

# Towards the Unification of System Design and Motion Synthesis for High-Performance Hopping Robots

Eric Ambrose<sup>1</sup>, Wen-Loong Ma<sup>1</sup> and Aaron D. Ames<sup>2</sup>.

**Abstract**—Robotic hopping requires high performance and precision, due to its extreme interactions with the environment. Designing a system that will perform optimally, or even stably, for this motion primitive is a significant challenge. In previous work, it was shown that designing a robot with two springs (one in series and one in parallel with the actuator) could dramatically improve performance. However, selecting these springs was an intricate process since their dynamics were tightly coupled, and accomplished through trial and error. This work presents a general optimization framework for interconnected systems that designs the time-based hopping motion, while also designing the shape of nonlinear springs on the robot to yield efficient hopping. Utilizing this method, hopping motions and spring designs were generated simultaneously and experimentally verified on a novel hopping robot.

## I. INTRODUCTION

The study of system design optimization has a long and rich history in both academia and industry. Shape design in aerodynamics [22], [18] using computational fluid dynamics (CFD) focuses on designing aircraft shape against airflow from the front and lateral directions. Logic optimization was used by the automated electronic design industry in circuit design given volume constraints for optimal manufacturing cost [8]. Structural optimization [15] designs mechanical systems according to metrics such as minimal weight or increased strength according to specific application scenarios. Lastly, the genetic algorithm uses evolution of population to guide the design of static components for complex environments [21].

One such example of design optimization study was from the realm of industrial robotic manipulation tasks, which could greatly benefit from even small improvements to efficiency [26]. Here, springs were placed in parallel to joint motors and optimized alongside the motion trajectories to boost efficiency. In this case, researchers were able to find a closed-form solution for the optimal spring parameters in terms of the trajectories, which allowed for the optimization to again be considered as a classic trajectory optimization problem. In another example from the field of robotic hopping, model-free design optimization techniques were used to streamline the design iteration process [25]. This method involved hopping experiments, using a robot which could have certain design parameters changed quickly between separate tests rather than using prior simulation to yield a single final design.

\*This work is supported by Disney Research and Development.

<sup>1</sup>E. Ambrose and W. Ma are with the department of mechanical engineering, <sup>2</sup>A. D. Ames is with the faculty of the department of Control and Dynamical Systems, California Institute of Technology, Pasadena, CA, 91125. eambrose, wma, ames@caltech.edu

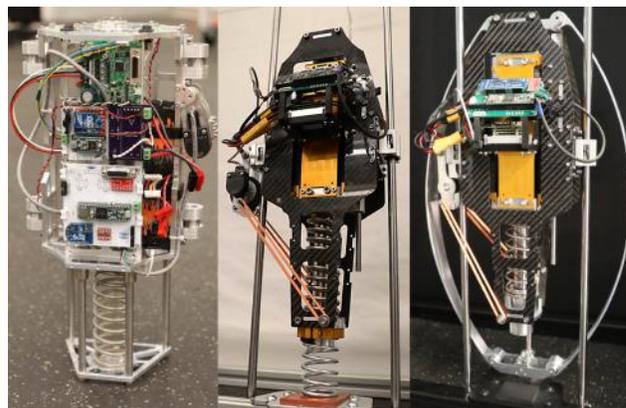


Fig. 1: Evolution of the moving-mass hopping robot. Mark I (left, [3]), Mark II (center, [2]), and now Mark III (right) is modified to feature the curved-springs for this work.

However, most traditional system design optimization problems often craft the design variables for a specific task. The goal of this paper is to make a first step towards simultaneously generating high-performance motions for robotic systems and finding their optimal design parameters. This is achieved through controlling and designing the hopping robot in Fig. 1 (right side). A related example is the leg design of quadrupedal robots such as [9].

Previous studies of designing motions and control laws for compliant bipeds [24] and hoppers [10] have demonstrated the effectiveness of model-based approaches using collocation-based optimization [17]. We have also explored different design methodologies of spring-loaded hoppers [3], and hopping with parallel elasticity [2]. These first two hopping robots can be seen in the left and center positions of Fig. 1. In this paper, we further exploit the use of high-performance mechanisms via curved plate springs to enhance performance. Plate springs have the advantage of being easy to design for a variety of nonlinear stiffness profiles, which can provide for improved energy storage and acceleration profiles [7], [20]. However, the nonlinear and compliant nature of these springs cannot be easily characterized by a traditional relationship, such as Hooke's law, and is thus a memoryless system. Instead, beam theory [5] must be used to develop the spatial-domain dynamical equations that govern the springs. To incorporate this with traditional time-domain dynamics of the torso, which are largely used to characterize robotic systems [11] and legged locomotion [14], we take inspiration from interconnected dynamical systems [4]. Through shared variables of spring force and length, the torso and the spring subsystems are dynamically coupled. We represent this formulation as an *interconnected time-spatial*

