Designing Dynamic Machines With Large-Scale Root Finding

Mark M. Plecnik, *Member, IEEE*, and Ronald S. Fearing, *Member, IEEE*

***Abstract*—Achieving high-performance dynamic behavior in a robot requires careful design of morphology. However, searching for a global optimum morphology in an intensely nonlinear design space is difficult, especially if stochastic seeding is used. In contrast to optimization, we encode design requirements into a polynomial system with a huge number of isolated roots. Each root describes an alternate robot morphology in the design space. Following this, the computation of nearly all isolated roots constitutes design space exploration. Previously, these systems were intractable, due to the heavy burden of degenerate roots. We relieve this burden by using the finite root generation (FRG) method to enable the discovery of nearly all isolated roots for a certain six-bar design problem for the first time. The FRG synthesis method enables the design of a transmission function from motor dynamics to a loaded end effector to influence the overall dynamic behavior. In an example, we formulate synthesis equations which were previously intractable, obtain 1 528 608 isolated roots (estimated 99.0%), and find 3764 physical designs. Design options are compared according to their sensitivity to joint errors.**

***Index Terms*—Computational geometry, dynamics, kinematics, mechanism design.**

I. Introduction

THERE are no simple tools for designing machines that exhibit some desired dynamic behavior. The leading approaches are based on optimization and intuition. In this article, we exclude the former and lessen dependence on the latter. We propose a scheme for design space exploration powered by the recent finite root generation (FRG) algorithm, a method for obtaining nearly all isolated roots to a system of equations. FRG was first introduced and benchmarked on a system with known roots in [1], but its main objective is to compute systems which were previously intractable. The first realization of this is presented here. In this article, the algorithm is applied beyond a known benchmark to a system that formerly remained unchartered due to its size and complexity. This new system was chosen because it is particularly useful for designing dynamic machines. It describes a planar six-bar linkage with variable transmission properties specified along its end effector path. As a main contribution, the solutions of these equations may be used to influence the dynamic behavior of a machine, demonstrating a new design capability.

This article differs from design methods based on optimization. For given robot dimensions, software tools such as rigid body dynamics solvers already exist to analyze dynamic behavior. Pairing these with general purpose optimizers forms an update loop between design parameters (robot dimensions) and the resulting behavior. Once a meaningful objective function is established, global optimums are sought but are challenged by design space nonlinearities. For example, nonlinearities decrease the likelihood that search strategies based on stochastic seeding will find relevant optimums in the design space.

Here, we take a different approach. Design requirements are translated into a system of equations which exhibit intense nonlinearities. The algorithms of numerical homotopy continuation are well equipped to handle these nonlinearities. By formulating a synthesis system to be square, its roots become isolated. These roots might be visualized as a spread of points across the design space. Each of these points indicates a portion of the design space where requirements are met. Therefore, the action of obtaining nearly all roots constitutes design space exploration. FRG is able to obtain 99.0% of candidate design points. In contrast, locating these points through a high-dimensional grid search in a nonlinear space would be computationally prohibitive.

In practice, synthesis systems are riddled with roots at infinity, a type of degenerate root which is physically irrelevant. This is where current homotopy root-finding algorithms fall short. It is often the case that upwards of 95% of computations are dedicated to discovering useless roots at infinity. Hauenstein *et al.* [2] introduced the regeneration homotopy method to reduce the quantity of divergent paths. Their method was applied to a system of degree 264,241,152 [3]. In contrast, FRG precludes all computations of roots at infinity, allowing the computation of nearly all isolated roots for synthesis systems larger than before. In this article, we find nearly all isolated roots for one such system, and demonstrate new design capabilities that FRG offers. The design work conducted in this article considers a planar six-bar linkage. It may be surprising to some readers that not all isolated roots of all planar linkages were found in the past. Perhaps even more surprising, the frontier of this knowledge begins as soon as one deviates from the simple four-bar linkage.

Manuscript received December 3, 2019; accepted January 13, 2020. Date of publication March 19, 2020; date of current version August 5, 2020. This work was supported by National Science Foundation through Award CMMI-1636302. This paper was recommended for publication by Associate Editor F. Boyer and Editor M. Yim upon evaluation of the reviewers' comments. *(Corresponding author: Mark M. Plecnik.)*

Mark M. Plecnik is with the Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN 46556 USA (e-mail: plecnikmark@nd.edu).

Ronald S. Fearing is with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720 USA (e-mail: ronf@eecs.berkeley.edu).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2020.2975425

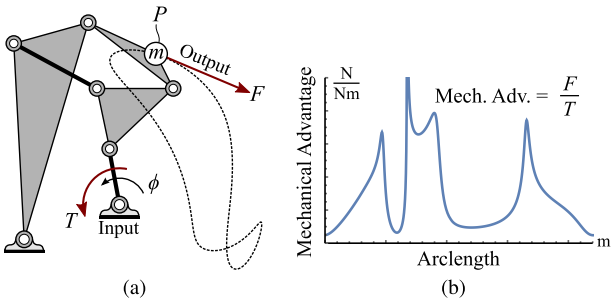


Fig. 1. Six-bar linkage transforms input torque to output force in a highly nonlinear way.

The reason these computations have not taken place in the past is the overwhelming presence of roots at infinity in the synthesis equations. These calculations have remained computationally prohibitive even with tremendous growth in computer speeds and parallelization. FRG approaches this challenge from the algorithm level.

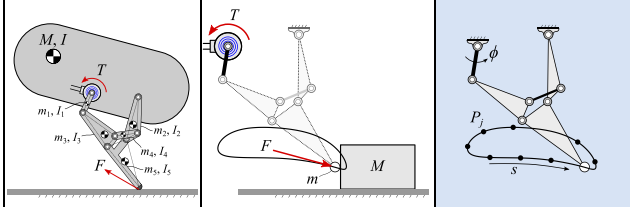
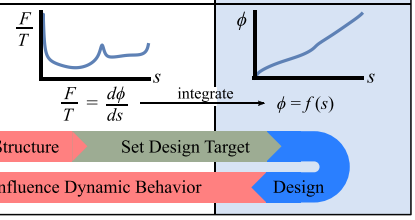
From the standpoint of new design capabilities, the finite, isolated roots that we compute are used to perform design exploration for mechanisms that produce a prescribed transmission function from its motor's dynamics onto the path of its end effector. This allows us to explicitly prescribe torque/force ratios between the motor and load at different points in the configuration space. Control over these force ratios gives us control over dynamic behavior. Performing design exploration over these dynamic machines gives us an unprecedented design tool.

Specifically, we desire to design machines consisting of rigid links and springs that perform some specified motion when subjected to expected loads. We narrow the scope of expected loads by considering only actuator torque (input) and the force acting on an end effector point (output), see Fig. 1. The time-varying magnitudes of this force and torque define inertial motion, but their ratio (called mechanical advantage) is purely geometric,[1] connecting kinematic constraints to dynamic behavior.

To explain our approach, consider the three models illustrated in Table I . The most complete model consists of the inertial parameters of a robot body, the inertial parameters of all leg links, a series-elastic actuator, and intermittent leg contact. This model is governed by hybrid, differential algebraic equations. Its form may be suitable for a general purpose optimizer, but it is not clear how to unearth any prevailing structure that dictates its dynamics. Instead, the dynamics could be distilled to a more simplified model, such as the middle column of Table I. Consider a series-elastic actuator pushing a mass equivalent to the robot's body through a variable transmission ratio (mechanical advantage). This model provides a good approximation of a robot propelling itself off the ground. What is more, it indicates useful forms of variable mechanical advantage, which represents a prevailing structure that dictates dynamic behavior. To instantiate one of these forms, the model of the right column of Table I is most useful. Mechanical advantage can be posed as geometric requirements between input–output parameters using

[1]Neglecting losses.

TABLE I
DETAILED DYNAMIC MODEL IS DIFFICULT TO WORK WITHIN A DESIGN CONTEXT.

	Detailed Dynamic Model	Simplified Dynamic Model	Synthesis Model
Equations	• Differential • Algebraic • Hybrid	• Differential • Algebraic	• Algebraic
Variables	• All configuration variables • All 1st derivatives • All 2nd derivatives	• Input-ouput config. variables • 1st derivative of input variable • 2nd derivative of output variable	• All configuration variables • All dimensions
Sketch			
Prevailing Structure	n/a		

A simplified dynamic model reveals underlying structure which influences the dynamics. Customization of this structure sets algebraic constraints. When these constraints are satisfied, the dynamics of the simplified and detailed models are influenced in an intended manner.

purely algebraic equations. If the design task of instantiating these requirements is successful, then the dynamic behavior of the more complete models up to the chain can be influenced.

Finding limb geometries that produce forms of mechanical advantage that suit our customization needs requires investigation of a sufficiently complex design space. We consider a planar six-bar linkage, shown in Fig. 1. Despite its simple structure, it is capable of producing complex, constrained motions, e.g., it may draw a degree 16 plane curve [4]. On the other hand, this same simple structure begets challenging nonlinearities within its synthesis equations. To corral these nonlinearities, FRG was implemented and found approximately 1.5×10^6 finite, isolated roots for a numerically general version of the synthesis system. This set of roots serves as a new design instrument, which can be used repeatedly in conjunction with the well-established parameter continuation technique [5] to efficiently compute nearly all isolated roots of any system of this structure in the future.

Our treatment of design involves dealing with algebraic equations rather than geometric constructions. This makes our method technically simple for designers without prior knowledge of mechanism parameters, reducing the need for geometric creativity. Furthermore, by generating multiple design options, a designer is better equipped to account for secondary requirements, such as sensitivity to fabrication errors.

In this article, we perform design space exploration by finding nearly all isolated roots of the synthesis system, something that was not possible on this scale in the past. In particular, we are able to perform design exploration for machines with control over the ratios between input torque and load. This gives us

control over dynamic behavior. Since these sets of isolated roots were only recently computable by FRG, we present a design exploration tool for dynamic machines that was not possible in the past.

To illustrate our approach and demonstrate its utility for designing a machine with some desired dynamic behavior, we design a leg mechanism suitable for a small running robot. The mechanism is powered by a series-elastic actuator, with mechanical advantage specified in order to deliver kinetic energy during push-off at powers greater than its motor's output. This behavior is termed power modulation [6]. In addition to achieving the required mechanical advantage, we design a foot path that recycles the leg each step with the motor running only in the forward direction. In a related work [7], this mechanism was prototyped and tested.

II. Literature Review

A. Mechanism Synthesis

Previous design methods that produce dynamic characteristics in mechanisms focus on reducing the shaking forces and moments that act on the fixed frame or reducing fluctuations in actuator torque. Berkof, Tepper, and Lowen [8]–[10] formulated conditions for fixing the center of mass of a moving linkage, which removes the shaking force during dynamic operation at any speed. Skreiner [11] demonstrated how to reduce speed fluctuations and pin forces by adding a spring to a four-bar. Conte *et al.* [12] included kinetostatic force equations at a sampling of configurations for a four-bar linkage with prescribed motion in order to compute optimal dimensions that minimize shaking forces/moments, bearing forces, or input torques. Yao and Yan [13] use noncircular gears to reduce fluctuations in driving torque for planar linkages. Yan and Yan [14] present a comprehensive optimization procedure for the design of four-bar linkages that satisfies kinematic requirements while reducing shaking force/moment and motor power dissipation. They consider link dimensions, mass, input speed trajectory, and servomotor control parameters as design variables. Moore *et al.* [15] computed the complete set of force and moment balanced four-bars.

A few approaches more explicitly deal with time and motion specifications. Sherwood [16] and Liniecki [17] designed slider-crank mechanisms that achieve desired positions and velocities coordinated in time by modifying dimensions between numerical dynamic computations. Halter and Carson [18] developed a design procedure to add mechanical elements to an existing mechanism in order to obtain a desired motion-time response. Desired generalized forces of to-be-synthesized mechanical elements were computed from inverse dynamics calculations. Design parameters of these mechanical elements were then computed through optimization as a curve fitting problem to the desired generalized forces. Zhen [19] applied dynamic spring synthesis to spatial mechanisms.

Starr [20] considered not just how to implement desired dynamic motions, but how to specify those motions using optimal control theory. Later, Manoochehri and Seireg [21] also employed these optimal control techniques. Chen and Tsai [22] discussed how to design gearing for a manipulator to possess both kinematic isotropy and good acceleration capacity.

Another dynamic consideration is the presence of vibrations in high-speed machinery. Matthew and Tesar [23] considered the effects of vibrations and inertial forces on cam design. Li and Kota [24] analyzed the frequency response of a micromachined stroke amplifying mechanism for an electrostatic actuator and conducted sensitivity analyses useful for synthesis. Motivated by their investigation of parallel manipulators, Martini *et al.* [25] studied mass balanced and elastically compensated closed chains through the addition of weights and springs. Design methods that consider the kineto-elastodynamic effects in mechanisms were presented by Imam and Sandor [26]. Several authors have investigated improving compliant wing flapping mechanisms through careful design. Baek *et al.* [27] consider the use of resonance to reduce the battery consumption, Khatait *et al.* [28] minimized peak driving torque, and Tantanawat and Kota [29] investigated the benefits of dynamic strain energy storage.

B. Design Principles for Legged Robots

One motivation for this article is to design dynamic machines suitable for legged robots. The literature indicates various design principles and outstanding challenges to direct the formulation of design requirements. As opposed to the previous section which focused on design methods to achieve a dynamic response, this section exposes design principles that have been indicated by the literature. The latter is more focused on requirements specification while the former is based on the techniques to achieve those requirements.

