Multi-modal Continuous Approximate Synthesis of Planar Four-bar Function Generators

Zachary A. Copeland and M. John D. Hayes*

Department Mechanical and Aerospace Engineering,
Carleton University,
Ottawa, ON, Canada
E-mail: zackcopeland@cmail.carleton.ca
E-mail: johnhayes@cunet.carleton.ca
*Corresponding author

Abstract: This paper introduces a novel multi-modal continuous approximate synthesis algorithm for planar four-bar function generators. Multi-modal in this sense means concurrently synthesising multiple functions between different joint variable parameter pairs in a four-bar linkage over the desired continuous input-output ranges between different pairs of variables. These are not competing functions, rather perturbed functions. The continuous multi-modal synthesis equation is the sum of the squared input-output equations integrated over the different prescribed input variable parameter ranges. Every planar four-bar mechanism explicitly generates six distinct functions each uniquely determined by one set of link parameter constants. We will examine the simultaneous continuous approximate synthesis of two related perturbed functions between different pairs of joint variables that, in general, require different link constants to generate. The optimisation involves identifying the best *compromise* link constants to approximately generate the two prescribed functions. Planar 4R and RRRP examples are presented where two different functions, one primary and the other perturbed secondary, are generated over continuous ranges between the specified input variable parameter and the associated output variable parameter. We evaluate the continuous multi-modal synthesis results by comparing the areas between the generated degree 4 planar algebraic curves in the parameter planes of the input and output joint variables to those of the prescribed input-output functions over their continuous ranges, thereby simultaneously evaluating a measure of the Euclidean norm of both the design and structural errors. The work presented herein is preliminary, and intended only as a proof-of-concept.

Keywords: Planar four-bar mechanisms; algebraic input-output (IO) equations; continuous approximate synthesis; multi-modal continuous approximate synthesis.

Reference to this paper should be made as follows: Copeland, Z.A. and Hayes, M.J.D. (2023) 'Multi-modal Continuous Approximate Synthesis of Planar Four-bar Function Generators', *Int. J. Mechanisms and Robotic Systems*, Vol. 5, No. 3, pp. 246-269.

Biographical notes: Zachary A. Copeland received his BEng (2015), and M.A.Sc (2017) in Mechanical Engineering from Carleton University, Ottawa, ON, Canada. He is currently a PhD candidate in Mechanical Engineering developing continuous approximate synthesis theory and techniques at Carleton. His interests also include manufacturing and production technologies, as well as the theory of machines and mechanisms. He enjoys manufacturing and implementation, as well as software design.

M. John D. Hayes received his BEng (1995), MEng (1996), and PhD (1999) degrees all from McGill University in Montreal, QC, Canada. He is currently a Full Professor in the Department of Mechanical and Aerospace Engineering at Carleton University in Ottawa, ON, Canada. His research interests include computational kinematics, algebraic geometry, and optimisation as they apply to mechanical systems design, analysis, calibration, and simulation.

1 Introduction

The study of planar four-bar linkages involves a large variety of problems: guiding a point along a specific curve or path, known as coupler curve or path generation; guiding a rigid body through a sequence of positions and orientations, known as the Burmester, or rigid body guidance problem [1]; guiding a rigid body along a time-dependent sequence of positions and orientations, usually called trajectory generation [2]; problems concerning the transmission of forces and torques [3]; or designing an optimally balanced linkage [4]. An additional important subset of this gamut is the function generation problem [5]. It consists of identifying a mechanism which is able to approximate, in some sense, a mathematical function between an input and output (IO) pair of joint variables for a given planar linkage kinematic architecture comprising RR-, RP-, PR-, or PP-dyads. In this context R and P indicate revolute and prismatic joints connecting a pair of rigid links, also known as R- and P-pairs. A great number of examples of planar function generator synthesis, addressing many different issues, are to be found in the literature, see [6, 7, 8, 9, 10, 11, 12] for a small sample.

The function generation problem is the focus of this paper. All movable mechanical systems generate functions between all of the joint variables as they move. Figure 1 illustrates one such planar four-bar 4R linkage where the reference coordinate systems, length and angle parameters have been assigned strictly according to the Denavit-Hartenberg (DH) parameter assignment rules [1, 13]. If links a_1 and a_3 are the input and output links, respectively, the IO function is specified as $\theta_4 = f(\theta_1)$. Once the four a_i link lengths are identified which approximately generate the prescribed function, the corresponding mechanism generates five additional functions, one for each of the five other distinct angle pairings $\theta_j = f(\theta_i)$. All six of these functions are determined by the identified values for the link lengths that approximately generate the lone prescribed function.

Exact synthesis results in a linkage that precisely generates the prescribed function, but only for the three precision IO pairs that generate the desired output for three prescribed input parameters creating a set of three synthesis equations in three unknown link length ratios. Whereas, approximate synthesis uses $n > 3$ precision IO pairs to create an overconstrained set of synthesis equations leading to a linkage that generates the desired function, in general, but only approximately over the desired displacement range due to errors induced by the synthesis. *Design* and *structural* errors [1] are important performance indicators used in the assessment and optimisation of mechanical systems intended as function generating linkages designed by means of approximate synthesis. The design error is the residual of the identified linkage in satisfying the synthesis equations [5], and is evaluated at each of $n > 3$ precision points, or poses, in a discrete set satisfying the prescribed function. Minimising the Euclidean norm of the design error leads to a linear least-squares problem. The structural error, on the other hand, is defined as the difference between the prescribed output angle, and

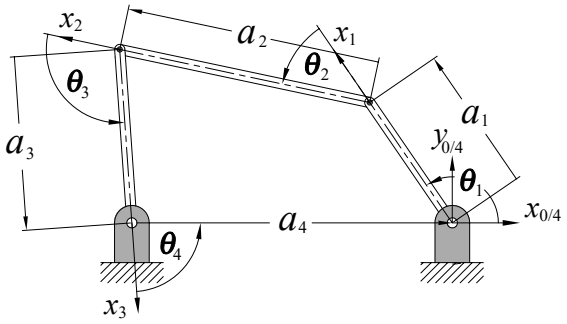


Figure 1: Planar 4R linkage with Denavit-Hartenberg coordinate systems and parameter assignments.

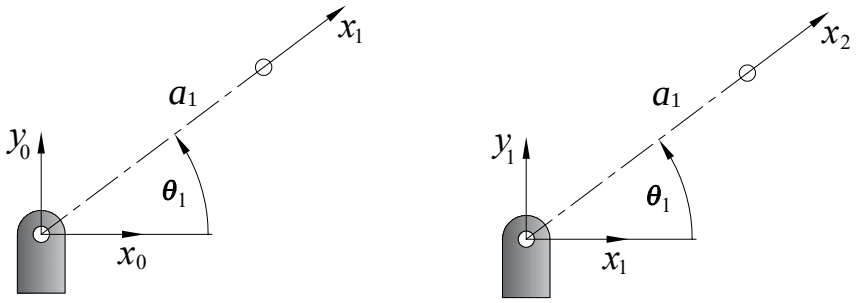
the output angle that is generated by the linkage at each precision point [8]. This problem is typically solved by minimising the norm of the array of the structural error evaluated at each precision point using some form of Gauss-Newton non-linear minimisation [14], requiring an iterative solution procedure that terminates when a desired minimum norm threshold is obtained. The structural error is arguably the metric that truly matters since it is directly related to the physical performance of the linkage.

It was observed in [15] that as the cardinality of the data set used to compute the design error minimising linkage becomes large, on the order of $n \geq 40$, the design error minimising linkage tends to converge to the structural error minimising linkage. Hence, one may avoid the non-linear structural error computation provided a sufficient number of precision points are specified. The natural question is then "how large must n be?" The obvious response is to extend the cardinality of the data set used to compute the design error minimising linkage to infinity by way of integration. Unfortunately, while it was demonstrated in [16] that this extension is possible through the integration of the trigonometric Freudenstein equation, the generalisation of the process is computationally prohibitive and any advantage obtained through the elimination of the need for an explicit solution to the non-linear structural error problem is lost to the numerical complexity of the integration. A less cumbersome continuous approximate synthesis method was needed, and was realised with the algebraic IO equations [17, 18, 19]. These algebraic IO equations were subsequently used for planar RRRP and PRRP function generator synthesis problems in [18], and used to extend the observations made in [15] to develop the algebraic *continuous approximate synthesis* technique reported in [20], which will be relied upon to develop a preliminary form of multi-modal continuous approximate synthesis reported in this paper.

2 The Denavit-Hartenberg Convention and the Planar 4R IO Equation

Before discussing the continuous approximate synthesis approach, it will be useful to recall the matrix method for kinematic analysis and synthesis of linkages, which we call the DH method, developed by Jacques Denavit and Richard Hartenberg and first published in 1955 [13], and subsequently in their textbook on kinematic synthesis [1] in 1964. The first step in the DH method applied to an arbitrary kinematic chain requires the identification and numbering of all joint axes.