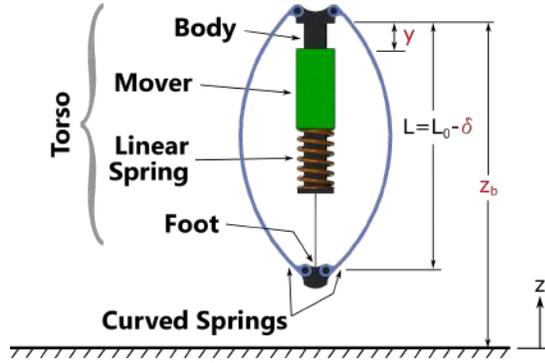


Fig. 2: The system's configuration coordinates, with a look at the core mechanism features.

*dynamical system*. A similar idea was used to decouple quadrupedal dynamics as two coupled bipedal subsystems [23]. Further, we can configure the curved spring design as decision variables for an optimization algorithm. This allows the optimization to concurrently find the solution to the interconnected dynamical system and the optimal spring design parameters. The end result is a custom-made hopping robot driven by the full-body dynamics and optimal design, which can hop stably and with greater energy efficiency.

Our main contributions are twofold. First, we propose a time-spatial dynamic modeling of the hopping robot that utilizes the curved springs in Fig. 1. Based on this, a generic optimization formulation is presented to solve for a solution to this dynamical system with a corresponding hopping motion, and find the optimal design variables for the curved springs. Secondly, the construction of a hopping robot, on which experiments are conducted and stable hopping is achieved. Therefore, we demonstrate the complete process of system and controller design, and experimental validation.

We organize the paper as follows. In Sec. II, we construct the dynamic model of this hopping robot as an interconnection between the traditional time-domain dynamics of the torso and the spatial-domain dynamics of the curved springs. In Sec. III we lay out an optimization algorithm to find the solution to this dynamical system, which is a periodic orbit representing a hopping motion, and the optimal value of the design variables. Lastly, we experimentally show the desired motion of this hopper that was designed in Sec. IV. Sec. V discusses the results and future work.

## II. INTERCONNECTED DYNAMICS

The dynamics of the hopping machine in Fig. 2 are composed of two main parts: the time-domain dynamics of the “torso”, whose electric motors’ controller is our *control target*; and the spatial-domain dynamics of a curved spring, whose physical parameters are our *design target*. The two subsystems are interconnected through the shared spring force and length variables,  $F_s$  and  $L$ . We will introduce each subsystem first, then show the interconnected full-body dynamics. Note that we used the following notations to distinguish the two types of derivative:

$$\dot{\square} \triangleq \frac{\partial \square}{\partial t}, \quad \square' \triangleq \frac{\partial \square}{\partial s}.$$

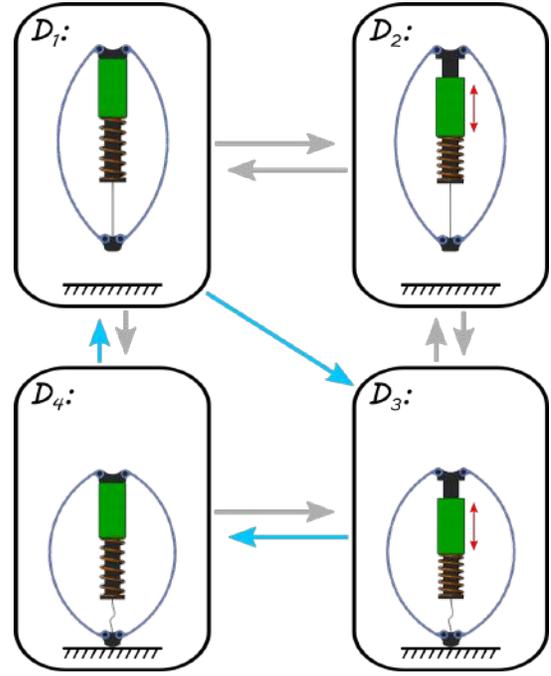


Fig. 3: Directed graph for the hybrid hopping system. Each phase of dynamics can be represented by a interconnected system as given by (13)(14). The blue arrows indicate the order of phases followed in this work.