With respect to small hexapedal running robots, Clark *et al.* [30] and Hoover *et al.* [31] indicate the importance of passive mechanics, self-stabilization, open-loop/feedforward control, minimal actuation, and integrated manufacturing techniques in order to negotiate the strict limits on power, computation, and weight associated with small robots. Zarrouk *et al.* [32] addressed the need of high speed turning in a hexapedal runner by taking advantage of the dynamic roll and pitch instabilities. The challenges of legged robots listed by Buehler [33] include underactuated dynamics, friction limited horizontal ground forces, short stance times to apply control, bandwidth limitations, and limited power/energy densities of commercial actuators. Collins *et al.* [34] leveraged passive-dynamics to create robots that mimic human walking with simple actuators that do not require precise joint-angle control. Curran *et al.* [35] indicated the role of series-elastic actuation in decoupling the limits of a dc motor from a dynamic task. Semini *et al.* [36] stressed the importance of high powered actuators. In the analysis of a symmetric five-bar linkage, Kenneally and Koditschek [37] measured performance according to the conversion of battery energy into body energy, the minimization of touchdown losses, and the storing/return of energy during stance. Blackman *et al.* [38] extended their analysis, highlighting the ability of this symmetric five-bar to achieve greater speeds and stability in one configuration, but greater jump height in another. Brown *et al.* [39] analyzed the ability of this five-bar

and other linkage designs to balance workloads between motors. Kalouche [40] designed a spatial linkage based on this five-bar. Wensing *et al.* [41] demonstrated the importance of torque density, high-bandwidth force control, and impact mitigation in a quadrupedal running robot.

C. Homotopy Continuation

The role of homotopy continuation methods in mechanism design traces back to Roth and Freudenstein's bootstrap method [42]. Many important advancements have been made since then. A full recount is inappropriate here, so we refer readers to general references [43]–[45]. Here, we note a few specific contributions most relevant to the current work. Multi-homogeneous homotopies take advantage of sparse monomial structures in order to reduce the number of homotopy paths to track [46]. Polyhedral homotopies are capable of obtaining the optimum number of paths [47]. To find approximately complete sets of roots more efficiently, Hauenstein *et al.* [2] introduced regeneration homotopy, which exploits the sparse monomial structure of a system one equation at a time to find nearly all of its isolated roots. Plecnik and Fearing [1] introduced the FRG method for efficiently collecting the roots of mechanism synthesis systems. This article compared FRG to other homotopy methods, such as those available through the software Bertini [48], on a well-known benchmark problem. The problem considered in this article is not a well-known benchmark. FRG quickly generates startpoints and start systems that track only to finite endpoints, eliminating expensive homotopy paths that diverge to infinity. More recently, methods based on monodromy loops have led to efficient root finding techniques [49], [50].

In this article, we create a design procedure for instantiating dynamic behavior into a machine by applying FRG to obtain nearly all isolated roots of a system that was previously computationally prohibitive. The process begins by posing a simplified system, referred to as a dynamic mock-up, which captures the desired dynamics, presented in Section III. The mock-up indicates geometric constraints which are encoded into relevant synthesis equations, presented in Section IV. The FRG method was applied to obtain nearly all isolated roots of these equations, presented in Section V. These roots serve as a design instrument to efficiently obtain the roots of the synthesis equations for different geometric requirements. We demonstrate this in Section VI by applying the results of FRG to the design of a leg mechanism that modulates power beyond its motor's output during a fully rotatable running motion. The sensitivity of design candidates to dimensional errors is analyzed in Section VII and Section VIII concludes this article.

III. Dynamic Mock-Up

To begin, we consider the dynamics of the mock-up system illustrated in Fig. 2. A motor twists a series spring, transmitting torque T to an input crank. This torque is transformed by the mechanical advantage of a to-be-designed linkage into a force F_t at an end effector point that is resisted by an attached mass m and in contact with a sliding mass M. The direction of F_t lies tangent to a point path $\mathcal{C}$ constrained by the to-be-designed linkage.

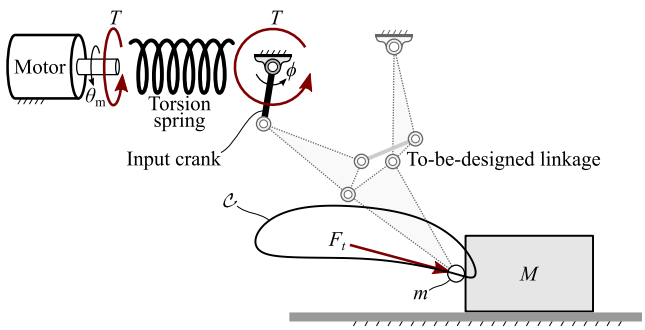


Fig. 2. Mock-up dynamic system used to inform desirable kinematic properties of a to-be-designed linkage. The fixed motor is connected in series with a spring to an input crank. Without any linkage existing yet, a user-defined angle-arc length relationship describes how torques at the crank transform to forces at the masses.

A. Specifying Mechanism Characteristics

Although this to-be-designed linkage does not yet exist, we may freely specify its would-be characteristics. In particular, we specify its constrained point path and variable mechanical advantage as a function of the point's position along this path. We accomplish this by clicking points on a computer screen, and interpolating curves through these points. We choose the Fourier-based interpolation method used in [51] to create periodic curves that are smooth everywhere.

For example, the red points shown in Fig. 3(a) were entered manually with a computer mouse. The interpolated curve is parameterized by the Fourier series

$$\tilde{x}(\tau) = \tfrac{1}{2}a_0 + \sum_{k=1}^{o} \left(a_k \cos(2\pi k\tau) + b_k \sin(2\pi k\tau)\right)$$
$$\tilde{y}(\tau) = \tfrac{1}{2}c_0 + \sum_{k=1}^{o} \left(c_k \cos(2\pi k\tau) + d_k \sin(2\pi k\tau)\right) \quad (1)$$

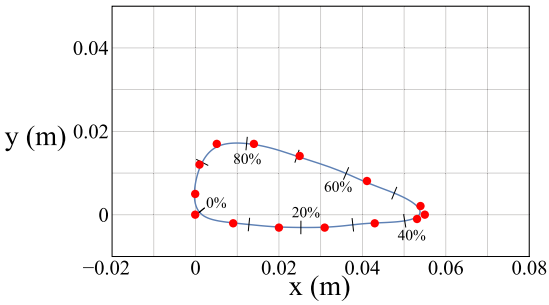
such that $\mathcal{C} = \{(\tilde{x}(\tau), \tilde{y}(\tau)) |\ \tau \in [0, 1]\}$. The values of Fourier coefficients for the example problem are displayed in Appendix A. Order o may be freely specified.

Next, we replace parameter τ with arc length s. To do this, we integrate along $\mathcal{C}$ to densely sample points (s, τ), then interpolate through them with a cubic Hermite spline creating a monotonic function for τ of s. Substituting this function into (1) yields a numerical arc length parameterization of $\mathcal{C}$

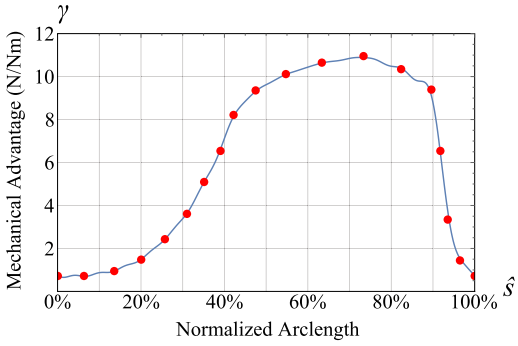
$$x(s) = \tilde{x}(\tau(s))$$
$$y(s) = \tilde{y}(\tau(s)). \quad (2)$$

Equation (2) is valid for $s = [0, s_{\text{total}}]$, the arc length of the entire closed curve.

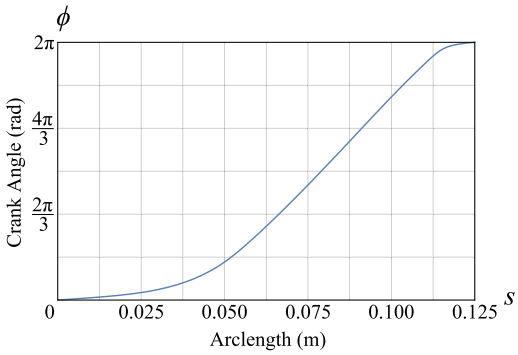
We follow a similar process for specifying mechanical advantage with a few modifications. First, mechanical advantage γ is defined as a function of normalized arc length $\hat{s}$ so that points are entered manually with a mouse in the $\hat{s}$-γ plane shown in Fig. 3(b). Second, the endpoints at $\hat{s} = 0$ and $\hat{s} = 1$ are always



(a)



(b)



(c)

Fig. 3. Mechanism properties were specified by clicking the red points on a computer screen and interpolating. Specified properties are (a) the point trace path and (b) mechanical advantage coordinated with path arc length. In (c), the angle-arc length relationship is computed by integrating (b).

specified and share the same γ value. Third, after interpolation, the resulting Fourier series $\gamma(\hat{s})$ must be vertically shifted. For a fully rotatable input crank, the area under $\gamma(\hat{s})$ must be 2π. To see this, note that mechanical advantage is equal to the ratio of input and output velocities

$$\frac{d\phi}{d\hat{s}} = \gamma(\hat{s}). \tag{3}$$

For a full traversal of $\mathcal{C}$, the crank must rotate 2π or rather

$$2\pi = \int_0^1 \gamma(\hat{s})d\hat{s}. \tag{4}$$

The Fourier coefficients of $\gamma(\hat{s})$ are given in Appendix A as values (α_k, β_k).

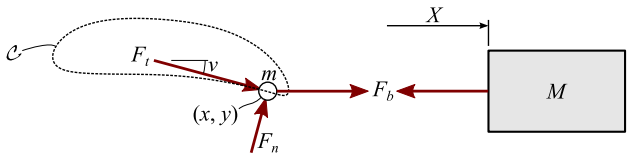


Fig. 4. Detail of force notation at the interaction of m and M. Corresponds to Fig. 2.

Next, function $\gamma(\hat{s})$ is rewritten to accept a nonnormalized arc length s argument

$$\gamma(s) := \frac{\gamma(s/s_{\text{total}})}{s_{\text{total}}}. \tag{5}$$

Note we take a notational liberty in (5) by redefining function γ in terms of itself instead of making up new notation. Integration of $\gamma(s)$ yields an angle-arc length relationship, which creates the specified mechanical advantage

$$\phi(s) = \int_0^{s_{\text{total}}} \gamma(s)ds + \text{constant}. \tag{6}$$

For our example, this curve is plotted in Fig. 3(c). For convenience, the constant of integration is chosen so that $\phi(0) = 0$.

B. Equations of Motion

After specifying the characteristics of a to-be-designed linkage, Fig. 3, the mock-up system illustrated in Fig. 2 may be simulated.

The motor dynamics obeys a linear torque-speed law

$$T = -\frac{T_{\text{stall}}}{\omega_{\text{free}}}\dot{\theta}_{\text{m}} + T_{\text{stall}} \tag{7}$$

where T_{stall} is stall torque and ω_{free} is free-running speed.

The torque transmitted through the spring follows:

$$T = -k_{\text{spring}}(\phi - \theta_{\text{m}}) - c_{\text{damp}}(\dot{\phi} - \dot{\theta}_{\text{m}}) \tag{8}$$

where k_{spring} is stiffness and c_{damp} is damping.

The mechanical advantage transformation from torque to output force is

$$F_t = T\gamma(s) \tag{9}$$

where F_t is the force acting on mass m tangent to curve $\mathcal{C}$, and $\gamma(s)$ follows (5).

Finally, the forces acting on masses m and M follow the diagrams drawn in Fig. 4. That is

$$\begin{aligned} F_t\cos\nu - F_n\sin\nu - F_b &= m\ddot{x} \\ F_t\sin\nu + F_n\cos\nu &= m\ddot{y} \\ F_b &= M\ddot{X} \end{aligned} \tag{10}$$

and if m and M are in contact

$$\ddot{x} = \ddot{X} \tag{11}$$

else

$$F_b = 0 \tag{12}$$

TABLE II
VALUES OF SIMULATION PARAMETERS

Parameter	Value	Comment
T_{stall}	0.33 Nm	Pololu 298:1 Micro Metal Gearmotor
ω_{free}	11.5 rad/s	
k_{spring}	0.14 Nm/rad	14 mm cylindrical cut of latex
c_{damp}	0.0005 Nm/(rad/s)	Estimated
m	0.02 kg	Stand-in for to-be-designed linkage inertia
M	0.23 kg	Typical mass of small robot

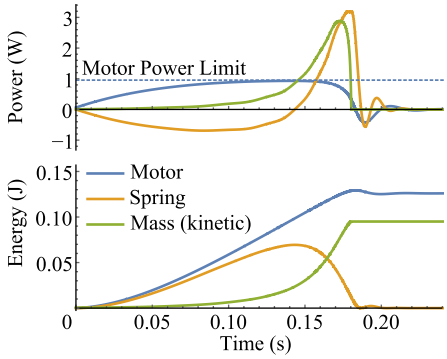


Fig. 5. Simulation results of the model shown in Fig. 2 for the mechanism properties shown in Fig. 3. The mechanical power output associated with the mass surpasses the motor's limit. Note that here we plot spring power as the negative of the time derivative of spring energy in order to illustrate its peak compared to the motor's limit.

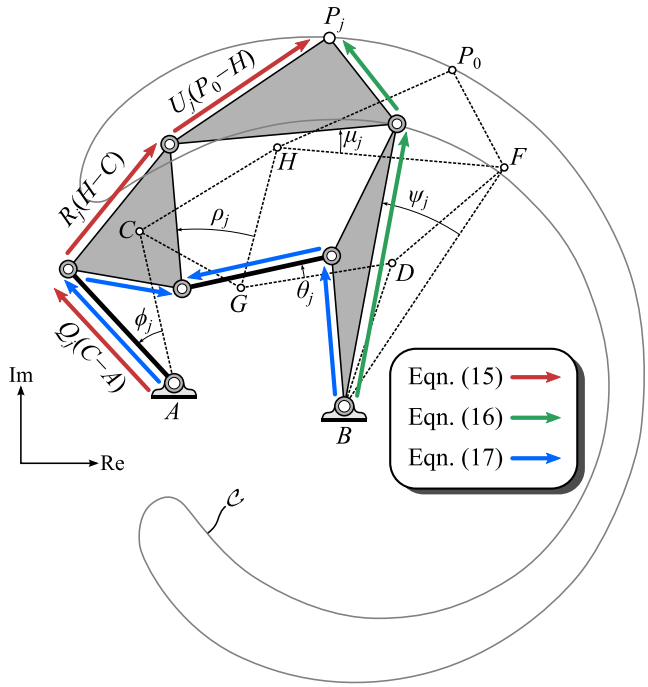


Fig. 6. Diagram describing the notation used in the synthesis formulation. Linkage pivots in a reference configuration are marked with complex numbers A, B, C, D, F, G, and H. The jth displaced configuration is shown here.

where F_n is the constraint force normal to $\mathcal{C}$, F_b is the reaction force between m and M, (x, y) are the coordinates of m constrained to $\mathcal{C}$ according to (2), X is the coordinate of M, and ν is the tangency angle of $\mathcal{C}$ at (x, y), that is

$$\nu = \arctan \frac{\dot{y}}{\dot{x}}. \tag{13}$$

Equations (7)–(13) in conjunction with $x(s)$, $y(s)$, $\gamma(s)$, and $\phi(s)$ specified in Section III-A form the equations of motion. In particular, we point out the appearance of mechanical advantage γ in (9). It is here that mechanical advantage influences dynamic behavior. To solve these equations, they were rewritten into the form of an initial value problem for use with standard ODE solvers.

