A numerical continuation approach using monodromy to solve the forward kinematics of cable-driven parallel robots with sagging cables

Aravind Baskar[a], Mark Plecnik[a], Jonathan D. Hauenstein[a], Charles W. Wampler[a]

[a] *University of Notre Dame, Notre Dame, 46556, Indiana, USA*

Abstract

Designing and analyzing large cable-driven parallel robots (CDPRs) for precision tasks can be challenging, as the position kinematics are governed by kineto-statics and cable sag equations. Our aim is to find all equilibria for a given set of unstrained cable lengths using numerical continuation techniques. The Irvine sagging cable model contains both non-algebraic and multi-valued functions. The former removes the guarantee of finiteness on the number of isolated solutions, making homotopy start system construction less clear. The latter introduces branch cuts, which could lead to failures during path tracking. We reformulate the Irvine model to eliminate multi-valued functions and propose a heuristic numerical continuation method based on monodromy, removing the reliance on a start system. We demonstrate this method on an eight-cable spatial CDPR, resulting in a well-constrained non-algebraic system with 31 equations. The method is applied to four examples from literature that were previously solved in bounded regions. Our method computes the previously reported solutions along with new solutions outside those bounds much faster, showing that this numerical method enhances existing approaches for comprehensively analyzing CDPR kineto-statics.

Keywords: Cable-driven parallel robot, Numerical continuation, Irvine cable model, Monodromy

1. Introduction

Cable-driven parallel robots (CDPRs) are a class of parallel manipulators in which an end-effector platform is controlled by multiple cables, whose lengths are adjusted to control its movements. These robots have proven effective for various applications, including visual sensing and manipulation in vast environments such as agricultural fields, sports stadiums, and theaters. Recently, there has been a resurgence of interest in designing CDPRs for precision tasks, such as 3D printing of large models [1, 2], construction robotics [3], and search and rescue missions [4]. Designing CDPRs for such tasks pose an ongoing research challenge due to the inherent complexity of their kinematics compared to traditional parallel robots with rigid legs. Cable-driven systems often necessitate redundant cables to fully constrain the end-effector platform, increase the workspace, and achieve improved distribution of tensions in the cables [5, 6]. Such redundant cables introduce further complexity into the analysis and design of CDPRs.

Unlike rigid links, cables are elastic and subject to sagging, making kinematic analysis inherently coupled with static equilibrium analysis, falling under the category of kineto-statics problems. For modeling the cables, the Irvine cable model [7, 8], which is derived based on Hooke's law accounting for elasticity and sagging under self-weight, is more realistic than simplified cable models that assume inelasticity and zero mass [9, 10, 11]. Recent experimental research indicates that, while such simplified models are practical in many cable-driven systems, there are outstanding challenges in using them for the design and analysis of large CDPRs where cable sag is pronounced [4, 12]. Hence, recent works have examined the kineto-statics in cable-driven robots, employing the more comprehensive Irvine cable model, as reviewed in [13]. Whereas the algebraic form of the simplified models allows the use of powerful techniques specialized to algebraic systems [14], these methods do not apply to the Irvine cable model due to its inclusion of non-algebraic functions such as the inverse hyperbolic sine and square-root function. This presents a challenge which we aim to address.

Our goal is to solve the forward (a.k.a. direct) kineto-static equilibrium problem for CDPRs with sagging cables to find all solutions. That is, given a specific CDPR architecture with prescribed unstrained cable lengths, we seek to identify all potential static equilibrium configurations, which correspond to the roots

of the kineto-static equations. This work does not delve into the stability considerations of these equilibrium configurations, e.g., see [15]. Instead, our current scope is focused on identifying all the equilibrium configurations. Identifying all possible configurations is much more challenging than finding a single equilibrium configuration using, for instance, a local Newton's method, which is already non-trivial for large systems. Nevertheless, solving the forward kineto-statics globally is crucial for characterizing the comprehensive workspace of these systems, which comprise numerous regions called *input modes* [16]. Input modes are the separate regions that result after partitioning the configuration space by the singular solutions of the forward kineto-static problem, which form their boundary sets. These boundary sets, also referred to as input singularities [16, 17] or Type II singularities [18] or direct kinematic singularities [19, 15], must generally be avoided since they present transmission trouble. Therefore, it is useful to partition the workspace via input modes for motion planning.

One might consider the alternative of analyzing the workspace of a CDPR via inverse kinematics. For example, this is highly effective for rigid-link 6SPS (Gough-Stewart) parallel robots where each end-effector pose is reached by a unique set of leg lengths. However, for CDPRs having redundant cables, an end-effector pose is generally reached by a positive dimensional family of unstrained cable lengths. Consequently, the forward kineto-statics problem is the preferred approach for CDPRs.

To fully analyze the workspace of a CDPR, one seeks to find all equilibrium roots over a set of input samples, considering that the input modes may change over the input space. Once a sufficient sample set has been created, local methods may be used to continue the solutions to nearby points in the respective input modes to generate a large sample set [20]. One may then build a full workspace model by training a neural network model [21, 22] or a graph-based model [16] on such sets.

While one expects the roots of the forward problem to be isolated, i.e., zero-dimensional, it cannot be assumed *a priori* that the total number of roots is finite. This is due to the non-algebraic nature of the Irvine cable model which, unlike algebraic systems, presents no theoretical bounds on the number of roots counted over the complex field. However, certain non-algebraic systems are known to yield a finite number of isolated roots over the real field when they contain *Pfaff functions* [23] (also called Pfaffian functions). Pfaff functions are a class of functions that can be written as polynomial functions of their own derivatives and other Pfaff functions. Polynomials, exponentials, and trigonometric functions are Pfaff functions that give rise to Pfaff manifolds over some open real domain. The inverse of a Pfaff function is also a Pfaff function if it does not vanish anywhere within the domain. It happens that even though the Irvine cable model is non-algebraic, it leads to a system of CDPR kineto-static equations that satisfy the Pfaffian conditions, hence number of real roots is finite.

In the past, interval analysis methods which exploit appropriately chosen bounds on end-effector configuration variables and cable tension have been employed to solve the global forward kinematic problem in cable-driven robots [24, 11, 25]. These methods exploit Kantorovich's theorem [26] to guarantee and certify that all roots within the chosen bounds are found. These methods work by subdividing the initial region into boxes, then contracting or subdividing each box until each descendant can be verified to either contain a single root or no root at all. However, for search spaces of large dimension, it can be computationally expensive to fully complete this subdivision process. While interval analysis methods have demonstrated effectiveness in solving CDPR forward problems, they face challenges as the number of cables controlling the end-effector increases. For instance, in a spatial system with eight cables, explored in [25], the search space encompassed 36 dimensions, and a single case required approximately 24 hours to solve. Further questions persist regarding the selection of the appropriate bounds for tension forces, which are known to exhibit high sensitivity to small variations in cable lengths, especially when nearing taut configurations.

Alternatively, numerical algebraic geometry techniques have been employed by researchers to solve forward kinematics for rigid-legged parallel manipulators, involving the resolution of multivariate polynomial systems [27, 28]. In [29], kineto-statics of a compliant parallel manipulator with spring elements was addressed by solving a large polynomial system in a similar manner. Typically, numerical polynomial continuation techniques begin by defining a *homotopy*, a continuous transformation from a known starting polynomial system to a target polynomial system where the endpoints need to be determined. This defines paths one must track numerically. Algorithms based on this idea find the complete solution list for moderately large polynomial systems. See, for example, [27, 30] for a general overview.

Numerical continuation techniques are often compared to other less sophisticated analytical methods, such as the multi-start Newton's method, which may identify numerous distinct root instances. By analytic, we

refer to *complex-analytic* functions which are complex differentiable in a neighbourhood of every point in the entire complex domain [31, p. 69]. Numerous studies involving large systems of nonlinear equations [32, 33] demonstrate that numerical continuation methods more efficiently exploit complex-analytic functions in comparison to multi-start Newton techniques. This efficiency is attributed to the guarantee that distinct starting points will lead to distinct endpoints when path crossings are absent. This approach is facilitated by including random complex parameters in the formulation of the homotopy so that path crossings are avoided with probability one.

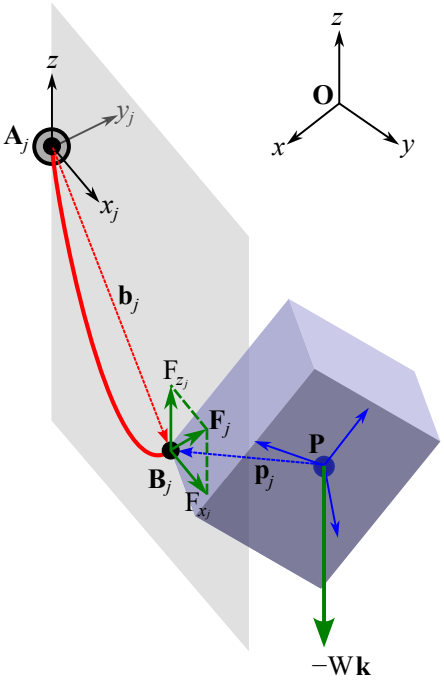
In [34], a real parameter continuation technique was employed to solve the global kineto-static problem of CDPRs. In this approach, an idealized cable system with infinite Young's modulus and zero cable mass was deformed into an elastic sagging cable system. The idealized system can be modeled as a kinematic problem instead of a kineto-static one and, hence, solved as a set of polynomial sub-systems to obtain starting points. Here, each sub-problem corresponds to a different subset of cables in tension. Continuation from the idealized system into the regime of sagging cables with finite Young's modulus and non-zero mass relies on the implicit function theorem, which applies in the neighborhood of the ideal case. However, since there is no guarantee on the extent of this nonsingular neighborhood, path failures may occur when the parameters are not sufficiently near the ideal case, as illustrated in [34][§ VII]. This trouble becomes more significant whenever the cable material of the target system has a low Young's modulus and/or high density. These failures primarily arise from encountering special points in the parameter space called *branch points* where two or more real paths merge [27, pp. 28]. At these points, the associated paths are not smooth, so numerical tracking fails due to ill-conditioning.

The limitations of real parameter continuation are applicable more broadly to all real parameterized systems. For a detailed illustration in a simple univariate root-finding problem, see [27, pp. 25-29]. This trouble can be avoided by conducting parameter continuation in complex space which can be achieved by following a random arc between the start and end parameter values in the complex plane instead of a straight line through real values. When constructed via random complex numbers, e.g., see "gamma trick" [27, pp. 102-104], such an arc avoids branch points with probability one. It is important to note that numerical continuation in the complex parameter space does not fully preclude numerical failures, as a path may yet fail if it passes close enough to a branch point, as ill-conditioning extends into a small neighborhood of a singularity. Nevertheless, robust predictor-corrector path tracking techniques, especially ones that adaptively increase precision when necessary [35, 36] effectively mitigate such failures.

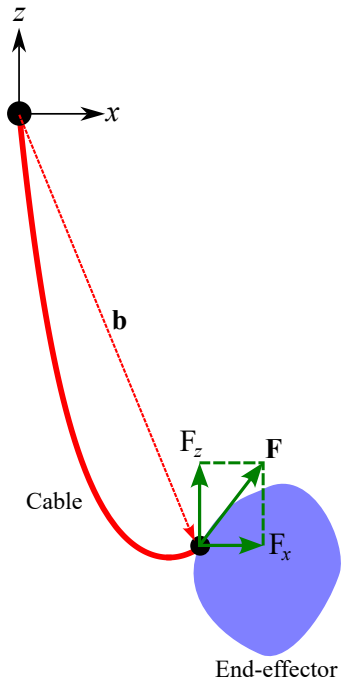
One can avoid branch points by moving into the complex domain only if the system under consideration is analytic. If the system contains multi-valued functions, such as the inverse hyperbolic sine and square-root functions appearing in the Irvine cable model, analyticity breaks down in any domain that contains *branch cut* discontinuities. In any mathematical software implementation, such cuts are necessary to yield a *principal* value over the complex domain [31, pp. 69-71]. Without specific measures for switching branches, predictor-corrector path tracking will fail or perform erratically whenever the path happens to cross a branch cut. See Section 2.1 for an illustration of such a failure. Our reformulation of the Irvine cable model avoids the problems of branch cuts by not using the multi-valued functions $\sinh^{-1}$ and $\sqrt{\cdot}$.