### A. Time-domain dynamics of the torso

As Fig. 2 shows, the torso contains all of the robot other than the curved springs. Hence, as detailed in [2], we can obtain the torso's *time-domain* dynamics with configuration coordinates  $q = (z_b, y, \delta)^\top \in \mathbb{R}^3$  and using the Euler-Lagrangian methods to obtain the equations of motion. Without considering any constraint, we have

$$M(q)\ddot{q} + H(q, \dot{q}) + B_s F_s = Bu + J_v^\top(q)\lambda_v, \quad (1)$$

with

$$M(q) = \begin{bmatrix} M_0 & -M_m & M_f \\ -M_m & M_m & 0 \\ M_f & 0 & M_f \end{bmatrix}, \quad H(q, \dot{q}) = \begin{bmatrix} c_b \dot{z}_b + M_0 g \\ F_p - M_m g \\ M_f g \end{bmatrix},$$

and the actuation matrices  $B_s^\top = (0, 0, 1)$ ,  $B^\top = (0, 1, 0)$ .  $M_m$ ,  $M_f$ , and  $M_0$  are the mass of the mover, foot, and total robot.  $c_b$  is the damping of the body as it moves vertically. The internal spring creates a standard linear extension force given by  $F_p = k_p y + c_p \dot{y}$ , where  $k_p$  and  $c_p$  are the stiffness and damping coefficients, respectively.  $F_s$  is the extension force of the curved springs being examined in this work.

There exist four different dynamic domains, denoted by  $\mathcal{D}_v$  with  $v \in \{1, 2, 3, 4\}$ . Each domain has a holonomic constraint representing certain locking mechanisms. Fig. 3 shows the domain map for this hybrid system. The hopping motion followed here contains three continuous-time domains: one aerial domain,  $\mathcal{D}_1$ , and two ground domains,  $\mathcal{D}_3$  and  $\mathcal{D}_4$ . The two ground domains are distinguished by whether the mover is in contact with the body. During each domain, a holonomic constraint is used to represent the contacts with internal components and the ground. In this case, these are

$$h_1(q) \triangleq (y, \delta)^\top, \quad h_3(q) \triangleq (z_b + \delta), \quad h_4(q) \triangleq (y, z_b + \delta)^\top. \quad (2)$$

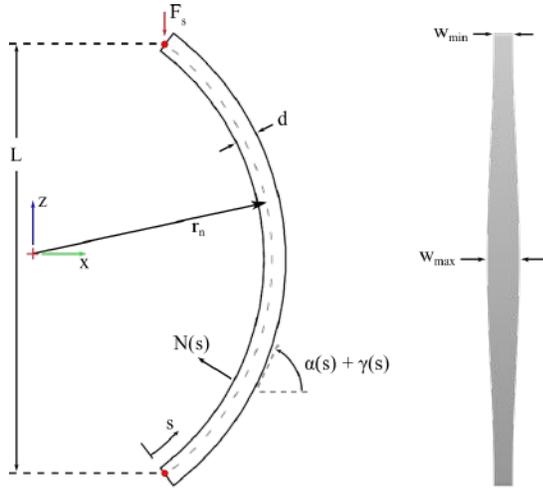


Fig. 4: The geometry properties of the curved spring from a front view (left) and side view (right).

and the corresponding Jacobian matrices can be defined as

$$J_v \triangleq \frac{\partial h_v(q)}{\partial q}.$$

Therefore, we can explicitly solve the constraint force  $\lambda_v$  for domain  $\mathcal{D}_v$ , and (1) simply becomes

$$M(q)\ddot{q} + H_v(q, \dot{q}) + B_{s,v}F_s = B_v u. \quad (3)$$

In  $\mathcal{D}_1$  the mover and foot are each locked to the body due to hard-stops present in the system. In both cases the spring forces will tend to keep the coordinates  $y$  and  $\delta$  fixed to that of the body coordinate,  $z_b$ . When the robot contacts the ground, the system enters  $\mathcal{D}_3$  and the foot becomes fixed to the surface, leading to a holonomic constraint acting on the foot, while the mover continues downward. During this phase the actuator will input force to the system until the mover reestablishes contact with the body, and the system enters  $\mathcal{D}_4$ . This final ground domain is very short and lasts until the curved springs extend back to their equilibrium length and the hardstop engages, at which time the system re-enters the air domain,  $\mathcal{D}_1$ , and the cycle repeats.

In order to distinguish the effects of the actuator input and curved-spring forces from the other dynamics, we write the equations of motion from (3) in an ODE format:

$$\dot{x}_1 = \underbrace{\begin{bmatrix} \dot{q} \\ -M^{-1}H_v \end{bmatrix}}_{f_v(x_1)} + \underbrace{\begin{bmatrix} 0 \\ M^{-1}B_v \end{bmatrix}}_{g_v(x_1)} u + \underbrace{\begin{bmatrix} 0 \\ -M^{-1}B_{s,v} \end{bmatrix}}_{\bar{g}_v(x_1)} F_s. \quad (4)$$

where  $x_1 = (q, \dot{q})^\top \in \mathbb{R}^6$  are the time-domain states.

### B. Spatial-domain dynamics of the curved springs

Normally for linear springs, the spring dynamics are given as a linear map:  $F_s = -kL - cL$  according to Hooke's law, where  $k$  is the stiffness and  $c$  is the coefficient of an additional damping term. However, the curved spring of interest here is governed by spatial-domain dynamics, which will be detailed next. Designing the springs from curved plates serves as a way of controlling their stiffness profile. This will allow for the timing of the ground phase to be synced with a desirable force input from the actuator,

resulting in a more efficient motion than would be possible by having two coils springs in the system as was done in [2].