C. Mock-Up Example

The mechanism characteristics shown in Fig. 3 were specified to create a recycling foot path with a fully rotatable crank that modulates the kinetic power of M beyond the motor's limit along the bottom portion of $\mathcal{C}$. These specifications were informed by intuition, past experience, and bioinspiration [6].

The values of physical parameters are displayed in Table II. The results of numerical simulation are shown in Fig. 5. Simulation shows that modulation of motor power by a factor of 3.0 is possible. A leg designed with these mechanical characteristics might deliver 95 mJ of kinetic energy in one stance phase.

IV. SYNTHESIS EQUATIONS

After determining desired mechanism characteristics in Section III, our objective is to synthesize a linkage that exhibits those characteristics. In particular, we desire a physical instantiation of $x(s)$, $y(s)$, and $\phi(s)$. The latter of the three is desired more so for its derivative $\gamma(s)$ rather than its direct curve, but we employ it directly as a design goal nonetheless. The design problem of finding a linkage which creates a desired endpoint path is called the *path generation*. When this path is additionally coordinated with the angle of an input crank, this problem is referred to as *timed curve generation* [52].

In this section, we formulate synthesis equations for a Stephenson II six-bar linkage. The choice to investigate six-bars is not arbitrary. These linkages are capable of single degree-of-freedom motions, which are much more complex than four-bars, the next simplest single degree-of-freedom linkage. However, the choice to particularly investigate the Stephenson II type of six-bar is partially arbitrary. There are other types worth investigation too, which are not considered in this article.

Specifically, we formulate the synthesis equations for a Stephenson II six-bar linkage to track an endpoint attached to one of its floating binary links through eight positions that are coordinated with its binary input crank angle. Later in Section V, an approximately complete set of isolated roots of these equations are obtained for the first time.

A. Formulation

A Stephenson II six-bar linkage is drawn in Fig. 6. It is composed of seven joints. Fig. 6 displays a reference configuration and the jth displaced configuration. There are $N-1$ displaced configurations. Joint coordinates in the reference configuration are stored in the real and imaginary components of the complex numbers A, B, C, D, F, G, and H. The coordinates of the end

effector point in the reference configuration is P_0 and in the jth displaced configuration is P_j. The angular displacement of each of the five moving links is measured by ϕ_j, ρ_j, ψ_j, θ_j, and μ_j as marked in Fig. 6.

Planar vectors written in complex form may be rotated by multiplication with rotation operators

$$Q_j = e^{i\phi_j}, \quad R_j = e^{i\rho_j}, \quad S_j = e^{i\psi_j},$$
$$T_j = e^{i\theta_j}, \quad U_j = e^{i\mu_j}, \qquad j = 1, \ldots, N-1. \tag{14}$$

For the synthesis problem of interest, we take it that all values of crank rotation Q_j, $j = 1, \ldots, N-1$ and end point position $P_j, j = 0, \ldots, N-1$ are known. All other symbols are variables to be solved for or eliminated. Note the index $j = 0$ does not exist for Q_j because this simply refers to the reference configuration.

To begin deriving the synthesis equations, first form three independent loop equations from Fig. 6

$$R_j(H - C) = P_j - A - Q_j(C - A) - U_j(P_0 - H) \tag{15}$$
$$S_j(F - B) = P_j - B - U_j(P_0 - F) \tag{16}$$
$$T_j(G - D) = A - B + Q_j(C - A) + R_j(G - C) - S_j(D - B)$$
$$j = 1, \ldots, N-1. \tag{17}$$

The vectors associated with (15)–(17) are labeled in Fig. 6. To eliminate R_j, S_j, and T_j from these equations, (15) and (16) are solved for R_j and S_j, respectively, then substituted into (17). The result is then multiplied by its own conjugate. Equations (15) and (16) are also multiplied by their own conjugates. After partial expansion, this yields the equations below. The overbar denotes conjugation

$$\beta_j + \bar{\beta}_j - P_j\bar{P}_j - P_0\bar{P}_0 = 0 \tag{18}$$
$$\xi_j + \bar{\xi}_j - P_j\bar{P}_j - P_0\bar{P}_0 = 0 \tag{19}$$
$$U_j\lambda\bar{\zeta}_j + \bar{U}_j\bar{\lambda}\zeta_j - \zeta_j\bar{\zeta}_j - \lambda\bar{\lambda} + (G - D)(\bar{G} - \bar{D}) = 0 \tag{20}$$
$$j = 1, \ldots, N-1.$$

The new intermediate variables β_j, ξ_j, ζ_j, λ are introduced only for compact presentation and illustrating equation structure. They are defined

$$\beta_j = U_j(P_0 - H)\left(\bar{P}_j - \bar{A} - \bar{Q}_j(\bar{C} - \bar{A})\right)$$
$$+ Q_j(C - A)(\bar{P}_j - \bar{A}) + A(\bar{P}_j + \bar{C} - \bar{A}) + H(\bar{P}_0 - \bar{C})$$
$$\xi_j = U_j(P_0 - F)(\bar{P}_j - \bar{B}) + P_j\bar{B} + F(\bar{P}_0 - \bar{B})$$
$$\zeta_j = A - B + Q_j(C - A) + \frac{G - C}{H - C}$$
$$\times (P_j - A - Q_j(C - A)) - \frac{D - B}{F - B}(P_j - B)$$
$$\lambda = \frac{G - C}{H - C}(P_0 - H) - \frac{D - B}{F - B}(P_0 - F). \tag{21}$$

Conjugates $\bar{\beta}_j$, $\bar{\xi}_j$, $\bar{\zeta}_j$, $\bar{\lambda}$ are defined as expected. Equations (18)–(20) were simplified by the fact that a rotation operator multiplied by its conjugate equals one, eliminating R_j, S_j, and T_j. The remaining unknown rotation operator is U_j. Its values must obey

$$U_j\bar{U}_j = 1. \tag{22}$$

In order to estimate the number of roots of our synthesis system, we convert (18)–(20) to polynomials. To do this, we first change our interpretation of the overbar. Rather than a conjugate operation, variables with an overbar are considered separate unknowns. That is, instead of considering the coordinated pair $(\mathrm{Re}[A], \mathrm{Im}[A])$, we consider isotropic coordinates $(A, \bar{A})$ [53]. The two are related by an invertible linear transformation

$$\begin{Bmatrix} A \\ \bar{A} \end{Bmatrix} = \begin{bmatrix} 1 & i \\ 1 & -i \end{bmatrix} \begin{Bmatrix} \mathrm{Re}[A] \\ \mathrm{Im}[A] \end{Bmatrix}. \tag{23}$$

Next, we introduce the following substitutions to make (20) polynomial and help reduce the total degree of our synthesis system:

$$a = A\bar{H}, \qquad \bar{a} = \bar{A}H \tag{24}$$
$$b = B\bar{F}, \qquad \bar{b} = \bar{B}F \tag{25}$$
$$c = (C - A)\bar{H}, \qquad \bar{c} = (\bar{C} - \bar{A})H \tag{26}$$
$$d = \frac{D - B}{F - B}, \qquad \bar{d} = \frac{\bar{D} - \bar{B}}{\bar{F} - \bar{B}} \tag{27}$$
$$g = \frac{G - C}{H - C}, \qquad \bar{g} = \frac{\bar{G} - \bar{C}}{\bar{H} - \bar{C}} \tag{28}$$
$$k = g(P_0 - H) - d(P_0 - F), \quad \bar{k} = \bar{g}(\bar{P}_0 - \bar{H}) - \bar{d}(\bar{P}_0 - \bar{F}). \tag{29}$$

Substituting (24)–(29) into (18)–(20) obtains

$$\beta_j + \bar{\beta}_j - P_j\bar{P}_j - P_0\bar{P}_0 = 0 \tag{30}$$
$$\xi_j + \bar{\xi}_j - P_j\bar{P}_j - P_0\bar{P}_0 = 0$$
$$U_jk\bar{\zeta}_j + \bar{U}_j\bar{k}\zeta_j - \zeta_j\bar{\zeta}_j - k\bar{k} \tag{31}$$
$$+ (g(H - C) + C - d(F - B) - B)$$
$$\times \left(\bar{g}(\bar{H} - \bar{C}) + \bar{C} - \bar{d}(\bar{F} - \bar{B}) - \bar{B}\right) = 0$$
$$j = 1, \ldots, N-1 \tag{32}$$

where

$$\beta_j = U_j\left(P_0\left(\bar{P}_j - \bar{A} - \bar{Q}_j(\bar{C} - \bar{A})\right) - \bar{P}_jH + \bar{a} + \bar{Q}_j\bar{c}\right)$$
$$+ Q_j(C - A)(\bar{P}_j - \bar{A}) + A(\bar{P}_j + \bar{C} - \bar{A}) + !H(\bar{P}_0 - \bar{C})$$
$$\xi_j = U_j\left(P_0(\bar{P}_j - \bar{B}) - \bar{P}_jF + \bar{b}\right) + P_j\bar{B} + P_0\bar{F} - b$$
$$\zeta_j = A - B + Q_j(C - A)$$
$$+ g\left(P_j - A - Q_j(C - A)\right) - d(P_j - B). \tag{33}$$

When $N = 8$, (22), (24)–(26), (29)–(33) are 36 equations in 36 unknowns

$$\left\{A, \bar{A}, B, \bar{B}, C, \bar{C}, F, \bar{F}, H, \bar{H}, a, \bar{a}, b, \bar{b}, c, \bar{c}, d, \bar{d}, g, \bar{g}, k, \bar{k}\right\},$$
$$\left\{U_j, \bar{U}_j\right\}, \qquad j = 1, \ldots, 7. \tag{34}$$

The total degree of a system of polynomials is computed as the product of the degree of each individual equation, in this case $2^7 2^8 2^7 2^7 4^7 \approx 8.8 \times 10^{12}$. The total degree represents an

upper bound on the number of isolated finite roots that exist. For sparse systems, such as the one at hand, the total degree is a huge overestimation. More accurate root counting techniques include the multihomogeneous number, product decomposition, and the BKK bound [53], but applying these techniques requires a more thorough investigation of the polynomials' structure than we offer in this article. A quick application of multihomogeneous root counting [46] to the above equations shows the degree to decrease to 3.5×10^{12} when variables are partitioned according to the eight groups displayed in (34). An advantage of FRG is that no special knowledge of the equations' structure is needed beforehand to achieve computational efficiency. After applying FRG in the next section, an estimate of 1.5×10^6 finite roots is computed.

V. Finite Root Generation

FRG was first proposed in [1]. The method is based in homotopy continuation and is used for obtaining nearly all isolated roots, both real and complex, to large kinematic synthesis systems.

Generally, homotopy solvers work to find roots of a polynomial *target system* by first constructing a polynomial *start system* with easily obtainable roots and a monomial structure at least more general than that of the target system. Then, a homotopy between the systems is constructed and start system roots, *startpoints*, are tracked continuously to target system roots, *endpoints*. Through this method, nearly all finite roots of the target system can be found.

The problem that arises is that start systems often possess far more startpoints than target systems possess finite endpoints. The residual of this mismatch, which tends to be huge, manifests itself as startpoints that track to roots at infinity. FRG circumvents these useless calculations by constructing start systems with the exact monomial structure as target systems. This is accomplished by generating random complex linkage parameters and extracting a start system from its motion and a startpoint from its dimensions. Note that a linkage with random complex dimensions may not have a physical manifestation, but its kinematics equations are still valid in complex space. This single startpoint will indeed track to a finite root of the target system. This aspect is the key to the method and occurs robustly in practice. To get a second root, this process is repeated. As the set of finite target roots we aim to obtain is very large, chances are that this second startpoint will track to a different endpoint, with no guarantee. Repeating this process, roots are accumulated. As FRG trials progress, eventually the odds of acquiring a new root become unfavorable. FRG trades away computations of roots at infinity for computations of duplicate roots. For sparse systems, the latter tends to be far fewer than the former, making it a good tradeoff. The frequency of duplicate roots was characterized in [54]. Tracking their occurrence allows an in-process estimate of the percentage of the total root set that has been obtained.

FRG provides additional computation savings by exploiting known solution structures in a straightforward way. For linkage synthesis equations, this structure is primarily provided by the presence of linkage cognates. The cognate of a single degree-of-freedom linkage is another linkage with different dimensions that exactly reproduces some aspect of the original linkage's continuous motion. For a given linkage type (e.g., Stephenson II) and motion of interest (e.g., timed curve generation), the presence and number of cognates is often known. As well, geometric transformations between cognates are known too. Using these transformations, an entire set of cognate roots can be quickly computed from one endpoint. Each root in a cognate set is a separate isolated solution, and cognate sets have no intersection with each other. Combining cognate structures with other existing symmetries (if present) partitions target roots into a smaller number of sets, thereby reducing the expected computational load by the multiple size of a single cognate root set.

