# On the Evaluation of a General Model for Optimum RCCC Path Generation

<table>
<thead>
<tr>
<th>Journal:</th>
<th><em>Transactions of the Canadian Society for Mechanical Engineering</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>Manuscript ID</td>
<td>TCSME-2017-0073</td>
</tr>
<tr>
<td>Manuscript Type:</td>
<td>Article</td>
</tr>
<tr>
<td>Date Submitted by the Author:</td>
<td>04-Oct-2017</td>
</tr>
<tr>
<td>Complete List of Authors:</td>
<td>LEE, WEN-TZONG; National Pingtung University of Science and Technology, Biomechatronics Engineering Cosme, Jose; US Army Research, Development and Engineering Center Russell, Kevin; New Jersey Institute of Technology</td>
</tr>
<tr>
<td>Is the invited manuscript for consideration in a Special Issue?</td>
<td>N/A</td>
</tr>
<tr>
<td>Keywords:</td>
<td>RCCC Linkage, Path Generation, Constrained Nonlinear Optimization, Matlab</td>
</tr>
</tbody>
</table>

[https://mc06.manuscriptcentral.com/tcsme-pubs](https://mc06.manuscriptcentral.com/tcsme-pubs)
ON THE EVALUATION OF A GENERAL MODEL FOR OPTIMUM RCCC PATH GENERATION

Wen-Tzong Lee¹, Jose Cosme², Kevin Russell³

¹Department of Biomechatronics Engineering,
National Pingtung University of Science and Technology, Neipu, Pingtung 91201, Taiwan
²Munitions Engineering and Technology Center, US Army Research,
Development and Engineering Center, Picatinny, NJ 07806-5000, USA
³Department of Mechanical and Industrial Engineering,
New Jersey Institute of Technology, Newark, New Jersey 07102, USA
E-mail: wtleenpust.edu.tw; jose.c.cosme.civ@mail.mil; kevin.russell@njit.edu

Received Month 0000, Accepted Month 0000
No. 00-CSME-00, E.I.C. Accession Number 0000
ABSTRACT

A general optimization model for the dimensional synthesis of defect-free Revolute-Cylindrical-Cylindrical-Cylindrical joint or RCCC path generators is formulated, implemented and evaluated in this work. With this optimization model, the RCCC dimensions required to approximate precision points are calculated. The model includes constraints to eliminate order, branch and circuit defects-defects that are common in dyad-based dimensional synthesis. Therefore, the novelty of this work is the development of a general optimization model for RCCC path generation that simultaneously considers order, branch and circuit defect elimination. This work conveys both the benefits and drawbacks realized when implementing the optimization model on a personal computer using the commercial mathematical analysis software package Matlab.

Keywords: RCCC Linkage, Path Generation, Constrained Nonlinear Optimization, Matlab.
1 INTRODUCTION

1.1 The RCCC Linkage

The Revolute-Cylindrical-Cylindrical-Cylindrical joint or RCCC linkage (Figure 1) is a 1-DOF spatial four-bar linkage that includes one revolute joint and 3 cylindrical joints. In recent years, a variety of research has been dedicated to the dimensional synthesis, analysis and application of the RCCC linkage.

Developments in RCCC dimensional synthesis include the application of Burmester curve theory (to produce linkage solution regions) for the synthesis of RCCC motion generators to achieve four prescribed coupler positions (Cao and Han 2016). A robust formulation for the dimensional synthesis of C-C and R-C dyads was utilized to produce RCCC motion generators to achieve four prescribed coupler positions (Bai and Angeles 2015). By establishing a relation between RCCC coupler curves and RCC dimension types, a dimensional synthesis method for RCCC path generation was developed (Sun et al. 2012). A numerical atlas-based method for RCCC motion generation was also developed as well as a procedure that allows the synthesis of the axodes of an RCCC linkage (Figliolini et al. 2016, Sun et al. 2017).

Developments in the analysis of RCCC linkages include the development of parametric equations for RCCC coupler curves (Marble and Pennock 2000). From these equations, it was shown that the coupler point curve of the RCCC linkage is of order 16. By modeling the Cardan joint as an RCCC linkage, a method to conduct a full dynamic analysis of the RCCC linkage (and subsequently investigate the mechanical efficiency of the Cardan joint) was developed (Cavacece et al. 2004a, 2004b). Dual number algebra versions of classical linear algebra-based algorithms were developed and applied to the kinematic position analysis of the RCCC linkage (Pennestri and Stefanelli 2007). RCCC branch defect elimination has also been addressed for motion generation for three precision positions (Reinholtz et al. 1986).

