ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference
IDETC 2016
August 21-24, 2016, Charlotte, North Carolina

IDETC2016-60428

FINDING ONLY FINITE ROOTS TO LARGE KINEMATIC SYNTHESIS SYSTEMS

Mark M. Plecnik*
Biomimetic Millisystems Lab
Postdoctoral Scholar
University of California
Berkeley, California 94720
Email: mplecnik@berkeley.edu

Ronald S. Fearing
Biomimetic Millisystems Lab
Department of Electrical Engineering and Computer Sciences
University of California
Berkeley, California 94720
Email: ronf@eecs.berkeley.edu

ABSTRACT

In this work, a new method is introduced for solving large polynomial systems for the kinematic synthesis of linkages. The method is designed for solving systems with degrees beyond 100,000, which often are found to possess a number of finite roots that is orders of magnitude smaller. Current root-finding methods for large polynomial systems discover both finite and infinite roots, although only finite roots have meaning for engineering purposes. Our method demonstrates how all infinite roots can be avoided in order to obtain substantial computational savings. Infinite roots are avoided by generating random linkage dimensions to construct start-points and start-systems for homotopy continuation paths. The method is benchmarked with a four-bar path synthesis problem.

INTRODUCTION

The kinematic synthesis of linkages often requires finding the finite isolated roots of large polynomial systems. The size of a polynomial system is measured by its number of roots, of which its degree provides an estimated maximum. It is interesting that the kinematic design of relatively simple mechanical systems involves formulating polynomial systems with degrees beyond 100,000. For example, to find all of the four-bar linkages which can pass a coupler point through nine arbitrary points in the plane involves solving a polynomial system with degree 286,720 [1]. Furthermore, more complex mechanical systems seem to result in exponential growth of the degree of their design equations [2, 3].

This growth arises from additional design parameters that increase the number of dimensions in a nonlinear design space. Finding complete solution sets to the polynomial systems which comprise kinematic design equations is a major challenge which tends to demand largescale computations. Many methods have been devised that avoid computing complete solution sets [4], however, the advantage of obtaining complete sets is to provide a full survey of design options that might not be found through methods based in optimization or differential evolution. The state of the art for solving large polynomial systems is polynomial homotopy continuation. Homotopy algorithms compute all roots to general polynomial systems in time proportional to the degree of the system. These methods are quite powerful but still have practical limits based on available computing resources.

Although kinematic polynomial systems tend to have large degrees, they also tend to have sparse monomial structures that indicate the number of finite roots they possess is much smaller than their degree. Returning to the four-bar example, it has been shown in Wampler et al. [1] that 8,652 of 286,720 roots are finite and represent linkage designs. The polynomial degree totals the number of finite roots, infinite roots, and their multiplicities. For engineering purposes, only finite roots are sought. Observations of the monomial structure of a system can often be made in order to intelligently multihomogenize a system and remove many

*Address all correspondence to this author.

roots at infinity leading to a new *multihomogeneous root count*. For smaller systems, this strategy can eliminate all roots at infinity, but for larger systems it is not always clear how the multihomogenization should proceed. Eliminating roots at infinity is important when applying homotopy algorithms because each root requires a path tracking computation. Counting the percentage of infinite roots for the four-bar example, it is estimated that 97% of computational effort is dedicated to discovering roots at infinity, which are filtered out in the end.

In this work, we introduce a new method built on homotopy continuation termed *Finite Root Generation* (*FRG*) that obtains all or most of the finite roots of kinematic polynomial systems while avoiding all computations of roots at infinity. Traditionally, homotopy proceeds with a single start-system and multiple start-points whereas FRG uses multiple start-systems each with one start-point that are constructed from random linkage dimensions so that each start-point is guaranteed to track to a finite root. Since start-points are randomly generated, the possibility exists for tracking to a nonsingular finite root multiple times. This duplication rate is modelled by the *coupon collector problem* from probability theory, and can be used to accurately estimate the number of finite roots without actually computing all of them.

We apply FRG to a numerically general target system that is suitable for constructing parameter homotopies such that an FRG solution set only needs to be computed once for a particular family of polynomial systems. Parameter homotopies provide a means for efficiently solving all subsequent systems belonging to that family. Applying FRG to the benchmark four-bar path generator problem, there were 8,214 unique finite roots found by tracking 25,922 homotopy paths, It is expected that 83,430 homotopy paths must be tracked in order to obtain all roots. This is compared to 286,720 for the best known multihomogeneous homotopy [1] and 152,224 for the best known regeneration homotopy [5].

LITERATURE REVIEW

Continuation methods were pioneered by Roth and Freudenstein [6] who developed the *Bootstrap Method* for solving the nonlinear path synthesis equations of a geared five-bar. They were motivated to provide an alternative to the Newton-Raphson method which required a good initial approximation in order to converge. In their work, the authors indeed constructed continuation start-points from arbitrary starting mechanisms. Since then, the field of numerical algebraic geometry has flourished resulting in sophisticated homotopy continuation algorithms [5, 7–10] that apply to general polynomial systems where start-points are constructed by combinatoric procedures that are advantageously blind to the physical systems represented by the equations. In this work, we return to the schema of constructing start-points by arbitrary mechanisms, but instead of finding one or two mechanism solutions, we aim to find complete root sets of numerically general synthesis systems. These general root sets are suitable for constructing parameter homotopies to efficiently solve later synthesis systems of the same family [11], a concept that appeared 26 years after Roth and Freudenstein [6].