In order to align with our previous publications we will be using the same, slightly modified DH parameter coordinate system naming convention in the interest of consistency. The slight modification is that we have as the relatively stationary coordinate system at the start of a kinematic chain the $x_0,\ y_0,\ z_0$ coordinate system. Hence, at any joint in the kinematic chain we measure, for example, the relative angle θ_i of link a_i about joint axis z_{i-1} from x_{i-1} to x_i, see Figure 2a. Whereas, in the original paper [13] the relatively stationary coordinate system at the start of a kinematic chain is the $x_1,\ y_1,\ z_1$ coordinate system, and the relative angle θ_i of link a_i is measured about z_i from x_i to x_{i+1}, see Figure 2b. Because the modification is so subtle, and because the mechanical engineering world has moved away from the original assignment rules found in [1, 13], we will henceforth refer to our version of the parametrisation simply as the DH method.



(a) Slightly modified definition of θ_1.

(b) Original definition of θ_1.

Figure 2: Enumeration of the DH coordinate systems and assignment rules.

The DH parametrisation involves the allocation of coordinate systems to each link in the chain that move with the link, using a set of rules to locate the origin of the coordinate system and the orientation of the basis vectors. The position and orientation of consecutive links is defined by a homogeneous transformation matrix which maps coordinates of points in the coordinate system attached to link i to those of the same points described in a coordinate system attached to link $i-1$.

To visualise the four DH parameters consider two sequential arbitrary neighbouring links, $i-1$ and i. Two such links are illustrated, together with their DH parameters, in Figure 3. The DH parameters [13] are defined in the following way with our subtle modification.

θ_i, joint angle: the angle from x_{i-1} to x_i measured about z_{i-1}.

d_i, link offset: the distance from x_{i-1} to x_i measured along z_{i-1}.

τ_i, link twist: the angle from z_{i-1} to z_i measured about x_i.

a_i, link length: the directed distance from z_{i-1} to z_i measured along x_i.

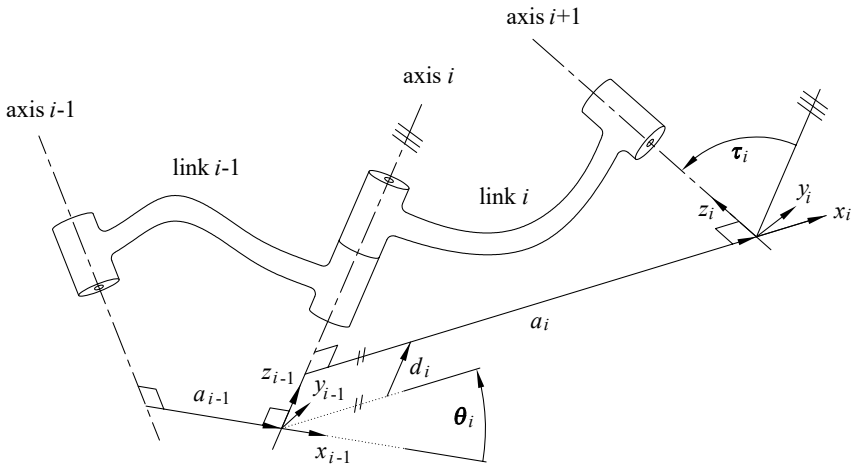


Figure 3: DH parameters in a general serial 3R kinematic chain.

The DH coordinate transformation matrix, using the European convention for homogeneous coordinate arrays $[w,\ x,\ y,\ z]^T$, where w is the homogenising coordinate, is

$$ {}^{i-1}\mathbf{T}_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ a_i \cos\theta_i & \cos\theta_i & -\sin\theta_i \cos\tau_i & \sin\theta_i \sin\tau_i \\ a_i \sin\theta_i & \sin\theta_i & \cos\theta_i \cos\tau_i & -\cos\theta_i \sin\tau_i \\ d_i & 0 & \sin\tau_i & \cos\tau_i \end{bmatrix}. \tag{1} $$

We then algebraise Equation (1) using the tangent half-angle substitutions for the joint and twist angles where

$$ v_i = \tan\left(\frac{\theta_i}{2}\right), \ \Rightarrow \cos\theta_i = \frac{1-v_i^2}{1+v_i^2}, \quad \sin\theta_i = \frac{2v_i}{1+v_i^2}, $$

$$ \alpha_i = \tan\left(\frac{\tau_i}{2}\right), \ \Rightarrow \cos\tau_i = \frac{1-\alpha_i^2}{1+\alpha_i^2}, \quad \sin\tau_i = \frac{2\alpha_i}{1+\alpha_i^2}. $$

The detailed computations leading to the results presented herein use the Maple library MyKinematics [21] which requires the European homogeneous coordinate convention. Regardless, those details remain unseen in this paper, but will be provided to the interested reader upon request.

The forward and inverse kinematics of serial chains are the concatenations of the individual transformation matrices in the appropriate order [22]. For example the forward kinematics problem of determining the position and orientation of the n^{th} link in a serial kinematic chain described in a relatively fixed non-moving base coordinate system 0, given the relevant DH parameters and values for the n joint variables becomes conceptually simple matrix multiplication.

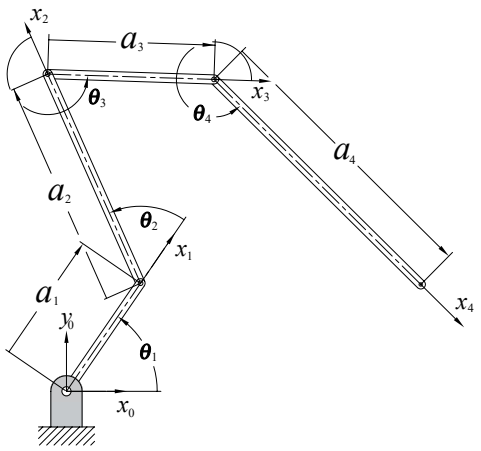


Figure 4: DH parameters assigned to a serial planar 4R linkage.

The DH method was largely intended for planar, spherical, and spatial four-bar simple closed kinematic chains, but has since become nearly universally applied and synonymous with the kinematics of mechanical systems in general, and robot mechanical systems in particular, see [23, 24, 25, 26] for instance, but there are many other modern examples. The serial nR chain is conceptually closed by equating the forward kinematics transformation matrix to the identity.

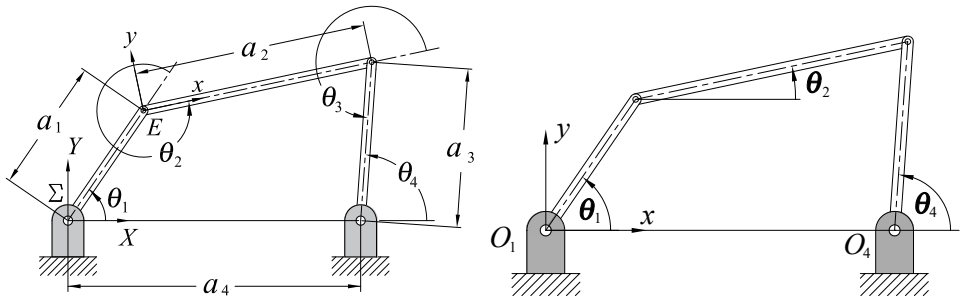
$$ {}^{0}\mathbf{T}_{n} = {}^{0}\mathbf{T}_{1}\,{}^{1}\mathbf{T}_{2}\,{}^{2}\mathbf{T}_{3}\cdots{}^{n-2}\mathbf{T}_{n-1}\,{}^{n-1}\mathbf{T}_{n} \;=\; \mathbf{I}. \tag{2} $$

The resulting matrix represents a set of implicit equations in terms of the link constants and all n joint angles. If we restrict ourselves to the planar 4R simple closed kinematic chain, and the IO equation that relates θ_4 to θ_1 then the intermediate angles θ_2 and θ_3 must be eliminated using the available equations. What remains is a single implicit equation in θ_4 and θ_1.

2.1 The Planar 4R Algebraic IO Equations

Consider the serial 4R kinematic chain illustrated in Figure 4. Equating the forward kinematics to the identity creates the kinematic closure equation. This closure means that the serial kinematic chain becomes a single loop parallel chain and the fourth link described by a_4 becomes stationary while links a_1, a_2, and a_3 move with a single degree of freedom relative motion. The serial chain can be closed such that the axis numbers circulate either clockwise (CW) or counter clockwise (CCW). The CCW circulation will be used herein.The CCW circulation found in [13], and illustrated in Figure 1, does not lead to the results 21^{st}-century mechanical engineers have come to expect. Rather, the convention has evolved away from the DH relative angles and reverts to Freudenstein's absolute angular measures, often combined with relative measures as seen in Figure 5a. However, the standard circulation found in nearly every text on the subject not penned by Denavit or Hartenberg is CW and absolute, see [27, 28, 29] for example. The absolute angle measures are not compatible with the original DH method.