Developments in the application of RCCC linkages include its application as a 1-DOF design alternative in a double chainstitch machine (Wang et al. 2005). Through the application of function
generation (a subcategory in dimensional synthesis)—specifically the application of curve fitting, the RCCC dimensions were obtained.

1.2 Scope of Work

The novelty of this work is the development, demonstration and evaluation of a general optimization model for RCCC path generation that simultaneously considers order, branch and circuit defect elimination. This work also conveys both the benefits and the drawbacks associated with the general RCCC optimization model and its implementation in the commercial mathematical analysis software package Matlab on a personal computer.

2 KINEMATIC DISPLACEMENT MODEL FOR THE RCCC LINKAGE

An analytical kinematic displacement model was presented by Suh and Radcliffe for the RCCC linkage (Suh and Radcliffe 1978). The driving link includes points \( a_0 \) and \( a_i \) and joint axis unit vectors \( u_{a_0} \) and \( u_{a_i} \) (Figure 1). Given a driving link displacement angle \( \theta \), the displacements of \( a_i \) and \( u_{a_i} \) are calculated from Equations (1) and (2) respectively.

\[
\begin{align*}
    a_j &= \left[ R_{\theta, u_{a_0}} \right] (a_i - a_0) + a_0 \tag{1} \\
    u_{a_j} &= \left[ R_{\theta, u_{a_0}} \right] u_{a_i} \tag{2}
\end{align*}
\]

The coupler link includes points \( b_1 \), \( c_i \) and \( p_i \) (the latter is an arbitrary coupler point) and joint axis unit vector \( u_{c_i} \) (Figure 1). The displacements of \( b_1 \), \( c_i \), \( p_i \) and \( u_{c_i} \) are calculated from Equations (3) through (6) respectively.

\[
\begin{align*}
    b_j &= a_j + s_{a_j} u_{a_j} \tag{3} \\
    c_j &= \left[ R_{\theta, u_{a_0}} \right] \left( \left[ R_{\theta, u_{a_0}} \right] (c_i - a_0) + a_0 - a_j \right) + a_j + s_{a_j} u_{a_j} \tag{4}
\end{align*}
\]
\[ p_j = \left[ R_{\lambda_j u_{a_j}} \right] \left( \left[ R_{\theta_j u_{a_0}} \right] (p_1 - a_0) + a_0 - a_j \right) + a_j + s_{a_j} u_{a_j} \]  

(5)

\[ u_{c_j} = \left[ R_{\lambda_j u_{a_j}} \right] \left( \left[ R_{\theta_j u_{a_0}} \right] u_{c_j} \right) \]  

(6)

The follower link includes points \( d_1, e_1 \) and \( f_0 \) and joint axis unit vector \( u_{e_0} \) (Figure 1). The displacements of \( d_1 \) and \( e_1 \) are calculated from Equations (7) and (8) respectively. Like Equation (6), Equation (9) calculates the displacement of joint axis vector \( u_{e_i} \) but with respect to follower link rotations about \( u_{e_0} \).

\[ d_j = c_j + s_{c_j} u_{c_j} \]  

(7)

\[ e_j = f_0 + s_{e_j} u_{e_0} \]  

(8)

\[ u_{c_j} = \left[ R_{\phi_j u_{e_0}} \right] u_{c_i} \]  

(9)

The sliding displacement of the RCCC cylindrical joints (Figure 1) \( s_a, s_e \) and \( s_c \) are calculated from Equation (10). This RCCC linkage loop closure equation is formulated by equating the vector path from \( a_0 \) to the displaced \( d_1 \) (which results in \( d_j = c_j + s_{c_j} u_{c_j} \)) to the vector path from \( f_0 \) to the displaced \( d_1 \) (which results in \( d_j = \left[ R_{\phi_j u_{e_0}} \right] (d_1 - e_1) + f_0 - s_{e_j} u_{c_j} \)). To avoid singularity in the inverted matrix in Equation (10), the joint axis vectors \( u_{a_i}, u_{c_i} \) and \( u_{e_i} \) (being unit vectors) should not all be assigned identical component values in RCCC kinematic analysis. Conversely, in RCCC dimensional synthesis, any solution where \( u_{a_i} = u_{c_i} = u_{e_i} \) is an invalid solution.

\[
\begin{bmatrix}
  s_{a_j} \\
  s_{c_j} \\
  s_{e_j}
\end{bmatrix} =
\begin{bmatrix}
  u_{a_x} & u_{c_x} & u_{e_{0y}} \\
  u_{a_y} & u_{c_y} & u_{e_{0y}} \\
  u_{a_z} & u_{c_z} & u_{e_{0z}}
\end{bmatrix}^{-1}
\begin{bmatrix}
  (f_0 - a_j) + \left[ R_{\phi_j u_{e_0}} \right] (d_1 - e_1) - \left[ R_{\lambda_j u_{a_0}} \right] \left( \left[ R_{\theta_j u_{a_0}} \right] (c_i - a_0) + a_0 - a_j \right) + s_{a_i} u_{a_j}
\end{bmatrix}
\]  