Finally, any time one intends to obtain isolated roots for several systems within a specific family, this is accomplished efficiently using the parameter homotopy method [5]. That is, in order to obtain a complete set of isolated roots for a new instance of our synthesis system, say for a different set of task parameters, we do not need to start from scratch. First, nearly all roots are found for a numerically general member of the family of systems of interest. This is called an *ab initio* solution and may be found using any method, in our case FRG. The *ab initio* roots may then serve as startpoints for a homotopy that transforms parameters of the *ab initio* system into any other member of the family of interest. These subsequent parameter homotopies only need to track as many paths as the number of roots from the *ab initio* system, saving computational effort. Following this strategy, the synthesis system below from which we obtain nearly all roots is for *ab initio* task parameters specified as random complex numbers that do not correspond to anything physical. In Section VI, this *ab initio* solution is used as the start in parameter homotopies for physically relevant target systems.

A. Reformulation of Synthesis Equations

FRG is heuristic and does not rely on root counts from algebraic geometry. Therefore, rather than moving forward with the formulation of our synthesis system presented in (22), (24)–(26), (29)–(33), we choose a formulation with fewer equations and unknowns, (18)–(22). This formulation is further reduced by substituting $\bar{U}_j = 1/U_j$, $j = 1, \ldots, 7$ into (18)–(20) to eliminate seven more variables. Although these equations are not polynomial, FRG applies without modification. We name this system $\mathcal{S}$, its variables $\mathbf{z}$, and its defining parameters $\mathbf{q}$

$$\mathcal{S}(\mathbf{z}, \mathbf{q}) = 0. \tag{35}$$

The vector of variables $\mathbf{z}$ contains the unknowns we intend to solve for

$$\begin{aligned} &\mathbf{z} = \{\mathbf{A}, \mathbf{U}\} \\ \text{where} \quad &\mathbf{A} = \{A, B, C, D, F, G, H, \bar{A}, \bar{B}, \bar{C}, \bar{D}, \bar{F}, \bar{G}, \bar{H}\} \\ &\mathbf{U} = \{U_1, \ldots, U_7\}. \end{aligned} \tag{36}$$

The vector of parameters $\mathbf{q}$ defines the system we are attempting to solve

$$\begin{aligned} &\mathbf{q} = \{\mathbf{P}, \bar{\mathbf{P}}, \mathbf{Q}, \bar{\mathbf{Q}}\} \\ \text{where} \quad &\mathbf{P} = \{P_0, \ldots, P_7\}, \qquad \bar{\mathbf{P}} = \{\bar{P}_0, \ldots, \bar{P}_7\} \\ &\mathbf{Q} = \{Q_1, \ldots, Q_7\}, \qquad \bar{\mathbf{Q}} = \{\bar{Q}_1, \ldots, \bar{Q}_7\}. \end{aligned} \tag{37}$$

B. Constructing Start Systems

Each FRG iteration begins with the generation of a single startpoint and corresponding start system. A startpoint is defined by a vector of variables $\mathbf{z}$ (36) and a start system is defined by a vector of parameters $\mathbf{q}$ (37). To generate a pair $(\mathbf{z}, \mathbf{q})$ that satisfies (35), some values are randomly generated and others are computed from those values. First, random complex values are assigned to $\mathbf{A}$, P_0, and $\bar{P}_0$. Values were chosen from a square in the complex plane with corners $-5-5i$ and $5+5i$ over a uniform distribution. Variables with the overbar notation were not chosen to be conjugate to their nonoverbar counterparts, but were also randomly selected, abandoning their physical meaning. We assume this provides a sufficiently general well-distributed set of startpoints.

Although physically relevant values were not used, the rest of the variables and parameters within $\mathbf{z}$ and $\mathbf{q}$ are computed from the forward kinematics equations of a physical six-bar model. Physical or not, the result still satisfies (35), which is our goal at the moment. As a six-bar is a one degree-of-freedom linkage, in order to extract random configurations from its motion, we must provide random values for a single configuration parameter. We choose $R_j\bar{U}_j, j = 1, \ldots, 7$, the complex number associated with angle $\rho_j - \mu_j$, because it makes the forward kinematics computations simple and fast. It may be noted that the complexity of the solution process of the forward kinematics is dependent on the choice of a specified configuration parameter. Perhaps intuitively, $R\bar{U}$ is a good choice because it defines one angle of the floating four-bar $DGHF$, which segments calculations to one variable at a time after that. Solving the forward kinematics equations provides all remaining values in $\mathbf{z}$ and $\mathbf{q}$. A homotopy path may then be tracked from the startpoint $\mathbf{z}$ that solves the system defined by $\mathbf{q}$ to a finite root that solves a target system defined by a different set of parameters.

FRG path tracking begins with startpoints and parameters that are complex and have no conjugate structure, then terminates to endpoints and parameters that are complex and have no conjugate structure. Furthermore, no special structure develops anywhere along homotopy paths with probability one by virtue of the "gamma trick" [46]. All homotopy paths are nicely behaved and the Jacobian matrix stays nonsingular throughout. Following this, the *ab initio* roots are considered general as they have no structure that enforces physical manifestations.

These roots supply a nearly complete set of startpoints ideal for ensuing repeated parameter homotopies. It is during this second step that homotopy paths are set to track to physically meaningful specifications of $\mathbf{q}$. The *ab initio* roots serve as startpoints and a percentage of them track to endpoints with physical manifestations. The special structure enforced by a physically relevant specification of $\mathbf{q}$ does not appear until the very end of the homotopy path.

In addition, we emphasize that many such parameter homotopies can be computed from the general *ab initio* roots. Each set will correspond to a different physical task and could have a different number of physically relevant solutions. If the *ab initio* system had only physically relevant roots stored from the onset, then the ensuing parameter homotopies would be problematic. First, because the physically relevant roots only represent a subset of all finite roots, and second, because there is only a small chance that this physically relevant subset would connect to the physically relevant roots of another physically relevant system. Therefore, we did not pursue this route.

C. Cognate Structure

The roots of $\mathcal{S}$ are organized into the structure afforded by the linkage cognates of a Stephenson II timed curve generator. More specifically, we are dealing with a Stephenson II where the trace point is connected to one of the floating binary links. We also could have attached the trace point to ternary link CGH. In this case, the mechanism would still be useful but the results here do not apply.

Dijksman has enumerated the timed curve cognates we seek [52, p. 183]. There are four. Their geometric construction is outlined in [55]. Here, we present their equations.

Given a linkage defined by $\mathbf{X}$

$$\mathbf{X} = \{A, B, C, D, F, G, H, P_0\} \tag{38}$$

its first two cognates are called the 1/2-cognate and 3/4-cognate.

To obtain the 1/2 cognate, we define a map $\kappa_{1/2} : \mathbb{C}^8 \mapsto \mathbb{C}^8$

$$\kappa_{1/2}(\mathbf{X}) := \begin{Bmatrix} \Upsilon(A) \\ B \\ \Upsilon(C) \\ (\Upsilon \circ \Xi)(D) \\ B + P_0 - F \\ (\Upsilon \circ \Xi)(G) \\ \Upsilon(H) \\ P_0 \end{Bmatrix} \tag{39}$$

$$\begin{aligned} \text{where} \quad \Xi(z) &:= \frac{B-F}{D-F}(z-H) + H \\ \Upsilon(z) &:= \frac{P_0-F}{H-F}(z-B) + B. \end{aligned}$$

To obtain the 3/4 cognate, we define a map $\kappa_{3/4} : \mathbb{C}^8 \mapsto \mathbb{C}^8$

$$\kappa_{3/4}(\mathbf{X}) := \begin{Bmatrix} (\Lambda \circ \Gamma)(A) \\ B \\ (\Lambda \circ \Gamma)(C) \\ \Lambda(D) \\ F \\ \Lambda(D + H - G) \\ \Lambda(H) \\ P_0 \end{Bmatrix} \tag{40}$$

$$\begin{aligned} \text{where} \quad \Gamma(z) &:= \frac{H-G}{C-G}(z-D) + D \\ \Lambda(z) &:= \frac{B-F}{\Gamma(B)-F}(z-F) + F. \end{aligned}$$

To gain some geometric intuition on Ξ, Υ, Γ, and Λ, note that they take the form of a stretch-rotation about a pole. For example, Γ may be interpreted as a rotation of $\angle HGC$[2] about D combined with a dilation by $|H-G|/|C-G|$ centered on D. Both $\kappa_{1/2}$ and $\kappa_{3/4}$ are their own inverses

$$\kappa_{1/2}^{-1} \equiv \kappa_{1/2}, \qquad \kappa_{3/4}^{-1} \equiv \kappa_{3/4}. \tag{41}$$

The final cognate is called the 1/2-3/4 cognate. It is obtained from the commutative composition of $\kappa_{1/2}$ and $\kappa_{3/4}$

$$\kappa_{1/2\text{-}3/4} := \kappa_{1/2} \circ \kappa_{3/4} \equiv \kappa_{3/4} \circ \kappa_{1/2}. \tag{42}$$

Following (39), (40), and (42), the set of six-bars defined by

$$\left\{\mathbf{X},\ \kappa_{1/2}(\mathbf{X}),\ \kappa_{3/4}(\mathbf{X}),\ (\kappa_{1/2} \circ \kappa_{3/4})(\mathbf{X})\right\} \tag{43}$$

all trace the exact same coupler curve with the exact same angular displacement of crank AC.

These equations demonstrate how to transform a linkage into its cognates. To be useful for FRG, we need to transform a solution $\mathbf{z}$ of the form shown in (36) to its cognate solutions.

For the 3/4-cognate solution, we need only to append conjugate unknowns and duplicate the vector $\mathbf{U}$ (36)

$$\mathbf{z}_{3/4} = \left\{\kappa_{3/4}(\mathbf{X}),\ \kappa_{3/4}(\bar{\mathbf{X}}),\ \mathbf{U}\right\}. \tag{44}$$

For the 1/2- and 1/2-3/4 cognate solutions, we append conjugate unknowns and transform the vector $\mathbf{U}$ according to the geometric construction given in [55]

$$\begin{aligned}
\mathbf{z}_{1/2} &= \left\{\kappa_{1/2}(\mathbf{X}),\ \kappa_{1/2}(\bar{\mathbf{X}}),\ \mathbf{U}'\right\} \\
\mathbf{z}_{1/2\text{-}3/4} &= \left\{(\kappa_{1/2} \circ \kappa_{3/4})(\mathbf{X}),\ (\kappa_{1/2} \circ \kappa_{3/4})(\bar{\mathbf{X}}),\ \mathbf{U}'\right\} \\
\text{where } \mathbf{U}' &= \{U_1', \ldots, U_7'\}, \\
U_j' &= \frac{P_j - B - U_j(P_0 - F)}{F - B}, \quad j = 1, \ldots, 7.
\end{aligned} \tag{45}$$

Equations (45) only apply to cases, where corresponding components Q_j and $\bar{Q}_j$ of $\mathbf{q}$ (37) are defined to be reciprocal. This is due to certain simplifications taken in Section IV. The start systems constructed in Section V-B and the target system constructed for the *ab initio* system were defined with Q_j and $\bar{Q}_j$, $j = 1, \ldots, 7$ as unrelated random complex numbers. Therefore, only (44) applies to them, reducing the size of our cognate sets to two. Defining Q_j and $\bar{Q}_j$ as reciprocal for start systems was possible, but this increased the numerical precision needed for path tracking such that the cost outweighed the benefit. With cognate sets of size two, the expected amount of computation is cut in half. For every finite root found, its cognate partner is readily computed from (44). To find nearly all roots, we only must find nearly all cognate sets, of which there are half as many.

D. Computational Work

Using the start system construction method of Section V-B and exploiting the cognate structure of Section V-C, FRG was applied to find the *ab initio* roots $\mathbf{z}$ of a numerically general target system $\mathcal{S}(\mathbf{z}, \mathbf{q}_1) = 0$. Target parameters $\mathbf{q}_1$ were randomly generated and are printed in Appendix B. After generating this *ab initio* solution, these roots can be used repeatedly in conjunction with the parameter homotopy method to obtain isolated roots for any other system $\mathcal{S}(\mathbf{z}, \mathbf{q}_2) = 0$, where $\mathbf{q}_2$ might be defined by a specific synthesis task. In Section VI, we will define $\mathbf{q}_2$ according to analyses in Section III, to design a novel machine.

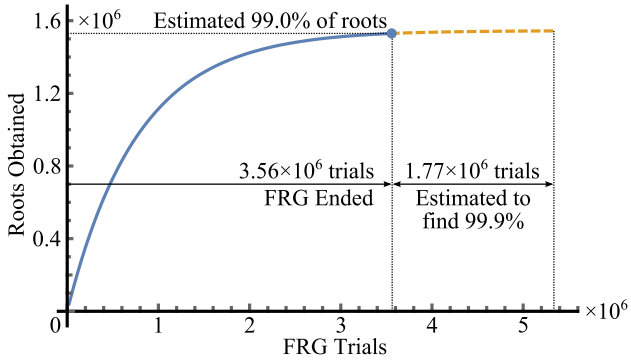


Fig. 7. Accumulation of roots over the procession of the FRG algorithm. An estimated 99.0% of roots were found in 3.56×10^6 trials. To find 99.9% would take about 50% more computational effort.

A single FRG iteration consists of generating a random startpoint/start system (see Section V-B), tracking a homotopy path from this startpoint to an endpoint root of the target system, computing the cognate set of that endpoint (see Section V-C), and finally checking whether the cognate set found is a duplicate from a previous iteration.[3] By tracking the occurrence of duplicates, and assuming equal probability of happening upon any root of the target set during each iteration, an in-process estimate of the number of finite roots found is computed. The algorithm ends when this estimate is sufficiently high.

Tracking a homotopy path involves following the values of root $\mathbf{z}$ from its startpoint $\mathbf{z} = \mathbf{z}_0$ that solves the start system $\mathcal{S}(\mathbf{z}, \mathbf{q}_0) = 0$ to its endpoint $\mathbf{z} = \mathbf{z}_1$ that solves the target system $\mathcal{S}(\mathbf{z}, \mathbf{q}_1) = 0$. Path tracking was performed using a predictor–corrector technique based on Runge–Kutta and Newton iterations. This technique is presented in Appendix C.