The detailed derivation of the planar algebraic IO equations for function generators are to be found in [18, 19], but will be briefly summarised here. Using the coordinate



(a) Hybrid CW angle measures. (b) Absolute CW angle measures.

Figure 5: Planar 4R with hybrid absolute-relative and absolute angle measures.

system assignments and DH parameters in Figure 1, the DH transformation matrix ${}^0\mathbf{T}_4$ is computed. This homogeneous transformation matrix is then mapped to the eight Study soma coordinates [30, 31] and the corresponding 8×1 Study array is equated to the identity array $[1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0]^T$. This results in eight equations in the DH parameter constants and the four $v_1,\ v_2,\ v_3,\ v_4$ variable angle parameters. The first equation is the trivial normalising condition, meaning that there are seven useful equations to work with. However, because we are concerned only with displacements all in the same plane, four of the soma coordinates identically vanish, leaving only three useful equations, see [17] for details. To obtain the v_1-v_4 algebraic IO equation the v_2 and v_3 intermediate joint angle parameters are eliminated using the three non-trivial equations and the Gröbner basis elimination monomial term ordering called "`lexdeg`" in Maple 2021, revealing:

$$Av_1^2v_4^2 + Bv_1^2 + Cv_4^2 - 8a_1a_3v_1v_4 + D = 0, \tag{3}$$

where,

$$\begin{aligned}
A &= A_1A_2 = (a_1 - a_2 + a_3 - a_4)(a_1 + a_2 + a_3 - a_4),\\
B &= B_1B_2 = (a_1 + a_2 - a_3 - a_4)(a_1 - a_2 - a_3 - a_4),\\
C &= C_1C_2 = (a_1 - a_2 - a_3 + a_4)(a_1 + a_2 - a_3 + a_4),\\
D &= D_1D_2 = (a_1 + a_2 + a_3 + a_4)(a_1 - a_2 + a_3 + a_4),\\
v_1 &= \tan\frac{\theta_1}{2}, \quad v_4 = \tan\frac{\theta_4}{2}.
\end{aligned}$$

The coefficients A, B, C, and D are products of bilinear factors of the a_i directed link lengths representing the constants to be identified in the synthesis. Following derivation steps listed in [19], the remaining five v_i-v_j IO equations are, respectively [32],

$$A_1B_2v_1^2v_2^2 + A_2B_1v_1^2 + C_1D_2v_2^2 - 8a_2a_4v_1v_2 + C_2D_1 = 0, \tag{4}$$

$$A_1B_1v_1^2v_3^2 + A_2B_2v_1^2 + C_2D_2v_3^2 + C_1D_1 = 0, \tag{5}$$

$$A_1D_2v_2^2v_3^2 + B_2C_1v_2^2 + B_1C_2v_3^2 - 8a_1a_3v_2v_3 + A_2D_1 = 0, \tag{6}$$

$$A_1C_1v_2^2v_4^2 + B_2D_2v_2^2 + A_2C_2v_4^2 + B_1D_1 = 0, \tag{7}$$

$$A_1C_2v_3^2v_4^2 + B_1D_2v_3^2 + A_2C_1v_4^2 + 8a_2a_4v_3v_4 + B_2D_1 = 0. \tag{8}$$

3 Continuous Approximate Synthesis

While multi-modal continuous approximate IO function generation is the subject of this paper, it is built upon the foundation of the continuous approximate synthesis algorithm [20]. This approach involves integrating the synthesis equations between the bounds of the minimum and maximum input angles. The inspiration for this is that it has been observed that as the cardinality of the prescribed discrete IO data set of precision pairs that satisfy the prescribed function and the corresponding overdetermined set of synthesis equations increases, the identified linkages that minimise the Euclidean norm, or L_2-norm as is also called, of the design and structural errors tend to converge to the same set of link lengths [15].

It is worth noting that the most common L_p-norms [33] for a continuous function f on a closed interval $[a, b]$, and in fact, the most commonly used vector norms [34] are the Chebyshev norm, the Euclidean norm, and the Manhattan norm, which are respectively defined to be [33]:

$$\begin{aligned}
\|f\|_\infty &= \max_{x\in[a,b]} |f(x)|; \\
\|f\|_2 &= \left(\int_a^b f(x)^2 dx \right)^{1/2}; \\
\|f\|_1 &= \int_a^b |f(x)| dx.
\end{aligned}$$

The term Manhattan norm arises because this vector norm corresponds to sums of distances along the basis vector directions, as one would travel along a rectangular street plan. The Manhattan and Chebyshev norms are the limiting cases $p = 1$ and $p = \infty$, respectively, of the family of L_p-norms. The L_p-norms obey the following relationship:

$$\|f\|_\infty \leq \cdots \leq \|f\|_2 \leq \|f\|_1.$$

Typically, the most appropriate norm must be selected to evaluate the magnitude of the objective function for the error minimisation, given a function that is to be approximated by the resulting linkage. However, it turns out that Lawson's algorithm [35, 36] can be used to sequentially minimise the Chebyshev norm via the minimisation of the Euclidean norm [37]. This means that the continuous approximate synthesis approach to structural or design error minimisation is independent of the L_p-norm because it applies to both the Chebyshev and Euclidean norms, and hence all intermediate ones.

The important implication of this observation is that the minimisation of any L_p-norm of the structural error can be accomplished indirectly via the minimisation of the corresponding norm of the design error, provided that a suitably large number of IO pairs is prescribed. Again, this is desirable because the design error, which indicates the error residual incurred by a specific linkage regarding the verification of the synthesis equations, results in a linear least-squares problem, while the structural error is the difference between the prescribed linkage output angle and the generated output angle for a prescribed input angle value, which leads to a nonlinear optimisation problem generally requiring an iterative solution [8].

If the question is: "how large must the data set cardinality be?"; the easy answer is: "it doesn't matter if the cardinality is infinite!". Hence, the following six-step algorithm [20]:

1. Square the algebraic IO equation for the desired planar four bar linkage architecture.

2. Separate this squared IO equation into an array containing the linkage coefficients, $\mathbf{c}$, and an array containing the corresponding variable angle parameters, $\mathbf{s}$.

3. Substitute the prescribed function between the input and output variable pairs $(v_i,\ v_j)$ into the output parameter $v_j = f(v_i)$ in the variable array, $\mathbf{s}$.

4. To establish the synthesis equation take the Euclidean inner product of $\mathbf{c}$ with the integral of $\mathbf{s}$ over the prescribed bounds thus: $\mathbf{c} \cdot \int_{v_{i_{\min}}}^{v_{i_{\max}}} \mathbf{s}\left(v_i,\ v_j = f(v_i)\right) dv_i$.

5. Generate an initial guess for the optimal linkage parameters from the exact precision point synthesis satisfying the algebraic IO equation.

6. Minimise the residual of this integrated synthesis equation for the a_i link lengths over the field of real numbers.

The output of this algorithm is the four link lengths, a_i, that minimise both the design and structural errors for the planar 4R linkage in generating the prescribed $v_j = f(v_i)$ function. This algorithm can be summarised by the equation

$$\min_{(a_1,a_2,a_3,a_4)\in\mathbb{R}} \left(\mathbf{c} \cdot \int_{v_{i_{\min}}}^{v_{i_{\max}}} \mathbf{s}(v_i, f(v_i)) dv_i \right) = 0. \tag{9}$$

The *Minimize* command used in Maple 2021 to solve the problem computes a local minimum of an objective function subject to constraints. If the problem is convex, as when the objective function and constraints are linear, for example, the solution will also be a global minimum. The algorithms that this command use assume the objective function and constraints are twice continuously differentiable.

In this context, the Euclidean norm of the structural error over every point in the generated function is nothing more than the area between the prescribed function and the generated function in the variable angle parameter plain, which is equivalent to the design error. Examples of the continuous approximate synthesis will be described in Section 4.

4 Multi-Modal Continuous Approximate Synthesis

The concept of continuous approximate synthesis for function generation from [20] will be extended in order to enable the simultaneous approximate generation of multiple different, though not arbitrary competing, prescribed functions between different pairs of joint variable parameters in a single planar, spherical, or spatial four-bar mechanism, which we call *multi-modal continuous approximate synthesis*. We propose that this can be accomplished with the following:

$$\min_{(a_1,a_2,a_3,a_4)\in\mathbb{R}} \left(\mathbf{c}_1 \cdot \int_{v_{i_{1_{\min}}}}^{v_{i_{1_{\max}}}} \mathbf{s}_1(v_{i_1}, f_1(v_{i_1})) dv_{i_1} + \mathbf{c}_2 \cdot \int_{v_{i_{2_{\min}}}}^{v_{i_{2_{\max}}}} \mathbf{s}_2(v_{i_2}, f_2(v_{i_2})) dv_{i_2} \right) = 0. \tag{10}$$

The typical function generation problem concerns $\theta_4 = f(\theta_1)$ and the corresponding v_1-v_4 IO equation; however, considering Figure 1, one may wish to also consider the θ_1-θ_3 pair of angles, or any other of the remaining four pairs. For this *proof-of-concept* of the proposed multi-modal continuous approximate synthesis method we shall begin with the synthesis of two arbitrarily different functions $v_4 = f_1(v_1)$ and $v_3 = f_2(v_1)$. The reason

for this choice is that the v_3 angle parameter is a measure of the transmission angle, which is useful as a metric to discriminate between four bar mechanisms which have practical use and those that do not.