Other solution techniques used to solve polynomial equations include resultant elimination methods, Gröbner bases, and interval analysis. Resultant elimination methods include modifications of Sylvester's dialytic elimination method [12] and involve procedures to transform root finding into a generalized eigenvalue problem. Su and McCarthy [13] show how to use this method to solve a synthesis problem with 64 roots. Masouleh et al. [14] use Gröbner bases combined with resultants to compute 220 roots for the forward kinematics of a spatial parallel manipulator. Gröbner bases transform multivariate root finding into root finding for a sequence of univariate polynomials. Interval analysis takes advantage of interval arithmetic to find roots within a range of the solution space. Lee et al. [15] provide an example of interval analysis applied to the synthesis of a RRR spatial serial chain.

There have been modifications of homotopy methods to eliminate unwanted solutions. Tari et al. [16] appended additional equations to synthesis systems to eliminate degenerate solutions, but their method does not improve computation times. Tsai and Lu [17] devised a procedure for nine point path synthesis of a four-bar where start-points were constructed from five point solutions without attempting to obtain the complete solution set. Perhaps today's most powerful method is called regeneration. Regeneration can greatly reduce the number of homotopy paths to track by solving a system "equation by equation" and discarding singular and infinite roots along the way [5]. However, regeneration still requires many roots at infinity to be tracked. Plecnik and McCarthy [2] used regeneration to approximate a 1.5 million solution set.

In this work, we introduce a method which constructs start-points and start-systems from randomly generated mechanisms in order to avoid tracking homotopy paths to infinite roots. We benchmark FRG on the four-bar path synthesis problem and find that the rate at which finite roots are collected is predicted by the coupon collector problem of probability theory, providing a means for estimating the size of the root set being discovered. Finally, we use the parameter homotopy method to apply the obtained root set to an actual linkage design problem.

DESCRIPTION OF THE METHOD

The Finite Root Generation method generates start-points and start-systems that track to the finite roots of a target-system when implemented with polynomial homotopy continuation. In order to ensure this is the case, a start-system must possess a monomial structure that is no more or no less general than the target system, and a start-point must be a finite root to that start-system. These features are ensured for a single start-point/start-

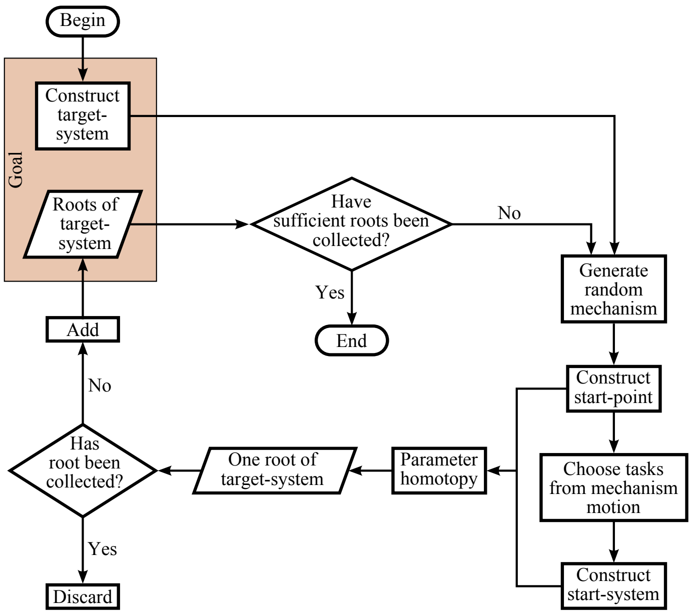

FIGURE 1. A FLOWCHART THAT DESCRIBES THE FINITE ROOT GENERATION METHOD.

system pair by constructing the start-point from a randomly generated linkage, and constructing the start-system from the motion of that linkage. A parameter homotopy then tracks that start-point to a single finite root of the target-system.

In order to find another finite root to the target-system, a new random linkage can be generated so that another start-point/start-system pair is constructed and tracked to (hopefully) another finite root of the target-system. If there are N finite roots to the target-system to be discovered, and there is equal probability of happening upon any root, then the probability of the second iteration happening upon a different finite root from the first is $(N-1)/N$ which for large N are good odds. If the root of the second iteration duplicates the first, then it is discarded, or if it is unique, then it is stored along with the root from the first iteration. The goal is to obtain all or most of the finite roots of the target-system. Iterations continue in this manner as illustrated in Fig. 1 until a sufficient number of roots have been collected. As more and more roots are collected, the possibility of finding another unique root decreases, indicating a situation of diminishing returns. By tracking the rate of success of finding new roots between iterations, the size of the total root set can be estimated.

Estimation Procedure

The decreasing probability of finding new roots is modelled by the coupon collector problem from probability theory [18]. That is, given a set of N unique coupons, how many trials T_N of picking random single coupons are expected in order to obtain all coupons. Coupons are sampled with replacement and all coupons have equal probability of being picked. The problem statement is slightly modified to ask how many trials T_n are expected to obtain n coupons where $n \leq N$. In our case, T_n is the number of homotopy paths and n is the number of finite roots we wish to obtain.