(10)
In Equations (1) through (10), matrices $[R_{0j,u_{a}}]$, $[R_{0j,u_{e}}]$ and $[R_{jj,u_{a}}]$ are 3x3 spatial rotation matrices about the fixed pivot axes $u_{a0}$ and $u_{e0}$ and the displaced driving link moving pivot joint axis $u_{a}$ respectively. RCCC linkage dimension variables $a_{0}$, $a_{1}$, $u_{a0}$, $u_{a1}$, $e_{1}$, $u_{e1}$, $f_{0}$, $u_{e0}$ and $p_{j}$ are all 3x1 vectors containing x, y and z-components.

3 GENERAL CONSTRAINED OPTIMIZATION MODEL FOR RCCC PATH GENERATION

The entire displacement model presented in Section 2 is included in the constrained nonlinear optimization model to be presented in this section. This optimization model includes the objective function

$$f(\mathbf{X}) = \sum_{j=2}^{n} \left\| \mathbf{p}_{j}^{*} - \mathbf{p}_{j} \right\|^{2}$$

(11)

In Equation (11), $\mathbf{X} = \left( a_{0}, u_{a0}, a_{1}, u_{a1}, c_{1}, u_{e1}, f_{0}, e_{1}, s_{a1}, s_{e1}, s_{c1}, \theta_{j}, \lambda_{j}, \phi_{j} \right)$ which are the design variables for the RCCC linkage (Figure 1). Variable $\mathbf{p}_{j}^{*}$ in Equation (11) represents the precision point and variable $\mathbf{p}_{j}$ represents the coupler point achieved by the synthesized RCCC motion generator using the kinematic displacement model in Section 2.

Equation (11) allows for the direct minimization of the difference between the precision points and the coupler points achieved by the synthesized RCCC motion generator. An objective function having this capability is ideal for path generation since, by definition, the objective in path generation is to replicate precision points as closely as possible.

Along with the objective function, the optimization model includes 6 kinds of equality constraints. Equations (12) through (15) are unity constraints for joint axis vectors $u_{a0}$, $u_{a1}$, $u_{e1}$ and $u_{e0}$ (Figure 1).

$$(u_{a0})^{T} (u_{a0}) - 1 = 0$$

(12)
\[(u_{a_j})^T(u_{a_j}) - 1 = 0\]  \hspace{1cm} (13)

\[(u_{c_j})^T(u_{c_j}) - 1 = 0\]  \hspace{1cm} (14)

\[(u_{e_0})^T(u_{e_0}) - 1 = 0\]  \hspace{1cm} (15)

Equations (16) and (17) are *constant twist constraints*-ensuring that the orientation of joint axis vectors $u_{a_j}$ to $u_{c_j}$ and the orientation of joint axis vectors $u_{c_j}$ to $u_{e_0}$ (respectively) remain fixed throughout RCCC motion.

\[(u_{a_j})^T(u_{c_j}) - (u_{a_j})^T(u_{c_j}) = 0, \quad j = 2 ... N\]  \hspace{1cm} (16)

\[(u_{e_0})^T(u_{c_j}) - (u_{e_0})^T(u_{c_j}) = 0, \quad j = 2 ... N\]  \hspace{1cm} (17)

Equations (18) through (20) are *constant length constraints* for link vectors $a_i - a_0$, $b_i - c_j$, and $d_i - e_j$, respectively-ensuring that the scalar lengths of these vectors remain fixed throughout RCCC motion.

\[(a_j - a_0)^T(a_j - a_0) - (a_j - a_0)^T(a_j - a_0) = 0, \quad j = 2 ... N\]  \hspace{1cm} (18)

\[(b_j - c_j)^T(b_j - c_j) - (b_j - c_j)^T(b_j - c_j) = 0, \quad j = 2 ... N\]  \hspace{1cm} (19)

\[(d_j - e_j)^T(d_j - e_j) - (d_j - e_j)^T(d_j - e_j) = 0, \quad j = 2 ... N\]  \hspace{1cm} (20)

Equations (21) and (25) are *orthogonality constraints*-ensuring that link vectors $a_i - a_0$, $b_i - c_j$, and $d_i - e_j$ are orthogonal to their respective joint axis vectors throughout RCCC motion ($u_{a_j} \perp a_i - a_0$, $u_{b_i} \perp b_i - c_j$, $u_{c_j} \perp c_j - d_j$, $u_{d_i} \perp d_i - e_j$ and $u_{e_j} \perp e_j - e_i$).