FRG was written in CUDA and run on a laptop GPU for 3 563 520 trials over 24 h. It found 764 894 distinct cognate sets indicating 1 529 788 finite roots. Of these, 1180 roots were suspected of being singular due to high condition numbers and so were discarded, leaving 1 528 608 nonsingular roots. According to the FRG estimation technique, this is 99.0% of roots. We note that this estimate assumes random startpoints were sufficiently distributed. Benchmark work in [54] gives evidence that this is a good assumption. But as the six-bar problem of this article has not been characterized in the past, our assumption may prove faulty in the end. About 0.14% of paths ended in numerical failure, usually from the Newton corrector step not converging due to precision limitations. Fig. 7 displays root accumulation over FRG iterations. The diminishing returns of FRG are documented

[2]To be more precise, this is the angle that rotates $\overline{GH}$ to lie parallel with $\overline{GC}$ (not vice versa). The "$\angle$" notation does not convey direction.

[3]A slight improvement can be made if cognate construction is computed only after a root is shown not to be a duplicate.

TABLE III
SPECIFICATION OF $\mathbf{q}_{\text{TASK}}$ FOR THE LEG MECHANISM EXAMPLE

j	P_j (cm)	ϕ_j (rad)
0	0.09504886+0.06372449i	0
1	0.90856802−0.19472945i	0.04956958
2	1.90242974−0.28657681i	0.11939315
3	3.52297908−0.26708542i	0.35571795
4	5.13673683−0.11927755i	0.99925301
5	4.18184656+0.74327464i	2.29135153
6	2.19553111+1.48251591i	4.08970859
7	0.60702434+1.66736265i	5.46542362

in [54]. We estimate the algorithm would have found 99.9% of roots with 1.77×10^6 more trials.

VI. APPLICATION TO LEG MECHANISM

The *ab initio* implementation of FRG presented in Section V-D found nearly all finite roots for a numerically general system $\mathcal{S}(\mathbf{z}, \mathbf{q}_1) = 0$. This root set can be used to efficiently obtain nearly all isolated roots for systems of the form $\mathcal{S}$ henceforth. Parameter homotopies can be constructed that use these roots as startpoints to track to the roots of some other systems of the form $\mathcal{S}(\mathbf{z}, \mathbf{q}_2) = 0$. However, the vector of parameters $\mathbf{q}$ [defined in (37)] for these systems were composed of physical specifications informed by the analysis of Section III.

When physically relevant values of $\mathbf{q}$ are specified, some of the roots $\mathbf{z}$ [defined in (36)] of system $\mathcal{S}$ will be physically relevant as well. These roots contain the pivot locations (design parameters) of six-bar linkages that produce the kinematic characteristics encoded in $\mathbf{q}$. These kinematic characteristics were chosen to instantiate some dynamic behavior as informed by simplified simulations (see Section III). Bringing this all together forms a design process capable of finding six-bar linkages that produce some dynamic behavior under known loading conditions. In this way, the *ab initio* root set obtained in Section V-D serves as an instrument for designing dynamic machines.

To demonstrate this process, we design a leg mechanism for a small running robot capable of producing propulsive motions with energetics beyond its motor's power limit. We continue from the task specification introduced in Section III, where we established a cyclic path and coordinated crank angle function that would dynamically modulate the power output of a series-elastic actuator under load to increase the kinetic energy during push-off of a single stance phase. This task was transferred to a vector $\mathbf{q}_{\text{task}}$ by picking eight end effector positions (x_j, y_j) and coordinated crank angles ϕ_j, printed in Table III. Components of $\mathbf{q}_{\text{task}}$ are specified as

$$P_j = x_j + iy_j, \qquad \bar{P}_j = x_j - iy_j, \qquad j = 0, \ldots, 7$$
$$Q_j = e^{i\phi_j}, \qquad \bar{Q}_j = e^{-i\phi_j}, \qquad j = 1, \ldots, 7. \quad (46)$$

Note that $(Q_0, \bar{Q}_0)$ is automatically specified as (1,1) without loss of generality, making the number of angles eight.

All isolated roots of the system $\mathcal{S}(\mathbf{z}, \mathbf{q}_{\text{task}}) = 0$ were obtained by setting up a parameter homotopy with the *ab initio* roots as startpoints. However, instead of computing 1 528 608 paths, one root from each cognate set was tracked, cutting this number in half. Only one root is needed to establish its cognate set of roots of the target system at the end of a homotopy. The parameter homotopy had a higher path failure rate of 41%. Of these failures, 91% were triggered by the Jacobian condition number rising above 10^{18}, which occurred on average at $t = 0.997$ (± 0.037 S.D.). Although it is not certain, this is an indicator that those paths might have been headed toward singular endpoints, which often are not useful for engineering applications. Singular endpoints are usually more common when $\mathcal{S}$ is defined with additional structure, such as the case with $\mathbf{q}_{\text{task}}$. A histogram of endpoint condition numbers is shown in Fig. 8. Informed by this, we chose a condition number of 10^{15} to separate endpoints considered nonsingular and singular.

There were 419 576 nonsingular cognate pairs of roots found. A subset Ω of size 1861 were physically relevant, that is when variables and their overbar counterparts, e.g., $(A, \bar{A})$, are conjugate. Recall from Section V-C that when parameters Q_j and $\bar{Q}_j$ are reciprocal, such as with the members of Ω, the cognate operator $\kappa_{1/2}$ extends the cognate sets from size two to four. Therefore, we should expect the cognate pairs of Ω to divide into half as

TABLE IV
DESIGN VARIABLES CORRESPONDING TO MECHANISMS (a)–(d), SHOWN IN Fig. 9

(a)	(b)
$A = 4.02063935+3.79240210i$	$A = -2.35985445+7.68751126i$
$B = 7.89860362+4.97580510i$	$B = 7.89860362+4.97580510i$
$C = 3.73248379+3.06554709i$	$C = -4.00265968+6.46756247i$
$D = 5.50640571+6.68329005i$	$D = 4.92974721+12.07139758i$
$F = 5.82450935+5.21859158i$	$F = 5.82450935+5.21859158i$
$G = 3.86373172+7.93708335i$	$G = -2.07604818+1.41429440i$
$H = 2.54366227+5.50642013i$	$H = -9.13830033+7.81064704i$
$P_0 = 0.09504886+0.06372449i$	$P_0 = 0.09504886+0.06372449i$
(c)	(d)
$A = 2.57830162+7.47750629i$	$A = 0.20753471+6.95450040i$
$B = 7.47421398+4.87297294i$	$B = 7.47421398+4.87297294i$
$C = 1.49446501+7.03351060i$	$C = -1.11124475+6.05474088i$
$D = 4.61659816+3.34161892i$	$D = 4.09961571+2.01979554i$
$F = 5.59922115+4.64881064i$	$F = 5.59922115+4.64881064i$
$G = 2.83013468+4.64176203i$	$G = 1.63963702+4.82891573i$
$H = 1.70261615+6.31480581i$	$H = -1.94332603+6.79984649i$
$P_0 = 0.09504886+0.06372449i$	$P_0 = 0.09504886+0.06372449i$

Complex numbers indicate x-y pivot locations in home configurations. Dimensions are cm.

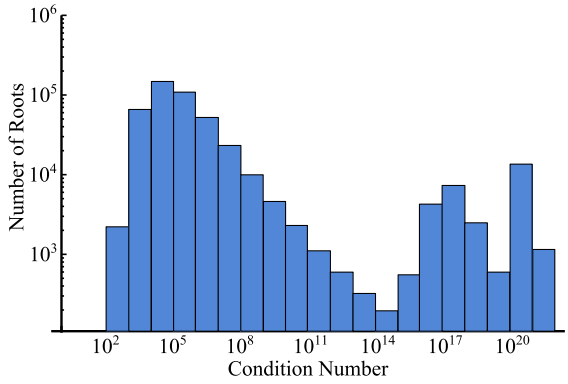


Fig. 8. Condition numbers of Jacobian matrices at the roots found during the leg mechanism example. These Jacobian matrices contain derivatives of the synthesis equations with respect to synthesis variables. They were not normalized beforehand. Roots with condition numbers greater than 10^{15} were assumed to indicate singular points.

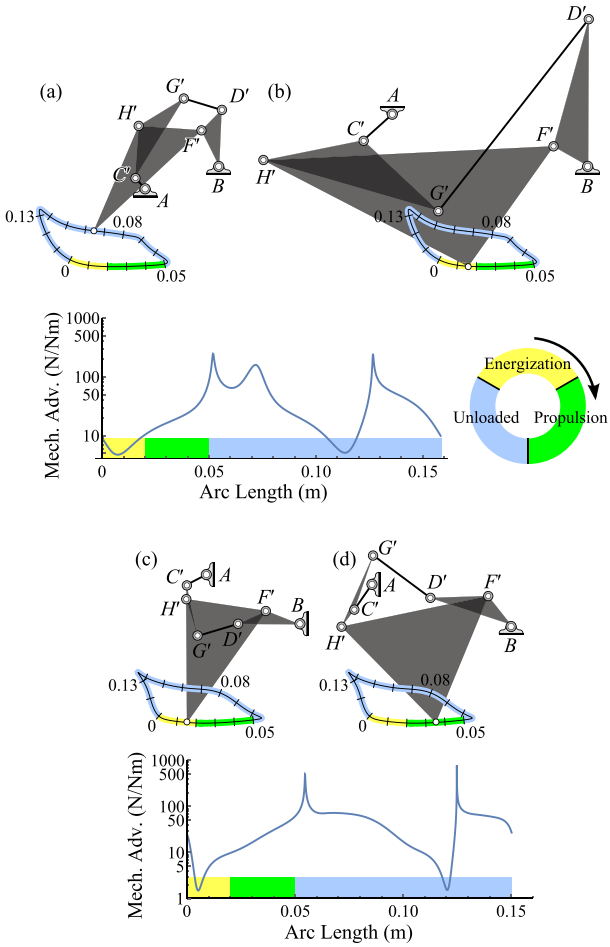


Fig. 9. Linkage designs that exhibit the desired dynamic behavior. Linkages (a) and (b) are cognates that trace the exact same path with the same coordination of mechanical advantage over arc length. The same goes for (c) and (d). Moving pivots are drawn in displaced positions C', D', F', G', H'. Home position values are listed in Table IV.

many cognate foursomes. However, $1861/2 \notin \mathbb{N}$, indicating a problem. Sorting out Ω, we instead found 920 cognate foursomes and 21 incomplete cognate pairs. Missing cognates were most likely lost during numerical work, but are readily producible in postprocess using the cognate transforms of Section V-C. Expanding out all 941 cognate foursomes obtains 3764 linkage designs.

These linkage designs are susceptible to a known list of linkage defects: circuit, branch, order, and full rotatability defects [56]. To sort these out, all designs were kinematically analyzed. Occasionally designs were found to possess circuit defects, but were still deemed useful. That is, if a mechanism circuit missed one or two task points but still approximated the motion that $\mathbf{q}_{\text{task}}$ was chosen to approximate in the first place, that mechanism was considered useful. In addition, linkages must have pivots in acceptable locations (e.g., not underground) and ideally be compact. A few linkage designs which satisfy these criteria are drawn in Fig. 9. The linkages shown in Fig. 9(a) and (b) are cognates, likewise with 9(c) and (d). That being the case, they produce the same point path with the same coordination of mechanical advantage across that path. Therefore, the plots printed under each pair in Fig. 9 describe both linkages above.

The cycling foot path is broken into three regions. First, the foot point makes ground contact and mechanical advantage decreases, nearly stalling motion and energizing the series-elastic element. Second, while still in stance mechanical advantage increases, converting elastic energy into high-powered propulsion. And third, the foot goes out of contact and recycles back to its starting position. Although only a single low mechanical advantage region was desired, a second is seen in the unloaded regions of Fig. 9. It was common to nearly all designs that had the foot path entirely below the ground pivots that installing a low mechanical advantage region on one part of the curve begot another low mechanical advantage region elsewhere. It is unclear whether it is possible to exclude this second region and maintain pivots above the foot path. This characteristic might be intrinsic of the design space. Since the second low spot appears in the unloaded region, it is not problematic.

VII. Sensitivity Analysis

A feature of the proposed approach is to survey the design space and produce multiple options. There are many criteria by which designs can be evaluated. Foremost, we consider dynamic behavior. Another important consideration is a design's sensitivity to fabrication errors. Since our design specifications involve instantiating a configuration space with near singular conditions, there is a good chance the resulting designs will be sensitive to dimensional errors. An analysis was conducted to compare the four designs shown in Fig. 9.

First, the combined effect of fabrication errors in the planar model was considered. Perturbations were introduced into the 14-D space made up of the seven planar pivot locations $\{A, B, C, D, F, G, H\}$. Normal distributions were defined inside circles centered at each pivot. In polar coordinates, deviation radii were selected with a standard deviation of 0.1 cm from the pivots' true location and deviation angles were selected with equal probability. For each design shown in Fig. 9, the motion of 1000 perturbed linkages was computed. For (a) and (b), mechanical advantage of the perturbed linkages is plotted in Fig. 10 over the course of their motion. In particular, we care to keep track of the minimum absolute value of mechanical advantage in the energization region. For brevity, we will refer to this as the *low MA value*. This minimum is important because it dictates the maximum amplification of ground forces through the mechanism into a reaction torque acting on the motor. Consequently, it effects how near the motor gets to producing its stall torque and how much energy might be transiently stored in the spring for a high-powered motion.

