{f,\text{reciprocal}} = \frac{-kG_c}{(k+1)(1+G_c\varepsilon)}. \tag{4.18}$$

To get a better sense of how these equations translate to device performance, it is helpful to look at the gauge factor plotted as a function of strain for a few gauge factors, as in figure 4.6 for elements with positive gauge factors. Each tile shows the relationship between the effective gauge factor of the notch frequency response to a given strain. The columns indicate increasing gauge factors (1, 5, and 10) and the rows indicate the particular topologies for the elements R and C: fixed and direct, fixed and additive, fixed and reciprocal. That is, one set (resistors or capacitors) is implemented with discrete components and the other with strain-responsive components, either in the direct, additive, or reciprocal configuration as described previously. Plots are given for multiple values of k, the fraction of the zero-strain impedance added in an additive or reciprocal manner. k is irrelevant in the direct topology.

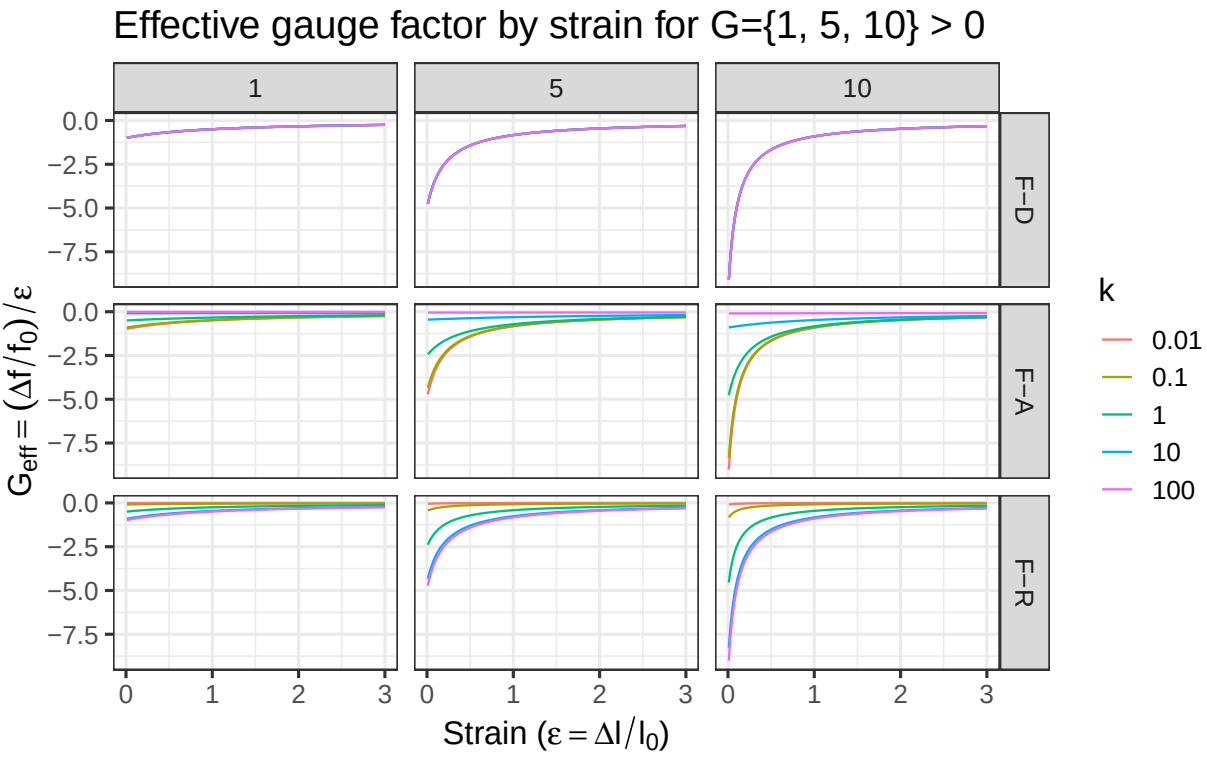


Figure 4.6: Effective gauge factor by strain. Gauge factors of 1, 5, and 10 are plotted in each respective column. Direct, additive, and reciprocal topologies are plotted in their respective rows. Lines are colored by k, the ratio of the fixed element to the zero-strain value of the responsive element.

In these configurations, it is helpful to note that, in all cases, the direct configuration sets an upper bound on the responsiveness of the sensor, starting at a value equal to the gauge factor of the underlying resistive or capacitive strain sensor and then decaying to zero with increased strain. The initial value and the decay rate can be tuned by modifying k. For the largest response, the direct connection should be used, or k should be high (in reciprocal configurations) or low (in additive configurations). Importantly, as has been stressed before, the plot gives no indication as to which of the elements, resistors or capacitors, have been replaced with strain gauges. It does not matter analytically, but of course there are trade-offs between the two technologies that are beyond the scope of this discussion.

When the gauge factors are negative, as in the case of the comb capacitor, the behavior looks quite different. Figure 4.7 shows a breakdown. Once again, the direct configuration gives an upper bound on the response rate. In this case, the frequency shift increases slowly to start, but increases rapidly near the singularity. k can be used once more to tune the curve, this time moving the asymptote somewhere convenient. Unlike in the previous case, the effective gauge factor can be substantially higher than that of the strain sensing element.

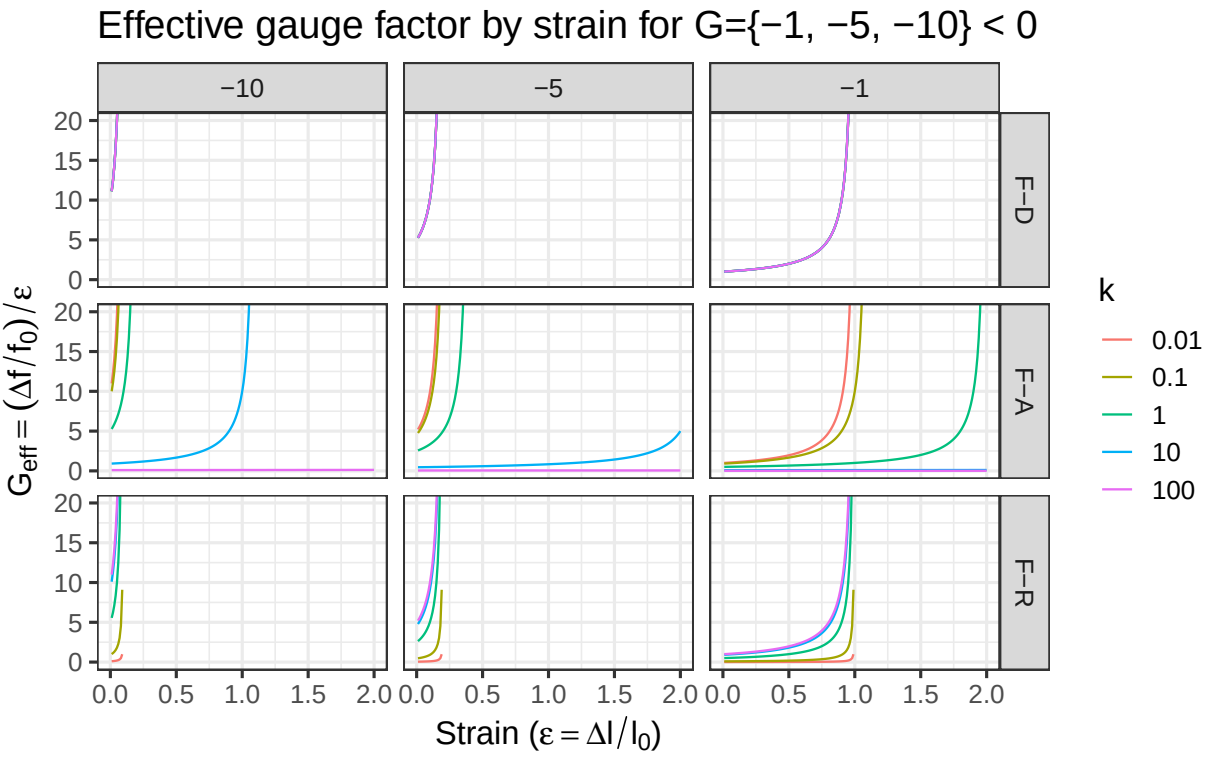


Figure 4.7: Effective gauge factor by strain. Gauge factors of -10, -5, and -1 are plotted in each respective column. Direct, additive, and reciprocal topologies are plotted in their respective rows. Lines are colored by k, the ratio of the fixed element to the zero-strain value of the responsive element.

Not included above are the cases where both the resistive and capacitive elements are made to be strain-responsive. This is perhaps not terribly interesting, but for completeness we will evaluate them anyway just to be sure. Following the same approach as before, we now need to incorporate two strain-sensitive elements. In the direct R, direct C case, we have

$$C = C_0 + G_c\varepsilon \tag{4.19}$$

$$R = R_0 + G_r\varepsilon. \tag{4.20}$$

The frequency response is now found to be

Table 4.1: Twin-T notch frequency shift in response to strain (part 1).

	Direct X	Additive X
Fixed Y	$\frac{(1)-(G_X\varepsilon+1)}{(G_X\varepsilon+1)}$	$\frac{(k_X+1)-(G_X\varepsilon+k_X+1)}{(G_X\varepsilon+k_X+1)}$
Direct Y	$\frac{(1)-(G_Y\varepsilon+1)(G_X\varepsilon+1)}{(G_Y\varepsilon+1)(G_X\varepsilon+1)}$	$\frac{(k_X+1)-(G_Y\varepsilon+1)(G_X\varepsilon+k_X+1)}{(G_Y\varepsilon+1)(G_X\varepsilon+k_X+1)}$
Additive Y		$\frac{(k_Y+1)(k_X+1)-(G_Y\varepsilon+k_Y+1)(G_X\varepsilon+k_X+1)}{(G_Y\varepsilon+k_Y+1)(G_X\varepsilon+k_X+1)}$

$$\begin{aligned}\frac{f-f_0}{f_0} &= \frac{\frac{1}{2\pi}\left(\frac{1}{RC}-\frac{1}{R_0C_0}\right)}{\frac{1}{2\pi}\frac{1}{R_0C_0}} \\ &= \frac{R_0C_0-RC}{RCR_0C_0}R_0C_0 \\ &= \frac{R_0C_0-RC}{RC} \\ &= \frac{R_0C_0-R_0(1+G_r\varepsilon)C_0(1+G_c\varepsilon)}{R_0(1+G_r\varepsilon)C_0(1+G_c\varepsilon)} \\ &= \frac{1-(1+G_r\varepsilon)(1+G_c\varepsilon)}{(1+G_r\varepsilon)(1+G_c\varepsilon)}. \end{aligned} \tag{4.21}$$

The resulting gauge factor is then

$$G_{f,dd} = \frac{1-(1+G_r\varepsilon)(1+G_c\varepsilon)}{\varepsilon(1+G_r\varepsilon)(1+G_c\varepsilon)} = \frac{-(G_rG_c\varepsilon+G_r+G_c)}{(1+G_r\varepsilon)(1+G_c\varepsilon)}. \tag{4.22}$$

The same process can be repeated for all other combinations of topologies. Those results are tabulated in tables 4.1 and 4.2. We have established that replacing resistive or capacitive elements in the circuit is equivalent when evaluated analytically from the perspective of direct, additive, or reciprocal topologies, so the tables are simplified by presenting the impedances more generally as X and Y to avoid duplication. The tables are left in an unreduced form to make the underlying structure clearer, but most of the equations simplify quite a bit. In all cases, the numerator will drop all terms that do not include strain ε, which will not produce a frequency shift at zero strain, as expected. More generally, the two terms in the numerator will cancel out at zero strain. As we shall see, this is true of the Bainter notch as well, even though the terms are more difficult to reduce in the same way.

Figure 4.8 shows the effective gauge factors when both impedance sets are strain-responsive. In the positive (4.8a) and negative (4.8b) cases, the effect is simply to amplify the same

Table 4.2: Twin-T notch frequency shift in response to strain (part 2).

	Reciprocal X
Fixed Y	$\frac{(G_X\varepsilon + k_X + 1) - (k_X + 1)(G_X\varepsilon + 1)}{(k_X + 1)(G_X\varepsilon + 1)}$
Direct Y	$\frac{(G_X\varepsilon + k_X + 1) - (k_X + 1)(G_Y\varepsilon + 1)(G_X\varepsilon + 1)}{(k_X + 1)(G_Y\varepsilon + 1)(G_X\varepsilon + 1)}$
Additive Y	$\frac{(k_Y + 1)(G_X\varepsilon + k_X + 1) - (k_X + 1)(G_X\varepsilon + 1)(G_Y\varepsilon + k_Y + 1)}{(k_X + 1)(G_X\varepsilon + 1)(G_Y\varepsilon + k_Y + 1)}$
Reciprocal Y	$\frac{(G_Y\varepsilon + k_Y + 1)(G_X\varepsilon + k_X + 1) - (k_Y + 1)(k_X + 1)(G_Y\varepsilon + 1)(G_X\varepsilon + 1)}{(k_Y + 1)(k_X + 1)(G_Y\varepsilon + 1)(G_X\varepsilon + 1)}$

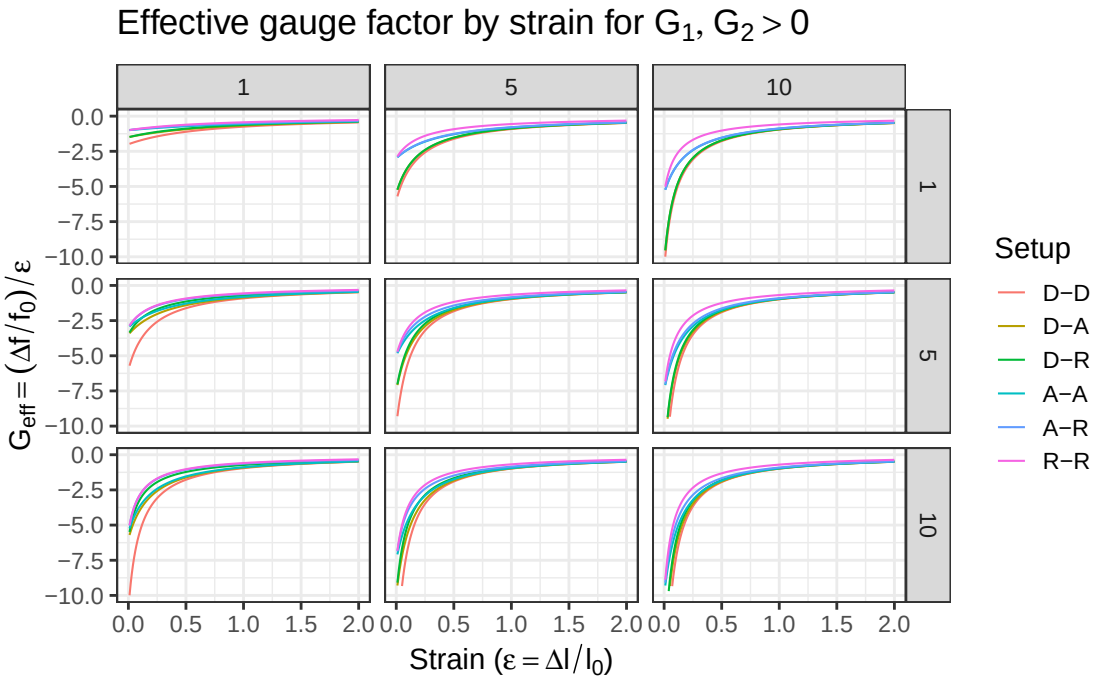
behavior.

Figure 4.9 shows what happens when the impedance elements have gauge factors with opposite signs. The frequency response is now a superposition of the two behaviors, which isn't terribly surprising. It is not clear in what, if any, cases the added complexity of modeling and implementation may be worthwhile to achieve this behavior. Notably, in some configurations the gauge factor has both negative and positive regimes, suggesting the sensor response will be ambiguous for some values. This behavior might be useful if two distinct strain limits are desired, one which can be detected at the designed negative frequency shift and another at a designed positive frequency shift.

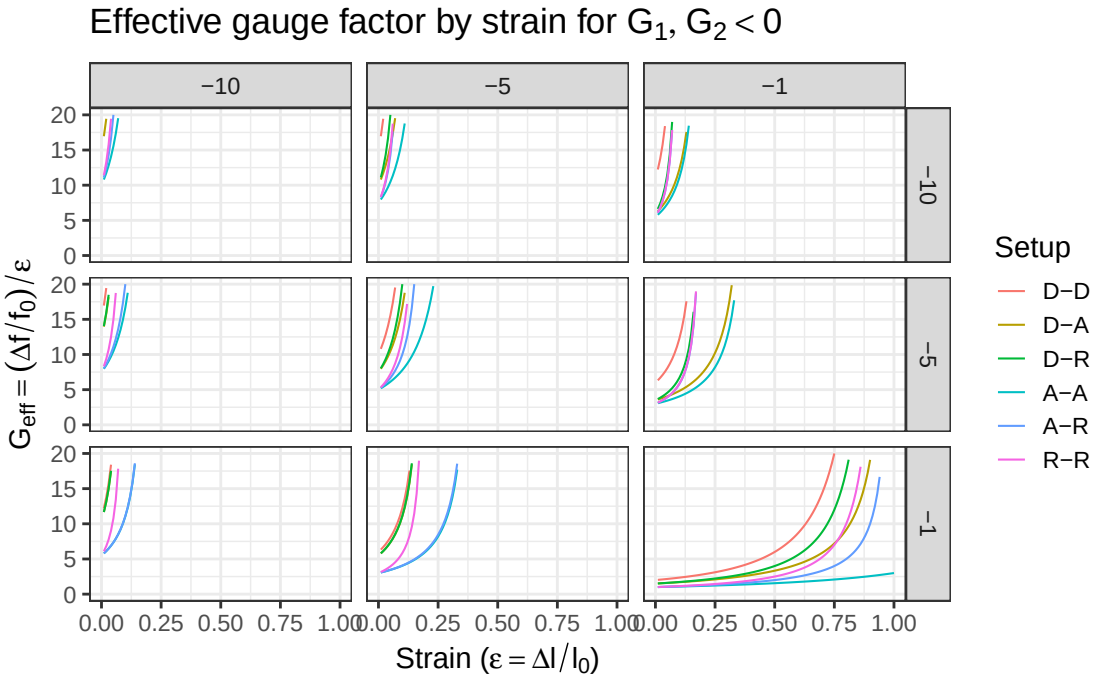
Maintaining component ratios

An important consideration for the twin-T network is that component ratios need to be maintained. This means that if strain-sensitive capacitors are used, for example, all capacitors must be strain-responsive, not just one. In addition, they need to have a proportional response under strain to maintain the proper ratio. Failing to maintain that ratio prevents proper operation as a notch filter. There is some tolerance, but straying too far from these ratios means that the attenuation will not be as good, the frequency shift will not be as pronounced, and the bandwidth will be larger. Figure 4.10 demonstrates some of the ill effects. Increasing or decreasing by a small amount, 10 % in this case, the correct response is maintained, but there is a slight frequency shift and the notch depth is degraded substantially. For a larger (50 %) mismatch, the notch practically disappears, or the filter might oscillate instead, causing a positive gain. The third-order transfer function becomes more complicated in this scenario as the pole-zero cancellation does not happen, so it becomes difficult to make a general statement about what might happen when the components are not sufficiently matched.

In addition to designing in the appropriate ratio, it is also important to be sure that the



(a) Effective gauge factor strain-responsive elements have positive gauge factors.



(b) Effective gauge factor strain-responsive elements have negative gauge factors.

Figure 4.8: Effective gauge factors for topologies using two strain-responsive impedance sets in either **D**irect, **A**dditive, or **R**eciprocal topologies. Plots are included for multiple gauge factors (1, 5, and 10) and for devices with positive (a) and negative (b) gauge factors. The ratio k between the fixed elements (i.e., non-strain-responsive) and the zero-strain values of the strain-responsive elements is fixed to $k = 1$.

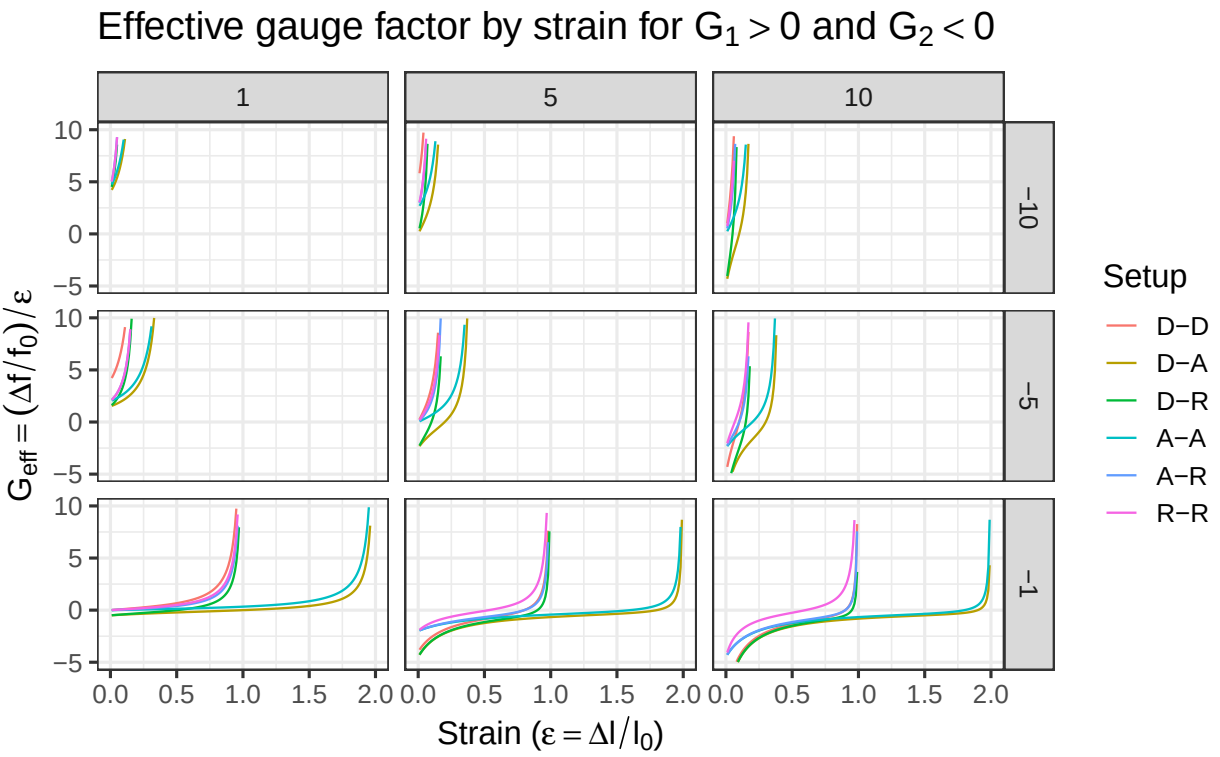


Figure 4.9: Effective gauge factors for topologies using two strain-responsive impedance sets in either **D**irect, **A**dditive, or **R**eciprocal topologies. Plots are included for multiple gauge factors (1, 5, and 10) for configurations mixing strain-sensitive elements with positive and negative gauge factors. The ratio k between the fixed elements (i.e., non-strain-responsive) and the zero-strain values of the strain-responsive elements is fixed to $k = 1$.

ratios are maintained in any given configuration. This is trivially seen in the case of direct placement. Suppose that one large and two small parallel plate capacitors are fabricated. The small capacitors have half the area of the large capacitor to realize the $2:1$ ratio. That is, $C_{small} = C_0$ and $C_{big} = 2C_0$. If the direct topology is used, we have

$$\begin{aligned} C_{small} &= (1 + G_c\varepsilon)C_0 \\ C_{big} &= (1 + G_c\varepsilon)(2C_0) \\ \frac{C_{big}}{C_{small}} &= \frac{(1 + G_c\varepsilon)(2C_0)}{(1 + G_c\varepsilon)C_0} \\ &= 2. \end{aligned}$$

This requires that the gauge factors and experienced strains are identical. Some more complicated geometric arrangement might be possible to achieve this without those constraints,

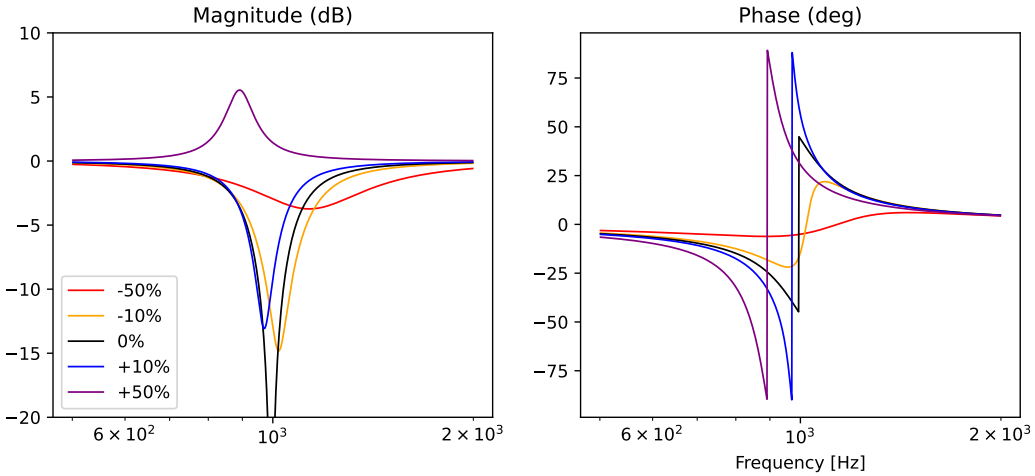


Figure 4.10: Active twin-T notch filter with a mismatched $2C$ capacitor. Plots are shown for the magnitude and phase response of the filter, designed to reject 1 kHz, when $2C$ is increased or decreased by some tolerance.

but sticking to those constraints is the most straightforward approach to implementing a sensor.

The additive case is similar. Now we have

$$\begin{aligned} C_{small} &= (1 + G_c\varepsilon)(1 + k)C_0 \\ C_{big} &= (1 + G_c\varepsilon)(1 + k)(2C_0) \\ \frac{C_{big}}{C_{small}} &= \frac{(1 + G_c\varepsilon)(1 + k)(2C_0)}{(1 + G_c\varepsilon)(1 + k)C_0} \\ &= 2. \end{aligned}$$

This now assumes that the gauge factor is the same between capacitors, the strain is experienced equally between components, and the fixed values are chosen in the same ratio, given by k.

The reciprocal case is somewhat more complicated, but yields the same result. Suppose C_0 is the zero-strain capacitance of the strain-sensitive element. If arranged in a reciprocal configuration with a fixed element of size $C_{fixed} = kC_0$, we have the total capacitance under a given strain as

$$C_{\text{reciprocal}} = \frac{(C_0(1 + G_c\varepsilon))\,(kC_0)}{C_0(1 + G_c\varepsilon) + kC_0} = \frac{k(1 + G_c\varepsilon)}{1 + k + G_c\varepsilon}C_0. \tag{4.23}$$

This is linearly proportional to C_0, so once again the ratio will remain correct under strain, as long as the same constraints as before are maintained: capacitors must have the same gauge factor, the same strain, and fixed elements must be selected using the same ratio k.

Considerations

Before opting for the twin-T notch topology, it is worth considering some drawbacks. The first is likely already obvious from the preceding sections: to operate the notch filter, it is necessary to fabricate three components at or near the appropriate ratio. This may greatly complicate the design and fabrication of a device, and makes it more difficult to fit elements densely into a small space.

Another drawback is the reasonably precise tuning requirement. It is not always possible to locate the appropriate resistor and capacitor values that meet the tolerance and ratio requirements. For this reason, it can be a challenge to precisely tune a twin-T notch even without considering the stretchable elements.

The final drawback that will be addressed is not so obvious. If large resistor values are used in the tees, at the megaohm scale and larger, practical opamps, even with high gain, cannot completely avoid broadband attenuation of the signal. See figure 4.11 for a comparison of the same active filter circuit, one using an ideal single-pole high-gain opamp model and the other using the model supplied by the vendor (NCS20071, onsemi). Although not nearly as bad as the attenuation seen in the passive filter network (figure 4.3), the attenuation greatly limits the number of sensors that can be distributed in a single network. Reducing resistances to below 1 MΩ resolves this problem. One reason why this arises in the first place is that the fabricated capacitors might have very small capacitances. In the implementation presented here, the capacitors in the network were 6 and 12 pF. With $C = 6\,\text{pF}$, a resistance of about $R = 26.5\,\text{M}\Omega$ is required to generate a 1 kHz notch. Very small capacitances necessitate either high resistances or high notch frequencies. That has trade-offs of its own which are explored later.

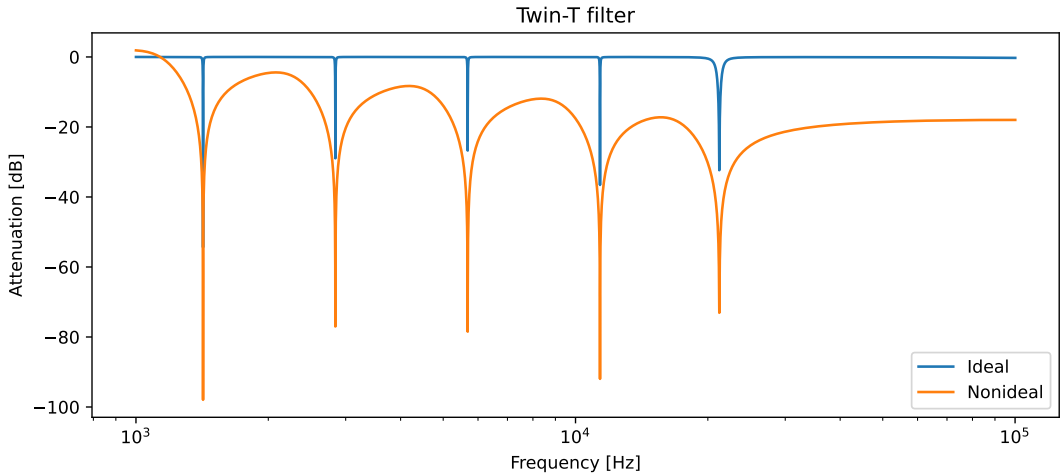


Figure 4.11: Transfer functions of five spectrally isolated notch filters using an active twin-T notch topology. Performance using ideal and non-ideal operational amplifier models are compared.

4.4.2 Bainter notch filter

In section 4.4.1 the active twin-T notch filter was described. Although effective, it presented a few challenges to the design, namely the difficulties in tuning; the requirement of multiple, tuned sensing elements; and the challenges with broadband attenuation in some circumstances. In 1975 Bainter published a notch filter design that can address many of the problems encountered with the twin T [8]. This is generally referred to as the *Bainter* notch filter, represented schematically in figure 4.12.