This idea has a philosophical existential question associated with it. Namely, when a mechanism is identified to generate, for example, $v_4 = f_1(v_1)$, the five other $v_j = f_2(v_i)$ functions are explicitly defined. Suppose a $v_3 = f_2(v_1)$ function was needed that was different from the one imposed by the initially generated $v_4 = f_1(v_1)$ function. The question we ask now is "does a linkage exist that is the best compromise between the competing prescribed functions?". The answer is, in general, no. However, we will show that polynomial interpolants can be used to perturb one of the functions and we can succeed. We define the design parameter space of planar 4R function generator linkages [3, 38, 39] as the four-dimensional homogeneous space spanned by the mutually orthogonal basis vectors $\mathbf{a}_1,\ \mathbf{a}_2,\ \mathbf{a}_3$ normalised with respect to frame length $a_4 = 1$. Distinct points in this homogeneous space, $(a_1 :\ a_2 :\ a_3 :\ 1)$, where the delimiter : has been used to indicate the use of homogeneous coordinate ratios, represent distinct planar 4R linkages. Each point is a linkage that generates six distinct functions between the six distinct angle parings between different links. The linkages identified to generate the prescribed $v_4 = f_1(v_1)$ and $v_3 = f_2(v_1)$ functions represent two distinct points, and therefore two distinct linkages. We will illustrate in Section 4.1.1 that in general the synthesis of competing functions is not possible in any useful way. However, in Section 4.1.2 we will show that it is possible to subtly perturb one of the functions leading to useful results.

4.1 Planar 4R Multi-modal Function Generation

Let the prescribed $v_4 = f_1(v_1)$ function be

$$v_4 = f_1(v_1) = 2 + \tan\left(\frac{v_1}{{v_1}^2 + 1}\right), \quad -\frac{1}{2} \leq v_1 \leq 2. \tag{11}$$

We proceed to identify the linkage that will approximately generate this function using continuous approximate synthesis. The first step is to square Equation (3), then separate the link length coefficients into arrays $\mathbf{c}_1$ and $\mathbf{s}_1$, yielding

$$\mathbf{c}_1 = \begin{bmatrix} A^2 \\ 2AB \\ B^2 \\ -16Aa_1a_3 \\ -16Ba_1a_3 \\ 2AC \\ 64a_1^2a_3^2 + 2AD + 2BC \\ 2BD \\ -16Ca_1a_3 \\ -16Da_1a_3 \\ C^2 \\ 2CD \\ D^2 \end{bmatrix}, \quad \mathbf{s}_1 = \begin{bmatrix} v_1^4v_4^4 \\ v_1^4v_4^2 \\ v_1^4 \\ v_1^3v_4^3 \\ v_1^3v_4 \\ v_1^2v_4^4 \\ v_1^2v_4^2 \\ v_1^2 \\ v_1v_4^3 \\ v_1v_4 \\ v_4^4 \\ v_4^2 \\ 1 \end{bmatrix} = \begin{bmatrix} v_1^4f_1(v_1)^4 \\ v_1^4f_1(v_1)^2 \\ v_1^4 \\ v_1^3f_1(v_1)^3 \\ v_1^3f_1(v_1) \\ v_1^2f_1(v_1)^4 \\ v_1^2f_1(v_1)^2 \\ v_1^2 \\ v_1f_1(v_1)^3 \\ v_1f_1(v_1) \\ f_1(v_1)^4 \\ f_1(v_1)^2 \\ 1 \end{bmatrix}. \tag{12}$$

We solve the exact synthesis problem to obtain an initial guess for the optimisation, using the prescribed function pairs that satisfy Equation (11):

$$(v_1,\ v_4)\ =\ \left(-\frac{1}{2},\ \frac{32287}{20471}\right);\ \left(\frac{3}{4},\ \frac{49597}{20471}\right);\ \left(2,\ \frac{48857}{19383}\right).$$

Note that to obtain these three precision IO pairs we have selected the lower and upper bounding values for v_1 and an arbitrary value in between, while the corresponding value of v_4 satisfies the prescribed function, Equation (11).

In this classic 4R exact synthesis problem we obtain a unique solution that contains the link length a_4 as a free parameter:

$$a_1 = -\frac{21111}{109000}a_4,\ a_2 = \frac{21021}{18196}a_4,\ a_3 = \frac{21518}{15263}a_4,\ a_4. \tag{13}$$

We arbitrarily set $a_4 = 1$ and evaluate the integral, then minimise the residual using the normalised link lengths in Equation (13) as the initial guess:

$$\min_{(a_1,a_2,a_3,a_4)\in\mathbb{R}} \left(\mathbf{c}_1 \cdot \int_{v_1=-\frac{1}{2}}^{v_1=2} \mathbf{s}_1(v_1, f_1(v_1))\right) = 0. \tag{14}$$

The minimisation is accomplished using the `Optimization` solvers in Maple 2021 which converge to the link lengths listed in Table 1.

Table 1 Continuous approximate synthesis results generating Equation (11).

Link length	a_1	a_2	a_3	a_4
Rational	$-\frac{13077}{78259}$	$\frac{45079}{42170}$	$\frac{27203}{20556}$	$\frac{101727}{110482}$
Floating point	-0.167098992	1.068982689	1.323360576	0.920756322
Normalised	-0.1814801460	1.160983273	1.437253857	1

Table 2 Structural error generating Equation (11).

Structural error	Exact synthesis	Continuous approximate synthesis
	0.024159094	-0.002471306

Comparisons of the structural error, defined as the area between the prescribed and generated functions, in the v_1-v_4 plane are enumerated in Table 2. One can see that the structural error for the function generated by the continuous approximate synthesis linkage is an order of magnitude smaller than that of the function generated by the exact synthesis linkage, as can be observed by casual visual inspection of the graphs plotted in Figure 6.

This approximately generated $v_4 = f_1(v_1)$ function exactly generates five additional $v_j = f_2(v_i)$ functions given the link lengths identified to approximately generate the prescribed function; exact in the sense that these functions have not been explicitly prescribed. These five functions between the angle parameters v_1-v_2, v_1-v_3, v_2-v_3, v_2-v_4, and v_3-v_4 are generated by the identified link lengths, and are illustrated in Figure 7 along with the prescribed and continuous approximate $v_4 = f_1(v_1)$ functions.

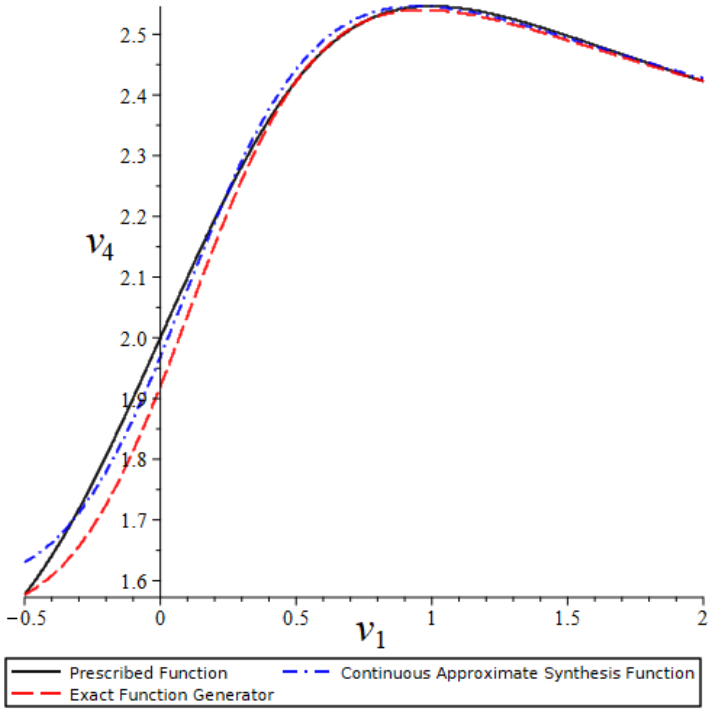


Figure 6: The prescribed, exact, and continuous synthesis approximation of Equation (11) in the v_1-v_4 plane.