Parameter continuation over the complex space presents a distinct challenge. Unlike in real parameter continuation, where successful path tracking ensures that real starting points conclude in a real endpoint, complex parameter continuation does not offer such automatic guarantees. In this context, a real starting point might not necessarily lead to a real endpoint, while a complex starting point may still yield one. This behavior complicates the construction of a start system for non-algebraic systems, including Pfaff systems, which may lack a finite upper bound on the total number of complex roots but admit a finite number of real solutions. It is not only non-algebraic systems that pose challenges as constructing a computationally affordable start system for large algebraic systems with sharp theoretical upper bounds remains an ongoing research challenge. Recent attention has turned to heuristic continuation methods to address this issue [37, 38, 39, 40, 41]. These newer methods take a different approach to the challenge by eliminating the need for constructing a start system. Instead, they are initiated with a lone seed solution often obtained through Newton's method. Further discussion and references on this topic can be found in Section 3.1. Indeed, this is the approach we adopt to solve the global kineto-statics problem of CDPRs.

In the following, we present a new continuation methodology that addresses the aforementioned research challenges. First, we reformulate the Irvine cable model to remove multi-valued functions through a change of variables. This enables the use of complex space to define smooth paths that avoid branch points with



(a) A generic CDPR with n sagging cables, one of which is depicted in its vertical plane.



(b) Irvine sagging cable model in plane.

Figure 1: Schematic of a cable-driven parallel robot

probability one. Secondly, for solving non-algebraic systems, especially Pfaff equations, we introduce a novel heuristic root accumulation strategy using *random monodromy loops*, eliminating the necessity for a start system. Using this algorithm, we demonstrate that the forward kinematics of large CDPRs with sagging cables can be solved in a faster time frame than existing approaches. We present numerical examples and benchmark the results against an existing data set from the literature for an eight-cable spatial CDPR system [42, 20]. This algorithm exhibits distinct trade-offs compared to existing interval analysis and continuation methods. While this heuristic algorithm cannot guarantee the discovery of all possible equilibrium configurations, numerical experiments across four examples consistently identified all the previously reported configurations found by interval analysis methods. Since our algorithm does not impose pre-specified search bounds, it also found additional solutions beyond the bounds specified in the prior literature. Whether these additional solutions are significant is a matter to be evaluated when building a full workspace model, which is beyond our present scope. Nonetheless, this method holds significant potential for generating comprehensive workspace data sets for large CDPRs and for solving the root-finding of other high-dimensional non-algebraic systems encountered in various applications.

2. Mathematical model of kineto-statics of a generic CDPR with n cables

Consider the schematic of a generic spatial CDPR with n cables as shown in Fig. 1a. Gravity acts downwards along the z-direction, $\hat{\mathbf{k}} = (0, 0, 1)^\top$. The end-effector is operated by n-cables, $j = 1, 2, \ldots, n$. Let $\mathbf{P} = (P_x, P_y, P_z)^\top$ be the position of the end-effector platform's center of mass in the global coordinate frame. The weight of the platform is $W = Mg$, where M is the mass of the end-effector platform and g the acceleration due to gravity. We assume no other external load in this work. The local frame of the end-effector is affixed to $\mathbf{P}$. Position vector $\mathbf{p}_j$ represents cable j connection point on the end-effector in the local frame of the end-effector. Let the orientation of the end-effector in the global coordinate frame be given by a 3×3 rotation matrix $[\mathbf{Q}]$. We use quaternion variables $\{q_0, q_1, q_2, q_3\}$ to represent the 3×3

rotation matrix $[\mathbf{Q}]$ as follows:

$$[\mathbf{Q}] = \begin{bmatrix} {q_0}^2 + {q_1}^2 - {q_2}^2 - {q_3}^2 & 2\left(q_1 q_2 + q_0 q_3\right) & 2\left(-q_0 q_2 + q_1 q_3\right) \\ 2\left(q_1 q_2 - q_0 q_3\right) & {q_0}^2 - {q_1}^2 + {q_2}^2 - {q_3}^2 & 2\left(q_0 q_1 + q_2 q_3\right) \\ 2\left(q_0 q_2 + q_1 q_3\right) & 2\left(-q_0 q_1 + q_2 q_3\right) & {q_0}^2 - {q_1}^2 - {q_2}^2 + {q_3}^2 \end{bmatrix},$$

along with a normalization constraint:

$${q_0}^2 + {q_1}^2 + {q_2}^2 + {q_3}^2 - 1 = 0. \tag{1}$$

It is worth noting that other representations for the rotation matrix, such as using Euler angle variables, Cayley-Klein variables, etc., are equally admissible here. We use quaternions for the standard advantage that gimbal lock does not appear in the quaternion representation of 3D rotations. Regardless of the choice, for any given orientation, all representations admit redundant solutions that exhibit a *group action*, e.g., [43]. This group action can be exploited to save computational costs during path tracking computations, irrespective of the chosen representation for the rotation matrix. We will elaborate on this in Section 2.3.

2.1. Position loop-closure equations

The j^{th} cable's fixed point is located at $\mathbf{A}_j$. Each cable can be assumed to lie in a vertical plane $X_j Z$ under static equilibrium conditions. Let the cable plane be at a z-rotation of ϕ_j which is represented as $[\mathbf{Z}(\phi_j)]$ with respect to the global coordinate plane XZ. Note that a z-rotation of $\phi_j + \pi$ also represents the same cable vertical plane with opposing x axes. The span of the j^{th} cable from its fixed point to the moving platform is represented by a local vector within the cable plane named $\mathbf{b}_j$. The position loop-closure equation associated with cable $j = 1, 2, \ldots, n$ is:

$$\mathbf{P} + [\mathbf{Q}]\ \mathbf{p}_j - [\mathbf{Z}(\phi_j)]\ \mathbf{b}_j - \mathbf{A}_j = \mathbf{0}, \tag{2}$$

where $\mathbf{b}_j$ is of the form $(b_{xj}, 0, b_{zj})^\top$ and given by the Irvine cable model in the local vertical cable plane in terms of the corresponding tension force $\mathbf{F}_j$ of the form $(F_{xj}, 0, F_{zj})^\top$. In the following, we revisit this model in detail and derive an equivalent representation made of complex-analytic functions which are amenable for numerical continuation methods.

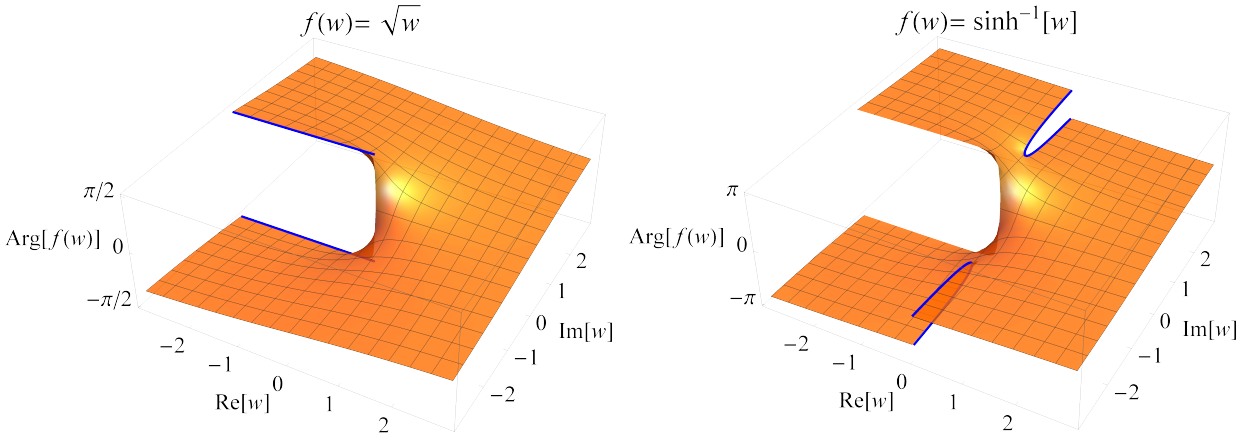
Irvine cable model

Consider the schematic of a cable in the vertical plane XZ as shown in Fig. 1b. Let the coiling system of the cable be assumed at the origin of the cable vertical plane. Acceleration due to gravity g is downwards along the z-direction. We neglect all lateral static forces on the cable, so the cable lies in a vertical plane. The cable stretches under tension and sags under self-weight. Let μ, A and E be the properties of the cables, namely, linear density, cross sectional area, and Young's modulus, respectively. According to the Irvine model [7] derived based on Hooke's law, the kineto-statics equations for a cable of unstrained length L are:

$$\mathbf{b} = \begin{pmatrix} b_x \\ 0 \\ b_z \end{pmatrix} = \begin{pmatrix} F_x \left(\frac{L}{EA} + \frac{1}{\mu g} \left(\sinh^{-1}\left[\frac{F_z}{F_x}\right] - \sinh^{-1}\left[\frac{F_z - \mu g L}{F_x}\right] \right) \right) \\ 0 \\ \frac{F_z L}{EA} - \frac{\mu g L^2}{2EA} + \frac{1}{\mu g} \left(\sqrt{{F_x}^2 + {F_z}^2} - \sqrt{{F_x}^2 + (F_z - \mu g L)^2} \right) \end{pmatrix}, \tag{3}$$

where $\mathbf{F} = (F_x, 0, F_z)^\top$ is the tension in the cable at its end, resolved into horizontal component F_x and vertical component F_z in the cable vertical plane. The y-coordinates of tension and position are both zero by assumption. Here, $F_x > 0$ is a necessary operating condition of the Irvine model for the cable to be in tension. This condition only accounts the configurations in the half-plane $b_x > 0$. Configurations in the half-plane $b_x < 0$ can be ignored, because they are already accounted for in a z-rotation of the cable vertical plane by π as noted earlier.

Equation (3) contains branch cuts because of the presence of the inverse hyperbolic sine and square-root functions which are multi-valued over the complex domain. Branch cuts are defined by convention



(a) Square-root function $f(w) = \sqrt{w}$

(b) Inverse hyperbolic sine function $f(w) = \sinh^{-1}[w]$

Figure 2: Principal branch of multi-valued functions of a complex variable w defined using branch cuts

for a principal definition. See Fig. 2 for one possible such definition for these functions involving branch cuts (shown in blue). In the case of the square-root function, a branch cut is conventionally defined in most programming libraries along the negative real axis $(-\infty, 0)$, while it is defined along the imaginary axis $(-i\infty, -i)$ and $(i, i\infty)$ for the inverse hyperbolic sine function. (Note that the apparent discontinuity along the negative real axis in Fig. 2b is not a branch cut of $\sinh^{-1}[w]$: it arises from a branch cut in the Arg[ ] function used for plotting this visualization.) If these standard definitions are used, the branch cuts cause failures during numerical continuation whenever a path crosses one of these discontinuities. Unless special branch switching logic is implemented, any robust predictor-corrector path-tracking algorithm slows down indefinitely upon the approach of such cuts and eventually grinds to a halt upon reaching the maximum number of steps. These failures do not manifest in the form of ill-conditioning.

