

# A Convex Optimization Framework for Robust-Feasible Series Elastic Actuators

Edgar A. Bolívar-Nieto<sup>a,\*</sup>, Tyler Summers<sup>b</sup>, Robert D. Gregg<sup>a</sup>, Siavash Rezazadeh<sup>c</sup>

<sup>a</sup>*Robotics Institute and Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI, USA*

<sup>b</sup>*Department of Mechanical Engineering, The University of Texas at Dallas, Richardson, TX, USA*

<sup>c</sup>*Department of Mechanical & Materials Engineering, University of Denver, Denver, CO, USA*

---

## Abstract

Kinematic and kinetic requirements for robotic actuators are subject to uncertainty in the motion of the load. Safety factors account for uncertainty in the design stage, but defining factors that translate to reliable systems without over-designing is a challenge. Bulky or heavy actuators resulting from overdesign are undesirable in wearable or mobile robots, which are prone to uncertainty in the load due to human-robot or robot-environment interaction. In this paper, we use robust optimization to account for uncertainty in the design of series elastic actuators. We formulate a robust-feasible convex optimization program to select the optimal compliance-elongation profile of the series spring that minimizes one or multiple of the following objectives: spring elongation, motor energy consumption, motor torque, or motor velocity. To preserve convexity when minimizing energy consumption, we lump the energy losses in the transmission as viscous friction losses, which is a viable approximation for series elastic actuators powered by direct or quasi-direct drives. Our formulation guarantees that the motor torque, winding temperature, and speed are feasible despite uncertainty in the load kinematics, kinetics, or manufacturing of the spring. The globally optimal spring could be linear or nonlinear. As simulation case studies, we design the optimal compliance-elongation profiles for multiple series springs for a robotic prosthetic ankle. The simulation case studies illustrate examples of our methodology, evaluate the performance of robust feasible designs against optimal solutions that neglect uncertainty, and provide insight into the selection of different objective functions. With this framework the designer specifies uncertainty directly in the optimization and over the specific kinematics, kinetics, or manufacturing parameters, aiming for reliable robots that reduce overdesign.

*Keywords:* Series elastic actuator, convex optimization, robust optimization, quasi-direct drives

---

## 1. Introduction

Twenty-five years ago, Pratt and Williamson [1] introduced the benefits of series elasticity to the design and control of actuators that interact with the environment. Springs connected in series with an electric motor, also known as series elastic actuators (SEAs), enabled the motor to control torque by controlling the spring elongation and improve torque tracking performance at low frequencies [2–4]. Fixed or variable series elasticity, in the form of SEAs or variable stiffness actuators (VSAs), can reduce the energy lost during impacts [5], improve safety for human-robot interaction [6], move loads with higher velocities [7], and reduce the energy consumption of the system [8–10]. From the power perspective, springs are mechanisms that store and release elastic potential energy, which modifies

the speed-torque requirements for the SEA motor. Springs are passive elements and cannot reduce the energy required to produce motion over a periodic cycle [11]; however, they can reduce the energy losses and adjust the speed-torque of the motor to make it robust against load uncertainty [12].

The torque-velocity requirements of the SEA motor depend on the load, which changes substantially during operation for human-robot or robot-environment interaction. For example, in the case of prosthetic robotic ankles, the requirements in motor torque and velocity change depending on the mass of the user, the extra mass that the user may carry, the speed of walking, the terrain, and the locomotion task [13]. Acknowledging uncertainty during the design stage is key for implementing reliable systems and introduces a trade-off between robustness and overdesign. Traditional safety factors account for uncertainty, but it is challenging to map these factors to specific representations of uncertainty or include them in the optimization stage of the design.

Brown and Ulsoy [14, 15] optimized parallel and series springs to improve energy efficiency despite task uncertainty. Their analysis of alternative maneuvers evaluated the performance of springs optimized for one maneuver when executing maneuvers for which they were not opti-

---

\*This work was supported by the National Science Foundation under Award Number 1830360 / 1953908. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NSF. R. D. Gregg holds a Career Award at the Scientific Interface from the Burroughs Wellcome Fund.

\*Corresponding author

*Email addresses:* [ebolivar@ieee.org](mailto:ebolivar@ieee.org) (Edgar A. Bolívar-Nieto), [tyler.summers@utdallas.edu](mailto:tyler.summers@utdallas.edu) (Tyler Summers), [rgregg@ieee.org](mailto:rgregg@ieee.org) (Robert D. Gregg), [siavash.rezazadeh@du.edu](mailto:siavash.rezazadeh@du.edu) (Siavash Rezazadeh)

mized. Their alternative maneuvers represented a stochastic parameterization of uncertainty. Our previous work [12] focused on the trade-off between uncertainty and feasibility of the actuator constraints. Using methods from robust optimization, a technique with origins in operations research [16], our designs satisfied motor-speed torque requirements despite the worst-case manifestation of uncertainty.

### Our contribution

A spring can be defined by the equation  $\tau_s = f(\delta_s)$ , where  $f: \mathbb{R} \rightarrow \mathbb{R}$  is the function that maps the spring elongation,  $\delta_s$ , to the spring torque,  $\tau_s$  [11]. Our contribution is to formulate a convex optimization program to design a series spring, i.e., define  $f(\delta_s)$ , and guarantee that the actuator is *robust feasible* (i.e., motor speed-torque, winding temperature, and spring elongation constraints are satisfied despite uncertainty in the load torque-speed, precision in manufacturing of the spring, and unmodeled dynamics). The optimization objective can be to minimize one or a positive-weighted sum of the following cost functions: motor energy consumption for any combination of periodic load trajectories (e.g., human walking and running) or any  $l_p$ -norm of motor torque or velocity for arbitrary load trajectories. To preserve convexity when minimizing energy consumption, we lump the energy losses in the transmission as viscous friction losses, which is a viable approximation for SEAs powered by direct or quasi-direct drives (see Sec. 2 for a definition of direct and quasi-direct drives).

Our previous work [11] discussed the convexity of the optimization program to minimize motor energy consumption for nonlinear springs, but there was no consideration of robust feasibility. In [12], we introduced robust feasibility but only for linear springs, did not match the actuator viscous friction losses with experimental values, and did not constrain the winding temperature. Our previous work only considered uncertain affine constraints [12], but constraining winding temperature requires the robust-feasible solution of an uncertain non-convex quadratic constraint. In order to include uncertainty due to load speed in our robust-feasible solution for nonlinear springs, we change our optimization variable to be a discrete trajectory of spring compliance as opposed to motor position in [11] or a scalar value of compliance in [12].

We present the model and constraints of SEAs in Sec. 2. We define the constraints as functions of the spring compliance, our optimization variable. Sec. 3 defines the robust-feasible design as a min-max program, which is then reduced to a computationally tractable convex optimization program. In the same section, we describe three possible choices of the minimization objective: motor energy consumption, motor torque, and motor velocity. In Sec. 4, we illustrate the application of our methodology for the design of SEA-powered prosthetic robotic ankles and discuss our simulation results. Sec. 5 summarizes the conclusions and limitations of our work.

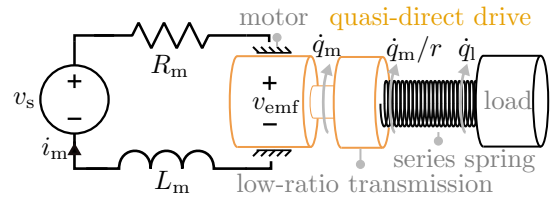


Figure 1: Electro-mechanical diagram of an SEA. Equations (1), (2) and (4) illustrate the system’s dynamics. We use the term quasi-direct drive to refer to a backdrivable low-ratio transmission combined with a high-torque motor, illustrated in orange in the diagram.

### Notation

In this paper, we use  $\mathbb{R}_+$  and  $\mathbb{R}_{++}$  to denote the set of non-negative and positive real numbers. Column vectors in  $\mathbb{R}^n$  are represented by bold lower-case characters and matrices in  $\mathbb{R}^{n \times m}$  are in bold upper-case characters. The subindex  $\mathbf{a}_i$  refers to the  $i$ -th element of the vector  $\mathbf{a}$ . We use  $\mathbf{1}$  to represent a vector with all its elements equal to 1 and dimensions to be interpreted by the context. We use  $\mathbf{b} = \text{abs}(\mathbf{a})$  to represent a vector  $\mathbf{b}$  with entries equal to the element-wise absolute value of  $\mathbf{a}$ . The term  $\text{diag}(\mathbf{x})$  refers to the square matrix with the vector  $\mathbf{x}$  along the diagonal. In the case of square matrices,  $\text{diag}(\mathbf{A})$  refers to the block diagonal matrix. The notation  $\mathbf{A}_i^*$  refers to the vector consisting of all the columns in the  $i$ -th row of matrix  $\mathbf{A}$ . The matrix inequality  $\mathbf{A} \succcurlyeq 0$  indicates that the matrix  $\mathbf{A}$  is positive semidefinite.

## 2. Dynamic Model and Constraints of SEAs

In this section, we illustrate the differential and algebraic equations that model the mechanical, electrical, thermal, and elastic behavior of SEAs. We use these equations to write the actuator constraints as affine or quadratic inequalities of the spring compliance; the robust-feasible solution of these constraints is fundamental to our method.

### 2.1. Electro-mechanical modeling

Using the Newton-Euler method, we sum the torques at the motor side (Fig. 1) to write the following equations of motion:

$$\tau_m = I_m \ddot{q}_m + b_m \dot{q}_m + \frac{\tau_s}{r} + \tau_{\text{unc}}, \quad (1)$$

$$\begin{aligned} \tau_l &= g(q_l, \dot{q}_l, \ddot{q}_l, \tau_e), \\ \tau_l &= -\tau_s, \end{aligned} \quad (2)$$

where  $I_m \in \mathbb{R}_{++}$  is the rotor inertia of the motor;  $b_m \in \mathbb{R}_{++}$  the motor’s viscous friction coefficient;  $r \in \mathbb{R}_{++}$  the reduction ratio of the gearbox;  $q_m, \dot{q}_m, \ddot{q}_m \in \mathbb{R}$  are the position, velocity, and acceleration of the motor respectively;  $\tau_s, \tau_m, \tau_l, \tau_{\text{unc}} \in \mathbb{R}$  are the series spring torque, motor’s electromagnetic torque, load torque, and uncertain torque respectively (Sec. 3.1.2 explains in detail the definition of  $\tau_{\text{unc}}$ ); and  $g: \mathbb{R}^4 \rightarrow \mathbb{R}$  is the function that defines

the load dynamics, e.g., in the case of an inertial load with viscous friction the load dynamics are defined by  $g(q_1, \dot{q}_1, \ddot{q}_1, \tau_e) = -I_1 \ddot{q}_1 - b_1 \dot{q}_1$ , where  $I_1$  is the inertia of the load, and  $b_1$  its corresponding viscous friction coefficient. We assume the load trajectory is known with some uncertainty and is defined as the set of variables  $q_1, \dot{q}_1, \ddot{q}_1, \tau_e$ ; Sec. 3.1 describes in detail our definition of uncertainty. The elongation of the series spring,  $\delta_s$ , is defined as

$$\delta_s := q_1 - q_m/r. \quad (3)$$

We assume that the torques generated by the gearbox can be lumped into the inertial and viscous friction torques of the motor, backlash is negligible, and the system is fully backdrivable. These assumptions imply that the motor speed after the gearbox is  $\dot{q}_m/r$  and Coulomb friction torques from the gearbox are negligible. We use these assumptions to show the convexity of our optimization program when minimizing energy consumption in Sec. 3. In practice, these assumptions apply for SEAs powered by *direct drives* (i.e., actuators without a transmission [17]) or *quasi-direct drives*. In this paper, a *quasi-direct drive* is any combination of an electric motor and a backdrivable gearbox where the transmission dynamics are negligible or fully captured by its inertial and viscous friction torques. Neglecting backlash and Coulomb friction of the gearbox is an approximation that depends on the type of transmission. In general, the lower the reduction ratio the more accurate is the approximation. The actuator in [18] with a 22:1 single-stage stepped-planet compound planetary gear transmission is an example of a *quasi-direct drive*. In [18], the authors experimentally showed that damping, inertia, and Coulomb friction have negligible effects in the performance of the actuator. Our energy analysis in Sec. 3 does not apply for non-backdrivable systems with considerable friction or backlash, which is typically the case for high-transmission-ratio SEAs.