4.1.1 First Multi-Modal Synthesis Attempt

Suppose we now wish to additionally identify a linkage that can approximately generate the v_1-v_4 function in Equation (11) and approximately generate a competing v_1-v_3 function that is very different from the v_1-v_3 function generated by the link lengths listed in Table 1. The pragmatic mechanical engineer response to such a wish is simply that it is not possible with a planar 4R. But, should it not be possible to identify a compromise linkage that will generate both desired functions with tolerable structural error? The naïve answer is surely "why not?!?".

Let us first look at this from the pragmatic mechanical engineer perspective and select the additional v_1-v_3 function to be

$$v_3 = f_2(v_1) = 2 + \tan\left(\frac{v_1^2}{{v_1}^2 + 1}\right), \quad -2 \leq v_1 \leq 2. \tag{15}$$

The v_1-v_3 function generated by the linkage that approximately generated the prescribed v_1-v_4 function can be seen in Figure 7, and is reproduced for comparison with the very different desired v_1-v_3 function in Figure 8. We select the range $-2 \leq v_1 \leq 2$ for the prescribed v_1-v_3 function, and $0 \leq v_1 \leq 2$ for the prescribed v_1-v_4 function, then select initial guesses for the four link lengths and compute Equation (10). We arbitrarily select $(a_1,\ a_2,\ a_3,\ a_4) = (1,\ 1,\ 1,\ 1)$ for link length initial guesses. This yields in the remarkably poor results illustrated in Figure 9.

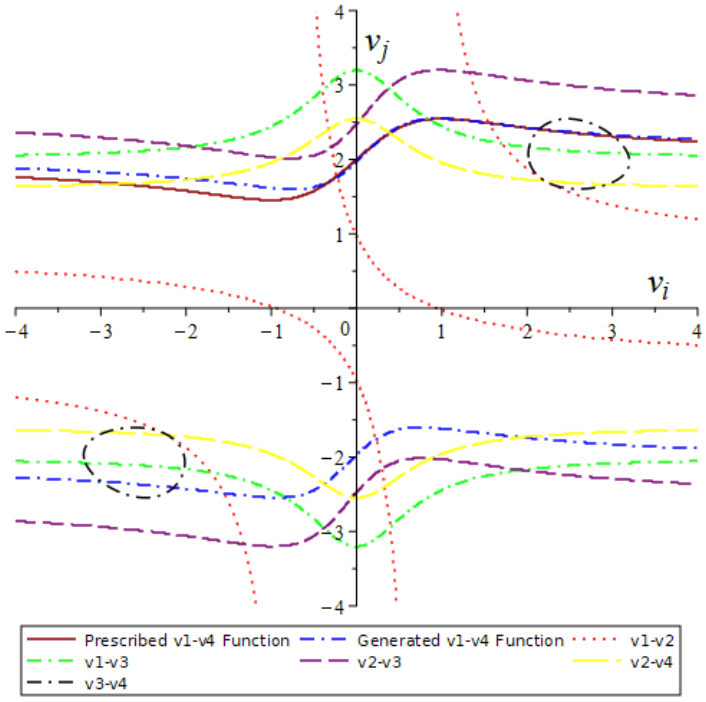


Figure 7: The prescribed, continuous synthesis approximate, and the five functions generated by the identified link lengths in the v_i-v_j planes.

4.1.2 Second Multi-Modal Synthesis Attempt

Clearly, the second, third, fourth, et c., prescribed functions need to be constrained with respect to the five functions generated by the link lengths that approximately generate the first prescribed function in the absence of a useful initial guess. Enter: polynomial interpolation. If we wish to generate a different, though heavily constrained, $v_3 = f_2(v_1)$ function we can specify a generatable function that is an interpolant of the one determined by the specified primary $v_4 = f_1(v_1)$ function. To do this we arbitrarily choose to use Lagrange polynomial interpolation [40].

The first step is to solve the v_1-v_3 IO equation imposed by the generated $v_4 = f_1(v_1)$ function. This yields the exact $v_3 = f_2(v_1)$ function generated by the identified a_i link lengths that approximately satisfy the specified $v_4 = f_1(v_1)$ function. Select n (v_1, v_3) IO pairs from the exact $v_3 = f_2(v_1)$ function generated by the identified a_i to use as inputs for the Lagrange polynomial interpolation formula. In general, this method takes the n points in an arbitrary x-y plane, with no two x_i the same and returns a polynomial of degree at most $d \leq n - 1$.

The Lagrange polynomial interpolant is a linear combination

$$L(x) = \sum_{i=1}^{n} y_i l_i(x)$$

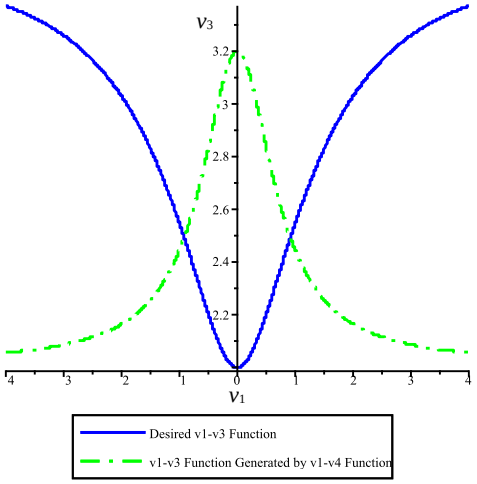
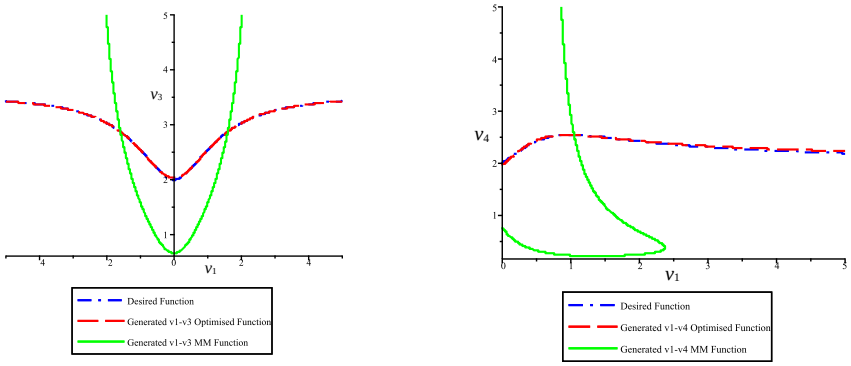


Figure 8: The desired competing v_1-v_3 function and the one generated by the linkage that approximates Equation 11.



(a) The v_1-v_3 multi-modal results.

(b) The v_1-v_4 multi-modal results.

Figure 9: Multi-modal synthesis results for two competing functions.

of Lagrange basis polynomials

$$l_i(x) = \prod_{\substack{1 \leq m \leq n \\ m \neq i}} \frac{x - x_m}{x_i - x_m} = \left(\frac{x - x_1}{x_i - x_1}\right)\left(\frac{x - x_2}{x_i - x_2}\right)\cdots\left(\frac{x - x_n}{x_i - x_n}\right).$$

For our computational *proof-of-concept* example we will use a system of primary and secondary prescribed functions. The primary function is arbitrary. But the secondary is some Lagrange polynomial interpolant of the function imposed by the link lengths identified that approximately generate the primary function. The link lengths that approximately generate the primary function will be used as initial guesses for the multi-modal synthesis with the secondary polynomial interpolant function. The primary function we wish to generate with

a planar 4R closed kinematic chain is Equation (11). The corresponding v_1-v_3 function exactly generated by the identified link lengths is obtained from the v_1-v_3 IO equation, Equation (5), using the a_i from the $v_4 = f_1(v_1)$ continuous approximate synthesis step listed in Table 1 is

$$v_3 = \pm \frac{11268158900\sqrt{\left(v_1^2 + \frac{28145}{62561}\right)\left(v_1^2 + \frac{43467}{38278}\right)}}{5593605380v_1^2 + 2516456313}. \tag{16}$$

Suppose that this crank-rocker four-bar linkage was required to precisely time two punch presses. Four holes created by the presses are required to be precisely located on a single automotive quarter panel which is advanced in a jig under the action of the input link of the mechanism. One quarter panel completely advances per 360° rotation of the input crank link. The first punch press is actuated by a trigger that is activated under the action of θ_4, while the second is actuated by θ_3. The $v_4 = f_1(v_1)$ trigger function is that of Equation (11). However, after the linkage is synthesised, the resulting $v_3 = f_2(v_1)$ function, Equation (16), does not satisfy the angle requirement. The trigger for this punch press must be actuated when the input angle locating the quarter panel has the precise values $\theta_1 = 0.00° \pm 0.05°$ and $\theta_1 = 90.00° \pm 0.05°$. At these input angles the corresponding values of θ_3 must be precisely $\theta_3 = 145.25° \pm 0.05°$ and $\theta_3 = 135.25° \pm 0.05°$. Unfortunately, while the values of θ_4 generated by the linkage obtained by continuous approximate synthesis as listed in Table 1 are within tolerance for the required input angles those for θ_3 are not. The required angle generated by this linkage at $\theta_1 = 0.00° \pm 0.05°$ is $\theta_3 = 145.50° \pm 0.05°$ and at $\theta_1 = 90.00° \pm 0.05°$ is $\theta_3 = 135.10° \pm 0.05°$, both out of tolerance, though only marginally, see Table 3. Relaxing the tolerances is deemed to not be an acceptable design course of action. In this case, subtly perturbing the $v_3 = f_2(v_1)$ function generated by the required $v_4 = f_1(v_1)$ function, Equation (11), may yield the required θ_4 and θ_3 output angles.