Illustration: Consider a simple homotopy equation, $H(w[t], t) := w[t] - \sqrt{e^{2\pi i t}} = 0$, from $t = 0$ to $t = 1$ with a start point $w[0] = 1$, where $\sqrt{\cdot}$ is the principal square-root function. Solving this equation using any certified predictor-corrector tracking algorithm fails as $t \to 0.5^-$ where the path encounters a branch cut of the principal square-root function along the negative real axis at -1. Note that $\lim_{t\to 0.5^-} w'[t] = -\pi \neq 0$. Thus it is not a *path crossing* failure which are typically caused by branch points where at least two solution branches of w merge. One way to circumvent this implementation problem is by introducing additional variables to keep track of which sheet of the multi-valued function the current point of the path lies on. A simpler approach is to use squaring to write an alternative homotopy that is free of branch cuts: $H(w[t], t) := w[t]^2 - e^{2\pi i t} = 0$. While this resolves the issue and allows us to track until $t = 1$, it introduces a second branch with a corresponding start point $w[0] = -1$. This is an unavoidable cost of resolving analyticity.

Change of variables

The illustration above shows how trouble stemming from branch cuts can be eliminated by rewriting a system to remove multi-valued functions. Similarly, to apply numerical continuation to the forward kineto-statics problem of CDPRs, we need to reformulate the Irvine cable model get rid of the square-root and inverse hyperbolic sine functions. To do so, we introduce a change of variables from F_x, F_z to new real variables α, β:

$$\frac{F_z}{F_x} = \sinh[\alpha], \quad \frac{F_z - \mu g L}{F_x} = \sinh[\beta]. \tag{4}$$

Assuming $\mu > 0$ and $L > 0$, F_x and F_z can be expressed uniquely in terms of the new variables α and β:

$$F_x = \frac{\mu g L}{\sinh[\alpha] - \sinh[\beta]}, \quad F_z = \frac{\mu g L \sinh[\alpha]}{\sinh[\alpha] - \sinh[\beta]}. \tag{5}$$

The next-step is to re-write the square-root functions in the Irvine model given by Eq. (3) in terms of the

new variables. First consider the following expression:

$$F_x{}^2 + F_z{}^2 = \frac{\mu^2 g^2 L^2 \cosh^2[\alpha]}{(\sinh[\alpha] - \sinh[\beta])^2}.$$

The numerator is always a square of a positive number ($\because \cosh[a] \geq 1,\ \forall\ a \in \mathbb{R}$), but the same is not generically true for the denominator. However, Irvine cable model requires $F_x > 0$ which translates to $\sinh[\alpha] > \sinh[\beta]$ based on the definition in Eq. (5). Since the hyperbolic sine function is strictly increasing over $\mathbb{R}$, this condition reduces to $\alpha > \beta$. With this assumption, it follows that the real positive square-root:

$$\sqrt{F_x{}^2 + F_z{}^2} = \frac{\mu g L \cosh[\alpha]}{\sinh[\alpha] - \sinh[\beta]}. \tag{6}$$

Similarly, it can be easily shown that the real positive square-root:

$$\sqrt{F_x{}^2 + (F_z - \mu g L)^2} = \frac{\mu g L \cosh[\beta]}{\sinh[\alpha] - \sinh[\beta]}. \tag{7}$$

Substituting Eqs. (4-7) into Eq. (3):

$$\mathbf{b} = \begin{pmatrix} \frac{\mu g L}{\sinh[\alpha]-\sinh[\beta]} \left(\frac{L}{EA} + \frac{1}{\mu g} \left(\sinh^{-1}[\sinh[\alpha]] - \sinh^{-1}[\sinh[\beta]] \right) \right) \\ 0 \\ \frac{\sinh[\alpha]}{\sinh[\alpha]-\sinh[\beta]} \frac{\mu g L^2}{EA} - \frac{\mu g L^2}{2EA} + L \frac{\cosh[\alpha]-\cosh[\beta]}{\sinh[\alpha]-\sinh[\beta]} \end{pmatrix}.$$

Since hyperbolic sine function is bijective over $\mathbb{R}$,

$$\mathbf{b} = \begin{pmatrix} \frac{L}{\sinh[\alpha]-\sinh[\beta]} \left(\frac{\mu g L}{EA} + \alpha - \beta \right) \\ 0 \\ \frac{\sinh[\alpha]+\sinh[\beta]}{\sinh[\alpha]-\sinh[\beta]} \frac{\mu g L^2}{2EA} + L\ \frac{\cosh[\alpha]-\cosh[\beta]}{\sinh[\alpha]-\sinh[\beta]} \end{pmatrix}.$$

It can be observed that this modified kineto-static equation is a function of $\nu = \frac{\mu g}{EA}$, a combined material constant of the cable with units of reciprocal length. It defines the strain of a vertically hung cable under its self-weight normalized by its unstrained length.

Reformulated sagging cable model

In summary, through a change of variables from F_x, F_z to α, β under the assumption that $\mu > 0$ and $L > 0$, we present an equivalent sagging cable model:

$$\mathbf{b} = \begin{pmatrix} b_x \\ 0 \\ b_z \end{pmatrix} = \frac{L}{\sinh[\alpha] - \sinh[\beta]} \begin{pmatrix} \nu\ L + \alpha - \beta \\ 0 \\ (\sinh[\alpha] + \sinh[\beta])\ \nu\ \frac{L}{2} + \cosh[\alpha] - \cosh[\beta] \end{pmatrix}, \tag{8}$$

where $\nu = \frac{\mu g}{EA}$. The corresponding cable tension in the vertical plane is given in Eq. (5). Equations (5,8) define the Irvine model in terms of the new real variables α, β with a necessary operating condition that $\alpha > \beta$. In some systems with metallic cables, it may be a reasonable approximation to neglect elasticity by setting $\nu \to 0$ as $E \to \infty$. This system is complex-analytic disregarding the inequality condition $\alpha > \beta$. Note that hyperbolic sine and cosine functions are single-valued over the complex domain. The equations cannot be made algebraic as they contain both algebraic and exponential forms of α and β. However, as noted earlier, this system defines Pfaff manifolds.

Note from Eq. (4) that the magnitudes of α and β grow in a slow *logarithmic* manner with respect to the ratio $\left|\frac{F_z}{F_x}\right|$. In addition to removing multi-valued functions, this new formulation may offer an advantage when numerically solving CDPR systems subjected to large magnitude forces. In particular, when the payload is suspended between horizontally taut cables so that $|F_x| \gg |F_z|$, the new variables α and β remain small in magnitude. Although for brevity we omit a derivation, we note that the cable profile in the vertical plane can be derived in terms of the new variables, following [7], in parametric form $s \in [0, 1]$:

$$\frac{L}{\sinh[\alpha] - \sinh[\beta]} \begin{pmatrix} s\ \nu\ L + \sinh^{-1}[s\ \sinh[\alpha] + (1-s)\ \sinh[\beta]] - \beta \\ 0 \\ (s\ \sinh[\alpha] + (2-s)\ \sinh[\beta])\ s\ \nu\ \frac{L}{2} + \sqrt{1 + (s\ \sinh[\alpha] + (1-s)\ \sinh[\beta])^2} - \cosh[\beta] \end{pmatrix}, \tag{9}$$

where $s = 0$ corresponds to the origin of the cable and $s = 1$ corresponds to the end-effector connection point given by Eq. (8).

As an additional note, refer to Appendix A for a further modification that proves advantageous in the context of inverse kineto-statics and also provides a geometric understanding of the revised model.

2.2. Force and torque equilibrium equations

With the relation between tension forces and cable displacement in hand, our next task is to form equations for force and torque balance. Let $\mathbf{F}_j = (F_{x_j}, 0, F_{z_j})^\top$ represent the tension force in cable j at the end-effector point. F_{x_j} and F_{z_j} are written in terms of the modified variables α_j and β_j following Eq. (5). Expressed in the world coordinate system, the sum of all forces on the end-effector including each cable reaction force $-\mathbf{F}_j$ and the weight of the platform $W = Mg$ acting along the negative z-direction gives the condition for force equilibrium:

$$\sum_{j=1}^{n} [\mathbf{Z}(\phi_j)]\,(-\mathbf{F}_j) - W\hat{\mathbf{k}} = \mathbf{0}. \tag{10}$$

The torque equilibrium equations with respect to $\mathbf{P}$ are written as:

$$\sum_{j=1}^{n} ([\mathbf{Q}]\ \mathbf{p}_j) \times ([\mathbf{Z}(\phi_j)]\,(-\mathbf{F}_j)) = \mathbf{0}. \tag{11}$$

Assuming all cables have identical linear density, $\mu > 0$, the force and the torque equilibrium equations can be simplified slightly by dividing out the constant μg. Equations (1,2,10,11) amount to $1+3n+3+3 = 3n+7$ kineto-statics equations. For the forward kinematic problem, the variables are $\{\alpha_j, \beta_j, \phi_j\}$ for $j = 1, 2, \ldots, n$ along with the position of the center of mass $\mathbf{P}$ of the platform and the quaternion variables $\{q_0, q_1, q_2, q_3\}$, which also totals to $3n + 7$ in number. For generic architecture parameters, this leads to a well-constrained system of equations defining the forward kinematics.

2.3. Discussions

The following discussions on the mathematical model are pertinent:

1. The material constants of the kineto-statics system appear only as the ratios $\nu = \frac{\mu g}{EA}$ and $\lambda = \frac{W}{\mu g} = \frac{M}{\mu}$.
2. Over the complex field with imaginary unit i, one may see that if $\{\alpha_j, \beta_j, \phi_j\}$ satisfy Eqs. (2,10,11) so does $\{\alpha_j + i\pi N_1, \beta_j + i\pi N_1, \phi_j + \pi(N_1 + 2N_2)\}$, for any two integers N_1, N_2. These group actions derive from our reformulation to eliminate branch cuts in favor of *many-to-one* functions and the fact that every vertical plane has two z-rotational representations with respect to the global coordinate frame. Similar to the examples in [43, 38], these group actions also apply to the homotopy paths we consider, so we only track one member of each group. When recording roots, we consider subsequent occurrences of members of the same group as repeat instances, and in particular, for real roots, we record the principal representation with real-valued α_j, β_j and $\phi_j \in (-\pi, \pi]$.

3. The quaternion variables admit two representations for any given $[\mathbf{Q}]$, namely, $\{q_0, q_1, q_2, q_3\}$ and $\{-q_0, -q_1, -q_2, -q_3\}$. This group action can be exploited similarly to Item 2.
4. A question remains on how best to involve the necessary inequality constraint, $\alpha_j > \beta_j,\ \forall\ j$. Handling this inequality upfront is cumbersome for an n-cable system because there arise n such inequalities and they cannot be evaluated over the complex field. More importantly, numerical continuation paths starting from a real root which violates these inequalities can result in a *valid* real root and vice versa. Hence, we ignore them during computations and compute all possible roots of the system including those which violate these inequalities. During post-processing, an additional check identifies the valid roots.

Remark on the planar case. When one applies this model to planar n-cable CDPRs, the number of variables can be reduced. For a system lying in the XZ-plane, the rotation $[\mathbf{Q}]$ simplifies to just a y-rotation. The y-components of the position loop-closure equations as well as the force equilibrium equations identically vanish. The same is true for the x and z components of the torque equilibrium equations. Further, ϕ_j can only admit 0 or π for all $j = 1, 2, \ldots, n$ in the planar case. This results in 2^n sub-systems considering all combinations. Unexpectedly, these 2^n sub-systems are equivalent to each other because of the group action associated to N_1. For instance, a real solution with cable j in the form $\{\alpha_j, \beta_j, \pi\}$ also occurs as a complex one $\{\alpha_j + i\pi N_1, \beta_j + i\pi N_1, 0\}$ for odd N_1. Hence, it is sufficient to solve one of the 2^n sub-systems, for example, the one with $\phi_j = 0,\ \forall\ j$, in order to obtain all of the valid configurations.