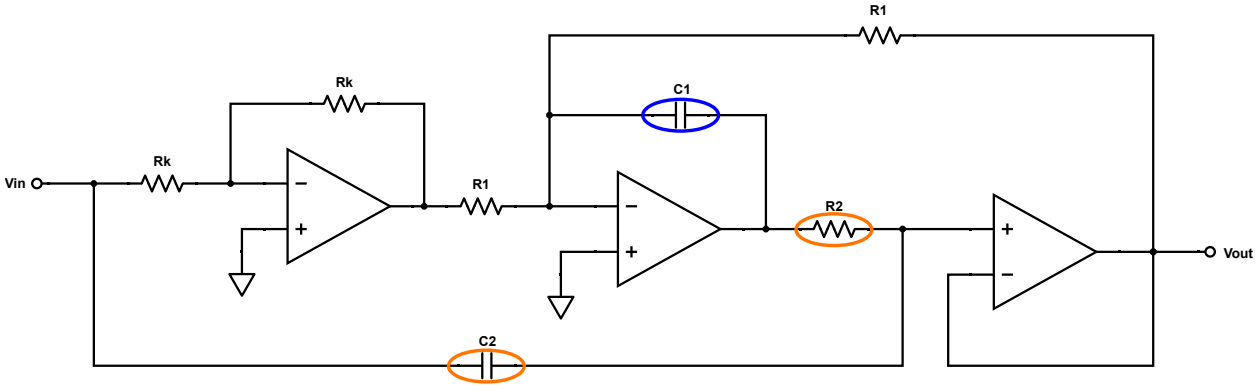


Figure 4.12: Bainter notch filter topology. The components chosen to be strain-responsive are circled in blue, and fixed elements used to design the notch frequency are circled in orange.

Generally speaking, the Bainter notch can be tuned to provide a high-pass, low-pass, or symmetric notch response (shown previously in figure 4.4), with tunable frequency and

Q-factor. Importantly, while other notch topologies (including the twin T) have a quality factor Q_z in the numerator that is largely dependent on the passive components that make up the filter, the Bainter notch does not. So long as an opamp with sufficiently high gain is selected (e.g., $A \geq 10^4$), the first-order term drops out of the numerator, as the quality factor of the zero is

$$Q_z = \sqrt{\frac{R_1 C_1 A(A+1)}{4R_2 C_2}} \approx \sqrt{\frac{R_1 C_1 A^2}{4R_2 C_2}} = A\sqrt{\frac{R_1 C_1}{4R_2 C_2}}. \tag{4.24}$$

Given the proportionality with the amplifier gain, it is easy to choose component values to achieve large Q-factors of 100 or higher, at which point the first-order term is much smaller than the second-order terms and can be largely ignored in the analysis. In other words, it is easy to design a notch filter whose notch depth is largely unaffected by precise component values. With the component values indicated in the schematic, a symmetric notch has been defined with the Q-factor of the pole determined by

$$Q_p = \sqrt{\frac{R_2 C_2}{R_1 C_1}} \tag{4.25}$$

and the notch frequency at

$$f_{\text{notch}} = \frac{1}{2\pi}\frac{Q_p}{R_2 C_2} = \frac{1}{2\pi}\frac{1}{Q_p R_1 C_1} = \frac{1}{2\pi}\sqrt{\frac{1}{R_1 R_2 C_1 C_2}}. \tag{4.26}$$

As indicated previously, Q_p should be designed to be at least 10 but not more than an order of magnitude below Q_z.

Frequency response to strain and effective gauge factor

Determining the frequency response and gauge factor is done in exactly the same manner as in section 4.4.1, so only the abridged version will be presented for the Bainter notch. The results are similar, only modestly complicated by the square root relationship. Assuming C_1 is made strain-responsive and connected directly, the frequency shift for a given strain is

$$\frac{f - f_0}{f_0} = \frac{1 - \sqrt{1 + G_c\varepsilon}}{\sqrt{1 + G_c\varepsilon}}. \tag{4.27}$$

This is reminiscent of the first cell of table 4.1, but now the term with the strain and capacitive gauge factor is trapped under a square root. The gauge factor is then found to be

$$G_{\text{f,dd}} = \frac{1}{\varepsilon}\left(\frac{1 - \sqrt{1 + G_c\varepsilon}}{\sqrt{1 + G_c\varepsilon}}\right), \tag{4.28}$$

Table 4.3: Bainter notch frequency shift in response to strain (part 1).

	Direct X	Additive X
Fixed Y	$\frac{\sqrt{1}-\sqrt{G_X\varepsilon+1}}{\sqrt{G_X\varepsilon+1}}$	$\frac{\sqrt{k_X+1}-\sqrt{G_X\varepsilon+k_X+1}}{\sqrt{G_X\varepsilon+k_X+1}}$
Direct Y	$\frac{\sqrt{1}-\sqrt{G_Y\varepsilon+1}\sqrt{G_X\varepsilon+1}}{\sqrt{G_Y\varepsilon+1}\sqrt{G_X\varepsilon+1}}$	$\frac{\sqrt{k_X+1}-\sqrt{G_Y\varepsilon+1}\sqrt{G_X\varepsilon+k_X+1}}{\sqrt{G_Y\varepsilon+1}\sqrt{G_X\varepsilon+k_X+1}}$
Additive Y		$\frac{\sqrt{k_Y+1}\sqrt{k_X+1}-\sqrt{G_Y\varepsilon+k_Y+1}\sqrt{G_X\varepsilon+k_X+1}}{\sqrt{G_Y\varepsilon+k_Y+1}\sqrt{G_X\varepsilon+k_X+1}}$

Table 4.4: Bainter notch frequency shift in response to strain (part 2).

	Reciprocal X
Fixed Y	$\frac{\sqrt{G_X\varepsilon+k_X+1}-\sqrt{k_X+1}\sqrt{G_X\varepsilon+1}}{\sqrt{k_X+1}\sqrt{G_X\varepsilon+1}}$
Direct Y	$\frac{\sqrt{G_X\varepsilon+k_X+1}-\sqrt{k_X+1}\sqrt{G_Y\varepsilon+1}\sqrt{G_X\varepsilon+1}}{\sqrt{k_X+1}\sqrt{G_Y\varepsilon+1}\sqrt{G_X\varepsilon+1}}$
Additive Y	$\frac{\sqrt{k_Y+1}\sqrt{G_X\varepsilon+k_X+1}-\sqrt{k_X+1}\sqrt{G_X\varepsilon+1}\sqrt{G_Y\varepsilon+k_Y+1}}{\sqrt{k_X+1}\sqrt{G_X\varepsilon+1}\sqrt{G_Y\varepsilon+k_Y+1}}$
Reciprocal Y	$\frac{\sqrt{G_Y\varepsilon+k_Y+1}\sqrt{G_X\varepsilon+k_X+1}-\sqrt{k_Y+1}\sqrt{k_X+1}\sqrt{G_Y\varepsilon+1}\sqrt{G_X\varepsilon+1}}{\sqrt{k_Y+1}\sqrt{k_X+1}\sqrt{G_Y\varepsilon+1}\sqrt{G_X\varepsilon+1}}$

noting that the strain will no longer factor nicely out of the numerator. Running through the same process for other variants, we uncover some nice symmetry with the original table. Evaluating the filter with an *additive* fixed capacitance (i.e., a fixed capacitor in parallel to the strain-responsive variant) the gauge factor works out to be

$$G_{\mathrm{f,da}} = \frac{1}{\varepsilon}\left(\frac{\sqrt{k+1}-\sqrt{G_c\varepsilon+k+1}}{\sqrt{G_c\varepsilon+k+1}}\right), \tag{4.29}$$

and a *reciprocal* fixed capacitance yields the following shift:

$$G_{\mathrm{f,dr}} = \frac{1}{\varepsilon}\left(\frac{\sqrt{G_c\varepsilon+k+1}-\sqrt{(k+1)(G_c\varepsilon+1)}}{\sqrt{(k+1)(G_c\varepsilon+1)}}\right). \tag{4.30}$$

Conveniently, this carries over to all the gauge factors, so tables 4.1 and 4.2 can also be used for the Bainter notch simply by putting each term under a radical. The modified forms are tabulated in tables 4.3 and 4.4.

Considerations

The Bainter notch helps address a few key issues with the twin T notch:

- the notch depth is not dependent on component values and tuning, it is inherent so long as a high-gain (i.e., $A \geq 10^4$) amplifier is used;
- multiple, matched strain-sensitive components need not be fabricated, a single strain gauge is sufficient;
- small values (e.g., for a strain-sensitive capacitor) are not detrimental to function, there is no broadband attenuation as in the case of the twin T.

With that said, there are a few important points to keep in mind when designing with the Bainter notch. First, the active twin T filter design works well with an opamp biased from ground to V_{cc} because the tees are connected to the output of the opamp. This avoids the need for the amplifier to generate negative voltages to operate the circuit normally. So long as the input has a bit of headroom, perhaps 10 % for good measure, the output should not clip. The Bainter does not have this same feature, and the opamps need to be negative biased. Alternatively, the ground references on the noninverting inputs can be swapped with a voltage divider. This has the effect of halving the acceptable input range and thus the measurement sensitivity. So long as the signal noise floor is low, this should not be detrimental to performance. Alternatively, a five-wire variant can be used that sends both V^+ and V^- to bias the opamps.

Another consideration is to ensure that the values used actually produce a Q_z large enough to ignore the first-order term in the numerator and to avoid signal amplification combined with a Q_p that is too high. This is typically easy to accomplish since modern amplifiers typically have very high gains. The OPA4991 used later, for example, specifies a gain of 125 dB, or $A = 10^{6.25}$. But the gain is not enough, exceptionally poor design of component values can still reduce Q_z enough to noticeably degrade the notch performance.

One final, less obvious design recommendation is to ensure that the cascaded filters have enough damping to avoid resonance. Suppose $C_1 = 100\,\text{pF}$, $C_2 = 100\,\text{nF}$, $R_1 = 1\,\text{k}\Omega$, and $R_2 = 100\,\text{k}\Omega$. This yields $Q_z > 2800$ with $A = 125\,\text{dB}$, $Q_p \approx 316$, and $f_{notch} \approx 5\,\text{kHz}$. Although at first glance these numbers seem fine, plotting the frequency response yields unexpected results, shown in figure 4.13. Shown in blue are the magnitude and phase response of the filter. The red, vertical, dashed line indicates the designed notch frequency. However, instead of a notch, the filter produces a 7 dB spike about 1 kHz below the notch. The phase plot is otherwise not informative; the expected asymptote is visible. At the non-inverting input of the second opamp, connected to C_1, a low-pass response is expected. The magnitude and phase of that node is plotted in dashed green lines, which instead shows excessive gain

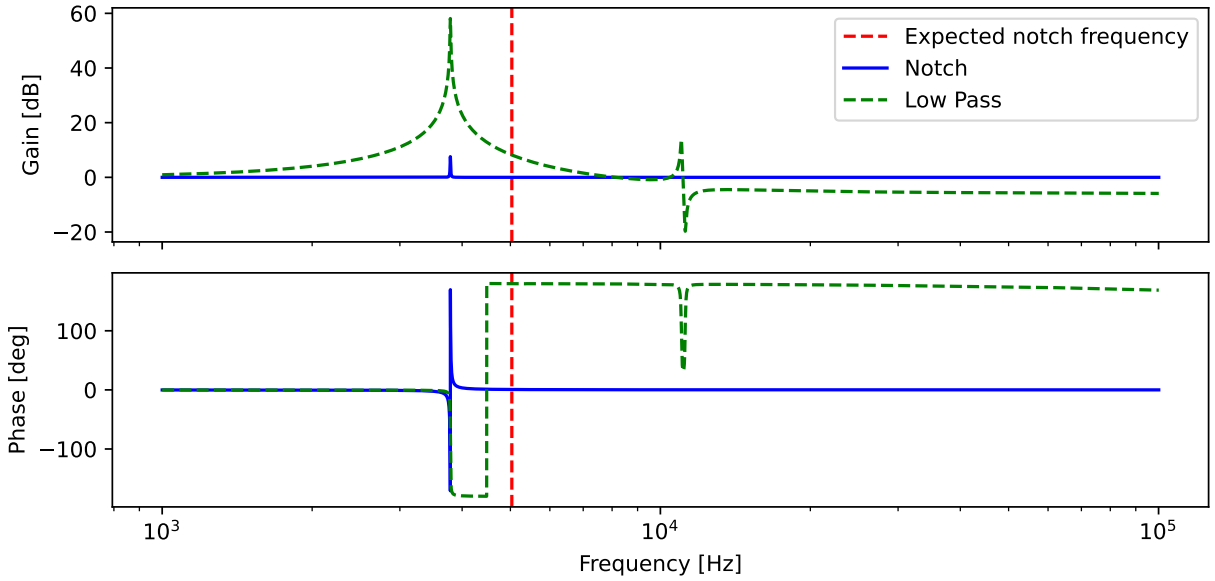


Figure 4.13: Poorly designed Bainter notch exhibiting resonance and producing amplification instead of attenuation below the notch frequency. Expected notch frequency is indicated by the vertical, dashed red line. Magnitude and phase response of the center opamp's non-inverting input (connected to C_1), expected to produce a low-pass response, is shown in dashed green.

and resonance.

The passive components can be modified somewhat so now $R_1 = 25\,\mathrm{k\Omega}$ and $C_2 = 4\,\mathrm{nF}$. This yields $Qz > 70000$, a much lower $Qp \approx 12$, while maintaining the same (expected) notch frequency. Figure 4.14 shows the new response that produces both the expected notch in the filter output and the low-pass behavior at C_1. In this case, the Q-factor of the pole was sacrificed a bit, so the notch is not as tight, but the response is stable and the notch produced is very clean.

Figure 4.15 shows a pole-zero plot normalized to the notch frequency. Here, the issue becomes apparent: the poles in the resonant circuit were much too close to the imaginary axis, about 0.1 % of the imaginary components. In the non-resonant case, the ratio is much higher, at about 4 %. General guidelines suggest maintaining real magnitudes of at least 1 % of the imaginary parts to avoid resonance issues and provide a margin to account for component tolerances or non-ideal opamp behavior. The Bainter notch contains few different feedback paths, so it is worth probing a few nodes in simulation to ensure the circuit has enough margin.

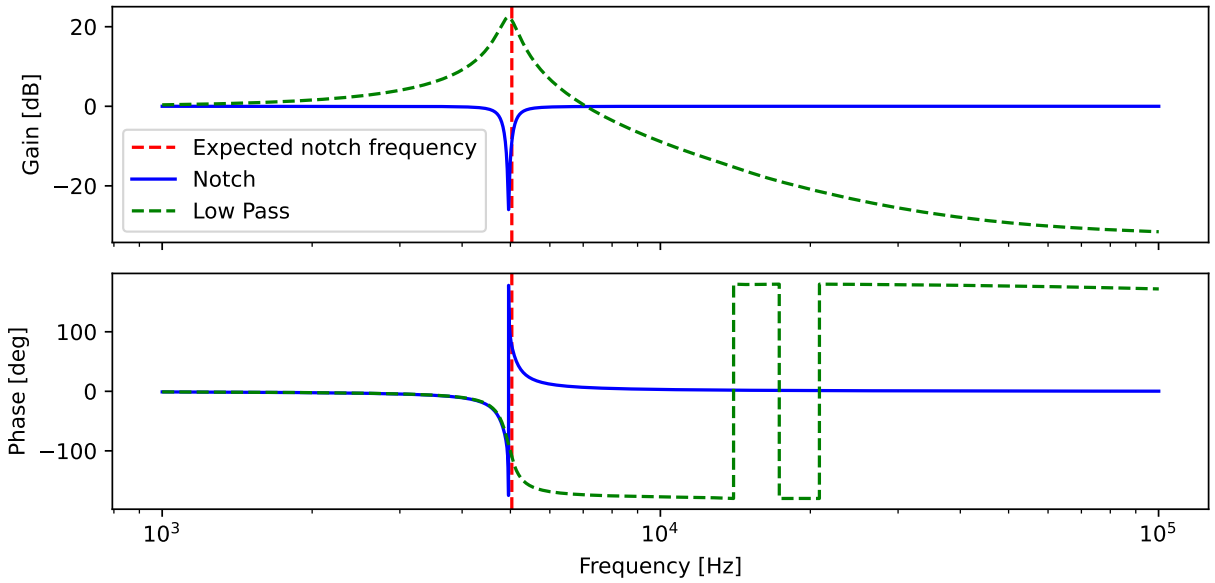


Figure 4.14: Improved design of Bainter notch removing resonance. Expected notch frequency is indicated by the vertical, dashed red line. Magnitude and phase response of the center opamp's non-inverting input (connected to C_1), expected to produce a low-pass response, is shown in dashed green.

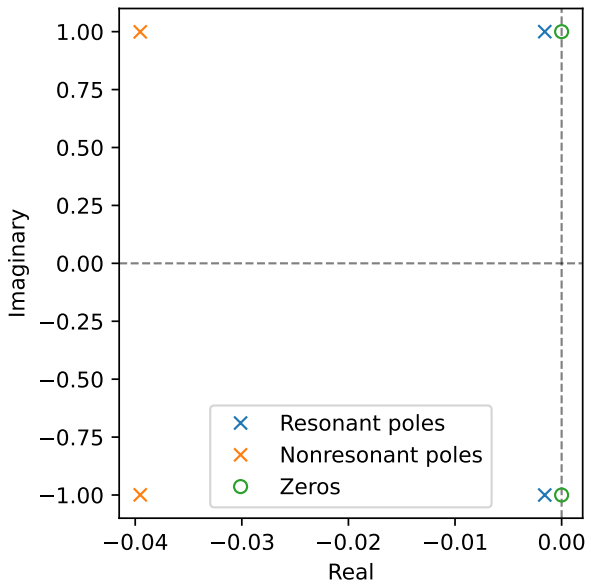


Figure 4.15: Pole-zero plot normalized to the notch frequency. Poles for the resonant (blue) and stable (orange) filters are shown.

4.5 Design guidelines for distributed sensing

There are a number of ways to implement a distributed sensor network using this framework. The exact mechanics of how to design a sensor network for a particular application will greatly vary the optimization parameters and priorities. Two procedures are highlighted here, and the procedure by which the design is accomplished should lay sufficient groundwork to enable optimization of a network for new applications. Example networks are also derived, which will give some intuition about the capabilities and limitations inherent in this type of system.

The first approach will address incorporating new sensors sequentially, starting with a known starting frequency and maximum strain and calculating the notch frequency and resolution. In the second approach, we start with the full frequency spectrum and sensor count and then determine the maximum strain and resolution. In both of these cases, a twin-T notch with the strain gauge directly connected is assumed to simplify the analysis and ensure that the presentation is clear, but the strategy holds for more complicated arrangements. As a final example, a densely packed three-sensor Bainter notch network is designed.

4.5.1 Preliminaries

In the sections that follow, it will be useful to have a few design terms specified. First, there is a particular mapping from a tensile strain to the resulting notch frequency. We define a model $f : \mathcal{S} \mapsto \mathcal{F}$ that maps the feasible range of strains $\mathcal{S} : [0, \varepsilon_1]$ onto the bandwidth $\mathcal{F} : [f_1, f_0]$, where ε_1 is the maximum strain, f_1 is the frequency at ε_1, and f_0 is the frequency at $\varepsilon = 0$. A negative relationship between strain and frequency is assumed (i.e., $f_1 < f_0$), consistent for models with positive gauge factors $G > 0$. For the next two sections we assume a strain sensor is added directly into the twin-T notch topology, implying the following model:

$$\begin{aligned} \frac{f(\varepsilon) - f_0}{f_0} &= \frac{-G\varepsilon}{1 + G\varepsilon} \\ f(\varepsilon) &= f_0 \left(1 - \frac{G\varepsilon}{1 + G\varepsilon}\right) \\ f(\varepsilon) &= f_0 \left(\frac{1 + G\varepsilon - G\varepsilon}{1 + G\varepsilon}\right) \\ f(\varepsilon) &= \frac{f_0}{1 + G\varepsilon} \ \forall \ \varepsilon \in \mathcal{S}. \end{aligned} \tag{4.31}$$

The primary motivation to explicitly state $\mathcal{S}$ is to provide a constraint for parameter identi-

fication. If ε can grow unbounded, the lower bound on every sensor element is $f_1 = 0\,\text{Hz}$, which makes the design of a network impossible and is physically unrealizable. Furthermore, ε_1 is not simply a *physical constraint* but rather a *design consideration*: the maximum strain used in designing a network should be the maximum strain *expected* given the application plus some tolerance. Otherwise, the bandwidth required for each sensor may be excessive and limit the number of sensors that can be positioned within a bandwidth. The available sensing bandwidth is not unbounded either, but is dependent on our ability to generate an interrogating signal at the target frequency, sample at a rate high enough to capture the waveforms with low noise, and sample at a frequency high enough for the application.

Two other important parameters will be evaluated that deal with the sensing resolution. The resolution is controlled by two parameters: Δf_r, the frequency resolution; and $\Delta\varepsilon_r$, the strain resolution. The frequency resolution is tuned by the PSD computation. It is assumed that the signal generator is capable of delivering spectral energy uniformly in the chirp bandwidth, which can be verified by computing the PSD of a chirp and confirming the uniform energy in the detection bandwidth. For practical frequencies (say, $1-100\,\text{kHz}$) this is easily achieved. The frequency resolution is then a tunable parameter of the PSD generation, which trades off resolution for noise. The strain resolution is more likely to be the designed parameter: what frequency resolution is required to give a particular strain resolution? In other cases, it might just be a fall-out of another design choice: given a bandwidth and sensor density, what is the best strain resolution? With that, we can start the network design.

4.5.2 Building a network sequentially

Suppose that we want to add sensors sequentially in a network. Since models are added one by one, each filter is designed independently, and there is no need to assume homogeneity. For a given sensor, assume a particular model for $f(\varepsilon)$, for example, the direct connection to a twin-T notch filter as in equation 4.31. Assume also a starting frequency f_1 and a maximum strain ε_1. Note that in this case we are designing from the frequency at maximum strain, i.e., $f_1 = f(\varepsilon_1)$. Figure 4.16 demonstrates why this is useful from the perspective of adding sensors sequentially: we always know the boundary condition of the previous sensor, i.e., the notch frequency; we know how close we want the max strain frequency f_1 to be, what we do not know is where the next zero-strain notch frequency is located.

We start by ignoring any margin between sensors (the α term in figure 4.16), so the zero-strain (notch) frequency of one sensor is the max-strain frequency of the next. For the direct connection model, there is nothing to tune except the gauge factor. We can determine the zero-strain notch frequency with a particular gauge factor by simply rearranging equation 4.31:

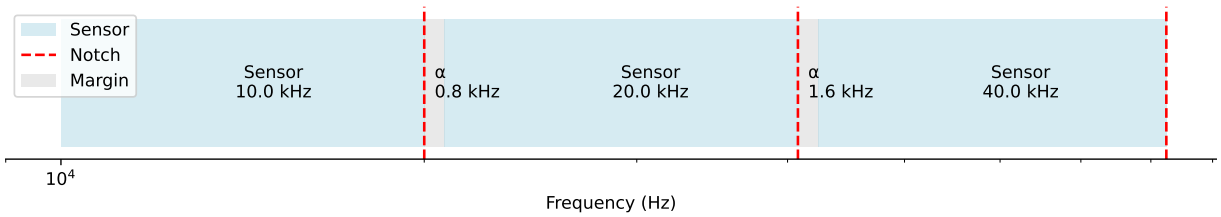


Figure 4.16: Spectral placement of sequential filters. Shaded blue regions represent a sensor's full-strain frequency bandwidth. The zero-strain notch is indicated by a red, vertical dashed line. The gray shaded regions represent the margin α between sensors.

$$f_1 = \frac{f_0}{1 + G\varepsilon_1},$$
$$f_0 = f_1(1 + G\varepsilon_1). \tag{4.32}$$

The sensor bandwidth is completely determined by the gauge factor of the strain sensor used. Although this was determined for the simplest case, it is worth noting that, as long as the gauge factor is constant given a value of ε, that is, it has no dependence on frequency or any other factor that changes during operation, then we can generalize the relationship in 4.32 and relate f_1 and f_0 to any function Γ of ε only:

$$\frac{f(\varepsilon) - f_0}{f_0} = \Gamma(\varepsilon)$$
$$f(\varepsilon) = f_0\left(1 - \Gamma(\varepsilon)\right). \tag{4.33}$$

Letting the constant $\Gamma_1 = \Gamma(\varepsilon_1)$, we have

$$f_1 = f_0(1 + \Gamma_1), \tag{4.34}$$
$$f_0 = f_1(1 + \Gamma_1)^{-1}. \tag{4.35}$$

With a known gauge factor, notch frequency, and frequency at maximum strain, we can derive a relationship between the strain and frequency resolutions. To achieve this frequency resolution we need to satisfy the constraint that

$$\begin{aligned}\Delta f_r &\leq f(\varepsilon_1 - \Delta\varepsilon_r) - f(\varepsilon_1) \\ &= f_0(1 + \Gamma(\varepsilon_1 - \Delta\varepsilon_r)) - f_0(1 + \Gamma_1) \\ &= f_0(\Gamma(\varepsilon_1 - \Delta\varepsilon_r) - \Gamma_1)\end{aligned}$$

where

$$\begin{aligned}f_0 &= \text{notch frequency} \\ \Delta f_r &= \text{frequency resolution} \\ \Delta\varepsilon_r &= \text{strain resolution} \\ \varepsilon_1 &= \text{maximum strain} \\ \Gamma_1 &= \Gamma(\varepsilon_1)\end{aligned}$$

The resolutions depend heavily on the particular topology and the underlying model, so unfortunately the relationship will not reduce further. For a particular strain resolution and assuming the direct twin-T model for Γ, we have that

$$f(\varepsilon_1 - \Delta\varepsilon_r) = \frac{f_0}{1 + G(\varepsilon_1 - \Delta\varepsilon_r)}. \tag{4.36}$$

By substituting in equation 4.31 and shifting terms, we can relate $\Delta\varepsilon_r$ and Δf_r:

$$\begin{aligned}\Delta f_r &\leq f(\varepsilon_1 - \Delta\varepsilon_r) - f(\varepsilon_1) \\ &= \frac{f_0}{1 + G(\varepsilon_1 - \Delta\varepsilon_r)} - \frac{f_0}{1 + G\varepsilon_1} \\ &= \frac{f_0(1 + G\varepsilon_1 - 1 - G\varepsilon_1 + G\Delta\varepsilon_r)}{(1 + G\varepsilon_1 - G\Delta\varepsilon_r)(1 + G\varepsilon_1)} \\ &= \frac{f_0 G\Delta\varepsilon_r}{(1 + G\varepsilon_1 - G\Delta\varepsilon_r)(1 + G\varepsilon_1)}\end{aligned}$$

Thus, we arrive at the relationship that specifies the necessary frequency resolution to achieve the desired strain resolution:

$$\Delta f_r \leq \frac{f_0 G\Delta\varepsilon_r}{(1 + G\varepsilon_1 - G\Delta\varepsilon_r)(1 + G\varepsilon_1)} = \frac{f_1 G\Delta\varepsilon_r}{1 + G(\varepsilon_1 - \Delta\varepsilon_r)}. \tag{4.37}$$

Lower notch frequencies require finer resolution to resolve changes in strain, which makes sense. If instead we want to determine the strain resolution given a known frequency resolution, equation 4.37 turns into an equality and just needs to be rearranged:

$$\Delta\varepsilon_r = \frac{\Delta f_r(1+G\varepsilon_1)^2}{G\left(f_0+\Delta f_r(1+G\varepsilon_1)\right)} = \frac{\Delta f_r(1+G\varepsilon_1)}{G(f_1+\Delta f_r)} \tag{4.38}$$

In both cases, it is worth noting that the frequency resolution should be *higher* than the resolution used for the PSD generation to avoid quantization errors and ensure the desired strain resolution is achieved. The minimum required sampling period, independent of sampling frequency, is then

$$T_{\min} = \frac{1}{\Delta f_r}, \tag{4.39}$$

but choosing this sampling period exactly means that the periodogram estimate to achieve a resolution of Δf_r will be very noisy. This should be scaled up, perhaps ten times, to benefit from Welch's method.

Another important point about the strain resolution is that it can be bounded by the frequency resolution. The frequency resolution is largely a hardware limitation, so typically the designer will like to know how fine the strain resolution can be given the hardware constraints. We can relate both terms to their bandwidth to compare them in terms of discrete, resolvable points. For strain, the bandwidth goes from zero to the maximum supported strain, so just ε_1. Therefore, given a desired resolution $\Delta\varepsilon_r$, we expect the number of resolvable points to be at least

$$N_{\varepsilon,\text{desired}} = \left\lfloor \frac{\varepsilon_1}{\Delta\varepsilon_r} \right\rfloor. \tag{4.40}$$

The frequency bandwidth ranges from f_1 to f_0, which is just

$$f_0 - f_1 = -f_0\Gamma_1. \tag{4.41}$$

Substituting our model for Γ we have

$$f_0 - f_1 = \frac{f_0 G\varepsilon_1}{1+G\varepsilon_1}. \tag{4.42}$$

Finally, we determine the number of resolvable frequency points in the sensor bandwidth to be

$$N_f = \left\lfloor \frac{f_0-f_1}{\Delta f_r} \right\rfloor = \left\lfloor \frac{-f_0\Gamma_1}{\Delta f_r} \right\rfloor = \left\lfloor \frac{f_0 G\varepsilon_1}{\Delta f_r(1+G\varepsilon_1)} \right\rfloor. \tag{4.43}$$

Looping back to the strain resolution, there can only be as many resolvable strain detections as there are for the frequency detection, so we can update our definition of 4.40 to

$$N_\varepsilon = \min\left\{N_{\varepsilon,\text{desired}}, N_f\right\}. \tag{4.44}$$

The finest strain resolution possible given the frequency resolution gives us a lower bound, namely

$$\Delta\varepsilon_r \geq \frac{\varepsilon_1}{N_f}. \tag{4.45}$$

This is useful to keep in mind to avoid trying to design a network with infeasible constraints.

In the direct case, the only variables that can be tuned are the gauge factor and resolution. In other configurations, we may also want to find reasonable values for one or more ratios k_i and one or more gauge factors G_i. In that case, it is helpful to consider an optimization problem using the constraints derived above, for example

$$\begin{aligned}
\min \quad & w_1\frac{f_0 - f_1}{f_0} + w_2\frac{\Delta\varepsilon_r}{\varepsilon_1} - w_3\frac{\Delta f_r}{f_0 - f_1} \\
\text{where:} \quad & \Gamma(\varepsilon) = -G\varepsilon/(G\varepsilon + k + 1) \\
& \Gamma_1 = \Gamma(\varepsilon_1) \\
& f_1 = f_0(1 + \Gamma_1) \\
& \Delta f_r = f_0(\Gamma(\varepsilon_1 - \Delta\varepsilon_r) - \Gamma_1) \\
& N_f = -f_0\Gamma_1/\Delta f_r \\
\text{subject to:} \quad & \Delta\varepsilon_r > \varepsilon_1/N_f \\
& f_0 - f_1 > f_0/Q
\end{aligned}$$

where we minimize bandwidth and strain resolution but maximize frequency resolution with priorities set by the weights $[w_1, w_2, w_3]$. Ideally, the sensor has a small bandwidth and fine strain resolution, but larger frequency resolution to reduce noise in the PSD computation. Although not listed here, other variables will also have constraints. For example, a constraint on the gauge factor dependent on sensor type and fabrication; minimal or maximal frequencies; and non-negative constraints on the ratios k. This can be accomplished using a sequential least squares program or more general nonlinear program using an interior point method, since the Jacobians are fairly simple to compute.