Table 3 Required and $v_4 = f_1(v_1)$ generated values of θ_3 at required θ_1.

Required θ_1	$0.00° \pm 0.05°$	$90.00° \pm 0.05°$
Required θ_3	$145.25° \pm 0.05°$	$135.25° \pm 0.05°$
Generated θ_3	$145.50°$	$135.10°$

To achieve this, we will attempt to use Lagrange polynomial interpolation to obtain a different, but constrained function using $n = 4$ points on the (upper signed) $v_3 = f_2(v_1)$ curve, Equation (16):

$$(v_1, v_3) \;=\; \left(-\frac{1}{2}, \frac{62167}{21933}\right), \left(\frac{1}{4}, \frac{80364}{26089}\right), \left(\frac{3}{5}, \frac{64227}{23462}\right), \left(\frac{11}{10}, \frac{39821}{16629}\right).$$

The resulting degree 3 Lagrange polynomial function $v_3 = f_2(v_1)$ is

$$v_3 = \frac{140152452564627675650}{146115499161206849967}v_1^3 - \frac{148500638129317309265}{97410332774137899978}v_1^2$$
$$-\frac{136182081139230857387}{584461996644827399868}v_1 + \frac{57010242995943671417}{17710969595297799996}, \tag{17}$$
$$\text{for} \quad -\frac{1}{10} \le v_1 \le \frac{5}{4}.$$

Let this be the specified secondary function. Both the interpolant, Equation (18), and the precise v_1-v_3 function, Equation (16), generated by the link lengths that were identified to approximately generate Equation (11) are illustrated in Figure 10.

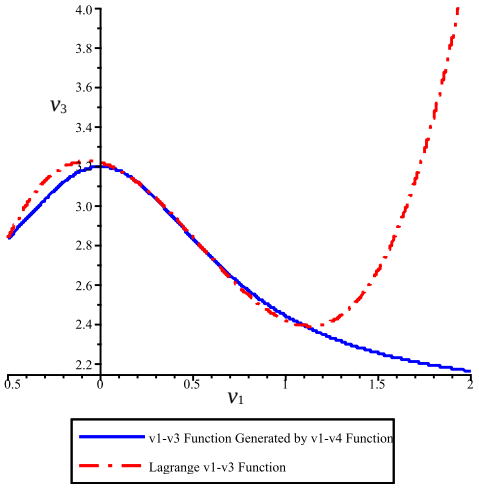
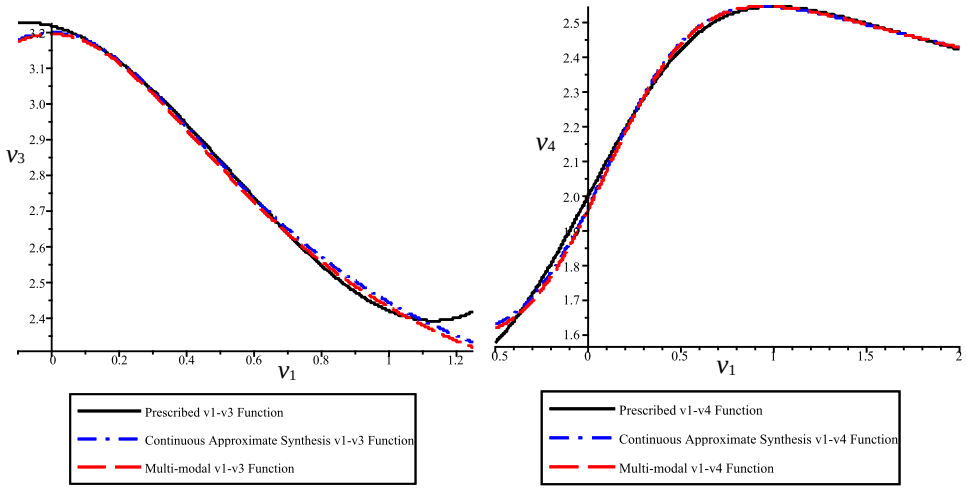


Figure 10: The polynomial interpolant, Equation (18), and the v_1-v_3 function, Equation (16), generated by the linkage that approximates Equation (11).

Careful examination of Figure 10 reveals that both Equation (16) and (18) are very close to each other in the range $-\frac{1}{10} \leq v_1 \leq \frac{5}{4}$. To demonstrate that our kinematic model of the geometry of multi-modal synthesis will lead to a computationally useful result, we will use these as the integration limits for the v_1-v_3 secondary function. Hence, the primary $v_4 = f_1(v_1)$, Equation (11), and secondary $v_3 = f_2(v_1)$, Equation (18), are used to generate the respective synthesis equations with variable angle parameters expressed as v_1 and $f_1(v_1)$ in the primary, and v_1 and $f_2(v_1)$ in the secondary. The two synthesis equations are squared, then the coefficients and variables are separated into the arrays $\mathbf{c}_1$, $\mathbf{s}_1(v_1, f_1(v_1))$, $\mathbf{c}_2$, and $\mathbf{s}_2(v_1, f_2(v_1))$. We then evaluate

$$\min_{(a_1,a_2,a_3,a_4)\in\mathbb{R}} \left(\mathbf{c}_1 \cdot \int_{v_1=-\frac{1}{2}}^{v_1=2} \mathbf{s}_1(v_1, f_1(v_1)) dv_1 + \mathbf{c}_2 \cdot \int_{v_1=-\frac{1}{10}}^{v_1=\frac{5}{4}} \mathbf{s}_2(v_1, f_2(v_1)) dv_1 \right). \quad (18)$$

The multi-modal computations for Equation (18) converge to the link lengths listed in Table 5. The results are graphically illustrated in Figure 11 and the structural errors, defined as the areas between the prescribed and generated functions are listed in Table 6. It is to be seen that the structural error for the $v_4 = f_1(v_1)$ results increases by a factor of nearly four, but is still tolerably small. While the structural error for the $v_3 = f_2(v_1)$ multi-modal results decreases modestly. However, the important outcome in this case is that at the required input angles the corresponding required values of θ_4 are still within tolerance, and those of θ_3 are as well. When $\theta_1 = 0.00^\circ \pm 0.05^\circ$ the multi-modal linkage generates $\theta_3 = 145.25^\circ \pm 0.05^\circ$ and at $\theta_1 = 90.00^\circ \pm 0.05^\circ$ we obtain $\theta_3 = 135.28^\circ \pm 0.05^\circ$, both within tolerance. The relevant values of this outcome are listed in Table 4.



(a) The v_1-v_3 multi-modal results. (b) The v_1-v_4 multi-modal results.

Figure 11: Multi-modal 4R results.

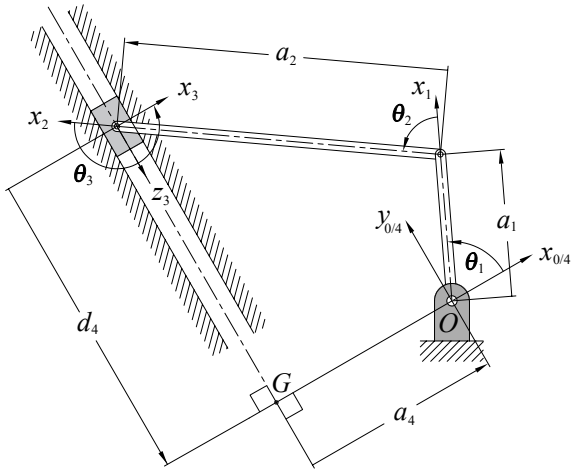


Figure 12: Planar RRRP linkages with Denavit-Hartenberg coordinate systems and parameter assignments.

Table 4 Required and multi-modal generated values of θ_3 at required θ_1.

Required θ_1	$0.00^\circ \pm 0.05^\circ$	$90.00^\circ \pm 0.05^\circ$
Required θ_3	$145.25^\circ \pm 0.05^\circ$	$135.25^\circ \pm 0.05^\circ$
Generated θ_3	145.25°	135.28°

Table 5 The $v_4 = f_1(v_1)$ and $v_3 = f_2(v_1)$ planar 4R multi-modal synthesis results.