Fig. 4 details the basic model of the curved, plate spring that will be used in this problem. Based on the work in [13], the internal relations for the moment along the beam and its derivative can be described in terms of the *parameterized arclength*,  $s$ , as

$$M(s) = (A(s)Eer_n)\gamma'(s) \quad (5)$$

$$M'(s) = -N(s) \quad (6)$$

where  $s \in [0, s_0]$  is the position along the arc length of the beam's neutral surface, which is shown as a grey, dashed arc within the beam in Fig. 4.  $M(s)$ ,  $N(s)$ , and  $\gamma(s)$  are the internal moment, force, and resultant deflection at that point due to the loading force,  $F_s$ .  $F_s \in \mathbb{R}$  is the external force acting on the spring along the vertical direction. Note:  $\alpha(s)$  represents the natural angle of the beam at point  $s$  when the spring is unloaded. The deflection angle  $\gamma(s)$  is relative to this  $\alpha(s)$  value. Furthermore,  $E$  represents the Modulus of Elasticity of the spring material,  $r_n$  is the radius of the unloaded spring's neutral surface, and  $e$  is the eccentricity of the neutral surface from spring center due to the curvature of the spring.

Throughout this paper, the radius of the neutral surface and the in-plane thickness of the spring,  $d$ , will be held constant, while the width,  $w(s)$ , is allowed to vary along the length of the spring. From these spatial metrics, the cross sectional area of the spring at the given position,  $A(s) = dw(s)$ , can be found. For this work, we define the variable width to be

$$w(s) = \left( \frac{w_{\min} - w_{\max}}{2} \right) \sin\left(\frac{2\pi}{s_0}s\right) + \left( \frac{w_{\min} + w_{\max}}{2} \right),$$

where  $w_{\min}$  and  $w_{\max}$  are the minimum and maximum width of the spring. This width profile was chosen to make the ends of the spring narrow while the middle is wider. This will allow for a better distribution of stress along the beam [7]. The internal normal force  $N(s)$  is given by

$$N(s) = -F_s \cos(\alpha(s) + \gamma(s)) \triangleq \hat{N}(s)F_s. \quad (7)$$

Rearranging the governing equations (5)-(7) gives the final deformation as:

$$\begin{aligned} \gamma''(s) &\triangleq \frac{\partial^2 \gamma}{\partial s^2} = -\frac{N(s) + Eer_n A'(s)\gamma'(s)}{A(s)Eer_n} \\ &= -\frac{A'(s)}{A(s)}\gamma'(s) - \frac{\hat{N}(s)}{A(s)Eer_n}F_s. \end{aligned} \quad (8)$$

We now define the spring's states as  $x_2 = (\gamma(s), \gamma'(s))^\top \in \mathbb{R}^2$  and  $\rho = (d, w_{\min}, w_{\max}, E) \in \mathbb{R}^4$  as the static parameters, i.e., design variables. We then can write (8) as in the form of an ordinary differential equation (ODE):

$$x_2' = f_\rho(s, x_2) + g_\rho(s, x_2)F_s. \quad (9)$$

For this system, the independent variable is now  $s$  rather than time, and the dependent variable is the deflection angle  $\gamma \triangleq \gamma(s)$ . Hence the solution to (9) can be expressed as

$$x_2(s)^\top = (\gamma(s), \gamma'(s)), \quad \forall s \in [0, s_0]. \quad (10)$$

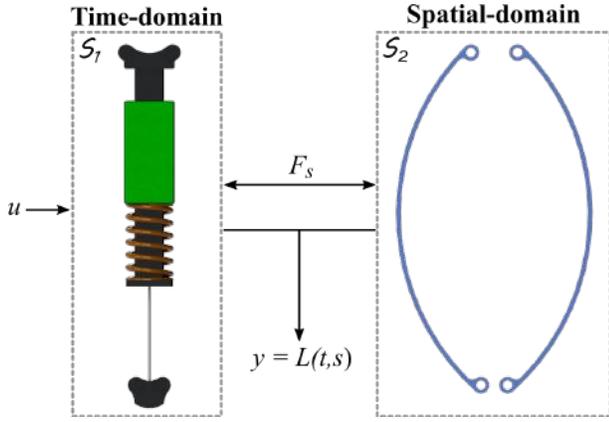


Fig. 5: The signal-flow diagram of the interconnected full-body dynamics.

Since  $s$  is a spatial characterization of the spring dynamics, we call (9) the spatial-domain dynamics of the curved-spring subsystem.