Using Kirchhoff's voltage law across the motor's winding (Fig. 1), we model the electrical behavior of the SEA's motor with the following equation:

$$v_s = i_m R_m + L_m \frac{di_m}{dt} + v_{emf}, \quad (4)$$

where  $v_s \in \mathbb{R}$  is the voltage of the source,  $i_m \in \mathbb{R}$  is the motor current,  $R_m \in \mathbb{R}_{++}$  is the motor resistance,  $L_m \in \mathbb{R}_{++}$  is the motor inductance, and  $v_{emf} \in \mathbb{R}$  is the electromotive voltage of the motor. To simplify the analysis, we will assume that the voltage drop at the motor's inductance is negligible compared to the voltage drop at the winding resistance, which is a common assumption in practice [19]. In SI units, the electromagnetic torque and electromotive voltage relate to motor current and speed as follows:

$$\tau_m = k_t i_m, \quad v_{emf} = k_t \dot{q}_m, \quad (5)$$

where  $k_t \in \mathbb{R}_{++}$  is the motor torque constant. Substituting the electromotive voltage and motor current in (5) into (4) and neglecting the motor's inductance voltage, we express

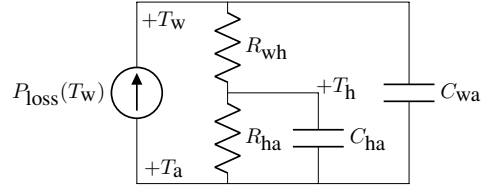


Figure 2: Thermal model of the motor's winding. It describes winding ( $T_w$ ) and housing ( $T_h$ ) temperatures as a function of the Joule heating generated by the motor current. In this electrical analogy, temperature is equivalent to voltage and heat flux is equivalent to current, as discussed in [20, 22, 23].

electromagnetic torque as a function of supplied voltage and motor speed

$$\tau_m = v_s \frac{k_t}{R_m} - \dot{q}_m \frac{k_t^2}{R_m}. \quad (6)$$

In this article, we will use the motor constant,  $k_m = k_t R_m^{-1/2}$ , to calculate motor heat losses in Sec. 3.3.1. The expression in (6) is typical for DC motors and could be adapted to brushless DC motors using field oriented control, representing the three-phase winding in the real and imaginary axis with the Clarke and Park transforms [20, 21].

In practice, the rotor inertia,  $I_m$ , is the sum of the motor's rotor inertia and the inertia of the gearbox (both are available in datasheets or through CAD). However, the motor's viscous friction coefficient,  $b_m$ , is rarely documented; one useful approximation is to estimate this coefficient from the no-load current,  $i_{mnl}$ , and no-load speed of the system,  $\dot{q}_{mnl}$ , using the equation:  $b_m = k_t i_{mnl} \dot{q}_{mnl}^{-1}$ . Experimentally, we can identify  $I_m$  and  $b_m$  fitting a first-order model to a system using  $i_m$  as input and  $\dot{q}_m$  as the output [18]. For more details in the experimental identification of the model parameters, we encourage the reader to check [18–20].

## 2.2. Thermal modeling

The electrical diagram in Fig. 2 models the motor winding ( $T_w$ ) and housing temperatures ( $T_h$ ) as a function of motor current, as reported in [20, 22, 23]. Balancing heat flux at the  $T_w$  and  $T_h$  nodes, we write the differential equations that model the thermal behavior as

$$\begin{aligned} P_{\text{loss}} &= \frac{T_w - T_h}{R_{wh}} + \frac{d(T_w - T_a)}{dt} C_{wa}, \\ \frac{T_w - T_h}{R_{wh}} &= \frac{T_h - T_a}{R_{ha}} + \frac{d(T_h - T_a)}{dt} C_{ha}, \end{aligned} \quad (7)$$

where  $T_w, T_h, T_a \in \mathbb{R}$  are the winding, housing, and ambient temperatures, respectively;  $R_{wh}, R_{ha} \in \mathbb{R}_{++}$  are the thermal resistances of winding-to-housing and housing-to-ambient;  $C_{wa}, C_{ha} \in \mathbb{R}_{++}$  are the thermal capacitances of winding-to-ambient and housing-to-ambient. The power lost due to Joule heating is

$$P_{\text{loss}} = i_m^2 R_m. \quad (8)$$

The winding's electrical resistance changes as a function of the winding's temperature based on

$$R_m = R_{m@a}(1 + \alpha_{Cu}(T_w - T_a)), \quad (9)$$

where  $R_{m@a}$  is the winding's electrical resistance at ambient temperature and  $\alpha_{Cu}$  is the copper's temperature coefficient of resistance. To give some perspective,  $R_m$  can increase around 50% with respect to  $R_{m@a}$  when  $T_w = 150^\circ\text{C}$  and  $T_a = 25^\circ\text{C}$ . Notice that changes in  $R_m$  imply that the motor constant is a function of temperature as well.

Accurate estimation of the winding's temperature is fundamental for the motor's health and generation of torque beyond the manufacturer's *continuous torque* (see Sec. 2.3.2). Some motor manufacturers, such as Maxon Motor, document the thermal capacitances and resistances in the motor's datasheet. If this information is not available, the designer can use thermal images or temperature sensors in the encapsulation of the winding or housing to identify the thermal parameters, as in [20, 22]. In general, it is good practice to update the thermal model after the final assembly. Different components in the mechanism, including the structural frame, can work as heat sinks and modify the thermal capacitances and resistances [24]. For example, in [23], the structural frame improved heat dissipation and increased the *continuous torque* by 46%.

### 2.3. Actuator constraints as functions of spring compliance

The SEA's motor and spring have constraints in torque, speed, winding temperature, and elongation. In this section, we represent these constraints as affine and quadratic inequalities of the spring compliance, which are computationally tractable forms for the robust design of the spring in Sec. 3.2. To numerically solve these inequalities, we discretize the continuous-time trajectories using  $n$  samples. Thus, sampling the continuous-time spring stiffness, i.e.,  $k_s = (d\tau_s)/(d\delta_s)$ , we write the spring compliance vector,  $\alpha_s \in \mathbb{R}_+^n$ , as follows:

$$\begin{aligned} \alpha_{si} &= \frac{d\delta_{si}}{d\tau_{si}}, \\ &= \frac{\dot{\delta}_{si}}{\dot{\tau}_{si}}, i = 1, \dots, n. \end{aligned} \quad (10)$$

This definition applies for  $\dot{\tau}_{si} \neq 0$ . For an energetically conservative spring,  $\dot{\tau}_{si} = 0$  implies that  $\dot{\delta}_{si} = 0$ . Thus,  $\alpha_{si}$  can be defined as  $\alpha_{si} = \alpha_{sj}$ , where  $j$  is the index of the last sample where  $\dot{\tau}_{sj} \neq 0$ .

#### 2.3.1. Constraining motor velocity

In practice, the input voltage and mechanical components, such as bearings, limit the velocity of the motor. Based on these limitations, we write the maximum motor speed as  $\dot{q}_{\max} \in \mathbb{R}_{++}$ , which means that the motor must satisfy  $|\dot{q}_m| \leq \dot{q}_{\max}$  during operation. Combining (10) and (3), we write the motor velocity as

$$\dot{q}_m = \mathbf{a} + \mathbf{B}\alpha_s, \quad (11)$$

where  $\mathbf{a} = \dot{q}_1 r$  and  $\mathbf{B} = -\text{diag}(\dot{\tau}_s)r$ . Using (11), we can enforce  $|\dot{q}_m| \leq \dot{q}_{\max}$  using the affine inequality

$$\begin{bmatrix} \mathbf{B} \\ -\mathbf{B} \end{bmatrix} \alpha_s \leq \begin{bmatrix} \mathbf{1}\dot{q}_{\max} - \mathbf{a} \\ \mathbf{1}\dot{q}_{\max} + \mathbf{a} \end{bmatrix}. \quad (12)$$

#### 2.3.2. Constraining maximum motor torque

Similarly as in (11), we write the motor acceleration as

$$\ddot{q}_m = \mathbf{c} + \mathbf{D}\alpha_s, \quad (13)$$

where  $\mathbf{c} = \ddot{q}_1 r$ ,  $\mathbf{D} = -\text{diag}(\ddot{\tau}_s)r - r\text{diag}(\dot{\tau}_s)\mathbf{D}_1$ , and  $\mathbf{D}_1$  is the first-order derivative operator matrix [10, 25]. Substituting (11) and (13) into (1) we write the motor's electromagnetic torque as

$$\tau_m = \mathbf{e} + \mathbf{F}\alpha_s, \quad (14)$$

where  $\mathbf{e} = I_m \mathbf{c} + b_m \mathbf{a} - \tau_s r^{-1} - \tau_{\text{unc}}$  and  $\mathbf{F} = I_m \mathbf{D} + b_m \mathbf{B}$ . The *continuous torque* limits the torque that can be applied in continuous operation, which is normally associated with the  $\text{RMS}(\tau_m)$  and winding temperature (see Sec. 2.3.3). Assuming that the winding temperature is within safe limits, electric motors may apply sporadic torques up to the *saturation torque*, which is the torque limit where parts of iron in the stator become magnetically saturated. The *saturation torque* defines the current limit before the current and torque relationship becomes highly nonlinear [26]. In this section, we assume that the *saturation torque* defines the maximum torque,  $\tau_{\max} \in \mathbb{R}$ . Using (14), we enforce  $|\tau_m| \leq \tau_{\max}$  with the inequality

$$\begin{bmatrix} \mathbf{F} \\ -\mathbf{F} \end{bmatrix} \alpha_s \leq \begin{bmatrix} \mathbf{1}\tau_{\max} - \mathbf{e} \\ \mathbf{1}\tau_{\max} + \mathbf{e} \end{bmatrix}. \quad (15)$$

#### 2.3.3. Constraining motor winding temperature

We can limit the temperature of the winding using one of two methods: 1) indirectly by constraining the RMS torque or 2) directly by explicitly modeling or sensing the winding temperature. In the Appendix A, we show that the direct method constrains winding temperature using a finite set of quadratic inequalities (A.4). The robust-feasible formulation of these inequalities complicates the notation and the robust solution of the optimization program. In the remainder of this paper, we will only consider the indirect method to simplify our robust-feasible formulation in Sec. 3.