3. Solving non-algebraic, complex-analytic systems via monodromy loops

In preparation for solving the global forward kinematic problem of CDPRs with sagging cables, we next propose a heuristic numerical continuation algorithm that exploits monodromy to solve non-algebraic, complex-analytic systems.

Numerical continuation is an effective technique to solve nonlinear system of equations which are complex-analytic in nature [44]. This technique works through predictor-corrector path tracking which continuously deforms the start point(s) of a known system into the end point(s) associated with a target system of interest. Specifically for finding all isolated roots of polynomial system of equations, theory on the construction of efficient start systems is well-developed exploiting the Bézout bounds [27, 28] as well as the sparse structure of these polynomial systems [45, 46]. For non-algebraic systems without any theoretical bounds on the number of roots, construction of efficient start systems for numerical continuation is an open research question. Start systems constructed through a simplified polynomial approximation of the original system is an option as done in [34] in the context of kinematics of CDPRs. However, this presents its own challenges, and in general, a reasonable polynomial approximation may not always be possible in other physical systems.

In many polynomial systems which arise in kinematic design and analysis, it is noted that the actual number of roots is only a small fraction of the number of start points of a well-constructed start system, e.g., [47]. Most of the paths in such a homotopy diverge, making such methods computationally expensive in disproportion to the actual root-count. An increasingly popular approach for countering this effect is to abandon the construction of exhaustive start systems and instead implement heuristics that involve random processes with the goal of obtaining most or all solutions probabilistically [37, 38, 39, 40, 41]. Among these, [38, 39, 40, 41] all use a monodromy approach, where *monodromy* describes how roots of a parameterized system of equations change as loops are traced in the parameter space around branch points.

In a polynomial system that has a finite root-count, a monodromy action permutes the set of all isolated roots as smooth paths of a homotopy are tracked around a closed loop in a parameter space. Given at least one root of the target system, one may seek to accumulate additional roots by tracking nontrivial monodromy loops. Although it is difficult to know ahead of time which loops in parameter space will land on new roots versus which ones will return to roots already known, a well-tested root accumulation strategy is to execute random monodromy loops [41, 48]. Under certain assumptions of independence, the accumulation of roots follows the Lincoln-Petersen mark-and-recapture model, where the computational expense is roughly proportional to the total number of finite roots.

For a non-algebraic system, the total root count in complex space cannot be assumed finite, so the process of accumulating roots might not terminate. However, if a non-algebraic system defines a Pfaff manifold, its real root count is finite. By modifying the way random monodromy loops are generated and recording only

real solutions, we can create an algorithm that has a chance of finding all the real roots, terminating when slow progress indicates that most real roots have likely been found. This is the approach we apply here to the CDPR forward kinematics problem.

3.1. Random monodromy loops

In this section, we present an algorithm using random monodromy loops and a heuristic root accumulation strategy to solve non-algebraic, complex-analytic systems. It is based on a common strategy for treating non-algebraic systems, wherein a homotopy is formed by adding an array of random constants to the given system. For example, see the so-called Newton homotopies in [32, 49].

Consider a nonlinear complex-analytic system of equations $\mathbf{f}(\mathbf{w}) = \mathbf{0}$ of n equations in n unknowns, $\mathbf{w}$, for which numerical approximations of many or all real roots $\mathbf{w}$ are sought. We construct a parameter homotopy with an *artificial* parameter array $\boldsymbol{\tau}$ of dimension n and path parameter $t \in \mathbb{R}$:

$$\mathbf{H}(\mathbf{w}[t], t) := \mathbf{f}(\mathbf{w}[t]) - \boldsymbol{\tau}(t), \tag{12}$$

where $\boldsymbol{\tau}(t)$ is defined such that $\boldsymbol{\tau}(0) = \mathbf{0}$. A random monodromy loop may be constructed then as:

$$\boldsymbol{\tau}(t) = \boldsymbol{\gamma}(1 - e^{2\pi i t}), \tag{13}$$

where $\boldsymbol{\gamma}$ is a random vector of non-zero complex numbers of dimension n. This circular monodromy loop is chosen for ease of implementation. (There exist other multi-nodal constructions such as a triangular segmented loop, petal shaped loop made of two different segments, etc., which may be equally preferable [50].) As the path parameter t varies along the real line, $\mathbf{w}$ traces a curve implicitly defined by $\mathbf{H}(\mathbf{w}[t], t) = \mathbf{0}$. Every integer value of t gives $\boldsymbol{\tau}(t) = \mathbf{0}$, thereby leading $\mathbf{w}$ to a root of $\mathbf{f}(\mathbf{w}) = \mathbf{0}$. All solutions of $\mathbf{f}(\mathbf{w}) = \mathbf{0}$, including all real solutions, lie in the fiber over $\boldsymbol{\tau} = \mathbf{0}$, but for a fixed $\boldsymbol{\gamma}$, one cannot be assured that they all lie on one connected component, referred to as *circuit* from hereon. We may increase the chances of connecting to all real solutions by repeating the procedure using different random vectors $\boldsymbol{\gamma}$.

It is worth mentioning that this circular monodromy loop may also be constructed in a *native* parameter space $\boldsymbol{p}$ of the system if the system is of the form $\mathbf{f}(\mathbf{w}, \boldsymbol{p}) = \mathbf{0}$. In the context of CDPR kineto-statics, the parameter space comprising of unstrained cable lengths $L_j, j = 1, 2, \ldots, 8$, and/or the material parameters ν and λ corresponds to such a native parameter space. However, these native parameters do not appear in a linear manner in the kineto-static equations unlike the artificial parameter space $\boldsymbol{\tau}$. Performing monodromy loops in a parameter space that appears linear can help improve the local conditioning of the system during path tracking, which justifies the introduction of artificial parameters. The native parameters are set as true constants during the numerical experiments.

With this construction, an algorithm for computing the real roots of the non-algebraic system $\mathbf{f}(\mathbf{w}) = \mathbf{0}$ is outlined as follows:

1. **Initialization**: We start by finding an initial starting solution $\mathbf{w}[0] = \mathbf{w}^*$ using a local Newton's method with an initial guess over the complex field. Note that finding a real starting root using Newton's method, while preferable, is non-trivial, especially in large systems. Hence, we rely on complex starting roots which are fairly easy to obtain within say 200 trials. Care must be taken to ensure that the starting root satisfies the system of equations within a small tolerance, say 10^{-8}, and is also a non-singular root of the system with a singular value no smaller than, say 10^{-16}. The algorithm allows for the possibility of starting with a user-defined initial point if it represents a good numerical approximation of a root.
2. **Monodromy Construction**: A random vector $\boldsymbol{\gamma}$ is initialized to define a monodromy loop for the current iteration. In order to track the curve defined by $\mathbf{H}(\mathbf{w}[t], t) = \mathbf{0}$, an equivalent ordinary differential system of equations (ODEs), referred as Davidenko equations [44, 27], are derived from Eq. (12,13) as follows:

$$[\mathbf{f}_{\mathbf{w}}]\mathbf{w}'[t] = -2\pi i \; \boldsymbol{\gamma} \; e^{2\pi i t}, \quad \mathbf{w}[0] = \mathbf{w}^*. \tag{14}$$

   This ODE system poses the path/curve tracking problem as an initial value problem (IVP). In this, we start from $t = 0$ with the initial point $\mathbf{w}^*$ and march forward in t. The IVP at hand cannot be effectively solved using a straightforward integrator due to the tendency of numerical errors to

accumulate over time. Hence, predictor-corrector path tracking techniques are commonly employed to resolve such problems in numerical continuation [44, 27]. More details on the predictor-corrector follow.

3. **Path Tracking**: Each step involves a coarse prediction half-step followed by an active Newton's correction half-step, which solves $\mathbf{H}(\mathbf{w}[t], t) = \mathbf{0}$ using the coarse prediction as the guess. We employ an adaptive step-size mixed precision tracker similar to the ones in [51, 52]. It uses double precision for numerical computations typically but boosts the precision if required up to 32 decimal digits. We largely adopt Algorithm 3 from [52] with slight modifications. For the prediction half-step, this algorithm employs a Padé predictor, a rational function approximation with a numerator polynomial of order 2 and a denominator polynomial of order 1. Because Padé rationals provide local approximations for both the function and the nearest pole(s) [51], this predictor is effective at navigating *near* path crossing scenarios, i.e., when $|[\mathbf{f_w}]| \ll 1$. Note that this Padé predictor requires computing local derivatives up to the fourth order, denoted as $\mathbf{w}^{(j)}[t]$ for $j = 1, 2, 3, 4$ at each step. The local derivatives up to the third order are useful to predict $\mathbf{w}$ at the next step, while the fourth order derivative is used to estimate the approximation error, as is standard in ODE predictors. When computing these derivatives, it is not recommended to use finite difference schemes as the results are impacted by catastrophic cancellation. Instead, we employ chain rules derived symbolically at time t to evaluate them numerically. Here, the first order derivative $\mathbf{w}'[t]$ is given by Eq. (14). For computing the higher order derivatives $\mathbf{w}^{(j)}[t],\ j = 2, 3, 4$, the following linear systems are derived:

$$[\mathbf{f_w}]\mathbf{w}^{(j)}[t] = \mathbf{r}_j[t],$$
$$\text{where} \quad \mathbf{r}_j[t] = -\left(\left(\frac{d}{du}\right)^j \mathbf{f}\left(\mathbf{w} + \sum_{k=1}^{j-1} \frac{1}{k!}\mathbf{w}^{(k)}[t]\ u^k\right)\right)\Bigg|_{u=0} - (2\pi i)^j\ \boldsymbol{\gamma}\ e^{2\pi i t}.$$

Refer to [53] for more information about the derivation of these rules from Eq. (14). Note that the same coefficient Jacobian matrix $[\mathbf{f_w}]$ applies to all the linear systems associated with all orders $j = 1, 2, \ldots$. The symbolic expressions for $[\mathbf{f_w}]$ and $\mathbf{r}_j[t]$ can be derived using any symbolic computer algebra system such as Wolfram Mathematica [54]. Using these rules, one can numerically compute the local derivatives in a sequential manner, starting from the first order $\mathbf{w}'[t]$. These computations are cost-effective, employing LU decomposition to solve the associated coefficient matrix $[\mathbf{f_w}]$ once and applying it for different vectors $\mathbf{r}_j[t]$ determined by the chain rules. The local derivatives not only assist in making local predictions but also play a crucial role in estimating an appropriate step length, taking into account approximation error estimates and the distance to the nearest pole.
For the correction half-step, this algorithm uses Newton's corrector as suggested in [52] to improve the results of the predictor. This algorithm incorporates a certification test given by Newton-Kantorovich [26] based on an approximation of the contraction rate of the successive Newton iterates. The purpose of this test is to ensure that the prediction is within the trust region and free of path jumps at all times. If this test fails at any step, the corresponding prediction is rejected. The step size is then reduced, and numerical precision is increased if necessary before proceeding again. The step size may be reduced up to a lower limit such as 10^{-16} upon which the path is truncated. Due to the iterative nature and the involvement of random processes in the overall algorithm, any root may often be uncovered multiple times through alternative paths in subsequent iterations. Hence, the significance of each individual path diminishes. This aspect of the algorithm allows for a more aggressive reduction in step size compared to the step reduction rule suggested in [52], facilitating the prompt truncation of numerically challenging paths to save compute time.