Fig. 10 identifies branch defective curves (which pass through zero) and defect free curves alongside the ideal form. The low MA value of the ideal linkage is indicated with a star. At times, mechanical advantage passed through zero. This indicates that a perturbed linkage encountered a singular configuration and is thus branch defective. For our application, these dimensional errors would be problematic. In application, they would bring the motor to stall as the linkage jams. On the other hand, if the low

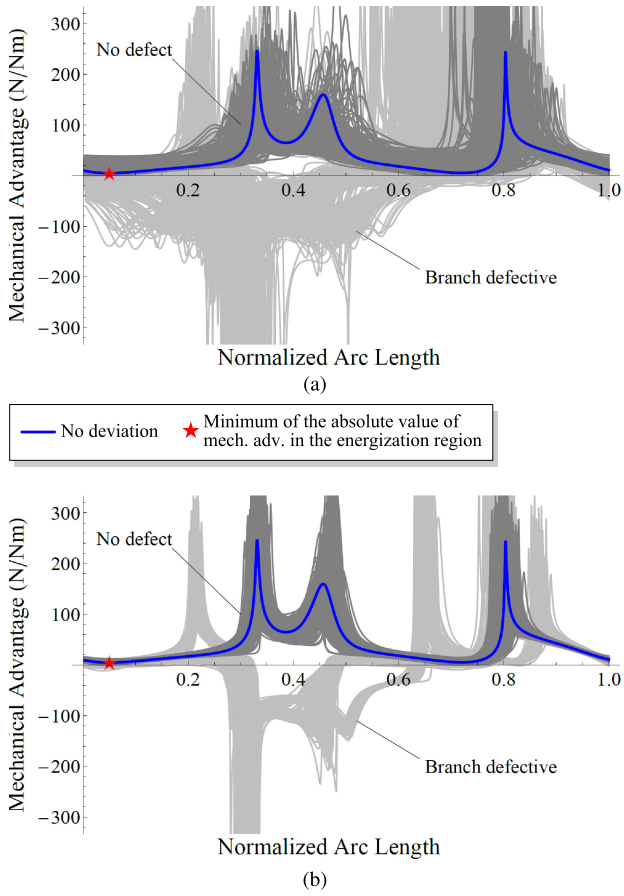


Fig. 10. Mechanical advantage profiles for mechanisms randomly deviated from the true pivot locations of mechanisms (a) and (b) shown in Fig. 9. Random noise in pivot positions simulates fabrication errors. Pivot locations were varied according to normal distributions centered at their true location with a standard deviation of 1 mm. A quantity of 1000 samples were generated for each plot. Sample mechanisms that possess a zero mechanical advantage configuration are branch defective. Statistics corresponding to these plots are given in Table V.

TABLE V
EFFECT OF RANDOM DEVIATIONS OF PIVOTS FROM THEIR TRUE POSITIONS FOR MECHANISMS (a)–(d) SHOWN IN Fig. 9

	(a)	(b)	(c)	(d)
Probability of branch defect	59.1%	22.8%	64.8%	48.6%
Average of the minimum absolute value of mechanical advantage in the energization region* (N/Nm)	13.3	5.2	6.5	4.7
Standard deviation of the above statistic (N/Nm)	9.4	2.1	3.8	3.1

*Excluding branch defective linkages.
Statistics correspond to the data plotted in Fig. 10.

MA value is too high, power modulation will not occur. Statistics corresponding to Fig. 10 are shown in Table V. They indicate a 22.8% chance for random dimensional errors of mechanism (b) to result in a branch defect. On the other hand, design (a) is higher with 59.1%.

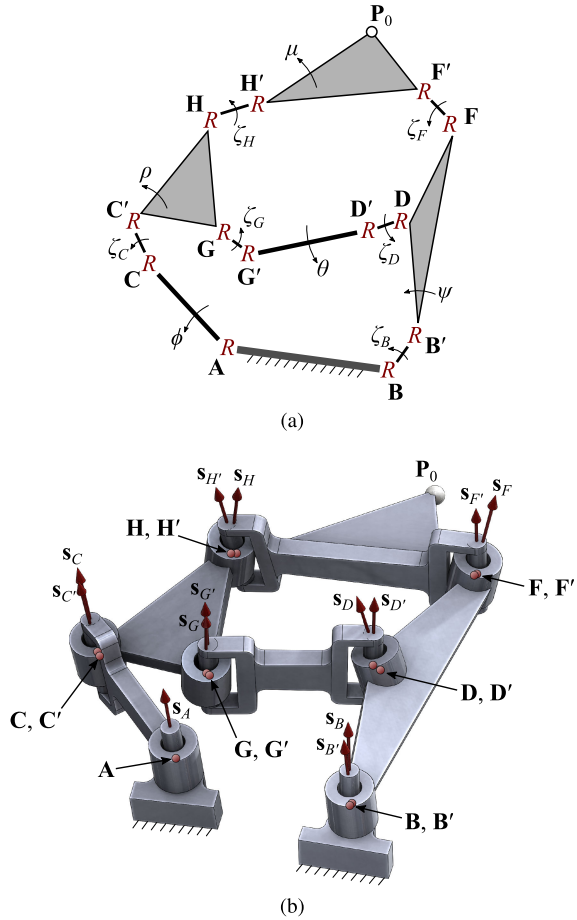


Fig. 11. Spatial 12-bar linkage used to understand spatial joint misalignments in a planar six-bar. (a) Schematic denotes 13 revolute (R) joints and plainly shows connectivity. (b) More realistic 3-D rendering illustrates misaligned joint axes.

To create a more realistic error model, we also considered the effect that spatial misalignments and joint play have on the low MA value. References on the sensitivity of parallel manipulators to fabrication errors include [57]–[59]. For our model, all but one revolute joint was split into two nonparallel, nonintersecting joint axes. This yields a 12-bar spatial one degree-of-freedom mechanism, shown in Fig. 11. This model is defined by 13 screw axes. The directional vectors for these screw axes are $\mathbf{s}_A$, $\mathbf{s}_B$, $\mathbf{s}_{B'}$, $\mathbf{s}_C$, $\mathbf{s}_{C'}$, $\mathbf{s}_D$, $\mathbf{s}_{D'}$, $\mathbf{s}_F$, $\mathbf{s}_{F'}$, $\mathbf{s}_G$, $\mathbf{s}_{G'}$, $\mathbf{s}_H$, and $\mathbf{s}_{H'}$, refer to Fig. 11. Reference points that locate each screw axis are represented by 3-D vectors $\mathbf{A}$, $\mathbf{B}$, $\mathbf{B}'$, $\mathbf{C}$, $\mathbf{C}'$, $\mathbf{D}$, $\mathbf{D}'$, $\mathbf{F}$, $\mathbf{F}'$, $\mathbf{G}$, $\mathbf{G}'$, $\mathbf{H}$, and $\mathbf{H}'$. Corresponding joint rotations around each axis are ϕ, ζ_B, ψ, ζ_C, ρ, ζ_D, θ, ζ_F, ζ_G, ζ_H, and μ. The angles ϕ, ψ, ρ, θ, and μ roughly correspond to their planar analogues in Section IV, except that they are relative instead of absolute. The new ζ angles correspond to new small links that represent joint deviations. $\mathbf{P}_0$ locates the endpoint in a home configuration.

To model this spatial linkage, first we define the screw matrix

$$[T(\theta, \mathbf{s}, \mathbf{c})] := \begin{bmatrix} [R(\theta, \mathbf{s})] & ([I] - [R(\theta, \mathbf{s})])\mathbf{c} \\ 0 & 1 \end{bmatrix} \tag{47}$$

where $[R(\theta, \mathbf{s})]$ is defined in equation (48) which appears at the bottom of this page. The two loop closure equations are then

$$[T_1]\begin{Bmatrix}\mathbf{G}'\\1\end{Bmatrix} - [T_2]\begin{Bmatrix}\mathbf{G}'\\1\end{Bmatrix} = 0 \tag{49}$$

and

$$[T_3]\begin{Bmatrix}\mathbf{F}'\\1\end{Bmatrix} - [T_4]\begin{Bmatrix}\mathbf{F}'\\1\end{Bmatrix} = 0 \tag{50}$$

where

$$\begin{aligned}[T_1] &= [T(\phi, \mathbf{s}_A, \mathbf{A})][T(\zeta_C, \mathbf{s}_C, \mathbf{C})]\\ &\quad\times [T(\rho, \mathbf{s}_{C'}, \mathbf{C}')][T(\zeta_G, \mathbf{s}_G, \mathbf{G})]\\ [T_2] &= [T(\zeta_B, \mathbf{s}_B, \mathbf{B})][T(\psi, \mathbf{s}_{B'}, \mathbf{B}')]\\ &\quad\times [T(\zeta_D, \mathbf{s}_D, \mathbf{D})][T(\theta, \mathbf{s}_{D'}, \mathbf{D}')]\\ [T_3] &= [T(\phi, \mathbf{s}_A, \mathbf{A})][T(\zeta_C, \mathbf{s}_C, \mathbf{C})][T(\rho, \mathbf{s}_{C'}, \mathbf{C}')]\\ &\quad\times [T(\zeta_H, \mathbf{s}_H, \mathbf{H})][T(\mu, \mathbf{s}_{H'}, \mathbf{H}')]\\ [T_4] &= [T(\zeta_B, \mathbf{s}_B, \mathbf{B})][T(\psi, \mathbf{s}_{B'}, \mathbf{B}')][T(\zeta_F, \mathbf{s}_F, \mathbf{F})].\end{aligned} \tag{51}$$

In addition, the orientation equations should be satisfied

$$[R_1]\mathbf{s}_{G'} - [R_2]\mathbf{s}_{G'} = 0 \tag{52}$$

and

$$[R_3]\mathbf{s}_{F'} - [R_4]\mathbf{s}_{F'} = 0 \tag{53}$$

where $[R_i]$ is the corresponding upper left rotation matrix of $[T_i]$. Equations (49) and (50) provide three equations each. The *x, y* components of (52) and (53) provide two equations each, for a total of 10 equations that govern the motion of the 11 joint axis angles. We name the governing equations $\mathbf{f}$.

Mechanical advantage of the spatial linkage is computed by taking the derivative of $\mathbf{f}$

$$\left[\frac{\partial \mathbf{f}}{\partial \boldsymbol{\theta}}\right]\dot{\boldsymbol{\theta}} = 0$$

$$\text{where}\quad \boldsymbol{\theta} = \{\phi, \rho, \psi, \theta, \mu, \zeta_B, \zeta_C, \zeta_D, \zeta_F, \zeta_G, \zeta_H\}. \tag{54}$$

The Jacobian matrix above is 10×11. Equation (54) is a homogeneous linear system. Its one-dimensional null space describes the joint velocity ratios at a particular configuration. By taking the derivative of

$$[T_3]\mathbf{P}_0 = \mathbf{P} \tag{55}$$

and substituting the solution of (54) into it, one may obtain values for $\dot{\phi}$ and $\dot{\mathbf{P}}$. Mechanical advantage is computed as

$$\text{MA} = \frac{\dot{\phi}}{|\dot{\mathbf{P}}|}. \tag{56}$$

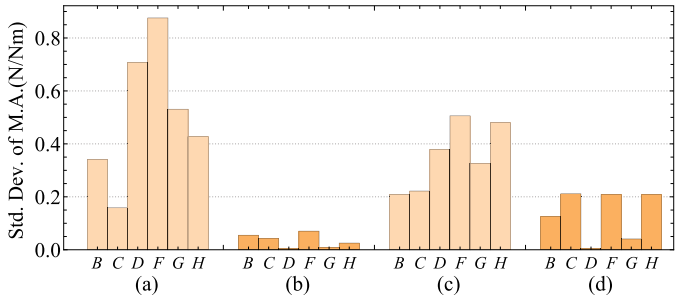


Fig. 12. Standard deviation provides a measure of sensitivity that each joint of each design has toward misalignment errors. Labels (a)–(d) correspond to the designs of Fig. 9.

Each planar design of Fig. 9 was taken at its low MA configuration and perturbed into a spatial 12-bar of the likes of Fig. 11. Perturbations took place in an axis-by-axis manner. A single perturbation split a joint axis apart in some direction by a distance r, then skewed these joint axes away from each other in some direction by an angle α. Distance r was varied from 0 to 3 mm and angle α was varied from 1° to 15°. Choosing 15 increments for r, α, and their respective directions lead to a population of 50 625 deviations for each joint axis. Additional random 0.5° perturbations were added to all other screw axes to keep each linkage spatially general. Mechanical advantage was computed according to (56). Variation in the value of MA is displayed in Fig. 12 for spatial perturbations of each joint of each design in Fig. 9. Standard deviations show that design (a) is the most sensitive to misalignments and slop within its joints. In particular, its joints D and F are the most sensitive. Design (b) is less sensitive to these dimensional errors.

In practice, designs should be fabricated with a precision adjustment to compensate for their sensitivity. A single dimensional deviation can often be identified to calibrate a linkage. Adjustment can lead a design from branch defective to low and high MA values. Of course, sensitivity to joint errors is only one measure of a design's worth. Other selection criteria include the packaging footprint and packaging convenience of a mechanism. The former considers the space occupied by the mechanism while the latter considers how complicated links need to be to avoid interferences. Despite its sensitivity, design (a) (see Fig. 9) is a strong candidate. Simulation of its motion takes a time evolution similar to Fig. 5 using the parameters of Table II, and reveals this mechanism is able to deliver kinetic energy at peak powers up to 3.0 times that of the motor's maximum.

VIII. Conclusion

In this article, we presented a method for designing the dynamic behavior of a loaded mechanism. First, we analyzed a

$$[R(\theta, \mathbf{s})] := \begin{bmatrix} c\theta + s_x^2(1-c\theta) & s_x s_y(1-c\theta) - s_z s\theta & s_x s_z(1-c\theta) + s_y s\theta \\ s_y s_x(1-c\theta) + s_z s\theta & c\theta + s_y^2(1-c\theta) & s_y s_z(1-c\theta) - s_x s\theta \\ s_z s_x(1-c\theta) - s_y s\theta & s_z s_y(1-c\theta) + s_x s\theta & c\theta + s_z^2(1-c\theta) \end{bmatrix} \tag{48}$$

simplified model of our proposed system in order to discover the kinematic characteristics that would result in the desired dynamics. The model was run according to user-defined kinematic characteristics in the absence of a mechanism design. Instead, these characteristics serve as a design goal for a to-be-designed mechanism found during a subsequent kinematic synthesis step. The prescribed characteristics included a single degree-of-freedom point path and a coordinated mechanical advantage profile. Kinematic synthesis was executed to discover six-bar linkages that exhibit the desired mechanism characteristics. These characteristics were encoded into design constraints resulting in a highly nonlinear system of design equations. Nearly all isolated roots of this system of equations were obtained for the first time in this article using the FRG technique. The generation of a complete set of isolated roots of such a large system allows an exploration of the design space to be conducted. We applied our methodology to the design of a leg mechanism for a small running robot powered by a series-elastic actuator. The leg mechanism is capable of propelling itself forward at powers beyond its motor's limit while recycling foot position during every stride with the motor only rotating in the forward direction. A few linkage design solutions were presented, and compared against each other according to their sensitivity to errors in the revolute joints.