**Boundary condition and numerical solution.** For the subsystem (9), a boundary condition is required to uniquely determine the solution. Since the ends of this spring are pinned, there is no moment transferred at these points [12], and we can set the boundary condition as  $\gamma'(0) = \gamma'(s_0) = 0$ . We then can solve for the numerical solutions  $x_2(s^j)$  for a specified grid  $j \in \{0, 1, 2, \dots, K\}$ , where  $s^j = js_0/K$ . This will be detailed in the next section.

**Output.** The output of the curved-spring subsystem is the vertical length of the spring, i.e.  $L(t, s)$  for at a given time  $t \in [0, T]$ . This can be calculated from the natural shape of the spring in conjunction with the deflection angles as,

$$y(s) \triangleq L(t, s) = \int_0^{s_0} \sin(\alpha(s) + \gamma(s)) ds \quad (11)$$

$$\approx s_{\Delta} \sum_j \sin(\alpha(s^j) + \gamma(s^j)) \quad (12)$$

Note that (11) defines the exact value while (12) is the trapezoidal approximation.

### C. Full-Body Dynamics

Due to the fact that the curved-spring subsystem evolves according to a spatial-domain, independent variable  $s$ , the full-body dynamics of the hopper in design becomes a *interconnected time, spatial-domain system*. This is achieved through an interconnected mechanism, characterized by the shared variables  $L(t, s)$ ,  $F_s$ . Concretely, we summarize the full-body dynamics from (4) and (10) as:

$$S_1: \quad \dot{x}_1 = f_v(x_1) + g_v(x_1)u + \bar{g}_v(x_1)F_s \quad (13)$$

$$S_2: \quad \begin{cases} x'_2 = f_2(s, x_2) + g_2(s, x_2)F_s \\ L \approx \frac{s_0}{K_2} \sum_j^{K_2} \sin(\alpha(s^j) + \gamma(s^j)) \end{cases} \quad (14)$$

where  $x_1(t) \triangleq (q, \dot{q})^\top$ ,  $x_2(t) \triangleq (\gamma, \gamma')^\top$ ,  $j = 0, 1, \dots, K_2$ , and  $s^j = js/K_2$ . As the block diagram shown in Fig. 5, this interconnection between the two subsystems happens at the output level instead of the states level, resulting in a coupled

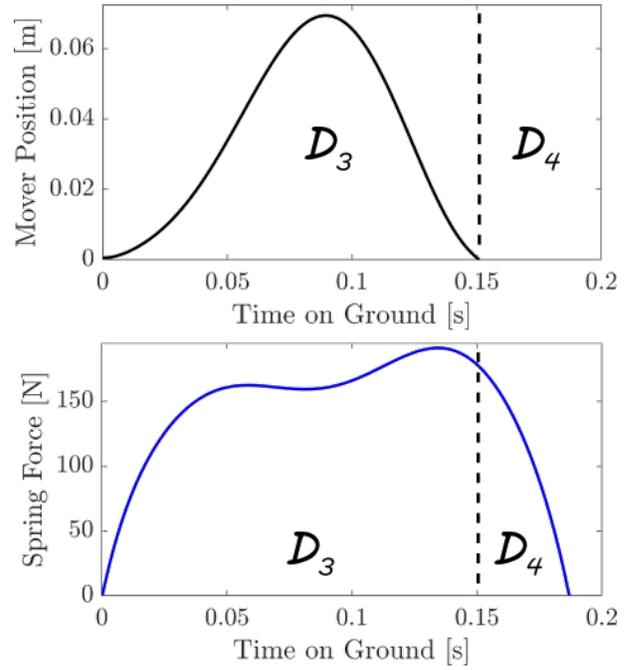


Fig. 6: Output results of the optimization: desired trajectory of the actuated mover coordinate (top), and expected spring force in each curved spring (bottom).

control system [23]. Since the curved springs only deflect while the robot is on the ground, we only need to match the shared variables during phases  $D_3$  and  $D_4$ . Throughout  $D_1$ ,  $L \equiv L_0$  and  $F_s \equiv 0$ .

### III. SYSTEM-LEVEL OPTIMIZATION

The goal of this section is to use optimization [6], [19] to simultaneously design highly-dynamic hopping motions and a curved spring of nonlinear stiffness which allows for such motions with high energy efficiency of the electric motor. Leveraging the previous construction of the multi-phase interconnected dynamics, we introduce a general optimization formulation that solves for a solution to the full-body dynamics, given by (13)-(14), and find the optimal spring parameters  $\rho$ . These parameters will be later used to guide our manufacturing of a hopping robot.