\[(u_{a_j})^T(a_j - a_0) = 0, \quad j = 1 ... N\]  \hspace{1cm} (21)

\[(u_{b_j})^T(b_j - c_j) = 0, \quad j = 1 ... N\]  \hspace{1cm} (22)

\[(u_{c_j})^T(b_j - c_j) = 0, \quad j = 1 ... N\]  \hspace{1cm} (23)
\[(\mathbf{u}_{ej})^T(\mathbf{d}_j - \mathbf{e}_j) = 0, \ j = 1 \ldots N \]  \hspace{1cm} (24)

\[(\mathbf{u}_{e_0})^T(\mathbf{d}_j - \mathbf{e}_j) = 0, \ j = 1 \ldots N \]  \hspace{1cm} (25)

Because the displaced value of the joint axis vector \(\mathbf{u}_{e_i}\) can be calculated using Equations (6) and (9), the displacement angles in these equations must be such that both equations produce the same \(\mathbf{u}_{e}\) value. Equation (26) ensures this. Because Equation (26) includes 3x3 spatial rotation matrices, when expanded and separated (into x, y and z-rows), this equation produces three separate constraints in the RCCC optimization model. Equation (10) can be expressed as Equation (27) - a closed loop constraint for the RCCC linkage loop. Like Equation (26), Equation (27) also produces three separate constraints after expansion and separation into x, y and z-rows.

\[
\begin{bmatrix}
R_{\phi_j} \mathbf{u}_{a_j} \\
R_{\theta_j} \mathbf{u}_{a_0} \\
R_{\phi_j} \mathbf{u}_{e_0}
\end{bmatrix}
- \begin{bmatrix}
R_{\phi_j} \mathbf{u}_{a_0} \\
R_{\theta_j} \mathbf{u}_{a_0} \\
R_{\phi_j} \mathbf{u}_{e_0}
\end{bmatrix} = 0, \ j = 2 \ldots N
\]  \hspace{1cm} (26)

\[
s_{a_j} \mathbf{u}_{a_j} + s_{c_j} \mathbf{u}_{c_j} + s_{e_j} \mathbf{u}_{e_j} - \left\{ \left( \begin{bmatrix}
n_j - a_j \\
\end{bmatrix} + \begin{bmatrix}
R_{\phi_j} \mathbf{u}_{e_0} \\
\end{bmatrix} \begin{bmatrix}
\mathbf{d}_j - \mathbf{e}_j \\
\end{bmatrix} - \begin{bmatrix}
R_{\phi_j} \mathbf{u}_{a_0} \\
\end{bmatrix} \begin{bmatrix}
\mathbf{c}_j - a_0 \\
\mathbf{a}_0 - a_j \\
\mathbf{s}_j \mathbf{u}_{a_j}
\end{bmatrix} \right) \right\} = 0, \ j = 2 \ldots N
\]  \hspace{1cm} (27)

The RCCC optimization model also includes two kinds of inequality constraints for branch and order defect elimination [13]. Inequality (28) eliminates branch defects in the RCCC linkage loop (considering the loop \(\mathbf{a}_0 - \mathbf{a}_i - \mathbf{c}_i - \mathbf{f}_0\)) because it ensures a constant cross product direction between vectors \(\mathbf{a}_i - \mathbf{f}_0\) and \(\mathbf{a}_i - \mathbf{f}_0\). A change in branch would result in a change in the direction (and subsequently the sign) of the cross-products. Inequality (29) eliminates order defects because it ensures constant counter-clockwise crank rotation (or clockwise rotation if \(\theta_j < \theta_{j-1}\) is used).

\[
[\begin{bmatrix}
\mathbf{a}_j - \mathbf{f}_0 \\
\mathbf{c}_j - \mathbf{f}_0 \\
\end{bmatrix} \times \begin{bmatrix}
\mathbf{c}_j - \mathbf{f}_0 \\
\mathbf{a}_j - \mathbf{f}_0 \\
\end{bmatrix}] \cdot [\begin{bmatrix}
\mathbf{a}_i - \mathbf{f}_0 \\
\mathbf{c}_i - \mathbf{f}_0 \\
\end{bmatrix}] > 0, \ j = 2 \ldots N
\]  \hspace{1cm} (28)

\[
\begin{cases}
\theta_j > \theta_{j-1} \\
\theta_N < 2\pi 
\end{cases}, \ j = 2 \ldots N
\]  \hspace{1cm} (29)

The displaced RCCC variables \(\mathbf{a}_j, \mathbf{u}_{a_j}, \mathbf{b}_j, \mathbf{c}_j, \mathbf{u}_{c_j}, \mathbf{d}_j\) and \(\mathbf{e}_j\) in the optimization model are
calculated by including their corresponding equations from the RCCC displacement model (Equations (1), (2), (3), (4), (6), (7) and (8) respectively) in Section 2.