4. **Root Accumulation**: As the path parameter t crosses a positive integer, we record the corresponding root, which is already certified based on the Newton-Kantorovich theorem as mentioned before. Among all the recorded roots in each iteration, we append only the distinct real ones to the list of starting points. Retaining all the complex roots would quickly become unmanageable, as these tend to vastly outnumber the real roots. We declare a root to be real if every imaginary component is smaller than a threshold, which in the present work was set to 10^{-8}. This threshold may be informed by an analysis of the distribution of imaginary components of the roots of the system under consideration. Sharpening of the numerical approximation via Newton's method using high precision can increase the robustness of such checks. An alternative to naive thresholding could be certified reality tests based on interval

methods [24] or Smale's α-theory [55, 56]. Following this, care must be taken to ensure that duplicate instances of roots are rejected, including alternative representations according to group actions if any. Each path continues until one of the following criteria is met:

- A circuit is completed, meaning that we return to the initial root $\mathbf{w}^*$.
- A previously known real start point or an alternate member of its group is recorded.
- A path fails due to numerical issues.
- A path proceeds without yielding any real roots for m successive instances of t crossing a positive integer.

The truncation parameter m may be tuned specific to the system for addressing cases when paths proceed indefinitely picking up complex roots especially in non-algebraic systems. Paths which are curtailed for the latter two reasons may be pursued along the negative t direction as well before initiating the next iteration.

5. **Iterative Refinement**: The above steps are repeated with different randomly generated $\boldsymbol{\gamma}$ and an updated list of real start points after each iteration. Hence, as more roots are found through the iterations, the number of paths tracked in an iteration also increases progressively. These paths may be tracked simultaneously in a parallel implementation. This process continues until either very few or no new real roots are found for, say, ten successive iterations, or a preset maximum limit of iterations/roots is reached. For better exploration of the variable space, we reseed a new random complex start point for each iteration using the Newton's method, along with the current set of real start points. We may also tune the magnitude of the elements of the randomly generated vector $\boldsymbol{\gamma}$ as a hyper-parameter to increase the yield. Further details regarding the generation of this vector and other algorithmic aspects specific to CDPR forward kineto-statics will be discussed in Section 4.2.

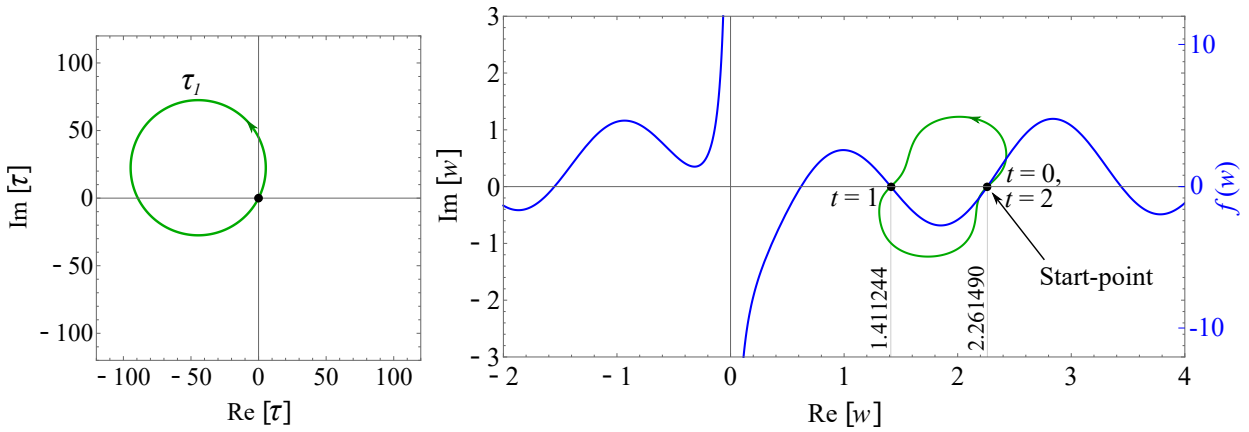
The algorithm described here employs a simple strategy for terminating the algorithm based on negligible or zero returns. This termination check is computationally efficient but lacks sophistication, yet no better alternatives exist for non-algebraic systems at this time. In the case of polynomial systems with a finite set of isolated roots, a computationally inexpensive second-order local linear trace test exists [57], which proves sufficient to conclude if all the roots belonging to a numerically irreducible set have been discovered. A termination criterion based on the mark-and-recapture model [41], utilizing likelihood estimates based on diminishing returns, also exists for polynomial systems. Another validation method for determining if all roots within a given variable bound have been found is through the application of multi-variate generalizations of Rouché's theorem [31, p. 153]. For an illustration of this technique in polynomial systems, refer to [58]. However, the challenge lies in the generalization of any such termination criteria to non-algebraic systems while maintaining computational affordability. This is beyond our present scope.

3.2. Illustration

Let us consider a root-finding problem of a non-algebraic equation:

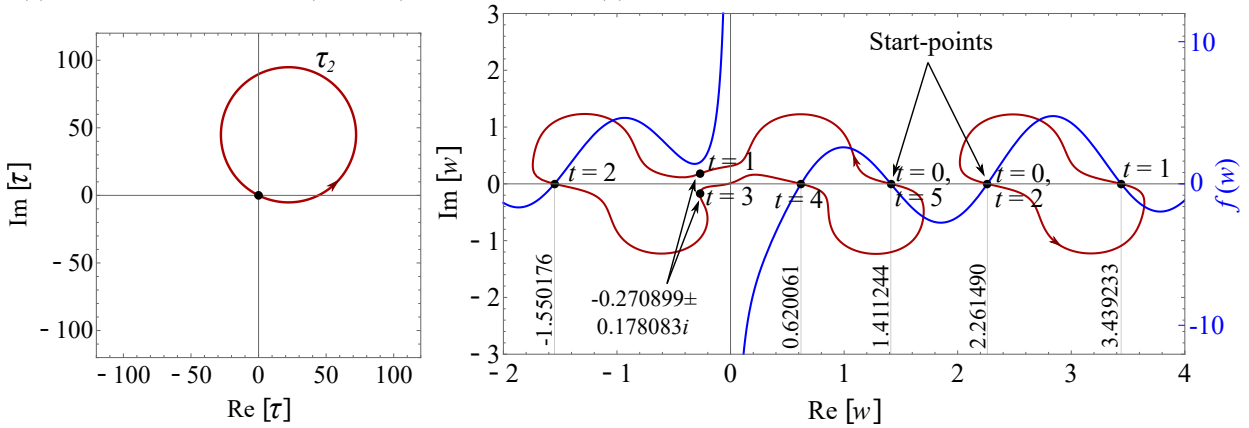
$$f(w) = 0.84 - \frac{1}{w} - \cos[w] - \frac{10}{3}\cos\left[\frac{10w}{3}\right] = 0.$$

This equation represents the stationary points of an objective function taken from [59]. For solving this equation, we construct a homotopy as given by Eq. (12,13). The algorithm is initiated from a real start point $w = 2.261490$ found through Newton's method with an initial guess. During the first iteration, the algorithm uses a randomly generated complex number $\gamma_1 = -44.6510 + 22.5008i$ of magnitude 50. The corresponding monodromy loop $\tau_1 = \gamma_1(1 - e^{2\pi it})$ is shown as an Argand diagram in Fig. 3a. When $t \in \mathbb{Z}$, $\tau_1 = 0$. The path proceeds from $t = 0$; picks up a new root $w = 1.411244$ at $t = 1$; and completes a circuit by returning to the start point at $t = 2$. This progression is illustrated in Fig. 3b. The real-valued function $f(w)$ is shown as an overlay on the same plot, with scale on the right, for correspondence of the roots of $f(w)$ when $t \in \mathbb{Z}$. The next iteration is initiated with a different random complex number $\gamma_2 = 22.1502 + 44.8260i$, see Fig. 3c. In this case, the two start points found already are seen to follow distinct circuits, each picking up new roots in process before completing the respective circuits as shown in Fig. 3d. The supplementary material contains two animations visualizing these iterations. Note that the number of unique roots within



(a) Argand diagram of $\tau_1 = \gamma_1(1 - e^{2\pi i t})$

(b) Iteration 1 visualized in a Argand diagram of w



(c) Argand diagram of $\tau_2 = \gamma_2(1 - e^{2\pi i t})$

(d) Iteration 2 visualized in a Argand diagram of w

Figure 3: Illustration of random monodromy loops algorithm in a 1D root-finding problem

a circuit can vary widely depending on the system and may not be known *a priori*. Furthermore, these roots need not all be real. This example showcases that two roots which belong in the same circuit in one monodromy may belong in different circuits in another. Such randomly permuted maps allow the algorithm to proceed and compute new roots efficiently with subsequent iterations. It is worth noting that the function considered has a pole at $w = 0$. This impacts path tracking in iteration 2 between $t = 3$ and $t = 4$ as the path goes very near to the pole, but successfully navigates it. As this example shows, numerical continuation techniques may even be used in some non-analytic systems with isolated poles because the probability of any random path reaching an isolated pole is nearly zero. This is unlike a branch cut, which is not an isolated discontinuity. In the following section, we apply this technique to solve the forward kineto-statics of CDPRs and benchmark our results against the existing ones from literature.

4. Numerical examples

In this section, we solve the forward kineto-static problem of an eight-cable CoGiRo CDPR [42, 20] via the heuristic random monodromy loops algorithm. This CDPR system has an end-effector platform with six DOF which is controlled by eight cables. The forward kinematic problem as per the formulation in Section 2 results in a non-algebraic system of 31 equations in 31 unknowns.

4.1. System Specifications and Material Properties

The cable properties are as follows:

$$\mu = 0.079 \text{ kg m}^{-1}, \quad E = 100 \text{ GPa}, \quad A = 4\pi \cdot 10^{-6} \text{ m}^2.$$

Acceleration due to gravity is assumed a constant $g = 9.81$ m s^{-2}. The coordinate points of the cable base points in the global frame and the end-effector points in the local frame are listed in Table 1. The end-effector center of mass is assumed to coincide with the origin of the local frame and it carries $M = 1$ kg. Based on these specifications, the material constants of this system evaluate to $\nu = 0.616717 \cdot 10^{-6}$ m^{-1} and $\lambda = 12.658228$ m. We analyze four distinct data sets, namely Data Sets $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, and $\mathcal{D}$, which contain unstrained cable lengths for the forward kinematic problem, as listed in Table 2. In the following section, we present additional details regarding the implementation of the heuristic algorithm that are specific to the present study.

Table 1: Coordinate Points for an Eight-Cable Cable-Driven Parallel Robot (CDPR) System

Base Points ($\mathbf{A}_j$, Global Frame)			
Label	x (m)	y (m)	z (m)
$\mathbf{A}_1$	-7.175120	-5.243980	5.462460
$\mathbf{A}_2$	-7.315910	-5.102960	5.472220
$\mathbf{A}_3$	-7.302850	5.235980	5.476150
$\mathbf{A}_4$	7.182060	5.347600	5.488300
$\mathbf{A}_5$	-7.160980	5.372810	5.485390
$\mathbf{A}_6$	7.323310	5.205840	5.499030
$\mathbf{A}_7$	7.301560	-5.132550	5.489000
$\mathbf{A}_8$	7.161290	-5.269460	5.497070

End-Effector Points ($\mathbf{p}_j$, Local Frame)			
Label	x (m)	y (m)	z (m)
$\mathbf{p}_1$	0.503210	-0.492830	0.000000
$\mathbf{p}_2$	-0.509740	0.350900	0.997530
$\mathbf{p}_3$	-0.503210	-0.269900	0.000000
$\mathbf{p}_4$	-0.503210	0.492830	0.000000
$\mathbf{p}_5$	0.496070	0.355620	0.999540
$\mathbf{p}_6$	0.499640	-0.340280	0.999180
$\mathbf{p}_7$	0.502090	0.274900	-0.000620
$\mathbf{p}_8$	-0.504540	-0.346290	0.997520

Table 2: Unstrained Lengths (m) of Eight Cables ($L_j, j = 1, 2, \ldots, 8$) - Four Data Sets ($\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, $\mathcal{D}$)