Example placing three sensors

Suppose that we want to place three twin-T notch filters designed using parallel plate capacitors (i.e., $G = 1$). The lowest notch frequency is $f_{min} = 10\,\text{kHz}$ and for each device we want to support up to $\varepsilon_1 = 100\,\%$ strain. There should be a margin between sensors of at least one-half the notch's zero-strain bandwidth. Now we want to design the zero-strain frequencies for each notch and choose a frequency resolution Δf_r to achieve a strain resolution of $\Delta\varepsilon_r = 5\,\%$.

Using equation 4.32, with $f_1 = 10\,\text{kHz}$, $\varepsilon_1 = 1$, and $G = 1$, we can determine the frequency of the first notch as $f_0 = 10000(1 + 1) = 20\,\text{kHz}$. The frequency resolution necessary to achieve $\Delta\varepsilon_r = 0.05$ should be chosen here, as the lowest frequency notch will require the greatest accuracy to resolve the maximum strain. Putting in the values yields approximately $\Delta f_r \leq 256\,\text{Hz}$. In practice, we would like a frequency resolution of $\Delta f_r/10 = 25.6\,\text{Hz}$. We can reasonably expect at least this performance sampling at about 25 Hz. At this point, it is worth noting that it might be prudent to increase the starting frequency or relax the strain resolution. Note that this is a very conservative constraint, as only the lowest frequency sensor would need this frequency resolution, and only to resolve the difference between the maximum strain of $\varepsilon = 1$ and $5\,\%$ less, i.e., $\varepsilon = 0.95$. For comparison, accepting twice the error rate at the maximum strain doubles the maximum sampling rate and still maintains a strain resolution of $5\,\%$ over $40\,\%$ strain. Figure 4.17 shows a few resolution plots for comparison. The starting notch frequency can also be increased as it is directly proportional to the required frequency resolution.

Adding the next sensor is straightforward. The only decision to be made is the starting frequency of the next sensor; then the previously described process can be repeated. The zero-strain notch frequency of the previous sensor is at 20 kHz. This is theoretically the maximum frequency response, so the minimum frequency response of the next sensor at maximum strain should not drop below this value. However, some margin should be added to ensure that the peaks never overlap and become undetectable. Using the $-3\,\text{dB}$ bandwidth of the previous sensor is sufficient to accomplish this. The bandwidth can be determined using the designed Q-factor. In the case of the twin-T,

$$\text{BW}_{3\,\text{dB}} = \frac{f_0}{Q} = 4(1 - K)f_0. \tag{4.46}$$

$$Q = \frac{f_0}{\text{BW}_{3\,\text{dB}}}$$

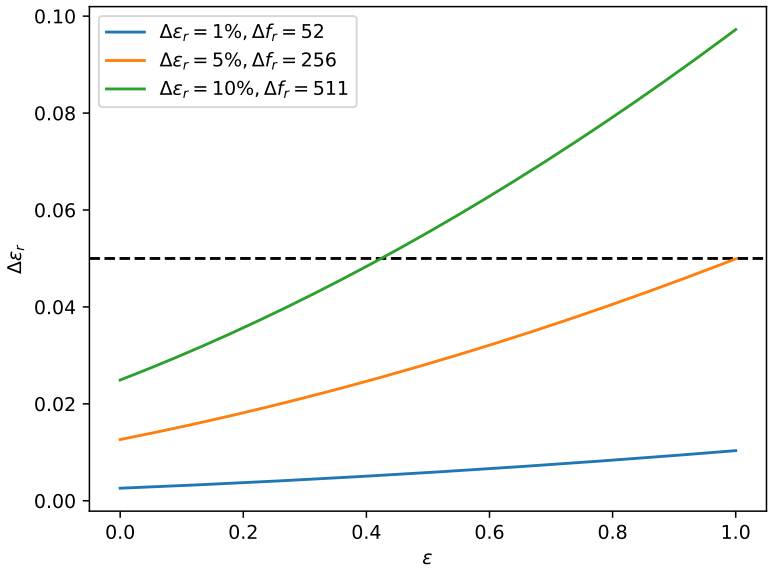


Figure 4.17: Single sensor design diagram.

Suppose that we choose $K = 0.99$ for a Q factor of $Q = 25$. Then the 3 dB bandwidth is 800 Hz, so the next sensor should be placed at $f_{1,\text{next}} = f_{0,\text{previous}} = 20.8$ kHz Now the notch frequency is calculated to be $f_{0,next} = 20800(1+1) = 41.6$ kHz. For this device, the minimum strain resolution using the frequency resolution of $\Delta f_r = 256$ Hz determined previously is $\Delta\varepsilon_r < 2.5\,\%$.

The final sensor is placed in the same manner. The 3 dB bandwidth is determined to be 1.664 kHz in this case, so $f_1 = 43.264$ kHz. Then the notch frequency is found to be $f_0 = 86.527$ kHz. It is apparent that the bandwidth is doubling with each added sensor. This is due to the gauge factor, which will be explored next.

4.5.3 Bounding the sensor limit for a given bandwidth

Suppose we have a given bandwidth and we would like to know how many sensors can be packed into that limited space. If we can assume that the model Γ is the same (i.e., all sensors have the same topology) and that the desired maximum strain is the same for each sensor, this can easily be derived. If the system is not homogeneous, the following approach can still be used to estimate the maximum density, but simplifying assumptions need to be made, for example choosing the *average* model and *average* strain. Otherwise, the sequential approach from section 4.5.2 should be used.

Suppose that we want to place a sensor network in the bandwidth $[f_L, f_H]$ Hz, assuming

a uniform maximum strain ε_1 for each sensor with underlying model $\Gamma(\varepsilon)$. We want to find N, the number of sensors that can be placed in this bandwidth. $\Gamma(\varepsilon_1) = \Gamma_1$, a constant. Starting from f_L, we can determine the frequency of the first notch, $f_1 = f_L(1+\Gamma_1)^{-1}$. In this section, f_n will indicate the n-th notch, a slight deviation from its previous use. Having established the first sensor's starting frequency f_1, we can move on to the second, which will operate from $f_2 = f_1(1+\Gamma_1)^{-1}$ to $f_1 = f_L(1+\Gamma_1)^{-1}$. The recursive relationship can now be seen. Given f_L,

$$\begin{aligned} f_1 &= f_L(1+\Gamma_1)^{-1} \\ f_2 &= f_1(1+\Gamma_1)^{-1} \\ &= f_L(1+\Gamma_1)^{-1}(1+\Gamma_1)^{-1} \\ &= f_L(1+\Gamma_1)^{-2} \\ f_3 &= f_2(1+\Gamma_1)^{-1} \\ &= f_L(1+\Gamma_1)^{-3} \\ &\dots \end{aligned}$$

and we can describe the full bandwidth in terms on N, noting that f_H is the notch frequency for the final sensor, i.e., $f_H = f_N$:

$$f_H = f_L(1+\Gamma_1)^{-N}. \tag{4.47}$$

Then to determine N, we compute the logarithm of both sides of the equation and rearrange terms:

$$N = \left\lfloor \frac{\log(f_L/f_H)}{\log(1+\Gamma_1)} \right\rfloor. \tag{4.48}$$

Notably absent from this determination is any sort of margin. Recall that previously we established the 3 dB bandwidth as a spacer between filters. If we can specify the margin in the form of a constant times the notch frequency f_{notch}, we can add it to the recursive form without too much trouble. The 3 dB bandwidth is just that sort of margin, assuming that Q is constant:

$$\mathrm{BW}_{3\,\mathrm{dB}} = \frac{f_{\mathrm{notch}}}{Q} = \alpha f_{\mathrm{notch}}, \tag{4.49}$$

where $\alpha = Q^{-1}$ is a constant and f_{notch} is the notch frequency. Note, however, that the first sensor does not need a margin at the notch frequency, it can align exactly on the specified

bandwidth. Revisiting our recursive rules, we have

$$\begin{aligned} f_1 &= f_L(1+\Gamma_1)^{-1} \\ f_2 &= (f_1+\alpha f_1)(1+\Gamma_1)^{-1} \\ &= f_1(1+\alpha)(1+\Gamma_1)^{-1} \\ &= f_L(1+\alpha)(1+\Gamma_1)^{-2} \\ f_3 &= f_2(1+\alpha)(1+\Gamma_1)^{-1} \\ &= f_L(1+\alpha)^2(1+\Gamma_1)^{-3} \\ &\dots \end{aligned}$$

Once again we can describe the full bandwidth in terms of N:

$$f_H = f_L(1+\alpha)^{N-1}(1+\Gamma_1)^{-N}. \tag{4.50}$$

N is found again by computing the logarithm and rearranging terms:

$$N_\alpha = \left\lfloor \frac{\log(f_L/f_H) - \log(1+\alpha)}{\log(1+\Gamma_1) - \log(1+\alpha)} \right\rfloor \tag{4.51}$$

If there is no bandwidth, equation 4.51 reduces to 4.48 as expected. Returning to the sequential example of section 4.5.2, plugging in the final bandwidth of $f_L = 10\,\text{kHz}$ and $f_H = 86.528\,\text{kHz}$, along with the parameters used to build the network ($\alpha = 1/Q = 25$, and $\Gamma_1 = \Gamma(\varepsilon_1) = -1/2$), equation 4.51 without the floor yields exactly $N = 3$ as expected. Alternatively, if we already knew how many sensors we wanted to place and either the lower or upper limit of the bandwidth, we can use equation 4.50 to determine the required bandwidth. As before, the frequency and strain resolution are completely determined by the particular topology and the lower frequency f_L.

We may be interested in the theoretical sensor limit in a given bandwidth. This is easy to define knowing only the bandwidth $[f_L, f_H]$ and the frequency resolution Δf_r. In the most extreme case, you only need one bit of resolution to qualify a sensor as having detected something. This limiting case may be useful if the device being designed only needs to detect that the strain exceeded some threshold, therefore only binary signal is required. Recalling the relations in equation 4.43, with no margin, the number of sensors that can fit in a bandwidth given only one resolvable point is just

$$N_{\max} = N_f = \left\lfloor \frac{f_H - f_L}{\Delta f_r} \right\rfloor. \tag{4.52}$$

Here we make the assumption that we can form some network that yields one resolvable frequency shift at the desired strain. No assumption of homogeneity is made. In fact, the sensors cannot be homogeneous or the sensor limit will be much lower, governed by equation 4.48 instead. More realistically, we will want to maintain some margin. Assuming the 3 dB margin as usual, we can expand the terms for the first few sensors:

$$\begin{aligned}
f_1 &= f_L + \Delta f_r \\
f_2 &= f_1(1+\alpha) + \Delta f_r \\
&= f_L(1+\alpha) + \Delta f_r(1+\alpha) + \Delta f_r \\
f_3 &= f_2(1+\alpha) + \Delta f_r \\
&= f_L(1+\alpha)^2 + \Delta f_r(1+\alpha)^2 + \Delta f_r(1+\alpha) + \Delta f_r \\
&\dots
\end{aligned}$$

From here we can see there is a term $f_L + \Delta f_r$ scaled by $(1+\alpha)^{N-1}$, along with a geometric series with $N-2$ terms, i.e.,

$$f_N = f_H = (f_L + \Delta f_r)(1+\alpha)^{N-1} + \sum_{k=0}^{N-2} \Delta f_r (1+\alpha)^k. \tag{4.53}$$

the geometric series can be rewritten in closed form, yielding

$$\begin{aligned}
f_H &= (f_L + \Delta f_r)(1+\alpha)^{N-1} + \Delta f_r \left(\frac{1-(1+\alpha)^{N-1}}{1-(1+\alpha)} \right) \\
&= (f_L + \Delta f_r)(1+\alpha)^{N-1} + \frac{\Delta f_r}{\alpha} \left((1+\alpha)^{N-1} - 1 \right) \\
&= \left(f_L + \Delta f_r + \frac{\Delta f_r}{\alpha} \right) (1+\alpha)^{N-1} - \frac{\Delta f_r}{\alpha}.
\end{aligned} \tag{4.54}$$

Now we can take a logarithm and solve for N. Doing so yields

$$N_{\max,\alpha} = \left\lfloor 1 + \frac{\log\left(\frac{f_H + \Delta f_r/\alpha}{f_L + \Delta f_r(\alpha+1)/\alpha} \right)}{\log(1+\alpha)} \right\rfloor \tag{4.55}$$

for $\alpha > 0$. For a bandwidth of 1 – 2 kHz, $Q = 25$, $\alpha = 1/Q$, and a frequency resolution of $\Delta f_r = 100$ Hz, this yields a maximum sensor count of 6. If the frequency resolution is set to 10 Hz instead, the sensor count goes up to 15. Increasing the notch Q-factor to 100 allows for 41 sensors. If there is no margin at all we would instead use equation 4.52 there would be

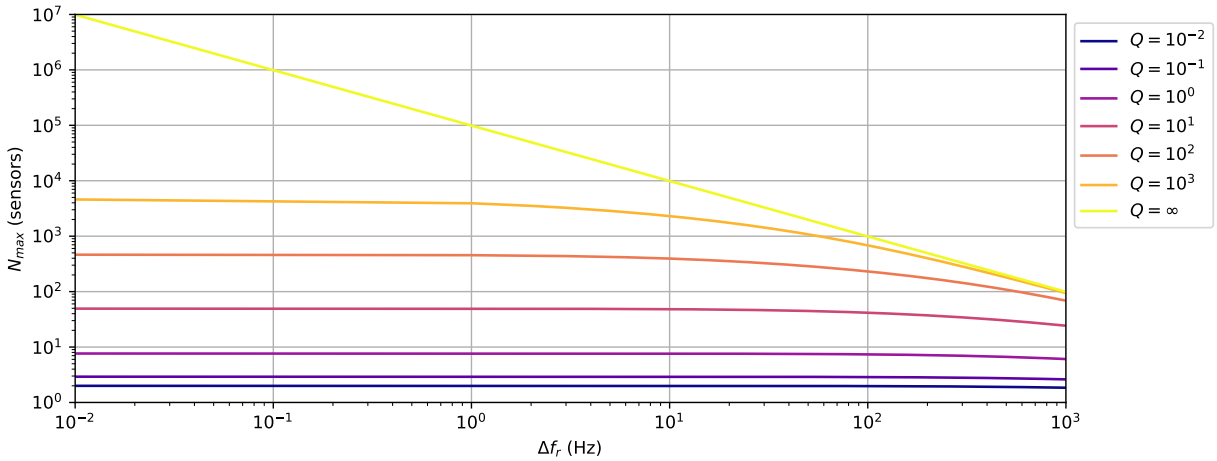


Figure 4.18: $N_{\text{max,3dB}}$ given the frequency resolution Δf_r for a total bandwidth between $1-100\,\text{kHz}$ plotted for different values of Q. $Q = \infty$ is the limiting $\alpha = 0$ case, with no margin.

room for 100 sensors. Using the original parameters but spanning instead from $1-10\,\text{kHz}$, 32 sensors could be fit. Figure 4.18 gives some intuitive insight to the upper limits. Upper limits on sensor counts as a function of the frequency resolution are plotted for a handful of Q values. The quality factor determines the margin, which is the largest factor in this limiting case. These plots represent the upper bounds on sensor density under the very restrictive assumption of binary detection. When greater measurement fidelity is required, these numbers will decrease.

4.6 Estimating strain from sensor output

The sensing device implements a notch filter with a strain-sensitive rejection frequency as outlined in section 4.4. In order to infer a strain, three procedures must be defined: how to detect the notch frequency, how to infer the strain, and how to distinguish responses from different sensors. These are the subjects of this section.

4.6.1 Notch frequency detection

With a notch filter in place, the notch frequency can be determined by chirping the network and looking for peaks. To interrogate the network, a signal generator is required. This can be implemented with a small direct digital synthesis (DDS) IC such as the AD9833 or AD9834 (Analog Devices). The output should be buffered and amplified to take full advantage of the range of the ADC in use. A chirp can be output by generating a sinusoid with a time-varying

output frequency. A logarithmic chirp is helpful when searching over multiple decades for frequency responses, as it uniformly spreads the spectral energy over each decade. The chirp frequency at a particular time is computed as

$$f_{\text{chirp,log}}(t) = f_0 \left(\frac{f_1}{f_0} \right)^{t/t_1}, \tag{4.56}$$

where f_0 and f_1 are the starting and ending frequencies of the chirp, respectively, and t_1 is the length of the chirp in units of t. This assumes that the chirp starts at time $t = 0$. If the notch frequency search is conducted over a much narrower frequency range, a simpler linear chirp may be warranted. A linear chirp can be generated with

$$f_{\text{chirp,linear}}(t) = f_0 + (f_1 - f_0)\frac{t}{t_1}. \tag{4.57}$$

Both the signal output and input need to be sampled to determine the response, as non-linearities, glitches, or other artifacts in the input need to be captured to ensure robust peak detection in the face of noise and interference. The signals should be captured at a sampling frequency ideally ten times that of the highest chirp frequency to reduce aliasing artifacts. For the 1 – 100 kHz search performed in this work, the data was sampled at 1.25 Msps.

With each collected chirp, the transfer function can be estimated as

$$\widehat{H}(f) = \frac{S_{xy}(f)}{S_{xx}(f)} \tag{4.58}$$

where S_{xx} is the power spectral density (PSD) of the input and S_{xy} is the cross power spectral density (CPSD) of the input and output. Both can be estimated using Welch's approach [122] which averages periodograms computed over overlapping time segments to reduce noise. At this point, peak detection already detects notches well, but there are often measurement artifacts that throw off a naïve search for minima. The transfer function can be smoothed using a Savitsky-Golay filter [96] which efficiently performs a least-squares fit of a low-order polynomial over a short window to allow for data smoothing without removing important features. In doing this, the notch frequency is easily identified even from a noisy signal, as shown in figure 4.19. This example was chosen because it includes 60 Hz mains hum in addition to signal attenuation downstream of the notch and somewhat poorly matched capacitors in a twin-T sensor configuration.

What is notable in this example is that, even in light of considerable noise and poor filter performance notch detection performs well. Later the details of the twin-T sensor implementation are explored and it is observed that the strain estimation performs reliably even under these conditions.

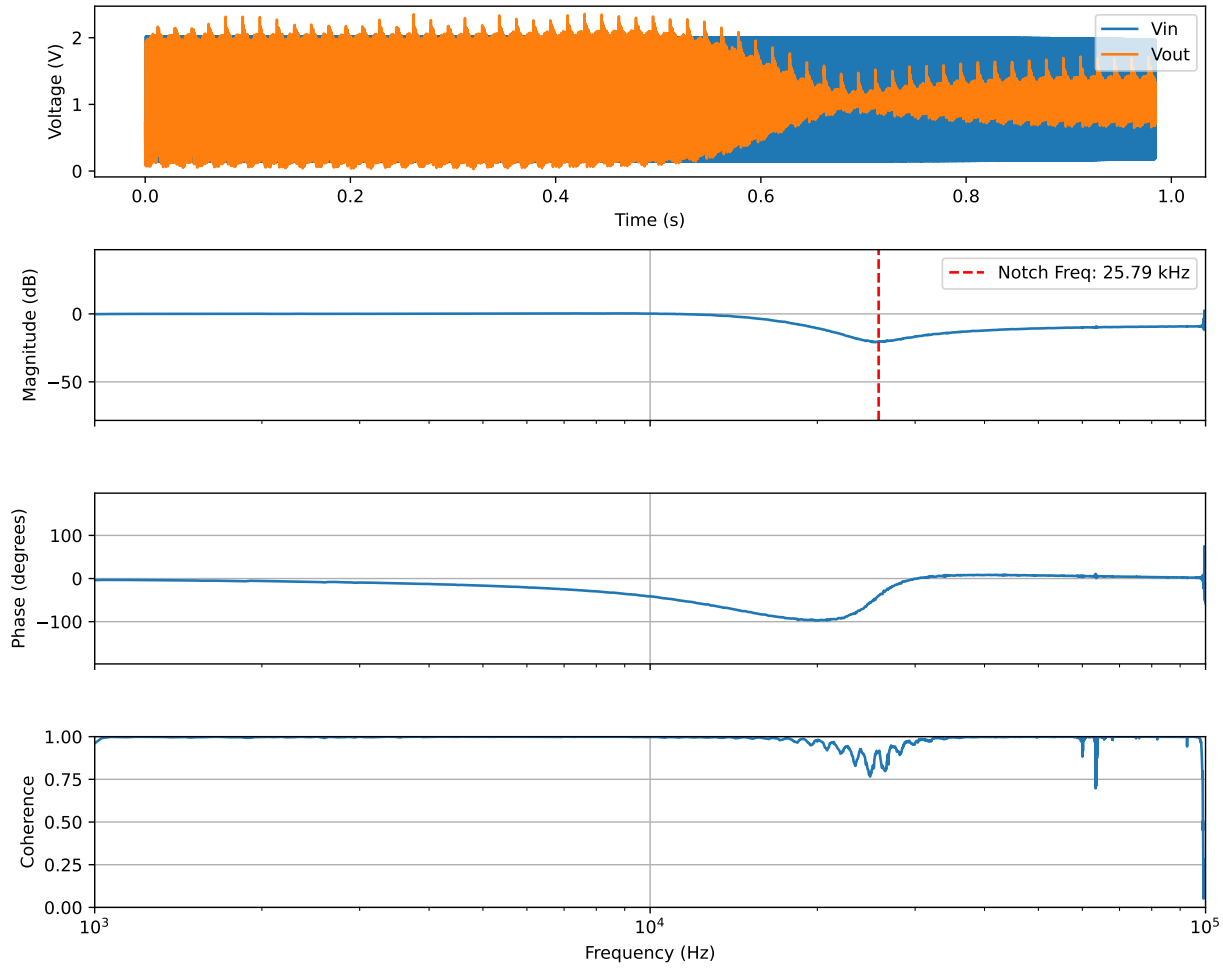


Figure 4.19: Notch detection performing well on poorly conditioned signal. The input (top, orange) is a logarithmic chirp with evident 60 Hz mains hum and notable post-notch attenuation. The notch is still detectable using the smoothed magnitude of the transfer function.

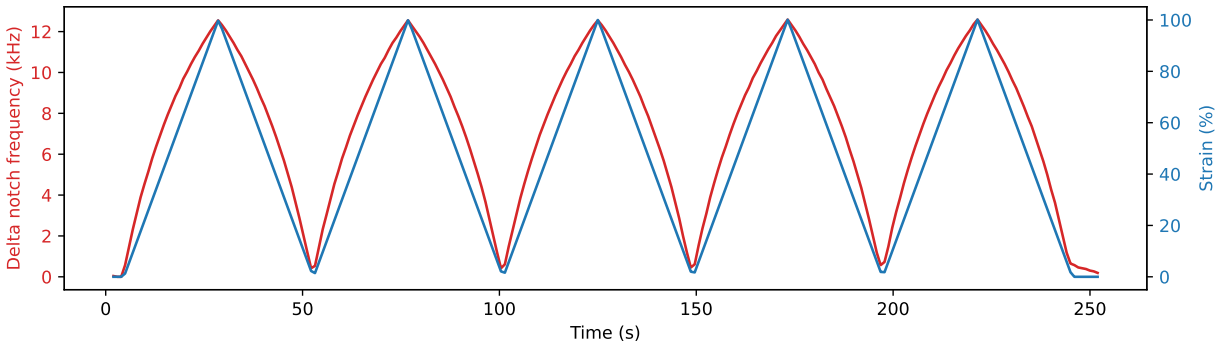


Figure 4.20: Measured device strain (blue) and sensor frequency shift (red).

4.6.2 Strain estimation

For each chirp, the procedure explained in the previous section (4.6.1) is used to estimate the notch frequency. Sensor data can be fit to the appropriate frequency shift model (that is, from tables 4.1 and 4.2 for the twin-T topology or tables 4.3 and 4.4 for the Bainter topology). Based on the implementation, typically only the zero-strain frequency and gauge factor need to be estimated, but these can also be manually calibrated using either a few points or derived directly from the design if the device capacitance and gauge factor are already known accurately.

A sample device was cyclically strained using an Instron device for precise measurement of the tensile strain. One-second-long logarithmic chirps were performed continuously using an Arduino-controlled DDS, with output buffered and amplified using a non-inverting amplifier. The frequency shift over five cycles of 100 % strain is shown superimposed with the measured strain in figure 4.20.

A least-squares fit using the Levenberg-Marquardt algorithm proposed by Moré [74] and implemented in Scipy's `curve_fit` routine. Doing so yields a near perfect fit to the data, as shown in figure 4.21. Figure 4.22 shows an excellent correlation between Instron ground truth strain measurements and strain predictions made using the sensor. Both the estimated gauge factor and the zero-strain frequency were close to the designed values.

4.6.3 Distributed sensing with multiple sensors

To incorporate multiple sensors in a single network, multiple notches must be identified and associated with the correct sensor. The easiest way to achieve this is to design spectrally isolated bandwidths for each sensor used. In this way, each sensor response can only fall within a set frequency band, so disambiguating them is trivial. Section 4.5 explains the

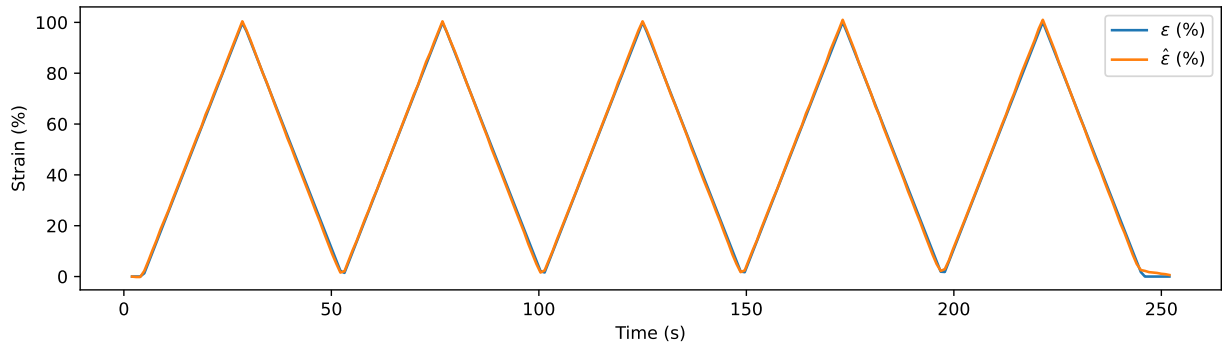


Figure 4.21: Estimated versus measured strain after fitting.

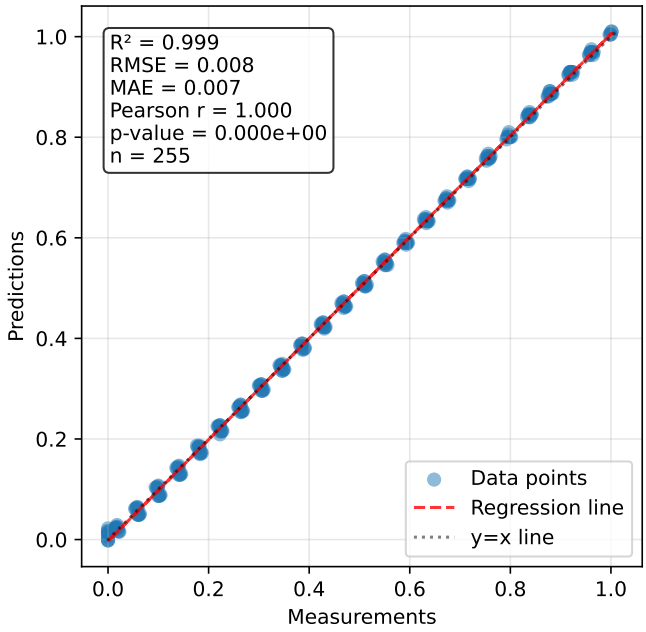


Figure 4.22: Measurement and prediction correlation.

design guidelines using this strategy. At a high level, the exact same approach is followed for the single-sensor detection, except that the peak detection is done independently for each sensor's bandwidth.

This is not the only approach to interfacing multiple devices on a single network. To support higher sensor densities by overlapping their bandwidths, a state estimator and tracker can be implemented. This approach might work well out of the box if it is not anticipated notch frequencies will overlap for considerable amounts of time, such that the tracker will not be able to quickly disambiguate tracks once they eventually diverge. Adding geometric constraints might also be helpful for disambiguation as some strain configurations might be be physically infeasible. However, shared-bandwidth approaches are out of the scope of this work.

For experimental validation of this approach, see section 4.7.2.

4.7 Hardware validation

4.7.1 Device fabrication

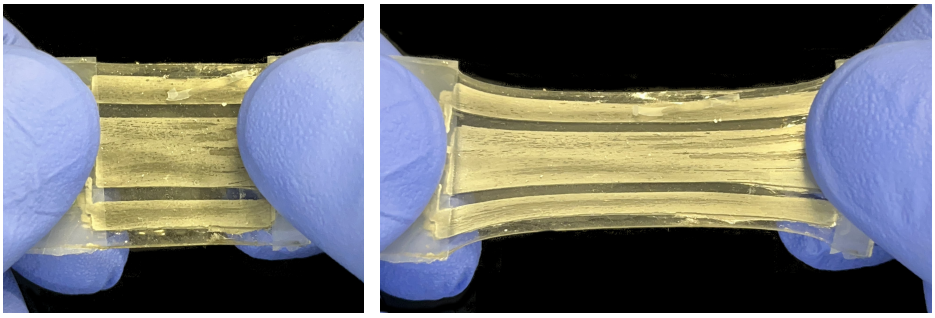
The device fabrication follows largely the same process regardless of the topology used. Printed circuit boards (PCBs) were designed for the twin-T and Bainter notch topologies, with terminals to connect strain-sensitive capacitors and a space to add a fixed capacitance in parallel (i.e., the *additive* model). Sensor PCBs are assembled to implement the designed filter specifications. Double-sided VHB tape (3M) is used as both the substrate for the device and the dielectric for strain-sensitive capacitors. Simple parallel-plate capacitors are made by patterning electrodes with oxidized eutectic Gallium-Indium liquid metal paste (OGaIn) on either side of the VHB using stencils cut with a CO_2 laser. The stencils are also patterned to connect the electrodes and device terminals to the PCBs. The PCB is pressed in place on the patterned VHB. A conductive, biphasic liquid metal ink developed by Reis Carneiro, Majidi, and Tavakoli in [88] is used to pattern elastic wires to connect the OGaIn electrodes to the PCB pads. To prepare the ink, styrene-isoprene-styrene block co-polymer (432393, styrene 14 % by weight, Sigma-Aldrich) is dissolved in toluene in a ratio of 25:75 by weight. The dissolved SIS is combined in a vial with silver flakes (41-071, Technic) and eutectic gallium-indium (EGaIn) in a weight ratio of 1:2:5. The vial is mixed using a planetary mixer (AR-100, Thinky) at 2000 rpm for three minutes without defoaming. The contents are mixed by hand with a tongue depressor for about one minute and then mixed in the planetary mixer for another three minutes. The ink is poured into a syringe and dispensed by hand. To form a reliable connection point for strain testing, regular Scotch tape is used as a stiffener on

either end to form clamping points and to avoid destroying the device while characterizing its response.