Link length	a_1	a_2	a_3	a_4
Floating point	-0.1478064777	0.9299394483	1.148016662	0.8023065449
Normalised	-0.1842269375	1.159082466	1.430895297	1

Table 6 The $v_4 = f_1(v_1)$ and $v_3 = f_2(v_1)$ planar 4R multi-modal synthesis structural errors.

	Structural error
$v_4 = f_1(v_1)$ only	-0.002471306
$v_4 = f_1(v_1)$ multi-modal	0.009542948
$v_3 = f_2(v_1)$ only	0.005358289
$v_3 = f_2(v_1)$ multi-modal	0.004161159

4.2 Planar RRRP Multi-modal Function Generation

Next we shall list the six algebraic IO equations for planar RRRP mechanisms and perform multi-modal synthesis. An arbitrary RRRP linkage is illustrated in Figure 12. The P-pair z_3-axis induces the two link twist angles listed in Table 7.

Table 7 DH parameters for the RRRP.

i	θ_i	d_i	a_i	τ_i	α_i
1	θ_1	0	a_1	0	0
2	θ_2	0	a_2	0	0
3	θ_3	0	0	$\pi/2$	1
4	0	d_4	a_4	-$\pi/2$	-1

By applying the methods in [19] to the DH parameters by algebraising the angle parameters with tangent half-angle equivalents, projecting the DH closure equation into Study's kinematic mapping image space as soma coordinates, then eliminating the intermediate joint variable parameters leads to the RRRP algebraic IO equation:

$$v_1^2 d_4^2 + R v_1^2 + d_4^2 - 4 a_1 v_1 d_4 + S = 0, \tag{19}$$

where

$$R = R_1 R_2 = (a_1 + a_2 - a_4)(a_1 - a_2 - a_4),$$
$$S = S_1 S_2 = (a_1 + a_2 + a_4)(a_1 - a_2 + a_4),$$
$$v_1 = \tan\frac{\theta_1}{2}.$$

Using the same approach [19], the five remaining joint variable parameter pairings lead to the following five RRRP algebraic IO equations:

$$R_2 v_1^2 v_2^2 + R_1 v_1^2 - S_2 v_2^2 + 4a_2 v_1 v_2 - S_1 = 0; \tag{20}$$
$$R_1 v_1^2 v_3^2 + R_2 v_1^2 - S_2 v_3^2 - S_1 = 0; \tag{21}$$
$$S_2 v_2^2 v_3^2 - R_2 v_2^2 - R_1 v_3^2 - 4a_1 v_2 v_3 + S_1 = 0; \tag{22}$$
$$v_2^2 d_4^2 - R_2 S_2 v_2^2 + d_4^2 - R_1 S_1 = 0; \tag{23}$$
$$v_3^2 d_4^2 + R_1 S_2 v_3^2 + d_4^2 + 4a_2 v_3 d_4 - R_2 S_1 = 0. \tag{24}$$

Our primary $d_4 = f_1(v_1)$ function is arbitrarily chosen to be

$$d_4 = 2 - \ln\left(\frac{v_1^2}{v_1^2 + 1}\right), \quad \frac{1}{10} \leq v_1 \leq 6. \tag{25}$$

To generate an initial guess for the multi-modal synthesis, we first perform exact followed by continuous approximate synthesis and identify the following link lengths:

$$a_1 = -\frac{21527}{19453},\ a_2 = \frac{62456}{9833},\ a_4 = \frac{66527}{13759}. \tag{26}$$

After following similar computation steps as for the planar 4R multi-modal synthesis example in Section 4.1.2, we determine the secondary $v_3 = f_2(v_1)$ function again as a degree 3 Lagrange interpolant:

$$v_3 = \frac{8575459781525718313}{2128203922635547524924}v_1^3 - \frac{926446934929263804951}{7094013075451825083080}v_1^2$$
$$+\frac{145850030457909132287}{123732786199741135170}v_1 + \frac{3255237430904027623667}{1773503268862956270770}. \tag{27}$$

We arbitrarily, but without loss of generality, assign the integration limits for this perturbed secondary function to be the same as those of the primary function. The multi-modal synthesis is then performed by evaluating

$$\min_{(a_1,a_2,a_4)\in\mathbb{R}} \left(\mathbf{c}_1 \cdot \int_{v_1=\frac{1}{10}}^{v_1=6} \mathbf{s}_1(v_1, f_1(v_1)) dv_1 + \mathbf{c}_2 \cdot \int_{v_1=\frac{1}{10}}^{v_1=6} \mathbf{s}_2(v_1, f_2(v_1)) dv_1 \right). \tag{28}$$

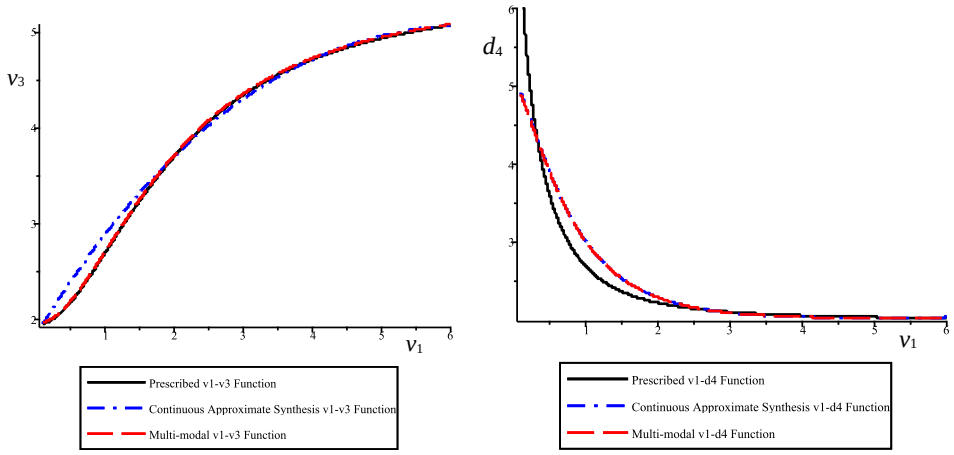
The numerical optimiser in Maple 2021 converges to the link lengths listed in Table 8, while the structural errors for each of the two generated functions are listed in Table 9. To help visualise the areas between the prescribed and generated functions the results are illustrated in Figure 13. It can be seen that the structural error decreases for the multi-modal synthesis results.

Table 8 The $d_4 = f_1(v_1)$ and $v_3 = f_2(v_1)$ planar RRRP multi-modal synthesis results.

Link length	a_1	a_2	a_4
Rational	$-\frac{26513}{23888}$	$\frac{85324}{13461}$	$\frac{127711}{26510}$
Floating point	-1.10988780904891	6.33860782950366	4.81746510770270

Table 9 The $d_4 = f_1(v_1)$ and $v_3 = f_2(v_1)$ planar RRRP multi-modal synthesis structural errors.

	Structural error
$d_4 = f_1(v_1)$ only	-0.24046271
$d_4 = f_1(v_1)$ multi-modal	-0.23104280
$v_3 = f_2(v_1)$ only	0.23488469
$v_3 = f_2(v_1)$ multi-modal	0.15360825



(a) The v_1-v_3 RRRP multi-modal results. (b) The v_1-d_4 RRRP multi-modal results.

Figure 13: Multi-modal RRRP results.

5 Conclusion

The main goal of this paper was to describe a novel four-bar planar mechanism algorithm that implicitly drives the cardinality of the IO data set used to generate the over constrained set of synthesis equations to infinity, and to use it to identify link parameters to simultaneously satisfy two desired functions between different IO pairs. This is accomplished by integrating the square of the algebraic IO equation for the desired kinematic architecture over the specified range of input parameter, v_i or d_i, where the output parameter, v_j or d_j, depending on the kinematic architecture, is expressed in terms of the prescribed function, $v_j = f(v_i)$, et c., for each desired function. The synthesis equation, which we have termed multi-modal,

is the sum of the squared IO equations integrated over the desired input parameter ranges. Because of this, we denote the entire procedure as multi-modal continuous approximate synthesis.

The algorithm was demonstrated with two multi-modal synthesis examples, in a proof-of-concept fashion, to simultaneously generate primary and perturbed secondary functions in each of a 4R and an RRRP planar linkage, and to demonstrate that generation of competing functions with a planar four-bar linkage is, in general, not possible. Comparing the prescribed and generated continuous functions over their specified ranges we have observed that both the design and structural errors are simply the difference in the areas under the prescribed and generated IO curves in the joint variable parameter planes. The multi-modal synthesis results lead to reductions in the structural errors, or at worst a reasonably modest increase. Certainly, any planar four-bar mechanism generates an output joint parameter that is a distinct function of the input joint variable parameter. The linkage that generates this distinct function also exactly determines five additional functions between the remaining pairs of variable joint parameters. The synthesis examples in this paper have demonstrated that it is possible to approximately generate two distinct, though heavily constrained, IO functions between different variable joint pairs that have not been already determined by the linkage geometry. This simple result illustrates the tremendous value represented by the algebraic IO equations as design and analysis tools.