In practice, many electric motors operate without measurements or estimations of the winding temperature. In this case, the designer can constrain the motor winding temperature to be within a safe region by constraining the RMS motor torque to be less than the *continuous torque*, i.e.,  $\text{rms}(\tau_m) \leq \tau_{\text{cont}}$ , where  $\tau_{\text{cont}} \in \mathbb{R}_{++}$  is the motor *continuous torque*. Normally, manufacturers report the *continuous torque* in the motor's datasheet. In our formulation, we use (14) to write  $\text{rms}(\tau_m) \leq \tau_{\text{cont}}$  as the following quadratic function of spring compliance

$$\begin{aligned} \sqrt{\tau_m^T \tau_m n^{-1}} &\leq \tau_{\text{cont}}, \\ \alpha_s^T \mathbf{F}^T \mathbf{F} \alpha_s + 2\mathbf{e}^T \mathbf{F} \alpha_s + \mathbf{e}^T \mathbf{e} &\leq \tau_{\text{cont}}^2 n. \end{aligned} \quad (16)$$

### 2.3.4. The motor torque-velocity constraint

An electric motor can operate as generator and actuator. As the rotor spins with constant voltage input, the generated electromotive voltage limits the current that can flow through the winding as seen in (6). In addition, the PWM to the motor can vary the motor voltage from zero up to the battery voltage. Thus, for positive velocity and torque, the equality in (6) becomes the following inequality:  $\tau_m \leq k_t R_m^{-1} v_s - k_t^2 R_m^{-1} \dot{q}_m$ , i.e., the higher the motor velocity the lower the torque that can provide. As a result, in the four quadrants of the speed-torque plane, the speed and torque of the motor should satisfy the inequality [27]:

$$|\tau_m| \leq \frac{k_t}{R_m} v_s - \frac{k_t^2}{R_m} |\dot{q}_m|,$$

which can be written as an affine function of compliance using (11) and (14) as follows:

$$\begin{bmatrix} \mathbf{F} + k_t^2 R_m^{-1} \mathbf{B} \\ \mathbf{F} - k_t^2 R_m^{-1} \mathbf{B} \\ -\mathbf{F} + k_t^2 R_m^{-1} \mathbf{B} \\ -\mathbf{F} - k_t^2 R_m^{-1} \mathbf{B} \end{bmatrix} \boldsymbol{\alpha}_s \leq \begin{bmatrix} \mathbf{1} k_t R_m^{-1} v_s - \mathbf{e} - \mathbf{a} k_t^2 R_m^{-1} \\ \mathbf{1} k_t R_m^{-1} v_s - \mathbf{e} + \mathbf{a} k_t^2 R_m^{-1} \\ \mathbf{1} k_t R_m^{-1} v_s + \mathbf{e} - \mathbf{a} k_t^2 R_m^{-1} \\ \mathbf{1} k_t R_m^{-1} v_s + \mathbf{e} + \mathbf{a} k_t^2 R_m^{-1} \end{bmatrix} \quad (17)$$

### 2.3.5. Constraining maximum spring elongation

The elongation of the spring relates to its compliance by  $d\delta_s = \alpha_s d\tau_s$ . Using the trapezoidal method to numerically approximate the integral, we obtain

$$\begin{aligned} \delta_s &= \int_{t_0}^{t_f} \alpha_s \dot{\tau}_s dt + \delta_{s0} \\ &\approx \sum_{i=1}^{i=n} \frac{\alpha_{s_i} \dot{\tau}_{s_i} + \alpha_{s_{(i+1)}} \dot{\tau}_{s_{(i+1)}}}{2} \Delta t + \delta_{s0}, \end{aligned}$$

which in vector form is equal to

$$\boldsymbol{\delta}_s = \mathbf{L} \boldsymbol{\alpha}_s + \mathbf{1} \delta_{s0}, \quad (18)$$

where  $\mathbf{L}$  is the cumulative integration operator in matrix form, i.e.,

$$\mathbf{L} = \frac{\Delta t}{2} \begin{bmatrix} 0 & \dots & \dots & \dots & 0 \\ \dot{\tau}_{s_1} & \dot{\tau}_{s_2} & 0 & \dots & 0 \\ \dot{\tau}_{s_1} & 2\dot{\tau}_{s_2} & \dot{\tau}_{s_3} & \dots & 0 \\ \vdots & & & \ddots & 0 \\ \dot{\tau}_{s_1} & 2\dot{\tau}_{s_2} & \dots & 2\dot{\tau}_{s_{n-1}} & \dot{\tau}_{s_n} \end{bmatrix}.$$

To guarantee that the elongation (18) stays within safe margins, we will enforce  $|\delta_s| \leq \delta_{s\max}$  satisfying the following inequality:

$$\begin{bmatrix} \mathbf{L} \\ -\mathbf{L} \end{bmatrix} \boldsymbol{\alpha}_s \leq \begin{bmatrix} \mathbf{1} \delta_{s\max} - \mathbf{1} \delta_{s0} \\ \mathbf{1} \delta_{s\max} + \mathbf{1} \delta_{s0} \end{bmatrix}. \quad (19)$$

To simplify notation, we lump the affine inequalities (12), (15), and (19) as

$$\mathbf{M} \boldsymbol{\alpha}_s \leq \mathbf{p} \quad (20)$$

where  $\mathbf{p} \in \mathbb{R}^m$ ,  $\mathbf{M} \in \mathbb{R}^{m \times n}$ , and  $m$  is the number of rows that results from stacking the affine inequalities vertically. (20) and (16) represent our complete set of constraints.

## 3. Spring Design for Robust Feasible Actuators

Previous research explored the effects of series springs on torque bandwidth [28], energy consumption [11], peak power [11, 29], and impedance control for human-robot interaction [3]. However, the benefits of series springs to enhance system robustness are not clearly understood. In this section, we illustrate how series springs can enhance robustness by guaranteeing that the elongation of the spring, motor torque, winding temperature, and motor speed are within feasible regions despite uncertainty in the load, unmodeled dynamics, or limited precision during manufacturing of the spring. In other words, our objective is to design  $\boldsymbol{\alpha}_s$  so that the constraints (20) and (16) are feasible despite uncertainty in  $\mathbf{M}$ ,  $\mathbf{p}$ ,  $\mathbf{F}$ , and  $\mathbf{e}$ . The solution of affine and quadratic inequalities with uncertainty in the parameters is part of the field of robust optimization [16]. In addition to robust feasibility, we illustrate how to design series springs to minimize energy consumption, motor torque, or motor velocity without sacrificing robustness during operation. For all the cases, we will show that our formulation leads to convex optimization programs with computationally tractable solutions. The numerical solvers for these optimization programs reach globally optimal solutions regardless of the initial conditions and within polynomial time.

### 3.1. Sources of uncertainty

Our formulation of the actuator constraints in Sec. 2.3 is broad enough to accommodate multiple sources of uncertainty. However, we will focus on parameters that are relevant to the application of powered prosthetic legs and exoskeletons. For example, the robot kinematics are uncertain due to unexpected changes in the terrain [30]; the kinetics may change if the user changes mass, e.g., by wearing a backpack; and the limited manufacturing precision may lead to uncertainty in the spring stiffness. Our approach can be extended to account for sources of uncertainty not considered in this section. In general, we will introduce two kinds of uncertainties: additive and multiplicative. For example, if the variable  $x$  is uncertain, we will use  $\varepsilon_{x_a}$  and  $\varepsilon_{x_m}$  to identify its corresponding additive and multiplicative uncertainties with respect to the nominal value  $\bar{x}$ , i.e.,  $x = \bar{x}(1 \pm \varepsilon_{x_m}) \pm \varepsilon_{x_a}$ . In this work, we consider the following sources of uncertainty:

#### 3.1.1. Manufacturing of the spring

Uncertainty in the spring stiffness is typical due to limited manufacturing tolerances. For instance, in [31] the designed stiffness of a rotational spring is 100 N·m. However, the stiffness of the manufactured springs varied with a standard deviation of about 10%. In this work, we consider additive uncertainty in the spring compliance, which manifest in  $r_i$  and the vector  $\mathbf{p}$ . For example,  $\mathbf{p} \in \mathcal{U}_{\alpha_s}$

as defined by the following uncertainty set:

$$\mathcal{U}_{\alpha_s} = \{\mathbf{x} \in \mathbb{R}^m : \bar{\mathbf{p}} - \text{abs}(\mathbf{M} \cdot \mathbf{1})\varepsilon_{\alpha_{sa}} \leq \mathbf{x} \leq \bar{\mathbf{p}} + \text{abs}(\mathbf{M} \cdot \mathbf{1})\varepsilon_{\alpha_{sa}}\}, \quad (21)$$

where  $\bar{\mathbf{p}}$  defines the nominal value of  $\mathbf{p}$ , i.e., without uncertainty, and  $\varepsilon_{\alpha_{sa}}$  defines the additive uncertainty in the compliance of the spring. In our case studies, we consider  $\varepsilon_{\alpha_{sa}} = \pm 20\%$  of the optimal linear-spring compliance (Table 1).

### 3.1.2. Unmodeled dynamics

Also defined as the uncertain torque,  $\boldsymbol{\tau}_{\text{unc}}$ , in (1). This unmodeled dynamics torque lumps unmodeled effects at the motor and load side such as cogging torque and friction. The elements of the  $n$  dimensional vector  $\boldsymbol{\tau}_{\text{unc}} \in \mathcal{U}_{\boldsymbol{\tau}_{\text{unc}}}$  are defined by the uncertainty set,

$$\mathcal{U}_{\boldsymbol{\tau}_{\text{unc}}} = \{\mathbf{x} \in \mathbb{R}^n : -\varepsilon_{\boldsymbol{\tau}_{\text{unc}i}} \leq x_i \leq \varepsilon_{\boldsymbol{\tau}_{\text{unc}i}}, i = 1, \dots, n\}, \quad (22)$$

where  $\boldsymbol{\varepsilon}_{\boldsymbol{\tau}_{\text{unc}}} \in \mathbb{R}^n$  is the vector of additive uncertainty that defines the box constraints in  $\boldsymbol{\tau}_{\text{unc}}$ .

### 3.1.3. Kinematics and kinetics of the load

In some applications such as rehabilitation robotics one of the biggest sources of uncertainty is the definition of the motion task. In the case of robotic knee and ankle prostheses, the reference kinematics and kinetics vary considerably as humans change walking speed or inclination during locomotion [13], or perform custom tasks that are difficult to predict, e.g., dancing, jumping, or adapting their natural gait due to irregular terrain. Even for the same motion task, such as level-ground walking, the mean coefficient of variation (standard deviation divided by the sample mean) over a stride of the knee joint angle can be as high as 25% [32].