Twin T design

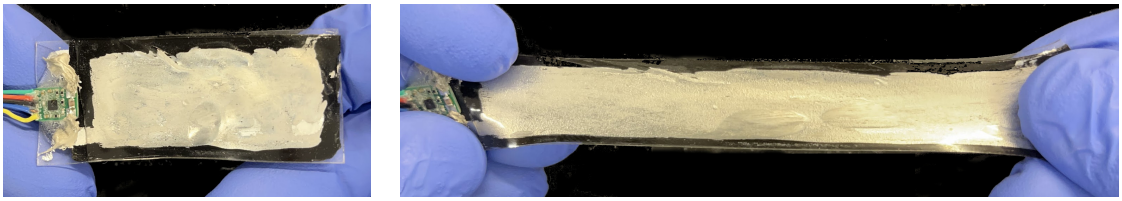
The twin-T notch filter was implemented with a custom PCB with the three filter capacitor connections exposed on the edge of the board to allow easy electrode connection. Pads were added for three parallel capacitors to be used if strain sensitivity was to be tuned. Figure 4.23 shows two photos of an implemented sensor, one with no tensile load and another stretched about twice its original length. At zero strain, the capacitors are 20 mm in length and roughly 3 mm (small capacitors) or 8 mm (large capacitor) wide with 2 mm spacing. The VHB dielectric is 0.5 mm thick.

To match the capacitors to the appropriate ratios, the capacitors were initially patterned to have a 1:2:1 area ratio. However, in doing this, the capacitance measured with a high-precision benchtop LCR meter (894, B&K Precision) was about 10:12:10 pF instead. This is likely due to fringing effects from the long electrodes in close proximity, similar to the effect seen by Fassler and Majidi [28]. Work by Shin and Bergbreiter combined models of parallel-place capacitors with those due to the fringing fields based on the 3-strip coplanar waveguide models developed by Ghione and Naldi in [32], yielding mean average percent error (MAPE) improvements on the order of about 15 % over the parallel-plate model [101]. However, the contribution of fringing field capacitance in this design is considerably less since the parallel-plate capacitance derives from the much larger surface area of the electrode, not from the relatively small sidewall as in Shin and Bergbreiter's work, which evaluated interdigitated comb capacitors. This may still limit the effectiveness of the sensor when the ground is noisy, as discussed by Ghione and Naldi in [32] and seen in figure 4.19 where a connection of the test equipment and sensor to mains AC generates substantial artifacts in the sensor response, despite no noticeable degradation in the same circuit with ceramic capacitors in place of the OGaIn strips. To some extent, this can be mitigated by decoupling the sensor grounds from the test equipment. The capacitors also needed to be slightly scaled to achieve the desired 1:2:1 ratio. Sensor performance did not seem noticeably degraded by these effects, the processing seems reasonably robust to noise, even with lower sampling periods. Although not evaluated theoretically, in testing any change in fringing capacitance with strain seems negligible, or at least proportional such that the fitted model which does not account for fringing effects still performs well.



(a) (b)

Figure 4.23: Strain-sensitive twin-T notch filter. (a) No tensile strain. (b) 200 % tensile strain.



(a) (b)

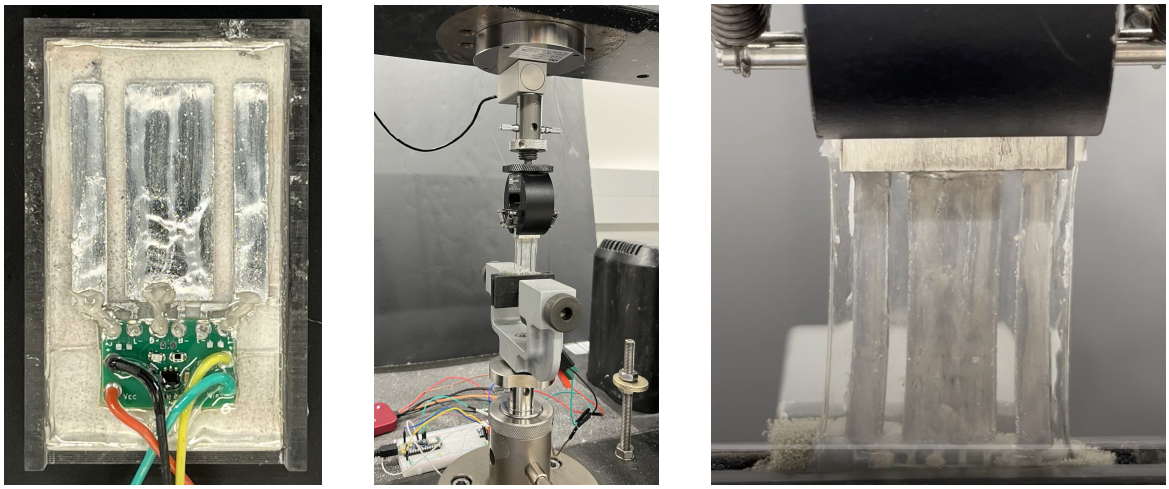
Figure 4.24: Strain-sensitive twin-T notch filter. (a) No tensile strain. (b) 200 % tensile strain.

Bainter design

The Bainter notch, while somewhat more difficult to analyze, was implemented with a custom PCB roughly equivalent to the twin-T notch. Doing so greatly improves the performance for designs with small capacitances. Fabrication is simplified because only one strain-sensitive element needs to be attached and matching within some tolerance is not required for good performance. In this case, OGaIn was simply painted on a strip of VHB without the use of a stencil. Another layer was added and the second electrode was painted on. The Bainter notch PCB was pressed in place on the VHB, and the liquid metal ink described previously was used to connect the electrodes to the PCB terminals. Scotch tape was used to stiffen the ends. Figure 4.24 shows an implemented Bainter notch sensor implementation unloaded and strained to about 200 %.

4.7.2 Sensor validation

Sensors of both notch varieties were characterized and demonstrated in single- and multi-unit configurations.



(a) (b) (c)

Figure 4.25: Experimental configuration for tensile testing. (a) A strain-sensitive twin-T notch filter. (b) Zoomed-out view of the filter clamped from above and below. (c) A close-up view of the sensor under strain.

Single unit strain tracking

The strain response of the fabricated sensors was evaluated using a universal testing system (5969, Instron) to perform controlled tensile tests. Chirps were generated using a direct digital synthesis (DDS) IC (AD9833, Analog Devices) controlled by a hobbyist microntroller (Feather nRF52840 Express, Adafruit). The chirp boundaries were marked with GPIO transitions and captured using a logic analyzer (Logic 8, Saleae). The same logic analyzer, configured for 1.25 Mbps analog capture, was used to record the chirp input along with the sensor output. An assembled twin-T sensor and the experimental setup are shown in Figure 4.25.

Three test scenarios were conducted with the twin-T notch filter. In the first, the sensor strain was maintained for five seconds and then strained another 5 % at 50 mm/min (4.2 %/s). This was repeated until the device was strained 50 %. In the second, the device underwent five 100 % strain cycles at 20 mm/min (1.7 %/s). The final experiment was a single-cycle staircase, with a five-second hold at each level and a 5 % strain ramp over 20 s (3 mm/min or 0.25 %/s) The results of the three tests are shown in figure 4.26. Each scenario described above occupies one row of sub-figures. The figures in the left column overlay the frequency shift (red) with the tensile strain (blue). A "bowing" effect is observed due to the non-linearity of the sensor response, exhibiting a larger frequency shift for lower strains and lower response at higher strain. The right column shows the results when the frequency shift is fitted to the appropriate model. The measured and predicted strains align nearly perfectly in all plots.

In the three scenarios, the estimated parameters were similar in all cases. The fitting

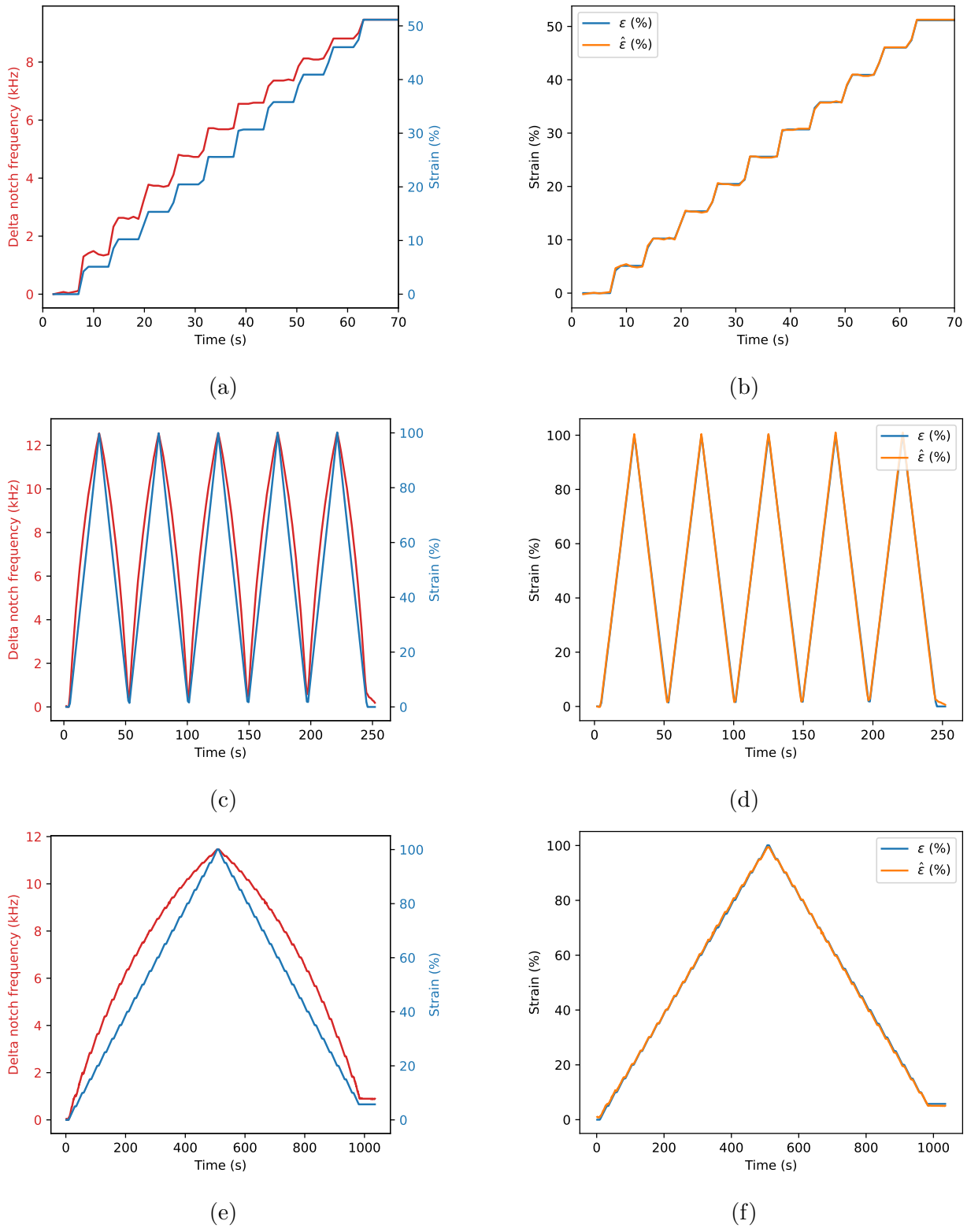


Figure 4.26: Twin-T notch strain sensing performance evaluation. (a) and (b) show the raw frequency shift (red) and fitted strain (blue) model, respectively, for a staircase strain evaluation. (c) and (d) show the same for a cyclic loading. (e) and (f) show a much longer duration evaluation.

of the model produced a fit to the gauge factor of $G = 0.95 \pm 0.05$ standard error, close to the expected $G = 1$ for parallel-plate capacitors. The fitted zero-strain frequency was $f_0 = 26\,\text{kHz} \pm 1.5\,\text{kHz}$ standard error. The larger error in the notch frequency over time has to do with the strain softening and elongation of the VHB substrate over time. To avoid starting the strain cycle while the sensor was in a buckled state, a small pre-strain of $100 - 200\,\text{mN}$ was applied at the beginning. The modeled frequency is also very close to the expected frequency based on the model, about 24.8 kHz. The difference may be due to fringing capacitance, component tolerances, or a mismatch of the capacitor ratios, although it is within 5 %.

Multiple unit strain tracking

Three Bainter notch sensors with differing zero-strain notch frequencies were fabricated. Figure 4.27 shows the sharp detection of the three notches with a single chirp. The three panels demonstrate complete decoupling of the signals, even as one sensor is strained so that its notch frequency overlaps with and eventually passes through another sensor's notch. Although figure 4.27 associates the sensors correctly with their respective notches, this may be much more difficult to do in practice if the notches are actually expected overlap. Some strategies for notch tracking in such scenarios were discussed previously.

Sensors can be easily reconfigured, and even heterogeneous networks can be formed. Figure 4.28 shows one Bainter notch connected on the same network as the twin-T notch. Twin-T notch performance is degraded due to the high resistances on the tee as described previously. However, in this case, it still remains decoupled from the other notch on the network since that attenuation occurs at frequencies higher than the notch, while the Bainter notch on the same network is operating at lower frequencies.

Aside from ensuring devices on the same network do not use conflicting frequency bandwidths, there is no additional work to be done to reconfigure sensors on a different network. In fact, the Bainter sensor used in the heterogeneous network from figure 4.28 was also included in the homogeneous network seen in figure 4.27. It also does not substantially impact the computational load on the system. All sensors are detected on a single chirp, so the additional computational burden comes from detecting and tracking additional notches. If the sensor bandwidths are mutually exclusive, there is almost no difference. It is also worth noting that the sensor count does not impact the sampling frequency either. So long as the design rules of section 4.5 are followed, the sampling frequency is only dependent on the designed strain and frequency resolution. One caveat to this rule is that, at high sampling rates, the chirp period becomes much smaller, so it can be more difficult to distribute the chirp energy uniformly across the frequency spectrum to generate a usable signal-to-noise

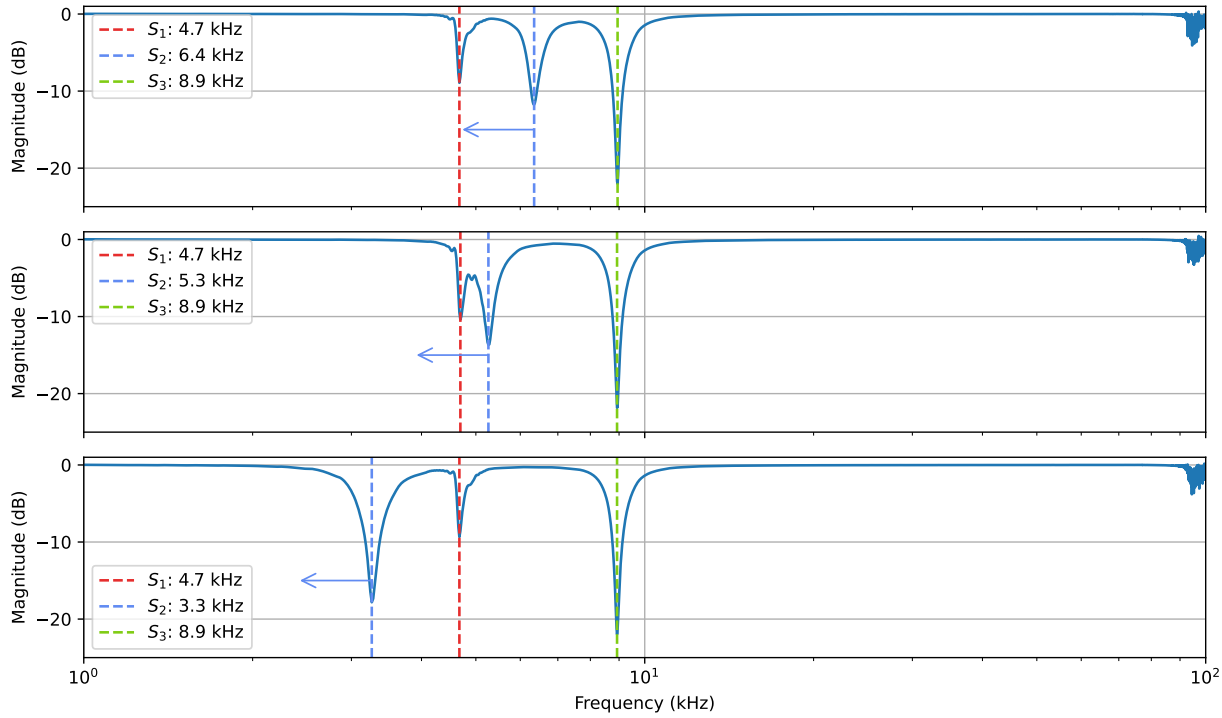


Figure 4.27: Multi-sensor notch detection on a three-sensor homogeneous sensor network with Bainter notches. The top plot shows the zero-strain response. The center sensor is strained to shift the notch frequency down, eventually overlapping (middle) and then passing through (bottom) the first sensor notch.

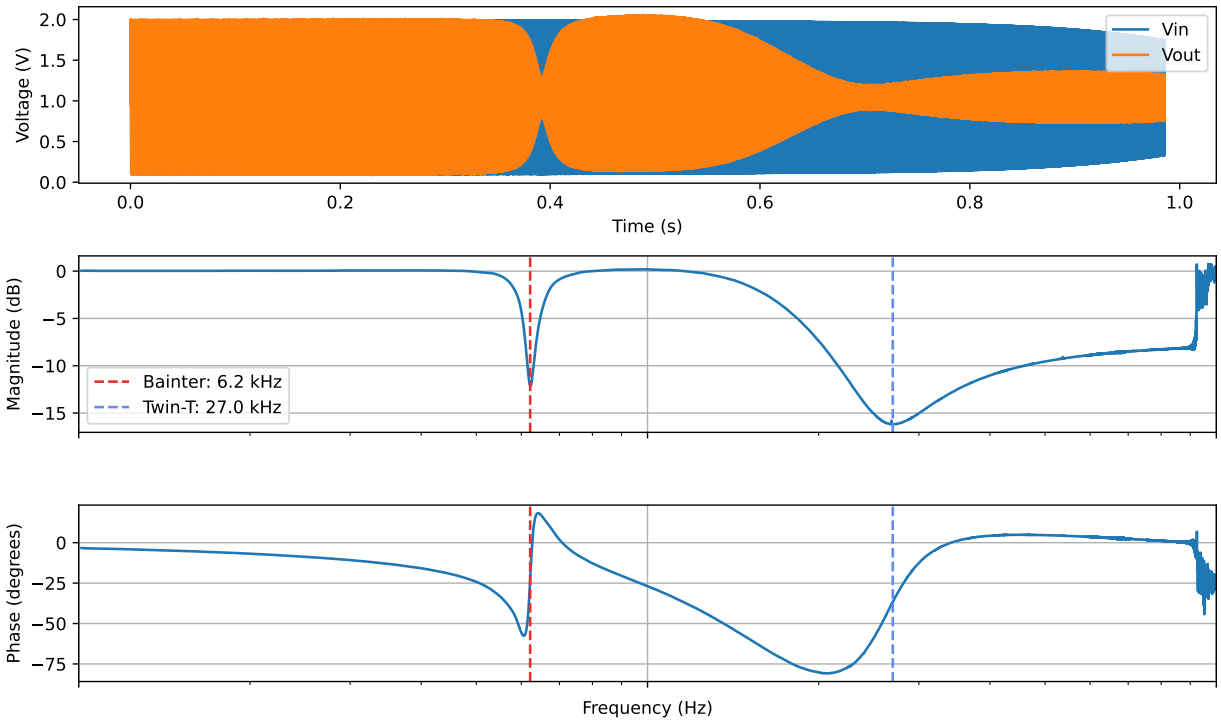


Figure 4.28: Multi-sensor notch detection on a heterogeneous network with one Bainter and one twin-T notch.

ratio. Figure 4.29 single chirp responses on the three-sensor network at three chirp periods corresponding to sampling at 1, 10, and 100 Hz. At faster sampling rates, the spectrum has more noise but it still exhibits well-defined notches.

4.8 Conclusions and future directions

I have demonstrated that encapsulating resistive or capacitive strain sensors in notch filters is an effective aid in the design of distributed sensor networks, especially for soft devices. Encapsulation provides sensor isolation, facilitating network reconfiguration and device modularity. Fixed resistors and capacitors can be used to tune the sensor response and sensitivity. The simple interface allows for easy connection of many sensors on a single network using only a few wires for the interface. These factors make such a network very flexible and accommodating to different design requirements for such networks, be it on electronic skins or oddly shaped robots. Design principles provide clear guidelines for network design.

The framework I developed provides a good model for designing such networks, but there are some interesting areas that we did not explore in this work. First, I used an oxidized liquid

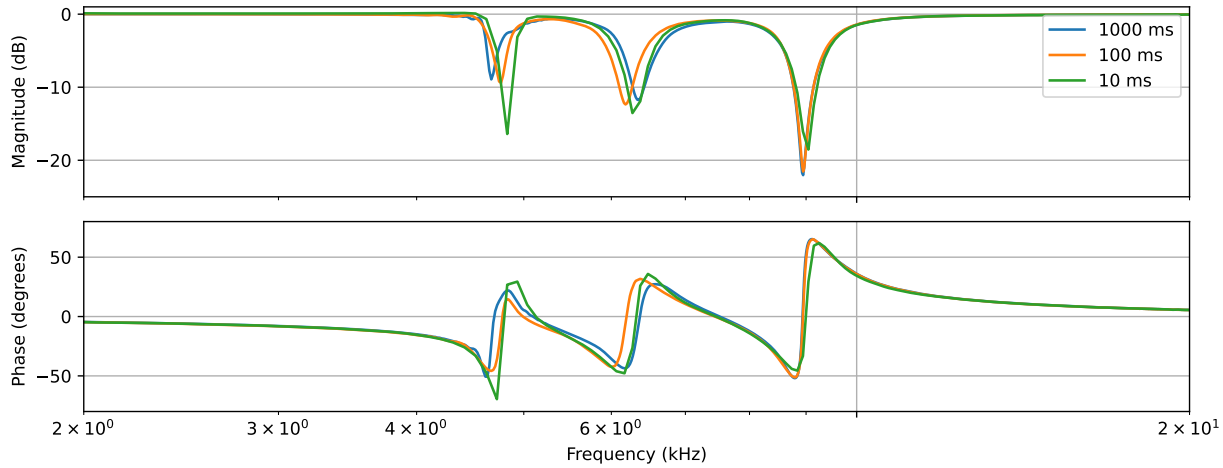


Figure 4.29: Computed transfer function magnitude and phase plots of a three-sensor network for 1 – 100 kHz logarithmic chirps over periods of 1000, 100, and 10 ms, corresponding to sensor sampling frequencies of 1, 10, and 100 Hz, respectively.

metal paste for the capacitive sensor electrodes. OGaIn has very high conductivity, so this works quite well in practice. However, if LMEE composites are used instead, the increased electrode resistance will affect the frequency response and may begin to diminish the observed capacitance at higher frequencies, as higher frequencies may not be able to penetrate the entire electrode as observed in previous works [123, 108, 103]. Next, while resistive strain sensors are covered in the theory developed and treated equivalently to capacitive sensors, this is only true so long as the sensors are operating in a reasonably linear regime. This is typical of capacitive sensors and a major benefit to using them, but resistive sensors can show considerably greater nonlinearity which, if used in nonlinear regions, would require reworking some of the equations. Finally, we did not cover hysteresis at all. Capacitive sensors typically exhibit low hysteresis anyway, so the effects were largely unobserved in this work, but in hysteresis can be substantial in resistive sensors especially. Existing works that treat this topic are largely still relevant, but there is one added layer of indirection through the filter transfer function, since the filter response is observed directly and the resistance only indirectly.

These challenges are intrinsic to these types of strain gauges in general, and it was not the intent of this work to address those specifically. It is worth noting that, aside from operating in nonlinear strain response regions, these challenges do not prove detrimental to the findings of this work. Despite these challenges, notch filter encapsulation provides a useful means of designing distributed strain-responsive sensor networks.

Chapter 5

Stiffness tuning for contact adaptation and improved sensing

In this chapter, I describe a novel sensor concept reliant on the close integration of traditional sensing components with nontraditional material systems to take advantage of their material properties and enable new capabilities. The key concept, as in the last chapter, is attention to how the sensors and materials can be designed into a device to augment each other's capabilities and either extend existing functionality or enable novel features, not easily attainable using traditional rigid materials. Multifunctional material systems more generally share similar challenges as soft systems, as their internal states can be complicated to estimate in practice due to complicated constitutive properties, dependent on variables difficult or impossible to measure in real-time.

But, just as sensorization of soft systems enables greater functionality, the same can be true in multifunctional material systems. In this chapter, I discuss a tunable-stiffness interface for robots and grippers used for both tactile sensing and traction adaptation. There are a variety of reasons to modulate stiffness at the contact interface, including greater slip control to adapt to changing conditions in locomotion or gripping tasks, along with tuning tactile sensing feedback.

An interesting device design that enables novel functionality in a multifunctional material interface is shown in figure 5.1. A shape-memory polymer (SMP) acts as an interfacial contact that can be hardened or softened to modulate its interaction with the environment. Active cooling or heating using a thermoelectric device (TED) bonded to the SMP enables this modulation. Originally evaluated for use in friction tuning for contact adaptation, I will demonstrate another interesting utility of the device as a tactile sensor whose capabilities are enhanced by the tunability of the interface its embedded within.

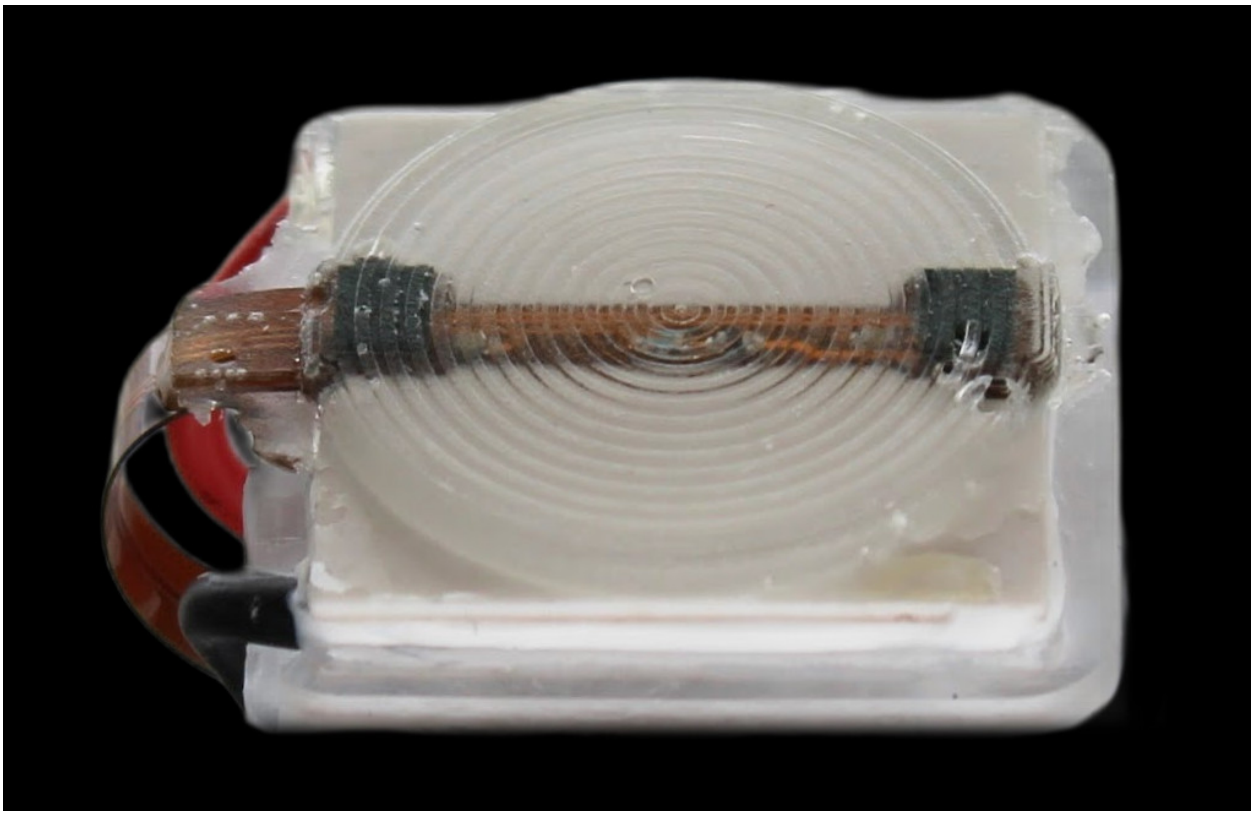


Figure 5.1: Tunable contact and tactile sensing device.

5.1 Overview

In this work, I introduce a novel end-effector interface design that pairs tunable-stiffness material with embedded temperature and acoustic sensing. I propose an easily fabricated, inexpensive, and tough multifunctional material system that can be used to tune contact friction and tactile sensing. Tunable contact friction enables gentler manipulation of soft, delicate, or fragile objects when desired, while also enabling a harder contact mode for sliding. Furthermore, with the appropriate surface design, the configuration can be used for dry adhesion gripping. The device also enables tunable contact dynamics for other applications, such as robot locomotion, where more or less traction would be desirable in different applications.

In addition to the benefits of controllable contact dynamics, the device also includes an embedded acoustic microphone for use in tactile sensing. I demonstrate the novel use of a tunable-stiffness interface to improve the discriminatory capability of the sensing device by allowing tactile measurements to be taken with both a soft and a hard interface. This improves material classification performance over simple multiple sampling with a static stiffness device. The device is easy to fabricate and utilizes a tough shape memory epoxy

interface for reliable operation and minimal fragility found in many other tunable stiffness end effectors.