APPENDIX A
FOURIER COEFFICIENTS

Fourier coefficients for the path displayed in Fig. 3(a).

k	a_k	b_k	c_k	d_k
0	5.0380E-02		1.0006E-02	
1	-2.3035E-02	8.1134E-03	-3.3929E-04	-9.9926E-03
2	1.4741E-04	-7.4818E-04	-1.9118E-03	-1.2695E-03
3	-9.8067E-04	1.7724E-03	-1.2544E-03	-5.7305E-04
4	-1.4305E-04	-6.3621E-04	-4.2266E-04	2.5945E-04
5	6.9575E-05	3.8911E-04	-3.7591E-04	1.0048E-04
6	-3.8162E-04	-2.8942E-04	2.5620E-05	5.2924E-05
7	8.3483E-05	5.0805E-05	-8.7071E-05	6.9878E-05

Fourier coefficients for the mechanical advantage curve displayed in Fig. 3(b).

k	α_k	β_k	k	α_k	β_k
0	4π				
1	-3.3934E+00	-4.3419E+00	9	1.3074E-01	3.7434E-02
2	-6.0540E-01	-8.4776E-01	10	9.7375E-02	2.8083E-03
3	-9.5055E-01	8.3538E-02	11	7.1065E-02	-7.1290E-02
4	-4.9012E-01	3.8890E-02	12	3.5888E-02	-6.8938E-02
5	-2.7556E-01	3.2529E-01	13	-2.9838E-02	-5.8048E-02
6	-1.4824E-01	2.4851E-01	14	-3.3223E-02	-4.2639E-02
7	6.3071E-02	2.2760E-01	15	-4.6873E-02	9.3571E-03
8	8.9012E-02	1.3820E-01	16	-3.7274E-02	-4.0571E-03

APPENDIX B
TARGET PARAMETERS

$P_0 = 4.241713408 - 4.925334736i$	$\bar{P}_0 = -3.76987048 + 0.79723779i$
$P_1 = 1.448005491 - 2.686249989i$	$\bar{P}_1 = -4.28563116 - 0.56502469i$
$P_2 = 2.287911195 - 0.912310849i$	$\bar{P}_2 = -2.16603080 + 0.88360558i$
$P_3 = -1.870502484 - 0.577807741i$	$\bar{P}_3 = 4.57926095 - 2.07401374i$
$P_4 = -0.762532942 + 0.904378820i$	$\bar{P}_4 = 4.80128973 - 1.46072184i$
$P_5 = -4.142822754 + 1.697671323i$	$\bar{P}_5 = 2.04126667 + 2.96214923i$
$P_6 = 1.023004534 - 3.026210036i$	$\bar{P}_6 = 3.98999969 - 0.23282959i$
$P_7 = -2.522295756 + 3.942421532i$	$\bar{P}_7 = -3.99644836 + 4.42685486i$
$Q_1 = 2.315936720 - 0.552780386i$	$\bar{Q}_1 = -0.21802981 + 3.58500421i$
$Q_2 = -3.018911858 + 0.969238414i$	$\bar{Q}_2 = 4.31106320 - 0.03375595i$
$Q_3 = -0.490634736 + 2.994316904i$	$\bar{Q}_3 = 4.93814807 + 4.55170197i$
$Q_4 = 0.239711851 - 1.341440198i$	$\bar{Q}_4 = -4.27121444 + 0.13553499i$
$Q_5 = 4.942917423 - 2.415216683i$	$\bar{Q}_5 = -0.73774348 + 1.83173884i$
$Q_6 = 1.599536874 + 2.367319121i$	$\bar{Q}_6 = 0.88946147 + 0.86574737i$
$Q_7 = 4.229853181 - 1.149104288i$	$\bar{Q}_7 = 4.03675786 + 1.46116468i$

APPENDIX C
HOMOTOPY PATH TRACKING

Here, we present a method for tracking the homotopy paths of a system $\mathbf{f}(\mathbf{z}, \mathbf{q}) = 0$. The vector of variables $\mathbf{z}$ has n components, and the vector of parameters $\mathbf{q}$ has m components. We define parameters for a start system and target system as $\mathbf{q}_0$ and $\mathbf{q}_1$, respectively. Our goal is to track the values of a root $\mathbf{z}$ from a startpoint $\mathbf{z} = \mathbf{z}_0$ that solves the start system $\mathbf{f}(\mathbf{z}, \mathbf{q}_0) = 0$ to an endpoint $\mathbf{z} = \mathbf{z}_1$ that solves the target system $\mathbf{f}(\mathbf{z}, \mathbf{q}_1) = 0$. To do this, we construct a homotopy parameterized by t

$$\mathbf{f}(\mathbf{z}, \mathbf{q}(t)) = 0 \tag{57}$$

$$\text{where} \quad \mathbf{q}(t) = \frac{\gamma(1-t)\mathbf{q}_0 + t\mathbf{q}_1}{\gamma(1-t) + t}. \tag{58}$$

The definition of $\mathbf{q}(t)$ follows the "gamma trick" [45]. At $t = 0$, (57) is the start system. At $t = 1$, it is the target system.

In order to gain some numerical benefits [46], path tracking is done in projective space. Therefore, we define homogeneous coordinates of $\mathbf{z}$ as

$$\mathbf{Z} = [Z_1, \ldots, Z_n, Z_0] \quad \text{where} \quad \begin{Bmatrix} \mathbf{z} \\ 1 \end{Bmatrix} = \mathbf{Z}/Z_0. \tag{59}$$

The modified version of $\mathbf{f}$ that receives $\mathbf{Z}$ as an argument is named $\mathbf{F}$. A linear patch is needed for projective computations, defined as

$$\mathbf{u} \cdot \begin{Bmatrix} \mathbf{Z} \\ t \end{Bmatrix} - 1 = 0 \tag{60}$$

where $\mathbf{u}$ is an $n + 2$ dimensional vector of random complex coefficients. Repackaging $\mathbf{Z}$ and t into a single vector $\mathbf{Y} = \{\mathbf{Z}, t\}$, the homotopy is written as

$$\mathbf{H}(\mathbf{Y}) = \begin{Bmatrix} \mathbf{F}(\mathbf{Z}, \mathbf{q}(t)) \\ \mathbf{u} \cdot \mathbf{Y} - 1 \end{Bmatrix} = 0. \tag{61}$$

$\mathbf{Y}$ may be considered a function of the homotopy path's arc length s. Taking the derivative of $\mathbf{H}$ with respect to s obtains

$$\left[\frac{\partial \mathbf{H}}{\partial \mathbf{Y}}\right]\frac{d\mathbf{Y}}{ds} = 0. \tag{62}$$

Expansion of the Jacobian matrix on the left side of (62) obtains

$$\left[\frac{\partial \mathbf{H}}{\partial \mathbf{Y}}\right] = \begin{bmatrix} \left[\frac{\partial \mathbf{F}}{\partial \mathbf{Z}}\right] & \frac{d\mathbf{F}}{dt} \\ \cdots \mathbf{u}^{\mathrm{T}} \cdots \end{bmatrix}$$

$$\text{where} \quad \frac{d\mathbf{F}}{dt} = \left[\frac{\partial \mathbf{F}}{\partial \mathbf{q}}\right]\frac{d\mathbf{q}}{dt}$$

$$\text{and} \quad \frac{d\mathbf{q}}{dt} = (\mathbf{q}_1 - \mathbf{q}_0)\frac{\gamma}{(\gamma(1-t)+t)^2}. \tag{63}$$

In (63), $[\frac{\partial \mathbf{H}}{\partial \mathbf{Y}}]$ is an $(n+1)\times(n+2)$ matrix, $[\frac{\partial \mathbf{F}}{\partial \mathbf{Z}}]$ is an $n\times(n+1)$ matrix, $\frac{d\mathbf{F}}{dt}$ is an $n\times 1$ vector, $[\frac{\partial \mathbf{F}}{\partial \mathbf{q}}]$ is an $n\times m$ matrix, $\frac{d\mathbf{q}}{dt}$ is an $m\times 1$ vector, and $\mathbf{u}^{\mathrm{T}}$ is a $1\times(n+2)$ vector.

At a given point on a homotopy path, the vector space tangent to the path may be obtained by solving for $\frac{d\mathbf{Y}}{ds}$ in the homogeneous linear system given in (62). Note that the row vectors of $[\frac{\partial \mathbf{H}}{\partial \mathbf{Y}}]$ are normal to the homotopy path so that the solutions of (62) define the 1-D vector space perpendicular to all $n+1$ normal vectors, i.e., tangent to the curve. Numerically, (62) is solved by appending a random row vector to $[\frac{\partial \mathbf{H}}{\partial \mathbf{Y}}]$, appending 1 to the zero vector on the right-hand side, then solving this square linear system. For numerical stability, the appended row vector may be the tangent vector from the previous homotopy step. The solution of this square system can be multiplied by complex scalars to achieve any vector in the $\mathbb{C}^1$ null space of $[\frac{\partial \mathbf{H}}{\partial \mathbf{Y}}]$. For the implementation given here, solutions were chosen to have unit magnitude and were scaled so that the last component of $\frac{d\mathbf{Y}}{ds}$ (corresponding to t) is a positive real number.

This solution process for discovering a tangent vector may be posed as an initial value problem, $\mathbf{Y}' = f(\mathbf{Y})$. At the ith tracking step, a Runge–Kutta method can be applied to this initial value problem to compute from the current value $\mathbf{Y}_c$ an appropriate step size Δs, an estimate of the next value $\tilde{\mathbf{Y}}_n$, and an estimate of the next tangent vector $\tilde{\mathbf{V}}_n$.

This estimate is corrected by solving (61) with Newton's method using $\tilde{\mathbf{Y}}_n$ as the initial guess. Since (61) has one less equation than variables in $\mathbf{Y}$, we modify the process slightly. An extra equation is appended to $\mathbf{H}(\mathbf{Y})$

$$\hat{\mathbf{H}}(\mathbf{Y}) = \begin{Bmatrix} \mathbf{H}(\mathbf{Y}) \\ \tilde{\mathbf{V}}_n \cdot (\mathbf{Y} - \delta\tilde{\mathbf{Y}}_n) \end{Bmatrix} = 0. \tag{64}$$

This is the equation of a plane nearly coincident to $\tilde{\mathbf{Y}}_n$ and normal to $\tilde{\mathbf{V}}_n$. The new variable δ is a small-valued complex scalar that serves as a correction factor to ensure an advancing value of t is computed. Assuming $\tilde{\mathbf{V}}_n$ is a good estimate of the homotopy path's tangent at the $i+1$th step, the intersection of the plane and path should be nearly perpendicular, aiding in numerical stability.

To compute the Newton iteration from $\mathbf{Y}_k$ to $\mathbf{Y}_{k+1}$, first evaluate the inverse of the Jacobian of (64) for $\mathbf{Y}_k$

$$[\,\Gamma\,]_k = \begin{bmatrix} [\frac{\partial \mathbf{H}}{\partial \mathbf{Y}}]_k \\ \tilde{\mathbf{V}}_n^{\mathrm{T}} \end{bmatrix}^{-1}. \tag{65}$$

Note that δ, which has not yet been evaluated, drops out of (65). To evaluate δ, consider the Newton update rule

$$\mathbf{Y}_{k+1} = \mathbf{Y}_k - [\,\Gamma\,]_k\hat{\mathbf{H}}(\mathbf{Y}_k). \tag{66}$$

The last component of (66) can be broken out and written

$$t_{k+1} = t_k - \mathbf{\Gamma}_{k,n+2}\cdot\begin{Bmatrix} \mathbf{H}(\mathbf{Y}_k) \\ \tilde{\mathbf{V}}_n\cdot(\mathbf{Y}_k - \delta\tilde{\mathbf{Y}}_n) \end{Bmatrix} \tag{67}$$

where $\mathbf{\Gamma}_{k,n+2}$ is the last row of $[\,\Gamma\,]_k$. At this point, everything in (67) has been numerically evaluated except t_{k+1} and δ. We set t_{k+1} equal to the last component of $\tilde{\mathbf{Y}}_n$, then solve for δ, which is straightforward. Now, we are able to evaluate $\mathbf{Y}_{k+1}$ in (66) and Newton's method proceeds as usual. The value it converges to is the next homotopy path point $\mathbf{Y}_n$. An accurate tangent vector $\mathbf{V}_n$ is also produced by Newton's method. The predictor–corrector path tracking process is repeated from these points. When $t=1$ is reached, stop conditions for the Newton corrector are set to obtain more accurate solutions.

Occasionally, path tracking fails due to numerical precision limitations. If this error is detected, the algorithm attempts to salvage the computation by switching projective linear patches, (60), before reporting the failure. Patch switching is described in [44].

To create a new patch, a vector $\mathbf{u}_{\text{new}}$ is randomly generated

$$\mathbf{u}_{\text{new}} = \{u_1\ \ldots\ u_n\ u_0\ u_t\}. \tag{68}$$

To continue computations, the vector of homogeneous coordinates $\mathbf{Z}$ must be multiplied by some scalar σ so that it satisfies the new projective patch

$$\mathbf{u}_{\text{new}}\cdot\begin{Bmatrix}\sigma\mathbf{Z}\\ t\end{Bmatrix} - 1 = 0. \tag{69}$$

Solving (69) for σ obtains

$$\sigma = \frac{1-u_t t}{\mathbf{u}_{\text{new}}\cdot\begin{Bmatrix}\mathbf{Z}\\ 0\end{Bmatrix}}. \tag{70}$$

Tracking continues with $\mathbf{Y}$ replaced with $\mathbf{Y}_{\text{new}} = \begin{Bmatrix}\sigma\mathbf{Z}\\ t\end{Bmatrix}$.

Numerical homotopy continuation softwares such as Bertini [48] and PHCpack [60] contain additional features that increase the robustness of path tracking, including adaptive precision, path crossing detection, and endgames [61].