Data Set	L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8
$\mathcal{A}$	10.538225	11.963397	10.135355	7.703723	10.615617	8.537152	9.969606	8.555879
$\mathcal{B}$	8.306209	7.608231	11.163294	12.406223	11.380175	11.947601	9.495229	8.985027
$\mathcal{C}$	12.192125	11.247648	9.199310	7.929109	8.596630	7.311818	10.839055	10.721407
$\mathcal{D}$	12.924710	11.785460	10.720089	8.037326	10.425254	6.715571	10.163088	9.316678

4.2. Key Considerations

The random vector $\boldsymbol{\gamma}$ used to construct monodromy loops in the artificial parameter space $\boldsymbol{\tau}$ is of dimension 31, matching the number of equations. Given that the kineto-static equations — quaternion constraint equation, position equations, force and torque equations — encompass various units, the elements of $\boldsymbol{\gamma}$ are generated within ranges that appropriately scale each component of the monodromy loops. Let $\gamma_\mathrm{q}, \boldsymbol{\gamma}_\mathrm{p}$, and $\boldsymbol{\gamma}_\mathrm{f}$ represent the set of elements corresponding to these equation types, respectively, such that the random vector $\boldsymbol{\gamma} = (\gamma_\mathrm{q}; \boldsymbol{\gamma}_\mathrm{p}; \boldsymbol{\gamma}_\mathrm{f})^\top$ is a composition of these sets. The element of the $\boldsymbol{\gamma}$ vector corresponding to the unit quaternion equation, namely γ_q, is intentionally set to be identically zero, requiring only 30 non-zero random complex numbers in each iteration. This is done to preserve the unit quaternion constraint equation without modification throughout the numerical computations, thereby maintaining the regularity assumptions of the underlying physical system during monodromy loops. For generating the other 30 random complex numbers in $\boldsymbol{\gamma}$, a magnitude limit of 0.5 is set on elements of $\boldsymbol{\gamma}_\mathrm{p}$, and 5 for those of $\boldsymbol{\gamma}_\mathrm{f}$. These values were obtained through a one-time coarse tuning to align with the scale of terms in the respective equations. The primary purpose of this tuning is to prevent trivial circuits, where starting points map onto themselves due to disproportionately small elements in $\boldsymbol{\gamma}$. Recall that the elements of $\boldsymbol{\gamma}$ correspond to the radii of the circular monodromy loops. Excessive magnitudes adversely affect computation time; thus, an optimal limit is maintained. While these values generally perform well for different cable length inputs in a specific CDPR system, their universal applicability to other CDPR systems with varying architectures, scales, and platform masses may be limited and should be adjusted accordingly.

Initial guesses for the Newton's method used to find start points are also crucial. The position coordinates of the end-effector can be chosen as complex values within a magnitude limit appropriate for the CDPR operating range's scale. Complex-valued guesses are selected for α and β with a magnitude limit of 10^{-3},

Table 3: Computational Summary of forward kinematics for a CDPR with eight cables

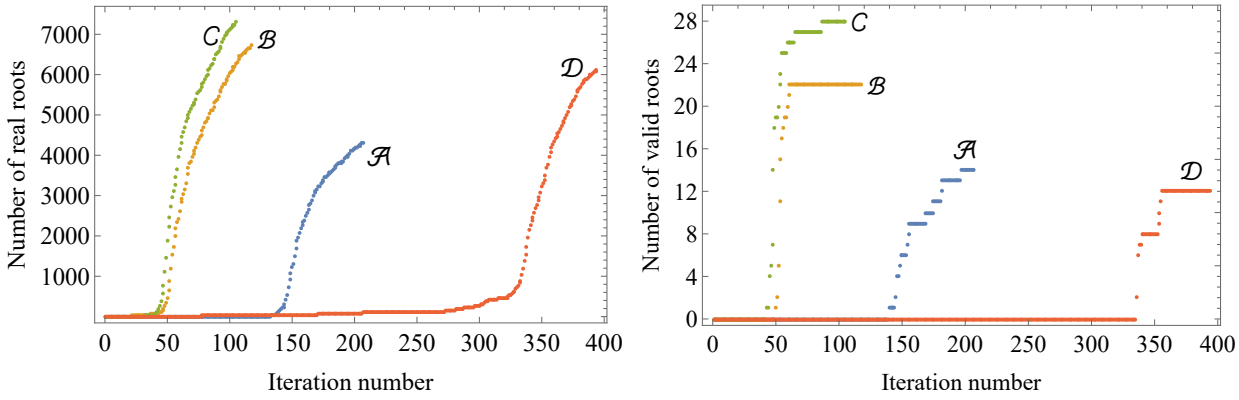
Data Set	# Iterations	# Real Roots	# Valid Roots ($\alpha_j > \beta_j \ \forall \ j$)	# Valid Roots of Benchmark [20]	Compute Time (hr)
$\mathcal{A}$	207	4306	14	11	2.64
$\mathcal{B}$	117	6717	22	21	3.33
$\mathcal{C}$	105	7301	28	19	3.22
$\mathcal{D}$	394	6119	12	8	3.46

while ϕ is set within a magnitude of π for all cables. Guess values for quaternion variables are normalized such that the unit quaternion equation is satisfied.

During monodromy path tracking, as described earlier in Section 3.1, a path is truncated if no real root is found for four consecutive instances of the homotopy path parameter t crossing an integer. While it is possible that the path might still find a real root if pursued further, we have determined through trial and error that the trade-off between computation time and yield is poor beyond four for this system. The total number of complex roots for this non-algebraic system is theoretically unbounded, and some paths are observed to proceed indefinitely, always landing on complex roots, necessitating this truncation limit.

As for overall algorithm termination criteria, we define the algorithm to terminate when the percentage increase of real roots from successive iterations falls below 1% and the number of valid roots remains constant for 10 consecutive iterations. To prevent premature termination, we set a minimum limit of 1000 real roots initially.

4.3. Results



(a) Progression of real roots

(b) Progression of valid roots

Figure 4: Progression of the number of roots plotted against the iterations for the kineto-statics of an eight-cable CDPR

Table 3 reports a summary of the computations performed in the four data sets chosen from [20]. All the computations for each of the four data sets were carried out using the Wolfram Mathematica software [54] on a dual 24-core Intel® Xeon™ 2.30 GHz system in the Center for Research Computing at the University of Notre Dame. In all four cases, the algorithm converged with both the percentage increase of real roots diminishing below 1% and the number of valid roots stalling for 10 consecutive iterations. In all cases, the heuristic numerical continuation algorithm found all the valid roots reported in the existing benchmark set [20] and more. In Data Sets $\mathcal{C}$ and $\mathcal{D}$, the algorithm identified up to 50% more configurations, with the majority of the newly discovered configurations exhibiting one or more cable tensions significantly higher than those reported in the benchmark. Although these newly found configurations may be practically unfeasible, they could still prove useful in generating workspace points through local continuation methods. All the valid configurations found in our numerical experiments are made available to the reader via supplementary material along with their visualizations. For each configuration, the numerical values are reported in the following order of variables: $P_x, P_y, P_z, q_0, q_1, q_2, q_3, \{\alpha_j, \beta_j, \phi_j\}$, $j = 1, 2, \ldots, 8$. Figure 4a illustrates the

algorithm's progression across iterations in all four data sets. The initial iterations are computationally inexpensive, and a critical mass of start points is reached before the algorithm accelerates in finding real roots as seen in Fig. 4a. The point at which this critical mass is reached can vary between runs due to the randomness associated with the technique, and it is not guaranteed to occur at the same iteration number or close. However, because of the low computational cost of the initial iterations, this variability does not pose a significant issue.

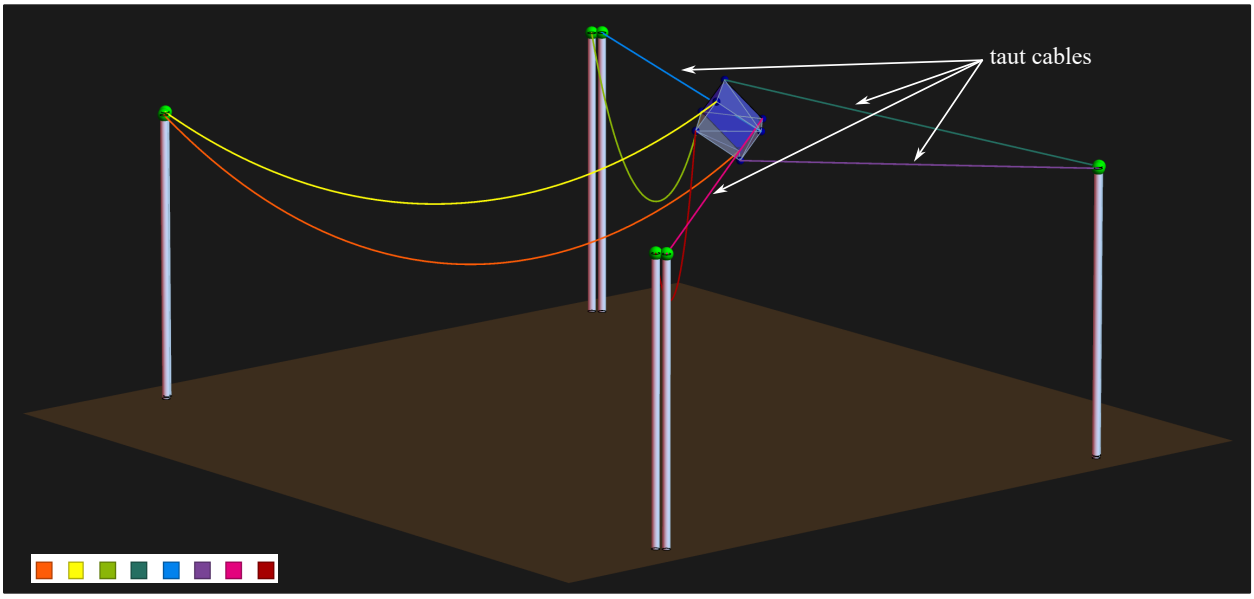


Figure 5: An equilibrium configuration (#16) to the forward kinematic problem associated with Data Set $\mathcal{C}$ with four cables in high tension

As mentioned earlier, during the monodromy runs, the inequality condition $\alpha_j > \beta_j \ \forall \ j$ is disregarded. While post-processing after each iteration, we check for this condition to identify the valid roots among the real roots found. Figure 4b records the progression of the number of valid roots against the number of iterations. It illustrates that in Data Set $\mathcal{B}$, all the valid roots are found in just a few iteration sequences. This is because all 22 roots found represent well-behaved configurations close to each other in the variable space, with relatively small tension forces in all the sagging cables. For example, see Fig. 6 for one such configuration. The cable profiles are drawn using Eq. (9). In the other data sets, which contain taut cable configurations with large cable tension forces (see Fig. 5) besides well-behaved ones, the number of valid roots increases in steps with batches of configurations occurring in iteration sequences. This has implications for computation time incurred. For instance, we selected an intensive termination criterion resulting in approximately three hours of computation time in each case (see Table 3). However, the majority of well-behaved configurations are discovered within the first hour of computation. Thus, it is important to note that the reported computation time depends on the desired level of diminishing returns. Furthermore, the computation time reported in Table 3 should be linked to the number of real roots, which ranges in the thousands, rather than to the number of valid roots. The latter represents only a small fraction of the total real roots in the context of CDPRs. The real roots undergo *post-facto* filtering through inequality checks to determine the valid roots, unrelated to the algorithm itself. Thus, in applications where such checks are not relevant, the heuristic algorithm demonstrates the potential to uncover and compute thousands of roots in large non-algebraic systems.