Once $n-1$ roots have been collected, the probability p_n of collecting the n^{th} root is

$$p_n = \frac{N-(n-1)}{N}. \tag{1}$$

Let X_n denote the expected number of trials to take place in the interval between finding $n-1$ roots and n roots. Because X_n is a random variable with a geometric distribution, it is computed as

$$X_n = \frac{1}{p_n}. \tag{2}$$

That means it is expected that N trials will be dedicated solely to finding the final root. By substituting Eqn. (1) into Eqn. (2) and summing X, the expected number of trials to obtain the n^{th} finite root can be expressed as

$$T_n \;=\; \sum_{i=1}^{n} X_i \;=\; \sum_{i=1}^{n} \frac{N}{N-(i-1)}. \tag{3}$$

The variance of T_n is computed as

$$\mathrm{Var}(T_n) = \sum_{i=1}^{n} \frac{N(i-1)}{(N-(i-1))^2}. \tag{4}$$

Eqn. (3) allows us to estimate the expected number of trials in order to obtain a certain number of roots when the total number N is known. For large polynomial systems of interest, N is not known but can be estimated by the rate of success α of finding new roots defined as

$$\alpha = \frac{n}{T_n} = \frac{n}{\sum_{i=1}^{n} \frac{N}{N-(i-1)}} \tag{5}$$

where n is the number of roots collected and T_n is the total number of trials it took to collect those roots. The right side of Eqn. (5) is obtained by substituting in from Eqn. (3).

As the probability of finding new roots decreases, α will tend to decrease too. The aim is next to compute the total number of roots from the success rate by rearranging Eqn. (5). To

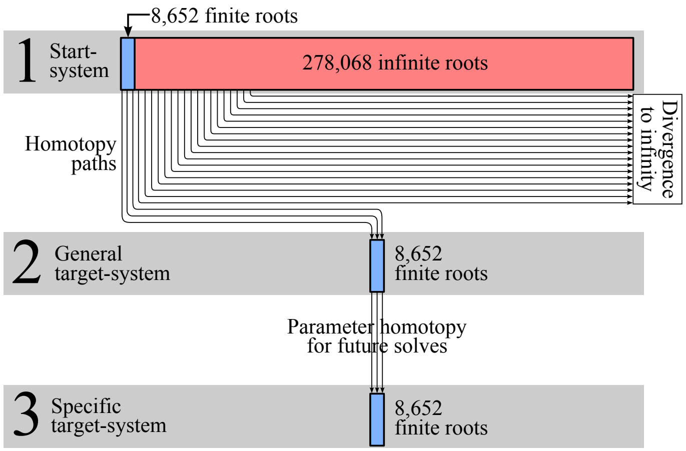

FIGURE 2. A VISUALIZATION OF MULTIHOMOGENEOUS HOMOTOPY.

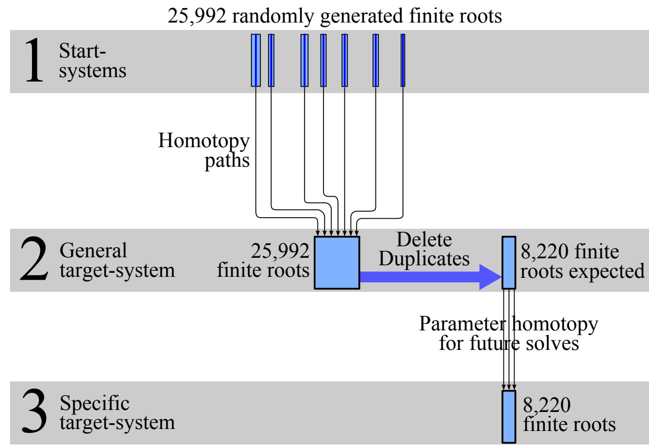

FIGURE 3. A VISUALIZATION OF THE FINITE ROOT GENERATION METHOD TO COLLECT 95% OF ROOTS.

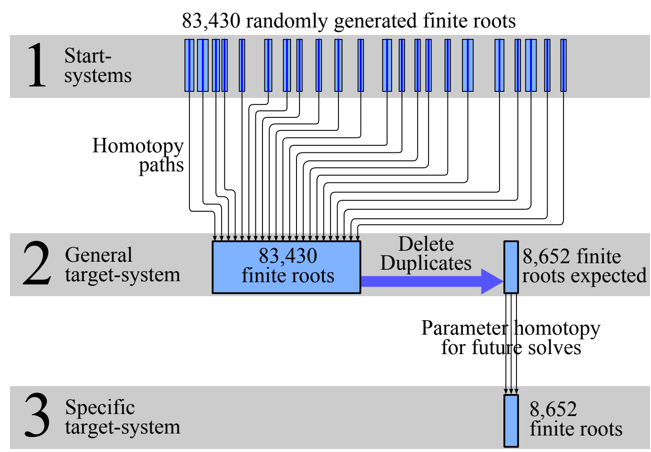

FIGURE 4. A VISUALIZATION OF THE FINITE ROOT GENERATION METHOD TO COLLECT 100% OF ROOTS.

facilitate this computation, α is redefined to replace the summation with an integral,

$$\alpha = \frac{n}{\int_0^n \frac{N}{N-i} di} = \frac{n}{N \ln\left(\frac{N}{N-n}\right)}. \tag{6}$$

Next, we substitute in $\hat{n} = n/N$ as the percentage of finite roots found,

$$\alpha = -\frac{\hat{n}}{\ln(1-\hat{n})}, \tag{7}$$

where Eqn. (7) is independent of N. Now the goal is to solve for the percentage of roots $\hat{n}$ that have been obtained based on the current success rate α. Inversion of Eqn. (7) requires the principle branch of the Lambert function W and takes the form

$$\hat{n} = \alpha W\left(-\frac{1}{\alpha} e^{-\frac{1}{\alpha}}\right) + 1. \tag{8}$$

Eqn. (8) can be used to estimate the percentage of roots obtained at any point in the FRG solution process. Note this equation only provides an estimate of the number of finite roots based on the success rate, unlike Bézout numbers which provide conclusive statements on the upper bound of finite roots. However, FRG estimates might indicate smaller finite root sets, but not with absolute certainty.