ACKNOWLEDGMENT

The authors would like to thank K. Fearing for construction of the prototype linkage. Any opinions, findings, and conclusions

or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] M. M. Plecnik and R. S. Fearing, "Finding only finite roots to large kinematic synthesis systems," *J. Mechanisms Robot.*, vol. 9, no. 2, pp. 021 005-1–021 005-8, Mar. 2017.

[2] J. Hauenstein, A. Sommese, and C. Wampler, "Regeneration homotopies for solving systems of polynomials," *Math. Comput.*, vol. 80, no. 273, pp. 345–377, 2011.

[3] M. M. Plecnik and J. M. McCarthy, "Computational design of stephenson II six-bar function generators for 11 accuracy points," *J. Mechanisms Robot.*, vol. 8, no. 1, pp. 011 017–011 017–9, Aug. 2015.

[4] E. J. F. Primrose, F. Freudenstein, and B. Roth, "Six-bar motion II. The Stephenson-1 and Stephenson-2 mechanisms," *Arch. Rational Mech. Anal.*, vol. 24, no. 1, pp. 42–72, Feb. 1967.

[5] A. P. Morgan and A. J. Sommese, "Coefficient-parameter polynomial continuation," *Appl. Math. Comput.*, vol. 29, no. 2, pp. 123–160, Jan. 1989.

[6] D. W. Haldane, M. M. Plecnik, J. K. Yim, and R. S. Fearing, "Robotic vertical jumping agility via series-elastic power modulation," *Sci. Robot.*, vol. 1, no. 1, Dec. 2016, Art. no. eaag2048.

[7] M. Plecnik, K. Fearing, and R. S. Fearing, "Adjustable power modulation for a leg mechanism suitable for running," in *Proc. Int. Conf. Robot. Autom.*, May 2019, pp. 9137–9142.

[8] R. S. Berkof and G. G. Lowen, "A new method for completely force balancing simple linkages," *J. Eng. Ind.*, vol. 91, no. 1, pp. 21–26, Feb. 1969.

[9] F. R. Tepper and G. G. Lowen, "General theorems concerning full force balancing of planar linkages by internal mass redistribution," *J. Eng. Ind.*, vol. 94, no. 3, pp. 789–796, Aug. 1972.

[10] G. G. Lowen, F. R. Tepper, and R. S. Berkof, "Balancing of linkages—An update," *Mechanism Mach. Theory*, vol. 18, no. 3, pp. 213–220, Jan. 1983.

[11] M. Skreiner, "Dynamic analysis used to complete the design of a mechanism," *J. Mechanisms*, vol. 5, no. 1, pp. 105–119, Mar. 1970.

[12] F. L. Conte, G. R. George, R. W. Mayne, and J. P. Sadler, "Optimum mechanism design combining kinematic and dynamic-force considerations," *J. Eng. Ind.*, vol. 97, no. 2, pp. 662–670, May 1975.

[13] Y.-A. Yao and H.-S. Yan, "A new method for torque balancing of planar linkages using non-circular gears," *J. Mech. Eng. Sci.*, vol. 217, pp. 495–503, May 2003.

[14] H.-S. Yan and G.-J. Yan, "Integrated control and mechanism design for the variable input-speed servo four-bar linkages," *Mechatronics*, vol. 19, no. 2, pp. 274–285, Mar. 2009.

[15] B. Moore, J. Schicho, and C. M. Gosselin, "Determination of the complete set of shaking force and shaking moment balanced planar four-bar linkages," *Mechanism Mach. Theory*, vol. 44, no. 7, pp. 1338–1347, Jul. 2009.

[16] A. A. Sherwood, "The dynamic synthesis of a mechanism with time-dependent output," *J. Mechanisms*, vol. 3, no. 1, pp. 35–40, Mar. 1968.

[17] A. Liniecki, "Synthesis of a slider-crank mechanism with consideration of dynamic effects," *J. Mechanisms*, vol. 5, no. 3, pp. 337–349, Sep. 1970.

[18] J. M. Halter and W. L. Carson, "Mechanism-force system synthesis to obtain a desired motion-time response," in *Proc. 4th World Congr. Theory Mach. Mechanisms*, Newcastle Upon Tyne, England, 1975, vol. 2, pp. 399–404.

[19] L. Zhen, "The analytical synthesis of spring elements in spatial mechanisms," *Mechanism Mach. Theory*, vol. 23, no. 1, pp. 47–54, Jan. 1988.

[20] P. J. Starr, "Dynamic synthesis of constraint paths," *J. Eng. Ind.*, vol. 95, no. 2, pp. 624–628, May 1973.

[21] S. Manoochehri and A. A. Seireg, "A computer-based methodology for the form synthesis and optimal design of robot manipulators," *J. Mech. Des.*, vol. 112, no. 4, pp. 501–508, Dec. 1990.

[22] D.-Z. Chen and L.-W. Tsai, "Kinematic and dynamic synthesis of geared robotic mechanisms," *J. Mech. Des.*, vol. 115, no. 2, pp. 241–246, Jun. 1993.

[23] G. K. Matthew and D. Tesar, "Cam system design: The dynamic synthesis and analysis of the one degree of freedom model," *Mechanism Mach. Theory*, vol. 11, no. 4, pp. 247–257, Jan. 1976.

[24] Z. Li and S. Kota, "Dynamic analysis of compliant mechanisms," in *Proc. Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf.*, vol. 5, Montreal, QC, Canada, Jan. 2002, pp. 43–50.

[25] A. Martini, M. Troncossi, M. Carricato, and A. Rivola, "Elastodynamic behavior of balanced closed-loop mechanisms: Numerical analysis of a four-bar linkage," *Meccanica*, vol. 49, no. 3, pp. 601–614, Mar. 2014.

[26] I. Imam and G. N. Sandor, "High-speed mechanism design—A general analytical approach," *J. Eng. Ind.*, vol. 97, no. 2, pp. 609–628, May 1975.

[27] S. S. Baek, K. Y. Ma, and R. S. Fearing, "Efficient resonant drive of flapping-wing robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Oct. 2009, pp. 2854–2860.

[28] J. P. Khatait, S. Mukherjee, and B. Seth, "Compliant design for flapping mechanism: A minimum torque approach," *Mechanism Mach. Theory*, vol. 41, no. 1, pp. 3–16, Jan. 2006.

[29] T. Tantanawat and S. Kota, "Design of Compliant mechanisms for minimizing input power in dynamic applications," *J. Mech. Des.*, vol. 129, no. 10, pp. 1064–1075, Oct. 2007.

[30] J. E. Clark *et al.*, "Biomimetic design and fabrication of a hexapedal running robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 4, pp. 3643–3649, May 2001.

[31] A. M. Hoover, S. Burden, X.-Y. Fu, S. S. Sastry, and R. S. Fearing, "Bio-inspired design and dynamic maneuverability of a minimally actuated six-legged robot," in *Proc. 3rd IEEE RAS EMBS Int. Conf. Biomed. Robot. Biomechatronics*, Sep. 2010, pp. 869–876.

[32] D. Zarrouk, D. W. Haldane, and R. S. Fearing, "Dynamic legged locomotion for palm-size robots," in *Proc. Micro- Nanotechnol. Sensors, Syst., Appl.*, May 2015, vol. 9467, Art. no. 94671S.

[33] M. Buehler, "Dynamic locomotion with one, four and six-legged robots," *J. Robot. Soc. Jpn.*, vol. 20, no. 3, pp. 237–242, Apr. 2002.

[34] S. Collins, A. Ruina, R. Tedrake, and M. Wisse, "Efficient bipedal robots based on passive-dynamic walkers," *Science*, vol. 307, no. 5712, pp. 1082–1085, Feb. 2005.

[35] S. Curran, B. T. Knox, J. P. Schmiedeler, and D. E. Orin, "Design of series-elastic actuators for dynamic robots with articulated legs," *J. Mechanisms Robot.*, vol. 1, no. 1, pp. 011 006-1–011 006-9, Jul. 2008.

[36] C. Semini, N. G. Tsagarakis, E. Guglielmino, M. Focchi, F. Cannella, and D. G. Caldwell, "Design of HyQ—A hydraulically and electrically actuated quadruped robot," *J. Syst. Control Eng.*, vol. 225, pp. 831–849, Sep. 2011.

[37] G. Kenneally and D. E. Koditschek, "Leg design for energy management in an electromechanical robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2015, pp. 5712–5718.

[38] D. J. Blackman *et al.*, "Leg design for running and jumping dynamics," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, Dec. 2017, pp. 2617–2623.

[39] J. M. Brown, J. L. Pusey, and J. E. Clark, "Design methodology of linkage morphology for high speed locomotion," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, Dec. 2017, pp. 730–736.

[40] S. Kalouche, "GOAT: A legged robot with 3D agility and virtual compliance," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2017, pp. 4110–4117.

[41] 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.

[42] B. Roth and F. Freudenstein, "Synthesis of path-generating mechanisms by numerical methods," *J. Eng. Ind.*, vol. 85, no. 3, pp. 298–304, Aug. 1963.

[43] A. P. Morgan, *Solving Polynomial Systems Using Continuation for Engineering and Scientific Problems*. Philadelphia, PA, USA: SIAM, 1987.

[44] A. J. Sommese and C. W. Wampler, *The Numerical Solution of Systems of Polynomials Arising in Engineering and Science*. Singapore: World Scientific, 2005.

[45] D. J. Bates, J. D. Hauenstein, A. J. Sommese, and C. W. Wampler, *Numerically Solving Polynomial Systems With Bertini*. Philadelphia, PA, USA: SIAM, Nov. 2013.

[46] A. Morgan and A. Sommese, "A homotopy for solving general polynomial systems that respects m-homogeneous structures," *Appl. Math. Comput.*, vol. 24, no. 2, pp. 101–113, Nov. 1987.

[47] B. Huber and B. Sturmfels, "A polyhedral method for solving sparse polynomial systems," *Math. Comput.*, vol. 64, no. 212, pp. 1541–1555, 1995.

[48] D. J. Bates, J. D. Hauenstein, A. J. Sommese, and C. W. Wampler, "Bertini: Software for numerical algebraic geometry." [Online]. Available: https://bertini.nd.edu/policy.html

[49] A. Baskar and S. Bandyopadhyay, "An algorithm to compute the finite roots of large systems of polynomial equations arising in kinematic synthesis," *Mechanism Mach. Theory*, vol. 133, pp. 493–513, Mar. 2019.

[50] T. Duff, C. Hill, A. Jensen, K. Lee, A. Leykin, and J. Sommars, "Solving polynomial systems via homotopy continuation and monodromy," *IMA J. Numer. Anal.*, vol. 39, pp. 1421–1446, 2019.
[51] M. M. Plecnik and J. M. McCarthy, "Controlling the movement of a TRR spatial chain with coupled six-bar function generators for biomimetic motion," *J. Mechanisms Robot.*, vol. 8, no. 5, pp. 051 005-1–051 005–10, May 2016.
[52] E. A. Dijksman, *Motion Geometry of Mechanisms*. Cambridge, U.K.: CUP Archive, Sep. 1976.
[53] C. W. Wampler, "Isotropic coordinates, circularity, and B'ezout numbers: Planar kinematics from a new perspective," in *Proc. ASME DETC/CIE Conf.*, Irvine, CA,USA, 1996.
[54] M. M. Plecnik and R. S. Fearing, "A study on finding finite roots for kinematic synthesis," in *Proc. ASME Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf.*, Aug. 2017, Paper V05BT08A083.
[55] E. A. Dijksman, "Six-Bar cognates of a Stephenson mechanism," *J. Mechanisms*, vol. 6, pp. 31–57, 1971.
[56] S. S. Balli and S. Chand, "Defects in link mechanisms and solution rectification," *Mechanism Mach. Theory*, vol. 37, no. 9, pp. 851–876, Sep. 2002.
[57] U. Kumaraswamy, M. S. Shunmugam, and S. Sujatha, "A unified framework for tolerance analysis of planar and spatial mechanisms using screw theory," *Mechanism Mach. Theory*, vol. 69, pp. 168–184, Nov. 2013.
[58] K.-L. Ting, J. Zhu, and D. Watkins, "The effects of joint clearance on position and orientation deviation of linkages and manipulators," *Mechanism Mach. Theory*, vol. 35, no. 3, pp. 391–401, Mar. 2000.
[59] C. Innocenti, "Kinematic Clearance sensitivity analysis of spatial structures with revolute joints," *J. Mech. Des.*, vol. 124, no. 1, pp. 52–57, Mar. 2002.
[60] J. Verschelde, "Polynomial homotopy continuation with PHCpack," *ACM Commun. Comput. Algebra*, vol. 44, no. 3/4, pp. 217–220, Jan. 2011.
[61] A. P. Morgan, A. J. Sommese, and C. W. Wampler, "A power series method for computing singular solutions to nonlinear analytic systems," *Numerische Math.*, vol. 63, no. 1, pp. 391–409, Dec. 1992.

Mark M. Plecnik (Member, IEEE) received the B.S. degree in mechanical engineering from the University of Dayton, Dayton, OH, USA, in 2010, the M.S. and Ph.D. degrees in mechanical engineering from the University of California, Irvine, IR, USA, in 2013 and 2015, respectively.

He is currently an Assistant Professor with the Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN, USA, which he joined in January 2019. He has applied his research interests toward consulting work for industry. His research interests include mechanisms, robotics, computational design, numerical algebraic geometry, machine dynamics, and the design of robot locomotion.

Dr. Plecnik received the A. T. Yang Memorial Award, in 2015, given out to the best paper in theoretical kinematics at the American Society of Mechanical Engineers International Design Engineering Technical Conferences.

Ronald S. Fearing (Member, IEEE) received the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, USA, in 1988, and the S.B. and S.M. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1983.

He is currently a Professor with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA, USA, which he joined in January 1988. His research interests include millirobotics, including flying and crawling millirobots, multirobot dexterous manipulation, parallel nanograsping (gecko adhesion), and rapid prototyping. He has also worked in tactile sensing, teletaction, and multifinger dexterous manipulation. He holds 17 U.S. and international patents.

Dr. Fearing received the Presidential Young Investigator Award in 1991.