Thus, explicit consideration of uncertainty in the kinematics and kinetics is critical for a robust design. In our work, we consider additive and multiplicative uncertainty for position, velocity, acceleration, and torque of the load. For example, we define the load position vector as  $\mathbf{q}_1 \in \mathcal{U}_{\mathbf{q}_1}$ , where

$$\mathcal{U}_{\mathbf{q}_1} = \{\mathbf{x} \in \mathbb{R}^n : \bar{\mathbf{q}}_1 - |\bar{\mathbf{q}}_1| \varepsilon_{q_{1m}} - \boldsymbol{\varepsilon}_{\mathbf{q}_{1ai}} \leq \mathbf{x} \leq \bar{\mathbf{q}}_1 + |\bar{\mathbf{q}}_1| \varepsilon_{q_{1m}} + \boldsymbol{\varepsilon}_{\mathbf{q}_{1ai}}, i = 1, \dots, n\} \quad (23)$$

is the uncertainty load position set and  $\bar{\mathbf{q}}_1$  is the nominal position vector for the load. The same definition applies to  $\mathcal{U}_{\dot{\mathbf{q}}_1}$ ,  $\mathcal{U}_{\ddot{\mathbf{q}}_1}$ ,  $\mathcal{U}_{\boldsymbol{\tau}_1}$ . We assume that these uncertainty sets are independent of each other, i.e., the uncertainty on velocity of the joint is independent of its uncertainty in position. In a multiple degree of freedom application, the uncertainty of one joint can affect the other through our external torque,  $g(q_m, \dot{q}_m, \ddot{q}_m, \tau_e)$ , in (2).

### 3.2. Design of spring compliance for a robust feasible actuator

The general idea of our robust design is to select a value of compliance,  $\alpha_s$ , to solve the following optimization program:

$$\begin{aligned} & \underset{\alpha_s}{\text{minimize}} && g(\alpha_s) \\ & \text{subject to} && \alpha_s^T \mathbf{F}^T \mathbf{F} \alpha_s + 2\mathbf{e}^T \mathbf{F} \alpha_s + \mathbf{e}^T \mathbf{e} \leq \tau_{\text{cont}}^2 n \\ & && \mathbf{M} \alpha_s \leq \mathbf{p}, \forall \{\mathbf{M}, \mathbf{p}, \mathbf{F}, \mathbf{e}\} \in \mathcal{U} \end{aligned} \quad (24)$$

where  $g$  is the objective function that depends on the optimization variable  $\alpha_s$ . A robust solution to this optimization program is a value of the spring compliance that satisfies the actuator constraints for all possible realizations of uncertainty within the uncertainty set, i.e.,  $\forall \{\mathbf{M}, \mathbf{p}, \mathbf{F}, \mathbf{e}\} \in \mathcal{U}$ , where  $\mathcal{U} = \mathcal{U}_{\mathbf{q}_1} \times \mathcal{U}_{\dot{\mathbf{q}}_1} \times \mathcal{U}_{\ddot{\mathbf{q}}_1} \times \mathcal{U}_{\boldsymbol{\tau}_1} \times \mathcal{U}_{\boldsymbol{\tau}_{\text{unc}}} \times \mathcal{U}_{\alpha_s}$ . The optimization program (24) has affine and quadratic inequalities. The following sections explain the robust feasible solution for each kind of constraint.

#### 3.2.1. Robust feasible solution of affine constraints

The definition of  $\mathcal{U}$  indicates whether the uncertainty affects the left-hand, right-hand, or both sides of the inequality constraints. For example, the uncertainty due to limited manufacturing precision of the spring affects exclusively  $\mathbf{p}$ , as defined in (21). However, uncertainty in  $I_m$  affects simultaneously  $\mathbf{M}$  and  $\mathbf{p}$  as it relates to the definition of left-hand and right-hand terms of the inequality, e.g.,  $\mathbf{F}$  and  $\mathbf{e}$  in (15). When the uncertainty manifest exclusively in  $\mathbf{p}$ , the robust solution is straightforward; we just need to find  $p_i^* = \inf\{p_i \in \mathcal{U} \mid i = 1, \dots, m\}$ , and replace  $p_i$  with  $p_i^*$  in the definition of (24). In the general case, when the uncertainty is coupled in the left-hand and right-hand side of the inequality, the robust solution of the optimization program requires the solution of the following min-max optimization program:

$$\begin{aligned} & \min_{\alpha_s} && g(\alpha_s) \\ & \text{s.t.} && \alpha_s^T \mathbf{F}^T \mathbf{F} \alpha_s + 2\mathbf{e}^T \mathbf{F} \alpha_s + \mathbf{e}^T \mathbf{e} \leq \tau_{\text{cont}}^2 n \\ & && \left[ \begin{array}{l} \max_{\mathbf{m}_i, p_i} \quad \mathbf{m}_i^T \alpha_s - p_i \\ \text{s.t.} \quad \mathbf{m}_i, p_i \in \mathcal{U} \end{array} \right] \leq 0, i = 1, \dots, m. \end{aligned} \quad (25)$$

The inner maximization program represents the worst possible realization of the uncertain parameters over the uncertainty set. The optimization variables in the inner optimization program are  $\mathbf{m}_i$  and  $p_i$  instead of the vector of spring compliance. The optimal values  $\mathbf{m}_i^*$  and  $p_i^*$  maximize the left hand side of the inequality, i.e., they represent the worst possible scenario. Ben-Tal et al. [16] present the solution of a similar version of (25) when  $\mathcal{U}$  is a polytope or a ellipsoid, which is our case as the uncertainty set is a collection of box constraints (Sec. 3.1). However, our solution of (25) will be slightly different from [16]. Our solution method will separate the dot product  $\mathbf{m}_i^T \alpha_s$  based on

known and uncertain elements of  $\mathbf{M}$ . Without this separation, the optimization program may be incorrectly classified as infeasible, as shown in this section. We solve (25) in two steps: 1) we rewrite the min-max program (25) as a min-min program using the Lagrange dual of the maximization program that defines the constraints; 2) we write the resulting min-min program as a single convex-optimization program.

In the solution of (25), we only include the uncertain elements of  $\mathbf{M}$  and  $\mathbf{p}$  into the inner maximization program by writing the constraints as

$$\begin{bmatrix} \max. & \mathbf{q}_{ui}^T \mathbf{x}_{ui} \\ \mathbf{q}_{ui} \\ \text{s.t.} & \mathbf{V}_i \mathbf{q}_{ui} \leq \mathbf{v}_i \end{bmatrix} + \mathbf{q}_{ci}^T \mathbf{x}_{ci} \leq 0, \quad i = 1, \dots, m. \quad (26)$$

where  $\mathbf{q}_{ui}$  is the vector with the uncertain elements of  $\mathbf{M}$  and  $\mathbf{p}$ ,  $\mathbf{x}_{ui}$  is the vector of compliance elements that multiply the uncertain coefficients,  $\mathbf{V}_i$  and  $\mathbf{v}_i$  define the box constraints on the elements of  $\mathbf{q}_{ui}$  as defined in Sec. 3.1,  $\mathbf{q}_{ci}$  is the vector with known elements of  $\mathbf{M}$  and  $\mathbf{p}$ ,  $\mathbf{x}_{ci}$  is the vector of compliance elements that multiply the known coefficients, and the subindex  $i$  refers to the  $i$ -th row in  $\mathbf{M}$  and  $\mathbf{p}$ . For example, if all the elements of row  $i$  are uncertain then

$$\mathbf{q}_{ui} = \begin{bmatrix} \mathbf{m}_i^T \\ p_i \end{bmatrix}, \quad \mathbf{x}_{ui} = \begin{bmatrix} \alpha_s \\ 1 \end{bmatrix}, \quad \mathbf{q}_{ci} = 0, \quad \mathbf{x}_{ci} = 0.$$

In our case,  $\mathbf{M}$  represents a sparse matrix, which implies that most of the elements in  $\mathbf{m}_i$  are zero by construction. These known coefficients and corresponding compliance elements are assigned to  $\mathbf{q}_{ci}$  and  $\mathbf{x}_{ci}$ , respectively.

As a first step, we write the Lagrange dual of the inner linear program [25, p. 219] in (26) as

$$\begin{bmatrix} \text{minimize} & \lambda_i^T \mathbf{v}_i \\ \lambda_i^T \\ \text{such that} & \lambda_i \geq 0 \\ & \mathbf{V}_i^T \lambda_i = \mathbf{x}_{ui} \end{bmatrix} + \mathbf{q}_{ci}^T \mathbf{x}_{ci} \leq 0, \quad (27)$$

where  $i = 1, \dots, m$  and  $\lambda_i \in \mathbf{R}^k$  are the Lagrange multipliers corresponding to the inequality constraints  $\mathbf{V}_i \mathbf{q}_{ui} \leq \mathbf{v}_i$ . The separation of known and uncertain parameters in (26) prevents the term  $\mathbf{V}_i^T \lambda_i = \mathbf{x}_{ui}$  to unnecessarily constrain elements of  $\alpha_s$  to be equal to zero, which may mistakenly produce an infeasible optimization program. Because the dual and primal problem have zero duality gap, we can replace the primal with the dual and have the same optimal point. The second step is to merge (27) into (25) as follows:

$$\begin{aligned} & \text{minimize}_{\alpha_s, \lambda_i} && g(\alpha_s) \\ & \text{subject to} && \alpha_s^T \mathbf{F}^T \mathbf{F} \alpha_s + 2\mathbf{e}^T \mathbf{F} \alpha_s + \mathbf{e}^T \mathbf{e} \leq \tau_{\text{cont}}^2 n \\ & && \lambda_i^T \mathbf{v}_i + \mathbf{q}_{ci}^T \mathbf{x}_{ci} \leq 0, \\ & && \mathbf{V}_i^T \lambda_i = \mathbf{x}_{ui}, \\ & && \lambda_i \geq 0, \\ & && \mathbf{Z}_{ui} [\alpha_s^T, 1]^T = \mathbf{x}_{ui}, \\ & && \mathbf{Z}_{ci} [\alpha_s^T, 1]^T = \mathbf{x}_{ci}, \quad i = 1, \dots, m. \end{aligned} \quad (28)$$

for all  $\{\mathbf{F}, \mathbf{e}\} \in \mathcal{U}$ , where  $\mathbf{Z}_{ui}$  and  $\mathbf{Z}_{ci}$  are the matrices that map the uncertain and certain coefficients of the compliance vector to  $\mathbf{x}_{ui}$  and  $\mathbf{x}_{ci}$ .

### 3.2.2. Robust feasible solution of quadratic constraints

Satisfying winding temperature limits despite the uncertainty in Sec. 3.1 requires a robust feasible solution to the quadratic constraint in (16). The uncertainty in Sec. 3.1 manifests explicitly as uncertainty in  $\mathbf{e}$  but not in  $\mathbf{F}$  because the terms  $I_m$ ,  $b_m$ ,  $\mathbf{D}$ , and  $\mathbf{B}$  in the definition of  $\mathbf{F}$  are known. This means that the linear and constant terms of the quadratic constraint (16) are uncertain but not the quadratic term  $\alpha_s^T \mathbf{F}^T \mathbf{F} \alpha_s$ . Ben-Tal et al. [16] present the general robust feasible solution of quadratic constraints when an ellipsoid defines the uncertainty set of the coefficients. In this section, we derive the specific robust feasible solution for quadratic constraints with uncertain linear and constant terms. We define the parameter  $\mathbf{e}$  to be uncertain but bounded by the ellipsoid  $\mathbf{e} \in \mathcal{E}$ , where

$$\mathcal{E} = \{\mathbf{x} | (\mathbf{x} - \mathbf{e}_c)^T \mathbf{P}^{-1} (\mathbf{x} - \mathbf{e}_c) \leq 1\}, \quad (29)$$

$\mathbf{e}_c$  is the nominal value of  $\mathbf{e}$ , and the square root of the eigenvalues of  $\mathbf{P}$  define the length of the semi-axes of the ellipsoid. For example, a ball with radius  $r$  is an ellipsoid with  $\mathbf{P} = r^2 \mathbf{I}$ . The ellipsoid  $\mathcal{E}$  can be also written as  $\{\mathbf{x} | \mathbf{x}^T \mathbf{P}^{-1} \mathbf{x} - 2\mathbf{e}_c^T \mathbf{P}^{-1} \mathbf{x} + \mathbf{e}_c^T \mathbf{P}^{-1} \mathbf{e}_c \leq 1\}$  [25].