This combination of objective functions and equality constraints allows for the direct minimization of the difference between the precision points and the achieved coupler points while calculating the dimensions of the RCCC linkage. The inequality constraints ensure the calculation of RCCC path generator solutions that are free of order and branch defects. Circuit defects are also mitigated since the entire closed-loop RCCC displacement model is incorporated in the optimization model (rather than independently calculating and assembling individual R-C and C-C dyads) (Balli and Chand 2002). For this work, this general RCCC path generation model was implemented in the commercial mathematical analysis software Matlab.

4 EXAMPLE

4.1 RCCC Path Generation: 5 Precision Points

In this example, the constrained optimization model in Section 2 was used to synthesize a defect-free RCCC linkage to approximate an open loop path of 5 precision points. Table 1 includes the spatial Cartesian coordinates for each precision point.

Table 2 includes the initial values and calculated values for the RCCC linkage dimensions. The calculated values in Table 2 were calculated from the optimization model codified in the commercial mathematical analysis software Matlab and solved using the sequential quadratic programming (SQP) method (Constrained Nonlinear Optimization Algorithms 2017). Table 1 includes the coupler points achieved by the synthesized RCCC motion generator (Figure 2). Table 1 also includes the scalar differences between the precision points and the coupler points achieved by the synthesized RCCC linkage. Figure 3 includes the coupler point path traced by the synthesized RCCC linkage and the precision points.

As intended, and as evidenced by the calculated and achieved data in Tables 1 and 2, the synthesized
RCCC motion generator is free of order and branch defects over a full driving link rotation range. Figure 4 includes plots of RCCC displacement angles $\lambda$ and $\phi$ (with respect to the driving link displacement angle $\theta$) over the calculated driving link rotation range. The continuity of these plots confirms that the synthesized RCCC linkage is also circuit defect free over the calculated driving link rotation range.

4.2 RCCC Path Generation: 9 Precision Points

In this example, the constrained optimization model in Section 2 (that was first demonstrated in Section 4.1) was used to synthesize a defect-free RCCC linkage to approximate 9 precision points in a closed loop. The closed loop of precision points includes the five precision points in Section 4.1 with an addition four points (to close the loop). Table 3 includes $p^*$-the (dimensionless) spatial Cartesian coordinates for each precision point.

Table 4 includes the initial values and calculated values for the RCCC linkage dimensions. Table 3 includes $p$-the coupler points achieved by the synthesized RCCC motion generator (Figure 5). Table 3 also includes the scalar differences between the precision points and the coupler points achieved by the synthesized RCCC linkage. Figure 6 includes the coupler point path traced by the synthesized RCCC linkage and the precision points.

As intended, and as evidenced by the calculated and achieved data in Tables 3 and 4, the synthesized RCCC motion generator is free of order and branch defects over a full driving link rotation range. Figure 7 includes plots of RCCC displacement angles $\lambda$ and $\phi$ (with respect to the driving link displacement angle $\theta$) over the calculated driving link rotation range. The continuity of these plots confirms that the synthesized RCCC linkage is also circuit defect free over the calculated driving link rotation range.

5 OPTIMIZATION MODEL IN MATLAB
The RCCC optimization model presented in Section 3 was codified in the commercial mathematical analysis software package Matlab and solved using the SQP method (Constrained Nonlinear Optimization Algorithms 2017). While the RCCC kinematic model and optimization model are given in matrix form in Sections 2 and 3, the full algebraic expansion of these models is required prior to solution calculation in Matlab. This requires substituting Equations (1) through (10) throughout the RCCC optimization model first (for the optimization model variables \(a_j, u_j, b_j, c_j, u_c, d_j, e_j,\) and \(p_j\)) and fully expanding Equations (11) through (27) and Inequality (28) in algebraic form.

Both the gradient and hessian of a constrained optimization model’s objective function and nonlinear constraints are required in the SQP method. If the gradients and Hessians are not explicitly provided by the user in algebraic form, they will be estimated in Matlab. As with any estimation, however, there is a possibility that it will poorly reflect reality. This possibility becomes more probable as the scale of the optimization model grows (which occurs when more precision points are considered). Therefore, the best SQP model gradient and Hessian formulations would be algebraic formulations provided by the user (thus producing the most robust SQP models). However, it is when formulating the Hessians for the objective function and the branch constraint (Equation (11) and Inequality (28) respectively) specifically that the first drawback is realized.