5.2 Theory of operation

Many types of devices need to physically interact with their environment. This includes vehicles and robots that locomote over various terrains, manipulators that grip and touch objects, and human-computer interfaces that bridge human and virtual experiences. The contact interfaces on such devices are typically static materials that are designed to have the optimal material characteristics for a particular use-case. However, more general-purpose systems might be needed to perform tasks that require dynamic changes in interfacial contact properties. In typical systems, this can be accomplished by swapping out one end effector for another or using a system with multiple contact surfaces that can be used to accomplish the different objectives.

Interface materials can instead be designed with tunable surface characteristics. Using stiffness-tunable materials, for example, allows the modulus to be changed drastically, over many orders of magnitude, changing the contact dynamics of the interface considerably. When rubbing any particular surface, a harder interface slides more easily than a softer surface, which can better conform to the substrate and generate more resistance to sliding because of the closer contact. Stiffness and texture influence the static and dynamic friction as well as the stick-slip behavior of the device. In all, a tunable-stiffness interface can be used to optimize the contact dynamics on demand for a given objective. The device proposed here uses a tunable-stiffness shape memory polymer (SMP) epoxy that can change its modulus about 270 times. Using a heating element and temperature sensor encapsulated in the surface material, the interface temperature can be controlled to toggle between the hard and soft material states.

Adjustable contact dynamics can also improve tactile inferences by generating different acoustic signatures when rubbing the same surface using different contact modes. Vibrations will experience greater damping in softer materials, and material interactions will tend to produce lower-frequency feedback that decays more rapidly as more energy is lost as heat. Adhesion typically increases with softer surfaces that conform better during contact, increasing the true surface area of the contact. These behaviors depend not only on the material of the contact surface but also on that of the objects that are being interacted with. Interrogating objects with different interfacial dynamics provides more information about the type of material interaction and can be used to better classify what is being contacted or to identify properties of those materials. In this work, microphones were embedded in the SMP

epoxy interface to collect feedback during material interactions. Simple convolutional neural networks were trained and used to classify the materials in different interaction instances to test the hypothesis that this tunability indeed provides a means of improving discriminatory capability.

5.3 Background and related works

This work builds on many recent advances in soft systems research to develop a novel multifunctional material device. This section first explores some existing work on tunable-friction interfaces, especially as it applies to robotics and manipulation. Then we turn to acoustic tactile sensing, where there have been exciting recent advances. Finally, we end with a discussion of the contributions of this work and the problems they are meant to address.

5.3.1 Contact adaptation with tunable-friction interfaces

On-demand contact adaptation is highly desirable in general-purpose robotics applications where it is desirable to use the same device for different purposes that require different capabilities in the end effector. One way of achieving this to some degree is to simply change end effectors for different tasks, but this is not always possible and not really desirable as it is largely a manual task that does not allow for adaptation to a potentially changing environment or changing objectives. Numerous works discuss static designs that achieve some level of friction modulation automatically by incorporating microscale features like veins to redirect surface fluids and contaminants to maintain surface contact in diverse environments or pillars and other protrusions to increase contact area on rougher surfaces. Thirunavukkarasu et al. designed such a system using biologically inspired hybrid structures for automatic modulation of friction [114]. In another study, Shin et al. reported a structure that automatically increased contact friction in wet conditions by wrinkling [102]. However, while the unique design and structure of the surfaces reported in both of these studies improve traction under certain conditions, they cannot be dynamically adjusted for the particular task at hand, which might instead benefit from *reduced* friction. An example of this is the work of Teeple et al., who use pneumatically actuated protrusions through a plastic interface on the palm of a robot hand to increase or decrease friction to improve object manipulation [112]. Controlling palm friction enabled a richer set of manipulation capabilities compared to a static interface, somewhat making up for the limited dexterity in current robot hands, as well as the typically low strength of soft manipulators. Depending on the chosen friction mode, the manipulator was able to roll or translate an object in its grip more easily. However, this

friction tuning is performed using pneumatics, implying significant size, weight, and power burdens on the hardware required to operate such a device. Similarly, Acharya et al. also demonstrated the utility of tunable contact interfaces for robot locomotion [1]. Pneumatic channels were used to selectively inflate chambers at the contact points of the robot limbs. This action forced the protrusion of a highly adhesion surface in contact with the ground, allowing the robot to maintain enough traction to climb steep inclines. However, the design suffered from the same issue, namely that the hardware required to operate the pneumatics greatly limits the ability to develop a system with a small form factor, untethered from bulky equipment and power supplies.

Electroadhesion provides an alternative approach to controllable friction, as discussed by Guo, Leng, and Rossiter [35]. Tuning the voltage potential across a contact pad pressed against another surface can induce some polarization and modulate friction even on nonconducting surfaces. The voltages required for electroadhesive friction can be quite large, on the order of a few kilovolts, which presents a danger both of rapid device degradation if there is a dielectric breakdown or damage to the device, as well as a potent safety risk if such a failure occurs when interacting with biological organisms. Chen and Bergbreiter demonstrated electroadhesive grippers on robot feet to enable contact tuning in a small form factor at just 200 V [18]. However, even this is still a fairly high voltage and can be dangerous for interaction if the device fails. Liefferink et al. control friction using a kirigami-designed metamaterial that, by straining the interface, can change the angle of surface asperities to increase or decrease the friction of the surface [62]. It is not clear how robust a kirigami metamaterial would be, especially in the presence of contaminants encountered during contacts that could get trapped in the cuts and folds, diminishing device performance. Fabrication of such devices and actuating the surface to modulate the friction could also pose challenges.

Sharifi et al. made a tunable-friction polydimethylsiloxane (PDMS) interface by filling embedded microchannels with a low-melting-point alloy [99]. Through joule heating, the rigid alloy in the channels is melted, reducing the rigidity of the device. Halting the current flow allows the metal to cool and harden again. This tunability allows an inchworm robot to climb up an incline by alternating the high-traction mode to anchor one end while the other end slides forward. For this purpose, the subsurface microchannels are very close to the contact interface (350 – 750 µm in this case). A robust and reliable system such as this can be difficult to deploy, especially under a thin, soft substrate like PDMS which can be easily damaged to open the channels and allow the liquid metal to escape. It may also be difficult to balance competing objectives of creating smaller channels to make higher resistance and more efficient joule heaters while avoiding easily collapsible channels when pressure is made during contact.

5.3.2 Acoustic tactile sensing

Roberts, Zadan, and Majidi review many of the strategies commonly used for soft tactile sensing in robotics [89]. Typical sensors operate primarily by transducing static pressures using resistive or capacitive strain gauges, field disturbances in a magnetic polymer, optical transmission intensity, and barometric sensing. These are very capable approaches for static and low-frequency features, but such approaches are not as effective for high-frequency sensing, especially when implemented as a soft system with significant hysteresis.

Using acoustic feedback for dynamic tactile sensing has become increasingly widespread in recent years. Lee et al. demonstrate the utility of vibrotactile feedback in contact localization for robotics [58], also hinting at the possible utility in better understanding and classifying such contacts to improve manipulation capabilities in field applications [57]. Rajendran et al. published a number of works [85, 86, 84] on their acoustic soft tactile (AST) sensors that pair a speaker with a microphone in soft and collapsible sound channels. Active acoustic excitation is used to continuously sense changes in wave propagation through the channel due to vibration, contact, or deformation, further demonstrating its utility in strawberry harvesting [87]. However, no texture sensing was reported for these sensors, as they were primarily used to enable transduction of force and force localization. Lu and Culbertson demonstrate object classification by placing a speaker and microphone on opposite ends of an object to analyze the propagation of acoustic waves through it [64]. They are able to achieve classification accuracy as high as 84 %. This approach probably would not generalize to more typical single-point contacts or sensing tasks that cannot grip completely around the entire object. Neither would it help to identify textures or contact forces.

However, active methods are not required for reliable tactile acoustic sensing. Svensson et al. conducted a user study to demonstrate that acoustic feedback from a condenser microphone rubbing against different surfaces and transducer as electrical stimulation could be fairly easily classified by humans [107]. Robots can do it too: Liu and Chen introduced SonicSense, which uses contact microphones on multiple grippers to enable material categorization and shape reconstruction [63]. Contact microphones were chosen for their limited sensitivity to ambient noise, although they can be quite bulky and are not ideal for smaller or denser sensor integrations. Pätzold et al. made process in passive acoustic detection of roughness, combining a contact microphone at the tactile interface with a nearby open-air MEMS microphone [83]. There is much room for improvement, as the system only attempted to make a coarse classification of whether the surface was rough or not. It may also be sufficient to use a single microphone for this purpose. Although results were shown for the configuration with only a piezoelectric contact microphone, an MEMS microphone at the contact interface, either

open to ambient air or encapsulated, may have achieved the same or better performance. Park et al. demonstrate texture classification, comparing conventional MEMS microphones to a hierarchical triboelectric sensor (TES) they designed [79]. The device performs very well on eight textures, reaching an accuracy of about 92 %. The textures used in this setup were line-patterned to enhance the dynamic response, so it is not clear from these results how well the classification would have proceeded on materials in the wild. Fabricating and incorporating TESs into a design is also more challenging than just using an off-the-shelf microphone. Chang et al. demonstrated tactile sensing using a MEMS microphone array embedded in PDMS and covered with a non-slip polyethylene tape [17]. They were able to demonstrate good performance in classifying contacts with four different textured surfaces with over 77 % accuracy using a transformer to featurize the raw audio. This process was not evaluated on different materials, and the devices require the use of a rigid tape that wears and requires replacement. Furthermore, a significant amount of data was needed to enable robust featurization and classifier training, 100 000 samples in this case. Characterizing such a sensor this way, especially for more complicated scenarios as in multiple texture and material classification, is probably untenable, although it is not clear how frequently the tape needs to be replaced from their reporting.

5.3.3 Contributions

In this work, I propose a design for a robust and easily fabricated tunable-friction contact device that can be assembled with cheap, off-the-shelf components and minimal supplies. With this design, I demonstrate a number of improvements over existing work.

- **Simple and robust tunable-friction interface**: A robust tunable-stiffness contact surface was developed using a tough shape memory epoxy with embedded sensing to allow dynamic tuning of the contact dynamics. This enables the device to transition between low- and high-friction gripping modes to adapt to the particular task. This was demonstrated using a simple unactuated finger attached to a robot arm that attempts to gently drag different materials. For the same applied normal force, the end effector was able to manipulate materials in the soft state that caused slippage in the hard state, enabling gentler manipulation of delicate objects.
- **Accurate material classification with minimal training data**: Sensor data was collected by rubbing the device with eight different surfaces. An off-the-shelf pretrained audio transformer was used to featurize raw data from contact interactions to accurately predict the target material with no fine-tuning and minimal training data. Models achieve baseline performance over 90 % classification accuracy across eight different

materials and a range of different rubbing speeds and contact forces.

- **Multi-stiffness measurement fusion for improved performance**: I demonstrate the novel use of stiffness tuning on the contact interface to improve classification inference. Pooling hard- and soft-state interactions achieves better classification performance compared with pooling samples collected using a single stiffness.

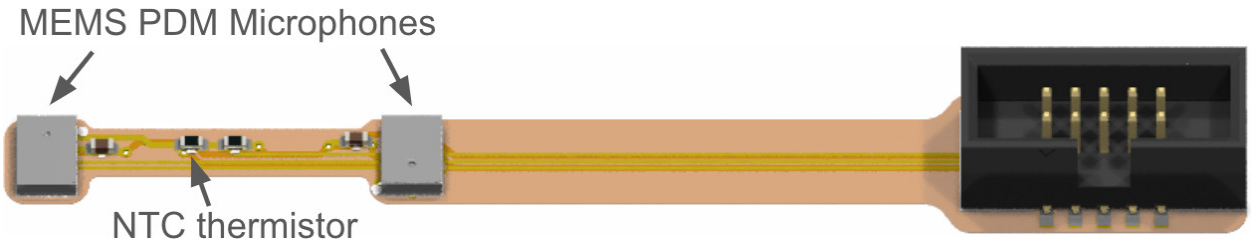
In this work, I present an integrated soft-material sensing system that uses a tunable-stiffness contact interface to improve tactile sensing and enable contact modulation on the same device. In the following sections, I outline the design and fabrication of two different realizations of the same sensor design and demonstrate their utility in sensing and manipulation.

5.4 Tunable-stiffness sensor pad

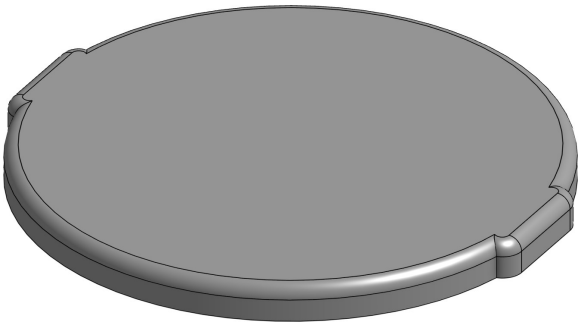
A flat, stiffness-tunable tactile sensor pad was developed, and its ability to classify the materials rubbed against it was evaluated. This design can be useful in applications including parallel grippers or mostly flat contact points, such as semi-rigid robot feet. Here, I will mainly focus on the sensing aspect of the device, saving a demonstration of the tunable contact dynamics for the following section.

5.4.1 Device design

The main components of the device design are the embedded sensors (PCB in figure 5.2a), the tunable stiffness interface (see an untextured and textured pad in figures 5.2b and 5.2c), and a heating element. On a flexible PCB, two MEMS pulse density modulation (PDM) microphones (ICS-41351, TDK InvenSense) are used at the edges of the contact surface. This was originally designed to enable contact localization, although this feature was not explored in this work. As demonstrated in section 5.5, the design works with a single microphone as well. In addition to the microphones, a temperature sensor was used to control the heating of the device and maintain a consistent surface temperature. A 10 kΩ negative temperature coefficient (NTC) thermistor was used in a voltage divider with a precise 10 kΩ resistor. A thermoelectric device (TED) was used to heat the surface (TES1-4902, generic). The use of a TED in the design allows active cooling to improve the response time of the device, but that feature will not be explored further in this work.



(a) Flex PCB with sensors and connector.



(b) Untextured SMP pad CAD rendering.

(c) Textured SMP pad CAD rendering.

Figure 5.2: Tunable-friction pad device components. (a) Flexible PCB populated with two microphones and a thermistor for sensing and temperature control. (b) CAD rendering of an untextured tunable-stiffness SMP pad. (c) CAD rendering of a textured tunable-stiffness SMP pad.

Calibration-free temperature estimation

The thermistor resistance was determined by measuring the voltage in the middle of the voltage divider, then the temperature was estimated from the resistance using an approach based on the Steinhart-Hart equation for thermistor calibration [106]:

$$\frac{1}{T} = A + B\ln(R) + C(\ln(R))^3. \tag{5.1}$$

Thermistor manufacturers typically provide resistance at one temperature, usually at $T = 25\,^{\circ}\mathrm{C}$, along with *beta* parameters for two or more other temperatures in the device's operating range collected during device characterization. The beta parameter is defined by dropping the cubic term from equation 5.1 and instead assuming a linear relationship. Using beta parameters for temperature estimation is less accurate than the full Steinhart-Hart model, but the values of C tend to be quite small anyway, on the order of 100×10^{-9} compared to A and B on the order of 1×10^{-3}, so the penalty is not excessive. Furthermore, the beta parameters are more readily available, whereas the Steinhart-Hart parameters are not usually provided. Next, we take the difference of the linear part of equation 5.1 at two calibration points, say $T_{\mathrm{ref}} = 25\,^{\circ}\mathrm{C}$ and $T_T = T^{\circ}\mathrm{C}$ with resistances R_{ref} and R_T, respectively:

$$\begin{aligned}
\frac{1}{T_T} - \frac{1}{T_{\mathrm{ref}}} &= A + B\ln(R_T) - A - \ln(R_{\mathrm{ref}}) \\
&= B\ln\left(\frac{R_T}{R_{\mathrm{ref}}}\right) \\
\frac{1}{B} &= \ln\left(\frac{R_T}{R_{\mathrm{ref}}}\right)\left(\frac{1}{T_T} - \frac{1}{T_{\mathrm{ref}}}\right)^{-1}.
\end{aligned}$$

Letting $\beta = 1/B$, we have

$$\beta_T = \ln\left(\frac{R_T}{R_{\mathrm{ref}}}\right)\left(\frac{1}{T_T} - \frac{1}{T_{\mathrm{ref}}}\right)^{-1}. \tag{5.2}$$

β values will be defined for some number of calibration temperatures. The resistances at those temperatures are not always defined but can be derived by rearranging equation 5.2:

$$R_T = R_{\mathrm{ref}}\exp\left(\beta_T\left(\frac{1}{T_T} - \frac{1}{T_{\mathrm{ref}}}\right)\right). \tag{5.3}$$

Using this information, we can determine an *effective* value for β at a particular temperature by interpolating between the given calibration points T_{low} and T_{high}:

$$w = \frac{\ln(R) - \ln(R_{T_{\text{low}}})}{\ln(R_{T_{\text{high}}}) - \ln(R_{T_{\text{low}}})}. \tag{5.4}$$

w is clamped between zero and one, and the value of β is

$$\beta = w\beta_{T_{\text{high}}} + (1 - w)\beta_{T_{\text{low}}}. \tag{5.5}$$

Equation 5.3 is then rearranged to determine the temperature from a given resistance measurement R:

$$T = \frac{1}{\beta} \ln\left(\frac{R}{R_{\text{ref}}}\right) + \frac{1}{T_{\text{ref}}}. \tag{5.6}$$

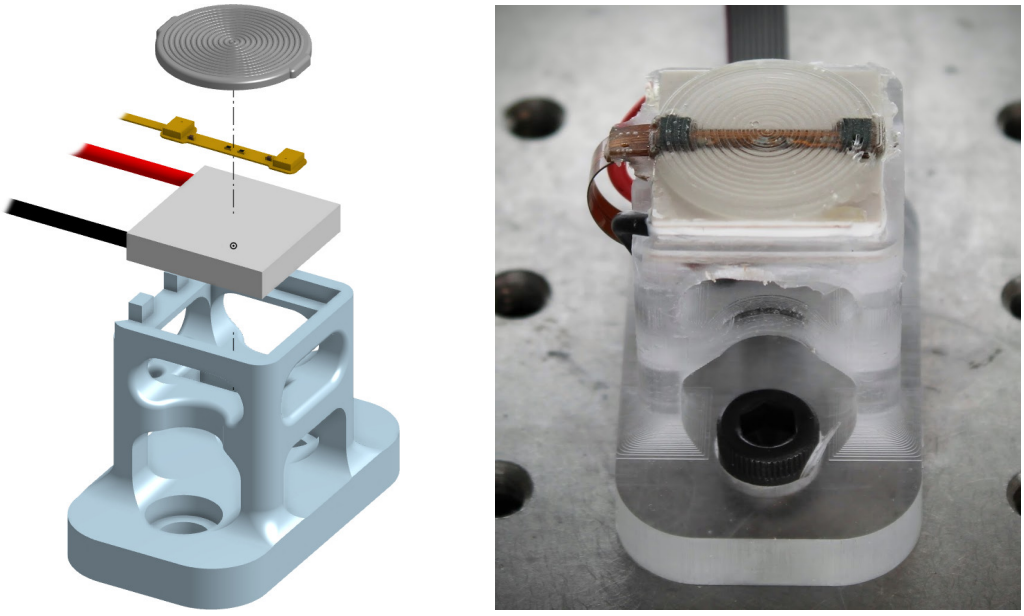
This yielded accurate temperature measurements without the need to collect calibration data manually. Although there will be device-to-device variation, selection of components with tight tolerances and feeding the voltage divider with a stable voltage (or, alternatively, measuring both the bus voltage along with the divider voltage) will yield accurate results. The measured variation did not significantly affect the operation of these devices, as it was typically below 1 °C.

5.4.2 Device fabrication

The flat sensor stackup, shown in figure 5.3a, consists of a thin (roughly 1.4 mm thick) shape memory polymer (SMP) substrate cured on top of a 20 mm × 20 mm TED with two microphones and a temperature sensor embedded inside. This device was bonded to a 3D-printed fixture and mounted to an optical table next to a robot arm and the control electronics, both of which are described later. A photograph of the assembled and mounted device is shown in figure 5.3b.

To form the SMP substrate, a negative mold was made from silicone first, as the epoxy resin bonds to most other materials. Commercial silicone (Smooth-Sil 950, Smooth-On) was mixed and deaerated for two minutes each cycle in a planetary centrifugal mixer (AR-100, Thinky). The mixed resin was added in small quantities to a 3D-printed negative mold and left under a vacuum for ten minutes between pours, three in total. For textured molds in particular, this greatly reduces the incidence of bubbles forming on the fine features of the mold, as gas trapped during pouring is otherwise unable to escape through the viscous resin. Once filled, the mold is left to cure overnight at room temperature.

The sensor PCB was assembled and then bonded to the thermoelectric device using a thin layer of thermally-conductive epoxy (8329TFF, MG Chemicals). The epoxy was left to cure for at least two hours before casting in the SMP. The PCB assembly and the silicone



(a) CAD assembly. (b) Photo of assembled device.

Figure 5.3: Pad sensor design with 20 mm diameter SMP pad, textured with concentric circles. (a) CAD rendering of the assembly. (b) Photo of the assembled device.

mold were placed in an oven at 60 °C for ten minutes to prepare them for casting.

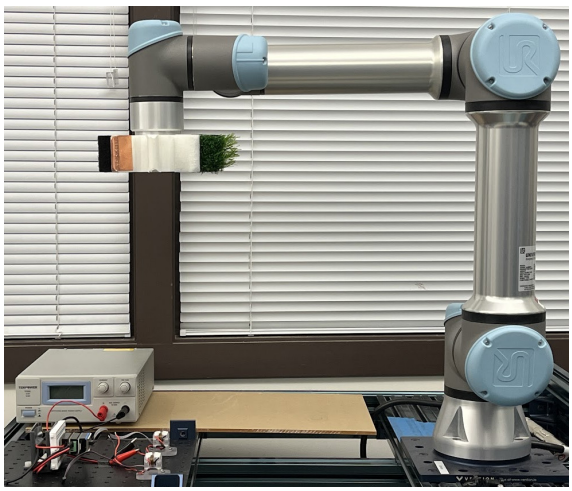
To form the SMP epoxy, a commercial epoxy resin (EPON-828, Hexion) was combined with a hardener (Jeffamine D400, Huntsman) in a ratio of 10:4 by weight (below the stoichiometric ratio of about 10:5.6 [52]) with some modifications from the process described by Buckner et al. in [14]. The resulting shape memory polymer has a transition temperature of about 44 °C and increases its modulus about 270 times from 9.25 MPa in the soft state to 2.45 GPa in the hard state, per the analysis of Rousseau and Xie [91]. A slightly larger hardener ratio was used compared with their 10:3.6 ratio, so it is probably slightly harder. Before mixing, the resin and hardener are heated in an oven at 60 °C for at least 20 minutes. The reduced viscosity of the resin is important to avoid bubble formation in the fine surface features. The chemicals were then combined in a vial and shear mixed with a plastic paddle slowly for about 30 seconds, then mixed in a Thinky for 2 minutes and a deareted for 5 minutes. The mixture was poured slowly into the mold. The filled mold was carefully rotated and stretched to ensure the epoxy mixtures was able to fill in all voids and allow any trapped air to escape, further reducing the formation of surface bubbles. A thin layer of epoxy was poured over the electronic components in the PCB assembly to ensure coverage of the microphone port holes and to discourage bubble formation when the unit is embedded in the mold. This should happen after the mixture has cooled and gelled for about 20 minutes or so, as the less viscous mixture might otherwise flow into the portholes of the microphone and prevent

acoustic transduction in the hardened state. The assembly was then gently pressed into the SMP-filled mold to allow the SMP surface to cast on top of the TED with the sensors embedded within. A 200 g weight was placed on top and the device was left to cure at room temperature overnight. The final cure took place in an oven at 80 °C for two hours. After removal from the oven, the device was superglued to the optical table fixture and then demolded.

5.4.3 Data collection

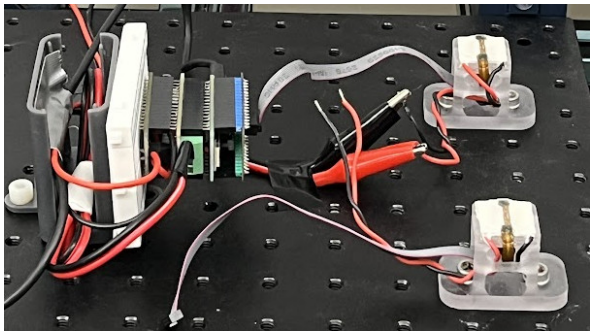
The assembled test fixture was mounted on an optical table next to a robot arm (UR5e, Universal Robotics) and collection hardware (shown in figure 5.4a). The arm was fitted with a 3D-printed thermoplastic polyurethane (TPU) end effector with eight contact surfaces, each covered with a different material to rub against the sensor (see figure 5.4b). Thin VHB tape (3M) was used to fix each material to the end effector paddle. The design allows the arm to switch between textures during data collection by simply rotating the arm's wrist joint. The material was rubbed against the sensor (figure 5.4c) at varying speeds (10 mm/s, 20 mm/s, and 40 mm/s) and normal forces (1 N, 2 N, and 4 N). The initial normal force was tuned using a force-torque sensor on the wrist. The end effector was slowly lowered toward the device until a force spike was seen, indicating contact was made. The position of the end effector was slowly adjusted until the desired normal force was achieved within 5 %. Ten sweeps were performed for each configuration, rubbing the material 10 mm in one direction, then returning 10 mm back to the starting position. The experimental procedure was controlled and sensor data was recorded and visualized using a purpose-built real-time interface (figure 5.4d). After collecting all data for a particular texture, the end effector was lifted, rotated to the next texture, and the process was repeated until the collection was complete for all eight textures. The textures used were burlap, canvas, copper foil, styrofoam, TPU, artificial turf, Velcro fuzz, and wax paper. These materials present a range of different textures, friction, and adhesive properties.

Sensor data was collected using a commercial off-the-shelf (COTS) microcontroller development board (Feather nRF52840 Express, Adafruit). The device firmware was written with the manufacturer's supplied software development kit (Nordic OpenConnect SDK, Nordic Semiconductor) to directly target the on-board microcontroller (nRF52840, Nordic Semiconductor) to make full use of the pulse-density modulated (PDM) microphone peripheral and direct memory access (DMA) capability for fast buffering of sensor data. Acoustic feedback was continuously captured from each microphone in the stereo pair at 16 kHz and telemetered over USB.

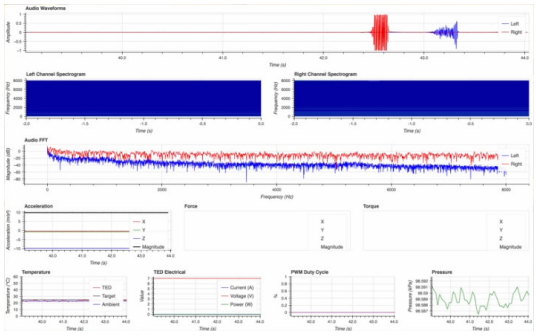


(a) Full test configuration.

(b) Eight-texture end effector.



(c) Closeup of sensor contact.



(d) Real-time interface.

Figure 5.4: Pad sensor material classification test setup. (a) The hardware configuration with the device, power supply, and robot arm for sensor excitation. (b) An eight-texture end effector used to select. (c) A microcontroller development board for sensor communication, along with two pad sensors connected to the optical table. (d) The real-time data logging and visualization interface.

The surface temperature was maintained at 30 °C in the hard state and 50 °C in the soft state. The embedded thermistor was used to estimate the instantaneous substrate temperature as described in section 5.4.1. Temperature control u at time t was achieved with a proportional-derivative (PD) controller,

$$u(t) = K_p e(t) + K_d \frac{de(t)}{dt}, \tag{5.7}$$

with the temperature set-point error $e(t) = T(t) - T_{set}$, its derivative $de(t)/dt$, and the proportional and derivative control gains K_p and K_d, respectively. The control output $u(t)$ controls the duty cycle for the pulse-width modulation (PWM) of the power through the thermoelectric, along with the direction of current flow. Directionality and PWM throttling is achieved with a full H-bridge motor controller (TB6612, Toshiba) through a development shield (DC Motor FeatherWing, Adafruit). For joule heating only, or if Peltier operation is not needed, the control signal is clamped to $u(t) \in [0, 1]$. Otherwise, it is clamped to $u(t) \in [-1, 1]$. A high-side current monitor (INA219, Texas Instruments) is used via a breakout board (INA219 FeatherWing, Adafruit) to ensure that the current through the thermoelectric remains under a safe limit to avoid failure of the Peltier device.

5.4.4 Data evaluation

The main objective of the data evaluation was to evaluate the utility of the collected data for the purpose of material classification and to compare the classification performance using both interface modes (hard and soft) versus only a single mode (hard *or* soft). A minimalist pipeline was used for data processing, model training, and evaluation to keep the process simple and identify a path to implement the full pipeline in an embedded environment, although the real-time embedded operation was not demonstrated here. Each step of the processing and evaluation is described in the sections below.

Audio data preprocessing and featurization

Individual sweep start and end times were recorded and those delineations were used to extract audio data from each individual sweep along with its collection parameters (sensor ID, hard/soft state, texture, normal force, and sweep speed). Only audio feedback was used for model training, the force-torque data was used only for experimental control and debugging. Temperature feedback was only used to control the surface temperature, and thus the stiffness, of the sensor interface. Since individual sample instances were data collected for a single sweep, slower sweeps resulted in larger data vectors than sweeps performed at a faster speed. To reduce the dependence of model training on speed, the waveforms of the faster sweeps

were up-sampled to match the vector length of the slowest sweep using polyphase filtering described by Crochiere and Rabiner in [24]. Whether resampled or not, the instances were not exactly the same length due to slight timing misalignments, so they were all trimmed to a uniform length of 32 000 samples.

The raw audio was featurized using a pre-trained wav2vec 2.0 model published by Baevski et al. in [6]. The model was trained on 960 hours of unlabeled audio from the librispreech corpus from Panayotov et al. [78]. While featurizing based on speech audio is not ideal for the completely different purpose of tactile recognition, there unfortunately does not yet exist any extensive audio corpus for surface interactions. However, as will be demonstrated in the next section, this featurization still provided a solid foundation on which to train an acoustic texture recognition model with very little data.

Classifier architecture and training

For material classification, we used a simple multilayer convolutional neural network architecture (for CNNs see the work of LeCun et al. [55]). The featurized dataset was fed into two CNN layers and three fully connected linear layers. Max-pooling was used at the output of both convolutional layers, and all layers used a ReLu activation function (Nair and Hinton [75]). A diagram of the architecture is shown in figure 5.5. Softmax was used to normalize the final output layer to form a set of uncalibrated class predictions. This procedure was repeated multiple times in a five-fold cross-validation framework. Separate models were trained for hard contacts, soft contacts, and a combination.