Our solution will follow similar steps as in Sec. 3.2.1. First, we formulate the uncertain program in (28) as a min-max program. Second, we rewrite the inner maximization program as a minimization program by writing the corresponding Lagrange dual. Third, we merge the inner and outer programs into a single convex semidefinite program (SDP). In our first step, we rewrite (28) as

$$\begin{aligned} & \min_{\alpha_s, \lambda_i} && g(\alpha_s) \\ & \text{s.t.} && \alpha_s^T \mathbf{F}^T \mathbf{F} \alpha_s + \begin{bmatrix} \max. & 2\mathbf{e}^T \mathbf{F} \alpha_s + \mathbf{e}^T \mathbf{e} \\ \mathbf{e} \\ \text{s.t.} & \mathbf{e} \in \mathcal{E} \end{bmatrix} \leq \tau_{\text{cont}}^2 n, \\ & && \lambda_i^T \mathbf{v}_i + \mathbf{q}_{ci}^T \mathbf{x}_{ci} \leq 0, \\ & && \mathbf{V}_i^T \lambda_i = \mathbf{x}_{ui}, \\ & && \lambda_i \geq 0, \\ & && \mathbf{Z}_{ui} [\alpha_s^T, 1]^T = \mathbf{x}_{ui}, \\ & && \mathbf{Z}_{ci} [\alpha_s^T, 1]^T = \mathbf{x}_{ci}, \quad i = 1, \dots, m. \end{aligned}$$

We rewrite the uncertain quadratic constraint in the previous equation as

$$\alpha_s^T \mathbf{F}^T \mathbf{F} \alpha_s - \begin{bmatrix} \min. & -\mathbf{e}^T \mathbf{I} \mathbf{e} - 2\alpha_s^T \mathbf{F}^T \mathbf{e} \\ \mathbf{e} \\ \text{s.t.} & \mathbf{e} \in \mathcal{E} \end{bmatrix} \leq \tau_{\text{cont}}^2 n. \quad (30)$$

The inner program is not convex as it minimizes a quadratic term with a negative definite Hessian, i.e., the matrix

$-\mathbf{I}$  is negative definite. The advantage of the ellipsoidal constraint and the Lagrange dual is that we can rewrite this non-convex program as a convex SDP. We write the corresponding Lagrangian as

$$L(\mathbf{e}, \lambda_q) = -\mathbf{e}^T \mathbf{I} \mathbf{e} - 2\boldsymbol{\alpha}_s^T \mathbf{F}^T \mathbf{e} + \lambda_q (\mathbf{e}^T \mathbf{P}^{-1} \mathbf{e} - 2\mathbf{e}_c^T \mathbf{P}^{-1} \mathbf{e} + \mathbf{e}_c^T \mathbf{P}^{-1} \mathbf{e}_c - 1).$$

The dual function is

$$g(\lambda_q) = \inf_{\mathbf{e}} L(\mathbf{e}, \lambda_q) = \begin{cases} g_1(\lambda_q) & (\lambda_q \mathbf{P}^{-1} - \mathbf{I}) \succcurlyeq 0, \lambda_q \geq 0, \\ & \lambda_q (\mathbf{e}_c^T \mathbf{P}^{-1} \mathbf{e}_c - 1) \in \mathcal{R}(\lambda_q \mathbf{P}^{-1} - \mathbf{I}) \\ -\infty & \text{otherwise} \end{cases}$$

where  $g_1(\lambda_q) = \lambda_q (\mathbf{e}_c^T \mathbf{P}^{-1} \mathbf{e}_c - 1) - (\mathbf{F} \boldsymbol{\alpha}_s + \lambda_q \mathbf{P}^{-1} \mathbf{e}_c)^T (\lambda_q \mathbf{P}^{-1} - \mathbf{I})^{-1} (\mathbf{F} \boldsymbol{\alpha}_s + \lambda_q \mathbf{P}^{-1} \mathbf{e}_c)$ . The dual and primal have zero duality gap, even if the primal (30) is non-convex [25, p. 654]. Using the slack variable  $\gamma$  and the dual function, we rewrite (30) as

$$\boldsymbol{\alpha}_s^T \mathbf{F}^T \mathbf{F} \boldsymbol{\alpha}_s + \begin{bmatrix} \min. & \gamma \\ \text{s.t.} & -g_1(\lambda_q) \leq \gamma \\ & \lambda_q \geq 0, \\ & (\lambda_q \mathbf{P}^{-1} - \mathbf{I}) \succcurlyeq 0 \end{bmatrix} \leq \tau_{\text{cont}}^2 n. \quad (31)$$

Using the Schur complement we include (31) into (28) as follows

$$\begin{aligned} \min. & \quad g(\boldsymbol{\alpha}_s) \\ \text{s.t.} & \quad \boldsymbol{\alpha}_s^T \mathbf{F}^T \mathbf{F} \boldsymbol{\alpha}_s + \gamma \leq \tau_{\text{cont}}^2 n, \\ & \quad \begin{bmatrix} \mathbf{I} - \lambda_q \mathbf{P}^{-1} & \mathbf{F} \boldsymbol{\alpha}_s + \lambda_q \mathbf{P}^{-1} \mathbf{e}_c \\ (\mathbf{F} \boldsymbol{\alpha}_s + \lambda_q \mathbf{P}^{-1} \mathbf{e}_c)^T & \lambda_q (\mathbf{e}_c^T \mathbf{P}^{-1} \mathbf{e}_c - 1) - \gamma \end{bmatrix} \succcurlyeq 0, \\ & \quad \lambda_q \geq 0, \\ & \quad \boldsymbol{\lambda}_i^T \mathbf{v}_i + \mathbf{q}_{ci}^T \mathbf{x}_{ci} \leq 0, \\ & \quad \mathbf{V}_i^T \boldsymbol{\lambda}_i = \mathbf{x}_{ui}, \\ & \quad \boldsymbol{\lambda}_i \geq 0, \\ & \quad \mathbf{Z}_{ui} [\boldsymbol{\alpha}_s^T, 1]^T = \mathbf{x}_{ui}, \\ & \quad \mathbf{Z}_{ci} [\boldsymbol{\alpha}_s^T, 1]^T = \mathbf{x}_{ci}, \quad i = 1, \dots, m, \end{aligned} \quad (32)$$

with optimization variables  $\boldsymbol{\alpha}_s$ ,  $\lambda_q$ , and  $\boldsymbol{\lambda}_i$ . The solution of this convex optimization program is a robust feasible solution to the uncertain program (24). Note that while (24) has an infinite number of constraints, (32) has a finite number and can be solved directly by existing numerical solvers.

### 3.3. Selecting the objective function

Any spring that represents a feasible point of (32) satisfies the actuator constraints despite uncertainty. Among

feasible points, we can select the robust solution that minimizes one or a combination of the following design objectives: motor energy consumption, motor torque, and motor velocity. The recommendations in Sec. 4 explain why these are relevant objectives for the design of series springs for quasi-direct drives. This section shows how  $g(\boldsymbol{\alpha}_s)$  can represent these objectives and yield a convex optimization program.

#### 3.3.1. Minimizing motor energy consumption

The sum of winding Joule heating,  $\tau_m^2 k_m^{-2}$ , and the rotor mechanical power,  $\tau_m \dot{q}_m$ , represents most of the energy consumption of an electric motor in a backdrivable SEA. When the motor and electric drive allow actuation and regeneration in the four quadrants of the speed and torque axes, the energy consumption is

$$E_m = \int_{t_0}^{t_f} (\tau_m^2 k_m^{-2} + \tau_m \dot{q}_m) dt, \\ = \int_{t_0}^{t_f} (\tau_m^2 k_m^{-2} + b_m \dot{q}_m^2 - \tau_1 \dot{q}_1) dt, \quad (33)$$

where we assume that the motion is periodic, i.e., kinematics and kinetics are the same at  $t_0$  and  $t_f$ . A previous version of our work [11] explains in detail the derivation of (33). Using numerical integration and the definitions in (11) and (14), we approximate the integration in (33) as

$$E_m \approx \left( \boldsymbol{\tau}_m^T \boldsymbol{\tau}_m k_m^{-2} + b_m \dot{\mathbf{q}}_m^T \dot{\mathbf{q}}_m - \boldsymbol{\tau}_1^T \dot{\mathbf{q}}_1 \right) \Delta t \\ = \boldsymbol{\alpha}_s^T \mathbf{G} \boldsymbol{\alpha}_s + \mathbf{h} \boldsymbol{\alpha}_s + w \quad (34)$$

where

$$\mathbf{G} = \left( \mathbf{F}^T \mathbf{F} k_m^{-2} + b_m \mathbf{B}^T \mathbf{B} \right) \Delta t, \\ \mathbf{h} = (2\mathbf{e}^T \mathbf{F} k_m^{-2} + 2b_m \mathbf{a}^T \mathbf{B}) \Delta t, \\ w = (\mathbf{e}^T \mathbf{e} k_m^{-2} + b_m \mathbf{a}^T \mathbf{a} - \boldsymbol{\tau}_1^T \dot{\mathbf{q}}_1) \Delta t.$$

The optimization program (32) is convex using (34) as the objective function; the objective function has a positive semidefinite quadratic form (the matrix  $\mathbf{G}$  is positive semidefinite; it is the sum of the Gramian matrices of  $\mathbf{F}$  and  $\mathbf{B}$ ) and the constraints are all intersections of convex sets such as polytopes or positive semidefinite cones.

#### 3.3.2. Minimizing motor torque or velocity

Using (14) or (11), we can write any vector norm of motor torque or motor velocity as the objective function,  $g(\boldsymbol{\alpha}_s)$ . For example, in Sec. 4, we will minimize the 2-norm of the torque or velocity vector multiplied by  $n^{-1/2}$  to minimize the RMS torque or RMS velocity. In both cases the objective function is a vector norm composed by an affine function of  $\boldsymbol{\alpha}_s$  and therefore (32) is convex [25, p. 79]. Minimizing motor velocity can be of interest for a system with a high reduction ratio. The higher the reduction ratio the more difficult is to model the energy losses



in the transmission. This is because friction, damping, hysteresis, non-backdrivability, and backlash in the transmission become more dominant as the number of stages in the transmission or reduction ratio increase. For those cases, minimizing the motor speed minimizes the motion of the motor, which indirectly reduces the difficult-to-model transmission losses.