BENCHMARK

In order to benchmark the FRG method, it was applied to the nine point path synthesis problem for four-bar linkages. The complete solution to this problem involves finding 8,652 finite roots but the smallest known formulation of this system counts 286,720 roots (including infinite roots). Multihomogeneous homotopy has been applied to solve this problem by tracking 286,720 homotopy paths to discover the 8,652 finite roots [1]. As an improvement, regeneration homotopy finds all finite roots while only tracking 152,224 paths. In this section, it is shown that FRG is able to collect $\sim 95\%$ of finite roots in just 25,922 homotopy paths, and it is computed that 100% is expected to be found in 83,430 homotopy paths. As well, it is noted that multihomogeneous and regeneration homotopies apply to polynomial systems generally while FRG has been specifically devised for large kinematics problems. A visualization of the solution process for a multihomogeneous homotopy is shown in Fig. 2. Visualizations of FRG to obtain 95% and 100% of roots are shown

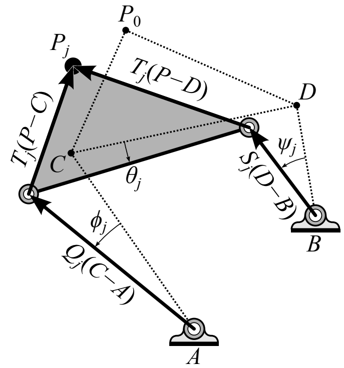

FIGURE 5. A FOUR-BAR PATH GENERATOR DISPLACED TO POSITION j.

in Figs. 3 and 4.

Formulation of Four-bar Path Synthesis

The objective of path generation is to find a four-bar linkage that can move a trace point attached to its coupler link through m points P_j, $j=0,\ldots,m-1$. As shown in Fig. 5, the locations of the four-bar's ground pivots are represented by $A=A_x+A_yi$ and $B=B_x+B_yi$. The locations of the moving pivots in position $j=0$ are represented by $C=C_x+C_yi$ and $D=D_x+D_yi$. The rotations of links AC, BD, and CDP from position 0 to j are measured by ϕ_j, ψ_j, and θ_j, respectively. Using the exponential rotation operators,

$$Q_j=e^{i\phi_j}, \qquad S_j=e^{i\psi_j}, \qquad T_j=e^{i\theta_j}, \tag{9}$$

two loop equations are formulated for each position j,

$$A+Q_j(C-A)+T_j(P_0-C)=P_j, \tag{10}$$
$$B+S_j(D-B)+T_j(P_0-D)=P_j, \quad j=1,\ldots,m-1. \tag{11}$$

As well, the rotation operators must necessarily satisfy

$$Q_j\bar{Q}_j=1, \tag{12}$$
$$S_j\bar{S}_j=1, \tag{13}$$
$$T_j\bar{T}_j=1, \qquad j=1,\ldots,m-1, \tag{14}$$

where the overbar denotes the complex conjugate. Unknowns Q_j are eliminated by solving for them in Eqn. (10) and substituting into Eqn. (12). Similarly, S_j is eliminated by solving Eqn. (11) and substituting into Eqn. (13). These substitutions obtain

$$\begin{bmatrix} a\bar{b}_j & \bar{a}b_j \\ c\bar{d}_j & \bar{c}d_j \end{bmatrix} \begin{Bmatrix} T_j \\ \bar{T}_j \end{Bmatrix} = \begin{Bmatrix} f\bar{f}-a\bar{a}-b_j\bar{b}_j \\ g\bar{g}-c\bar{c}-d_j\bar{d}_j \end{Bmatrix}, \tag{15}$$

where

$$\begin{aligned} a&=P_0-C, & b_j&=A-P_j, & f&=C-A, \\ c&=P_0-D, & d_j&=B-P_j, & g&=D-B. \end{aligned} \tag{16}$$

Finally, solving Eqn. (15) for $(T_j,\bar{T}_j)$, substituting into Eqn. (14), clearing the denominator, and factoring, we obtain polynomial system $\mathcal{T}$,

$$\mathcal{T}: \quad \mathbf{a}^T\bar{\mathbf{b}}_j\mathbf{b}_j^T\bar{\mathbf{a}}-\mathbf{c}^T\bar{\mathbf{d}}_j\mathbf{d}_j^T\bar{\mathbf{c}}=0, \qquad j=1,\ldots,8, \tag{17}$$

where

$$\mathbf{a}=\begin{Bmatrix} a(g\bar{g}-c\bar{c}) \\ c(f\bar{f}-a\bar{a}) \\ a \\ c \end{Bmatrix}, \quad \mathbf{b}_j=\begin{Bmatrix} b_j \\ -d_j \\ -b_jd_j\bar{d}_j \\ b_j\bar{b}_jd_j \end{Bmatrix},$$

$$\mathbf{c}=\begin{Bmatrix} a\bar{c} \\ \bar{a}c \end{Bmatrix}, \quad \mathbf{d}_j=\begin{Bmatrix} b_j\bar{d}_j \\ -\bar{b}_jd_j \end{Bmatrix}. \tag{18}$$

Eqn. (17) contains the eight unknowns $\{A,B,C,D,\bar{A},\bar{B},\bar{C},\bar{D}\}$ and represents a square system of eight equations for the case $m=9$. This is the benchmark system and is equivalent to formulations for the same problem given in [1] and [5].