In Matlab, the algebraic hessian for the objective function expanded for nine precision points (\(N = 9\) in Equation (11)) has a file size of 1.62 gigabytes. A nine precision point objective function requires eight inequality constraints for branch elimination (\(j = 2 \ldots 8\) for Inequality (28)) and the algebraic hessian for each of these inequality constraints has a size of 151 megabytes. The total size (considering the equation files, their gradient files and their hessian files) of a complete 9 precision point RCCC optimization model will exceed 5 gigabytes. As the total file size increases, so does the memory requirement for the computer used. A 9 precision point RCCC path generation model was the largest scale possible in Matlab on a 64-bit Windows laptop with 16 gigabytes of memory and a 2.1 gigahertz
processor. Therefore, the combined file size of the optimization model (which places a limit on the number of precision points a user can specify) is its first drawback.

A second drawback is realized when utilizing the optimization model in Matlab. In Section 4.1, a defect-free RCCC path generator was synthesized to approximate 5 precision points. As shown in this example (in Table 2), general initial values were specified to calculate the linkage solution. In Section 4.2, a defect-free RCCC path generator was synthesized to approximate 9 precision points. As shown in this example (in Table 4), more precise initial values were specified to calculate the linkage solution. Therefore, the second drawback with the optimization model is that more precise initial values are required (to enable the model to converge to a solution) as more precision positions are used.

6 DISCUSSION

As the scale of the optimization model increases, so does the difficulty in calculating decent RCCC path generator solutions. For example, the initial guesses in Table 4 were not chosen arbitrarily. After numerous arbitrary selection attempts failed to produce a satisfactory solution for this work, the initial values in Table 4 were determined by drafting RCCC linkages in 3D space (via CAD software). Reducing the RCCC optimization model scale by reducing the number of precision points makes it more practical in Matlab for the SQP method. By reducing the optimization model scale (to \( N = 5 \) in Section 4.1), arbitrary initial guesses produce more satisfactory results and the SQP tolerance settings can remain fixed. However reducing the model scale subsequently reduces the number of precision points allowed-making the model less practical for use in RCCC path generation.

One possible but unexplored option to make the optimization model presented in this work more practical for an indefinite number of precision points is to employ an alternate solution algorithm. Evolutionary Algorithms are algorithms based on a natural selection process that mimics biological evolution. In the context of planar linkage motion generation, the evolutionary algorithm method offers
the simplicity of implementation and fast convergence to the optimal solution with no need of a substantial knowledge of the solution space (Acharyya and Mandal 2009, Cabrera et al 2002).

7 CONCLUSION

A general optimization model for the problem of RCCC path generation with simultaneous order, branch and circuit defect elimination was formulated, implemented and evaluated in this work. With this model, the dimensions of circuit, order and branch defect-free RCCC linkages required to approximate precision positions are calculated. While the primary benefit of the presented optimization model is defect-free optimum RCCC path generation, one drawback (using the SQP method in Matlab on a 64 bit Windows laptop) is the file sizes—particularly of the objective function and branch inequality constraint Hessians. Another drawback is made obvious when utilizing the optimization model for 5 and 9 precision point synthesis. While in a 5 precision point example, a defect-free RCCC path generator was synthesized using general initial values, the initial values specified in a 9 precision point example were very precise. Therefore, as the number of precision points expands, so does the level of involvement in specifying the model initial values and the SQP tolerance parameters in Matlab.

REFERENCES


Cao, Y. and Han, J. 2016. Synthesis of RCCC linkage to visit four given positions based on solution region. Transactions of the Chinese Society for Agricultural Machinery, 47(8): 399-405.


Constrained Nonlinear Optimization Algorithms. Available from


List of Tables

Table 1. Prescribed and achieved rigid-body position coordinates and scalar differences

<table>
<thead>
<tr>
<th>Pos. #</th>
<th>$p^*$</th>
<th>$p$</th>
<th>$p^* - p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.7441, 3.4819, 3.2363</td>
<td>1.7441, 3.4819, 3.2363</td>
<td>0, 0, 0</td>
</tr>
<tr>
<td>2</td>
<td>2.5219, 2.7003, 3.6504</td>
<td>2.5763, 2.8393, 3.7070</td>
<td>0.0544, 0.1390, 0.0566</td>
</tr>
<tr>
<td>3</td>
<td>3.5392, 2.2275, 3.8268</td>
<td>3.5500, 2.2410, 3.8524</td>
<td>0.0108, 0.0135, 0.0256</td>
</tr>
<tr>
<td>4</td>
<td>3.7956, 2.5629, 2.5312</td>
<td>3.7690, 2.4785, 2.5045</td>
<td>0.0266, 0.0844, 0.0267</td>
</tr>
<tr>
<td>5</td>
<td>2, 4, 1</td>
<td>2.0102, 4.0419, 1.0058</td>
<td>0.0102, 0.0419, 0.0058</td>
</tr>
</tbody>
</table>