### 3.3.3. Minimizing peak power

Traditionally, the size and mass of the electric motor have been considered proportional to the peak power of the motor; thus, minimizing peak power is a typical objective for the co-design of series springs and electric motors [11, 29]. Minimizing peak power is beneficial when the selection of the motor is part of the design. However, for a given motor, minimizing energy consumption is more beneficial than minimizing peak power as long as the motor torque and motor speed constraints are satisfied. In our framework, if the motor has not been selected, we can solve the feasibility problem of (32) to reduce size and mass of the system, i.e., we can check if (32) is robust-feasible for a given motor and iterate the process with progressively smaller motors. Convex optimization solvers can find a global solution of our optimization program within seconds, which speeds-up the iteration process.

## 4. Case Studies: Robust-Feasible Robotic Ankles

In the following case studies, we design multiple optimal series springs for the robotic ankle of a hypothetical 65 kg user walking on level ground. Our purpose is to provide working examples of our methodology, evaluate the performance of robust feasible designs, and provide insight into the selection of different objective functions, as defined in Sec. 3.3. The mean kinematics and kinetics of the ankle joint during human walking, as reported in [32] and discretized in  $n = 1000$  samples, defined the nominal reference  $\mathbf{q}_1$  and  $\boldsymbol{\tau}_1$ . We used the variance of the kinematics and kinetics in [32] to define the uncertainty in the load kinematics and kinetics, i.e.,  $\varepsilon_{\tau_{1m}}$ ,  $\varepsilon_{\tau_{1a}}$ ,  $\varepsilon_{\dot{q}_{1m}}$ , and  $\varepsilon_{\dot{q}_{1a}}$  (Table 1). The uncertainty in the load kinetics and kinematics can be interpreted as the result of the interplay between the human and the wearable robot or admissible error in the controller design. Changes in user mass are proportional to the multiplicative uncertainty in load torque,  $\varepsilon_{\tau_{1m}}$ . In our case studies, the load torque had a multiplicative uncertainty of 15% (Table 1). This can be interpreted as 15% admissible change in user mass.

We designed the optimal springs for the ankle joints of the second generation of the robotic leg at the University of Texas at Dallas (UTD) [18, 33] and the Open Source Leg (OSL) at the University of Michigan [20, 22]. The ILM85x26 brushless motor from TQ-RoboDrive coupled with a 22:1 geared transmission powered the UTD leg [18] and the ActPack v0.2b brushless motor from Dephy Inc. coupled with a 50:1 belt transmission powered the OSL

Table 1: Uncertainty based on the variance in [31, 32].  $\varepsilon_{\alpha_{sa}}$ ,  $\tau_{unc}$ ,  $\varepsilon_{\tau_{1m,a}}$ ,  $\varepsilon_{\dot{q}_{1m,a}}$  refer to the uncertainty in the spring compliance, the modeling torque, the load torque (multiplicative and additive), and the load velocity (multiplicative and additive), respectively. Sec. 3.1 describes in detail all sources of uncertainty.

Uncertainty	Units
$\varepsilon_{\alpha_{sa}}$	$\pm 20\%$ of linear spring compliance
$\tau_{unc}$	$\pm 260$ mN·m (10% motor rated torque)
$\varepsilon_{\tau_{1m}}$	$\pm 0.15$
$\varepsilon_{\tau_{1a}}$	$\pm 4.42$ N·m (10% RMS load torque)
$\varepsilon_{\dot{q}_{1m}}$	$\pm 0.2$
$\varepsilon_{\dot{q}_{1a}}$	$\pm 0.7440$ rad/s (20% RMS load velocity)

[22] (Table 2). The UTD leg is of interest because its ankle actuator has been proven to satisfy our definition of a quasi-direct drive [18]. Likewise, for the normative ankle joint moments, the OSL leg has also been shown to be backdrivable [31]. Furthermore, the excellent documentation of the OSL leg’s actuators [20] provides a different perspective to our case studies, while its open-source nature can help our analysis reach to a broader audience. Our expression of energy consumption (33) neglects the energy dissipated by Coulomb friction. This assumption may be less accurate for the OSL since its transmission dynamics have not been as well-characterized as the UTD leg.

We used the values of  $I_m$  and  $b_m$  experimentally validated in [18, 20, 22] and the rest of the parameters from the manufacturer’s datasheets. The values of uncertainty in manufacturing of the spring matched the variance of stiffness for the manufactured rotational springs in [31]. Tables 2 and 1 illustrate the nominal and uncertain parameters of the SEA and the task. In all simulations, CVX (Matlab Software for Disciplined Convex Programming) executed the solvers Mosek [34] or Gurobi [35] with precision settings `cvx_precision default`. The optimization program (32) is convex and can be solved in polynomial time (e.g., with  $n = 1000$ , a Windows 10 PC with an Intel i7 - 9700 processor computed 8000 solutions in less than 1.5 s each). The source code for these simulations is available as a capsule in CodeOcean, with the corresponding hyperlink available in the online version of this article.

The robust-feasible solution modifies the torque-elongation profile of the spring depending on whether or not the actuator constraints for the nominal case are active. Active constraints are inequality constraints where the inequality is satisfied at the boundary, i.e., the optimal solution  $x^*$  satisfies  $f(x^*) \leq 0$  at  $f(x^*) = 0$  [25]. Inactive constraints represent inequality constraints satisfied with some slack, i.e.,  $f(x^*) + a < 0$  for some  $a \in \mathbb{R}_{++}$ . In Sec. 4.1, we use energy consumption and RMS motor velocity as the optimization objectives to analyze nominal solutions when all the constraints are inactive. Sec. 4.2 analyzes active constraints by minimizing RMS motor torque.

Table 2: Simulation parameters for the ILM85x26 brushless motor from TQ-RoboDrive and the ActPack v0.2b from Dephy Inc. The values are in the q-axis using the direct quadrature transformation in [21]. The values of  $I_m$  and  $b_m$  match the experimental results for the reflected inertia and damping reported in [18, 20, 22]. For example, [18] reports the inertia of the ILM85x26 at the load side to be  $0.0696 \text{ kg}\cdot\text{m}^2$ , which is equivalent to an inertia of  $1.438 \text{ kg}\cdot\text{cm}^2$  at the rotor side with a 22:1 transmission ratio.

Motor parameter	ILM85x26	ActPack
Torque constant, $k_t$ (N·m/A)	0.24	0.14
Winding resistance, $R_m$ (mΩ)	323	186
Continuous torque, $\tau_{\text{cont}}$ (N·m)	2.6	1.1
Motor inertia, $I_m$ (kg·cm <sup>2</sup> )	1.438	1.2
Gear ratio, $r$	22	50
Viscous fric., $b_m$ (mN·m·s/rad)	0.861	0.16
Max. torque, $\tau_{\text{max}}$ (N·m)	8.3	4.02
Max. velocity, $\dot{q}_{\text{max}}$ (rpm)	1500	2455
Voltage (V)	48	36

#### 4.1. Robust feasibility in the case of inactive constraints

For inactive constraints with enough slack the optimal springs from the nominal and robust-feasible solutions will be the same. In this scenario, the purpose of the robust-feasible solution is to certify that the nominal solution will be feasible despite of the uncertainty. Table 3 summarizes the optimal solutions using energy consumption and RMS velocity as the objective functions. The dual variables of our numerical solution to (32) define what constraints are active or inactive.

Our solutions to (32) are strictly feasible [25]. Thus, from Slater’s condition, strong duality holds for our convex formulation, i.e., there is no gap between the dual and primal optimal [25]. Strong duality allows us to analyze which constraints are active for our optimal solution using complementary slackness [25]; if the optimal dual variable  $\lambda_i$  associated with an inequality constraint is  $\lambda_i > 0$  then our  $i$ th constraint is active [25]. The dual variables are also useful for a sensitivity analysis. The higher the dual variable the more sensitive is the optimal with respect to perturbations of that constraint. Table 3 includes the value of the maximum dual variable and its index  $i$  within the set of inequality constraints. Given the order in which we programed the constraints and  $n$ , for  $0 \leq i \leq 1998$ ,  $1999 \leq i \leq 3996$ ,  $3997 \leq i \leq 7992$ , and  $i = 7993$  the inequality represents an elongation, torque, torque-velocity, and RMS torque constraint, respectively. From the values of the maximum dual variables and the slack in the left and right hand sides of the associated inequality constraints, we state that minimizing energy consumption and RMS velocity lead to optimal designs with inactive constraints.

##### 4.1.1. Minimizing energy consumption

Our global-optimal nonlinear and linear springs reduced energy consumption by 3.5% (UTD) and 7.8% (OSL) with respect to the rigid case (Table 3). To contextualize this energy reduction it is fundamental to understand how series

Table 3: Simulation results using energy, RMS velocity, and RMS torque as the objective function. The reported stiffness corresponds to the stiffness of the optimal linear spring. For comparison, we show the energy consumption and RMS velocity for an actuator without spring (rigid) and the linear and nonlinear optimal springs. LHR and RHS refer to the left-hand and right-hand sides of the inequalities at constraint  $i$ . The dual variables correspond to the nominal program to design the optimal linear spring.

		UTD	OSL
Energy (J)	nonlin.	38.14	23.62
	linear	38.23	23.62
	rigid	39.51	25.61
Max. dual variable $\lambda_i$		3.62e-08	1.17e-07
Index dual variable $i$		6539	7561
LHS/RHS at constraint $i$		21.7/46.9	-28.6/5.6
Stiffness nominal (N·m/rad)		197.80	176.79
Stiffness robust (N·m/rad)		197.92	176.86
RMS velo. (rad/s)	nonlin.	21.52	44.84
	linear	23.72	49.41
	rigid	35.84	74.66
Max. dual variable $\lambda_i$		1.38e-07	1.1e-07
Index dual variable $i$		999	3426
LHS/RHS at constraint $i$		0.38/1.6	-0.01/2.1
Stiffness nominal (N·m/rad)		285.3	285.3
Stiffness robust (N·m/rad)		285.29	285.3
RMS torque (N·m)	nonlin.	1.87	0.87
	(nominal) linear	1.88	0.88
RMS torque (N·m)	nonlin.	1.89	0.90
	(robust-feasible) linear	1.90	0.91
RMS torque rigid (N·m)		1.94	0.98
Max. dual variable $\lambda_i$		1.23e-03	1.44e-03
Index dual variable $i$		6539	6537
LHS/RHS at constraint $i$		46.8/46.8	42.25/42.35
Stiffness nominal (N·m/rad)		91.53	123.5
Stiffness robust (N·m/rad)		131.74	183.3

springs affect motor speed and torque. We can interpret energy consumption as a weighted sum of the square of RMS torque and square of RMS velocity, as shown in (34). In this weighted sum, the RMS torque is a dominant term relative to RMS velocity, i.e., the heat losses are higher than the viscous losses for quasi-direct drives where the motor torque is high and thus dominant [36]. In a quasi-direct or direct drive, the only chance for a series spring to modify torque is to modify the motor position such that the viscous and inertial torques of the motor modify the load torque. However, the inertial and viscous torques in these case studies are negligible, e.g., the RMS motor torque for the UTD leg without series spring was 1.94 N·m out of which 0.026 N·m and 0.07 N·m are RMS viscous and inertial torques. As a consequence, the energy savings from using series springs was less than 8% in both cases.