Classifier performance evaluation

The hard and soft texture classifiers worked quite well on their own, with classification accuracies of 89.4 % and 88.9 % for the hard and soft models using an untextured sensor despite very little training data and unrelated speech audio for training the autoencoder used to featurize the data. The textured sensor did not perform as well in the soft state, with hard and soft classification accuracies of 89 % and 80 %, respectively. This performance gap agrees with our prior intuition that a firmer interface should generally perform better at texture classification, since a softer material is able to deform more, yielding a less consistent acoustic signature. Furthermore, the greater adhesion of the soft substrate is more likely to exhibit larger and more frequent stick-slip events, which adds to the difficulty. It is interesting that the untextured unit did not see a similar degradation in performance on the soft model, perhaps indicating that the additional surface area contact yields more unique behavior across materials or just transduces the signal better. That said, the original hypothesis was that

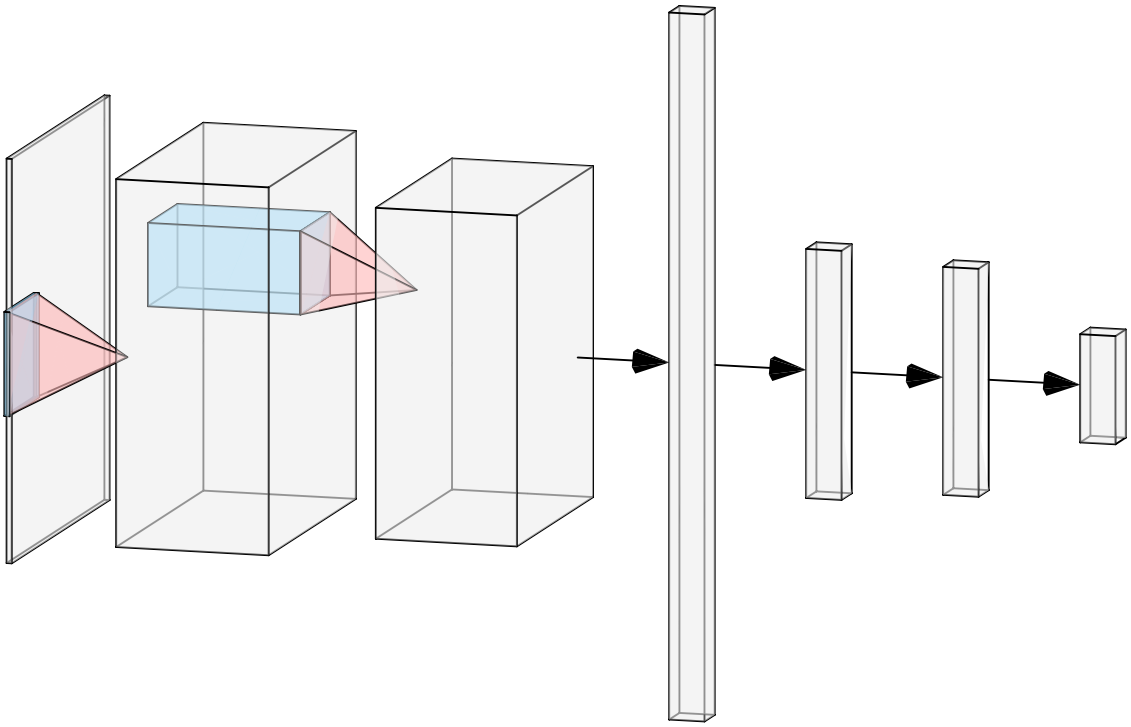


Figure 5.5: Convolutional neural network (CNN) architecture. ReLU is used between layers, and a max-pooling operation is performed between tensors. Class predictions are defined as the softmax of the output layer. Image generated with NN-SVG from LeNail [59]

there would be some complementary behavior, namely that the hard interface would be able to classify some materials better and some materials worse than the soft interface, even if overall performance is better. Indeed, this is what we see in the confusion matrices shown in figures 5.6 and 5.7.

Although the hard interface is generally better overall, there are some differences between classes. On the untextured sensor, the performance on canvas and wax paper is actually better in the soft sensor mode. Similarly, on the textured sensor, the soft mode achieves considerably better detection performance on TPU classification. To get a better sense of the performance difference between the two, we can look at an advantage matrix, shown on the left side of figure 5.8 for the untextured sensor and figure 5.9 for the textured variant. This matrix presents the advantage one classifier has over the other on correct classifications as well as misclassifications. Cells shaded pink show better performance on the soft classifier, whereas those shaded green indicate better performance on the hard sensor model. On the diagonal, "better" implies that the model correctly classified that class *more* often. In off-diagonal cells, "better" implies that the model made that particular misclassification error *less* often. This confusion matrix better highlights the performance differences between the models. It also highlights that the specific per-class misclassifications can be quite different

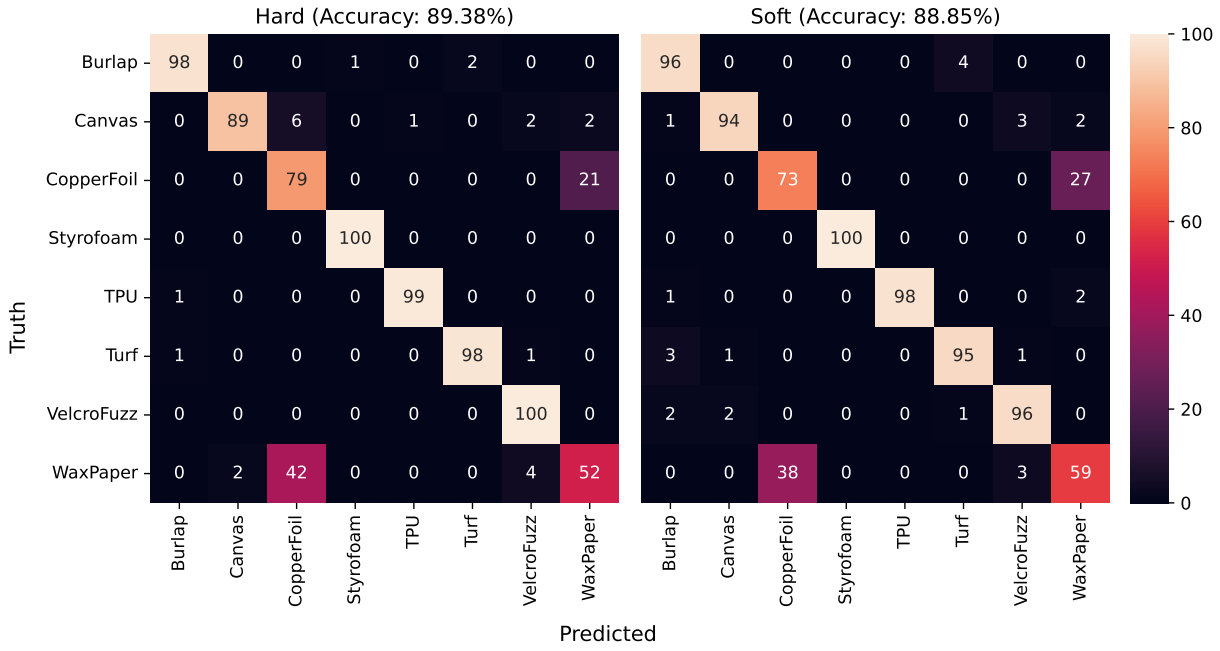


Figure 5.6: **Untextured device** five-fold cross-validated single-stiffness material classification confusion matrices for sensing in the hard (left) and soft (right) states. Values are indicated in rounded percentages.

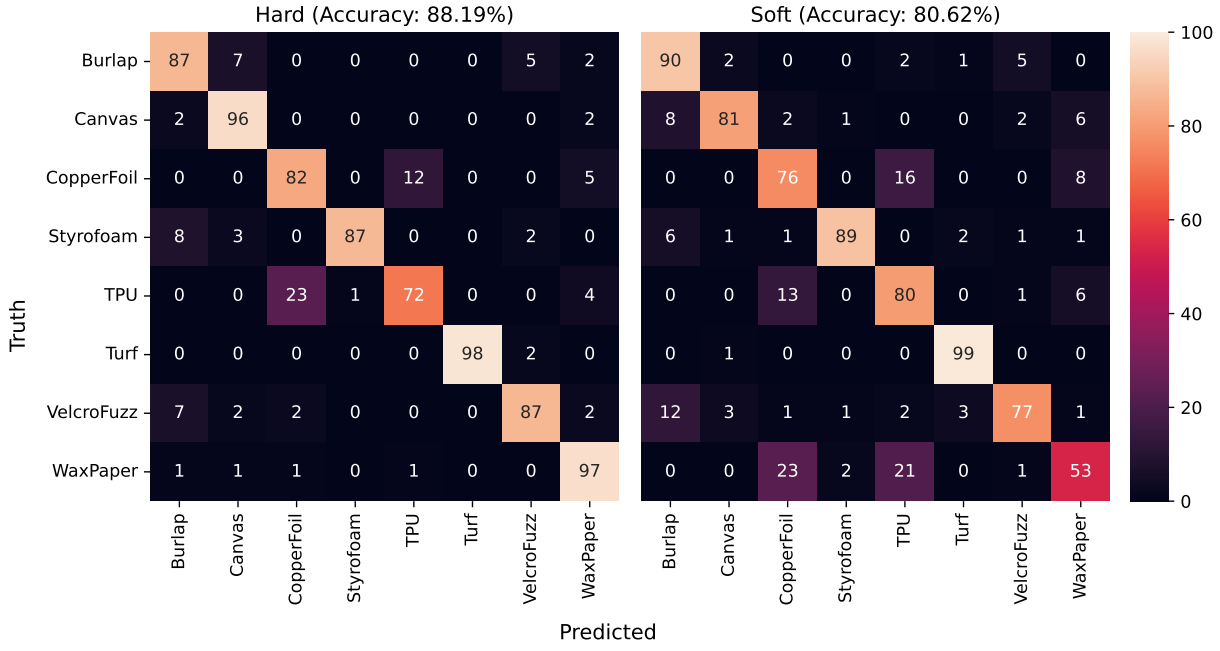


Figure 5.7: **Textured device** five-fold cross-validated single-sensor material classification confusion matrices for sensing in the hard (left) and soft (right) states. Values are indicated in rounded percentages.

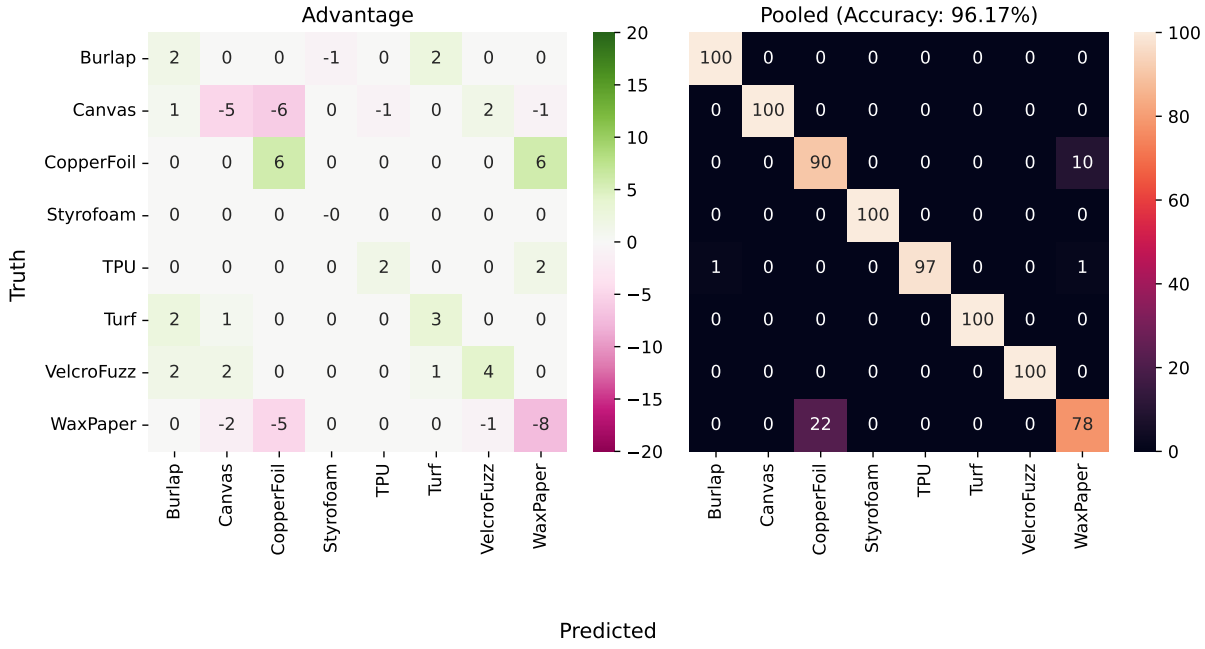


Figure 5.8: **Untextured sensor** (left) advantage matrix comparing the hard and soft confusion matrices and (right) confusion matrix of the pooled model.

between the two contact modes.

Given the performance characteristics of the two models, it seemed promising that the combination of the two sensor modes should improve the classification performance. The hard and soft sample instances were randomly paired. Naïvely training a new CNN on these combined samples did not yield much better performance. The model performed slightly below the hard model, as it seemed to propogate not just the strengths but also the weaknesses of both models instead of attempting to combine just their strengths. However, if the original predictions are simply pooled, the performance improves considerably. Both linear and logarithmic opinion pooling were used, as described by Genest and Zidek in [31]. We treated the softmax predictions as if they were probabilities, although we did not attempt to calibrate them to form an actual probability distribution (doing so may improve performance further). Linear opinion pooling simply computes a weighted sum of the probability vectors, i.e., for $N = 2$ sensor modes, a set of weights $w_1, \ldots, w_n$ such that $w_i \in [0, 1]$ and $\sum w_i = 1$, the pooled probability vector is defined as

$$P_{\text{linear}} = \sum_{i=1}^{n} w_i P_i \tag{5.8}$$

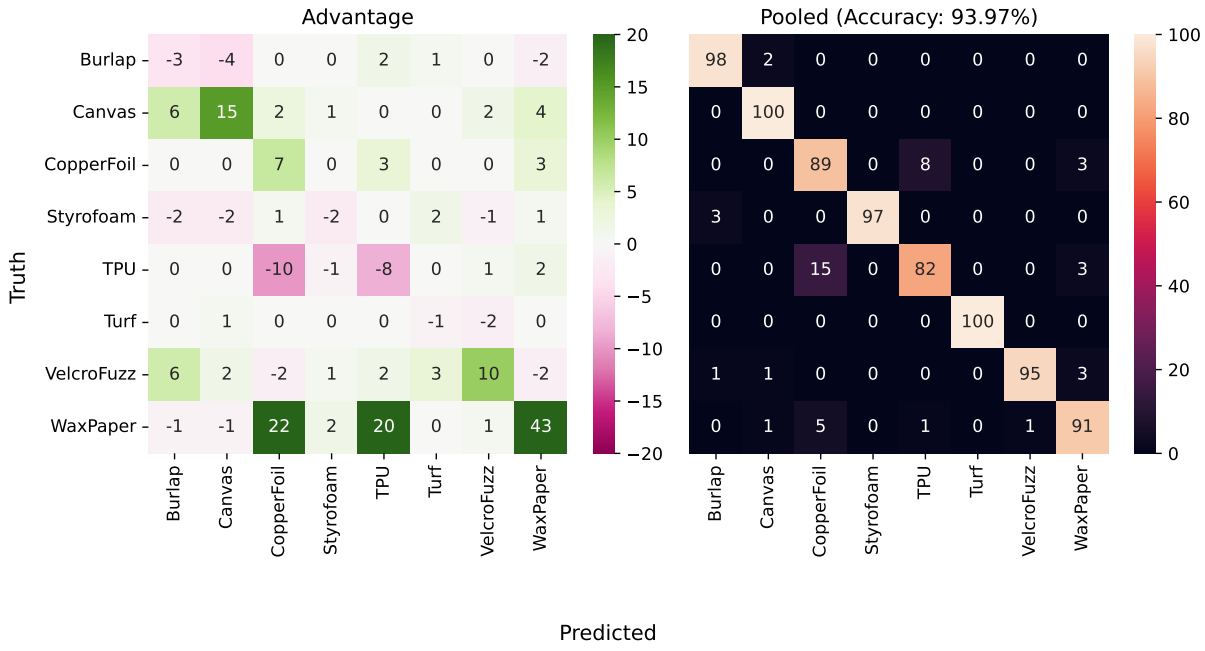


Figure 5.9: **Textured sensor** (left) advantage matrix comparing the hard and soft confusion matrices and (right) confusion matrix of the pooled model.

where P_i is the probability vector of the ith model and P_{linear} in the pooled probability vector. For the two-class problem, this is just

$$P_{\text{linear}} = wP_1 + (1 - w)P_2. \tag{5.9}$$

Based on this, the procedure scales well for more than two sensing modes, for example, if there were three or more sensing surfaces with different stiffnesses. Logarithmic opinion pooling works similarly but instead combines the probabilities such that

$$P_{\log} \propto \prod_{i=1}^{n} P_i^{w_i}. \tag{5.10}$$

To form a proper probability distribution, this value can be normalized by the sum of these factors, but since arg max is used to determine the class label, normalization is not needed. The "logarithmic" label is justified in the more typical application of the pooling, which instead looks at the log probability:

$$\log(P_{\log}) \propto \sum_{i=1}^{n} w_i \log(P_i). \tag{5.11}$$

$$\sum_{i=1}^{n} w_i \log(P_i) \propto \log(P_{\log}) \tag{5.12}$$

For the two-class case, this is

$$\log(P_{\log}) \propto w \log(P_1) + (1 - w) \log(P_2). \tag{5.13}$$

Both pooling methods greatly improve the accuracy of the combined classification. Logarithmic pooling of hard and soft contacts boosts the performance of the untextured sensor to 96.17 %, compared to 89.38 % hard or 88.85 % soft. The confusion matrix for the logarithmic pooled classifier is shown on the right side of figure 5.8. Likewise, on the textured model, the combination of contact modes increases classification performance to 93.97 %, compared to the original models that achieve 88 % hard or 80 % soft. The confusion matrix for the logarithmic pooled classifier is shown on the right side of figure 5.9.

From the pooled models we see that the performance improvement is across all classes, some quite significantly, as the model leverages the strengths of the two constituent models. Interestingly, it also greatly outperforms either in a few classes. For example, wax paper classification performance on the untextured sensor was below 60 % for both the hard and soft models However, when pooled, the model achieves 78 % accuracy on that class, substantially better than either individual model. It is likely the case that both models give some density to the correct class but distribute their misclassification density differently, so after pooling the correct classification gets boosted significantly relative to the now downweighted misclassifications. This effect is not seen if multiple samples using the same sensor stiffness are used. Figures 5.10 and 5.11 show confusion matrices for the untextured and textured sensors, respectively, when two observations are made and pooled in the same way. The classification performance improves with multiple samples as we would expect, but the gain is not nearly as significant.

The previous results generated using sensors with openings on the interface to expose the microphone's port hole to air. This configuration is not ideal, as the microphone can pick up ambient noise from a great distance because of its high sensitivity, and the exposed opening makes the device less robust to contamination that can enter through the opening and cause the microphone to malfunction. Additional sensors were fabricated with the microphone completely sealed off inside the interface. Figure 5.12 shows the performance of the textured sensor in this configuration. Compared to open-air sensors, classification performance drops to 83.89 % with the hard interface and 73.26 % with the soft interface. The closed port hole likely attenuates important acoustic signals that do not propagate as well through the epoxy. Performance might improve by instead fabricating an open-air sensor and then backfilling the

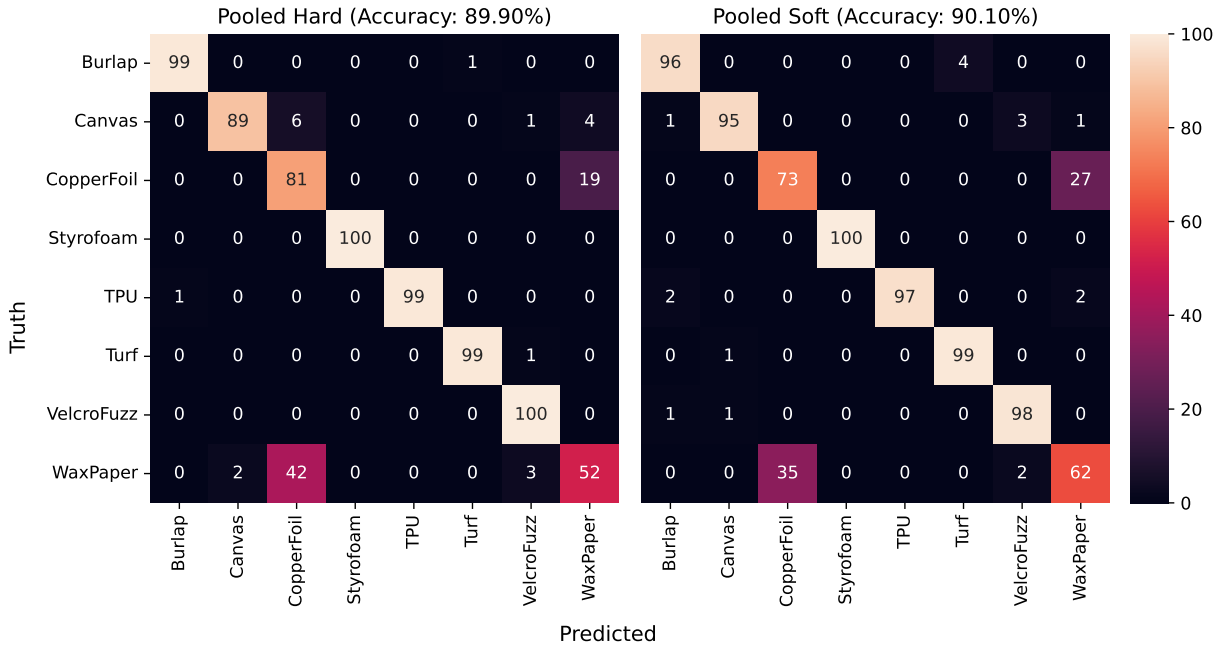


Figure 5.10: **Untextured device** five-fold cross-validated two-sample, single-stiffness material classification confusion matrices for sensing in the hard (left) and soft (right) states. Values are indicated in rounded percentages.

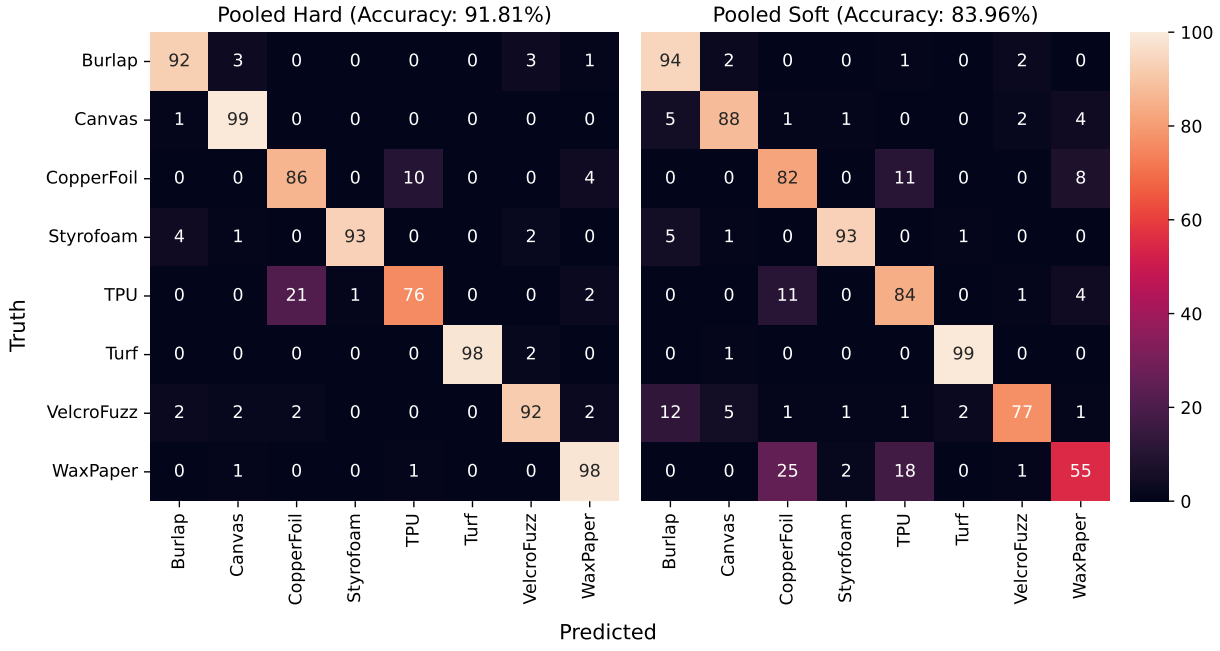


Figure 5.11: **Textured device** five-fold cross-validated two-sample, single-stiffness material classification confusion matrices for sensing in the hard (left) and soft (right) states. Values are indicated in rounded percentages.

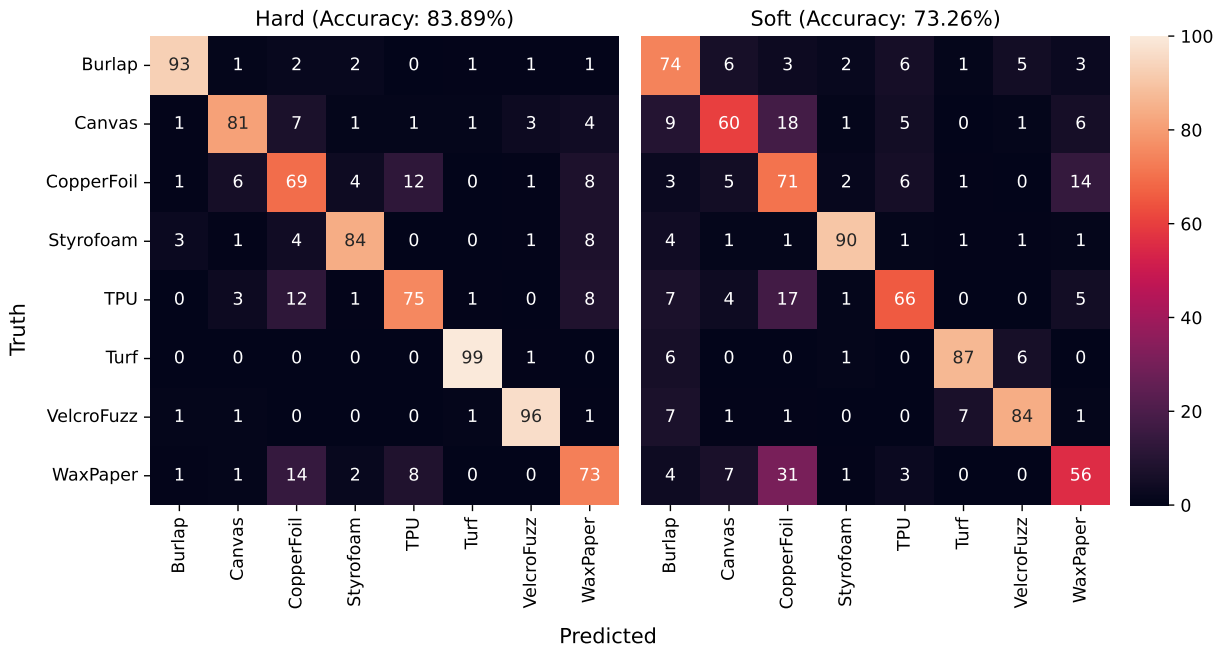


Figure 5.12: **Sealed microphone, textured device** five-fold cross-validated two-sample, single-stiffness material classification confusion matrices for sensing in the hard (left) and soft (right) states. Values are indicated in rounded percentages.

port hole with a soft silicone that can transduce the vibrations better than the epoxy can.

However, pooling the predictions from the interactions of the hard and soft sensors produces a considerable increase in accuracy to over 90 %. Figure 5.13 shows the advantage matrix (left) and the pooled confusion matrix (right) for this model. Although overall the performance of the hard model is almost always better, it is noteworthy that there seems to be a much greater variance in the distribution of errors compared to the open-air models. This may suggest that forcing the acoustic waves to propagate through the epoxy, which changes considerably between the two sensing modes, alters the received waveforms much more distinctly compared to the open-air variants.

The pooled two-sample models also improve, but again they do not achieve the same performance. Figure 5.14 shows the confusion matrices for the pooled, two-sample hard and soft models using the sealed sensor.

Taken together, the main trends seen are: better (sometimes considerably) performance of the hard interface classification performance compared with the soft; improvement on taking multiple samples; and even greater improvement when those two samples are evaluated with different contact stiffnesses. This is summarized in the chart in figure 5.15.

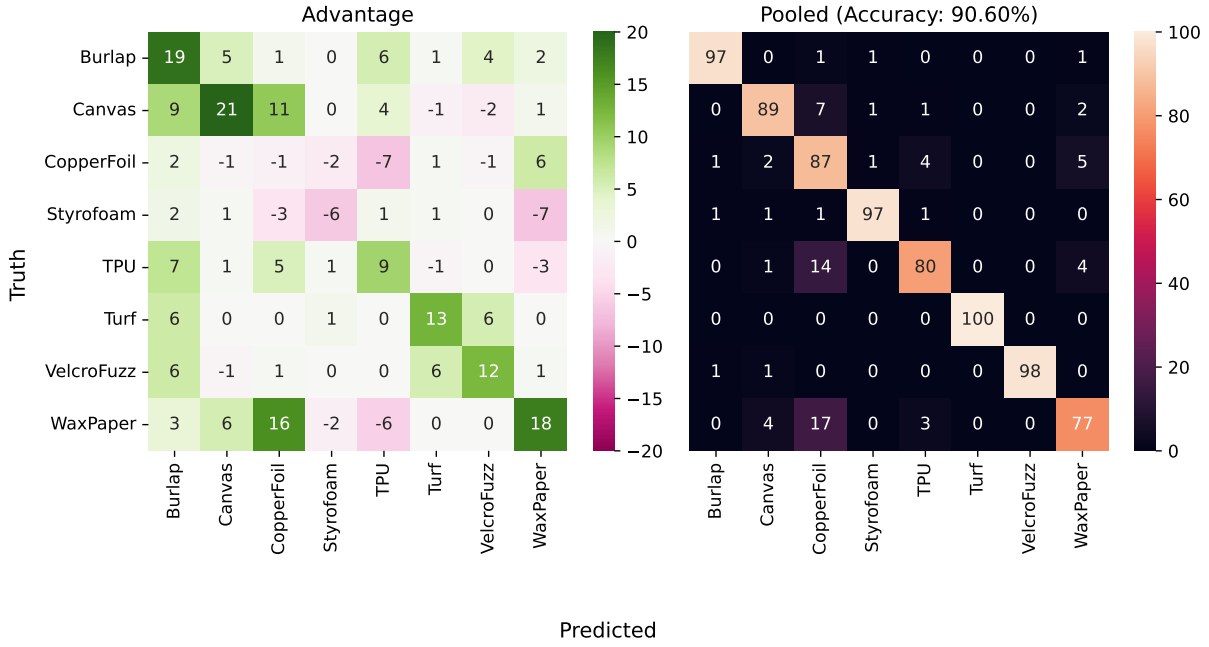


Figure 5.13: **Sealed microphone, textured device** (left) advantage matrix comparing the hard and soft confusion matrices and (right) confusion matrix of the pooled model.