Table 2. Initial and calculated RCCC linkage dimensions

<table>
<thead>
<tr>
<th>variable</th>
<th>initial values</th>
<th>calculated values</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_0$</td>
<td>5, 5, 5</td>
<td>0.1376, −12.0373, −15.1085</td>
</tr>
<tr>
<td>$u_{a_0}$</td>
<td>0.577, 0.577, 0.577</td>
<td>−0.6623, 0.7436, 0.0919</td>
</tr>
<tr>
<td>$a_1$</td>
<td>1, 1, 1</td>
<td>1.3047, −11.7963, −8.6493</td>
</tr>
<tr>
<td>$u_{a_1}$</td>
<td>−0.577, −0.577, −0.577</td>
<td>0.6860, −0.7216, 0.0935</td>
</tr>
<tr>
<td>$c_1$</td>
<td>4, 4, 4</td>
<td>3.9925, 5.0748, −4.4567</td>
</tr>
<tr>
<td>$u_{c_1}$</td>
<td>−0.577, 0.577, 0.577</td>
<td>−0.4547, 0.7214, −0.5224</td>
</tr>
<tr>
<td>$u_{e_0}$</td>
<td>−0.577, −0.577, 0.577</td>
<td>−0.3261, −0.9443, 0.0449</td>
</tr>
<tr>
<td>$f_0$</td>
<td>1, 5, 10</td>
<td>5.0001, 12.0787, −7.7171</td>
</tr>
<tr>
<td>$s_{a_{1-5}}$</td>
<td>3, 3.5, 3.6, 3.7, 3.8</td>
<td>−9.9377, −8.6260, −7.4068, −8.0662, −11.2543</td>
</tr>
<tr>
<td>$s_{e_{1-5}}$</td>
<td>2, 1.8, 1.6, 1.4, 1.2</td>
<td>3.4088, 4.7991, 5.9805, 4.81, 1.1074</td>
</tr>
<tr>
<td>$s_{e_{1-5}}$</td>
<td>1.5, 1.6, 1.7, 1.8, 1.9</td>
<td>−5.1921, −4.3179, −3.2913, −2.6645, −3.4877</td>
</tr>
<tr>
<td>$\theta_{2-5}$</td>
<td>10°, 20°, 30°, 40°</td>
<td>10.8518°, 26.6826°, 58.2985°, 79.8130°</td>
</tr>
<tr>
<td>$\lambda_{2-5}$</td>
<td>25°, 35°, 45°, 55°</td>
<td>13.9573°, 33.7701°, 70.5941°, 92.6988°</td>
</tr>
<tr>
<td>$\phi_{2-5}$</td>
<td>−5°, −10°, −15°, −20°</td>
<td>−1.5011°, −4.4633°, −11.8831°, −16.8450°</td>
</tr>
</tbody>
</table>
Table 3. Prescribed and achieved rigid-body position coordinates and scalar differences

<table>
<thead>
<tr>
<th>Pos. #</th>
<th>( p^* )</th>
<th>( P )</th>
<th>( p^* - p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2, 4, 1</td>
<td>2, 4, 1</td>
<td>0, 0, 0</td>
</tr>
<tr>
<td>2</td>
<td>0.7214, 5.1329, 1.4157</td>
<td>0.6424, 5.1118, 1.4096</td>
<td>0.079, 0.0211, 0.0061</td>
</tr>
<tr>
<td>3</td>
<td>0.1306, 5.3727, 2.2391</td>
<td>0.1143, 5.3563, 2.2390</td>
<td>0.0163, 0.0164, 0.0001</td>
</tr>
<tr>
<td>4</td>
<td>0.7311, 5.6259, 1.8763</td>
<td>0.5909, 5.5188, 1.8559</td>
<td>0.1402, 0.1071, 0.0204</td>
</tr>
<tr>
<td>5</td>
<td>1.084, 4.6948, 2.3849</td>
<td>1.0439, 4.6381, 2.1888</td>
<td>0.0401, 0.0567, 0.1961</td>
</tr>
<tr>
<td>6</td>
<td>1.7441, 3.4819, 3.2363</td>
<td>1.6268, 3.2695, 3.1188</td>
<td>0.1173, 0.2124, 0.1175</td>
</tr>
<tr>
<td>7</td>
<td>2.5219, 2.7003, 3.6504</td>
<td>2.3659, 2.5245, 3.6453</td>
<td>0.156, 0.1758, 0.0051</td>
</tr>
<tr>
<td>8</td>
<td>3.5392, 2.2275, 3.8268</td>
<td>3.4336, 2.1441, 3.7978</td>
<td>0.1056, 0.0834, 0.029</td>
</tr>
<tr>
<td>9</td>
<td>3.7956, 2.5629, 2.5312</td>
<td>3.7295, 2.5243, 2.449</td>
<td>0.0661, 0.0386, 0.0822</td>
</tr>
</tbody>
</table>