In Fig. 7, we present a breakdown of all the real roots based on the number of cables in each root that satisfy the valid cable condition $\alpha > \beta$. The numbers indicated correspond to the observed distribution, which is depicted as a histogram in red. The last bar in the histogram signifies the valid roots where all eight cables satisfy this condition. The histogram reveals an intriguing trend: as more real roots are discovered, we observe the emergence of an approximate binomial distribution in all four cases. If one assumes an equal probability $\sigma \in [0, 1]$ for satisfying each valid cable inequality condition in a real root, then the expected

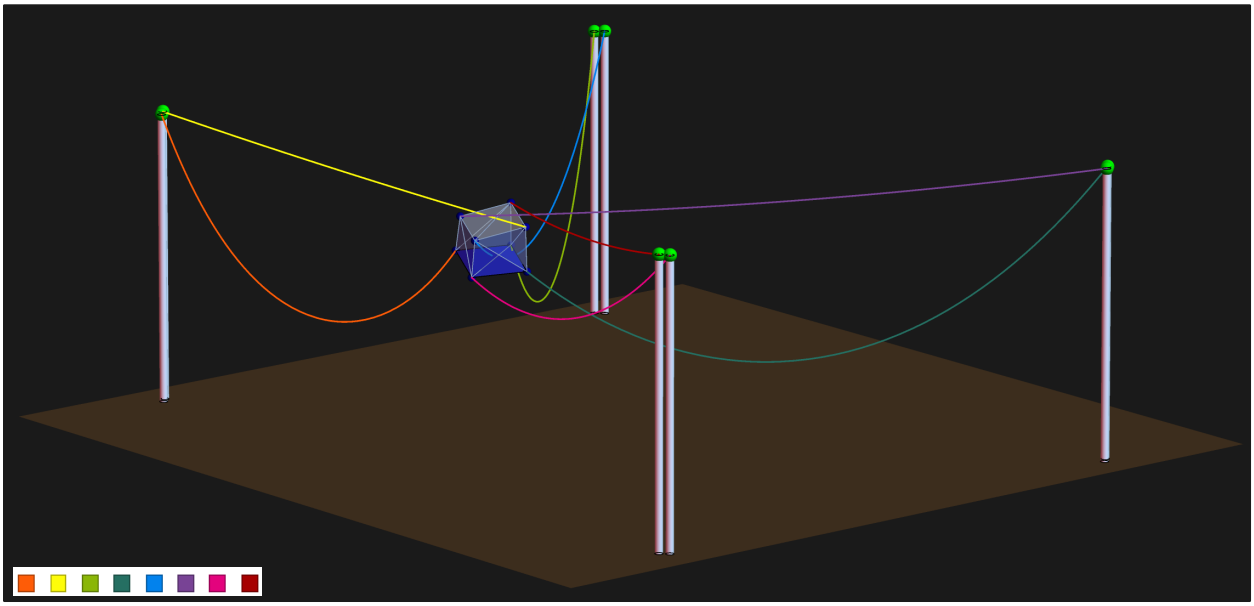


Figure 6: A previously unreported equilibrium configuration (#14) to the forward kinematic problem associated with Data Set $\mathcal{B}$

binomial probability mass function for a real root having k valid cables is given by

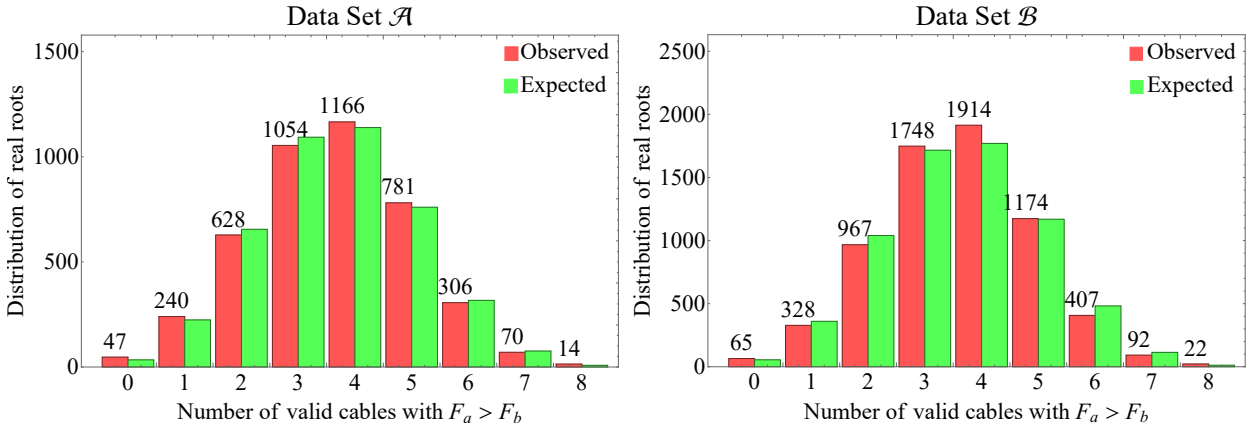
$$P(X = k) = \frac{8!}{k!\ (8-k)!} \cdot \sigma^k \cdot (1-\sigma)^{8-k}, \quad k = 0, 1, \ldots, 8. \tag{15}$$

The binomial distribution in Fig. 7 is not symmetric; in other words, σ skews slightly lower than 0.5 in all four cases, approximately around 0.45. The corresponding expected binomial distributions, derived using a maximum likelihood estimator are shown side-by-side in green in the respective figures. However, it is important to note that a Pearson chi-square goodness-of-fit test yielded significant results, indicating a bad distribution fit, in three out of the four cases, except for the first one when using a one-sided significance level of 0.05 and seven degrees of freedom. This choice of seven degrees of freedom for the chi-square goodness-of-fit test is appropriate due to the estimation of a single parameter, σ, using the maximum likelihood estimator in a scenario with nine categories. It remains to be studied whether this binomial distribution trend occurs across various CDPR systems and whether it can be employed to establish a goodness-of-fit-based termination criterion for the random monodromy loops algorithm applied in the context of global kineto-statics for CDPRs.

5. Properties and Relations

Numerical continuation via random monodromy loops was shown to be computationally efficient for solving the global kineto-statics of large CDPRs in comparison with existing approaches. The primary focus of the present work is not to provide a real-time routine but to offer a robust way to discover nearly all possible configurations for a handful of unstrained cable lengths. This will facilitate the generation of data sets of workspace points in conjunction with local parameter continuation methods. Ultimately, this will aid in the creation of full workspace models that may be useful for real-time implementation [22, 16]. In this regard, the current approach is similar in scope to previous approaches [25, 34] but presents different trade-offs as summarized below.

Interval analysis methods [25], when implemented properly, can guarantee the discovery of all roots within set variable bounds due to their exhaustive search. However, in large systems with many cables, these methods encounter the 'curse of dimensionality', especially when expanding the search bounds, highlighting a level of uncertainty in determining the optimal choice of these bounds. Unlike these search methods, analytic continuation techniques such as the one in the present work can seamlessly traverse the variable



(a) Expected $\sigma = 0.454743$, p-value $= 0.085406$

(b) Expected $\sigma = 0.452099$, p-value $= 2.81 \cdot 10^{-7}$

(c) Expected $\sigma = 0.449288$, p-value $= 1.36 \cdot 10^{-6}$

(d) Expected $\sigma = 0.445599$, p-value $= 0.003754$

Figure 7: Comparison of observed distribution (red) vs. expected distribution (green) of the number of real roots based on the number valid cables within each root

space unrestricted by variable bounds. This flexibility could also be a drawback if the variable bounds are to be strictly defined. Nevertheless, in the current technique, the dimensionality problem manifests differently in the form of invalid real roots that violate the cable inequality condition $\alpha > \beta$. As a rule of thumb, for an n-cable system, one can generally expect that only a small fraction of the real roots will be valid. For example, assuming equal probability for a real root to satisfy each valid cable inequality condition, this fraction would be 2^{-n}. Thus, this is a trade-off we must accept since, during monodromy runs, real roots may transition from invalid to valid and vice versa.

Unlike classical continuation methods [34], this heuristic continuation method employing monodromy eliminates the necessity of constructing start systems. This removes a significant overhead as the method can be easily initiated using local Newton's method. However, due to the heuristic nature of root accumulation, this method, by itself, does not guarantee the discovery of all possible roots globally or even within set variable bounds. The diminishing returns in the total number of real roots found over the iterations offer weak indications towards the near completeness of the solution set. Nevertheless, numerical experiments indicate that these heuristics are robust and consistently uncover all reported solutions in the literature faster.

In terms of computation time, almost all of it is spent on path tracking, with a negligible amount allocated to data processing within and in between iterations. Therefore, improving path tracking routines and implementing efficiency enhancements can further reduce computation time. Smarter root accumulation strategies that promptly discard sufficiently explored start points as the iterations progress may be considered. By not retaining all the real roots indiscriminately, this can help save compute time.

6. Conclusion

We describe a numerical continuation method to solve the global forward kineto-statics of large CDPRs with sagging cables. As a first step, we reformulate the popular Irvine cable model to eliminate all multi-valued functions, enabling effective analytical continuation for studying these systems. This new kineto-static model depends on two consolidated material constants, in addition to the geometric parameters, and offers computational advantages for modeling and solving CDPR systems with large tension forces in the cables. Next, we present a novel strategy for root accumulation utilizing random monodromy loops in an iterative manner to find roots for nonlinear, non-algebraic, and complex-analytic systems. This strategy involves tracking circuits while performing monodromy loops instead of following lone paths, potentially identifying multiple roots in the process. The rationale behind this approach is that multiple valid roots are often closely located in the variable space and this continuation technique can efficiently uncover them. With this technique, we solve the global kineto-static problem of an eight-cable spatial CDPR system comparatively faster than existing interval analysis methods. We employ a termination criterion based on diminishing returns across iterations so that this technique cannot, in theory, guarantee the finding of all real roots of the system. Despite this, in the considered numerical examples, the algorithm not only discovered all previously reported solutions within predefined variable bounds mentioned in the literature but also identified several additional solutions outside these bounds. This method can augment existing local parameter continuation methods to generate large workspace data sets, which may be used to build workspace models for CDPRs with sagging cables. The heuristic numerical continuation technique presented here also holds potential relevance for other applications dealing with high-dimensional non-algebraic systems.

Acknowledgment

The authors express their gratitude to the anonymous reviewers for their comprehensive critique, which greatly improved the paper. The first author was supported in part from Wolfram Research, Inc., and, in particular, acknowledge mentorship from Daniel Lichtblau. This material is based upon work supported by the National Science Foundation under Grant No.'s CMMI-2041789 and CMMI-2144732. The last author was supported in part by the Huisking Foundation, Inc.

Appendix A. Alternative Representation of the Irvine Cable Model

Equations (5, 8), which define the Irvine cable model in terms of the variables α and β, can be further revised using the following linear change of variables:

$$\alpha = \chi + \psi, \quad \beta = \chi - \psi.$$

Using this transformation, the cable model simplifies into the following:

$$\begin{pmatrix} b_x \\ 0 \\ b_z \end{pmatrix} = \frac{L}{\sinh[\psi]} \begin{pmatrix} \left(\frac{\nu L}{2} + \psi\right) \operatorname{sech}[\chi] \\ 0 \\ \left(\frac{\nu L}{2} \cosh[\psi] + \sinh[\psi]\right) \tanh[\chi] \end{pmatrix},$$

where $\nu = \frac{\mu g}{EA}$. The necessary inequality condition $\alpha > \beta$ reduces to $\psi > 0$ under this transformation. In [60], we demonstrate that this modification facilitates the real-time determination of cable tension and profile in elastic sagging cables uniquely for any valid set of geometric and material parameters.

This model also presents a geometric understanding of the Irvine cable model. Here, $\psi > 0$ directly gives a measure of cable sag. The greater the value of ψ, the greater the cable sag for a given χ, with $\psi \to 0^+$ approaching a fully taut configuration. Moreover, the variable χ ranges from $-\infty$ to ∞ and roughly indicates the orientation of the cable axis within the cable half plane considered, $b_x > 0$. In this context, the term 'cable axis' refers to the line joining the cable endpoints. When $\chi = 0$, the cable axis is horizontal. Positive values of χ correspond to cable configurations in the quarter where $b_z > 0$, while negative values correspond to configurations in the quarter where $b_z < 0$. The limits at $\pm\infty$ represent vertical orientations of the cable axis.