#### 4.1.2. Minimizing RMS velocity

Our global-optimal nonlinear and linear springs reduced RMS velocity by 40% and 33.8% with respect to the rigid case (Table 3). The percentage of reduction and the optimal springs were the same for the OSL and UTD. The definition of motor velocity in (11) is independent of the motor parameters and all the constraints were inactive; hence, the optimal elongation-torque profile of the spring was motor-agnostic. Minimizing motor velocity is beneficial to reduce the motion of the motor and indirectly minimize the energy consumption without an accurate energy model, e.g., SEAs with a high-ratio transmission. Modeling energy losses of SEAs with a high-ratio transmission requires an accurate estimate of friction and torques that make the system non-backdrivable. These effects are difficult to model as they depend in temperature and lubrication.

#### 4.2. Robust feasibility in the case of active constraints

Minimizing RMS torque produced optimal springs with active torque-speed constraints for the OSL and the UTD prosthetic legs (Table 3). When the optimal solution has at least one active actuator constraint, the constraint may become unfeasible in the presence of uncertainty. In such case, the robust-feasible and nominal solution will have different optimal series springs. Fig. 3 shows the optimal linear and nonlinear springs for the nominal and robust-feasible solutions. The linear and nonlinear springs reduced less than 3.7% (UTD) and 11.3% (OSL) of the RMS torque compared to the rigid case (Table 3). As discussed in Sec. 4.1.1, the motor torque is not sensitive to changes in stiffness. Thus, the robust-feasible and nominal solutions had a similar cost, i.e., there was a weak trade-off between optimality and robustness. However, the changes in stiffness between the robust-feasible and nominal solution had an important impact in the feasibility of motor speed as shown in Fig. 4.

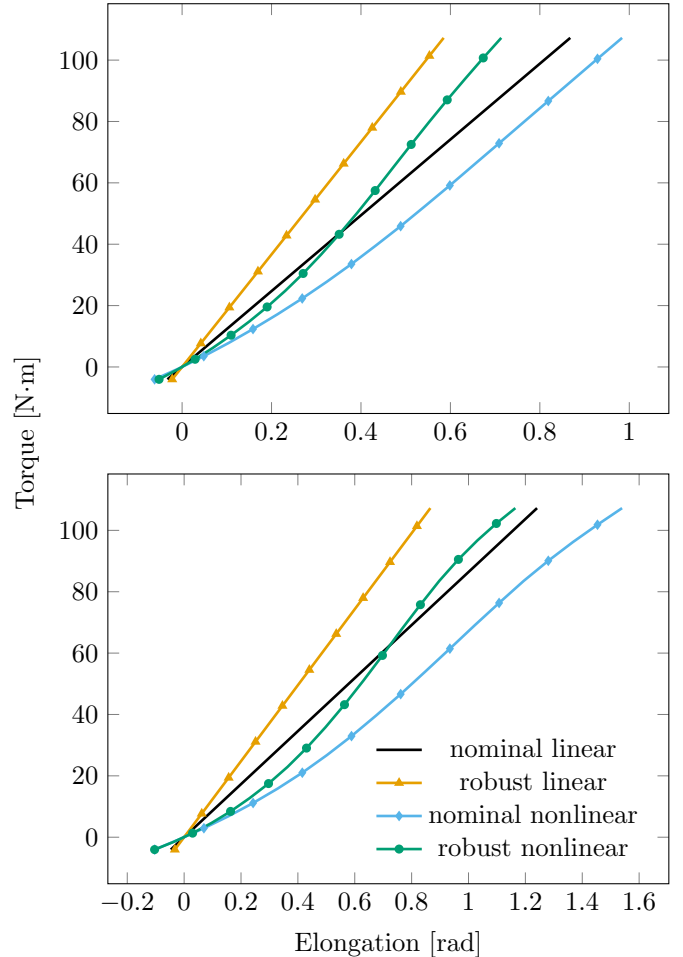


Figure 3: Series springs that minimize RMS motor torque for the ActPack (top) and ILM85x26 (bottom). We designed the optimal SEAs for the ankle joint of a 65 kg subject walking in level ground [32] with cadence of 105 steps per minute. The biomechanical data in [32] used a sample of 16 participants with average age of 25.6 years and standard deviation of 6.2 years. The robust feasible springs result from solving (32) using the uncertainty in Table 1.

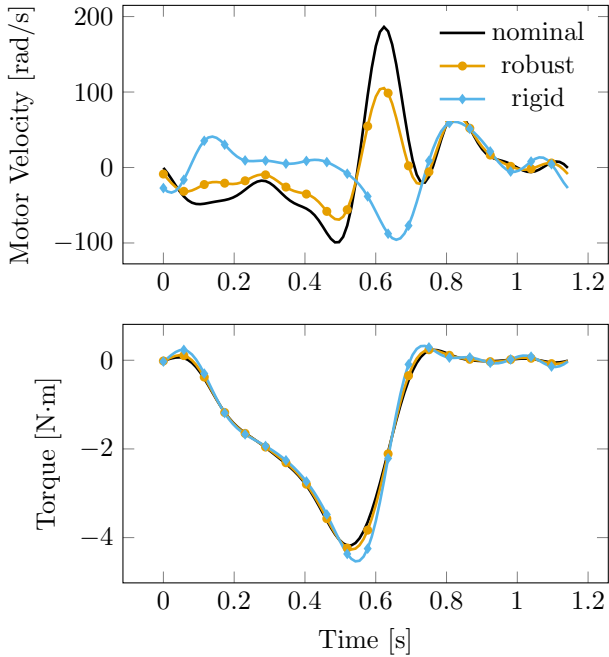


Figure 4: Angular velocity for the ILM85x26 motor using the linear spring that minimizes RMS torque (Fig. 3). The robust solution leads to a motor speed that satisfies the actuator constraints despite the uncertainty in Table 1. Notice that robustness did not imply a significant trade-off in terms of RMS torque reduction (nominal: 0.88 N·m, robust: 0.91 N·m, Table 3).

## 5. Discussion and Conclusion

We introduced a method for the optimal robust-feasible design of springs for SEAs. Our robust-feasible solution implies that the resulting SEA will satisfy all the actuator constraints of Sec. 2.3 despite any of the sources of uncertainty introduced in Sec. 3.1. Our formulation works for arbitrary motion of the load except when we design the spring to minimize energy consumption. Minimizing energy consumption requires the load motion to be periodic, e.g., walking, running, or a combination of those two. Our optimization framework can minimize any of the objective functions in Sec. 3.3. We showed that the optimal robust-feasible spring is equivalent to the optimal point of a convex program. The numerical solution of this program can be computed efficiently and reliably with plenty of commercial and open source solvers [25].

When minimizing energy consumption, we assume that the reference trajectory is periodic, the transmission has negligible backlash and Coulomb friction, and the energy losses in the transmission can be represented by viscous friction losses. These assumptions imply that our energy savings results apply for direct and quasi-direct drives. Our energy consumption analysis does not apply for SEAs with high-ratio transmissions where the system may not be backdrivable or friction is not negligible.

In general, the potential energy savings from serial elasticity are load and motor dependent. As discussed in Sec. 4.1.1, in applications where the inertial and viscous

torques are low and the system is backdrivable, the energy savings may not be enough to justify the mechanical complexity and mass inherent of the extra series spring, e.g., series elasticity reduced 3.5% (UTD) and 7.8% (OSL) of energy consumption for the prosthetic ankles in our case studies in Sec. 4.

As discussed in Sec. 4.1.2, the optimal spring that minimized RMS motor speed was motor agnostic. Compared to linear springs, nonlinear springs were only beneficial when minimizing motor RMS speed. The nonlinear springs minimized around 10% more RMS velocity than linear springs. For all the other cases, the linear and nonlinear springs had almost the same cost. Considering the uncertainty in the compliance of the final springs, we note that the manufactured spring can be different from the optimal and still guarantee a reduction of cost and feasibility of the constraints. Our future work will extend this formulation to optimize parallel springs and co-design elastic components, the electric motor, and the control algorithm.

## References

- [1] G. Pratt and M. Williamson, “Series Elastic Actuators,” in *Proc. 1995 IEEE/RSJ Int. Conf. Intell. Robot. Syst. Hum. Robot Interact. Coop. Robot.*, vol. 1. IEEE Comput. Soc. Press, 1995, pp. 399–406.
- [2] D. W. Robinson, “Design and Analysis of Series Elasticity in Closed-loop Actuator Force Control,” Ph.D. dissertation, Massachusetts Institute of Technology, 2000.
- [3] D. P. Losey *et al.*, “A Time-Domain Approach to Control of Series Elastic Actuators: Adaptive Torque and Passivity-Based Impedance Control,” *IEEE/ASME Trans. Mechatronics*, vol. 21, no. 4, pp. 2085–2096, 2016.
- [4] D. Rollinson *et al.*, “Design and Modeling of a Series Elastic Element for Snake Robots,” in *Dynamic Systems and Control Conference*, vol. 1. American Society of Mechanical Engineers, oct 2013, pp. 2–6.
- [5] J. Hurst and A. Rizzi, “Series compliance for an efficient running gait,” *IEEE Robot. Autom. Mag.*, vol. 15, no. 3, pp. 42–51, 2008.
- [6] A. Bicchi and G. Tonietti, “Fast and “Soft-Arm” Tactics,” *IEEE Robot. Autom. Mag.*, vol. 11, no. 2, pp. 22–33, 2004.
- [7] D. J. Braun *et al.*, “Robots Driven by Compliant Actuators: Optimal Control Under Actuation Constraints,” *IEEE Trans. Robot.*, vol. 29, pp. 1085–1101, 2013.
- [8] B. Vanderborght *et al.*, “Exploiting Natural Dynamics to Reduce Energy Consumption by Controlling the Compliance of Soft Actuators,” *Int. J. Rob. Res.*, vol. 25, no. 4, pp. 343–358, 2006.
- [9] A. Jafari, N. G. Tsagarakis, and D. G. Caldwell, “A novel intrinsically energy efficient actuator with adjustable stiffness (AwAS),” *IEEE/ASME Trans. Mechatronics*, vol. 18, no. 1, pp. 355–365, 2013.
- [10] E. Bolívar, S. Rezazadeh, and R. D. Gregg, “A General Framework for Minimizing Energy Consumption of Series Elastic Actuators With Regeneration,” in *ASME Dynamic Systems and Control Conference*, 2017, p. V001T36A005.
- [11] E. A. Bolívar Nieto, S. Rezazadeh, and R. Gregg, “Minimizing Energy Consumption and Peak Power of Series Elastic Actuators: a Convex Optimization Framework for Elastic Element Design,” *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 3, pp. 1334–1345, 2019.
- [12] E. A. Bolívar Nieto, S. Rezazadeh, T. Summers, and R. D. Gregg, “Robust optimal design of energy efficient series elastic actuators: Application to a powered prosthetic ankle,” in *IEEE Int. Conf. Rehabil. Robot. (ICORR)*, Aug 2019.
- [13] K. R. Embry *et al.*, “Modeling the kinematics of human locomotion over continuously varying speeds and inclines,” *IEEE*

- Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 12, pp. 2342–2350, 2018.
- [14] W. R. Brown and A. G. Ulsoy, “A Maneuver Based Design of a Passive-Assist Device for Augmenting Active Joints,” *J. Mechanisms Robotics*, vol. 5, no. 3, p. 031003, 2013.
- [15] —, “Robust design of Passive Assist Devices for multi-DOF robotic manipulator arms,” *Robotica*, vol. 35, no. 11, pp. 2238–2255, nov 2017.
- [16] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust Optimization*, ser. Princeton Series in Applied Mathematics. Princeton University Press, October 2009.
- [17] H. Asada and K. Youcef-Toumi, *Direct-Drive Robots: Theory and Practice*. The MIT Press, 1987.
- [18] T. Elery, S. Rezazadeh, C. Nesler, and R. D. Gregg, “Design and Validation of a Powered Knee-Ankle Prosthesis with High-Torque, Low-Impedance Actuators,” *IEEE Trans. Robot.*, 2020.
- [19] T. Verstraten *et al.*, “Modeling and design of geared DC motors for energy efficiency: Comparison between theory and experiments,” *Mechatronics*, vol. 30, pp. 198–213, 2015.
- [20] U. H. Lee, C.-W. Pan, and E. J. Rouse, “Empirical Characterization of a High-performance Exterior-rotor Type Brushless DC Motor and Drive,” in *IEEE/RSJ Int. Conf. Intell. Robot. Syst.* IEEE, 2019, pp. 8018–8025.
- [21] R. H. Park, “Two-reaction theory of synchronous machines generalized method of analysis-part I,” *Trans. Am. Inst. Electr. Eng.*, vol. 48, no. 3, pp. 716–727, jul 1929.
- [22] A. F. Azocar, L. M. Mooney, J.-f. Duval, A. M. Simon, L. J. Hargrove, and E. J. Rouse, “Design and clinical implementation of an open-source bionic leg,” *Nat. Biomed. Eng.*, vol. 4, no. 10, oct 2020.
- [23] T. Lenzi *et al.*, “Design, development, and testing of a lightweight hybrid robotic knee prosthesis,” *Int. J. Rob. Res.*, vol. 37, no. 8, pp. 953–976, 2018.
- [24] J. M. Hollerbach, I. W. Hunter, and J. Ballantyne, “A Comparative Analysis of Actuator Technologies for Robotics,” in *Robot. Rev.* 2, 1992, pp. 299–342.
- [25] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY: Cambridge University Press, 2004.
- [26] P. M. Wensing, A. Wang, S. Seok, D. Otten, J. Lang, and S. Kim, “Proprioceptive Actuator Design in the MIT Cheetah: Impact Mitigation and High-Bandwidth Physical Interaction for Dynamic Legged Robots,” *IEEE Trans. Robot.*, vol. 33, no. 3, pp. 509–522, jun 2017.
- [27] S. Rezazadeh and J. W. Hurst, “On the Optimal Selection of Motors and Transmissions for Electromechanical and Robotic Systems,” in *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2014, pp. 4605–4611.
- [28] S. K. Au and H. M. Herr, “Powered ankle-foot prosthesis,” *IEEE Robot. Autom. Mag.*, vol. 15, no. 3, pp. 52–59, 2008.
- [29] K. W. Hollander *et al.*, “An Efficient Robotic Tendon for Gait Assistance,” *J. Biomech. Eng.*, vol. 128, no. 5, p. 788, 2006.
- [30] M. Tucker, J. Olivier, A. Pagel, H. Bleuler, M. Bouri, O. Lambercy, J. del R Millán, R. Riener, H. Vallery, and R. Gassert, “Control strategies for active lower extremity prosthetics and orthotics: a review,” *J. Neuroengineering and Rehabilitation*, vol. 12, no. 1, 2015.
- [31] A. F. Azocar, L. M. Mooney, L. J. Hargrove, and E. J. Rouse, “Design and Characterization of an Open-Source Robotic Leg Prosthesis,” in *2018 7th IEEE Int. Conf. Biomed. Robot. Biomechatronics*, 2018, pp. 111–118.
- [32] D. A. Winter, “Biomechanical Motor Patterns in Normal Walking,” *J. Mot. Behav.*, vol. 15, no. 4, pp. 302–330, dec 1983.
- [33] T. Elery *et al.*, “Design and Benchtop Validation of a Powered Knee-Ankle Prosthesis with High-Torque, Low-Impedance Actuators,” in *IEEE Int. Conf. Robotics & Automation*, 2018.
- [34] M. Aps, “MOSEK Optimization Suite,” 2017. [Online]. Available: <http://docs.mosek.com/8.0/intro.pdf>
- [35] L. Gurobi Optimization, “Gurobi optimizer reference manual,” 2020. [Online]. Available: <http://www.gurobi.com>
- [36] S. Seok *et al.*, “Design Principles for Energy-Efficient Legged Locomotion and Implementation on the MIT Cheetah Robot,” *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 3, pp. 1117–1129, 2015.

## Appendix A. Direct method to constrain winding temperature

In this appendix, we write the winding temperature as a set of quadratic functions of spring compliance. Using (7), (8), (9), and the state variables  $x_h = (T_h - T_a)$  and  $x_w = (T_w - T_a)$ , we write the thermal dynamics in state space as

$$\begin{bmatrix} \dot{x}_h \\ \dot{x}_w \end{bmatrix} = \mathbf{A}_T \begin{bmatrix} x_h \\ x_w \end{bmatrix} + (\mathbf{B}_{T1}x_w + \mathbf{B}_{T2})i_m^2, \quad (\text{A.1})$$

where

$$\mathbf{A}_T = \begin{bmatrix} -(R_{ha} + R_{wh})(R_{ha}R_{wh}C_{ha})^{-1} & (R_{wh}C_{ha})^{-1} \\ (R_{wh}C_{wa})^{-1} & -(C_{wa}R_{wh})^{-1} \end{bmatrix},$$

$$\mathbf{B}_{T1} = \begin{bmatrix} 0 \\ R_{m@a}\alpha_{Cu}C_{wa}^{-1} \end{bmatrix}, \quad \mathbf{B}_{T2} = \begin{bmatrix} 0 \\ R_{m@a}C_{wa}^{-1} \end{bmatrix}.$$

The term  $\mathbf{B}_{T1}x_w i_m^2$  in (A.1) makes this system nonlinear. The copper’s temperature coefficient of resistance,  $\alpha_{Cu}$ , is positive, which implies that any  $T_w$  higher than  $T_a$  will increase  $R_m$ . The higher  $R_m$  the more heat is dissipated per unit of current, as seen in (8). We can generate a linear approximation of (A.1) assuming that  $R_m$  is constant and corresponds to the worst-case scenario, where the winding is at its maximum admissible temperature. Thus, we linearize (A.1) replacing (9) with

$$R_m = R_{m@a}(1 + \alpha_{Cu}(T_{\max} - T_a)),$$

where  $T_{\max}$  is the maximum admissible winding temperature. Depending on the manufacturer,  $T_{\max} \approx 150^\circ\text{C}$ . This assumption will over-estimate the winding temperature, especially when the winding is close to ambient temperature. However, the model provides the same steady-state winding temperature in the case that  $T_w = T_{\max}$ , as seen in Fig. A.5. The resulting model is linear with respect to  $i_m^2$  as follows:

$$\begin{bmatrix} \dot{x}_h \\ \dot{x}_w \end{bmatrix} = \mathbf{A}_T \begin{bmatrix} x_h \\ x_w \end{bmatrix} + \mathbf{B}_T i_m^2, \quad (\text{A.2})$$

where  $\mathbf{B}_T = \mathbf{B}_{T1}(T_{\max} - T_a) + \mathbf{B}_{T2}$ . We sample the winding and housing temperatures at  $n$  discrete times by defining the vector  $\mathbf{x} \in \mathbb{R}^{2n}$  as  $\mathbf{x} = [(x_h)_1, (x_w)_1, \dots, (x_h)_n, (x_w)_n]$ , where  $(x_{(h/w)})_i$  refers to the  $i$ th sample of the housing or winding temperature. Using  $\dot{\mathbf{x}} \approx \mathbf{D}_1\mathbf{x}$ , we rewrite (A.2) in discrete-time as

$$\mathbf{x} = \mathbf{A}_{Td}i_m^2, \quad (\text{A.3})$$

where  $\mathbf{D}_1 \in \mathbb{R}^{2n \times 2n}$  is the first-order derivative operator in matrix form [11];  $i_m^2 \in \mathbb{R}^n$ ; and

$$\mathbf{A}_{Td} = (\mathbf{D}_1 - \text{diag}(\mathbf{A}_T))^{-1}\text{diag}(\mathbf{B}_T),$$

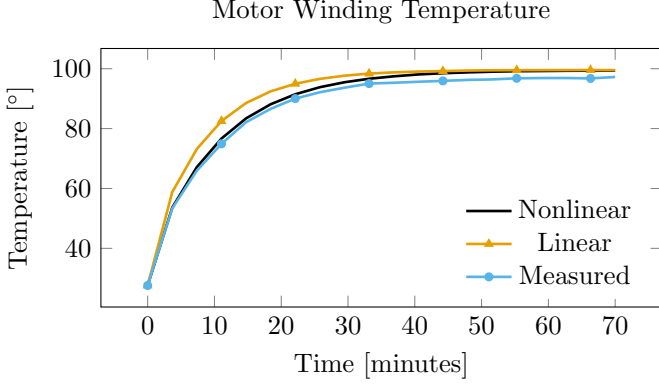


Figure A.5: Measured winding temperature for the quasi-direct motor ‘ActPack’ in [20]. We used the parameters in [20] to estimate the winding temperature using the nonlinear model (Eq. A.1) and the simplified linear model (Eq. A.2). The maximum percent error and its location depend on ambient temperature, the current provided to the motor, and the thermal time constant of the actuator. For the conditions of this simulation, the maximum percent error between the measured temperature and the nonlinear model is 2.9% at 10 minutes, and between the linear and nonlinear model is 8.8% at 8.8 minutes. We performed the simulations for 70 minutes.

where

$$\text{diag}(\mathbf{B}_T) = \begin{bmatrix} \mathbf{B}_T & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathbf{B}_T \end{bmatrix}$$

is a  $\mathbb{R}^{2n \times n}$  non-square matrix. With (A.3), (14), and  $\tau_m = k_t i_m$  we express the winding temperature as a function of compliance:

$$\mathbf{x}_i = \frac{1}{k_t^2} \sum_{j=1}^n (\mathbf{A}_{Td})_{ij} (\mathbf{e}_j^2 + 2\mathbf{e}_j \mathbf{F}_{j*}^T \boldsymbol{\alpha}_s + \boldsymbol{\alpha}_s^T \mathbf{F}_{j*}^T \mathbf{F}_{j*} \boldsymbol{\alpha}_s).$$

Thus, the following convex quadratic inequalities constrain the winding temperature to be less than the maximum admissible value

$$\frac{1}{k_t^2} \sum_{j=1}^n (\mathbf{A}_{Td})_{ij} (\mathbf{e}_j^2 + 2\mathbf{e}_j \mathbf{F}_{j*}^T \boldsymbol{\alpha}_s + \boldsymbol{\alpha}_s^T \mathbf{F}_{j*}^T \mathbf{F}_{j*} \boldsymbol{\alpha}_s) \leq T_{\max}, \quad (\text{A.4})$$

where  $i$  refers to the states that match the winding temperatures, i.e.,  $\mathbf{x}_i = (x_w)_j$ ,  $j = 1, \dots, n$ .