**Decision variables.** We first discretize the time horizon as an evenly distributed grid  $\{t^i\}_{i=0,1,\dots,K_1}$  where  $t^{K_1} = T$  for the time-domain dynamics. Then we similarly define the spatial grid as  $\{s^j\}_{j=0,1,\dots,K_2}$ . We denote the time-domain states at time  $t^i$  as  $x_1^i$  and the spatial-domain states at  $s^j$  as  $x_2^j$ . Correspondingly, the shared spring force at time  $t^i$  is defined as  $F_s^i$ . Define the decision variable as:

$$X \triangleq \{x_1^0, P, (\gamma^0)^i, F_s^i, \rho\} \quad \text{with } i = 0, 1, \dots, K_1$$

where  $x_1^0$  is the initial condition of (13) when  $t = 0$ ,  $P$  is the vector of Bezier polynomial coefficients describing the desired motion of the mover coordinate,  $y$ , and  $\gamma^0$  is a boundary value of (14). Note that  $(\gamma')^0 = 0$  was a given condition, and  $(\gamma^0)^i$  is the boundary value at time  $t^i$ . The mover coordinate is the degree of freedom which is directly

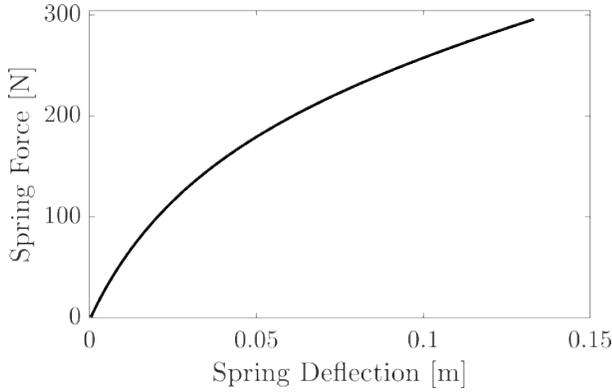


Fig. 7: Final spring force profile as a function of deflection showing the softening nature of the designed springs.

controlled by the actuator, and will be the key trajectory to track on the actual hardware in experiments.

**Optimization problem.** We denote  $p(x_i) \leq 0$  as the path constraints for the time-domain dynamics, then we have the optimization problem using a direct-shooting method as:

**System-Level Design Optimization:**

$$\begin{aligned}
 & \min_X J(X) && \text{(OPT)} \\
 & \text{s.t. } C_1. x_1^{i+1} = x_1^i + \int_{t^i}^{t^{i+1}} (f_v(x_1) + g_v(x_1)u(x_1) + g_v(x_1)F_s^i) dt && i = 0, \dots, K_1-1 \\
 & C_2. \begin{cases} x_2^{j+1} = x_2^j + \int_{s^j}^{s^{j+1}} f_\rho(s, x_2) + g_\rho(s, x_2)F_s^i & j = 0, \dots, K_2-1 \\ \frac{s_0}{K_2} \sum_j \cos(s^j + \gamma^j) - L^i = 0 & i = 0, \dots, K_1 \end{cases} \\
 & C_3. (\gamma')^0 = 0, (\gamma')^{K_2} = 0 \\
 & C_4. p(x_1^i) \leq 0 && i = 0, \dots, K_1 \\
 & C_5. b(x_1^i) = 0 && i = 0, \dots, K_1
 \end{aligned}$$

Essentially, for the initial condition  $x_1^0$  and every spring force  $F_s^i$  picked by the optimization algorithm,  $C_1$  solves for its states at time  $t^i$  with the assumed closed-loop controller  $u_v(x_1)$  (details of the tracking controller can be found in [3]) for all phases  $\mathcal{D}_v$ . We post the boundary condition as an equality condition  $C_3$ , so that for each sampling time  $t^i$  and the shared spring force  $F_s^i$  from  $C_1$ , the dynamics of (14) is then solved. Its output  $L^i$  is also shared by (13) as one of the torso's states. Note that this characterization of  $F_s^i$  and  $L^i$  is the same as the Lagrange multiplier. Further, we posted a range of inequality constraints such as hopping height and max spring deflection,  $\delta_{max}$ , in  $C_4$ . This max deflection was based on the max allowable travel given the size of the robot, as well as the max stress that the plates can handle before failure. A set of equality constraints  $C_5$  is additionally used to enforce the boundary conditions of the torso dynamics to match each other through the ground impact, so that the optimization solution is periodic within the interconnected dynamics. This periodic solution then can be implemented on the hardware as a cyclic hopping motion. Lastly, the cost function was chosen to be  $J(X) = \int_0^T u(x_1)^2 dt$ , to minimize the effort of the actuator over a single hop within the constraints described.

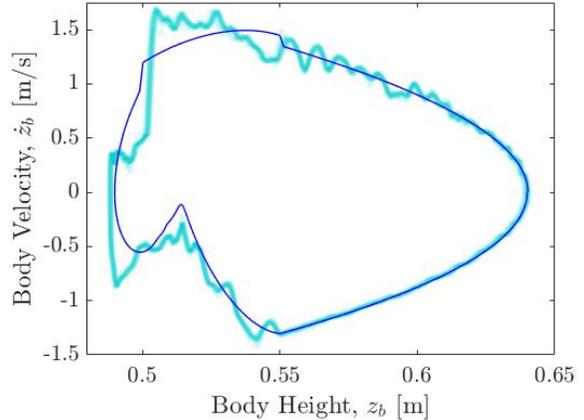


Fig. 8: Phase portrait of the body coordinate with simulated trajectory (dark blue) and experimental results from 20 consecutive hops (light blue).