The algebraic IO equations described herein, together with the multi-modal continuous approximate synthesis algorithm, stand to enable designers of industrial automated production and assembly systems to approach optimisation in a new way: different linkages in the mechanical system that are capable of generating multiple different prescribed functions so that each link in the chain can simultaneously perform different tasks. While the practicality of this is, of course, conjecture, it does suggest the continued generalisation and development of multi-modal continuous approximate synthesis is justified and worth the investigative effort. The next step involves research on how to determine suitable initial guesses for the multi-modal synthesis that will yield useful results without heavily constrained secondary functions, and to what degree this is possible. The authors believe this knowledge is to be uncovered in the geometry of the associated design parameter spaces.

References

[1] R.S. Hartenberg and J. Denavit. *Kinematic Synthesis of Linkages*. McGraw-Hill Book Co., New York, N.Y., U.S.A., 1964.

[2] J. Angeles, A. Alivizatoss, and R. Akhras. An Unconstrained Nonlinear Least-squares Method of Optimization of RRRR Planar Path Generators. *Mechanism and Machine Theory*, 23(5):343–353, 1988.

[3] C.M. Gosselin and J. Angeles. Optimization of Planar and Spherical Function Generators as Minimum-defect Linkages. *Mechanism and Machine Theory*, 24(4):293–307, 1989.

[4] C.M. Gosselin, B. Moore, and J. Schicho. Dynamic Balancing of Planar Mechanisms Using Toric Geometry. *Journal of Symbolic Computation*, 44(9):1346–1358, 2009.

[5] F. Freudenstein. *Design of Four-link Mechanisms.* PhD thesis, Columbia University, New York, N.Y., USA, 1954.

[6] F. Freudenstein. Approximate Synthesis of Four-bar Linkages. *Trans. ASME*, 77:853–861, 1955.

[7] D.J. Wilde. Error Linearization in the Least-Squares Design of Function Generating Mechanisms. *ASME, J. of Mech. Des.*, 104(4):881–884, 1982.

[8] S.O. Tinubu and K.C. Gupta. Optimal Synthesis of Function Generators Without the Branch Defect. *ASME, J. of Mech., Trans., and Autom. in Design*, 106(3):348–354, 1984.

[9] Z. Liu and J. Angeles. Data Conditioning in the Optimization of Function-generating Linkages. In *Advances in Design Automation: Proc. 19th Annual ASME Design Automation Conference*, pages 419–426, 1993.

[10] M. Shariati and M. Norouzi. Optimal Synthesis of Function Generator of Four-bar Linkages Based on Distribution of Precision Points. *Mechanica*, 45(3):1007–1021, 2011.

[11] J. Zhang, J. Wang, and X. Du. Time-dependent Probabilistic Synthesis for Function Generator Mechanisms. *Mechanism and Machine Theory*, 46(9):1236–1250, 2011.

[12] P.A. Simionescu. A Restatement of the Optimum Synthesis of Function Generators with Planar Four-bar and Slider-crank Mechanisms Examples. *International Journal of Mechanisms and Robotic Systems*, 3(31):60–79, 2016.

[13] J. Denavit and R.S. Hartenberg. A Kinematic Notation for Lower-pair Mechanisms Based on Matrices. *J. of Applied Mechanics*, pages 215–221, 1955.

[14] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C,* 2^{nd} Edition. Cambridge University Press, Cambridge, England, 1992.

[15] M.J.D Hayes, K. Parsa, and J Angeles. The Effect of Data-set Cardinality on the Design and Structural Errors of Four-bar Function-generators. In *Proceedings of the Tenth World Congress on the Theory of Machines and Mechanisms,* Oulu, Finland, pages 437–442, 1999.

[16] A. Guigue and M.J.D. Hayes. Continuous Approximate Synthesis of Planar Function-generators Minimising the Design Error. *Mechanism and Machine Theory*, 101:158–167, 2016.

[17] M.J.D. Hayes, M.L. Husty, and M. Pfurner. Input-output Equation for Planar Four-bar Linkages. In *Advances in Robot Kinematics, 2018*, pages 12–19. Springer, 2018.

[18] M. Rotzoll, M.J.D. Hayes, and M.L. Husty. An Algebraic Input-output Equation for Planar RRRP and PRRP linkages. *Transactions of the Canadian Society for Mechanical Engineering*, 44(4):520–529, 2019.

[19] M. Rotzoll, M.J.D. Hayes, M.L. Husty, and M. Pfurner. A General Method for Determining Algebraic Input-output Equations for Planar and Spherical 4R inkages. In *International Symposium on Advances in Robot Kinematics*, pages 90–97, 2020.

[20] Z. Copeland, M. Rotzoll, and M.J.D. Hayes. Concurrent Type and Dimensional Continuous Approximate Function Generator Synthesis for All Planar Four-bar Mechanisms. In S. Nokleby and P. Cardou, editors, *11th CCToMM Symposium on Mechanisms, Machines, and Mechatronics,* Ontario Tech University, Oshawa, ON, Canada, 2021.

[21] M. Pfurner and M.L. Husty. MyKinematics: Maple Library for Kinematics Using Study's Soma Coordinates. Private communication, *IFToMM 2019 World Congress*, Kraków, Poland, July 2, 2019.

[22] M.L. Husty, M. Pfurner, and H.-P. Schröcker. A New and Efficient Algorithm for the Inverse Kinematics of a General Serial 6R Manipulator. *Mechanism and Machine Theory*, 47:66–81, 2007.

[23] J.J. Craig. *Introduction to Robotics, Mechanics and Control,* 2^{nd} Edition. Addison-Wesley Publishing Co., Reading, Mass., U.S.A., 1989.

[24] M.L. Husty, A. Karger, H. Sachs, and W. Steinhilper. *Kinematik und Robotik*. Springer-Verlag, Berlin, Germany, 1997.

[25] L.W. Tsai. *Robot Analysis: the Mechanics of Serial and Parallel Manipulators*. Wiley-Interscience, New York, N.Y., U.S.A., 1999.

[26] L. Sciavicco and B. Sciliano. *Modeling and Control of Robot Manipulators*. Springer-Verlag, New York, N.Y., U.S.A., 2000.

[27] J.M. McCarthy and G.S. Soh. *Geometric Design of Linkages,* 2^{nd} Edition, Interdiciplinaty Applied Mathematics. Springer, New York, N.Y., 2011.

[28] R.L. Norton. *Kinematics and Dynamics of Machinery,* 5^{th} Edition. McGraw-Hill, New York, N.Y., U.S.A., 2012.

[29] J.J. Uicker, Jr., G.R. Pennock, and J.E. Shigley. *Theory of Machines and Mechanisms,* 5^{th} Edition. Oxford University Press, New York, N.Y., U.S.A., 2017.

[30] M.L. Husty and H.-P. Schröcker. Kinematics and algebraic geometry. In J.M. McCarthy, editor, *21st Century Kinematics*, pages 85–123. Springer-Verlag, London, U.K., 2013.

[31] E. Study. *Geometrie der Dynamen*. Teubner Verlag, Leipzig, Germany, 1903.

[32] M. Rotzoll, Q. Bucciol, and M.J.D. Hayes. Algebraic Input-output Angle Equation Derivation Algorithm for the Six Distinct Angle Pairings in Arbitrary Planar 4R Linkages. In *20th International Conference on Advanced Robotics, ICAR 2021,* Ljubljana, Slovenia, 2021.

[33] G. Dahlquist and Å Björck. *Numerical Methods, translated by Anderson*. Prentice-Hall, Inc., U.S.A., 1969.

[34] J. E. Gentle. *Numerical Linear Algebra for Applications in Statistics*. Springer, New York, U.S.A., 1998.

[35] C. L. Lawson. *Contributions to the Theory of Linear Least Maximum Approximations*. PhD thesis, UCLA, Los Angeles, CA, U.S.A., 1961.

[36] J. R. Rice and K. H. Usow. The Lawson Algorithm and Extensions. *Mathematics of Computation*, 22(101):118–126, 1967.

[37] F. Angeles and J. Angeles. Synthesis of Function-Generating Linkages with Minimax Structural Error: the Linear Case. In *Proceedings of the 13th IFToMM World Congress,* Guanajuato, Mexico, 2011.

[38] C.M. Gosselin and J. Angeles. Mobility Analysis of Planar and Spherical Linkages. *Computers in Mechanical Engineering*, July/August 1988.

[39] M. Rotzoll and M.J.D. Hayes. Mobility Classification in the Design Parameter Space of Spherical 4R Linkages. In S. Nokleby and P. Cardou, editors, *11th CCToMM Symposium on Mechanisms, Machines, and Mechatronics,* Ontario Tech University, Oshawa, ON, Canada, 2021.

[40] E. Waring. Problems Concerning Interpolations. *Philosophical Transactions of the Royal Society*, 69:59–67, 1779.