Table 4. Initial and calculated RCCC linkage dimensions

<table>
<thead>
<tr>
<th>variable</th>
<th>initial values</th>
<th>calculated values</th>
</tr>
</thead>
<tbody>
<tr>
<td>( a_0 )</td>
<td>3, 0, 2</td>
<td>4.031, 0.9264, 1.8908</td>
</tr>
<tr>
<td>( u_{a_0} )</td>
<td>0.8, 0.5, 0.1</td>
<td>0.8538, 0.5047, 0.1281</td>
</tr>
<tr>
<td>( a_1 )</td>
<td>3, 0, 0</td>
<td>4.5479, 0.5713, -0.1557</td>
</tr>
<tr>
<td>( u_{a_1} )</td>
<td>-1, 0, 0</td>
<td>-0.9916, -0.0097, -0.1289</td>
</tr>
<tr>
<td>( c_1 )</td>
<td>0, 4, 0</td>
<td>2.3192, 4.2728, -0.6055</td>
</tr>
<tr>
<td>( u_{c_1} )</td>
<td>-0.7, 0, 0.7</td>
<td>-0.6568, 0.03, 0.7534</td>
</tr>
<tr>
<td>( u_{e_1} )</td>
<td>-0.2, -0.6, 0.7</td>
<td>-0.2024, -0.6562, 0.7269</td>
</tr>
<tr>
<td>( f_0 )</td>
<td>1, 4, 5</td>
<td>2.3482, 5.8447, 4.5401</td>
</tr>
<tr>
<td>( s_{a_1-9} )</td>
<td>3, 3, 2, 2, 0.2, -1.3, -1.2, -0.8, -0.4, 1.2</td>
<td>2.2319, 2.8212, 1.9218, -0.9383, -1.509, -0.9834, -0.8587, -1.0147, -0.0024</td>
</tr>
<tr>
<td>( s_{c_1-9} )</td>
<td>1.8, 1.4, 1.5, 1.8, 1.4, 1.4, 1.7, 2.4, 2.4</td>
<td>3.7616, 3.4502, 3.248, 2.2036, 0.8682, 0.5008, 0.9542, 2.336, 3.9099</td>
</tr>
<tr>
<td>( s_{e_1-9} )</td>
<td>2.2, 1.8, 0.5, 0.2, 1.1, 0.6, 0, -0.2, 0.9</td>
<td>0.2171, 0.3054, -0.406, 0.6094, 0.9227, 0.0291, -0.7587, -1.6827, -1.4236</td>
</tr>
<tr>
<td>( \theta_{2-9} )</td>
<td>40°, 80°, 120°, 160°, 200°, 240°, 280°, 320°</td>
<td>40.468°, 79.4864°, 127.294°, 166.0489°, 208.0983°, 242.4815°, 278.9502°, 319.9969°</td>
</tr>
<tr>
<td>( \lambda_{2-9} )</td>
<td>40°, 80°, 150°, -150°, -120°, -100°, -80°, -40°</td>
<td>45.0918°, 89.0663°, 157.2311°, -152.5271°, -120.8654°, -102.6855°, -82.3111°, -46.2434°</td>
</tr>
<tr>
<td>( \phi_{2-9} )</td>
<td>-25°, -35°, -15°, 25°, 45°, 50°, 50°, 30°</td>
<td>-23.8752°, -32.3377°, -8.4225°, 23.6173°, 42.8057°, 49.6101°, 48.6728°, 30.1032°</td>
</tr>
</tbody>
</table>
List of Figure Captions

Fig. 1. RCCC linkage with joint descriptions and displacement variables

Fig. 2. Synthesized RCCC linkage

Fig. 3. Precision points and coupler point path traced by the synthesized RCCC linkage

Fig. 4. Coupler and follower displacement angles (wrt crank displacement angle) for RCCC linkage

Fig. 5. Synthesized RCCC linkage

Fig. 6. Precision points and coupler point path traced by the synthesized RCCC linkage

Fig. 7. Coupler and follower displacement angles (wrt crank displacement angle) for RCCC linkage
Coupler curve traced by synthesized RCCC linkage

217x172mm (300 x 300 DPI)