**Optimal hopping motion and design variables.** This optimization problem was implemented as described in *MATLAB* using *fmincon*. In order to facilitate manufacturing of the springs, the material's elastic modulus of the curved-springs was set for Al-7075 ( $E = 72GPa$ ), while the other spring design parameters were allowed to vary. Other materials such as steel and titanium would be better suited for large strain deformation like this, but were not considered here for the sake of time and limited manufacturing resources at hand. In order to stay within the yield stress limit of Al-7075,  $\sigma_y = 503MPa$ , the feasible height of hopping was limited. The optimization was able to generate results for hop heights up to 17 cm off the ground, before the max stress of the material reached unsafe levels. Fig. 6 shows the mover trajectory and temporal spring force profile for the result of a 9 cm hop height, which requires a max input force of 49 N.

The chosen spring design was based on this same 9 cm hop result which gave the geometric spring properties of  $d = 3.25$  mm,  $w_{min} = 24$  mm, and  $w_{max} = 41$  mm. The resultant spring force profile for this spring design is shown in Fig. 7. The stiffness of these springs vary from 2390 N/m at zero deflection to 1080 N/m at 10 cm of deflection. At this 10 cm mark, the springs are storing 16.45 J and the max stress within the plates is 283 MPa. The duration of the ground phase using this result is 0.18 s, which is almost 3 times longer than what was seen in previous work with a stiff coil spring. The ground phase elongation alone allows for lower peak actuator force and less stressful impacts on the robot. This resultant spring and motion design was taken and implemented as described in the next section.

IV. ROBOT DESIGN AND EXPERIMENTS

The robot used for the experimental validation of this spring design and hopping motion was previously constructed using two linear coil springs [2]. To replace the lower coil spring, a set of curved springs were machined out of Aluminum 7075-T651 and mounted to this robot through slight modifications to the top and bottom of the robot's body (see Fig. 1). A linear guide was also attached in order to constrain the spring deflection to the vertical direction.

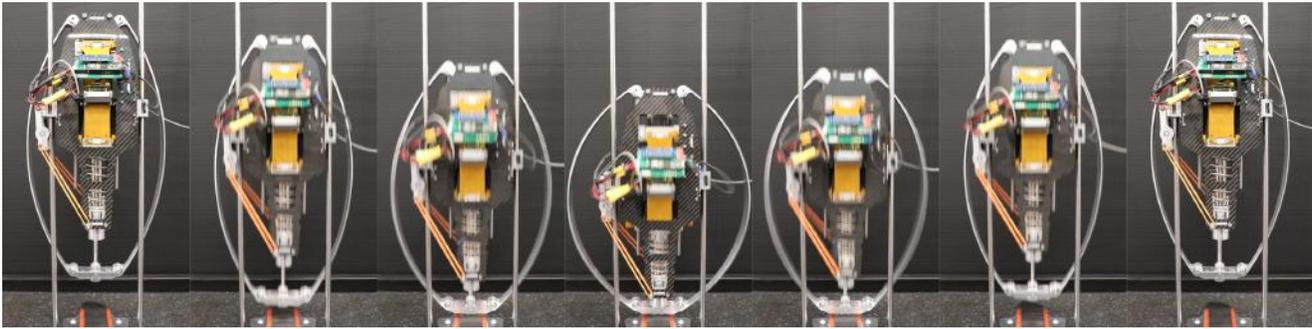


Fig. 9: Tiles from the experiment over a single hop.

The hopping robot was connected to a frame of vertical rails to facilitate its 1-dimensional motion. Experiments were run utilizing an open-loop input of motor force as a time-based trajectory as in [2]. The trajectory used here was generated through optimization for a hop height of 9 cm off the ground. Experiments were performed for trials of 20 consecutive hops. A phase portrait of the body coordinate,  $z_b$ , for one of these trials is shown in Fig. 8. Vibration of the rails and curved springs was visually observed during the trials and was present in the state tracking data as seen in the phase portrait. The plot shows the consistent nature of both the overall hop height and the effects of the ground impact and system vibrations, despite the motion being driven by an open-loop playback of the input trajectory. Motion tiles of a single hop are shown in Fig. 9, which show the full extent of the deflection within the plate springs. The supplemental video shows the experiments in real time and slow-motion.

## V. CONCLUSION

The end result of this work is a method of modeling connected systems of time and spatial domains through their dynamics and shared variables, with the goal of fully capturing their interactions. We were able to formulate an optimization which could generate both optimal motions and physical design parameters for a hopping robot. This method and results were verified on hardware using a novel hopping robot, confirming the efficacy of modeling this as a coupled system. Future work will involve implementing this optimization scheme within FROST [16], as well as using machine learning to characterize the relationship of spring deflection and force of the nonlinear curved springs in an attempt to treat this as a solely time-based system.