Methods

FRG begins by creating a target-system for which it is intended to find all roots. The target system is constructed to be numerically general rather than represent a specific synthesis task. The reason for this is that the target-system is intended to be solved once where thereafter the target-system and its roots can be used to construct parameter homotopies that efficiently solve for specific synthesis systems. For numerical experiments, the target-system was constructed by generating 18 random complex numbers within the box defined by corners $(-1-i,\ 1+i)$, assigning these numbers to P_j and $\bar{P}_j$, $j=0,\ldots,8$, and substituting into $\mathcal{T}$.

$$
\begin{aligned}
P_0 &= 0.776874 - 0.642684i, & \bar{P}_0 &= 0.873111 + 0.292468i,\\
P_1 &= 0.549479 + 0.418241i, & \bar{P}_1 &= 0.689034 - 0.944901i,\\
P_2 &= -0.261986 - 0.403618i, & \bar{P}_2 &= 0.854109 - 0.482044i,\\
P_3 &= 0.767234 - 0.378801i, & \bar{P}_3 &= -0.268805 - 0.865098i,\\
P_4 &= -0.068090 + 0.919907i, & \bar{P}_4 &= -0.263339 - 0.451987i,\\
P_5 &= 0.055654 + 0.675009i, & \bar{P}_5 &= -0.221326 + 0.775947i,\\
P_6 &= -0.134960 - 0.424099i, & \bar{P}_6 &= 0.165736 + 0.319065i,\\
P_7 &= -0.239858 + 0.041043i, & \bar{P}_7 &= 0.248468 + 0.077381i,\\
P_8 &= 0.635501 - 0.474101i, & \bar{P}_8 &= 0.342355 - 0.501518i.
\end{aligned}
\tag{19}
$$

FRG requires the construction of start-points and start-systems from randomly generated mechanisms which is accomplished by generating five random complex numbers within the box defined by $(-0.5 - 0.5i,\ 0.5 + 0.5i)$. Four of these numbers define a start-point

$$s_p = \{A, B, C, D, \bar{A}, \bar{B}, \bar{C}, \bar{D}\}, \tag{20}$$

and the fifth is assigned to P_0. These five numbers define a four-bar from which eight configurations are selected with disregard to whether they belong to the same circuit. These eight configurations define points $(P_j, \bar{P}_j)$, $j = 1, \ldots, 8$ which along with $(P_0, \bar{P}_0)$ are substituted into $\mathcal{T}$ to obtain a start-system that corresponds to s_p.

Path tracking was completed using the algorithm within the software BERTINI [19]. This path tracking algorithm utilizes random numbers to form projective patches and defines homotopies based on a random parameter γ (Ref. [5], pp. 25). Occasionally, BERTINI would experience a path tracking failure, e.g. reaching a minimum step size. In these cases, an FRG iteration would be abandoned and not counted valid FRG trial. Table 1 shows a few start-points and start-system parameters used to construct solutions to the target-system defined the parameters of Eqn. (19).

RESULTS

The FRG method was applied to find the roots of a numerically general version of $\mathcal{T}$. From Eqns. (3) and (4), it is computed that in order to find $\hat{n} = 95\%$ of the roots of $\mathcal{T}$, the expectation is to track 25,922 paths with a standard deviation of 372 paths. The expected number of paths for other values of $\hat{n}$ can be found in Fig. 6, namely, if $\hat{n} = 100\%$ of roots are to be obtained, the expectation is to track 83,430 paths with a standard deviation of 11,092 paths. An attempt to find 100% of roots was not made.

FRG was applied only to the case of finding 95% of finite roots where the algorithm was run for 25,922 iterations which

TABLE 1. START-POINTS AND START-SYSTEM PARAMETERS USED TO TRACK HOMOTOPIES TO SOLUTIONS OF THE TARGET SYSTEM DEFINED BY PARAMETERS SHOWN IN EQN. (19). HOMOTOPIES FOR THE SOLUTIONS BELOW WERE CONSTRUCTED WITH $\gamma = 0.573937 + 1.472072i$.