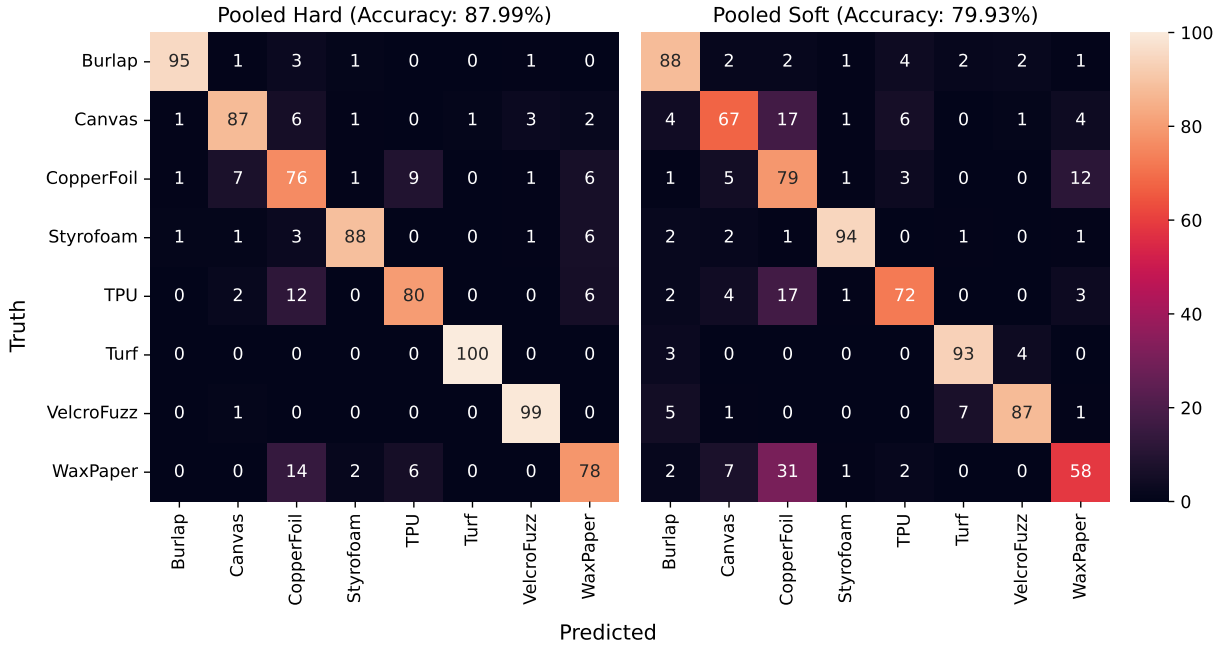


Figure 5.14: **Sealed microphone, textured device** five-fold cross-validated two-sample, single-stiffness material classification confusion matrices for sensing in the hard (left) and soft (right) states. Values are indicated in rounded percentages.

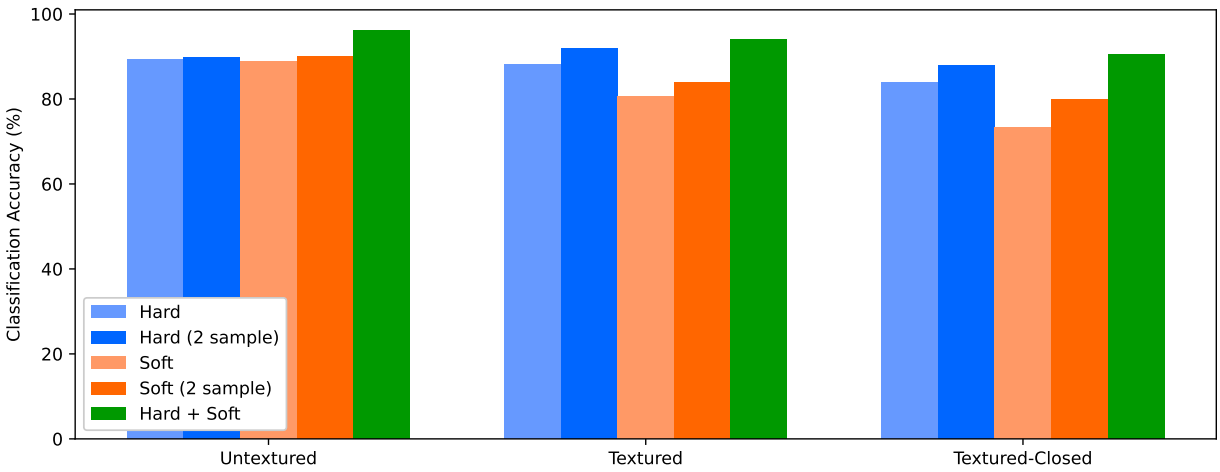


Figure 5.15: Comparison of total classification accuracy for each model and sensor configuration.

5.5 Tunable-stiffness finger

An unactuated dome-shaped finger device was fabricated and attached to the robot arm. Most of the details of the fabrication are the same as in the fabrication of the pad devices discussed in section 5.4 so we will skip repetitive details. This section focuses on changes to the design and experimentation for use as a tunable stiffness finger for contact tuning.

5.5.1 Tunable-stiffness finger device design

The main design elements of the finger sensor were the same, but the form factor was changed to produce a finger-shaped appendage at the end of the robot arm, with a dome-shaped stiffness-tunable contact surface for contact tuning and sensing instead of the flat surface used previously. Because the interface is molded, there is some flexibility in customizing the form factor. A single microphone was used in the center of the dome along with a thermistor, as before. The enclosure was 3D-printed in two parts: the upper piece that provides the structure for the sensor, the heater, and the interface; and the bottom piece holding the connector. This forms the main device assembly that can then be incorporated into a robot end effector. Photos of the assembled device with a closeup of the tip texture are shown in figure 5.16.

Although a textured surface was not explicitly added to this design, as was the case for pads, the use of a higher layer height of 100 µm adds sufficient texture for sensing. Unlike the pad sensors, a thermoelectric device is not used to heat the substrate. Instead, a flexible copper-clad FR4 substrate was patterned with an ultraviolet laser (ProtoLaser U4, LPKF) in

(a) Finger assembly. (b) Texture of the fingertip.

Figure 5.16: Stiffness-tunable finger device. (a) Assembled finger device with embedded sensing and joule heater. (b) Tip texture is defined by 3D-printed layer height.

a spiral design with cutouts to form a dome-shaped joule heater. The heater's CAD model and a photo are shown in figure 5.17.

To attach the sensor to the robot arm, a finger-shaped fixture was designed and 3D printed, shown in figure 5.18. The finger fixture is unactuated, instead relying on the motion of the arm to actively *feel* and interact with a surface. The body of the finger incorporates several interlocking helical structures to stabilize the structure, allowing for limited bending while preventing axial torsion, inspired by upcoming work of Plottel and Desatnik. The fixture was printed with flexible resin (Flexible 80A, Formlabs). The diameter of the opening is slightly less than that of the sensor, so insertion of the sensor was sufficient to hold it tightly in place for the studies.

5.5.2 Data collection

The assembled fixture was mounted to the end of the robot arm, as seen in figure 5.19a. In this setup, sensor data was collected and the interface stiffness was controlled using the same hardware as in the case of the pads. Then, dragging experiments were performed with a few different materials. A canvas substrate was affixed to a flat surface to provide some friction against dragging. A strip of no-slip tape was added across the center to introduce a high-friction region of the workspace. Three different materials were dragged from one side of the platform to the other in the hard and soft interface modes with the same normal

(a) Jouler heater CAD. (b) Stretched dome.

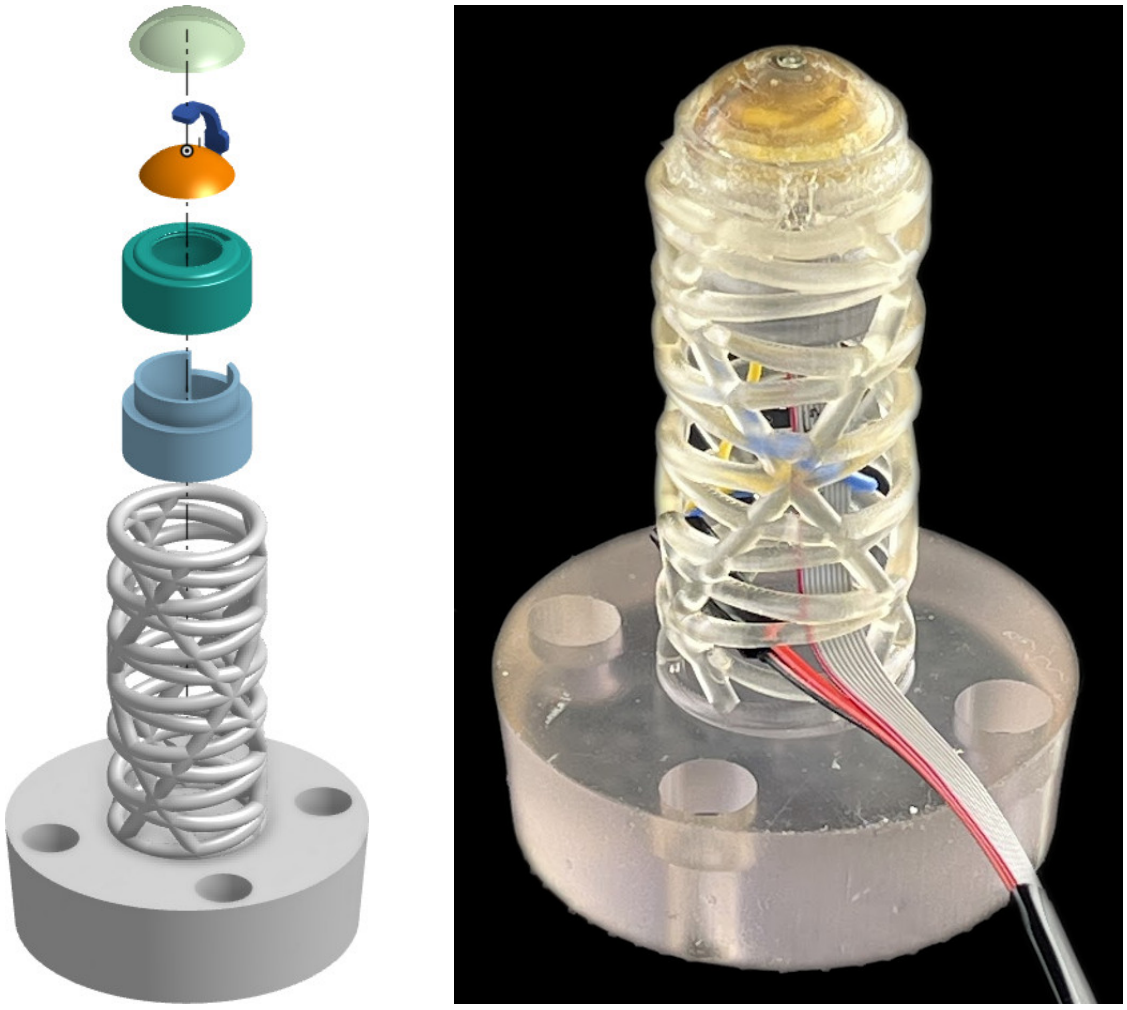
Figure 5.17: Dome-shaped joule heater. (a) The old (top) and new (bottom) PCB CAD designs. (b) A photo of a heater stretched over a black epoxy dome.

force. The first material moved was a playing card with a smooth, slightly textured surface that exhibited limited friction against the canvas. The second material was a small burlap swatch. Finally, a strip of friction tape was slid. The backing of the friction tape was intact, giving this sample a unique juxtaposition of very high friction against the canvas but very low friction for finger interaction. A photo of the experiment mid-slide is shown in figure 5.19b.

5.5.3 Data evaluation

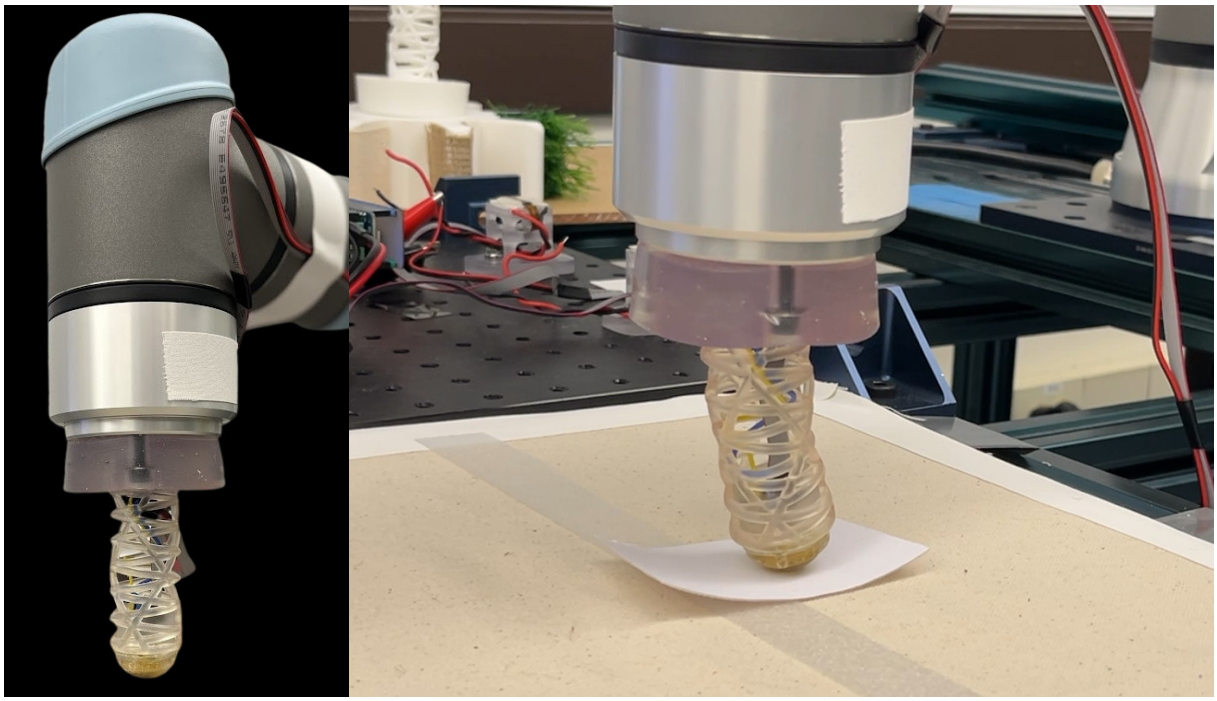
The three cases evaluated in this experiment demonstrate three unique sliding scenarios: a low-friction object (the playing card), a high-friction object (the burlap fabric), and an object that presents a high-surface interfacing with the base canvas substrate (the friction tape) but a low-friction surface for the interacting manipulator (the tape backing). Those three scenarios are presented in the first three rows of figure 5.20, with each row corresponding to the respective materials, indicated by the labels to the left of the row. On each row, the two leftmost columns present the initial and final frames of the manipulator in the dragging experiment while the end effector interface is in the hard state. The two rightmost columns present the initial and final frames of the interaction with the interface in the soft state.

In the case of the playing card, the light texturing and low friction make sliding over the canvas relatively simple in either the hard or the soft state. If the surface were smoother, it might have been more of a challenge in the hard state. However, once the card reaches the high-friction divider imposed by the friction tape, the card gets stuck on the tape while the interface in the hard state slides off. In the soft state, for the same normal force, the interface maintains contact with the card and slides over the high-friction zone without problem. Although increasing the normal force may sometimes resolve the issue of sliding, a higher



(a) Fixture CAD. (b) Fixture with sensor.

Figure 5.18: Finger UR5 mounting fixture. (a) The exploded CAD rendering of the fixture design. From top to bottom: SMP interface, embedded sensor PCB, joule heater, top device case, bottom device case, UR5 mounting fixture. (b) A photo of the fixture with the device embedded, all mounted on the robot arm.



(a) Mounted fixture. (b) Finger dragging a sample.

Figure 5.19: Finger material drag experimental setup. (a) The tunable-stiffness finger tip mounted to the wrist of a UR5e arm. (b) The arm, mid-experiment, dragging no-slip tape against the canvas platform.

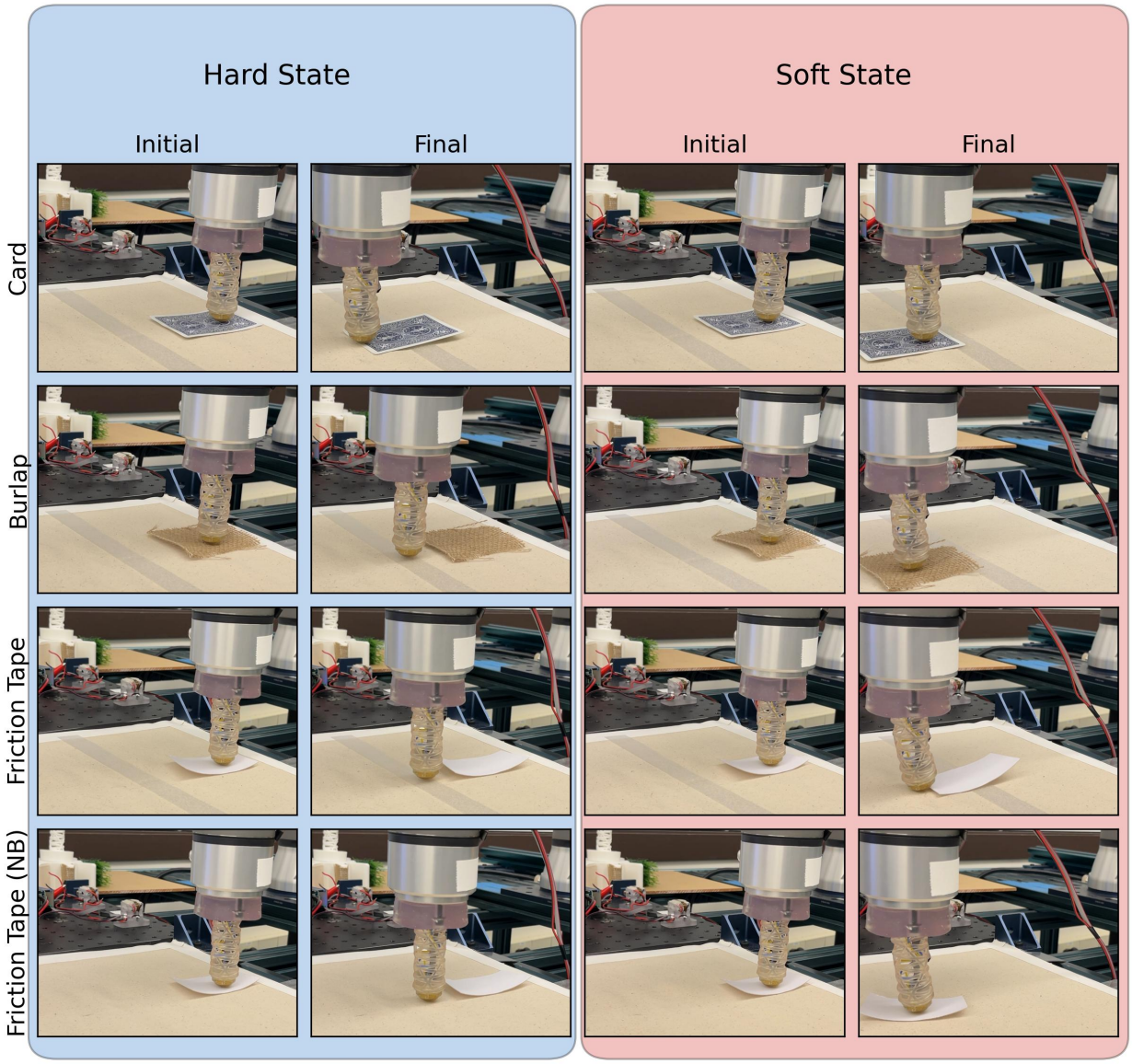


Figure 5.20: Finger drag comparisons for different materials between hard state and soft state operation. Materials are dragged across canvas, with a line of friction tape in the middle. In the final row, the friction tape has been removed.

normal force can also work against the robot over a high-friction zone, making some materials even more likely to get trapped on the tape. It should be noted that the soft interface allows the card to move over the high-friction strip without having to apply a higher normal force, and this is perhaps where such a capability in a manipulator can be beneficial.

Burlap presents a different challenge for the hard interface. At low normal forces, when the soft interface easily drags the material, the hard interface slides across the burlap. The friction between the burlap and the canvas is otherwise too high. A slightly higher normal force does allow for dragging of the burlap, and it does a bit better over the friction tape as well, mostly pulling the material across. However, the end effector still slips. Meanwhile, even set to the lower normal force, the manipulator easily drags the burlap over both the canvas and the friction tape while in the soft state.

The final test material was selected to be very challenging to drag. The friction tape grips the canvas well, whereas the tape backing was a smooth, non-tacky plastic, which offered little help to the finger. For the hard interface, this combination of very high sliding friction and very low interaction friction made the task impossible. Applying a greater normal force did not allow the end effector to drag the friction tape either; the end effector simply slid off the surface. However, in the soft state, the manipulator was able to move the material fairly easily across the canvas. Once the high-friction object met the high friction strip, even the soft surface failed to drag the object at any normal force applied. The friction differential was just too large. The final frame in the third row of figure 5.20 shows that the material gets stuck on the tape strip and the end effector slides off, even when soft. The final row shows the canvas substrate with the friction tape removed, and demonstrates the end effector otherwise has no problem moving the object over the canvas in the soft state.

5.6 Conclusions and future directions

Through this work, I have demonstrated the synergistic benefits of combining a tunable-stiffness epoxy interface with embedded acoustic sensing for robotics and sensing. The key innovation lies in the integration of these two technologies, which had previously been explored separately but never combined to enhance material discrimination capabilities.

Casting shape memory epoxy to form a tunable-stiffness interface offers significant advantages over previous designs. Although prior work has demonstrated the utility of variable stiffness in sensing, manipulation, and robotics, as opposed to relying on either complicated motion strategies or manually swapping in task-specific end effectors. However, existing implementations were often complex to fabricate and lacked mechanical robustness. The epoxy-based approach presented here provides a simpler, more durable solution. Thermal

activation is also a much simpler strategy to enable robots to dynamically adjust their contact properties rather than incorporating some form of actuation to actively change the surface, as was employed in most previous works.

The acoustic sensing capabilities demonstrated here reinforce previous findings that highlight the potential of low-cost microphones for tactile sensing. The novel contribution lies in demonstrating how stiffness tuning can enhance material discrimination. The epoxy coating serves two purposes: to protect the sensitive electronic components that lie close to the contact surface and to modulate acoustic feedback through the interface to provide richer information about the interaction.

There are several promising directions that emerge for future research:

- **Greater tunability**: The current binary nature of the SMP epoxy, which is either on or off, limits fine control over the interface properties. Although there is some variability in the transition region, the transition is typically fairly narrow, and that variability is due to an incomplete viscoelastic transition in the bulk of the SMP. It is neither easy to predict nor consistent how the contact dynamics will behave at those intermediate temperatures. Future work could explore multimaterial designs that incorporate multiple epoxies with different transition temperatures or investigate alternative materials that offer more gradual and controllable property changes while maintaining mechanical robustness.
- **Optimized surface texturing**: While the current flat substrate performed well compared to the circle texturing, more sophisticated texture designs may be able to better leverage the material's distinct characteristics between the hard and soft states. Texturing could be especially valuable when simultaneously inferring both material type and surface texture. The exploration of anisotropic texturing patterns combined with complex touch gestures to extract additional material information could also be an interesting avenue for exploration.
- **Inference of material properties**: Future tactile sensing research should likely focus more on inferring properties of materials rather than performing discrete classification. This approach would better generalize to novel materials, allowing robots to reason about previously unexperienced surfaces based on their fundamental physical properties. This information is also more relevant to how a robot should generally approach interactions with a particular object, whereas identifying a particular material is really just a proxy for a set of properties. This capability is essential for truly adaptive and general-purpose robotic manipulation.

This work provides an example of how the holistic design and integration of electronics

paired with a complimentary multifunctional material system can enable new capabilities. The robust integration of tunable material interfaces with tactile sensing may open new possibilities in robotic manipulation and tactile perception.

Part III

Final remarks

Chapter 6

Conclusions and future directions

In this work, I have presented an exploration of how the integration of soft materials and embedded electronics can greatly expand the capabilities of emerging soft robotics and soft wearable systems. By more tightly bridging the design and interaction between advanced materials, integrated electronics, and computation, I have demonstrated an approach to enable capabilities that would be more difficult or impossible to achieve with traditional rigid systems or modular designs. Figure 6.1 recaps some of the projects discussed in this work.

In chapter 2, the function of a robot limb was improved by adding sensors, enabling estimation and tracking of the limb's configuration. We were able to model the dynamics and then optimize open-loop controls based on the calibrated models that would have been far more difficult to do relying on an uncalibrated model or by simply hand-tuning. However, the brittleness of these models, stemming from restrictive and easily violated assumptions about kinematic and dynamic behavior, underscores the need for tighter integration between materials and sensors to overcome such limitations.

In chapter 3, I developed platforms for low-power and battery-free sensing and wireless communication to accommodate the tighter energy constraints of wearable systems. This is equally relevant to untethered soft robots that typically have limited load-carrying capacity which makes the use of large batteries difficult. I developed a hardware platform to enable energy harvesting, biometric sensing, and wireless communication of sensor waveform data without the need for batteries using soft theremoelectric generators designed by Zadan et al. [133]. Soft TEGs conform better to the body than typical rigid ceramic TEGs, enabling us to harvest body heat more efficiently and power the biometric sensor. This work highlights the feasibility of embedding sustainable energy harvesting mechanisms in soft devices, a useful feature for future wearables and lightweight untethered robots.

I discussed not only how electronics and computing can be used to enhance or enable

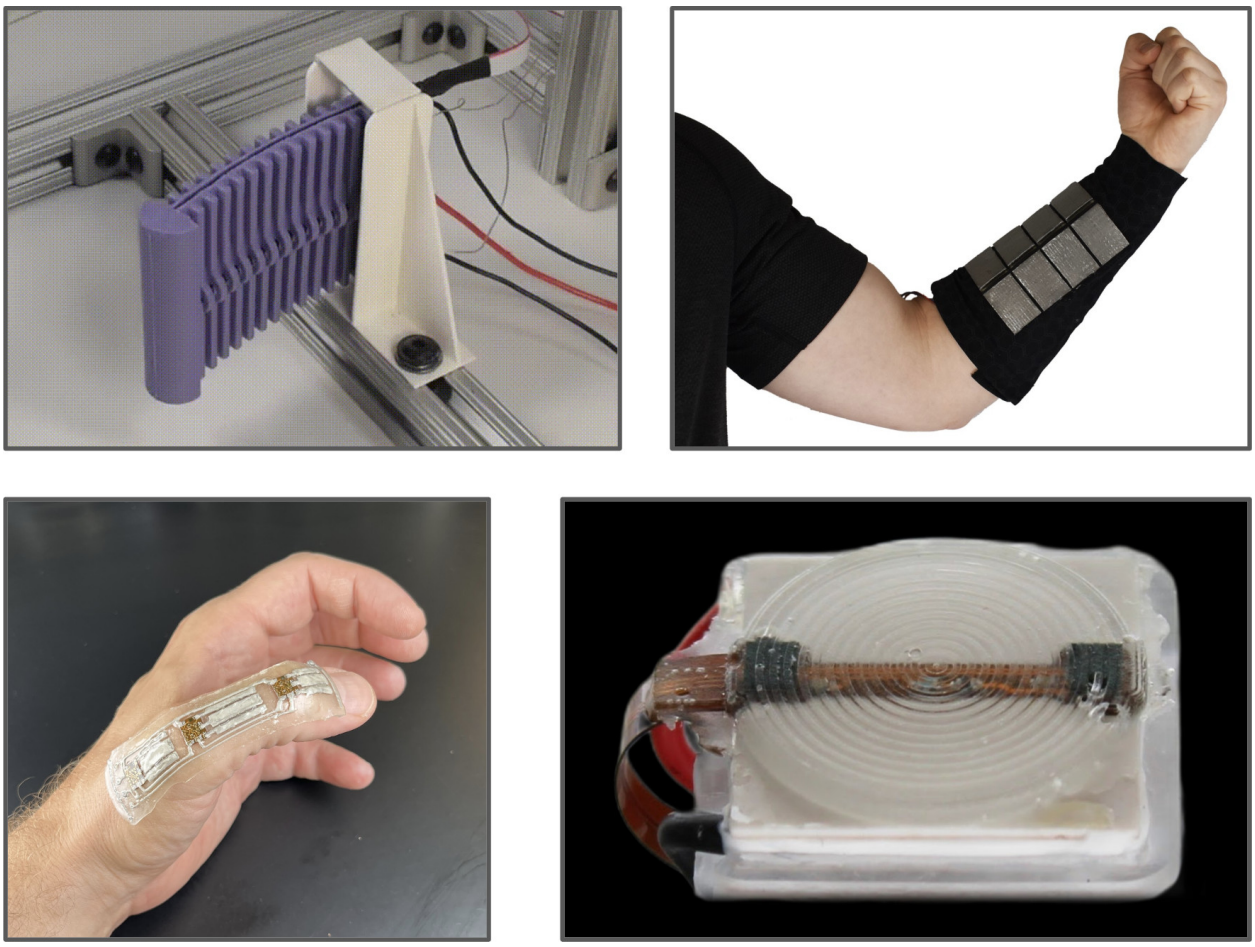


Figure 6.1: Summary of the projects highlighted in this work. (Top left) Soft robot limb modeled and controlled in chapter 2. (Top right) TEGSense sleeve for low-power, battery-free wearable sensing from chapter 3. (Bottom left) Encapsulated strain sensor from chapter 4. (Bottom right) Tunable-stiffness interface for contact adaptation and sensing described in chapter 5.

certain capabilities but also how a closer, more holistic design of those pieces together with advanced materials can be beneficial. The importance of co-designing materials, sensors, and electronics emerged as a central theme in chapters 4 and 5.