## REFERENCES

- [1] Video of experiments for this work. <https://youtu.be/S81eUVXF4d4>.
- [2] E. Ambrose and A. D. Ames. Improved performance on moving-mass hopping robots with parallel elasticity. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2457–2463, 2020.
- [3] E. Ambrose, N. Csomay-Shanklin, Y. Or, and A. Ames. Design and comparative analysis of 1d hopping robots. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5717–5724, 2019.
- [4] G. Antonelli. Interconnected dynamic systems: An overview on distributed control. *IEEE Control Systems Magazine*, 33(1):76–88, 2013.
- [5] J. R. Barber. *Intermediate mechanics of materials*, volume 175. Springer Science & Business Media, 2010.
- [6] J. T. Betts. Survey of numerical methods for trajectory optimization. *Journal of guidance, control, and dynamics*, 21(2):193–207, 1998.
- [7] H. B. Brown and G. Zeglin. The bow leg hopping robot. In *Robotics and Automation (ICRA), IEEE International Conference*, 1998.
- [8] D. Buchfuhrer and C. Umans. The complexity of boolean formula minimization. *Journal of Computer and System Sciences*, 77(1):142 – 153, 2011. Celebrating Karp’s Kyoto Prize.
- [9] M. Chadwick, H. Kolvenbach, F. Dubois, H. Lau, and M. Hutter. Vitruvio: An open-source leg design optimization toolbox for walking robots. *IEEE Robotics and Automation Letters*, PP:1–1, 08 2020.
- [10] A. H. Chang, C. Hubicki, A. Ames, and P. A. Vela. Every hop is an opportunity: Quickly classifying and adapting to terrain during targeted hopping. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3188–3194. IEEE, 2019.
- [11] R. Featherstone. *Rigid Body Dynamics Algorithms*. Kluwer international series in engineering and computer science: Robotics. Springer, 2008.
- [12] N. Georgiev and J. Burdick. Design and analysis of planar rotary springs. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4777–4784. IEEE, 2017.
- [13] N. Georgiev and J. Burdick. Optimization-based design and analysis of planar rotary springs. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 927–934. IEEE, 2018.
- [14] J. W. Grizzle, C. Chevallereau, R. W. Sinnet, and A. D. Ames. Models, feedback control, and open problems of 3D bipedal robotic walking. *Automatica*, 50(8):1955 – 1988, 2014.
- [15] R. T. Haftka and Z. Gürdal. *Elements of Structural Optimization*. Springer Netherlands, 1992.
- [16] A. Hereid and A. D. Ames. Frost: Fast robot optimization and simulation toolkit. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, Canada, Sept. 2017. IEEE/RSJ.
- [17] A. Hereid, C. M. Hubicki, E. A. Cousineau, and A. D. Ames. Dynamic humanoid locomotion: A scalable formulation for HZD gait optimization. *IEEE Transactions on Robotics*, pages 1–18, 2018.
- [18] A. Jameson, L. Martinelli, and N. A. Pierce. Optimum aerodynamic design using the navier–stokes equations. *Theoretical and computational fluid dynamics*, 10(1):213–237, 1998.
- [19] M. P. Kelly. Transcription methods for trajectory optimization: a beginners tutorial. *arXiv preprint arXiv:1707.00284*, 2017.
- [20] B. Kooi. The design of the bow. *Proc. Kon. Ned. Akad. v. Wetensch*, 97(3):1–27, 1994.
- [21] J. D. Lohn, G. S. Hornby, and D. S. Linden. *An Evolved Antenna for Deployment on Nasa’s Space Technology 5 Mission*, pages 301–315. Springer US, Boston, MA, 2005.
- [22] Z. Lyu, G. K. W. Kenway, and J. R. R. A. Martins. Aerodynamic shape optimization investigations of the common research model wing benchmark. *AIAA Journal*, 53(4):968–985, 2015.
- [23] W.-L. Ma, N. Csomay-Shanklin, and A. D. Ames. Coupled control systems: Periodic orbit generation with application to quadrupedal locomotion. *IEEE Control Systems Letters*, 2020.
- [24] J. Reher, W.-L. Ma, and A. D. Ames. Dynamic walking with compliance on a cassie bipedal robot. *2019 18th European Control Conference (ECC)*, pages 2589–2595, 2019.
- [25] K. A. Saar, F. Giardina, and F. Iida. Model-free design optimization of a hopping robot and its comparison with a human designer. *IEEE Robotics and Automation Letters*, 3(2):1245–1251, 2018.
- [26] N. Schmit and M. Okada. Optimal design of nonlinear springs in robot mechanism: simultaneous design of trajectory and spring force profiles. *Advanced Robotics*, 27(1):33–46, 2013.