	Solution 1	**Solution 2**	**Solution 3**
Start-points			
A	$-0.185931-0.490157i$	$0.226527-0.447966i$	$0.296386-0.339809i$
B	$-0.441407-0.138234i$	$0.416623-0.143895i$	$0.037220-0.462856i$
C	$0.064826+0.359267i$	$-0.023306-0.024924i$	$-0.476241-0.281314i$
D	$-0.477876+0.440497i$	$-0.344134-0.420386i$	$-0.044317+0.028224i$
$\bar{A}$	$-0.185931+0.490157i$	$0.226527+0.447966i$	$0.296386+0.339809i$
$\bar{B}$	$-0.441407+0.138234i$	$0.416623+0.143895i$	$0.037220+0.462856i$
$\bar{C}$	$0.064826-0.359267i$	$-0.023306+0.024924i$	$-0.476241+0.281314i$
$\bar{D}$	$-0.477876-0.440497i$	$-0.344134+0.420386i$	$-0.044317-0.028224i$
Start-system parameters			
P_0	$-0.207946+0.369011i$	$0.384005+0.244221i$	$-0.155698-0.422040i$
P_1	$-0.115369+0.351828i$	$-0.321480-0.783846i$	$0.273905-1.211253i$
P_2	$0.393717-0.190376i$	$-0.042075-1.265047i$	$-0.064827-1.195472i$
P_3	$-0.312109+0.319537i$	$-0.524641-0.489862i$	$-0.086717+0.118875i$
P_4	$-0.608101-0.973176i$	$1.141800-0.508283i$	$-0.025993-0.630360i$
P_5	$-1.065637-0.419560i$	$-0.329956+0.122805i$	$0.404738-0.750646i$
P_6	$-0.754964-0.722653i$	$-0.152737-0.865539i$	$-0.736730-0.262438i$
P_7	$-1.022605-0.077858i$	$0.224757-0.522349i$	$-0.100214-0.501486i$
P_8	$-0.778031-0.906804i$	$0.152844-0.728661i$	$-0.340800+0.386734i$
$\bar{P}_0$	$-0.207946-0.369011i$	$0.384005-0.244221i$	$-0.155698+0.422040i$
$\bar{P}_1$	$-0.115369-0.351828i$	$-0.321480+0.783846i$	$0.273905+1.211253i$
$\bar{P}_2$	$0.393717+0.190376i$	$-0.042075+1.265047i$	$-0.064827+1.195472i$
$\bar{P}_3$	$-0.312109-0.319537i$	$-0.524641+0.489862i$	$-0.086717-0.118875i$
$\bar{P}_4$	$-0.608101+0.973176i$	$1.141800+0.508283i$	$-0.025993+0.630360i$
$\bar{P}_5$	$-1.065637+0.419560i$	$-0.329956-0.122805i$	$0.404738+0.750646i$
$\bar{P}_6$	$-0.754964+0.722653i$	$-0.152737+0.865539i$	$-0.736730+0.262438i$
$\bar{P}_7$	$-1.022605+0.077858i$	$0.224757+0.522349i$	$-0.100214+0.501486i$
$\bar{P}_8$	$-0.778031+0.906804i$	$0.152844+0.728661i$	$-0.340800-0.386734i$
Solutions obtained			
A	$1.208688-0.064110i$	$-0.278014-3.903201i$	$1.363042+0.327298i$
B	$0.229275+0.728512i$	$-2.851690-4.028216i$	$-0.236355-0.424120i$
C	$1.149090-0.376585i$	$-0.356765-3.870173i$	$0.744465-0.469276i$
D	$0.724958-0.465222i$	$-2.452992-3.479546i$	$0.939745-0.691008i$
$\bar{A}$	$0.322662-0.323042i$	$0.942263-5.099374i$	$0.026897-0.657391i$
$\bar{B}$	$-0.968032+1.383404i$	$-5.662954-2.367803i$	$-0.079412-0.799504i$
$\bar{C}$	$0.512174+0.503870i$	$2.411689-5.258779i$	$0.931018+0.261914i$
$\bar{D}$	$-1.469395+0.697904i$	$-5.912599-3.042769i$	$-1.178260-0.792065i$

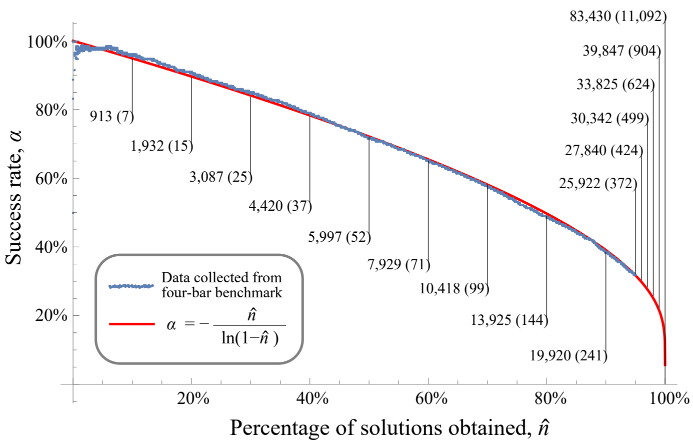

FIGURE 6. THE SUCCESS RATE α PLOTTED AGAINST THE PERCENTAGE OF SOLUTIONS OBTAINED $\hat{n}$ FOR BOTH THE NUMERICAL EXPERIMENT AND THEORETICAL CURVE. LABELS INDICATE THE EXPECTED NUMBER OF PATHS TO TRACK FOR VARIOUS $\hat{n}$ WITH STANDARD DEVIATIONS IN PARENTHESIS.

found 8,214 unique finite roots (94.94%). For this numerical experiment, the success rate α was tracked as roots were collected and the curve is plotted in Fig. 6 alongside the theoretical curve of Eqn. (7). At the 25,922$^{\text{nd}}$ iteration, the success rate was $\alpha = 31.69\%$, which when substituted into Eqn. (8) estimates 95.01% of roots have been collected, or rather that the total size of the finite root set is 8,645. This estimation is in comparison to the known size of 8,652. The estimation procedure should be useful for large root sets of unknown size.