In chapter 4 I developed a framework for a stretchable strain sensor network that enabled distributed sensing with tunable sensitivity and signal isolation. This was accomplished even while keeping the network simple to integrate into a larger system, addressing key challenges of scalability, complexity, and signal interference in previous works. By enabling distributed sensing throughout a soft structure, for example, we can better mimic biological systems where localized sensing enables greater functionality. Such a framework could scale to applications like large-area sensing skins for wearable health monitoring or highly articulated soft robots.

A similar effect was seen in chapter 5 where I introduced a novel multifunctional material design for contact interfaces, allowing both the tuning of contact friction and improved tactile sensing. This design loosely mimics a key feature of biological structures, such as human fingertips, which can modulate surface friction to optimize grip and enhance sensory perception. Combining sensor feedback collected from both the hard and soft states improved its ability to discriminate between materials contacted based on their unique tactile signatures. Applications for this work range from robotic manipulation in unstructured environments to prosthetic devices that enable better tactile feedback.

Based on this work, we see that there is great promise in the future of soft systems with enhanced capabilities. It has also illuminated several interesting avenues for future research, such as:

- **Scalable distributed sensing**: The sensor framework introduced in chapter 4 could be laid out slightly differently to enable printed or stenciled structuring directly onto soft substrates using stretchable conductive inks. There are relatively few components and the connectivity is fairly simple. This would enable easy fabrication of large sensing skins and wearable sensors, as all of the circuit wiring, including the stretchable capacitors, could be structured directly, followed by the pick-and-place of a few discrete components for each sensor cell. More concretely, doing so could allow integration of a distributed bend sensing network in the limb introduced in chapter 2. This would enable localized bend sensing and remove the need for restrictive and easily invalidated kinematic assumptions.
- **Modeling of tribology and contact dynamics**: The tunable-stiffness interface presented in chapter 5 offers opportunities to study how surface interactions change under varying stiffness conditions. Such studies could help derive and validate models based on first-principles that better link material properties like texture, friction, and adhesion to their observed sensor responses. These insights could improve tactile perception in robots.
- **Exploring other tunable material properties**: The concept of using tunable stiffness to augment sensory feedback could be extended to other material properties, such as tunable conductivity, structure, or magnetization. Doing so could greatly expand the ability of a tactile sensor like the one we introduced to infer fundamental characteristics of materials and objects the device is touching.
- **AI-driven tactile perception**: Through modeling and machine learning, robots could move beyond simple material classification schemes and instead infer material *characteristics* such as texture, tackiness, or wetness. The objective could be to look at

how tactile sensations can inform a robot's understanding of material *qualities* instead of arbitrarily classifying particular material instances. A promising avenue could be integrating tactile sensing more closely with other data modalities, including leveraging large language models (LLMs) to interpret tactile feedback using human-readable descriptions. For instance, a robot could identify unknown objects by querying an LLM with labels of the inferred material properties derived from the sensor data, along with any other contextual information known to the robot. The LLM has access to the much richer tactile dataset embedded in the corpus of human literature compared to the relatively meager, limited, and often incompatible datasets collected from tactile sensors. This could provide a unified framework for interpreting tactile sensations from different devices and scaling its applicability across various domains. Such studies can be greatly enhanced by devices that can change contact dynamics while interrogating surfaces to improve characterization.

- **Capable design tools for soft systems**: Current fabrication processes require broad expertise in electronics, material science, and mechanical design, hamstringing our capability to design highly integrated soft systems. Automated design tools could help improve design quality and democratize access to capable soft system development. For example, a tool could be developed that designs strain sensor networks tailored to specific geometries or applications. A lack of easy-to-use tools is a limiting factor in the development of more capable soft systems.

In all, I believe this work highlights the utility of integrating electronics into advanced material systems more cohesively to enhance capabilities of soft systems, and it highlights very interesting avenues for future research.

Bibliography

[1] Sampada Acharya et al. “Gecko adhesion based sea star crawler robot”. In: *Frontiers in Robotics and AI* 10 (2023), p. 1209202.

[2] Carlo Alessi et al. “Rod models in continuum and soft robot control: a review”. In: *arXiv preprint arXiv:2407.05886* (2024).

[3] Thibaut André, Philippe Lefèvre, and Jean-Louis Thonnard. “Fingertip moisture is optimally modulated during object manipulation”. In: *Journal of neurophysiology* 103.1 (2010), pp. 402–408.

[4] Costanza Armanini et al. “Soft robots modeling: A structured overview”. In: *IEEE Transactions on Robotics* 39.3 (2023), pp. 1728–1748.

[5] Nazek El-Atab et al. “Soft actuators for soft robotic applications: a review”. In: *Advanced Intelligent Systems* 2.10 (2020), p. 2000128.

[6] Alexei Baevski et al. “wav2vec 2.0: A framework for self-supervised learning of speech representations”. In: *Advances in neural information processing systems* 33 (2020), pp. 12449–12460.

[7] Hedan Bai et al. “Stretchable distributed fiber-optic sensors”. In: *Science* 370.6518 (2020), pp. 848–852.

[8] James R Bainter. “Active filter has stable notch, and response can be regulated”. In: *Electronics* 48 (1975), pp. 115–117.

[9] Ryan Bena et al. “SMARTI: A 60-mg Steerable Robot Driven by High-Frequency Shape Memory Alloy Actuation”. In: *IEEE Robotics and Automation Letters* (2021).

[10] HN Bhargaw, M Ahmed, and P Sinha. “Thermo-electric behaviour of NiTi shape memory alloy”. In: *Transactions of Nonferrous Metals Society of China* 23.8 (2013), pp. 2329–2335.

[11] Raunaq Bhirangi et al. “Reskin: versatile, replaceable, lasting tactile skins”. In: *5th Annual Conference on Robot Learning*. 2021.

[12] Eric Brown et al. “Universal robotic gripper based on the jamming of granular material”. In: *Proceedings of the National Academy of Sciences* 107.44 (2010), pp. 18809–18814.

[13] D. Bruder, C. D. Remy, and R. Vasudevan. "Nonlinear System Identification of Soft Robot Dynamics Using Koopman Operator Theory". In: *2019 International Conference on Robotics and Automation (ICRA)*. May 2019, pp. 6244–6250. DOI: 10.1109/ICRA.2019.8793766.

[14] Trevor L Buckner et al. "A move-and-hold pneumatic actuator enabled by self-softening variable stiffness materials". In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2017, pp. 3728–3733.

[15] David B Camarillo et al. "Mechanics modeling of tendon-driven continuum manipulators". In: *IEEE Transactions on Robotics* 24.6 (2008), pp. 1262–1273.

[16] Tak Shing Chan and Andreas Carlson. "Physics of adhesive organs in animals". In: *The European Physical Journal Special Topics* 227.17 (2019), pp. 2501–2512.

[17] Eric T Chang et al. "An investigation of multi-feature extraction and super-resolution with fast microphone arrays". In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2024, pp. 3388–3394.

[18] Abraham Simpson Chen and Sarah Bergbreiter. "Electroadhesive feet for turning control in legged robots". In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2016, pp. 3806–3812.

[19] Xiaoqian Chen et al. "A review of soft manipulator research, applications, and opportunities". In: *Journal of Field Robotics* 39.3 (2022), pp. 281–311.

[20] Zixi Chen et al. "Data-driven methods applied to soft robot modeling and control: A review". In: *IEEE Transactions on Automation Science and Engineering* (2024).

[21] Shing Shin Cheng, Yeongjin Kim, and Jaydev P Desai. "Modeling and characterization of shape memory alloy springs with water cooling strategy in a neurosurgical robot". In: *Journal of Intelligent Material Systems and Structures* 28.16 (2017), pp. 2167–2183.

[22] Matteo Cianchetti et al. "Biomedical applications of soft robotics". In: *Nature Reviews Materials* 3.6 (2018), pp. 143–153.

[23] Dorin-Sabin Copaci et al. "Flexible shape memory alloy actuators for soft robotics: Modelling and control". In: *International Journal of Advanced Robotic Systems* 17.1 (2020), p. 1729881419886747.

[24] Ronald E Crochiere and Lawrence R Rabiner. "Interpolation and decimation of digital signals—A tutorial review". In: *Proceedings of the IEEE* 69.3 (1981), pp. 300–331.

[25] Cosimo Della Santina, Christian Duriez, and Daniela Rus. "Model-based control of soft robots: A survey of the state of the art and open challenges". In: *IEEE Control Systems Magazine* 43.3 (2023), pp. 30–65.

[26] Cosimo Della Santina et al. "Model-based dynamic feedback control of a planar soft robot: trajectory tracking and interaction with the environment". In: *The International Journal of Robotics Research* 39.4 (2020), pp. 490–513.

[27] Weiqiang Dou et al. "Soft robotic manipulators: Designs, actuation, stiffness tuning, and sensing". In: *Advanced Materials Technologies* 6.9 (2021), p. 2100018.

[28] A Fassler and C Majidi. "Soft-matter capacitors and inductors for hyperelastic strain sensing and stretchable electronics". In: *Smart Materials and Structures* 22.5 (2013), p. 055023.

[29] Satoshi Funabashi et al. "Multi-fingered in-hand manipulation with various object properties using graph convolutional networks and distributed tactile sensors". In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 2102–2109.

[30] Joey Z. Ge, Longlong Chang, and Néstor O. Pérez-Arancibia. "Preisach-model-based position control of a shape-memory alloy linear actuator in the presence of time-varying stress". en. In: *Mechatronics* 73 (Feb. 2021), p. 102452. ISSN: 0957-4158. DOI: 10.1016/j.mechatronics.2020.102452. URL: http://www.sciencedirect.com/science/article/pii/S0957415820301197 (visited on 02/03/2021).

[31] Christian Genest and James V Zidek. "Combining probability distributions: A critique and an annotated bibliography". In: *Statistical Science* 1.1 (1986), pp. 114–135.

[32] Giovanni Ghione and Carlo U Naldi. "Coplanar waveguides for MMIC applications: Effect of upper shielding, conductor backing, finite-extent ground planes, and line-to-line coupling". In: *IEEE transactions on Microwave Theory and Techniques* 35.3 (1987), pp. 260–267.

[33] Nathaniel N Goldberg et al. "On planar discrete elastic rod models for the locomotion of soft robots". In: *Soft robotics* 6.5 (2019), pp. 595–610.

[34] Moritz A Graule et al. "Somo: Fast and accurate simulations of continuum robots in complex environments". In: *2021 IEEE International Conference on Intelligent Robots and Systems (IROS)*. 2021.

[35] Jianglong Guo, Jinsong Leng, and Jonathan Rossiter. "Electroadhesion technologies for robotics: A comprehensive review". In: *IEEE Transactions on Robotics* 36.2 (2019), pp. 313–327.

[36] Carter S Haines et al. "Artificial muscles from fishing line and sewing thread". In: *Science* 343.6173 (2014), pp. 868–872.

[37] Chidanand Hegde et al. "Sensing in soft robotics". In: *ACS nano* 17.16 (2023), pp. 15277–15307.

[38] Tess Hellebrekers, Oliver Kroemer, and Carmel Majidi. "Soft magnetic skin for continuous deformation sensing". In: *Advanced Intelligent Systems* 1.4 (2019), p. 1900025.

[39] Weicheng Huang et al. "Dynamic simulation of articulated soft robots". In: *Nature Communications* 11.1 (2020), pp. 1–9.

[40] Xiaonan Huang et al. "Numerical Simulation of an Untethered Omni-Directional Star-Shaped Swimming Robot". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2021, pp. 11884–11890.

[41] Xiaonan Huang et al. "Shape memory materials for electrically-powered soft machines". In: *Journal of Materials Chemistry B* 8.21 (2020), pp. 4539–4551.

[42] Phillip Hyatt et al. "Configuration Estimation for Accurate Position Control of Large-Scale Soft Robots". In: *IEEE/ASME Transactions on Mechatronics* 24.1 (Feb. 2019). ISSN: 1941-014X. DOI: 10.1109/TMECH.2018.2878228.

[43] Fumiya Iida and Cecilia Laschi. "Soft robotics: Challenges and perspectives". In: *Procedia Computer Science* 7 (2011), pp. 99–102.

[44] Martin Kaltenbrunner et al. "An ultra-lightweight design for imperceptible plastic electronics". In: *Nature* 499.7459 (2013), pp. 458–463.

[45] Rohan Katoch and Jun Ueda. "Trajectory planning for antagonistic non-linearly quantized SMA actuator arrays based on evolutionary optimization". In: *2015 American Control Conference (ACC)*. IEEE. 2015, pp. 2631–2636.

[46] Matthew Kelly. "An introduction to trajectory optimization: How to do your own direct collocation". In: *SIAM Review* 59.4 (2017), pp. 849–904.

[47] DongWook Kim, Myungsun Park, and Yong-Lae Park. "Probabilistic modeling and Bayesian filtering for improved state estimation for soft robots". In: *IEEE Transactions on Robotics* 37.5 (2021), pp. 1728–1741.

[48] DongWook Kim and Yong-Lae Park. "Contact localization and force estimation of soft tactile sensors using artificial intelligence". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2018, pp. 7480–7485.

[49] Jaehoon Kim, Junhyung Kim, and Yong-Lae Park. "Multi-Modal Modular Textile Sensor for Physical Human–Robot Interaction Using Band-Stop Filters". In: *Advanced Functional Materials* 34.7 (2024), p. 2308571.

[50] Junhyung Kim, Suhan Kim, and Yong-Lae Park. "Single-input single-output multi-touch soft sensor systems using band-pass filters". In: *npj Flexible Electronics* 6.1 (2022), p. 65.

[51] Sangbae Kim, Cecilia Laschi, and Barry Trimmer. "Soft robotics: a bioinspired evolution in robotics". In: *Trends in biotechnology* 31.5 (2013), pp. 287–294.

[52] Donald A. Klosterman and Mary Galaska. "Power and Thermal Technology for Air and Space. Delivery Order 0006: Nano-filled Polymers for Electrical Insulation". In: (2006). DOI: 10.21236/ada465312.

[53] Cecilia Laschi, Barbara Mazzolai, and Matteo Cianchetti. "Soft robotics: Technologies and systems pushing the boundaries of robot abilities". In: *Science robotics* 1.1 (2016), eaah3690.

[54] Cecilia Laschi, Barbara Mazzolai, and Matteo Cianchetti. "Soft robotics: Technologies and systems pushing the boundaries of robot abilities". In: *Science Robotics* 1.1 (Dec. 2016), eaah3690. ISSN: 2470-9476. DOI: 10.1126/scirobotics.aah3690. URL: http://robotics.sciencemag.org/lookup/doi/10.1126/scirobotics.aah3690.

[55] Yann LeCun et al. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[56] Jeng-Hun Lee, Kilwon Cho, and Jang-Kyo Kim. "Age of flexible electronics: emerging trends in soft multifunctional sensors". In: *Advanced Materials* 36.16 (2024), p. 2310505.

[57] Moonyoung Lee et al. "Contact classification for agriculture manipulation in cluttered canopies". In: *AI for Agriculture and Food Systems*. 2021.

[58] Moonyoung Lee et al. "SonicBoom: Contact Localization Using Array of Microphones". In: *arXiv preprint arXiv:2412.09878* (2024).

[59] Alexander LeNail. "NN-SVG: Publication-Ready Neural Network Architecture Schematics." In: *J. Open Source Softw.* 4.33 (2019), p. 747.

[60] Jun Li, Wai-Yeung Wong, and Xiao-ming Tao. "Recent advances in soft functional materials: preparation, functions and applications". In: *Nanoscale* 12.3 (2020), pp. 1281–1306.

[61] Menglong Li et al. "Integration and Testing of a Three-Axis Accelerometer in a Woven E-Textile Sleeve for Wearable Movement Monitoring". en. In: *Sensors* 20.18 (Jan. 2020), p. 5033. DOI: 10.3390/s20185033. (Visited on 01/06/2022).

[62] Rinse W Liefferink et al. "Geometric control of sliding friction". In: *Extreme Mechanics Letters* 49 (2021), p. 101475.

[63] Jiaxun Liu and Boyuan Chen. "Sonicsense: Object perception from in-hand acoustic vibration". In: *arXiv preprint arXiv:2406.17932* (2024).

[64] Shihan Lu and Heather Culbertson. "Active Acoustic Sensing for Robot Manipulation". In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2023, pp. 3161–3168.

[65] A. T. Luong et al. "Long short term memory model based position-stiffness control of antagonistically driven twisted-coiled polymer actuators using model predictive

control". In: *IEEE Robotics and Automation Letters* (2021), pp. 1–1. ISSN: 2377-3766. DOI: 10.1109/LRA.2021.3068905.

[66] Carmel Majidi. "Soft Robotics: A Perspective - Current Trends and Prospects for the Future". In: *Soft Robotics* 1.1 (Mar. 2014), pp. 5–11. ISSN: 2169-5172. DOI: 10.1089/soro.2013.0001. URL: https://www.liebertpub.com/doi/10.1089/soro.2013.0001.

[67] Carmel Majidi. "Soft-matter engineering for soft robotics". In: *Advanced Materials Technologies* 4.2 (2019), p. 1800477.

[68] Carmel Majidi et al. "Liquid metal polymer composites: From printed stretchable circuits to soft actuators". In: *Flexible and Printed Electronics* 7.1 (2022), p. 013002.

[69] Andrew D. Marchese, Russ Tedrake, and Daniela Rus. "Dynamics and trajectory optimization for a soft spatial fluidic elastomer manipulator". en. In: *The International Journal of Robotics Research* 35.8 (July 2016), pp. 1000–1019. ISSN: 0278-3649. DOI: 10.1177/0278364915587926. URL: https://doi.org/10.1177/0278364915587926 (visited on 02/12/2020).

[70] Barbara Mazzolai et al. "Octopus-Inspired soft arm with suction cups for enhanced grasping tasks in confined environments". In: *Advanced Intelligent Systems* 1.6 (2019), p. 1900041.

[71] Barbara Mazzolai et al. "Roadmap on soft robotics: multifunctionality, adaptability and growth without borders". In: *Multifunctional Materials* 5.3 (2022), p. 032001.

[72] Gianmarco Mengaldo et al. "A concise guide to modelling the physics of embodied intelligence in soft robotics". In: *Nature Reviews Physics* 4.9 (2022), pp. 595–610.

[73] Mohammadreza Mollaei and Stephen Mascaro. "Optimal control of multi-input sma actuator arrays on a trajectory using graph theory: Modified a-star search algorithm". In: *Dynamic Systems and Control Conference*. Vol. 45301. American Society of Mechanical Engineers. 2012, pp. 743–750.

[74] Jorge J Moré. "The Levenberg-Marquardt algorithm: implementation and theory". In: *Numerical analysis: proceedings of the biennial Conference held at Dundee, June 28–July 1, 1977*. Springer. 2006, pp. 105–116.

[75] Vinod Nair and Geoffrey E Hinton. "Rectified linear units improve restricted boltzmann machines". In: *Proceedings of the 27th international conference on machine learning (ICML-10)*. 2010, pp. 807–814.

[76] Hussein Nesser and Gilles Lubineau. "Strain sensing by electrical capacitive variation: From stretchable materials to electronic interfaces". In: *Advanced Electronic Materials* 7.10 (2021), p. 2100190.

[77] Looh Augustine Ngiejungbwen, Hind Hamdaoui, and Ming-Yang Chen. "Polymer optical fiber and fiber Bragg grating sensors for biomedical engineering Applications: A comprehensive review". In: *Optics & Laser Technology* 170 (2024), p. 110187.

[78] Vassil Panayotov et al. "Librispeech: an asr corpus based on public domain audio books". In: *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE. 2015, pp. 5206–5210.

[79] Jonghwa Park et al. "Frequency-selective acoustic and haptic smart skin for dual-mode dynamic/static human-machine interface". In: *Science advances* 8.12 (2022), eabj9220.

[80] Myungsun Park, Taejun Park, and Yong-Lae Park. "Parametric analysis of multi-material soft sensor structures for enhanced strain sensitivity". In: *Extreme Mechanics Letters* 60 (2023), p. 101983.

[81] Zach J. Patterson, Andrew P. Sabelhaus, and Carmel Majidi. "Robust Control of a Multi-Axis Shape Memory Alloy-Driven Soft Manipulator". In: *IEEE Robotics and Automation Letters* (2022). DOI: 10.1109/LRA.2022.3143256.

[82] Zach J. Patterson et al. "An Untethered Brittle Star-Inspired Soft Robot for Closed-Loop Underwater Locomotion". In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Oct. 2020, pp. 8758–8764. DOI: 10.1109/IROS45743.2020.9341008.

[83] Bastian Pätzold et al. "Audio-based roughness sensing and tactile feedback for haptic perception in telepresence". In: *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2023, pp. 1387–1392.

[84] S. Vishnu Rajendran, Simon Parsons, and E. Amir Ghalamzan. "Single and Bi-Layered 2-D Acoustic Soft Tactile Skin". In: *2024 IEEE 7th International Conference on Soft Robotics (RoboSoft)*. IEEE. 2024, pp. 133–138.

[85] S. Vishnu Rajendran, Simon Parsons, and E. Amir Ghalamzan. "Towards Continuous Acoustic Tactile Soft Sensing". In: (2023).

[86] S. Vishnu Rajendran et al. "Acoustic soft tactile skin (AST skin)". In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2024, pp. 4105–4111.

[87] S. Vishnu Rajendran et al. "Enabling tactile feedback for robotic strawberry handling using ast skin". In: *Annual Conference Towards Autonomous Robotic Systems*. Springer. 2024, pp. 209–221.

[88] Manuel Reis Carneiro, Carmel Majidi, and Mahmoud Tavakoli. "Gallium-Based Liquid–Solid Biphasic Conductors for Soft Electronics". In: *Advanced Functional Materials* 33.41 (2023), p. 2306453.

[89] Peter Roberts, Mason Zadan, and Carmel Majidi. "Soft tactile sensing skins for robotics". In: *Current Robotics Reports* 2 (2021), pp. 343–354.

[90] Hugo Rodrigue et al. "An Overview of Shape Memory Alloy-Coupled Actuators and Robots". In: *Soft Robotics* 4.1 (Mar. 2017), pp. 3–15. ISSN: 2169-5172. DOI: 10.1089/soro.2016.0008. URL: https://www.liebertpub.com/doi/10.1089/soro.2016.0008.

[91] Ingrid A Rousseau and Tao Xie. "Shape memory epoxy: composition, structure, properties and shape memory performances". In: *Journal of Materials Chemistry* 20.17 (2010), pp. 3431–3441.

[92] Anthony P Russell, Alyssa Y Stark, and Timothy E Higham. "The integrative biology of gecko adhesion: historical review, current understanding, and grand challenges". In: *Integrative and comparative biology* 59.1 (2019), pp. 101–116.

[93] Andrew P Sabelhaus and Carmel Majidi. "Gaussian Process Dynamics Models for Soft Robots with Shape Memory Actuators". In: *2021 IEEE 4th International Conference on Soft Robotics (RoboSoft)*. IEEE. 2021, pp. 191–198.

[94] Andrew P. Sabelhaus et al. "Model-Predictive Control With Inverse Statics Optimization for Tensegrity Spine Robots". In: *IEEE Transactions on Control Systems Technology* 29.1 (Jan. 2021), pp. 263–277. ISSN: 1558-0865. DOI: 10.1109/TCST.2020.2975138.

[95] SM Hadi Sadati et al. "Mechanics of continuum manipulators, a comparative study of five methods with experiments". In: *Annual Conference Towards Autonomous Robotic Systems*. Springer. 2017, pp. 686–702.

[96] Abraham Savitzky and Marcel JE Golay. "Smoothing and differentiation of data by simplified least squares procedures." In: *Analytical chemistry* 36.8 (1964), pp. 1627–1639.

[97] Dylan Shah et al. "Stretchable Shape-Sensing Sheets". In: *Advanced Intelligent Systems* 5.12 (2023), p. 2300343.

[98] Dylan S Shah et al. "Robotic Skins with Integrated Actuation, Sensing, and Variable Stiffness". In: *IEEE Robotics and Automation Letters* (2023).

[99] Siavash Sharifi et al. "Dynamically tunable friction via subsurface stiffness modulation". In: *Frontiers in Robotics and AI* 8 (2021), p. 691789.

[100] Benjamin Shih et al. "Electronic skins and machine learning for intelligent soft robots". In: *Science Robotics* 5.41 (2020), eaaz9239.

[101] Hee-Sup Shin and Sarah Bergbreiter. "Effect of finger geometries on strain response of interdigitated capacitor based soft strain sensors". In: *Applied Physics Letters* 112.4 (2018).

[102] Jooyeon Shin et al. “Reversible wrinkling surfaces for enhanced grip on wet/dry conditions”. In: *ACS Applied Materials & Interfaces* 14.42 (2022), pp. 48311–48320.

[103] Harshal A Sonar et al. “An any-resolution pressure localization scheme using a soft capacitive sensor skin”. In: *2018 IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE. 2018, pp. 170–175.

[104] Hamid Souri et al. “Wearable and stretchable strain sensors: materials, sensing mechanisms, and applications”. In: *Advanced Intelligent Systems* 2.8 (2020), p. 2000039.

[105] Jon Åge Stakvik et al. “On implementation of the Preisach model identification and inversion for hysteresis compensation”. In: (2015).

[106] John S Steinhart and Stanley R Hart. “Calibration curves for thermistors”. In: *Deep sea research and oceanographic abstracts*. Vol. 15. 4. Elsevier. 1968, pp. 497–503.

[107] Pamela Svensson et al. “Electrotactile feedback for the discrimination of different surface textures using a microphone”. In: *Sensors* 21.10 (2021), p. 3384.

[108] Andreas Tairych and Iain A Anderson. “A numerical method for measuring capacitive soft sensors through one channel”. In: *Smart Materials and Structures* 27.3 (2018), p. 035016.

[109] Andreas Tairych and Iain A Anderson. “Distributed sensing: multiple capacitive stretch sensors on a single channel”. In: *Electroactive Polymer Actuators and Devices (EAPAD) 2017*. Vol. 10163. SPIE. 2017, p. 1016306.

[110] Z. Q. Tang et al. “A Probabilistic Model-Based Online Learning Optimal Control Algorithm for Soft Pneumatic Actuators”. In: *IEEE Robotics and Automation Letters* 5.2 (Apr. 2020), pp. 1437–1444. ISSN: 2377-3766. DOI: 10.1109/LRA.2020.2967293.

[111] Benjamin C-K Tee et al. “A skin-inspired organic digital mechanoreceptor”. In: *Science* 350.6258 (2015), pp. 313–316.

[112] Clark B Teeple et al. “Controlling palm-object interactions via friction for enhanced in-hand manipulation”. In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 2258–2265.

[113] Seppe Terryn et al. “A review on self-healing polymers for soft robotics”. In: *Materials Today* 47 (2021), pp. 187–205.

[114] Naveen Thirunavukkarasu et al. “Responsive friction modulation of 3D printed elastomeric bioinspired structures”. In: *Tribology International* 175 (2022), p. 107823.

[115] Deepak Trivedi et al. “Soft robotics: Biological inspiration, state of the art, and future research”. In: *Applied bionics and biomechanics* 5.3 (2008), pp. 99–117.

[116] Ryan L Truby. “Designing soft robots as robotic materials”. In: *Accounts of Materials Research* 2.10 (2021), pp. 854–857.

[117] Ryan L Truby, Cosimo Della Santina, and Daniela Rus. "Distributed proprioception of 3D configuration in soft, sensorized robots via deep learning". In: *IEEE Robotics and Automation Letters* 5.2 (2020), pp. 3299–3306.

[118] Enrico Turco et al. "Grasp Planning With a Soft Reconfigurable Gripper Exploiting Embedded and Environmental Constraints". In: *IEEE Robotics and Automation Letters* 6.3 (July 2021). ISSN: 2377-3766. DOI: 10.1109/LRA.2021.3072855.

[119] Jue Wang and Alex Chortos. "Control strategies for soft robot systems". In: *Advanced Intelligent Systems* 4.5 (2022), p. 2100165.

[120] Liyu Wang et al. "Controllable and reversible tuning of material rigidity for robot applications". In: *Materials Today* 21.5 (2018), pp. 563–576.

[121] Robert J Webster III and Bryan A Jones. "Design and kinematic modeling of constant curvature continuum robots: A review". In: *The International Journal of Robotics Research* 29.13 (2010), pp. 1661–1683.

[122] Peter Welch. "The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms". In: *IEEE Transactions on audio and electroacoustics* 15.2 (1967), pp. 70–73.

[123] Edward L White, Michelle C Yuen, and Rebecca K Kramer. "Distributed sensing in capacitive conductive composites". In: *2017 IEEE SENSORS*. IEEE. 2017, pp. 1–3.

[124] George M Whitesides. "Soft robotics". In: *Angewandte Chemie International Edition* 57.16 (2018), pp. 4258–4273.

[125] Arthur B. Williams. "Band-Reject Filters". en. In: 1st Edition. Analog Filter and Circuit Design Handbook. New York: McGraw-Hill Education, 2014. ISBN: 9780071816717. URL: https://www.accessengineeringlibrary.com/content/book/9780071816717/chapter/chapter6.

[126] Shuying Wu et al. "Strategies for designing stretchable strain sensors and conductors". In: *Advanced Materials Technologies* 5.2 (2020), p. 1900908.

[127] Hao Yang et al. "Design and Implementation of a Soft Robotic Arm Driven by SMA Coils". In: *IEEE Transactions on Industrial Electronics* 66.8 (Aug. 2019), pp. 6108–6116. ISSN: 1557-9948. DOI: 10.1109/TIE.2018.2872005.

[128] Oncay Yasa et al. "An overview of soft robotics". In: *Annual Review of Control, Robotics, and Autonomous Systems* 6.1 (2023), pp. 1–29.

[129] Jessica Yin, Tess Hellebrekers, and Carmel Majidi. "Closing the loop with liquid-metal sensing skin for autonomous soft robot gripping". In: *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE. 2020, pp. 661–667.

[130] Jessica Yin et al. “Wearable soft technologies for haptic sensing and feedback”. In: *Advanced Functional Materials* 31.39 (2021), p. 2007428.

[131] Seoung-Mok Yum et al. “Fingerprint ridges allow primates to regulate grip”. In: *Proceedings of the National Academy of Sciences* 117.50 (2020), pp. 31665–31673.

[132] Mason Zadan, Mohammad H Malakooti, and Carmel Majidi. “Soft and stretchable thermoelectric generators enabled by liquid metal elastomer composites”. In: *ACS applied materials & interfaces* 12.15 (2020), pp. 17921–17928.

[133] Mason Zadan et al. “Stretchable Thermoelectric Generators for Self-Powered Wearable Health Monitoring”. In: *Advanced Functional Materials* (2024), p. 2404861.

[134] Shadab Zaidi et al. “Actuation technologies for soft robot grippers and manipulators: A review”. In: *Current Robotics Reports* 2.3 (2021), pp. 355–369.

[135] Shuyun Zhuo et al. “Temperature and locomotion dual self-sensing soft robots based on liquid crystal polymer foams”. In: *Journal of Materials Chemistry C* (2025).