References

[1] E. Barnett, C. Gosselin, Large-scale 3D printing with a cable-suspended robot, Additive Manufacturing 7 (2015) 27–44.

[2] J. Merlet, Y. Papegay, The new exhibition blind machines, a large 3D printing machine, in: 2023 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2023, pp. 9721–9727.

[3] J.-B. Izard, A. Dubor, P.-E. Hervé, E. Cabay, D. Culla, M. Rodriguez, M. Barrado, Large-scale 3D printing with cable-driven parallel robots, Construction Robotics 1 (2017) 69–76.

[4] J.-P. Merlet, Marionet, a family of modular wire-driven parallel robots, in: Advances in Robot Kinematics: Motion in Man and Machine: Motion in Man and Machine, Springer, 2010, pp. 53–61.

[5] R. G. Roberts, T. Graham, T. Lippitt, On the inverse kinematics, statics, and fault tolerance of cable-suspended robots, Journal of Robotic Systems 15 (10) (1998) 581–597.

[6] M. Carricato, J.-P. Merlet, Stability analysis of underconstrained cable-driven parallel robots, IEEE Transactions on Robotics 29 (1) (2012) 288–296.

[7] H. M. Irvine, Cable structures, The MIT Press Series in Structural Mechanics, 1981.

[8] J.-P. Merlet, Some properties of the Irvine cable model and their use for the kinematic analysis of cable-driven parallel robots, Mechanism and Machine Theory 135 (2019) 271–280.

[9] E. Stump, V. Kumar, Workspaces of Cable-Actuated Parallel Manipulators, Journal of Mechanical Design 128 (1) (2005) 159–167.

[10] G. Abbasnejad, M. Carricato, Direct geometrico-static problem of underconstrained cable-driven parallel robots with n cables, IEEE Transactions on Robotics 31 (2) (2015) 468–478.

[11] A. Berti, J.-P. Merlet, M. Carricato, Solving the direct geometrico-static problem of underconstrained cable-driven parallel robots by interval analysis, The International Journal of Robotics Research 35 (6) (2016) 723–739.

[12] K. Kozak, Q. Zhou, J. Wang, Static analysis of cable-driven manipulators with non-negligible cable mass, IEEE Transactions on Robotics 22 (3) (2006) 425–433.

[13] J.-P. Merlet, R. Tissot, A panorama of methods for dealing with sagging cables in cable-driven parallel robots, in: International Symposium on Advances in Robot Kinematics, Springer, 2022, pp. 122–130.

[14] M. Raghavan, B. Roth, Solving Polynomial Systems for the Kinematic Analysis and Synthesis of Mechanisms and Robot Manipulators, Journal of Vibration and Acoustics 117 (B) (1995) 71–79.

[15] S. Briot, J.-P. Merlet, Direct kinematic singularities and stability analysis of sagging cable-driven parallel robots, IEEE Transactions on Robotics (2023).

[16] P. B. Edwards, A. Baskar, C. Hills, M. Plecnik, J. D. Hauenstein, Output mode switching for parallel five-bar manipulators using a graph-based path planner, in: 2023 IEEE International Conference on Robotics and Automation (ICRA), 2023, pp. 9735–9741.

[17] O. Bohigas, M. Manubens, L. Ros, Singularities of Robot Mechanisms: Numerical Computation and Avoidance Path Planning, Vol. 41 of Mechanisms and Machine Science, Springer International Publishing, Cham, 2017.

[18] C. Gosselin, J. Angeles, Singularity analysis of closed-loop kinematic chains., IEEE transactions on robotics and automation 6 (3) (1990) 281–290.

[19] M. Urizar, V. Petuya, O. Altuzarra, A. Hernandez, Computing the configuration space for motion planning between assembly modes, in: Computational Kinematics: Proceedings of the 5th International Workshop on Computational Kinematics, Springer, 2009, pp. 35–42.

[20] J.-P. Merlet, Data base for the direct kinematics of cable-driven parallel robot (CDPR) with sagging cables, `hal.science/hal-03540335v2` (2021).

[21] I. Chawla, P. Pathak, L. Notash, A. Samantaray, Q. Li, U. Sharma, Inverse and forward kineto-static solution of a large-scale cable-driven parallel robot using neural networks, Mechanism and Machine Theory 179 (2023) 105107.

[22] J.-P. Merlet, Advances in the use of neural network for solving the direct kinematics of CDPR with sagging cables, in: International Conference on Cable-Driven Parallel Robots, Springer, 2023, pp. 30–39.

[23] A. Khovanskii, Fewnomials and Pfaff manifolds, in: Proceedings of the International Congress of Mathematicians, Vol. 1, 1983, p. 2.

[24] J.-P. Merlet, Interval analysis for certified numerical solution of problems in robotics, International Journal of Applied Mathematics and Computer Science 19 (3) (2009) 399–412.

[25] J.-P. Merlet, J. Alexandre-dit Sandretto, The forward kinematics of cable-driven parallel robots with sagging cables, in: Cable-Driven Parallel Robots: Proceedings of the Second International Conference on Cable-Driven Parallel Robots, Springer, 2015, pp. 3–15.

[26] J. M. Ortega, The Newton-Kantorovich Theorem, The American Mathematical Monthly 75 (6) (1968) 658–660.

[27] D. J. Bates, A. J. Sommese, J. D. Hauenstein, C. W. Wampler, Numerically solving polynomial systems with Bertini, SIAM, 2013.

[28] D. J. Bates, J. D. Hauenstein, A. J. Sommese, C. W. Wampler, Bertini: Software for numerical algebraic geometry, Available at `bertini.nd.edu`.

[29] H. Tari, H.-J. Su, J. D. Hauenstein, Classification and complete solution of the kinetostatics of a compliant Stewart–Gough platform, Mechanism and Machine Theory 49 (2012) 177–186.

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

[31] L. V. Ahlfors, Complex analysis, McGraw-Hill, Inc., 1979.

[32] D. Mehta, T. Chen, J. D. Hauenstein, D. J. Wales, Communication: Newton homotopies for sampling stationary points of potential energy landscapes, The Journal of chemical physics 141 (12) (2014).

[33] M. M. Plecnik, The kinematic design of six-bar linkages using polynomial homotopy continuation, University of California, Irvine, 2015.

[34] J.-P. Merlet, A generic numerical continuation scheme for solving the direct kinematics of cable-driven parallel robot with deformable cables, in: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2016, pp. 4337–4343.

[35] D. J. Bates, J. D. Hauenstein, A. J. Sommese, C. W. Wampler, Adaptive multiprecision path tracking, SIAM Journal on Numerical Analysis 46 (2) (2008) 722–746.

[36] D. J. Bates, J. D. Hauenstein, A. J. Sommese, C. W. Wampler, II, Stepsize control for path tracking, in: Interactions of classical and numerical algebraic geometry, Vol. 496 of Contemp. Math., Amer. Math. Soc., Providence, RI, 2009, pp. 21–31.

[37] M. M. Plecnik, R. S. Fearing, Finding only finite roots to large kinematic synthesis systems, Journal of Mechanisms and Robotics 9 (2) (2017) 021005.

[38] J. D. Hauenstein, L. Oeding, G. Ottaviani, A. J. Sommese, Homotopy techniques for tensor decomposition and perfect identifiability, Journal für die reine und angewandte Mathematik (Crelles Journal) 2019 (753) (2019) 1–22.

[39] A. Baskar, S. Bandyopadhyay, An algorithm to compute the finite roots of large systems of polynomial equations arising in kinematic synthesis, Mechanism and Machine Theory 133 (2019) 493–513.

[40] T. Duff, C. Hill, A. Jensen, K. Lee, A. Leykin, J. Sommars, Solving polynomial systems via homotopy continuation and monodromy, IMA Journal of Numerical Analysis 39 (3) (2019) 1421–1446.

[41] J. D. Hauenstein, S. N. Sherman, Using monodromy to statistically estimate the number of solutions, in: 2nd IMA Conference on Mathematics of Robotics (2020), Vol. 21, Springer Proceedings in Advanced Robotics, 2022, pp. 37–46.

[42] M. Gouttefarde, D. Q. Nguyen, C. Baradat, Kinetostatic analysis of cable-driven parallel robots with consideration of sagging and pulleys, Advances in Robot Kinematics (2014) 213–221.

[43] C. W. Wampler, A. P. Morgan, A. J. Sommese, Complete Solution of the Nine-Point Path Synthesis Problem for Four-Bar Linkages, Journal of Mechanical Design 114 (1) (1992) 153–159.

[44] E. L. Allgower, K. Georg, Numerical Continuation Methods: An Introduction, Vol. 13, Springer Science & Business Media, 2012.

[45] J. Verschelde, P. Verlinden, R. Cools, Homotopies exploiting Newton polytopes for solving sparse polynomial systems, SIAM Journal on Numerical Analysis 31 (3) (1994) 915–930.

[46] B. Huber, B. Sturmfels, A polyhedral method for solving sparse polynomial systems, Math. Comp. 64 (212) (1995) 1541–1555.

[47] M. M. Plecnik, J. Michael McCarthy, Computational design of Stephenson II six-bar function generators for 11 accuracy points, Journal of Mechanisms and Robotics 8 (1) (2016) 011017.

[48] A. Baskar, M. Plecnik, Synthesis of six-bar timed curve generators of Stephenson-type using random monodromy loops, Journal of Mechanisms and Robotics 13 (1) (2021) 011005.

[49] J. D. Hauenstein, A. C. Liddell Jr, Certified predictor–corrector tracking for Newton homotopies, Journal of Symbolic Computation 74 (2016) 239–254.

[50] HomotopyContinuation.jl, Solving parametrized systems with monodromy - Strategies, `https://docs.juliahub.com/HomotopyContinuation/bmV6V/1.4.3/monodromy/#Strategies-1`, accessed: 2024-01-17.

[51] S. Telen, M. V. Barel, J. Verschelde, A robust numerical path tracking algorithm for polynomial homotopy continuation, SIAM Journal on Scientific Computing 42 (6) (2020) A3610–A3637.

[52] S. Timme, Mixed precision path tracking for polynomial homotopy continuation, Advances in Computational Mathematics 47 (5) (2021) 75.

[53] W. Mackens, Numerical differentiation of implicitly defined space curves, Computing 41 (3) (1989) 237–260.

[54] Wolfram Research Inc., Mathematica, Version 13.2.0.0, Champaign, IL, 2023.

[55] J. D. Hauenstein, F. Sottile, Algorithm 921: alphaCertified: Certifying solutions to polynomial systems, ACM Trans. Math. Softw. 38 (4) (2012).

[56] J. D. Hauenstein, V. Levandovskyy, Certifying solutions to square systems of polynomial-exponential equations, Journal of Symbolic Computation 79 (2017) 575–593.

[57] D. A. Brake, J. D. Hauenstein, A. C. Liddell Jr, Decomposing solution sets of polynomial systems using derivatives, in: International Congress on Mathematical Software, Springer, 2016, pp. 127–135.

[58] J. Verschelde, A. Haegemans, Homotopies for solving polynomial systems within a bounded domain, Theoretical computer science 133 (1) (1994) 165–185.

[59] L. Timonov, An algorithm for search of a global extremum, Engineering Cybernetics 15 (3) (1977) 38–44.

[60] A. Baskar, M. Plecnik, J. D. Hauenstein, C. W. Wampler, A real-time algorithm for computing the tension force in a suspended elastic sagging cable, US Symposium on Mechanical Systems & Robotics and RoManSy (MSR-RoManSy) (2024).