EXAMPLE

The motivation behind the Finite Root Generation method is to obtain solution sets to large polynomial systems for kinematic design. In this paper, the method is introduced and only applied to four-bar path synthesis for benchmarking. In order to bridge FRG with tangible kinematic design, this section presents an example. The objective is to design a linkage that traces a D-shaped path potentially for a walking mechanism.

To begin, nine points were selected,

$$\begin{aligned} &P_0 = 0 - 0.0625i, && P_3 = 0.95 + 0.2i, && P_6 = -0.95 + 0.2i,\\ &P_1 = 0.5 - 0.05i, && P_4 = 0.35 + 0.4i, && P_7 = -1 + 0i,\\ &P_2 = 1 - 0i, && P_5 = -0.35 + 0.4i, && P_8 = -0.5 - 0.05i. \end{aligned} \quad (21)$$

Note that the straight portion of the "D" was curved (Figs. 7 and 8) in the hope that this would result in four-bars with smaller link lengths. Substituting Eqn. (21) into $\mathcal{T}$ forms a synthesis system which was solved by a parameter homotopy constructed from the 8,214 roots found in the previous section. The results include 493 roots that represent physical link geometry of which 13 were found to be free of branch and circuit defects. Note that a finite root is necessary but not sufficient for a solution to represent physical geometry. In other words, the number of finite roots provides a bound for useful engineering solutions.

The odd number of linkage results found is problematic as the synthesis equations are symmetric meaning that any root with values $\{A, B, C, D\}$ should have a corresponding root where values A and B are swapped and values C and D are swapped. This odd number of roots is an artifact of working with an incomplete root set. A second less obvious symmetry is that all roots should exist in cognate sets of three [20]. Combined with the above symmetry, roots can be grouped in sets of six where if one member is known the other five can be generated. With this in mind, the example problem could be reduced to collecting $N = 1,442$ root subsets, but this would be less useful for benchmarking FRG with existing results that were not computed in that manner.

Examples of defect-free and defective linkages are given in Figs. 7 and 8. The linkage shown in Fig. 7(a) has the potential to be used as a walking mechanism although its large link lengths are not ideal. The linkages shown in Fig. 8 are similar to Chebyshev cognate linkages which are known to trace a similar "D". A third Chebyshev-like linkage was found as well which is the horizontal mirror of Fig. 8(a). The linkage shown in Fig. 8(b) is branch-defective meaning it cannot be actuated from a base joint. Linkage designs were analyzed for branch and circuit defects by computing mechanism singularities and ranges of motion. This analysis did not consider order defects. The lack of strong design candidates from the example's results motivates a search into more complex linkage topologies, which is out of the scope of this paper.

CONCLUSION

In this paper, a new technique was introduced for solving large polynomial systems for the kinematic design of linkages. The technique is termed Finite Root Generation and provides a means of generating homotopy continuation start-points that track to the finite roots of a numerically general target-system. It is shown that the rate at which FRG is able to find unique roots is modelled closely by the coupon collector problem. In order to validate the advantage of FRG, it was applied to the benchmark problem of nine point path synthesis for a four-bar linkage. The benchmark obtained 94.94% of finite roots in 25,922 homotopy paths and it is expected that 100% could be obtained in 83,430 paths. The best known regeneration homotopy finds all roots in 152,224 paths.

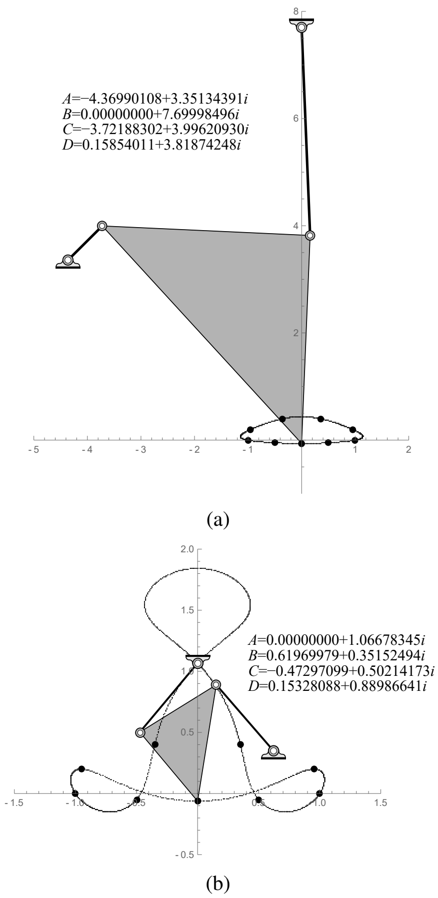

FIGURE 7. (a) A DEFECT-FREE MECHANISM AND (b) A MECHANISM WITH A BRANCH DEFECT.

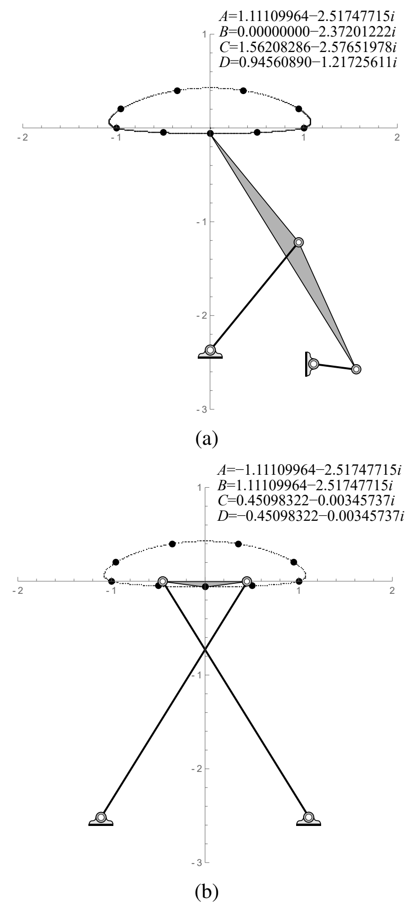

FIGURE 8. SYNTHESIS RESULTS THAT RESEMBLE CHEBYSHEV LINKAGES. (a) A DEFECT-FREE MECHANISM AND (b) A MECHANISM WITH A BRANCH DEFECT.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of National Science Foundation award CMMI-1549667. 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] C. W. Wampler, A. J. Sommese, and A. P. Morgan, 1992. "Complete Solution of the Nine-Point Path Synthesis Problem for Four-Bar Linkages," *Journal of Mechanical Design*, 114(1):153–159.

[2] M. Plecnik and J. M. McCarthy, 2015. "Computational Design of Stephenson II Six-bar Function Generators for 11 Accuracy Points," *Journal of Mechanisms and Robotics*, JMR-15-1055, 8(1):011017-011017-9.

[3] M. Plecnik and J. M. McCarthy, 2015. "Kinematic Synthesis of Stephenson III Six-bar Function Generators," *Mechanism and Machine Theory*, 97:112–126.

[4] R. R. Root and K. M. Ragsdell, 1976. "A Survey of Optimization Methods Applied to the Design of Mechanisms," *Journal of Engineering for Industry*, 98(3):1036–1041.

[5] D. J. Bates, J. D. Hauenstein, A. J. Sommese, and C. W. Wampler, 2013. *Numerically Solving Polynomial Systems with Bertini*, SIAM Press, Philadelphia.

[6] B. Roth and F. Freudenstein, 1963. "Synthesis of Path-Generating Mechanisms by Numerical Methods," *Journal of Engineering for Industry*, 85(3):298-304.

[7] S. M. Wise, A. J. Sommese, and L. T. Watson, 2000. "Algorithm 801: POLSYS_PLP—A Partitioned Linear Product Homotopy Code for Solving Polynomial Systems of Equations," *ACM Transactions on Mathematical Software*, 26(1):176–200.

[8] H. J. Su, J. M. McCarthy, M. Sosonkina, and L. T. Watson, 2006. "Algorithm 857: POLSYS_GLP—A Parallel General Linear Product Homotopy Code for Solving Polynomial Systems of Equations," *ACM Transactions on Mathe-*

matical Software, 32(4):561–579.

[9] J. Verschelde, 2011. "Polynomial Homotopy Continuation With PHCpack," *ACM Communications in Computer Algebra*, 44(3/4):217–220.

[10] T. Chen, T. L. Lee, and T. Y. Li, 2014. "Hom4PS-3: A Parallel Numerical Solver for Systems of Polynomial Equations Based on Polyhedral Homotopy Continuation Methods," *Proceedings of the 4th International Conference on Mathematical Software*, pp. 183–190, August 5–9, 2014, Seoul, South Korea.

[11] T. Y. Li, T. Sauer, and J. A. Yorke, 1989. "The Cheater's Homotopy: An Efficient Procedure for Solving Systems of Polynomial Equations," *SIAM Journal on Numerical Analysis*, 26(5):1241–1251.

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

[13] H. Su and J. M. McCarthy, 2005. "The Synthesis of an RPS Serial Chain to Reach a Given Set of Task Positions," *Mechanism and Machine Theory*, 40(7):757–775.

[14] M. T. Masouleh, C. Gosselin, M. Husty, and D.R. Walter, 2011. "Forward Kinematic Problem of 5-RPUR Parallel Mechanisms (3T2R) With Identical Limb Structures," *Mechanism and Machine Theory*, 46(7):945–959.

[15] E. Lee, C. Mavroidis, and J. P. Merlet, 2002. "Five Precision Points Synthesis of Spatial RRR Manipulators Using Interval Analysis," *Proceedings of the ASME 2002 IDETC/CIE Conference*, Paper No. DETC2002/MECH-34272, September 29–October 2, 2002, Montreal, Canada.

[16] H. Tari, H. Su, and T. Li, 2010. "A Constrained Homotopy Technique for Excluding Unwanted Solutions from Polynomial Equations Arising in Kinematics Problems," *Mechanism and Machine Theory*, 45(6):898–910.

[17] L. W. Tsai and J. J. Lu, 1990. "Coupler-Point-Curve Synthesis Using Homotopy Methods," *Journal of Mechanical Design*, 112(3):384–389.

[18] R. Isaac, 1995. "Baseball Cards, The Law of Large Numbers, and Bad News for Gamblers," *The Pleasures of Probability*, Springer, New York, pp. 77–93.

[19] D. J. Bates, J. D. Hauenstein, A. J. Sommese, and C. W. Wampler. Bertini: Software for Numerical Algebraic Geometry. Available at bertini.nd.edu with permanent doi: dx.doi.org/10.7274/R0H41PB5.

[20] S. Roberts, 1875. Three-Bar Motion in Plane Space. *Proceedings of the London Mathematical Society*, s1-7(1):14–23.